Quick Guide for using the getStat, getStaDir, and getHsEx FORTRAN codes

Xiaolan L. Wang, Yang Feng, and Rodney Chan
Climate Research Division
Environment and Climate Change Canada

Lorenzo Mentaschi
Joint Research Center

1. Quick Guide for using the getStat and getStatDir codes

1.1 Introduction

This set of FORTRAN codes (getStat.f, getStatDir.f and ecnetcdf.f) is designed to calculate the COWCLIP standard set of statistics from daily or sub-daily input wave data (significant wave height, mean wave period, mean wave direction, swell wave height). Both the input wave data and the output statistics are in netCDF format. The time definition in the NetCDF file should be relative values, in the following form:

```
double time(time) ;
time:standard_name = "time" ;
time:units = "hours since YYYY-MM-DD HH:MM:SS";
  (e.g., "hours since 1950-7-1 00:00:00";)
time:calendar = "Type" ;
  (Here, the "Type" could be "Gregorian" or
                             "standard" (i.e. Gregorian) or
                             "365_day" or "no leap" or
                             "360_day")
time:axis = "T" ;
```

The COWCLIP standard is to calculate the statistics from 6-hourly data. But this set of codes can be used to calculate the statistics from daily or other sub-daily (e.g., 3-hourly or hourly) wave data. The temporal resolution of the input data is for the user to choose.

The getStat.f and getStatDir.f are Fortran90 programs for calculating statistics of the wave data chosen by the user. The difference between the two is that getStat estimates scalar statistics valid for scalar variables (like significant wave height or mean period); while getStatDir estimates circular statistics valid for directional variables like the mean direction.

The getStat.f is for calculating the following 7 statistics from an input wave data chosen by the user:

avg – the mean
p10 – the 10th Percentile
p50 – the 50th Percentile
p90 – the 90th Percentile
p95 – the 95th Percentile
p99 – the 99th Percentile
max – the maximum

In order to calculate these statistics for the COWCLIP-required 17 time-frames [12 monthly, 4 seasonal (DJF, MAM, JJA and SON), and an annual value], users need to run the program three times, one for each of the following 3 target time-frame resolutions:

MLY - for monthly statistics
SNL - for seasonal statistics
ANL - for annual statistics

1.2 Setup

Users are required to compile the getStat.f with their choice of Fortran90 compiler. A netCDF4 library is also required. Once these requirements are met, first, use your Fortran90 compiler and netCDF4 library to compile ecnetcdf.f with the -c option, for example,

```
pgf90 ecnetcdf.f -lnetcdf -lhdf5_hl -lhdf5 -lz -c
```

Then, compile the getStat.f together with the just compiled ecnetcdf.o, for example,

```
pgf90 getStat.f -lnetcdf -lhdf5_hl -lhdf5 -lz ecnetcdf.o -o getStat
```

For ifort compiler:

```
ifort -lnetcdf -lnetcdff -lhdf5 -lhdf5_hl -lz ecnetcdf.f getStat.f –o getStat
```

If the library cannot be found by the Fortran90 compiler, you may want to try the -L and -I options, for example,

```
pgf90 ecnetcdf.f -lnetcdf -lhdf5_hl -lhdf5 -lz -c
-I/your/dir/path/netcdf4/include -L/your/dir/path/netcdf4/lib

pgf90 getStat.f -lnetcdf -lhdf5_hl -lhdf5 -lz ecnetcdf.o
```

```
-I/your/dir/path/netcdf4/include -L/your/dir/path/netcdf4/lib -o getStat
```

For ifort compiler:

```
ifort -I/your/dir/path/netcdf4/include -L/your/dir/path/netcdf4/lib -lnetcdf -lnetcdff -lhdf5
-lhdf5_hl -lz ecnetcdf.f getStat.f -o getStat
```

1.3 Usage

Use the following command to use getStat.f:

```
getStat -i input_filename -o output_filename
        -v variable -s output_time_frame
```

Here, all input arguments are required: The input_filename and (output_filname) is the filename of the input (output) dataset in netCDF format; the variable is the target variable name (same as used in the netCDF input dataset); the output_time_frame is MLY for calculating monthly statistics, SNL for calculating seasonal statistics, and ANL for calculating annual statistics. For example, use the following to calculate monthly statistics of Hs (significant wave heights):

```
getStat -i input.nc -o output.nc -v Hs -s MLY
```

Users should include the time-frame of the statistics and the temporal resolution of the input data in the output filename, such as

```
_mlystat_from_6hly_ (monthly statistics from 6-hourly data)
_snlstat_from_6hly_ (seasonal statistics from 6-hourly data)
_anlstat_from_6hly_ (annual statistics from 6-hourly data)
_mlystat_from_3hly_ (monthly statistics from 3-hourly data)
_snlstat_from_dly_ (seasonal statistics from daily data)
…
```

Note that if a grid point has one missing value in the target period (e.g., an ice-covered grid point in January 1990), the statistics for that grid point are set to missing for that target period (e.g., January 1990).

1.4 Input and output files

The input file must be in netCDF format with time as the unlimited dimension.

The output file will be in netCDF format, which will retain the relevant metadata of the input netCDF file and the coordinate variables.

The time variable will be updated with the new chosen time-frame (monthly, seasonal, or annual). The only non-coordinate variables will be the newly calculated statistics. The names of the new variables will be the target variable with a suffix of the statistics, for example, if the target variable is Hs, then the statistics will be

Hs_avg, Hs_p10, Hs_p50, Hs_p90, Hs_p95, Hs_p99, Hs_max

1.5 getStatDir

The program getStatDir.f is analogous to getStat.f, but elaborates circular statistics meaningful for angular measures, such as the mean direction. The input variables are assumed to be expressed in degrees. The output parameters are:

avg – the circular mean
std – the circular standard deviation

The compilation of getStatDir.f can be achieved following the instructions valid for getStat (section 1.2), but applying them to file getStatDir.f instead of file getStat.f. Like getStat.f, getStatDir.f can produce monthly, seasonal and annual statistics, and the command line to run it is the same as indicated in section 1.3. Similarly to getStat.f (section 1.4), getStatDir.f produces output netcdf files with a time step consistent with time frame chosen for the statistics, and with two variables

<varname>_avg, <varname>_std ,

where <varname> is the name of the angular variable selected with the –v parameter.

2. Quick Guide for using the getHsEx codes

2.1 Introduction

This set of Fortran90 codes (getHsEx.f and ecnetcdf.f) is designed to calculate the WMO/WCRP/JCOMM joint Expert Team on Climate Change Detection and Indices (ETCCDI) set of extreme wave height indices (tabled below) from daily maximum or sub-daily data of significant wave height (Hs). The HsMx and HsAx are not included in the calculation because they are already calculated by calling the getStat.f. Thus, there is only one target time-frame resolution here, which is annual.

The ETCCDI set of extreme significant wave height indices:

ID	Indicator name	Definitions	Units
HsMx	Monthly max Hs	Monthly maximum value of significant wave height (Hs)	m
HsAx	Annual max Hs	Annual maximum value of significant wave height (Hs)	m
HsRo	Rough wave days	Annual count of days when daily max Hs > 2.5 m	days
HsHi	High wave days	Annual count of days when daily max Hs > 6 m	days
fHsRo	Frequency of rough wave days	Annual percentage of days when daily max Hs > 2.5 m	%
fHsHi	Frequency of high wave days	Annual percentage of days when daily max Hs > 6 m	%
fHs90p	Frequency of top decile wave days	Annual percentage of days when daily max Hs > 90th percentile of daily max Hs in the base period	%
fHs10p	Frequency of low decile wave days	Annual percentage of days when daily max Hs < 10th percentile of daily max Hs in the base period	%
HHsDI	Top decile wave spell duration indicator	Annual count of days with at least 2 consecutive days when daily max Hs > 90th percentile of daily max Hs in the base period	days

2.2 Setup

The getHsEx.f, and getStat.f share the same netCDF input data format (see Section 1.1 above). Users are required to compile the getHsEx.f with their choice of Fortran90 compiler. A netCDF4 library is also required. Once these requirements are met, first,

use your Fortran90 compiler and netCDF4 library to compile ecnetcdf.f with the -c option, for example,

```
pgf90 ecnetcdf.f -lnetcdf -lhdf5_hl -lhdf5 -lz -c
```

Then, compile the getHsEx.f together with the just compiled ecnetcdf.o, for example,

```
pgf90 getHsEx.f -lnetcdf -lhdf5_hl -lhdf5 -lz ecnetcdf.o -o getHsEx
```

If the library cannot be found by the Fortran90 compiler, you may want to try the -L and -I options, for example,

```
pgf90 ecnetcdf.f -lnetcdf -lhdf5_hl -lhdf5 -lz -c
-I/your/dir/path/netcdf4/include -L/your/dir/path/netcdf4/lib

pgf90 getHsEx.f -lnetcdf -lhdf5_hl -lhdf5 -lz ecnetcdf.o
-I/your/dir/path/netcdf4/include -L/your/dir/path/netcdf4/lib -o getHsEx
```

2.3 Usage

The ETCCDI standard is to calculate the extreme indices from daily extreme values, which can be derived from sub-daily data (such as hourly, 3-hourly or 6-hourly). However, this set of codes can be used to calculate the extreme indices from daily maximum or sub-daily wave data. The temporal resolution of the input data is for the user to choose; but if it is daily, it must be daily maximum values (in this case, the sub-daily resolution used to calculate the daily maxima should be specified if possible).

Use the following command to use the getHsEx.f:

```
getHsEx -i input_filename -o output_filename -v variable
        -bs BP_startYear -be BP_endYear  -A scale
```

The required arguments, input_filename and output_filename, are the filename of the input and output dataset, respectively. Both the input wave data and the output statistics are in netCDF format. The required argument variable is the target variable name, which is Hs (significant wave height) here.

The optional arguments BP_startYear and BP_endYear are the first and last year of the chosen base period, which is needed to

define the extreme indices. If BP_startYear is not specified, the program will use the first year of complete data as the first year of the base period. If BP_endYear is not specified, the program will use the last year of complete data as the last year of the base period.

The optional argument scale is the factor for converting the unit of the input wave height data to meter. If it is not specified, it takes the default value of 1, assuming the unit of the input Hs is meter. The scale should be 0.01 if the unit of the input Hs is centimeter.

Users should include the extreme indices time-frame (annual) and the temporal resolution of the input data in the output filename, such as

_anlEx_from_6hly_ (annual extreme indices from 6-hourly data)
_anlEx_from_3hly_ (annual extreme indices from 3-hourly data)
_anlEx_from_hly_ (annual extreme indices from hourly data)
_anlEx_from_dlymax_ (annual extreme indices from daily maxima; specify the resolution of data used to derive the daily maxima if possible)
…

The following are the rules used to treat missing values. A daily maximum value is derived if there are two or more non-missing sub-daily values for that day. The annual extreme indices for a year are set to missing if there more than 15 missing daily maximum values in that year (same as the ETCCDI RClimDex standard for treating missing values). The percentile values of the base period are set to missing if more than 20% of the daily maximum values in the base period are missing.

2.4 Input and output files

The input file must be in netCDF format with time as the unlimited dimension.

The output file will be in netCDF format, which will retain the relevant metadata of the input netCDF file and the coordinate variables.

The time variable will be updated with the new time-frame, which is annual. The only non-coordinate variables will be the newly calculated extreme indices of Hs:

HsRo, HsHi, fHsRo, fHsHi, fHs90p, fHs10p, HHsDI