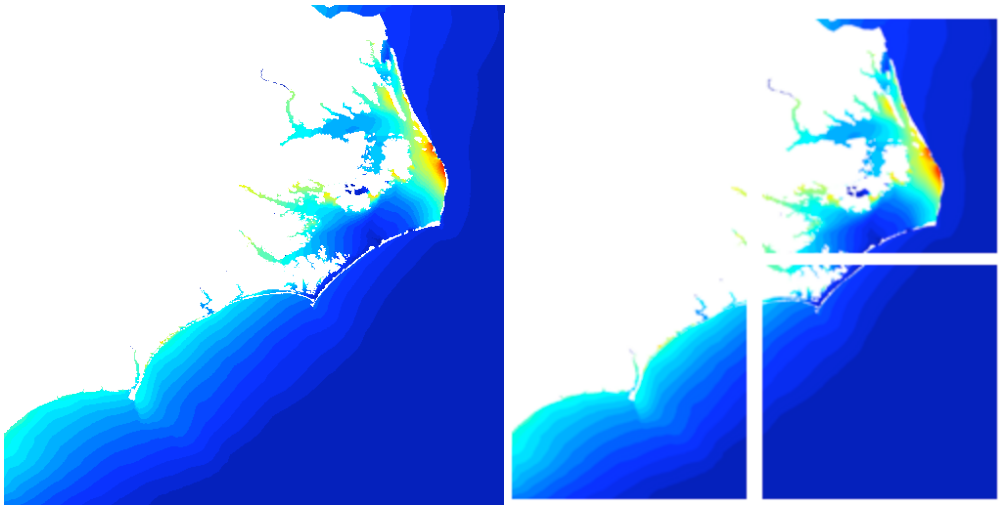
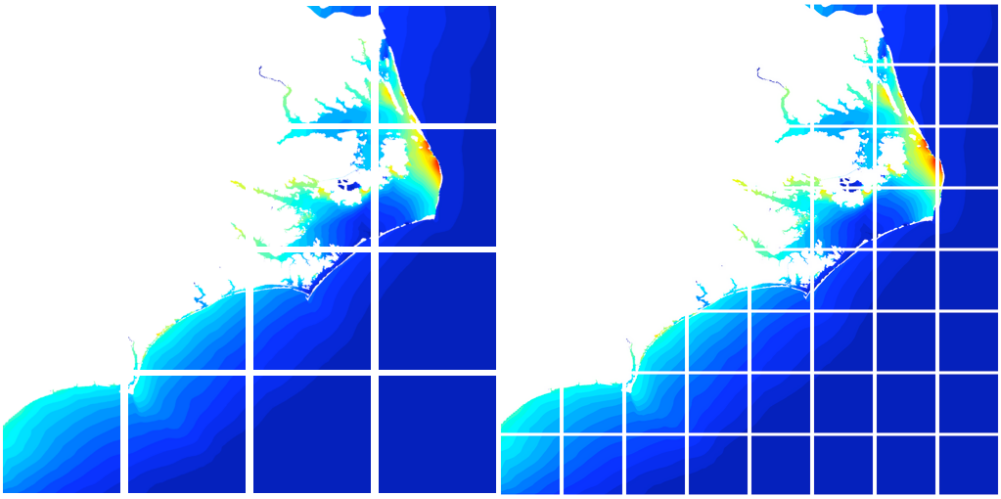
Purpose: generate tiled images of ADCIRC solutions for display in Google Earth.

Abstract:

The code adc_max_simple_plot_gmt.sh generates tiled images from an ADCIRC solution file (.63) that contains only ONE time step; i.e., the maxele.63 file. [This will be generalized in the future.] . The tiling is done in powers of 2, such that each level of tiles (n=1:nLevels) contains 2^(2*(n-1)) tiles. E.g., n=1:4 yields p=1, 4, 16, 64 tiles, which gives the levels shown below. The code also writes the kml files needed to overlay the levels in Google Earth, and compresses the contents into a kmz file.





) **Requirements:**
GMT 4.2+
ImageMagick
Ghostscript
gcc
netcdf if you want to process NetCDF files.

) **Safeguards:**

Make sure that no .gmtdefaults4 file exists in your home directory. We have found that this creates havoc.

) **Setup:**

Define the top-level (1) viewing window in adc_max_simple_plot_gmt.sh. The tile resolution is 512x512, so make sure this region is square.

```
    # View limits:
    NORTH=37
    SOUTH=33
    WEST=-79
    EAST=-75
```

) **Usage:**

```
$PPDIR/adc_max_simple_plot_gmt.sh -f fortname -g grid -p prefix
       -n nLevels [-d startDate] [-o outputmode] [-k kmzPrefix] [-m mode]
       [-c colorrange] [-u units]

where the arguments are as follows:

-f: The name of the fort.63 file to visualize.
-g: The name of the gmt versions of the relevant ADCIRC grid.
-p: The prefix for all artifacts including the resulting KMZ file.
    You can specify the prefix of the kmz file separately using -k
    but you probably don't want to: that functionality is provided
    for the use of other programs.
-n: The number of levels of plots in the KMZ files. See
    TiledREADME.pdf for more details.
-d: The starting date of the simulation being visualized. This
    is in yyyy-mm-ddThh:mm:ss. Setting a date greater than -1
    causes a timestamp to be added to the kml file. If you are
    producing a single kmz file using "both" (the default) as the output
    mode, omit this parameter.
-o: The output mode: this controls what this script actually produces.
      The modes are:
        both: produce both the colormap and the complete kmz. Default
        colormap: produce only the colormap
        kmz: produce the kmz, including a previously produced colormap
             This is used when multiple kmzs are sharing a color map
-k: The prefix for the kmz file produced. The default is
    the prefix specified with the -p flag
-m: The mode for the fort.63 file.  Possible values are
       full: Full ascii format. All nodes are in the file.
       netCDFScoop: NetCDF SCOOP format.
       compact: Currently unsupported.Compact ascii format. Only nodes with data are in the file.
    the default is "full".
-t: The timestep in the fort.63 file to visualize. This is
    only supported with netCDF format fort.63 files. All other fort.63 files
    must contain a single time step.
-c: The colon seperated min max values to use for the colormap. For
    example -c 0:2 will produce a colormap between 0 and 2 units. If this
    is not specified a color map is generated automatically using the minimum
    and maximum of the data values of the grid nodes contained within the view limits.
-u: The units for the display.  Must be either M (meters) or
    FT for feet.  Default is meters.

Note that you do not need to execute the command in $PPDIR.  Output files will consist of a kmz file named prefix.kmz plus one "montage"
file for each requested level.  If you set nLevels to 3 you will get 3 montage files named prefix-montage.{1-3}.png
```

) **Installation:**

- Unpack the tar file. You should see the following set of subdirectories in the RenciGETools-releasenumber directory:

```
doc: contains documents including this file.
examples: contains some usage examples.
grids: contains the gmt format grid files we are including with this release.
src: contains all of the source codes for the release. This is the directory in which you have
     to build all the source codes and edit the config_simple_gmt_pp.sh file.
```

- Define location of GMT,GRD files, etc installation in config_simple_gmt_pp.sh.
  See examples in the "SYS DIRS" part of the file:

```
#----------- SYS DIRS -------------
#----------- SYS DIRS -------------
ImageMagick=<Location in ImageMagick binaries>
ZIP=/usr/bin/zip
PPDIR=<Location of this script>
GRDFILES=<Location of gmt versions of ADCIRC grid files>
GMTHOME=<Location of GMT installation>
GS=<Location of ghostscript interpretor (gs)>
```

- export the directory PPDIR to the shell:

```
In bash:
export PPDIR=<Location of this script>

In csh:
setenv PPDIR <Location of this script>
```

- Compile kml file writer

```
gcc WriteTiledKML.c -o WriteTiledKML
```

- Compile the minimum/maximum finder.

```
If you want to use NetCDF files:

gcc -Wall -DUSE_NETCDF -o FindMax FindMax.c -I/opt/netcdf/3.6.2/include -L/opt/netcdf/3.6.2/lib -lnetcdf

Be careful to replace all instance of /opt/netcdf/3.6.2 with the location of your NetCDF install.

Without NetCDF:

gcc -Wall -o FindMax FindMax.c
```

- Compile the gmt grd file builder

```
To compile on a Mac with g95 :
```

```
g95 -lSystemStubs grd2gmt.f90 -O4 -o grd2gmt.x

To compile using gfortran you need to comment out the line

INTEGER, EXTERNAL :: iargc

These are the only compilers we have tried.
```

- Make the gmt versions of the adcirc grid

```
Usage: grd2gmt.x <grdfilename> <gridname>
Example: ./grd2gmt.x lawest_2007_r10.grd lawest_2007_r10
Example: ./grd2gmt.x fort.14 lawest_2007_r10

The gmt versions of the grid only need to be built once, and placed into the directory $GRDFILES. By default, $GRDFILES is equal to
$PPDIR, but you can set that in config_simple_bmt_pp.sh as desired.
The files are loaded by gmt by the <gridname> used to make the files.
```

) **Example files:**

We have included a sample maxele.63 from Hurricane Hanna. We have also included the gmt grid files for the nciv6a grid, the nciv6a grid file (nc_inundation_v6a.grd) as well as some sample outputs. The outputs and the maxele file are in the examples sub directory: the gmt grid files are in the main source code directory because on our machine we have set GRDFILES equal to PPDIR. Feel free to move things around.

The outputs were generated with the following command:

```
[howard@tornado:/data/ncfmp/ncfmp/tools/graph/RenciGETools/RenciGETools-1.0/examples/simple_plot] $ $PPDIR/adc_max_simple_plot_gmt.sh -f
maxele.63 -g nciv6a -p example -n 3
```

and resulted in the following file listing:

```
[howard@tornado:/data/ncfmp/ncfmp/tools/graph/RenciGETools/RenciGETools-1.0/examples/simple_plot] $ ll
total 8.9M
-rw-rw-r--  1 howard renci   86 Oct 14 14:57 example_command.sh
-rw-rw-r--  1 howard renci 221K Oct 14 11:42 example.kmz
-rw-rw-r--  1 howard renci  20K Oct 14 11:40 example-montage.1.png
-rw-rw-r--  1 howard renci  56K Oct 14 11:40 example-montage.2.png
-rw-rw-r--  1 howard renci 146K Oct 14 11:41 example-montage.3.png
-rw-r--r--  1 howard renci 8.5M Oct 14 10:54 maxele.63
```

Note that in this example, we have accepted many of the defaults: in particular our maxele.63 file is in the "full" format. Since the initial release of this code only supported the full format, this command line is the equivalent of the "status quo" of the previous release.

) **Making GoogleEarth animations from fort.63 files**

In addition to adc_max_simple_plot_gmt.sh, this release includes a perl script that can assemble GE animations from fort.63 files. It's a new capability, so don't be surprised to find bugs in it.

) **Installation**

If you want to make movies from NetCDF files, you need to compile the NetCDF GMT contour file builder:

```
gcc -Wall -o BuildContourFile BuildContourFile.c -I/opt/netcdf/3.6.2/include -L/opt/netcdf/3.6.2/lib -lnetcdf

Be careful to replace all instance of /opt/netcdf/3.6.2 with the location of your NetCDF install.
```

If you want to make movies from either full or compact format ascii fort.63 files, you need to compile the ascii contour file builder.

```
gfortran -o splitFort63 splitFort63.f
```

) **Usage**

```
MakeGEMovie.pl  --dataFile=fort.63 --grid=gridName --prefix=prefix
                --nLevels=nLevels --startDate=startDate --firstStep=firstStep
                --nSteps=nSteps --stepInterval=stepInterval [--outputDir=dir]
                [--fort63Mode=mode] [--colorRange=colorRange] [--units=units]

This function reads an ADCIRC fort.63 formatted file and converts it into
a GoogleEarth KMZ file.  It does this using a combination of Fortran, C and sh
script codes. Each timestep of the fort.63 file is visualized individually.
Because we are using GMT to produce the plots, you have to have the GMT
versions of the grid on which the fort.63 files were produced. See the file
TiledREADME.pdf in this directory for details on how to make these.

The arguments are

    --dataFile:  The name of the fort.63 file to visualize.
    --fort63Mode: Supported modes are:

        full: The data values are contained in the fort.63 formatted
              file specified by the --dataFile parameter.  All nodes are
              represented in the file. The nodes composing the grid
              are contained in the file specified by the --gridFile parameter

        compact: The data values are contained in the fort.63 formatted
                 file specified by the --dataFile parameter.  Nodes without
                 valid data are not represented in the file. The nodes composing
                 the grid are contained in the file specified by the --gridFile
                 parameter.

        netCDFScoop: the grid and value data are in a netCDF file that
                     conforms to that used in the ADCIRC implementation of
                     the SURA Coastal Ocean Observing and Prediction (SCOOP)
                     Program.
    --grid: the name of the gmt versions of the relevant ADCIRC grid. Not used if the fort63Mode is netCDFScoop.
    --prefix: the prefix for the resulting KMZ file.
```

```
    --nLevels: the number of levels of plots in the KMZ files. See
               TiledREADME.pdf for more details.
    --startDate: the starting date of the simulation being visualized. This
                 is in yyyymmddhhmmss format.
    --firstStep: the first timestep of the simulation to be visualized.
    --nSteps: the number of timesteps of the simulation to be visualized.
    --stepInterval: the intervals of the steps to be visualized: 1 means
                    visualize every step, 2 means every other step ...
    --outputDir: the directory in which to deposit the artifacts.
    --colorRange: The colon seperated min max values to use for the colormap.
                  For example --colorRange 0:2 will produce a colormap
                  between 0 and 2 units.
    --units: The units for the display.  Must be either M for meters or
             FT for feet. Default is M.

Note that this call depends on the value of the environment var PPDIR.
Set it to the dir that contains the executables...
```

) **Examples**

) **Known Issues**

adc_max_simple_plot_gmt.sh does not accept compact format fort.63 files.
The MakeGEMovie script will accept compact format fort.63 files, but will fail if the file contains only one time step.
At least on some machines, the speed of the movies is too fast.
In some cases, the file zeta.cpt is not deleted.
The software lacks an easy build/installation method.

) **Questions:**

Questions should be directed to howard@renci.org