

Chapter 11

Netcdf and climate data operators

11.1 NetCDF

NetCDF (Network Common Data Form) is a set of software libraries and self-describing, machine-independent data formats that support the creation, access, and sharing of array-oriented scientific data. The project homepage is hosted by the Unidata program at the University Corporation for Atmospheric Research (UCAR). They are also the chief source of netCDF software, standards development, updates etc. The format is an open standard.

The software libraries supplied by UCAR provide read-write access to netCDF files, encoding and decoding the necessary arrays and metadata. The core library is written in C, and provides an API for C, C++ and Fortran applications. An independent implementation, also developed and maintained by Unidata, is written in 100% Java, which extends the core data model and adds additional functionality. Interfaces to netCDF based on the C library are also available in other languages including R (ncdf and nvar packages), Perl, Python, Ruby, Matlab, IDL, and Octave.

A wide range of application software has been written which makes use of netCDF files. These range from command line utilities to graphical visualization packages.

- A commonly used set of Unix command line utilities for netCDF files is the NetCDF Operators (NCO) suite, which provide a range of commands for manipulation and analysis of

netCDF files including basic record concatenating, slicing and averaging.

- **NcBrowse** is a generic netCDF file viewer that includes Java graphics, animations and 3D visualizations for a wide range of netCDF file conventions.
- **ncview** is a visual browser for netCDF format files. Typically you would use ncview to get a quick and easy, push-button look at your netCDF files. You can view simple movies of the data, view along various dimensions, take a look at the actual data values, change color maps, invert the data, etc.
- **Panoply** is a netCDF file viewer developed at the NASA Goddard Institute for Space Studies which focuses on presentation of geo-gridded data. It is written in Java and thus platform independent. Although its feature set overlaps with ncBrowse and ncview, Panoply is distinguished by offering a wide variety of map projections and ability to work with different scale color tables.
- The **NCAR Command Language** is used to analyze and visualize data in netCDF files (among other formats).
- **PyNIO** is a Python programming language module that allows read and/or write access to a variety of data formats, including netCDF.
- **Ferret** is an interactive computer visualization and analysis environment designed to meet the needs of oceanographers and meteorologists analyzing large and complex gridded data sets. Ferret offers a Mathematica-like approach to analysis; new variables may be defined interactively as mathematical expressions involving data set variables. Calculations may be applied over arbitrarily shaped regions. Fully documented graphics are produced with a single command.
- **nCDF-Browser** is a visual nCDF browser, written in the IDL programming language. Variables, attributes, and dimensions can be immediately downloaded to the IDL command line

for further processing. All the Coyote Library files necessary to run nCDF-Browser are available in the zip file.

- ArcGIS version 9.2 supports netCDF files. The Multidimension Tools toolbox can be used to create raster layers, feature layers, and table views from netCDF data in ArcMap, or convert feature, raster, and table data to netCDF.
- Origin 8 imports netCDF files as matrix books where each book can hold a 4D array. Users can select a subset of the imported data to make surface, contour or image plots.
- The Geospatial Data Abstraction Library provides support for read and write access to netCDF data.

Exercise netcdf

In this exercise we will learn to know various helpful tools and techniques that are used in the analysis and processing of climate data. First, we will learn to know the NetCDF file format in a few short practical demonstrations. Second, some data analysis will be performed on available gridded climate data. In a third step we will transfer our knowledge to shell-programming, in that we combine a number of common command-line tools in order to solve a simple scientific problem. To this end, a Linux-shell, in combination with common shell-programs, is necessary. Since not every attendee of this course has such a computing environment available, the shell-program will be demonstrated by the tutor.

Topics that this exercise considers are:

- **Network Common Data Form (NetCDF)**: general information on the topic and how to use NetCDF files
- **Climate Data Operators (CDO)**: several examples of data reduction, analysis and transformation of NetCDF files, including operator piping

- **Bourne-again shell** (Bash): definition of variables, if-then-else construct, checking for the existence of a specific file, for-counter-loops, integer-arithmetic, pipes, initializing of shell variables with program output, generation of strings via variable concatenation; these programming-methods are illustrated at the example of a simple scientific model of sea-level rise
- **Stream EDitor** (SED): removing spaces from a string
- **basic calculator** (bc): simple floating-point arithmetic

In the following some information regarding tools and methods of this exercise is collected in the form of a very general overview. This information collection is intended as a supplement to the lecture. Few further information sources are listed in the subsection “Further Reading”. This list is obviously far from being complete, there is a vast amount of freely-accessible information available on-line.

A very precise definition on the characteristic and purpose of NetCDF is given in the NetCDF FAQ, “What is netCDF?” (for a link, see section “Further Reading”):

“NetCDF (network Common Data Form) is a set of interfaces for array-oriented data access and a freely distributed collection of data access libraries for C, Fortran, C++, Java, and other languages. The netCDF libraries support a machine-independent format for representing scientific data. Together, the interfaces, libraries, and format support the creation, access, and sharing of scientific data.”

NetCDF is a data-container that has been established as a widely used file-standard in science and engineering. It has been developed for storing array-oriented values in compact and interchangeable files. The most important characteristics of NetCDF files can as well be found in the NetCDF FAQ, “What is netCDF?”. An excerpt, that highlights the advantages of NetCDF with respect to this exercise, is listed here:

- **Self-Describing.** A NetCDF file includes a description of the data that it contains.

- **Portable.** A NetCDF file can be accessed by computers that apply different formats of storing integers, characters, and floating-point numbers.
- **Scalable.** A small subset of a large dataset may be accessed efficiently.
- **Appendable.** Data may be appended to a properly structured NetCDF file without copying the dataset or redefining its structure.

These characteristics make NetCDF a perfect choice for storing any kind of array-oriented data. The data form that we will work with in this exercise is as well array-oriented - therefore, we will use NetCDF as data container for both input and output of computations.

Since NetCDF is a binary format (in contrast to ASCII-text, which can be examined and edited by means of any common text editor), reading, writing and changing of NetCDF files necessitates the use of dedicated software. Fortunately, such software is freely available and can easily be installed on any UNIX system. The following tools are of particular importance:

- `ncview` (lightweight but mighty explorer for NetCDF files)
- `ncbrowse` (a Java-based alternative to `ncview` for Windows)
- `Panoply` (a flexible Java-based generator of geographic maps of NetCDF data)
- `ncdump` (tool for “dumping” the contents of a NetCDF file to human-readable ASCII-text; the complete description and structure of the NetCDF file is preserved)
- `ncgen` (complementary to `ncdump`, generates a binary NetCDF file from a NetCDF ASCII-dump)
- `ncks` (mighty toolbox for modification of NetCDF files)
- `cdo` (mighty toolbox for analysis and modification of NetCDF files, strong focus on climatological data)

11.2 The Bash, a popular UNIX-Shell

Shells with their scripting ability are probably the most powerful tool of UNIX-systems, which make UNIX-computers the choice for tackling complex scientific problems that involve the analysis and processing of large amounts of data. Particularly the bash-shell is a famous tool for scientists and commonly used during the daily work routine. Yet, giving a comprehensive overview on the use and ability of the bash clearly exceeds the scope of this course. Please refer to a bash-scripting guide referenced below if you would like to gain further insights into the topic and gain abilities in shell-scripting - and note: “... the only way to really learn scripting is to write scripts” (Advanced Bash-Scripting Guide).

Further Reading

The following resources provide an introduction to tools and methods considered in this exercise:

- NetCDF: The NetCDF FAQ
(www.unidata.ucar.edu/software/netcdf/docs/faq.html)
The NetCDF Fact Sheet
(http://www.unidata.ucar.edu/publications/factsheets/current/netcdf_factsheet.pdf)
- CDO: The CDO User’s Guide
(<https://code.zmaw.de/projects/cdo/embedded/1.6.3/cdo.html>)
The CDO Reference Card
(http://www.iac.ethz.ch/edu/courses/master/modules/radiation_and_climate_change/download/cdo_refcard.pdf)
- Bash: Bash Guide for Beginners
(<http://www.tldp.org/LDP/Bash-Beginners-Guide/html/>)
Advanced Bash-Scripting Guide
(<http://www.tldp.org/LDP/abs/html/>)

Practical exercises for UNIX, cdo, netcdf

For those of you who consider to work in a scientific field with a strong focus on programming or the analysis of large amounts of data: Consider to gain experience with a UNIX-environment, e.g. Ubuntu. In many scientific fields, definitely in climate sciences, UNIX-like operating-systems are the computing environments of choice. Many tools that are necessary for efficiently working in such scientific fields are not natively available on Windows-systems; even if ports are available, they may still suffer from limitations or incompatibilities. It definitely makes sense for you to setup an own partition on your laptop with a Linux-system, and to learn how to use and program the available software tools. For every free UNIX-environment (e.g. Ubuntu) and the included software tools, a vast amount of detailed, yet free, documentation, addressing both novices and experts, can be found on-line.

Visualizing the content of a NetCDF file

Among the provided data files you find a gridded global distribution of sea-ice concentration (variable seaice) and surface temperature (variable tsurf) retrieved from a simulation with a comprehensive climate model (file INIOM_PD_3901-4000_tsurf_seaice.nc). NetCDF is a binary data format that cannot be directly visualized with a simple text editor, special software exists for this purpose. Windows-users please use the program **ncbrowse**, linux-users may alternatively use the native Linux-tool **ncview**.

Please open the NetCDF file and visually inspect the content of variable tsurf. Try to generate an animation of the time evolution of tsurf and answer the following questions:

- What obvious time-dependent pattern is visible?
- How can this pattern be explained, considering that you see the time evolution of a global climatological field?

Creating a PDF that visualizes a regional selection of the content of a NetCDF file

Now please visualize a global distribution of seaice also found in the provided NetCDF file INIOM_PD_3901-4000_tsurf_seaice.nc. The software to be used for this task is **panoply**. Create two plots of Arctic sea-ice distribution, from 45°N to the North Pole, for March and September of model year 3911, 18:00:00. Apply a meaningful colorbar and export the figure to PDF. Answer the following questions:

- Where do you identify the southernmost extent of sea-ice in boreal winter?
- Where is the sea-ice retreat in summer, by visual inspection, most prominent?

Hint: You can create a map centered on the North Pole by applying a stereographic projection.

Identifying the spatial resolution and physical unit of a NetCDF data set

By use of the tool **ncdump** it is possible to generate a human-readable description of a NetCDF file, that may also include the full data record. Sometimes you are only interested in the header information, and do not want the data record to be extracted (extracting the data record to ASCII can lead to HUGE dumps). If you have ncdump available, extract only the NetCDF data header of file INIOM_PD_3901-4000_tsurf_seaice.nc. This can be done in a Linux shell-terminal via entering the following command:

```
ncdump -h INIOM_PD_3901-4000_tsurf_seaice.nc > headerdump.nc
# the parameter '-h' forces ncdump to omit the extensive data record
# the character '>' is an output-redirection command, that makes
# the output produced by ncdump is written to a file (headerdump.nc)
```

Open the resulting file, or alternatively the file INIOM_PD_39010131_tsurf_seaice.nc.dump that has been provided to you, in an arbitrary text-viewer or -editor, analyze the file content, and answer the following questions:

- What is the horizontal data resolution in units of degrees? Hint: Calculate the meridional (latitudinal) and zonal (longitudinal) resolution of the data set from the number of longi-

tudes and latitudes. Assume that grid cells are equally spaced, and the data set has a global coverage.

- What is the physical unit of variable `tsurf`?
- If you analyze the header dump of the file `INIOM_PD_3901-4000_tsurf_seaice.nc`: What is the time resolution of the data? Assume that the complete data set covers a time span of 100 calendar years.

11.3 Reducing data sets with climate data operators

While NetCDF defines a file format (and supporting programs and routines) that can store climatological data in a practical way, the CDO are a collection of operators that allow analysis and modification of gridded binary climatological data. In climate sciences, the CDO have become a very common software tool due to the vast number of available operators and their flexibility:

- more than 400 designated operators are available
- operator-piping allows the application of complex methods on climatological data in a compact way
- the CDO are command-line programs; in combination with shell-scripts they can be automated, and enhanced complexity of data processing and analysis may be achieved

CDO operator piping in combination with shell-programming can be demonstrated shortly in the following very short bash-script, where two input files are interpolated to a common resolution, the resulting fields are added, and the sum is time-averaged, the result being stored in a new file. Shell-programming allows for the diagnostic output of additional information to the screen, here the spatial-average of the field resulting from the CDO-operator-chain. You may find this code in file `fldmean.sh` that is among the distributed files.

```
#!/bin/bash

#select level 6, interpolate to 1x1 degree, convert from deg C to K
cdo timmean -addc,273.15 -remapcon,r360x180 -sellevel,6 input.nc output.nc
#note: the rightmost command ist executed first

#compute global mean
spat_avg=$(cdo output -fldmean output.nc)

#print result to screen
echo
echo
echo "average of global ocean surface temperature is ${spat_avg} K."

#clean up
rm output.nc
```

In the following tasks we will process NetCDF files using the CDO. In order to fulfill these tasks, you may refer to the documentation that is available online (<https://code.zmaw.de/projects/cdo/embedded/index.html>). For convenience, some useful CDO commands are summarized in the following listing. You may find this code in file `cdo_examples.txt` that is among the distributed files.

```
#extract a variable named "varname" from file input.nc
cdo selvar,varname input.nc output.nc

#extract the first month of all years in file input.nc
cdo selmon,1 input.nc output.nc

#calculate a time average over a time series input.nc
cdo timmean input.nc output.nc

#generate a seasonal mean from input.nc
cdo seasmean input.nc output.nc

#generate a year mean from input.nc
cdo yearmean input.nc output.nc

#calculate an average annual cycle from file input.nc
cdo ymonmean input.nc output.nc

#select a region from input.nc, longitude "a" to "b", latitude "c" to "d"
cdo sellonlatbox,a,b,c,d input.nc output.nc

#calculate a spatial average of field input.nc
cdo fldmean input.nc output.nc
```

```
#write the output of a CDO operator "a" to the screen (omits file output.nc)
cdo output -a input.nc

#calculate the difference between two NetCDF files input1.nc and input2.nc
cdo sub input1.nc input2.nc output.nc

#multiply two fields input1.nc and input2.nc
cdo mul input1.nc input2.nc output.nc

#add a constant "a" to field input.nc
cdo addc,a input.nc output.nc

#select only regions of input2.nc, for which mask input1.nc is true (i.e. 1)
#represents an if-then programming construct
cdo ifthen input1.nc input2.nc output.nc

#use input2.nc, where mask input1.nc is true - otherwise use input3.nc
#represents an if-then-else programming construct
cdo ifthenelse input1.nc input2.nc input3.nc output.nc

#reduce a data range (a,b) in input.nc to the constant value "c"
cdo setrtoc,a,b,c input.nc output.nc

#replace a data range (a,b) in input.nc by the missing value ("NaN")
cdo setrtomiss,a,b input.nc output.nc

#calculate the trend of a time series in input.nc;
#the trend is defined by offset "a" and slope "b" of the regression line;
#"a" is stored in a.nc, "b" is stored in b.nc
cdo trend input.nc a.nc b.nc

#calculate the horizontal area covered by each grid cell of input.nc
cdo gridarea input.nc output.nc
```

Reducing NetCDF data sets

The file INIOM_PD_3901-4000_tsurf_seaice.nc contains two time series of climatological fields.

Reduce the NetCDF file by performing the following tasks using the CDO:

- Task 1: Split the data set in two separate data sets, one for variable `tsurf`, one for variable `seaice`.
- Task 2: Calculate a time average over the full time period available in each of the separate data sets created in Task 1.

- Task 3: For each of the data sets created in Task 1 calculate an average annual cycle that is representative for the full time period of the time series.

Hint: In order to create an average annual cycle (multi-year monthly mean), you have to generate a new data set that contains twelve months. The data stored in each time step (month) n of this new data set o must represent the average over all corresponding months contained in the full time series of the initial data set i , i.e. (see CDO documentation):

$$o(n, x) = \text{mean}(i(t, x), \text{month}(i(t)) == n); \quad n \in (1, 12)$$

- Task 4: Calculate a seasonal mean from the average annual cycle of both variables retrieved in Task 3.
- Task 5: Select only the Northern Hemisphere of the time average retrieved in Task 2.
- Task 6: Calculate the global average temperature from the data set retrieved in Task 2.
- Task 7: Repeat Task 4 using CDO-pipes, i.e. pipe the output of the CDO operator that calculates the average annual cycle of a data set created in Task 3 into the operator for the seasonal mean. Is there any difference between the file generated in this task with respect to the result retrieved in Task 4?

In the following we will further analyze and transform files that were generated during previous tasks.

- Task 8: Transfer the time-average temperature field, retrieved in Task 2 above, to units of °C. What is the average temperature in °C over the Northern Hemisphere?
- Task 9: Calculate global average monthly temperatures for the average annual cycle retrieved in Task 3.

- Task 10: What is the global- and time-averaged temperature over land and ocean? As input you may use the file generated in Task 2. You need to supply to CDO a mask that defines distributions of land and ocean; use variable SLM of file T31GR30_jan_surf.nc.
- Task 11: What is the global average trend in the temperature time series created in Task 1?
- Task 12: Calculate the monthly average sea-ice cover in units of km^2 in the Northern Hemisphere. Use the average annual cycle of the sea-ice field generated in Task 3.

11.3.1 A simple model of sea level rise

In this task we will learn how shell-programming can help in solving scientific problems. We will, based on simple assumptions, write a shell-script that generates a NetCDF data set of a rising sea level, attributed to a loss of land ice, and produces a data set that illustrates the resulting continental flooding. This shell-script can be considered as a very simplified model of continental flooding due to sea-level rise. The only necessary input for this model is a global data set of surface elevation.

The world's water is unevenly distributed among four major climate subsystems and three different states of matter. Respective climate subsystems are the atmosphere (water vapor and droplets, ice crystals), the ocean (liquid water and frozen, solid water), and the land surface (liquid and solid water). The subsystem of the Earth composed of frozen water is commonly referred to as the Cryosphere. It includes permafrost, lake and river ice, sea ice, snow, glaciers, ice caps and ice sheets. Within the Cryosphere, ice sheets are the largest storage of frozen water. At present, there are two major ice-sheets, the Greenland Ice-Sheet (GIS) and the Antarctic Ice-Sheet (AIS), containing water volumes of 7.3 m and 56.6 m sea level equivalent, respectively (see Table 4.1 of the contribution of Working Group 1 to the Fourth Assessment Report by the IPCC, page 342 of <http://www.ipcc.ch/pdf/assessment-report/ar4/wg1/ar4-wg1-chapter4.pdf>). This means that, assumed both the GIS and the AIS melted completely, the global average sea level would rise by 63.9 m, leading to a flooding of large parts of coastal regions, where 44% of humankind live.

The relative volume distribution of water between the different physical states of matter (solid, liquid, evaporated) depends on the average surface temperature of the Earth. There have been geologic time scales that were much warmer than present, where virtually no ice sheets were available (see e.g. Fig. 2 of Zachos et al., 2001, <http://science.sciencemag.org/content/292/5517/686.full>). Anthropogenic emissions of, particularly, carbon dioxide, are expected to increase the average surface temperature on Earth via a modification of the global radiative energy balance, since carbon dioxide in the atmosphere contributes to the so-called greenhouse effect (see the publication by Arrhenius, 1896, available at http://www.rsc.org/images/Arrhenius1896_tcm18-173546.pdf). Due to global warming, the volume of water stored in ice sheets will become smaller, and ultimately increase the volume of the global oceans, thus causing sea level to rise.

Outline of the exercise Here, we develop a simple model of sea level rise which illustrates regions of the earth that are flooded if the GIS and AIS melt. Our work employs command line tools that are freely available and should be present on the computer of any scientist that performs numerical modelling or substantial scientific data processing. If you run a linux system, these tools should either already be available, or should be installed with a minimum amount of work.

As it is the case for any scientific model, which is per definition only an idealization of a natural system, our methodology is based on several simplifications. First of all, it is assumed that the melt process of available ice sheets is linear and occurs over a time period of 1000 y. Furthermore, we do not apply a physical flow model that simulates water inflow from the coast, since such a model can hardly be designed and formulated within the framework of this exercise. Instead, we will calculate water height over land via subtracting the rising sea level from the global elevation field. Negative values indicate the presence of water over land.

Our work depends on the availability of a gridded land elevation data set and a land-sea-mask that separates the land surface into regions that belong to land and ocean. These data sets are also the only input files that are necessary for our analysis. The files are taken from a set of boundary

conditions for numerical models of the circulation of the atmosphere. These have been derived from global elevation data sets, that are commonly generated using modern satellite-based radar interferometry, e.g. by the satellite system ICESat (icesat.gsfc.nasa.gov/icesat/). For use in climate modelling, this data has been interpolated to the resolution of the climate model. The orography in a climate model itself is therefore also a model - a model of the land surface elevation, whose accuracy depends on the quality of the measurement and on the grid resolution to which it is interpolated to. Higher resolution of the model grid means better agreement of the gridded orography with the original high-resolution orography data set. The data set used here originates from the boundary conditions of the Atmosphere General Circulation Model ECHAM5 at T63-resolution, which corresponds to a horizontal resolution of 1.9° by 1.9° . Close to the equator, this corresponds to roughly 200 km by 200 km per grid cell.

```
#!/bin/bash

#some definitions that control the script
n_time_steps=1000 #number of time steps over which sea level rises
time_unit='1year' #resolution of the time axis
time_reference='2000-01-01,00:00:00' #reference time (first date)
sl_start=0 #sea level ramp starts with elevation of 0 m
sl_end=75 #sea level ramp ends with elevation of 75 m

#definition of input files
orography_file='data/orography.nc'
lsm_file='data/land_sea_mask.nc'

#definition of output files
sea_level_output_file="sea_level_time_series.nc"
flooding_file='flooding_due_to_ice_melt.nc'

#clean up from (potential) previous runs of this script
if [ -e ${sea_level_output_file} ]
then
  rm ${sea_level_output_file}
fi
if [ -e ${flooding_file} ]
then
  rm ${flooding_file}
fi

#find minimum and maximum values of orography (necessary for file processing)
min_val=$(cdo output -fldmin ${orography_file} | sed 's/ //g')
max_val=$(cdo output -fldmax ${orography_file} | sed 's/ //g')
```

```

#subtract / add small value from / to min_val and max_val
#in order to increase the data range
#(cdo setrtomiss, used below, does not include limits of the
#processed data range, but we need to include these)
min_val=$(echo "scale=3; ${min_val}-1" | bc)
max_val=$(echo "scale=3; ${max_val}-1" | bc)

#generate a NetCDF file that contains n_time_steps, all values NaN;
#this file will be used to generate a time-dependent sea level data set;
#our approach, taking an existing NetCDF file and modifying it, saves
#lots of tedious work in setting up a new NetCDF file from scratch,
#which involves definition of dimensions, variables, and the definition of
#links between these
for ((i=0; i<${n_time_steps}; i+=1))
do
    #generate n_time_steps copies of orography,
    #add a time axis, and replace all values by NaN
    echo "creating file timestep_${i}.nc ..."
    cdo setrtomiss,${min_val},${max_val} \
        -settaxis,${time_reference},${time_unit} \
        ${orography_file} timestep_${i}.nc 2>/dev/null
done

#merge the above generated files to one NetCDF file
cdo mergetime timestep_?.nc timestep_??nc timestep_???.nc timeseries.nc

#set a proper time axis for the whole time series
cdo settaxis,${time_reference},${time_unit} timeseries.nc timeseries_timeaxis.nc

#clean up temporary files
rm timestep_*.nc timeseries.nc

#run a loop that generates a linear sea level ramp, starting a sea_level_start
#and ending at sea_level_end
c_sl=${sl_start}
for ((i=0; i<${n_time_steps}; i+=1))
do
    #redefine current sea level c_sl depending on the value of i
    c_sl=$(echo "scale=3; ${sl_start}+${sl_end}*${i}/(${n_time_steps}-1)" | bc)

    #generate a three digit string of i containing a leading zero (for file names)
    if [ ${i} -le 9 ]
    then
        counter_str="00${i}"
    else
        if [ ${i} -le 99 ]
        then
            counter_str="0${i}"
        else
            counter_str="${i}"
        fi
    fi
done

```



```

fi

#generate from each time step of timeseries_timeaxis.nc a NetCDF file that
#contains a global field corresponding to the respective sea level at that time
echo "creating file sea_level_time_series_${counter_str}.nc ..."
cdo setmisstoc,${c_sl} \
  -setrtomiss,${min_val},${max_val} \
  -seltimestep,$((i+1)) \
  timeseries_timeaxis.nc sea_level_time_series_${counter_str}.nc 2>/dev/null
done

#merge the single time steps of the sea level time series into one file
cdo mergetime sea_level_time_series_???.nc ${sea_level_output_file}

#clean up temporary files
rm sea_level_time_series_???.nc timeseries_timeaxis.nc

#subtract sea level from elevation in order to calculate water height over land
#negative values will depict water
cdo sub ${orography_file} ${sea_level_output_file} tmp.nc

#remove the elevation and sea level from the data set so that all values in file
#flooding_file depict the water level caused by flooding
cdo mulc,-1 \
  -mul ${lsm_file} \
  -setmisstoc,0 \
  -setrtomiss,0,${max_val} \
  tmp.nc ${flooding_file}

#clean up
rm tmp.nc

```

Exercise 73 – Shell and netcdf

Modify the above source code with the purpose of simulating the flooding that results from 1010 years of sea-level rise due to ocean warming. To this end you need to:

1. Make a literature research with the purpose of finding an estimate of rates r of current or near-future annual sea-level rise due to ocean warming; commonly such estimates are in the order of a few millimeters per year. Convert your choice of r to units of m/y
2. Modify the above given shell-script so that it creates a 1010 y linear ramp that contains the time-dependent global sea level SSH , starting with zero and ending with the value $SSH_{\text{end}} = r \cdot 1010 \text{ y}$.

Run both the unmodified version of the model, simulating sea-level rise due to land-ice melt,

and the modified version of the model, simulating sea-level rise due to ocean warming.

Note: in order to execute a bash script “script.sh”, open a Linux shell-session, navigate to the folder where the script is located, and execute the command “./script.sh”. You might have to activate executable rights for this script by executing “chmod u+x script.sh” in advance.

Analyze and discuss your results:

1. Make a time-dependent analysis of the flooding that is predicted by this simple model of sea level rise; you could, for example, generate plots of the flooding over land for two or three interesting time steps that you select from the 1010 y time series.
2. Make a similar analysis for the sea level rise that is simulated by the unchanged original shell-script, which considers melting of land-ice.
3. Compare the results of both analyses and discuss differences. Where are regions located, where people should not live anymore in the future?
4. Make a critical analysis of the flooding simulated by this very simple model.

For the last point, these are questions that you might consider:

1. What are weak points of the simulation? Consider in your discussion the term “volume conservation”, the stationarity of the applied sea level rise, and the spatial resolution of the orographic data. How would you expect the simulated flooding to look like if the spatial resolution of orography was higher?
2. The model, in its current version, has at least one dramatic flaw. Where is the simulated flooding, that should only be attributed to water inflow from the ocean, obviously wrong? What is the reason for the erroneous result?
3. Could you imagine what additional physical processes a corrected version of the model would need to contain in order to avoid the identified problem? Give a short description of your ideas, maybe illustrate them with some sketches.

Exercise 74 – Evaluate possible zones of wine production and climatic conditions

For wine production there exists some empirical laws:

Northern Hemisphere: degree of latitude 40 – 50

Southern Hemisphere: degree of latitude 30 – 40

The climatic demands are:

1. Vegetation period 180 – 250 days (days with daily mean > 5 °C)
2. Mean temperature 9 °C– 21 °C
3. White wines optimal annual mean temperature 9.5–11.5 °C
4. Red Wines optimal annual mean temperature 10.5–13 °C
5. Sunshine hours per year: 1500–2000 h
6. Precipitation 480 mm–700 mm

Task: Please calculate the possible zones of red wine production using 4 and 6 using temperature and precipitation constraints. The resulting figure should be similar to Fig. 11.1. Here are the links for getting the CRU surface temperature:

http://climexp.knmi.nl//CRUData/cru_ts3.23.1901.2014.tmp.dat_1.nc

and precipitation data:

http://climexp.knmi.nl//CRUData/cru_ts3.23.1901.2014.pre.dat_1.nc

Choose two different reference periods: 1901-1920 and 1991-2010

Calculate furthermore the zones for future scenarios (RCP4.5 and RCP8) for the years 2081-2100. Data are accessible here:

http://cmip-pcmdi.llnl.gov/cmip5/data_getting_started.html.

For this, choose one particular model and compare the zones with a reference period using the same model (e.g., 1960-1980). Some people could speculate that the alcohol belts (Fig. 11.2) may be shifting under global warming. Can this be substantiated by your results?

Hint: use the cdo commands from the lecture!

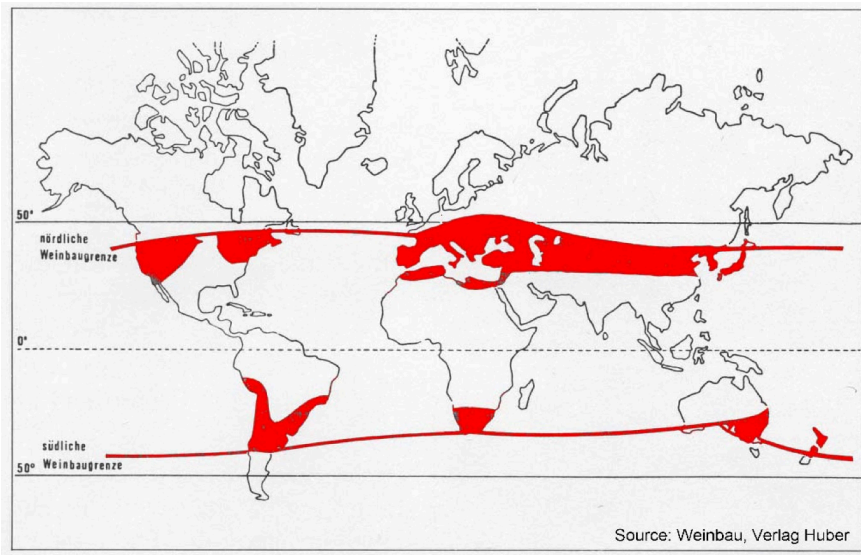


Figure 11.1: Zones of today's wine production.

Alcohol Belts

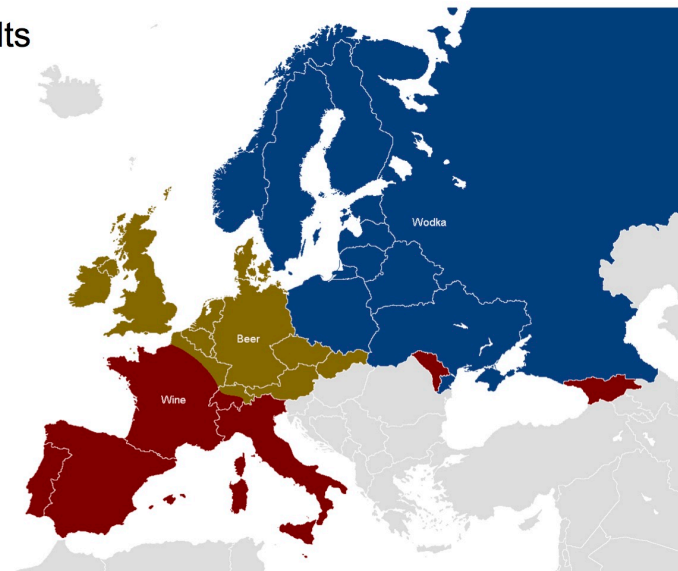


Figure 11.2: Alcohol belts in Europe.

Bibliography

Abramowitz, M. and Stegun, I. A. (1965). Handbook of mathematical functions with formulas, graph, and mathematical tables. *Applied Mathematics Series*, 55:1046.

Arnold, L. (1995). *Random dynamical systems*. Springer.

Arnold, L. (2001). *Hasselmann's program revisited: The analysis of stochasticity in deterministic climate models*, volume 49. Birkhäuser, Boston.

Baker, G. L. and Blackburn, J. A. (2005). *The pendulum: a case study in physics*, volume 8. Oxford University Press Oxford.

Barber, D., Dyke, A., Hillaire-Marcel, C., Jennings, J., Andrews, J., Kerwin, M., Bilodeau, G., McNeely, R., Southon, J., Morehead, M., and Gagnonk, J.-M. (1999). Forcing of the cold event of 8,200 years ago by catastrophic drainage of laurentide lakes. *Nature*, 400(6742):344–348.

Bhatnagar, P., Gross, E. P., and Krook, M. K. (1954). A model for collision process in gases. i. small amplitude processes in charged and neutral one-component system. *Phys. Rev*, 94:511.

Boltzmann, L. (1896). *Vorlesungen über Gastheorie : 2 Volumes (in German)*. Leipzig 1895/98 UB: O 5262-6.

Boltzmann, L. (1995). *Lectures on Gas Theory*. Dover Publ. New York. ISBN 978-0486684550.

Broecker, S. and Peng, T.-H. (1982). *Tracers in the Sea*. Columbia University.

- Broecker, W. S. (1987). The biggest chill. *Natural History*, 97(2):74–82.
- Broecker, W. S. et al. (1991). The great ocean conveyor. *Oceanography*, 4(2):79–89.
- Brüning, R. and Lohmann, G. (1999). Charles s. peirce on creative metaphor: a case study on the conveyor belt metaphor in oceanography. *Foundations of science*, 4(4):389–403.
- Bryan, F. (1986). High latitude salinity effects and inter-hemispheric thermohaline circulations. *Nature*, 323(3):301–304.
- Buckingham, E. (1914). On physically similar systems; illustrations of the use of dimensional equations. *Physical Review*, 4(4):345–376.
- Budyko, M. I. (1969). The effect of solar radiation variations on the climate of earth. *Tellus*, 21:611–619.
- Busch, W. (1865). *Max und Moritz (in German); Max and Maurice, a Juvenile History in Seven Tricks* . Braun und Schneider, München.
- Cercignani, C. (1987). *The Boltzmann equation and its applications*. Springer New York. ISBN 978-0387966373.
- Cercignani, C. (1990). *Mathematical methods in kinetic theory*. Plenum, 2 edition. ISBN 978-0306434600.
- Chelton, D. B. and Schlax, M. G. (1996). Global Observations of Oceanic Rossby Waves. *Science*, 272:234–238.
- Chen, D., Gerdes, R., and Lohmann, G. (1995). A 1-d atmospheric energy balance model developed for ocean modelling. *Theoretical and Applied Climatology*, 51:25–38.
- Chorin, A. J. and Hald, O. H. (2006). Stochastic tools in mathematics and science. surveys and tutorials in the applied mathematical sciences, vol. 1.

- Chorin, A. J., Kast, A. P., and Kupferman, R. (1999). Unresolved computation and optimal predictions. *Communications on pure and applied mathematics*, 52(10):1231–1254.
- Chorin, A. J., Kupferman, R., and Levy, D. (2000). Optimal prediction for hamiltonian partial differential equations. *Journal of Computational Physics*, 162(1):267–297.
- Courant, R., Friedrichs, K., and Lewy, H. (1928). Über die partiellen Differenzgleichungen der mathematischen Physik. *Mathematische Annalen*, 100:32–74.
- Courant, R., Friedrichs, K., and Lewy, H. (1967). On the partial difference equations of mathematical physics. *IBM J. Res. Dev.*, 11(2):215–234.
- Dansgaard, W., Johnsen, S., Clausen, H., Dahl-Jensen, D., Gundestrup, N., Hammer, C., C.S. Hvidberg, J. S., Sveinbjornsdottir, A., Jouzel, J., and Bond, G. (1993). Evidence for general instability of past climate from a 250-kyr ice-core record. *Nature*, 364:218–220.
- d’Humieres, D., Bouzidi, M., and Lallemand, P. (2001). Thirteen-velocity three-dimensional lattice boltzmann model. *PRE*, 63(6, Part 2).
- Dijkstra, H., Raa, L. T., and Weijer, W. (2004). A systematic approach to determine thresholds of the ocean’s thermohaline circulation. *Tellus A*, 56 (4):362.
- Doedel, E. J., Champneys, A. R., Fairgrieve, T. F., Kuznetsov, Y. A., Sandstede, B., and Wang, X. (1997). Continuation and bifurcation software for ordinary differential equations (with homcont). Available by anonymous ftp from ftp cs concordia ca, directory pub/doedel/auto.
- Egger, J. (2001). Master equations for climatic parameter sets. *Climate Dynamics*, 18(1-2):169–177.
- Einstein, A. (1905). Investigations on the theory of the brownian movement. *Ann. der Physik*, 17:549–560.

- Einstein, A. (1926). Die Ursache der Mäanderbildung der Flußläufe und des sogenannten Baer-schen Gesetzes. *Naturwissenschaften*, 14:223–224.
- Evans, D. J. and Morriss, G. (2008). *Statistical mechanics of nonequilibrium liquids*. Cambridge University Press.
- Fairbanks, R. G. (1989). A 17, 000-year glacio-eustatic sea level record: influence of glacial melting rates on the younger dryas event and deep-ocean circulation. *Nature*, 342(6250):637–642.
- Feigenbaum, M. J. (1980). The transition to aperiodic behaviour in turbulent systems. *Commun. Math. Phys.*, 77.
- Flammer, C. (1957). *Spheroidal wave functions*. Stanford University Press.
- Frisch, U. (1996). *Turbulence: the legacy of A.N. Kolmogorov*. Cambridge University Press. ISBN 0-521-45103-5.
- Gerkema, T., Zimmerman, J., Maas, L., and Van Haren, H. (2008). Geophysical and astrophysical fluid dynamics beyond the traditional approximation. *Reviews of Geophysics*, 46(2).
- Gill, A. E. (1982). *Atmosphere-ocean dynamics*, volume 30. Academic Press. International Geophysics Series.
- Givon, D., Kupferman, R., and Stuart, A. (2004). Extracting macroscopic dynamics: model problems and algorithms. *Nonlinearity*, 17(6):R55.
- Gottwald, G. (2010). On recent trends in climate dynamics. *AMS Gazette*, 37(5).
- Grassberger, P. and Procaccia, I. (1983). Measuring the strangeness of strange attractors. *Physica D: Nonlinear Phenomena*, 9(2):189–208.
- Haken, H. (1996). Slaving principle revisited. *Physica D: Nonlinear Phenomena*, 97(1):95–103.

- Haney, R. L. (1971). Surface thermal boundary conditions for ocean circulation models. *Journal of Physical Oceanography*, 1:241–248.
- Hasselmann, K. (1976). Stochastic climate models. Part I. Theory. *Tellus*, 6:473–485.
- He, X. and Luo, L. S. (1997). Theory of the lattice Boltzmann method: From the Boltzmann equation to the lattice Boltzmann equation. *Phys. Rev. E*, 56(6):6811–6817.
- Holton, J. R. (2004). *An Introduction to Dynamic Meteorology*. Elsevier Academic Press, Burlington, MA.
- Kambe, T. (2007). *Elementary Fluid Mechanics*. World Scientific Publishing.
- Kuznetsov, Y. A. (1998). *Elements of applied bifurcation theory*, volume 112. Springer, New York.
- Landau, L. D. and Lifshitz, E. M. (1959). *Fluid Mechanics*, volume 6 of *Course of Theoretical Physics*. Pergamon Press, Oxford.
- Langevin, P. (1908). On the theory of brownian motion. *Comptes Rendues*, 146:530–533.
- Leith, C. (1975). Climate response and fluctuation dissipation. *Journal of the Atmospheric Sciences*, 32(10):2022–2026.
- Lohmann, G. (2003). Atmospheric and oceanic freshwater transport during weak atlantic overturning circulation. *Tellus A*, 55(5):438–449.
- Longuet-Higgins, M. S. (1968). The eigenfunctions of laplace’s tidal equations over a sphere. *Philosophical Transactions for the Royal Society of London. Series A, Mathematical and Physical Sciences*, pages 511–607.
- Lorenz, E. (1982). Atmospheric predictability experiments with a large numerical model. *Tellus A*, 34:505–513.
- Lorenz, E. N. (1960). Maximum simplification of the dynamic equations. *Tellus*, 12(3):243–254.

- Lorenz, E. N. (1963). Deterministic nonperiodic flow. *Journal of the atmospheric sciences*, 20(2):130–141.
- Lorenz, E. N. (1976). Nondeterministic theories of climatic change. *Quaternary Research*, 6(4):495–506.
- Lorenz, E. N. (1984). Irregularity: a fundamental property of the atmosphere*. *Tellus A*, 36(2):98–110.
- Lucarini, V., Blender, R., Herbert, C., Pascale, S., Ragone, F., and Wouters, J. (2014). Mathematical and physical ideas for climate science. *Rev. Geophys.*
- Maas, L. R. (1994). A simple model for the three-dimensional, thermally and wind-driven ocean circulation. *Tellus A*, 46(5):671–680.
- Manabe, S. and Stouffer, R. (1993). Century-scale effects of increased atmospheric CO_2 on the ocean atmosphere system. *Nature*, 364:215–218.
- Mandelbrot, B. B. (1967). How long is the coast of Britain: Statistical self-similarity and fractal dimension. *Science*, 155:636–638.
- Mandelbrot, B. B. (1983). *The fractal geometry of nature*. Macmillan.
- Matsuno, T. (1966). Quasi-geostrophic motions in the equatorial area. *J. Meteor. Soc. Japan*, 44(1):25–43.
- Mori, H. (1965). Transport, collective motion, and brownian motion. *Progress of Theoretical Physics*, 33(3):423–455.
- Mori, H., Fujisaka, H., and Shigematsu, H. (1974). A new expansion of the master equation. *Progress of Theoretical Physics*, 51(1):109–122.
- Müller and Maier-Reimer (2000). Trapped rossby waves. *Phys. Rev. E*, 61:1468 – 1485.

- Müller, D., Kelly, B., and O'Brien, J. (1994). Spheroidal eigenfunctions of the tidal equation. *Physical review letters*, 73(11):1557.
- Müller, D. and O'Brien, J. (1995). Shallow water waves on the rotating sphere. *Physical Review E*, 51(5):4418.
- Olbers, D. (2001). A gallery of simple models from climate physics. *In: Stochastic Climate Models, Progress in Probability (Eds.: P. Imkeller and J. von Storch)*, 49:3–63.
- Peitgen, H.-O. and Richter, P. (1986). *The Beauty of Fractals*. Heidelberg: Springer-Verlag.
- Proudman, J. (1916). On the motion of solids in a liquid possessing vorticity. *Proc. R. Soc. Lond. A*, 92:408–424.
- Rahmstorf, S. (1996). On the freshwater forcing and transport of the Atlantic thermohaline circulation. *Climate Dynamics*, 12:799–811.
- Rayleigh, L. (1916). On convection currents in a horizontal layer of fluid, when the higher temperature is on the under side. *Phil. Mag.*, 6:529–546.
- Rooth, C. (1982). Hydrology and ocean circulation. *Progress in Oceanography*, 11:131–149.
- Rossby, C.-G. (1939). "relation between variations in the intensity of the zonal circulation of the atmosphere and the displacements of the semi-permanent centers of action". *Journal of Marine Research*, 2 (1):38–55.
- Saltzman, B. (1962). Finite amplitude free convection as an initial value problem – i. *Journal of the Atmospheric Sciences*, 19:329–341.
- Shannon, C. E. (1948). A Mathematical Theory of Communication. *Bell System Technical Journal*, 27 (3):379–423.
- Stommel, H. (1961). Thermohaline convection with two stable regimes of flow. *Tellus*, 13:224–230.

- Strogatz, S. (2000). *Non-linear Dynamics and Chaos: With applications to Physics, Biology, Chemistry and Engineering*. Perseus Books.
- Taylor, G. (1917). Motion of solids in fluids when the flow is not irrotational. *Proc. R. Soc. Lond. A*, 93:92–113.
- Townsend, S., Lenosky, T., Muller, D., Nichols, C., and Elser, V. (1992). Negatively curved graphitic sheet model of amorphous carbon. *Physical Review Letters*, 69(6):921–924.
- Tritton, D. J. (1988). *Physical Fluid Dynamics*. Oxford University Press, Science Publication. ISBN 978-0-19-854493-7.
- Uhlenbeck, G. E. and Ornstein, L. S. (1930). On the theory of the brownian motion. *Physical review*, 36(5):823.
- van Kampen, N. G. (1981). *Stochastic processes in physics and chemistry*. North Holland. ISBN 978-0-444-52965-7.
- Wüst, G. (1935). Schichtung und Zirkulation des Atlantischen Ozeans. Das Bodenwasser und die Stratosphäre. *Wiss. Ergebn. Dtsch. Atlant. Exped. 'Meteor' 1925-1927*, 6(1):1–288.
- Zwanzig, R. (1960). Ensemble method in the theory of irreversibility. *The Journal of Chemical Physics*, 33:1338.
- Zwanzig, R. (1980). Problems in nonlinear transport theory. In *Systems far from equilibrium*, pages 198–225. Springer.
- Zwanzig, R. (2001). *Nonequilibrium statistical mechanics*. Oxford University Press, USA.