Supplement of Geosci. Model Dev., 7, 1395–1408, 2014
http://www.geosci-model-dev.net/7/1395/2014/
doi:10.5194/gmd-7-1395-2014-supplement

Geoscientific Model Development
Open Access

Supplement of

Semi-Lagrangian transport of oxygen isotopes in polythermal ice sheets: implementation and first results

T. Goelles et al.

Correspondence to: T. Goelles (thomas.golles@unis.no, thomas.goelles@gmail.com)

1 Additional plots for comparison of backtracking schemes

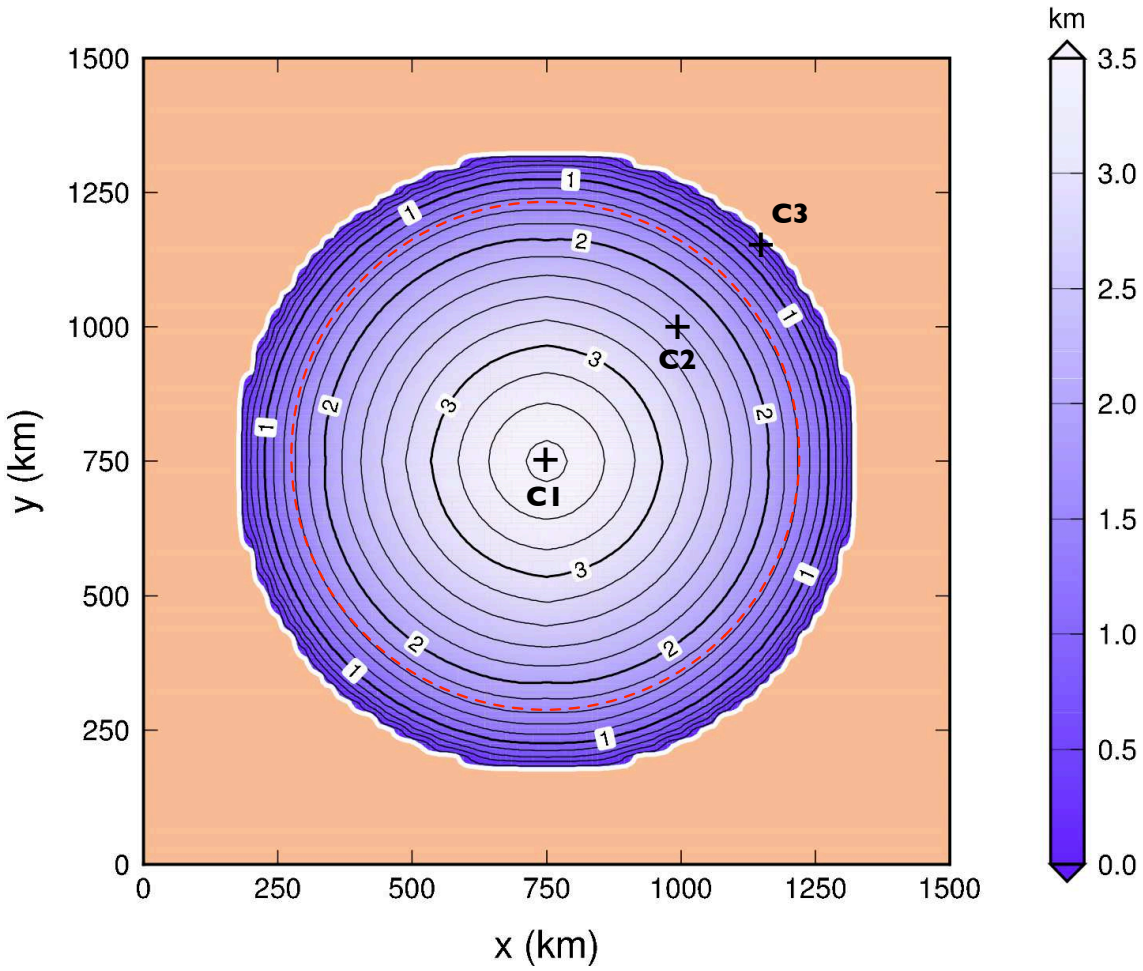

Figure 1: Steady-state surface elevation of the EISMINT experiment A at simulation time 300 ka. The three crosses mark the core locations C1 and C2 in the accumulation zone and C3 in the ablation zone where the equilibrium line is indicated in red.

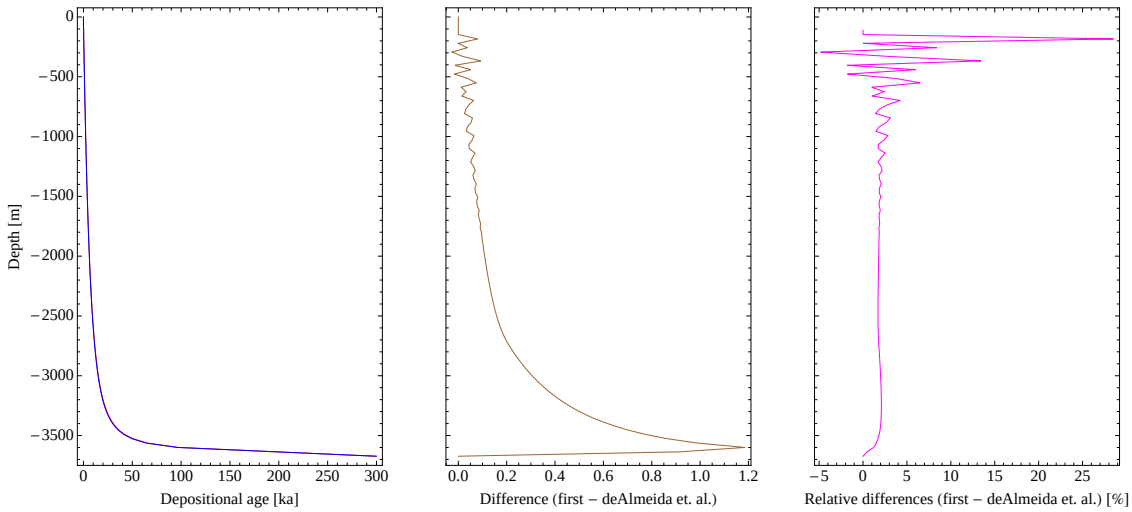

Figure 2: Simulated ice core C1 in the ice sheet centre ($x = y = 750$ ka). The figure on the left shows depth against depositional age in ka with first order in red and second order (de Almeida et al.) in blue. The middle figure shows the absolute difference (first - de Almeida et al.) and the right figure the relative difference in %.

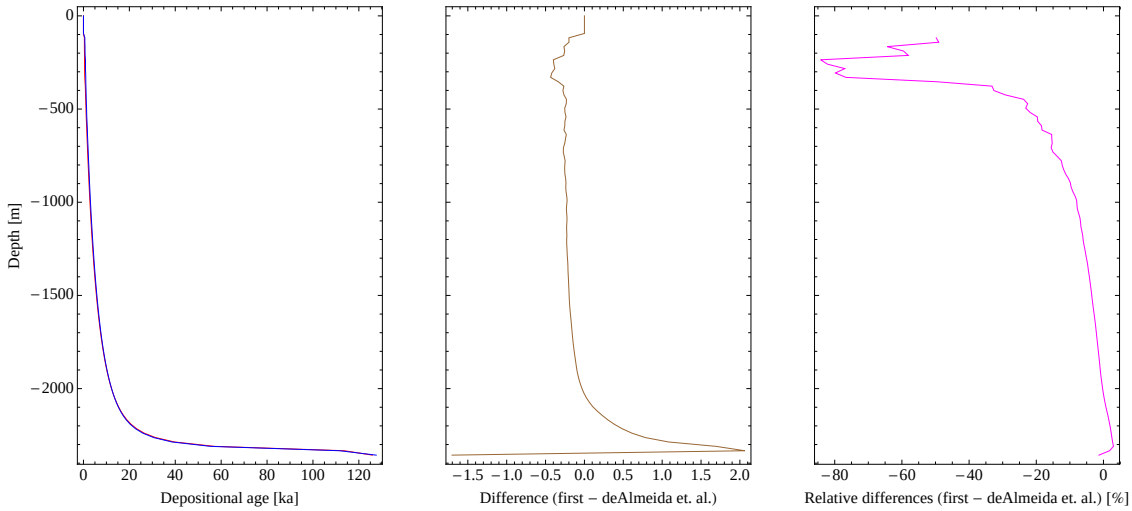

Figure 3: Ice core C2 with the same figure arrangement as in C1.

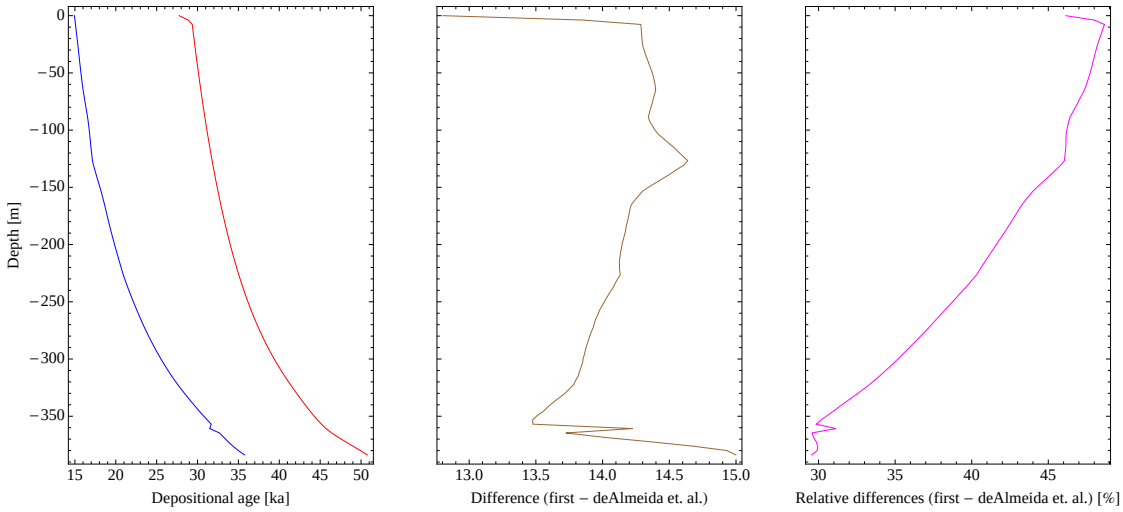

Figure 4: Ice core C3 in the ablation area ($x = y = 1150$ ka). The first order scheme (red) produces ice age put o 15% higher than the second order scheme by de Almeida et. al.

2 Quick Start Manual for ISOPOLIS 1.0

This Quick Start Manual is based on Greve (2008).

2.1 Licence

ISOPOLIS is based on SICOPOLIS 2.9 by Ralf Greve under the Creative Commons Attribution Share- Alike license version 3.0 and released under the same license.

2.1.1 License of SICOPOLIS 2.9

2.1.2 License of ISOPOLIS 1.0

2.2 Installation Guide

ISOPOLIS has been tested under Linux and Mac OS 10.6.3. and above.

Besides standard Unix programs you need following software:

- the GNU build system
- the Subversion revision control system: `http://subversion.apache.org/`
- Fortran compiler such as gfortran or ifort
- netCDF (3.6.1 or higher) library compiled with your Fortran compiler: `http://www.unidata.ucar.edu/software/netcdf/`
- the netCDF Operator (NCO): `http://nco.sourceforge.net/`
- a netCDF viewer such as Panoply: `http://www.giss.nasa.gov/tools/panoply/` (optional but recommended)
- GMT for plotting: (`http://gmt.soest.hawaii.edu/`) (optional but recommended)

If you want to contribute to the development of ISOPOLIS you need to request to join the project at `http://aforge.awi.de`. There you will find all the necessary information to access the SVN-repository. If you just want to run the software with your model setups without altering the code you can follow the step-by-step guide.

Note that all the commands are for the *Bash* shell.

1. install all the above mentioned software
2. check out the latest version of ISOPOLIS with:
   svn co http://aforge.awi.de/svn/isopolis/tags/v1.0
3. set the Fortran Compiler environment variable:
   export FC=gfortran or export FC=ifort
4. run the install script *install.sh* with your path to the netcdf library: the general command is ./install.sh path-to-netcdf-installation
   for example ./install.sh /opt/local

The install script will install SICOPOLIS, SICOGRAPH, SICOTRACE and SICOSTRAT using automake onto your computer.

2.3 Files and Directories

You will get the following three folders (in addition to doc which contains this document):

sicopolis

sicograph

sicotrace

The folder **sicotrace** contains SICOTRACE and SICOSTRAT, the subprograms needed for computation of trajectories and tracer transport.

SICOPOLIS

runs

The shell script *sico.sh* (bash) for running a single simulation under UNIX/LINUX˙ Shell script *mult_sico.sh* for running multiple simulations by repeated calls of *sico.sh*
Subdirectories **headers** specification files *sico_specs_runname.h* (*runname* : name of the run).
Shell script *compare.sh* to compare model output with reference run, in order to see if SICOPOLIS is still working correctly.

- Files included for runs grl40_tes_wre1000, grl20_test_wre1000 and grl10_test_wre1000

  → Greenland ice sheet, resolution 40 / 20 / 10 km, t = 1990 CE..2350CE (similar to the WRE1000 run by Greve (2004) and run #11 by Greve and Otsu (2007)).

- File included for run eismint_tm

  → EISMINT Phase 2 Simplified Geometry Exmperiment K, resolution 25 km, t=0ka...300ka Payne and others (2000)

- Files included for runs ant80_paleo04_init, ant80_paleo04, ant40_paleo04 init and ant40_paleo04

  → Antarctic ice sheet, resolution 80 and 40 km. t = -522ky ... -422ka for spin-up runs (...init), t=-422ka ... 0ka for the main runs (see Greve (2006) and Greve (2005))

- Files included for runs grl40_paleo01_init, grl40_paleo01, grl20_paleo01_init and grl20_paleo01

  → Greenland ice sheet, resolution 40 km for grl40..., 20 km for grl20...,
  t = -422 ka . . . - 250 ka for the spin-up runs (... init), t = - 250 ka . . . 0 ka for the main runs [similar to run hf_pmod2 by Greve (2005)]

- File included for run nhem80_nt012_new

  → northern hemisphere, resolution 80 km, t = -250 ka . . . 0 ka [similar to run nt012 by Greve and others (1999)].

- File nctest.h included for run nctest, only for testing purpose if netcdf is running correctly

src

Directory which contains the main program file sicopolis.F90.

- Subdirectory **subroutines/general**: general subroutines, for any modelled domain.
- Subdirectory **subroutines/ant**: subroutines specific for the Antarctic ice sheet.
- Subdirectory **subroutines/emtp2sge**: subroutines specific for the EISMINT Phase 2 Simplified Geometry Experiments.
- Subdirectory **subroutines/grl**: subroutines specific for the Greenland ice sheet.
- Subdirectory **subroutines/nhem**: subroutines specific for the northern hemi- sphere.

- Subdirectoryds **scans,tibet,heino,nmars,smars** for Scandinavia, Tibet, ISMIP HEINO, and the north and south polar caps of Mars.

sico_in

Directory which contains input data files for SICOPOLIS.

- Subdirectory **general**: general input files, for any modelled domain. Subdirectory **ant**: input files specific for the Antarctic ice sheet.
- Subdirectory **emtp2sge**: input files specific for the EISMINT Phase 2 Simplified Geometry Experiments.
- Subdirectory **grl**: input files specific for the Greenland ice sheet.
- Subdirectory **nhem**: input files specific for the northern hemisphere.
- Subdirectorys **scans,tibet,heino,nmars,smars** for Scandinavia, Tibet, ISMIP HEINO, and the north and south polar caps of Mars.

sico_out

Directory into which output files of SICOPOLIS simulations are written. It already contains a subdirectory with the name **R_grl40_test_wre1000** with contains the reference run. The run-script *sico.sh* will create a subdirectory with the name **F_runname** and all belonging files are stored therein.

tools

This directory contains the subdirectory **netcdf** with a tool to convert ASCII input files into a single netCDF file per model-setup.

SICOTRACE

This directory contains the two subprograms **SICOTRACE** and **SICOSTRAT**.

The directory **trace** contains **SICOTRACE** and has the following subdirectories:

input

This directory contains the subdirectory input files for forward trajectories (Lagrange)

matlab

This directory contains the subdirectory **forward** which contains the Matlab plotting routines for forward directory. This is not maintained anymore and may be deleted in future versions.

src

Directory which contains the main program file sicotrace.F90 and the subdirectory **subroutines** which contains the general subroutines.

The directory **strat** contains **SICOSTRAT** and has the following subdirectories:

plots

Empty folder where all the plots are stored in subfolders with the name **runname**

src

Directory which contains the main program file sicostrat.F90 and the shell scipt *dev.sh* and the subdirectories:

- Subdirectory **subroutine**: contains the general subroutines
- Subdirectory **gmt**: contains the GMT scripts and cpt files for plotting

SICOGRAPH

This directory contains SICOGRAPH a plotting tool to visualize SICOPLOLIS model outputs and reads netCDF files.

plots

Empty folder where all the plots are stored in subfolders with the name **F_runname**

src

Directory which contains the main program file *sicograph.F90* and the shell script *sicograph.sh* and all subroutines.

- Subdirectory **paramter_files**: contains the general subroutines
- Subdirectory **gmt_scripts**: contains the GMT scripts and cpt files for plotting

2.4 How to run a Simulation

Now there are three possible ways to run a simulation, depending what you want:

1. Ice sheet dynamics only
2. Semi-Lagrangian transport of tracers
3. Calculate forward trajectories

2.4.1 1. Ice sheet dynamics only

In the first and second case you have to run *sico.sh* in the subdirectory runs in SICOPOLIS. The help of *$ ./sico.sh -h* looks like this:

```
Usage: sico.sh -m<model>
    [-a]             => running SICOPOLIS and SICOTRACE
    [-c]             => continuing a previous run
    [-e<endtime>]    => if -a; endtime of simulation
    [-n]             => skip make clean
    [-d<dir>]        => append <dir> to output dir
    [-f]             => overwrites  output-directory
                        if it allready exists
    [-i<intervall>]  => if -a; intervall of sequentially running
                        the programms
    [-k]             => if -a; keep SICOPOLIS netcdf outputfiles
    [-o<save_step>]  => saving every save_step step or prov_arch
    [-s<starttime>]  => if -a; starttime of simulation
```

To get an overview of the available runs: *$ ./sico.sh -m?*

```
ant40_paleo04
ant40_paleo04_init
ant40_tm
ant80_paleo04
ant80_paleo04_init
eismint_tm
grl10_test_wre1000
grl20_paleo01
grl20_paleo01_init
grl20_test_wre1000
grl20_tm
grl40_paleo01
grl40_paleo01_init
grl40_test_wre1000
nctest
nhem80_nt012_nc2
nhem80_nt012_new
```

To run the netcdf test-run: *$ ./sico.sh -m nctest*. This will run SICOPOLIS with the header file *sico_specs_nctest.h* in the subdirectory headers. This is a short run (about 10 seconds computational time).

If you want to run the same model setup again and you want to overwrite the results you have to use the *-f* option.

2.4.2 Output

Output files of simulations are written to directory **sico_out/F_nctest/**.

```
nctest.log
nctest.ser
nctest01.erg
nctest_1.nc
nctest_nsteps.txt
out_sicopolis_nctest.dat
sico_specs.h
```

nctest.log lists the main specifications of simulation
nctest.ser is the time-series file which contains global parameters:

- Time, t
- Surface-temperature anomaly, D_Ts, or glacial index, glac_ind (forcing)
- Sea level, z_sl (forcing)
- Maximum ice thickness, H_max
- Maximum ice elevation, zs_max
- Ice volume, V_g
- Volume of the temperate ice, V t
- Freshwater production due to melting and calving, V_fw
- Sea-level equivalent of ice volume, z_sle
- Ice area, Aib
- Area covered by temperate ice, Atb
- Water drainage due to basal melting, V_bm
- Water drainage from the temperate layer, V_tld
- Maximum thickness of the temperate layer, H_t_max
- Maximum surface velocity, vs_max

nctest01.erg binary file with all fields needed if the computation is continued later on.

nctest_1.nc the netCDF file with all output fields and also some input parameters. The netCDF file is complied with the CF-1.4 conventions. The file has a maximum size of 10Gb if the output is bigger then a this limit a new file with name **nctest_2.nc** is created.

You can open the file with the recommended viewer Panoply and immediately make some plots for example of the surface topography, velocities and so on.

nctest_nsteps.txt a temporary file.
out_sicopolis_nctest.dat the terminal output of sicopolis.
sico_specs.h a copy of the header-file.

2.4.3 2. Semi-Lagrangian transport of tracers

This is also done with the same script *sico.sh*. The script runs SICOPOLIS and SICOTRACE by turns and deletes unnecessary variables in order to save disk-space. If you want to keep the original SICOPOLIS output use the *-k* option.

An example for this would be:
$ ./sico.sh -m eismint_tm -a -s0 -e300000 -i50000 -o10.

I will explain this in detail. As before *-m eismint_tm* defines the used header-file, this time the EISMINT phase 2 experiment. The option *-a* defines the subsequent use of SICOPOLIS and SICOTRACE. The option *-s* is followed by the initial year, in this example 0 and *-e* followed by the number 300000 is the last year of the simulation.

The option *-i* is followed by the years when SICOPOLIS is stopped and SICOTRACE does the post-processing of the output. So in this example in total two runs of both programs are performed. Be aware not to chose the interval to high so that the data-files are bigger than 10Gb and the interval has to be an even multiple of the time-step. If you use a NCO version without 64bit support you may also chose the interval so that the data-files are not bigger than 2Gb.

The option *-o* is followed by the number of time-steps where the full date-set is stored. In this example only every 10th time-step the full depositional archive is stored. So later on the post processing with SICOSTRAT can also only be done with for every 10th time step.

2.4.4 Output

Output files of simulations are written to directory **sico_out/F_eismint_tm/**.

```
dep_arch_0.nc
eismint_tm.cont
eismint_tm.core
eismint_tm.log
eismint_tm.ser
eismint_tm01.erg
eismint_tm_1.core
eismint_tm_1.ser
eismint_tm_nsteps.txt
eismint_tm_prov_arch_1.nc
eismint_tm_prov_arch_2.nc
eismint_tm_prov_arch_3.nc
eismint_tm_prov_arch_4.nc
eismint_tm_prov_arch_5.nc
eismint_tm_prov_arch_6.nc
endtime.txt
intervall.txt
log_1.txt
log_2.txt
log_3.txt
log_4.txt
```

```
log_5.txt
log_6.txt
n_dep.txt
nlabel.txt
out_sicopolis_eismint_tm_1.dat
out_sicopolis_eismint_tm_2.dat
out_sicopolis_eismint_tm_3.dat
out_sicopolis_eismint_tm_4.dat
out_sicopolis_eismint_tm_5.dat
out_sicopolis_eismint_tm_6.dat
out_sicotrace_eismint_tm_1.dat
out_sicotrace_eismint_tm_2.dat
out_sicotrace_eismint_tm_3.dat
out_sicotrace_eismint_tm_4.dat
out_sicotrace_eismint_tm_5.dat
out_sicotrace_eismint_tm_6.dat
run.txt
saves_step.txt
sico_specs.h
```

This time you have more files resulting from the subsequent runs of SICOPOLIS and SICOTRACE. So the files ending with the number 1 are the one from the first run. I will describe now the additional files you get compared to if you run SICOPOLIS alone.
dep_arch_0.nc is the depositional archive needed for the post-processing with SICOSTRAT
eismint_tm.cont binary file with all fields needed if the computation with SICOTRACE is continued later on.
endtime.txt a temporary file, needed if the computation is continued later on.
intervall.txt a temporary file, needed if the computation is continued later on.
n_dep.txt a temporary file, needed if the computation is continued later on.
nlabel.txt a temporary file, needed if the computation is continued later on.
run.txt a temporary file, needed if the computation is continued later on.

If you want to continue a previous run you can now to this without changing the headerfile by running the shell script *sico.sh* again.
$ ./sico.sh -m eismint_tm -c -a -e400000
This will use the output from the previous example and start in the year 300000 and end in the year 400000. The interval and -o options are read from files.

2.4.5 3. Calculate forward trajectories

First you need to run SICOPOLIS and than run SICOTRACE in Lagrangian mode. Start a run of SICOPOLIS and go to **sicotrace/trace/src**.

You need to compile SICOTRACE once with:

```
make clean
make
make sicotrace
```

or alternatively run the *dev.sh* script used for development.

Start SICOTRACE with for example:

./sicotrace eismint_tm eismint_AK_1> tracer.log

This will calculate forward trajectories for the model-run "eismint_tm" which has to be performed with SICOPOLIS beforehand. The details about the particle-distribution is the file **input/eismint_AK_1.txt**. The log-file stores the terminal-output into the text-file tracer.log

Example for the eismint_AK_1.txt inputfile:

```
% this is the configuration file for forward tracers
% The particles are released at the surface
% time_start at which the particles are depositioned (years)
6000
% coordinate point "i" : as integer
30
% numbers of particles in positive x direction
1
% coordinate point "j" : as integer
0
% numbers of particles in positive y direction
60
% numbers of particles in between "i" grid points
0
% numbers of particles in between "j" grid points
3
```

Explanation: Start of deposition in the year 6000 at position i=30 with 60 particles in y-direction on the grid-points and 3 in between every grid-point = 220 particles.

2.4.6 Output

Output-files are written to directory **sico_out/F_eismint_tm/** in the subdirectory **eismint_AK_1** as ASCII files. The file name has to following pattern; **i_j_is_js.tracer** with i and j being the grid-points and is, js the points in between grid-points.

2.5 Plotting with SICOGRAPH

The output described in section 2.4 can be visualized with any netCDF plotting tool at the user's preference. One possibility is to use SICOGRAPH, based on the Generic Mapping Tools GMT.

To see which runs are available go to **sicograph/src** and run *./sicograph.sh -m?* If you know the runname *./sicograph.sh -m runname* and you'll get a menu which allows to choose the type of plot you wish to produce. For example, try the option:

(1) Iso-surface topography

and enter desired time (from 0 with timestep is : 200) > 30000

Plot (1) with or (2) without colour bar? > 1

Plot (1) with or (2) without contour labels? > 1

You will find the plot as an EPS file in the subdirectory
sicograph/plots/F_runname

2.6 Stratigraphy and ice cores with SICOSTRAT

SICOSTRAT can do three things, plot cross-sections and ice cores and store the δ^{18}O field as a netCDF. The plots are stored in **sicotrace/strat/plots** and the δ^{18}O field in **sicopolis/sico_out/F_runname** .

To get an overview of the command-line program SICOSTRAT go to **sicotrace/strat/src** and compile (only once) SICOSTRAT with:

```
make clean
make
make sicostrat
```

and run SICOSTRAT in the shell *./sicostrat -h* you will get this help:

```
SICOSTRAT v1.0 to calculate and plot stratigraphy from SICOTRACE
outputs three different modes:plotting of ice cores,
plotting of cross-sections and writing the d18O field as netCDF
 ---------- CORE MODE ------------
 ./sicostrat core <model> <time> <y> <x> <variable> [-d] [-o] [-g]
 <model> the SICOPOLIS runname, required
 <time> time for plotting in years, required
 <y> y location of core in meters, required
 <x> x location of core in meters, required
 <variable> x, y, t or O  the plotting variable, O means delta18O
 example: ./sicostrat core eztracer_cm 200000 750000 750000 O
 example 2: Vostok location with degrees
 input in N  E with at least one digit after the comma, (max. 3)
 ./sicostrat core ant40 -200000 -78.464 106.837 O -g
 ---------- CROSS MODE ------------
 ./sicostrat cross <model> <time> <i/j> <loc> <variable> [-b] [-c] [-d]
        [-l] [-o] [-g]
 <model> the SICOPOLIS runname, required
 <time> time for plotting in years, required
 <i/j> cut along i or j, required
 <loc> index for cutting, required
 <variable> x, y, t or O  the plotting variable, O means deltaO18
 example: ./sicostrat cross eztracer_cm 200000 i 30 x -d
 ---------- d18O OUTPUT MODE ------------
 ./sicostrat  <model> <time>
 ---------- OPTIONS ------------
 [-b] for cross-mode, plotting of color-bar
 [-c] for cross-mode, plotting of contour lines
 [-d] for draft plots, with additional text on the plot
 [-g] for core-mode, for input and output in degrees
 [-l] for cross-mode, only plot the lower region (max. zb +500m)
 [-o] for manually produced output with SICOTRACE
```

2.6.1 Examples

1) ice core

./sicostrat core <model> <time> <y> <x> <variable> [-d] [-o] [-g]

./sicostrat core grl20_tm2 0 72.350 -37.380 x -g> strat.log
./sicostrat core grl20_tm2 0 72.350 -37.380 y -g> strat.log
./sicostrat core grl20_tm2 0 72.350 -37.380 t -g> strat.log
./sicostrat core grl20_tm2 0 72.350 -37.380 O -g> strat.log

This will generate plots in the folder **sicotrace/strat/plots** of the GRIP ice core. The first argument *core* is needed to lounch sicostrat in core mode. The second argument is the runname. The third input is the desired year (0=Present day). Then there are the coordinates in degrees N and E. The 6. argument is either *x,y,t,O* for plotting the depositional x,y,t or the derived δ^{18}O profiles. The option -g means than input and output are in degrees.

Other options are: -d for draft mode with additional infos in the plots.

In the subdirectory **sicotrace/strat/src/gmt/cores** contains δ^{18}O data from ice-cores and an info file with references and coordinates.

2.) cross sections

./sicostrat cross <model> <time> <i/j> <loc> <variable> [-c] [-d] [-l] [-o] [-g]

./sicostrat cross grl20_tm2 0 i 36 x -c -b > strat.log

This will generate a plot of a cross section along i=36 of the depositional x-coordinate with contour-lines (-c) and a color-bar (-b)

3.) output of δ^{18}O

./sicostrat <mode> <time> > strat.log

./sicostrat grl20_tm2 0 > strat.log
Will produce a netCDF output of the derived δ^{18}O 3D field at present-time t=0 in the **sicopolis/sico_out/grl20_tm2** folder.

References

Greve, R., 2004. Evolution and dynamics of the Greenland ice sheet over past glacial- interglacial cycles and in future climate-warming scenarios, Proceedings of the 5th International Workshop on Global Change: Connection to the Arctic (GCCA5), University of Tsukuba, Japan, 42–45.

Greve, R., 2005. Relation of measured basal temperatures and the spatial distribution of the geothermal heat flux for the Greenland ice sheet, *Annals of Glaciology*, **42**, 424–432.

Greve, R., 2006. Large-scale simulation of the Antarctic ice sheet over climate cycles, online, accessed 14.June.2010. URL `http://hdl.handle.net/2115/34433`.

Greve, R., 2008. SICOPOLIS V2.9 - Quick Start Manual, accessed 14.June.2010. URL `sicopolis.greveweb.net/docu/SICOPOLIS_V29_Quick_Start.pdf`.

Greve, R. and Otsu, S., 2007. The effect of the north-east ice stream on the Greenland ice sheet in changing climates, *The Cryosphere Discussions*, **1**(1), 41–76.

Greve, R., Wyrwoll, K. H., and Eisenhauer, A., 1999. Deglaciation of the Northern Hemisphere at the onset of the Eemian and Holocen, *Annals of Glaciology*, **28**, 1–8.

Payne, A. J., Huybrechts, P., Abe-Ouchi, A., Calov, R., Fastook, J. L., Greve, R., Marshall, S. J., Marsiat, I., Ritz, C., Tarasov, L., and Thomassen, M. P. A., 2000. Results from the EISMINT model intercomparison: the effects of thermomechanical coupling, *Journal of Glaciology*, **46**(153), 227–238.