

SICOPOLIS V3.2

– Quick Start Manual –

RALF GREVE

Institute of Low Temperature Science, Hokkaido University,
Kita-19, Nishi-8, Kita-ku, Sapporo 060-0819, Japan

March 2, 2016

Copyright 2009–2016 Ralf Greve

(with contributions by Heinz Blatter, Reinhard Calov, Thorben Dunse, Ben Galton-Fenzi, Thomas Gölles, Philipp Hancke, Nina Kirchner, Sascha Knell, Alex Robinson, Tatsuru Sato, Malte Thoma)

This file is part of SICOPOLIS.

SICOPOLIS is free software. It can be redistributed and/or modified under the terms of the GNU General Public License (<http://www.gnu.org/licenses/>) as published by the Free Software Foundation, either version 3 of the License, or (at the user's option) any later version.

SICOPOLIS is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

1 Requirements

- Unix-like system (e.g., Linux).
- Fortran 90/95 compiler.
So far, the Intel and GNU Fortran compilers are supported. Make sure that the environment variable FC is set to either ifort (Intel) or gfortran (GNU). If you wish to use a different compiler, please contact <help@sicopolis.net>.
- Optional: GNU Autotools (automake, autoheader, aclocal, autoconf, make).
- SICOPOLIS supports output either in native binary or in NetCDF format (see Sect. 6). For the latter (recommended), you need an installation of NetCDF version 3.6.x or newer (<http://www.unidata.ucar.edu/software/netcdf/>).
- For the shallow-shelf/shelfy-stream solver, a recent version (1.4.13 or newer) of the Library of Iterative Solvers for Linear Systems (Lis) is required (<http://www.ssisc.org/lis/>).

2 Download

- **Option 1: Using subversion [recommended]**

The subversion repository of SICOPOLIS is kindly hosted by the FusionForge system AWIForge of the Alfred Wegener Institute for Polar and Marine Research (AWI) in Bremerhaven, Germany (front page: <https://swrepo1.awi.de/projects/sicopolis/>).

1. Check out from the subversion repository:

```
svn checkout --username anonsvn --password anonsvn \  
https://swrepo1.awi.de/svn/sicopolis/tags/version32 sicopolis  
(or username/password from your own AWIForge account)
```

2. You should then have a new directory “sicopolis” that contains the entire program package. In this directory, execute the script (bash) copy_templates.sh:

```
./copy_templates.sh
```

- **Option 2: Downloading a tarball**

1. Download the gzipped tar archive sicopolis_v32.tgz from the SICOPOLIS web page (<http://sicopolis.greveweb.net/>).

2. Unpacking:

```
tar -x -v -z -f sicopolis_v32.tgz
```

3. You should then have a new directory “sicopolis” that contains the entire program package. In this directory, execute the script (bash) `copy_templates.sh`:

```
./copy_templates.sh
```

3 Initial configuration

3.1 Manual configuration [recommended]

1. Locate the file `sico_configs.sh` in the directory `sicopolis/runs`, and open it with a text editor.

2. Set the flags `NETCDF_FLAG`, `LIS_FLAG`, `OPENMP_FLAG` and `LARGE_DATA_FLAG` according to your needs.

Pre-selected is `NETCDF_FLAG='yes'` (with NetCDF) and the rest ‘no’.

If you wish to create output in native binary format rather than in NetCDF format, set `NETCDF_FLAG` to ‘no’. For simulations with shallow-shelf dynamics (for floating ice) or hybrid shallow-ice/shelfy-stream dynamics (for grounded ice), `LIS_FLAG` and `OPENMP_FLAG` must both be set to ‘yes’. For high-resolution simulations (e.g., Greenland/5 km or Antarctica/10 km), `LARGE_DATA_FLAG` must be set to ‘yes’.

3. If `NETCDF_FLAG='yes'`, set `NETCDFHOME` to the correct path of your NetCDF installation.

If `LIS_FLAG='yes'`, set `LISHOME` to the correct path of your Lis installation.

4. Depending on your system, some additional settings might have to be added in `sico_configs.sh` (`module load` commands for dynamic loading etc.).

3.2 Building with GNU Autotools

As an alternative to the manual configuration described in Section 3.1, SICOPOLIS can also be built with the GNU Autotools. The configuration file `sico_configs.sh` can then be ignored. Instead, it is required to execute the script `autogen.sh`:

```
./autogen.sh [ --with-netcdf=/my_netcdf_path \  
              --with-lis=/my_lis_path \  
              --enable-openmp=yes \  
              --enable-large_data=yes ]
```

The four options have the same function as the flags discussed in Section 3.1. If the NetCDF library is linked, set `/my_netcdf_path` to the correct path of your NetCDF installation. If the Lis library is linked, set `/my_lis_path` to the correct path of your Lis installation.

4 Files and directories in “sicopolis”

- **runs:**

Configuration file `sico_configs.sh`

(only needed for manual configuration; see Section 3.1).

Shell script (bash) `sico.sh` for running a single simulation.

Shell scripts (bash) `multi_sico_1.sh` and `multi_sico_2.sh` for running multiple simulations by repeated calls of `sico.sh`.

Subdirectory **headers**: specification files `sico_specs_run_name.h` (*run_name*: name of run).

- File included for run `v32_emtp2sge25_expA`
 - EISMINT Phase 2 Simplified Geometry Experiment A, resolution 25 km, $t = 0 \dots 200$ ka (Payne et al. 2000).
The thermodynamics solver for this run is the one-layer melting-CTS enthalpy scheme (ENTM), while all other runs employ the polythermal two-layer scheme (POLY) (Greve and Blatter 2016).
- File included for run `v32_grl20_ss25ka`
 - Greenland ice sheet, resolution 20 km, short steady-state run ($t = 0 \dots 25$ ka) for modern climate conditions (unpublished).
- File included for run `v32_ant40_ss25ka`
 - Antarctic ice sheet without ice shelves, resolution 40 km, short steady-state run ($t = 0 \dots 25$ ka) for modern climate conditions (unpublished).
- Files included for runs `v32_grl20_paleo01_init` and `v32_grl20_paleo01`
 - Greenland ice sheet, resolution 20 km, $t = -422 \dots -250$ ka for the spin-up run (`..._init`), $t = -250 \dots 0$ ka for the main run [similar to run `hf_pmod2` by Greve (2005), but with the new topography data based on Bamber et al. (2013)].

- Files included for runs v32_grl20_wre1000 and v32_grl10_wre1000
 - Greenland ice sheet, resolution 20 / 10 km, $t = 1990 \text{ CE} \dots 2350 \text{ CE}$ [similar to run #11 by Greve and Otsu (2007), but with the new topography data based on Bamber et al. (2013)].
- File included for run v32_nhem80_nt012_new
 - northern hemisphere, resolution 80 km, $t = -250 \dots 0 \text{ ka}$ [similar to run nt012 by Greve et al. (1999)].
- Files included for runs v32_grl20_sr_paleo44_xxx (xxx = init100a, fixtopo1, fixtopo2, 100a), v32_grl20_sr_future44_ctl and v32_grl20_sr_future44_c2
 - Greenland ice sheet, low-resolution (20 km) versions of the paleoclimatic spin-up ($t = -250 \dots 0 \text{ ka}$), experiment CTL ($t = 0 \dots 500 \text{ a}$, constant climate control run) and experiment C2 [$t = 0 \dots 500 \text{ a}$, $1.5 \times \text{AR4}$ climate forcing (based on the A1B emission scenario) over the first 94 years, then held steady] carried out for the SeaRISE community effort (Greve and Herzfeld 2013).
- Files included for runs v32_ant40_sr_spinup01_xxx (xxx = init100a, fixtopo1, fixtopo2, fixtopo3, 20a), v32_ant40_sr_future01_ctl and v32_ant40_sr_future01_m2
 - Antarctic ice sheet with ice shelves, low-resolution (40 km) versions of the paleoclimatic spin-up ($t = -250 \dots 0 \text{ ka}$), experiment CTL ($t = 0 \dots 500 \text{ a}$, constant climate control run) and experiment M2 ($t = 0 \dots 500 \text{ a}$, sub-ice-shelf melting increased to $20 \text{ m i. eq. a}^{-1}$) carried out for the SeaRISE community effort (Sato and Greve 2012).
- File included for run v32_heino50_st
 - ISMIP HEINO standard run ST, resolution 50 km, $t = 0 \dots 200 \text{ ka}$ (Calov et al. 2010).

- **src:**

Directory that 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 hemisphere.
- Accordingly subdirectories subroutines/asf, scand, tibet, nmars and smars for Austfonna, Scandinavia, Tibet and the north and south polar caps of Mars, respectively.
- Subdirectory **subroutines/xyz**: see Appendix A.

- **sico_in**:

Directory that 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 **empt2sge**: 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.
- Accordingly subdirectories asf, scand, tibet, nmars and smars for Austfonna, Scandinavia, Tibet and the north and south polar caps of Mars, respectively.
- Subdirectory **xyz**: see Appendix A.

- **sico_out**:

Empty directory into which output files of SICOPOLIS simulations are written.

- **docu**:

Directory that contains some documentation.

- Subdirectory **quick_start**:
L^AT_EX source for this manual (PDF must be built with make).
- Subdirectory **doxygen**:
Documentation created by Doxygen.
 - * html/index.html → Source code browser.
 - * latex/refman.pdf (must be built with make) → Reference manual.

- **tools**:

See Sects. 7 and 8.

- **license:**

Directory that contains a copy of the GNU General Public License (version 3).

5 How to run a simulation

1. It is recommended to run SICOPOLIS with output in NetCDF format. If you wish to create output in native binary format instead, set NETCDF to 1 (rather than 2) in all specification files *sico_specs_run_name.h*.

2. In order to run simulation *v32_grl20_ss25ka*, use the script *sico.sh*. The command is

```
(./sico.sh [-z] -m v32_grl20_ss25ka) >out_001.dat 2>&1 &
```

(from directory *sicopolis/runs*, bash required). Accordingly for the other simulations. The option *-z* is only required if SICOPOLIS was configured manually. If it was built with the GNU Autotools, omit it. For further options, try *./sico.sh -h*.

3. Alternatively, if you prefer to run all simulations consecutively, execute the script *multi_sico_1.sh*:

```
(./multi_sico_1.sh [-z]) >out_multi_100.dat 2>&1 &
```

Again, the option *-z* is only required if SICOPOLIS was configured manually. For further options, try *./multi_sico_1.sh -h*.

Computing times

The approximate computing times for the simulations, run with the Intel Fortran Compiler 15.0.3 for Linux (optimisation options *-xHOST -O3 -no-prec-div*) on a 12-Core Intel Xeon E5-2697 v2 (2.7 GHz) PC under openSUSE 13.1 (64 bit), are listed in Table 1 (Appendix B).

6 Output files

Output files are written by default to the directory *sicopolis/sico_out/run_name* (this can be changed with the *-d* option). Four types are produced:

- **run_name.log:**

ASCII file that lists the main specifications of simulation *run_name*.

- **run_name.ser:**

Time-series file (ASCII) that contains global parameters:

- Time, t
- Surface-temperature anomaly, D_Ts , or glacial index, $glac_ind$ (forcing)
- Sea level, z_sl (forcing)
- Total ice volume, V
- Grounded ice volume, V_g
- Floating ice volume, V_f
- Total ice area, A
- Grounded ice area, A_g
- Floating ice area, A_f
- Maximum ice thickness, H_max
- Maximum ice elevation, zs_max
- Volume of the temperate ice, V_t
- Freshwater production due to melting and calving, V_fw
- Sea-level equivalent of ice volume, V_sle
- Area covered by temperate ice, A_t
- 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

● **run_name.core:**

Time-series file (ASCII) that contains for selected locations xxx:

- Time, t
- Surface-temperature anomaly, D_Ts , or glacial index, $glac_ind$ (forcing)
- Sea level, z_sl (forcing)
- Thickness, H_xxx
- Surface velocity, v_xxx
- Basal temperature, T_xxx
- Basal frictional heating, Rb_xxx

For the Greenland ice sheet, these data are written for six locations:

GRIP (xxx=GR), GISP2 (xxx=G2), Dye 3 (xxx=D3), Camp Century (xxx=CC), NorthGRIP (xxx=NG), NEEM (xxx=NE).

For the Antarctic ice sheet, these data are written for six locations:

Vostok (xxx=Vo), Dome A (xxx=DA), Dome C (xxx=DC), Dome F (xxx=DF), Kohlen (xxx=Ko), Byrd (xxx=By).

For the northern hemisphere and the EISMINT Phase 2 Simplified Geometry Experiments, no such data are written.

- **run_name0001.nc/.erg, run_name0002.nc/.erg, ...:**

Complete set of fields (topography, velocity, temperature etc., written either in NetCDF (*.nc) or in native binary (*.erg) format; see subroutines `output_nc` and `output1`, respectively) for selected time slices defined in specifications file. For example, simulation `v32_grl20_ss25ka` produces three files `v32_grl20_ss25ka0001.nc`, `v32_grl20_ss25ka0002.nc` and `v32_grl20_ss25ka0003.nc`, which correspond to $t = 0$, 10 ka and 25 ka, respectively.

7 Plotting with SICOGRAPH

The output described in Sect. 6 can be visualised with any plotting tool at the user's preference. One possibility is to use SICOGRAPH, which is part of the SICOPOLIS package and based on the Generic Mapping Tools GMT (<http://gmt.soest.hawaii.edu/>).

7.1 Installation

1. If you do not have an installation of GMT version 4.x yet, download and install the latest version according to the instructions on the GMT web site.
2. SICOGRAPH is included in the subversion repository of SICOPOLIS. You'll find it in the directory `sicopolis/tools/sicograph`.

7.2 Producing plots

In order to plot the output of simulation `v32_grl20_ss25ka`, go to the directory `sicopolis/tools` and execute the script `tools.sh` interactively:

```
./tools.sh [-z] -p sicograph -m v32_grl20_ss25ka
```

(bash required; accordingly for the other simulations). The option `-z` is only required if

SICOPOLIS was configured manually. If it was built with the GNU Autotools, omit it. For further options, try `./tools.sh -h`.

You'll get a menu that allows you to choose the type of plot you wish to produce. For example, try the option

(1) Ice-surface topography

and enter

Number of time-slice file (with leading zeros, 4 digits) > 0003

Time-slice file contains:

(0) only 2-d arrays, (1) full set of 2-d and 3-d arrays > 1

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

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

You will find the plot in the subdirectory `sicograph/gmt_scripts/plots` as file `v32_grl20_ss25ka0003_zs.eps` (in EPS format). As a second example, try

(41) Time series

and in the following sub-menu choose

(5) Total ice volume

This produces the file `v32_grl20_ss25ka_V_tot.eps` in the subdirectory `sicograph/gmt_scripts/plots`.

7.3 Manipulating plot appearance

For all types of plots, the files in the subdirectory `sicograph/parameter_files` control the limits and labels of the x - and y -axes. In addition, for the plan-view plots, the files in the subdirectory `sicograph/gmt_scripts/cpt` control the colour scales (`*.cpt`) and contour levels (`*.zzz`). If a file is missing, the corresponding parameters are computed automatically.

8 Some useful tools

In addition to SICOGRAPH, the directory `sicopolis/tools` contains some further useful tools.

- Program **make_searise_output**:

Generating SeaRISE output (see <http://tinyurl.com/srise-umt-out>) from the NetCDF time-slice files produced by SICOPOLIS (see Sect. 6). For simulation *run_name*, to be executed by

```
./tools.sh [-z] -p make_searise_output -m run_name
```

The option `-z` is only required if SICOPOLIS was configured manually. If it was built with the GNU Autotools, omit it. For further options, try `./tools.sh -h`.

- Program **resolution_doubler**:

Doubling the horizontal resolution of a NetCDF time-slice output file produced by SICOPOLIS (see Sect. 6). For simulation *run_name*, to be executed by

```
./tools.sh [-z] -p resolution_doubler -m run_name
```

Again, the option `-z` is only required if SICOPOLIS was configured manually. For further options, try `./tools.sh -h`.

For example, run `v32_grl10_wre1000` (10 km resolution) requires the resolution-doubled final output of run `v32_grl20_paleo01` (20 km resolution) as initial condition. In order to create it, execute the resolution doubler for run `v32_grl20_paleo01` and enter

```
Number of time-slice file (with leading zeros, 4 digits) > 0003
```

This will convert the original time-slice file `v32_grl20_paleo010003.nc` to the resolution-doubled file `v32_grl20_paleo01_dbl_0003.nc` that serves as initial conditions for run `v32_grl10_wre1000`.

- Subdirectory **matlab**:

MATLAB functions for reading SICOPOLIS output files (see Sect. 6). These functions can be used for plotting with MATLAB.

A Domain XYZ

This framework allows creating new domains (Laurentide ice sheet, simple testing geometry etc.). The directory `sicopolis/src/subroutines/xyz`, which hosts the domain-specific subroutines, is by default empty. If you want to create a new domain, copy the subroutines from the most similar existing domain (northern hemisphere, EISMINT etc.), e.g.:

```
cp sicopolis/src/subroutines/nhem/*.F90 \
   sicopolis/src/subroutines/xyz/
```

Then modify the routines according to your needs. Input files (topography etc.) must be placed in `sicopolis/sico_in/xyz` and specified in the run-specification header file `*.h` as usual. The domain must be defined by the domain code `'#define XYZ'` in the header file. If the new domain requires new global variables, they can be defined in the module `sicopolis/src/subroutines/xyz/sico_vars.F90`.

The subroutines for ISMIP HEINO are available in `sicopolis/src/subroutines/xyz/heino`, and the input files are in `sicopolis/sico_in/xyz`. If you copy the subroutines from `sicopolis/src/subroutines/xyz/heino` to `sicopolis/src/subroutines/xyz`, you can run ISMIP HEINO experiments (e.g., the run `v32_heino50.st` for which a header file is available).

B Table: Simulations and computing times

Run	Model time	Time step [†]	CPU time [‡]
v32_emtp2sge25_expA	200 ka	20 a	6.9 min
v32_grl20_ss25ka	25 ka	5 a	7.4 min
v32_ant40_ss25ka	25 ka	10 a	7.0 min
v32_grl20_paleo01_init	172 ka	5 a	1.0 hrs
v32_grl20_paleo01	250 ka	5 a	1.5 hrs
v32_grl20_wre1000	360 a	5 a	7.4 sec
v32_grl10_wre1000*	360 a	1 a	2.5 min
v32_nhem80_nt012_new	250 ka	5 a	1.9 hrs
v32_grl20_sr_paleo44_init100a	100 a	5 a	4.4 sec
v32_grl20_sr_paleo44_fixtopo1	125 ka	5 a	0.7 hrs
v32_grl20_sr_paleo44_fixtopo2	124.9 ka	5 a	0.7 hrs
v32_grl20_sr_paleo44_100a	100 a	5 a	2.6 sec
v32_grl20_sr_future44_ctl	500 a	1 a	0.8 min
v32_grl20_sr_future44_c2	500 a	1 a	0.8 min
v32_ant40_sr_spinup01_init100a	100 a	2 / 10 a [†]	16.5 sec
v32_ant40_sr_spinup01_fixtopo1	5 ka	1 / 5 a [†]	0.5 hrs
v32_ant40_sr_spinup01_fixtopo2	120 ka	2 / 10 a [†]	5.4 hrs
v32_ant40_sr_spinup01_fixtopo3	124.98 ka	2 / 10 a [†]	5.7 hrs
v32_ant40_sr_spinup01_20a	20 a	1 / 5 a [†]	7.2 sec
v32_ant40_sr_future01_ctl	500 a	1 / 5 a [†]	2.7 min
v32_ant40_sr_future01_m2	500 a	1 / 5 a [†]	1.5 min

Table 1: Model times, time steps and computing (CPU) times for the simulations, run with the Intel Fortran Compiler 15.0.3 for Linux (optimisation options `-xHOST -O3 -no-prec-div`) on a 12-Core Intel Xeon E5-2697 v2 (2.7 GHz) PC under openSUSE 13.1 (64 bit).

[†]: If one value is given, this is the common dynamic (velocity, ice thickness) and thermodynamic (temperature, water content, age) time step. If two values are given (marked by the dagger ([†]) symbol), the first one is the dynamic, the second one the thermodynamic time step.

[‡]: All runs were done on one core only. The `v32_ant40_sr_xxx` runs that include ice shelves can be done on multiple cores using OpenMP for the SSA solver; however, at the employed, rather low resolution of 40 km the solver does not scale well, and the gain in wall clock time by using multiple cores is very small.

*: For this run, see the remark in Sect. 8 under the item “Program `resolution_doubler.F90`”.

References

- Bamber, J. L., J. A. Griggs, R. T. W. L. Hurkmans, J. A. Dowdeswell, S. P. Gogineni, I. Howat, J. Mouginot, J. Paden, S. Palmer, E. Rignot and D. Steinhage. 2013. A new bed elevation dataset for Greenland. *Cryosphere*, **7** (2), 499–510. doi:10.5194/tc-7-499-2013.
- Calov, R., R. Greve, A. Abe-Ouchi, E. Bueller, P. Huybrechts, J. V. Johnson, F. Pattyn, D. Pollard, C. Ritz, F. Saito and L. Tarasov. 2010. Results from the Ice-Sheet Model Intercomparison Project – Heinrich Event INtercOmparison (ISMIP HEINO). *J. Glaciol.*, **56** (197), 371–383. doi:10.3189/002214310792447789.
- Greve, R. 2005. Relation of measured basal temperatures and the spatial distribution of the geothermal heat flux for the Greenland ice sheet. *Ann. Glaciol.*, **42**, 424–432. doi:10.3189/172756405781812510.
- Greve, R. and H. Blatter. 2016. Comparison of thermodynamics solvers in the polythermal ice sheet model SICOPOLIS. *Polar Sci.*, **10** (1), 11–23. doi:10.1016/j.polar.2015.12.004.
- Greve, R. and U. C. Herzfeld. 2013. Resolution of ice streams and outlet glaciers in large-scale simulations of the Greenland ice sheet. *Ann. Glaciol.*, **54** (63), 209–220. doi:10.3189/2013AoG63A085.
- Greve, R. and S. Otsu. 2007. The effect of the north-east ice stream on the Greenland ice sheet in changing climates. *Cryosphere Discuss.*, **1** (1), 41–76. doi:10.5194/tcd-1-41-2007.
- Greve, R., K.-H. Wyrwoll and A. Eisenhauer. 1999. Deglaciation of the Northern Hemisphere at the onset of the Eemian and Holocene. *Ann. Glaciol.*, **28**, 1–8. doi:10.3189/172756499781821643.
- Payne, A. J., P. Huybrechts, A. Abe-Ouchi, R. Calov, J. L. Fastook, R. Greve, S. J. Marshall, I. Marsiat, C. Ritz, L. Tarasov and M. P. A. Thomassen. 2000. Results from the EISMINT model intercomparison: the effects of thermomechanical coupling. *J. Glaciol.*, **46** (153), 227–238. doi:10.3189/172756500781832891.
- Sato, T. and R. Greve. 2012. Sensitivity experiments for the Antarctic ice sheet with varied sub-ice-shelf melting rates. *Ann. Glaciol.*, **53** (60), 221–228. doi:10.3189/2012AoG60A042.