

C-GOLDSTEIN

A brief guide to running the model

Gemma Stephenson

April 11, 2008

Abstract

The steps required to build, configure, execute and utilise the C-GOLDSTEIN Earth System Model are described herein. In addition to the model itself, the configuration tool `mkgo.in` is described. This tool can be used to generate multiple configuration files required for computer experiments. This is not an official manual to the C-GOLDSTEIN software and is not intended to be a complete user guide.

Contents

1	Introduction	4
1.1	Purpose of this document	4
1.2	Scope of this document	4
1.3	Structure of this document	4
1.4	The C-GOLDSTEIN model	5
2	Build and Installation	6
2.1	Requirments	6
2.2	Building	6
3	Configuration	7
3.1	Ancillary data	7
3.2	Inputs overview	7
3.3	Time step input parameters	8
3.4	I/O input parameters	9
3.5	Ocean input parameters	9
3.6	Atmosphere input parameters	11
3.7	Sea Ice input parameters	12
4	Multiple run executions	12
4.1	The mkgoin program	12
4.2	Design file	13
4.3	Modifying mkgoin	14
4.4	Executing C-GOLDSTEIN with mkgoin generated inputs	14
4.5	Spinup without observed CO ₂	14

4.6	Spinup with observed CO ₂ (1800 - 2000)	15
4.7	Forecast	15
5	Output	16
5.1	Location	16
5.2	Plots	16
A	Directory Structure	19
B	Example of an input goin file for a version 1 Standard Spinup Experiment of C-GOLDSTEIN	20
C	Example of an input goin file for a Version 2 Standard Spinup Experiment of C-GOLDSTEIN	21
D	Fortran program: mkgoin.f	21
E	Example of an input goin file for a continued run to the year 3000	25

1 Introduction

1.1 Purpose of this document

The C-GOLDSTEIN software was not developed to a production level and the code remains in a development phase. This has caused some confusion as to the correct installation and execution procedures for different hardware platforms. The purpose of this document is to provide more detailed and up-to-date guidelines on compiling and executing C-GOLDSTEIN, than available in Marsh *et al.* (2002). The reader is assumed to have only a very basic understanding of the Unix environment, in which the model operates, and the FORTRAN language in which the model is coded. Accordingly some knowledge of Earth system modelling is assumed, but only at the most basic level.

1.2 Scope of this document

This document fundamentally consists a collection of observations made during attempts to perform the procedures described. It is not a definitive user guide and should not be treated as such. The author did not design, and is not involved in the development of the C-GOLDSTEIN software. All code and guidelines are intended as suggestions only; this document does not constitute a definitive method for running C-GOLDSTEIN.

1.3 Structure of this document

This section describes the purpose, scope and structure of this document along with a very brief introduction to the C-GOLDSTEIN software. Section 2 details the C-GOLDSTEIN build and installation procedures. Section 3 contains guidelines on how to run the model as a single run, or in a batch mode using the `mkgo.in` tool. A list of input variables is given along with a corresponding description. This is followed by a suggested method for running a spinup and subsequent runs of the model to the year 2000 and into the future. Finally, Section 5 provides details on understanding the output and methods to extract required information from the output.

1.4 The C-GOLDSTEIN model

The C-GOLDSTEIN software encodes a computationally fast Earth System Model (ESM) developed by R. Marsh and N. R. Edwards (Marsh *et al.*, 2002). It comprises three coupled model components:

- The Global Ocean-Linear Drag Salt and Temperature Equation Integrator (GOLDSTEIN) ocean model.
- An atmospheric Energy Moisture Balance Model (EMBM).
- A simple sea ice model.

Details of the model computational components, variable parameters, input and output data are given in Marsh *et al.* (2002) along with a brief description of installation and execution procedure. This document refers to two versions of C-GOLDSTEIN:

- Version 1 is the basic model.
- Version 2 incorporates observed carbon dioxide values up to the year 2000 and has forecasting capabilities thereafter.

2 Build and Installation

This section outlines the procedures required to build a C-GOLDSTEIN executable.

2.1 Requirments

In order to build C-GOLDSTEIN the following software must be available:

- Unix operating system (SOLARIS 9, Suse 9 and Red Hat 10 are known to work)
- The Make tool (GNU is known to work)
- A text editor
- FORTRAN 77 compiler (G77, gfortran and Intel Fortran 9.0 are known to work)
- FORTRAN 90 compiler (Intel Fortran 9.0 and gfortran are known to work)

2.2 Building

The binary (or executable) is generated automatically within the terminal with the issuing of the following statement.

```
$> make goldstein
```

This instructs the **Make** program to perform the actions listed in the **Makefile** file. Unless your system is identical to the system on which C-GOLDSTEIN was developed (unlikely!), you will need to modify this **Makefile** in order to build C-GOLDSTEIN on your system. **Makefile** is within the **genie-cgoldstein** directory.

The correct FORTRAN compiler for your system may be selected by commenting out the other listed compilers. It is important to note that the **r8** flag, or its equivalent, is required for all compilers. Failure to follow this instruction will result in multiple underflow errors. Correct execution of the **make goldstein** command will result in the **goldstein** binary. You may have to set the executable file permission in order to perform later tasks. This may be achieved with the command

```
$> chmod o+x goldstein
```

Users familiar with Makefile constructs should have no difficulty in producing a valid executable.

3 Configuration

This section outlines the procedures required to configure the C-GOLDSTEIN model. Each instance of C-GOLDSTEIN execution requires an input configuration file, along with various data sets. A method for automatic generation of the multiple input configuration files needed for ensemble execution is described. The input configuration file for any run is described in Section 3.2.

3.1 Ancillary data

C-GOLDSTEIN requires various ancillary input files such as `taux.u.interp` and `uncep.silo`. These should be located in the `genie-cgoldstein` directory and do not require modification by the user.

3.2 Inputs overview

The C-GOLDSTEIN executable requires a `go.in` file, containing values for input parameters and various time-step configurations. The contents of a `go.in` file may be assigned to five groups:

1. Time step input parameters - these control the duration of a run and temporal resolution of the model outputs. See Section 3.3.
2. I/O input parameters - input/output filenames etc. See Section 3.4.
3. Ocean input parameters - see Section 3.5 and Table 1.
4. Atmosphere input parameters - see Section 3.6 and Table 2.

5. Sea Ice input parameters - see Section 3.7 and Table 3.

3.3 Time step input parameters

These parameters control the temporal resolution of the recorded output data and the duration of the run.

`nsteps`

This specifies the length of the run. There are 100 timesteps per year, as detailed below, therefore the value of `nsteps` is 100 times the required number of years.

`npstp`

During a run, diagnostic information of the run is written to a `out` file when `nsteps` is at multiples of `npstp`.

`iwstp`

When `nsteps` is at multiples of `iwstp`, files are saved so the run could be restarted from this point if required.

`itstp`

When `nsteps` is at multiples of `itstp` output files are written. For example, if data were required once per year `itstp` would be 100.

`ianav`

This is an annual average.

`timesteps per year`

100

`A/0 dt ratio`

Atmosphere to ocean timestep ratio. This has a default value of 5.

`rel`

Velocity Relaxation (as defined by Marsh *et al.* (2002) in Appendix A) with default value 0.90.

3.4 I/O input parameters

These parameters dictate the location of various input and outputs.

`n/c`

This specifies whether the run is a new or continuing run.

`output file number`

Output files will be identified with this.

`input file name`

If `c` was specified (a continued run), `input file name` is the restart file from which this run will continue.

3.5 Ocean input parameters

The following table lists the many parameters used to alter the behaviour of the ocean in the model. Note some parameter values are nonsensical and intended to become meaningful through spinup.

Input	Input Name in Model	Default Value
Initial ocean temperature in northern hemisphere	temp0	10
Initial ocean temperature in southern hemisphere	temp1	10
Wind stress scale	scl_tau or scf	2.00
Ocean horizontal diffusivity	ocean diffusivity iso or diff1	2000.
Ocean vertical diffusivity	ocean diffusivity dia or diff2	1e-4.
Ocean drag coefficient	inverse minimum drag in days or adrag	2.5
Initial humidity over ocean	relh0_ocean	0.
Atlantic-to-Pacific freshwater flux adjustment values	extra1a, extra1b, extra1c, extra1d	-0.03, 0.17, 0.18, 0
Scaling factor for Atlantic to Pacific moisture flux	scl.fwf	1

Table 1: Ocean Input Parameters for C-GOLDSTEIN

3.6 Atmosphere input parameters

The following table lists the parameters used to alter the behaviour of the atmosphere in the model. Note some parameter values are nonsensical and intended to become meaningful through spinup.

Input	Input Name in Model	Default Value
Atmospheric heat diffusivity	atm. diff. amp. for T or diffamp(1)	5.0e6
Atmospheric moisture diffusivity	atm. diff. amp. for q or diffamp(2)	1.0e6
Width of atmospheric heat diffusivity profile	atm. diff. amp. dist'n width or diffwid	1.0
Slope of atmospheric heat diffusivity profile	atm. diff. amp. slope or difflin	0.1
Zonal heat advection factor	atm. advection factors for T_z or betaz1	0.
Meridional heat advection factor	atm. advection factors for T_m	0.
Zonal moisture advection factor	atm. advection factors for q_z or betaz2	0.4
Meridional moisture advection factor	atm. advection factors for q_m	0.4
Scales co2 concentration relative to 350ppm	scl_co2	1.0
Specifies a compound annual % rate of increase	pc_co2_rise	0.0
Climate sensitivity	delf2x	5.77
Solar Constant	solconst	1368
Initial temperature of the atmosphere	tatm	0.
Initial humidity over land	relh0_land	0.
Threshold Relative Humidity above which precipitation occurs	rmax	0.85
e-folding timescale of Carbon removal	t_absorb	150.

Table 2: Atmosphere Input Parameters for C-GOLDSTEIN

3.7 Sea Ice input parameters

The following table lists the parameters used to alter the behaviour of sea ice in the model.

Input	Input Name in Model	Default Value
Sea ice diffusivity	sea-ice eddy diffusivity or <code>diffsic</code>	2000.
Sensitivity of Greenland Ice Sheet melt to warming	<code>k_gis</code>	0.01

Table 3: Sea Ice Input Parameters for C-GOLDSTEIN

4 Multiple run executions

4.1 The `mkgo` program

Once an experimental design has been chosen the corresponding `go` input file must be created for each instance of the model. The `mkgo` program provides a simplified mechanism for generating multiple `go` files.

The procedure for `mkgo` operation is as follows:

1. Ensure the existence of the directories `../go/` and `../goout/`.
2. Provide a `go.std` file. This is a typical `go` file which will provide all input parameters not being changed in the experiment.
3. Generate a design file listing the value of each parameter being changed in the experiment. Each run of the model requires a value for all variables being changed. This file is described in Section 4.2.
4. Modify the `mkgo.f` source, identifying the parameters to be changed in the experiment. This is further described in Section 4.3.
5. Build and execute the `mkgo` program to generate the multiple `go` files.

6. Execute the C-GOLSTEIN model once for each `goin` file. This is further described in Section 4.4.

4.2 Design file

The design lists, for each model run, a value for each parameter being varied at any point in the experiment. Any parameter that remains constant throughout the experiment will be taken from the `goin.std` file. Examples of `goin.std` to generate a batch of spinup input files in version 1 or 2 of C-GOLDSTEIN are shown in Appendices B and C respectively. Differences between `goin` files for versions 1 and 2 are explained further in Section 4.6.

The design file itself is simply a space delimited text file enumerating each combination of input parameters required in the experiment. For example, an experiment consisting of 10 runs, varying only ocean drag coefficient, atmospheric moisture diffusivity and climate sensitivity would require a design file as shown in Figure 1. This file must be called `Design` and located in the `genie-cgoldstein` directory. The first column consists of the values for the ocean drag coefficient, the second for atmospheric moisture diffusivity and the third for climate sensitivity.

```
3.496280 1573731 3.696902
3.182065 1530661 4.249042
4.083741 1596162 3.362610
4.290861 1637941 5.113775
3.417705 1744212 4.840546
3.758303 1650016 5.653014
3.571851 1709990 7.804285
2.994917 1589768 5.515842
3.354139 1674013 6.146303
3.790954 1618322 4.405619
```

Figure 1: Example of a design file

Note that there is no header information. Accordingly the read routine in `mkgoin.f` must be modified for each experiment, as shown in Section 4.3.

4.3 Modifying mkgoin

Depending on which input parameters are being varied, the `mkgoin.f` source code may need to be altered and recompiled. This source code is located in the `genie-cgoldstein` directory. In the previous example the parameters ocean drag coefficient, atmospheric moisture diffusivity and climate sensitivity were altered. This would require the changes shown in Figure 2 to be made to the line beginning `read(97,)` of `mkgoin.f`.

```
read(97,*,iostat=iend,end=99)
&   adrag,diffamp(2),delf2x
```

Figure 2: Example of adapted part of `mkgoin.f`

The FORTRAN source code for `mkgoin.f` is given in Appendix D.

4.4 Executing C-GOLDSTEIN with mkgoin generated inputs

The `mkgoin` program will generate multiple `goin` files. The previous example will result in the files `../goin/goin.0` to `../goin/goin.9`.

The first instance of the model may be executed as follows:

```
goldstein < ../goin/goin.0 > ../goout/out.0
```

This will run C-GOLDSTEIN at the first input configuration, `goin.0`, located in the directory `goin`. At multiples of `npstp` information will be written to `out.0`, located in the directory `goout`. Output will be written to the `results/` directory as detailed in Section 5.

4.5 Spinup without observed CO₂

It is recommended to spinup C-GOLDSTEIN for 4000 years. An example of a `goin` file to run such a spin up with default parameter values can be found in Appendix B. To

run the model to equilibrium without incorporating observed values of carbon dioxide, version 1 of C-GOLDSTEIN must be executed.

4.6 Spinup with observed CO₂ (1800 - 2000)

To spinup C-GOLDSTEIN with observed CO₂ values from 1800 to the year 2000, version 2 of the model must be executed. This version, in addition to extended source code, contains the file `co2_historical.dat` within the `genie-cgoldstein` directory. This file consists of observed values of atmospheric carbon dioxide, as reported by Johnston (2007). Ice core observations are used pre 1958. The layout of a `go.in` file for version 2 is slightly different to that for version 1, due to minor variations in the source code of the two models. A `go.in` file for version 2 also requires further input parameters in addition to those existing in a `go.in` file for version 1. Two additional inputs are `t_absorb` and `k_gis`, see Sections 3.6 and 3.7 respectively for details. An example of a `go.in` file to spinup to the year 2000 is given in Appendix C.

4.7 Forecast

C-GOLDSTEIN can forecast past 2000 after an appropriate spinup. Input parameters for these forecasts are based on IPCC emission scenarios (IPCC, 2000). The data for the 6 emissions scenarios detailed by IPCC (2000) are located in the `genie-cgoldstein` directory for version 2.

Having chosen an emissions scenario, (or created a new one), there are 2 methods to implement it as detailed below.

1. In the `genie-cgoldstein` directory copy the chosen emissions scenario to `emissions.dat`.
2. In the `genie-cgoldstein` directory go to `gseta.F`. Where currently `emissions.dat` is opened change the file name to the chosen scenario.

An example of a `go.in` file to continue the previously spunup version 2 model run, from the `go.in` file in Appendix C, up to the year 3000 is shown in Appendix E.

5 Output

5.1 Location

A directory called `results` must be created along with a subdirectory with a three character name for the ensemble of runs. For an ensemble of spinup runs, for example, the directory may be `results/spn`. The model will then write output data to this directory. Output data is in ascii, identified by the 3 character name given to each run in `mkgo.in.f`, see Section 4.1. The first member of an ensemble typically has the name 000 and for a run of this name, Table 4 gives details of the output files. Temperature is measured in Celcius and salinity is measured in psu.

5.2 Plots

Various fields and time series can be plotted from the data. A number of `matlab` programs for creating these plots are located in the `genie-cgoldstein` directory. In particular, `tstq.m` is a program to visualise temperature and salinity and `tplot` produces various time series data.

Output file	Description
000.n	'Restart' files which can be used as input files in continuing runs
000.t	Mean ocean temperature in various regions
000.s	Mean ocean salinity in various regions
000.airt	Global mean air temperature
000.q	Global mean specific humidity
000.opsi	Meridional overturning streamfunction
000.zpsi	Zonal overturning streamfunction
000.psi	Barotropic streamfunction
000.rho	Density
000.cost	2d array showing the frequency of convection during the run
000.relh	Relative humidity
000.fx0a	Net heat flux into atmosphere
000.pptn	Precipitation rate
000.evap	Evaporation rate
000.runoff	River runoff
000.fwfxneto	Net freshwater flux into ocean
000.fx0neto	Net heat flux into ocean from atmosphere and sea ice
000.fofy	Poleward heat flux in Atlantic, Pacific and total

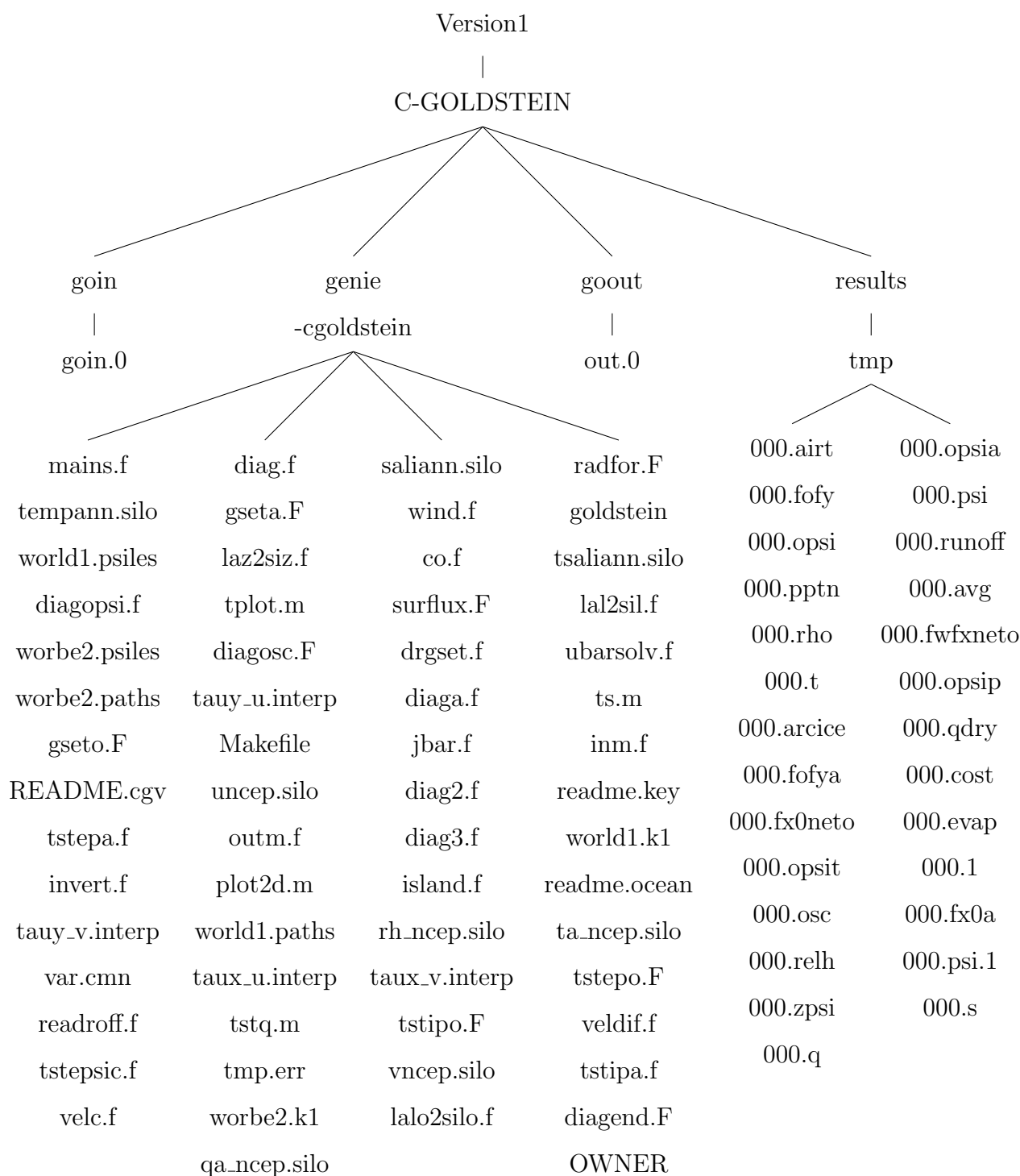
Table 4: Output files

Acknowledgements

The author would like to acknowledge the help of Robert Marsh, who provided all code and much help and guidance. The author would also like to acknowledge David Poulter who helped with the structure and style of this document.

Appendix

A Directory Structure



B Example of an input goin file for a version 1 Standard Spinup Experiment of C-GOLDSTEIN

```
400001 400000 20000 100 100 nsteps npstp iwstp itstp ianav
n new/continuing
100 5 timesteps per year and A/O dt ratio
10 10 0.90 2.00 temp0 temp1 rel scl_tau
2000. 1e-4 ocean diffusivites iso/dia (or horiz/vert)
2.5 inverse minimum drag in days
5.0e6 1e6 1.0 0.1 atm. diff. amp. for T & q & dist'n width, slope
0. 0. 0.4 0.4 atm. advection factors for T_z, T_m, q_z, q_m
1.0 0.0 scl_co2 pc_co2_rise
2000. sea-ice eddy diffusivity
0. 0. 0. tatm relh0_ocean relh0_land
-0.03 0.17 0.18 extrala extralb extralc
tmp/tmp output file number
tmp/tmp.avg input file name
```

Figure 2: Input file for Spinup

C Example of an input goin file for a Version 2 Standard Spinup Experiment of C-GOLDSTEIN

```

400001 10000 20000 100 100  nsteps npstp iwstp itstp ianav
n                               new/continuing
100  5                          timesteps per year and A/O dt ratio
10                               temp0
10                               temp1
0.9                             rel
2                               scl_tau
2000.                          horiz ocean diff.
1.e-04                          vertical ocean diff.
2.5                             adrag
5.0e6                          diffamp(1)
1e6                             diffamp(2)
1.0                             diffwid
0.1                             difflin
0.                               betaz(1)
0.                               betam(1)
0.4                             betaz(2)
0.4                             betam(2)
1.0                             scl_co2
0.0                             pc_co2_rise
5.77                            delf2x
1                               iscen1
0.85                            rmax
2000                            diffsic
0.                               tatm
0.                               relh0_ocean
0.                               relh0_land
1                               scl_fwf
0.                               extra1d
1368                            solconst
tmp/spn                        output file number
tmp/tmp.avg                    input file name
150.                            t_absorb
1e-02                          k_gis

```

Figure 3: Input file for Spinup

D Fortran program: mkgoin.f

```

program mkgoin

real*8 tv,temp0,temp1,rel,scf,diff(2),adrag,diffamp(2),width
.,slope,betaz(2),betam(2),scl_co2,pc_co2_rise,delf2x,diffsic
.,tatm,relh0_ocean,relh0_land,etxra1a,extra1b,extra1c,scl_fwf
integer nsteps,npstp,iwstp,itstp,ndta,ianav

```

```

integer icount
character ans,lout*3,lin*6
character filegoin*20, filegoout*20, outfile*7, infile*7
&,fileerr*20, filelog*20
character*10 fmt
character*10 fmt1
character*10 fmt10
character*10 fmt100

```

c different formats for runs 0-9, 10-99, 100+

```
fmt1  ='(a13,i1.1)'
```

```
fmt10 ='(a13,i2.2)'
```

```
fmt100='(a13,i3.3)'
```

```
open(1,file='goin.std',status='old')
```

```
read(1,*)nsteps,npstp,iwstp,itstp,ianav
```

```
read(1,'(a1)')ans
```

```
read(1,*)tv,ndta
```

```
read(1,*)temp0,temp1,rel,scf
```

```
read(1,*)diff(1),diff(2)
```

```
read(1,*)adrag
```

```
read(1,*)diffamp(1),diffamp(2),width,slope
```

```
read(1,*)betaz(1),betam(1),betaz(2),betam(2)
```

```
read(1,*)scl_co2,pc_co2_rise,delf2x
```

```
read(1,*)sea_ice
```

```
read(1,*)tatm,relh0_ocean,relh0_land
```

```
read(1,*)extra1a,extra1b,extra1c,scl_fwf
```

```
read(1,'(a7)')lout
```

```
read(1,'(a11)')lin
```

c open design file

```
open(97,file='Design',status='old')
```

c input file

```
infile='tmp/tmp.avg'
```

```
icount = -1
```

```
iend=0
```

```
do while (iend.eq.0)
```

```
read(97,*,iostat=iend,end=99)
```

```
& adrag,diffamp(2),scl_fwf,delf2x,diff(2)
```

```
icount=icount+1
```

```
if(icount.le.9) write(outfile,'(a6,i1)') 'tmp/00',icount
```

```
if(icount.ge.10.and.icount.le.99)
```

```
& write(outfile,'(a5,i2)') 'tmp/0',icount
```

```
if(icount.ge.100) write(outfile,'(a4,i3)') 'tmp/',icount
```

c setup correct format

```
if(icount.le.9) then
```

```
fmt=fmt1
```

```
else if (icount.le.99) then
```

```
fmt=fmt10
```

```
else
```

```

    fmt=fmt100
endif

write(filegoin,fmt) '../goin/goin.',icount
write(filegoout,fmt) '/goout/out.',icount

open(2,file=filegoin,status='new')
write(2,*)nsteps,npstp,iwstp,itstp,ianav
write(2,'(a1)')ans
write(2,*)tv,ndta
write(2,*)temp0,temp1,rel,scf
write(2,*)diff(1),diff(2)
write(2,*)adrag
write(2,*)diffamp(1),diffamp(2),width,slope
write(2,*)betaz(1),betam(1),betaz(2),betam(2)
write(2,*)scl_co2,pc_co2_rise,delf2x
write(2,*)sea_ice
write(2,*)tatm,relh0_ocean,relh0_land
write(2,*)extra1a,extra1b,extra1c,scl_fw
write(2,'(a7)')outfile
write(2,'(a7)')infile

open(3,file=filegoout,status='new')
close(2)
close(3)

99    continue
enddo

stop
end

```


E Example of an input goin file for a continued run to the year 3000

```
100001 10000 20000 100 100  nsteps npstp iwstp itstp ianav
c                               new/continuing
100  5                          timesteps per year and A/O dt ratio
10                               temp0
10                               temp1
0.9                             rel
2                               scl_tau
2000.                          horiz ocean diff.
1.e-04                          vertical ocean diff.
2.5                             adrag
5.0e6                          diffamp(1)
1e6                             diffamp(2)
1.0                             diffwid
0.1                             difflin
0.                               betaz(1)
0.                               betam(1)
0.4                             betaz(2)
0.4                             betam(2)
1.0                             scl_co2
0.0                             pc_co2_rise
5.77                            delf2x
1                               iscen1
0.85                            rmax
2000                            diffsic
0.                               tatm
0.                               relh0_ocean
0.                               relh0_land
1                               scl_fwf
0.                               extrald
1368                            solconst
ctd/000                        output file number
tmp/spn.0                      input file name
150.                            t_absorb
1e-02                           k_gis
```

Figure 4: Input file for continued run

References

IPCC (2000). *Special Report on Emissions Scenarios*. Cambridge University Press.

Johnston, W. R. (2007). Historical data relating to global climate change. *WEB Resource*:
<http://www.johnstonsarchive.net/environment/co2table.html>.

Marsh, R., Edwards, N. R. and Shepherd, J. G. (2002). Development of a fast climate model (c-goldstein) for earth system science. *Southampton Oceanography Centre Internal Document*, **83**.

TROPICS (2007). The tapenade tutorial. *WEB Resource*: *www-sop.inria.fr/tropics/*.

Zachary, D. S. (2004). *Automatic Differentiation for convex optimization and its applications to an intermediate complexity global climate model*. Tech. rep., Logilab (LOGIstic LABoratory), University of Geneva.