Solver for Hydrologic Unstructured Domain (SHUD)

User Guide

Lele Shu (lele.shu@gmail.com)

2019-12-22

Contents

1 Overview

This file is a user guide or technical documentation of the SHUD modeling system. PDF version of the User Guide is available via :SHUD User Guide

The Solver for Hydrologic Unstructured Domain (SHUD - pronounced "SHOULD") is a multi-process, multi-scale hydrological model where major hydrological processes are fully coupled using the semi-discrete **Finite Volume Method** (FVM).

SHUDtoolbox is an open-source GIS and hydrological analysis toolbox designed for the SHUD modeling system. The SHUDtoolbox provides access to the digital data sets (terrain, forcing, and parameters) and tools necessary to drive the model, as well as a collection of GIS-based pre- and post-processing tools.

Collectively the system is referred to as the **SHUD Modeling System**.

The SHUD and SHUDtoolbox is an open-source software, freely available for download at SHUD website or Github Page along with installation and user guides.

1.1 Standing on the shoulders of giants

As a descendant of PIHM, SHUD inherits the fundamental idea of solving hydrological variables in CVODE. The code has been completely rewritten in a new programming language, with a new discretization and corresponding improvements to the underlying algorithms, adapting new mathematical schemes and a new user-friendly input/output data format. Although SHUD is forked from PIHM's track, SHUD still inherits the use of CVODE for solving the ODEs but modernizes and extends PIHM's technical and scientific capabilities. The SHUD is imcompatible to PIHM.

It is our intention (me and previous PIHM group)to begin a debate on the role of *Community Models* in the hydrologic sciences.

SHUD and PIHM represent our strategy for the synthesis of *multi-state*, *multi-scale* distributed hydrologic models using the integral representation of the underlying physical process equations and state variables.

Our interest is in devising a concise representation of watershed and/or river basin hydrodynamics, which allows interactions among major physical processes operating simultaneously, but with the flexibility to add or eliminate states/processes/constitutive relations depending on the objective of the numerical experiment or purpose of the scientific or operational application.

To satisfy the objectives, the SHUD…

- is a distributed hydrologic model, based on the semi-discrete **Finite Volume Method (FVM)** in which domain discretization is an unstructured triangular irregular network (e.g. Delaunay triangles) generated with constraints (geometric, and parametric). A local prismatic control volume is formed by the vertical projection of the Delaunay triangles forming each layer of the model. Given a set of constraints (e.g. river network support, watershed boundary, altitude zones, ecological regions, hydraulic properties, climate zones, etc), an "optimal" mesh is generated. River volume cells are also prismatic, with trapezoidal or rectangular cross-section, and are generated along or cross edges of Delaunay triangles. The local control volume contains all equations to be solved and is referred to as the model kernel.
- is a physically-based model in which all equations used are describing the physics of the hydrological processes which control the catchment. The physical model is able to predict the water in the ungage water system, to estimate the sediment, pollutants, and vegetation, etc., such that it is practical to be coupled with biochemistry, geomorphology, limnology, and other water-related research. The global ODE system is assembled by combining all local ODE systems throughout the domain and then solved by a state-of-the-art parallel ODE solver known as CVODE developed at the Lawrence Livermore National Laboratory.
- is a fully-coupled hydrologic model, where the state and flux variables in the hydrologic system are solved within the same time step and conserve the mass. The fluxes are infiltration, overland flow, groundwater recharge, lateral groundwater flow, exchange of river and soil/groundwater and river discharge.
- is of an adaptable temporal and spatial resolution. The spatial resolution of the model varies from meters to kilometers based requirement of modeling and computing resources. The internal time step of the iteration step is adjustable; it is able to export the status of the catchment in less 1 second to days. Also, the time interval for exporting results is configured flexibly. The flexible spatial and temporal resolution is rather valuable for community model coupling.
- is an open-source model; anyone can access the source code, use and submit their improvement.
- is a long-term yield and single-event flood model.

1.2 Brief History of PIHM system

- 2005 PIHM v1.0

Dr. Yizhong Qu (**?**) developed and verified the first version of PIHM in 2001-2005 during his Ph.D. in Pennsylvania State Unversity, following the blueprint of Freeze and Harlan (1969). This version of PIHM is the soul of the PIHM model.

- 2009 PIHMgis

Dr. Gopal Bhartt (**?**) developed the PIHMgis with support of C++, Qt GUI library, TRIANGLE library, and QGIS developing kit. The development of PIHMgis makes the learning curve of PIHM moderate and benefits the developing, modeling and coupling.

- 2015 MM-PIHM

Dr. Yuninh Shi led and developed the MM-PIHM (Multi-Module PIHM), which embedded all modules from PIHM family, such as RT-PIHM, LE-PIHM, flux-PIHM, BGC-PIHM, etc. together. The sophisticated design and coupling of the MM-PIHM is the summit of the PIHM as a *Community Model* that combined all water-related modules together.

- 2019 SHUD

Based on the accumulated contribution of PIHM modeling and coupling with related researches, it is necessary to solve the known bugs and limitations, improve the performance of the model with parallel methods, and adopt new updates from SUNDIALS solver and programming strategy.

Several publications that may helps:

- (**?**)
- (**?**)
- (**?**)
- (**?**)
- (**?**)
- (**?**)
- (**?**)
- (**?**)
- (**?**)
- (**?**)
- (**?**)

2 Install SHUD and SHUDtoolbox

2.1 SUNDIALS/CVODE

The SHUD model requires the support of the SUNDIALS or CVODE library. **SUNDIALS** is a SUite of Nonlinear and Differential/ALgebraic equation Solvers, consists of six solvers. **CVODE** is a solver for stiff and nonstiff ordinary differential equation (ODE) systems (initial value problem) given in explicit form $y' = f(t, y)$. The methods used in CVODE are variable-order, variable-step multistep methods. You can install the entire SUNDIALS suite or CVODE only.

Since the SUNDIALS/CVODE keeps updating periodically and significantly, the function names and structure are changed accordingly, we suggest to use the specific version of the solver, rather than the latest solver.

SUNDIALS/CVODE is available in LLNL: https://computation.llnl.gov/projects/sundials/sundials-software

The installation of CVODE v3.x:

1. Go to your Command-Line and enter your workspace and unzip your CVODE source code here.

2. make directories for CVODE, including *builddir*.

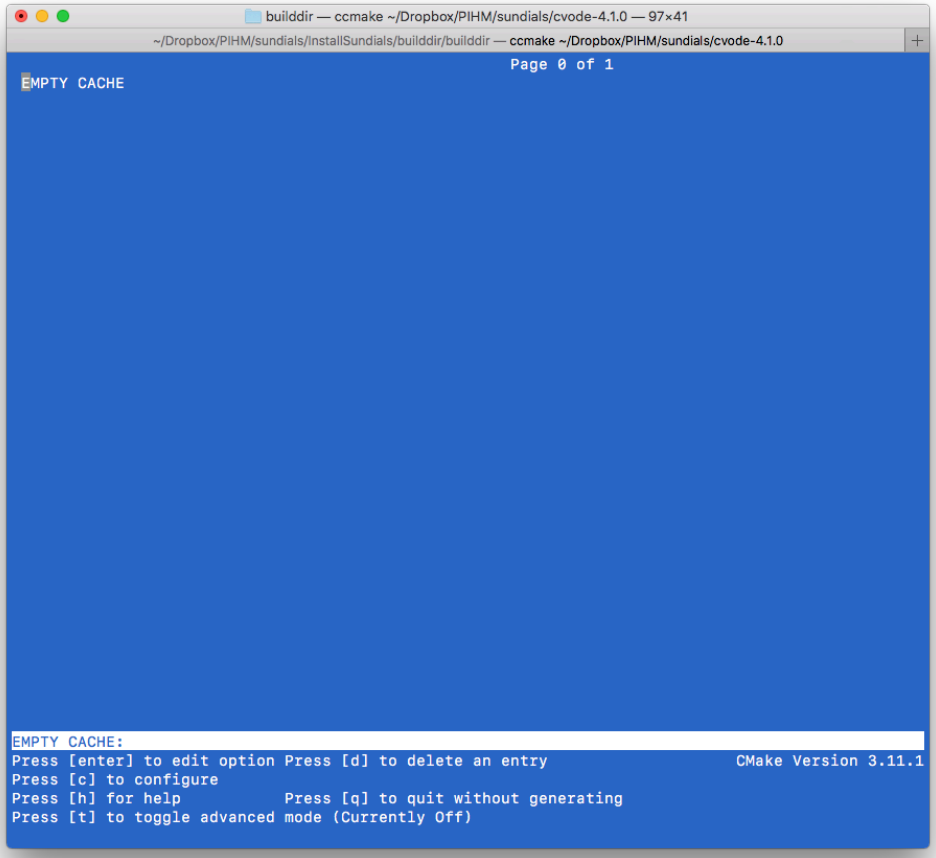
   ```
   mkdir builddir
   cd builddir/
   ```

3. Try ccmake. Install `cmake` if you don't have one.

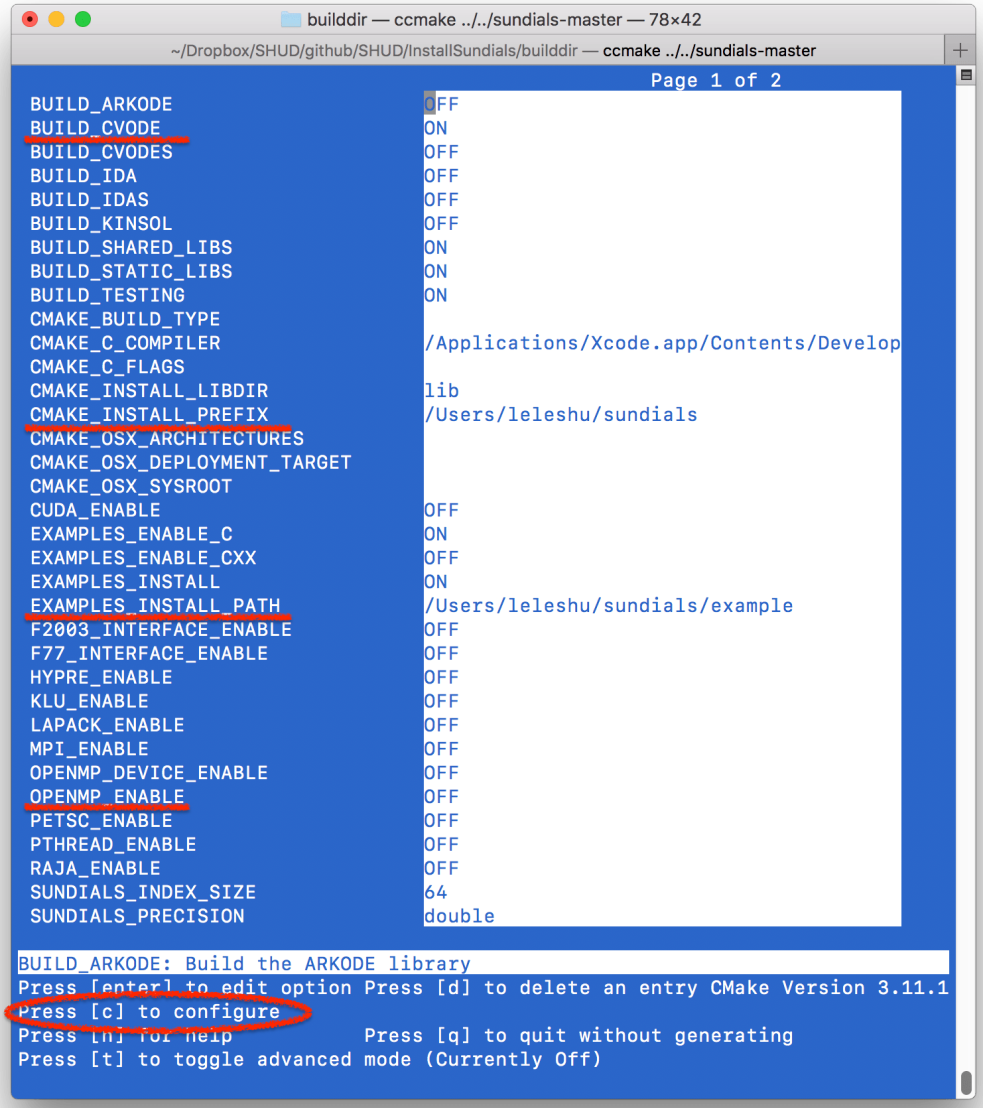
   ```
   ccmake
   ```

4. Run ccmake to configure your compile environment.

   ```
   ccmake ../sundials/cvode-5.0.0
   ```



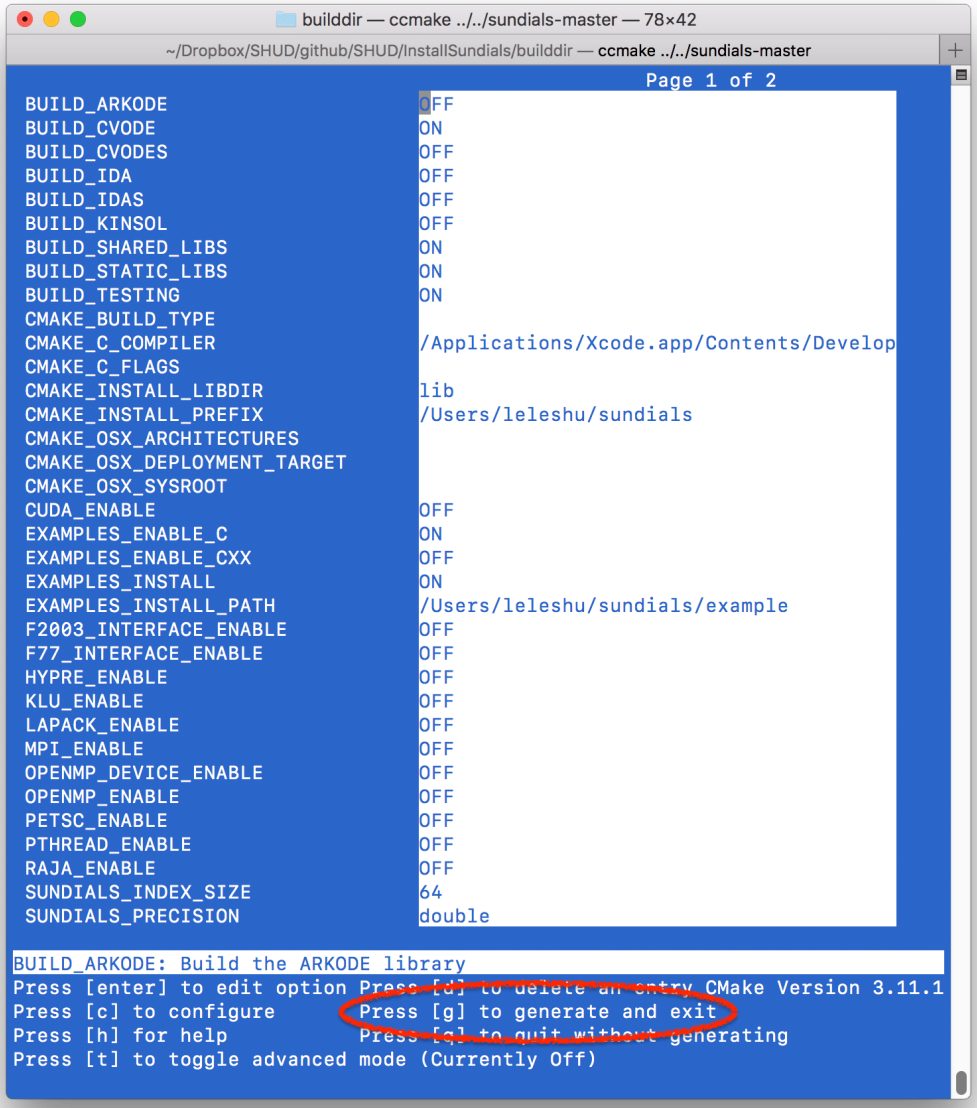
This is an empty configure. Press `c` to start the configuration.



The default configuration. Make sure the value for three lines:

```
BUILD_CVODE = ON
CMAKE_INSTALL_PREFIX = ~/sundials
EXAMPLES_INSTALL_PATH = ~/sundials/examples
```

After the modification of values, press `c` to confirm configuration.



The ccmake configures the environment automatically. When the configuration is ready, press g to generate and exit.

1. Then you run commands below:

```
make
make install
```

2.2 SHUD

Configuration in *Makefile*:

1. Path of *SUNDIALS_DIR*. [**CRITICAL**]. If you install SUNDIALS into *~/sundials*, you don't change this line..
2. Path of OpenMP if the parallel is preferred.
3. Path of SRC_DIR, default is `SRC_DIR = .`
4. Path of BUILT_DIR, default is `BUILT_DIR = .`

After updating the SUNDIALS path in the *Makefile*, user can compile the SHUD with:

```
make clean
make shud
```

There are more options to compile the SHUD code:

- `make all` - clean, then make both shud and shud_omp
- `make help` - help information
- `make shud` - make SHUD executable
- `make shud_omp` - make shud_omp with OpenMP support

2.2.1 OpenMP

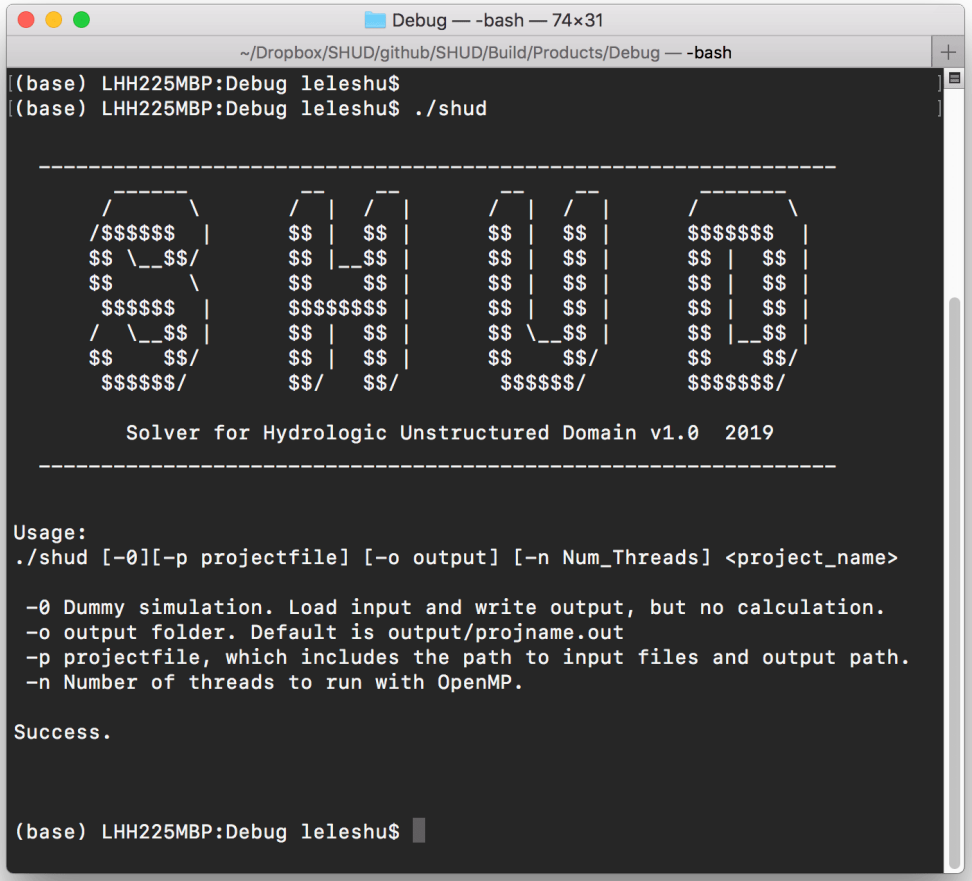
If parallel-computing is prefered, please install OpenMP. For mac:

```
brew install llvm clang
brew install libomp
compile flags for OpenMP:
  -Xpreprocessor -fopenmp -lomp
Library/Include paths:
  -L/usr/local/opt/libomp/lib
  -I/usr/local/opt/libomp/include
```

2.2.2 Run SHUD executables.

After the successful installation and compile, you can run SHUD models using

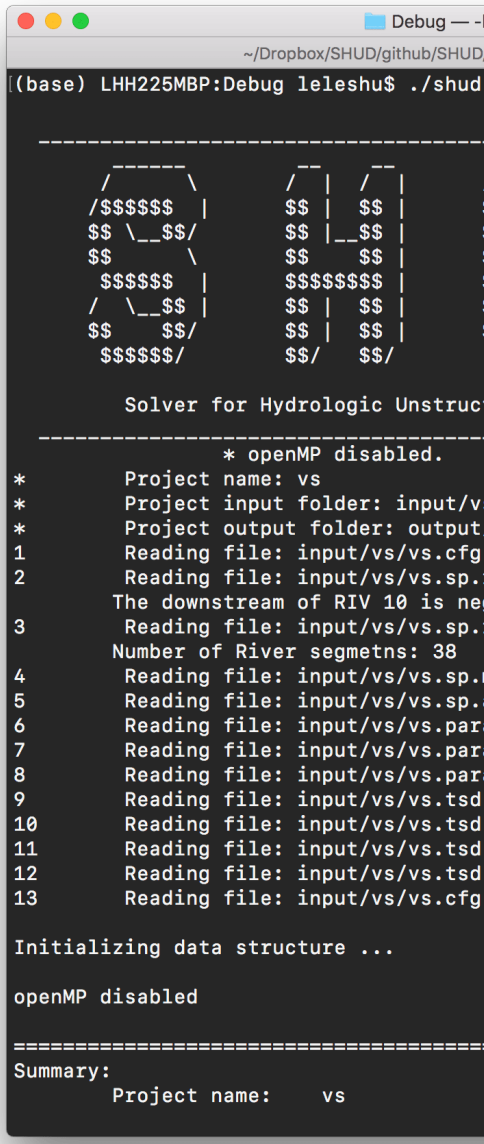
```
./shud <projectname>
```



Command line pattern is:

```
./shud [-0][-p projectfile] [-o output] [-n Num_Threads] <project_name>
```

- `-0` Dummy simulation. Load input and write output, but no calculation.
- `<project name>` is the name of the project.
- `[-p projectfile]` Specify the project file, which includes the path to input files and output path.
- `[-o output_folder]` Output directory. Default is output/projname.out
- `[-n Num_Threads]` Number of threads to run with OpenMP, which works with `shud_omp` only. Usage:



When the `shud` program starts to run, the screen should look like this:

2.3 SHUDtoolbox

This SHUDtoolbox is an R package. What you need is to install the package as a source code package. For example:

```
install_github('SHUD-System/SHUDtoolbox')
```

The prerequisite packages for SHUDtoolbox are:

- Rcpp
- reshape2
- ggplot2
- gridExtra
- grid
- fields
- xts
- hydroGOF
- zoo
- raster (>= 2.1.0)
- sp
- rgeos
- RTriangle
- rgdal (>= 1.1.0)
- proj4
- abind
- utils
- lubridate
- geometry
- methods
- ncdf4
- GGally
- doParallel

One of the required packages, RTriangle, must be installed via GitHub instead of CRAN, using command:

```
install_github('shulele/RTriangle/pkg')
```

3 Input files

List of input files:

File	Category	Comments	Header	# of column	
.mesh	sp	Domain cell (triangular mesh)	Yes		
.att	sp	Attribute table of triangular cells	Yes		
.riv	sp	Rivers	Yes		
.rivseg	sp	Topologic relation b/w River and cell	Yes		
.calib	cfg	Calibration on physical parameters	Yes		
.para	cfg	Parameters of the model configurature	Yes		
.ic	cfg	Intial conditions	Yes		
.geol	para	Physical parameters for Geology layers	Yes		
.soil	para	Physical parameters for Soil layers	Yes		
.lc	para	Physical parameters for Land cover layers	Yes		
.forc	tsd	List of files to the Time-series forcing data	Yes		
.csv	tsd	Time-series forcing data	Yes		
.lai	tsd	Time-series LAI data	Yes		
.obs	tsd	Time-series observational data for calibration purpose only	Yes		
.mf	tsd	Time-series Melt Factor data	Yes		
.rl	tsd	Time-series Roughness Length data	Yes		
gis/domain	Shapefile	Shapefile of .mesh file	x	x	
gis/river	Shapefile	Shapefile of .riv file	x	x	
gis/seg	Shapefile	Shapefile of .rivchn file	x	x	

The files in folder *gis* and *fig* are not involved in SHUD modeling, but they are very useful for your data pre- and post-processing.

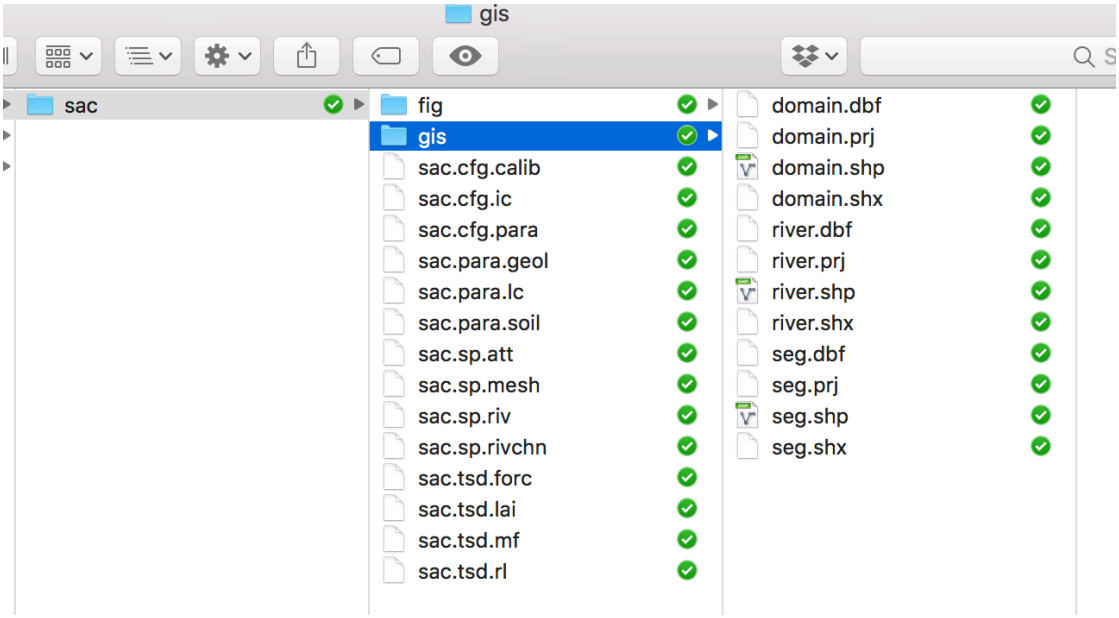
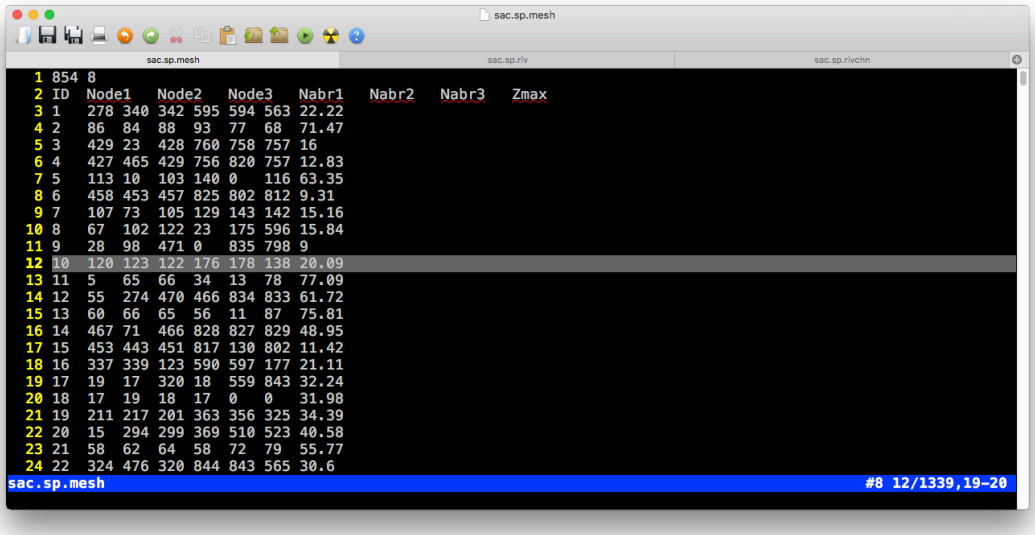
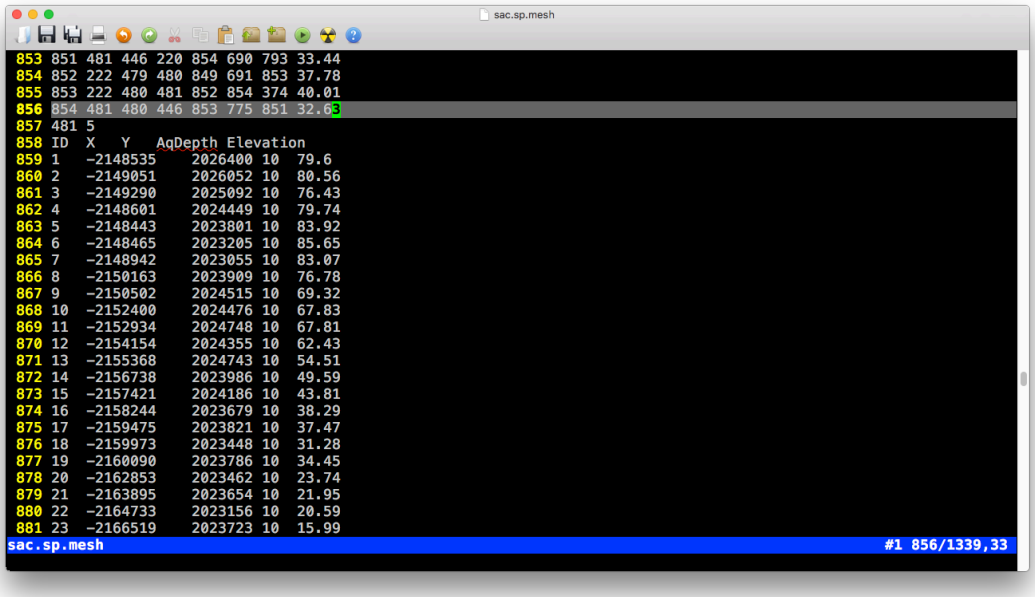


Figure 3.1: The screenshot of input files for SHUD

3.1 Spatial data

3.1.1 .sp.mesh file





There are two tables in the .mesh file, the one is a table of cells and the other is a table of nodes of cells.

- **Block 1 (cell information)**
- Pre-table

Value1	Value2
Number of rows (N_{cell})	Number of columns (8)

- Table

	Colname	Meaning	Range	Unit	Comments
ID		Index of cell i		$1 \sim N_{cell}$	-
Node1		Node 1 of cell i		$1 \sim N_{node}$	-
Node2		Node 2 of cell i		$1 \sim N_{node}$	-
Node3		Node 3 of cell i		$1 \sim N_{node}$	-
Nabr1		Index of Neighbor 1 of cell i		$1 \sim N_{cell}$	-
Nabr2		Index of Neighbor 2 of cell i		$1 \sim N_{cell}$	-
Nabr3		Index of Neighbor 3 of cell i		$1 \sim N_{cell}$	-
Zmax		Surface elevation of cell i		-9999 ~ +inf	m

- **Block 2 (node information)**
- Pre-table:

Value1	Value2
Number of rows (N_{node})	Number of columns (5)

- Table

	Colname	Meaning	Range	Unit	Comments
ID		Index of node i		$1 \sim N_{cell}$	-
X		X coordinate of node i		$1 \sim N_{node}$	-
Y		Y coordinate of node i		$1 \sim N_{node}$	-
AqDepth		Thickness of aquifer i		$0 \sim +\text{inf}$	m
Elevation		Surface elevation of node i		-9999 ~ +inf	m

3.1.2 .sp.att file

- Pre-table

Value1	Value2
Number of rows (N_{cell})	Number of columns (7)

- Table

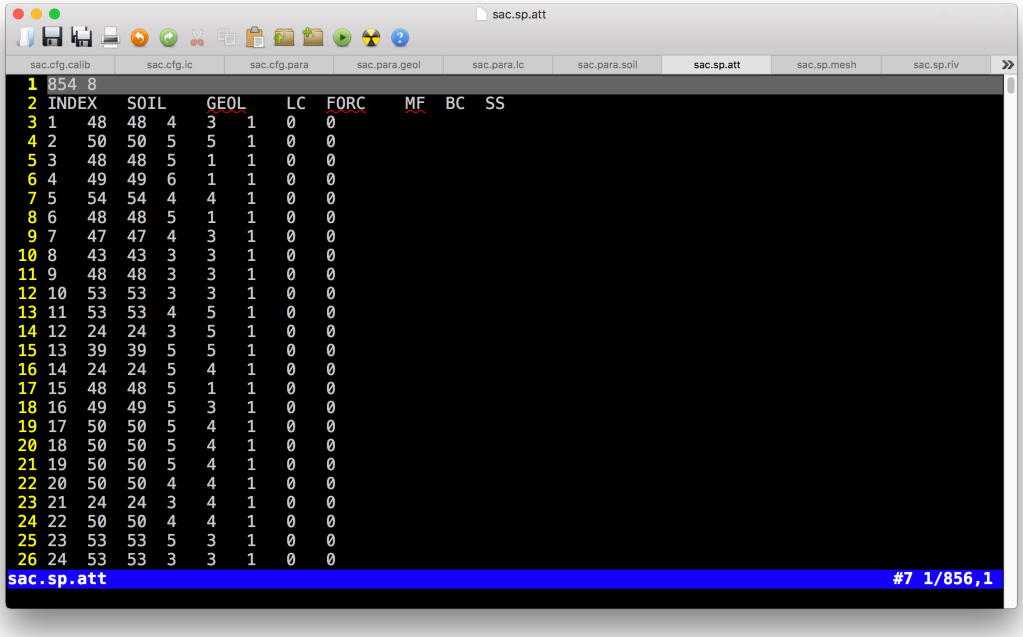


Figure 3.2: Example of .sp.att file

	Colname	Meaning	Range	Unit	Comments
ID		Index of cell i	$1 \sim N_{cell}$	-	
SOIL		Index of soil type	$1 \sim N_{soil}$	-	
GEOL		Index of geology type	$1 \sim N_{geol}$	-	
LC		Index of land cover type	$1 \sim N_{lc}$	-	$N_{lc} = N_{lai}$
FORC		Index of forcing site	$1 \sim N_{forc}$	-	
MF		Index of melt factor	$1 \sim N_{mf}$	-	
BC		Index of boundary condition	$1 \sim N_{bc}$	-	
SS		Index of Source/Sink condition	$1 \sim N_{bc}$	-	

3.1.3 .sp.riv file

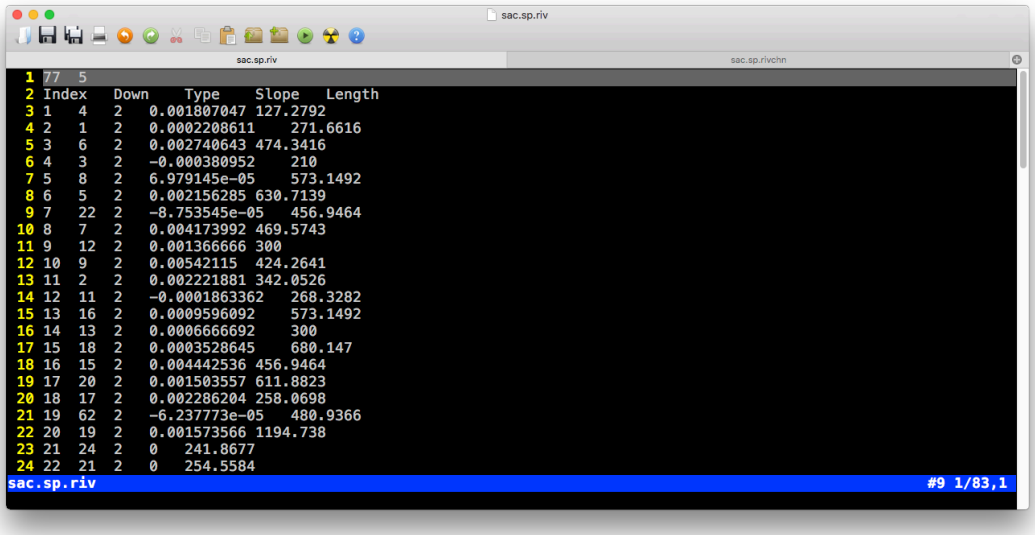


Figure 3.3: Example of .sp.riv file

Colname	Meaning	Range	Unit	Comments
ID	Index of river i	1 ~ N_{river}	-	
DOWN	Index of downstream river	1 ~ N_{river}	-	Negative vlaue indicates outlet
Type	Index of river parameters	1 ~ $N_{rivertype}$	-	
Slope	Slope of river bed	-10 ~ 10	m/m	Height/Length
Length	Length of the river i	0 ~ inf	m	

3.1.4 .sp.rivseg file

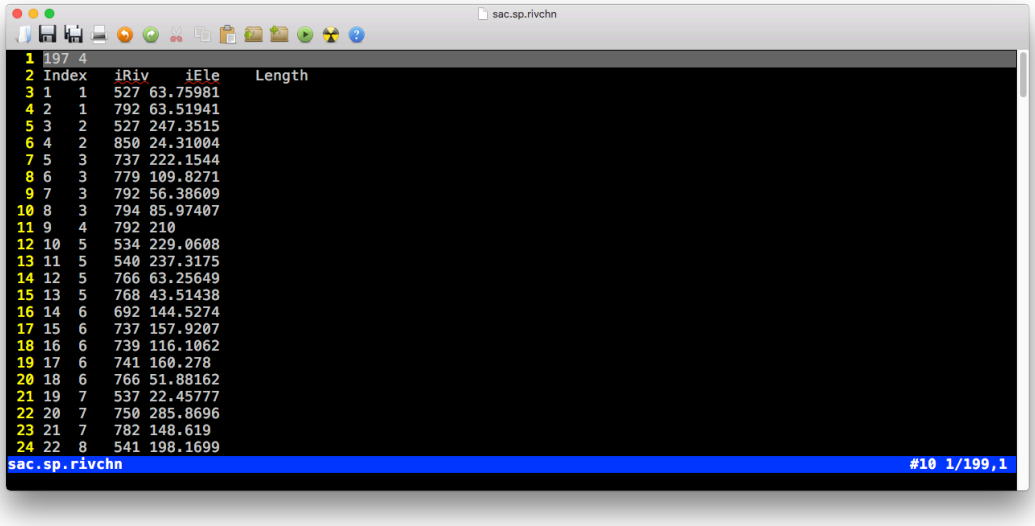


Figure 3.4: Example of .sp.rivseg file

- Pre-table

	Value1	Value2
Number of rows ($N_{segment}$)	Number of columns (4)	

- Table

Colname	Meaning	Range	Unit	Comments
ID	Index of segments i	1 ~ $N_{segment}$	-	

	Colname Meaning	Range	Unit	Comments
iRiv	Index of river	1 ~ N_{river}		-
iEle	Index of cell	1 ~ N_{cell}		-
Length	Length of the segments i	0 ~ inf		m

3.2 Model configuration files

3.2.1 .cfg.para file

- Table

Colname	Meaning	Range	Unit	Default value
VERBOSE	Verbose mode	-	-	0
INIT_MODE	Initial condition mode	0~3	-	3 (0=Relief conditon, 1=Dry condition, 2=Default guess, 3=Warm start)
ASCII_OUTPUT	ASCII ouput	1/0	-	0
Binary_OUTPUT	Binary output	1/0	-	1
SPINUPDAY	Days for spinup	0 ~ inf	*day*	0
SCR_INTV	Number of threads for OpenMP	0 ~ $N_{threads}$	*min*	1440
ABSTOL	Abosolute tolerance for CVODE solver	1e-6 ~ 0.1	-	0.0001
RELTOL	Relative tolerance for CVODE solver	1e-6 ~ 0.1	-	0.0001
INIT_SOLVER_STEP	Initial time step for CVODE solver	-	*min*	1
MAX_SOLVER_STEP	Maximum time step for CVODE solver	1~60	*min*	10
ET_STEP	Time step of Evapotranspiration	1~360	*min*	60
START	Start Time	0 ~ inf	*day*	0
END	End Time	-	*day*	-
dt_ye_snow	Time step of output snow storage	0 ~ inf	*min*	1440
dt_ye_surf	Time step of output surface storage	0 ~ inf	*min*	1440
dt_ye_unsat	Time step of output unsaturated storage	0 ~ inf	*min*	1440

Colname	Meaning	Range	Unit	Default value
dt_Qe_surf	Time step of output surface cell flux	0 ~ inf	*min*	1440
dt_Qe_sub	Time step of output subsurface cell flux	0 ~ inf	*min*	1440
dt_qe_et0	Time step of output cell flux, interception	0 ~ inf	*min*	1440
dt_qe_et1	Time step of output cell flux, transpiration	0 ~ inf	*min*	1440
dt_qe_et2	Time step of output cell flux, evaporation	0 ~ inf	*min*	1440
dt_qe_etp	Time step of output cell flux, potential ET	0 ~ inf	*min*	1440
dt_qe_prcp	Time step of output cell flux, interception	0 ~ inf	*min*	1440
dt_qe_infil	Time step of output cell flux, interception	0 ~ inf	*min*	1440
dt_qe_rech	Time step of output cell flux, interception	0 ~ inf	*min*	1440
dt_yr_stage	Time step of output river stage	0 ~ inf	*min*	1440
dt_Qr_down	Time step of output river flux, downstream	0 ~ inf	*min*	1440
dt_Qr_surf	Time step of output river flux, surface flow	0 ~ inf	*min*	1440
dt_Qr_sub	Time step of output river flux, base flow	0 ~ inf	*min*	1440
dt_Qr_up	Time step of output river flux, upstream	0 ~ inf	*min*	1440

3.2.2 .cfg.calib file

- Table

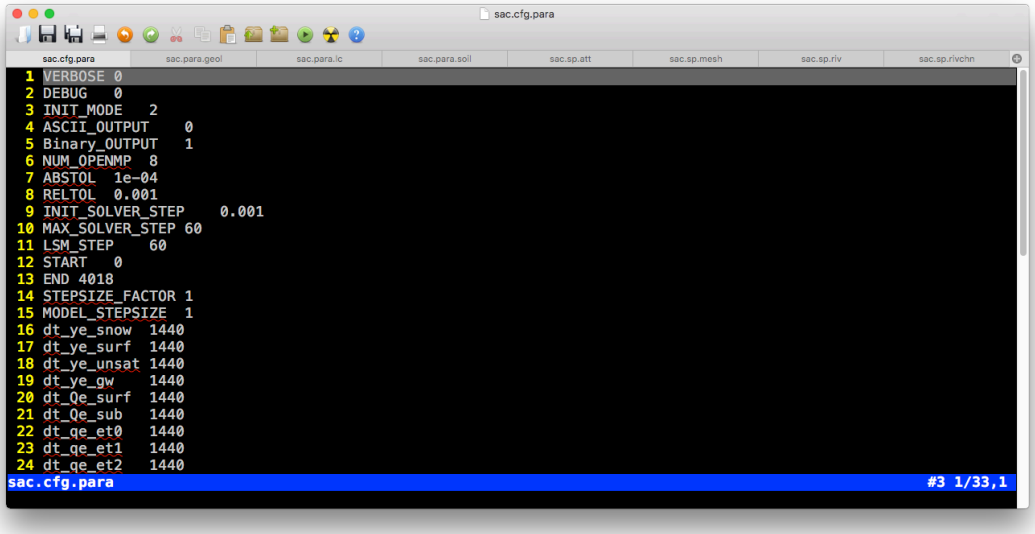


Figure 3.5: Example of .cfg.para file

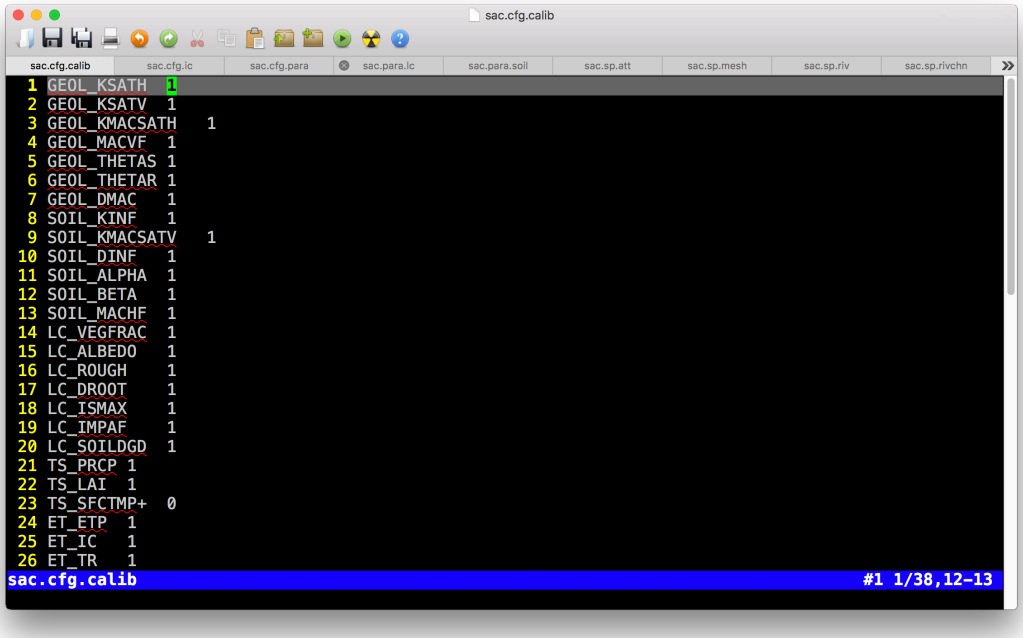


Figure 3.6: Example of .cfg.calib file

Colname	Meaning	Range	Unit	Comments
GEOL_KSATH	Horizontal conductivity of ground water	?		-
GEOL_KSATV	Vertical conductivity of ground water	?		-
GEOL_KMACSATH	Horizontal conductivity of macropore	?		-
GEOL_DMAC	Macropore depth			-
GEOL_THETAS	Porosity, saturated soil moisture			-
GEOL_THETAR	Residual soil moisture			-
GEOL_MACVF	Vertical macropore areal fraction			-
SOIL_KINF	Vertical conductivity of top soil	?		-
SOIL_KMACSATV	Vertical conductivity of soil macropore	?		-
SOIL_DINF	Infiltration depth	?		-
SOIL_DROOT	Root depth			-
SOIL_ALPHA	α value in van Genuchten equation			-
SOIL_BETA	β value in van Genuchten equation			-
SOIL_MACHF	Horizontal macropore areal fraction			-
LC_VEGFRAC	Vegetation fraction			-
LC_ALBEDO	Emissitive reflection ratio			-
LC_ROUGH	Manning's roughness of cell surface			-
LC_SOILDGD	Soil degradation			-
LC_IMPAF	Impervious areal fraction			-
LC_ISMAX	Maximum interception			-
AQ_DEPTH+	Thichness of aquifer			m
TS_PRCP	Precipitation			-
TS_SFCTMP+	Temperature			C
ET_ETP	Transpiration			-
ET_IC	Interception			-
ET_TR	Evaporation			-
ET_SOIL	Evaporation			-
RIV_ROUGH	Manning's roughness of river			-
RIV_KH	Conductivity of river bed			-
RIV_DPTH+	Depth of river cross section			m
RIV_WDTH+	Width of river cross section			m
RIV_SINU	Sinuosity of river path			-
RIV_CWR	C_{wr} in Chezy equation			-
RIV_BSLOPE+	Slope of river bed			m/m
IC_GW+	Initial condition of groundwater			m
IC_RIV+	Initial condition of river stage			m

3.2.3 .cfg.ic file

- **Block 1 (cell initial condition)**
- Pre-table

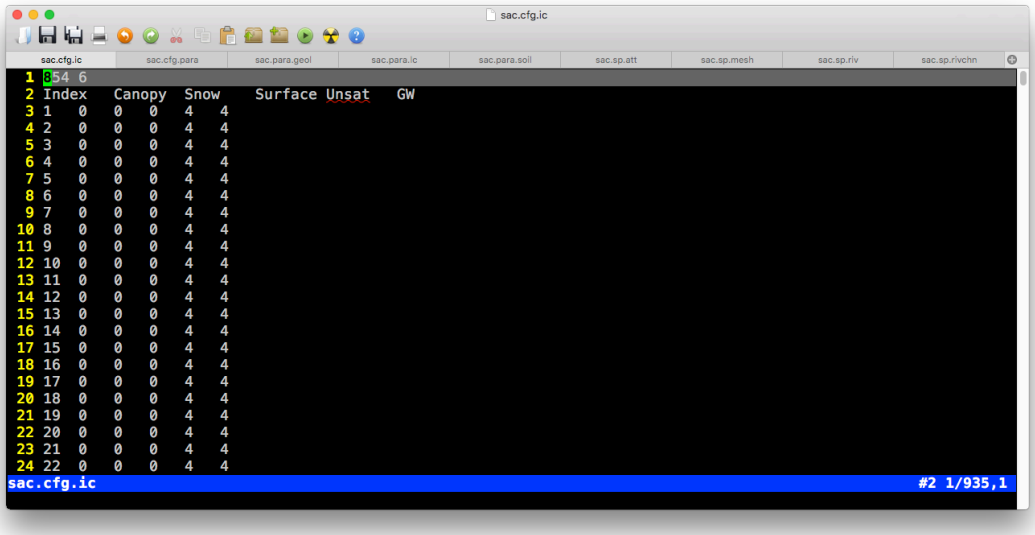


Figure 3.7: Example of .cfg.ic file

	Value1	Value2
Number of rows (N_{cell})	Number of columns (6)	

- Table

	Colname	Meaning	Range	Unit	Comments
ID		Index of cell i	$1 \sim N_{cell}$	-	
Canopy		Canopy storage of cell i	$0 \sim$ inf	m	
Snow		Snow storage of cell i	$0 \sim$ inf	m	
Surface		Surface storage of cell i	$0 \sim$ inf	m	
Unsat		Unsaturated storage of cell i	$0 \sim$ inf	m	
GW		Groundwater head of cell i	$0 \sim$ inf	m	

- **Block 2 (river initial condition)**
- Pre-table:

	Value1	Value2
Number of rows (N_{riv})	Number of columns (2)	

- Table

	Colname	Meaning	Range	Unit	Comments
ID		Index of river i	$1 \sim N_{riv}$	-	
Stage		Stage of river i	$0 \sim$ inf	m	

3.3 Time-series data

3.3.1 .tsd.forc file

- Line 1: `Number of forcing sites | Start day (YYYYMMDD)`
- Line 2: Directory to the spreadsheet

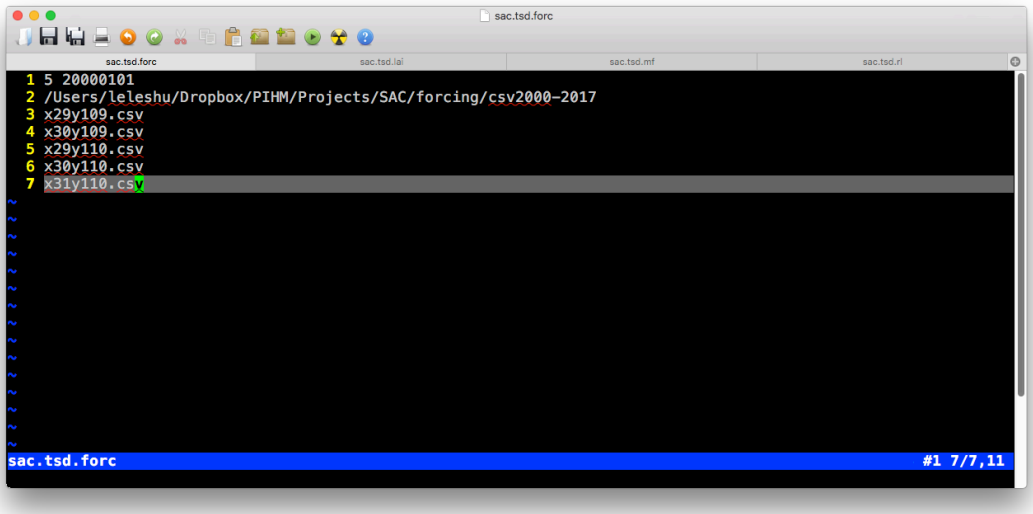


Figure 3.8: Example of .tsd.forc file

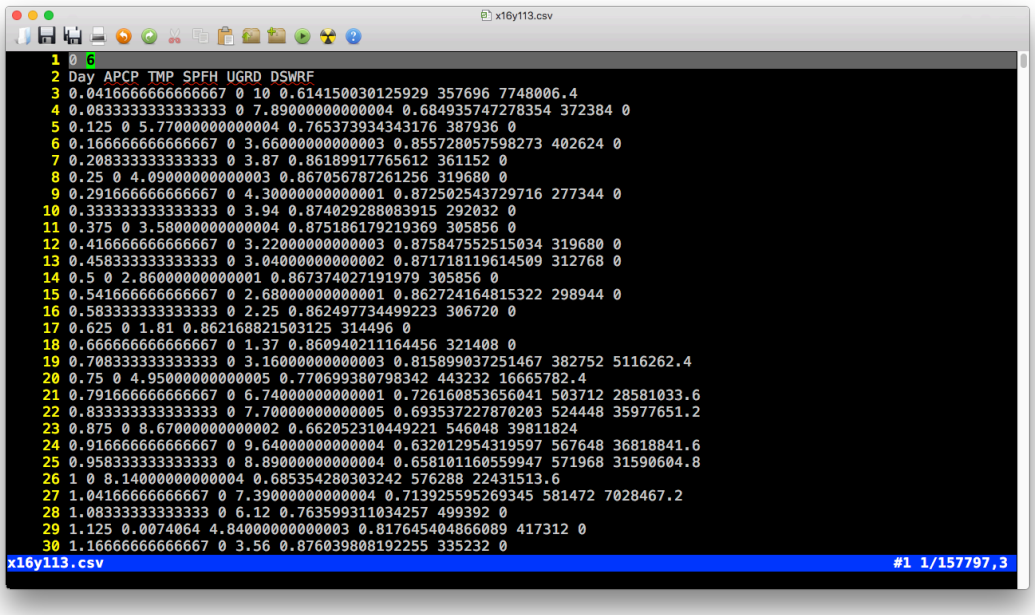


Figure 3.9: Example of .csv forcing file

- Table

	Colname	Meaning	Range	Unit	Comments
Day	Time		0 ~ N_{day}	day	
PRCP	Precipitation		0 ~ 1	m/day	
TEMP	Temperature		-100 ~ 70	C	
RH	Relative Humidity		0 ~ 1	–	
wind	Wind Speed		0 ~ inf	m/day	
Rn	Solar (shortwave) radiation		?	$J/day/m^2$	

3.3.2 .tsd.lai file

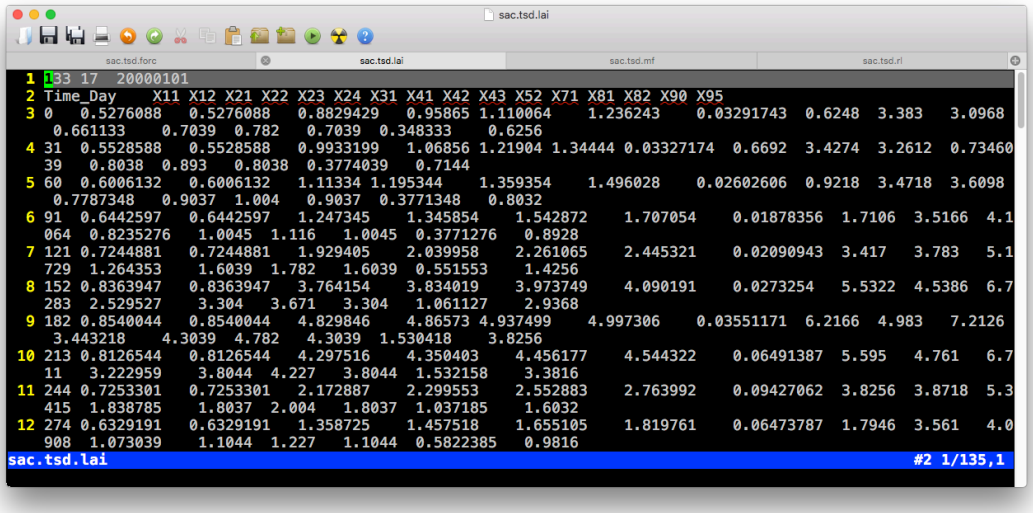


Figure 3.10: Example of .tsd.lai file

- Pre-table:

	Value1	Value2	Value3
Number of day (N_{time})	Number of columns (N_{lc})	Start day (YYYYMMDD)	

- Table

	Colname	Meaning	Range	Unit	Comments
TIME	Time		0 ~ N_{time}	day	
Column 2	LAI of land cover 1		0 ~ inf	m^2/m^2	

	Colname	Meaning	Range	Unit	Comments
Column i		LAI of land cover $i-1$	$0 \sim$ inf	m^2/m^2	
...		...	...	...	

3.3.3 .tsd.rl file

- Pre-table:

	Value1	Value2	Value3
Number of day (N_{time})	Number of columns (N_{lc})		Start day (YYYYMMDD)

- Table

	Colname	Meaning	Range	Unit	Comments
TIME		Time	$0 \sim N_{time}$	day	
Column 2		Roughness length of land cover 1	$0 \sim$ inf	m	
Column i		Roughness length of land cover $i-1$	$0 \sim$ inf	m	
...		...	...	...	

3.3.4 .tsd.mf file

- Pre-table:

	Value1	Value2	Value3
Number of day (N_{time})	Number of columns (N_{mf})		Start day (YYYYMMDD)

- Table

	Colname	Meaning	Range	Unit	Comments
TIME		Time	$0 \sim N_{time}$	day	
Column 2		Melt factor 1	$0 \sim$ inf	-	
Column i		Melt factor $i-1$	$0 \sim$ inf	-	
...		...	...	...	

3.3.5 .tsd.obs file

- Pre-table:

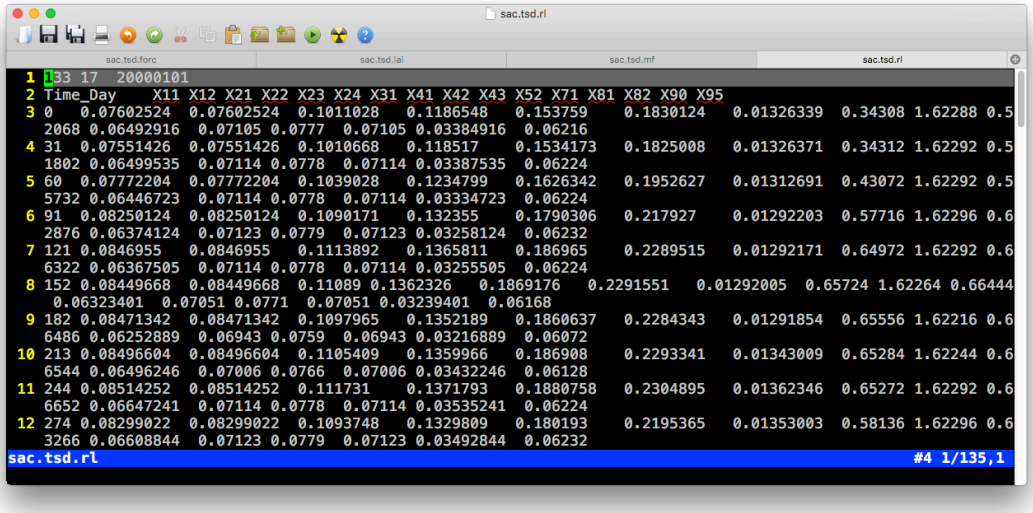


Figure 3.11: Example of .tsd.rl file

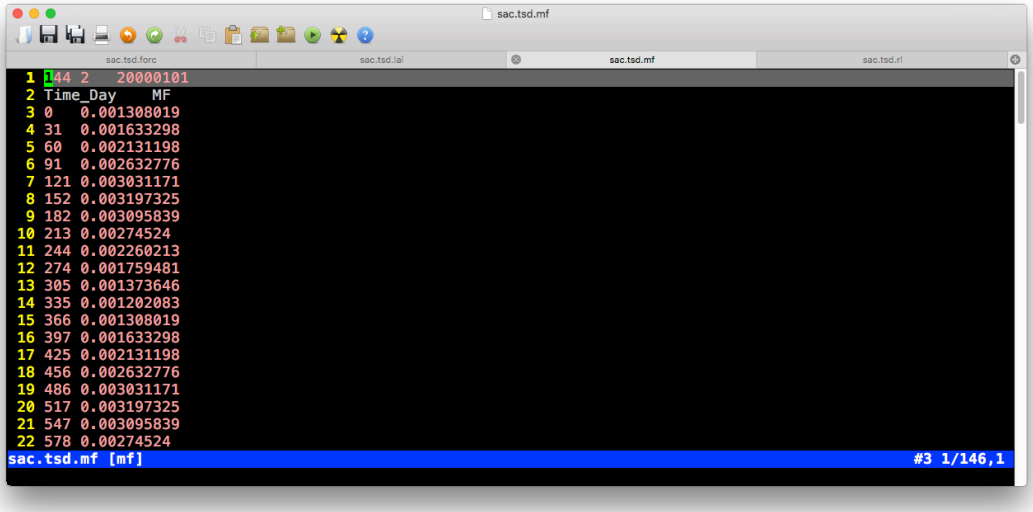


Figure 3.12: Example of .tsd.mf file

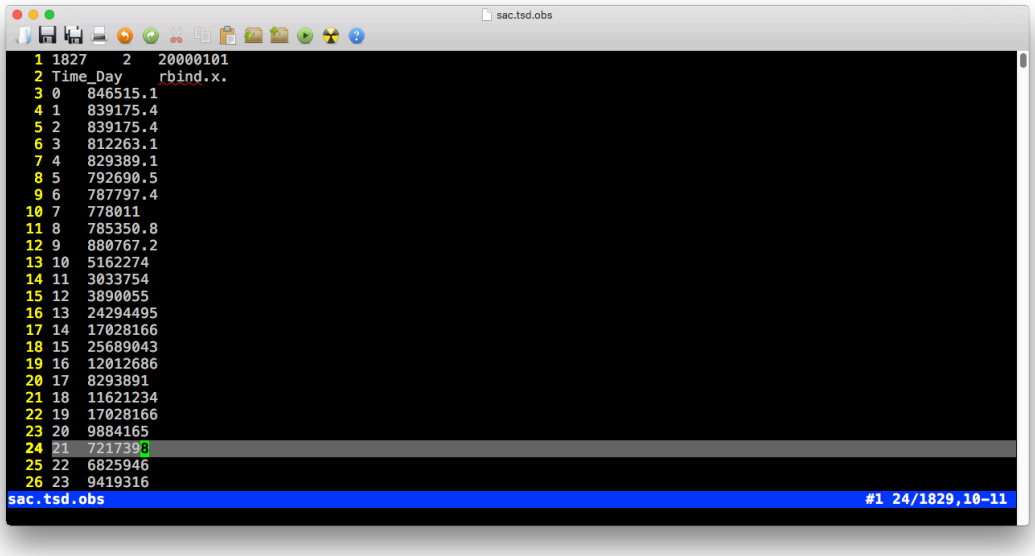


Figure 3.13: Example of .tsd.obs file

	Value1	Value2	Value3
Number of day (N_{time})	Number of columns (N_{obs})	Start day (YYYYMMDD)	

- Table

	Colname	Meaning	Range	Unit	Comments
TIME		Time	$0 \sim N_{time}$	*day*	
Column 2		Observational data 1	?	?	
Column i		Observational data $i-1$	?	?	
...		...	...	...	

4 Output files

4.1 Output file names

Format of output file names:

[Project Name].[Identifier].[Format]

-The *[Project Name]* is user defined name of the project, so every input and output files must start with the *[Project Name]*. -The *[Format]* is one of *csv* or *dat*. *csv* is spreadsheet format and *dat* is bindary format.

The *[Identifier]* is a combination of variables features, that in format of: **[Model Unit][Variable Type][Variable Name]**. *[Model Unit]* is one of three options of *ele* (elemtns), *riv* (river) or *lak* (lake). Variable type includes *y*, *v* and *q* that are state variable (in L), specific flux (in $L^3/L^2/T$) and flux (in L^3/T) respectively.

The list of output files is in following table.

Identifier	Mod unit	Type	Var Name	Meaning	Unit	
.eleyic.	ele	y	ic	Storage of Interception	m	
.eleysnow.	ele	y	snow	Storage of snow equivalence	m	
.eleysurf.	ele	y	surf	Storage of surface	m	
.eleyunsat.	ele	y	unsat	Storage of vados zone	m	
.eleygw.	ele	y	gw	Groundwater head	m	.GW
.elevetp.	ele	v	etp	Potential ET	$\frac{m^3}{m^2 d}$	
.eleveta.	ele	v	eta	Actual ET	$\frac{m^3}{m^2 d}$	
.elevetic.	ele	v	etic	Evap of interception	$\frac{m^3}{m^2 d}$	
.elevettr.	ele	v	ettr	Transpiration	$\frac{m^3}{m^2 d}$	
.elevetev.	ele	v	etev	Soil Evaporation	$\frac{m^3}{m^2 d}$	
.elevprcp.	ele	v	prcp	Precipitation	$\frac{m^3}{m^2 d}$	
.elevnetprcp.	ele	v	netprcp	Net Precipitation	$\frac{m^3}{m^2 d}$	
.elevinfil.	ele	v	infil	Infiltration Rate	$\frac{m^3}{m^2 d}$	
.elevexfil.	ele	v	infil	Exfiltration Rate	$\frac{m^3}{m^2 d}$	
.elevrech.	ele	v	rech	Recharge Rate	$\frac{m^3}{m^2 d}$	
.eleqsurf.	ele	q	surf	Overland flow	m^3/d	
.eleqsub.	ele	q	sub	Subsurface flow	m^3/d	
.rivystage.	riv	y	stage	River Stage	m	

Identifier	Mod unit	Type	Var Name	Meaning	Unit
.rivqup.	riv	q	up	Flux to upstream	m^3/d
.rivqdown.	riv	q	down	Flux to downstream	m^3/d
.rivqsurf.	riv	q	surf	Flux to landsurface	m^3/d
.rivqsub.	riv	q	sub	Flux to subsurface	m^3/d

4.2 Data format in ASCII (.csv) file

N - Number of column of output data, excluding the time column. m - Number of time-step. StartTime - String of date/time (YYYYMMDD or YYYYMMDD.hhmmss)

	N	StartTime		
T_1	$v_{1\cdot1}$	$v_{1\cdot2}$	...	$v_{1\cdot N}$
T_2	$v_{2\cdot1}$	$v_{2\cdot2}$	...	$v_{2\cdot N}$
T_3	$v_{3\cdot1}$	$v_{3\cdot2}$	...	$v_{3\cdot N}$
...	...	...	...	...
T_m	$v_{m\cdot1}$	$v_{m\cdot2}$	...	$v_{m\cdot N}$

4.3 Data format in binary (.dat) file

The value saved in binary file are identical from ASCII format, but different data structure.

ID	i	Value	Format	Length	
1		-	N	double	8
2		-	StartTime	double	8
3		0	T_1	double	8
4		1	$v_{1\cdot1}$	double	8
5		2	$v_{1\cdot2}$	double	8
...		...	...	double	8
$(N+1)*(T-1)+i+3$		N	$v_{1\cdot N}$	double	8
$(N+1)*(T-1)+i+3$		0	T_2	double	8
$(N+1)*(T-1)+i+3$		1	$v_{2\cdot1}$	double	8
$(N+1)*(T-1)+i+3$		2	$v_{2\cdot2}$	double	8
$(N+1)*(T-1)+i+3$		...	...	double	8
$(N+1)*(T-1)+i+3$		N	$v_{2\cdot N}$	double	8
$(N+1)*(T-1)+i+3$		0	T_3	double	8
$(N+1)*(T-1)+i+3$		1	$v_{3\cdot1}$	double	8
$(N+1)*(T-1)+i+3$		2	$v_{3\cdot2}$	double	8

ID	i	Value	Format	Length	
$(N+1)*(T-1)+i+3$	...	...	double	8	
$(N+1)*(T-1)+i+3$	N	$v_{3\cdot N}$	double	8	
$(N+1)*(T-1)+i+3$	...	...	double	8	
$(N+1)*(T-1)+i+3$	...	...	double	8	
$(N+1)*(T-1)+i+3$	...	...	double	8	
$(N+1)*(T-1)+i+3$	...	...	double	8	
$(N+1)*(m-1)+i+3$	0	T_m	double	8	
$(N+1)*(m-1)+i+3$	1	$v_{m\cdot 1}$	double	8	
$(N+1)*(m-1)+i+3$	2	$v_{m\cdot 2}$	double	8	
$(N+1)*(m-1)+i+3$	...	...	double	8	
$(N+1)*(m-1)+i+3$	N	$v_{m\cdot N}$	double	8	