Supplement of Geosci. Model Dev., 15, 4853–4879, 2022
https://doi.org/10.5194/gmd-15-4853-2022-supplement

Geoscientific Model Development
Open Access

Supplement of

Snow Multidata Mapping and Modeling (S3M) 5.1: a distributed cryospheric model with dry and wet snow, data assimilation, glacier mass balance, and debris-driven melt

Francesco Avanzi et al.

Correspondence to: Francesco Avanzi (francesco.avanzi@cimafoundation.org)

S1 User Manual

S1.1 Run preparation

S3M v5.1 requires two categories of input, compulsory data: **dynamic**, weather and **static**, topographic inputs. Optionally, the model also ingests **assimilation** data, in the form of either snow-depth and SCA maps or SWE maps (Updating and SWE maps, respectively, see Section 2 in the main text). The format file required by S3M v5.1 for all inputs is the NetCDF format in the standard GNU zip compression algorithm (gzip, extension .nc.gz).

S1.1.1 Static data

Mandatory static data include:

- a Digital Elevation Model (DEM, in m ASL);
- a raster with metric areas of each computational-grid cell (so-called AreaCell, in m^2);
- a glacier mask.

These static rasters must have the same geographic grid and reference system, which will define the computational grid of the model. The glacier mask will indicate which pixels are covered by glaciers, using a unique integer identifier that is passed through the parameter list (so-called Namelist or Infofile, see below for details). S3M v5.1 will use this identifier to select pixels for which glacier melt must be computed *if module G1 is activated* (see Section 2.4 in the main text). If modules G2 or G3 are selected, S3M v5.1 will use glacier thickness instead (see next paragraph).

Optionally, the user can also provide:

- a glacier-ID raster assigning specific pixels to a glacier according to an inventory;
- a glacier-thickness raster (in m);
- a debris-coefficient raster (see Equation 40 in the main text);
- a Δh pivot table (see Section 2.4 in the main text).

The glacier-ID raster expects each glacier to be indicated using an integer, and S3M v5.1 will initially compile a list of these unique integers as a glacier inventory for the simulation. If the Δh parametrization is chosen, then S3M expects one record of the pivot table for each of these integers. If any of these maps is not supplied, S3M v5.1 will initialize it on the fly using -9999 values, which is S3M's identifier for Not-a-Number values; note that this may lead to inconsistent or erroneous results. S3M v5.1 expects all static input rasters to be included in one single NetCDF file called Terrain_Data.nc.gz (Figure S1). Some basic Python code to generate Terrain_Data.nc.gz is reported at https://github.com/c-hydro/fp-geo-s3m and https://doi.org/10.5281/zenodo.4639614, including a JSON configuration file.

Terrain_Data.nc.gz	S3M, Static Data	Local File
AreaCell	AreaCell	Geo2D
crs	crs	—
GlacierDebris	Glacier debris	Geo2D
GlacierID	Glacier ID	Geo2D
GlacierMask	GlacierMask	Geo2D
Latitude	latitude coordinate	Geo2D
Longitude	longitude coordinate	Geo2D
PivotTable	PivotTable DeltaH	2D
Terrain	Terrain	Geo2D
Thickness	Thickness	Geo2D

(a) Content of Terrain_Data.nc.gz

MeteoData_20160901000...	MeteoData VdA	Local File
AirTemperature	AirTemperature	Geo2D
crs	crs	—
IncRadiation	IncRadiation	Geo2D
Latitude	latitude coordinate	2D
Longitude	longitude coordinate	2D
Rain	Rain	Geo2D
RelHumidity	RelHumidity	Geo2D
Terrain	geometric height a.s.l.	Geo2D
time	time	—
Wind	Wind	Geo2D

(b) Content of MeteoData_yyyymmddHHMM.nc.gz

Figure S1. Content of the NetCDF files expected by S3M v5.1 to ingest static and dynamic input data (related to topography and weather, respectively). Note the names of each field, which must be strictly followed in order for S3M v5.1 to load the underlying data. Also note that these NetCDF files include grids with latitude and longitude as well as information regarding time and the reference system. Some Python code that can be adapted to generate these files is reported at https://github.com/c-hydro/fp-s3m and https://github.com/c-hydro/fp-geo-s3m. These images were obtained by opening example NetCDF files using Panoply (https://www.giss.nasa.gov/tools/panoply/)

S1.1.2 Dynamic data

Mandatory weather input data include:

- air temperature;
- relative humidity;
- incoming shortwave radiation;
- total precipitation.

Weather data must be supplied as distributed raster files according to a common geographic grid and reference system, similar to static data. This geographic grid may in principle be different from the computational grid used by the model, as S3M v5.1 includes a regridding algorithm that automatically checks for concistency and resamples input data using a nearest-neighbor

Updating_201603031100....	MeteoData VdA	Local File
crs	crs	—
Kernel	Kernel	Geo2D
Latitude	latitude coordinate	2D
Longitude	longitude coordinate	2D
SCA	SnowCoverArea	Geo2D
SnowHeight	SnowHeight	Geo2D
SQA	SnowQualityArea	Geo2D
Terrain	geometric height a.s.l.	Geo2D
time	time	—

(a) Content of Updating_yyyymmddHHMM.nc.gz

SWEass_201602020000.n...	MeteoData VdA	Local File
crs	crs	—
Latitude	latitude coordinate	2D
Longitude	longitude coordinate	2D
SWEAssimilated	SWEAssimilated	Geo2D
Terrain	geometric height a.s.l.	Geo2D
time	time	—

(b) Content of SWEass_yyyymmddHHMM.nc.gz

Figure S2. Content of the NetCDF files expected by S3M v5.1 to ingest assimilation data. Note the names of each field, which must be strictly followed in order for S3M v5.1 to load the underlying data. Also note that these NetCDF files include grids with latitude and longitude as well as information regarding time and the reference system. These images were obtained by opening example NetCDF files using Panoply (https://www.giss.nasa.gov/tools/panoply/)

approach. Note, however, that this nearest-neighbor approach may be unsuitable for specific applications, as for example it does not conserve precipitation mass and may lead to inconsistent results in case the meteorological grid is smaller than the static grid. The timestep of input data must be the same as the one chosen for model computations. If any of these weather-input maps is not supplied, S3M v5.1 will initialize it on the fly using -9999 values, which is S3M's identifier for Not-a-Number values; note that this may lead to inconsistent or erroneous results.

S3M v5.1 expects weather-input rasters for each time step to be included in one NetCDF file called MeteoData_yyyymmddHHMM.nc.gz, where yyyymmddHHMM must be replaced with the time-step year (four digits), month, day, hour, and minute (all two digits). Content of one of these files is showed in Figure S1; note that wind fields are not necessary for S3M v5.1. Some code to prepare input files for S3M v5.1 using Python is available at https://github.com/c-hydro/fp-s3m.

S1.1.3 Assimilation data

Assimilation data are supplied to S3M v5.1 in a similar format as weather data (Figure S2): snow-depth and SCA are bundled in an Updating_yyyymmddHHMM.nc.gz NetCDF file, while SWE data are supplied in a SWEass_yyyymmddHHMM.nc.gz, where yyyymmddHHMM must be replaced with the time-step year (four digits), month, day, hour, and minute (all two digits). If assimilation is activated (see next paragraph), S3M v5.1 will look for each of these files every computational timestep; if any of these files is available for a given timestep, it will be loaded and ingested by the model, otherwise the model will throw a warning message and simply no assimilation for that time step will be performed. Some code that can be adapted to prepare these files for S3M v5.1 using Python is available at https://github.com/c-hydro/fp-s3m.

S1.1.4 The Namelist

Besides input and assimilation data, a key step during run preparation is to set up a list of all model options, including paths, modules, and parameter values. In S3M v5.1, this list is referred to as a Namelist or Infofile and is supplied as an ordinary txt file in a pre-defined format. One example of Namelist is available at https://github.com/c-hydro/s3m-dev, while Table S1 details its entries, along with their format, meaning, and options. Further details on parameters and modules can be found in the main text (Section 2). Note that the name of each parameter in the Namelist may be different from notation used in Section 2, mainly because the Namelist reflects definitions used in the source code over the course of ∼15 years of model development. However, comments in the Namelist and details in Table S1 guarantees correspondence with Section 2.

S1.2 Run execution

S1.2.1 Compiling S3M

S3M v5.1 runs on Linux Debian/Ubuntu 64bit environments, and is expected to run with any other Linux system. On the other hand, no portability to Windows platform is currently possible, while portability to Macos systems is in principle possible but untested. The model requires a number of libraries and packages to be pre-installed on the machine, such as the netcdf4 library to handle NetCDF files and a fortran compiler to build the source code (gnu fortran or Intel fortran). Flood-PROOFS, CIMA Research Foundation's toolkit for hydrologic forecasting, offers a number of shell scripts to set up all required libraries (see https://github.com/c-hydro/fp-envs) and to automatically compile S3M v5.1 (see https://github.com/c-hydro/s3m-dev) on Linux Debian/Ubuntu 64bit environments. The user is strongly recommended to use these pre-existing shell scripts, as they automatically configure all required packages for running the model, perform a number of consistency checks, and allow one to pre-set executable name and properly link the netcdf4 library to model executable. The ReadMe file at https://github.com/c-hydro/s3m-dev explains this set-up phase step by step.

Once all libraries are installed and source code is compiled, S3M v5.1 is launched by storing in a given directory the executable (e.g., S3M_v5_p1.x file) and the Namelist (e.g., S3M_namelist.txt, see Section S1.1 and Table S1). The user must then point to this directory through the command line and write[1]:

```
$ ./S3M_v5_p1.x S3M_namelist.txt
```

Upon model initialization and throughout model execution, S3M v5.1 will return several pieces of information on the terminal, including as a minimum time-step data and warnings. Note that the user can increase the amount of information reported on the terminal by setting the debugging mode through the Namelist (see Table S1).

[1]Exact wording may change depending on the exact names of these files in the directory.

S1.2.2 The restart file

S3M v5.1 supports reading initial conditions from an external file and use those conditions to restart a simulation. This restart file is in NetCDF format and should include – as a minimum – rasters of SWE, SWE_D, SWE_W, snow age A_s, snow albedo, bulk-dry-snow density ρ_D, cumulative daily snowfall and melt, and average air temperature over the previous 1 and 10 days ($\bar{T}_{10d}$). If glacier modules G2 or G3 are activated, S3M will also look for glacier thickness and cumulative annual ice melt (only needed for G3), unless the user has instructed S3M to load glacier thickness from the static-data input file (Table S1).

If any of these rasters is not available, S3M will set them to -9999 (missing values). If the restart option is not activated in the namelist (see Table S1), then S3M will initialize them as appropriate. Because of its nature of forecasting model, the restart file in S3M is simply the relevant output file from a previous simulation; if so, an output file with timestamp 11PM is preferred as it is the most complete output file for that simulation day (see Section S1.3 for output-file format).

S1.3 Run post-processing

Throughout model run, S3M saves NetCDF files with a number of select output variables. The frequency of these output files is chosen by the user through the Namelist (Table S1). Similar to all input files, outputs come in the standard GNU-zip-compression format and are automatically generated with name S3M_yyyymmddHHMM.nc.gz, where yyyymmddHHMM is the time-step year (four digits), month, day, hour, and minute (all two digits).

Figure S3 shows the content of one of these output files, with Table S2 detailing the meaning of each field and how they relate to model variables in Section 2 (main text). Field names in the output files are occasionally different from notation used in Section 2 (main text), because S3M-output files are used by other models within CIMA's Flood-PROOFS toolkit and so naming strikes a balance across disciplinary jargon, model versions, and legacy with other tools. Note that the list in Figure S3 refers to the extended output mode as defined in the Namelist (Table S1); Table S2 specifies which variables are also saved with a basic output mode.

S3M_201609012300.nc.gz	S3M_201609012300.nc.gz	Local File
AgeS	Snow Age	Geo2D
AlbedoS	Snow Albedo	Geo2D
H_S	Bulk Snow Depth	Geo2D
Latitude	Latitude Coordinate	2D
Longitude	Longitude Coordinate	2D
MeltingG	Glacier Melt	Geo2D
MeltingS	Snow Melt	Geo2D
MeltingSDayCum	Daily Cumulative Snow Melt	Geo2D
Outflow	Snowpack Runoff	Geo2D
Precip	Total Precipitation Amount	Geo2D
RainFall	Rainfall Amount	Geo2D
REff	Effective Rainfall	Geo2D
RefreezingS	Snow Refreezing	Geo2D
Rho_D	Dry Snow Density	Geo2D
RhoS	Bulk-Snow Density	Geo2D
RhoS0	Fresh-Snow Density	Geo2D
SnowFall	Snowfall Amount	Geo2D
SnowfallCum	Daily Cumulative Snowfall	Geo2D
SnowMask	Snow Mask	Geo2D
SWE	Snow Water Equivalent	Geo2D
SWE_D	Dry SWE	Geo2D
SWE_W	Wet SWE	Geo2D
T_10Days	Average T 10 Days	Geo2D
T_1Days	Air Temperature Last 1 Day	Geo2D
Theta_W	Bulk Vol. LWC	Geo2D
time	time definition of output datasets	—
times	times definition of output datasets	—

Figure S3. Content of the S3M NetCDF output file. This image was obtained by opening this NetCDF file using Panoply (https://www.giss.nasa.gov/tools/panoply/)

Table S1: Entries of S3M Namelist, their format, meaning, and options.

Entry	Format	Meaning	Options
sDomainName	String	Domain name	-
iFlagDebugSet	Integer	Flag for debugging	0 (no) or 1 (yes)
iFlagDebugLevel	Integer	Debugging verbosity	0 to 3
iFlagTypeData_Forcing_Gridded[a]	Integer	MeteoData format	1 (bin int), 2 (bin dbl), 3 (NetCDF)
iFlagTypeData_Updating_Gridded	Integer	Updating format	1 (bin int), 2 (bin dbl), 3 (NetCDF)
iFlagTypeData_Ass_SWE_Gridded	Integer	Ass. SWE format	1 (bin int), 2 (bin dbl), 3 (NetCDF)
iFlagRestart	Integer	Restarting a run[b]	0 (no) or 1 (yes)
iFlagSnowAssim	Integer	Assimilating Updating maps	0 (no) or 1 (yes)
iFlagSnowAssim_SWE	Integer	Assimilating SWE maps	0 (no) or 1 (yes)
iFlagIceMassBalance	Integer	Glacier module	0 (G1), 1 (G2), 2 (G3)
iFlagThickFromTerrData	Integer	Loading glacier thickness from static data[c]	0 (no) or 1 (yes)
iFlagGlacierDebris	Integer	Glacier-debris correction (Eq. 40)	0 (no) or 1 (yes)
iFlagOutputMode	Integer	Output-file format	0 (basic) or 1 (extended)
iFlagAssOnlyPos	Integer	Assimilate only pos. differences	0 (no) or 1 (yes)
a1dGeoForcing	Real	Comma-sep. MeteoData lower-left angle coordinate	-
a1dResForcing	Real	Comma-sep. MeteoData cell sizes	-
a1iDimsForcing	Integer	Comma-sep. MeteoData dimensions	-
iSimLength	Integer	Simulation length in hours	-
iDtModel	Integer	Model time-step in seconds	-
iDtData_Forcing	Integer	Model time-step in seconds	-
iDtData_Output	Integer	Output time-step in seconds	-
iDtData_Updating	Integer	Updating time-step in seconds	-
iDtData_AssSWE	Integer	Ass. SWE time-step in seconds	-
iScaleFactor_Forcing	Integer	MeteoData binary-data scaling factor[a]	-
iScaleFactor_Update	Integer	Updating binary-data scaling factor	-
iScaleFactor_SWEass	Integer	Ass. SWE binary-data scaling factor	-
sTimeStart	String	Timestamp of simulation start	-
sTimeRestart	String	Timestamp of restart[b]	-
sPathData_Static_Gridded	String	Path to static input data[d]	-
sPathData_Forcing_Gridded	String	Path to weather input data[e]	-
sPathData_Updating_Gridded	String	Path to Updating data[e]	-
sPathData_Output_Gridded	String	Path to output data[e]	-
sPathData_Restart_Gridded	String	Path to restart data[b]	-
sPathData_SWE_Assimilation_Gridded	String	Path to Ass. SWE data[e]	-

Table S1: The table continues in the next page

Table S1: Continued from previous page

Entry	Format	Meaning	Options
a1dArctUp	Real	Parameter m'_r for four elevation bands[f]	-
a1dAltRange	Real	Elevation-band limits for parameter m'_r in m[g]	-
iGlacierValue	Integer	Glacier identifier in glacier mask	-
dRhoSnowFresh	Real	Maximum fresh-snow density in kg m^{-3}	-
dRhoSnowMax	Real	Maximum bulk-snow density in kg m^{-3}	-
dRhoSnowMin	Real	Minimum bulk-snow density in kg m^{-3}	-
dSnowQualityThr	Real	Snow-quality threshold for assimilation	-
dMeltingTRef	Real	Threshold-temperature for melting (T_τ, in °C)	-
dIceMeltingCoeff	Real	Ice-melting coefficient [h]	Legacy param., see note and set to 1.
iSWEassInfluence	Integer	Number of validity days after SWE-map issue date [i]	
dWeightSWEass	Real	Maximum weight W in Equation 34.	
dRefreezingSc	Real	Optional multiplicative factor in Equation 19. [j]	See note and set to 1.
dModFactorRadS	Real	Parameter m'_{rad} [k]	-
sWYstart	String	Water-year starting month (two digits)	-
dDebrisThreshold	Real	Threshold-value in f_{debris} to apply Equation 40[l]	-
sCommandZipFile	String	Command to zip files	-
sCommandUnzipFile	String	Command to unzip files	-
sCommandRemoveFile	String	Command to remove files	-
dRhoW	Real	Liquid-water density in kg m^{-3}	-
sReleaseVersion	String	Model version	-
sAuthorNames	String	Authors	-
sReleaseDate	String	Release date	-

[a] S3M v5.1 accepts assimilation and weather input data in binary format (integer or double) in addition to NetCDF. However, this is only allowed for legacy reasons and is discouraged for new applications, so we do not discuss the binary format in this paper.

[b] See Section S1.2 for details on restarting a run.

[c] If 0 is selected, then S3M will load glacier thickness from a restart file, which is useful when pausing and restarting multi-year simulations.

[d] This must include the full path to the file between single quote marks, without the file name (e.g., '/home/S3M/').

[e] These data can either be stored in one folder, or preferably in a year/month/day directory. In the first case, one should specify here the full path of the folder between single quote marks (e.g., '/home/S3M/data/'). In the second case, one can use automatic directory construction through, e.g., '/home/S3M/data/$yyyy/$mm/$dd/'.

[f] Comma separated.

[g] Three comma-separated values. The four elevation bands are defined as (1) all pixels below first value in a1dAltRange; (2) all pixels between the second and the first values in a1dAltRange; (3) all pixels between the third and second values in a1dAltRange; (4) all pixels above the third value in a1dAltRange.

[h] In previous, unpublished versions of S3M, this parameter was a multiplicative term in Equation 18 to compute a melting parameter for bare ice. This modification was supposed to account for albedo decay on ice, which is now an explicit variable in S3M (see Equation 15). While this parameter is still tunable in the current version of S3M, mainly for legacy reasons, it is recommended to set it to 1 for physical reasons. Accordingly, it was not included in Section 2 (main text).

[i] The Kernel function in Equation 34 is set to 0 after this date and the assimilation-SWE map is discarded.

[j] This parameter is an optional, multiplicative term in Equation 19 to reduce m_r in refreezing conditions, similar to Avanzi et al. (2015) or Schaefli et al. (2014). Owing to reasons discussed in Section 2, it is recommended to set this parameter to 1.

[k] Compared to m'_r, m'_{rad} cannot be inputed for elevation bands. This will be included in future releases.

[l] Equation 40 is only applied for pixels where f_{debris} is larger than this threshold value.

Table S2: Output-file content (see Section S1 for details).

Entry	Model variable	Meaning	Basic mode?	Comments
AgeS	A_s	Snow age	Y	-
AlbedoS	α_S	Snow albedo	Y	-
Cum_WY_MeltingG	b_a	Cum. annual mass balance	Y	Only with modules G2 and G3
H_S	h_S	Bulk-snow depth	N	-
Ice_Thickness	h_G	Ice thickness	Y	Only with modules G2 and G3
Ice_Thickness_Change	Δh_G	Ice-thickness change	Y	Only with modules G2 and G3
MeltingG	M_G	Glacier melt	N	-
MeltingS	M	Snow melt	N	-
MeltingSDayCum	-	Cumulative daily snowmelt	Y	Only saved at 11PM
Outflow	O	Snowpack runoff	N	-
Precip	P	Total precipitation	N	-
RainFall	R_f	Rainfall amount	N	-
REff	$O + M_G$	Equivalent precipitation[a]	Y	-
RefreezingS	R	Refreezing	N	-
Rho_D	ρ_D	Dry bulk-snow density	Y	-
RhoS	ρ_S	Bulk-snow density	N	-
RhoS0	ρ_f	Fresh-snow density	N	-
SnowFall	S_f	Snowfall amount	N	-
SnowfallCum	-	Cumulative daily snowfall	Y	Only saved at 11PM
SnowMask	-	Snowmask[b]	Y	-
SWE_D	SWE_D	Dry Snow Water Equivalent	Y	-
SWE_W	SWE_W	Wet Snow Water Equivalent	Y	-
T_10Days	$\bar{T}_{10d}$	Average 10-day temperature	Y	-
T_1Days	-	Average 1-day temperature	Y	Only saved at 11PM
Theta_W	θ_W	Bulk vol. liquid water content	N	-

[a] Equivalent precipitation is the sum of glacier melt and snowpack runoff (Avanzi et al., 2021).

[b] Pixels with at least 0.1 mm of SWE (Section 2).

References

Avanzi, F., Yamaguchi, S., Hirashima, H., and De Michele, C.: Bulk volumetric liquid water content in a seasonal snowpack: modeling its dynamics in different climatic conditions, Advances in Water Resources, 86, 1 – 13, https://doi.org/10.1016/j.advwatres.2015.09.021, 2015.

Avanzi, F., Ercolani, G., Gabellani, S., Cremonese, E., Pogliotti, P., Filippa, G., Morra di Cella, U., Ratto, S., Stevenin, H., Cauduro, M., and Juglair, S.: Learning about precipitation lapse rates from snow course data improves water balance modeling, Hydrology and Earth System Sciences, 25, 2109–2131, https://doi.org/10.5194/hess-25-2109-2021, https://hess.copernicus.org/articles/25/2109/2021/, 2021.

Schaefli, B., Nicótina, L., Imfeld, C., Da Ronco, P., Bertuzzo, E., and Rinaldo, A.: SEHR-ECHO v1.0: a Spatially Explicit Hydrologic Response model for ecohydrologic applications, Geoscientific Model Development, 7, 2733–2746, https://doi.org/10.5194/gmd-7-2733-2014, https://www.geosci-model-dev.net/7/2733/2014/, 2014.