

## Description of the “snow technology” – SMFE

*E. Churyulin<sup>1,2</sup>, V. Kopeykin<sup>1</sup>, I. Rozinkina<sup>1</sup>, M. Chumakov<sup>3</sup>, E. Kuzmina, G. Rivin<sup>1</sup>*

*1 – Hydrometeorological Centre of Russia*

*2 – Lomonosov Moscow State University*

*3 – Gazprom*

### **Introduction:**

The “snow technology” – SMFE is used for work with snow cover on territory of Russian Federation. There are two the most interesting output characteristics of snow cover: snow density (RHO) and snow water equivalent (SWE). The “snow technology” is based on variety of input parameters and involve different complex of assimilation initial information. The main feature “snow technology” is work with model data (output data of COSMO-Ru model), satellite data and initial data from a meteorological network (SYNOP-code). We are using “snow technology” for two schemes of calculations: calculation snow parameters on real meteorological network and calculation it for COSMO grid points. For the territory of Russian Federation, we are applying “snow technology” for three areas of investigation with different model step of special resolution: COSMO-Ru-13-ENA – model step 13 km; area of investigation – all territory of country; COSMO-Ru-ETR – model step 7 km; area of investigation – Europe part of country; COSMO-Ru-2-CFO – model step 2 km; area of investigation – central part of country. Technology had been applied in semi-operational mode for forecast of spring flood since 2016.

### **Methods:**

The “snow technology” consist of several program modules, which have to be configure before a real run of model. There are three main phase of application this technology (image 1):

- 1) Preprocessing
  - a. Preparing a constant initial data for work with “snow technology”;
  - b. Downloading and transforming an initial “satellite” data for reconstruction them to necessary format;
  - c. Downloading an initial meteorological information from database or COSMO-Ru model.
- 2) The model of snow:
  - a. The main part of the “snow technology” – module SMFE.
- 3) Postprocessing:
  - a. Interpolating of calculated meanings on COSMO-grid;
  - b. Transforming of calculated meanings and creating grib-file.

We have to make definite manipulations on every step. In technology, these manipulations are scrips or program modules. For run of it, we should use script “*snow\_oper\_1day.sh*”. For correct explanation of script, we have to give an initialization data in special format “*{year} {month} {day}* “. In case, if the initialization data absent, script is using the current data. In case, if we want to calculate model in date range, we should use script “*snow\_oper\_period.sh*”. Moreover, before run we have to correct script parameters. In testset – modules have names with capital letters, – folders have names with capital letters. Testset contain files for ETR territory of Russian Federation.

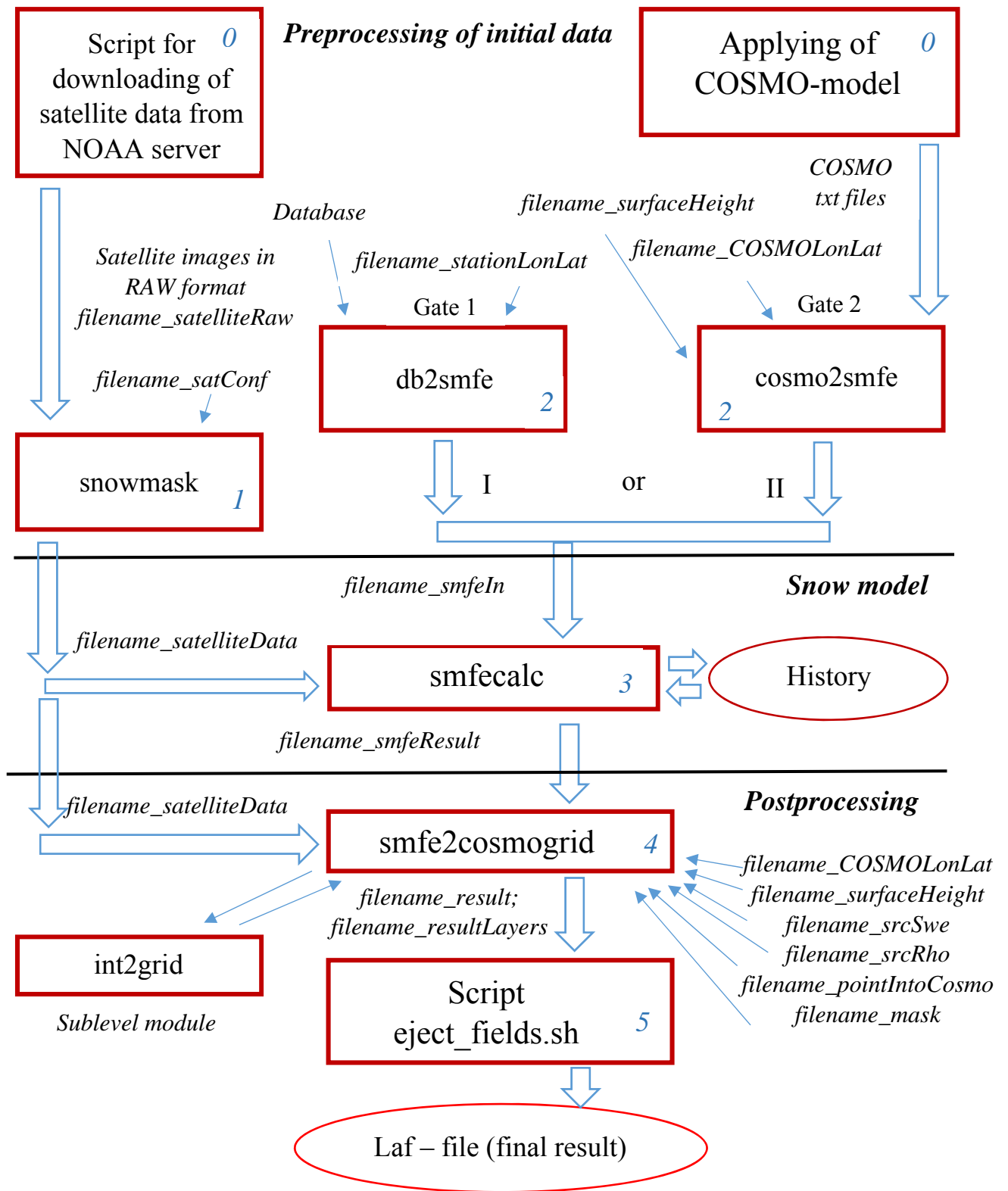


Image. 1 – Blockdiagram of “snow technology”

*Creating files, which are containing necessary information for work with “snow technology”*

Files, which have to be prepare once per interesting time. These files are constant for present configuration:

- 1) filename\_COSMOLonLat – coordinates of COSMO grid points;
- 2) filename\_surfaceHeight – aerography COSMO;
- 3) filename\_stationsLonLat – coordinates of real meteorological stations;
- 4) filename\_pointsIntoCOSMO – points, inside territory of COSMO;
- 5) filename\_mask – mask land/sea.

*Preparing constant initial data for work with “snow technology”*

*The Preparatory stage*

At this stage, special modules for download initial data are using. These data have to be prepare previously and data must be ready for exploitation. There are two different parts of preparing initial data:

**First part:** Downloading satellite information about boundary of snow cover

For downloading satellite information from NOAA server (*ftp://140.90.213.161/autosnow/4kmNH/*) use script “get\_data\_extftp.sh”. At this step, raw satellite data for previously day download and save in bin-file (“filename\_satelliteRaw”). In test set, satellite information about snow cover boundary with 4-km special resolution adapt for model. Script and data is located in folder “SATELLITE\_4KM”.

*Table 1. Information flow for script, which apply for download data from NOAA server*

Input information:	Output information:
file on external server NOAA NH.C17_{date}_BLND_SIM	file: filename_satelliteRaw

For adapting script, we have to correct several parameters, which is written inside script: address of server, where satellite data are stored (variable “EXTFTP”), filename with data (variable “filename\_satelliteData”), and settings of yours proxy server (keys of call utility “curl”).

**Second part:** Receiving “initial” fields from COSMO-Ru

At this stage, data with “initial” fields receive from COSMO-Ru model. After that, received data save in special directory and can be used for further work. In test set, script “COSMO\_data\_preporation.sh” is located in folder “FIELDS\_COSMO7”.

*Table 2. Information flow for script, which apply for download data from COSMO-Ru*

<b>Input information:</b>	<b>Output information:</b>
<b>COSMO past forecasts on grid (lbff-files)</b>	file: filename_snowDepth
<b>Initial fields for COSMO-model</b>	file: filename_p24Sum
<b>(SWE and RHO) – (original laf-file)</b>	file: filename_t2m0
	file: filename_t2m6
	file: filename_t2m12
	file: filename_t2m18
	file: filename_srvSwe
	file: filename_srcRho

For adapting script, we have to correct several parameters, which is written inside script.

*Downloading and transforming an initial “satellite” data for reconstruction them to necessary format. The first stage*

At this stage, we edit satellite data. For this manipulation, we implemented a special program module “snowmask”. “snowmask” module allow us set area of investigation and change the original system coordinates of satellite image to COSMO-Ru module system coordinates. In order to create or change area of investigation necessary prepare a configuration file (“filename\_satConf”). In test set, the configuration file is located in next folder “SATELLITE\_4KM/GRID\_ETR”.

*Example of configuration file:*

```
&COSMOGrid  
NorthLat = 35.0, NorthLong = 215.0,  
COSMON = 700, COSMOM = 620,  
LeftLong = -19.0, LowLat = -19.0,  
LongStep = 0.0625, LatStep = 0.0625
```

where: first line	–	coordinates of the North pole;
second line	–	number of nodes for the COSMO area;
third line	–	coordinates of the low left corner;
fourth line	–	grid step.

**Note 1:** NorthLong should not be written as a negative value.

**Note 2:** Satellite data NOAA have limit area. At the first phase, we have to check it and use only reliable data

**Note 3:** We can use any satellite data with information about snow cover boundary

*Table 3. Information flow for module SnowMask*

Input information:	Output information:
file: filename_satConf	file: filename_satelliteData
file: filename_satelliteRaw	

We can tune module according to parameters of namelist “snowmask.nl”, which is generated by script «snow\_oper\_1day.sh». You can to get acquainted with list of settings and their numerical values for test set in application №1.

*Downloading an initial meteorological information from database or COSMO-Ru model.  
The Second stage*

Before applying “snow technology”, we have to choose resource of initial data. For any resource of information, we have to create module gate, extracting data from resource and retaining data in output file of model, in a strictly defined format.

In test example, we implemented two options of module-gate:

***The first module-gate:*** Receiving real data from meteorological stations

At this case, we are apply module “db2dmfe”. Module allow us download data from internal the database of Hydrometcenter of Russia, about snow cover state, T2m (four runs), precipitation. After that, real data are available for further work of “snow technology”.

*Table 4. Information flow for module “db2smfe”*

Input information: <i>Meteo information from Data Base</i> <i>file: filename_stationLonLat</i>	Output information: <i>file: filename_smfeIn</i>
--	---

***Note 1:*** If we are applying module “db2smfe”, our “snow technology” SMFE is working only for meteorological stations

We can tune module according to parameters of namelist “db2smfe.nl”, which is generated by script “snow\_oper\_1day.sh”. You can to get acquainted with list of settings and their numerical values for test set in application №1.

***The second module-gate:*** Receiving data from COSMO-Ru model in grid points

At this case, we are apply module “cosmo2smfe”. Module allow us download data from the second part of preparatory stage. After that, model data are available for further work of snow technology.

*Table 5. Information flow for module “cosmo2smfe”*

Input information: <i>COSMO initial information</i> <i>file: filename_COSMOLonLat</i> <i>file: filename_surfaceHeight</i>	Output information: <i>file: filename_smfeIn</i>
--	---

***Note 1:*** If we are applying module “cosmo2smfe”, our “snow technology” SMFE is working only for COSMO-grid points

We can tune module according to parameters of namelist “cosmo2smfe.nl”, which is generated by script “snow\_oper\_1day.sh. You can to get acquainted with list of settings and their numerical values for test set in application №1.

*The main part of the “snow technology” – module SMFE  
The third stage*

At this stage, the program module SMFE calculates quantitative characteristics of snow cover in model of snow. For obtaining parameters of snow cover are applied series of formulas and mathematical dependencies. Internal structure of program module SMFE contain seven software branch. These software are different, but complement each other.

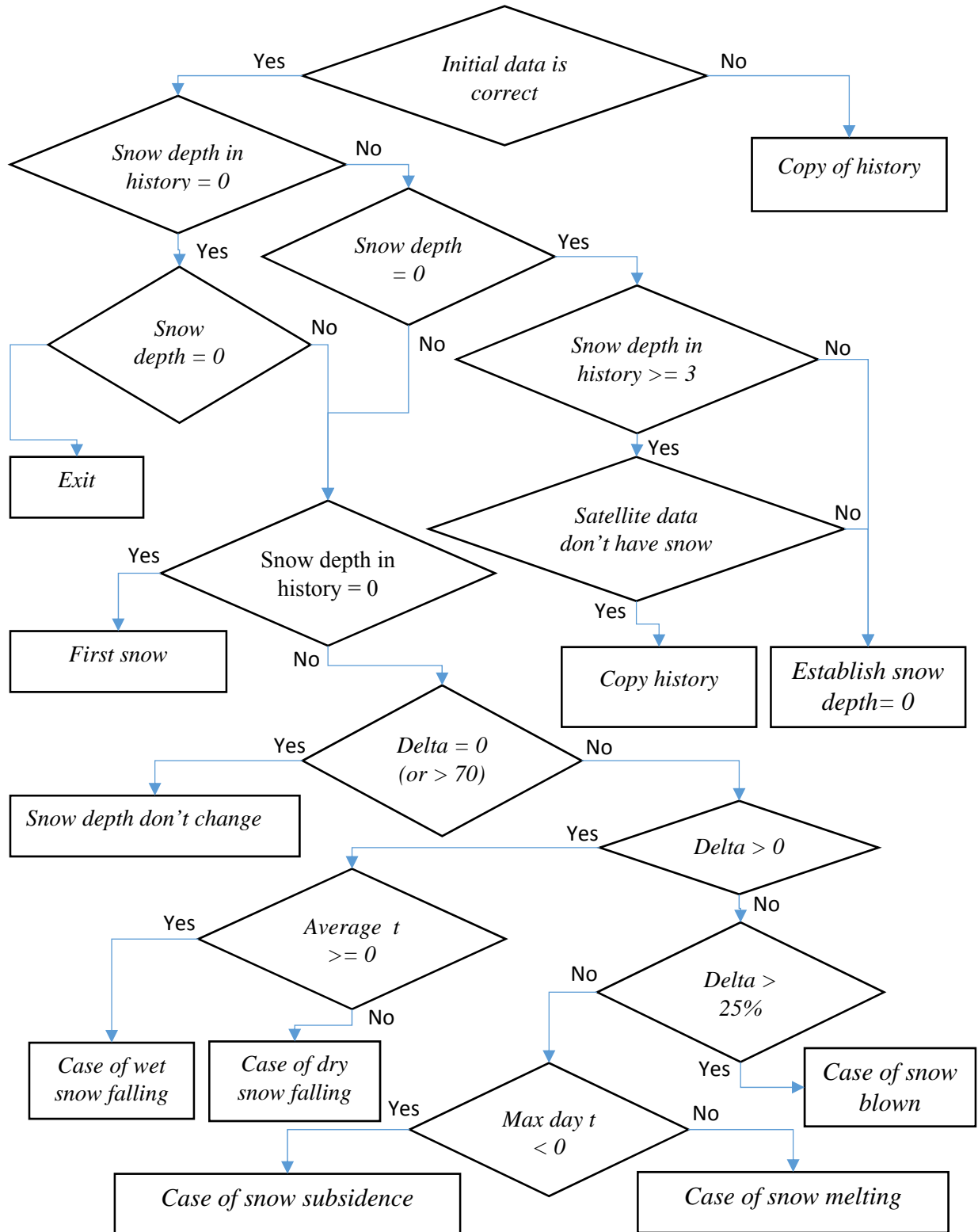


Image. 2 – Blockdiagram of logical principle of SMFE

**First software branch:** Case of first snow

We apply this software branch, if a set of conditions are satisfied: firstly – depth of snow for previous day is equal zero (snow cover was absent), secondly – current depth of snow isn't equal zero. At this step, we use Canadian and Japanese formulas for first snow.

Software branch from 2 to 7 works only in case, if snow cover was for previous day.

**Second software branch:** Case of snow depth don't change

The application of the second calculated branch occurs if the difference between the snow height for the previous day and the snow height at the current time is zero or the difference (delta) between two values of the snow height is more than seventy centimeters, which allows to exclude gross errors in the initial data.

**Third software branch:** Case of wet snow falling

The use of the third calculation branch occurs if the difference between the snow height for the previous day and the snow height at the current time is greater than zero (snow accumulation takes place), and the mean daily air temperature is more than zero degrees, indicating that the snow cover contains a large amount of moisture.

**Fourth software branch:** Case of dry snow falling

Use of the fourth calculated branch occurs if the difference between snow depth for the previous day and snow depth at the current time is greater than zero (there is accumulation of snow cover). But the average daily air temperature is less than zero degrees, which indicates that the snow cover is dry and there is not a lot of moisture in it.

**Fifth software branch:** Case of snow blown

The fifth branch occurs if the difference between the snow depth for the previous day and snow depth at the current time is less than zero (the snow cover decreases), while the snow cover has changed by more than 25%.

**Sixth software branch:** Case of snow subsidence

The sixth branch occurs if the difference between the snow depth for the previous day and the snow height at the current time is less than zero (the snow cover decreases), while the snow cover has changed less than 25%. An important condition for starting this branch is the value of the maximum daily air temperature, in this case this value should be less than zero degrees.

**Seventh software branch:** Case of snow melting

The seventh branch occurs if the difference between the snow height for the previous day and the snow height at the current time is less than zero (the snow cover decreases), while the snow cover has changed less than 25%. An important condition for starting this branch is the value of the maximum daily air temperature, in this case this value should be greater than zero degrees.

*Table 6. Information flow for module "smfecalc"*

Input information:	Output information:
<i>file: filename_satelliteData</i>	<i>file: filename_smfeResult</i>
<i>file: filename_smfeIn</i>	
<i>file: filename_history</i>	

The file database (file: “filename\_history”) is intended for storage and application of data about state of snow cover characteristics SWE and RHO for previous day for every layers. An important addition is fact, that for correct work of model we should have an actual version of history. In case, if model have to recount, previous history must be deleted, otherwise model works incorrect. Moreover, in case of software failure or making adjustments to the model or in time data, there is a special system of everyday data backup for “history”, this procedure gives opportunity of calculation snow data since any day in any time.

We can tune module according to parameters of namelist “smfecalc.nl”, which is generated by script “snow\_oper\_1day.sh”. You can to get acquainted with list of settings and their numerical values for test set in application №1.



*Interpolating of calculated meanings on COSMO-grid  
The fourth stage*

We implemented the “snow2cosmogrid” module. Module was implemented for interpolation output fields from SMFE model (fields of characteristic snow cover (SWE and SD)) to result grid (COSMO-grid).

*Table 6. Information flow for module “snow2cosmogrid”*

Input information: <i>file: filename_satelliteData</i> <i>file: filename_smfeResult</i> <i>file: filename_COSMOLonLat</i> <i>file: filename_surfaceHeight</i> <i>file: filename_srcRho</i> <i>file: filename_srcSwe</i> <i>file: filename_pointIntoCOSMO</i> <i>file: filename_mask</i>	Output information: <i>file: filename_result</i> <i>file: filename_resultLayers</i>
---	---

There are three options of interpolation:

- 1) Interpolation Method 1 – Interpolating of data from SYNOP to COSMO grid. This method apply for territory with high density of meteorological stations of observation. This approach doesn’t use fields of first guess;
- 2) Interpolation Method 2 – Calculating of relations between SYNOP data and fields of first guess. After that, this method apply interpolation and correct fields of first guess;
- 3) Interpolation Method 3 – applying results of calculations of SMFE at each cell of COSMO-model grid. This method doesn’t have interpolation, but it correct interesting fields in output laf-file.

The main result of “snow2cosmogrid” module is output files “filename\_result“ and “filename\_resultLayers”. In test set output files of module is located in “*RESULT\_ETR*” folder.

Module “snow2cosmogrid” produces interpolation set of fields, for this procedure module cause a special program “int2grid”. “int2grid” module allow us interpolate single fields of interesting parameters. For example: fields of snow cover (SWE, RHO, SD).

*Table 7. Informational flow for module “int2grid”*

Input information: <i>file: input_field</i> <i>file: input_grid.txt</i> <i>file: output_grid.txt</i>	Output information: <i>file: output_field</i>
---	--

*Transforming of calculated meanings and creating grib-file  
The fifth stage*

The special script for transformation “filename\_result“ and “filename\_resultLayers” files from txt format to laf-file was implemented by D.Blinov. After run of script, a primary COSMO laf-file is replaced by recalculated SMFE technology laf-file (result of “snow technology”). New laf-file file keep a primary size and name. It can be useful for applied tasks. In general, only several minutes need for calculating model. Work time of it depends on amount of stations or grid points and available computer resources.

*Table 8 – Information flow for script “eject-fields.sh”*

Input information:	Output information:
<i>file: filename_result</i>	<i>Laf-file for COSMO-model with</i>
<i>file: filename_resultLayers</i>	<i>specified SWE and RHO fields</i>
<i>original laf-file for COSMO-model</i>	

In test set, final (output) files (results of script) are located in folder “LAF\_LBFF7”.

**Results:**

The main result of “snow technology” are adjusted laf\_files. Results of our technology are applying hydrology specialists in semi-operational regime in everyday practice, in case if they have territory without real data or density of meteorological network is rare.

**Discussion:**

We are going to upgrade “snow technology” on account of implementation new mathematics algorithms, algorithms of forecast snow cover, its structure and snow retention of tree crowns, moreover we are going to use an increased resolution step.

**Conclusion:**

In this manual, you can find main calculation steps, which have to be make for run of “snow technology”. There are all input and output files, which have to be edit for correct work, moreover in current version of program, you can find test set with all modules, parameters and data for example of efficiency of model

**References:**

- 1) Kazakova E., (2015). Daily estimations of local values and objective analysis of *Snow* water equivalent based on COSMO-Ru technology. Ph.D dissertation, Moscow - 181pp;
- 2) Kazakova E., Rozinkina I., (2016). Snow water equivalent analysis based on modeling as Synop observation operator – experience of COSMO-Ru technology (presentation). 38-th EWGLAM and 23-th SRNWP Meeting;
- 3) Kazakova E., Rozinkina I., Chumakov M., (2013). Realization of the parametric snow cover model SMFE for snow characteristics calculation according to standard net meteorological observations. COSMO newsletter, No.13: April 2013;
- 4) Kazakova E., Rozinkina I., (2011). Testing of snow parametrization schemes in COSMO-Ru: Analysis and Results. COSMO newsletter, No.13: February 2011.

Application №1 (the first part of “snow\_oper\_1day.sh” script, which is generating namelists for several modules):

```

#=== Settings =====
# 1) region -----
region='ETR'
dkm=7

# 2) Directory -----
DIR./SNOW_OPER_STATIONS

DIR_ROOT=$DIR
DIR_COSMO=$DIR/FIELDS_COSMO7
DIR_SATELLITE=$DIR/SATELLITE_4KM
DIR_INTERPOLATION=$DIR/FIELD_INTERPOLATION/COSMO-Ru7
DIR_SMFE=$DIR/SMFE_ETR
DIR_RESULT=$DIR/RESULT_ETR
cd $DIR

# 3) Parameters -----
srcDataType=1          # Use(1)/no(0) station observations (in case of no
there                  # should be forecasts for the previous day at 24,30,36,
                       # 42 UTC)
interpolationMethod=2 # 1 -interpolation from SYNOP to COSMO grid
                       # (no first guess);
                       # only for small territories with dense SYNOP network
                       # 2-calculation of relations between SYNOP
                       # and first guess,
                       # their interpolation and first guess changes
                       # 3-use calculations of SMFE at each cell of COSMO-
model                  # grid (no interpolation, only change fields in laf-
file)                  #
sweRhoLayersCount=0   # Aamount of layers for output from snow model

COSMOGridsCount=434000
stationsCount=2163
satelliteDataCount=434000
pointsCount=2163

#=== Generation namelists =====

cat >COSMO2smfe.nl <<MARKER
&COSMO2smfe
    COSMOGridsCount = $COSMOGridsCount,
    filename_COSMOLonLat = "${DIR_INTERPOLATION}/COSMO_lonlat.txt",
    filename_surfaceHeight = "${DIR_INTERPOLATION}/hsurf.txt",
    filename_snowDepth =
"${DIR_COSMO}/hsnow_COSMO_${region}_${year}${month}${day}.txt",
    filename_p24Sum =
"${DIR_COSMO}/prec_COSMO_${region}_${year}${month}${day}.txt",
    filename_t2m0 =
"${DIR_COSMO}/t2m_0_COSMO_${region}_${year}${month}${day}.txt",
    filename_t2m6 =
"${DIR_COSMO}/t2m_6_COSMO_${region}_${year}${month}${day}.txt",
    filename_t2m12 =
"${DIR_COSMO}/t2m_12_COSMO_${region}_${year}${month}${day}.txt",
    filename_t2m18 =
"${DIR_COSMO}/t2m_18_COSMO_${region}_${year}${month}${day}.txt",
    filename_srcRho =
"${DIR_COSMO}/rho_COSMO_${region}_${year}${month}${day}.txt",
    filename_srcSwe =
"${DIR_COSMO}/swe_COSMO_${region}_${year}${month}${day}.txt",
    filename_smfeIn = "${DIR_SMFE}/smfeIn_${year}_${month}_${day}.txt"
/
MARKER

cat >snowmask.nl <<MARKER
&snowmask

```

```

    year = $year,
    month = $month,
    day = $day,
    hour = $hour,
    filename_satelliteRaw =
"${DIR_SATELLITE}/data/NH.C17_${year}${dayInYear}_BLND_SIM",
    filename_satelliteData =
"${DIR_SATELLITE}/data/sat_noaa_${region}_${year}${month}${day}.txt",
    filename_satConf = "${DIR_SATELLITE}/GRID_${region}/COSMOconf.nl",
/
MARKER

cat >db2smfe.nl <<MARKER
&db2smfe
    year = $year,
    month = $month,
    day = $day,
    hour = $hour,
    stationsCount = $stationsCount,
    dBHost = "192.168.97.72",
    dBName = $db,
    dBCode = 260601,
    filename_stationsLonLat = "${DIR_ROOT}/STATION_${region}.txt",
    filename_smfeIn = "${DIR_SMFE}/smfeIn_${year}_${month}_${day}.txt"
/
MARKER

cat >smfecalc.nl <<MARKER
&smfecalc
    pointsCount = $pointsCount,
    satelliteDataCount = $satelliteDataCount,
    sweRhoLayersCount = $sweRhoLayersCount,
    year = $year,
    month = $month,
    day = $day,
    hour = $hour,
    filename_history = "${DIR_SMFE}/history.bin",
    filename_smfeIn = "${DIR_SMFE}/smfeIn_${year}_${month}_${day}.txt"
    filename_smfeResult = "${DIR_SMFE}/snow_${year}_${month}_${day}.txt"
    filename_satelliteData =
"${DIR_SATELLITE}/data/sat_noaa_${region}_${year}${month}${day}.txt"
/
MARKER

cat >smfe2COSMOgrid.nl <<MARKER
&smfe2COSMOgrid
    COSMOGridsCount = $COSMOGridsCount,
    pointsCount = $pointsCount,
    satelliteDataCount = $satelliteDataCount,
    sweRhoLayersCount = $sweRhoLayersCount,
    interpolationMethod = $interpolationMethod,
    int2gridFactor = 30,
    radius = 8000,
    lonBottomPoint=-19.0,
    latBottomPoint=30.0,
    filename_int2gridExe = "${DIR_ROOT}/int2grid/int2grid.exe",
    filename_int2gridIn = "${DIR_INTERPOLATION}/synop_ex.txt",
    filename_int2gridGrid = "${DIR_INTERPOLATION}/grid.txt",
    filename_int2gridOut = "${DIR_INTERPOLATION}/result.txt",
    filename_COSMOLonLat = "${DIR_INTERPOLATION}/COSMO_lonlat.txt",
    filename_surfaceHeight = "${DIR_INTERPOLATION}/hsurf.txt",
    filename_srcRho = "${DIR_COSMO}/rho_COSMO_${region}_${year}${month}${day}.txt",
    filename_srcSwe = "${DIR_COSMO}/swe_COSMO_${region}_${year}${month}${day}.txt",
    filename_satelliteData =
"${DIR_SATELLITE}/data/sat_noaa_${region}_${year}${month}${day}.txt",
    filename_smfeResult = "${DIR_SMFE}/snow_${year}_${month}_${day}.txt",
    filename_result = "${DIR_RESULT}/${year}${month}${day}_snow_modif_COSMO.txt",
    filename_resultLayers =
"${DIR_RESULT}/${year}${month}${day}_snow_modif_layers.txt",
    filename_pointsIntoCOSMO = "${DIR_INTERPOLATION}/STATION_INT0.txt",
    filename_mask = "${DIR_INTERPOLATION}/mask.txt"
/
MARKER

```