ECOCLIMAP-SG : technical documentation

1) Building of the new map

We take the ESA-CCI global land cover map v1.6.1 at 300-m resolution as a basis for our new map. (see https://www.esa-landcover-cci.org/?q=node/158).

We aim at fitting the following classes, selected for ECOCLIMAP-SG :

1. sea and oceans
2. lakes
3. rivers

(existing covers 1, 2, 3 in ECOCLIMAP)

4. bare land *(vegtype 1 in ECOCLIMAP)*
5. bare rock *(vegtype 2 in ECOCLIMAP)*
6. permanent snow *(vegtype 3 in ECOCLIMAP)*
7. boreal broadleaf deciduous *(vegtype 16 in ECOCLIMAP)*
8. temperate broadleaf deciduous *(vegtype 4 in ECOCLIMAP)*
9. tropical broadleaf deciduous *(vegtype 13 in ECOCLIMAP)*
10. temperate broadleaf evergreen *(vegtype 11 in ECOCLIMAP)*
11. tropical broadleaf evergreen *(vegtype 6 in ECOCLIMAP)*
12. boreal needleleaf evergreen *(vegtype 5 in ECOCLIMAP)*
13. temperate needleleaf evergreen *(vegtype 15 in ECOCLIMAP)*
14. boreal needleleaf deciduous *(vegtype 17 in ECOCLIMAP)*
15. shrubs *(vegtype 19 in ECOCLIMAP)*
16. boreal grassland *(vegtype 18 in ECOCLIMAP)*
17. temperate grassland *(vegtype 10 in ECOCLIMAP)*
18. tropical grassland *(vegtype 11 in ECOCLIMAP)*
19. winter C3 crops *(vegtype 7 in ECOCLIMAP)*
20. summer C3 crops *(vegtype 7 in ECOCLIMAP)*
21. C4 crops *(vegtype 8 in ECOCLIMAP)*
22. flooded trees *(vegtype 4 in ECOCLIMAP)*
23. flooded grassland *(vegtype 12 in ECOCLIMAP)*
24. LCZ1: compact high-rise
25. LCZ2: compact midrise
26. LCZ3: compact low-rise
27. LCZ4: open high-rise
28. LCZ5: open midrise

29: LCZ6: open low-rise
30: LCZ7: lightweight low-rise
31: LCZ8: large low-rise
32: LCZ9: sparsely built
33: LCZ10: heavy industry
(international standard classes for urban areas, Stewart & Oke, 2012).

We will now consider separately each ESA-CCI cover and explain how we translate it the the further classification.

1.1.

210 "Water bodies"

Difficulty: to separate sea from lakes from rivers.

Data sources used:

- Hand-made map from ESA-CCI LCC – v1.5.1 (published before Januaray 2016) through a visual comparison with google-maps (unfinished).
- SWBD from USGS : https://en.wikipedia.org/wiki/SRTM_Water_Body_Data (stops 60°N)
- GSHHG (A Global Self-consistent, Hierarchical, High-resolution Geography Database) and WDBII (CIA World DataBank II) from NOAA http://www.soest.hawaii.edu/pwessel/gshhg/

Scripts to apply to do the separation according to the input data:

a) make_new_map.sh (in grass GIS)

- inputs (ordered as they are prefered, from the second one) :
  - water_esacci = ESA-CCI LCC with 1 where there is water, 0 elsewere
  - rivers_lakes_sea_map_end1@francoise = hand made map containing 1 for rivers, 2 for lakes and 3 for sea (part 1)
  - new_residu_sea_melange_step2@francoise = hand made map containing 1 for rivers, 2 for lakes and 3 for sea (part 2)
  - swbd_usgs_tot = map from the USGS rasterized at 300m resolution, 1 for sea, 2 for lakes and 3 for rivers
  - WDBII_river_f_all = 1 if rivers (rasterization of a vector map)
  - GSHHS_f_water = 1 if rivers (rasterization of a vector map)
  - 4 when nothing is found (default value)
- ouput:
  - new_map_tot

b) make_interpol2.F90 (fortran program):

This FORTRAN program interpols to the nearest neighbour with a halo of around 1 to 3km. By default, it sets “lakes”.

- Input: new_map_tot (binary format, type ECOCLIMAP.dir)
- Output: new_map_tot2 (binary format, type ECOCLIMAP.dir)

1.2.

200 "Bare areas"
201 "Consolidated bare areas"
202 "Unconsolidated bare areas"
220 "Permanent snow and ice"

Difficulty: do distinguish the bare land and the bare rock inside the cover 200.

Data sources used:

GLC2000: http://forobs.jrc.ec.europa.eu/products/glc2000/products.php
Tile Afrique: Bare rock = 24
Tile Asia: Bare rock = 20 et 21
Tile China: Bare rock = 18
Tile Asie du Sud Est: plateau calcaire = 14.
Tile South Asia: bare rock = 42
Tile North America: Bare rock = 23

Scripts to apply in GRASS-GIS to do the separation according to the input data:

a) make_bare_rock_glc.sh: merging of GLC tile information at 1/112° resolution

- inputs:
  - glc2000_africa, glc2000_china, glc2000_north_america, glc2000_se_asia, glc2000_south_asia, glc2000_asia

- output:
  - bare_rock_glc

b) make_ecosg_bare_soil_rock.sh

- inputs:
  - ESACCI_LC_V1@francoise = for the location of ESA-CCI bare land / rock covers
  - bare_rock_glc@francoise = upper output
  - new_map_tot2 = result of step 1.1.
- output:
  - ecosg_bare_soil_rock

1.3.

50 "Tree cover, broadleaved, evergreen, closed to open (>15%)"
60 "Tree cover, broadleaved, deciduous, closed to open (>15%)"
61 "Tree cover, broadleaved, deciduous, closed (>40%)"
62 "Tree cover, broadleaved, deciduous, open (15-40%)"
70 "Tree cover, needleleaved, evergreen, closed to open (>15%)"
71 "Tree cover, needleleaved, evergreen, closed (>40%)"
72 "Tree cover, needleleaved, evergreen, open (15-40%)"
80 "Tree cover, needleleaved, deciduous, closed to open (>15%)"
81 "Tree cover, needleleaved, deciduous, closed (>40%)"
82 "Tree cover, needleleaved, deciduous, open (15-40%)"

Difficulty: to do the climatic separation

Data sources used:
ecoclimap_frac_and_params_v7_3.py and bioclim_LPJ.nc (map and programs used by R.Alkama to pass from 12 to 19 vegtypes in ECOCLIMAP, provided by C.Delire).

Scripts to apply in GRASS-GIS to do the separation according to the input data:
make_foret_climat.sh

- inputs:
  - ESACCI_LC_V1@francoise = for the location of the forest covers
  - bioclim_bin0, bioclim_bin5, bioclim_bin6 = extracted from bioclim_LPJ.nc according with ecoclimap_frac_and_params_v7_3.py
- output:
- new_foret_pure = map with translation of ESA-CCI LCC pure forest covers to ECOCLIMAP-SG values changed further 7 (→8, TEBD), 8 (→ 12, BONE), 9 (→ 11, TRBE), 16 (→ 9,TRBD), 17 (→ 10,TEBE), 18 (→13, TENE), 19 (→ 7, BOBD), 20 (→ 14, BOND)

1.4.

90 "Tree cover, mixed leaf type (broadleaved and needleleaved)"

Difficulty: to give pure values to mixed pixels

Resources used: **functions r.mask and r.surf.random of GRASS-GIS**

r.masks allow to isolate the ESA-CCI LCC cover 90
r.surf.random gives randomly values 1 or 2 to the cover 90

Mixed pixels are put into broadleaf deciduous and evergreen needleleaf, according to the location of cover 90, mainly in the Northern hemisphere.

Script to apply in GRASS-GIS to do the separation according to the input data:
make_foret_tot.sh

- inputs:
  - ESACCI_LC_V1@francoise = to locate the cover 90
  - new_foret_pure = upper output
  - foret_mixte = map in output of r.surf.random
  - bioclim_bin0, bioclim_bin5, bioclim_bin6 = as above for the climate distinction
  - ecosg_bare_soil_rock = output of step 1.2
- output:
  - ecosg_foret_tot = with same above numbers (to be changed further)

1.5.

120 Shrubland
121 "Shrubland evergreen"
122 "Shrubland deciduous"
130 Grassland

Shrubland classes (120, 121, 122) are interpreted as new cover "shrubs" (15).
NB: to be seen later if we prefer to distinguish evergreen from deciduous.

Difficulty: to separate the climatic areas for grassland.

Script to apply in GRASS-GIS to do the separation according to the input data: **make_new_shrubs_grassland.sh** (also gives the final numbers to the forest classes)

- inputs:
  - ESACCI_LC_V1@francoise = for the location of cover 130
  - bioclim_bin10, bioclim_bin8 = xtracted from bioclim_LPJ.nc according with ecoclimap_frac_and_params_v7_3.py
  - ecosg_foret_tot = output upper
- output:
  - ecosg_shrubs_grassland

1.6.

30 Mosaic cropland (>50%) / natural vegetation (tree, shrub, herbaceous cover) (<50%)
40 Mosaic natural vegetation (tree, shrub, herbaceous cover) (>50%) / cropland (<50%)
100 "Mosaic tree and shrub (>50%) / herbaceous cover (<50%)"
110 "Mosaic herbaceous cover (>50%) / tree and shrub (<50%)"

We mosaic as follows (arbitrarily):

- cover 30 → 70% crops / 10% trees / 10% shrubs / 10% herbaceous
- cover 40 → 25% crops / 25% trees / 25% shrubs / 25% herbaceous
- cover 100 → 30% trees / 30% shrubs / 40% herbaceous
- cover 110 → 20% trees / 20% shrubs / 60% herbaceous

These mosaics are realized with the help of the GRASS-GIS functions r.mask and r.surf.random, as in step 1.4. Crops will be treated next step.

Difficulty: define a tree type.

Method:

- we realize a map of the 4 forest types from the map ecosg_foret_tot
- we resample this map at 1° resolution
- we make an interpolation to the neareast neighbour so that each world point at 1° resolution gets a forest type

Program used : **make_interpol_foret.F90**

- inputs:
  - ecosg_foret_tot_1d = map obtained applying the GRASS-GIS command

r.resamp.stats input="only_foret_tot@SWBD" output="ecosg_foret_tot_1d method="mode"

- output = ecosg_foret_tot2_1d

Scripts to apply in GRASS-GIS to do the separation according to the input data:

a) make_mosaic_crop.sh:

- inputs:
  - ESACCI_LC_V1@francoise: to locate cover 30 and cover 40
  - foret_mixte = obtained with the function r.surf.random asking for 10 values from 1 to 10 for cover 30
  - foret_mixte2 = obtained with the function r.surf.random asking for 4 values for cover 40
- output:
  - mosaic_crop = map with 1 for crops, 2 for trees, 3 for shrub, 4 for grassland from covers 30 and 40

b) make_natural_veg_mosaic.sh:

- inputs:
  - mosaic_crop = above output
  - ecosg_foret_tot2_1d = output of make_interpol_foret.F90
  - bioclim_bin0, bioclim_bin5, bioclim_bin6, bioclim_bin8 = s above for the climate distinction
- output:
  - natural_veg_mosaic = maps of natural part from covers 30 and 40, randomly shared between trees, shrubs and herbaceous

c) make_mosaic_100_110.h:

- inputs:
  - ESACCI_LC_V1@francoise = to locate cover 100 and cover 110*
  - aleat_100_110 = obtained with the function r.surf.random asking for 5 values from 1 to 5
  - ecosg_foret_tot2_1d = map obtained applying the GRASS-GIS command
  - *r.resamp.stats input="only_foret_tot@SWBD" output="ecosg_foret_tot_1d method="mode"*
  - bioclim_bin0, bioclim_bin5, bioclim_bin6, bioclim_bin8 = as above for the climate distinction
- output:
  - mosaic_100_110 = map with random tree / herbaceous / shrubs

1.7.

10 "Cropland, rainfed "
11 "Herbaceous cover"
12 "Tree or shrub cover"
20 "Cropland, irrigated or post-flooding"

The cover 12 is affected to broadleaf deciduous with a low height (vineyards, fruit trees).
The cover "irrigated crops" was suppressed → the cover 20 is put in the cover 10.
The crops obtained at previous step (mosaics crops / other natural vegetation) are also put inside the cover 10.

Difficulty: distinguish winter C3 / summer C3 / C4 crops.

A. To get the C4 information from ECOCLIMAP2

We use a map of 10 random values to represent the percentages of C4 from ECOCLIMAP2 from 10% to 10%.

Scripts to apply in GRASS-GIS:

make_first_crop.sh

- inputs:
  - ESACCI_LC_V1@francoise = to locate covers 10, 11, 20
  - mosaic_crop = map obtained at step 1.6.a)
  - foret_mixte = obtained with the function r.surf.random asking for 10 values
  - VEGTYPE_P8_ECO2.3 = percentage of C4 from ECOCLIMAP2, rounded from 0 to 10
- output:
  - first_crop = map of crops with the percentage of C4 from ECOCLIMAP2 integrated

B. Use of the FA statistics.

Data sources used:

- map of boundaries of countries ISO3Code_2014 (no longer available on web), vector map converted in a raster at 300m resolution.
- FAO data for 2014: http://www.fao.org/faostat/en/#data/QC/visualize (2014 = last available year). Number of hectares cultivated for each country.

We accept that the sum of the cultivated hectares for each country corresponds to the total surface cultivated by country.

Method:

- we do the connexion between the countries from FAO and from ISO3Code_2014.
- We do the connexion between the FAO crops and the types C3w, C3s and C4.
- We make some maps of percentages of C3w, C3s and C4 by country according to the FAO and according to the map coming from make_first_crop.sh (percentages based on the sum C3+C4, wooden are taken apart).
- We equilibrate the percentages of C4 by country to approach those from the FAO with a random distribution of C4 in the crops of the countryes, close to +-5% (use of a map of 20 random values).
- We make a percentage of C3s with a randow distribution inside the C3 part of each country, according to the percentage given by the FAO, close to +-5%.

Useful files / programs / scripts:

- FAOSTAT_data_1-9-2017.ods = contains the statistics from FAO by country and crop for 2014
- ISO3Code_2014_base.ods = list of countries
- area_code.csv = contains the list of FAO countries.
- make_liste_pays_num.F90:
  - inputs: area_code.csv
  - output: area_code2.csv = list of the FAO countries with associated numbers
- liste_pays_fao_code.csv = connexion between the numbers of countries from the FAO (first column) and from ISO3Code_2014_base.ods (hand made connexion)
- fao_code.csv = FAO number of each crop associated to its type (C3w, C3s, C4, wood).
- fao_simple2.csv = FAO codes for the country and for the crop, and number of hectares cultivated for the countries, withouth the lines where the number of hectares is 0.
- make_pourcentage_cultures.F90:
  - inputs:
    - fao_code.csv
    - fao_simple2.csv
    - liste_pays_fao_code.csv
  - outputs: to recode the country map in percentages of C3w, C3s and C4 with GRASS.
    - pourcentages_C3h_par_pays_b.csv
    - pourcentages_C3e_par_pays_b.csv
    - pourcentages_c4_par_pays_b.csv
- stats_first_crop_iso3 = statistics by country calculated for the first_crop map.

Scripts to apply in GRASS-GIS:

a) make_new_crops_20.sh:

- inputs:
  - ISO3Code_2014_c4_new = percentages of C4 by countries for first_map
  - ISO3Code_2014_c4_fao = percentages of C4 by countries for the FAO
  - first_crop = map obtained upper
  - foret_mixte2= random map of 20 values
- output:
  - new_crops = map with the information of C4 from the FAO integrated

b) make_new_crops2.sh:

- inputs:
  - new_crop = map obtained in step a)
  - ISO3Code_2014_c3e_fao = percentages of C3s by country for the fao
  - ISO3Code_2014_c3h_fao = percentages od C3w by country for the fao
  - monde_random_c3 = random map of 20 values
- output:
  - new_crop2 = map of crops with the information of C3W/S from the FAO integrated

C. Use of the AGRESTE data for France

<u>Data sources</u>:

- AGRESTE statistics = https://stats.agriculture.gouv.fr/disar/faces/report/tableauForm.jsp
- map of French deparments = https://www.data.gouv.fr/fr/datasets/decoupage-administratif-communal-francais-issu-d-openstreetmap/

We download the detail of cultivated hectares by crop type for each French department, in order to obtain the total C3/C4 by department.
To be noted: don’t work in the DOM because bananas are not taken into account.

We do the same thing as for the FAO data, in this case.

<u>Useful files / programs / scripts:</u>

- depts_details2.ods = data dowloaded from AGRESTE, by department
- depts_details3.ods = simplified version of depts_details2.ods
- france_detail_reduite.ods = connexion between each crop and its type (C3w, C3s, C4)
- read_moy_depts.F90:
  - inputs:
    - depts_details3.csv
  - outputs:
    - depts_types_cultures.csv
    - recode_depts_c3h.txt = percentages of C3w by departments, to recode with GRASS
    - recode_depts_c3e.txt = percentages of C3s by departments, to recode with GRASS
    - recode_depths_c4.txt = percentages of C4 by departments, to recode with GRASS

Script to apply in GRASS-GIS:

- **make_new_crops_france.sh**
  - inputs:
    - depts_c4_first_crops = percentages of C4 for first_crops, by French department
    - depts_c4_agreste = percentages of C4 for agreste, by French departement
    - first_crop = map obtained at step 1.7.a
    - foret_mixte2 = random map of 20 values
  - output:
    - new_crop_france = map with the C4 information from AGRESTE integrated

- **make_new_crops2_france.sh**
  - inputs:
    - new_crop_france = map obtained upper
    - depts_c3e_agreste = percentages of C3s for agreste, by French department
    - depts_c3h_agreste= percentages of C3w for agreste, by French department
    - france_random_c3 = random map of 20 values for C3 crops
  - output:
    - new_crop2_france =map with the C3W/S information from AGRESTE integrated

D. Use of the USDA data on the USA

Data sources used:
- detailed map of crops by USA state, for 2015, at 30m resolution: https://nassgeodata.gmu.edu/CropScape/

Method:
- we translate each crop from USDA in types C3w, C3s, C4
- we recode each state map translating the crop to its type, suppressing all that is not croplanp
- we reproject each map in our region
- we agregate each map to 300m resolution
- we gather the states maps in one only map
- we put this map on the ESACCI map of crops on the USA
- ESACCI crops points not covered by USDA are set to C3s.

Useful files / programs / scripts:
- nomenclature_usda.ods = connexion between the USDA crops numbers and the types C3w, C3s, C4 and woody.
- usda_simple.csv = same list but only with the numbers
- make_recode_usda.F90:
  - inputs:
    - usda_simple.csv
  - output:
    - usda_crops_recode.txt: file to recode in grass

Script to apply in GRASS-GIS:
make_new_usa_crop.sh:
- inputs:
  - first_crop = map obtained at step 1.7.a
  - usa_usda = mask to apply only on the USA
  - agreg_all2 = map obtained following the method above
- output:
  - new_usa_crops = map with the crop information from USDA taken into account

E. gathering of the crops maps obtained in previous steps

Script to apply in GRASS-GIS:
make_all_crops.sh:
- inputs:
  - new_usa_crops = map obtained at step D, with 0 where there is no crops
  - new_crop2_france_b = map obtained at step C adapted for the global patch (global, 0 where there is no crops)
  - new_crop2 = map obtained at step B, with 0 where there is no crops
  - first_crop = map obtained at step A
- output:
  - all_crops =all crops maps integrated

1.8.

150 "Sparse vegetation (tree, shrub, herbaceous cover) (<15%)"
152 "Sparse shrub (<15%)"
153 "Sparse herbaceous cover (<15%)"

We mosaic as follows (arbitrarily):

- cover 140 → 70% bare rock / 30% herbaceous
- cover 150 → 90% bare soil / 5% shrubs / 5% herbaceous
- cover 152 → 90% bare soil / 10% shrubs
- cover 153 → 90% bare soil / 10% herbaceous

We implement the same method as for the other cases of mosaics.

Script to apply in GRASS-GIS:
make_sparse_lichen.sh:

- inputs:
    - ESACCI_LC_V1 = to locate covers 140, 150, 152, 153
    - aleat_sparse_lichen = random map of 20 values for lichens and sparse
    - bioclim_bin10, bioclim_bin8 = climate areas already used before
- output:
    - sparse_lichen = map of sparse and lichens shared as explained above

1.9.

160 "Tree cover, flooded, fresh or brakish water"
170 "Tree cover, flooded, saline water »
180 "Shrub or herbaceous cover, flooded, fresh/saline/brakish water"

160+170 and 180 five 2 new covers.

Script to apply in GRASS-GIS:
make_inondes.sh

- inputs:
    - ESACCI_LC_V1@francoise = to locate covers 160 & 170 & 180
- output:
    - classes_inondees

1.10.

190 "Urban areas"

Data sources used:

- CLC2012 on Europe http://land.copernicus.eu/pan-european/corine-land-cover/clc-2012
- http://ghsl.jrc.ec.europa.eu/, GHSL_LABEL, ftp://s-jrciprvm-ftp-ext.jrc.it/BETA_LABEL/, user: ghshare, pwd: nB34VOALk7 ("Aneta FLORCZYK" <aneta.florczyk@jrc.ec.europa.eu>)

Translation adopted:

CORINE urban classes:

1	Continuous urban fabric => LCZ2
2	Discontinuous urban fabric
3	Industrial or commercial units => LCZ8
4	Road and rail networks and associated land => LCZ9
5	Port areas => LCZ10
6	Airports => LCZ8

7	Mineral extraction sites => LCZ9
8	Dump sites => LCZ9
9	Construction sites => LCZ9
10	Green urban areas => LCZ9
11	Sport and leisure facilities => LCZ9

Beta label urban classes:
11 = Roads \\ => LCZ8

12 = Built-up with highly reflecting roof (associated to productive and commercial use)\\ => LCZ8

13 = Very light built-up NDVI > 0.4\\ => LCZ9

14 = Light built-up 0.3 < NDVI <= 0.4\\ => LCZ6

15 = Medium built-up 0.2 < NDVI <= 0.3\\ => LCZ5 (ou 6)

16 = Strong built-up NDVI <= 0.2 and low rise buildings (3D roughness <25m)\\ => LCZ3

17 = Strong built-up NDVI <= 0.2 and medium rise buildings (3D roughness 25-50m)\\ => LCZ2

18 = Strong built-up NDVI <= 0.2 and high rise buildings (3D roughness 50-100m)\\ => LCZ1

19 = Strong built-up NDVI <= 0.2 and very high rise buildings (3D roughness > 100m)\\ => LCZ1

NB :

24. LCZ1: compact high-rise

25. LCZ2: compact midrise
26. LCZ3: compact low-rise
27. LCZ4: open high-rise
28. LCZ5: open midrise
29: LCZ6: open low-rise
30: LCZ7: lightweight low-rise
31: LCZ8: large low-rise
32: LCZ9: sparsely built
33: LCZ10: heavy industry

Script to apply in GRASS-GIS:
make_new_town.sh:

- inputs:
  - ESACCI_LC_V1 = to locate cover 190
  - clc_town_300m = CORINE2012 agregated to 300m
  - agreg_villes_beta_label = beta_label with all tiles
- output:
  - new_town

1.11. Gathering of all pieces

We use the GRASS command **r.patch** to gather the pieces:

- all_crops
- natural_veg_mosaic
- mosaic_100_120
- sparse_lichen

- classes_inondees
- new_town
- ecosg_shrubs_grassland

1.12. Summary of scripts and input maps at each step

- ESACCI_LC_V1 is the common required input map.
- *input map – data* are the input maps needed again in case of realization of a new map from an update version of ESA-CCI LCC.
- *Input map – temp* are the input maps that would need to be built again in this case
- *grass commands* are the commands used to make the *input maps – temp*
- *r.mapcalc script / fortran program* are the programs or commands needed to build the output map of the step.

step	*Input maps - data*	*Input maps - temp*	*Grass commands*	*r.mapcalc script / fortran program*
1.1. water bodies	- rivers_lakes_sea_map_end1 - new_residu_sea_melange_step2 - swbd_usgs_tot - WDBII_river_f_all - GSHHS_f_water	- water_esa_cci	- r.mapcalc	- make_new_map.sh → new_map_tot
1.2. bare areas	- glc2000_africa / china / north_america / se_asia / south_asia / asia			- make_bare_rock_glc.sh → bare_rock_glc - make_ecosg_bare_soil_rock.sh → ecosg_bare_soil_rock
1.3. pure forest	- bioclim_bin0 / 5 / 6			- make_foret_climat.sh → new_foret_pure
1.4. mixed forest	- bioclim_bin 0 / 5 / 6	- foret_mixte	- r.mask + r.surf_random	- make_foret_tot.sh → ecosg_foret_tot
1.5. shrubland	- bioclim_bin 10 / 8			- make_new_shrubs_grassland.sh → ecosg_shrubs_grassland
1.6. mosaic veg	- bioclim_bin0 / 5 / 6 / 8 / 10	- ecosg_foret_tot_1d - foret_mixte - foret_mixte2	- r.resamp.stats - r.surf.random - r.surf.random	- make_interpol_foret.F90 → ecosg_foret_tot2_1d - make_mosaic_crop.sh → mosaic_crop - make_natural_veg_mosaic.sh → natural_veg_mosaic - make_mosaic_100_110.sh → mosaic_100_110
1.7. cropland	- VEGTYPE_P8_ECO2.3 - ISO3Code_2014 - usa_usda	- foret_mixte - foret_mixte2 - monde_random_c3 - france_random_c3 - ISO3Code_2014_c4_new - ISO3Code_2014_c4_fao - ISO3Code_2014_c3e_fao - ISO3Code_2014_c3h_fao - depts_c4_first_crops - depts_c4_agreste - depts_c3e_agreste - depts_c3h_agreste - agreg_all2	- r.surf.random - r.surf.random - r.surf.random - r.surf.random - r.recode - r.recode - r.recode - r.recode - r.recode - r.recode - r.recode - r.recode - r.recode + r.resamp.stats +	- make_first_crop.sh → first_crop - make_pourcentages_cultures.F90 → pourcentages_c*_par_pays_b.csv - make_new_crops_20.sh → new_crops - make_new_crops2.sh → new_crop2 - read_moy_depts.F90 → recode_depts_c*.txt - make_new_crops_france.sh → new_crop_france - make_new_crops2_france.sh → new_crop2_france - make_recode_usda.F90 → usda_crops_recode.txt

			r.patch	- make_new_usa_crops.sh → new_usa_crops - make_all_crops.sh → all_crops
1.8. sparse veg	- bioclim_bin10 / 8	- aleat_sparse_lichen	- r.surf.random	- make_sparse_lichen.sh → sparse_lichen
1.9. flooded veg				- make_inondes.sh → classes_inondees
1.10. urban		- clc_town_300m - agreg_villes_beta_label	- r.mapcalc + r.resamp.stats - r.proj + r.patch + r.resamp.stats	- make_new_town.sh → new_town
1.11. merge			r.patch	→ ecosg_final_map

- To apply the scripts above, it's also required to know how to use the GRASS function g.region.
- Concerning the crops, if a new map has to be made, the preliminary work of downloading FAO, AGRESTE and USDA data and treat them will be needed again.

1.13. Known problems

- the climate map is at 1° resolution and leads to artificial limits between the classes → needs to be smoothed or redone with a finer map
- the shrubs are quite a lot represented in the map, and would need a finer distinction between broadleaf and needleleaf
- the deciduous needleleaf in Russia would need to be mixed with other types of forests

2) Maps of primary parameters

2.1. Maps of LAI

- LAI data were downloaded at https://land.copernicus.eu/global/.
- They are at 300m resolution for years 2014-2016.

Programs / GRASS scripts to run to obtain the maps read by SURFEX:

- **make_new_lai.F90:** makes the average of the available years in binary format
- **make_interpol.F90:** interpols each grid point time series when values are missing
- **resample.sh:** projects the map on our ECOCLIMAP grid (GRASS)
- **make_compress_files.sh:**
  - **make_new_val0.F90:** puts the information of vegtype in the data binary file
  - **make_compres_files_300m0.F90:** compresses the data binary file for all vegtypes

2.2. Maps of ALBEDO

- ALBEDO data were downloaded at https://land.copernicus.eu/global/.
- They are at 1km resolution for years 1999-20164

Programs / GRASS scripts to run to obtain the maps read by SURFEX:

- **make_new_alb.F90:** makes the average of the available years in binary format
- **make_interpol_alb.F90:** interpols each grid point time series when values are missing
- **import.sh:** projects the map on our ECOCLIMAP grid (GRASS)
- **make_new_alb_veg.F90:** separates the soil albedo from the vegetation albedo using the ECOCLIMAP formulas

        - inputs:
            - LAI maps obtained before = for the calculation of the VEG of crops
            - new_albvis_soil = obtained with the script **make_alb_soil.sh**:
                - inputs:
                    - SAND_HWSD_MOY_v2@hwsd
                    - ecosg_final_map
- **make_compress_files_ANS/AVS/ANV/AVV.sh:**
    - **make_new_val0.F90:** puts the information of vegtype in the data binary file
    - **make_compres_files_300m0.F90:** compresses the data binary file for all vegtypes

2.3. Map of height of trees

The base map was taken from **https://landscape.jpl.nasa.gov/** (Simard 2012).

Scripts used to build the map:
a) make_new_height_tree.sh:

- inputs:
    - ecosg_final_map = to locate the tree covers
    - HEIGHT_TREE_*_c = height of trees from current ECOCLIMAP extrapolated on the ECOCLIMAP SG covers, for each tree vegtype
- output:
    - new_height_tree = height of trees from current ECOCLIMAP extrapolated on the ECOCLIMAP SG covers

b) make_new_height_tree_b.sh:

- inputs:
    - new_height_tree = from above
    - height_of_trees_simard = direct download
    - height_of_trees_simard_1d = resample to 1 degree with r.resamp.stats
- output:
    - new_height_tree_b = height of trees used in ECOCLIMAP-SG, before compression

3) How to use ECOCLIMAP-SG in SURFEX on Beaufix

3.1. Where to find the needed input files

- /scratch/work/faroux/PGD/**ecosg_final_map.dir / .hdr** : map of covers at 300m resolution

- /scratch/work/faroux/PARAMETERS/LAI/ESG/V4/**LAI_*_c.dir / .hdr**: compressed maps of LAI to be read by Surfex, for each 10 day – period

- /scratch/work/faroux/PARAMETERS/ANS/ESG/V3/**AL_NI_SOI_*_c.dir / .hdr**: compressed maps of soil NIR albedo to be read by Surfex, for each 10 day - period
- /scratch/work/faroux/PARAMETERS/ANV/ESG/V3/**AL_NI_VEG_*_c.dir / .hdr**: compressed maps of vegetation NIR albedo to be read by Surfex, for each 10 day - period
- /scratch/work/faroux/PARAMETERS/AVS/ESG/V3/**AL_VI_SOI_*_c.dir / .hdr**:
- compressed maps of soil VIS albedo to be read by Surfex, for each 10 day - period
- /scratch/work/faroux/PARAMETERS/AVV/ESG/V3/**AL_VI_VEG_*_c.dir / .hdr**: compressed maps of vegetation VIS albedo to be read by Surfex, for each 10 day – period

- /scratch/work/faroux/PARAMETERS/HT/**new_ht_c.dir / .hdr**: compressed map of height of trees to be read by Surfex

3.2. Changes in the namelist

```
&NAM_FRAC
LECOSG = T
/

&NAM_COVER
YCOVER = “ecosg_final_map”
YCOVERFILETYPE = “DIRECT”
/

&NAM_DATA_ISBA
NTIME = 36
XUNIF_DICE(1) = 0.5
XUNIF_DICE(2) = 0.2
XUNIF_DICE(3) = 0.2
XUNIF_DICE(4) = 1.0
XUNIF_DICE(5) = 1.5
XUNIF_DICE(6) = 2.0
XUNIF_DICE(7) = 1.5
XUNIF_DICE(8) = 3.0
XUNIF_DICE(9) = 1.0
XUNIF_DICE(10) = 2.0
XUNIF_DICE(11) = 1.0
XUNIF_DICE(12) = 1.0
XUNIF_DICE(13) = 1.0
XUNIF_DICE(14) = 1.2
XUNIF_DICE(15) = 3.0
XUNIF_DICE(16) = 1.2
XUNIF_DICE(17) = 1.5
XUNIF_DICE(18) = 1.5
XUNIF_DICE(19) = 2.0
XUNIF_DICE(20) = 1.5
XUNIF_ROOT_DEPTH(1) = 0.5
XUNIF_ROOT_DEPTH(2) = 0.2
XUNIF_ROOT_DEPTH(3) = 0.2
XUNIF_ROOT_DEPTH(4) = 1.0
XUNIF_ROOT_DEPTH(5) = 1.5
XUNIF_ROOT_DEPTH(6) = 2.0
XUNIF_ROOT_DEPTH(7) = 1.5
XUNIF_ROOT_DEPTH(8) = 3.0
XUNIF_ROOT_DEPTH(9) = 1.0
XUNIF_ROOT_DEPTH(10) = 2.0
XUNIF_ROOT_DEPTH(11) = 1.0
XUNIF_ROOT_DEPTH(12) = 1.0
XUNIF_ROOT_DEPTH(13) = 1.0
XUNIF_ROOT_DEPTH(14) = 1.2
XUNIF_ROOT_DEPTH(15) = 3.0
XUNIF_ROOT_DEPTH(16) = 1.2
XUNIF_ROOT_DEPTH(17) = 1.5
XUNIF_ROOT_DEPTH(18) = 1.5
XUNIF_ROOT_DEPTH(19) = 2.0
XUNIF_ROOT_DEPTH(20) = 1.5
XUNIF_GROUND_DEPTH(1) = 0.5
XUNIF_GROUND_DEPTH(2) = 0.2
XUNIF_GROUND_DEPTH(3) = 0.2
XUNIF_GROUND_DEPTH(4) = 2.0
XUNIF_GROUND_DEPTH(5) = 3.0
XUNIF_GROUND_DEPTH(6) = 4.0
XUNIF_GROUND_DEPTH(7) = 3.0
XUNIF_GROUND_DEPTH(8) = 6.0
XUNIF_GROUND_DEPTH(9) = 2.0
XUNIF_GROUND_DEPTH(10) = 3.0
XUNIF_GROUND_DEPTH(11) = 2.0
XUNIF_GROUND_DEPTH(12) = 2.0
XUNIF_GROUND_DEPTH(13) = 2.0
XUNIF_GROUND_DEPTH(14) = 1.2
XUNIF_GROUND_DEPTH(15) = 4.0
XUNIF_GROUND_DEPTH(16) = 1.2
XUNIF_GROUND_DEPTH(17) = 1.5
XUNIF_GROUND_DEPTH(18) = 1.5
XUNIF_GROUND_DEPTH(19) = 3.0
XUNIF_GROUND_DEPTH(20) = 2.5
```

CFNAM_LAI(1,1) = 'LAI_0105_c'
CFTYP_LAI(1,1) = 'DIRTYP'
CFNAM_LAI(1,2) = 'LAI_0115_c'
CFTYP_LAI(1,2) = 'DIRTYP'
CFNAM_LAI(1,3) = 'LAI_0125_c'
CFTYP_LAI(1,3) = 'DIRTYP'
CFNAM_LAI(1,4) = 'LAI_0205_c'
CFTYP_LAI(1,4) = 'DIRTYP'
CFNAM_LAI(1,5) = 'LAI_0215_c'
CFTYP_LAI(1,5) = 'DIRTYP'
CFNAM_LAI(1,6) = 'LAI_0225_c'
CFTYP_LAI(1,6) = 'DIRTYP'
CFNAM_LAI(1,7) = 'LAI_0305_c'
CFTYP_LAI(1,7) = 'DIRTYP'
CFNAM_LAI(1,8) = 'LAI_0315_c'
CFTYP_LAI(1,8) = 'DIRTYP'
CFNAM_LAI(1,9) = 'LAI_0325_c'
CFTYP_LAI(1,9) = 'DIRTYP'
CFNAM_LAI(1,10) = 'LAI_0405_c'
CFTYP_LAI(1,10) = 'DIRTYP'
CFNAM_LAI(1,11) = 'LAI_0415_c'
CFTYP_LAI(1,11) = 'DIRTYP'
CFNAM_LAI(1,12) = 'LAI_0425_c'
CFTYP_LAI(1,12) = 'DIRTYP'
CFNAM_LAI(1,13) = 'LAI_0505_c'
CFTYP_LAI(1,13) = 'DIRTYP'
CFNAM_LAI(1,14) = 'LAI_0515_c'
CFTYP_LAI(1,14) = 'DIRTYP'
CFNAM_LAI(1,15) = 'LAI_0525_c'
CFTYP_LAI(1,15) = 'DIRTYP'
CFNAM_LAI(1,16) = 'LAI_0605_c'
CFTYP_LAI(1,16) = 'DIRTYP'
CFNAM_LAI(1,17) = 'LAI_0615_c'
CFTYP_LAI(1,17) = 'DIRTYP'
CFNAM_LAI(1,18) = 'LAI_0625_c'
CFTYP_LAI(1,18) = 'DIRTYP'
CFNAM_LAI(1,19) = 'LAI_0705_c'
CFTYP_LAI(1,19) = 'DIRTYP'
CFNAM_LAI(1,20) = 'LAI_0715_c'
CFTYP_LAI(1,20) = 'DIRTYP'
CFNAM_LAI(1,21) = 'LAI_0725_c'
CFTYP_LAI(1,21) = 'DIRTYP'
CFNAM_LAI(1,22) = 'LAI_0805_c'
CFTYP_LAI(1,22) = 'DIRTYP'
CFNAM_LAI(1,23) = 'LAI_0815_c'
CFTYP_LAI(1,23) = 'DIRTYP'
CFNAM_LAI(1,24) = 'LAI_0825_c'
CFTYP_LAI(1,24) = 'DIRTYP'
CFNAM_LAI(1,25) = 'LAI_0905_c'
CFTYP_LAI(1,25) = 'DIRTYP'
CFNAM_LAI(1,26) = 'LAI_0915_c'
CFTYP_LAI(1,26) = 'DIRTYP'
CFNAM_LAI(1,27) = 'LAI_0925_c'
CFTYP_LAI(1,27) = 'DIRTYP'
CFNAM_LAI(1,28) = 'LAI_1005_c'
CFTYP_LAI(1,28) = 'DIRTYP'
CFNAM_LAI(1,29) = 'LAI_1015_c'
CFTYP_LAI(1,29) = 'DIRTYP'
CFNAM_LAI(1,30) = 'LAI_1025_c'
CFTYP_LAI(1,30) = 'DIRTYP'
CFNAM_LAI(1,31) = 'LAI_1105_c'
CFTYP_LAI(1,31) = 'DIRTYP'
CFNAM_LAI(1,32) = 'LAI_1115_c'
CFTYP_LAI(1,32) = 'DIRTYP'
CFNAM_LAI(1,33) = 'LAI_1125_c'
CFTYP_LAI(1,33) = 'DIRTYP'
CFNAM_LAI(1,34) = 'LAI_1205_c'
CFTYP_LAI(1,34) = 'DIRTYP'
CFNAM_LAI(1,35) = 'LAI_1215_c'
CFTYP_LAI(1,35) = 'DIRTYP'
CFNAM_LAI(1,36) = 'LAI_1225_c'
CFTYP_LAI(1,36) = 'DIRTYP'
CFNAM_H_TREE(1) = "new_ht_c"
CFTYP_H_TREE(1) = "DIRTYP"
CFNAM_ALBNIR_VEG(1,1) = 'AL_NI_VEG_0105_c'
CFTYP_ALBNIR_VEG(1,1) = 'DIRTYP'

CFNAM_ALBNIR_VEG(1,2) = 'AL_NI_VEG_0115_c'
CFTYP_ALBNIR_VEG(1,2) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,3) = 'AL_NI_VEG_0125_c'
CFTYP_ALBNIR_VEG(1,3) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,4) = 'AL_NI_VEG_0205_c'
CFTYP_ALBNIR_VEG(1,4) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,5) = 'AL_NI_VEG_0215_c'
CFTYP_ALBNIR_VEG(1,5) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,6) = 'AL_NI_VEG_0225_c'
CFTYP_ALBNIR_VEG(1,6) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,7) = 'AL_NI_VEG_0305_c'
CFTYP_ALBNIR_VEG(1,7) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,8) = 'AL_NI_VEG_0315_c'
CFTYP_ALBNIR_VEG(1,8) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,9) = 'AL_NI_VEG_0325_c'
CFTYP_ALBNIR_VEG(1,9) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,10) = 'AL_NI_VEG_0405_c'
CFTYP_ALBNIR_VEG(1,10) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,11) = 'AL_NI_VEG_0415_c'
CFTYP_ALBNIR_VEG(1,11) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,12) = 'AL_NI_VEG_0425_c'
CFTYP_ALBNIR_VEG(1,12) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,13) = 'AL_NI_VEG_0505_c'
CFTYP_ALBNIR_VEG(1,13) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,14) = 'AL_NI_VEG_0515_c'
CFTYP_ALBNIR_VEG(1,14) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,15) = 'AL_NI_VEG_0525_c'
CFTYP_ALBNIR_VEG(1,15) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,16) = 'AL_NI_VEG_0605_c'
CFTYP_ALBNIR_VEG(1,16) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,17) = 'AL_NI_VEG_0615_c'
CFTYP_ALBNIR_VEG(1,17) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,18) = 'AL_NI_VEG_0625_c'
CFTYP_ALBNIR_VEG(1,18) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,19) = 'AL_NI_VEG_0705_c'
CFTYP_ALBNIR_VEG(1,19) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,20) = 'AL_NI_VEG_0715_c'
CFTYP_ALBNIR_VEG(1,20) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,21) = 'AL_NI_VEG_0725_c'
CFTYP_ALBNIR_VEG(1,21) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,22) = 'AL_NI_VEG_0805_c'
CFTYP_ALBNIR_VEG(1,22) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,23) = 'AL_NI_VEG_0815_c'
CFTYP_ALBNIR_VEG(1,23) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,24) = 'AL_NI_VEG_0825_c'
CFTYP_ALBNIR_VEG(1,24) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,25) = 'AL_NI_VEG_0905_c'
CFTYP_ALBNIR_VEG(1,25) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,26) = 'AL_NI_VEG_0915_c'
CFTYP_ALBNIR_VEG(1,26) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,27) = 'AL_NI_VEG_0925_c'
CFTYP_ALBNIR_VEG(1,27) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,28) = 'AL_NI_VEG_1005_c'
CFTYP_ALBNIR_VEG(1,28) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,29) = 'AL_NI_VEG_1015_c'
CFTYP_ALBNIR_VEG(1,29) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,30) = 'AL_NI_VEG_1025_c'
CFTYP_ALBNIR_VEG(1,30) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,31) = 'AL_NI_VEG_1105_c'
CFTYP_ALBNIR_VEG(1,31) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,32) = 'AL_NI_VEG_1115_c'
CFTYP_ALBNIR_VEG(1,32) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,33) = 'AL_NI_VEG_1125_c'
CFTYP_ALBNIR_VEG(1,33) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,34) = 'AL_NI_VEG_1205_c'
CFTYP_ALBNIR_VEG(1,34) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,35) = 'AL_NI_VEG_1215_c'
CFTYP_ALBNIR_VEG(1,35) = 'DIRTYP'
CFNAM_ALBNIR_VEG(1,36) = 'AL_NI_VEG_1225_c'
CFTYP_ALBNIR_VEG(1,36) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,1) = 'AL_VI_VEG_0105_c'
CFTYP_ALBVIS_VEG(1,1) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,2) = 'AL_VI_VEG_0115_c'
CFTYP_ALBVIS_VEG(1,2) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,3) = 'AL_VI_VEG_0125_c'
CFTYP_ALBVIS_VEG(1,3) = 'DIRTYP'

```
CFNAM_ALBVIS_VEG(1,4) = 'AL_VI_VEG_0205_c'
CFTYP_ALBVIS_VEG(1,4) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,5) = 'AL_VI_VEG_0215_c'
CFTYP_ALBVIS_VEG(1,5) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,6) = 'AL_VI_VEG_0225_c'
CFTYP_ALBVIS_VEG(1,6) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,7) = 'AL_VI_VEG_0305_c'
CFTYP_ALBVIS_VEG(1,7) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,8) = 'AL_VI_VEG_0315_c'
CFTYP_ALBVIS_VEG(1,8) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,9) = 'AL_VI_VEG_0325_c'
CFTYP_ALBVIS_VEG(1,9) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,10) = 'AL_VI_VEG_0405_c'
CFTYP_ALBVIS_VEG(1,10) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,11) = 'AL_VI_VEG_0415_c'
CFTYP_ALBVIS_VEG(1,11) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,12) = 'AL_VI_VEG_0425_c'
CFTYP_ALBVIS_VEG(1,12) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,13) = 'AL_VI_VEG_0505_c'
CFTYP_ALBVIS_VEG(1,13) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,14) = 'AL_VI_VEG_0515_c'
CFTYP_ALBVIS_VEG(1,14) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,15) = 'AL_VI_VEG_0525_c'
CFTYP_ALBVIS_VEG(1,15) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,16) = 'AL_VI_VEG_0605_c'
CFTYP_ALBVIS_VEG(1,16) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,17) = 'AL_VI_VEG_0615_c'
CFTYP_ALBVIS_VEG(1,17) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,18) = 'AL_VI_VEG_0625_c'
CFTYP_ALBVIS_VEG(1,18) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,19) = 'AL_VI_VEG_0705_c'
CFTYP_ALBVIS_VEG(1,19) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,20) = 'AL_VI_VEG_0715_c'
CFTYP_ALBVIS_VEG(1,20) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,21) = 'AL_VI_VEG_0725_c'
CFTYP_ALBVIS_VEG(1,21) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,22) = 'AL_VI_VEG_0805_c'
CFTYP_ALBVIS_VEG(1,22) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,23) = 'AL_VI_VEG_0815_c'
CFTYP_ALBVIS_VEG(1,23) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,24) = 'AL_VI_VEG_0825_c'
CFTYP_ALBVIS_VEG(1,24) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,25) = 'AL_VI_VEG_0905_c'
CFTYP_ALBVIS_VEG(1,25) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,26) = 'AL_VI_VEG_0915_c'
CFTYP_ALBVIS_VEG(1,26) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,27) = 'AL_VI_VEG_0925_c'
CFTYP_ALBVIS_VEG(1,27) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,28) = 'AL_VI_VEG_1005_c'
CFTYP_ALBVIS_VEG(1,28) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,29) = 'AL_VI_VEG_1015_c'
CFTYP_ALBVIS_VEG(1,29) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,30) = 'AL_VI_VEG_1025_c'
CFTYP_ALBVIS_VEG(1,30) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,31) = 'AL_VI_VEG_1105_c'
CFTYP_ALBVIS_VEG(1,31) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,32) = 'AL_VI_VEG_1115_c'
CFTYP_ALBVIS_VEG(1,32) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,33) = 'AL_VI_VEG_1125_c'
CFTYP_ALBVIS_VEG(1,33) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,34) = 'AL_VI_VEG_1205_c'
CFTYP_ALBVIS_VEG(1,34) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,35) = 'AL_VI_VEG_1215_c'
CFTYP_ALBVIS_VEG(1,35) = 'DIRTYP'
CFNAM_ALBVIS_VEG(1,36) = 'AL_VI_VEG_1225_c'
CFTYP_ALBVIS_VEG(1,36) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,1) = 'AL_NI_SOI_0105_c'
CFTYP_ALBNIR_SOIL(1,1) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,2) = 'AL_NI_SOI_0115_c'
CFTYP_ALBNIR_SOIL(1,2) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,3) = 'AL_NI_SOI_0125_c'
CFTYP_ALBNIR_SOIL(1,3) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,4) = 'AL_NI_SOI_0205_c'
CFTYP_ALBNIR_SOIL(1,4) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,5) = 'AL_NI_SOI_0215_c'
CFTYP_ALBNIR_SOIL(1,5) = 'DIRTYP'
```

CFNAM_ALBNIR_SOIL(1,6) = 'AL_NI_SOI_0225_c'
CFTYP_ALBNIR_SOIL(1,6) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,7) = 'AL_NI_SOI_0305_c'
CFTYP_ALBNIR_SOIL(1,7) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,8) = 'AL_NI_SOI_0315_c'
CFTYP_ALBNIR_SOIL(1,8) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,9) = 'AL_NI_SOI_0325_c'
CFTYP_ALBNIR_SOIL(1,9) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,10) = 'AL_NI_SOI_0405_c'
CFTYP_ALBNIR_SOIL(1,10) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,11) = 'AL_NI_SOI_0415_c'
CFTYP_ALBNIR_SOIL(1,11) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,12) = 'AL_NI_SOI_0425_c'
CFTYP_ALBNIR_SOIL(1,12) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,13) = 'AL_NI_SOI_0505_c'
CFTYP_ALBNIR_SOIL(1,13) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,14) = 'AL_NI_SOI_0515_c'
CFTYP_ALBNIR_SOIL(1,14) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,15) = 'AL_NI_SOI_0525_c'
CFTYP_ALBNIR_SOIL(1,15) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,16) = 'AL_NI_SOI_0605_c'
CFTYP_ALBNIR_SOIL(1,16) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,17) = 'AL_NI_SOI_0615_c'
CFTYP_ALBNIR_SOIL(1,17) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,18) = 'AL_NI_SOI_0625_c'
CFTYP_ALBNIR_SOIL(1,18) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,19) = 'AL_NI_SOI_0705_c'
CFTYP_ALBNIR_SOIL(1,19) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,20) = 'AL_NI_SOI_0715_c'
CFTYP_ALBNIR_SOIL(1,20) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,21) = 'AL_NI_SOI_0725_c'
CFTYP_ALBNIR_SOIL(1,21) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,22) = 'AL_NI_SOI_0805_c'
CFTYP_ALBNIR_SOIL(1,22) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,23) = 'AL_NI_SOI_0815_c'
CFTYP_ALBNIR_SOIL(1,23) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,24) = 'AL_NI_SOI_0825_c'
CFTYP_ALBNIR_SOIL(1,24) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,25) = 'AL_NI_SOI_0905_c'
CFTYP_ALBNIR_SOIL(1,25) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,26) = 'AL_NI_SOI_0915_c'
CFTYP_ALBNIR_SOIL(1,26) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,27) = 'AL_NI_SOI_0925_c'
CFTYP_ALBNIR_SOIL(1,27) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,28) = 'AL_NI_SOI_1005_c'
CFTYP_ALBNIR_SOIL(1,28) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,29) = 'AL_NI_SOI_1015_c'
CFTYP_ALBNIR_SOIL(1,29) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,30) = 'AL_NI_SOI_1025_c'
CFTYP_ALBNIR_SOIL(1,30) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,31) = 'AL_NI_SOI_1105_c'
CFTYP_ALBNIR_SOIL(1,31) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,32) = 'AL_NI_SOI_1115_c'
CFTYP_ALBNIR_SOIL(1,32) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,33) = 'AL_NI_SOI_1125_c'
CFTYP_ALBNIR_SOIL(1,33) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,34) = 'AL_NI_SOI_1205_c'
CFTYP_ALBNIR_SOIL(1,34) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,35) = 'AL_NI_SOI_1215_c'
CFTYP_ALBNIR_SOIL(1,35) = 'DIRTYP'
CFNAM_ALBNIR_SOIL(1,36) = 'AL_NI_SOI_1225_c'
CFTYP_ALBNIR_SOIL(1,36) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,1) = 'AL_VI_SOI_0105_c'
CFTYP_ALBVIS_SOIL(1,1) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,2) = 'AL_VI_SOI_0115_c'
CFTYP_ALBVIS_SOIL(1,2) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,3) = 'AL_VI_SOI_0125_c'
CFTYP_ALBVIS_SOIL(1,3) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,4) = 'AL_VI_SOI_0205_c'
CFTYP_ALBVIS_SOIL(1,4) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,5) = 'AL_VI_SOI_0215_c'
CFTYP_ALBVIS_SOIL(1,5) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,6) = 'AL_VI_SOI_0225_c'
CFTYP_ALBVIS_SOIL(1,6) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,7) = 'AL_VI_SOI_0305_c'
CFTYP_ALBVIS_SOIL(1,7) = 'DIRTYP'

```
CFNAM_ALBVIS_SOIL(1,8) = 'AL_VI_SOI_0315_c'
CFTYP_ALBVIS_SOIL(1,8) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,9) = 'AL_VI_SOI_0325_c'
CFTYP_ALBVIS_SOIL(1,9) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,10) = 'AL_VI_SOI_0405_c'
CFTYP_ALBVIS_SOIL(1,10) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,11) = 'AL_VI_SOI_0415_c'
CFTYP_ALBVIS_SOIL(1,11) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,12) = 'AL_VI_SOI_0425_c'
CFTYP_ALBVIS_SOIL(1,12) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,13) = 'AL_VI_SOI_0505_c'
CFTYP_ALBVIS_SOIL(1,13) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,14) = 'AL_VI_SOI_0515_c'
CFTYP_ALBVIS_SOIL(1,14) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,15) = 'AL_VI_SOI_0525_c'
CFTYP_ALBVIS_SOIL(1,15) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,16) = 'AL_VI_SOI_0605_c'
CFTYP_ALBVIS_SOIL(1,16) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,17) = 'AL_VI_SOI_0615_c'
CFTYP_ALBVIS_SOIL(1,17) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,18) = 'AL_VI_SOI_0625_c'
CFTYP_ALBVIS_SOIL(1,18) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,19) = 'AL_VI_SOI_0705_c'
CFTYP_ALBVIS_SOIL(1,19) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,20) = 'AL_VI_SOI_0715_c'
CFTYP_ALBVIS_SOIL(1,20) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,21) = 'AL_VI_SOI_0725_c'
CFTYP_ALBVIS_SOIL(1,21) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,22) = 'AL_VI_SOI_0805_c'
CFTYP_ALBVIS_SOIL(1,22) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,23) = 'AL_VI_SOI_0815_c'
CFTYP_ALBVIS_SOIL(1,23) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,24) = 'AL_VI_SOI_0825_c'
CFTYP_ALBVIS_SOIL(1,24) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,25) = 'AL_VI_SOI_0905_c'
CFTYP_ALBVIS_SOIL(1,25) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,26) = 'AL_VI_SOI_0915_c'
CFTYP_ALBVIS_SOIL(1,26) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,27) = 'AL_VI_SOI_0925_c'
CFTYP_ALBVIS_SOIL(1,27) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,28) = 'AL_VI_SOI_1005_c'
CFTYP_ALBVIS_SOIL(1,28) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,29) = 'AL_VI_SOI_1015_c'
CFTYP_ALBVIS_SOIL(1,29) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,30) = 'AL_VI_SOI_1025_c'
CFTYP_ALBVIS_SOIL(1,30) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,31) = 'AL_VI_SOI_1105_c'
CFTYP_ALBVIS_SOIL(1,31) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,32) = 'AL_VI_SOI_1115_c'
CFTYP_ALBVIS_SOIL(1,32) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,33) = 'AL_VI_SOI_1125_c'
CFTYP_ALBVIS_SOIL(1,33) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,34) = 'AL_VI_SOI_1205_c'
CFTYP_ALBVIS_SOIL(1,34) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,35) = 'AL_VI_SOI_1215_c'
CFTYP_ALBVIS_SOIL(1,35) = 'DIRTYP'
CFNAM_ALBVIS_SOIL(1,36) = 'AL_VI_SOI_1225_c'
CFTYP_ALBVIS_SOIL(1,36) = 'DIRTYP'
/
```

- The namelist NAM_DATA_ISBA contains uniform values for XUNIF_DICE, XUNIF_ROOT_DEPTH and XUNIF_GROUND_DEPTH (can be changed).
- The type ‘DIRTYP’ means that one map contains the information for all vegetation types.

3.3. To run PGD and PREP with ECOCLIMAP-SG

It can be useful to use the parallelization of the PGD and PREP steps availables in version 8.1, because with ECOCLIMAP-SG PGD and PREP can be quite long and heavy.