Installation and use guide

version 2.0

November 2021

DiSAA – Department of Agricultural and Environmental Sciences

University of Milan

IdrAgra Installation and use guide - version 2.0

November 2021

DiSAA – Department of Agricultural and Environmental Sciences

University of Milan

contact: Prof. Claudio Gandolfi – claudio.gandolfi@unimi.it

Table of Contents

1 Software installation and use

Both the IdrAgra model and the Cropcoef module are distributed as Command Line Interface, CLI, program. Minimal experiences with the Command prompt of Windows O.S. is required. Save all the executable in a folder and annotate the path (e.g. C:\path_to_idragra). Prepare all the necessary simulation file and save them in the same root folder (e.g. C:\path_to_simulation).

1.1 System requirements

Please check on your PC to have the following software installed:

- O.S. Windows 7 or later;
- MATLAB runtime 9.9 (https://it.mathworks.com/products/compiler/matlab-runtime.html);
- MinGW compiler (https://sourceforge.net/projects/mingw/).

Please check that both the binary folder of the MATLAB runtime (e.g. C:\Program Files\MATLAB\MATLAB Runtime\v99\runtime\win64) and the MinGW compiler (e.g. C:\MinGW\bin) are in the user or system Path variable.

To edit/check Path variable, open Control Panel → System → Advanced → Environment Variables. Then, double click on the Path variable in both the user variables list and in the system variables list and search for the required paths. If missing, add them to the list.

Any text editor is useful to prepare the required simulation files. Among others, the Scite editor (https://www.scintilla.org/SciTE.html) is suggested because it supports also batch file execution.

1.2 CropCoef

Currently, the CropCoef model is distributed as an application: CropCoef_v4.exe.

From the windows command line, type and execute the following two lines:

    cd c:\path_to_simulation
    c:\path_to_idragra\CropCoef_v4.exe "cropcoef.txt"

Where cropcoef.txt is the name of the CropCoef parameters file that is saved under the simulation folder. Note that all the required files and relative sub-folders must be under the same root path.

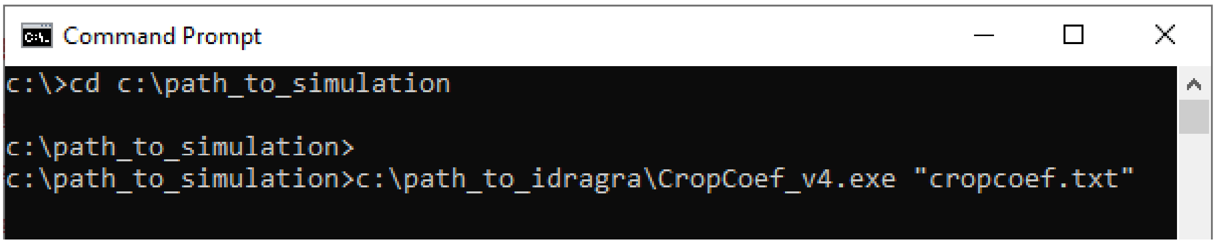

Figure 1: Commands to run CropCoef.

1.3 IdrAgra

The IdrAgra model is distributed as CLI application: idragra_XXX.exe, here XXX indicates the release number. From the windows command line, type and execute the following two lines:

cd c:\path_to_simulation

c:\path_to_idragra\idragra_XXX.exe "idragra_parameters.txt"

where "idragra_parameters.txt" is the name of the IdrAgra parameters file that is saved under the simulation folder. Note that all the required files and relative sub-folders must be under the same root path.

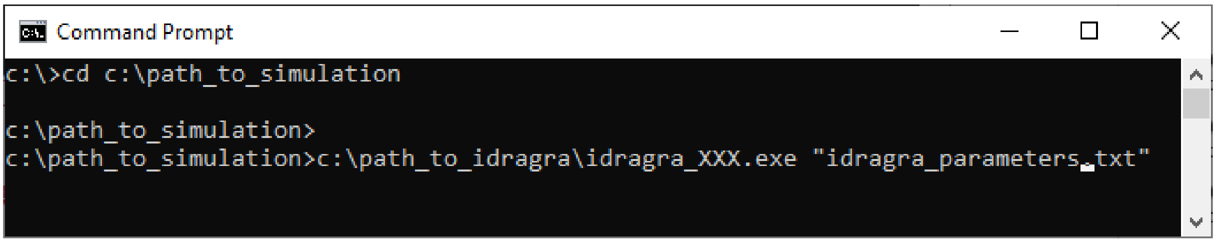

Figure 2: Commands to run IdrAgra.

2 CropCoef input files

Input files for the CropCoef code must be located all in the same folder that will be referred to as *<Working Folder>*[1] hereafter. An overview of the input structure is shown in Figure 3. Mandatory input files that must be present in the *<Working Folder>* are:

- main input file *<cropcoef>.txt*, containing the names of the input files and the paths to reach them;
- file *<weather_stations>.dat* with the number and list of the names of weather stations;
- folder *<meteo_data>* containing the meteorological time series recorded at the weather stations listed in the file *<weather_stations>.dat*; each timeseries must be contained in a file, whose name is the same reported in *<weather_stations>.dat;*
- folder *<crop_inputs>*, containing information about land use and crop parameters; land use information are contained in the file '*soil_uses.txt*', while the subfolder '*crop_parameters*' includes the crop parameters files (*<crop>.tab,* one file per crop) for the different crops listed in '*soil_uses.txt*'.

The working folder may also contain another file named '*CO2_conc.dat*' that is required only when the user runs a simulation where the canopy resistance and the biomass water productivity must be corrected according to CO_2 concentration values that are different from the default of 330 ppm.

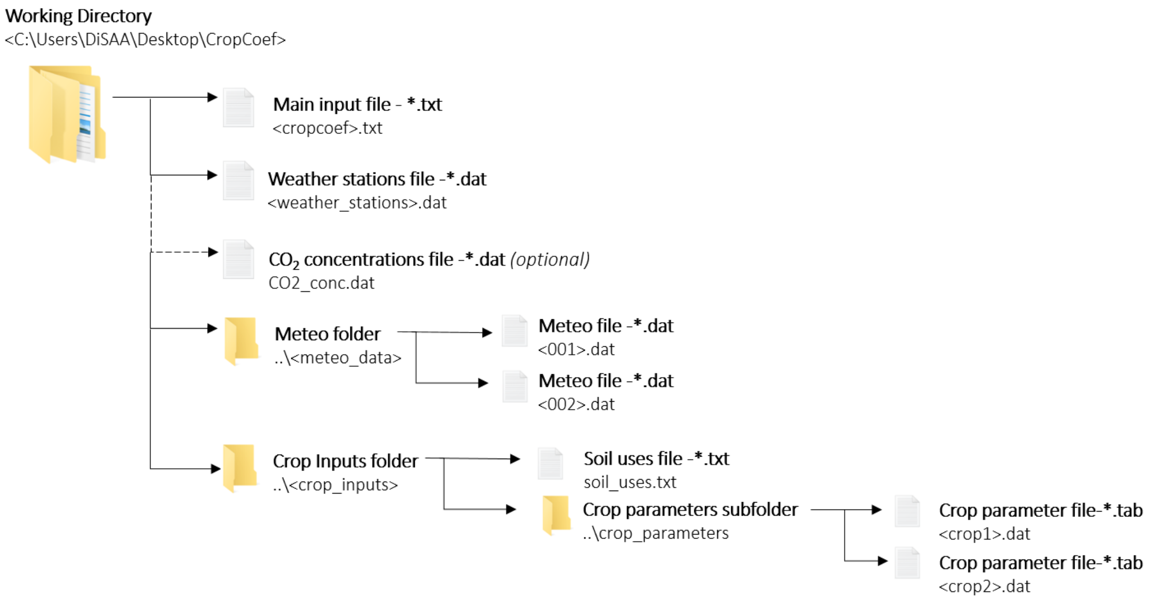

Figure 3: Structure of CropCoef inputs (the symbols '<>' indicate names that can be defined by the user).

[1] *<name>* indicates a folder, file or variable whose name is customizable; '*name*' indicates a folder, file or variable whose name is fixed and non-modifiable

2.1 CropCoef controls

The <*cropcoef*>*.txt* file contains the names and paths to the files required by the CropCoef model and a switch to select the correction of canopy resistance (and crop water productivity) based on CO_2 concentrations. Contents and structure of the file are shown in Table 1. When the model is launched, the user is asked to search for the file folder and select the main CropCoef input file. This folder will be the working folder where model inputs and model outputs must be located.

Table 1: Main CropCoef input file.

#### Main CropCoef file ####		
# Part 1: Inputs		
WeathStatFilename	= weather_stations.dat	# name of file with information about weather stations
MeteoDataFolder	= meteo_data	#name of the folder with meteo data files
CropInputsFolder	= crop_inputs	# name of the folder with crop inputs files
# Part 2: Outputs		
OutputFolder	= crop_series	# name of the folder with output files
# Part 3: Simulation Options		
CanopyResMod	= 1	# value of canopy resistance printed in output [0 = def value (70 s/m), 1 = calculated as function of CO2]
# end of main CropCoef files		

2.2 Weather stations inputs

The weather station input file <*weather_stations*>*.dat* contains the number of weather stations used for the simulation (variable '*StatNum*' in Table 2) and a table with the list of weather stations. For each station, user must provide the code number of the station (column 1) that must be in the form <*yyy*>*.dat* (where *yyy* is any sequence of maximum 26 numeric characters), the x coordinate (column 2) and y coordinate (column 3) in a selected reference system (default projection is WGS 84 / UTM zone 32, EPSG projection: 32632). CropCoef does not use spatial coordinates of weather stations, but the same file is also read by the IdrAgra model that requires the spatial coordinates of weather stations to spatialize meteorological data.

Table 2: Weather stations input file.

# StatNum: number of meteorological stations		
StatNum = 5		
# Table: table containing meteorological station list and their coordinates (the 1st line of the table is skipped)		
# Table starts with the label "Table =" and ends with the label "EndTable ="		
Table =		
SAR code	x_coord	y_coord
100.dat	520118.8913825110	5038170.9371612700
109.dat	531812.7895762110	5009569.5267708000
114.dat	520712.9616276510	5018618.3093939000
123.dat	544064.5435133280	5013079.5112939100
137.dat	540719.5263381300	5032417.1316996700
EndTable =		

2.3 Meteorological time series

Meteorological time series files <*yyy*>*.dat*, one for each of the stations listed in <*weather_stations*>*.dat* must be located in the <*meteo_data*> folder. An example of meteorological series file is shown in Table 3. The first row is a header row specifying the location of the weather station, the second row reports the latitude (degrees) and the altitude (m a.s.l.) of the weather station, the third row reports the initial and final dates of the time series, whereas the fourth row is the header of the meteorological table. Meteorological data required are: maximum temperature (°C), minimum temperature (°C), daily total precipitation (mm), maximum relative humidity (%), minimum relative humidity (%), daily average wind speed (m s^{-1}) and total daily solar radiation (MJ m^{-2} d^{-1}).

All the meteorological series must have the same starting and ending dates.

Table 3: Example of meteorological data file.

Station ID: 132, Located in: Stezzano						
45.64	266					
01/01/1993 -> 31/12/2014						
T_max	T_min	P_tot	U_max	U_min	V_med	RG_CORR
2.14	-5.29	0	88.77	54.96	1.38	5.15
-1.06	-9.03	0	85.9	67.96	1.37	1.53
…	…	…	…	…	…	…
7.24	2.28	0	95.47	67.96	1.41	2.21

2.4 CO_2 concentrations (optional)

The file '*CO2_conc.dat*' reports the yearly values of CO_2 concentrations (p.p.m.) as shown in Table 4.

Table 4: CO_2 concentrations file.

Year	CO2
2005	400
2006	400

2.5 Soil Uses

Soil uses are listed in the '*soil_uses.txt*' file (Table 5) that must be located in the <*crop_inputs*> folder (whose name must be declared in <*cropcoef*>*.txt*, see Table 1). The file contains information about the soil uses, including crop sequences (double cropping) within the same year. Data are organized in four columns as follows:

- a reference code for each crop sequence (Cr_ID, column 1),
- the name of the first crop in the sequence (Crop1, column 2);
- the name of the second crop in the sequence, if any, otherwise the '*' character if only one crop is grown in the year (Crop2, column 3);
- comments (starting with the symbol '#, column 4').

Names of crops (<*crop*>*.tab*) refer to the names of the files where crop parameters are listed (see § 2.6). If the second crop is missing, user must enter an asterisk ('*'), whereas input of the first crop is mandatory.

Table 5: Soil uses file.

# List of crops to be simulated, columns must be separated by one or more tab characters			
Cr_ID	Crop1	Crop2	#Comments
1	corn600.tab	*	# Corn FAO 600
2	winter_cereals.tab	*	# Wheat and barley, autumn-sown
3	bare_soil.tab	*	# Bare soil
4	corn300.tab	*	# Corn FAO 300
5	corn300.tab	winter_cereals.tab	# Corn FAO 300 + barley
endTable			

2.6 Crop parameters

Crop parameters files <*crop*>*.tab,* one for each of the different crops listed in '*soil_uses.txt*', must be located in the <*crop_parameters*> subfolder inside the <*crop_inputs*> folder. An example of the crop parameters file is shown inTable 6. Required input parameters include:

- *SowingDate_min*: minimum sowing date expressed as day of the year [1-366];
- *SowingDelay_max*: maximum number of days when sowing can be performed;
- *HarvestDate_max*: maximum harvest date expressed as day of the year [1-366];
- *HarvNum_max*: maximum number of harvests/cuts per the year;
- *CropsOverlap*: minimum number of days between two subsequent crops in case of double cropping;
- *Tsowing*: minimum sowing temperature [°C];
- *Tdaybase*: minimum temperature for crop growth [°C];
- *Tcutoff*: maximum temperature for crop growth [°C];
- *Vern*: response to vernalisation [1=Yes, 0=No];
- *Tv_min*: minimum temperature for optimal vernalisation [°C];

- *Tv_max*: maximum temperature for optimal vernalisation [°C];
- *VFmin*: vernalization factor at the beginning of the vernalisation process [-];
- *Vstart*: number of days required for vernalisation to start;
- *Vend*: number of days required for vernalisation to end;
- *Vslope*: vernalisation curve parameter;
- *ph_r*: photoperiod impact [0=Day-neutral plants, 1=Long-day plants, 2=Short-day plants];
- *daylength_if*: day length threshold below (above) which no accumulation of physiological time occurs for long-day (short-day) crops;
- *daylength_ins*: day length threshold above (below) which maximum accumulation of physiological time occurs for long-day (short-day) crops;
- *WP*: biomass water productivity [t/ha] (C4 crops = 0.30 – 0.35, C3 crops = 0.15 – 0.20, some leguminous crops < 0.15 t/ha);
- *fsink*: crop sink strength coefficient;
- *Tcrit_HS*: critical temperature threshold for heat stress [°C];
- *Tlim_HS:* limit temperature threshold for heat stress [°C];
- *HI*: harvest index [-];
- *kyT*: water stress coefficient for the overall crop growth cycle [-];
- *ky1*: water stress coefficient for the ini stage [-];
- *ky2*: water stress coefficient for the dev stage [-];

Table 6: Crop parameters file.

# mais classe 600		
SowingDate_min	= 91	# minimum sowing date (1-366)
SowingDelay_max	= 14	# maximum number of days allowed for sowing after SowingDate_min
HarvestDate_max	= 274	# maximum harvest date (1-366)
HarvNum_max	= 1	# maximum number of harvest/cuts per the year
CropsOverlap	= 7	# minimum number of days between two subsequent crops in case of double cropping
Tsowing	= 9	# minimum sowing temperature [°C]
Tdaybase	= 9	# minimum temperature for crop growth [°C]
Tcutoff	= 30	# maximum temperature for crop growth [°C]
Vern	= 0	# response to vernalization [1=Yes, 0=No]
Tv_min	= 3	# minimum temperature for optimal vernalization [°C]
Tv_max	= 10	# maximum temperature for optimal vernalization [°C]
VFmin	= 0	# vernalization factor at the beginning of the vernalization process [-]
Vstart	= 10	# number of days required for vernalization to start
Vend	= 50	# number of days required for vernalization to end
Vslope	= 7	# vernalization curve parameter
ph_r	= 0	# photoperiod impact [0=Day-neutral plants, 1=Long-day plants, 2=Short-day plants]

daylength_if	= 8	# day length threshold below (above) which no accumulation of physiological time occurs for long-day (short-day) crops
daylength_ins	= 20	# day length threshold above (below) which maximum accumulation of physiological time occurs for long-day (short-day) crops
WP	= 0.334	# biomass water productivity [t/ha] (C4 crops = 0.30 – 0.35, C3 crops = 0.15 – 0.20, some leguminous crops < 0.15 t/ha)
fsink	= 0.1	# crop sink strength coefficient
Tcrit_HS	= 32	# critical temperature threshold for heat stress [°C]
Tlim_HS	= 40	# limit temperature threshold for heat stress[°C]
HI	= 0.45	# harvest index
kyT	= 1.25	# water stress coefficient for the overall crop growth cycle
ky1	= 0.40	# water stress coefficient for the ini stage
ky2	= 0.90	# water stress coefficient for the dev stage
ky3	= 1.50	# water stress coefficient for the mid stage
ky4	= 0.50	# water stress coefficient for the end stage
pRAW	= 0.5	# parameter to compute RAW
aInterception	= 0.6	# parameter to calculate interception
cl_CN	= 2	# CN class
Irrigation	= 1	# irrigation (1 = Yes, 0 = No)

table of GDD [°C], Kcb [-], LAI [-], crop height [m], root depth [m]; missing values are entered as *

GDD	Kcb	LAI	Hc	Sr
35	0	0	0	0
40	0.15	0.05	0.02	0.3
170	0.15	0.5	0.6	0.5
650	1.15	5.2	3	0.85
1400	1.15	4.7	3	0.85
1720	0.15	3.7	2.5	0.85

endTable

- *ky3*: water stress coefficient for the mid stage [-];
- *ky4*: water stress coefficient for the end stage [-];
- *pRAW*: parameter to compute RAW [-];
- *aInterception*: parameter to calculate interception [mm];
- *Ky*: parameter of crop yield reduction [-];
- *cl_CN*: CN class (set in according to the CN class table);
- *Irrigation*: irrigation (1 = Yes, 0 = No);

In addition to these parameters, the user must provide a table with the Growing Degree Days [°C] required to reach each stage and the corresponding value of K_{cb} [-], *LAI* [m^2m^{-2}], crop height [m] and root depth [m]. Missing values must be indicated with an asterisk ('*').

3 CropCoef output files

Output files generated by CropCoef are saved in the <*OutputFolder*> declared in <*cropcoef*>*.txt* (see Table 1). The model automatically generates one folder ('*Pheno_*'<*yyy*>) for each of the weather stations considered for the simulation. Each folder contains seven <*crop-outputs*>*.dat* files with a different format depending on the type of crop output.

Output files with daily time series, as synthetised in Table 7, are: '*Kcb.dat*' (daily series of K_{cb} values), '*LAI.dat*' (daily series of LAI values), '*H.dat*' (daily series of crop height), '*Sr.dat*' (daily series of root depth) and '*CN_value.dat*' (daily series of codes accounting for crop stages used to relate *CN* values to the specific crop stage). Sample daily files are shown in Table 8 and Table 9 where the number of columns is equal to the number of crop combinations (CrID) listed in the '*soil_uses.txt*' file (Table 5). The daily series is obtained by a linear interpolation of the values, once identified the days corresponding to inflection points of the curve of each variable (first occurrence of thermal time greater than the input GDD thresholds).

Table 7: Crop parameters' files for each meteorological station.

Crop parameter series files	
Kcb.dat	Basal crop coefficient [real]
LAI.dat	Leaf Area Index [real]
H.dat	Crop height [real]
Sr.dat	Root depth [real]
CropParam.dat	Crop parameters: irrigation flag [integer], CN class [integer], p_{tab} [real], a_I [real], T_{lim} [real], T_{crit} [real], HI [real], ky_T [real], ky_1 [real], ky_2 [real], ky_3 [real] and ky_4 [real]
CNvalue.dat	Stage of growth codes to apply Curve Number method [integer]
WPadj.dat	Normalized biomass water productivity [real]

Table 8: CropCoef output of K_{cb} time series.

CrID_1	CrID_2	…	CrID_24
0.0000	0.0000	…	0.0000
…	…	…	…
1.0534	1.1039	…	0.8574
…	…	…	…

Table 9: CropCoef output of CN values. 2nd – nth lines: daily values of land use code (even columns) and of code to account for seasonal variations (0: before ploughing and after harvesting, 1: between ploughing and normal peak height, 2: between normal peak height and harvest time).

CrID_1	CrID_2	…	CrID_24
1	0		0
…	…	…	…
0	2	…	2
…	…	…	…

Two additional output files are saved in each folder named '*WPadj.dat*' and '*CropParam.dat*'. The file '*WPadj.dat*' reports yearly values of the normalized biomass water productivity corrected based on CO_2 concentrations if the variable *CanopyResMod* in the main input file is equal to 1 (Table 1), as shown in Table 10. In case of a double crop, two columns are reported for each CrID (e.g., if CrID_1 is a combination of two subsequent crops, the output file will have two columns for CrID_1 with headers equal to 'CrID1_1' and 'CrID1_2', respectively). The file '*CropParam.dat*' reports a list of input parameters that are set in the crop parameters input file (§ 2.6) and printed in output to be read by IdrAgra, as shown in Table 11. Finally, if '*CanopyResMod*' is set equal to 1 in *<cropcoef>.txt* (Table 1), an output file '*CanopyRes.dat*' with corrected value of canopy resistance is printed in the *<OutputFolder>* declared in the same *<cropcoef>.txt* file (Table 1); the contents of this output file are shown in Table 12.

Table 10: *WPadj.dat* output file.

Year	CrID_1	CrID_2	…	CrID_24
1993	0.15	0.34	…	0.18
…	…	…	…	…
2012	0.15	0.34	…	0.18
…	…	…	…	…

Table 11: *Crop_param.dat* output file.

Var	CrID_1	CrID_2	…	CrID_24
Irrig	0	1	…	1
CNclass	3	2	…	2
pRAW	0.5	0.5	…	0.4
aInt	0.6	0.6	…	2.5
Tlim	31	40	…	40
Tcrit	27	32	…	35
HI	0.45	0.57	…	0.65
kyT	1.05	1.25	…	1.05
ky1	0.2	0.4	…	0.4
ky2	0.6	0.9	…	0.4
ky3	0.75	1.5	…	1.1
ky4	0.5	0.5	…	0.4

Table 12: ***CanopyRes.dat*** **output file.**

Year	CanRes
2005	75.88
2006	75.88

4 IdrAgra input files

Input files for the IdrAgra model must be stored in the same <*Working Folder*> folder used for CropCoef.

All model inputs must be provided as either maps (grid files in ASCII raster format) or tables (in text format, with tabs or spaces as separators).

4.1 Organisation of input data

The *Working Folder* can be organised as preferred; the only mandatory files that have to be in the root are the executable code ('*IdrAgra.exe*'), the weather station input file (<*weather_stations*>*.dat*) and a text file (<*IdrAgra_parameters*>*.txt*), containing the simulation control parameters, that is used when the code is run in "-f" mode (see § 4.8).

The following structure is strongly recommended for the remaining input files (Figure 4):

- all base maps are stored in the "<*spatial_data*>" folder;
- crop growth tables, as generated by *CropCoef* module, are stored in the "<*crop_series*>" folder;
- meteorological time series are stored in the "<*meteo_data*>" folder, as described in § 2.2 and §2.3);
- irrigation methods tables are stored in the "<*irrmeth_data*>" folder;
- irrigation sources information are stored in the "<*watsour_data*>" folder;
- water content initial condition maps (optional) are stored in the "<*initial_condition*>" folder.

The directories' names are customizable to give the possibility to better organise the data and to accommodate more datasets in the same root working folder.

The output folder <*sim_results*> and the optional water content final condition folder <*final_condition*> will be created inside the working root folder.

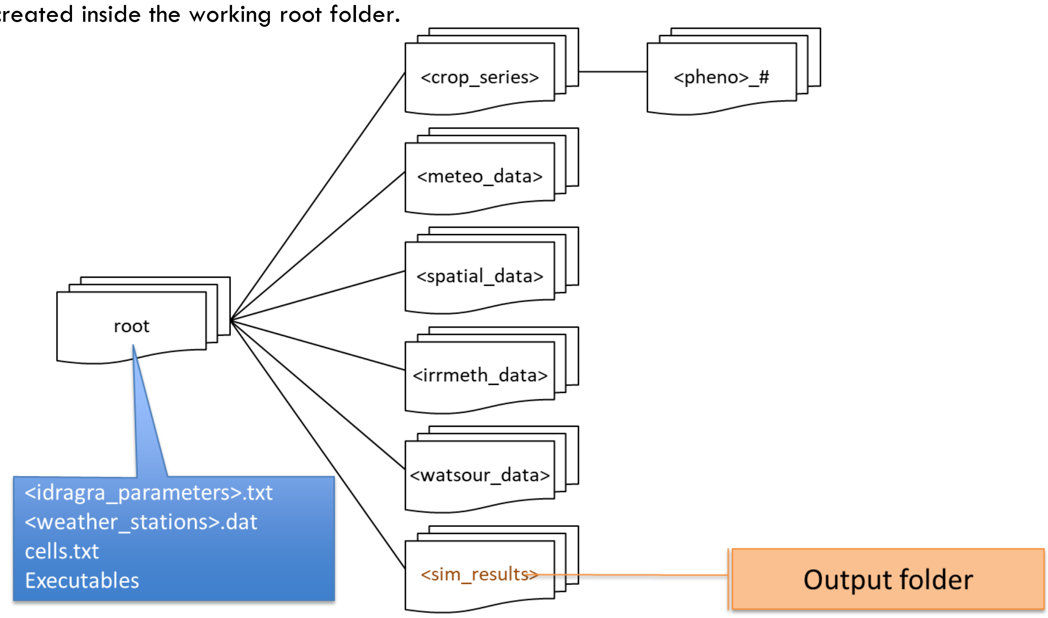

Figure 4: Dataset structure for IdrAgra

4.2 Simulation control parameters

The simulation parameters file *<IdrAgra_parameters>.txt*, whose name is customizable (e.g., *IdrAgra_parameters_BlueRiver_20210716.txt*), has to be provided in the root folder to run the program. An example of the file contents is reported in Table 13; the comment lines are included to help understanding the meaning and scope of the different simulation parameters.

Table 13: Structure of the simulation parameters file *<IdrAgra_parameters>.txt*: grey lines (lines that start with #) are explanatory comments.

Input file for IdrAgra
Note: lines starting with <#> are comments
1. General section
# 1.1. Input and output folders and files
# 1.1.1. Output folder
# OutputPath: path to output folder [name_outputpath\\]
OutputPath = output_folder\\
# 1.1.2. Input folders
# InputPath: path to spatialized input folder (default: spatial_data\\)
InputPath = spatial_data\\
# MeteoPath: path to meteorological stations folder (default: meteo_data\\)
MeteoPath = meteo_data\\
# MeteoFileName: file, located in root folder, in which meteorological stations' filenames are stored (default: weather_stations.dat)
MeteoFileName = weather_stations.dat
# PhenoPath: path to phenological parameters folder (default: crop_series\\)
PhenoPath = crop_series\\
# PhenoFileRoot: first part of the name of phenological parameters subfolders (labelled as PhenoFileRoot_MeteoNum.dat) (default: pheno_)
PhenoFileRoot = pheno_
# IrrMethPath: path to irrigation methods folder (default: irrmeth_data\\)
IrrMethPath = irrmeth_data\\
# IrrMethFileName: file, located in irrigation methods folder, in which irrigation methods filenames are stored (default: irrmethods.txt)
IrrMethFileName = irrmethods.txt
# WatSourPath: path to water sources folder (default: watsour_data\\)
WatSourPath = watsour_data\\
# 1.2. Simulation settings
# 1.2.1. Type of simulation (default: 2)
# Mode: type of simulation [0...4]
# Mode = 0 # simulation without irrigation
# Mode = 1 # simulation with irrigation, mode consumptions
# Mode = 2 # simulation with irrigation, field capacity needs satisfaction
# Mode = 3 # simulation with irrigation, fixed volumes
# Mode = 4 # fixed irrigation applications, data and volumes are specified in a file
Mode = 1
# 1.2.2. Simulation conditions
# InitialThetaFlag: switch, external setting of initial soil moisture condition [T F] (T=true, F= false)
# InitialThetaFlag = T # initial moisture condition read from external file

```
# InitialThetaFlag = F # internally generated initial moisture condition by running the first year and using its output as initial moisture for the simulation
InitialThetaFlag = T
# InitialConditionPath: path to initial condition folder (default: spatial_data\\)
InitialConditionPath = initial_condition\\
# InitialCondition: root of initial condition files (default: IC_theta)
# (Default: IC_thetaI.asc, IC_thetaII.asc)
InitialCondition = IC_theta
# FinalConditionPath: path to final condition folder (default: sim_results\\)
FinalConditionPath = final_condition\\
# FinalCondition: root of final condition files (default: FC_theta)
# (Default: FC_thetaI.asc, FC_thetaII.asc)
FinalCondition = FC_theta
# CapillaryFlag: switch, simulation of capillary rise [T F] (T=true, F= false)
# CapillaryFlag = T # capillary rise simulated
# CapillaryFlag = F # capillary rise not simulated
CapillaryFlag = T
# SoilUseVarFlag: switch, simulation uses yearly soil uses [T F] (T=true, F= false)
# SoilUseVarFlag = T # soil uses changes every year
# SoilUseVarFlag = F # soil uses does not change
SoilUseVarFlag = F
# 1.2.3. Meteorological inputs
# MeteoStatTotNum: number of meteorological stations (default: 1)
MeteoStatTotNum = 5
# MeteoStatWeightNum: number of nearest meteorological stations used in the spatial interpolation of crop phenology (default: 1)
MeteoStatWeightNum = 5
# 1.2.4. Soil uses inputs
# SoilUsesNum: number of considered soil uses in each of phenological series
SoilUsesNum = 22
# RandSowDaysWind: number of days of window for sowing date randomization
RandSowDaysWind = 6
# 1.2.5. Periodical output setting
# MonthlyFlag: output interval (monthly, weekly, or specific interval) (default: monthly)
# MonthlyFlag = monthly | MonthlyFlag = month | MonthlyFlag = m | MonthlyFlag = T # switch, output each month
# MonthlyFlag = weekly | MonthlyFlag = week | MonthlyFlag = w # switch, output each week
# MonthlyFlag = periodic | MonthlyFlag = p | MonthlyFlag = F # switch, periodical output
MonthlyFlag = monthly
# if MonthlyFlag = weekly, choose day of the week - output will be recorded on that day with reference to past week
# Weekday = Monday | Weekday = Mon | Weekday = 1
# Weekday = Tuesday | Weekday = Tue | Weekday = 2
# Weekday = Wednesday | Weekday = Wed | Weekday = 3
# Weekday = Thursday | Weekday = Thu | Weekday = 4
# Weekday = Friday | Weekday = Fri | Weekday = 5
# Weekday = Saturday | Weekday = Sat | Weekday = 6
# Weekday = Sunday | Weekday = Sun | Weekday = 7
Weekday = Saturday
# if MonthlyFlag = periodic, choose output interval - output will be recorded from StartDate to EndDate every DeltaDate days
```

```
# StartDate: start Julian day [1...366] for periodic output (default: 10)
StartDate = 1
# EndDate: end Julian day [1...366] for periodic output (default: 100)
EndDate = 365
# DeltaDate: output interval, Julian days [1... 366] (default: 30)
DeltaDate = 365
### 2. Simulation specifications ###
# 2.1. Soil conductivity parameters
# 01q_eva: 10th percentile of soil conductivity for evaporative layer
01q_eva = 2.99
# 09q_eva: 90th percentile of soil conductivity for evaporative layer
09q_eva = 20.43
# 01q_trasp: 10th percentile of soil conductivity for traspirative layer
01q_trasp = 2.09
# 09q_eva: 90th percentile of soil conductivity for traspirative layer
09q_eva = 39.66
# 2.2. Irrigation inputs
# StartIrrSeason: Julian day [1...366] in which irrigation season starts (default: 91)
StartIrrSeason = 91
# EndIrrSeason: Julian day [1...366] in which irrigation season starts (default: 304)
EndIrrSeason = 304
# 2.3. Layers depth [m]
# zEvap: evaporative layer depth [m]
zEvap = 0.10
# zRoot: transpirative layer depth [m]
zRoot = 0.90
# 2.4. Curve Number parameters
# LambdaCN: Initial Abstraction ratio (Ia/S) (default: 0.2)
LambdaCN = 0.2
### 3. DTx specifications ###
# DTxMode: DTx calculation off [none], DTx statistical analysis [analysis] or DTx application [application]
# DTxMode = none
# DTxMode = analysis
# DTxMode = application
DTxMode = analysis
# DTxNumXs: number of calculated indices (one for each integration period), i.e. elements of DTx_x
DTxNumXs = 3
# DTx_x: integration period (DT10 sums transpirative deficit of 10 days)
DTx_x = 10 20 30
# DTxDeltaDate: DTx calculation interval
DTxDeltaDate = 10 #
# DTxDelayDays: delay from the first day of year to start calculation
DTxDelayDays = 1
# if DTxMode = analysis, choose minimum cardinality (i.e. number of elements for a valid estimate)
# DTxMinCard: minimum cardinality for statistical analysis
DTxMinCard = 3
```

4.3 Maps

All maps, stored in the <*spatial_data*> folder, that are needed to run the model are listed in Table 14. An example of their structure is reported in Table 16.

Table 14: IdrAgra input maps.

General	
Default name	**Description**
domain.asc	Mask map that defines model boundaries. Mask cells are defined with the code 1 [integer]
Topography	
slope.asc	Slope gradient [real] ($m \cdot m^{-1}$)
Land	
soiluse.asc soiluse_*yyyy*.asc	Map with land use classes [integer] Land use classes map can be substituted by yearly maps with land use classes [integer]. In this case, it should be provided one map for each year *yyyy* of simulation (e.g. *soiluse_2014.asc* for year 2014)
hydr_cond.asc	Hydrologic condition to apply CN method, that indicates the effects of cover type and treatment on infiltration and runoff (generally assumed equal to 1 for cropland) [integer] (adimensional)
Collective runtime	
Ksat_I.asc	Saturated hydraulic conductivity of the evaporative layer ($K_{s,E}$)[real] ($cm \cdot h^{-1}$)
Ksat_II.asc	Saturated hydraulic conductivity of the transpirative layer ($K_{s,T}$) [real] ($cm \cdot h^{-1}$)
N_I.asc	Brooks-Corey exponent for the evaporative layer (n_E) [real] (adimensional)
N_II.asc	Brooks-Corey exponent for the transpirative layer (n_T) [real] (adimensional)
CapRisePar_a3.asc	Capillary rise parameter a_3[real] (adimensional)
CapRisePar_a4.asc	Capillary rise parameter a_4[real] (adimensional)
CapRisePar_b1.asc	Capillary rise parameter b_1[real] (adimensional)
CapRisePar_b2.asc	Capillary rise parameter b_2[real] (adimensional)
CapRisePar_b3.asc	Capillary rise parameter b_3[real] (adimensional)
CapRisePar_b4.asc	Capillary rise parameter b_4[real] (adimensional)
ThetaI_r.asc	Residual soil water content of the evaporative layer ($\theta_{r,E}$) [real] ($m^3 \cdot m^{-3}$)
ThetaI_wp.asc	Soil water content of the evaporative layer at wilting point ($\theta_{wp,E}$) [real] ($m^3 \cdot m^{-3}$)
ThetaI_fc.asc	Soil water content of the evaporative layer at field capacity ($\theta_{fc,E}$) [real] ($m^3 \cdot m^{-3}$)
ThetaI_sat.asc	Saturated soil water content of the evaporative layer ($\theta_{s,T}$) [real] ($m^3 \cdot m^{-3}$)

Thetall_r.asc	Residual soil water content of the transpirative layer ($\theta_{r,T}$) [real] ($m^3 \cdot m^{-3}$)
Thetall_wp.asc	Soil water content of the transpirative layer at wilting point ($\theta_{wp,T}$) [real] ($m^3 \cdot m^{-3}$)
Thetall_fc.asc	Soil water content of the transpirative layer at field capacity ($\theta_{fc,T}$) [real] ($m^3 \cdot m^{-3}$)
Thetall_sat.asc	Saturated soil water content of the transpirative layer ($\theta_{s,T}$) [real] ($m^3 \cdot m^{-3}$)
hydr_group.asc	Hydrologic soil group classification to apply CN method. The correspondence is A=1, B=2, C=3 and D=4 [integer] (adimensional)
waterdepth.asc	groundwater table level [real] (m)
Meteorological spatialization	
Meteo_*n*.asc	Weighting meteorological data parameters [real] (adimensional). A number of weighting meteorological data parameters maps. It should be provided the same number of weighting meteorological data parameters maps as stated in *MeteoStatWeightNum* variable in the <IdrAgra_parameters>.txt file, if provided. Elsewhere, only one map named *Meteo_1.asc*, should be provided.
Irrigation districts information	
irr_units.asc	Irrigation units identification codes [integer]
irr_meth.asc irr_meth_*yyyy*.asc	Irrigation methods codes [integer] Irrigation methods codes map can be substituted by yearly maps of irrigation method codes [integer]. In this case, it should be provided one map for each year *yyyy* of simulation (e.g. *irr_meth_2014.asc* for year 2014)
appl_eff.asc appl_eff_*yyyy*.asc	Field application irrigation efficiency [real] $[0-1]$ (adimensional) Field application irrigation efficiency map can be substituted by yearly maps of field application irrigation efficiency [real] $[0-1]$. In this case, it should be provided one map for each year *yyyy* of simulation (e.g. *appl_eff_2014.asc* for year 2014)
conv_eff.asc	Conveyance and distribution efficiency [real] $[0-1]$ (adimensional)

If maps of the water content initial condition are provided to the model, they must be stored in the <*initial_condition*> folder and are named according to Table 15.

Table 15: Initial condition input maps.

Default name	**Description**
IC_thetal.asc	Initial soil water content of the evaporative layer (θ_E) [real] ($m^3 \cdot m^{-3}$)
IC_thetall.asc	Initial soil water content of the transpirative layer (θ_T) [real] ($m^3 \cdot m^{-3}$)

Table 16: Map (*.asc) structure. Ncols: number of columns, nrows: number of rows, xllcorner: longitude of the low left corner [m], yllcorner: latitude of the low left corner [m], cellsize: cell size [m], NODATA_value: no data value.

ncols	46				
nrows	30				
xllcorner	528482.668863637140				
yllcorner	5018365.885824285448				
cellsize	250				
NODATA_value	-9999				
-9999.00000	-9999.00000	-9999.00000	0.21159	0.23475	…
0.19327	0.19327	0.23475	0.23475	0.20835	…
--9999.00000	0.23475	0.23475	0.20835	0.20835	…
-9999.00000	0.23475	0.20835	0.20835	0.23475	…
0.25273	-9999.00000	0.23475	0.20835	0.23475	…
0.20835	0.25273	--9999.00000	0.23475	0.20835	…
0.20835	0.20835	0.23475	0.23475	0.20835	…
…	…	…	…	…	…

All maps must have identical location attributes (number of rows, columns, etc.). IdrAgra needs to know the size of each grid cell to calculate water volumes equivalent to rainfall depth values given in input. IdrAgra obtains this information from the input parameters file *<IdrAgra_parameters>.txt*. This will only work if all maps area in an equi-areal projection, and the map coordinates and cell size are defined in meters.

4.3.1 **Role of mask map**

The mask map (i.e. "*domain.asc*") defines the model domain: cells that belong to the study area have a value of 1, cells outside the study area 0. In order to avoid unexpected results, it is vital that all maps that are related to topography, land use, soil, irrigation methods and meteorological distribution are defined (i.e., do not contain any missing value) for each pixel that is "true" (has a value equal to 1) in *domain.asc*. Undefined pixels will lead to missing values in output.

4.3.2 **Optional map stacks**

Land use and irrigation method can be defined either as static maps (i.e., using the same land use and irrigation method for each year of simulation) or as a map stack. A map stack is a series of maps, where each map represents the value of a variable (e.g., soil use) during a given year. This allows using different land use, or irrigation methods for each year of simulation. The name of each map is made up combining the default name with the year of simulation (e.g., *soiluse_2014.asc*).

The map stacks that are needed to run *IdrAgra* are listed in the Table 17.

Table 17: IdrAgra map stacks.

Map stacks	
Land	
soiluse_*yyyy*.asc	Yearly maps with land use classes [integer]. It should be provided one map for each year *yyyy* of simulation (e.g. *soiluse_2014.asc* for year 2014).
Irrigation methods	
irr_meth_*yyyy*.asc	Yearly maps of irrigation methods [integer]. It should be provided one map for each year yyyy of simulation (e.g. *irr_meth_2014.asc* for year 2014).
meth_eff_*yyyy*.asc	Yearly maps with field application irrigation efficiency [real] $[0-1]$. It should be provided one map for each year yyyy of simulation (e.g. *meth_eff_2014.asc* for year 2014).

4.4 Phenological input files

Phenological inputs are read directly from the *CropCoef* output files, as described in § 3.

4.5 Irrigation methods tables

Irrigation methods tables are stored in the default folder *irrmeth_data*\\. Irrigation methods must be listed in '*irrmethods.txt*', and the characteristics of each of them must be described in a separate file, which is named <*method_name*>.txt. Examples of '*irrmethods.txt*' and <*method_name*>*.txt* structure are reported in Table 18 and Table 19.

Table 18: Irrigation method list file ('*irrmethods.txt*') structure: grey lines (lines that start with #) are comments. Irrigation methods list must contain the stated number of irrigation methods.

```
# IrrMethNum: number of considered irrigation methods
IrrMethNum = 5
# List of irrigation methods parameters' files
# List starts with the label "List =" and ends with the label "EndList ="
List =
surface_irrigation_Q178m.txt
…
micro_irrigation.txt
EndList =
```

Table 19: Irrigation method file (<*method_name*>*.txt*) structure: grey lines (lines that start with #) are comments.

Irrigation method: sprinkler irrigation
Id: irrigation method code
Id = 3
# Qadaq: irrigation water depth [mm]
Qadaq = 40
# K_stress: water stress coefficient for the activation of irrigation from water diversions
K_stress = 0.7
# K_stresswells: water stress coefficient for the activation of irrigation from private wells
K_stresswells= 0.99
# fw: exposed and wetted soil fraction
fw = 1.0
# Min_a, Max_a, Min_b & Max_b: parameters of percolation model
Min_a = 0
Max_a = 0
Min_b = 0
Max_b = 0
# a, b, c: parameters of irrigation losses model
# Irrigation losses model is described by the equation:
# Irrigation_losses = a + b * wind_speed + c * average_temperature
# If irrigation losses do not depend from wind speed or average temperature, set:
# a = fixed irrigation loss (expressed as a percentage between 0 & 100); b = 0; c = 0
a = -2.1
b = 1.91
c =0.231
# InterceptionFlag: irrigation water is intercepted by foliage?
# InterceptionFlag = TRUE or T if irrigation water is interception by foliage
# InterceptionFlag = FALSE or F if irrigation water is not interception by foliage
InterceptionFlag = F
# Irrigation run time
1 = 0.000   # Irrigation between 0:00 and 0:59
…
12 = 0.333   # Irrigation between 11:00 and 11:59
13 = 0.333   # Irrigation between 12:00 and 12:59
14 = 0.334   # Irrigation between 13:00 and 13:59
…
24 = 0.000   # Irrigation between 23:00 and 23:59

4.6 Diversions tables

Diversions tables are stored in the default folder *watsour_data*\, that contains water sources daily flow series and their distribution for each irrigation unit. The file '*watsources.txt*' contains the ratio of the total flow of each water source that is supplied to each Irrigation Unit. An example of '*watsources.txt*' contents is reported in Table 20.

Table 20: Irrigation water distribution file (*watsources.txt*) structure. 1st column: subdomain code (that identify each irrigation unit, as read in *irr_units.asc*), 2nd column: source code (that relates the irrigation unit with a source listed in 'irr_units'.txt), 3rd column: source type (1. Monitored sources I, 2. Monitored sources II, 3. Internal reuse, 4. Runtime collective), 4th column: flow ratio from source to subdistrict.

IU_ID	SOURCE_CODE	SOURCE_TYPE	FLOW_RATIO
101	m_01119	1	0.130769
102	m_01119	1	0.115384
103	m_01119	1	0.09615
….			
204	m_01117	1	0.177633

In the case of **Monitored sources,** daily timeseries of withdrawals must be provided for each of the and for the shole simulation period. The series of monitored sources must be stored in the file '*monit_sources_i.txt*'. If, for any reason, the user prefers to separate two groups of sources, a second file, '*monit_sources_ii.txt*' can be used for this purpose. The file '*int_reuse.txt*' contains the series of daily water volumes that are reused within the study area. All these files have an identical format, as reported in Table 21.

Table 21: Surface water sources (*monit_sources_i.txt* and *monit_sources_ii.txt*), and internal reuse (*int_reuse.txt*) daily flow series structure. 1st line: sources code, 2nd line: nominal flow ($m^3 \cdot s^{-1}$) for the source, 3rd line: starting date (dd/mm/yyyy) and ending date (dd/mm/yyyy), 4th – nth lines: daily flow series ($m^3 \cdot s^{-1}$).

m_01119	m_01117
3.7712	1.341
01/01/2005 -> 31/12/2006	
0	0
…	…
0.878571428	0.291666667
0.877619048	0.291388889
…	…

Collective runtime sources must be listed in *cr_sources.txt*, and each source is parameterized in a separate file *<CRS_name>.txt.*, one for each source. Examples of '*cr_sources.txt*' and *<CRS_name>.txt* structure are reported in Table 22 and Table 23.

Table 22: Collective runtime sources list file (*cr_sources.txt*) structure: grey lines (lines that start with #) are comments

# CRS_TotNum: number of collective runtime sources
CRS_TotNum = 1
# List: CRS parameters' files
# List starts with the label "List =" and ends with the label "EndList ="
List =
crs1.txt
EndList =

Table 23: Collective runtime sources parameters file (<*CRS_name* >.txt) structure.

# SourceAcronym: collective runtime source name, that link the source to its irrigation unit(s)		
SourceAcronym = crs1		
# Qmax: maximum flow rate		
Qmax = 4.640		
# Qnom: nominal flow rate		
Qnom = 1.350		
# ActThrS: minimum activation threshold		
ActThrS = 0.9		
# Table: flow rate ratio (with respect to nominal flow rate) activation for each activation threshold		
Table =		
Id	Activation threshold	Flow rate ratio
1	0.9	0.5
2	0.8	0.7
3	0.5	0.9
EndTable =		

4.7 Selection of individual cells

Saving the daily input and output variables for all the cells could require huge storage space if the model domain is large; therefore, IdrAgra normally saves the variables at longer intervals (e.g., weekly or monthly). However, in this case IdrAgra allows saving the whole set of daily input and output variables for a user-selected number of individual cells, to allow analysing and representing the dynamics of storages and fluxes in detail. The coordinates of the cells of interest must be reported in the file '*cells.txt*', whose structure is shown in Table 24**Errore. L'origine riferimento non è stata trovata.**.

Table 24: Example of the content of the '*cells.txt*' file.

ncells = 3		
table =		
id	column	row
1	20	60
5	21	60
8	40	10
endtable =		

IdrAgra will then generate an output file '*aaaa_cellinfo_xx_yy.csv*' for each of the cells in '*cells.txt*'; *x* and *y* in the output file name indicate the cell position. The file '*aaaa_parameters_xx_yy.csv*' contains the list of values of all the crop parameters read in input for the cell and the complete list of values of all the daily output variables.

4.8 IdrAgra options

The options that can be selected are as follows:

- *-h, -help* prints the help on the screen, which lists the launch modes, and closes the program

- *-d, - default* print default parameters on screen, and close the program
- *-p, -preview* prints the parameters read on the screen before starting the program
- *-v, -verbose* prints more output, both at basin scale and on selected cells (with more computation time)
- *-s, -summary* prints only the total irrigation need / use at the reference time step (monthly or periodic) and annually only the irrigation output
- *-t, -teta, -theta* prints the humidity outputs of the first and second layer in a * .asc file at the end of each simulation year
- *- f <parameters>, -filename <parameters>* read the parameters from the input file <parameters> that can be different from the default file IdrAgra_parameters.txt

A combination of options can also be used: for example, to run a simulation in verbose mode with a configuration defined in a file that we assume is named '*modified_parameters.txt*' - the following command string must be used: *IdrAgra_20190130.exe -v -f modified_parameters.txt*.

If the output folder indicated in *<IdrAgra_parameters>.txt* is already present, IdrAgra will ask if you want to overwrite the folder: if you press enter the files will be overwritten, otherwise, it is necessary to rename the output sub-folder in *<IdrAgra_parameters>.txt*.

5 IdrAgra outputs

5.1 IdrAgra output maps

The model generates different maps, that can be broadly divided into:

- yearly maps: a map for each year *y* of simulation, with the name *<yyyy>_<map name>.asc* (e.g. *1993_biomass_pot.asc is 1993 potential biomass map.asc*);
- periodic output map: a map for each period of integration:
  - if the values are cumulated over a week, the maps are named as *<yyyy>_week<n>_<map name>.asc* (e.g. *1993_week1_et_act.asc* is the actual evapotranspiration map for the 1st week of 1993);
  - if the values are cumulated over a month, the maps are named as *<yyyy>_month<n>_<map name>.asc* (e.g. *1993_month7_et_asc.asc* is the actual evapotranspiration map for July 1993);
  - finally, if the values are cumulated over a different time step, the maps are named as *<yyyy>_step<n>_<map name>.asc* (e.g. *1993_step1_et_act.asc* is the actual evapotranspiration map for the 1st integration period of 1993).

Output map structure is completely analogous to the one of input maps (see Table 16).

5.1.1 Yearly output maps

Yearly maps that are generated are listed in Table 25. Yearly maps that are generated only in debug mode (executing the code in "-v" mode) are listed in Table 26. To verify that input maps are read correctly, a copy of input maps is also generated, named *out_<maps>.asc*.

Table 25: IdrAgra output yearly maps – standard mode.

Default name	Description
Yearly maps – standard mode	
<yyyy>_biomass_pot_<c>.asc	Yearly maps of potential biomass [real]. A map for each year *y* of simulation and each *c* crop (e.g. 1993_biomass_pot_1.asc) is generated.
<yyyy>_eff_tot.asc	Yearly maps of irrigation efficiency [real]. A map for each year *y* of simulation is generated.
<yyyy>_eva_act_agr.asc	Yearly maps of cumulative actual evapotranspiration for crop season [real]. A map for each year *y* of simulation is generated.

<yyyy>_eva_pot_agr.asc	Yearly maps of cumulative potential evapotranspiration for crop season [real]. A map for each year *y* of simulation is generated.
<yyyy>_flux_tot.asc	Yearly maps of net flux from the transpirative layer to groundwater (percolation – capillary rise) [real]. A map for each year *y* of simulation is generated.
<yyyy>_irr_loss.asc	Yearly maps of irrigation application losses [real]. A map for each year *y* of simulation is generated.
<yyyy>_irr_mean.asc	Yearly maps of mean irrigation application [real]. A map for each year *y* of simulation is generated.
<yyyy>_irr_nr.asc	Yearly maps of irrigation application number [integer]. A map for each year *y* of simulation is generated.
<yyyy>_irr_tot.asc	Yearly maps of cumulative irrigation [real]. A map for each year *y* of simulation is generated.
<yyyy>_prec_tot.asc	Yearly maps of cumulative precipitation [real]. A map for each year *y* of simulation is generated.
<yyyy>_run_tot.asc	Yearly maps of cumulative runoff [real]. A map for each year *y* of simulation is generated.
<yyyy>_trasp_act_tot.asc	Yearly maps of cumulative actual transpiration for crop season [real]. A map for each year *y* of simulation is generated.
<yyyy>_trasp_pot_tot.asc	Yearly maps of cumulative potential transpiration for crop season [real]. A map for each year *y* of simulation is generated.
<yyyy>_yield_act_<c>.asc	Yearly maps of actual yield [real]. A map for each year *y* of simulation and each c crop is generated.
<yyyy>_yield_pot_<c>.asc	Yearly maps of potential yield [real]. A map for each year *y* of simulation and each c crop is generated.

Table 26: IdrAgra output yearly maps – debug mode.

Default name	Description
Yearly maps – debug mode	
<yyyy>_dij_<c>.asc	Yearly maps of growing period [real]. A map for each year *y* of simulation and each *c* crop is generated.
<yyyy>_eff_prec_tot.asc	Yearly maps of precipitation efficiency [real]. A map for each year *y* of simulation is generated.
<yyyy>_eva_tot.asc	Yearly maps of cumulative actual evaporation [real]. A map for each year *y* of simulation is generated.
<yyyy>_fcCS_<c>.asc	Yearly maps of water-stress yield reduction factor weighted between the four crop stages [real]. A map for each year *y* of simulation and each *c* crop is generated.
<yyyy>_fcT_<c>.asc	Yearly maps of overall water-stress yield reduction factor [real]. A map for each year *y* of simulation and each *c* crop is generated.
<yyyy>_fHS_<c>.asc	Yearly maps of heat-stress yield reduction factor [real]. A map for each year *y* of simulation and each *c* crop is generated.
<yyyy>_ii0_<c>.asc	Yearly maps of emergence date [integer]. A map for each year *y* of simulation and each *c* crop is generated.
<yyyy>_iie_<c>.asc	Yearly maps of harvesting date [integer]. A map for each year *y* of simulation and each *c* crop is generated.
<yyyy>_irandom.asc	Yearly maps of growing period shift [integer]. A map for each year *y* of simulation is generated.
<yyyy>_iter1.asc	Yearly maps of maximum number of iterations for the evaporative layer [integer]. A map for each year *y* of simulation is generated.
<yyyy>_iter2.asc	Yearly maps of maximum number of iterations for the transpirative layer [integer]. A map for each year *y* of simulation is generated.
<yyyy>_prec_agr.asc	Yearly maps of cumulative precipitation for crop season [real]. A map for each year *y* of simulation is generated.

<yyyy>_T_act_sum_<s>.asc	Yearly maps of actual transpiration for s^{th} crop stage [real]. A map for each year *y* of simulation and each s^{th} crop stage is generated.
<yyyy>_T_pot_sum_<s>.asc	Yearly maps of potential transpiration for s^{th} crop stage [real]. A map for each year *y* of simulation and each s^{th} crop stage is generated.

5.1.2 Periodic output maps

Periodic maps that are generated are listed in Table 27. Periodic maps that are generated only in debug mode are listed in Table 28.

Table 27: IdrAgra output periodic maps – standard mode. "Step" is substituted by "week" for weekly outputs and by "month" for monthly outputs.

Default name	**Description**
Periodic maps – standard mode	
<yyyy>_step<n>_caprise.asc	Periodic maps of cumulative capillary rise from groundwater to the transpirative layer [real]. A map for each output interval – set in # 1.2.5 of simulation parameters file – *n* of each year *y* of simulation (e.g. 1993_step1_caprise.asc for the 1^{st} output interval of 1993) is generated
<yyyy>_step<n>_et_act.asc	Periodic maps of cumulative actual evapotranspiration [real]. A map for each output interval *n* of each year *y* is generated.
<yyyy>_step<n>_et_pot.asc	Periodic maps of cumulative potential evapotranspiration [real]. A map for each output interval *n* of each year *y* is generated.
<yyyy>_step<n>_flux2.asc	Periodic maps of cumulative net flux from the transpirative layer to groundwater (percolation – capillary rise) [real]. A map for each output interval *n* of each year *y* is generated.
<yyyy>_step<n>_irr.asc	Periodic maps of cumulative irrigation [real]. A map for each output interval *n* of each year *y* is generated.

<yyyy>_step<n>_irr_units.asc	Periodic maps of cumulative net irrigation from district's water supply [real]. A map for each output interval *n* of each year *y* is generated.
<yyyy>_step<n>_irr_loss.asc	Periodic maps of irrigation application loss [real]. A map for each output interval *n* of each year *y* is generated.
<yyyy>_step<n>_irr_privw.asc	Periodic maps of cumulative irrigation from private runtime sources [real]. A map for each output interval *n* of each year *y* is generated.
<yyyy>_step<n>_prec.asc	Periodic maps of cumulative precipitation [real]. A map for each output interval *n* of each year *y* is generated.
<yyyy>_step<n>_runoff.asc	Periodic maps of cumulative runoff [real]. A map for each output interval *n* of each year *y* is generated.
<yyyy>_step<n>_trasp_act.asc	Periodic maps of cumulative actual transpiration [real]. A map for each output interval *n* of each year *y* is generated.
<yyyy>_step<n>_trasp_pot.asc	Periodic maps of cumulative potential transpiration [real]. A map for each output interval *n* of each year *y* is generated.

Table 28: IdrAgra output periodic maps – debug mode. "Step" is substituted by "week" for weekly outputs and by "month" for monthly outputs.

Default name	**Description**
Periodic maps – debug mode	
<yyyy>_step<n>_eva.asc	Periodic maps of cumulative actual evaporation [real]. A map for each output interval *n* of each year *y* is generated.
<yyyy>_step<n>_perc1.asc	Periodic maps of cumulative percolation from the evaporative layer to the transpirative layer [real]. A map for each output interval *n* of each year *y* is generated.
<yyyy>_step<n>_perc2.asc	Periodic maps of cumulative percolation from the transpirative layer to groundwater [real]. A map for each output interval *n* of each year *y* is generated.

<yyyy>_step<n>_prec_eff.asc	Periodic maps of cumulative effective precipitation [real]. A map for each output interval *n* of each year *y* is generated.
<yyyy>_step<n>_theta1.asc	Periodic maps of soil water content of the evaporative layer at the end of the time step [real]. A map for each output interval *n* of each year *y* is generated.
<yyyy>_step<n>_theta2.asc	Periodic maps of soil water content of the transpirative layer at the end of the time step [real]. A map for each output interval *n* of each year *y* is generated.

5.1.3 Transpiration deficit output maps

Transpiration deficit maps that are generated are listed in Table 29.

Table 29 IdrAgra transpiration deficit maps.

Default name	**Description**
TD distribution	
dtx_alpha_*d*.asc	α parameter of TD distribution for the integration period *x*, calculated for the d^{th} x-day period of the year
dtx_beta_*d*.asc	β parameter of TD distribution for the integration period *x*, calculated for the d^{th} x-day period of the year
dtx_zero_prob_*d*.asc	Probability of zero of TD distribution for the integration period *x*, calculated for the d^{th} x-day period of the year

5.2 IdrAgra output tables

IdrAgra generates a series of tab separated values files (*.csv) that can be imported and analysed into a spreadsheet. These files fall into two categories:

- files that describe irrigation unit daily series, that are generated only in “use” mode (Table 30); files that help analysing the behaviour of an individual cell on a day-to-day basis (Table 35).
- other tab separated values files that are generated in debug mode, to verify that meteorological and phenological input are correctly processed (Table 39).

Table 30: Use mode tab separated values files output.

Default name	**Description**
Use mode output tables – Standard mode	
<yyyy>_Qirr.csv	Daily series of irrigation water net supply to each irrigation unit for each year *y* [real].
<yyyy>_Qprivatewells. csv	Daily series of irrigation water supply from private runtime sources for each irrigation unit for each year *y* [real].
<yyyy>_Qcrs. csv	Daily series of irrigation water supply from collective runtime sources for each irrigation unit for each year *y* [real].
<yyyy>_Qrem. csv	Daily series of irrigation water supply to each irrigation unit for each year *y* that are not used in day *i* and are transferred to day *i+1* [real].
<yyyy>_Qirrunits. csv	Daily series of irrigation water gross supply to each irrigation unit for each year *y* [real].
<yyyy>_Qsurplus. csv	Daily series of irrigation water discharge for each irrigation unit for each year *y* [real].
<yyyy>_Watshift. csv	Daily series of the number of cells checked for irrigation for each year *y* [integer].

Daily series of irrigation water gross supply to each irrigation unit during irrigation season are listed in *<yyyy>_Qirrunits.csv* (Table 31). The individual contribution of each type of water source is then listed respectively in *<yyyy>_Qcrs.csv* (Table 32) for collective runtime sources and in *<yyyy>_Qprivate.csv* for private runtime sources, if activated.

Daily series of irrigation water net supply and water discharges, with the same file structure, are then listed in *<yyyy>_Qirr.csv* (Table 33) and in *<yyyy>_Qsurplus.csv* (Table 34).

Table 31: Daily series of irrigation water gross supply to each irrigation unit (*<yyyy>_Qirrunits.csv*) output. The daily series starts and ends according to StartIrrSeason and EndIrrSeason input.

DoY	Source	Source_1	Source_2	…	Source_n
105	Montored sources (i)	3.161	0	…	0
105	Montored sources (ii)	0	0		0
105	Internal reuse	0	0.703006		1.687351
105	Collective runtime sources	0	0		0.164987
…		…	…	…	…
273	Collective runtime sources	0	0	…	0.164987

Table 32: Daily series of irrigation water supply from collective runtime sources (*<yyyy>_Qcrs.csv*) output. The daily series starts and ends according to StartIrrSeason and EndIrrSeason input.

DoY	Source_1	Source_2	…	Source_n
105	4.641136	0.35	…	0.1725
106	4.641136	0.35	…	0.1725
…	…	…	…	…
272	4.641136	0.35	…	0.1725
273	4.641136	0.35	…	0.1725

Table 33: Daily series of irrigation water net supply to each irrigation unit (*<yyyy>_Qirr.csv*) output. The daily series starts and ends according to StartIrrSeason and EndIrrSeason input.

DoY	SubDistr_1	SubDistr_2	…	SubDistr_n
105	0.298032	0	…	0.028935
106	2.790799	0.028935	…	0
…	…	…	…	…
272	0.960648	0	…	0
273	1.349827	0.120081	…	0

Table 34: Daily series of irrigation water discharge for each irrigation unit (*<yyyy>_Qsurplus.csv*) output. The daily series starts and ends according to StartIrrSeason and EndIrrSeason input.

DoY	SubDistr_1	SubDistr_2	…	SubDistr_n
105	3.161	0	…	0.187042
106	12.520001	0.152834	…	0.187042
…	…	…	…	…
272	11.559352	0	…	0.168336
273	11.170174	0	…	0.168336

Table 35: Single cell tab separated values files output.

Default name	Description
Single cell output tables – Standard mode	
<yyyy>_cell_<x>_<y>.csv	Daily series of (*x*, *y*) cell balance for each year *yyyy*.
<yyyy>_cellinfo_<x>_<y>.csv	(*x*, *y*) cell balance parameters for each year *yyyy*.
<yyyy>_cellparameters_<x>_<y>.csv	(*x*, *y*) cell yield production parameters for each year *yyyy*.
Single cell output tables – Debug mode	
<yyyy>_convergence_<x>_<y>.csv	Daily series of (*x*, *y*) cell maximum number of iterations for the evaporative and the transpirative layers

To analyse and verify water balance for the cells selected in *cells.txt*, if provided, two cell input parameters files are generated, one referred to the spatialised input (*<yyyy>_cellinfo_<x>_<y>.csv*, Table 36) and the other concerning crop parameters (*<yyyy>_cellparameters_<x>_<y>. csv*, Table 37).

Finally, daily series of variables are print in *<yyyy>_cell_<x>_<y>. csv*.

In debug mode, the daily series of the number of interaction cycles for each hour and layer of the selected cells (*<yyyy>_convergence_<x>_<y>.csv*) are also generated (Table 38).

Table 36: (x, y) cell balance parameters for year yyyy (*<yyyy>_cellinfo_<x>_<y>.csv*).

input files		
Domain		1
Slope		1.11E-04
Soil use		13
Irrigation subdistrict		50514
Irrigation method		1
Method efficiency		null
Conveyance efficiency		1
Hydrologic soil group		2
Hydrologic condition		1
Thetal_sat		0.534407973
Thetal_fc		0.253692001
Thetal_wp		0.120725997
Thetal_r		7.41E-02
Thetall_sat		0.464414001
Thetall_fc		0.224792004
Thetall_wp		0.112239003
Thetall_r		7.56E-02
ksat_I		2.76938796
ksat_II		4.03598881
expn_I		8.61455822
expn_II		8.85784435
CapFluxParam_a3		-1.29999995
CapFluxParam_a4		4.5999999
CapFluxParam_b1		-0.170000002
CapFluxParam_b2		-0.270000011
CapFluxParam_b3		6.5999999
CapFluxParam_b4		-0.649999976
Water depth		10.3811703
am_perc1		7.41103554
am_perc2		7.01468611
bm_perc1		0.50613755
bm_perc2		0.64485985
Meteorological Station nr	1	656.495178
Meteorological Station nr	2	150.272095
Meteorological Station nr	3	136.164597
Meteorological Station nr	4	134.040207
Meteorological Station nr	5	653.027893

Table 37: (x, y) cell yield production parameters for year yyyy (*<yyyy>_cellparameters_<x>_<y>. csv*).

input files	
WPadj	0.349999994
HI	0.569999993
KyT	1.25
Ky1	0.400000006
Ky2	0.899999976
Ky3	1.5
Ky4	0.5
Tcrit	32
Tlim	40
kini	0.150000006
dij	0.98312211
ge	97
irandom	20

Table 38: Daily series of (*x, y*) cell maximum number of iterations for the evaporative and the transpirative layers (*<yyyy>_convergence_<x>_<y>.csv*) output.

Date	hour	mmax1	nIter1	mmax2	nIter2
1	1	1	1	1	1
1	2	1	1	1	1
1	3	1	2	1	1
…	…	…	…	…	…
366	23	1	1	1	1
366	24	1	1	1	1

Table 39: Meteorological and phenological processing tab separated values files output, generated in debug mode.

Default name	**Description**
Output tables – Debug mode	
<yyyy>_et0_stations. csv	Daily series of reference evapotranspiration for each meteorological station and for each year *y* [real].
<yyyy>_PhenoLengths. csv	Growing period parameters for each meteorological station and for each year *y* [integer].

Finally, daily series of reference evapotranspiration (*<yyyy>_et0_stations.csv*; Table 40) and growing period parameters for each meteorological station (*<yyyy>_PhenoLengths.csv*) are provided.

Table 40: Daily series of reference evapotranspiration for each meteorological station (*et0_stations.csv*) output.

data	100.dat	106.dat	…	1211.dat
1	1.18988585	0.845626652	…	1.33743763
2	0.432076991	0.865205526	…	0.34808284
…	…	…	…	…
366	0.926681459	0.757623971	…	0.6051687