3. SWIM Code Structure and Input Parameters

In this chapter the code structure of the both parts of the modelling system:

- the SWIM/GRASS interface, and
- the simulation part of SWIM

are described. The SWIM/GRASS interface is applied to prepare necessary (but not all) input files to run the simulation part of SWIM. The simulation part of SWIM performs simulation of ecohydrological processes in river basins or regions.

In section 3.1 the code structure of the SWIM/GRASS interface is given, and in section 3.2 the code of the SWIM simulation part is described. The latter is described in more detail. The code development is continuing, and it is assumed that SWIM users should be able to understand and, if necessary, to modify some modules/subroutines of the simulation part or add new modules. Therefore the latter is described in more detail.

In Section 3.3 input and output files are described, and in Section 3.4 all input parameters are listed and defined.

3.1 Structure of the SWIM/GRASS Interface

It is recommended to read the overview of SWIM/GRASS interface (Section 1.3) before reading this section.

The code includes menus and all menu-driven operations as described in Section 1.3. The subroutine *main* displays the first menu, which allows either to create a new project, or to copy, remove, or work on an existing project. The subroutine *main_menu()* provides the main menu, which lists steps to be completed by the interface.

When the step is chosen, the *sub_menu()* switches between the following important subroutines (see **Fig. 3.1**):

- *get_basin_info* - to request and extract a basin information from the user supplied map layer,
- *hydro_struct_swim* – to create structure file for a basin based on basin map, land use map and soil map,
- *get_topo_info* – to request an elevation map and extract the topographic properties using programs *ram.sub-basin* and *compute_slp_len*,
- *rw_gw_swim* – to read the extracted groundwater properties and write them in SWIM format,
- *com_rout_info_swim* – to compute routing structure, including basin number, starting elevation, ending elevation, starting and ending accumulation cells, stream length and the next draining sub-basin number
- *get_climate_station_s* – to extract the numbers of nearest 3 climate stations for a basin or each sub-basin,
- *get_precipit_station_s* – to extract the number of nearest 3 precipitation stations for a basin or each sub-basin,
- *write_cio_1* - to write the extracted control properties in SWIM format,
- *save_swim_project* – to save the project status information in proj_file which is the project_name with .proj extension. This program will be used at the end of each project, in order to keep track of the project status.

All files included in SWIM/GRASS interface are listed and described in **Tab. 3.1.** One file contains one subroutine with a similar or the same name. **Tab. 3.2** presents the file *swimmake,* which is used to compile SWIM/GRASS interface.

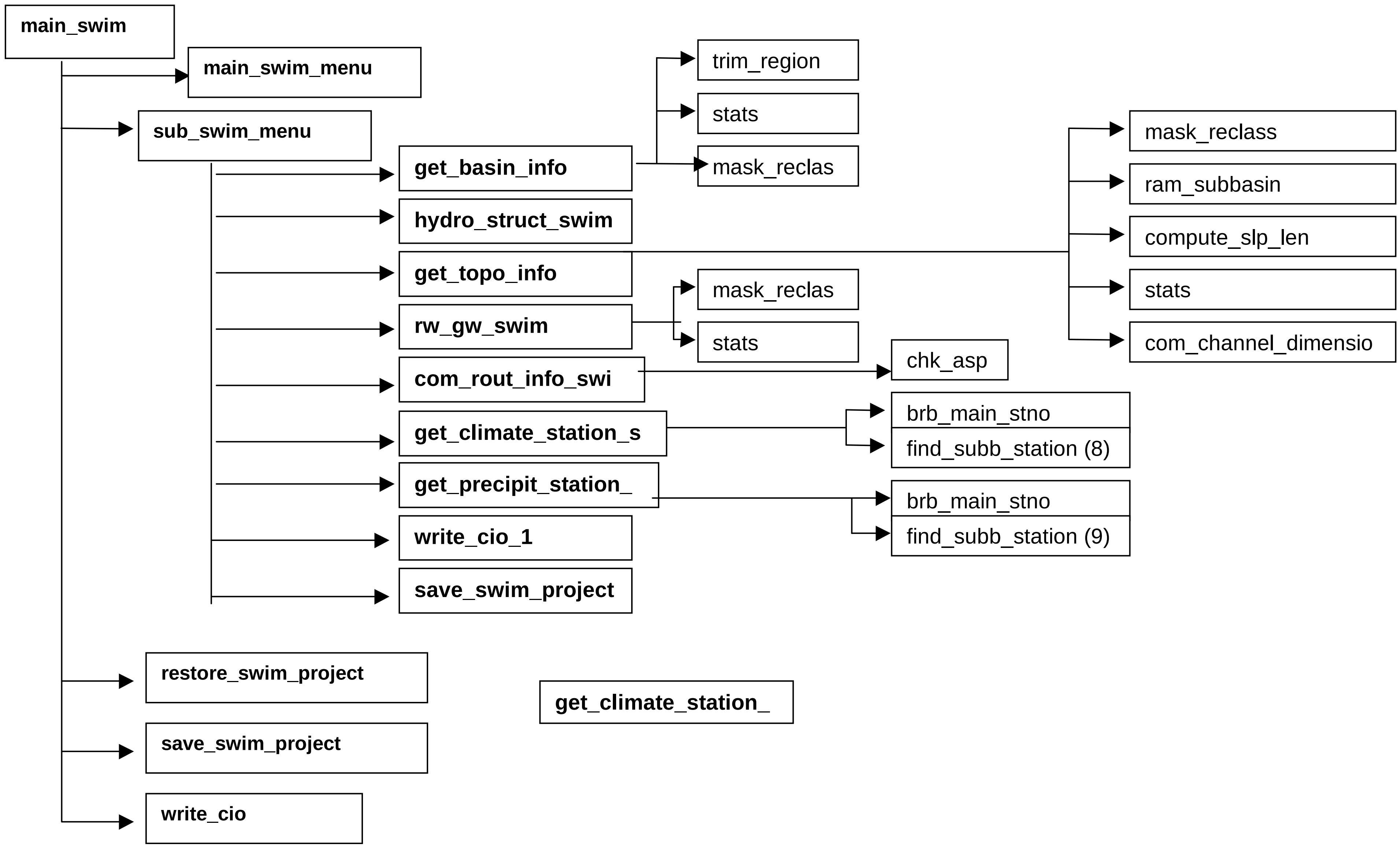

Fig. 3.1 Function tree of SWIM/GRASS interface

Table 3.1 Description of files included in SWIM/GRASS interface

No	*I line: file name* *II line: subroutine name*	*file/subroutine function*
1	cell_open.c int cell_open(name,mapset)	To open an old map with *name* in the *mapset* and return the file id.
2	cell_open_new.c int cell_open_new(name)	To open a new map with *name* in the current mapset and return the file id.
3	check_name.c int check_name(name,array,n)	To check if *name* corresponds to *array*.
4	chk_asp.c chk_asp()	To check the routing (aspect) data that was created in *com_rout_info().* This will interactively allow the user to change the aspect of the basins and to store that information in *proj_name.asp* file under *proj_name* directory. It will be read by restore_project while starting the project if the status [8] is done.
5	com_rout_info_swim.c com_rout_info()	To extract routing info from *basin->p*, *t.aspect*, and *temp_basin_acc* maps; The information gathered will be stored in a structure which will have basin number, starting elevation, ending elevation, starting and ending accumulation cells, stream length and the next draining sub-basin number.
6	compute_slp_len.c compute_slp_len(elev_map)	To generate a slope and aspect maps from the given elevation map using the neighbourhood technique for slope prediction. It creates a new temporary map called *temp_slope* which has the values of slope in tenths of percent and also creates a slope length map according to the unit stream power theory.
7	display_info.c display_info()	An option to display a raster map, a site map, a vector map, to display basin number, and to restore the screen
8	dummy_lcra.c dummy_data()	dummy data for .cod file (not used in SWIM)
9	find_subb_stations.c find_subb_stations()	This subroutine establishes correspondence between sub-basins and climate/precipitation stations
10	forms.c form1()	Forms to fill in (variant SWRRB, not used in SWIM)
11	get_basin_info.c get_basin_info()	This subroutine requests/extracts a basin information from user or from user supplied layer
12	get_climate_station_s.c get_climate_station()	This subroutine extracts the numbers of nearest 3 climate stations for a basin or each sub-basin using program brb_main_stationno.c. The climate station number list has to be stored in a file under the active directory. The station number(s) is (are) stored in climstat_3.dat under full_path. A label file called proj_name.climstat_3 is defined which can be used to mark the stations in a map.
13	get_crop_info.c get_crop_info()	This subroutine requests/extracts the crop properties from user supplied land use map. Each sub-basin is masked and the dominant land use name is selected then the *findcrop* routine is called to write out the land use properties in the needed format (SWAT version, not used in SWIM).
14	get_irr_nutrient_info.c get_irr_nutrient_info()	This automatically creates .mco file for each sub-basin depending on the user's choice for auto fertilizer and auto irrigation (SWAT variant, not used in SWIM).

15	get_mapset.c char *get_mapset(name)	To return the mapset of the map layer with the name. *name*: Name of the map whose mapset is needed. The mapset location is returned.
16	get_new_name.c char *get_new_name(promt,name)	To get the new map *name* in the current mapset using the specified *prompt*. It checks for the existence for the map layer with the same name and returns name, if succeeds, else quits. *promt*: any string to give info to the user what the program expects. *name*: name of the new map layer.
17	get_old_name.c char *get_old_name(prompt)	To get the old map name in any mapset using the specified *prompt*. It checks for the existence for the map layer and returns name, if succeeds else quits. *prompt*: To identify the layer one needs.
18	get_precipit_station_s.c get_pecipitation_station()	This subroutine extracts the number of nearest 3 precipitation stations for a basin or each sub-basin using program brb_main_stationno.c. The precipitation station number list has to be stored in a file in the active path. The station number(s) is (are) stored in prstat_3.dat under full_path. A label file called proj_name.prstat_3 is defined which can be used to mark the stations in a map.
19	get_rain_temp_info.c get_rain_temp_info()	This subroutine requests/extracts the rain gauge and temperature gauge station properties from user supplied map. From each sub-basin get the file name that is correspond to that sub-basin. The file has to be in SWAT format (not used in SWIM).
20	get_res_inflow.c get_res_inflow()	This requests/extracts reservoir, inflow and (re)compute the routing structures to create .fig file through a menu system (not used in SWIM).
21	get_soil_info.c get_soil_info()	This requests/extracts the soil properties from user supplied soils map. Each sub-basin is masked and the dominant soil name is picked and the *findsoil* routine is called to write out the soil properties in SWAT format (not used in SWIM).
22	get_topo_info.c get_topo_info()	This subroutine requests/extracts an elevation map from user and extracts the topological properties such as stream length and stream slope using *ram.sub-basin* program and average overland slope and slope length using *compute_slp_len* subroutine. It creates several intermediate layers like *temp_LS, temp_slen, temp_sslp, and temp_slope.*
23	hydro_struct_swim.c hydro_struct()	This function requests a basin map, land use map and soil map for one area, starts r.stats for these three maps, stores the output in "*proj_name.str*" under full_path except these where one of the first numbers is zero.
24	main_swim.c main()	Main program
25	main_swim_menu.c main_menu()	This subroutine provides the first menu to start with the SWIM/GRASS project.
26	mask_reclass.c mask_reclass(layer_name,cellnum,flag)	To create a temporary file for reclassification the *cellnum* into MASK layer
27	read_basin.c read_basin()	This subroutine reads the extracted basin properties in SWIM format. The basin file are stored in "data_dir" and are read in to the SWIM variables.

28	read_cod.c read_cod()	This subroutine reads the extracted control properties in SWIM format. The basin file are stored in "data_dir" and are read in to the SWIM variables.
29	read_crop.c read_crop(num)	This subroutine reads the extracted crop properties that are stored in SWIM format by the *findcrop* program from user supplied crop map. The crop files are stored in "data_dir" and are read in to the SWIM variables for each sub-basin "num"
30	read_res.c read_res(num)	This subroutine reads the user specified reservoir data that are stored in SWIM format. The reservoir files are stored in "data_dir" as res_num.res and are read in to the SWIM variables for each sub-basin "num" (not used in SWIM)
31	read_rout.c read_rout(num)	This subroutine reads the extracted sub-basin routing properties in SWIM format. The basin file are stored in "data_dir" and are read in to the SWIM variables for sub-basin "num".
32	read_soil.c read_soil(num)	This subroutine reads the extracted soil properties that are stored in SWIM format by the *findsoil* program from user supplied soils map. The soil files are stored in "data_dir" and are read in to the SWIM variables for each sub-basin "num"
33	read_sub-basin.c read_sub-basin(num)	This subroutine reads the extracted sub-basin properties in SWIM format. The basin file are stored in "data_dir" and are read in to the SWIM variables for sub-basin "num".
34	read_weather.c read_weather()	This subroutine reads the generated weather parameters from the weath_gen program which is stored in SWAT format. The weather station was selected is the most closest station from the approximate centre of the Basin. The weather file is stored in "data_dir" and are read in to the SWAT variables for the whole basin (not used)
35	restore_swim_project.c restore_project()	This subroutine program retrieves the project status information from proj_file which is the project_name with .proj extension into appropriate variables.
36	rm_rast_map.c rm_rast_map(map)	This subroutine removes current raster map
37	rw_gw_swim.c read_gw()	This subroutine reads the extracted groundwater properties in SWIM format. The groundwater parameter file are stored in "data_dir" and are read in to the SWIM variables.
38	save_swim_project.c save_project()	This program saves the project status information in proj_file which is the project_name with .proj extension. This program will be used at the end of each project, in order to keep track of the project status
39	stats.c stats(layer_name, flag, stat)	This routine gets the categories, area/# of cells of layer. this routine uses r.stats program flag: 1 - stores the output in a variables 2 - returns the cell number that has maximum occurrence 3 - returns the average value of the cell number 4 - returns the weighted average value of the cell number
40	sub_swim_menu.c sub_menu()	This subroutine is the major menu in a loop to update the various data from either layers or user inputs This program will be used while working on a ongoing project.
41	what.c	This subroutine checks current GRASS window

42	write_basin.c write_basin()	This subroutine writes the extracted basin properties in SWIM format. The basin file are stored in "data_dir" and are read as the SWIM variables.
43	write_cio_1.c write_cio()	This subroutines writes the extracted control properties in SWIM format. The basin file are stored in "data_dir" and are read as the SWIM variables.
44	write_cod.c write_cod()	This subroutine writes the extracted control properties in SWIM format. The basin file are stored in "data_dir" and are read as the SWIM variables.
45	write_crop.c write_crop(num,crp_fl)	This subroutine writes the extracted crop properties from crp_fl database in SWIM format. The crop files are stored in "data_dir" and are read as the SWIM variables for each sub-basin "num"
46	write_res.c write_res(num)	This subroutine writes the reservoir data provided by user in SWIM format The reservoir files are stored in "data_dir" as res_num.res and are read as the SWIM variables for each sub-basin "num"
47	write_rout.c write_rout()	This subroutine writes the extracted sub-basin routing properties in SWIM format. The basin file are stored in "data_dir" and are read as the SWIM variables for sub-basin "num".
48	write_soil.c write_soil(num)	This subroutine writes the extracted soil properties in SWIM format. The soil files are stored in "data_dir" and are read as the SWIM variables for each sub-basin "num"
49	write_sub-basin.c write_sub-basin()	This subroutine writes the extracted sub-basin properties in SWIM format. The basin file are stored in "data_dir" and are read as the SWIM variables for sub-basin "num".
50	write_weather.c	This subroutine writes the generated weather parameters from the weath_gen program and stores in SWIM format. The weather file is stored in "data_dir" and are read as the SWIM variables for the whole basin

Table 3.2 File *swimmake* used to compile SWIM/GRASS interface

```
Bin = ../bin
LIBDIR = /usr/local/grass/source/src/libes/LIB
RASTERLIB = $(LIBDIR)/libraster.a
DISPLAYLIB = $(LIBDIR)/libdisplay.a
D_LIB = $(LIBDIR)/libD.a
GISLIB = $(LIBDIR)/libgis.a
VASK = $(LIBDIR)/libvask.a
LIBES  = $(D_LIB) $(DISPLAYLIB) $(RASTERLIB) $(GISLIB) $(VASK)
EXTRACFLAGS = -L/usr/local/grass/source/src/libes

PGM = swim_input
LIST = \
        cell_open.o\
        cell_open_new.o\
        check_name.o\
        chk_asp.o\
        com_rout_info_swim.o\
        compute_slp_len.o\
        display_info.o\
        dummy_lcra.o\
        find_subb_stations.o\
        forms.o\
        get_basin_info.o\
        get_climate_station_s.o\
        get_crop_info.o\
        get_irr_nutrient_info.o\
        get_mapset.o\
        get_new_name.o\
        get_old_name.o\
        get_precipit_station_s.o\
        get_rain_temp_info.o\
        get_res_inflow.o\
        get_soil_info.o\
        get_topo_info.o\
        hydro_struct_swim.o\
        main_swim.o
        main_swim_menu.o\
        mask_reclass.o\
        read_basin.o\
        read_cod.o\
        read_crop.o\
        read_res.o\
        read_rout.o\
        read_soil.o\
        read_sub-basin.o\
        read_weather.o\
        restore_swim_project.o\
        rm_rast_map.o\
        rw_gw_swim.o\
        save_swim_project.o\
        stats.o\
        sub_swim_menu.o\
        what.o\
        write_basin.o\
        write_cio_1.o\
        write_cod.o\
        write_crop.o\
        write_res.o\
        write_rout.o\
        write_soil.o\
        write_sub-basin.o\
        write_weather.o\

$(PGM): $(LIST) $(LIBES)
            $(CC) $(LDFLAGS) -o  $@ $(LIST) $(LIBES) -lm -lcurses
```

3.2 Structure of the SWIM Simulation Part

3.2.1 Files and their Functions

The simulation part of the model code consists of 33 files listed in **Tab. 3.3**. They can be subdivided regarding their main functions into the following parts:

1) main administrative files, representing three-level disaggregation procedure: basin - sub-basin - hydrotope,
2) climate data input or generation,
3) hydrological processes,
4) erosion, crop/vegetation growth and nutrient processes,
5) routing of water, sediments and nutrients,
6) administrative subroutines (common blocks, read input files, initialisation of variables, writing of results, and statistical evaluation of results).

Every file contains one or several subroutines with similar functions. Altogether there are 85 subroutines in the simulation part. General functions performed in the files are also shortly described in **Tab. 3.3**.

The block-scheme of the model operations is presented in **Fig. 3.2**. It shows the sequence of computing different processes.

The file *Makefile* used to compile SWIM code is given in **Tab. 3.4**.

Table 3.3 Files and subroutines included in SWIM code

No	*File name*	*Subroutines*	*Main function*
I.	**Main administrative files, representing three-level disaggregation procedure: Basin – Sub-basin - Hydrotope**		
1.	main.f	main	Initialisation (calls subroutines reading input data), annual and daily loops, routing structure, and aggregation of results for the basin
2.	sub-basin.f	sub-basin()	Sub-basin operations: initial conditions in hydrotopes in the first day, calls hydrotop, aggregates hydrotop outputs, and provides outputs from sub-basin for routing
3.	hydrotop.f	hydrotop()	Simulation of all hydrological, vegetation and nutrient cycling processes in hydrotopes
II. Climate parameters read or generate			
4.	cliread.f	cliread + sub2prst	Reading observed climate and hydrological data
5.	clicon.f	clicon + clgen()	Generation of daily climate data from monthly statistical data
III. Hydrological processes			
6.	solt.f	solt() + snom()	Calculation of soil temperature and snow melt
7.	curn.f	curno() + volq() + peakq() + tran()	Simulation of daily runoff, peak runoff rate and transmission losses for hydrotopes
8.	evap.f	evap()	Calculation of soil evaporation and potential plant transpiration
9.	perc.f	purk() + perc() + percrack()	Calculation of percolation and lateral subsurface flow from soil
10.	gwat.f	gwmod()	Calculation of groundwater contribution to streamflow
IV Erosion, crop and nutrients			
11.	eros.f	ecklsp() + ysed()+ enrsb() + orgnsed()+ psed()	Simulation of erosion processes
12.	crop.f	crpmd() + operat() + growth()	Simulation of crop planting, growth and harvesting
13.	veget.f	vegmd()	Simulation of non-crop vegetation
14.	vegfun.f	wstress() + tstress() + npstress() + scurve() + ascrv() + adjustbe()	Special functions for crop and vegetation: water, temperature and N&P stress, CO2 adjustment of the biomass-energy ratio (alpha factor)
15.	cropyld.f	cryld_brb	Calculation of crop yield for districts in Brandenburg
16.	ncycle.f	ncycle() + nlch() + nuptake() + fert()	Simulation of N cycle in soil

17.	pcycle.f	pcycle() + psollch() + puptake()	Simulation of P cycle in soil
V Routing of water, sediments and nutrients			
18.	route.f	route()+ add()+ transfer()	Calculation of water, sediment and nutrient routing
19.	routfun.f	rthyd()+ rtsed()+ enrrt()+ rtorgn+ rtpsed+ ttcoefi()+ coefs()+qman	Routing functions
VI Administrative subroutines			
20.	compar.f		Common parameters: dimensions
21.	common.f		Common blocks: parameters and variables
22.	open.f	open + opensub + opensoil + openstruct + closef + caps()	Opening and closing input/output files
23.	readcod.f	readcod	Reading xxx.cod* and xxx.fig input files: codes for print and routing
24.	readbas.f	readbas	Reading xxx.bsn and xxx.str input files: basin and calibration parameters, and hydrotope structure
25.	readcrp.f	readcrp	Reading crop.dat input file
26.	readsub.f	readsub	Call of readwet, reading sub-basin input files: xxxNN.sub**, xxxNN.gw, and xxxNN.rte
27.	readsol.f	readsol + rflowtt()	Reading soil parameters from soilNN.dat files
28.	readwet.f	readwet	Reading monthly weather statistical parameters for the basin
29.	init.f	blockdata + init + initsums + initsub	Block data and initialisation of variables
30.	initcrop.f	initcrop()	Initialisation of crop management parameters
31.	genres.f	wr_daily, wr_month(), wr_annual	Writing daily, monthly and annual general results
32.	flohyd.f	flomon() + floann + floave + crop_gis + hydro_gis	Writing monthly, annual and average annual water and N flows for selected hydrotopes; Writing crop yield and annual water flows (for hydrotopes) in the GRASS input format
33.	stat.f	alpha() + gammad() + distn() + gcycl() + randn() + xmonth + xnash()	Statistical functions and criteria of fit

* xxx substitutes here a basin name specified when using SWIM/GRASS interface

** xxxNN substitutes here a sub-basin number NN name established automatically by SWIM/GRASS interface

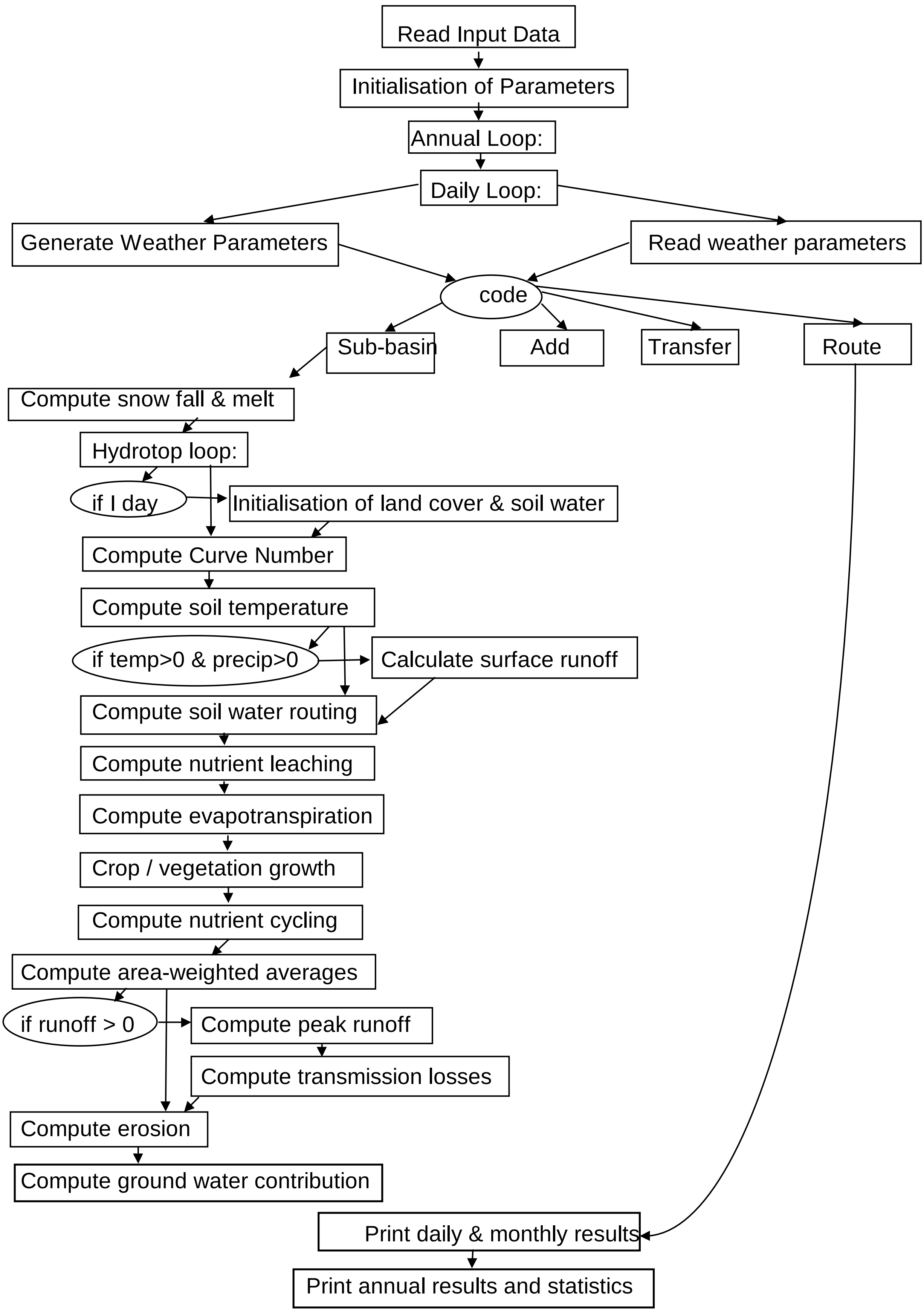

Fig. 3.2 Scheme of operations in the simulation part of SWIM

Table 3.4 File *Makefile* used to compile SWIM code

```
OBJ =     clicon.o\
          cliread.o\
          crop.o\
          cropyld.o\
          curn.o\
          eros.o\
          evap.o\
          flohyd.o\
          genres.o\
          gwat.o\
          hydrotop.o\
          init.o\
          initcrop.o\
          main.o\
          ncycle.o\
          open.o\
          pcycle.o\
          perc.o\
          readcod.o\
          readcrp.o\
          readbas.o\
          readsub.o\
          readsol.o\
          readwet.o\
          route.o\
          routfun.o\
          solt.o\
          sub-basin.o\
          stat.o\
          veget.o\
          vegfun.o
FFLAGS = -g -qflttrap=invalid:zerodivide:overflow:enable

swim: $(OBJ)
	f77 $(FFLAGS)   $(OBJ) -lm -bloadmap:map.out  -o swim
$(OBJ): common.f
.f.o:
	f77 $(FFLAGS) -c $*.f
clean:
	rm -f $(OBJ) swim
```

3.2.2 Subroutines and their Functions

All the subroutines included in SWIM code are shortly described in **Tab. 3.5**. In addition, the Table indicates, where every subroutine is called.

Table 3.5 Description of subroutines included in SWIM simulation part

File	*Subroutine*	*Subroutine description*	*Called in*
main.f	main	Main program. Calls subroutines reading input data and initialisation subroutines. Establishes annual and daily loops, and the routing structure. Aggregates results for the basin.	
sub-basin.f	sub-basin	Sub-basin operations: initialisation in hydrotopes, call hydrotop, aggregation of hydrotope outputs, setting lateral flows for routing	main
hydrotop.f	hydrotop	Simulation of all hydrological, vegetation and nutrient cycling processes in hydrotopes	sub-basin
cliread.f	cliread	this subroutine read climate data	main
	sub2prst	this subroutine establishes the correspondence between sub-basins and precipitation stations	main
clicon.f	clicon	this subroutine controls weather inputs	main
	clgen()	this subroutine simulates daily solar radiation, daily precipitation, and max. and min. air temperature at the user option	clicon
solt.f	solt	this subroutine estimates daily average temperature at the bottom of each soil layer	hydrotop
	snom	this subroutine calculates daily snow melt when the average air temperature exceeds 0 degrees	sub-basin
curn.f	curno()	this subroutine sets curve number parameters	hydrotop
	volq()	this subroutine predicts daily runoff given daily precipitation and snowmelt using a modified curve number approach	hydrotop
	peakq()	this subroutine computes the peak runoff rate using a modification of the Rational Formula	sub-basin
	tran()	this subroutine computes channel transmission losses	sub-basin
evap.f	evap()	this subroutine computes the amount of soil evaporation and potential plant evaporation using Ritchie's model	hydrotop

perc.f	purk()	this is the master percolation component It divides each layer's flow into 4 mm slugs and manages the routing process	hydrotop
	perc()	this subroutine computes percolation and lateral subsurface flow from a soil layer when field capacity is exceeded - hillflow method	purk
	percrack	this subroutine computes percolation by crack flow	purk
gwat.f	gwmod()	this subroutine estimates groundwater contribution to streamflow	sub-basin
eros.f	ecklsp()	this subroutine calculates K, P, and LS factors for hydrotope	hydrotop
	ysed()	this subroutine predicts daily soil loss caused by water erosion using the Modified Universal Soil Loss Equation	sub-basin
	enrsb()	this subroutine computes enrichment ratio for sub-basin	sub-basin
	orgnsed()	this subroutine computes organic N loss with erosion	sub-basin
	psed()	this subroutine computes P loss with erosion	sub-basin
crop.f	crpmd()	Main crop routine: calls operat() and growth() subroutines	hydrotop
	operat	this subroutine sets crop operations: planting, harvesting, and kill	crpmd
	growth	this subroutine predicts daily potential growth of total crop biomass and roots and calculates the leaf area index. It incorporates residue and decays residue on ground surface. It calls subroutines *wstress* and *tstress* and adjusts daily dry matter to stress.	crpmd
veget.f	vegmd()	this subroutine predicts daily potential growth of total plant biomass and roots and calculates the leaf area index.	hydrotop
vegfun.f	wstress()	this subroutine distributes potential plant transpiration through the root zone and calculates actual plant water use based on soil water availability. It estimates water stress factor for crops.	crpmd, vegmd
	tstress()	this subroutine computes temperature stress for crop growth	growth, vegmd
	npstress()	this subroutine computes N & P stress factor	nuptake, puptake
	scurve()	S-curve function	growth, adjustbe
	ascrv()	S-curve function	readcrp
	adjustbe()	this subroutine adjusts biomass-energy ration to CO_2 concentration	growth

cropyld.f	cryld_brb	this subroutine calculates crop yield for districts in BRB, closed if not Brandenburg	main
ncycle.f	ncycle()	this subroutine calculates N cycle: daily N mineralisation and immobilisation considering fresh organic material (crop residue) and active and stable humus	hydrotop
	nlch()	this subroutine computes nitrate leaching from soil	hydrotop
	nuptake()	this subroutine computes N-uptake by crops and natural plants	growth, vegmd
	fert()	this subroutine applies N and P specified by date and amount	hydrotop
pcycle.f	pcycle	this subroutine computes P cycle: P flux between labile, active mineral and stable mineral P pools	hydrotop
	psollch	this subroutine computes soluble P leaching	hydrotop
	puptake	this subroutine computes P-uptake by crops and natural vegetation	growth, vegmd
route.f	route()	this subroutine controls the channel routing	main
	add()	this subroutine adds outputs for	main
	transfer()	this subroutine controls the channel routing	main
routfun.f	rthyd()	this subroutine routes a daily flow through a reach using a constant storage coefficient	route
	rtsed()	this subroutine routes sediment from sub-basin to basin outlets, accounting for deposition (based on fall velocity) and degradation in stream	route
	enrrt()	this subroutine computes enrichment coefficient for N routing	route
	rtorgn	this subroutine computes organic N routing	route
	rtpsed	this subroutine computes P routing	route
	ttcoefi()	this subroutine computes travel time coeffs phi() for the simplified routing	main
	coefs()	this subroutine calculates routing coefficients	ttcoefi
	qman()	this subroutine computes flow using Manning equation	ttcoefi
open.f	open	this subroutine opens main input files	main
	opensub	this subroutine opens sub-basin input files	readsub
	opensoil	this subroutine opens soil data files 15	readsol
	openstruct	this subroutine opens 16 - structure file	readbas
	closef	this subroutine closes files 11,12,13,14	main
	caps()	this subroutine removes extra blanks	open, opensub, opensoil, openstruct

readcod.f	readcod	this subroutine reads codes for printing, routing	main
readbas.f	readbas	this subroutine reads basin parameters and calibration parameters	main
readcrp.f	readcrp	this subroutine reads crop parameters	main
readsub.f	readsub	this subroutine reads sub-basin input parameters	main
readsol.f	readsol	this subroutine reads soil input parameters	main
	rflowtt()	this subroutine computes return flow travel time	readsol
readwet.f	readwet()	this subroutine reads monthly statistical weather parameters, provided by user	readsub
init.f	block data	block data	
	init	this subroutine initialises variables in main	main
	initsums	this subroutine initialises variables for GIS output	main
	initsub	this subroutine initialises sub-basin variables	sub-basin
initcrop.f	initcrop()	this subroutine initialises crop management	main
genres.f	wr_daily	this subroutine writes daily general results	main
	wr_month()	this subroutine writes monthly general results	main
	wr_annual	this subroutine writes annual general results	main
flohyd.f	flomon()	this subroutine writes monthly water and N flows for selected hydrotopes	main
	floann	this subroutine writes annual water and N flows for selected hydrotopes	main
	floave	this subroutine writes average annual water and N flows for selected hydrotopes	main
	crop_gis	this subroutine writes crop yield for GRASS	sub-basin
	hydro_gis	this subroutine writes annual sums of water flows for hydrotopes (for GRASS)	sub-basin
stat.f	alpha()	this subroutine computes alpha, the fraction of total rainfall that occurs during 0.5h	sub-basin, clicon
	gammad()	this function provides numbers from gamma distribution	alpha
	distn()	this function computes the distance from the mean of a normal distribution	clgen
	gcycl()	this function cycles the random number generator	main
	randn()	this function provides random numbers ranging from 0. to 1.	init, readsub, gcycl, clicon, clgen, gammad
	xmonth	this subroutine calculates the month, given the day of the year	main
	xnash()	this subroutins computes criteria of fit	main

3.2.3 Main Administrative Subroutines and the Parameter Read Part

The subroutine *main* performs initialisation of the simulation run by reading input data and initialising variables and parameters. It establishes annual and daily loops, and inside the daily loop it calls sequentially the *sub-basin* subroutine for every sub-basin to calculate all processes in sub-basin, and then routes lateral flows to the basin outlet following the routing structure file *xxx.fig*. The main subroutine also writes daily, monthly and annual results for the basin.

The subroutine *sub-basin* performs sub-basin operations: initialisation of variables for hydrotopes, calling of *hydrotope* subroutine, aggregation of hydrotope outputs, and setting lateral flows for routing.

The subroutine *hydrotop* controls simulation of all hydrological, vegetation and nutrient cycling processes in hydrotopes, by calling different related subroutines.

The structure of *main, sub-basin* and *hydrotop* subroutines is shown in the following **Tables 3.6, 3.7** and **3.8**. The user has to keep in mind that some of the calls are conditional, though it is not indicated in the Tables.

Table 3.6 Structure of the subroutine MAIN

```
MAIN PROGRAM BEGIN:
call open
call readcod
call gcycl(ign,xx)
call readbas
call sub2prst
call readsub
call readsol
call init
call ttcoefi(j), j=1,lu
call closef
ANNUAL LOOP: CYCLE 540:iy = 1, nbyr
        call initcrop(iy)
        DAILY LOOP: CYCLE 530: i = id1, nd
                call initsums
                call cliread
                SWITCH:  icode: (1,2,5)
                        call sub-basin(icode,ihout,inum1,inum2,inum3,rnum1)
                        call route(icode,ihout,inum1,inum2,inum3,rnum1,nrch)
                        call transfer (icode,ihout,inum1,inum2,inum3)
                        call add (icode,ihout,inum1,inum2,inum3)
                call wr_daily
                call xmonth
                call flomon(mo1)
                call wr_month(mo1)
                call floann
                call wr_annual
        END OF CYCLE 530
        call xnash(runo,runs,365)
        call cryld_brb
END OF CYCLE 540
call floave
MAIN PROGRAM END
```

Table 3.7 Structure of the subroutine SUB-BASIN

```
SUB-BASIN BEGIN:

call initsub

call snom(j)

call alpha(j

EAP LOOP: CYCLE 100: jea = 1, neap(j)

        call hydrotop(j,jea,k,n)

              call crop_gis(j,jea,k)

              call hydro_gis(j,jea)

END OF EAP LOOP 100

call peakq(j)

call tran(j)

call ysed(j)

call enrsb(j)

call orgnsed(j)

call psed(j)

call gwmod(j)

SUB-BASIN END
```

Table 3.8 Structure of the subroutine HYDROTOP

```
HYDROTOP BEGIN:
call curno(cn2(k,n),j,jea,k,n)
call solt(zz,j,jea,k,n)
call volq(j,jea,k,n)
call ecklsp(j,jea,k,n)
call purk(j,jea,k,n)
        call perc(j,je,k,n,j1,j2)
call evap((j,jea,k,n)
call fert(j,jea,k,n,ii)
SWITCH n:
        call crpmd(j,jea,k,n)
                call operat(j,je,k,n)
                call wstress(j,je,k,n)
                call growth(j,je,k,n)
                        call tstress(tgx,j,n)
                        call adjustbe()
                        call nuptake(j,je,k,n)
                                call npstress(uno3,uno3pot,uu)
                        call puptake(j,je,k,n)
                                call npstress(uap,uapot,uu)
                        call scurve()
        call vegmd(j,jea,k,n)
                call wstress(j,je,k,n)
                call tstress(tgx,j,n)
                call nuptake(j,je,k,n)
                        call npstress(uno3,uno3pot,uu)
                call puptake(j,je,k,n)
                        call npstress(uap,uapot,uu)
call nlch(j,jea,k,n)
call psollch(j,jea,k,n)
call ncycle(j,jea,k,n)
call pcycle(j,jea,k,n)
HYDROTOP END
```

Table 3.9 Structure of the subroutines OPEN, READCOD, READBAS, READCRP, READSUB, READSOL, READWET

Subroutine	*Objective*	*Internal structure**
open	open files	read 1 - file = file.cio **call caps**
readcod	read codes	read 2 - file = codedat read 3 - file = routin
readbas	read basin parameters, crop parameters and basin structure file	read 4 - file = basndat read 7 - file = struct.dat **call openstruct**::: read 8 - file = str.cio call caps open 9 - file = strdat write 9 **call openstruct**::: read 8 - file = str.cio call caps open 9 - file = strdat read 9 - file = strdat
readcrp	reads crop parameters	read 5 - file = crop.dat **call ascrv**() **call ascrv**() **call ascrv**()
readsub	reads sub-basin parameters	**call opensub** call caps **call readwet** read 12 - file = subdat read 13 - file = gwdat read 14 - file = routdat
readwet	reads weather parameters	read 11 - file = wgendat
readsol	reads soil parameters	**call opensoil** call caps **call rflowtt()**

* for some files an internal name is indicated (e.g. routin), see Table 3.10 for clarification

3.3 Input and Output Files

For application of the model the user has to prepare a number of input files. Regarding the way of data preparation, all the files can be subdivided into following categories:

- files created by SWIM/GRASS interface,
- climate and hydrological data prepared by user,
- soil data – standard (BÜK-1000) or created by user in the same format
- standard crop database (file **crop.dat** is ready),
- four additional files prepared by user from example files.

The following input files are prepared by SWIM/GRASS interface (see **Tab. 3.10**):

- **file.cio** – 1 file,
- **str.cio** – 1 file,
- **xxx.fig** – 1 file,
- **xxx.str** – 1 file,
- **xxxNN.sub** – M files, where M is the number of sub-basins,
- **xxxNN.gw** – M files, where M is the number of sub-basins,
- **xxxNN.rte** – M files, where M is the number of sub-basins.

The user has to prepare either one climate/hydrological file **clim.dat**, which includes all climate and hydrological data (see Sections 4.2.2 and 4.2.5), or four files

- **prec.dat**
- **temp.dat**,
- **radi.dat**
- **runoff.dat**,

which include separately all precipitation, temperature, radiation, and runoff data for the basin. In addition, a file sub2prst.dat may be prepared, which indicates the correspondence between sub-basins and precipitation stations. This is especially useful in case of a large number of sub-basins.

The user has either to prepare all soil data, or to use available soil database for the Elbe, BÜK-1000, if this map is used. In addition, a file soil.cio has to be prepared or copied, which includes a list of all soil data file names.

The rest four files are the following:

- **xxx.cod**, which includes program codes for printing;
- **xxx.bsn**, which includes a set of basin parameters and a set of calibration parameters,
- **wstor.dat**, which includes initial water storages for the reaches (may be all put to 0, or taken from the test model run at the end of the year), and
- **wgen.dat**, which includes monthly climate statistics.

The monthly statistical data are used to run the weather generator. In case if real climate data are used, SWIM requires only average monthly maximum and minimum temperatures for the basin (needed for soil temperature routine). They can be calculated using an additional program *wgenpar* from the available climate data, taking as long series as possible. Then the calculated average monthly maximum and minimum temperatures have to be substituted into wgen.dat file.

Table 3.10 Input files

Unit No.	*Internal name*	*External name*	*read in*	*Description*	*Type*
1	file.cio	file.cio	open, opensub	**C**ontrol **I**nput **O**utput file. It contains all the input file names that are used by the model.	created by SWIM/GRASS
2	codedat	xxx.cod	readcod	This is the input control code file. In contains the number of sub-basins, the number of years of simulation, beginning year of simulation and print codes.	created by user from an example file
3	routin	xxx.fig	readcod	This is the basin configuration input file. It contains the routing commands to route and add flows through a basin.	created by SWIM/GRASS
4	basndat	xxx.bsn	readbas	This is the general basin input file. It contains a set of general basin parameters (including drainage area) and a set of parameters that can be used for calibration.	created by user from an example file
7	struct	xxx.str	readbas	This is the basin structure file. It describes sub-basins and hydrotopes by land use categories and soil types.	created by SWIM/GRASS
8	strlist	str.cio	openstruct	This is the control file for the basin structure. It contains file names to write sub-basin structure.	created by SWIM/GRASS
9	direc//strdat	Struc/subbNN.str	readbas	These are sub-basin structure files created by SWIM	created by SWIM
10	sub-prst.dat	sub-prst.dat	sub2prst	This file establishes correspondence between sub-basins and precipitation stations.	created by user
5	cropdb	crop.dat	readcrp	This is the crop data base input file. It contains crop specific parameters that are input to the model.	standard file
11	wgen.dat	wgen.dat	readwet	This is the weather generator input file. It contains monthly statistical parameters required for generating daily weather. Some of the parameters are needed by the model, even if observed weather data are used.	created by user from an example file
12	direc//subdat	Sub/xxxNN.sub	readsub	This is the general sub-basin input file. It contains general inputs specific for each sub-basin (area, land and channel slopes and lengths, etc.	created by SWIM/GRASS

13	direc//gwdat	Sub/xxxNN.gw	readsub	This is the groundwater input file. It contains shallow aquifer data, including a recession parameter, specific yield, a revap coefficient, and a deep aquifer percolation coefficient.	created by SWIM/GRASS
14	direc//routdat	Sub/xxxNN.rte	readsub	This is the sub-basin routing input file. This file contains data on channel dimensions (length, slope, width, depth, etc.) for the min channel in the sub-basin.	created by SWIM/GRASS
20	wstor.dat	wstor.dat	readsub	This file includes data on initial water storage in m^3 in the reaches corresponding to the sub-basins.	created by user from an example file
15	soillist	soil.cio	opensoil	This is the control file for the soil database. It contains soil file names.	created by user or standard (BÜK-1000)
16	direct//soildat	Soil/soilNN.dat	readsol	These are soil parameter input files. They contain soil physical and chemical parameters.	created by user or standard (BÜK-1000)
21	clim.dat	clim.dat	cliread	This is climate (and) hydrological input data file. It may include all necessary climate and hydrological data, or only climate data for the basin.	created by user
22	prec.dat	prec.dat	cliread	This is precipitation input data file. It includes precipitation data from all used precipitation stations in/close to the basin.	created by user
23	temp.dat	temp.dat	cliread	This is temperature input data file. It includes temperature data from all used climate stations in/close to the basin.	created by user
24	radi.dat	radi.dat	cliread	This is radiation input data file. It includes radiation data from all used climate stations in/close to the basin.	created by user
25	runoff.dat	runoff.dat	cliread	This is hydrological input data file. It includes water discharge in the basin outlet (used for the hydrological validation).	created by user

Table 3.11 Output files

Unit No.	*File name*	*Where used*	*Description*
GENERAL OUTPUT			
31	Res/wgen.out	readwet	Weather generator output
32	Res/rin.out	readbas, readsub, readsol	Write input parameters
SUBROUTINES OUTPUT			
41	Res/curn.out	curno, volq	specific subroutine output
42	Res/solt.out	solt	specific subroutine output
43	Res/tran.out	tran	specific subroutine output
44	Res/perc.out	purk, perc	specific subroutine output
45	Res/evap.out	evap	specific subroutine output
46	Res/crop.out	crop, cropyld	specific subroutine output
47	Res/eros.out	ecklsp, ysed, sub-basin	specific subroutine output
48	Res/nutr.prn	ncycle, sub-basin	specific subroutine output
49	Res/rout.out	ttcoefi, rtsed	specific subroutine output
50	Res/wstr.out	wstress	specific subroutine output
HYDROTOPE OUTPUT			
51	Res/htp-1.prn	subbasin	Daily water outputs for a hydrotop (chosen by user)
52	Res/htp-2.prn	subbasin	Daily water outputs for a hydrotop (chosen by user)
53	Res/htp-3.prn	subbasin	Daily water outputs for a hydrotop (chosen by user)
54	Res/htp-4.prn	subbasin	Daily water outputs for a hydrotop (chosen by user)
55	Res/htp-5.prn	vegmd	Daily water outputs for a hydrotop (chosen by user)
56	Res/htp-6.prn	vegmd	Daily water outputs for a hydrotop (chosen by user)
57	Res/htp-7.prn	vegmd	Daily water outputs for a hydrotop (chosen by user)
SUB-BASIN OUTPUTS			
61	Res/subd.prn	subbasin	Daily water flow outputs for all sub-basins
62	Res/subm.prn	wr_month, wr_annual	Monthly sub-basin outputs for all sub-basins
63	Res/sub-1.prn	subbasin	Daily water outputs for a chosen sub-basin
64	Res/sub-2.prn	subbasin	Daily water outputs for a chosen sub-basin
65	Res/sub-3.prn	subbasin	Daily water outputs for a chosen sub-basin
BASIN & RIVER OUTPUTS			
71	Res/bad.prn	wr_daily	Daily water outputs for basin
72	Res/bam.prn	wr_month	Monthly water outputs for basin

73	Res/bay.prn	wr_annual	Annual water outputs for basin
74	Res/rch.prn	route	Reach outputs
75	Res/rvQ.prn	route	Reach outputs (from route)
76	Res/rvaddQ.prn	add	Reach outputs (from add)
70	Res/rvQ-mn.prn	main	Monthly reach outputs (from route)
80	Res/rvQ-ev.out	xnash	Evaluation of hydrological results
MONTHLY, ANNUAL & AVERAGE ANNUAL WATER and N FLOWS			
77	Flo/floMON.prn	flomon	Monthly water and N flows for 3 hydrotopes
78	Flo/floANN.prn	floann	Annual water and N flows for 3 hydrotopes
79	Flo/floAVE.prn	floave	Average water and N flows for 9 hydrotopes
MONTHLY WATER and N FLOWS for 9 CHOSEN SOILS/HYDROTOPES			
81	Flo/fm-s1	main, flomon	Monthly water and N flows for a chosen hydrotope
82	Flo/fm-s2	main, flomon	Monthly water and N flows for a chosen hydrotope
83	Flo/fm-s3	main, flomon	Monthly water and N flows for a chosen hydrotope
84	Flo/fm-s4	main, flomon	Monthly water and N flows for a chosen hydrotope
85	Flo/fm-s5	main, flomon	Monthly water and N flows for a chosen hydrotope
86	Flo/fm-s6	main, flomon	Monthly water and N flows for a chosen hydrotope
87	Flo/fm-s7	main, flomon	Monthly water and N flows for a chosen hydrotope
88	Flo/fm-s8	main, flomon	Monthly water and N flows for a chosen hydrotope
89	Flo/fm-s9	main, flomon	Monthly water and N flows for a chosen hydrotope
ANNUAL WATER and N FLOWS for 9 CHOSEN SOILS/HYDROTOPES			
91	Flo/fa-s1	floann	Annual water and N flows for a chosen hydrotope
92	Flo/fa-s2	floann	Annual water and N flows for a chosen hydrotope
93	Flo/fa-s3	floann	Annual water and N flows for a chosen hydrotope
94	Flo/fa-s4	floann	Annual water and N flows for a chosen hydrotope
95	Flo/fa-s5	floann	Annual water and N flows for a chosen hydrotope
96	Flo/fa-s6	floann	Annual water and N flows for a chosen hydrotope
97	Flo/fa-s7	floann	Annual water and N flows for a chosen hydrotope
98	Flo/fa-s8	floann	Annual water and N flows for a chosen hydrotope
99	Flo/fa-s9	floann	Annual water and N flows for a chosen hydrotope
GIS OUTPUTS			
33	GIS/yld-gis.out	crop_gis	Crop yield for all hydrotopes – as GRASS input
34	GIS/wat-gis.out	crop_gis	Water stress factor for hydrotopes – as GRASS input
35	GIS/tem-gis.out	crop_gis	Temperature stress factor for hydrotopes – as GRASS input
36	GIS/pre-gis.out	hydro_gis	Annual precipitation for hydrotopes – as GRASS input

37	GIS/eva-gis.out	hydro_gis	Annual evapotranspiration for hydrotopes – as GRASS input
38	GIS/run-gis.out	hydro_gis	Annual runoff for hydrotopes – as GRASS input
39	GIS/gwr-gis.out	hydro_gis	Annual groundwater recharge for hydrotopes – as GRASS input
CROP OUTPUT			
58	Res/cryld.prn	operat	Original calculated crop yield for every year and for every hydrotope in cropland (sub-basin, soil) (considering all applied crop types)
59	Res/cryld-av.prn	main	Averaged crop yield over period a) for every soil and sub-basin, b) for every soil, and c) for the basin (considering all applied crop types)
66	Res/yld-dst.prn	cryld_brb	Distribution function for crop yield
67	Res/yldkr-1.prn	cryld_brb	Crop yield for kreise in BRB with the weighting coefficients accounting for wheat areal distribution
68	Res/yldkr-2.prn	cryld_brb	Crop yield for kreise in BRB with the weighting coefficients accounting for barley areal distribution

3.4 Input Parameters

In this Section input parameters are described. They are arranged in accordance with the input files. Almost all files listed in **Tab. 3.10** are included, except files .CIO, climate and hydrological files. The files .CIO are not included, because they are created by SWIM/GRASS interface and do not need any editing. The format of the climate and hydrological input data is to a certain extent flexible, and preparation of this data is described in Section 4.3.

3.4.1 INPUT FILE - .cod

The *xxx.cod* file includes program codes (where xxx is the basin name given when using GRASS interface). An example of the *xxx.cod* input file format is presented in the following Table:

```
PROGRAM CODES:
nbyr  iyr idaf idal   lu irch nsim msim  ign  ipd iprn iwst isst
   6 1983    1  365   64    1    4    4    5    1    0    1    1
HYDROTOPE PRINT: sub No & hyd No
isb1 ih1  isb2 ih2  isb3 ih3  isb4 ih4  isb5 ih5  isb6 ih6  isb7 ih7
   1   5     1  10     1   7    12  14     1   5     2   8     4   8
SUBBASIN PRINT: sub No
isu1  isu2  isu3
   0     0     0
PROCESS PRINT: 1/0 print/not print, sub No & hyd No
   0     1            icurn       icursb
   0     1            isolt       isosb
   0                  itran
   0     1     5      iperc       ipesb      ipehd
   0     1     5      ievap       ievsb      ievhd
   0     2     26     icrop       icrsb      icrso
   0     7            ieros       iersb
   0     1     1      inutr       inusb      inuhd
   0                  irout
   0     1     5      iswu        iwssb      iwshd
SPECIAL PRINT:
   0                  igis   - print to GIS output?
   0                  iflom  - print monthly water and N flows?
   0                  ifloa  - print annual water and N flows?
```

Parameters included in the *.cod* file are:

nbyr Number of years of simulation. It can range from 1 to 100 years.

iyr Beginning year of simulation. Usually the actual beginning year of record is used.

idaf Beginning (julian) day of simulation. Usually equal to 1.

idal Last (julian) day of simulation. Usually equal to 365 or 366.

lu Number of subbasins in the basin. Corresponds to the subbasin map specified for SWIM/GRASS interface and can be taken from .cod file created by the interface.

irch Reach of measured water and sediment yields. Usually equal to 1 (corresponds to the subbasin in the basin outlet).

nsim Code for rainfal input, if weather generator is used as climate input (clicon).
[2] = simulated single precipitation for entire basin,
[4] = simulated multiple precipitation for entire basin.

msim Code for temperature input, if weather generator is used as climate input (clicon).
[2] = simulated single temperature for entire basin,
[4] = simulated multiple temperature for entire basin.

ign number of times to cycle random number generator (used in main). The random number generator seeds are contained in the SWIM program data statements. If IGN = 0, the simulation begins with these seeds. Setting IGN>0 allows the user to start each simulation with different seeds, if desired. Each time the generators cycle, they produce a new set of seeds. This feature is convenient for simulating several different weather sequences at a particular location.

ipd Print code for general results (in genres) (0= monthly, 1=daily, 2=yearly).

iprn Print code for rin.out and wgen.out files
[0] = print subbasin parameters in the file rin.out (from readsub),
[1] = print weather parameters in the file wgen.out (from readwet).

iwst Code for stat collection on monthly water yield
[0] = to skip statistical comparison,
[1] = to calculate statistics on the simulated water yield.

isst Code for stat collection on monthly sediments yield
[0] = to skip statistical comparison,
[1] = to calculate statistics on the simulated sediment yield.

isb1, isb2, isb3, isb4 subbasin number for hydrotope output (used in *subbsin*).

isb5, isb6, isb7 subbasin number for hydrotope output (used in *vegmd*).

ih1, ih2, ih3, ih4 hydrotope number for hydrotope output (corresponding to subbasins is1, is2, is3, and is4, used in *subbsin*)

ih5, ih6, ih7 hydrotope number for hydrotope output (corresponding to subbasins is5, is6, is7, used in *vegmd*)

isu1,isu2, isu3 subbasin number for subbasin output (used in *subbasin*)

icurn code = 1/0 to print or not from the subroutines *curno* and *volq*

icursb subbasin number for printing from the subroutines *curno* and *volq*

isolt code = 1/0 to print or not from the subroutine *solt*

isosb subbasin number for printing from the subroutine *solt*

itran code = 1/0 to print or not from the subroutine *tran*

iperc code = 1/0 to print or not from the subroutine *purk()*

ipesb subbasin number for printing from the subroutine *purk()*

ipehd hydrotope number for printing from the subroutine *purk()*

ievap code = 1/0 to print or not from the subroutine *evap*

ievsb subbasin number for printing from the subroutine *evap*

ievhd hydrotope number for printing from the subroutine *evap*

icrop code = 1/0 to print or not from the subroutine *crpmd*

icrsb subbasin number for printing from the subroutine *crpmd*

icrso soil number for printing from the subroutine *crpmd*

ieros code = 1/0 to print or not from the subroutines *ecklsp(), ysed()*

iersb subbasin number for printing from the subroutines *ecklsp(), ysed()*

inutr code = 1/0 to print or not from the subroutine *ncycle()*

inusb subbasin number for printing from the subroutine *ncycle(),* and for printing water and nutrient flows from subroutines *flomon(), floann(),* and *floave()*

inuhd hydrotope number for printing from the subroutine *ncycle(),* and initial hydrotope number for printing water and nutrient flows from subroutines *flomon(), floann(),* and *floave().* Water and nutrient flows will be written for hydrotopes inuhd, inuhd+1, ... inuhd+8.

irout code = 1/0 to print or not from the subroutines *rthyd()* and *rtsed()*

iwstr code = 1/0 to print or not from the subroutine *wstress()*

iwssb subbasin number for printing from the subroutine *wstress()*

iwshd hydrotope number for printing from the subroutine *wstress()*

igis code = 1/0 to call or not *crop_gis()* and *hydro_gis()* from the subroutine *subbasin*. If igis = 1, *crop_gis()* and *hydro_gis()* will print crop yield and hydrological flows for GRASS input.

iflom code = 1/0 to call or not *flomon()* from *main*. If iflom = 1, monthly water and nutrient flows will be written for the subbasin inusb and the hydrotopes inuhd, ... , inuhd+8.

ifloa code = 1/0 to call or not *floann()* and *floave()* from *main*. If ifloa = 1, annual and average annual water and nutrient flows will be written for the subbasin inusb and the hydrotopes inuhd, ... , inuhd+8.

3.4.2 INPUT FILE - .fig

The *xxx.fig* file describes the basin routing structure (where xxx is the basin name given when using GRASS interface). An example of the *xxx.fig* input file format is presented in the following Table. The first line includes parameter names used by SWIM. The first two lines are not included in the actual files.

	icodes()	**ihouts()**	**inum1s()**	**inum2s()**	**inum3s()**	**inum4s()**
subbasin	1	1	1	1		
subbasin	1	2	2	2		
subbasin	1	3	3	3		
subbasin	1	4	4	4		
subbasin	1	5	5	5		
subbasin	1	6	6	6		
subbasin	1	7	7	7		
subbasin	1	8	8	8		
subbasin	1	9	9	9		
subbasin	1	10	10	10		
add	5	11	7	8	8	1
route	2	12	6	11	6	3
add	5	13	12	6	6	3
add	5	14	13	9	6	3
route	2	15	5	14	5	5
add	5	16	15	5	5	5
add	5	17	16	4	5	5
route	2	18	3	17	3	7
add	5	19	18	3	3	7
add	5	20	19	10	3	7
route	2	21	2	20	2	9
add	5	22	21	2	2	9
route	2	23	1	22	1	10
add	5	24	23	1	1	10
finish	0					

Parameters included in the *.fig* file are:

icodes() code to switch between routing subroutines
[1] = subbasin
[2] = route flow
[5] = add flows

ihouts() Hydrological Storage Location

inum1s() Subbasin No. (if subbasin), or Reach No. (if route), or Inflow hydrograph 1 (if add)

inum2s() Inflow Hydrograph (if route), or Inflow hydrograph 2 (if add)

inum3s() Subbasin No. (if add and route)

inum4s() Fractional Dimension (if add and route)

3.4.3 INPUT FILE - .bsn

The *xxx.bsn* file includes general basin parameters and calibration parameters (where xxx is the basin name given when using GRASS interface). An example of the *xxx.bsn* input file format is presented in the following Table.

```
SWITCH PARAMETERS
1         isc           =0/1, SC: read/calc
0         icn           =0/1, CN: dif for soils/cnum1,cnum3 for all soils
0         idlef         =0/1, day length effect in crop: without/with
0.6       thc           =0...1., evap correction on sky emissivity
BASIN, INITIALIZATION & CALIBRATION PARAMETERS
da        p2(1)     bff       brt       ffcb  Original basin parameters
574.76    1.000     1.000     0.500     0.000
cnum1     cnum2     cnum3                     Curve number, if icn=1
50.       55.       80.
gwq0      abf0                                Groundwater parameters
0.200     0.5
ekc0      prf       spcon     spexp           Erosion parameters
1.0       1.000     0.0001    1.000
snow1     storc1    stinco                    Initial water storage
0.        0.5         0.90
chwc0     chxk0     chcc0                     Channel parameters
0.700     0.05      0.0
roc1      roc2      roc3      roc4            Routing coefficients
0.        3.0       0.        13.0
sccor     prcor     rdcor                     Correction factors
1.20      1.00      1.00
```

Parameters included in the *.bsn* file are:

isc code for saturated conductivity
[0] – read from database
[1] – calculated in SWIM from clay content, sand content and porosity using the method of Brakensiek

icn code for curve number method
[0] – modified CN-method as in SWAT,
[1] – CN = const for all soil and land use categories.
In the case icn = 1 the user can set CN for conditions 1 equal to cnum1, and CN for conditions 3 equal to cnum3. The parameters cnum1 and cnum3 can be used as calibration parameters.

idlef code for taking into account day length effect on crop development
[0] – without the day length effect on crop development (as in SWAT),
[1] – with the day length effect on crop development.

thc correction factor for potential evapotranspiration on sky emissivity
[0] – without the sky emissivity factor,
[1] – with the sky emissivity factor.
The user can set also thc to intermediate values in the range 0 – 1 and use this parameter for calibration.

da basin area in km^2. This parameter should be taken from .bsn file produced by GRASS interface.

p2(1) rainfall correction factor, equal to the ratio of average rainfall to average annual rainfall for the gage in the basin outlet. When daily rainfall data is taken from a rain gauge located at a considerable distance from the basin, it may be necessary to use a rainfall correction factor other than one. If the difference between annual precipitation in the basin and rain gauge is known, a rainfall correction factor from 0.5 to 1.5 can be used.This parameter should be taken from .bsn file produced by GRASS interface. Usually is not used in SWIM.

bff baseflow factor for basin, is used to calc return flow travel time. The return flow travel time is then used to calculate percolation in soil from layer to layer. The bff factor is given in **Tab. 3.12** for different streams:

Table 3.12 Baseflow factor bff

Flow Characteristics	*BFF*
Perennial streams, flow >75% time	1.00
Flow 55-75% time	0.75
Flow 40-55% time	0.50
Flow 20-40% time	0.25
Ephemeral streams	0.00

brt basin lag time in days. Basin lag time lags the subsurface flow. For BRT= 0 all subsurface flow reaches the sub-basin outlets on the day it occurs. Judgment is required to set BRT to as many days as subsurface flow from a precipitation event is expected to contribute to streamflow.This parameter should be taken from .bsn file produced by GRASS interface. It is not used in the current model version.

ffcb fraction of field capacity as initial water storage. This parameter should be taken from .bsn file produced by GRASS interface. It is not used in the current SWIM version.

cnum1 CN, conditions 1 for the case if icn = 1.

cnum2 CN, conditions 2 for the case if icn = 1.

cnum3 CN, conditions 3 for the case if icn = 1.

gwq0 initial groundwater flow contribution to streamflow, mm/day

abf0	alpha factor for grounwater. This parameter characterizes the groundwater recession (the rate at which groundwater flow is returned to the stream).
ekc0	soil erodibility correction factor. This parameter is used to correct all values ek() of soil erodibility obtained from soil database.
prf	coefficient to estimate peak runoff in stream, used in calculation of sediment routing.
spcon	rate parameter for estimation of sediment transport (between 0.0001 and 0.01).
spexp	exponent for estimation of sediment transport (between 1. and 1.5).
snow1	initial snow content in the basin (mm).
storc1	initial water storage in streams correction coefficient.
stinco	initial water content in the basin as a fraction of field capacity.
chwc0	coefficient to correct the channel width for all reaches. The channel width is estimated by GRASS interface.
chxk0	correction coefficient for channel USLE K factor
chcc0	correction coef. for channel USLE C factor
roc1, roc2	routing coefficients to calculate the storage time constant for the reach for the surface flow, xkm, from the initial estimation phi(10) and phi(13) based on channel length and celerity (in subroutine ttcoefi(j))
roc3, roc4	routing coefficients to calculate the storage time constant for the reach for the subsurface flow, xkm, from the initial estimation phi(10) and phi(13) based on channel length and celerity (in subroutine ttcoefi(j))
sccor	correction factor for saturated conductivity (applied for all soils)
prcor	correction factor for precipitation. Usually is not used in SWIM.
rdcor	correction factor for the maximum plant root depth. Used in subroutine *readcrp*.

3.4.4 INPUT FILE - .str

The *xxx.str* file includes basin hydrotope structure parameters (where xxx is the basin name given when using GRASS interface), considering sub-basin, land use and soil. An example of the *xxx.str* input file format is presented in the following Table. The first line includes parameter names used by SWIM. The first two lines are not included in the actual files.

j	n	k	ar()	ncell
1	1	26	40000	1
1	2	12	120000	3
1	5	12	1160000	29
1	5	26	1760000	44
1	8	26	840000	21
1	9	12	1040000	26
1	9	26	80000	2
1	12	12	160000	4
1	12	26	40000	1
2	5	12	200000	5
2	5	26	320000	8
2	9	12	200000	5
2	12	12	80000	2
3	1	12	40000	1
3	2	17	40000	1
3	4	26	40000	1
3	5	12	760000	19
3	5	17	800000	20
3	5	19	480000	12
3	5	26	5400000	135
3	7	17	40000	1
3	7	26	40000	1
3	8	19	120000	3
3	9	26	840000	21
3	12	12	80000	2
4	1	26	80000	2
4	5	12	160000	4
4	5	26	680000	17
4	8	26	40000	1
4	9	26	80000	2
4	12	26	40000	1
5	1	26	320000	8
5	4	26	240000	6
5	5	19	40000	1
5	5	26	6560000	164
5	8	26	760000	19
5	9	26	2120000	53
5	12	26	160000	4
6	5	26	80000	2
6	8	26	40000	1
6	9	26	40000	1
6	12	26	40000	1

Parameters included in the *.str* file are:

j subbasin number (from SWIM/GRASS interface)

n land use category number (from SWIM/GRASS interface)
n = 1 - water
n = 2 - settlement
n = 3 - industry
n = 4 - road
n = 5 - cropland
n = 6 - set-aside
n = 7 - grassland, extensive use (meadow)
n = 8 - grassland, intensive use (pasture)
n = 9 - forest mixed
n = 10 - forest evergreen
n = 11 - forest deciduous
n = 12 - wetland nonforested
n = 13 - wetland forested
n = 14 - heather (grass + brushland)
n = 15 - bare soil

k soil type (from SWIM/GRASS interface)

ar hydrotope area, corresponding to the hydrotope with the land use n and the soil k in the subbasin j. (from SWIM/GRASS interface)

ncell number of cells, corresponding to the hydrotope with the land use n and the soil k in the subbasin j. (from SWIM/GRASS interface)

3.4.5 INPUT FILE - sub-prst.dat

The *sub-prst.dat* file establishes the correspondence between sub-basins and precipitation stations in case if only one precipitation station is used for every sub-basin. It lists all sub-basins and the corresponding precipitation stations numbered from 1 to nst, where nst is the total number of precipitation stations used.

is1() subbasin No.

ip2() precip. stat No., corresponding to the subbasin is1() (chosen as the closest the the subbasin central point)

3.4.6 INPUT FILE - crop.dat

The *crop.dat* file provides crop/vegetation parameters for 71 crops/vegetation types, including some aggregated types of vegetation, like deciduous forest. This is the standard file. It is presented in **Tab. 3.13**. Then **Tab. 3.14** lists abbreviated crop names, full crop names, and land cover categories.

Table 3.13 Crop data base (file crop.dat)

icnum	cpnm	ird	be	hi	to	tb	blai	dlai	dlp1	dlp2	bn1	bn2	bn3	bp1	bp2	bp3	cnyld	cpyld	rdmx	cvm	almn	sla	pt2	hun	cpnm
1	**AGRL**	1	40.0	0.50	25.0	8.0	5.0	0.80	15.05	50.95	0.0440	0.0164	0.0128	0.0062	0.0023	0.0018	0.0175	0.0025	2.0	0.200	0.0	0.0	660.44	1500	**AGRL**
2	**ASPR**	1	90.0	0.80	35.0	10.0	4.2	1.00	25.23	40.86	0.0620	0.0500	0.0400	0.0050	0.0040	0.0020	0.0700	0.0065	1.5	0.200	0.0	0.0	660.95	1500	**ASPR**
3	**BROC**	1	26.0	0.80	24.0	4.0	4.2	1.00	25.23	40.86	0.0620	0.0090	0.0070	0.0050	0.0040	0.0030	0.0530	0.0073	0.7	0.200	0.0	0.0	660.95	1500	**BROC**
4	**CABG**	1	19.0	0.80	24.0	4.0	3.0	1.00	25.23	40.86	0.0620	0.0070	0.0040	0.0050	0.0035	0.0020	0.0270	0.0033	0.7	0.200	0.0	0.0	660.95	1500	**CABG**
5	**CANT**	1	30.0	0.50	32.0	16.0	3.0	0.60	15.05	50.95	0.0663	0.0255	0.0148	0.0053	0.0020	0.0012	0.0250	0.0022	1.3	0.030	0.0	0.0	660.39	1500	**CANT**
6	**CAUF**	1	21.0	0.80	24.0	7.0	2.5	1.00	25.23	40.86	0.0620	0.0070	0.0040	0.0050	0.0035	0.0020	0.0400	0.0057	0.7	0.200	0.0	0.0	660.95	1500	**CAUF**
7	**CELR**	1	27.0	0.80	24.0	7.0	2.5	1.00	25.23	40.86	0.0620	0.0150	0.0100	0.0060	0.0050	0.0030	0.0220	0.0052	0.7	0.200	0.0	0.0	660.95	1500	**CELR**
8	**CORN**	1	40.0	0.50	25.0	8.0	5.0	0.80	15.05	50.95	0.0440	0.0164	0.0128	0.0062	0.0023	0.0018	0.0175	0.0025	2.0	0.200	0.0	0.0	660.44	1500	**CORN**
9	**CORN**	1	39.0	0.55	35.0	8.0	3.0	0.50	15.05	50.95	0.0470	0.0177	0.0138	0.0048	0.0018	0.0014	0.0210	0.0070	2.0	0.200	0.0	0.0	660.44	1500	**CORN**
10	**COTP**	1	15.0	0.40	27.5	10.0	4.0	0.95	15.01	50.95	0.0580	0.0192	0.0177	0.0081	0.0027	0.0025	0.0190	0.0029	2.2	0.200	0.0	0.0	660.19	1500	**COTP**
11	**COTS**	1	15.0	0.50	27.5	10.0	4.0	0.95	15.01	50.95	0.0580	0.0192	0.0177	0.0081	0.0027	0.0025	0.0140	0.0020	2.2	0.200	0.0	0.0	660.19	1500	**COTS**
12	**CRRT**	1	30.0	1.12	24.0	7.0	3.5	0.60	15.01	50.95	0.0550	0.0075	0.0012	0.0060	0.0030	0.0020	0.0130	0.0037	1.1	0.200	0.0	0.0	660.20	1500	**CRRT**
13	**CUCM**	1	30.0	0.27	32.0	16.0	1.5	0.60	15.05	50.95	0.0663	0.0075	0.0048	0.0053	0.0025	0.0012	0.0200	0.0042	1.1	0.030	0.0	0.0	660.39	1500	**CUCM**
14	**EGGP**	1	30.0	0.59	35.0	18.0	3.0	0.60	15.05	50.95	0.0663	0.0255	0.0075	0.0053	0.0020	0.0015	0.0220	0.0041	1.1	0.030	0.0	0.0	660.39	1500	**EGGP**
15	**GRSG**	1	35.0	0.50	27.5	10.0	5.0	0.80	15.05	50.95	0.0440	0.0164	0.0128	0.0060	0.0022	0.0018	0.0200	0.0028	2.0	0.200	0.0	0.0	660.38	1500	**GRSG**
16	**HMEL**	1	30.0	0.55	35.0	16.0	4.0	0.60	15.05	50.95	0.0070	0.0040	0.0020	0.0026	0.0020	0.0017	0.0080	0.0010	1.1	0.030	0.0	0.0	660.39	1500	**HMEL**
17	**ONIO**	1	30.0	1.25	29.0	7.0	1.5	0.60	15.01	50.95	0.0400	0.0300	0.0020	0.0021	0.0020	0.0019	0.0210	0.0032	0.7	0.200	0.0	0.0	660.20	1500	**ONIO**
18	**PEPR**	1	30.0	0.60	27.0	18.0	5.0	0.60	15.05	50.95	0.0600	0.0350	0.0250	0.0053	0.0020	0.0012	0.0030	0.0020	1.1	0.030	0.0	0.0	660.39	1500	**PEPR**
19	**POTA**	1	30.0	0.95	18.0	7.0	5.0	0.60	15.01	50.95	0.0550	0.0200	0.0120	0.0060	0.0025	0.0019	0.0130	0.0020	2.0	0.200	0.0	0.0	660.20	1500	**POTA**
20	**POTA**	1	30.0	1.41	18.0	3.0	5.0	0.95	15.01	50.95	0.0550	0.0200	0.0120	0.0060	0.0025	0.0019	0.0130	0.0020	2.0	0.200	0.0	0.0	660.20	2000	**POTA**
21	**RICE**	1	25.0	0.50	25.0	10.0	6.0	0.80	30.01	70.95	0.0500	0.0200	0.0100	0.0060	0.0030	0.0018	0.0200	0.0030	0.9	0.030	0.0	0.0	660.31	1500	**RICE**
22	**SBAR**	1	30.0	0.42	15.0	0.0	6.0	0.90	15.01	50.95	0.0600	0.0231	0.0134	0.0084	0.0032	0.0019	0.0234	0.0033	2.0	0.010	0.0	0.0	660.45	1900	**SBAR**
23	**SGBT**	1	30.0	2.00	18.0	4.0	5.0	0.60	5.05	50.95	0.0550	0.0200	0.0120	0.0060	0.0025	0.0019	0.0130	0.0020	2.0	0.200	0.0	0.0	660.20	1500	**SGBT**
24	**SGHY**	1	35.0	0.50	27.5	10.0	5.0	0.85	15.01	50.95	0.0440	0.0164	0.0128	0.0060	0.0022	0.0018	0.0200	0.0028	2.0	0.030	0.0	0.0	660.38	1500	**SGHY**
25	**SLMA**	1	60.0	1.00	20.0	5.0	8.0	0.70	15.05	50.95	0.0440	0.0164	0.0128	0.0062	0.0023	0.0018	0.0175	0.0025	1.0	0.200	0.0	0.0	660.44	2400	**SLMA**
26	**SPIN**	1	30.0	0.95	24.0	4.0	4.2	0.95	10.05	90.95	0.0620	0.0400	0.0300	0.0050	0.0040	0.0035	0.0580	0.0061	0.7	0.200	0.0	0.0	660.95	1500	**SPIN**
27	**STRW**	1	30.0	0.45	32.0	10.0	3.0	0.60	15.05	50.95	0.0663	0.0255	0.0148	0.0053	0.0020	0.0012	0.0120	0.0024	0.7	0.030	0.0	0.0	660.39	1500	**STRW**
28	**SUGC**	1	25.0	0.50	25.0	11.0	6.0	0.75	15.01	50.95	0.0100	0.0040	0.0025	0.0075	0.0030	0.0019	0.0000	0.0000	2.0	0.001	0.0	0.0	660.40	1500	**SUGC**
29	**SUNF**	1	35.0	0.40	25.0	6.0	5.0	0.55	15.01	50.95	0.0500	0.0230	0.0146	0.0063	0.0029	0.0023	0.0280	0.0061	2.2	0.200	0.0	0.0	660.45	1500	**SUNF**
30	**SWHT**	1	30.0	0.40	15.0	0.0	6.0	0.80	15.01	50.95	0.0600	0.0231	0.0130	0.0084	0.0032	0.0019	0.0234	0.0033	2.0	0.030	0.0	0.0	660.39	1500	**SWHT**
31	**TOBC**	1	39.0	0.55	25.0	8.0	4.5	0.70	15.05	50.95	0.0470	0.0177	0.0138	0.0048	0.0018	0.0014	0.0140	0.0016	2.0	0.200	0.0	0.0	660.44	1500	**TOBC**
32	**TOMA**	1	30.0	0.33	27.0	10.0	3.0	0.95	15.05	50.95	0.0663	0.0300	0.0250	0.0053	0.0035	0.0025	0.0240	0.0038	1.5	0.030	0.0	0.0	660.39	1500	**TOMA**
33	**WMEL**	1	30.0	0.50	35.0	18.0	1.5	0.60	15.05	50.95	0.0663	0.0075	0.0048	0.0053	0.0025	0.0012	0.0110	0.0014	1.1	0.030	0.0	0.0	660.39	1500	**WMEL**
icnum	**cpnm**	**ird**	**be**	**hi**	**to**	**tb**	**blai**	**dlai**	**dlp1**	**dlp2**	**bn1**	**bn2**	**bn3**	**bp1**	**bp2**	**bp3**	**cnyld**	**cpyld**	**rdmx**	**cvm**	**almn**	**sla**	**pt2**	**hun**	**cpnm**

Tab. 3.13, continued

icnum	cpnm	ird	be	hi	to	tb	blai	dlai	dlp1	dlp2	bn1	bn2	bn3	bp1	bp2	bp3	cnyld	cpyld	rdmx	cvm	almn	sla	pt2	hun	cpnm
34	**BARL**	2	30.0	0.40	15.0	0.0	6.0	0.80	15.01	50.95	0.0600	0.0231	0.0134	0.0084	0.0032	0.0019	0.0234	0.0033	2.0	0.030	0.0	0.0	660.45	1500	**BARL**
35	**BARL**	2	35.0	0.42	25.0	0.0	3.5	0.60	20.10	49.95	0.0590	0.0226	0.0131	0.0057	0.0022	0.0013	0.0210	0.0017	2.0	0.010	0.0	0.0	660.39	1500	**BARL**
36	**BARL**	2	30.0	0.42	15.0	0.0	6.0	0.90	15.01	50.95	0.0600	0.0231	0.0134	0.0084	0.0032	0.0019	0.0234	0.0033	2.0	0.010	0.0	0.0	660.45	2300	**BARL**
37	**LETT**	2	23.0	0.80	18.2	0.0	4.2	1.00	25.23	40.86	0.0360	0.0250	0.0210	0.0084	0.0032	0.0019	0.0260	0.0049	0.8	0.010	0.0	0.0	660.25	1500	**LETT**
38	**LETL**	2	19.0	0.80	18.2	0.0	4.2	1.00	25.23	40.86	0.0360	0.0250	0.0210	0.0084	0.0032	0.0019	0.0260	0.0049	0.8	0.010	0.0	0.0	660.25	1500	**LETL**
39	**OATS**	2	30.0	0.35	15.0	0.0	6.0	0.80	15.01	50.95	0.0600	0.0231	0.0134	0.0084	0.0032	0.0019	0.0234	0.0033	2.0	0.030	0.0	0.0	660.45	1500	**OATS**
40	**RAPE**	2	35.0	0.30	14.0	0.0	4.5	0.60	15.01	50.95	0.0500	0.0200	0.0110	0.0070	0.0025	0.0015	0.0350	0.0067	2.0	0.050	0.0	0.0	660.40	1500	**RAPE**
41	**RYE**	2	35.0	0.40	12.5	0.0	6.0	0.80	15.01	50.95	0.0600	0.0231	0.0134	0.0084	0.0032	0.0019	0.0234	0.0033	2.0	0.030	0.0	0.0	660.45	1500	**RYE**
42	**RYE**	2	35.0	0.40	12.5	0.0	6.0	0.80	15.01	50.95	0.0600	0.0231	0.0134	0.0084	0.0032	0.0019	0.0234	0.0033	2.0	0.300	0.0	0.0	660.45	2000	**RYE**
43	**WHTD**	2	30.0	0.40	15.0	0.0	6.0	0.80	15.01	50.95	0.0600	0.0231	0.0130	0.0084	0.0032	0.0019	0.0209	0.0050	2.0	0.030	0.0	0.0	660.45	1500	**WHTD**
44	**WWHT**	2	30.0	0.40	15.0	0.0	6.0	0.60	15.01	50.95	0.0600	0.0231	0.0134	0.0084	0.0032	0.0019	0.0234	0.0033	2.0	0.030	0.0	0.0	660.39	1500	**WWHT**
45	**WWHT**	2	30.0	0.42	15.0	0.0	6.0	0.80	15.01	50.95	0.0600	0.0231	0.0134	0.0084	0.0032	0.0019	0.0234	0.0033	2.0	0.030	0.0	0.0	660.39	2300	**WWHT**
46	**HAY**	3	35.0	0.01	25.0	12.0	5.0	0.99	15.01	50.95	0.0600	0.0231	0.0134	0.0084	0.0032	0.0019	0.0234	0.0033	2.0	0.003	0.0	0.0	660.38	1500	**HAY**
47	**PAST**	3	35.0	0.01	25.0	12.0	5.0	0.99	15.01	50.95	0.0600	0.0231	0.0134	0.0084	0.0032	0.0019	0.0234	0.0033	2.0	0.003	0.0	0.0	660.38	1500	**PAST**
48	**SPAS**	3	35.0	0.01	25.0	8.0	5.0	0.99	15.01	50.95	0.0600	0.0231	0.0134	0.0084	0.0032	0.0019	0.0234	0.0033	2.0	0.003	0.0	0.0	660.38	1500	**SPAS**
49	**WPAS**	3	30.0	0.01	15.0	0.0	5.0	0.99	15.01	50.95	0.0600	0.0231	0.0134	0.0084	0.0032	0.0019	0.0234	0.0033	2.0	0.003	0.0	0.0	660.39	1500	**WPAS**
50	**RNGB**	3	30.0	0.01	25.0	8.0	5.0	0.99	15.01	50.95	0.0600	0.0231	0.0134	0.0084	0.0032	0.0019	0.0234	0.0033	2.0	0.003	0.0	0.0	660.33	1500	**RNGB**
51	**COVC**	3	35.0	0.01	25.0	8.0	5.0	0.99	15.01	50.95	0.0600	0.0231	0.0134	0.0084	0.0032	0.0019	0.0234	0.0033	2.0	0.003	0.0	0.0	660.38	1500	**COVC**
52	**URBN**	3	8.0	0.01	25.0	8.0	4.0	0.99	15.01	50.95	0.0600	0.0231	0.0134	0.0084	0.0032	0.0019	0.0234	0.0033	2.0	0.003	0.0	0.0	660.33	1500	**URBN**
53	**WATR**	3	0.0	0.01	25.0	8.0	0.0	0.99	15.01	50.95	0.0600	0.0231	0.0134	0.0084	0.0032	0.0019	0.0234	0.0033	2.0	0.003	0.0	0.0	660.33	1500	**WATR**
54	**WETL**	3	30.0	0.01	25.0	8.0	3.0	0.99	15.01	50.95	0.0600	0.0231	0.0134	0.0084	0.0032	0.0019	0.0234	0.0033	2.0	0.003	0.0	0.0	660.33	1500	**WETL**
55	**WETN**	3	30.0	0.01	25.0	8.0	3.0	0.99	15.01	50.95	0.0600	0.0231	0.0134	0.0084	0.0032	0.0019	0.0234	0.0033	2.0	0.003	0.0	0.0	660.33	1500	**WETN**
56	**FRSD**	4	16.0	0.01	20.0	6.0	2.0	0.85	10.50	25.99	0.0060	0.0020	0.0015	0.0007	0.0004	0.0003	0.0015	0.0003	2.0	0.001	0.0	18.0	660.20	1500	**FRSD**
57	**FRSE**	4	16.0	0.01	20.0	2.0	6.0	0.85	10.50	25.99	0.0060	0.0020	0.0015	0.0007	0.0004	0.0003	0.0015	0.0003	2.0	0.001	1.8	4.5	660.20	1500	**FRSE**
58	**FRST**	4	16.0	0.01	20.0	2.0	5.0	0.85	10.50	25.99	0.0060	0.0020	0.0015	0.0007	0.0004	0.0003	0.0015	0.0003	2.0	0.001	1.2	11.0	660.20	1500	**FRST**
59	**FORD**	4	16.0	0.01	20.0	2.0	5.0	0.85	10.50	25.99	0.0060	0.0020	0.0015	0.0007	0.0004	0.0003	0.0015	0.0003	2.0	0.001	0.0	18.0	660.20	2500	**FORD**
60	**FORE**	4	16.0	0.01	20.0	2.0	3.0	0.85	10.50	25.99	0.0060	0.0020	0.0015	0.0007	0.0004	0.0003	0.0015	0.0003	2.0	0.001	1.8	4.5	660.20	2500	**FORE**
61	**FORM**	4	16.0	0.01	20.0	2.0	4.0	0.85	10.50	25.99	0.0060	0.0020	0.0015	0.0007	0.0004	0.0003	0.0015	0.0003	2.0	0.001	1.2	11.0	660.20	2500	**FORM**
62	**PINE**	4	16.0	0.75	20.0	2.0	5.0	0.85	10.50	25.99	0.0060	0.0020	0.0015	0.0007	0.0004	0.0003	0.0015	0.0003	2.0	0.001	1.8	4.5	660.20	1500	**PINE**
63	**WETF**	4	30.0	0.01	25.0	8.0	3.0	0.99	15.01	50.95	0.0600	0.0231	0.0134	0.0084	0.0032	0.0019	0.0234	0.0033	2.0	0.003	1.2	11.0	660.33	1500	**WETF**
64	**GRBN**	5	25.0	0.10	27.0	10.0	1.5	0.90	10.05	80.95	0.0040	0.0030	0.0015	0.0040	0.0035	0.0015	0.0290	0.0038	1.1	0.200	0.0	0.0	660.34	1500	**GRBN**
65	**LIMA**	5	25.0	0.30	27.0	10.0	2.5	0.90	10.05	80.95	0.0040	0.0030	0.0015	0.0035	0.0030	0.0015	0.0360	0.0045	1.5	0.200	0.0	0.0	660.34	1500	**LIMA**
66	**PEAS**	5	25.0	0.30	27.0	7.0	2.5	0.60	10.05	80.95	0.0040	0.0030	0.0015	0.0030	0.0020	0.0015	0.0410	0.0051	1.1	0.200	0.0	0.0	660.34	1500	**PEAS**
67	**PNUT**	5	20.0	0.40	25.0	13.5	5.0	0.75	15.01	50.95	0.0524	0.0265	0.0258	0.0074	0.0037	0.0035	0.0650	0.0091	2.0	0.200	0.0	0.0	660.25	1500	**PNUT**
68	**SOYB**	5	25.0	0.30	25.0	10.0	5.0	0.90	15.01	50.95	0.0524	0.0265	0.0258	0.0074	0.0037	0.0035	0.0650	0.0091	2.0	0.200	0.0	0.0	660.31	1500	**SOYB**
69	**LEN1**	6	20.0	0.55	14.0	1.0	4.0	0.90	15.02	50.95	0.0524	0.0320	0.0286	0.0074	0.0037	0.0035	0.0400	0.0050	2.0	0.200	0.0	0.0	660.25	1500	**LEN1**
70	**WPEA**	6	20.0	0.55	14.0	1.0	4.0	0.90	15.02	50.95	0.0400	0.0260	0.0232	0.0070	0.0040	0.0030	0.0380	0.0050	2.0	0.050	0.0	0.0	660.25	1500	**WPEA**
71	**ALFA**	7	20.0	0.01	15.0	1.0	5.0	0.90	15.01	50.95	0.0500	0.0300	0.0200	0.0071	0.0042	0.0028	0.0250	0.0035	2.0	0.010	0.0	0.0	660.25	1500	**ALFA**

Table 3.14 Crop abbreviated names, full names and the corresponding land cover categories

Crop number	Abbreviated crop name	***Crop seasonality identificator***	Crop name	Land cover category
		Annual crops		
1	AGRL	1	agricultural land	row crop
2	ASPR	1	asparagus	row crop
3	BROC	1	broccoli	row crop
4	CABG	1	cabbage	row crop
5	CANT	1	cantaloupe	row crop
6	CAUF	1	cauliflower	row crop
7	CELR	1	celery	row crop
8	CORN	1	corn for grain	row crop
9	CORN	1	corn for grain	row crop
10	COTP	1	cotton, stripped	row crop
11	COTS	1	cotton, picked	row crop
12	CRRT	1	carrot	row crop
13	CUCM	1	cucumber	row crop
14	EGGP	1	eggplant	row crop
15	GRSG	1	sorghum	row crop
16	HMEL	1	honey melon	row crop
17	ONIO	1	onion	row crop
18	PEPR	1	pepper	row crop
19	POTA	1	potatoes	row crop
20	POTA	1	potatoes	row crop
21	RICE	1	rice	small grain
22	SBAR	1	spring barley	small grain
23	SGBT	1	sugar beet	row crop
24	SGHY	1	sorghum hay	row crop
25	SLMA	1	silage maize	row crop
26	SPIN	1	spinach	row crop
27	STRW	1	strawberries	row crop
28	SUGC	1	sugarcane	row crop
29	SUNF	1	sunflower	row crop
30	SWHT	1	spring wheat	small grain
31	TOBC	1	tobacco	row crop
32	TOMA	1	tomato	row crop
33	WMEL	1	water melon	row crop
		Annual winter crops		
34	BARL	2	barley	small grain
35	BARL	2	barley	small grain
36	BARL	2	winter barley	small grain
37	LETT	2	lettuce	row crop
38	LETL	2	lettuce, leaf	row crop
39	OATS	2	oats	small grain
40	RAPE	2	rape	row crop
41	RYE	2	rye	small grain
42	RYE	2	winter rye	small grain
43	WHTD	2	durum wheat	small grain
44	WWHT	2	winter wheat	small grain
45	WWHT	2	winter wheat	small grain

		Perennial land cover (non-forested)		
46	HAY	3	hay	perennial grass
47	PAST	3	pasture	perennial grass
48	SPAS	3	summer pasture	perennial grass
49	WPAS	3	winter pasture	perennial grass
50	RNGB	3	range-brush	brush
51	COVC	3	cover crop	grass
52	URBN	3	urban	urban
53	WATR	3	water	water
54	WETL	3	wetland	perennial grass
55	WETN	3	wetland nonforested	perennial grass
		Woods		
56	FRSD	4	forest deciduous	woods
57	FRSE	4	forest evergreen	woods
58	FRST	4	forest	woods
59	FORD	4	forest deciduous	woods
60	FORE	4	forest evergreen	woods
61	FORM	4	forest mixed	woods
62	PINE	4	pine	woods
63	WETF	4	wetland forested	woods
		Annual legumes		
64	GRBN	5	green beans	row crop
65	LIMA	5	lima	row crop
66	PEAS	5	peas	row crop
67	PNUT	5	peanut	row crop
68	SOYB	5	soybean	row crop
		Annual winter legumes		
69	LEN1	6	lentil	row crop
70	WPEA	6	winter peas	row crop
		Perennial legumes		
71	ALFA	7	alfalfa	perennial grass

Parameters included in the *crop.dat* file are:

icnum(ic) crop number. This is a reference number only

ic crop number in the database (file crop.dat)

cpnm(ic) crop name. This is a four character name to represent the crop.

ilcc(ic) land cover category:
[1] annual crop (row crop / small grain)
[2] annual winter crop (row crop / small grain)
[3] perennial (grass, brush, urban, water)
[4] woods
[5] annual legumes (row crop)
[6] annual winter legumes (row crop)
[7] perennial legumes (grass)

be(ic) biomass-energy ratio. This is the potential (unstressed) growth rate (including roots) per unit of intercepted photosynthetically active radiation. This parameter should be one of the last to be adjusted. Adjustments should be based on research results. This parameter can greatly change the rate of growth, incidence of stress during the season and the resultant crop yield. Care should be taken to make adjustments in the parameter only based on data with no drought, nutrient or temperature stress.

hi(ic) harvest index. This crop parameter is based on experimental data where crop stresses have been minimized to allow the crop to attain its potential. The model adjusts hi() as water stress occurs in the period near flowering.

to(ic) optimal temperature for plant growth, degrees C. This parameter is very stable for cultivars within a species. It should not be changed once it is determined for a species. Varietal or maturity type differences are accounted for by different sums of heat units.

tb(ic) base temperature for plant growth, degrees C. This parameter is very stable for cultivars within a species. It should not be changed once it are determined for a species. Varietal or maturity type differences are accounted for by different sums of heat units.

blai(ic) Maximum potential leaf area index. BLAI may need to be adjusted for drought-prone regions where planting densities are much smaller or irrigated conditions where densities are much greater.

dlai(ic) Fraction of growing season when leaf area declines. The fraction of the growing season in heat units in divided by the total heat units accumulated between planting and crop maturity. If the date at which leaf area normally declines is known, one of the drought options in SWIM can be used to estimate the fraction of heat units accumulated. The estimated heat units at maximum leaf area can then be divided by the heat units at maturity to estimate the fraction of the growing season at which leaf-area-index start to decline.

dlp1(ic) complex number: before decimal: fraction of growing season, after decimal: max corresponding LAI. First point on optimal leaf area development curve.

dlp2(ic) complex number: before decimal: fraction of growing season, after decimal: max corresponding LAI. Second point on optimal leaf area development curve. Explanation: Two points on optimal (non-stress) leaf area development curve. Numbers before decimal are % of growing season. Numbers after decimal are fractions of maximum potential LAI. These two points are based on research results on the % of maximum leaf area at two points in the development of leaf area.

bn1(ic) nitrogen uptake parameter #1: normal fraction of N in crop biomass at emergence, kg N/kg biomass.

bn2(ic) nitrogen uptake parameter #2: normal fraction of N in crop biomass at 0.5 maturity, kg N/kg biomass.

bn3(ic) nitrogen uptake parameter #3: normal fraction of N in crop biomass at maturity, kg N/kg biomass.

bp1(ic) phosphorus uptake parameter #1: normal fraction of P in crop biomass at emergence, kg P/kg biomass.

bp2(ic) phosphorus uptake parameter #2: normal fraction of P in crop biomass at 0.5 maturity, kg P/kg biomass.

bp3(ic) phosphorus uptake parameter #3: normal fraction of P in crop biomass at maturity, kg P/kg biomass.

cnyld(ic) fraction of nitrogen in crop yield, kg N/kg yield.

cpyld(ic) fraction of phosphorus in crop yield, kg P/kg yield.

rdmx(ic) maximum plant rooting depth (m)

cvm(ic) minimum value of C factor for water erosion. This parameter should be adjusted with intercropping or no-tillage simulations to reflect the added cover.

almn(ic) LAI minimum (for forest and natural perennial vegetation).

sla(ic) specific leaf area (m^2/kg). It is in SWIM used for forest and natural perennial vegetation.

pt2(ic) 2nd point on radiation use efficiency curve: complex number: The value to the left of the decimal is a CO^2 atmospheric concentration higher than the ambient (in units of microliters CO^2/liter air, i.e. 450 or 660). The value to the right of the decimal is the corresponding biomass-energy ratio divided by 100. Typical values of the ratio are 1.1 to 1.2 for C4 crops and 1.3 to 1.4 for C3 crops. (Kimball, B.A. 1983).

3.4.7 INPUT FILE - wgen.dat

The *wgen.dat* file includes monthly statistical weather parameter for basin or sub-basins. An example of the *wgen.dat* input file is presented in the following Table. The lines 3 – 13 include 12 monthly values.

```
 SYTHETIC WEATHER DATA
 45.000  66.000  54.000  31.770
 2.28  2.09  6.13 12.10 18.09 19.70 22.28 21.85 17.36 13.16  6.42  4.16
-2.33 -3.22   .10  3.65  7.92 10.25 12.66 12.42  9.84  6.73  1.76   .15
 0.18  0.14  0.11  0.10  0.09  0.08  0.07  0.07  0.08  0.10  0.13  0.15
56.  104.  213.  267.  512.  471.  417.  351.  282.  152.   74.   42.
 7.    7.    9.    9.   10.   10.   10.   10.    9.    9.    7.    7.
 0.37  0.34  0.35  0.33  0.32  0.29  0.35  0.41  0.51  0.55  0.56  0.38
 0.69  0.67  0.69  0.64  0.64  0.61  0.65  0.70  0.72  0.78  0.79  0.4577
 0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00
 2.7   2.6   2.7   3.2   4.2   5.0   4.9   5.0   3.7   3.2   3.0   3.0
 3.3   3.05  2.54  2.79  2.54  3.30  5.08  5.84  4.57  4.57  3.81  3.05
 2.86  4.07  3.49  2.19  1.70  2.02  2.51  1.84  2.00  2.90  2.14  2.04
```

The next Table describes format of the example file *wgen.dat*, including parameter names given above the values.

```
 SYTHETIC WEATHER DATA
tp5     tp6     tp24    ylt
45.000  66.000  54.000  31.770
obmx(mo,j)
 2.28  2.09  6.13 12.10 18.09 19.70 22.28 21.85 17.36 13.16  6.42  4.16
obmn(mo,j)
-2.33 -3.22   .10  3.65  7.92 10.25 12.66 12.42  9.84  6.73  1.76   .15
cvt(mo,j)
 0.18  0.14  0.11  0.10  0.09  0.08  0.07  0.07  0.08  0.10  0.13  0.15
obsl(mo,j)
56.  104.  213.  267.  512.  471.  417.  351.  282.  152.   74.   42.
wim(mo)
 7.    7.    9.    9.   10.   10.   10.   10.    9.    9.    7.    7.
prw(1,mo,j)
 0.37  0.34  0.35  0.33  0.32  0.29  0.35  0.41  0.51  0.55  0.56  0.38
prw(2,mo,j)
 0.69  0.67  0.69  0.64  0.64  0.61  0.65  0.70  0.72  0.78  0.79  0.4577
wvl(mo,j)
 0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00
rst(mo,1,i)
 2.7   2.6   2.7   3.2   4.2   5.0   4.9   5.0   3.7   3.2   3.0   3.0
rst(mo,2,j)
 3.3   3.05  2.54  2.79  2.54  3.30  5.08  5.84  4.57  4.57  3.81  3.05
rst(mo,3,j)
 2.86  4.07  3.49  2.19  1.70  2.02  2.51  1.84  2.00  2.90  2.14  2.04
```

Parameters included in the *wgen.dat* file are:

tp5(j) 10 year frequency of 0.5 h rainfall (mm). Maybe basin- or subbasin-specific.

tp6(j) 10 year frequency of 6.0 h rainfall (mm). Maybe basin- or subbasin-specific.

tp24(j) Number of years of records for estimating max 0.5 h rainfall. Maybe basin- or subbasin-specific.

ylt(j) latitude of the basin or subbasin centre

obmx(mo,j) average monthly maximum air temperature, degree C.

onmn(mo,j) average monthly minimum air temperature, degree C.

cvt(mo,j) coefficient of variation for monthly temperature

obsl(mo,j) monthly average of daily solar radiation (ly)

wim monthly maximum 0.5 h rainfall (mm) for the period of record

prw(1,mo,j) monthly probability of wet day after dry day.

prw(2,mo,j) monthly probability of wet day after dry day.

wvl(mo,j) monthly number days of precipitation in a month.

rst(mo,1,j) monthly mean event of daily rainfall.

rst(mo,2,,j) monthly standard deviation of daily rainfall.

rst(mo,3,,j) monthly skew coefficient of daily rainfall.

3.4.8 INPUT FILE - .sub

The *xxxNN.sub* file includes sub-basin parameters (where xxx is the basin name given when using GRASS interface, NN is the sub-basin number -1). An example of the *xxxNN.sub* input file format is presented in the following Table.

```
SUBBasin DATA 15 16
   0.006   0.000   0.150   0.000   3.785   0.002   2.494   0.370   0.075   0.150
   0.000 500.000   0.500 104.573   0.007
```

The next Table describes format of the example file *xxxNN.sub* , including parameter names given above the values.

```
SUBBasin DATA 15 16
   flu(j)  dum    salb(j)  sno(j) chl(1,j) chs(j) chw(1,j) chk(1,j) chn(j) ovn(j)
   0.006   0.000   0.150   0.000   3.785   0.002   2.494   0.370   0.075   0.150
   rt(j)  css(j)   ecp(j) sl(j)    stp(j)
   0.000 500.000   0.500 104.573   0.007
```

Parameters included in the *xxxNN.sub* file are:

flu(j) fraction of subbasin area in the whole basin

salb(j) soil albedo.

sno(j) initial water content of snow (mm). For long-term simulations the water content of snow on the ground at the beginning of simulation, SNO is usually not known, but in most cases the estimate is not critical after the first year. If a measured value of SNO is available at the beginning of the simulation, it should be used.

chl(1,j) main channel length (km). The channel length is the distance along the channel from the sub-basin outlet to the most distant point in the sub-basin. It is estimated from a topographic map by GRASS interface and used to calculate the sub-basin time of concentration.

chs(j) main channel slope (m/m). The average channel slope is computed by dividing the difference in elevation between the sub-basin outlet and the most distant point in the sub-basin. It is estimated from a topographic map by GRASS interface and used to calculate the sub-basin time of concentration.

chw(1,j) average width of main channel (m). It is estimated from a topographic map by GRASS interface. Not used by the current model version.

chk(1,j) effective hydraulic conductivity in channel alluvium (mm/hr). It is set to a default value of 0.37 by GRASS Interface. Not used by the current model version.

chn(j) Channel N value. The channel "n" value is Mannings's "n" value. It is set to a default value 0.075 by GRASS interface and used to calculate the sub-basin time of concentration. This value may be corrected by user.

ovn(j) Overland flow N value. The surface roughness factor is Manning's "n" values. It is set to a default value 0.150 by GRASS interface and used to calculate the sub-basin time of concentration. This value may be corrected by user. The following **Tab. 3.15** contains suggested values of Manning's "n" for various condition.

Table 3.15 Values of Manning's "n" for overland flow

Overland Flow	Value Chosen	Range
Fallow, no residue	0.0100	0.008-0.012
Conventional tillage, no residue	0.0900	0.06-0.12
Conventional tillage, residue	0.1900	0.16-0.22
Chisel plow, no residue	0.0900	0.06-0.12
Chisel plow, residue	0.1300	0.10-0.16
Fall disking, residue	0.4000	0.30-0.50
No-till, no residue	0.0700	0.04-0.10
No-till, (0.5-1.0 t/ha)	0.1200	0.07-0.17
No-till (2.0-9.0 t/ha)	0.3000	0.17-0.47
Rangeland (20% cover)	0.6000	0.40-0.70
Short grass prairie	0.1500	0.10-0.20
Dense grass	0.2400	0.17-0.30
Bermuda grass	0.4100	0.30-0.48

rt(j) Return flow travel time(days). Return flow travel time is required for subsurface flow from the centroid of the sub-basin to reach the sub-basin outlet. The value of RT is input for each sub-basin by the SWIM user, or can be calculated by SWIM if rt=0.0 from soil hydraulic properties and flow characteristics. Experienced hydrologists familiar with the base flow characteristics of watersheds within a region should have little problem assigning reasonable values to RT. However, if the user is not familiar with the watershed, SWIM will estimate RT based on the soil's saturated conductivity and a parameter called the baseflow factor.

css(j) Sediment concentration in return flow (ppm). Sediment concentration in return flow is usually very low and does not contribute significantly to total sediment yields unless return flow is very high. Unless the user is aware of unusual situations where CSS is extremely high, a value of [500.] ppm is a good estimate and will yield realistic results.

ecp(j) USLE erosion control practice factor P. Values of the USLE erosion control practice factor provided by Wischmeier and Smith are contained in **Tab. 3.16**.

Table 3.16 Water erosion control practice factor P and slope-length limits for contouring

Land Slope (%)	P Value	Max length* (ft.)
1 to 2	0.60	400
3 to 5	0.50	300
6 to 8	0.50	200
9 to 12	0.60	120
13 to 16	0.70	80
17 to 20	0.80	60
21 to 25	0.90	50

* Limit may be increased by 25% if residue cover after crop seedlings will regularly exceed 50%.

sl(j) Average slope length (m). The average slope length is estimated for each sub-basin with the Contour-Extreme Point Method (Williams and Berndt, 1977).

stp(j) Average slope steepness (m/m)-

rsdin(j) Initial residue cover (kg/ha). The RSDIN is the initial residue cover at the start of simulation

3.4.9 INPUT FILE - .gw

The *xxxNN.gw* file includes groundwater parameters (where xxx is the basin name given when using GRASS interface, NN is the sub-basin number -1). An example of the *xxxNN.gw* input file format is presented in the following Table.

```
Groundwater DATA 15 16
    1.0000    0.5000    0.0480    0.0030  200.0000    0.2000    0.0500    0.0000
```

The next Table describes the format of the example file *xxxNN.gw*, including parameter names given above the values.

```
Groundwater DATA 15 16
    gwht(j)   gwq(j)    abf(j)    syld(j) delay(j)    revapc(j) rchrgc(j)
revapmn(j)
    1.0000    0.5000    0.0480    0.0030  200.0000    0.2000    0.0500    0.0000
```

Parameters included in the *xxxNN.gw* file are:

gwht(j) Initial groundwater height (m)

gwq(j) Initial groundwater flow contribution to streamflow (mm/day)
In SWIM: gwq(j) = gwq0

abf(j) alpha factor for groundwater. ABF characterizes the groundwater recession and the rate at which groundwater flow is returned to the stream.
In SWIM: abf(j) = abf0

syld(j) Specific yield

delay(j) groundwater delay (days). The time it takes for water leaving the bottom of the root zone until it reaches the shallow aquifer where it can become groundwater flow.
In SWIM: delay(j) = exp(-1./(delay(i)+1.e-6))

revapc(j) Revap coefficient (0-1) is the fraction of recharge (root zone percolation) that goes to REVAP. The amount of evaporation from the shallow aquifer is determined by multiplying potential ET by REVAPC.

rchrgc(j) Deep aquifer percolation coefficient (0-1). The amount of water that percolates into the deep aquifer (from the shallow aquifer) is determined by multiplying root zone percolation by RCHRGC.

revapmn(j) Revap storage (mm). Shallow aquifer storage must exceed REVAPMN before groundwater flow can begin.

3.4.10 INPUT FILE - .rte

The *xxxNN.rte* file includes channel routing parameters (where xxx is the basin name given when using GRASS interface, NN is the sub-basin number -1). An example of the *xxxNN.rte* input file format is presented in the following Table.

```
SUBBasin Routing DATA 15 16
  19.447   0.793   0.002   3.785   0.050   1.000   0.280   1.000
```

The next Table describes format of the example file *xxxNN.rte*, including parameter names given above the values.

```
SUBBasin Routing DATA 15 16
  chw(2,j) chd(j) chss(j) chl(2,j) chnn(j) chk(2,j) chxk(j) chc(j)
  19.447   0.793   0.002   3.785   0.050   1.000   0.280   1.000
```

Parameters included in the *xxxNN.rte* file are:

chw(2,j) Average channel width (m). It is estimated from a topographic map by GRASS interface and used for routing. May be substituted by actual channel width if available.

chd(j) Average channel depth (m). If detailed channel cross-section data is unavailable, this parameter is estimated from a topographic map by GRASS interface and used for routing. May be substituted by actual channel depth if available.

chss(j) Channel slope (m/m). If detailed channel cross-section data is unavailable, this parameter is estimated from a topographic map by GRASS interface and used for routing. May be substituted by actual channel slope if available.

chl(2,j) Channel length (km). If detailed channel data is unavailable, this parameter is estimated from a topographic map by GRASS interface and used for routing. May be substituted by actual channel length if available.

chnn(j) Channel n value (mm/hr). The channel "n" value is Mannings' "n" value. It is set to a default value 0.050 (as for natural streams with few trees, stones and brush) by GRASS interface and used to calculate routing. This value may be corrected by user. The typical values of chnn() are given in **Tab. 3.17** for different streams.

Table 3.17 Values of Manning's "n" for channels

Channel Flow	*Value Chosen*	*Range*
Excavated or dredged		
Earth, straight and uniform	0.0250	0.016-0.033
Earth, winding and sluggish	0.0350	0.023-0.05
Not maintained, weeds and brush	0.0750	0.04-0.14
Natural streams		
Few trees, stones, or brush	0.0500	0.025-0.065
Heavy timber and brush	0.1000	0.05-0.15

chk(2,j) Effective hydraulic conductivity in channel alluvium (mm/hr). It is set to a default value 1.0 (as for very low loss rate) by GRASS interface and used to calculate routing. This value may be corrected by user. Effective hydraulic conductivity of the channel alluvium is given in **Tab. 3.18** for various channel bed material.

Table 3.18 Effective hydraulic conductivity of the channel alluvium

Bed Material Group	Bed Material Characteristics	Effective Hydraulic Conductivity (mm/hr)
1 Very high loss rate	Very clean gravel and large sand d50>2 mm	>127
2 High loss rate	Clean sand and gravel under field conditions, d50>2 mm	51-127
3 Moderate high loss rate	Sand and gravel mixture with less than a few percent silt-clay	25-76
4 Moderate loss rate	Mixture of sand and gravel with significant amounts of silt-clay	6.4-25
5 Very low loss rate	Consolidated bed material with high silt-clay content	0.025-2.5

chxk(j) erodibility of stream channel, or USLE soil factor K for channel (range: 0-1). chxk=0 indicates a nonerosive channel, while chxk = 1 indicates no resistance to erosion. It is set to a default value 0.28 by GRASS interface and used in calculating routing. This value may be corrected by user.

chc(j) Cover factor for stream channel, or USLE soil factor C for channel (range: 0-1). If there is no vegatative cover, chc = 1. It is set to a default value of 1.0 (no vegetative cover) by GRASS interface and used to calculate routing. This value may be corrected by user.

3.4.11 INPUT FILE - wstor.dat

The *wstor.dat* file sets initial storage values for the reaches. It lists all reaches and the corresponding initial water storage in m^3. The values for initialisation can be obtained from a test run, considering the simulated water storage at the end of the first year.

i1 reach, corresponding to the subbasin i1

sdtsav initial water storage in the reach i1

3.4.12 INPUT FILE - soilNN.dat

The *soilNN.dat* files include soil parameters (where, NN is the soil type number). An example of the *soilNN.dat* input file format is presented in the following Table.

```
  36   5
Tschernosem aus Loess
  Ap      Ap Ahl    AhBt     Bt
  Ut3     Ut3      Ut3      Lu      Ut3
  10.    300.0   600.0  1000.0  1100.0
  15.     15.     15.     25.     15.
  75.     75.     75.     60.     75.
  10.     10.     10.     15.     10.
   1.40    1.40    1.50    1.70    1.6
  52.5    52.5    50.5    45.5    40.5
  26.5    26.5    26.     17.5    23.5
  41.5    41.5    40.     39.     36.
   2.0     2.0     1.2     0.9     0.
   0.2     0.2     0.1     0.      0.0


   0.56
  16.7    16.7    10.4     4.2    10.4
```

The next Table describes format of the example file *soilNN.dat,* including parameter names given above the values.

```
  k   ns(k)
  36   5
  sname(k)
 Tschernosem aus Loess
  Ap      Ap Ahl    AhBt     Bt
  Ut3     Ut3      Ut3      Lu      Ut3
  z(l,k)
  10.    300.0   600.0  1000.0  1100.0
  cla(l,k)
  15.     15.     15.     25.     15.
  sil(l,k)
  75.     75.     75.     60.     75.
  san(l.k)
  10.     10.     10.     15.     10.
  por(l,k)
   1.40    1.40    1.50    1.70    1.6
  poros(l,k)
  52.5    52.5    50.5    45.5    40.5
  awc(l,k)
  26.5    26.5    26.     17.5    23.5
  fc(l.k)
  41.5    41.5    40.     39.     36.
  cbn(l,k)
  2.0     2.0     1.2     0.9     0.
  wn(l,k)
  0.2     0.2     0.1     0.      0.0

  ek(k)
   0.56
  sc(l,k)
  16.7    16.7    10.4     4.2    10.4
```

Parameters included in the soilNN.dat file are:

k	soil type number in a database
ns(k)	number of soil layers for soil k. Number of soil layers for soil. The number of soil layers is assigned by the user. Usually the depths to the bottom of the layers are assigned in accordance with available database. Up to 10 layers are allowed, the first layer should be a depth of 10 mm.
snam(k)	name of soil k
z(l,k)	depth to bottom of layers l=1,...,ns(k) in mm
cla(l,k)	clay content in %
sil(l,k)	silt content %
san(l,k)	sand content, %
por(l,k)	bulk density (g/cm^3) (input), then recalculated to porosity
poros(l,k)	porosity, % (if available). If available water capacity, field capacity or total porosity values are missing, they can be estimated based on texture as in **Tab. 3.19**.
awc(l,k)	available water capacity, %. If available water capacity, field capacity or total porosity values are missing, they can be estimated based on texture as in **Tab. 3.19**.
fc(l,k)	field capacity, %. If available water capacity, field capacity or total porosity values are missing, they can be estimated based on texture as in **Tab. 3.19**.
cbn(l,k)	organic carbon content (%)
wn(l,k)	organic nitrogen content (%)
wno3(l,k)	initial NO3-N content (kg/ha), if available. Otherwise, it will be estimated from wn().
ap(l,k)	labile (soluble) phosphorus (g/t), if available. Otherwise, it will be estimated.
ek(k)	soil erodibility factor K (for USLE)
sc(l,k)	saturated conductivity (mm/h)

The following **Tab. 3.19** may be useful, if all necessary soil parameters are not available.

Table 3.19 Mean physical properties of soils (from Svetlosanov and Knisel, 1982)

Texture	*Volume (m/m)*				
	Bulk Density (gm/cm)	*Total Porosity*	*Field Capacity 1/3 bar*	*Wilting Point 15 bar*	*Available Water Capacity*
Coarse sand	1.600	0.40	0.11	0.03	0.080
Sand	1.600	0.40	0.16	0.03	0.130
Fine sand	1.500	0.43	0.18	0.03	0.150
Very fine sand	1.500	0.43	0.27	0.03	0.250
Loamy coarse sand	1.600	0.40	0.16	0.05	0.110
Loamy sand	1.600	0.40	0.19	0.05	0.140
Loamy fine sand	1.600	0.40	0.22	0.05	0.180
Loamy very fine sand	1.600	0.40	0.37	0.05	0.320
Coarse sandy loam	1.600	0.40	0.19	0.08	0.110
Sandy loam	1.600	0.40	0.22	0.08	0.140
Fine sandy loam	1.700	0.36	0.27	0.08	0.190
Very fine sandy loam	1.600	0.40	0.37	0.08	0.290
Loam	1.600	0.40	0.26	0.11	0.150
Silt loam	1.500	0.43	0.32	0.12	0.200
Silt	1.400	0.47	0.27	0.03	0.240
Sandy clay loam	1.600	0.40	0.30	0.18	0.120
Clay loam	1.600	0.40	0.35	0.22	0.130
Silty clay loam	1.400	0.47	0.36	0.20	0.160
Sandy clay	1.600	0.40	0.28	0.20	0.130
Silty clay	1.500	0.48	0.40	0.30	0.140
Clay	1.400	0.47	0.39	0.28	0.110

3.4.13 BLOCK DATA in the file init.f

Curve Numbers for 15 land use categories are initialized in block data (file init.f). The following values summarized in **Tab. 3.20** are assigned for land use categories and soil groups in SWIM. **Tab. 3.21** presents SCS runoff curve numbers for a variety of land use/land cover categories.

Table 3.20 Curve Numbers for land use categories and four soil groups used in SWIM

Land Use No.	*Land use category*	*Soil group*				*Source in SCS Tables*
		A	*B*	*C*	*D*	
1	Water	100.	100.	100.	100.	
2	Settlement	72.	79.	85.	88.	Urban areas, medium density
3	Industry	81.	88.	91	93	Industrial
4	Road	98.	98.	98.	98.	Paved streets and roads
5	Cropland	65.	75.	82.	86.	Row crops, contoured
6	Set-aside	66.	77.	85.	89.	Rotation meadow
7	Extensive grassland (meadow)	30.	58.	71	78.	Meadow, continuous grass
8	Intensive grassland (pasture)	49.	69.	79.	84.	Pasture, continuous forage for grazing
9	Mixed forest	36.	60.	73.	79.	Woods, fair
10	Evergreen forest	36.	60.	73.	79.	Woods, fair
11	Deciduous forest	36.	60.	73.	79.	Woods, fair
12	Wetland nonforested	85.	85.	85.	85.	
13	Wetland forested	85.	85.	85.	85.	
14	Heather (grass + brushland)	35.	56.	70.	77.	Brush-weed-grass mixture with brush the major element
15	Bare soil	77.	86.	91.	94.	

Table 3.21 SCS Curve Numbers for a variety of land use/land cover categories

Land Use/Crop	Cover	Condition	A	B	C	D
Fallow	Straight row	---	77.0	86.0	91.0	94.0
Row crops	Straight row	Poor	72.0	81.0	88.0	91.0
		Good	67.0	78.0	85.0	89.0
	Contoured	Poor	70.0	79.0	84.0	88.0
		Good	65.0	65.0	82.0	86.0
	Contoured and terraced	Poor	66.0	74.0	80.0	82.0
		Good	62.0	71.0	78.0	81.0
Small grain	Straight row	Poor	65.0	76.0	84.0	88.0
		Good	63.0	75.0	83.0	87.0
	Contoured	Poor	63.0	74.0	82.0	85.0
		Good	61.0	73.0	81.0	84.0
	Contoured and terraced	Poor	61.0	72.0	79.0	82.0
		Good	59.0	70.0	78.0	81.0
Close-seeded legumes* or rotation meadow	Straight row	Poor	66.0	77.0	85.0	89.0
		Good	58.0	72.0	81.0	85.0
	Contoured	Poor	64.0	75.0	83.0	85.0
		Good	55.0	69.0	78.0	83.0
	Contoured and terraced	Poor	63.0	73.0	80.0	83.0
		Good	51.0	67.0	76.0	80.0
Pasture or range	Straight row	Poor	68.0	79.0	86.0	89.0
		Fair	49.0	69.0	79.0	84.0
		Good	39.0	61.0	74.0	80.0
	Contoured	Poor	47.0	67.0	81.0	88.0
		Fair	25.0	59.0	75.0	83.0
		Good	6.0	35.0	70.0	79.0
Meadow		Good	30.0	58.0	71.0	78.0
Woods		Poor	45.0	66.0	77.0	83.0
		Fair	36.0	60.0	73.0	79.0
		Good	25.0	55.0	70.0	77.0
Farmsteads		---	59.0	74.0	82.0	86.0
Roads (dirt)**		---	72.0	82.0	87.0	89.0
Roads (hard surface)**		---	74.0	84.0	90.0	92.0