The FORTRAN version of the Van Keulen - Seligman CSMP-Spring wheat model

Simulation Reports CABO-TT, no. 30

W. Stol, H. van Keulen, D.W.G. van Kraalingen

cabo-dlo

May 1993

sequence of calculations gives results in FORTRAN and CSMP that are identical, is shown for a simple simulation of exponential growth in Listings 1 and 2.

Listing 1: CSMP program of exponential growth and output (only the relevant output is reproduced):

```
TITLE DEMONSTRATION
INCON IH=1.
PARAMETER RGR=0.1
H  = INTGRL (IH, GR)
GR = RGR*H
METHOD RECT
TIMER TIME=0.0, FINTIM=10., DELT=1.0, PRDEL=1.0
PRINT H, GR
END
STOP
ENDJOB

0TIMER VARIABLES      RECT     INTEGRATION       START TIME   = .00000
    DELT           DELMIN        FINTIM        PRDEL         OUTDEL      DELT
   1.0000      1.00000E-06   10.000       1.000          .00000     1.0000
1  DEMONSTRATION
0  TIME          H             GR
   .000000       1.0000        .10000
   1.00000       1.1000        .11000
   2.00000       1.2100        .12100
   3.00000       1.3310        .13310
   4.00000       1.4641        .14641
   5.00000       1.6105        .16105
   6.00000       1.7716        .17716
   7.00000       1.9487        .19487
   8.00000       2.1436        .21436
   9.00000       2.3579        .23579
   10.0000       2.5937        .25937
1$$$ SIMULATION HALTED FOR FINISH CONDITION  TIME      10.000
1$$$ CONTINUOUS SYSTEM MODELING PROGRAM III   V2.0     EXECUTION OUTPUT
```

Listing 2: FORTRAN program of exponential growth and output.

```
      PROGRAM DEMO
      IMPLICIT REAL (A-Z)
      PARAMETER (RGR=0.1, FINTIM=10., DELT=1.0)

      H  = 1.0
      GR = 0.0
      TIME = 0.0

      OPEN (20, FILE='RES.OUT', STATUS='NEW')
      WRITE (20,'(A9,2A13)') 'TIME','H','GR'

10    IF (TIME.LE.FINTIM) THEN
         H  = H+GR*DELT                          <--integration
                                                 <--driving variables (none)
         GR = RGR*H                              <--rate calculation
         WRITE (20,'(3G13.5)') TIME, H, GR       <--output
         TIME = TIME+DELT                        <--time=time+delt
      GOTO 10
```

```
      END IF

      STOP
      END
```

TIME	H	GR
.00000	1.0000	.10000
1.0000	1.1000	.11000
2.0000	1.2100	.12100
3.0000	1.3310	.13310
4.0000	1.4641	.14641
5.0000	1.6105	.16105
6.0000	1.7716	.17716
7.0000	1.9487	.19487
8.0000	2.1436	.21436
9.0000	2.3579	.23579
10.000	2.5937	.25937

In theory, the sequence in which various state variables are updated is not important because their values should not depend on each other, but should be fully determined by the rate variables. In practice, however, state variables may sometimes be derived from other state variables (e.g. root/shoot ratio or total dry weight of leaf blades equals dry weight of dead leaf blades plus dry weight of green leaf blades). It is therefore important to put the state calculations in the proper sequence, as is also necessary for the rate calculations.

To ensure that the results of the simulation are correct, the different types of calculations (integration, driving variables and rate calculations) should be strictly separated. In other words, all states should be updated, then all driving variables should be calculated, after which all rates of change should be calculated. If this rule is not applied rigorously, there is a risk that some rates will pertain to states at the current time whereas others will pertain to states from the previous time step.

Since the calculations of rates and states cannot be mixed during a time step but should be executed separately, all the state calculations have to be merged into one block as do all the rate calculations. Often, different subprocesses are interacting (e.g. a plant transpiring water from the soil). In many cases these interactions among the subprocesses are only weak. The water content at different depths in the soil is needed for the plant/soil system in the plant submodel. This is then used to determine water uptake for transpiration in dependence of rooting depth. The submodels for the plant and soil water thus share a limited amount of information, but they may contain very detailed descriptions of plant growth and soil moisture redistribution with many different rate and state calculations.

In view of the above, it is not a good solution to combine all the state calculations from the different subprocesses into one large program section and all the rate calculations in another. But it is feasible to separate the state and rate calculations within the subprocess descriptions (such as the plant) and have a calling program decide which of the two to execute. With this method, the states can be calculated separately from the rates, whereas rates and states pertaining to the same subprocess are within the same subprogram. This technique is also discussed by van Kraalingen and Rappoldt (1989).

This concept of 'task-controlled execution' is illustrated in figure 2. The program lines of the plant and soil water subprocesses are separated into rate and state sections and only one of these is executed during a single call. Note that this program structure performs the calculations in exactly the same order as the circle given in figure 1.

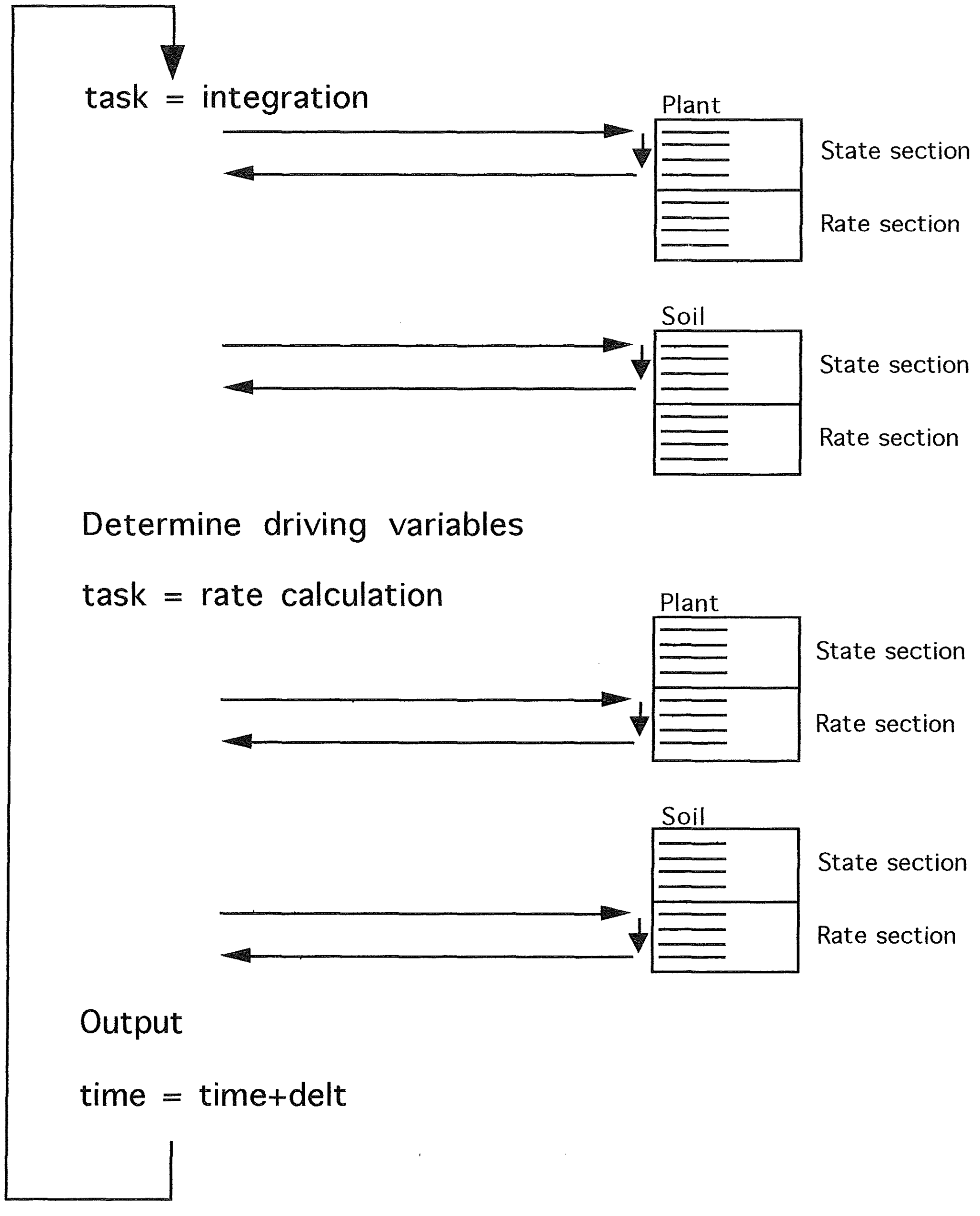

Figure 2 **General structure for incorporating several subprocesses containing integration and rate calculations into a single simulation model.**

So far, we have not discussed how to initialize the states, or where to enter the simulation circle and where to leave it (see figure 1).

It is convenient to leave the circle somewhere between time update and integration, because there the time and corresponding rates have been written to the output device and after the time update it seems logical to check whether the finish time (FINTIM) has been exceeded or whether further simulation is required. Consequently, the circle should also be entered between time update and integration. The most convenient way to initialize the subprocesses is to have this operation controlled by the main program. This makes reruns possible, because in the main program the whole model can be reset to its initial state and be run again, with different weather data for instance. These refinements to figure 1 are shown in figure 3.

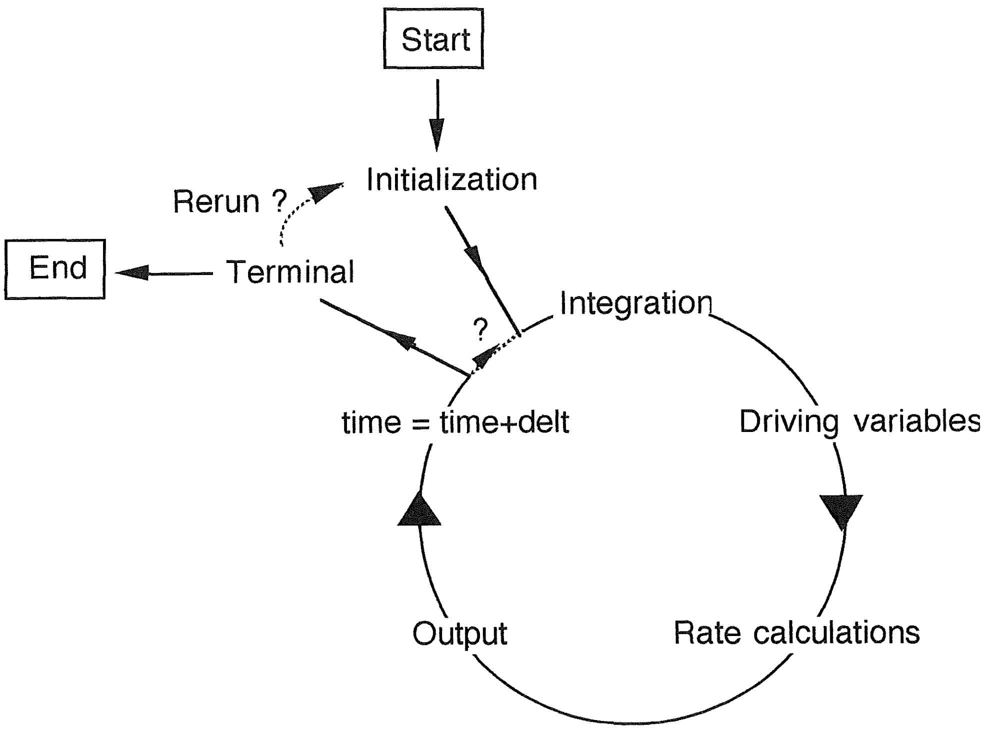

Figure 3 The order in which calculations are executed when simulating continuous systems using Euler integration, illustrating where to enter and leave the circle and how reruns are implemented.

The question mark between time = time+delt and integration indicates the point at which it is decided whether or not to execute another time step. If the decision is "no", the model proceeds to the terminal section; if it is "yes" the circle is run once more. After proceeding to the terminal section, it must be decided whether a rerun is required. If the decision is "yes" the model has to be re-initialized and a new simulation run is started.

As shown in Figure 2, the modularity of the subprocess descriptions is preserved by introducing the concept of task-controlled execution (the calling program decides what the subroutine should do: either integration or rate calculation). To be able to do reruns, the various subprocess descriptions that can also be driven by the task variable have to be initialized externally, and some terminal calculations (e.g. harvest index) have to be done. Thus, a subprocess description in the FSE program should recognize four different tasks: initialization, integration, rate calculation and terminal calculation.

A consequence of this structure is that the first step after initialization is integration. This does no harm if the rates have been set to zero explicitly, so that the first integration has no effect on the value of the states. In practice, this means incorporating many rate assignments to zero into the model. To avoid this, integration can be skipped if the previous task was initialization (during which the states have been assigned values anyway). The subsequent rate calculation will then use the state variables to initialize the rates of change. This shortcut is implemented in the main program of FSE, but has not been shown in Figure 3.

3. Principles of the translation

With respect to translation principles only the general approach which was applied to the CSMP spring-wheat model will be given. More detailed instructions are given by van Kraalingen & Penning de Vries, (1990) in the MACROS documentation.

The translation can be divided into five phases:

- Separate parameters, functions and initial values of state variables from the rate and state equations and put them into specified datafiles for time, soil and plant processes.

- Rewrite the more or less free-formatted CSMP-source code in statements according to the FORTRAN-format. That means that statement labels such as (10 CONTINUE) should be in position 1 to 5, a possible continuation character ($ or 1) on position 6. The FORTRAN statements and equations themselves can go from position 7 to 72. Characters from position 73 onwards are ignored by the FORTRAN-compiler.
  These adaptations to the FORTRAN-syntax do not alter the lay-out of the original model in CSMP dramatically. This has particular advantages for the users of the simulation model, who can use both the original documentation in the Simulation Monograph or this report as technical reference document.

- Assign all the program statements to sections depending on their specific functions: initialization of state variables, calculation of rates or integration of state variables. These sections are successively called by the main program which is an implementation of FSE (Fortran Simulation Environment), developed by van Kraalingen (1991). The principle applied here is that of task-specified calls from the main program to the submodel. This type of organization of simulation (sub-)models was documented by van Kraalingen and Rappoldt (1989). The general idea behind this was discussed in Chapter 2.

- Sort the statements into an executable sequence. To enable parallel integration on analogous computers some boundary conditions have to be fulfilled. Each variable should first be calculated on the left-hand side of the equation before its value is used on the right-hand side. In other words, all variables should first be initialized or calculated before they are used in equations to calculate other variables. Normally, only sorting of rate equations is necessary to comply with the terms of parallel integration. In this model however, sorting of initialization and the integration sections was necessary as well. There were several reasons. The first is the addition of soil state variables per layer to a sum over the whole profile. The second reason is that some intermediate integrals such as the total dry weight of vegetative organs are calculated from other integrals. The third reason is that in this particular model initialization of some integrals was implemented in the integral statement itself by adding the product of an initial rate and an impulse. This impulse will be initiated after completion of germination of the seeds and emergence of the seedlings. This integral (state variable: PUSHDI, rate variable: PUSHD) should be the first integral in the integration section because of the formulation of the initial rate. The initial rate after emergence of some plant processes is formulated as:

```
STATE = INTGRL (STATE, RATE+INITIAL_RATE*EMERGENCE_IMPULSE, DELT)
```

- Initialize rate equations explicitly. Within CSMP, rate equations are explicitly initialized when time equals zero. That means that all rates are evaluated once when time equals zero without integration of state variables (RKEEP=0). After that, rates are calculated again with (RKEEP=1) which means that integration of state variables is performed as well. In the spring-wheat model some rates of soil processes depend on other rates of soil processes. Explicit initialization when RKEEP equals zero has then significant effects. To ensure that the output of the FORTRAN model is similar to that of the CSMP model, the rate calculation section in the FORTRAN model was evaluated twice when time equals zero and RKEEP equals 1.

4. Adaptations to the CSMP-version

During translation to FORTRAN-77 some parts of the source have been replaced with newer procedures. The procedure to calculate daily total gross assimilation according to Goudriaan and van Laar (1978) was updated with a new algorithm by Spitters et al. (1989). In the original CSMP model the input of weather-data was by means of CSMP-table statements with appropriate data. In the FORTRAN-model weather-data were supplied by the WEATHER facility (van Kraalingen et al., 1990) which is integrated in the FORTRAN Simulation Environment.

4.1. Assimilation procedure

The calculation of daily gross assimilation was done according to a procedure from Goudriaan and van Laar (1978) in the CSMP-version. In 1986 a more proper and accurate procedure was developed by Goudriaan and later described by Spitters et al. (1989). The calculation of daily total gross assimilation in this procedure is done by means of a weighted three-point Gaussian integration over both daytime and canopy depth. Spitters (1986) shows that this method gives reliable results in comparison to measured assimilation rates. A detailed comparison between the old and the new assimilation procedure was carried out by Bastiaans et al. (1986). They concluded that the new procedure calculates higher daily assimilation rates than the old procedure under the same input conditions. Underestimation of daily assimilation rates will be greater at lower values of the green area index. In the old procedure the daily assimilation rate is first calculated for a green area index (GRAI) of 5 and later adjusted for lower values of GRAI.

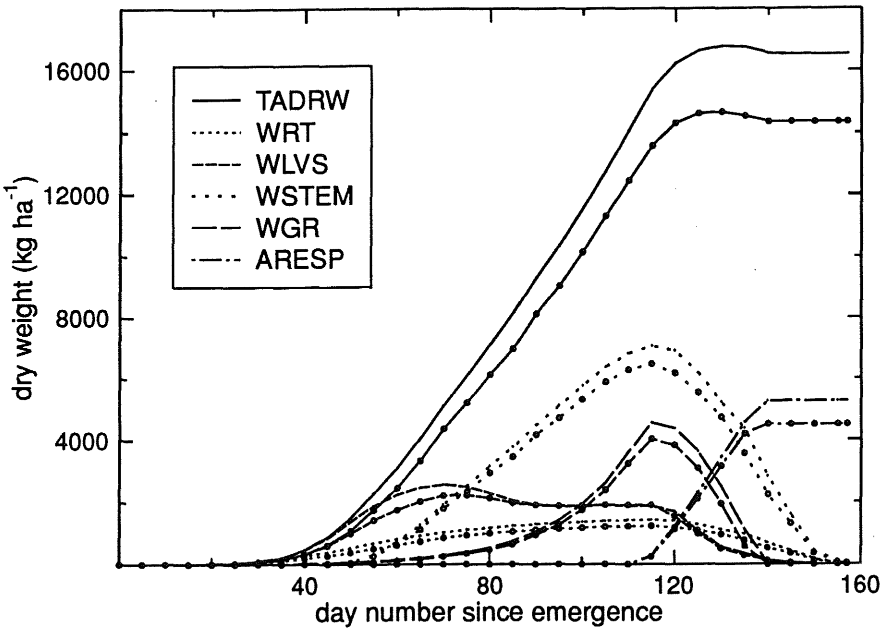

Figure 4 Comparison of the simulated dry matter accumulation in the spring wheat model with the old (PHOTPR) and with the new procedure for calculating daily total gross assimilation (subroutine ASTRO-TOTASS). TADRW, total above ground dry weight, WRT, dry weight of roots, WLVS, dry weight of leaves, WSTEM, dry weight of live stem and sheaths, WGR, dry weight of grain and ARESP, weight of reserve carbohydrates.

This procedure shows weaknesses. The underestimation of the daily assimilation rates with the old procedure was larger at short daylengths and at higher radiation intensities, when the sky is only partly overcast. Figure 4 shows simulated data of total dry matter accumulation with the old (PHOTPR) and with the new procedure (subroutines ASTRO-TOTASS). The simulation was carried out with the standard-run of the spring-wheat model, site Migda, Israel, 1979-1980, High-N; 260 kg nitrogen in the soil profile in inorganic form. Comparison of the old and the new procedure for final total above-ground dry weight shows a difference of +2150 kg ha^{-1} and +765 kg ha^{-1} in grain yield. Figure 5 shows the difference in simulated green area index which results from the higher assimilation rates in the vegetative phase.

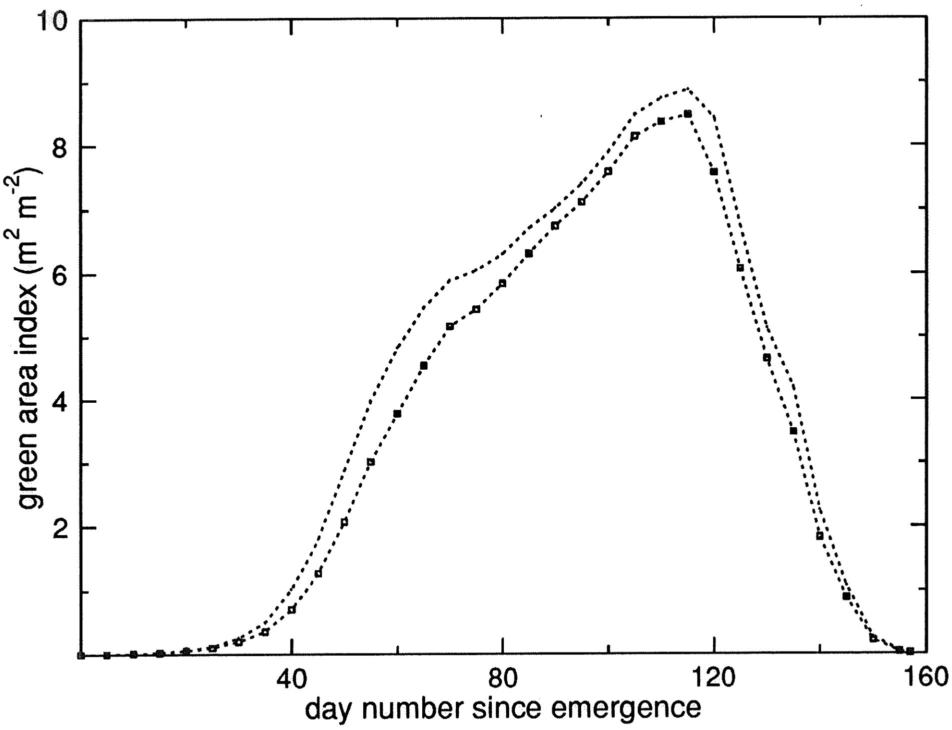

Figure 5 Comparison of the simulated leaf area development in the spring wheat model with the old (PHOTPR) and with the new procedure for calculating daily total gross assimilation (subroutine ASTRO-TOTASS).

4.2. Weather-data

As stated in the introduction of this chapter, the CSMP-tables with weather data were replaced by the standard-datafiles of the CABO/TPE weather system. In this Section a summary explanation is given about the organization of this system within the spring-wheat model. The weather system consist of a set of subprograms which supplies the weather data and datafiles with the measured data. The subprograms WEATHR and STINFO, programmed in standard FORTRAN-77, are included on the floppy-disc in the file WEATHER.FOR. The files with the weather-data are in ASCII-format. These files can easily be accessed by the user and the reading program.

Files with weather data stored in the CABO/TPE WEATHER format contain the following information:

* Filename: unique filename which holds information on country, station number and specific year. The name of the file of Migda, a weather station in Israel coded with station number one with data of 1979 has the file name ISR1.979 for example. The filename consists of concatenated strings of country name, station and year number.

* Fileheader: gives information about: station name, author of the datafile, source, comments: about special circumstances, conversion of data, location data: longitude, latitude and altitude, layout of the records with weather data; column number, weather variables, units. The file-header is interpreted as comment by the programs due to the asterisk in the first column.

* Data-records: weather files can contain three types of records: a location record, normal data records and optional status records. On the location record there are five (real) data, free-formatted, separated by blanks. The first three data give the location of the weather-station: longitude and latitude given in decimal degrees and the altitude of the station in meters. The last two data in the location record are two coefficients in the Ångström formula. Global radiation can be estimated with this formula from daily sunshine duration in hours, using two site-specific regression coefficients. The two last coefficients on the location record are zero when global radiation is measured . The WEATHR-subroutine serves, depending on the values of the coefficients A and B, directly measured data or an estimated value of global radiation using the Ångström formula when both values are unequal to zero. Essential from the point of view of the reading program is that there are five values on the record, all specified as real variables and separated by blanks.

* Status records: Each data record can be preceded by a status record which is meant to code the weather variables with temporary attributes. These attributes can have the values of 1 to 4 with the following meaning;

1: data are measured and available
2: data were calculated by linear interpolation
3: data were taken from a nearby station (or intelligent guesses).
4: data are not available; dummy value (-99.) is returned

The attributes in the record are preceded by 3 values, the first is -999 to make clear that the record is a status record, the second is year number, the third is day number. The files ISR1.979 and ISR1.980 on the floppy-disk can be used as examples to produce weather files for your location.

5. Detailed description of module SWHEAT

5.1. Introduction

In this chapter the documentation of the spring-wheat module SWHEAT is given. Reading this description together with the program listing of the spring-wheat module, the different parts of the program will be explained. Although the module is named SWHEAT, it consists of the description of the carbon and nitrogen balance of crop and soil, the water-balance of the soil and the development of the spring-wheat crop. Modular programming techniques learn that such different processes should be split up. Because of the complex structure of the model, this was not actually done, so the version presented here is a good one in terms of execution order and documentation, but a less good one with respect to modular design of transportable modules. Subsections of the module were labelled from column number 73 onwards. The initial section comprises label 10 to label 930, the rate calculations section labels 940 to 4320 and the integration section labels 4330 to 4990. A terminal section does not exist in the module SWHEAT.

5.2. Interface of module with FSE driving routine

The declaration section of the subroutine SWHEAT contains a list of parameters that are supplied from the main program and declaration statements. Table 1 gives information about the parameters that are supplied.

Table 1. Input parameters

a. Variables for task specifications:

Name	Datatype	Description
ITASK	I4	Determines action of the subroutine

b. Variables for input/output specifications:

Name	Datatype	Description
IUNITP	I4	Unit number of plant datafile
IUNITO	I4	Unit number of output file
IUNITS	I4	Unit number of soil datafile
FILEP	C*	Name of plant datafile
FILES	C*	Name of soil datafile
OUTPUT	L4	Flag that indicates if model output to file is required
TERMNL	L4	Flag that indicates if simulation should terminate

c. Variables for timing specification:

Name	Datatype	Description	Dimension	Function
TIME	R4	Time of simulation	d	T
DATE	R4	Day number of simulation	d	T
DELT	R4	Time step of integration	d	T

d. Variables for environment specification:

Name	Datatype	Description	Dimension	Function
LAT	R4	Latitude	degrees	I
RADI	R4	Global radiation	KJ m^{-2} d^{-1}	I
TMIN	R4	Minimum temperature	° C	I
TMAX	R4	Maximum temperature	° C	I
VAPOUR	R4	Vapour pressure	kPa	I
WIND	R4	Average wind speed	m s^{-1}	I
RAIN	R4	Precipitation	mm d^{-1}	I

Input parameters can be divided in classes according to their function: control, timing or environmental parameters or to their function in the communication, so input to or output from the subroutine. Declaration of model variables is done between label 10 and label 20. All REAL variables are declared implicitly in this module with the FORTRAN IMPLICIT statement. The advantage of this way of declaration of variables in simulation models is less explicit declaration of variables, a disadvantage is that every typing error in a variable name introduces a new model variable. These types of error can nowadays easily be recognized by software maintenance programs especially FORCHECK (Kruyt, 1989). Integer variables are explicitly declared. Arrays are dimensioned using the FORTRAN DIMENSION statement which defines the maximum size of the arrays. Four parameters are in use for array dimensioning; ILMAX1, ILMAX2, IARL1 and IARL2. The first two parameters are used to define the maximum numbers of soil compartments and the number of fluxes over soil compartment boundaries. ILMAX1 has a value of 10 and ILMAX2 has a value of 11. The parameters IARL1 and IARL2 are used to define the maximum length of arrays used for forcing functions and for a number of arrays with states and rates of leaf age classes. Parameter IARL1 has a value of 30 and parameter IARL2 has a value of 110.

In the following section, parts of the initial section of the module SWHEAT are described according to their specific function in the program. From label 20 to label 30 the value of the variable ITASK is evaluated. This is done with the TTUTIL-subroutine CHKTSK (Rappoldt & van Kraalingen, 1990). This subroutine takes care of the right order of execution of the simulation algorithm. When the new value of ITASK does not match with the preceding one, the job is terminated with an error message.

5.3. Initialization section of module

To facilitate multiple runs, all (initial-) state variables were explicitly initialized instead of implicitly by the FORTRAN-compiler. Each time the module is called with an ITASK-value of 1 it is completely reset.

```
CALL RDINIT (IUNITP, IUNITO, FILEP)
```

L30: The module RDINIT is called to open and read the plant datafile on unit IUNITP with name FILEP, error messages are send to an output file on IUNITO.

```
PUSHDI = 0.
PUSHD = 0.
```

L4000: The state variable PUSHDI, which monitors the end of germination and PUSHD the corresponding rate are initialized at zero.

```
CALL RDSREA ('TGERD'  ,TGERD)
CALL RDSREA ('TSDAY'  ,TSDAY)
CALL RDSREA ('SOWD'   ,SOWD)
CALL RDSREA ('GERDAT',GERDAT)
CALL RDSREA ('RADCF'  ,RADCF)
CALL RDSREA ('EXPTF'  ,EXPTF)
CALL RDSREA ('CTEMPF',CTEMPF)
CALL RDSREA ('FTMPA'  ,FTMPA)
CALL RDSREA ('RAINF'  ,RAINF)
```

L50: The variables TGERD, cumulative number of days with favourable conditions for germination, TSDAY, total number of consecutive days with conditions conducive for seed deterioration, SOWD, sowing date, GERDAT, date on which germination is completed, RADCF, factor to convert radiation from KJ m^{-2} to J m^{-2}, EXPTF, variable to adapt actual temperature regime to alternative climate scenarios, CTEMPF, magnitude of the effect of evaporative cooling on soil and canopy temperature, FTMPA, reciprocal of the mean standard canopy temperature, RAINF, parameter which enables variation in rainfall intensity, are initialized.

```
      TRAIN = 0.
```

L60: The state variable TRAIN, total seasonal rainfall, is initialized at zero.

```
      TOTRAN = 0.
      CALL RDSREA ('PI'    ,PI)
      CALL RDSREA ('DRAGC' ,DRAGC)
      CALL RDSREA ('KARMAN',KARMAN)
      CALL RDSREA ('IW'    ,IW)
      CALL RDSREA ('REFHT' ,REFHT)
      CALL RDSREA ('RHOCP' ,RHOCP)
      CALL RDSREA ('PSCH'  ,PSCH)
      CALL RDSREA ('SCM'   ,SCM)
      CALL RDSREA ('RC'    ,RC)
      CALL RDSREA ('EXC'   ,EXC)

      CALL RDAREA ('CROHTB',CROHTB ,IARL1 ,ICRHTB)
      CALL RDAREA ('ALPHTA',ALPHTA ,IARL1 ,IALPTA)
      CALL RDAREA ('ALPHTB',ALPHTB ,IARL1 ,IALPTB)
      CALL RDAREA ('ALPHTC',ALPHTC ,IARL1 ,IALPTC)
      CALL RDAREA ('ALPHTD',ALPHTD ,IARL1 ,IALPTD)
      CALL RDAREA ('ALPHTE',ALPHTE ,IARL1 ,IALPTE)
      CALL RDAREA ('ALPHTF',ALPHTF ,IARL1 ,IALPTF)
      CALL RDAREA ('WCRRT' ,WCRRT  ,IARL1 ,IWCRRT)

      CALL RDSREA ('NFDEV' ,NFDEV)
      CALL RDSREA ('CULTP' ,CULTP)
      CALL RDSREA ('CULTM' ,CULTM)

      CALL RDSREA ('DVSI'  ,DVSI)
      CALL RDSREA ('DVSTS' ,DVSTS)
      CALL RDSREA ('DVSSE' ,DVSSE)
      CALL RDSREA ('DVSSPS',DVSSPS)
      CALL RDSREA ('DVSST' ,DVSST)
      CALL RDSREA ('DVSFS' ,DVSFS)
      CALL RDSREA ('DVSPRE',DVSPRE)
      CALL RDSREA ('DVSSPE',DVSSPE)
      CALL RDSREA ('DVSFE' ,DVSFE)
      CALL RDSREA ('DVSAN' ,DVSAN)
      CALL RDSREA ('DVSGS' ,DVSGS)
      CALL RDSREA ('DVSSGF',DVSSGF)
      CALL RDSREA ('DVSEGF',DVSEGF)
```

L70: The variable TOTRAN, total seasonal crop transpiration is initialized at zero. The parameters PI, the circumference of a circle divided by its diameter, DRAGC, drag coefficient of the leaf blades, KARMANN, von Karmann's constant, IW, turbulence intensity, REFHT, reference height for measuring windspeed, RHOCP, volumetric heat capacity of the air, PSCH, psychrometer constant, SCM, minimum stomatal conductance, RC, cuticular resistance for water flow and EXC, extinction coefficient for global radiation are defined. After this, some arrays are initialized: CROHTB, height of the crop as function of development stage, ALPHTA to ALPHT describing the relation between average hourly radiation intensity and ALPHA, proportionality factor in the calculation of the contribution of the drying power of the air to crop transpiration, for values of LAI of 0, 0.2, 2., 3.5, 5. and 10, and WCRRT, weight to conductivity ratio of the roots as function of development stage. Parameters, NFDEV, reduction factor on development rate due to N-deficiency, CULTP, reduction factor on cultivar-characteristic pre-anthesis development rate and CULTM, definition of cultivar-characteristic post-anthesis development rate are initialized. The state variables DVSI, initial development stage, DVSTS, development stage at start of tillering, DVSSE, development stage at the start of floral initiation, DVSPS, development stage at start of spikelet differentiation, DVSST, development stage at start of stem elongation, DVSFS, development stage at start of floret formation, DVSPRE, development stage at end of tillering, DVSSPE, development stage at terminal spikelet formation, DVSFE, development stage at end of floret formation, DVSAN, development stage at start of grain fill and DVSEGF, development stage at end of grain fill are defined.

```
DVSV = 0.
```

L80: The state variable DVSV, the development stage in the pre-anthesis phase, is initialized at zero.

```
ANTHES = 0.
```

L90: The state variable ANTHES, day number of anthesis date, is initialized at zero.

```
DVSR = 0.
```

L100: The state variable DVSR, the development stage in the post-anthesis phase, is initialized at zero.

```
DVS = DVSI
```

L110: The state variable DVS, the overall development stage is assigned the value of the state variable DVSI, the initial development stage.

```
CALL RDSREA ('EFFE'  ,EFFE)
CALL RDSREA ('KDIF'  ,KDIF)
CALL RDSREA ('SCV'   ,SCV)

CALL RDSREA ('TLRGA' ,TLRGA)
CALL RDSREA ('RMRESL',RMRESL)
CALL RDSREA ('RMRESS',RMRESS)
CALL RDSREA ('RMRESR',RMRESR)
CALL RDSREA ('RMRESG',RMRESG)
CALL RDSREA ('Q10'   ,Q10)
CALL RDSREA ('EFCPR' ,EFCPR)
CALL RDSREA ('EFCCH' ,EFCCH)
CALL RDSREA ('EFCPRG',EFCPRG)
```

```
      CALL RDSREA ('TCDDH' ,TCDDH)
      CALL RDSREA ('FWDB'  ,FWDB)
      CALL RDSREA ('MRDRSH',MRDRSH)
      CALL RDSREA ('LAILM' ,LAILM)
      CALL RDSREA ('MXRDR' ,MXRDR)
      CALL RDSREA ('AVLTLF',AVLTLF)
      CALL RDSREA ('RSLDS' ,RSLDS)
      CALL RDSREA ('RFST'  ,RFST)
      CALL RDSREA ('FSCHG' ,FSCHG)
      CALL RDSREA ('RFRT'  ,RFRT)
      CALL RDSREA ('DGRRT' ,DGRRT)
      CALL RDSREA ('MXRTD' ,MXRTD)
      CALL RDSREA ('RTF'   ,RTF)
      CALL RDSREA ('RESLI' ,RESLI)
      CALL RDSREA ('RESL1' ,RESL1)
      CALL RDSREA ('RESL2' ,RESL2)
      CALL RDSREA ('TCTR'  ,TCTR)
      CALL RDSREA ('FRNGL' ,FRNGL)
      CALL RDSREA ('FRNGL1',FRNGL1)
      CALL RDSREA ('RESLR' ,RESLR)
      CALL RDSREA ('SWDF'  ,SWDF)

      CALL RDAREA ('TMPFT' ,TMPFT  ,IARL1 ,ITMPFT)
      CALL RDAREA ('FTLVST',FTLVST ,IARL1 ,IFTLST)
      CALL RDAREA ('REDWST',REDWST ,IARL1 ,IRDWST)
      CALL RDAREA ('RFNST' ,RFNST  ,IARL1 ,IRFNST)
      CALL RDAREA ('FDMT'  ,FDMT   ,IARL1 ,IFDMT)
      CALL RDAREA ('FTSTET',FTSTET ,IARL1 ,IFSTET)
      CALL RDAREA ('FDSRT' ,FDSRT  ,IARL1 ,IFDSRT)
      CALL RDAREA ('REDTTB',REDTTB ,IARL1 ,IRDTTB)
      CALL RDAREA ('FTRLT' ,FTRLT  ,IARL1 ,IFTRLT)
      CALL RDAREA ('PGRIGT',PGRIGT ,IARL1 ,IPGIGT)
```

L130: The following list of parameters is read from file, EFFE, initial light use efficiency at the light compensation point, KDIF, the extinction coefficient for diffuse PAR, SCV, scattering coefficient of leaves for PAR, TLRGA, threshold level of reserves for reduction of gross assimilation, RMRESL, maintenance requirement factor of leaf blades (at 20° C), RMRESS, maintenance requirement factor of stem and sheaths, RMRESR, maintenance requirement factor of roots, RMRESG, maintenance requirement factor of the grain, Q10, factor accounting for effect of temperature on maintenance respiration, EFCPR, conversion efficiency of primary assimilates into structural proteins, EFCCH, conversion efficiency of primary assimilates into structural carbohydrates, EFCPRG, conversion efficiency of primary assimilates into structural grain proteins, TCDDH, time constant for dehydration of plant tissue, FWDB, fraction of water left in dying plant tissue, MRDRSH, maximum relative death rate of leaf blades due to shading, LAILM, threshold value of LAI beyond which death due to shading starts, MXRDR, maximum relative death rate of leaf blades due to carbohydrate shortage, AVLTLF, average life span of leaf blades at standard temperature (15 °C), RSLDS, ratio of stem to leaf death, RFST, proportionality factor between relative death rate of stem and sheaths and leaf blades, FSCHG, fraction of 'surplus' carbohydrates allocated to the roots, RFRT, proportionality factor between relative death rate of roots and stem and sheaths, DGRRT, maximum rate of root extension, MXRTD, maximum rooting depth, RTF, factor representing the effect of stress on the rate of root extension, RESLI, initial concentration of non-structural carbohydrates in the canopy, RESL1, residual concentration of non-structural carbohydrates for translocation to vegetative structures before end of floret formation, RESL2, residual concentration of non-structural carbohydrates for translocation to vegetative structures after end of floret formation, TCTR, time constant for translocation of reserves, FRNGL, minimum nitrogen concentration in the grain, FRNGL1, nitrogen concentration in the grain at which dry matter accumulation starts to be affected, RESLR, residual concentration of reserves for translocation to the grain, SWDF, variable to set different sowing densities. The arrays, TMPFT, reduction factor on gross CO_2-assimilation as a function of temperature, FTLVST, fraction of current assimilate supply allocated to leaf blades as a function of development stage, REDWST, reduction factor on leaf blade growth as a function of relative transpiration deficiency, RFNST, reduction factor on leaf growth as a function of relative nitrogen concentration in leaf blades, FDMT, fraction dry matter in vegetative tissue as a function of development stage, FTSTET, fraction of assimilate supply allocated to stem and sheaths as a function of development stage, FDSRT, fraction of assimilate supply allocated to roots as a function of development stage, REDTTB, reduction factor on root extension as a function of soil temperature, FTRLT, fraction of assimilate supply allocated to the reserve pool as function of development stage, PGRIGT, potential rate of dry matter accumulation of an individual grain as a function of canopy temperature are defined.

```
      DO 10 I = 1,110
         LEAFW(I) = 0.
         ALFT(I)  = 0.
         LEAFA(I) = 0.
10      CONTINUE
```

L120: The arrays of state variables, LEAFW, leaf dry weight, ALFT, accumulated temperature sum and LEAFA, leaf area, all per leaf class, are explicitly initialized at zero.

```
      RTD = 0.
```

L130: The state variable RTD, rooting depth, is initialized at zero.

```
      EGFDAY = 0.
```

L140: The state variable EGFDAY, day number at end of grain filling, is initialized at zero.

```
      WLVSI = 0.
```

L150: The state variable WLVSI, dry weight of live leaf blades at emergence, is initialized at zero.

```
      WLVS = 0.
```

L160: The state variable WLVS, dry weight of live leaf blades, is initialized at zero.

```
      WRTI = 0.
```

L170: The state variable WRTI, dry weight of live roots at emergence, is initialized at zero.

```
      WRT = 0.
```

L180: The state variable WRT, dry weight of live roots, is initialized at zero.

```
      WSTEM = 0.
```

L190: The state variable WSTEM, dry weight of live stem and sheaths, is initialized at zero.

```
WGR = 0.
```

L200: The state variable WGR, dry weight of the grain, is initialized at zero.

```
TVEGM = 0.
```

L210: The state variable TVEGM, total dry weight of vegetative crop organs, is initialized at zero.

```
DSTR = 0.
```

L220: The state variable DSTR, total dry weight of dead material above-ground, is initialized at zero.

```
TADRW = 0.
```

L230: The state variable TADRW, total above-ground dry weight, is initialized at zero.

```
SRR = 0.
```

L240: The state variable SRR, shoot:root ratio, is initialized at zero.

```
ARESPI = 0.
```

L250: The state variable ARESPI, initial dry weight of reserve carbohydrates at emergence, is initialized at zero.

```
ARESP = 0.
```

L260: The state variable ARESP, dry weight of reserve carbohydrates, is initialized at zero.

```
TLNI = 0.
```

L270: The state variable TLNI, initial tiller density at emergence, is initialized at zero.

```
TLNIX = 0.
```

L280: The state variable, TLNIX, auxiliary variable to save original value of TLNI, is initialized at zero.

```
TLN = 0.
```

L290: The state variable TLN, tiller density, is initialized at zero.

```
AWTL = 0.
```

L300: The state variable AWTL, average dry weight per tiller, is initialized at zero.

```
CALL RDSREA ('CHFTB',CHFTB)
CALL RDSREA ('RDRT' ,RDRT)
CALL RDSREA ('STCEF',STCEF)
CALL RDSREA ('EB'   ,EB)
CALL RDSREA ('CHFEB',CHFEB)
CALL RDSREA ('STCSF',STCSF)
CALL RDSREA ('SB'   ,SB)
CALL RDSREA ('CHFSB',CHFSB)
CALL RDSREA ('STCFF',STCFF)
CALL RDSREA ('FB'   ,FB)
CALL RDSREA ('CHFFB',CHFFB)
CALL RDSREA ('STCGF',STCGF)
CALL RDSREA ('GB'   ,GB)
```

```
CALL RDAREA ('TCTFT',TCTFT ,IARL1 ,ITCTFT)
CALL RDAREA ('RFTT' ,RFTT  ,IARL1 ,IRFTT)
```

L310: The following parameters are read from file, CHFTB, minimum carbohydrate requirement for completion of tiller formation, RDRT, relative death rate of non-fertile tillers, STCEF, maximum value of time constant for ear formation, EB, basic time constant for ear formation, CHFEB, minimum carbohydrate requirement for completion of ear formation, STCSF, maximum value of time constant for spikelet formation, SB, basic time constant for spikelet formation, CHFSB, minimum carbohydrate requirement for completion of spikelet formation, STCFF, maximum value of time constant for floret formation, FB, basic time constant for floret formation, CHFFB, minimum carbohydrate requirement for completion of fertile floret formation, STCGF, maximum value of time constant for grain formation, GB, basic time constant for grain formation. The arrays, TCTFT, time constant for tiller formation as a function of canopy temperature, RFTT, effect of temperature on grain set as a function of minimum temperature are defined.

```
DTLN = 0.
```

L310: The state variable DTLN, the total number of dead tillers, is initialized at zero.

```
TNNR = 0.
```

L320: The state variable TNNR, the total number of non-reproductive tillers, is initialized at zero.

```
LWDTL = 0.
```

L330: The state variable LWDTL, average dry weight of non-reproductive tillers, is initialized at zero.

```
SWDTL = 0.
```

L340: The state variable SWDTL, final dry weight of stem and sheaths of non-reproductive tillers, is initialized at zero.

```
EARN = 0.
```

L350: The state variable EARN, ear number, is initialized at zero.

```
NSPS = 0.
```

L360: The state variable NSPS, total number of spikelets, is initialized at zero.

```
FFNR = 0.
```

L370: The state variable, total number of fertile florets, is initialized at zero.

```
GRN = 0.
```

L380: The state variable GRN, grain density, is initialized at zero.

```
ARLFI = 0.
```

L390: The state variable ARLFI, initial green area of leaf blades at emergence, is initialized at zero.

```
ARLF = 0.
```

L400: The state variable ARLF, green area of leaf blades, is initialized at zero.

```
ARLFE = 0.
```

L410: The state variable ARLFE, total area of green and senesced leaf blades, is initialized at zero.

```
LAI = 0.
```

L420: The state variable LAI, leaf area index, is initialized at zero.

```
GRAI = 0.
CALL RDSREA ('FLFARM' ,FLFARM)
CALL RDSREA ('ARPEAR' ,ARPEAR)
CALL RDSREA ('MXSTAR' ,MXSTAR)
CALL RDSREA ('FLFARI' ,FLFARI)

CALL RDAREA ('AGEFT'  ,AGEFT ,IARL1 ,IAGEFT)

CALL RDSREA ('FRNN'   ,FRNN)
CALL RDSREA ('NDPAR'  ,NDPAR)
CALL RDSREA ('HTFAC'  ,HTFAC)
CALL RDSREA ('TCU'    ,TCU)

CALL RDSREA ('FNMIN'  ,FNMIN)
CALL RDSREA ('FNMXA'  ,FNMXA)
CALL RDSREA ('FNMXR'  ,FNMXR)
CALL RDSREA ('FNMXSA' ,FNMXSA)
CALL RDSREA ('FNMNSR' ,FNMNSR)
CALL RDSREA ('FNRTMX' ,FNRTMX)
CALL RDSREA ('FNRTMN' ,FNRTMN)
CALL RDSREA ('RN'     ,RN)
CALL RDSREA ('TCUD'   ,TCUD)
CALL RDSREA ('UMXR'   ,UMXR)
CALL RDSREA ('CF'     ,CF)
CALL RDSREA ('RRTORT' ,RRTORT)

CALL RDAREA ('RDRNT' ,RDRNT ,IARL1 ,IRDRNT)
CALL RDAREA ('BNT'   ,BNT   ,IARL1 ,IBNT)
CALL RDAREA ('LNT'   ,LNT   ,IARL1 ,ILNT)
CALL RDAREA ('PRNAGT',PRNAGT,IARL1 ,IPRAGT)
CALL RDAREA ('FNEXT' ,FNEXT ,IARL1 ,IFNEXT)
```

L430: The state variable GRAI, green area index, is initialized at zero. The parameters, FLFARM, minimum value of specific leaf area reached at start of stem extension, ARPEAR, maximum green area per individual ear, MXSTAR, maximum green area of stem and sheaths, FLFARI, specific leaf area at emergence are defined. The array AGEFT, describing the relative green area of ears as function of development stage is defined. The parameters FRNN, ratio between minimum concentration of nitrogen for unrestricted transpiration and maximum nitrogen concentration in leaf blades, NDPAR, variable indicating whether even (+1) or uneven (-1) nitrogen distribution is assumed, HTFAC, multiplication factor used to convert average nitrogen concentration in leaf blades to effective concentration, used in calculation of maximum assimilation rate, TCU, time constant for uptake of nitrogen, FNMIN, absolute minimum nitrogen concentration in leaf blades, FNMXA, range in maximum nitrogen concentration between young and mature leaf blades, FNMXR, maximum nitrogen concentration in mature leaf blades, FNMXSA, range in maximum nitrogen concentration between young and mature stem and sheaths, FNMNSR, maximum nitrogen concentration in mature stem and sheaths, FNRTMX, maximum nitrogen concentration in young roots, FNRTMN, maximum nitrogen concentration in mature roots, RN, residual nitrogen concentration in the roots, TCUD, time constant for uptake of nitrogen by diffusion, UXMR, maximum rate of nitrogen uptake by a closed canopy, CF, conversion factor from weight to relative root extension, RRTORT, basic relative rate of nitrogen turnover in vegetative biomass are defined. The arrays RDRNT, relative death rate of leaf blades due to nitrogen shortage as a function of relative nitrogen concentration, BNT, residual non-remobilizable concentration of nitrogen in leaf blades as a function of development stage, PRNAGT, potential rate of nitrogen accumulation in the grain as a function of canopy temperature, FNEXT, fraction of labile nitrogen exported from vegetative tissue to the grain as a function of average nitrogen concentration in the vegetative tissue are defined.

```
FNMAX = FNMXR+FNMXA
```

L440: Variable FNMAX, maximum concentration of nitrogen in leaf blades is calculated as the sum of FNMXR, the maximum nitrogen content in mature leaf blades and FNMXA, the range in maximum nitrogen concentration between young and mature leaf blades.

```
FRNVI = FNMAX
```

L450: Variable FRNVI, nitrogen concentration in leaf blades at emergence is initialized with the value of FNMAX, the maximum concentration of nitrogen in leaf blades.

```
RNFAC = FNMAX-FNMIN
```

L460: Variable RNFAC, range in nitrogen concentration in plant organs, used in calculations of maintenance respiration, is initialized with the difference between the maximum and the absolute minimum nitrogen concentration in leaf blades.

```
IFNRT = FNRTMN+(1.-DVSI)*(FNRTMX-FNRTMN)
```

L470: Variable IFNRT, initial concentration of nitrogen in the roots at emergence, is calculated as the sum of FNRTMN, the minimum concentration of nitrogen in the roots and the product of the range between minimum and maximum concentration, FNRTMX , of nitrogen in the roots and the difference between one and DVSI.

```
ANLVI = 0.
```

L480: The state variable ANLVI, initial weight of nitrogen in the leaf blades at emergence, is initialized at zero.

```
ANLV = 0.
```

L490: The state variable ANLV, weight of nitrogen in the leaf blades, is initialized at zero.

```
ANSTE = 0.
```

L500: The state variable ANSTE, weight of nitrogen in stem and leaf sheaths, is initialized at zero.

```
ANRTI = 0.
```

L510: The state variable ANRTI, initial weight of nitrogen in the roots at emergence, is initialized at zero.

```
ANRT = 0.
```

L520: The state variable ANRT, weight of nitrogen in the roots, is initialized at zero.

```
      ANGR = 0.
```

L530: The state variable ANGR, weight of nitrogen in the grain, is initialized at zero.

```
      NDSTR = 0.
```

L540: The state variable NDSTR, weight of nitrogen in dead above ground material, is initialized at zero.

```
      TNABM = 0.
```

L550: The state variable TNABM, total weight of nitrogen in above-ground material, is initialized at zero.

```
      INIV = 0.
```

L560: The state variable INIV, initial weight of nitrogen in the vegetation at emergence, is initialized at zero.

```
      AMAXN = 0.
      CLOSE (IUNITP, STATUS='DELETE')
```

L570: Variable AMAXN, maximum rate of gross CO_2 assimilation of a single leaf as function of nitrogen concentration and temperature, is initialized at zero. After this assignment, the plant datafile PLANT.DAT is closed.

```
      CALL RDINIT (IUNITS, IUNITO, FILES)
      TNLCH = 0.
```

L580: The routine RDINIT is called to initialize and read the soil datafile SOIL.DAT. The state variable TNLCH, total nitrogen loss by leaching, is initialized at zero.

```
      TNVOL = 0.
```

L590: The state variable TNVOL, total nitrogen loss by volatilization, is initialized at zero.

```
      CALL RDSREA ('REFCF'  ,REFCF)
      CALL RDSREA ('STBC'   ,STBC)
      CALL RDSREA ('GAMMA'  ,GAMMA)
      CALL RDSREA ('LHVAP'  ,LHVAP)
      CALL RDSREA ('RUNOFC' ,RUNOFC)
      CALL RDSREA ('WCLIM'  ,WCLIM)
      CALL RDSREA ('PROP'   ,PROP)
      CALL RDSREA ('CONVER' ,CONVER)
      CALL RDSREA ('WSCONV' ,WSCONV)

      CALL RDAREA ('RADTBA' ,RADTBA ,IARL1 ,IRADTA)
      CALL RDAREA ('RADTBB' ,RADTBB ,IARL1 ,IRADTB)
      CALL RDAREA ('RADTBC' ,RADTBC ,IARL1 ,IRADTC)
      CALL RDAREA ('RADTBD' ,RADTBD ,IARL1 ,IRADTD)
      CALL RDAREA ('RADTBE' ,RADTBE ,IARL1 ,IRADTE)
      CALL RDAREA ('RADTBF' ,RADTBF ,IARL1 ,IRADTF)
      CALL RDAREA ('RADTBG' ,RADTBG ,IARL1 ,IRADTG)
      CALL RDAREA ('RADTBH' ,RADTBH ,IARL1 ,IRADTH)
      CALL RDAREA ('RADTBI' ,RADTBI ,IARL1 ,IRADTI)
```

```
      CALL RDAREA ('RADTBJ' ,RADTBJ ,IARL1 ,IRADTJ)

      CALL RDAREA ('WREDT'  ,WREDT  ,IARL1 ,IWREDT)
      CALL RDAREA ('EDPTFT' ,EDPTFT ,IARL1 ,IEDPTN)
      CALL RDAREA ('REDFDT' ,REDFDT ,IARL1 ,IRDFDT)
      CALL RDAREA ('IRRT'   ,IRRT   ,IARL1 ,IIRRT)
      CALL RDAREA ('ROSPT'  ,ROSPT  ,IARL1 ,IROSPT)
```

L600: The parameters REFCF, reflection coefficient of open water for short wave radiation, STBC, Stefan-Boltzmann constant, GAMMA, psychrometric constant, LHVAP, latent heat of vaporization, RUNOFC, runoff coefficient, WCLIM, volumetric soil moisture content at air dryness, PROP, proportionality factor for calculation of soil moisture contribution to soil surface evaporation per soil compartment, CONVER, variable to convert radiation from cal cm^{-2} to J m^{-2}, WSCONV, convert windspeed from m s^{-1} to km d^{-1} are defined. The arrays RADTBA-RADTBJ, total visible radiation as a function of day number and latitude, WREDT, reduction factor on actual rate of water uptake as a function of fraction available moisture per soil compartment, EDPTFT, reduction factor on soil moisture uptake due to low available moisture as a function of fraction available moisture per soil compartment, REDFDT, reduction factor on soil surface evaporation as a function of dimensionless moisture content in top soil compartment, IRRT, effective irrigation as a function of day number, ROSPT, reduction factor on soil moisture uptake as a function of nitrogen concentration per soil compartment are defined.

```
      TEVAP = 0.
```

L610: State variable TEVAP, total seasonal soil surface evaporation, is initialized at zero.

```
      TOTINF = 0.
```

L620: State variable TOTINF, total seasonal infiltration, is initialized at zero.

```
      TDRAIN = 0.
```

L630: State variable TDRAIN, total loss of soil moisture by drainage beyond potential rooting zone, is initialized at zero.

```
      TNTRMT = 0.
```

L640: State variable TNTRMT, total nitrogen fertilizer application, is initialized at zero.

```
      RANO3 = 0.

      CALL RDSINT ('ILAYER' ,ILAYER)
      CALL RDAREA ('THCKN'  ,THCKN ,IARL1 ,ITHCKN)
      CALL RDAREA ('WLTPT'  ,WLTPT ,IARL1 ,IWLTPT)
      CALL RDAREA ('FLDCP'  ,FLDCP ,IARL1 ,IFLDCP)

      CALL RDAREA ('DRFA'   ,DRFA  ,IARL1 ,IDRFA)
      CALL RDSREA ('FOMI'   ,FOMI)
      CALL RDSREA ('HUMI'   ,HUMI)
      CALL RDSREA ('NHUMI'  ,NHUMI)

      CALL RDAREA ('IBIOMN' ,IBIOMN ,IARL1   ,IIBIOM)
      CALL RDAREA ('IAS'    ,IAS    ,IARL1   ,IIAS)
      CALL RDAREA ('DFFOM'  ,DFFOM  ,IARL1+1 ,IDFFOM)
```

L650: State variable RANO3, proportion of nitrate in total inorganic nitrogen in top soil compartment, is initialized at zero. The following variables and parameters are read from file: the integer variable ILAYER, number of soil compartments, arrays THCKN, thickness, WLTPT, volumetric soil moisture content at wilting point, FLDCP, volumetric soil moisture content at field capacity, DRFA, factor defining initial soil moisture content as fraction of soil moisture content at wilting point, each per soil compartment, FOMI, initial total weight of fresh organic material in soil profile, HUMI, initial total weight of stable organic material in soil profile, NHUMI, initial total weight of nitrogen in stable organic material in soil profile, arrays IBIOMN, initial weight of nitrogen in microbial biomass, IAS, initial weight of mineral nitrogen in a soil compartment and DFFOM, distribution factor for fresh and stable organic material in soil profile, each per soil compartment.

```
      TDT(1) = 0.
      DO 20 I = 2,ILAYER+1
         TDT(I) = TDT(I-1)+THCKN(I-1)
20     CONTINUE
```

L660: Depths of top of consecutive compartments are calculated by addition of the thickness of compartments and stored in the array TDT(I).

```
      TNINT = 0.
      DO 30 I = 1,ILAYER
         TNINT = TNINT+IAS(I)
30     CONTINUE
```

L670: The state variable TNINT, initial weight of inorganic nitrogen in the soil profile, is calculated by addition of the weights of inorganic nitrogen in the separate soil compartments.

```
      BIOMNI = 0.
      DO 40 I = 1,ILAYER
         BIOMNI = BIOMNI+IBIOMN(I)
         DBIOMN(I) = 0.
40     CONTINUE

      CALL RDSREA ('RDLIGN' ,RDLIGN)
      CALL RDSREA ('RDCELL' ,RDCELL)
      CALL RDSREA ('RDCAPR' ,RDCAPR)
      CALL RDSREA ('FRNF'   ,FRNF)
```

L680: The state variable BIOMNI, initial weight of nitrogen in microbial biomass in the soil profile, is calculated by addition of the weights of nitrogen in microbial biomass in the separate soil compartments. The variables RDLIGN, relative rate of decomposition of lignin under optimum conditions, RDCELL, relative rate of decomposition of cellulose and hemicellulose under optimum conditions, RDCAPR, relative rate of decomposition of easily decomposable carbohydrates and proteins under optimum conditions and FRNF, concentration of nitrogen in fresh organic material, are read from file.

```
      DO 50 I = 1,ILAYER
         WATER(I)  = DRFA(I)*WLTPT(I)*THCKN(I)
         AWATER(I)  = AMAX1(0.,WATER(I)-THCKN(I)*WLTPT(I))
         STORC(I)  = FLDCP(I)-WLTPT(I)
         IFOM(I)   = DFFOM(I)*FOMI
         FOM(I)    = IFOM(I)
         FON(I)    = IFOM(I)*FRNF
         NHUM(I)   = DFFOM(I)*NHUMI
         HUM(I)    = DFFOM(I)*HUMI
         BIOMN(I)  = IBIOMN(I)
         ASLT(I)   = IAS(I)
50    CONTINUE
```

L690: In this section a number of arrays for soil state variables per soil compartment are initialized, WATER, amount of soil moisture is calculated as the product of the dryness factor, DRFA, volumetric soil moisture content at wilting point, WLTPT and THCKN, thickness of the soil compartment, AWATER, amount of available soil moisture is calculated as the difference between the amount of soil moisture and the product of the thickness of the soil compartment and the volumetric soil moisture content at wilting point, STORC, storage capacity for available soil moisture is calculated as the difference between volumetric soil moisture content at field capacity, FLDCP and at wilting point, WLTPT, IFOM, the weight of initial fresh organic material, is calculated as the product of the distribution factor for fresh and stable organic material, DFFOM and FOMI, the total initial weight of fresh organic material in the soil profile, FOM is initialized with the values of IFOM, FON, the amount of nitrogen in fresh organic material is calculated as the product of the initial weight of fresh organic material, IFOM and FRNF, concentration of nitrogen in fresh organic material, NHUM, weight of nitrogen in stable organic material is calculated as the product of DFFOM, distribution factor of fresh organic material and NHUMI, initial total weight of nitrogen in stable organic material in soil profile, HUM, weight of stable organic material is calculated as the product of the same distribution factor and HUMI, initial total weight of stable organic material in the soil profile, BIOMN, weight of nitrogen in the microbial biomass is initialized with the values of the initial weight of the microbial biomass, IBIOMN, ASLT, the weight of mineral nitrogen is initialized with the values of IAS, initial weight of mineral nitrogen.

```
      WTOT = 0.
```

L700: The variable WTOT, total amount of soil moisture in the soil profile, is initialized at zero.

```
      NTOT = 0.
```

L710: The variable NTOT, total weight of inorganic nitrogen in the soil profile, is initialized at zero.

```
      BIOMNT = 0.
```

L720: The variable BIOMNT, total weight of nitrogen in microbial biomass in the soil profile, is initialized at zero.

```
      TFOM = 0.
```

L730: The variable TFOM, total weight of fresh organic material in the soil profile, is initialized at zero.

```
      TFON = 0.
```

L740: The variable TFON, total weight of nitrogen in fresh organic material in the soil profile, is initialized at zero.

```
      HUMT = 0.
```

L750: The variable HUMT, total weight of stable organic material in the soil profile, is initialized at zero.

```
      NHUMT = 0.
```

L760: The variable NHUMT, total weight of nitrogen in stable organic material in the soil profile, is initialized at zero.

```
      DO 60 I = 1,ILAYER
        IF (TDT(I)+0.5.GT.MXRTD) GO TO 60
```

L770: A number of variables are calculated with total values over the soil profile. If the depth of the upper boundary of a certain soil compartment plus 0.5 cm exceeds the maximum rooting depth of the crop, the addition stops. In other words, this procedure calculates totals over the maximum rootable soil profile.

```
      WTOT = WTOT+WATER(I)
```

L780: The variable WTOT, total amount of soil moisture in the soil profile, is calculated as the sum of the values of soil moisture in the relevant soil compartments.

```
      BIOMNT = BIOMNT+BIOMN(I)
```

L790: The variable BIOMNT, total weight of nitrogen in microbial biomass in the soil profile, is calculated as the sum of the weight of nitrogen in microbial biomass in the relevant soil compartments.

```
      TFOM = TFOM+FOM(I)
```

L800: The variable TFOM, total weight of fresh organic material in the soil profile, is calculated as the sum of the weight of fresh organic material in the relevant soil compartments.

```
      TFON = TFON+FON(I)
```

L810: The variable TFON, total weight of nitrogen in fresh organic material in the soil profile, is calculated as the sum of the weight of nitrogen in fresh organic material in the relevant soil compartments.

```
      HUMT = HUMT+HUM(I)
```

L820: The variable HUMT, total weight of stable organic material in the soil profile, is calculated as the sum of the weight of stable organic material in the relevant soil compartments.

```
      NHUMT = NHUMT+NHUM(I)
60    CONTINUE
```

L830: The variable NHUMT, total weight of nitrogen in stable organic material in the soil profile, is calculated as the sum of the weight of nitrogen in stable organic material, in the relevant soil compartments.

```
      CALL RDSREA ('NCR'     ,NCR)
      CALL RDSREA ('NCIW'    ,NCIW)
```

```
      CALL RDSREA ('CNRMIC' ,CNRMIC)
      CALL RDSREA ('DMINR'  ,DMINR)
      CALL RDSREA ('FCELL'  ,FCELL)
      CALL RDSREA ('FLIGN'  ,FLIGN)
      CALL RDSREA ('FNIMH'  ,FNIMH)
      CALL RDSREA ('FRC'    ,FRC)
      CALL RDSREA ('LNH4'   ,LNH4)
      CALL RDSREA ('MRGRB'  ,MRGRB)
      CALL RDSREA ('NCH'    ,NCH)
      CALL RDSREA ('NH4FP'  ,NH4FP)
      CALL RDSREA ('RRMIC'  ,RRMIC)
      CALL RDSREA ('TCMG'   ,TCMG)
      CALL RDSREA ('TCN'    ,TCN)
      CALL RDSREA ('TCV'    ,TCV)

      CALL RDAREA ('IRRT'   ,IRRT   ,IARL1 ,IIRRT)
      CALL RDAREA ('MFT'    ,MFT    ,IARL1 ,IMFT)
      CALL RDAREA ('DISTF'  ,DISTF  ,IARL1 ,IDISTF)

      CALL RDSINT ('NF'     ,NF)
      CALL RDAREA ('NAPDAY' ,NAPDAY,IARL1 ,IAPDAY)
      CALL RDAREA ('NTRMNT' ,NTRMNT,IARL1 ,INTRMT)
```

L850: The following parameters are read from file: NCR, concentration of nitrogen in rainwater, NCIW, concentration of nitrogen in irrigation water, CNRMIC, C/N ratio of microbial biomass, DMINR, relative decomposition rate of stable organic material under optimum conditions, FCELL, initial fraction of cellulose and hemicellulose in fresh organic material, FLIGN, initial fraction of lignin in fresh organic material, FNIMH, concentration of nitrogen in decomposing fresh organic material and microbial biomass immobilised in stable organic material, FRC, fraction of carbon in fresh organic material, LNH4, limiting concentration for volatilization of ammonia, MRGRB, maximum relative growth rate of microbial biomass, NCH, nitrogen content of stable organic material, NH4FP, a variable which defines the type of fertilizer used, i.e. ammonium (1) or nitrate (0), RRMIC, relative rate of maintenance respiration of microbial biomass, TCMG, time constant for growth of microbial biomass, TCN, time constant for nitrification, TCV, time constant for volatilization of ammonia. The arrays IRRT, effective irrigation as a function of day number, MFT, reduction factor on the rate of decomposition of organic material as a function of moisture content in the compartment and DISTF, vertical fertilizer distribution factor are read. Finally a number of variables are read which define nitrogen fertilizer management, variable NF, defining the number of fertilizer applications, array NAPDAY, defining the day numbers of nitrogen application and the array NTRMNT which defines the nitrogen fertilizer application rate.

```
      IF (NF.NE.IAPDAY.OR.NF.NE.INTRMT) THEN
         WRITE (*,'(A,/,A)')
   $       ' There have been errors in the file: SOIL.DAT',
   $       ' Check number of data on nitrogen management'
         STOP
      END IF
```

L851: In this section it is checked whether the specified number of application days and application rates equals NF, the number of fertilizer applications.

```
      NNA = 1
```

L855: The variable NNA, number of the first forthcoming nitrogen application, is initialized at 1.

```
      CARBAL = 0.
```

L860: The state variable CARBAL, carbon balance of the soil, is initialized at zero.

```
      NBAL = 0.
```

L870: The state variable NBAL, nitrogen balance of the soil, is initialized at zero.

```
      TSI = 5.*(TMIN + TMAX)
```

L880: The state variable TSI, initial soil temperature is calculated as the product of 5 times the sum of the minimum and maximum temperature. The reason is that TS is defined as a ten-day running average.

```
      TS = 0.1*TSI
```

L890: The state variable TS, soil temperature, is calculated as 0.1 times TSI, initial soil temperature.

```
      DO 70 I = 1,12
         TMPCX(I) = 0.
70     CONTINUE

      CALL RDAREA('TFT'  ,TFT  ,IARL1 ,ITFT)
      CALL RDAREA('TECT' ,TECT ,IARL1 ,ITECT)
```

L900: The array TMPCX, soil temperatures during a ten-day period, is initialized at zero. The arrays TFT, reduction factor on decomposition of organic material as function of soil temperature, and TECT, reduction factor on root conductivity as function of soil temperature, are defined.

```
      DTGA = 0.
```

L910: The rate variable DTGA, daily potential gross assimilation of a canopy, is initialized at zero.

```
      DGAS = 0.
```

L920: The rate variable DGAS, actual daily total gross assimilation of a canopy, is initialized at zero.

```
      LG = 0
      CLOSE (IUNITS, STATUS='DELETE')
```

L930: Variable LG, a counter to designate leaf classes by day of initiation, is initialized at zero. With the closing of the soil datafile, the initialization section is completed.

5.4. Rate calculation section of module

```
      ELSE IF (ITASK.EQ.2) THEN
```

L940: Start of simulation loop.

```
      IF ( DATE.GE.SOWD ) THEN

        IF (PUSHD.GE.0.5 .OR. TGERD.GE.9.) THEN
           PUSHD = 0.
           TGERD = 100.
           GO TO 500
        END IF

        WGER = 0.

        DO 450 I = 1,3
           WGER = WGER + WATER(I)
 450      CONTINUE

        GERD = 0.
        SDAY = 0.

        IF (WGER.GT.100.*WLTPT(1)*1.2) THEN
           GERD = 1.
        ELSE IF (GERD.EQ.0..AND. TGERD.GT.4.) THEN
           SDAY = 1.
        END IF

        TGERD = TGERD + GERD
        TSDAY = TSDAY + SDAY

        IF (TSDAY.GT.6.) THEN
           TGERD = 0.
           TSDAY = 0.
        ELSE IF (TGERD.GT.5.) THEN
           PUSHD = 1.
           GERDAT = DATE
        END IF

        END IF
500     CONTINUE
```

L950: Procedure to evaluate germination and emergence. This procedure is evaluated daily from the day of sowing onwards. If there are no favourable conditions for emergence only the carbon-, water- and nitrogen balance of the soil are simulated. If germination is not completed (PUSHD equals zero) or the number of days favourable for germination is less than 9, the germination and emergence procedure will be continued. If the number of days with favourable germination conditions is greater than or equals 9 then TGERD, cumulative number of days with favourable conditions for germination is set to 100, and PUSHD will be reset to zero, the procedure will be continue after label 500. The amount of soil moisture in the upper three soil compartments is calculated if the number of days favourable for germination is less than 9. Then the variables GERD and SDAY, the latter indicating whether a day counts for deterioration of germinating seeds (1) or not (0) are initialized at zero, GERD is assigned the value one if the amount of soil moisture in the upper three soil layers exceeds 1.2 times the moisture content at wilting point. The variable SDAY is assigned the value one if GERD was not assigned the value one and cumulative number of days favourable for germination exceeds 4. This formulation expresses the notion that if germination has proceeded less than four days the process can halt temporarily and be resumed if conditions become favourable again. If germination has proceeded for more than four days, irreversible changes have taken place, and germination fails. The values of GERD and SDAY (0 or 1) are then added to the cumulative state variables TGERD and TSDAY, respectively. The values of TGERD and TSDAY are assigned zero and plant germination and emergence cannot take place anymore if the cumulative number of seed deterioration days, TSDAY exceeds 6. If the value of TSDAY is less than or equals 6 and the value of TGERD, the total number of favourable days for germination exceeds 5, then the rate variable PUSHD is assigned the value one and the state variable GERDAT is assigned the value of the variable DATE.

```
      IF (NINT(PUSHD).EQ.1) THEN

      CALL RDINIT (IUNITP, IUNITO, FILEP)
```
L970: Section in which plant state variables are initialized from file PLANT.DAT on unit IUNITP, if PUSHD equals 1.

```
      CALL RDSREA ('WLVSI'   ,WLVSI)
      WLVSI = SWDF*WLVSI
```
L980: The variable WLVSI, dry weight of live leaf blades at emergence, is initialized and multiplied with the variable SWDF which is optionally used to set different sowing densities.

```
      CALL RDSREA ('WRTI'    ,WRTI)
      WRTI = SWDF*WRTI
```
L990: The variable WRTI, dry weight of live roots at emergence, is initialized and multiplied with the variable SWDF (L980).

```
      CALL RDSREA ('IRTD'    ,IRTD)
```
L1000: The variable IRTD, rooting depth at emergence, is read from file.

```
      ARLFI = WLVSI*FLFARI
```
L1010: The variable ARLFI, green area of leaf blades at emergence, is calculated as the product of WLVSI, dry weight of live leaf blades at emergence, and FLFARI, specific leaf area at emergence.

```
      ARESPI = RESLI*(WLVSI+WRTI)
```
L1020: The variable ARESPI, initial dry weight of reserve carbohydrates in the plant, is calculated as the product of RESLI, actual concentration of reserve carbohydrates, and the sum of dry weight of live leaf blades and live roots at emergence.

```
      CALL RDSREA ('TLNI'    ,TLNI)
      TLNI = SWDF*TLNI
```

L1030: The variable TLNI, initial tiller density is read from file and multiplied with the variable SWDF (L980).

```
TLNIX = TLNI
```

L1040: The variable TLNIX, auxiliary variable to save original value of TLNI is assigned the value of TLNI.

```
WSTEM = 0.
```

L1050: The state variable WSTEM, dry weight of live stem and sheaths, is initialized at zero.

```
WGR = 0.
```

L1060: The state variable WGR, dry weight of the grain, is initialized at zero.

```
ANLVI = WLVSI*FRNVI
```

L1070: The variable ANLVI, weight of nitrogen in leaf blades at emergence, is calculated as the product of the dry weight of live leaf blades at emergence, WLVSI and FRNVI, concentration of nitrogen in leaf blades at emergence.

```
ANRTI = WRTI*IFNRT
CLOSE (IUNITP, STATUS='DELETE')
```

L1080: The variable ANRTI, weight of nitrogen in the roots at emergence is calculated as the product of dry weight of live roots at emergence, WRTI and IFNRT, initial concentration of nitrogen in the roots. Then the temporary file on UNITP is closed and deleted.

```
RAD = RADCF*RADI
```

L1090: The variable RAD, total global radiation in kJ m^{-2} d^{-1}, is calculated as the product of variable RADCF, conversion factor on global radiation and the variable RADI, the daily global radiation supplied by the subroutine WEATHR in the MAIN program.

```
EAVT = TMAX-0.25*(TMAX-TMIN)
```

L1100: The variable EAVT, effective air temperature during the daytime period, is calculated as the daily maximum temperature minus 0.25 times the difference between daily maximum and daily minimum temperature.

```
RAIN = RAIN*RAINF
```

L1110: The variable RAIN, rain intensity in mm d^{-1} is calculated as the product of RAIN (!), the rain intensity supplied by the subroutine WEATHR in the MAIN program and the factor RAINF, which enables simulation runs with different rainfall intensities.

```
CALL ASTRO (DATE, LAT, SC, DS0, SINLD, COSLD, DAYL, DSINB, DSINBE)
```

L1120: The subroutine ASTRO is called to calculate astronomical daylength and some intermediate variables in the calculation of daily total gross assimilation from day number, DATE and latitude, LAT.

```
TMPA = (TMIN+TMAX)*0.5
```

L1170: The variable TMPA, average daily air temperature in °C is calculated as the sum of the minimum temperature, TMIN and the maximum temperature, TMAX times 0.5.

```
SVPA = 4.58*EXP(17.4*TMPA/(TMPA+239.))
```

L1180: The variable SVPA, average daily saturated vapour pressure of the atmosphere in mm Hg, is calculated from TMPA (L1170), the average daily air temperature .

```
VPA = 1000.*VAPOUR/133.28
```

L1190: The variable VPA, actual average daily vapour pressure in mm Hg is calculated from VAPOUR, vapour pressure in kPa, supplied by the subroutine WEATHR in the main program.

```
DPT = AMIN1(239./(17.381/LOG(VAPOUR/0.6107)-1.),TMPA)
```

L1200: The variable DPT, average daily dew point temperature, is calculated from the variables VAPOUR (L1190) and TMPA (L1170). The calculated value of dew point temperature is restrained with an AMIN1 function to the average daily temperature to avoid condensation conditions.

```
SVPAM =6.11*EXP(17.4*EAVT/(EAVT+239.))
```

L1210: The variable SVPAM, average effective saturated vapour pressure during daytime in mbar is calculated from the variable EAVT, effective air temperature during daytime.

```
VPAM = AMIN1(SVPAM-0.1,1.33*VPA)
```

L1220: The variable VPAM, actual vapour pressure during daytime in mbar is calculated as the minimum of the average effective saturated vapour pressure minus 0.1 and the actual average daily vapour pressure of the atmosphere in mm Hg times 1.33.

```
WSA = 1.333E5*WSCONV*WIND
```

L1230: The variable WSA, average daily wind speed during daytime in cm d^{-1} is calculated as the product of 1.333 times 10^5 and WSCONV, wind speed conversion factor and WIND, average daily windspeed in m s^{-1} supplied by the subroutine WEATHR in the MAIN program.

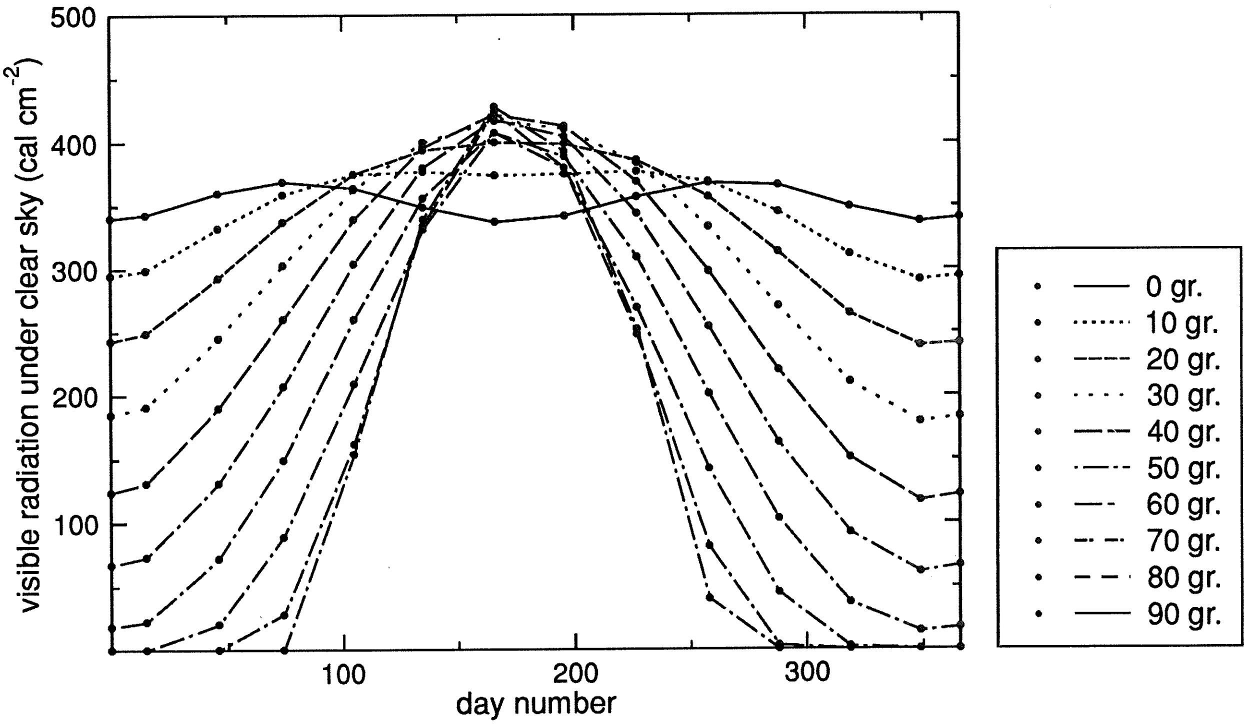

Figure 6 **The functions RADTBA till RADTBJ, total visible radiation under a clear sky as a function of latitude and day number.**

```
IF (LAT.GE.0..AND.LAT.LT.10.) THEN
  A = LINT(RADTBA,IRADTA,DATE)
  B = LINT(RADTBB,IRADTB,DATE)
  RADCL = A+(LAT)*(B-A)/10.
ELSE IF (LAT.GE.10..AND.LAT.LT.20.) THEN
  A = LINT(RADTBB,IRADTB,DATE)
  B = LINT(RADTBC,IRADTC,DATE)
  RADCL = A+(LAT-10.)*(B-A)/10.
ELSE IF (LAT.GE.20..AND.LAT.LT.30.) THEN
  A = LINT(RADTBC,IRADTC,DATE)
  B = LINT(RADTBD,IRADTD,DATE)
  RADCL = A+(LAT-20.)*(B-A)/10.
ELSE IF (LAT.GE.30..AND.LAT.LT.40.) THEN
  A = LINT(RADTBD,IRADTD,DATE)
  B = LINT(RADTBE,IRADTE,DATE)
  RADCL = A+(LAT-30.)*(B-A)/10.
ELSE IF (LAT.GE.40..AND.LAT.LT.50.) THEN
  A = LINT(RADTBE,IRADTE,DATE)
  B = LINT(RADTBF,IRADTF,DATE)
  RADCL = A+(LAT-40.)*(B-A)/10.
ELSE IF (LAT.GE.50..AND.LAT.LT.60.) THEN
  A = LINT(RADTBF,IRADTF,DATE)
  B = LINT(RADTBG,IRADTG,DATE)
  RADCL = A+(LAT-50.)*(B-A)/10.
```

```
ELSE IF (LAT.GE.60..AND.LAT.LT.70.) THEN
  A = LINT(RADTBG,IRADTG,DATE)
  B = LINT(RADTBH,IRADTH,DATE)
  RADCL = A+(LAT-60.)*(B-A)/10.
ELSE IF (LAT.GE.70..AND.LAT.LT.80.) THEN
  A = LINT(RADTBH,IRADTH,DATE)
  B = LINT(RADTBI,IRADTI,DATE)
  RADCL = A+(LAT-70.)*(B-A)/10.
ELSE IF (LAT.GE.80..AND.LAT.LT.90.) THEN
  A = LINT(RADTBI,IRADTI,DATE)
  B = LINT(RADTBJ,IRADTJ,DATE)
  RADCL = A+(LAT-80.)*(B-A)/10.
END IF
```

L1240: The variable RADCL, total visible radiation in cal cm^{-2} d^{-1} under a clear sky as a function of latitude (Fig. 6) and day number, is calculated by two-dimensional linear interpolation with LAT, latitude in decimal degrees and DATE, day number as independent variables.

```
DGRCL = 2.*RADCL*CONVER
```

L1250: The variable DGRCL, daily total global radiation under a completely clear sky in J m^{-2} d^{-1} is calculated as 2 times the value of RADCL, total visible radiation under a completely clear sky in cal cm^{-2} times CONVER, a factor to convert from cal cm^{-2} to J m^{-2}.

```
DGROV = 0.2*DGRCL
```

L1260: The variable DGROV, daily total global radiation under an overcast sky in J m^{-2} d^{-1} is calculated as 0.2 times DGRCL (L1250).

```
LFOV = LIMIT(0.,1.,1.-(RAD-DGROV)/(DGRCL-DGROV+REANOT(DGRCL-DGROV)))
```

L1270: The variable LFOV, fraction of time that the sky is overcast, is calculated as one minus the fraction clear. The fraction clear is calculated as the difference between the actual global radiation, RAD, minus the global radiation on a completely overcast day, DGROV, divided by the difference between the global radiation on a completely clear day minus the global radiation on a completely overcast day. The variable LFOV is restrained between zero and one by a LIMIT function.

```
LWR = STBC*(TMPA+273.)**4*(0.58-0.09*SQRT(VPA))*(1.-0.9*LFOV)
```

L1280: The variable LWR, net outgoing longwave radiation on a daily basis used in the Penman equation, is calculated from STBC, Stefan-Boltzmann constant, TMPA, average daily air temperature, VPA, average daily vapour pressure of the atmosphere and LFOV, fraction overcast.

```
ELWR = STBC*(EAVT+273.)**4*(0.58-0.09*SQRT(VPAM))*(1.-0.9*LFOV)*DAYL/24.
```

L1290: The variable ELWR, net outgoing longwave radiation is calculated using the previous equation and EAVT, the effective air temperature during daytime and is divided by the ratio DAYL, astronomical daylength, to length of day and night, 24., to estimate the long-wave radiation during daytime, instead of the average daily value.

```
HNOT = AMAX1(0.,0.75*RAD-ELWR)
```

L1300: The variable HNOT, net global radiation, is calculated as 0.75 times daily global radiation, RAD supplied by the subroutine WEATHR in the MAIN program, minus the net outgoing long wave radiation. The value of HNOT is restrained to zero with an AMAX1 function.

```
SLOPE = 17.4*SVPAM*(1.-EAVT/(EAVT+239.))/(EAVT+239.)
```

L1310: The variable SLOPE, slope of the saturated vapour pressure curve at effective daytime temperature in mbar °C^{-1}, is calculated from the saturated vapour pressure at effective air temperature during daytime.

```
HRAD = RAD/DAYL
```

L1320: The variable HRAD, average hourly radiation intensity during daytime in J m^{-2} is calculated as RAD, daily total global radiation divided by DAYL, astronomical daylength in hours.

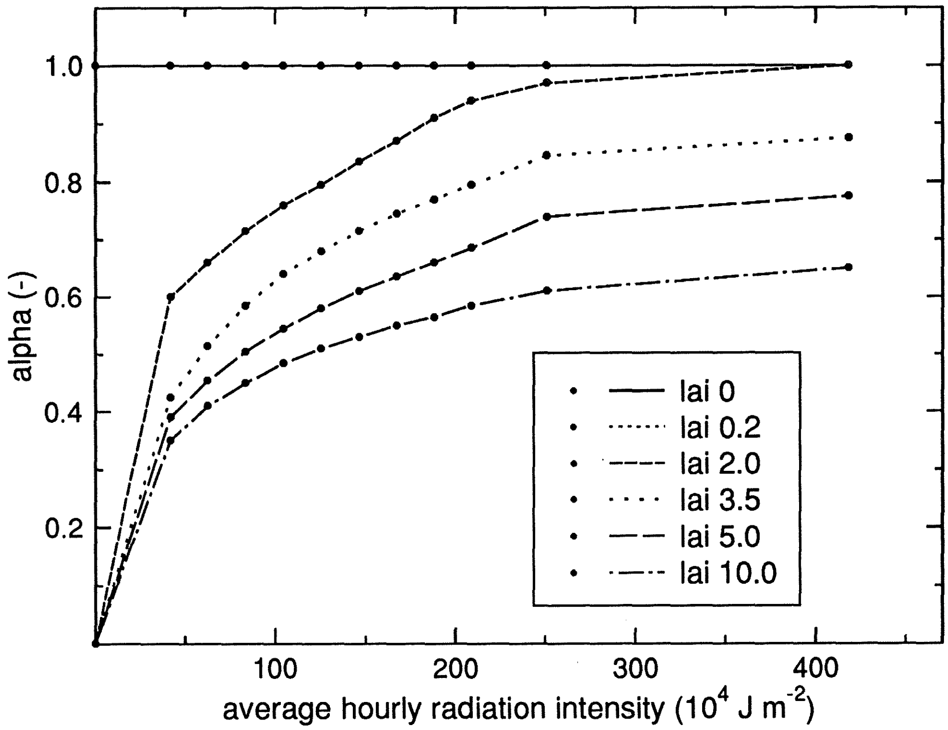

Figure 7 The functions ALPHTA till ALPHTF, proportionality factors in the calculation of the contribution of the drying power of the air to crop transpiration as function of average hourly radiation intensity at 6 different levels of LAI (function for LAI-0 coincide with function for LAI-0.2).

```
IF (LAI.GE.0..AND.LAI.LE.0.2) THEN
  A = LINT (ALPHTA,IALPTA,HRAD)
  B = LINT (ALPHTB,IALPTB,HRAD)
  ALPHA = A+(LAI)*(B-A)/0.2
ELSE IF (LAI.GT.0.2.AND.LAI.LE.2.0) THEN
  A = LINT (ALPHTB,IALPTB,HRAD)
  B = LINT (ALPHTC,IALPTC,HRAD)
  ALPHA = A+(LAI-0.2)*(B-A)/1.8
ELSE IF (LAI.GT.2.0.AND.LAI.LE.3.5) THEN
  A = LINT (ALPHTC,IALPTC,HRAD)
  B = LINT (ALPHTD,IALPTD,HRAD)
```

```
    ALPHA = A+(LAI-2.0)*(B-A)/1.5
  ELSE IF (LAI.GT.3.5.AND.LAI.LE.5.0) THEN
    A = LINT (ALPHTD,IALPTD,HRAD)
    B = LINT (ALPHTE,IALPTE,HRAD)
    ALPHA = A+(LAI-3.5)*(B-A)/1.5

  ELSE IF (LAI.GT.5.0.AND.LAI.LE.10.0) THEN
    A = LINT (ALPHTE,IALPTE,HRAD)
    B = LINT (ALPHTF,IALPTF,HRAD)
    ALPHA = A+(LAI-5.0)*(B-A)/5.0
  ELSE IF (LAI.GE.10.) THEN
    ALPHA = LINT(ALPHTF,IALPTF,HRAD)
  END IF
```

L1330: The variable ALPHA, proportionality factor on calculation of contribution of drying power of the air to crop transpiration, is calculated from LAI, leaf area index and HRAD, average hourly radiation intensity during daytime and the functions ALPHTA till ALPHTF (Fig. 7).

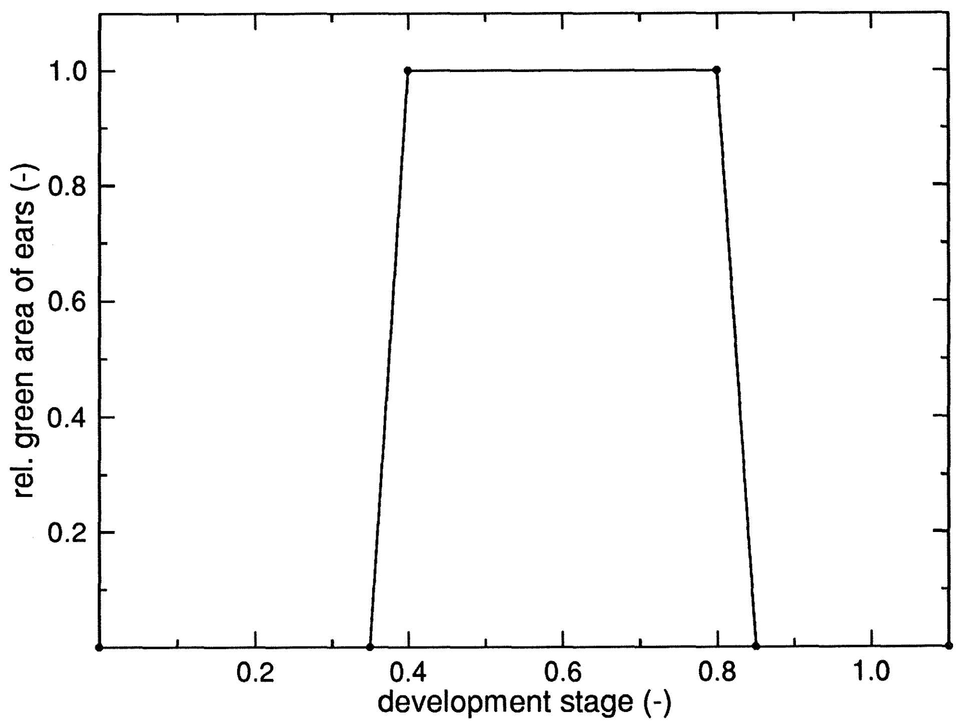

Figure 8 The function AGEFT, relative green area of ears as function of development stage.

```
  EARAR = EARN*ARPEAR*LINT(AGEFT,IAGEFT,DVS)
```

L1340: The variable EARAR, green area of the ears is calculated as the product of EARN, ear number, ARPEAR, maximum green area per individual ear and the relative green area of ears which is calculated by linear interpolation in the function AGEFT (Fig. 8), with development stage, DVS, as independent variable.

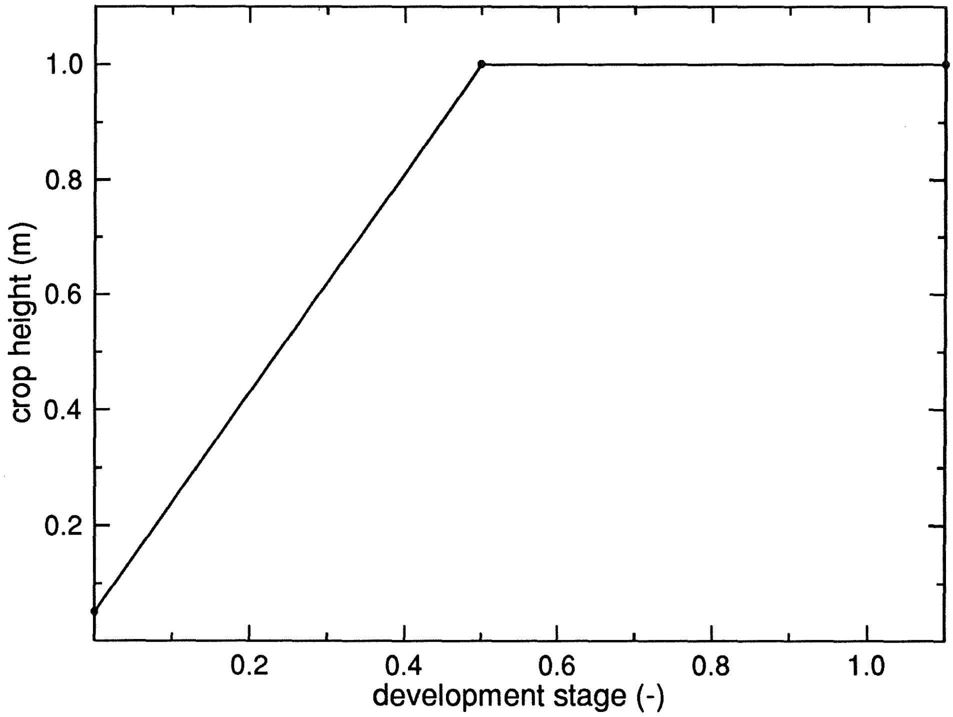

Figure 9 **The function CROHTB, height of the crop as a function of development stage.**

```
CROPHT = LINT(CROHTB,ICRHTB,DVS)
```

L1350: The variable CROPHT, height of the crop, is calculated by linear interpolation in the function CROHTB (Fig. 9) with development stage, DVS, as independent variable.

```
STAREA = WSTEM/5000.*CROPHT*MXSTAR
```

L1360: The variable STAREA, green area of stems, is calculated from WSTEM, dry weight of live stem and sheaths, divided by 5000 times CROPHT, crop height in meters times MXSTAR, maximum green area of stem and sheaths.

```
GRAI = 1.E-4*(ARLF+EARAR+STAREA)
```

L1370: The variable GRAI, green area index of the canopy, including ear and stem is calculated as the sum of ARLF, green area of the leaf blades, EARAR, green area of the ears and STAREA, green area of the stem and sheaths, times 10^{-4} to convert to m^2 m^{-2}.

```
FRNV = ANLV/(WLVS+REANOT(WLVS))
```

L1380: The variable FRNV, concentration of nitrogen in live leaf blade tissue, is calculated as the weight of nitrogen in the leaf blades, ANLV, divided by the dry weight of live leaf blades, WLVS.

```
EFRNV = INSW(DVSV-DVSAN,FRNV,INSW(NDPAR,FRNV,FRNV*HTFAC))
```

L1390: The variable EFRNV, effective nitrogen concentration for assimilation, is calculated with an INSW function. If DVSV, pre-anthesis development stage, is less than the value of DVSAN, development stage at anthesis, EFRNV is assigned the value of FRNV. After anthesis, i.e. when DVSV-DVSAN is equal to or greater than zero, EFRNV will be assigned the value of FRNV if NDPAR equals -1, else when NDPAR equals one, the product of FRNV and the distribution factor HTFAC.

```
IF (PUSHDI.GT.0) THEN
```

L1400: Daily total gross assimilation is calculated only after crop emergence.

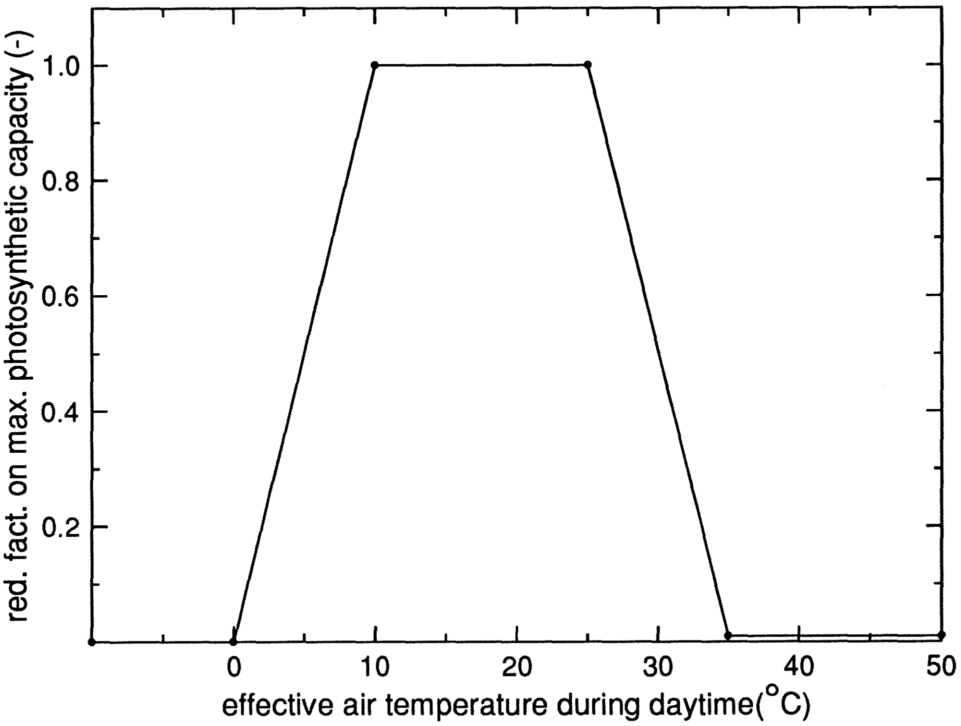

Figure 10 The function TMPFT, the reduction factor on gross CO_2 assimilation as a function of effective air temperature during daytime.

```
AMAXN = AMAX1(0.,(725.*EFRNV-2.75)*LINT(TMPFT,ITMPFT,EAVT))
```

L1410: The variable AMAXN, light saturated CO_2 assimilation rate, is calculated as a linear function of the effective nitrogen concentration for assimilation, EFRNV, times a reduction factor which is calculated by linear interpolation in the function TMPFT (Fig. 10) which relates maximum photosynthetic capacity to effective air temperature during daytime, EAVT. The value of AMAXN is restrained to zero with an AMAX1 function.

```
AMAX = INSW(AMAXN,PAMAX,AMAXN)
```

L1420: The variable AMAX, maximum rate of gross CO_2 assimilation of a single leaf assumes the value of PAMAX, its value one time interval ago, if AMAXN (L1410) is negative, else AMAX assumes the value of AMAXN.

```
PAMAX = AMAX
```

L1430: The variable PAMAX stores the value of AMAX, maximum rate of gross CO_2 assimilation of a single leaf.

```
      CALL TOTASS (SC, DAYL, SINLD, COSLD, DSINBE, RAD,
     $             SCV, AMAX, EFFE, KDIF, GRAI, DTGA)
```

L1440: The subroutine TOTASS is called to calculate daily total gross assimilation DTGA from solar constant, SC, astronomical daylength, DAYL, seasonal offset of sine of solar height, SINLD, amplitude of sine of solar height, COSLD, daily total of effective solar height, DSINBE, daily total global radiation, RAD, scattering coefficient for diffuse light, SCV, maximum rate of gross CO_2 assimilation of a single leaf, AMAX, initial light use efficiency, EFFE, extinction coefficient for diffuse light, KDIF and green area index, GRAI.

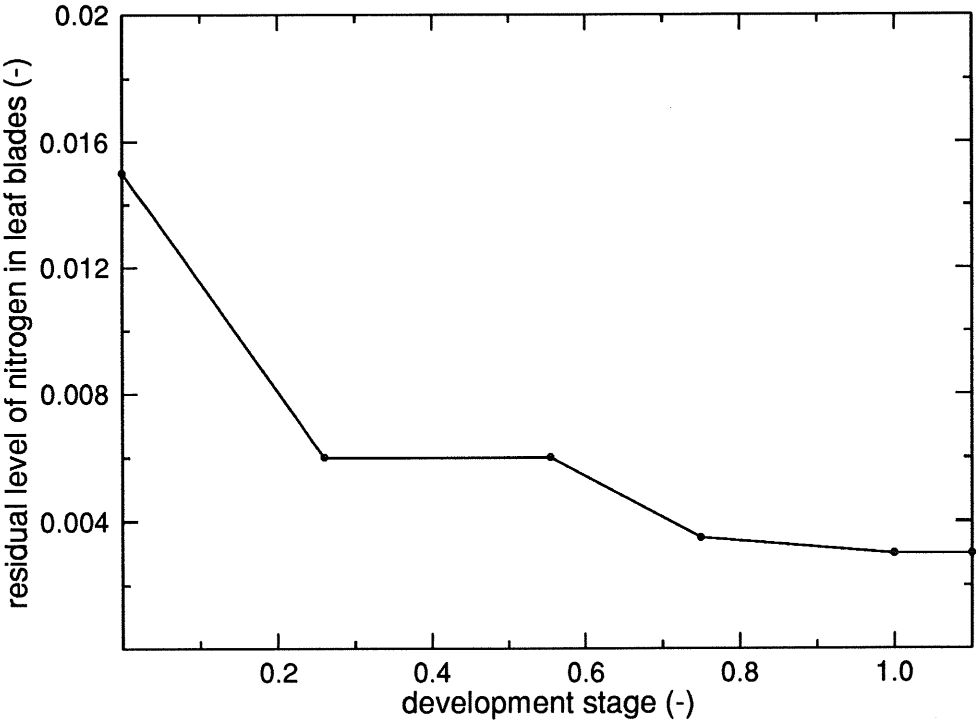

Figure 11 The function BNT, residual level of nitrogen in leaf blades as function of development stage.

```
BN = LINT(BNT,IBNT,DVS)
```

L1450: The variable BN, residual concentration of nitrogen in leaf blades is calculated by linear interpolation in the function BNT (Fig. 11) with the development stage, DVS, as independent variable.

```
SONCT = AMAX1(0.,1.-DVS/((1.+DVSEGF)/2.))
```

L1460: The variable SONCT, intermediate variable used to calculate maximum nitrogen concentration in plant organs as function of DVS, development stage and DVSEGF, the development stage in the post-anthesis phase at end of grain fill. The variable SONCT, is restrained to zero with an AMAX1 function.

```
FNMX = FNMXR + SONCT*FNMXA
```

L1470: The variable FNMX, maximum concentration of nitrogen in leaf blade tissue is calculated as the sum FNMXR, the maximum nitrogen concentration in mature leaf blades and the product of the slope of the maximum nitrogen concentration in tissue versus development stage and FNMXA, the range in maximum nitrogen concentration between young and mature leaf blades.

```
FNC = LIMIT(0.,1.,(FRNV-BN)/(FNMX-BN))
```

L1480: The variable FNC, relative nitrogen concentration in leaf blades, is calculated as the difference between the concentration of nitrogen in leaf blades, FRNV (L1380) and BN, the residual fraction in leaf blades, divided by the difference between the maximum concentration of nitrogen in leaf blades, FNMX (L1470) and the residual concentration of nitrogen in leaf blades, BN. FNC, is restrained between zero and one with a LIMIT function.

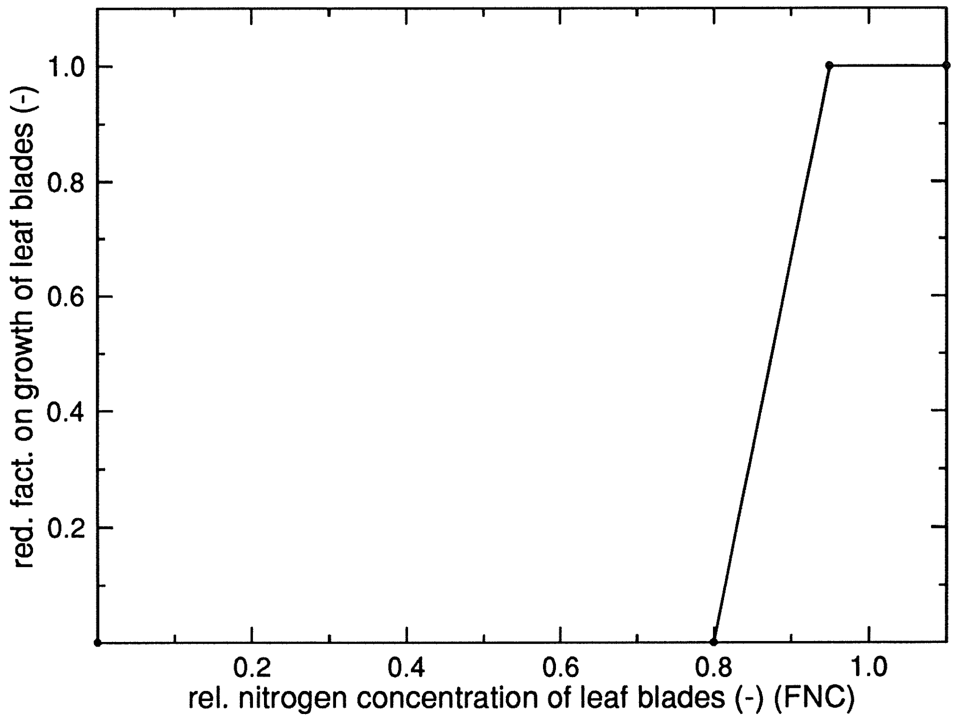

Figure 12 The function RFNST, reduction factor on growth of leaf blades as function of the relative nitrogen concentration in leaf blades.

```
RFNS = LINT(RFNST,IRFNST,FNC)
```

L1490: The variable RFNS, reduction factor on growth of leaf blades due to nitrogen stress is calculated by linear interpolation in the function RFNST (Fig. 12) with FNC (L1480), relative nitrogen concentration in leaf blades, as independent variable.

```
RTURB = 0.
```

L1500: The variable RTURB, turbulent resistance for water vapour exchange is set to zero.

```
IF (GRAI.GT.1.E-10) THEN
LMIX = SQRT(4.*0.02*CROPHT/(PI*GRAI))
```

L1510: Calculations on potential transpiration listed from label 1510 to 1570 are only performed when GRAI, the green area index exceeds zero, to avoid zero divisions. The variable LMIX, mixing length is calculated from CROPHT, crop height, PI, the circumference of a circle divided by its diameter and GRAI, green area index.

```
ALPH = SQRT(DRAGC*GRAI*CROPHT/(2.*LMIX*IW))
```

L1520: The variable ALPH, an intermediate variable in the calculation of turbulent resistance above the crop is calculated from DRAGC, the drag coefficient of the leaf blades, GRAI, green area index, CROPHT (L1350), crop height, LMIX (L1510), mixing length and IW, the turbulent intensity.

```
D = CROPHT-SQRT(LMIX*IW*CROPHT/ALPH)/KARMAN
```

L1530: The variable D, zero plane displacement is calculated from CROPHT, crop height, LMIX, mixing length, IW, turbulent intensity, ALPH (L1520), intermediate variable and KARMAN, von Karmann's constant.

```
ZNOT = (CROPHT-D)*EXP(-CROPHT/(ALPH*(CROPHT-D)))
```

L1540: The variable ZNOT, zero plane displacement, is calculated from CROPHT, crop height and the intermediate variables of turbulent resistance above the crop D and ALPH.

```
LNREF = ALOG((REFHT-D)/ZNOT)
```

L1550: The variable LNREF, intermediate variable in the calculation of turbulent resistance above the crop, is calculated from REFHT, the reference height for measuring wind speed and the intermediate variables D (L1530) and ZNOT (L1540).

```
REST = 0.74*LNREF*LNREF/(KARMAN*KARMAN*1.1574E-7*WSA)
```

L1560: The variable REST, intermediate variable in the calculation of turbulent resistance above the crop, is calculated from LNREF (L1550), intermediate variable, KARMAN, von Karmann's constant and WSA (L1230), wind speed.

```
RTURB = REST/864.E4
```

L1570: The variable RTURB, turbulent resistance for water vapour exchange, is calculated from REST (L1560), the intermediate variable.

```
RA = 4.41E-3*SQRT(1./(WSA+REANOT(WSA)))+RTURB*GRAI
```

L1580: The variable RA, total resistance for water vapour transport from the canopy to the atmosphere, is calculated as the sum of the laminar resistance which is a function WSA, wind speed and the product of the turbulent resistance, RTURB and GRAI, the green area index.

```
SC = SCM*RFNS
```

L1590: The variable SC, actual stomatal conductance as influenced by the nitrogen status of the canopy, is calculated as the product of SCM, the maximum stomatal conductance for water vapour exchange and RFNS (L1490), a dimensionless factor which accounts for the effect of nitrogen shortage on stomatal conductance.

```
RS = 1./(SC+REANOT(SC))
```

L1600: The variable RS, actual stomatal resistance for water vapour exchange, is calculated as the reciprocal of SC, the actual stomatal conductance.

```
S = (RA+RS)/RA
```

L1610: The variable S, an intermediate variable in the calculation of potential crop transpiration, is calculated by dividing the sum of RA (L1580) total resistance of the canopy for water vapour transport and RS (L1600), stomatal resistance, by RA.

```
CC = 1./(SLOPE+S*PSCH)
```

L1620: The variable CC, an intermediate variable in the calculation of potential crop transpiration is calculated as the reciprocal of the sum of the variable SLOPE (L1310) and the product of S (L1610), an intermediate variable and PSCH, the psychrometric constant.

```
      APTRAN = CC*((1.-EXP(-EXC*GRAI))*HNOT*SLOPE+ALPHA*GRAI*RHOCP
   $          /RA*(SVPAM-VPAM)*DAYL/24.)/LHVAP
```

L1630: The variable APTRAN, potential crop transpiration is calculated from the intermediate variable CC (L1620), extinction coefficient for global radiation, EXC, green area index, GRAI, net global radiation, HNOT, slope of the saturated vapour pressure curve at effective daytime temperature, SLOPE (L1310), proportionality factor for calculation of contribution of drying power of the air to crop transpiration, ALPHA (L1330), volumetric heat capacity of the air, RHOCP, total resistance for water vapour transport from canopy to the atmosphere, RA (L1580), vapour pressure deficit during daytime, variable SVPAM minus VPAM, astronomical daylength, DAYL and LHVAP, the latent heat of vaporization.

```
S1 = (RA+RC)/RA
```

L1640: The variable S1, intermediate variable in calculation of cuticular water loss, is calculated as the sum of RA, the resistance of water vapour transport from the canopy to the atmosphere and RC, the cuticular resistance, divided by RA.

```
CC1 = 1./(SLOPE+S1*PSCH)
```

L1650: The variable CC1, intermediate variable in calculation of cuticular water loss, is calculated as the reciprocal of the sum of the variable SLOPE and the product of the intermediate variable S1 (L1640) and PSCH, the psychrometric constant.

```
PCTRAN = APTRAN*CC1/CC
```

L1660: The variable PCTRAN, potential cuticular transpiration under the assumption of fully closed stomata, is calculated as the product of APTRAN (L1630) and the ratio of the intermediate variables CC1 (L1650) and CC (L1620).

```
WSR = WSCONV*WIND
```

L1690: The variable WSR, wind run in km d^{-1} is calculated as the product of the factor WSCONV and WIND, daily wind speed in m s^{-1}, supplied by the subroutine WEATHR in the main program.

```
HZERO = RAD*(1.-REFCF)-LWR
```

L1700: The variable HZERO, net absorbed radiation by an open water surface, is calculated from RAD, global radiation, REFCF, reflection coefficient of open water for shortwave radiation and LWR, net outgoing long wave radiation.

```
EA = 0.35*(SVPA-VPA)*(0.5+WSR/1.6/100.)*LHVAP
```

L1710: The variable EA, drying power term in the Penman equation is calculated from the vapour pressure deficit, SVPA minus VPA, WSR, wind run and LHVAP, the latent heat of vaporization.

```
DELTA = 17.4*SVPA*(1.-TMPA/(TMPA+239.))/(TMPA+239.)
```

L1720: The variable DELTA, slope of the saturated vapour pressure curve at air temperature is calculated as the first derivative of the analytical expression.

```
EVAP = (HZERO*DELTA/GAMMA+EA)/(1.+DELTA/GAMMA)*1./LHVAP
```

L1730: The variable EVAP, potential rate of soil surface evaporation is calculated from the net absorbed radiation of an open water surface, HZERO (L1700), variable DELTA (L1720), psychrometric constant GAMMA, drying power in the Penman equation, EA (L1710) and LHVAP, the latent heat of vaporization.

```
PEVAP1 = EVAP*EXP(-5.E-5*ARLFE)
PEVAP = INSW(-RAIN,INSW(PEVAP1-RAIN,PEVAP1,PEVAP1-RAIN),PEVAP1)
```

L1740: The variable PEVAP1, intermediate variable in the calculation of the potential rate of soil evaporation is calculated from EVAP (L1730) taking into account the shading effect of ARLFE, total area of green and senesced leaf blades. The variable PEVAP, the potential rate of soil evaporation as limited by canopy cover gets the value of PEVAP1 if daily rainfall, RAIN, is zero or exceeds the value of PEVAP1, else if the daily rainfall is less than PEVAP1, PEVAP equals the difference between PEVAP1 and RAIN.

```
IRRID = LINT(IRRT,IIRRT,DATE)
```

L1745: The variable IRRID, current irrigation rate is calculated by linear interpolation in the array IRRT with day number, DATE, as independent variable.

```
INFR = (1.-RUNOFC)*RAIN*INSW(PEVAP-PEVAP1,0.,1.) + IRRID
```

L1750: The variable INFR, infiltration rate is calculated from RUNOFC, the run-off coefficient, RAIN, daily rainfall, and IRRID (L1745), the irrigation rate.

```
        RWF(1) = INFR
        DO 150 I = 2,ILAYER+1
         RWF(I)=AMAX1(0.,RWF(I-1)-(FLDCP(I-1)*THCKN(I-1)-WATER(I-1))/DELT)
         IF (TDT(I).GE.MXRTD.AND.TDT(I-1).LT.MXRTD) K = I
150     CONTINUE
```

L1760: The array RWF(I), rate of flow of soil moisture through the upper boundary of soil compartment I is calculated from RWF(I-1), flow through the upper boundary of the overlying layer, FLDCP(I), field capacity of compartment I, THCKN(I), thickness of that compartment and its actual soil moisture content, WATER(I) and the timestep, DELT. The soil compartment reached at maximum rooting depth is stored in variable K.

```
RDRAIN = RWF(K)
```

L1770: The variable RDRAIN, rate of drainage of soil moisture beyond the potential rooting zone is assigned the value of RWF(K), the rate of flow of soil moisture into the soil compartment marked by the variable K (L1760).

```
WCPR = (WATER(1)/THCKN(1)-WCLIM)/(FLDCP(1)-WCLIM)
```

L1780: The variable WCPR, dimensionless soil moisture content in the top compartment is calculated from the soil moisture content in the first soil layer WATER(1), thickness of the first compartment, THCKN(1), volumetric soil moisture content at air dryness, WCLIM and FLDCP(1), field capacity of the first soil layer.

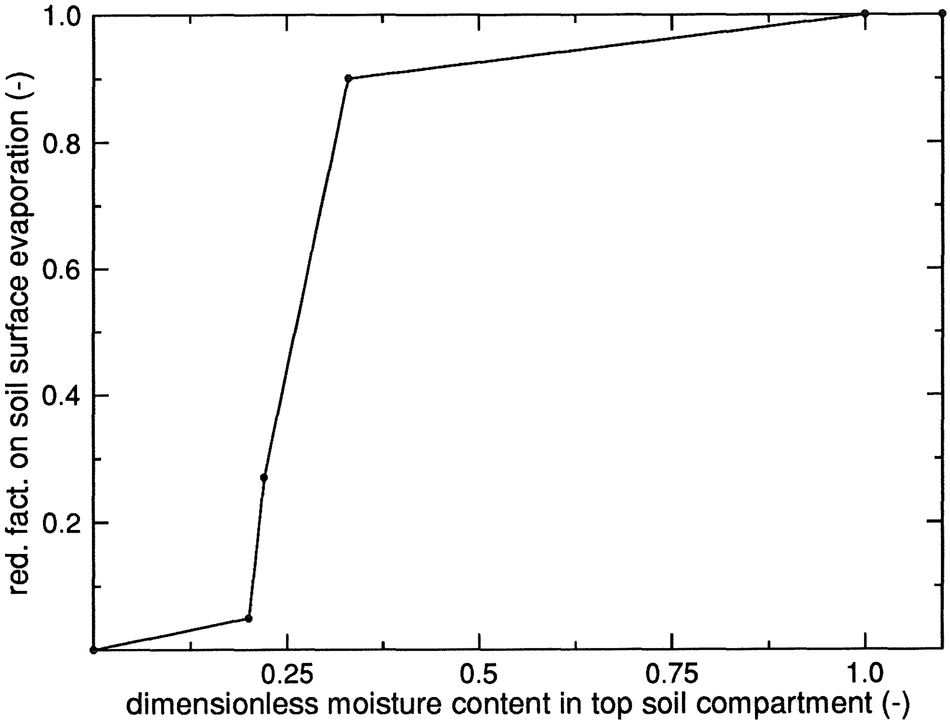

Figure 13 **The function REDFDT, reduction factor on soil surface evaporation as a function of dimensionless moisture content in top soil compartment.**

```
AEVAP = AMAX1(0.,PEVAP*LINT(REDFDT,IRDFDT,WCPR))
```

L1790: The variable AEVAP, current rate of soil evaporation as a function of canopy cover and dryness of top soil compartment is calculated as the product of PEVAP, the potential soil evaporation as a function of soil cover and the reduction factor REDFDT (Fig. 13) which is a function of WCPR (L1780).

```
MWRTD = 0.
```

L1800: The variable MWRTD, maximum amount of soil moisture that can be stored in rooted depth, is initialized at zero.

```
DO 160 I = 1,ILAYER
AWATER(I) = AMAX1(0.,WATER(I)-THCKN(I)*WLTPT(I))
```

L1810: The array AWATER, amount of available soil moisture in soil compartment I is calculated from WATER, actual soil moisture content in mm, THCKN, thickness of soil compartment and WLTPT, the volumetric soil moisture content at wilting point in that compartment.

```
RTL(I) = LIMIT(0.,THCKN(I),RTD-TDT(I))*INSW(TDT(I)-MXRTD,1.,0.)
```

L1820: The array RTL, root length in soil compartment I is calculated from THCKN, thickness of the compartment, RTD, current rooting depth, TDT, the depth of the upper boundary of soil compartment I below soil surface and MXRTD, the maximum rooting depth. The variable RTL is limited to zero with a LIMIT function.

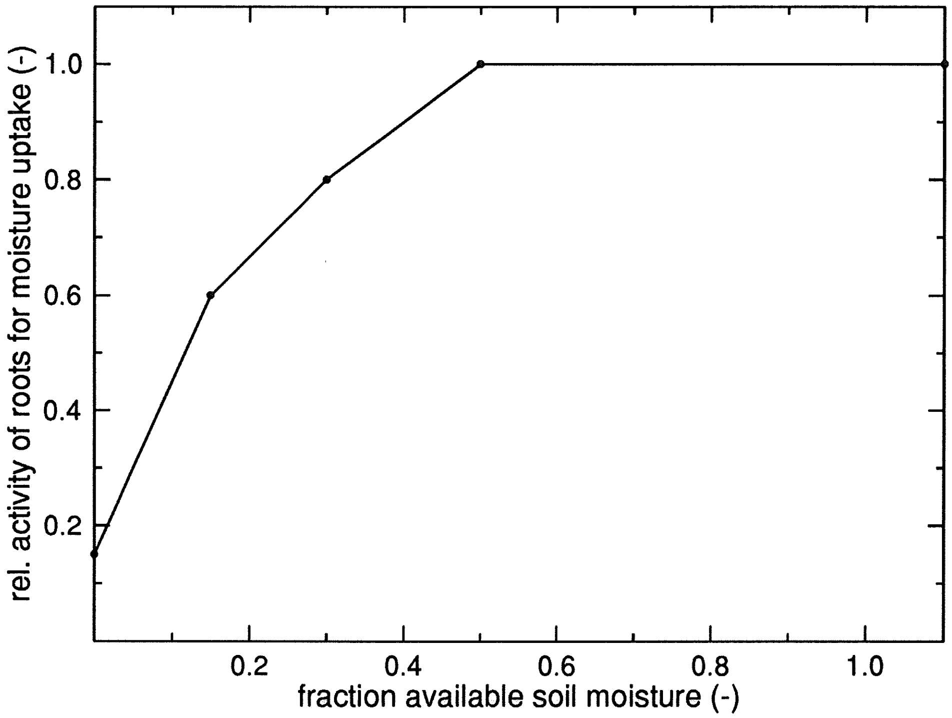

Figure 14 The function EDPTFT, reduction factor on soil moisture uptake due to low available moisture as a function of fraction available moisture per soil compartment.

```
EDPTF(I) = LINT(EDPTFT,IEDPTN,AWATER(I)/(THCKN(I)*STORC(I)))
```

L1830: The array EDPTF, reduction factor on soil moisture uptake due to low available moisture in soil compartment I is calculated by linear interpolation in the function EDPTFT (Fig. 14), with the ratio available soil moisture in layer I and storage capacity of layer I (fraction available soil moisture) as independent variable.

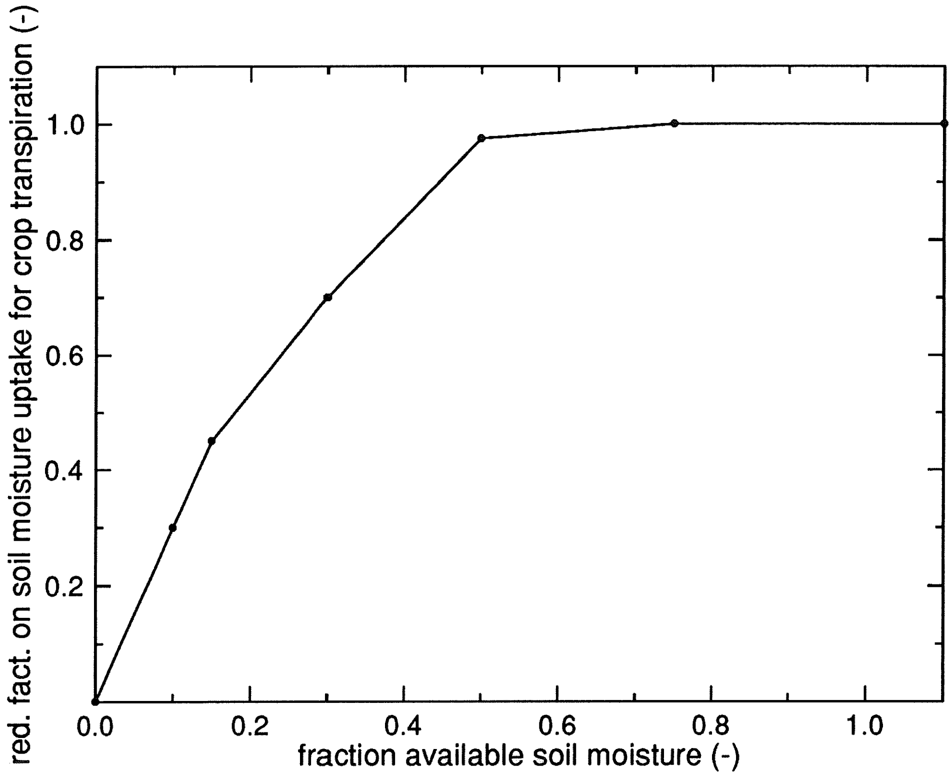

Figure 15 The function WREDT, reduction factor on soil moisture uptake for crop transpiration per soil compartment as function of fraction available soil moisture.

```
      WRED(I) = LINT(WREDT,IWREDT,AWATER(I)/(THCKN(I)*STORC(I)))
```

L1840: The array WRED, reduction factor on moisture uptake for crop transpiration in layer I due to low soil moisture is calculated by linear interpolation in the function WREDT (Fig. 15) with the ratio available soil moisture in layer I and storage capacity of layer I as independent variable.

```
      AWATF(I) = INSW(-AWATER(I),1.,0.)
```

L1850: The array AWATF(I), indicating the presence or absence of plant available soil moisture in soil compartment I is calculated with an INSW function from the amount of available soil moisture in soil compartment I, AWATER(I) (L1810).

```
      MWRTD = MWRTD+RTL(I)*STORC(I)
160   CONTINUE
```

L1860: The variable MWRTD, maximum amount of soil moisture that can be stored in rooted depth is calculated by accumulating the products of root length in layer I, RTL(I) and STORC(I), the storage capacity in layer I.

```
      ERLT = 0.
```

L1870: The variable ERLT, total effective root length, is initialized at zero.

```
        RWRBT = 0.
        DO 170 I = 1,ILAYER
          IF (TDT(I)+0.5.GT.MXRTD) GO TO 170
          ERLT = ERLT+RTL(I)*EDPTF(I)
          RWRBT = RWRBT+RTL(I)/THCKN(I)*AWATER(I)/(MWRTD+REANOT(MWRTD))
170     CONTINUE
```

L1880: The variable RWRBT, relative amount of available soil moisture in the root zone, is initialized at zero. For each layer I whose upper boundary does not exceed MXRTD, the maximum rooting depth, the variables ERLT, effective length of the roots, and RWRBT, the contribution to the relative amount of available soil moisture in the root zone are accumulated. The effective root length in compartment I is calculated as the product of root length, RTL(I) and EDPTFT(I) (L1830).

```
        TRPMM = APTRAN/(ERLT+REANOT(ERLT))
```

L1890: The variable TRPMM, potential rate of moisture uptake per unit effective root length is calculated by dividing potential crop transpiration, APTRAN, by the total effective root length.

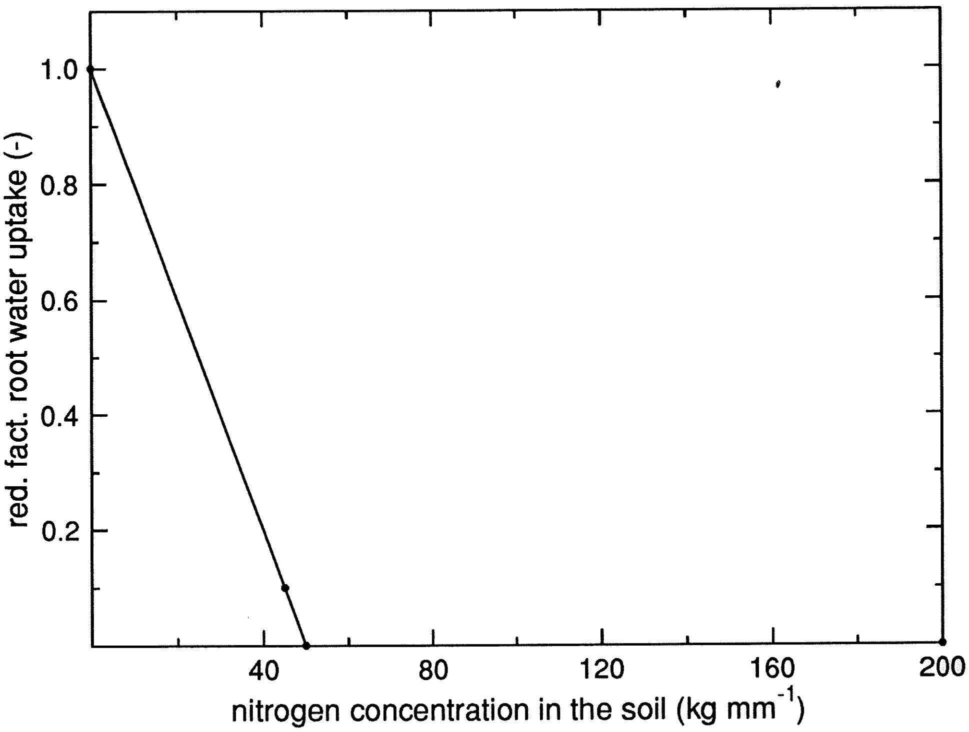

Figure 16 The function ROSPT, reduction factor on root moisture uptake as a function of nitrogen concentration per soil compartment.

```
         TRANW = 0.
         DO 180 I = 1,ILAYER
           TRR(I) =TRPMM*RTL(I)*EDPTF(I)*WRED(I)*LINT(ROSPT,IROSPT,CONC(I))
           TRANW = TRANW+TRR(I)
180      CONTINUE
```

L1900: The variable TRANW, actual rate of canopy transpiration, is initialized at zero. The array TRR, the actual rate of soil moisture uptake for transpiration per soil compartment is calculated as the product of TRPMM, the potential rate of soil moisture uptake per unit effective root length, RTL(I), root length in compartment I, EDPTF(I) (L1830), reduction factor on soil moisture uptake due to low available moisture in soil compartment I, WRED(I), reduction factor on effective root length as a function of soil moisture content in compartment I and ROSPT, the reduction factor on soil moisture uptake as a function of nitrogen concentration in soil compartment I (Fig. 16). The variable TRANW, the actual rate of canopy transpiration is calculated by accumulating TRR(I), the actual rates of moisture uptake for transpiration over the soil compartments.

```
         SUMT = 0.
         DO 190 I = 1,ILAYER
           VAR(I) = AMAX1(WATER(I)/THCKN(I)-WCLIM,0.)*EXP(-PROP*0.001*
     $              (TDT(I)+0.5*THCKN(I)))
           SUMT = SUMT+VAR(I)*THCKN(I)
190      CONTINUE
```

L1910: The variable SUMT, intermediate variable in the calculation of soil surface evaporation, is initialized at zero. The array VAR, auxiliary variable in the calculation of soil surface evaporation is calculated from WATER(I), the amount of soil moisture in soil compartment I, THCKN(I), thickness of the soil compartment, WCLIM, volumetric soil moisture content at air dryness, PROP, proportionality factor for calculation of soil moisture contribution to soil surface evaporation by soil compartment and TDT(I), the depth of the upper boundary below soil surface of compartment I. The variable SUMT is calculated as the sum of the products of VAR(I) and THCKN(I) over the layers.

```
         SWPBT = 0.
         DO 200 I = 1,ILAYER
           ER(I) = AMIN1(AMAX1(WATER(I)-WCLIM*THCKN(I),0.),AEVAP*THCKN(I)*
     $          VAR(I) /(SUMT+REANOT(SUMT)))
           SWPBT = SWPBT+AWATF(I)*REAAND(RTD-TDT(I),TDT(I+1)-RTD)
           EVTOT = EVTOT+ER(I)
200      CONTINUE
```

L1920: The variable SWPBT, a switch variable which indicates whether the root tip is in a wet soil compartment (1) or not (0) is set to zero. The array ER, rate of moisture extraction for soil surface evaporation from soil compartment I is calculated as the minimum of the actual amount of soil moisture in layer I, WATER(I), minus WCLIM, soil moisture content at air dryness times THCKN(I), thickness of the soil compartment, or AEVAP, the potential rate of soil evaporation, times THCKN(I) times the contribution from layer I to soil surface evaporation, the ratio VAR(I) over SUMT. The variable SWPBT is calculated as the product of AWATF, a factor that indicates presence or absence of available moisture in soil compartment I times the outcome of the logical expression REAAND which assumes the value 1 if the total layer can be explored by the roots. The variable EVTOT, current rate of soil surface evaporation is calculated as the sum of ER(I), the rate of soil moisture extraction from soil compartment I for soil surface evaporation.

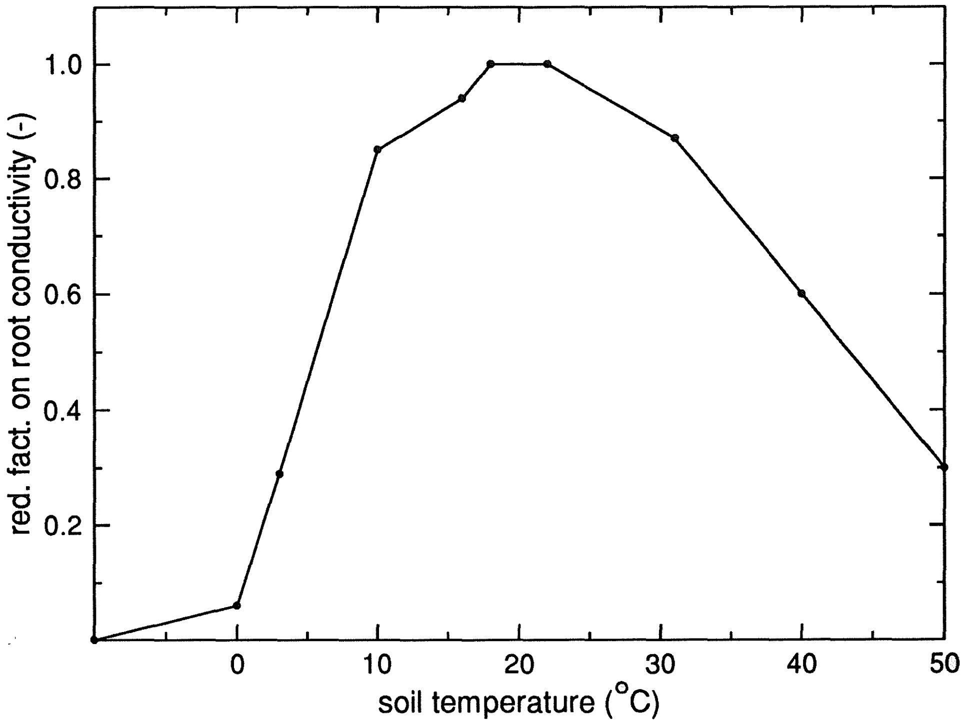

Figure 17 The function TECT, reduction factor on root conductivity as function of soil temperature.

```
TEC = LINT(TECT,ITECT,TS)
```

L1950: The variable TEC, which accounts for the effect of soil temperature on root conductivity, is calculated by linear interpolation in the function TECT (Fig. 17) with TS, the soil temperature, as independent variable.

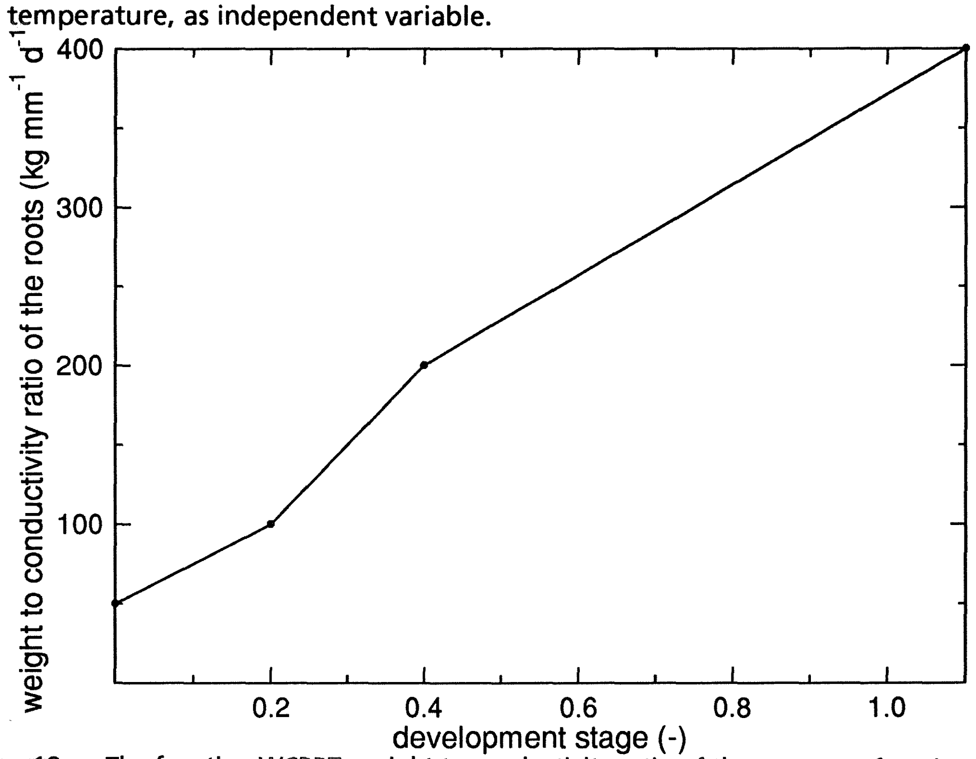

Figure 18 The function WCRRT, weight to conductivity ratio of the roots as a function of development stage.

```
        ACOND = WRT/(LINT(WCRRT,IWCRRT,DVS))*TEC
```

L1960: The variable ACOND, actual conductivity of the root system, is calculated from the dry weight of the roots, WRT, divided by the weight to conductivity ratio obtained by linear interpolation in the function WCRRT (Fig. 18) with DVS, the development stage, as independent variable multiplied by TEC (L1950).

```
        TRAN = AMIN1(TRANW,ACOND)
```

L1970: The variable TRAN, the current rate of canopy transpiration is calculated as the minimum of TRANW, the soil moisture available for canopy transpiration and ACOND, the conductivity of the root system.

```
        DO 210 I = 1,ILAYER
          DWAT(I) = RWF(I)-RWF(I+1)-TRR(I)*TRAN/(TRANW+REANOT(TRANW)) - ER(I)
210     CONTINUE
```

L1980: The array DWAT, rate of change in amount of soil moisture per soil compartment, is calculated from RWF(I), the flow of soil moisture into the soil compartment, RWF(I+1), flow of soil moisture out of the compartment, TRR(I), potential rate of soil moisture uptake for transpiration multiplied by the ratio TRAN (L1970) over TRANW (L1900) and ER(I), the rate of soil moisture extraction for soil surface evaporation.

```
        RTRDEF = (APTRAN-TRAN)/(APTRAN+REANOT(APTRAN))
```

L1990: The variable RTRDEF, relative transpiration deficit, is calculated as the difference between APTRAN (L1630), the potential transpiration and TRAN, the actual transpiration as a fraction of APTRAN.

```
        RESL = ARESP/(ARESP+WLVS+WRT+WSTEM+REANOT(ARESP+WLVS+WRT+WSTEM)
```

L2000: The variable RESL, concentration of non-structural carbohydrates (reserves) in the crop, is calculated as ARESP, dry weight of the reserve carbohydrates divided by the sum of ARESP, WLVS, dry weight of the leaf blades, WRT, dry weight of the roots, and WSTEM, dry weight of the stems.

```
        REDFRL = 1.-LIMIT(0.,1.,(RESL-TLRGA)/0.05)
```

L2010: The variable REDFRL, a factor accounting for the effect of carbohydrate accumulation on gross canopy assimilation, is calculated as one minus the difference between RESL (L2000) and TLRGA, the threshold concentration of reserves, beyond which assimilation is affected, which was read from file, divided by 0.05. The value of REDFRL is restrained between zero and one with a LIMIT function.

```
        DGAS = DTGA*CCO2TS*TRAN/(APTRAN+REANOT(APTRAN))*REDFRL
```

L2020: The variable DGAS, daily gross carbohydrate production is calculated as the product of DTGA, daily gross CO_2 assimilation, CCO2TS, conversion factor from CO_2 to reduced sugars, the ratio TRAN, actual canopy transpiration over APTRAN, potential canopy transpiration and REDFRL (L2010).

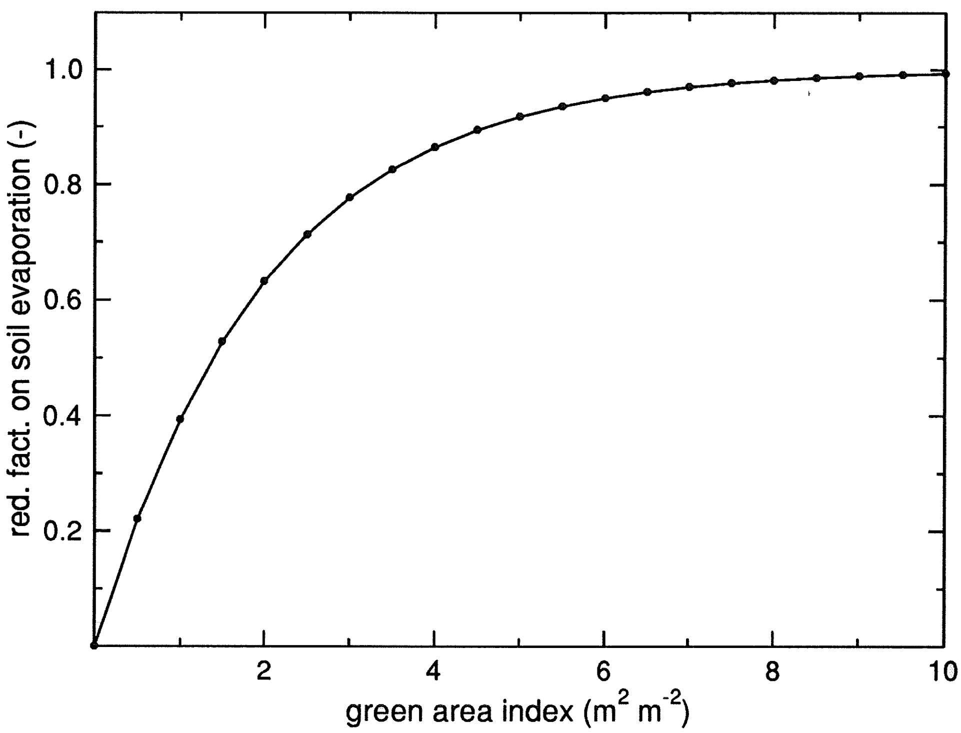

Figure 19 Reduction factor on soil evaporation as function of the green area index.

```
EVAPR = EVAP*(1.-EXP(-0.5*GRAI))
```

L2030: The variable EVAPR, rate of soil surface evaporation from wet soil surface in the presence of a crop, is calculated from EVAP (L1730), potential rate of soil surface evaporation, times a reduction factor which is calculated with a negative exponential equation with GRAI, green area index in the exponent (Fig. 19).

```
TMPC = TMPA+CTEMPF*(1.-FTMPA*TMPA*TRAN/(EVAPR+REANOT(EVAPR)))
```

L2040: The variable TMPC, average daily canopy temperature, is calculated from TMPA, average daily air temperature, CTEMPF, the maximum effect of evaporative cooling on canopy temperature, FTMPA, reciprocal of mean standard canopy temperature (1/15) and the ratio TRAN (L1970) over EVAPR (L2030).

```
APFERT = 0.
IF (NNA.LE.IAPDAY.AND.TIME.GE.NAPDAY(NNA)) THEN
  APFERT = NTRMNT(NNA)
  NNA = NNA+1
END IF
```

L2050: The variable APFERT, the fertilizer application rate, is initialized at zero. If NNA, the actual number of nitrogen applications that has been applied is less than or equals the maximum number IAPDAY which was read from file, and the current time is greater than or equals the date of the subsequent application NAPDAY(NNA), the variable APFERT assumes the value of NTRMNT(NNA), where NNA represents the number of the actual nitrogen application.

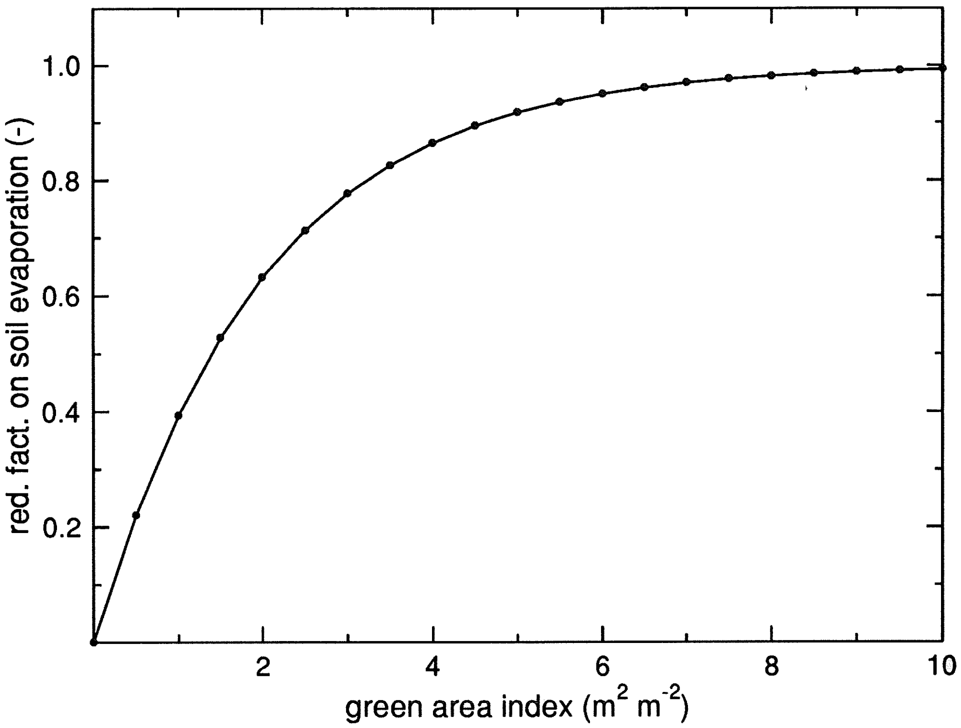

Figure 19 Reduction factor on soil evaporation as function of the green area index.

```
EVAPR = EVAP*(1.-EXP(-0.5*GRAI))
```

L2030: The variable EVAPR, rate of soil surface evaporation from wet soil surface in the presence of a crop, is calculated from EVAP (L1730), potential rate of soil surface evaporation, times a reduction factor which is calculated with a negative exponential equation with GRAI, green area index in the exponent (Fig. 19).

```
TMPC = TMPA+CTEMPF*(1.-FTMPA*TMPA*TRAN/(EVAPR+REANOT(EVAPR)))
```

L2040: The variable TMPC, average daily canopy temperature, is calculated from TMPA, average daily air temperature, CTEMPF, the maximum effect of evaporative cooling on canopy temperature, FTMPA, reciprocal of mean standard canopy temperature (1/15) and the ratio TRAN (L1970) over EVAPR (L2030).

```
APFERT = 0.
IF (NNA.LE.IAPDAY.AND.TIME.GE.NAPDAY(NNA)) THEN
  APFERT = NTRMNT(NNA)
  NNA = NNA+1
END IF
```

L2050: The variable APFERT, the fertilizer application rate, is initialized at zero. If NNA, the actual number of nitrogen applications that has been applied is less than or equals the maximum number IAPDAY which was read from file, and the current time is greater than or equals the date of the subsequent application NAPDAY(NNA), the variable APFERT assumes the value of NTRMNT(NNA), where NNA represents the number of the actual nitrogen application.

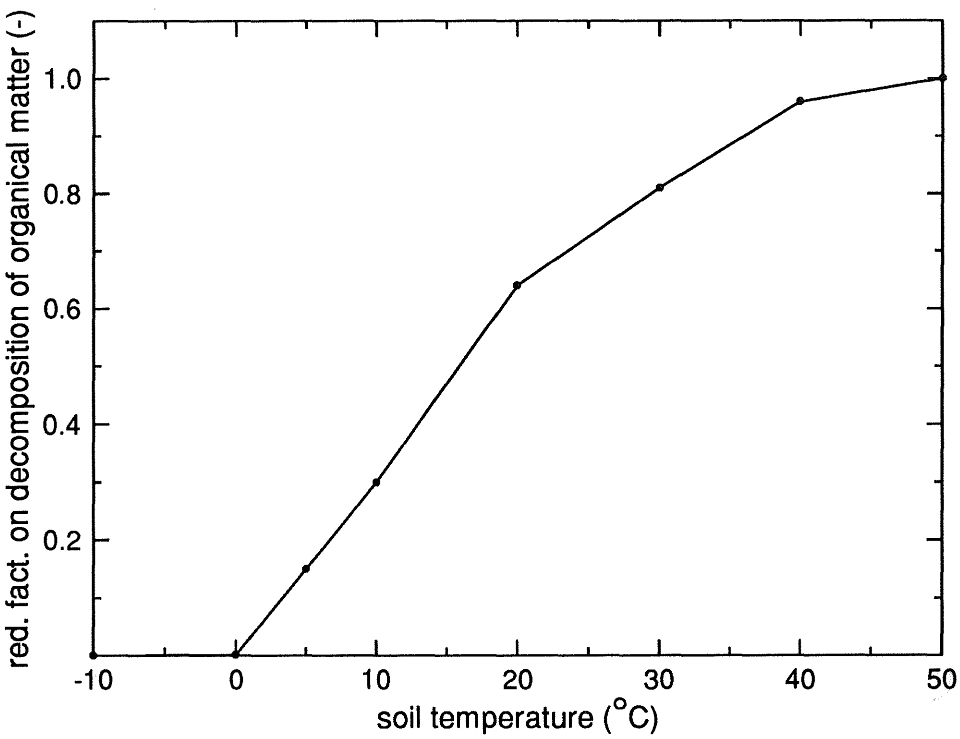

Figure 20 The function TFT, reduction factor on decomposition of organic matter as function of soil temperature.

```
TF = LINT(TFT,ITFT,TS)
```

L2060: The variable TF, a factor accounting for the effect of soil temperature on decomposition of organic material, is calculated by linear interpolation in the function TFT (Fig. 20) with TS, soil temperature as independent variable.

```
YG = 0.25*EFCPR + 0.75*EFCCH
```

L2070: The variable YG, growth efficiency of microbial biomass, which is equivalent to the conversion efficiency of primary assimilates into grain dry matter, is calculated from EFCPR, conversion efficiency of primary assimilates into structural proteins and EFCCH, conversion efficiency of primary assimilates into structural carbohydrates.

```
DO 220 I = 1,ILAYER
CNR(I) = FRC*FOM(I)/(FON(I)+ASLT(I)+REANOT(FON(I)+ASLT(I)))
```

L2090: The array CNR, carbon to nitrogen ratio of fresh organic material per soil compartment is calculated as the product of FRC, fraction of carbon in fresh organic material, and FOM(I), weight of fresh organic material in soil compartment I, divided by the sum of FON(I), weight of nitrogen in fresh organic material in soil compartment I and ASLT(I), weight of mineral nitrogen in soil compartment I.

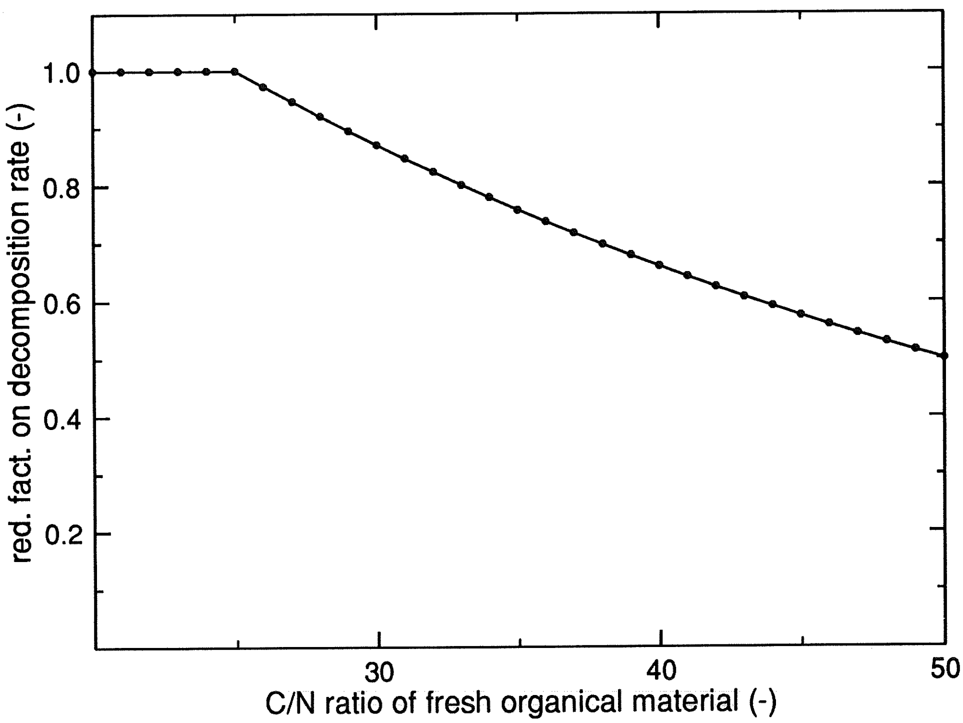

Figure 21 Relationship between the carbon to nitrogen ratio of fresh organic material in a soil compartment and its relative rate of decomposition.

```
CNRF(I) = AMIN1(1.,EXP(-.693*(CNR(I)-25.)/25.))
```

L2100: The factor CNRF(I), effect of the carbon to nitrogen ratio of fresh organic material, on the relative rate of decomposition in soil compartment I is calculated from CNR(I) (L2090) (Fig. 21).

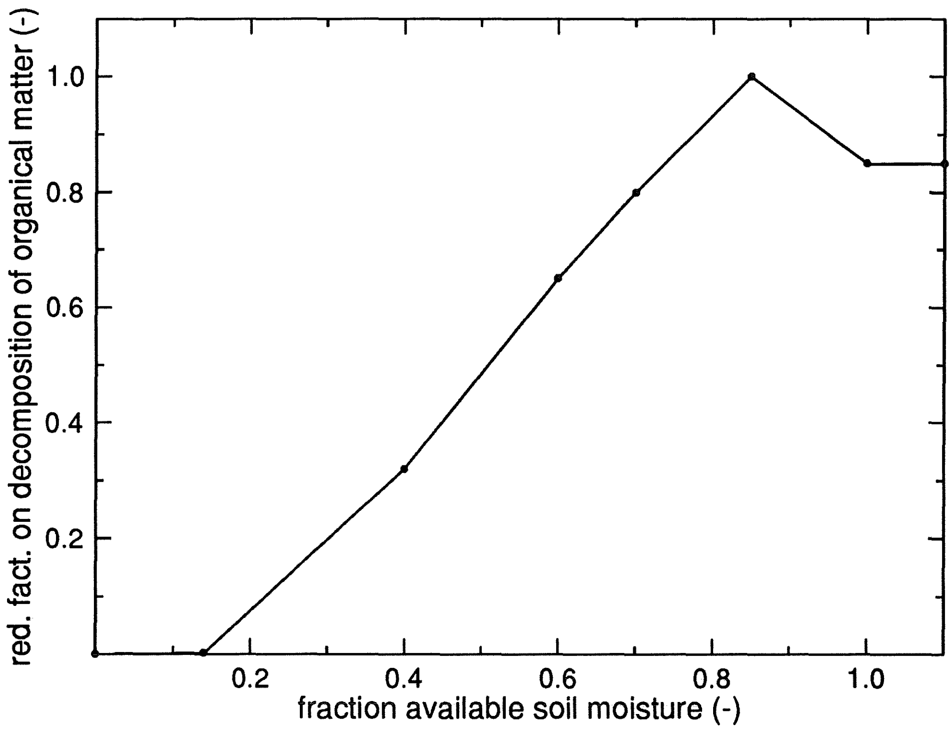

Figure 22 Function MFT, reduction factor on decomposition of organic matter as function of fraction available soil moisture.

```
MF(I) = LINT(MFT,IMFT,AWATER(I)/(STORC(I)*THCKN(I)))
```

L2110: The array MF, a factor accounting for the effect of available soil moisture on rate of decomposition of organic material per soil compartment is calculated by linear interpolation in the function MFT (Fig. 22) with the fractional available soil moisture as independent variable.

```
FOMRES(I) = FOM(I)/(IFOM(I)+REANOT(IFOM(I)))
```

L2120: The array FOMRES, fraction of fresh organic material not yet decomposed per soil compartment is calculated by dividing FOM(I), the weight of fresh organic material in soil compartment I by IFOM(I), the initial weight of fresh organic material in soil compartment I.

```
      RDECR(I) = INSW(FOMRES(I)-FLIGN,RDLIGN,
$              INSW(FOMRES(I)-(FLIGN+FCELL),RDCELL,RDCAPR))
```

L2130: The array RDECR, rate of decomposition of fresh organic material per soil compartment, is calculated from FOMRES(I) (L2120), FLIGN, fraction of lignin in initial fresh organic material, RDLIGN, relative rate of decomposition of lignin under optimum conditions, FCELL, fraction of cellulose and hemicellulose in original fresh organic material, RDCELL, relative rate of decomposition of cellulose and hemicellulose under optimum conditions and RDCAPR, relative rate of decomposition of easily decomposable carbohydrates and proteins under optimum conditions.

```
        LDEC = BIOMN(I)*CNRMIC/FRC*(MRGRB/YG+RRMIC)
```

L2140: The variable LDEC, maximum rate of decomposition of fresh organic material, as dictated by maximum growth rate of microbial biomass in soil compartment I is calculated from BIOMN(I), the weight of nitrogen in microbial biomass in soil compartment I, CNRMIC, carbon to nitrogen ratio of microbial biomass, FRC, fraction of carbon in fresh material, MRGRB, maximum relative growth rate of microbial biomass, YG (L2070), growth efficiency of microbial biomass and RRMIC, relative rate of maintenance respiration of microbial biomass.

```
        DECR(I) = AMIN1(LDEC,FOM(I)*RDECR(I)*TF*CNRF(I)*MF(I))
```

L2150: The array DECR, rate of decomposition of fresh organic material per soil compartment, is calculated as the product of FOM(I), weight of fresh organic material in soil compartment I, RDECR(I) (L2130) and the reduction factors TF (L2060), CNRF(I) (L2100) and MF(I) (L2110). The variable DECR(I) is restrained to a maximum, the value of the variable LDEC (L2140), with an AMIN1 function.

```
        RNRL(I) = DECR(I)*FON(I)/(FOM(I)+REANOT(FOM(I)))
```

L2160: The array RNRL, rate of nitrogen mineralization from decomposing fresh organic material, is calculated from the product of DECR(I) (L2150) and FON(I), the weight of nitrogen in fresh organic material in soil compartment I, divided by FOM(I), the weight of fresh organic material in soil compartment I.

```
        RHMIN(I) = NHUM(I)*DMINR*TF*MF(I)
```

L2170: The array RHMIN, rate of nitrogen mineralization from stable organic material per soil compartment, is calculated as the product of NHUM(I), the weight of nitrogen in stable organic material and DMINR, the relative rate of decomposition of stable organic material under optimum conditions, and the reduction factors TF (L2060) and MF(I) (L2110).

```
220     CONTINUE
        DO 230 I = 1,ILAYER
        CADEC(I) = 0.4*DECR(I)+10.*RHMIN(I)
```

L2180: The array CADEC, rate of carbon release through decomposition of organic material per soil compartment, is calculated as the sum of 0.4 times the value of DECR(I) (L2150) and 10 times the value of RHMIN(I) (L2170).

```
        CFBMG(I) = INSW(DBIOMN(I),DBIOMN(I)*CNRMIC,DBIOMN(I)*CNRMIC/YG)
```

L2200: The array CFBMG, rate of release or immobilization of carbon due to changes in microbial biomass per soil compartment, is calculated as the product of DBIOMN(I), the rate of change in nitrogen in microbial tissue in compartment I and CNRMIC, the carbon to nitrogen ratio of microbial biomass, if DBIOMN(I) is positive or zero (release). If DBIOMN(I) is negative (immobilization) the same product as previously mentioned is divided by YG, the efficiency of conversion of primary assimilates into microbial biomass.

```
        BIOMXC(I) = AMAX1(CADEC(I)-CFBMG(I),0.5*CADEC(I))/RRMIC
```

L2210: The array BIOMXC, maximum weight of carbon in microbial biomass, limited by carbon availability is calculated from the difference between CADEC(I) (L2180), rate of carbon release through decomposing organic material in soil compartment I and CFBMG(I), the rate of release or immobilization of carbon in soil compartment I due to changes in microbial biomass divided by RRMIC. This difference is restrained to half the value of CADEC(I) divided by RRMIC, the relative rate of maintenance respiration of microbial biomass.

```
      BIOMXN(I) = (ASLT(I)+BIOMN(I))*CNRMIC
```
L2220: The array BIOMXN, maximum weight of carbon in microbial biomass limited by nitrogen availability per soil compartment is calculated as the sum of ASLT(I), the weight of mineral nitrogen in soil compartment I and BIOMNI(I), the weight of nitrogen in microbial biomass in soil compartment I times CNRMIC, the carbon to nitrogen ratio of microbial biomass.

```
      BIOMX(I) = AMIN1(BIOMXC(I),BIOMXN(I),BIOMN(I)*CNRMIC*(1.+MRGRB))
```
L2230: The array BIOMX, maximum weight of carbon in microbial biomass per soil compartment, limited by either nitrogen or carbon availability is calculated as the minimum of the variables, BIOMXC(I) (L2210), BIOMXN(I) (L2220) and the product of BIOMN(I), weight of nitrogen in microbial biomass in soil compartment I and CNRMIC, the carbon to nitrogen ratio of microbial biomass times one plus MRGRB, the maximum relative growth rate of microbial biomass.

```
      DBN(I) = (BIOMX(I)/CNRMIC-BIOMN(I))/TCMG
```
L2240: The array DBN, 'uncorrected' rate of change in nitrogen in microbial tissue per soil compartment, is calculated from the variables BIOMX(I) (L2230), CNRMIC (L2230), BIOMN(I), weight of nitrogen in microbial biomass in soil compartment I and TCMG, the time constant for microbial growth.

```
      DBIOMN(I) = INSW(DBN(I),DBN(I)*DELT*RRMIC,DBN(I))
```
L2250: The array DBIOMN, 'effective' rate of change in nitrogen in microbial tissue per soil compartment, is calculated from the variables DBN(I) (L2240), time step of integration DELT and RRMIC, the relative rate of maintenance respiration of microbial biomass.

```
      RNRLB(I) = -1.*AMIN1(0.,DBIOMN(I))
```
L2260: The array RNRLB, rate of nitrogen mineralization from dying microorganisms per soil compartment, is calculated from the variable DBIOMN(I) (L2250). If array DBIOMN(I) is negative, RNRLB assumes the same value but with opposite sign.

```
      RNAC(I) = AMAX1(0.,DBIOMN(I))
230   CONTINUE
```
L2270: The array RNAC, rate of nitrogen immobilization by soil microbes per soil compartment, is assigned the maximum of zero and the value of DBIOMN(I) (L2250).

```
      SGFF = INSW(DVSR-DVSSGF,0.,1.)
```
L2280: The factor SGFF, which indicates whether development stage is beyond start of grain fill (1) or not (0) is calculated from the difference between DVSR, the development stage in the post-anthesis phase minus DVSSGF, the development stage in the post-anthesis phase at start of grain fill.

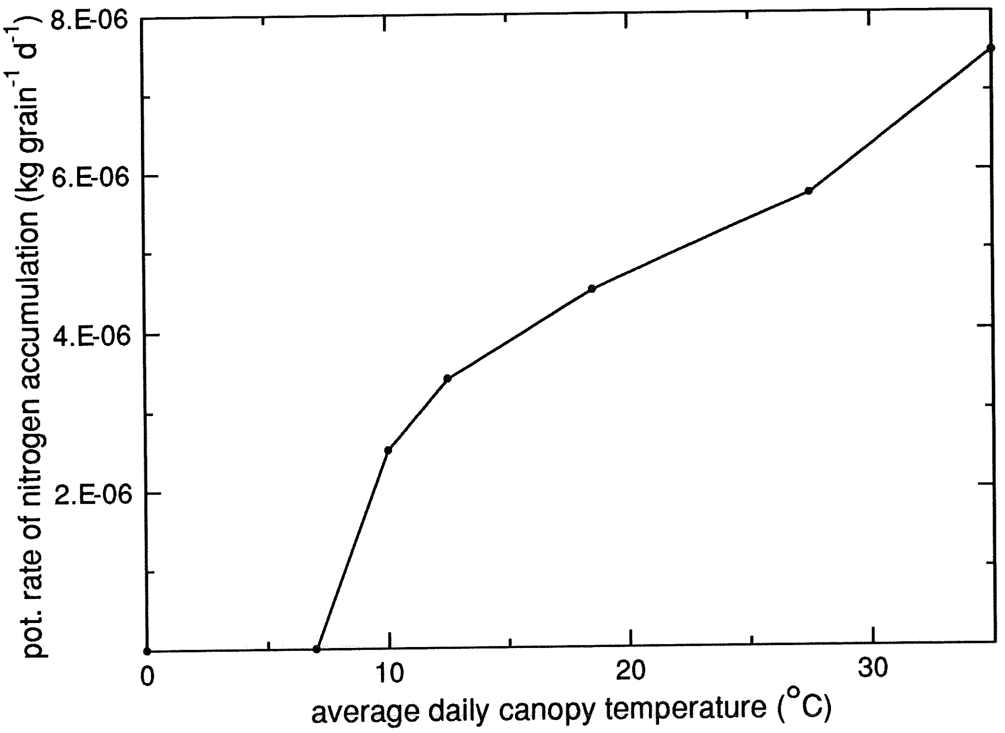

Figure 23 Function PRNAGT, potential rate of nitrogen accumulation in the grain as function of canopy temperature.

```
PRNAGR = GRN*LINT(PRNAGT,IPRAGT,TMPC)*SGFF
```

L2290: The variable PRNAGR, potential rate of nitrogen accumulation in the grain, is calculated as the product of GRN, grain density, the potential rate of nitrogen accumulation per grain which is calculated by linear interpolation in the function PRNAGT (Fig. 23) with TMPC, the canopy temperature as independent variable and SGFF (L2280).

```
AVNLV = AMAX1(0.,ANLV-WLVS*BN)
```

L2300: The variable AVNLV, weight of nitrogen in leaf blades available for translocation to the grain, is calculated as ANLV, total weight of nitrogen in the leaf blades minus the product of WLVS, dry weight of the leaf blades and BN, the residual non-remobilizable concentration of nitrogen in leaf blades. The value of AVNLV is restrained to a minimum of zero with an AMAX1 function.

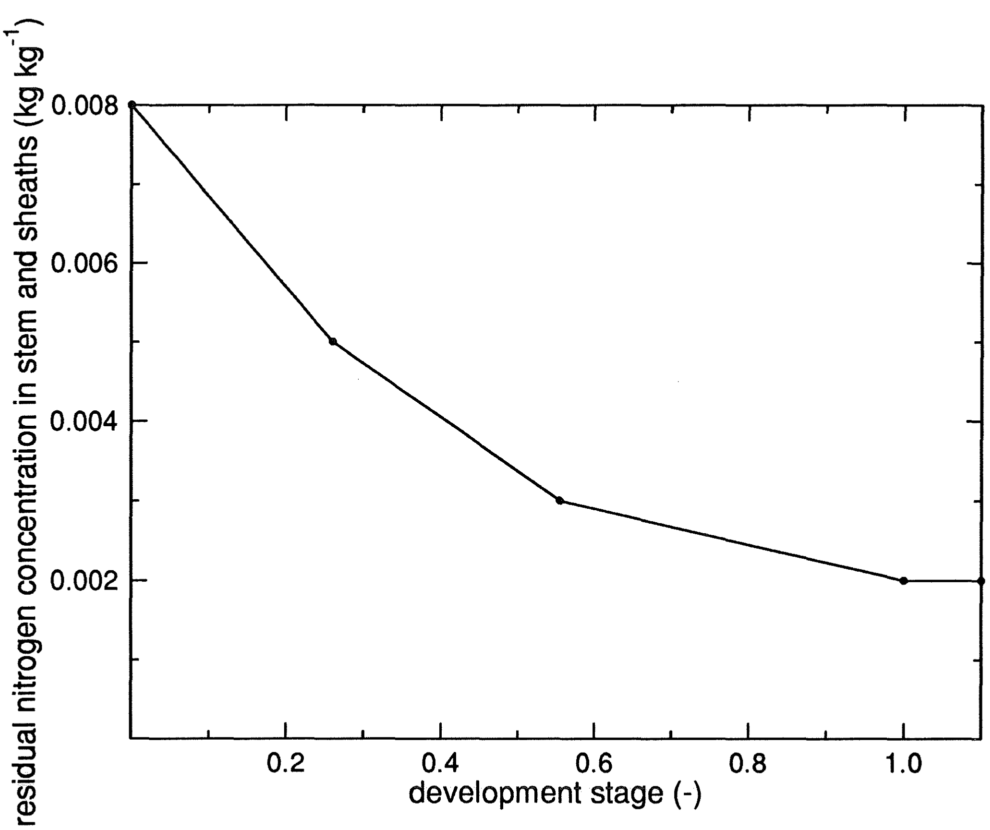

Figure 24 Function FNT, residual non-remobilizable concentration of nitrogen in stem and sheaths as function of development stage.

```
LN = LINT(LNT,ILNT,DVS)
```

L2310: The variable LN, residual non-remobilizable concentration of nitrogen in stem and sheaths is calculated by linear interpolation in the function LNT (Fig. 24) with the development stage, DVS, as independent variable.

```
AVNSTE = AMAX1(0.,ANSTE-WSTEM*LN)
```

L2320: The variable AVNSTE, weight of nitrogen in stem and sheaths available for translocation to the grain, is calculated as ANSTE, total weight of nitrogen in stem and sheaths, minus the product of WSTEM, dry weight of live stem and sheaths and LN (L2310), the residual non-remobilizable nitrogen concentration in stem and sheaths. The value of AVNSTE is restrained to a minimum of zero with an AMAX1 function.

```
AVNRT = AMAX1(0.,ANRT-WRT*RN)
```

L2330: The variable AVNRT, weight of nitrogen in roots available for translocation to the grains is calculated as ANRT, total weight of nitrogen in roots minus the product of WRT, dry weight of the roots, and RN, the residual non-remobilizable concentration of nitrogen in roots. The value of AVNRT is restrained to a minimum of zero with an AMAX1 function.

```
AVN = AVNLV+AVNSTE+AVNRT
```

L2340: The variable AVN, total weight of nitrogen in vegetative tissue available for translocation to the grain, is calculated as the sum of AVNLV (L2300), AVNSTE (L2320) and AVNRT (L2330).

```
FNVEG = (ANRT+ANSTE+ANLV)/(WRT+WSTEM+WLVS+REANOT(WLVS+WRT+WSTEM))
```

L2350: The variable FNVEG, average concentration of nitrogen in vegetative tissue, is calculated as the ratio the sum of the weight of nitrogen in roots, stem, leaf sheaths and leaf blades, variables ANRT, ANSTE and ANLV, respectively and the sum of the dry weights of roots, stem and leaf sheaths and leaf blades, variables WRT, WSTEM and WLVS, respectively.

```
FRNG = ANGR/(WGR+REANOT(WGR))
```

L2360: The variable FRNG, concentration of nitrogen in the grain is calculated by dividing ANGR, weight of nitrogen in the grain by WGR, dry weight of the grain.

```
DVSRP = DVSRX
DVSRX = DVSR
```

L2370: The variable DVSRX, development stage one time interval ago is assigned to the variable DVSRP, which is used in a procedure to initialize nitrogen translocation to the grain. After this assignment the current development stage in the post-anthesis phase, DVSR, is assigned to DVSRX.

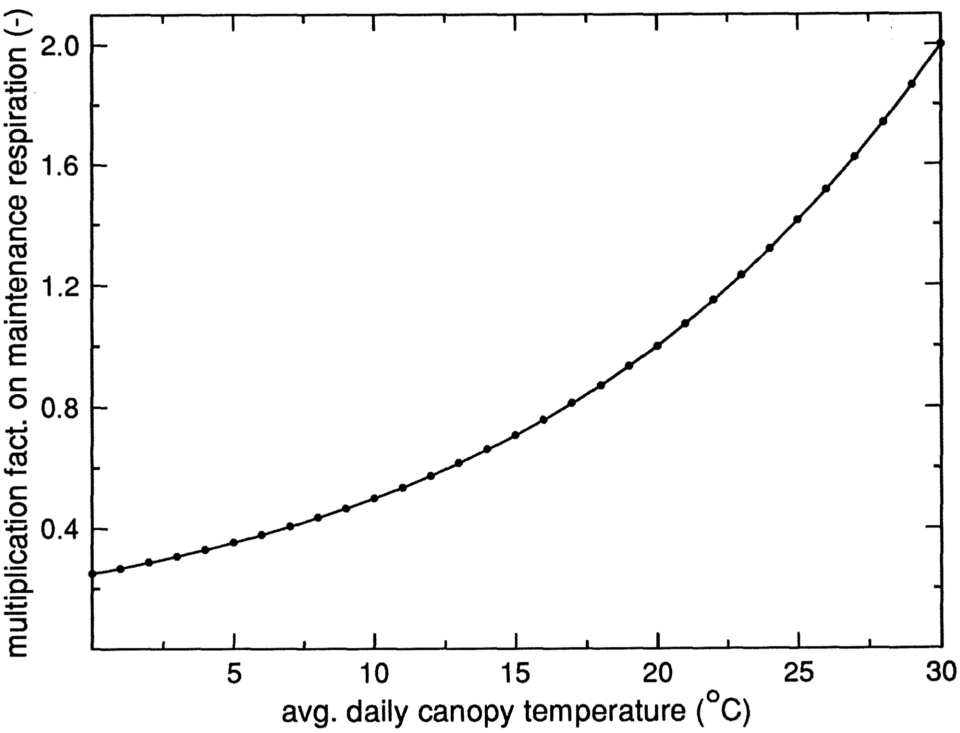

Figure 25 Factor TEF, multiplication factor on maintenance respiration as function of average daily canopy temperature.

```
TEF = Q10**(0.1*TMPC-2.0)
```

L2380: The factor TEF, effect of canopy temperature on maintenance respiration is calculated as the variable Q10, the increase in respiration intensity per 10 °C increase in temperature to the power of 0.1 times TMPC , the average daily canopy temperature minus 2.0 (Fig. 25).

```
EFFWS = INSW(-TRAN,INSW(PCTRAN-TRAN,1.,2.),0.)
```

L2390: The variable EFFWS, effect of soil moisture stress on turnover of nitrogen in vegetative tissue is calculated from the variables TRAN (L1970) and PCTRAN (L1660). If actual transpiration is positive, EFFWS equals one when actual transpiration exceeds potential cuticular transpiration and two when cuticular transpiration exceeds soil moisture uptake.

```
RTORT = RRTORT*TEF*EFFWS*INSW(-ARESP,1.,1.2)
```

L2400: The variable RTORT, rate of nitrogen turnover in vegetative tissue, is calculated as the product of RRTORT, basic relative rate of nitrogen turnover in vegetative biomass, TEF (L2380) and EFFWS (L2390), times 1.2, if the dry weight of reserve carbohydrates, ARESP, equals zero or times 1, if reserve carbohydrates are available.

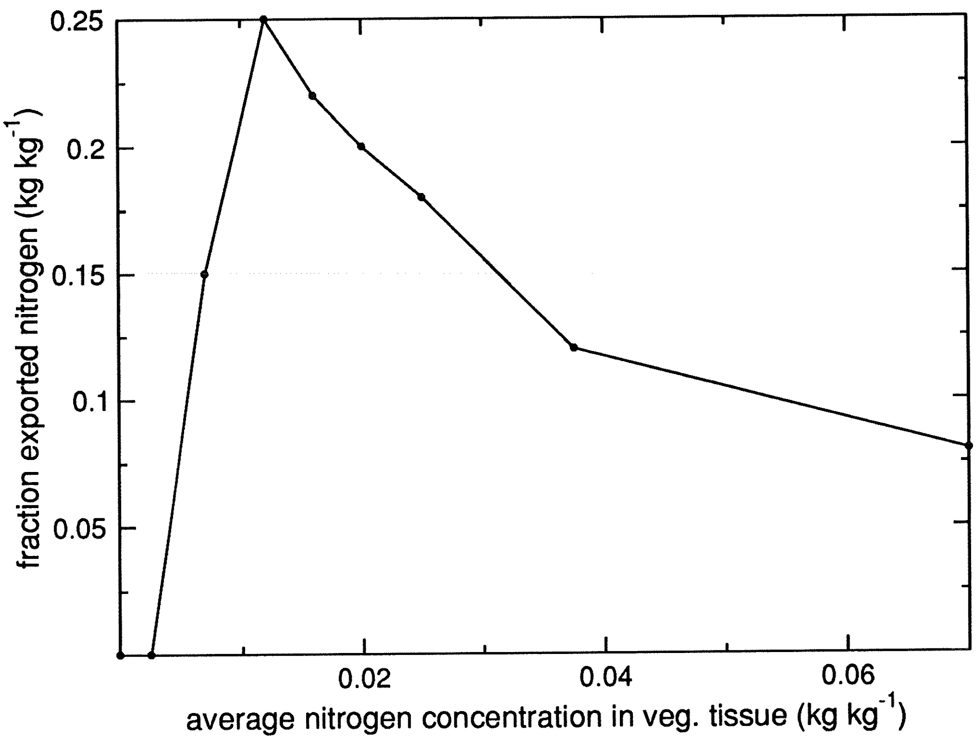

Figure 26 Function FNEXT, fraction of labile nitrogen exported from vegetative tissue to the grain as function of the average nitrogen concentration in the vegetative tissue.

```
RNEXP = AMAX1(0.,RTORT*AVN*LINT(FNEXT,IFNEXT,FNVEG))
```

L2410: The variable RNEXP, rate of nitrogen export from vegetative tissue to the grain, is calculated as the product of RTORT (L2400), AVN (L2340) and the fraction of labile nitrogen exported from vegetative tissue to the grain, calculated by linear interpolation in the function FNEXT (Fig. 26) with FNVEG (L2350), the average nitrogen concentration in vegetative tissue as independent variable.

```
PUSHN = REAAND(DVSR-DVSSGF,DVSSGF-DVSRP)
```

L2420: The switch variable PUSHN is calculated using the REAAND function with DVSR, the current development stage in the post-anthesis phase, DVSSGF, the development stage in the post-anthesis phase at the start of grain filling and DVSRP, development stage in the post-anthesis phase one time interval ago, as arguments.

```
RNTG = INSW(FRNG-0.035,AMIN1(PRNAGR,RNEXP),0.)*(1.-PUSHN)
```

L2430: The variable RNTG, rate of nitrogen accumulation in grain, is calculated with an INSW function with FRNG, the concentration of nitrogen in the grain as argument. Nitrogen translocation to grain stops when FRNG reaches the value of 0.035. When actual values of FRNG are below that level, RNTG assumes the minimum value of either PRNAGR, the potential or RNEXP, the actual rate of nitrogen translocation.

```
RNTLS = RNTG*AVNLV/(AVN+REANOT(AVN))
```

L2440: The variable RNTLS, rate of transfer of nitrogen from the leaf blades to the grain, is calculated as the product of RNTG (L2430) and the ratio of AVNLV (L2300) and AVN (L2340).

```
SFPRD = 0.5*RNTG*6.25
```

L2450: The variable SFPRD, energy contribution from catabolyzed proteins, available for maintenance respiration is calculated as 0.5 times RNTG (L2430) times 6.25, the conversion factor from nitrogen to protein content.

```
RMNLVS = WLVS*RMRESL*((FRNV-FNMIN)/RNFAC+1.)*TEF-
$          SFPRD*WLVS/(TVEGM+REANOT(TVEGM))
```

L2460: The variable RMNLVS, is calculated from WLVS, dry weight of live leaf blades, RMRESL, relative rate of maintenance respiration of leaf blades times the difference between FRNV, the current concentration of nitrogen in leaf blades and FNMIN, the absolute minimum nitrogen concentration in leaf blades divided by RNFAC, the range in nitrogen concentration between maximum and minimum concentration plus 1, times TEF (L2380) minus SFPRD (L2450) times the ratio WLVS to TVEGM, total dry weight of the vegetative crop organs.

```
RMNG = WGR*RMRESG*((FRNG-FNMIN)/RNFAC+1.)*TEF*LIMIT(0.,1.,3.-3.*DVSR)
```

L2470: The variable RMNG, rate of maintenance respiration of the grain is calculated from WGR, the dry weight of the grain, RMRESG, relative maintenance respiration rate of the grain, the difference between FRNG, the concentration of nitrogen in the grain and FNMIN (L2460), divided by RNFAC (L2460), plus 1, to account for the nitrogen content of the grain times TEF (L2380). The results are restrained with a LIMIT function to proportionally decrease RMNG after DVSR exceeds 0.66.

```
FRNR = ANRT/(WRT+REANOT(WRT))
```

L2480: The variable FRNR, concentration of nitrogen in root tissue, is calculated by dividing ANRT, the weight of nitrogen in the roots by WRT, dry weight of the roots.

```
RMNR = WRT*RMRESR*((FRNR-FNMIN)/RNFAC+1.)*TEF-
$          SFPRD*WRT/(TVEGM+REANOT(TVEGM))
```

L2490: The variable RMNR, rate of maintenance respiration of root tissue, is calculated similar to the maintenance respiration of leaf blades (L2460) with WRT, dry weights of the roots, RMRESR, relative maintenance respiration of the roots and FRNR, the concentration of nitrogen in the roots as root-specific variables.

```
FRNST = ANSTE/(WSTEM+REANOT(WSTEM))
```

L2500: The variable FRNST, concentration of nitrogen in stem and sheath tissue, is calculated by dividing ANSTE, total nitrogen in stem and leaf sheaths by WSTEM, the dry weight of live stem and sheaths.

```
      RMNST = WSTEM*RMRESS*((FRNST-FNMIN)/RNFAC+1.)*TEF
$             -   SFPRD*WSTEM/(TVEGM+REANOT(TVEGM))
```

L2510: The variable RMNST, rate of maintenance respiration of stem tissue is calculated similar to maintenance respiration of leaf blades (L2460), with WSTEM, the dry weight of the stems, RMRESS, relative maintenance respiration of the stems and FRNST (L2500), the concentration of nitrogen in the stems as stem-specific variables.

```
      FCHNX = DGAS-RMNLVS-RMNG-RMNR-RMNST
```

L2520: The variable FCHNX, net flow of carbohydrate, i.e. difference between gross assimilation and maintenance respiration requirements is calculated as DGAS, the rate of gross canopy assimilation in CH_2O minus the respiration rates RMNLVS (L2460), of leaf blades, RMNG (L2470), of grain, RMNR (L2520), of roots and RMNST (L2520), of stem and sheaths.

```
      FCHN = AMAX1(0.,FCHNX)
```

L2530: The variable FCHN, net flow of carbohydrates assumes the value of FCHNX (L2520), limited to positive values using the AMAX1 function.

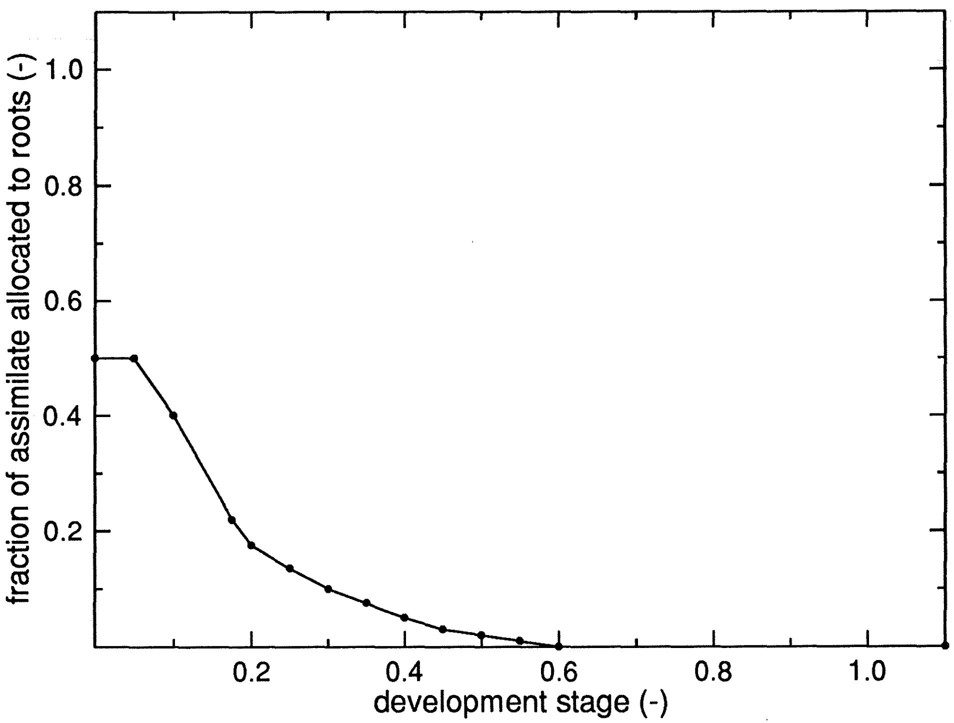

Figure 27 Function FDSRT, fraction of net assimilate allocated to the roots as function of development stage.

```
FDSR = LINT(FDSRT,IFDSRT,DVS)
```

L2540: The variable FDSR, fraction of assimilate allocated to the roots is calculated by linear interpolation in the function FDSRT (Fig. 27) with the development stage, DVS, as independent variable.

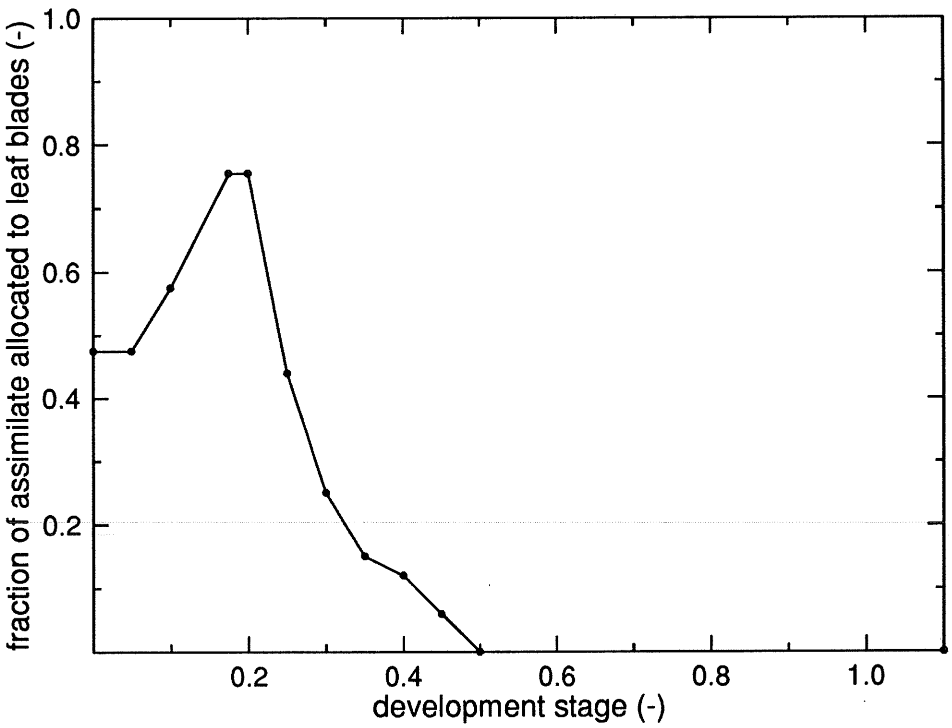

Figure 28 Function FTLVST, fraction of net assimilate allocated to leaf blades as function of development stage.

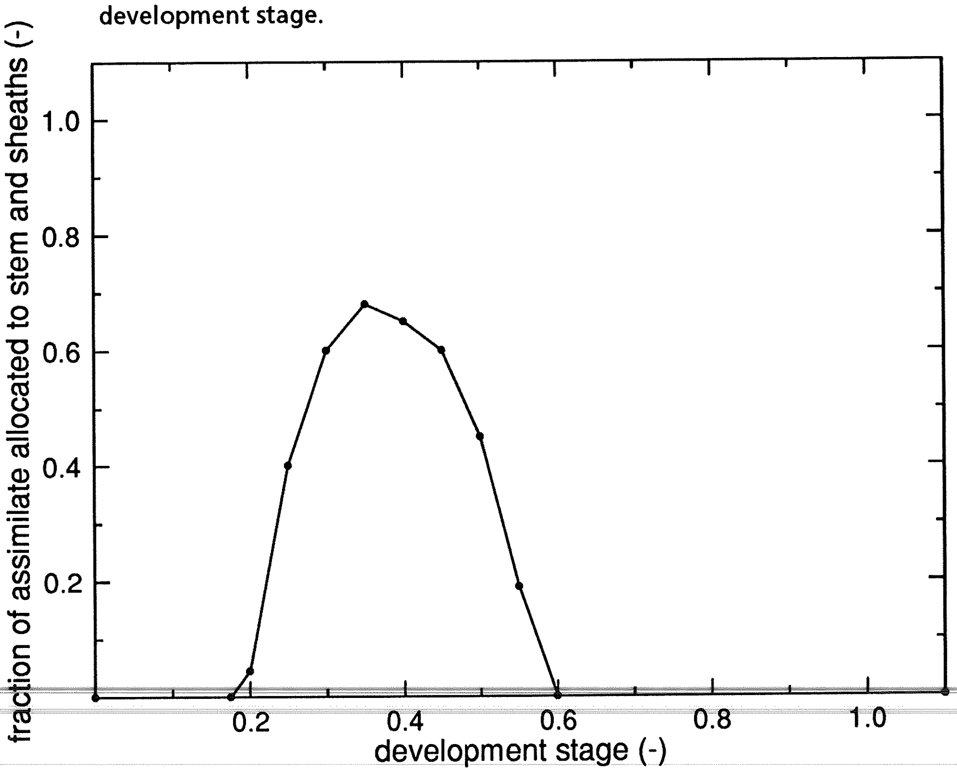

Figure 29 Function FTSTET, fraction of net assimilate allocated to stem and sheaths as function of development stage.

```
FTLVS = LINT(FTLVST,IFTLST,DVS)
```

L2550: The variable FTLVS, fraction of net assimilate allocated to leaf blades, is calculated by linear interpolation in the function FTLVST (Fig. 28) with the development stage as independent variable.

```
FTSTE = LINT(FTSTET,IFSTET,DVS)
```

L2560: The variable FTSTE, fraction of net assimilate allocated to stem and sheaths, is calculated by linear interpolation in the function FTSTET (Fig. 29) with the development stage as independent variable.

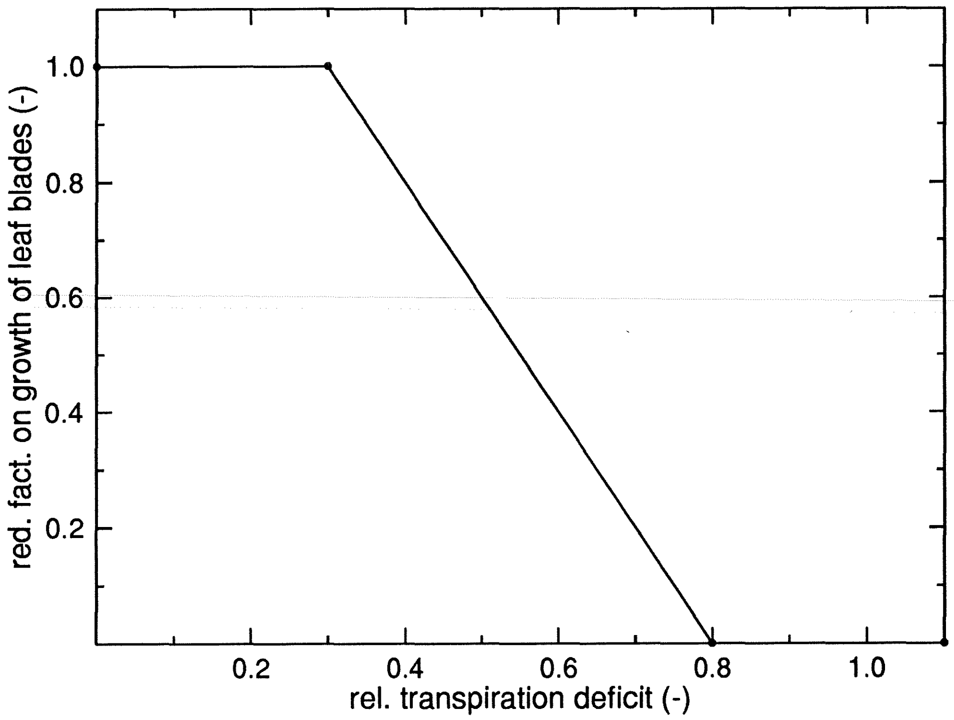

Figure 30 **Function REDWST, reduction factor on leaf blade growth as a function of relative transpiration deficit.**

```
RFWS = LINT(REDWST,IRDWST,RTRDEF)
```

L2570: The variable RFWS, accounting for the effect of soil moisture shortage on growth of leaf blades, is calculated by linear interpolation in the function REDWST (Fig. 30), with the relative transpiration deficit, RTRDEF as independent variable.

```
RFSTRS = AMIN1(RFWS,RFNS)
```

L2580: The variable RFSTRS, accounting for the effect of either soil moisture or nitrogen shortage on leaf growth, is assigned the minimum of the variables RFWS (L2570) or RFNS (L1490), with an AMIN1 function.

```
SCHFLV = (FTLVS+FTSTE)*FCHN*(1.-RFSTRS)
```

L2590: The variable SCHFLV, 'surplus' carbohydrate flow originating from inhibition of growth of leaf blades, is calculated as the sum of the fractions of current assimilate allocated to leaf blades and stem and sheaths, FTLVS and FTSTE, respectively, corrected for the growth check due to water or nitrogen shortage (1.-RFSTRS), times the net flow of carbohydrate FCHN.

```
FCHTR = FDSR*FCHN+SCHFLV*FSCHG
```

L2600: The variable FCHTR, current flow of carbohydrates to roots, is calculated as the product of FDSR (L2540), the fraction of current assimilate allocated to roots and FCHN (L2530), the net flow of carbohydrates plus the product of SCHFLV (L2590) and FSCHG (L130), fraction of 'surplus' carbohydrate allocated to the roots, which was read from file.

```
FPRT = 6.25*ANRT/(WRT+REANOT(WRT))
```

L2610: The variable FPRT, fraction of protein in the roots, is calculated as 6.25 times ANRT, total nitrogen in the roots, divided by WRT, dry weight of the roots.

```
GRRT = FCHTR*(FPRT*EFCPR+(1.-FPRT)*EFCCH)
```

L2620: The variable GRRT, rate of increase in dry weight of the roots is calculated from FCHTR, current flow of carbohydrates to the roots, FPRT (L2610), fraction of protein in the roots, EFCPR, efficiency of conversion of primary assimilates into structural proteins and EFCCH, efficiency of conversion of primary assimilates into structural carbohydrates.

```
OFNRT = FNRTMN+(1.-DVS)*(FNRTMX-FNRTMN)
```

L2630: The variable OFNRT, maximum nitrogen concentration in the roots as function of development stage, is calculated as FNRTMN, maximum nitrogen concentration in mature roots, plus the product of one minus the overall development stage, DVS, and the range in maximum nitrogen concentration between young, FNRTMX, and mature roots, FNRTMN.

```
ARESPF = INSW(-ARESP,1.,0.)
```

L2640: The variable ARESPF, a factor indicating presence (1) or absence (0) of reserve carbohydrates, is calculated with an INSW function from ARESP, the weight of reserve carbohydrates.

```
DNRT = INSW(FRNR-OFNRT,((WRT+GRRT*DELT)*OFNRT-ANRT)/TCU,0.)*ARESPF
```

L2650: The variable DNRT, nitrogen demand of the roots, is set to zero in the absence of reserve carbohydrates, i.e. if variable ARESPF (L2640) equals zero. In the presence of reserves it is calculated as the sum of WRT, dry weight of the roots plus GRRT the rate of increase in dry weight of the root system times DELT, multiplied by OFNRT (L2630), the maximum nitrogen concentration in the roots, minus the weight of nitrogen in the roots, ANRT. The total weight in kg ha^{-1} is divided by TCU, the time constant for uptake of nitrogen to calculate the daily rate.

```
RESLL = INSW(DVSV-DVSFE,RESL1,RESL2)
```

L2670: The variable RESLL, non-remobilizable concentration of reserve carbohydrates in the vegetation for transfer to vegetative structures, assumes the value of RESL1 before the end and the value of RESL2 after end of floret formation.

```
TRFRS = AMAX1(0.,RESL-RESLL)*(WSTEM+WLVS+WRT)/TCTR*
$    INSW(FTLVS-0.10,0.,1.)*INSW(RFSTRS-0.99,0.,1.)
```

L2680: The variable TRFRS, rate of translocation of reserves to vegetative structures is calculated from RESL, the current concentration of non-structural carbohydrates (reserves) in the canopy, RESLL, non-remobilizable concentration of reserves in the vegetation for transfer to vegetative structures, dry weight of the stem and live sheaths, leaf blades and roots, WSTEM, WLVS, WRT respectively and TCTR, the time constant for translocation of reserves. The fraction of current assimilate supply allocated to leaf blades as function of development stage, FTLVS and RFSTRS are used in INSW functions to stop translocation of reserves to vegetative tissue if either the value of FTLVS (L2550) is below 0.1 or RFSTRS (L2580) is below 0.99.

```
FCHTLV = FTLVS*FCHN*RFSTRS+TRFRS
```

L2690: The variable FCHTLV, current flow of carbohydrates to leaf blades, is calculated as the product of FTLVS (L2550), FCHN (L2530) and RFSTRS (L2580) plus TRFRS (L2680), the amount of translocated reserve carbohydrates.

```
EPVC = 6.25*ANLV/(WLVS+REANOT(WLVS))
```

L2700: The variable EPVC, fraction of protein in leaf blade tissue is calculated as 6.25 times ANLV, weight of nitrogen in the leaf blades divided by WLVS, dry weight of the leaf blades.

```
GRLVS = FCHTLV*(EPVC*EFCPR+(1.-EPVC)*EFCCH)
```

L2710: The variable GRLVS, rate of increase in dry weight of leaf blades, is calculated as the product of FCHTLV (L2690) and the conversion efficiency of primary assimilates in leaf blade tissue, which is calculated from EPVC, the fraction of proteins in the leaf blades and the conversion efficiency into structural proteins and structural carbohydrates, EFCPR and EFCCH, respectively.

```
ONLV = (WLVS+GRLVS*DELT)*FNMX
```

L2720: The variable ONLV, maximum weight of nitrogen in leaf blades, is calculated as the sum of WLVS, dry weight of leaf blades and the actual growth rate of leaf blades, GRLVS, times DELT, multiplied by FNMX, maximum nitrogen concentration of leaf blades as function of development stage, DVS.

```
NDEM = INSW(FRNV-FNMX,(ONLV-ANLV)/TCU,0.)*ARESPF
```

L2730: The variable NDEM, nitrogen demand of leaf blades, is set to zero if FRNV, the actual concentration of nitrogen in leaf blades, equals the maximum, FNMX, or if ARESPF, indicator for the presence of reserve carbohydrates is zero; else NDEM is calculated as the difference between ONLV (L2720), the maximum weight of nitrogen in leaf blades and ANLV, the actual weight, divided by TCU, the time constant for uptake.

```
OFNST = FNMNSR + SONCT*FNMXSA
```

L2740: The variable OFNST, maximum concentration of nitrogen in stem and sheath tissue, is calculated as FNMNSR, the maximum nitrogen concentration in mature stem and sheaths plus the product of the intermediate variable SONCT and FNMXSA, the range in maximum nitrogen concentration between young and mature stem and sheaths.

```
FCHST = FTSTE*FCHN*RFSTRS
```

L2750: The variable FCHST, current flow of carbohydrates to stem and sheaths, is calculated as the product of FTSTE (L2560), fraction carbohydrate allocated to stem and sheaths, FCHN (L2530), net flow of carbohydrates and RFSTRS (L2580), reduction factor on leaf growth.

```
FPST = 6.25*ANSTE/(WSTEM+REANOT(WSTEM))
```

L2760: The variable FPST, fraction of protein in stem and sheaths, is calculated as 6.25 times ANSTE, weight of nitrogen in stem and sheath tissue divided by WSTEM, the dry weight of live stem and sheaths.

```
GRRSTE = FCHST*(FPST*EFCPR+(1.-FPST)*EFCCH)
```

L2780: The variable GRRSTE, rate of increase in dry weight of stem and sheaths, is calculated from the product of FCHST (L2750) and the conversion efficiency of primary assimilates in stem and sheath tissue which is calculated from FPST, fraction of proteins in stem and sheaths and the conversion efficiency into structural proteins and structural carbohydrates, the variables EFCPR and EFCCH, respectively.

```
      NDEMST = INSW(FRNST-OFNST,((WSTEM+GRRSTE*DELT)*OFNST -
$              ANSTE)/TCU,0.)*ARESPF
```

L2780: The variable NDEMST, nitrogen demand of live stem and sheaths, is set to zero if FRNST, the actual concentration of nitrogen in stem and sheaths is equal to or exceeds the value of OFNST, the maximum nitrogen concentration in stem and sheaths or if ARESPF, the indicator for the presence of reserve carbohydrates equals zero. Else NDEMST is calculated as the sum of WSTEM, dry weight of stem and sheaths and the actual growth rate of stem and sheaths times DELT, multiplied by OFNST minus ANSTE, the actual weight of nitrogen in stem and leaf sheaths, divided by TCU, the time constant for nitrogen uptake.

```
TNDEM = DNRT+NDEM+NDEMST
```

L2800: The variable TNDEM, total demand for nitrogen of vegetative plant material is calculated as the sum of DNRT, the demand of the roots, NDEM, the demand of the leaf blades and NDEMST the demand of stem and sheaths.

```
NBR = RAIN*NCR+IRRID*NCIW
```

L2810: The variable NBR, contribution of nitrogen by rain and free living microorganisms is calculated as the product of RAIN, rain intensity and NCR, the nitrogen concentration in rain water taking also into account fixation by free living microorganisms plus the product of IRRID, the current irrigation rate times NCIW, the nitrogen content of irrigation water.

```
SLTF(1) = NBR
```

L2820: The variable SLTF(1), rate of inflow of nitrogen in the top soil compartment, assumes the value of NBR (L2810).

```
CONP(1) = (NCR*RAIN+NCIW*IRRID) / (RAIN+IRRID + REANOT(RAIN+IRRID))
```

L2830: The variable CONP(1), concentration of mineral nitrogen in the water entering the top soil compartment, is calculated as the sum of NCR times RAIN, the contribution of mineral nitrogen by rainfall, and NCIW times IRRID, the contribution by irrigation water, divided by the sum of both rainfall and irrigation.

```
TNRT = 0.
```

L2840: The variable TNRT, total available mineral nitrogen in wet rooted zone, is initialized at zero.

```
TNUME = 0.
```

L2850: The variable TNUME, total nitrogen uptake by mass flow, is initialized at zero.

```
      DO 240 I = 1,ILAYER
      CONC(I) = (ASLT(I)+(RWF(I)*CONP(I)-RNAC(I))*DELT)/(WATER(I) +
     $         RWF(I)*DELT)
```

L2860: The array CONC, concentration of mineral nitrogen per soil compartment, is calculated as the sum of ASLT(I), the weight of mineral nitrogen in soil compartment I, and the product of RWF(I), the rate of flow of soil moisture into a soil compartment, and CONP(I), the concentration of mineral nitrogen in the overlying soil compartment, minus RNAC(I), the rate of nitrogen immobilization by soil microbes in soil compartment I, times DELT. The total weight of mineral nitrogen is divided by the sum of WATER(I), the actual soil moisture content in compartment I, and RWF(I), the rate of flow of soil moisture into that compartment times DELT to calculate the concentration ('perfect mixing' principle).

```
      CONP(I+1) = CONC(I)
```

L2870: The variable CONP(I+1), concentration of mineral nitrogen entering soil compartment I+1 assumes the value of CONC(I) (L2860).

```
      SLTF(I+1) = RWF(I+1)*CONC(I)
```

L2880: The variable SLTF(I+1), rate of inflow of nitrogen in soil compartment I+1, is calculated as the product of RWF(I+1), the rate of flow of soil moisture into compartment I+1, and CONC(I) (L2860).

```
      RNUM(I) = AMIN1(TRR(I)*CONC(I),ASLT(I)/DELT-RNAC(I))* INSW(-TNDEM,1.,0.)
```

L2890: The array RNUM, rate of nitrogen uptake by mass flow per soil compartment, is set to zero if TNDEM, the nitrogen demand of the canopy is zero else; it is calculated as the minimum of the product of TRR(I), the actual rate of moisture uptake for transpiration in soil compartment I, and CONC(I) (L2860) or the difference between ASLT(I), the weight of mineral nitrogen in soil compartment I, divided by DELT minus RNAC(I), the rate of immobilization of nitrogen in compartment I.

```
      TNRT = TNRT+(ASLT(I)-RNAC(I)*DELT)*RTL(I)/THCKN(I)*AWATF(I)
```

L2900: The variable TNRT, total amount of available mineral nitrogen in wet rooted zone, is calculated by adding for each layer in which soil moisture is available (AWATF(I) equals 1) the difference between the weight of mineral nitrogen in soil compartment I and RNAC(I) (L2270) times DELT; this weight is multiplied with the ratio root length in compartment I, RTL(I) and the thickness of the compartment, THCKN(I).

```
      FERTAP(I) = APFERT*DISTF(I)
```

L2910: The array FERTAP(I), rate of addition of mineral nitrogen by fertilizer application per soil compartment, is calculated as the product of APFERT, the fertilizer application rate and DISTF(I), the distribution factor for soil compartment I.

```
      TNUME = TNUME+RNUM(I)
240   CONTINUE
```

L2920: The variable TNUME, intermediate variable in the calculation of total nitrogen uptake by mass flow, is calculated by adding the value of RNUM(I) (L2890), the rate of nitrogen uptake by mass flow for each soil compartment I.

```
TNUM = AMIN1(TNUME,TNDEM)
```

L2930: The variable TNUM, total rate of nitrogen uptake by mass flow, assumes the value of the variable TNUME (L2920) restrained to a maximum of TNDEM (L2800) with an AMIN1 function.

```
      DO 250 I = 1,ILAYER
         RNUM(I) = RNUM(I)*TNUM/(TNUME+REANOT(TNUME))
250   CONTINUE
```

L2935: The array RNUM, rate of nitrogen uptake by mass flow per soil compartment is recalculated by multiplying the original value of RNUM(I) by the ratio of TNUM (L2930) and TNUME (L2920).

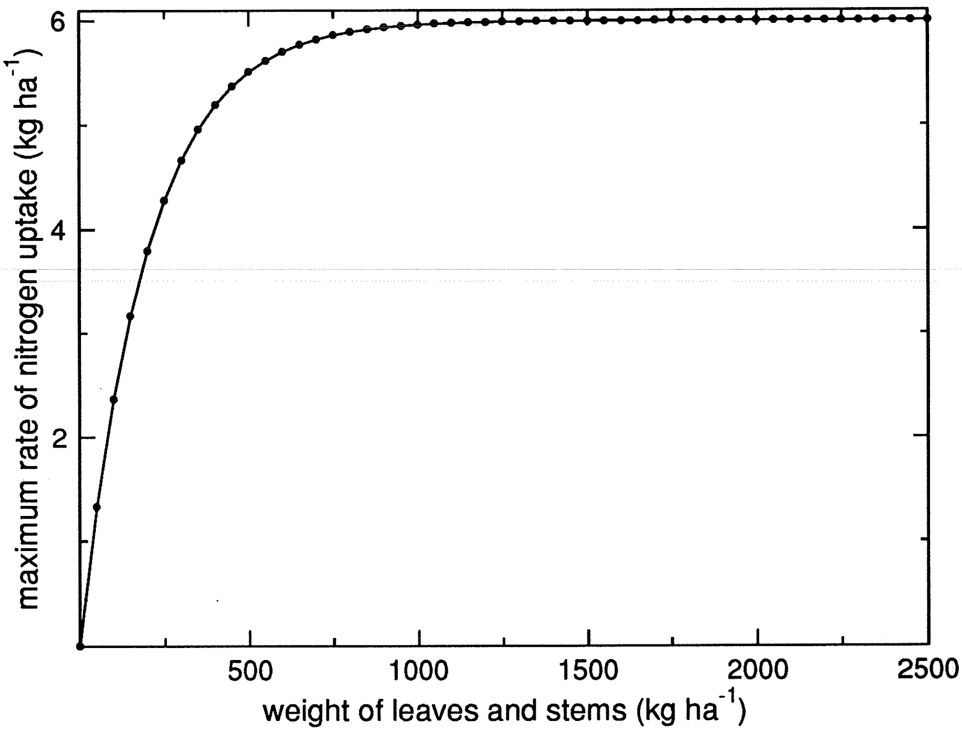

Figure 31 Relation between the sum of the dry weight of leaves and stems and the maximum rate of nitrogen uptake by the vegetation, assuming a potential rate of nitrogen uptake by a closed canopy of 6 kg ha^{-1} d^{-1} and a conversion factor from dry weight to relative root extension of 100.

```
MXRUP = UMXR*(1.-EXP(-0.5*(WLVS+WSTEM)/CF))
```

L2940: The variable MXRUP, maximum rate of nitrogen uptake by the vegetation, is calculated from UXMR, the maximum rate of nitrogen uptake by a closed canopy and an exponential reduction factor in which the sum of the dry weights of the leaf blades and stems, WLVS and WSTEM and CF, the conversion factor from dry weight to relative root extension, are arguments in the exponent (Fig. 31).

```
PNUDP = AMIN1(TNDEM,TNRT/DELT,MXRUP*TCUD/DELT)
```

L2950: The variable PNUDP, maximum rate of nitrogen uptake by the canopy, is calculated as the minimum of TNDEM (L2800), the demand, TNRT (L2900) divided by DELT, availability or MXRUP (L2940), maximum uptake rate times TCUD, the time constant of nitrogen uptake by diffusion, divided by DELT.

```
      RVOLA = ASLT(1)*(1.-RANO3)/TCV *
   $        LIMIT(0.,1.,ASLT(1)*(1.-RANO3)/LNH4)*NH4FP
```

L2960: The variable RVOLA, rate of ammonia volatilization from the top compartment, is calculated from ASLT(1), amount of mineral nitrogen in top soil compartment, RANO3, the proportion of nitrate in the total inorganic nitrogen store in the top soil compartment, TCV, time constant for ammonia volatilization, LNH4, threshold concentration of NH4 below which no volatilization occurs and NH4FP a variable which indicates the fraction of ammoniacal compounds in fertilizer.

```
      RNUDF = INSW(DVS-0.6,AMAX1(0.,(PNUDP-TNUM)*DELT/TCUD),0.)
```

L2980: The variable RNUDF, required contribution from diffusion to nitrogen supply, is set to zero if DVS, the overall development stage is greater than or equals 0.6; else it is calculated as the difference between PNUDP (L2950) and TNUM (L2930) times DELT and divided by TCUD, the time constant for uptake of nitrogen by diffusion. The value of RNUDF is restrained to a minimum of zero with an AMAX1 function.

```
         TNUDF = 0.
```

L2990: The variable TNUDF, total rate of nitrogen uptake by diffusion, is initialized at zero.

```
      DO 260 I = 1,ILAYER
         RNUD(I) = RNUDF*(ASLT(I)-RNAC(I)*DELT)/(TNRT+REANOT(TNRT))*
   $            RTL(I)/THCKN(I)
```

L3000: The array RNUD, potential rate of nitrogen uptake by diffusion per compartment, is calculated from the variables RNUDF (L2980), ASLT(I), weight of mineral nitrogen in compartment I, RNAC(I), weight of immobilized nitrogen, DELT, TNRT (L2900), RTL(I), effective root length in compartment I and THCKN(I), the thickness of compartment I.

```
      RNUDB(I) = AMIN1(RNUD(I),ASLT(I)/DELT-RNUM(I)-RNAC(I))*AWATF(I)
```

L3010: The array RNUDB, actual rate of nitrogen uptake by diffusion per soil compartment, is set to zero if no soil moisture is available when AWATF equals zero. If AWATF equals one, RNUDB(I) is calculated as the minimum of RNUD(I) and the difference between ASLT(I) divided by DELT and RNUM(I), the uptake by mass flow in compartment I minus RNAC(I), the immobilization by soil microbes in compartment I.

```
      TNUDF = TNUDF+RNUDB(I)
```

L3020: The variable TNUDF, total rate of nitrogen uptake by diffusion, is calculated by adding the values of RNUDB(I) (L3010) for each soil compartment I.

```
      IF (I.EQ.1) THEN
         RVOL(1) = RVOLA
      ELSE
         RVOL(I) = 0.
      END IF
```

L3030: The array RVOL, rate of ammonia volatilization per soil compartment is set to zero for all soil compartments except the top soil compartment, where it is assigned the value of RVOLA (L2960).

```
      DASLT(I) = SLTF(I)-SLTF(I+1)-RNUM(I)-RNUDB(I)-RNAC(I)-RVOL(I)
     $            +RHMIN(I)+(1.-FNIMH)*(RNRLB(I)+RNRL(I))+FERTAP(I)
260   CONTINUE
```

L3040: The array DASLT, rate of change in weight of mineral nitrogen per soil compartment, is calculated as SLTF(I), the rate of inflow in soil compartment I, minus SLTF(I+1), the rate of outflow, minus RNUM(I), the rate of nitrogen uptake by mass flow in soil compartment I, minus RNUDB(I), the rate of nitrogen uptake by diffusion in compartment I, minus RNAC(I), the rate of nitrogen immobilization by soil microbes in soil compartment I, minus RVOL(I), the rate of volatilization, plus RHMIN(I), the rate of nitrogen mineralization from stable organic material in soil compartment I, plus the sum of the contribution of nitrogen mineralization from decomposing microorganisms (RNRLB(I)) in soil compartment I and fresh organic material (RNRL(I)) in soil compartment I, times the proportion of nitrogen from that material that is immobilized in stable organic material, 1 minus FNIMH plus FERTAP(I), the weight of mineral nitrogen added as fertilizer in soil compartment I.

```
      RNU = TNUM+TNUDF
```

L3050: The variable RNU, total rate of nitrogen uptake by the vegetation, is calculated as the sum of TNUM, total nitrogen uptake by mass flow and TNUDF, total uptake by diffusion.

```
      RNURT = RNU*DNRT/(TNDEM+REANOT(TNDEM))
```

L3060: The variable RNURT, rate of nitrogen accumulation in the roots, is calculated by multiplying RNU (L3050) by the ratio DNRT, nitrogen demand of the roots, and TNDEM, total nitrogen demand.

```
      RNUVP = RNU-RNURT
```

L3070: The variable RNUVP, rate of nitrogen uptake by above ground organs, is calculated as the difference between RNU (L3050) and RNURT (L3060).

```
      RNUV = RNUVP*NDEM/(NDEM+NDEMST+REANOT(NDEM+NDEMST))
```

L3080: The variable RNUV, rate of nitrogen uptake by leaf blades, is calculated as RNUVP (L3070) times the ratio of NDEM (L2730) and NDEM plus NDEMST, the nitrogen demand of the stem.

```
      RNUST = RNUVP*NDEMST/(NDEM+NDEMST+REANOT(NDEM+NDEMST))
```

L3090: The variable RNUST, rate of nitrogen uptake by stem and sheaths, is calculated as RNUVP times the ratio of NDEMST and NDEM plus NDEMST.

```
      PTNLST = AMAX1(0.,(WSTEM*OFNST-ANSTE-RNUST*DELT)/TCU)
```

L 3100: The variable PTNLST, potential rate of nitrogen accumulation in the stem from translocation, is calculated as the product of WSTEM, dry weight of stem and leaf sheaths and OFNST, the maximum concentration in stem and sheaths, minus ANSTE, the weight of nitrogen in stem and leaf sheaths, minus RNUST (L3090) times DELT. The result is divided by TCU, the time constant for nitrogen uptake. The value of PTNLST is restrained to a minimum of zero with an AMAX1 function.

```
      SLTFD = SLTF(K)
```

L3120: The variable SLTFD, rate of leaching of nitrogen beyond the potential rooting zone, assumes the value of SLTF(K), the inflow of nitrogen in soil compartment K. The value of K was calculated in a procedure within section L1760.

```
TMPCX(1) = TMPC
DTMPX = DELAY(11,11,TMPCX)
DTMPA = DTMPX+INSW(TIME-10.,0.1*TSI,0.)
```

L3130: The variable DTMPA, average daily canopy temperature ten time intervals ago which is used for the calculation of soil temperatures is calculated using TMPC (L2040), average daily canopy temperature, dummy array TMPCX and the DELAY function. During the first ten time intervals of simulation, DTMPA equals TMPA the average daily temperature on the day of initialization of the plant module.

```
ABGDMF = INSW(-(WLVS+WSTEM+WGR),1.,0.)
```

L3140: The factor ABGDMF, a switch variable assuming the value 1, if dry matter is present above ground, otherwise 0, is calculated with an INSW function from the sum of WLVS, WSTEM and WGR.

```
DVSVF = INSW(DVSV-1.,0.,1.)
```

L3150: The value of the factor DVSVF, indicating whether development stage is beyond anthesis (1) or not (0) is set with an INSW function. If DVSV, the development stage in the pre-anthesis phase exceeds or equals one, variable DVSVF assumes the value one, else zero.

```
NFD = INSW(DVSV-DVSFE, NFDEV+(1.-NFDEV)*RFNS, 1.)
```

L3160: The variable NFD, effect of crop nitrogen status on phenological development up to end of flower initiation, is calculated, if DVSV is lower than DVSFE, as NFDEV plus one minus NFDEV times RFNS (L1490), else it is set to one.

```
      DVRV = CULTP*AMAX1(0.,0.00094*TMPC-0.00046)
$            *(1.-DVSVF)*ABGDMF*NFD*INSW(PUSHDI-0.5,0.,1.)
```

L3170: The variable DVRV, rate of development in the pre-anthesis phase, is calculated from the variables CULTP, a cultivar-specific factor to account for differences in pre-anthesis development rate, TMPC (L2040), DVSVF (L3150), ABGDMF (L3140), NFD (L3170) and a switch function in which PUSHDI is argument. Before emergence, when the variable PUSHDI equals zero, DVRV is set to zero.

```
DVSVP = DVSVX
DVSVX = DVSV
```

L3180: The variable DVSVP, development stage in the pre-anthesis phase one time interval ago, assumes the value of DVSVX, a dummy variable that is used to store that development stage. Subsequently DVSVX assumes the value of DVSV.

```
DVSP = DVSX
DVSX = DVS
```

L3190: The variable DVSP, overall development stage in the pre-anthesis phase one time interval ago, assumes the value of DVSX, a dummy variable that is used to store that development stage. Subsequently DVSX assumes the value of DVS.

```
DVSPRF = INSW(DVSV-DVSPRE,0.,1.)
```

L3200: The factor DVSPRF, indicating whether development stage is beyond the end of tiller formation (1) or not (0) is calculated with an INSW function.

```
    DVRR = CULTM*INSW(PUSHDI-0.5,0.,1.)*(0.000913*TMPC+0.003572)
$          *DVSVF*ABGDMF
```

L3210: The variable DVRR, rate of development in the post-anthesis phase, is calculated using the variables CULTM, a cultivar-specific factor to account for differences in post-anthesis development rate, PUSHDI, TMPC (L2040), DVSVF (L3150) and ABGDMF (L3140). Before emergence, when the variable PUSHDI equals zero, DVRR is set to zero.

```
    EGFF = INSW(DVSR-DVSEGF,0.,1.)
```

L3220: The variable EGFF, indicating whether development stage is beyond end of grain fill is calculated with an INSW function; when DVSR is lower than DVSEGF, EGFF equals zero, else it assumes the value one.

```
    DEHYD = AMAX1(0.,1.E4*(PCTRAN-TRAN)*DELT)
```

L3240: The variable DEHYD, rate of dehydration of plant tissue, difference between cuticular water loss and water uptake is calculated as 10^4 times the difference between PCTRAN, rate of cuticular water loss with fully closed stomata and TRAN, the actual rate of canopy water uptake as dictated by actual soil moisture conditions, times DELT.

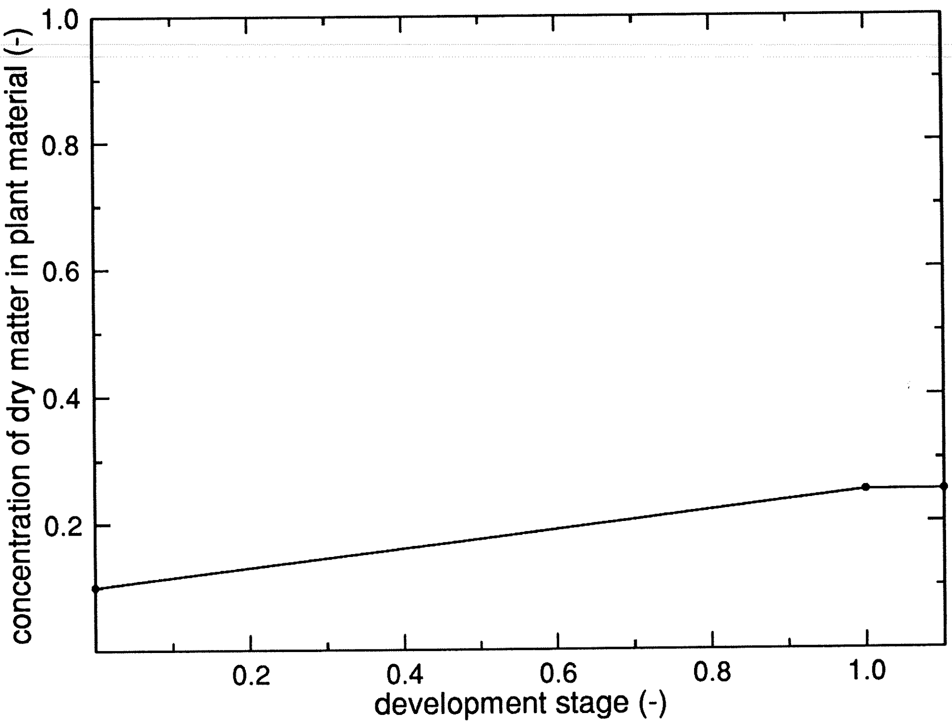

Figure 32 Function FDMT, fraction dry matter in vegetative tissue as function of development stage.

```
    FDM = LINT(FDMT,IFDMT,DVS)
```

L3250: The variable FDM, fraction dry matter in vegetative tissue, is calculated by linear interpolation in the function FDMT (Fig. 32), with development stage, DVS, as independent variable.

```
         DRLVWS = AMIN1(WLVS,DEHYD*FDM/(1.-FDM-FWDB)*WLVS/(WLVS+
      $           0.5*WSTEM+REANOT(WLVS)))/TCDDH
```

L3260: The variable DRLVWS, potential death rate of leaf blades due to water shortage, is calculated from the variables DEHYD (L3240), the rate of dehydration of plant tissue, FDM (L3250), FWDB, fraction of water left in dying plant tissue, WLVS and WSTEM, dry weight of leaf blades and stem and sheaths, respectively and TCDDH, the time constant for dehydration of plant tissue. The potential death rate of leaf blades can not exceed WLVS, dry weight of the leaf blades.

```
         RDRW = INSW(DVS-0.5,AMIN1(0.005,DRLVWS/(WLVS+REANOT(WLVS))),
      $           DRLVWS/(WLVS+REANOT(WLVS)))
```

L3270: The variable RDRW, relative death rate of leaf blades due to soil moisture shortage, is calculated as DRLVWS, the potential death rate of leaf blades due to soil moisture shortage, divided by WLVS, the dry weight of the leaf blades. The value of RDRW is restrained to a maximum of 0.005 with an AMIN1 function before the overall development stage of the crop, DVS, reaches 0.5; after anthesis there is no maximum defined.

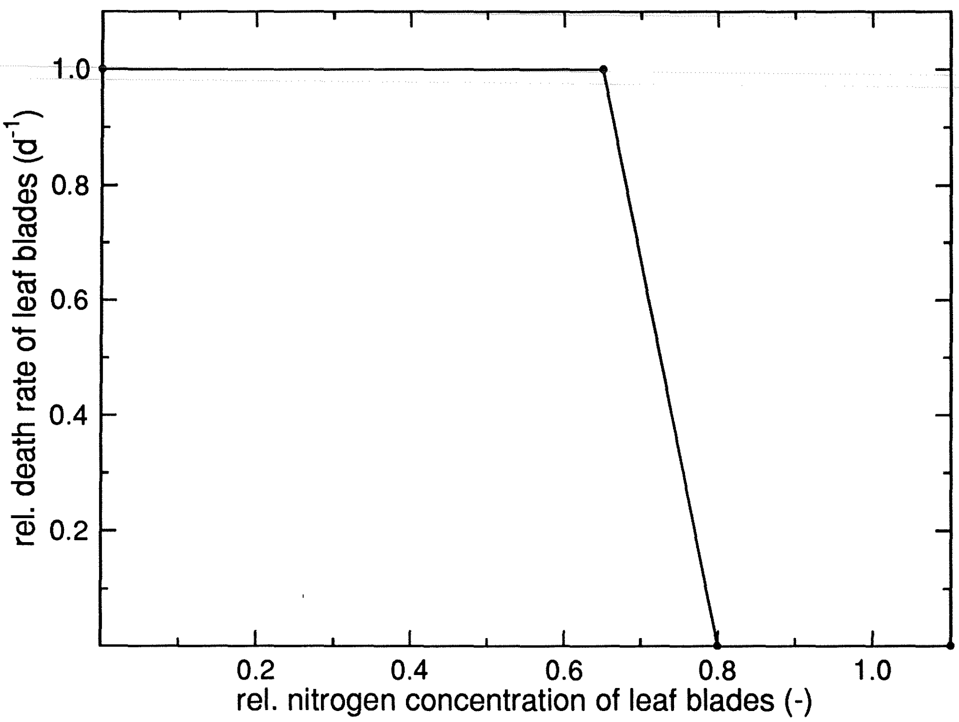

Figure 33 Function RDRNT, relative death rate of leaf blades due to nitrogen shortage as function of the relative nitrogen concentration in leaf blades.

```
         RDRN = 0.2*LINT(RDRNT,IRDRNT,FNC)
```

L3280: The variable RDRN, relative death rate of leaf blades due to nitrogen shortage, is calculated as 0.2 times the value calculated by linear interpolation in the function RDRNT (Fig. 33) with the relative nitrogen concentration in leaf blades, FNC, as independent variable.

```
RDRL = LIMIT(0.,MRDRSH,(LAI-LAILM)*MRDRSH/LAILM)
```

L3290: The variable RDRL, relative death rate of leaf blades due to shading, is calculated as the difference between LAI, the leaf area index, and LAILM, the threshold value of LAI beyond which death due to shading starts, times the ratio MRDRSH, the maximum relative death rate of leaf blades due to shading and LAILM. The value of RDRL is restrained between zero and MRDRSH with a LIMIT function.

```
      RDRCS = -MXRDR *
    $   LIMIT(-1.,0.,FCHNX/((DGAS-FCHNX)+REANOT(DGAS-FCHNX)))
```

L3300: The variable RDRCS, relative death rate of leaf blades due to carbohydrate exhaustion, is calculated as the product of minus MXRDR, the maximum relative death rate of leaf blades due to carbohydrate shortage, times the ratio FCHNX, net flow of carbohydrates, and the difference between gross assimilation and net flow of carbohydrates. Consequently, RDRCS will only differ from zero when the costs of maintenance respiration exceed gross assimilation.

```
RDR = AMAX1(RDRW,RDRN,RDRL,RDRCS)
```

L3310: The variable RDR, overall relative death rate of leaf blades, assumes the value of the maximum of RDRW (L3270), RDRN (L3280), RDRL (L3290) and RDRCS (L3300) using an AMAX1 function.

```
FLFAR = AMAX1(FLFARM,FLFARI-(FLFARI-FLFARM)*DVSV/DVSST)
```

L3320: The variable FLFAR, current specific leaf area, is calculated as the maximum of FLFARM, minimum value of specific leaf area and FLFARI, maximum specific leaf area at emergence, minus the difference between maximum and minimum specific leaf area times the ratio DVSV, development stage in the pre-anthesis phase and DVSST, the development stage in the pre-anthesis phase at start of stem elongation.

```
DRNT = RDRT*DTLN*(1.-DTLN/(TNNR+REANOT(TNNR)))
```

L3330: The variable DRNT, death rate of non-reproductive tillers, is calculated as the product of RDRT, the relative death rate of non-reproductive tillers, DTLN, total number of dead tillers, and one minus the ratio DTLN over TNNR, the total number of non-reproductive tillers.

```
RWLLDT = AMIN1(WLVS/4.,DRNT*LWDTL)
```

L3340: The variable RWLLDT, rate of loss of dry weight of leaf blades through dying of leaf blades of non-reproductive tillers, is calculated as the product of DRNT (L3330) and LWDTL, the average dry weight of leaf blades of non-reproductive tillers at the end of ear formation. The value of RWLLDT is restrained to a quarter of WLVS, total dry weight of the leaf blades, with an AMIN1 function.

```
DRQ = RDR*WLVS+RWLLDT
```

L3350: The variable DRQ, potential death rate of leaf blades due to soil moisture shortage, nitrogen shortage, shading or carbohydrate shortage, is calculated as the product of RDR (L3310) and WLVS, the dry weight of the leaf blades, plus RWLLDT (L3340).

```
         PDRLVS = 0.
         RDLFA = 0.
         DRQR = 0.
         DO 270 I = 1,110
           DLEAF(I) = 0.
           DALFT(I) = 0.
           DLA(I) = 0.
270      CONTINUE
```

L3360-3405: The following variables are initialized at zero: PDRLVS, potential death rate of leaf blades due to senescence, RDLFA, rate of decline in leaf area due to senescence, DRQR, intermediate variable in calculation of potential death rate of leaf blades due to senescence, DLEAF(I), rate of change in leaf dry weight in class I, DALFT(I), rate of change in cumulative temperature sum per leaf class and DLA(I), the specific leaf area of leaf blades in class I.

```
       IF (GRLVS.GT.0..OR.NINT(PUSHD).EQ.1) THEN
         LG = LG + 1

         IF (LG.GT.110) THEN
            LG = 110
         END IF

         DLEAF(LG) = GRLVS
         DLA(LG) = FLFAR*INSW(LEAFA(LG)-0.5,1.,0.)

*-----initialization of first leaf class at emergence
         IF (NINT(PUSHD).EQ.1) THEN
            DLEAF(1) = WLVSI/DELT
            DLA(1) = FLFARI/DELT
         END IF
       END IF

       IF (LG.NE.0) THEN

         DO 290 I = 1,LG
            DALFT(I) = TMPC

            IF (ALFT(I).GE.(15.0*AVLTLF)) THEN
               DLEAF(I) = -LEAFW(I)/DELT
               PDRLVS = PDRLVS + LEAFW(I)/DELT
               RDLFA = RDLFA+LEAFW(I)*LEAFA(I)/DELT
            END IF

290       CONTINUE

         IF (PDRLVS.LT.DRQ) THEN
            RDLFA = 0.
            DRQX = DRQ
```

```
          DO 310 I = 1,LG
            LFD = LEAFW(I)/DELT

            IF (I.EQ.LG) THEN
              LFD = LFD + GRLVS
            END IF

            DLEAF(I) = AMIN1(LFD,DRQX)
            RDLFA = RDLFA+DLEAF(I)*LEAFA(I)
            DLEAF(I) = -DLEAF(I)
            DRQX = DRQX - LFD

            IF (I.EQ.LG.AND.DRQX.GT.0.) THEN
              DRQR = DRQX
            END IF

            IF (DRQX.LE.0.) THEN
              GO TO 325
            END IF
310         CONTINUE
        END IF
      END IF

325   CONTINUE
      DRLVS = AMAX1(0.,DRQ-DRQR,PDRLVS)
```

L3420: The variable DRLVS, current overall death rate of leaf blades is calculated as the maximum of the difference between DRQ (L3410) and DRQR, intermediate variable in calculation of potential overall death rate of leaf blades, or PDRLVS, the potential death rate of leaf blades due to senescence.

```
      RDRD = PDRLVS/(WLVS+REANOT(WLVS))
```

L3430: The variable RDRD, relative death rate of leaf blades due to senescence, is calculated as the potential death rate of the leaf blades, PDRLVS divided by WLVS, the dry weight of the leaf blades.

```
      DRSWS = AMIN1(WSTEM/DELT,DEHYD/(1.-FDM-FWDB)*0.5*WSTEM/
     $        (WLVS+0.5*WSTEM+REANOT(0.5*WSTEM+WLVS))/TCDDH)
```

L3432: The variable DRSWS, potential death rate of stem and sheaths due to soil moisture shortage, is calculated from the variables DEHYD (L3240), FDM, fraction of dry matter in vegetative tissue, FWDB, fraction of water left in dying plant tissue, WSTEM and WLVS, dry weight of live stem and sheaths and leaf blades, respectively and TCDDH, the time constant for dehydration. The value of DRSWS is restrained to WSTEM, the actual dry weight of stem and sheaths divided by DELT with an AMIN1 function.

```
      RDRWS = DRSWS/(WSTEM+REANOT(WSTEM))
```

L3434: The variable RDRWS, relative death rate of stem and sheaths due to soil moisture shortage, is calculated as DRSWS (L3432) divided by WSTEM, the dry weight of live stem and sheaths.

```
      RDRWDS = AMAX1(RFST*RDRD,RDRWS)
```

L3436: The variable RDRWDS, relative death rate of stem and sheaths due to either soil moisture shortage or senescence, is calculated as the maximum of the product of RFST, the proportionality factor between relative death rate of stem and sheaths and leaf blades, times RDRD (L3430) and RDRWS (L3434).

```
      RDRS = INSW(-DRLVS,AMAX1(RFST*RDR,RSLDS*RDRD),RDRCS)
```

L3440: The variable RDRS, relative death rate of stem and sheaths, is calculated as RDRCS (L3300), relative death rate of leaf blades due to carbohydrate exhaustion, if DRLVS (L3420) the current death rate of leaf blades due to all causes equals zero. If DRLVS is positive, RDRS is calculated as the maximum of the products of RFST and RDR, or RSLDS and RDRD (L3430).

```
      DRRT = WRT*RDRS*RFRT
```

L3450: The variable DRRT, death rate of the roots, is calculated as WRT, the dry weight of the roots times RDRS, the relative death rate of stem and sheaths, times RFRT, the proportionality factor between the relative death rates of roots and stem and sheaths.

```
      RNLDR = DRRT*INSW(RDRWDS-RFST*RDRN,RN,FRNR)
```

L3460: The variable RNLDR, rate of nitrogen loss from dying roots is calculated as DRRT, the death rate of the roots, times FRNR, the concentration of nitrogen in the roots. If roots die due to nitrogen shortage (RFST times RDRN exceeds RDRWDS), DRRT is multiplied by RN, the residual nitrogen concentration in the roots.

```
      DO 330 I=1,ILAYER
         DFON(I) = -RNRL(I)+DFFOM(I)*RNLDR
```

L3480: The array DFON, the rate of change in weight of nitrogen in fresh organic material per soil compartment, is calculated as the sum of minus RNRL(I), the rate of nitrogen mineralization through decomposing fresh organic material in soil compartment I plus the product of DFFOM(I), the distribution factor for fresh and stable organic material in the soil profile times RNLDR, the rate of nitrogen loss from dying roots.

```
      DHUM(I) = (FNIMH*(RNRL(I)+RNRLB(I))-RHMIN(I))/NCH*10.
```

L3490: The array DHUM, rate of change in weight of stable organic material per soil compartment, is calculated from FNIMH, the concentration of nitrogen in decomposing fresh organic material and microbial biomass immobilized in stable organic material, RNRL(I), RNRLB(I), rate of nitrogen mineralization from dying microorganisms in soil compartment I, RHMIN(I), rate of nitrogen mineralization from stable organic material in soil compartment I and NCH, the nitrogen concentration in stable organic material.

```
      DNHUM(I) = FNIMH*(RNRL(I)+RNRLB(I))-RHMIN(I)
```

L3500: The array DNHUM, rate of change in weight of nitrogen in stable organic material per soil compartment I is calculated from the variables FNIMH, RNRL(I), RNRLB(I) and RHMIN(I), all described in L3490.

```
      DFOM(I) = -DECR(I)+DFFOM(I)*DRRT
330   CONTINUE
```

L3510: The array DFOM, rate of change in weight of fresh organic material per soil compartment is calculated as minus DECR(I), the rate of decomposition of fresh organic material in soil compartment I plus the product of DFFOM(I) and DRRT (L3450).

```
CRCWLV = GRLVS-DRLVS+PUSHD*WLVSI/DELT
```

L3520: The variable CRCWLV, 'uncorrected' rate of change in dry weight of the leaf blades, is calculated as GRLVS, the rate of increase in dry weight of the leaf blades, minus DRLVS (L3420), the current death rate of leaf blades. The variable WLVSI, initial dry weight of live leaf blades divided by DELT is added at emergence when the value of PUSHD equals one.

```
RNL = DRLVS*(FRNV-(FRNV-BN)*(1.-RFNS))
```

L3530: The variable RNL, rate of loss of nitrogen through dying leaf blades, is calculated from the variables DRLVS (L3420), FRNV, concentration of nitrogen in leaf blades, BN, residual non-remobilizable concentration of nitrogen in leaf blades and RFNS, a factor accounting for the effect of nitrogen shortage on nitrogen loss from dying leaf blades.

```
TRNLL = AMIN1(RNL+RNTLS,ANLV/DELT)
```

L3540: The variable TRNLL, rate of nitrogen loss from live leaf blades, is calculated as the sum of the variables RNL (L3530) and RNTLS (L2440), the rate of transfer of nitrogen from the leaf blades to the grain. The maximum value of TRNLL is restrained to ANLV, total weight of nitrogen in the leaf blades divided by DELT.

```
CRCANL = RNUV-TRNLL+PUSHD*ANLVI/DELT
```

L3550: The variable CRCANL, 'uncorrected' rate of change in weight of nitrogen in live leaf blades, is calculated as RNUV (L3080), rate of uptake of nitrogen by leaf blades, minus TRNLL (L3540). The initial weight of nitrogen in the leaf blades at emergence, variable ANLVI is added to CRCANL at emergence, when the value of PUSHD equals one.

```
RCANLV = INSW(CRCANL,AMAX1(-ANLV/DELT,CRCANL),CRCANL)
```

L3560: The variable RCANLV, current rate of change in weight of nitrogen in live leaf blades, limited to weight present, assumes the value of CRCANL (L3550) restrained to the value of ANLV divided by DELT. If CRCANL is positive, RCANLV assumes the unrestricted value of CRCANL.

```
TNLST = AMIN1(PTNLST,RNL-DRLVS*BN)
```

L3570: The variable TNLST, rate of translocation of nitrogen from dying leaf blades to stem and sheaths, is calculated as RNL (L3530), the rate of loss of nitrogen from dying leaf blades minus the product of DRLVS (L3420), current death rate of leaf blades and BN, the residual non-remobilizable concentration of nitrogen in leaf blades. The maximum value of TLNST is restrained to PTNLST, potential rate of nitrogen accumulation in the stem from translocation, with an AMIN1 function.

```
RCWLVS = INSW(CRCWLV,AMAX1(-WLVS/DELT,CRCWLV),CRCWLV)
```

L3580: The variable RCWLVS, 'effective' rate of change in dry weight of live leaf blades, is calculated as CRCWLV (L3520), the 'uncorrected' rate of change, restrained to the negative value of WLVS divided by DELT, if CRCWLV is negative and larger than the current dry weight of the leaf blades.

```
DRSTE = RDRS*WSTEM
```

L3590: The variable DRSTE, death rate of stem and sheaths, is calculated as the product of RDRS (L3440), the relative death rate of stem and live sheaths , and WSTEM, the dry weight of live stem and sheaths.

```
        RWLSDT = AMIN1(DRNT*SWDTL,WSTEM/DELT-DRSTE)
```
L3600: The variable RWLSDT, rate of loss of dry weight of stem and sheaths through dying of stem and sheaths of non-reproductive tillers, is calculated as the minimum of the product of DRNT (L3330), the death rate of non-reproductive tillers and SWDTL, the final dry weight of stem and sheaths of non-reproductive tillers, or the difference between WSTEM, the dry weight of stem and sheaths, divided by DELT and DRSTE (L3590).

```
        CRCWST = GRRSTE-DRSTE-RWLSDT
```
L3610: The variable CRCWST, 'uncorrected' rate of change in dry weight of stem and sheaths, is calculated as GRRSTE, the rate of increase in dry weight of stem and sheaths minus DRSTE (L3590), the death rate of stem and sheaths, minus RWLDST (L3600), the loss through death of non-reproductive tillers.

```
        RNLDST = DRSTE*INSW(WLVS-1.,FRNST,INSW(RDRWDS-RFST*RDRN,LN,FRNST))
```
L3650: The variable RNLDST, rate of nitrogen loss through dying stem and sheaths, is calculated as the product of the variables DRSTE (L3590) and FRNST, the fraction of nitrogen in stem and sheaths; if stem and sheaths die due to nitrogen shortage (RFST times RDRN exceeds RDRWDS), DRSTE is multiplied by LN, the residual nitrogen concentration in stem and sheaths.

```
        RCWST = INSW(CRCWST,AMAX1(-WSTEM/DELT,CRCWST),CRCWST)
```
L3660: The variable RCWST, current rate of change in dry weight of stem and sheaths, limited to dry weight present, assumes the value of the variable CRCWST. If CRCWST has a negative value (decrease) its value is restrained to WSTEM, the dry weight of stems present, divided by DELT.

```
        CRCWR1 = GRRT-DRRT+PUSHD*WRTI/DELT
```
L3670: The variable CRCWR1, auxiliary variable in calculation of rate of change in dry weight of the roots is calculated as GRRT, rate of increase in dry weight of the roots, minus DRRT, the death rate of the roots. At emergence, when PUSHD equals one, initial dry weight of live roots at emergence, WRTI, is added.

```
        CRCWRT =INSW(PUSHD-0.5,INSW(-(WSTEM/DELT+RCWST+WLVS/DELT+RCWLVS)
     $              ,CRCWR1,-WRT/DELT),CRCWR1)
```
L3680: The variable CRCWRT, 'uncorrected' rate of change in dry weight of the roots, is assigned the value of CRCWR1 at emergence, after emergence it is also assigned the value CRCWR1, except when no live leaf blades and stem and sheaths are present, then all roots are supposed to die also (CRCWRT = -WRT/DELT).

```
        RCWRT = INSW(CRCWRT,AMAX1(-WRT/DELT,CRCWRT),CRCWRT)
```
L3690: The variable RCWRT, 'effective' rate of change in dry weight of the roots is assigned to the value of CRCWRT. If CRCWRT is negative and exceeds the value of minus WRT divided by DELT, then RCWRT is limited to the dry weight of the roots present.

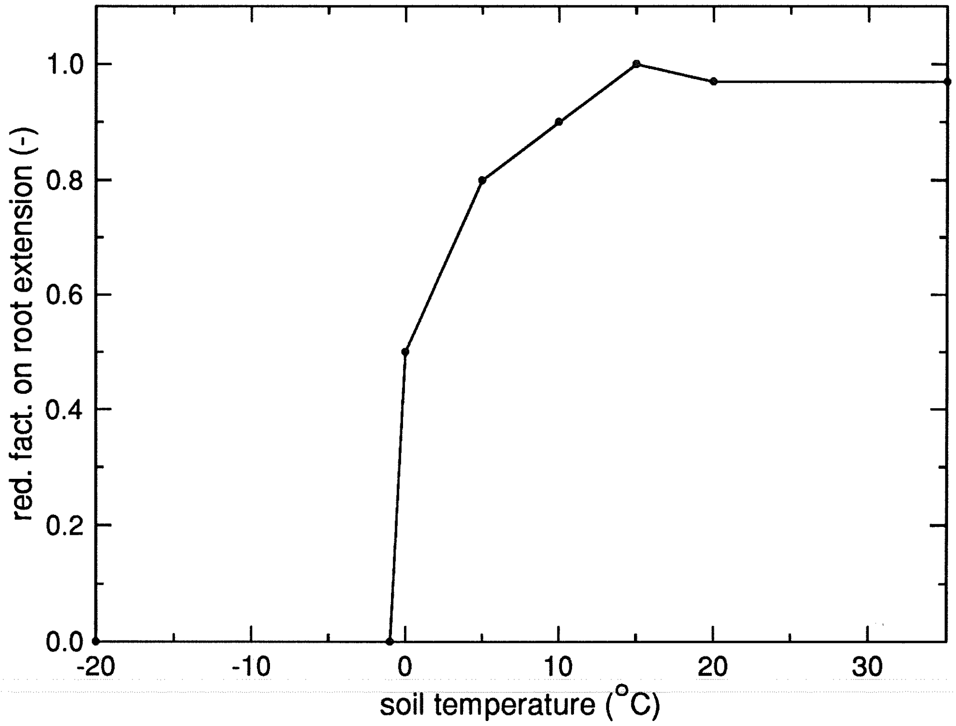

Figure 34 Function REDTTB, reduction factor on root extension as function of soil temperature.

```
      RGRRL = SWPBT*DGRRT*LINT(REDTTB,IRDTTB,TS)
$        * INSW(RTD-MXRTD,1.,0.)
$        * (RTF+1.-RTF*RFSTRS)*INSW(WLVS-WLVSI,0.,1.)
$        * INSW(-FDSR,1.,0.)
```

L3700: The variable RGRRL, rate of increase in rooting depth is calculated as the product of SWPBT, a switch variable to indicate whether the root tip is in a wet soil compartment (1) or not (0), DGRRT, maximum rate of root extension under optimum conditions, and a reduction factor on root extension which is calculated by linear interpolation in the function REDTTB (Fig. 34) with soil temperature, TS, as independent variable. If RTD, the actual rooting depth, is equal to or exceeds MXRTD, the maximum rooting depth, or WLVS, the dry weight of the leaf blades, is below WLVSI, the initial leaf dry weight, or FDSR, the fraction of current assimilate supply allocated to the roots is zero, RGRRL is set to zero.

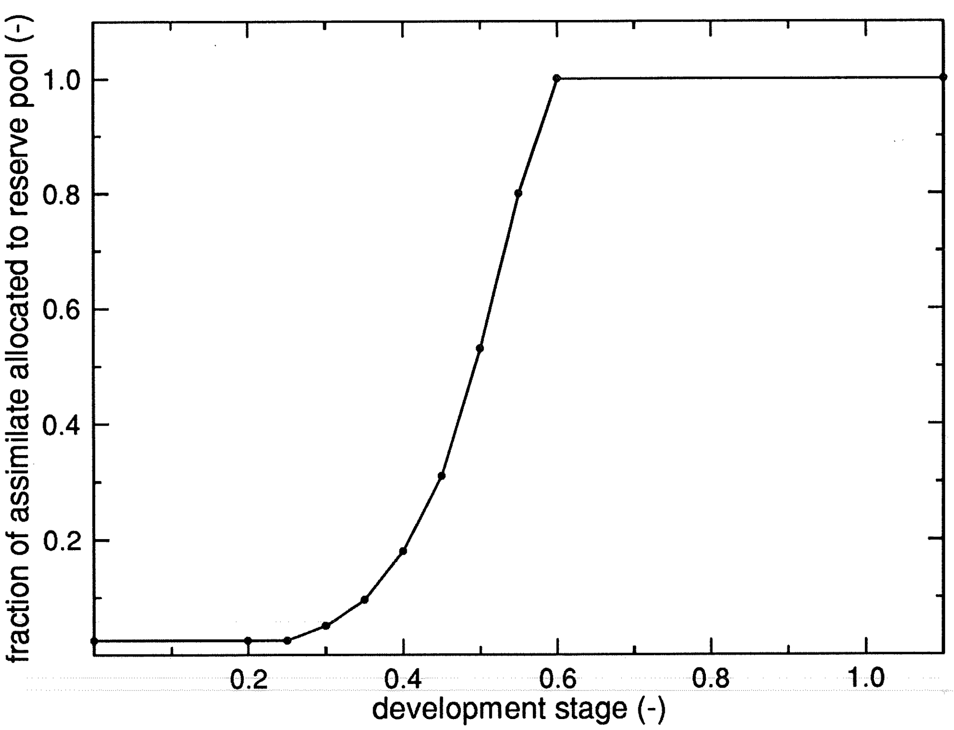

Figure 35 Function FTRLT, fraction of current assimilate supply allocated to the reserve pool as function of development stage.

```
FTRL = LINT(FTRLT,IFTRLT,DVS)
```

L3710: The variable FTRL, fraction of current assimilate supply allocated to the reserve pool is calculated by linear interpolation in the function FTRLT (Fig. 35), with the development stage, DVS, as independent variable.

```
FCHTRS = FTRL*FCHN+SCHFLV*(1.-FSCHG)
```

L3720: The variable FCHTRS, current flow of carbohydrates to reserves, is calculated as FTRL (L3710) times FCHN, net flow of carbohydrates, plus the product of SCHFLV, 'surplus' carbohydrate due to growth check of leaf blades and stem under influence of soil moisture or nitrogen stress, times one minus FSCHG, the fraction of 'surplus' carbohydrate not allocated to the roots.

```
FPGC = LIMIT(0.,1.,5.7*ANGR/(WGR+REANOT(WGR)))
```

L3730: The variable FPGC, fraction of protein in the grain, is calculated as 5.7 times ANGR, the weight of nitrogen in the grain, divided by WGR, the dry weight of the grain. FPGC is restrained between zero and one with a LIMIT function.

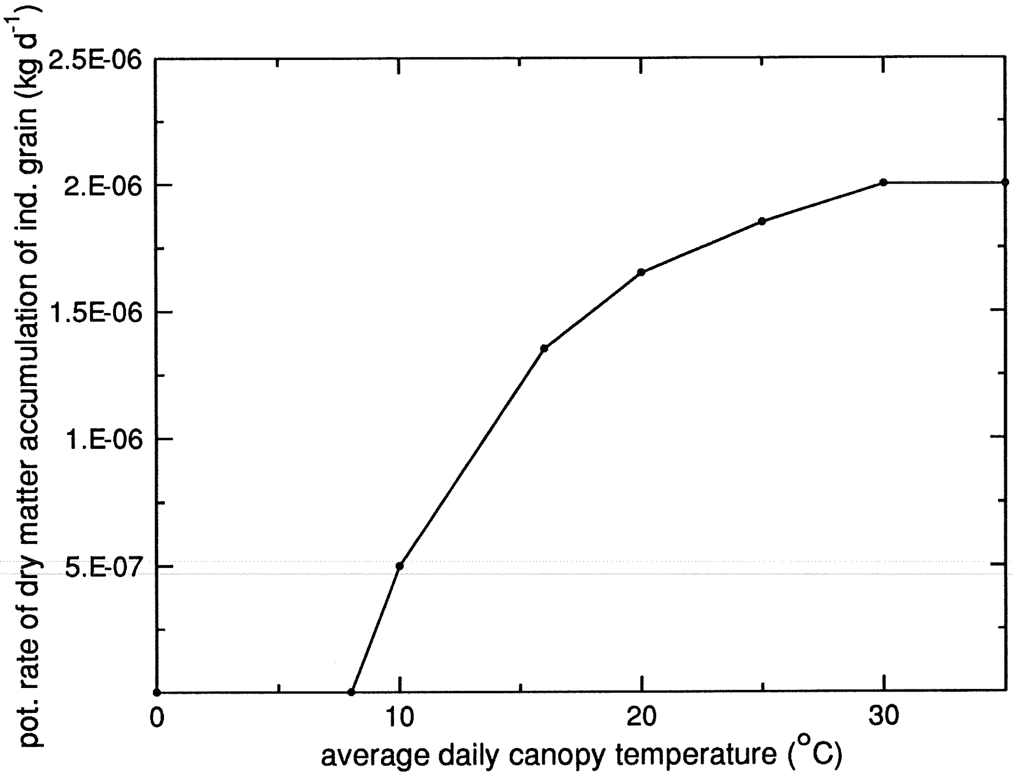

Figure 36 Function PGRIGT, potential rate of dry matter accumulation of an individual grain as function of average daily canopy temperature.

```
PGRIG = LINT(PGRIGT,IPGIGT,TMPC)*SGFF
```

L3740: The variable PGRIG, potential rate of dry matter accumulation of an individual grain, is calculated by linear interpolation in the function PGRIGT (Fig. 36) with TMPC, average daily canopy temperature as independent variable, multiplied with the variable SGFF, which indicates whether the development stage is beyond the start of grain fill (1) or not (0).

```
      PGRRG = GRN*PGRIG*(1.-EGFF)*INSW(PUSHN-0.5,
     $        (1.-LIMIT(0.,1.,(FRNGL-FRNG)/(FRNGL-FRNGL1))),1.)
```

L3750: The variable PGRRG, potential rate of dry matter accumulation in the grain, is calculated from GRN, grain density, PGRIG (L3740) and EGFF a variable which indicates whether development stage is beyond the end of grain fill (1) or not (0), variable PUSHN that assumes the value one on the day that grain fill starts and is otherwise zero, FRNG, concentration of nitrogen in the grain, FRNGL, minimum nitrogen concentration in the grain, and FRNGL1, the nitrogen concentration in the grain below which dry matter accumulation starts to be affected.

```
CEGR = FPGC*EFCPRG+(1.-FPGC)*EFCCH
```

L3760: The variable CEGR, efficiency of conversion of primary assimilation products in grain dry matter, is calculated as FPGC (L3730), fraction of protein in the grain, times EFCPRG, the efficiency of conversion of primary assimilates into structural grain proteins, plus one minus FPGC times EFCCH, the efficiency of conversion of primary assimilates into structural carbohydrates.

```
RESRES = RESLR*(WSTEM+WLVS+WRT)
```

L3770: The variable RESRES, dry weight of non-remobilizable reserves in the canopy, is calculated as the product of RESLR, concentration of non-remobilizable reserves for translocation to the grain, multiplied by the sum of WSTEM, WLVS and WRT, dry weight of live stem and sheaths, leaf blades and roots, respectively.

```
ARTTG = AMIN1(PGRRG,AMAX1(0.,((ARESP-RESRES)/TCTR)*CEGR))
```

L3780: The variable ARTTG, actual rate of transfer of reserve carbohydrates from vegetative tissue to the grain, is calculated as ARESP, the dry weight of reserve carbohydrates, minus RESRES (L3770), divided by TCTR, the time constant for translocation of reserves and multiplied with CEGR (L3760) to convert from carbohydrate to dry weight. The value of ARTTG is restrained between zero and the value of PGRRG (L3750).

```
      LMR = (ARESPF-1.)*EGFF*
     $      LIMIT(0.,RMNG,RMNG*(1.-(DVSR-0.4)/(0.7-0.4)))
```

L3790: The variable LMR, rate of loss of dry weight of the grain due to maintenance respiration before ripening of the grain, is calculated when no other carbohydrate sources are available; when variable ARESPF equals zero. When the development stage of the crop is beyond end of grain fill; i.e. variable EGFF equals one, the variable LMR is assigned the value of RMNG, the rate of maintenance respiration of the grain. When DVSR, the post-anthesis development stage, has a value between 0.4 and 0.7, LMR is reduced proportionally with the post-anthesis development stage.

```
GRGR = ARTTG - LMR
```

L3800: The variable GRGR, rate of increase in dry weight of the grain, is calculated as ARTTG (L3780) minus LMR (L3790).

```
RNTRS = RNTG*AVNRT/(AVN+REANOT(AVN))
```

L3810: The variable RNTRS, rate of translocation of nitrogen from roots to the grain, is calculated as RNTG, the rate of nitrogen accumulation in the grain, times the ratio of AVNRT, weight of nitrogen in the root available for translocation to the grain, and AVN, total weight of nitrogen in vegetative tissue available for translocation to the grain.

```
CRCANR = RNURT-RNLDR-RNTRS+PUSHD*ANRTI/DELT
```

L3820: The variable CRCANR, 'uncorrected' rate of change in weight of nitrogen in the roots is calculated as RNURT, rate of nitrogen uptake by the roots, minus RNLDR, rate of nitrogen loss from dying roots, minus RNTRS (L3810). At emergence, when PUSHD equals one, initial weight of nitrogen in the roots is added to CRCANR.

```
RCANRT = INSW(CRCANR,AMAX1(-ANRT/DELT,CRCANR),CRCANR)
```

L3830: The variable RCANRT, 'effective' rate of change in weight of nitrogen in the roots, is assigned the value of CRCANR, but its value is limited to ANRT, the weight of nitrogen in the roots, divided by DELT.

```
CRMR = INSW(FCHNX,AMIN1(ARESP/DELT-(ARTTG/(CEGR+REANOT(CEGR))
$       +TRFRS),-FCHNX),0.)
```

L3840: The variable CRMR, rate of consumption of reserves for maintenance respiration, assumes a value if FCHNX, the net flow of carbohydrates is negative. It can not exceed ARESP, the dry weight of reserve carbohydrates divided by DELT, minus the dry weight of reserve carbohydrates translocated to the grain divided by CEGR, the efficiency of conversion of primary assimilation products in grain dry matter, plus TRFRS, the dry weight of reserve carbohydrates transferred to vegetative structures. The value of CRMR is restrained to minus FCHNX.

```
RCRES = FCHTRS-ARTTG/(CEGR+REANOT(CEGR))-TRFRS-CRMR
```

L3850: The variable RCRES, rate of change in dry weight of non-structural carbohydrates, is calculated as FCHTRS, the current flow of carbohydrates to reserves, minus ARTTG, the rate of transfer of reserve carbohydrates from vegetative tissue to the grain, divided by CEGR, the conversion efficiency of primary assimilation products in grain dry matter, minus TRFRS, the rate of translocation of reserves to vegetative structures, minus CRMR (L3840).

```
OTGW = WGR/(GRN+REANOT(GRN))*1.E6
```

L3860: The variable OTGW, individual grain dry weight in mg, is calculated as WGR, the dry weight of the grain, divided by GRN, the grain density, times 10^6.

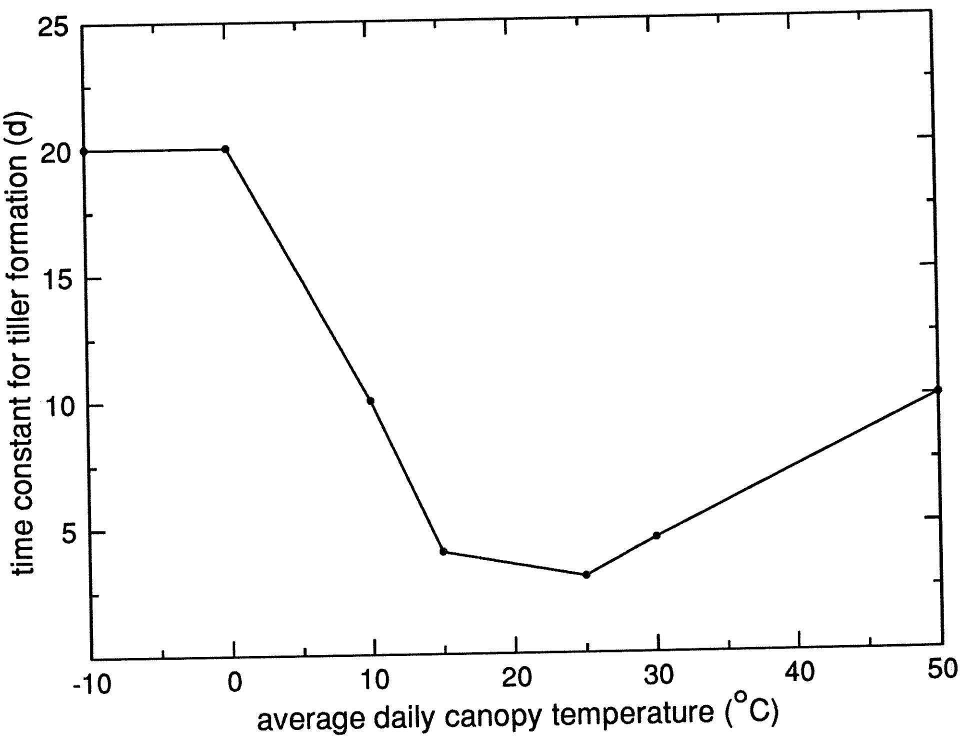

Figure 37 Function TCTFT, time constant for tiller formation as a function of canopy temperature.

```
TCTF = LINT(TCTFT,ITCTFT,TMPC)*LIMIT(0.3,1.,0.3+LAI*0.7)
```

L3870: The variable TCTF, time constant for tiller formation, is calculated by linear interpolation in the function TCTFT (Fig. 37) with TMPC , average daily canopy temperature as independent variable, times a reduction factor which increases proportionally from 0.3 to 1 when the leaf area index increases from zero to one.

```
CFTUDM = CHFTB*DVRV/(DVSPRE-DVSTS)
```

L3890: The variable CFTUDM, current carbohydrate requirement for tiller formation, is calculated as the product of the minimum carbohydrate requirement for completion of tiller formation, CHFTB, times DVRV, the development rate in the pre-anthesis phase, divided by the difference between DVSPRE and DVSTS, development stages at start and end of tillering, respectively.

```
TLNM = (FCHTLV+FCHST+FCHTRS)/(CFTUDM+REANOT(CFTUDM))
```

L3900: The variable TLNM, maximum number of tillers that can be maintained by current assimilate supply, is calculated as the sum of the current flow of carbohydrates to leaf blades, stem and sheaths and reserves, FCHTLV, FCHST and FCHTRS, respectively, divided by CFTUDM (L3890).

```
GRNT = (1.-DVSPRF)*AMAX1(0.,(TLNM-TLN)/TCTF*FNC)
```

L3910: The variable GRNT, rate of increase in tiller density, assumes the value zero, if the development stage is beyond end of tillering; else it is calculated as the difference between TLNM (L3900), the maximum number of tillers that can be maintained by current assimilate supply, and TLN, current tiller density, divided by TCTF, time constant for tiller formation, times FNC, the relative nitrogen concentration in leaf blades.

```
TLNIXX = TLNIX+REANOT(TLNIX)
```

L3920: The variable TLNIXX, an auxiliary variable used to avoid zero division assumes the value of variable TLNIX, an auxiliary variable to save the original value of TLNI.

```
RNLSDT = RWLSDT*LN
```

L3930: The variable RNLSDT, rate of nitrogen loss through stem and sheaths of dying tillers, is calculated as the product of RWLSDT, the rate of loss of dry weight of stem and sheaths through dying of non-reproductive tillers, and LN, the residual non-remobilizable concentration of nitrogen in stem and sheaths.

```
RNTSS = RNTG*AVNSTE/(AVN+REANOT(AVN))
```

L3940: The variable RNTSS, rate of translocation of nitrogen from stem and sheaths to the grain, is calculated as RNTG, the rate of nitrogen accumulation in the grain, times the ratio of AVNSTE, weight of nitrogen in stem and sheaths available for translocation to the grain and AVN, total weight of nitrogen in vegetative tissue available for translocation to the grain.

```
TRNLS = AMIN1(ANSTE/DELT,RNLDST+RNLSDT+RNTSS)
```

L3950: The variable TRNLS, rate of nitrogen loss from live stem and sheaths is calculated as the sum of RNLDST (L3650), rate of nitrogen loss through dying stem and sheaths, RNLSDT (L3930) and RNTSS (L3940).

```
ANGRI = PUSHN*GRGR*0.035
```

L3960: The variable ANGRI, initial weight of nitrogen in the grain on the first day of grain growth, is calculated as the product of the switch variable PUSHN, that assumes the value one on the day that grain fill starts, and is otherwise zero and GRGR, the rate of increase in dry weight of the grain, times 0.035, the initial N concentration in the grain.

```
CRCANS = RNUST-TRNLS+TNLST-ANGRI
```

L3970: The variable CRCANS, 'uncorrected' rate of change in weight of nitrogen in stem and sheaths is calculated as RNUST, the rate of nitrogen uptake by stem and sheaths minus TRNLS (L3950), the total rate of nitrogen loss from stem and sheaths, plus TNLST, the rate of nitrogen transfer from dying leaf blades to stem and sheaths minus ANGRI, the initial weight of nitrogen in the grain, which, rather arbitrarily, is assumed to be supplied from the stem.

```
RCANST = INSW(CRCANS,AMAX1(-ANSTE/DELT,CRCANS),CRCANS)
```

L3980: The variable RCANST, 'effective' rate of change in weight of nitrogen in stem and sheaths, is assigned the value of CRCANS (L3970); if CRCANS is negative, the value of RCANST is limited to minus ANSTE, the weight of nitrogen in stem and leaf sheaths, divided by DELT.

```
CFEUDM = CHFEB*DVRV/(DVSPRE-DVSSE)
```

L3990: The variable CFEUDM, current carbohydrate requirement for ear formation, is calculated as CHFEB, the minimum carbohydrate requirement for completion of ear formation, times DVRV, development rate in the pre-anthesis phase, divided by the difference between DVSPRE, the development stage in the pre-anthesis phase at end of tillering, and DVSSE, the development stage in the pre-anthesis phase at start of floral initiation.

```
ALTN = (FCHTLV+FCHST+FCHTRS)/(CFEUDM+REANOT(CFEUDM))
```

L4000: The variable ALTN, maximum number of ears that can be maintained by current carbohydrate supply, is calculated as the sum of the current carbohydrate flows to leaf blades, FCHTLV, stem and sheaths, FCHTST and reserves, FCHTRS divided by CFEUDM (L3990), the current carbohydrate requirement for ear formation.

```
MXNE = LIMIT(0.,TLN,ALTN)
```

L4010: The variable MXNE, maximum number of ears that can be formed is assigned the value of ALTN (L4000), the maximum number of ears that can be maintained by current carbohydrate supply. The value of MXNE is restrained to the value of TLN, the actual number of tillers, with a LIMIT function.

```
TCEF = EB+AMAX1(0.,(TLN/TLNIXX-1.)*(STCEF-EB))
```

L4020: The variable TCEF, time constant for ear formation, is calculated as the sum of EB, the basic time constant for ear formation and the product of tiller density, TLN, divided by initial tiller density, TLNIXX minus one, and the difference between STCEF, the maximum value of time constant for ear formation and EB.

```
PUSHE = REAAND(DVSSE-DVSVP,DVSV-DVSSE)
```

L4030: The variable PUSHE, a switch variable that assumes the value one on the day of start of floral initiation, and is otherwise zero, is calculated with a REAAND function in which DVSV, the development stage in the pre-anthesis phase, DVSSE, the development stage at start of floral initiation, and DVSVP, the development stage in the pre-anthesis phase one time interval ago, are arguments.

```
REARF1 = AMAX1(0.,(MXNE-EARN)/TCEF)*(1.-PUSHE)
```

L4040: The variable REARF1, potential rate of increase in ear number as determined by assimilate supply, is calculated as the difference between MXNE (L4010), the maximum number of ears that can be formed and EARN, the actual ear number divided by TCEF, the time constant for ear formation; on start of floral initiation REARF1 is set to zero.

```
REARF = INSW(DVSV-DVSSE,0.,(1.-DVSPRF)*REARF1)
```

L4050: The variable REARF, 'effective' rate of increase in ear number limited to appropriate phenological phase, is calculated if the development stage, DVSV exceeds DVSSE, the development stage at start of floral initiation. Variable REARF is then calculated as the product of one minus DVSPRF, a factor indicating whether development stage is beyond the end of tiller formation (1) or not (0) times REARF1 (L4040), the potential rate of increase in ear number.

```
CFSUDM = CHFSB*DVRV/(DVSSPE-DVSSPS)
```

L4060: The variable CFSUDM, current carbohydrate requirement for spikelet formation, is calculated as CHFSB, the minimum carbohydrate requirement for completion of spikelet formation times DVRV, the development rate in the pre-anthesis phase, divided by the difference in development stage between terminal spikelet formation, DVSSPE, and start of spikelet differentiation, DVSSPS.

```
ALSN = (FCHTLV+FCHST+FCHTRS)/(CFSUDM+REANOT(CFSUDM))
```

L4070: The variable ALSN, maximum number of spikelets that can be maintained by current carbohydrate supply, is calculated as the sum of the current flow of carbohydrates to leaf blades, stem and sheaths, and reserves, FCHTLV, FCHST and FCHTRS, respectively, divided by CFSUDM, current carbohydrate requirement for spikelet formation.

```
MXNSP = LIMIT(0.,EARN*25.,ALSN)
```

L4080: The variable MXNSP, maximum number of spikelets that can be formed assumes the value of the variable ALSN, restricted to an upper limit of 25 spikelets per ear, with a LIMIT function.

```
TCSF = SB+AMAX1(0.,(EARN/TLNIXX-1.)*(STCSF-SB))
```

L4090: The variable TCSF, time constant for spikelet formation, is calculated as the sum of SB, the basic time constant for spikelet formation plus the ratio of EARN, total ear number and TLNIXX an auxiliary variable which stores the initial tiller density, minus 1, times the difference between STCSF, the maximum value of the time constant for spikelet formation and SB.

```
RSPLF1 = AMAX1(0.,(MXNSP-NSPS)/TCSF)
```

L4100: The variable RSPLF1, 'effective' rate of spikelet formation as determined by assimilate availability, is calculated as the difference between MXNSP, the maximum number of spikelets that can be formed, minus NSPS, the total number of spikelets, divided by TCSF, the time constant for spikelet formation.

```
RSPLF = INSW(DVSV-DVSSPS,0.,INSW(DVSSPE-DVSV,0.,RSPLF1))
```

L4110: The variable RSPLF, rate of spikelet formation limited to the appropriate phenological phase is assigned the value of RSPLF1, if the development stage DVSV is between DVSSPS, the development stage at start of spikelet differentiation and DVSSPE, development stage at terminal spikelet formation, else RSPLF is zero.

```
CFFUDM = CHFFB*DVRV/(DVSFE-DVSFS)
```

L4120: The variable CFFUDM, current carbohydrate requirement for floret formation, is calculated as the product of CHFFB, the minimum carbohydrate requirement for completion of fertile floret formation, times DVRV, the development rate in the pre-anthesis phase, divided by the difference between development stages at end and at start of floret formation, DVSFE and DVSFS, respectively.

```
ALFN = (FCHTLV+FCHST+FCHTRS)/(CFFUDM+REANOT(CFFUDM))
```

L4130: The variable ALFN, maximum number of fertile florets that can be maintained by current carbohydrate supply, is calculated as the sum of the current flow of carbohydrates to leaf blades, stem and sheaths and reserves, divided by CFFUDM (L4120), the current carbohydrate requirement for floret formation.

```
MXNFFL = LIMIT(0.,NSPS*4,ALFN)
```

L4140: The variable MXNFFL, maximum number of fertile florets that can be formed, assumes the value of variable ALFN (L4130), restricted to four florets per spikelet with a LIMIT function.

```
TCFF = FB+AMAX1(0.,(NSPS/TLNIXX-5.)*(STCFF-FB)/20.)
```

L4150: The variable TCFF, time constant for fertile floret formation, is calculated from FB, the basic time constant for floret formation, the ratio of NSPS, total number of spikelets, and TLNIXX, initial tiller density, and times the difference of STCFF, maximum value of the time constant for floret formation and FB.

```
RFFF1 = AMAX1(0.,(MXNFFL-FFNR)/TCFF)
```

L4160: The variable RFFF1, rate of fertile floret formation as determined by assimilate supply, is calculated as the difference between MXNFFL, the maximum number of fertile florets that can be formed and FFNR, current number of fertile florets, divided by TCFF (L4150), the time constant for fertile floret formation.

```
RFFF = INSW(DVSV-DVSFS,0.,INSW(DVSFE-DVSV,0.,RFFF1))
```

L4170: The variable RFFF, rate of formation of fertile florets limited to appropriate phenological phase, is assigned the value of variable RFFF1 if DVSV, the development stage in the pre-anthesis phase, is between DVSFS, development stage at the start of floret formation, and DVSFE, development stage at the end of floret formation, else RFFF is zero.

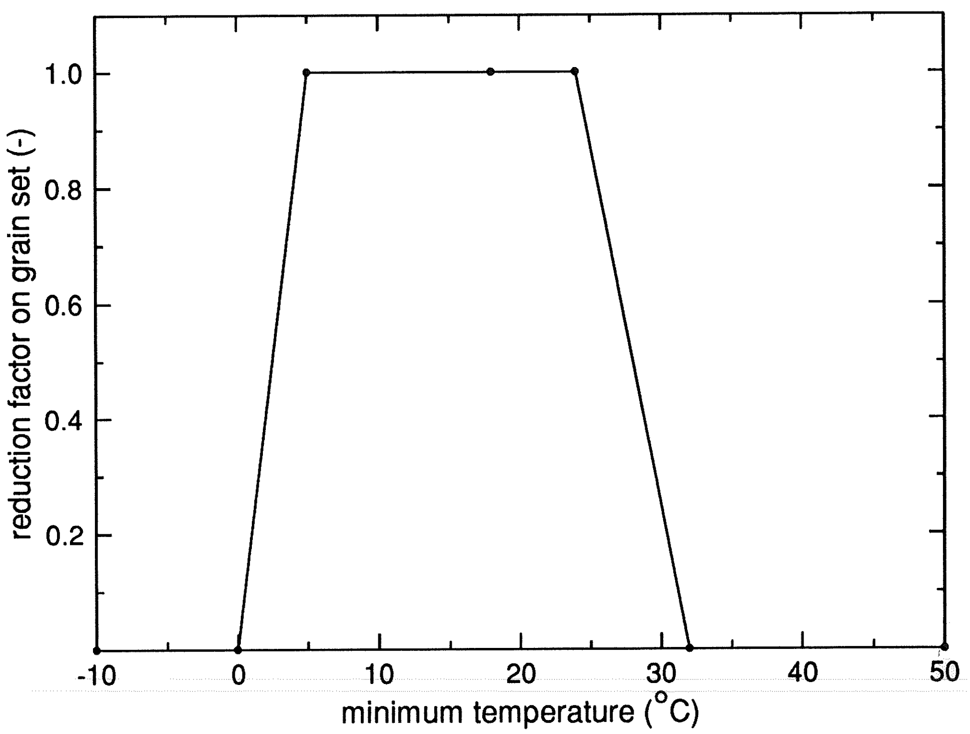

Figure 38 **Function RFTT, reduction factor on grain set as function of minimum temperature.**

```
RFT = LINT(RFTT,IRFTT,TMIN)
```

L4180: The variable RFT, effect of minimum temperature on grain set, is calculated by linear interpolation in the function RFTT (Fig. 38) with the minimum temperature, TMIN, as independent variable.

```
TCGF = GB+AMAX1(0.,(FFNR/TLNIXX-10.)*(STCGF-GB)/90.)
```

L4190: The variable TCGF, time constant for grain formation, is calculated from GB, the basic time constant for grain formation, and the ratio of FFNR, number of fertile florets, and TLNIXX , auxiliary variable that stores initial tiller density times the difference of STCGF, the maximum value of time constant for grain formation and GB.

```
RGRN = INSW(DVSR-DVSGS,0.,(1.-SGFF)*(FFNR-GRN)/TCGF*RFT)
```

L4200: The variable RGRN, rate of increase in grain density, is calculated as FFNR, the number of fertile florets, minus GRN, current grain density, divided by TCGF (L4190), time constant for grain formation multiplied by RFT, a factor which accounts for the effect of temperature on grain set. If DVSR, the development stage in the post-anthesis phase, is below DVSGS, development stage in the post-anthesis phase at start of grain set, or SGFF, a switch variable indicating whether development stage is beyond start of grain fill equals 1, RGRN is set to zero.

```
PUSHT = REAAND(DVSV-DVSPRE,DVSPRE-DVSVP)
```

L4210: The variable PUSHT, a switch variable that assumes the value one on the day that tillering ceases, and is otherwise zero is calculated from DVSV, the development stage in the pre-anthesis phase, DVSPRE, development stage at end of tillering and DVSVP, development stage in the pre-anthesis phase one time interval ago.

```
CRCLFA = GRLVS*FLFAR-RDLFA+PUSHD*ARLFI/DELT
```

L4220: The variable CRCLFA, 'uncorrected' rate of change in leaf area, is calculated as the product of GRLVS, the rate of increase in dry weight of the leaf blades and FLFAR, the current specific leaf area, minus RDLFA, the rate of decline in leaf area due to senescence. At emergence, when PUSHD equals one, initial green area of leaf blades, ARLFI, divided by DELT, is added to CRCLFA.

```
RCLFA = INSW(CRCLFA,AMAX1(-ARLF/DELT,CRCLFA),CRCLFA)
```

L4230: The variable RCLFA, current rate of change in green leaf area is assigned the value of the variable CRCLFA (L4220). When CRCLFA is negative, its value is limited to the green area of the leaf blades present, ARLF, divided by DELT with an AMAX1 function.

```
SLWA = (WLVS+ARESP*WLVS/(TVEGM+REANOT(TVEGM)))/(ARLF+REANOT(ARLF))
```

L4240: The variable SLWA, specific leaf dry weight, is calculated from WLVS, the dry weight of live leaf blades, plus ARESP, the dry weight of the reserve carbohydrates, times the ratio WLVS over TVEGM, the total dry weight of vegetative crop organs. This sum is divided by ARLF, the green area of the leaf blades.

```
R1 = DATE*REAAND(DVSR-DVSEGF,DVSEGF-DVSRP)
```

L4250: The variable R1, rate variable used in the integration of the state variable EGFDAY, is calculated from the day number of simulation, DATE, development stage in the post-anthesis phase, DVSR, development stage at end of grain fill, DVSEGF and DVSRP, development stage one time interval ago.

```
R2 = GRNT-DRNT+PUSHD*TLNI/DELT
```

L4260: The variable R2, rate variable used in the integration of the state variable TLN, is calculated as the rate of increase in tiller density, GRNT minus the death rate of non-reproductive tillers, DRNT. The initial tiller density, TLNI divided by DELT is added at emergence.

```
R3 = PUSHT*(TLN-EARN)*0.01+DRNT
```

L4270: The variable R3, an intermediate variable used in the integration of the state variable DTLN, is calculated as the product of the variable PUSHT (L4210) and the difference between current tiller density, TLN, and total number of ears, EARN, times 0.01 plus DRNT, death rate of non-reproductive tillers.

```
R4 = PUSHT*(TLN-EARN)
```

L4280: The variable R4, an intermediate variable used in the integration of the state variable TNNR, is calculated as the product of PUSHT (L4210) and the difference between tiller density, TLN, and total number of ears, EARN.

```
R5 = PUSHT*WLVS/(TLN+REANOT(TLN))
```

L4280: The variable R5, an intermediate variable, used in the integration of the state variable LWDTL, is calculated as the product of PUSHT (L4210) and the dry weight of live leaf blades, WLVS, divided by the current tiller density, TLN.

```
R6 = PUSHT*WSTEM/(TLN+REANOT(TLN))
```

L4280: The variable R6, an intermediate variable used in the integration of the state variable SWDTL, is calculated as the product of PUSHT (L4210) and the dry weight of live stem and sheaths, WSTEM, divided by current tiller density, TLN.

```
GTW = RCWLVS+RCWST+RCWRT+GRGR
```

L4310: The variable GTW, increase in total dry weight is calculated as the sum of the current rate of change in dry weight of live leaf blades, RCWLVS, the current rate of change in dry weight of stem and sheaths, RCWST, the current rate of change in dry weight of the roots, RCWRT and the rate of increase in dry weight of the grain, GRGR.

```
IF (OUTPUT .OR. TERMNL) THEN
   CALL OUTDAT (2, 0, 'DVS',     DVS)
   CALL OUTDAT (2, 0, 'PUSHD',   PUSHD)
   CALL OUTDAT (2, 0, 'TADRW',   TADRW)
   CALL OUTDAT (2, 0, 'WLVS',    WLVS)
   CALL OUTDAT (2, 0, 'ARLF',    ARLF)
   CALL OUTDAT (2, 0, 'SLWA',    SLWA)
   CALL OUTDAT (2, 0, 'GRAI',    GRAI)
   CALL OUTDAT (2, 0, 'WRT',     WRT)
   CALL OUTDAT (2, 0, 'RTD',     RTD)
   CALL OUTDAT (2, 0, 'WSTEM',   WSTEM)
   CALL OUTDAT (2, 0, 'WGR',     WGR)
   CALL OUTDAT (2, 0, 'GRN',     GRN)
   CALL OUTDAT (2, 0, 'OTGW',    OTGW)
   CALL OUTDAT (2, 0, 'FFNR',    FFNR)
   CALL OUTDAT (2, 0, 'NSPS',    NSPS)
   CALL OUTDAT (2, 0, 'EARN',    EARN)
   CALL OUTDAT (2, 0, 'TLN',     TLN)
   CALL OUTDAT (2, 0, 'ARESP',   ARESP)
   CALL OUTDAT (2, 0, 'RESL',    RESL)
   CALL OUTDAT (2, 0, 'TRFRS',   TRFRS)
   CALL OUTDAT (2, 0, 'ARTTG',   ARTTG)
   CALL OUTDAT (2, 0, 'FCHN',    FCHN)
   CALL OUTDAT (2, 0, 'AMAXN',   AMAXN)
   CALL OUTDAT (2, 0, 'DTGA',    DTGA)
   CALL OUTDAT (2, 0, 'DGAS',    DGAS)
   CALL OUTDAT (2, 0, 'APTRAN', APTRAN)
   CALL OUTDAT (2, 0, 'TRAN',    TRAN)
   CALL OUTDAT (2, 0, 'TOTRAN', TOTRAN)
   CALL OUTDAT (2, 0, 'EVTOT',   EVTOT)
   CALL OUTDAT (2, 0, 'TEVAP',   TEVAP)
   CALL OUTDAT (2, 0, 'WTOT',    WTOT)
   CALL OUTDAT (2, 0, 'TRAIN',   TRAIN)
   CALL OUTDAT (2, 0, 'INFR',    INFR)
   CALL OUTDAT (2, 0, 'PEVAP',   PEVAP)
   CALL OUTDAT (2, 0, 'PEVAP1', PEVAP1)
   CALL OUTDAT (2, 0, 'EVAP',    EVAP)
   CALL OUTDAT (2, 0, 'TDRAIN', TDRAIN)
   CALL OUTDAT (2, 0, 'NTOT',    NTOT)
   CALL OUTDAT (2, 0, 'BIOMNT', BIOMNT)
   CALL OUTDAT (2, 0, 'TNDEM',   TNDEM)
   CALL OUTDAT (2, 0, 'RNUV',    RNUV)
   CALL OUTDAT (2, 0, 'TNABM',   TNABM)
```

```
      CALL OUTDAT (2, 0, 'ANLV',    ANLV)
      CALL OUTDAT (2, 0, 'ANGR',    ANGR)
   END IF
```

L4320: Output section, when either of the logical variables OUTPUT or TERMNL are set to TRUE the current values of the variables listed are saved to a temporary output file. One can modify the number of variables or rearrange the order to adapt the model's output file to one's own needs.

5.5. Integration section of module

```
*-----integration section

   PUSHDI = INTGRL(PUSHDI,PUSHD,DELT)
```

L4330: The state variable PUSHDI, which monitors the end of germination, is integrated over the timestep DELT using PUSHD as rate variable.

```
   TRAIN = INTGRL(TRAIN,RAIN,DELT)
```

L4340: The state variable TRAIN, total seasonal rainfall, is integrated over the timestep DELT using RAIN, rain intensity, as rate variable.

```
   TEVAP = INTGRL(TEVAP,EVTOT,DELT)
```

L4350: The state variable TEVAP, total seasonal soil surface evaporation, is integrated over the timestep DELT using EVTOT, current rate of soil surface evaporation, as rate variable.

```
   TOTINF = INTGRL(TOTINF,INFR,DELT)
```

L4360: The state variable TOTINF, total seasonal infiltration, is integrated over the timestep DELT using INFR, rate of infiltration, as rate variable.

```
   TDRAIN = INTGRL(TDRAIN,RDRAIN,DELT)
```

L4370: The state variable TDRAIN, total loss of soil moisture by drainage beyond potential rooting zone, is integrated over the timestep DELT using RDRAIN, rate of drainage, as rate variable.

```
   TNTRMT = INTGRL(TNTRMT,APFERT,DELT)
```

L4380: The state variable TNTRMT, total nitrogen fertilizer application, is integrated over the timestep DELT using APFERT, fertilizer application rate, as rate variable.

```
   RANO3 = INTGRL(RANO3,(1.-RANO3)/TCN,DELT)
```

L4390: The state variable RANO3, proportion of nitrate in total inorganic nitrogen store in top soil compartment, is integrated over the timestep DELT using one minus the value of RANO3 divided by the time constant for nitrification, TCN, as rate variable.

```
   WATER(I) = INTGRL(WATER(I),DWAT(I),DELT)
```

L4410: The state variable WATER(I), amount of soil moisture in soil compartment I, is integrated over the timestep DELT using DWAT(I), rate of change in amount of soil moisture in compartment I, as rate variable.

```
ASLT(I) = INTGRL(ASLT(I),DASLT(I),DELT)
```

L4420: The state variable ASLT(I), weight of mineral nitrogen in soil compartment I, is integrated over the timestep DELT using DASLT(I), rate of change in weight of mineral nitrogen in soil compartment I, as rate variable.

```
FOM(I) = INTGRL(FOM(I),DFOM(I),DELT)
```

L4430: The state variable FOM(I), weight of fresh organic material in soil compartment I, is integrated over the timestep DELT using DFOM(I), rate of change in weight of fresh organic material in soil compartment I, as rate variable.

```
FON(I) = INTGRL(FON(I),DFON(I),DELT)
```

L4440: The state variable FON(I), weight of nitrogen in fresh organic material in soil compartment I, is integrated over the timestep DELT using DFON(I), rate of change in weight of nitrogen in fresh organic material in soil compartment I, as rate variable.

```
HUM(I) = INTGRL(HUM(I),DHUM(I),DELT)
```

L4450: The state variable HUM(I), weight of stable organic material in soil compartment I, is integrated over the timestep DELT using DHUM(I), rate of change in weight of stable organic material in soil compartment I, as rate variable.

```
NHUM(I) = INTGRL(NHUM(I),DNHUM(I),DELT)
```

L4460: The state variable NHUM(I), weight of nitrogen in stable organic material in soil compartment I, is integrated over the timestep DELT using DNHUM(I), rate of change in weight of nitrogen in stable organic material in soil compartment I, as rate variable.

```
BIOMN(I) = INTGRL(BIOMN(I),DBIOMN(I),DELT)
```

L4470: The state variable BIOMN(I), weight of nitrogen in microbial biomass in soil compartment I, is integrated over the timestep DELT using DBIOMN(I), rate of change in weight of nitrogen in microbial biomass in soil compartment I, as rate variable.

```
WTOT = 0.
NTOT = 0.
TFON = 0.
TFOM = 0.
NHUMT = 0.
HUMT = 0.
BIOMNT = 0.
```

L4480: The state variables WTOT, NTOT, TFON, TFOM, NHUMT, HUMT, BIOMNT, representing accumulated totals over the potential rooting zone, are initialized at zero.

```
DO 350 I = 1,ILAYER
   IF (TDT(I)+0.5.GT.MXRTD) GO TO 350
WTOT = WTOT+WATER(I)
```

L4490: The state variable WTOT, total amount of soil moisture in the potential rooting zone is calculated as the sum of the amounts of soil moisture in soil compartment I. If the top of the soil compartment is below the potential rooting depth, summation is skipped.

```
NTOT = NTOT+ASLT(I)
```

L4500: The state variable NTOT, total weight of nitrogen in microbial biomass in the potential rooting zone, is calculated as the sum of the weights in soil compartment I.

```
TFON = TFON+FON(I)
```

L4510: The state variable TFON, total weight of nitrogen in fresh organic material in the potential rooting zone, is calculated as the sum of the weights in soil compartment I.

```
NHUMT = NHUMT+NHUM(I)
```

L4520: The state variable NHUMT, total weight of nitrogen in stable organic material in the potential rooting zone, is calculated as the sum of the weights in soil compartment I.

```
TFOM = TFOM+FOM(I)
```

L4530: The state variable TFOM, total weight of fresh organic material in the potential rooting zone, is calculated as the sum of the weights in soil compartment I.

```
HUMT = HUMT+HUM(I)
```

L4540: The state variable HUMT, total weight of stable organic material in the potential rooting zone, is calculated as the sum of the weights in soil compartment I.

```
BIOMNT = BIOMNT+BIOMN(I)
```

L4550: The state variable BIOMNT, total weight of nitrogen in microbial biomass in the potential rooting zone, is calculated as the sum of the weights in soil compartment I.

```
TS = INTGRL(TS,0.1*(TMPC-DTMPA)/DELT,DELT)
```

L4560: The state variable TS, soil temperature, is integrated over the timestep DELT using the product of 0.1 times the difference between TMPC, canopy temperature and DTMPA, average daily air temperature divided by DELT as rate variable.

```
TOTRAN = INTGRL(TOTRAN,TRAN,DELT)
```

L4570: The state variable TOTRAN, total seasonal crop transpiration, is integrated over the timestep DELT using the rate of canopy transpiration as dictated by actual soil moisture conditions, TRAN, as rate variable.

```
ANTHES = INTGRL(ANTHES,DATE*REAAND(DVS-0.5,0.5-DVSP)/DELT,DELT)
```

L4580: The state variable ANTHES, anthesis date, is integrated over the timestep DELT using the variable DATE, day number, at the moment that the current overall development stage, DVS exceeds 0.5 and the development stage one time interval ago, DVSP is less than 0.5.

```
DVSV = INTGRL(DVSV,DVRV+PUSHD*DVSI/DELT,DELT)
```

L4590: The state variable DVSV, development stage in the pre-anthesis phase, is integrated over the timestep DELT using DVRV, development rate in the pre-anthesis phase, as rate variable. At emergence, when PUSHD equals one, DVSI, development stage in the pre-anthesis phase at emergence, is added.

```
DVSR = INTGRL(DVSR,DVRR,DELT)
```

L4600: The state variable DVSR, development stage in the post-anthesis phase, is integrated over the timestep DELT using DVSR, development rate in the post-anthesis phase, as rate variable.

```
DVS = 0.5*(DVSV+DVSR)
```

L4610: The state variable DVS, overall development stage, is calculated as 0.5 times the sum of the state variables DVSV (L4590) and DVSR (L4600).

```
LEAFW(I) = INTGRL(LEAFW(I),DLEAF(I),DELT)
```

L4620: The state variable LEAFW(I), leaf dry weight in class I, is integrated over the timestep DELT using DLEAF(I), rate of change of leaf dry weight in class I, as rate variable.

```
ALFT(I) = INTGRL(ALFT(I),DALFT(I),DELT)
```

L4630: The state variable ALFT(I) , accumulated temperature sum in leaf class I, is integrated over the timestep DELT using DALFT(I), rate of change in cumulative temperature sum in leaf class I, as rate variable.

```
LEAFA(I) = INTGRL(LEAFA(I),DLA(I),DELT)
```

L4640: The state variable LEAFA(I), leaf area in leaf class I, is integrated over the timestep DELT using DLA(I), specific leaf area of leaf blades, as rate variable.

```
WSTEM = INTGRL(WSTEM,RCWST,DELT)
```

L4650: The state variable WSTEM, dry weight of live stem and sheaths, is integrated over the timestep DELT using RCWST, 'effective' rate of change in dry weight of stem and sheaths, as rate variable.

```
WRT = INTGRL(WRT,RCWRT,DELT)
```

L4660: The state variable WRT, dry weight of the roots, is integrated over the timestep DELT using RCWRT, 'effective' rate of change in dry weight of the roots, as rate variable.

```
RTD = INTGRL(RTD,RGRRL+PUSHD*IRTD/DELT,DELT)
```

L4670: The state variable RTD, rooting depth, is integrated over the timestep DELT using RGRRL, rate of increase in rooting depth as rate variable. At emergence, when PUSHD equals one, IRTD, initial rooting depth, is added.

```
ARESP = INTGRL(ARESP,RCRES+PUSHD*ARESPI/DELT,DELT)
```

L4680: The state variable ARESP, dry weight of reserve carbohydrates, is integrated over the timestep DELT using RCRES, rate of change in dry weight of non-structural carbohydrates, as rate variable. At emergence, when PUSHD equals one, ARESPI, initial dry weight of reserve carbohydrates, is added.

```
WGR = INTGRL(WGR,GRGR,DELT)
```

L4690: The state variable WGR, dry weight of the grain, is integrated over the timestep DELT using GRGR, rate of increase in dry weight of the grain, as rate variable.

```
EGFDAY = INTGRL(EGFDAY,R1,DELT)
```

L4700: The state variable EGFDAY, day number at end of grain filling, is integrated over the timestep DELT using R1 as intermediate rate variable.

```
WLVS = INTGRL(WLVS,RCWLVS,DELT)
```

L4710: The state variable WLVS, dry weight of live leaf blades, is integrated over the timestep DELT using RCWLVS, current rate of change in live leaf blade weight, as rate variable.

```
TLN = INTGRL(TLN,R2,DELT)
```

L4720: The state variable TLN, tiller density, is integrated over the timestep DELT using R2 as intermediate rate variable.

```
DTLN = INTGRL(DTLN,R3,DELT)
```

L4730: The state variable DTLN, total number of dead tillers, is integrated over the timestep DELT using R3 as intermediate rate variable.

```
AWTL = (WLVS+WSTEM)/(TLN+REANOT(TLN))
```

L4740: The state variable AWTL, average dry weight per tiller, is calculated as the sum of the dry weights of live leaf blades and live stem and sheaths, divided by TLN (L4720).

```
TNNR = INTGRL(TNNR,R4,DELT)
```

L4750: The state variable TNNR, total number of non-reproductive tillers, is integrated over the timestep DELT using R4 as intermediate rate variable.

```
LWDTL = INTGRL(LWDTL,R5,DELT)
```

L4760: The state variable LWDTL, average dry weight of leaf blades of non-reproductive tillers, at end of ear formation, is integrated over the timestep DELT using R5 as intermediate rate variable.

```
SWDTL = INTGRL(SWDTL,R6,DELT)
```

L4770: The state variable SWDTL, final dry weight of stem and sheaths of non-reproductive tillers at end of ear formation, is integrated over the timestep DELT using R6 as intermediate rate variable.

```
EARN = INTGRL(EARN,REARF+PUSHE*TLNI,DELT)
```

L4780: The state variable EARN, total number of ears, is integrated over the timestep DELT using REARF, rate of increase in ear number, limited to the appropriate phenological phase, as rate variable. At floral initiation, when PUSHE equals one, TLNI, initial tiller density, is added.

```
NSPS = INTGRL(NSPS,RSPLF,DELT)
```

L4790: The state variable NSPS, total number of spikelets, is integrated over the timestep DELT using RSPLF, current rate of spikelet formation, limited to the appropriate phenological phase, as rate variable.

```
FFNR = INTGRL(FFNR,RFFF,DELT)
```

L4800: The state variable FFNR, number of fertile florets, is integrated over the timestep DELT using RFFF, current rate of fertile floret formation, limited to the appropriate phenological phase, as rate variable.

```
GRN = INTGRL(GRN,RGRN,DELT)
```

L4810: The state variable GRN, grain density, is integrated over the timestep DELT using RGRN, rate of increase in grain density, as rate variable.

```
ARLFE = INTGRL(ARLFE,GRLVS*FLFAR+PUSHD*ARLFI/DELT,DELT)
```

L4820: The state variable ARLFE, total area of green and senesced leaf blades, is integrated over the timestep DELT using the product of rate of increase in dry weight of the leaf blades, GRLVS and the current specific leaf area, FLFAR as rate variable. At emergence, when PUSHD equals one, ARLFI, initial green area of leaf blades, is added.

```
ARLF = INTGRL(ARLF,RCLFA,DELT)
```

L4830: The state variable ARLF, green area of live leaf blades, is integrated over the timestep DELT using RCLFA, effective rate of change in green leaf area, as rate variable.

```
LAI = 1.E-4*ARLF
```

L4840: The state variable LAI, leaf area index in $m^2\ m^{-2}$, is calculated as the value of ARLF times 10^{-4}.

```
ANLV = INTGRL(ANLV,RCANLV,DELT)
```

L4850: The state variable ANLV, weight of nitrogen in live leaf blades, is integrated over the timestep DELT using RCANLV, effective rate of change in weight of nitrogen in leaf blades, as rate variable.

```
ANSTE = INTGRL(ANSTE,RCANST,DELT)
```

L4860: The state variable ANSTE, weight of nitrogen in live stem and leaf sheaths, is integrated over the timestep DELT using RCANST, effective rate of change in weight of nitrogen in stem and sheaths, as rate variable.

```
ANRT = INTGRL(ANRT,RCANRT,DELT)
```

L4870: The state variable ANRT, weight of nitrogen in the roots, is integrated over the timestep DELT using RCANRT, effective rate of change in weight of nitrogen in the roots, as rate variable.

```
ANGR = INTGRL(ANGR,RNTG+ANGRI,DELT)
```

L4880: The state variable ANGR, weight of nitrogen in the grain, is integrated over the timestep DELT using the sum of RNTG, rate of nitrogen accumulation in the grain and ANGRI, initial weight of nitrogen in the grain, as rate variable.

```
TVEGM = WLVS+WSTEM+WRT
```

L4890: The state variable TVEGM, total dry weight of vegetative crop organs, is calculated as the sum of the dry weight of live leaf blades, live stem and sheaths and roots, WLVS, WSTEM and WRT, respectively.

```
NDSTR = INTGRL(NDSTR,RNL-TNLST+RNLDST+RNLSDT,DELT)
```

L4900: The state variable NDSTR, weight of nitrogen in dead above-ground material is integrated over the timestep DELT using the sum of RNL, rate of loss of nitrogen from dying leaf blades, TNLST, rate of nitrogen transfer from dying leaf blades to stem and sheaths, (negative) RNLDST, rate of nitrogen loss from dying stem and sheaths and RNLSDT, rate of nitrogen loss from stem and sheaths of dying tillers, as rate variable.

```
DSTR = INTGRL(DSTR,DRLVS+DRSTE+RWLSDT,DELT)
```

L4910: The state variable DSTR, total dry weight of above-ground dead material is integrated over the timestep DELT using the sum of DRLVS, effective death rate of leaf blades, DRSTE, death rate of stem and sheaths and RWLSDT, rate of loss of dry weight of stem and sheaths through dying of stem and sheaths of non-reproductive tillers, as rate variable.

```
SRR = (WSTEM+WLVS+WGR+DSTR)/(WRT+REANOT(WRT))
```

L4920: The state variable SRR, shoot root ratio is calculated as the sum of the dry weight of live stem and sheaths, leaf blades, roots and the total dry weight of above-ground dead material, WSTEM, WLVS, WGR, DSTR, respectively, divided by WRT, dry weight of live roots.

```
TNLCH = INTGRL(TNLCH,SLTFD,DELT)
```

L4930: The state variable TNLCH, total nitrogen loss by leaching, is integrated over the timestep DELT using SLTFD, rate of leaching of nitrogen beyond potential rooting zone, as rate variable.

```
INIV = INTGRL(INIV,PUSHD*(ANLVI+ANRTI)/DELT,DELT)
```

L4940: The state variable INIV, initial weight of nitrogen in the vegetation, is integrated over the timestep DELT using the sum of the initial weight of nitrogen in the leaf blades, ANLVI and the initial weight of nitrogen in the roots, ANRTI divided by DELT, as rate variable.

```
TNVOL = INTGRL(TNVOL,RVOLA,DELT)
```

L4950: The state variable TNVOL, total nitrogen loss by volatilization, is integrated over the timestep DELT using RVOLA, the rate of ammonia volatilization from the top soil compartment, as rate variable.

```
TNABM = ANLV+ANSTE+ANGR+NDSTR
```

L4960: The state variable TNABM, total weight of nitrogen in above-ground material, is calculated as the sum of the weights of nitrogen in leaf blades, stem and sheaths, grain and above-ground dead material, ANLV, ANSTE, ANGR and NDSTR, respectively.

```
TADRW = WLVS+WSTEM+WGR+DSTR+ARESP*(1.-WRT/(TVEGM+REANOT(TVEGM)))
```

L4970: The state variable TADRW, total above-ground dry weight is calculated as the sum of the dry weights of leaf blades, stem and sheaths, grain and above-ground dead material, WLVS, WSTEM, WGR and DSTR, respectively plus the dry weight of reserve carbohydrates in above-ground material which is calculated as the product of ARESP and the fraction of total dry matter in the shoot, calculated as one minus the ratio of dry weight of the roots and TVEGM, total dry weight of vegetative crop organs.

```
CARBAL = FCHN-FCHTR-FCHTLV-FCHST-ARTTG /
$         (CEGR+REANOT(CEGR))-CRMR-RCRES
```

L4980: The state variable CARBAL, carbon balance, which should be zero throughout the simulation, is calculated as the net flow of carbohydrate, FCHN minus the carbohydrate flows to the following sinks: roots, FCHTR, leaf blades, FCHTLV, stem and sheaths, FCHST, transfer of reserve carbohydrate from vegetative tissue to the grain, ARTTG divided by the conversion efficiency of primary assimilation products in grain dry matter, CEGR, loss through maintenance respiration, CRMR and loss through change in dry weight of non-structural carbohydrates, RCRES.

```
NBAL = TNINT+TNTRMT+INIV+TOTINF*0.02+FRNF*FOMI+NHUMI+
```

```
      $       BIOMNI-BIOMNT-NTOT-TFON-TNVOL-NHUMT-ANLV-ANSTE-ANRT-
      $       ANGR-NDSTR-TNLCH
```

L4990: The state variable NBAL, nitrogen balance, which should be zero throughout the simulation, is calculated as the sum of the total weight of inorganic nitrogen in the soil profile, TNINT, total nitrogen fertilizer application, TNTRMT, initial weight of nitrogen in the vegetation at emergence, INIV, total seasonal infiltration, TOTINF times 0.02, product of the concentration of nitrogen in fresh organic material, FRNF and the initial weight of fresh organic material in the soil profile, and FOMI, initial total weight of nitrogen in stable organic material in the soil profile, NHUMI and the initial weight of nitrogen in microbial biomass in the soil profile at the start of simulation, BIOMNI. The terms subtracted are, the total weights of nitrogen in microbial biomass in the soil profile, BIOMNT, inorganic nitrogen in the soil profile, NTOT, nitrogen in fresh organic material, TFON, total nitrogen loss through volatilization, TNVOL, nitrogen in stable organic material, NHUMT, in leaf blades, ANLV, in stem and sheaths, ANSTE, in roots, ANRT, in grain, ANGR, in dead-above ground material, NDSTR and the total nitrogen loss by leaching, TNLCH.

```
      IF (DVS.GE.1.) TERMNL = .TRUE.
```

L5000: The logical variable TERMNL, a flag that indicates whether simulation should terminate is set to the value TRUE, if DVS, overall development stage exceeds or equals one.

6. Compilation and linking of the model

When the source-code of the spring-wheat model has been changed, the model should be compiled and linked before execution. When model parameters, functions or initial values of state variables have to be changed, the relevant variables in the datafiles can be adapted, and the model executed directly.

The diskette contains four FORTRAN-77 source files:

- MAIN.FOR; Fortran Simulation Environment (FSE); supervisor of the simulation algorithm,
- SWHEAT.FOR; spring-wheat model; phenological development, carbon, nitrogen and soil moisture balance,
- REST.FOR; FORTRAN equivalents of CSMP-functions, subroutines for calculation of daily total gross assimilation,
- WEATHER.FOR; CABO/TPE weather system.

Compilation and linking will be explained for both personal computers and µVAX computers.

6.1. Compilation and linking on personal computers

The commands in this paragraph are suited for use in combination with the Microsoft FORTRAN compiler version 5.0 (Microsoft, 1989). It is assumed that the compiler exists on the hard-disk in a subdirectory that can be found within the PATH statement. All the files on the diskette should be copied into a directory on the hard-disk, e.g. the directory C:\MODEL. Compilation of the source file of the subroutine SWHEAT can then be started by typing:

```
FL /FPc /Gt /4Nb /Os /AH /Zi /c SWHEAT.FOR <return>
```

This command invokes the compiler to compile the file SWHEAT.FOR, which contains the source code of the spring wheat model; upper and lower case characters are significant within this command. The file SWHEAT.FOR can only be compiled with the options /4Nb /Os because of its size, that will instruct the compiler to minimize the size of the object code and to disable extended error handling at runtime. The object file produced by the compiler is suitable for use on personal computers with and without mathematical coprocessor.

After successful compilation, the object-files produced can be linked to an executable file (MAIN.EXE). The object library file TTUTIL.LIB should exist in the directory where the other FORTRAN library, LLIBFORE.LIB exists. The Microsoft linker can be invoked with the command:

```
LINK /se:1024 MAIN+SWHEAT+REST+WEATHER,,NUL.MAP,TTUTIL.LIB+LLIBFORE.LIB;<return>
```

The executable file can now be executed from the DOS command line by typing:

```
MAIN <return>
```

Model execution has been tested on AT-type personal computers with and without mathematical coprocessor.

6.2. Compilation and linking on μVAX/VMS-computers

Compiling and linking of the spring-wheat model on a μVAX/VMS computer will be explained. It is assumed that both the weather facility and the FORTRAN utility library TTUTIL are available in an object library. First, insert, the following statements in the file LOGIN.COM in your default directory:

```
$ FOR*TRAN :== FORTRAN/STANDARD/CHECK=(NOUNDERFLOW,OVERFLOW,BOUNDS)
$ DEFINE WEATHER "<directory-of-library>WEATHER.OLB"
$ DEFINE TTUTIL  "<directory-of-library>TTUTIL.OLB "
```

then execute command file (@LOGIN), move towards the model directory and type:

```
$ FOR MAIN,SWHEAT,REST <return>
```

After compilation, the object files MAIN.OBJ, SWHEAT.OBJ, REST.OBJ can be linked together with the WEATHER library and TTUTIL utility library into an executable file with:

```
$ LINK MAIN,SWHEAT,REST,WEATHER/L,TTUTIL/L <return>
```

After linking, the executable file called MAIN.EXE should exist which can be executed by typing:

```
$ RUN MAIN <return>
```

7. Model-options and execution

7.1. Simulation environment and output options

In this chapter it will be explained how to operate the model, how to define location and time within the Fortran Simulation Environment (FSE), how to change output options and execute multiple runs for different environments, or different model parameters. The environment for the simulation run is specified in the file TIMER.DAT. FSE will read this file and supply the appropriate data with respect to time, location and weather for the spring wheat module.

The file TIMER.DAT starts with two lines beginning with an exclamation mark. After the first exclamation mark the name of the directory where the weather data exist should be specified in. When the weather data, in CABO/TPE weather format (van Kraalingen, 1990), are in the model directory, only an exclamation mark is necessary in the first position of the line following the comment block. When the weather data exist in another directory, the name of that directory should be specified; "!C:\SYS\WEATHER\" or on the μVAX computer of CABO-DLO: "!WEATHER_DATA:". If a considerable number of weather files are available, they should preferably exist on a separate directory. After the second exclamation mark in the file, the acronym for the country name of the location should be filled in, NL, for the Netherlands as an example. These acronyms for country names can be defined by the user, but should not exceed 6 characters. After definition of the directory and country code, the variable ISTN, station number and IYEAR the start year of simulation are read from file. Country code, station number and start year of simulation are concatenated to a complete file name, for example:

```
FRANCE1.958 or ISR1.979
```

The directory and the name of the weather datafile are now specified to the model. The start of the simulation model is now specified as day number, 1 to 365 or 366 in leap years. The variable FINTIM, defining the last day of the simulation run, is specified in day number also, but in local time; on the first day of the simulation run, the corresponding variable TIME equals zero.
The output interval can be defined with the variable PRDEL. The model can only be executed with a time step for integration of 1, which should be defined in the variable DELT.
Output options can be specified with the variable ITABLE. Three possible file formats can be selected. When ITABLE equals 4, the output is formatted in tabulated form. When ITABLE equals 5 the output is in tab-delimited tables that can easily be included in spreadsheets. When ITABLE equals 6, FSE uses TTPLOT format, which results in pairwise output of state variables (y) against time (x).
During execution model output is written to the temporary file RES.BIN. After completion of the simulation run the output file, RESULTS.DAT is formatted. One can delete the temporary file RES.BIN by means of the variable IDTMP in the file TIMER.DAT. The temporary output file is deleted when IDTMP equals 1 and is saved when IDTMP equals 0.

The array-variable HARDAY will force FSE to synchronize modeloutput with data of periodic harvests from a field experiment. Model outcome can then easily be compared with experimental data.

7.2. Rerun option

FSE allows execution of multiple runs, varying initial values, parameters or forcing functions. Comparing results of reruns can indicate the model's sensitivity to certain model parameters or crop characteristics. The rerun facility of FSE is activated when a file with the name RERUNS.DAT exists on the model directory. If this file does not exist only one run is executed, the default run, that uses the values from the files TIMER.DAT, SOIL.DAT and PLANT.DAT. Executing multiple runs with FSE, the first run is always done with this default parameter set. If a file RERUNS.DAT exists, default values of initial values of state variables, model parameters or functions will be exchanged with those specified in the file RERUNS.DAT. When doing multiple runs for several years, for example, the reruns file looks like:

```
IYEAR = 1985
IYEAR = 1986
IYEAR = 1987
```

Reruns can be made more complicated by combining different parameter values at one time. Note that the variables NAPDAY and NTRMNT are arrays:

```
IYEAR = 1986 ; NAPDAY = 45.,130.,150. ; NTRMNT = 80.,60.,40.
IYEAR = 1986 ; NAPDAY = 45.,130.,150. ; NTRMNT = 3*60.

IYEAR = 1987 ; NAPDAY = 45.,130.,150. ; NTRMNT = 80.,60.,40.
IYEAR = 1987 ; NAPDAY = 45.,130.,150. ; NTRMNT = 3*60.
```

A complete rerun parameter-set is recognized as the name of the first variable in the reruns file is repeated. Note that for each run a complete set of rerun values is required. Normal data files differ from rerun files in that a variable name can exist only once.

8. Data file formats

Datafiles are formatted according to the syntax of the READ routines of the TTUTIL FORTRAN utility library (Rappoldt & van Kraalingen, 1990). Three subroutines RDSREA, RDSINT, RDAREA, read respectively values of real, integer or arrays of real variables. The files PLANT.DAT, SOIL.DAT, TIMER.DAT and RERUNS.DAT have identical formats, with the exception that TTUTIL read routines allow multiple occurrences of the same variable name only in the file RERUNS.DAT.

Some notes about editing the data files;

- the files should be saved in plain text mode (ASCII), without control characters, tabs, or extended ASCII characters,
- each variable in the datafile consists of a variable name and corresponding value(s) separated by an equal ('=') sign,
- although not necessary for model execution, a description of each variable and corresponding units will help the user to understand the model,
- the names of model variables cannot exceed six characters,
- when using array variables the numbers within the array should be separated by commas.

Functions can be formatted in this way:

```
*     effect of soil temperature on
*     decomposition of organic material
TFT = 0., 0.001, 5., 0.150, 10., 0.300, 20., 0.640 ! UNITS = -
```

or,

```
*     effect of soil temperature on
*     decomposition of organic material
TFT = 0., 0.001,
      5., 0.150,
     10., 0.300,
     20., 0.640 ! UNITS = -
```

- variable names can be defined on the same line if they are separated by a semi-colon, so 'A= 1.1; B=1.2E6; C=3.1428', is valid.
- the maximum length of lines within a data file is 80 characters.

When a file has an incorrect format, introduced by the user, or is corrupted for another reason, the TTUTIL-read routines will find the place and kind of error.

Literature

Bastiaans, L., B. Habekotté, H. van Oene & R. Werner, 1986. Konkurrentie tussen het gewas suikerbiet en de onkruiden melganzevoet en muur; veldexperiment en simulatie van monocultures. Deel 2: Simulatie van monocultures, Report Dept. of Theoretical Production Ecology, 45 pp.

Goudriaan, J., 1986. A simple and fast numerical method for the computation of daily totals of crop photosynthesis. Agricultural and Forest Meteorology 38, 251-255

Goudriaan, J. & H.H. van Laar, 1978. Calculation of daily totals of the gross CO_2 assimilation of leaf canopies. Netherlands Journal of Agricultural Science 26, 373-382

IBM, 1975. Continuous System Modeling Program III (CSMP III), Program Reference Manual, Fourth Edition, White Plains, New York, 206 pp.

Keulen, H. van & N.G. Seligman, 1987. Simulation of water use, nitrogen nutrition and growth of a spring-wheat crop, Simulation Monographs, PUDOC, Wageningen, 309 pp.

Kraalingen, D.W.G. van, 1991. The FSE system for crop simulation, Simulation Report CABO-TT no. 23 , Centre for Agrobiological Research, Dept. of Theoretical Production Ecology, PO Box 430, Wageningen, 77 pp.

Kraalingen, D.W.G. van & C. Rappoldt, 1989. Subprograms in simulation models, Simulation Report CABO-TT no. 18, Centre for Agrobiological Research, Dept. of Theoretical Production Ecology, PO Box 430, Wageningen, 54 pp.

Kraalingen, D.W.G. van & F.W.T. Penning de Vries, 1990. The FORTRAN version of CSMP-MACROS (Modules for Annual CRop Simulation), Simulation Report CABO-TT no. 21, Centre for Agrobiological Research, Dept. of Theoretical Production Ecology, PO Box 430, Wageningen, 145 pp.

Kraalingen, D.W.G. van, W. Stol, P.W.J. Uithol & M.G.M. Verbeek, 1990. User manual of CABO/TPE Weather System, Centre for Agrobiological Research, Dept. of Theoretical Production Ecology, PO Box 430, Wageningen, 27 pp.

Kruyt, E.W., 1989. FORCHECK: A FORTRAN Verifier and Programming Aid, version 09, IBM PC/DOS, User's Guide, Department of Physiology, Leiden University, Leiden

Microsoft, 1989. Microsoft FORTRAN Reference version 5.0, Microsoft Code View and Utilities User's Guide, Advanced Topics, Microsoft Editor User's Guide version 2.3, Microsoft Corporation

Penning de Vries, F.W.T. & H.H. van Laar (Eds.), 1982. Simulation of plant growth and crop production. Simulation Monographs, PUDOC, Wageningen, 320 pp.

Rappoldt, C. & D.W.G. van Kraalingen, 1990. Reference manual of the FORTRAN utility library TTUTIL with applications, Simulation Report CABO-TT no. 20, Centre for Agrobiological Research, Dept. of Theoretical Production Ecology, PO Box 430, Wageningen, 122 pp.

Spitters, C.J.T., 1986. Separating the diffuse and direct component of global radiation and its implications for modeling canopy photosynthesis. II. Calculations of canopy photosynthesis. Agricultural and Forest Meteorology 38, 231-242

Spitters, C.J.T., H. van Keulen & D.W.G. van Kraalingen, 1989. A simple and universal crop growth simulator: SUCROS87. In: R. Rabbinge, S.A. Ward & H.H. van Laar (Eds.), Simulation and systems management in crop protection, Simulation Monographs, Pudoc, Wageningen, 147-181

Spitters, C.J.T., H.A.J.M. Toussaint & J. Goudriaan, 1986. Separating the diffuse and direct component of global radiation and its implications for modeling canopy photosynthesis. I. Components of incoming radiation. Agricultural and Forest Meteorology 38, 217-229

Wit, C.T. de & J. Goudriaan, 1978. Simulation of ecological processes, Simulation Monographs, Pudoc, Wageningen, 175 pp.

Appendix I:

Source and data file listings

Program: MAIN.FOR

```
*-----------------------------------------------------------------------*
*                                                                       *
*                     S P R I N G - W H E A T                           *
*             Simulation of water use, nitrogen nutrition               *
*                 and growth of a spring wheat crop                     *
*                                                                       *
*                             Version 2                                 *
*                           February 1990                               *
*                                                                       *
*                          H. van Keulen                                *
*                           N. Seligman                                 *
*                       D.W.G. van Kraalingen                           *
*                                                                       *
* This is the main program of the FORTRAN version of the CABO crop      *
* growth simulation model Spring-Wheat. This version is taken from      *
* the CSMP program WHEAT which, together with the scientific            *
* background, can be found in:                                          *
*                                                                       *
* Van Keulen, H. & N.G. Seligman, Simulation of water use, nitrogen     *
* nutrition and growth of a spring wheat crop, Simulation Monographs,   *
* PUDOC Wageningen. 309 pp.                                             *
*                                                                       *
* The documentation and listing of this program can be obtained by      *
* submitting a request to CABO, c/o H. van Keulen, P.O. Box 14, 6700    *
* AA Wageningen, The Netherlands.                                       *
*                                                                       *
*-----------------------------------------------------------------------*

      PROGRAM MAIN
      IMPLICIT REAL (A-Z)

*-----STANDARD DECLARATIONS
      INTEGER      ITASK,  INSETS, IRUN,   I1, I2, I3
      INTEGER      IUNITR, IUNITT, IUNITO, IUNITP, IUNITS
      INTEGER      ISTAT,  IDAY,   IYEAR,  ISTN,   ILEN
      INTEGER      ITABLE, IDTMP,  IMNHD,  INHD
      LOGICAL      OUTPUT, TERMNL, WTRMES
      CHARACTER*7  CNTR
      CHARACTER*80 WTRDIR, FILER, FILES, FILET, FILEO, FILEP
      PARAMETER (IMNHD=20, TINY=1.E-4)
      REAL HARDAY(IMNHD)

*-----SPECIAL DECLARATION REQUIRED TO WRITE WARNINGS TO OUTPUT FILE
      COMMON /LOGCOM/ IUNITO, TIME, IRUN

*-----DECLARATIONS NECESSARY FOR USE OF SOIL-WATER BALANCE
*     LINE 1: FILE I/O UNIT, NUMBER OF LAYERS, AND MAXIMUM NUMBER
*             OF LAYERS
*        ,, 2: MAXIMUM NUMBER OF LAYERS
*        ,, 3: VOLUMETRIC WATER CONTENTS
*        ,, 4: THICKNESS, TRANSPIRATION RATES AND WATER CONTENTS
*        ,, 5: STRING THAT HOLDS NAME OF SOIL DATA FILE

*     UNIT NUMBERS FOR RERUN (R), TIMER (T), OUTPUT (O), PLANT DATA (P)
*     AND SOIL DATA (S) FILES. WTRMES FLAGS ANY MESSAGES FROM
*     THE WEATHER SYSTEM

      IUNITR = 20
      IUNITT = 30
      IUNITO = 40
      IUNITP = 50
      IUNITS = 60
      WTRMES = .FALSE.

*-----FILE NAMES
      FILER   = 'RERUNS.DAT'
      FILET   = 'TIMER.DAT'
      FILEO   = 'RESULTS.DAT'
      FILEP   = 'PLANT.DAT'
      FILES   = 'SOIL.DAT'

*-----GET DIRECTORY AND COUNTRY NAME OF WEATHER DATA FROM TIMER FILE
      CALL FOPEN (IUNITT, FILET, 'OLD', 'NVT')
      CALL MOFILP (IUNITT)
      READ (IUNITT,'(A)') WTRDIR
      READ (IUNITT,'(A)') CNTR
      I2 = ILEN (WTRDIR)
      I3 = ILEN (CNTR)
      IF (I2.NE.0.AND.I3.NE.0) THEN
         I2 = MAX(2,I2)
         I3 = MAX(2,I3)
         WTRDIR(1:I2) = WTRDIR(2:I2)//' '
         CNTR (1:I3) = CNTR(2:I3)//' '
      ELSE
         CALL ERROR
     &         ('FSE-WHEAT', 'empty record while reading weather data')
      END IF
      CLOSE (IUNITT)

*-----OPEN OUTPUT FILE, READ NUMBER OF RERUN SETS
      CALL FOPEN (IUNITO, FILEO, 'NEW', 'DEL')
      CALL RDSETS (IUNITR, IUNITO, FILER, INSETS)

      DO 10 I1=0,INSETS

      IRUN = I1+1
      WRITE (*,'(A)') ' '

*-----SELECT DATA SET
      CALL RDFROM (I1, .TRUE.)

*-----INITIALIZATION SECTION
      ITASK  = 1
      TERMNL = .FALSE.

*-----READ VARIABLES FROM TIMER FILE
      CALL RDINIT (IUNITT, IUNITO, FILET)
      CALL RDSREA ('DAYB',   DAYB)
      CALL RDSREA ('FINTIM', FINTIM)
      CALL RDSREA ('PRDEL',  PRDEL)
      CALL RDSREA ('DELT',   DELT)
      CALL RDSINT ('IYEAR',  IYEAR)
      CALL RDSINT ('ISTN',   ISTN)
      CALL RDSINT ('ITABLE', ITABLE)
      CALL RDSINT ('IDTMP' , IDTMP)
      CALL RDAREA ('HARDAY', HARDAY, IMNHD, INHD)
      CLOSE (IUNITT, STATUS='DELETE')

      CALL TIMER (ITASK, DAYB, DELT, PRDEL, FINTIM,
     &            IYEAR,  TIME, DAY, IDAY, TERMNL, OUTPUT)
      CALL OUTDAT (ITASK, IUNITO, 'TIME', TIME)
      CALL STINFO (1111, WTRDIR, ' ', CNTR, ISTN, IYEAR,
     &            ISTAT, LONG, LAT, ELV, A, B)
      WTRMES = WTRMES.OR.(ISTAT.NE.0)
      CALL WEATHR (IDAY, ISTAT, RADI, TMIN, TMAX, VAPOUR, WIND, RAIN)

      WTRMES = WTRMES.OR.(ISTAT.NE.0)
      TERMNL = TERMNL.OR.(ISTAT.LT.0.AND.ISTAT.GT.-111111)

      CALL SWHEAT (
     $ ITASK, IUNITP, IUNITO, IUNITS, FILEP, FILES, OUTPUT, TERMNL,
     $ TIME, DAY, DELT,
     $ LAT,
     $ RADI, TMIN, TMAX, VAPOUR, WIND, RAIN)

*-----DYNAMIC SIMULATION SECTION

20    IF (.NOT.TERMNL) THEN

      WRITE (*,'(A,I4,A,I5,A,F7.2)')
     &  ' Run:', IRUN, ', Year:', IYEAR, ' Day:', DAY

      IF (ITASK.NE.1) THEN
*-----   INTEGRATION OF RATES SECTION
         ITASK = 3

         CALL SWHEAT (
     $    ITASK, IUNITP, IUNITO, IUNITS, FILEP, FILES, OUTPUT, TERMNL,
     $    TIME, DAY, DELT,
     $    LAT,
     $    RADI, TMIN, TMAX, VAPOUR, WIND, RAIN)
      END IF

      ITASK = 2
*-----CALCULATION OF DRIVING VARIABLES SECTION
      IF (OUTPUT.OR.TERMNL) THEN
         CALL OUTDAT (ITASK, IUNITO, 'TIME', TIME)
         CALL OUTDAT (ITASK, IUNITO, 'DAY', DAY)
      END IF

      CALL STINFO (1111, WTRDIR, ' ', CNTR, ISTN, IYEAR,
     &            ISTAT, LONG, LAT, ELV, A, B)
      WTRMES = WTRMES.OR.(ISTAT.NE.0)
      CALL WEATHR (IDAY, ISTAT, RADI, TMIN, TMAX, VAPOUR, WIND, RAIN)

      WTRMES = WTRMES.OR.(ISTAT.NE.0)
      TERMNL = TERMNL.OR.(ISTAT.LT.0.AND.ISTAT.GT.-111111)
*-----DISCARD NEGATIVE VALUES FOR RAINFALL
      IF (RAIN.LT.0.) THEN
         WRITE (*,*) ' RAIN MISSING ON DAY: ', IDAY
         RAIN = 0.
      END IF

*-----CALCULATION OF RATES SECTION
      CALL SWHEAT (
     $ ITASK, IUNITP, IUNITO, IUNITS, FILEP, FILES, OUTPUT, TERMNL,
     $ TIME, DAY, DELT,
     $ LAT,
     $ RADI, TMIN, TMAX, VAPOUR, WIND, RAIN)

*-----TIME UPDATE, CHECK FOR FINTIM AND OUTPUT
      CALL TIMER (ITASK, DAYB, DELT, PRDEL, FINTIM,
     &            IYEAR,  TIME, DAY, IDAY, TERMNL, OUTPUT)

*-----generate output to file if day is equal to a harvest day
      IF (NINT(HARDAY(1)).NE.0) THEN
         DO 30 I2=1,INHD
            IF (DAY.GT.(HARDAY(I2)-TINY).AND.DAY.LT.(HARDAY(I2)+TINY))
     &         OUTPUT = .TRUE.
30       CONTINUE
      ELSE IF (INHD.GT.1) THEN
         CALL ERROR('FSE-WHEAT','harvest data in TIMER.DAT not correct')
      END IF
      GOTO 20
      END IF

*-----TERMINAL SECTION
      ITASK = 4

*-----GENERATE OUTPUT FILE USING NORMAL TABLE FORMAT
*     AND DELETE TEMPORARY OUTPUT FILE.

      CALL OUTDAT (ITABLE, 20, 'SPRING-WHEAT simulation model',0.)

      CALL SWHEAT (
     $ ITASK, IUNITP, IUNITO, IUNITS, FILEP, FILES, OUTPUT, TERMNL,
     $ TIME, DAY, DELT,
     $ LAT,
     $ RADI, TMIN, TMAX, VAPOUR, WIND, RAIN)

*     DELETE TEMPORARY OUTPUT FILE DEPENDENT ON SWITCH FROM TIMER FILE
      IF (IDTMP.EQ.1) CALL OUTDAT (99, 0, ' ', 0.)

10    CONTINUE

*-----DELETE TEMPORARY RERUN FILE
      CLOSE (IUNITR, STATUS='DELETE')

      IF (WTRMES) THEN
         WRITE (*,'(A,/,A)')
     $      ' There have been errors and/or warnings from',
     $      ' the weather system, check file WEATHER.LOG'
         WRITE (IUNITO,'(A,/,A)')
     $      ' There have been errors and/or warnings from',
     $      ' the weather system, check file WEATHER.LOG'
      END IF

      STOP
      END
```

Program: SWHEAT.FOR

```
*----------------------------------------------------------------------*
*  SUBROUTINE SWHEAT                                                   *
*  Authors: Willem Stol & Daniel van Kraalingen                        *
*  Date   : November 1992                                              *
*  Version: 2.0                                                        *
*  Purpose: This subroutine simulates water and nitrogen limited       *
*           dry matter production of spring-wheat.                     *
*           It includes the water-, carbon- and nitrogen balance of    *
*           the soil and the crop.                                     *
*                                                                      *
*  FORMAL PARAMETERS:  (I=input,O=output,C=control,IN=init,T=time)     *
*  name   type meaning                                    units  class *
*  ----   ---- -------                                    -----  ----- *
*  control                                                             *
*  ITASK   I4  Determines action of the subroutine,                    *
*              1=initialization, 2=rate calculation,                   *
*              3=integration, 4=terminal                    -     C,I  *
*  IUNITP  I4  Unit number of plant data file,              -     C,IN *
*  IUNITO  I4  Unit number of output file,                  -     C,IN *
*  IUNITS  I4  Unit number of soil data file,               -     C,IN *
*  FILEP   C*  Name of plant data file,                     -     C,IN *
*  FILES   C*  Name of soil data file,                      -     C,IN *
*  OUTPUT  L4  Flag that indicates if output to file is                *
*              required,                                    -     C,I  *
*  TERMNL  L4  Flag that indicates if simulation should                *
*              terminate                                    -    C,I,O *
*                                                                      *
*  timing                                                              *
*  TIME    R4  Time of simulation                           d       T  *
*  DATE    R4  Day number of simulation                     d       T  *
*  DELT    R4  Time step of integration                     d       T  *
*                                                                      *
*  environment                                                         *
*  LAT     R4  Latitude of weather station             degrees      I  *
*  RADI    R4  Global radiation                        KJ/m2/d      I  *
*  TMIN    R4  Minimum temperature             degrees Celsius      I  *
*  TMAX    R4  Maximum temperature             degrees Celsius      I  *
*  VAPOUR  R4  Vapour pressure                             kPa      I  *
*  WIND    R4  Average wind speed                          m/s      I  *
*  RAIN    R4  Precipitation                              mm/d      I  *
*                                                                      *
*  FATAL ERROR CHECKS (execution terminated, message): -               *
*                                                                      *
*  Certain sequences of ITASK, see subroutine CHKTSK                   *
*  Carbon balance check                                                *
*                                                                      *
*  SUBROUTINES and FUNCTIONS called:                                   *
*  from TTUTIL : CHKTSK, RDINIT, RDSREA, RDAREA, OUTDAT, LINT, INSW,   *
*                INTGRL, LIMIT                                         *
*  from TTLIB :  ASTRO, TOTASS                                         *
*  other: REANOT, DELAY                                                *
*                                                                      *
*  FILE usage : -Plant definition file FILEP opened and closed for     *
*                ITASK=1, unit numbers used are IUNITP and IUNITP+1    *
*               -Soil definition file FILES opened and closed for      *
*                ITASK=1, unit numbers used are IUNITS and IUNITS+1    *
*               -Output file with unit IUNITO for output and warnings  *
*----------------------------------------------------------------------*

      SUBROUTINE SWHEAT (
     $ ITASK, IUNITP, IUNITO, IUNITS, FILEP, FILES, OUTPUT, TERMNL,
     $ TIME, DATE, DELT,
     $ LAT,
     $ RADI, TMIN, TMAX, VAPOUR, WIND, RAIN)

      IMPLICIT REAL(A-Z)

*-----maximum array length for variables within soil
*     compartments (1), compartment boundaries (2) and
*     boxcar-train (3) with temperatures
      INTEGER ILMAX1,ILMAX2,ILMAX3
      PARAMETER (ILMAX1=10,ILMAX2=11,ILMAX3=12)

      INTEGER IARL1,IARL2
      PARAMETER (IARL1=30,IARL2=110)

*-----conversion factor from weights co2 to ch2o
      PARAMETER (CCO2TS=0.6818)

*-----standard declarations of variables
      INTEGER ITASK,  ITOLD
      INTEGER IUNITO, IUNITP, IUNITS
      CHARACTER*80 FILES,  FILEP
      LOGICAL OUTPUT, TERMNL

*-----declarations of variables specific to swheat module

*-----counter variables used in do-loops
      INTEGER I, K, NF, NNA, LG

      INTEGER ILAYER

      DIMENSION WATER(ILMAX1),AWATER(ILMAX1),DWAT(ILMAX1)

      INTEGER IIAS
      DIMENSION IAS(ILMAX1)

      DIMENSION RWF(ILMAX2),RTL(ILMAX2),TDT(ILMAX2),AWATF(ILMAX2)

      INTEGER IIBIOM,IDFFOM,IFLDCP,IWLTPT
      DIMENSION BIOMN(ILMAX2),DFFOM(ILMAX2),FLDCP(ILMAX2),WLTPT(ILMAX2)

      DIMENSION FOM(ILMAX2),NHUM(ILMAX2),RNRLB(ILMAX2)

      INTEGER ITHCKN
      DIMENSION THCKN(ILMAX2)

      DIMENSION STORC(ILMAX2),TRR(ILMAX2),CONC(ILMAX2),WRED(ILMAX1)

      DIMENSION VAR(ILMAX2),ER(ILMAX2),EDPTF(ILMAX2)

      INTEGER IWREDT,IEDPTN,IRDFDT,IIRRT
      DIMENSION WREDT(IARL1),EDPTFT(IARL1),REDFDT(IARL1),IRRT(IARL1)

      INTEGER IROSPT,IDISTF,IDRFA
      DIMENSION ROSPT(IARL1),DISTF(ILMAX1),DRFA(ILMAX1)

      DIMENSION IFOM(ILMAX1),CNR(ILMAX2),CNRF(ILMAX2),MF(ILMAX2)
      DIMENSION FOMRES(ILMAX2),RDECR(ILMAX2),DECR(ILMAX2),RNRL(ILMAX2)
      DIMENSION RHMIN(ILMAX2),DFON(ILMAX2),DHUM(ILMAX2),DNHUM(ILMAX2)
      DIMENSION DFOM(ILMAX2),CADEC(ILMAX2),CFBMG(ILMAX2),BIOMXC(ILMAX2)
      DIMENSION BIOMXN(ILMAX2),BIOMX(ILMAX2),DBIOMN(ILMAX2)
      DIMENSION IBIOMN(ILMAX2)
      DIMENSION HUM(ILMAX2),FON(ILMAX2),ASLT(ILMAX2),DBN(ILMAX2)
      DIMENSION RNAC(ILMAX2),SLTF(ILMAX2),CONP(ILMAX2),RNUM(ILMAX2)
      DIMENSION FERTAP(ILMAX2),RNUD(ILMAX2),RNUDB(ILMAX2),DASLT(ILMAX2)
      DIMENSION RVOL(ILMAX2)

      INTEGER IMFT,IAPDAY,INTRMT,ITFT,ITECT
      DIMENSION MFT(IARL1),NAPDAY(IARL1),NTRMNT(IARL1),TFT(IARL1)
      DIMENSION TECT(IARL1)

      DIMENSION TMPCX(ILMAX3)

      INTEGER IRADTA,IRADTB,IRADTC,IRADTD
      DIMENSION RADTBA(IARL1),RADTBB(IARL1),RADTBC(IARL1),RADTBD(IARL1)

      INTEGER IRADTE,IRADTF,IRADTG,IRADTH
      DIMENSION RADTBE(IARL1),RADTBF(IARL1),RADTBG(IARL1),RADTBH(IARL1)

      INTEGER IRADTI,IRADTJ,ICRHTB,IALPTA
      DIMENSION RADTBI(IARL1),RADTBJ(IARL1),CROHTB(IARL1),ALPHTA(IARL1)

      INTEGER IALPTB,IALPTC,IALPTD,IALPTE
      DIMENSION ALPHTB(IARL1),ALPHTC(IARL1),ALPHTD(IARL1),ALPHTE(IARL1)

      INTEGER IALPTF,IWCRRT,ITMPFT,IFTLST
      DIMENSION ALPHTF(IARL1),WCRRT(IARL1),TMPFT(IARL1),FTLVST(IARL1)

      INTEGER IRDWST,IRFNST,IFDMT,IFSTET
      DIMENSION REDWST(IARL1),RFNST(IARL1),FDMT(IARL1),FTSTET(IARL1)

      INTEGER IFDSRT,IRDTTB,IFTRLT,IPGIGT
      DIMENSION FDSRT(IARL1),REDTTB(IARL1),FTRLT(IARL1),PGRIGT(IARL1)

      DIMENSION DLEAF(IARL2),DALFT(IARL2),DLA(IARL2),LEAFA(IARL2)
      DIMENSION LEAFW(IARL2),ALFT(IARL2)

      INTEGER ITCTFT,IRFTT,IAGEFT,IRDRNT
      DIMENSION TCTFT(IARL1),RFTT(IARL1),AGEFT(IARL1),RDRNT(IARL1)

      INTEGER IBNT,ILNT,IPRAGT,IFNEXT
      DIMENSION BNT(IARL1),LNT(IARL1),PRNAGT(IARL1),FNEXT(IARL1)

      SAVE

*-----end of declaration section                                    20

      DATA ITOLD /4/

*-----the task that the subroutine should do (itask) against
*     the task that was done during the previous call (itold)
*     is checked. only certain combinations are allowed. these
*     are:
*           new task:                old task:
*           initialization           terminal
*           integration              rate calculation
*           rate calculation         initialization, integration
*           terminal                 <any old task>
*
*     note: there is one combination that is correct but will
*     not cause calculations to be done i.e. if integration
*     is required immediately after initialization.

      CALL CHKTSK ('SUB', IUNITO, ITOLD, ITASK)

*-----initialization section                                        30
      IF (ITASK.EQ.1) THEN

*-----initialize file with plantdata
         CALL RDINIT (IUNITP, IUNITO, FILEP)

*-----switch variable used to initialize crop state variables       40
*     1 at emergence otherwise 0, -
         PUSHDI = 0.
         PUSHD = 0.

*-----definitions and units of plant-parameters, variables          50
*     and functions are included in the file: plant.dat
         CALL RDSREA ('TGERD' ,TGERD)
         CALL RDSREA ('TSDAY' ,TSDAY)
         CALL RDSREA ('SOWD'  ,SOWD)
         CALL RDSREA ('GERDAT',GERDAT)
         CALL RDSREA ('RADCF' ,RADCF)
         CALL RDSREA ('EXPTF' ,EXPTF)
         CALL RDSREA ('CTEMPF',CTEMPF)
         CALL RDSREA ('FTMPA' ,FTMPA)
         CALL RDSREA ('RAINF' ,RAINF)

*-----total seasonal rainfall, mm                                   60
         TRAIN = 0.

*-----total seasonal crop transpiration, mm                         70
         TOTRAN = 0.

         CALL RDSREA ('PI'    ,PI)
         CALL RDSREA ('DRAGC' ,DRAGC)
         CALL RDSREA ('KARMAN',KARMAN)
         CALL RDSREA ('IW'    ,IW)
         CALL RDSREA ('REFHT' ,REFHT)
         CALL RDSREA ('RHOCP' ,RHOCP)
         CALL RDSREA ('PSCH'  ,PSCH)
         CALL RDSREA ('SCM'   ,SCM)
         CALL RDSREA ('RC'    ,RC)
         CALL RDSREA ('EXC'   ,EXC)

         CALL RDAREA ('CROHTB',CROHTB ,IARL1 ,ICRHTB)
         CALL RDAREA ('ALPHTA',ALPHTA ,IARL1 ,IALPTA)
         CALL RDAREA ('ALPHTB',ALPHTB ,IARL1 ,IALPTB)
         CALL RDAREA ('ALPHTC',ALPHTC ,IARL1 ,IALPTC)
         CALL RDAREA ('ALPHTD',ALPHTD ,IARL1 ,IALPTD)
         CALL RDAREA ('ALPHTE',ALPHTE ,IARL1 ,IALPTE)
         CALL RDAREA ('ALPHTF',ALPHTF ,IARL1 ,IALPTF)
```

```
      CALL RDAREA ('WCRRT' ,WCRRT  ,IARL1 ,IWCRRT)

      CALL RDSREA ('NFDEV' ,NFDEV)
      CALL RDSREA ('CULTP' ,CULTP)
      CALL RDSREA ('CULTM' ,CULTM)

      CALL RDSREA ('DVSI'  ,DVSI)
      CALL RDSREA ('DVSTS' ,DVSTS)
      CALL RDSREA ('DVSSE' ,DVSSE)
      CALL RDSREA ('DVSSPS',DVSSPS)
      CALL RDSREA ('DVSST' ,DVSST)
      CALL RDSREA ('DVSFS' ,DVSFS)
      CALL RDSREA ('DVSPRE',DVSPRE)
      CALL RDSREA ('DVSSPE',DVSSPE)
      CALL RDSREA ('DVSFE' ,DVSFE)
      CALL RDSREA ('DVSAN' ,DVSAN)
      CALL RDSREA ('DVSGS' ,DVSGS)
      CALL RDSREA ('DVSSGF',DVSSGF)
      CALL RDSREA ('DVSEGF',DVSEGF)

*-----development stage of the crop in the pre-anthesis phase, -     80
      DVSV = 0.

*-----anthesis date, julian day number                                90
      ANTHES = 0.

*-----development stage of the crop in the post-anthesis phase, -   100
      DVSR = 0.

*-----initialization of actual development stage with the           110
*     development stage at emergence, -
      DVS = DVSI

      CALL RDSREA ('EFFE'  ,EFFE)
      CALL RDSREA ('KDIF'  ,KDIF)
      CALL RDSREA ('SCV'   ,SCV)

      CALL RDSREA ('TLRGA' ,TLRGA)
      CALL RDSREA ('RMRESL',RMRESL)
      CALL RDSREA ('RMRESS',RMRESS)
      CALL RDSREA ('RMRESR',RMRESR)
      CALL RDSREA ('RMRESG',RMRESG)
      CALL RDSREA ('Q10'   ,Q10)
      CALL RDSREA ('EFCPR' ,EFCPR)
      CALL RDSREA ('EFCCH' ,EFCCH)
      CALL RDSREA ('EFCPRG',EFCPRG)
      CALL RDSREA ('TCDDH' ,TCDDH)
      CALL RDSREA ('FWDB'  ,FWDB)
      CALL RDSREA ('MRDRSH',MRDRSH)
      CALL RDSREA ('LAILM' ,LAILM)
      CALL RDSREA ('MXRDR' ,MXRDR)
      CALL RDSREA ('AVLTLF',AVLTLF)
      CALL RDSREA ('RSLDS' ,RSLDS)
      CALL RDSREA ('RFST'  ,RFST)
      CALL RDSREA ('FSCHG' ,FSCHG)
      CALL RDSREA ('RFRT'  ,RFRT)
      CALL RDSREA ('DGRRT' ,DGRRT)
      CALL RDSREA ('MXRTD' ,MXRTD)
      CALL RDSREA ('RTF'   ,RTF)
      CALL RDSREA ('RESLI' ,RESLI)
      CALL RDSREA ('RESL1' ,RESL1)
      CALL RDSREA ('RESL2' ,RESL2)
      CALL RDSREA ('TCTR'  ,TCTR)
      CALL RDSREA ('FRNGL' ,FRNGL)
      CALL RDSREA ('FRNGL1',FRNGL1)
      CALL RDSREA ('RESLR' ,RESLR)
      CALL RDSREA ('SWDF'  ,SWDF)

      CALL RDAREA ('TMPFT' ,TMPFT  ,IARL1 ,ITMPFT)
      CALL RDAREA ('FTLVST',FTLVST ,IARL1 ,IFTLST)
      CALL RDAREA ('REDWST',REDWST ,IARL1 ,IRDWST)
      CALL RDAREA ('RFNST' ,RFNST  ,IARL1 ,IRFNST)
      CALL RDAREA ('FDMT'  ,FDMT   ,IARL1 ,IFDMT)
      CALL RDAREA ('FTSTET',FTSTET ,IARL1 ,IFSTET)
      CALL RDAREA ('FDSRT' ,FDSRT  ,IARL1 ,IFDSRT)
      CALL RDAREA ('REDTTB',REDTTB ,IARL1 ,IRDTTB)
      CALL RDAREA ('FTRLT' ,FTRLT  ,IARL1 ,IFTRLT)
      CALL RDAREA ('PGRIGT',PGRIGT ,IARL1 ,IPGIGT)

*-----initialization of leaf weights, accumulated temperature       120
*     sum and leaf area per leaf class
      DO 10 I = 1,110
         LEAFW(I) = 0.
         ALFT(I)  = 0.
         LEAFA(I) = 0.
10    CONTINUE

*-----rooting depth, mm                                              130
      RTD = 0.

*-----end of grain filling day, julian day number                    140
      EGFDAY = 0.

*-----initial dry weight of live leaf blades, at emergence,          150
*     kg/ha
      WLVSI = 0.

*-----dry weight of live leaf blades, kg/ha                          160
      WLVS = 0.

*-----initial dry weight of live roots, at emergence, kg/ha          170
      WRTI = 0.

*-----dry weight of live roots, kg/ha                                180
      WRT = 0.

*-----dry weight of live stem and sheaths, at emergence, kg/ha       190
      WSTEM = 0.

*-----dry weight of the grain, kg/ha                                 200
      WGR = 0.

*-----total weight of the vegetative crop organs, kg/ha              210
      TVEGM = 0.

*-----total weight of dead material above ground, kg/ha              220
      DSTR = 0.

*-----total above ground dry weight, kg/ha                           230
      TADRW = 0.

*-----shoot root ratio, -                                            240
      SRR = 0.

*-----initial weight of reserve carbohydrates, at emergence,         250
*     kg/ha
      ARESPI = 0.

*-----weight of reserve carbohydrates, kg/ha                         260
      ARESP = 0.

*-----initial tiller density, at emergence, no/ha                    270
      TLNI = 0.

*-----auxilary variable to save original value of tlni, no/ha        280
      TLNIX = 0.

*-----tiller density, no/ha                                          290
      TLN = 0.

*-----average weight per tiller, kg                                  300
      AWTL = 0.

      CALL RDSREA ('CHFTB',CHFTB)
      CALL RDSREA ('RDRT' ,RDRT)
      CALL RDSREA ('STCEF',STCEF)
      CALL RDSREA ('EB'   ,EB)
      CALL RDSREA ('CHFEB',CHFEB)
      CALL RDSREA ('STCSF',STCSF)
      CALL RDSREA ('SB'   ,SB)
      CALL RDSREA ('CHFSB',CHFSB)
      CALL RDSREA ('STCFF',STCFF)
      CALL RDSREA ('FB'   ,FB)
      CALL RDSREA ('CHFFB',CHFFB)
      CALL RDSREA ('STCGF',STCGF)
      CALL RDSREA ('GB'   ,GB)

      CALL RDAREA ('TCTFT',TCTFT ,IARL1 ,ITCTFT)
      CALL RDAREA ('RFTT' ,RFTT  ,IARL1 ,IRFTT)

*-----total number of dead tillers, no/ha                            310
      DTLN = 0.

*-----total number of non-reproductive tillers, no/ha                320
      TNNR = 0.

*-----average weight of non-reproductive tillers at end              330
*     of ear formation, kg/tiller
      LWDTL = 0.

*-----final weight of stem and sheaths of non-reproductive           340
*     tillers, kg/tiller
      SWDTL = 0.

*-----total number of ears, no/ha                                    350
      EARN = 0.

*-----total number of spikelets, no/ha                               360
      NSPS = 0.

*-----total number of fertile florets, no/ha                         370
      FFNR = 0.

*-----grain density, no/ha                                           380
      GRN = 0.

*-----initial green area of leaf blades, at emergence, m2/ha         390
      ARLFI = 0.

*-----green area of leaf blades, m2/ha                               400
      ARLF = 0.

*-----total area of green and senesced leaf blades, m2/ha            410
      ARLFE = 0.

*-----leaf area index, m2/m2                                         420
      LAI = 0.

*-----green area index, m2/m2                                        430
      GRAI = 0.

      CALL RDSREA ('FLFARM' ,FLFARM)
      CALL RDSREA ('ARPEAR' ,ARPEAR)
      CALL RDSREA ('MXSTAR' ,MXSTAR)
      CALL RDSREA ('FLFARI' ,FLFARI)

      CALL RDAREA ('AGEFT'  ,AGEFT ,IARL1 ,IAGEFT)

      CALL RDSREA ('FRNN'   ,FRNN)
      CALL RDSREA ('NDPAR'  ,NDPAR)
      CALL RDSREA ('HTFAC'  ,HTFAC)
      CALL RDSREA ('TCU'    ,TCU)

      CALL RDSREA ('FNMIN'  ,FNMIN)
      CALL RDSREA ('FNMXA'  ,FNMXA)
      CALL RDSREA ('FNMXR'  ,FNMXR)
      CALL RDSREA ('FNMXSA' ,FNMXSA)
      CALL RDSREA ('FNMNSR' ,FNMNSR)
      CALL RDSREA ('FNRTMX' ,FNRTMX)
      CALL RDSREA ('FNRTMN' ,FNRTMN)
      CALL RDSREA ('RN'     ,RN)
      CALL RDSREA ('TCUD'   ,TCUD)
      CALL RDSREA ('UMXR'   ,UMXR)
      CALL RDSREA ('CF'     ,CF)
      CALL RDSREA ('RRTORT' ,RRTORT)

      CALL RDAREA ('RDRNT' ,RDRNT ,IARL1 ,IRDRNT)
      CALL RDAREA ('BNT'   ,BNT   ,IARL1 ,IBNT)
      CALL RDAREA ('LNT'   ,LNT   ,IARL1 ,ILNT)
      CALL RDAREA ('PRNAGT',PRNAGT,IARL1 ,IPRAGT)
      CALL RDAREA ('FNEXT' ,FNEXT ,IARL1 ,IFNEXT)

*-----maximum nitrogen concentration in leaf blades, kg/kg           440
      FNMAX = FNMXR+FNMXA

*-----initial fraction of nitrogen in leaves at emergence,           450
*     equal to maximum, kg/kg
      FRNVI = FNMAX

*-----range of nitrogen fractions in plant organs, used in           460
```

```
*     calculation of maintenance respiration, kg/kg
         RNFAC = FNMAX-FNMIN

*-----initial fraction of nitrogen in the roots at emergence,      470
*     kg/kg
         IFNRT = FNRTMN+(1.-DVSI)*(FNRTMX-FNRTMN)

*-----initial weight of nitrogen in the leaf blades, at             480
*     emergence, kg/ha
         ANLVI = 0.

*-----weight of nitrogen in the leaf blades, at emergence, kg/ha   490
         ANLV = 0.

*-----weight of nitrogen in the stem and leaf sheaths, kg/ha        500
         ANSTE = 0.

*-----initial weight of nitrogen in the roots, at emergence,        510
*     kg/ha
         ANRTI = 0.

*-----weight of nitrogen in the roots, kg/ha                        520
         ANRT = 0.

*-----weight of nitrogen in the grains, kg/ha                       530
         ANGR = 0.

*-----weight of nitrogen in dead above ground material, kg/ha       540
         NDSTR = 0.

*-----total weight of nitrogen in above ground material, kg/ha      550
         TNABM = 0.

*-----initial weight of nitrogen in the vegetation at               560
*     emergence, kg/ha
         INIV = 0.

*-----maximum rate of gross co2 assimilation of a single leaf in    570
*     in dependence of nitrogen concentration and temperature,
*     kg/ha/h
         AMAXN = 0.
         CLOSE (IUNITP, STATUS='DELETE')

*-----initialize soil-data file                                     580
         CALL RDINIT (IUNITS, IUNITO, FILES)

*-----total nitrogen loss by leaching, kg/ha                        590
         TNLCH = 0.

*-----total nitrogen loss by volatilization, kg/ha                  600
         TNVOL = 0.

         CALL RDSREA ('REFCF'  ,REFCF)
         CALL RDSREA ('STBC'   ,STBC)
         CALL RDSREA ('GAMMA'  ,GAMMA)
         CALL RDSREA ('LHVAP'  ,LHVAP)
         CALL RDSREA ('RUNOFC' ,RUNOFC)
         CALL RDSREA ('WCLIM'  ,WCLIM)
         CALL RDSREA ('PROP'   ,PROP)

         CALL RDAREA ('RADTBA' ,RADTBA ,IARL1 ,IRADTA)
         CALL RDAREA ('RADTBB' ,RADTBB ,IARL1 ,IRADTB)
         CALL RDAREA ('RADTBC' ,RADTBC ,IARL1 ,IRADTC)
         CALL RDAREA ('RADTBD' ,RADTBD ,IARL1 ,IRADTD)
         CALL RDAREA ('RADTBE' ,RADTBE ,IARL1 ,IRADTE)
         CALL RDAREA ('RADTBF' ,RADTBF ,IARL1 ,IRADTF)
         CALL RDAREA ('RADTBG' ,RADTBG ,IARL1 ,IRADTG)
         CALL RDAREA ('RADTBH' ,RADTBH ,IARL1 ,IRADTH)
         CALL RDAREA ('RADTBI' ,RADTBI ,IARL1 ,IRADTI)
         CALL RDAREA ('RADTBJ' ,RADTBJ ,IARL1 ,IRADTJ)

         CALL RDSREA ('CONVER' ,CONVER)
         CALL RDSREA ('WSCONV' ,WSCONV)

         CALL RDAREA ('WREDT'  ,WREDT  ,IARL1 ,IWREDT)
         CALL RDAREA ('EDPTFT' ,EDPTFT ,IARL1 ,IEDPTN)
         CALL RDAREA ('REDFDT' ,REDFDT ,IARL1 ,IRDFDT)
         CALL RDAREA ('IRRT'   ,IRRT   ,IARL1 ,IIRRT)
         CALL RDAREA ('ROSPT'  ,ROSPT  ,IARL1 ,IROSPT)

*-----total seasonal soil surface evaporation, mm                   610
         TEVAP = 0.

*-----total seasonal infiltration, mm                               620
         TOTINF = 0.

*-----total water loss by drainage beyond potential rooting         630
*     zone, mm
         TDRAIN = 0.

*-----total nitrogen fertilizer application, kg/ha                  640
         TNTRMT = 0.

*-----relative weight of nitrate in total inorganic                 650
*     nitrogen, stored in top soil compartment, kg/ha
         RANO3 = 0.

         CALL RDSINT ('ILAYER' ,ILAYER)
         CALL RDAREA ('THCKN'  ,THCKN ,IARL1 ,ITHCKN)
         CALL RDAREA ('WLTPT'  ,WLTPT ,IARL1 ,IWLTPT)
         CALL RDAREA ('FLDCP'  ,FLDCP ,IARL1 ,IFLDCP)

         CALL RDAREA ('DRFA'   ,DRFA  ,IARL1 ,IDRFA)
         CALL RDSREA ('FOMI'   ,FOMI)
         CALL RDSREA ('HUMI'   ,HUMI)
         CALL RDSREA ('NHUMI'  ,NHUMI)

         CALL RDAREA ('IBIOMN' ,IBIOMN ,IARL1   ,IIBIOM)
         CALL RDAREA ('IAS'    ,IAS    ,IARL1   ,IIAS)
         CALL RDAREA ('DFFOM'  ,DFFOM  ,IARL1+1 ,IDFFOM)

*-----depths of top of consecutive compartments, mm                 660
         TDT(1) = 0.
         DO 20 I = 2,ILAYER+1
            TDT(I) = TDT(I-1)+THCKN(I-1)
20       CONTINUE

*-----initial weight of inorganic nitrogen in total soil            670
*     profile, kg/ha
         TNINT = 0.
         DO 30 I = 1,ILAYER
            TNINT = TNINT+IAS(I)
30       CONTINUE

*-----calculation of total initial weight of nitrogen in            680
*     microbial biomass, kg/ha
         BIOMNI = 0.
         DO 40 I = 1,ILAYER
            BIOMNI = BIOMNI+IBIOMN(I)
            DBIOMN(I) = 0.
40       CONTINUE

         CALL RDSREA ('RDLIGN' ,RDLIGN)
         CALL RDSREA ('RDCELL' ,RDCELL)
         CALL RDSREA ('RDCAPR' ,RDCAPR)
         CALL RDSREA ('FRNF'   ,FRNF)

*-----initial weights of: water, available water, storage capacit  690
*     fresh organic matter, nitrogen in fresh organic matter, nitrogen
*     in stable organic matter, stable organic matter and nitrogen in
*     microbial biomass in soil compartments, units respectively;
*     mm, mm, m**3/m**3, 7(kg/ha)
         DO 50 I = 1,ILAYER
            WATER(I) = DRFA(I)*WLTPT(I)*THCKN(I)
            AWATER(I) = AMAX1(0.,WATER(I)-THCKN(I)*WLTPT(I))
            STORC(I) = FLDCP(I)-WLTPT(I)
            IFOM(I)  = DFFOM(I)*FOMI
            FOM(I)   = IFOM(I)
            FON(I)   = IFOM(I)*FRNF
            NHUM(I)  = DFFOM(I)*NHUMI
            HUM(I)   = DFFOM(I)*HUMI
            BIOMN(I) = IBIOMN(I)
            ASLT(I)  = IAS(I)
50       CONTINUE

*-----total amount of soil moisture in the total soil profile,      700
*     mm
      WTOT = 0.

*-----total weight of inorganic nitrogen in the total soil          710
*     profile, kg/ha
      NTOT = 0.

*-----total weight of nitrogen in microbial biomass in soil         720
*     profile, kg/ha
      BIOMNT = 0.

*-----total weight of fresh organic material in the total soil      730
*     profile, kg/ha
      TFOM = 0.

*-----total weight of nitrogen in fresh organic material in         740
*     the total soil profile, kg/ha
      TFON = 0.

*-----total weight of stable organic material in the total          750
*     soil profile, kg/ha
      HUMT = 0.

*-----total weight of nitrogen in stable organic material in        760
*     the total soil profile, kg/ha
      NHUMT = 0.

      DO 60 I = 1,ILAYER
         IF (TDT(I)+0.5.GT.MXRTD) GO TO 60

*-----total weight of soil moisture in the total soil profile,      770
*     mm
         WTOT = WTOT+WATER(I)

*-----total weight of inorganic nitrogen in the total soil          780
*     profile, kg/ha
         NTOT = NTOT+ASLT(I)

*-----total weight of nitrogen in microbial biomass in soil         790
*     profile, kg/ha
         BIOMNT = BIOMNT+BIOMN(I)

*-----total weight of fresh organic material in the total           800
*     soil profile, kg/ha
         TFOM = TFOM+FOM(I)

*-----total weight of nitrogen in fresh organic material            810
*     in the total soil profile, kg/ha
         TFON = TFON+FON(I)

*-----total weight of stable organic material in the total          820
*     soil profile, kg/ha
         HUMT = HUMT+HUM(I)

*-----total weight of nitrogen in stable organic material           830
*     in the total soil profile, kg/ha
         NHUMT = NHUMT+NHUM(I)
60    CONTINUE

         CALL RDSREA ('NCR'    ,NCR)
         CALL RDSREA ('NCIW'   ,NCIW)
         CALL RDSREA ('CNRMIC' ,CNRMIC)
         CALL RDSREA ('DMINR'  ,DMINR)
         CALL RDSREA ('FCELL'  ,FCELL)
         CALL RDSREA ('FLIGN'  ,FLIGN)
         CALL RDSREA ('FNIMH'  ,FNIMH)
         CALL RDSREA ('FRC'    ,FRC)
         CALL RDSREA ('LNH4'   ,LNH4)
         CALL RDSREA ('MRGRB'  ,MRGRB)
         CALL RDSREA ('NCH'    ,NCH)
         CALL RDSREA ('NH4FP'  ,NH4FP)
         CALL RDSREA ('RRMIC'  ,RRMIC)
         CALL RDSREA ('TCMG'   ,TCMG)
         CALL RDSREA ('TCN'    ,TCN)
         CALL RDSREA ('TCV'    ,TCV)

         CALL RDAREA ('IRRT'   ,IRRT   ,IARL1 ,IIRRT)
         CALL RDAREA ('MFT'    ,MFT    ,IARL1 ,IMFT)
         CALL RDAREA ('DISTF'  ,DISTF  ,IARL1 ,IDISTF)

         CALL RDSINT ('NF'     ,NF)
         CALL RDAREA ('NAPDAY' ,NAPDAY,IARL1 ,IAPDAY)
         CALL RDAREA ('NTRMNT' ,NTRMNT,IARL1 ,INTRMT)
```

```
*-----    check on data of nitrogen applications rate and date           850
          IF (NF.NE.IAPDAY.OR.NF.NE.INTRMT) THEN
             WRITE (*,'(A,/,A)')
     $          ' There have been errors in the file: SOIL.DAT',
     $          ' Check number of data on nitrogen management'
             STOP
          END IF

*------   initial number of nitrogen application, -                      855
          NNA = 1

*-----    carbon balance, kg/ha                                          860
          CARBAL = 0.

*-----    nitrogen balance, kg/ha                                        870
          NBAL = 0.

*-----    initial soil temperature, gr.c                                 880
          TSI = 5.*(TMAX + TMIN)

*-----    soil temperature, gr.c                                         890
          TS = 0.1*TSI

*-----    initialization of array with soil temperatures, gr.c           900
          DO 70 I = 1,12
             TMPCX(I) = 0.
70        CONTINUE

          CALL RDAREA('TFT'   ,TFT  ,IARL1 ,ITFT)
          CALL RDAREA('TECT'  ,TECT ,IARL1 ,ITECT)

*-----    potential daily gross assimilation of the canopy               910
*         expressed in ch2o, kg/ha/d
          DTGA = 0.

*-----    rate of canopy gross assimilation in ch2o, kg/ha/d             920
          DGAS = 0.

*-----    counter to designate leaf classes by day of initiation,        930
          LG = 0
          CLOSE (IUNITS, STATUS='DELETE')

*-----end of initialization

*-----rate calculation section

      ELSE IF (ITASK.EQ.2) THEN

*-----procedure to estimate emergence date as a function of              950
*     soil wetness in top 3 soil layers to a depth of 100 mm.

*-----pushd is a switch variable, assuming value 1. at day of
*     emergence otherwise 0., used for initialization of
*     vegetation state variables

      IF ( DATE.GE.SOWD ) THEN

          IF (PUSHD.GT.0. .OR. TGERD.GE.9.) THEN
             PUSHD = 0.
             TGERD = 100.
             GO TO 500
          END IF

          WGER = 0.

          DO 450 I = 1,3
             WGER = WGER + WATER(I)
 450      CONTINUE

          GERD = 0.
          SDAY = 0.

          IF (WGER.GT.100.*WLTPT(1)*1.2) THEN
             GERD = 1.
          ELSE IF (GERD.EQ.0. .AND. TGERD.GT.4.) THEN
             SDAY = 1.
          END IF

          TGERD = TGERD + GERD
          TSDAY = TSDAY + SDAY

          IF (TSDAY.GT.6.) THEN
             TGERD = 0.
             TSDAY = 0.
          ELSE IF (TGERD.GT.5.) THEN
             PUSHD = 1.
             GERDAT = DATE
          END IF

      END IF

500   CONTINUE

*-----if emergence conditions occur, initialize vegetation                970
*     state variables with initial values
      IF (NINT(PUSHD).EQ.1) THEN

          CALL RDINIT (IUNITP, IUNITO, FILEP)

*-----weight of live leaf blades at emergence, kg/ha                      980
          CALL RDSREA ('WLVSI'  ,WLVSI)
          WLVSI = SWDF*WLVSI

*-----weight of live roots at emergence, kg/ha                            990
          CALL RDSREA ('WRTI'   ,WRTI)
          WRTI = SWDF*WRTI

*-----initial rooting depth at emergence, mm                             1000
          CALL RDSREA ('IRTD'   ,IRTD)

*-----initial green area of leaf blades at emergence, m**2/ha            1010
          ARLFI = WLVSI*FLFARI

*-----initial weight of reserve carbohydrates in the plant,              1020
*     kg/ha
          ARESPI = RESLI*(WLVSI+WRTI)

*-----initial tiller density, no/ha                                      1030
          CALL RDSREA ('TLNI'   ,TLNI)
          TLNI = SWDF*TLNI

*-----auxilary variable to save original value of tlni                   1040
          TLNIX = TLNI

*-----initial weight of live stem and sheaths at emergence,              1050
*     kg/ha
          WSTEM = 0.

*-----initial weight of the grains at emergence, kg/ha                   1060
          WGR = 0.

*-----initial weight of nitrogen in leaf blades at emergence,            1070
*     kg/ha
          ANLVI = WLVSI*FRNVI

*-----initial weight of nitrogen in the roots at emergence,              1080
*     kg/ha
          ANRTI = WRTI*IFNRT
          CLOSE (IUNITP, STATUS='DELETE')

      END IF

*-----conversion of daily total global radiation from kj/m**2            1090
*     to j/m**2
      RAD = RADCF*RADI

*-----effective air temperature during daylight period, gr.c             1100
      EAVT = TMAX-0.25*(TMAX-TMIN)

*-----modification of rain intensity, mm/d                               1110
      RAIN = RAIN*RAINF

*-----subroutine astro computes day length and daily radiation           1120
*     characteristics from julian day and latitude
      CALL ASTRO (DATE, LAT,
     $            SC, DS0, SINLD, COSLD, DAYL, DSINB, DSINBE)

*-----average daily air temperature, gr.c                                1170
      TMPA = (TMAX+TMIN)*0.5

*-----average daily saturated vapour pressure of the atmosphere,         1180
*     mbar
      SVPA = 4.58*EXP(17.4*TMPA/(TMPA+239.))

*-----actual average daily vapour pressure of the atmosphere,            1190
*     conversion from kpa to mm hg, mm hg
      VPA = 1000.*VAPOUR/133.28

*-----average daily dew point temperature, amin1 to avoid                1200
*     condensation conditions, gr.c
      DPT = AMIN1(239./(17.381/LOG(VAPOUR/0.6107)-1.),TMPA)

*-----average "effective" saturated vapour pressure during               1210
*     daylight, mbar
      SVPAM =6.11*EXP(17.4*EAVT/(EAVT+239.))

*-----actual vapour pressure during daylight, mbar                       1220
      VPAM = AMIN1(SVPAM-0.1,1.33*VPA)

*-----average daily wind speed during daylight, cm/d                     1230
      WSA = 1.333E5*WSCONV*WIND

*-----total visible radiation under a clear sky as a function            1240
*     of latitude, j/m**2
      IF (LAT.GE.0..AND.LAT.LT.10.) THEN
         A = LINT(RADTBA,IRADTA,DATE)
         B = LINT(RADTBB,IRADTB,DATE)
         RADCL = A+(LAT)*(B-A)/10.
      ELSE IF (LAT.GE.10..AND.LAT.LT.20.) THEN
         A = LINT(RADTBB,IRADTB,DATE)
         B = LINT(RADTBC,IRADTC,DATE)
         RADCL = A+(LAT-10.)*(B-A)/10.
      ELSE IF (LAT.GE.20..AND.LAT.LT.30.) THEN
         A = LINT(RADTBC,IRADTC,DATE)
         B = LINT(RADTBD,IRADTD,DATE)
         RADCL = A+(LAT-20.)*(B-A)/10.
      ELSE IF (LAT.GE.30..AND.LAT.LT.40.) THEN
         A = LINT(RADTBD,IRADTD,DATE)
         B = LINT(RADTBE,IRADTE,DATE)
         RADCL = A+(LAT-30.)*(B-A)/10.
      ELSE IF (LAT.GE.40..AND.LAT.LT.50.) THEN
         A = LINT(RADTBE,IRADTE,DATE)
         B = LINT(RADTBF,IRADTF,DATE)
         RADCL = A+(LAT-40.)*(B-A)/10.
      ELSE IF (LAT.GE.50..AND.LAT.LT.60.) THEN
         A = LINT(RADTBF,IRADTF,DATE)
         B = LINT(RADTBG,IRADTG,DATE)
         RADCL = A+(LAT-50.)*(B-A)/10.
      ELSE IF (LAT.GE.60..AND.LAT.LT.70.) THEN
         A = LINT(RADTBG,IRADTG,DATE)
         B = LINT(RADTBH,IRADTH,DATE)
         RADCL = A+(LAT-60.)*(B-A)/10.
      ELSE IF (LAT.GE.70..AND.LAT.LT.80.) THEN
         A = LINT(RADTBH,IRADTH,DATE)
         B = LINT(RADTBI,IRADTI,DATE)
         RADCL = A+(LAT-70.)*(B-A)/10.
      ELSE IF (LAT.GE.80..AND.LAT.LT.90.) THEN
         A = LINT(RADTBI,IRADTI,DATE)
         B = LINT(RADTBJ,IRADTJ,DATE)
         RADCL = A+(LAT-80.)*(B-A)/10.
      END IF

*-----daily total global radiation under a completely clear              1250
*     sky, j/m**2/d
      DGRCL = 2.*RADCL*CONVER

*-----daily total global radiation under an overcast sky,                1260
*     j/m**2/d
      DGROV = 0.2*DGRCL

*-----fraction of the day that sky is overcast, restrained               1270
*     between zero and one, -
      LFOV = LIMIT(0.,1.,1.-(RAD-DGROV)/(DGRCL-DGROV+
     $       REANOT(DGRCL-DGROV)))

*-----net outgoing long wave radiation used in penman equation            1280
*     j/m**2/d
      LWR = STBC*(TMPA+273.)**4*(0.58-0.09*SQRT(VPA))*(1.-0.9*LFOV)

*-----net outgoing long wave radiation, j/m**2/d                          1290
```

```
*      ELWR = STBC*(EAVT+273.)**4*(0.58-0.09*SQRT(VPA)) *
*     $           (1.-0.9*LFOV)*DAYL/24.

      ELWR = STBC*(EAVT+273.)**4*(0.58-0.09*SQRT(VPAM)) *
     $           (1.-0.9*LFOV)*DAYL/24.

*-----net global radiation, j/m**2/d                               1300
      HNOT = AMAX1(0.,0.75*RAD-ELWR)

*-----slope of saturated vapour pressure curve at 'effective'      1310
*     day time temperature, mbar/degree c
      SLOPE = 17.4*SVPAM*(1.-EAVT/(EAVT+239.))/(EAVT+239.)

*-----average hourly radiation intensity during daylight hours,    1320
*     j/m**2/h
      HRAD = RAD/DAYL

*-----alpha = twovar(alphat,hrad,lai)                              1330
      IF (LAI.GE.0..AND.LAI.LE.0.2) THEN
         A = LINT (ALPHTA,IALPTA,HRAD)
         B = LINT (ALPHTB,IALPTB,HRAD)
         ALPHA = A+(LAI)*(B-A)/0.2
      ELSE IF (LAI.GT.0.2.AND.LAI.LE.2.0) THEN
         A = LINT (ALPHTB,IALPTB,HRAD)
         B = LINT (ALPHTC,IALPTC,HRAD)
         ALPHA = A+(LAI-0.2)*(B-A)/1.8
      ELSE IF (LAI.GT.2.0.AND.LAI.LE.3.5) THEN
         A = LINT (ALPHTC,IALPTC,HRAD)
         B = LINT (ALPHTD,IALPTD,HRAD)
         ALPHA = A+(LAI-2.0)*(B-A)/1.5
      ELSE IF (LAI.GT.3.5.AND.LAI.LE.5.0) THEN
         A = LINT (ALPHTD,IALPTD,HRAD)
         B = LINT (ALPHTE,IALPTE,HRAD)
         ALPHA = A+(LAI-3.5)*(B-A)/1.5
      ELSE IF (LAI.GT.5.0.AND.LAI.LE.10.0) THEN
         A = LINT (ALPHTE,IALPTE,HRAD)
         B = LINT (ALPHTF,IALPTF,HRAD)
         ALPHA = A+(LAI-5.0)*(B-A)/5.0
      ELSE IF (LAI.GE.10.) THEN
         ALPHA = LINT(ALPHTF,IALPTF,HRAD)
      END IF

*-----calculation of daily gross co2 assimilation, procedure
*     based on spitters, van keulen and van kraalingen

*-----green area of the ears, m**2/ha                              1340
      EARAR = EARN*ARPEAR*LINT(AGEFT,IAGEFT,DVS)

*-----height of the crop, m                                        1350
      CROPHT = LINT(CROHTB,ICRHTB,DVS)

*-----green area of stems, m**2/ha                                 1360
      STAREA = WSTEM/5000.*CROPHT*MXSTAR

*-----total green area, including stems and ears, m**2/m**2        1370
      GRAI = 1.E-4*(ARLF+EARAR+STAREA)

*-----fraction of nitrogen in live leaf blade tissue, kg/kg        1380
      FRNV = ANLV/(WLVS+REANOT(WLVS))

*-----effective nitrogen concentration for assimilation            1390
*     taking into account heterogeneous distribution in
*     the plant, kg/kg
      EFRNV = INSW(DVSV-DVSAN,FRNV,INSW(NDPAR,FRNV,FRNV*HTFAC))

*-----calculate assimilation after completion of germination       1400
*     process
      IF (PUSHDI.GT.0) THEN

*-----light saturated co2 assimilation rate as a function          1410
*     of nitrogen concentration in the leaf blades and
*     temperature, kg/ha/h
      AMAXN = AMAX1(0.,(725.*EFRNV-2.75)*LINT(TMPFT,ITMPFT,EAVT))

*-----maximum rate of gross co2 assimilation of a single leaf,     1420
*     kg/ha/h
      AMAX = INSW(AMAXN,PAMAX,AMAXN)

*-----maximum rate of gross co2 assimilation of a single leaf,     1430
*     one time interval ago, kg/ha/h
      PAMAX = AMAX

*-----subroutine totass is called to calculate daily total gross   1440
*     assimilation (dtga)
      CALL TOTASS (SC, DAYL, SINLD, COSLD, DSINBE, RAD,
     $             SCV, AMAX, EFFE, KDIF, GRAI, DTGA)
      END IF

*-----residual level of nitrogen in leaf blades, not to be         1450
*     translocated, kg/kg
      BN = LINT(BNT,IBNT,DVS)

*-----slope of optimum nitrogen concentration in tissue vs.        1460
*     development rate, -
      SONCT = AMAX1(0.,1.-DVS/((1.+DVSEGF)/2.))

*-----maximum fraction of nitrogen in leaf blade tissue, kg/kg     1470
      FNMX = FNMXR + SONCT*FNMXA

*-----relative nitrogen concentration, to account for nitrogen     1480
*     shortage on tiller formation and leaf blade growth, -
      FNC = LIMIT(0.,1.,(FRNV-BN)/(FNMX-BN))

*-----reduction factor for growth of leaf blades due to nitrogen   1490
*     stress, -
      RFNS = LINT(RFNST,IRFNST,FNC)

*-----turbulent resistance for water vapour exchange, d/cm         1500
      RTURB = 0.

*-----potential transpiration
      IF (GRAI.GT.1.E-10) THEN

*-----mixing length, -                                             1510
         LMIX = SQRT(4.*0.02*CROPHT/(PI*GRAI))

*-----intermediate variable, -                                     1520
         ALPH = SQRT(DRAGC*GRAI*CROPHT/(2.*LMIX*IW))

*-----zero plane displacement, m                                   1530
         D    = CROPHT-SQRT(LMIX*IW*CROPHT/ALPH)/KARMAN

*-----zero plane displacement, m                                   1540
         ZNOT = (CROPHT-D)*EXP(-CROPHT/(ALPH*(CROPHT-D)))

*-----intermediate variable, -                                     1550
         LNREF = ALOG((REFHT-D)/ZNOT)

*-----intermediate variable, -                                     1560
         REST  = 0.74*LNREF*LNREF/(KARMAN*KARMAN*1.1574E-7*
     $              WSA+REANOT(WSA))

*-----turbulent resistance for water vapour exchange, d/cm         1570
         RTURB = REST/864.E4
      END IF

*-----total resistance for water vapour transport from canopy      1580
*     to the atmosphere, sum of laminar resistance and
*     turbulent resistance, d/cm
      RA = 4.41E-3*SQRT(1./(WSA+REANOT(WSA)))+RTURB*GRAI

*-----actual stomatal conductance, as determined by nitrogen       1590
*     status of the vegetation, cm/d
      SC = SCM*RFNS

*-----actual stomatal resistance, applied in calculation of        1600
*     potential crop transpiration. (/rfns, to account for
*     effect of n-deficiency on rs, radin and parker, 1981, d/cm
      RS = 1./(SC+REANOT(SC))

*-----intermediate variable in calculation of potential crop       1610
*     transpiration, -
      S = (RA+RS)/RA

*-----intermediate variable in calculation of potential crop       1620
*     transpiration, gr.c/mbar
      CC = 1./(SLOPE+S*PSCH)

*-----potential crop transpiration, mm/d                           1630
      APTRAN = CC*((1.-EXP(-EXC*GRAI))*HNOT*SLOPE+ALPHA*GRAI*RHOCP
     $         /RA*(SVPAM-VPAM)*DAYL/24.)/LHVAP

*-----calculation of potential cuticular transpiration,            1640
*     applied in determining leaf death due to water stress, -
      S1 = (RA+RC)/RA

*-----intermediate variable in calculation of cuticular water      1650
*     loss, gr.c/mbar
      CC1 = 1./(SLOPE+S1*PSCH)

*-----rate of cuticular water loss with fully closed stomata,      1660
*     mm/d
      PCTRAN = APTRAN*CC1/CC

*-----potential cuticular transpiration, under the assumption      1670
*     of fully closed stomata

* actual transpiration

*-----weight to conductivity ratio of the root system, assuming    1680
*     an average age distribution of the root system, 200 kg
*     of root dry matter is needed per hectare to take up 1 mm
*     of water per day

*-----calculation of potential soil evaporation (penman)

*-----wind run, km/d                                               1690
      WSR = WSCONV*WIND

*-----net absorbed radiation by open water surface, j/m**2/d       1700
      HZERO = RAD*(1.-REFCF)-LWR

*-----drying power term in penman equation, j/m**2/d               1710
      EA = 0.35*(SVPA-VPA)*(0.5+WSR/1.6/100.)*LHVAP

*-----slope of the saturated vapour pressure curve at air          1720
*     temperature, mm hg/gr c
      DELTA = 17.4*SVPA*(1.-TMPA/(TMPA+239.))/(TMPA+239.)

*-----potential rate of soil surface evaporation, mm/d             1730
      EVAP = (HZERO*DELTA/GAMMA+EA)/(1.+DELTA/GAMMA)*1./LHVAP

*-----calculation of actual soil evaporation

*-----potential soil evaporation as function of soil               1740
*     cover, mm/d
      PEVAP1 = EVAP*EXP(-5.E-5*ARLFE)
      PEVAP = INSW(-RAIN,INSW(PEVAP1-RAIN,PEVAP1,PEVAP1-RAIN),PEVAP1)

*-----irrigation rate, mm/d                                        1745
      IRRID = LINT(IRRT,IIRRT,DATE)

*-----infiltration rate, rainfall and irrigation, runoff           1750
*     introduced via fixed fraction of rainfall, mm/d
*     INFR = (1.-RUNOFC)*RAIN + IRRID
      INFR = (1.-RUNOFC)*RAIN*INSW(PEVAP-PEVAP1,0.,1.) + IRRID

*-----rate of water flow through the top of soil                   1760
*     compartments, mm/d
      RWF(1) = INFR
      DO 150 I = 2,ILAYER+1
       RWF(I)=AMAX1(0.,RWF(I-1)-(FLDCP(I-1)*THCKN(I-1)-WATER(I-1))/DELT)
       IF (TDT(I).GE.MXRTD.AND.TDT(I-1).LT.MXRTD) K = I
150   CONTINUE

*-----rate of drainage of soil moisture beyond potential           1770
*     rooting zone, mm/d
      RDRAIN = RWF(K)

*-----water content in top compartment, -                          1780
      WCPR = (WATER(1)/THCKN(1)-WCLIM)/(FLDCP(1)-WCLIM)

*-----potential soil evaporation as function of soil               1790
*     cover and dryness of top soil compartment, mm/d
      AEVAP = AMAX1(0.,PEVAP*LINT(REDFDT,IRDFDT,WCPR))

*-----change in water content, transpiration and soil evaporation

*-----maximum amount of water that can be stored in                1800
*     rooted depth, mm
```

```
      MWRTD = 0.

      DO 160 I = 1,ILAYER

*-----available water in a soil compartment, mm                      1810
         AWATER(I) = AMAX1(0.,WATER(I)-THCKN(I)*WLTPT(I))

*-----root length in a soil compartment, mm                          1820
         RTL(I) = LIMIT(0.,THCKN(I),RTD-TDT(I))*INSW(TDT(I)-MXRTD,1.,0.)

*-----reduction factor for soil moisture uptake due to               1830
*     low available moisture in a soil compartment, -
         EDPTF(I) = LINT(EDPTFT,IEDPTN,AWATER(I)/(THCKN(I)*STORC(I)))

*-----reduction factor for the effect of low soil moisture           1840
*     on moisture uptake for crop transpiration, -
         WRED(I)  = LINT(WREDT,IWREDT,AWATER(I)/(THCKN(I)*STORC(I)))

*-----factor indicating presence (1) or absence (0) of               1850
*     moisture available to a plant in a soil compartment, -
         AWATF(I) = INSW(-AWATER(I),1.,0.)

*-----maximum amount of water that can be stored in                  1860
*     rooted depth, mm
         MWRTD    = MWRTD+RTL(I)*STORC(I)
160   CONTINUE

*-----total 'effective' root length, mm                              1870
      ERLT = 0.

*-----relative amount of available water in root zone, mm           1880
      RWRBT = 0.
      DO 170 I = 1,ILAYER
         IF (TDT(I)+0.5.GT.MXRTD) GO TO 170
         ERLT = ERLT+RTL(I)*EDPTF(I)
         RWRBT = RWRBT+RTL(I)/THCKN(I)*AWATER(I)/(MWRTD+REANOT(MWRTD))
170   CONTINUE

*-----potential rate of moisture uptake per unit                     1890
*     'effective' root length, mm/d/mm
      TRPMM = APTRAN/(ERLT+REANOT(ERLT))

*-----actual rate of canopy transpiration, mm/d                      1900
      TRANW = 0.

*-----actual rate of moisture uptake for transpiration
*     in a soil compartment, mm/d
      DO 180 I = 1,ILAYER
      TRR(I) =TRPMM*RTL(I)*EDPTF(I)*WRED(I)*LINT(ROSPT,IROSPT,CONC(I))
      TRANW = TRANW+TRR(I)
180   CONTINUE

*-----intermediate variable in calculation of soil                   1910
*     surface evaporation
      SUMT = 0.
      DO 190 I = 1,ILAYER
         VAR(I) = AMAX1(WATER(I)/THCKN(I)-WCLIM,0.)*EXP(-PROP*0.001*
     $            (TDT(I)+0.5*THCKN(I)))
         SUMT = SUMT+VAR(I)*THCKN(I)
190   CONTINUE

*-----switch variable to indicate whether root tip is in             1920
*     wet soil compartment (1) or not (0), -
      SWPBT = 0.

*-----rate of soil surface evaporation, mm/d                         1930
      EVTOT = 0.
      DO 200 I = 1,ILAYER

*-----rate of moisture extraction from a soil compartment            1940
*     due to soil surface evaporation, mm/d
        ER(I) = AMIN1(AMAX1(WATER(I)-WCLIM*THCKN(I),0.),AEVAP*THCKN(I)*
     $          VAR(I) /(SUMT+REANOT(SUMT)))
        SWPBT = SWPBT+AWATF(I)*REAAND(RTD-TDT(I),TDT(I+1)-RTD)
        EVTOT = EVTOT+ER(I)
200   CONTINUE

*-----factor accounting for the effect of soil temperature           1950
*     on root conductivity, -
      TEC = LINT(TECT,ITECT,TS)

*-----actual conductivity of the root system, as determined          1960
*     by root weight and soil temperature, mm/d
      ACOND = WRT/(LINT(WCRRT,IWCRRT,DVS))*TEC

*-----actual rate of transpiration of the vegetation, mm/d           1970
      TRAN = AMIN1(TRANW,ACOND)

*-----rate of change in amount of water in a soil                    1980
*     compartment, mm/d
      DO 210 I = 1,ILAYER
        DWAT(I) = RWF(I)-RWF(I+1)-TRR(I)*TRAN/(TRANW+REANOT(TRANW)) -
     $            ER(I)
210   CONTINUE

*-----relative transpiration deficit, -                              1990
      RTRDEF = (APTRAN-TRAN)/(APTRAN+REANOT(APTRAN))

*-----reserve level in the plant, kg/kg                              2000
      RESL = ARESP/(ARESP+WLVS+WRT+WSTEM+REANOT(ARESP+WLVS+WRT+
     $       WSTEM))

*-----reduction factor for gross assimilation due to                 2010
*     accumulation of reserve substances, -
      REDFRL = 1.-LIMIT(0.,1.,(RESL-TLRGA)/0.05)

*-----daily gross co2 assimilation and conversion from               2020
*     assimilated co2 to ch2o, kg/ha/d
      DGAS = DTGA*CCO2TS*TRAN/(APTRAN+REANOT(APTRAN))*REDFRL

*-----rate of soil surface evaporation from wet soil                 2030
*     surface in presence of crop, used for calculation of
*     average daily canopy temperature, mm/d
      EVAPR = EVAP*(1.-EXP(-0.5*GRAI))

*-----average daily canopy temperature, gr.c                         2040
      TMPC = TMPA+CTEMPF*(1.-FTMPA*TMPA*TRAN/(EVAPR+REANOT(EVAPR)))

*-----fertilizer application rate, kg/ha/d                           2050
      APFERT = 0.
      IF (NNA.LE.IAPDAY.AND.TIME.GE.NAPDAY(NNA)) THEN
         APFERT = NTRMNT(NNA)
         NNA = NNA+1
      END IF

*-----decomposition of organic material                              2060

*-----factor accounting for the effect of soil temperature
*     on decomposition of organic material, -
      TF = LINT(TFT,ITFT,TS)

*-----growth efficiency of microbial biomass, kg/kg                  2070
      YG = 0.25*EFCPR + 0.75*EFCCH

*-----calculation of rate of nitrogen immobilization,                2080
*     kg n/ha/d
      DO 220 I = 1,ILAYER

*-----carbon to nitrogen ratio of fresh organic material,            2090
*     kg/kg
         CNR(I) = FRC*FOM(I)/(FON(I)+ASLT(I)+REANOT(FON(I)+ASLT(I)))

*-----factor accounting for the effect of c/n ratio of               2100
*     fresh organic material on rate of decomposition, -
         CNRF(I) = AMIN1(1.,EXP(-.693*(CNR(I)-25.)/25.))

*-----factor accounting for the effect of soil moisture on           2110
*     rate of decomposition of organic material, -
         MF(I) = LINT(MFT,IMFT,AWATER(I)/(STORC(I)*THCKN(I)))

*-----fraction of fresh organic material not yet decomposed,         2120
*     kg/kg
         FOMRES(I) = FOM(I)/(IFOM(I)+REANOT(IFOM(I)))

*-----rate of decomposition of fresh organic material,               2130
*     kg/ha/d
         RDECR(I) = INSW(FOMRES(I)-FLIGN,RDLIGN,
     $              INSW(FOMRES(I)-(FLIGN+FCELL),RDCELL,RDCAPR))

*-----maximum rate of decomposition of fresh organic                 2140
*     material, kg/ha/d
         LDEC = BIOMN(I)*CNRMIC/FRC*(MRGRB/YG+RRMIC)

*-----rate of decomposition of fresh organic material,               2150
*     kg/ha/d
         DECR(I) = AMIN1(LDEC,FOM(I)*RDECR(I)*TF*CNRF(I)*MF(I))

*-----rate of nitrogen mineralization through decomposing            2160
*     fresh organic material, kg/ha/d
         RNRL(I) = DECR(I)*FON(I)/(FOM(I)+REANOT(FOM(I)))

*-----rate of nitrogen mineralization from stable organic            2170
*     material, kg/ha/d
         RHMIN(I) = NHUM(I)*DMINR*TF*MF(I)
220   CONTINUE

*-----rate of change in nitrogen in microbial biomass,               2180
*     kg/ha/d
      DO 230 I = 1,ILAYER

*-----rate of carbon release through decomposition of               2190
*     organic material, kg/ha/d
         CADEC(I) = 0.4*DECR(I)+10.*RHMIN(I)

*-----rate of release or immobilization of carbon due               2200
*     to changes in microbial biomass, kg/ha/d
         CFBMG(I) = INSW(DBIOMN(I),DBIOMN(I)*CNRMIC,DBIOMN(I)*CNRMIC/YG)

*-----maximum weight of carbon in microbial biomass,                 2210
*     limited by carbon availability, kg/ha
         BIOMXC(I) = AMAX1(CADEC(I)-CFBMG(I),0.5*CADEC(I))/RRMIC

*-----maximum weight of carbon in microbial biomass,                 2220
*     limited by nitrogen availability, kg/ha
         BIOMXN(I) = (ASLT(I)+BIOMN(I))*CNRMIC

*-----maximum weight of carbon in microbial biomass, limited         2230
*     by nitrogen or carbon availability, kg/ha
         BIOMX(I) = AMIN1(BIOMXC(I),BIOMXN(I),
     $              BIOMN(I)*CNRMIC*(1.+MRGRB))

*-----rate of change in nitrogen in microbial tissue,  not           2240
*     corrected, kg/ha/d
         DBN(I) = (BIOMX(I)/CNRMIC-BIOMN(I))/TCMG

*-----rate of change in nitrogen in microbial tissue,                2250
*     kg/ha/d
         DBIOMN(I) = INSW(DBN(I),DBN(I)*DELT*RRMIC,DBN(I))

*-----rate of nitrogen mineralization from dying micro-              2260
*     organisms, kg/ha/d
         RNRLB(I) = -1.*AMIN1(0.,DBIOMN(I))

*-----rate of nitrogen immobilization by soil microbes,              2270
*     kg/ha/d
         RNAC(I) = AMAX1(0.,DBIOMN(I))
230   CONTINUE

*-----factor accounting for start of grain fill, -                   2280
      SGFF = INSW(DVSR-DVSSGF,0.,1.)

*-----potential rate of nitrogen accumulation in the grain,          2290
*     kg n/ha/d
      PRNAGR = GRN*LINT(PRNAGT,IPRAGT,TMPC)*SGFF

*-----"available" nitrogen in the leaves for translocation,          2300
*     kg/ha
      AVNLV = AMAX1(0.,ANLV-WLVS*BN)

*-----residual fraction of nitrogen in stem tissue, non-             2310
*     transferable, kg/kg
      LN = LINT(LNT,ILNT,DVS)

*-----"available" nitrogen in the stem and sheaths for               2320
*     translocation, kg/ha
      AVNSTE = AMAX1(0.,ANSTE-WSTEM*LN)

*-----"available" nitrogen in roots for translocation,               2330
*     kg/ha
      AVNRT = AMAX1(0.,ANRT-WRT*RN)
```

```
*-----total nitrogen available for translocation, kg/ha              2340
      AVN = AVNLV+AVNSTE+AVNRT

*-----average fraction of nitrogen in vegetative tissue,             2350
*     kg/kg
      FNVEG = (ANRT+ANSTE+ANLV)/(WRT+WSTEM+WLVS+REANOT(WLVS+WRT+WSTEM))

*-----fraction of nitrogen in the grain, kg/kg                       2360
      FRNG = ANGR/(WGR+REANOT(WGR))

*-----development stage in post-anthesis phase one time              2370
*     interval ago, -
      DVSRP = DVSRX
      DVSRX = DVSR

*-----effect of temperature on maintenance respiration, -            2380
      TEF = Q10**(0.1*TMPC-2.0)

*-----effect of moisture stress on rate of nitrogen                  2390
*     turnover, lal and sharma, 1973, -
      EFFWS = INSW(-TRAN,INSW(PCTRAN-TRAN,1.,2.),0.)

*-----relative rate of nitrogen turnover in vegetative               2400
*     tissue, turnover assumed to increase after
*     exhaustion of reserve carbohydrates, d**-1
      RTORT = RRTORT*TEF*EFFWS*INSW(-ARESP,1.,1.2)

*-----rate of nitrogen export from vegetative tissue,                 2410
*     kg/ha/d
      RNEXP = AMAX1(0.,RTORT*AVN*LINT(FNEXT,IFNEXT,FNVEG))

*-----switch variable for start of grain fill                        2420
      PUSHN = REAAND(DVSR-DVSSGF,DVSSGF-DVSRP)

*-----rate of nitrogen translocation to the grain,                   2430
*     kg n/ha/d
      RNTG = INSW(FRNG-0.035,AMIN1(PRNAGR,RNEXP),0.)*(1.-PUSHN)

*-----rate of transfer of nitrogen from the leaves to the            2440
*     grain, kg n/ha/d
      RNTLS = RNTG*AVNLV/(AVN+REANOT(AVN))

*-----energy gain from degradation of proteins prior to              2450
*     export, kg/ha/d, penning de vries, 1982
      SFPRD = 0.5*RNTG*6.25

*-----rate of maintenance respiration of leaf blades,                2460
*     kg/ha/d
      RMNLVS = WLVS*RMRESL*((FRNV-FNMIN)/RNFAC+1.)*TEF-
     $         SFPRD*WLVS/(TVEGM+REANOT(TVEGM))

*-----rate of maintenance respiration for grain, kg/ha/d             2470
      RMNG = WGR*RMRESG*((FRNG-FNMIN)/RNFAC+1.)*TEF
     $       *LIMIT(0.,1.,3.-3.*DVSR)

*-----fraction of nitrogen in root tissue, kg/kg                     2480
      FRNR = ANRT/(WRT+REANOT(WRT))

*-----rate of maintenance respiration for root tissue,               2490
*     kg/ha/d
      RMNR = WRT*RMRESR*((FRNR-FNMIN)/RNFAC+1.)*TEF
     $       - SFPRD*WRT/(TVEGM+REANOT(TVEGM))

*-----fraction of nitrogen in stem/sheath tissue, kg/kg              2500
      FRNST = ANSTE/(WSTEM+REANOT(WSTEM))

*-----rate of maintenance respiration for stem tissue,               2510
*     kg/ha/d
      RMNST = WSTEM*RMRESS*((FRNST-FNMIN)/RNFAC+1.)*TEF
     $        -  SFPRD*WSTEM/(TVEGM+REANOT(TVEGM))

*-----"net" flow of carbohydrate, kg/ha/d                            2520
      FCHNX = DGAS-RMNLVS-RMNG-RMNR-RMNST

*-----"net" flow of carbohydrate i.e. difference between             2530
*     gross assimilation and maintenance respiration in
*     kg/ha/d limited to positive values
      FCHN = AMAX1(0.,FCHNX)

*-----fraction of assimilate diverted to root system                 2540
*     under optimum conditions, kg/kg
      FDSR = LINT(FDSRT,IFDSRT,DVS)

*-----fraction of net assimilate available for growth of             2550
*     leaf blades, kg/kg
      FTLVS = LINT(FTLVST,IFTLST,DVS)

*-----fraction of assimilate supply allocated to stem                 2560
*     and sheats, function of development stage, kg/kg
      FTSTE = LINT(FTSTET,IFSTET,DVS)

*-----reduction factor for growth of leaf blades, due                2570
*     to water stress, -
      RFWS = LINT(REDWST,IRDWST,RTRDEF)

*-----reduction factor for growth of leaf blades, due                2580
*     to either water or nitrogen stress, -
      RFSTRS = AMIN1(RFWS,RFNS)

*-----"surplus" carbohydrates originating from inhibition            2590
*     of growth of leaf blades, kg/ha
      SCHFLV = (FTLVS+FTSTE)*FCHN*(1.-RFSTRS)

*-----flow of carbohydrates available for growth of root             2600
*     system, kg/ha/d
      FCHTR = FDSR*FCHN+SCHFLV*FSCHG

*-----fraction protein in the roots, value one time                  2610
*     interval ago being used, kg/kg
      FPRT = 6.25*ANRT/(WRT+REANOT(WRT))

*-----rate of increase in weight of root system, kg/ha/d             2620
      GRRT = FCHTR*(FPRT*EFCPR+(1.-FPRT)*EFCCH)

*-----maximum nitrogen concentration in the roots, kg/kg             2630
      OFNRT = FNRTMN+(1.-DVS)*(FNRTMX-FNRTMN)

*-----factor indicating presence (1) or absence (0) of               2640
*     reserve carbohydrates, -
      ARESPF = INSW(-ARESP,1.,0.)

*-----nitrogen demand of root tissue, kg/ha/d                        2650
      DNRT = INSW(FRNR-OFNRT,((WRT+GRRT*DELT)*OFNRT-ANRT)/TCU,0.)
     $       *ARESPF

*-----non-remobilizable concentration of reserve carbo-              2670
*     hydrates in the vegetation for transfer to vegetative
*     structures, kg/kg
      RESLL = INSW(DVSV-DVSFE,RESL1,RESL2)

*-----rate of translocation of reserve carbohydrates to              2680
*     vegetative tissue, function of reserve level, nitrogen
*     status of the leaves and development stage of the
*     vegetation, kg/ha/d
      TRFRS = AMAX1(0.,RESL-RESLL)*(WSTEM+WLVS+WRT)/TCTR*
     $    INSW(FTLVS-0.10,0.,1.)*INSW(RFSTRS-0.99,0.,1.)

*-----flow of carbohydrate for growth of leaf blades,                2690
*     kg/ha/d
      FCHTLV = FTLVS*FCHN*RFSTRS+TRFRS

*-----fraction of proteins in leaf blade tissue, value               2700
*     one time interval ago used, kg/kg
      EPVC = 6.25*ANLV/(WLVS+REANOT(WLVS))

*-----rate of increase in weight of leaf blades, kg/ha/d             2710
      GRLVS = FCHTLV*(EPVC*EFCPR+(1.-EPVC)*EFCCH)

*-----maximum weight of nitrogen in leaf blade tissue,               2720
*     kg/ha
      ONLV = (WLVS+GRLVS*DELT)*FNMX

*-----nitrogen demand of live leaf blade tissue, kg/ha/d              2730
      NDEM = INSW(FRNV-FNMX,(ONLV-ANLV)/TCU,0.)*ARESPF

*-----maximum fraction of nitrogen in stem/sheath tissue,            2740
*     kg/kg
      OFNST = FNMNSR + SONCT*FNMXSA

*-----flow of carbohydrates to stem and sheaths, kg/ha/d             2750
      FCHST = FTSTE*FCHN*RFSTRS

*-----fraction of protein in stem and sheaths, kg/kg                 2760
      FPST = 6.25*ANSTE/(WSTEM+REANOT(WSTEM))

*-----flow of carbohydrates available for growth of stem             2770
*     and sheaths limited by water and n stress when leaf
*     growth is dominant, kg/ha/d
      GRRSTE = FCHST*(FPST*EFCPR+(1.-FPST)*EFCCH)

*-----nitrogen demand of live stem and sheath tissue,                2780
*     kg/ha/d
      NDEMST = INSW(FRNST-OFNST,((WSTEM+GRRSTE*DELT)*OFNST
     $               -ANSTE)/TCU,0.)*ARESPF

*-----total demand for nitrogen of vegetative plant material.        2800
*     material. since nitrogen to grains is translocated only,
*     this is the total demand of the vegetation, kg/ha/d
      TNDEM = DNRT+NDEM+NDEMST

*-----rate of influx of nitrogen by rain and irrigation              2810
*     water, kg/ha/d
      NBR = RAIN*NCR+IRRID*NCIW

*-----rate of inflow of nitrogen in first soil                       2820
*     compartment, kg/ha/d
      SLTF(1) = NBR

*-----concentration of mineral nitrogen in overlying                 2830
*     soil compartment, kg/mm
      CONP(1) = (NCR*RAIN+NCIW*IRRID) /
     $          (RAIN+IRRID + REANOT(RAIN+IRRID))

*-----total available nitrogen mineral nitrogen in wet               2840
*     rooted zone, kg/ha
      TNRT = 0.

*-----total nitrogen uptake by mass flow, kg/ha/d                    2850
      TNUME = 0.

      DO 240 I = 1,ILAYER

*-----concentration of mineral nitrogen in a soil                    2860
*     compartment, kg/mm
         CONC(I) = (ASLT(I)+(RWF(I)*CONP(I)-RNAC(I))*DELT)/(WATER(I) +
     $             RWF(I)*DELT)

*-----concentration of mineral nitrogen in overlying                 2870
*     soil compartment, kg/mm
         CONP(I+1) = CONC(I)

*-----rate of inflow of nitrogen in a soil compartment,              2880
*     kg/ha/d
         SLTF(I+1) = RWF(I+1)*CONC(I)

*-----rate of nitrogen uptake by mass flow, kg/ha/d                  2890
         RNUM(I) = AMIN1(TRR(I)*CONC(I),ASLT(I)/DELT-RNAC(I))*
     $             INSW(-TNDEM,1.,0.)

*-----total available mineral nitrogen in wet rooted zone,           2900
*     kg/ha
         TNRT = TNRT+(ASLT(I)-RNAC(I)*DELT)*RTL(I)/THCKN(I)*AWATF(I)

*-----rate of addition of mineral nitrogen by fertilizer             2910
*     application in a soil compartment, kg/ha/d
         FERTAP(I) = APFERT*DISTF(I)

*-----total nitrogen uptake by mass flow, kg/ha/d                    2920
         TNUME = TNUME+RNUM(I)
240   CONTINUE

*-----total rate of nitrogen uptake by mass flow, kg/ha/d             2930
      TNUM = AMIN1(TNUME,TNDEM)

*-----rate of nitrogen uptake by mass flow in a soil                 2935
*     compartment, kg/ha/d
      DO 250 I = 1,ILAYER
         RNUM(I) = RNUM(I)*TNUM/(TNUME+REANOT(TNUME))
250   CONTINUE
```

```
*-----maximum rate of nitrogen uptake by the vegetation,          2940
*     kg/ha/d
      MXRUP = UMXR*(1.-EXP(-0.5*(WLVS+WSTEM)/CF))

*-----potential rate of uptake of nitrogen by the vege-           2950
*     tation, determined either by demand, availability
*     or maximum uptake rate, kg/ha/d
      PNUDP = AMIN1(TNDEM,TNRT/DELT,MXRUP*TCUD/DELT)

*-----rate of volatilization of ammonia from top                  2960
*     compartment, kg/ha/d
      RVOLA = ASLT(1)*(1.-RANO3)/TCV *
     $        LIMIT(0.,1.,ASLT(1)*(1.-RANO3)/LNH4)*NH4FP

*-----required contribution from diffusion to nitrogen            2980
*     supply of the canopy, kg/ha/d
      RNUDF = INSW(DVS-0.6,AMAX1(0.,(PNUDP-TNUM)*DELT/TCUD),0.)

*-----total rate of nitrogen uptake by diffusion, kg/ha/d          2990
      TNUDF = 0.

      DO 260 I = 1,ILAYER

*-----potential rate of nitrogen uptake by diffusion,              3000
*     kg/ha/d
         RNUD(I) = RNUDF*(ASLT(I)-RNAC(I)*DELT)/(TNRT+REANOT(TNRT))*
     $             RTL(I)/THCKN(I)

*-----rate of nitrogen uptake by diffusion, kg/ha/d                3010
         RNUDB(I) = AMIN1(RNUD(I),ASLT(I)/DELT-RNUM(I)-RNAC(I))*AWATF(I)

*-----total rate of nitrogen uptake by diffusion, kg/ha/d          3020
         TNUDF = TNUDF+RNUDB(I)

*-----rate of ammonia volatilization in a soil compartment,        3030
*     only effective in top soil compartment, kg/ha/d
         IF (I.EQ.1) THEN
            RVOL(1) = RVOLA
         ELSE
            RVOL(I) = 0.
         END IF

*-----rate of change in weight of mineral n per soil               3040
*     compartment, kg/ha/d
         DASLT(I) = SLTF(I)-SLTF(I+1)-RNUM(I)-RNUDB(I)-RNAC(I)-RVOL(I)
     $              +RHMIN(I)+(1.-FNIMH)*(RNRLB(I)+RNRL(I))+FERTAP(I)
260   CONTINUE

*-----rate of nitrogen uptake by the vegetation, kg/ha/d           3050
      RNU = TNUM+TNUDF

*-----rate of nitrogen uptake by the roots, kg/ha/d                3060
      RNURT = RNU*DNRT/(TNDEM+REANOT(TNDEM))

*-----rate of nitrogen uptake by above ground vegetative           3070
*     plant material, kg/ha/d
      RNUVP = RNU-RNURT

*-----rate of uptake of nitrogen by leaf blade tissue,             3080
*     kg/ha/d
      RNUV = RNUVP*NDEM/(NDEM+NDEMST+REANOT(NDEM+NDEMST))

*-----rate of nitrogen uptake by stem and sheaths,                 3090
*     kg/ha/d
      RNUST = RNUVP*NDEMST/(NDEM+NDEMST+REANOT(NDEM+NDEMST))

*-----nitrogen in live stem and sheath tissue

*-----potential rate of nitrogen accumulation in the stem          3100
*     from translocation, kg/ha/d
      PTNLST = AMAX1(0.,(WSTEM*OFNST-ANSTE-RNUST*DELT)/TCU)

*-----rate of leaching of nitrogen beyond potential                3120
*     rooting zone, kg/ha/d
      SLTFD = SLTF(K)

*-----canopy temperature ten time intervals ago, gr.c              3130
      TMPCX(1) = TMPC
      DTMPX = DELAY(11,11,TMPCX)
      DTMPA = DTMPX+INSW(TIME-10.,0.1*TSI,0.)

*-----factor to account for presence of above ground               3140
*     material, 1 if dry matter is present above ground
*     otherwise 0, -
      ABGDMF = INSW(-(WLVS+WSTEM+WGR),1.,0.)

*-----factor accounting for end of pre-anthesis phase, 1 is        3150
*     beyond anthesis, 0 is not, -
      DVSVF = INSW(DVSV-1.,0.,1.)

*-----effect of n deficiency on development up to end of           3160
*     flower initiation, -
      NFD = INSW(DVSV-DVSFE, NFDEV+(1.-NFDEV)*RFNS, 1.)

*-----rate of development in the pre-anthesis phase, d**-1         3170
      DVRV = CULTP*AMAX1(0.,0.00094*TMPC-0.00046)
     $       *(1.-DVSVF)*ABGDMF*NFD*INSW(PUSHDI-0.5,0.,1.)

*-----development stage in pre-anthesis phase, one time            3180
*     interval ago, -
      DVSVP = DVSVX
      DVSVX = DVSV

*-----development stage one time interval ago, -                   3190
      DVSP = DVSX
      DVSX = DVS

*-----factor accounting for end of tillering and ear               3200
*     formation 1 is beyond end of tiller formation,
*     0 is not, -
      DVSPRF = INSW(DVSV-DVSPRE,0.,1.)

*-----rate of development in the post-anthesis phase,              3210
*     d**-1
      DVRR = CULTM*INSW(PUSHDI-0.5,0.,1.)*(0.000913*TMPC+0.003572)
     $       *DVSVF*ABGDMF

*-----factor accounting for end of grain fill, 1 is                3220
*     beyond end of grain fill, 0 is not, -
      EGFF = INSW(DVSR-DVSEGF,0.,1.)

*-----dehydration of plant tissue, difference between              3240
*     cuticular water loss and water uptake, kg/ha/d
      DEHYD = AMAX1(0.,1.E4*(PCTRAN-TRAN)*DELT)

*-----fraction dry matter in vegetative tissue, function           3250
*     of development stage, kg/kg
      FDM = LINT(FDMT,IFDMT,DVS)

*-----potential death rate of leaf blades due to                   3260
*     dehydration, kg/ha/d
      DRLVWS = AMIN1(WLVS,DEHYD*FDM/(1.-FDM-FWDB)*WLVS/(WLVS+
     $         0.5*WSTEM+REANOT(WLVS)))/TCDDH

*-----relative death rate of leaf blades due to water              3270
*     shortage, d**-1
      RDRW = INSW(DVS-0.5,AMIN1(0.005,DRLVWS/(WLVS+REANOT(WLVS))),
     $       DRLVWS/(WLVS+REANOT(WLVS)))

*-----relative death rate of leaf blades due to nitrogen           3280
*     shortage, d**-1
      RDRN = 0.2*LINT(RDRNT,IRDRNT,FNC)

*-----relative death rate of leaf blades due to shading,           3290
*     d**-1
      RDRL = LIMIT(0.,MRDRSH,(LAI-LAILM)*MRDRSH/LAILM)

*-----relative death rate of leaves due to carbohydrate            3300
*     exhaustion, d**-1
      RDRCS = -MXRDR *
     $   LIMIT(-1.,0.,FCHNX/((DGAS-FCHNX)+REANOT(DGAS-FCHNX)))

*-----overall relative death rate of leaf blades, d**-1            3310
      RDR = AMAX1(RDRW,RDRN,RDRL,RDRCS)

*-----specific leaf area, m**2/kg                                  3320
      FLFAR = AMAX1(FLFARM,FLFARI-(FLFARI-FLFARM)*DVSV/DVSST)

*-----death rate of non-reproductive tillers, no/ha/d**-1          3330
      DRNT = RDRT*DTLN*(1.-DTLN/(TNNR+REANOT(TNNR)))

*-----rate of loss of weight of leaf blades through dying          3340
*     of leaf blades of non-reproductive tillers, kg/ha/d
      RWLLDT = AMIN1(WLVS/4.,DRNT*LWDTL)

*-----potential death rate of leaf blades due to water,            3350
*     nitrogen or carbohydrate shortage or shading, kg/ha/d
      DRQ = RDR*WLVS+RWLLDT

*-----potential death rate of leaf blades due to                   3360
*     senescence, kg/ha/d
      PDRLVS = 0.

*-----rate of decline in leaf area due to senescence,              3370
*     m**2/ha/d
      RDLFA = 0.

*-----intermediate variable in calculation of potential            3380
*     death rate of leaf blades due to senescence, kg/ha/d
      DRQR = 0.

      DO 270 I = 1,110

*-----rate of change of leaf weight in class i, kg/ha/d            3390
         DLEAF(I) = 0.

*-----rate of change in cumulative temperature sum per             3400
*     leaf class, gr.c/d
         DALFT(I) = 0.

*-----specific leaf area of leaves in class i, m**2/kg             3410
         DLA(I) = 0.
270   CONTINUE

      IF (GRLVS.GT.0..OR.NINT(PUSHD).EQ.1) THEN
         LG = LG + 1

         IF (LG.GT.110) THEN
            LG = 110
         END IF

         DLEAF(LG) = GRLVS
         DLA(LG) = FLFAR*INSW(LEAFA(LG)-0.5,1.,0.)

*-----initialization of first leaf class at emergence
         IF (NINT(PUSHD).EQ.1) THEN
            DLEAF(1) = WLVSI/DELT
            DLA(1) = FLFARI/DELT
         END IF
      END IF

      IF (LG.NE.0) THEN

         DO 290 I = 1,LG
            DALFT(I) = TMPC

            IF (ALFT(I).GE.(15.0*AVLTLF)) THEN
               DLEAF(I) = -LEAFW(I)/DELT
               PDRLVS = PDRLVS + LEAFW(I)/DELT
               RDLFA = RDLFA+LEAFW(I)*LEAFA(I)/DELT
            END IF

290      CONTINUE

         IF (PDRLVS.LT.DRQ) THEN
            RDLFA = 0.
            DRQX = DRQ

            DO 310 I = 1,LG
               LFD = LEAFW(I)/DELT

               IF (I.EQ.LG) THEN
                  LFD = LFD + GRLVS
               END IF

               DLEAF(I) = AMIN1(LFD,DRQX)
               RDLFA = RDLFA+DLEAF(I)*LEAFA(I)
               DLEAF(I) = -DLEAF(I)
               DRQX = DRQX - LFD
```

```
            IF (I.EQ.LG.AND.DRQX.GT.0.) THEN
               DRQR = DRQX
            END IF

            IF (DRQX.LE.0.) THEN
               GO TO 325
            END IF

310         CONTINUE
         END IF
      END IF

325   CONTINUE

*-----death rate of leaf blades due to all causes, kg/ha/d           3420
      DRLVS = AMAX1(0.,DRQ-DRQR,PDRLVS)

*-----relative death rate of leaves due to senescence,                3430
*     d**-1
      RDRD = PDRLVS/(WLVS+REANOT(WLVS))

*-----potential death rate of stem and sheaths due to                 3432
*     water shortage, kg/ha/d
      DRSWS = AMIN1(WSTEM/DELT,DEHYD/(1.-FDM-FWDB)*0.5*WSTEM/
     $        (WLVS+0.5*WSTEM+REANOT(0.5*WSTEM+WLVS))/TCDDH)

*-----relative death rate of stem and sheaths due to                  3434
*     water shortage, d**-1
      RDRWS = DRSWS/(WSTEM+REANOT(WSTEM))

*-----relative death rate of stem and sheaths due to                  3436
*     water shortage or senescence, d**-1
      RDRWDS = AMAX1(RFST*RDRD,RDRWS)

*-----relative death rate of stem and sheaths due to                  3440
*     water shortage, d**-1
      RDRS = INSW(-DRLVS,AMAX1(RFST*RDR,RSLDS*RDRD),RDRCS)

*-----death rate of the roots, kg/ha/d                                3450
      DRRT = WRT*RDRS*RFRT

*-----rate of nitrogen loss from dying roots, kg/ha/d                 3460
      RNLDR = DRRT*INSW(RDRWDS-RFST*RDRN,RN,FRNR)

*-----nitrogen mineralization from fresh and stable                   3470
*     organic material, kg/ha/d
      DO 330 I = 1,ILAYER

*-----rate of change in weight of nitrogen in fresh                   3480
*     organic material, kg/ha/d
         DFON(I)  = -RNRL(I)+DFFOM(I)*RNLDR

*-----rate of change in weight of stable organic                      3490
*     material in a soil compartment, kg/ha/d
         DHUM(I)  = (FNIMH*(RNRL(I)+RNRLB(I))-RHMIN(I))/NCH*10.

*-----rate of change in weight of nitrogen in stable                  3500
*     organic material, kg/ha/d
         DNHUM(I) = FNIMH*(RNRL(I)+RNRLB(I))-RHMIN(I)

*-----rate of change in weight of fresh organic material,             3510
*     kg/ha/d
         DFOM(I)  = -DECR(I)+DFFOM(I)*DRRT
330   CONTINUE

*-----rate of change in weight of the leaf blades,                    3520
*     not corrected, kg/ha/d
      CRCWLV = GRLVS-DRLVS+PUSHD*WLVSI/DELT

*-----rate of loss of nitrogen by dying leaf blades,                  3530
*     kg/ha/d
      RNL = DRLVS*(FRNV-(FRNV-BN)*(1.-RFNS))

*-----rate of nitrogen loss from live leaf blades,                    3540
*     kg/ha/d
      TRNLL = AMIN1(RNL+RNTLS,ANLV/DELT)

*-----rate of change in weight of nitrogen in leaf                    3550
*     blades, kg/ha/d
      CRCANL = RNUV-TRNLL+PUSHD*ANLVI/DELT

*-----rate of change in weight of nitrogen in leaf                    3560
*     blades, limited to amount present, kg/ha/d
      RCANLV = INSW(CRCANL,AMAX1(-ANLV/DELT,CRCANL),CRCANL)

*-----rate of translocation of nitrogen from dying                    3570
*     leaf blades to stem and sheaths, kg/ha/d
      TNLST = AMIN1(PTNLST,RNL-DRLVS*BN)

*-----rate of change in weight of live leaf blades,                   3580
*     limited to amount present, kg/ha/d
      RCWLVS = INSW(CRCWLV,AMAX1(-WLVS/DELT,CRCWLV),CRCWLV)

*-----death rate of stem tissue, due to either water                  3590
*     shortage, nitrogen deficiency, senescence or
*     carbohydrate exhaustion, kg/ha/d
      DRSTE = RDRS*WSTEM

*-----rate of loss of weight of stem and sheaths through              3600
*     dying of stem and sheaths of non-reproductive
*     tillers, kg/ha/d
      RWLSDT = AMIN1(DRNT*SWDTL,WSTEM/DELT-DRSTE)

*-----rate of change in weight of stem and sheaths, kg/ha             3610
      CRCWST = GRRSTE-DRSTE-RWLSDT

*-----rate of nitrogen loss from dying stem and sheaths,              3650
*     kg/ha/d
      RNLDST = DRSTE*INSW(WLVS-1.,FRNST,
     $         INSW(RDRWDS-RFST*RDRN,LN,FRNST))

*-----rate of weight increase of stem and sheaths, kg/ha/d            3660
      RCWST = INSW(CRCWST,AMAX1(-WSTEM/DELT,CRCWST),CRCWST)

*-----auxilary variable in calculation of rate of change              3670
*     in weight of roots, kg/ha/d
      CRCWR1 = GRRT-DRRT+PUSHD*WRTI/DELT

*-----rate of change in weight of the roots, not corrected,           3680
*     kg/ha/d
      CRCWRT =INSW(PUSHD-0.5,INSW(-(WSTEM/DELT+RCWST+WLVS/DELT+RCWLVS)
     $               ,CRCWR1,-WRT/DELT),CRCWR1)

*-----rate of change of weight of the root system, limited            3690
*     to amount present, kg/ha
      RCWRT = INSW(CRCWRT,AMAX1(-WRT/DELT,CRCWRT),CRCWRT)

*-----rate of increase in rooting depth, mm/d                         3700
      RGRRL = SWPBT*DGRRT*LINT(REDTTB,IRDTTB,TS)
     $    * INSW(RTD-MXRTD,1.,0.)
     $    * (RTF+1.-RTF*RFSTRS)*INSW(WLVS-WLVSI,0.,1.)
     $    * INSW(-FDSR,1.,0.)

*-----fraction of assimilate supply allocated to the                  3710
*     reserve pool, function of development stage, kg/kg
      FTRL = LINT(FTRLT,IFTRLT,DVS)

*-----rate of increase in weight of reserves, kg/ha/d                 3720
      FCHTRS = FTRL*FCHN+SCHFLV*(1.-FSCHG)

*-----fraction of proteins in the grain (dry weight),                 3730
*     value one time interval ago, kg/kg
      FPGC = LIMIT(0.,1.,5.7*ANGR/(WGR+REANOT(WGR)))

*-----potential rate of dry matter accumulation of an                 3740
*     individual grain, function of canopy temperature,
*     kg/grain/d
      PGRIG = LINT(PGRIGT,IPGIGT,TMPC)*SGFF

*-----potential rate of dry matter accumulation in the                3750
*     grains, kg/ha/d
      PGRRG = GRN*PGRIG*(1.-EGFF)*INSW(PUSHN-0.5,
     $        (1.-LIMIT(0.,1.,(FRNGL-FRNG)/(FRNGL-FRNGL1))),1.)

*-----efficiency of conversion of primary assimilation                3760
*     products in grain dry matter, kg/ha
      CEGR = FPGC*EFCPRG+(1.-FPGC)*EFCCH

*-----weight of non-remobilizable reserve carbohydrates               3770
*     in the canopy, kg/ha
      RESRES = RESLR*(WSTEM+WLVS+WRT)

*-----actual rate of translocation of reserve carbohydrates           3780
*     from vegetative tissue to the grain, kg/ha/d
      ARTTG = AMIN1(PGRRG,AMAX1(0.,((ARESP-RESRES)/TCTR)*CEGR))

*-----rate of weight loss of the grain due to maintenance             3790
*     respiration before ripening of the grain when other
*     carbohydrate sources are exhausted, kg/ha/d
      LMR = (ARESPF-1.)*EGFF*
     $      LIMIT(0.,RMNG,RMNG*(1.-(DVSR-0.4)/(0.7-0.4)))

*-----rate of increase in dry weight of the grain, kg/ha/d            3800
      GRGR = ARTTG - LMR

*-----rate of translocation of nitrogen from root to                  3810
*     the grain, kg/ha/d
      RNTRS = RNTG*AVNRT/(AVN+REANOT(AVN))

*-----rate of nitrogen uptake by the roots, not                       3820
*     corrected, kg/ha/d
      CRCANR = RNURT-RNLDR-RNTRS+PUSHD*ANRTI/DELT

*-----rate of change in weight of nitrogen in the roots,              3830
*     limited to amount present, kg/ha/d
      RCANRT = INSW(CRCANR,AMAX1(-ANRT/DELT,CRCANR),CRCANR)

*-----rate of consumption of reserve carbohydrates for                3840
*     maintenance respiration, kg/ha/d
      CRMR = INSW(FCHNX,AMIN1(ARESP/DELT-(ARTTG/(CEGR+REANOT(CEGR))
     $       +TRFRS),-FCHNX),0.)

*-----rate of increase in weight of reserve carbohydrates,            3850
*     kg/ha/d
      RCRES = FCHTRS-ARTTG/(CEGR+REANOT(CEGR))-TRFRS-CRMR

*-----individual grain weight, kg*1.e-6/grain                         3860
      OTGW = WGR/(GRN+REANOT(GRN))*1.E6

*-----tiller formation

*-----time constant for tiller formation, function of                 3870
*     canopy temperature, d
      TCTF = LINT(TCTFT,ITCTFT,TMPC)*LIMIT(0.3,1.,0.3+LAI*0.7)

*-----carbohydrate requirement for tiller formation,                  3890
*     kg/tiller/d
      CFTUDM = CHFTB*DVRV/(DVSPRE-DVSTS)

*-----maximum number of tillers that can be maintained                3900
*     by assimilate supply, no/ha
      TLNM = (FCHTLV+FCHST+FCHTRS)/(CFTUDM+REANOT(CFTUDM))

*-----growth rate of number of tillers, no/ha/d                       3910
      GRNT = (1.-DVSPRF)*AMAX1(0.,(TLNM-TLN)/TCTF*FNC)

*-----auxiliary variable used to avoid zero division,                 3920
*     no/ha
      TLNIXX = TLNIX+REANOT(TLNIX)

*-----rate of nitrogen loss from stem and sheaths of dying            3930
*     tillers, kg/ha/d
      RNLSDT = RWLSDT*LN

*-----rate of translocation of nitrogen from stem and                 3940
*     sheaths to the grain, kg/ha/d
      RNTSS = RNTG*AVNSTE/(AVN+REANOT(AVN))

*-----total rate of nitrogen loss from stem and sheaths,              3950
*     kg/ha/d
      TRNLS = AMIN1(ANSTE/DELT,RNLDST+RNLSDT+RNTSS)

*-----initial weight of nitrogen in the grain on the first            3960
*     first day of grain growth, kg/ha
      ANGRI = PUSHN*GRGR*0.035

*-----rate of change in weight of nitrogen in live stem               3970
*     and leaf sheaths, kg/ha/d
      CRCANS = RNUST-TRNLS+TNLST-ANGRI
```

```
*-----rate of change in weight of nitrogen in stem and          3980
*     sheaths, limited to amount present, kg/ha/d
      RCANST = INSW(CRCANS,AMAX1(-ANSTE/DELT,CRCANS),CRCANS)

*-----ear formation

*-----carbohydrate requirement for ear formation per            3990
*     unit development, kg/ear/d
      CFEUDM = CHFEB*DVRV/(DVSPRE-DVSSE)

*-----maximum number of ears that can be maintained by          4000
*     current carbohydrate supply, no/ha
      ALTN = (FCHTLV+FCHST+FCHTRS)/(CFEUDM+REANOT(CFEUDM))

*-----maximum number of ears that can be formed, no/ha          4010
      MXNE = LIMIT(0.,TLN,ALTN)

*-----time constant for ear formation, d                        4020
      TCEF = EB+AMAX1(0.,(TLN/TLNIXX-1.)*(STCEF-EB))

*-----switch variable 1 on the day of start of floral           4030
*     initiation, otherwise 0
      PUSHE = REAAND(DVSSE-DVSVP,DVSV-DVSSE)

*-----potential rate of increase in ear number as               4040
*     determined by assimilate supply, no/ha/d
      REARF1 = AMAX1(0.,(MXNE-EARN)/TCEF)*(1.-PUSHE)

*-----rate of increase in ear number, limited to                4050
*     appropriate phenological phase, no/ha/d
      REARF = INSW(DVSV-DVSSE,0.,(1.-DVSPRF)*REARF1)

*-----spikelet formation

*-----carbohydrate requirement for spikelet formation           4060
*     per unit development, kg/spikelet/d
      CFSUDM = CHFSB*DVRV/(DVSSPE-DVSSPS)

*-----maximum number of spikelets, that can be maintained       4070
*     by current assimilate availability, no/ha
      ALSN = (FCHTLV+FCHST+FCHTRS)/(CFSUDM+REANOT(CFSUDM))

*-----maximum number of spikelets that can be formed, no/ha     4080
      MXNSP = LIMIT(0.,EARN*25.,ALSN)

*-----time constant for spikelet formation, d                   4090
      TCSF = SB+AMAX1(0.,(EARN/TLNIXX-1.)*(STCSF-SB))

*-----rate of spikelet formation as determined by               4100
*     assimilate availability, no/ha/d
      RSPLF1 = AMAX1(0.,(MXNSP-NSPS)/TCSF)

*-----rate of spikelet formation, limited to appropriate        4110
*     phenological phase, no/ha/d
      RSPLF = INSW(DVSV-DVSSPS,0.,INSW(DVSSPE-DVSV,0.,RSPLF1))

*-----fertile floret formation

*-----carbohydrate requirement for floret formation per         4120
*     unit development, kg/floret/d
      CFFUDM = CHFFB*DVRV/(DVSFE-DVSFS)

*-----maximum number of fertile florets, limited by             4130
*     assimilate availability, no/ha
      ALFN = (FCHTLV+FCHST+FCHTRS)/(CFFUDM+REANOT(CFFUDM))

*-----maximum number of fertile florets that can be formed,     4140
*     no/ha
      MXNFFL = LIMIT(0.,NSPS*4,ALFN)

*-----time constant for fertile floret formation, d             4150
      TCFF = FB+AMAX1(0.,(NSPS/TLNIXX-5.)*(STCFF-FB)/20.)

*-----rate of formation of fertile florets, limited by          4160
*     assimilate availability, no/ha/d
      RFFF1 = AMAX1(0.,(MXNFFL-FFNR)/TCFF)

*-----rate of formation of fertile florets, limited to          4170
*     appropriate phenological phase, no/ha/d
      RFFF = INSW(DVSV-DVSFS,0.,INSW(DVSFE-DVSV,0.,RFFF1))

*-----grain formation

*-----effect of minimum temperature on grain set, -             4180
      RFT = LINT(RFTT,IRFTT,TMIN)

*-----time constant for grain formation, d                      4190
      TCGF = GB+AMAX1(0.,(FFNR/TLNIXX-10.)*(STCGF-GB)/90.)

*-----rate of increase grain density, no/ha/d                   4200
      RGRN = INSW(DVSR-DVSGS,0.,(1.-SGFF)*(FFNR-GRN)/TCGF*RFT)

*-----switch variable, used to define size of non-              4210
*     reproductive tillers, 1 at end of ear forming
*     phase, otherwise 0, -
      PUSHT = REAAND(DVSV-DVSPRE,DVSPRE-DVSVP)

*-----rate of change in leaf area, not corrected,               4220
*     m**2/ha/d
      CRCLFA = GRLVS*FLFAR-RDLFA+PUSHD*ARLFI/DELT

*-----rate of change in green leaf area, limited to             4230
*     amount present, m**2/ha/d
      RCLFA = INSW(CRCLFA,AMAX1(-ARLF/DELT,CRCLFA),CRCLFA)

*-----specific leaf weight, kg/m**2                             4240
      SLWA = (WLVS+ARESP*WLVS/(TVEGM+REANOT(TVEGM)))
     $       / (ARLF+REANOT(ARLF))

*-----rate of increase of grain filling period, d               4250
      R1 = DATE*REAAND(DVSR-DVSEGF,DVSEGF-DVSRP)

*-----rate of increase of tiller number, no/ha                  4260
      R2 = GRNT-DRNT+PUSHD*TLNI/DELT

*-----rate of increase of dead tillers, no/ha                   4270
      R3 = PUSHT*(TLN-EARN)*0.01+DRNT

*-----rate of increase of non-earbearing tillers at             4280
*     dvspre, no/ha
      R4 = PUSHT*(TLN-EARN)

*-----rate of increase of leaf weight of average non-           4290
*     reproductive tiller, kg/d
      R5 = PUSHT*WLVS/(TLN+REANOT(TLN))

*-----rate of increase of stem weight of average non-           4300
*     reproductive tiller, kg/d
      R6 = PUSHT*WSTEM/(TLN+REANOT(TLN))

*-----rate of increase of total weight, kg/ha/d                 4310
      GTW = RCWLVS+RCWST+RCWRT+GRGR

      IF (OUTPUT .OR. TERMNL) THEN
         CALL OUTDAT (2, 0, 'DVS',    DVS)
         CALL OUTDAT (2, 0, 'PUSHD',  PUSHD)
         CALL OUTDAT (2, 0, 'TADRW',  TADRW)
         CALL OUTDAT (2, 0, 'WLVS',   WLVS)
         CALL OUTDAT (2, 0, 'ARLF',   ARLF)
         CALL OUTDAT (2, 0, 'SLWA',   SLWA)
         CALL OUTDAT (2, 0, 'GRAI',   GRAI)
         CALL OUTDAT (2, 0, 'WRT',    WRT)
         CALL OUTDAT (2, 0, 'RTD',    RTD)
         CALL OUTDAT (2, 0, 'WSTEM',  WSTEM)
         CALL OUTDAT (2, 0, 'WGR',    WGR)
         CALL OUTDAT (2, 0, 'GRN',    GRN)
         CALL OUTDAT (2, 0, 'OTGW',   OTGW)
         CALL OUTDAT (2, 0, 'FFNR',   FFNR)
         CALL OUTDAT (2, 0, 'NSPS',   NSPS)
         CALL OUTDAT (2, 0, 'EARN',   EARN)
         CALL OUTDAT (2, 0, 'TLN',    TLN)
         CALL OUTDAT (2, 0, 'ARESP',  ARESP)
         CALL OUTDAT (2, 0, 'RESL',   RESL)
         CALL OUTDAT (2, 0, 'TRFRS',  TRFRS)
         CALL OUTDAT (2, 0, 'ARTTG',  ARTTG)
         CALL OUTDAT (2, 0, 'FCHN',   FCHN)
         CALL OUTDAT (2, 0, 'AMAXN',  AMAXN)
         CALL OUTDAT (2, 0, 'DTGA',   DTGA)
         CALL OUTDAT (2, 0, 'DGAS',   DGAS)
         CALL OUTDAT (2, 0, 'APTRAN', APTRAN)
         CALL OUTDAT (2, 0, 'TRAN',   TRAN)
         CALL OUTDAT (2, 0, 'TOTRAN', TOTRAN)
         CALL OUTDAT (2, 0, 'EVTOT',  EVTOT)
         CALL OUTDAT (2, 0, 'TEVAP',  TEVAP)
         CALL OUTDAT (2, 0, 'WTOT',   WTOT)
         CALL OUTDAT (2, 0, 'TRAIN',  TRAIN)
         CALL OUTDAT (2, 0, 'INFR',   INFR)
         CALL OUTDAT (2, 0, 'PEVAP',  PEVAP)
         CALL OUTDAT (2, 0, 'PEVAP1', PEVAP1)
         CALL OUTDAT (2, 0, 'EVAP',   EVAP)
         CALL OUTDAT (2, 0, 'TDRAIN', TDRAIN)
         CALL OUTDAT (2, 0, 'NTOT',   NTOT)
         CALL OUTDAT (2, 0, 'BIOMNT', BIOMNT)
         CALL OUTDAT (2, 0, 'TNDEM',  TNDEM)
         CALL OUTDAT (2, 0, 'RNUV',   RNUV)
         CALL OUTDAT (2, 0, 'TNABM',  TNABM)
         CALL OUTDAT (2, 0, 'ANLV',   ANLV)
         CALL OUTDAT (2, 0, 'ANGR',   ANGR)
      END IF

*-----end of rate calculation

      ELSE IF (ITASK.EQ.3) THEN

*-----integration section

*-----variable to monitor end of germination, -                 4330
      PUSHDI = INTGRL(PUSHDI,PUSHD,DELT)

*-----total seasonal rainfall, mm                               4340
      TRAIN = INTGRL(TRAIN,RAIN,DELT)

*-----total seasonal soil surface evaporation, mm               4350
      TEVAP = INTGRL(TEVAP,EVTOT,DELT)

*-----total seasonal infiltration, mm                           4360
      TOTINF = INTGRL(TOTINF,INFR,DELT)

*-----total amount of water loss by drainage beyond             4370
*     potential rooting zone, mm
      TDRAIN = INTGRL(TDRAIN,RDRAIN,DELT)

*-----total seasonal nitrogen fertilizer application,           4380
*     kg n/ha
      TNTRMT = INTGRL(TNTRMT,APFERT,DELT)

*-----relative weight of nitrate in total inorganic             4390
*     nitrogen store in top soil compartment, -
      RANO3 = INTGRL(RANO3,(1.-RANO3)/TCN,DELT)

*-----state variables and totals for the whole soil             4400
*     profile
      DO 340 I = 1,ILAYER

*-----amount of soil moisture in a soil compartment, mm         4410
      WATER(I) = INTGRL(WATER(I),DWAT(I),DELT)

*-----weight of mineral nitrogen in a soil compartment,         4420
*     kg/ha
      ASLT(I) = INTGRL(ASLT(I),DASLT(I),DELT)

*-----weight of fresh organic matter in a soil compartment,     4430
*     kg/ha
      FOM(I) = INTGRL(FOM(I),DFOM(I),DELT)

*-----weight of organic nitrogen in fresh organic matter,       4440
*     in a soil compartment, kg/ha
      FON(I) = INTGRL(FON(I),DFON(I),DELT)

*-----weight of stable organic matter in a soil                 4450
*     compartment, kg/ha
      HUM(I) = INTGRL(HUM(I),DHUM(I),DELT)

*-----weight of organic nitrogen in stable organic              4460
*     matter, in a soil compartment, kg/ha
      NHUM(I) = INTGRL(NHUM(I),DNHUM(I),DELT)

*-----weight of nitrogen in microbial biomass in a              4470
*     soil compartment, kg/ha
```

```
      BIOMN(I) = INTGRL(BIOMN(I),DBIOMN(I),DELT)
340   CONTINUE

*-----procedure wtot,ntot,tfon,nhumt,tfom,humt,biomnt =              4480
*     sumpro(water,aslt,fon,nhum,fom)

      WTOT = 0.
      NTOT = 0.
      TFON = 0.
      TFOM = 0.
      NHUMT = 0.
      HUMT = 0.
      BIOMNT = 0.

      DO 350 I = 1,ILAYER
         IF (TDT(I)+0.5.GT.MXRTD) GO TO 350

*-----total amount of water in the profile, mm                        4490
         WTOT = WTOT+WATER(I)

*-----total weight of inorganic nitrogen in the profile,              4500
*     kg/ha
         NTOT = NTOT+ASLT(I)

*-----total nitrogen in fresh organic material in profile,            4510
*     kg/ha
         TFON = TFON+FON(I)

*-----total nitrogen in humus in profile, kg/ha                       4520
         NHUMT = NHUMT+NHUM(I)

*-----total fresh organic material in profile, kg/ha                  4530
         TFOM = TFOM+FOM(I)

*-----total humus in profile, kg/ha                                   4540
         HUMT = HUMT+HUM(I)

*-----total nitrogen in microbial biomass, kg/ha                       4550
         BIOMNT = BIOMNT+BIOMN(I)
350   CONTINUE

*-----soil temperature, running ten-day average of canopy             4560
*     temperature, gr.c
      TS = INTGRL(TS,0.1*(TMPC-DTMPA)/DELT,DELT)

*-----total seasonal transpiration of the vegetation, mm              4570
      TOTRAN = INTGRL(TOTRAN,TRAN,DELT)

*-----anthesis day, julian day number                                 4580
      ANTHES = INTGRL(ANTHES,DATE*REAAND(DVS-0.5,0.5-DVSP)/DELT,
     $  DELT)

*-----development stage in the pre-anthesis phase, 0 at               4590
*     emergence, 1 at anthesis, -
      DVSV = INTGRL(DVSV,DVRV+PUSHD*DVSI/DELT,DELT)

*-----development stage in the post-anthesis phase, 0                 4600
*     at anthesis, 1 at dead ripeness, -
      DVSR = INTGRL(DVSR,DVRR,DELT)

*-----development stage as fraction of total growth cycle,             4610
*     used in determination of assimilate partitioning, -
      DVS = 0.5*(DVSV+DVSR)

      DO 360 I = 1,110

*-----leaf weight per leaf class, kg/ha                                4620
         LEAFW(I) = INTGRL(LEAFW(I),DLEAF(I),DELT)

*-----accumulated temperature sum per leaf class, gr./d                4630
         ALFT(I)  = INTGRL(ALFT(I),DALFT(I),DELT)

*-----leaf area per leaf class, m**2/ha                                4640
         LEAFA(I) = INTGRL(LEAFA(I),DLA(I),DELT)

360   CONTINUE

*-----weight of live stem and sheaths, kg/ha                           4650
      WSTEM = INTGRL(WSTEM,RCWST,DELT)

*-----weight of live roots, kg/ha                                      4660
      WRT = INTGRL(WRT,RCWRT,DELT)

*-----rooting depth, mm                                                4670
      RTD = INTGRL(RTD,RGRRL+PUSHD*IRTD/DELT,DELT)

*-----weight of reserve carbohydrates in the vegetation,               4680
*     kg/ha
      ARESP = INTGRL(ARESP,RCRES+PUSHD*ARESPI/DELT,DELT)

*-----dry weight of the grain, kg/ha                                   4690
      WGR = INTGRL(WGR,GRGR,DELT)

*-----end of grain filling period, julian day number                   4700
      EGFDAY = INTGRL(EGFDAY,R1,DELT)

*-----dry weight of live leaf blades, kg/ha                            4710
      WLVS = INTGRL(WLVS,RCWLVS,DELT)

*-----tiller density, no/ha                                            4720
      TLN = INTGRL(TLN,R2,DELT)

*-----total number of dead tillers, no/ha                              4730
      DTLN = INTGRL(DTLN,R3,DELT)

*-----average weight per tiller, kg/tiller                             4740
      AWTL = (WLVS+WSTEM)/(TLN+REANOT(TLN))

*-----total number of non-reproductive tillers at end                  4750
*     of tillering, no/ha
      TNNR = INTGRL(TNNR,R4,DELT)

*-----average weight of leaves of non-reproductive                     4760
*     tillers at end of ear formation, kg/tiller
      LWDTL = INTGRL(LWDTL,R5,DELT)

*-----final weight of stem and sheaths of non-                         4770
*     reproductive tillers at end of ear formation,
*     kg/tiller
      SWDTL = INTGRL(SWDTL,R6,DELT)

*-----total number of ears, no/ha                                      4780
      EARN = INTGRL(EARN,REARF+PUSHE*TLNI,DELT)

*-----total number of spikelets, no/ha                                 4790
      NSPS = INTGRL(NSPS,RSPLF,DELT)

*-----total number of fertile florets, no/ha                           4800
      FFNR = INTGRL(FFNR,RFFF,DELT)

*-----grain density, no/ha                                             4810
      GRN = INTGRL(GRN,RGRN,DELT)

*-----total area of green and senesced leaf blades,                    4820
*     m**2/ha
      ARLFE = INTGRL(ARLFE,GRLVS*FLFAR+PUSHD*ARLFI/DELT,DELT)

*-----total green area of the leaf blades, m**2/ha                     4830
      ARLF = INTGRL(ARLF,RCLFA,DELT)

*-----leaf area index, m**2/m**2                                       4840
      LAI = 1.E-4*ARLF

*-----weight of nitrogen in live tissue of leaf blades,                4850
*     kg/ha
      ANLV = INTGRL(ANLV,RCANLV,DELT)

*-----weight of nitrogen in stem and leaf sheats, kg/ha                4860
      ANSTE = INTGRL(ANSTE,RCANST,DELT)

*-----weight of nitrogen in the roots, kg/ha                           4870
      ANRT = INTGRL(ANRT,RCANRT,DELT)

*-----weight of nitrogen in the grain, kg/ha                           4880
      ANGR = INTGRL(ANGR,RNTG+ANGRI,DELT)

*-----total weight of vegetative organs, kg/ha                         4890
      TVEGM = WLVS+WSTEM+WRT

*-----total weight of nitrogen in dead above ground                    4900
*     vegetative material, kg/ha
      NDSTR = INTGRL(NDSTR,RNL-TNLST+RNLDST+RNLSDT,DELT)

*-----total weight of dead above ground vegetative                     4910
*     material, kg/ha
      DSTR = INTGRL(DSTR,DRLVS+DRSTE+RWLSDT,DELT)

*-----shoot root ratio, kg/kg                                          4920
      SRR = (WSTEM+WLVS+WGR+DSTR)/(WRT+REANOT(WRT))

*-----total nitrogen loss by leaching, kg/ha                           4930
      TNLCH = INTGRL(TNLCH,SLTFD,DELT)

*-----total initial weight of nitrogen in the vegetation               4940
*     at emergence, kg/ha
      INIV = INTGRL(INIV,PUSHD*(ANLVI+ANRTI)/DELT,DELT)

*-----total nitrogen loss by volatilization, kg/ha                     4950
      TNVOL = INTGRL(TNVOL,RVOLA,DELT)

*-----total weight of nitrogen in above ground material,               4960
*     kg/ha
      TNABM = ANLV+ANSTE+ANGR+NDSTR

*-----total above ground dry weight, excluding root                    4970
*     reserves, kg/ha
      TADRW = WLVS+WSTEM+WGR+DSTR+ARESP*(1.-WRT/(TVEGM+REANOT(TVEGM)))

*-----carbon balance, kg/ha                                            4980
      CARBAL = FCHN-FCHTR-FCHTLV-FCHST-ARTTG /
     $         (CEGR+REANOT(CEGR))-CRMR-RCRES

*-----nitrogen balance, should be zero throughout the                  4990
*     simulation to ensure no losses, kg/ha
      NBAL = TNINT+TNTRMT+INIV+TOTINF*0.02+FRNF*FOMI+NHUMI+
     $       BIOMNI-BIOMNT-NTOT-TFON-TNVOL-NHUMT-ANLV-ANSTE-ANRT-
     $       ANGR-NDSTR-TNLCH

*-----set logical name terminal when the crop reached                  5000
*     maturity
      IF (DVS.GE.1.) TERMNL = .TRUE.

*-----end of subroutine

      END IF

      ITOLD = ITASK

      RETURN
      END
```

Program: REST.FOR

```
      REAL FUNCTION DELAY (N,TD,XIN)
      IMPLICIT REAL (A-Z)

      INTEGER I,N,TD
      DIMENSION XIN(N+1)

      DO 10 I = TD,1,-1
         XIN(I+1) = XIN(I)
10    CONTINUE

      DELAY = XIN(TD+1)
      RETURN
      END

      REAL FUNCTION REANOT (X1)

*     This function emulates the CSMP function NOT
*     REANOT is similar to logical expression
*     REANOT = 1  when  X1 <=0
*     REANOT = 0  when  X1 > 0

*     REANOT - Function result        O
*     X1     - first argument         I
*
*     No subroutines and/or functions called

*     Author: Daniel van Kraalingen
*     Date  : November 1989

*     formal parameters
      REAL X1

**    no local variables
      SAVE

      IF (X1.LE.0.) THEN
         REANOT = 1.
      ELSE
         REANOT = 0.
      END IF

      RETURN
      END

*----------------------------------------------------------------------*
* SUBROUTINE ASTRO                                                     *
* Authors: Daniel van Kraalingen                                       *
* Date   : 9-Aug-1987                                                  *
* Modified by Jan Goudriaan 4 Febr 1988                                *
* Modified by Jan Goudriaan and Kees Spitters 7 December 1989          *
* Purpose: This subroutine calculates astronomic daylength,            *
*          photoperiodic daylength, diurnal radiation characteristics *
*          such as the daily integral of sine of solar elevation and  *
*          solar constant.                                             *
*          Measured daily total of global radiation is used to find   *
*          atmospheric transmissivity and fraction diffuse radiation  *
* FORMAL PARAMETERS:  (I=input,O=output,C=control,IN=init,T=time)     *
* name   meaning                                    units  class      *
* ----   -------                                    -----  -----      *
* DAY    Day number (Jan 1st = 1)                     -      I        *
* LAT    Latitude of the site                      degrees   I        *
* SC     Solar constant                            J m-2 s-1 O        *
* DS0    Daily extraterrestrial radiation          J m-2 d-1 O        *
* SINLD  Seasonal offset of sine of solar height     -       O        *
* COSLD  Amplitude of sine of solar height           -       O        *
* DAYL   Astronomical daylength (base = 0 degrees)   h       O        *
* DSINB  Daily total of sine of solar height         s       O        *
* DSINBE Daily total of effective solar height       s       O        *
*                                                                      *
* FATAL ERROR CHECKS (execution terminated, message)                  *
* condition: LAT > 67, LAT < -67                                       *
*                                                                      *
* SUBROUTINES and FUNCTIONS called : ERROR                             *
* FILE usage : none                                                    *
*----------------------------------------------------------------------*

      SUBROUTINE ASTRO (DAY, LAT,
     &                  SC, DS0, SINLD, COSLD, DAYL, DSINB, DSINBE)
      IMPLICIT REAL (A-Z)
      SAVE

*-----PI and conversion factor from degrees to radians
      PARAMETER (PI=3.141592654, RAD=0.017453292)

*-----check on input range of parameters
      IF (LAT.GT.67.)  CALL ERROR ('ASTRO','LAT > 67')
      IF (LAT.LT.-67.) CALL ERROR ('ASTRO','LAT < -67')

*-----declination of the sun as function of daynumber (DAY)
      DEC = -ASIN (SIN (23.45*RAD)*COS (2.*PI*(DAY+10.)/365.))

*-----SINLD, COSLD and AOB are intermediate variables

      SINLD = SIN (RAD*LAT)*SIN (DEC)
      COSLD = COS (RAD*LAT)*COS (DEC)
      AOB   = SINLD/COSLD

*-----daylength (DAYL)
      DAYL   = 12.0*(1.+2.*ASIN (AOB)/PI)

      DSINB  = 3600.*(DAYL*SINLD+24.*COSLD*SQRT (1.-AOB*AOB)/PI)
      DSINBE = 3600.*(DAYL*(SINLD+0.4*(SINLD*SINLD+COSLD*COSLD*0.5))
     &         +12.0*COSLD*(2.0+3.0*0.4*SINLD)*SQRT (1.-AOB*AOB)/PI)

*-----solar constant (SC) and daily extraterrestrial (DS0)
      SC  = 1370.*(1.+0.033*COS (2.*PI*DAY/365.))
      DS0 = SC*DSINB

      RETURN
      END

*----------------------------------------------------------------------*
* SUBROUTINE TOTASS                                                    *
* Authors: Daniel van Kraalingen                                       *
* Date   : 10-Dec-1987                                                 *
* Modified by Jan Goudriaan 5-Febr-1988                                *
* Modified by Jan Goudriaan and Kees Spitters 7 December 1989          *
* Purpose: This subroutine calculates daily total gross                *
*          assimilation (DTGA) by performing a Gaussian integration   *
*          over time. At three different times of the day,            *
*          radiation is computed and used to determine assimilation   *
*          whereafter integration takes place.                         *
*                                                                      *
* FORMAL PARAMETERS:  (I=input,O=output,C=control,IN=init,T=time)     *
* name   meaning                                    units  class      *
* ----   -------                                    -----  -----      *
* SC     Solar constant                            J m-2 s-1 I        *
* DAYL   Astronomical daylength (base = 0 degrees     h      I        *
* SINLD  Seasonal offset of sine of solar height      -      I        *
* COSLD  Amplitude of sine of solar height            -      I        *
* DSINBE Daily total of effective solar height        s      I        *
* DTR    Daily total of global radiation            J/m2/d   I        *
* SCV    Scattering coefficient of leaves for visible                 *
*          radiation (PAR)                            -      I        *
* AMAX   Assimilation rate at light saturation      kg CO2/  I        *
*                                                  ha leaf/h          *
* EFF    Initial light use efficiency              kg CO2/J/ I        *
*                                                  ha/h m2 s          *
* KDIF   Extinction coefficient for diffuse light            I        *
* LAI    Leaf area index                             ha/ha   I        *
* DTGA   Daily total gross assimilation          kg CO2/ha/d O        *
*                                                                      *
*                                                                      *
* SUBROUTINES and FUNCTIONS called : ASSIM                             *
* FILE usage : none                                                    *
*----------------------------------------------------------------------*

      SUBROUTINE TOTASS (SC, DAYL, SINLD, COSLD, DSINBE,
     &                   DTR, SCV, AMAX, EFF, KDIF, LAI, DTGA)
      IMPLICIT REAL(A-Z)
      REAL XGAUSS(3), WGAUSS(3)
      INTEGER I, IGAUSS
      SAVE

      PARAMETER (PI=3.141592654)

      DATA IGAUSS /3/
      DATA XGAUSS /0.1127, 0.5000, 0.8873/
      DATA WGAUSS /0.2778, 0.4444, 0.2778/

*-----assimilation set to zero and three different times
*     of the day (HOUR)
      DTGA = 0.

      DO 10 I=1,IGAUSS

*--------at the specified HOUR, radiation is computed and
*        used to compute assimilation
         HOUR = 12.0+DAYL*0.5*XGAUSS(I)

*--------sine of solar elevation
         SINB  = MAX (0., SINLD+COSLD*COS (2.*PI*(HOUR+12.)/24.))

*--------diffuse light fraction (FRDIF) from atmospheric
*        transmission (ATMTR)
         PAR   = 0.5*DTR*SINB*(1.+0.4*SINB)/DSINBE
         ATMTR = PAR/(0.5*SC*SINB)

         IF (ATMTR.LE.0.22) THEN
            FRDIF = 1.
         ELSE IF (ATMTR.LE.0.35) THEN
            FRDIF = 1.-6.4*(ATMTR-0.22)**2
         ELSE
            FRDIF = 1.47-1.66*ATMTR
         END IF
         FRDIF = MAX (FRDIF, 0.15+0.85*(1.-EXP (-0.1/SINB)))

*--------diffuse PAR (PARDIF) and direct PAR (PARDIR)
         PARDIF = MIN (PAR, SINB*FRDIF*ATMTR*0.5*SC)
         PARDIR = PAR-PARDIF

         CALL ASSIM (SCV,AMAX,EFF,KDIF,LAI,SINB,PARDIR,PARDIF,FGROS)

*--------integration of assimilation rate to a daily total (DTGA)
         DTGA = DTGA+FGROS*WGAUSS(I)

10    CONTINUE

      DTGA = DTGA * DAYL

      RETURN
      END

*----------------------------------------------------------------------*
* SUBROUTINE ASSIM                                                     *
* Authors: Daniel van Kraalingen                                       *
* Date   : 10-Dec-1987                                                 *
* Modified by Jan Goudriaan 5-Febr-1988                                *
* Purpose: This subroutine performs a Gaussian integration over        *
*          depth of canopy by selecting three different LAI's and      *
*          computing assimilation at these LAI levels. The             *
*          integrated variable is FGROS.                               *
*                                                                      *
* FORMAL PARAMETERS:  (I=input,O=output,C=control,IN=init,T=time)     *
* name   meaning                                    units  class      *
* ----   -------                                    -----  -----      *
* SCV    Scattering coefficient of leaves for visible                 *
*          radiation (PAR)                            -      I        *
* AMAX   Assimilation rate at light saturation      kg CO2/  I        *
*                                                  ha leaf/h          *
* EFF    Initial light use efficiency              kg CO2/J/ I        *
*                                                  ha/h m2 s          *
* KDIF   Extinction coefficient for diffuse light            I        *
* LAI    Leaf area index                             ha/ha   I        *
* SINB   Sine of solar height                         -      I        *
* PARDIR Instantaneous flux of direct radiation (PAR) W/m2   I        *
* PARDIF Instantaneous flux of diffuse radiation(PAR) W/m2   I        *
* FGROS  Instantaneous assimilation rate of          kg CO2/  O        *
*        whole canopy                                ha soil/h         *
*                                                                      *
```

```
*  SUBROUTINES and FUNCTIONS called : none                            *
*  FILE usage : none                                                  *
*---------------------------------------------------------------------*

      SUBROUTINE ASSIM (SCV, AMAX, EFF, KDIF, LAI, SINB, PARDIR, PARDIF,
     &                  FGROS)
      IMPLICIT REAL(A-Z)
      REAL XGAUSS(3), WGAUSS(3)
      INTEGER I1, I2, IGAUSS
      SAVE

*-----Gauss weights for three point Gauss
      DATA IGAUSS /3/
      DATA XGAUSS /0.1127, 0.5000, 0.8873/
      DATA WGAUSS /0.2778, 0.4444, 0.2778/

*-----reflection of horizontal and spherical leaf angle distribution
      SQV  = SQRT(1.-SCV)
      REFH = (1.-SQV)/(1.+SQV)
      REFS = REFH*2./(1.+2.*SINB)

*-----extinction coefficient for direct radiation and total direct flux
      CLUSTF = KDIF / (0.8*SQV)
      KDIRBL = (0.5/SINB) * CLUSTF
      KDIRT  = KDIRBL * SQV

*-----selection of depth of canopy, canopy assimilation is set to zero
      FGROS = 0.

      DO 10 I1=1,IGAUSS
         LAIC = LAI * XGAUSS(I1)

*--------absorbed fluxes per unit leaf area: diffuse flux, total direct
*        flux, direct component of direct flux.
         VISDF = (1.-REFH)*PARDIF*KDIF  *EXP (-KDIF  *LAIC)
         VIST  = (1.-REFS)*PARDIR*KDIRT *EXP (-KDIRT *LAIC)
         VISD  = (1.-SCV) *PARDIR*KDIRBL*EXP (-KDIRBL*LAIC)

*--------absorbed flux (J/M2 leaf/s) for shaded leaves and assimilation
*        of shaded leaves
         VISSHD = VISDF + VIST - VISD
         IF (AMAX.GT.0.) THEN
            FGRSH  = AMAX * (1.-EXP(-VISSHD*EFF/AMAX))
         ELSE
            FGRSH = 0.
         END IF

*--------direct flux absorbed by leaves perpendicular on direct beam and
*        assimilation of sunlit leaf area

         VISPP  = (1.-SCV) * PARDIR / SINB
         FGRSUN = 0.
         DO 20 I2=1,IGAUSS
            VISSUN = VISSHD + VISPP * XGAUSS(I2)
            IF (AMAX.GT.0.) THEN
               FGRS = AMAX * (1.-EXP(-VISSUN*EFF/AMAX))
            ELSE
               FGRS = 0.
            END IF
            FGRSUN = FGRSUN + FGRS * WGAUSS(I2)
20       CONTINUE

*--------fraction sunlit leaf area (FSLLA) and
*        local assimilation rate (FGL)
         FSLLA = CLUSTF * EXP(-KDIRBL*LAIC)
         FGL   = FSLLA  * FGRSUN + (1.-FSLLA) * FGRSH

*--------integration of local assimilation rate
*        to canopy assimilation (FGROS)
         FGROS = FGROS + FGL * WGAUSS(I1)

10    CONTINUE
      FGROS = FGROS * LAI

      RETURN
      END
```

Data file: PLANT.DAT

```
*--------------------------------------------------
* data file 'plant.dat' : contains initial crop data
* wheat-model. these data are user or site specific.
* they can be altered.
*--------------------------------------------------

*-----initial number of tillers (main stems)
TLNI = 2.6E6  ! UNITS = TILLER/HA

*-----sowing density factor
SWDF = 1.0  ! UNITS = -

*-----absolute minimum concentration of nitrogen
*     in leaf blades
FNMIN = 0.005  ! UNITS = -

*-----initial fraction of reserves in plant organs
RESLI = 0.03  ! UNITS = -

*-----initial weight of leaves
WLVSI = 5.  ! UNITS = KG/HA

*-----initial weight of roots
WRTI = 5.  ! UNITS = KG/HA

*-----cumulative number of days with favourable conditions
*     for germination
TGERD = 0. ! UNITS = D

*-----total number of consecutive days with conditions
*     conducive for seed detioration
TSDAY = 0. ! UNITS = D

*-----sowing date (julian day number)
SOWD = 330. ! UNITS = -

*-----day number on which germination is completed
*     (julian day number)
GERDAT = 366. ! UNITS = -

*-----parameter to convert kj/m**2 to j/m**2
RADCF = 1.E3 ! UNITS = -

*-----parameter to account for expected temperature change
*     in the near future
EXPTF = 0. ! UNITS = -

*-----effect of evaporative cooling on canopy and soil
*     surface temperature

*-----magnitude of the effect
CTEMPF = 0. ! UNITS = GR.C

*-----reciprocal of mean standard canopy temperature
FTMPA = 0.07 ! UNITS = 1/GR.C

*-----daily rainfall
RAINF = 1. ! UNITS = MM

*-----circumference of a circle divided by its diameter
PI = 3.1416 ! UNITS = -

*-----drag coefficient of the leaves
DRAGC = 0.2 ! UNITS = -

*-----von karman's constant
KARMAN = 0.4 ! UNITS = -

*-----turbulence intensity
IW = 0.5 ! UNITS = -

*-----reference height for measuring wind speed
REFHT = 2. ! UNITS = M

*-----volumetric heat capacity of the air
RHOCP = 12.4 ! UNITS = J/M3/GR.K

*-----psychometric constant
PSCH = 0.67 ! UNITS = MBAR/GR.C

*-----minimum stomatal conductance, applicable under optimum
*     nitrogen conditions, value is equivalent to 0.625 cm/s
SCM = 5.4E4 ! UNITS = CM/D

*-----cuticular resistance for water flow, twenty times
*     stomatal resistance
RC = 1.15E-4 ! UNITS = D/CM

*-----extinction coefficient for global radiation
EXC = 0.5 ! UNITS = -

*-----height of the crop as a function of development stage
*     of the crop
CROHTB =  0.0, 0.05,
          0.5, 1.00,
          1.1, 1.00  ! UNITS: -

*-----functions describing the relation betwee average hourly
*     radiation intensity and alpha as a function of lai

*-----value of lai: 0
ALPHTA =  0., 1.00,
    41.820E5, 1.00  ! UNITS: -

*-----value of lai : 0.2
ALPHTB =  0., 1.00,
    41.820E5, 1.00  ! UNITS: -

*-----value of lai : 2.0
ALPHTC =  0., 0.000,
     4.182E5, 0.600,
     6.273E5, 0.660,
     8.364E5, 0.715,
    10.455E5, 0.760,
    12.546E5, 0.795,
    14.637E5, 0.835,
    16.728E5, 0.870,
    18.819E5, 0.910,
    20.910E5, 0.940,
    25.092E5, 0.970,
    41.820E5, 1.000  ! UNITS: -

*-----value of lai : 3.5
ALPHTD =  0., 0.000,
     4.182E5, 0.425,
     6.273E5, 0.515,
     8.364E5, 0.585,
    10.455E5, 0.640,
    12.546E5, 0.680,
    14.637E5, 0.715,
    16.728E5, 0.745,
    18.819E5, 0.770,
    20.910E5, 0.795,
    25.092E5, 0.845,
    41.820E5, 0.875  ! UNITS: -

*-----value of lai : 5.0
ALPHTE =  0., 0.000,
     4.182E5, 0.390,
     6.273E5, 0.455,
     8.364E5, 0.505,
    10.455E5, 0.545,
    12.546E5, 0.580,
    14.637E5, 0.610,
    16.728E5, 0.635,
    18.819E5, 0.660,
    20.910E5, 0.685,
    25.092E5, 0.740,
    41.820E5, 0.775  ! UNITS: -

*-----value of lai : 10.0
ALPHTF =  0., 0.000,
     4.182E5, 0.350,
     6.273E5, 0.410,
     8.364E5, 0.450,
    10.455E5, 0.485,
    12.546E5, 0.510,
    14.637E5, 0.530,
    16.728E5, 0.550,
    18.819E5, 0.565,
    20.910E5, 0.585,
    25.092E5, 0.610,
    41.820E5, 0.650  ! UNITS: -

*-----weight to conductivity ratio of the roots
WCRRT =  0.0,  50.,
         0.2, 100.,
         0.4, 200.,
         1.1, 400.  ! UNITS: KG/MM/D

*-----maximum reduction of development rate due to n
*     deficiency
NFDEV = 1.  ! UNITS: -

*-----parameter to define cultivar characteristic pre-
*     anthesis development rate
CULTP = 1.  ! UNITS: -

*-----parameter to define cultivar characteristic post-
*     anthesis development rate
CULTM = 1.  ! UNITS: -


*-----relevant development stages during pre-anthesis phase

*-----development stage at emergence,
DVSI = 0.      ! UNITS: -

*-----development stage at start of tillering
DVSTS = 0.     ! UNITS: -

*-----development stage at start of floral initiation
DVSSE = 0.22  ! UNITS: -

*-----development stage at start of spikelet differentiation
DVSSPS = 0.24 ! UNITS: -

*-----development stage at start of stem elongation
DVSST  = 0.35 ! UNITS: -

*-----development stage at start of floret formation
DVSFS  = 0.40 ! UNITS: -

*-----development stage at end of tillering
DVSPRE = 0.50 ! UNITS: -

*-----development stage at termination of spikelet
*     differentiation
DVSSPE = 0.52 ! UNITS: -

*-----development stage at end of floret formation
DVSFE = 0.60  ! UNITS: -

*-----development stage at anthesis
DVSAN = 1.00  ! UNITS: -


*-----relevant development stages during post-anthesis phase

*-----development stage at start of grain set
DVSGS = 0.01  ! UNITS: -

*-----development stage at start of grain filling
DVSSGF = 0.11  ! UNITS: -

*-----development stage at end of effective grain filling
DVSEGF = 0.70  ! UNITS: -

*-----initial light use efficiency at the light
*     compensation point
EFFE = 0.50 ! UNITS = KG/HA/H / J/M**2/S
```

```
*-----extinction coefficient for diffuse light
KDIF = 0.6 ! UNITS = -

*-----scattering coefficient of leaves for visible radiation (par)
SCV = 0.2 ! UNITS = -

*-----threshold level of reserves for reduction of gross
*     assimilation
TLRGA = 0.30 ! UNITS = KG/KG

*-----maintenance requirement factor leaf blades (at degr. 20 c)
RMRESL = 0.011 ! UNITS = KG CH2O/KG DM/D

*-----maintenance requirement factor stem and sheaths (at degr. 20 c)
RMRESS = 0.007 ! UNITS = KG CH2O/KG DM/D

*-----maintenance requirement factor roots (at degr. 20 c)
RMRESR = 0.005 ! UNITS = KG CH2O/KG DM/D

*-----maintenance requirement factor grains (at degr. 20 c)
RMRESG = 0.011 ! UNITS = KG CH2O/KG DM/D

*-----q10-factor for effect of temperature on maintenance
*     respiration
Q10  =  2. ! UNITS = -

*-----weight efficiencies for the formation of proteins
*     (assuming n as no3) from primary photosynthates
*     complementary fraction used for growth respiration
EFCPR  =  0.44 ! UNITS = KG/KG

*-----weight efficiencies for the formation of carbohydrates
*     from primary photosynthates complementary fraction used
*     for growth respiration
EFCCH  =  0.825 ! UNITS = KG/KG

*-----weight efficiencies for the formation of grain proteins
*     from primary photosynthates complementary fraction used
*     for growth respiration
EFCPRG =  0.69 ! UNITS = KG/KG

*-----time constant for death of plant tissue due to water
*     shortage
TCDDH = 6. ! UNITS = D

*-----fraction water in dying tissue
FWDB = 0.1 ! UNITS = KG/KG

*-----maximum relative death rate of leaf blades due to
*     shading values derived from puckridge & donald (1967)
MRDRSH = 0.03 ! UNITS = D**-1

*-----threshold value of beyond which death due to shading
*     starts
LAILM = 4. ! UNITS = M**2/M**2

*-----maximum relative death rate of leaf blades due to
*     carbohydrate shortage
MXRDR = 0.3 ! UNITS = D**-1

*-----average life span of the leaves at 15 deg c average
*     temperature
AVLTLF = 50. ! UNITS = D

*-----ratio of stem to leaf death, due to senescence assuming
*     half the "stem" to be sheaths, dying at half the
*     relative death rate of the blades
RSLDS = 0.25 ! UNITS = -

*-----proportionality factor between relative death rate of
*     stem and sheats and leaf blades
RFST = 0.60 ! UNITS = -

*-----proportionality factor between relative death rate of
*     roots and stem and sheaths
RFRT = 0.8 ! UNITS = -

*-----fraction of surplus carbohydrate flow diverted to root
*     growth
FSCHG = 0.8 ! UNITS = -

*-----extension rate of the roots under optimum conditions
DGRRT = 12. ! UNITS = MM/D

*-----factor accounting for the effect of stress on the rate
*     of root extension
RTF = 1. ! UNITS = MM

*-----rooting depth at emergence
IRTD = 80. ! UNITS = MM

*-----maximum depth of rooting
MXRTD = 1800. ! UNITS = MM

*-----limiting reserve concentration for translocation to
*     vegetative tissue before end of floret formation
RESL1 = 0.05 ! UNITS = KG/KG

*-----limiting reserve concentration for translocation to
*     vegetative tissue after end of floret formation
RESL2 = 0.15 ! UNITS = KG/KG

*-----time constant for translocation of reserves
TCTR = 2. ! UNITS = D

*-----residual reserve concentration in the tissue
FRNGL  = 0.0135 ! UNITS = KG/KG
FRNGL1 = 0.009 ! UNITS = KG/KG

*-----residual non-remobilizable concentration of reserves for
*     translocation to the grain
RESLR  = 0.010 ! UNITS = KG/KG

*-----function relating maximum photosynthetic capacity to
*     average daytime air temperature
TMPFT = -10., 0.00,
          0., 0.00,
         10., 1.00,
         25., 1.00,
         35., 0.01,
         50., 0.01  ! UNITS: -

*-----fraction of current assimilate to leaf blades, function
*     of development stage
FTLVST = 0.000, 0.475,
         0.050, 0.475,
         0.100, 0.575,
         0.175, 0.755,
         0.200, 0.755,
         0.250, 0.440,
         0.300, 0.250,
         0.350, 0.150,
         0.400, 0.120,
         0.450, 0.060,
         0.500, 0.000,
         1.100, 0.000  ! UNITS: -

*-----reduction factor for growth of leaf blades due to water
*     stress as a function of relative transpiration deficit
REDWST = 0.0, 1.,
         0.3, 1.,
         0.8, 0.,
         1.1, 0.  ! UNITS: -

*-----function relating reduction in growth of leaf blades to
*     nitrogen status of the blades
RFNST = 0.00, 0.0,
        0.80, 0.0,
        0.95, 1.0,
        1.10, 1.0  ! UNITS: -

*-----concentration of dry matter in plant material fraction of
*     current assimilate allocated to stem/sheaths, function of
*     development stage
FDMT = 0.0, 0.10,
       1.0, 0.25,
       1.1, 0.25  ! UNITS: -

*-----fraction of current assimilate supply allocated to
*     stem and sheaths as a function of development stage
FTSTET = 0.000, 0.000,
         0.175, 0.000,
         0.200, 0.045,
         0.250, 0.400,
         0.300, 0.600,
         0.350, 0.680,
         0.400, 0.650,
         0.450, 0.600,
         0.500, 0.450,
         0.550, 0.190,
         0.600, 0.000,
         1.100, 0.000  ! UNITS: -

*-----fraction of current assimilate supply allocated to
*     the roots as a function of development stage
*     (basic data from jonker,1966)
FDSRT = 0.000, 0.500,
        0.050, 0.500,
        0.100, 0.400,
        0.175, 0.220,
        0.200, 0.175,
        0.250, 0.135,
        0.300, 0.100,
        0.350, 0.075,
        0.400, 0.050,
        0.450, 0.030,
        0.500, 0.020,
        0.550, 0.010,
        0.600, 0.000,
        1.100, 0.000  ! UNITS: -

*-----reduction factor on root extension as a function
*     of soil temperature
REDTTB = -20., 0.,
          -1., 0.,
           0., 0.50,
           5., 0.80,
          10., 0.90,
          15., 1.00,
          20., 0.97,
          35., 0.97  ! UNITS: -

*-----fraction of current assimilate supply allocated to the
*     reserve pool as a function of developmental stage
FTRLT = 0.000, 0.025,
        0.200, 0.025,
        0.250, 0.025,
        0.300, 0.050,
        0.350, 0.095,
        0.400, 0.180,
        0.450, 0.310,
        0.500, 0.530,
        0.550, 0.800,
        0.600, 1.000,
        1.100, 1.000  ! UNITS: -

*-----potential growth rate of individual grains, function of
*     air temperature - data basically from sofield et al, 1977
PGRIGT = -10., 0.,
           0., 0.,
           8., 0.,
          10., 5.E-7,
          16., 1.35E-6,
          20., 1.65E-6,
          25., 1.85E-6,
          30., 2.E-6,
          35., 2.E-6  ! UNITS: KG DM/GRAIN/D

*-----basic carbohydrate requirement for tiller formation
CHFTB = 7.0E-4 ! UNITS = KG/TILLER

*-----relative death rate of non-earbearing tillers
RDRT = 0.2 ! UNITS = D**-1
```

```
*-----time constant for tiller formation, function of air
*     temperature data basically derived from friend, 1966
TCTFT = -10., 0.0,
         0., 20.0,
        10., 10.0,
        15.,  4.0,
        25.,  3.0,
        30.,  4.5,
        50., 10.0  ! UNITS: D

*-----basic carbohydrate requirement for ear formation
CHFEB = 1.4E-3 ! UNITS = KG/EAR

*-----basic time constant for ear formation
EB = 3. ! UNITS = D

*-----time constant for ear initiation
STCEF = 6. ! UNITS = D

*-----basic carbohydrate requirement for spikelet formation
CHFSB = 1.4E-4 ! UNITS = KG/SPIKELET

*-----basic time constant for spikelet formation
SB = 3. ! UNITS = D

*-----time constant for spikelet differentiation
STCSF = 6. ! UNITS = D

*-----basic carbohydrate requirement for floret formation
CHFFB = 3.5E-5 ! UNITS = KG/FLORET

*-----basic time constant for formation of florets
FB = 3. ! UNITS = D

*-----maximum value of time constant for formation of florets
STCFF = 6. ! UNITS = D

*-----effect of temperature on grain set, basic data from
*     hoshikawa(1960)
RFTT = -10., 0.,
        0., 0.,
        5., 1.,
       18., 1.,
       24., 1.,
       32., 0.,
       50., 0.  ! UNITS: -

*-----basic time constant for grain formation
GB = 2. ! UNITS = D

*-----maximum value of time constant for grain formation
STCGF = 4. ! UNITS = D

*-----first leaves thinner, later leaves thicker, when dvsv
*     gt dvsst, then flfar = flfarm, leaf area ratio, data
*     from aase, 1976 and others
FLFARM = 20. ! UNITS = M**2/KG LEAF BLADE WEIGHT

*-----specific leaf area at emergence
FLFARI = 30. ! UNITS = M**2/KG LEAF BLADE WEIGHT

*-----area per individual ear at maximum
*     effect of age on active ear area
ARPEAR = 2.5E-3 ! UNITS = M**2

*-----maximum stem area
MXSTAR = 3.E4 ! UNITS = M**2

*-----effect of development stage on green ear area
*     calculated basically from data by stoy, 1965
AGEFT = 0.00, 0.,
        0.35, 0.,
        0.40, 1.,
        0.80, 1.,
        0.85, 0.,
        1.10, 0.  ! UNITS: -

*-----ratio between minimum concentration of nitrogen for
*     unrestricted transpiration and maximum nitrogen
*     concentration of leaf blades, fraction derived
*     from data by yoshida & coronel for rice
FRNN = 0.33  ! UNITS: -

*-----parameter to activate uneven nitrogen distribution in
*     the crop, 1 gives even distribution, -1 heterogeneous
*     ! units: -
NDPAR = -1.  ! UNITS: -

*-----parameter to indicate degree of heterogenity
HTFAC = 1.  ! UNITS: -

*-----time constant for nitrogen uptake, range in maximum
*     fraction of n in leaf blade tissue, absolute minimum
*     fraction of n in leaf blade tissue
TCU = 1.  ! UNITS: D

*-----range in maximum nitrogen concentration between young
*     and mature leaf blades
FNMXA = 0.06  ! UNITS: KG/KG

*-----maximum nitrogen concentration in mature leaf blades
FNMXR = 0.01  ! UNITS: KG/KG

*-----range in optimum nitrogen concentration in the stem
*     lowest optimum nitrogen concentration in the stem
FNMXSA = 0.05  ! UNITS: KG/KG

*-----maximum nitrogen concentration in leaf blades,
*     function of development stage
FNMNSR = 0.003  ! UNITS: KG/KG

*-----maximum nitrogen concentration in young roots
FNRTMX = 0.030  ! UNITS: KG/KG

*-----maximum nitrogen concentration in mature roots
FNRTMN = 0.005  ! UNITS: KG/KG

*-----residual concentration of nitrogen in the root system
RN = 0.002  ! UNITS: KG/KG

*-----requirement for nitrogen uptake by diffusion, difference
*     between potential rate of uptake and uptake by mass flow,
*     tcud is a time constant for uptake by diffusion
TCUD = 1.5  ! UNITS: KG N/HA/D

*-----absolute maximum rate of uptake by the vegetation,
UMXR = 6.  ! UNITS: KG N/HA/D

*-----conversion factor from weight to relative root extension
*     it allows the plant to maintain frnv close to fnmx, in
*     the early stages of growth if n supply is adequate
CF = 100.  ! UNITS: KG/HA

*-----basic relative rate of turnover of nitrogen in vegetative
*     tissue
RRTORT = 0.25  ! UNITS: D**-1

*-----relative death rate of leaf blade tissue due to n
*     deficiency
RDRNT =   0., 1.,
       0.65, 1.,
       0.80, 0.,
        1.1, 0.  ! UNITS: D**-1

*-----residual level in leaf tissue as function of development
*     stage
BNT = 0.000, 0.0150,
      0.260, 0.0060,
      0.555, 0.0060,
      0.750, 0.0035,
      1.000, 0.0030,
      1.100, 0.0030  ! UNITS: KG/KG

*-----residual fraction of nitrogen in stems as function of
*     development stage
LNT = 0.000, 0.008,
      0.260, 0.005,
      0.555, 0.003,
      1.000, 0.002,
      1.100, 0.002  ! UNITS: KG/KG

*-----potential rate of nitrogen accumulation per individual
*     grain as function of average air temperature (sofield
*     et al., 1977)
PRNAGT = -10.0, 0.,
           0.0, 0.,
           7.0, 0.,
          10.0, 0.025E-6,
          12.5, 0.034E-6,
          18.5, 0.045E-6,
          27.5, 0.057E-6,
          35.0, 0.075E-6  ! UNITS: KG/HA/GRAIN

*-----function relating fraction exported nitrogen to average
*     nitrogen concentration in vegetative tissue, data derived
*     from dalling, 1976
FNEXT = 0.0000,  0.00,
        0.0025,  0.00,
        0.0070,  0.15,
        0.0120,  0.25,
        0.0160,  0.22,
        0.0200,  0.20,
        0.0250,  0.18,
        0.0375,  0.12,
        0.0700,  0.08 ! UNITS: KG/KG
```

Data file: SOIL.DAT

```
*-------------------------------------------------
* data file 'soil.dat' : contains initial soil data
* for wheat-model. these data are user or site
* specific. they can be altered.
*-------------------------------------------------

*-----number of soil compartments
ILAYER = 10   ! UNITS = -

*-----thickness of consecutive compartments
THCKN = 20.,30.,50.,2*100.,5*300.   ! UNITS = MM

*-----initial dryness factor of consecutive compartments
*     as a fraction of moisture content at wilting point
DRFA = 0.5, 0.75, 0.8, 0.9, 2*1., 1.35, 1.5, 1.6, 1.7 ! UNITS = -

*-----initial total amount of fresh organic material and
*     stable organic material in soil profile
FOMI = 3000.    ! UNITS = KG/HA

*-----initial total amount of stable organic material in
*     soil profile
HUMI = 28000.   ! UNITS = KG/HA

*-----initial total amounts of nitrogen in stable organic
*     material
NHUMI = 2800.   ! UNITS = KG/HA

*-----initial amount of nitrogen in microbial biomass per
*     compartment
IBIOMN = 6*1., 4*0.   ! UNITS = KG/HA

*-----initialization of the amount of inorganic nitrogen
*     in soil compartments
IAS = 20., 30., 56., 12., 12., 120., 15., 3*0.   ! UNITS = KG/HA

*-----initialization of the distribution of organic matter
*     in the soil identical for stable and fresh organic
*     material
DFFOM = 0.05, 0.075, 0.125, 0.21, 0.17, 0.37, 4*0. ! UNITS = -

*-----total daily visible radiation on completely clear days
*     as a function of latitude and day of the year, cal/cm**2

*-----latitude = 0.0 degrees
RADTBA =
  0.,340.,  15.,343.,  46.,360.,  74.,369., 105.,364.,
135.,349., 166.,337., 196.,342., 227.,357., 258.,368.,
288.,366., 319.,349., 349.,337., 366.,340.

*-----latitude = 10.0 degrees
RADTBB =
  0.,295.,  15.,299.,  46.,332.,  74.,359., 105.,375.,
135.,377., 166.,374., 196.,375., 227.,377., 258.,369.,
288.,345., 319.,311., 349.,291., 366.,294.

*-----latitude = 20.0 degrees
RADTBC =
  0.,243.,  15.,249.,  46.,293.,  74.,337., 105.,375.,
135.,394., 166.,400., 196.,399., 227.,386., 258.,357.,
288.,313., 319.,264., 349.,239., 366.,241.

*-----latitude = 30.0 degrees
RADTBD =
  0.,185.,  15.,191.,  46.,245.,  74.,303., 105.,363.,
135.,400., 166.,417., 196.,411., 227.,384., 258.,333.,
288.,270., 319.,210., 349.,179., 366.,183.

*-----latitude = 40.0 degrees
RADTBE =
  0.,124.,  15.,131.,  46.,190.,  74.,260., 105.,339.,
135.,396., 166.,422., 196.,413., 227.,369., 258.,298.,
288.,220., 319.,151., 349.,117., 366.,122.

*-----latitude = 50.0 degrees
RADTBF =
  0.,67.,   15., 73.,  46.,131.,  74.,207., 105.,304.,
135.,380., 166.,418., 196.,405., 227.,344., 258.,254.,
288.,163., 319., 92., 349., 61., 366., 66.

*-----latitude = 60.0 degrees
RADTBG =
  0., 18.,  15., 22.,  46., 72.,  74.,149., 105.,260.,
135.,356., 166.,408., 196.,389., 227.,309., 258.,201.,
288.,103., 319., 37., 349., 14., 366., 17.

*-----latitude = 70.0 degrees
RADTBH =
  0.,  0.,  15.,  0.,  46., 20.,   74., 89., 105.,209.,
135.,331., 166.,408., 196.,380.,  227.,269., 258.,142.,
288., 45., 319.,  2., 349.,  0.,  366.,  0.

*-----latitude = 80.0 degrees
RADTBI =
  0.,  0.,  15.,  0.,  46.,  0.,  74., 28., 105.,162.,
135.,334., 166.,424., 196.,380., 227.,248., 258., 81.,
288.,  3., 319.,  0., 366.,  0.

*-----latitude = 90.0 degrees
RADTBJ =
  0.,  0.,  15.,  0.,  46.,  0.,  74.,  0., 105.,154.,
135.,339., 166.,428., 196.,393., 227.,252., 258., 40.,
288.,  0., 366.,  0.

*-----converts cal/cm**2 to j/m**2
CONVER = 4.182E4 ! UNITS = -

*-----converts from m/s to km/d
WSCONV = 86.4 ! UNITS = -

*-----reflection coefficient of open water for short
*     wave radiation
REFCF = 0.05 ! UNITS = -

*-----stefan-bolzmann constant
STBC = 4.93E-3 ! UNITS = J/M**2/S/K**4

*-----psychometric constant
GAMMA = 0.49 ! UNITS = MM Hg/GR.C

*-----latent heat of vapourization
LHVAP = 2.390E6 ! UNITS = J/KG

*-----function relating reduction in soil evaporation to
*     dimensionless water content in top soil compartment
REDFDT = -0.50,  0.00,
          0.00,  0.00,
          0.20,  0.05,
          0.22,  0.27,
          0.33,  0.90,
          1.00,  1.00,
          1.10,  1.00 ! UNITS = -

*-----runoff coefficient
RUNOFC = 0. ! UNITS = -

*-----effective irrigation as a function of day number
IRRT =  0., 0.,
      366., 0. ! UNITS = -

*-----proportionality factor soil evaporation
PROP = 15. ! UNITS = -

*-----volumetric soil moisture content at air dryness
WCLIM = 0.025 ! UNITS = CM**3/CM**3

*-----volumetric soil moisture content at wilting point
WLTPT = 10*0.075 ! UNITS = CM**3/CM**3

*-----volumetric soil moisture content at field capacity
FLDCP = 10*0.23 ! UNITS = CM**3/CM**3

*-----function relating actual rate of water uptake to
*     relative water content in compartment
WREDT = 0.00, 0.000,
        0.10, 0.300,
        0.15, 0.450,
        0.30, 0.700,
        0.50, 0.975,
        0.75, 1.000,
        1.10, 1.000 ! UNITS = -

*-----function relating effectiviness of roots to moisture
*     content in compartment
EDPTFT = 0.00,  0.15,
         0.15,  0.60,
         0.30,  0.80,
         0.50,  1.00,
         1.10,  1.00 ! UNITS = -

*-----function relating root water uptake to osmotic pressure
*     in the soil
ROSPT = 0.,   1.0,
       45.,   0.1,
       50.,   0.0,
      200.,   0.0 ! UNITS = -

*-----relative rate of decomposition of lignin under
*     optimum conditions
RDLIGN = 0.0095   ! UNITS = D**-1

*-----relative rate of decomposition of cellulose and
*     hemicellulose under optimum conditions
RDCELL = 0.05   ! UNITS = D**-1

*-----relative rate of decomposition of easily decomposable
*     carbohydrates and proteins under optimum conditions
RDCAPR = 0.8   ! UNITS = D**-1

*-----fraction of carbon in fresh organic material
FRC = 0.4   ! UNITS = KG/KG

*-----fraction of nitrogen in fresh organic material
FRNF = 0.01   ! UNITS = KG/KG

*-----composition of fresh organic material,lignin,cellulose
*     /hemicellulose, carbohydrate/proteins

*-----fraction of lignin in initial fresh organic material
FLIGN = 0.1   ! UNITS = KG/KG

*-----fraction of cellulose and hemicellulose in initial
*     fresh organic material
FCELL = 0.7   ! UNITS = KG/KG

*-----c/n ratio of microbial biomass
CNRMIC = 8.   ! UNITS = KG/KG

*-----time constant for microbial growth
TCMG = 2.   ! UNITS = D

*-----relative rate of maintenance respiration of microbial
*     biomass
RRMIC = 0.10   ! UNITS = D**-1

*-----maximum relative growth rate for microbial biomass
MRGRB = 1.   ! UNITS = D**-1

*-----nitrogen content of stable organic material
NCH = 0.04   ! UNITS = KG/KG

*-----relative decomposition rate of stable organic material
*     under optimum conditions
DMINR = 8.3E-5   ! UNITS = D**-1

*-----fraction of n in decomposing fresh organic material and
*     microbial biomass immobilised in stable organic material
FNIMH = 0.5   ! UNITS = KG/KG

*-----concentration of nitrogen in rainwater, kg/mm/ha
NCR = 0.02   ! UNITS = KG/HA/MM
```

```
*-----concentration of nitrogen in irrigation water
NCIW = 0.02   ! UNITS = KG/MM/HA

*-----parameter to designate type of fertilizer:
*     1 = nh4, 0 = no3
NH4FP = 1.   ! UNITS = -

*-----time constant for volatilization of ammonia
TCV = 10.   ! UNITS = D

*-----limiting concentration for volatilization of ammonia
LNH4 = 2.2   ! UNITS = KG/MM

*-----time constant for nitrification
TCN = 10.   ! UNITS = D

*-----function relating rate of decomposition of organic
*     material to moisture content in compartment
*     a. according to beek&frissel
MFT = 0.00,  0.000,
      0.14,  0.002,
      0.40,  0.320,
      0.60,  0.650,
      0.70,  0.800,
      0.85,  1.000,
      1.00,  0.850,
      1.10,  0.850   ! UNITS = D

*-----b. according to robinson (1957)
XMFT = 0.00,  0.000,
       0.14,  0.002,
       0.40,  0.320,
       0.60,  0.650,
       0.70,  0.800,
       0.85,  1.000,
       1.00,  0.850,
       1.10,  0.850 ! UNITS =

*-----vertical fertilizer distribution factor
DISTF = 1.,9*0.   ! UNITS = -


*-----effect of soil temperature on decomposition of
*     organic material
TFT = -10., 0.000,
        0., 0.001,
        5., 0.150,
       10., 0.300,
       20., 0.640,
       30., 0.810,
       40., 0.960,
       50., 1.000   ! UNITS = -

*-----effect of soil temperature on root conductivity
TECT = -10., 0.00,
         0., 0.06,
         3., 0.29,
        10., 0.85,
        16., 0.94,
        18., 1.00,
        22., 1.00,
        31., 0.87,
        40., 0.60,
        50., 0.30   ! UNITS = -

*--------------------------------------------------------------
* variables for nitrogen management
*--------------------------------------------------------------

*-----number of fertilizer applications
NF = 3   ! UNITS = -

*-----day number of nitrogen application,
*     simulation time
NAPDAY = 3*0.  ! UNITS = D

*-----nitrogen fertilizer application rate
NTRMNT = 3*0.  ! UNITS = KG/HA/D
```

Data file: TIMER.DAT

```
*---------------------------------------------------
* data file 'timer.dat' : contains initial time data
* for wheat-model, these data are user specific, they
* can be altered.
* put the directory where your weather data are stored
* behind the first exclamation mark and the country
* code behind the second. these variables cannot be
* used in rerun files
* do not use blank lines between this comment block
* and the two lines with the exclamation mark !
*----------------------------------------------------
!
!ISR

*-----station number
ISTN = 1

*-----timer variables
*-----start year of simulation
IYEAR = 1979

*-----start day of simulation, julian day number
DAYB = 330.

*-----day at end of simulation (simulation time)
FINTIM = 1000.

*-----print interval, d
PRDEL = 10.

*-----timestep for integration, d
DELT = 1.

*-----format of output-file:
*     4: normal table
*     5: tab-delimited table (for excel)
*     6: ttplot format
ITABLE = 4

*-----switch variable; delete temporary output-
*     file (1) or not (0)
IDTMP = 1

*-----list of harvest data for which output is desired
*     0 = no harvest data
HARDAY = 0.
```

Data file: RERUNS.DAT

```
IYEAR = 1986 ; NAPDAY = 45.,130.,150. ; NTRMNT = 80.,60.,40.
IYEAR = 1986 ; NAPDAY = 45.,130.,150. ; NTRMNT = 3*60.

IYEAR = 1987 ; NAPDAY = 45.,130.,150. ; NTRMNT = 80.,60.,40.
IYEAR = 1987 ; NAPDAY = 45.,130.,150. ; NTRMNT = 3*60.
```

Output file: WHEAT.OUT

```
*--------------------------------------------------------------------------
* Output table number  :  0 (=first output table)
* Output table format  : Table output
* SPRING-WHEAT simulation model
```

TIME	DAY	GRAI	TADRW	WLVS	WSTEM	WGR
.000000	330.00	.00000	.00000	.00000	.00000	.00000
10.0000	340.00	1.50000E-02	5.1500	5.0000	.00000	.00000
20.0000	350.00	5.98239E-02	22.024	21.064	.00000	.00000
30.0000	360.00	.25857	105.75	101.49	.00000	.00000
40.0000	5.0000	1.0220	472.05	454.71	.00000	.00000
50.0000	15.000	2.8796	1496.6	1372.7	70.720	.00000
60.0000	25.000	4.8204	3108.7	2242.9	719.93	.00000
70.0000	35.000	5.8923	5116.1	2564.2	1922.3	.00000
80.0000	45.000	6.2987	7116.8	2292.4	3152.3	.00000
90.0000	55.000	7.0101	9288.5	1918.3	4476.4	.00000
100.000	65.000	7.8861	11518.	1874.1	5759.4	.00000
110.000	75.000	8.7222	14018.	1878.7	6800.8	.00000
120.000	85.000	8.4167	16159.	1697.4	6896.0	1224.7
130.000	95.000	5.0597	16674.	556.94	5102.6	3440.0
140.000	105.00	2.0390	16405.	134.46	2516.6	5178.3
150.000	115.00	.22715	16394.	.00000	378.58	5178.3
157.000	122.00	2.08453E-02	16394.	.00000	34.742	5178.3