This project is part of the PRIMA programme supported by the European Union

PRIMA-SECTION 2-2022

"Modelling and Technological Tools to Prevent Surface and Ground-Water Bodies from Agricultural Non-Point Source Pollution Under Mediterranean Conditions"

NPP-SOL

FLOWS software and related handbook

Deliverable number: D4.1

This project is part of the PRIMA programme supported by the European Union

Non-Point Pollution SOLutions (NPP-SOL)

Deliverable 4.1: FLOWS software and related handbook

Author(s)	Antonio Coppola, Shawkat B.M. Hassan (UNIBAS)
Approved by work package 4 leader	Antonio Coppola (University of Basilicata)
Date of approval	01/01/2024
Approved by project coordinator	Antonio Coppola (University of Basilicata)
Date of approval	01/01/2024
Call	PRIMA Section 2 2022- Multi-topic Topic 2.1.1 RIA (Prevent and reduce land and water salinization and pollution due to agri-food activities).
Project website	https://mel.cgiar.org/projects/npp-sol

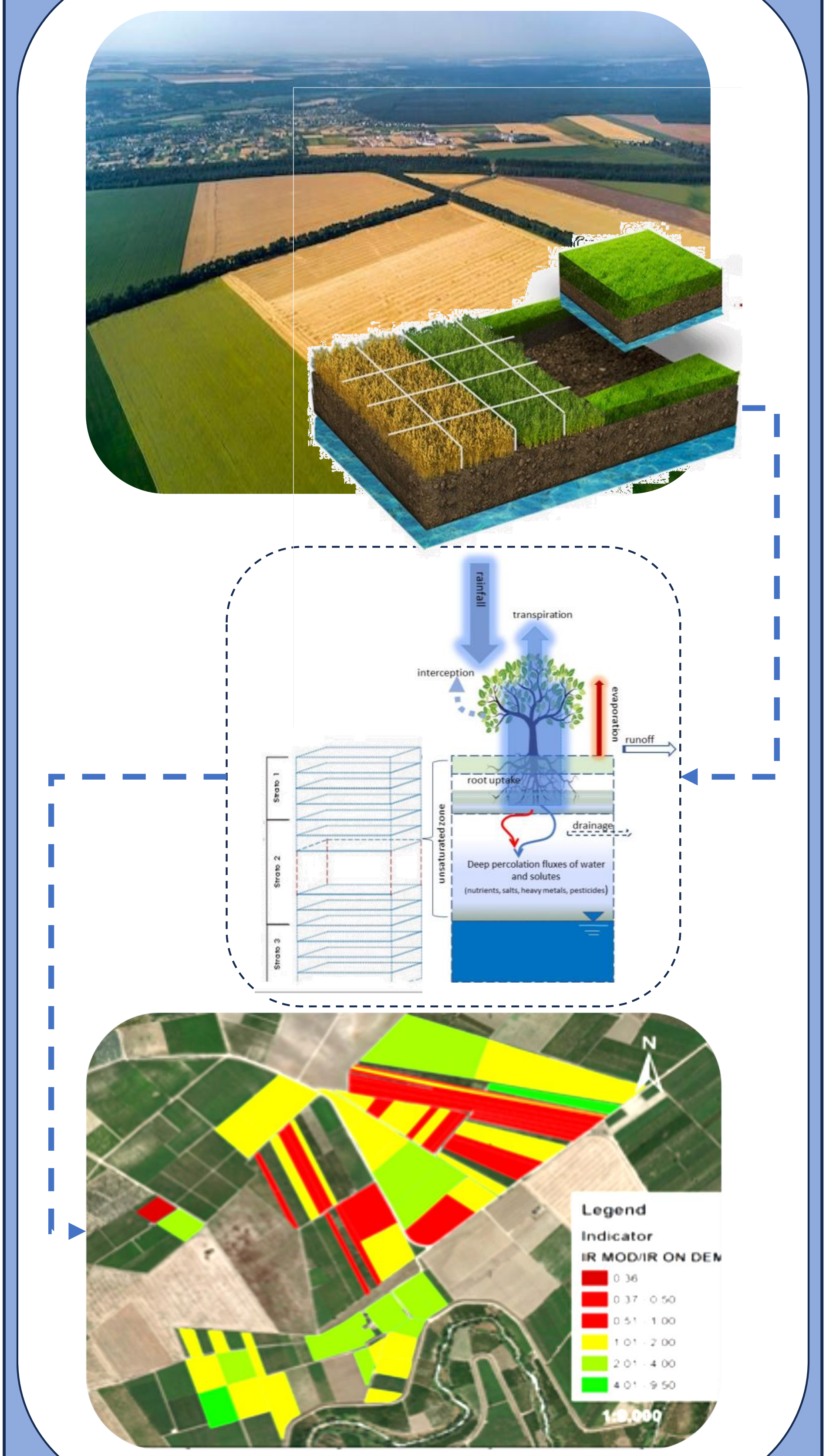
NPP-SOL

PRIMA
PARTNERSHIP FOR RESEARCH AND INNOVATION IN THE MEDITERRANEAN AREA

This project is part of the PRIMA programme supported by the European Union

FLOWS - Flow of Water and Solutes in Agricultural and Environmental Systems

Version 3.0 Handbook



This project is part of the PRIMA programme supported by the European Union

FLOWS - FLOw of Water and Solutes in agricultural and environmental systems

Version 3.0

Theory and user manual

<u>Code development</u>

Antonio Coppola (code versions 1.0, 2.0 and 3.0)

Loredana Randazzo (code version 1.0)

Caterina Fenu and Anna Concas (code optimization and graphical interface – version 3.0)

<u>Code testing and benchmarking</u>

Antonio Coppola

Angelo Basile

Giovanna Dragonetti

Alessandro Comegna

Shawkat Basel Mostafa Hassan

Units for input data

Mass=g

Length=cm

Time= hour or *days*

Pressure heads=cm water column

Table of Contents

Abstract

FLOWS (FLOws of Water and Solutes in soils is a dynamic physically-based model to simulate water flow and solute transport in the soil-plant-atmosphere system. The numerical code was written in Matlab. One-dimensional vertical transient water flow is simulated by numerically solving the 1D form of the Richards equation (RE) using an implicit, backward, finite differences scheme with explicit linearization. As for the solutes, the code solves the 1D Advection Dispersion equation (ADE) by an explicit, central difference scheme, allowing for a relatively easy inclusion of non-linear adsorption and other non-linear processes. The model produces information on the time evolution of (among many other outputs): Soil water contents and pressure potentials in the soil profile; Solute (tracers, adsorbed, volatile and reactive solutes) concentrations in the soil profile; Water and solute uptake and actual evapotranspiration for simulated crops; Deep percolation water and solute fluxes; Root uptake of water and solutes, also in presence of water and osmotic stresses; Water fluxes and related solute fluxes to runoff; Drainage water fluxes and related solute fluxes; Irrigation fluxes computed by the model; Temperature in the soil profile. Compared to other existing models, some specificities of FLOWS model are: 1) integrated simulation of carbon, nitrogen and phosphorus transformations and transport contextually to the simulation of water contents and fluxes; 2) hysteresis in the macroscopic root uptake; 3) irrigation management, by allowing to optimize the time scheduling and volumes according to some criterion based on the average pressure head in the root zone; 4) the possibility of running multi-site simulations, which may be especially important for spatially distributed simulations; 5) an interactive graphical user interface, allowing the user to easily handle even large input files and seeing the simulation results by graphical display of the main output variables..

1. Introduction

The movement of water in the soil and associated solute transport have a role of primary importance in many applications in the field of hydrology and agriculture. In the sound management of irrigation water, in relation to specific environmental conditions and cropping systems, knowledge of local water flow conditions in zones explored by the root systems is indispensable. Once the irrigation method has been established, only knowledge of the laws governing water flow allows the necessary irrigation frequencies and rates to be established to optimise the distribution of soil moisture, reducing within established limits the effects of water stress and containing water wastage. Only by studying water dynamics in soil can the contribution of groundwater to water consumption be quantitatively determined. Moreover, the water volumes infiltrating into the soil due to rainfall are strictly linked and governed by the laws of water flow in the soil. No evaluation of water quantities being added to groundwater circulation can be made without first determining the water volumes moving in the zone between the soil surface and the aquifer.

Knowledge of water fluxes, velocities and contents is also indispensable to study the flow of solutes and pollutants and to predict all exchanges, whether chemical or microbiological, occurring in the soil. Within environmental protection initiatives, great weight is attributed to the continuous contribution of solutes characterising any agricultural activity, especially if intensive. Nor should one underestimate the hazards of non-agricultural solute fluxes, and the need to dispose of wastewater, muds and industrial effluent.

To evaluate the nature of the risk represented the presence of these solutes it is important to define the processes governing their movement downward from the soil surface through the vadose zone as far as the aquifer. Only if we know such transport processes can optimal management plans be developed for environmental control with a view to preventing degradation phenomena.

The complexity of such flow processes has encouraged the widespread use in the sector of soil hydrology increasingly sophisticated mathematical models corresponding as closely as possible to real phenomena, which can supply quantitative evaluations also in the presence of highly complex systems, such as soil-plant-atmosphere continuum (SPAC).

In general, such models are based on laws governing water flow and all the physical and chemical processes affecting water transport. These laws have been widely confirmed by experiments and are often reported also in papers in disciplines collateral to soil hydrology (Bear, 1979; Jensen, 1980). Water flow is studied with reference to a porous medium which, on a macroscopic scale, may be considered continuous and in which the various quantities and physical properties are considered functions of position and time. Reference is generally made to isothermal flow processes, to a Newtonian liquid phase and interconnected gaseous phase with a pressure equal to that of the atmosphere. Moreover, due to relatively low resistance to air flow, the movement of the gaseous phase is neglected and only water flow is referred to.

FLOWS is a Dynamic Physically-Based model, written in Matlab, just based on the governing laws and assumptions described above. Specifically, it simulates water flow and solute transport in the soil-plant-atmosphere system by numerically solving the 1D form of the Richards equation (RE) for water flow and the 1D Advection Dispersion equation (ADE) for solute transport in presence of decay and adsorption.

The model produces information on the time evolution of (see figure 1a,b):

- Soil water contents and pressure potentials in the soil profile;
- Solute (tracers, adsorbed and reactive solutes) concentrations in the soil profile. The model also allows for: 1) Simulating Organic Matter decomposition and Carbon, Nitrogen and Phosphorus transport. In this case, carbon, nitrogen and phosphorus transformations are all controlled by the dynamics of the organic matter decomposition and, thus, also by the C:N and C:P ratios; 2) Simulating only Nitrogen transport. In this second case, nitrogen mineralization is simulated as an empirical decay reaction and

independently on organic matter decomposition dynamics, thus without accounting for the C:N ratio in the organic matter.

- Water and solute uptake and actual evapotranspiration for simulated crops;
- Deep percolation water and solute fluxes;
- Root uptake of water and solutes;
- Water fluxes and related solute fluxes to runoff;
- Drainage water fluxes and related solute fluxes.
- Irrigation fluxes computed by the model;
- Temperature in the soil profile

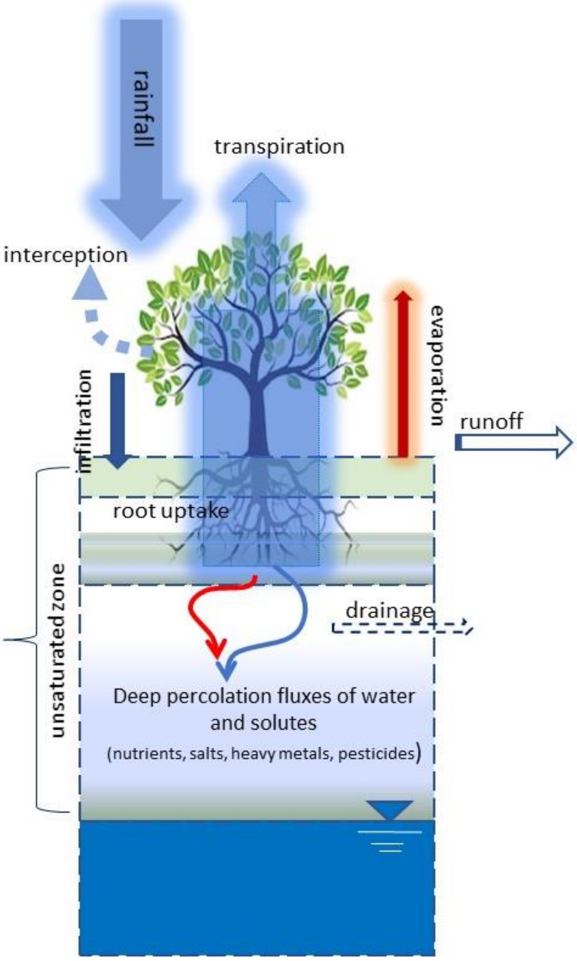
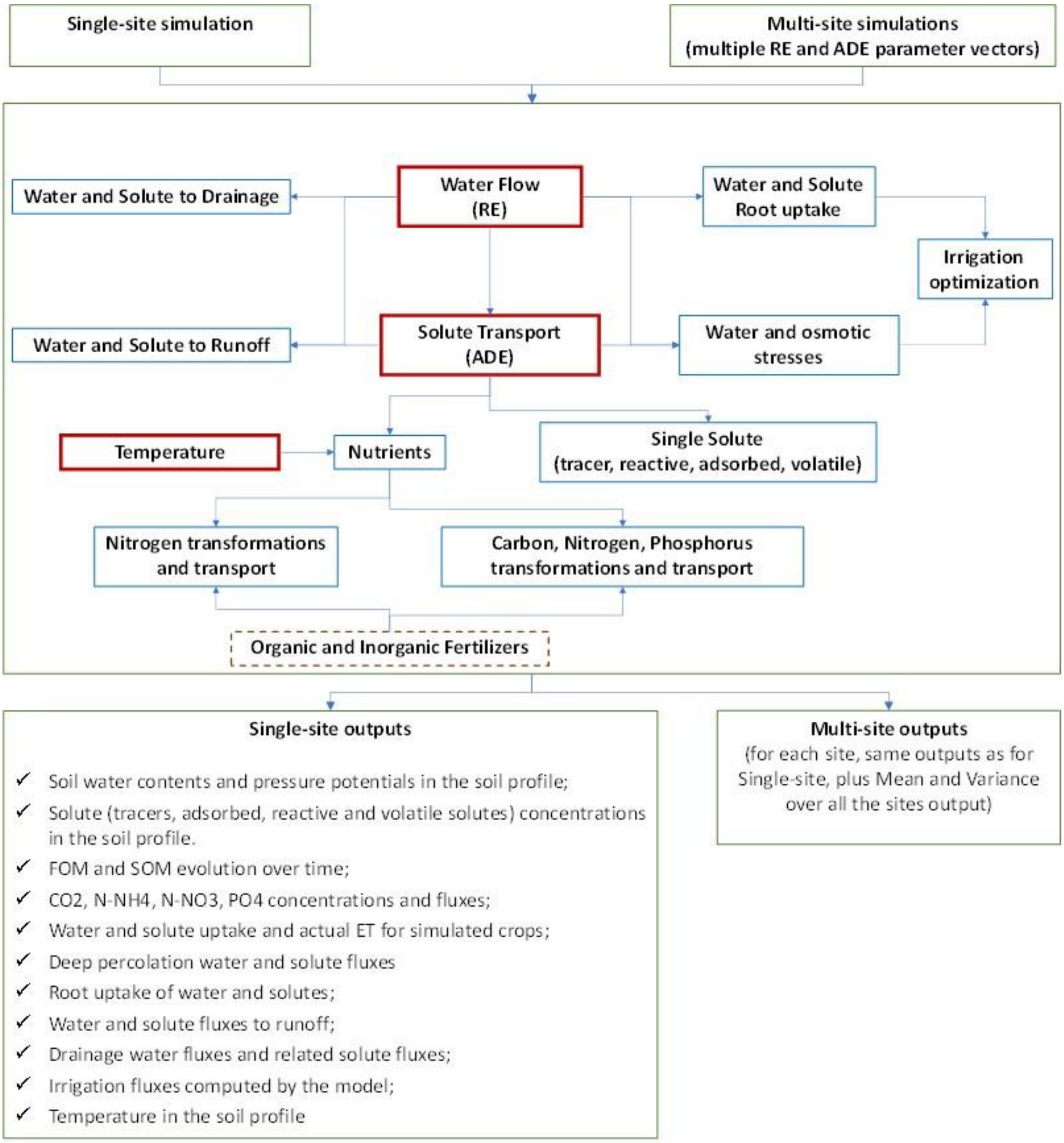


Figure 1a. Schematic view of water flow and solute transport processes simulated by FLOWS


Single-site simulation
Multi-site simulations (multiple RE and ADE parameter vectors)
Water and Solute to Drainage
Water Flow (RE)
Water and Solute Root uptake
Irrigation optimization
Water and Solute to Runoff
Solute Transport (ADE)
Water and osmotic stresses
Temperature
Nutrients
Single Solute (tracer, reactive, adsorbed, volatile)
Nitrogen transformations and transport
Carbon, Nitrogen, Phosphorus transformations and transport
Organic and Inorganic Fertilizers

Single-site outputs

- ✓ Soil water contents and pressure potentials in the soil profile;
- ✓ Solute (tracers, adsorbed, reactive and volatile solutes) concentrations in the soil profile.
- ✓ FOM and SOM evolution over time;
- ✓ CO2, N-NH4, N-NO3, PO4 concentrations and fluxes;
- ✓ Water and solute uptake and actual ET for simulated crops;
- ✓ Deep percolation water and solute fluxes
- ✓ Root uptake of water and solutes;
- ✓ Water and solute fluxes to runoff;
- ✓ Drainage water fluxes and related solute fluxes;
- ✓ Irrigation fluxes computed by the model;
- ✓ Temperature in the soil profile

Multi-site outputs
(for each site, same outputs as for Single-site, plus Mean and Variance over all the sites output)

Figure 1b. Flow chart of all the processes simulated in FLOWS, their interactions and simulation options. The model provides all the outputs listed in the lower box. In the case of multiple simulations (for example spatially distributed simulations or repeated simulations in stochastic frameworks), all the outputs are provided for each of the input parameter vectors, along with the average and variance of each output.

The user may opt for several hydraulic property models: 1) unimodal van Genuchten-Mualem model; 2) bimodal Durner-Mualem model; 3) bimodal Ross & Smettem-Mualem model; Russo-Gardner model.

As for water flow, the model allows to set several top and bottom boundary conditions, both constant or variable over time: potential and fluxes at the upper boundary, potential, fluxes and hydraulic gradient at the bottom boundary. Initial conditions may be given as pressure heads, which can be either constant along the soil profile or variable node by node.

As for solute transport, the model allows for either constant or variable solute application at the soil surface. Also, initial conditions may be either constant or variable node by node. Dispersivity and decay may also be given for each node. As for sorption, the model allows for both linear and Freundlich isotherms. In this case, the user has to input the parameters of the sorption isotherm. The model also describes the solute flow in gaseous phase.

In the case of nitrogen transport, the model allows the user providing several forms (both organic and mineral) of nitrogen fertilizers: 1) Manure; 2) Cover crops; 3) Urea; 4) N-NH4 and N-NO3 solid and liquid fertilizers. In this case, the user has to input the fertilization depth (all the nitrogen fertilizers are assumed to be incorporated uniformly within this depth). Organic nitrogen forms are assumed to consist of both slow and rapid mineralization fractions, as well as passive (inert) fractions. The model allows for nitrification, denitrification and immobilization coefficients to be given node by node in the whole simulation domain.

The crop is simulated in a so-called static way, so that the crop growth is not simulated dynamically by the model but the user has to specify the crop development stage by giving as input the evolution over time of the leaf area index, root depth, reference evapotranspiration, as well as the crop coefficient as a function of development stage to convert reference evapotranspiration to the potential evapotranspiration of the considered crop.

With this approach, the model “sees” the crop as a root system drawing water from the soil profile according to the atmospheric water vapour demand and the soil water availability, and as a soil cover which partitions the evapotranspiration in evaporation and transpiration components, and that partly intercepts rainfall or irrigation water.

The model computes internally actual evaporation and transpiration. Transpiration is distributed in a node-by-node root uptake in the root zone according to the root distribution. Several root distributions may be adopted: 1) Uniform; 2) Triangular (Prasad); 3) Vrugt; 4) Logistic. The actual root uptake accounts for either water stress or salinity stress or both and is calculated by reducing the potential transpiration according to a reduction function based on the water potential and the osmotic potential. The user may select different options for water uptake reduction: 1) Feddes reduction function for water stress; 2) van Genuchten reduction function for water stress; 3) Mass and Hoffman reduction function for osmotic stress; 4) van Genuchten reduction function for osmotic stress; 5) multiplicative combinations of water and osmotic reduction function as described above.

The irrigation volumes may be computed by the model according to a criterion based on the average pressure head in the root depth. The model allows the user for selecting several irrigation periods during the crop growth season. Moreover, the user has to input the irrigation depth and the critical pressure head to start irrigation. Irrigation volumes may also be provided by the user. In this case, the code sums up the irrigation fluxes to the natural rain fluxes.

The drainage is included as a sink term based on the Hooghoudt theory for lateral flow and transport to the drains.

The runoff comes from both a Dunnian and Hortonian runoff production mechanisms. The model also calculates solute concentrations in the runoff by a physically based model for predicting solute transfer from soil solution to runoff.

The time-scale of the model may be minutes, hours, days.

All the evaluations above can be carried out in a stochastic (Montecarlo) framework, thus providing uncertainties bands for each of the main outputs.

An intercomparison with Hydrus 1D model provides a benchmarking of the FLOWS model.

The specificity of FLOWS is that the user is provided with a compact model interface (a single window) where all the input data and parameters may be set. In a single window, the user visualizes the whole simulation configuration. The interface guides the user in all the settings by providing suggestions and warnings, making settings faults unlikely. This way, the model allows for running simulations even users not perfectly within the field of physically based hydrological models.

All the input (node and time conditions, profile settings, vegetation settings, solute settings) may be provided as excel tables which have simply to be uploaded from the main interface.

All the outputs are also provided as excel files, each including a matrix with the evolution of the output variable given for all the depth nodes and all the time steps. Additionally, at the end of each simulation run, the model produces plots of the water content, pressure heads, water fluxes and solute concentrations for selected times and depths, allowing a rapid evaluation of the results and goodness of the simulation run.

Another important feature of FLOWS concerns the nitrogen transformations and transport, which are simulated in an integrated way, so that at each simulation time step the water contents and fluxes calculated by solving the Richards equation are used as input for the nitrogen equations.

2. Water flow in soil

One-dimensional vertical transient water flow in this model is simulated by numerically solving the 1D form of the Richards equation (equation 1) using an implicit, backward, finite differences scheme with explicit linearization similar to that adopted in the SWAP model (van Dam et al., 1997):

$$C(h)\frac{\partial h}{\partial t}=\frac{\partial}{\partial z}\left(K(h)\frac{\partial h}{\partial z}-K(h)\right)-S_w \qquad (1)$$

where C(h)=dθ/dh [L^{-1}] is the soil water capacity, h [L] is the soil water pressure head, t [T] is time, z [L] is the vertical coordinate being positive upward, *K*(h) [L T^{-1}] the hydraulic conductivity and S_w [T^{-1}] is a sink term describing water uptake by plant roots, S_r, and/or lateral water drainage, S_{dr}, so that S_w=S_r+S_{dr}.

2.1. Hydraulic properties

Richards' equation requires the water retention function, *θ(h),* and the hydraulic conductivity, *K(h)* or *K(Se)*, function to be known. These functions interrelate pressure head, *h*, water content, *θ*, and hydraulic conductivity, *K*. Several water retention and hydraulic conductivity functions can be selected in the model.

2.1.1. Water retention

Unimodal van Genuchten model (van Genuchten, 1980):

$$S_e=\frac{\theta-\theta_r}{\theta_s-\theta_r}=[1+|\alpha_{VG}h|^n]^{-m} \qquad h<0$$
$$\theta=\theta_s \qquad h\geq 0 \qquad (2)$$

where *h* is the pressure head (*h*≤0), S_e is effective saturation and θ is the water content (θ_s and θ_r are the water content at *h*=0 and for *h*→∞, respectively). α_{VG} (cm^{-1}), *n* and *m*=1-1/*n* are shape parameters. Note the effective saturation, S_e, is to be considered as a cumulative distribution function of pore size with a density function *f(h)* which may be expressed by:

$$f(h)=\frac{dSe}{dh} \qquad (3)$$

In natural soils, the presence of aggregates frequently results in a retention function curve having at least two points of inflection. To represent such behaviour, a double porosity approach can be used which assumes that the pore space from θ_r to θ_s consists of two $f_i(h)$ distributions obtained by equation 3, each occupying a fraction ϕ_i of that pore space (Durner, 1994; Coppola, 2000).

Bimodal Durner model (Durner, 1994):

$$S_e=\sum\varphi_i\left[\frac{1}{1+(\alpha_{VG,i}h)^{n_i}}\right]^{m_i} \quad 0<\varphi_i<1 \text{ and } \sum\varphi_i=1 \quad i=1,2 \qquad (4)$$

in which ϕ_i is the weighting of the total pore space fraction to be attributed to the *i*th subcurve, and $\alpha_{VG,i}$, n_i and m_i still represent the fitting parameters for each of the partial curves.

Bimodal Ross & Smettem model (Ross and Smettem, 1993):

$$S_e = \varphi_1(1+\alpha_{RS}h)\,exp(-\alpha_1 h) + \varphi_2\left[\frac{1}{1+(\alpha_{VG}h)^{n_2}}\right]^{m_2}$$
$$0 < \varphi_i < 1 \quad \text{and} \quad \sum \varphi_i = 1 \quad i = 1{,}2 \tag{5}$$

where the following simple one-parameter function:

$$S_e = (1+\alpha_{RS}h)\,exp(-\alpha_{RS}h) \tag{6}$$

describes macroporosity in aggregated soils.

Unimodal Russo (Russo, 1988):

$$S_e = \frac{\theta-\theta_r}{\theta_s-\theta_r} = [\exp\left(-0.5\alpha_{GR}h\right)(1+0.5\alpha_{GR}h)]^{\left(\frac{2}{\mu_R+2}\right)} \tag{7}$$

where α_{GR} is the soil parameter appearing in the Gardner's model for hydraulic conductivity (see next section on hydraulic conductivity models) related to the pore size distribution, while μ_R is a parameter related to tortuosity. The Russo water retention model is generally coupled to the Gardner's model for hydraulic conductivity.

.

2.1.2. Hydraulic conductivity

Mualem's model (Mualem, 1986):

The model is based on the capillary bundle theory (Mualem, 1986) and relates relative hydraulic conductivity, K_r, to f(h) by the equation:

$$K_r(h) = \frac{K(h)}{K_0} = S_e^{\tau}[\eta(h)/\eta(0)]^2$$
$$\eta(h) = \int_{-\infty}^{\infty} h^{-1}f(h)dh \tag{8}$$

in which K_0 is the hydraulic conductivity at h=0, and τ is a parameter accounting for the dependence of the tortuosity and the correlation factors on the water content.
For the case of the van Genuchten unimodal soil water retention model and assuming m=1-1/n, equation 8 becomes:

$$K_r(S_e) = \frac{K(S_e)}{K_0} = S_e^{\tau}\left[1-\left(1-S_e{}^{1/m}\right)^m\right]^2 \tag{9}$$

For the bimodal case, equation 8 becomes:

$$K_r(h) = \frac{K(h)}{K_0} = S_e^{\tau}\left[\sum_{i=1}^{2} \varphi_i\, \eta_i(h) \Big/ \sum_{i=1}^{2} \varphi_i\, \eta_i(0)\right]^2 \tag{10}$$

with parameters as defined earlier.

In the case of bimodal Durner water retention, the equation 10 becomes (Priesack and Durner., 2006):

$$K_r(S_e) = \frac{K(S_e)}{K_0} = \left(\sum_{i=1}^{2} \varphi_i S_{e,i}\right)^{\tau} \left\{\frac{\sum_{i=1}^{2} \varphi_i \alpha_{VG,i}\left[1-\left(1-S_{e,i}^{1/m_i}\right)^{m_i}\right]}{\sum_{i=1}^{2} \varphi_i \alpha_{VG,i}}\right\}^2 \tag{11}$$

For the Ross & Smettem model, the η(h) functions in equation 8 are (Ross and Smettem, 1993):

$$\begin{aligned} \eta_1(h) &= \alpha_{RS}\, exp(\,\alpha_{RS} h) \\ \eta_2(h) &= \alpha_{VG} n B\left({S_e}^{1/m}, m + 1/n\,, 1 - 1/n\right) \end{aligned} \tag{12}$$

where $B(\ )$ is the incomplete beta function.

Equation 10 assumes that all pores in the material can interact [Ross and Smettem, 1993].

In this case, an instantaneous equilibrium (i.e. instantaneous exchange of water) between the two pore systems is implicitly assumed. Nevertheless, for some materials part of the pore space may bypass the rest. This is the case when for example the cutans on the walls of the aggregates reduces the water transport between intra and inter-aggregate spaces. The conductivities of the two porous spaces are in parallel and, for such independent distributions, the following equations apply:

$$\begin{aligned} k(h) &= \sum_{i=1}^{2} k_{0i}\, k_{ri}(h) \\ k_{ri}(h) &= S_e^{\tau}[\eta_i(h)/\eta_i(0)]^2 \end{aligned} \tag{13}$$

Gardner's model (Gardner, 1958):

The Russo model for water retention (equation 7) is generally combined to the Gardner's model for hydraulic conductivity

$$K_r(h) = \frac{K(h)}{K_0} = exp(-\alpha_{GR} h) \tag{14}$$

The parameter α_{GR} is a sorptive number related to the pore size distribution of the medium, with larger values associated with coarse-textured soils. It is also the reciprocal of capillary length parameter, λ_c, (α_{GR}= λ_c^{-1}), which can be interpreted as the length of the capillary fringe or the air-entry pressure head value.

2.2. The Sink Term, S_w

2.2.1. Root Water Uptake Sink Term, Sr

The root water sink term, S_r, in equation 1 is calculated in FLOWS according to a so-called macroscopic approach, frequently adopted in hydrologically oriented soil-plant-atmosphere continuum modeling for describing plant water uptake (Feddes et al., 1978; Feddes and Raats, 2004). It is essentially an empirical approach dealing with the root system and needs to be calibrated for different plants and climatic conditions.

The macroscopic sink term strictly depends on two aspects: i) the root density distribution and ii) the roots activity over the root zone during the growth season of a crop. Activity is used here in hydraulic terms, to express the physical dependence of root uptake on changes in the total hydraulic head of soil water, which in turn affect water fluxes to the roots, thus influencing water uptake. In the macroscopic sink term, reduced root activity is accounted for by introducing an uptake reduction function, $\alpha(h,h_{os})$, depending on the local water, h, and osmotic, h_{os}, potentials experienced by roots at any depths along the root-zone. Thus, the sink term is a function of both the soil water and osmotic potentials, $S_r(h,h_{os})$.

Calculating the Root Uptake Sink Term

In the macroscopic approaches to root uptake (Feddes *et al.*, 1978), the potential root water uptake flux per unit depth at a specific depth, $S_{r,p}$ [T^{-1}], is simulated by distributing potential transpiration, T_p [L T^{-1}], over the root zone depth, D_r [L], on the basis of a normalized root density distribution, *g(z)* [L^{-1}], with depth z.

The function *g(z)* distributes the potential transpiration rate, T_p, through the root zone in proportion to the root distribution (Feddes *et al.*, 1978; Feddes and Raats, 2004):

$$S_{r,p}(z) = g(z)T_p \tag{15}$$

with

$$\int_0^{Dr} g(z)\,dz = 1 \tag{16}$$

and thus

$$T_p = \int_0^{Dr} S_{r,p}(z)\,dz \tag{17}$$

Root Density Distribution

In general, the normalized root density distribution, *g(z)*, may be obtained by normalizing the root length density distribution R_{ld} [cm cm^{-3}] by its integral across the rooting depth, D_r:

$$g(z) = \frac{R_{ld}(z)}{\int_0^{D_r} R_{ld}(z)dz} \qquad (18)$$

Rld may be calculated as the ratio of the total length, *Lr*(*z*), of roots in a sample to the sample volume. *g(z)dz* is the fraction of roots located between *z* and *z*+*dz*.

Several root density distributions, *g(z),* may be selected in the model for simulating the sink term in equation 1, assuming root distributions to be either homogeneous (Feddes et al., 1978) or variable with depth (Raats, 1974; Prasad, 1988; Vrugt et al., 2001), the latter accounting for the fact that in a moist soil the roots can mainly extract water from the upper root zone layers. FLOWS consider several root density distribution functions.

A Uniform distribution over the whole root zone, which reduces simply to:

$$g(z) = \frac{1}{D_r} \qquad (19)$$

where D_r [L] is the maximum root depth.

A Prasad-type triangular distribution (Prasad, 1988), with root water uptake at the bottom of the root zone, D_r, equal to zero:

$$g(z) = \frac{2(D_r - z)}{D_r^2} \qquad (20)$$

A Vrugt-type distribution (Vrugt et al., 2001), which allows for calculating the dimensionless spatial root distribution, $\beta(z)$:

$$\omega(z) = \left(1 - \frac{z}{D_r}\right) exp\left(\frac{p_z}{D_r}|z^* - z|\right) \qquad (21)$$

where p_z [-] and z^* [L] are empirical parameters. The model allows for nonsymmetrical root water uptake. The non-symmetry with soil depth is determined by the ratio of the p_z value for $z \le z^*$ and the p_z value for $z > z^*$. To reduce the number of parameters to be optimized, p_z may be set to unity for $z > z^*$, whereas it is fitted for $z \le z^*$.

For the Vrugt's model the *g(z)* distribution is:

$$g(z) = \frac{\omega(z)}{\int_0^{D_r} \omega(z)dz} \qquad (22)$$

A Logistic distribution, which gives the cumulative root density distribution:

$$\int_0^{z=Dr} g(z)\,dz = \frac{1}{\left[1 + a\, exp\left[-b\left(\frac{z}{Dr} - c\right)\right]\right]^{\frac{1}{a}}} \tag{23}$$

so that

$$g(z) = \frac{bexp\left[b\left(c - \frac{z}{D_r}\right)\right]}{D_r\left\{1 + aexp\left[b\left(c - \frac{z}{D_r}\right)\right]\right\}^{\frac{1}{a}+1}} \tag{24}$$

where a, b and c are the coefficients of the logistic function and z is the depth.

2.1.2.3. Water and osmotic stress functions

Low water contents and/or the presence of soluble salts in the soil lower the total hydraulic head and may reduce the water fluxes to the roots, thus reducing root activity and water uptake. Reduction coefficients to decrease the maximum water uptake according to the water and osmotic stresses may be calculated independently and multiplied to calculate the actual root uptake, T_a, as:

$$S_r = \alpha_w(h)\alpha_s(h_{os})S_p = \alpha_w(h)\alpha_s(h_{os})g(z)T_p \tag{25}$$

with α_w and α_s being reduction factors depending on the local (at a given z) water pressure head, h [L] and osmotic head, h_{os} [L], respectively.

Accordingly,

$$T_a = \int_0^{Dr} S_r(z)\,dz \tag{26}$$

and

$$\frac{T_a}{T_p} = \int_0^{D_r} g(z)\,\alpha_w(h)\alpha_s(h_o)dz = \beta(h, h_{os}) \tag{27}$$

with T_a being the actual transpiration rate and β a dimensionless water stress index integrated over the whole rooted profile (Jarvis, 1989; Shouse et al., 2011), providing a measure of total plant stress. A value of β equal to 1 indicates that there is no stress in the soil root zone and that the actual transpiration rate T_a is equal to the potential transpiration rate T_p.

When the soil is irrigated by keeping soil water content under optimal conditions, the eventual reduction in root uptake may only be induced by osmotic stress. Under only osmotic stresses, α_w=1 and root uptake parameterization is reduced to finding the factor $\alpha_{s,}$ depending on the osmotic potential (h_{os}) induced by salts in the soil water.

Many of the functional forms that have been proposed for water and osmotic stress uptake reduction comes from the often-assumed relationship (De Wit, 1958; Doorenbos and Kassam, 1979):

$$\frac{T_a}{T_p} = \beta(h, h_{os}) = \frac{Y}{Y_p} \qquad (28)$$

where Y is the yield and Y_p is the potential yield (i.e., the maximum yield that would be obtained under optimal growing conditions).

Water stress reduction factor, α_w

Feddes and Raats (2004) reviewed various functional forms for α_w that have been proposed over the years. We mention here two general model types that have been used most often: piecewise linear functions and continuous smooth functions.

Feddes approach (Feddes, 1978):

To describe water stress, Feddes et al. (1978) proposed a piecewise linear reduction function parameterized by four critical values of the water pressure head, $h_4 < h_3 < h_2 < h_1$:

$$\alpha_w(h) = \begin{cases} \frac{h - h_4}{h_3 - h_4}, & \mathrm{h}_3 > h > h_4 \\ 1, & \mathrm{h}_2 \geq h \geq h_3 \\ \frac{h - h_1}{h_2 - h_1}, & \mathrm{h}_1 > h > h_2 \\ 0, & \mathrm{h} \leq h_4 \; or \;\; h \geq h_1 \end{cases} \qquad (29)$$

The figure 2a shows a plot of equation 29. In this model, water uptake is reduced at high and low water contents. Uptake is at the potential rate when the pressure head is $h_3 \leq h \leq h_2$, drops off linearly when $h>h_2$ or h<h_3, and becomes zero when $h \leq h_4$ (*permanent wilting* pressure head range) or $h \geq h_1$(*anaerobiosis* pressure head range). In general, the value of h_3 is expected to be a function of evaporative demand, with h_{3low} and h_{3high} in figure 1a referring to either a low or high evaporative demand.

Van Genuchten approach (van Genuchten, 1987):

Van Genuchten (1987) proposed an alternative smooth, S-shaped reduction function to account for water stress:

$$\alpha_w(h) = \frac{1}{1 + \left(\frac{h}{h_{50}}\right)^{p_1}} \tag{30}$$

where h_{50} and p_1 are adjustable parameters, the former being the water pressure head where uptake is halved. The figure 2b shows a plot of equation 30.

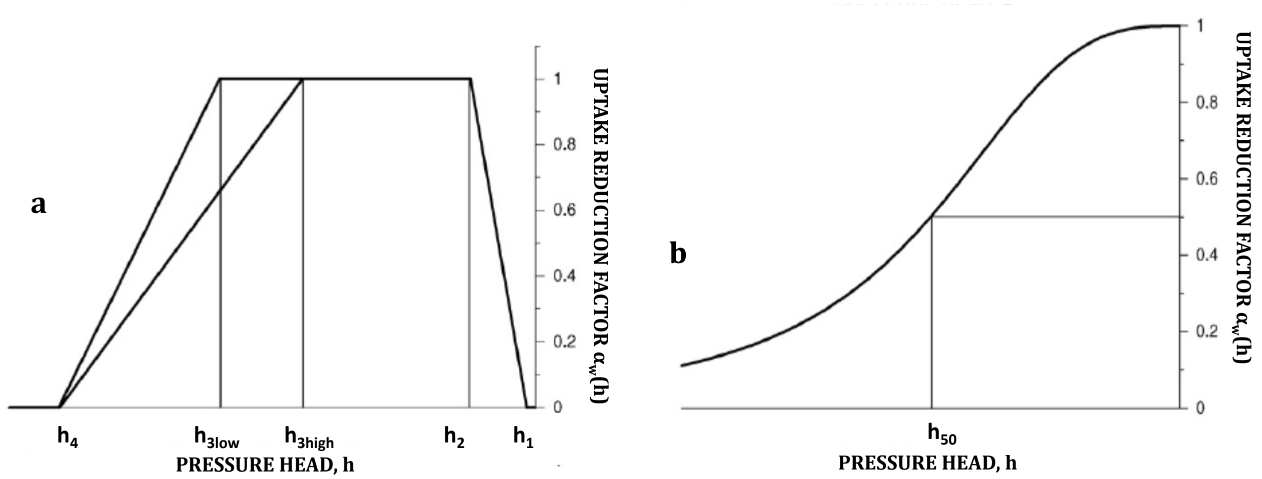


Figure 2: Plots of (a) the Feddes et al. (1978) water stress uptake reduction function (equation 29), and (b) the S-shaped function of van Genuchten (1987) (equation 30)

Salinity stress reduction factor, α_s

Despite the large body of studies on the topic (see amongst others Molz and Remson, 1970; van Genuchten and Hoffman, 1984; Feddes and Raats, 2004; Skaggs *et al.*, 2006; Homaee *et al.*, 2002a,b,c), proper modelling and parameterisation of root water uptake as a function of salinity stresses remains a major challenge. The process is mainly determined by the specific soil-water-salt conditions and distributions which become established in the root zone according to the local soil physical-hydrological characteristics. This is why the well-known salt tolerance database presented by the U.S. Salinity Laboratory (Maas and Hoffman, 1977) should be handled with prudence in numerical modelling as most of those data were collected under well-controlled experimental conditions in which uniform salt distribution over the root zone was established. As confirmed by several researchers (Homaee *et al.*, 2002a,b,c; Skaggs *et al.*, 2006; Shouse *et al.*, 2011), the reduction function to be used in numerical modelling should be determined by analysing the dynamics of water uptake under transient salinity conditions.

Several functional forms have been proposed for uptake reduction due to salinity (see amongst others Feddes and Raats, 2004; van Genuchten and Hoffman, 1984). The response function can be written in terms of concentration, or electrical conductivity of either the soil water or the soil saturation extract, or osmotic pressure head (Maas, 1986; Maas, 1990; Maas et al., 1999; Maas and Hoffman 1977; van Genuchten *et al.*, 1984).

Mass and Hoffman approach (Mass and Hoffman, 1977):
The effects of salinity stress on root water uptake can be described using the piecewise linear (threshold–slope) function (figure 3):

$$\alpha_s(h_{os}) = \begin{cases} 1, & \mathrm{a} \le h_{os} \le 0 \\ 1 + b(h_{os}\text{-a}), & \mathrm{a} > h_{os} > \mathrm{a}\text{-}\dfrac{1}{b} \\ 0, & h_{os} \le \mathrm{a}\text{-}\dfrac{1}{b} \end{cases} \tag{31}$$

where a and b are the adjustable parameters, often referred to as the salinity threshold and slope, respectively. The approach mirrors the Maas and Hoffman (1977) model for reduction of yield due to salt stresses

$$\frac{Y}{Y_p} = \begin{cases} 1 & \mathrm{EC}_e \le A \\ 1 - B(\mathrm{EC}_e - A). & A < \mathrm{Ec}_e \le \dfrac{100}{B} + A \\ 0, & \mathrm{Ec}_e > \dfrac{100}{B} + A \end{cases} \tag{32}$$

with EC_e being the electrical conductivity of the soil saturation extract (dSm^{-1}). Note, however, that the parameter sets in equations 31 and 32 are not the same: A and B parameterize total yield reductions as a function of average root zone salinity, whereas a and b parameterize local reductions in the root water uptake rate as a function of osmotic head.

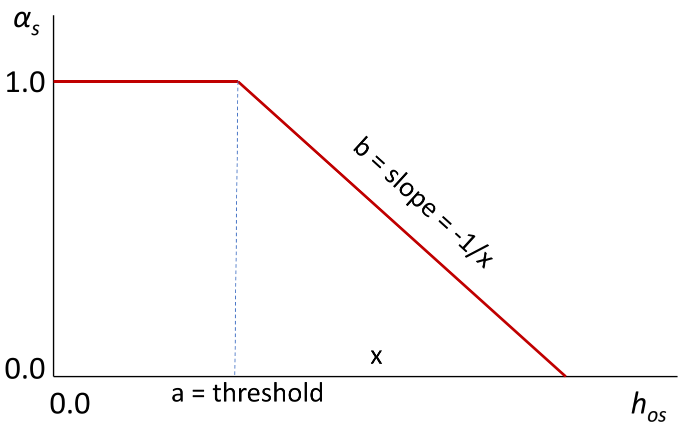


Figure 3. Graph of the Mass and Hoffman type salinity stress uptake reduction function (equation 31)

van Genuchten and Hoffman approach (van Genuchten and Hoffman, 1984):
van Genuchten and Hoffman (1984) proposed an S-shaped salinity-stress function in terms of osmotic pressure heads, similar to the approach for the water stress reduction factor:

$$\alpha_s(h_{os}) = \frac{1}{1 + \left(\frac{h_{os}}{h_{os,50}}\right)^{p_2}} \tag{33}$$

where p_2 and $h_{os,50}$ are the adjustable parameters, the latter being the osmotic pressure head where uptake is halved.
The osmotic potential, expressed as osmotic head h_{os}, is assumed to be a linear function of soil solution salinity, σ_w (dSm^{-1}), according to *U. S. Salinity Laboratory Staff* (USSL Staff, 1954)*:*

$$h_{os} = -360\sigma_w \tag{34}$$

360 is a factor to convert the salinity-based values (dSm^{-1}) to osmotic head (cm).

Combined water and salinity stress
A major challenge is how to combine the effects of water and salinity stress. Uptake reductions due to a combination of water and salinity stresses could be modeled by assuming that the stresses are somehow additive or multiplicative. A generalized model for additive stresses is obtained by assuming that water uptake occurs in response to some weighted sum of the soil water pressure and osmotic heads:

$$\alpha(h, h_{os}) = \frac{1}{1 + \left[\frac{\alpha_1 h + \alpha_2 h_{os}}{h_{os,50}}\right]^{p_2}} \tag{35}$$

with simple additivity resulting when $\alpha_1 = \alpha_2 = 1$. Although additivity has been inferred from a number of laboratory and field experiments (e.g., Wadleigh, 1946; Meiri and Shalhevet, 1973; Childs and Hanks, 1975; du Plessis, 1985; Bresler and Hoffman, 1986), its general applicability remains uncertain (Shalhevet and Hsiao, 1986), especially for field conditions subject to relatively wide ranges in pressure heads (wetting/ drying cycles).

Linear or weighted additivity is only one of several possibilities for combining the effects of water and salinity stress. Another approach presumes that water and salinity effects are multiplicative. In the general case this leads to:

$$\alpha(h, h_{os}) = \alpha(h)\alpha(h_{os}) \tag{36}$$

which, in the case of S-shaped curves, gives:

$$\alpha(h, h_{os}) = \frac{1}{1 + \left(\frac{h}{h_{50}}\right)^{p_1}} \frac{1}{1 + \left(\frac{h_{os}}{h_{os,50}}\right)^{p_2}} \tag{37}$$

2.2.2. Drainage Sink Term, Sdr

The drainage is included in the Richards equation as a sink term based on the Hooghoudt theory for lateral flow to the drains (see figure 4). It has been demonstrated that a 1-D solution to the flow and transport system, combined with Hooghoudt theory, gives similar results as a 2-D solution with explicit representation of drain tiles based on the previous studies, provided that an appropriate implementation of the Hooghoudt theory is used. This issue has been analysed in details by Mollerup et al (2012).

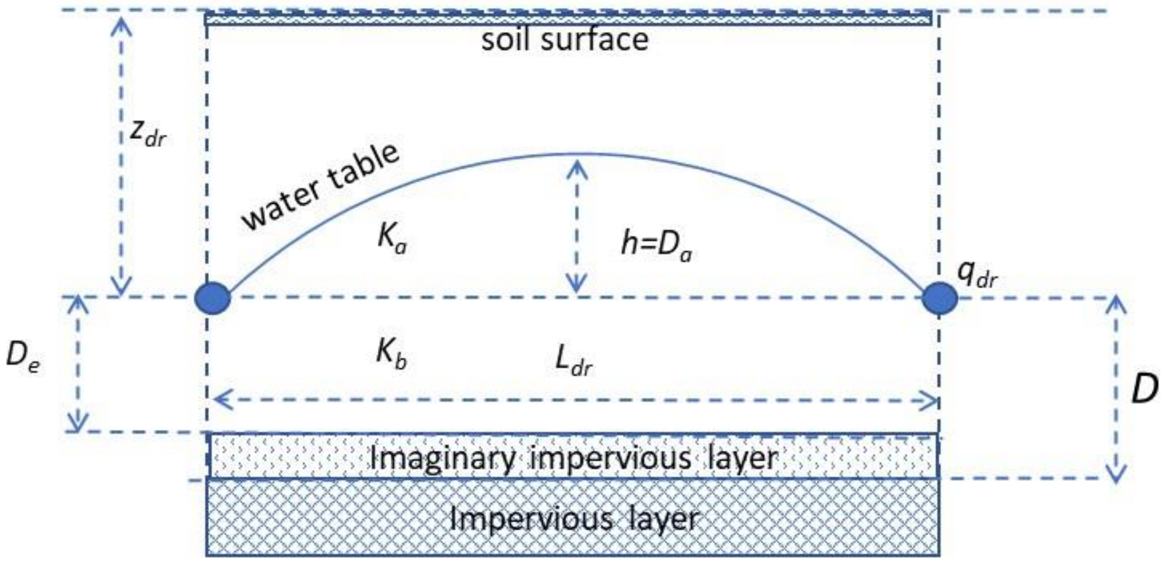


Figure 4. Schematic view of the system and parameters involved in the Hooghoudt's equation

Water will flow toward the drain anytime the water table is located above either a drainpipe or a on open field drain. According to Hooghoudt (1940), the steady state drain flux per unit surface area q_{dr} can be computed as:

$$q_{dr} = \frac{8K_b D_{eq} D_a + 4K_a D_a^2}{L_{dr}^2} \tag{38}$$

where K_a is the hydraulic conductivity of the saturated layer above the drain level and K_b is the hydraulic conductivity of the layer below the drain level. L_{dr} is the distance between the drains. The water table height above the drain at $0.5L_{dr}$ is denoted D_a (corresponding to h in the Hooghoudt work).

D_{eq} is the equivalent drain depth depending on the vertical distance between the drains and the impervious layer, D_{imp}, as well as on the drain hydraulic radius, R_{dr}.

D_{eq} is introduced instead of D_{imp} if the drain pipe or the open drains bottom do not reach the impervious layer. Actually, the Hooghoudt theory for lateral flow to drains assumes horizontal flow lines (according to the Dupuit-Forchheimer theory). In the case of drains above the impervious layer, the flow lines will converge towards the drain (radial flow lines) and will thus no longer be horizontal.

Consequently, the flow lines are longer and extra head loss is required to have the same volume of water flowing into the drains, which results in a higher watertable. To be able to use the concept of horizontal flow, Hooghoudt (1940) assumed an imaginary impervious layer above the real one, which decreases the thickness of the layer through which the water flows towards the drains;

van der Molen and Wesseling (1991) proposed an analytical solution to calculate D_{eq}:

$$\begin{gathered} D_{eq} = \frac{1}{8}\frac{\pi L_{dr}}{ln\left(\frac{L_{dr}}{\pi R_{dr}}\right) + F(x)} \\ x = \frac{2\pi D_{imp}}{L_{dr}} \\ F(x) = \frac{\pi^2}{4x} + ln\left(\frac{x}{2\pi}\right) \qquad \text{for } x < 0.5 \\ F(x) = \sum_{i=1}^{\infty} \frac{4e^{(-4i+2)x}}{(2i-1)(1-e^{(-4i+2)x})} \qquad \text{for } x \geq 0.5 \end{gathered} \tag{39}$$

The $F(x)$ for $x < 0.5$ converges rapidly for $x > 1$.

According to the solution technique proposed by Mollerup et al (2012), the FLOWS model calculates S_{dr} by firstly dividing q_{dr} in the two components coming from above the drain level, q_a, and below the drain level, q_b, respectively:

$$q_a = \frac{4K_a D_a^2}{L_{dr}^2}$$
$$q_b = \frac{8K_b D_{eq} D_a}{L_{dr}^2} \tag{40}$$

Following the Daisy model approach (Hansen et al., 2012), in FLOWS K_a and K_b are compartments-weighted averages calculated by considering respectively the contributions of all the saturated nodes above and below the drain level:

$$K_a = \frac{\sum_i \xi \Delta z_i K_{0,i}}{D_a}, \qquad D_a = \sum_i \xi \Delta z_i \quad \text{for all the nodes } i \textbf{ above} \text{ the drain level}$$
$$K_b = \frac{\sum_i \xi \Delta z_i K_{0,i}}{D_b}, \qquad D_b = \sum_i \xi \Delta z_i \quad \text{for all the nodes } i \textbf{ below} \text{ the drain level} \tag{41}$$

with $\xi = 1$ in the case of saturated nodes and $\xi = 0$ in the case of unsaturated nodes (thus not contributing to the flow to drains) and Δz_i the thickness of the simulation node compartments. The weighted-average allows to account for possibility that the layer above the drain (and that below the drain) may consist of nodes with different saturated hydraulic conductivities. With $K_{0,i}$ equal for all the nodes in the layer above the drain (homogeneous layer), K_a would be equal to $K_{0,i}$. The same for the layer below the drain level.

Finally, the S_{dr} in each of the N simulation nodes is calculated by distributing the q_a and q_b fluxes over the saturated nodes below and above the drain level:

$$S_{dr,a} = \frac{\xi K_{0,i}}{K_a D_a} q_a \quad \text{for all the nodes } i \textbf{ above} \text{ the drain level}$$
$$S_{dr,b} = \frac{\xi K_{0,i}}{K_b D_b} q_b \quad \text{for all the nodes } i \textbf{ below} \text{ the drain level} \tag{42}$$
$$S_{dr} = S_{dr,a} + S_{dr,b}$$

The code requires as input for drainage fluxes calculation the distance of the drain from the soil surface, z_{dr}, the distance of the impermeable layer from the soil surface, z_{imp}, the drain interaxis, *L*L_{dr}, the hydraulic radius of the drain, R_{dr}. Da is calculated by the model according to the simulations nodes actually saturated from the impermeable layer upward. q_{dr} and thus S_{dr} are calculated only if D_a>0.

2.3. Numerical solution of the Richards' equation

There exist analytical solutions to equation 1 only in particular cases and in practical problems the Richards' equation has to be solved numerically. In the standard finite difference scheme used by Coppola and Randazzo (2006) and Coppola et al., (2009; 2012; 2015; 2019), the general form of the Richards' equation is expressed as an implicit pressure based scheme, to be solved for each of the *nz* (in this code 100) discretization nodes in which the flow domain has been divided, numbered starting from the top boundary of the simulation domain:

$$C_i^{j+1/2}\frac{h_i^{j+1}-h_i^j}{\Delta t} = \frac{1}{\Delta z_i}\left\{K_{i-1/2}^j\left[\frac{h_{i-1}^{j+1}-h_i^{j+1}}{\Delta z_i}+1\right]-K_{i+1/2}^j\left[\frac{h_i^{j+1}-h_{i+1}^{j+1}}{\Delta z_i}+1\right]\right\}-S_w^j \qquad (43)$$

where $C_i^{j+1/2}$is the derivative approximation to the differential water capacity Δt is the time step (*j* is the time level) of calculation, Δz_i is the thickness of the i^{th} node, $K_{i-1/2}^j$and$K_{i+1/2}^j$are the internodal hydraulic conductivities, S_w^j is the water sink rate (water uptake by plant roots, S_r, and/or lateral water drainage, S_{DR}, so that S_w=S_r+S_{DR}). In the case of layered soils, Δz_i is calculated in each layer in a way that a node is centred at the layers interface.

K and S_w are calculated using an explicit linearization, i.e. using *h*-values from time level *j*. The approximation with respect to spatial derivative is taken as a fully implicit one leading to a convenient form of solution where the $K(h)$ values need to be calculated only once during each time step, which significantly reduces the computational burden. The same type of approximation has been used in the SWAP model (van Dam, 1997) and by Karvoneen (1988) and Karvoneen et al. (1999). The internodal hydraulic conductivities used in equations 43 are calculated as the arithmetic average between the nodes (van Dam et al. 1997):

$$\begin{aligned} K_{i-1/2}^j &= \frac{K(h_{i-1}^j)+K(h_i^j)}{2} \\ K_{i+1/2}^j &= \frac{K(h_{i+1}^j)+K(h_i^j)}{2} \end{aligned} \qquad (44)$$

The linear system of *nz* equations obtained by writing equations 43 from *i*=2 to *i*=*nz*-1 with the appropriate boundary conditions, produces a tridiagonal matrix with the following coefficients:

$$\begin{aligned}
&\alpha_i h_{i-1}^{j+1} + \eta_i h_i^{j+1} + \gamma_i h_{i+1}^{j+1} = \delta_i \quad , i = 2,..,nz-1 \\
&\alpha_i = -\frac{\Delta t K_{i-1/2}^{j}}{\Delta {z_i}^2} \\
&\gamma_i = -\frac{\Delta t K_{i+1/2}^{j}}{\Delta {z_i}^2} \\
&\eta_i = C_i^{j+1/2} - \alpha_i - \gamma_i \\
&\delta_i = C_i^{j+1/2} h_i^j + \frac{\Delta t}{\Delta z_i}\left(K_{i-1/2}^j - K_{i+1/2}^j\right) - \Delta t\left[S_r^j + S_{DR}^j\right]
\end{aligned} \qquad (45)$$

which is solved by the Thomas' algorithm.

Flow rates and pressure heads, whether constant or variable over time, can be assumed as the upper boundary condition. Gradients of different value, pressure heads or flow rates, again whether constant or variable, can be assumed at the bottom of the soil profile. For the first and last nodes (*i*=1 and *i*=*nz*), respectively, the coefficients shown in equation 45 for the intermediate nodes will change according to the boundary conditions adopted.

Flux condition (Neumann condition) at the upper boundary

In this case, the term $K_{i-{}^1/_2}^j\left[\frac{h_{i-1}^{j+1}-h_i^{j+1}}{\Delta z_i}+1\right] = -q_{surf}$, with q_{surf} being the upper flow rate, so that the equation and the coefficients of the tridiagonal matrix for the top node become:

$$\begin{aligned}
&\eta_i h_i^{j+1} + \gamma_i h_{i+1}^{j+1} = \delta_i \quad i = 1 \\
&\gamma_i = -\frac{\Delta t K_{i+1/2}^{j}}{\Delta z_{top}\Delta z_i} \\
&\eta_i = C_i^{j+1/2} - \gamma_i \\
&\delta_i = C_i^{j+1/2} h_i^j + \frac{\Delta t}{\Delta z_{top}}\left(-q_{surf} - K_{i+1/2}^j\right) - \Delta t\left[S_r^j + S_{DR}^j\right]
\end{aligned} \qquad (46)$$

with Δz_{top} the thickness of the top compartment (in the case of equal Δz_i for all the nodes, $\Delta z_{top} = 0.5\Delta z_i$)

Head condition (Dirichlet condition) at the upper boundary

In this case, the term $K_{i-1/2}^{j}\left[\frac{h_{i-1}^{j+1}-h_{i}^{j+1}}{\Delta z_i}+1\right]=K_{i-1/2}^{j}\left[\frac{h_{surf}-h_{i}^{j+1}}{\Delta z_i}+1\right]$, with h_{surf} being the pressure head applied at the top boundary, so that the equation and the coefficients of the tridiagonal matrix for the top node become:

$$\begin{aligned}
&\eta_i h_i^{j+1}+\gamma_i h_{i+1}^{j+1}=\delta_i \quad i=1\\
&\gamma_i=-\frac{\Delta t K_{i+1/2}^{j}}{\Delta z_{top}\Delta z_i}\\
&\eta_i=C_i^{j+1/2}+\frac{\Delta t K_{i-1/2}^{j}}{\Delta z_{top}\Delta z_i}-\gamma_i\\
&\delta_i=C_i^{j+1/2}h_i^j+\frac{\Delta t}{\Delta z_i}\left(K_{i-1/2}^{j}-K_{i+1/2}^{j}\right)+\frac{\Delta t K_{i-1/2}^{j}}{\Delta z_{top}\Delta z_i}h_{surf}-\Delta t\left[S_r^j+S_{DR}^j\right]
\end{aligned} \tag{47}$$

Flux condition (Neumann condition) at the bottom boundary

In this case, the term $K_{i+1/2}^{j}\left[\frac{h_{i}^{j+1}-h_{i+1}^{j+1}}{\Delta z_i}+1\right]=-q_{bot}$, with q_{bot} being the bottom flow rate, so that the equation and the coefficients of the tridiagonal matrix for the bottom node become:

$$\begin{aligned}
&\alpha_i h_{i-1}^{j+1}+\eta_i h_i^{j+1}=\delta_i \quad i=nz\\
&\alpha_i=-\frac{\Delta t K_{i-1/2}^{j}}{\Delta {z_i}^2}\\
&\eta_i=C_i^{j+1/2}-\alpha_i\\
&\delta_i=C_i^{j+1/2}h_i^j+\frac{\Delta t}{\Delta z_i}\left(K_{i-1/2}^{j}+q_{bot}\right)-\Delta t\left[S_r^j+S_{DR}^j\right]
\end{aligned} \tag{48}$$

Head condition (Dirichlet condition) at the bottom boundary

In this case, the term $K_{i+1/2}^{j}\left[\frac{h_{i}^{j+1}-h_{i+1}^{j+1}}{\Delta z_i}+1\right]=K_{i+1/2}^{j}\left[\frac{h_{i}^{j+1}-h_{bot}}{\Delta z_i}+1\right]$, with h_{bot} being the pressure head applied at the bottom boundary, so that the equation and the coefficients of the tridiagonal matrix for the top node become:

$$\alpha_i h_{i-1}^{j+1}+\eta_i h_i^{j+1}=\delta_i \quad i=nz \tag{49}$$

$$\eta_i = C_i^{j+1/2} - \alpha_i + \frac{\Delta t K_{i+1/2}^{j}}{\Delta z_{bot}\Delta z_i}$$

$$\delta_i = C_i^{j+1/2} h_i^j + \frac{\Delta t}{\Delta z_i}\left(K_{i-1/2}^{j} - K_{i+1/2}^{j}\right) + \frac{\Delta t K_{i+1/2}^{j}}{\Delta z_{bot}\Delta z_i} h_{bot} - \Delta t\left[S_r^j + S_{DR}^j\right]$$

with Δz_{bot} the thickness of the bottom compartment (in the case of equal Δz_i for all the nodes, $\Delta z_{bot} = 0.5\Delta z_i$).

Gradient condition at the bottom boundary

In the case of a fixed gradient (*grad*) imposed at the bottom boundary (any values may be set for the gradient, even if the most used is *grad*=1 to impose a so-called *free drainage*). In the general case, the value of h_{bot} to be used as head bottom boundary condition is obtained as:

$$h_{bot} = h_{nz} - (grad - 1) * \Delta z_{bot} \qquad (50)$$

Seepage face condition at the bottom boundary

This is a hybrid condition allowing to simulate the bottom boundary condition in the case the bottom of the simulation domain is open to the atmosphere (laboratory columns, for example). In this case, no downward bottom flux will be observed until the bottom becomes saturated. During the simulation run, the model verifies if the pressure head at the bottom is lower or higher than zero. In the first case, the model sets a zero-flux boundary condition:

$$q_{bot} = 0 \qquad (51)$$

Once the pressure head becomes zero or positive, the model switches to a zero-head boundary condition.

$$h_{bot} = 0 \qquad (52)$$

Iteration scheme

In solving the Richards' equation, the differential water capacity $C_i^{j+1/2}$ has to be evaluated.

The approximation of $C_i^{j+1/2}$ used in our numerical code is known as the Standard Chord Slope (Rathfelder and Abriola, 1994; Huang et al., 1996a):

$$C_i^{j+1/2} = \frac{\theta_i^{j+1} - \theta_i^j}{h_i^{j+1} - h_i^j} \qquad (51)$$

which requires iterations. The iterative process continues until the difference of the calculated pressure heads between two successive iteration levels in each node becomes less than a predefined tolerance ε:

$$\left|h_i^{j+1,m} - h_i^{j+1,m-1}\right| \leq \varepsilon \tag{52}$$

with *m* being the iterative level; a value of ε=0.001 cm is adopted by default in the code.

Thus, for a given time step, the linear system has to be solved as many times as the iterative process requires for converging.

The iteration number required for convergence, $M=m_{conv}$, is also used for setting an automatic variable simulation time step according to the following criteria:

$$\begin{array}{ll} M \leq 3 & \Delta t^{j+1} = 1.3\,\Delta t^{j} \\ 3 < M < \mathrm{M}_{max} & \Delta t^{j+1} = \Delta t^{j} \\ M \geq \mathrm{M}_{max} & \Delta t^{j+1} = 0.7\Delta t^{j} \end{array} \tag{53}$$

having set the maximum iteration number M_{max}=10. Under the third condition in equation 53, the time step Δt is adjusted and the iteration restarted again for the same time step.

2.4. Top boundary conditions for water flow

Flux top boundary conditions: In the case of simulations carried out under real atmospheric fluxes, the upper boundary condition for vadose zone depends on climatic conditions. Either constant (over time) or variable fluxes may be imposed at the top boundary.

Rainfall: In the code, rainfall represents negative q_{surf}. The model allows the generation of infiltration excess runoff either if rainfall intensity exceeds the soil surface infiltration capacity (Hortonian mechanism of runoff) or in the case of profile complete saturation (Dunnian mechanism of runoff). The first condition (Hortonian) is formulated using a maximum infiltration velocity at soil surface, fs_{max} (*fluxsurf_max* in the code)

$$fs_{max} = K_{m,max}\left(\frac{h_1 - h_{surf,max}}{\Delta z_{top}} - 1\right) \tag{54}$$

where $K_{m,max}$ is the arithmetic average between the surface and top node hydraulic conductivities, h_1 is the pressure head at the top node, $h_{surf,max}$ is the maximum pressure head allowed at the surface (generally $h_{surf,max}$ =0).

In the case *of* $q_{surf} > fs_{max}$, the code switches the upper boundary condition to an head boundary condition by imposing $h_{top} = h_{surf,max}$ at the top boundary. If the condition abs(q_{surf}) > K_0 for the first node is also verified, the difference $q_{surf} - fs_{max}$ becomes a runoff flux.

As for the Dunnian mechanism of runoff, at each time step Δt the model calculates the storage at whole saturation and the actual storage along the whole simulation domain. If the storage difference is lower than a predefined tolerance, $\Delta\theta_s - \Delta\theta \leq \varepsilon$, all the incoming flux, q_{surf}, during the current Δt will become runoff.

Evapotranspiration: In the code, evapotranspiration produces positive q_{surf}. Potential evapotranspiration ET_p is calculated using the reference evapotranspiration, ET_r, method. In this case, the ET_p is calculated as $ET_p=kc\times ET_r$, kc being a crop factor, which depends on the crop type, the growth stage, and the method employed to obtain ET_r.

In both cases, ET_p is partitioned into T_p and E_p on the basis of Beer's law (Ritchie, 1972):

$$\begin{aligned} E_p &= ET_p \times e^{-k_{ex}\times LAI} \\ T_p &= ET_p - E_p \end{aligned} \tag{55}$$

where k_{ex} is an extinction coefficient, frequently set to be 0.5, and LAI is leaf area index.

In the case of bare soil, the crop resistance equals to zero and $E_p=ET_p$. By using the ET_r method, the variable kc is substituted by a single valued soil factor k_{soil} and $E_p=k_{soil}\times ET_r$.

The code distributes potential transpiration T_p over the root zone on the basis of the root density distribution and reduces T_p to actual transpiration, T_a, on the basis of soil matric potential and electrical conductivity of the soil solution (see the section *Calculating the Root Uptake Sink Term*)

The code requires ET_r, LAI and kc, as well as the extinction factor, k_{ex}, as input over the simulation period.

In the case of evapotranspiration surface boundary condition, the code calculates the maximum evaporation, E_{max}, allowed given the difference of atmospheric pressure, h_{atm} (in cm of water column) and the pressure head in the first simulation node, h_1, as:

$$E_{max} = -K_{m,max}\left(\frac{h_{atm} - h_1}{\Delta z_{top}} - 1\right) \tag{56}$$

where $K_{m,max}$is again the arithmetic average between the surface and top node hydraulic conductivities

In the case of $E_p < E_{max}$, $q_{surf}=E_p$ else, $q_{surf} = E_{max}$.

In the case of irrigation, negative irrigation fluxes, q_{irr}, are added to the atmospheric fluxes, q_{surf}. The irrigation fluxes may come either from irrigation volumes actually supplied by a farmer or irrigation volumes calculated by the model according to a criteria based on the average pressure head in the root depth, D_r (see the section below on the irrigation criteria adopted by the code).

Pressure head top boundary conditions: In this case, either constant or variable pressure heads may be imposed at the top boundary. It is especially important to simulate, for example, a water ponding imposed at the soil surface or any infiltration experiment carried out under controlled surface pressure heads. As already discussed above, such a condition also comes into play in the case of initial flux boundary condition, when $q_{surf} > fs_{max}$, and the upper boundary condition switches from a flux to a to a pressure head boundary condition, with $h_{top} = h_{surf,max}$.

2.5. Bottom boundary conditions for water flow

Flux bottom boundary conditions: This condition (constant or variable over time) allows for water outflow (downward, negative q_{bot}) or inflow (upward, positive q_{bot}) through the bottom boundary of the soil profile. A special case is when q_{bot}=0, so that the water cannot leave nor enter the soil profile. This may happen for example because of an impeding soil layer at the bottom of the simulation domain.

Pressure head bottom boundary conditions: Again, either constant or variable pressure heads may be imposed at the top boundary. It is especially important to simulate, for example, any infiltration experiment carried out under controlled bottom pressure heads. The special case of h_{bot}=0 may be used, for example, to simulate the presence of a constant water table at the bottom boundary of the simulation domain. Similarly, the case of variable h_{bot} may be used to simulate, for example, a fluctuating free water surface at the bottom of the soil profile.

3. Solute transport in soils

The advection–dispersion equation (ADE) is used for prediction of solute transport (agrochemicals, salts, heavy metals, …):

$$\frac{\partial \theta C}{\partial t}+\rho_b\frac{\partial C_s}{\partial t}+\frac{\partial \theta_g C_g}{\partial t}=-\frac{\partial qC}{\partial z}+\frac{\partial}{\partial z}\left(\theta D_h\frac{\partial C}{\partial z}\right)+\frac{\partial}{\partial z}\left(\theta_g D_g^s K_H\frac{\partial C}{\partial z}\right)-S_s \qquad (57)$$

including linear and non-linear adsorption, linear volatilization, linear decay and proportional root uptake in unsaturated/saturated soil. In the equation, C [M L^{-3}], C_s [M M^{-1}] and C_g [M L^{-3}], are the amount of solute in the liquid, adsorbed and gaseous phases, respectively, q [L T^{-1}] is the darcian water flux, ρ_b [M L^{-3}] is the bulk density, D_h [L^2 T^{-1}] the hydrodynamic dispersion coefficient, D_g^s is the dispersion coefficient in the gaseous phases [L^2 T^{-1}], θ_g is the volumetric air content in soil, S_s [ML^{-3}T^{-1}] is a source-sink term for solutes, K_H is the dimensionless Henry constant. Hydrodynamic dispersion is related to the molecular diffusion constant of the substance in bulk water, D_0 [L^2 T^{-1}], and the average pore water velocity, $v=q/\theta$, as:

$$D_h=\lambda v+\eta(\theta)D_0 \qquad (58)$$

where λ [L] is the dispersivity and η a tortuosity coefficient. However, as discussed by several researchers (Comegna et al., 1999; Vanderborght and Vereecken, 2007; among others) the contribution of diffusion to the hydrodynamic dispersion is often very small.

By introducing an effective dispersion coefficient, D_e,

$$D_e=D_h+\frac{\theta_g D_g^s K_H}{\theta} \qquad (59)$$

the equation 57 becomes

$$\frac{\partial \theta C}{\partial t}+\rho_b\frac{\partial C_s}{\partial t}+\frac{\partial \theta_g C_g}{\partial t}=-\frac{\partial qC}{\partial z}+\frac{\partial}{\partial z}\left(\theta D_e\frac{\partial C}{\partial z}\right)-S_s \qquad (60)$$

3.1. Linear and Non-Linear solute sorption

The model assumes equilibrium interaction between the solution, C, and adsorbed, C_s, concentrations of solute on soil particles. The adsorption isotherm relating C_s and C is described in the code by either a *linear* or the *Freundlich* non-linear equation. The *Freundlich* isotherm is a flexible function for many organic and inorganic solutes:

$$C_s = K_F C^p \qquad (61)$$

where p (dimensionless) is the Freundlich exponent, commonly less than unity for most adsorbing solutes and enhanced with decreasing C, and K_F [L^3M^{-1}] is the Freundlich partitioning coefficient (Selim and Amacher, 1997). The *linear* isotherm is a special case where the Freundlich exponent is unity and $K_F = K_D$ [$L^{3p}M^{-p}$] (the slope of the linear isotherm) is known as the distribution coefficient.

3.2. Linear solute volatilization

The gaseous phase is also considered in the model to account for volatilization and gas phase diffusion of organic contaminants. Actually, even if the latter may decay mostly because of chemical and microbiological degradation, volatilization may be equally important for some substances such as pesticides. A linear equilibrium volatilization is assumed, so that gaseous, C_g, and liquid, C, concentrations are linearly related through the Henry constant, K_H:

$$C_g = K_H C \qquad (62)$$

3.3. Solute source-sink term

The solute source-sink term comes from the contribution of solute decay, $S_{s,\mu}$, solute root uptake, $S_{s,r}$ and solute losses coming from drainage fluxes, $S_{s,dr}$:

$$\begin{gathered} S_{s,\mu} = \mu(\theta C + \rho_b C_s) \\ S_{s,r} = K_{ru} S_w C \\ S_{s,dr} = S_{dr} C \\ S_s = S_{s,\mu} + S_{s,r} + S_{s,dr} \end{gathered} \qquad (63)$$

where μ is a first order decay rate coefficient of transformation (T^{-1}), accounting for linear decomposition, and K_{ru} [-] is a factor for solute root uptake, accounting for positive or negative selection of solute ions relative to the amount of root water uptake S_W.

3.4. Numerical solution of the Advection-Dispersion equation

The code solves the ADE equation by an explicit, central difference scheme:

$$
\begin{aligned}
&\frac{\theta_i^{j+1}C_i^{j+1}+\rho_b C_{s,i}^{j+1}+\theta_{g,i}^{j+1}C_{g,i}^{j+1}-\theta_i^{j}C_i^{j}-\rho_b C_{s,i}^{j}-\theta_{g,i}^{j}C_{g,i}^{j}}{\Delta t}\\
&\qquad=\frac{q_{i-1/2}^{j}C_{i-1/2}^{j}-q_{i+1/2}^{j}C_{i+1/2}^{j}}{\Delta z}\\
&\qquad+\frac{1}{\Delta z}\left[\frac{\theta_{i-1/2}^{j}D_{e,i-1/2}^{j}\left(C_{i-1}^{j}-C_i^{j}\right)}{\Delta z}-\frac{\theta_{i+1/2}^{j}D_{e,i+1/2}^{j}\left(C_i^{j}-C_{i+1}^{j}\right)}{\Delta z}\right]\\
&\qquad-\mu_i^{j}\left(\theta_i^{j}C_i^{j}+\rho_b C_{s,i}^{j}\right)-K_r S_i^{j}C_i^{j}
\end{aligned}
\tag{64}
$$

As discussed by van Dam et al. (1997), compared to an implicit iterative scheme, this explicit scheme allows for a relatively easy inclusion of non-linear adsorption and other non-linear processes. In the code, the following criterium:

$$
\Delta t^{j} \leq \frac{\Delta z^2\theta^2}{2\lambda|q|} \tag{65}
$$

allows for the stability of this explicit solution scheme (van Genuchten and Wierenga, 1974). It applies for tracers and is thus safe for adsorbed solutes.

3.5. Boundary conditions for solute transport

The equation 64 has to be completed with auxiliary conditions describing the boundary conditions and the initial concentration in the soil profile.

As for the top boundary condition, by multiplying the applied C_{input} (M/L^3) by the top boundary flux q_{surf} (and/or q_{irr}) (L/T), one obtains the specific mass of solute, M_s, applied per unit surface area. Accordingly, if one wants to apply a given solute mass per unit area at the surface, the concentration to be used as input concentration, (C_{input} in the code), is:

$$
C_{input}=\frac{M_s}{q_{surf}\Delta t_{app}} \tag{66}
$$

where Δt_{app} is the duration of the solute application. This equation also allows for calculating the time of application of the solute in order to apply a solute mass with a known concentration solution and a known surface flux.

The code allows for either constant or variable concentrations at the soil surface.

In the first case, the code requires to input the starting time (tC_{input}) and the end time of application (tC_{end}), as well as the applied concentration (C_{input}). Obviously, Δt_{app}= tC_{end} - tC_{input}.

For the case of variable concentrations, the code requires the C_{input} time by time.

The FLOWS code uses either a so-called concentration- type (first-type or Dirichlet-type) input condition or a flux-type (third type or Cauchy-type) boundary condition (van Genuchten and Parker, 1984) at z=0:
The first-type condition may be used to prescribe concentration at the top of the soil profile (z=0), as follows:

$$C(0,t) = C_{input} \tag{67}$$

that is, the concentration is specified at the inlet boundary. The ADE solution with this inlet boundary condition provides solute fluxes (or flux concentrations) at any nodes in the soil profile. In other words, the *C* values coming from the ADE solution with the first-type inlet condition have to be interpreted as the mass of solute per unit fluid discharge, C_f, and the mass flux, M_f [$ML^{-2}T^{-1}$], that is the mass passing through a node *i* per unit surface and unit time, is calculated as

$$M_{f_i}^{\,j} = q_i^j C_{f_i}^{\,j} \tag{68}$$

with *q* being the darcian water flux at node *i* and time step *j*. Accordingly, for mass conservation, the integral over time of equation 68 at a node *i* should result in the mass applied at the soil surface

$$\int_o^\infty M_{f_i} dt \tag{69}$$

which for a finite simulation time, t_{max}, may be approximated as

$$\sum_{j=1}^{j(t_{max})} M_{f_i}^{\,j} \Delta t^j = M_s \tag{70}$$

with M_s the specific mass of solute, that is the mass applied per unit surface area, t_{max} the time at the end of the simulation period and *j* the time step counter. The equation 70 is strictly correct only at the nodes where all the solute mass applied at the soil surface is recovered (passed through the node downward) in the time period from 0 to t_{max}. In all the other nodes, the sum in the equation 70 would provide a recovered mass lower than M_s.
The third-type condition may be used to prescribe the concentration flux at the inlet boundary (z=0) as follows:

$$C \text{ - } \frac{D_h(z)}{v}\frac{\partial C}{\partial z}\bigg|_{z=0} = C_{input} \tag{71}$$

where $v=q/\theta$ is the average pore water velocity and the subscript z=0 means that equation has to be evaluated just inside the inlet boundary (Huang et al., 1996b).

The ADE solution with the third-type inlet boundary condition provides volume-averaged (or resident) concentrations at any nodes in the soil profile.

In other words, the *C* values coming from the ADE solution with the third-type inlet condition have to be interpreted as the mass of solute per unit water volume, C_r, and the soil volume averaged mass, M_r [ML^{-3}], at each node *i* and time step *j*, may be calculated as:

$$M_{r}{}_{i}^{j} = \theta_i^j C_{r}{}_{i}^{j} \tag{72}$$

The applied specific mass of solute, M_s, may be obtained by calculating the integral over depth of M_r:

$$M_s = \int_o^\infty M_r{}^j dz \tag{73}$$

The use of either first-type or third type boundary conditions may produce discrepancies between flux and resident concentrations. These discrepancies may be quite insignificant in the case of dispersivities values in the order of centimetres (Parker and van Genuchten, 1984). This may be explained by considering the relationship between C_f and C_r:

$$C_f = C_r - \frac{D}{v}\frac{\partial C_r}{\partial z} \tag{74}$$

so that mathematically $C_f \rightarrow C_r$ with $D/v \rightarrow 0$.

However, with increasing dispersive transport over convective transport, the distributions of C_f and C_r gradually diverge, especially at early times and for relatively short transport domains (van Genuchten and Parker, 1984).

As bottom boundary conditions, in the case of downward flow, FLOWS multiplies the flux through the bottom of the soil profile by the concentration in the last node *i=nz* of the transport domain $q_{bot}{}^j C_{nz}^j$ to obtain the mass outflow at the bottom boundary. In the case of upward flux, the mass inflow at the bottom boundary is calculated as ${q_{bot}}^{j} C^{j}_{bot}$, with C_{bot} being, for example, the concentration in the groundwater.

In the case of soil drainage, the mass flux, M_{dr}, leaving each node *i* at the time step *j* by drainage is calculated as:

$$M_{dr}{}_{i}^{j} = S_{dr}{}_{i}^{j} C_{i}^{j} \tag{75}$$

As for the initial conditions, the user needs to specify the solute concentrations, *Cin* (gcm^{-3}), in the soil water. *Cin* may indifferently be C_r or C_f.

The code allows for setting a different dispersivity, λ, value for each of the simulation nodes.

As for the solute adsorption, the code allows for either a linear adsorption isotherm or a Freundlich isotherm. In the first case, the code requires to input the slope (the partition coefficient) of the isotherm. In the Freundlich option, the exponent of the non-linear isotherm is also required.

As for the decay, a first order decay rate coefficient of transformation, μ, has to be given as input. The code allows for setting a different decay coefficient value for each of the simulation nodes.

3.6. Solute transfer from soil solution to runoff water

In FLOWS, solute flux, q_s (M/L^2/T), across the soil surface interface is related to the difference in concentration between soil solution, C, and runoff water, C$_{\mathrm{run}}$, through a mass transfer coefficient, k_{run} (L/T):

$$q_s(0, \mathrm{t}) = -\theta D_h \frac{\partial C}{\partial z} + v\theta C \bigg|_{z=0} = (-\theta k_{run}[C(0,t) - C_{run}]) \tag{76}$$

The equation above accounts for the convective and dispersive modes of transfer between soil and runoff water (Wallach and van Genuchten, 1990). Convective mass transport is directed downward, while diffusive-dispersive transport is directed upward. k_{run} (L/T) is mainly controlled by the diffusion coefficient but it is also influenced by flow characteristics such as runoff water depth, rainfall intensity and duration, surface roughness.

Assuming that C_{run} can be neglected, the equation above becomes:

$$-D_h \frac{\partial C}{\partial z} + (v + k_{run})C \bigg|_{z=0} = 0 \tag{77}$$

applying for finite values of k_{run}, from which the flux to runoff may be calculated.

4. FLOWS modules for specific applications in agro-environmental hydrological modelling

4.1. Irrigation

In the case of irrigation, irrigation fluxes, q_{irr}, are added to the atmospheric (rainfall) fluxes, q_{surf}. The irrigation fluxes may come either from irrigation volumes actually supplied by a farmer or irrigation volumes calculated by the model according to a criterion based on the average pressure head in the root depth, D_r (see the section below on the irrigation criterion adopted by the code).

Irrigation options

FLOWS considers three options: 1) No irrigation; 2) Irrigation calculated by the model; 3) Irrigation provided by the user.

Irrigation calculated by the model

The criterion used by the model to calculate the time for irrigation and the irrigation volume is based on a comparison between a pressure head, h_{av}, averaged over the root depth, D_r, as simulated by the model, with a critical pressure head, h_{crit}, inducing some stresses to the crop (in terms of yield, product quality, …). Schematically, the irrigation criterion is summarized in figure 5a,b.

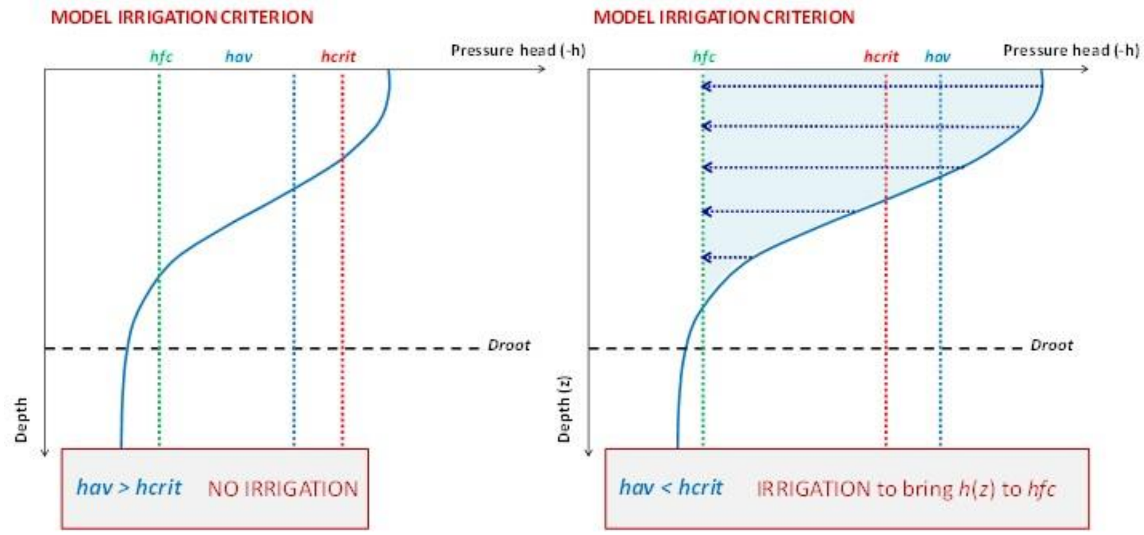


Figure 5. Graphical view of the criterion used by the model to calculate the time for irrigation and the irrigation volume. (a) h_{av} higher than h_{crit}, no irrigation is required; (b) h_{av} lower than h_{crit}, irrigation is required to bring the pressure head at the field capacity, h_{fc}.

If h_{av} remains higher than the critical pressure head, h_{crit}, the average pressure head h lies above the stress condition and no irrigation is required (figure 5a). Irrigation starts any time h_{av} becomes lower than h_{crit} (figure 1b). In other words, it is assumed that stress starts when the average pressure head in the root zone becomes lower than the threshold for water stress.
Irrigation aims to bring the actual water content at each depth to the water content at field capacity. At the beginning of each simulation input time (t_{star} in the code), the model calculates the gross irrigation height as the difference of water storage, Δ_{stor}, between the water content at the field capacity and the actual water content in the *irrigation depth*, z_{irr}, to be supplied on that day. The *irrigation depth* is the depth over which the h_{av}, to be compared to h_{crit}, is computed by the model. The *irrigation depth* may be set at any value even if a reasonable value should be the maximum root depth D_r;

$$\Delta_{stor} = \int_0^{z_{irr}} (\theta_{fc} - \theta(z))dz \quad \text{(cm of water)} \tag{78}$$

Figure 6 provides a schematic view of this calculation.

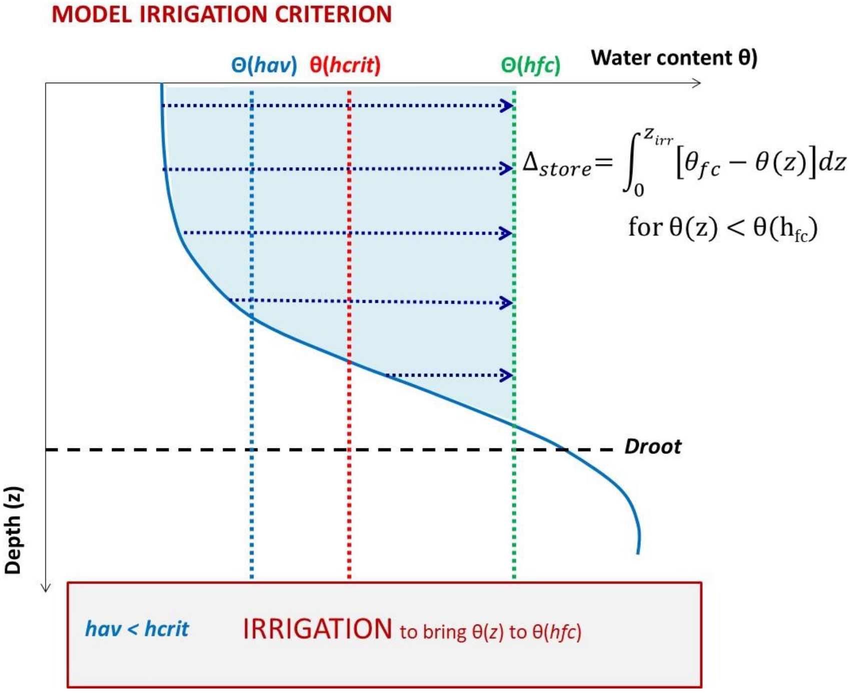

MODEL IRRIGATION CRITERION
Θ(hav)
θ(hcrit)
Θ(hfc)
Water content θ)
$\Delta_{store} = \int_0^{z_{irr}} [\theta_{fc} - \theta(z)]dz$
for θ(z) < θ(h_fc)
Droot
Depth (z)
hav < hcrit
IRRIGATION to bring θ(z) to θ(hfc)

Figure 6. Graphical view of the irrigation volume calculation according to the criterion shown in Figure 5.

Net irrigation, $q_{irr,net}$, to be supplied on day t is calculated as the difference between $q_{irr,gross}$ and the eventual rain falling at the time when the irrigation starts:

$$q_{irr,net} = q_{irr,gross} - q_{surf} \tag{79}$$

If the rainfall exceeds $q_{irr,gross}$, irrigation water is not supplied.

It is also possible to select an on-farm irrigation efficiency, IE, of the irrigation system used for the crop considered, so that the actual irrigation amount, $q_{irr,act,}$ may be obtained as:

$$q_{irr,act} = q_{irr,net} \ / \ IE \quad \text{(cm of water)} \tag{80}$$

The code allows for selecting the periods of irrigation during the growth season, as well as the soil depth to be irrigated and the threshold value, h_{crit}, for irrigation.

Irrigation provided by the user

With this option, the model allows the user to introduce the q_{irr} fluxes in addition to the fluxes coming from rainfall, q_{surf}

4.2. Modelling Temperature distribution in soil

Many physical, chemical, and biological processes in soils are influenced by soil temperature and its evolution in space and time. Thus, predicting and interpreting processes involving soil, water and vegetation interactions, also requires accurate modelling of the soil temperature distributions, $T(z, t)$, as a function of depth z and time t.

Temperature modelling along a vertical soil profile is frequently carried out by considering the heat conduction equation (Holmes et al., 2008):

$$\frac{\partial T}{\partial t} = \frac{\partial}{\partial z} D_T \frac{\partial T}{\partial z} \tag{81}$$

where $D_T = \frac{\lambda_h}{C_h}$ is the soil thermal diffusivity (L^2T^{-1}), λ_h is the soil thermal conductivity and C_h is the soil volumetric heat capacity.

The equation above can be solved by assuming periodically variable heat fluxes at z=0 and that at large depths, T becomes independent on the depth z.

At z=0, periodically variable heat fluxes results from both daily and annual cycles. At z=0, the conductive heat flux, q_{cnd}, is:

$$q_{\mathrm{cnd}} = G = \ H - Rn - LE \tag{82}$$

where H, *Rn and LE* are respectively the convective heat flux, the net radiation and the evaporative (latent) heat flux. As for the bottom boundary condition, it is generally assumed that for $z \rightarrow \infty$ the temperature approaches the mean air temperature, T_a, so that $T(z,t)=T_a$.

Equation 81 assumes that the change in temperature over a given interval is governed only by heat conduction. Actually, as discussed by Holmes et al. (2008), the convective heat flux, H is assumed to act in a very shallow surface layer, of 0.1-0.2 cm depth. As for the heat flux through evaporation, LE, it may extend to 5 cm below the surface for dry soils. So, to be rigorous, the right side of the equation should include a second term describing the latent energy loss, at least in the shallow boundary layer where evaporation takes place. Below this layer, the change in temperature is controlled only by heat conduction, as described by equation 81. In many numerical approaches to model soil temperature, this approximation is frequently extended to all depths below the soil surface and equation 81 is adopted to describe the temperature distribution along the whole soil profile. The solution to the heat flow equations proposed by van Wijk and De Vries (1963) represents an early example of such an approximation. The authors described the daily and annual evolution of soil temperature by sine waves of amplitude A_T (°C), varying around an average temperature, T_{av} (°C). T_{av} is considered constant with depth, due to the assumption of heat conservation. Assuming D_T to be constant with depth and time, a semi-infinite soil profile and the following sinusoidally varying temperature at the soil surface:

$$T(0.t)=T_{av}+A_T\sin\left[{}^1\!/_2\,\pi+\omega(t-t_{Tmax})\right] \qquad (83)$$

van Wijk and De Vries (1963) provided an analytical solution to equation 84 as:

$$T(z.t)=T_{av}+A_T e^{z/z_D}\sin\left[{}^1\!/_2\,\pi+\omega(t-t_{Tmax})\text{-}\frac{z}{z_D}\right] \qquad (84)$$

with $\omega=2\pi/\tau$ the angular frequency (1/d), τ the period of the wave (d), t the time starting from the January first and t_{Tmax} corresponding to the time when the temperature reaches the maximum. The damping depth, z_D, is the depth at which the amplitude of surface temperature oscillations is reduced by e^{-1} and is described as:

$$z_D=\sqrt{\frac{2D_T}{\omega}} \qquad (85)$$

in general, the amplitude of the temperature wave is maximum at the soil surface and decreases with depth.

The solution above may be used for either diurnal (τ=1) or annual (τ=365) temperature evolution. For daily fluctuations, the maximum temperature occurs shortly after solar noon at the surface, but lags in time with increasing depth.

FLOWS model uses a combined approach, provided by van Wijk and De Vries (1963), based on the superposition of the annual and daily sinusoidal fluctuation around a constant value of the soil. The approach sums two sinusoids, one for daily and the other for annual fluctuations, each having a constant amplitude and allows describing the daily fluctuations by accounting for the change of the daily average temperature at the surface over the year. Following this combined approach, the top boundary condition becomes:

$$T(0.t)=T_{av.y}+A_{T.y}\sin\left[{}^1\!/_2\,\pi+\omega_y(t-t_{Tmax.y})+A_{T.d}\sin\left[{}^1\!/_2\,\pi+\omega_d(t-t_{Tmax.d})\right]\right] \qquad (86)$$

with subscripts *y* and *d* used respectively for annual and daily values.
In the equation, the term $T_{av,y}+A_{T,y}\sin\left[{}^{1}/_{2}\,\pi+\omega_y\left(t-t_{Tmax,y}\right)\right]$ represents approximately the average temperature for the day corresponding to the time *t*, whereas the diurnal variation around this average temperature is described by the term $A_{T,d}\sin\left[{}^{1}/_{2}\,\pi+\omega_d\left(t-t_{Tmax,d}\right)\right]$.
With the combined surface boundary condition, the temperature along the soil profile over time is obtained as:

$$T(z.t)=T_{av.y}+A_{T.y}e^{-z/z_{D.y}}\sin\left[{}^{1}/_{2}\,\pi+\omega_y\left(t-t_{Tmax.y}\right)-\frac{z}{z_{D.y}}\right] + A_{T.d}e^{-z/z_{D.d}}\sin\left[{}^{1}/_{2}\,\pi+\omega_d\left(t-t_{Tmax.d}\right)-\frac{z}{z_{D.d}}\right] \qquad (87)$$

The approach was also used by Elias et al. (2004).
Assuming D_T (and thus the ratio $\frac{\lambda_h}{C_h}$) constant is an approximation of the real field soil conditions. Also, the boundary condition provided by equation 9 is not always a satisfying approximation for soil surface temperature. Finite difference numerical solutions of the equation 81 with λ_h and C_h not considered constant with depth and time provides more flexibility because may use more realistic boundary conditions (Wierenga and de Wit, 1970; Hanks et al., 1971; Milly, 1982; Horton et al., 1984).
However, when compared to experimental data, the analytical solution in equation 9 provides acceptable results (Jury et al., 1991). More recently, Holmes et al. (2008) evaluated the analytical solution to model soil temperature profiles in a bare soil. They found the commonly used solution to the heat flow equation by van Wijk and De Vries (1963) perform well when applied at deeper soil layers. Higher errors were found when applying the model very close to the surface. These errors were found to be related to the overlooking of latent energy loss in the shallower soil layer below the surface, an issue which also remains for finite difference numerical solutions of the more general heat transport equation.
The figure 7 provides an example of application of equation 87 to a soil with the following temperature conditions and thermal diffusivity (table 1):

Table 1. Temperature conditions at the soil surface and thermal diffusivity used for the graphs in figure 7 and figure 8

	T_{av} (°C)	A_T (°C)	ω (1/d)	τ (d)	t_{Tmax} (d)	D_T (cm^2/d)
Annual values (y)	12	10	0.0172	365	200	170

Daily values (d)		7	6.28	1	0.5	170

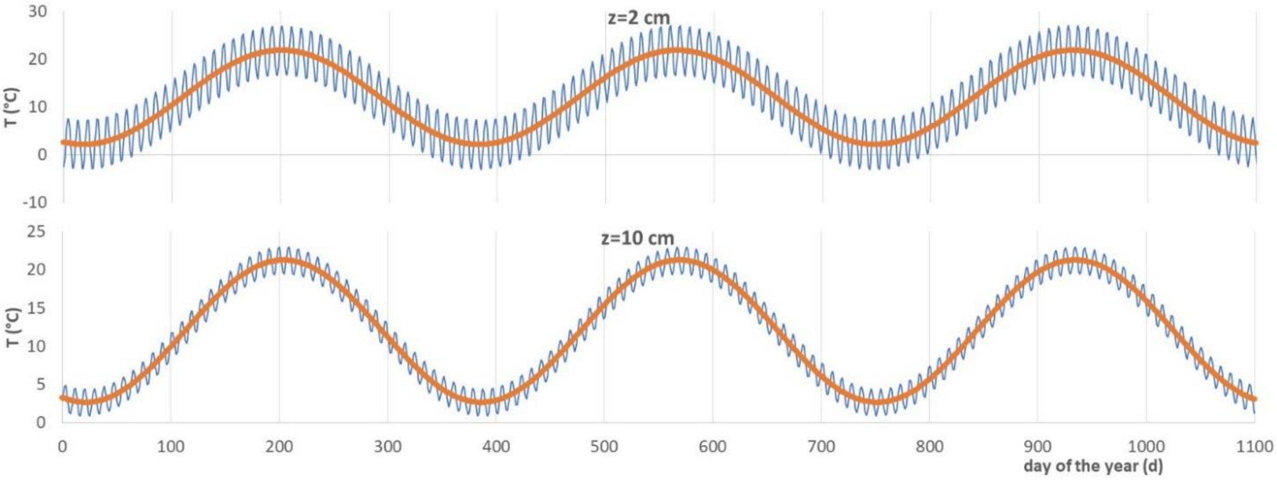


Figure 7. Daily (blue lines) and yearly (orange lines) fluctuations of temperature at 2 cm and 10 cm depth over three years obtained using the parameters provided in the table 1. The orange line and the blue line respectively provide the average daily temperature and the daily fluctuations about this average

The figure 8 provides the evolution of the average daily temperature over time at different depths for the same conditions given in the table 1.

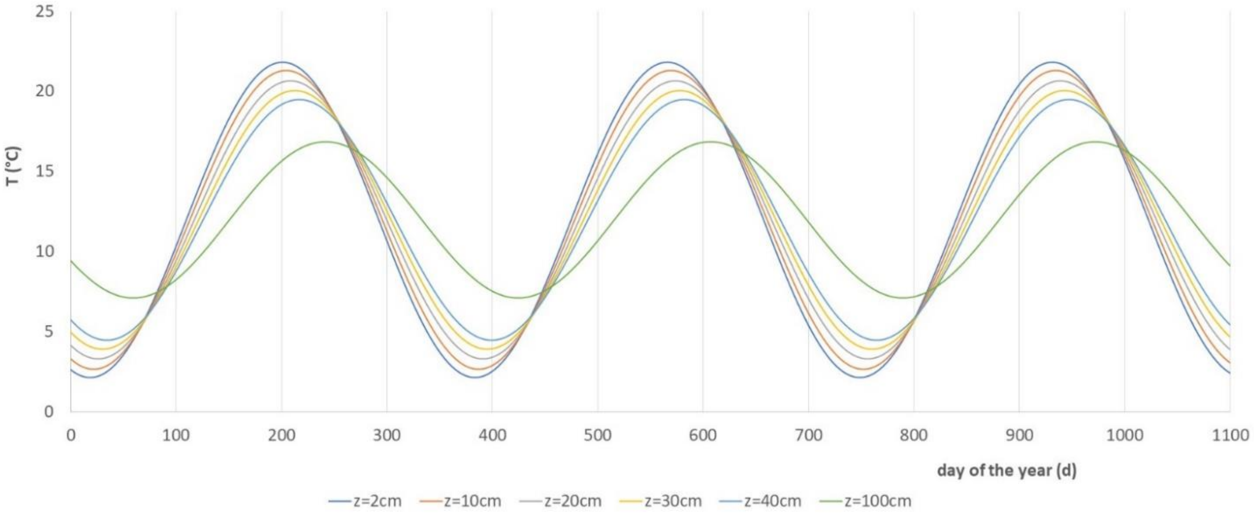


Figure 8. Yearly fluctuations of average daily temperature at 6 depths over three years obtained using the parameters provided in the table 1

4.3. Nutrients transformations and transport in soil

FLOWS model handles additions of fertilizers as organic matter in the forms of manure and crop residue, as well as mineral fertilizers. The model requires the incorporation depth (called z_{fert} in the FLOWS code) as input. It is assumed that fertilizer addition is distributed uniformly along the incorporation depth. FLOWS allows for: 1) Simulating Organic Matter decomposition and Carbon, Nitrogen and Phosphorus transport. In this case, carbon, nitrogen and phosphorus transformations are all controlled by the dynamics of the organic matter decomposition and, thus, also by the C:N and C:P ratios; 2) Simulating only Nitrogen transport. In this second case, nitrogen mineralization is simulated as an empirical decay reaction and independently on organic matter decomposition dynamics, thus without accounting for the C:N ratio in the organic matter (Stanford and Smith, 1972; Watts and Hanks, 1978, Kersebaum and Richter, 1991).

4.3.1. Simulating Organic Matter decomposition and Carbon, Nitrogen and Phosphorus transport

FLOWS handles additions of nutrients in the forms of manure, crop residue, mineral fertilizers. The model also assumes that manure applications contain 50% of urea and 50% of organic matter (Gusman and Marino, 1999).

Production of Carbon, Nitrogen and Phosphorus from organic matter

FLOWS considers two pools of organic matter: 1) Fresh Organic Matter (FOM) and 2) Stable Organic Matter or Humus (SOM) (Williams et al., 1983). FOM consists of organic matter in manure and crop residue (RSD), plus microbial biomass (MCB). SOM comes from a relatively fast microbial-induced decay of FOM. SOM, in turn, also decays due to microbial activity. The decay of both SOM and FOM produces carbon dioxide (CO_2), N-NH4 and P-PO4, which are thus transported in soil in both gas (the CO_2) and liquid phases (CO_2, N-NH4 and P-PO4).

In the following, the fate of Carbon, Nitrogen and Phosphorus will be dealt with separately, even if, as obvious, they are strictly interrelated.

Organic Carbon transformation processes in FLOWS

In FLOWS, the FOM decay is described as a first order decay chain according to the approach proposed by Jones et al. (1985) (see figure 9) and used in the EPIC model (Williams et al., 1983).

Organic Carbon transformation processes and pools in FLOWS

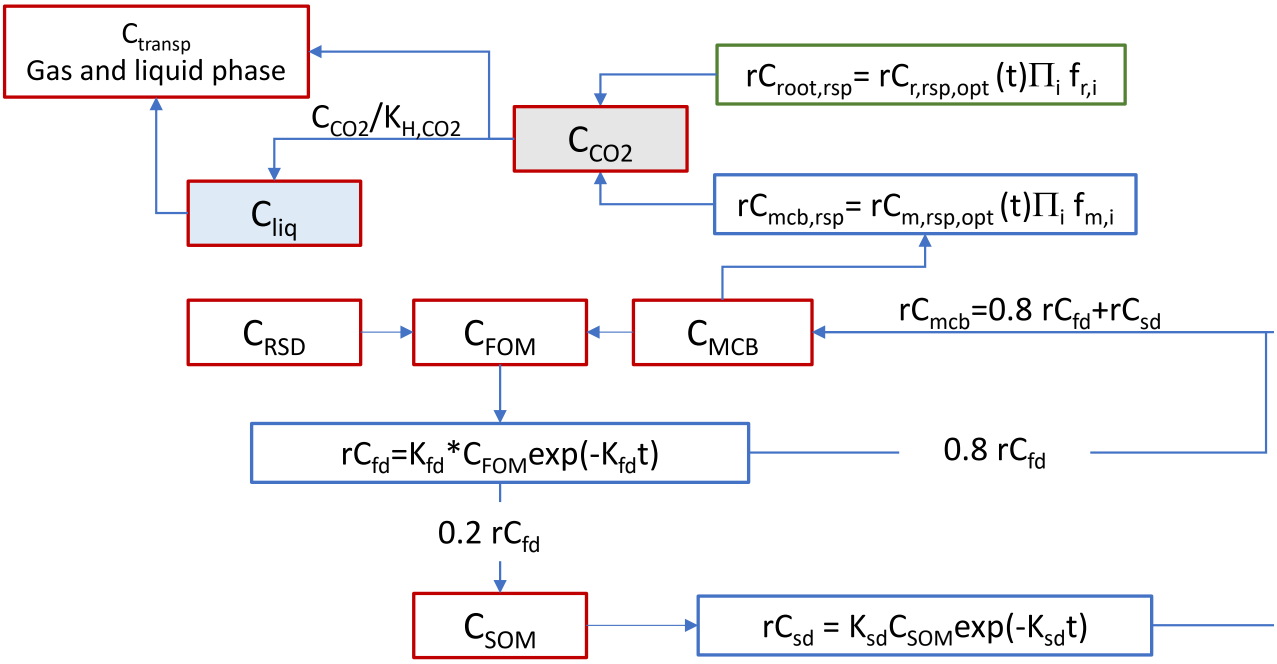

C_{transp} Gas and liquid phase
$C_{CO2}/K_{H,CO2}$
C_{CO2}
$rC_{root,rsp}= rC_{r,rsp,opt}(t)\Pi_i f_{r,i}$
C_{liq}
$rC_{mcb,rsp}= rC_{m,rsp,opt}(t)\Pi_i f_{m,i}$
C_{RSD}
C_{FOM}
C_{MCB}
$rC_{mcb}=0.8\ rC_{fd}+rC_{sd}$
$rC_{fd}=K_{fd}*C_{FOM}exp(-K_{fd}t)$
$0.8\ rC_{fd}$
$0.2\ rC_{fd}$
C_{SOM}
$rC_{sd} = K_{sd}C_{SOM}exp(-K_{sd}t)$

Figure 9. Schematic view of organic carbon transformation processes and CO_2 production in FLOWS. Red blocks are pools, blue blocks are rates and the green block refers to the CO_2 root respiration rate. The prefix *r* is for rates. Labels meaning: *fd*= decay of FOM to stable and microbes pools; *sd*= decay of SOM to microbes pool; *mcb,rsp*=microbes respiration; *root,rsp*=root respiration; $C_{m,rsp,opt}$=optimal CO_2 production by microbial respiration; $C_{r,rsp,opt}$=optimal CO_2 production by root respiration; $K_{H,CO2}$=Henry's constant for CO_2; *R*=universal gas constant

The rate of mineralization of FOM, rC_{fd} (g cm^{-2} d^{-1}) (from now on, the prefix *r* is for rate):

$$rC_{fd} = K_{fd}C_{FOM}exp(-K_{fd}t) \tag{88}$$

follows an exponential decay according to a decay parameter, K_{fd} (d^{-1}) obtained from an optimal decay constant for fresh organic matter, K_{fom}, which is reduced according to four reduction factors related to the soil temperature, F_T, water content, F_w, the carbon-nitrogen ratio and the carbon – phosphorus ratio, F_{CN} and F_{CP}, respectively:

$$K_{fd} = K_{fom}F_T F_W^{0.5} min(F_{CN}F_{CP}) \tag{89}$$

The four reduction factors calculations are reported in the following table (table 2):

Table 2. Reduction factors for the fresh organic matter decay constant, *Kfom* (Jones et al., 1984)

Decay constant, *Kfom*	$K_{fom} = 0.8 \qquad \frac{FOM_{unm}}{FOM_{in}} \geq 0.8$ $K_{fom} = 0.05 \qquad 0.8 > \frac{FOM_{unm}}{FOM_{in}} \geq 0.1$ $K_{fom} = 0.0095 \qquad \frac{FOM_{unm}}{FOM_{in}} < 0.1$	(90)
Temperature reduction factor	$F_T = 0.9T/[T + \exp{(7.63 - 0.312T)}]$	(91)
Water content reduction factor	$F_w = \frac{\theta}{\theta_{fc}} \qquad \theta < \theta_{fc}$ $F_w = \frac{\theta_{fc}}{\theta} \qquad \theta \geq \theta_{fc}$	(92)
C:N reduction factor	$F_{CN} = \exp\left[-0.693\left(\frac{C_{FOM}}{N_{FOM} + N_{IN}} - 25\right)/25\right]$	(93)
C:P reduction factor	$F_{CP} = \exp\left[-0.693\left(\frac{C_{FOM}}{P_{FOM} + P_{IN}} - 200\right)/200\right]$	(94)

In the table 2, θ and θ_{fc} are the actual water content and the water content at field capacity, C_{FOM}, N_{FOM}, P_{FOM} (g cm^{-2}) are the carbon, nitrogen and phosphorus content of the FOM, N_{IN} and P_{IN} (g cm^{-2}) are the nitrogen and the phosphorus contents form inorganic sources (mineral fertilizers). All the organic matter transformations are assumed to involve the whole FOM in soil. Thus, all the reduction factors in table 2 are calculated as averages in the whole incorporation depth, *zfert*.

Following the approach proposed by Jones et al. (1985), FLOWS assumes that about 20 percent of the carbon contained in the FOM, C_{FOM}, coming from the carbon in the microbial mass, C_{MCB}, and that in the residuals, C_{RSD}, mineralizes rapidly at a rate appropriate to non-structural carbohydrates. According to the same approach, about 70 percent of the residue mineralizes more slowly at a rate typical of cellulose-like materials, whereas the remaining 10 percent decomposes even more slowly at a rate appropriate to lignin. Accordingly, *Kfom*, assumes a different value depending on the ratio of the FOM to be still unmineralized, FOM_{unm}, to the initial FOM, FOM_{in}, in the fertilization depth, *zfert* (see equation 90 in table 2).

About 20 percent of the carbon coming from mineralization (0.2 *rCfd*) is incorporated into the stable organic matter (C_{SOM}) (the humus pool). The remaining (0.8 *rCfd*) goes to the microbes' pool and will be partly respired (see below).

The carbon in C_{SOM}, in turn, will mineralize according to a first order rate constant, *Ksd*:

$$rC_{sd} = K_{sd} C_{FOM} exp(-K_{sd} t) \tag{95}$$

The carbon mineralized from the C_{SOM} will cumulate to that coming from the FOM and will follow the same fate. Partly, the total mineralized carbon will be used by the microbes (rC_{mcb}) and partly will go to CO_2 by microbial respiration at a rate $rC_{mcb,rsp}$.

Microbial and Root respiration

FLOWS calculates microbial respiration, $rC_{mcb,rsp}$, and root respiration, $rC_{root,rsp}$ in a similar way, by following the approach proposed by Simunek and Suarez (1993).

FLOWS considers only the production of CO_2 from microbial and root respiration, whereas neglects that coming from other chemical reactions with soil organic and mineral components, by assuming them of relatively minor importance for CO_2 balance in soil. According to Šimůnek and Suarez (1993), optimal CO_2 production coming from microbial respiration, $rC_{m,rsp,opt}$ (g cm^{-2} d^{-1}), and root respiration, $rC_{r,rsp,opt}$ (g cm^{-2} d^{-1}), is assumed to be a function of soil depth and is affected by water content (or pressure head), temperature, pre-existing CO_2 concentration in soil. Both the diurnal and seasonal influences are assumed to be already accounted for by the effect of the temperature.

Thus, the microbial and root respiration (g cm^{-3} d^{-1}) are calculated by equations 96 and 97, respectively:

$$rC_{mcb,rsp} = rC_{m,rsp,opt} f(z) f(h) f(T) f(CO_2) \tag{96}$$

$$rC_{root,rsp} = rC_{r,rsp,opt} f(z) f(h) f(T) f(CO_2) \tag{97}$$

The equations for calculating the factors in equation 96-97 are summarised in table 3.

Table 3. Equations for calculation of the factors for the dependence of microbial (equation 96) and root (equation 97) respiration on soil depth (z), pressure head (h), temperature (T) and CO_2 concentration (Šimůnek and Suarez, 1993)

Depth dependence, *f(z)* (cm^{-1})	$f(z) = g(z)$	(98)
Water content (pressure head) dependence, *f(h)* (dimensionless)	$f(h) = \frac{log\lvert h\rvert - log\lvert h_1\rvert}{log\lvert h_2\rvert - log\lvert h_1\rvert}$ $\quad h_1 < h < h_2$ $f(h) = \frac{log\lvert h\rvert - log\lvert h_3\rvert}{log\lvert h_2\rvert - log\lvert h_3\rvert}$ $\quad h_2 < h < h_3$ $f(h) = 0$ $\quad h < h_3;\ h > h_1$	(99)

Temperature dependence, *f(T)* (dimensionless)	$f(T) = \exp\left[\frac{E(T - T_{20})}{RTT_{20}}\right]$	(100)
CO2 dependence on its own concentration, *f(CO₂)* (dimensionless)	$f(CO2) = \frac{0.21C_{CO2,g}}{0.42 - C_{CO2,g} - K_M^*}$	(101)

with *g(z)* being the root distribution function (see [SH1]paper 1), $C_{CO2,g}$ the concentration of CO_2 in the gas phase, $K_M^* = 0.21 - K_M$, with K_M the Michaelis-Menten constant, *R* the universal gas constant (8.314 J mol^{-1} K^{-1}), *T* the absolute temperature (°C + 273.15), T_{20}=293.15K and *E* the activation energy for the reaction. According to Suarez and Šimůnek (1993), FLOWS assumes K_M^* =0.19 cm^3cm^{-3}, E=5000 J mol^{-1}, the pressure head for optimal soil respiration, h_2 = -100cm in water column, the pressure head when respiration ceases, h_3 = -10^7cm and the pressure head when the soil is in close-to-saturation (anaerobiosis point), h_1 = air entry pressure head. Note that T_{20} is the temperature when f(T)=1. For T> T_{20} f(T)>1 and for T< T_{20} f(T)<1.

FLOWS uses a default value of 0.00036 g cm^{-2} d^{-1} for $rC_{r,rsp,opt}$, corresponding to the value of 0.002 m^3 m^{-2} d^{-1} suggested by Suarez and Simunek, 1993 and assuming carbon dioxide weight 0.001836 g cm^{-3} (at the pressure of 1 atm = 101325 Pa and temperature of 25°C). As for optimal microbial respiration, FLOWS assumes $rC_{m,rsp,opt}$ =0.4 rC_{fsm} + 0.6 rC_{sm}.
In fact, given an average Carbon Use Efficiency of 0.4 (Jones et al, 1983; Spohn et al., 2016), 0.4 of rC_{sm} and 0.32 of C_{fsm} (0.4 of 0.8 C_{fsm}) should be used to build the microbial biomass (immobilization), while the complement to 1 of rC_{sm} (0.6) and of rC_{fsm} (0.48) should be used for microbial respiration. In FLOWS, the immobilized *C* adds to the microbial pool (C_{MCB}) and thus to the fresh organic matter (C_{FOM}) pool.
The respiration from roots and microbes are assumed to be additive, so that the production rate of CO_2 coming from respiration (rC_{CO2}) (g cm^{-3} d^{-1}) is:

$$rC_{CO2} = rC_{mcb,rsp} + rC_{root,rsp} \tag{102}$$

rC_{CO2} will partly pass to liquid phase, C_{liq}, according to the following equation:

$$C_{liq} = \frac{rC_{CO2}}{K_{H,CO2}} \tag{103}$$

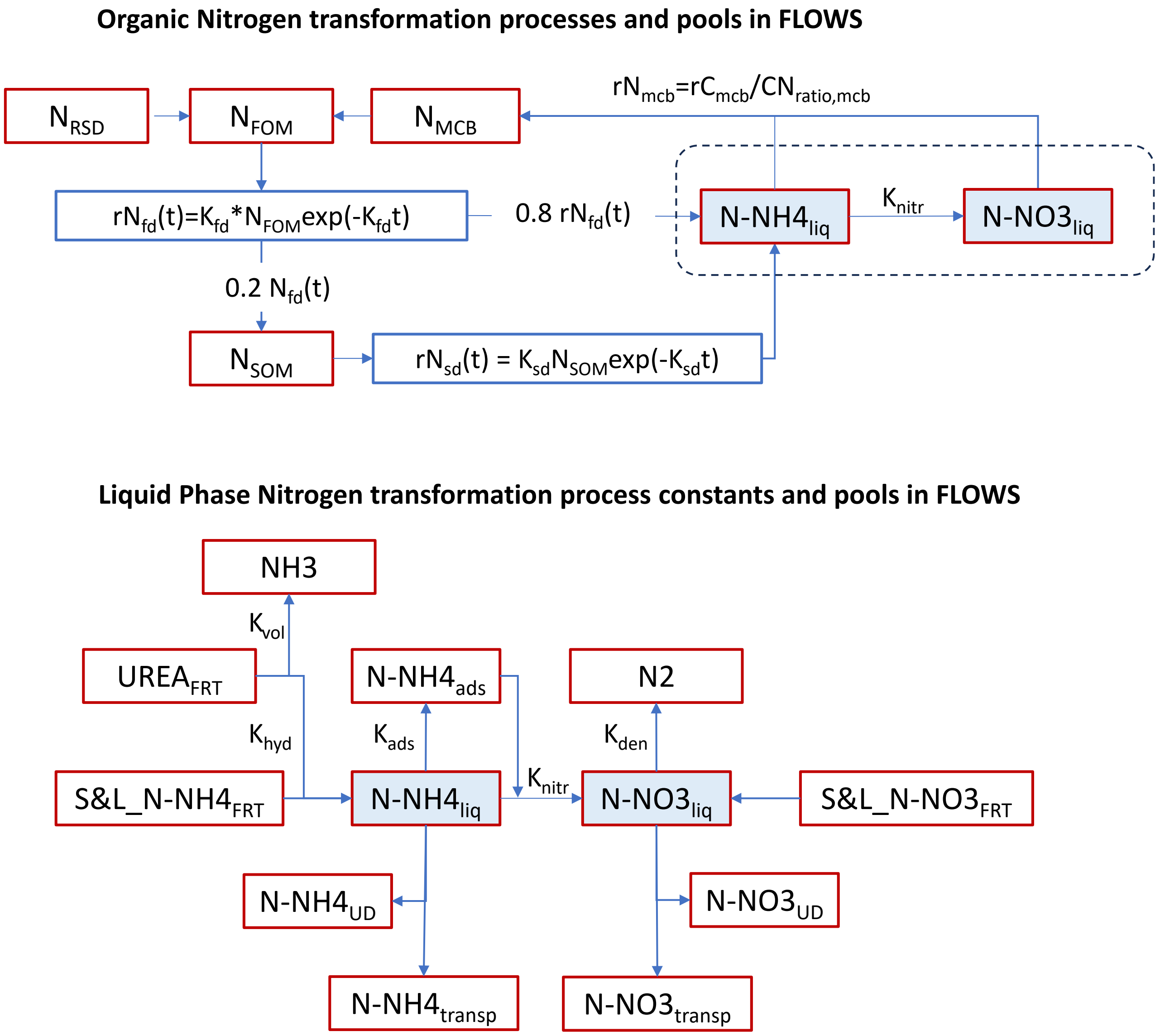
while the residual part, C_{gas}, will remain in gas phase. Both are passed to the ADE for transport process. Both C_{liq} and C_{gas} will be in g cm^{-2} d^{-1}. They are thus divided by the incorporation depth, z_{fert}, to obtain the concentrations in each calculation node in g cm^{-3} d^{-1} required by the ADE.
In equation 103, $K_{H,CO2}$ is the Henry's constant for CO_2 (dimensionless). It is obtained as (Sander, 2015).

$$
\begin{gathered}
K_{H,CO2} = \frac{1}{H_{cc}} = \frac{C_{gas}}{C_{liq}} \\
H_{cc} = H_{cp}RT \\
H_{cp} = \frac{C_{liq}}{p}
\end{gathered}
\tag{104}
$$

with H_{cp} the Henry solubility (mol m^{-3} Pa^{-1}), expressed as the ratio of the concentration of a species in the liquid (aqueous) phase and the partial pressure, p (Pa) of that species in the gas phase under equilibrium conditions. For the CO_2, H_{cp} is 0.00034 mol m^{-3} Pa^{-1} (see table 6 in Sander, 2015). H_{cc} is the dimensionless Henry solubility, R is the gas constant (8.314 J mol^{-1} K^{-1}) and T the absolute temperature (°C + 273.15).

Organic Nitrogen and Phosphorus transformation processes in FLOWS

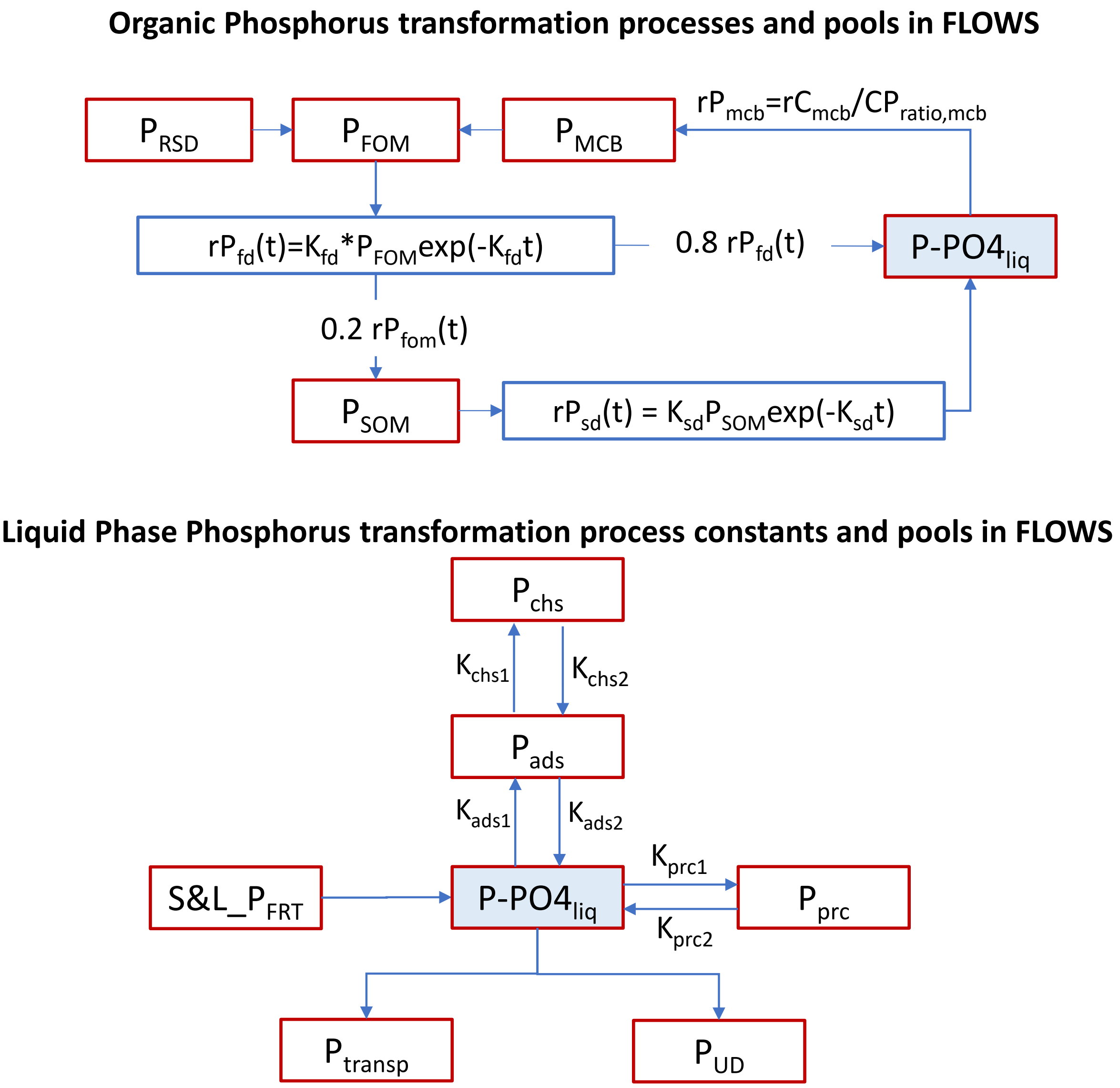
Organic Nitrogen and Phosphorus are introduced in FLOWS by a fertilizers table. The organic nitrogen and phosphorus transformations considered in FLOWS are summarised schematically in figure 10 and figure11.



Organic Nitrogen transformation processes and pools in FLOWS

Liquid Phase Nitrogen transformation process constants and pools in FLOWS

Figure 10. Schematic view of organic and mineral nitrogen transformation processes in FLOWS. The mineralization follows exactly the same paths already seen for organic carbon, with nitrogen mineralization proportional to that of the carbon. Constants K_{fd} and K_{sd} are the same as for carbon. The prefix *r* is for rates. The nitrogen immobilization in the microbial pool is regulated by the C:N ratio in the microbes ($CN_{ratio,mcb}$), which, according to Cleveland and Liptzin (2007), is assumed to be 8.5. The meaning of pools and constants is given in the text



Organic Phosphorus transformation processes and pools in FLOWS

P_{RSD}
P_{FOM}
P_{MCB}
$rP_{mcb}=rC_{mcb}/CP_{ratio,mcb}$
$rP_{fd}(t)=K_{fd}*P_{FOM}exp(-K_{fd}t)$
$0.8\ rP_{fd}(t)$
$P\text{-}PO4_{liq}$
$0.2\ rP_{fom}(t)$
P_{SOM}
$rP_{sd}(t) = K_{sd}P_{SOM}exp(-K_{sd}t)$

Liquid Phase Phosphorus transformation process constants and pools in FLOWS

Figure 11. Schematic view of organic and mineral phosphorus transformation processes in FLOWS. The mineralization follows the same paths already seen for organic carbon, with phosphorus mineralization proportional to that of the carbon. Constants K_{fd} and K_{sd} are the same as for carbon. The prefix *r* is for rates. The phosphorus immobilization in the microbial pool is regulated by the C:P ratio in the microbes ($CP_{ratio,mcb}$), which, according to Cleveland and Liptzin (2007), is assumed to be 60. The meaning of pools and constants is given in the text

In FLOWS, organic nitrogen and phosphorus mineralization follows the same path of the carbon and the immobilization rates (rN_{mcb} and rP_{mcb}) are proportional to that of the carbon according to C:N and the C:P ratios of the microbial mass in soil, which are assumed to be about 8.5 and 60, respectively (see Cleveland and Liptzin, 2007):

Nitrogen immobilization (compare figures 7 and 8)	$rN_{mcb} = \frac{rC_{mcb}}{CN_{ratio,mcb}}$	(105)
Phosphorus immobilization (compare figures 7 and 9)	$rP_{mcb} = \frac{rC_{mcb}}{CP_{ratio,mcb}}$	(106)

In other words, N and P mineralize according to equation 88, with C_{FOM} replaced by N_{FOM} and P_{FOM}, respectively. Likewise, N_{SOM} and P_{SOM} decay according to equation 95.

The nitrogen (N-NH4 and N-NO3) and phosphorus (P-PO4) coming from organic mineralization enter into the mineral N-NH4 and N-NO3 and P-PO4 pools and add to that eventually coming from mineral fertilizers, which for nitrogen can be urea (UREA$_{FRT}$), as well as solid and liquid NH4 and NO3 fertilizers (S&L_NH4$_{FERT}$, S&L_NO3 $_{FERT}$) sources; for phosphorus, solid and liquid PO4 fertilizers (S&L_PO4 $_{FERT}$).

The subsequent fate of these different pools of mineral N-NH4 and N-NO3 and P-PO4 is determined by different transformation reactions, frequently described as first-order decay, each with a specific constant, as well as by transport.

As for nitrogen, urea is hydrolysed according to a first-order decay with a constant k_{hyd}, producing NH4. Part of this volatilizes according to a decay constant k_{vol}:

Urea hydrolysis and ammonium volatilization (Liang et al., 2007)	$rNH4_{hUR} = UR_0 K_{hyd}\, exp(-k_{hyd}t)$ $rNH3_{vUR} = NH4_{hUR} K_{vol}\, exp(-k_{vol}t)$	(107)

Beside immobilization, N-NH4 and N-NO3 in the soil solution (N-NH4$_{liq}$ and N-NO3$_{liq}$) phase may undergo nitrification (k_{nitr}) and denitrification (k_{den}), respectively. N-NH4 may also be adsorbed to the solid fraction of the soil according to a linear isotherm with coefficient of distribution k_{ads}. Finally, soil N-NH4 and N-NO3 may be drawn by root uptake and/or artificial drainage (both unified under a sink term, labelled by subscript UD).

The remaining N-NH4 and N-NO3 concentrations are thus transported through soil by advection-dispersion (subscript *transp* in figures 10). Accordingly, for nitrogen transport, the advection–dispersion equation (ADE) is used twice:

$$\frac{\partial \theta C_{NH}}{\partial t} + \rho_b \frac{\partial C_{a,NH}}{\partial t} = \frac{\partial q C_{NH}}{\partial z} + \frac{\left(\partial \theta D \frac{\partial C_{NH}}{\partial z}\right)}{\partial z} - Ss_{NH} \qquad (108)$$

$$\frac{\partial\theta C_{NO}}{\partial t}+\rho_b\frac{\partial C_{a,NO}}{\partial t}=\frac{\partial qC_{NO}}{\partial z}+\frac{\left(\partial\theta D\frac{\partial C_{NO}}{\partial z}\right)}{\partial z}-Ss_{NO} \qquad (109)$$

The subscripts NH and NO are for ammonium, N-NH4, and nitrates, N-NO3, forms of nitrogen, respectively. We assume here that both N-NH4 and N-NO3 may be adsorbed to the solid phase (the N-NO3 on a positively charged surface). Note that the two equations are linked through the nitrification process which transforms part of the N-NH4 (a sink for equation 108) to N-NO3 (a source for the equation 109).

Specifically, Ss_{NH} and Ss_{NO} in the equations 108 and 109 are the source-sink terms of solute, all function of depth *z* and time *t*, and can be written respectively as:

$$Ss_{NH}=-Ss_{minNH}-Ss_{urea}-\ Ss_{frtNH}+Ss_{nit}+Ss_{vol}+Ss_{upNH}+Ss_{drNH}$$
$$Ss_{NO}=-Ss_{nit}-Ss_{frtNO}+Ss_{den}+Ss_{upNO}+Ss_{drNO} \qquad (110)$$

where Ss_{minNH} and Ss_{urea} are the N-NH4 coming from mineralization (see equation 88) and urea hydrolysis (see equation 107), respectively, Ss_{nit} is the N-NH4 transformed by nitrification, Ss_{imm} is the immobilized N-NO3. Ss_{den} and Ss_{vol} are the N-NO3 and the N-NH4 lost by denitrification and volatilization, respectively. Ss_{upNH} and Ss_{upNO} are the N-NH4 and N-NO3 root uptake, respectively. Likewise, Ss_{drNH} and Ss_{drNO} refer to the N-NH4 and N-NO3 losses through the artificial drainage.

Finally, Ss_{frtNH} and Ss_{frtNO} are the N-NH4 and N-NO3 source terms coming from mineral fertilizers additions (other than urea).

The source terms Ss_{minNH}, Ss_{urea} Ss_{frtNH}, and Ss_{frtNO}, are obtained by dividing the mineralization, urea hydrolysis and mineral fertilizer addition rates (all in g/cm^2 of soil per day) by the incorporation depth, *zfert* (cm), to obtain the corresponding *Ss* terms (in g/cm^3 of soil per day) to be added to each of the calculation nodes within *zfert*.

The table 4 summarises the equations controlling the liquid phase nitrogen transformation processes considered in FLOWS and also schematically illustrated in the figure 10 in the block about *Liquid phase nitrogen transformation process constants and pools in FLOWS*.

Table 4. The nitrogen transformation processes implemented in FLOWS. In the table, the *Ss* rates are in g/cm³ of soil per day

Nitrification (Cabon et al., 1991)	$Ss_{nit} = k_{nit} \times 1.07^{(T-T_{opt})} \frac{\theta}{\theta_{fc}} C_{NH} \quad \theta \leq \theta_{fc}$ $Ss_{nit} = k_{nit} \times 1.07^{(T-T_{opt})} \frac{\theta_{fc}}{\theta} C_{NH} \quad \theta > \theta_{fc}$	(111)
Denitrification (Lafolie, 1991; McGechan and Wu, 2001)	$Ss_{den} = 0 \quad \theta \leq \theta_d$ $Ss_{den} = k_{den} \times 1.07^{(T-T_{opt})} \frac{\theta - \theta_d}{\theta_{fc} - \theta} C_{NO} \quad \theta_d \leq \theta \leq \theta_s$ $\theta_d = 0.627\theta_{fc} - 0.0267 \frac{\theta_s - \theta}{\theta_s} \quad \theta_{fc}$	(112)
Root uptake of Nitrogen	$S_{upNH} = K_{r,NH} S_r C_{NH}$ $S_{upNO} = K_{r,NO} S_r C_{NO}$	(113)
Nitrogen losses to artificial drainage	$S_{drNH} = S_{dr} C_{NH}$ $S_{drNO} = S_{dr} C_{NO}$	(114)

In the table 4, UR_0 and *UR_NH4* are the initial urea concentration and the N-NH4 produced by the urea hydrolysis, respectively; k_{hUR} and k_{vUR} are the first-order rate coefficients for the two decay reactions (hydrolysis and volatilization, respectively).
k_{nit}, *and* k_{den}, are respectively the optimal first-order rate coefficients for nitrification of N-NH4, and denitrification of N-NO3; T is the actual soil temperature (°C); T_{opt} is the optimum temperature for the process (°C) (Cabon et al., 1991); θ_d is the threshold water content for denitrification.
$K_{r,NH}$ and $K_{r,NO}$ are the root uptake preference factors (dimensionless) for either the N-NH4 or the N-NO3, accounting for positive or negative selection of solute ions relative to the amount of soil water that is extracted (van Dam et al., 1997). For passive uptake, K_r=1.
S_r and S_{dr} are the source-sink terms related respectively to root uptake and artificial drainage respectively, and θ_{fc} is the water content at field capacity (assumed as the water content at pressure head h=-330 cm in water column).

As for phosphorus, FLOWS describes the fate of liquid phase phosphorus (P_{liq}) according to the decay reaction chains proposed by Mansell et al. (1977a). In general, inorganic P is rapidly converted from orthophosphate to less soluble forms. Phosphorus soil solution concentrations generally observed after inorganic P supply to the soil are of the order of 1μg/cm or less (Mansell et al., 1977b). The initial rapid removal of phosphorus from the soil solution may be attributed to relatively fast reactions, such as physical adsorption to soil colloidal material. The adsorption of P is usually considered to be reversible (Van der Zee and Van Riemsdijk, 1986; Barrow et al., 1981). Other additional much slower reactions involve precipitation of phosphorus as AI, Fe, and Ca phosphates. Also, a portion of the physically adsorbed phosphorus may slowly become surrounded by matrices of Fe and AI components and become occluded (chemisorption). Mechanisms of adsorption, precipitation, and chemical immobilization operate simultaneously and continuously with time to remove phosphorus from the soil solution.
The Mansell et al. (1977a) approach, also adopted in FLOWS, assumes the transfer of phosphorus between solution, adsorbed, chemisorbed and precipitated to be controlled by six reversible reactions (see the block in the figure 11 about *Liquid phase phosphorus transformation process constants and pools in FLOWS*). While adsorption occurs on pore walls and colloids, chemisorption (also defined occlusion by the Mansell et al., 1977) refers to (slow) transformation of weaker physical bonds of the adsorbed phosphorus to stronger chemical bonds. FLOWS assumes adsorption to follow a Nth order kinetic (with an average N of 0.35, as suggested by Mansell et al., 1977a) and a constant $K_{ads1,}$ whereas all the other reactions (desorption, chemisorption and mobilization, precipitation and dissolution) follow a first order kinetic, respectively with constants K_{ads2}, K_{chs1}, K_{chs1}, K_{prc1}, K_{prc2}, even if higher order may easily be developed in the code. Reactions rates depend on pH, nature and content of clay minerals, organic matter, carbonates, cation saturation and amount of phosphate applied. The

table 5 summarises the equations controlling the liquid phase phosphorus transformation processes considered in FLOWS.

The remaining P-PO4 concentrations are thus transported through soil by advection-dispersion (subscript *transp* in figure 11):

$$\frac{\partial \theta C_P}{\partial t} = \frac{\partial q C_P}{\partial z} + \frac{\left(\partial \theta D \frac{\partial C_P}{\partial z}\right)}{\partial z} - Ss_P \tag{115}$$

with Ss_P being the sum of adsorption, desorption, precipitation, dissolution, root uptake, drainage losses and fertilizer (organic/inorganic) components:

$$\begin{aligned} Ss_P = -Ss_{minP} - Ss_{frtP} - \theta\left(K_{ads1}C_P^N + K_{prc1}C_P\right) + \rho_b\left(K_{ads2}C_{a.P} + K_{prc2}C_{p.P}\right) \\ + Ss_{upP} + Ss_{drP} \end{aligned} \tag{116}$$

In the equations 116, as well as in the equations in the table 5, subscript *P* is for P-PO4, whereas subscripts *a*, *p* and *c* are respectively for adsorbed, precipitated and chemisorbed phosphorus concentrations.

As for nitrogen, the source terms Ss_{minP} and Ss_{frtP}, are obtained by dividing the mineralization rate and mineral phosphorus fertilizer addition rates (all in g/cm^2 of soil per day) by the incorporation depth, *zfert* (cm), to obtain the corresponding *Ss* terms (in g/cm^3 of soil per day) to be added to each of the calculation nodes within *zfert*.

Table 5. The phosphorus transformation processes implemented in FLOWS (Mansell et al., 1977a). In the table, the *Ss* reaction rates are in g/cm^3 of soil per day

Phosphorus adsorption	$Ss_{ads} = \frac{\partial(\rho_b C_{a,P})}{\partial t} = K_{ads1}\theta C_P^N - (K_{ads2} + K_{chs1})\rho_b C_{a,P} + K_{chs2} C_P$	(117)
Phosphorus chemisorption	$Ss_{chs} = \frac{\partial(\rho_b C_{c,P})}{\partial t} = K_{chs1}\rho_b C_{a,P} - K_{chs2}\rho_b C_{c,P}$	(118)
Phosphorus precipitation	$Ss_{prc} = \frac{\partial(\rho_b C_{p,P})}{\partial t} = K_{prc1}\theta C_P - K_{prc2}\rho_b C_{p,P}$	(119)
Root uptake of Phosphorus	$S_{upP} = K_{r,P} S_r C_P$	(120)
Phosphorus losses to artificial drainage	$Ss_{drP} = S_{dr} C_P$	(121)

4.3.2. Simulating only Nitrogen transport

FLOWS allows to simulate only nitrogen transport by partly following the empirical approach adopted in the RISK-N model (Gusman and Marino, 1999) for the mineralization of organic nitrogen. In this case, soil organic N is conceptually divided into active and passive fractions (Gusman and Marino, 1999). The active fraction includes organic N involved in the process of mineralization and is divided into rapid and slow mineralization fractions. The rapidly mineralizing N consists of recent additions of manure and crop residue. The slow fraction consists of resident soil N still mineralizing, as well as the remaining organic N from past manure and crop residue applications.

As for the previous case, the model handles additions of nitrogen in the forms of manure, crop residue, mineral fertilizers. The model requires the incorporation depth (called *zfert* in the FLOWS code) as input. It is assumed that nitrogen addition is distributed uniformly along the incorporation depth. The model assume that manure applications contain 50% of urea and 50% of organic N. As for crop residues incorporation, it is assumed that 50% consists of rapidly mineralizing fraction, 45% of slowly mineralizing organic N and 5% is considered as passive fraction (Gusman and Marino, 1999).

Again, the mineralization process is assumed to be described by a first-order decay equation (Stanford and Smith, 1972; Watts and Hanks, 1978):

$$rNH4_{min} = k_{mOM} NOM_0 \, exp(-k_{mOM} t) \tag{122}$$

with $rNH4_{min}$ the rate of N-NH4 production from mineralization (g/cm^2 of soil per day), NOM_0 the initial pool of nitrogen in organic matter and k_{mOM} is a first order rate constant for the mineralization process. Now, the latter does not depend on the C/N ratio but still depends on soil temperature and water content, being a function of microbial processes. Also, as the mineralization rates change according to the type of organic N pools to be mineralized (rapid or slow), two different equations can be used to determine k_{mOM} in equation 122 (Kersebaum and Richter, 1991):

$$\begin{gathered} k_{mOM}(T) = k_{rp}(T) = 5.6 \times 10^{12} \, exp\left(\frac{-9800}{T + 273}\right) F_w \\ k_{mOM}(T) = k_{sw}(T) = 4.0 \times 10^{9} \, exp\left(\frac{-8400}{T + 273}\right) F_w \end{gathered} \tag{123}$$

where T is the actual soil temperature (°C) and the water content factor for mineralization, F_w, is calculated as in equation 92. For the calculation of $k_{rp}(T)$ and $k_{sw}(T)$, FLOWS uses the averages of both F_w and T within the incorporation depth, *zfert*.

Again, *rNH4_{min}* is divided by *zfert* to obtain the Ss_{frtNH}.

As for immobilization, in this case FLOWS does not consider the C/N ratio but follows an empirical equation decay (Cabon et al., 1991):

$$
\begin{aligned}
Ss_{imm} &= k_{im} \times 1.05^{(T-T_{opt})} F_w C_{NO} \qquad \theta \leq \theta_{fc} \\
Ss_{imm} &= k_{im} \times 1.05^{(T-T_{opt})} \frac{1}{F_w} C_{NO} \qquad \theta > \theta_{fc}
\end{aligned}
\tag{124}
$$

with a decay constant k_{im} and where T_{opt} is the optimal temperature for the process. All the other processes (Urea hydrolysis; Ammonia volatilization (N-NH4 to N-NH3), Nitrification (N-NH4 to N-NO3); Denitrification (N-NO3 to –N-N2); Plant uptake; Drainage losses, are all described by using the equations already seen for the previous case.

Again, the transport of N-NH4 and N-NO3 is described by solving simultaneously the equations 108-109, with Ss_{NO} term in equation 110 now including also the Ss_{imm} losses.

As mentioned, FLOWS deals with water and nitrogen transformations and transport in an integrated way, so that at each simulation time step the water contents and fluxes calculated by solving the Richards equation are used as input for the nitrogen equations, as illustrated in the figure 13

4.4. Calculation of the Osmotic Potential for salinity stress

For the calculation of the salinity stress reduction factor, α_s, the model requires the osmotic potential, h_{os}, to be used in the Mass and Hoffman approach (Mass and Hoffman, 1977) and in the van Genuchten and Hoffman approach (van Genuchten and Hoffman, 1984), as well as in the combined approach.

The code uses the concentration coming from the ADE solution for the calculation of the h_{os} values in all the nodes.

Firstly, the code calculates the soil solution electrical conductivity, EC as

$$EC = C * \frac{10^6}{640} \tag{125}$$

with EC in dSm^{-1} and C in gcm^{-3}.

Thus, the osmotic potential, h_{os}, is obtained as:

$$h_{os} = 360EC \tag{126}$$

5. Benchmarking (intercode comparison)

The reliability of FLOWS in numerical simulations has been evaluated by comparing the model results to those coming from HYDRUS 1D simulation for a set of water flow and solute transport processes involving different top and bottom boundary conditions.

This intercode comparison is especially effective as both the models are based on the same fundamental equations to describe water flow (Richards equation) and solute transport (Advection-Dispersion equation). Both the codes use similar strategies to adapt time stepping to a convergence criterium, allowing the time step size to increase when the code converges rapidly and decrease when there are convergence problems. Both the codes allow to set the several top and bottom boundary conditions, both constant or variable over time: potential and fluxes at the upper boundary, potential, fluxes and hydraulic gradient at the bottom boundary. Initial conditions may be given as pressure heads, which can be either constant along the soil profile or variable node by node. They also allow to use both unimodal and bimodal hydraulic properties as input for different soil layers.

An important difference lies in how the models deals with precipitation and evaporation when they occur simultaneously. HYDRUS 1D subtracts evaporation from precipitation and applies net precipitation to the soil. FLOWS applies precipitation as it is and sets evaporation to zero on rain input times, while still allowing transpiration. This approach is the same adopted in the VS2DTI (Healy, 1990). Of course, these different approaches may have different impacts depending on the time resolution of simulations, having for example less effects in the case of hourly or shorter precipitation inputs than of daily input data.

The following figures provide a comparison between FLOWS and HYDRUS 1D results for different top and bottom boundary conditions. In all the graphs, the following labels have been used:

- **TB =** Top Boundary; **BB =** Bottom Boundary**; IC =** Initial Condition
- **q=** flux (positive upward); **h$_{\mathbf{in}}$ =** initial pressure head;
- **Cin = initial concentration**
- **HP =** Hydraulic Properties;
- **t$_{\mathbf{max}}$ =** maximum simulation time
- λ **=** dispersivity

In all cases, subscript ***w*** is for water, subscript ***s*** is for solutes**.**

TB$_w$ $q_{in=}$= 0.1 cm/h; **BB$_w$** = free drainage; **IC$_w$** h_{in} = -100 cm

Two layers (0 – 30; 30-100 cm)

HP van Genuchten - Mualem

t$_{max}$ = 30 days

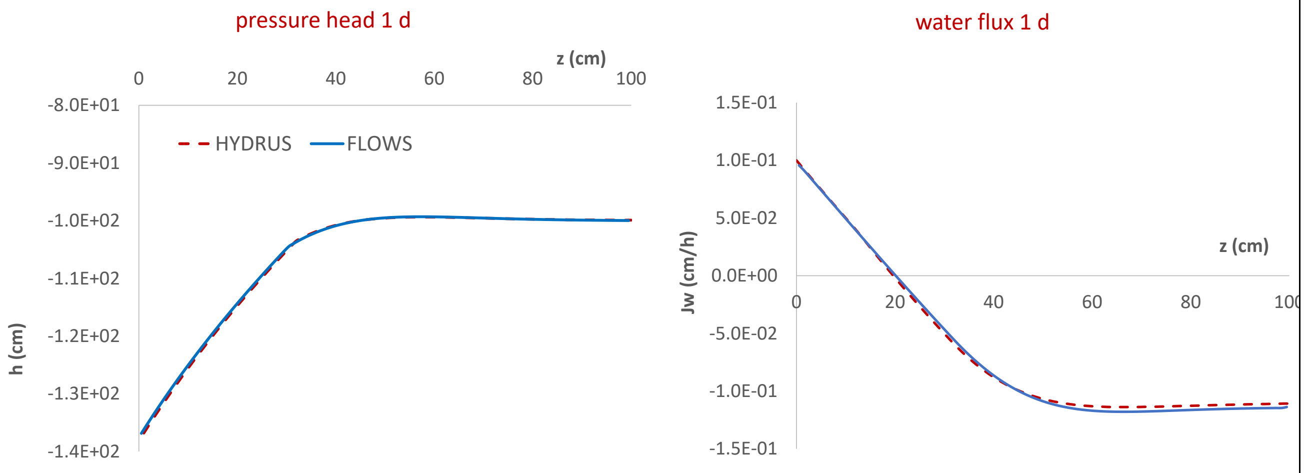


Figure 12 Pressure head and water flux profiles at t = 1 day. Blu solid line = FLOWS simulations; Red dashed line = HYDRUS simulations

TB$_w$ $q_{in=}$= -1.0 cm/h; **BB$_w$** = free drainage; **IC$_w$** h_{in} = -100 cm

TB$_s$ q_s = 0.04 g/cm3; Pulse duration 1 h; **BB$_s$** = $\Delta C/\Delta z$ = 0; **IC$_s$** Cin = 0 g/cm3

Two layers (0 – 30; 30-100 cm)

HP van Genuchten - Mualem

λ = 1 cm

t$_{max}$ = 30 days

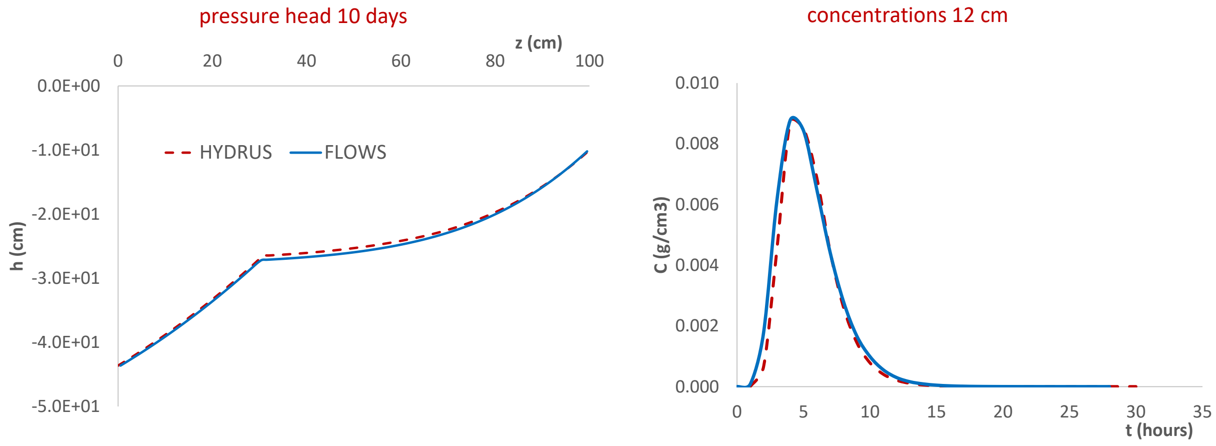


Figure 13 Pressure head profile at t = 10 days and concentrations over time at depth z = 12. Blu solid line = FLOWS simulations; Red dashed line = HYDRUS simulations

TB$_w$ $q_{in=}$= variable; **BB**$_w$ = free drainage; **IC**$_w$ h_{in} = -100 cm

Two layers (0 – 30; 30-100 cm)

HP van Genuchten - Mualem

λ = 1 cm

t_{max} = 30 days

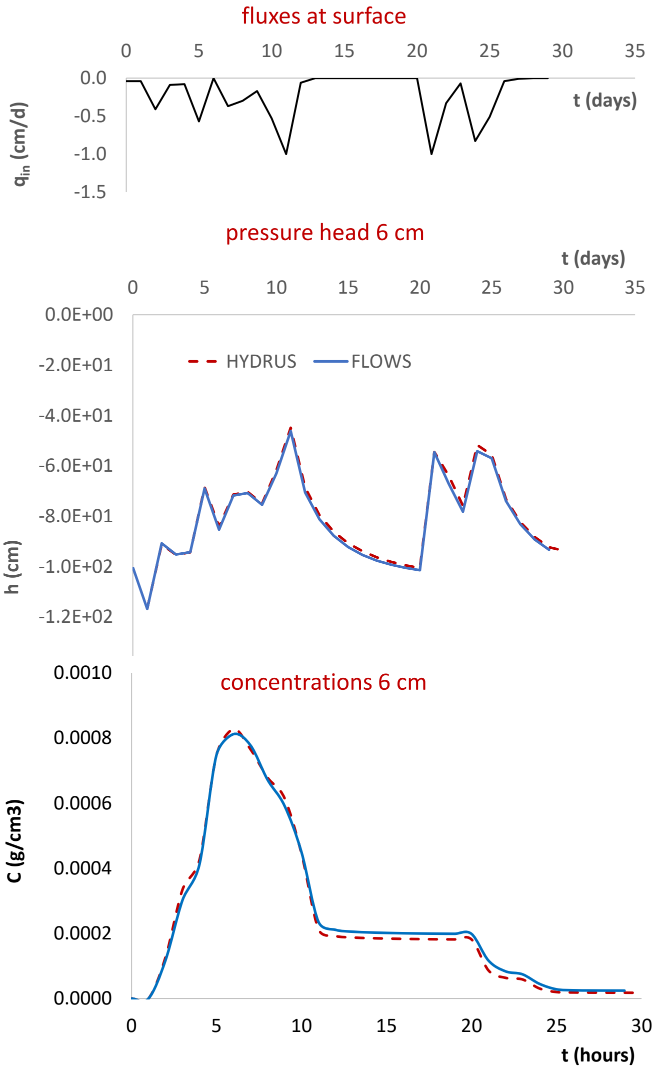


Figure 14 Pressure head and concentrations over time at depth z = 6 cm. Blu solid line = FLOWS simulations; Red dashed line = HYDRUS simulations. The graph on the top shows the fluxes at the **TB** (negative downward)

TB$_w$ $q_{in=}$= variable; **BB$_w$** = free drainage; **IC$_w$** h_{in} = -100 cm

Two layers (0 – 30; 30-100 cm)

HP van Genuchten - Mualem

λ = 1 cm

t$_{max}$ = 30 days

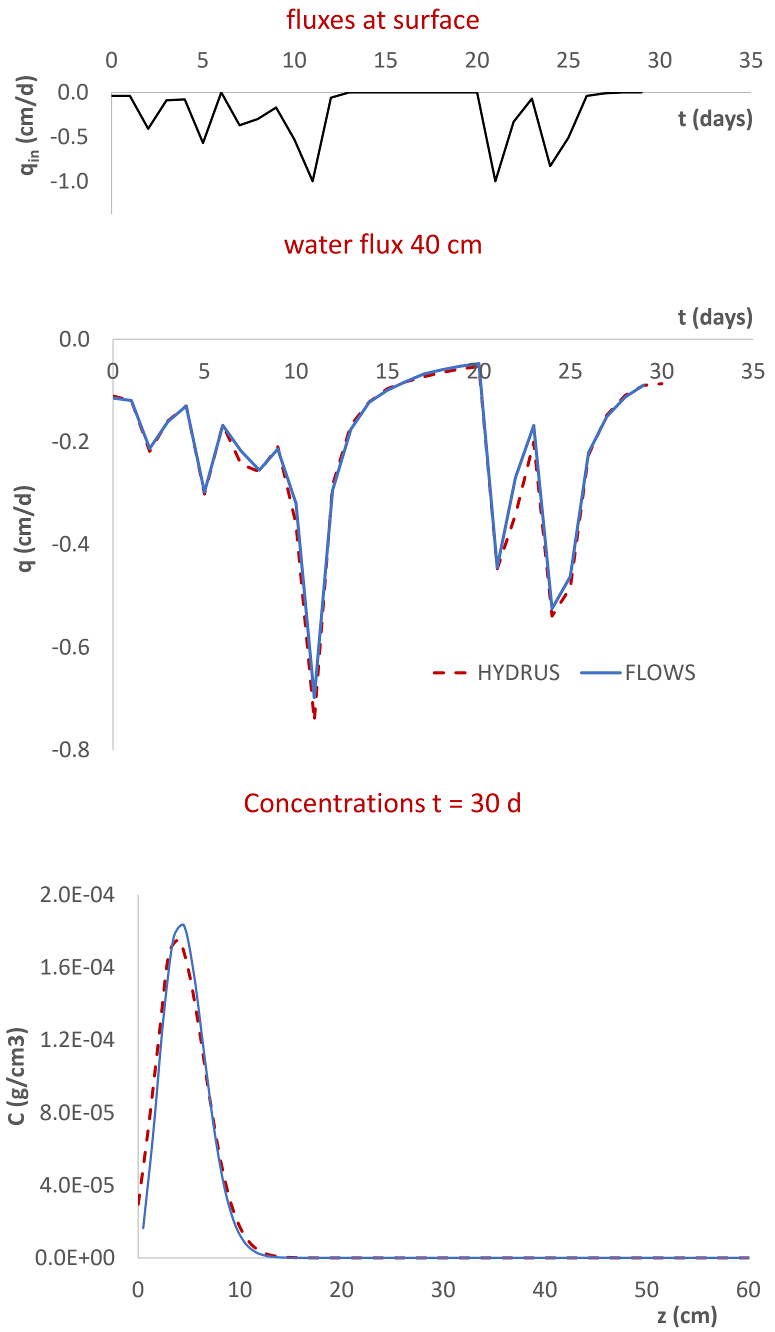


Figure 15 Water fluxes over time at depth z = 40 cm and concentrations profile at t = 30 days. Blu solid line = FLOWS simulations; Red dashed line = HYDRUS simulations. The graph on the top shows the fluxes at the **TB** (negative downward)

TB$_w$ q$_{in=}$= variable; **BB**$_w$ = free drainage; **IC**$_w$ h$_{in}$ = -100 cm

Two layers (0 – 30; 30-100 cm)

HP van Genuchten - Mualem

t$_{\mathbf{max}}$ = 214 days

λ = 1 cm

Fertilizers:

Manure 40000 kg; t = 10d

Urea 200 kg; t = 50d

NO3 100 kg; t = 90 d

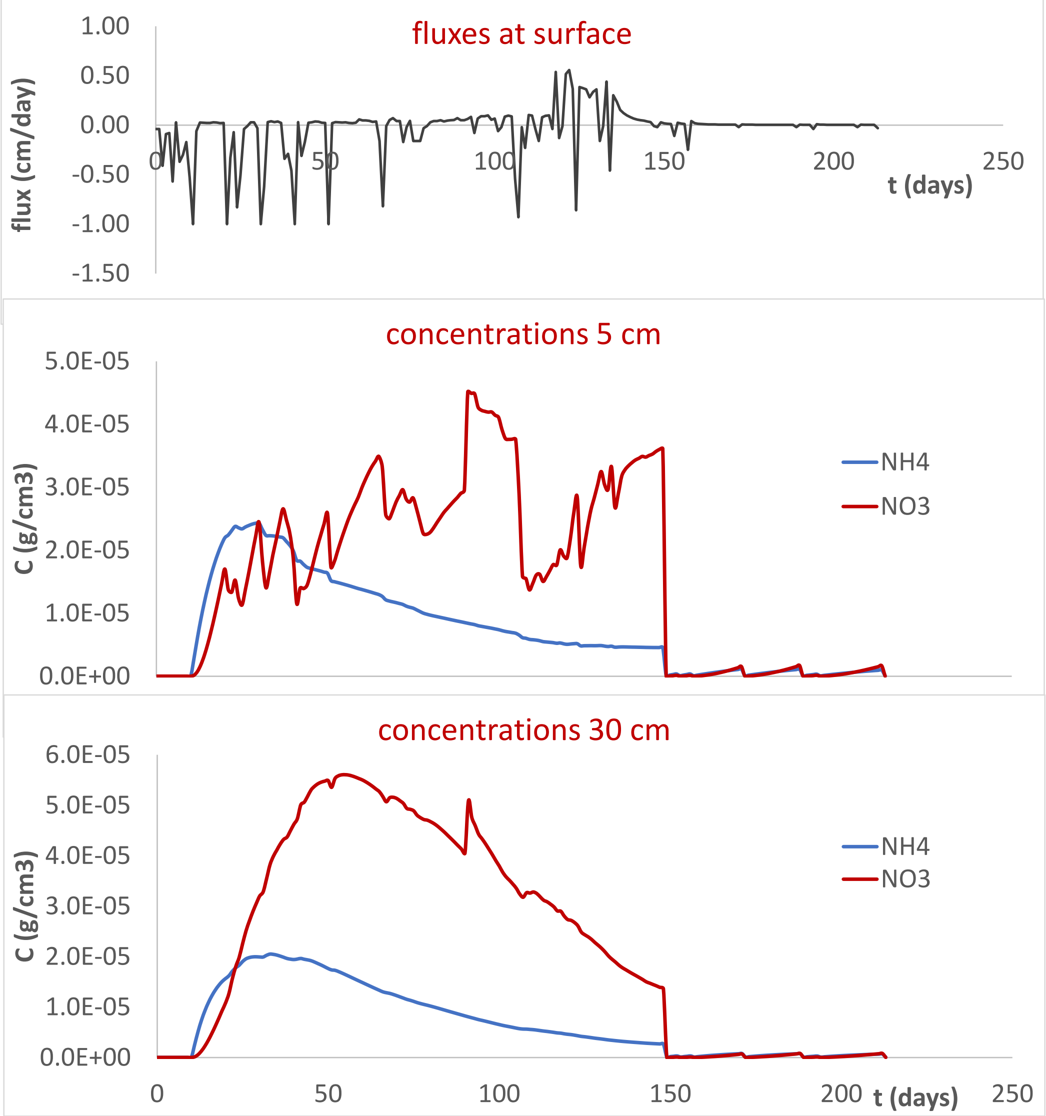


Figure 16 Concentrations of NH_4^+ and NO_3^- over time at depths z = 5 cm and z = 30 cm. The graph on the top shows the fluxes at the **TB** (negative downward)

6. Multisite and Montecarlo simulations

FLOWS allows for simulations in multiple profiles (multi-site simulations). For each simulation profile, the model produces a set of outputs and store them in a specific folder. At the end of the multiple simulations, the model calculates two additional folders, respectively with the average and with the variance of the values of each output variable. The model also produces an output with a list of the failed simulations (with the corresponding failure flag related to convergence problems). The multi-site configuration may be especially useful in the following cases:

1. Simulations to be carried out on several soil profiles in a large area characterized by a spatial variability of the soil hydraulic properties. This may be required, for example, to calculate distributed outputs, such as (among many others): i. irrigation needs at district scale; ii. deep percolation fluxes of water and solutes to analyse groundwater recharge and/or groundwater vulnerability; iii….
2. Uncertainty Analyses and/or Sensitivity Analysis requiring simulation to be carried several times with different soil hydraulic parameters coming from probability density functions for each of them;

The last issue may be dealt with in a stochastic framework by Montecarlo techniques, and the average and variance of specified outputs may be used to analyse the relative importance of each input parameters on the model output. In this specific case (Montecarlo configuration), the user is allowed to select, for each soil layer in the profile, the soil hydraulic and dispersive parameters to be considered as stochastic and provide the mean for each of them, as well as their covariance matrix, allowing to generate random vectors of correlated parameters (see, for example, Carsel and Parrish, 1988; Coppola et al., 2009).

6.1. Montecarlo simulations and Model Predictions Uncertainty

A model, even very sophisticated, may only provide a simplified representation of the actual processes involved in the water flow and solute transport being studied. A strictly deterministic model output is generally limited because of several, interacting factors: 1. Parametric uncertainty is unavoidable, as the characterisation of soil hydraulic and transport properties is not a simple task; 2. Even when accurate soil hydrological characterizations have been carried out, the natural heterogeneity of the soil properties represents a source of uncertainty; 3. The model dependence itself on events that are somehow stochastic, such as rainfall and radiation (evapotranspiration) makes a deterministic response impossible. The most correct way to use a model is at least to provide model predictions with an uncertainty band of the output.

To be of real utility in processes interpretation and decision making, thus, a model requires the generation of quantitative measures of confidence in the model predictions.

Accordingly, FLOWS includes a tool based on the Montecarlo method, allowing for the propagation of inherent uncertainty in model parameters into uncertainty in the model predictions of selected model outputs (Coppola et al., 2009).

Montecarlo methods are those in which properties of the distributions of random variables are investigated by the use of simulated random numbers. Montecarlo simulations are especially important for evaluating the prediction uncertainties of the models and represent a valuable tool in identifying the overall model sensitivity to selected model input parameters. This widely used approach is conceptually simple and is based on the idea of approximating stochastic processes by a large number of equally probable realizations. *N* successive sets of realizations are created by randomly drawing one value from the assumed distribution of each of the *M* input parameters ($M=p_i$ for i=1,2,…, *M*) or from a joint multivariate density function for correlated parameters. A deterministic response vector *Y* ($Y=f(M)$) is calculated by running the deterministic model *N* times such that the statistical moments of the desired output dependent variables can be estimated. As an example, by solving *N* times the Richards equation with proper boundary and initial conditions, one can obtain the mean and variance of the water content due to the fluctuations of the p^{th} random parameter according to its distribution:

$$\langle\theta_p(z,t)\rangle = \frac{1}{N}\sum_{j=1}^{N}\theta_p(z,t)$$

$$\sigma_p^2 = \frac{1}{N-1}\sum_{j=1}^{N}\left[\theta_p(z,t) - \langle\theta_p(z,t)\rangle\right] \qquad (127)$$

where ⟨⟩ denotes ensemble averaging.

FLOWS provides mean and variance for all of the model outputs.

To do that, Montecarlo numerical simulation methods require probability density functions of model input parameters. For correlated parameters the parameter covariance matrix is also necessary in order to develop a joint probability distribution (JPD) forming the basis for building the *N* random vectors of correlated parameters. As pointed out by Smith and Diekkruger (1996), using the statistical moments of the parameters of the hydraulic functions for generating the random field to be used in Montecarlo simulations implies the assumption that soil hydraulic variability can be described by the statistical distribution of such parameters.

APPENDIX A

USER INSTRUCTION

A.1. FLOWS installation

FLOWS works under Matlab environment.

These are the steps the user has to follow for setting up a simulation project:

1) Firstly, the user has to upload the folder *FLOWS_USER* on the PC.

The main folder contains two other folders: 1) **codes**; 2)**demo**.

The **codes** folder contains the main code (*compute*) and several other codes mostly devoted to the calculation of the function called by the *compute* code. The codes are only accessible to the administrator.

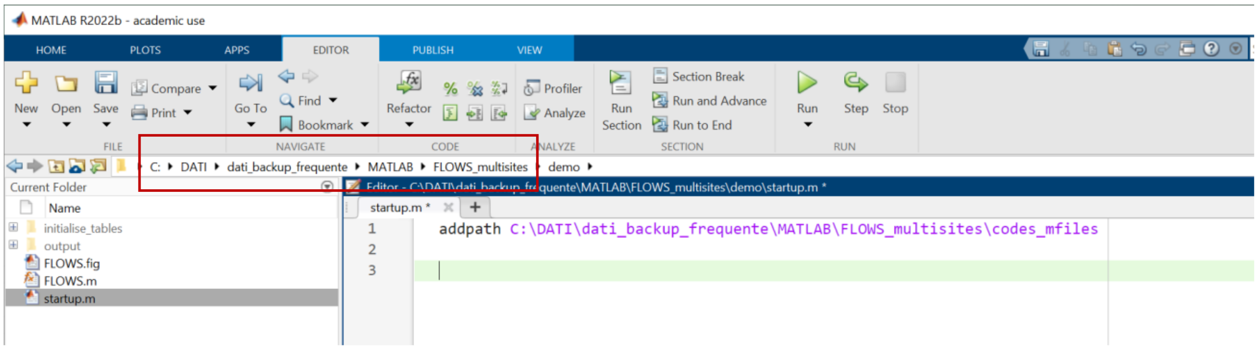
The **demo** folder contains a file called ***FLOWS.fig*** (which settings are included in the file *FLOWS.m*) allowing the user for opening the user interface, where the user may set the different simulation options, parameters and data.

The demo folder also includes a file called *startup.m*, containing the path for the input and output files and a folder called.

Finally, the **demo** folder contains two folders:

- the *output* folder, which, in turn, includes .txt and .mat files (see the section 6.4. Model output. The .txt files allows the user drawing graphs and doing calculations. The .mat files are used internally by the code to build the graphs of selected variables (namely, pressure heads, water contents, concentrations and water fluxes) for selected depths and times;
- the *initialise_tables* folder, including .xls files called by the interface to build the input tables for the time and node conditions, for the profile settings and the nitrates settings. The folder includes the following files:
- *time; nodes.xls*: They are files used by the model to initialise the **time settings** and **node settings** tables (see the section below, 6.3. Model input);
- *profile.xls*: This file is used by the model to initialise the profile settings table;
- *profile_multisites.xls*: This file is used by the model to initialise the profile settings table in the case of multisite configuration (see 6.3. Model input);
- *nitrates.xls*: This file is used by the model to initialise the inputs of nitrate and phosphorus fertilizers table.

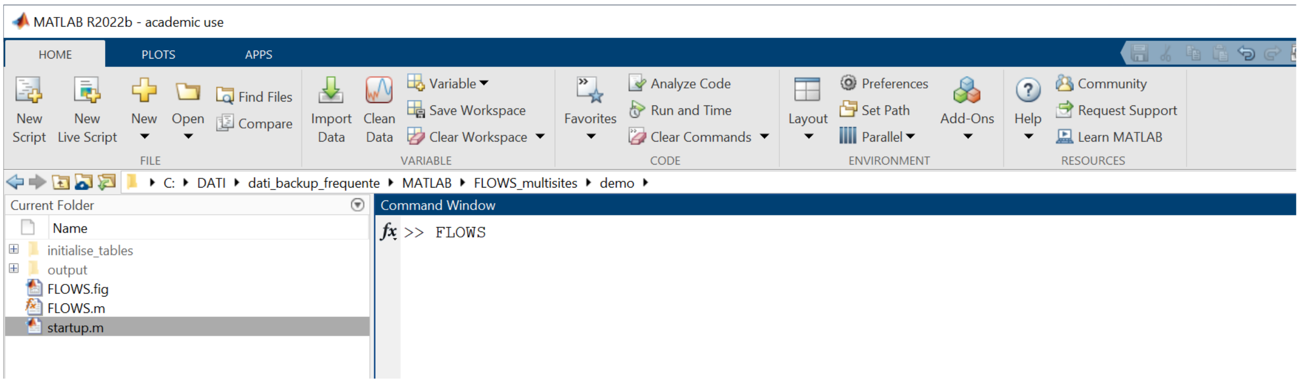
2) Then, the user copies the folder *demo* and rename the new folder, for example, to PROJECT_TEST (or any other name given by the user). Then, the user opens the *startup* file in the PROJECT_TEST and changes the path line according to the path reported in the Matlab (that in the red rectangle in the figure I below);



```
addpath C:\DATI\dati_backup_frequente\MATLAB\FLOWS_multisites\codes_mfiles
```

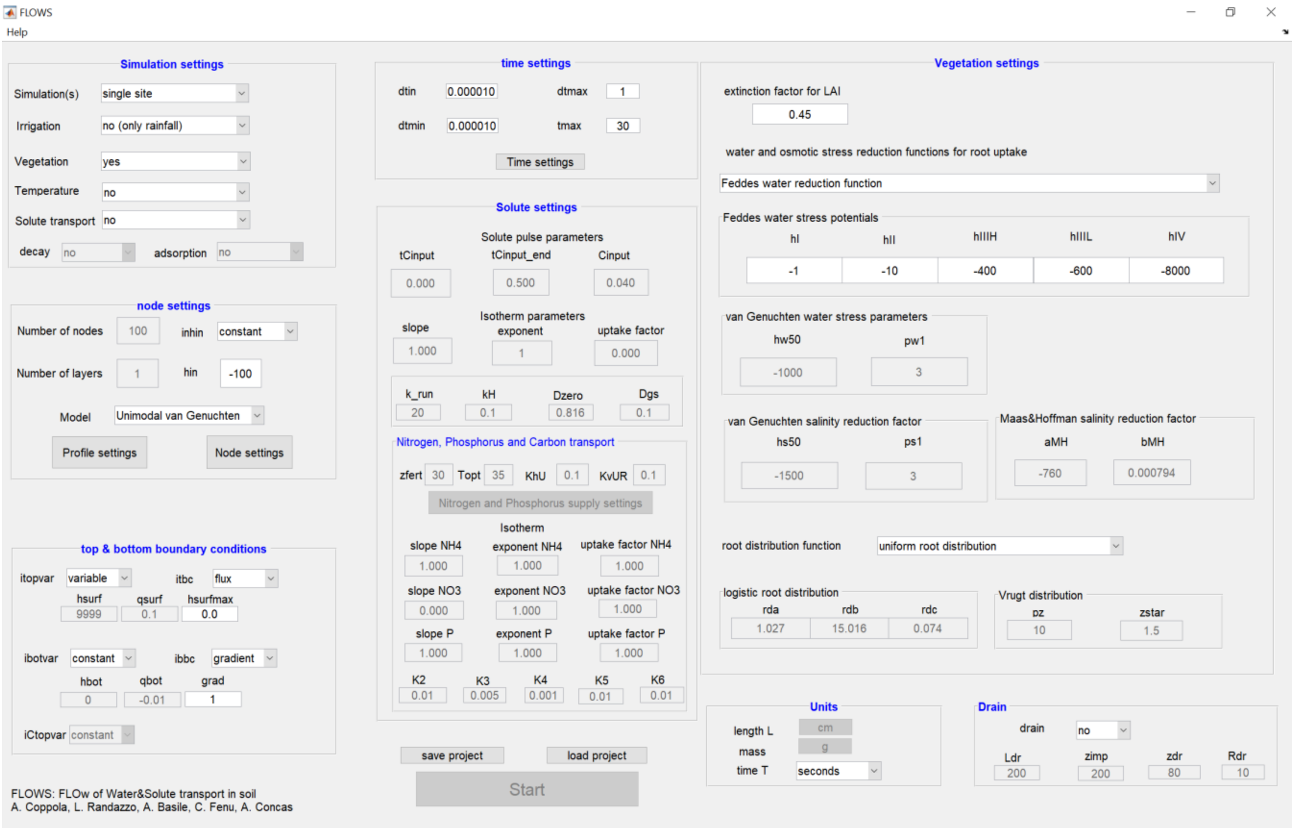
Figure i. The user must change the path line in the *startup* file with that indicated in the red rectangle

3) After changing the *startup* file in the new folder PROJECT_TEST, the user may open the **FLOWS** interface by writing down FLOWS in the Matlab command window, just as in the figure ii. This way, the user enters the **FLOWS** *user interface* (see the figure iii).



```
>> FLOWS
```

Figure ii. Opening the **FLOWS** *user interface*



Simulation settings
Simulation(s): single site
Irrigation: no (only rainfall)
Vegetation: yes
Temperature: no
Solute transport: no
decay: no adsorption: no

node settings
Number of nodes: 100 inhin: constant
Number of layers: 1 hin: -100
Model: Unimodal van Genuchten

top & bottom boundary conditions
itopvar: variable itbc: flux
hsurf 9999 qsurf 0.1 hsurfmax 0.0
ibotvar: constant ibbc: gradient
hbot 0 qbot -0.01 grad 1
iCtopvar: constant

time settings
dtin 0.000010 dtmax 1
dtmin 0.000010 tmax 30

Solute settings
Solute pulse parameters: tCinput 0.000 tCinput_end 0.500 Cinput 0.040
Isotherm parameters: slope 1.000 exponent 1 uptake factor 0.000
k_run 20 kH 0.1 Dzero 0.816 Dgs 0.1

Nitrogen, Phosphorus and Carbon transport
zfert 30 Topt 35 KhU 0.1 KvUR 0.1
Isotherm
slope NH4 1.000 exponent NH4 1.000 uptake factor NH4 1.000
slope NO3 0.000 exponent NO3 1.000 uptake factor NO3 1.000
slope P 1.000 exponent P 1.000 uptake factor P 1.000
K2 0.01 K3 0.005 K4 0.001 K5 0.01 K6 0.01

Vegetation settings
extinction factor for LAI: 0.45
water and osmotic stress reduction functions for root uptake: Feddes water reduction function
Feddes water stress potentials: hI -1 hII -10 hIIIH -400 hIIIL -600 hIV -8000
van Genuchten water stress parameters: hw50 -1000 pw1 3
van Genuchten salinity reduction factor: hs50 -1500 ps1 3
Maas&Hoffman salinity reduction factor: aMH -760 bMH 0.000794
root distribution function: uniform root distribution
logistic root distribution: rda 1.027 rdb 15.016 rdc 0.074
Vrugt distribution: pz 10 zstar 1.5

Units
length L cm
mass g
time T seconds

Drain
drain no
Ldr 200 zimp 200 zdr 80 Rdr 10

FLOWS: FLOw of Water&Solute transport in soil
A. Coppola, L. Randazzo, A. Basile, C. Fenu, A. Concas

Figure iii. Overall view of the **FLOWS** interface

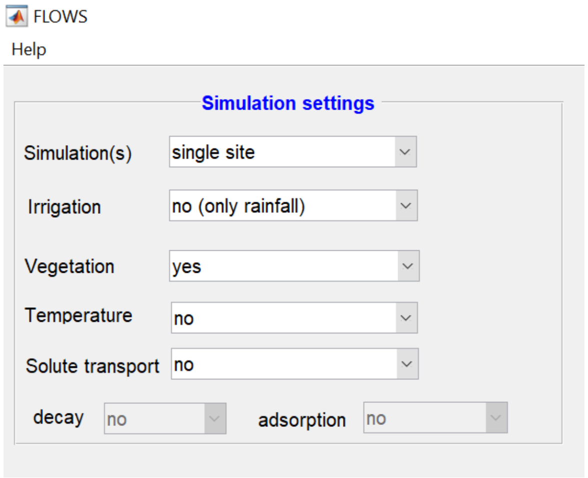
A.2. Instructions for FLOWS graphical interface users

The interface contains seven blocks (plus a small block to set the Units):

1. Simulation settings;
2. Node settings;
3. Top and bottom boundary conditions;
4. Time settings;
5. Solute settings (including a sub-block for Nitrogen Transport);
6. Vegetation settings;
7. Drainage settings.

A detailed description and explanation of the settings of each of the six blocks is given below.

1. Simulation settings block



FLOWS

Help

Simulation settings

Simulation(s): single site

Irrigation: no (only rainfall)

Vegetation: yes

Temperature: no

Solute transport: no

decay: no

adsorption: no

Figure iv. Simulation settings block

This block contains four different menus:

Simulation(s) menu

The simulation menu allows to opt for single or multiple simulations:

Single site = allows for a single simulation and the parameters are read from the profile settings;

Multisites = allows for multiple simulations for multiple sites and the parameters are read from a profile settings excel table reporting the parameters for all sites;

Montecarlo = allows for multiple simulations, with multiple parameter vectors randomly generated starting from the statistical distribution of the parameters. In this case, the user must input the mean and variance of each of the parameters considered to be stochastic, along with their covariance matrix in the case of correlated parameters. This module will be released soon in a new version of the code.

Irrigation menu

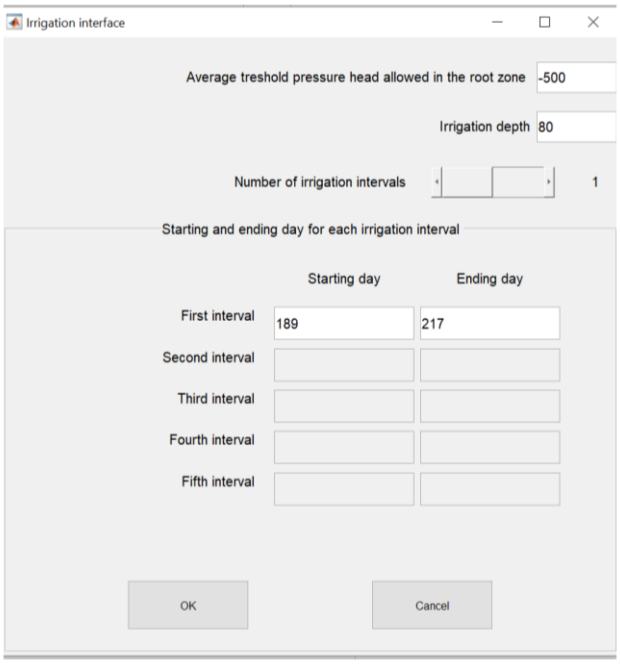
This menu contains three options:

no (only rainfall) = no irrigation. Rainfall is the only inflow allowed;

yes (computed by the model) = Irrigation fluxes are calculated by the model according to the criterium described in the section *2.6 Irrigation;*

yes (given by the user) = Irrigation fluxes are calculated by the irrigation volumes actually supplied by the farmer. With this option, a specific column, labelled as *TBirr* (*top boundary irrigation*), will automatically show up in the *time conditions* file (see the *Time settings* block).

In the case the *yes (computed by the model)* option has been selected, the following window will show up:



Irrigation interface

Average treshold pressure head allowed in the root zone: -500

Irrigation depth: 80

Number of irrigation intervals: 1

Starting and ending day for each irrigation interval

	Starting day	Ending day
First interval	189	217
Second interval		
Third interval		
Fourth interval		
Fifth interval		

Figure v. Window for irrigation settings

This window allows the user to set:

- the *threshold pressure* (the critical pressure h_{crit}) below which the "model irrigates";
- the *irrigation depth*, that is the depth over which the h_{av}, to be compared to h_{crit}, is computed by the model. The *irrigation depth* may be set at any value even if a reasonable value should be the maximum root depth D_r;
- the *irrigation intervals*. Up to five periods may be selected, by shifting the *Number of irrigation intervals* bar and entering the initial and final time for each interval. This is to allow the user to account for specific irrigation habits of farmers, who, for example, are used to stress the crop in some of the growth stages to improve organoleptic properties of the crop. By leaving the model "to irrigate" anytime $h_{av} < h_{crit}$, would not allow for considering the specific farmer behaviour.

Vegetation menu

This menu considers two options to account or not for the presence of vegetation:

no = the evapotranspiration is not simulated. The upper boundary condition has to be imposed by the user either in terms of flux or pressure head. This is the case for example of simulations of laboratory experiments carried out under controlled top boundary conditions;

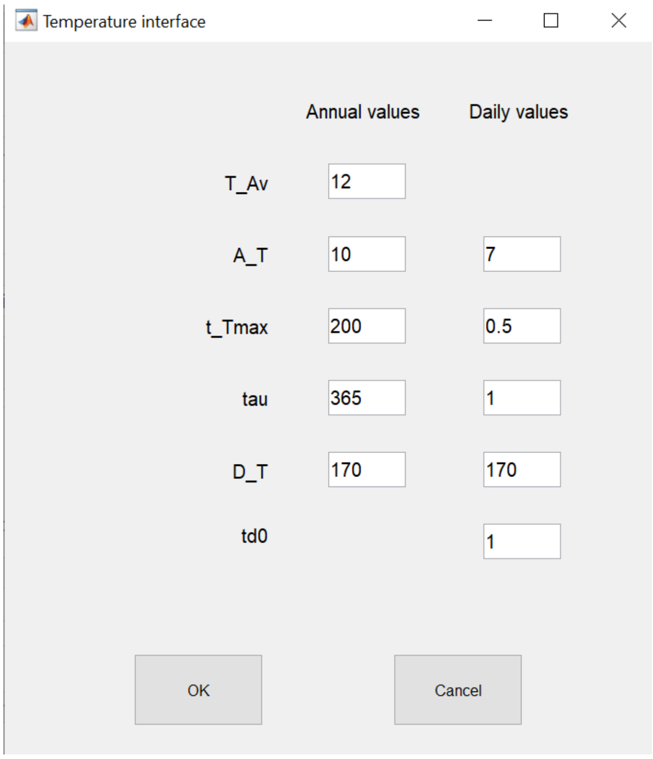
yes = the presence of vegetation is simulated according to the Vegetation settings (see later). The case of the bare soil is a special case of this option. In this case, the *LAI* and D_r columns in the time conditions have to be set to zero and the K_c has to be set at the bare soil value to convert the ET_r (the reference evapotranspiration) to the potential evaporation, E_p, from a bare soil.

Temperature menu

This menu considers two options to simulate or not the temperature:

no = the temperature is not simulated;

yes = the temperature is simulated according to the combined approach provided by van Wijk and De Vries (1963), based on the superposition of the annual and daily sinusoidal fluctuation around a constant value of the soil. In this case a Temperature settings window will show up and the user will be asked to set the following temperature simulation parameters:



	Annual values	Daily values
T_Av	12	
A_T	10	7
t_Tmax	200	0.5
tau	365	1
D_T	170	170
td0		1

Figure vi. The *Temperature settings* block

where the user is required to set the parameters:

- the *average annual temperature*, $T_{av,y}$ (°C);
- the daily and annual amplitudes, $A_{T,d}$ and $A_{T,y}$ (°C) of the sine wave oscillating around the average daily, $T_{av,y}$, and annual temperature;
- the daily, $t_{Tmax,d}$, and annual $t_{Tmax,y}$, time (days) when the temperature reaches the maximum.
- The diurnal, $\tau=1$, and annual, $\tau=365$, wave period;
- The soil thermal diffusivity, D_T, ($cm^2 day^{-1}$), that is, the ratio of the soil thermal conductivity to the soil volumetric heat capacity for the day ($D_{T,d}$) or the year ($D_{T,y}$);
- The day of the year, t_{d0}, corresponding to the time t = 0 of the simulation.

In the case of either nitrogen transformation and transport simulation (*solute transport = yes, ADE – nitrogen*) or carbon, nitrogen and phosphorus transformation and transport simulation (*solute transport = yes ADE C-N-P*) the temperature is set to *yes* and the temperature setting window automatically appears. Also, the time units are automatically set to *days*.

***Solute transport* menu**

This menu considers two options to account or not for the solute transport:

no = solute transport is not simulated;

yes - *ADE* = solute transport is simulated by the Advection-Dispersion equation, according to the *Solute Transport settings* (see below);

yes – *ADE nitrogen* = this option allows for only nitrogen transformation and transport simulations by solving the Advection-Dispersion equation twice, once for N-NH4 and once for N-NO3, with appropriate exchange and source-sink terms, according to the procedure described in the section **4.3.2. Simulating only nitrogen transport**;

Yes – ADE C-N-P = this option allows for carbon, nitrogen and phosphorus transformation and transport, according to the transformation-transport chain described in the section **4.3.1. Simulating Organic Matter decomposition and Carbon, Nitrogen and Phosphorus transport**

In the case the *yes* - *ADE* is selected, the two menu frames named *decay* and *adsorption* show up and further options become available:

Decay:

- *no* = solute decay is not simulated*;*
- *yes* = in the case of *single site* simulation, a specific column will be added to the node conditions input (see the section on the node settings below) allowing the user to input a decay coefficient (one for each of the 100 nodes)*.* In the case of *multiple sites* simulations, the user will be asked to input a decay value for each of the soil layers in each of the sites to be simulated. In this case, a columns for decay will be added to the profile settings table reporting the parameters for all sites.

Adsorption:

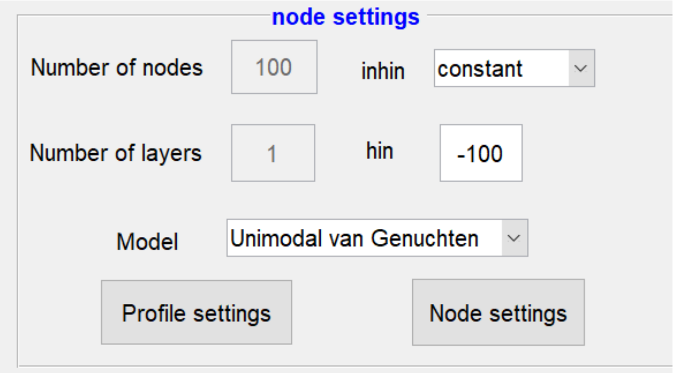
- *no* = adsorption is not simulated;
- *yes – linear* = adsorption is simulated by assuming a linear adsorption isotherm;
- *yes – Freundlich* = adsorption is simulated by assuming a Freundlich adsorption isotherm.

The menu on the adsorption is related to the two frames in the Solute settings block (*slope* and *exponent* in the Solute settings block). They refer to the slope and the exponent of the equation 61, C_{ad}=$K_F C^p$. In the *yes - linear* case, the code requires to input the slope (the distribution coefficient) of the isotherm. The frame named *exponent* remains overshadowed. In the *yes - Freundlich* option, the exponent of the non-linear isotherm is also required, and the frame will show up.

In the case either the *yes – ADE nitrogen* or the *yes – ADE C-N-P* option is selected, the sub-block *Nitrogen, Phosphorus and Carbon Transport* in the **Solute settings** menu will show up while all the other solute parameters will remain overshadowed.

2. Node settings block

This block contains information about the flow field discretization and the initial conditions for simulation.
The block allows to define the soil profile in terms of number of layers and their corresponding hydraulic properties, as well as the initial conditions for water flow simulations.



node settings

Number of nodes 100

inhin constant

Number of layers 1

hin -100

Model Unimodal van Genuchten

Figure vii. The *node settings* block

The block includes constant option frames and two windows: *Profile settings* and *Node settings.*

The options to be set are:
number of nodes (nz) = it is set at 100 by default;
number of layers (nlay) = number of horizontal layers in the soil profile. The number of layers may only be changed from the *Profile settings* table in the same block;

inhin menu

It allows to set the index for the initial pressure head condition:
constant = pressure head set at the value *hin* in all the profile nodes;
variable = pressure heads read from the *node settings* table in the same block);
If *inhin* = constant, the *hin* frame is will show up
hin = pressure head value for the *inhin* = constant.

Model menu

This menu allows for selecting one of the following models for hydraulic properties:

- *Unimodal van Genuchten* = van Genuchten model for water retention and Mualem model for hydraulic conductivity;
- *Bimodal Durner* = Durner model for water retention and Mualem model for hydraulic conductivity;
- *Bimodal Ross & Smettem* = Ross and Smettem model for water retention and Mualem model for hydraulic conductivity
- *Unimodal Gardner & Russo* = Russo model for water retention and Gardner model for hydraulic conductivity

Profile settings: The table in the window allows for setting the number of layers and the parameters of the hydraulic properties model selected for each layer.

The profile setting table contains a number of lines equal to the number of layers and a number of columns variable according to the selected model. For each layer a line shows the following variables:

layer = the layer in the soil profile;

depth = depth of the layer (the value for the last layer corresponds to the maximum depth of the simulated flow field);

rho = bulk density;

tetas = saturated water content;

tetar = residual water content;

The other parameters may change according to water retention and hydraulic conductivity models

Unimodal van Genuchten water retention:

alfvg = α_{VG} of the unimodal water retention function;

en = n of the unimodal water retention function;

<u>Bimodal Durner water retention:</u>

fi = weight for macroporosity;

alfvg = α_{VG1} for macroporosity;

en = n_1 for macroporosity;

alfvg2 = α_{VG2} for matrix;

en2 = n_2 for matrix;

Ross and Smettem water retention

fi = weight for macroporosity;

alfrs = α_{RS} for macroporosity;

en2 = *n* for the matrix;

alfvg2 =α_{VG} for the matrix;

Unimodal Russo water retention:

alfgr = α_{GR} of the unimodal water retention function;

mu = μ of the unimodal water retention function;

Hydraulic conductivity models

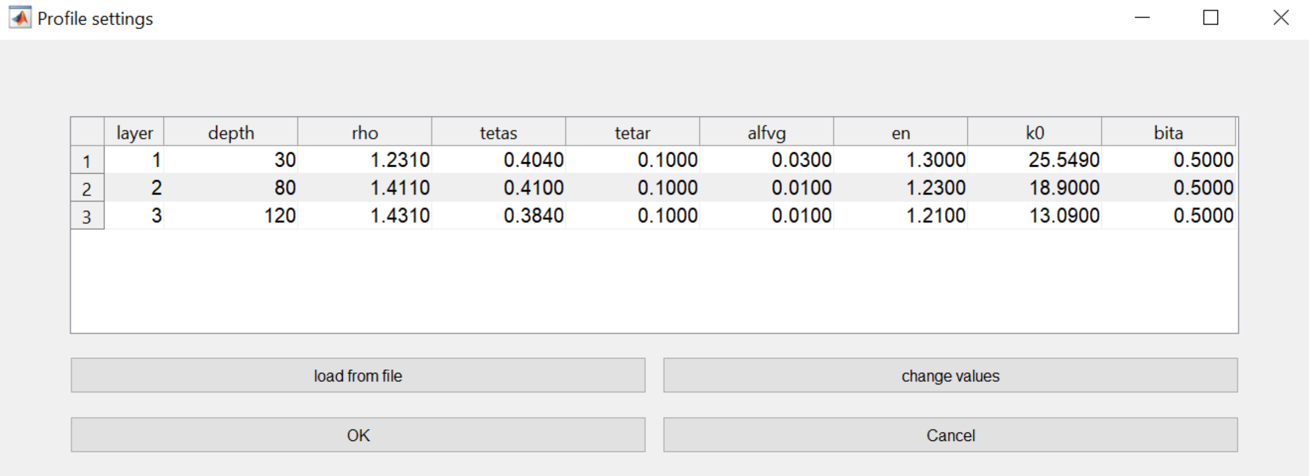
K0 = saturated hydraulic conductivity for the *unimodal van Genuchten, bimodal Durner, bimodal Ross and Smettem* and *unimodal Russo*

bita = exponent in the Mualem hydraulic conductivity model;

The *change values* button allows for changing the water retention and hydraulic conductivity parameter values.

It is possible to save a given parameter configuration by clicking OK and saving the file as *profile_settings* (or any other name decided by the user). The file will be saved in both .xls and .m format file.

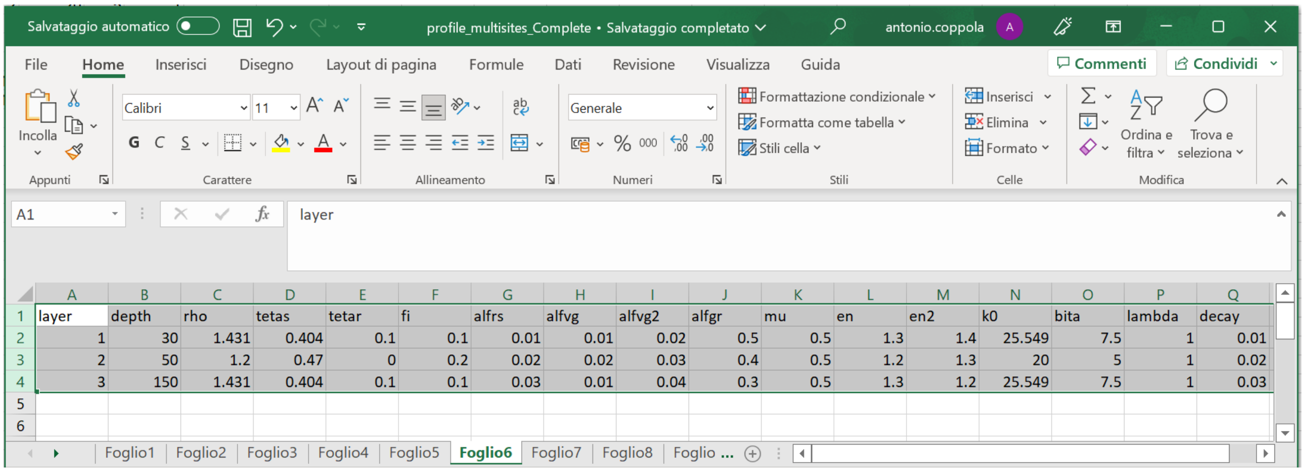
It is also possible to load a previously saved parameter configuration by clicking the *load from file* button. If in the **Simulations settings** the option *multisites* has been selected, along with the solute option *yes – ADE*, two columns will be added to the table, reporting the dispersivity λ and the decay μ values (only if the option *decay – yes* has been chosen) for each of the layers in each simulation site.



	layer	depth	rho	tetas	tetar	alfvg	en	k0	bita
1	1	30	1.2310	0.4040	0.1000	0.0300	1.3000	25.5490	0.5000
2	2	80	1.4110	0.4100	0.1000	0.0100	1.2300	18.9000	0.5000
3	3	120	1.4310	0.3840	0.1000	0.0100	1.2100	13.0900	0.5000

Figure viii. The *profile settings* window

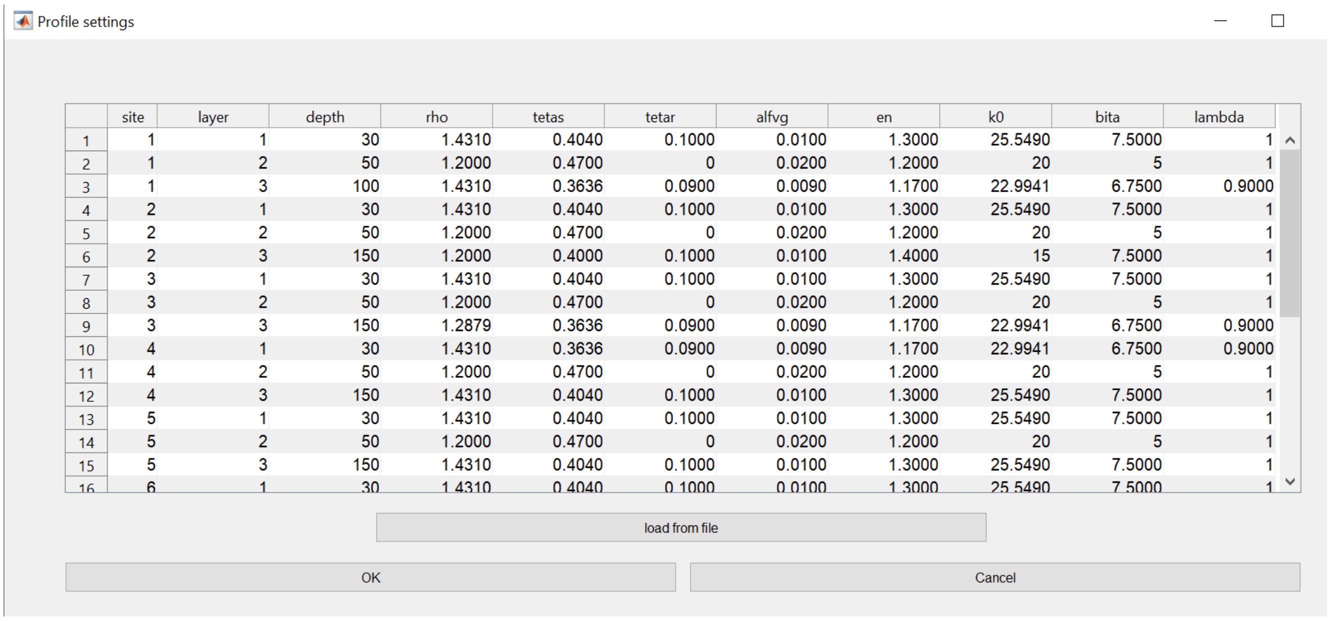
In the case of *multi-sites* simulations, when opening the *profile settings* window, the user will be asked for selecting the number of sites to be simulated. Thus, the model will query the file *profile_multisite.xls* in the folder *initialise_tables*. This file includes several sheets (currently 150 by default), which may be increased by the user), each referring to a different site, with three layers with default values of all the possible parameters involved in the different water retention-hydraulic conductivity models considered by the model (see the following figure).



layer	depth	rho	tetas	tetar	fi	alfrs	alfvg	alfvg2	alfgr	mu	en	en2	k0	bita	lambda	decay
1	30	1.431	0.404	0.1	0.1	0.01	0.01	0.02	0.5	0.5	1.3	1.4	25.549	7.5	1	0.01
2	50	1.2	0.47	0	0.2	0.02	0.02	0.03	0.4	0.5	1.2	1.3	20	5	1	0.02
3	150	1.431	0.404	0.1	0.1	0.03	0.01	0.04	0.3	0.5	1.3	1.2	25.549	7.5	1	0.03

Figure ix. The *profile multisite .xls* file

Thus, the model builds a table (see an example in the next figure) like that already shown for the *single-site* profile settings, but now with an additional column (the first column in the table) indicating the sequence of sites considered in the multi-site configuration.



Profile settings

	site	layer	depth	rho	tetas	tetar	alfvg	en	k0	bita	lambda
1	1	1	30	1.4310	0.4040	0.1000	0.0100	1.3000	25.5490	7.5000	1
2	1	2	50	1.2000	0.4700	0	0.0200	1.2000	20	5	1
3	1	3	100	1.4310	0.3636	0.0900	0.0090	1.1700	22.9941	6.7500	0.9000
4	2	1	30	1.4310	0.4040	0.1000	0.0100	1.3000	25.5490	7.5000	1
5	2	2	50	1.2000	0.4700	0	0.0200	1.2000	20	5	1
6	2	3	150	1.2000	0.4000	0.1000	0.0100	1.4000	15	7.5000	1
7	3	1	30	1.4310	0.4040	0.1000	0.0100	1.3000	25.5490	7.5000	1
8	3	2	50	1.2000	0.4700	0	0.0200	1.2000	20	5	1
9	3	3	150	1.2879	0.3636	0.0900	0.0090	1.1700	22.9941	6.7500	0.9000
10	4	1	30	1.4310	0.3636	0.0900	0.0090	1.1700	22.9941	6.7500	0.9000
11	4	2	50	1.2000	0.4700	0	0.0200	1.2000	20	5	1
12	4	3	150	1.4310	0.4040	0.1000	0.0100	1.3000	25.5490	7.5000	1
13	5	1	30	1.4310	0.4040	0.1000	0.0100	1.3000	25.5490	7.5000	1
14	5	2	50	1.2000	0.4700	0	0.0200	1.2000	20	5	1
15	5	3	150	1.4310	0.4040	0.1000	0.0100	1.3000	25.5490	7.5000	1
16	6	1	30	1.4310	0.4040	0.1000	0.0100	1.3000	25.5490	7.5000	1

Figure x. The *profile multi-site settings* table

As in the *single-site* configuration, the model will upload in this table only the columns with the parameters actually involved in the water retention-hydraulic conductivity model selected by the user.

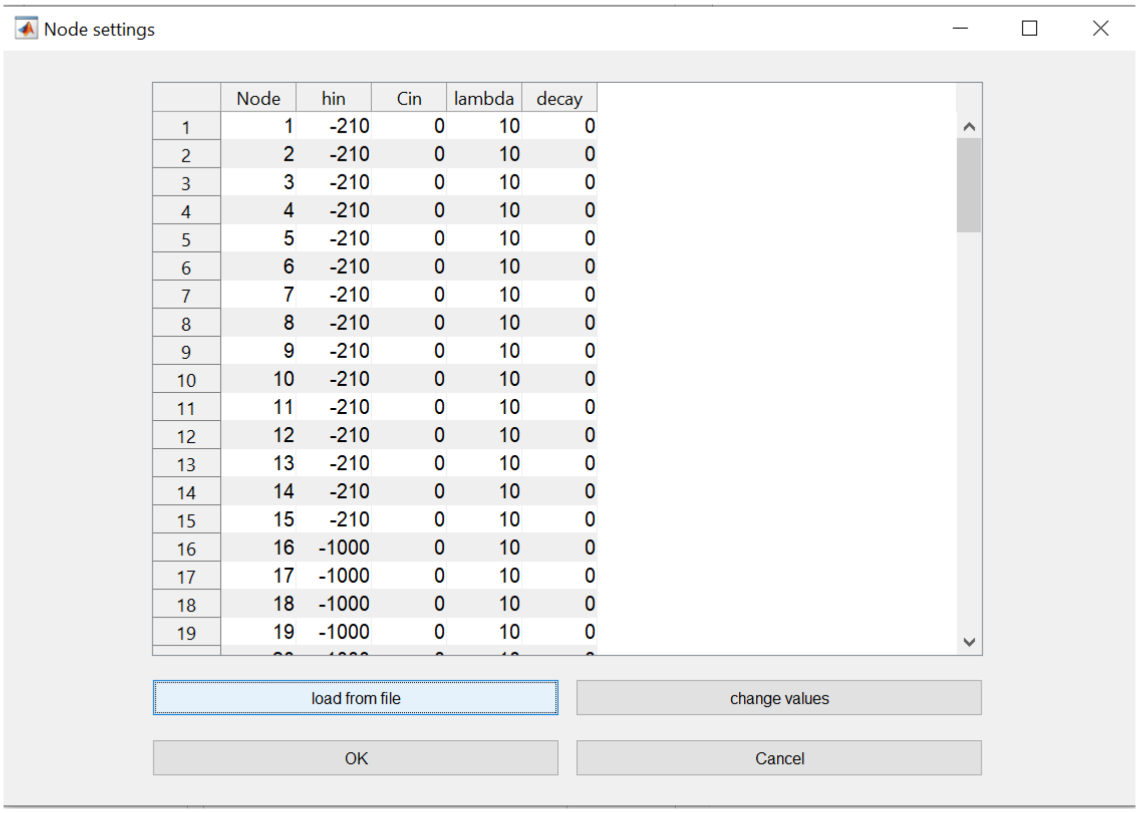
As already mentioned, in the case of solute transport, one or two additional columns are added, reporting respectively the value of the dispersivity, λ, and of the decay coefficient, μ (in the case of *decay = yes*), for each layer of each site.

Once clicking on the OK button, the user is asked to save the profile_settings file, and the model will save a *profile_??.xls* file (*??* are for the name given by the user) in the demo folder. This .xls file contains the same table shown in the figure x.

From this step on, the user may modify this .xls file by changing the parameter values, as well as the number of sites and the number of layers for each site. Then, by using the *load from file* button in the *profile settings* table, the user may upload this new .xls table and save it again by clicking the button OK.

In the case of *Montecarlo* configuration, the table *profile settings* is the same as the previous one, but is directly produced starting from the statistical distribution of the soil hydrological parameters.

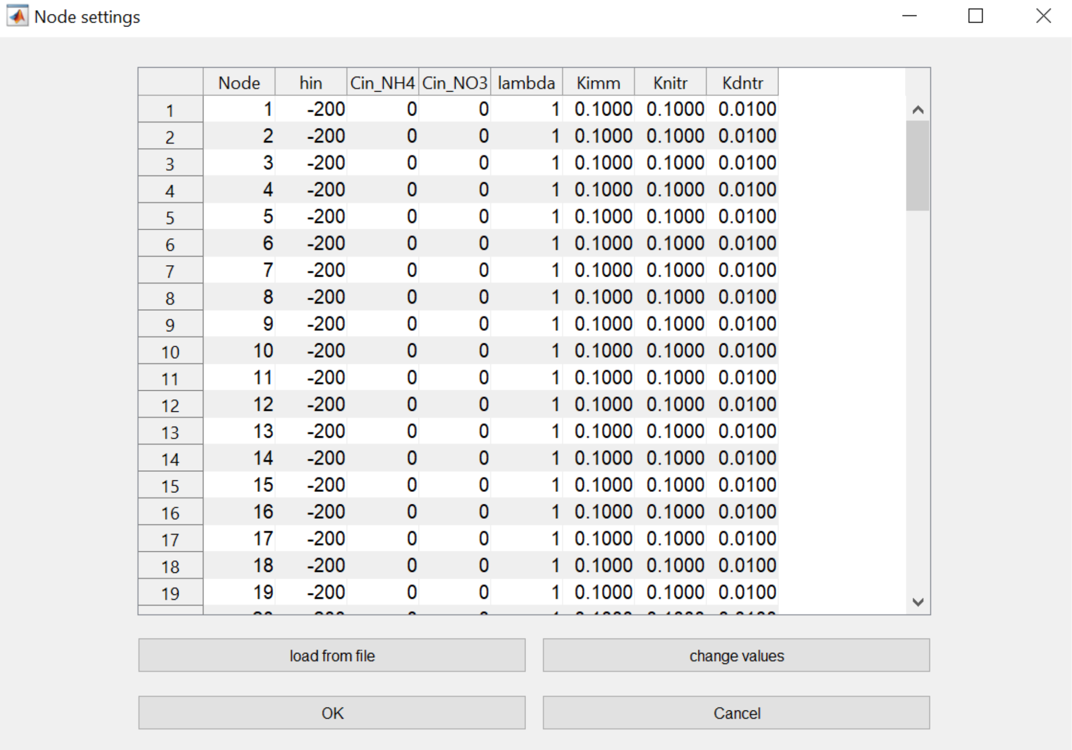
Node settings: The table in the window allows the user to upload the initial profile conditions. The *Node settings* table only appears when *inhin* = *variable* and/or the solute transport has to be simulated. The window contains a number of lines equal to the number of simulation nodes (100 by default) and up to five columns: 1) Node; and for each node 2) initial pressure head, *hin*; 3) initial concentration, *Cin*; 4) dispersivity, *lambda*; 5) decay coefficient, *decay*. Column only shows up when *inhin* = variable. Columns 3 to 5 only show up in the case of solute transport simulations and for the *single site* configuration. In the case of *multisite* configuration, the dispersivity and the decay coefficient are given in the last two columns of the profile settings, a value for each layer of each soil profile to be simulated.



	Node	hin	Cin	lambda	decay
1	1	-210	0	10	0
2	2	-210	0	10	0
3	3	-210	0	10	0
4	4	-210	0	10	0
5	5	-210	0	10	0
6	6	-210	0	10	0
7	7	-210	0	10	0
8	8	-210	0	10	0
9	9	-210	0	10	0
10	10	-210	0	10	0
11	11	-210	0	10	0
12	12	-210	0	10	0
13	13	-210	0	10	0
14	14	-210	0	10	0
15	15	-210	0	10	0
16	16	-1000	0	10	0
17	17	-1000	0	10	0
18	18	-1000	0	10	0
19	19	-1000	0	10	0

Figure xi. The node settings window

In the case of Nitrogen transport (*solute transport = yes – ADE nitrogen*) or Nitrogen, Phosphorus and Carbon transport (*solute transport = yes – ADE C-N-P*) the node settings will include two columns for the initial concentrations of NH4 and NO3 and three more columns with the constants, changing node by node, of immobilization (*Kimm*), of nitrification (*Knitr*) and of denitrification (*Kdntr*) (see next figure).



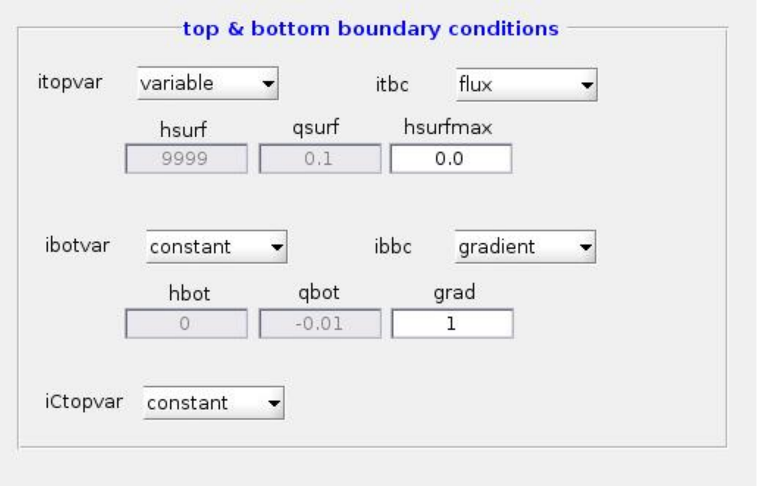
Node settings

	Node	hin	Cin_NH4	Cin_NO3	lambda	Kimm	Knitr	Kdntr
1	1	-200	0	0	1	0.1000	0.1000	0.0100
2	2	-200	0	0	1	0.1000	0.1000	0.0100
3	3	-200	0	0	1	0.1000	0.1000	0.0100
4	4	-200	0	0	1	0.1000	0.1000	0.0100
5	5	-200	0	0	1	0.1000	0.1000	0.0100
6	6	-200	0	0	1	0.1000	0.1000	0.0100
7	7	-200	0	0	1	0.1000	0.1000	0.0100
8	8	-200	0	0	1	0.1000	0.1000	0.0100
9	9	-200	0	0	1	0.1000	0.1000	0.0100
10	10	-200	0	0	1	0.1000	0.1000	0.0100
11	11	-200	0	0	1	0.1000	0.1000	0.0100
12	12	-200	0	0	1	0.1000	0.1000	0.0100
13	13	-200	0	0	1	0.1000	0.1000	0.0100
14	14	-200	0	0	1	0.1000	0.1000	0.0100
15	15	-200	0	0	1	0.1000	0.1000	0.0100
16	16	-200	0	0	1	0.1000	0.1000	0.0100
17	17	-200	0	0	1	0.1000	0.1000	0.0100
18	18	-200	0	0	1	0.1000	0.1000	0.0100
19	19	-200	0	0	1	0.1000	0.1000	0.0100

Figure xii. The node settings window in the case of nitrogen transport

3. Top and Bottom boundary conditions

This block includes five different menu. They allows for setting if the water and solute boundary (top, bottom) conditions are constant or variable, if the water boundary condition refesr to either a head (Dirichlet) or flux (Newman) condition. In the case of constant water boundary conditions, there are frames to set the corresponding value for the pressure head or flux.



top & bottom boundary conditions

itopvar: variable
itbc: flux
hsurf: 9999
qsurf: 0.1
hsurfmax: 0.0

ibotvar: constant
ibbc: gradient
hbot: 0
qbot: -0.01
grad: 1

iCtopvar: constant

Figure xiii. The *top and bottom boundary conditions* block

***itopvar* menu**

It refers to the top boundary conditions for water flow simulation:
constant = the top boundary condition is constant over the whole simulation time;
variable = the top boundary condition is variable over the simulation time;

***itbc* menu**

It refers to the top boundary conditions for water flow simulation:
flux = the top boundary condition (constant or variable) is a flux (Newman) condition. Positive and negative values are respectively for upward and downward fluxes;
potential = the top boundary condition (constant or variable) is a pressure head (Dirichlet) condition;
In the case of constant top boundary condition, the user has to set one of the following values, depending on the selected *itbc* option, in the corresponding box:
hsurf = the pressure head value imposed at the soil surface (top boundary);
qsurf = the flux value imposed at the soil surface (top boundary).
The frames with variables not involved in the simulation will be automatically overshadowed.
In the case of *itbc* = *flux*, the user has also to set the value in the following frame:
hsurfmax = maximum value of the pressure head at the soil surface in the case of ponding formation during the simulation. This value is used in the code for calculating the maximum flux (*fluxsurfmax*) at the soil surface. If *qsurf* exceeds the *fluxsurfmax*, the difference *qsurf* - *fluxsurfmax* will be considered runoff. When this condition occurs, the code switches the *itbc* condition from flux to pressure head at the *hsurfmax* value. In the case of *itopvar*=*constant*, this condition will be kept for the remaining simulation time; in the case of *itopvar*=*variable* this condition will be kept until the new *qsurf* condition becomes lower than the previous *qsurf* which has induced the runoff.

***ibotvar* menu**

It refers to the bottom boundary conditions for water flow simulation:
constant = the bottom boundary condition is constant over the whole simulation time;
variable = the bottom boundary condition is variable over the simulation time;

***ibbc* menu**

It refers to the bottom boundary conditions for water flow simulation:
flux = the bottom boundary condition (constant or variable) is a flux (Newman) condition. Positive and negative values are respectively for upward and downward fluxes;
potential = the bottom boundary condition (constant or variable) is a pressure head (Dirichlet) condition;

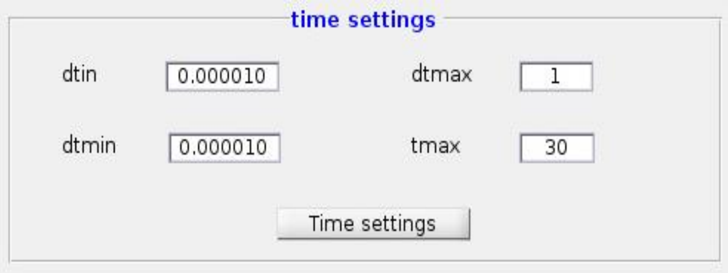
gradient = the bottom boundary condition is set at a constant hydraulic head gradient, *dH/dz=const*.;
seepage = the bottom boundary condition is set a zero flux until the bottom boundary will saturate. During the simulation run, the model verifies if the pressure head at the bottom becomes zero or higher than zero. In that case, the model switches to a zero-head boundary condition.

In the case of constant bottom boundary condition, the user has to set one of the following values, depending on the selected *ibbc* option, in the corresponding box:
hbot = the pressure head value imposed at the simulated profile bottom (bottom boundary);
qbot = the flux value imposed at the simulated profile bottom (bottom boundary).
When the gradient option has been selected, the following frame will automatically show up:
grad = the gradient value (generally 1, even if any gradient value may be set).
The frames with variables not involved in the simulation will be automatically overshadowed.

***iCtopvar* menu**
It refers to the top boundary concentration for *solute transport* simulation:
constant = the top boundary concentration is constant over the whole simulation time. In this case, the user will have to input the *solute pulse parameters* in the **Solute settings** block (see below);
variable = the top boundary concentration is variable over the simulation time. With this option, the user has to input the time variable concentrations in the table include in the **time settings** block (see below).
In the case of *solute transport* = *no*, the *iCtopvar* menu is overshadowed

4. Time settings

This block allows to set the following time parameters for simulations:



time settings

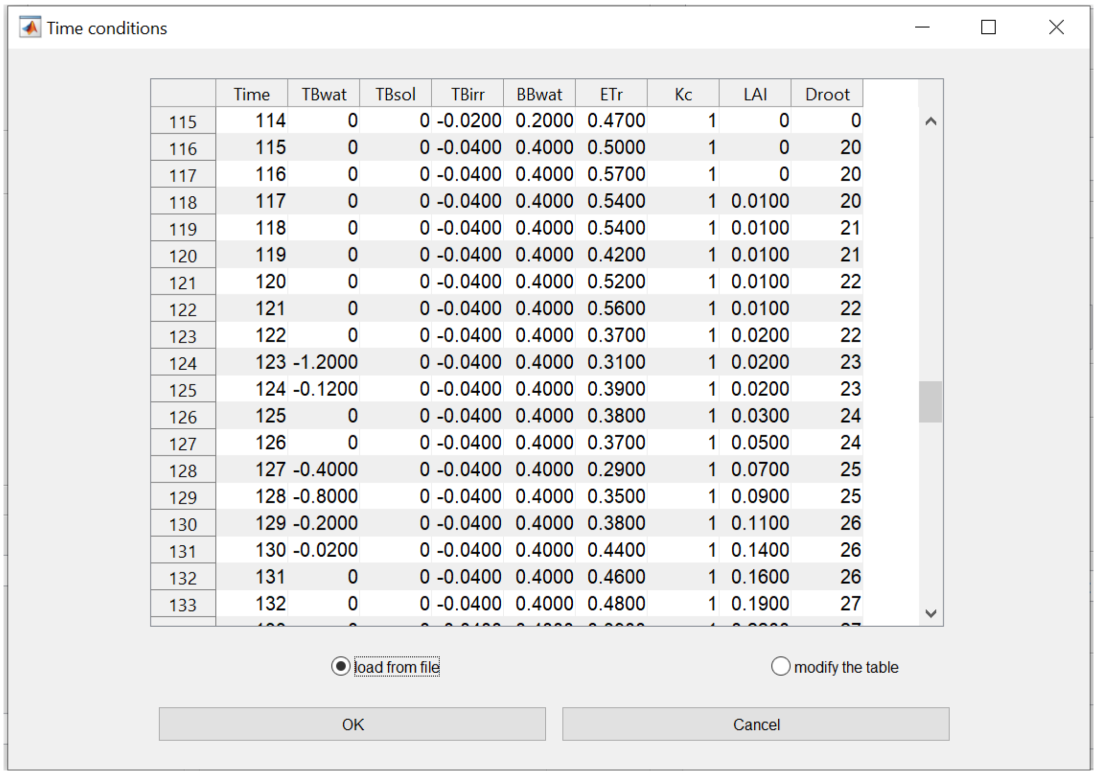
dtin 0.000010
dtmax 1
dtmin 0.000010
tmax 30

Figure xiv. The *time settings* block

dtin = initial simulation *Δ*t ;
dtmax = maximum *Δ*t allowed for simulation;

dtmin = minimum *Δ*t allowed for simulation;
tmax = total simulation time;

The block also includes a window named *Time settings.*
The table in the window allows for setting the time variable conditions in terms of water and solute top and bottom boundary conditions, as well as for vegetation.
The *Time conditions* window may contain up to twelve columns:



Time conditions

	Time	TBwat	TBsol	TBirr	BBwat	ETr	Kc	LAI	Droot
115	114	0	0	-0.0200	0.2000	0.4700	1	0	0
116	115	0	0	-0.0400	0.4000	0.5000	1	0	20
117	116	0	0	-0.0400	0.4000	0.5700	1	0	20
118	117	0	0	-0.0400	0.4000	0.5400	1	0.0100	20
119	118	0	0	-0.0400	0.4000	0.5400	1	0.0100	21
120	119	0	0	-0.0400	0.4000	0.4200	1	0.0100	21
121	120	0	0	-0.0400	0.4000	0.5200	1	0.0100	22
122	121	0	0	-0.0400	0.4000	0.5600	1	0.0100	22
123	122	0	0	-0.0400	0.4000	0.3700	1	0.0200	22
124	123	-1.2000	0	-0.0400	0.4000	0.3100	1	0.0200	23
125	124	-0.1200	0	-0.0400	0.4000	0.3900	1	0.0200	23
126	125	0	0	-0.0400	0.4000	0.3800	1	0.0300	24
127	126	0	0	-0.0400	0.4000	0.3700	1	0.0500	24
128	127	-0.4000	0	-0.0400	0.4000	0.2900	1	0.0700	25
129	128	-0.8000	0	-0.0400	0.4000	0.3500	1	0.0900	25
130	129	-0.2000	0	-0.0400	0.4000	0.3800	1	0.1100	26
131	130	-0.0200	0	-0.0400	0.4000	0.4400	1	0.1400	26
132	131	0	0	-0.0400	0.4000	0.4600	1	0.1600	26
133	132	0	0	-0.0400	0.4000	0.4800	1	0.1900	27

● load from file ○ modify the table

Figure xv. The *time settings* window

1. *Time* (the t_{star}*(kk)* in the text);

and for each *Time*

2. *Top boundary water (TBwat)* (the *qsurf* or *hsurf* in the text)= variable top boundary condition for water (flux, pressure head). *TBwat* will set the *qsurf* if *itbc=flux* and *hsurf* if *itbc = potential;*
3. *Top boundary solute (TBsol)* (*Cinput* in the text)= variable concentrations at the soil surface;
4. *Top boundary irrigation (TBirr)* (the *qirr* in the text)= variable irrigation fluxes at the soil surface;
5. *Bot boundary water (BBwat)* (the *qbot* or *hbot* in the text)= variable bottom boundary condition (flux, pressure head); *BBwat* will set the *qbot* if *ibbc=flux* and *hsurf* if *ibbc =*

potential. The column will disappear if *ibbc = grad*, in which case the *hbot* is calculated internally in the model according to the value set for *grad*;

6. *ET_r* = Reference Evapotranspiration
7. *Kc* = Crop coefficient;
8. *LAI* = Leaf area index.
9. *Droot* (*D_r* in the text)= Maximum depth of roots over time;
10. *Top boundary NH4 (TBNH4)* = variable NH4 concentrations at the soil surface;
11. *Top boundary NO3 (TBNO3)* = variable NH4 concentrations at the soil surface;
12. *Top boundary P (TBPO4)* = variable PO4 concentrations at the soil surface;

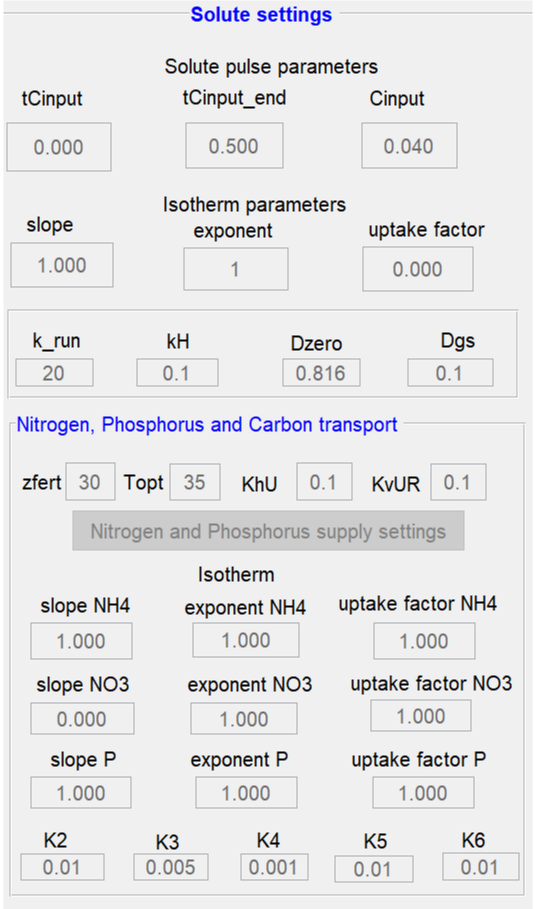
Column 2 only shows up in the case of *itopvar* = *variable*;
Column 3 only shows up in the case of *solute transport* = *yes* – ADE and *iCtopvar* = *variable*
Column 4 only shows up in the case of *irrigation* = *yes (given by the user)*;
Column 5 only shows up in the case of *ibotvar* = *variable* and/or *ibbc* = *flux* or *potential;*
Columns 6 to 9 only show up in the case of *vegetation = yes;*
Columns 10 to 12 only show up in the case of *solute transport = yes – ADE nitrogen* or *solute transport = yes – ADE C-N-P*

In the case of no rainfall, the user has to set *TBwat*=0 (The *ET_r* has to be added in a separate column)

The Time conditions window may be changed value by value by opening the window directly from the graphic box. In an easier way, all the values in the window may be changed by opening the excel file ***time_conditions.xls***. In this case, the user has to be aware that the .xls file always includes all the nine columns and that, depending on the water flow, solute transport and vegetation settings, only some of them will be actually involved in the simulation

5. Solute settings block

This block includes different settings for solute transport simulations.



Solute settings

Solute pulse parameters

tCinput	tCinput_end	Cinput
0.000	0.500	0.040

Isotherm parameters

slope	exponent	uptake factor
1.000	1	0.000

k_run	kH	Dzero	Dgs
20	0.1	0.816	0.1

Nitrogen, Phosphorus and Carbon transport

zfert 30 Topt 35 KhU 0.1 KvUR 0.1

Isotherm

slope NH4	exponent NH4	uptake factor NH4
1.000	1.000	1.000
slope NO3	**exponent NO3**	**uptake factor NO3**
0.000	1.000	1.000
slope P	**exponent P**	**uptake factor P**
1.000	1.000	1.000

K2	K3	K4	K5	K6
0.01	0.005	0.001	0.01	0.01

Figure xvi. The *Solute settings* block

Solute pulse parameters: this part refers to the case of *iCtopvar* = *constant* (if *iCtopvar* = *variable*, the frames will be overshadowed):
tCinput = initial time of application of the solute pulse;
tCinput_end = end time of application of the solute pulse;
Cinput = concentration of the solute pulse. The specific mass of solute, M_s [M/L^2], applied during the time *tCinput_end* – *tCinput,* may be calculated as (see equation 68):

$$M_s = (qsurf + qirr) \cdot Cinput \ \cdot (tCinput_{end} - tCinput)$$

where *qsurf* and *qirr* are the fluxes at the top boundary coming from water rainfall (*qsurf*) and/or irrigation (*qirr*).

Isotherm parameters: This part allows to set the slope and the exponent of the solute adsorption isotherm (see equation 61 C_{ad}=$K_F C^p$)
These two frames named *slope* and *exponent* are related to the option selected in the ***solute transport menu*** described in a previous section. If the selected option is *yes = ADE* and *adsorption*

= *yes – linear*, only the frame *slope* will show up, while the frame *exponent* will remain overshadowed. In the case of *adsorption = yes – Freundlich*, the frame *exponent* will also show up to set the corresponding value.

Kr = Uptake factor. This frame refers to the solute uptake by roots. It sets the solute root uptake preference factor, in the equation 64. It accounts for positive or negative selection of ions relative to the amount of soil water extracted by roots.

k_{run} = Mass transfer coefficient, controlling the solute fluxes from the surface soil layer to runoff.

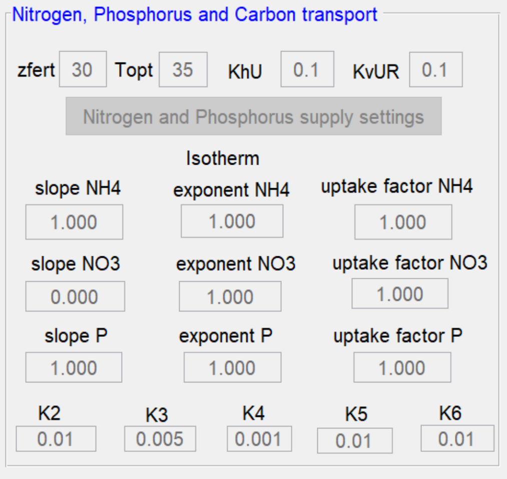
D_0 = Diffusion coefficient in the liquid phase. It sets the molecular diffusion constant of the solute in bulk water. In the case of nitrogen transport simulation (*solute transport = yes - ADE nitrogen*), $D_{0,NH4}$, $D_{0,NO3}$ and $D_{0,P}$ are set internally at the value of 1.61, 1.47 and 0.533 cm^2/d, respectively. $D_{0,CO2}$ is also calculated inside the code for each node as as $D_{0,CO2}$ = 0.8369*exp(0.0269*T) (Engineering ToolBox, 2008. Gases Solved in Water - Diffusion Coefficients. (Available at: https://www.engineeringtoolbox.com/diffusion-coefficients-d_1404.html).

D_g^s= Dispersion coefficient in the gaseous phase.

K_H =*Henry constant*: It is the slope of the linear equilibrium relationship between gaseous, C_g, and liquid, C, concentrations.

5.1. Nitrogen, Phosphorus and Nitrogen transport sub-block

This sub-block includes the settings for either the Nitrogen transport (in the case of *solute transport = yes ADE* option) or of Nitrogen, Phosphorus and Carbon transport (in the case of *solute transport = yes ADE C-N-P* option).



Nitrogen, Phosphorus and Carbon transport

zfert 30 Topt 35 KhU 0.1 KvUR 0.1

Nitrogen and Phosphorus supply settings

Isotherm

slope NH4	exponent NH4	uptake factor NH4
1.000	1.000	1.000
slope NO3	exponent NO3	uptake factor NO3
0.000	1.000	1.000
slope P	exponent P	uptake factor P
1.000	1.000	1.000

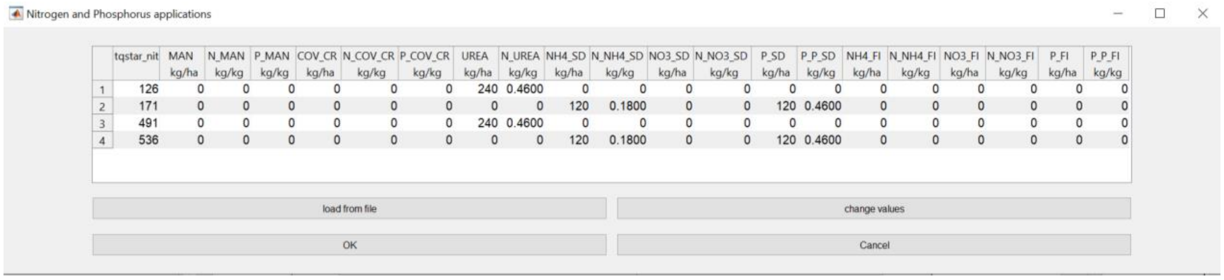
K2	K3	K4	K5	K6
0.01	0.005	0.001	0.01	0.01

Figure xvii. The *Nitrogen, Phosphorus and Carbon transport settings* sub-block

Under both the options, the user will be asked for setting the following parameters:

zfert = the incorporation depth of fertilizers. It is assumed that nitrogen addition is distributed uniformly along the incorporation depth;
Topt = the optimum temperature for the immobilization, nitrification and denitrification processes and set by default at 35°C in the model;
KhUR = the first-order rate coefficients for the urea hydrolysis decay reaction;
KvUR = the first-order rate coefficients for the urea volatilization decay reaction;
Isotherm parameters for both the NH4 and the NO3 nitrogen forms and for PO4 (in the case of *ADE C-N-P*): under both the cases, the option *adsorption = yes – linear*, is automatically set and only the frame *slope* will show up, while the frame *exponent* will remain overshadowed and fixed to 1. Note that the slope for phosphorus corresponds to the parameter K_{ads1} in figure 11.
Kr = Uptake factor for NH4, NO3 and PO4: These frames refer to the N-NH4, N-NO3 and PO4 uptake by roots.
K2, K3, K4, K5 and K6 correspond respectively to K_{ads2}, K_{chs1}, K_{chs2}, K_{prc1} and K_{prc2} in figure 11. They represent the first order kinetic constant related to phosphorus reactions of desorption, chemisorption and mobilization, precipitation and dissolution, respectively.

The sub-block also includes a window named *Nitrogen and Phosphorus applications*. By clicking the *Nitrogen and Phosphorus supply settings* button, the user will firstly be asked for the number of fertilizer applications during the simulation (three in the window shown below). Thus, the following window will show up:



	tqstar_nit	MAN kg/ha	N_MAN kg/kg	P_MAN kg/kg	COV_CR kg/ha	N_COV_CR kg/kg	P_COV_CR kg/kg	UREA kg/ha	N_UREA kg/kg	NH4_SD kg/ha	N_NH4_SD kg/kg	NO3_SD kg/ha	N_NO3_SD kg/kg	P_SD kg/ha	P_P_SD kg/kg	NH4_FI kg/ha	N_NH4_FI kg/kg	NO3_FI kg/ha	N_NO3_FI kg/kg	P_FI kg/ha	P_P_FI kg/kg
1	126	0	0	0	0	0	0	240	0.4600	0	0	0	0	0	0	0	0	0	0	0	0
2	171	0	0	0	0	0	0	0	0	120	0.1800	0	0	120	0.4600	0	0	0	0	0	0
3	491	0	0	0	0	0	0	240	0.4600	0	0	0	0	0	0	0	0	0	0	0	0
4	536	0	0	0	0	0	0	0	0	120	0.1800	0	0	120	0.4600	0	0	0	0	0	0

Figure xviii. The *Nitrogen applications settings* window

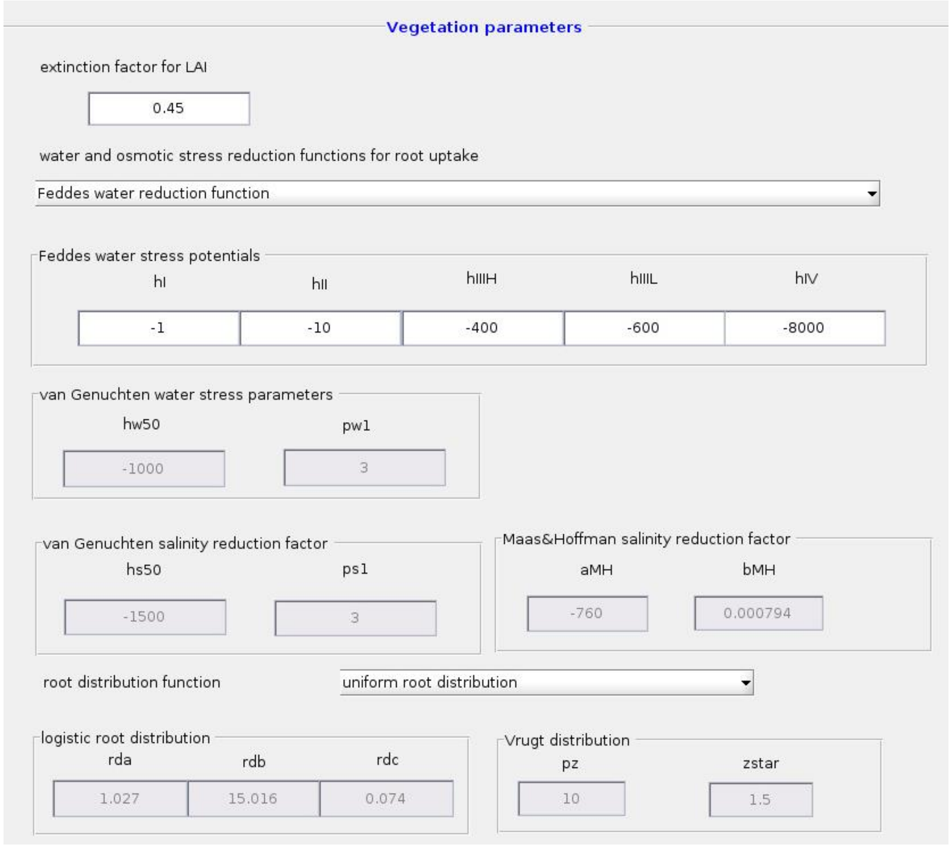
The window allows setting additions of nitrogen in the forms of manure, crop residue, mineral fertilizers. All the applications are in kg/ha and are internally converted to g/cm^2.
The model requires the following inputs:
tqstar_nit = times of fertilizer applications;
MAN and *N-MAN* = manure amount in kg/ha and the nitrogen content in the fertilizer (kg/kg);
COV-CR and *N-COV-CR* = cover crop amount in kg/ha and the nitrogen content in the fertilizer (kg/kg);
UREA and *N-UREA* = urea amount in kg/ha and the nitrogen content in the fertilizer (kg/kg);

NH4-SD, and *N-NH4-SD* = solid mineral ammonium fertilizer amount in kg/ha and the nitrogen content in the fertilizer (kg/kg);
NO3-SD and *N-NO3-SD* = solid mineral nitrate fertilizer amount in kg/ha and the nitrogen content in the fertilizer (kg/kg);
PO4-SD, and *P-PO4-SD* = solid mineral phosphorus fertilizer amount in kg/ha and the phosphorus content in the fertilizer (kg/kg);
NH4-FI and *N-NH4-FI* = liquid (supplied by injection or fertigation, for example) mineral ammonium fertilizer amount in kg/ha and the nitrogen content in the fertilizer (kg/kg);
NO3-FI and *N-NO3-FI* = liquid (supplied by injection or fertigation, for example) mineral nitrate fertilizer amount in kg/ha and the nitrogen content in the fertilizer (kg/kg);
PO4-FI, and *P-PO4-FI* = liquid mineral phosphorus fertilizer amount in kg/ha and the phosphorus content in the fertilizer (kg/kg);

Once set all the fertilizer applications, by clicking the button OK the user will be asked to save the table as a .mat file and .xls file named *nitrogen_settings_?*? (the double question points are for the name the user wants to give to the file).

6. Vegetation settings block

This block contains the settings related to the vegetation (evapotranspiration, water and osmotic stresses, root distribution).



Vegetation parameters

extinction factor for LAI

0.45

water and osmotic stress reduction functions for root uptake

Feddes water reduction function

Feddes water stress potentials

hI	hII	hIIIH	hIIIL	hIV
-1	-10	-400	-600	-8000

van Genuchten water stress parameters

hw50	pw1
-1000	3

van Genuchten salinity reduction factor

hs50	ps1
-1500	3

Maas&Hoffman salinity reduction factor

aMH	bMH
-760	0.000794

root distribution function: uniform root distribution

logistic root distribution

rda	rdb	rdc
1.027	15.016	0.074

Vrugt distribution

pz	zstar
10	1.5

Figure xix. The *vegetation parameters* block

Extinction factor for LAI= exponent of the Beer's law for separating potential evaporation and transpiration from total potential evapotranspiration

***Water and Osmotic stress reduction functions for root uptake* menu**

This menu allows for selecting the type of reduction function for root water uptake related too either water stress, osmotic stress or both. In the latter case, the code assumes a multiplicative combination of water and salinity stresses.

- Feddes water reduction function
- van Genuchten water reduction function
- Mass & Hoffman salinity reduction function
- van Genuchten salinity reduction function
- Multiplicative van Genuchten water and salinity reduction function
- Multiplicative Feddes (water) and Mass & Hoffman (salinity) reduction function

The value in the following frames has to be set depending on the option selected for the stress reduction function:

Feddes water stress potentials
hI, hII, hIIIH, hIIIL, hIV = critical pressure heads for the Feddes water reduction function;
van Genuchten water stress parameters
hw50 = the soil water pressure head at which root water uptake is reduced by 50%
pw1 = a fitting parameter frequently assumed to be 3
van Genuchten salinity stress parameters
hs50 = the soil water osmotic potential at which root water uptake is reduced by 50%
ps1 = a fitting parameter frequently assumed to be 3
Mass & Hoffman salinity stress parameters
aMH = threshold osmotic potential value for the Mass and Hoffman salinity reduction function
bMH = slope of the Mass and Hoffman salinity reduction function

It should be noted that *aMH* and *bMH* parameterize local reductions in the root water uptake rate as a function of osmotic head. In this sense, they should not be confused with the parameters A and B frequently found in the literature and that parameterize total yield reductions as a function of average root zone salinity.

Depending on the water and salinity reduction functions selected, only the frames with the parameters actually involved in the simulation will show up. The other will be automatically overshadowed

***Root Distribution function* menu**

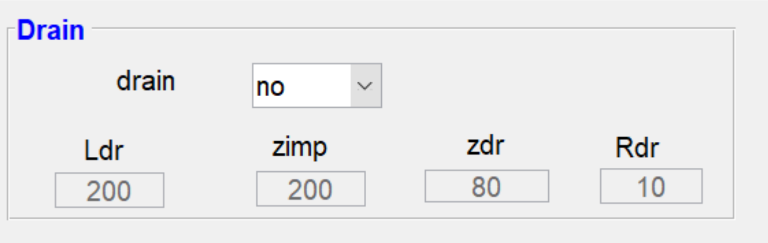
- *Uniform root distribution* (the distribution is just 1/*Droot* and does not need parameters to be input in this window);
- *Logistic root distribution*;
- *Prasad root distribution* (it is a triangular distribution and does not need parameters to be input in this window);
- *Vrugt distribution*

rda, rdb rdc = *Logistic root distribution* parameters
pz, zstar = *Vrugt root distribution* parameters

Depending on the root distribution function selected, only the frames with the parameters actually involved in the simulation will show up. The other will be automatically overshadowed.

7. Drainage settings block

This block allows setting the parameters used by the model for calculating the fluxes to artificial drains (both drainpipes and open field drains).



Drain

drain no

Ldr	zimp	zdr	Rdr
200	200	80	10

Figure xx. The *drainage settings* block

It considers two options to account or not for the solute transport:
no = artificial drainage is not simulated (all the frames will be overshadowed);
yes = artificial drainage is simulated according to the Hooghoudt's approach. In this case, the following frames will show up to set the required input parameters:
Ldr = drain distance;
zimp = the depth of the impermeable layer (from the soil surface);
zdr = depth of drainpipes installation or of the base of open drains;
Rdr = the hydraulic radius of the drain

8. Units

This block allows to set up the units for data and parameters used in the text. The code uses cm for length and g for mass. The time units have to be selected by the user as seconds, minutes, hours or days. These units will be used by the code to set the titles in the graphs generated by the code at the end of a simulation run (see the section below *Model outputs*)

A.3. Model output

The output files include both text (.dat) and matlab (.mat) format data.
The first type may be open as .xls files and used for new numerical and graphical elaborations, for example using excel.
The table i provides a list of the output files, with a description of the variables recorded in the output files:

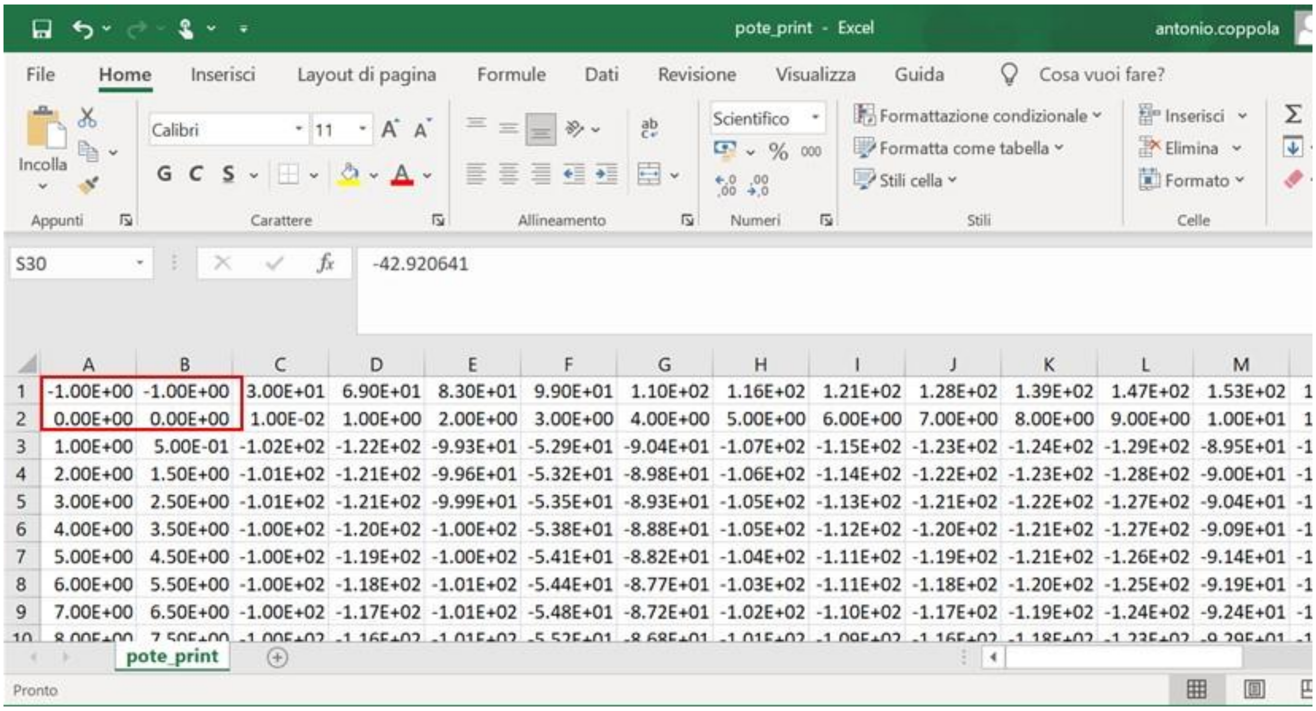
Table i. Output of FLOWS

	Output file	**File content**
1	pote_tot	Pressure heads (L) as a function of depth and time for all the simulation times
2	pote_print	Pressure heads (L) as a function of depth and time for print times

3	teta_print	Water contents (L^{3}L^{-3}) as a function of depth and time for print times
4	cap_print	Water capacity (the derivative of the water retention curve) (L^{3}L^{-3}L^{-1}) as a function of depth and time for print times
5	flux_print	Water fluxes (LT^{-1}) as a function of depth and time for print times
6	sink_print	Sink (water uptake) (T^{-1}) as a function of depth and time for print times
7	sink_drain	Artificial Drainage losses (T^{-1}) as a function of depth and time for print times
8	conc_print	Solute concentrations (ML^{-3}) as a function of depth and time for print times
9	conc_DR	Solute mass (ML^{-2}T^{-1}) leaving the soil profile by drainage as a function of depth and time for print times
10	conc_NH4_print	Ammonium nitrogen (N-NH4) solute concentrations (ML^{-3}) as a function of depth and time for print times
11	conc_NO3_print	Nitrate nitrogen (N-NH4) solute concentrations (ML^{-3}) as a function of depth and time for print times
12	conc_PO4_print	Phosphorus (P-PO4) solute concentrations (ML^{-3}) as a function of depth and time for print times
13	conc_CO2_print	CO2 mass (ML^{-2}T^{-1}) produced in the soil profile by root and microbial respiration as a function of depth and time for print times
14	conc_NH4_DR	Ammonium nitrogen (N-NH4) mass (ML^{-2}T^{-1}) leaving the soil profile by drainage as a function of depth and time for print times
15	conc_NO3_DR	Nitrate nitrogen (N-NO3) mass (ML^{-2}T^{-1}) leaving the soil profile by drainage as a function of depth and time for print times
16	conc_PO4_DR	Phosphorus (P-PO4) mass (ML^{-2}T^{-1}) leaving the soil profile by drainage as a function of depth and time for print times
17	flux_surf_bot	Water fluxes (LT^{-1}) at top and bottom boundary as a function of time for print times

18	runoff	Water and solute fluxes to runoff (LT^{-1}) as a function of time for print times
19	qirr	Irrigation fluxes (LT^{-1}) (gross and net irrigation) for print times in the case of irrigation computed by the model

All the .dat files from 1 to 13 in the table i include a matrix of data with 102 lines (number of nodes +2) and a number of columns = number of print times +2 (see the figure xv). Only in the case of *pote_tot*, the file contains as many columns as the actual number of Δt used during the simulation +2.



S30 -42.920641

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	-1.00E+00	-1.00E+00	3.00E+01	6.90E+01	8.30E+01	9.90E+01	1.10E+02	1.16E+02	1.21E+02	1.28E+02	1.39E+02	1.47E+02	1.53E+02	1
2	0.00E+00	0.00E+00	1.00E-02	1.00E+00	2.00E+00	3.00E+00	4.00E+00	5.00E+00	6.00E+00	7.00E+00	8.00E+00	9.00E+00	1.00E+01	1
3	1.00E+00	5.00E-01	-1.02E+02	-1.22E+02	-9.93E+01	-5.29E+01	-9.04E+01	-1.07E+02	-1.15E+02	-1.23E+02	-1.24E+02	-1.29E+02	-8.95E+01	-1
4	2.00E+00	1.50E+00	-1.01E+02	-1.21E+02	-9.96E+01	-5.32E+01	-8.98E+01	-1.06E+02	-1.14E+02	-1.22E+02	-1.23E+02	-1.28E+02	-9.00E+01	-1
5	3.00E+00	2.50E+00	-1.01E+02	-1.21E+02	-9.99E+01	-5.35E+01	-8.93E+01	-1.05E+02	-1.13E+02	-1.21E+02	-1.22E+02	-1.27E+02	-9.04E+01	-1
6	4.00E+00	3.50E+00	-1.00E+02	-1.20E+02	-1.00E+02	-5.38E+01	-8.88E+01	-1.05E+02	-1.12E+02	-1.20E+02	-1.21E+02	-1.27E+02	-9.09E+01	-1
7	5.00E+00	4.50E+00	-1.00E+02	-1.19E+02	-1.00E+02	-5.41E+01	-8.82E+01	-1.04E+02	-1.11E+02	-1.19E+02	-1.21E+02	-1.26E+02	-9.14E+01	-1
8	6.00E+00	5.50E+00	-1.00E+02	-1.18E+02	-1.01E+02	-5.44E+01	-8.77E+01	-1.03E+02	-1.11E+02	-1.18E+02	-1.20E+02	-1.25E+02	-9.19E+01	-1
9	7.00E+00	6.50E+00	-1.00E+02	-1.17E+02	-1.01E+02	-5.48E+01	-8.72E+01	-1.02E+02	-1.10E+02	-1.17E+02	-1.19E+02	-1.24E+02	-9.24E+01	-1
10	8.00E+00	7.50E+00	-1.00E+02	-1.16E+02	-1.01E+02	-5.52E+01	-8.68E+01	-1.01E+02	-1.09E+02	-1.16E+02	-1.18E+02	-1.23E+02	-9.29E+01	-1

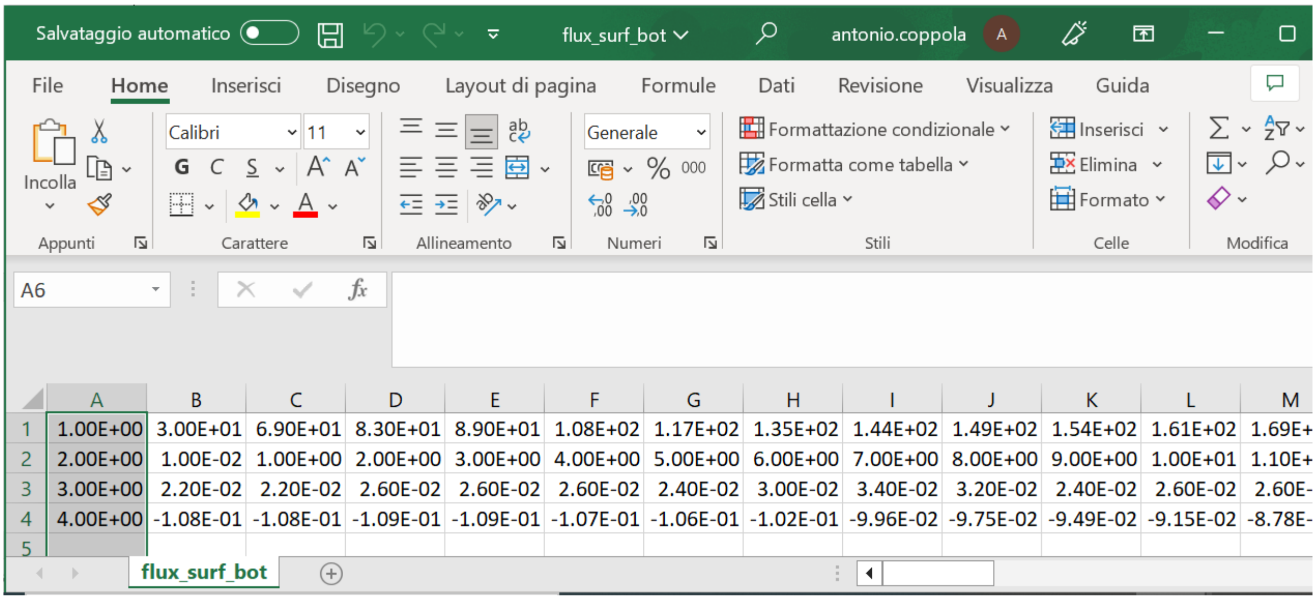
Figure xxi. An example of an output file (for all files from 1 to 13 in the table i)

In the matrix, the first four cells (those into the red rectangle) are dummy numbers.
The first two lines report the simulation time in terms, respectively, of number time steps (*j* in the time discretization used in the code) and corresponding print time.
Similarly, the first two columns report the simulation node (i in the depth discretization used in the code) and the corresponding actual depth along the soil profile.
All the other numbers refer to the specific variable recorded in the file (potentials, sink, fluxes,….)

The figure xxii reports an example of the output file 14 in the table i (named *flux_surf_bot*). In the file, the first column reports the line number. The first two lines reports the simulation time in terms, respectively, of number time steps (*j* in the time discretization used in the code) and corresponding print time. The other numbers represent:

- the fluxes passing through the soil surface (line 3);
- the fluxes passing through the bottom of the simulated domain (line 4).

(Note that positive values are for upward fluxes and negative values are for downward fluxes).



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	1.00E+00	3.00E+01	6.90E+01	8.30E+01	8.90E+01	1.08E+02	1.17E+02	1.35E+02	1.44E+02	1.49E+02	1.54E+02	1.61E+02	1.69E+
2	2.00E+00	1.00E-02	1.00E+00	2.00E+00	3.00E+00	4.00E+00	5.00E+00	6.00E+00	7.00E+00	8.00E+00	9.00E+00	1.00E+01	1.10E+
3	3.00E+00	2.20E-02	2.20E-02	2.60E-02	2.60E-02	2.60E-02	2.40E-02	3.00E-02	3.40E-02	3.20E-02	2.40E-02	2.60E-02	2.60E-
4	4.00E+00	-1.08E-01	-1.08E-01	-1.09E-01	-1.09E-01	-1.07E-01	-1.06E-01	-1.02E-01	-9.96E-02	-9.75E-02	-9.49E-02	-9.15E-02	-8.78E-
5													

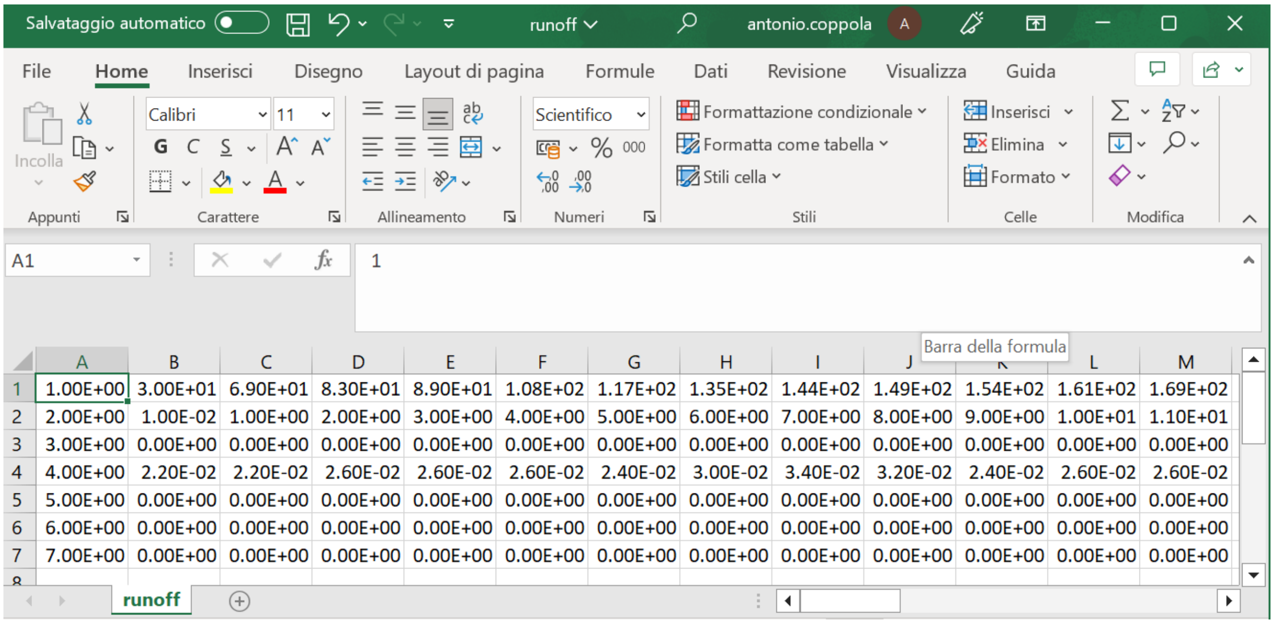
Figure xxii. An example of output file named *flux_surf_bot*

The figure xxiii reports an example of the output file 15 in the table i (named *runoff*). The first column and the first two lines are as in the previous table. The other numbers represent respectively:

- the water fluxes applied at the **TB** (rainfall, irrigation, …)(line 3);
- the water fluxes passing through the soil surface (line 4);
- the water fluxes to runoff (line 5);
- the solute fluxes to runoff (line 6).

In the case of nitrogen and phosphorus transport:

- the N-NH4 solute fluxes to runoff (line 6);
- the N-NO3 solute fluxes to runoff (line 7).
- the P-PO4 solute fluxes to runoff (line 8)



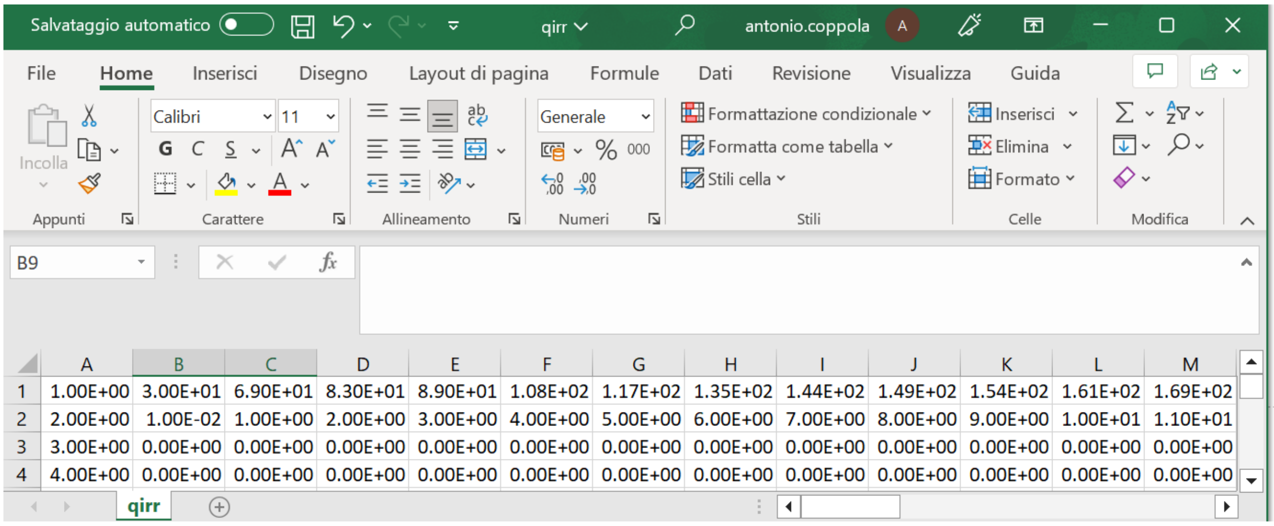
	A	B	C	D	E	F	G	H	I	J	K	L	M
1	1.00E+00	3.00E+01	6.90E+01	8.30E+01	8.90E+01	1.08E+02	1.17E+02	1.35E+02	1.44E+02	1.49E+02	1.54E+02	1.61E+02	1.69E+02
2	2.00E+00	1.00E-02	1.00E+00	2.00E+00	3.00E+00	4.00E+00	5.00E+00	6.00E+00	7.00E+00	8.00E+00	9.00E+00	1.00E+01	1.10E+01
3	3.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
4	4.00E+00	2.20E-02	2.20E-02	2.60E-02	2.60E-02	2.60E-02	2.40E-02	3.00E-02	3.40E-02	3.20E-02	2.40E-02	2.60E-02	2.60E-02
5	5.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
6	6.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
7	7.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00

Figure xxiii. An example of output file named *runoff*

The figure xxivi shows an example of the output file 16 in the table i (named *qirr*) and reports:

- the gross irrigation fluxes (line 3);
- the net irrigation fluxes (line 4).

FLOWS produces the *qirr* output file only in the case of the option *irrigation = yes (computed by the model)*



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	1.00E+00	3.00E+01	6.90E+01	8.30E+01	8.90E+01	1.08E+02	1.17E+02	1.35E+02	1.44E+02	1.49E+02	1.54E+02	1.61E+02	1.69E+02
2	2.00E+00	1.00E-02	1.00E+00	2.00E+00	3.00E+00	4.00E+00	5.00E+00	6.00E+00	7.00E+00	8.00E+00	9.00E+00	1.00E+01	1.10E+01
3	3.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
4	4.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00

Figure xxiv. An example of output file named *qirr*

The matlab data are used for graphical output, showing selected output variables at the end of each model run for selected times and depths (namely, water contents, pressure heads, concentrations, fluxes and eventual runoff). At the end of a simulation run, the model produces the following graphs for water potentials, water contents, concentrations and fluxes:

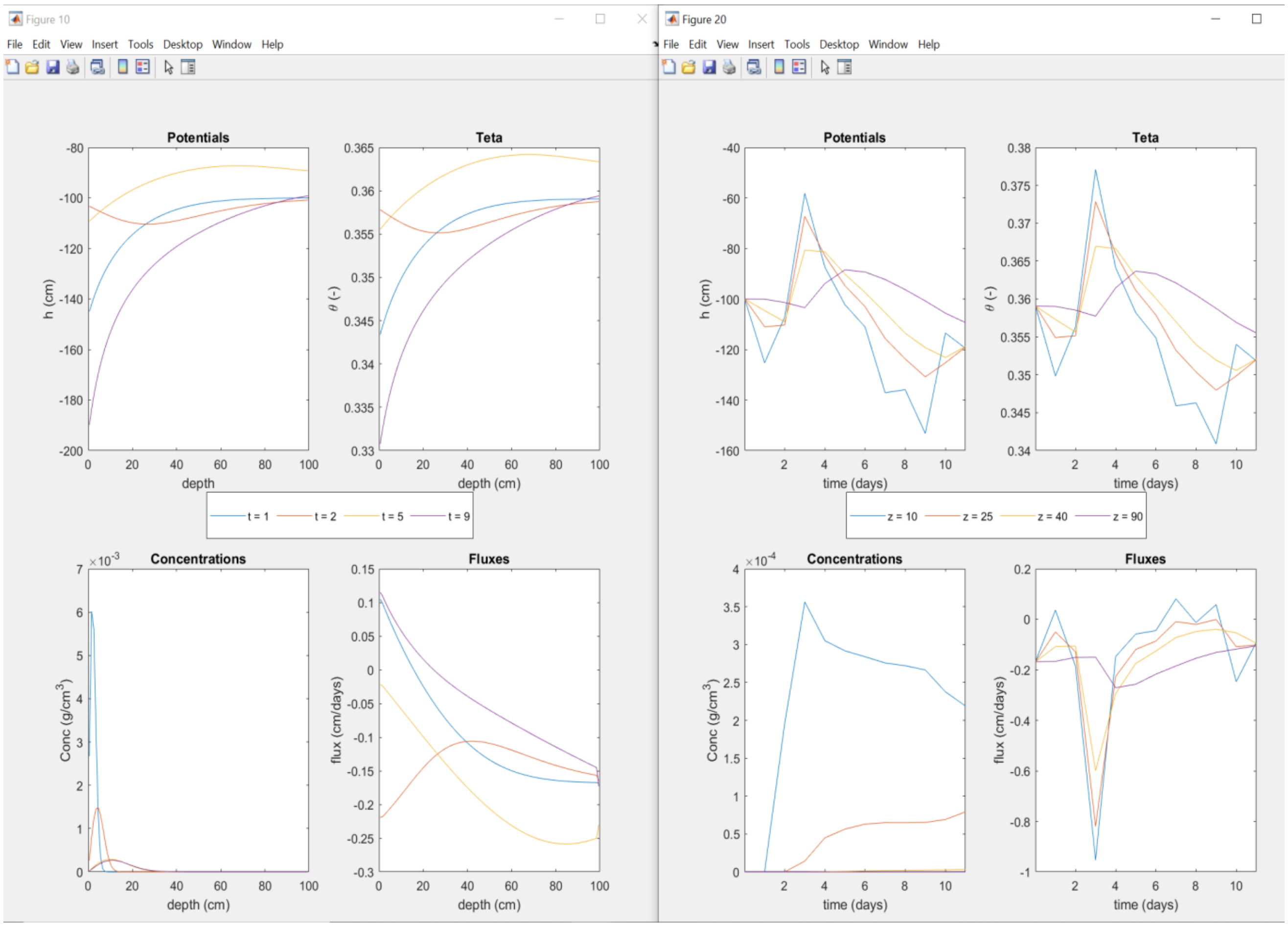


Figure xxv. Graphs of water potentials, water contents, concentrations and fluxes built by the model

An additional graph is provided for the eventual runoff fluxes:

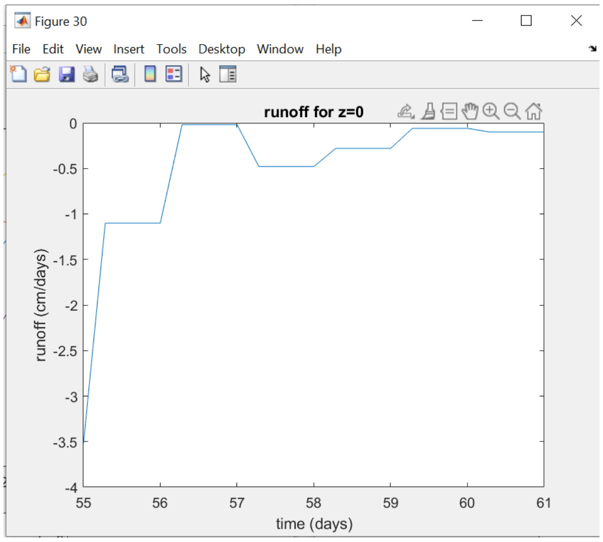


Figure xxvi. Graph for runoff built by the model

In the case of nitrogen transport the following additional graphs will appear, replacing the concentrations graphs saw in the case of a single solute transport:

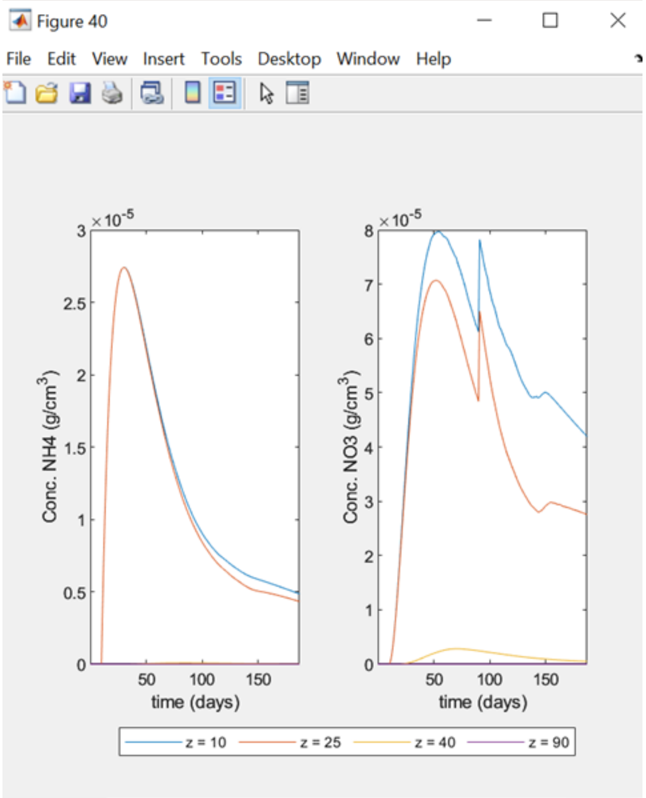
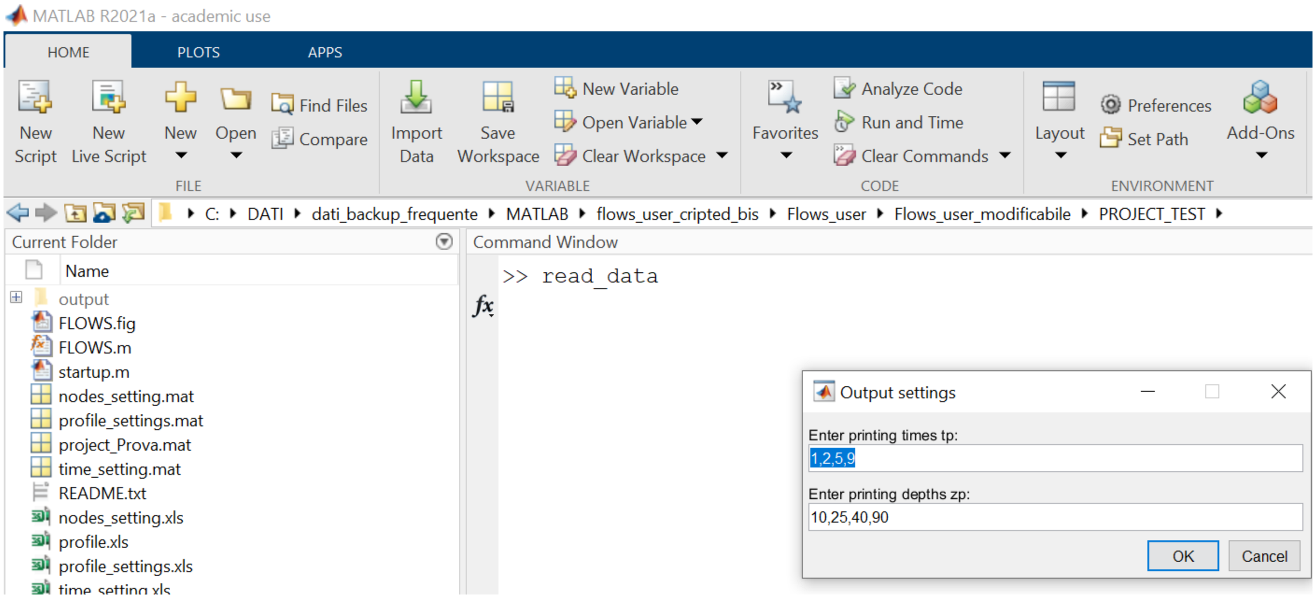


Figure xxvii. Graphs for N-NH4 and N-NO3 concentrations over time built by the model

The user may decide the print times and the depths to be used in the plots (up to four times and depths). Once closed, it is still possible to ask the code for new graphs (for different times and depths) by remaining in the current folder project (for example, PROJECT_TEST) and typing *read_data* in the Matlab command window. A window will show up asking the user to input times and depths for drawing the graphs



```
>> read_data
```

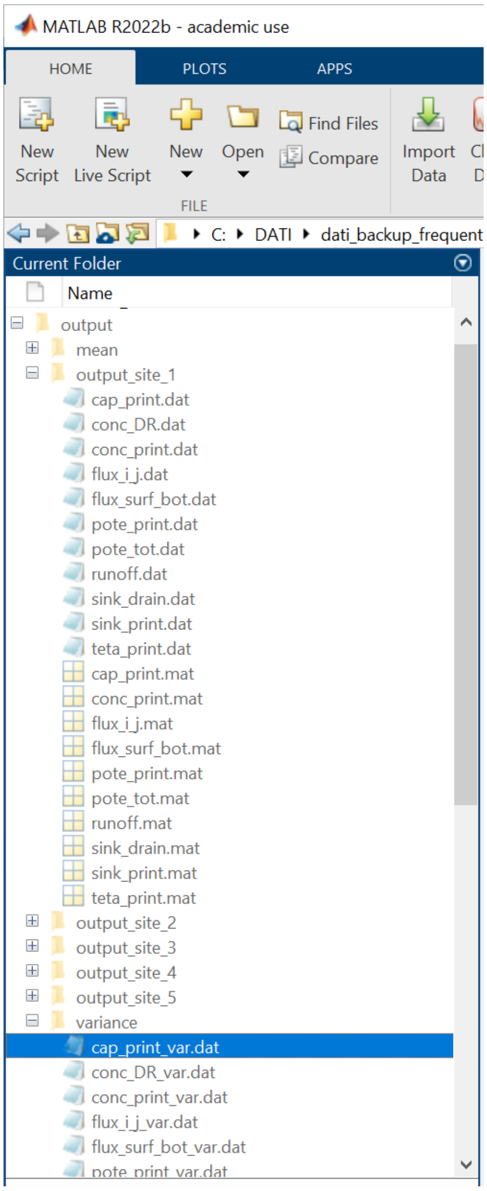
Enter printing times tp: 1,2,5,9

Enter printing depths zp: 10,25,40,90

Figure xxviii. Using the *read_data* command to plot new graphs

Multiple sites configuration outputs

In the case the *multi-sites* configuration has been selected, the model will produce an output folder including a folder for each simulated site, containing the same outputs produced for the *single-site* configuration. Additionally, the model also produces two additional output folders, reporting respectively the mean and variance of the output variables calculated for each site (see the following figure, providing an example of *multi-sites* simulation with 5 sites). The mean and the variance may be useful in the case of simulations carried out in a stochastic framework, where the different sites do not really correspond to spatially distributed points but may represent different hydraulic parameter vectors for a single simulation point.



```
output
  mean
  output_site_1
    cap_print.dat
    conc_DR.dat
    conc_print.dat
    flux_i_j.dat
    flux_surf_bot.dat
    pote_print.dat
    pote_tot.dat
    runoff.dat
    sink_drain.dat
    sink_print.dat
    teta_print.dat
    cap_print.mat
    conc_print.mat
    flux_i_j.mat
    flux_surf_bot.mat
    pote_print.mat
    pote_tot.mat
    runoff.mat
    sink_drain.mat
    sink_print.mat
    teta_print.mat
  output_site_2
  output_site_3
  output_site_4
  output_site_5
  variance
    cap_print_var.dat
    conc_DR_var.dat
    conc_print_var.dat
    flux_i_j_var.dat
    flux_surf_bot_var.dat
    pote_print_var.dat
```

Figure xxix. The output folders for the *multi-site* configuration

The graphs built by FLOWS at the end of the *multi-site* simulations will be the same as those for *single-site* simulations and will show the average of the simulation outputs.

References

Barrow, N.J., Madrid, L., Posner, A.M., 1981. A partial model for the rate of adsorption and desorption of phosphate by goethite, J. Soil Sci., 32, 399-407.

Bear J., 1979, Hydraulics and groundwater, McGraw Hill, N.Y.De Wit, C.T., 1958. Transpiration and Crop Yields. Versl. Landbouwk. On-derz., vol. 64. Wageningen University, The Netherlands, pp. 59–84.

Bresler, E., Hoffman, G.J., 1986. Irrigation management for soil salinity control: theories and tests. Soil Sci. Soc. Am. J. 50, 1552–1560

Cabon, F., Cirard G., and Ledoux, E. 1991. Modelling of the nitrogen cycle in farm land areas. Fert. Res. 27:161–169.

Carsel, R.F., Parrish, R.S., 1988. Developing joint probability distributions of soil water retention characteristics. Water Resources Research 24, 755–769.

Childs, S.W., Hanks, R.J., 1975. Model of soil salinity effects on crop growth. Soil Sci. Soc. Am. Proc. 39, 617–622

Cleveland, C.C. and D. Liptzin. 2007. C:N:P stoichiometry in soil: is there a ''Redfield ratio'' for the microbial biomass? Biogeochemistry. 85, 235–252. doi: 10.1007/s10533-007-9132-0

Comegna V, Coppola A, Sommella A. 1999: Nonreactive solute transport in variously structured soil materials as determined by laboratory-based time domain reflectometry (TDR). Geoderma, 92, 167–184. doi: 10.1016/S0016-7061(99)00030-0

Coppola, A. 2000: Unimodal and bimodal descriptions of hydraulic properties for aggregated soils. Soil Sci. Soc. Am. J., 64, 1252–1262.

Coppola A., Dragonetti G., Abdallah M., Zdruli P, Lamaddalena N., 2019. Monitoring and Modeling soil water recharge from ephemeral flows and its depletion in a reclaimed wadi basin in Egypt. Ecohydrology, doi: 10.1002/eco.2084Res., 48, W08527. doi:10.1029/2011WR011376

Coppola, A., Basile A., Comegna A., Lamaddalena N., 2009. 2008. Monte Carlo analysis of field water flow comparing uni- and bimodal effective hydraulic parameters for structured soil, Journal of Contaminant Hydrology, doi: 10.1016/j.jconhyd.2008.09.007.

Coppola, A., Randazzo, L., 2006. A MATLAB code for the transport of water and solutes in unsaturated soils with vegetation. Tech. Rep. Soil and Contaminant Hydrology Laboratory, Dept. DITEC University of Basilicata.

Coppola, A., Comegna, A., Dragonetti, G., Gerke, H.H., Basile, A. Simulated preferential water flow and solute transport in shrinking soils. 2015. Vadose Zone J. 2015, 14, doi:10.2136/vzj2015.02.0021.

Coppola, G. Dragonetti, A. Comegna, P. Zdruli, N. Lamaddalena, S. Pace and L. De Simone. 2014. Mapping solute deep percolation fluxes at regional scale by integrating a process-based vadose zone model in a Monte Carlo approach, Soil Science and Plant Nutrition, doi: 10.1080/00380768.2013.855615N.

Coppola, H. H. Gerke, A. Comegna, A. Basile, V. Comegna, 2012. Dual-permeability model for flow in shrinking soil with dominant horizontal deformation. Water Resources Research, Vol. 48, W08527, doi:10.1029/2011WR011376.

De Wit, C.T. 1958. Transpiration and Crop Yields. Agricultural Research Reports 64.6 Pudoc, Wageningen, 88 p.

Doorenbos, J., and A. H. Kassam. 1979. Yield response to water. FAO Irrig & Drain Paper No. 33. FAO, Rome, Italy, p. 193.

du Plessis, H.M., 1985. Evapotranspiration of citrus as affected by soil water deficit and soil salinity. Irrig. Sci. 6, 51–61

Durner, W., 1994. Hydraulic conductivity estimation for soils with heterogeneous pore structure. Water Resour. Res. 30, 211–223.

Elias E.A., Cichota R., Torriani H.H., De Jong van Lier, Q., 2004. Analytical soil–temperature model correction for temporal variation of daily amplitude. Soil Sci. Soc. Am. J. 68:784–788

Feddes RA, Kowalik PJ, Zaradny H. 1978. Simulation of Field Water Useand Crop Yield. Centre for Agricultural Publishing and Documentation: Wageningen, The Netherlands 189 p.

Feddes RA, Raats PAC. 2004. Parameterizing the soil-water-plant root system, 95–141. In Unsaturated-zone Modeling: Progress, Challenges, Applications. UR Frontis Ser. 6. Feddes RA et al. (ed). Kluwer Acad. Press: Wageningen, The Netherlands.

Gardner W.R. (1958). Some steady-state solutions of the unsaturated moisture flow equation with applications to evaporation from a water table. Soil science, 85:228-232.

Gusman AJ, Marino MA. Analytical modeling of nitrogen dynamics in soils and ground water. Journal of Irrigation and Drainage Engineering 1999;125(6):330-337.

Hansen, S., P. Abrahamsen, C. T. Petersen, and M. Styczen (2012), Daisy: Model use, calibration and validation, Trans. ASABE, 55(4), 1315– 1333.

Healy, R. W., Simulation of solute transport in variably saturated porous media with supplemental information on modifications to the U.S. Geological Survey's computer program VS2D, U.S. Geol. Surv. Water Resour. Invest. Rep., 90-4025, 125 pp., 1990.

Holmes, T. R. H., M. Owe, R. A. M. De Jeu, and H. Kooi, 2008. Estimating the soil temperature profile from a single depth observation: A simple empirical heatflow solution, Water Resour. Res., 44, W02412, doi:10.1029/2007WR005994

Homaee M, Feddes RA, Dirksen C. 2002a. Simulation of root water uptake. I. Non-uniform transient salinity using different macroscopic reduction functions. Agricultural Water Management 57: 89–109.

Homaee M, Feddes RA, Dirksen C. 2002b. Simulation of root water uptake. II. Non-uniform transient water stress using different reduction functions. Agricultural Water Management 57: 111–126.

Homaee M, Feddes RA, Dirksen C. 2002c. Simulation of root water uptake. III. Non-uniform transient combined salinity and water stress. Agricultural Water Management 57: 127–144.

Hooghoudt, S.B. 1940. General consideration of the problem of field drainage by parallel drains, ditches, watercourses, and channels. Publ. No.7 in the series Contribution to the knowledge of some physical parameters of the soil (titles translated from Dutch). Bodemkundig Instituut, Groningen, The Netherlands

Huang K., B. P. Mohanty, and M. T. Van Genuchten, 1996a. A new convergence criterion for the modified Picard iteration method to solve the variably saturated flow equation. Journal of Hydrology, 78:69–91.

Huang K., M.T. van Genuchten and Zhang R., 1996b. Exact solutions for one-dimensional transport with asymptotic scale-dependent dispersion. Appl. Math. Modelling 1996, Vol. 20, Published by Elsevier Science Inc. 6.55 Avenue of the Americas, New York, NY 10010

Jarvis NJ. 1989. A simple empirical model of root water uptake. Journal of Hydrology 107(1): 57–72.

Jensen, M.E. (1980), Design and operation of farm irrigation system, American Society of Agricultural Engineers, St. Joseph

Jones C.A., Cole C. V, Sharpley A.N. and Williams J.R. (1984). A Simplified Soil and Plant Phosphorus Model: I. Documentation. Soil Science Society of America Journal, 48(4): 800–805. John Wiley & Sons, Ltd. doi: https://doi.org/10.2136/sssaj1984.03615995004800040020x

Karvonen, T, H. Koivusalo, M. Jauhiainen, J. Palko, K. Weppling, 1999. A hydrological model for predicting runoff from different land use areas. Journal of Hydrology 217 (1999) 253–265

Karvonen, T., 1988. A model for predicting the effect of drainage on soil moisture, soil temperature and crop yield. Publications of Lab. Hydrol. and Water Resour. Manage. 1. Helsinki University of Technology, Finland.

Kersebaum, K. C., and Richter, J., 1991. ‘‘Modelling nitrogen dynamics in a plant-soil system with a simple model for advisory purposes.’’ Fert. Res., 27, 273–281

Kroes, J.G., J.C. Van Dam, P. Groenendijk, R.F.A. Hendriks, C.M.J. Jacobs, 2008. SWAP version 3.2. Theory description and user manual. Wageningen, Alterra, Alterra Report 1649, Swap32 Theory description and user manual. 262 pages

Lafolie, F. 1991. Modelling water flow, nitrogen transport and root uptake including physical non-equilibrium and optimization of the root water potential. Fert. Res. 27:215–231.

Liang X. Q., Chen Y. X., Li H., Tian G. M., Ni W. Z., He M. M. and Zhang Z. J., 2007. Modeling transport and fate of nitrogen from urea applied to a near-trench paddy field. Environ. Pollut., 150(3), 313-320.

Maas EV, Grattan SR. 1999. Crop yields as affected by salinity. In Agricultural Drainage.Skaggs RW and Schilfgaarde J van (eds). Agronomy Monograph. ASA, CSSA, SSA: Madison, WI 38:55–108.

Maas EV, Hoffman GJ. 1977. Crop salt tolerance-current assessment. Journal of the Irrigation and Drainage Division 103(2): 115–134.

Maas EV. 1986. Salt tolerance of plants. Applied Agricultural Research 1: 12–26.

Maas, E.V., 1990., Crop salt tolerance. In: Tanji, K. (Ed.), Agricultural Salinity Assessment and Management. ASCE Manuals & Reports on Engineering Practice No. 71, ASCE, NY, pp. 262–304.

Mansell, R.S., Selim, H.M. and Fiskell, J.GA. 1977. Simulated transformations and transport of phosphorus in soil. Soil Sci. 124, 102–109.

Mansell, H. M. Selim, P. Kanchanasut, J. M. Davidson and J. G. A. Fiskell, 1977. Experimental and Simulated Transport of Phosphorus Through Sandy Soils. Water Resources Research, Vol. 13, NO. I, pg.189-194

MeGechan, M.B., and L. Wu. 2001. A review of carbon and nitrogen processes in European soil nitrogen dynamics models. p. 103–171. In M.J. Shaffer et al. (ed.) Modeling carbon and nitrogen dynamics for soil management. Lewis publishers, CRC Press, Boca Raton, FL.

Meiri, A., Shalhevet, J., 1973. Crop growth under saline conditions. In: Yaron, B., Vaadia, Y. (Eds.), Arid Zone Irrigation. Ecological Studies No. 5. Springer Verlag, Heidelberg, pp. 277–290.

Mollerup, M., P. Abrahamsen, C. T. Petersen, and S. Hansen (2014), Comparison of simulated water, nitrate, and bromide transport using a Hooghoudt-based and a dynamic drainage model, Water Resour. Res., 50, 1080–1094, doi:10.1002/2012WR013318.

Molz FJ, Remson I. 1970. Extraction term models of soil moisture use by transpiring plants. Water Resources Research 6(5): 1346–1356.

Mualem, Y., 1986. Hydraulic conductivity of unsaturated soils: Prediction and formulas, In: Klute, A. (Ed.), Methods of soil analysis, Part 1, Physical and mineralogical methods, 2nd ed. Agronomy, vol. 9(2). American Society of Agronomy, Madison, Wisconsin, pp. 799–823.

Parker and van Genuchten, 1984. Flux-Averaged and Volume-Averaged Concentrations in Continuum Approaches to Solute Transport. Water Resources Research 20(7):866-872

Picioreanu, C., van Loosdrecht, M. C. M. & Heijnen, J. J. (1997). Modelling the effect of oxygen concentration on nitrite accumulation in a biofilm airlift suspension reactor. Water Sci Technol 36(7), 147-156. Serve per I coefficient di diffusione di nitrato e ammonio in acqua

Prasad R. 1988. A linear root water uptake model. Journal of Hydrology, 99(3): 297–306.

Priesack, E, Durner W., 2006. Closed-Form Expression for the Multi-Modal Unsaturated Conductivity Function. Vadose Zone Journal 5:121–124).

Raats, PAC. 1974. Steady flows of water and salt in uniform soil profiles with plant roots. Soil Science Society of America Journal 38(5): 717–722.

Rathfelder K., and L. M. Abriola, 1994. Mass conservative numerical solutions of the head based Richards equation. Water Resour. Res., 30:2579–2586.

Ritchie, J.T., 1972. A model for predicting evaporation from a row crop with incomplete cover. Water Resour. Res., 8, 1204-1213.

Ross, P.J., Smettem, R.J., 1993. Describing soil hydraulic properties with sums of simple functions. Soil Science Society American Journal 57, 26–29.

Russo D. (1988). Determining soil hydraulic properties by parameter estimation: On the selection of a model for the hydraulic properties. Water Resour. Res, 24(3): 453-459.

Selim HM, Amacher MC 1997: Reactivity and transport of heavy metals in soils. Monogr. 90–2. U.S. Army Cold Regions Res. Eng. Lab., Hanover, NH.

Shalhevet, J., Hsiao, Th.C., 1986. Salinity and drought. Irrig. Sci. 7, 249–264.

Shouse PJ, Ayars JE, Šimůnek J. 2011. Simulating root water uptake from a shallow saline groundwater resource. Agricultural Water Management 98(5): 784–790.

Šimůnek, J., and Suarez, D. L. 1993. Modeling of carbon dioxide transport and production in soil: 1. Model development. Water Resources Research, 29(2), 487–497. https://doi.org/https://doi.org/10.1029/92WR02225

Skaggs TH, van Genuchten MTh, Shouse PJ, Poss JA. 2006a. Macroscopic approaches to root water uptake as a function of water and salinity stress. Agricultural Water Management 86: 140–149.

Smith, R.E., Diekkruger, B., 1996. Effective soil water characteristics and ensemble soil water profiles in heterogeneous soils. Water Resources Research 32 (7), 1993–2002. USSL Staff (1954) Diagnosis and improvement of saline and alkali soils. USDA Handbook No 60.

Spohn, M., 2015. Microbial respiration per unit microbial biomass depends on litter layer carbon-to-nitrogen ratio. Biogeosciences, 12, 817–823, 2015. doi:10.5194/bg-12-817-2015

Spohn, M., Klaus, K., Wanek, W., Richter, A. 2016. Microbial carbon use efficiency and biomass turnover times depending on soil depth – Implications for carbon cycling. Soil Biology and Biochemistry, 96, 74–81. https://doi.org/https://doi.org/10.1016/j.soilbio.2016.01.016

Stanford, G., Smith, S. J. 1972. Nitrogen Mineralization Potentials of Soils. Soil Science Society of America Journal, 36(3), 465–472. https://doi.org/https://doi.org/10.2136/sssaj1972.03615995003600030029x

Suarez, D. L., and Šimůnek, J. 1993. Modeling of carbon dioxide transport and production in soil: 2. Parameter selection, sensitivity analysis, and comparison of model predictions to field data. Water Resources Research, 29(2), 499–513. https://doi.org/https://doi.org/10.1029/92WR02226

van Dam, J.C., Huygen, J., Wesseling, J.G., Feddes, R.A., Kabat, P., van Walsum, P.E.V., Groenendijk, P., van Diepen, C.A., 1997. SWAP version 2.0, Theory. Simulation of water flow, solute transport and plant growth in the Soil–Water–Atmosphere–Plant environment. Technical Document 45, DLOWinand Staring Centre, Report 71, Department Water Resources, Agricultural University, Wageningen.

van der Molen, W. H., and J. Wesseling (1991), A solution in closed form and a series solution to replace the tables for the thickness of the equivalent layer in Hooghoudt's drain spacing formula, Agric. Water Manage., 19(1), 1991, doi:10.1016/0378-3774(91)90058-Q.

Van Der Zee, S. and Van Riemsdijk, W.H., Sorption kinetics and transport of phosphate in sandy soil, Geoderma, 38, 293-309, 1986

van Genuchten MTh, Hoffman GJ, 1984. Analysis of crop salt tolerance data. In Soil Salinity Under Irrigation – Process and Management. Shainberg I and Shalhevet J (eds). Ecological Studies 51, Springer-Verlag: New York; 258–271.

van Genuchten, M.T., 1987. A numerical model for water and solute movement in and below the root zone. USDA-ARS, US Salinity Laboratory, Riverside. Research Report no. 121.

van Genuchten, M.Th., and P.J. Wierenga, 1974. Simulation of one-dimensional solute transfer in porous media. New Mexico State University Agric. Exp. Stn. Bull. 628, New Mexico

van Genuchten, M.Th., 1980. A closed-form equation for predicting the hydraulic conductivity of unsaturated soils. Soil Science Society American Journal 44, 892–898 van Genuchten, M. Th. and Parker, J. C., 1984, Boundary conditions for displacement experiments through short laboratory soil columns,Soil Sci. Soc. Am. J. 48, 703–708

Vanderborght J, Vereecken H 2007: Review of dispersivities for transport modeling in soils. Vadose Zone J., 6, 29–52.

Vrugt JA., Hopmans JW, Šimunek J. 2001. Calibration of a twodimensional root water uptake model. Soil Science Society of America Journal 65 (4): 1027–1037. Washington DC, USA 160 pp

Wadleigh, C.H., 1946. The integrated soil moisture stress upon a root system in a large container of saline soil. Soil Sci. 61, 225–238.

Wallach, R. and M.T van Genuchten, 1990. A physically based model for predicting solute transfer from soil solution to rainfall induced runoff water. Water Resources Research, vol. 26, n.9, 2119-2126. DOI: 10.1029/WR026i009p02119

Watts, D. G., Hanks, R. J. 1978. A Soil-Water-Nitrogen Model for Irrigated Corn on Sandy Soils. Soil Science Society of America Journal, 42(3), 492–499. https://doi.org/https://doi.org/10.2136/sssaj1978.03615995004200030024x

Williams, J.R., Renard, K.G. and Dyke, P.T., 1983. EPIC: A new method for assessing erosion's effect on soil productivity. Journal of Soil and water Conservation, 38(5), pp.381-383.