

NOTA 1494 I

januari 1984

Instituut voor Cultuurtechniek en Waterhuishouding
Wageningen

ALTERRA
Wageningen Universiteit & Research centre
Omgevingswetenschappen
Centrum Water & Klimaat
Team Integraal Waterbeheer

FLOWEX: EEN NUMERIEK MODEL VOOR SIMULATIE VAN VERTICALE STROMING
VAN WATER DOOR ONVERZADIGDE GELAAGDE GROND

J. Buitendijk

Nota's van het Instituut zijn in principe interne communicatiemiddelen, dus geen officiële publikaties. Hun inhoud varieert sterk en kan zowel betrekking hebben op een eenvoudige weergave van cijferreeksen, als op een concluderende discussie van onderzoeksresultaten. In de meeste gevallen zullen de conclusies echter van voorlopige aard zijn omdat het onderzoek niet is afgesloten. Bepaalde nota's komen niet voor verspreiding buiten het Instituut in aanmerking

I N H O U D

	blz.
1. INLEIDING	1
2. MODELBESCHRIJVING	2
2.1. Onverzadigde stroming	2
2.2. Stroming tussen lagen met verschillende eigenschappen	5
2.3. Randvoorwaarde aan de bovenzijde	8
2.3.1. Neerslag en verdamping	8
2.3.2. Runoff	9
2.3.3. Infiltratie en bodemverdamping	9
2.4. Randvoorwaarde aan de onderzijde	10
2.4.1. Relatie drainafvoer-grondwaterstand voor uniforme profielen	10
2.4.2. Relatie drainafvoer-grondwaterstand voor gelaagde profielen	13
2.4.3. Grondwaterstand beneden drainniveau	14
2.5. Beginvoorwaarde	15
3. MODELINVOERGEGEVENS	15
3.1. Neerslag en verdamping	15
3.2. Bodemeigenschappen	16
3.2.1. $\psi(\theta)$ -relaties	16
3.2.2. $k(\psi)$ -relaties	16
3.3. Draindiepte en -intensiteit	18
3.4. Tijdstapgrootte	18

	blz.
4. BESCHRIJVING VAN HET PROGRAMMA FLOWEX	19
4.1. Hoofdprogramma FLOWEX	19
4.2. Subroutines	22
4.2.1. INIFLW	22
4.2.2. MDISTR	22
4.2.3. KPSI	22
4.2.4. INFIL	23
4.2.5. FLUX	23
4.2.6. ALGEM	23
4.2.7. ITERA	23
4.2.8. ALGEMI	24
4.2.9. GWST	24
4.2.10. GWS2	24
4.2.11. THETA	24
4.2.12. TETPSI	25
4.2.13. PSITET	25
4.2.14. SORT (IARRAY, LEN)	25
4.3. Input	25
4.3.1. INFO-file	25
4.3.2. Meteo-files	28
4.4. Outputfiles	28
5. LITERATUUR	29
6. PROGRAMMALIJST EN VOORBEEDEN VAN IN- EN UITVOER	
6.1. FORTRAN-tekst FLOWEX	30
6.2. INFO-file	58
6.3. METEO-file	60
6.4. OUTPUT-file	61

1. INLEIDING

De behoefte aan kennis van de invloed van neerslag en verdamping op de vochtomstandigheden van de grond is groot. Bodemeigenschappen zoals bewerkbaarheid en draagkracht voor berijden, beweiden en bespenen zijn direct aan de vochtomstandigheden gekoppeld. Bij verbeteringsmaatregelen zoals drainage of het losmaken van gronden is steeds de centrale vraag in welke mate het vocht- en/of luchtgehalte hierdoor worden beïnvloed. Ook in andere vakgebieden is er een toenemende behoefte aan een goede beschrijving van het vochttransport door de grond. Zo is de dynamiek van vele bodemcomponenten nauw gekoppeld aan de dynamiek van het bodemvocht. Vandaar dat bij de beschrijving van de beweging van bodemcomponenten de beschrijving van stroming van water door de grond veelal centraal staat. Daarbij komt nog dat bij bestudering van processen die door het klimaat worden beïnvloed lange waarnemingsreeksen nodig zijn om betrouwbare resultaten te krijgen. Langjarige experimenten hebben niet alleen het bezwaar dat ze veel tijd en geld kosten maar ook dat de resultaten vaak beperkt zijn en niet overdraagbaar naar andere omstandigheden.

In het hedendaagse onderzoek wordt daarom steeds vaker van modellen gebruik gemaakt. Deze nota beschrijft het model FLOWEX dat de stroming van water door onverzadigde grond simuleert en daarbij tevens geschikt is voor het doorrekenen van lange tijdsreeksen. FLOWEX is gebaseerd op dezelfde geïntegreerde stromingsvergelijking als gebruikt in het model FLOW van WIND en VAN DOORNE (1975). FLOWEX is echter uitgebreider (extended) en verschilt van FLOW vooral hierin, dat FLOWEX:

1. toepasbaar is op niet alleen uniforme maar ook gelaagde bodemprofielen;
2. ook bij aanzienlijk drogere bodemomstandigheden van toepassing is;
3. voorzien is van een andere (niet iteratieve) grondwaterstandsberkening.

Omdat FLOWEX op dezelfde wijze geïntegreerde stromingsvergelijking is gebaseerd als FLOW heeft het ook dezelfde voordelen die FLOW kenmerken, zoals een goed sluitende waterbalans, grote rekensnelheid en eenvoudig te programmeren (WIND en VAN DOORNE, 1975; RICHTER, 1980; PAUL, 1978). Hoewel FLOWEX een model is voor onbegroeide grond kan het ook worden gebruikt voor begroeide grond zolang geen vochtonttrekking door wortels uit diepere bodemlagen optreedt. De in de programma-lijst (par. 4.5.) gegeven versie van FLOWEX heeft betrekking op een twee-lagenprofiel. Bovendien wordt geen rekening gehouden met kwel of wegzijging aan de onderkant. Met de in hoofdstuk 2 gegeven theorie zijn deze aanvullingen eenvoudig te realiseren.

Inmiddels is met FLOWEX ruime ervaring opgedaan in het HELP-onderzoek. Uit uitvoerige vergelijking van modelresultaten met veldmetingen van drukhoogte van het bodemvocht en van grondwaterstand blijkt FLOWEX de fluctuaties van deze grootheden in de tijd goed te beschrijven. Bovendien blijken lange tijdreeksen zonder problemen verwerkt te kunnen worden.

Opmerking: Sinds enige tijd wordt aanbevolen voor de drukhoogte van het bodemvocht het symbool h te gebruiken.

In het aan deze nota toegevoegde programma van FLOWEX wordt de drukhoogte echter nog met het symbool ψ aangeduid. Vandaar is voor het gemak van de lezer het symbool ψ voor de drukhoogte ook in de modelbeschrijving gehanteerd.

2. MODELBESCHRIJVING

2.1. Onverzadigde stroming

De stroming van water door de grond gebeurt bijna altijd onder niet-stationaire omstandigheden. Stroomsnelheid, vochtgehalte en drukhoogte van het bodemvocht variëren niet alleen met de diepte maar ook in de tijd. De twee basisvergelijkingen die voor stroming van water door onverzadigde grond worden gebruikt zijn de:

Darcy stromingsvergelijking:

$$v = - k(\psi) \left(\frac{\partial \psi}{\partial z} + 1 \right) \quad (1)$$

Continuïteitsvergelijking:

$$\frac{\partial \theta}{\partial t} = - \frac{\partial v}{\partial z} \quad (2)$$

Combinatie van de vergelijkingen (1) en (2) geeft voor de één-dimensionale verticale stroming:

$$\frac{\partial \theta}{\partial t} = \frac{\partial}{\partial z} \left[k(\psi) \left(\frac{\partial \psi}{\partial z} + 1 \right) \right] \quad (3)$$

waarin: v = vert. vol. flux negatief in neerwaartse richting ($\text{cm}^3 \cdot \text{cm}^{-2} \cdot \text{d}^{-1}$)
 θ = volume vochtgehalte ($\text{cm}^3 \cdot \text{cm}^{-3}$)
 ψ = drukhoogte van het bodemvocht (cm)
 k = doorlatendheid afhankelijk van ψ ($\text{cm} \cdot \text{d}^{-1}$)
 z = verticale coördinaat (cm)
 t = tijd (d)

Vergelijking (3) is een partiële differentiaal vergelijking die niet lineair is vanwege het afhankelijk zijn van k en ψ van θ .

Uitgaande van een exponentieel verband tussen k en ψ zoals voorgesteld door RIJTEMA (1965):

$$k = k_0 e^{\alpha \psi} \quad (4)$$

waarin: k_0 = doorlatendheid bij $\psi=0$ ($\text{cm} \cdot \text{d}^{-1}$)
 α = bodemconstante (cm^{-1})

geven WIND en VAN DOORNE (1975) een geïntegreerde oplossing van de Darcy-stromingsvergelijking (1):

$$v = - \frac{k_i - k_{i-1}}{e^{\alpha \Delta z} - 1} - k_{i-1} \quad (5)$$

waarin: Δz = afstand tussen de middens van 2 lagen (cm)
 k_{i-1} resp. k_i = doorlatendheid van boven- respectievelijk onderliggende laag ($\text{cm} \cdot \text{d}^{-1}$)

Voor de afleiding van vergelijking (5) is gesteld dat ψ is een continue functie van z en dat v is constant gedurende de tijdstap en over het diepte interval.

Beginnend met een $\theta(z)$ -profiel van de grond wordt met behulp van de $\psi(\theta)$ -relatie (= pF-curve) een $\psi(z)$ -profiel verkregen. Met behulp van vergelijking (4) wordt uit het $\psi(z)$ -profiel de bijbehorende $k(z)$ -verdeling bepaald. Vervolgens worden met vergelijking (5) de stroomsnelheden tussen de middens van de lagen over de diepte berekend. Overeenkomstig de continuïteitsvergelijking (2) wordt na iedere tijdstap de vochtinhaltsverandering van elke laag gedurende die tijdstap berekend met:

$$\Delta \theta_i = (v_i - v_{i-1}) \frac{\Delta t}{\Delta z} \quad (6)$$

waarin: $\Delta \theta$ = vochtinhaltsverandering ($\text{cm}^3 \cdot \text{cm}^{-3}$)
 v_i resp. v_{i+1} = flux uit de boven- respectievelijk de onderliggende laag

Het nieuwe $\theta(z)$ -profiel aan het begin van de volgende tijdstap wordt verkregen uit:

$$\theta(i, t) = \theta(i, t-1) + \Delta \theta_i \quad (7)$$

De gang van deze procedure is in figuur 1 samengevat.

waarin: indices 4 en 5 verwijzen naar de laag boven respectievelijk
 onder het grensvlak

$$k_t = k \text{ op het grensvlak}$$

$$a = e^{\alpha \frac{1}{2} \Delta z}$$

Mits ψ op het grensvlak bekend is, kan k_t met vergelijking (4)
 worden berekend. Invulling van vergelijking (4) in vergelijking (8)
 geeft:

$$v = \frac{k_{o4} e^{\alpha_4 \psi} - k_4}{a_4 - 1} - k_4 = \frac{k_5 - k_{o5} e^{\alpha_5 \psi}}{a_5 - 1} - k_{o5} e^{\alpha_5 \psi} \quad (9)$$

met $e_4 = e^{\alpha_4 \psi}$ en $e_5 = e^{\alpha_5 \psi}$ kan vergelijking (9) worden herschreven:

$$\frac{k_{o4} e_4 - k_4}{a_4 - 1} - \frac{k_4 (a_4 - 1)}{(a_4 - 1)} = \frac{k_5 - k_{o5} e_5}{a_5 - 1} - \frac{k_{o5} e_5 (a_5 - 1)}{(a_5 - 1)} \quad (10)$$

$$\frac{k_{o4} e_4 - k_4 a_4}{a_4 - 1} = \frac{k_5 - a_5 k_{o5} e_5}{a_5 - 1} \quad (11)$$

$$\frac{k_{o4}}{a_4 - 1} e_4 + \frac{-k_4 a_4}{a_4 - 1} = \frac{k_5}{a_5 - 1} + \frac{-a_5 k_{o5}}{a_5 - 1} e_5 \quad (12)$$

$$\beta_1 \quad \beta_2 \quad \gamma_1 \quad \gamma_2$$

$$u = \beta_1 e_4 + \beta_2 \quad w = \gamma_1 + \gamma_2 e_5 \quad (13)$$

Met u en w zijn een dalende respectievelijk stijgende functie
 verkregen. Het snijpunt van deze functies geeft de gezochte waarde van
 ψ op het grensvlak. Dit snijpunt kan worden benaderd met behulp van
 iteraties volgens Newtons' methode van de eerste afgeleide (fig. 2).

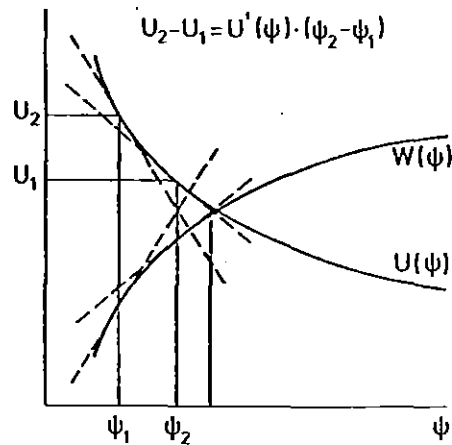


Fig. 2. Benadering van het snijpunt van de functies $u(\psi)$ en $w(\psi)$ met behulp van iteraties volgens Newtons' methode van de eerste afgeleide

Met:

$$\frac{d}{d\psi} e_4 = \alpha_4 e_4 \quad \text{en} \quad \frac{d}{d\psi} e_5 = \alpha_5 e_5 \quad \text{kunnen}$$

de functies u en w overeenkomstig de algemene vergelijking voor de raaklijn

$$y_2 = f'(x) \cdot (x_2 - x_1) + y_1 \quad (14)$$

worden geschreven als

$$u_2 = \alpha_4 \beta_1 e_4 (\psi_2 - \psi_1) + u_1 \quad w_2 = \alpha_5 \gamma_2 e_5 (\psi_2 - \psi_1) + w_1 \quad (15)$$

In het snijpunt van de raaklijnen geldt:

$$u(\psi) = w(\psi) \quad (16)$$

dus:

$$\alpha_4 \beta_1 e_4 (\psi_2 - \psi_1) + u_1 = \alpha_5 \gamma_2 e_5 (\psi_2 - \psi_1) + w_1 \quad (17)$$

$$\alpha_4 \beta_1 e_4 (\psi_2 - \psi_1) - \alpha_5 \gamma_2 e_5 (\psi_2 - \psi_1) = w_1 - u_1 \quad (18)$$

$$\psi_2 = \frac{w_1 - u_1}{\alpha_4 \beta_1 e_4 - \alpha_5 \gamma_2 e_5} + \psi_1 \quad (19)$$

Bij de eerste iteratie wordt voor ψ_1 in vergelijking (19) het gemiddelde van de ψ -waarden in de laag direct boven en onder de grenslaag ingevoerd. De iteratie wordt gestopt indien $\psi_2 - \psi_1 < 0,0001$. Dit wordt meestentijds al na 2 tot 3 iteraties bereikt.

2.3. Randvoorwaarde aan de bovenzijde

Voor het bodemoppervlak geldt de volgende waterbalans:

$$\Delta S = (P - E) - R - v_o \quad (20)$$

waarin: S = oppervlakteberging

P = neerslag

E = potentiële verdamping

R = runoff

v_o = flux door het bodemoppervlak

2.3.1. Neerslag en verdamping

De dikte van de oppervlakteberging wordt aan het begin van iedere tijdstap (Δt) berekend volgens:

$$S_t = S_{t-\Delta t} + (P - E) \Delta t \quad (21)$$

Zolang $S_t > 0$ treedt infiltratie op.

Indien $S_t < 0$ wordt doordat op het moment van toevoer van de verdamping $S_{t-\Delta t} = 0$ of $S_{t-\Delta t} < (P - E) \Delta t$ dan is er sprake van een flux naar het oppervlak toe ($v_o > 0$). Door uitdroging van de oppervlaktelaag neemt de doorlatendheid af en wordt de bodemverdamping sterk gereduceerd (mulch-effect).

Dit wordt in het model gerealiseerd door de toegevoerde verdamping te reduceren in afhankelijkheid van de drukhoogte van het bodemvocht in het midden van de bovenste laag volgens:

$$E_r = E (1 + b\psi) \quad (22)$$

waarin: E_r = gereduceerde verdamping

b = bodemparameter

ψ = drukhoogte bodemvocht in midden van bovenste laag

2.3.2. Runoff

Voor berekening van de runoff zijn twee opties mogelijk:

1. runoff ontstaat wanneer een tevoren op te geven dikte van de oppervlakteberging (S_{\max}) wordt overschreden volgens:

$$R = S - S_{\max} \quad (23)$$

waarna voor de volgende tijdstap $S = S_{\max}$ wordt gesteld;

2. runoff wordt als functie van de plasdikte berekend:

$$R = f(S) \quad (24)$$

WIND en VANDENBERG (1983) gebruikten hiervoor de relatie

$$R = c(S)^2 \quad (25)$$

waarbij een lage waarde van de evenredigheidsconstante $c(\text{cm}^{-1} \cdot \text{d}^{-1})$ een hoge weerstand voor de oppervlaktestroming in rekening brengt.

2.3.3. Infiltratie en bodemverdamping

Zolang $S > 0$ treedt infiltratie op. Ter plaatse van het bodemoppervlak geldt dan $\psi = 0$ en daarom $k = k_o \cdot v_o$, dat is de neerwaartse flux vanaf het bodemoppervlak naar het midden van de bovenste laag, kan dan worden berekend met vergelijking (5), met invulling van de halve laagdikte voor Δz (zie fig. 1).

Voor berekening van de opwaartse flux (bodemverdamping) zou in principe vergelijking (5) ook kunnen worden toegepast door evenwicht te veronderstellen tussen de drukhoogte van het bodemvocht aan het oppervlak en de atmosfeer. Hierbij kan ψ aan het oppervlak afhankelijk van temperatuur en luchtvochtigheid waarden krijgen in de orde van $-10^4 > \psi > 10^6$ cm. Dit is echter een bereik waarvoor vergelijking (4) niet meer geldt.

Vandaar dat in geval van een flux naar het bodemoppervlak voor het midden van de bovenste laag wordt gesteld:

$$v_o = E_r \quad (26)$$

2.4. Randvoorwaarde aan de onderzijde

2.4.1. Relatie drainafvoer-grondwaterstand voor uniforme profielen

De randvoorwaarde aan de onderzijde van het model wordt gegeven door de drainage-afvoerfunctie. Volgens HOOGHOUTD (1940) en ERNST (1956) geldt voor de hoogte van de grondwaterstand halverwege twee evenwijdige drainreeksen (zie fig. 3).

$$h_m - h_o = -v_D R_{\text{vert}} - v_D (R_{\text{hor}} + R_{\text{rad}}) \quad (27)$$

waarin: v_D = drainafvoer ($\text{cm} \cdot \text{d}^{-1}$)
 h_m = potentiaal ter plaatse van de grondwaterstand (m)
 h_o = potentiaal ter plaatse van het waterniveau in de drains (m)
 R_{vert} = weerstand voor verticale stroming (d)
 R_{hor} = weerstand voor horizontale stroming (d)
 R_{rad} = weerstand voor radiale stroming (d)

$$R_{\text{vert}} = \frac{h_m - h_o}{k_o} \quad (28)$$

$$(R_{\text{hor}} + R_{\text{rad}}) = \frac{L^2}{8k_o d} \quad (29)$$

waarin: k_o = verzadigde doorlatendheid ($\text{m} \cdot \text{d}^{-1}$)
 L = drainafstand (m)
 d = dikte van de zogenaamde equivalentlaag (m)

Substitutie van verg. (28) en (29) in (27) geeft:

$$h_m - h_o = -v_D \left(\frac{h_m - h_o}{k_o} \right) - v_D \frac{L^2}{8k_o d} \quad (30)$$

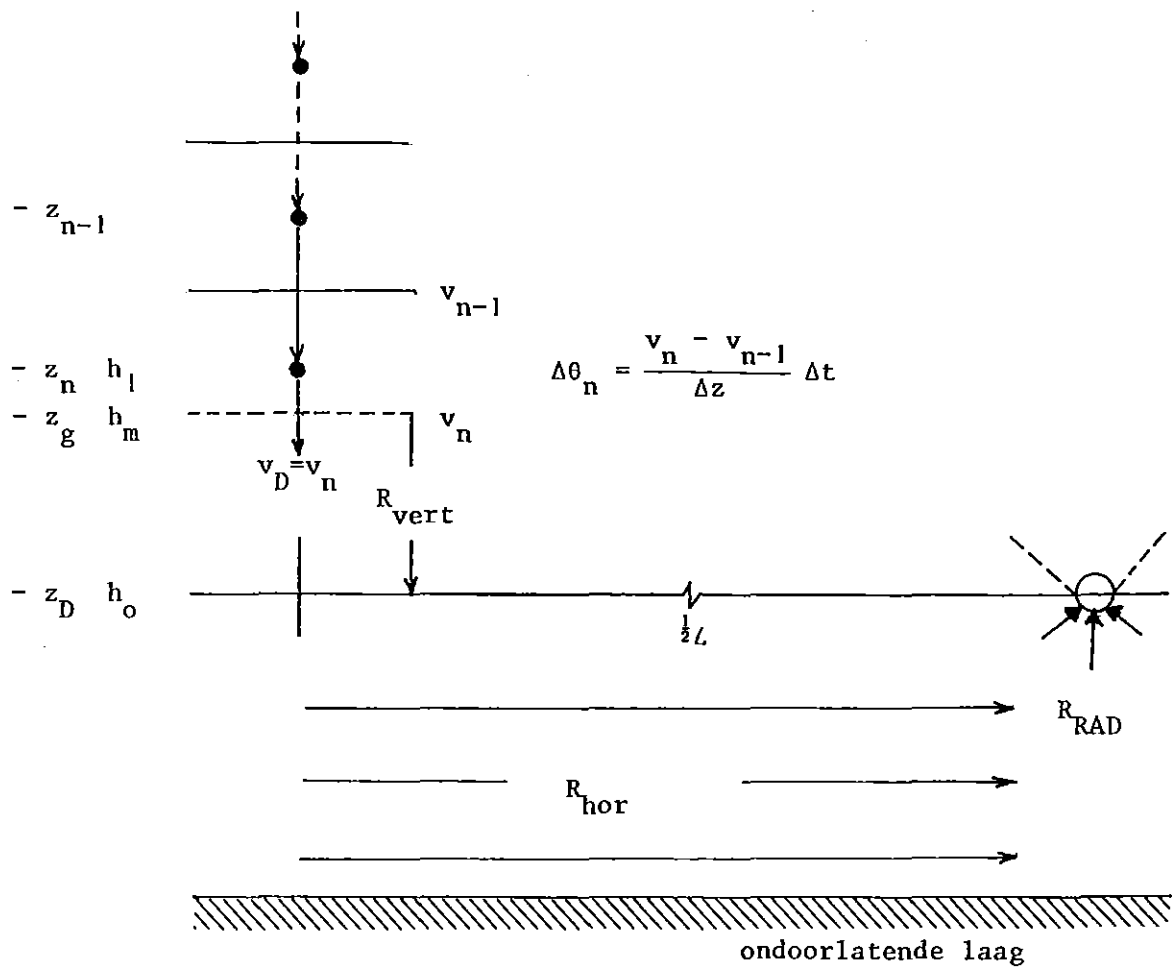


Fig. 3. Afvoer van evenwijdige drains gesplitst in verticale, horizontale en radiale stroming en de verandering van het vochtgehalte in de diepste onverzadigde laag als gevolg hiervan

In vergelijking (30) is $8k_o d/L^2 = A$ de drainageintensiteit ($d-l$).
 Met $h_m - h_o = h_g$ wordt vergelijking (30):

$$h_g = -v_D \left(\frac{h_g}{k_o} + \frac{l}{A} \right) \quad (31)$$

of expliciet naar v_D :

$$v_D = - \frac{l}{\frac{l}{k_o} + \frac{l}{Ah_g}} \quad (32)$$

De drainafvoer v_D is bovendien gelijk aan de flux lopend van het centrum van de onderste onverzadigde laag (op een hoogte boven drainniveau $h_1 = z_n - z_D$: zie fig. 3) naar de grondwaterstand. Hiervoor geldt:

$$v_D = - \bar{k} \left(\frac{\psi_n - \psi_g}{h_1 - h_g} + 1 \right) \quad (33)$$

waarin: \bar{k} = rekenkundig gemiddelde van doorlatendheden in het centrum van de onderste onverzadigde laag en ter plaatse van de grondwaterstand waar $k = k_o$ (cm.d^{-1})
 ψ_n en ψ_g = drukhoogte ter plaatse van z_n respectievelijk grondwaterstand (cm)

Omdat $\psi_g = 0$, zijn met de vergelijkingen (32) en (33) twee vergelijkingen met twee onbekenden, v_D en h_g , verkregen, die met $\frac{1}{A} = A^*$ (= drainageweerstand (d)) aldus worden opgelost.

$$\frac{1}{\frac{1}{k_o} + \frac{A^*}{h_g}} = \bar{k} \left(\frac{\psi_n}{h_1 - h_g} + 1 \right) \quad (34)$$

Uitwerking geeft:

$$h_g^2 - h_g \left(h_1 + \frac{A^* \bar{k} k_o - \bar{k} \psi_1}{k_o - \bar{k}} \right) + A^* \bar{k} k_o \frac{h_1 + \psi_1}{k_o - \bar{k}} = 0 \quad (35)$$

Wordt $\frac{\bar{k}}{k_o - \bar{k}} = t$ gesteld dan wordt de volgende vierkantsvergelijking gevonden waaruit h_g als wortel kan worden opgelost.

$$h_g^2 - h_g (h_1 + t (A^* k_o - \psi_1)) + A^* k_o t (h_1 + \psi_1) = 0 \quad (36)$$

Met bekende h_g wordt v_D verkregen na herschrijving van vergelijking (32) uit:

$$v_D = - \frac{h_g}{\frac{h_g}{k_o} + A^*} \quad (37)$$

2.4.2. Relatie drainafvoer-grondwaterstand voor gelaagde profielen

Vergelijking (37) kan worden toegepast zolang h_g niet groter wordt dan de dikte van de laag waarin de drainage ligt. In geval van profielen bestaande uit lagen met verschillende doorlatendheden (k_o 's), dan moet hiermee rekening worden gehouden bij berekening van de weerstand voor verticale stroming. Voor een twee-lagen profiel met de grondwaterstand in de bovenste laag (fig. 4) verandert vergelijking (31) in:

$$h_g = -v_D \left(\frac{l}{A} + \frac{D}{k_{o1}} + \frac{h_g - D}{k_{o2}} \right) \quad (38)$$

of met $B = \frac{D}{k_{o1}} - \frac{D}{k_{o2}}$ en $\frac{l}{A} = A^*$

$$h_g = -v_D \left(A^* + B + \frac{h_g}{k_{o2}} \right) \quad (39)$$

waarin: D = afstand tussen draaindiepte en overgang tussen de twee lagen (cm)

k_{o1} resp. k_{o2} = doorlatendheid van onderste respectievelijk bovenste laag ($m \cdot d^{-1}$)

Expliciet naar v_D geschreven wordt vergelijking (37):

$$v_D = - \frac{h_g}{\frac{h_g}{k_{o1}} + A^* + B} \quad (40)$$

Voor profielen bestaande uit 3 lagen met de grondwaterstand in de onderste respectievelijk middelste laag moet voor berekening van v_D vergelijking (37) respectievelijk (40) worden toegepast. In geval van de grondwaterstand in de bovenste laag wordt in vergelijking (40) de k_o van de bovenste laag ingevoerd en B aangepast volgens (zie fig. 4):

$$B = \frac{D_1}{k_{o1}} + \frac{D_2}{k_{o2}} - \frac{D_1 + D_2}{k_{o3}} \quad (41)$$

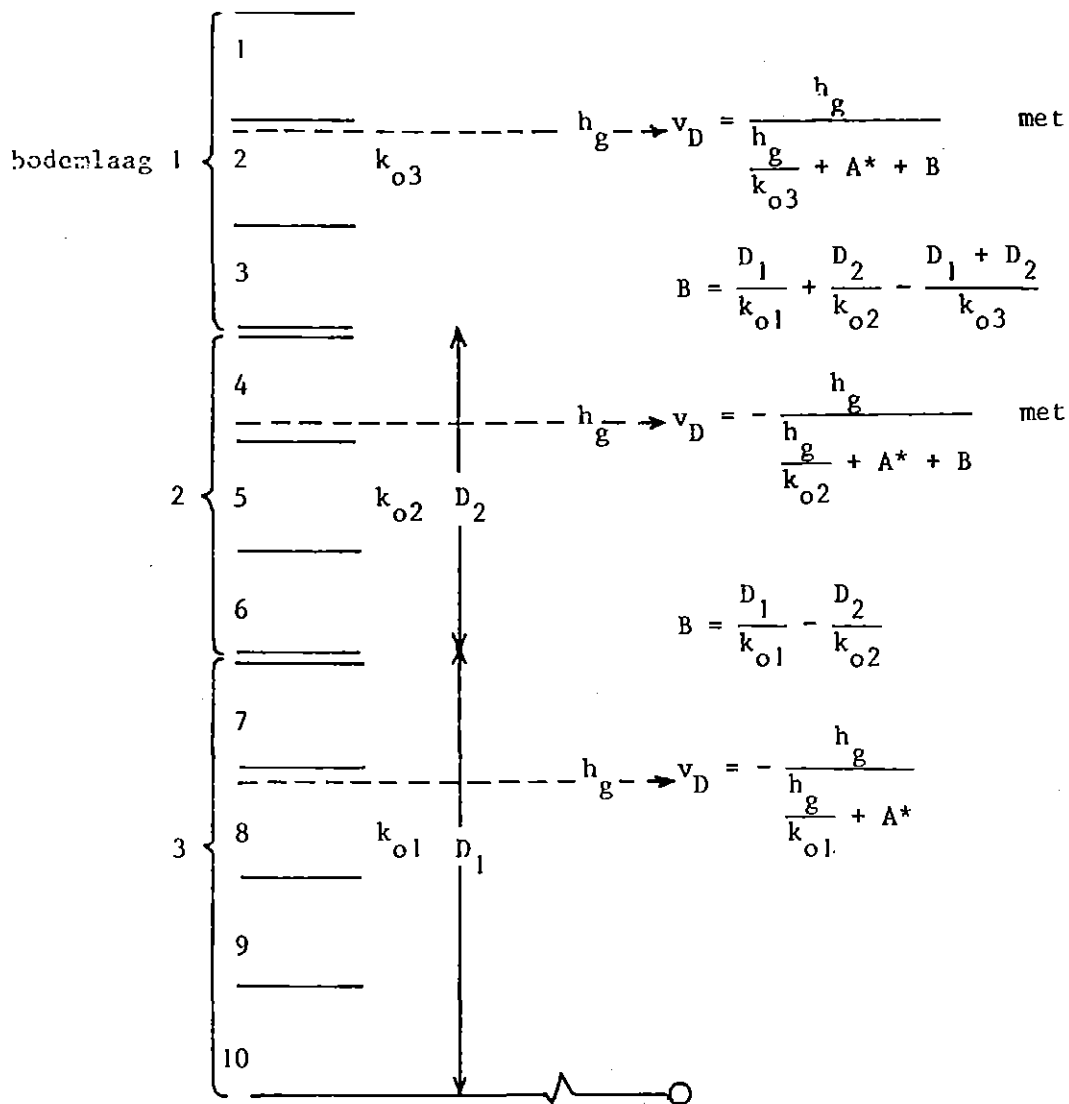


Fig. 4. Berekening van de drainafvoer uit gelaagde profielen in afhankelijkheid van de verticale weerstand en hoogte van de grondwaterstand

2.4.3. Grondwaterstand beneden drainniveau

Ten gevolge van capillaire opstijging kan de grondwaterstand beneden drainniveau dalen en wordt $v_D = 0$. In dat geval wordt aangenomen dat de drukhoogte in het midden van de diepste onverzadigde laag boven de grondwaterstand gelijk is aan de afstand van dit punt tot de grondwaterstand dus:

$$h_g = z_n - \psi_n \quad (42)$$

waarin: h_g = diepte grondwaterstand (cm)
 z_n = diepte midden van onderste onverzadigde laag (cm)
 ψ_n = drukhoogte in z_n (cm)

2.5. B e g i n v o o r w a a r d e

Als beginvoorwaarde kunnen verdelingen met de diepte van zowel vochtgehalte, drukhoogte, doorlatendheid als stroomsnelheid worden opgegeven. Om een en ander te vergemakkelijken is in FLOWEX de mogelijkheid ingebouwd het drukhoogteprofiel te berekenen bij een met de diepte constante stroomsnelheid. De berekening van het drukhoogteprofiel start bij de grondwaterstand, waar $\psi=0$. Bij opgegeven v wordt daarom eerst de grondwaterstand berekend met vergelijking (31) (par. 2.4.1.) of bij meer-lagen profielen afhankelijk van de hoogte van de grondwaterspiegel met vergelijkingen van het type (39).

Combinatie van de Darcy stromingsvergelijking (verg. 1) met vergelijking (4) geeft na integratie (VAN WIJK, 1980):

$$\psi_i = \frac{1}{\alpha} \ln \left[\frac{v}{k_o} (e^{\alpha \Delta z} - 1) + e^{(\psi_{i+1} + \Delta z)} \right] \quad (43)$$

waarin: ψ_i resp. ψ_{i+1} = drukhoogte in midden van boven- respectievelijk onderliggende laag (cm)
 Δz = afstand tussen middens van de twee lagen (cm)

Wanneer de doorlatendheidsconstanten k_o en α bekend zijn kan bij elke willekeurige v voor zowel homogene als gelaagde profielen met behulp van vergelijking (43) het drukhoogteprofiel boven de grondwaterstand worden berekend.

3. MODELINVOERGEGEVENS

3.1. N e e r s l a g e n v e r d a m p i n g

Regen en verdamping kunnen op elke willekeurige tijdsbasis worden ingevoerd. Regen op dagbasis wordt in het model beschouwd als gelijkmatig over de dag gevallen te zijn. Dagelijkse verdamping wordt in het model ingevoerd als negatieve regen voorkomend tussen 10 en 15 uur.

3.2. B o d e m e i g e n s c h a p p e n

3.2.1. $\psi(\theta)$ -relaties

De $\psi(\theta)$ -relatie (pF-curve) wordt in tabelvorm opgegeven in stappen van hele volumeprocenten.

Tussenliggende waarden worden in het model via interpolatie verkregen. Voor bepaling van de tijdstapgrootte is vooral de helling $dk/d\theta$ van belang (zie par. 3.4.) Vandaar is het raadzaam de $\psi(\theta)$ -relatie in het hele natte traject in stappen van halve volumeprocenten op te geven.

3.2.2. $k(\psi)$ -relaties

Omdat de in het model toegepaste geïntegreerde stromingsvergelijking uitgaat van een exponentiële relatie tussen k en ψ moet de $k(\psi)$ -relatie overeenkomstig vergelijking (4) als een exponentiële functie worden ingevoerd. Op dit punt is FLOWEX een uitbreiding en daarmee een verbetering van het model FLOW van WIND en VAN DOORNE (1975). FLOW gaat uit van één exponentiële $k(\psi)$ -relatie. Hierdoor is FLOW slechts toe te passen in een vrij smal traject van hoge (= natte) drukhoogten. Dit traject wordt smaller naarmate de grond zwaarder is.

Om aan dit bezwaar tegemoet te komen wordt in FLOWEX, in navolging van RICHTER (1980) de $k(\psi)$ -relatie ingevoerd als een serie $k_0 e^{\alpha\psi}$ -lijnstukken. In de huidige versie kunnen dit er maximaal drie zijn.

Hiermee kan elke vorm van de $k(\psi)$ -relatie goed worden benaderd over een zeer breed traject van drukhoogten (fig. 5). Voor elk $k_0 e^{\alpha\psi}$ -lijnstuk worden k_0 , α en het traject van drukhoogten waarvoor het bewuste segment geldt opgegeven.

Een complicatie kan zich voordoen bij de berekening van v met vergelijking (5). v is de flux die loopt tussen de middens van 2 lagen. Indien de ψ 's in deze 2 lagen zodanig verschillen dat voor berekening van k twee verschillende $k_0 e^{\alpha\psi}$ -lijnstukken, dus twee verschillende k_0 's en α 's moeten worden gebruikt dan moet in de term $(e^{\alpha\Delta z} - 1)$ in vergelijking (5) met een gemiddelde α worden gewerkt. Voor berekening van de gemiddelde α worden bij de heersende ψ -waarden de bijbehorende α 's gezocht (zie fig. 5) en wordt de gemiddelde α ($\bar{\alpha}$) volgens onderstaande betrekking bepaald.

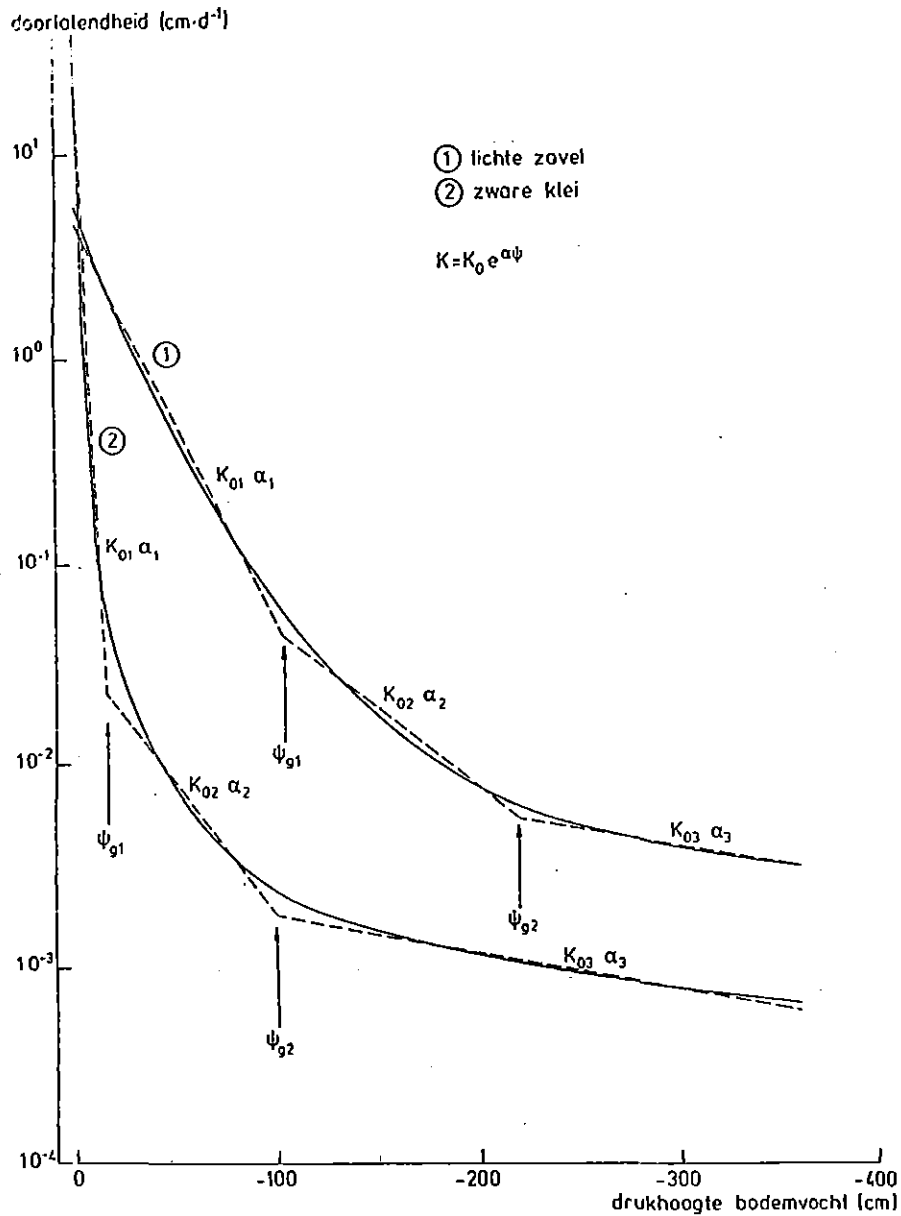


Fig. 5. Doorlatendheidskarakteristieken benaderd met een serie $k_0 e^{\alpha \psi}$ -lijnstukken

$$\bar{\alpha} = \frac{(\psi_1 - \psi_{g1}) \alpha_1 + (\psi_{g1} - \psi_{g2}) \alpha_2 + (\psi_{g2} - \psi_2) \alpha_3}{\psi_1 - \psi_2} \quad (44)$$

- waarin: ψ_1 resp. ψ_2 = de hoogste (natste) respectievelijk laagste drukhoogte (cm)
- ψ_{g1} resp. ψ_{g2} = snijpunt van eerste met tweede respectievelijk tweede met derde lijnstuk (cm)
- $\alpha_1, \alpha_2, \alpha_3$ = α 's behorend bij de respectievelijke lijnstukken (cm⁻¹)

Vergelijking (44) heeft betrekking op het geval waarbij ψ_1 en ψ_2 op niet-aangrenzende lijnstukken liggen. Liggen ψ_1 en ψ_2 op aangrenzende lijnstukken dan wordt $\psi_{g2} = \psi_{g1}$ gemaakt en bijbehorende α 's en ψ_g waarden gezocht.

3.3. D r a i n d i e p t e e n - i n t e n s i t e i t

Elke grond heeft een ontwateringsbasis en afvoerintensiteit, anders zou zo'n grond in ons klimaat onder water staan. Hiermee wordt in het model rekening gehouden door een draindiepte en -intensiteit in te voeren. In principe kan elke willekeurige draindiepte worden ingevoerd. In combinatie met de drainagediepte moet om aan de randvoorwaarde aan de onderkant te voldoen een drainageintensiteit (zie hiervoor par. 2.4.1.) worden gegeven.

3.4. T i j d s t a p g r o o t t e

In FLOWEX wordt gediscretiseerd naar de diepte (Δz) en de tijd (Δt). Naarmate Δz en Δt groter zijn zullen de rekentijd en daarmee de rekenkosten lager zijn, hetgeen doorslaggevend kan zijn bij het doorrekenen van lange tijdreeksen. De keuze van de stapgrootte wordt echter beperkt door de stabiliteitsvoorwaarde. Het blijkt namelijk dat bij te grote Δt 's de rekenresultaten gaan oscilleren. De amplitude van deze oscillaties wordt steeds groter bij volgende tijdstappen uiteindelijk uitmondend in 'fatal errors', waarbij de computer het rekenproces zal stoppen. Ter voorkoming hiervan wordt in het begin van het programma de maximaal toelaatbare Δt berekend met behulp van het door WIND en VAN DOORNE (1975) gegeven stabiliteitscriterium:

$$\Delta t < \frac{d\theta}{dk} \cdot \frac{e^{\alpha\Delta z} - 1}{e^{\alpha\Delta z} + 1} \cdot \Delta z \quad (45)$$

Hieruit blijkt dat Δt groter mag zijn naarmate Δz groter en $dk/d\theta$ kleiner is. De grootste veranderingen van k met θ doen zich voor in het hele natte traject. De eerste instabiliteit wordt dan ook waargenomen in de lagen direct boven de grondwaterstand. Omdat $dk/d\theta$ wordt bepaald door het beloop in het uiterst natte traject van zowel de doorlatend-

heidskarakteristiek als de pF-curve, kunnen kleine wijzigingen hierin resulteren in een sterke toename van Δt . Experimenteren hiermee liet zien dat deze kleine wijzigingen niet altijd ten koste van de nauwkeurigheid van het rekenresultaat behoeven te gaan.

Berekening van het stabiliteitscriterium gebeurt op basis van het eerste (natste) lijnstuk van de $k(\psi)$ -lijnstukkencurve (zie fig. 5). In geval van niet-uniforme bodemprofielen wordt een maximaal toelaatbare Δt berekend op basis van de $k(\psi)$ en $k(\theta)$ relatie van elk van de bodemmaterialen. De als kleinste gevonden tijdstapgrootte wordt bij de modelberekeningen toegepast.

In de praktijk blijkt bovenstaand stabiliteitscriterium zeer goed te voldoen. Zolang Δt niet groter gekozen wordt dan de met vergelijking (45) berekende waarde treedt geen instabiliteit op. Waarden van Δt slechts weinig boven het stabiliteitscriterium geven wel instabiliteit.

4. BESCHRIJVING VAN HET PROGRAMMA FLOWEX

4.1. H o o f d p r o g r a m m a F L O W E X

Het totale pakket bestaat uit het hoofdprogramma FLOWEX en een aantal subroutines geschreven in FORTRAN-IV.

FLOWEX bestaat in feite uit een aantal DO-loops waarbinnen zich het gehele rekenproces afspeelt:

```
DO 10  IDRAIN = 1,NDRAIN  ! aantal draindiepten
DO 20  IDRINT = 1,NDRINT  ! aantal drainintensiteiten
DO 30  IJAAR = 1,NJAAR    ! aantal jaren/reeksen
DO 40  I = 1,NPARTS      ! aantal dagdelen
DO 50  ISTEP = 1,NSTEPS  ! aantal tijdstappen binnen één dagdeel

CALL  SUBR 1
CALL  SUBR 2
:
:
:
enz.
```

```
50 CONTINUE  
    EVENTUEEL OUTPUT  
40 CONTINUE  
    NIEUWE INPUTFILE 'METEO'  
30 CONTINUE  
    NIEUWE INPUTFILE 'INFO'  
20 CONTINUE  
10 CONTINUE  
    STOP
```

De waarden van NDRAIN, NDRINT en NJAAR worden in een keer vanuit de file INFO01DAT (zie 4.3.1.) gelezen. NPARTS wordt gelezen vanaf de METEO-files (zie 4.3.2.). NSTEPS bevat het aantal tijdstappen binnen 1 dagdeel (= 0,2 dag). Een dagdeel hoeft echter niet een veelvoud van de tijdstapgrootte te zijn. In dat geval wordt de laatste afwijkende tijdstap (R) berekend volgens:

$$R = 0,2 - (NSTEPS \times \Delta t) \quad (46)$$

Na het aantal van NSTEPS-berekeningen dat wil zeggen aan het einde van elk dagdeel wordt de gewenste output naar de outputfile geschreven. Is DO-loop 40 voltooid, dat wil zeggen het per reeks (= jaar of deel van het jaar) opgegeven aantal dagdelen is doorgerekend, dan wordt de gebruikte METEO-file gesloten en de volgende geopend en gelezen. Is DO-loop 30 voltooid, dat wil zeggen het opgegeven aantal jaren is doorgerekend dan wordt de volgende INFO-file geopend en gelezen. Met het doorlopen van DO-loop 10 zijn alle opgegeven combinaties van drain-diepte en -intensiteit doorgerekend en stopt het programma.

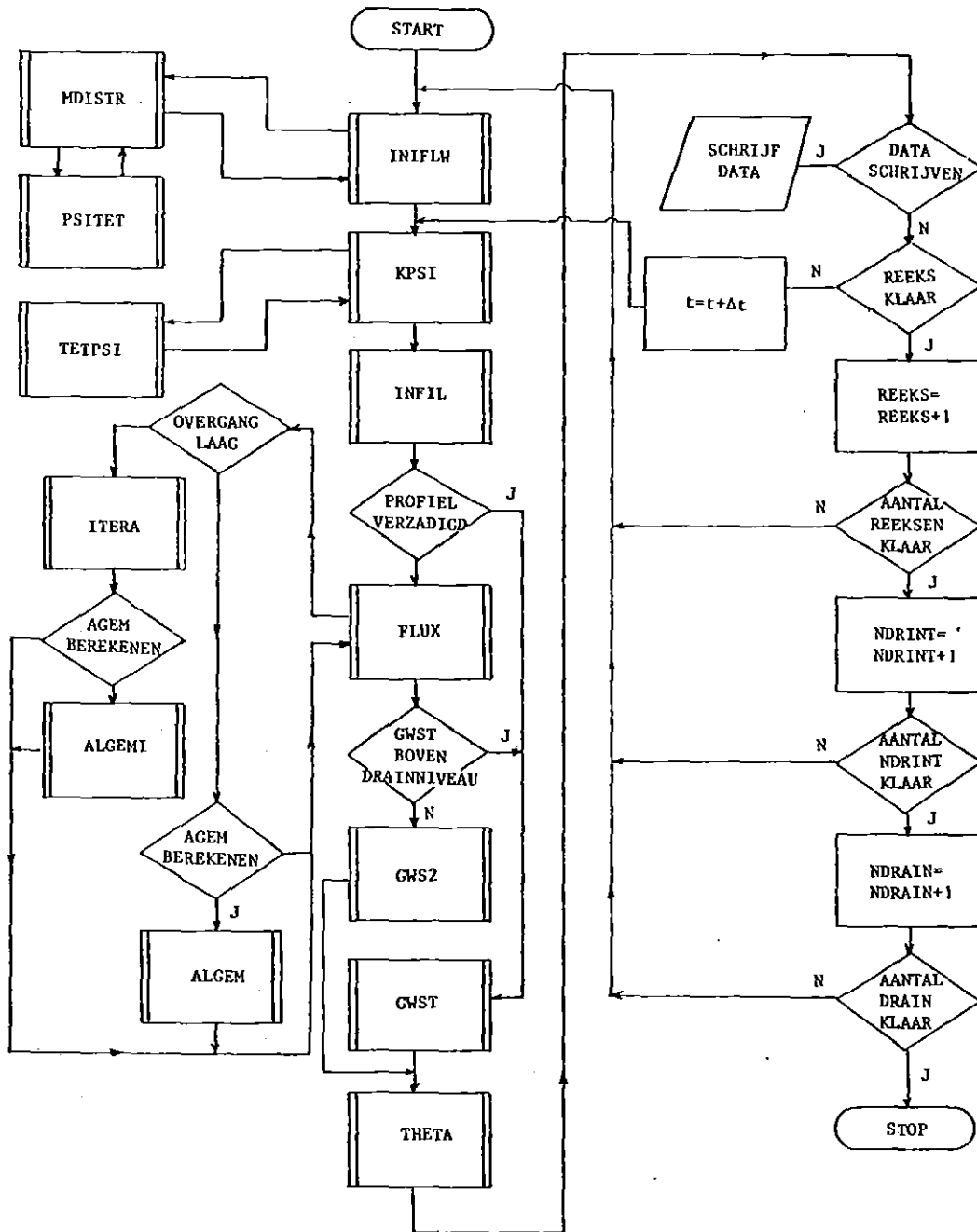


Fig. 6. Stroomschema van het programma FLOWEX

4.2. S u b r o u t i n e s

Het stroomschema van FLOWEX is gegeven in figuur 6.

4.2.1. INIFLW

In subroutine INIFLW wordt alle input afgehandeld en aan alle variabelen een initiële waarde toegekend.

Terwille van de efficiency van het rekenproces worden alle vergelijkingen die constanten opleveren al vast uitgerekend. Voor elke opgegeven laag van het profiel wordt de voor de diverse subroutines relevante informatie zodanig in het geheugen opgeborgen dat met het laagnummer als index alle gegevens direct in de gewenste vorm opvraagbaar zijn. Bij een op te geven flux wordt in INIFLW afhankelijk van draindiepte en -intensiteit de begingrondwaterstand berekend met de in par. 2.4.1. en 2.4.2. gegeven vergelijkingen.

Vervolgens wordt de subroutine MDISTR aangeroepen welke de initiële vochtverdeling boven deze grondwaterstand berekent. Ook wordt in INIFLW de tijdstapgrootte berekend en wel voor de acht natste stappen van de pF-curve van zowel boven- als ondergrond. Met subroutine SORT worden deze waarden gesorteerd en de grootst toegestane tijdstap gevonden.

4.2.2. MDISTR

Deze subroutine berekent met vergelijking (43) de verdeling van de drukhoogte van het bodemvocht met de diepte afhankelijk van een op te geven stationaire flux. Met PSITET (4.2.13.) worden deze drukhoogten in vochtgehalten vertaald.

4.2.3. KPSI

Deze subroutine roept voor elke onverzadigde laag subroutine TETPSI (4.2.12.) aan om het vochtgehalte om te zetten in een drukhoogte. Daarna wordt nagegaan op welk lijnstuk van de $k(\psi)$ -lijnstukken curve de gevonden ψ ligt. Met de bijbehorende k_0 en α en vergelijking (4) wordt vervolgens k berekend.

4.2.4. INFIL

In deze subroutine wordt de infiltratie van maaiveld naar het midden van de eerste laag berekend (zie par. 2.3.3.). In geval van verdamping wordt de mate van verdampingsreductie en de vochtonttrekking uit de bovenste laag berekend. Ook wordt hier de administratie van eventuele oppervlakte-afvoer bijgehouden.

4.2.5. FLUX

Deze subroutine berekent de fluxen tussen de verschillende lagen met vergelijking (5). Getest wordt of de drukhoogten in de twee lagen waartussen de flux loopt op hetzelfde lijnstuk van de $k(\psi)$ -lijnstukken curve liggen. Is dit niet het geval dan wordt ALGEM (4.2.6.) aange-roepen.

Als de berekening op de overgang van twee lagen met verschillende $k(\psi)$ en $\psi(\theta)$ -relaties komt dan wordt ITERA (4.2.7.) aangeroept.

4.2.6. ALGEM

In INIFLW worden tevoren bij elk van de α 's uit de $k(\psi)$ -lijnstukken curve waarden voor de term $e^{\alpha\Delta z}$ uit vergelijking (5) voor berekening van de flux aangemaakt. Liggen de ψ 's in twee opeenvolgende lagen echter niet op hetzelfde lijnstuk van de $k(\psi)$ -lijnstukken curve dan moet in $e^{\alpha\Delta z}$ met een gemiddelde α worden gewerkt. Deze wordt in subroutine ALGEM berekend met vergelijking (44) aan de hand van de ψ 's in de twee lagen.

4.2.7. ITERA

Deze subroutine voert de iteratie uit volgens vergelijking (19) waarmee de ψ op de overgang tussen twee lagen met verschillende $k(\psi)$ en $\psi(\theta)$ -relaties wordt berekend (zie par. 2.2.). Als blijkt dat de aldus verkregen ψ op de overgang niet ligt op hetzelfde lijnstuk van de $k(\psi)$ -lijnstukken curve als de ψ in de laag direct onder de overgang, dan wordt subroutine ALGEM aangeroept. Tenslotte wordt de flux over de overgang tussen beide verschillende bodemlagen berekend.

4.2.8. ALGEM

Deze subroutine doet hetzelfde als ALGEM echter toegespitst op de overgang tussen twee lagen met verschillende eigenschappen. Behalve het zoeken van de juiste α berekent ALGEM ook nog de term $e^{\alpha \frac{1}{2} \Delta z}$ zoals die in vergelijking (5) voor berekening van de flux over de overgang wordt gebruikt. De flux zelf wordt in ITERA berekend.

4.2.9. GWST

Deze subroutine berekent de grondwaterstand en de drainafvoer met de voor verschillende situaties relevante vergelijkingen als gegeven in de par. 2.4.1. en 2.4.2.

Grondwaterstanden < 5 cm-mv kunnen niet worden berekend. In die gevallen wordt grondwaterstand = 0. Een eventueel aanwezige plas op het maaiveld wordt in die situatie erbij opgeteld zodat een grondwaterstand boven maaiveld wordt gevonden. De drainafvoer wordt als diepste flux doorgegeven voor berekening van de vochtinhaltsverandering in de onderste onverzadigde laag door subroutine THETA.

4.2.10. GWS2

Deze subroutine berekent de grondwaterstand beneden drainniveau met vergelijking (42). Indien de berekende grondwaterstand is gedaald in een nog niet eerder in de berekening betrokken laag dan wordt een vlag gezet. Dit is het signaal dat in de volgende tijdstap het aantal lagen in de berekening met één moet worden vergroot. De gebruikte berekeningsmethode laat bij stijgende grondwaterstanden een sprongsgevijs verloop zien. Hiervoor is een eenvoudige demping ingebouwd.

4.2.11. THETA

Deze subroutine berekent met vergelijking (6) en (7) de vochtverandering in het profiel.

Het kan voorkomen dat het vochtgehalte in een laag groter wordt dan het maximale vochtgehalte zoals gegeven door de pF-curve. In dat geval wordt het verschil toegevoegd aan de bovenliggende laag.

Op het einde van de routine wordt getest hoeveel onverzadigde lagen er zijn en doorgegeven aan het hoofdprogramma.

4.2.12. TETPSI

Deze subroutine zet aan de hand van de vocht karakteristiek vochtgehalten om in drukhoogten via lineaire interpolatie. Wordt een vochtgehalte gevonden dat buiten het opgegeven deel van de pF-curve ligt dan stopt het programma.

4.2.13. PSITET

Deze subroutine doet het omgekeerde van TETPSI, namelijk drukhoogten worden via de vocht karakteristiek omgezet in vochtgehalten.

4.2.14. SORT (LARRAY, LEN)

Algemene subroutine door INIFLW aangeroepen, die een aantal (LEN) getallen uit een array sorteert van laag naar hoog.

4.3. I n p u t

Het programma FLOWEX is zodanig georganiseerd dat het zonder ingreep over lange perioden zoals 's nachts of in het weekend kan worden ingezet. Daardoor is het zeer geschikt voor het doorrekenen van lange tijdreeksen. Om dit mogelijk te maken worden alle gegevens gelezen van files met vaste namen die eventueel tijdens de executie van het programma kunnen worden aangepast.

De input van algemene en bodemfysische gegevens is ondergebracht in INFO-files. Voor zowel elke draindiepte als elke drainintensiteit moet een INFO-file aangemaakt worden met een oplopende nummering: INFO01.DAT, INFO02.DAT, enz. Files met gegevens over neerslag en verdamping (METEO-files) hebben een vaste naam op te geven bij punt 14 in de INFO-file.

4.3.1. INFO-file

00 COMMT Regel voor eventuele toelichting

01 NJAAR NDRAIN NDRINT

 aantal drainintensiteiten

 aantal draindiepten

 aantal door te rekenen jaren

- 02 TIJD NLAAG NTOPL
aantal lagen bovengrond
totaal aantal lagen tot en met de drainlaag. De drains
worden geacht in het centrum van de drainlaag te liggen
dag nr. 1e dag
- 03 LAAGDIKTE (1), 1 De laagdikte (in cm) en volgorde nr van 30 lagen
LAAGDIKTE (2), 2
: :
: :
LAAGDIKTE (30), 30
- 04 IRMOD POOLMX of CS POOL BFACT
constante in berekening verdampingsreductie
plasdikte bij start van de berekening
als ICMOD = 0 : POOLMX is de maximale plasdikte waar-
bij runoff gaat optreden
als IRMOD = 1 : CS weerstand in berekening van de
runoff
IRMOD = 0 : berekening van runoff volgens vergelijking (23)
IRMOD = 1 : berekening van runoff volgens vergelijking (25)
- 05 DD DRINT VCONST
constant flux (cm) bij de start van de bereke-
ning (neerwaartse stroming negatief)
drainintensiteit (d^{-1})
draindiepte (cm) ten opzichte van maaiveld
- 06 THETAT (I), PSIT (I)
Vochtkarakteristiek van de bovengrond, opgedeeld in maximaal 40
vochtgehalte-drukhoogte combinaties, beginnend bij het laagste
vochtgehalte.
Ter afsluiting van de reeks wordt een negatief vochtgehalte
ingevoerd. Vochtgehalten in fracties, drukhoogten in cm.

- 07 $k_o (1, I)$, $I = 1,3$
Waarden van k_o (cm.d^{-1}) behorend bij elk van de lijnstukken van de $k(\psi)$ -lijnstukken curve van de bovengrond.
- 08 ALFA (1, I), $I = 1,3$
Waarden van α behorend bij elk van de lijnstukken van de $k(\psi)$ -lijnstukken curve van de bovengrond.
- 09 GRENS (1, I), $I = 1,2$
Waarden van de drukhoogte (cm) behorend bij de knikpunten van de $k(\psi)$ -lijnstukken curve beginnend met de grootste (minst negatieve) waarde van ψ .
- 10 THETAL (I), PSIL (I)
Als 06 maar van de ondergrond.
- 11 $k_o (2, I)$, $I = 1,3$
Als 07 maar van de ondergrond.
- 12 ALFA (2, I), $I = 1,3$
Als 08 maar van de ondergrond.
- 13 GRENS (2, I), $I = 1,2$
Als 09 maar van de ondergrond.
- 14 INPUT, naam van de METEO-file met gegevens over regen en verdamping.
Deze naam wordt opgegeven volgens een vaste indeling:

Azz.AAA

waarin: A = door de gebruiker te kiezen letters

zz = getallen, aanduiding voor het te behandelen jaar.

Voorbeeld: file R83.FLW bevat de regen en verdampingsgegevens van 1983.

- 15 OUTPUT, naam van de file waarop de output wordt weggeschreven.
Deze naam wordt opgegeven volgens een vaste indeling:

AAAAxAy.Azz

waarin: A = door de gebruiker te kiezen letters
x = getal, aanduiding voor drainageintensiteit
y = getal, aanduiding voor drainagediepte
zz = getallen, aanduiding voor het te behandelen jaar
(deze worden overgenomen uit de METEO-file en hoeven
niet ingevuld te worden).

Voorbeeld: filenaam OUT1V2.083 betekent:

drainageintensiteit 1
draindiepte 2
meteogegevens van 1983.

- 16 ANSWER. Afhankelijk van JA of NEE wordt de grootste toelaatbare tijdstap al of niet berekend.
- 17 DELTAT. Waarde van de tijdstap Δt (d) als bij 16 NEE is ingevuld.

4.3.2. Meteo-files

De neerslag per dag wordt verdeeld over 5 delen (dagdelen = NPARTS). Verdamping wordt ingevoerd als een negatieve waarde en gesommeerd bij eventueel aanwezige neerslag.

De opbouw van deze files is als volgt.

Op het eerste record staat NPARTS, dan volgt een aantal records gelijk aan NPARTS met neerslag- of verdampingsgegevens. Elk record bevat één neerslag- of verdampingscijfer.

Na afhandeling van NPARTS dagdelen wordt elke keer het jaartal in de filenaam, opgegeven bij punt 14 van de INFO-file, met één verhoogd tot het aantal te berekenen jaren is bereikt.

4.4. O u t p u t f i l e s

Ter identificatie van de outputgegevens worden vooraf in de outputfile alle gegevens zoals die in de INFO-file zijn opgegeven naar de outputfile geschreven.

In de huidige versie van FLOWEX wordt 5x per dag informatie over de drukhoogteverdeling in het profiel en de grondwaterstand naar de outputfile geschreven. Het aantal records met gegevens volgend op de algemene informatie uit INFO is dus gelijk aan NPARTS x het aantal opgegeven jaren.

De opbouw van de records is als volgt.

Het eerste getal (N) geeft aan hoeveel drukhoogtewaarden er per record zijn. Daarachter volgen N drukhoogtewaarden en daarachter de grondwaterstand.

Om ruimte te besparen worden de getallen na met -10 te zijn vermenigvuldigd in Integerformat (I3, NI5) weggeschreven. Hierdoor vervallen de decimale punten en in de meeste gevallen de min-tekens.

Het zal duidelijk zijn dat op elk gewenst moment output over vochtgehalte, drukhoogte, doorlatendheid, flux, drainafvoer, grondwaterstand, runoff en andere in de berekening meespelende grootheden door plaatsing van een WRITE-statement beschikbaar zijn.

5. LITERATUUR

- ERNST, L.F., 1956. Calculation of the steady flow of groundwater in vertical cross sections. *Neth. J. Agric. Sci.* 4: 126-131.
- HOOGHOUDT, S.B., 1940. Bijdrage tot de kennis van enige natuurkundige grootheden van den grond, deel 7. *Versl. Landbouwk. Onderz.* 46(14): 515-707. Ministry Agric., The Hague.
- PAUL, C.L., 1978. Prediction of trafficability of tile-drained farmland. Thesis University of British Columbia: 133 p.
- RICHTER, J., 1980. A single numerical solution for the vertical flow equation of water through unsaturated soils. *Soil Science* 129, 3: 138-144.
- RIJTEMA, P.E., 1965. An analysis of actual evapotranspiration. *Agric. Res. Rep.* 659. Pudoc, Wageningen: 107 p.
- WIND, G.P. and W. VAN DOORNE, 1975. A numerical model for the simulation of unsaturated vertical flow of moisture in soils. *J. Hydrol.* 24: 1-20
- and A. VANDENBERG, 1983. The generation of river alimentation in response to precipitation; a soil physical approach. *ICW-nota* 1437: 19 p.
- WIJK, A.L.M. VAN, 1980. A soil technical study on effectuating and maintaining adequate playing conditions of grass sports fields. *Agric. Res. Rep.* 903 Pudoc, Wageningen: 124 p.


```

C-----
0014 CALL INIFLW           ! Initializing -
C-----
C
C-----
0015 NDPART=5             ! one day is divided in 5 parts
0016 NSTEPS=INT((1./NDPART)/DELTAT) ! # timesteps within 1/3 day
0017 RESTIM=(1./NDPART)-NSTEPS*DELTAT ! remainder
0018 DO 15 I=1,NSTEPS      !
0019 TSTEP(I)=DELTAT       ! fill TSTEP with timestep
0020 15 CONTINUE          !
0021 IF (RESTIM,LE.0.0) GO TO 25 ! skip if zero
0022 TSTEP(I)=RESTIM       ! put remainder in last TSTEP
0023 NSTEPS=NSTEPS+1      ! increase # timesteps
C-----
C
C-----
0025 25 DO 40 I=1,NPARTS  ^----- number of days * 5
C-----
0026 DO 50 ISTEP=1,NSTEPS ^----- number of timesteps within 'I'
C-----
0027 DELTAT=TSTEP(ISTEP)  ^----- timestep
C-----
C
0028 IF (IF2.EQ.1) NLayer=NLayer +1 ! tests whether unsaturated
0030 IF (IFLAG.EQ.1.AND.NLayer.NE.NLAAG) ! level is falling
1 NLayer=NLayer+1
C-----
0032 CALL KPSI           ! Calculate pressure heads and hydr. conductivity -
C-----
C
C-----
0033 CALL INFIL         ! Calculate infiltration etc. -
C-----
C
C-----
0034 NL=J-2             ! Test whether saturation level lies
0035 IF (NL.LT.1) GO TO 35 ! above or below drainage level
C-----
C
C-----
0037 CALL FLUX         ! Calculate fluxes between all layers -
C-----
C
C-----
0038 IF (NL.LT.NLAAG-1) GO TO 35 ! we're above drainage level -
C-----
C
C-----
0040 CALL GWS2         ! Calculate groundwaterlevel below drainagelevel -
C-----
C
0041 GO TO 45
C-----
C
0042 35 CALL GWST       ! Calculate groundwaterlevel above drainagelevel -
C-----
C
C-----
0043 45 CALL THETA     ! Calculate soil water content in each layer -
C-----
C
0044 TYD=TYD+DELTAT    ! update time
C-----
0045 50 CONTINUE       ! next timestep
C-----
C
C-----
0046 DO 55 NM=1,NLayer  ! most values are negative
0047 55 IPSIAL(NM)=PSI(NM) * -10. + 0.5 ! to save space on disc all
0048 IGWS =GRWST * -10. + 0.5 ! data are converted to positive
0049 IRNOFF =SUMRNF * 10. + 0.5 ! integer values
0050 WRITE (12,100) NLayer, (IPSIAL(NM),NM=1,NLayer), IGWS, IRNOFF
0051 SUMRNF=0.0
C-----
C
0052 40 CONTINUE       ! next 1/5 of a day
0053 100 FORMAT (13,3115)
    
```

```
C-----
C      Update 'METEO'-file
C-----
0054      INPUT(3)=INPUT(3)+1
0055      IF (INPUT(3).LE.'9') GO TO 65
0057      INPUT(3)='0'
0058      INPUT(2)=INPUT(2)+1
C-----
C      Close in- and outputfiles
C-----
0059      65 CLOSE (UNIT= 4)
0060          CLOSE (UNIT=10)
0061          CLOSE (UNIT=11)
0062          CLOSE (UNIT=12)
0063          SKIP=.TRUE.
0064          DELTAT=TSTEP(1)
C
0065      30 CONTINUE
C-----
C      Update 'INFO'-filename
C-----
0066      SKIP=.FALSE.
0067      INFO(6)=INFO(6)+1
0068      IF (INFO(6).LE.'9')      GO TO 20
0070      INFO(6)='0'
0071      INFO(5)=INFO(5)+1
C
0072      20 CONTINUE          ! next drainage intensity
C
0073      10 CONTINUE          ! next draindepth
C-----
C-----
C-----
C-----
C-----
0074      CALL EXIT
0075      END
```

```

0001      SUBROUTINE INIFLW
C          *****
C
C-----
C          INIFLW belongs to the main program 'FLOWEX'
C
C          This routine takes care of all input needed by FLOWEX.
C          Input data are read from an 'INFO'-file and written to a scratch
C          file. As soon as the name of the outputfile is known, these
C          data are written to the outputfile for identification purposes.
C
C          An initial value is assigned to variables that need initializing.
C          Equations or part of equations that yield a constant are
C          substituted for efficiency reasons.
C
C          The initial groundwaterlevel is calculated using eq. (31) or (39).
C          This has been described in detail in chapter 2.4.1
C
C          Subroutine 'MDISTR' is called to calculate the initial soil water
C          distribution of the profile due to an initial flux with eq. (43)
C          described in chapter 2.5.
C
C          This routine calculates also timesteps with eq. (45) for the
C          eight wettest points of the soil water retention curve of the top-
C          soil as well as the subsoil. The longest allowable timestep is found
C          after sorting these sixteen values by subroutine 'SDRT'.
C          This has been described in chapter 3.4.
C          In case of fatal errors in the inputdata error messages are printed
C          on the console terminal and the program stops.
C-----
C
C          **** Local variable declaration ****
0002      BYTE INPUT(8), OUTPUT(11), INFO(12), COMMNT(80), ANSWER, TIMCAL, SKIP
0003      DIMENSION TEMPK(30), TSTEP(60), CONST(10)
C
C          **** Common variable declaration ****
0004      DIMENSION ALFA(30,3), A(30,3), AHDZ(30,3), GRENS(30,2), DELTAZ(30)
          1, ID(30), THETAT(40), THETAL(40), PSIT(40), PSIL(40), TETMAX(40),
          1, REGEN(1370)
0005      REAL*8 TETA(30), PSI(30), V(30)
0006      REAL K(30), K0(30,3), KGEM, KK, Kover, LAAGD(31)
C
C          **** Common block ****
0007      COMMON PSI, TETA, V, ALFA, K0, K, LAAGD, DELTAT, A, AHDZ, GRENS, AGEM,
          1, GGA, REGEN, POOL, POOLMX, SRPLUS, TETMAX, GRWST, DD, HG, ASTER, SUMRNF
          1, ASTERB, BFACT, THETAL, THETAT, PSIL, PSIT, MAXL, MAXT, NL, IFLAG, IF2,
          1, ID1, ID2, JJ, J, I, NTOPL, NLOWL, ITRANS, NLAAG, NLAYER, NPARTS, TYD
          1, IT, DELTAZ, PSITER, Kover, NDRAIN, NDRINT, NJAAR, NSKIP, VCONST, IARMOD, CS
          1, INPUT, OUTPUT, INFO, SKIP
C
0008      DATA SKIP/.FALSE./
C-----
C
C          files
C-----
0009      OPEN (UNIT=10, NAME=INFO, TYPE='OLD')          ! === "INFO"file =====
0010      OPEN (UNIT=4, NAME='SCRAT', TYPE='SCRATCH')    ! === "SCRATCH"file ===
C-----
C          start reading data
C-----
C
C          ----- comment ----- line 00 -----
0011      READ (10,100)          NCHAR, (COMMNT(NN), NN=1, NCHAR)    ! header
0012      WRITE (4,200)          (COMMNT(NN), NN=1, NCHAR)
0013      READ (10,100)          NCHAR, (COMMNT(NN), NN=1, NCHAR)
0014      WRITE (4,200)          (COMMNT(NN), NN=1, NCHAR)
    
```

```

C ----- loop parameters ----- line 01 -----
C
0015 READ (10,100) NCHAR, (COMMNT(NN), NN=1, NCHAR) ! header
0016 WRITE (4,200) (COMMNT(NN), NN=1, NCHAR)
0017 READ (10,*) NJAAR, NDRAIN, NDRINT
0018 WRITE (4,*) NJAAR, NDRAIN, NDRINT

C ----- initial time, layer definition ----- line 02 -----
C
0019 READ (10,100) NCHAR, (COMMNT(NN), NN=1, NCHAR) ! header
0020 WRITE (4,200) (COMMNT(NN), NN=1, NCHAR)
0021 READ (10,*) TYD, NLAAG, NTOPL
0022 WRITE (4,*) TYD, NLAAG, NTOPL
C | ^ ^----- # layers topsoil
C | ^ ^----- # layers until drainagelayer
C | ^ ^----- initial time
0023 ITRANS=NTOPL+1
0024 NLOWL=NLAAG-NTOPL

C ----- thickness of each layer ----- line 03 -----
C
0025 READ (10,100) NCHAR, (COMMNT(NN), NN=1, NCHAR) ! header
0026 WRITE (4,200) (COMMNT(NN), NN=1, NCHAR)
0027 DD 10 I=1,30
0028 READ (10,*) LAAGD(I+1), LAAGNR
0029 WRITE (4,*) LAAGD(I+1), LAAGNR
0030 10 CONTINUE
0031 NLAYER=I-1

C ----- runoff parameters and evap. constant ---- line 04 -----
C
0032 READ (10,100) NCHAR, (COMMNT(NN), NN=1, NCHAR) ! header
0033 WRITE (4,200) (COMMNT(NN), NN=1, NCHAR)
0034 READ (10,*) IRMOD, POOLMX, POOL, BFACT
0035 WRITE (4,*) IRMOD, POOLMX, POOL, BFACT
0036 IF (IRMOD.EQ.1) CS=POOLMX

C ----- drainage constants, initial flux ---- line 05 -----
C
0038 READ (10,100) NCHAR, (COMMNT(NN), NN=1, NCHAR) ! header
0039 WRITE (4,200) (COMMNT(NN), NN=1, NCHAR)
0040 READ (10,*) DD, DRINT, VCONST
0041 WRITE (4,*) DD, DRINT, VCONST

C=====
C soil water retention curve topsoil =
C=====
C ----- line 06 -----
C
0042 READ (10,100) NCHAR, (COMMNT(NN), NN=1, NCHAR) ! header
0043 WRITE (4,200) (COMMNT(NN), NN=1, NCHAR)
0044 DO 55 ITEL = 1,40
0045 READ (10,*) THETAT (ITEL), PSIT (ITEL)
C | ^ ^----- soil water pressure head
C | ^ ^----- soil water content
0046 WRITE (4,*) THETAT (ITEL), PSIT (ITEL)
0047 IF (THETAT (ITEL).LT.0) GO TO 56
0049 55 CONTINUE
0050 56 MAXT=ITEL-1
0051 IF (MAXT.LE.39) GO TO 57

C
C+++++ fatal error ++++++
0053 ITM=MAXT-39
0054 TYPE 201, ITM
0055 GO TO 9999 ! abort the program
C+++++

C ----- hydraulic conductivity ----- line 07 -----
C
0056 57 READ (10,100) NCHAR, (COMMNT(NN), NN=1, NCHAR) ! header
0057 WRITE (4,200) (COMMNT(NN), NN=1, NCHAR)
0058 READ (10,*) (KO(I, I), I=1,3)
0059 WRITE (4,*) (KO(I, I), I=1,3)

C ----- constant in k(h) ----- line 08 -----
C
0060 READ (10,100) NCHAR, (COMMNT(NN), NN=1, NCHAR) ! header
    
```

```

0061      WRITE ( 4,200)      (COMMNT (NN),NN=1,NCHAR)
0062      READ  (10,*)        (ALFA(1,J),J=1,3)
0063      WRITE ( 4,*)        (ALFA(1,J),J=1,3)
C
C ----- limits in k(h) ----- line 09 -----
C
0064      READ (10,100)      NCHAR, (COMMNT (NN),NN=1,NCHAR) ! header
0065      WRITE ( 4,200)      (COMMNT (NN),NN=1,NCHAR)
0066      READ  (10,*)        (GRENS(1,J),J=1,2)
0067      WRITE ( 4,*)        (GRENS(1,J),J=1,2)
C
C -----
C      initialize parameters topsoil
C -----
C
0068      SRPLUS=0.0
0069      IF2=0
0070      IFLAG=0
0071      LAAGD(1)=0.0
0072      V(1)=0.0
0073      DBG=0.0
C
0074      DO 20 I=1,NTDPL
0075          DBG      = DBG+LAAGD(I+1)
0076          TETMAX(I)= THETAT (MAXT)
0077          TETA(I)  = TETMAX(I)
0078          DELTAZ(I)= (LAAGD(I)+LAAGD(I+1))*0.5
0079          PSI(I)   = 0.0D0
0080          K(I)     = KO(1,1)
C
0081      DO 20 J=1,3
0082          KO(I,J)  = KO(1,J)
0083          ALFA(I,J) = ALFA(1,J)
0084          A(I,J)   = EXP(ALFA(I,J)*DELTAZ(I))
0085          AHDZ(I,J) = EXP(ALFA(1,J)*LAAGD(I+1)*0.5)
0086          IF(J.EQ.3) GO TO 20
0088          GRENS(I,J) = GRENS(1,J)
0089      20 CONTINUE
C
C -----
C      soil water retention curve subsoil
C -----
C
C ----- line 10 -----
C
0090      READ (10,100)      NCHAR, (COMMNT (NN),NN=1,NCHAR) ! header
0091      WRITE ( 4,200)      (COMMNT (NN),NN=1,NCHAR)
0092      DO 60 ITEL = 1,40
0093      READ  (10,*) THETAL (ITEL),PSIL (ITEL)
C
C          ^ ^----- soil water pressure head
C          ^ ^----- soil water content
0094      WRITE ( 4,*) THETAL (ITEL),PSIL (ITEL)
0095      IF (THETAL (ITEL).LT.0) GO TO 61
0097      60 CONTINUE
0098      61 MAXL=ITEL-1
0099      IF (MAXL.LE.39) GO TO 62
C
C ++++++ fatal error ++++++
0101      ITM=MAXL-39
0102      TYPE 202, ITM
0103      GO TO 9999 ! abort the program
C ++++++
C
C ----- hydraulic conductivity ----- line 11 -----
C
0104      62 READ (10,100)      NCHAR, (COMMNT (NN),NN=1,NCHAR) ! header
0105      WRITE ( 4,200)      (COMMNT (NN),NN=1,NCHAR)
0106      READ  (10,*)        (KO(ITRANS,I),I=1,3)
0107      WRITE ( 4,*)        (KO(ITRANS,I),I=1,3)
C
C ----- constant in k(h) ----- line 12 -----
C
0108      READ (10,100)      NCHAR, (COMMNT (NN),NN=1,NCHAR) ! header
0109      WRITE ( 4,200)      (COMMNT (NN),NN=1,NCHAR)
0110      READ  (10,*)        (ALFA(ITRANS,J),J=1,3)
0111      WRITE ( 4,*)        (ALFA(ITRANS,J),J=1,3)
C
C ----- limits in k(h) ----- line 13 -----

```

```

C
0112 READ (10,100) NCHAR,(COMMNT(NN),NN=1,NCHAR) ! header
0113 WRITE(4,200) (COMMNT(NN),NN=1,NCHAR)
0114 READ (10,*) (GRENS(ITRANS,J),J=1,2)
0115 WRITE (4,*) (GRENS(ITRANS,J),J=1,2)

C
C -----
C initialize parameters subsoil
C -----
C
0116 DO 30 I=1TRANS,30
0117 TETMAX(I) = THETAL(MAXL)
0118 TETA(I) = TETMAX(I)
0119 DELTAZ(I) = (LAGD(I)+LAGD(I+1))*0.5
0120 PSI(I) = 0.0D0
0121 K(I) = KO(ITRANS,1)

C
0122 DO 30 J=1,3
0123 KO(I,J) = KO(ITRANS,J)
0124 ALFA(I,J) = ALFA(ITRANS,J)
0125 A(I,J) = EXP(ALFA(I,J)*DELTAZ(I))
0126 ANDZ(I,J) = EXP(ALFA(I,J)*LAGD(I+1)*0.5)
0127 IF(J.EQ.3) GO TO 30
0129 GRENS(I,J)= GRENS(ITRANS,J)
0130 30 CONTINUE

C
C=====
C calculate ASTER and ASTERB : drainage constants
C=====
0131 HLP=0.0
0132 DO 40 I=ITRANS,NLAAG ! distance from transition to drainage
0133 HLP = HLP+LAGD(I+1)
0134 40 CONTINUE
0135 HLP = HLP-(LAGD(NLAAG+1)*0.5)
0136 HLP = (HLP/KO(ITRANS,1))-(HLP/KO(1,1))
0137 ASTER=1./DRINT
0138 ASTERB=ASTER+HLP

C=====
C calculate initial groundwaterlevel due to the given flux (VCONST)
C=====
0139 OPB=-(VCONST*ASTER)/(1.+(VCONST/KO(ITRANS,1)))
0140 GRWST=DD+OPB
0141 IF(-GRWST.GE.DBG) GO TO 42
0143 OPB=-(VCONST*ASTERB)/(1.+(VCONST/KO(1,1)))
0144 GRWST=DD+OPB

C=====
0145 42 CALL MDISTR ! calculate initial distribution of soil watercontent
C=====

C
0146 IF(SKIP) GO TO 95

C
C ----- name of inputfile ----- line 14 -----
C
0148 READ (10,100) NCHAR,(COMMNT(NN),NN=1,NCHAR) ! header
0149 WRITE(4,200) (COMMNT(NN),NN=1,NCHAR)
0150 READ (10,100) NCHAR,(INPUT(NN),NN=1,NCHAR)
0151 WRITE(4,200) (INPUT(NN),NN=1,NCHAR)

C
C ----- name of outputfile ----- line 15 -----
C
0152 READ (10,100) NCHAR,(COMMNT(NN),NN=1,NCHAR) ! header
0153 WRITE(4,200) (COMMNT(NN),NN=1,NCHAR)
0154 READ (10,100) NCHAR,(OUTPUT(NN),NN=1,NCHAR)
0155 WRITE(4,200) (OUTPUT(NN),NN=1,NCHAR)

C
C ----- calc. timestep [y/n] ----- line 16 -----
C
0156 READ (10,100) NCHAR,(COMMNT(NN),NN=1,NCHAR) ! header
0157 WRITE(4,200) (COMMNT(NN),NN=1,NCHAR)
0158 READ (10,105) TIMCAL
0159 WRITE(4,200) TIMCAL

C
C ----- timestep ----- line 17 -----
C
0160 READ (10,100) NCHAR,(COMMNT(NN),NN=1,NCHAR) ! header
0161 WRITE(4,200) (COMMNT(NN),NN=1,NCHAR)
0162 IF(TIMCAL.NE.'N') GO TO 29 ! calculate timestep
    
```

```

0164      ILUN=10
0165      GO TO 99      ! skip timestep calculations
C=====
C      calculate timestep      -- topsoil --
C=====
0166      29 LEN=8
0167      I2=0
0168      DO 70 I=MAXT,MAXT-LEN,-1
0169      I2=I2+1
0170      ID(I2)=1
0171      IF(PSIT(I).LE.GRENS(1,1)) ID(I2)=2
0173      IF(PSIT(I).LE.GRENS(1,2)) ID(I2)=3
0175      TEMPK(I2)=KO(1,ID(I2)) * EXP(ALFA(1,ID(I2))) * PSIT(I)
0176      T1=EXP(ALFA(1,ID(I2))*10.)+1.
0177      T2=EXP(ALFA(1,ID(I2))*10.)-1.
0178      CONST(I2)=T1/T2/10.
0179      70 CONTINUE
C
0180      JT=MAXT
0181      DO 71 J=1,LEN
0182      DK=TEMPK(J)-TEMPK(J+1)
0183      DT=THETAT(JT)-THETAT(JT-1)
0184      DKDT=DK/DT
0185      TSTEP(J)=1./(DKDT*CONST(J))
0186      JT=JT-1
0187      71 CONTINUE
C=====
C      calculate timestep      -- subsoil --
C=====
0188      LEN2=8
0189      I2=0
0190      DO 80 I=MAXL,MAXL-LEN2,-1
0191      I2=I2+1
0192      ID(I2)=1
0193      IF(PSIL(I).LE.GRENS(ITRANS,1)) ID(I2)=2
0195      IF(PSIL(I).LE.GRENS(ITRANS,2)) ID(I2)=3
0197      TEMPK(I2)=KO(ITRANS,ID(I2)) * EXP(ALFA(ITRANS,ID(I2))) * PSIL(I)
0198      T1=EXP(ALFA(ITRANS,ID(I2))*10.)+1.
0199      T2=EXP(ALFA(ITRANS,ID(I2))*10.)-1.
0200      CONST(I2)=T1/T2/10.
0201      80 CONTINUE
C
0202      JT=MAXL
0203      DO 81 J=1,LEN2
0204      JJ=J+LEN
0205      DK=TEMPK(J)-TEMPK(J+1)
0206      DT=THETAL(JT)-THETAL(JT-1)
0207      DKDT=DK/DT
0208      TSTEP(JJ)=1./(DKDT*CONST(J))
0209      JT=JT-1
0210      81 CONTINUE
C=====
C      look for the smallest
C=====
0211      LEN=LEN+LEN2
0212      CALL SQRT(TSTEP,LEN)
0213      DELTAT=TSTEP(1)
C      ^----- this is the timestep
0214      TYPE 101,DELTAT      ! show it
0215      TYPE 102      ! o.k.?
0216      ACCEPT 100,ANSWER
0217      IF(ANSWER.NE.'N') GO TO 95      ! yes
0219      ILUN=5
0220      TYPE 103      ! request for a better one
0221      99 READ (ILUN,*) DELTAT
0222      95 WRITE (4,104) DELTAT
C
C=====
C      in- and outputfile handling
C=====
C
0223      OUTPUT(9)=INPUT(2)
0224      OUTPUT(10)=INPUT(3)
0225      TYPE 109,OUTPUT
C
0226      OPEN (UNIT=12,NAME=OUTPUT,ERR=97,TYPE='NEW')
C-----
C      write all data from 'INFO'-file to the outputfile
    
```



```

0227 REWIND 4
0228 84 READ ( 4,100,END= 85) NCHAR, (COMMNT(NN),NN=1,NCHAR)
0229 WRITE(12,200) (COMMNT(NN),NN=1,NCHAR)
0230 GO TO 84
C-----
C open 'METEO'-file and read data
C-----
0231 85 OPEN (UNIT=11,NAME=INPUT, ERR=98,TYPE='OLD')
0232 READ (11,*,ERR=98) NPARTS
0233 DO 5 I=1,NPARTS
0234 READ (11,*,ERR=98) RAIN
0235 REGEN(I)=-RAIN * 5.
0236 5 CONTINUE
C
0237 RETURN ! to the main program FLOWEX *
C-----
C error conditions
C-----
C
0238 97 CALL ERRSNS(IFAULT,12)
0239 TYPE 107,OUTPUT,IFAULT
0240 GO TO 9999
0241 98 CALL ERRSNS(IFAULT,12)
0242 TYPE 107, INPUT,IFAULT
0243 9999 STOP ' !! Fatal error in inputdata found by INIFLW !! '
C-----
C FORMATS
C-----
0244 100 FORMAT(Q,80A1)
0245 101 FORMAT(' The largest allowable timestep is: ',F8.4,' days.'
*,/,T40,'=====',//)
0246 102 FORMAT(' Is this timestep correct ? (Y/N) (RET)',%)
0247 103 FORMAT(' Type a better value for DELTAT: ',%)
0248 104 format(F8.4,/,', ',70('='))
0249 105 FORMAT(10A1)
0250 107 FORMAT(' Problems with file: ',10A1,' Error',I3)
0251 109 FORMAT('OBusy with: ',14A1)
0252 110 FORMAT('O')
0253 200 FORMAT(X,80A1)
0254 201 FORMAT('OToo many data in soil water retention curve of
1 topsoil',/,', ' reduce number of data with',I2,/)
0255 202 FORMAT('OToo many data in soil water retention curve of
1 subsoil',/,', ' reduce number of data with',I2,/)
C-----
C-----
C-----
0256 END
    
```

```

0001      SUBROUTINE MDISTR
C      *****
C
C=====
C      MDISTR belongs to the main program 'FLOWEX'
C
C      This routine calculates the initial pressure head distribution
C      in the profile due to a certain flux with eq. (43).
C      The value of this flux (VCONST) has been read by subroutine 'INIFLW'
C      on line 05 of the INFO-file.
C
C      Also subroutine 'PSITET' is called to convert pressure heads to
C      to soil water content.
C=====
C
C***** declarations *****
C
0002      BYTE INPUT(8), OUTPUT(11), INFO(12), COMMNT(80), ANSWER, TIMCAL, SKIP
0003      DIMENSION DZ(31)
0004      DIMENSION ALFA(30,3), A(30,3), AHDZ(30,3), GRENS(30,2), DELTAZ(30)
          1, ID(30), THETAT(40), THETAL(40), PSIT(40), PSIL(40), TETMAX(40),
          1REGEN(1370)
0005      REAL*8 TETA(30), PSI(30), V(30)
0006      REAL K(30), KO(30,3), KGEM, KK, Kover, LAAGD(31)
C
C***** common block *****
C
0007      COMMON PSI, TETA, V, ALFA, KO, K, LAAGD, DELTAT, A, AHDZ, GRENS, AGEM,
          1AG, GGA, REGEN, POOL, POOLMX, SPLUS, TETMAX, GRWST, DD, HG, ASTER, SUMRNF
          1, ASTERB, BFACT, THETAL, THETAT, PSIL, PSIT, MAXL, MAXT, NL, IFLAG, IF2,
          1ID, ID1, ID2, JJ, J, I, NTOPL, NLOWL, ITRANS, NLAAG, NLayer, NPARTS, TYD
          1, IT, DELTAZ, PSITER, Kover, NDRAIN, NDRINT, NJAAR, NSKIP, VCONST, IRMOD, CS
          1, INPUT, OUTPUT, INFO, SKIP
C=====
C
C
0008      SUM=0.0
0009      DO 10 I=1,30                ! determine index of layer
0010      SUM=SUM-DELTAT(I)          ! containing groundwaterlevel
0011      DZ(I)=-DELTAT(I)
0012      IF(SUM.LE.GRWST) GO TO 11
0014 10  CONTINUE
C
0015 11  DZ(I)=GRWST-(SUM+DELTAT(I))
C      ^----- thickness of unsaturated part of that layer
0016      I=I-1
0017      PSI(I)=0.0
0018      ID(I)=1
0019      DO 20 J=I,1,-1
0020      T1=EXP(ALFA(J, ID(J)) * DZ(J+1)) - 1.
0021      T2=EXP(ALFA(J, ID(J)) * (PSI(J+1) + DZ(J+1)))
0022      T3=(VCONST / KO(J,1)) * T1 + T2
C
0023      PSI(J)=1. / ALFA(J, ID(J)) * ALOG(T3)    ! calculate pressure heads
C
0024      IF(J.EQ.1) GO TO 19          ! all heads calculated
0026      ID(J-1)=1
0027      IF (PSI(J).LT.GRENS(J,1)) ID(J-1)=2
0029      IF (PSI(J).LT.GRENS(J,2)) ID(J-1)=3
0031 19  CALL PSITET                  ! convert to soil water content
0032 20  CONTINUE
C
0033      NLayer=I                    ! number of unsaturated layers
C
0034      RETURN
C=====
0035      END
    
```

```

0001      SUBROUTINE KPSI
          *****
          C
          C-----
          C      KPSI belongs to the main program 'FLOWEX'
          C
          C      This routine calls for all unsaturated layers (NLAYER from 'THETA')
          C      subroutine TETPSI to convert the soil water content to a pressure head.
          C      Where the relation between k and pressure head consists of three
          C      sections, it has to be found out within which section the calculated
          C      pressure head falls.
          C      Depending on the section, an identifier (ID) equals the value 1, 2 or 3
          C      This identifier points also to the right values of ALFA and k0 to
          C      in the calculation of the hydraulic conductivity
          C
          C      This subject has been described in chapter 2.1
          C-----
          C
          C***** Declarations *****
          C
          0002      DIMENSION ALFA(30,3), A(30,3), AMDZ(30,3), GRENS(30,2), DELTAZ(30)
          0003      1, ID(30), THETAT(40), THETAL(40), PSIT(40), PSIL(40), TETMAX(40),
          0004      1REGEN(1370)
          0003      REAL*8 TETA(30), PSI(30), V(30)
          0004      REAL K(30), K0(30,3), KGEM, KK, Kover, LAAGD(31)
          C
          C***** Common block *****
          C
          0005      COMMON PSI, TETA, V, ALFA, K0, K, LAAGD, DELTAT, A, AMDZ, GRENS, AGEM,
          0005      1AG, GBA, REGEN, POOL, POOLMX, SRPLUS, TETMAX, GRWST, DD, HG, ASTER, SUMRNF
          0005      1, ASTERB, BFACT, THETAL, THETAT, PSIL, PSIT, MAXL, MAXT, NL, IFLAG, IF2,
          0005      1ID, ID1, ID2, JJ, J, I, NTOPL, NLOWL, ITRANS, NLAG, NLAYER, NPARTS, TYD
          0005      1, IT, DELTAZ, PSITER, Kover, NDRAIN, NDRINT, NJAAR, NSKIP, VCONST, IRMOD, CS
          0005      1, INPUT, OUTPUT, INFO, SKIP
          C
          C-----
          C
          0006      DO 10 J=1, NLAYER
          C      ^----- number of unsaturated layers
          0007      CALL TETPSI
          C      ^----- soil water content to pressure head
          C
          0008      ID(J)=1
          0009      IF (PSI(J).LT.GRENS(J,1)) ID(J)=2
          0011      IF (PSI(J).LT.GRENS(J,2)) ID(J)=3
          C      ^----- identifies section of k(h)-relation
          C
          0013      K(J)=K0(J, ID(J))*EXP(ALFA(J, ID(J))*PSI(J)) ! Eq. (4)
          0014      10 CONTINUE
          C
          0015      RETURN
          C-----
          0016      END
    
```

```

0001      SUBROUTINE INFIL
C      *****
C
C-----
C      INFIL belongs to the main program 'FLOWEX'
C
C      This routine calculates the reduction of the evaporation according
C      to eq. (22).
C      Depending on the value of 'IRMOD', runoff (if relevant) is calculated
C      with eq. (23) or eq. (25)
C      The total amount of runoff is calculated until it is written
C      to the outputfile.
C
C      Downward or upward flux between surface and the centre of the first
C      layer is calculated with eq. (5) resp. eq. (26)
C
C      This subject has been described in detail in chapter 2.3
C-----
C
C***** Declarations *****
C
0002      DIMENSION ALFA(30,3), A(30,3), AMDZ(30,3), GRENS(30,2), DELTAZ(30)
          1, ID(30), THETAT(40), THETAL(40), PSIT(40), PSIL(40), TETMAX(40),
          1REGEN(1370)
0003      REAL*8 TETA(30), PSI(30), V(30)
0004      REAL K(30), K0(30,3), KBEM, KK, Kover, LAAGD(31)
C
C***** Common block *****
C
0005      COMMON PSI, TETA, V, ALFA, K0, K, LAAGD, DELTAT, A, AMDZ, GRENS, AGEM,
          1AG, GGA, REGEN, POOL, POOLMX, SRPLUS, TETMAX, GRWST, DD, HG, ASTER, SUMRNF
          1, ASTERB, BFACT, THETAL, THETAT, PSIL, PSIT, MAXL, MAXT, NL, IFLAG, IF2,
          1ID, ID1, ID2, JJ, J, I, NTOPL, NLOWL, ITRANS, NLAAG, NLayer, NPARTS, TYD
          1, IT, DELTAZ, PSITER, Kover, NDRAIN, NDRINT, NJAAR, NSKIP, VCONST, IRMOD, CS
          1, INPUT, OUTPUT, INFO, SKIP
C-----
C
0006      RAIN=REGEN(1)
0007      IF (RAIN.GT.0.) RAIN=RAIN*(1.+BFACT*PSI(1))
0009      POOL=POOL + RAIN * DELTAT + SRPLUS
0010      SRPLUS=0.0
C-----
0011      IF (POOL.GT.0.0) GO TO 20
0013      IF (IRMOD.EQ.1) GO TO 5
0015      IF (POOL.GE.POOLMX) GO TO 15
C-----
0017      RUNOFF=POOLMX-POOL
0018      POOL=POOLMX
0019      GO TO 10
C-----
0020      5  RUNOFF = CS * (POOL * POOL) * DELTAT
0021      POOL = POOL + RUNOFF
C-----
0022      10 SUMRNF=SUMRNF+RUNOFF
0023      15 RUNOFF=0.0
C
C      'SUMRNF' is cleared in the main program when
C      the value has been written to the outputfile
C-----
C      Calculate infiltration in case of rainfall
C
0024      Vin=((K(1)-K0(1,1))/(A(1,1)-1))-K0(1,1)
0025      P=POOL/DELTAT
0026      IF (P.GT.Vin) Vin=P
0028      POOL=POOL-Vin*DELTAT
0029      V(1)=Vin
0030      GO TO 25
C-----
C      Calculate upward flux in case of evaporation
C
0031      20 V(1)=POOL/DELTAT
0032      POOL=0.0
C-----
0033      25 RETURN
0034      END
    
```

```

0001      SUBROUTINE FLUX
          C *****
          C
          C-----
          C  FLUX belongs to the main program 'FLOWEX'
          C
          C
          C  This routine calculates the flux between the centres of the layers
          C  with eq. (5).
          C
          C  If the pressure heads of the relevant layers occur on different
          C  sections of the k(h)-relation, subroutine "ALGEM" is called to
          C  calculate an average value of ALFA
          C
          C  If a flux has to be calculated between layers with different soil-
          C  physical characteristics, subroutine "ITERA" is called to calculate
          C  the pressure head at the transition between these layers
          C
          C
          C  This subject has been described in detail in chapter 2.2
          C
          C-----
          C
          C***** Declarations *****
          C
          0002      DIMENSION ALFA(30,3), A(30,3), AHDZ(30,3), GRENS(30,2), DELTAZ(30)
          1, ID(30), THETAT(40), THETAL(40), PSIT(40), PSIL(40), TETMAX(40),
          1REGEN(1370)
          0003      REAL*8 TETA(30), PSI(30), V(30)
          0004      REAL K(30), KO(30,3), KGEM, KK, Kover, LAAGD(31)
          C
          C***** Common block *****
          C
          0005      COMMON PSI, TETA, V, ALFA, KO, K, LAAGD, DELTAT, A, AHDZ, GRENS, AGEM,
          1AG, GGA, REGEN, POOL, POOLMX, SRPLUS, TETMAX, GRWST, DD, HG, ASTER, SUMRNF
          1, ASTERB, BFACT, THETAL, THETAT, PSIL, PSIT, MAXL, MAXT, NL, IFLAG, IF2,
          1ID, ID1, ID2, JJ, J, I, NTOPL, NLOWL, ITRANS, NLAAG, NLAYER, NPARTS, TYD
          1, IT, DELTAZ, PSITER, Kover, NDRAIN, NDRINT, NJAAR, NSKIP, VCONST, IMOD, CS
          1, INPUT, OUTPUT, INFO, SKIP
          C-----
          C
          0006      DO 10 JJ=2, NL+1
          C          ^----- NL = number of layers involved in the calculation
          0007      ID1=ID(JJ-1)
          0008      ID2=ID(JJ )
          0009      AGEM=A(JJ, ID2)
          0010      IF(JJ.NE. ITRANS) GO TO 15
          C
          0012      CALL ITERA          ! calculate pressure head at transition
          0013      GO TO 10
          C
          C
          0014      15 IF(ID2.EQ.ID1) GO TO 20
          C          ^----- pressure heads on the same section
          C          of the k(h)-relation
          C
          0016      CALL ALGEM          ! calculate a new value of ALFA
          C
          0017      20 V(JJ)=(K(JJ)-K(JJ-1))/(AGEM-1)-K(JJ-1)          ! Eq. (5)
          0018      10 CONTINUE
          C
          0019      RETURN
          0020      END
    
```

```

0001      SUBROUTINE ALGEM
C          *****
C
C-----
C      ALGEM belongs to the main program 'FLOWEX'
C
C      This routine is called by subroutine "FLUX"
C
C      This routine calculates a value of ALFA when the pressure heads of
C      two adjacent layers are lying on different sections of
C      the k(h)-relation.
C      A new value is calculated according to the values of PSI.
C
C      The k(h)-relation consists of 3 sections, so there are following
C      possibilities: PSI1 - section 1      PSI2 - section 2
C                      PSI1 - section 1      PSI2 - section 3
C                      PSI1 - section 2      PSI2 - section 3
C      and reversed: PSI2 - section 1      PSI1 - section 2
C                      PSI2 - section 1      PSI1 - section 3
C                      PSI2 - section 2      PSI1 - section 3
C
C      The routine tests for all these cases. In order to come to an universal
C      calculation some exchange of data has to be done.
C
C      This subject has been described in detail in chapter 3.2.2
C-----
C***** Declarations *****
C
0002      DIMENSION ALFA(30,3), A(30,3), AHDZ(30,3), GRENS(30,2), DELTAZ(30)
          1, ID(30), THETAT(40), THETAL(40), PSIT(40), PSIL(40), TETMAX(40),
          1REGEN(1370)
0003      REAL*8 TETA(30), PSI(30), V(30)
0004      REAL K(30), KO(30,3), KGEM, KK, Kover, LAAGD(31)
C
C***** Common block *****
C
0005      COMMON PSI, TETA, V, ALFA, KO, K, LAAGD, DELTAT, A, AHDZ, GRENS, AGEM,
          1AG, GGA, REGEN, POOL, POOLMX, SRPLUS, TETMAX, GAWST, DD, HG, ASTER, SUMRNF
          1, ASTERB, BFACT, THETAL, THETAT, PSIL, PSIT, MAXL, MAXT, NL, IFLAG, IF2,
          1ID, ID1, ID2, JJ, J, I, NTOPL, NLOWL, ITRANS, NLAAG, NLAYER, NPARTS, TYD
          1, IT, DELTAZ, PSITER, Kover, NDRAIN, NDRINT, NJAAR, NSKIP, VCONST, IAMOD, CS
          1, INPUT, OUTPUT, INFO, SKIP
C-----
C      ** some simplifications **
C
0006      ID1 =ID(JJ-1)
0007      ID2 =ID(JJ )
0008      A1  =ALFA(JJ-1, ID1)
0009      A2  =ALFA(JJ , ID2)
0010      PSI1      =PSI(JJ-1)      ! upper layer
0011      PSI2      =PSI(JJ )      ! lower layer
C-----
0012      IF(PSI1.GT.PSI2) GO TO 5
C-----
C      ** exchange the values in case PSI2 > PSI1
C
0014      PSI1      =PSI2
0015      PSI2      =PSI(JJ-1)
0016      ID1 =ID2
0017      ID2 =ID(JJ-1)
0018      AHULP      =A1
0019      A1  =A2
0020      A2  =AHULP
C-----
0021      5 IDIFF      =IABS(ID2-ID1)
0022      IF(IDIFF.GT.1) GO TO 10
C-----
C      ----- in this case the values belong to section 1 and 3
C
0024      G1  =GRENS(JJ-1, ID1) ! adjacent
0025      G2  =G1                !
0026      SUPP =0.0              ! sections
0027      GO TO 15
C
    
```

```
0028      10 G1      =GRENS(JJ-1,1)      ! section 1
0029          G2      =GRENS(JJ-1,2)      !           and
0030          SUPP      =(G1-G2)*ALFA(JJ-1,2) !           section 3
-----
C
C
0031      15 AG      =((PSI1-G1)*A1 + (G2-PSI2)*A2 + SUPP) / (PSI1-PSI2) ! Eq. (44)
C
0032          AGEM      =EXP(AG * 0.5*(LAAGD(JJ)+LAAGD(JJ-1))) ! Part of eq. (5)
C
0033          RETURN
C-----
0034          END
```

```

0001      SUBROUTINE ITERA
C          *****
C
C-----
C      ITERA belongs to the main program 'FLOWEX'
C
C      This routine performs an iteration process to calculate the value
C      the pressure head (PSI) at the transition of topsoil to the subsoil.
C
C      If the pressure head calculated at the transition and the pressure head
C      of the lower layer are no lying on the same section of the k(h)-relation,
C      subr. "ALGEMI" is called in order to calculate the right value of ALFA.
C
C      Finally the flux between the relevant layers is calculated.
C
C      For a description of ID see the header above subroutine "KPSI"
C
C      This subject has been described in detail in par. 2.2
C-----
C***** Declarations *****
C
0002      DIMENSION ALFA(30,3), A(30,3), AHDZ(30,3), GRENS(30,2), DELTAZ(30)
          1, ID(30), THETAT(40), THETAL(40), PSIT(40), PSIL(40), TETMAX(40),
          1REGEN(1370)
0003      REAL*8 TETA(30), PSI(30), V(30)
0004      REAL K(30), KO(30,3), KGEM, KK, Kover, LAAGD(31)
C
C***** Common block *****
C
0005      COMMON PSI, TETA, V, ALFA, KO, K, LAAGD, DELTAT, A, AHDZ, GRENS, AGEM,
          1AG, GGA, REGEN, POOL, POOLMX, SRPLUS, TETMAX, GRWST, DD, HG, ASTER, SUMRNF
          1, ASTERB, BFACT, THETAL, THETAT, PSIL, PSIT, MAXL, MAXT, NL, IFLAG, IF2,
          1ID, ID1, ID2, JJ, J, I, NTOPL, NLOWL, ITRANS, NLAAG, NLayer, NPARTS, TYD
          1, IT, DELTAZ, PSITER, Kover, NDRAIN, NDRINT, NJAAR, NSKIP, VCONST, IRMOD, CS
          1, INPUT, OUTPUT, INFO, SKIP
C
C-----
C
0006      J1=JJ      ! index of layer below transition
0007      L=JJ-1     ! index of layer above transition
C
C
0008      PSITER=(PSI(L)+PSI(J1))/2. ! start iteration with average PSI
C
0009      DO 10 I2=1,100
C          ^----- maximum number of iterations
0010      BETA1      =KO(L, ID1)/(AHDZ(L, ID1)-1)
0011      BETA2      =(-K(L)*AHDZ(L, ID1))/(AHDZ(L, ID1)-1)
0012      GAMMA1     =K(J1)/(AHDZ(J1, ID2)-1)
0013      GAMMA2     =(KO(J1, ID2)*(-AHDZ(J1, ID2)))/(AHDZ(J1, ID2)-1)
0014      EXP4       =EXP(ALFA(L, ID1)*PSITER)
0015      EXP5       =EXP(ALFA(J1, ID2)*PSITER)
C
0016      TERM1      =(-ALFA(J1, ID2)*GAMMA2*EXP5*PSITER)+(ALFA(L
          1, ID1)*BETA1*EXP4*PSITER)
0017      TERM2      =ALFA(L, ID1)*BETA1*EXP4-ALFA(J1, ID2)*GAMMA2*EXP5
C
0018      PSINEW     =TERM1/TERM2
C          ^----- new calculated value of PSI [ eq. (19) ]
0019      DIFF       =PSINEW-PSITER
C          ^----- difference between previous and new value
0020      PSITER     =PSINEW
C          ^----- new value becomes previous value
C
0021      IF (ABS(DIFF).LT.0.0001) GO TO 15 ! iteration process finished
C-----
0023      ID1=1      !\
0024      IF (PSITER.LT.GRENS(L, 1)) ID1=2  !-> determine ID
0026      IF (PSITER.LT.GRENS(L, 2)) ID1=3  !/
0028      ID2=1      !\
0029      IF (PSITER.LT.GRENS(J1, 1)) ID2=2  !-> determine ID
0031      IF (PSITER.LT.GRENS(J1, 2)) ID2=3  !/
0033      10 CONTINUE
0034      STOP ' Iteration criterion not realizable in 100 steps'
C-----
    
```



```

      ID2=1      !\
0036 IF(PSITER.LT.GRENS(L,1)) ID2=2      !-> determine ID of final 'PSITER'
0038 IF(PSITER.LT.GRENS(L,2)) ID2=3      !/
      C
0040 Kover=K0(L, ID2)*EXP(ALFA(L, ID2)*PSITER)      ! eq. (4)
      C      ^----- hydraulic conductivity at transition
0041      AG=ALFA(L, ID2)
0042      IF(ID2.EQ. ID(L)) GO TO 5
      C-----
0044      CALL ALGEMI ! calculate average ALFA
      C-----
0045      5 V(J1)=(Kover-K(L))/(EXP(AG*LAGD(J1)*.5)-1)-K(L)      ! eq. (5)
      C      ^
      C      ^----- flux between the relevant layers
0046      RETURN
      C-----
0047      END
```

```

0001      SUBROUTINE ALGEMI
          *****
          C
          C=====
          C  ALGEMI belongs to the main program 'FLOWEX'
          C
          C  This routine is called by subroutine "ITERA" and is almost identical to
          C  "ALGEM" but dedicated to the transition between layers with different
          C  soil-fysical properties.
          C
          C  This routine calculates a value of ALFA if the pressure head of the
          C  layer above the transition belongs to an other section of the
          C  k(h)-relation than the pressure head at the transition (just calculated
          C  by subroutine "ITERA").
          C
          C  A new value is calculated according to the values of the pressure
          C  heads of these two layers.
          C
          C  The k(h)-relation consists of 3 sections, so there are following
          C  possibilities: PSI1 - section 1      PSI2 - section 2
          C                      PSI1 - section 1  PSI2 - section 3
          C                      PSI1 - section 2  PSI2 - section 3
          C  and reversed: PSI2 - section 1      PSI1 - section 2
          C                      PSI2 - section 1  PSI1 - section 3
          C                      PSI2 - section 2  PSI1 - section 3
          C
          C  The routine tests for all these cases. In order to come to an universal
          C  calculation some exchange of data has to be done.
          C
          C  This subject has been described in detail in chapter 3.2.2
          C=====
          C
          C***** Declarations *****
          C
          0002      DIMENSION ALFA(30,3),A(30,3),AHDZ(30,3),GRENS(30,2),DELTAZ(30)
          1, ID(30), THETAT(40), THETAL(40), PSIT(40), PSIL(40), TETMAX(40),
          0003      IREGEN(1370)
          0004      REAL*8 TETA(30), PSI(30), V(30)
          REAL K(30), KO(30,3), KGEM, KK, Kover, LAAGD(31)
          C
          C***** Common block *****
          C
          0005      COMMON PSI, TETA, V, ALFA, KO, K, LAAGD, DELTAT, A, AHDZ, GRENS, AGEM,
          1, GGA, REGEN, POOL, POOLMX, SRPLUS, TETMAX, GRWST, DD, HG, ASTER, SUMRNF
          1, ASTERB, BFACT, THETAL, THETAT, PSIL, PSIT, MAXL, MAXT, NL, IFLAG, IF2,
          1, ID, ID1, ID2, JJ, J, I, NTOPL, NLOWL, ITRANS, NLAGG, NLayer, NPARTS, TYD
          1, IT, DELTAZ, PSITER, Kover, NDRAIN, NDRINT, NJAAR, NSKIP, VCONST, IRMOD, CS
          1, INPUT, OUTPUT, INFO, SKIP
          C=====
          C
          C      ** some simplifications **
          C
          0006      J1 =JJ
          0007      L  =JJ-1
          0008      ID1 =ID(L)
          0009      A1  =ALFA(L, ID1)
          0010      A2  =ALFA(J1, ID2)
          0011      PSI1  =PSI(L) ! upper layer
          0012      PSI2  =PSITER ! transition
          C
          C
          0013      IF(PSI1.GT.PSI2) GO TO 5
          C      ----- exchange the values if PSI2 > PSI1
          0015      PSI1  =PSI2
          0016      PSI2  =PSI(L)
          0017      IDHULP  =ID1
          0018      ID1  =ID2
          0019      ID2  =IDHULP
          0020      AHULP  =A1
          0021      A1  =A2
          0022      A2  =AHULP
          C
          0023      5 IDIFF  =IABS(ID2-ID1)
          0024      IF(IDIFF.GT.1) GO TO 10
          C      ^----- in this case the values belong to section 1 and 3
          C
    
```

```
0026      G1      =GRENS(L, ID1)  ! adjacent
0027      G2      =G1              !
0028      SUPP    =0.0            ! values
0029      GO TO 15

C
C
0030      10 G1     =GRENS(L, 1)    ! section 1
0031      G2      =GRENS(L, 2)    !
0032      SUPP    =(G1-G2)*ALFA(L, 2) ! and section 3

C
C
0033      15 AG= ((PSI1-G1)*A1 + (G2-PSI2)*A2 + SUPP) / (PSI1-PSI2) ! Eq. (44)

C
C
0034      RETURN
C=====
0035      END
```

```

0001      SUBROUTINE GWST
C          *****
C
C-----
C      GWST belong to the main program 'FLOWEX'
C
C      This routine calculates the height of the groundwaternable above
C      the drainage system and the drainoutflow
C      The groundwaternable below the drainagesystem is calculated by 'GWS2'.
C
C      The drainoutflow is used by subroutine 'THETA' to calculate the
C      change in soil watercontent of the deepest unsaturated layer.
C
C      This subject has been described in detail in chap. 2.4.1 and 2.4.2.
C
C-----
C***** Declarations *****
C
0002      DIMENSION ALFA(30,3), A(30,3), AHDZ(30,3), GRENS(30,2), DELTAZ(30)
          1, ID(30), THETAT(40), THETAL(40), PSIT(40), PSIL(40), TETMAX(40),
          1REGEN(1370)
0003      REAL*8 TETA(30), PSI(30), V(30)
0004      REAL K(30), KO(30,3), KGEM, KK, Kover, LAAGD(31)
C
C***** Common block *****
C
0005      COMMON PSI, TETA, V, ALFA, KO, K, LAAGD, DELTAT, A, AHDZ, GRENS, AGEM,
          1AG, GGA, REGEN, POOL, POOLMX, SRPLUS, TETMAX, GRWST, DD, HG, ASTER, SUMRNF
          1, ASTERB, BFACT, THETAL, THETAT, PSIL, PSIT, MAXL, MAXT, NL, IFLAG, IF2,
          1ID, ID1, ID2, JJ, J, I, NTOPL, NLOWL, ITRANS, NLAAG, NLayer, NPARTS, TYD
          1, IT, DELTAZ, PSITER, Kover, NDRAIN, NDRINT, NJAAR, NSKIP, VCONST, IRMOD, CS
          1, INPUT, OUTPUT, INFO, SKIP
C-----
0006      H1=.0                      ! Calculate distance between
0007      DD 5 LD=J, NLAAG             ! the surface and the centre
0008      H1=H1+DELTAZ(LD)           ! of the deepest unsaturated
0009      5 CONTINUE                 ! layer
C-----
0010      IFLAG=0
0011      J=J-1
0012      IF(J.EQ.0) J=1
0014      IF(K(J).EQ.KO(J+1,1)) K(J)=K(J)-.02*KO(J+1,1)
0016      KGEM=(K(J)+KO(J+1,1))/2.
0017      T=KGEM/(KO(J+1,1)-KGEM)    ! calculate 't' from eq. (36)
C-----
0018      KK=KO(J+1,1)
0019      IF(J.GE.ITRANS-1) GO TO 10   ! Adjust the value of the
0021      KK=KO(J,1)                  ! drainage resistance A*
0022      AST=ASTERB                   ! from eq.(34) for layered
0023      GO TO 20                     ! soils.
0024      10 AST=ASTER
0025      20 AKO=AST*KK
C-----
0026      B=H1+T*(AKO-PSI(J))         !
0027      C=AKO*T*(H1+PSI(J))         ! Quadratic equation (36)
0028      HG=(B-SQRT(B**2-4*C))/2.    !
C-----
0029      IF(H1-DELTAZ(J+1).GT.HG) IFLAG=1 ! Test if a saturated layer
C                                     ! becomes unsaturated
0031      GRWST=DD+HG                 ! Groundwaternable below surface
0032      IF(GRWST.LT.-5.0) GO TO 30  ! Values >-5.0 are treated as 0.0
0034      HG=-(DD+POOL)              ! Add 'POOL' to groundwaternable
0035      GRWST=DD+HG
0036      30 VD=-HG/((HG/KK)+AST)     ! Drain outflow. Eq. (37)
C-----
0037      NL=J+1                      ! Make drain outflow
0038      V(NL)=VD                    ! the deepest flux
C-----
0039      RETURN
0040      END
    
```

```

ROUTINE GWS2
C *****
C
C-----
C   GWS2 belongs to the main program 'FLOWEX'
C
C   This routine calculates the groundwaterlevel below
C   the drainage system with eq. (42)
C   In case of rising groundwatertables jumps in the course are smoothed
C
C   This subject has been described in detail in chapter 2.4.3
C-----
C***** Declarations *****
C
0002   DIMENSION ALFA(30,3), A(30,3), AHDZ(30,3), GRENS(30,2), DELTAZ(30)
      1, ID(30), THETAT(40), THETAL(40), PSIT(40), PSIL(40), TETMAX(40),
      1REGEN(1370)
0003   REAL*8 TETA(30), PSI(30), V(30)
0004   REAL K(30), KO(30,3), KGEM, KK, Kover, LAAGD(31)
C
C***** Common block *****
C
0005   COMMON PSI, TETA, V, ALFA, KO, K, LAAGD, DELTAT, A, AHDZ, GRENS, AGEM,
      1AG, GGA, REGEN, POOL, POOLMX, SRPLUS, TETMAX, GRWST, DD, HG, ASTER, SUMRNF
      1, ASTERB, BFACT, THETAL, THETAT, PSIL, PSIT, MAXL, MAXT, NL, IFLAG, IF2,
      1ID, ID1, ID2, JJ, J, I, NTOPL, NLOWL, ITRANS, NLAG, NLayer, NPARTS, TYD
      1, IT, DELTAZ, PSITER, Kover, NDRAIN, NDRINT, NJAAR, NSKIP, VCONST, IRMOD, CS
      1, INPUT, OUTPUT, INFO, SKIP
C-----
C
0006   DATA OLDDP/0.0/
C-----
0007   IFLAG=0           ! clear flag possibly set in "GWST"
0008   IF2=0            ! clear flag possibly set in this routine
0009   NL=J
C-----
0010   DEPTH=0.0        ! calculate depth of
0011   DO 5 N1=1, J-1   ! deepest unsaturated
0012   DEPTH=DEPTH+DELTAZ(N1) ! layer
0013   5 CONTINUE
C-----
0014   IF(OLDDP.GT.DEPTH) GO TO 10 ! rising groundwatertable
0016   GRWST=-(DEPTH-PSI(J-1)) ! eq. (42)
0017   IF((PSI(J-1)+DELTAZ(J)).LT.0.0) IF2=1 ! next layer unsaturated
0019   OLDDP=DEPTH      ! save 'DEPTH'
0020   GO TO 15         ! return
C-----
0021   10 DEPTH2=DEPTH+0.5*DELTAZ(J) ! smooth out jumps in rising gr.w.table
0022   GRWST=-(DEPTH2-0.5*PSI(J-1))
0023   OLDDP=DEPTH
C
0024   15 RETURN
C-----
0025   END

```

```

0001      SUBROUTINE THETA
          C *****
          C
          C=====
          C THETA belongs to the main program 'FLOWEX'
          C
          C
          C This routine calculates for each unsaturated layer the soil water
          C content after every timestep according to eq. (6) and eq. (7).
          C
          C In case the soil water content exceeds the maximum (given by the soil
          C water retention curve) the surplus is added to the previous layer.
          C In case all layers are saturated the surplus will be added to POOL
          C in subroutine "INFIL"
          C
          C The drain discharge calculated in subroutine "GWST" is withdrawn
          C from the deepest unsaturated layer
          C
          C At the end the new number of unsaturated layers is calculated for the
          C next time step.
          C
          C This subject has been described in chapter 2.1
          C=====
          C ***** Declarations *****
          C
0002      DIMENSION ALFA(30,3),A(30,3),AHDZ(30,3),GRENS(30,2),DELTAZ(30)
          1, ID(30), THETAT(40), THETAL(40), PSIT(40), PSIL(40), TETMAX(40),
0003      1REGEN(1370)
          REAL*8 TETA(30), PSI(30), V(30)
0004      REAL K(30), KO(30,3), KGEM, KK, Kover, LAAGD(31)
          C
          C***** Common block *****
          C
0005      COMMON PSI, TETA, V, ALFA, KO, K, LAAGD, DELTAT, A, AHDZ, GRENS, AGEM,
          1AG, GGA, REGEN, POOL, POOLMX, SRPLUS, TETMAX, GRWST, DD, HG, ASTER, SUMRNF
          1, ASTERB, BFACT, THETAL, THETAT, PSIL, PSIT, MAXL, MAXT, NL, IFLAG, IF2,
          1ID, ID1, ID2, JJ, J, I, NTOPL, NLOWL, ITRANS, NLAAG, NLayer, NPARTS, TYD
          1, IT, DELTAZ, PSITER, Kover, NDRAIN, NDRINT, NJAAR, NSKIP, VCONST, IMOD, CS
          1, INPUT, OUTPUT, INFO, SKIP
          C
          C=====
          C
0006      NL2=NLAAG
0007      IF(NL.GT.NLAAG+1) NL2=NL-1
          C
0009      DO 20 L=1,NL-1
0010      L0=L-1
0011      L1=L
0012      L2=L+1
          C-----
          C
0013      DELTET=-((V(L1)-V(L2))/LAAGD(L2))*DELTAT ! Eq. (6)
0014      IF(L1.EQ. NL2) DELTET=-((V(L1)-V(L2))/LAAGD(L2))*DELTAT
          C
          C      ^
          C      ----- in this case we're in the deepest unsaturated
          C                      layer. V(L1) = drain discharge
          C
0016      TETA(L1)=TETA(L1)+DELTET ! Eq. (7)
0017      5 IF(TETA(L1).LE.TETMAX(L1)) GO TO 20 ! no saturation
0019      IF(L1.EQ.1) GO TO 10 ! saturation
          C-----
0021      TETA(L0)=TETA(L0)+((TETA(L1)-TETMAX(L1))
          1*LAAGD(L2)/LAAGD(L1)) ! calculate surplus and
          C ! add to previous layer
0022      TETA(L1)=TETMAX(L1) ! layer L1 saturated
          C
0023      L1=L0 ! look whether the addition of surplus causes an
0024      GO TO 5 ! overflow of the maximum soil-water content
          C-----
0025      10 SRPLUS=(TETA(1)-TETMAX(1))*-LAAGD(2) ! SRPLUS will be added to the pool
0026      TETA(1)=TETMAX(1) ! in case profile is saturated
0027      20 CONTINUE
          C-----
0028      NLayer=1 ! calc. # unsaturated layers
0029      DO 30 K1=1,30
0030      IF(TETA(K1).GE.TETMAX(K1)) GO TO 30
    
```

```
0032      NLayer=K1  
0033      C      ^----- number of unsaturated layers  
0033      30 CONTINUE  
0034      C  
0034      C-----  
0034      RETURN  
0035      END
```

```

0001      SUBROUTINE TETPSI
C          *****
C
C-----
C      TETPSI belongs to the main program 'FLOWEX'
C
C      This routine converts moisture contents to pressure heads with the
C      figures given at line 06 or line 10 of the INFO-file.
C      The value of the pressure head is found by means of interpolation
C
C      Moisture retention curve top-soil:
C
C      THETAT is the array containing the moisture contents
C      PSIT   is the array containing the pressure heads
C
C      Moisture retention curve sub-soil:
C
C      THETAL is the array containing the moisture contents
C      PSIL   is the array containing the pressure heads
C-----
C***** Declarations *****
C
0002      DIMENSION ALFA(30,3), A(30,3), AHDZ(30,3), GRENS(30,2), DELTAZ(30)
          1, ID(30), THETAT(40), THETAL(40), PSIT(40), PSIL(40), TETMAX(40),
          1REGEN(1370)
0003      REAL*8 TETA(30), PSI(30), V(30)
0004      REAL K(30), K0(30,3), KGEM, KK, Kover, LAGD(31)
C
C***** Common block *****
C
0005      COMMON PSI, TETA, V, ALFA, K0, K, LAGD, DELTAT, A, AHDZ, GRENS, AGEM,
          1AG, GGA, REGEN, POOL, POOLMX, SRPLUS, TETMAX, GRWST, DD, HG, ASTER, SUMANF
          1, ASTERB, BFACT, THETAL, THETAT, PSIL, PSIT, MAXL, MAXT, NL, IFLAG, IF2,
          1ID, ID1, ID2, JJ, J, I, NTOFL, NLOWL, ITRANS, NLAG, NLAYER, NPARTS, TYD
          1, IT, DELTAZ, PSITER, Kover, NDRAIN, NDRINT, NJAAR, NSKIP, VCONST, IRMOD, CS
          1, INPUT, OUTPUT, INFO, SKIP
C-----
C
0006      IF(J.GE.ITRANS) GO TO 50 !-----> subsoil
C          ^----- points to the subsoil
C-----
C      ** Top_soil **
C
0008      T=TETA(J)
0009      IF(T.GE.THETAT(MAXT)) GO TO 60      ! saturation
C
0011      DO 10 JO=1,MAXT
C          ^----- max number of items in the arrays PSIT and THETAT
0012      J1=JO+1
0013      IF(THETAT(JO).LE.T.AND.THETAT(J1).GT.T) GO TO 15
C          ^----- find the adjacent values of THETAT
0015      10 CONTINUE
0016      GO TO 55      ! Error message
C
C      ** find the value of TETA by interpolation
C
0017      15 PSI(J)=((T-THETAT(JO))/(THETAT(J1)-THETAT(JO)))*
          1(PSIT(J1)-PSIT(JO))+PSIT(JO)
0018      GO TO 80 ! return
C-----
C      ** sub-soil **
C
0019      50 T=TETA(J)
0020      IF(T.GE.THETAL(MAXL)) GO TO 60
0022      DO 70 JO=1,MAXL
C          ^----- max of items in the arrays PSIL and THETAL
0023      J1=JO+1
0024      IF(THETAL(JO).LE.T.AND.THETAL(J1).GT.T) GO TO 75
C          ^----- find the adjacent values of THETAL
0026      70 CONTINUE
0027      GO TO 55      ! ** error message **
C
C      ** find the value of TETA by interpolation **
C
0028      75 PSI(J)=((T-THETAL(JO))/(THETAL(J1)-THETAL(JO)))*
          1(PSIL(J1)-PSIL(JO))+PSIL(JO)
0029      GO TO 80 ! return

```



```
0030      60 PSI(J)=0.0          ! saturation
0031      80 RETURN
C
C-----
C      ** Error message **
C
0032      55 TYPE 100, I,J,T ! type timestep, layer and value of moist. cont.
0033      100 FORMAT(' *** Routine "TETPSI" aborted ***',//,
      *' value of TETA out of range !!',
      *' I=',I6,' J=',I3,' TETA=',F10.6,//)
C-----
0034      STOP
0035      END
```

```

0001      SUBROUTINE PSITET
C      *****
C
C=====
C      PSITET belongs to the main program 'FLOWEX'
C
C      This routine converts pressure heads to moisture contents with the
C      figures given at line 06 or line 10 of the INFO-file.
C      The value of the moisture content is found by means of interpolation
C
C      Moisture retention curve top-soil:
C
C      PSIT is the array containing the pressure heads
C      THETAT is the array containing the moisture contents
C
C      Moisture retention curve sub-soil:
C
C      PSIL is the array containing the pressure heads
C      THETAL is the array containing the moisture contents
C=====
C
C
C***** Declarations *****
0002      DIMENSION ALFA(30,3), A(30,3), AHDZ(30,3), GRENS(30,2), DELTAZ(30)
          1, ID(30), THETAT(40), THETAL(40), PSIT(40), PSIL(40), TETMAX(40),
0003      1REGEN(1370)
0004      REAL*8 TETA(30), PSI(30), V(30)
          REAL K(30), KO(30,3), KGEN, KK, Kover, LAAGD(31)
C
C***** Common block *****
0005      COMMON PSI, TETA, V, ALFA, KO, K, LAAGD, DELTAT, A, AHDZ, GRENS, AGEM,
          1AG, GGA, REGEN, POOL, POOLMX, SRPLUS, TETMAX, GRWST, DD, HG, ASTER, SUMRNF
          1, ASTERB, BFACT, THETAL, THETAT, PSIL, PSIT, MAXL, MAXT, NL, IFLAG, IF2,
          1ID, ID1, ID2, JJ, J, I, NTOPL, NLOWL, ITRANS, NLAAG, NLAYER, NPARTS, TYD
          1, IT, DELTAZ, PSITER, Kover, NDRAIN, NDRINT, NJAAR, NSKIP, VCONST, IRMOD, CS
          1, INPUT, OUTPUT, INFO, SKIP
C
C=====
0006      IF(J.GE.ITRANS) GO TO 50 !-----) subsoil
          ^----- points to the subsoil
C
C-----
C      ** top-soil **
0008      P=PSI(J)
0009      IF(P.GE.0.0) GO TO 35
0011      DO 10 J0=1,MAXT
C          ^--- max number if items in the arrays PSIT and THETAT
0012      J1=J0+1
0013      IF(PSIT(J0).LE.P.AND.PSIT(J1).GT.P) GO TO 15
C          ----- find the adjacent values of PSIT
0015      10 CONTINUE
0016      GO TO 55 ! Error message
C
C      ** find the value of TETA by interpolation
C
0017      15 TETA(J)=((P-PSIT(J0))/(PSIT(J1)-PSIT(J0)))*
          1(THETAT(J1)-THETAT(J0))+THETAT(J0)
0018      GO TO 80 ! return
C
0019      35 TETA(J)=THETAT(MAXT)
0020      GO TO 80 ! return
C-----
C      ** sub-soil **
C
0021      50 P=PSI(J)
0022      IF(P.GE.0.0) GO TO 60
0024      DO 70 J0=1,MAXL
C          ^----- max of items in the arrays PSIL and THETAL
0025      J1=J0+1
0026      IF(PSIL(J0).LE.P.AND.PSIL(J1).GT.P) GO TO 75
C          ----- find the adjacent values of PSIL
0028      70 CONTINUE
0029      GO TO 55 ! ** error message **
C
C      ** find the value of TETA by interpolation
    
```

```
C
0030 75 TETA(J)=((P-PSIL(J0))/(PSIL(J1)-PSIL(J0)))*
      1*(THETAL(J1)-THETAL(J0))+THETAL(J0)
0031 GO TO 80 ' return
C
0032 60 TETA(J)=THETAL(MAXL)
0033 80 RETURN
C
C-----
C ** Error message **
C
0034 55 TYPE 100, I,J,P ! type timestep, number layer and value of press. head
0035 100 FORMAT(' *** Routine "PSITET" aborted ***',//,
      *' value of PSI out of range !!',
      *' I=',I6,' J=',I3,' PSI=',F10.6,//)
C-----
0036 STOP
0037 END
```

```
0001      SUBROUTINE SORT(TSTEP,LEN)
C          *****
C
C=====
C
C      This is a general purpose routine to sort data in a vector
C      from low to high values using the QUICKSORT method.
C
C      The arguments required are 'ARRAY' : vector with data
C      'LEN' : number of data in 'ARRAY'
C      On return to the calling program the vector 'ARRAY' contains
C      the sorted data, the value of LEN remains unchanged.
C=====
0002      DIMENSION TSTEP(LEN)
C
0003      NTIME=LEN/2
0004      1 IFL=0
0005      DO 10 I=1,LEN-NTIME
0006          K=I+NTIME
0007          IF(TSTEP(K).GE.TSTEP(I)) GO TO 10
0009          T=TSTEP(I)
0010          TSTEP(I)=TSTEP(K)
0011          TSTEP(K)=T
0012          IFL=1
0013      10 CONTINUE
0014          IF(IFL.EQ.1) GO TO 1
0016          IF(NTIME.EQ.1) GO TO 5
0018          NTIME=NTIME/2
0019          GO TO 1
C
0020      5 RETURN
C=====
0021      END
```

Voorbeeld van een 'INFO'-file

```

-- COMMENT LINE ----- 00 --
ZEGVELD 29 NOV-83
-- NUMBER OF TIME SERIES DRAINDEPTHS DRAIN. INTENSITIES - 01 --
-- INIT. TIME NUMBER OF LAYERS NUMBER OF LAYERS TOPSOIL ----- 02 --
0.0 6 2
-- THICKNESS AND NUMBER OF EACH LAYER ----- 03 --
10.0 1
10.0 2
10.0 3
10.0 4
10.0 5
20.0 6
10.0 7
10.0 8
10.0 9
10.0 10
10.0 11
10.0 12
10.0 13
10.0 14
10.0 15
10.0 16
10.0 17
10.0 18
10.0 19
10.0 20
10.0 21
10.0 22
10.0 23
10.0 24
10.0 25
10.0 26
10.0 27
10.0 28
10.0 29
10.0 30
IRMOD POOLMX of CS POOL BFACTOR ----- 04 --
1 1. 0.0 0.003
DRAIN. INTENS. VCONST. ----- 05 --
-60.0 0.0018 -0.001
-- SOIL WATER RETENTION CURVE OF THE TOPSOIL ----- 06 --
.60 -105
.618 -101.07
.61 -93
.62 -81.5
.63 -70
.64 -60
.65 -50.5
.66 -44
.67 -38
.68 -32.5
.6875 -28.81
.69 -27.5
.70 -23.0
.71 -19
.72 -15.5
.73 -12.5
.74 -9.5
.75 -7.5
.76 -5.5
.77 -4
.78 -2.5
.79 0.00
-1 -1
-- K0 OF TOPSOIL ----- 07 --
5.0 1.4125 0.16596
-- ALFA OF TOPSOIL ----- 08 --
0.0767 0.03281 0.001162
-- BREAK POINTS IN K(h) RELATION ----- 09 --
-28.81 -101.07
-- SOIL WATER RETENTION CURVE OF THE SUBSOIL ----- 10 --
    
```

Voorbeeld van een 'INFO'-file (vervolg)

```
.60      -105
.618     -101.07
.61      -93
.62      -81.5
.63      -70
.64      -60
.65      -50.5
.66      -44
.67      -38
.68      -32.5
.6875    -28.81
.69      -27.5
.70      -23.0
.71      -19
.72      -15.5
.73      -12.5
.74      -9.5
.75      -7.5
.76      -5.5
.77      -4
.78      -2.5
.79      0.00
-1       -1
-- KO OF SUBSOIL ----- 11 --
5.0      1.4125 0.16596
-- ALFA OF SUBSOIL ----- 12 --
0.0767   0.03281 0.001162
-- BREAK POINTS IN K(h) RELATION ----- 13 --
-28.81   -101.07
-- FIRST INPUT FILE NAME ----- 14 --
RES.FLW
-- FIRST OUTPUT FILE NAME ----- 15 --
TEST01.065
-- CALCULATE TIME STEP ? ----- 16 --
YES
-- TIMESTEP ----- 17 --
.0822    (DAYS)
```

Voorbeeld van een 'METEO'-file

```

1370      number of records with data in this file
0.00      1- 9-65      1
0.00      2 each day is divided in five parts
-0.20     3 a negative value means evaporation
0.00      4
0.00      5
0.08      2- 9-65
0.08
0.08
0.08
0.08
0.60      3- 9-65
0.60
0.60
0.60
0.12      4- 9-65
0.14
0.12
0.12
0.12
0.00      5- 9-65
0.00
-0.07
0.00
0.00
0.10      6- 9-65
0.08
0.10
0.08
0.08
0.00      7- 9-65
0.00
-0.09
0.00
0.00
0.12      8- 9-65
0.12
0.12
0.12
0.14
0.02      9- 9-65
0.02
0.00
0.00
0.00
0.00      10- 9-65
-0.21
0
  
```

Gedeelte van een 'OUTPUT'-file

```
-- BREAK POINTS IN K(h) R.L.
--28.81000      -101.0700
-- FIRST INPUT FILE NAME -----
R65.FLW
-- FIRST OUTPUT FILE NAME -----
TEST01.065
-- CALCULATE TIME STEP ? ----- 10
Y
-- TIMESTEP ----- 17 --
0.0419
```

```
=====
```

5	544	444	344	244	144	594	0
5	544	444	344	244	144	594	0
5	661	455	345	244	144	594	0
5	643	476	350	245	145	595	0
5	618	481	356	247	145	595	0
5	548	475	358	250	147	597	0
5	498	455	357	251	149	599	0
5	470	435	352	251	150	600	0
6	447	418	344	250	150	0	600
5	429	402	336	247	149	599	0
5	238	337	325	243	147	597	0
5	150	229	275	232	143	593	0
5	115	161	202	199	130	580	0
5	97	123	148	150	94	544	0
5	87	101	112	104	53	503	0
4	148	117	99	72	454	0	
4	175	138	97	29	379	0	
3	191	145	85	336	0		
3	197	141	64	314	0		
3	194	132	51	301	0		
3	216	130	41	292	0		
3	227	133	38	288	0		
3	250	138	37	288	0		
3	246	142	40	291	0		
3	243	142	42	293	0		
3	216	138	43	294	0		
3	207	130	40	290	0		
3	196	123	35	285	0		
3	193	117	28	278	0		
3	189	111	20	270	0		
3	202	109	13	263	0		
3	208	110	12	262	0		
3	231	116	14	264	0		
3	225	121	20	270	0		
3	222	122	23	277			
3	193	117	27				
3	179	106					
3							