**SWAT source code modified for rain-on-snow events (snom.f file)**

subroutine snom

!! ~ ~ ~ PURPOSE ~ ~ ~

!! this subroutine predicts daily snom melt when the average air

!! temperature exceeds 0 degrees Celcius and rain-on-snow events

!! ~ ~ ~ INCOMING VARIABLES ~ ~ ~

!! name |units |definition

!! ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~

!! elevb(:,:) |m |elevation at center of band

!! elevb\_fr(:,:)|none |fraction of subbasin area within elevation

!! |band

!! iida |julian date |day being simulated (current julian date)

!! ihru |none |HRU number

!! pcpband(:,:) |mm H2O |precipitation for the day in band in HRU

!! precipday |mm H2O |precipitation on the current day in the HRU

!! sub\_sftmp |deg C |Snowfall temperature

!! |Mean air temperature at which precipitation

!! |is equally likely to be rain as snow/freezing

!! |rain.

!! sub\_smfmn |mm/deg C/day |Minimum melt rate for snow during year (Dec.

!! |21) where deg C refers to the air temperature

!! sub\_smfmx |mm/deg C/day |Maximum melt rate for snow during year (June

!! |21) where deg C refers to the air temperature

!! |SMFMX and SMFMN allow the rate of snow melt

!! |to vary through the year. These parameters

!! |are accounting for the impact of soil

!! |temperature on snow melt.

!! sub\_smtmp |deg C |Snow melt base temperature

!! |Mean air temperature at which snow melt will

!! |occur.

!! sno\_hru(:) |mm H2O |amount of water in snow in HRU on current day

!! snocov1 |none |1st shape parameter for snow cover equation

!! |This parameter is determined by solving the

!! |equation for 50% snow cover

!! snocov2 |none |2nd shape parameter for snow cover equation

!! |This parameter is determined by solving the

!! |equation for 95% snow cover

!! snocovmx |mm H2O |Minimum snow water content that corresponds

!! |to 100% snow cover. If the snow water content

!! |is less than SNOCOVMX, then a certain

!! |percentage of the ground will be bare.

!! snoeb(:,:) |mm H2O |snow water content in elevation band on

!! |current day

!! snotmp(:) |deg C |temperature of snow pack in HRU

!! snotmpeb(:,:)|deg C |temperature of snow pack in elevation band

!! tavband(:,:) |deg C |average temperature for the day in band in HRU

!! sub\_timp |none |Snow pack temperature lag factor (0-1)

!! |1 = no lag (snow pack temp=current day air

!! |temp) as the lag factor goes to zero, the

!! |snow pack's temperature will be less

!! |influenced by the current day's air

!! |temperature

!! tmpav(:) |deg C |average air temperature on current day for

!! |HRU

!! tmx(:) |deg C |maximum air temperature on current day for

!! |HRU

!! tmxband(:,:) |deg C |maximum temperature for the day in band in HRU

!! ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~

!! ~ ~ ~ OUTGOING VARIABLES ~ ~ ~

!! name |units |definition

!! ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~

!! precipday |mm H2O |amount of water in effective precipitation

!! |in HRU

!! precipdt(:) |mm H2O |precipitation for the time step during day

!! sno\_hru(:) |mm H2O |amount of water in snow in HRU on current day

!! snoeb(:,:) |mm H2O |snow water content in elevation band on

!! |current day

!! snofall |mm H2O |amount of precipitation falling as freezing

!! |rain/snow on day in HRU

!! snomlt |mm H2O |amount of water in snow melt for the day in

!! |HRU

!! snotmp(:) |deg C |temperature of snow pack in HRU

!! snotmpeb(:,:)|deg C |temperature of snow pack in elevation band

!! ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~

!! ~ ~ ~ LOCAL DEFINITIONS ~ ~ ~

!! name |units |definition

!! ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~

!! ib |none |counter

!! j |none |HRU number

!! smfac |

!! smleb |mm H2O |amount of snow melt in elevation band on

!! |current day

!! smp |mm H2O |precipitation on current day for HRU

!! snocov |none |fraction of HRU area covered with snow

!! sum |mm H2O |snow water content in HRU on current day

!! xx |none |ratio of amount of current day's snow water

!! |content to the minimum amount needed to

!! |cover ground completely

!! ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~

!! ~ ~ ~ SUBROUTINES/FUNCTIONS CALLED ~ ~ ~

!! Intrinsic: Real, Sin, Exp

!! ~ ~ ~ ~ ~ ~ END SPECIFICATIONS ~ ~ ~ ~ ~ ~

use parm

integer :: j, ib

real :: sum, smp, smfac, smleb

real :: xx, snocov

j = 0

j = ihru

sum =0.

smp =0.

isub = hru\_sub(j)

if (elevb(1,isub) > 0. .and. elevb\_fr(1,isub) > 0.) then

!! elevation bands

!! compute snow fall and melt for each elevation band

do ib = 1, 10

if (elevb\_fr(ib,isub) < 0.) exit

snotmpeb(ib,j) = snotmpeb(ib,j) \* (1.-sub\_timp(ib,isub)) +

& tavband(ib,j) \* sub\_timp(ib,isub)

if (tavband(ib,j) < sub\_sftmp(ib,isub)) then

!! compute snow fall if temperature is below sftmp

snoeb(ib,j) = snoeb(ib,j) + pcpband(ib,j)

snofall = snofall + pcpband(ib,j) \* elevb\_fr(ib,isub)

else

!! compute snow melt if temperature is above smtmp

if (tmxband(ib,j) > sub\_smtmp(ib,isub)) then

smfac = 0.

smleb = 0.

smfac = (sub\_smfmx(ib,isub) + sub\_smfmn(ib,isub)) / 2. +

& Sin((iida - 81) / 58.09) \*

& (sub\_smfmx(ib,isub) - sub\_smfmn(ib,isub)) / 2. !! 365/2pi = 58.09

smleb = smfac \* (((snotmpeb(ib,j) + tmxband(ib,j)) / 2.)

& - sub\_smtmp(ib,isub))

!! adjust for areal extent of snow cover

if (snoeb(ib,j) < snocovmx) then

xx = 0.

snocov = 0.

xx = snoeb(ib,j) / snocovmx

snocov = xx / (xx + Exp(snocov1 - snocov2 \* xx))

else

snocov = 1.

endif

!! Calculate snowmelt from rain-on-snow events and add to smleb

!! (Added by DM 6/18/2020)

if (snoeb(ib,j) > 0 .and. tavband(ib,j) > sub\_sftmp(ib,isub) .and.

& pcpband(ib,j) > 0) then

smleb = smleb + 6.12E-10 \* 24. \* ((tavband(ib,j) + 273.) \*\* 4. -

& 273. \*\* 4.) + 0.0125 \* pcpband(ib,j) \* 1 \*

& max(tavband(ib,j), 0.) + 8.5 \* 0.15 \* (24. / 6.) \* ((0.9 \*

& 2.7489E8 \* exp(-4278.63 / (tavband(ib,j) + 242.792)) - 6.11) +

& 0.00057 \* 33.86 \* (29.9 - 0.335 \* elevb(ib,isub) + 0.00022 \*

& elevb(ib,isub) \*\* 2.4) \* tavband(ib,j))

endif

smleb = smleb \* snocov

if (smleb < 0.) smleb = 0.

if (smleb > snoeb(ib,j)) smleb = snoeb(ib,j)

snoeb(ib,j) = snoeb(ib,j) - smleb

snomlt = snomlt + smleb \* elevb\_fr(ib,isub)

endif

endif

sum = sum + snoeb(ib,j) \* elevb\_fr(ib,isub)

smp = smp + pcpband(ib,j) \* elevb\_fr(ib,isub)

end do

!! add/sub aggregate snow fall and melt from effective precip

!! and snow cover

precipday = smp + snomlt - snofall

if (precipday < 0.) precipday = 0.

if (nstep > 0) then

do ii = 1, nstep

precipdt(ii+1) = precipdt(ii+1) + (snomlt - snofall) / nstep

if (precipdt(ii+1) < 0.) precipdt(ii+1) = 0.

end do

end if

sno\_hru(j) = sum

else

!! no elevation bands

ib = 1

!! estimate snow pack temperature

snotmp(j)=snotmp(j) \* (1. - sub\_timp(ib,isub)) + tmpav(j) \*

& sub\_timp(ib,isub)

if (tmpav(j) <= sub\_sftmp(ib,isub)) then

!! calculate snow fall

sno\_hru(j) = sno\_hru(j) + precipday

snofall = precipday

precipday = 0.

precipdt = 0.

endif

if (tmx(j) > sub\_smtmp(ib,isub) .and. sno\_hru(j) > 0.) then

!! adjust melt factor for time of year

smfac = 0.

snomlt = 0.

smfac = (sub\_smfmx(ib,isub) + sub\_smfmn(ib,isub)) / 2. +

& Sin((iida - 81) / 58.09) \*

& (sub\_smfmx(ib,isub) - sub\_smfmn(ib,isub)) / 2. !! 365/2pi = 58.09

snomlt = smfac \* (((snotmp(j)+tmx(j))/2.)-sub\_smtmp(ib,isub))

!! adjust for areal extent of snow cover

if (sno\_hru(j) < snocovmx) then

xx = 0.

xx = sno\_hru(j) / snocovmx

snocov = xx / (xx + Exp(snocov1 - snocov2 \* xx))

else

snocov = 1.

endif

!! Calculate snowmelt from rain-on-snow events and add to snomlt

!! (Added by DM 6/18/2020)

if (sno\_hru(j) > 0 .and. tmpav(j) > sub\_sftmp(ib,isub) .and.

& precipday > 0) then

snomlt = snomlt + 6.12E-10 \* 24. \* ((tmpav(j) + 273.) \*\* 4. - 273. \*\*

& 4.) + 0.0125 \* precipday \* 1 \* max(tmpav(j), 0.) + 8.5 \* 0.15 \*

& (24. / 6.) \* ((0.9 \* 2.7489E8 \* exp(-4278.63 / (tmpav(j) +

& 242.792)) - 6.11) + 0.00057 \* 33.86 \* (29.9 - 0.335 \* 200. +

& 0.00022 \* 200. \*\* 2.4) \* tmpav(j))

endif

!! Finish snowmelt calculations

snomlt = snomlt \* snocov

if (snomlt < 0.) snomlt = 0.

if (snomlt > sno\_hru(j)) snomlt = sno\_hru(j)

sno\_hru(j) = sno\_hru(j) - snomlt

precipday = precipday + snomlt

if (nstep > 0) then

do ii = 1, nstep

precipdt(ii+1) = precipdt(ii+1) + snomlt / nstep

end do

end if

if (precipday < 0.) precipday = 0.

else

snomlt = 0.

end if

end if

return

end