

---

## CODES FOR CHAPTER 11

---

### **voigt.f**

Fortran77 subroutine `voigt(S,bL,bD,deta,keta)` calculates the spectral absorption coefficient for a Voigt-shaped line based on the fast algorithm by Humlíček [1].

Input:

- `S` = is the line intensity  $S$ , in  $\text{cm}^{-2}$ ,
- `bL` = is the Lorentz line width  $b_L$ , in  $\text{cm}^{-1}$ ,
- `bD` = is the Doppler line width  $b_D$ , in  $\text{cm}^{-1}$ ,
- `deta` = is the spectral distance  $\Delta\eta$  away from the line center, at which  $\kappa_\eta$  is to be evaluated.

Output:

- `keta` = is the spectral absorption coefficient of the Voigt line  $\kappa_\eta$  at  $\eta = \eta_0 \pm \Delta\eta$ , where  $\eta_0$  is the wavenumber of the line center.

### **nbkdistdb.f90**

Program `nbkdistdb` is a Fortran90 code to calculate narrow band  $k$ -distributions for a number of temperatures and a number of wavenumber ranges, for a gas mixture containing  $\text{CO}_2$ ,  $\text{H}_2\text{O}$ ,  $\text{CH}_4$  and soot. The spectral absorption coefficient is calculated directly from the HITRAN or HITEMP databases.

Input:

- `Tmin` = minimum temperature for which a  $k$ -distribution is to be calculated, in K,
- `Tmax` = maximum temperature for which a  $k$ -distribution is to be calculated, in K,
- `numT` = number of different temperatures to be considered; equally spaced between `Tmin` and `Tmax`,
- `P` = total pressure of gas mixture, bar,
- `xmfr(3)` = mole fraction vector; `xmfr(1)`= mole fraction of  $\text{CO}_2$ , `xmfr(2)`= mole fraction of  $\text{H}_2\text{O}$ , `xmfr(3)`= mole fraction of  $\text{CH}_4$ ,
- `fvsoot` = volume fraction of soot,
- `nsoot`, `ksoot` = complex index of refraction for the soot; its absorption coefficient is assumed linear in wavenumber, using equation (12.112),
- `wvnm.b` = minimum wavenumber considered,  $\text{cm}^{-1}$ ,
- `wvnm.e` = maximum wavenumber considered,  $\text{cm}^{-1}$ ,
- `wvnm.buf` = line wing influence of spectral lines centered in the wavenumber range `wvnm.buf`  $\text{cm}^{-1}$  below `wvnm.b` and above `wvnm.e` are considered in the absorption coefficient calculation,  $\text{cm}^{-1}$ ,
- `wvnm.st` = wavenumber step (equally spaced) with which the absorption coefficient for the mixture is calculated from the HITRAN or HITEMP database,  $\text{cm}^{-1}$ ,
- `kdrnge` = wavenumber range for individual  $k$ -distributions; `wvnm.e-wvnm.b` should be an integer multiple of `kdrnge`, in  $\text{cm}^{-1}$ ,
- `n.pwrk` = number of different  $k$ -bin values considered in the construction of the  $k$ -distribution,
- `pwr` = exponent for  $k$ -bin values spacing:  $k$ -bins are equally spaced in  $k^{\text{pwr}}$  between `kmin` (=minimum  $k$  to be considered) and `kmax` (=maximum absorption coefficient across spectrum).
- `nq` = number of quadrature points for radiative calculations, i.e., the number of RTE evaluations to be performed before spectral integration (over cumulative  $k$ -distribution  $g$ ),
- `iwr` = absorption coefficient switch: `iwr=0` to make a single complete run, i.e., evaluating  $\kappa_\eta$  from HITRAN or HITEMP (without storing them), followed by generation of  $k$ -distributions, `irw=1` same, but absorption coefficient is stored for future use, and `iwr=2`: precalculated absorption coefficients are read in and  $k$ -distributions are generated.

**ipl** = linear vs. pressure-based absorption coefficient switch:  
**ipl=0**: calculate linear absorption coefficient, in  $\text{cm}^{-1}$   
**ipl=1**: calculate pressure-based absorption coefficient (allowed only for single absorbing gas!), in  $\text{cm}^{-1} \text{bar}^{-1}$ ; if the pressure-based absorption coefficient for a dilute gas is desired, set **xmfr=1.d-3** (=0.1%)  
**ipr** = output switch: see under output  
Output:

**ipr=0**: For each of the **numkd=wvnm\_e-wvnm\_b/kdrnge** narrow band ranges only the **nq** quadrature points, weights, and  $k(T, g)$  (for all temperatures) are printed: the first line of the output file, called **nbkvsgq.dat** by default, contains the first and last wavenumbers of the first narrow band range, followed by **nq** lines containing **gq** (the  $i$ -th quadrature point), **wq** (the  $i$ -th quadrature weight), and **numT** values of  $k_q [= k(T_j, g_i)]$ ; one for each temperature  $T_j$ . This is followed by a line containing the first and last wavenumbers of the second narrow band range, etc.

**ipr=1**: Besides the output for **ipr=0** a second output file is prepared with the complete  $k$ -distribution information, i.e., for each narrow band and each temperature all **n\_pwrk** values of  $k, f$  and  $g$  are printed: the first line of the output file, called **nbkvsg.dat** by default, contains the first temperature and first and last wavenumbers of the first narrow band range, followed by **n\_pwrk-1** lines containing **k** (the  $i$ -th  $k$ -bin value), **ff** [its  $k$ -distribution value  $f(k)$ ], and **gg** [its cumulative  $k$ -distribution value  $g(k)$ ]. This is followed by a line containing the second temperature and first and last wavenumbers of the first narrow band range, etc., looping over all temperatures and narrow band ranges.

**ipr=2**: Only the complete  $k$ -distribution information is printed, i.e., only output file **nbkvsg.dat** is generated.

### Example:

We consider a set of narrow band  $k$ -distributions for a linear absorption coefficient (**ipl=0**) of pure  $\text{CO}_2$ , for a mole fraction of 10% (**xmfr(3)=(/0.1d0,0.d0,0.d0/)**). The absorption coefficient is calculated in this run (**iwr=1**), and is stored in file **C:\absco\absctmp.dat** (for a wavenumber range from  $2320 \text{ cm}^{-1}$  to  $2380 \text{ cm}^{-1}$ , but also considering lines centered at wavenumbers as low as  $2315 \text{ cm}^{-1}$  and as high as  $2385 \text{ cm}^{-1}$ , **wvnmbuf=5.**) with a  $\delta\eta = 0.001 \text{ cm}^{-1}$ . We will calculate the  $k$ -distributions for 4 temperatures, equally spaced between  $T_{\min} = 300 \text{ K}$  and  $T_{\max} = 1200 \text{ K}$  (**numT=4**): this results in the 4 temperatures of 300 K, 600 K, 900 K and 1200 K. Each  $k$ -distribution will be over a range of  $\Delta\eta = 10 \text{ cm}^{-1}$  wavenumbers (**kdrnge=10.**), i.e., there will be 6 narrow bands. We will use 500  $k$ -bins (**n\_pwrk=500**) with **pwr=0.1** (this spreads the  $k$ -bins over many orders of magnitude, but places more and more bins into large magnitudes; see output file). We also set **klmin=10<sup>-9</sup> (cm<sup>-1</sup>)**, i.e., we will consider absorption coefficient contributions as small as  $10^{-9} \text{ cm}^{-1}$ . Finally, we set **ipr=1** and **nq=10**, i.e., besides truncated  $k$ -distributions ready-made for numerical quadrature, using 10 quadrature points, we want to also print to file the full  $k$ -distributions. The top of the program with input parameters, therefore, looks like this:

```

MODULE Key
  IMPLICIT NONE
!HITRAN/HITEMP DATABASE
  INTEGER :: lu
  INTEGER,PARAMETER :: rows=1400000
  DOUBLE PRECISION,PARAMETER :: wvnm_b=2320.d0,wvnm_e=2380.d0,wvnmbuf=5.d0,wvnmst=0.001d0
  DOUBLE PRECISION :: data(rows,6),wvnm_1=wvnm_b-wvnmbuf,wvnm_r=wvnm_e+wvnmbuf
END MODULE Key

PROGRAM Main
  USE Key
! Input parameters
  INTEGER,PARAMETER :: numT=4,n_pwrk=500,nq=10,iwr=1,ipl=0,ipr=1
  DOUBLE PRECISION,PARAMETER :: P=1.d0,Tmin=300d0,Tmax=1200d0,kdrnge=10.

```

```

DOUBLE PRECISION,PARAMETER :: xmfr(3)=(/0.10d0,0.00d0,0.d0/),pwr=0.1d0,klmin=1.d-9
DOUBLE PRECISION,PARAMETER :: fvsoot=0.d-6,nsoot=1.89d0,ksoot=0.92d0
where we have changed the values for wvnm_b, wvnm_e, wvnmst, numT, n_pwrk, iwr, ipr, nq,
Tmin, Tmax and xmfr to fit our needs. Also, in this simulation we have set file names as
! Open output files
IF(ipr<2) OPEN(7,FILE='nbkvsgco2.dat',STATUS='unknown')
IF(ipr>0) OPEN(8,FILE='nbkvsgco2.dat',STATUS='unknown')
! File containing absorption coefficient
IF(iwr>0) OPEN(9,FILE='C:\absco\absctmp.dat',STATUS='unknown')

```

i.e., the absorption coefficient as calculated here is placed into `c:\absco\absctmp.dat` (and can be reused later by setting `iwr=2`), while the long  $k$ -distribution output (500 values for each temperature and narrow band) will be put into `nbkvsgco2.dat`, and the short, quadrature-ready output into `nbkvsgqco2.dat`. Note that the header lines for `absctmp.dat` are formatted such that the absorption coefficient can be plotted from them using the Tecplot drafting package. The other output files will need some reformatting before they can be used for plotting.

We will also assume that Numerical Recipes subroutines are available, leaving the following lines unchanged:

```

! Selection of g-values for numerical quadrature, using a Numerical Recipes routine
! If Numerical Recipes is not available, set nq=12, comment out the following 8 lines of code,
! and uncomment the 5-line REAL declaration following it
REAL          :: gqs(nq),wqs(nq),kq(numt,nq),gq(nq),wq(nq),gaujac,alf=3.,bet=-.6,sum
! Get quadrature coefficients from Numerical recipies
sum=0.
CALL GAUJAC(gqs,wqs,nq,alf,bet)
do iq=1,nq
  gq(iq)=0.5*(1.-gqs(iq))
  wq(iq)=wqs(iq)/(2.**((alf+bet+1)*gq(iq)**alf*(1.-gq(iq))**bet)
  sum=sum+wq(iq)
enddo
! Correction to make sum(wq)=1
wq=wq/sum
! End quadrature coefficients from Numerical recipies
! Selection of precalculated g-values for numerical quadrature, for nq=12,alf=3.,bet=0.
! REAL          :: kq(numt,nq), &
!      gq(nq)=(/ 5.120075E-02,1.170678E-01,2.015873E-01,3.007074E-01,4.095012E-01,5.225285E-01, &
!      6.341280E-01,7.387071E-01,8.310236E-01,9.064499E-01,9.612060E-01,9.925594E-01/),&
!      wq(nq)=(/ 5.556622E-02,7.576839E-02,9.258290E-02,1.048306E-01,1.118451E-01,1.132605E-01, &
!      1.090012E-01,9.927844E-02,8.457905E-02,6.563999E-02,4.341329E-02,1.904792E-02/)

```

This will calculate quadrature points `gq` and weights `wq` using Gaussian quadrature of moments (`alf=3` sets 3rd order moments). For users without access to Numerical Recipes the `gq` and `wq` calculated here have been put in data statements and may be used instead by following the guidelines above.

The absorption coefficient placed into `c:\absco\absctmp.dat` has the following form:

```

variables = "wvn" "absco0300K" "absco0600K" "absco0900K" "absco1200K"
zone i= 60001
2320.00000  0.43878E+00  0.34411E+00  0.33293E+00  0.35420E+00
2320.00100  0.43694E+00  0.34266E+00  0.33335E+00  0.35600E+00
2320.00200  0.43512E+00  0.34125E+00  0.33386E+00  0.35783E+00
2320.00300  0.43333E+00  0.33988E+00  0.33447E+00  0.35968E+00
2320.00400  0.43157E+00  0.33856E+00  0.33516E+00  0.36155E+00
.
.
.

```

It is formatted for easy plotting using Tecplot, and has 60,001 absorption coefficient values between  $2320\text{ cm}^{-1}$  and  $2380\text{ cm}^{-1}$ , spaced  $0.001\text{ cm}^{-1}$  apart. The output file `nbkvsgco2.dat` has this form:

T= 300.K, wvnm\_lft= 2320.000000cm-1, wvnm\_rgt= 2330.000000cm-1

k	f	g
0.325271D+00	0.615250D-02	0.625249D-02
0.328970D+00	0.262559D-02	0.887808D-02
0.332708D+00	0.209533D-02	0.109734D-01
0.336484D+00	0.188093D-02	0.128543D-01
0.340299D+00	0.183458D-02	0.146889D-01
.		
.		
.		
0.277993D+02	0.340523D-03	0.997833D+00
0.280016D+02	0.402225D-03	0.998235D+00
0.282052D+02	0.521735D-03	0.998757D+00
0.284102D+02	0.124290D-02	0.100000D+01

T= 600.K, wvnm\_lft= 2320.000000cm-1, wvnm\_rgt= 2330.000000cm-1

k	f	g
0.187475D+00	0.525121D-02	0.535120D-02
0.189577D+00	0.199556D-02	0.734676D-02
0.191700D+00	0.138701D-02	0.873377D-02
.		
.		
.		

Finally, output file nbkvsgqco2.dat contains quadrature  $k$ -values as:

wvnm\_lft= 2320.000000cm-1, wvnm\_rgt= 2330.000000cm-1

gq	wq	kq(T1)	kq(T2)	...
0.729136D-01	0.813193D-01	0.400407D+00	0.242578D+00	0.183572D+00 0.160547D+00
0.165015D+00	0.108536D+00	0.541222D+00	0.297547D+00	0.226229D+00 0.204381D+00
0.280173D+00	0.128592D+00	0.672925D+00	0.335421D+00	0.278275D+00 0.240816D+00
0.410404D+00	0.139547D+00	0.867542D+00	0.418648D+00	0.336797D+00 0.295848D+00
0.546441D+00	0.140538D+00	0.118950D+01	0.584868D+00	0.422875D+00 0.361822D+00
0.678556D+00	0.131471D+00	0.163219D+01	0.902429D+00	0.560377D+00 0.425924D+00
0.797291D+00	0.112988D+00	0.286678D+01	0.156588D+01	0.795057D+00 0.518610D+00
0.894140D+00	0.864116D-01	0.739453D+01	0.262995D+01	0.130120D+01 0.731875D+00
0.962165D+00	0.536406D-01	0.168294D+02	0.783404D+01	0.286791D+01 0.123165D+01
0.996473D+00	0.169570D-01	0.268487D+02	0.142687D+02	0.658947D+01 0.326066D+01

wvnm\_lft= 2330.000000cm-1, wvnm\_rgt= 2340.000000cm-1

gq	wq	kq(T1)	kq(T2)	...
0.729136D-01	0.813193D-01	0.716314D+00	0.299593D+00	0.223759D+00 0.171072D+00
0.165015D+00	0.108536D+00	0.788507D+00	0.371208D+00	0.277792D+00 0.221426D+00
0.280173D+00	0.128592D+00	0.943415D+00	0.436240D+00	0.339969D+00 0.280705D+00
.				
.				
.				

for each of the 6 narrow bands.

Note that the code has an accuracy-checking mechanism built in: an average narrow band absorption coefficient is calculated directly through line-by-line integration of the absorption coefficient, equation (11.60), and is compared with the mean absorption coefficient as calculated from the  $k$ - $g$ -distribution. If the discrepancy exceeds 0.5% a message is printed to the screen, warning that  $k$ -bin spacing is too coarse ( $n\_pwrk$  too small) to properly resolve the absorption coefficient. For the above example, the choice of  $n\_pwrk=500$  results in an error larger than 0.5% only for 2340–2350  $\text{cm}^{-1}$  narrow band at 300 K (0.52%), as indicated by the warning message.

#### **nbkdistsg.f90**

Subroutine nbkdistsg calculates a single narrow band  $k$ -distribution from a given set of spectral absorption

coefficients and corresponding wavenumbers.

Input:

**Deta** = wavenumber range for which a  $k$ -distribution is to be calculated, in  $\text{cm}^{-1}$ ,  
**numk** = number of absorption coefficient datapoints, equally spaced in wavenumbers,  
**n\_pwrk** = number of  $k$ -boxes for  $k$ -distribution,  
**pwr** = exponent for setting of  $k$ -box values; i.e.,  $k$ -values are chosen in equal steps of  $k^{\text{pwr}}$ ,  
**nq** = number of quadrature points for Gaussian quadrature,  
**ipr** = print switch: **ipr=0**: prints  $k$  and  $w$  (Gaussian quadrature weights) vs.  $g$  only for Gaussian A quadrature points; **ipr=1**: prints  $k$  and  $w$  vs.  $g$  for Gaussian quadrature points, as well as  $k$  vs.  $f$  and  $g$  for all **n\_pwrk**  $k$ -bins; **ipr=2**: prints only  $k$  vs.  $f$  and  $g$  for all **n\_pwrk**  $k$ -bins.  
 file named **absco.dat** containing absorption coefficient data is required: The first line must contain **numk** and **Deta** (in I5,F7.4 format); second through (**numk**+1)th lines contain **wvnm**, **absco** (in e12.4 format).  
 Output:

**nbkvsg.dat**: Output file in Tecplot format (if **ipr=1** or **2**), containing one line giving wavenumber range, then  $k$ ,  $f$ ,  $g$  for **n\_pwrk**  $k$ -values.

**nbkvsgq.dat**: Output file in Tecplot format (if **ipr=0** or **1**), containing one line giving wavenumber range, then  $k$ ,  $w$ ,  $g$  for **nq** Gaussian quadrature points (**nq**=12 set as default: see discussion on Gaussian quadrature in **nbkdistsdb.f90**).

**nbkdistsg.f90** is a streamlined version of **nbkdistsdb.f90** and, thus, much of the discussion in the example for **nbkdistsdb.f90** pertains here, as well. As provided, **nbkdistsg.f90** is embedded in a stand-alone program called **nbkdistsngl.f90**, which first calculates the absorption coefficient data for the mixture in Example 11.5, then calls **nbkdistsg.f90** to determine the  $k$ -distribution given in Fig. 11-19.

#### **wbmxxx.f, wbmxxxcl.f, wbmxxxcl.exe**

Double precision Fortran77 subroutines **wbmxxx(T,PSIr,PHIr)**, where **xxx** stands for the different gases **h20**, **co2**, **ch4**, **co**, **no** and **so2**, calculate for a given temperature **T** the ratios  $\text{PSIr} = \Psi^*(T)/\Psi^*(T_0)$  [from equations (11.144) and (11.148)] and  $\text{PHIr} = \gamma/\gamma_0 = \sqrt{T_0/T} \Phi(T)/\Phi(T_0)$  [from equation (11.149)], i.e., the functions shown in Figs. 11-23 through 11-25, for all bands given in Table 11.3 in the order as listed (in order of decreasing band center wavelengths). For example, a call to **wbmch4(1200.,PSIr,PHIr)** would produce 4 values each for **PSIr** and **PHIr**, and **PSIr(3)** would contain the value of  $\Psi^*(1200\text{K})/\Psi^*(T_0) = 1.29540$  for the  $2.4\mu\text{m}$  band of methane, etc. The stand-alone programs **wbmxxxcl.f** perform the identical calculations, prompting the user for input (**T**), and printing **PSIr** and **PHIr** to the screen for all bands listed in Table 11.3.

#### **emwbm.f, ftwbm.f, wangwbm.f**

Double precision Fortran77 functions to calculate the nondimensional total band absorptance  $A^*$  from the Edwards and Menard model, Table 11.2 (**emwbm(tau,beta)**), the Felske and Tien model, equation (11.156) (**ftwbm(tau,beta)**), and the Wang model, equation (11.158) (**wangwbm(tau,beta)**).

#### **wbmodels.f, wbmodels.exe**

Stand-alone double precision Fortran77 front end for the functions **emwbm**, **ftwbm** and **wangwbm**; the user is prompted to input **tau** ( $= \tau_0$ , optical thickness at band center) and **beta** ( $= \beta$ , overlap parameter); the nondimensional total band absorptance  $A^*$  is printed to the screen, as calculated from three band models (Edwards and Menard, Felske and Tien, and Wang models).

#### **wbkvsg.f**

Double precision Fortran77 subroutine **wbkvsg(beta,kmax,kmin,n,k,g)** calculates the  $\kappa^*$  vs.  $g^*$  distribution of equation (11.170).

Input:

beta =  $\beta$ , the overlap parameter,  
kmax =  $\kappa_{\max}^*$ , the maximum  $\kappa^*$ -value to be output,  
kmin =  $\kappa_{\min}^*$ , the minimum  $\kappa^*$ -value to be output,  
n = the number of  $\kappa^*$  and  $g^*$  values to be output, [equally spaced in  $\ln(\sqrt{\kappa^*})$ ],

Output:

k, g =  $\kappa^*, g^*$ , n values each for  $\kappa^*$  and  $g^*$ , [equally spaced in  $\ln(\sqrt{\kappa^*})$ ].

The integral in equation (11.170) is evaluated by first transforming the integration variable from  $\kappa^*$  to  $a = \ln(\sqrt{\kappa^*})$ , or

$$g^* = \int_{\ln(\sqrt{\kappa^*})}^{a_{\max}=\ln(\sqrt{10^5})} \left[ \operatorname{erfc}(\sqrt{\beta} \sinh a) - e^{\beta} \operatorname{erfc}(\sqrt{\beta} \cosh a) \right] da,$$

followed by a simple Newton-Cotes integration. Beginning point of the integration is  $a_{\max}$  and a minimum step size for the numerical integration is determined and used. However, only values for  $k_{\max} > \kappa^* > k_{\min}$  for n values equally spaced in  $a$  are output to arrays k and g.

Notes:

- (i) Values of  $k_{\max} > 10^5$  are truncated;
- (ii) Program assumes availability of double precision functions `derfc`, `dcosh` and `dsinh`.

As an example we consider the  $k$ -distribution of Example 10.9. Writing a small Fortran calling program

```

program callwbkvsg
integer n,i
real*8 beta,kmax,kmin,k(1000),g(1000),c1,c2,kdim,deta
OPEN(9,FILE='wbkvsg.dat',STATUS='unknown')
beta=0.211d0
kmax=1.d1
kmin=1.d-3
n=40
c1=54.84*41.2/138.15/100.    ! rho-alpha/omega with kappa in cm-1
c2=138.15/2.                ! omega/2
WRITE(9,9)
call wbkvsg(beta,kmax,kmin,n,k,g)
DO i=1,n
  kdim=c1*k(i)
  deta=c2*g(i)
  WRITE(9,10) k(i),g(i),kdim,deta
ENDDO
CLOSE(9)
9  FORMAT('      kstar      gstar      kdim      deta')
10 FORMAT(3f10.5,f8.2)
stop
end

```

leads to

	kstar	gstar	kdim	deta
10.00000	0.00942	1.63547	0.65	
7.89652	0.01448	1.29146	1.00	
6.23551	0.02141	1.01980	1.48	
4.92388	0.03064	0.80529	2.12	
3.88816	0.04264	0.63590	2.95	
3.07029	0.05791	0.50214	4.00	
2.42446	0.07702	0.39651	5.32	
1.91448	0.10058	0.31311	6.95	
1.51178	0.12924	0.24725	8.93	
1.19378	0.16374	0.19524	11.31	
0.94267	0.20484	0.15417	14.15	
0.74438	0.25339	0.12174	17.50	

0.58780	0.31028	0.09613	21.43
0.46416	0.37645	0.07591	26.00
0.36652	0.45290	0.05994	31.28
0.28943	0.54061	0.04733	37.34
0.22855	0.64057	0.03738	44.25
0.18047	0.75370	0.02952	52.06
0.14251	0.88079	0.02331	60.84
0.11253	1.02244	0.01840	70.62
0.08886	1.17894	0.01453	81.44
0.07017	1.35022	0.01148	93.27
0.05541	1.53570	0.00906	106.08
0.04375	1.73426	0.00716	119.79
0.03455	1.94425	0.00565	134.30
0.02728	2.16360	0.00446	149.45
0.02154	2.38997	0.00352	165.09
0.01701	2.62110	0.00278	181.05
0.01343	2.85504	0.00220	197.21
0.01061	3.09039	0.00173	213.47
0.00838	3.32632	0.00137	229.77
0.00661	3.56244	0.00108	246.08
0.00522	3.79859	0.00085	262.39
0.00412	4.03475	0.00067	278.70
0.00326	4.27092	0.00053	295.01
0.00257	4.50708	0.00042	311.33
0.00203	4.74324	0.00033	327.64
0.00160	4.97940	0.00026	343.95
0.00127	5.21557	0.00021	360.27
0.00100	5.45173	0.00016	376.58

### **totemiss.f**

Double precision Fortran77 subroutine `totemiss(ph2o, pco2, ptot, Tg, L, epsh2o, epsco2, epstot)` calculates the total emissivity of an isothermal gas mixture, using Leckner's model, equations (11.177) through (11.181).

Input:

`ph2o` =  $p_{\text{H}_2\text{O}}$ , partial pressure of water vapor, in bar,  
`pco2` =  $p_{\text{CO}_2}$ , partial pressure of carbon dioxide, in bar,  
`ptot` =  $p$ , total mixture pressure, in bar,  
`Tg` =  $T_g$ , gas column temperature, in K,  
`L` =  $L$ , gas column length, in m,

Output:

`epsh2o` =  $\epsilon_{\text{H}_2\text{O}}$ , total emissivity of water vapor in the mixture,  
`epsco2` =  $\epsilon_{\text{CO}_2}$ , total emissivity of carbon dioxide in the mixture,  
`epstot` =  $\epsilon_{\text{CO}_2+\text{H}_2\text{O}}$ , total emissivity of the mixture, considering overlap effects.

### **totabsor.f**

Double precision Fortran77 subroutine `totabsor(ph2o, pco2, ptot, Tg, Tw, L, absh2o, absco2, abstot)` calculates the total absorptivity of an isothermal gas mixture, using Leckner's model, equations (11.177) through (11.181).

Input:

`ph2o` =  $p_{\text{H}_2\text{O}}$ , partial pressure of water vapor, in bar,  
`pco2` =  $p_{\text{CO}_2}$ , partial pressure of carbon dioxide, in bar,  
`ptot` =  $p$ , total mixture pressure, in bar,  
`Tg` =  $T_g$ , gas column temperature, in K,  
`Tw` =  $T_w$ , wall (or irradiation source) temperature, in K,

$L$  =  $L$ , gas column length, in m,

Output:

$\text{absh}_2\text{o}$  =  $\alpha_{\text{H}_2\text{O}}$ , total absorptivity of water vapor in the mixture,

$\text{absco}_2$  =  $\alpha_{\text{CO}_2}$ , total absorptivity of carbon dioxide in the mixture,

$\text{abstot}$  =  $\alpha_{\text{CO}_2+\text{H}_2\text{O}}$ , total absorptivity of the mixture, considering overlap effects.

Note: `totabsor` calls (i.e., requires) subroutine `totemiss`

### **Leckner.f, Leckner.exe**

Stand-alone frontend for `totemiss(ph2o,pco2,ptot,Tg,L,epsh2o,epsco2,epstot)` and `totabsor(ph2o,pco2,ptot,Tg,Tw,L,absh2o,absco2,abstot)`. User is prompted to input `ph2o`, `pco2`, `ptot`, `Tg`, `Tw` and `L` (see above), and the corresponding total emissivities and absorptivities are printed to the screen.

### **References**

1. Humlíček, J.: "Optimized computation of the Voigt and complex probability functions," *Journal of Quantitative Spectroscopy and Radiative Transfer*, vol. 27, p. 437, 1982.