```
      subroutine van(mobs,npar,iy,im,id,ih,exper,
     2           idim,jdim,gscale,rmax,sdevfg,sdevob,
     3           lunobs,lunfgs,lunana,lunref,
     4           latm,lonm,dlat,dlon,clat,
     5           lfgs,lref,maxiter,alpha)
c     +-----------------------------------------------------------------+
c     + A simple analysis program for VAN                               +
c     +  1. variable declaration                                        +
c     +     1.1  dimension                                              +
c     +     1.2  namelist                                               +
c     +     1.3  grid paramters                                         +
c     +     1.4  observational data fields                              +
c     +     1.5  observations selected for analysis                     +
c     +     1.6  information about rejected data                        +
c     +     1.7  covariance matrix fields                               +
c     +     1.8  administrative parameters                              +
c     +  2. derived parameters                                          +
c     +  3. write input parameters (namelist)                           +
c     +  4. open input and output files                                 +
c     +     4.1 observation file                                        +
c     +     4.2 reference file (OI)                                     +
c     +     4.3 firstguess file                                         +
c     +     4.4 result file                                             +
c     +  5. read and (gross) check observations                         +
c     +  6. analysis loop over all parameters                           +
c     +     6.1 read firstguess                                         +
c     +     6.2 read references                                         +
c     +     6.3 VAN                                                     +
c     +         6.3.1 select influencing observations                   +
c     +         6.3.2 construct (only) covariance matrix                +
c     +         6.3.3 use fgs as ana before at iter=0                   +
c     +         6.3.4 background innovation vector                      +
c     +         6.3.5 initialize Z                                      +
c     +         6.3.6 transform Z to observation locations              +
c     +         6.3.7 calculation innovation vector                     +
c     +         6.3.8 cost function gradient for observation grad(Jo)   +
c     +         6.3.9 cost function gradient for background  grad(Jb)   +
c     +         6.3.10 cost function gradient grad=grad(Jo)+grad(Jb)    +
c     +         6.3.11 minimization step                                +
c     +         6.3.12 transform from Z to X                            +
c     +  7. derived variables                                           +
c     +  8. statistics                                                  +
c     +  9. store analyses and observations                             +
c     + 10. close obs, ana and chk files                               +
c     +     end of the analysis                                         +
c     + A1. write format                                                +
c     +-----------------------------------------------------------------+
c 1   ============================================== variable declaration =
      implicit none
c 1.1 ------------------------------------------------------------ dimension -
      integer mobs,npar,idim,jdim,nhor
c 1.2 ------------------------------------------------------------ namelist -
      integer   iy,im,id,ih
      integer   maxiter(npar)
      real      latm,lonm,dlat,dlon,rmax,alpha(npar)
      real      gscale(npar),sdevfg(npar),sdevob(npar)
      character exper*3
      logical  lfgs,lref
c 1.3 ------------------------------------------------------------ grid -
      integer iana,jana
      real    lats,lonw,dlatfgs,dlonfgs,dlatref,dlonref
      real    refrms
      real    ana(idim,jdim),fgs(idim,jdim),ref(idim,jdim)
      real    p_ana(idim,jdim),t_ana(idim,jdim),u_ana(idim,jdim)
      real    v_ana(idim,jdim),d_ana(idim,jdim)
      real    zana(idim,jdim)
      real    grad_o(idim,jdim),grad_b(idim,jdim),grad(idim,jdim)
      real    zgrad_o(idim,jdim)
c 1.4 ------------------------------------- observational data fields -
      integer   nobs,iiiobs(mobs),nrobs(mobs),flgobs(mobs,npar)
      real      oriobs(mobs,npar),latobs(mobs),lonobs(mobs)
      character filnm*12
c 1.5 ---------------------------- observations selected for analysis -
      integer nana,iiival(mobs),nrval(mobs)
      real    latval(mobs),lonval(mobs),valobs(mobs),valmean
      real    valfgs(mobs),valinn(mobs),valana(mobs)
      real    valinb(mobs)
c 1.6 ------------------------------- information about rejected data -
      integer nskip,lskip(mobs),flag(mobs),dev(mobs)
      logical skip(mobs)
c 1.7 ---------------------------------------- covariance matrix fields -
      real b(idim*jdim*idim*jdim)
c 1.8 ------------------------------------- administrative parameters -
      real    lat,lon,cost,jo,jb
      integer i,j,k,i1,i2,iter
      integer ipar,ios
      integer lunobs,lunfgs,lunana,lunref
      real    clat ! cos(lat)=111111.1111
c 2   ================================================ default values =
      nhor=idim*jdim
c 3   ================================== write some input parameters =
      write(6,1001)
      write(6,'(2a)')' exp : ',exper
      write(6,'(a,4(1x,i2.2))')' date:',iy-1900,im,id,ih
      write(6,1001)
      write(6,*)' parameter              pressure       temperature'
      write(6,1002)
      write(6,*)' sdev of fg          ',sdevfg
      write(6,*)' sdev of obs         ',sdevob
      write(6,*)' Gaussian parameters ',gscale
      write(6,1001)
c 4   ================================== open input and output files =
c 4.1 ------------------------------------------------ observation file -
      write(filnm(1:12),'(4i2.2,a4)') mod(iy,100),im,id,ih,'.obs'
      open(lunobs,file=filnm,iostat=ios,status='old')
c 4.2 ------------------------------------------------ reference  file -
      if(lref) then
         write(filnm(1:12),'(4i2.2,a4)') mod(iy,100),im,id,ih,'.ref'
         open(lunref,file=filnm,iostat=ios,status='old')
         read(lunref,*) iana,jana,lats,lonw,dlatref,dlonref
         if(dlatref.ne.dlat.or.dlonref.ne.dlon) then
            write(6,*) 'from ref, dlon and dlat =',dlonref,dlatref
            write(6,*) 'namelist, dlon and dlat =',dlon,dlat
            stop
         endif
      endif
c 4.3 ------------------------------------------------ firstguess file -
      if(lfgs) then
         write(filnm(1:12),'(4i2.2,a4)') mod(iy,100),im,id,ih,'.fgs'
         open(lunfgs,file=filnm,iostat=ios,status='old')
         read(lunfgs,*) iana,jana,lats,lonw,dlatfgs,dlonfgs
         if(dlatfgs.ne.dlat.or.dlonfgs.ne.dlon) then
            write(6,*) 'from fgs, dlon and dlat =',dlonfgs,dlatfgs
            write(6,*) 'namelist, dlon and dlat =',dlon,dlat
            stop
         endif
      endif
c 4.4 ------------------------------------------------------ result file -
      write(filnm(1:12),'(4i2.2,a,a3)')mod(iy,100),im,id,ih,'.',exper
      open(lunana,file=filnm,iostat=ios,status='unknown')
c 5   ============================ read and (gross) check observations =
      call getobs(lunobs,npar,mobs,nobs,
     +            iiiobs,nrobs,latobs,lonobs,oriobs,flgobs)
c 6   ============================== analysis loop over all parameters =
      do ipar=1,2
c 6.1    ------------------------------------------- read firstguess -
         if(lfgs) call getfld(idim,jdim,fgs,lunfgs)
c 6.2    ------------------------------------------- read references -
         if(lref) call getfld(idim,jdim,ref,lunref)
c 6.3    ------------------------------------------- VAN analysis -
  write(6,*) 'VAN: Variational Analysis with No inversion'
c 6.3.1  ........................... select influencing observations .
         call choose_box(1,idim,jdim,0,0,
     +         iiiobs,latobs,lonobs,oriobs(1,ipar),
     +         nrobs,nobs,latm,lonm,dlat,dlon,rmax,mobs,1,
     +         iiival,nrval,latval,lonval,valobs,nana)
  write(6,*) 'selected obs : ',nana
c 6.3.2  ....................... construct (only)  covariance matrix .
  call bmatrix(idim,jdim,b,latm,lonm,dlat,dlon,clat,
     +            sdevfg(ipar),gscale(ipar),2)
c 6.3.3  ........................... use fgs as ana before at iter=0 .
  write(6,*)' ana=fgs '
  if(.not.lfgs)
     1    call avefgs(idim*jdim,fgs,mobs,npar,oriobs,nrobs,ipar)
  call hcpa2b(idim*jdim,fgs,ana)
         if( lref ) then
               call hrms(idim*jdim,ana,ref,refrms)
        iter=0
chans             write(6,*)'iter = ',iter,' rms = ',refrms
              write(6,*)iter,refrms
         endif
c 6.3.4  ............................. background innovation vector .
         call H(idim,jdim,nana,latval,lonval,
     +            latm,lonm,dlat,dlon,ana,valana)
         do i=1,nana
     valinb(i) = valana(i)-valobs(i)
  enddo
c 6.3.5  ............................................ initialize Z .
  call hseta2c(idim*jdim,zana,0.)
c        ......................... loop for minimization iterations .
         do iter = 1,maxiter(ipar)
c 6.3.6     ........... transform zana to observation locations (Hx) .
            call hab2c(nhor,nhor,1,b,zana,ana)
     call H(idim,jdim,nana,latval,lonval,
     +            latm,lonm,dlat,dlon,ana,valana)
c 6.3.7     ........................ calculate inc innovation vector .
            do i=1,nana
        valinn(i) = valana(i)+valinb(i)
     enddo
c 6.3.8     ........ cost function gradient for observation grad(Jo) .
            call hab2c(1,nana,1,valinn,valinn,jo)
     jo = jo /(2.*sdevob(ipar)*sdevob(ipar))
            do i=1,nana
           valinn(i) = valinn(i) /(sdevob(ipar)*sdevob(ipar))
     enddo
     call hseta2c(idim*jdim,zgrad_o,0.)
     call H_ad(idim,jdim,nana,latval,lonval,
     +            latm,lonm,dlat,dlon,zgrad_o,valinn)
     call hab2c(nhor,nhor,1,b,zgrad_o,grad_o)
c 6.3.9     ......... cost function gradient for background grad(Jb) .
            call hab2c(nhor,nhor,1,b,zana,grad_b)
            call hab2c(1,nhor,1,zana,grad_b,jb)
     jb = jb/2.
c 6.3.10     .......... cost function gradient grad=grad(Jo)+grad(Jb) .
            do i=1,idim
     do j=1,jdim
        grad(i,j)=grad_b(i,j) + grad_o(i,j)
     enddo
     enddo
c 6.3.11    ..................................... minimization step .
```

```
            call stemin(nhor,zana,grad,alpha(ipar))
     if( .not.lref )
     1        write(6,'(a,i5,a,3e12.5)')
     2           'iter = ',iter, '  jb,jo,j= ',jb,jo,jb+jo
c 6.3.12 ............................. transform from zana to ana.
     if( lref .or. iter.eq.maxiter(ipar)) then
        call hab2c(nhor,nhor,1,b,zana,ana)
        do i=1,idim
        do j=1,jdim
           ana(i,j)=fgs(i,j)+ana(i,j)
        enddo
        enddo
        if(ipar.eq.1) then
           call hcpa2b(idim*jdim,ana,p_ana)
        else
           call hcpa2b(idim*jdim,ana,t_ana)
        endif
        if(lref) then
           call hrms(idim*jdim,ana,ref,refrms)
chans           write(6,*)'iter = ',iter,' rms = ',refrms
           write(6,*)iter,refrms
        endif
     endif
         enddo ! end of iteration loop
      enddo    ! end of parameter loop
c 7   ================================================ derived variables =
      call derivf(t_ana,p_ana,
     +            u_ana,v_ana,d_ana,idim,jdim,latm,clat,dlat)
c 8   ========================================================= statistics =
      write(6,*)' statistics :'
      call statis('   p min/max = ',idim,jdim,p_ana)
      call statis('   t min/max = ',idim,jdim,t_ana)
      call statis('   u min/max = ',idim,jdim,u_ana)
      call statis('   v min/max = ',idim,jdim,v_ana)
      call statis(' dyn min/max = ',idim,jdim,d_ana)
      write(6,1001)
c 9   ========================== store the analyses and observations =
      write(lunana,'(2i4,4f10.4)')idim,jdim,
     +             latm-jdim/2*dlat,lonm-idim/2*dlon,dlat,dlon
      call putfld('p',lunana,idim,jdim,p_ana,idim,jdim)
      call putfld('t',lunana,idim,jdim,t_ana,idim,jdim)
cc    call putfld('u',lunana,idim,jdim,u_ana,idim,jdim)
cc    call putfld('v',lunana,idim,jdim,v_ana,idim,jdim)
      call putobs('obs',mobs,npar,nobs,lunana,
     x  iiiobs,latobs,lonobs,oriobs,flgobs)
c 10  ================================== close obs, ana and chk files =
      close(lunobs)
      close(lunfgs)
      close(lunana)
      close(lunref)
c     ========================================== end of the analysis =
      write(6,*)'analysis finished without any error!'
      write(6,1001)
      return
c A1  ======================================================== formats =
1001  format("============================================================")
1002  format("------------------------------------------------------------")
6211  format(1x,a,20(/,2x,10i7))
6212  format(1x,a,20(/,2x,10f7.1))
      end
```