

```

      subroutine psas(mobs,npar,iy,im,id,ih,exper,
2         idim,jdim,gscale,rmax,sdevfg,sdevob,
3         lunobs,lunfsg,lunana,lunref,
4         latm,lonm,dlat,dlon,clat,
5         lfgs,lref,maxiter,alpha)
c
c +-----+
c + A simple analysis program for VAN
c + 1. variable declaration
c + 1.1 dimension
c + 1.2 namelist
c + 1.3 grid parameters
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 VAN
c + 6.2.1 select influencing observations
c + 6.2.2 construct (only) covariance matrix
c + 6.2.3 use fgs as ana before at iter=0
c + 6.2.4 background innovation vector
c + 6.2.5 initialize W
c + --- loop for minimization iterations -----
c + 6.2.6 cost function gradient
c + 6.2.7 cost function
c + 6.2.8 minimization step
c + ---- end of loop for iteration -----
c + 6.2.9 transform from W to X
c + 7. store observations
c + 8. close obs and ana files
c + end of the analysis
c + A1. write format
c +-----+
c 1 ===== variable declaration =====
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,dlatfsg,dlonfsg
      real ana(idim,jdim),fsg(idim,jdim)
c 1.4 ----- observational data fields -----
      integer nobs,iiobs(mobs),nrobs(mobs),flgobs(mobs,npar)
      real oriobs(mobs,npar),latobs(mobs),lonobs(mobs)
      character film*12
c 1.5 ----- observations selected for analysis -----
      integer nana,iiival(mobs),nrval(mobs)
      real latval(mobs),lonval(mobs),valobs(mobs),valmean
      real valfsg(mobs),valinn(mobs),valana(mobs)
      real wana(mobs*mobs),wgrad(mobs*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 co(mobs*mobs),bh(idim*jdim*mobs)
c 1.8 ----- administrative parameters -----
      real cost,jb,jol,jol2,jol3
      integer i,j,iter
      integer ipar,ios
      integer lunobs,lunfsg,lunana,lunref
      real clat ! cos(lat)=111111.1111
c 2 ===== default values =====
c 3 ===== write some input parameters =====
      write(6,1001)
      write(6,'(2a)') exp : ',exper
      write(6,'(a4(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 4.3 ----- firstguess file -----
      if(lfgs) then
        write(filnm(1:12),'(4i2.2,a4)') mod(iy,100),im,id,ih,'.fsg'
        open(lunfsg,file=filnm,iostat=ios,status='old')
        read(lunfsg,*) iana,jana,lats,lonw,dlatfsg,dlonfsg
      end if
      if(dlatfsg.ne.dlat.or.dlonfsg.ne.dlon) then
        write(6,*) 'from fgs, dlon and dlat =',dlonfsg,dlatfsg
        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')
      write(lunana,'(2i4,4f10.4)') idim,jdim,
      + latm-jdim/2*dlat,lonm-idim/2*dlon,dlat,dlon
c 5 ===== read and (gross) check observations =====
      call getobs(lunobs,npar,mobs,nobs,
      + iiobs,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,fsg,lunfsg)
c 6.2 ----- OI analysis -----
        write(6,*) 'PSAS: Physical-space Statistical Analysis System'
c 6.4.1 ----- select influencing observations -----
        call choose_box(1,idim,jdim,0,0,
        + iiobs,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 6.4.2 ----- construct covariance matrices -----
        call covmat(latm,clat,sdevfg(ipar),sdevob(ipar),
        + gscale(ipar),nana,latval,lonval,co)
        call bhm(idim,jdim,nana,bh,latval,lonval,
        + latm,lonm,dlat,dlon,clat,sdevfg(ipar),gscale(ipar),5)
c 6.4.3 ----- use fgs as ana before at iter=0 -----
        write(6,*) ' ana=fgs '
        if(.not.lfgs)
          1 call avefsg(idim*jdim,fsg,mobs,npar,oriobs,nrobs,ipar)
          call hcpa2b(idim*jdim,fsg,ana)
c 6.4.4 ----- background innovation vector -----
          call H(idim,jdim,nana,latval,lonval,
          + latm,lonm,dlat,dlon,ana,valana)
c 6.4.5 ----- initialize W -----
          call hseta2c(nana,wana,0.)
c ----- loop for minimization iterations -----
          do iter = 1,maxiter(ipar)
c 6.4.6 ----- cost function gradient -----
            call hab2c(1,nana,1,wana,valana,jol1)
            call hab2c(1,nana,1,wana,valobs,jol2)
            jol = jol1-jol2
            call hab2c(nana,nana,1,co,wana,wgrad)
            call hab2c(1,nana,1,wana,wgrad,jb)
            jb = jb/2.
            do i=1,nana
              wgrad(i)=wgrad(i) + valana(i)-valobs(i)
            enddo
c 6.4.7 ----- cost function -----
            cost = Jol + jb
c 6.4.8 ----- minimization step -----
            call stemin(nana,wana,wgrad,alpha(ipar))
            write(6,'(a,i5,a,3e12.5)')
            + ' iter = ',iter, ' w cost= ',jb,jol,cost
          enddo
c 6.4.9 ----- transform from wana to ana -----
          call hab2c(nhor,nana,1,bh,wana,ana)
          do i=1,idim
            do j=1,jdim
              ana(i,j)=fsg(i,j)+ana(i,j)
            enddo
          enddo
          if(ipar.eq.1) then
            call statis(' p min/max = ',idim,jdim,ana)
            call putfld('p',lunana,idim,jdim,ana,idim,jdim)
          else
            call statis(' t min/max = ',idim,jdim,ana)
            call putfld('t',lunana,idim,jdim,ana,idim,jdim)
          endif
        enddo
c 7 ===== store observations =====
        call putobs('obs',mobs,npar,nobs,lunana,
        + iiobs,latobs,lonobs,oriobs,flgobs)
c 8 ===== close obs, ana and chk files =====
        close(lunobs)
        close(lunfsg)
        close(lunana)
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

```