```
*********************************************************
*                                                       *
*   Here we DO NOT take care of possible roundoff       *
*   errors in nummerical diagonalization                *
*   procedures.                                         *
*                                                       *
*   In calculating sqe for TRUE min. we have used       *
*   mass matrices coming out from the most general      *
*   calculation scheme (all the fields participate)     *
*   by setting the extra fields equal to zero.          *
*   Numerical methods for diagonalization are used      *
*   everywhere.                                         *
*                                                       *
*   sf  :  Soft parameters at some scale.               *
*   sqe :  Array containing the SQUERES of mass         *
*          eigenvalues.                                 *
*                                                       *
*       Light yukawas do not run                        *
*                                                       *
*********************************************************
      SUBROUTINE fdmass(sf,x1,x2,sqe)
      IMPLICIT DOUBLE PRECISION (a-h,o-z)
      DOUBLE PRECISION sf(NUU),sqe(48)
      DOUBLE PRECISION dc2(4),dm2(4),sh1(4),sh2(4)
      COMMON /NUEQ/ NUU
CC
CC
      do  ii=1,48
         sqe(ii)=0.d0  !  Useful for partial spectrum
      enddo
CC
      call assvl(sf,g,g2,g3,Yu3,Yd3,Yl3,rM1,rM2,rM3,rmu,
     # sqm3,Au3,Ad3,Al3,sqmq3,sqmq2,sqmq1,sqmuc3,sqmuc2,sqmuc1,
     # sqmdc3,sqmdc2,sqmdc1,sqml3,sqml2,sqml1,sqmec3,sqmec2,
     # sqmec1,sqm1,sqm2,au1,au2,ad1,ad2,al1,al2,yu1,yu2,yd1,
     # yd2,yl1,yl2)

      gp=g2*g2
      sco=g*g + g2*g2

      call dvferm(yl1,x1,sqe(1))
      call dvferm(yl2,x1,sqe(2))
      call dvferm(yl3,x1,sqe(3))

      call dvferm(yu1,x2,sqe(4))
      call dvferm(yu2,x2,sqe(5))
      call dvferm(yu3,x2,sqe(6))

      call dvferm(yd1,x1,sqe(7))
      call dvferm(yd2,x1,sqe(8))
      call dvferm(yd3,x1,sqe(9))

      call dvgabo(gp,x1,x2,sqe(10))
      call dvgabo(sco,x1,x2,sqe(11))

      call dvchino(rmu,rm2,g2,x1,x2,dc2)
      sqe(12)=dc2(1)
      sqe(13)=dc2(2)
      sqe(14)=dc2(3)
      sqe(15)=dc2(4)

      call dvneutr(g,g2,rmu,rm1,rm2,x1,x2,dm2)
```



```
      sqe(16)=dm2(1)
      sqe(17)=dm2(2)
      sqe(18)=dm2(3)
      sqe(19)=dm2(4)

      call dvsnet(sqml1,g,g2,x1,x2,sqe(20))
      call dvsnet(sqml2,g,g2,x1,x2,sqe(21))
      call dvsnet(sqml3,g,g2,x1,x2,sqe(22))

      call dvslept(sqml1,sqmec1,yl1,al1,rmu,g,g2,x1,x2,sqe(23),sqe(24))
      call dvslept(sqml2,sqmec2,yl2,al2,rmu,g,g2,x1,x2,sqe(25),sqe(26))
      call dvslept(sqml3,sqmec3,yl3,al3,rmu,g,g2,x1,x2,sqe(27),sqe(28))

      call dvsqu(sqmq1,sqmuc1,yu1,au1,rmu,g,g2,x1,x2,sqe(29),sqe(30))
      call dvsqu(sqmq2,sqmuc2,yu2,au2,rmu,g,g2,x1,x2,sqe(31),sqe(32))
      call dvsqu(sqmq3,sqmuc3,yu3,au3,rmu,g,g2,x1,x2,sqe(33),sqe(34))

      call dvsqd(sqmq1,sqmdc1,yd1,ad1,rmu,g,g2,x1,x2,sqe(35),sqe(36))
      call dvsqd(sqmq2,sqmdc2,yd2,ad2,rmu,g,g2,x1,x2,sqe(37),sqe(38))
      call dvsqd(sqmq3,sqmdc3,yd3,ad3,rmu,g,g2,x1,x2,sqe(39),sqe(40))

      call dvhig(sqm1,sqm2,sqm3,g,g2,x1,x2,sh1,sh2)
      sqe(41)=sh1(1)
      sqe(42)=sh1(2)
      sqe(43)=sh1(3)
      sqe(44)=sh1(4)
      sqe(45)=sh2(1)
      sqe(46)=sh2(2)
      sqe(47)=sh2(3)
      sqe(48)=sh2(4)

CC
       return
       END
```

```
*==> dvchino.f <==
*********************************************
*                                           *
*   Field dependent masses for charginos    *
*                                           *
*   chs(4,4)     Mass matrix                *
*   dc2(4)       Array of squere eigenvalues*
*                                           *
*********************************************
      SUBROUTINE dvchino(rmu,rm2,g2,x1,x2,dc2)
      IMPLICIT DOUBLE PRECISION (a-h,o-z)
      PARAMETER(n=4,np=4)
      DOUBLE PRECISION chs(4,4),dc(4),e(4),z(4,4),dc2(4)
C
      do  m1=1,np
         do  m2=1,np
            chs(m1,m2)=0.d0
         enddo
      enddo
      chs(1,3)=rm2
      chs(1,4)=-g2*x1
      chs(2,3)=-g2*x2
      chs(2,4)=rmu
      do  n1=2,np
         do  n2=1,n1-1
            chs(n1,n2)=chs(n2,n1)   !  Symmetric matrix
         enddo
      enddo
      call tred2(chs,n,np,dc,e)
      call tqli(dc,e,n,np,z)
C-------------------------------------------------
C   Array dc contains the masses. We need squeres.
C-------------------------------------------------
      do  k=1,np
         dc2(k)=dc(k)*dc(k)
      enddo
C
      return
      END

*==> dvferm.f <==
*********************************************
*                                           *
*   Field dependent mass for fermions       *
*                                           *
*   y    :  Yukawa                          *
*   x    :  Vev                             *
*   sqe  :  Squere mass eigenvalue          *
*                                           *
*   y    |x                                 *
*   ---- |-----------------                 *
*   yl1  |x1                                *
*   yl2  |x1    Leptons                     *
*   yl3  |x1                                *
*   ---- |-----------------                 *
*   yu1  |x2                                *
*   yu2  |x2    UP Quarks                   *
*   yu3  |x2                                *
*   ---- |-----------------                 *
*   yd1  |x1                                *
*   yd2  |x1   DOWN Quarks                  *
*   yd3  |x1                                *
```



```
*                                           *
*********************************************
      SUBROUTINE dvferm(y,x,sqe)
      IMPLICIT DOUBLE PRECISION (a-h,o-z)
C
      sqe=(y*x)**2
C
      return
      END

*==> dvgabo.f <==
*********************************************
*                                           *
*   Field dependent mass for gauge bosons   *
*                                           *
*            g2*g2           W              *
*          /                                *
*    cf --                                  *
*          \                                *
*            g*g + g2*g2     Z              *
*                                           *
*    x1,x2   Independent variables          *
*    sqe     Squere eigenvalue              *
*                                           *
*********************************************
      SUBROUTINE dvgabo(cf,x1,x2,sqe)
      IMPLICIT DOUBLE PRECISION (a-h,o-z)
C
      sqe=cf*(x1**2+x2**2)/2.d0
C
      return
      END

*==> dvhig.f <==
*********************************************
*                                           *
*   Field dependent masses for higgs sector *
*                                           *
*   h1(4,4)     Mass matrix                 *
*   h2(4,4)     Mass matrix                 *
*                                           *
*   sh1(4)      Array of squere eigenvalues *
*   sh2(4)      Array of squere eigenvalues *
*                                           *
*   sm1 -> sqm1                             *
*   sm2 -> sqm2                             *
*   sm3 -> sqm3                             *
*                                           *
*********************************************
      SUBROUTINE dvhig(sm1,sm2,sm3,g,g2,x1,x2,sh1,sh2)
      IMPLICIT DOUBLE PRECISION (a-h,o-z)
      PARAMETER(n=4,np=4)
      DOUBLE PRECISION hm1(4,4),sh1(4),e1(4),z1(4,4)
      DOUBLE PRECISION hm2(4,4),sh2(4),e2(4),z2(4,4)
C
      sco=g*g + g2*g2
C
      chi=sm1 + sco*dfx( dsqrt(2.d0)*x1 , x2)/4.d0
      psi=sm1 + sco*dfx(x1,x2)/4.d0 + g2*g2*x2*x2/2.d0
      zed=-sco*x1*x2/4.d0
C
      wmeg=sco*x1*x1/4.d0
```

```
      alph=sm3 - sco*x1*x2/4.d0
      bet=sm2 + sco*dfx(x2,x1)/4.d0 + g2*g2*x1*x1/2.d0
C
      gama=sm2 + sco*dfx( dsqrt(2.d0)*x2 , x1)/4.d0
      delt=sco*x2*x2/4.d0
      th=-sm3 + g2*g2*x1*x2/2.d0
C
      do  j=1,4
         do k=1,4
           hm1(j,k)=0.d0
         enddo
      enddo
      hm1(1,1)=psi
      hm1(1,4)=th
      hm1(2,2)=bet
      hm1(2,3)=th
      hm1(3,2)=th
      hm1(3,3)=psi
      hm1(4,1)=th
      hm1(4,4)=bet
C
C
      hm2(1,1)=chi
      hm2(1,2)=zed
      hm2(1,3)=wmeg
      hm2(1,4)=alph
      hm2(2,2)=gama
      hm2(2,3)=alph
      hm2(2,4)=delt
      hm2(3,3)=chi
      hm2(3,4)=zed
      hm2(4,4)=gama
      do  m1=2,4
         do  m2=1,m1-1
            hm2(m1,m2)=hm2(m2,m1)  !  Symmetric matrix
         enddo
      enddo
      call tred2(hm1,n,np,sh1,e1)
      call tqli(sh1,e1,n,np,z1)
C
      call tred2(hm2,n,np,sh2,e2)
      call tqli(sh2,e2,n,np,z2)
C
      return
      END

*==> dvneutr.f <==
*********************************************
*                                           *
*   Field dependent masses for neutralinos  *
*                                           *
*   tn(4,4)     Mass matrix                 *
*   dm2(4)       Array of squere eigenvalues *
*                                           *
*********************************************
      SUBROUTINE dvneutr(g,g2,rmu,rm1,rm2,x1,x2,dm2)
      IMPLICIT DOUBLE PRECISION (a-h,o-z)
      PARAMETER (n=4,np=4)
      DOUBLE PRECISION tn(4,4),dm(4),e(4),z(4,4),dm2(4)
C
      tn(1,1)=rm1
      tn(1,2)=0.d0
```



```
      tn(1,3)=g*x1/dsqrt(2.d0)
      tn(1,4)=-g*x2/dsqrt(2.d0)
      tn(2,2)=rm2
      tn(2,3)=-g2*x1/dsqrt(2.d0)
      tn(2,4)=g2*x2/dsqrt(2.d0)
      tn(3,3)=0.d0
      tn(3,4)=-rmu
      tn(4,4)=0.d0
        do  i1=2,4
           do   i2=1,i1-1
              tn(i1,i2)=tn(i2,i1)  ! Symmetric matrix
           enddo
        enddo
        call tred2(tn,n,np,dm,e)
        call tqli(dm,e,n,np,z)
C-------------------------------------------------
C   Array dc contains the masses. We need squeres.
C-------------------------------------------------
      do  k=1,4
         dm2(k)=dm(k)*dm(k)
      enddo
C
      return
      END

*==> dvslept.f <==
*********************************************
*                                           *
*   Field dependent masses for sleptons     *
*                                           *
*   y           Yukawa                      *
*   a           Trilinear                   *
*   slt(2,2)    Mass matrix                 *
*   sq1         Eigenvalue                  *
*   sq2         Eigenvalue                  *
*                                           *
*                                           *
*   sm2    |  smc2    |  y    |  a          *
*   ------|---------|------|------          *
*   sqml1  | sqmec1   | yl1   | al1          *
*   sqml2  | sqmec2   | yl2   | al2          *
*   sqml3  | sqmec3   | yl3   | al3          *
*                                           *
*********************************************
      SUBROUTINE dvslept(sm2,smc2,y,a,rmu,g,g2,x1,x2,sq1,sq2)
      IMPLICIT DOUBLE PRECISION (a-h,o-z)
      PARAMETER(n=2,np=2)
      DOUBLE PRECISION slt(2,2),dl2(2),e(2),z(2,2)
C
      slt(1,1)=smc2 + (y*x1)**2 + g*g*dfx(x2,x1)/2.d0
      slt(1,2)=y*(a*x1 + rmu*x2)
      slt(2,1)=y*(a*x1 + rmu*x2)
      slt(2,2)=sm2 + (y*x1)**2 + dfx(g,g2)*dfx(x1,x2)/4.d0
C
      call tred2(slt,n,np,dl2,e)
      call tqli(dl2,e,n,np,z)
C
      sq1=dl2(1)
      sq2=dl2(2)
C
      return
      END
```

```
*==> dvsnet.f <==
*********************************************
*                                           *
*   Field dependent masses for sneutrinos   *
*                                           *
*   sfm2      Soft mass                     *
*   sqe       Mass squered                  *
*                                           *
*   sfm2 : {sqml1, sqml2, sqml3}            *
*********************************************
      SUBROUTINE dvsnet(sfm2,g,g2,x1,x2,sqe)
      IMPLICIT DOUBLE PRECISION (a-h,o-z)
C
      sco=g*g + g2*g2
      sqe=sfm2 + sco*dfx(x1,x2)/4.d0
C
      return
      END

*==> dvsqd.f <==
*********************************************
*                                           *
*   Field dependent masses for DOWN quarks  *
*                                           *
*   y            Yukawa                     *
*   a            Trilinear                  *
*   sq(2,2)      Mass matrix                *
*   dsq1         Eigenvalue                 *
*   dsq2         Eigenvalue                 *
*                                           *
*                                           *
*   sm2    |  smc2    |  y     |  a         *
*   ------|----------|------|------          *
*   sqmq1  | sqmdc1   | yd1    | ad1        *
*   sqmq2  | sqmdc2   | yd2    | ad2        *
*   sqmq3  | sqmdc3   | yd3    | ad3        *
*                                           *
*********************************************
      SUBROUTINE dvsqd(sm2,smc2,y,a,rmu,g,g2,x1,x2,dsq1,dsq2)
      IMPLICIT DOUBLE PRECISION (a-h,o-z)
      PARAMETER(n=2,np=2)
      DOUBLE PRECISION sq(2,2),dq2(2),e(2),z(2,2)
C
      gsum=g*g + 3.d0*g2*g2
C
      sq(1,1)=smc2 + (y*x1)**2 + g*g*dfx(x2,x1)/6.d0
      sq(1,2)=y*(a*x1 + rmu*x2)
      sq(2,1)=y*(a*x1 + rmu*x2)
      sq(2,2)=sm2 + (y*x1)**2 + gsum*dfx(x2,x1)/12.d0
C
C
      call tred2(sq,n,np,dq2,e)
      call tqli(dq2,e,n,np,z)
C
      dsq1=dq2(1)
      dsq2=dq2(2)
C
      return
      END

*==> dvsqu.f <==
```



```
*********************************************
*                                           *
*   Field dependent masses for UP quarks    *
*                                           *
*   y            Yukawa                     *
*   a            Trilinear                  *
*   sq(2,2)      Mass matrix                *
*   usq1         Eigenvalue                 *
*   usq2         Eigenvalue                 *
*                                           *
*                                           *
*   sm2    |  smc2    |  y     |  a         *
*   ------|----------|------|------          *
*   sqmq1  | sqmuc1   | yu1    | au1        *
*   sqmq2  | sqmuc2   | yu2    | au2        *
*   sqmq3  | sqmuc3   | yu3    | au3        *
*                                           *
*********************************************
      SUBROUTINE dvsqu(sm2,smc2,y,a,rmu,g,g2,x1,x2,usq1,usq2)
      IMPLICIT DOUBLE PRECISION (a-h,o-z)
      PARAMETER(n=2,np=2)
      DOUBLE PRECISION sq(2,2),dq2(2),e(2),z(2,2)
C
      sq(1,1)=smc2 + (y*x2)**2 - g*g*dfx(x2,x1)/3.d0
      sq(1,2)=-y*(rmu*x1 + a*x2)
      sq(2,1)=-y*(rmu*x1 + a*x2)
      tal=dfx( g , dsqrt(3.d0)*g2 )*dfx(x2,x1)/12.d0
      sq(2,2)=sm2 + (y*x2)**2 + tal
C
C
      call tred2(sq,n,np,dq2,e)
      call tqli(dq2,e,n,np,z)
C
      usq1=dq2(1)
      usq2=dq2(2)
C
      return
      END
```