```
      subroutine gaussfg_init(natom,nbonds,ib,jb,nquant,labels,nlink)

!     callするとき:
!       gaussfg_init(natom,mbona,ix(i18),ix(i20),nquant,labels,nlink)

!     natom           : 全原子数 ( リンク原子含まず )
!     nbonds          : 水素を含まないbondの数
!     ib(*)           : (bondする原子iのインデックス-1)x3
!     jb(*)           : (bondする原子jのインデックス-1)x3
!     nquant          : QM原子数 ( リンク原子含まず )
!     labels(*)       : QM原子のインデックスリスト
!     nlink           : リンク原子の数
!
!     nqm_gfg         - QM原子数
!     nln_gfg         - リンク原子数
!     nqmln_gfg       - QM原子数+リンク原子数
!     iqmln_gfg(*)    - リンクしているQM原子インデックス
!     immln_gfg(*)    - リンクしているMM原子インデックス
!
!     gh_route_gfg    - Gaussian route section
!     gh_molsp_gfg    - Gaussian molecule specification section

      use iflport ! Intel Fortran 7.0
!      use ifport ! Intel Fortran 9.0

      implicit none
      integer natom,nbonds,ib(*),jb(*),nquant,labels(*),nlink

      integer natom_gfg, nqm_gfg, nln_gfg, nqmln_gfg,
     &        iqmln_gfg(1000), immln_gfg(1000)
      logical debug_gfg, charg_gfg, keeprwf_gfg                         ! YK ADDED
      integer nsstart_gfg, nskeep_gfg, nsdel_gfg, igcharg_gfg           ! YK ADDED
      character gh_route_gfg*800, sep_gfg*2                             ! YK A&CHANGED

      common /gfg/ natom_gfg, nqm_gfg, nln_gfg, nqmln_gfg,
     &             iqmln_gfg, immln_gfg,
     &             debug_gfg, charg_gfg, keeprwf_gfg,                   ! YK ADDED
     &             nsstart_gfg, nskeep_gfg, nsdel_gfg, igcharg_gfg,     ! YK ADDED
     &             gh_route_gfg, sep_gfg                                ! YK ADDED

      integer i, j, ii, jj, mm
      integer io1, ierr
      parameter (io1=70)
      logical err, islink0, isroute, isname, isiop                      ! YK ADDED
      character line*130, line2*130                                     ! YK ADDED
      integer ipstt, ipend, lenline, ipstt2, ipend2, rdline             ! YK ADDED

      ! gaussian.headerがないときはリターン
      inquire(file='gaussian.header', exist=err)                        ! YK CHANGED
      if(.not. err) return                                              ! YK CHANGED

      ! 定数
      natom_gfg = natom
      nqm_gfg = nquant
      nln_gfg = nlink
      nqmln_gfg = nqm_gfg+nln_gfg

      ! Gaussianヘッダーの読み込み
C YKYKYK 1                                                              ! YK DELETED
C      open(io1,file='gaussian.header',status='old',action='read')
C      read(io1,'(a)') gh_route_gfg ! forceやchargeが入っていてはいけない
C      read(io1,'(a)') gh_molsp_gfg
C      close(io1)
C YKYKYK 1                                                              ! YK DELETED
C YKYKYK 1                                                              ! YK CHANGED
```

```
66          !  We want to add link 0 commands such as %mem and to change the place or the name
67          ! of Read-Write file.
68          !**** sample 'gaussian.header' file ****
69          ! %nprocshared=2
70          ! %mem=512MB
71          !  int=/work/yamada.int
72          ! %rwf=/work/yamada.rwf
73          !   rhf/3-21+G** scf=tight
74          ! # rhf/6-31G(d,p) scf=tight
75          !
76          ! QM: Ammonia with 2 Waters and MM : 283 TIP3P QM/MM-MD calculation
77          !
78          ! 0 1
79          !*************************************
80          write(gh_route_gfg, '(a)') repeat(' ', 800) ! INITIALIZATION
81          write(sep_gfg, '(a)') "$&"                  ! SEPARATION CHARACTER
82          open(io1,file='gaussian.header',status='old',action='read')
83          islink0 = .true.
84          isroute = .false.
85          isname  = .false.
86          isiop   = .false.
87          ipend   = 0
88          do
89            read(io1, '(a)', iostat = ierr) line
90            if(ierr /= 0) exit
91            lenline = len_trim(line)
92            if(islink0) then
93              if(index(line, "#") /= 0) then
94                islink0 = .false.
95                isroute = .true.
96
97                ! Do not permit double "iop"
98                ipstt2 = index(line, 'iop')
99                if(ipstt2 /= 0) then
100               ipend2 = index(line(ipstt2: lenline), ')') + ipstt2 - 1
101               ipstt2 = ipend2 - 1
102               write(line2, '(a)')
103     &           line(1: ipstt2)//",6/13=1"//line(ipend2: lenline)
104               lenline = len_trim(line2)
105               write(line, '(a)') trim(line2)
106               isiop = .true.
107             endif
108             print *, trim(line), isiop
109             ! Delete force option if it is in route
110             ipstt2 = index(line, 'force')
111             if(ipstt2 /= 0) then
112               ipend2 = index(line(ipstt2: lenline), 'e') + ipstt2 - 1
113               ipstt2 = ipend2 - 6
114               write(line2, '(a)')
115     &           line(1: ipstt2)//line(ipend2: lenline)
116               lenline = len_trim(line2)
117               write(line, '(a)') trim(line2)
118             endif
119             ! Delete force option if it is in route
120             ipstt2 = index(line, 'charge')
121             if(ipstt2 /= 0) then
122               ipend2 = index(line(ipstt2: lenline), 'e') + ipstt2 - 1
123               ipstt2 = ipend2 - 7
124               write(line2, '(a)')
125     &           line(1: ipstt2)//line(ipend2: lenline)
126               lenline = len_trim(line2)
127               write(line, '(a)') trim(line2)
128             endif
129           elseif(index(line, "%") == 0) then
130             cycle
```

```
131             endif
132           elseif(isroute) then
133             if(lenline == 0) then
134               isroute = .false.
135               isname  = .true.
136               if(.not.isiop) then
137                 write(line2, '(a)') "iop(6/13=1) force"//sep_gfg
138               else
139                 write(line2, '(a)') "force"//sep_gfg
140               endif
141               lenline = len_trim(line2)
142               write(line, '(a)') trim(line2)
143             else
144               ! Do not permit double "iop"
145               ipstt2 = index(line, 'iop')
146               print *, "K", ipstt2
147               if(ipstt2 /= 0) then
148                 ipend2 = index(line(ipstt2: lenline), ')') + ipstt2 - 1
149                 ipstt2 = ipend2 - 1
150                 write(line2, '(a)')
151       &           line(1: ipstt2)//",6/13=1"//line(ipend2: lenline)
152                 lenline = len_trim(line2)
153                 write(line, '(a)') trim(line2)
154                 isiop = .true.
155               endif
156               !D print *, trim(line), isiop
157               ! Delete force option if it is in route
158               ipstt2 = index(line, 'force')
159               if(ipstt2 /= 0) then
160                 ipend2 = index(line(ipstt2: lenline), 'e') + ipstt2 - 1
161                 ipstt2 = ipend2 - 6
162                 write(line2, '(a)')
163       &           line(1: ipstt2)//line(ipend2: lenline)
164                 lenline = len_trim(line2)
165                 write(line, '(a)') trim(line2)
166               endif
167               ! Delete force option if it is in route
168               ipstt2 = index(line, 'charge')
169               if(ipstt2 /= 0) then
170                 ipend2 = index(line(ipstt2: lenline), 'e') + ipstt2 - 1
171                 ipstt2 = ipend2 - 7
172                 write(line2, '(a)')
173       &           line(1: ipstt2)//line(ipend2: lenline)
174                 lenline = len_trim(line2)
175                 write(line, '(a)') trim(line2)
176               endif
177             endif
178           elseif(isname) then
179             if(lenline == 0) isname = .false.
180           endif
181           ipstt   = ipend + 1
182           ipend   = ipend + lenline + 2   ! for separater following 'line'
183           if(ipend > 800) call abort('CHANGE LENGTH of gh_route_gfg')
184           write(gh_route_gfg(ipstt: ipend), '(a)') trim(line)//sep_gfg
185         enddo
186         close(io1)
187     C YKYKYK 1                                                          ! YK CHANGED
188
189     C YKYKYK 2
190           !**** sample 'gaussian.setting' file **** (This file maybe merge into KEYWORD file)
191           ! debug_gfg   : keep gaussian-debug.log or not
192           ! charg_gfg   : keep gaussian-charg.log or not
193           !               * You can also check specified structure is reasonable or not by this file *
194           ! keeprwf_gfg : keep Read-Write file or not
195           !
```

```
196         ! nsstart_gfg : Step Number to start sampling for charge analysis (ID/PD charge)
197         ! nskeep_gfg  : The interval Number to keep Gaussian input data in gaussian-charg.log /
198         !               Read Write file / PD charge in gd-charge.wpp
199         ! nsdel_gfg   : The interval Number to delete them
200         ! igcharg_gfg : index of charge derived from gaussian
201         !               0/1/2/3/4 means None/Mulliken/PD Charge(ESP)/AIM/NPA/
202         !
203         !   TIME/STEP
204         ! 0 ---------------||------|------------|------|-------------|------|-------------|--->
205         !            nsstart_gfg
206         !               ---->      <----- nskeep_gfg
207         !                     ----->            <------ nsdel_gfg
208         inquire(file='gaussian.setting', exist=err)
209         ! INITIALIZATION (According to default setting, any files are not kept)
210         debug_gfg   = .false.
211         charg_gfg   = .false.
212         keeprwf_gfg = .false.
213         nsstart_gfg = 999999
214         nskeep_gfg  = 1
215         nsdel_gfg   = 0
216         igcharg_gfg = 0
217         if(err) then ! GAUSSIAN.SETTING file exists
218           open(io1,file='gaussian.setting',status='old', action='read')
219           read(io1, '(a)') line
220           debug_gfg   = (index(line,"DEBUG") + index(line,"debug") /= 0)
221           charg_gfg   = (index(line,"CHGANL") + index(line,"chganl") /= 0)
222           keeprwf_gfg =
223        &           (index(line, "KEEPRWF") + index(line, "keeprwf") /= 0)
224           ipstt = index(line,"NSSTART") + index(line,"nsstart")
225           if(ipstt /= 0) nsstart_gfg = rdline(ipstt, line)
226           ipstt = index(line,"NSKEEP") + index(line,"nskeep")
227           if(ipstt /= 0) nskeep_gfg  = rdline(ipstt, line)
228           ipstt = index(line,"NSDELETE") + index(line,"nsdelete")
229           if(ipstt /= 0) nsdel_gfg   = rdline(ipstt, line)
230           ipstt = index(line,"GQMCHG") + index(line,"gqmchg")
231           if(ipstt /= 0) igcharg_gfg = rdline(ipstt, line)
232           close(io1, status = 'keep')
233           if((igcharg_gfg > 1).and.(index(gh_route_gfg,'pop') == 0)) then
234             write(6, '(/5x, 2a/)')
235        &      "**** Warning! : Pop option is not in route section. ",
236        &      "So GQMCHG is set to 0 ****"
237             igcharg_gfg = 0
238           endif
239           if((igcharg_gfg>1).and.(nsstart_gfg==999999)) nsstart_gfg = 0
240   CD        print '(a, 3l2, 4i5)', "SETTING",  debug_gfg, charg_gfg,
241   CD     &    keeprwf_gfg, nsstart_gfg, nskeep_gfg, nsdel_gfg, igcharg_gfg
242         endif
243
244         if(debug_gfg) ierr = runqq('rm', '-f gaussian-debug.log')   ! log
245         if(charg_gfg) ierr = runqq('rm', '-f gaussian-charg.log')   ! log
246         if(igcharg_gfg /= 0) ierr = runqq('rm', '-f gd-charge.wpp') ! log
247
248         write(6, '(a, 3l2, 4i5/)') "G SETTING: ",  debug_gfg, charg_gfg,
249        &    keeprwf_gfg, nsstart_gfg, nskeep_gfg, nsdel_gfg, igcharg_gfg
250   C YKYKYK 2
251
252
253         ! リンク原子に結合しているQM原子,MM原子の取得
254         mm = 0
255         do i=1, nbonds
256           ii = 0
257           jj = 0
258           do j=1, nquant
259             if((ib(i)/3+1)==labels(j)) ii=1
260             if((jb(i)/3+1)==labels(j)) jj=1
```

```
261           enddo
262           if((ii+jj)==1) then
263             if(ii==1) then
264               iqmln_gfg(mm+1) = ib(i)/3+1
265               immln_gfg(mm+1) = jb(i)/3+1
266             else
267               iqmln_gfg(mm+1) = jb(i)/3+1
268               immln_gfg(mm+1) = ib(i)/3+1
269             endif
270             mm = mm + 1
271           endif
272         enddo
273
274         end
275
276   ! ----------------------------------------------------------------------
277
278         SUBROUTINE gaussfg(XPARAM,INT,ESCF,FULSCF,GRAD,LGRAD,
279        .                   coord,nquant,crdsmm,chrgs,iread,dxyzqm,
280        .                   dxyzcl,icalcf,iqmres,mark,jqatms)
281   c
282   c     Routine has been modified for qm-mm coupling and includes COSMO.
283   c
284   c     Variables for qm-mm:
285   c
286   c     coord(3,numatm) - Cartesian coordinates of qm atoms.  This was
287   c                       originally a MOPAC array defined locally in
288   c                       COMPFG and filled by GMETRY.  Now Cartesian
289   c                       coordinates are passed directly.
290   c     nquant          - Number of qm atoms.
291   c     crdsmm(*)       - Main AMBER array containing coordinates for
292   c                       all atoms (used here for mm atoms).
293   c     chrgs(*)        - Atomic charges for mm atoms.
294   c     iread           - Flag for reading in density matrix 0=no, 1=yes.
295   c     dxyzqm(3,*)     - Quantum mechanical derivatives from qm-mm
296   c                       interactions.
297   c     dxyzcl(3,*)     - Classical derivatives from qm-mm interactions.
298   c     icalcf          - Flag for calculating atomic forces 0=yes, 1=no.
299   c     iqmres(*)       - Atom-based interaction list.
300   c     mark(*)         - Center of geometry atom marker list.
301   c     jqatms(*)       - Atom numbers for qm atoms (based on AMBER scheme).
302   c
303   c
304   c               -- S. Dixon (8/11/94)
305   c
306         IMPLICIT DOUBLE PRECISION (A-H,O-Z)
307   *COMDECK SIZES
308   ************************************************************************
309   *   THIS FILE CONTAINS ALL THE ARRAY SIZES FOR USE IN MOPAC.
310   *
311   *     THERE ARE ONLY 5 PARAMETERS THAT THE PROGRAMMER NEED SET:
312   *     MAXHEV = MAXIMUM NUMBER OF HEAVY ATOMS (HEAVY: NON-HYDROGEN ATOMS)
313   *     MAXLIT = MAXIMUM NUMBER OF HYDROGEN ATOMS.
314   *     MAXTIM = DEFAULT TIME FOR A JOB. (SECONDS)
315   *     MAXDMP = DEFAULT TIME FOR AUTOMATIC RESTART FILE GENERATION (SECS)
316   *     ISYBYL = 1 IF MOPAC IS TO BE USED IN THE SYBYL PACKAGE, =0 OTHERWISE
317   *     SEE ALSO NMECI, NPULAY AND MESP AT THE END OF THIS FILE
318   *
319         PARAMETER (MAXHEV=40, MAXLIT=40)
320         PARAMETER (MAXTIM=3600, MAXDMP=3600)
321         PARAMETER (ISYBYL=1)
322   *
323   ************************************************************************
324   *
325   *   THE FOLLOWING CODE DOES NOT NEED TO BE ALTERED BY THE PROGRAMMER
```

```
*
************************************************************************
*
*    ALL OTHER PARAMETERS ARE DERIVED FUNCTIONS OF THESE TWO PARAMETERS
*
*      NAME                    DEFINITION
*     NUMATM         MAXIMUM NUMBER OF ATOMS ALLOWED.
*     MAXORB         MAXIMUM NUMBER OF ORBITALS ALLOWED.
*     MAXPAR         MAXIMUM NUMBER OF PARAMETERS FOR OPTIMISATION.
*     N2ELEC         MAXIMUM NUMBER OF TWO ELECTRON INTEGRALS ALLOWED.
*     MPACK          AREA OF LOWER HALF TRIANGLE OF DENSITY MATRIX.
*     MORB2          SQUARE OF THE MAXIMUM NUMBER OF ORBITALS ALLOWED.
*     MAXHES         AREA OF HESSIAN MATRIX
*     MAXALL         LARGER THAN MAXORB OR MAXPAR.
************************************************************************
      PARAMETER (VERSON=7.00D0)
      PARAMETER (NUMATM=MAXHEV+MAXLIT)
      PARAMETER (MAXORB=4*MAXHEV+MAXLIT)
      PARAMETER (MAXPAR=3*NUMATM)
      PARAMETER (MAXBIG=MAXORB*MAXORB*2)
      PARAMETER (N2ELEC=(50*MAXHEV*(MAXHEV-1)+10*MAXHEV*MAXLIT
     +                     +(MAXLIT*(MAXLIT-1))/2))
      PARAMETER (MAXHES=(MAXPAR*(MAXPAR+1))/2,MORB2=MAXORB**2)
      PARAMETER (MPACK=(MAXORB*(MAXORB+1))/2)
      PARAMETER (MAXPR=6*MAXORB)
      PARAMETER (MAXALL=4*MAXHEV+3*MAXLIT)
       PARAMETER (NMECI=11,  NPULAY=MPACK, MMCI=60)
      PARAMETER (MESP=10)
      PARAMETER (LENABC=600)
      PARAMETER (LENAB2=LENABC*(LENABC+5))
      PARAMETER (NPPA = 1082, MAXNSS = 500)
      PARAMETER (MAXDEN=10*MAXHEV+MAXLIT)
************************************************************************
*DECK MOPAC
      dimension crdsmm(*),chrgs(*),dxyzqm(3,*),dxyzcl(3,*),iqmres(*),
     .          mark(*),jqatms(*)
      DIMENSION XPARAM(MAXPAR),GRAD(MAXPAR)
      LOGICAL LGRAD, FULSCF
      COMMON/KILLBTL/KILLIT
      COMMON /GEOVAR/ NVAR,LOC(2,MAXPAR),IDUMY,DUMY(MAXPAR)
      COMMON /GEOSYM/ NDEP,LOCPAR(MAXPAR),IDEPFN(MAXPAR),LOCDEP(MAXPAR)
      COMMON /GEOM  / GEO(3,NUMATM)
      COMMON /ATHEAT/ ATHEAT
      COMMON /WMATRX/ WJ(N2ELEC), WK(N2ELEC)
      COMMON /ENUCLR/ ENUCLR
      COMMON /NATYPE/ NZTYPE(107),MTYPE(30),LTYPE
      COMMON /ELECT / ELECT
      PARAMETER (MDUMY=MAXPAR**2-MPACK)
      COMMON /SCRACH/ RXYZ(MPACK), XDUMY(MDUMY)
      COMMON /HMATRX/ H(MPACK)
      COMMON /GEOKST/ NATOMS,LABELS(NUMATM),
     1                NA(NUMATM), NB(NUMATM), NC(NUMATM)
      COMMON /ERRFN  / ERRFN(MAXPAR), AICORR(MAXPAR)
      COMMON /VECTOR/ C(MORB2),EIGS(MAXORB),CBETA(MORB2),EIGB(MAXORB)
      COMMON /LAST  / LAST
      COMMON /NUMCAL/ NUMCAL
      COMMON /SCFTYP/ EMIN, LIMSCF
      COMMON /MOLMEC/ HTYPE(4),NHCO(4,20),NNHCO,ITYPE
     1       /MOLKST/ NUMAT,NAT(NUMATM),NFIRST(NUMATM),NMIDLE(NUMATM),
     2                NLAST(NUMATM), NORBS, NELECS,NALPHA,NBETA,
     3                NCLOSE,NOPEN,NDUMY,FRACT
C COSMO change
      LOGICAL ISEPS, USEPS , UPDA, CFIRST
      COMMON /ISEPS/  ISEPS, USEPS, UPDA
      DATA CFIRST /.TRUE./
```

```
C end of COSMO change
C***********************************************************************
C
C   COMPFG CALCULATES (A) THE HEAT OF FORMATION OF THE SYSTEM, AND
C                     (B) THE GRADIENTS, IF LGRAD IS .TRUE.
C
C   ON INPUT  XPARAM = ARRAY OF PARAMETERS TO BE USED IN INTERNAL COORDS
C             LGRAD  = .TRUE. IF GRADIENTS ARE NEEDED, .FALSE. OTHERWISE
C             INT    = .TRUE. IF HEAT OF FORMATION IS TO BE CALCULATED
C             FULSCF = .TRUE. IF FULL SCF TO BE DONE, .FALSE. OTHERWISE.
C
C   ON OUTPUT ESCF  = HEAT OF FORMATION.
C             GRAD   = ARRAY OF GRADIENTS, IF LGRAD = .TRUE.
C
C***********************************************************************
      COMMON /KEYWRD/KEYWRD
      COMMON/ANALZZ/ICALX1,ICALX2,ICALX3,ICALCN
      CHARACTER*241 KEYWRD
      LOGICAL DEBUG, INT, PRINT, ANALYT, LARGE, USEDCI,
     1FORCE, TIMES, AIDER
      DIMENSION COORD(3,NUMATM), W(N2ELEC), DEGREE(3), XPAREF(MAXPAR)
     1,DELTAP(NMECI**2) ,DELTA(NMECI*MAXORB)
      SAVE DEGREE, PRINT, DEBUG, aider,analyt,large,usedci,
     1 force,times
      EQUIVALENCE (W,WJ)
C     DATA ICALCN /0/
      logical first_call
      data first_call /.true./
      save first_call

      !wpp
      !上の部分について不要な部分多数あり
      ! メモ：
      ! * gaussgfのnquantにはlink原子の数がすでに含まれている
      ! * jqatms(i)は 原子iがQMのとき、その原子番号
      !                    Linkのとき、-1
      !                      MMのとき、0  を返す
      !
      ! 1 hartree = 627.51d0 kcal/mol
      ! 1 bohr = 0.52917706d0 angstrom

      integer io1, ierr
      parameter (io1=70)

      double precision qscale, escale, fscale
      data qscale /18.2223d0/
      parameter (escale=627.51d0, fscale=escale/0.52917706d0)

      integer natom_gfg, nqm_gfg, nln_gfg, nqmln_gfg,
     &        iqmln_gfg(1000), immln_gfg(1000)
      logical debug_gfg, charg_gfg, keeprwf_gfg                         ! YK ADDED
      integer nsstart_gfg, nskeep_gfg, nsdel_gfg, igcharg_gfg           ! YK ADDED
      character gh_route_gfg*800, sep_gfg*2                             ! YK A&CHANGED

      common /gfg/ natom_gfg, nqm_gfg, nln_gfg, nqmln_gfg,
     &             iqmln_gfg, immln_gfg,
     &             debug_gfg, charg_gfg, keeprwf_gfg,                   ! YK ADDED
     &             nsstart_gfg, nskeep_gfg, nsdel_gfg, igcharg_gfg,     ! YK ADDED
     &             gh_route_gfg, sep_gfg                                ! YK ADDED

      integer i, j, indxmm, ni_mm, jmm, jcoord
      logical fcharg, mmflgs(natom_gfg)
      double precision rx, ry, rz, gx, gy, gz, gmx, gmy, gmz
      double precision ln_r, one_r, gr3
      double precision e_qm, e_qmmm, e_mm                               ! YK ADDED
```

```
456         integer ipstt, ipend, len_gh, ntime_call, nstep_cnt               ! YK ADDED
457         data ntime_call /-2/    ! because of initial setting in roar      ! YK ADDED
458         data nstep_cnt /0/                                                ! YK ADDED
459         save ntime_call, nstep_cnt                                        ! YK ADDED
460         logical gdelete                                                   ! YK ADDED
461         character grwf*130, distime*10                                    ! YK ADDED
462
463   !      if(first_call) then
464   !        first_call = .false.
465   !      endif
466         ntime_call = ntime_call + 1                                       ! YK ADDED
467   CD     print '(a, 3l2, 4i5)', "SETTING2",  debug_gfg, charg_gfg,
468   CD    &  keeprwf_gfg, nsstart_gfg, nskeep_gfg, nsdel_gfg, igcharg_gfg
469
470         !リンク原子の位置を決める
471         do i=1, nln_gfg
472           do j=1,nqm_gfg
473             if(jqatms(j)==iqmln_gfg(i)) then
474               rx = crdsmm((immln_gfg(i)-1)*3+1) - coord(1,j)
475               ry = crdsmm((immln_gfg(i)-1)*3+2) - coord(2,j)
476               rz = crdsmm((immln_gfg(i)-1)*3+3) - coord(3,j)
477               exit
478             endif
479           enddo
480           ln_r = 1.09d0/sqrt(rx*rx+ry*ry+rz*rz)
481           do j=1,nqm_gfg
482             if(jqatms(j)==iqmln_gfg(i)) then
483               coord(1,nqm_gfg+i) = coord(1,j) + rx*ln_r
484               coord(2,nqm_gfg+i) = coord(2,j) + ry*ln_r
485               coord(3,nqm_gfg+i) = coord(3,j) + rz*ln_r
486               exit
487             endif
488           enddo
489         enddo
490
491         ! QMと相互作用するMM原子を検索
492         fcharg = .false.
493         mmflgs = .false.
494         indxmm = 1
495         do i=1,nqmln_gfg
496           ni_mm = iqmres(indxmm)
497           indxmm = indxmm + 1
498           do j=1,ni_mm
499             jmm = iqmres(indxmm)
500             jcoord = (jmm-1)*3
501             mmflgs(jmm) = .true.
502             indxmm = indxmm + 1
503           enddo
504         enddo
505         if(any(mmflgs == .true.)) fcharg = .true.
506
507         ! Link原子に結合しているMM原子を省く
508         if(fcharg) then
509           do i=1, nln_gfg
510             mmflgs(immln_gfg(i)) = .false.
511           enddo
512         endif
513
514         ! gaussian input fileを作成する
515         open(io1,file='gaussian.in',status='replace',action='write')
516   C YKYKYK 3                                                              ! YK DELETED
517   C      ! gjf:header
518   C      write(io1,'(a)') '%nosave'
519   C      write(io1,'(a)') '%rwf=elec.rwf'
520   C      if(fcharg) then
```

```
C        write(io1,'(a)') trim(gh_route_gfg)//' force iop(6/13=1) charge'
C      else
C        write(io1,'(a)') trim(gh_route_gfg)//' force iop(6/13=1)'
C      endif
C      write(io1,'(a)')
C      write(io1,'(a)') 'QM/MM'
C      write(io1,'(a)')
C      write(io1,'(a)') gh_molsp_gfg
C YKYKYK 3                                                          ! YK DELETED
C YKYKYK 3                                                          ! YK CHANGED
       len_gh = len_trim(gh_route_gfg)
       ipend  = 0
       !D write(6, *) trim(gh_route_gfg)
       do
         if(ipend == len_gh) exit
         ipstt = ipend + 1
         ipend = index(gh_route_gfg(ipstt: len_gh), sep_gfg) + ipstt - 2
         !D print '(a)', gh_route_gfg(ipstt: len_gh)
         if(fcharg .and.
     &     (index(gh_route_gfg(ipstt: ipend), "force") /= 0)) then
           write(io1, '(a)') gh_route_gfg(ipstt: ipend)//" charge"
           ipend = ipend + 2
           cycle
         endif
         if(index(gh_route_gfg(ipstt: ipend), "%rwf") /= 0) then
           if(charg_gfg .or. keeprwf_gfg) then
             ipend = ipend - 4
             if(ntime_call >= 0) then
               write(grwf, '(a, i10.10, a)')
     &           gh_route_gfg(ipstt: ipend)//"-", ntime_call, ".rwf"
             else
               write(grwf,'(a)') gh_route_gfg(ipstt: ipend)//"-Init.rwf"
             endif
             write(io1, '(a)') trim(grwf)
             ipend = ipend + 6
             cycle
           else
             write(grwf, '(a)') gh_route_gfg(ipstt: ipend)
           endif
         endif
         if(ipstt < ipend) then
           write(io1, '(a)') gh_route_gfg(ipstt: ipend)
         elseif(ipstt > ipend) then
           write(io1, *)
         else
           call abort("Something Wrong")
         endif
         ipend = ipend + 2
       enddo
C YKYKYK 3                                                          ! YK CHANGED

      ! gjf:QM
      !labels,coordは先頭から使うので、i->jqatms(i)としなくてよい
      do i=1,nqmln_gfg
        write(io1,'(i,3f)') labels(i),(coord(j,i),j=1,3)
      enddo
      write(io1,'(a)')

      ! gjf:MM
      if(fcharg) then
        do i=1,natom_gfg
          if(mmflgs(i)) then
            write(io1,'(4f)')
     &        (crdsmm((i-1)*3+j),j=1,3), chrgs(i)/qscale
          endif
```

```
          enddo
          write(io1,'(a)')
          do i=1,natom_gfg
            if(mmflgs(i)) then
              write(io1,'(4f)')
     &        (crdsmm((i-1)*3+j),j=1,3)
            endif
          enddo
          write(io1,'(a)')
        endif

        call flush(io1)                                                 ! YK ADDED
        close(io1)

C YKYKYK 4                                                              ! YK ADDED
        gdelete = .true.
        if(ntime_call >= nsstart_gfg) then
          gdelete   = .false.
          nstep_cnt = nstep_cnt + 1
          if(nstep_cnt == nskeep_gfg) then
            nsstart_gfg = nsstart_gfg + (nskeep_gfg + nsdel_gfg)
            nstep_cnt   = 0
          endif
        endif
C YKYKYK 4                                                              ! YK ADDED
        ! Gaussianを実行する

C YKYKYK 5                                                              ! YK DELETED
C       ierr = runqq('cat','gaussian.in >> gaussian-debug.log')
C       ierr = runqq('rm','-f elec.rwf')
C YKYKYK 5                                                              ! YK DELETED
C YKYKYK 5                                                              ! YK CHANGED
        if(debug_gfg)
     &  ierr = runqq('cat','gaussian.in >> gaussian-debug.log')
        if(charg_gfg .and. .not.gdelete) then
          ierr = runqq('cat','gaussian.in >> gaussian-charg.log')
          ierr = runqq('echo','"--Link1--" >> gaussian-charg.log')
        endif
C YKYKYK 5                                                              ! YK CHANGED

        ierr = runqq('g03b3','gaussian.in') ! output: elec.rwf
        if(ierr==-1) then
          write(6,*) 'FATAL ERROR IN GAUSSFG - g03 FAILS'
          stop
        endif

C YKYKYK 6                                                              ! YK DELETED
C       ierr = runqq('rm','-f gdatas.wpp')
C       ierr = runqq('./PrtGVal3.exe','elec.rwf') ! output: gdatas.wpp
C       if(ierr==-1) then
C         write(6,*) 'FATAL ERROR IN GAUSSFG - PrtGval3 FAILS'
C         stop
C       endif
C YKYKYK 6                                                              ! YK DELETED
C YKYKYK 6                                                              ! YK CHANGED
        ierr = runqq('rm','-f gdatas.wpp')
        len_gh = len_trim(grwf)
        if(len_gh > 128) call abort("CHANGE LENGTH of len_gh")
        write(grwf(len_gh + 1: len_gh + 2), '(a, i1.1)')
     &  " ", igcharg_gfg
        !D print *, grwf(6: len_gh + 2)
        ierr = runqq('/home/home2/yamada/PRTGVAL/PrtGVal4b.exe',
     &  grwf(6: len_gh + 2)) ! output: gdatas.wpp
        if(ierr==-1) then
          write(6,*) 'FATAL ERROR IN GAUSSFG - PrtGVal FAILS'
```

```
651           stop
652         endif
653
654         if(.not.keeprwf_gfg .or. gdelete)
655        &  ierr = runqq('rm','-f '//grwf(6: len_gh))
656         if(igcharg_gfg /= 0 .and. .not.gdelete) then
657           if(ntime_call >= 0) then
658             write(distime, '(i10.10)') ntime_call
659           else
660             write(distime, '(a)') "Init"
661           endif
662           ierr = runqq('echo',trim(distime)//' >> gd-charge.wpp')
663           ierr = runqq('cat','gd-charge.tmp >> gd-charge.wpp')
664         endif
665         ierr = runqq('rm','-f gd-charge.tmp')
666   C YKYKYK 6                                                          ! YK CHANGED
667
668         ! ファイルからエネルギーと力を読み込む
669         open(io1,file='gdatas.wpp',status='old',action='read')
670         rewind(io1)
671
672         read(io1,'(a)')
673         ! エネルギー
674   C YKYKYK 7                                                          ! YK DELETED
675   C       read(io1,'(3f)') escf, elect, enuclr
676   C!      write(*,*) '!wpp ener:', escf*escale, elect*escale, enuclr*escale
677   C       elect = 0.d0
678   C       enuclr = escf*(escale/23.060362d0)
679   C       escf = (elect+enuclr)*23.060362d0
680   C YKYKYK 7                                                          ! YK DELETED
681   C YKYKYK 7                                                          ! YK CHANGED
682         elect  = 0.0d0
683         enuclr = 0.0d0
684         read(io1, '(2x, 3f23.15)') e_qm, e_qmmm, e_mm
685         escf = e_qm * escale ! Hartree -> kcal/mol
686   C YKYKYK 7                                                          ! YK CHANGED
687         if(escf<emin.or.emin==0.d0) emin=escf
688
689         if(icalcf/=0) then
690           close(io1)
691           return !エネルギーだけ計算して力はupdateしない
692         endif
693
694         read(io1,'(a)')
695         ! QM原子とLink原子
696         !dxyzqmは先頭から使う
697         do i=1,nqmln_gfg
698           read(io1,'(3f)') gx, gy, gz
699           dxyzqm(1,i) = -gx*fscale
700           dxyzqm(2,i) = -gy*fscale
701           dxyzqm(3,i) = -gz*fscale
702         enddo
703         read(io1,'(a)')
704         ! MM原子
705         !dxyzclはそのままの配列
706         if(fcharg) then
707           do i=1,natom_gfg
708             if(mmflgs(i)) then
709               read(io1,'(3f)') gx, gy, gz
710               dxyzcl(1,i) = -gx*fscale
711               dxyzcl(2,i) = -gy*fscale
712               dxyzcl(3,i) = -gz*fscale
713             endif
714           enddo
715         endif
```

```
716
717         close(io1)
718
719         ! Link原子の力をQM原子,MM原子に分配する
720         do i=1, nln_gfg
721           gx = dxyzqm(1,nqm_gfg+i)
722           gy = dxyzqm(2,nqm_gfg+i)
723           gz = dxyzqm(3,nqm_gfg+i)
724           do j=1,nqm_gfg
725             if(jqatms(j)==iqmln_gfg(i)) then
726               rx = crdsmm((immln_gfg(i)-1)*3+1) - coord(1,j)
727               ry = crdsmm((immln_gfg(i)-1)*3+2) - coord(2,j)
728               rz = crdsmm((immln_gfg(i)-1)*3+3) - coord(3,j)
729               exit
730             endif
731           enddo
732           one_r = 1.d0/sqrt(rx*rx+ry*ry+rz*rz)
733           rx = rx*one_r
734           ry = ry*one_r
735           rz = rz*one_r
736           ln_r = 1.09d0*one_r
737           gr3 = gx*rx+gy*ry+gz*rz
738           gmx = ln_r*(gx-gr3*rx)
739           gmy = ln_r*(gy-gr3*ry)
740           gmz = ln_r*(gz-gr3*rz)
741           dxyzcl(1,immln_gfg(i)) = dxyzcl(1,immln_gfg(i)) + gmx
742           dxyzcl(2,immln_gfg(i)) = dxyzcl(2,immln_gfg(i)) + gmy
743           dxyzcl(3,immln_gfg(i)) = dxyzcl(3,immln_gfg(i)) + gmz
744           do j=1,nqm_gfg
745             if(jqatms(j)==iqmln_gfg(i)) then
746               dxyzqm(1,j) = dxyzqm(1,j) + (gx-gmx)
747               dxyzqm(2,j) = dxyzqm(2,j) + (gy-gmy)
748               dxyzqm(3,j) = dxyzqm(3,j) + (gz-gmz)
749               exit
750             endif
751           enddo
752           dxyzqm(1,nqm_gfg+i) = 0.0d0
753           dxyzqm(2,nqm_gfg+i) = 0.0d0
754           dxyzqm(3,nqm_gfg+i) = 0.0d0
755         enddo
756
757   !      write(*,'(a)') 'gaussfg: gradient(QM)'
758   !      do i=1,nqmln_gfg
759   !        write(*,'(i5,3(2x,e15.7))') jqatms(i),(dxyzqm(j,i),j=1,3)
760   !      enddo
761   !      write(*,'(a)') 'gaussfg: gradient(MM)'
762   !      do i=1,natom_gfg
763   !        write(*,'(i5,3(2x,e15.7))') i,(dxyzcl(j,i),j=1,3)
764   !      enddo
765
766    4000 RETURN
767         END
768
769   ! ----------------------------------------------------------------------
770         function rdline(ipos, line) result(num)
771           implicit none
772           integer, intent(in)          :: ipos
773           character(len=*), intent(in) :: line
774
775           integer :: num, lenline, ipstt, ipend
776
777           lenline = len_trim(line)
778           ipstt = index(line(ipos: lenline), '=') + ipos
779           do
780             ipend = index(line(ipstt: lenline), ' ') + ipstt - 1
```

```
781             if(ipend == ipstt) then
782               ipstt = ipend + 1
783             elseif(ipend < ipstt) then
784               ipend = lenline
785               exit
786             else
787               exit
788             endif
789           enddo
790
791           read(line(ipstt: ipend), '(i)') num
792           return
793         end function rdline
```