Korean J. Crystallography
Vol. 17, No. 2, pp.55~59, 2006

ORTHONDISANG: Interatomic Distances and Angles of Atoms with Orthonormal Coordinates

Young-Soo Park[a], Won-Sik Han[b], Tae-Jin Kim[b], Sang Ook Kang[b] and Il-Hwan Suh[b]

[a]Miraegisul Co. 1419 HO, Hongin Officetel Bongmyeong-dong, Yuseong-gu, Daejeon 305-710, Korea
[b]Department of Material chemistry, Korea University, Jochiwon 339-700, Korea

1. The Crystallographic Problem and Method of Solution

Computer programs for the molecular orbital calculation such as Gaussian 98 W,[1] DMol3,[2,3] LUMMOX[4] and ADF[5] utilize only the orthonormal coordinates. Therefore in order to apply MO calculation method to any molecule elucidated by X-ray crystallography, the atomic coordinates must be transformed into orthonormal ones. The program ORTHON[6] can be used for this purpose.

The program LUMMOX additionally requires that one of atoms in a molecule with the orthonormal coordinates must be translated to the origin of the orthonormal coordinate system, which means all other atoms in the molecule are also translated accordingly. The program ORTHONZERO[7] can take care of the translating role.

The title computer program ORTHONDISANG not only confirms how accurately both ORTHON and ORTHONZERO transform and translate the input coordinates for the MO calculation but also verify how accurately the Gaussian, ADF and DMol3 do their geometry optimization by offering interatomic distances and angles of atoms with the orthonormal coordinates.

2. How to Run the Program

The procedure to run the ORTHONDISANG is:

2.1. Enter ORTHONDISANG.exe

2.2. This program requires an input file name, in free format, which must be prepared by yourself.

The data for the input file are:

atomic names and their orthonormal coordinates x, y, z.

An example:

c1	4.89083	4.02806	5.71916
c2	4.37874	4.01137	4.41700
c3	3.65648	5.10563	3.91097

You can put in up to 20000 atoms in the input file, one atom per line.

2.3. After reading the input file name, the program requires a name of an output file for the output data.

2.4. Lastly the program requires the minimum and maximum bond lengths you want to include in this calculation. They can be written, for instance, as:

0.7, 1.8

An example of an output data is:

Input data: atomic names and orthonormal coordinates

c1	4.89083	4.02806	5.71916
c2	4.37874	4.01137	4.41700
c3	3.65648	5.10563	3.91097

Atomic bond distances (Angstrom) and bond angles (radian and degree)

A B C	d(A-B)	d(B-C)	d(C-A)	s	radian	degree
c1-c2-c3	1.39933	1.40539	2.44015	-0.51384	2.11045	120.92014

3. Software and Hardware Environment

The ORTHONDISANG is compiled with Intel Fortran 8.0 complier and will run on windows PC such as winxp/win2003/win2000.

The source program of ORTHONDISANG is

shown below.

5. Key words

Molecular orbital (MO), Gaussian 98W, ADF, DMol3

References

1) Gaussian 98, Revision A.3, M. J. Frisch, G. W. Trucks, H. B. Schlegel, G. E. Scuseria, M. A. Robb, J. R. Cheeseman, V. G. Zakrzewski, J. A. Montgomery, Jr., R. E. Stratmann, J. C. Burant, S. Dapprich, J. M. Millam, A. D. Daniels, K. N. Kudin, M. C. Strain, O. Farkas, J. Tomasi, V. Barone, M. Cossi, R. Cammi, B. Mennucci, C. Pomelli, C. Adamo, S. Clifford, J. Ochterski, G. A. Petersson, P. Y. Ayala, Q. Cui, K. Morokuma, D. K. Malick, A. D. Rabuck, K. Raghavachari, J. B. Foresman, J. Cioslowski, J. V. Ortiz, B. B. Stefanov, G. Liu, A. Liashenko, P. Piskorz, I. Komaromi, R. Gomperts, R. L. Martin, D. J. Fox, T. Keith, M. A. Al-Laham, C. Y. Peng, A. Nanayakkara, C. Gonzalez, M. Challacombe, P. M. W. Gill, B. Johnson, W. Chen, M. W. Wong, J. L. Andres, C. Gonzalez, M. Head-Gordon, E. S. Replogle, and J. A. Pople, Gaussian, Inc., Pittsburgh PA, 1998.
2) B. Delley, *J. Chem. Phys.*, **92**, 508 (1990).
3) B. Delley, *J. Chem. Phys.*, **113**, 7756 (2000).
4) Akinobu Shiga, LUMMOX, (private communication).
5) SCM/Vrije Universiteit, Theoretical Chemistry, Amsterdam, The Netherlands. http://www.scm.com.
6) Il-Hwan Suh, Young-Soo Park and Jin-Gyu Kim, ORTHON: transformation from triclinic axes and atomic coordinates to orthonormal ones, *J. Appl. Crst.*, **33**, 994 (2000).
7) Young-Soo Park, Won-Sik Han, Tae-Jin Kim, Sang Ook Kang, Il-Hwan Suh, *Korean J. of Crystallography*, 17(1), 19 (2006).

```
Program ORTHONDISANG

character input*80,output*80,aname*80
character atom(20000)*4
real a(20000), b(20000), c(20000)
real dist, the, deg,s, d1,d2,d3
real mn,mx
integer i,j,k,cnt,pos,m,n

print *,"
print *,'          ****************************************'
print *,'          *          Program ORTHONDISANG        *'
print *,'          *            by Young-Soo Park         *'
print *,'          *       e-mail: gillboard@daum.net     *'
print *,'          ****************************************'
print *,"
print *,'     This program calculates atomic bond distances and
+bond'
print *,'     angles provided the atoms have the orthonormal coo
+rdinates.'
print *,"
print *,' (1) This program requires an input file name, IN FREE'
print *,'     FORMAT, which must be provided by YOURSELF.'
print *,"
```

```
      print *,'     The data for the input file are:'
      print *,'     atomic names and their orthonormal coordinates x,
     +y, z.'
      print *,'     An example:'
      print *,'                c1  4.89083  4.02806  5.71916'
      print *,'                c2  4.37874  4.01137  4.41700'
      print *,'                c3  3.65648  5.10563  3.91097'
      print *,''
      print *,'     You can put in up to 20000 atoms in the input file
     +, one'
      print *,'     atom per line.'
      print *,''
      print *,' (2) After reading the input file name, the program req
     +uires a name of'
      print *,'     an output file for the output data.'
      print *,''
      print *,' (3) Lastly the program requires the minimum and maximu
     +m bond lengths you'
      print *,'     want to include in this calculation. They can be w
     +ritten as:'
      print *,'          0.7,1.8'
      print *,''

      type 111
  111  format(' [Enter Input file...?] : ',$)
      accept 80,input
      print*,' '

      type 222
  222  format(' [Enter Output file..?] : ',$)
      accept 80,output
      print*,' '

      type 333
  333  format(' [Enter min, max distance..?] : ',$,$)
      accept *,mn,mx

      open(unit=11,file=input)
      open(unit=12,file=output)

      write(12,*) 'Input data: atomic names and orthonormal coordinates'
      write(12,*) ''
      do 10 n=0,20000
      read(11,80, END=9) aname
      write(12,80) aname
   10  continue
```

```
 9     rewind(unit=11)

       do 20 n=0,20000
       read(11,*,END=900) atom(n), a(n), b(n), c(n)
       if (atom(n).eq.'') then
       goto 900
       endif
 20    continue
 900   close(unit=11)

       total=n-1

       write(12,*) ''
       write(12,*) 'Atomic bond distances (Angstrom) and angles (radian
      +and degree)'
       write(12,*) ''
       write(12,400) 'A     B     C          d(A-B)      d(B-C)      d(A-C)
      +      s              radian       degree   '
       do 201 i=0,total
       do 202 j=0,total
       do 203 k=i+1,total

       if(i.ne.j.and.j.ne.k) then
       call getD(a(i),b(i),c(i),a(j),b(j),c(j),d1)
       call getD(a(j),b(j),c(j),a(k),b(k),c(k),d2)
       call getD(a(k),b(k),c(k),a(i),b(i),c(i),d3)
       call getT(a(i),b(i),c(i),a(j),b(j),c(j),a(k),b(k),c(k),s,the,deg)

       if(d1.ge.mn.and.d1.le.mx)then
       if(d2.ge.mn.and.d2.le.mx)then
       write(12,FMT=300) atom(i), atom(j), atom(k),d1,d2,d3,s,the, deg
       endif
       endif

       endif

 203   continue
 202   continue
 201   continue
       goto 999

 80    format(a80)
 100   format(a4,1x,3(f12.5))
 200   format(a4,'-',a4,1x,f12.5)
 300   format(a4,'-',a4,'-',a4,1x,6(f12.5))
 400   format(a87)
```

```
888    close(11)
       close(12)
       print*,' '
       stop '    !! Invalid input file format !!'

999    close(11)
       close(12)
       print*,' '
       stop '    !! Program terminated successfully !!'

      end

      subroutine getD(x1, y1, z1, x2, y2, z2, d)

      real x1, y1, z1, x2, y2, z2, d
      d = sqrt( ((x1-x2)**2) + ((y1-y2)**2) + ((z1-z2)**2) )
      return
      end

      subroutine getT(x1, y1, z1, x2, y2, z2, x3, y3, z3, s, t, d)

      real x1, y1, z1, x2, y2, z2, x3, y3, z3, t, d
      real a, b, c, s, pi
      pi=3.1415926536

      a= ((x1-x2)*(x3-x2)) + ((y1-y2)*(y3-y2)) + ((z1-z2)*(z3-z2))
      b= sqrt( ((x1-x2)**2) + ((y1-y2)**2) + ((z1-z2)**2) )
      c= sqrt( ((x3-x2)**2) + ((y3-y2)**2) + ((z3-z2)**2) )

      s= (a/(b*c))
      t= acos(s)
      d= t*(180.0/pi)
      return
      end
```