

文部科学省 ポスト「京」重点課題⑥
「革新的クリーンエネルギーシステムの実用化」

第 2 回東京ハンズオン:ABINIT-MP 講習会

手引書

2019 年 3 月 27 日

目次

1	課題 1 : TrpCage の IFIE 解析と CHPI 解析	1
1.1	入力データ作成	1
1.2	データの転送と FMO 計算の実行	8
1.3	CHPI 解析	10
1.4	IFIE 解析	13
2	課題 2 : エストロゲン受容体ーリガンド複合体の IFIE 解析	18
2.1	入力データ作成	18
2.2	FMO 計算	18
2.3	IFIE 解析	19
3	課題 3 : ペプチド結合での切断	23
3.1	入力データ作成	24
3.2	FMO 計算	25
3.3	IFIE 解析	25
4	課題 4 : 軌道レベルでの電荷移動相互作用解析 (CAFI 解析)	27
4.1	入力データの作成	27
4.2	FMO 計算	29
4.3	CAFI 解析	30
5	課題 5 : 軌道レベルでの分極相互作用解析 (FILM 解析)	31
5.1	入力データの作成	31
5.2	FMO 計算	31
5.3	FILM 解析	32
6	課題 6 : 局所応答分散 (LRD) と MP2 の比較	34
6.1	入力データの作成	34
6.2	FMO 計算	35
6.3	結果の確認	35

1 課題 1 : TrpCage の IFIE 解析と CHPI 解析

本課題では小規模タンパク質 TrpCage の FMO-MP2 計算の入力データの作成、計算の実行を行い、フラグメント間相互作用エネルギー (IFIE) 及び CH/ π (CHPI) 相互作用解析を行います。

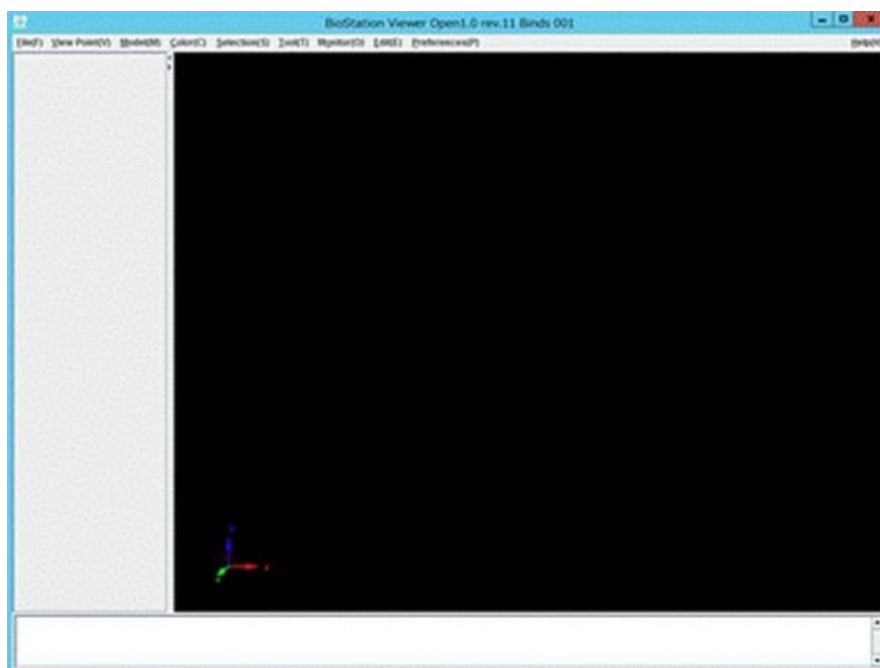
※今回の演習では、計算時間の短縮のため、HF/STO-3G で計算を流します。

1.1 入力データ作成

BioStation Viewer を起動します。デスクトップ上のアイコンをクリックします。



BioStation Viwer が起動します。



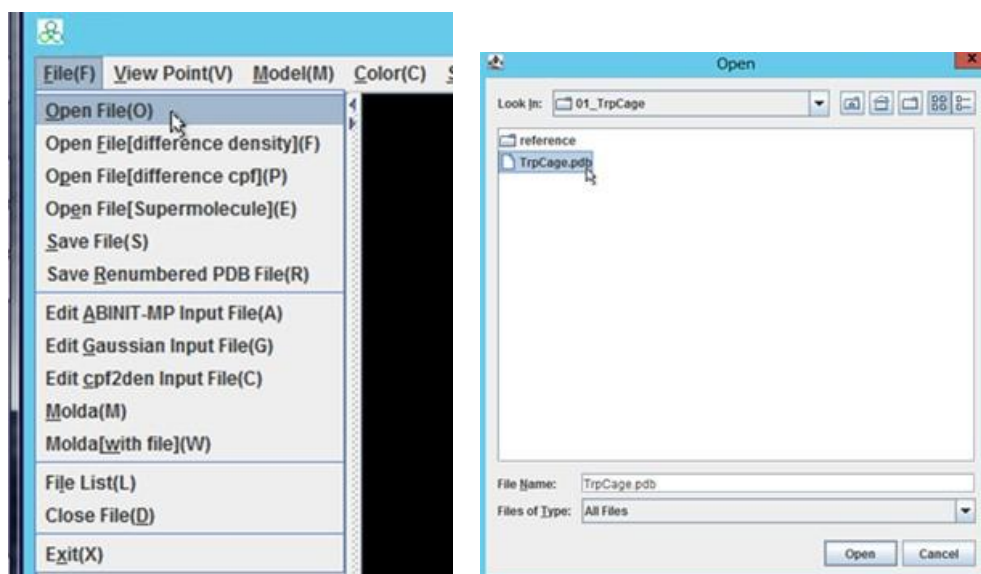
TrpCage の構造を BioStation Viwer 読み込みます。TrpCage の構造はデスクトップ上の以下のフォルダに格納されています。

ABINIT-MP_HANDSON_20190327\01_TrpCage

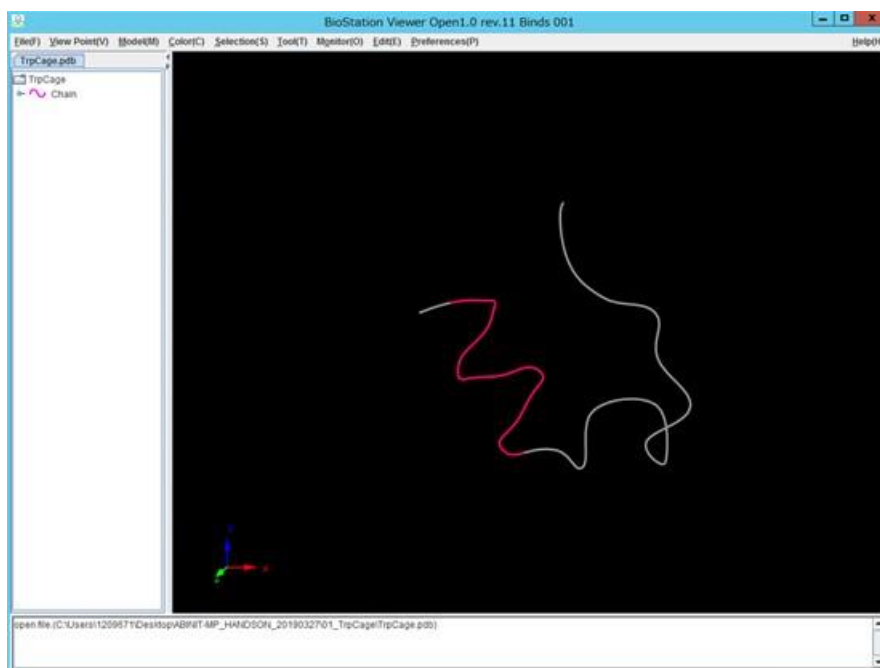
同じファイル一式は、計算に使用する FX-10 の以下のフォルダにも置いてあります。赤字部分は各位のアカウント番号となります。

/home/acourse/abinit01/ABINIT-MP_HANDSON_20190327/01_TrpCage

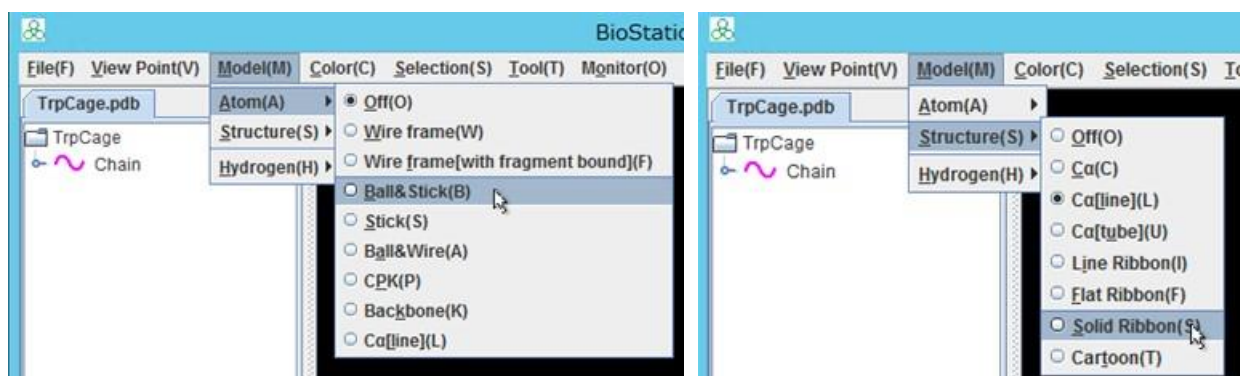
構造ファイル（PDB ファイル）を読み込むには、File→Open File を実行するか、BioStation Viwer の画面に PDB ファイルをドラッグ&ドロップします。この説明以降はドラッグ&ドロップの方法のみ説明します。



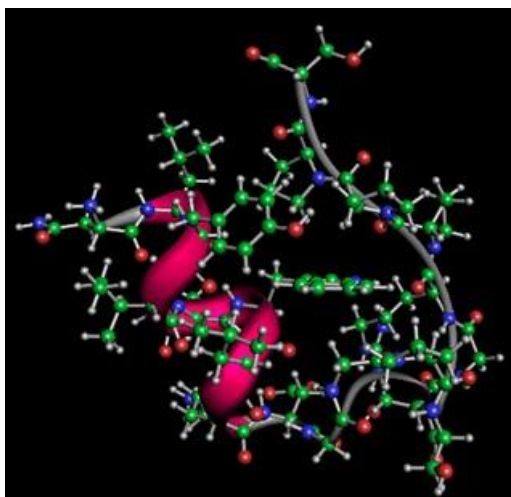
TrpCage の構造が表示されます。デフォルトの表示ではタンパク質の骨格のみが表示されます。（主鎖の C α 原子をラインでつないだ表示）



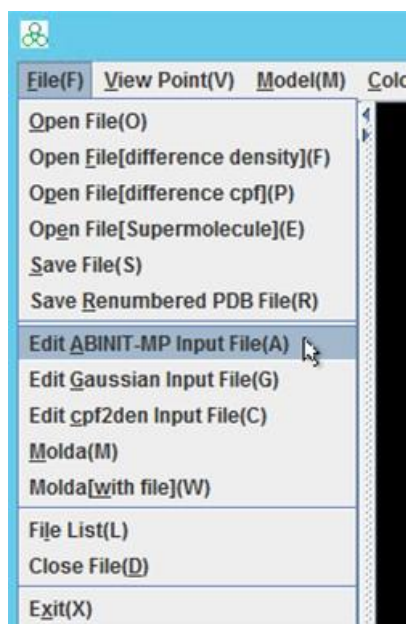
Model→Atom→Ball&Stick と Model→Structure→Solid Ribbon を実行して表示方法を変更しましょう。



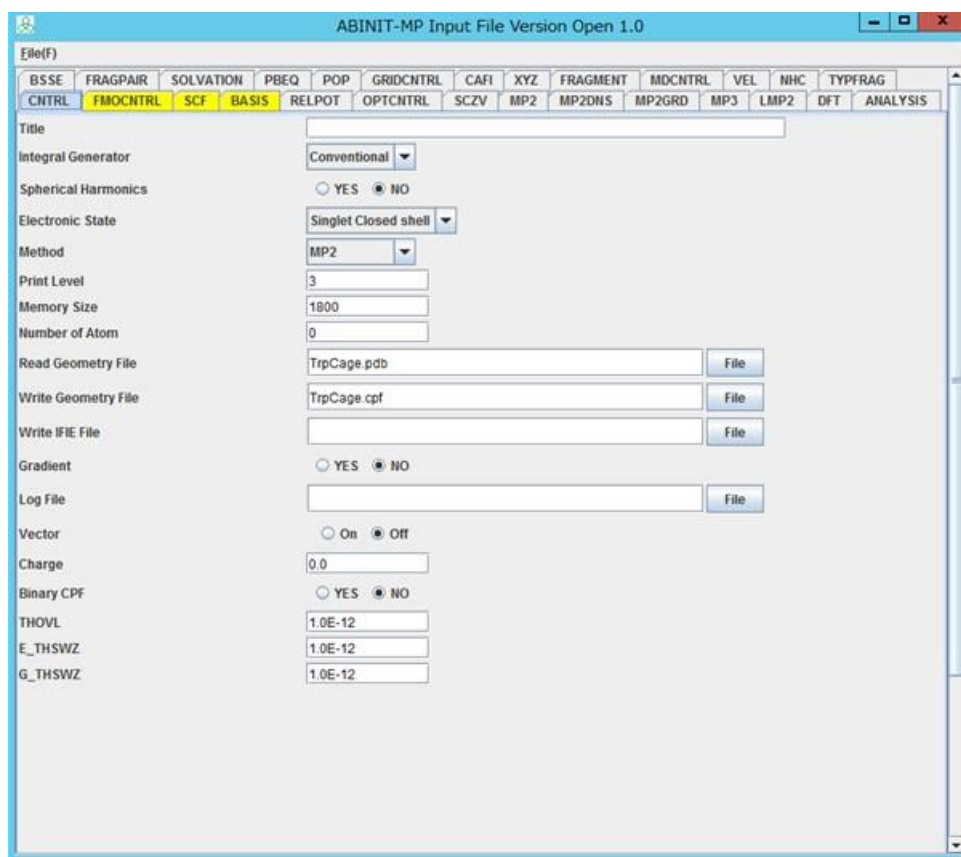
TrpCage が以下のように表示されます。



FMO 計算の入力ファイルを作成するには、File→Edit ABINIT-MP Input File を実行します。



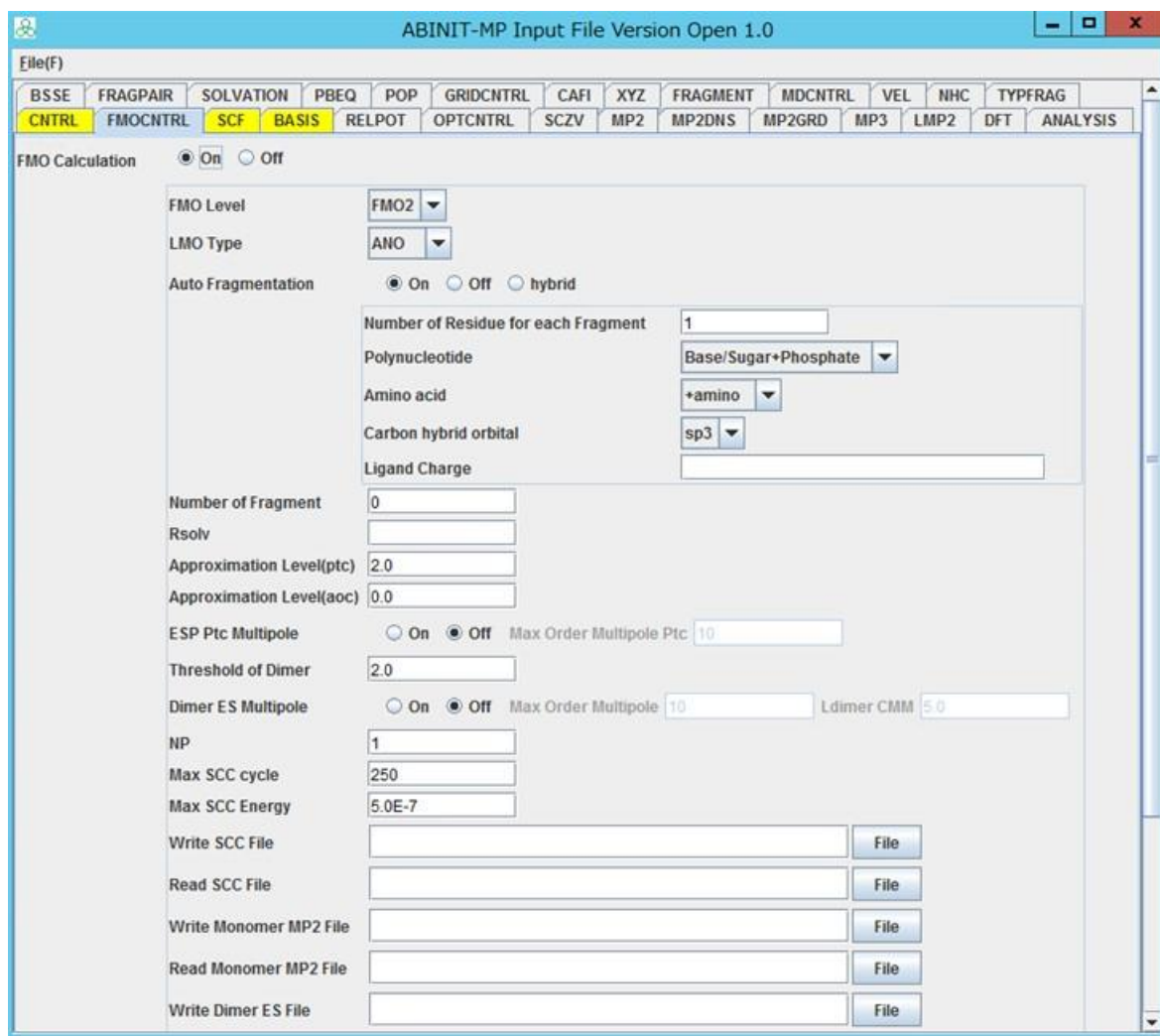
File→Edit ABINIT-MP Input File を実行すると、以下のウィンドウが起動します。CNTRL、FMOCNTRL、SCF など複数のタブがあり、それらの中で各種のパラメータが指定できます。



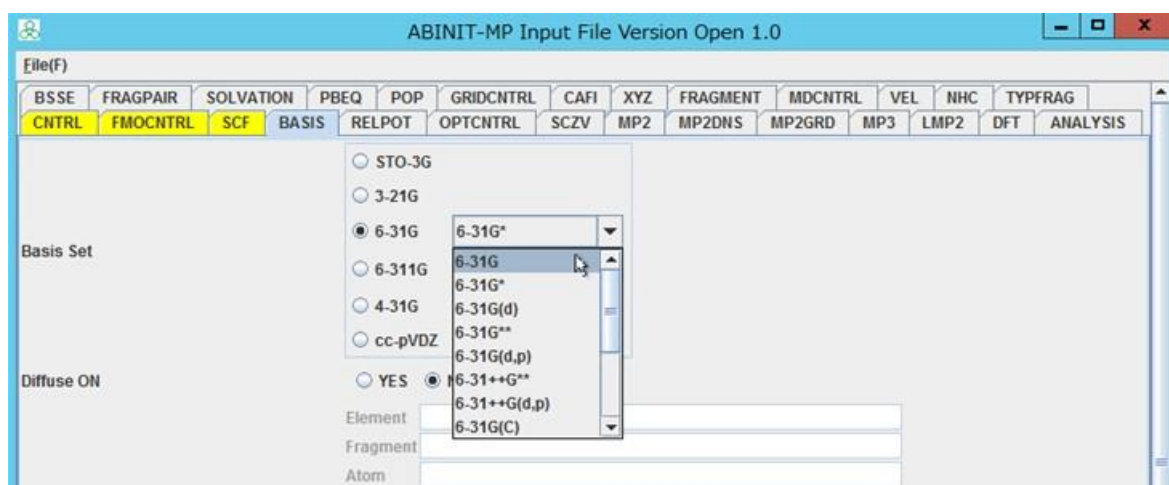
CNTRL タブの Method で計算レベルを制御できます。MP2 になっていることを確認してください。次に、Read Geometry File を TrpCage.pdb とします。Write Geometry File に TrpCage-FMO-MP2.cpf と入力します。



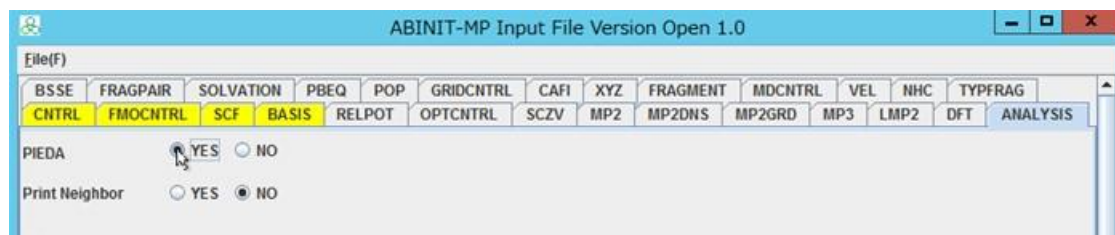
次に、FMOCTRL タブにて FMO 関連のパラメータを設定します。本課題では特に変更はありません。



BASIS タブにて基底関数関連のパラメータを指定します。6-31G*から 6-31G に変更してください。
通常は 6-31G*を用いた計算を行うことが推奨されますが、今回は 6-31G で実施します。



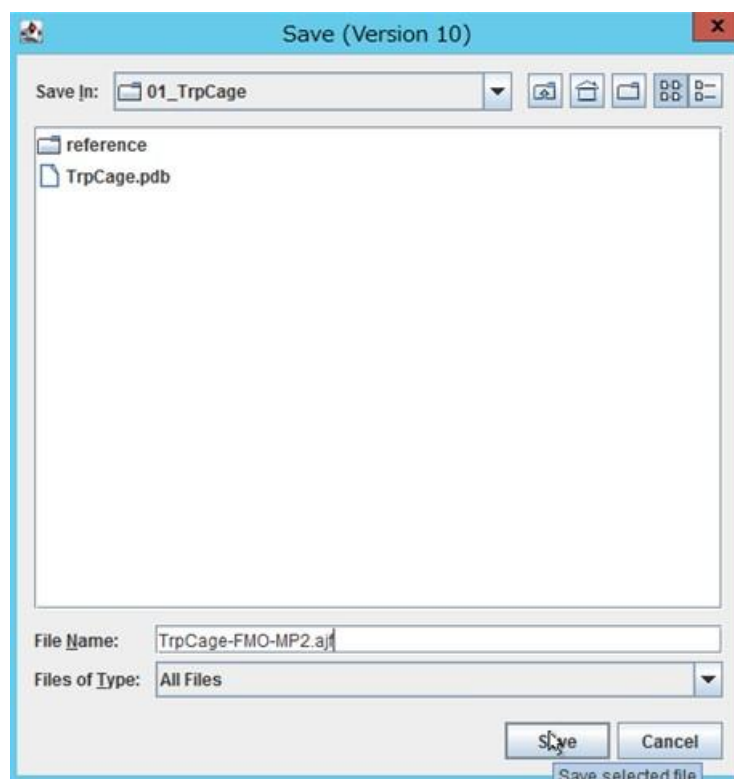
また、Analysis タブにて PIEDA を “YES” とすることで PIEDA の計算も行われます。



パラメータの設定が完了したら File→Save File を実行して入力ファイルを保存します。



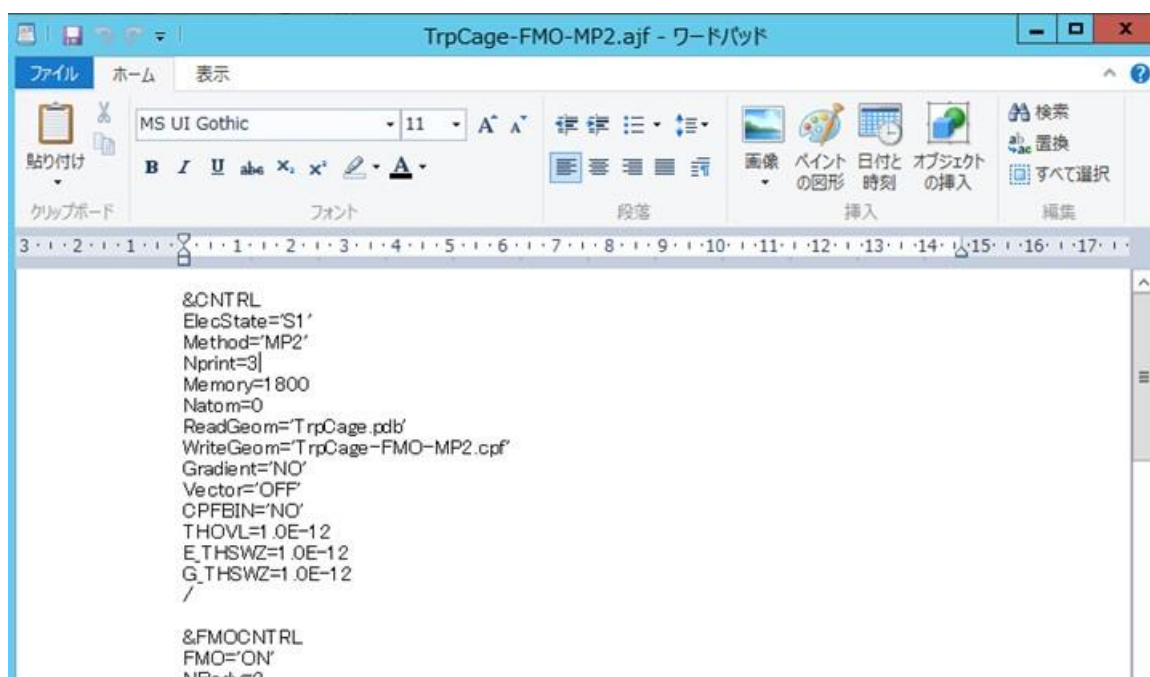
ここでは TrpCage-FMO-MP2.ajf という名前で保存します。



PDB ファイルのあるフォルダに TrpCage-FMO-MP2.ajf が出力されます。



テキストエディタなどで開いて中身を確認します。



今回の演習では、計算時間の短縮のため、HF/STO-3G レベルの計算を流します。

まず、TrpCage-FMO-MP2.ajf をコピーして、TrpCage-FMO-HF.ajf を作成します。テキストエディタで開き、&CNTRL ネームリストの Method='MP2'のところを、Method='HF'に変更します。WriteGeom も 'TrpCage-FMO-HF.ajf' にしておきましょう。さらに、&BASIS ネームリストの 'BasisSet='6-31G' を 'BasisSet='STO-3G'に変更します。

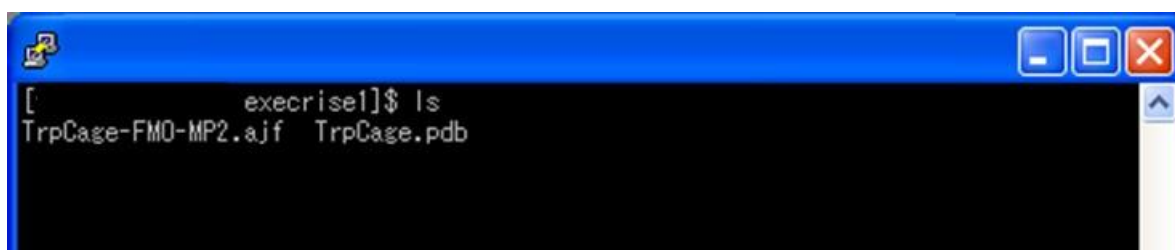
これで、HF/STO-3G 計算用の入力ファイルができました。

1.2 データの転送と FMO 計算の実行

次に Linux の計算サーバに入力データ（PDB ファイルと AJF ファイル）をアップロードし、FMO 計算を実行します。ここでは WinSCP というソフトウェアを用います。



ターミナル画面を開き、アップロードした入力データがあるディレクトリに移動します。



AJF ファイルを開き、パラメータをもう一度確認します。

```
vi TrpCage-FMO-HF.ajf
```

以下、計算機ごとに変更する必要があるパラメータを示します。これらは計算を流す前によく確認し、必要に応じて変更します。

```
&CNTRL
...
Memory=1800    ← 1 プロセスあたりの使用メモリ上限
...
ReadGeom=' TrpCage.pdb' ← PDB ファイルへのパス（計算機によっては絶対パスで指定しないとエラーになる）
WriteGeom=' TrpCage-FMO-HF.cpf'
...
/

&FMOCNTRL
...
NP=1    ← 1 フラグメントを計算するのに使用するプロセス数
...
```

また、BioStaion Viewer と ABINIT-MP のバージョンが異なる場合や、ユーザーが手作業で AJF ファイルを作成した場合には、AJF ファイルのネームリストの順番等を整形する python スクリプトを使用することが可能です。

使用する ABINIT-MP のバージョン毎に python スクリプトが用意されています。今回は ABINIT-MP Open1 Rev.15 を使用しますので、AJF ファイルの整形を以下のとおり実施してから計算を行って下さい。

```
python3.0 ~/bin/mkinp_openver1rev15.py < TrpCage-FMO-MP2.ajf > TrpCage-FMO-MP2.ajf2
```

計算機によっては pjsub などのコマンドを使用してジョブを投入する必要があります。RIST FX10 の場合は、以下のような pjsub 用のスクリプト(run_smp.sh)を使用します。

```
#!/bin/sh

#----- pjsub option -----#
#PJM -L "node=1"                # 使用するノード数(FX10 では、1 ノードあたり 16 コア)
#PJM --mpi "proc=4"             # 使用する全プロセス数 (プロセス数=使用コア数/スレッド数)
#PJM -L "elapse=01:00:00"       # 総計算時間の指定(FX10 の会話ジョブは最大 6 時間)
#PJM -j

. /work/system/Env_base        #環境変数の設定
export OMP_NUM_THREADS=1       #スレッドの指定 (1 プロセスあたりに使用するコア数)

#----- ABINIT-MP option -----#
ABINIT_DIR=/home/acourse/abinit01/bin/  # ABINIT-MP バイナリのあるディレクトリを指定
                                         赤字部分を各位のアカウント番号に変更してください

FILE_NAME=TrpCage-FMO-HF        # ファイル名

AJF_NAME=$FILE_NAME.ajf2        # 入力ファイル
OUT_NAME=$FILE_NAME.log         # 出力ファイル

#----- Program execution -----#
mpiexec -stdin $AJF_NAME -of $OUT_NAME $ABINIT_DIR/abinitmp_smp  # FMO 計算実行
```

上記のスクリプトは以下に置かれています。赤字部分は各位のアカウント番号となります。

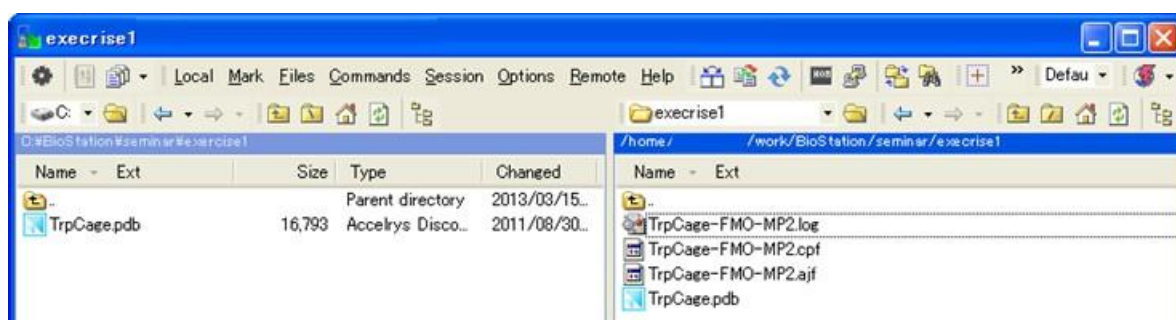
```
/home/acourse/abinit01/ABINIT-MP_HANDSON_20190327/run_mpi.sh
```

上記のスクリプトを pjsub コマンドでサブミットします。

```
pjsub --interact run_mpi.sh
```

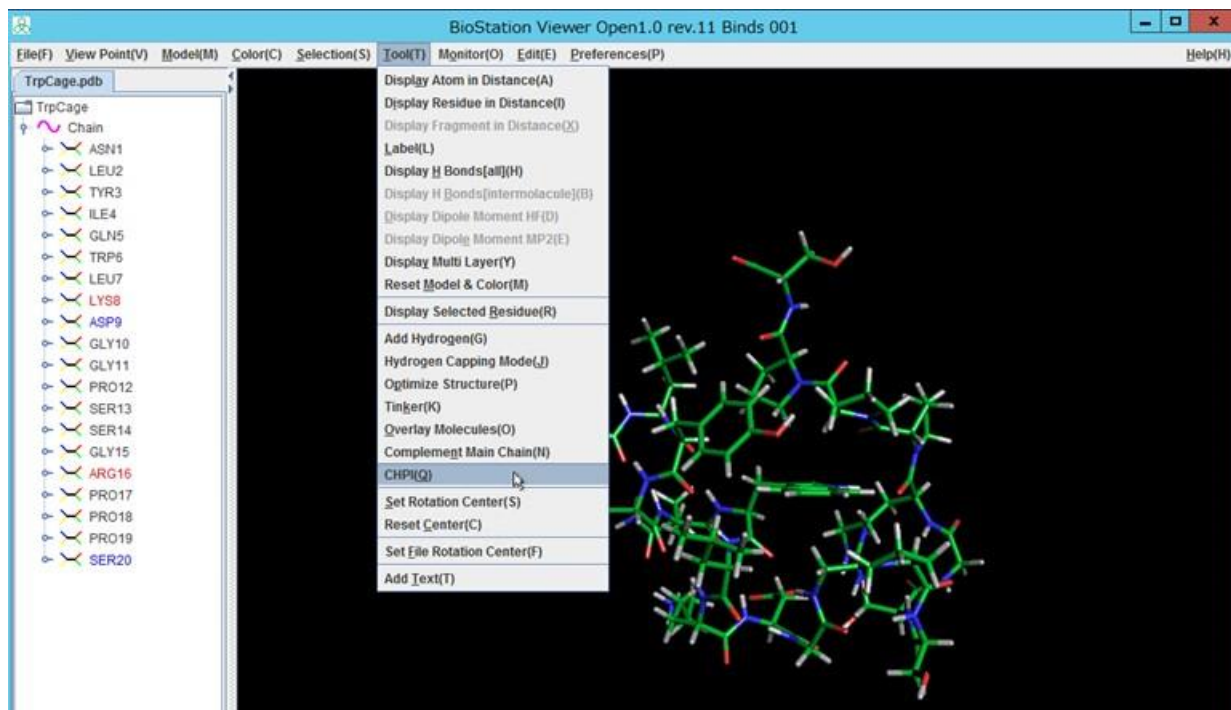
ジョブの投入方法は計算機ごとに異なりますので、詳しい使い方は各計算機のマニュアルを参照するか管理者にお問い合わせください。

計算が完了すると TrpCage-FMO-HF.log と TrpCage-FMO-HF.cpf が出力されるのでこれらのファイルをダウンロードします。

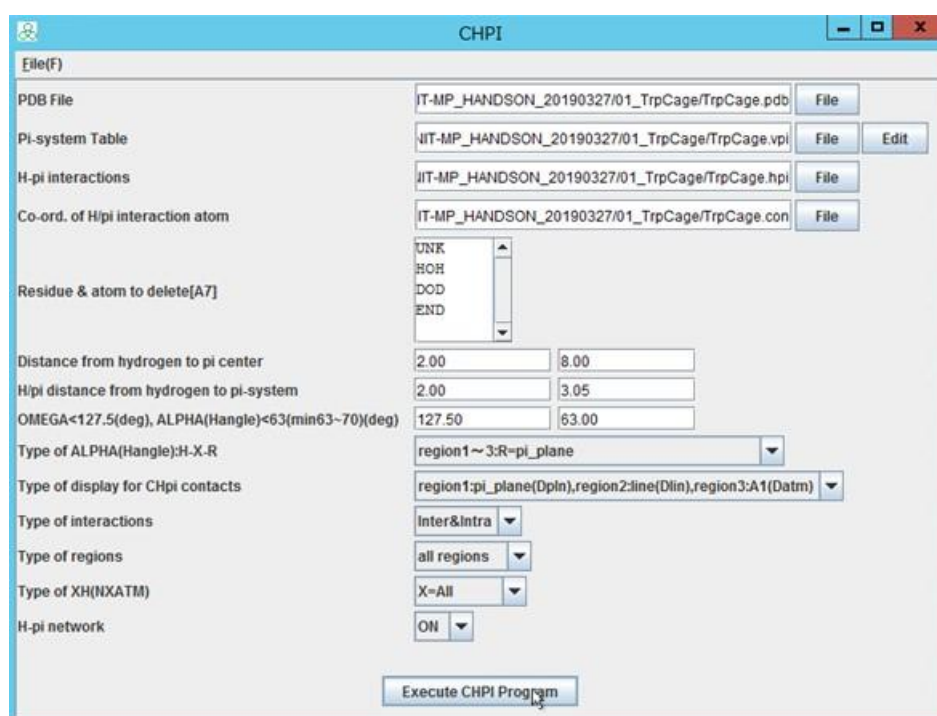


1.3 CHPI 解析

CHPI 解析を行うには Tool→CHPI を実行します。CHPI プログラムは西尾先生、梅沢先生が開発された CH/ π 相互作用を解析するためのプログラムです。



Tool→CHPI を実行すると CHPI 解析のパラメータ設定を行うウィンドウが起動します。通常は変更する必要がないためここではデフォルト設定のまま **Execute CHPI Program** ボタンをクリックします。各パラメータの詳細については書籍「有機化学のための分子間力入門、西尾元宏、講談社サイエンティフィック」をご参照ください。



CHPI	
File(F)	
PDB File	IT-MP_HANDSON_20190327/01_TrpCage/TrpCage.pdb [File]
Pi-system Table	IT-MP_HANDSON_20190327/01_TrpCage/TrpCage.vpi [File] [Edit]
H-pi interactions	IT-MP_HANDSON_20190327/01_TrpCage/TrpCage.hpi [File]
Co-ord. of H/pi interaction atom	IT-MP_HANDSON_20190327/01_TrpCage/TrpCage.con [File]
Residue & atom to delete[A7]	UNK HOH DOD END
Distance from hydrogen to pi center	2.00 8.00
H/pi distance from hydrogen to pi-system	2.00 3.05
OMEGA<127.5(deg), ALPHA(Hangle)<63(min63~70)(deg)	127.50 63.00
Type of ALPHA(Hangle):H-X-R	region1~3:R-pi_plane
Type of display for CHpi contacts	region1:pi_plane(Dpin),region2:line(Dlin),region3:A1(Datm)
Type of interactions	Inter&Intra
Type of regions	all regions
Type of XH(NXATM)	X=All
H-pi network	ON
Execute CHPI Program	

DOS プロンプトが起動し、CHPI プログラムの出力が表示されます。Please input exit と表示されれば完了です。exit コマンドを実行するか、ウィンドウの×ボタンをクリックして DOS プロンプトを閉じます。

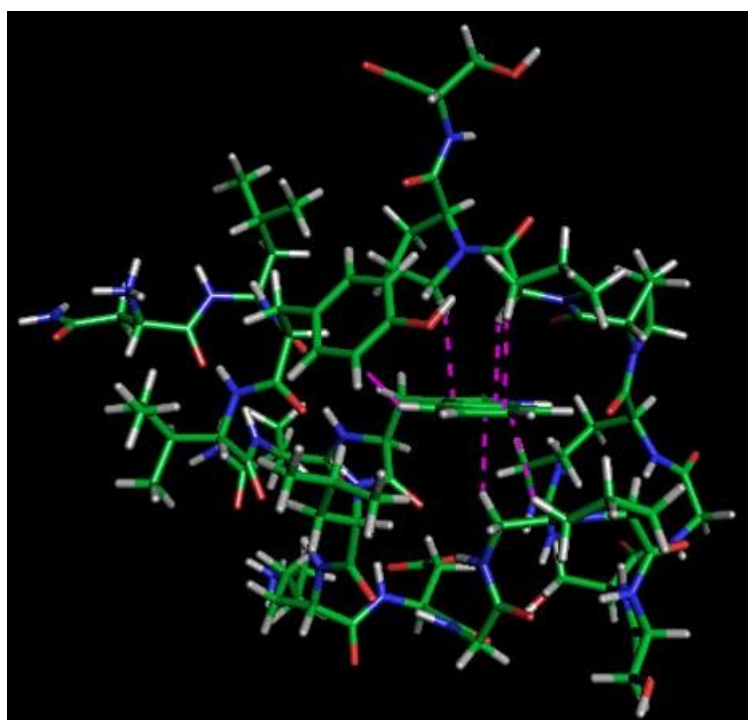
```
C:\Windows\system32\cmd.exe
0:R=pi_plane(all regions)
1:R=pi_plane(region-1),R=line(region-2),R=C1(region-3)
2:R=C1(all regions)
:0
Type of display? 0:(pi_plane,line,C1) 1:C1, 2:pi_center
:0
Type of interactions? 0:intra&inter 1:inter 2:intra
:0
Type of regions?
0:all regions 1:region-1 2:region-2 3:region-3
:0
Type of XH? 0:ALL 1:C, 2:N, 3:O, 4:S, 5:except C
:0
H-pi network? 0:off 1:on
:1

**Inter&Intramolecular H/pi interactions**

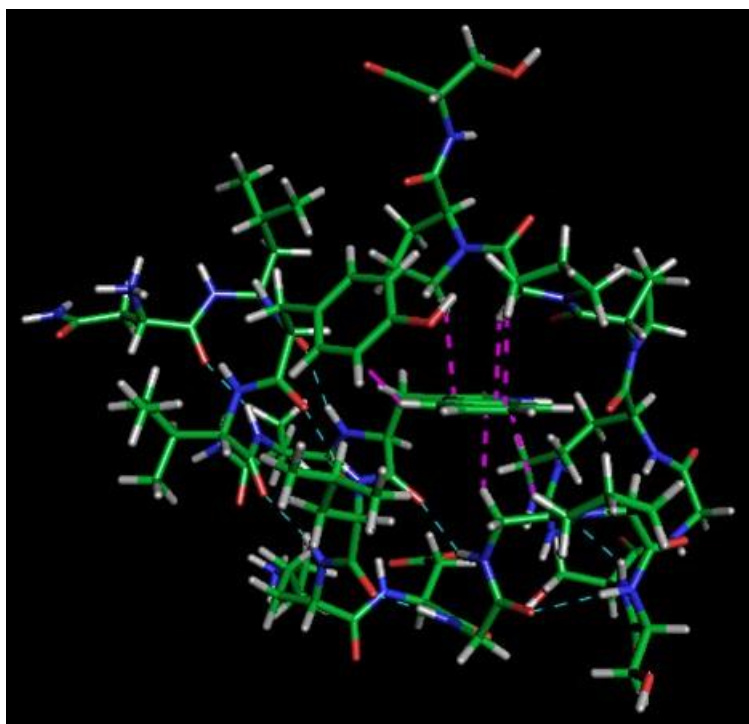
Num of H/pi interactions : 6 (region_1: 3, region_2: 2, region_3: 1)

Please type 'exit', and press 'Enter.'
C:\Program Files\ABINIT-MP Open Consortium\BioStationViewer>
```

以下のように CH/ π 相互作用がある箇所に紫の破線が表示されます。



さらに、水素結合を表示してみます。Tool → Display H Bonds [all] を選択すると、水素結合の位置に水色の破線が表示されます。



以上の CHPI 表示や水素結合表示は、PDB 立体構造から相互作用の候補位置を検索するツールです。実際に相互作用があるかどうかは、IFIE 解析をすることで定量的に解析することができます。

※ CHPI プログラムを使う際の注意点

PDB ファイルの書式が整っていない場合には CHPI プログラムはエラーとなります。そのような場合には、File → Save Renumbered PDB File を選択し、書式の整った PDB ファイルを出力したのち、新しく作成した PDB ファイルを用いて再度実行してください。

1.4 IFIE 解析

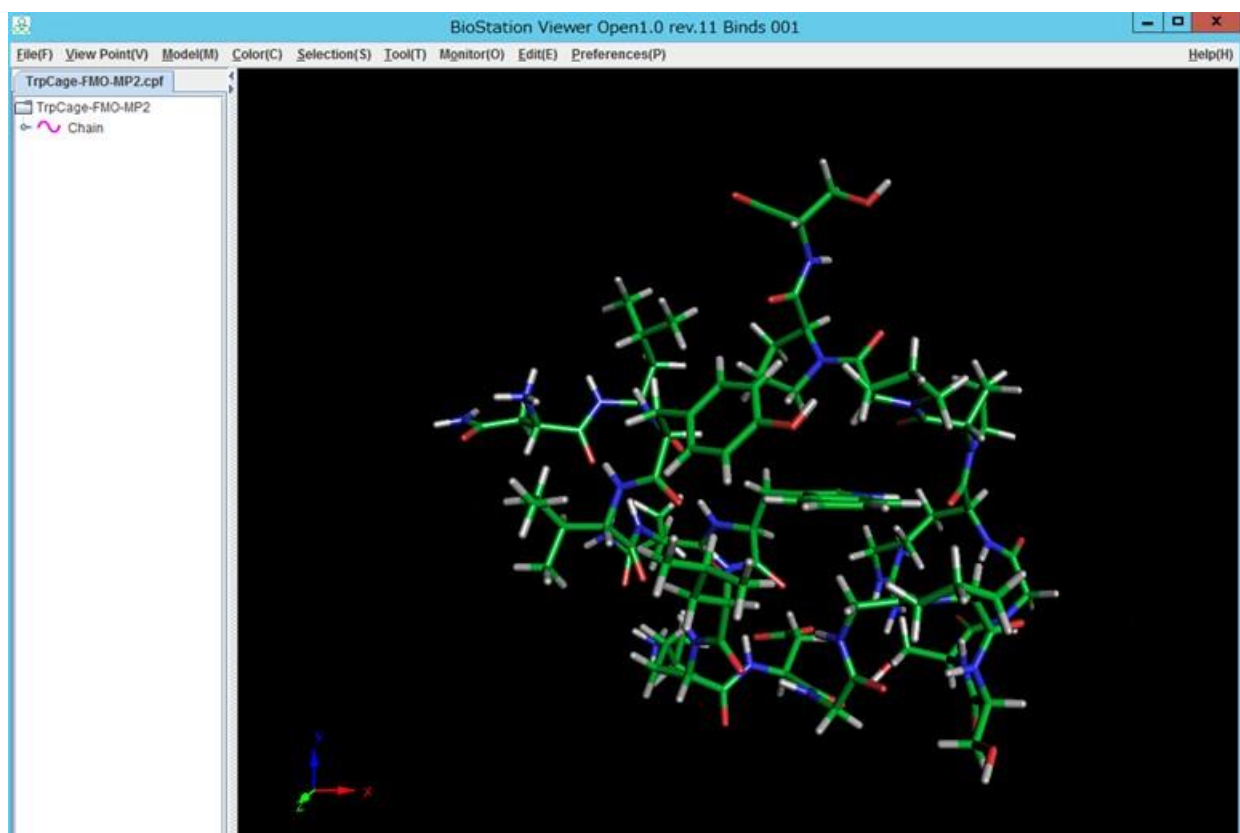
IFIE 解析を行います。まず、CPF ファイルを BioStation Viewer の画面にドラッグ&ドロップして読み込みます。実際に計算を流した HF/STO-3G の計算結果のファイルは、デスクトップの以下のフォルダにおいてあります。

`ABINIT-MP_HANDSON_20190327\01_TrpCage\reference\TrpCage-FMO-HF.cpf`

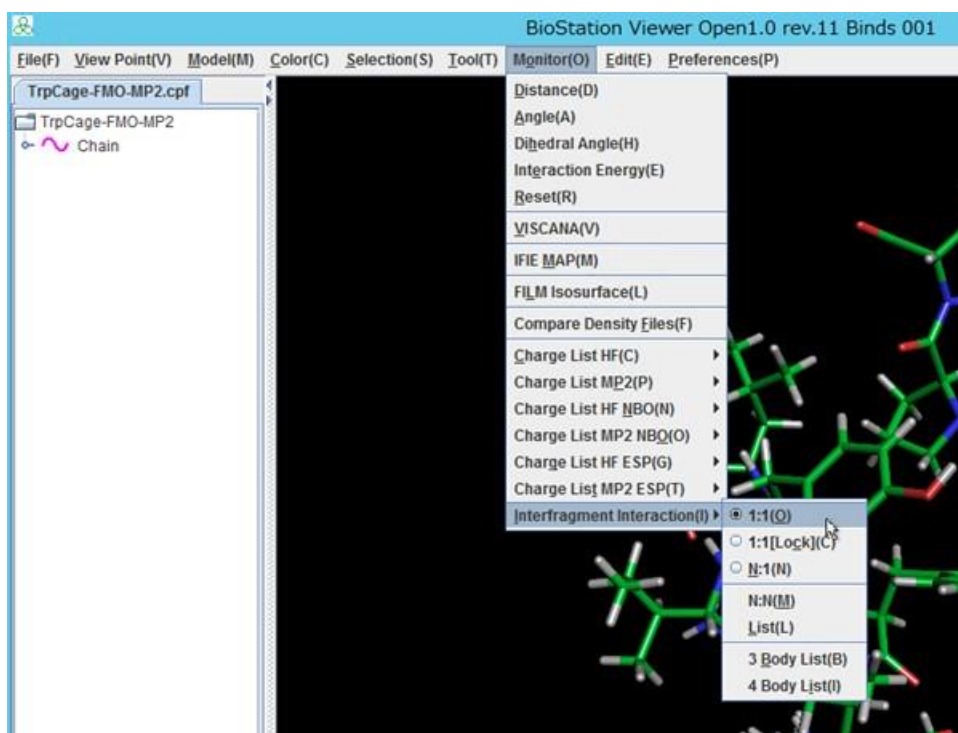
MP2/6-31G の結果についても、計算結果ファイルがデスクトップの以下フォルダに置いてあります。

`ABINIT-MP_HANDSON_20190327\01_TrpCage\reference\TrpCage-FMO-MP2.cpf`

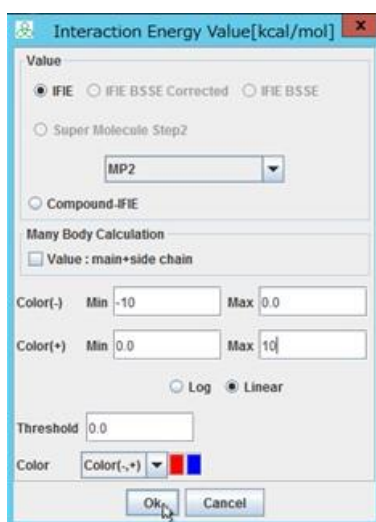
Model→Atom→Stick、Model→Structure(s)→Off とすることで下図の様に分子構造が表示できます。



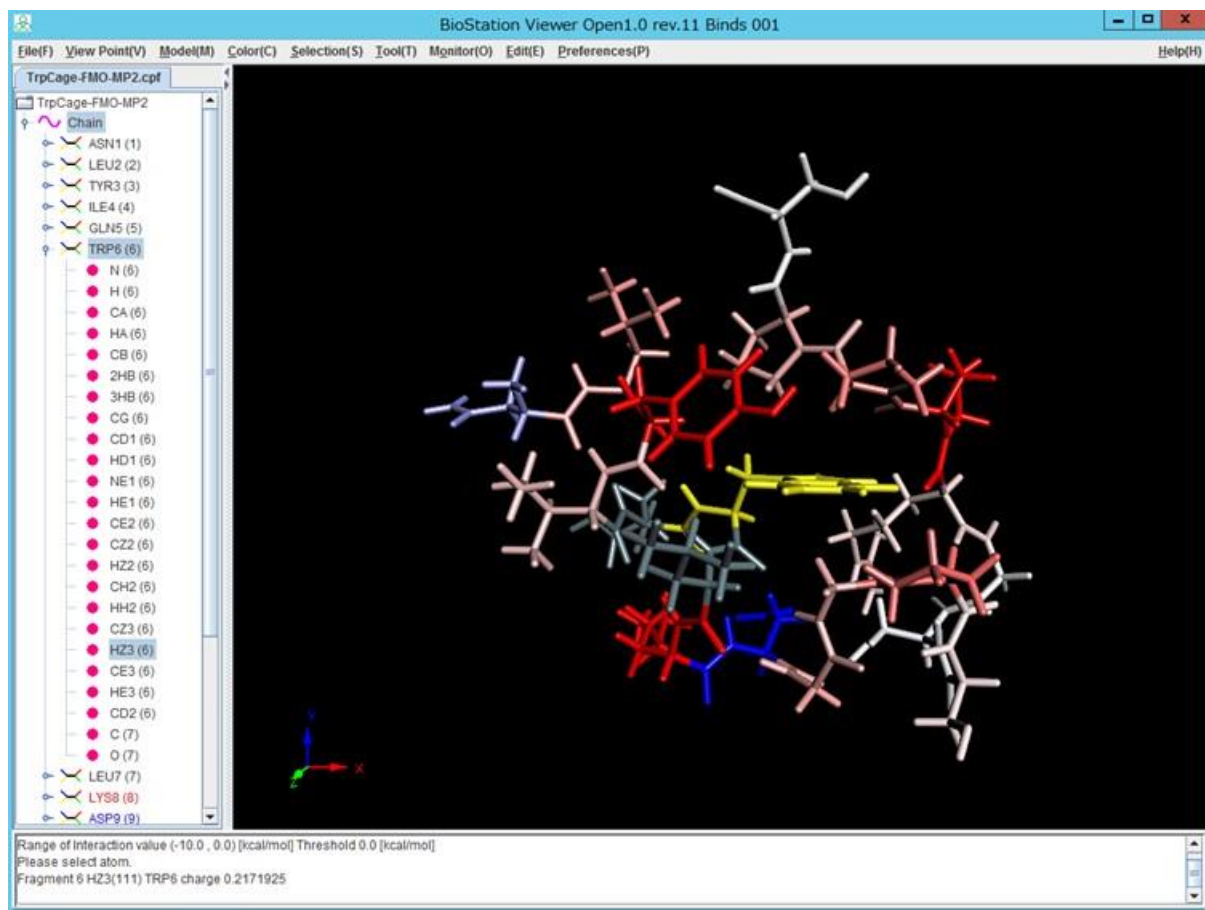
Monitor→Interfragment Interaction→1:1 を実行します。ここで IFIE 解析には 1:1、N:1、N:N などの選択肢がありますが、1:1 はフラグメント 1 対フラグメント 1 つの IFIE を解析します。注目する残基 (リガンドなど) が複数のフラグメントに分割されている場合は N:1 などを使うことがあります。タンパク質対 DNA の IFIE を見る場合などに N:N を使うことがあります。



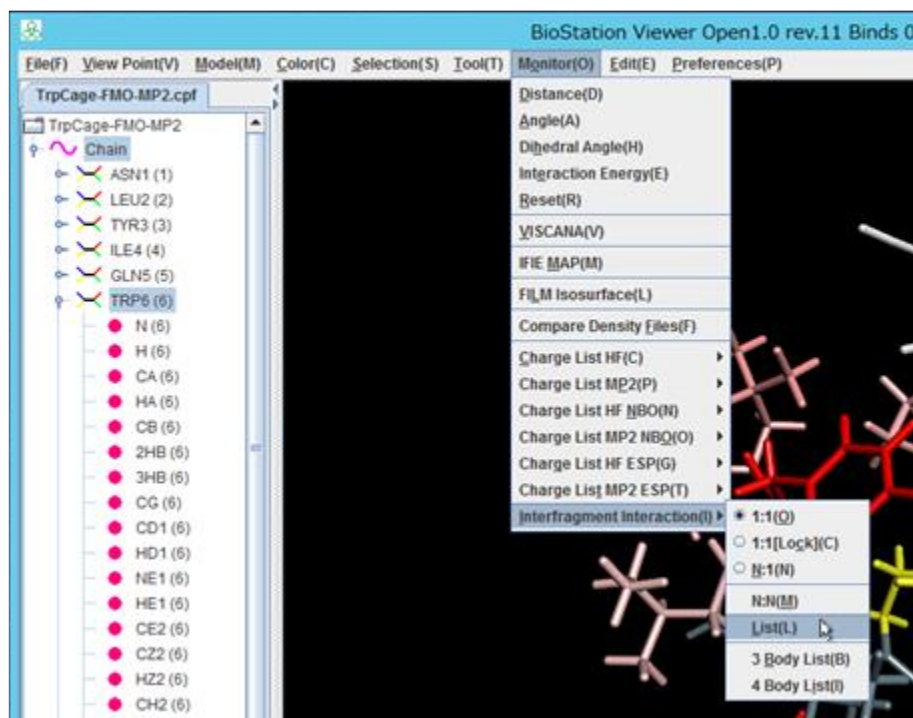
IFIE 解析のパラメータ設定ウィンドウが起動します。本課題では FMO-MP2 計算を行ったので Value のプルダウンは MP2 を選択します。Color のテキストボックスでは IFIE を青赤のグラデーションで表示する際の上限值と下限値を指定します。ここでは -10 から 10 とします。



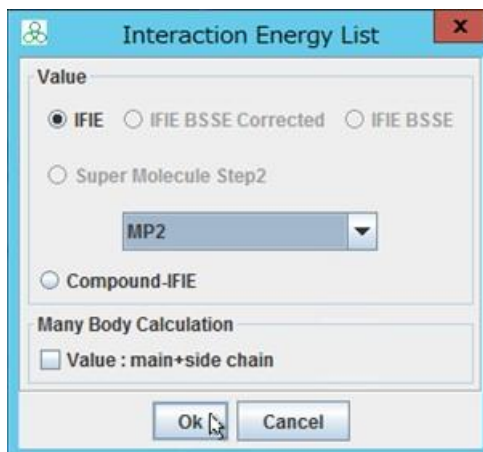
OK ボタンをクリックし、IFIE 解析したいフラグメントをクリックすると、そのフラグメントとの IFIE に基づいて周囲のフラグメントが色付けされます。黄色は現在注目しているフラグメント、赤は注目しているフラグメントと引力的な相互作用をしているフラグメント、青は注目しているフラグメントと斥力的な相互作用をしているフラグメント、灰色はそのフラグメントと共有結合しているため IFIE を評価できないフラグメントを意味します。



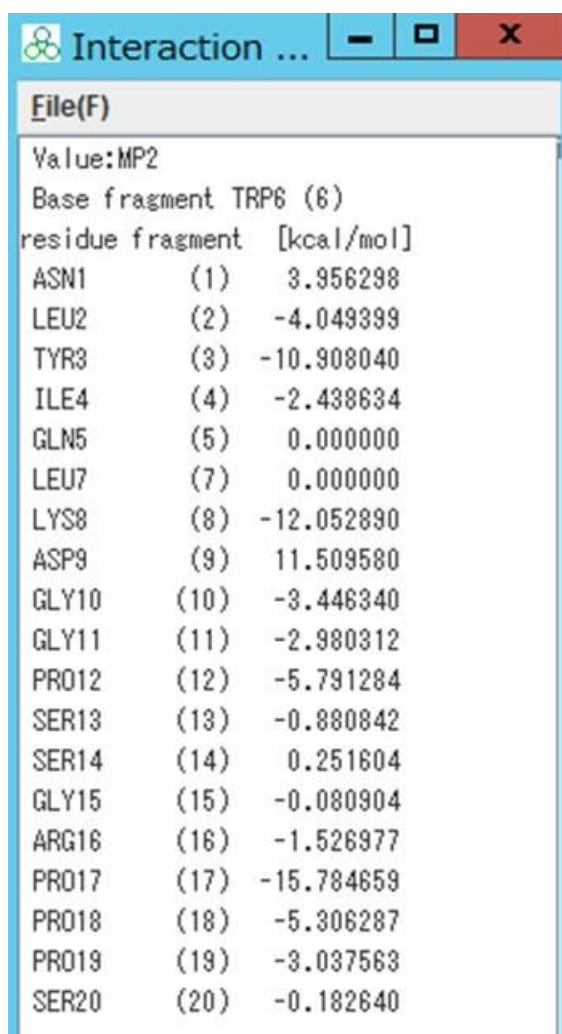
IFIE の数値を表示する場合には Monitor→Interfragment Interaction→List を実行します。



本課題では MP2 レベルの IFIE を表示します。

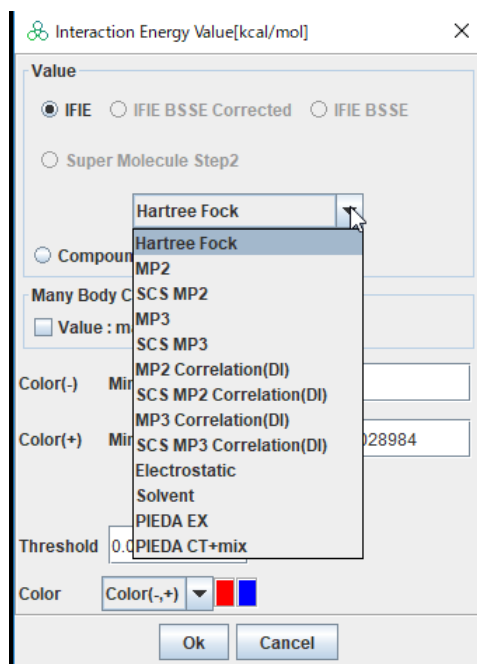


Monitor→Interfragment Interaction→List を実行すると注目しているフラグメントとその周囲のフラグメント間の IFIE が kcal/mol 単位で表示されます。



Value:MP2		
Base fragment TRP6 (6)		
residue	fragment	[kcal/mol]
ASN1	(1)	3.956298
LEU2	(2)	-4.049399
TYR3	(3)	-10.908040
ILE4	(4)	-2.438634
GLN5	(5)	0.000000
LEU7	(7)	0.000000
LYS8	(8)	-12.052890
ASP9	(9)	11.509580
GLY10	(10)	-3.446340
GLY11	(11)	-2.980312
PRO12	(12)	-5.791284
SER13	(13)	-0.880842
SER14	(14)	0.251604
GLY15	(15)	-0.080904
ARG16	(16)	-1.526977
PRO17	(17)	-15.784659
PRO18	(18)	-5.306287
PRO19	(19)	-3.037563
SER20	(20)	-0.182640

PIEDA の各成分を表示する場合には、Electrostatic (ES 項)、PIEDA EX (EX 項)、PIEDA CT+mix (CT 項)、MP2 Correlation (DI) (DI 項) をそれぞれ選択することで、相互作用のカラー表示やエネルギーリストを得ることができます。



2 課題 2 : エストロゲン受容体ーリガンド複合体の IFIE 解析

本課題ではエストロゲン受容体 (EST) とそのリガンド (エストロゲン; EST) 複合体の FMO-MP2 計算を行い、IFIE 解析を行います。ここで構造は 50 残基にモデル化した構造を用います。

※今回の演習では、計算時間の短縮のため、HF/STO-3G で計算を流します。

2.1 入力データ作成

課題 2 では課題 1 で作成した AJF ファイルを編集して入力データを作成します。まず、課題 1 で作成した AJF ファイルを別名でコピーします。

```
cp TrpCage-FMO-HF.ajf2 ER_50res-FMO-HF.ajf2
```

ER_50res-FMO-HF.ajf を編集します。

```
vi ER_50res-FMO-HF.ajf2
```

CNTRL ネームリストの ReadGeom キーワードと WriteGeom キーワードを変更します。

```
&CNTRL
...
ReadGeom='ER_50res.pdb'
WriteGeom='ER_50res-FMO-HF.cpf'
...
/
```

以上で入力データの作成は完了です。

2.2 FMO 計算

FMO 計算の実行方法は課題 1 の時と同じです。RIST の FX10 で流す場合には、sh スクリプトを編集して ajf ファイル名を変更した上で計算を流します。

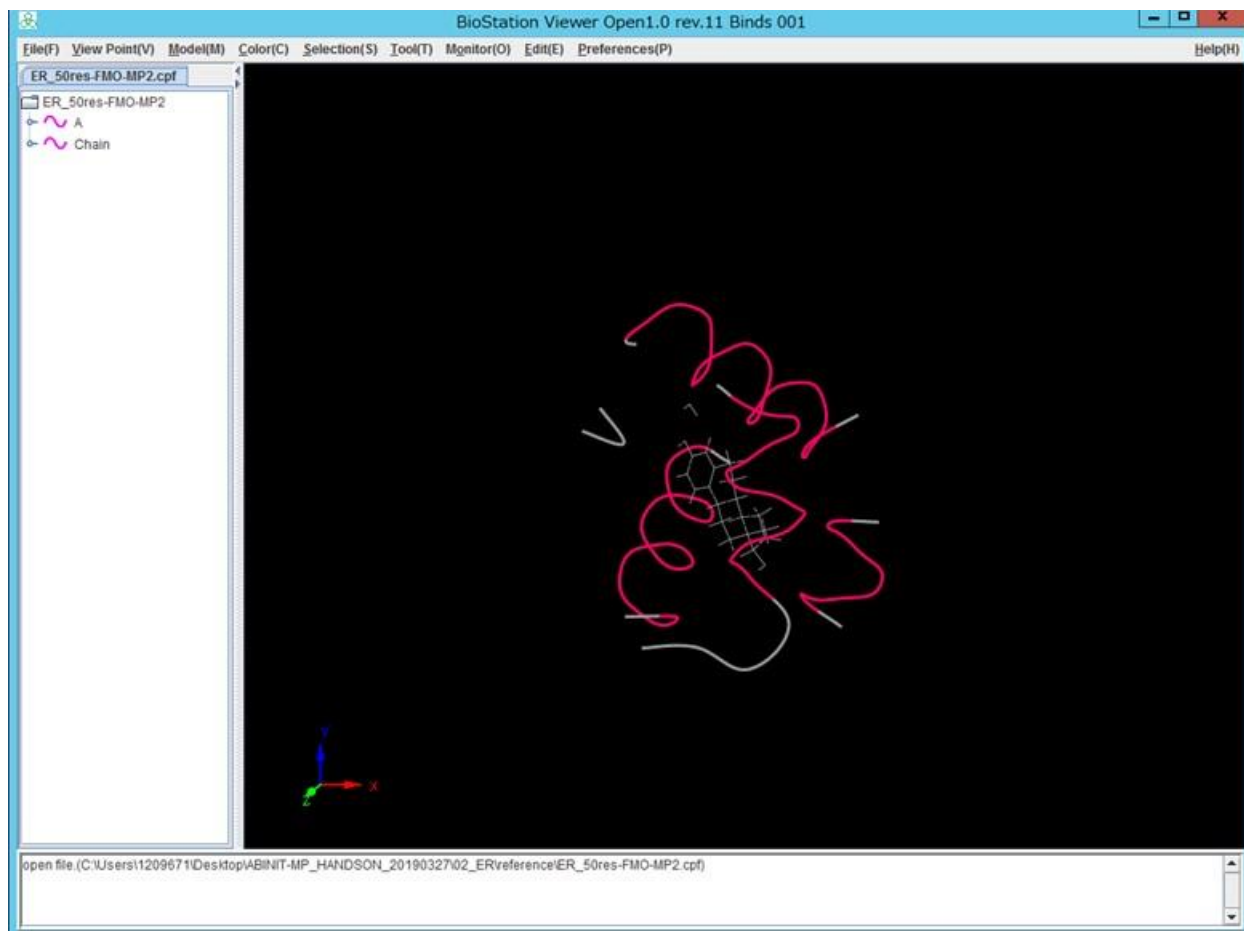
```
pjsub --interact run_mpi.sh
```

計算が完了すると ER_50res-FMO-HF.log と ER_50res-FMO-HF.cpf が出力されるのでこれらのファイルをダウンロードします。

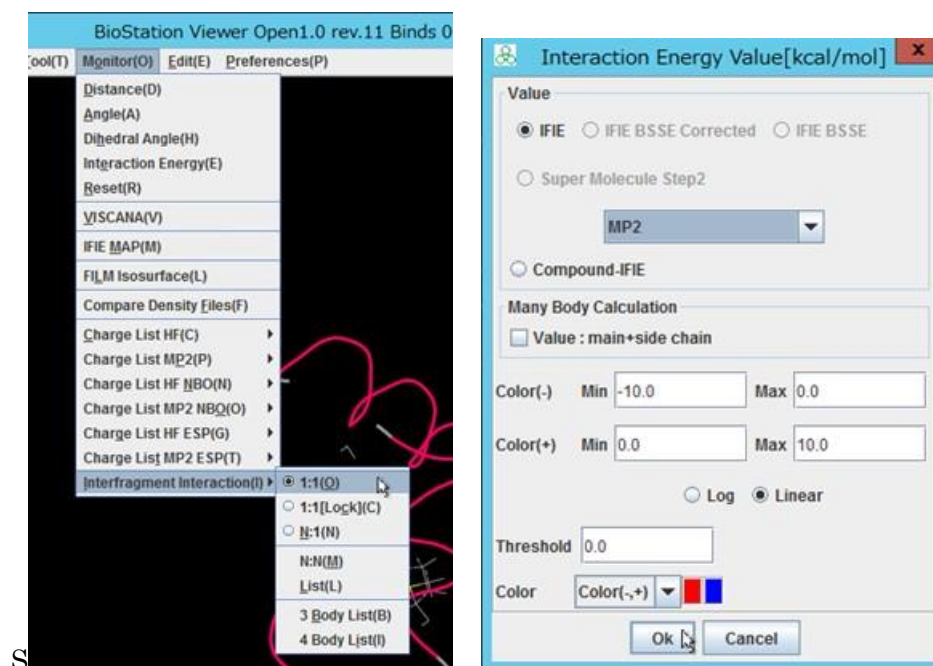
2.3 IFIE 解析

CPF ファイルを BioStation Viewer の画面にドラッグ&ドロップして読み込みます。ここでは、以下のフォルダに置いてある **事前に計算を行っておいた MP2 計算結果** を用います。

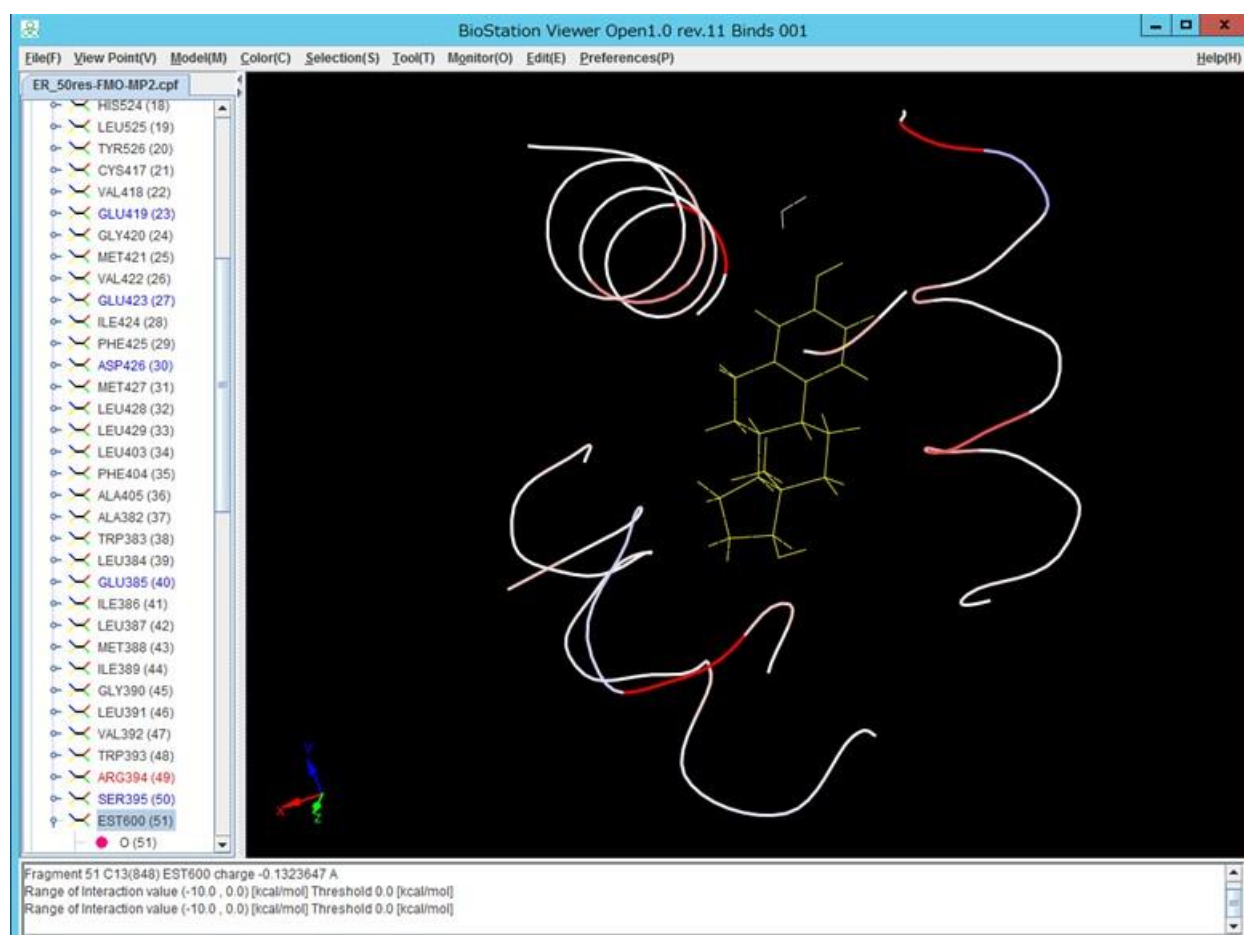
ABINIT-MP_HANDSON_20190327\02_ER\reference\ER_50res-FMO-MP2.cpf



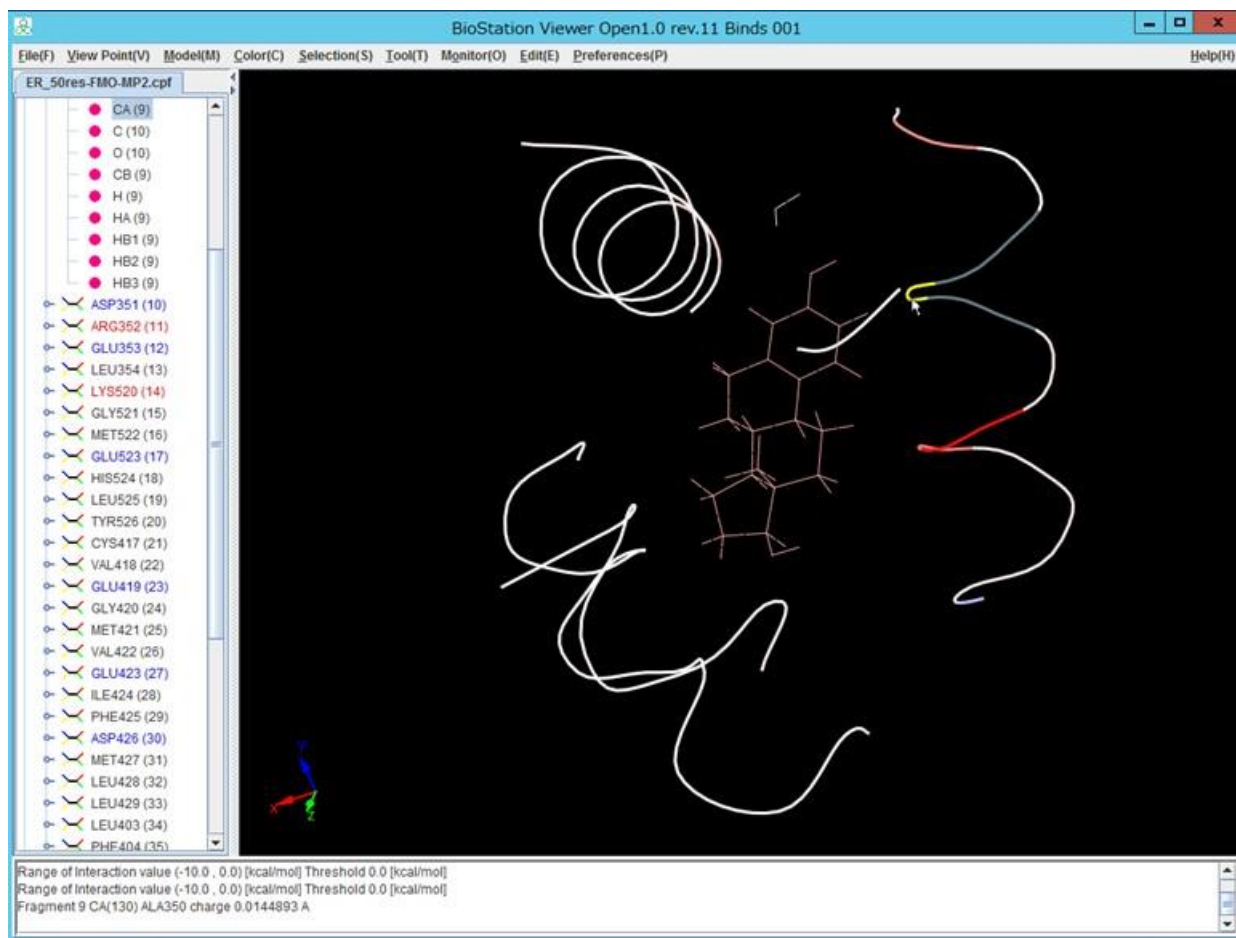
リガンドをクリックし、Monitor→Interfragment Interaction→1:1 を実行します。Value のプルダウンは MP2 を選択し、Color のテキストボックスでは -10 から 10 とします。



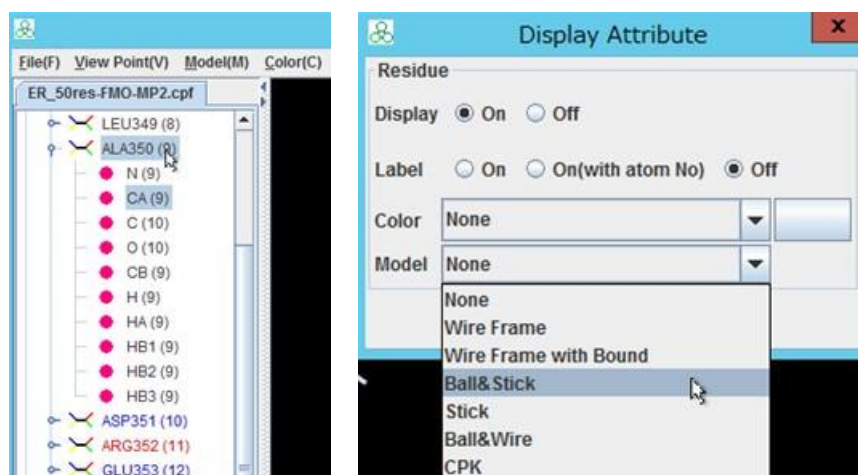
タンパク質の立体構造がリガンドとの IFIE に基づいて色付けされます。



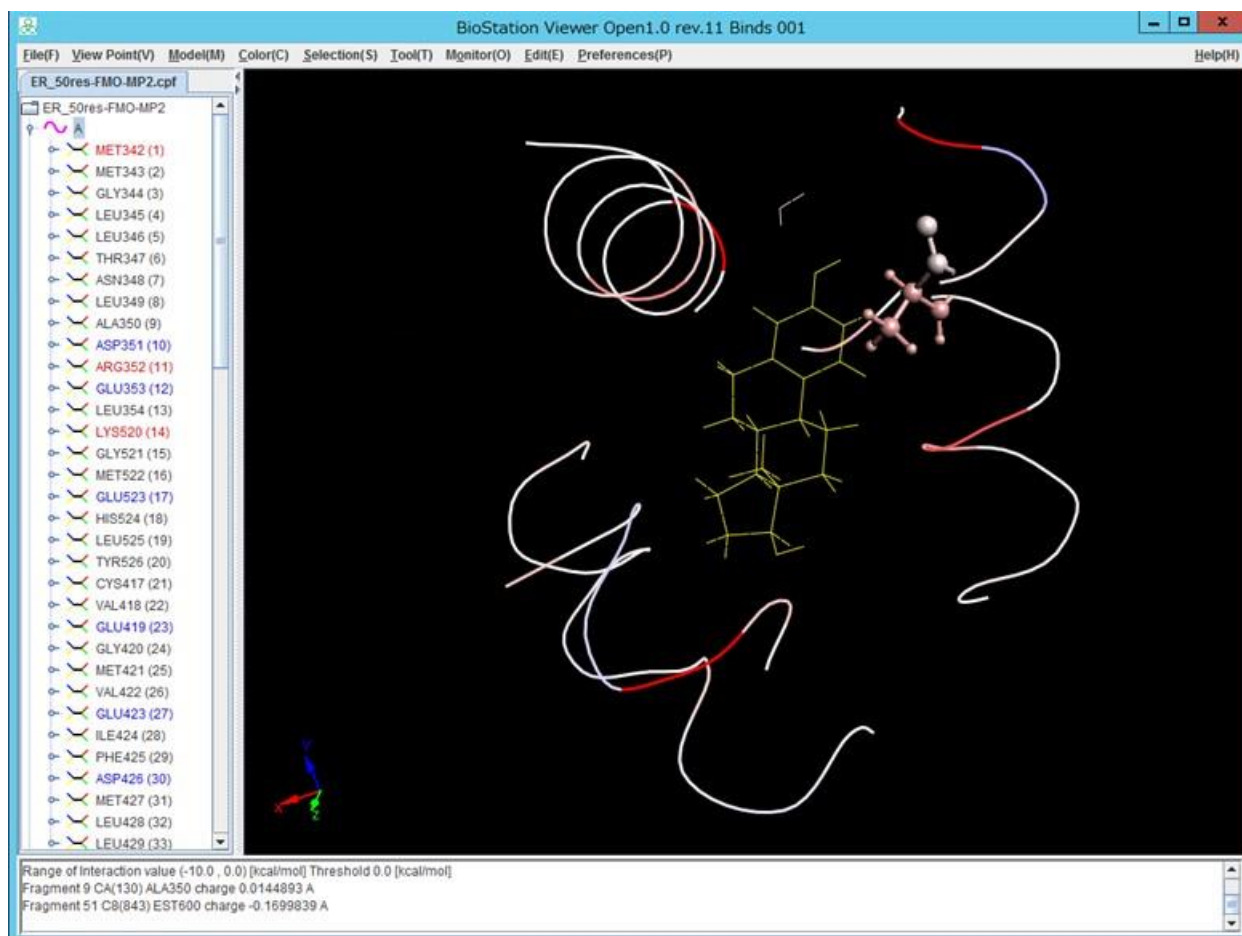
リガンドではなく、アミノ酸をクリックすれば、そのアミノ酸との IFIE に基づいて周囲の残基を色付けすることができます。ここでクリックした ALA350 を Ball&Stick 表示にしてみましょう。



画面左のツリー構造にて ALA350 を右クリックし、Display Attribute ウィンドウで Model を Ball&Stick にします。



再びリガンドに焦点を戻す(リガンドをクリックする)と、リガンドと ALA350 が相互作用する様子を分子構造と IFIE を照らし合わせて見ることができます。



また、Monitor→Interfragment Interaction→List を実行すると注目しているフラグメントとその周囲のフラグメント間の IFIE が kcal/mol 単位で表示されます。

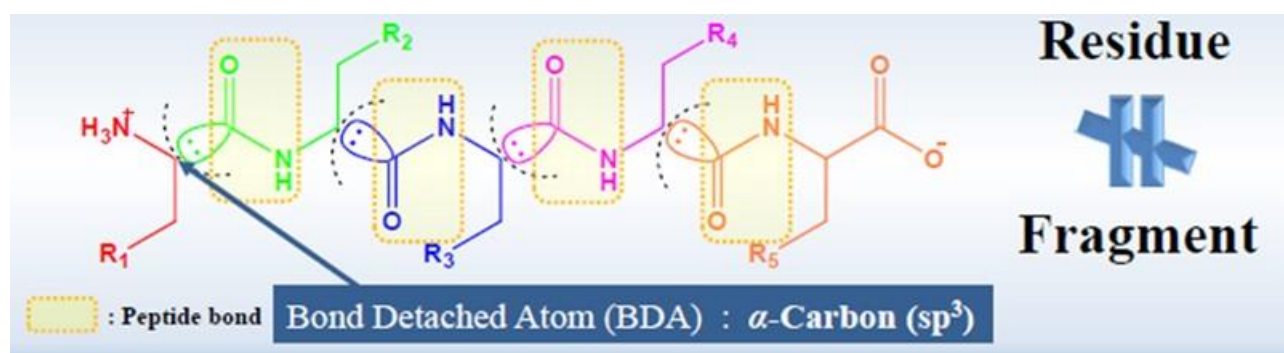
Interaction ...		
File(F)		
Value:MP2		
Base fragment EST600 (51)		
residue	fragment	[kcal/mol]
MET342	(1)	0.053758
MET343	(2)	-0.962543
GLY344	(3)	-0.640144
LEU345	(4)	-0.303180
LEU346	(5)	-3.667930
THR347	(6)	-6.625988
ASN348	(7)	-0.173007
LEU349	(8)	-0.628563
ALA350	(9)	-3.251051
ASP351	(10)	-0.938571
ARG352	(11)	3.006596
GLU353	(12)	-34.561676
LEU354	(13)	-0.553879
LYS520	(14)	-2.377891

3 課題 3 : ペプチド結合での切断

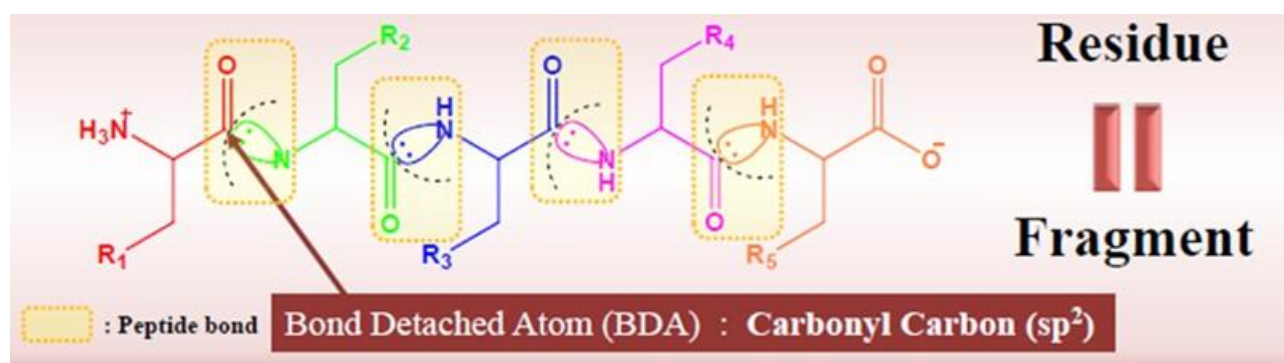
本課題では課題 2 で用いた ER50 残基モデルの FMO 計算において、フラグメント切断箇所を従来の sp^3 炭素から sp^2 炭素に変えた場合の解析を行います。まず、BioStation Viewer をもちいて sp^2 切断用の AJF ファイルを作成します。従来の sp^3 切断の結果については課題 2 のものを使用し、FMO 計算の結果を比較します。

※今回の演習では、計算時間の短縮のため、HF/STO-3G で計算を流します。

従来の sp^3 切断イメージ



sp^2 切断イメージ

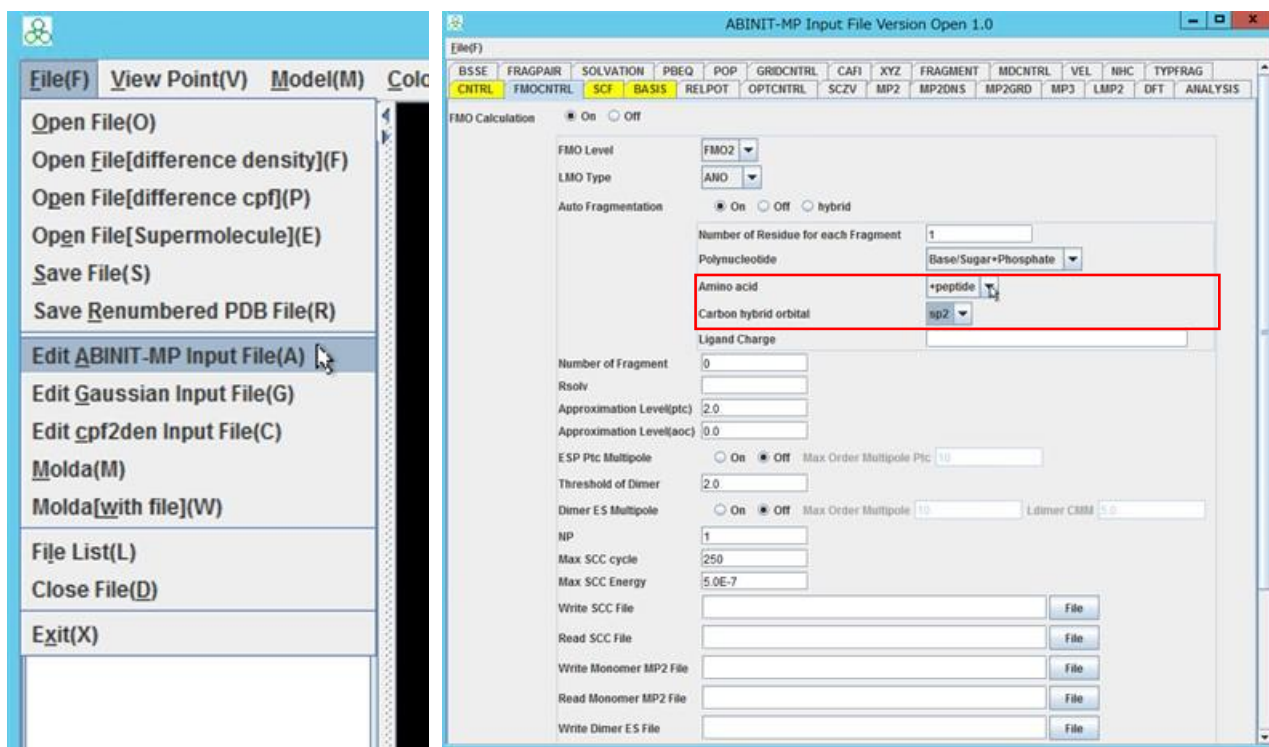


3.1 入力データ作成

以下の pdb ファイルを BioStation Viewer で開きます。

ABINIT-MP_HANDSON_20190327\03_sp2\ER_50res.pdb

File→Edit ABINIT-MP Input File から AJF ファイル作成画面を開きます。FMOCNTRL タブにある AutoFragmentation を”ON”にし、Amino acid を”+peptide”、Carbon hybrid orbital を “sp²” にします。この設定により、ABINIT-MP で sp² 自動分割が実施され、sp² 射影演算子を用いた計算を行うことができます。CNTRL タブの Write Geometry File を “ER_50res_sp2.cpf” とした上で、AJF 作成画面の File→Save File から “ER_50res_sp2.ajf” としてファイルを保存して下さい。



sp³切断の場合には、FMOCNTRL タブの Amino acid を”+amino”、Carbon hybrid orbital を “sp³” とすればよいですが、ここでは課題 2 の結果を使用するのでファイルを作成する必要はありません。なお、今回の演習では、計算時間の短縮のため、HF/STO-3G レベルの計算を流します。まず、ER_50res_sp2.ajf をコピーして、ER_50res_mp2_HF.ajf を作成します。テキストエディタで開き、&CNTRL ネームリストの Method='MP2'のところを、Method='HF'に変更します。WriteGeom も'ER_50res_sp2_HF.ajf' にしておきましょう。さらに、&BASIS ネームリストの 'BasisSet='6-31G' を'BasisSet=STO-3G'に変更します。

その後、python スクリプトを用いて “ER_50res_sp2_HF.ajf2” へと整形してください。

3.2 FMO 計算

FMO 計算の実行方法は課題 1 の時と同じです。RIST の FX10 で流す場合には、sh スクリプトを編集して ajf ファイル名を変更した上で計算を流します。

```
pjsub --interact run_mpi.sh
```

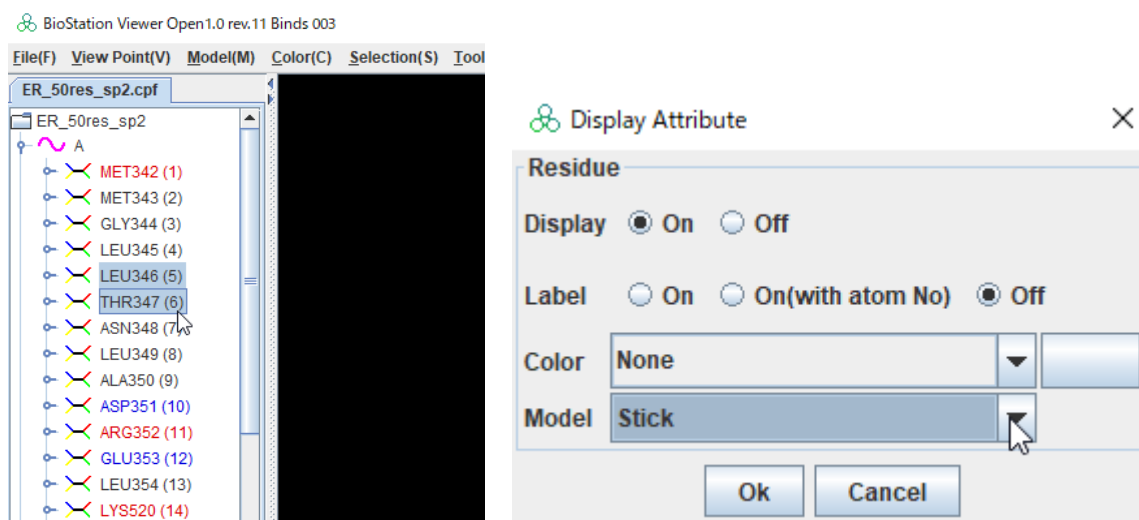
計算が完了したら ER_50res_sp2_HF.cpf をダウンロードします。

3.3 IFIE 解析

CPF ファイルを BioStation Viewer の画面にドラッグ&ドロップして読み込みます。ここでは、以下のフォルダに置いてある **事前に計算を行っておいだ MP2 計算結果** を用います。

```
ABINIT-MP_HANDSON_20190327\03_sp2\reference\ER_50res_sp2.cpf
```

今後の解析のために、リガンド、LEU346、THR347 をスティック表示にしましょう。左側のツリー図から、対象を選択し、右クリックすると表示形式の変更画面が現れます。



IFIE の可視化には、Monitor→Interfragment Interaction→1:1 を実行します。Value のプルダウンは MP2 を選択し、Color のテキストボックスでは-10 から 10 とします。リガンドをクリックすると、リガンドに対する IFIE が可視化されます。さらに、Monitor→Interfragment Interaction→List から、リガンドに対する MP2 IFIE のリストを表示すると、以下のようになります。



同様の表示を課題 2 で用いた通常のフラグメント分割の結果に対して行くと、以下のようになります。(ファイルは ABINIT-MP_HANDSON_20190327¥02_ER¥reference¥ER_50res-FMO-MP2.cpf)



sp²分割を行うことによってアミノ酸毎にフラグメント化されるため、LEU346 と THR347 の IFIE が sp³分割した場合から変化し、LEU346 とリガンドの安定化が強くなっていることが分かります。

4 課題 4：軌道レベルでの電荷移動相互作用解析（CAFI 解析）

本課題では水の 3 量体を題材に、軌道レベルでの電荷移動相互作用解析の方法を説明します。

4.1 入力データの作成

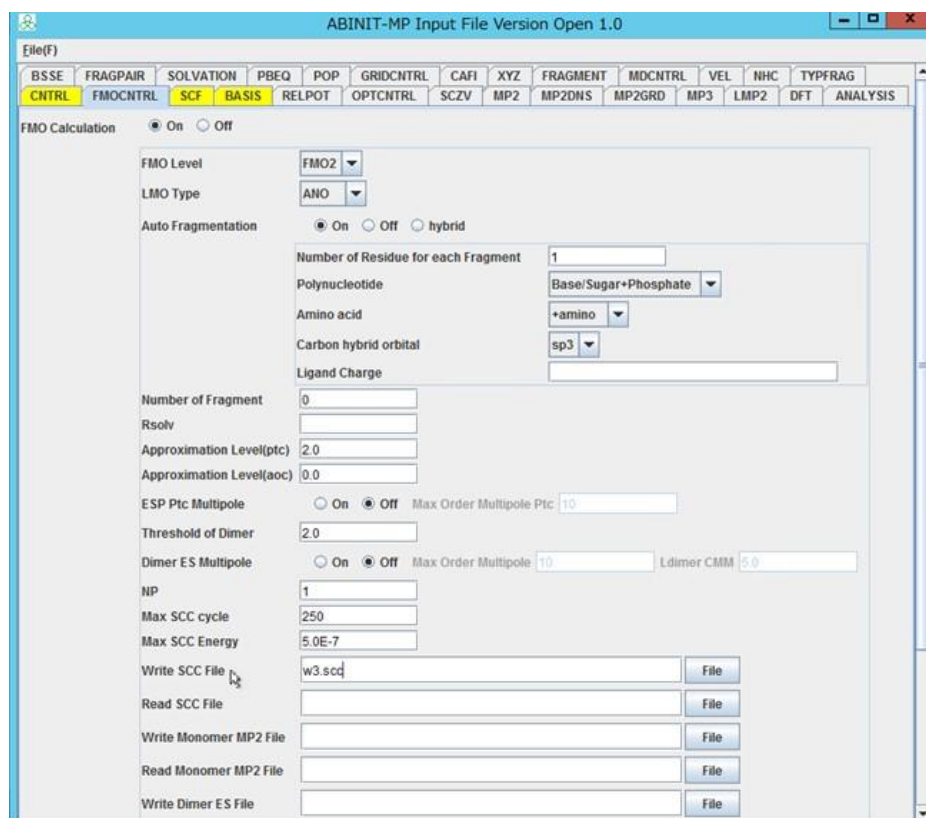
以下の pdb ファイルを BioStation Viewer で開きます。

ABINIT-MP_HANDSON_20190327\04_CAFI\w3.pdb

CAFI の計算は以下の 2 段階で実施します。

- ① 通常の FMO 計算を実施し、scc ファイル、cpf ファイルを作成する。(w3-1st.ajf)
- ② CAFI 計算を実施し、可視化用の pno ファイルを作成する。(w3-2nd.ajf)

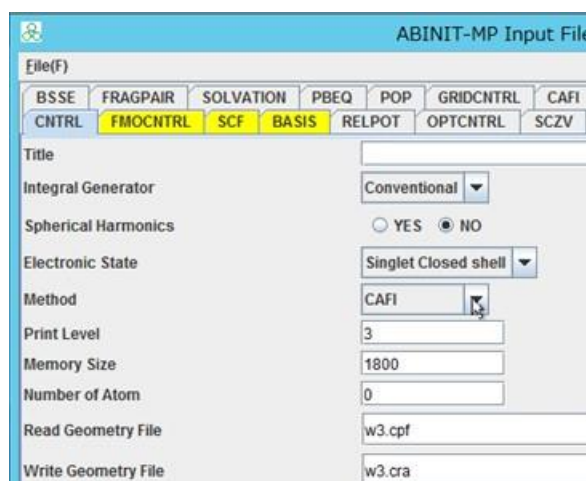
File→Edit ABINIT-MP Input File から AJF ファイル作成画面を開き、2 つの ajf ファイルを作成します。まずは、w3-1st.ajf です。CNTRL タブで Method を”Hartree Fock”とし、FMOCNTRL タブの Write SCC File にて “w3.scc” を指定してください。その上で、File→Save File から “w3-1st.ajf” として保存してください。



さらに、保存した “w3-1st.ajf” をテキストエディタ（メモ帳、ワードパッド等）で開き、&CNTRL に以下の様に CPFVER=4.2 を追記してください。

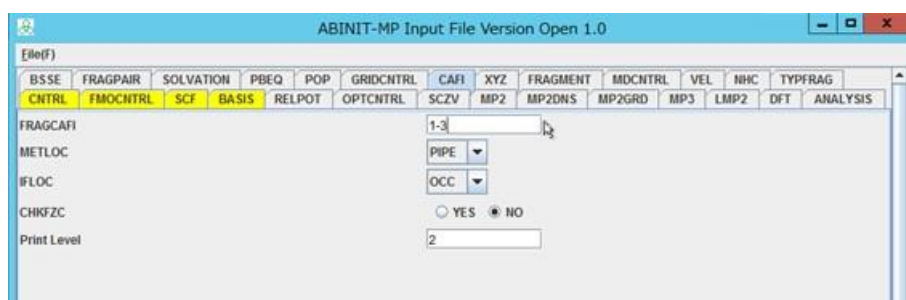
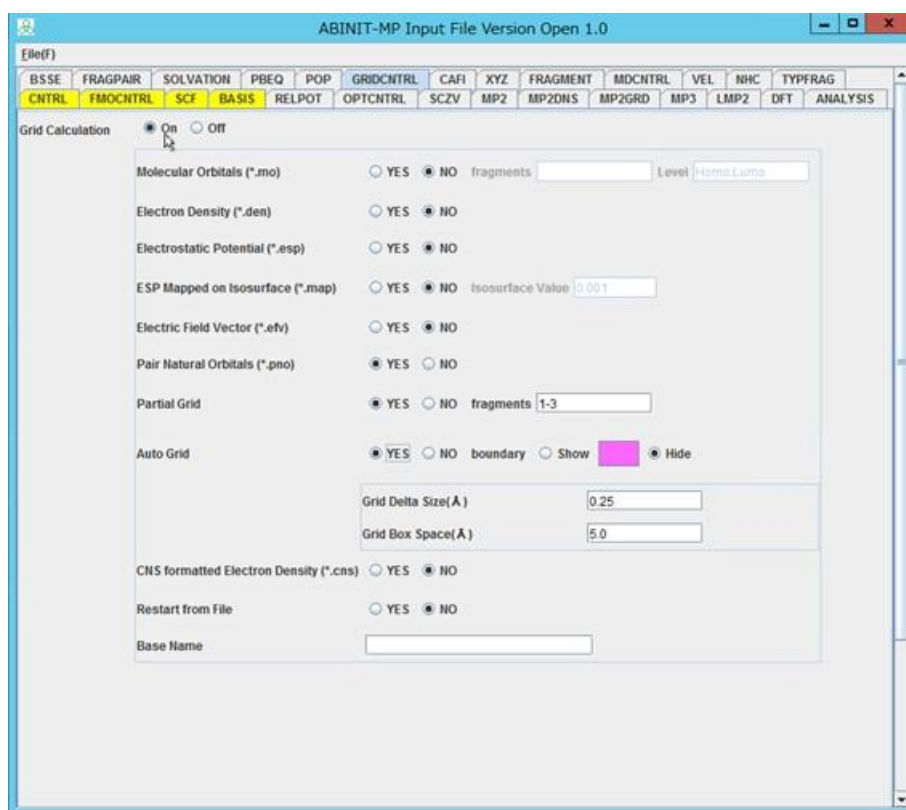
```
w3_1st.ajf
&CNTRL
ElecState='S1'
Method='HF'
Nprint=3
Memory=1800
Natom=0
ReadGeom='w3.pdb'
WriteGeom='w3.cpf'
Gradient='NO'
Vector='OFF'
CPFBIN='NO'
THOVL=1.0E-12
E_THSWZ=1.0E-12
G_THSWZ=1.0E-12
CPFVER=4.2
/
```

続いて、2 つ目の入力ファイル “w3_2nd.ajf” を作成します。上記と同様に、ABINIT-MP 入力ファイル作成画面を開き、CNTRL タブにて Method を “CAFI” としてください。さらに、Read Geomrtry File を “w3.cpf”、Write Geometry File を “w3.cra” としてください。また、FMOCNTRL タブで Write SCC File の “w3.scc” を削除し、Read SCC File に “w3.scc” を記入してください。



次に、GRIDCNTRL タブにて Grid Calculation を “On” とすると、次頁上図の様な画面になりますので、Pair Natural Orbitals(*.pno)を “YES” とし、Partial Grid も “YES” とした上で fragments を “1-3” と指定してください。さらに、CAFI タブにて FRAGCAFI を “1-3” と指定してください。（次頁下図）

作成した 2 つのファイルを FX10 に転送し、python スクリプトを用いて整形し、“w3-1st.ajf2” 及び “w3-2nd.ajf2” を作成してください。



4.2 FMO 計算

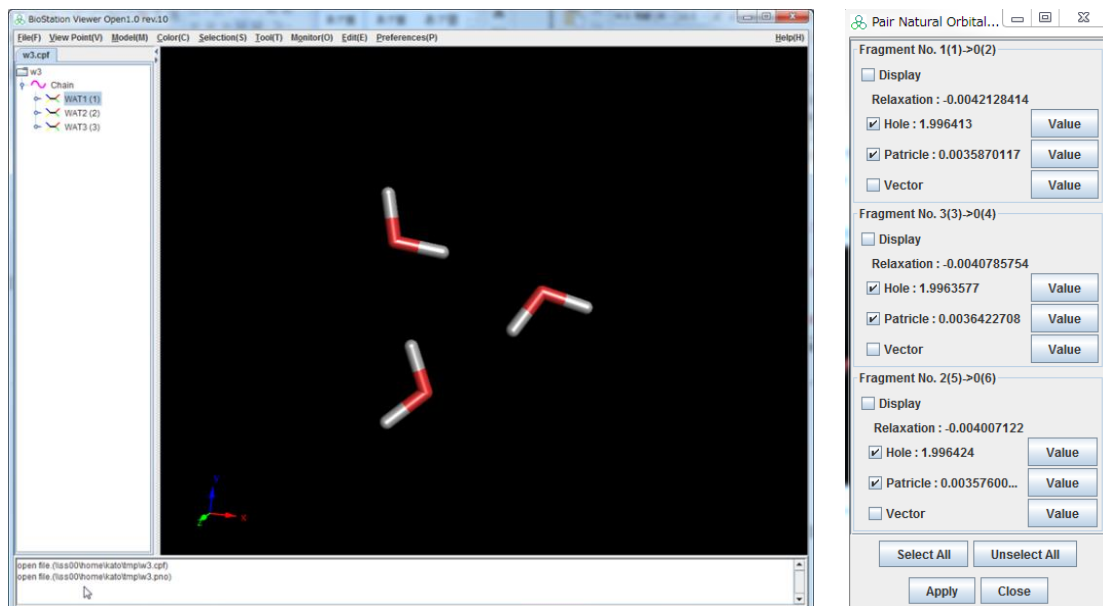
FMO 計算の実行方法は課題 1 の時と同じです。RIST の FX10 で流す場合には、sh スクリプトを編集して ajf ファイル名を変更した上で計算を流します。なお、本課題の系は 3 フラグメントとなるため、run_mpi.sh 内で#PJM -mpi “proc=3”とした上で計算を行ってください。

```
pjsub --interact run_mpi.sh
```

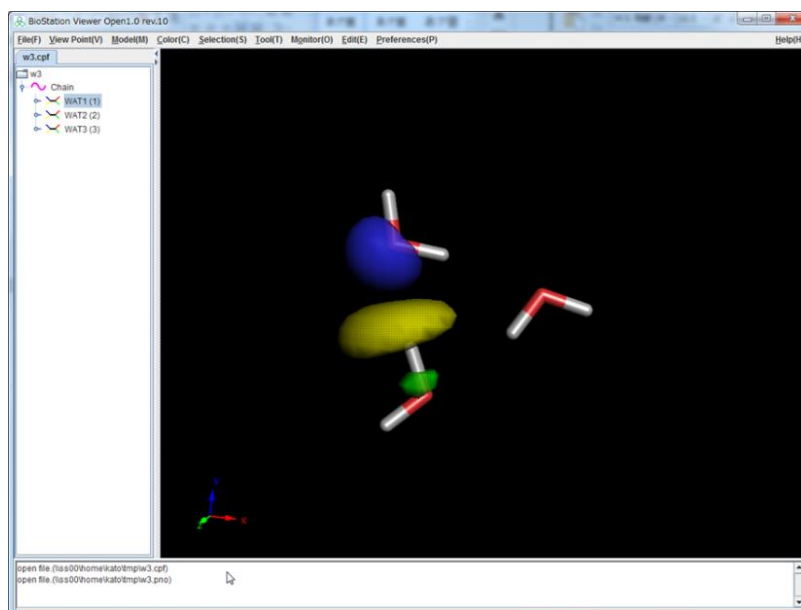
1 回目 (w3-1st.ajf) の計算完了後、2 回目 (w3-2nd.ajf) を用いた計算も行ってください。計算が完了すると生成されるファイルのうち、w3.cpf と w3.pno をダウンロードしてください。

4.3 CFI 解析

ダウンロードした cpf ファイルを BioStation Viewer にドラッグ&ドロップすると、左下図の様に計算した構造が描画されます。その後、pno ファイルをドラッグ&ドロップすると、右下図の CFI 解析結果の画面がポップアップで表示されます。



ポップアップされた画面には、Relaxation energy, Hole と Particle の占有数が表示されます。可視化したい軌道の Display にチェック⇒Apply（画面下部）とすると、下図の様にチェックした軌道が描画されます。（下図は、Hole, Particle の Value を押すとポップアップで表示される設定画面にて、Transparency を 20 としたものです。）



5 課題 5：軌道レベルでの分極相互作用解析（FILM 解析）

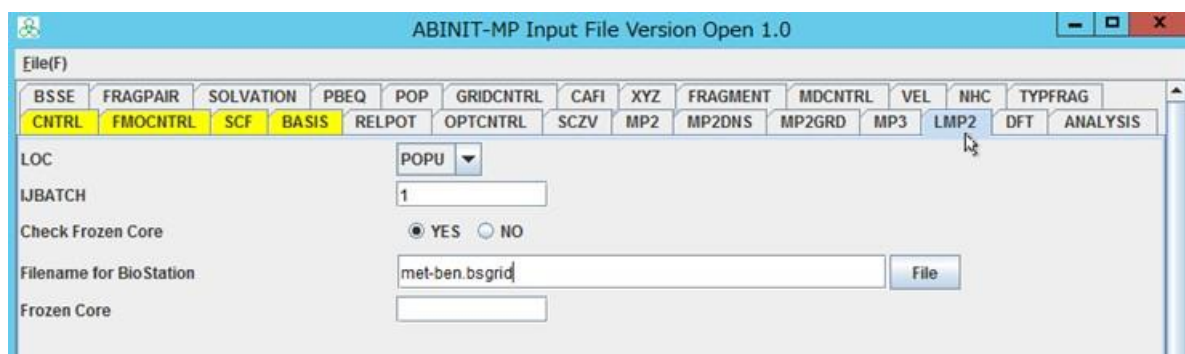
本課題ではメタン - ベンゼンのモデル系を題材に、軌道レベルでの分極相互作用解析の方法を説明します。

5.1 入力データの作成

以下の pdb ファイルを BioStation Viewer で開きます。

```
ABINIT-MP_HANDSON_20190327\05_FILM\met-ben.pdb
```

File→Edit ABINIT-MP Input File から AJF ファイル作成画面を開き、CNTRL タブにて Method を “LMP2” に設定します。次に、LMP2 タブにて Filename for BioStation に “met-ben.bsgird” を記入します。その上で File→Save File から “met-ben.ajf” として保存してください。



ajf ファイル、pdb ファイルを FX10 に転送し、python スクリプトを用いて ajf ファイルを整形してください。

5.2 FMO 計算

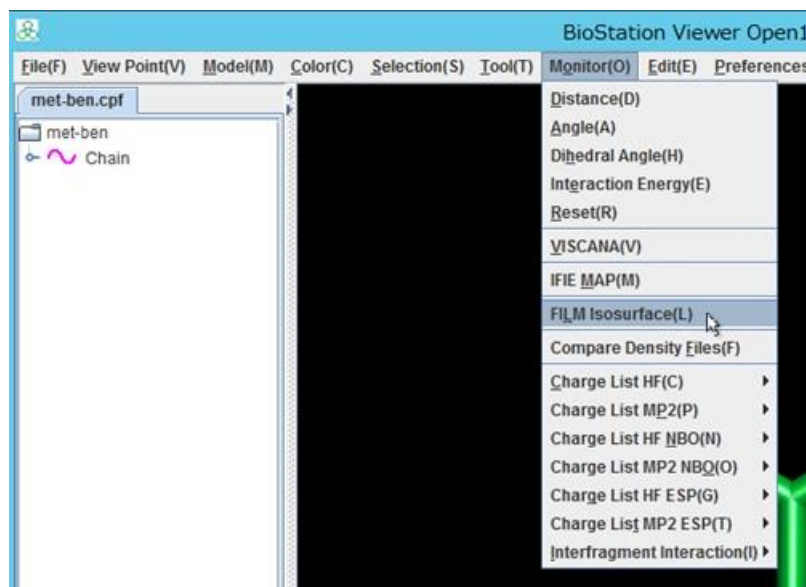
FMO 計算の実行方法は課題 1 の時と同じです。RIST の FX10 で流す場合には、sh スクリプトを編集して ajf ファイル名を変更した上で計算を流します。なお、本課題の系は 2 フラグメントとなるため、run_mpi.sh 内で #PJM -mpi “proc=1” とした上で計算を行ってください。

```
pjsub --interact run_mpi.sh
```

計算が完了すると生成されるファイルのうち、met-ben.cpf と met-ben.bsgird_2_1.lmp2 をダウンロードしてください。

5.3 FILM 解析

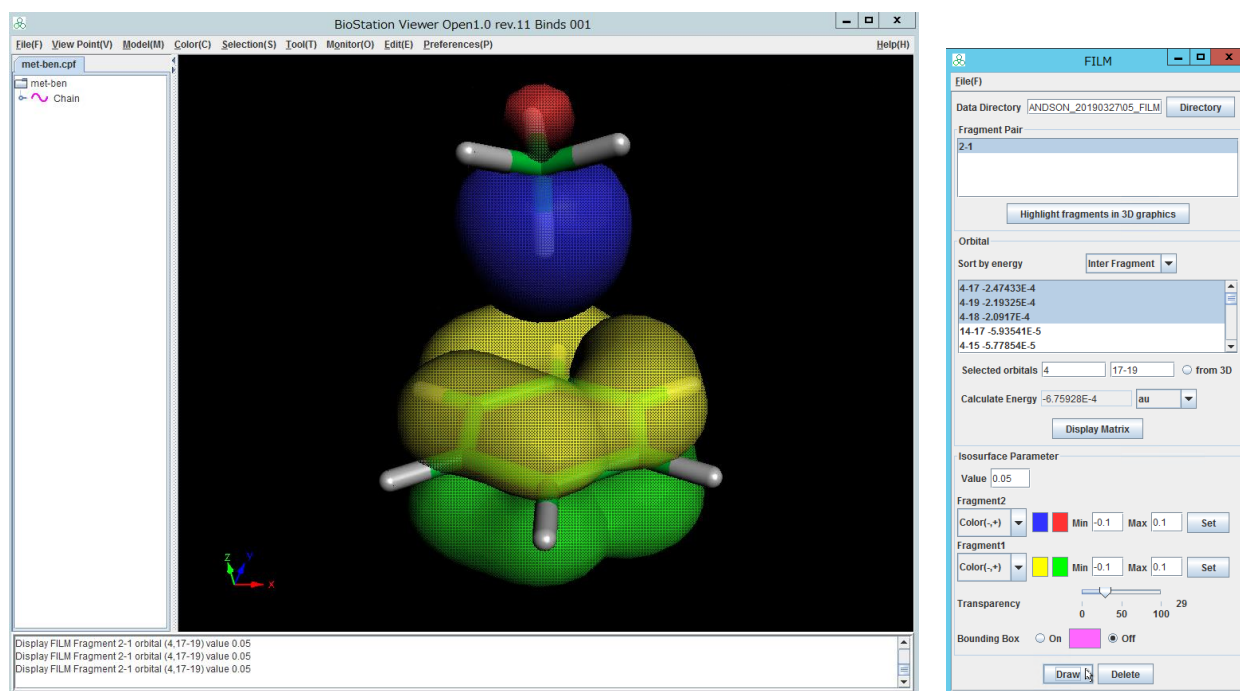
ダウンロードした cpf ファイルを BioStation Viewer で読み込んだ後、Monitor→FILM Isosurface をクリックします。



すると、左下図のような FILM 設定画面が表示されますので、Data Directory として、“met-ben.bsgrid_2_1.lmp2”がある Directory を指定してください。正しくディレクトリが指定されると、右下図のように、.lmp2 ファイルが読み込まれ、Fragment Pair や Orbital 間のエネルギーが表示されます。



FILM 設定画面の Orbital 部は複数選択が可能 (Shift を押しながらクリック) となっているため、4-17, 4-19, 4-18 を選択した状態で Draw を押すと、ベンゼンの π 軌道とメタンの CH 間にある CH- π 相互作用の様子を可視化することが出来ます。Isosurface Parameter の Value や Transparency を変更することで、可視化の調整が可能です。



また、FILM 設定画面の中央部にある Calculate Energy から対応するエネルギーの確認ができます。その際、エネルギーの単位系として au 及び kcal/mol が選択可能です。

6 課題 6：局所応答分散（LRD）と MP2 の比較

本課題では水の 3 量体を例にとり、局所応答分散（LRD）と MP2 の比較を行います。

6.1 入力データの作成

以下の pdb ファイルを BioStation Viewer で開きます。

ABINIT-MP_HANDSON_20190327\06_LRD\w3.pdb
--

まず、MP2 用 AJF ファイルを作成します。File→Edit ABINIT-MP Input File から AJF ファイル作成画面を開き、CNTRL タブにて Method を“MP2”に、Write Geometry File を“w3-mp2.cpf”に設定して下さい。さらに、ANALYSIS タブにて PIEDA を“YES”にした上で、“w3-mp2.ajf”として保存して下さい。次に、LRD 用 AJF ファイルですが、こちらは先ほど作成した MP2 用の AJF ファイルを“w3-lrd.ajf”としてコピーし、以下の様に Method を“HF+D”に、WriteGeom を“w3-lrd.cpf”にテキストエディタで編集して下さい。

w3-lrd.ajf
&CNTRL ElecState='S1' Method='HF+D' Nprint=3 Memory=1800 Natom=0 ReadGeom='w3.pdb' WriteGeom='w3-lrd.cpf' Gradient='NO' Vector='OFF' CPFBIN='NO' THOVL=1.0E-12 E_THSWZ=1.0E-12 G_THSWZ=1.0E-12 /

作成した ajf ファイル及び pdb ファイルを FX10 に転送し、python スクリプトを用いて ajf ファイルの整形を行ってください。

6.2 FMO 計算

FMO 計算の実行方法は課題 1 の時と同じです。RIST の FX10 で流す場合には、sh スクリプトを編集して ajf ファイル名を変更した上で計算を流します。なお、本課題の系は 3 フラグメントとなるため、run_mpi.sh 内で#PJM -mpi “proc=3”とした上で計算を行ってください。

```
pjsub --interact run_mpi.sh
```

MP2 及び HF+D の 2 つの計算を順に行ってください。計算が完了すると生成されるファイルのうち、w3-mp2.log と w3-lrd.log をダウンロードしてください。

6.3 結果の確認

まず、w3-mp2.log から見ていきます。テキストエディタで開き “PIEDA” を検索すると、ファイルの後半に以下の箇所を見ることができます。これは IFIE をエネルギー成分に分解した PIEDA の結果であり、MP2 の寄与として DI(MP2)の値を得ることができます。

w3-mp2.log						
## PIEDA						
IJ-PAIR		ES	EX	CT+mix	DI(MP2)	q(I=>J)
		/ kcal/mol	/ kcal/mol	/ kcal/mol	/ kcal/mol	/ e

2	1	-12.268820	8.549969	-3.606582	-1.906203	-0.035945
3	1	-10.995870	7.774764	-3.216175	-1.819632	0.031929
3	2	-12.339160	8.476232	-3.472605	-1.881427	-0.034884

一方、w3-lrd.log においても同じく PIEDA の結果を見てみると、以下の様な値が得られます。

w3-lrd.log						
## PIEDA						
IJ-PAIR		ES	EX	CT+mix	DI(MP2)	q(I=>J)
		/ kcal/mol	/ kcal/mol	/ kcal/mol	/ kcal/mol	/ e

2	1	-12.268820	8.549969	-3.606582	-0.351977	-0.035945
3	1	-10.995870	7.774764	-3.216175	-0.357710	0.031929
3	2	-12.339160	8.476232	-3.472605	-0.353146	-0.034884

今回の水 3 量体の系においては両者の計算時間の差はほとんどありませんが、より大きな系を計算する場合には、LRD の方が計算コストも少なく済むという利点もあります。

以上