

高精度全電子計算に基づくレクチン-糖鎖間相互作用解析

研究代表者: 中野 祥吾 (静岡県立大学・食品栄養科学部)

共同研究者: 常盤 恭樹 (東北大学大学院・理学研究科)

舘野 浩明 (産業技術総合研究所・創薬基盤研究部門)

平林 淳 (産業技術総合研究所・創薬基盤研究部門)

フラグメント分子軌道法について

生体分子間の相互作用解析にMO法が必要な理由

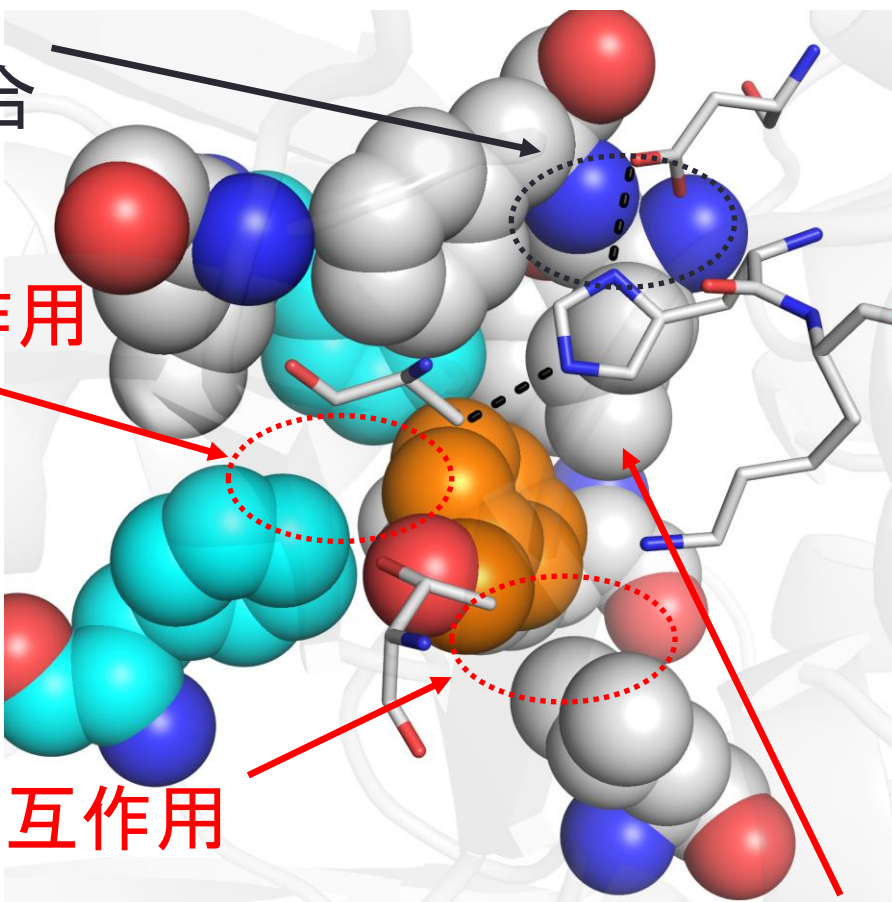
水素結合
イオン結合

π - π 相互作用

CH- π 相互作用

疎水性相互作用

タンパク質-リガンド間に存在する相互作用



分子力学法 (MM法)

- ・水素結合やイオン結合は見積もれる。
- ・弱い相互作用 (赤字) は見積もれない。

分子軌道法 (MO法)

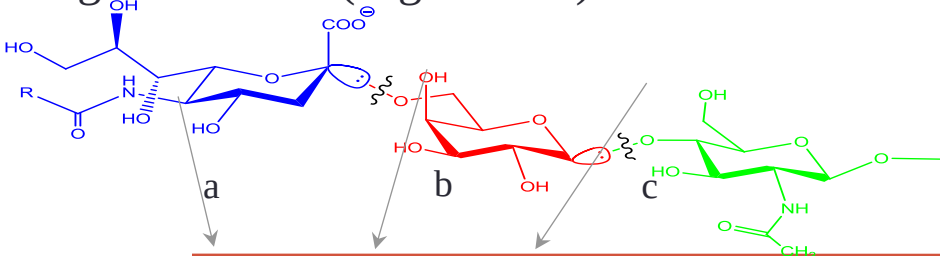
- ・弱い相互作用 (赤字) を含め、相互作用を見積もれる。

・レクチン-糖鎖の相互作用を正しく見積もるには、MO法が必要となる。

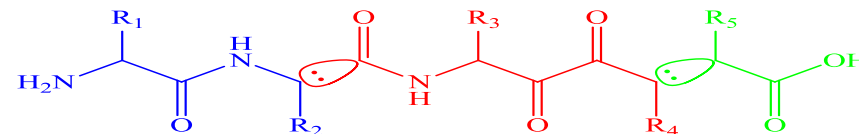
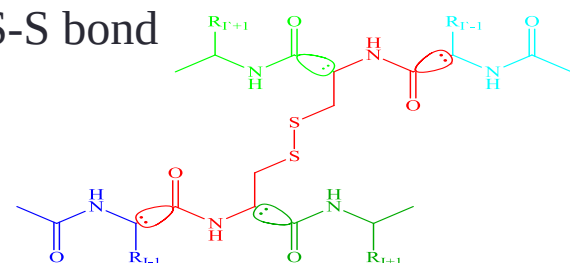
フラグメント分子軌道法†

- 分子集合体を数十原子程度の小さな**フラグメント**に分割
- 各**フラグメント**を並列化計算で高速処理

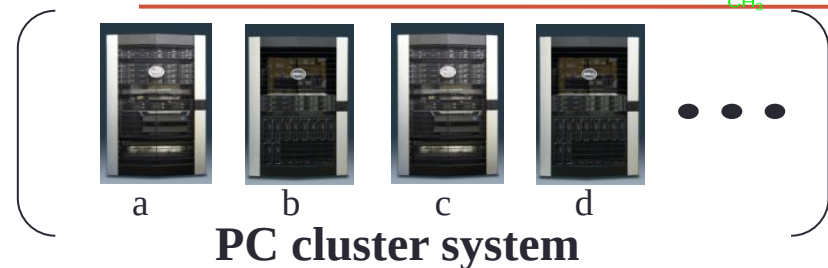
Oligosaccharide (sugar chain)



S-S bond



Protein

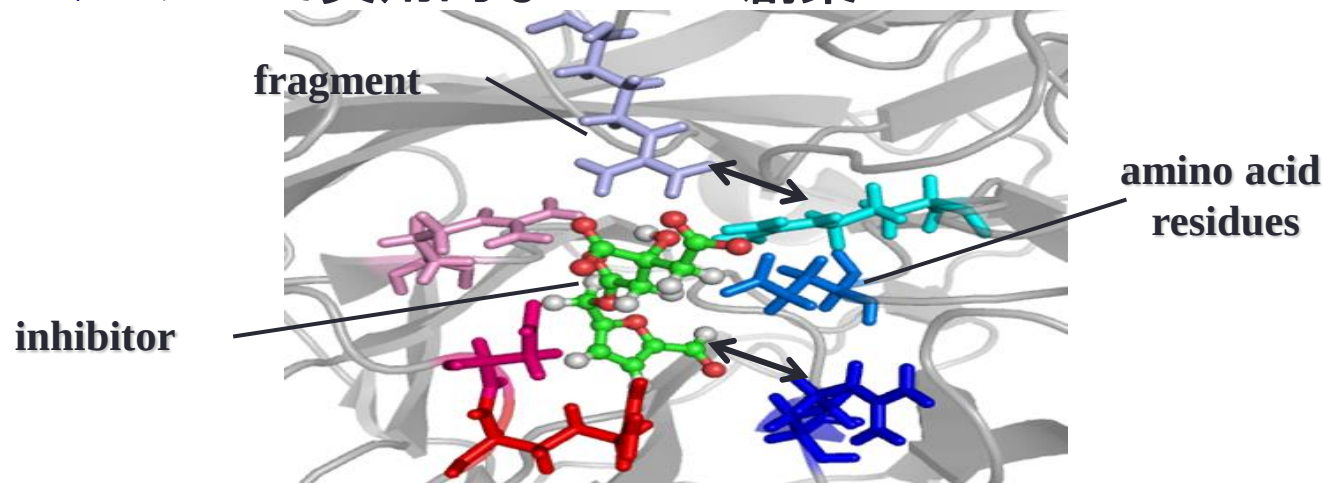


➤ 変異実験とリンクした解析

➤ **Laboレベル**のPCで実用的な*in silico*創薬

FMO 計算プログラムPAICS ‡

- フリーダウンロード
- pdbからの容易な入力作成
- 結果の自動処理(可視化)



† D.G. Fedorov and K.Kitaura Eds., "The Fragment Molecular Orbital Method: Practical Applications to Large Molecular Systems" CRC press (2009).

‡ <http://www.paics.net>.

研究計画の進め方

糖鎖構造データベース

PyMolプラグイン(開発言語:Python)
担当: 研究代表者

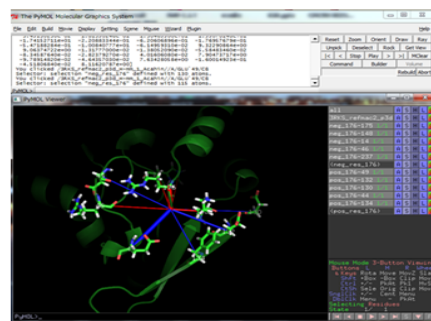
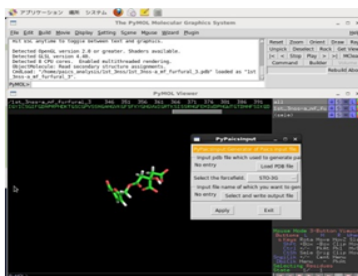
データ解析用Ruby script (開発言語:Ruby)
担当: 共同研究者

PyMol上で構造を見ながらデータベースを変換(手動及び自動)。

データベースを直接変換(自動)

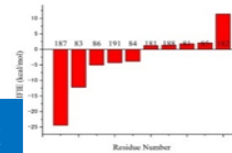
PaicsPy_input

Ruby script



PAICS

Ruby script



糖鎖の正確なエネルギーポテンシャル

PaicsPy_output

計算結果(エネルギーポテンシャル)をPyMol上で可視化。

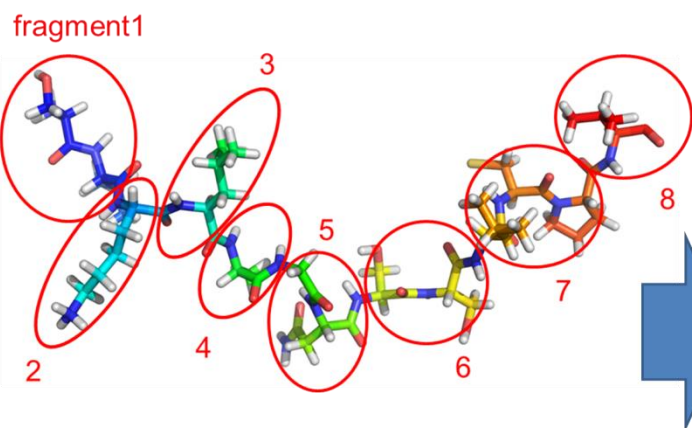
構造生物学の研究者

計算化学の研究者

PyMolを介して糖鎖構造データベース利用を促進。

研究成果1. PaicsPy_inputの開発

- PyPaics_input: PDB (or MOL2) data can be converted to input file of PAICS.



```
mpi_np 1
mem_mbyte 1792

ATOM
5748      1
1  7      cc-pVDZso_007  44.433131  71.000787  66.108292
2  6      cc-pVDZso_006  42.356321  70.112621  64.377302
3  6      cc-pVDZso_006  39.931804  71.645183  64.804377
4  8      cc-pVDZso_008  38.376559  70.921419  66.372853
5  6      cc-pVDZso_006  41.813971  67.261019  64.753353
6  6      cc-pVDZso_006  40.530846  66.057269  62.461120
7  8      cc-pVDZso_008  44.092980  65.953333  65.225785
8  1      cc-pVDZso_001  42.944027  70.384736  62.419540
9  1      cc-pVDZso_001  40.604544  66.968114  66.399309
```

```
FRAGMENT
374
⋮
⋮
⋮

* ifragment=1
FRAG_ATOM 1 14 0
1 2 5 6 7 8 9 10 11 12 13 14 15 16

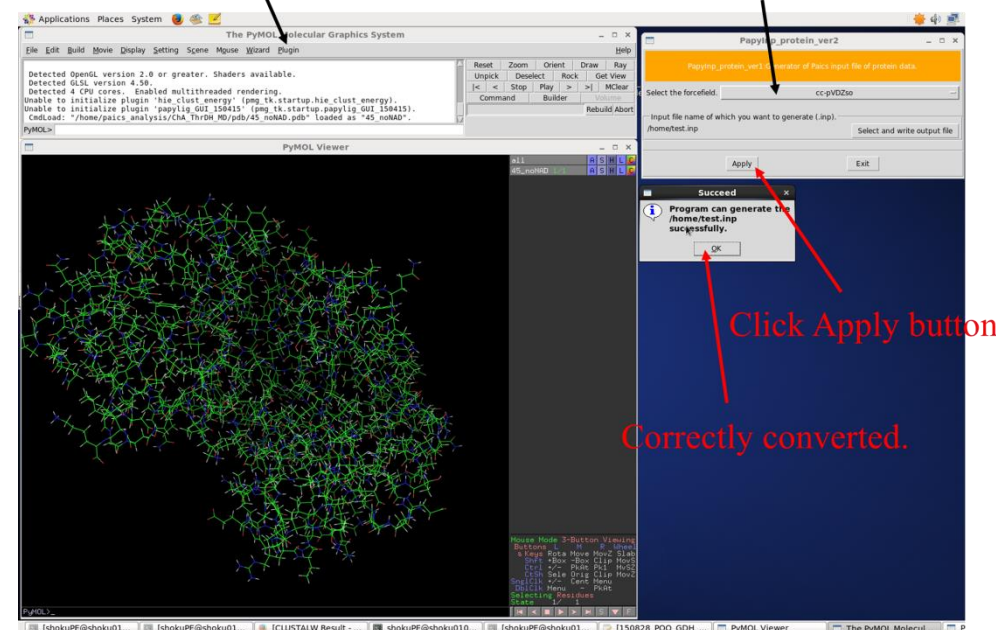
* ifragment=2
FRAG_ATOM 0 16 1
17 18 21 22 23 24 25 26 27 28 29 30 31 32 3 4
2

* ifragment=3
FRAG_ATOM -1 15 1
33 34 37 38 39 40 41 42 43 44 45 46 47 19 20
18

* ifragment=4
FRAG_ATOM 1 22 1
48 49 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66
67 68 69 35 36
34
```

Input file of PAICS.

- Click the Plugin -> PaicsPy_input
- Selection of base function.
- Input output file name.



Correctly converted.

PDBからPAICS入力形式へのファイル変換。

ファイル変換用Plug-in, PaicsPy_input

・PAICS入力ファイルをPyMOL上で作成するplug-inを開発した。

研究成果2. 相互作用解析用ruby scriptの開発

```

3981 ## Save Plot using Gxplot ***
3982 def plotGxplot(textFile, frstNumber, log)
3983
3984     ## Gxplot
3985     cmd = "gxplot"
3986
3987     ## Windows
3988     if checkOS == "Windows" then
3989         if which(cmd) != "" then
3990             if (testFile) then
3991                 # puts "gxplot -persist setting_total.plt"
3992                 puts "gxplot setting_total.plt"
3993                 log.info "", "gxplot setting_total.plt", "", false
3994                 if ((frstNumber.size == 1) then
3995                     # puts "gxplot -persist setting_pharmacophores.plt"
3996                     puts "gxplot setting_pharmacophores.plt"
3997                     log.info "", "gxplot setting_pharmacophores.plt", "", false
3998                 end
3999                 log.info "", "Process : Plot finish", "", textFile
4000             else
4001                 system("gxplot -persist setting_total.plt")
4002                 system("gxplot setting_total.plt")
4003                 log.info "", "gxplot setting_total.plt", "", false
4004                 if ((frstNumber.size == 1) then
4005                     system("gxplot -persist setting_pharmacophores.plt")
4006                     system("gxplot setting_pharmacophores.plt")
4007                     log.info "", "gxplot setting_pharmacophores.plt", "", false
4008                 end
4009                 log.info "", "Process : Plot finish", "", textFile
4010             end
4011         end
4012     end
4013
4014     ## Other OS (and on Gxwin terminal)

```

```
=====
>>                                     <<<
>> Parallelized ab-initio calculation system          <<<
>>                                     <<<
>> based on                                           <<<
>>                                     <<<
>> fragment molecular orbital method                 <<<
>>                                     <<<
>> =====<<<=====

now time   : 01/01/2013 , 14:58:37

These logs are printed out by the CPU of MPI rank =      0 .
If you want to get logs by the other CPUs, the following input lists should be set.

w_log_file , w_result_file

=====
input information
=====

< memory >

succeed in allocating memory @ all processors :           1792 Mbyte

< mpi parallel >

total processor number =         48

processor    1 : hpcs02
processor    2 : hpcs02
processor    3 : hpcs02
processor    4 : hpcs02
processor    5 : hpcs02
processor    6 : hpcs02
processor    7 : hpcs02
processor    8 : hpcs02
processor    9 : hpcs03
processor   10 : hpcs03
processor   11 : hpcs03
processor   12 : hpcs03
```

PAICS output

PAICS output file

Residue A

Residue B

[illegible]

csv file

- ・PAICS出力ファイルをcsv形式に変換するscriptを開発した。

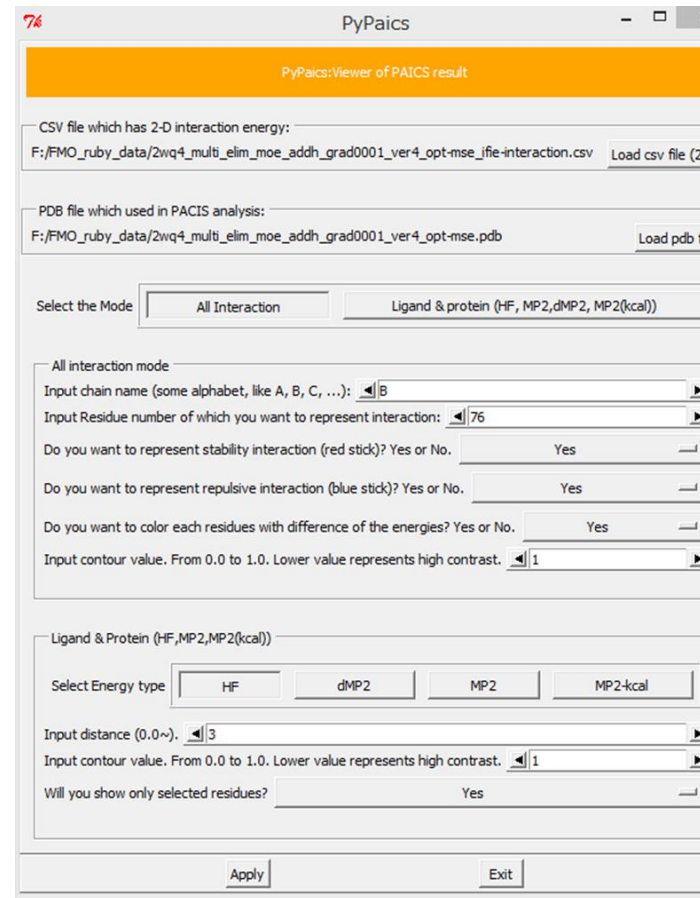
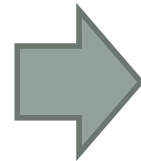
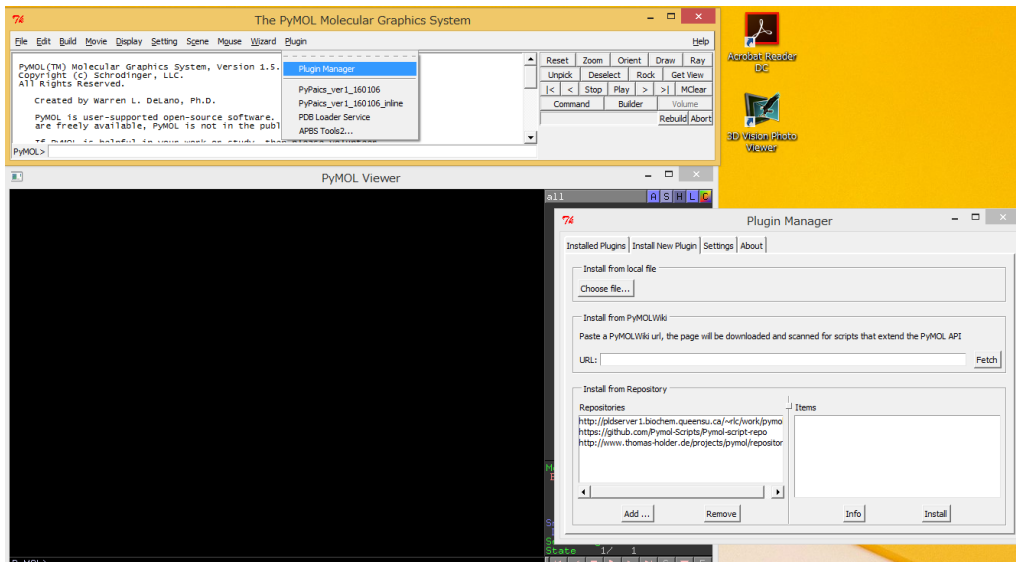
研究成果3. PaicsPy_outputの開発

Residue A →

↓ Residue B

					1	2	3	4	5	6	7	8
1	XXX	MET	p			0	0	0	0	0	0	0
2	VAL	VAL	p		-9472.25	0	0	0	0	0	0	0
3	THR	THR	p		1.809737	-9451.51	0	0	0	0	0	0
4	ALA	ALA	p		1.313377	-4.48356	-9471.63	0	0	0	0	0
5	HIE	HIS	p		1.281374	0.409136	-5.89231	-9457.96	0	0	0	0
6	PHE	PHE	p		0.928087	-0.04267	-0.42043	-3.52723	-9466.97	0	0	0
7	VAL	VAL	p		0.876631	0.207706	-0.25289	-0.19265	-7.26028	-9460.51	0	0
8	LEU	LEU	p		0.142445	-0.0433	0.01506	0.03263	0.247866	-3.95645	-9444.12	0

csv file



相互作用を
可視化。

・PDBとcsv fileを読み込み、相互作用エネルギーをPyMOL上で可視化するPlug-in
→ レクチン-糖鎖構造 (PDB ID: 2WQ4) を用いてinstallから可視化まで実演を行う。

PaicsPy_output (All interaction mode)

1. Load pdb from “File” menu.

2. Start “PyPaics” from plugin menubar.

3. Load csv file from this tab.

4. Load pdb file from this tab.

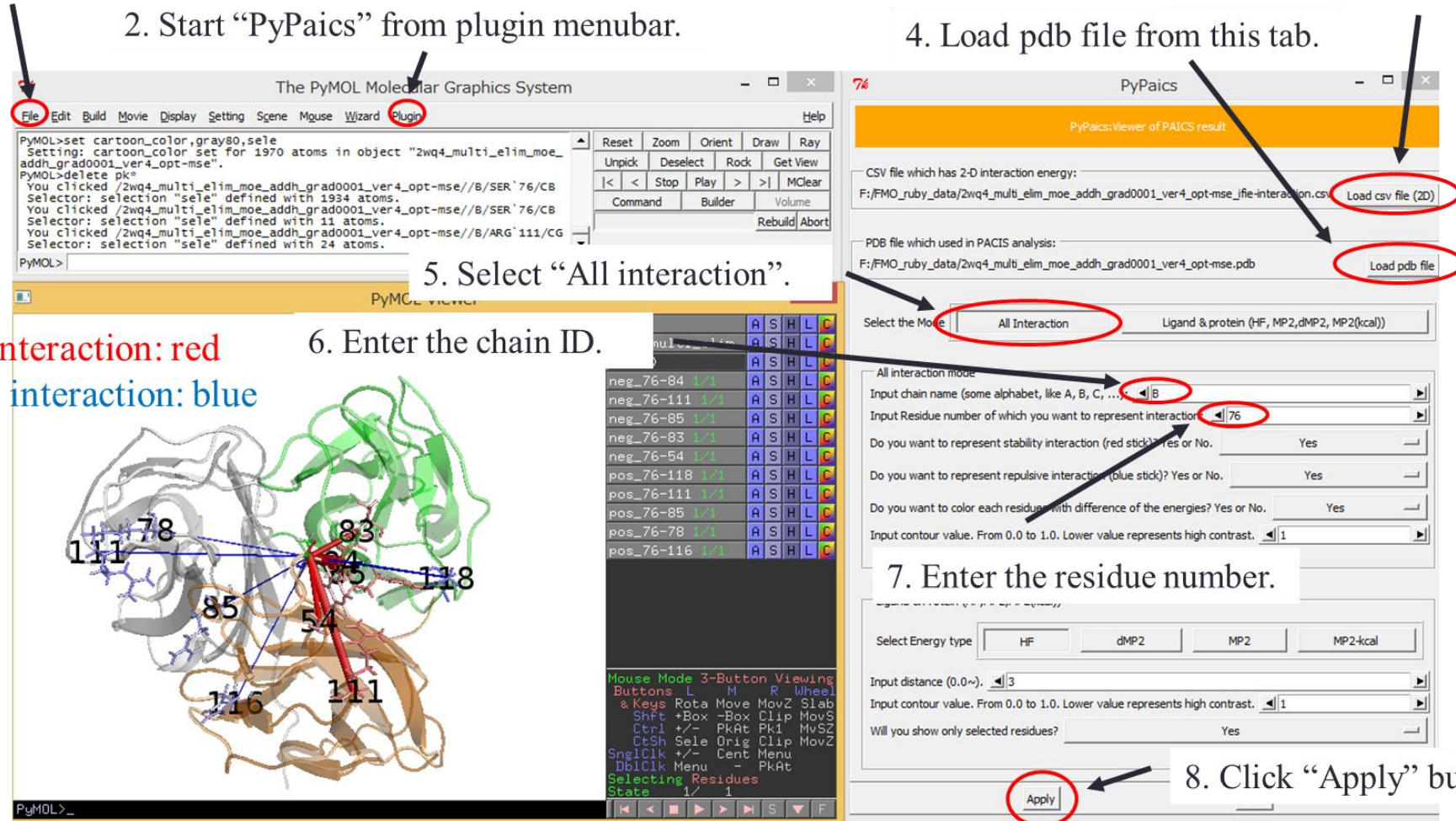
5. Select “All interaction”.

6. Enter the chain ID.

Stable interaction: red
Repulsive interaction: blue

7. Enter the residue number.

8. Click “Apply” button.



・レクチンのドメイン間相互作用を定量的に可視化が可能。

PaicsPy_output (Ligand & Protein mode)

1. Load pdb from “File” menu.

2. Start “PyPaics” from plugin menubar.

3. Load csv file from this tab.

4. Load pdb file from this tab.

5. Select “Ligand & protein”.

7. Enter distance.

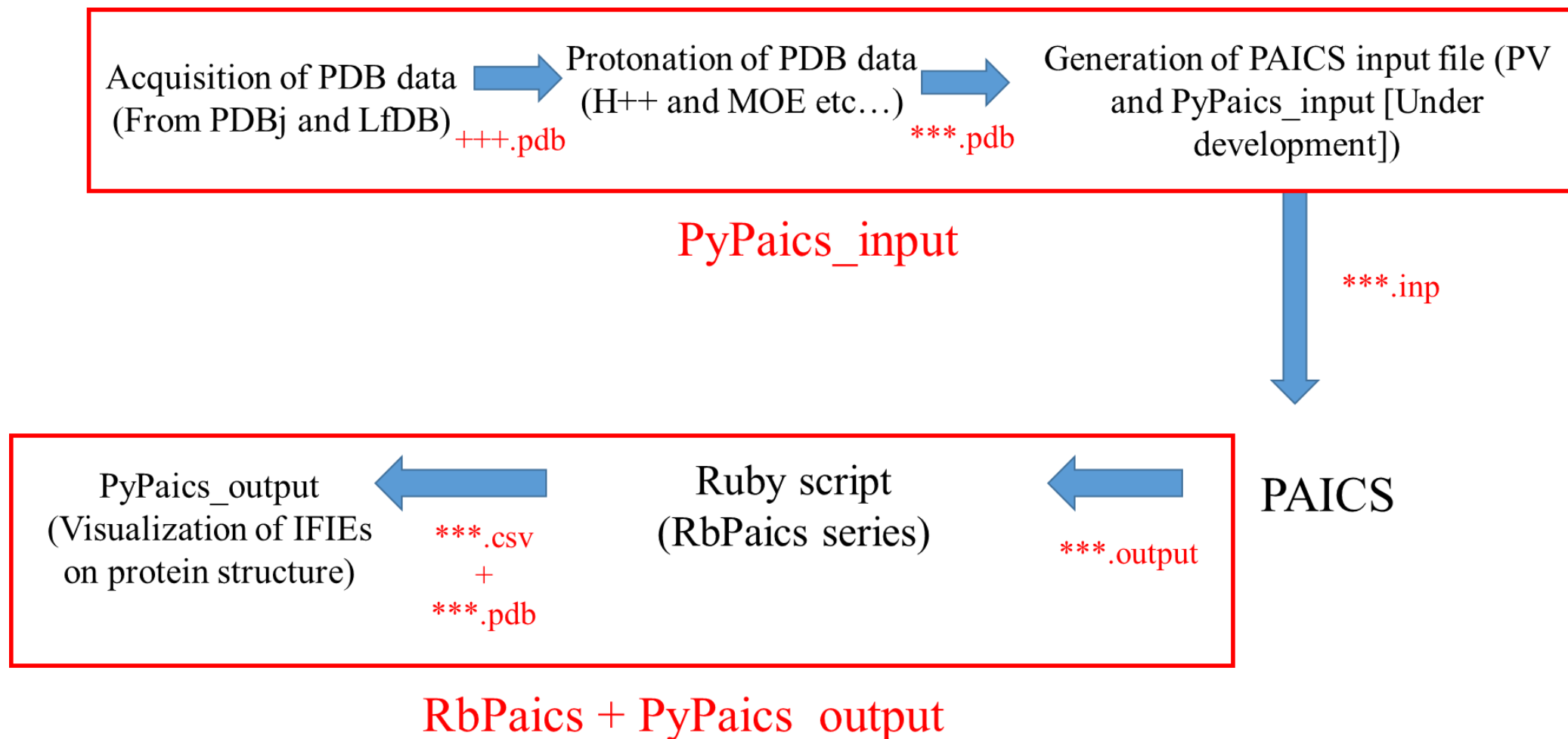
6. Select energy type.

8. Click “Apply” button.

Stable interaction: red
Repulsive interaction: blue

・レクチン-糖鎖間の相互作用を定量的に可視化が可能。

PyPaics実行の流れ (まとめ)

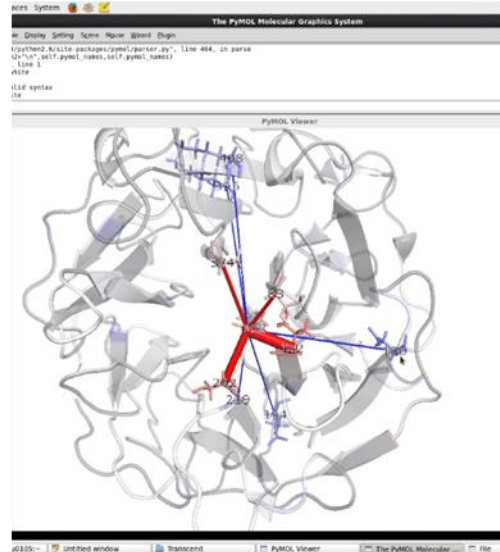


・PyPaicsを用いて、データベース上のPDBファイルに相互作用エネルギーという情報を付加可能 → データベース上の構造情報の利用を促進。

期待される成果と将来展望



データベースに登録されている個々の構造情報に....

[illegible]

・医薬品や有用化合物の合理的設計

相互作用エネルギー情報を
リンクさせたデータベースを
PaicsPyを介して作成。