Laboratory of Bioinformatics and Computational Intelligence

labic.utfpr.edu.br/

PathMolD-AB: Generation of Spatiotemporal Pathways of Protein Folding Using Molecular Dynamics with a Coarse-grained Model

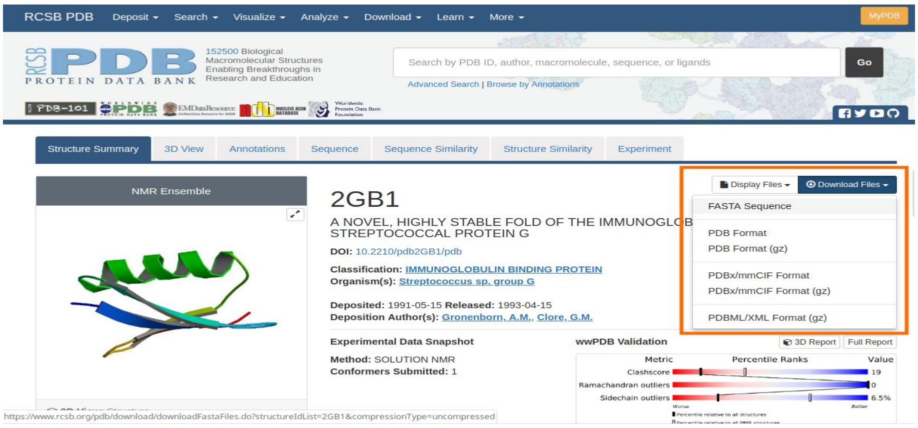
Step-by-step: Generate your first protein folding pathway and analyze it

The purpose of this tutorial is to describe in a simplified way how to generate your first folding trajectory data and visualize the generated folding data for your analysis with PathMolD-AB. The steps required to accomplish this goal are presented below.

step1) Generate your protein AB sequence based on a FASTA file
step2) Set the protein information and the simulation parameters
step3) Compile and execute the protein folding simulation
step4) Realize the protein folding analysis

obs: words highlighted with yellow represent files and red represent line commands.

##
step1) generate your protein AB sequence based on a FASTA file
##
1.1) Download the FASTA file in the PDB database, the square in the figure below highlights the download location.



2GB1

A NOVEL, HIGHLY STABLE FOLD OF THE IMMUNOGLOB
STREPTOCOCCAL PROTEIN G

DOI: 10.2210/pdb2GB1/pdb

Classification: IMMUNOGLOBULIN BINDING PROTEIN
Organism(s): Streptococcus sp. group G

Deposited: 1991-05-15 **Released:** 1993-04-15
Deposition Author(s): Gronenborn, A.M., Clore, G.M.

Experimental Data Snapshot

Method: SOLUTION NMR
Conformers Submitted: 1

wwPDB Validation

https://www.rcsb.org/pdb/download/downloadFastaFiles.do?structureIdList=2GB1&compressionType=uncompressed

In this sample, we download the 2GB1 fasta file from https://www.rcsb.org/structure/2GB1 as presented in the below Figure

1.2) Access the directory SRC, then, execute the command below.

format

```
$ python AB_sequence.py <pdb id of the fasta file>
```

example

```
$ python AB_sequence.py 2GB1
```

*Make sure that the file fasta is in the INPUT directory.
*The AB_sequence.py was developed in Python 2.7 language.
*Here, we use the Alberts AB classification.

The program will generate the AB sequence file of the 2GB1 protein in the INPUT directory.
example: 2GB1.txt

The content of the 2GB1.txt file is show below.

```
ABBBAAABABBABABBBBBAABAABABBAABBBABBBAABABABBBBABBBABABB
```

,where A and B represents the Hydrophobic and Polar amino acids, respectively.

###
step2) set the protein information and the simulation parameters
###

Insert the information about the protein and the simulation parameters into the input file. The file 2GB1_56.in (in the INPUT directory) is a sample of this input file for the protein 2GB1. The descriptions of each information are present below.

sequence : AB sequence of the protein (use step1 to get this information)
ProtLen : protein size, number of amino acids
LV : length of the 2D or 3D box
stepLimit : max number of steps
temperature : temperature
savepathways : y enable the function that saves amino acids coordinates in text files. This files can be used to show an example of a pathway
pathwaysstep : steps between generations of text coordinate files.

The content of the 2GB1_56.in file is show below.

```
sequence = ABBBAAABABBABABBBBBAABAABABBAABBBABBBAABABABBBBABBBABABB
ProtLen = 56
LV = 112
stepLimit = 3000000
temperature = 0.1
savepathways = y
pathwaysstep = 3000
```

###
step 3) compile and execute the protein folding simulation
###

*The PathMolD-AB program was developed in C language/CUDA.

3.1) ONLY FOR CPU - Access the SRC_CPU directory and execute the command below to compile the MD program.

example

```
$ gcc -o a.out main_CPU.c func_MD_CPU.c -lm
```

3.2) Execute the MD program with the executable file (a.out) with the input file (2GB1.in) and a seed number, to generate a random initial structure of the simulation of the protein folding.

format

```
$ ./executable <input file> <seed>
```

example

```
$ ./a.out ../INPUT/2GB1_56.in 0 && mv pathways56_0.txt ../OUTPUT/pathways56_0.txt
```

after the execution of MD simulation, it will be generated a output file (pathway data) with the information about the protein structure, free energy and radius of gyration along the iterations (in our sample, the output file generated was pathways56_0.txt and we also move this file to the OUTPUT directory).

3.3) ONLY FOR GPU

*parallel PathMolD-AB is written in C and CUDA.

Access the SRC directory and compile the parallel PathMolD-AB program

example

```
$ nvcc --gpu-architecture = compute_61 --device -c main.c functions.cu
$ nvcc --gpu-architecture = compute_61 main.o functions.o
```

3.4) execute the parallel PathMolD-AB program

format

```
$ ./a.out <input file> <output file> <seed> > <GPU ID>
```

example

```
$ ./a.out ../INPUT/2GB1_56.in ../OUTPUT/pathways56 0 0
```

<input file> : parameters of the simulation (generate by the step 2)
<output file> : name of the output file/ pathway data
<seed> : seed used to generate the initial structure of the protein

<GPU ID>   : ID of the GPU that will run the simulation. For computers with only one GPU, use the ID as zero.

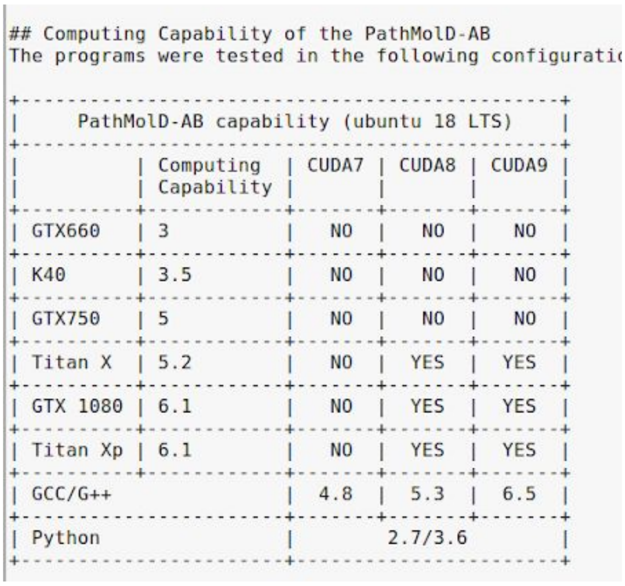
*the root directory contains a makefile to run the programs.

```
$ make ab              # convert the AB sequence from the amino acid sequence
$ make gpu             # compile the parallel molecular dynamic version
$ make cpu              # compile the sequential molecular dynamic version
$ make run_gpu          # execute the parallel molecular dynamic version
$ make run_cpu          # execute the sequential molecular dynamic version
$ make vizualize       # generating the folding simulation video
```

Laboratory of Bioinformatics and Computational Intelligence

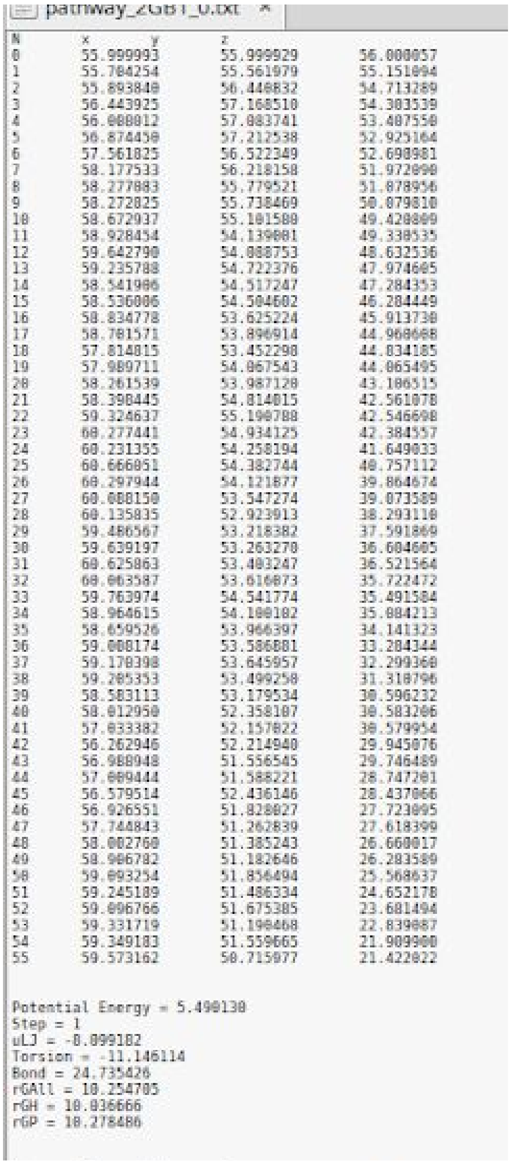
labic.utfpr.edu.br/

*parallel PathMolD-AB configurations tested (left) a sample of the pathway data generated by PathMolD-AB program (right)



```
## Computing Capability of the PathMolD-AB
The programs were tested in the following configuratic

+---------------------------------------------------+
|     PathMolD-AB capability (ubuntu 18 LTS)        |
+---------------------------------------------------+
|          | Computing  | CUDA7 | CUDA8 | CUDA9 |
|          | Capability |       |       |       |
+----------+------------+-------+-------+-------+
| GTX660   | 3          |   NO  |   NO  |   NO  |
+----------+------------+-------+-------+-------+
| K40      | 3.5        |   NO  |   NO  |   NO  |
+----------+------------+-------+-------+-------+
| GTX750   | 5          |   NO  |   NO  |   NO  |
+----------+------------+-------+-------+-------+
| Titan X  | 5.2        |   NO  |  YES  |  YES  |
+----------+------------+-------+-------+-------+
| GTX 1080 | 6.1        |   NO  |  YES  |  YES  |
+----------+------------+-------+-------+-------+
| Titan Xp | 6.1        |   NO  |  YES  |  YES  |
+----------+------------+-------+-------+-------+
| GCC/G++               |  4.8  |  5.3  |  6.5  |
+-----------------------+-------+-------+-------+
| Python                |        2.7/3.6        |
+-----------------------+-----------------------+
```



pathway_2GB1_0.txt

```
N        x        y          z
0        55.999993           55.999929           56.000057
1        55.704254           55.561979           55.151094
2        55.893840           56.440832           54.713289
3        56.443925           57.168510           54.303539
4        56.000812           57.003741           53.407550
5        56.874450           57.212538           52.925164
6        57.561825           56.522349           52.690981
7        58.177533           56.218158           51.972090
8        58.277083           55.779521           51.078956
9        58.272825           55.730469           50.079810
10       58.672937           55.101580           49.420009
11       58.928434           54.139001           49.330535
12       59.642790           54.000753           48.632536
13       59.235708           54.722376           47.974605
14       58.541900           54.517247           47.284353
15       58.536006           54.504602           46.204449
16       58.834778           53.625224           45.913730
17       58.701571           53.890914           44.960600
18       57.814015           53.452298           44.034185
19       57.909711           54.067543           44.065495
20       58.261539           53.987120           43.106515
21       58.390445           54.814015           42.561078
22       59.324637           55.190780           42.546690
23       60.277441           54.934125           42.304557
24       60.231355           54.258194           41.649033
25       60.666051           54.382744           40.757112
26       60.297944           54.121877           39.864674
27       60.008150           53.547274           39.073589
28       60.135835           52.923913           38.293110
29       59.486567           53.218382           37.591009
30       59.639197           53.263270           36.604605
31       60.625863           53.403247           36.521564
32       60.063587           53.610073           35.722472
33       59.763974           54.541774           35.491504
34       58.964615           54.100182           35.004213
35       58.659526           53.906397           34.141323
36       59.000174           53.586881           33.204344
37       59.170398           53.645957           32.299360
38       59.205353           53.499250           31.310796
39       58.583113           53.179534           30.596232
40       58.012950           52.358107           30.583206
41       57.033382           52.157022           30.579954
42       56.262946           52.214940           29.945076
43       56.980948           51.556545           29.746489
44       57.009444           51.508221           28.747201
45       56.579514           52.436146           28.437066
46       56.926551           51.828027           27.723095
47       57.744843           51.262839           27.618399
48       58.002760           51.385243           26.660017
49       58.906782           51.182646           26.283589
50       59.093254           51.856494           25.568637
51       59.245189           51.486334           24.652178
52       59.096766           51.675385           23.681494
53       59.331719           51.190468           22.839087
54       59.349183           51.559665           21.909900
55       59.573162           50.715077           21.422022


Potential Energy = 5.490130
Step = 1
uLJ = -0.099182
Torsion = -11.146114
Bond = 24.735426
rGAll = 10.254705
rGH = 10.036666
rGP = 10.278486
```

##

---

step 4) realize the protein folding analysis

##

*The pathway_print_multi-subplot.py script was developed in Python 2.7/3.6 language.

With the output file generated by the PathMolD-AB program (step 3), you can generate images of the protein folding trajectory with the program pathway_print_multi-subplot.py.

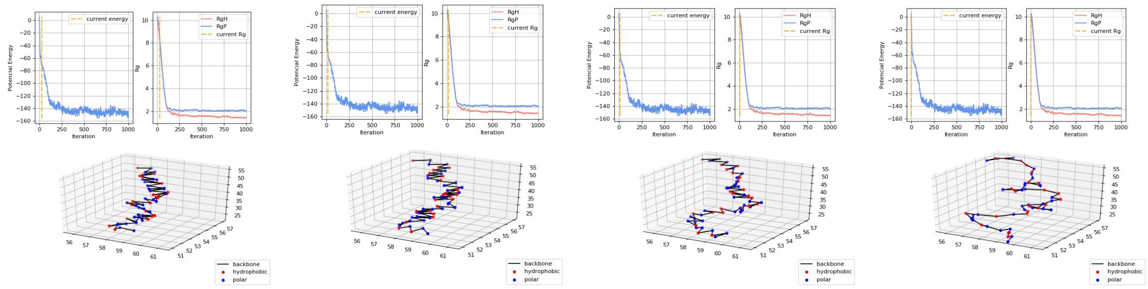
4.1) Before executing pathway_print_multi-subplot.py, you need to set the parameters to read and save the files. Then, open pathway_print_multi-subplot.py file and change the variables *path_pathways*, *filename*, *filesequence* and *path_save*.

path_pathways : folder of the protein pathway data
filename : name of the pathway data (generated by step 3)
filesequence : file containing the AB sequence of the protein (generated by step 1)
path_save :folder where the images will be saved

4.2)execute the command

```
$ python pathway_print_multi-subplot.py
```

Then, the program will produces images of the folding information at each iteration. The samples of the frame 0, 9, 18, and 27 are described below.



The image files of the folding process will be saved in the 'img' folder in this tutorial, and a video of protein protein folding will be generated called folding.mp4.

Laboratory of Bioinformatics and Computational Intelligence

labic.utfpr.edu.br/

Thanks for using PathMolD-AB!!

----

for any doubt send a email to leandrotakeshihattori@gmail.com

Link for the Dataset of Spatiotemporal Pathways of Protein Folding:
https://mega.nz/#F!C5QkHQ6A!Ng2xowc2hVPoHHiSB7ww-w