POAP - Parallelized Open Babel & Autodock suite Pipeline

Operating Manual

by

Mr.A.Samdani & Dr.V.Umashankar
Centre for Bioinformatics
Vision Research Foundation
Sankara Nethrlaya
Tamil Nadu, India
Chennai 600006

Website: http://www.sankaranethralaya.org/research-bioinformatics.html

Email : vumashankar@gmail.com
Email: samdani1593@gmail.com

PREREQUISITES FOR POAP RUN:

Linux Operating system 64 bit with bash shell

GNU parallel (/usr/local/bin)
Link to download and installation of GNU Parallel (http://ftp.gnu.org/gnu/parallel/)

Open Babel 2.4.0 or later: (http://open-babel.readthedocs.io/en/latest/Installation/install.html)

- ✓ Obconformer (/usr/local/bin)
- ✓ Confab (present in version 2.4.0

MGL TOOLS
Ensure to set path to the path where mgltools have been installed (export MGLROOT=/usr/local/MGLtools-1.5.7)
Link to download and installation of MGLTOOLS (http://mgltools.scripps.edu/downloads)

Virtual screening:

Autodock Vina (http://vina.scripps.edu/download.html**)**

- ✓ vina (/usr/local/bin)
- ✓ vina_split (/usr/local/bin)

MGL Tools version 1.5.6 or 1.5.7(need for AutodockZn)
Autodock version 4.2.6 (Compulsory for AutodockZn)

Binaries:

- ✓ autodock4 (/usr/local/bin)
- ✓ autogrid4 (/usr/local/bin)

AutodockZn

The scripts and forcefield required AutodockZn can be downloaded from the link (http://autodock.scripps.edu/resources/autodockzn-forcefield).

Autodock version 4.2.6 (http://autodock.scripps.edu/downloads/autodock-registration/autodock-4-2-download-page/)

Binaries:

- ✓ autodock4 (/usr/local/bin)
- ✓ autogrid4 (/usr/local/bin)

Scripts required for AutodockZn:

- ✓ prepare_gpf4zn.py
- ✓ zinc_psuedo.py

Forcefiled:

- ✓ AD4Zn.dat

OPERATING INSTRUCTIONS:

Ligand preparation:

The options available in ligand preparation with POAP_lig.bash:

-s, -s For Interactive mode

-h, -help To print help file

-l, -ligand To specify ligand directory

For example [bash POAP_lig.bash -l /home/bioinfo/ligands/]

-ff, -forcefield To select forcefield.

The available forcefield are:

UFF

GAFF

Ghemical

MMFF94

MMFF94s

For example [bash POAP_lig.bash -l /home/bioinfo/ligands/ -ff MMFF94]

-t, -T To specify 3D conversion process and speed

Available speed:

Fastest

Fast

Med or medium

Slow

Slowest

For example [bash POAP_lig.bash -l /home/bioinfo/ligands/ -t med]

-c, -C -c <algorithm> <number of conformers> <best(or)all>

To specify the conformer algorithm and number of conformers to be generated and to specify whether to proceed best structure alone or to retain all the conformers structure generated(this is valid for only Genetic Algorithm, Random rotor and Weighted Rotor based conformational search)

Available conformer search:

Genetic algorithm (GA)

Random Rotor Search (RA)

Weighted Rotor Search (WR)

Obconformer(OB)

Confab(CF)

For example [bash POAP_lig.bash -l /home/bioinfo/ligands/ -c WR 50 best]

-m, -M -m <Minimization algorithm> <Number of steps>

To specify the minimization algorithm and number of steps

cg - conjugate gradient

sd - steepest descent algorithm

Other cut off values are default

For example [bash POAP_lig.bash -l /home/bioinfo/ligands/ -m cg 5000]

-pdbqt, -PDBQT -pdbqt <py/ob>

To specify direct pdbqt conversion

py - conversion using prepare_ligand4.py script
ob - conversion using openbabel
For example [bash POAP_lig.bash -l /home/bioinfo/ligands/ -pdbqt py]

-o, -O -o <sdf/mol2/pdbqt>
For example [script -o sdf]
For pdbqt conversion specify py/ob followed by format to useprepare_ligand4.py or obabel to perform pdbqt conversion
-o pdbqt<py/ob>
For example [bash POAP_lig.bash -l /home/bioinfo/ligands/ -o pdbqt py]

-tcms, -TCMS To perform 3D conversion, conformational search, minimization and final output in sdf file format
T- 3D conversion with medium speed
C- conformer generation using weighted rotor of 50 conformers retaining the best structure alone
M- Minimisation using conjugate gradient with 2500 steps and other default values.
S- Final output in sdf file format
For example [bash POAP_lig.bash -l /home/bioinfo/ligands/ -tcms]

-tcmm, -TCMM To perform 3D conversion, conformational search, minimization and final output in mol2 file format
T- 3D conversion with medium speed
C- conformer generation using weighted rotor of 50 conformers retaining the best structure alone
M- Minimisation using conjugate gradient with 2500 steps and other default values.
M- Final output in mol2 file format
For example [bash POAP_lig.bash -l /home/bioinfo/ligands/ -tcmm]

-tcmp, -TCMP To perform 3D conversion, conformational search, minimization and final output in pdbqt file format
T- 3D conversion with medium speed
C- conformer generation using weighted rotor of 50 conformers retaining the best structure alone
M- Minimisation using conjugate gradient with 2500 steps and other default values.
P- Final output in pdbqt file format using prepare_ligand4.py script
For example [bash POAP_lig.bash -l /home/bioinfo/ligands/ -tcmp]

-cms, -CMS To perform conformational search, minimization and final out in sdf file format
C- conformer generation using weighted rotor of 50 conformers retaining the best structure alone
M- Minimisation using conjugate gradient with 2500 steps and other default values.
S- Final output in sdf file format
For example [bash POAP_lig.bash -l /home/bioinfo/ligands/ -cms]

-cmm, -CMM To perform conformational search, minimization and final out in mol2 file format
C- conformer generation using weighted rotor of 50 conformers retaining the best structure alone
M- Minimisation using conjugate gradient with 2500 steps and other default values.
M- Final output in mol2 file format
For example [bash POAP_lig.bash -l /home/bioinfo/ligands/ -cmm]

-cmp, -CMP To perform conformational search, minimization and final out in pdbqt file format
C- conformer generation using weighted rotor of 50 conformers retaining the best structure alone
M- Minimisation using conjugate gradient with 2500 steps and other default values.
P- Final output in pdbqt file format using prepare_ligand4.py script
For example [bash POAP_lig.bash -l /home/bioinfo/ligands/ -cmp]

-ms, -MS To perform minimization and final out in sdf file format
M- Minimisation using conjugate gradient with 2500 steps and other default values.
S- Final output in sdf file format
For example [bash POAP_lig.bash -l /home/bioinfo/ligands/ -ms]

-mm, -MM To perform minimization and final out in mol2 file format
M- Minimisation using conjugate gradient with 2500 steps and other default values.
M- Final output in mol2 file format
For example [bash POAP_lig.bash -l /home/bioinfo/ligands/ -mm]

-mp, -MP To perform minimization and final out in pdbqt file format
M- Minimisation using conjugate gradient with 2500 steps and other default values.
P- Final output in pdbqt file format using prepare_ligand4.py script
For example [bash POAP_lig.bash -l /home/bioinfo/ligands/ -mp]

Instructions for Ligand preparation in Interactive mode:

1. To prepare the ligands by interactive mode start the POAP_lig.bash with –s flag and enter

   bash POAP_lig.bash -s

   This will start the ligand preparation in interactive mode which allows users to provide the input and values for various stages to proceed on.

2. After running the script, the user will be prompted to specify the directory path where the ligands needed to be prepared are kept**. [Note: make sure no other files other than ligands present in the directory].** Enter the directory path location which contains ligand files.

   Please provide directory path containing ligands which are to be prepared
   >>>/home/bioinfo/Desktop/sam/ligands

   Press <Enter> key after typing the directory.

3. The prompt will display the ligand file format and total number of ligands found in the directory specified. The number of CPU threads available in the working system will be displayed and the user will be prompted to enter the maximum number of jobs to execute in parallel. [NOTE: Please provide the number of jobs less than or equal to the number of CPU threads available.]

   Enter the number of jobs to be run in parallel
   Maximum 252 jobs can be run in parallel

Enter the number of jobs less than 252
>>>8

Press <Enter> key

4. Once the No. CPU's are provided, the user will be prompted to enter force field type to be used for preparing the ligands.

Enter the forcefield for performing ligand optimization:
[1] Gaff Gaff force field.
[2] Ghemical Ghemical force field.
[3] MMFF94 MMFF94 force field.
[4] MMFF94s MMFF94s force field.
[5] UFF Universal Force Field.

Enter the forcefield option number
>>>3

Enter the respective option and Press <Enter> key

5. Next the user will be prompted to enter into 3D conversion panel for conversion the 2D ligand data into 3D fork, in case the input ligands are in 2D format. If you wish to convert 3D enter 1 else to skip enter 2 and Press the<Enter> key. On doing this, the user will be prompted to enter the speed for 3D conversion process. Enter the respective options and Press <Enter> key.

++++++++++++++++++++++++++++
+ 2D ---> 3D Co-ordinates +
++++++++++++++++++++++++++++

Do you want to generate 3D conformation if your input is in 2D co-ordinates?
[1] yes
[2] no
>>>1
++++++++++++++++++++++++++++
+ 2D ---> 3D Co-ordinates +
++++++++++++++++++++++++++++

Enter the desired speed for 3D-co-ordinate generation
[1] fastest
[2] fast
[3] med or medium
[4] slow or better
[5] slowest or best
>>>3

6. Next, the command prompt will display the conformation generation options, displaying the different types that can be generated. Enter a relevant option to perform respective conformer generation. The conformer generation using Genetic Algorithm, Random Rotor and Weighted Rotor methods, and all the conformers can be retained or the ligand conformer with lowest energy structure alone can be taken further. While in confab, all the systematically generated structures will be taken further. In case of obconformer selection, the lowest energy structure after performing minimization steps specified will be taken further to specified output generation. Enter 0 to skip conformer generation step and to directly proceed for minimization step.

   ~~~Conformers Generation~~~
   Enter 0 to skip conformer generation

   Enter an option
   [1] To generate ligand conformations using GA (Genetic Algorithm)
   [2] To generate ligand conformations using Random rotor search method
   [3] To generate ligand conformations using Weighted rotor search method
   [4] To generate best conformer using obconformer
   [5] To generate conformer using confab
   >>>3

   Enter the number of conformation to be generated using weighted rotor search method
   >>>50

   Do you want to retain all the conformers generated?
   [1] To retain all the conformers
   [2] To select the best structure alone
   >>>2

7. In the minimization step, the type of algorithm, number of steps and other cut-off values needed to be specified. For default cut off values, 0 can be entered.

   ~~~~Minimization~~~~
   Enter [1] for conjugate or [2] for steepest descent algorithm
   >>>1

   Enter the number of minimization steps to perform
   >>>5000

   Enter the convergence criteria: Enter [0] to set default 1e-6
   >>>0

   Enter the VDW cut-off distance: Enter [0] to set default 6.0
   >>>0

Enter the Electrostatic cut-off distance: Enter [0] to set default 10.0
>>>0

Enter the frequency to update the non-bonded pairs: Enter [0] to set default 10
>>>0

Enter [1] To add hydrogens [0] not to add hydrogens
>>>1

8. Finally, the required output format needed to be specified. If PDBQT conversion selected, provide whether pdbqt conversion needed to be carried out using prepare_ligand4.py script or obabel. In case the user intends to user wants to choose any other format which can best used for tools other than autodock, users can choose other options (like mol2, sdf etc).

   Enter an option for output file format
   [1] sdf
   [2]mol2
   [3]pdbqt
   >>>3

   Enter
   [1] To convert PDBQT used prepare_ligand4.py script
   [2] To convert PDBQT using open babel
   >>>1

9. Finally Press <Enter> key to start the ligand preparation process.

   ----Press Return to start the Ligand preparation-----
   >>>

10. On completion of Ligand preparation, the prepared ligands will be stored in the ligand directory

Virtual screening:

The options available in the virtual screening process using POAP_vs.bash:
-s, -s For Interactive mode
-h, -help To print help file
-l, -ligand To specify ligand directory
For example [bash POAP_vs.bash -l /home/bioinfo/ligands/]
-r, -receptor To specify receptor directory
For example [bash POAP_vs.bash -r /home/bioinfo/receptor/]
-w, -work To specify receptor directory
For example [bash POAP_vs.bash -w /home/bioinfo/working/]
-vv, -vsvina To perform Virtual screening using Autodock Vina
For example [bash POAP_vs.bash -vv]
-mv, -mdvina To perform Multiple protein docking using Autodock Vina
For example [bash POAP_vs.bash -mv]
-va, -vsad To perform Virtual screening using Autodock
For example [bash POAP_vs.bash -va]
-ma, -mdad To perform Multiple protein docking using Autodock
For example [bash POAP_vs.bash -ma]
-vz, -vsaz To perform Virtual screening using AutodockZn
For example [bash POAP_vs.bash -vz]
-mz, -mdaz To perform Multiple protein docking using AutodockZn
-x, -ex Autodock Vina Exhaustiveness(default=8)
-t, -top To select number of top protein-ligand complex
-j, -Jobs To specify number of jobs to run in parallel
[Note: For vina -j represent number of CPU threads to use to run
single VINA job]

Virtual screening using VINA:

1. Start the VS bash POAP_vs.bash script with –s, this will show different options of Virtual screening to be performed, enter 1 to perform virtual screening using VINA and Press <Enter> key

   bash POAP_vs.bash -s
   POAP - Virtual screening Module:

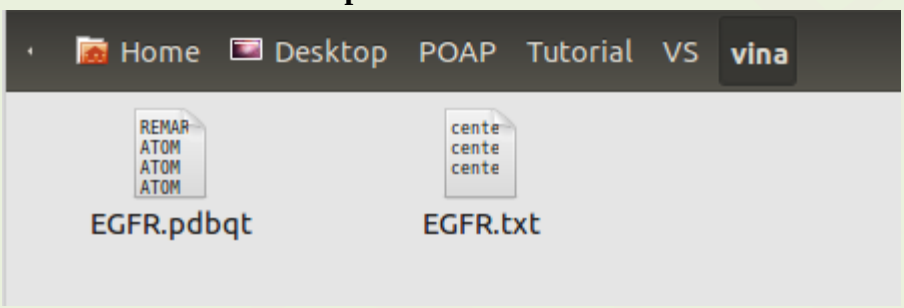
   Enter option
   [1] To perform VS with Autodock Vina
   [2] To perform Multiple Protein docking with Autodock Vina
   [3] To perform VS with Autodock
   [4] To perform Multiple protein docking with Autodock
   [5] To perform VS with AutodockZn
   [6] To perform Multiple protein docking with AutodockZn
   >>>1

2. Further, the user will be prompted to provide the directory path where the ligands in pdbqt format are present. After entering the directory path, Press <Enter> key.

   Enter the directory where ligands have been prepared
   >>>/home/bioinfo/Desktop/POAP/Tutorial/ligand/pdbqt

3. Next, the command prompt will display the number of ligands present and prompt the user to provide the directory path for protein and configuration text file containing the grid co-ordinates and other parameters present needed to be specified. **Note: Do not enter exhaustiveness value in the configuration file, it needs to be provided only when prompted during the execution of pipeline.** After entering the directory path Press <Enter> key.

   Enter the directory containing prepared protein and configuration file
   >>>/home/bioinfo/Desktop/POAP/Tutorial/VS/vina/



4. Next the command will prompt the user to provide the working directory path to perform the virtual screening process. **Note: Make sure that working directory is empty.**

   Enter the working directory
   >>>/home/bioinfo/Desktop/POAP/working/

5. In the next step, the user will be prompted to provide the exhaustiveness value for running the VINA job. Default is 8. After entering the value, Press <Enter> key.

   Enter the exhaustiveness for Autodock Vina
   >>>8

6. This will display the number of CPU threads available in the working system and prompt the user to provide the number of CPU threads to utilize to run single VINA job. The prompt will display the number of VINA jobs which will run in parallel mode. Accept the changes once confirmed by entering 1 and Press <Enter> key.
   24 number of CPU processors detected in your system

   Enter the number of CPU to be taken for running a single vina job:
   >>>8
   3 vina jobs will run in parallel

Enter
[1] To proceed
[2] To change again
>>>1

7. Further, the user will be prompted to provide the number of top complexes for which protein ligand complex coordinates in PDB format will be generated which can be further used for Visualization purposes. After entering the desired number, Press <Enter> key.

   ##Enter the number of ligand complex needed to be generated after VS##
   For the top hits
   >>>5

8. Finally, Press <Enter> key to confirm the start of the virtual screening process.

   Please enter return to start the virtual screening process
   >>>

Multiple protein virtual screening using VINA:

1. Start the script **bash POAP_vs.bash** in the bash terminal, this will display the different options to perform VS, enter 2 to perform multiple protein virtual screening using VINA and Press <Enter> key

   bash POAP_vs.bash -s
   POAP - Virtual screening Module:

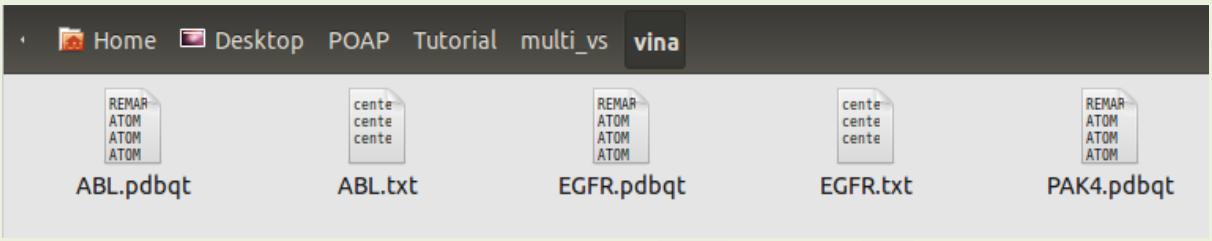
   Enter option
   [1] To perform VS with Autodock Vina
   [2] To perform Multiple Protein docking with Autodock Vina
   [3] To perform VS with Autodock
   [4] To perform Multiple protein docking with Autodock
   [5] To perform VS with AutodockZn
   [6] To perform Multiple protein docking with AutodockZn
   >>>2

2. Next, the user will be prompted to provide the directory path where the ligands in pdbqt format are present. After entering the directory path, Press <Enter> key.

   Enter the directory where ligands have been prepared
   >>>/home/bioinfo/Desktop/POAP/Tutorial/ligand/pdbqt
3. Further, on entering the ligand details, the command prompt will display the total number of number of ligands present. Next and the user will be prompted to provide the directory path for multiple proteins and respective configuration files, containing the grid co-ordinates and

other parameters. **Note: Do not enter exhaustiveness value in the configuration file. Protein coordinate file name and the configuration file name should be identical. After entering the directory path Press <Enter> key.**

Enter the directory containing prepared protein and configuration file
>>>/home/bioinfo/Desktop/POAP/Tutorial/multi_vs/vina/



4. Next the user will be prompted to enter the working directory path to perform the virtual screening process. **Note: Make sure that working directory is empty.**

   Enter the working directory
   >>>/home/bioinfo/Desktop/POAP/working/

5. In the next step, the user needs to provide the exhaustiveness for running the VINA job. Default is 8. After entering the number Press <Enter> key.

   Enter the exhaustiveness for Autodock Vina
   >>>8

6. Further, the command prompt will display the number of CPU threads available in the working system and prompt to provide the number of CPU threads to utilize to run single VINA job. For example 8 CPU's to utilize to run a vina job. The command interface will display the number of VINA jobs which will run in parallel mode. Accept the changes by feeding 1 and <Enter>

   Enter the number of CPU to be taken for running a single vina job:
   >>>8
   3 vina jobs will run in parallel
   Enter
   [1] To proceed
   [2] To change again
   >>>1

7. The user will be further prompted to provide the number of top complexes for which protein ligand complexes in pdb format needed to be generated towards using for graphical visualization. After entering the preferred number, Press <Enter> key.

##Enter the number of ligand complex needed to be generated after VS##
For the top hits
>>>5

8. Finally, Press <Enter> key to start the virtual screening process.

Please enter return to start the virtual screening process
>>>

Virtual screening using Autodock:

1. Start the script **bash POAP_vs.bash** and the prompt will display the different options to perform, enter 3 to perform virtual screening using Autodock and Press <Enter> key

bash POAP_vs.bash -s
POAP - Virtual screening Module:

Enter option
[1] To perform VS with Autodock Vina
[2] To perform Multiple Protein docking with Autodock Vina
[3] To perform VS with Autodock
[4] To perform Multiple protein docking with Autodock
[5] To perform VS with AutodockZn
[6] To perform Multiple protein docking with AutodockZn
>>>3

2. The prompt will ask you to provide the directory path where the ligands in pdbqt format are present. After entering the directory path Press <Enter> key.

Enter the directory where ligands have been prepared
>>>/home/bioinfo/Desktop/POAP/Tutorial/ligand/pdbqt

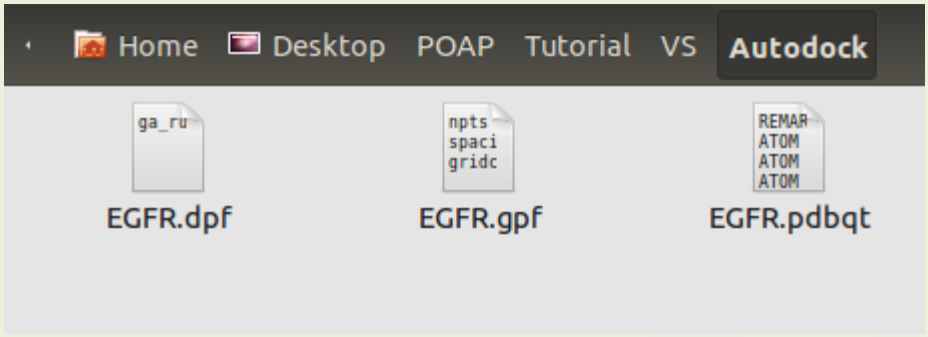
3. The prompt will now display the number of ligands present and will ask you to provide ligand database atom types or to calculate the atom types from the directory given. Give the respective option of interest and Press <Enter> key.

Enter
[1] To provide atom types for preparing map files
[2] To calculate atom types from the ligand directory
>>>2

4. The prompt will ask you to enter the directory path for protein and reference gpf, dpf file containing the grid co-ordinates and other docking parameters which can be specified in the

respective files. The parameters in these reference file will be taken as input to perform autogrid and autodock. After entering the directory path, Press <Enter> key.

Enter the directory containing prepared protein and configuration file
>>>/home/bioinfo/Desktop/POAP/Tutorial/VS/autodock



5. Next, the command will prompt and ask the user to provide the working directory path to perform the virtual screening process. **Note: Make sure that working directory is empty.**

   Enter the working directory
   >>>/home/bioinfo/Desktop/POAP/working/

6. The prompt will further display the number of CPU threads available in the working system and will ask you to provide the number of CPU threads to utilize to run autodock jobs in parallel. For example if 8 jobs entered then 8 autodock jobs will be run in parallel at a given instant. Press <Enter> key after entering the number of autodock jobs to run in parallel.

   Enter the number of jobs to be run in parallel: [0] to run as many parallel jobs as possible
   >>>24

7. Further, the prompt will ask you to provide the number of top complex to which protein ligand complex needed to be generated. After entering the numbers Press <Enter> key.

   ##Enter the number of ligand complex needed to be generated after VS##
   For the top hits
   >>>5

8. Finally, Press <Enter> key to start the virtual screening process for which prompt is asked.

   Please enter return to start the virtual screening process
   >>>

Multiple protein virtual screening using Autodock:

1. Start the script **bash POAP_vs.bash** and the prompt will display the different options to perform, enter 4 to perform virtual screening using Autodock and Press <Enter> key

   bash POAP_vs.bash -s
   POAP - Virtual screening Module:

   Enter option
   [1] To perform VS with Autodock Vina
   [2] To perform Multiple Protein docking with Autodock Vina
   [3] To perform VS with Autodock
   [4] To perform Multiple protein docking with Autodock
   [5] To perform VS with AutodockZn
   [6] To perform Multiple protein docking with AutodockZn
   >>>4

2. The prompt will ask you to provide the directory path where the ligands in pdbqt format are present. After entering the directory path Press <Enter> key.

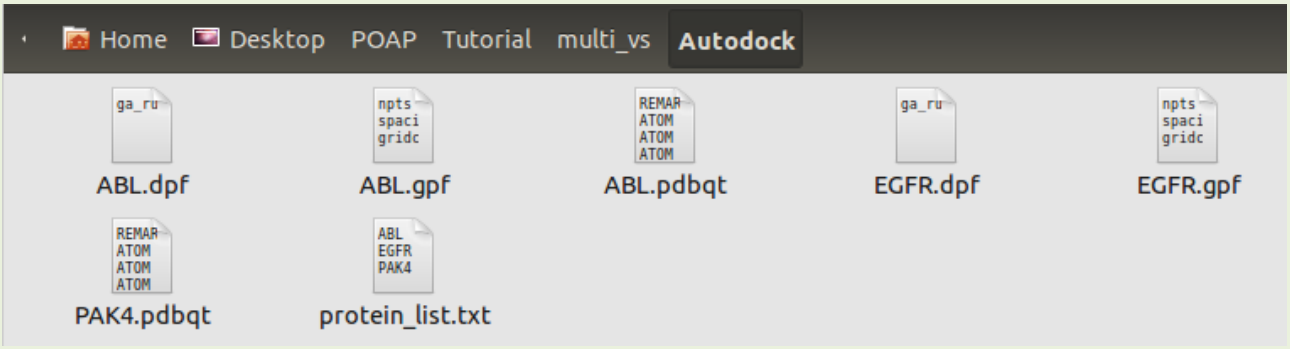
   Enter the directory where ligands have been prepared
   >>>/home/bioinfo/Desktop/POAP/Tutorial/ligand/pdbqt

3. The prompt will now display the number of ligands present and will ask you to provide ligand database atom types or to calculate the atom types from the directory given. Give the respective option of interest and Press <Enter> key.

   Enter
   [1] To provide atom types for preparing map files
   [2] To calculate atom types from the ligand directory
   >>>2

4. The prompt will ask you to enter the directory path for multiple protein and its respective reference gpf, dpf file containing the grid co-ordinates and other docking parameters which can be specified in the respective files. The parameters in these reference files will be taken as input to perform autogrid and autodock. After entering the directory path Press <Enter> key. **Note: Reference gpf and dpf file name should be similar to the protein file name.**

   Enter the directory containing prepared protein and configuration file
   >>>/home/bioinfo/Desktop/POAP/Tutorial/multi_vs/Autodock



5. Next, the command interface will prompt and the user to provide the working directory path to perform the virtual screening process. **Note: Make sure that working directory is empty.**

   Enter the working directory
   >>>/home/bioinfo/Desktop/POAP/working/

6. The prompt will display the number of CPU threads available in the working system and will ask you to provide the number of CPU threads to utilize to run autodock jobs in parallel. For example if 8 jobs entered, then 8 autodock jobs will be run in parallel at a given instant. Press <Enter> key after entering the number of autodock jobs to run in parallel.

   Enter the number of jobs to be run in parallel: [0] to run as many parallel jobs as possible
   >>>24

7. The prompt will further ask the user to provide the number of top complex to which protein ligand complex needed to be generated. After entering the numbers, Press <Enter> key.

   ##Enter the number of ligand complex needed to be generated after VS##
   For the top hits
   >>>5

8. Finally, Press <Enter> key to start the virtual screening process.

   Please enter return to start the virtual screening process
   >>>

Virtual screening using AutodockZn:

1. Start the script **bash POAP_vs.bash** and the prompt will display the different options to perform, enter 5 to perform virtual screening using AutodockZn and Press <Enter> key

   bash POAP_vs.bash -s
   POAP - Virtual screening Module:

   Enter option
   [1] To perform VS with Autodock Vina
   [2] To perform Multiple Protein docking with Autodock Vina
   [3] To perform VS with Autodock
   [4] To perform Multiple protein docking with Autodock
   [5] To perform VS with AutodockZn
   [6] To perform Multiple protein docking with AutodockZn
   >>>5

2. The prompt will ask you to provide the directory path where the ligands in pdbqt format are present. After entering the directory path, Press <Enter> key.

   Enter the directory where ligands have been prepared
   >>>/home/bioinfo/Desktop/POAP/Tutorial/ligand/pdbqt

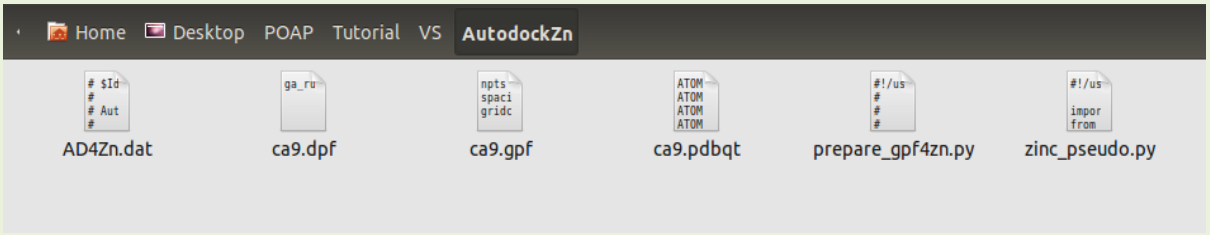
3. The prompt will now display the number of ligands present and will ask you to provide ligand database atom types or to calculate the atom types from the directory given. Give the respective option of interest and Press <Enter> key.

   Enter
   [1] To provide atom types for preparing map files
   [2] To calculate atom types from the ligand directory
   >>>2

4. The prompt will the user you to enter the directory path for protein and reference gpf, dpf file containing the grid co-ordinates and other docking parameters which can be specified in the respective files. The parameters in these reference file will be taken as input to perform autogrid and autodock. **Note: The forcefield parameter file, prepare_gpf4Zn.py and zinc_pseudo.py file should be present in the same directory containing proteins also.** After entering the directory path, Press <Enter> key.

   Enter the directory containing prepared protein and configuration file

   >>>/home/bioinfo/Desktop/POAP/Tutorial/VS/Autodockzn



5. Next, the command will prompt and ask you to provide the working directory path to perform the virtual screening process. **Note: Make sure that working directory is empty.**

   Enter the working directory
   >>>/home/bioinfo/Desktop/POAP/working/

6. The prompt will display the number of CPU threads available in the working system and will ask you to provide the number of CPU threads to utilize to run autodock jobs in parallel. For example if 8 jobs entered, then 8 autodock jobs will be run in parallel at a given instant. Press <Enter> key after entering the number of autodock jobs to run in parallel.

   Enter the number of jobs to be run in parallel: [0] to run as many parallel jobs as possible
   >>>24

7. Further, the prompt will ask user to provide the number of top complex to which protein ligand complex needed to be generated. After entering the numbers Press <Enter> key.

   ##Enter the number of ligand complex needed to be generated after VS##
   For the top hits
   >>>5

8. Finally, Press <Enter> key to start the virtual screening process.

   Please enter return to start the virtual screening process
   >>>

Multiple protein virtual screening using AutodockZn:

1. Start the script **bash POAP_vs.bash** and the prompt will display the different options to perform, enter 4 to perform virtual screening using AutodockZn and Press <Enter> key
   bash POAP_vs.bash -s
   POAP - Virtual screening Module:

   Enter option
   [1] To perform VS with Autodock Vina
   [2] To perform Multiple Protein docking with Autodock Vina
   [3] To perform VS with Autodock
   [4] To perform Multiple protein docking with Autodock
   [5] To perform VS with AutodockZn
   [6] To perform Multiple protein docking with AutodockZn
   >>>6

2. The prompt will ask you to provide the directory path where the ligands in pdbqt format are present. After entering the directory path Press <Enter> key.

   Enter the directory where ligands have been prepared
   >>>/home/bioinfo/Desktop/POAP/Tutorial/ligand/pdbqt

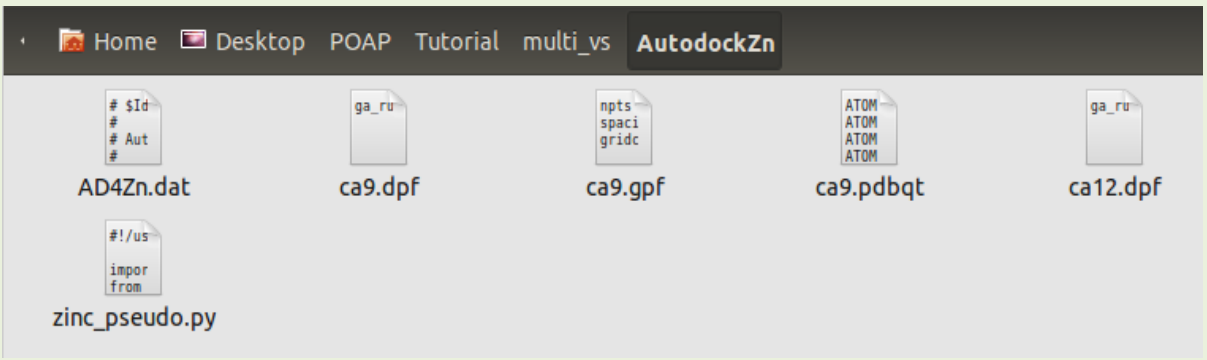
3. The prompt will now display the number of ligands present and will ask you to provide ligand database atom types or to calculate the atom types from the directory given. Give the respective option of interest and Press <Enter> key.

   Enter
   [1] To provide atom types for preparing map files
   [2] To calculate atom types from the ligand directory
   >>>2

4. The prompt will ask the user to enter the directory path for multiple protein and their respective reference gpf, dpf file containing the grid co-ordinates and other docking parameters which can be specified in the respective files. **Note: The parameters in these reference file will be taken as input to perform autogrid and autodock**. The forcefield parameter file and prepare_gpf4Zn.py zinc_pseudo.py file should be present in the same directory containing proteins also. After entering the directory path Press <Enter>**key. Note: Reference gpf and dpf file name should be identical to the protein file name.**

   Enter the directory containing prepared protein and configuration file

   >>>/home/bioinfo/Desktop/POAP/Tutorial/multi_vs/Autodockzn/



5. Next, the command will prompt and ask you to provide the working directory path to perform the virtual screening process. Note: Make sure that working directory is empty.

   Enter the working directory
   >>>/home/bioinfo/Desktop/POAP/working/

6. The prompt will display the number of CPU threads available in the working system and will ask you to provide the number of CPU threads to utilize to run autodock jobs in parallel. For example if 8 jobs entered then 8 autodock jobs will be run in parallel at a given instant. Press <Enter> key after entering the number of autodock jobs to run in parallel.

   Enter the number of jobs to be run in parallel: [0] to run as many parallel jobs as possible
   >>>24

7. The prompt will ask you to provide the number of top complex to which protein ligand complex needed to be generated. After entering the numbers Press <Enter> key.

   ##Enter the number of ligand complex needed to be generated after VS##
   For the top hits
   >>>5

8. Finally, Press <Enter> key to start the virtual screening process for which prompt is asked.

   Please enter return to start the virtual screening process
   >>>

**************************END**************************