**Ligand GA Usage Instructions**

Ligand GA requires one input, the configuration file. The input file has the parameters, paths, and filenames described in the Ligand GA Parameters document.

Ligand\_GA(Config\_File)

is the command to run Ligand\_GA in Matlab.

Several example Ligand GA Config files are given in the example directories of the download. After the externally used software paths to GOLD, Corina, and MGLTools are initially in place, the configuration file can be changed for the Ligand GA runs.

Multiple runs with the same InitialPopFile will generally give different sets of output molecules. The fitness function is complicated, depending on the molecule structure and interactions with the protein, and the nature of the GA will give different searches. Computing resources and initial population diversity can have an impact due to the number of mutation operations and atom types. Note that, as an example, the possibility of an atom mutating to another will generally favor higher valence atoms such as C or N and there is an inherent requirement for large sampling.

*GOLD\_config\_file*

The GOLD\_config\_file has to be created for the protein and docking parameters that is uses. This can be done using the CCDC HERMES software in the CSD-Enterprise package which GOLD is a component of. The GOLD\_config\_file is internally copied, parsed, and changed for each of the molecules in the population at each iteration. This file is exampled in 2 different contexts in the examples. The first uses default parameters for the GOLD GA. The second uses a restricted values in the POPULATION set:

popsize = 20

select\_pressure = 1.1

n\_islands = 5

maxops = 5000

niche = 2

This results in a much faster docking calculation than the default parameters created by HERMES at the cost of less search and accuracy. This is important for complicated molecules such as Simeprevir which require more docking computation and more stereoisomers. In the post processing of the output population of molecules more accurate docking calculation can be done on the molecules by using the default GOLD parameters or other parameters.

There are four files mentioned in the GOLD config file. The 1st is ligand\_data\_file which should be of the form

ligand\_dir/molecule1/molecule.001.mol2 10

with 10 being a changeable default number of poses. The file has to be specified in this manner with ligand\_dir the directory containing the protein information and output. The directory molecule1 is changed internally for each of the molecules at each iteration, but the initial config file has to be in this format even if the directory molecule1 does not exist – it will be created by the execution of Ligand GA.

The second file in the GOLD config file is directory

ligand\_dir/molecule1

and it has to be of this form.

The third file is concatenated\_output, which tells GOLD to produce a single file with the all of the docked pose output, in addition to single files for each pose,

ligand\_dir/molecule1/docked\_molecule\_file.001.mol2

This file also has to be in this format in the GOLD configuration file even if non-existent in ligand\_dir.

The last file is the protein\_datafile,

ligand\_dir/prepared\_protein.mol2

This is the protein mol2 file prepared for docking. HERMES can be used to prepare this file. An example protein preparation is protonation or adding some or all solvents. There are many possible preparations of the protein mol2 file as the GOLD config file shows. The documentation and tutorials of GOLD explains this in detail.

*InitialPopFile*

This is the initial population file used to start Ligand GA. It can be any set and number of molecules and is of the form of a column of SMILES expressions,

CC(C)CC1=CC=C(C=C1)C(C)C(O)=O

CC(C)CC1=CC=C(C=C1)C(C)C(O)=O

…

including an output population of a previous Ligand GA run. An effective GA shouldn’t use too small of a population for diversity and effectiveness of search. 100 is preferred and the size is limited by computing resources. Ligand GA will use all CPUs available in spawning a different docking process for each molecule at each iteration.

*OutFileName*

This is the output .dat Matlab file containing the population and fitness calculations at each iteration. Ligand\_GA\_Output\_Function is used to pull out unique molecules and sorts them according to docking scores. The file can be used in conjunction with databases of known molecules for structural and chemical purposes including comparison. The molecules can be inspected manually for further analysis or modification.

*Ligand\_dir*

This is the working directory of the protein ligand complex used by Ligand GA.

*Ligand GA Parameters in Ligand GA config file*

The various molecular and Ligand GA parameters in the configuration file are described in Ligand\_GA\_Parameters. Most are self-explanatory, but it has to be pointed out that an appropriate choice is required for the type of output wanted from Ligand GA. The mutation probabilities and atomic content can influence strongly the molecular evolution.

With the Ligand\_GA\_Config file, the software can be started with

Ligand\_GA(Ligand\_GA\_config)

The default is set for 10000 generations which means that apart from the other stopping criteria the program will continue to run indefinitely. The execution can be stopped and the population information of the run will reside in the OutFileName .dat file for further processing.

The software is also designed to run concurrent jobs if the ligand\_dir is different for the instances. In a single run, there are periods of low and high utilization of the resources. The mutation and crossover functions in the Matlab GA are a period of lower utilization although the GA is vectorized. In the fitness evaluation stage of Ligand GA all of the docking jobs are created and the software will check until GOLD is done processing these before continuation to the next iteration. This stage has high utilization given the number of docking jobs for the population and stereoisomers, and the complexity of docking.

Evaluation of the population in real-time is shown in a pop-up display of the best and mean fitnesses as a function of iteration. GA’s typically have a rapid improvement towards optimal solution with a leveling off and then slow improvement. The goal of Ligand GA is -not- to find an optimal solution, which is unlikely but theoretically possible, but rather finding sets of good solutions close to the user’s molecular criteria.