

# **A Genetic Algorithms approach for Drug Design**

**Lim Teck Sin**

**Pembroke College**



**Oxford University Computing Laboratory**

**Programming Research Group**

**8 - 11 Keble Road**

**Oxford, OX1 3QD**

*Thesis submitted in partial fulfillment of the requirements for  
the Degree of Master of Science in Computation*

**University of Oxford**

## FORWARD

The ability to design ligands that can bind targets specifically and selectively is an important goal in drug design. By studying the structures and considering various interactions of the molecules, computers can be used to predict de novo drugs.

One of the approaches utilises hydrophobicity scales to determine the optimum complementary peptide for a given target. Attempts have been made to use Genetic Algorithms (GA) to derive optimal peptides. The preliminary results indicate that the predictions can be used in protein purification [M Chung et al, unpublished results]. Although the molecular basis for interaction between a peptide and its complement are poorly understood, it seems that GA maybe useful in tackling protein-ligand interactions. (Chapter 2)

As the next step, it is proposed to move beyond complementary peptides (a purely two dimensional approach) into predicting three dimensional interactions between a protein and its ligand, using GA techniques.

Due to time constraint, it is proposed that the project will only involve the application of GA to minimise the structure of a ligand. The strategy involves generating a pool of the conformers which have identical chemical formula but different structural conformation. Each of these has a fitness based on energy calculations. The computation will involve van der Waals forces, bond lengths, bond angles and torsion angles and perhaps ionic charges as well. A more stable conformer with lower energy will have better fitness. By varying the conformations via crossover and mutation, and the propagation of fitter conformers, the programs should be able to predict conformers which are stable. The results obtained can then be verified by comparing with data obtained via crystallography database. (Chapter 5)

Before the above three dimensional computation can be done, a number of feasibility studies have to be performed:

- a. Investigate a suitable GA implementation to search for structures. Besides being limited by the quality of data, the results obtain will also depend a lot on the quality of the search performed.(Chapter 1, 2 and 4)

b. Investigate if GA can be successfully parallelised. A fast and parallel GA implementation will allow more experiments to be performed and may also improve the accuracy of the results. (Chapter 3)

The ability to minimise conformers properly should pave the way for the identification of suitable ligands to bind targets. It is envisaged that GA can be used to find the optimum target-ligand pairs by creating generations of minimised and identifying those having the best binding fitness with target. A successful parallel implementation of GA will help to accelerate the achieving of this goal.

## **ACKNOWLEDGEMENT**

I would like to thank the Center for Natural Products Research (CNPR) for its funding and interest in the project. The valuable comments of T Hiroshi on GA, R Miller on BSP, C S Low on 3D graphical libraries, Dr Ganesan, Dr T W Tan and Dr S Subbiah on biological aspects of molecular modelling are much appreciated. I am also grateful for the supervision provided by Dr S Muggleton.

# CONTENTS

<b>1.</b>	<b>GENETIC ALGORITHMS</b>	
1.1	THE GENETIC ALGORITHMS	1
1.2	REPRESENTATIONS OF GA	2
1.3	GA PARAMETERS	2
1.4	GA POPULATION	2
	1.4.1 Population Size	2
	1.4.2 Population Initialization	3
1.5	THE SIMILIARITY TEMPLATE THEORY	4
	1.5.1 Exploration and Exploitation	4
	1.5.2 Implicit Parallelism	4
1.6	SELECTION	4
	1.6.1 Fitness-proportional Selection	5
	1.6.2 Rank-protional Selection	5
	1.6.3 Population-elitist Selection	5
	1.6.4 Random Replacement Selection	6
1.7	CROSSOVER	6
	1.7.1 OnePoint Crossover	6
	1.7.2 Template or Uniform Crossover	6
	1.7.4 Why OnePoint Crossover in Nature?	7
1.8	MUTATION	7
1.9	SELECTION, CROSSOVER AND MUTATION	8
1.10	TERMINATING GA EXECUTION	8
1.1	SEGA	8
	1.11.1 Reorder	9
	1.11.2 HillClimbers	9
	1.11.3 HillCrossover	9
	1.11.4 HillMutation	10
	1.11.5 SEGA and CHC	10

<b>2.</b>	<b>GA and Complementary Peptides</b>	
2.1	COMPLEMENTARY PEPTIDES	12
2.2	COMPUTING COMPLEMENTARITY	13
2.3	GA AND COMPLEMENTARY PEPTIDES SEARCH	14
2.4	RESULTS	15
2.4.1	Computational Results	15
2.4.2	Biological experiment Results	16
	2.4.2.1 Site-1	16
	2.4.2.2 Site-2	17
2.5	DISCUSSION	17
2.6	IMPROVING MIXEDGA AND SEGA	20
2.6.1	Handling the objective function	20
2.6.2	Searching for the best combination of operators	21
2.6.3	Improving the algorithm	22
<b>3.</b>	<b>PARALLELISATION OF GA</b>	
3.1	BSP COMPUTATION MODEL	24
	3.1.1 Barrier synchronisation of the processors	24
	3.1.2 Partitioning of data	24
	3.1.3 Implementation of a master-slave paradigm	25
	3.1.4 BSP parameters	25
3.2	PARALLELISING GA	27
3.3	IMPLEMENTATION	28
	3.3.1 IntraPopulation Parallelisation	28
	3.3.2 BSP Cost Computation	29
3.4	RESULTS	29
3.5	DISCUSSION	30
3.6	IMPROVING THE PARALLEL GA	31
	3.6.1 InterPopulation Parallelisation	32
	3.6.2 Non-Generational and SEGA Parallelisation	33
	3.6.3 DGA	33
	3.6.4 Micro and Coarse-grained Parallelisation	34

<b>4.</b>	<b>BALANCING THE GA</b>	
4.1	LIMITATIONS OF TRADITIONAL GA	35
4.2	BIOLOGICAL GROUPINGS FOR GA	36
4.3	EXPLOITATION AND EXPLORATION GROUPINGS FOR GA	37
4.4	A CLASSIFICATION TABLE FOR GA	37
4.5	A SYSTEMATIC GA APPROACH	38
4.6	AUTOMATIC BALANCING	39
<b>5.</b>	<b>GA AND STRUCTURE-BASED DRUG DESIGN</b>	
5.1	THE DRUG DISCOVERY PROCESS	41
5.2	STRUCTURE-BASED DRUG DESIGN	41
5.3	THE FLEXIBLE LIGAND	42
5.4	THE ENERGY CALCULATIONS	42
5.5	APPROACHES FOR STRUCTURE-BASED DRUG DESIGN	43
5.6	GA AND STRUCTURE-BASED DRUG DESIGN	44
	5.6.1 PopulationGeneration	44
	5.6.2 HillCrossover	45
	5.6.3 Repair Operator	45
	5.6.4 HillMutation	45
	5.6.4 ReOrder	46
5.7	REPRESENTATION FOR THE MOLECULE	46
	5.7.1 LinkList Method	46
	5.7.2 Nearest-neighbour Representation	47
	5.7.3 Parallelising the Representation	48
	5.7.4 Structuring the Representation	48
5.8	HANDLING OF RESULTS	48
	<b>APPENDIX A - EXPERIMENTING WITH TGA AND SEGA</b>	
	Experiment 1	50
	Experiment 2	52
	Experiment 3	53
	Experiment 4	55

Experiment 5	57
Experiment 6	58
Experiment 7	59
Experiment 8	61
Experiment 9	63
Experiment 10	72
Experiment 11	74
APPENDIX B - GA FORMAL SPECIFICATIONS	85
REFERENCES	92