An Automated Ligand Evolution System using Bayesian Optimization Algorithm*

Masaharu Munetomo and Kiyoshi Akama
Information Initiative Center
Hokkaido University
North 11, West 5, Sapporo,
Hokkaido, Japan
munetomo@iic.hokudai.ac.jp

Haruki Maeda
Graduate School of Information Science and Technology
Hokkaido University
North 13, West 8, Sapporo,
Hokkaido, Japan
maeda@uva.cims.hokudai.ac.jp

Abstract: Ligand docking checks whether a drug chemical called ligand matches the target receptor protein of human organ or not. Docking by computer simulation is becoming popular in drug design process to reduce cost and time of the chemical experiments. This paper presents a novel approach generating optimal ligand structures from scratch based on *de novo* ligand design approach employing Bayesian optimization algorithm to realize an automated design of drug and other chemical structures. The proposed approach searches an optimal structure of ligand that minimizes bond energy to the receptor protein, and the structure of ligand is generated by adding small fragments of molecules to the base structure. The decision of adding fragments are controlled by Bayesian optimization algorithm which is considered as a promising approach in probabilistic model-building genetic algorithms. We have built a system that automatically generates an optimal structure of ligand, and through numerical experiments performed on a PC cluster, we show the effectiveness of our approach compared to the conventional approach using classical genetic algorithms.

Key–Words: automated drug design, ligand docking, screening, de novo ligand design approach, probabilistic model-building genetic algorithms, estimation of distribution of algorithms, Bayesian optimization algorithms

1 Introduction

To develop new medicines, drug industry needs to find promising chemical drug structures called ligands that will match the target receptor protein of an organ in a human body. Conventional approach in drug design called *screening* by *docking* checks matching between the molecule of the target organ and ligand chemicals taken from their database. The degree of matching is calculated by minimizing energy potential between the molecule and the ligand. Some ligand docking software packages such as AutoDock[1] and BioStation[2] have been developed to minimize energy potential between the ligand and the receptor protein. Simulated annealing (SA)[15] or other similar meta-heuristics are usually employed for the energy minimization.

We have developed a ligand docking system employing an estimation of distribution algorithm (EDA)[16] and SA elsewhere. However, simple energy minimization problem between the protein and ligands in a database is not so difficult; simulated annealing alone seems sufficient to have an optimal solution that minimizes the energy. Simple docking is not useful for researchers who try to find a new ligand structure which is not stored in the database in advance.

In order to support researchers who want to search for new ligand structures to be tested through chemical experiments, we propose a *de novo* ligand evolution framework that generates ligand structures by combining additional fragments to a base fragment. The optimization of the structure is realized by Bayesian Optimization Algorithm (BOA)[19, 20] which is a promising approach in EDAs that models probabilistic models of solutions based on Bayesian networks.

In this paper, we explain a classical approach to ligand docking in section 2., an overview of the *de novo* ligand design in section 3., an overview of Bayesian optimization algorithm we employ in section 4, and discuss our framework in section 5. The results of the numerical experiments are presented in section 6.

*This paper is an extended version of the paper "De Novo Ligand Evolution using Bayesian Optimization Algorithms", presented in the 10th WSEAS International Conference on Evolutionary Computing (EC'09).

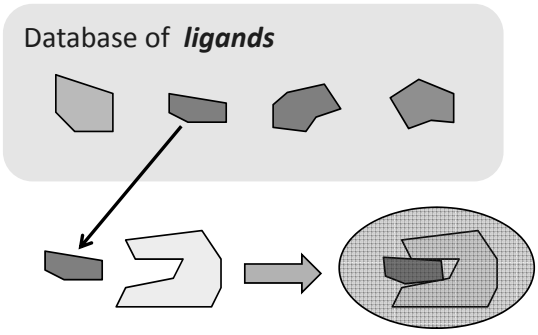

Figure 1: Ligand docking (screening)

2 Ligand docking and screening

In the drug design process, since it is expensive to test drugs through chemical experiments, computer simulations are employed to find candidates of chemical structures called ligands to be tested by the chemical experiments that may match the target protein structure. A conventional approach called *screening* simply checks whether ligands in their database matches the receptor protein by calculating minimal energy between both of them. Figure 1 illustrates an overview of screening process by ligand docking. Candidates of ligands are stored in their database in advance and for each ligand structure in the database, we check whether it matches the target receptor protein by minimizing the bond energy between them through computer simulations and optimizations.

Computer simulation cannot be perfect since we cannot avoid some errors in calculating bond energies. Therefore, the results obtained by the ligand docking are employed to reduce the number of chemical experiments performed by the chemist who want to find a promising structure of drug chemicals.

3 De novo Ligand Design

The conventional approach such as screening cannot find promising structure of ligands which are not stored in the database. *De novo* ligand design, on the other hand, constructs the structure of ligands by adding their fragments from scratch. Figure 2 shows an overview of the de novo ligand design[8].

First, an initial fragment is set, and then we generate structure of ligands automatically by adding fragments based on some heuristics. This approach generates ligand structure automatically, which can be used by the drug designers without having detailed information on the target receptor proteins.

The de novo ligand design approach consists of the following components.

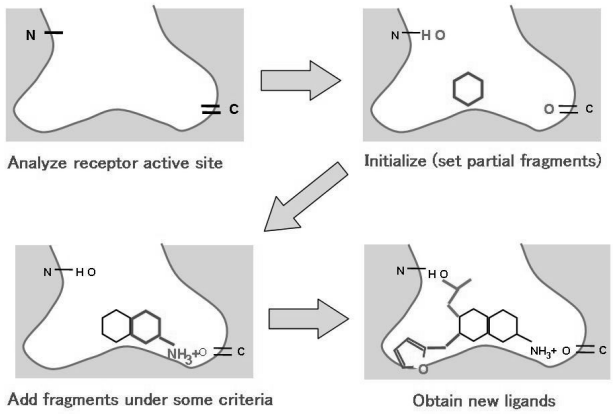

Figure 2: An overview of *de novo* ligand design

1. Evaluation of a candidate of ligand
2. Policy of adding fragments to the ligand
3. Optimization algorithm to control fragment addition to the ligand

Evaluation of a ligand is performed by calculating bond energy between the ligand and the target receptor protein. We can employ any ligand docking software packages for screening by evaluating the energy.

In order to generate good candidates of complex ligand structure, we need to control the policy of adding fragments to the ligand, that is, how to prepare and combine a variety of fragments, substructure of ligands.

Also, we need to choose an appropriate optimization algorithm for the energy minimization problem. The optimization problem to search an appropriate combination of fragments is a complex combinatorial problem which is classified into a NP-complete problem. Conventional optimization algorithms such as breadth-first, depth-first, Monte-Carlo methods, and evolutionary algorithms have been applied to solve this problem.

Glen et. al employs simple genetic algorithm to generate optimal structure of ligands from scratch[13]. This approach introduces mutation operators that add/delete fragments, change bond status, and so on.

4 Bayesian Optimization Algorithms

In the field of genetic and evolutionary computation, a series of advanced algorithms employing probabilistic models have been developed since classical approach

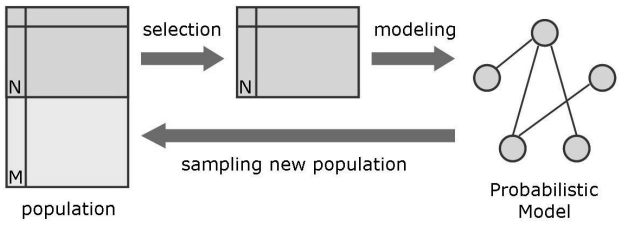

Figure 3: Estimation of Distribution Algorithm (EDA)

such as simple genetic algorithms cannot solve some problems where we cannot ensure tight linkage in the encoded strings.

Instead of applying genetic operators in genetic algorithms (GAs), estimation of distribution algorithms (EDAs) estimate distribution of alleles in a population of strings to build a probabilistic model of promising solutions, and the model is employed to generate offspring for the next generation. Fig.3 shows an overview of EDA.

Bayesian optimization algorithm (BOA) is a promising approach in EDAs, which employs Bayesian networks as its probabilistic model in order to model dependencies among alleles. Bayesian network is a directed acyclic graph that represents dependencies as conditional probabilities. The probabilistic model employed in the BOA represents a distribution of conditional probabilities as follows:

$$p(X) = \prod_{i=1}^{n} p(X_i|\Pi_{X_i}) \tag{1}$$

where $X = (X_1, \cdots, X_n)$ is a vector of random variables, each of which represents the probability of 1's occurrence, and Π_{Xi} is a list of variables that become *parents* of variable X_i.

The model-building process of the BOA searches an optimal strucure of Bayesian network that maximizes the Bayesian-Dirichlet (BD) metric defined as follows which measures a quality of the newtork with respect to the data set of the current promising solutions.

$$p(D, B|\xi) = p(B|\xi) \prod_{i=0}^{l-1} \prod_{\pi_{X_i}} \frac{m'(\pi_{X_i})!}{(m'(\pi_{X_i}) + m(\pi_{X_i}))!} \times \prod_{x_i} \frac{(m'(x_i, \pi_{X_i}) + m(x_i, \pi_{X_i}))!}{m'(x_i, \pi_{X_i})!} \tag{2}$$

where $p(B|\xi)$ is a prior probability of network B, D is the population of promising solutions, $m(\pi_{X_i})$ is the number of instances where $\prod_{X_i} = \pi_{X_i}$ in D, and $m(x_i, \pi_{X_i})$ is the number of instances where $X_i = x_i$ and $\prod_{X_i} = \pi_{X_i}$ in D.

The BOA performs the following algorithm:

1. A population of N strings are initialized randomly.
2. M strings ($M < N$) are selected from a population of N strings based on their fitness values.
3. A Bayesian network is generated that maximizes the BD metric. A greedy search heuristics is employed to find an optimal structure of the network.
4. N strings are generated as offspring of the next generation based on the probabilistic model of the Bayesian network.
5. Goto 2. unless some termination criterion is satisfied.

A greedy search is performed as follows to find a Bayesian networks that maximize the BD metric

1. Initial network B is initialized consisting only of nodes without any link.
2. A pair of nodes are randomly chosen and an edge is added to the network B that connects the pair to generate B'.
3. If B' has any cycle, the added edge is discarded and go to 2.
4. Calculate the metrics of B and B', and replace B by B' if the metric of B' is larger than that of B.
5. Go to 2. unless a terminate condition is satisfied.

By employing Bayesian networks, the BOA can solve wide spectrum of problems with complex interaction among genes.

5 De novo Ligand Evolution using Bayesian Optimization Algorithm

De novo ligand evolution framework that we propose employs Bayesian optimization algorithm to obtain optimal structure of ligands from a large number of possible structure consisting of a variety of possible fragments. Fig.4 illustrates an overview of our approach to realize de novo ligand evolution.

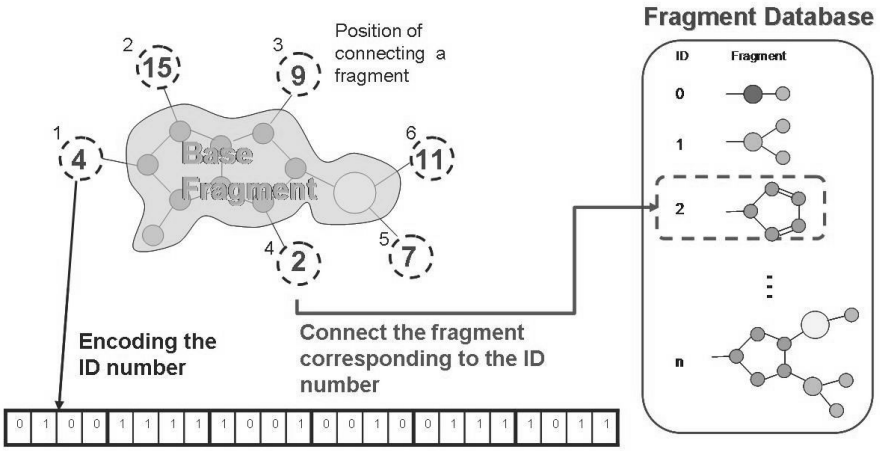

Figure 4: An overview of the proposed framework

5.1 Overview of the Proposed System

The proposed ligand evolution system consists of the following three major components:

- Ligand Construction Module
- Ligand Evaluation Module
- Optimization Module

The input to the system is the structure of the target receptor protein, the base fragment, and the set of additional fragments, which are described by the PDB (Protein DataBase) format[3]. The objective of the system is to find a candidate structure of ligand that minimizes the bond energy between the ligand and the target receptor protein and also maximizes the number of atoms of the ligand since we are not interested in smaller structure of ligands which are already stored in the current database of ligands.

Inside the system, each ligand structure is encoded into a binary strings, and the ligand construction module decodes the string to generate structure of ligands by following the process of adding fragments to the base fragment as in section 5.4.

The generated ligand is evaluated by the ligand evaluation module. The module employs BioStation Dock to calculate bond energies between the ligand and the receptor protein. We input the strutures of them with PDB format, and then BioStation Dock generates candidates of active sites and performs replica exchanging, a variation of simulated annealing, to find a configuration that minimize the energy. Fitness values are calculated by the obtained value of the energy and the number of atoms of the ligand generated.

Based on the fitness values, the optimization module selects a set of promising solutions, which are employed to generate Bayesian networks. According to the probabilistic model as in the Bayesian networks, we generate offspring of the ligand structures for the next generations. After convergence, we can obtain a candidate of ligand structure that minimizes the bond energy to the target receptor protein.

5.2 Ligands and Fragments

A ligand consists of a base fragment and additional fragments. Figure 4 illustrates an overview of the ligand construction and its encoding. A base fragment is a basis of the ligand structure, which is selected from molecules existent in natural chemicals. The base fragment has several candidates of connectors, where additional fragments can be connected to extend the structure of the ligand. In the proposed framework, the positions of *H* (hydrogen) atom on the base fragment is set as the connectors, which can be replaced by additional fragments.

An additional fragment is a sub-structure of ligands which can be connected to the positions of the *H* atoms on the base fragment. In the proposed system, we store several candidates of the additional fragments in the database, which is to be selected in generating structure of ligands.

Structures of ligands and receptor proteins are stored by PDB format in their database. Fragments of ligand structure are stored following the format as shown in figure 5. The fragment database consists of entries to specify fragments (@FRAG) and their connections (@CONNECT).

```
fragment database
@FRAGNUM # of fragments
@FLAG fragment_ID fragment_filename
@FLAG fragment_ID fragment_filename
    :
@CONNECT fragment_ID
  atom_ID_to_connect_base_fragment
  atom_ID_to_be_removed
  connection_ID
@CONNECT fragment_ID
  atom_ID_to_connect_base_fragment
  atom_ID_to_be_removed
  connection_ID
    :
```

Figure 5: Fragment database format

5.3 String Representation

Each string represents a configuration of fragments added to the base fragment. As in figure 4, we encode ID numbers assigned in the fragment database into binary substrings. When we set the number of positions N_{base} where additional fragments can be added, and the number of possible additional fragments as N_{add}, the length of strings is calculated as follows:

$$l = N_{base} \times \log_2 \lceil N_{add} + 1 \rceil \tag{3}$$

where $\lceil N_{add} + 1 \rceil$ means the number of possible additional fragments (N_{add}) plus that of H atom without addition (1).

5.4 Adding Fragments

Figure 6 illustrates the process of adding a fragment to the base fragment. When we add a fragment, rotation of the additional fragment is set to determine the degrees to the base fragment, and also, distance between the additional and the base fragments is calculated according to the database of bond distances.

In the rotation process, we calculate degrees of rotation θ_{rot} and ϕ_{rot} from those of the base fragment θ_{base}, ϕ_{base} and those of the additional fragment θ_{frag}, θ_{frag} as follows:

$$\theta_{rot} = \theta_{base} - \theta_{frag} \tag{4}$$

$$\phi_{rot} = \phi_{base} - \phi_{frag} \tag{5}$$

Employing the degrees calculated as above, we rotate atoms in the additional fragment. First, we perform rotation around y-axis as follows:

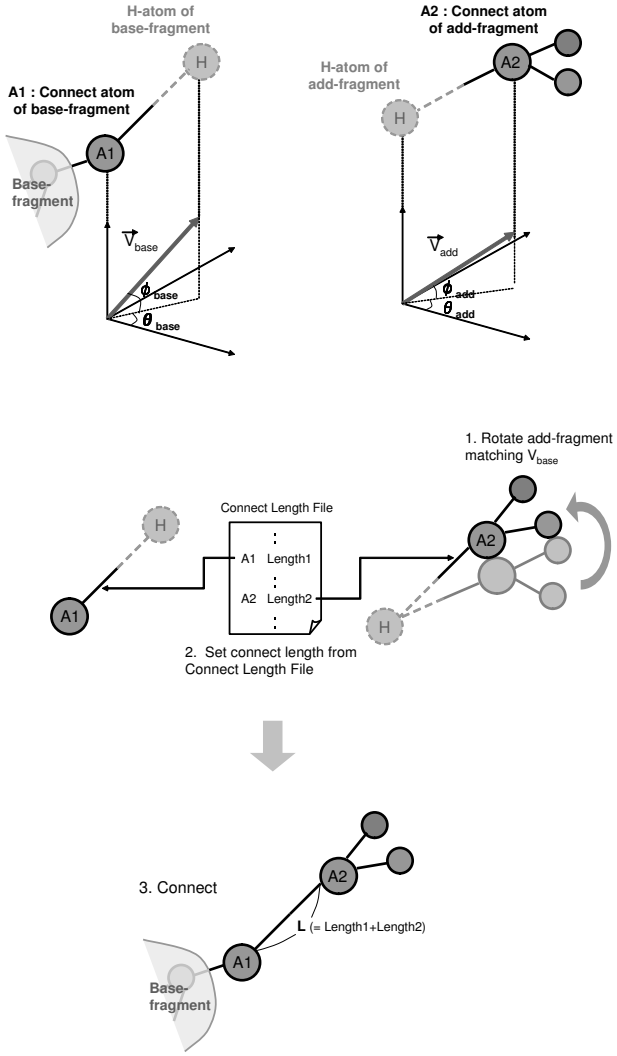

Figure 6: Process of adding fragments

$$\begin{aligned} x' &= x\cos(\phi) - z\sin(\phi) \\ y' &= y \\ z' &= x\sin(\phi) + z\cos(\phi) \end{aligned} \tag{6}$$

Next, rotation around z-axis is performed as follows to obtain final results:

$$\begin{aligned} x_{rot} &= x'\cos(\theta) + y'\sin(\theta) \\ y_{rot} &= -x'\sin(\theta) + y'\cos(\theta) \\ z_{rot} &= z' \end{aligned} \tag{7}$$

After the rotation is finished, we calculate bond distances based on the database of the distances empirically calculated as in the table 1.

In the distance database, average distances between orbital of the atoms are stored. In connecting the fragments, an atom from the base fragment and another one from the additional fragment are selected and their distance is calculated based on the database. For example, when we connect two C-sps, distance for $C.3$ is selected from the database and the distance between a $C.3$ and another $C.3$ is calculated by $0.771 \times 2 = 1.542$.

Table 1: An example of database of bond distances

C.3	0.771	O.co2	0.596
C.2	0.668	N.3	0.740
C.1	0.602	N.2	0.676
C.ar	0.967	N.1	0.689
O.3	0.715	N.am	0.661
O.2	0.615	N.pl3	0.736
O.ar	0.604	N.4	0.739

5.5 Evaluations

We employ Biostation Dock[2] to calculate bond energy between the ligand and the receptor protein. Biostation performs minimization of the bond energy by controlling the position of the ligand (x, y, z), the angles of bonding $(\theta_x, \theta_y, \theta_z)$, and dihedral angles $(\phi_1, \cdots, \phi_n)$ that determine the structure of the ligand.

To calculate the energy, XUFF is employed which considers modified charge equilibration (MQEq) and universal force field (UFF). The XUFF is based on the following equation:

$$E_{XUFF} = E_{UFF} + E_{MQEq} + E_{vdW} + \Delta G_{GB/SA} \quad (8)$$

$$E_{UFF} = E_{bond\_strech} + E_{angle\_bend} + E_{torsion} \quad (9)$$

where E_{UFF} is the energy with UFF consisting of bonding stretch energy between atoms E_{bond_strech}, energy with variable angle of bending E_{angle_bend} and that with torsion of dihedral angles $E_{torsion}$. E_{MQEq} represents energy calculated by MQEq method, E_{vdW} is energy based on van der Waals potentials, and $\Delta G_{GB/SA}$ is energy that considers the solvent effects with the GB/SA (Generalized Born/Surface Area) method.

Energy minimization procedure in BioStation employs a variation of simulated annealing (SA), a local search meta-heuristics that allows acceptance of solutions that degrades the solution by a perturbation as in the following probability.

$$accpt(x') = \begin{cases} 1 & \Delta E < 0 \\ \exp(-\frac{\Delta E}{T}) & \text{otherwise} \end{cases} \quad (10)$$

where ΔE is the difference of the energy E by a perturbation of solution from x to x'. Biostation Dock employs the replica exchange method that performs SA threads in parallel, and each SA thread exchanges its replica of current solutions to the other SA thread. By exchanging solutions, it increases robustness of solution candidates obtained by the system.

The purpose of ligand evolution is to find candidates of ligand structures to be tested by the actual experiments. We evaluate fitness values of the structures according to the following policies:

1. We assign higher fitness values to the structure consisting of more atoms. This is because we want to find complex ligand structures which are not stored in the database in advance

2. We assign low fitness values to the structures with energy lower than a threshold that is necessary to be *docked* to the receptor proteins.

Therefore, overall fitness function of the ligand is calculated by the following equation:

$$f(S) = w_e f_e(E_{XUFF}) + w_m f_m(N), \quad \text{if } E_{XUFF} < E_{th} \quad (11)$$

$$f(S) = w_e f_e(E_{XUFF}), \text{ otherwise,} \quad (12)$$

where E_{th} is the threshold value of energy, w_e is the weight to the energy fitness function f_e, and w_m is the weight to the fitness function f_m of the number of atoms N in the ligand structure S.

5.6 Application of Bayesian Optimization

De novo ligand evolution framework that we propose performs the following algorithm to a population of binary strings which encode ligand structures as in figure 4:

1. A base fragment and a set of additional fragments are generated and stored in a database.

2. An initial population of strings is generated randomly.

3. Evaluations of strings:

(a) For each string in a population, fragments are selected from a set of additional fragments based on the encoded string and are added to the base fragment to generate a ligand.

(b) Calculate energy of the generated structure of the ligand using docking software.

(c) Calculate the fitness value of the ligand based on the energy and the number of atoms.

4. Select the ligands based on the calculated fitness values.

5. Construct a Bayesian network based on the selected individuals of ligands.

6. Generate offsprings based on the Bayesian network and the population of current strings.

7. Go to 3 . unless a termination criterion is satisfied

5.7 Parallel implementation

Since calculation of the bond energy with BioStation Dock needs a considerable computation time, we need to seek for parallel implementation of the system. We employ MPI (Message Passing Interface)[4] to parallelize the calculation of energy to evaluate fitness values. We perform the following sequence in the parallel environment such as PC clusters with a shared file system such as NFS (Network File System).

1. A root node generates a population of strings (ligands)

2. In the root node, register filenames of the ligands for each strings

3. The root node distributes the information on the ligands and the receptor proteins to the other nodes.

4. Each node executes BioStation Dock in parallel to obtain output of the minimum energy between them.

5. The root node collects the results from the other nodes in the cluster.

6. The root node evaluates fitness values and performs one generation of Bayesian optimization algorithm to generate strings for the next generation.

7. go to 1. unless termination condition is not satisfied.

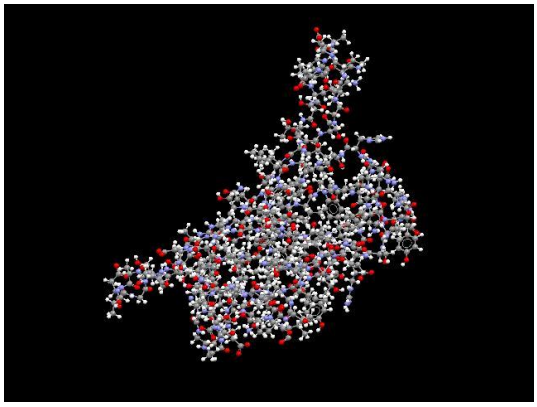

Figure 7: The target receptor protein

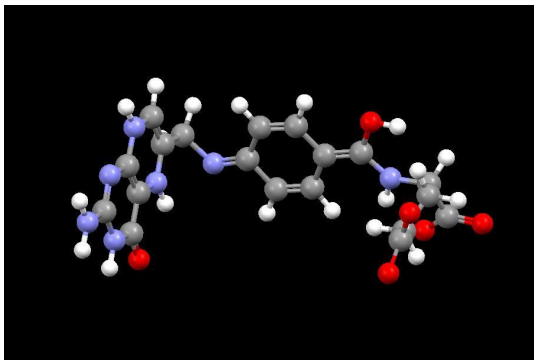

Figure 8: The base fragment

6 Empirical Results

We perform numerical experiments to illustrate the effectiveness of the proposed framework compared with conventional approach employing simple genetic algorithms. We employ a small PC cluster consisting of 12 computing nodes (IBM x3455 with Dual-Core Opteron 2210) to perform fitness evaluations in parallel by using MPI, since energy calculation with BioStation Dock is computational expensive which usually needs several hours to evaluate a single fitness.

We try to generate ligand structure to be *docked* to the target receptor protein in figure 7. We employ fol molecule in figure 8 as the base fragment which has 19 H atoms.

Figure 9 illustrates a set of additional fragments which are employed in the following experiments. Since additional fragments can be connected to the base ligand by replacing its H atoms, the length of the string becomes $19 \times \lceil \log_2(7+1) \rceil = 57$ from equation (2).

Population size is set to 120 and we perform experiments for 20 generations. The parameters for evaluations are as follows: $E_{threshold} = 10000$, $w_e =$

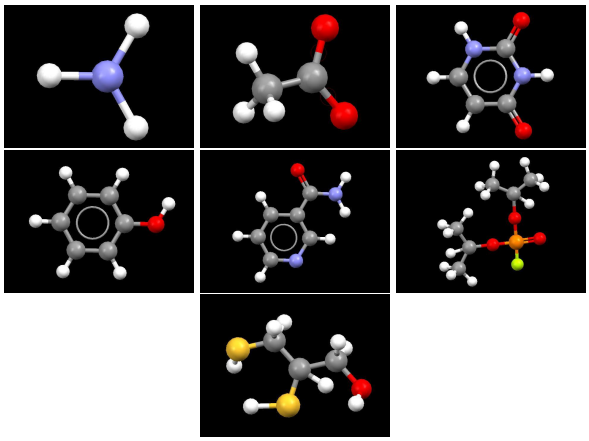

Figure 9: The set of additional fragments

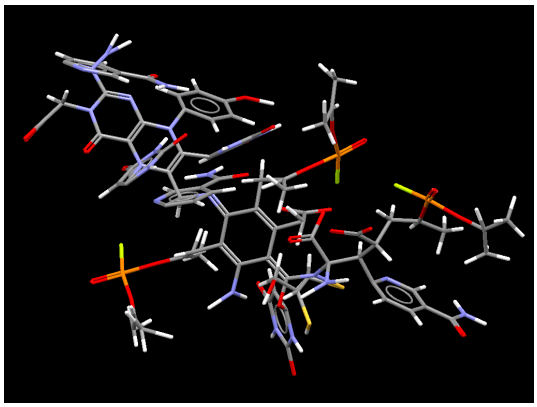

Figure 10: Structure of the initial ligand molecule

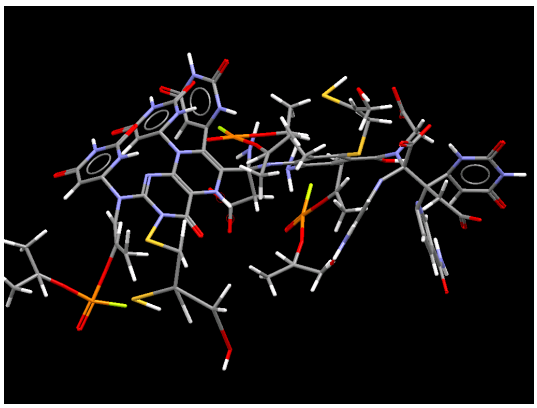

Figure 11: Best structure of ligand in the 12th generations

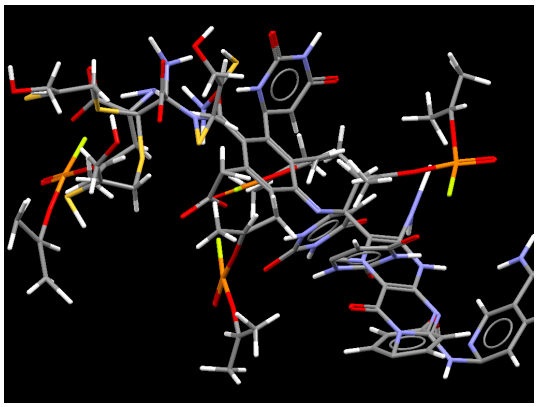

Figure 12: Best structure of ligand in the 14th generations

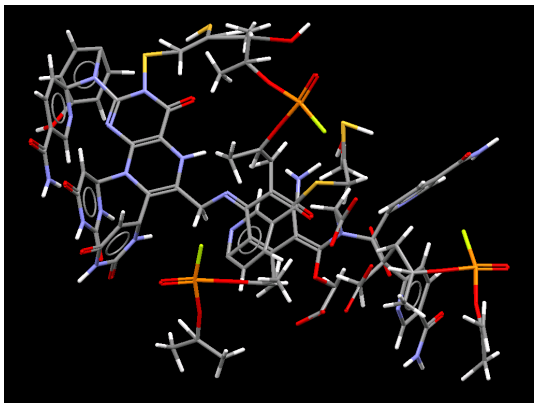

Figure 13: Best structure of ligand in the 18th generations

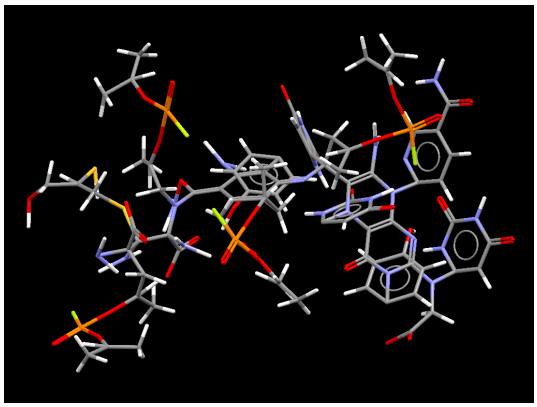

Figure 14: Best sturcture of lingand obtained after the 20th generations

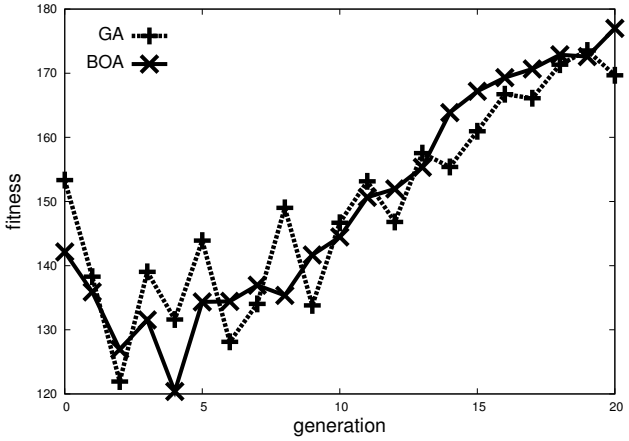

Figure 15: A comparison of the average fitness values

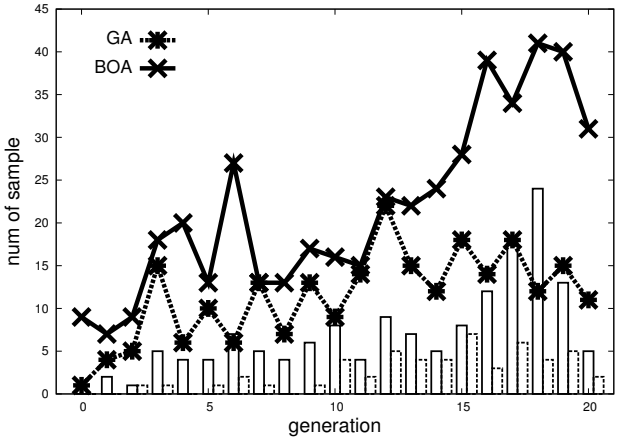

Figure 16: The number of candidates of ligand structures obtained

0.001, $w_m = 1.0$, $f_e(x) = x$ and $f_m(x) = x$. Simulated annealing (SA)[15] is performed inside Biostation Dock that perform local search to minimize structural energies. The parameters of is as follows: initial temperature $T_0 = 1.0$ and cooling schedule is $T_i = 0.9 \times T_{i-1}$.

Figure 10 shows the initial structure. Figure 11, 12, and 13 show the solution obtained in the 12th, 14th and 18th generations. The final solution obtained after the 20th generations is in figure 14. The energy is greatly decreased from 6639.9 (initial) to -1168, and the number of atoms is increased by 191 from the base fragment. Overall fitness value of the obtained result is 192.2.

We perform experiments to compare the proposed algorithm to the ligand evolution employing a simple GA. The parameters of the GA are as follows: population size $N = 120$, one-point crossover is performed with the probability $P_c = 1.0$ and the probability of mutation is $P_m = 0.05$.

Figure 15 shows a comparison of average fitness values in a population of strings between the proposed algorithm employing Bayesian optimization and that employing a simple GA. The x-axis shows the number of generations performed and the y-axis shows the average fitness values in a population. This figure shows that the proposed method employing BOA achieves better fitness values than that employing GA which shows some instability of fitness in the earlier generations.

Figure 16 shows a comparison of the number of allowable and excellent solutions between the proposed algorithm employing Bayesian optimization and that employing a simple GA. The x-axis shows the number of generations and the y-axis shows the number of instances. The lines show the number of allowable solutions with energy less than 10,000 and the boxes shows the number of excellent solutions with negative energy value. Solid lines and boxes show the numbers by the proposed approach and dashed lines and boxes show those by a simple GA.

The result shows that the proposed method can obtain around three times as much as allowable solutions and around five times as much as good solutions than those with conventional approach which employs simple GAs. Since the objective of the ligand evolution is to find good candidates of ligand structure, the result here apparently illustrates the effectiveness and robustness of our approach employing Bayesian optimization.

7 Conclusion

The paper presents a novel approach in generating structure of ligands automatically to reduce cost and time of experiments in the drug design process. The proposed approach is based on de novo ligand design approach that constructs new ligand structures by combining additional fragments to the base fragment. We introduce Bayesian optimization algorithms, a promising approach in probabilistic model-building genetic algorithms to search optimal structure of ligands considering complex interaction among additional fragments.

Through numerical experiments performed on a cluster of PCs, we show the effectiveness of the proposed framework; that is, the proposed algorithm employing BOA can generate promising ligand structures automatically and the number of good structure of ligands is much larger than that employing classical GAs. Although the obtained results should be checked

by chemists through chemical experiments, the framework proposed here can reduce the number of chemical experiments by presenting a set of promising structure of ligands with lower bond energy estimated to the target receptor proteins.

Acknowledgements: The research was supported by the Grant-in-Aid for Young Researchers, Ministry of Education, Culture, Sports, Science and Technology, Japan. (grant No.18700219).

References:

[1] AutoDock: http://autodock.scripps.edu/

[2] Biostation, Revolutionary Simulation Software (RSS): http://www.ciss.iis.u-tokyo.ac.jp/rss21/en/theme/life/synergy/index.html

[3] Worldwide Protein Data Bank (PDB): http://www.wwpdb.org/

[4] Message Passing Interface (MPI) Forum: http://www.mpi-forum.org/

[5] Koki Tsukamoto, Yutaka Akiyama. Automated Docking of NADH to the Active Site of Nitric Oxide Reductase from Fusarium Oxysporum and Semi-empirical Calculations of the Electron Transfer Mechanism and the Hydrogen-bonding Network, WSEAS Transactions on Computers, 5(8):1701-1706 (2006)

[6] Shih-Ching Ou, Chun-Yen Chung , Hung-Yuan Chung, Wen-Tsai Sung, Chia-Chih Tsai, Cheng-Chih Chien , Da-Yu Su , Shi-Yong Lin. Molecular Docking for Protein Folding Structure and Drug-likeness Prediction, WSEAS Transactions on Biology and Biomedicine, 2(1):57-63 (2005)

[7] Shih-Ching Ou, Chun-Yen Chung, Wen-Tsai Sung, Chia-Chih Tsai, Chin-Chih Chien, Da-Yu Su. Virtual Screening and Computer-Aided Drug Design in Molecular Docking Via Lyapunov Function, WSEAS Transactions on Biology and Biomedicine, 1(4):384-389 (2004)

[8] Gisbert Schneider and Uli Fechner. Computer-based, de novo design of drug-like molecules. Nature Reviews Drug Discovery, 4(8):649-663 (2005)

[9] Zhaowen Luo, Renxiao Wang, and Luhua Lai. Rasse: A new method for structure-based drug design. Journal of Chemical Information and Computer Sciences, 36(6):1187-1194 (1996)

[10] Yoshihiko Nishibata and Akiko Itai. Automatic creation of drug candidate structures based on receptor structure. starting point for artificial lead generation. Tetrahedron, 47(43):8985-8990 (1991)

[11] Valerie J. Gillet, A. Peter Johnson, Paulina Mata, Sandor Sike, and Philip Williams. Sprout: A program for structure generation. Journal of Computer-Aided Molecular Design, 7(2):127-153 (1993)

[12] M.B. Eisen, D.C. Wiley, M. Karplus, and R.E. Hubbard. HOOK: a program for finding novel molecular architectures that satisfy the chemical and steric requirements of a macromolecule binding site. Proteins, 19:199-221 (1994)

[13] Robert C. Glen and A. W. Rayne. A genetic algorithm for the automated generation of molecules within constraints. Journal of Computer-Aided Molecular Design, 9(2):181- 202 (1995)

[14] Dimitris Dimitropoulos, Kim Henrick, Jawahar Swaminathan, Adel Golovin. Analytical Processing of the PDB on the Web: Examining Ligand Fragments and their Environment, WSEAS Transactions on Biology and Biomedicine, 3(6):414-420 (2006)

[15] S. Kirkpatrick, Jr. Gelatt, C. D., and M. P. Vecchi. Optimization by Simulated Annealing. Science, 220(4598):671-680 (1983)

[16] Pedro Larraanaga and Jose A. Lozano. Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation. Kluwer Academic Publishers, Norwell, MA, USA (2001)

[17] Martin Pelikan, David E. Goldberg, Fernando Lobo, A survey of optimization by building and using probabilistic models. Computational Optimization and Applications, 21(1):5-20 (2002)

[18] Judea Pearl. Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1988)

[19] Martin Pelikan, David E. Goldberg, and Erick Cantu-Paz. BOA: The Bayesian optimization algorithm. Proceedings of the 1999 Genetic and Evolutionary Computation Conference, 525-532 (1999)

[20] Martin Pelikan, David E. Goldberg, and Erick Cantu-Paz. Linkage Problem, Distribution Estimation, and Bayesian Networks. Evolutionary Computation, 8(3):311-340, MIT Press (2000)