PCCP

Accepted Manuscript

ROYAL SOCIETY OF CHEMISTRY

www.rsc.org/pccp

Journal Name

RSCPublishing

ARTICLE

Protein-ligand docking using fitness learning-based artificial bee colony with proximity stimuli

Cite this: DOI: 10.1039/xoxxoooooox

Shota Uehara*, Kazuhiro J. Fujimoto* and Shigenori Tanaka*,

Received ooth January 2012,
Accepted ooth January 2012

DOI: 10.1039/xoxxoooooox

www.rsc.org/

Protein-ligand docking is an optimization problem which aims to identify the binding pose of a ligand with the lowest energy in the active site of a target protein. In this study, we employed a novel optimization algorithm called fitness learning-based artificial bee colony with proximity stimuli (F*l*ABCps) for docking. Simulation results revealed that F*l*ABCps improved the success rate of docking, compared to four state-of-the-art algorithms. The present results also showed superior docking performance of F*l*ABCps, in particular for dealing with highly flexible ligands and proteins with a wide and shallow binding pocket.

Introduction

Protein-ligand docking plays an essential role for structure-based drug design (SBDD), which aims to identify the binding structure of a ligand with the high affinity to a target protein using computer simulation. In lead identification, virtual screening based on docking simulation enables us to perform more efficient drug screening than experimental high throughput screening (HTS) in terms of cost and efficiency.[1] Also in lead optimization, successful docking leads to a rational molecular design based on the three-dimensional structure of a target protein and binding ligands.[2] Incorporating SBDD, a number of drugs have been successfully developed.[3-5]

Protein-ligand docking is regarded as an optimization problem, which identifies the binding pose of a ligand with the lowest energy in an active site of a target protein. Various scoring functions have been developed for an accurate calculation of binding affinity.[6-8] Energy landscapes of the scoring functions are usually complicated and exhibit rugged funnel shape.[9] Hence, successful docking simulations require an efficient optimization algorithm. Inefficient optimization algorithms often give solutions trapped in some local optimum points of a scoring function, which results in an incorrect binding pose of a ligand and a wrong estimation of the binding affinity. In particular, highly flexible ligands with many rotational bonds are known to be more difficult for the docking simulation, due to their large number of optimization parameters.[10]

Various optimization algorithms have been developed for the protein-ligand docking. Genetic algorithm (GA) based approaches are the most general, which are implemented, *e.g.*, in GOLD[11] and AutoDock[12]. In addition, some variants of particle swarm optimization (PSO)[13] have been developed, such as SODOCK[10] and PSO@AutoDock[14]. It was reported that the PSO based approaches improve the docking accuracy better than GA. Both GA and PSO quickly find the global optimum point for a simple problem, because of their high convergence ability. However, these algorithms potentially have the risk of premature convergence to some local optimum point, in particular for the multi-modal, non-convex or highly multi-dimensional problems.[15,16] In this meaning, more efficient optimization algorithm is strongly required for protein-ligand docking.

In this study, we attempted to apply a novel optimization algorithm, called fitness learning-based artificial bee colony with proximity stimuli (F*l*ABCps)[17], to the protein-ligand docking. Artificial bee colony (ABC) algorithm[18] is a simple and powerful optimization algorithm for the multi-dimensional and multi-modal functions, inspired from intelligent behaviors of honey bee swarm. It has been reported that the ABC based algorithms give better results for various optimization problems than the conventional algorithms.[19-21] F*l*ABCps is a variant of the ABC algorithm, extending its applicability to more complicated optimization problems like the protein-ligand docking.

The docking performance of F*l*ABCps was examined in comparison with four state-of-the-arts algorithms: ABC, SODOCK, PSO and LGA. Lamarckian genetic algorithm (LGA)[22] is a variant of GA, which is implemented in AutoDock as a default algorithm. The present results revealed that F*l*ABCps improved the success rate of the docking compared to the other algorithms, in particular for highly flexible ligands with many optimization parameters. In addition, we analyzed the relationship between the structure of the binding pocket and the energy landscape of the scoring function. This analysis clearly showed that F*l*ABCps is a

suitable algorithm for dealing with receptor proteins which have a wide and shallow binding pocket.

Methods

Classical artificial bee colony algorithm

The artificial bee colony (ABC) is a swarm based meta-heuristic algorithm proposed by Karaboga et al.[18] for numerical optimization problems. It was inspired from the intelligent foraging behavior of honey bees. ABC is composed of three kinds of honey bees: employed bees, onlooker bees and scout bees. First, an employed bee is assigned to a particular food source. She carries nectar to the hive and shares information on the nectar amount of the food source with onlooker bees waiting on the hive. Second, an onlooker bee chooses a rich food source, based on the nectar information. If one food source has much nectar amounts, a large number of onlooker bees are assigned to the source. Finally, a scout bee carries out random search for discovering new food sources.

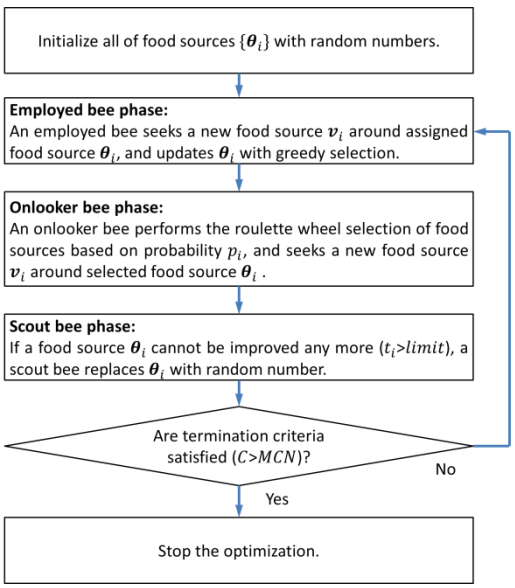

Fig. 1 Flowchart of the ABC algorithm.

In ABC, a colony of artificial honey bees (agents) search for rich food sources (good solutions to a given problem). The position of a food source represents a solution vector of the optimization problem, and a quality of the food source (nectar amount) is represented by a fitness value calculated with the scoring function. The number of food sources *SN* is equal to the number of employed bees or onlooker bees. The three kinds of bees search for a global optimum point in *D*-dimensional real parameter space, where *D* corresponds to the number of optimization parameters (*e.g.*, translation, orientation and conformation of ligand for the flexible protein-ligand docking). A *D*-dimensional solution vector on a food source is described as

$$\boldsymbol{\theta}_i^C = \left[\theta_{i,1}^C, \theta_{i,2}^C, \theta_{i,3}^C, \cdots, \theta_{i,D}^C\right], \quad (1)$$

where i=1,2,…,*SN* is an index of food sources and C=0,1,…,*MCN* (maximum count number) is a current cycle number. In the beginning of optimization (C=0), each parameter of food sources is initialized with uniformly distributed random numbers restricted to certain ranges. A fitness value for a food source is then calculated as

$$fitness_i = \begin{cases} 1/(1+f_i) & \text{if } f_i \geq 0 \\ 1+\text{abs}(f_i) & \text{if } f_i < 0 \end{cases}, \quad (2)$$

where f_i is an actual value of scoring function F to be optimized ($f_i = F(\boldsymbol{\theta}_i^C)$). Since we consider a minimization condition here, a food sources with a lower score of the scoring function have a higher fitness value. After the initialization, ABC performs the optimization process through cycles of three exploration steps by employed bees, onlooker bees and scout bees until the termination criteria are satisfied (Fig. 1).

In the employed bee phase, the employed bees seek a new food source around the assigned food sources, where a new food source is explored in the direction to another food source by perturbing a single optimization parameter as

$$v_{i,j}^C = \theta_{i,j}^C + \phi\left(\theta_{k,j}^C - \theta_{i,j}^C\right). \quad (3)$$

Here, $k \in \{1,2,\ldots,SN\}$ is an index of randomly selected food source except for i. Similarly, $j \in \{1,2,\ldots,D\}$ is an index randomly selected from the D-dimensional parameters. ϕ is a random number in the range of [-1,1]. If a new food source $\boldsymbol{v}_i^C$ has a higher fitness value than the current food source $\boldsymbol{\theta}_i^C$, an employed bee updates $\boldsymbol{\theta}_i^C$ to $\boldsymbol{v}_i^C$. After all the employed bees finish exploiting, they go back to the hive and share the information on the food sources (nectar amounts) with the onlooker bees waiting on the hive.

In the onlooker bee phase, the onlooker bees perform a probabilistic selection of food sources for exploiting. A probability of a food source to be selected is calculated with the fitness values, given by

$$p_i = \frac{fitness_i}{\sum_{l=1}^{SN} fitness_l}. \quad (4)$$

Based on this probability, the onlooker bees perform the roulette wheel selection for the decision of the food source, so that a higher fitness food source is intensively explored by a large number of the onlooker bees. The onlooker bee searches for a new food source around the selected food source using eqn (3), and updates the current food source with the greedy selection in the same way as the employed bee.

In the scout bee phase, a food source which cannot be improved anymore is replaced by a new food source created with random numbers. To find these exhausted food sources, a trial counter t_i is used at each ith food source. If the employed or onlooker bee is unable to improve the previous fitness value of the ith food source, t_i is increased by unity. The trial counter t_i is reset to zero when the ith food source is successfully improved. When t_i reaches the maximum trial number, *limit*, the ith food source is replaced with random numbers and t_i is reset to zero. In this way, the scout bees play an important role in keeping the diversity of population.

F*l*ABCps algorithm

Fitness learning-based ABC with proximity stimuli (F*l*ABCps) is a variant of ABC proposed by Swagatam et al.[17] They introduced three vital modifications to the classical ABC,

to achieve the superior performance for real-world optimization problems.

First, an improved positional modification scheme is introduced. This scheme is developed on the basis of the fitness learning mechanism and the directive component towards adjacent food sites. In the classical ABC, the positional modification given by eqn (3) is performed with a randomly selected food source $\boldsymbol{\theta}_k^C$. Alternatively, elite food sources and neighbor food sources are used in F*l*ABCps for the positional modification, which gives a superior balance of bee's exploration between global search and local search.

Second, a multi-dimensional perturbation scheme is introduced to the positional modification. As mentioned above, the single parameter perturbation is used in the classical ABC, which sometimes leads to the slow convergence for highly multi-dimensional problems.[23] On the other hands, all optimization parameters are updated in PSO and GA, which result in the premature convergence for the complicated problems (*i.e.*, trapped solutions in some local optimum points of scoring function). In F*l*ABCps, a subset of the *D*-dimensional parameters is randomly selected for the positional modification, based on the Rechenberg's 1/5th mutation rule[24]. It helps in efficient convergence of solutions, properly avoiding the premature convergence.

Third, proximity-based stimuli are employed for the food site selection by the onlooker bees. In the classical ABC, the onlooker bees perform the roulette wheel selection of food sources using the probability of eqn (4), which contributes to an intensive exploitation around a high fitness food sources. However, this selection scheme sometimes causes the overcrowding of the onlooker bees at the best-so-far food source, which results in the premature convergence. To circumvent this problem, F*l*ABCps introduces a weighted probability based on the proximity-based stimuli. Since the weighted probability reflects the locality of the food sources, neighbor food sources around the high fitness food sources get more chances to be selected by the onlooker bees.

The performance of F*l*ABCps was examined for two real-world optimization problems including numerous local peaks, non-linearity, interdependence and bound constraints.[17] As a result, F*l*ABCps provided the best solutions among nine state-of-the-arts optimization algorithms. A detailed description and pseudo-code of F*l*ABCps are available in ESI†.

Simulation set-up

The docking performance of F*l*ABCps was evaluated by comparison with four state-of-the-art algorithms: ABC, SODOCK, PSO and LGA. They were assessed under the identical conditions: (I) these examinations were performed in the framework of AutoDock4, (II) a flexibility of a ligand was described with translation, orientation and conformation, and a protein was treated as a rigid object, (III) 85 complexes in Astex diverse set[25] was used for the evaluation of docking performances, (IV) a binding pocket was set with a cubic box $(22.5 \times 22.5 \times 22.5 \text{Å}^3)$ centered at the crystal ligand, (V) the AutoDock energy function[6] was used for scoring function and (VI) the maximum number of energy evaluations was set to 2,500,000. The parameters for F*l*ABCps were determined empirically, so that the population number *SN* and the maximum trial number *limit* were set to 500 and 200, respectively. Setting parameters for the five algorithms are shown in Table S2 (ESI†).

Results and Discussion

Docking accuracy of F*l*ABCps

Table 1 shows the results of the docking calculations obtained with F*l*ABCps, ABC, SODOCK, PSO and LGA for 85 complexes of Astex diverse set. The docking performances were examined in terms of the success rate of the pose prediction and the searching ability of the lowest energy. In addition, 85 complexes in Astex diverse set were divided into three groups according to the number of rotational bonds of ligands, which were used for examining the dependence of the docking accuracy on the number of optimization parameters.

Table 1 Docking results by comparison of five algorithms for 85 complexes of Astex diverse set

N_r[a]	N_c[b]	Success rate [%][c]					No. of wins[d]				
		F*l*ABCps	ABC	SODOCK	PSO	LGA	F*l*ABCps	ABC	SODOCK	PSO	LGA
0~4	25	84.0	84.0	84.0	80.0	84.0	19	4	1	0	1
5~7	31	87.1	80.6	83.9	64.5	77.4	23	3	1	2	2
8~16	29	89.7	79.3	79.3	44.8	55.2	17	4	5	3	0
Total	85	87.1	81.2	82.4	62.4	71.8	59	11	7	5	3

[a]N_r represents the number of rotational bonds for ligands. [b]N_c represents the number of complexes. [c]The rate of successful docking that RMSD from the crystal structure is less than 2Å. [d]The number of wins in finding the lowest energy in the scoring function among the five algorithms.

The success rate of the docking was evaluated with root mean square deviation (RMSD) of the predicted ligand pose from the crystal structure. The simulation results showed that F*l*ABCps provided the best performance of all the five algorithms with the success rate of 87.1%. In general, the docking for highly flexible ligands is more difficult than that for less flexible ligands, due to their large number of optimization parameters.[10] Even for such highly flexible ligands (N_r=8~16), F*l*ABCps can successfully find the correct binding poses with 89.7%, whereas the other methods lowered their success rates. This result indicated that F*l*ABCps might be extended to more complicated systems, such as the partially flexible protein docking including side-chain flexibility of proteins[26] or the docking under explicit water molecules[27].

The present results also showed that F*l*ABCps gave the best results (*i.e.*, the lowest energy) for the 59 complexes. Assuming that the scoring function can describe the correct binding energy, the lowest energy in the scoring function corresponds to the actual binding affinity between a ligand and a protein. Thus, F*l*ABCps is found to give more accurate estimations of the binding affinity, compared with the other four algorithms. The classical ABC gave the success rate of 81.2%, which was better than PSO and LGA. Thus, the basic strategy of ABC is superior to that of GA and PSO for the

protein-ligand docking. From these results, F*l*ABCps is found to be a more suitable algorithm for solving the protein-ligand docking than the conventional algorithms. The detailed results of these simulations are given in Table S3 (ESI†).

Structural analysis of the binding pocket of neprilysin

Next, we analyzed the performance of F*l*ABCps with respect to the binding pocket structure and the energy landscape of the scoring function. The performance of F*l*ABCps was compared with LGA which is a major algorithm implemented in AutoDock. For this analysis, we used the crystal structure of neprilysin (pdb id: 1R1H) and its potent inhibitor N-[3-[(1-aminoethyl)(hydroxy)phosphoryl]-2-(1,1'-biphenyl-4-ylmethyl)propanoyl]alanine (BIR), because LGA could hardly find the correct binding pose of this ligand. We performed 1000 times of docking calculations with LGA and sampled 1000 different docking poses of the ligand. As a result, two specific clusters named cluster-1 and cluster-2 were found on the basis of the structural similarity of their binding poses (Fig 2A). The population of cluster-1 and cluster-2 totally accounts for 58% of all the sampled poses. The main difference between the two clusters was in the direction of two aromatic rings of the ligand (Fig. 2B).

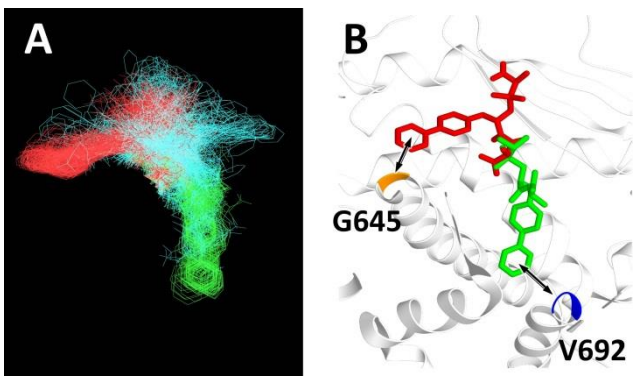

Fig. 2 (A) Superposition of 1000 sampled poses of BIR. Poses of cluster-1, cluster-2 and the others are shown in green, red and cyan, respectively. (B) Definitions of cluster-1 and cluster-2. White ribbon represents the backbone structure of neprilysin. Green colored pose is a representative structure of cluster-1 where the distance between the center of the aromatic ring of the ligand and VAL692 shown in blue is less than 6.5Å. Red colored pose is a representative structure of cluster-2 where the distance between the center of the aromatic ring of the ligand and GLY645 shown in orange is less than 6.5Å.

The 1000 sampled poses were also calculated with F*l*ABCps, which resulted in the same two clusters as the LGA ones. Fig. 3 shows distributions of cluster-1 and cluster-2 with respect to the RMSD from the crystal structure of the ligand. The distributions obtained with F*l*ABCps are completely different from those with LGA. In LGA, we found 10% population for cluster-1 and 48% population for cluster-2, whereas 40% population for cluster-1 and 9% population for cluster-2 were observed in F*l*ABCps. In common, the docking pose with the RMSD less than 2Å is regarded as the successful reproduction of the crystal structure of the ligand. Therefore, the poses of cluster-1 correspond to the crystal structure (see Fig. 4). The lowest binding energies for cluster-1 and cluster-2 were -15.53kcal/mol and -11.97kcal/mol, respectively. These results showed that F*l*ABCps successfully found the correct binding poses at the global minimum (poses of cluster-1). In contrast, LGA gave the binging poses trapped in the local minimum of the scoring function (poses of cluster-2).

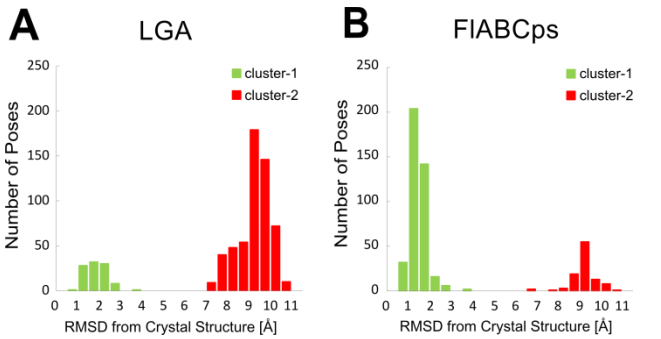

Fig. 3 Distribution of cluster-1 and cluster-2 with respect to the RMSD from the crystal structure of BIR: (A) 1000 docking poses with LGA; (B) 1000 docking poses with F*l*ABCps. Green and red colored bars refer to cluster-1 and cluster-2, respectively.

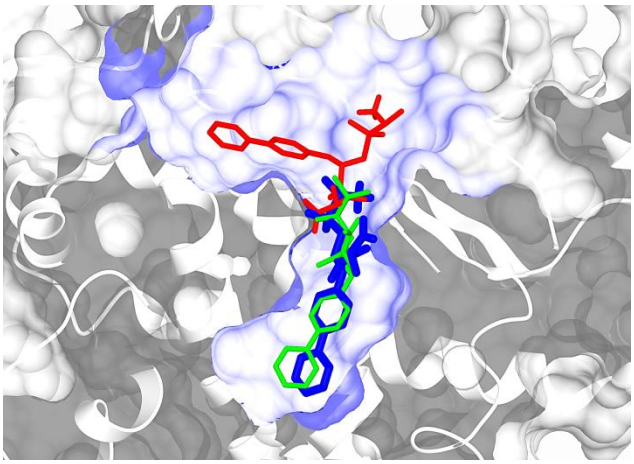

Fig. 4 Molecular structures of the binding pocket of neprilysin and BIR. Blue colored pose is the crystal structure, green colored pose is a representative structure of cluster-1, and red colored pose is a representative structure of cluster-2.

Regarding the molecular structures, the neprilysin has two specific docking regions in its binding pocket: the wide and shallow region on which the poses of cluster-2 are located, and the narrow and deep region on which the poses of cluster-1 and the crystal ligand are located (Fig. 4). In other words, the narrow and deep region corresponds to the global minimum, and the wide and shallow region corresponds to one of the local minima of the scoring function.

Next, we analyzed the energy landscape of the scoring function around the two clusters. Here, we used the RMSD from the crystal structure for simplicity. Fig. 5 shows the energy distributions of cluster-1 and cluster-2 with respect to the RMSD from the crystal structure. The energy distributions plotted on the RMSD space can approximate the multi-dimensional energy landscape of the scoring function. In addition, the RMSD standard deviations of two clusters can be

regarded as the widths of the energy wells in the multi-dimensional spaces. Supposing that these distributions refer to the normal distribution, the energy landscape can be approximated by a Gaussian function. If the centers of these energy wells are set to the individual lowest energy structures, these bell curves reflect the shapes of the multi-dimensional energy landscape around cluster-1 and cluster-2. The standard deviations of cluster-1 and cluster-2 from the individual lowest energy structures were 1.8Å and 2.9Å, respectively. Therefore, the poses of cluster-1 were located on the narrow and steep energy well of the global minimum, whereas the poses of cluster-2 were trapped in the wide and gradual energy well of the local minimum. These results can be interpreted as follows. The neprilysin has the wide and shallow region in its binding pocket on which the poses of cluster-2 are located. Around this region, the scoring function gives the wide and gradual energy well of the local minimum. The conventional algorithms, including GA and PSO, usually show the high convergence ability for simple problems. However, they often give solutions trapped in some local minima, when dealing with multi-modal and multi-dimensional problems.[15,16] Also in our simulation results, most of the LGA calculations gave the binding poses trapped in the local minimum (cluster-2). In contrast, F*l*ABCps successfully found the correct binding poses existing in the global minimum (cluster-1), properly avoiding such a local minimum. Some kinds of proteins which have a wide and shallow binding pocket were supposed to provide a challenging task for *in silico* docking. This is because such kinds of proteins usually contain a large number of local minima on their energy landscape of the scoring function. F*l*ABCps would be a suitable algorithm for such proteins with these features as kinases.

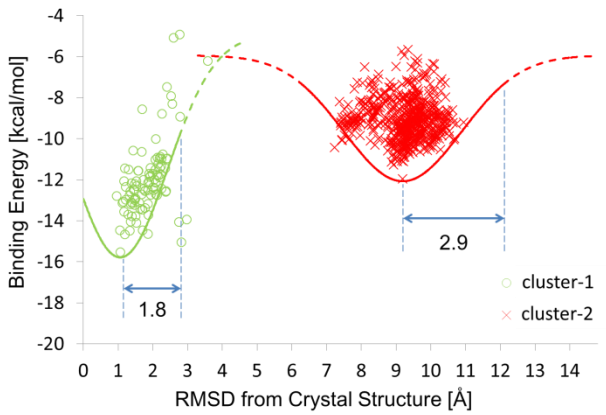

Fig. 5 Scatter plots of the binding energies of cluster-1 and cluster-2 with respect to the RMSD from the crystal structure. Poses of cluster-1 are shown by green circles and those for cluster-2 are shown by red crosses. Solid lines in the individual clusters show Gaussian distributions of the RMSD from the pose with the lowest energy; standard deviation of cluster-1 is 1.8Å and that of cluster-2 is 2.9Å.

The scoring (objective) functions for protein-ligand docking are generally constructed by summation of interatomic potentials between all pairs of protein and ligand atoms.[28] Eventually, these functions with numerous terms describe non-convex and multi-modal solution space, even if the pairwise interatomic potentials are simple convex functions. These kinds of objective functions are often used for optimization problems of molecular sciences in which any interatomic potentials are calculated. Nature-inspired metaheuristic optimization algorithms are then developed to solve such kinds of problems with non-convex or multi-modal functions that are not amenable to the approach via differentiations as in the steepest descent method. F*l*ABCps is one of the most robust optimization algorithms for the problems containing a number of local minima and/or highly multi-dimensional solution space.

Conclusions

In this work, we introduced a novel optimization algorithm F*l*ABCps for the protein-ligand docking. The performance of F*l*ABCps was assessed in comparison with the four state-of-the-art docking algorithms. Simulation results revealed that F*l*ABCps gave significantly accurate docking poses of the ligands, compared with the other four algorithms. The results also showed that F*l*ABCps provided the best performance for the highly flexible ligands with many optimization parameters. In addition, we analyzed the simulation results in terms of the energy landscape of the scoring function and the shape of the binding pocket of the receptor protein. Some kinds of proteins were supposed to be a challenging task for the docking, because they usually possess a large number of wide and gradual energy wells corresponding to the local minima in the scoring function. For these proteins, the conventional optimization algorithms can hardly find the correct binding pose of ligand. In contrast, F*l*ABCps successfully find the correct binding poses, properly avoiding such local minima. Consequently, F*l*ABCps would become an useful algorithm for more complicated optimization problems concerning *in silico* drug discovery.

Notes and references

Department of Computational Science, Graduate School of System Informatics, Kobe University, 1-1, Rokkodai, Nada, Kobe, Hyogo 657-8501, Japan. E-mail: uehara@eniac.scitec.kobe-u.ac.jp, fujimoto@ruby.kobe-u.ac.jp, tanaka2@kobe-u.ac.jp;

† Electronic Supplementary Information (ESI) available: a detailed description of the FlABCps algorithm, setting parameters of the five algorithms for docking simulation, complete results of docking experiments for astex diverse set. See DOI: 10.1039/b000000x/

1 A. N. Jain, *Curr. Opin. Drug Discov. Devel.*, 2004, **7**, 396–403.
2 D. Joseph-McCarthy, J. C. Baber, E. Feyfant, D. C. Thompson and C. Humblet, *Curr. Opin. Drug Discov. Devel.*, 2007, **10**, 264–74.
3 A. Wlodawer and J. Vondrasek, *Annu. Rev. Biophys. Biomol. Struct.*, 1998, **27**, 249–84.
4 G. Kryger, I. Silman and J. L. Sussman, *Structure*, 1999, **7**, 297–307.
5 Y. S. Babu, P. Chand, S. Bantia, P. Kotian, A. Dehghani, Y. El-Kattan, T. H. Lin, T. L. Hutchison, A. J. Elliott, C. D. Parker, S. L. Ananth, L. L. Horn, G. W. Laver and J. A. Montgomery, *J. Med. Chem.*, 2000, **43**, 3482–6.
6 R. Huey, G. M. Morris, A. J. Olson and D. S. Goodsell, *J. Comput. Chem.*, 2007, **28**, 1145–52.
7 R. Wang, L. Lai and S. Wang, *J. Comput. Aided. Mol. Des.*, 2002, **16**, 11–26.

8 H. Gohlke, M. Hendlich and G. Klebe, *J. Mol. Biol.*, 2000, **295**, 337–356.
9 D. W. Miller and K. A. Dill, *Protein Sci.*, 1997, **6**, 2166–79.
10 H.-M. Chen, B.-F. Liu, H.-L. Huang, S.-F. Hwang and S.-Y. Ho, *J. Comput. Chem.*, 2007, **28**, 612–23.
11 G. Jones, P. Willett, R. C. Glen, a R. Leach and R. Taylor, *J. Mol. Biol.*, 1997, **267**, 727–48.
12 G. M. Morris, D. S. Goodsell, R. S. Halliday, R. Huey, W. E. Hart, R. K. Belew and A. J. Olson, *J. Comput. Chem.*, 1998, **19**, 1639–1662.
13 C. Sammut and G. I. Webb, in *Encyclopedia of Machine Learning*, Springer US, NY, 2010, vol. 33, pp. 760-766.
14 V. Namasivayam and R. Günther, *Chem. Biol. Drug Des.*, 2007, **70**, 475–84.
15 J. Andre, P. Siarry and T. Dognon, *Adv. Eng. Softw.*, 2001, **32**, 49–60.
16 B. Alatas, E. Akin and a. B. Ozer, *Chaos, Solitons & Fractals*, 2009, **40**, 1715–1734.
17 S. Das, S. Biswas and S. Kundu, *Appl. Soft Comput.*, 2013, **13**, 4676–4694.
18 D. Karaboga and B. Basturk, *J. Glob. Optim.*, 2007, **39**, 459–471.
19 R. Ma, X. Xu, L. Zhao, R. Cao and Q. Fang, *Int. J. Biomath.*, 2013, **06**, 1350038.
20 B. Akay, *Appl. Soft Comput.*, 2013, **13**, 3066–3091.
21 D. Karaboga and C. Ozturk, *Appl. Soft Comput.*, 2011, **11**, 652–657.
22 J. Fuhrmann, A. Rurainski, H.-P. Lenhof and D. Neumann, *J. Comput. Chem.*, 2010, **31**, 1911–8.
23 B. Akay and D. Karaboga, *Inf. Sci. (Ny).*, 2012, **192**, 120–142.
24 B. Thomas, and B. F. David, in *Handbook of Evolutionary Computation*, ed. M. Zbigniew, Oxford University Press, Oxford, 1997, pp. A2.3:6-A2.3:7.
25 M. J. Hartshorn, M. L. Verdonk, G. Chessari, S. C. Brewerton, W. T. M. Mooij, P. N. Mortenson and C. W. Murray, *J. Med. Chem.*, 2007, **50**, 726–41.
26 M. I. Zavodszky and L. A. Kuhn, *Protein Sci.*, 2005, **14**, 1104–14.
27 M. L. Verdonk, G. Chessari, J. C. Cole, M. J. Hartshorn, C. W. Murray, J. W. M. Nissink, R. D. Taylor and R. Taylor, *J. Med. Chem.*, 2005, **48**, 6504–15.
28 S.-Y. Huang, S. Z. Grinter and X. Zou, *Phys. Chem. Chem. Phys.*, 2010, **12**, 12899–12908.

