J Comput Aided Mol Des (2009) 23:1–12
DOI 10.1007/s10822-008-9232-5

An improved adaptive genetic algorithm for protein–ligand docking

Ling Kang · Honglin Li ·
Hualiang Jiang · Xicheng Wang

Received: 1 February 2007 / Accepted: 30 July 2008 / Published online: 6 September 2008

Abstract A new optimization model of molecular docking is proposed, and a fast flexible docking method based on an improved adaptive genetic algorithm is developed in this paper. The algorithm takes some advanced techniques, such as multi-population genetic strategy, entropy-based searching technique with self-adaptation and the quasi-exact penalty. A new iteration scheme in conjunction with above techniques is employed to speed up the optimization process and to ensure very rapid and steady convergence. The docking accuracy and efficiency of the method are evaluated by docking results from GOLD test data set, which contains 134 protein–ligand complexes. In over 66.2% of the complexes, the docked pose was within 2.0 Å root-mean-square deviation (RMSD) of the X-ray structure. Docking time is approximately in proportion to the number of the rotatable bonds of ligands.

Keywords Genetic algorithms · Information entropy · Molecular docking · Optimization design · Penalty function · Self-adaptation

Ling Kang and Honglin Li have contributed equally to this work.

L. Kang
Department of Computer Science and Engineering, School
of Electronic and Information Engineering, Dalian University
of Technology, Dalian 116023, China

L. Kang · H. Li · X. Wang (✉)
Department of Engineering Mechanics, State Key Laboratory
of Structural Analysis for Industrial Equipment,
Dalian University of Technology, Dalian 116023, China
e-mail: guixum@dlut.edu.cn

H. Li · H. Jiang
Drug Discovery and Design Center, State Key Laboratory
of Drug Research, Shanghai Institute of Materia Medica,
Chinese Academy of Sciences, Shanghai 201203, China

Introduction

Molecular docking problem is generally cast as a problem of finding the low-energy binding modes of a ligand based on the "lock and key mechanism" [1], within the active site of a receptor, whose structure is known [2]. It plays an important role in drug design, which is demonstrated by the vast amount of literature [3–10] devoted to the optimization methods for molecular docking design since the pioneering work of Kuntz et al. [11]. Protein–ligand docking (PLD) for drug molecular optimization design is an ideal approach to virtual screening, i.e., to search large sets of compounds for finding new lead structure. A fundamental problem for molecular docking is that the design space is very large and grows combinatorially with the number of degrees of freedom of the interacting molecules. The computation of the ligand–receptor interaction energy at all possible docking configurations cannot be completed in a reasonable amount of computing time, for example, a typical protein receptor might occupy a volume of some 60 Å^{-3}, even with a moderate translational resolution of 1 Å, this leaves 216,000 translations to search. For a rotational resolution of 20° in each axis and potential ligand with 35 atoms and protein receptor with 3,500, 1.5×10^{14} pairwise nonbonding evaluations will be needed to scan the complete range of possible docking configurations. Even if one can evict 99% of the points by employing various assumptions, we will still require 1.5×10^{12} evaluations. If billions of compounds are to be screened in this way, the required computational power becomes a limiting feature. Therefore, simpler and more efficient methods are continuously being researched into.

In this paper, an entropy-based optimization model is constructed to obtain the narrowing coefficients of the searched space for multi-population evolution very easily. Then a new iteration scheme in conjunction with multi-population genetic strategy and an entropy-based searching technique is developed to search optimal molecular orientation and conformation. The elitist maintaining strategy and efficient convergent rule are used to close the global solution, and the contracted space is employed as convergence criterion instead of the genetic generations used in the most of the genetic algorithms, so that docking time is dramatically decreased. Furthermore, a novel adaptive strategy is employed; the probabilities of the crossover and mutation operators are optimized as the added design variables in the evolution process. These strategies can speed up the optimizing process and ensure very rapid and steady convergence.

In order to evaluate the new docking method, we have conducted a numerical experiment with 134 protein–ligand complexes from the publicly available GOLD test set [12]. Comparisons with Glide [13], GOLD [12], FlexX [14], and Surflex [15] indicate that docking accuracy of our method is comparable to these methods. The computational efficiency of the method has significantly improvement. Docking time is approximately in proportion to the number of rotatable bonds of ligands.

Methodology

Optimization design model of molecular docking

The optimization problem for molecular docking can be written as follows

$$\begin{array}{ll} \min & f(\mathbf{d}) \\ \text{s.t.} & g_j(\mathbf{d}) \leq 0, \quad j = 1, 2, \ldots, q \end{array} \tag{1}$$

where $\mathbf{d} = \left\{T_x, T_y, T_z, R_x, R_y, R_z, T_{b1}, \ldots, T_{bn}\right\}^T$ is a vector of $q(q = n + 6)$ design variables, $T_{b1},\ldots,T_{bn}$ are the torsion angles of the rotatable bonds for flexible ligand docking, T_x, T_y, T_z, R_x, R_y, R_z are the position coordinates and rotational angles of the anchor for the matching-based orientation search. The objective function $f(\mathbf{x})$ is intermolecular interaction energy

$$f(\mathbf{d}) = \sum_{i=1}^{n_{lig}} \sum_{j=1}^{n_{rec}} \left(\frac{A_{ij}}{r_{ij}^a} - \frac{B_{ij}}{r_{ij}^b} + 332.0 \frac{q_i q_j}{D r_{ij}} \right) \tag{2}$$

where each term is a double sum over the ligand atom i and the receptor atom j, r_{ij} is the distance between atom i in ligand and atom j in receptor, A_{ij}, B_{ij} are van der Waals repulsion and attraction parameters, a, b are van der Waals repulsion and attraction exponents, q_i, q_j are point charges on atoms i and j, D is dielectric function, and 332.0 is a factor for conversion of electrostatic energy into kilocalories per mole. The constraints $g(\mathbf{x})$ may be represented as the size limits of the design variables, and certain behavior constraints of the molecule exist, as are shown below:

$$\begin{cases} \underline{T_x} \leq T_x \leq \overline{T_x} \\ \underline{T_y} \leq T_y \leq \overline{T_y} \\ \underline{T_z} \leq T_z \leq \overline{T_z} \\ -\pi \leq angle \leq \pi, \; angle = R_x, R_y, R_z, T_{b1}, \cdots, T_{bn} \end{cases} \tag{3}$$

In the protein–ligand docking process, the binding free energy is a function of the Cartesian coordinates of the ligand atoms only. The Cartesian coordinates of all ligand atoms can be determined by solving the optimization problem (1). This indicates that the optimal conformation of a flexible ligand is determined by translational (T_x, T_y, T_z), rotational (R_x, R_y, R_z) and torsional motions T_{b1}, T_{b2},…,T_{bn} (n is the number of torsion bonds) (T_{bi}, i = 1, 2,…, n, n is the number of torsion bonds). The former variables, which account for the six degrees of freedom for a rigid body, can also be interpreted as the orientation of the ligand; T_{bi} is the angle of the ith flexible bond. Since the movement of the ligand should be limited in a pocket confined to the active site of the receptor the design subspace of (T_x, T_y, T_z) is defined as a cuboid circumscribed in the pocket. ($\underline{T_x}$, $\underline{T_y}$, $\underline{T_z}$) and ($\overline{T_x}, \overline{T_y}, \overline{T_z}$) are the minimum and maximum Cartesian coordinates of the circumscribed cuboid. The defined design subspace is larger than the pocket, it not only ensures that the ligand can move freely within the binding pocket, but also cuts down on computational costs by avoiding the complexity of resolving the actual boundary. The remaining variables are allowed to vary between $-\pi$ and π rad.

Transformation of optimization model

As mentioned above, the problem (1) involves lots of constraints, so it is difficult to solve it. In order to solve problem (1) efficiently, first, we introduce some definitions and theorems as follows.

Definition 1 If ψ is a positive real variable, and $G = \left\{g_j(\mathbf{d})\right\}$, $j = 1, \ldots, q$, is a set of constraint functions, then

$$E(G) = (1/\psi) \ln \sum_{j=1}^{q} \exp(\psi g_j(\mathbf{d})) \tag{4}$$

is a parametric constraint evaluation (PCE) function. The optimization problem (1) is transformed into the following model by means of PCE function:

$$\begin{array}{ll} \min & f(\mathbf{d}) \\ \text{s.t.} & g_\psi(\mathbf{d}) = (1/\psi) \ln \sum_{i=1}^{q} \exp(\psi g_i(\mathbf{d})) \leq 0 \end{array} \tag{5}$$

Definition 2 If, for any $F(\mathbf{d}) = \left\{f_1(\mathbf{d}), f_2(\mathbf{d}), \ldots, f_q(\mathbf{d})\right\}$, and $\overline{F}(\mathbf{d}) = \left\{\overline{f}_1(\mathbf{d}), \overline{f}_2(\mathbf{d}), \ldots, \overline{f}_q(\mathbf{d})\right\}$, $F(\mathbf{d}), \overline{F}(\mathbf{d}) \in E^q$

with $f_j(\mathbf{d}) \le \overline{f}_j(\mathbf{d}), j = 1, 2, \ldots, q$, and there exists at least one $j_0, (1 \le j_0 \le q)$, such that $f_{j_0}(\mathbf{d}) < \overline{f}_{j_0}(\mathbf{d})$, then $F(\mathbf{d}) \le \overline{F}(\mathbf{d})$ or, simply $F \le \overline{F}$.

Definition 3 If, for any, $F, \overline{F} \in E^q$, with $F \le \overline{F}$, $E(F) < E(\overline{F})$, then $E(F)$ is a strictly monotone increasing function of F.

Lemma *The PCE function $E(G)$ is a strictly monotone increasing function of G, and if $\psi \to \infty$ then*

$$(1/\psi) \ln \sum_{j=1}^{q} \exp(\psi g_j(\mathbf{d})) = \max g_j(\mathbf{d}) \quad j = 1, 2, \ldots, q \tag{6}$$

Proof Let

$$G = \{g_j(\mathbf{d})\} \le \overline{G} = \{\overline{g}_j(\mathbf{d})\}, \quad j = 1, 2, \ldots, q \tag{7}$$

By Definition 2

$$g_j(\mathbf{d}) \le \overline{g}_j(\mathbf{d}), \quad j = 1, 2, \ldots, q \tag{8}$$

and there exists at least one $j_0 (1 \le j_0 \le q)$ such that

$$g_{j_0}(\mathbf{d}) < \overline{g}_{j_0}(\mathbf{d}) \tag{9}$$

Then for $\psi > 0$,

$$\psi g_{j_0}(\mathbf{d}) < \psi \overline{\mathbf{g}}_{j_0}(\mathbf{d}) \tag{10}$$

$$\exp(\psi g_{j_0}(\mathbf{d})) < \exp(\psi \overline{g}_{j_0}(\mathbf{d})) \tag{11}$$

Hence

$$\sum_{j=1}^{q} \exp(\psi g_j(\mathbf{d})) < \sum_{j=1}^{q} \exp(\psi \overline{g}_j(\mathbf{d})) \tag{12}$$

Taking logarithms on both sides and dividing byψ

$$E(G) = (1/\psi) \ln \sum_{j=1}^{q} \exp(\psi g_j(\mathbf{d})) < (1/\psi) \ln \sum_{j=1}^{q} \exp(\psi \overline{g}_j(\mathbf{d})) \tag{13}$$

i.e. $E(F)$ is a strictly monotone increasing function of increasing function of F. The ψ norm of the q-dimensional vector

$$E_G = \left\{e^{g_1(\mathbf{d})}, e^{g_2(\mathbf{d})}, \ldots, e^{g_q(\mathbf{d})}\right\}^T \tag{14}$$

is given by

$$N_\psi(E_G) = \left(\sum_{j=1}^{q} e^{\psi g_j(\mathbf{d})}\right)^{(1/\psi)} \tag{15}$$

The uniform norm, also called the maximum norm, is defined by

$$N_\infty(E_G) = \lim_{\psi \to \infty} N_\psi(E_G) \tag{16}$$

Since $e^{g_j(\mathbf{d})} > 0$ by Jensen's inequality, the norm is a strictly monotone decreasing function of its order, i.e.

$$N_s < N_r \quad \text{for } r < s \tag{17}$$

The importance of this inequality is that it holds also in the limit as $s \to \infty$. Thus, Eq. 16 may be written as

$$N_\infty(E_G) = \max\left(\mathrm{e}^{g_j(\mathbf{d})}\right) < N_r(E_G) \tag{18}$$

Taking logarithms on both side of Eq. 18 and substituting from Eqs. 15 and 16 gives

$$\lim_{\psi \to \infty} (1/\psi) \ln \sum_{j=1}^{q} \exp(\psi g_j(\mathbf{d})) = \max(g_j(\mathbf{d})) \tag{19}$$

and the proof is completed.

The PCE function plays an important role in the proposed method. By means of the PCE function, we can give the following theorem and simplify the problem (1) with multiconstraints as an optimization problem with a single constraint only.

Theorem 1 *If $\psi \to \infty$, then the optimization problem (1) and*

$$\begin{cases} \min & f(\mathbf{d}) \\ \text{s.t.} & g_\psi(\mathbf{d}) = (1/\psi) \ln\left\{\sum_{k=1}^{q} \exp[\psi g_k(\mathbf{d})]\right\} \le 0 \end{cases} \tag{20}$$

have the same Kuhn–Tucker points.

Proof The Lagrange augmented function problem (20) is

$$L(\mathbf{d}, \gamma) = f(\mathbf{d}) + (\gamma/\psi) \ln \sum_{j=1}^{q} \exp(\psi g_j(\mathbf{d})) \tag{21}$$

where $\gamma > 0$ is the Lagrange multiplier of corresponding constraint. The Kuhn–Tucker condition for problem (20) is given as

$$\partial f(\mathbf{d})/\partial d_i + (\gamma/\psi)\left\{\sum_{j=1}^{q} \exp(\psi g_j(\mathbf{d})) \cdot \partial g_j(\mathbf{d})/\partial d_i\right\} / \sum_{j=1}^{q} \exp[\psi g_j(\mathbf{d})] = 0$$

$$(1/\psi) \ln \sum_{j=1}^{q} \exp(\psi g_j(\mathbf{d})) \le 0 \tag{23}$$

$$(\gamma/\psi) \ln \sum_{j=1}^{q} \exp(\psi g_j(\mathbf{d})) = 0, \quad \gamma \ge 0 \tag{24}$$

By means of Lemma 1 and Eq. 23, if $\psi \to \infty$, then

$$(1/\psi) \ln \sum_{j=1}^{q} \exp(\psi g_j(\mathbf{d})) = \max g_j(\mathbf{d}) \le 0, \quad j = 1, 2, \ldots, q \tag{25}$$

i.e.

$$g_j(\mathbf{d}) \leq 0 \tag{26}$$

Substituting

$$\gamma = \psi, \mu_j = \frac{\exp\left[\psi g_j(\mathbf{d})\right]}{\sum_{j=1}^{q} \exp\left[\psi g_j(\mathbf{d})\right]} \tag{27}$$

into Eq. 22 gives

$$\partial f(\mathbf{d})/\partial d_i + \sum_{j=1}^{q} \mu_j \frac{\partial g_j(\mathbf{d})}{\partial d_j} = 0 \tag{28}$$

Combining Eqs. 24 and 27, if $\psi \rightarrow \infty$, then

$$\begin{cases} g_j(\mathbf{d}) = 0 & \text{if } \mu_j > 0 \\ g_j(\mathbf{d}) < 0 & \text{if } \mu_j = 0 \end{cases} \tag{29}$$

Equations 26, 28 and 29 are identical the Kuhn–Tucker condition of the problem (1). Hence the problems (20) and (1) have the same Kuhn–Tucker points and vice versa. The Theorem 1 is proved.

Kuhn–Tucker points are obtained by solving the Kuhn–Tucker conditions, which are necessary condition for the optimum solution of non-linear programming with equality and inequality constraints [16]. Theorem 1 shows that to solve problem (1) with multi constraints can be substituted by solving a simple problem (20) with a single constraint only.

In order to solve the problem (20) by using genetic algorithm, we transfer it into the following unconstraint optimization problem by using quasi-exact penalty function:

$$\min \varphi_\psi(\mathbf{d}) = f(\mathbf{d}) + (\alpha/\psi)\ln\left\{1 + \sum_{i=1}^{q} \exp(\psi g_i(\mathbf{d}))\right\} \tag{30}$$

The parameter ψ can be chosen in the range 10^3–10^5 and α is penalty factor, $\alpha > 0$.

Adaptive entropy-based genetic algorithm

The objective function of problem (30) is nonlinear and the design space is non-convex, the sensitivity analysis is very difficult. There is a critical need to study alternate strategies for optimal design that are not susceptible to the pitfalls of methods of nonlinear programming. Genetic algorithms provide such a capability of their successful adaptation and implementation in a series of optimal design problems. But genetic search process is a time-consuming work, so that hindered them from applied to molecular docking optimization problem, especially to massively among a virtual library of billions of small molecules for compounds that can bind to known protein binding sites. In such circumstances, a novel adaptive genetic algorithm (GA) is here proposed, in which an entropy-based searching technique with multi-population and the quasi-exactness penalty function are developed to ensure rapid and steady convergence.

By means of Eq. 30, the fitness function of genetic algorithm may be written as:

$$\max F(\mathbf{d}) = C - \varphi_\psi(\mathbf{d}) \tag{31}$$

Problem (31) can be solved as an evolutionary design model, in which $F(\mathbf{d})$ is the fitness function, C is a large positive number to ensure $F > 0$. The quasi-exact penalty function $\varphi_\psi(\mathbf{d})$ is developed to solve nonlinear programming (NLP) problems with equality and inequality constraints.

The traditional genetic algorithm involves five basic operators. These include the coding of string, the fitness function, reproduction, crossover and mutation. The probabilities of the crossover and mutation operators p_c and p_m must be provided in GA, and are generally provided as initial data. However, these genetic parameters can make the convergence of the algorithm slow and unsteady if they are not appropriately defined. Here the probabilities p_c and p_m are assigned to be the added design variables to overcome the difficulty in confirming the genetic parameters. The lower and upper limits of p_c and p_m can be defined in a reasonable region (here $0.6 \leq p_c \leq 1.0, 0.0 \leq p_m \leq 0.1$).

For multi-population genetic strategy, the genetic algorithm begins from generating arbitrarily m populations with all the same searching space, i.e. design space. If $F_j(\mathbf{d})$ $(j = 1, \ldots, m)$ represent that the best value of the fitness function occurs in the jth population, then we need to maximum $F_j(\mathbf{d})$ $(j = 1, \ldots, m)$ by means of a genetic operations, i.e. to solve the following optimization problem:

$$\min -F_j(\mathbf{d}), \quad j = 1, 2, \ldots, m \tag{32}$$

Problem (32) is a multi-objective optimization, which is very difficult to solve completely. For the improved genetic algorithm with narrowing of the search space, we need only to know efficient narrowing coefficients for the searched space.

Shannon's theorem [17] has wide-ranging applications in both communications and data storage applications. This theorem is of foundational importance to the modern field of information theory [18]. There are similarities between the process of optimization and communication of information theory. Information entropy or Shannon entropy H of a discrete set of probabilities p_1, …, p_n is defined by

$$\begin{aligned} H = &\ -\sum p_i \ln p_i \\ \text{s.t.} &\ \sum p_i = 1, \quad p_i \in [0, 1] \end{aligned} \tag{33}$$

Shannon entropy can be used to measure the uncertainty about the realization of a random variable. If p_j is here defined as a probability that the optimal solution of the problem (32) occurs in the population j, then Shannon entropy will be decreased during optimization process of problem (32) and an entropy-based optimization model can be constructed as follows:

$$\begin{cases} \min & -\sum_{j=1}^{m} p_j F_j(\mathbf{d}) \\ \min & H = -\sum_{j=1}^{m} p_j \ln(p_j) \\ \text{s.t.} & \sum_{j=1}^{m} p_j = 1, \quad p_j \in [0,1] \end{cases} \tag{34}$$

where H is the information entropy.

Theorem 2 *The optimization problem (34) and (32) both have the same optimal solution.*

Proof Suppose that $\mathbf{d}^*$ and $\mathbf{p}^* = \{p_1^*, p_2^*, \cdots, p_m^*\}$ are the optimal solution of problem (34), so that

$$\min H^* = -\sum_{j=1}^{m} p_j^* \ln(p_j^*) = p_l^* \ln(1) = 0 \tag{35}$$

where $p_l^* = 1, p_i^* = 0$ for $i \neq l$, i.e. the optimal solution of the problem (34) occurs in the population l. Hence

$$\min\ -\sum_{j=1}^{m} p_j F_j(\mathbf{d}^*) = \min -F_l(\mathbf{d}^*) \tag{36}$$

Obviously, $\mathbf{d}^*$ are also the optimal solution of problem (32). It can be similarly proved that the optimal solution of problem (32) is also the optimal solution of problem (34), and the proof is completed.

The solution p_j of Eq. 34 can be obtained easily and explicitly.

$$p_j^* = \exp(vF_j(\mathbf{d})) / \sum_{j=1}^{m} \exp(vF_j(\mathbf{d})) \tag{37}$$

in which

$$v = (\beta - 1)/\beta \tag{38}$$

v is called as the quasi-weight coefficient (here $\beta = 0.5$).

The $(1 - p_j)$ can be used as the coefficients of narrowing searching space in the modified genetic algorithm. When the optimal solution occurs in the lth population, then $(1 - p_l^*) = 0$, and its searching space is not narrowing. Using multi-population genetic strategy with narrowing down searching space, the M populations with N members are generated in the given space.

Design space is defined as initial searching space $D(0)$. M populations with N members are generated in the given space. After a new generation is independently evolved in each population, the searching space of each population is narrowed according to the following equation:

$$\begin{aligned} & D_j(K) = (1 - p_j) D_j(K-1) \\ & \underline{d_{ji}}(K) = \max\left\{\left[d_{ji}^*(K) - 0.5(1 - p_j) D_j(K)\right], \underline{d_{ji}}(0)\right\} \\ & \overline{d_{ji}}(K) = \max\left\{\left[d_{ji}^*(K) + 0.5(1 - p_j) D_j(K)\right], \overline{d_{ji}}(0)\right\} \end{aligned} \tag{39}$$

where $D_j(K)$ is the searching space of the population j at Kth iteration. $\underline{d_{ji}}(K)$ and $\overline{d_{ji}}(K)$ are the modified lower and upper limits of ith design variable in the population j at Kth iteration, respectively. $d_{ji}{}^*(K)$is the value of design variable i of the best member in the population j.

Equation 39 is employed to control the narrowing of searching space for each population. If $(1 - p_l^*) = 0$, the optimal solution occurs in the lth population, and its searching space is not narrowing. Then the convergence criterion of the proposed method can be defined as: when the searching space in the best population has been reduced to a very small area (a given tolerance), the global optimal solution can be obtained approximately. Using narrowed space as the convergence criterion could controls the convergence of the algorithm effectively.

The algorithm consists of the following steps:

Step 1. Generate an initial population and implement the duplicate operator.
Step 2. Perform crossover and mutation operators among populations.
Step 3. Narrow down the design spaces of each population and find the best individual; reserve according to the elitist strategy, next, check the convergence to ensure that the searching space in the best population has been reduced to the given tolerance satisfied. If it has, go to step 4; otherwise, return to step 2.
Step 4. Output the optimization results and stop the process.

Results and discussion

Test data set

The GOLD test data set, originally proposed by Jones et al. [12], was chosen for our studies. Each complex was separated into a probe molecule and a docking ligand according to the biological interacting pairs. Each protein molecule was obtained by excluding ligands, all structural water molecules, cofactors, and metal ions from the receptor pdb file. Next, a mol2 file was generated by adding hydrogen atoms and Kallman charge using Sybyl6.8. Residues around the bound ligand within a radius of 6.5 Å were isolated from the protein to define as the active site. The ligands were then prepared by adding hydrogen atoms and Gasteiger-Marsili atomic charges adopted in Sybyl6.8. The heavy atoms number of the ligands ranged from 6 to

55, with 83.6% of the ligands possessing fewer than 30 such atoms. Besides, The rotatable bonds of the ligands ranged widely from 0 to 22, with greater than 88.8% of the ligands possessing fewer than 15 such bonds.

Docking accuracy

Generally, the primary criteria for evaluating docking methods are docking accuracy, scoring veracity, screening efficiency, and computational speed [15]. Docking accuracy and speed are more important indexes than others. Docking accuracy is based on the root-mean-square deviation (RMSD) value of the locations of all heavy atoms in the model from those of the crystal structure. In general, the docking accuracy is acceptable if the RMSD value between the docked pose and X-ray crystal structure is less than 2.0 Å. Depending on the RMSD values, the results was assigned to four categories. The first, excellent, was for those predictions in which the top scoring pose was within 0.5 Å RMSD from experimental results. If the RMSD values were between 0.5 and 2.0 Å, the results would be assigned to the good category. A third category, close, was used for those predictions that the RMSD values were between 2.0 and 2.5 Å. And a fourth category, errors, was used for those predictions that the RMSD values were between 2.5 and 3.0 Å. Finally, the fifth category, wrong, was used for completely incorrect predictions with RMSD values larger than 3.0 Å. The energy of a conformer was computed by Eq. 2 with the nonbonding 12-6 Lennard-Jones and electrostatic energy terms. The docking accuracy and speed of our program was evaluated by the docking results on an SGI Fuel Workstation.

Table 1 summarizes the performance of our docking method. As shown, the docking program yielded 30 docking solutions with a RMSD values below 0.5 Å. Nearly 46% of the results were excellent results. If we consider acceptable results to be contained in the excellent and good categories, then our program achieved a 70% prediction rate.

Table 1 Docking accuracy of the improved method

RMSD (Å)	Number of ligands	Ratio of the ligands in the test data set (%)
0.0–0.49	30	22.4
0.5–0.99	31	23.1
1.0–1.49	17	12.7
1.5–1.99	10	7.5
2.0–2.49	15	11.2
2.5–2.99	3	2.2
3.0–3.49	4	3
≥3.5	24	17.9

Figure 1 shows a representative example for excellent docking; The three-dimensional structure of the complex of 1FKG with FKBP12 has been determined by X-ray crystallography to a resolution of 2.0 Å [19]. Flexibility of FKBP12 was described using 11 rotatable bonds. The active site was determined by flooding-filling to a radius of 6.5 Å (roughly corresponding to all solvent-accessible cavity atoms). And the movement of the ligand was limited in a cuboid circumscribed in the active site. The defined cuboid is around the bound ligand within a radius of 6.5 Å shown in Fig. 1a. Only 61 genetic iterations were needed to obtain the highest fitness score. All the atoms of the ligand of 1FKG were correctly placed with RMSD of 0.27 Å (see Fig. 1b). The algorithm run took 14.32 s.

Table 2 gives the relationship among RMSD values, docking time and the flexibility of the ligands. RMSD values and docking time increase with ligand flexibility, and the docking time is approximately in proportion to the number of rotatable bonds of ligands, i.e. the docking time increases approximately 1 s per adding a rotatable bond (see Fig. 2). Furthermore, with the increasing of ligand size, docking accuracy and efficiency will also decrease (see Table 3).

Table 4 provides comparisons with other programs [12–15]. Their scoring results are insignificant because these programs adopt the different score function, so Table 4 contains only the RMSD values of the poses corresponding to the ligands optimized by using the docking score. To avoid biases from different data sets, we only compare the results within the subsets of the GOLD data set, which contain 99 complexes, 120 complexes, 81 complexes, 122 complexes and 96 complexes for GOLD, FlexX, Surflex, Glide and DOCK6, respectively. The results show that our program has good docking accuracy. The further comparisons of docking accuracy with above programs are given in Tables 5–7.

Table 5 gives the comparison with GOLD; this paper gives an average RMSD value of 2.12 Å, whereas it is 3.00 Å for GOLD. However, the successful results of GOLD were slightly better than our program, the RMSD values of 66 solutions in Gold results are less than 2.0 Å, while it is 64 in our results. Tables 6 and 7 give comparison with FlexX and Surflex, the average RMSD value of our program is quite better than them. Table 8 gives comparisons with Glide for RMSD values of the ligands with the different number of rotatable bonds. The average RMSD value of our program is 1.99 Å, whereas Glide is 1.91 Å. However, this comparison may not be completely fair to our program, because Glide gives the RMSD values of the predicted poses to energy-minimized ligand not to the native ligand [13].We also give comparison with DOCK6 (the new version of DOCK). As shown in Table 9, our program yielded the 66.7% success rate with RMSD values less than 2 Å. In contrast, DOCK6

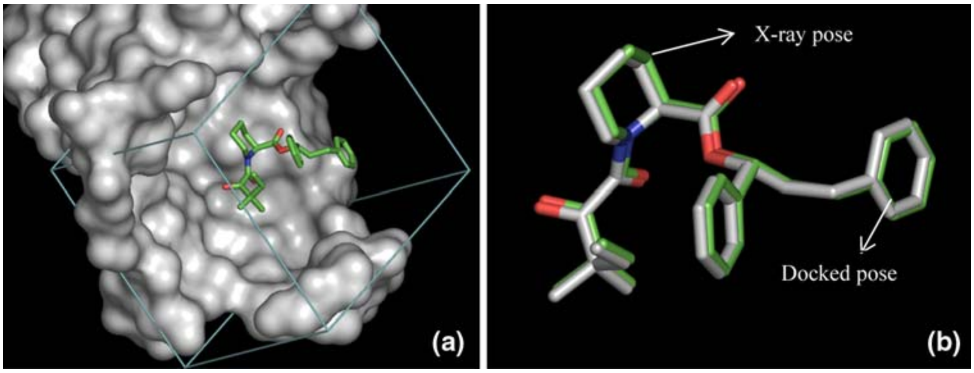

Fig. 1 Example of docking. 1FKG (0.27 Å RMSD, 11 rotatable bonds). (**a**) The defined cuboid around the bound ligand within a radius of 6.5 Å, (**b**) Docked structure was fitted to the protein-bound X-ray structure merged into the reference protein coordinates

Table 2 RMSD values and computational time for the ligands with different number of rotatable bonds

N_{rot}[a]	$N_{complexes}$[b]	min RMSD	max RMSD	avg RMSD	min Time	max Time	avg Time
0–4	43	0.21	3.54	1.30	0.56	4.25	2.05
5–9	58	0.13	8.98	2.03	1.6	13.8	5.13
10–14	18	0.17	7.71	2.46	5.46	20.41	11.90
15–19	6	0.20	7.69	2.69	10.55	34.25	17.71
20–24	7	0.27	15.37	3.24	13.16	30.39	22.12
25–29	2	6.24	13.20	9.72	32.35	42	37.18

[a] Number of rotatable bonds in the ligands

[b] Number of the complexes

got a 60.4% successful rate. For these 96 complexes, most of them have 0–9 rotatable bonds. The results show that our program is superior for molecular docking at this level of rotatable bonds.

Docking speed

An advantage of the method is its significant docking speed. Current docking programs present a decreasing performance with the increasing number of conformational degrees of freedom considered [20]. Direct comparison of docking speed is somewhat problematic because of differences in hardware. However, we can still offer a comparison from several recent works [12, 15, 21]. According to the reports of these works, docking per molecule needs 50–100 s for FlexX, DOCK and GOLD on an SGI Indigo2 R10 K processor [15]. In this paper, the average time of docking a ligand for above dataset is about 5.36 s, and the maximum time is 42 s on a SGI Fuel Workstation.

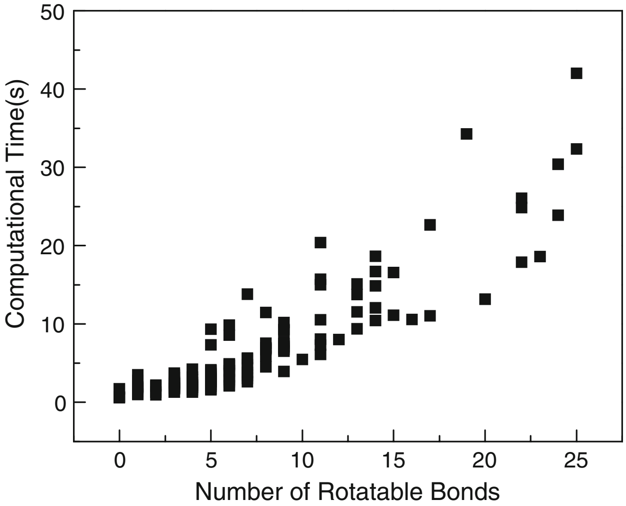

Fig. 2 The relation between computational time and the number of rotatable bonds of ligands

Docking speed is a critical issue in the application of a docking method, especially in virtual screening [15]. Using the same data set, Fig. 2 gives a plot of mean docking time of the 134 ligands from the data set versus number of rotatable bonds. It shows that the docking time is approximately in proportion to the number of rotatable bonds of ligands. This indicates that our program is fast enough for virtual screening on large-scale chemical databases.

Conclusions

This paper presents a rapid adaptive genetic algorithm for flexible molecular docking. The testing results for the test data set show that the docking time is approximately in proportion to the number of rotatable bonds of ligands, and can dock a ligand to protein target in a few seconds on SGI Fuel Workstation. The proposed method is suitable to

Table 3 RMSD values and docking time of the ligands with the different number of the heavy atoms

N_{heavy}[a]	$N_{complexes}$[b]	min RMSD	max RMSD	avg RMSD	min Time	max Time	avg Time
1–10	19	0.13	3.52	1.177	0.56	2.85	1.67
11–20	55	0.26	6.15	1.49	0.69	23.87	4.11
21–30	38	0.27	8.98	2.40	1.94	17.87	6.73
31–40	12	0.17	15.37	4.17	1.73	24.88	14.66
41–50	7	0.20	6.24	1.93	9.84	32.35	22.91
51–60	3	0.34	13.20	5.73	13.8	42	30.02

[a] Number of heavy atoms in the ligands

[b] Number of the complexes

Table 4 Comparisons with Glide, GOLD, Surflex, FlexX and DOCK6 for Docking accuracy

PDB code	This paper[a]	Glide[b]	GOLD[c]	Surflex[d]	FlexX[e]	DOCK6[f]
1AAQ	0.57	1.3	12.85	N/A	1.75	N/S
1ABE	0.24	0.17	0.86	0.27	1.16	0.15
1ACJ	0.37	0.28	4	3.89	0.49	0.26
1ACL	0.66	N/A	N/A	N/A	N/A	N/S
1ACK	1.09	N/A	4.99	1.18	N/A	0.47
1ACM	0.92	0.29	0.81	1.43	1.39	1.40
1ACO	0.71	1.02	0.86	3.39	0.96	5.25
1AEC	7.69	N/A	N/A	N/A	N/A	N/S
1AHA	0.29	0.11	0.51	0.37	0.56	0.21
1APT	0.81	0.58	1.62	N/A	1.89	N/S
1ASE	3.26	N/A	0.49	N/A	N/A	2.34
1ATL	0.56	0.94	N/A	7.01	2.06	N/S
1AZM	0.57	1.87	2.52	N/A	2.37	1.00
1BAF	3.63	0.76	6.12	6.52	8.27	3.79
1BBP	0.87	4.96	N/A	1.07	3.75	N/S
1BLH	2.39	N/A	1.95	N/A	N/A	2.82
1BMA	3.22	9.31	N/A	1	13.41	N/S
1BYB	0.70	10.49	N/A	N/A	1.62	N/S
1CBS	1.09	1.96	N/A	1.77	1.68	N/S
1CBX	0.26	0.36	0.54	0.7	1.35	1.32
1CDG	0.97	3.98	N/A	N/A	4.87	4.11
1CIL	1.95	3.82	N/A	N/A	3.85	1.13
1COM	0.99	3.64	N/A	0.86	1.62	3.86
1COY	0.44	0.28	0.86	0.54	1.06	0.28
1CPS	0.32	3	0.84	N/A	0.99	0.43
1CTR	1.51	3.56	N/A	N/A	2.82	1.72
1DBB	1.17	0.41	1.17	0.54	0.81	0.63
1DBJ	0.52	0.2	0.72	0.88	1.22	1.80
1DID	0.41	3.82	3.72	N/A	4.22	2.78
1DIE	2.20	0.79	1.03	N/A	4.71	2.08
1DR1	0.77	1.47	1.41	1.25	5.64	1.01
1DWD	0.17	1.32	1.71	1.68	1.66	N/S
1EAP	7.71	2.32	3	4.89	3.72	N/S
1EED	6.24	5.9	12.43	N/A	9.78	7.19
1EPB	1.87	1.78	2.08	2.87	2.77	N/S
1ETA	8.98	2.92	11.21	N/A	8.46	4.42

Table 4 continued

PDB code	This paper[a]	Glide[b]	GOLD[c]	Surflex[d]	FlexX[e]	DOCK6[f]
1ETR	6.34	1.48	4.23	4.05	7.24	5.13
1FEN	0.67	0.66	N/A	1.18	1.39	N/S
1FKG	0.27	1.25	1.81	1.81	7.59	5.37
1FKI	2.62	1.92	0.71	0.7	0.59	4.13
1FRP	0.57	0.27	N/A	0.75	1.89	0.95
1GHB	0.67	1.89	1.45	N/A	1.33	2.84
1GLP	1.35	0.29	1.35	N/A	6.43	0.85
1GLQ	3.60	0.29	1.35	5.68	6.43	N/S
1HDC	1.09	0.58	10.49	1.8	11.74	N/S
1HDY	1.53	1.74	0.94	0.66	N/A	1.65
1HEF	4.23	5.3	1.87	N/A	15.32	5.09
1HFC	6.63	2.24	N/A	N/A	2.51	N/S
1HRI	0.34	1.59	14.01	1.98	10.23	N/S
1HSL	0.48	1.31	0.97	0.51	0.59	1.47
1HYT	0.44	0.28	1.1	0.55	1.62	3.99
1ICN	0.86	2.34	8.63	N/A	10.52	N/S
1IDA	0.34	11.88	12.12	N/A	11.95	N/S
1IGJ	3.65	1.3	9.42	N/A	7.17	N/S
1IMB	0.67	0.89	N/A	N/A	4.71	N/S
1IVE	2.66	2.61	2.16	N/A	5.34	1.79
1LAH	0.21	0.13	N/A	0.3	0.28	0.14
1LCP	2.29	1.98	N/A	2.01	1.65	1.97
1LDM	1.42	0.3	1	0.44	0.74	1.79
1LIC	0.43	4.87	10.78	3.46	5.07	N/S
1LMO	5.75	0.93	N/A	N/A	4.49	3.78
1LNA	6.15	0.95	N/A	0.88	5.4	3.00
1LPM	3.48	N/A	N/A	1.87	N/A	N/S
1LST	0.13	0.14	0.87	0.33	0.71	0.62
1MCR	2.16	4.33	6.23	N/A	10.04	1.95
1MDR	0.58	0.52	0.36	0.68	0.88	1.89
1MMQ	4.32	0.92	N/A	N/A	0.52	N/S
1MRG	0.47	0.3	N/A	0.7	0.81	0.35
1MRK	0.69	1.2	1.01	0.85	3.55	1.61
1MUP	0.50	4.37	3.96	N/A	3.82	3.14
1NCO	0.31	6.99	N/A	8.26	5.85	1.11
1NIS	0.41	0.97	4.29	N/A	1.41	2.56
1PBD	3.52	0.21	0.57	N/A	0.33	0.79
1PHA	0.67	0.69	1.24	N/A	N/A	N/S
1PHD	3.51	1.22	0.85	N/A	0.65	N/S
1PHG	2.14	4.32	1.35	4.44	4.74	5.48
1POC	15.37	5.09	1.27	N/A	9.25	4.45
1RDS	2.14	3.75	4.78	9.83	4.89	N/S
1RNE	13.20	10.08	2	N/A	12.24	1.51
1ROB	1.48	1.85	3.75	0.82	7.7	0.88
1SLT	1.44	0.51	0.78	N/A	1.63	4.07
1SNC	0.87	1.91	N/A	4.92	7.48	N/S
1SRJ	3.40	0.58	0.42	0.39	2.36	2.04
1STP	5.66	0.59	0.69	0.51	0.65	0.32
1TDB	2.12	1.46	10.48	N/A	10.1	1.91

Table 4 continued

PDB code	This paper[a]	Glide[b]	GOLD[c]	Surflex[d]	FlexX[e]	DOCK6[f]
1TKA	2.75	2.28	1.88	1.96	1.17	N/S
1TMN	7.02	2.8	1.68	1.3	0.86	N/S
1TNG	0.28	0.19	N/A	0.22	1.93	0.16
1TNI	1.17	2.18	N/A	2.97	2.71	1.20
1TNL	2.21	0.23	N/A	2.26	0.71	1.06
1TPH	1.12	0.2	N/A	N/A	1.5	1.21
1TPP	2.10	1.12	0.43	N/A	1.11	2.38
1TRK	7.45	1.64	N/A	1.22	1.57	1.29
1TYL	3.54	1.06	N/A	N/A	2.34	2.98
1UKZ	0.75	0.37	N/A	0.77	0.94	0.79
1ULB	1.80	0.28	0.32	0.77	3.37	0.36
1WAP	0.28	0.12	N/A	0.3	0.57	0.21
1XID	1.78	4.3	0.92	N/A	2.01	3.52
1XIE	2.32	3.86	0.69	N/A	1.94	3.11
2ADA	0.43	0.53	0.4	0.32	0.67	N/S
2AK3	2.13	0.71	5.08	0.6	0.91	2.54
2CGR	6.09	0.38	0.99	1.63	3.53	1.30
2CHT	1.97	0.42	0.59	0.42	4.58	1.14
2CMD	0.48	0.65	N/A	1.6	3.75	1.14
2CTC	0.40	1.61	0.32	0.38	1.97	1.12
2DBL	2.06	0.69	1.31	0.81	1.49	6.31
2GBP	0.35	0.15	N/A	0.63	0.92	0.51
2LGS	2.13	7.55	N/A	1.22	4.63	4.08
2MCP	1.28	1.3	4.37	N/A	2.07	1.18
2MTH	1.20	N/A	10.12	N/A	N/A	3.58
2PHH	0.64	0.38	0.72	0.44	0.43	1.52
2PK4	1.31	0.86	1.34	N/A	1.66	0.99
2PLV	0.73	1.88	13.92	N/A	7.85	N/S
2R07	2.16	0.48	8.23	1.35	11.63	N/S
2SIM	0.86	0.92	0.92	1.1	1.99	0.99
2YHX	1.50	3.84	1.19	N/A	2.25	4.96
3AAH	0.46	N/A	0.42	0.68	N/A	5.53
3CLA	4.24	N/A	5.45	N/A	N/A	4.29
3CPA	0.64	2.4	1.58	1.9	2.53	N/S
3GCH	1.99	N/A	2.64	N/A	N/A	3.47
3HVT	0.84	0.77	1.12	1.64	10.26	0.69
3PTB	2.11	0.27	0.96	0.54	0.55	1.38
3TPI	0.41	0.49	0.8	0.52	1.07	0.35
4CTS	0.85	0.19	1.57	2.2	1.53	1.49
4DFR	1.09	1.12	1.44	1.6	1.4	N/S
4EST	6.61	N/A	1.38	N/A	N/A	N/S
4FAB	0.92	4.5	5.69	N/A	4.95	1.11
4PHV	0.20	0.38	1.11	N/A	1.12	0.83
5P2P	0.27	1.82	1.55	N/A	1	N/S
6ABP	0.33	0.4	1.08	0.28	1.12	0.26
6RNT	1.29	2.22	1.2	7.03	4.79	2.86
6RSA	1.07	N/A	4.42	0.78	N/A	0.84
7TIM	1.22	0.14	0.78	1.2	1.49	N/S
8GCH	1.51	0.3	0.86	4.51	8.91	3.29

[a] Best pose (Å) for energy score, not the best result corresponding to RMSD

[b] The results of Friesner and co-workers [13]

[c] The results of Jones and co-workers [12]

[d] The results of Jain [15]

[e] The results of Kramer and co-workers [14]

[f] The results by running DOCK6 with the default parameter

N/A, no result in the published papers

N/S, program running error or no solutions

Table 5 Comparisons with GOLD for RMSD values of the ligands with the different number of rotatable bonds

N_{rot} ($N_{complexes}$)	Average RMSD (Å)		$N_{complexes}$ (≤0.5 Å/≤2 Å/≤2.5 Å)	
	This paper	GOLD	This paper	GOLD
0–4(32)	1.26	1.81	8/25/27	5/24/25
5–9(43)	1.91	3.13	9/26/35	3/27/28
10–14(10)	3.07	2.17	2/5/6	0/7/7
15–19(6)	2.69	5.86	3/4/4	0/3/3
20–24(6)	3.67	5.51	1/4/4	0/4/4
25–29(2)	9.72	7.22	0/0/0	0/1/1
Total (99)	2.12	3.00	22/64/76	8/66/68

Table 6 Comparisons with FlexX for RMSD values of the ligands with the different number of rotatable bonds

N_{rot} ($N_{complexes}$)	Average RMSD (Å)		$N_{complexes}$ (≤0.5 Å/≤2 Å/≤2.5 Å)	
	This paper	FlexX	This paper	FlexX
0–4(37)	1.28	2.07	11/27/32	4/25/27
5–9(53)	1.98	3.63	11/34/42	0/25/29
10–14(17)	2.57	4.60	3/10/11	0/5/6
15–19(4)	0.46	7.17	3/4/4	0/1/1
20–24(7)	3.24	5.53	0/5/5	0/4/4
25–29(2)	9.72	11.01	0/0/0	0/0/0
Total (120)	2.00	3.64	28/80/94	4/60/67

Table 7 Comparisons with Surflex for RMSD values and computational time of the ligands with the different number of rotatable bonds

N_{rot} ($N_{complexes}$)	Average RMSD (Å)		$N_{complexes}$ (≤0.5 Å/≤2 Å/≤2.5 Å)	
	This paper	Surflex	This paper	Surflex
0–4(31)	1.05	1.07	12/30/30	10/28/28
5–9(36)	1.80	1.71	8/28/28	3/30/30
10–14(13)	2.63	3.79	3/8/8	0/7/7
15–19(1)	0.43	3.46	1/1/1	0/0/0
Total (81)	1.63	1.82	24/67/67	13/65/65

Table 8 Comparison with Glide for RMSD values of the ligands with the different number of rotatable bonds

N_{rot} ($N_{complexes}$)	Average RMSD (Å)		$N_{complexes}$ (≤0.5 Å/≤2 Å/≤2.5 Å)	
	This paper	Glide	This paper	Glide
0–4(38)	1.28	1.32	11/28/33	20/31/32
5–9(54)	1.96	1.43	11/35/43	12/41/45
10–14(17)	2.57	2.54	3/10/11	2/10/11
15–19(4)	0.46	4.87	3/4/4	1/2/2
20–24(7)	3.24	3.78	1/5/5	0/4/4
25–29(2)	9.72	7.99	0/0/0	0/0/0
Total (122)	1.99	1.91	29/82/96	35/87/94

Table 9 Comparison of with DOCK6 for RMSD values of the ligands with the different number of rotatable bonds

N_{rot} ($N_{complexes}$)	Average RMSD (Å)		$N_{complexes}$ (≤0.5 Å/≤2 Å/≤2.5 Å)	
	This paper	DOCK6	This paper	DOCK6
0–4(42)	1.25	1.56	12/32/37	11/33/34
5–9(41)	2.08	2.32	9/25/33	3/20/23
10–14(8)	2.16	3.17	2/6/6	0/3/3
15–19(1)	0.20	0.83	1/1/1	0/1/1
20–24(2)	9.80	4.77	0/0/0	0/0/0
25–29(2)	9.71	4.35	0/0/0	0/1/1
Total (96)	2.02	2.14	24/64/77	14/58/62

virtual screening. However, there are still several aspects we should improve, such as developing better docking strategies, improving score functions and considering the flexibility of protein target, the further work is under doing.

Acknowledgements The authors gratefully acknowledge financial support for this work from the National Natural Science Foundation (No. 10772042), the Subsidized by the Special Funds for Major State Basic Research Project (No. 2004CB518901) and High Science and Technology Project (No. 2006AA01A124) of China.

References

1. Wolff ME (1994) Burger's medicinal chemistry and drug discovery, Volume 1, Principles and practice, 5th edn. Wiley, New York, pp 497–571
2. Ewing TJ, Makino S, Skillman AG, Kuntz ID (2001) J Comput Aided Mol Des 15:411–428
3. Meng EC, Gschwend DA, Blaney JM, Kuntz ID (1993) Proteins 17:266–278
4. Goodsell DS, Morris GM, Olson AJ (1996) J Mol Recognit 9:1–5
5. Morris GM, Goodsell DS, Huey R, Olson AJ (1996) J Comput Aided Mol Des 10:293–304
6. McMartin C, Bohacek RS (1997) J Comput Aided Mol Des 11:333–344
7. Baxter CA, Murray CW, Clark DE, Westhead DR, Eldridge MD (1998) Proteins 33:367–382
8. Liu M, Wang S (1999) J Comput Aided Mol Des 13:435–451
9. Rarey M, Kramer B, Lengauer T, Klebe G (1996) J Mol Biol 261:470–489
10. Lee K, Czaplewski C, Kim SY, Lee J (2005) J Comput Chem 26:78–87
11. Kuntz ID, Blaney JM, Oatley SJ, Langridge R, Ferrin TE (1982) J Mol Biol 161:269–288
12. Jones G, Willett P, Glen RC, Leach AR, Taylor R (1997) J Mol Biol 267:727–748
13. Friesner RA, Banks JL, Murphy RB, Halgren TA, Klicic JJ, Mainz DT, Repasky MP, Knoll EH, Shelley M, Perry JK, Shaw DE, Francis P, Shenkin PS (2004) J Med Chem 47:1739–1749
14. Kramer B, Rarey M, Lengauer T (1999) Proteins 37:228–241
15. Jain AN (2003) J Med Chem 46:499–511
16. Venkayya VB (1997) Int. J Numer Meth Eng 13:203–228
17. Shannon CE (1948) Bell Syst Technical J 27:379–423, 623–656
18. Cover TM, Thomas JA (1991) Elements of information theory. Wiley, New York
19. Dennis A, Holt JIL, Yamashita DS, Oh HJ, Konialian AL, Yen HK, Rozamus LW, Brandt M, Bossard MJ, Levy MA, Eggleston DS, Liang J, Schultz LW, Stout TJ, Clardy J (1993) J Am Chem Soc 115:9925–9938
20. McConkey BJ, Sobolev V, Edelman M (2002) Bioinformatics 18:1365–1373
21. Bissantz C, Folkers G, Rognan D (2000) J Med Chem 43: 4759–4767