J Math Chem (2009) 46:182–198
DOI 10.1007/s10910-008-9454-8

ORIGINAL PAPER

A novel conformation optimization model and algorithm for structure-based drug design

Ling Kang · Honglin Li · Xiaoyu Zhao · Hualiang Jiang · Xicheng Wang

Received: 17 April 2008 / Accepted: 11 August 2008 / Published online: 30 August 2008

Abstract In this paper, we present a multi-scale optimization model and an entropy-based genetic algorithm for molecular docking. In this model, we introduce to the refined docking design a concept of residue groups based on induced-fit and adopt a combination of conformations in different scales. A new iteration scheme, in conjunction with multi-population evolution strategy, entropy-based searching technique with narrowing down space and the quasi-exact penalty function, is developed to address the optimization problem for molecular docking. A new docking program that accounts for protein flexibility has also been developed. The docking results indicate that the method can be efficiently employed in structure-based drug design.

Keywords Information entropy · Genetic algorithm · Molecular docking · Multi-scale optimization model · Residue groups

1 Introduction

Structure-based drug design (SBDD) is pivotal technique to drug discovery. It came into existence in the early 1980s as a result of multidisciplinary efforts. The techniques

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

L. Kang · H. Li · X. Zhao · X. Wang (✉)
Department of Engineering Mechanics, State Key Laboratory of Structural Analysis for Industrial Equipment, Dalian University of Technology, Dalian 116023, People's Republic of 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, People's Republic of China
e-mail: hljiang@mail.shcnc.ac.cn

in this field are by no means yet mature. However they developed quickly and there have been a lot of successes to date. It is widely accepted that drug discovery research has become increasingly data intensive and require more informatics tools to enhance the process. Novel theoretical approaches could become a major avenue for drug discovery efforts at the post-genomic era. Molecular docking serves as a main method of SBDD to simulate the interactions of two molecules (such as ligand and receptor) and to predict their binding mode and affinity. In recent years, molecular docking has emerged as an important technology in the field. A fundamental problem with molecular docking is that orientation space is very large and grows combinatorially with the number of degrees of freedom of the interacting molecules. Therefore, simpler and efficient methods are continuously being researched into.

Over the past two decades, many automated docking approaches have been developed and can be roughly divided into rigid-docking, flexible ligand-docking and flexible protein-docking methods. The rigid-docking methods, such as the previous DOCK program [1], treat both ligands and proteins as rigid. In contrast, ligands are considered flexible and proteins rigid for flexible ligand-docking methods, including evolutionary algorithms [2, 3], simulated annealing [4], the fragment based approach [5], and other algorithms [6–8]. Despite the diversity of the scoring functions and search algorithms used in these methods, they are either flexible or rigid docking methods. The consideration of protein flexibility is less advanced than that of ligand flexibility. Protein flexibility has been ignored in most docking programs since the evaluation of protein-ligand interaction energies at all possible docking configurations is a prohibitively time-consuming process. However, it has become increasingly clear that protein flexibility plays a paramount role in protein-ligand complex formation and should be considered during the docking process [9, 10].

Molecular docking is a difficult optimization problem. It contains a large number of design variables. The objective function is a highly nonlinear function, and it is an implicit function of the design variables. To solve it may involve a costly computational effort. However, multi-scale methods can lead to efficient solution schemes for overcoming these difficulties. The multi-scale method relies on a construct known as space decomposition. This involves the application of a scaling operation to the original data. The effect of the scaling operator is to remove the information in the data corresponding to the highest level of detail. As fine detail is removed, the larger scale features in the data are emphasized. Repeated application of the scaling operator returns increasingly larger structures in the data until either the required scale has been achieved, or there is no more information left in the resulting decomposition. The selection of the scaling operator is of crucial importance if multi-scale analysis via scale-space decomposition is to be completed successfully. We can consider the conformational changes of the ligand and the receptor with these different scales and elucidate a more effective and efficient multi-scale docking method.

In the present report, we establish a multi-scale optimization model for molecular docking, through the introduction of the concept of the residue groups in the receptor. This model includes both current semi-flexibility docking and refined docking. An entropy-based genetic algorithm is developed to solve the optimization model of molecular docking. The algorithm proposes a new iteration scheme in conjunction with the multi-population evolution, entropy-based searching technique with narrowing down space and the quasi-exact penalty function.

In order to evaluate the new optimization model and docking method, we have conducted a numerical experiment with 52 protein-ligand complexes from the publicly available GOLD test set [2]. Comparisons with four docking programs, namely Glide [11], GOLD [2], FlexX [5] and DOCK6 [1, 12], show that docking accuracy has been significantly improved by the new model and algorithm.

2 Multi-scale optimization model for molecular docking

2.1 Coarse-scale optimization model for molecular docking

Molecular docking is fundamentally an optimization problem of predicting the interaction energy between small organic molecules and biological receptors. Mathematically, it can be written as follows

$$\begin{aligned} &\min \ f(\mathbf{d}) \\ &s.t. \ g_j(\mathbf{d}) \le 0, \ \ j = 1, 2, \ldots, q \end{aligned} \tag{1}$$

where $\mathbf{d}$ is a vector of design variables that is comprised of the state variables of molecules. The objective function $f(\mathbf{d})$ is the interaction energy between ligand and protein.

We assume that the ligand being studied is flexible and the corresponding receptor is rigid. The design variable $\mathbf{d}$ can then be described as follows

$$\mathbf{d} = \left\{T_x, T_y, T_z, R_x, R_y, R_z, T_{b1}, T_{b2}, \ldots, T_{bn}\right\}^T \tag{2}$$

where $T_x, T_y, T_z, R_x, R_y, R_z$ are the position coordinates and rotational angles of the anchor for the matching-based orientation search, and $T_{b1}, T_{b2}, \ldots, T_{bn}$ are the torsional angles of the rotatable bonds required for flexible ligand docking.

The objective function $f(\mathbf{d})$ consists of the Coulomb and van der Waals terms of force field functions:

$$f\left(\mathbf{d}\right) = \sum_{i=1}^{n_{lig}} \sum_{j=1}^{n_{rec}} \left(\frac{A_{ij}}{r_{ij}^{12}} - \frac{B_{ij}}{r_{ij}^{6}} + 332.0 \frac{q_i q_j}{D r_{ij}} \right) \tag{3}$$

where each term is a double sum over the ligand atom i and the receptor atom j, n_{lig}, n_{rec} are the number of atoms in the ligand and the receptor, respectively; A_{ij}, B_{ij} are van der Waals repulsion and attraction parameters, r_{ij} is the distance between atoms i and j, q_i, q_j are the point charges on atoms i and j, D is dielectric function, and 332.0 is a factor that converts the electrostatic energy into kilocalories per mole. Equation 3 contains the intermolecular terms present in the AMBER [13] molecular mechanics function, except for an explicit hydrogen-bonding term. We assume that hydrogen bond energies can largely be accounted for in the electrostatic term.

The constraints $g_j(\mathbf{d})$, $j = 1, 2, \ldots, q$ may be represented as the size limits of the design variables and certain behavior constraints of the molecule exist such as ’the rule of 5′ [14], i.e., they are less than 5 H-bond donors, 10 H-bond acceptors, and the molecular weight (MWT) is less than 500 and the Log P (CLogP) is less than 5 (or MlogP is less than 4.15), and so on. The constraints are here the size limits of the design variables only and shown as follows:

$$\begin{cases} \underline{T_x} \le T_x \le \overline{T_x} \\ \underline{T_y} \le T_y \le \overline{T_y} \\ \underline{T_z} \le T_z \le \overline{T_z} \\ -\pi \le angle \le \pi, \quad angle = R_x, R_y, R_z, T_{b1}, \ldots, T_{bn} \end{cases} \tag{4}$$

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 shown in Eq. 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 $(T_{bi}, i = 1, 2, \ldots, n,\ n$ 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 space 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 space 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.

2.2 Refined-scale optimization model for molecular docking

In the above model (Eqs. 1–4), we only consider the ligand flexibility. However, changes in the receptor structure upon ligand binding are frequently observed [15], and as such, both the structure of the ligand and the receptor change during the binding process.

The rapid development of increasingly powerful computational methods is making it possible to consider protein flexibility in docking programs. In order to fully clarify this notion, we cite the following approaches that consider the receptor flexibility in docking process. One such approach involves the use of conformational ensembles to generate energy-weighted or geometry-weighted average grids [16, 17], which require several conformationally distinct protein structures. Other commonly-used approaches include the sampling of predetermined side chain rotamer libraries [18, 19], the construction of protein structures using discrete protein conformations [20] or molecular modeling [21], the use of relaxed complex methods based on molecular dynamics [22, 23], and the use of soft docking [24].

We introduced the concept of the residue groups in the receptor. The residues within the binding site are divided into several residue groups, and the center coordinates of each residue group introduced into the optimization process as design variables. Thus we establish a refined-scale optimization model based on the problem (1), and added the following design variables:

$$\left\{C_{1x}, C_{1y}, C_{1z}, \ldots, C_{mx}, C_{my}, C_{mz}\right\}^T \tag{5}$$

where m is the number of residue groups, and (C_{ix}, C_{iy}, C_{iz}) $(i = 1, 2, \ldots, m)$ are the positional coordinates of the center for each residue group. And the constraints added are introduced into $g(\mathbf{d})$ as:

$$\begin{cases} \underline{C}_{ix} \leq C_{ix} \leq \overline{C}_{ix} \; i = 1, \ldots, m \\ \underline{C}_{iy} \leq C_{iy} \leq \overline{C}_{iy} \; i = 1, \ldots, m \\ \underline{C}_{iz} \leq C_{iz} \leq \overline{C}_{iz} \; i = 1, \ldots, m \end{cases} \tag{6}$$

3 Entropy-based adaptive genetic algorithm for molecular docking

The objective function of problem (1) is nonlinear and implicit function of the design variables 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, and their successful adaptation and implementation in a series of optimal design problems. The principal advantages of the genetic algorithms reside in the fact that no sensitivity analysis is required and global optimal solution can be obtained. In addition, genetic algorithms also have advantages such as simple formulation, easy programming. But genetic search process is the time-consuming work, so that hindered them from applied to multi-scale molecular docking optimization problems, 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 new method 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.

It is very difficult to solve the problem (1) directly due to the much more constraints. In order to solve it efficiently, we transform the constrained optimization model into unconstrained model.

3.1 Transformation of the optimization model

First, we introduce some definitions and theorems.

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

$$E(G) = (1/\psi) \ln \sum_{j=1}^{q} \exp(\psi g_j(\mathbf{d})) \tag{7}$$

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

$$\begin{aligned} &\min \ f(\mathbf{d}) \\ &s.t. \ \ g_\psi(\mathbf{d}) = (1/\psi) \ln \sum_{j=1}^{q} \exp(\psi g_j(\mathbf{d})) \le 0 \end{aligned} \tag{8}$$

Definition 2 If, for any $G(\mathbf{d}) = \{g_1(\mathbf{d}), g_2(\mathbf{d}), \ldots, g_q(\mathbf{d})\}$, and $\overline{G}(\mathbf{d}) = \{\overline{g}_1(\mathbf{d}), \overline{g}_2(\mathbf{d}), \ldots, \overline{g}_q(\mathbf{d})\}$, $G(\mathbf{d}), \overline{G}(\mathbf{d}) \in E^q$ with $g_j(\mathbf{d}) \le \overline{g}_j(\mathbf{d})$, $j = 1, 2, \ldots, q$, and there exists at least one j_0, $(1 \le j_0 \le q)$, such that $g_{j_0}(\mathbf{d}) < \overline{g}_{j_0}(\mathbf{d})$, then $G(\mathbf{d}) \le \overline{G}(\mathbf{d})$ or, simply $G \le \overline{G}$.

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

Lemma 1 *The PCE function $E(G)$ is a strictly monotone increasing function of G, and if $\psi \to \infty$ then*

$$(1/\psi) \ln \sum_{j=1}^{q} \exp(\psi g_j(\mathbf{d})) = \max \ g_j(\mathbf{d}) \ j = 1, 2, \ldots, q \tag{9}$$

Proof Let

$$G = \{g_j(\mathbf{d})\} \le \overline{G} = \{\overline{g}_j(\mathbf{d})\}, \ j = 1, 2, \ldots, q \tag{10}$$

By Definition 2

$$g_j(\mathbf{d}) \le \overline{g}_j(\mathbf{d}), \ j = 1, 2, \ldots, q \tag{11}$$

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

$$g_{j_0}(\mathbf{d}) < \overline{g}_{j_0}(\mathbf{d}) \tag{12}$$

Then for $\psi > 0$,

$$\psi g_{j_0}(\mathbf{d}) < \psi \overline{g}_{j_0}(\mathbf{d}) \tag{13}$$

$$\exp(\psi g_{j_0}(\mathbf{d})) < \exp(\psi \overline{g}_{j_0}(\mathbf{d})) \tag{14}$$

Hence

$$\sum_{j=1}^{q} \exp(\psi g_j(\mathbf{d})) < \sum_{j=1}^{q} \exp(\psi \overline{g}_j(\mathbf{d})) \tag{15}$$

Taking logarithms on both sides and dividing by ψ

$$E(G) = (1/\psi) \ln \sum_{j=1}^{q} \exp(\psi g_j(\mathbf{d})) < (1/\psi) \ln \sum_{j=1}^{q} \exp(\psi \overline{g}_j(\mathbf{d})) \tag{16}$$

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

$$E_G = \left\{ e^{g_1(\mathbf{d})}, e^{g_2(\mathbf{d})}, \ldots, e^{g_q(\mathbf{d})} \right\}^T \tag{17}$$

is given by

$$N_\psi(E_G) = \left(\sum_{j=1}^{q} e^{\psi g_j(\mathbf{d})} \right)^{(1/\psi)} \tag{18}$$

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

$$N_\infty(E_G) = \lim_{\psi \to \infty} N_\psi(E_G) \tag{19}$$

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

$$N_s < N_r \quad \text{for } r < s \tag{20}$$

The importance of this inequality is that it holds also in the limit as $s \to \infty$. Thus, Eq. 19 may be written as

$$N_\infty\,(E_G) = \max\left(e^{g_j(\mathbf{d})}\right) < N_r(E_G) \tag{21}$$

Taking logarithms on both side of Eq. 21 and substituting from Eqs. 18 and 19 gives

$$\lim_{\psi \to \infty} (1/\psi) \ln \sum_{j=1}^{q} \exp(\psi g_j(\mathbf{d})) = \max(g_j(\mathbf{d})) \tag{22}$$

and the proof is completed.

The PCE function plays an important role in the proposed method. The following theorem aids understanding of its properties.

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

$$\begin{aligned} &\min\ f(\mathbf{d}) \\ &s.t.\ \ g_{\psi}(\mathbf{d}) = (1/\psi)\ln\sum_{j=1}^{q}\exp(\psi g_j(\mathbf{d})) \le 0 \end{aligned} \tag{23}$$

have the same Kuhn-Tucker points.

Proof The Lagrange augmented function problem (23) is

$$L(\mathbf{d},\mu) = f(\mathbf{d}) + (\mu/\psi)\ln\sum_{j=1}^{q}\exp(\psi g_j(\mathbf{d})) \tag{24}$$

where $\mu > 0$ is the Lagrange multiplier of corresponding constraint. The Kuhn-Tucker condition for problem (23) is given as

$$\partial f(\mathbf{d})/\partial d_i + (\mu/\psi)\left\{\sum_{j=1}^{q}\exp(\psi g_j(\mathbf{d}))\cdot\partial g_j(\mathbf{d})/\partial d_i\right\}\Big/\sum_{j=1}^{q}\exp\left[\psi g_j(\mathbf{d})\right] = 0 \tag{25}$$

$$(1/\psi)\ln\sum_{j=1}^{q}\exp\left(\psi g_j(\mathbf{d})\right) \le 0 \tag{26}$$

$$(\mu/\psi)\ln\sum_{j=1}^{q}\exp\left(\psi g_j(\mathbf{d})\right) = 0,\ \mu \ge 0 \tag{27}$$

By means of Lemma 1 and Eq. 26, if $\psi \to \infty$, then

$$(1/\psi)\ln\sum_{j=1}^{q}\exp(\psi g_j(\mathbf{d})) = \max\ g_j(\mathbf{d})\ \le 0\,,\quad j = 1, 2, \ldots, q \tag{28}$$

i.e.

$$g_j(\mathbf{d}) \le 0 \tag{29}$$

Substituting

$$\mu = \psi,\ \mu_j = \frac{\exp\left[\psi g_j(\mathbf{d})\right]}{\sum_{j=1}^{q}\exp\left[\psi g_j(\mathbf{d})\right]} \tag{30}$$

into Eq. 25 gives

$$\frac{\partial f(\mathbf{d})}{\partial d_i} + \sum_{j=1}^{q} \mu_j \frac{\partial g_j(\mathbf{d})}{\partial d_i} = 0 \tag{31}$$

Combining Eqs. 27 and 30, if $\psi \to \infty$, then

$$\begin{cases} g_j(\mathbf{d}) = 0 & \text{if } \mu_j > 0 \\ g_j(\mathbf{d}) < 0 & \text{if } \mu_j = 0 \end{cases} \tag{32}$$

Equations 29, 31 and 32 are identical the Kuhn-Tucker condition of the problem (1). Hence the problems (23) and (1) have the same Kuhn-Tucker points and vice versa. The theorem 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 [25]. Theorem 1 shows that to solve problem (1) with multi constraints can be substituted by solving a simple problem (23) with a single constraint only. Unlike some optimality criteria methods, there is no need to find active constraints. The μ_j in Eq. 30 can give the active level of the constraints.

Problem (23) can be solved by using quasi-exact penalty function:

$$\varphi_\psi(\mathbf{d}) = f(\mathbf{d}) + (\alpha/\psi) \ln \left\{ 1 + \sum_{i=1}^{q} \exp(\psi g_i(\mathbf{d})) \right\} \tag{33}$$

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

Fitness function of genetic algorithm by means of Eq. 33 may be written as:

$$\max \quad F(\mathbf{d}) = C - \varphi_\psi(\mathbf{d}) \tag{34}$$

Problem (34) 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 $F > 0$.

3.2 Entropy-based adaptive genetic algorithm

Genetic algorithm (GA) was formally introduced in the 1970s by John Holland [26], which uses ideas based on natural genetics and biological evolution. The three most important aspects of using genetic algorithm are: the fitness function, the genetic representation, and the definition and implementation of the genetic operators. Some molecular docking programs are based on GA such as GOLD [2].

Based on the traditional genetic algorithm, we develop an entropy-based adaptive genetic algorithm. 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.

3.2.1 Multi-population genetic strategy

Multi-population genetic strategy is used in our algorithm to keep diversity among different populations and avoid premature problem to some extent. That is, M populations, each containing N individuals, are generated randomly with all the same searching space. Selection and mutation operations for each population are performed independently while crossover operation is carried out between different populations.

For multi-population evolution, the genetic algorithm begins from generating arbitrarily M populations with all the same searching space, i.e. initial 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 search, i.e. to solve the following optimum problem:

$$\min \ -F_j(\mathbf{d}), \ j = 1, 2, \ldots, M \tag{35}$$

Problem (35) is a multi-objective optimization, which is very difficult to solve completely.

3.2.2 Entropy-based searching technique

Shannon's theorem [27, 28] has wide-ranging applications in both communications and data storage applications. This theorem is of foundational importance to the modern field of information theory. There are similarities between the process of optimization and communication of information theory.

By information entropy principle, an entropy-based optimization model can be constructed as follows:

$$\begin{cases} \min - \sum\limits_{j=1}^{M} p_j F_j(\mathbf{d}) \\ \min H = - \sum\limits_{j=1}^{M} p_j \ln(p_j) \\ s.t. \sum\limits_{j=1}^{M} p_j = 1, p_j \in [0, 1] \end{cases} \tag{36}$$

where H is the information entropy, p_j is here defined as a probability that the optimal solution of the problem (35) occurs in the population j. It can easily be proved that the optimization problem (35) and (36) both have the same optimal solution. The solution

p_j of Eq. 36 can be obtained explicitly.

$$p_j^* = \exp(\gamma F_j(\mathbf{d})) \Big/ \sum_{j=1}^{M} \exp(\gamma F_j(\mathbf{d})) \tag{37}$$

in which

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

γ is here called as the quasi-weight coefficient. The $1 - p_j$ can be used as the coefficients of narrowing searching space in the modified genetic algorithm. For the genetic algorithm with narrowing of the search space, we need only to know the efficient narrowing coefficients for the searched space.

Using multi-population evolution with narrowing down space, M populations with N members are generated in the given space. Note that the design space is defined as initial searching space $S(0)$. During genetic evolution, the searching space of each population is narrowed according to the following equation:

$$\begin{aligned} &S(K) = (1 - p_j)S(K - 1) \\ &\underline{d}_i(K) = \max\left\{\left[d_i^*(K) - 0.5(1 - p_j)S(K)\right], \underline{d}_i(0)\right\} \\ &\overline{d}_i(K) = \min\left\{\left[d_i^*(K) + 0.5(1 - p_j)S(K)\right], \overline{d}_i(0)\right\} \end{aligned} \tag{39}$$

where $\underline{d}_i(K)$ and $\overline{d}_i(K)$ are the modified lower and upper limits of ith design variable at Kthiteration, respectively. $d_i^*(K)$ is the value of design variable i of the best member in the population j.

The traditional termination of genetic algorithm may be determined by some criteria, e.g., a prescribed maximum number of generations or function evaluations, or a preset precision for the optimum. Here, the narrowed searching space is regarded as the terminal criteria. Equation 39 is employed to control the narrowing of design space for each population. If $\left(1 - p_l^*\right) = 0$, 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.

Entropy-based searching technique with narrowing down space is taken to control the size of searching space, and the contracted space is employed as convergence criterion to terminate the evolution process so that it can speed up the convergence to obtain the optimal solution.

3.2.3 Adaptive strategy

In traditional genetic algorithm, the probabilities p_c and p_m of the crossover and mutation operators must be provided 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 $0.7 \leq p_c \leq 1.0,\ 0.0 \leq p_m \leq 0.1$).

3.2.4 Algorithm organization

The algorithm consists of the following steps:

Step 1. Generate M initial populations and implement the duplicate operator.

Step 2. Perform genetic 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. If it has, go to step 4; otherwise, return to step 2.

Step 4. Output the optimization results and stop.

4 Computational performance

4.1 Test data set

A subset of the GOLD data set, originally proposed by Jones et al. [2], was chosen to exam our method. 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 all structural water molecules, ligands, cofactors, and metal ions from the receptor pdb file. Next, a mol2 file was generated by the addition of requisite hydrogen atoms and Kollman charge using Sybyl6.8 [29]. Residues around the bound ligand within a radius of 6.5 Å were isolated from the protein, to define the active site. The ligands were then prepared by adding hydrogen atoms and Gasteiger-Marsili atomic charges adopted in Sybyl6.8 [29]. The rotatable bonds of the ligands ranged widely from 0 to 25, with greater than 88% of the ligands possessing less than 15 such bonds.

Based on the coarse and refined-scale optimization models and algorithms, we developed a new molecular docking program. According to test the proposed docking method, docking accuracy and the efficiency are selected as the primary evaluating criteria [30].

4.2 Docking accuracy

The key characteristic of a good docking program is its ability to reproduce the experimental binding modes of ligands [31]. The success of a docking algorithm in predicting a ligand-binding pose is normally measured in terms of the root-mean-square deviation (RMSD) between the experimentally observed heavy-atom positions of the ligands and the ones predicted by the algorithm [32]. The RMSD is defined as:

$$\left\{\sum_{i=1}^{N}\left[(x_i^0-x_i)^2+(y_i^0-y_i)^2+(z_i^0-z_i)^2\right]\Big/N\right\}^{1/2} \tag{40}$$

where N is the heavy atom number of a ligand, and (x_i^0, y_i^0, z_i^0) and (x_i, y_i, z_i) are the coordinates of the ith atom of X-ray crystal and docked structures, respectively. In general, the docking accuracy is acceptable if the RMSD value between the docked model and X-ray crystal structure is less than 2.0 Å.

Tables 1 and 2 summarize the performance of the two multi-scale docking methods against the 52-complex dataset. The refined-scale approach is significantly better than the coarse-scale approach. As shown, the refined-scale docking program yielded a 57.7% excellent result with a RMSD values below 1.0 Å. Nearly 90.4% of the results are within 2.0 Å. For the coarse-scale approach, 46.2% of the solutions have a RMSD below 1.0 Å and 65.4% of the solutions have a RMSD within 2.0 Å. Further, docking accuracy decreases with increasing ligand flexibility. For example, for refined-scale docking, there are 73.7% results with RMSD values below 1.0 Å for ligands with 0 to 4 rotatable bonds, while this value is 54.5% for ligands with 5 to 9 rotatable bonds.

Table 3 shows the heavy-atom RMSD of the best scored (lowest-energy) results, where the values of Glide, GOLD and FlexX, were obtained from Friesner et al. [11], Jones et al. [2], Kramer et al. [5], respectively. And the values of DOCK6 were got

Table 1 Results for our program based on the coarse-scale model

N_{rot}[a]	$N_{complexes}$[b]	$N_{complexes}$ having an RMSD in the listed range (Å)					
		≤0.5	>0.5, ≤1.0	>1.0, ≤1.5	>1.5, ≤2.0	>2.0, ≤3.0	>3.0
0–4	19	6	6	2	0	4	1
5–9	22	5	3	3	3	4	4
10–14	5	1	1	2	0	0	1
15–19	2	1	1	0	0	0	0
20–24	2	0	0	0	0	0	2
25–29	2	0	0	0	0	0	2

[a] Number of rotatable bonds in the ligands
[b] Number of complexes

Table 2 Results for our program based on the refined-scale model

N_{rot}[a]	$N_{complexes}$[b]	$N_{complexes}$ having an RMSD in the listed range (Å)					
		≤0.5	>0.5, ≤1.0	>1.0, ≤1.5	>1.5, ≤2.0	>2.0, ≤3.0	>3.0
0–4	19	9	5	3	2	0	0
5–9	22	6	6	4	4	1	1
10–14	5	2	1	2	0	0	0
15–19	2	1	0	0	1	0	0
20–24	2	0	0	0	0	1	1
25–29	2	0	0	0	1	0	1

[a] Number of rotatable bonds in the ligands
[b] Number of complexes

Table 3 RMSD values of the docking results with the best scores for each method

PDB code	This paper[a]		Glide[b]	GOLD[c]	FlexX[d]	DOCK6[e]
	Coarse-scale	Refined-scale				
1ABE	0.24	0.39	0.17	0.86	1.16	0.15
1ACJ	0.37	0.29	0.28	4	0.49	0.26
1ACM	0.92	0.70	0.29	0.81	1.39	1.40
1AHA	0.29	0.20	0.11	0.51	0.56	0.21
1AZM	0.57	0.25	1.87	2.52	2.37	1.00
1BAF	3.63	1.78	0.76	6.12	8.27	3.79
1CBX	0.26	0.28	0.36	0.54	1.35	1.32
1COY	0.44	0.38	0.28	0.86	1.06	0.28
1CPS	0.32	0.34	3	0.84	0.99	0.43
1DBB	1.17	0.69	0.41	1.17	0.81	0.63
1DBJ	0.52	0.32	0.2	0.72	1.22	1.80
1DID	0.41	0.23	3.82	3.72	4.22	2.78
1DIE	2.2	1.22	0.79	1.03	4.71	2.08
1DR1	0.77	0.75	1.47	1.41	5.64	1.01
1EED	6.24	1.65	5.9	12.43	9.78	7.19
1EPB	1.87	0.67	1.78	2.08	2.77	2.35
1ETA	8.98	2.49	2.92	11.21	8.46	4.42
1ETR	6.34	0.81	1.48	4.23	7.24	5.13
1FKG	0.27	0.44	1.25	1.81	7.59	5.37
1FKI	2.62	1.12	1.92	0.71	0.59	4.13
1GLP	1.35	0.41	0.29	1.35	6.43	0.85
1HEF	4.23	2.25	5.3	1.87	15.32	5.09
1HYT	0.44	0.32	0.28	1.1	1.62	3.99
1ICN	0.86	1.80	2.34	8.63	10.52	6.64
1IDA	0.34	0.28	11.88	12.12	11.95	6.59
1IVE	2.66	1.15	2.61	2.16	5.34	1.79
1LDM	1.42	1.15	0.3	1	0.74	1.79
1LST	0.13	0.21	0.14	0.87	0.71	0.62
1MCR	2.16	1.54	4.33	6.23	10.04	1.95
1MDR	0.58	1.51	0.52	0.36	0.88	1.89
1MRK	0.69	0.52	1.2	1.01	3.55	1.61
1MUP	0.50	0.47	4.37	3.96	3.82	3.14
1PBD	3.52	0.26	0.21	0.57	0.33	0.79
1PHG	2.14	1.96	4.32	1.35	4.74	5.48
1POC	15.37	3.43	5.09	1.27	9.25	4.45
1RNE	13.20	6.68	10.08	2	12.24	1.51
1ROB	1.48	1.69	1.85	3.75	7.7	0.88
1SLT	1.44	1.02	0.51	0.78	1.63	4.07
1SRJ	3.40	1.22	0.58	0.42	2.36	2.04
1STP	5.66	3.25	0.59	0.69	0.65	0.32
1TDB	2.12	0.93	1.46	10.48	10.1	1.91
1XID	1.78	0.3	4.3	0.92	2.01	3.52
1XIE	2.32	0.56	3.86	0.69	1.94	3.11
2PHH	0.64	0.64	0.38	0.72	0.43	1.52
2SIM	0.86	1.26	0.92	0.92	1.99	0.99
2YHX	1.50	1.37	3.84	1.19	2.25	4.96
3PTB	2.11	0.74	0.27	0.96	0.55	1.38
4CTS	0.85	0.3	0.19	1.57	1.53	1.49
4FAB	0.92	0.59	4.5	5.69	4.95	1.11
6ABP	0.33	0.69	0.4	1.08	1.12	0.26

Table 3 continued

PDB code	This paper[a]		Glide[b]	GOLD[c]	FlexX[d]	DOCK6[e]
	Coarse-scale	Refined-scale				
6RNT	1.29	1.45	2.22	1.2	4.79	2.86
8GCH	1.51	1.74	0.3	0.86	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 [11]
[c] The results of Jones and co-workers [2]
[d] The results of Kramer and co-workers [5]
[e] The results by running DOCK6 [1, 12] with the default parameter setting

Table 4 Comparisons with Glide, GOLD, FlexX, and DOCK6 for the docking accuracy with varying RMSD range

RMSD (Å)	Number of complexes					
	Coarse-scale	Refined-scale	Glide	GOLD	FlexX	DOCK6
≤ 0.5	13	18	18	2	3	7
$>0.5, \leq 1.0$	11	12	7	19	9	7
$>1.0, \leq 1.5$	7	9	5	11	6	6
$>1.5, \leq 2.0$	3	8	4	4	5	9
$>2.0, \leq 2.5$	6	2	2	2	4	3
$>2.5, \leq 3.0$	2	2	3	1	1	2
$>3.0, \leq 3.5$	1	0	0	0	0	3
≥ 3.5	9	1	13	13	24	15

with the default parameter setting. Since each method adopted different scoring functions, their docking abilities could not be directly compared by the reported RMSD values in Table 3. However, all reported RMSD values of the well-known methods could be used as a baseline for evaluating the performance of our method. Table 4 and Fig. 1 show the comparisons with other programs, providing the numbers of ligands having an RMSD in various ranges. In 47 (90.4%) of 52 cases, our program (refined-scale) returned a pose within 2.0 Å RMSD, while Glide, GOLD, FlexX, and DOCK6 returned 34 (65.4%), 36 (69.2%), 23 (44.2%), and 29 (55.8%) cases below 2.0 Å, respectively. Additionally, Table 5 presents the comparisons according to the flexibility of the ligands being evaluated. The average RMSD value of our program is much better than others, indicating that our program is superior for molecular docking at this level of rotatable bond count.

4.3 Docking speed

Docking speed is a critical issue in the application of a docking method, especially in virtual screening [31]. Unfortunately, detailed timing data were not published in the benchmarking study. And direct comparison of docking speed is somewhat problematic because of differences in hardware and methodology. Here, we can offer the docking time for our method. According to the docking results, docking per molecule

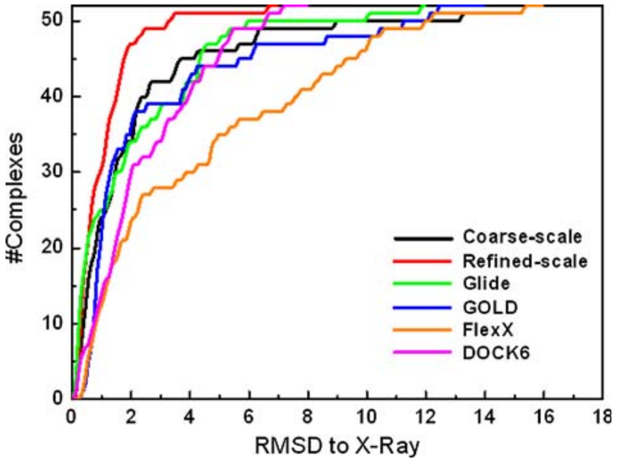

Fig. 1 Comparison of RMSD (heavy atoms) of the solution poses among different programs

Table 5 Comparisons with Glide, GOLD, FlexX, and DOCK6 for RMSD values of ligands with varying numbers of rotatable bonds

Rotatable bonds in the ligands (complexes)	Average RMSD					
	Coarse-scale	Refined-scale	Glide	GOLD	FlexX	DOCK6
0–4(19)	1.15	0.69	1.22	1.62	1.72	1.53
5–9 (22)	1.99	1.07	1.82	2.60	4.29	2.30
10–14 (5)	2.05	0.79	0.89	1.82	4.98	3.28
15–19 (2)	0.60	1.04	7.11	10.38	11.24	6.62
20–24 (2)	8.61	2.44	5.43	5.19	11.45	5.58
25–29 (2)	9.72	6.68	10.08	2.00	12.24	1.51
Total (52)	2.24	1.10	2.19	2.75	4.47	2.58

needs 200–500s for refined-docking method on a SGI Fuel workstation. The average time of docking a ligand for above data set is about 386.68s.

In summary, we have introduced the concept of residue groups and presented a multi-scale optimization mode. Then an improved genetic algorithm for flexible molecular docking has been developed. Based on the model and algorithm, we have developed a new docking program. Compared to other related programs, the docking results for our program are more accurate with respect to the RMSD of the docked ligands. Furthermore, our calculations converge rapidly and steadily. However, certain aspects of this program still require improvement, such as the scoring functions and optimization algorithms. This is the focus of our future research.

Acknowledgements The authors gratefully acknowledge financial support for this work from the National Natural Science Foundation (No. 10772042), the National Basic Research Program of China (No. 2009CB918501), the High Science and Technology Project (No. 2006AA01A124, No. 2007AA02Z304) of China and the Shanghai Committee of Science and Technology (No. 107dz22004).

References

1. I.D. Kuntz, J.M. Blaney, S.J. Oatley, R. Langridge, T.E. Ferrin, J. Mol. Biol. **161**, 269 (1982)
2. G. Jones, P. Willett, R.C. Glen, A.R. Leach, R. Taylor, J. Mol. Biol. **267**, 727 (1997)
3. C.M. Oshiro, I.D. Kuntz, J.S. Dixon, J. Comput. Aided Mol. Des. **9**, 113 (1995)
4. C.J. Sherman, R.C. Ogden, S.T. Freer, J. Med. Chem. **38**, 466 (1995)
5. B. Kramer, M. Rarey, T. Lengauer, Proteins. **37**, 228 (1999)
6. P.N. Palma, L. Krippahl, J.E. Wampler, J.J. Moura, Proteins. **39**, 178 (2000)
7. J. Wang, P.A. Kollman, I.D. Kuntz, Proteins. **36**, 1 (1999)
8. D.R. Westhead, D.E. Clark, C.W. Murray, J. Comput. Aided Mol. Des. **11**, 209 (1997)
9. H.A. Carlson, J.A. McCammon, Mol. Pharmacol. **57**, 213 (2000)
10. S.J. Teague, Nat. Rev. Drug. Discov. **2**, 527 (2003)
11. R.A. Friesner, J.L. Banks, R.B. Murphy, T.A. Halgren, J.J. Klicic, D.T. Mainz, M.P. Repasky, E.H. Knoll, M. Shelley, J.K. Perry, D.E. Shaw, P. Francis, P.S. Shenkin, J. Med. Chem. **47**, 1739 (2004)
12. D.T. Moustakas, P.T. Lang, S. Pegg, E. Pettersen, I.D. Kuntz, N. Brooijmans, R.C. Rizzo, J. Comput. Aided Mol. Des. **20**, 601(2006)
13. S.J. Weiner, P.A. Kollman, D.A. Case, U.C. Singh, C. Ghio, G. Alagona, S. Profeta, P. Weiner, J. Am. Chem. Soc. **106**, 765 (1984)
14. C.A. Lipinski, F. Lombardo, B.W. Dominy, P.J. Feeney, Adv. Drug. Dev. Rev. **23**, 3 (1997)
15. N. Brooijmans, I.D. Kuntz, Annu. Rev. Biophys. Biomol. Struct. **32**, 335 (2003)
16. R.M. Knegtel, I. D. Kuntz, C.M. Oshiro, J. Mol. Biol. **266**, 424 (1997)
17. F. Osterberg, G.M. Morris, M.F. Sanner, A.J. Olson, D.S. Goodsell, Proteins. **46**, 34 (2002)
18. T.M. Frimurer, G.H. Peters, L.F. Iversen, H.S. Andersen, N.P. Moller, O.H. Olsen, Biophys. J. **84**, 2273 (2003)
19. A.Y. Yang, P. Kallblad, R.L. Mancera, J. Comput. Aided Mol. Des. **18**, 235 (2004)
20. C.N. Cavasotto, R.A. Abagyan, J. Mol. Biol. **337**, 209 (2004)
21. W. Sherman, T. Day, M.P. Jacobson, R.A. Friesner, R. Farid, J. Med. Chem. **49**, 534 (2006)
22. J.H. Lin, A.L. Perryman, J.R. Schames, J.A. McCammon, Biopolymers. **68**, 47 (2003)
23. R. Tatsumi, Y. Fukunishi, H. Nakamura, J. Comput. Chem. **25**, 1995 (2004)
24. F. Jiang, S.H. Kim, J. Mol. Biol. **219**, 79 (1991)
25. V.B. Venkayya, Int. J. Numer. Meth. Eng. **13**, 203 (1978)
26. J.H. Holland, *Adaptation in Natural and Artificial Systems* (University of Michigan Press, Ann Arbor, Michigan, 1975)
27. C.E. Shannon, Bell Syst. Tech. J. **27**, 379 (1948)
28. C.E. Shannon, Bell Sys. Tech. J. **27**, 623 (1948)
29. Sybyl (molecular modeling package), Version 6.8 (Tripos Associates: St. Louis, MO, 2000)
30. A.N. Jain, J. Med. Chem. **46**, 499 (2003)
31. M.L. Verdonk, J.C. Cole, M.J. Hartshorn, C.W. Murray, R.D. Taylor, Proteins. **52**, 609 (2003)
32. S.F. Sousa, P.A. Fernandes, M.J. Ramos, Proteins. **65**, 15 (2006)