A hybrid biogeography-based optimization method for the inverse kinematics problem of an 8-DOF redundant humanoid manipulator[*]

Zi-wu REN[†1,2], Zhen-hua WANG[†‡2], Li-ning SUN[2]

(*¹School of Computer Science & Technology, Soochow University, Suzhou 215021, China*)

(*²Robotics and Microsystems Center, Soochow University, Suzhou 215021, China*)

[†]E-mail: zwren@suda.edu.cn; wangzhenhua@suda.edu.cn; wzh@hit.edu.cn

Abstract: The redundant humanoid manipulator has characteristics of multiple degrees of freedom and complex joint structure, and it is not easy to obtain its inverse kinematics solution. The inverse kinematics problem of a humanoid manipulator can be formulated as an equivalent minimization problem, and thus it can be solved using some numerical optimization methods. Biogeography-based optimization (BBO) is a new biogeography inspired optimization algorithm, and it can be adopted to solve the inverse kinematics problem of a humanoid manipulator. The standard BBO algorithm that uses traditional migration and mutation operators suffers from slow convergence and prematurity. A hybrid biogeography-based optimization (HBBO) algorithm, which is based on BBO and differential evolution (DE), is presented. In this hybrid algorithm, new habitats in the ecosystem are produced through a hybrid migration operator, that is, the BBO migration strategy and DE/best/1/bin differential strategy, to alleviate slow convergence at the later evolution stage of the algorithm. In addition, a Gaussian mutation operator is adopted to enhance the exploration ability and improve the diversity of the population. Based on these, an 8-DOF (degree of freedom) redundant humanoid manipulator is employed as an example. The end-effector error (position and orientation) and the 'away limitation level' value of the 8-DOF humanoid manipulator constitute the fitness function of HBBO. The proposed HBBO algorithm has been used to solve the inverse kinematics problem of the 8-DOF redundant humanoid manipulator. Numerical simulation results demonstrate the effectiveness of this method.

Key words: Inverse kinematics problem, 8-DOF humanoid manipulator, Biogeography-based optimization (BBO), Differential evolution (DE)

1 Introduction

The inverse kinematics problem for a redundant humanoid manipulator is described as calculation of the values of joint positions when given the position and orientation of the end effector. It is very important for trajectory planning, motion control, and workspace analysis of the manipulator (Yin *et al.*, 2011).

Currently, the methods for solving the inverse kinematics problem of a robot can be divided into closed-form solutions and numerical methods (Chen P *et al.*, 2012). The closed-form methods have merits of high-accuracy solutions, fast solving speed, and easy identification of all possible solutions, but

they rely heavily on the particular geometric features and lack universality (Zhao *et al.*, 2006; Yin *et al.*, 2011). So, the closed-form methods are available only for manipulators with simplified structures. Numerical methods search for the optimal solution through iterations in the variable space, and finally obtain the numerical solution. The Newton-Raphson method (Kajita, 2005) is relatively simple, but it needs gradient-based information to solve the equivalent minimization problem, and the quality of the numerical solution would depend upon initial parameters. Neural networks (Köker *et al.*, 2004) can approximate the inverse kinematic relations of the robot using their approximation capability. They can map the Cartesian configuration into the corresponding joint angles through training the weights and thresholds of the network. However, a great number of training samples should be given to guarantee the generalization ability of the network. Genetic algorithms (GAs) (Nearchou, 1998) can also be employed to solve the inverse kinematics problem with their global and parallel search characteristics. However, traditional GAs have the shortcomings of premature and convergence stagnation, which degrade the optimization performance and influence the solution accuracy. In addition, there are some other methods which can solve the problem of the redundant manipulator. For example, Tian *et al.* (2011) proposed a method with a database query, but this method would require much prior knowledge and a lot of efforts to constitute the database. Ma *et al.* (2007) presented an inverse kinematics method in which a certain joint variable is fixed, but it would reduce the flexibility of the redundant manipulator. Overall, there exist quite a few methods to settle the inverse kinematics problem of a robot. Generally, with regard to a redundant humanoid manipulator, redundant joints exist for its inverse kinematics problem, and there exist infinite solutions of joint positions when given the position and orientation of the end effector. To solve this problem, a certain optimization criterion should be defined in advance, and then an effective optimization algorithm would be used to select an appropriate inverse solution among all available solutions that meet the demand of the position and orientation of the end effector.

A new population-based evolutionary algorithm for global optimization, i.e., biogeography-based optimization (BBO), was proposed according to the geographical distribution of species and migration features (Simon, 2008). Compared with other evolutionary methods, for example, GA (Goldberg, 1989) and particle swarm optimization (PSO) (Kennedy and Eberhart, 1995), BBO has demonstrated better search performance on various unconstrained benchmark functions (Simon, 2008). BBO does not require any prior knowledge of the objective function gradient, and it is simple in concept, easy to implement, and effective in computation. Currently, it has been applied widely to different areas such as aircraft engine health estimation (Simon, 2008), multiobjective generation dispatching (Chen DJ *et al.*, 2012), and parameter identification of chaos systems (Wang and Xu, 2011).

Although BBO has been applied in many practical optimization problems for its potential advantages and characteristics, it has its own shortcomings. BBO has a good exploitation ability, but it is poor at exploring search space and global optimum. Therefore, BBO may be trapped in local optima when the problem dimension is high or there are numerous local optima. To overcome these flaws and improve the optimization performance, recent work has focused generally on two aspects. One is the improvement on search mechanism of biogeography-based optimization, such as modification of two main operators of BBO (Gong *et al.*, 2010; Ma and Simon, 2011; Yang *et al.*, 2013), or the use of a cosine migration model (Chen DJ *et al.*, 2012); the other is on the framework of the hybrid biogeography-based optimization, that is, combination of BBO with other optimization methods, such as simplex methods (Wang and Xu, 2011) and the predator-prey approach (Costa e Silva *et al.*, 2012), to improve optimization efficiency and searching ability.

In this paper, a novel hybrid biogeography-based optimization (HBBO) algorithm based on differential evolution (DE) (Storn and Price, 1997) with the BBO algorithm is developed to solve the inverse kinematics problem of an 8-DOF redundant humanoid manipulator. It uses a hybrid migration operator that includes two strategies, i.e., the standard BBO migration strategy and the DE/best/1/bin differential strategy, to generate new habitats in the ecosystem, and adopts a Gaussian mutation mechanism to efficiently enhance the convergence property and the quality of solutions. Also, the fitness function of the hybrid method consists of two parts,

i.e., the end-effector error (position and orientation) and the 'away limitation level' value of the 8-DOF humanoid manipulator. Due to combining the merits of two different methods, this hybrid method is more efficient and precise for the inverse kinematics problem.

2 Description of the inverse kinematics problem

Denote the desired position vector and orientation matrix of the end effector as $\boldsymbol{p}_{\text{ref}}$ and $\boldsymbol{R}_{\text{ref}}$ respectively, the actual position vector and orientation matrix of the end effector as $\boldsymbol{p}$ and $\boldsymbol{R}$ respectively, and the variable vector of the manipulator which contains n joints as $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_n)$. Then the errors between the desired and the actual locations of the end effector can be shown by the following equations (Wang and Chen, 1991; Yin *et al.*, 2011):

$$\text{Position error: } \Delta p(\boldsymbol{\theta}) = \|\boldsymbol{p}_{\text{ref}} - \boldsymbol{p}(\boldsymbol{\theta})\|^2, \quad (1)$$

where $\|\cdot\|$ represents the Euclidean distance.

$$\text{Orientation error: } \Delta o(\boldsymbol{\theta}) = \|\Delta\boldsymbol{\omega}\|^2, \quad (2)$$

where $\Delta\boldsymbol{\omega}$ is the vector of angular velocity that can be calculated through the corresponding deviation orientation matrix $\Delta\boldsymbol{R}$.

Denote the element of $\Delta\boldsymbol{R}$ by Δr_{ij} ($i, j = 1, 2, 3$), and then $\Delta\boldsymbol{R}$ can be shown as follows:

$$\Delta\boldsymbol{R} = \begin{pmatrix} \Delta r_{11} & \Delta r_{12} & \Delta r_{13} \\ \Delta r_{21} & \Delta r_{22} & \Delta r_{23} \\ \Delta r_{31} & \Delta r_{32} & \Delta r_{33} \end{pmatrix}. \quad (3)$$

Then $\Delta\boldsymbol{\omega}$ can be computed as

$$\Delta\boldsymbol{\omega} = \begin{cases} (0\ 0\ 0)^{\text{T}}, & \Delta\boldsymbol{R} = \boldsymbol{E}, \\ \dfrac{\theta}{2\sin\theta} \begin{pmatrix} \Delta r_{32} - \Delta r_{23} \\ \Delta r_{13} - \Delta r_{31} \\ \Delta r_{21} - \Delta r_{12} \end{pmatrix}, & \Delta\boldsymbol{R} \neq \boldsymbol{E}, \end{cases} \quad (4)$$

where $\theta = \arccos((\Delta r_{11} + \Delta r_{22} + \Delta r_{33} - 1)/2)$ and $\boldsymbol{E}$ is the unit matrix.

$$\text{Total error: } e(\boldsymbol{\theta}) = \Delta p(\boldsymbol{\theta}) + \Delta o(\boldsymbol{\theta}). \quad (5)$$

Based on the above, the inverse kinematics problem of the manipulator is to find a vector solution $\boldsymbol{\theta}^*$, which meets the requirement $e(\boldsymbol{\theta}^*) \leq \varepsilon$ ($\varepsilon \to 0$). Then the inverse kinematics problem of the robot can be transformed into the following equivalent minimization problem:

$$\begin{aligned} &\text{minimize} \quad e(\boldsymbol{\theta}) = e(\theta_1, \theta_2, \ldots, \theta_n) \\ &\text{subject to} \quad \theta_k^{\text{l}} \leq \theta_k \leq \theta_k^{\text{u}}, \quad k = 1, 2, \ldots, n, \quad (6) \end{aligned}$$

where $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_n)$ is a variable vector in $\mathbb{R}^n$, θ_k ($k = 1, 2, \ldots, n$) is the kth joint variable, and θ_k^{l}, θ_k^{u} denote the lower and upper bounds of θ_k respectively. Generally, when the total error reaches an accuracy level of 10^{-5}, the actual position and orientation of the end effector are deemed to meet the desired requirement.

3 Principle and description of hybrid biogeography-based optimization

3.1 Biogeography-based optimization

BBO (Simon, 2008) is a new population-based heuristic optimization algorithm, which uses the concepts and models from biogeography to describe natural ways of distributing species, i.e., how species migrate, how they arise and become extinct (Costa e Silva *et al.*, 2012). The BBO algorithm achieves information sharing and updating through constituting a migration model, relying on species migration and mutation operators between habitats.

In BBO, each solution is considered as a habitat in an ecosystem, and each solution component is called a suitability index variable (SIV). The solution quality of each habitat is evaluated through its habitat suitability index (HSI), which is analogous to solution fitness in a population-based optimization algorithm. Solutions with higher HSI signify habitats with many species, and vice versa. High-HSI habitats tend to share their features with low-HSI habitats, and poor habitats are likely to accept new features from good ones during the process.

BBO includes two main operators, a migration operator and a mutation operator. The migration operator is probabilistic and modifies individuals by migrating features among habitats. The probability with which the candidate solution $\boldsymbol{H}_i$ is selected as immigrating habitat is proportional to its immigration rate λ_i, and the probability that solution $\boldsymbol{H}_j$ acts as an emigrating habitat is proportional to its emigration rate ϑ_j. Then the migration operator can be described as follows:

$$\Omega(\lambda, \vartheta) : H_i(\text{SIV}) \leftarrow H_j(\text{SIV}), \quad (7)$$

where $\Omega(\lambda, \vartheta)$ is the migration operator, which represents feature adjustment of habitat by migration rates λ and ϑ, $i, j \in \{1, 2, \cdots, m\}$, and m is the population size. With this migration, BBO can share the information among habitats.

In BBO, each habitat has its own immigration rate λ and emigration rate ϑ. A habitat with many species means it has a relatively high ϑ and low λ; on the contrary, a habitat with few species possesses relatively low ϑ and high λ. The rates ϑ and λ of each habitat can be calculated according to the following linear migration model:

$$\lambda_i = I\left(1 - \frac{k(i)}{m}\right), \tag{8}$$

$$\vartheta_i = E\left(\frac{k(i)}{m}\right), \tag{9}$$

where I and E represent the maximum immigration rate and maximum emigration rate respectively, and $k(i)$ is the species number of habitat i (ordered according to the fitness: 1 is the worst while m is the best).

Mutation in BBO is a probabilistic operator that alters a habitat's SIV randomly with a prior probability. Similar to the mutation operator of GAs, it benefits the diversity of the population. Suppose the probability of a habitat is P_i. Then the mutation probability π_i is inversely proportional to the habitat probability, which is obtained as follows:

$$\pi_i = \pi_{\max}\left(1 - \frac{P_i}{P_{\max}}\right), \tag{10}$$

where $\pi_{\max}$ is the predetermined maximum mutation probability, $P_{\max} = \arg\max P_i \ (i = 1, 2, ..., m)$. The calculation of P_i can be referred to Simon (2008).

3.2 Differential evolution

DE is a population-based intelligent search approach, which solves the optimization problem through individuals' cooperation and competition. In each iteration, DE implements differential mutation and crossover operators on the current population to produce a temporary population, and then employs a greedy selection procedure to make one-to-one choice between two populations.

For an n-dimensional optimization problem, suppose the population size is m. The DE/best/1/ bin mutation operator is performed on the current individual $\boldsymbol{x}_i^t$ according to the following equation to produce a mutant vector $\boldsymbol{v}_i^t$ first:

$$\boldsymbol{v}_i^t = \boldsymbol{x}_{\mathrm{gbest}}^t + F\left(\boldsymbol{x}_{r_1}^t - \boldsymbol{x}_{r_2}^t\right), \tag{11}$$

where $r_1, r_2 \in \{1, 2, \cdots, m\}$ are randomly chosen indices at the tth iteration and $r_1 \neq r_2 \neq i$, $\boldsymbol{x}_{\mathrm{gbest}}^t$ is the best individual of the current population, and $F \in [0, 2]$ is called a scaling factor that is used to control the amount of perturbation in the process. Based on the mutant vector, a trial vector $\boldsymbol{u}_i^t$ is constructed through a crossover operation which combines components from the population vector $\boldsymbol{x}_i^t$ and its corresponding mutant vector $\boldsymbol{v}_i^t$ according to

$$u_{ij}^t = \begin{cases} v_{ij}^t, & \mathrm{rand}(\cdot) \leq \mathrm{CR} | j = \mathrm{randn}, \\ x_{ij}^t, & \text{otherwise}, \end{cases} \tag{12}$$

where the subscript j represents the jth element in the corresponding individual, $\mathrm{rand}(\cdot)$ is a uniform number in range [0, 1], CR is the crossover probability, and randn is a randomly chosen integer within the set $\{1, 2, \ldots, m\}$. Finally, the fitnesses of $\boldsymbol{x}_i^t$ and $\boldsymbol{u}_i^t$ are compared, and the better one is chosen to generate offspring through greedy selection:

$$\boldsymbol{x}_i^{t+1} = \begin{cases} \boldsymbol{u}_i^t, & f\left(\boldsymbol{u}_i^t\right) \text{ superior to } f\left(\boldsymbol{x}_i^t\right), \\ \boldsymbol{x}_i^t, & \text{otherwise}. \end{cases} \tag{13}$$

More details on the DE algorithm can be referred to Storn and Price (1997).

3.3 Hybrid biogeography-based optimization

Although BBO has distinctive merits that some other evolutionary algorithms do not possess, it still has several disadvantages. In BBO, the update of habitats relies mainly on the migration operator that is likely to create similar habitats, and the random mutation operator is not efficient for introducing new components to form a better habitat. Therefore, the standard BBO algorithm suffers from a lack of diversity, which limits the optimization performance, especially for complex optimization problems.

An efficient method for increasing population diversity is hybridization of two different evolutionary algorithms. By analyzing the search mechanisms of BBO and DE, we find that BBO generates a new habitat based on the migration of species over time and space, while the DE algorithm adopts three operators, i.e., the differential operator, crossover, and greedy selection, to produce a new individual. These

two different search mechanisms can be incorporated with each other to generate new habitats, which forms a hybrid method named HBBO. The hybrid migration operator in HBBO is illustrated in Algorithm 1, where 'gbest' is the best habitat index of the current population.

Algorithm 1 Hybrid migration operator in HBBO

1: **for** each habitat $i \in \{1, 2, \ldots, m\}$ **do**
2:     Randomly choose index $r_1 \neq r_2 \neq i$
3:     Normalize the immigration rate λ
4:     **for** each SIV $k \in \{1, 2, \ldots, n\}$ **do**
5:        Select habitat $\boldsymbol{H}_i$ with probability $\propto \lambda_i$
6:        **if** $\boldsymbol{H}_i$ is selected **then**
7:           Select habitat $\boldsymbol{H}_j$ with probability $\propto \vartheta_j$
8:           $H_i(k) \leftarrow H_j(k)$
9:        **else**
10:           **if** $\mathrm{rand}(\cdot) \leq \mathrm{CR}|k = \mathrm{randn}$ **then**
11:              $H_i(k) \leftarrow H_{\mathrm{gbest}}(k) + F(H_{r_1}(k) - H_{r_2}(k))$
12:           **else**
13:              $H_i(k) \leftarrow H_i(k)$
14:           **end if**
15:        **end if**
16:     **end for**
17: **end for**

From this operation it can be seen that the new habitat feature in HBBO is created through either the BBO migration strategy or the DE differential & crossover strategy, which can improve the population diversity, and greatly help to enhance exploration capability. Moreover, this hybrid migration operator indicates an attractive modification from other viewpoints. The good habitats can adopt the DE/best/1/bin differential operator to mimic features of $\boldsymbol{H}_{\mathrm{gbest}}$, and they are less likely to be degraded, while poor habitats can still accept a lot of new features from good habitats due to the BBO migration operator. In this sense, the current population can be exploited. Based on this analysis, we can see that the hybrid migration operator can effectively balance global exploration and local exploitation.

The mutation operator is another main operator in BBO, which is able to improve the quality of solutions. It is frequently used in evolutionary algorithms, particularly in the evolutionary programming (EP) algorithm. All of the mutation schemes that have been used for EP can also be used for BBO. Inspired by this, a Gaussian mutation which is often employed in the EP algorithm would be introduced

to the HBBO algorithm. The Gaussian mutation operator has the following merits: First, it can be easily implemented for real-coded variables. Second, it possesses either local search or global search. Especially in the early and middle stages of the evolution, Gaussian mutation can improve the diversity of the population and enhance the exploration capability.

The probability density function of the Gaussian distribution is (Feller, 1971)

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right], \qquad (14)$$

where μ is the mean and σ is the standard deviation. As for a real-valued random variable X distributed normally with mean μ and variance σ^2, it can also be written as

$$X \sim N\left(\mu, \sigma^2\right). \qquad (15)$$

Then the Gaussian mutation operator with mean $\mu = 0$ and variance $\sigma^2 = 1$ can be described by

$$H_i'(k) = H_i(k) + N_k(0, 1), \qquad (16)$$

where $H_i(k)$ is the kth SIV of habitat $\boldsymbol{H}_i$, and $N_k(0, 1)$ indicates that the random number is generated for the kth SIV. Algorithm 2 gives the pseudocode of the Gaussian mutation operator in HBBO.

Algorithm 2 Gaussian mutation operator in HBBO

1: **for** each habitat $i \in \{1, 2, \ldots, m\}$ **do**
2:     Compute the probability P_i
3:     Select SIV $H_i(k)$ with mutation probability m_i
4:     **if** $H_i(k)$ is selected **then**
5:        Update $H_i(k)$ with the Gaussian mutation operator
6:     **end if**
7: **end for**

Based on the above description, the pseudocode of the HBBO algorithm is given by Algorithm 3.

4 Inverse kinematics problem for an 8-DOF redundant humanoid manipulator using HBBO

4.1 Kinematics analysis of an 8-DOF humanoid manipulator

Consider the simulation model and the joint structure model for an 8-DOF redundant humanoid

manipulator shown in Fig. 1. Table 1 gives the range of each joint according to the connecting rod configuration and structure design of the 8-DOF humanoid manipulator.

The unit vectors of the eight joint axis directions $a_0 - a_7$ for the humanoid manipulator are as follows:

$$\begin{cases} a_0 = (0,0,1), & a_1 = (0,1,0), \\ a_2 = (1,0,0), & a_3 = (0,0,1), \\ a_4 = (1,0,0), & a_5 = (0,0,1), \\ a_6 = (1,0,0), & a_7 = (0,1,0). \end{cases} \quad (17)$$

Algorithm 3 HBBO algorithm

1: Generate a random set of habitats $H_1, H_2, ..., H_m$
2: Evaluate the HSI value for each habitat
3: Initialize the generation counter $t = 1$
4: **while** the halting criterion is not satisfied **do**
5:     Sort the population from best to worst
6:     Compute the migration rates λ and ϑ for each habitat based on HSI
7:     Implement the hybrid migration operator shown in Algorithm 1
8:     Update the probability for each habitat
9:     Perform the Gaussian mutation operator shown in Algorithm 2
10:     Recompute the HSI value for the population
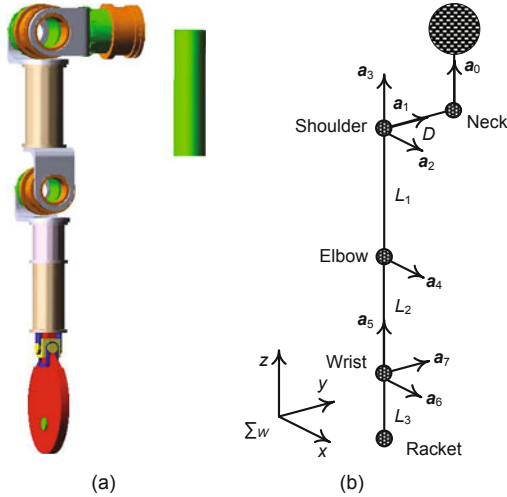11: $t = t + 1$
12: **end while**

Fig. 1 Simulation model (a) and joint structure model (b) for an 8-DOF humanoid manipulator. a_0 is the rotation direction of the waist joint, and $a_1 - a_7$ are the rotation directions of the shoulder, elbow, and wrist joint, respectively. $\sum_W$ is the world-coordinate system, D the shoulder breadth, L_1 the length from the shoulder center to the elbow center, L_2 the length from the elbow center to the wrist center, and L_3 the length from the wrist center to the racket center

Define the joint variables of the 8-DOF redundant humanoid manipulator as an 8×1 vector, i.e., $\boldsymbol{\theta} = (\theta_0, \theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7)^T$. Then the position and orientation (p_j, R_j) of each connecting rod are

$$\begin{cases} p_j = p_i + R_i b_j, \\ R_j = R_i R_{a_j}(q_j), \end{cases} \quad (18)$$

where p_i and R_i are the absolute position and orientation of the mother connecting rod in the world-coordinate system respectively, a_j and b_j are the unit vectors of axis directions and the origin coordinate in the coordinate system of the mother connecting rod respectively, and $R_{a_j}(q_j)$ is the rotation matrix when the axis vector a_j turns around q_j rad. The rotation matrix can be calculated from the following Rodrigues formula (Kajita, 2005):

$$R_{a_j}(q_j) = E + \hat{a}_j \sin q_j + \hat{a}_j^2(1 - \cos q_j), \quad (19)$$

where

$$\hat{a} = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix}^\wedge = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix}.$$

For this 8-DOF redundant humanoid manipulator, if the waist joint variable θ_0 is equal to zero, then the body orientation (or neck orientation) R_0 would be equal to E. Otherwise, the body orientation is $R_0 = R_z(q_0)$. Suppose the position and orientation of the body (or the neck) are (p_0, R_0), and the position and orientation of each connecting rod in the world-coordinate system are described as (p_i, R_i) $(i = 1, 2, \cdots, 8)$. Then the connecting rod orientation R_i $(i = 1, 2, \cdots, 8)$ for the 8-DOF humanoid manipulator can be determined by

$$\begin{cases} R_1 = R_0 \cdot R_y(q_1), & R_2 = R_1 \cdot R_x(q_2), \\ R_3 = R_2 \cdot R_z(q_3), & R_4 = R_3 \cdot R_x(q_4), \\ R_5 = R_4 \cdot R_z(q_5), & R_6 = R_5 \cdot R_x(q_6), \\ R_7 = R_6 \cdot R_y(q_7), & R_8 = R_7, \end{cases} \quad (20)$$

Table 1 Joint ranges of the 8-DOF humanoid manipulator

Joint	LB (°)	UB (°)	Joint	LB (°)	UB (°)
θ_0	−30	30	θ_4	−20	120
θ_1	−126	90	θ_5	−180	180
θ_2	−133	15	θ_6	−80	80
θ_3	−180	90	θ_7	−42	85

LB: lower bound; UB: upper bound

where $\boldsymbol{R}_8$ is the ball racket orientation. The connecting rod position $\boldsymbol{p}_i$ $(i = 1, 2, \ldots, 8)$ is given by

$$
\begin{cases}
\boldsymbol{p}_1 = \boldsymbol{p}_0 + \boldsymbol{R}_0 \cdot \boldsymbol{b}_1, & \boldsymbol{b}_1 = (0 \ -D \ 0)^{\mathrm{T}}, \\
\boldsymbol{p}_2 = \boldsymbol{p}_1 + \boldsymbol{R}_1 \cdot \boldsymbol{b}_2, & \boldsymbol{b}_2 = (0 \ 0 \ 0)^{\mathrm{T}}, \\
\boldsymbol{p}_3 = \boldsymbol{p}_2 + \boldsymbol{R}_2 \cdot \boldsymbol{b}_3, & \boldsymbol{b}_3 = (0 \ 0 \ 0)^{\mathrm{T}}, \\
\boldsymbol{p}_4 = \boldsymbol{p}_3 + \boldsymbol{R}_3 \cdot \boldsymbol{b}_4, & \boldsymbol{b}_4 = (0 \ 0 \ -L_1)^{\mathrm{T}}, \\
\boldsymbol{p}_5 = \boldsymbol{p}_4 + \boldsymbol{R}_4 \cdot \boldsymbol{b}_5, & \boldsymbol{b}_5 = (0 \ 0 \ -L_2)^{\mathrm{T}}, \\
\boldsymbol{p}_6 = \boldsymbol{p}_5 + \boldsymbol{R}_5 \cdot \boldsymbol{b}_6, & \boldsymbol{b}_6 = (0 \ 0 \ 0)^{\mathrm{T}}, \\
\boldsymbol{p}_7 = \boldsymbol{p}_6 + \boldsymbol{R}_6 \cdot \boldsymbol{b}_7, & \boldsymbol{b}_7 = (0 \ 0 \ 0)^{\mathrm{T}}, \\
\boldsymbol{p}_8 = \boldsymbol{p}_7 + \boldsymbol{R}_7 \cdot \boldsymbol{b}_8, & \boldsymbol{b}_8 = (0 \ 0 \ -L_3)^{\mathrm{T}},
\end{cases}
\tag{21}
$$

where $\boldsymbol{p}_8$ is the position of the racket center.

4.2 Inverse kinematics solution for the 8-DOF humanoid manipulator based on HBBO

As shown in Section 2, the inverse kinematics problem of the 8-DOF redundant humanoid manipulator can be turned into an equivalent minimization problem. Thus, it can be solved using a numerical optimization method. An optimal or near-optimal solution that makes the total error $e(\boldsymbol{\theta})$ close to zero can be found by the HBBO algorithm from the set of all possible solutions.

For the 8-DOF redundant humanoid manipulator shown in Fig. 1, there are two redundant joints. Hence, given a certain position and orientation of the end effector, there exist infinite inverse kinematics solutions that can meet the position and orientation requirement. According to the joint variable range (Table 1), each joint has its physical constraints because of structure and configuration design of the manipulator connecting rod. To determine an inverse kinematics solution which is maximumly away from the joint position limit, the following 'away limitation level' criterion, meaning the extent of the solution away from the joint position limitation for the redundant manipulator, is introduced:

$$
\rho = \max \left[\left| \frac{\theta_0 - \theta_0^{\mathrm{mid}}}{(\theta_0^{\mathrm{max}} - \theta_0^{\mathrm{min}})/2} \right|, \left| \frac{\theta_1 - \theta_1^{\mathrm{mid}}}{(\theta_1^{\mathrm{max}} - \theta_1^{\mathrm{min}})/2} \right|, \right.
$$
$$
\left. \ldots, \left| \frac{\theta_7 - \theta_7^{\mathrm{mid}}}{(\theta_7^{\mathrm{max}} - \theta_7^{\mathrm{min}})/2} \right| \right],
\tag{22}
$$

where θ_i^{min}, θ_i^{mid}, and θ_i^{max} $(i = 0, 1, \ldots, 7)$ denote the lower bound, middle value, and upper bound of the ith joint, respectively. $\rho > 1$ means that the obtained inverse kinematics solution is beyond the joint variable limitation problems.

According to the above, to obtain the inverse kinematics solution, both $e(\boldsymbol{\theta})$ and ρ should be minimized. Then the fitness function of the HBBO algorithm can be constituted from these two objectives, and formulated as

$$
\min f(\boldsymbol{\theta}) = e(\boldsymbol{\theta}) + \alpha \cdot \rho,
\tag{23}
$$

where α is a fixed small positive weight parameter, which can balance the total error with the 'away limitation level'. Obviously, α should be set reasonably. Generally, α can be chosen from 10^{-4} to 10^{-7}. Specific appropriate values should be obtained through parameter effect analysis on the quality of the inverse kinematics solution.

5 Simulation

Recall the joint structure model (Fig. 1). Assume the shoulder width $D = 0.14$ m, the length from the shoulder center to the elbow center $L_1 = 0.26$ m, the length from the elbow center to the wrist center $L_2 = 0.25$ m, and the length from the wrist center to the racket center $L_3 = 0.14$ m. Suppose the shoulder center is the origin of the world-coordinate system $\sum_W$, the position and orientation of the neck (or the body) are $\boldsymbol{p}_0 = (0.00 \ 0.14 \ 0.00)^{\mathrm{T}}$ m and $\boldsymbol{R}_0 = \boldsymbol{E}$ (the waist joint variable $\theta_0 = 0$), respectively. For the humanoid manipulator, if the desired position and orientation of the racket center are respectively given by

$$
\boldsymbol{p}_8 = (0.25 \ 0.35 \ -0.35)^{\mathrm{T}} \text{ m}, \tag{24}
$$
$$
\boldsymbol{R}_8 = \boldsymbol{R}_z(0)\boldsymbol{R}_y(-\pi/2)\boldsymbol{R}_x(\pi/2), \tag{25}
$$

then the position and orientation of the end effector can be determined by

$$
\boldsymbol{R}_7 = \boldsymbol{R}_8, \tag{26}
$$
$$
\boldsymbol{p}_7 = \boldsymbol{p}_8 - \boldsymbol{R}_7 \cdot (0 \ 0 \ -L_3)^{\mathrm{T}}
$$
$$
= (0.25 \ 0.21 \ -0.35)^{\mathrm{T}} \text{ m}. \tag{27}
$$

To determine an appropriate value of α, parameter effect analysis on the quality of the inverse kinematics solution is executed for the HBBO algorithm. Suppose the fitness (Eq. (23)) reaches an accuracy level of 10^{-5}, then we can consider that the algorithm has tended to converge. As to the parameter setting in HBBO, both the maximum immigration rate I and the maximum emigration rate E are 1, the predetermined maximum mutation probability is $\pi_{\mathrm{max}} = 0.05$, and the scaling factor and crossover probability of the DE strategy are

$F = 0.60$, CR $= 1.00$, respectively. Also, the optimization variable dimension, i.e., the DOF of the manipulator, is $n=8$, and the range limitation of each joint variable θ_i $(i = 0, 1, \ldots, 7)$ is shown in Table 1.

In addition, the population size of HBBO is set as $m=30$, and the maximum evolutionary iteration is set as $T=2500$; that is, HBBO has 75 000 function evaluations during the whole evolution procedure. Table 2 lists the best, average, worst, standard deviation, and convergence rate values over 10 independent runs when α is chosen at different values for the HBBO algorithm.

From Table 2, it can be seen that although the best fitness under $\alpha = 10^{-5}$ is slightly inferior to those results under $\alpha = 10^{-6}$ and $\alpha = 10^{-7}$, its average solution quality is much superior to the others. Moreover, the convergence rate is highest when $\alpha = 10^{-5}$. Thus, as to the HBBO algorithm, 10^{-5} can be considered as an appropriate value for α.

To evaluate the effectiveness of the proposed HBBO algorithm, the standard genetic algorithm (SGA), DE/rand/1/bin differential evolution (DE), and BBO algorithm are also conducted on the aforementioned problem. In these experiments the parameters of each method are set as follows: For SGA, fitness-proportionate selection, arithmetic crossover, and uniform mutation are adopted; the crossover probability is 0.90 and the mutation probability is 0.05. For DE and BBO, each parameter is set to the same value as in HBBO. In addition, the α in Eq. (23) is set to 10^{-5} for all methods, and the population size, the maximum evolutionary iterations, etc., are also set at the same level as those in HBBO. Each algorithm is performed 10 independent runs, and the results are shown in Table 3. It can be seen

that:

1. Compared to SGA and BBO, HBBO exhibits more accurate results and a superior performance regardless of the best, average, worst, and standard deviation values over the 10 independent runs, which indicates that HBBO can find a much better solution, and its solution quality is excellent.

2. Compared to DE, although the best fitness (8.4257×10^{-6}) given by DE is very close to that given by HBBO (8.0733×10^{-6}), HBBO gives smaller average, worst, and standard deviation results than DE; hence, HBBO has a much better optimization performance.

In summary, the results show that HBBO outperforms SGA, DE, and BBO, and is competent for the inverse kinematics problems of this 8-DOF humanoid manipulator. Fig. 2 illustrates the average fitness evolution curves of the 10 independent runs under these different algorithms. It can be seen that the average evolution curve of HBBO is much closer to the optimal solution, i.e., the horizontal axis in the
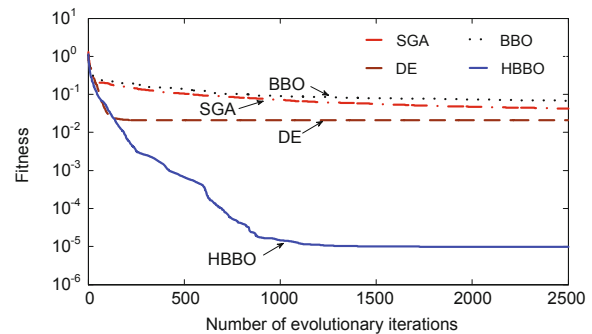

Fig. 2   Average evolution curves under different algorithms on the inverse kinematics problem of the humanoid manipulator

Table 2   HBBO results over 10 independent runs of the inverse problem for different α

α	Best	Average	Worst	Standard deviation	Convergence rate
10^{-4}	9.7847×10^{-5}	2.9094×10^{-2}	2.9004×10^{-2}	9.1686×10^{-2}	20%
10^{-5}	8.0733×10^{-6}	9.8058×10^{-6}	1.0437×10^{-5}	6.4780×10^{-7}	100%
10^{-6}	9.0915×10^{-7}	2.0063×10^{-2}	2.0056×10^{-1}	6.3419×10^{-2}	90%
10^{-7}	9.3703×10^{-8}	1.0826×10^{-2}	9.8405×10^{-2}	3.0845×10^{-2}	70%

Table 3   Results over 10 independent runs of the inverse kinematics problem under four different methods

Method	Best	Average	Worst	Standard deviation
Standard genetic algorithm	3.6036×10^{-2}	4.2291×10^{-2}	4.7253×10^{-2}	4.2084×10^{-3}
Differential evolution	8.4257×10^{-6}	2.0782×10^{-2}	5.7474×10^{-2}	2.8712×10^{-2}
Biogeography-based optimization	1.5693×10^{-2}	6.8393×10^{-2}	1.6791×10^{-1}	6.0563×10^{-2}
Hybrid biogeography-based optimization	8.0733×10^{-6}	9.8058×10^{-6}	1.0437×10^{-5}	6.4780×10^{-7}

figure, which shows the effectiveness of the HBBO algorithm.

To further describe the process, a typical evolution curve of the obtained best fitness (8.0733×10^{-6}) over 10 independent runs with HBBO is given (Fig. 3). In this inverse kinematics solution, the corresponding total error $e(\boldsymbol{\theta})$ is 3.0053×10^{-8}; that is, the accuracy level of the error has reached an order of 10^{-8}, the 'away limitation level' is $\rho = 0.8073$, and the corresponding inverse kinematics solution of the manipulator is given by

$$\boldsymbol{\theta} = [0.2474 \ -0.3854 \ -0.0364 \ -0.5922$$
$$0.8955 \ 1.4594 \ 0.4948 \ -0.5195] \ \text{rad}.$$

By contrast, the total errors of the best fitness over 10 independent runs with BBO and SGA algorithms have reached only an accuracy level of 10^{-2}. The inverse results obtained through BBO and SGA algorithms are not satisfactory.

From the evolution curve in Fig. 3, we can also see that HBBO needs only about 500 iterations to reach a fitness accuracy level of 10^{-6}, and the convergence is very rapid. The HBBO algorithm can obtain the desired inverse kinematics solution. Fig. 4 shows the corresponding position and orientation of
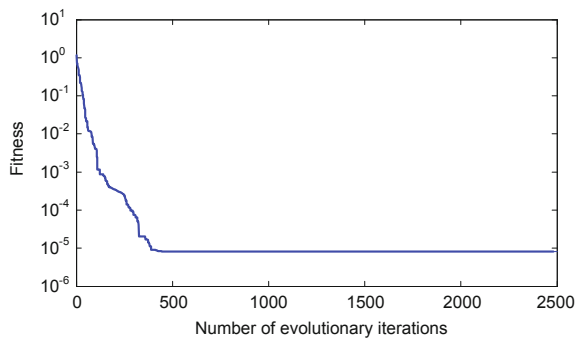

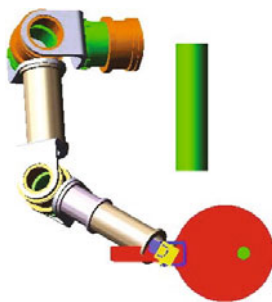**Fig. 3 A typical evolution curve of the fitness under HBBO**

Fig. 4   Position and orientation of the 8-DOF humanoid manipulator configurations

the connecting rod configurations for the 8-DOF humanoid manipulator. It can be seen that the connecting rod configuration of the humanoid manipulator meets the position and orientation requirement of the ball racket as described in Eqs. (24) and (25), which shows the effectiveness of the solution.

The HBBO algorithm has solved the inverse kinematics problem of the 8-DOF humanoid manipulator through an evolution iteration procedure, and it spends a little time to obtain the problem solution. As to the task which does not require high real-time operation, such as cup grasping or refrigerator opening, it could meet the real-time requirement; however, for those high real-time operation tasks, e.g., ping-pong hitting or baseball playing, this method would not satisfy the demand. Even so, the solution obtained using the HBBO algorithm is helpful for research on the inverse kinematics problem for this redundant humanoid manipulator with closed-form solution methods.

6 Conclusions

In this paper, HBBO has been presented to solve the inverse kinematics problem of the redundant humanoid manipulator. It combines BBO with DE to constitute a hybrid method. Moreover, a Gaussian mutation operator was introduced into this hybrid method to weaken the premature problem of the algorithm. Based on this, the HBBO algorithm was applied to solve the inverse kinematics problem. In this method, the end-effector error (position and orientation) and the 'away limitation level' criterion of the 8-DOF humanoid manipulator constitute the fitness function of HBBO. Simulation results demonstrated that this method is better than some other algorithms for the inverse kinematics problem studied.

Despite these promising results, there is still room for improving our work in several aspects. For future work, the solution given by the HBBO algorithm makes it possible to derive the joint analytic expression of the inverse kinematics problem with closed-form solution methods, and thus the work envelope can be demonstrated effectively if the joint variables are computed through the analysis formula when the Cartesian coordinates of the end effector are given. In addition, it is intended to develop new algorithms based on BBO, by hybridizing with other

metaheuristics such as particle swarm optimization and harmony search, and apply these new hybrid algorithms to some other optimization problems in the robotic field, such as robot control and parameter identification of the robot system.

References

Chen, D.J., Gong, Q.W., Qiao, H., *et al.*, 2012. Multi-objective generation dispatching for wind power integrated system adopting improved biogeography-based optimization algorithm. *Proc. CSEE*, **32**(31):150-158 (in Chinese).

Chen, P., Liu, L., Yu, F., *et al.*, 2012. A geometrical method for inverse kinematics of a kind of humanoid manipulator. *Robot*, **34**(2):211-216 (in Chinese). [doi:10.3724/SP.J.1218.2012.00211]

Costa e Silva, M.A., Coelho, L.D.S., Lebensztajn, L., 2012. Multiobjective biogeography-based optimization based on predator-prey approach. *IEEE Trans. Magn.*, **48**(2):951-954. [doi:10.1109/TMAG.2011.2174205]

Feller, W., 1971. An Introduction to Probability Theory and Its Applications. Wiley, New York, USA.

Goldberg, D.E., 1989. Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley Professional, London, UK.

Gong, W.Y., Cai, Z.H., Ling, C.X., *et al.*, 2010. A real-coded biogeography-based optimization with mutation. *Appl. Math. Comput.*, **216**(9):2749-2758. [doi:10.1016/j.amc.2010.03.123]

Kajita, S.J., 2005. Humanoid Robots. Guan, Y., translator, 2007. Tsinghua University Press, Beijing, China (in Chinese).

Kennedy, J., Eberhart, R., 1995. Particle swarm optimization. Proc. IEEE Int. Conf. on Neural Networks, p.1942-1948. [doi:10.1109/ICNN.1995.488968]

Köker, R., Öz, C., Çakar, T., *et al.*, 2004. A study of neural network based inverse kinematics solution for a three-joint robot. *Robot. Auton. Syst.*, **49**(3-4):227-234. [doi:10.1016/j.robot.2004.09.010]

Ma, B.J., Fang, Y.C., Zhang, X.B., 2007. Inverse kinematics analysis for a mobile manipulator with redundant DOFs. Proc. 26th Chinese Control Conf., p.118-122. [doi:10.1109/CHICC.2006.4346874]

Ma, H.P., Simon, D., 2011. Blended biogeography-based optimization for constrained optimization. *Eng. Appl. Artif. Intell.*, **24**(3):517-525. [doi:10.1016/j.engappai.2010.08.005]

Nearchou, A.C., 1998. Solving the inverse kinematics problem of redundant robots operating in complex environments via a modified genetic algorithm. *Mech. Mach. Theory*, **33**(3):273-292. [doi:10.1016/S0094-114X(97)00034-7]

Simon, D., 2008. Biogeography-based optimization. *IEEE Trans. Evol. Comput.*, **12**(6):702-713. [doi:10.1109/TEVC.2008.919004]

Storn, R., Price, K., 1997. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *J. Global Optim.*, **11**(4):341-359. [doi:10.1023/A:1008202821328]

Tian, Y., Chen, X.P., Jia, D.Y., *et al.*, 2011. Design and kinematic analysis of a light weight and high stiffness manipulator for humanoid robots. *Robot*, **33**(3):332-339 (in Chinese). [doi:10.3724/SP.J.1218.2011.00332]

Wang, L., Chen, C.C., 1991. A combined optimization method for solving the inverse kinematics problems of mechanical manipulators. *IEEE Trans. Robot. Autom.*, **7**(4):489-499. [doi:10.1109/70.86079]

Wang, L., Xu, Y., 2011. An effective hybrid biogeography-based optimization algorithm for parameter estimation of chaotic systems. *Expert Syst. Appl.*, **38**(12):15103-15109. [doi:10.1016/j.eswa.2011.05.011]

Yang, G.P., Liu, S.Y., Zhang, J.K., *et al.*, 2013. Control and synchronization of chaotic systems by an improved biogeography-based optimization algorithm. *Appl. Intell.*, **39**(1):132-143. [doi:10.1007/s10489-012-0398-0]

Yin, F., Wang, Y.N., Wei, S.N., 2011. Inverse kinematic solution for robot manipulator based on electromagnetism-like and modified DFP algorithms. *Acta Autom. Sin.*, **37**(1):74-82. [doi:10.3724/SP.J.1004.2011.00074]

Zhao, J., Wang, W.Z., Cai, H.G., 2006. Generation of closed-form inverse kinematics for reconfigurable robots. *Chin. J. Mech. Eng.*, **42**(8):210-214 (in Chinese). [doi:10.3901/JME.2006.08.210]