IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received July 9, 2020, accepted July 24, 2020, date of publication July 28, 2020, date of current version August 7, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3012564

Motion Planning of Redundant Manipulator With Variable Joint Velocity Limit Based on Beetle Antennae Search Algorithm

YAOZONG CHENG[1], CHUNXU LI[2], SHUAI LI[3], (Senior Member, IEEE), AND ZHAN LI[1]
[1]School of Automation Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China
[2]Faculty of Science and Engineering, School of Engineering, University of Plymouth, Plymouth PL4 8AA, U.K.
[3]College of Engineering, Swansea University, Swansea SA2 8PP, U.K.

Corresponding author: Zhan Li (zhan.li@uestc.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61603078, and in part by the Fundamental Funds for the Central Universities.

ABSTRACT Redundant manipulators play important roles in many industrial and service applications by assisting people fulfill heavy and repetitive jobs. However, redundant manipulators are coupled highly-nonlinear systems which exert difficulty of redundancy resolution computation. Conventional methods such as pseudo-inverse-based approaches obtain the resolved joint angles from joint velocity level, which may bring about more computational cost and may neglect joint velocity limits. In this work, a motion planning method based on beetle antennae search algorithm (BAS) is proposed for motion planning of redundant manipulators with the variable joint velocity limit. Such proposed work does not need to resolve the velocity kinematics equation as the conventional methods do, and the proposed method can directly deal with the forward kinematics equation to resolve the desired joint angles. The simulation and experiment on the five-link planar manipulator and the Kuka industrial manipulator system demonstrate the efficiency of the proposed method for motion planning of redundant manipulator, and reveal the reliable performance of the BAS algorithm as compared with genetic algorithm (GA), particle swarm optimization (PSO), firefly algorithm(FA) and quantum behaved particle swarm algorithm(QPSO) methods.

INDEX TERMS Kinematic control, beetle antennae search (BAS), redundant manipulator.

I. INTRODUCTION

Redundant manipulators nowadays have been widely utilized in many industrial and service applications. Different from workers who manually perform industrial operations, redundant manipulators have enormous and obvious advantages by freeing people from heavy and repetitive labor and replacing people to work in dangerous environments with high reliability. As a comprehensive device of mechanical, control, sensing and information processing disciplines, the development level of redundant robots is an important factor to measure the level of industrial automation in manufacturing. Although redundant robot arms have been used in many industrial scenes today, humans are still superior to robotic arms in high-precision work such as the grinding of optical lenses. Therefore, accurate control of redundant manipulators is still essential to investigate.

In order to enable the redundant manipulators to achieve accurate motion control performance, motion planning is needed before imposing servo control of joint angle/position. Taking advantage of redundancy, motion planning can be promisingly achieved with more flexibility. Conventional redundancy resolution methods mainly need to model the forward kinematics equation first and perform inverse kinematics solving by computing the pseudo-inverse of the Jacobian matrix in the level of joint velocity. However, such pseudo-inverse based approach can not generalize the analytical solution to the joint angle level, and other numerical methods might be less computationally efficient. To throw further light on this problem, some researches have conducted efficient alternatives [1]–[16]. Among these works, computational intelligence methods are demonstrated to avoid unnecessary nonlinearity modeling and resolution in the analytic way.

The associate editor coordinating the review of this manuscript and approving it for publication was Shun-Feng Su.

As the redundancy resolution problem for motion planning can be formulated as constrained optimization problems, these intelligent algorithms such as neural network (NN), genetic algorithm (GA) and particle swarm optimization (PSO) were applied to solve the constrained optimization problems [3], [17]. NN has the characteristics of adaptive learning, which can realize parallel distributed processing, with strong robustness and fault tolerance. GAs are often combined with polynomial interpolation in robotic trajectory planning problems, with a small risk of falling into local optimal solutions [18], [19]. However, GA itself needs to go through the process of encoding, decoding, selection, crossover and mutation. The operation time is difficult to guarantee all the time. The PSO algorithm has the characteristics of a simple structure and easy adjustment of parameters. For convex problems, compared to GAs, the cross-variation operation of particle swarms is not needed to perform, and the optimal solution can be found with faster convergence [20]. Still, the PSO needs to randomly generate a lot of particles when generating next populations, which causes the convergence rate of the algorithm decrease.

The beetle antennae search (BAS) algorithm is a new type of computational intelligence algorithms which is applied in scientific and engineering applications, e.g., geomechanics analysis [21], network training [22], controller parameter tuning in servo system [23], unmanned aerial vehicle path planning [24], route planning [25], pattern classification [22], power system [26], ship predictive collision avoidance [27], bridge sensor placement [28], and investment portfolio problems [29]. Because of the excellent nonlinear optimization ability, BAS can be regarded as a potential efficient optimizer in robot applications. For instance, by the algorithm named TPBSO (trajectory-planning beetle swarm optimization), the point-to-point problem of redundant robotic arms on the plane is solved [30]. BAORNN (beetle antennae olfactory recurrent neural network) algorithm focuses on the combination of BAS and recurrent neural network, so as to achieve motion planning of redundant robotic arms in three dimensions. BAORNN also solves problems from the velocity layer [31]. The basic principle of the BAS algorithm is as follows. There is a beetle searching for food. When the beetle is searching, it does not know where the object is, but feeds on the strength of the food. The beetle has two long antennae. If the smell received by the left antennae is larger than the right side, the beetle will fly to the left side for the next step. Otherwise it will fly to the right side. According to such simple principle, beetle can effectively find food. Compared with many NNs, BAS does not require a large number of training samples, and its end position accuracy is guaranteed due to its convergence principle. As comparison with GAs, BAS's search is more directional, which represents faster convergence speed. Compared to PSO-based method, the BAS algorithm does not have to have a large number of particles to get an accurate trajectory.

In order to excavate the computation abilities of the beetles to the greater extent, some researchers have proposed some improved variants of the basic BAS algorithm to improve its search efficiency, e.g., BAS with particle swarm optimization (PSO) [32] and BAS with fallback [33]. Some variants such as beetle swarm antennae search (BSAS) and BAS without parameter tuning (BAS-WPT) have been used, improving performances of the original basic BAS algorithm. The BSAS mainly improves the number of beetles to enhance the performance of a single iterative search, which means releasing more virtual beetles at a time, but only one beetle is moving in the search space. Different from that, the BAS-WPT is a practical modification of BAS and corrects the dependence of BAS on parameter adjustment and adds a method of adding new or fused constraints-penalty functions.

In this work, in order to achieve accurate motion planning of redundant manipulators, a concise BAS algorithm is proposed to solve the redundancy resolution issue. As redundancy resolution and motion planning of redundant robots encounter highly-coupled nonlinearity in forward and inverse kinematics, and BAS has been shown to be an efficient algorithm for handling with such nonlinearity issues appeared in other problems [34], [35]. Motivated by this point, we would like to take advantage of the efficiency of the BAS for motion planning of redundant manipulators. All the joints of the redundant manipulator are considered to be revolute joints, this paper mainly focuses on positioning of the end-effector, and estimation and control of the orientation of the end-effector is not considered in this work. During manipulation in workspace, joints of redundant robots should have dynamic motion limits to ensure safety therefore the joint velocity should fall into variable limits. In this paper, we focus on the proposing a practical BAS algorithm in the kinematic control of redundant manipulator with variable joint velocity limit.

To summarize, the main contributions of this paper are as follows:

1) Unlike the conventional method for redundancy resolution in velocity kinematics level, this work proposes a BAS-based redundancy resolution method for motion planning of the redundant manipulator in the joint angle level by dealing with the forward kinematics equation directly.
2) Different from TPBSO and BAORNN, which are also used in the motion planning of redundant manipulators, this paper uses a basic lightweight BAS and adds the minimum number of iterations to ensure the convergence ability, and successfully applies it to motion planning of 7-DOF redundant robotic arm.
3) Compared with other methods such as GA and PSO, BAS possesses smaller data scales and is relatively concise in programming and operation, making BAS easier to transplant and suitable for platforms with limited computing and storage resources.

Simulation and experimental validation are performed on a five-link planar manipulator and the Kuka iiwa industrial manipulator for verifying the efficiency of proposed BAS method.

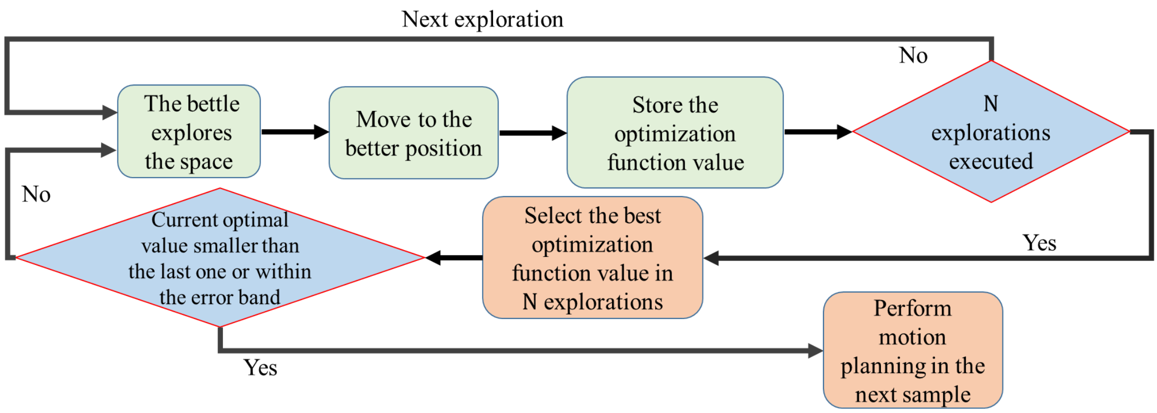


FIGURE 1. The general diagram of BAS for motion planning.

The remainder of this paper is organized as follows. Section 2 introduces the problem formulation on motion planning of redundant manipulators. In Section 3, the detail of the algorithm proposed for motion control of the redundant manipulator in this paper is addressed. In Section 4, the simulation results with discussions are presented, in the meanwhile, the experimental results of verification on Kuka industrial manipulator are presented. Section 5 concludes the paper.

II. PROBLEM FORMULATION

In this section, we will present kinematic modeling of redundant manipulators with variable joint velocity limit and formulating of redundancy resolution problem, and propose the optimization framework.

A. KINEMATICS MODELING OF REDUNDANT MANIPULATOR

The forward kinematics of the redundant manipulator of articular type with n joints can be modeled between the motion of the joint angle and the end-effector, and its relation between Cartesian space and joint space is governed by

$$p(t) = f(\theta(t)) \tag{1}$$

where $p \in \mathbb{R}^m$ denotes the position vector of the end-effector, $\theta(t) \in \mathbb{R}^n$ denotes the joint angle variable vector, and $f(\cdot)$ denotes the nonlinear forward kinematics mapping from joint space to Cartesian space [36]. As mentioned above, all the joints of the redundant manipulator are considered to be revolute joints, this paper mainly focuses on positioning of the end-effector, and estimation and control of the orientation of the end-effector is not considered in this work.

In order to resolve the desired joint angle from the target position path $p_d(t) \in \mathbb{R}^m$ of the end-effector of the redundant manipulator, one has to formulate the following equation to solve

$$p_d(t) = f^{-1}(\theta(t)) \tag{2}$$

Such inverse kinematics solving problem needs to deal with highly-coupled nonlinearity, and it is often difficult to obtain the generalized analytic solution. Conventional redundancy resolution formulation is to be done at the joint velocity level by deriving corresponding velocity kinematics equation as follows.

$$\dot{p} = J\dot{\theta} \tag{3}$$

where $J \in \mathbb{R}^{m\times n}$ denotes the Jacobian matrix which is generated by $J = \partial f/\partial\theta$. Under such circumstance, to resolve the joint angle, one has to obtain the resolved joint angular velocity and make integration. Furthermore, solving pseudo-inverse of J is always needed in this case. In this paper, to get rid of such computation operation, we try to directly solve the inverse kinematics equation to get the desired joint angle variable. To resolve the joint angle, we propose to optimize the following distance index

$$\text{minimize} \quad o(p_d(t), \theta(t)) = \|p_d(t) - f(\theta(t))\|^2 \tag{4}$$

Optimization formulation (4) is a time-varying nonlinear minimization problem as the chained homogeneous matrices make.

B. VARIABLE JOINT VELOCITY LIMIT

In many types of industrial environments, due to the limitation of the workspace for manipulation, redundant manipulators' end-effectors have to be restricted within specific spaces, which makes the joint angles of the redundant manipulators limited with given ranges, i.e., for redundant manipulators, the position of the end-effector is within the following range

$$p_{\min} < p(t) < p_{\max} \tag{5}$$

where $p_{\min}$ and $p_{\max}$ respectively denote the lower and upper limits of the position vector $p(t)$. With redundancy resolution, the joint angles should be within the following range

$$\theta_{\min} < \theta(t) < \theta_{\max} \tag{6}$$

where $\theta_{\min}$ and $\theta_{\max}$ are the limits of joints of the redundant manipulator.

Many factors make the movement modeling of the joints of the redundant manipulators not ideal enough, such as the uncertainty of the redundant manipulator system itself, the unknown external environment, and the friction factor. To pursue the accuracy of trajectory tracking, it is impossible to ignore that the limit of the joint velocity might not be constant. Therefore, it is indispensable to examine that the joint limits may be time-varying or time-dependent during different task segments. In this scenario, the lower and upper bounds of the joint velocity are time-varying, i.e.,

$$\dot{\theta}_{\min} := \dot{\theta}_{\min}(t) \text{ and } \dot{\theta}_{\max} := \dot{\theta}_{\max}(t)$$

which indicates that the joint velocity is time-varying. To model the possible oscillation effect of the joint limits, we propose the following constraints

$$\dot{\theta}_{\min} = -\dot{\theta}_{\max} = \phi(t) = H + W\sin(\omega t) \tag{7}$$

where H and W are scaling parameters for joint variable limit, and ω denotes the oscillation frequency.

III. MOTION PLANNING BASED ON BAS ALGORITHM

The proposed BAS algorithm for motion planning of the redundant manipulator is designed based on the following procedure and configuration. The exploration space for the beetle during searching is configured as the joint space of the redundant manipulator, and left and right antennae of the beetle are the two directions in every searching iteration. The smell that the beetle can detect is configured as the position tracking error of the end-effector of the redundant manipulator. Different from other approaches to generate joint angle trajectories off-line, the joint angle trajectory produced by the proposed BAS algorithm may be feasible to configure for real-time applications due to computational efficiency. The proposed BAS algorithm for motion planning is designed by the following procedures in the ensuing subsections.

Algorithm 1: The Proposed BAS Algorithm for Redundant motion planning With the Variable Joint Limit

Input: The forward kinematics function $f(\theta(t))$ of the redundant manipulator; the desired path $p_d(t)$ of the end-effector; optimal function $o(p_d(t), \theta(t))$; parameters c_1 and c_2; minimal iteration index n

Output: Optimally-resolved joint angle $\theta(t)$ for motion control

```
t ← 0; θ(0) ←Initial joint angle
T ←Total time of motion
T_s ←The time sampling parameter
while t < T/T_s do
    Initially generate δ(t) and λ(t) according to (12) and (13)
    for i <= n do
        i) Generate a randomly-normalized vector b ∈ ℝ^n
        ii) Compute antennaes θ_l(t) ∈ ℝ^n and θ_r(t) ∈ ℝ^n with b according to formulas (8) and (9)
        iii) Make θ_l(t) and θ_r(t) within the exploration limit based on (10)
        iv) Calculate the fitness of the antennaes using (7)
        v) Determine the position of the ith beetle using (11) and standardize it into the allowed space with (10)
        vi) Record the fitness and location of the beetle
        vii) Update the location of the beetle; i = i + 1
    end
    Compare and select the beetle with best fitness
    if the remaining best beetle is within the allowable error band or better than the one of the last iteration then
        t ← t + T_s
        Export the position of remaining beetle at time instant t + T_s
    end
end
```

A. GENERATING SEARCHING ANTENNAE FOR BEETLE IN JOINT SPACE

To start with motion planning for the redundant manipulator, the searching strategy of the beetle should be established first. In every execution time instant t of the redundant manipulator, there is the joint angle put into the search space of the beetle. According to the principle of generating the searching antennae, we propose to use the following joint space direction generating law

$$\theta_l(t) = R(\theta(t) + \lambda(t)b) \tag{8}$$

$$\theta_r(t) = R(\theta(t) - \lambda(t)b) \tag{9}$$

where $\theta_l(t) \in \mathbb{R}^n$ and $\theta_r(t) \in \mathbb{R}^n$ are respectively left and right antennae for searching the optimal joint angle by the beetle, $R(\cdot) : \mathbb{R}^n \to \mathbb{R}^n$ denotes the searching limit function, $\theta(t) \in \mathbb{R}^n$ denotes the current joint angle of the redundant manipulator at time instant t, $\lambda(t) \in \mathbb{R}$ denotes the antenna length/distance of the beetle as the exploration parameter, and $b \in \mathbb{R}^n$ denotes the randomly-generated searching direction vector for the beetle. The entry of the mapping function $R(\cdot) : \mathbb{R}^n \to \mathbb{R}^n$ is $r(\cdot) : \mathbb{R} \to \mathbb{R}$ with the motion limit function as follows

$$r(\theta_i(t)) = \begin{cases} \theta_i(t), & \theta_{\min,i}(t) < \theta_i(t) < \theta_{\max,i}(t) \\ \theta_{\max,i}(t), & \theta_i(t) > \theta_{\max,i}(t) \\ \theta_{\min,i}(t), & \theta_i(t) < \theta_{\min,i}(t) \end{cases} \tag{10}$$

where $\theta_i(t) \in \mathbb{R}$ is the ith element of the joint angle vector $\theta(t) \in \mathbb{R}^n$.

B. STATE UPDATE OF THE BEETLE IN SEARCHING SPACE

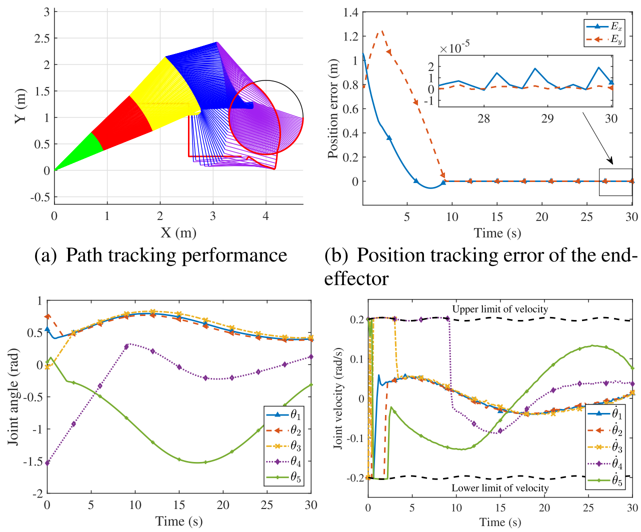
After constructing the beetle for motion planning of the redundant manipulator, according to the searching status of the antennae, the position of the beetle is updated in searching space. The updating rule is depicted by

$$\theta'(t+\Delta t) = R[\theta(t) - \delta(t)\mathrm{sgn}(o(p_d(t), \theta_l(t)) - o(p_d(t), \theta_r(t)))b] \tag{11}$$

where $\theta'(t+\Delta t) \in R^n$ denotes the temporary updated position, $\delta(t)$ denotes the motion step size of the beetle, $\mathrm{sgn}(\cdot)$ denotes the saturation sign function on the input z as follows

$$\mathrm{sgn}(z) = \begin{cases} 1, & z > 0 \\ 0, & z = 0 \\ -1, & z < 0 \end{cases}$$

Next, compute updated coordinates of the beetle in searching space. After the beetle updates its motion state for n steps, find the best optimal function $o(p_d(t), \theta(t))$ and choose the corresponding $\theta'(t+\Delta t)$ which might be configured as the actual resolved joint angle $\theta(t+\Delta t)$. In order to guarantee the converge of tracking errors by the proposed algorithm, it is still needed to evaluate whether the chosen $\theta'(t+\Delta t)$ can guarantee the current error function value can be better than the one in the previous step or the error can be controlled in the error band.

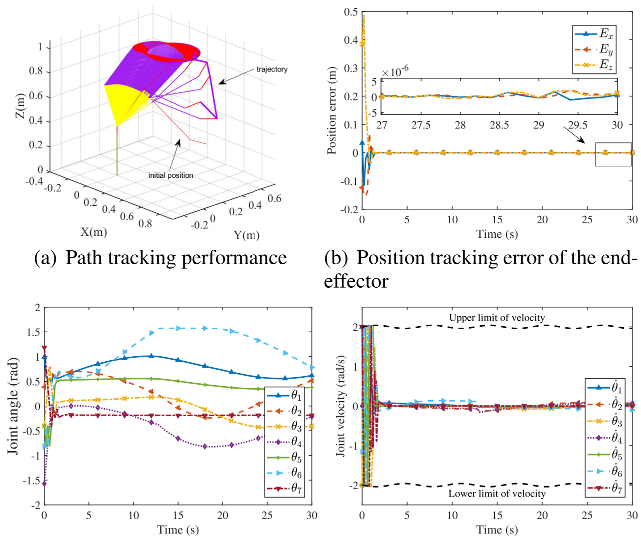


(a) Path tracking performance (b) Position tracking error of the end-effector (c) Resolved joint angle (d) Resolved joint velocity

FIGURE 2. Comprehensive performance of motion planning of the five-link manipulator with the variable joint velocity limit by the proposed method.

C. STEP-SIZE AND ANTENNA EXPLORATION RANGE PARAMETER UPDATING

At the beginning stage of the exploration, the distance between the targeted position and the current position of the end-effector may be larger than the follow-up stages of the exploration. In this case, the corresponding parameters such as the step-size and antenna exploration range parameter might be not optimal, so these parameters are required to update as the updating of $\theta(t)$.



(a) Path tracking performance (b) Position tracking error of the end-effector (c) Resolved joint angle (d) Resolved joint velocity

FIGURE 3. Comprehensive performance of motion planning of the Kuka manipulator with the joint velocity limit by the proposed method.

For the step size parameter $\delta(t)$ and the detection distance parameter $\lambda(t)$, they often need to be appropriately chosen through necessary optimal computation. After absorbing some of the experience of BAS-WPT [37] and BAORNN [31], we define the updating law for $\delta(t)$ and $\lambda(t)$ as follows

$$\lambda(t) = c_1\sqrt{o(p_d(t-1), \theta'(t))} \tag{12}$$

$$\delta(t) = c_2\lambda(t) \tag{13}$$

where c_1 and c_2 are parameters for tuning the convergence speed of the BAS algorithm, and c_1 and c_2 should be configured properly to provide well-conditioned convergence according to the time sampling parameter T_s. The overall steps of the proposed algorithm for redundant motion planning with the variable joint limit are presented in the flow chart shown in Fig. 1 and the algorithm block.

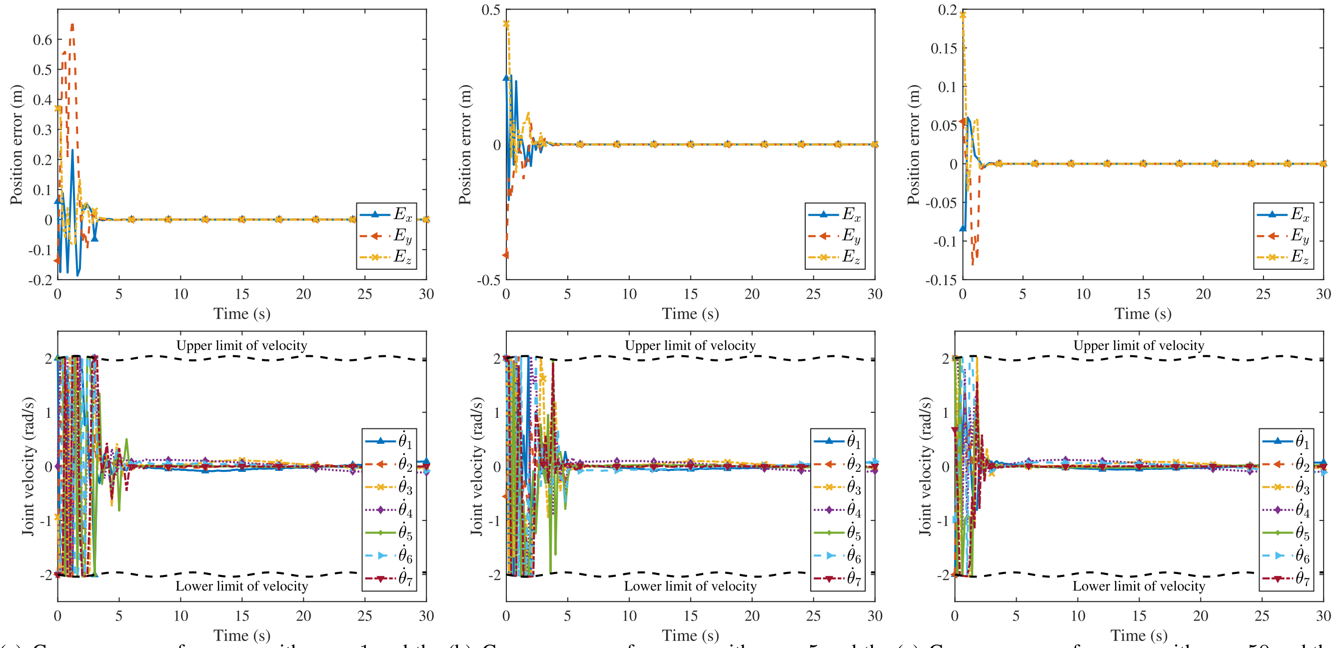
The convergence of the basic BAS algorithm has been proved in Theorem 1 of [38]. In Theorem 1, properly choosing the step size can guarantee that the BAS algorithm is asymptotic convergent with probability 1. For the redundant resolution in this work, one can extend the proof of Theorem 1 in [38] and validate its convergence performance.

IV. SIMULATION AND EXPERIMENT RESULTS

In this section, in order to verify and evaluate the performance of the proposed BAS algorithm for motion planning, the simulation and the experiment on the five-link planar manipulator and Kuka iiwa industrial manipulator with the desired tracking path being a circle are performed. The space and time equation of circle path is

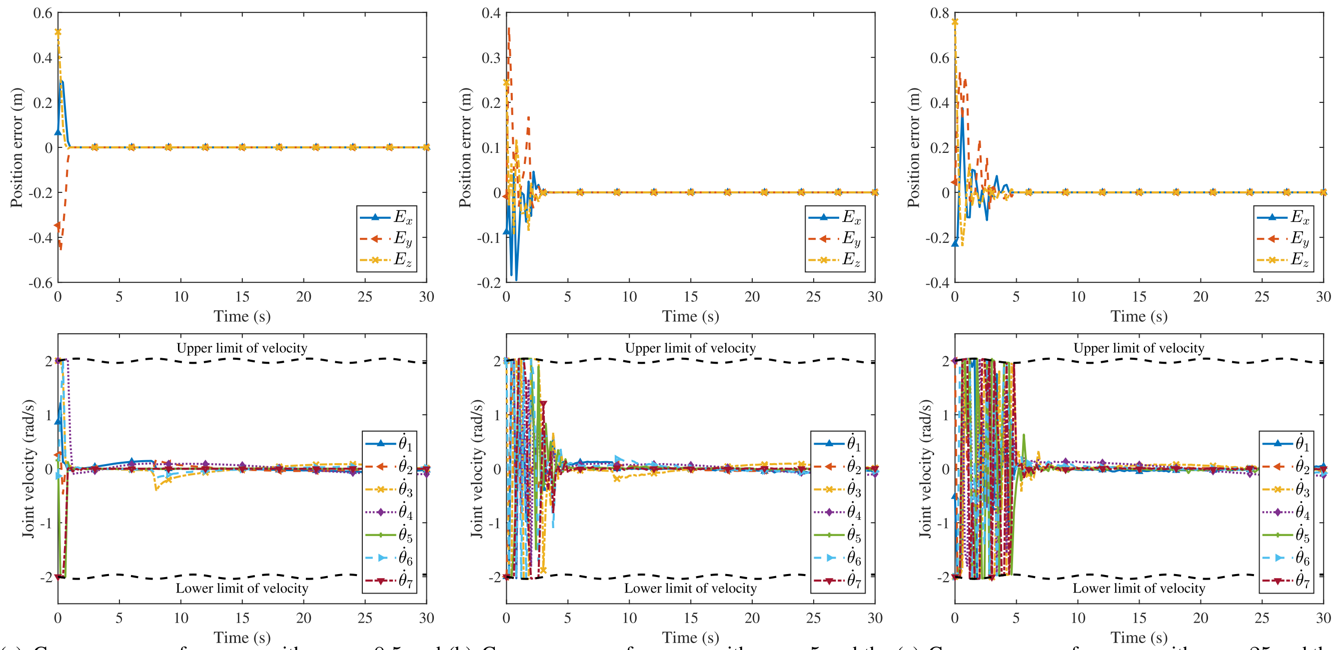
$$p_d(t) = p_0 + \frac{D}{2}\cos(\frac{2\pi t}{T})A + \frac{D}{2}\sin(\frac{2\pi t}{T})B \tag{14}$$

where $p_0 \in \mathbb{R}^3$ denotes the center of the circle path, $D > 0 \in \mathbb{R}$ denote the diameter of the circle, $A \in \mathbb{R}^3$ and $B \in \mathbb{R}^3$



(a) Convergence performance with $n = 1$ and the computational time being 11.6 s. (b) Convergence performance with $n = 5$ and the computational time being 16.4 s. (c) Convergence performance with $n = 50$ and the computational time being 18.9 s.

FIGURE 4. Comparison of convergence with different times *n* of explorations *n* in joint angle resolution loops. Upper: position tracking error of the end-effector; lower: resolved joint velocity.



(a) Convergence performance with $c_1 = 0.5$ and the computational time being is 99.5s. (b) Convergence performance with $c_1 = 5$ and the computational time being 32.2s. (c) Convergence performance with $c_1 = 25$ and the computational time being 5.1s.

FIGURE 5. Performance comparison with different coefficient parameters c_1. Upper: position tracking error of the end-effector; lower: resolved joint velocity.

denote parameter matrices projecting the circle in different directions.

A. SIMULATION RESULTS

The simulation of the proposed BAS algorithm is performed in MATLAB (2017b) platform in a personal computer with the CPU being Intel i7-8500U. The task execution time interval T_s is set as 0.2 s for the redundant manipulator to reach the next sampled point of the trajectory solution, and the total motion duration time is set as 30 s.

1) SIMULATION ON THE FIVE-LINK PLANAR MANIPULATOR

The proposed BAS algorithm is firstly validated on the motion planning of the five-link planar manipulator without the joint velocity limit configured. The parameters for the desired circular path are $p_0 = [4 \ \ 1 \ \ 0]^T$ m, $D = 1.4$ m, $A = [1 \ \ 0 \ \ 0]^T$ and $B = [0 \ \ 1 \ \ 0]^T$.

The proposed method is applied to motion planning of the five-link planar manipulator with the variable joint velocity limit. The joint velocity (speed) limits are set as $\dot{\theta}_{\min} = -0.2 - 0.004\sin(t)$ rad/s and $\dot{\theta}_{\max} = 0.2 + 0.004\sin(t)$ rad/s. Fig. 2 shows the comprehensive performance of motion planning of the five-link planar manipulator with the joint velocity limit based on the proposed BAS method. We can observe that, the end-effector of the five-link planar manipulator can track the desired circle path well with steady-state position errors being around 2×10^{-5} m. This demonstrates that the proposed BAS method can work promisingly on the redundancy resolution of the five-link planar manipulator joint velocity limits considered. Especially, seen from Fig. 2(d), the resolved joint velocity can be always limited to the joint velocity (speed) boundaries. All of these indicate that the proposed BAS method can successfully work for motion planning of the five-link planar manipulator with and without variable joint velocity limits.

2) SIMULATION ON THE KUKA IIWA INDUSTRIAL MANIPULATOR

The proposed BAS algorithm is next validated on the motion planning of the Kuka iiwa industrial manipulator with the variable joint velocity limit. The parameters for the desired circular path are $p_0 = [0.3 \ \ 0.3 \ \ 1.0]^T$ m, $D = 0.2$ m, $A = [1 \ \ 0 \ \ 0]^T$ and $B = [0 \ \ 1 \ \ 0]^T$. The joint velocity (speed) limits are set as $\dot{\theta}_{\min} = -2-0.004\sin(t)$ rad/s and $\dot{\theta}_{\max} = 2+0.004\sin(t)$ rad/s (which is different from the case of the five-link planar manipulator). Fig. 3 shows the comprehensive performance of motion planning of the Kuka manipulator with the joint velocity limit by the proposed BAS method. We could evidently see that, the end-effector of the Kuka iiwa manipulator can track the desired circular path well with the resolved joint angles calculated by the proposed BAS method. Meanwhile, the variable joint velocity limits can be always satisfied. The results demonstrate the proposed method can work well on the Kuka industrial manipulator from aspect of simulation.

3) EVALUATION ON THE PERFORMANCE WITH DIFFERENT PARAMETERS CONFIGURED

This part focuses on the times of explorations required per iteration for the beetle in the algorithm utilized, coefficient parameter of the distance of detection $\lambda(t)$ and coefficient parameter of the step size $\delta(t)$. We analyze the meaning of each parameter to the manipulator system and summarize the general tuning rule for parameter adjustment. It is worth noting that all evaluations and discussions are based on that parameters are within reasonable limits.

a: TIMES OF EXPLORATIONS OF THE BEETLE IN EACH ITERATION n

The exploration times represents the search times of the beetle in each joint angle resolution loop. The more times of explorations need more computation costs, which makes the execution time of the resolution loop longer. However, for the BAS algorithm, the next resolution loop can not be entered until the condition is met, where the condition is that the updated optimization/fitness function is less than the previous one or within the error band. The convergence speed of the BAS algorithm can not be guaranteed when the number n is small. A comparison of the convergence results is shown in Fig. 4 with the same parameter pair c_1 and c_2 set.

It can be clearly seen from Fig. 4 that, when the number of explorations n increases, the position tracking error of the end-effector can faster converge to zero and the resolved joint velocity can reach to the steady state with less oscillations. That is to say the converge can be accelerated but more computational costs are required when n increases.

b: COEFFICIENT PARAMETER c_1 OF THE DISTANCE OF DETECTION $\lambda(t)$

As we mentioned, the coefficient parameter c_1 of the exploration distance represents the level of its exploration ability. Based on (12), the size of c_1 affects the speed of exploration when the other parameters are the same. The simulation results with different c_1 are shown in Fig. 5. Obviously choosing a larger coefficient parameter c_1 will significantly decrease the time of operations/computation. However, at the same time, enlarging c_1 will also make the joint velocity oscillate rapidly for a longer time, making the body of the manipulator tremble.

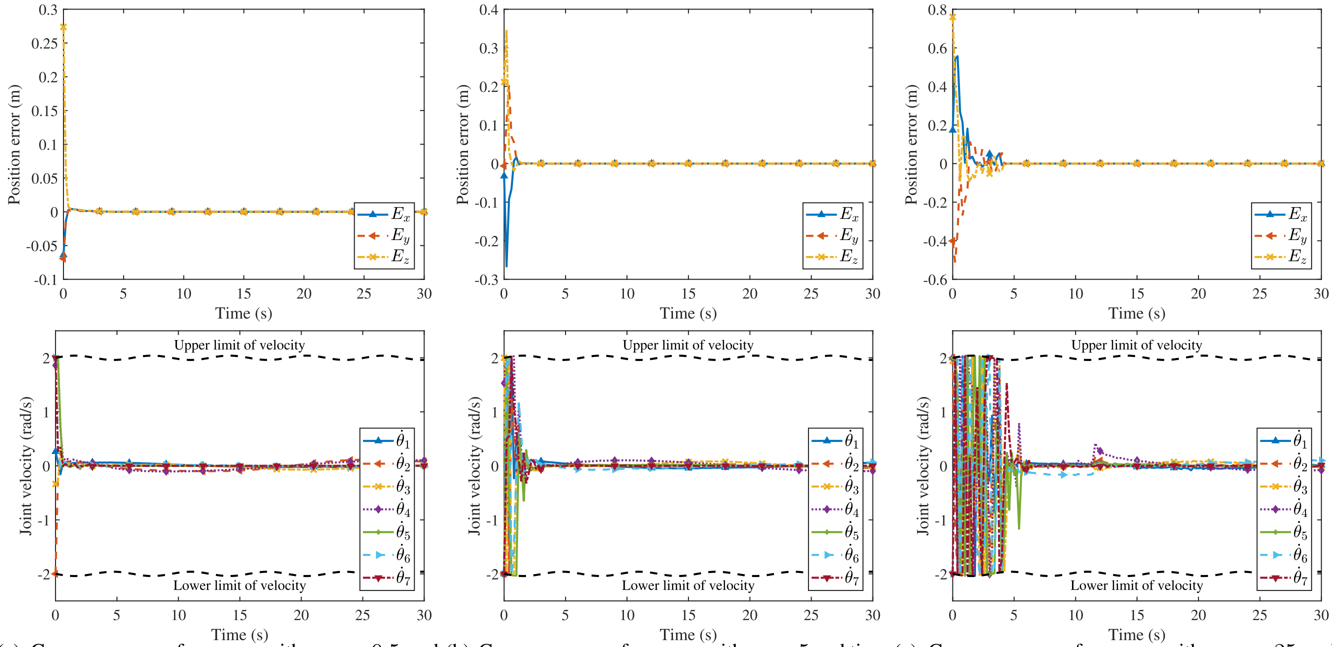
c: COEFFICIENT PARAMETER c_2 OF THE STEP SIZE $\delta(t)$

The coefficient parameter c_2 represents the size of the distance that the beetle moves after learning the direction. The size of c_2 directly determines the range of movement of the beetle, because in a broad sense, if the step size of the beetle is not large enough, the appropriate solution may can not be reached if the number of times the beetle moves is the same.

Similar to c_1, the increase in c_2 also means an increase in the search ability of the beetle which means shorter search time. But at the same time, a stronger search capability also means a larger oscillation range, and longer oscillation time.

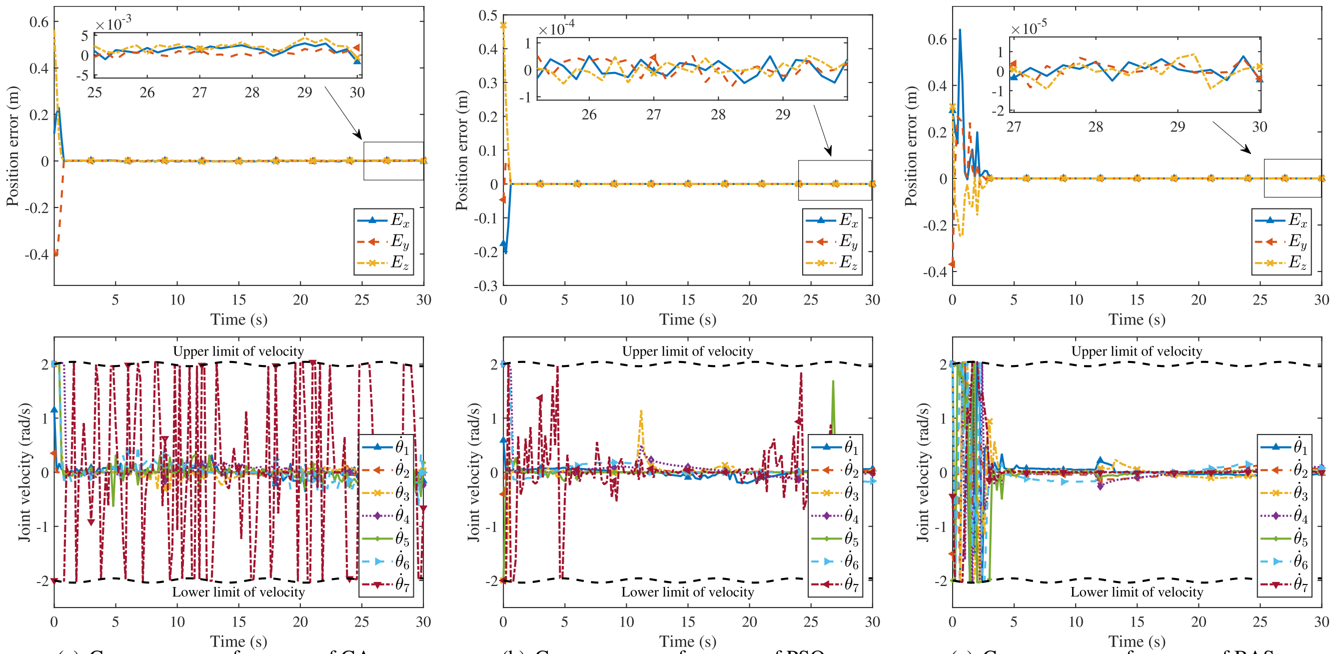
4) COMPARISON WITH OTHER TYPICAL INTELLIGENT ALGORITHMS

To demonstrate the superiority and practicability of the BAS algorithm proposed in this paper, we also comparatively present the performance results of the GA, the PSO, the firefly algorithm (FA) [39] and the quantum behaved particle swarm algorithm (QPSO) [40] for the same motion planning problem. PSO and GA algorithms for comparison are from the Matlab (R2017b) function library.



(a) Convergence performance with $c_2 = 0.5$ and time is 261.1s. (b) Convergence performance with $c_2 = 5$ and time is 34.8s. (c) Convergence performance with $c_2 = 25$ and time is 11.6s.

FIGURE 6. Performance comparison with different coefficient parameters c_2. Upper: position tracking error of the end-effector; lower: resolved joint velocity.



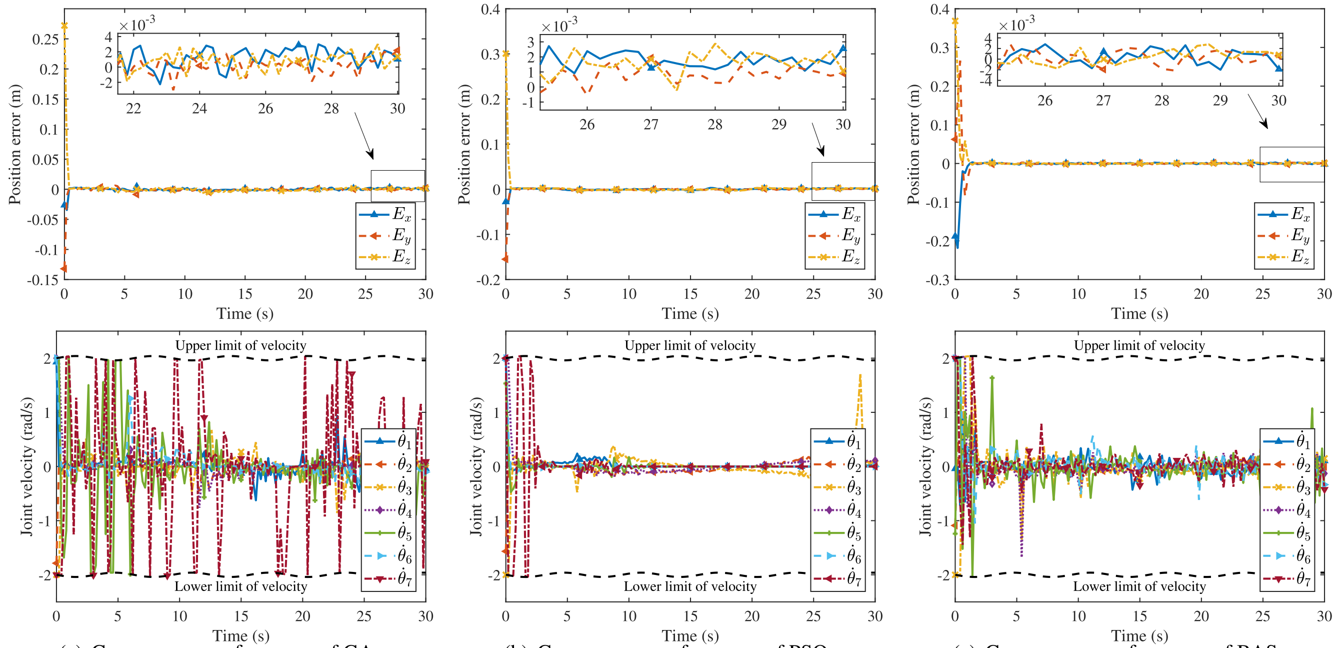
(a) Convergence performance of GA. (b) Convergence performance of PSO. (c) Convergence performance of BAS.

FIGURE 7. Performance comparison of different algorithms with a running time of around 15s. Upper: position tracking error of the end-effector; lower: resolved joint velocity.

We compare the performances of the five methods in the following two cases, i.e., (1) comparison of accuracy under the same level of running time, and (2) comparison of the running time under the same level of precision or error band.

a: COMPARISON OF ACCURACY AT THE SAME LEVEL OF RUNNING TIME

We utilize five algorithms on the same platform to solve motion planning of redundant manipulator with the same variable joint velocity limit. The comparison of the motion



(a) Convergence performance of GA. (b) Convergence performance of PSO. (c) Convergence performance of BAS.

FIGURE 8. Performance comparison of different algorithms with precision unit of mm. Upper: position tracking error of the end-effector; lower: resolved joint velocity.

planning results at the same level of the running time of GA, PSO and BAS is shown in Fig.7. Obviously, within the same level of motion planning duration, the results of the PSO and GA methods produce position errors of in the level of 10^{-4} m and 10^{-3} m, while the position error by the BAS method is at the level of 10^{-5} m. Even if there are performance fluctuations in computational time caused by the different algorithms due to randomness of parameter generator/configuration, we can clearly see that the proposed BAS algorithm in this paper may show superiority in tracking accuracy.

Furthermore, we performed multiple tests and calculated the average errors of the three methods in three space dimensions X, Y and Z in the steady state. The comparison results for GA, PSO, FA, QPSO and BAS methods are shown in Tab.1. It can be seen that the error of the BAS is significantly smaller at a similar time in all the three dimensions.

TABLE 1. Comparison of average position errors of GA, PSO, FA, QPSO and BAS methods.

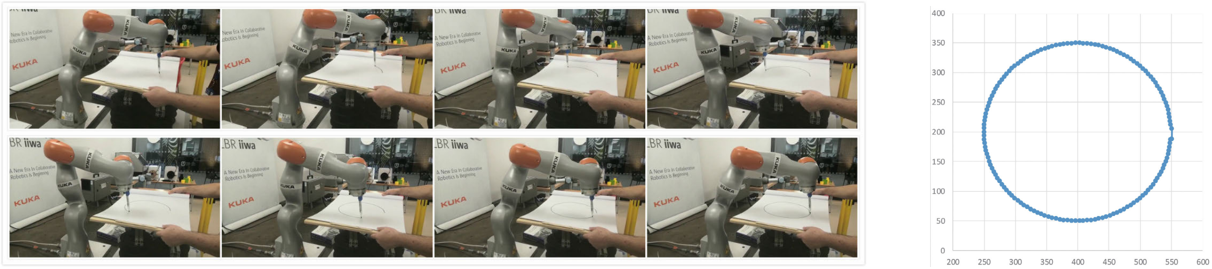
Position error	X direction (m)	Y direction (m)	Z direction (m)
GA	0.78 ± 0.25 $\times 10^{-3}$	0.10 ± 0.29 $\times 10^{-3}$	1.08 ± 0.42 $\times 10^{-3}$
PSO	0.01 ± 0.22 $\times 10^{-4}$	0.04 ± 0.18 $\times 10^{-4}$	0.10 ± 0.18 $\times 10^{-4}$
FA	0.02 ± 0.09 $\times 10^{-4}$	0.00 ± 0.11 $\times 10^{-4}$	0.03 ± 0.12 $\times 10^{-4}$
QPSO	0.02 ± 0.08 $\times 10^{-3}$	0.01 ± 0.07 $\times 10^{-3}$	0.07 ± 0.27 $\times 10^{-3}$
BAS	$\mathbf{0.10 \pm 0.04}$ $\mathbf{\times 10^{-5}}$	$\mathbf{0.04 \pm 0.05}$ $\mathbf{\times 10^{-5}}$	$\mathbf{0.17 \pm 0.05}$ $\mathbf{\times 10^{-5}}$

b: COMPARISON OF RUNNING TIME UNDER THE SAME LEVEL OF ACCURACY

When the error level is of 10^{-3} m, it can be seen from the Fig.8 that both the PSO and GA methods can not decrease oscillations of the joint velocity in a short time even in the steady state, while the BAS method can force the joint velocity much lower than the velocity limits. This demonstrates that the BAS is more reliable in satisfying variable joint limits. More importantly, the running time of the five methods is shown in Tab.2. Clearly seen from this table, the BAS method is much superior to the other four methods. Specifically, the average running time of the BAS method is 0.53 ± 0.04 s, while the average running time of the GA, PSO, FA and QPSO methods are respectively 25.0 ± 10.2 s, 5.0 ± 0.7 s, 6.9 ± 0.6 s and 6.1 ± 0.4 s.

B. EXPERIMENTAL VERIFICATION

After we validate the proposed BAS method for motion planning of the five-link manipulator and Kuka industrial manipulator in simulation, we apply the proposed BAS method on the motion control experiment on the real Kuka iiwa industrial manipulator. The parameters of the proposed BAS method follow the aforementioned configuration discipline. In view of the performance comparison results in the previous simulation, as the BAS method has shown sufficiently superiority than other four algorithm in tracking accuracy and computational efficiency, we thus choose the superior one (i.e., the BAS one) for experimental validation in the Kuka robot system. The resolved joint angles data from the BAS method are sent into the Kuka manipulator's operation



(a) Snapshots of the Kuka industrial manipulator for drawing the circle based on the proposed BAS method for redundancy resolution.

(b) The drawn circle reflected from the Kuka industrial manipulator's operation system.

FIGURE 9. Experimental verification.

TABLE 2. Comparison of convergence time in 25 tests.

Algorithm	GA	PSO	FA	QPSO	BAS
Test 1	18.7 s	4.5 s	7.5 s	6.0 s	**0.52** s
Test 2	17.4 s	4.8 s	7.8 s	5.9 s	**0.48** s
Test 3	27.0 s	4.1 s	7.5 s	6.5 s	**0.54** s
Test 4	36.9 s	6.1 s	9.1 s	5.5 s	**0.51** s
Test 5	17.8 s	4.5 s	7.1 s	5.7 s	**0.60** s
Test 6	24.6 s	3.9 s	6.8 s	6.6 s	**0.57** s
Test 7	35.6 s	4.1 s	6.3 s	5.7 s	**0.67** s
Test 8	20.0 s	4.2 s	7.1 s	5.6 s	**0.51** s
Test 9	10.4 s	5.1 s	6.9 s	6.4 s	**0.53** s
Test 10	18.0 s	4.5 s	6.9 s	5.6 s	**0.56** s
Test 11	22.7 s	5.2 s	6.5 s	6.2 s	**0.52** s
Test 12	21.8 s	4.3 s	6.5 s	5.9 s	**0.55** s
Test 13	31.6 s	6.0 s	6.8 s	6.4 s	**0.50** s
Test 14	40.5 s	5.0 s	6.8 s	6.3 s	**0.54** s
Test 15	14.5 s	5.3 s	6.4 s	5.8 s	**0.55** s
Test 16	27.6 s	6.5 s	7.2 s	6.6 s	**0.47** s
Test 17	42.6 s	4.5 s	6.9 s	6.2 s	**0.56** s
Test 18	37.7 s	5.2 s	6.9 s	6.5 s	**0.51** s
Test 19	11.0 s	5.8 s	6.6 s	6.4 s	**0.49** s
Test 20	38.7 s	4.8 s	6.7 s	6.3 s	**0.48** s
Test 21	26.4 s	7.0 s	6.7 s	6.4 s	**0.53** s
Test 22	19.5 s	5.9 s	6.9 s	6.4 s	**0.55** s
Test 23	40.0 s	5.3 s	6.8 s	6.5 s	**0.53** s
Test 24	10.6 s	5.0 s	6.8 s	5.7 s	**0.61** s
Test 25	15.8 s	4.9 s	7.0 s	6.7 s	**0.53** s
Average	25.0 ±10.2 s	5.0 ±0.7 s	6.9 ±0.6 s	6.1 ±0.4 s	**0.53 ±0.04** s

system to perform path tracking control. The snapshots of the experiment session on the Kuka manipulator are shown in Fig. 9(a), and the tracked path reflected from the Kuka manipulator's operation system is shown by Fig. 9(b), which is a circle as expected. For dynamic illustration, a supplement video is provided as an attachment with the paper. From the experimental results, we can see that the proposed method can work promisingly in the motion planning task on the Kuka iiwa industrial manipulator system.

V. CONCLUSION

In this work, a novel motion planning method based on BAS algorithm has been proposed for motion planning of redundant manipulators with the variable joint velocity limit considered. The proposed work does not need to resolve the velocity kinematics equation as the conventional methods do by dealing with Jacobian matrices, and it can directly resolve in the level of forward kinematics equation to obtain the desired joint angles. The simulation and experiment on the five-link manipulator and the Kuka industrial manipulator system have demonstrated the efficiency of the proposed method for motion planning of redundant manipulator, and reveal the reliable performance of the proposed algorithm. As compared with other four computational intelligence methods, i.e., the GA, PSO, FA and QPSO algorithms, the proposed BAS method has stronger solving ability and occupies less computational resources, which makes this method easy to be transplanted and has the possibility to be deployed to the manipulator control system.

REFERENCES

[1] V.-T. Nguyen, C.-Y. Lin, S.-F. Su, W. Sun, and M. J. Er, "Global finite time active disturbance rejection control for parallel manipulators with unknown bounded uncertainties," *IEEE Trans. Syst., Man, Cybern. Syst.*, early access, Apr. 30, 2020, doi: 10.1109/TSMC.2020.2987056.

[2] D. Chen, S. Li, Q. Wu, and X. Luo, "New disturbance rejection constraint for redundant robot manipulators: An optimization perspective," *IEEE Trans. Ind. Informat.*, vol. 16, no. 4, pp. 2221–2232, Apr. 2020.

[3] W. Gong, D. Chen, and S. Li, "Active sensing of robot arms based on zeroing neural networks: A biological-heuristic optimization model," *IEEE Access*, vol. 8, pp. 25976–25989, 2020.

[4] V.-T. Nguyen, S.-F. Su, A.-T. Nguyen, and V.-T. Nguyen, "Adaptive non-singular fast terminal sliding mode tracking control for parallel manipulators with uncertainties," in *Proc. Int. Conf. Syst. Sci. Eng. (ICSSE)*, Jul. 2019, pp. 522–525.

[5] M.-C. Le, S.-F. Su, V.-T. Nguyen, L.-W. Chen, and V.-Y. Nguyen, "Adaptive neural network controller-based chattering-free sliding mode for 6-Dof industrial manipulators," in *Proc. Int. Conf. Syst. Sci. Eng. (ICSSE)*, Jul. 2019, pp. 75–80.

[6] Z.-Y. Sun, C.-Y. Liu, S.-F. Su, and W. Sun, "Robust stabilization of high-order nonlinear systems with unknown sensitivities and applications in humanoid robot manipulation," *IEEE Trans. Syst., Man, Cybern. Syst.*, early access, Aug. 30, 2019, doi: 10.1109/TSMC.2019.2931768.

[7] V.-T. Nguyen, C.-Y. Lin, S.-F. Su, and Q.-V. Tran, "Adaptive PID tracking control based radial basic function networks for a 2-DOF parallel manipulator," in *Proc. Int. Conf. Syst. Sci. Eng. (ICSSE)*, Jul. 2017, pp. 309–312.

[8] V.-T. Nguyen, C.-Y. Lin, S.-F. Su, N.-Q. Nguyen, and Q.-V. Tran, "Adaptive PD networks tracking control with full-state constraints for redundant parallel manipulators," in *Proc. Joint 17th World Congr. Int. Fuzzy Syst. Assoc. 9th Int. Conf. Soft Comput. Intell. Syst. (IFSA-SCIS)*, Jun. 2017, pp. 1–5.

[9] S. Mbakop, G. Tagne, O. Lakhal, R. Merzouki, and S. V. Drakunov, "Path planning and control of mobile soft manipulators with obstacle avoidance," in *Proc. 3rd IEEE Int. Conf. Soft Robot. (RoboSoft)*, May 2020, pp. 64–69.

[10] A. S. Lafmejani, A. Doroudchi, H. Farivarnejad, X. He, D. Aukes, M. M. Peet, H. Marvi, R. E. Fisher, and S. Berman, "Kinematic modeling and trajectory tracking control of an octopus-inspired hyper-redundant robot," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 3460–3467, Apr. 2020.

[11] Y. Mo, Z. Jiang, H. Li, H. Yang, and Q. Huang, "A biomimetic planning control method to anthropomorphize a 7-DOF redundant manipulator in unstructured environments," in *Proc. IEEE 9th Annu. Int. Conf. CYBER Technol. Autom., Control, Intell. Syst. (CYBER)*, Jul. 2019, pp. 695–700.

[12] J. Zeng, H. Wu, and Z. Wang, "A method for selecting redundant solutions of 7-DOF manipulator based on time optimality," in *Proc. IEEE 2nd Int. Conf. Autom., Electron. Electr. Eng. (AUTEEE)*, Nov. 2019, pp. 452–456.

[13] J. Oh, H. Bae, and J.-H. Oh, "Analytic inverse kinematics considering the joint constraints and self-collision for redundant 7DOF manipulator," in *Proc. 1st IEEE Int. Conf. Robotic Comput. (IRC)*, Apr. 2017, pp. 123–128.

[14] H. Su, W. Qi, C. Yang, J. Sandoval, G. Ferrigno, and E. D. Momi, "Deep neural network approach in robot tool dynamics identification for bilateral teleoperation," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 2943–2949, Apr. 2020.

[15] H. Su, C. Yang, G. Ferrigno, and E. De Momi, "Improved human–robot collaborative control of redundant robot for teleoperated minimally invasive surgery," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 1447–1453, Apr. 2019.

[16] Z. Li, W. Zuo, and S. Li, "Zeroing dynamics method for motion control of industrial upper-limb exoskeleton system with minimal potential energy modulation," *Measurement*, vol. 163, Oct. 2020, Art. no. 107964.

[17] G. Han, Z. Minglu, and Z. Xiaojun, "Summary of space trajectory planning for redundant manipulators," *Mech. Transmiss.*, vol. 10, pp. 176–180, Oct. 2016.

[18] W. Hao, "Research on the comparison of four intelligent algorithm," *Fire Control Command Control*, vol. S2, pp. 71–75, Oct. 2008.

[19] J. Wu, H. Wu, Y. Song, Y. Cheng, W. Zhao, and Y. Wang, "Genetic algorithm trajectory plan optimization for EAMA: EAST articulated maintenance arm," *Fusion Eng. Des.*, vols. 109–111, pp. 700–706, Nov. 2016.

[20] M. Najjarzadeh and A. Ayatollahi, "A comparison between genetic algorithm and PSO for linear phase FIR digital filter design," in *Proc. 9th Int. Conf. Signal Process.*, Oct. 2008, pp. 2134–2137.

[21] Y. Sun, J. Zhang, G. Li, Y. Wang, J. Sun, and C. Jiang, "Optimized neural network using beetle antennae search for predicting the unconfined compressive strength of jet grouting coalcretes," *Int. J. Numer. Anal. Methods Geomechan.*, vol. 43, no. 4, pp. 801–813, Mar. 2019.

[22] Q. Wu, Z. Ma, G. Xu, S. Li, and D. Chen, "A novel neural network classifier using beetle antennae search algorithm for pattern classification," *IEEE Access*, vol. 7, pp. 64686–64696, 2019.

[23] Y. Fan, J. Shao, and G. Sun, "Optimized PID controller based on beetle antennae search algorithm for electro-hydraulic position servo control system," *Sensors*, vol. 19, no. 12, p. 2727, Jun. 2019.

[24] Q. Wu, "Intelligent beetle antennae search for uav sensing and avoidance of obstacles.," *Sensors*, vol. 19, no. 8, pp. 1758–1759, Apr. 2019.

[25] Y. Mu, B. Li, D. An, and Y. Wei, "Three-dimensional route planning based on the beetle swarm optimization algorithm," *IEEE Access*, vol. 7, pp. 117804–117813, 2019.

[26] Q. Li, Z. Wang, and A. Wei, "Research on optimal scheduling of wind-pv-hydro-storage power complementary system based on bas algorithm," in *Proc. 2nd Int. Seminar Comput. Intell., Eng. Technol.*, 2018, pp. 1–4.

[27] S. Xie, X. Chu, M. Zheng, and C. Liu, "Ship predictive collision avoidance method based on an improved beetle antennae search algorithm," *Ocean Eng.*, vol. 192, Nov. 2019, Art. no. 106542. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0029801819306766

[28] J. Yang and Z. Peng, "Beetle-swarm evolution competitive algorithm for bridge sensor optimal placement in SHM," *IEEE Sensors J.*, vol. 20, no. 15, pp. 8244–8255, Aug. 2020.

[29] T. Chen, Y. Zhu, and J. Teng, "Beetle swarm optimisation for solving investment portfolio problems," *J. Eng.*, vol. 2018, no. 16, pp. 1600–1605, Nov. 2018.

[30] L. Wang, Q. Wu, F. Lin, S. Li, and D. Chen, "A new trajectory-planning beetle swarm optimization algorithm for trajectory planning of robot manipulators," *IEEE Access*, vol. 7, pp. 154331–154345, 2019.

[31] A. H. Khan, S. Li, and X. Luo, "Obstacle avoidance and tracking control of redundant robotic manipulator: An RNN-based Metaheuristic approach," *IEEE Trans. Ind. Informat.*, vol. 16, no. 7, pp. 4670–4680, Jul. 2020.

[32] M.-J. Lin and Q.-H. Li, "A hybrid optimization method of beetle antennae search algorithm and particle swarm optimization," in *Proc. Int. Conf. Electr., Control, Autom. Robot.*, Xiamen, China, Sep. 16-17, 2018, pp. 396–401.

[33] Q. Wu, H. Lin, Y. Jin, Z. Chen, S. Li, and D. Chen, "A new fallback beetle antennae search algorithm for path planning of mobile robots with collision-free capability," *Soft Comput.*, vol. 24, no. 3, pp. 2369–2380, Feb. 2020.

[34] Z. Dongyao, C. Pengwei, and L. Kuan, "Indoor location algorithm based on the search optimization of the beetle," *J. Hubei Univ. Nat.*, vol. 36, no. 4, pp. 427–431 and 455, Jan. 2018.

[35] C. X. H. Y. Q. Houjun, "Research on cable water resistance based on bas-bp model," *J. Hubei Univ. Technol.*, vol. 34, no. 4, pp. 35–39, 10 2019.

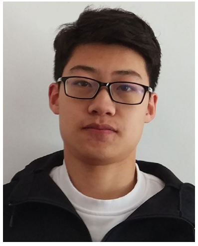
[36] Z. Li, C. Li, S. Li, and X. Cao, "A fault-tolerant method for motion planning of industrial redundant manipulator," *IEEE Trans. Ind. Informat.*, early access, Dec. 2, 2019, doi: 10.1109/TII.2019.2957186.

[37] X. Jiang and S. Li, "Beetle antennae search without parameter tuning (BAS-WPT) for multi-objective optimization," 2017, *arXiv:1711.02395*. [Online]. Available: http://arxiv.org/abs/1711.02395

[38] Y. Zhang, S. Li, and B. Xu, "Convergence analysis of beetle antennae search algorithm and its applications," *ArXiv*, vol. abs/1904.02397, 2019. [Online]. Available: https://arxiv.org/abs/1904.02397

[39] S. Dereli and R. Köker, "Calculation of the inverse kinematics solution of the 7-DOF redundant robot manipulator by the firefly algorithm and statistical analysis of the results in terms of speed and accuracy," *Inverse Problems Sci. Eng.*, vol. 28, no. 5, pp. 601–613, May 2020, doi: 10.1080/17415977.2019.1602124.

[40] S. Dereli and R. Köker, "A meta-heuristic proposal for inverse kinematics solution of 7-DOF serial robotic manipulator: Quantum behaved particle swarm algorithm," *Artif. Intell. Rev.*, vol. 53, no. 2, pp. 949–964, Feb. 2020.



YAOZONG CHENG is currently pursuing the bachelor's degree with the School of Automation Engineering, University of Electronic Science and Technology of China. His research interests include industrial robots, optimization methods, and intelligent control of vehicles.

CHUNXU LI received the B.E. degree in engineering from the Qingdao University of Science and Technology, Qingdao, China, in 2014, and the M.Sc. degree in engineering and Ph.D. degree in electrical and electronic engineering from Swansea University, Swansea, U.K., in 2016 and 2019, respectively. He is currently working as a Lecturer with the University of Plymouth. His research interests include robotics, automation, and computational intelligence.

SHUAI LI (Senior Member, IEEE) received the B.E. degree in precision mechanical engineering from the Hefei University of Technology, Hefei, China, in 2005, the M.E. degree in automatic control engineering from the University of Science and Technology of China, Hefei, in 2008, and the Ph.D. degree in electrical and computer engineering from the Stevens Institute of Technology, Hoboken, NJ, USA, in 2014. He is currently an Associate Professor (Reader) with Swansea University, Wales, U.K., leading the Robotic Laboratory, conducting research on robot manipulation and impedance control, multi-robot coordination, distributed control, intelligent optimization and control, and legged robots. He is the General Co-Chair of the 2018 International Conference on Advanced Robotics and Intelligent Control. He is the founding Editor-in-Chief of the *International Journal of Robotics and Control*.

ZHAN LI received the B.S. and M.S. degrees from Sun Yat-sen (Zhongshan) University, Guangzhou, China, in 2009 and 2011, respectively, and the Ph.D. degree from the INRIA DEMAR team affiliated at LIRMM/CNRS and the University of Montpellier II, Montpellier, France. He is currently working as an Associate Professor with the University of Electronic Science and Technology of China.