International Journal of Modern Engineering Research (IJMER)
www.ijmer.com Vol. 2, Issue. 5, Sep.-Oct. 2012 pp-4434-4438 ISSN: 2249-6645

Optimal Motion Planning For a Robot Arm by Using Artificial Bee Colony (ABC) Algorithm

P.V. Savsani, [1] R. L. Jhala, [2]
[1](Research Scholar PAHER University, Udaipur, Rajasthan, India)
2(Professor, Marwadi Education Foundation, Dept. of Mech. Engg.,Gujarat, India)

Abstract: *This work is concentrated to optimize the trajectory for planer two-link robot. The whole travel of the trajectory is divided into two parts, which consists of fourth order polynomial trajectory for the one part and fifth order trajectory for the second part. There are many optimization algorithms, which can be implemented to solve such problems. Evolutionary algorithms are also tried by many researchers to solve such trajectory optimization problems. Evolutionary algorithms are capable of giving global optimum solution and traditional optimization techniques converge to local optima and so they are not suitable for the trajectory problems. In this work artificial bee colony (ABC) algorithm is implemented to solve trajectory optimization problem. The objective function for the proposed ABC is to minimize traveling time and space, while not exceeding a maximum pre-defined torque. The objective function consist of four parameter, excessive driving torque, total joint traveling distance, total joint Cartesian length and total consumed time for robot motion.*

Keywords: *Artificial bee colony algorithm, 2R robotic arm, Trajectory.*

I. INTRODUCTION

Many researches on optimal trajectory planning have been reported in free workspace as well as in obstacle existence work space. The evolutionary algorithm is extensively used as practical optimization computation methods, because of the advantage that this algorithm can avoid local optimum value. This work focused on the trajectory optimization of 2R robotic arm in free work space as well as circular obstacle existence robot work space using ABC. Forth order and fifth order polynomials are used to describe the segments that connect initial, intermediate and final point at joint space.

In last year's, evolutionary algorithms have been applied in large number of fields. An optimal galloping trajectory proposed by Giju Chae and Jong Hyeon Park[2] which cost low energy and guarantees the stability of the quadruped robot. They optimized trajectory based on energy and stability using GA, which provides a robust and global solution to a multi-body, highly nonlinear dynamic system. For generating smooth trajectory planning for specified path, Zoller and Zentan [3] focused on the problem of the trajectory planning and dealt with constant kinetic energy motion planning. This method produced trajectory characteristics smoother and better than which did obtained from time optimal method. Zhe Tang et al. [6] proposed a third–order spline interpolation based trajectory-planning method to plan a smooth biped swing leg trajectory by reducing the instant velocity change.

Chwa et al. [7] proposed Missile Guidance Algorithm to generate on–line trajectory planning of robot arms of the interception of a fast maneuvering object. The authors employed the guidance law throughout the tracking phase, and dynamic constraints such as torque and velocity constraints and satisfied the matching condition of the position and velocity at the time of the interception altogether.

Garg and M. Kumar [8] use GA techniques for robot arm to identify the optimal trajectory based on minimum joint torque requirements. The authors use polynomial of 4th degree in time for trajectory representation to joint space variables. Pires and Machado [10] propose a path planning method based on a GA while adopting the direct kinematics and the inverse dynamics. The optimal trajectory is the one that minimize the path length, the ripple in the time evolution and the energy requirements, without any collision with the obstacle in the workspace. S. G. Yue et al. [4] focused on the problem of point-to-point trajectory planning of flexible redundant robot manipulator (FRM) in joint space. The proposed trajectory to minimize vibration of FRMs is based on GA.

Pires et al [5] use genetic algorithm to optimize a planar robot manipulator trajectory. The main purpose of this paper is to present the optimum robot trajectory in free workspace and obstacle existence workspace by using ABC. In the first part of the work mathematical model is formed. In the second part optimization of the developed robot trajectory is done by using ABC.

II. ARTIFICIAL BEE COLONY (ABC) TECHNIQUE

Artificial Bee Colony (ABC) Algorithm is an optimization algorithm based on the intelligent foraging behaviour of honey bee swarm. The colony of artificial bees consists of three groups of bees: employed bees, onlookers and scouts [9,11,12]. An employed bee searches the destination where food is available. They collect the food and returns back to its origin where they perform waggle dance depending on the amount of food available at the destination. The onlooker bee watches the dance and follows employed bee depending on the probability of the available food means more onlooker bee will follow the employed bee associated with the destination having more amount of food. The employed bee whose food source becomes abandoned convert into a scout bee and it searches for the new food source. For solving optimization problems the population is divided into two parts consisting of employed bees and onlooker bees. An employed bee searches the solution in the search space and the value of objective function associated with the solution is the amount of food associated with that solution. Employed bee updates its position using Equation (1) and it updates new position if it is better than the previous position, i.e it follows greedy selection.

$$v_{ij} = x_{ij} + R_{ij}(x_{ij} - x_{kj}) \qquad (1)$$

Where v_{ij} is the new position of employes bee, x_{ij} is the current position of employed bee, *k* is a random number between (1, *N(population size)/2)* $\neq i$ and *j* =1, 2,...,*Number of design variables*. R_{ij} is a random number between (-1, 1). An onlooker bees chooses a food source depending on the probability value associated with that food source, p_i , calculated using Equation (2).

$$p_i = \frac{F_i}{\sum_{n=1}^{N/2} F_n} \qquad (2)$$

Where F_i is the fitness value of the solution *i* and *N/2* is the number of food sources which is equal to the number of employed bees.

The Employed bee whose position of the food source cannot be improved for some predetermined number of cycles than that food source is called abandoned food source. That employed bee becomes scout and searches for the new solution randomly using Equation (3).

$$x_i^j = x_{\min}^j + rand(0,1)(x_{\max}^j - x_{\min}^j) \qquad (3)$$

The value of predetermined number of cycles is an important control parameter of the ABC algorithm, which is called "limit" for abandonment. The value of limit is generally taken as Number of employed bees.
Step by step procedure for the implementation of ABC is given as follows.

Step 1

Initialize ABC parameters which are necessary for the algorithm to proceed. These parameters includes population size which indicates the number of employed bees and onlooker bees, number of generations necessary for the termination criteria, value of limit, number of design variables and respective range for the design variables.

Step 2

Generate random population equal to the population size specified. Each population member contains the value of all the design variables. This value of design variable is randomly generated in between the design variable range specified. First half of the population will consist of employed bees. Each population member associated with employed bees indicates each food source.

Step 3

Obtain the value of objective function for employed bees. The value of objective function so obtained indicates the amount of nectar (food) associated with that destination (food source).

Step 4

Update the position of employed bees using Equation (1). If the value of objective function of the new solution is better than the existing solution , replace the existing solution with the new one.

Step 5

Calculate probability associated with the different solutions using Equation (2). Onlooker bee follows a solution depending on the probability of that solution. So more the probability of the solution more will be the onlooker bee following that solution.

Step 6

Update the position of onlooker bees using Equation (1). If the value of objective function of the new solution is better than the existing solution, replace the existing solution with the new one

Step 7

Identify abandon solution and replace it with the newly generated solution using Equation (3)

Step 8

Continue all the steps from step 3 until the specified number of generations are reached.

III. MATHEMATICAL MODELLING

For the present work two degree of freedom planar robotic arm is considered as shown in figure 1, where the endeffector is required to move from starting point to goal point in free work space as well as without colliding with the obstacle in work space. For the motion planning, point-to-point trajectory is taken which is connected by several segments with continuous acceleration at the intermediate via point. ABC is used as optimization tool.

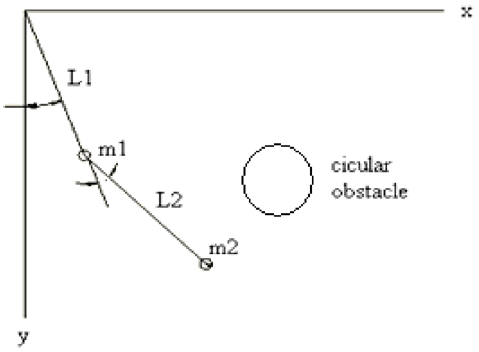


Figure 1 Two degree of freedom planar robotic arm

For the motion planning, point-to-point trajectory is taken which is connected by several segments with continuous acceleration at the intermediate via point as shown in figure 2. For a robot, the number of degrees of freedom of a manipulator is n and the number of end-effectors degree of freedom is m. If one wishes to be able to specify the position, velocity, and acceleration at the beginning and the end of a path segment, a fourth order and a fifth order polynomial are used. Let us assume that there is *mp* intermediate via points between the initial and final points[1].

$$\theta_{i,i+1}(t) = a_{i0} + a_{i1}t_i + a_{i2}t_i^2 + a_{i3}t_i^3 + a_{i4}t_i^4, \quad (i = 0,....,mp-1) \qquad (4)$$

Where ($a_{i0},...,a_{i4}$) are constants, and the constraint are given as:

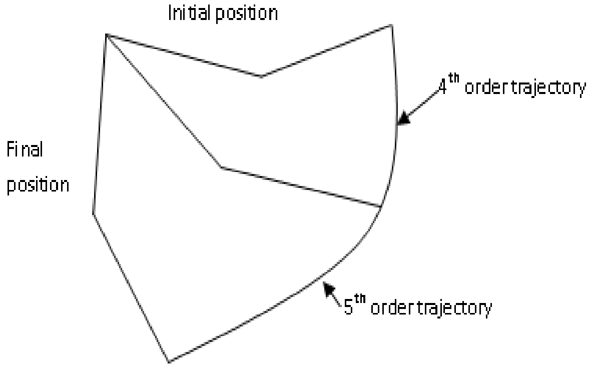


Figure 2 Intermediate point on planed trajectory

$$\theta_i = a_{i0} \quad (5)$$

$$\theta_{i+1} = a_{i0} + a_{i1}T_i + a_{i2}T_i^2 + a_{i3}T_i^3 + a_{i4}T_i^4 \quad (6)$$

$$\dot{\theta} = a_{i1} \quad (7)$$

$$\dot{\theta}_{i+1} = a_{i1} + 2a_{i2}T_i + 3a_{i3}T_i^2 + 4a_{i4}T_i^3 \quad (8)$$

$$\ddot{\theta} = 2a_{i2} \quad (9)$$

Where T_i is the execution time from point i to point $i+1$. The five unknowns can be solved as[1]. The intermediate point *(i+1)'s* acceleration can be obtained as:

$$\ddot{\theta}_{i+1} = 2a_{i2} + 6a_{i3}T_i + 12a_{i4}T_i^2 \quad (10)$$

The segment between the number *mp* of intermediate points and the final point can be described by fifth order polynomial as:

$$\theta_{i,i+1}(t) = b_{i0} + b_{i1}t_i + b_{i2}t_i^2 + b_{i3}t_i^3 + b_{i4}t_i^4 + b_{i5}t_i^5, \quad (i = mp) \quad (11)$$

Where the constraints are given as:

$$\theta_i = b_{i0} \quad (12)$$

$$\theta_{i+1} = b_{i0} + b_{i1}T_i + b_{i2}T_i^2 + b_{i3}T_i^3 + b_{i4}T_i^4 + b_{i5}T_i^5 \quad (13)$$

$$\dot{\theta}_i = b_{i1} \quad (14)$$

$$\dot{\theta}_{i+1} = b_{i1} + 2b_{i2}T_i + 3b_{i3}T_i^2 + 4b_{i4}T_i^3 + 5b_{i5}T_i^4 \quad (15)$$

$$\ddot{\theta}_i = 2b_{i2} \quad (16)$$

$$\ddot{\theta}_{i+1} = 2b_{i2} + 6b_{i3}T_i + 12b_{i4}T_i^2 + 20b_{i5}T_i^3 \quad (17)$$

In addition, these constraints specify a linear set of six equations with six unknowns whose solution is explained in [1]. As formulated above, the total parameters to be determined are the joint angles of each intermediate via point ($n\times m_p$ parameters), the joint angular velocities of each intermediate point ($n\times m_p$ parameters), the execution time for each segment (m_p+1 parameters), and the posture of the final configuration (n-m). Therefore, for 2-link robot case, it used m_p= 1, n =2 and one degree of freedom of redundancy for the final point, there are six parameters to be determined.

IV. FITNESS FUNCTION

Present robot trajectory use the four parameters to meet the criteria of the robotic manipulator in free work space. All parameters are translated into penalty functions to be minimized. Each parameter is computed individually and is integrated in the fitness function evaluation. The fitness function f_f adopted for evaluating the aspirant trajectories is defined as:

$$f_f = \beta_1 f_{ot} + \beta_2 f_q + \beta_3 f_c + \beta_4 t_T \quad (18)$$

The optimization goal consists in finding a set of design parameters that minimize f_f according to the priorities given by the weighting factors β_i (i = 1,.., 4), where each different set of weighting factors must results in a different solution. For this work the weight factors are,

$$[\beta_1, \beta_2, \beta_3, \beta_4] = [1,2,2,1].$$

The f_{ot} index represents the amount of excessive driving, in relation to the maximum torque τ_i max, that is demanded for the i^{th} joint motor for the trajectory. The index f_q represents the total joint traveling distance of the manipulator, The index f_c represents total Cartesian trajectory length and The index t_T represents the total consumed time for robot motion. All four index are calculated as given in [1]. For obstacle existence workspace, obstacle avoidance objective function *f*ob has been combined with free space fitness function to form over all fitness function *f*, as shown below:

$$f = f_f / f_{ob} \quad (20)$$

By f_{ob}, the robot manipulator has the ability to avoid the obstacle collision during its movement from point to point in side the workspace and it is calculated as [1]. For the present work six parameter are optimized :

$$\left[q_1, q_2, q_{3,}\ \dot{q}_1, \dot{q}_2, \dot{q}_3\ t_1, t_2 \right]$$

Where q_i and q_i are intermediate joint angle and velocity for i^{th} joint respectively, t_1 is execution time from initial to intermediate via point, and t_2 is execution time from intermediate to final point. Limits of all the variables are as follows:

$$-\pi \le q_i \le +\pi \quad (21)$$

$$-\pi/4 \le \dot{q}_i \le +\pi/4 \quad (22)$$

$$0.1 \le t_i \le 8 \quad (23)$$

$$-\pi \le q_g \le +\pi \quad (24)$$

Implementation of ABC

For the case study, robotic arm movement is considered as motion of human hand, which start at (x=0 m, y= -0.76 m,) and get final position (x= 0.76 m, y= 0). Various parameters for the illustration are taken as: Length of link 1 =0.3048 m ,Length of link 2=0.3048 m , Mass of link 1 and 2= 175 gms. For ABC maximum generation =2000 and population= 10. The main target here is to minimizing traveling time and space, while not exceeding the predefined torque(joint-1=45 N.m, joint-2=20N.m) ,in free workspace and without collision with any obstacle. Finally results of ABC are discussed.

V. RESULTS AND DISCUSSION

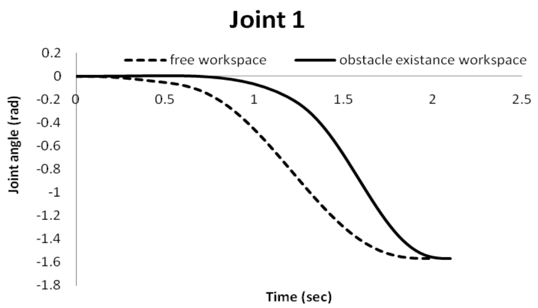
Trajectory of 2R robot is optimized using ABC. Table-1 shows total traveling time, total joint traveling distance and total cartensian trajectory length for both free work space and obstacle avoidance work space.

Table 1 Comparison of optimum results obtained in free workspace and obstacle existence workspace

Result value	Free workspace	Obstacle existence workspace
Total traveling time(sec)	2.0106	2.1078
Total joint traveling distance(rad)	1.5776	6.6167
Total Cartesian trajectory length(m)	3.1403	3.2623
Fitness value	11.2367	16.4495

As noted from the table 1 the values of all the parameters are more in obstacle existence workspace. The amount of increment allows the robot to manipulate in the work space without colliding with obstacle. For the optimum result function evaluation is taken as 20000 for both free and obstacle existence workspace.

Figure 3 shows the variation of joint angle with respect to time in free robot workspace. Dotted line represents the variation of joint angle in free workspace while full line shows the angle variation in obstacle existence workspace. Joint 1 angle varies from 0 to ~ -1.6 rad in 2 sec when the arm moves in free workspace, while it takes more than 2 sec in obstacle existence workspace to cover the same range of angle. Joint 2 angle is almost zero when the robot arm moves in free workspace and variation of 0 to ~1.5 rad in is obtained when it moves in obstacle existence workspace.



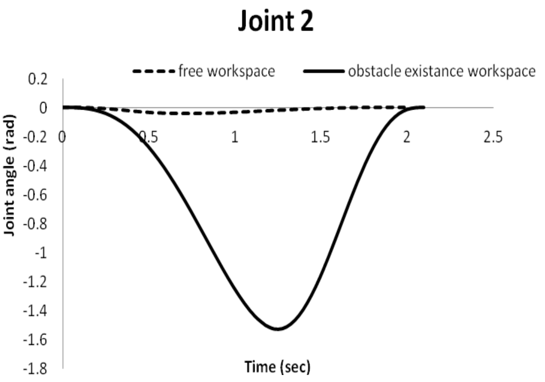


Figure 3 joint angle variation with respect to time

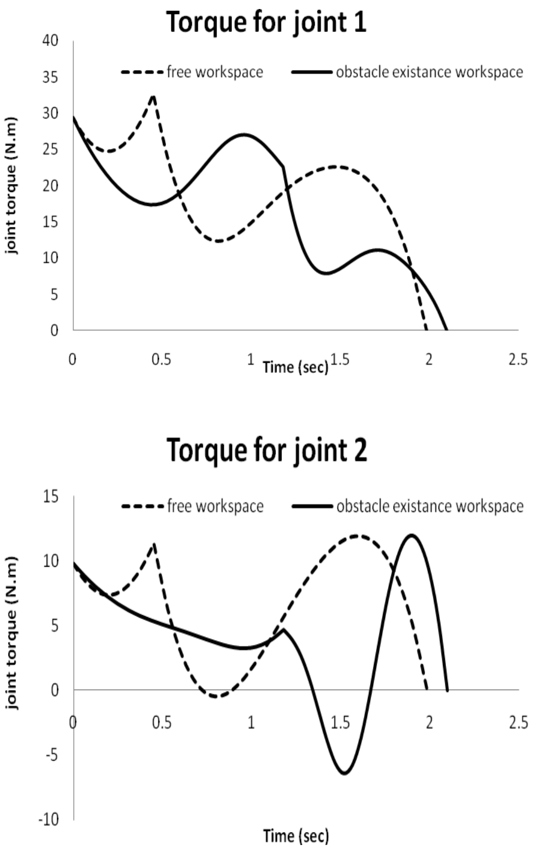


Figure 4 torque variation with respect to time

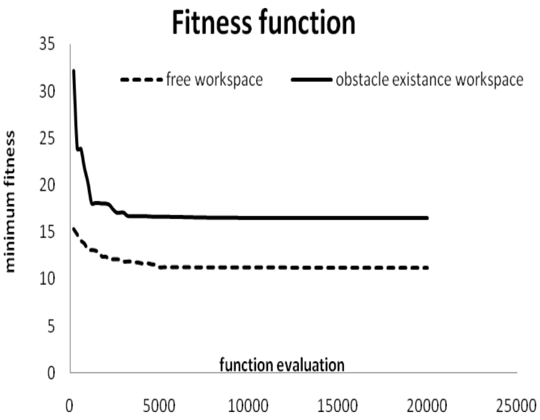


Figure 5 fitness function with function evalution

Figure 4 depicts that for both the workspace (free and obstacle existence) torque for joint 1 and joint 2 are within the predefined limit. Torque of joint 1 varies from 0 to ~33 Nm and joint 2 varies from 0 to ~ 12 Nm when robot arm manipulate in free workspace over 2 sec. Manipulation of robot arm in obstacle existence workspace gives the torque variation for joint 1(0 to ~30 Nm) and joint 2 (~ -6 Nm to ~13Nm) over 2.1 sec.

Fitness value variation with free and obstacle existence workspace is shown in figure 5. Fitness value in obstacle existence workspace is more than the fitness value in free workspace.

VI. CONCLUSIONS

Trajectory planning method based on ABC with specific objective functions is presented. Case study of human motion is taken for the 2R planner robotic arm and trajectory is optimized in free and obstacle existence workspace. Comparison shows that fitness value for the free workspace is lesser than value in obstacle existence workspace. The joint torque of the robot did not exceed its maximum pre-defined torque. Since ABC uses the direct kinematics, the singularities do not constitute a problem.

REFERENCES

[1] Kazem, I. B.,Mahdi, A. I.,Oudha, A. T. (2008). Motion Planning for a Robot Arm by Using Genetic Algorithm. Jordan Journal of Mech. and Ind. Engg. 2, pp. 131-136.

[2] Chae, G., Park, J., H. (2007). Galloping Trajectory Optimization and Control for Quadruped Robot Using Genetic Algorithm. In: IEEE International Conference on Robotics and Biomimetics, pp. 1166-1171.

[3] Zoller, Z., Zentan, P. (1999). Constant Kinetic Energy Robot Trajectory Planning. peridica polytechnicaser. Mech. Eng., 43(2), pp. 213 – 228.

[4] Yue, S.,G., Henrich, D., Xu, X., L., Toss, S.K. (2002). Point-to-Point Trajectory Planning of Flexible Redundant Robot Manipulators Using Genetic Algorithms. J. Robotica, 20, pp. 269—280.

[5] Solteriro, P., Tenreiro, M., Moura O., 2003, “Fractional Order Dynamic in a Genetic Algorithm”.In:11th International Conference On Advanced Robotics, pp. 264-269. Colombia, Portugal.

[6] Tang, Z., Zhou, C., Sun Z. (2003). Trajectory Planning For Smooth Transition of A Biped Robot. In:IEEE International Conference on Robotic & Automation,pp. 2455—2460.

[7] Chwa, D., Kang, J., Im, K.,Choi, J. (2003). Robot Arm Trajectory Planning Using Missile Guidance Algorithm. In: SICE Annual Conference, pp. 2056-2061.

[8] Grag, P. , Kumar, M. (2002). Opimization Techniques Applied To Multiple Manipulators for Path Planning and Torque Minimization. Engineering Applications of Artificial Intelligence,15(3), pp. 241-252.

[9] Savsani, V., Rao, R., Vakharia, D. P. (2010). Multi objective optimization of mechanical elements using artificial bee colony optimization technique. ASME Early Career Technical Journal

[10] Solteiro, P., Machado, J. (2000). A GA Perspective Of The Energy Requirement For Manipulators Maneuvering In A Workspace With Obstacles. Cec 2000-Congress On Evolutionary Computation, pp. 1110- -1116, Santiago, California, USA.

[11] Craig, J. (1989). Introduction to Robotics: Mechanics and Control.Wesley

[12] Karaboga, D. (2005).An Idea Based On Honey Bee Swarm For Numerical Optimization. Technical Report-TR06, Erciyes University, Engineering Faculty, Computer Engineering Department.

[13] Basturk, B. Karaboga, D. (2006). An Artificial Bee Colony (ABC) Algorithm for Numeric function Optimization. IEEE Swarm Intelligence Symposium, May 12-14, Indianapolis, Indiana, USA.