Energy Efficient Trajectory Planning by a Robot Arm using Invasive Weed Optimization Technique

Abhronil Sengupta[1], Tathagata Chakraborti[2], Amit Konar[3]
Dept. of Electronics and Telecommunication Engg.
Jadavpur University
Kolkata, India
senguptaabhronil@gmail.com[1], tathagata.net@live.com[2], konaramit@yahoo.co.in[3]

Atulya Nagar
Computer and Math Sciences
Liverpool Hope University
Liverpool, England
nagara@hope.ac.uk

***Abstract*— In this paper we propose an energy efficient method for path planning by a robot arm. First we have developed a cost function that provides a set of via points determining a suitable path according to the obstacles present in the surroundings and other restrictions in its motion. Then we fit a suitable polynomial to this set of interior points to ensure that the journey is smooth and takes place with the consumption of minimum energy. The set of via-points as well as the minimum energy path have been determined using IWO (Invasive Weed Optimization) which is a new search heuristic based on the colonizing property of weeds. Application of such evolutionary algorithms in trajectory planning is advantageous because the exact solution to the path planning problem is not always available beforehand and must be determined dynamically. The entire procedure has been demonstrated through simulations. The above formulation is very simple and versatile and can be effectively applied to a variety of situations.**

Keywords- robot arm, mechanical energy, optimization, IWO, trajectory planning, obstacle avoidance.

I. INTRODUCTION

The problem of motion planning and obstacle avoidance has been a very important area of research for the last decade. It finds wide applications in various fields including industries and medical science. There are mainly two areas of concern. Firstly, there is the need to find a solution to the inverse kinematics problem for a given goal. The forward kinematics functions may give multiple solutions to the same problem and it is of interest to find the correct solution. Furthermore these solutions are to be obtained using various minimization techniques. This is to ensure that the final path obtained caters to the desired conditions of minimum path, time, energy or some such parameter. Thus for the obstacle free case the cost function has terms to minimize the above criteria only, whereas in the case of obstacles there is an additional penalty term as will be discussed in later sections.

Several research works have been conducted in this field during the past few years. Saab and VanPutte [1] used topographical maps to solve the problem. In [2] an algorithm has been proposed for obstacle avoidance using convexity of arm links and obstacles. However, the method suffers from the problem of laborious computations.

Recently genetic algorithms (GA) have been used in this field. Tian and Collins et al. [3] have proposed a method for a planar robot arm motion planning using GA. However only point obstacles have been considered and the environment is entirely two dimensional. Thus scope of practical implementation is limited. Other works in this field include [4] and [5].

Neural networks have also been used by some researches to solve the problem of obstacle avoidance in motion planning. Ziqiang Mao and T.C. Hsia et al. [6] employed neural networks to solve the inverse kinematics problem of redundant robot arms.

Other researches in this field include Potential Field method [7]. Helguera et al. [8] proposes an effective approach to solve many of the problems created by the Potential Field method.

In our paper we propose a novel method to solve the problem of motion planning without obstacle collision using Invasive Weed Optimization (IWO) proposed by Mehrabian and Lucas [9]. It mimics the ecological behavior of weeds. This metaheuristic algorithm has attracted researchers because of its reduced computational cost. A suitable cost function has been chosen according to the present application which has been minimized to track the optimal path between an initial and final configuration of the robot arm joints. The algorithm has been described for a two-degree-of-freedom robot arm but can be easily extended to other arm configurations. Let us first take a look at the prototype robot arm used for our work.

II. PROPOSED WORK

A. The Robot Arm

In the following discussion we use a robot arm with two links and two movable sections each having one degree of freedom. Their movements are described by angles theta and phi which are defined with respect to the co-ordinate system as shown in Figure 1.

- The first section of length L1 moves in the vertical plane as described by theta (θ) measured from the +ve z-axis. The vertical plane in which it moves is displaced from the x-axis by an angle α.

978-1-4577-1124-4/11/$26.00 ©2011 IEEE

- The second section of length L2 moves in the horizontal plane as described by phi (ϕ) measured from the +ve x-axis.

The arm ends in a gripper whose co-ordinates are given by

$$x = L1 * \sin(\theta) * \cos(\alpha) + L2 * \cos(\phi) \quad (1)$$
$$y = L1 * \sin(\theta) * \sin(\alpha) + L2 * \sin(\phi) \quad (2)$$
$$z = L1 * \cos(\theta) \quad (3)$$

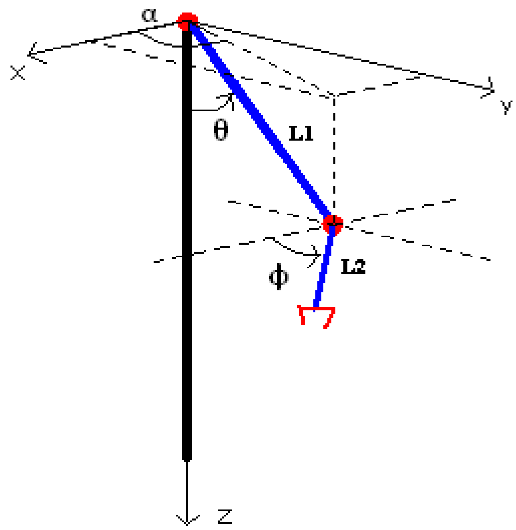

Figure 1.Variation of phi (ϕ) and theta (θ)

Here for our analysis we take, without loss of generality, α = 90deg, so that the upper arm is entirely confined in the Y-Z plane.

B. Determination of Via-Points

Here we determine a set of via-points in between the initial position and the goal by using IWO and then fit cubic polynomials in between these interior points to form a smooth trajectory subject to constraints of continuous angular velocities and energy minimization. First we describe the cost function used to locate the interior points:

1) Minimization of Redundant Joint Rotations

Here we purpose to minimize the energy consumed in redundant arm movements. We know that for a rotation of angle $\Delta\Omega_i$ at the ith joint, the energy expended is given by $\sum_i T_i \Delta\Omega_i$, where T_i is the torque provided by the motor at the i^{th} joint. Thus the problem reduces to one of ensuring that the final angle is attained as quickly as possible. Thus the cost function becomes

$$f = K1 * |\theta_f - \theta| + K2 * |\phi_f - \phi| \quad (4)$$

where K1, K2 are constants of proportionality and (θ_f,ϕ_f) denotes the goal position. In the above equation $\theta = \theta_{prev} + \Delta\theta$ and $\phi = \phi_{prev} + \Delta\phi$ where $(\theta_{prev}, \phi_{prev})$ represents the previous via point and $(\Delta\theta, \Delta\phi)$ denotes the angular displacement between the two via points.

2) Obstacle Avoidance

Obstacle avoidance in case of robotic arm movements is a lot more complex proposition than in the case of point objects. This is because in the current situation we must not only make sure that the end point, i.e. the gripper avoids collision but also ensure that in all its orientations all the parts of the arm avoid collisions with the obstacles. First we model the robot arm as a series of consecutive spheres each of radius R1, where R1 is determined by the amount of safety margin required. The proposed model is shown in Figure 2. Now the obstacle may be of different shapes and sizes and developing different penalty terms for each of them is a rather futile process. As a simple model we have considered obstacles to be spherical in our calculations, where the obstacles in cases being non-spherical have been replaced with spheres that just enclose the body, i.e. a square or a rectangle is replaced with its circum-sphere and so on, as illustrated in Figure 2.

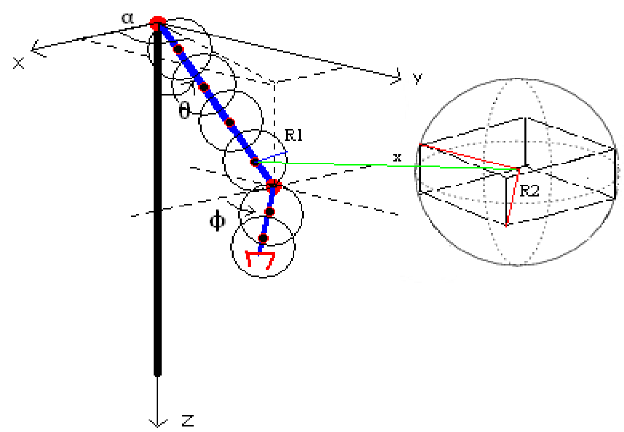

Figure 2.Sensors on robot arm and representation of obstacles by spheres

In the above discussion it is clear that if the distance 'x' between the sensor and the centre of the sphere representing the obstacle is less than R1 + R2 then there is a chance of collision and hence in such cases the cost function should incorporate a large penalty term. In cases where this distance is larger, the penalty term should be negligible. Thus we have for the i^{th} sensor and j^{th} obstacle a penalty term of the form

$$K3 * \exp(-x_{ij}/(R1_i + R2_j)) \quad (5)$$

where K3 is a constant of proportionality and is in general much greater than K1 and K2 since this lends more weight to the penalty term as obstacles must be avoided at any cost. Here we have taken the penalty term in the form of an exponential. The term is equal to K3 at the centre of the obstacle (x=0) and decays exponentially to K3/e at x = R1+R2. Thus the penalty component of the cost function finally takes the form (summing over all obstacles and all sensors):

$$\sum_i \sum_j K3 * C_{ij} * \exp(-x_{ij}/(R1_i + R2_j)) \quad (6)$$

Here the constant C_{ij} ensures that the penalty term is not added beyond $x_{ij}=R1_i+R2_j$. Thus

$$C_{ij} = \begin{cases} 1, & x_{ij} < R1_i + R2_j \\ 0, & otherwise. \end{cases}$$

3) Constraints on Joint Angle

The last term in the cost function incorporates a large penalty if the arm tries to take angles beyond the given restrictions. This is to ensure that in trying to avoid obstacles and minimize energy along its path the arm always takes only those values that agree with the constraints on the joint angle displacements. The penalty term is of the form $K4 * \sum_j C_j$ where K4 is very large in relation to K1, K2, K3 and

$$C_j = \begin{cases} 1 \; if \; angular \; displacement \; at \; the \; j^{th} \; joint \; is \\ \quad outside \; the \; defined \; limit \\ 0 \; if \; angular \; dispacements \; are \; within \; limits \end{cases}$$

For the present problem, the constraints are

$0 \leq \theta \leq 90deg \; and \; 0 \leq \phi \leq 180deg$.

Thus in its final form, the cost function becomes:

$$f = K1 * |\theta_f - \theta| + K2 * |\phi_f - \phi| + \sum_i \sum_j K3 * C_{ij} * \exp\left(-\frac{x_{ij}}{R1_i + R2_j}\right) + K4 * \sum_j C_j \quad (7)$$

To ensure uniform distribution of via-points in the joint space, the optimizer has been employed to produce optimized values of $\Delta\theta$ and $\Delta\phi$ within a certain given range ($-\Delta\theta_{max}$, $\Delta\theta_{max}$) and ($-\Delta\phi_{max}$, $\Delta\phi_{max}$) which have been added to the θ_{prev} and ϕ_{prev} values, and the process is repeated till the goal is reached.

C. Fitting an Optimum Trajectory

Here we develop an energy efficient method of fitting a smooth trajectory to the set of via-points found above. Let us consider that we have *n*+1 via-points (including initial and final points). We fit smooth cubic polynomials for theta and phi as functions of time in between each of these points as shown below:

$$\theta = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \quad (8)$$

$$\phi = b_0 + b_1 t + b_2 t^2 + b_3 t^3 \quad (9)$$

where the coefficients are determined partly by a set of boundary conditions and partly by energy minimization criterion. The boundary conditions for the polynomial between the i^{th} and $(i+1)^{th}$ via-points are as follows:

for i = 1,2...n; $\theta = \theta_i, \phi = \phi_i$; at t = 0

$\bar{\theta} = \bar{\theta}_i, \bar{\phi} = \bar{\phi}_i$..

and $\theta = \theta_{i+1}, \phi = \phi_{i+1}$. at t = T (10)

where T is the time to move from the i^{th} to the $(i+1)^{th}$ via-point, and $\bar{\theta}$,$\bar{\phi}$ are the first time derivatives of theta and phi respectively. Condition of continuity of angular displacements is of course necessary, whereas condition of continuity of angular velocity at every via-point avoids jerky movement of the arm. At the initial point angular velocities are zero. Using the boundary conditions, we get the coefficients as:

for i =1, 2...n; $a_0 = \theta_i, b_0 = \phi_i$;

$$a_1 = \bar{\theta}_i, b_1 = \bar{\phi}_i;$$

$$a_2 = {}^1/_{T^2}\,(\theta_{i+1} - \theta_i - \bar{\theta}_i.T - a_3.T^3),$$

$$b_2 = {}^1/_{T^2}\,(\phi_{i+1} - \phi_i - \bar{\phi}_i.T - b_3.T^3) \quad (11)$$

Evidently the above conditions give three equations for four unknown coefficients, and the final equation is provided by the energy term. At any point the mechanical energy of the arm will be given by the summation of kinetic and potential energies of each of the arm sections as shown below:

$$E = \sum_i (K.E._i + P.E._i)$$

where $K.E._i = {}^1/_2 . m_i . v_i^2 + {}^1/_2 . I_{CM_i} . \omega_i^2$

and $P.E._i = m_i . g . h_i$

Here ω_i = angular velocity of i^{th} link, i.e. $\omega_1 = \bar{\theta}$ & $\omega_2 = \bar{\phi}$.

v_i = translational velocity $= \sqrt{\bar{x}_i^2 + \bar{y}_i^2 + \bar{z}_i^2}$, where $\bar{x}_i, \bar{y}_i, \bar{z}_i$ are the first time derivatives of the co-ordinate of the centre of mass (CM) of the i^{th} link.

I_{CM_i} = moment of inertia of the i^{th} link about its CM.

m_i = mass of the i^{th} link.

h_i = height of the i^{th} link from the ground, which is taken as the reference for zero potential energy.

Upon calculations, the total mechanical energy of the system is found to be equal to

$$E = {}^1/_2\, m_2 L_1^2 . \bar{\theta}^2 + {}^1/_6\,(m_1 L_1^2 . \bar{\theta}^2 + m_2 L_2^2 . \bar{\phi}^2) + {}^1/_2\, m_2 L_1 L_2 \cos(\theta)\cos(\phi) . \bar{\theta} . \bar{\phi} + gH(m_1 + m_2) - gL_1({}^1/_2\, m_1 + m_2)\cos(\theta) \quad (12)$$

In this equation we put the expressions for θ, ϕ and $\bar{\theta}$, $\bar{\phi}$; and replace a_2, b_2 in terms of a_3,b_3 so that E is now a function of time, and a_3,b_3. The energy integrated over a single time interval will give a measure of the total energy consumed and this is a function of a_3 and b_3 (the integration is done using recursive adaptive Simpson quadrature technique). Thus we can find an optimum value of the remaining coefficient by minimizing F_K, where

$$F_K(a_3, b_3) = \int_0^T E.dt \quad (13)$$

This completes description of the entire trajectory subject to smooth motion and minimization of energy consumed.

D. The Optimization Technique

Here we have used IWO (Invasive Weed Optimization) as our optimization technique. The algorithm is described below in the next page.

III. AN OUTLINE OF IWO ALGORITHM

A. Generation of an initial population:

An initial population of weeds (solutions) is generated randomly over the D dimensional space.

B. Reproduction:

The plants will now produce seeds depending on their relative fitness which will be spread out over the problem space. Each seed, in turn, will grow into a flowering plant. Thus, if S_{max} and S_{min} denote respectively the number of seeds produced by plants with best and worst fitness respectively then seed count of plants will increase linearly from S_{min} to S_{max} depending on their corresponding fatnesses.

C. Dispersal of seeds through search space:

The produced seeds are randomly distributed over the D dimensional search space by random numbers drawn from a normal distribution with zero mean but with a varying variance. This leads to a local search around each plant and naturally the search is much more extensive in the neighborhood of fitter plants. However, the standard deviation (SD), σ, of the normal distribution decreases over the generations from an initial value, σ_{max}, to a value, σ_{min}, and is determined by the following equation:

$$\sigma = \left(\frac{iter_{max} - iter}{iter_{max}}\right)^n (\sigma_{max} - \sigma_{min}) + \sigma_{min} \tag{14}$$

where σ is the SD at the current generation and $iter_{max}$ is the maximum number of iterations while n is the non linear modulation index. This is the adaptation property of the algorithm.

D. Competitive exclusion:

If a plant does not reproduce it will become extinct. Hence this leads to the requirement of a competitive exclusion in order to eliminate plants with low fitness values. This is done to limit the maximum number of plants in the colony. Initially fast reproduction of plants take place and all the plants are included in the colony. The fitter plants reproduce more than the undesirable ones. The elimination mechanism is activated when the population exceeds a stipulated P_{max}. The plants and produced seeds are ranked together as a colony and plants with lower fitness values are eliminated to limit the population count to P_{max}. This is the selection property of the algorithm. The above steps are repeated until maximum number of iterations is reached.

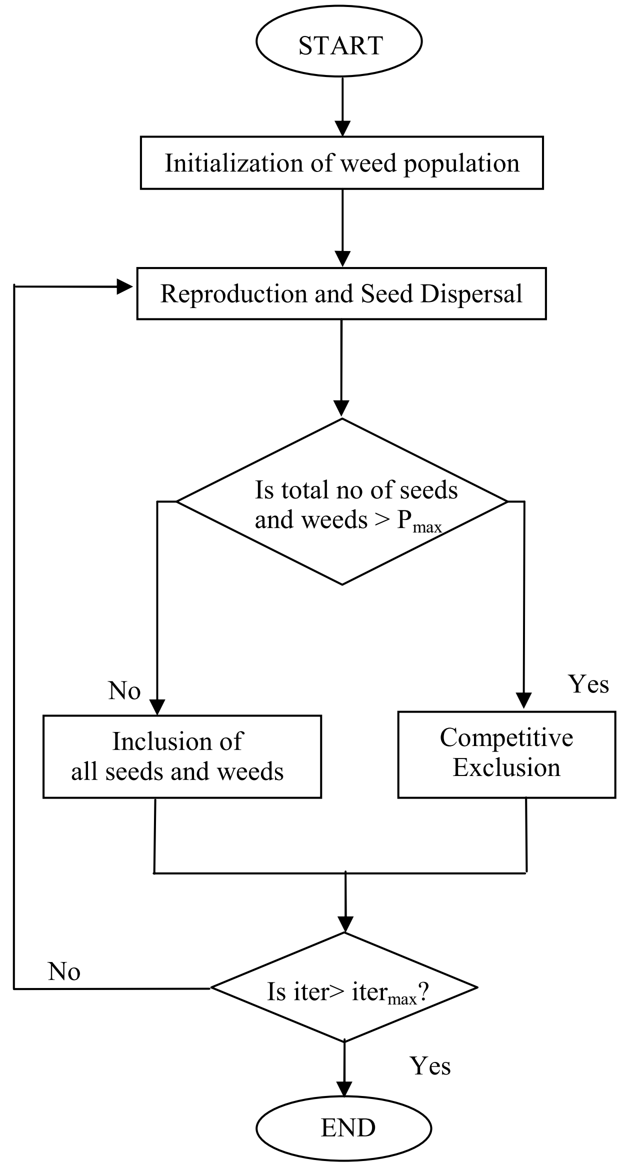

Figure 3.Flowchart for IWO algorithm

IV. Simulation Results

A. Simulation of the Trajectory

Here we perform a test run to see if the arm can reach its goal by following the above procedure. The arm starts from initial position defined by $(\theta, \phi)_{initial}$ and reaches the goal defined by $(\theta, \phi)_{final}$. The different parameters used are summarized below. It should be noted here that these are not absolute measurements of any quantity but are relative values estimated to hold in proportionality.

Note here that for the cubical obstacle shown the edge length is 1.5, so that R2 for the equivalent sphere becomes $(\sqrt{3}/2)*1.5=1.29$. As mentioned before the constant K4 is made much larger than K3, which is again larger in comparison to K1 and K2.

The path taken by the robot arm is shown below in Figure 4. The plot of θ converges with the plot of both θ and ϕ in absence of any obstacle but the plot of ϕ bends away in case of an obstacle in its path. The minimum distance to the four blocks (obstacles) are plotted along the path as shown in Figure 5. Here also we see how the plots curve away as they near the distance equal to the equivalent radii of the blocks (=1.29). In Figure 6 we plot the minimum distance to the four blocks in absence of the penalty term in the cost function and note how it goes below the safety margin (indicating hit).

TABLE I. SIMULATION PARAMETERS

SIMULATION PARAMETERS	ESTIMATED VALUES
$(\theta, \phi)_{initial}$	(0,0) deg
$(\theta, \phi)_{final}$	(90,90) deg
K1	10
K2	10
K3	100
K4	10000
R1	0.5
R2	1.29
H	20
T	1
m1	1
m2	1
L1	5
L2	5
obstacle1	(3,8,3)
obstacle2	(3,9.5,3)
obstacle3	(3,8,4.5)
obstacle4	(3,9.5,4.5)
(-Δθ, Δθ)	(-20,20)deg
(-Δϕ, Δϕ)	(-20,20)deg

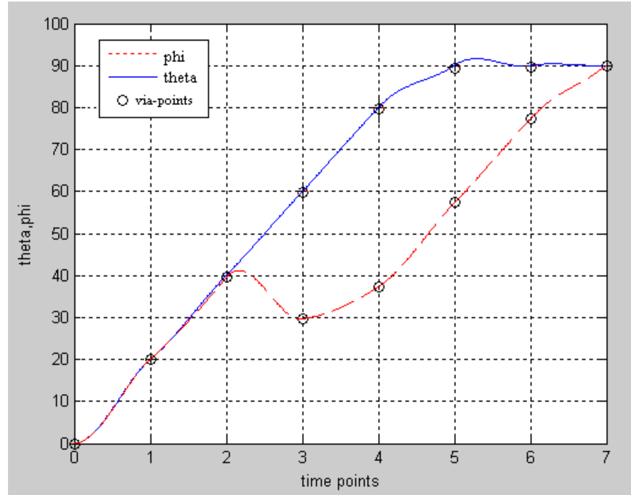

Figure 4.Plot of theta and phi along the path

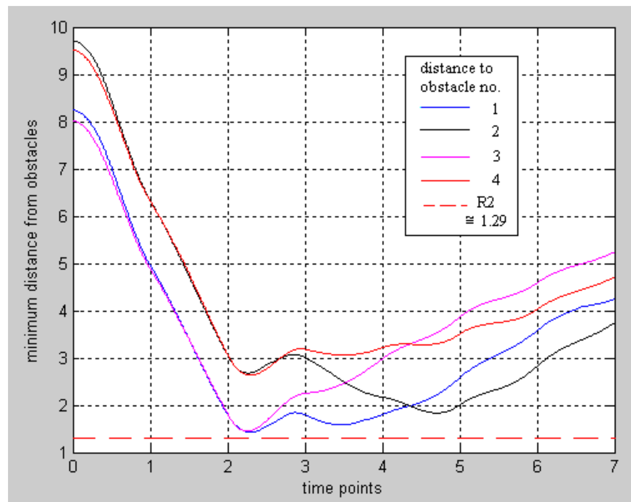

Figure 5.Plot of minimum distance to obstacles

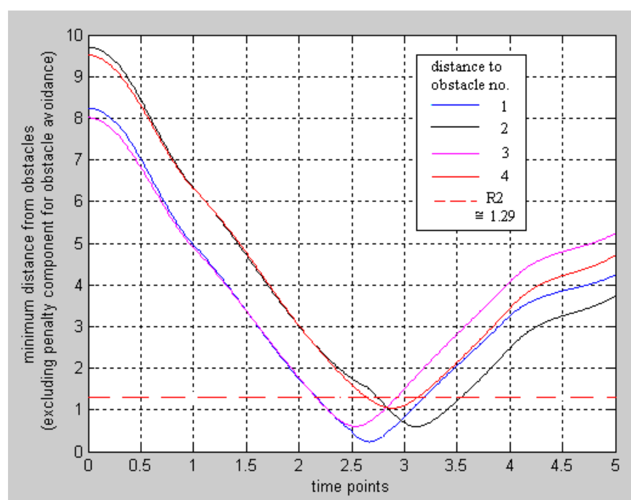

Figure 6.Plot of minimum distance to obstacles (excluding penalty term)

B. Optimiser Parameters

The parameters used for the IWO algorithm are as follows. These parameters have been developed after extensive cultivation and have been found to give optimum performance.

TABLE II. PARAMETERS OF THE OPTIMISER

DESCRIPTION	VALUE
initial population	5
max plant no.	10
max seed no.	3
min seed no.	0
modulation index	3
$iter_{max}$	100

The σ_{max} and σ_{max} have been taken as 10% and 0.004% of the search range respectively.

V. Conclusions

The algorithm determines a set of potential via points for obstacle avoidance. The obtained trajectory ensures smooth motion of the end-effector and optimizes energy expenditure during motion from one via point to the next. As we have already seen above, model simulations show that the IWO algorithm can be successfully used as an optimization tool in motion planning and obstacle avoidance for robotic arms. It is an effective optimization tool that can be exploited to give successful results in this scenario. This process helps in dynamically determining the most suitable path subject to conditions in the surroundings. Favorable results from simulations call for further research in this area. There is scope for adopting such algorithms in real robots and environments, and in cases of further complexities, for example on cases where there are larger number of joints and multiple solutions for the inverse kinematics problem.

Future research work will involve studying the mathematical effects of the optimizer parameters on the efficiency of the problem. Further analysis can be done on the applicability of other popular evolutionary algorithms in the motion planning of the robotic arm and studying their relative performance.

References

[1] Y Saab and M. VanPutte, "Shortest path planning on topographical maps," IEEE Transactions on Systems, Man, and Cybernetics–Part A 1999;29(1) ,pp.139-150.

[2] E.G. Gilbert and D.E. Johnson , "Distance function and their application to robot path planning in the presence of obstacles," IEEE J. of Robotics and Automation RA-1(1), pp.21-30 (1985).

[3] L. Tian and C. Collins, "An effective robot trajectory planning using a genetic algorithm," Elsevier, Mechatronics 14 (2004),pp.455-470.

[4] A.M.S. Zalzala and K.K.Chan, "An evolutionary solution for the control of mechanical arms," in Proceedings of ICARCV 1994, Singapore,1994.

[5] D. Pack, G. Toussaint and R. Haupt, " Robot trajectory planning using a genetic algorithm," SPIE 1996;2824,pp.171-182.

[6] Z. Mao and T.C. Hsia, " Obstacle avoidance inverse kinematics solution of redundant robots by neural networks," Robotica (1997), vol.15, pp.3-10, 1997 Cambridge University Press.

[7] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile manipulators," Int. J. of Rob. Res., vol. 5, no. 1, pp. 90–98, 1986.

[8] C. Helguera and S.Zegloul, "A local based method for manipulators path planning in heavy cluttered environments," in Proceedings of the 2000 IEEE Int. Conf. on Robotics and Automation, San Fransisco, CA, USA,2000, pp.3467-3472.

[9] A. R. Mehrabian and C. Lucas, "A novel numerical optimization algorithm inspired from weed colonization," Ecological Informatics, vol. 1, pp. 355–366, 2006.

[10] S.Lang, "Topological spaces," Real and Functional Analysis (Springer-Verlag, New York Inc.,1993) Chapter 2.