Advanced Materials Research Vols. 488-489 (2012) pp 1772-1776
Online available since 2012/Mar/15 at www.scientific.net
© (2012) Trans Tech Publications, Switzerland
doi:10.4028/www.scientific.net/AMR.488-489.1772

Online Trajectory Generation of a 2 Link Robot in Presence of Obstacle

Mahdi Esmaeilzadeh Abardeh[1, a], Alireza Akbarzadeh[2, b]

[1, 2] Faculty of Mechanical Engineering, Ferdowsi University of Mashhad, Mashhad, Iran

[a]Mahdi_Esmaeilzadeh_A@yahoo.com, [b]Ali_Akbarzadeh_T@yahoo.com

Keywords: Obstacle Crossing, Two Link Robot, PSO, Online Trajectory Generation.

Abstract. Online coordination of multiple robots working on a single workstation requires special attention. In these applications it is generally necessary that the robot arm follow a desired path in workspace so that it does not crash with any obstacle or the other robots. In this paper a two link planar robot crossing a rectangular obstacle is considered. The proposed idea is to define the relationship between obstacle's dimensions and the required joints trajectories parameters which allow the robot to reach its destination in the presence of an obstacle. First, the desired path for robot avoiding the obstacle is defined using a fourth degree polynomial. Corresponding robot joints trajectories are defined using a sinusoidal function with four parameters. Next, Design of Experiments (DOE) technique is utilized. Three levels for width and length of the obstacle are used as input and a full factorial DOE with nine experiments is defined. Instead of using the inverse kinematics, Particle Swarm Optimization (PSO) algorithm is used to obtain parameters of the robot joint sinusoidal functions. A second degree regression is used to obtain the relationship between each of the four sinusoidal function parameters and the obstacle dimensions. The obtained regression equations allow online changes to the trajectory as obstacle dimensions change. Four case studies, different obstacle dimensions, are simulated using the two link robot. The results show that using the obtained relationships the robot reaches its desired destination, with high accuracy, while avoiding the obstacles.

Introduction

Today, speed, accuracy and ability of robots in various fields are increasingly used in different industrial applications. In these applications it is generally necessary that the robot arm crosses a desired path and moves in workspace so that it does not crash with any obstacle. Often in these cases, the robot move in a dynamic environment and obstacle dimensions are not predictable. In such circumstances it is necessary that robot motion path be determined according to obstacle dimensions. Many of the techniques presented in the field of obstacle avoidance, require solving inverse kinematics of the robot. Additionally, some researchers use optimization techniques for path planning that are time consuming. Ata, A. A. [1] has studied a three link planar robot passing a desired obstacle using optimization algorithm of Generalized Pattern Search (GPS).They obtained the primary path using a cubic spline and three points around the obstacle. This path was next optimized using GPS algorithm with the goal of obstacle avoidance and robot joints movement minimization. Saska, M. [2] has proposed a method for mobile robots path planning based on Particle Swarm Optimization (PSO) optimization of Ferguson cubic splines. In his method, the mobile robot motion path is developed by connecting the splines. The resulting splines are next optimized with the goal of minimum path traveled to the target while avoiding obstacles. Bitwas, A. et al [3] have presented optimal trajectory planning of a robot with 3 degrees of freedom using genetic algorithm. In his study, robot trajectory is modeled using a fourth degree polynomial. The coefficients of the polynomial are obtained with the simultaneous goal of minimizing robot energy and maximizing the distance to the obstacle. Ozaki, H. [4] investigated optimal B-spline joint trajectory generation for collision-free movements of a manipulator under dynamic constraints. The values of the B-spline control points which describe the reference trajectories are optimized using the Complex Method while considering full dynamic and kinematic constraints. Wu, L. [5], has presented a method by Octree model for passing obstacles while moving.

Most of the above discussed methods as well as other similar robot trajectory generation methods used for collision free path generation are offline and is time consuming. In the present paper, the proposed collision free trajectory can be generated online to pass a rectangular obstacle with any dimensions. A two link robot is used for simulation and demonstration of the proposed method.

Main Idea

An obstacle encountered by the robot can be embedded in a rectangular area that encompasses the outside perimeter of the obstacle. If the relationship between dimensions of the obstacle and the robot joint parameters are known, then it is possible to determine the joint parameters which avoid obstacle collision. This approach eliminates the need to solve the robot inverse kinematics and is computationally efficient. The resulting relationships are simple and allow online trajectory generation. To achieve this goal, Design of Experiments (DOE) technique is used [6]. DOE approach is extensively used in engineering applications. For example, DOE is used in path planning to determine ideal grinding condition [7]. In this study, full factorial experiment is conducted and parameters of the robot joints for trajectories that avoid the collision are determined for different values of obstacle dimensions. Particle swarm optimization (PSO) algorithm is used instead of robot inverse kinematics to determine the robot joints trajectory [8,9,10].

Introducing the robot kinematics and path parameters

Consider a two link robot as shown in Fig. 1. This robot has 2 degrees of freedom and moves in the horizontal plane. Assume sinusoidal trajectory for each of the robot joint angles. We can write,

$$\theta_1 = A_1 \sin(\omega_1 t + \phi_1) + \gamma_1. \quad , \quad \theta_2 = A_2 \sin(\omega_2 t + \phi_2) + \gamma_2. \tag{1}$$

Parameters A, ω, ϕ, γ in the above relationships are unknown parameters which their optimal value is to be determined by the optimization algorithm. The position of end effector in the robot is obtained by the following relationships:

$$x_a = l_1 \cos(\theta_1) + l_2 \cos(\theta_1 + \theta_2). \quad , \quad y_a = l_1 \sin(\theta_1) + l_2 \sin(\theta_1 + \theta_2). \tag{2}$$

The obstacle in the path of the robot is assumed rectangular (with rounded edges) with length=*l* and width=*w*. The position of the obstacle is shown in Fig. 2. Five constraint points are manually chosen around the obstacle. A desired trajectory that passes through the five control points will insure the robot does not cross the obstacle. The Fig. 2 shows the five constraint points and robot's desired path. Coordinates of the constraint points are as follows:

$$\begin{cases} d = 0.56 \\ l_1 = 0.30\,,\, l_2 = 0.35 \end{cases} \quad \begin{cases} x_1 = L/2 + 0.12 \\ y_1 = 0 \end{cases} \quad \begin{cases} x_2 = L/2 + 0.03 \\ y_2 = d - w - 0.03 \end{cases} \quad \begin{cases} x_3 = 0 \\ y_3 = d - w - 0.10 \end{cases}$$

Position of points 4 and 5 is considered symmetrical with respect to the origin of base frame {B}. Therefore, equation of optimal path for the end-effector can be defined using a 4th degree polynomial as,

$$y_d = c_4 x_d^4 + c_3 x_d^3 + c_2 x_d^2 + c_1 x_d + c_0. \tag{3}$$

Coefficients of Eq. 3 can be determined using the five constraint points.

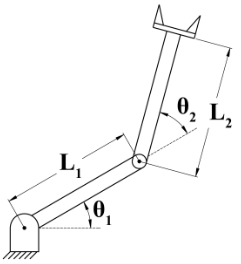

Fig. 1.Structure of two link robot

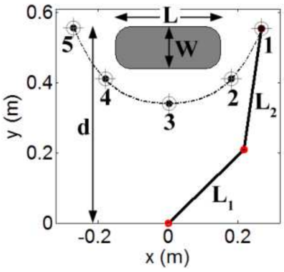

Fig. 2.Introducing parameters of problem

Determination of objective function

An objective function can be defined as sum of square's error between the actual and desired path as,

$$fitness = \sum_{j=1}^{n}\left((x_{dj}-x_{aj})^2+(y_{aj}-y_{dj})^2\right). \tag{4}$$

In the above equation the path is divided into *n* segments. (x_{dj},y_{dj}) and (x_{aj},y_{aj}) are the jth position of desired and actual points, respectively. It is critical that the points around the start and end of the path are strictly followed. Therefore, these points are given significantly more weight than the other points on the path. The objective function is modified as follows:

$$\begin{aligned} fittness &= \sum_{j=1}^{3}\left((x_{dj}-x_{aj})^2+(y_{dj}-y_{aj})^2\right)\times 100 \\ &+\sum_{j=4}^{n-3}\left((x_{dj}-x_{aj})^2+(y_{dj}-y_{aj})^2\right)+\sum_{j=n-2}^{n}\left((x_{dj}-x_{aj})^2+(y_{dj}-y_{aj})^2\right)\times 100. \end{aligned} \tag{5}$$

In addition to the requirement of the tip to follow the desired path, the body of the actual robot, links, must also avoid interference with the obstacle. See Fig. 3. Therefore, the last link is divided into K points and a penalty is assumed for any of these points that become within a 1 cm space of the obstacle.

$$Penalty = \sum_{j=1}^{n}\sum_{l=1}^{i}(y_{obstacle}-y_{interference})^2. \tag{6}$$

Where the index *i* represents the number of points that interfere with the obstacle, $y_{obstacle}$ is the height of the obstacle lowest point and $y_{interference}$ is the height of interference point.

Consider Fig 4. According to Eq. 2, the robot end effector actual position is a function of time. On the other hand, Eq. 3, the desired end-effector position (x_d,y_d) is not is a function of time. Note that the fitness function, Eq. 4, compares these two values. Therefore, for any given time, we must obtain the corresponding actual and desired position. To accomplish this we assume the end-effector is following the desired path with constant speed v . Therefore, time at any point along the path can be computed by,

$$S(x)=\int_{x}^{a}\sqrt{1+(dy_d/dx_d)^2}\,dx_d. \quad , \quad t=\frac{S(x)}{v}. \tag{7}$$

Where *S* represents the length of arc traveled along the desired path and *a* represents the value along the *x*-axis of the starting point. Using Eq. 7 for each x_d , one can calculate time, t, and subsequently obtain x_a and y_a .

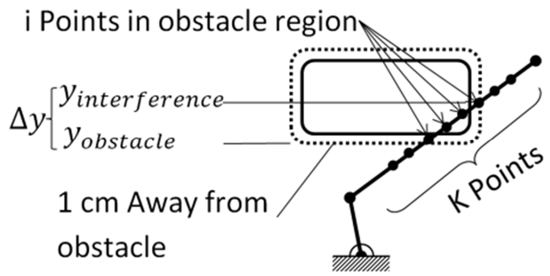

Fig. 3. Robot arm and the obstacle

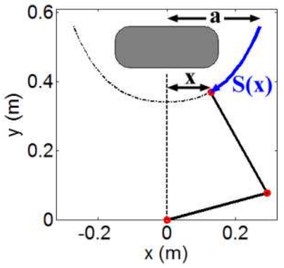

Fig. 4. Desired end-effector path

Experiments Design and Relationships Extraction

A full factorial DOE based on three levels for each dimension of the obstacle is designed. A total of nine experiments are required. See Table 1. In each experiment optimization algorithm is implemented and optimal values for each of the robot joint trajectory parameters $(A, \omega, \phi, \gamma)$ are recorded.

Table 1.Order of experiments performed

		L		
		0.30	0.25	0.20
W	0.12	Test 1	Test 2	Test 3
	0.10	Test 4	Test 5	Test 6
	0.08	Test 7	Test 8	Test 9

Experiments Results. To obtain the relationship for each of the parameters $(A, \omega, \phi, \gamma)$ with obstacle length and width, it is necessary that the values obtained from PSO algorithm are homogeneous. See Table 2. To do this, we can write,

$$-A\sin(\omega t+\phi)=A\sin(-\omega t-\phi).$$
$$A\sin(\omega t+\phi+\pi)=A\sin(-\omega t-\phi). \quad \text{and} \quad -A\sin(\omega t+\phi)=A\sin(\omega t+\phi+\pi). \tag{8}$$

Table 2. Experiments results for the joints

	Joint #1				Joint #2				Obstacle	
Test No.	A_1	ω_1	ϕ_1	γ_1	A_2	ω_2	ϕ_2	γ_2	L	W
1	-0.816	0.953	0.378	1.068	1.633	0.768	0.131	0.445	0.30	0.12
2	-0.768	0.999	0.389	1.053	1.474	0.836	0.103	0.586	0.25	0.12
3	-0.736	1.034	0.408	1.057	1.362	0.903	0.090	0.681	0.20	0.12
4	-0.795	0.971	0.440	1.106	1.497	0.820	0.106	0.500	0.30	0.10
5	-0.781	0.996	0.467	1.116	1.449	0.864	0.131	0.543	0.25	0.10
6	-0.766	1.017	0.500	1.135	1.389	0.914	0.152	0.587	0.20	0.10
7	-0.801	0.970	0.504	1.165	1.522	0.827	0.158	0.401	0.30	0.08
8	-0.786	0.992	0.549	1.178	1.397	0.895	0.153	0.514	0.25	0.08
9	-0.797	0.993	0.602	1.222	1.323	0.955	0.167	0.574	0.20	0.08

Relationship of the path parameters. MATLAB software is used and a 2nd order regression is performed. The relationship between joint trajectory parameters and obstacle size is obtained as,

$$\begin{aligned}
&A_1(L,W)=-1.475+2.963L+6.987W-2.869L^2-19.07LW-8.39W^2.\\
&\omega_1(L,W)=0.4736+2.15L+6.109W-2.338L^2-14.8LW-10.74W^2.\\
&\phi_1(L,W)=1.6-3.08L-9.415W+1.539L^2+16.82LW+6.086W^2.\\
&\gamma_1(L,W)=2.286-3.97L-9.701W+4.052L^2+16.95LW+11.19W^2.\\
&A_2(L,W)=1.734-2.731L-5.851W+5.735L^2+17.88LW+16.36W^2.\\
&\omega_2(L,W)=1.208-0.9549L-0.04626W-0.1239L^2-1.739LW-4.679W^2.\\
&\phi_2(L,W)=0.801-2.273L-6.411W+1.942L^2+12.59LW+9.89W^2.\\
&\gamma_2(L,W)=-0.2707+3.232L+10.7W-6.575L^2-15.95LW-24.31W^2.
\end{aligned} \tag{9}$$

Simulation

In this section, four case studies are simulated. It is shown that using Eq. 9. The robot can successfully generate and follow the desired trajectory and avoid the different size obstacles. Consider Fig 5. Each of the case studies uses an obstacle with different values for width and length. It should be noted that the obstacle dimensions used in these simulations are different from the values used in the DOE table. Summary of simulation results is graphically represented in Fig.5.

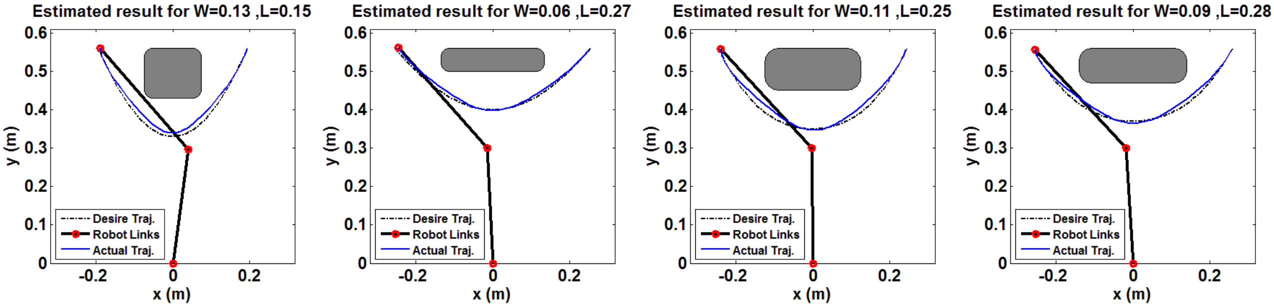

Fig. 5. Four case studies – variable size obstacles

Conclusion

In this paper, a two link planar robot moving in a workspace with variable dimension rectangular obstacle is studied. A relationship between obstacle's dimensions and the required joints trajectories parameters which allow the robot to reach its destination in the presence of an obstacle is obtained. First, motions of robot joints are modeled using a sinusoidal equation with four parameters. Next, design of experiments methodology is used and a full factorial experiment is defined. Obstacle width and length are used as input. Using particle swarm optimization algorithm the four parameters of the robot joint sinusoidal functions are determined. A second degree regression is used to obtain the relationship between each of the four sinusoidal function parameters and the obstacle dimensions. Four case studies are simulated. The results indicate that the developed regression model is quite effective in avoiding the obstacle in the robot workspace. This paper contributes by presenting an algorithm which results in an on-line trajectory generation method for robots with the presence of variable size obstacle. The algorithm potentially improves robotic cell productivity.

References

[1] Ata, A.A. and T.R. Myo, Collision-free trajectory planning for manipulators using generalized pattern search. International Journal of Simulation Modelling, 2006. 5(4): p. 145-154.

[2] Saska, M., et al. Robot path planning using particle swarm optimization of ferguson splines. 2006. Hamburg.

[3] Biswas, A., B.L. Deekshatulu, and S.S. Roy. Energy optimal trajectory planning of a robotic manipulator using genetic algorithm. 2010. Durgapur, West Bengal.

[4] Ozaki, H. and C.-j. Lin. Optimal B-spline joint trajectory generation for collision-free movements of a manipulator under dynamic constraints. 1996. Minneapolis, MN, USA: IEEE.

[5] Wu, L. and Y. Hori. Real-time collision-free path planning for robot manipulator based on octree model. 2006. Istanbul.

[6] Jiju Antony. Design of experiments for engineers and scientists, 2003, Butterworth-Heinemann

[7] Wang, Y.T. and Jan, Y.J. Path planning for robot-assisted grinding processes. 2001. ICRA. IEEE International Conference on Robotics and Automation, 331 - 336 vol.1

[8] Kennedy, J. and R. Eberhart. Particle swarm optimization. 1995. Perth, Aust: IEEE.

[9] Shi, Y. and R. Eberhart. Modified particle swarm optimizer. 1998. Anchorage, AK, USA: IEEE.

[10] Eberhart, R. and J. Kennedy. New optimizer using particle swarm theory. 1995. Nagoya, Jpn: IEEE.

Key Engineering Materials II
10.4028/www.scientific.net/AMR.488-489

Online Trajectory Generation of a 2 Link Robot in Presence of Obstacle
10.4028/www.scientific.net/AMR.488-489.1772