International Journal of Engineering, Mathematical and Physical Sciences
ISSN: 2517-9934
Vol:6, No:12, 2012

Motion Control of a 2-link Revolute Manipulator in an Obstacle-Ridden Workspace

Avinesh Prasad, Bibhya Sharma, and Jito Vanualailai

Abstract**—**In this paper, we propose a solution to the motion control problem of a 2-link revolute manipulator arm. We require the end-effector of the arm to move safely to its designated target in a priori known workspace cluttered with fixed circular obstacles of arbitrary position and sizes. Firstly a unique velocity algorithm is used to move the end-effector to its target. Secondly, for obstacle avoidance a turning angle is designed, which when incorporated into the control laws ensures that the entire robot arm avoids any number of fixed obstacles along its path enroute the target. The control laws proposed in this paper also ensure that the equilibrium point of the system is asymptotically stable. Computer simulations of the proposed technique are presented.

Keywords**—**2-link revolute manipulator, motion control, obstacle avoidance, asymptotic stability.

I. Introduction

A robot manipulator is a mechanism made up of rigid links connected by different joints [1]. The two basic types of joints commonly found in literature are rotational (revolute) or translational (prismatic). A revolute joint is like a hinge that allows relative rotation between two links whereas a prismatic joint provides a linear sliding movement between two links [2].

The pioneering work in the field of path planning of a robot arm was done by Meyer [3] in 1993 where the author proposed a findpath scheme based on the velocities of the various components of the planar revolute arm. Vanualailai et.al [2] in 1998 used the Lagrange method to derive a set of differential equations governing planar robot system and then proposed a solution for the motion control of two robot arms using the Lyapunov-based control scheme. Sacks [4] in 2003 studied path planning for planar articulated robots using configuration spaces and compliant motion. Vanulailai et.al [5], [6] in 2004 and 2007 used Lyapunov-based control scheme to solve the findpath problem of 2-link and 3-link revolute manipulator arms. Sharma et.al [7] in 2011 used neural networks to move a planar robot arm in the presence of fixed and artificial obstacles.

Other work in this area involve that of Martinez et al. in [8] via Kohonen networks; Josin et al. [9] using inverse kinematics and neural networks; Guez et al. [10] using neural networks and Lee et al. [11] who studied a 3 degree of freedom manipulator. A major problem faced in [10] and [11] was the incorporation of the system singularities into the proposed solution.

In this paper, we provide a relatively simple approach to solve the motion control of a 2-link revolute robot arm which is made up of two rotational joints. We first derive the kinematic model of the robot arm and then model its motion from an initial state to the target in the presence of fixed obstacles. The mechanical singularities associated with the system are treated as artificial obstacles and are avoided using the obstacle avoidance scheme. Our main aim is to design a set of continuous control laws that ensure asymptotic stability of the system. A unique and scalable algorithm for target convergence and obstacle avoidance is proposed that works for any number fixed obstacles of various sizes.

The rest of the paper is organized as follows: In Section II, we derive the kinematic model of the revolute arm. The motion planning and control of the planar robot arm in the absence of obstacles is considered in Section III. In Section IV, we propose an obstacle avoidance technique to control the motion of the robot arm in a workspace cluttered with fixed circular obstacles. Computer simulation of the generated path with the proposed technique is also presented. The stability of the system is studied in Section V. Finally, in Section VI concluding remarks on the contributions and future work are made.

II. Modelling The Revolute ARM

We have considered a simple 2-link revolute manipulator arm that has two rotational joints in the z_1z_2 - plane as shown in the Fig. 1. The arm consists of two rigid links which are connect via revolute joints; the first link can rotate through 360° and the second link which carries the payload at the gripper can rotate through 180° with respect to the first link.

With the help of Fig. 1, we assume that:

i. the 2-link revolute manipulator arm is anchored at the origin;
ii. link 1 has a length of ℓ_1 and has an angular position $\theta_1(t)$, measured counterclockwise from the z_1-axis at time *t*;
iii. link 2 has a length of ℓ_2 and has an angular position $\theta_2(t)$, measured counterclockwise from link 1 at time *t*;
iv. the coordinate of the end-effector is (x, y).

Avinesh Prasad is an Assistant Lecturer in Mathematics at the University of the South Pacific. He is currently pursuing a PhD Degree in the School of Computing, Information & Mathematical Sciences, University of the South Pacific, Suva, Fiji (e-mail: Prasad_ai@usp.ac.fj).

Bibhya Sharma and Jito Vanualailai are Associate Professors at the University of the South Pacific.

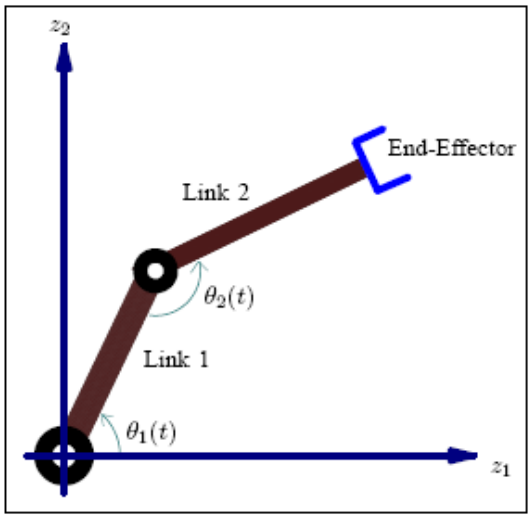

Fig. 1 Schematic representation of a 2-link revolute manipulator arm

Remark: We can express the position of the end-effector completely in terms of the state variables, $\theta_1(t)$ and $\theta_2(t)$ as

$$x = \ell_1 \cos\theta_1 - \ell_2 \cos(\theta_1 + \theta_2),$$
$$y = \ell_1 \sin\theta_1 - \ell_2 \sin(\theta_1 + \theta_2).$$

We first look at the motion of the end-effector of the robot arm. Let a velocity of *v* be applied to the end-effector. Then the kinematic equations of the end-effector can be expressed as

$$\dot{x} = u_1,$$
$$\dot{y} = u_2$$

where u_1 and u_2 are the z_1 and z_2 components, respectively, of *v*. We now look for the kinematic model of this 2-link manipulator arm. Let $d = \sqrt{x^2 + y^2}$ be the distance from the origin to the end-effector, then using the cosine rule, we see that

$$d^2 = \ell_1^2 + \ell_2^2 - 2\ell_1\ell_2 \cos\theta_2$$

Differentiate this with respect to *t* and simplify, we see that

$$\dot{\theta}_2 = \frac{\ell_1 \cos\theta_1 - \ell_2 \cos(\theta_1 + \theta_2)}{\ell_1\ell_2 \sin\theta_2} u_1 + \frac{\ell_1 \sin\theta_1 - \ell_2 \sin(\theta_1 + \theta_2)}{\ell_1\ell_2 \sin\theta_2} u_2$$

Next, we look at the kinematic equation relating to $\theta_2(t)$. Let ϑ be the angular position of the end-effector relative to the origin, then

$$\tan\vartheta = \frac{y}{x}$$

from which we get

$$\dot{\vartheta} = \frac{u_2 \cos\vartheta - u_1 \sin\vartheta}{d}.$$

Using the cosine rule again, we see that

$$\ell_2^2 = \ell_1^2 + d^2 - 2\ell_1 d \cos(\theta_1 - \vartheta).$$

Differentiate this with respect to t and simplify, we get

$$\dot{\theta}_1 = \frac{u_2 \cos\vartheta - u_1 \sin\vartheta}{d} + \frac{(u_1 \cos\vartheta + u_2 \sin\vartheta)(\ell_1 \cos(\theta_1 - \vartheta) - d)}{\ell_1 d \sin(\theta_1 - \vartheta)}.$$

Thus the kinematic equations for the revolute manipulator arm is

$$\left.\begin{aligned}
\dot{\theta}_1 &= \frac{u_2 \cos\vartheta - u_1 \sin\vartheta}{d} + \frac{(u_1 \cos\vartheta + u_2 \sin\vartheta)(\ell_1 \cos(\theta_1 - \vartheta) - d)}{\ell_1 d \sin(\theta_1 - \vartheta)} \\
\dot{\theta}_2 &= \frac{\ell_1 \cos\theta_1 - \ell_2 \cos(\theta_1 + \theta_2)}{\ell_1\ell_2 \sin\theta_2} u_1 + \frac{\ell_1 \sin\theta_1 - \ell_2 \sin(\theta_1 + \theta_2)}{\ell_1\ell_2 \sin\theta_2} u_2 \\
\theta_1(0) &= \operatorname{atan2}(y(0), x(0)) + \cos^{-1}\left[\frac{x(0)^2 + y(0)^2 + \ell_1^2 - \ell_2^2}{2\ell_1\sqrt{x(0)^2 + y(0)^2}}\right] \\
\theta_2(0) &= \cos^{-1}\left[\frac{\ell_1^2 + \ell_2^2 - x(0)^2 - y(0)^2}{2\ell_1\ell_2}\right]
\end{aligned}\right\} \quad (1)$$

system (1) is a description of the instantaneous angular velocities of the revolute manipulator arm. Here u_1 and u_2 are classified as the controllers. We shall use the vector notation $\mathbf{x}(t) = (\theta_1(t), \theta_2(t))$ to refer to the angular position of the robot in the z_1z_2 -plane.

III. Motion Control in the Absence of Obstacles

In our motion control problem, we want the end-effector of the robot arm to start from an initial position, move towards its target and converge to the center of the target. The target, *T*, considered in this paper is a disk of center (p_1, p_2) and radius r_T which is described as:

$$T = \left\{(z_1, z_2) \in \mathbf{R}^2 : (z_1 - p_1)^2 + (z_2 - p_2)^2 \le r_T^2\right\}.$$

A. Target Convergence

For the end-effector of the robot arm to move from its initial position to the target position, we adopt the following form of velocity algorithm from [7] which is depended on the initial and final positions of the robot:

$$v = \frac{|v_0| \left\|(p_1 - x, p_2 - y)\right\|}{\left\|(p_1 - x(0), p_2 - y(0))\right\|},$$

where $|v_0|$ is the initial velocity of the end-effector at $t = 0$. The function v is defined, continuous and positive over the domain

$$D = \left\{\mathbf{x} \in \mathbf{R}^2 : (x(0),\ y(0) \ne (p_1,\ p_2)\right\}.$$

Now if we define

$$\mathbf{x}_e = \operatorname{atan2}(p_2, p_1) + \cos^{-1}\left[\frac{p_1^2 + p_2^2 + \ell_1^2 - \ell_2^2}{2\ell_1\sqrt{p_1^2 + p_2^2}}\right] + \cos^{-1}\left[\frac{\ell_1^2 + \ell_2^2 - p_1^2 - p_2^2}{2\ell_1\ell_2}\right] \in D,$$

then we see that $\mathbf{x}_e$ is an equilibrium point of system (1).

For $\mathbf{x}(t) \ne \mathbf{x}_e$, we further define $\xi(t)$ as the angular position of the target center relative to the end-effector at time *t*. The angle $\xi(t)$ is defined implicitly

$$\tan\xi(t) = \begin{cases} \dfrac{p_2 - \ell_1\sin\theta_1 + \ell_2\sin(\theta_1 + \theta_2)}{p_1 - \ell_1\cos\theta_1 + \ell_2\cos(\theta_1 + \theta_2)}, & \text{if } \mathbf{x}(t) \ne \mathbf{x}_e; \\ \tan\xi(t-1), & \text{if } \mathbf{x}(t) = \mathbf{x}_e. \end{cases}$$

B. Mechanical Singularities

In reality, the motion of the end-effector is restricted in the sense that the second link of the revolute 2-link manipulator can neither be fully stretched nor it can be folded back [5]. That is the angle θ_2 is restricted as

$$0 \le |\theta_2| \le \pi$$

In order to observe this restriction, we treat the line passing through the points $(0,0)$ and $(\ell_1\cos\theta_1, \ell_1\sin\theta_1)$ as an artificial obstacle for the end-effector. The end-effector can avoid this line by simply avoiding the closest point on the line [12]. The closest point on the line measured form the position of the end-effector is given by $(\lambda^*\ell_1\cos\theta_1, \lambda^*\ell_1\sin\theta_1)$ where

$$\lambda^* = 1 - \frac{\ell_2\cos\theta_2}{\ell_1}$$

This closest point on the line is treated as an artificial obstacle and thus it will simply be avoided by carefully defining the controllers.

The distance from the end-effector to the closest point on the line is given by $R_0 = \ell_2|\sin\theta_2|$. Let $d_{\max} > 0$ be a pre-determined scalar and define

$$f_0 = \left[\ell_1\cos\theta_1 - \ell_2\cos(\theta_1 + \theta_2)\right]p_2 - \left[\ell_1\sin\theta_1 - \ell_2\sin(\theta_1 + \theta_2)\right]p_1$$

$$\alpha_0 = \begin{cases} 0, & \text{if } R_0 \ge d_{\max} \\ d_{\max} - R_0 & \text{if } R_0 < d_{\max} \end{cases}$$

$$\beta_0 = \begin{cases} 1, & \text{if } f_0 \le 0 \\ -1 & \text{if } f_0 > 0 \end{cases}$$

$$\varepsilon_0 = \tan^{-1}\left(\frac{\alpha_0\beta_0}{R_0}\right)$$

For the avoidance of the line obstacle, we propose the following form of the controllers u_1 and u_2:

$$\left.\begin{aligned} u_1 &= v\cos(\xi + \varepsilon_0), \\ u_2 &= v\sin(\xi + \varepsilon_0)\ . \end{aligned}\right\} \tag{2}$$

Remark: With the form of controllers given in (2), we see that as the end-effector comes closer to the line obstacle, then the quantity R_0 will decrease. This will increase $|\varepsilon_0|$ since R_0 appears in the denominator. Hence an increase in $|\varepsilon_0|$ will force the end-effector to move away from the obstacle.

Substituting for *v* and ξ into (2), we obtain

$$\left.\begin{aligned} u_1 &= \frac{|v_0|\left[(p_1 - x)R_0 - (p_2 - y)\alpha_0\beta_0\right]}{\left\|(p_1 - x(0), p_2 - y(0))\right\|\sqrt{\alpha_0^2 + R_0^2}}, \\ u_2 &= \frac{|v_0|\left[(p_2 - y)R_0 - (p_1 - x)\alpha_0\beta_0\right]}{\left\|(p_1 - x(0), p_2 - y(0))\right\|\sqrt{\alpha_0^2 + R_0^2}}. \end{aligned}\right\} \tag{3}$$

The controllers are bounded and continuous at every point over the domain

$$D = \left\{\mathbf{x} \in \mathbf{R}^2 : (x(0),\ y(0) \ne (p_1,\ p_2) \cap R_0 > 0\right\}.$$

Simulation 1: The computer is used to numerically integrate system (1) to obtain the solution $(r(t), \theta(t))$ and plot the trajectory of the end-effector, which converges to the target position (p_1, p_2) and stays there as $t \to +\infty$. For our simulation, Table I gives the values of the different parameters, and Fig. 2 gives the trajectory of the arm. It was noticed that due to the unique forms of the controllers, the arm slowed down as the end-effector approached its target.

TABLE I
VALUES OF DIFFERENT PARAMETERS USED IN THE SIMULATION

Initial Configuration	
Initial position	(6,7) m
Initial velocity	1 m/s
Final Configuration	
Target position	(8,-6) m
Radius of the target	0.2 m
Other Parameters	
Workspace dimensions	$0 \le z_1 \le 10$, $0 \le z_2 \le 10$
Robot dimensions	$\ell_1 = 6\,\text{m},\ \ell_2 = 6\,\text{m}$
Sensing region	$d_{\max} = 1\,\text{m}$

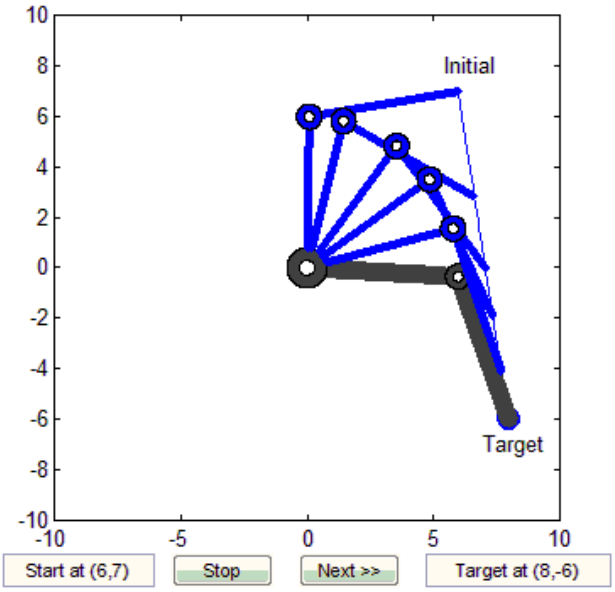

Fig. 2 The motion of the arm from an initial configuration to a final configuration as determined by the controllers in (3)

IV. MOTION CONTROL IN THE PRESENCE OF OBSTACLES

Let us fix $q > 0$ obstacle within the boundaries of the workspace. We assume that the lth obstacle is circular with center given as (o_{l1}, o_{l2}) and radius r_l. The lth stationary obstacle is defined below.

Definition 1: The lth stationary obstacle is a disk-shaped obstacle with center (o_{l1}, o_{l2}) and radius r_l. Precisely, the lth stationary obstacle is the set

$$FO_l = \left\{(z_1,\ z_2) \in \mathbf{R}^2 : (z_1 - o_{l1})^2 + (z_2 - o_{l2})^2 \le ro_l^2\right\}$$

for $l = 1,2,\ldots,q$.

Assumption 1: The lth obstacle must be placed the robot's workspace in such a way that its center (o_{l1}, o_{l2}) must satisfy the inequality $o_{l1}^2 + o_{l2}^2 > (\ell_1 + ro_l)^2$.

Remark: *Assumption 1 is justified since link 1 needs to move freely in the circular region*

$$C = \left\{(z_1, z_2) \in \mathbf{R}^2 : z_1^2 + z_2^2 \le \ell_1^2\right\}$$

This also ensures that link 1 is not trapped in between two obstacles.

Thus for the end-effector to converge to its target safely, we need the entire link 2 to avoid a fixed obstacle. For this, we utilize the minimum distance technique (MDT) proposed by Sharma in [12], where the author calculated the minimum distance from a robot to a line segment and the resultant closest point was avoided by the robot. However, in our case, we want the line segment (link 2) to avoid a fixed obstacle.

Adopting the concept from [12], we find the point on link 2 that will be closest to a fixed obstacle. Let (x_l^*, y_l^*) be a point on link 2 that is closest to the lth fixed obstacle, then it can be shown that

$$x_l^* = \ell_1 \cos\theta_1 + (\lambda_l - 1)\ell_2 \cos(\theta_1 + \theta_2),$$
$$y_l^* = \ell_1 \sin\theta_1 + (\lambda_l - 1)\ell_2 \sin(\theta_1 + \theta_2),$$

where

$$\lambda_l = 1 + \frac{1}{\ell_2^2}\Big[(o_{l1} - \ell_1\cos\theta_1)\ell_2\cos(\theta_1 + \theta_2) + (o_{l2} - \ell_1\sin\theta_1)\ell_2\sin(\theta_1 + \theta_2)\Big].$$

Note that $\lambda_l \in [0, 1]$. If $\lambda_l > 1$, then we let $\lambda_l = 1$ and if $\lambda_l < 0$, then we let $\lambda_l = 0$. Otherwise we accept the value of λ_l between 0 and 1.

Now, let $R_l = \sqrt{(x_l^* - o_{l1})^2 + (y_l^* - o_{l2})^2} - ro_l$ be the distance from the center of the lth obstacle to the point (x_l^*, y_l^*) and define

$$f_l = \left[\ell_1 \sin\theta_1 - \ell_2 \sin(\theta_1 + \theta_2)\right] p_1 - \left[\ell_1 \cos\theta_1 - \ell_2 \cos(\theta_1 + \theta_2)\right] p_2$$

$$\alpha_l = \begin{cases} 0, & \text{if } R_l \ge d_{\max} \\ d_{\max} - R_l & \text{if } R_l < d_{\max} \end{cases}$$

$$\beta_l = \begin{cases} 1, & \text{if } f_l \le 0 \\ -1 & \text{if } f_l > 0 \end{cases}$$

$$\varepsilon_l = \frac{\alpha_l \beta_l}{R_l}$$

for $l = 1,2,\ldots,q$. In order for the point (x_l^*, y_l^*) to avoid the fixed obstacles and for the end-effector to avoid the artificial obstacle, we modify the controllers u_1 and u_2 given in (2) as:

$$\left.\begin{aligned} u_1 &= v\cos\left(\xi + \tan^{-1}\left(\sum_{l=1}^{q}\varepsilon_l\right)\right), \\ u_2 &= v\sin\left(\xi + \tan^{-1}\left(\sum_{l=1}^{q}\varepsilon_l\right)\right). \end{aligned}\right\} \quad (4)$$

Substituting for v and ξ into (4), we obtain

$$\left.\begin{aligned} u_1 &= \frac{|v_0|\left[(p_1-x)-(p_2-y)\sum_{l=0}^{q}\frac{\alpha_l\beta_l}{R_l}\right]}{\|(p_1-x(0),\, p_2-y(0))\|\sqrt{1+\left(\sum_{l=0}^{q}\frac{\alpha_l\beta_l}{R_l}\right)^2}}, \\ u_2 &= \frac{|v_0|\left[(p_2-y)-(p_1-x)\sum_{l=0}^{q}\frac{\alpha_l\beta_l}{R_l}\right]}{\|(p_1-x(0),\, p_2-y(0))\|\sqrt{1+\left(\sum_{l=0}^{q}\frac{\alpha_l\beta_l}{R_l}\right)^2}}. \end{aligned}\right\} \quad (5)$$

The controllers are bounded and continuous at every point over the domain

$$D = \{\mathbf{x}\in\mathbf{R}^2 : (x(0),\, y(0)) \neq (p_1,\, p_2) \cap R_l > 0 \ \text{ for } l = 1,2,\ldots,q\}$$

Simulation 2: To illustrate the effectiveness of our proposed controller, we have generated the trajectories of the revolute arm from some initial configuration to the final configuration. This is shown in Example 1 and Example 2 below.

Example 1: The robot arm encounters a fixed obstacle which it has to avoid along its journey to the target. The values of the different parameters used in the simulation (if different from Simulation 1) are given in Table II.

TABLE II
VALUES OF DIFFERENT PARAMETERS USED IN THE SIMULATION.

Initial Configuration	
Initial position	(8,-5) m
Initial velocity	1 m/s
Final Configuration	
Target position	(5,8) m
Radius of the target	0.2 m
Obstacle Parameters	
Obstacle center	(6,4) m
Obstacle radius	0.8 m

Fig. 3 shows the trajectory of the robot arm from the initial to the final states. Note that, given appropriate initial conditions, the robot arm avoided the obstacle and the end-effector converged to its designated target.

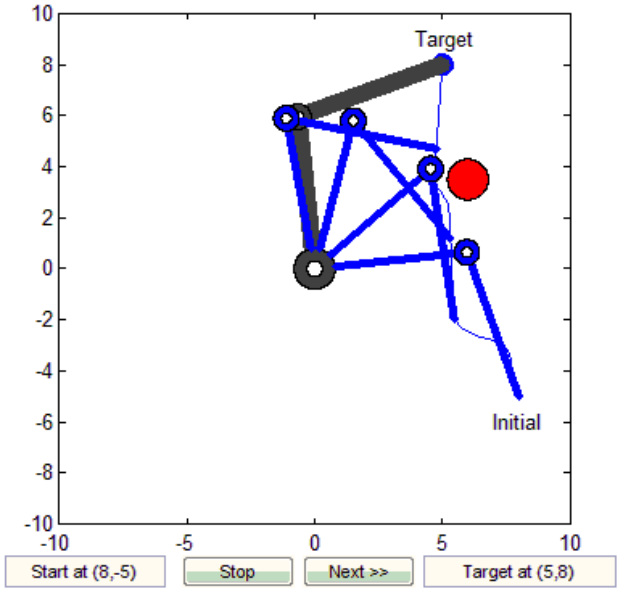

Fig. 3 Trajectory of the end-effector of the robot arm with initial position (8,−5) and the target placed at (5, 8)

Example 2: Here, we illustrate the effectiveness of the control laws among several fixed obstacles. The initial and final position of the end-effector is shown in Fig. 4 while the positions and sizes of the fixed obstacles were generated randomly.

Fig. 4 shows the convergence of the end-effector to its designated target in the presence of multiple obstacle in the workspace. The robot arm converges nicely to the target whilst avoiding the obstacles along its path.

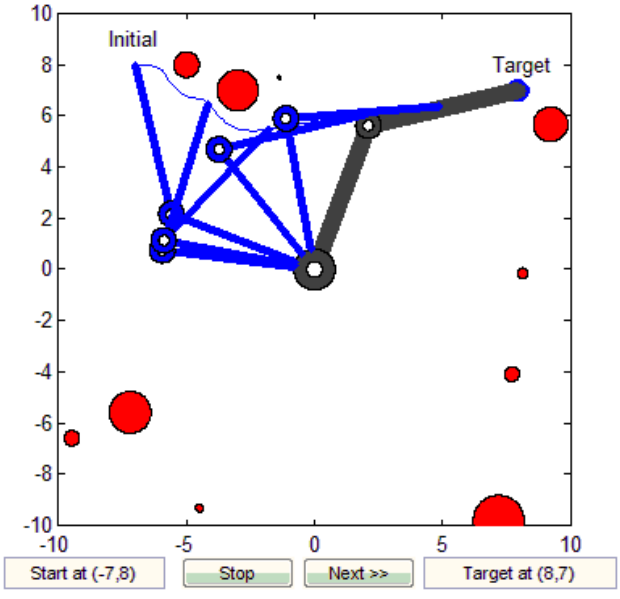

Fig. 4 Trajectory of the end-effector of the robot arm with initial position (−7, 8) and the target placed at (8, 7)

V. Stability Analysis

The controllers u_1 and u_2 defined by (5) are bounded and continuous at every point on the neighborhood of the equilibrium point, and lie in neighborhood of the equilibrium point for all $t \geq 0$. Given this, the discussions above yield the stability of system (1):

Theorem 1: If the initial position $(x(0), y(0))$ of the end-effector does not intersect with the target position and the Assumption 1 holds, then the point $\mathbf{x}_e$ is an asymptotic stable equilibrium point of system (1).

Proof. Consider the Lyapunov function

$$L(\mathbf{x}) = \frac{1}{2}\left\|(p_1 - x, p_2 - y)\right\|^2$$

which is defined, continuous and positive over the domain

$$D = \left\{\mathbf{x} \in \mathbf{R}^2 : (x(0),\ y(0) \neq (p_1,\ p_2) \cap R_l > 0 \ \text{ for } l = 1, 2, \ldots, q\right\}$$

It is clear that $L(\mathbf{x})$ has continuous first partial derivatives in the neighborhood D of the equilibrium point $\mathbf{x}_e$ of system (1). Moreover, in the region D, we see that $L(\mathbf{x}_e) = 0$ and $L(\mathbf{x}) > 0$ for all $\mathbf{x} \neq \mathbf{x}_e$. Now, the time-derivative of $L(\mathbf{x})$ along a trajectory of system (1) is given by

$$\dot{L}(\mathbf{x}) = -\frac{v\left\|(p_1 - x, p_2 - y)\right\|}{\sqrt{1 + \left(\sum_{l=0}^{q} \frac{\alpha_l \beta_l}{R_l}\right)^2}},$$

again, it is clear that in the region D, $\dot{L}(\mathbf{x}_e) = 0$ and $\dot{L}(\mathbf{x}) < 0$ for all $\mathbf{x} \neq \mathbf{x}_e$. Hence it can be concluded that the point $\mathbf{x}_e$ is an asymptotic stable equilibrium point of system (1).

VI. Conclusion

The paper presents a simple approach for solving the motion control of a 2-link revolute manipulator. A tailored target convergence and obstacle avoidance scheme is developed and the control laws are designed to move the end-effector towards its goal and avoid any fixed obstacles along its path.

The control laws proposed in this paper also ensure an asymptotic stability of the system. This has been proven using the Direct Method of Lyapunov. The stabilization property of the system has aslo been verified numerically via the computer simulations.

Future work will consider the motion control of 3-dimensional manipulator arms, mobile car-like robots and mobile manipulator arms.

References

[1] S. Tejomurtula and S. Kak, "Inverse kinematics in robotics using neural networks," *Information Science*, vol. 116, no. 2-4, pp. 147–164, January 1999.

[2] J. Vanualailai, S. Nakagiri, and J. Ha, "A solution to twodimension findpath problem," *Dynamics and Stability of Systems*, vol. 13, pp. 373–401, 1998.

[3] W. Meyer, "Moving a planar robot arm," *MAA Notes: The Mathematical Association of America*, vol. , no. 29, pp. 180–192, 1993.

[4] E. Sacks, "Path planning for planar articulated robots using configuration spaces and compliant motion," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 3, pp. 381–390, 2003.

[5] J. Vanualailai and B. Sharma, "Moving a robot arm: an interesting application of the Direct Method of Lyapunov," *CUBO: A Mathematical Journal*, vol. 6, no. 3, pp. 131–144, 2004.

[6] J. Vanualailai, B. Sharma, and A. Ali, "Lyapunov-based kinematic path planning for a 3-link planar robot arm in a structured environment," *Global Journal of Pure and Applied Mathematics*, vol. 3, no. 2, pp. 175–190, 2007.

[7] B. Sharma, A. Prasad, and J. Vanualailai, "A collision-free algorithm of a point-mass robot using neural networks," *Journal of Artificial Intelligence*, vol. 3, no. 1, pp. 49–55, 2012.

[8] T.M. Martinetz, H.J. Ritter, and K.J. Schulten, "Threedimensional neural net for learning visomotor coordination of a robot arm," *EEE Trans. on Neural Networks*, vol. 1, no. 1, 1990.

[9] G. Josin, D. Charney, and D. White, "A neural-representation of an unknown inverse kinematic transformation," in *First European Conference on Neural Networks, Paris*, Paris, France, June 988.

[10] A. Guez and Z. Ahmad, "Solution to the inverse kinematics problem in robotics by neural networks," in *Proc. IEEE International Conference on Neural Networks*, BSan Diego, USA, 1988, pp. 617–624.

[11] H. Jack, D.M.A. Lee, R.O. Buchal, and W.H. ElMaraghy, "Neural networks and the inverse kinematics problem," *The Journal of Intelligent Manufacturing*, vol. 4, pp. 43–46, 1993.

[12] B. Sharma. *New Directions in the Applications of the Lyapunov-based Control Scheme to the Findpath Problem*. PhD thesis, University of the South Pacific, Suva, Fiji Islands, July 2008. PhD Dissertation.