Electronic Journal: Southwest Journal of Pure and Applied Mathematics
Internet: http://rattler.cameron.edu/swjpam.html
ISSN 1083-0464
Issue 1, July 2003, pp. 43–55.
Submitted: August 4, 2002. Published: July 1 2003.

NONLINEAR CONTROLLERS FOR A PLANAR ROBOT ARM IN A CONSTRAINED ENVIRONMENT

Jito Vanualailai and Bibhya Sharma

Abstract. In this article, we explore the fundamentals of an emerging technique applicable, at least, in principle, to robot navigation, or motion planning. Termed the Second Method of Lyapunov, it is currently a powerful mathematical technique used to study the qualitative behaviour of natural or man-made systems that could be modeled, in an approximate way, by differential equations. We review the Lyapunov method and then in a simple and direct way, we use it to propose a theoretical technique to control the motion of a planar arm in a constrained environment. The controllers are mathematical entities which are nonlinear in nature. Computer simulations are used to illustrate the effectiveness of the proposed controllers.

A.M.S. (MOS) Subject Classification Codes. 34D20, 37B25, 70E60

Key Words and Phrases. Lyapunov Stability, Lyapunov Function, Robot Dynamics and Control, Findpath Problem, Motion Planning

1. Introduction

In [1], Meyer cleverly brings within the grasp of mathematicians the interesting mathematical and physical concepts associated with planar robot arms or manipulators.

An interesting two-dimensional geometric problem arising from Meyer's work is as follows: *Given a robot and a description of its working space or* workspace, *which could be cluttered with stationary or mobile solid objects or obstacles, propose a collision-free path that will lead the robot from the desired starting point to the desired location or target.* This is known as the *findpath problem*, the quest for the solutions of which is, at present, one of the most interesting theoretical undertakings in robotics research. For a review of various findpath schemes, see, for example, Sheu and Xue [2]. A more recent review is by Kumar et al. [3].

Meyer proposed a findpath scheme based on the velocities of the various components of the arm. In this article, we build on the work of Meyer by applying

Department of Mathematics and Computing Science, University of South Pacific, Suva, FIJI
E-mail Address: vanualailai@usp.ac.fj

a relatively new scheme that is based on acceleration. The scheme gives us the advantage of including constraints that might affect the operation of the arm. First proposed in 1990 by Stonier [4] and then elaborated on in 1995 and 1998 by Vanualailai et al. [5],[6], with follow-ups by Ha and Shim in 2000 [7] and 2001 [8], the scheme is based on elementary differentiation, and thus, is within the grasp of the college sophomore. However, at the heart of the control scheme is a powerful mathematical technique, called the *Direct Method of Lyapunov* [9] or simply, the method of Lyapunov, that requires some understanding of the nature of solutions of autonomous systems of first-order ordinary differential equations.

We start by re-looking at the equations used by Meyer to control the arm via velocity.

2. Velocity Control

Figure 1 shows a schematic represenation of the simplified robot arm considered by Meyer.

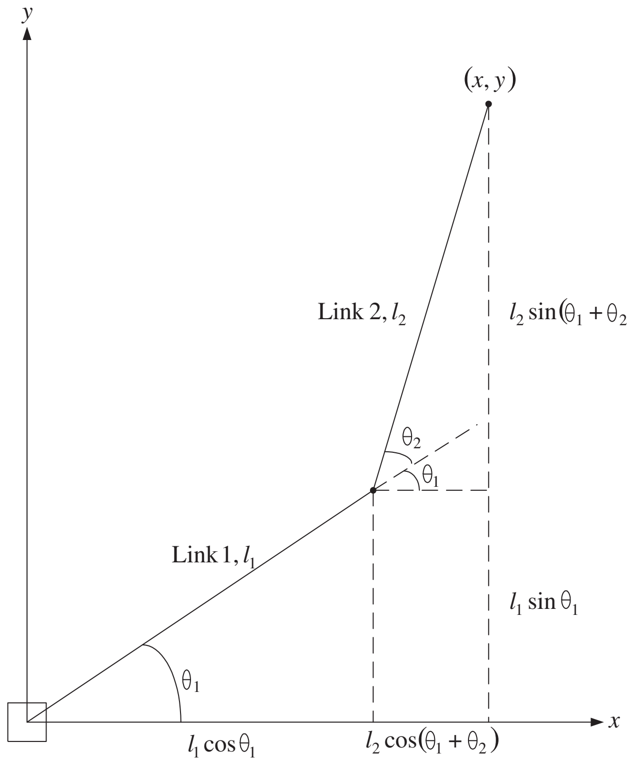

Figure 1: Schematic representation of the planar robot arm.

The position $(x(t), y(t))$ of the end-effector at time t is given by the forward kinematic equations

$$\begin{cases} x(t) = l_1 \cos\theta_1(t) + l_2 \cos(\theta_1(t) + \theta_2(t))\,, \\ y(t) = l_1 \sin\theta_1(t) + l_2 \sin(\theta_1(t) + \theta_2(t))\,. \end{cases} \tag{1}$$

The instantaneous velocity $(x'(t), y'(t))$ can be obtained from the equations

$$(2) \qquad \begin{cases} x'(t) = -y\theta_1'(t) - l_2 \sin(\theta_1(t) + \theta_2(t))\theta_2'(t)\,, \\ y'(t) = x\theta_1'(t) + l_2 \cos(\theta_1(t) + \theta_2(t))\theta_2'(t)\,. \end{cases}$$

One way to guide the end-effector is to control the velocities, (x', y') and θ_1' and θ_2'. If we want to maintain a constant velocity, say, $(x'(t), y'(t)) = (v_1, v_2)$ for all time $t \geq 0$, then our *velocity control scheme* is as follows:

Step 1. For the end-effector, fix the desired velocity, say, $(x'(t), y'(t)) = (v_1, v_2)$ at all time $t \geq 0$.

Step 2. Record the current position of the end-effector, say $(x, y) = (p_1, p_2)$, with the left-hand-side of (1) and then derive the angular positions θ_1 and θ_2.

Step 3. Insert $(x, y) = (p_1, p_2)$, $(x', y') = (v_1, v_2)$, θ_1 and θ_2 into (2) and then derive the angular velocities θ_1' and θ_2'.

The computer controlling the robot then moves the joints at the angular velocities θ_1' and θ_2' while maintaining the constant velocity (v_1, v_2) of the end-effector at the point $(x, y) = (p_1, p_2)$.

Our scheme to be proposed essentially allows one to control the velocities $\theta_1'(t)$ and $\theta_2'(t)$ at every time t by controlling the instantaneous angular accelerations, $\theta_1''(t)$ and $\theta_2''(t)$. But, first, let us briefly look at the Direct Method of Lyapunov on which our scheme will be based.

3. The Direct Method of Lyapunov

In 1892, Lyapunov [9] established a qualitative method of analyzing solutions of autonomous systems of first-order differential equations of the form

$$(3) \qquad \begin{cases} x_1'(t) = f_1(x_1(t), x_2(t), \ldots, x_n(t))\,, \\ \qquad \vdots \\ x_n'(t) = f_n(x_1(t), x_2(t), \ldots, x_n(t))\,, \end{cases}$$

with the known solution, $(x_1(t_0), \ldots, x_n(t_0))$, at some time $t = t_0 \geq 0$. This known solution is referred to as the *initial state* of system (3). The basic assumption is that the solutions of (3) exist at any given time t and are unique, with each solution depending on the initial state and continuable to $+\infty$. That is, we have unique and continuous solutions of the form $x_1(t) = x_1(t; t_0, x_1(t_0))$, $\cdots$, $x_n(t) = x_n(t; t_0, x_n(t_0))$, which may be regarded either as a curve in the space of the $n+1$ variables $x_1, x_2, \ldots, t$ or as a curve in the space of n variables $x_1, x_2, \ldots, x_n$ with t regarded as a parameter. In the latter case (which will be our principal concern), the curve is called a *trajectory* in the *phase space* – the space of n variables, $x_1, x_2, \ldots, x_n$. If at a particular point $(p_1, \ldots, p_n)$ in the phase space, we have that $(f_1(p_1, \ldots, p_n), \ldots, f_n(p_1, \ldots, p_n)) \equiv (0, \ldots, 0)$ for all time $t \geq 0$, then we say that the point is an *equilibrium point* of system (3). We will henceforth assume that at least one such point exists.

A question that the Lyapunov method can answer is as follows: *Let $(p_1, \ldots, p_n)$ be an equilibrium point of system (3) in a region of the phase space. If we start from the initial state within the region and close to the equilibrium point, does the trajectory $(x_1(t), \ldots, x_n(t))$ remain within the region and close to $(p_1, \ldots, p_n)$ for all time t?*

If the answer is yes, then we say that the point $(p_1, \ldots, p_n)$ is *stable*.

Precisely, if we let $\mathbf{x}(t) = (x_1(t), \ldots, x_n(t))$, and $\mathbf{x}^* = (p_1, \ldots, p_n)$, then $\mathbf{x}^*$ is a stable equilibrium point of (3) if for all $t_0 \geq 0$ and $\epsilon > 0$, there is $\delta(t_0, \epsilon) > 0$ such that

$$\|\mathbf{x}(t_0) - \mathbf{x}^*\| < \delta(t_0, \epsilon)\,,$$

implies

$$\|\mathbf{x}(t; t_0, \mathbf{x}(t_0)) - \mathbf{x}^*\| < \epsilon\,,$$

for all $t \geq t_0$. The beautiful aspect of the Lyapunov method, as revealed in the theorem below, is that we do not need the explicit form of the solutions to establish stability.

Theorem 1 (Lyapunov [9]**).** *Let $\mathbf{x}^*$ be an equilibrium point of system (3). If, in an open neighborhood $\mathbb{D}$ of the equilibrium point, there exists a real scalar function V such that* (a) *$V(\mathbf{x})$ is continuous on $\mathbb{D}$ and has continuous first partial derivatives with respect to $\mathbf{x}$,* (b) *$V(\mathbf{x}^*) = 0$, $\mathbf{x}^* \in \mathbb{D}$,* (c) *$V(\mathbf{x}) > 0$ for all $\mathbf{x} \in \mathbb{D}$, $\mathbf{x} \neq \mathbf{x}^*$, and* (d) *the time-derivative of V with respect to system (3) is*

$$\frac{d}{dt}\,[V(\mathbf{x})]_{(3)} = \sum_{i=1}^{n} \frac{\partial}{\partial x_i}\,[V(\mathbf{x})]\, f_i(\mathbf{x}) \leq 0$$

for all $\mathbf{x} \in \mathbb{D}$, then $\mathbf{x}^*$ is a stable equilibrium point of system (3).

The scalar function V in Theorem 1 is called a *Lyapunov function for system (3) on* $\mathbb{D}$. To understand therefore the behaviour of trajectories in the neighbourhood of the equilibrium hinges only on discovering a Lyapunov function without the need to find the explicit solutions, which, particularly for nonlinear systems, are usually difficult to formulate in closed form. This single aspect of the Lyapunov method makes it a powerful tool in the stability analysis of nonlinear systems. The Lyapunov method is now a critical component in specializations such as control engineering, power system engineering, robotics, neural networks, chaos and economics, to name but a few. A good review of the method, including recent advances and several applications can be found in Sastry [10].

For our application, where we desire the convergence of trajectories to equilibrium points, Theorem 1 tells us that stability ensures only boundedness of solutions in a neighbourhood of $\mathbf{x}^*$. However, it is known that if in addition $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_n(\mathbf{x})$ is bounded for $\mathbf{x}$ bounded, then whenever $d[V(\mathbf{x})]/dt < 0$ for $\mathbf{x} \neq \mathbf{x}^*$ and $d[V(\mathbf{x}^*)]/dt = 0$ in Theorem 1, the equilibrium point $\mathbf{x}^*$ is not only stable, but also attracts trajectories to it. That is, $\mathbf{x}^*$ is *asymptotically stable*, which is clearly more desirable than stability from a practical point of view. In this paper, to simplify our discussion, it is suffice to guarantee stability. The computer is then used to find initial conditions that guarantee attractiveness.

4. Acceleration Control

As mentioned earlier, we intend to control the end-effector by controlling the angular accelerations, $\theta_1''(t)$ and $\theta_2''(t)$. If we are to achieve this via the method of Lyapunov, then, first, we need to have a system of first-order differential equations describing the motion of the planar robot arm. Thus, if we let

$$\begin{aligned}
x_1(t) &= \text{ the } x\text{-component of the position of the end-effector,}\\
x_2(t) &= \text{ the } y\text{-component of the position of the end-effector,}\\
x_3(t) &= \text{ the angular position, } \theta_1(t), \text{ of Link 1 ,}\\
x_4(t) &= \text{ the angular position, } \theta_2(t), \text{ of Link 2 ,}\\
x_5(t) &= \text{ the angular velocity, } \theta_1'(t), \text{ of Link 1 ,}\\
x_6(t) &= \text{ the angular velocity, } \theta_2'(t), \text{ of Link 2 ,}\\
u_1(t) &= \text{ the angular acceleration, } \theta_1''(t), \text{ of Link 1 ,}\\
u_2(t) &= \text{ the angular acceleration, } \theta_2''(t), \text{ of Link 2,}
\end{aligned}$$

at time t, then using equations (2), we have (on suppressing t),

$$\begin{cases} x_1' = -x_2x_5 - l_2\sin(x_3+x_4)x_6\,, \quad x_3' = x_5\,, \quad x_5' = u_1\,, \\ x_2' = x_1x_5 + l_2\cos(x_3+x_4)x_6\,, \quad x_4' = x_6\,, \quad x_6' = u_2\,. \end{cases} \tag{4}$$

Before we actually start looking for a Lyapunov function for system (4), we may want to look for any difficulty inherent in the basic geometric structure of the robot. As shown by Meyer, there is indeed a problem: there exists at least one direction in which the end-effector cannot be moved no matter how we choose the joint velocities θ_1' and θ_2'. This arises when $\theta_2 = 0, \pi$. In our scheme, we want to avoid these *singular configurations*. We may also want to take into account other constraints in the system. These include, for example, the allowable space or area to work in, or the allowable velocity or acceleration of the end-effector. Hence, our *acceleration control scheme* via the Lyapunov method can be stated as follows:

Step 1. Let the final destination of the end-effector be $(x, y) = (p_1, p_2)$, achieved at the angular positions $x_3 = p_3$ and $x_4 = p_4$ and let us call $(p_1, p_2, p_3, p_4, 0, 0)$ the *target*.

Step 2. Identify system constraints (these include singular configurations) and appropriately construct mathematical functions to model them. Call the constraints *antitargets* or *obstacles* to be avoided.

Step 3. Construct a Lyapunov function such that u_1 and u_2 render system (4) stable, meaning that $(p_1, p_2, p_3, p_4, 0, 0)$ is an equilibrium point of system (4) and that the trajectory $(x_1, x_2, x_3, x_4, x_5, x_6)$ starts and remains, for all time $t \geq 0$, near the target $(p_1, p_2, p_3, p_4, 0, 0)$ while it avoids stationary and/or mobile obstacles in the workspace.

In Step 1, intuitively, we want to have a kind of a yardstick that measures, at time t, the position of the end-effector from the point $(x, y) = (p_1, p_2)$ and the rate at which it approaches or moves away from (p_1, p_2). The following choice of probable functions accomplishes this (on suppressing t),

$$V_0(x_1, \ldots, x_6) = \frac{1}{2}\left[(x_1-p_1)^2 + (x_2-p_2)^2) + x_5^2 + x_6^2\right]\,,$$

noting that

$$V_0(p_1, p_2, p_3, p_4, 0, 0) = 0\,,$$

and

$$V_0 > 0 \text{ for } (x_1, x_2, x_3, x_4, x_5, x_6) \neq (p_1, p_2, p_3, p_4, 0, 0)\,.$$

In Step 3, if a trajectory ever converges to the target, then it remains there for all time t since $V_0(p_1, p_2, p_3, p_4, 0, 0) = 0$. The use of V_0, together with appropriate collision avoidance schemes, should enable us to construct Lyapunov functions that ensure that trajectories remain near $(p_1, p_2, p_3, p_4, 0, 0)$ while avoiding obstacles.

In the next section, we consider several possible avoidance scenarios, in which we use the vector notations $\mathbf{x} = (x_1, \ldots, x_n)$ and $\mathbf{x}^* = (p_1, p_2, p_3, p_4, 0, 0)$.

5. Possible Avoidance Scenarios

5.1 Scenario 1: Avoidance of Singular Configurations.

The singular configurations occur when $\theta_2 = x_4 = 0$, $\theta_2 = x_4 = \pi$ in the anticlockwise direction or $\theta_2 = x_4 = -\pi$ in the clockwise direction. For the configurations, consider the functions

$$W_1(\mathbf{x}) = |x_4| \quad \text{and} \quad W_2(\mathbf{x}) = \pi - |x_4|\,,$$

for $x_4 \in (-\pi, 0) \cup (0, \pi)$. Now, let us, for the moment, consider the effect of the ratios

$$\frac{\beta_1}{W_1} \quad \text{and} \ \frac{\beta_2}{W_2}\,,$$

for some constants β_1 and β_2. If the robot arm approaches any one of the singular configurations, it is clear that one of the ratios will increase. Hence, if the ratios form parts of a Lyapunov function for system (4), intuitively the ratios will act as avoidance functions that repel the robot arm from the singular configurations. Indeed, by the mere fact that we will have a Lyapunov function, all trajectories will converge to a neighbourhood of the target, implying therefore that we cannot have the situation where $W_1 = W_2 = 0$. This is in turn means that we cannot have, at any time, the angular position of Link 2 as $p_4 = 0$, $p_4 = \pi$ or $p_4 = -\pi$. In this sense, we can think about singular configurations as mobile obstacles.

5.1.1 A Lyapunov Function.

Let us finally suggest the following function as a tentative Lyapunov function for system (4):

$$V(\mathbf{x}) = V_0(\mathbf{x}) + F(\mathbf{x}) \sum_{i=1}^{2} \frac{\beta_i}{W_i(\mathbf{x})}\,,$$

where β_1 and β_2 are positive constants and

$$F(\mathbf{x}) = \frac{1}{2}\left[(x_1 - p_1)^2 + (x_2 - p_2)^2\right]\,.$$

Clearly, V is continuous and positive on the domain

$$\begin{aligned}\mathbb{D}(V) &= \{\mathbf{x} \in \mathbb{R}^6 : W_1 > 0, W_2 > 0\} \\ &= \{\mathbf{x} \in \mathbb{R}^6 : -\pi < x_4 < 0 \text{ or } 0 < x_4 < \pi\}\,,\end{aligned}$$

and $V(\mathbf{x}^*) = 0$, $\mathbf{x}^* \in \mathbb{D}(V)$, so that we may take $\mathbb{D}(V)$ as the neighborhood $\mathbb{D}$ of the equilibrium point $\mathbf{x}^*$ in Lyapunov's Theorem 1.

Along a particular trajectory of system (4), we have (on suppressing t),

$$\begin{aligned}\frac{d}{dt}[V]_{(4)} = &\left\{[(x_2 - p_2)x_1 - (x_1 - p_1)x_2]\left(1 + \sum_{i=1}^{2}\frac{\beta_i}{W_i}\right) + u_1\right\}x_5 \\ &+ \left\{l_2\left[(x_2 - p_2)\cos(x_3 + x_4) - (x_1 - p_1)\sin(x_3 + x_4)\right]\left(1 + \sum_{i=1}^{2}\frac{\beta_i}{W_i}\right)\right. \\ &\left. -F\frac{x_4}{|x_4|}\left(\frac{\beta_1}{W_1^2} - \frac{\beta_2}{W_2^2}\right) + u_2\right\}x_6 \\ \stackrel{\text{def}}{=}\ &\{G_1(\mathbf{x}) + u_1\}x_5 + \{G_2(\mathbf{x}) + u_2\}x_6\,.\end{aligned}$$

For some numbers $\alpha_1 > 0$ and $\alpha_2 > 0$, define

$$-\alpha_1 x_5 = G_1 + u_1 \quad \text{and} \quad -\alpha_2 x_6 = G_2 + u_1\,.$$

Then

$$V_{(4)}{}' = -\alpha_1 x_5^2 - \alpha_2 x_6^2\,,$$

provided we have the *nonlinear controllers* as $u_1 = -\alpha_1 x_5 - G_1$ and $u_2 = -\alpha_2 x_6 - G_2$. Since $V_{(4)}{}'$ nonpositive and continuous for all $\mathbf{x} \in \mathbb{D}(V)$, by Theorem 1, V is a Lyapunov function for system (4) on $\mathbb{D}(V)$ establishing the stability of $\mathbf{x}^*$. Hence, the stable system that governs the motion of the arm is

$$(5) \qquad \begin{cases} x_1' = -x_2 x_5 - l_2\sin(x_3 + x_4)x_6\,, & x_3' = x_5\,, \quad x_5' = -\alpha_1 x_5 - G_1\,, \\ x_2' = x_1 x_5 + l_2\cos(x_3 + x_4)x_6\,, & x_4' = x_6\,, \quad x_6' = -\alpha_2 x_6 - G_2\,. \end{cases}$$

We see that an equilibrium point is $(p_1, p_2, p_3, p_4, 0, 0)$. Our Lyapunov function guarantees only stability of this equilibrium point. Clearly, a Lyapunov function which ensures asymptotic stability is more desirable. This is the most challenging part of the Lyapunov method, given the difficulty in constructing such a function for the pathpath problem. A new development in this direction is to guarantee stability and also prove that certain sets of initial conditions ensure asymptotic stability. This looks promising and at present is applicable to a point-mass system with one stationary obstacle and one stationary target. The interested reader can refer to the paper by Ha and Shim [8] for more details.

In this article, to simplify discussions, it is enough to consider a stable equilibrium point and then use a computer to help us decide which system parameters to use for a desirable run. These parameters are β_1 and β_2, which we shall call *control parameters*, and α_1 and α_2, which we shall call *convergence parameters*. It can be shown that the larger a control parameter is, the greater is the repulsion from the associated obstacle, and the larger a convergence parameter is, the slower is the trajectory [5].

5.1.2 A Computer Simulation.

The computer is used to numerically integrate system (5) to obtain the solution $(x_1, \ldots, x_6)$ and plot the points $(x_1(t), x_2(t))$ at time t in the x_1x_2-plane until the points converge to a neigborhood of (p_1, p_2) and stay there as $t \to \infty$. For our example, Table I gives the parameters, and Figure 2 gives the trajectory of the arm. Notice the slowing down of the arm as it approaches its final configuration. This could be explained in terms of the potential energy "cup" of the Liapunov function [5]. The arm reaches its final configuration in about 15 units of time. In Figure 2, the links are drawn every 1 unit of time.

Table I: A Scenario 1 example.

Lengths of Links	$l_1 = l_2 = 3$
Initial Conditions	$\mathbf{x} = \left(-3\sqrt{3}/2, 3/2, \pi/2, 2\pi/3, \pi/90, \pi/90\right)$
Planar Target	$(p_1, p_2) = (4, 4)$
Control Parameters	$\beta_1 = \beta_2 = 0.01$
Convergence Parameters	$\alpha_1 = 110$, $\alpha_2 = 20$

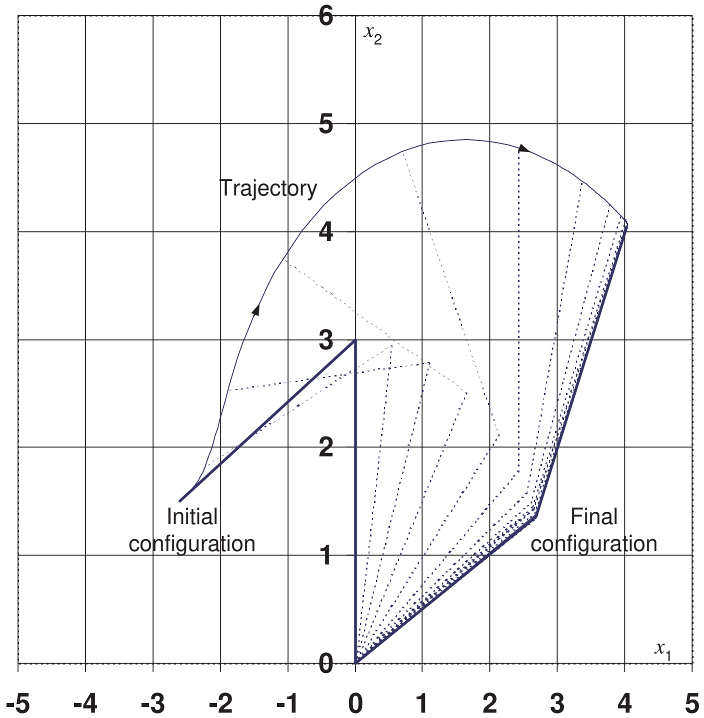

Figure 2: The motion of the arm from an initial configuration to a final configuration as determined by system (5).

5.2 Scenario 2: Movement in A Rectangle.

An interesting scene involves restricting the motion of the arm within a rectangular region in the first quadrant of the xy-plane. We place the bottom left corner of the rectangle at $(0, 0)$ where the base of the arm is located as shown in Figure 3, in which we call the constraints or obstacles the *left wall, right wall, floor* and *roof.* These are fixed obstacles. The mobile obstacles are the singular configurations.

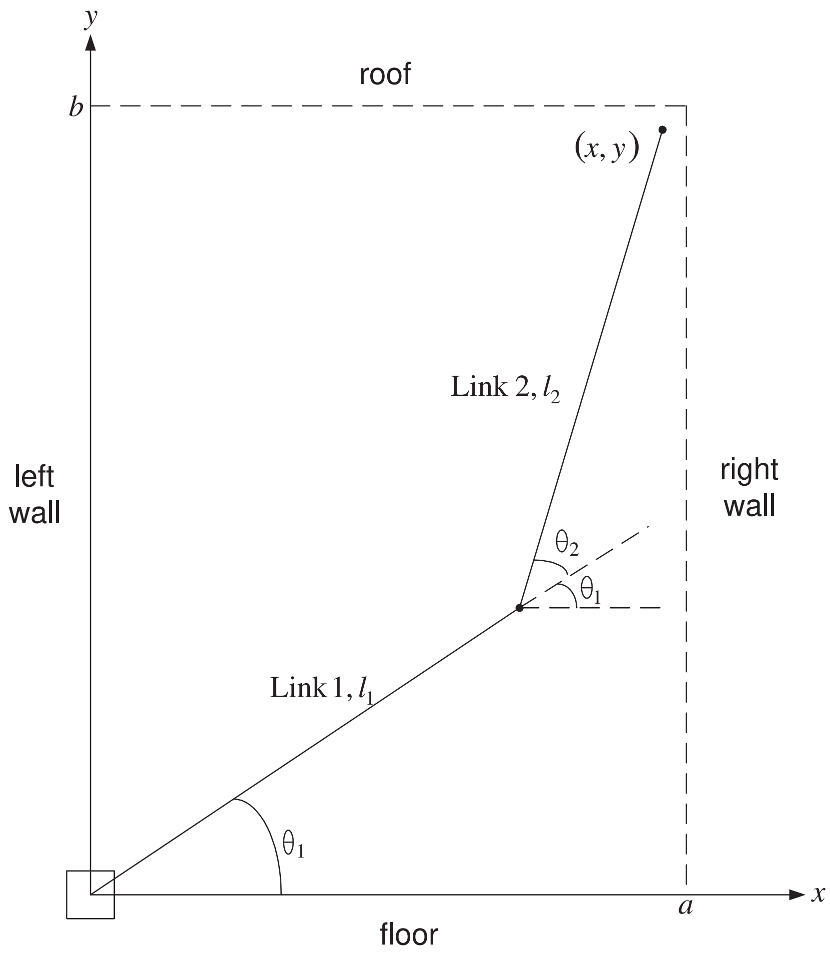

Figure 3: The planar robot arm in a constrained environment.

For obvious geometrical reasons, we construct the following avoidance functions that ensure that the end-effector (and, hence, Link 2), Link 1 and the joint do not collide with the fixed obstacles.

5.2.1 Avoidance Functions for the End-effector.
Table II gives the functions that will ensure that the end-effector does not collide with fixed obtacles.

Table II: Avoidance functions for the end-effector.

Obstacle	Avoidance Function	Sign
Left wall, $x_1 = 0$	$W_3 = x_1$	$W_3 > 0$ for all $x_1 \in (0, a)$
Right wall, $x_1 = a$	$W_4 = a - x_1$	$W_4 > 0$ for all $x_1 \in (0, a)$
Floor, $x_2 = 0$	$W_5 = x_2$	$W_5 > 0$ for all $x_2 \in (0, b)$
Roof, $x_2 = b$	$W_6 = b - x_2$	$W_6 > 0$ for all $x_2 \in (0, b)$

5.2.2 Avoidance Functions for Link 1.
Table III gives the functions that will ensure that Link 1 does not collide with fixed obtacles.

Table III: Avoidance functions for Link 1.

Obstacle	Avoidance Function	Sign
Floor, $x_3 = 0$	$W_7 = x_3$	$W_7 > 0$ for all $x_3 \in (0, \pi/2)$
Left wall, $x_3 = \pi/2$	$W_8 = \pi/2 - x_3$	$W_8 > 0$ for all $x_3 \in (0, \pi/2)$

5.2.3 Avoidance Functions for Elbow.
Table IV gives the functions that will ensure that the joint or elbow between Link 1 and Link 2 does not collide with fixed obtacles.

Table IV: Avoidance functions for elbow.

Obstacle	Avoidance Function	Sign
Roof, $x_2 = b$	$W_9 = b - l_1 \sin x_3$	$W_9 > 0$ in rectangle
Right wall, $x_1 = a$	$W_{10} = a - l_1 \cos x_3$	$W_{10} > 0$ in rectangle

5.2.4 A Lyapunov Function.
Here, we simply provide a Lyapunov function and the controllers, the derivation of which is similar to that in Scenario 1.

For constants $\beta_i > 0$, $i = 1, \ldots, 10$, we consider

$$V = V_0 + F \sum_{i=1}^{10} \frac{\beta_i}{W_i},$$

on the domain $\mathbb{D}(V) = \{\mathbf{x} \in \mathbb{R}^6 : W_i > 0, i = 1, \ldots, 10\}$, where V_0, W_1, W_2 and F are as those in Scenario 1. Then along a trajectory of system (4) in $\mathbb{D}(V)$, we have, for $\alpha_1, \alpha_2 > 0$, the time-derivative

$$V_{(4)}{}' = -\alpha_1 x_5^2 - \alpha_2 x_6^2 \,,$$

if we define the controllers as

$$u_1 = -\alpha_1 x_5 - G_3 \quad \text{and} \quad u_2 = -\alpha_2 x_6 - G_4 \,,$$

where

$$\begin{aligned} G_3 = G_3(\mathbf{x}) = {} & [(x_2 - p_2)x_1 - (x_1 - p_1)x_2] \left(1 + \sum_{i=1}^{10} \frac{\beta_i}{W_i}\right) \\ & - F \left[\left(-\frac{\beta_3}{W_3^2} + \frac{\beta_4}{W_4^2} \right) x_2 + \left(\frac{\beta_5}{W_5^2} - \frac{\beta_6}{W_6^2} \right) x_1 \right. \\ & \left. + \frac{\beta_7}{W_7^2} - \frac{\beta_8}{W_8^2} - \frac{\beta_9}{W_9^2} l_1 \cos x_3 + \frac{\beta_{10}}{W_{10}^2} l_1 \sin x_3 \right] \,, \end{aligned}$$

and

$$\begin{aligned} G_4 = G_4(\mathbf{x}) = {} & l_2 \left[(x_2 - p_2) \cos(x_3 + x_4) - (x_1 - p_1) \sin(x_3 + x_4) \right] \left(1 + \sum_{i=1}^{10} \frac{\beta_i}{W_i}\right) \\ & - F \left[\left(\frac{\beta_1}{W_1^2} - \frac{\beta_2}{W_2^2} \right) \frac{x_4}{|x_4|} + \left(-\frac{\beta_3}{W_3^2} + \frac{\beta_4}{W_4^2} \right) l_2 \sin(x_3 + x_4) \right. \\ & \left. + \left(\frac{\beta_5}{W_5^2} - \frac{\beta_6}{W_6^2} \right) l_2 \cos(x_3 + x_4) \right] \,. \end{aligned}$$

These render the system

$$(6) \qquad \begin{cases} x_1' = -x_2 x_5 - l_2 \sin(x_3 + x_4) x_6 \,, & x_3' = x_5 \,, \quad x_5' = -\alpha_1 x_5 - G_3 \,, \\ x_2' = x_1 x_5 + l_2 \cos(x_3 + x_4) x_6 \,, & x_4' = x_6 \,, \quad x_6' = -\alpha_2 x_6 - G_4 \,, \end{cases}$$

stable at the equilibrium point $\mathbf{x}^* = (p_1, p_2, p_3, p_4, 0, 0)$.

5.2.5 A Computer Simulation.

Table V shows the parameters used for a case in Scenario 2, and Figure 4 gives the motion of the arm. The arm reaches its final configuration in about 10 units of time.

Table V: A Scenario 2 example.

Lengths of Links	$l_1 = l_2 = 3$
Initial Conditions	$\mathbf{x} = (2.95, 0.52, 7\pi/18, -2\pi/3, \pi/90, \pi/90)$
Planar Target	$(p_1, p_2) = (2.2, 5.2)$
Control Parameters	$\beta_1 = \beta_2 = \beta_3 = \beta_4 = \beta_6 = \beta_8 = \beta_9 = \beta_{10} = 1\,,$ $\beta_5 = 10, \beta_7 = 5$
Convergence Parameters	$\alpha_1 = \alpha_2 = 100$
Right wall, Roof	$a = 3.2$, $b = 5.5$

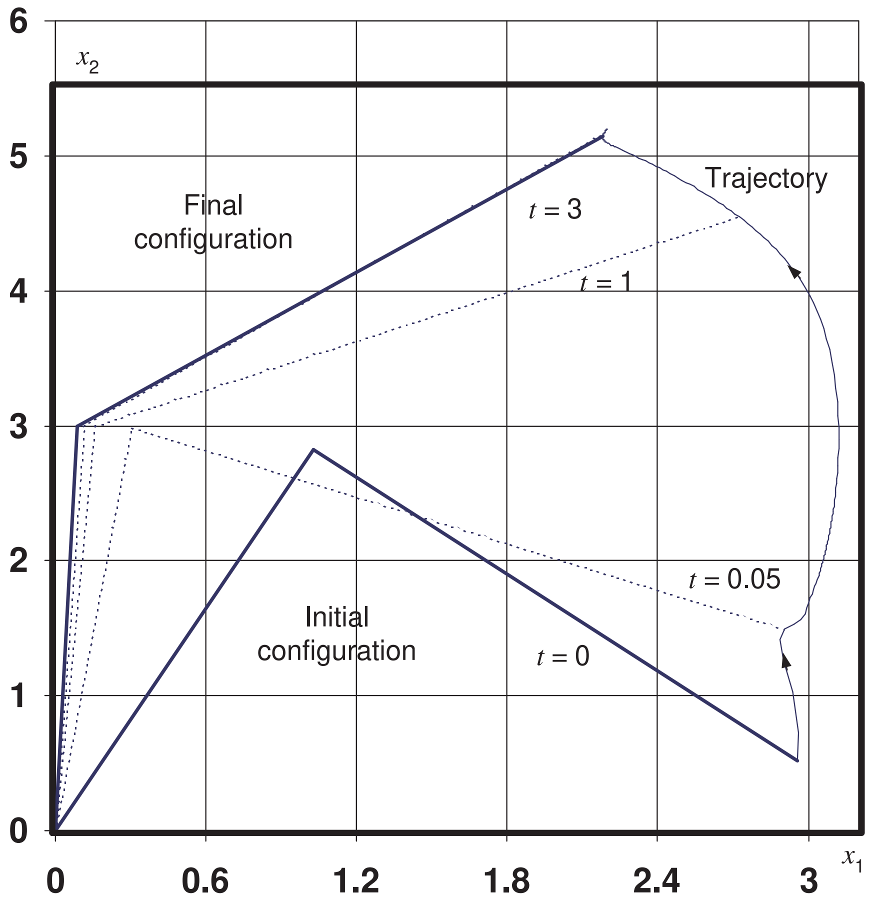

Figure 4: The motion of the robot arm from an initial configuration to a final configuration as determined by system (6).

6. Conclusion

A known drawback of the Lyapuov method, which is classified in the robotic literature as a *potential field* method, is the possibility of having collision-free paths leading not to the intended target but to "traps" outside of the target. These traps, like the target, are in fact local minima or points of zero kinetic and potential energy. As mentioned earlier, an encouraging development to deal with this problem was recently reported by Ha and Shim [8]. Another likely drawback of the Lyapunov method might well turn out to be the large amount of computation required for a multirobot environment. However, this may eventually cease to be a problem given the ever improving power of the digital computer. Hence, as far as the application of the Lyapunov method to the findpath problem is concerned, there are, at the present stage of development, some encouraging signs that could promote the wider use of the method.

References

1. Meyer, W., *Moving a Planar Robot Arm*, MAA Notes Number 29, The Mathematical Association of America (1993), 180–192.
2. Sheu, P. C. Y. and Xue, Q., *Intelligent Robotic Planning Systems*, World Scientific, Singapore, 1993.
3. Kumar, V., Žefran, M. and Ostrowski, J., *Motion Planning and Control of Robots*, Handbook of Industrial Robotics, J. Wiley and Sons, 1997.
4. Stonier, R. J., *Use of Liapunov Techniques for Collision-avoidance of Robot Arms*, Control and Dynamic Systems: Advances in Control Mechanics, Part 2 of 2, vol. 35, 1990, p. 185–214..
5. Vanualailai, J., Nakagiri, S. and Ha, J., *Collision Avoidance in a Two-point System via Liapunov's Second Method*, Mathematics and Computers in Simulation **39** (1995), 125–141.
6. Vanualailai, J., Ha, J. and Nakagiri, S., *A Solution to the Two-dimensional Findpath Problem*, Dynamics and Stability of Systems **13** (1998), 373–401.
7. Ha, J. and Shim, J., *A Findpath Problem in the Presence of Moving Obtacles*, Korean Journal of Computational and Applied Mathematics **7** (2000), 125–137.
8. ———, *An Asymptotic Stability Involving Collision and Avoidance*, Korean Journal of Computational and Applied Mathematics **8** (2001), 605–616.
9. Lyapunov, A. M., *The General Problem of the Stability of Motion*, 1892; International Journal of Control: Lyapunov Centenary Issue 5 (1992).
10. Sastry, S., *Nonlinear Systems: Analysis, Stability and Control*, Springer-Verlag, New York, 1999.