



Robotic Tasks Using Path Control: Two Case Studies *

JAVIER MORENO and RAFAEL KELLY

División de Física Aplicada, CICESE, Apdo. Postal 2615, Adm. 1, Ensenada, B.C., 22800 Mexico;
e-mail: rkelly@cicese.mx

(Received: 4 December 2001)

Abstract. Some robotic tasks usually achieved through motion control – trajectory tracking control – can be also well performed by resorting to “path control” philosophy. This is the case for applications where motion coordination among the robot joints is more important than joint tracking of a timed desired reference. This paper illustrates this concept by means of two academic case studies – theory and experiments – using a two degrees-of-freedom direct-drive revolute arm.

Key words: motion control, path control, LaSalle’s invariance principle, direct-drive robot.

1. Introduction

The dynamics of a serial n -link robot manipulator can be written as [13]:

$$M(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{f}(\dot{\mathbf{q}}) = \boldsymbol{\tau}, \quad (1)$$

where \mathbf{q} is the $n \times 1$ vector of joint displacements, $\dot{\mathbf{q}}$ is the $n \times 1$ vector of joint velocities, $\boldsymbol{\tau}$ is the $n \times 1$ vector of applied torques inputs, $M(\mathbf{q})$ is the $n \times n$ symmetric positive definite manipulator inertia matrix, $C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}$ is the $n \times 1$ vector of centripetal and Coriolis torques, $\mathbf{g}(\mathbf{q})$ is the $n \times 1$ vector of gravitational torques, and $\mathbf{f}(\dot{\mathbf{q}})$ is the $n \times 1$ vector of torques due to the friction which depends on the joint velocity.

The problem of motion control of manipulators – also termed trajectory tracking control – has been well studied in the past 20 years, see, e.g., [1, 5], and reference therein. Motion control is specified through trajectories depending on time, which encodes the requested task to be performed by the robot arm. This way of formulating the robot control problem has given good results for many practical applications. Notwithstanding, most of the robotic tasks can be well performed by motion coordination of the joints without regard of timing. This is the case of applications such as cutting, milling and deburring, where it is unnecessary to keep up with a desired timed trajectory. This class of tasks can be performed well using a path-based approach to robot control.

* Work partially supported by CONACyT, SNI and CYTED.

A control formulation which does not specify a nominal timed trajectory for encoding the requested task is the *event-based control* [14]. In event-based control a robotic task is represented by a path parameterized in terms of a scalar called the motion reference variable. The control algorithm is designed to track the location specified by the motion reference variable. The dynamics of the motion reference variable is in accord to a nominal motion plan and, it may be modified when robot's sensors detect unusual circumstances, such as the presence of obstacles. Really, and in a manner of speaking, in event-based control the dynamics of the motion reference variable is governed by a "sensor-based motion plan". However, the experimental case studies presented in [14] are constrained to compute the commanded position and velocity through estimation of the path arc-length given for the task space position of the end-effector. Moreover, there exist open issues concerning the implications of defining the motion reference in a different way than an arc-length estimation, since the motion reference definition must fulfill with conditions required in the stability analysis of the path-based closed-loop system.

In [3] is proposed a method to design non-time based controllers. The method departs from replacing the time variable in the specified desired trajectory by a transformation called state-to-reference projection. Although the general results presented there are illustrated with an application to mobile robot and a motion task specified in the plane, no method is proposed to find suitable transformations that match with the required assumptions for the closed-loop stability, whereby application to complex nonlinear systems, such as a n degrees-of-freedom manipulator, is unclear.

Thus, the idea of using a control algorithm based in the specification of a path and a motion plan requires further study. We address the path control of robot manipulators taking into account that a robotic task can be encoded using a *desired path* and a *desired velocity profile*. The objective of this paper is to provide, from an academic point of sight, a better understanding of the concept of path control through two new path control algorithms. As another contribution of this paper, we have carried out experiments to assess the performance of the proposed path controllers in a two degrees-of-freedom direct-drive arm with revolute joints.

This paper is organized as follows. The task encoding issue is discussed in Section 2. Motion control to solve the requested task is presented in Section 3. The first and second case study on path control are analyzed in Section 4. Experimental evaluation is presented in Section 5. Finally, some concluding remarks are given in Section 6.

2. Task Encoding and Path Control Objective

In this section we introduce basic definitions that shall be invoked throughout this paper. The first two definitions below are related to the way of specifying the motion to be performed by a robot manipulator.

DEFINITION (Desired path). A *desired path* in the joint configuration space is given as a map [6]

$$\mathbf{q}_d(s): [s_0, \infty) \rightarrow \mathbb{R}^n, \quad (2)$$

where $s \in [s_0, \infty)$ is called the path parameter and s_0 a real constant. The image of $\mathbf{q}_d(s)$ is called curve of $\mathbf{q}_d(s)$ in the joint configuration space where the actual joint position \mathbf{q} is desired to live. It is assumed that $\mathbf{q}_d(s)$ is twice continuously differentiable.

DEFINITION (Desired velocity profile). A *desired velocity profile* is defined by a map [2]

$$\vartheta(s): [s_0, \infty) \rightarrow \mathbb{R}_{>0}, \quad (3)$$

where s_0 is a real constant. It is assumed that $\vartheta(s)$ is continuous and differentiable.

The desired path and the desired velocity profile can encode most motion tasks to be performed by manipulators. This will be clear later on. Once that desired path $\mathbf{q}_d(s)$ and the desired velocity profile $\vartheta(s)$ have been specified, then we are able to establish the path control objective.

DEFINITION (Path control objective). The path control objective in joint space is established as

$$\lim_{t \rightarrow \infty} \text{dist} \left(\begin{bmatrix} \mathbf{q}(t) \\ \dot{\mathbf{q}}(t) \end{bmatrix}, \Gamma \right) = 0, \quad (4)$$

where the curve $\Gamma \subset \mathbb{R}^{2n}$ in the state space is defined by

$$\Gamma = \left\{ \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix} \in \mathbb{R}^{2n}: \mathbf{q} = \mathbf{q}_d(s) \text{ and } \dot{\mathbf{q}} = \frac{d\mathbf{q}_d(s)}{ds} \vartheta(s); \forall s \in [s_0, \infty) \right\}, \quad (5)$$

and $\text{dist}(\mathbf{x}, A)$ denotes the smallest distance from a vector $\mathbf{x} \in \mathbb{R}^{2n}$ to any point in a set $A \subset \mathbb{R}^{2n}$, i.e.,

$$\text{dist}(\mathbf{x}, A) = \inf_{\mathbf{p} \in A} \|\mathbf{x} - \mathbf{p}\|.$$

In order to give an interpretation to definition of set Γ the following definition is of worth.

DEFINITION (Motion plan). The velocity profile $\vartheta(s)$ in (3) defines a *motion plan* as the following differential equation [14]

$$\dot{s} = \vartheta(s), \quad s(t_0) = s_0. \quad (6)$$

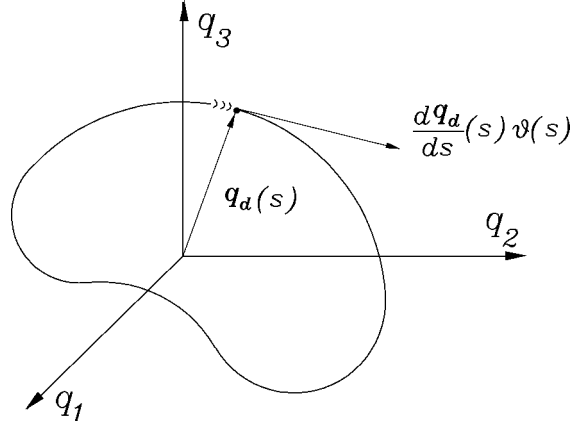


Figure 1. Motion in function of the path parameter s .

Let us notice that using the motion plan (6), the time derivative of $\mathbf{q}_d(s)$ can be computed as

$$\frac{d}{dt}\mathbf{q}_d(s) = \frac{d\mathbf{q}_d(s)}{ds}\vartheta(s), \quad (7)$$

which is a function of the path parameter s . In this way, the set Γ in (5) defines the desired position and velocity in the state space as function of the path parameter s .

Since the vector $(d\mathbf{q}_d(s)/ds)\vartheta(s)$ defines a tangent vector for each point of curve defined by the path $\mathbf{q}_d(s)$, then definition of desired velocity profile $\vartheta(s)$ has effect in the path speed $\|(d\mathbf{q}_d(s)/ds)\vartheta(s)\|$. It should be noticed that if s denotes the path arc-length, then the path speed is given by $\vartheta(s)$ [6]. Figure 1 depicts the path described by a desired path $\mathbf{q}_d(s) \in \mathbb{R}^3$ and the velocity described by $(d\mathbf{q}_d(s)/ds)\vartheta(s)$.

The computation of the velocity profile function $\vartheta(s)$ is usually carried out using a trajectory planning algorithm. As it is indicated by the motion plan (6), the velocity profile function $\vartheta(s)$ represents the relation between $s(t)$ and $\dot{s}(t)$ [2]. Notwithstanding, sometimes the velocity profile computation is achieved as a problem of dynamic programming – see, e.g., [8] for reviewing the phase-plane optimization technique for minimum time trajectory planning. On the other hand, also based in optimization techniques, in [14] is proposed a method for planning velocity profiles through trajectory constraints, namely, velocity, acceleration and jerk-free motion constraints.

It can be shown that solution $s(t)$ of motion plan (6) holds the following properties:

- P1. $s(t) \in [s_0, \infty)$ for all $t \geq t_0$,
- P2. $s(t) \rightarrow \infty$ as $t \rightarrow \infty$.

Moreover, a timed desired position trajectory $\mathbf{q}_r(t) = \mathbf{q}_d(s(t))$ is obtained by solving the motion plan (6) to get $s(t)$ [2].

2.1. AN ACADEMIC EXAMPLE

In order to illustrate basic concepts, let us consider a quite simple, but non trivial, two degrees-of-freedom robot arm. The requested task is to drive the arm in such a way that its joint position $\mathbf{q}(t)$ traces a desired path with prescribed velocity in the joint configuration space. For illustration purpose, let us consider as such a desired path the simple geometric curve given by a circle of radius r_0 centered at the origin of the joint configuration space. The path should be traced with constant tangent velocity – speed – v_0 as depicted in Figure 2.

A proper way for encoding the requested task is through the timed trajectory given by

$$\mathbf{q}_{\text{nom}}(t) = \begin{bmatrix} r_0 \cos\left(\frac{v_0}{r_0}t\right) \\ r_0 \sin\left(\frac{v_0}{r_0}t\right) \end{bmatrix}, \quad (8)$$

which matches the specifications.

We can also notice that the requested task can be encoded by the following desired path

$$\mathbf{q}_d(s) = \begin{bmatrix} r_0 \cos(s) \\ r_0 \sin(s) \end{bmatrix}, \quad (9)$$

where $s \in [s_0, \infty)$, and the desired velocity profile given by

$$\vartheta(s) = \frac{v_0}{r_0}. \quad (10)$$

The set Γ defined in (5) results

$$\Gamma = \left\{ \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix} \in \mathbb{R}^4: \mathbf{q} = \mathbf{q}_d(s) = \begin{bmatrix} r_0 \cos(s) \\ r_0 \sin(s) \end{bmatrix} \text{ and} \right. \\ \left. \dot{\mathbf{q}} = \frac{d\mathbf{q}_d(s)}{ds} \vartheta(s) = \begin{bmatrix} -v_0 \sin(s) \\ v_0 \cos(s) \end{bmatrix} \forall s \in [s_0, \infty) \right\}. \quad (11)$$

Thus, the requested task of tracing a circle of radius r_0 centered at the origin of the joint configuration space at a given speed v_0 is achieved provided that path control objective (4) be satisfied.

It is worthwhile to note that for the requested task, the motion plan (6) is given by

$$\dot{s} = \frac{v_0}{r_0}. \quad (12)$$

Using the motion plan (12), we can compute the time derivative of $\mathbf{q}_d(s)$ as

$$\frac{d}{dt} \mathbf{q}_d(s) = \frac{d\mathbf{q}_d(s)}{ds} \vartheta(s) = \begin{bmatrix} -v_0 \sin(s) \\ v_0 \cos(s) \end{bmatrix}, \quad (13)$$

which is a function of the path parameter s .

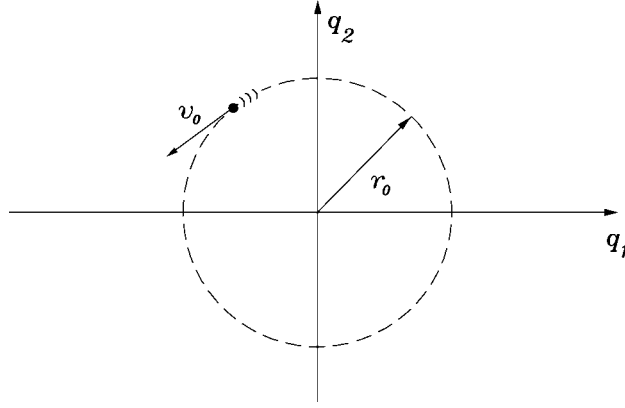


Figure 2. Requested task.

By solving motion plan (12), we have that $s(t) = (v_0/r_0)[t - t_0] + s_0$. By substituting $s(t)$ into the desired path $\mathbf{q}_d(s)$ in (9) we obtain the timed desired position trajectory $\mathbf{q}_r(t)$ that encode the requested task:

$$\mathbf{q}_r(t) = \mathbf{q}_d(s(t)) = \begin{bmatrix} r_0 \cos\left(\frac{v_0}{r_0}[t - t_0] + s_0\right) \\ r_0 \sin\left(\frac{v_0}{r_0}[t - t_0] + s_0\right) \end{bmatrix}, \quad (14)$$

which is identical to the timed trajectory $\mathbf{q}_{\text{nom}}(t)$ in (8) with $s_0 = t_0 = 0$. This shows that the information contained in the timed trajectory (8) can be captured by the desired path (9) and by the desired velocity profile (10). Thus, it can be shown that most of the robotic tasks can be encoded through a desired path and a desired velocity profile.

Two control approaches can be devised for solving the requested task. The first control approach is motion control – also termed tracking control –, which is based in the specification of a timed trajectory $\mathbf{q}_r(t)$ – for the requested task given by (14). Motion control has been well studied in the literature [11, 13]. The second approach is “path control”, which is based in the specification of a desired path and desired velocity profile. Thus, the use of timed desired trajectories of position, as it is required in the motion control approach, is obviated. In this paper we pay attention to the path control approach.

3. Motion Control

Motion control – trajectory tracking control – is the obvious way for solving a requested task. To this end, the desired joint position trajectory $\mathbf{q}_r(t)$ defined by (14) can be used.

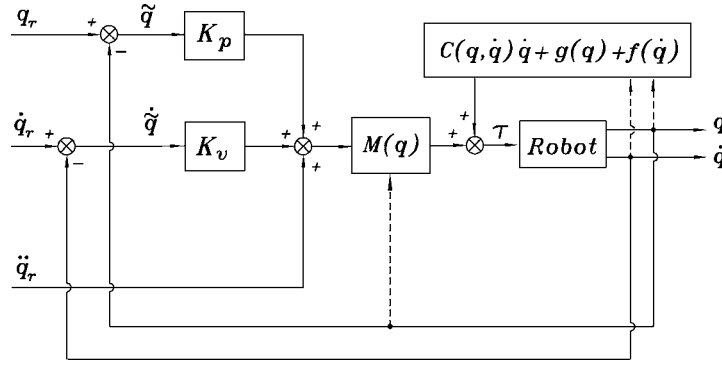


Figure 3. Motion control: computed-torque controller.

Let $\tilde{\mathbf{q}} = \mathbf{q}_r - \mathbf{q}$ denote the joint position error. Thus the task is accomplished provided that the motion control objective defined by

$$\lim_{t \rightarrow \infty} \tilde{\mathbf{q}}(t) = \mathbf{0} \quad (15)$$

is satisfied.

Let us consider the computed-torque control algorithm given by [13, 11]

$$\boldsymbol{\tau} = M(\mathbf{q})[\ddot{\mathbf{q}}_r + K_v \dot{\tilde{\mathbf{q}}} + K_p \tilde{\mathbf{q}}] + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{f}(\dot{\mathbf{q}}), \quad (16)$$

where K_v and K_p and 2×2 symmetric positive definite matrices. The closed-loop equation is obtained by substituting the control law (16) into the robot dynamics (1)

$$\ddot{\tilde{\mathbf{q}}} + K_v \dot{\tilde{\mathbf{q}}} + K_p \tilde{\mathbf{q}} = \mathbf{0}. \quad (17)$$

This a linear autonomous differential equation which is globally asymptotically stable [11, 13]. Therefore the motion control objective (15) is achieved, so the requested robotic task is accomplished.

Beside the computed-torque control (16), many other control structures such as PD+ control [7] and Slotine–Li’s controller – nonadaptive case – [12] can be also utilized. A block diagram of motion controller (16) is shown in Figure 3.

Observe that in the motion control approach, once that timed desired trajectory $\mathbf{q}_r(t)$ has been specified, the primal objective is to satisfy (15). It is clear that in the motion control approach, to carry out the task involves the time in an explicit way, since the task is encoded by a timed trajectory and the objective is to track the position specified by this function. As it will be seen later, in the path control approach the time does not play an important role in the task performance.

4. Path Control

The path control approach is illustrated in this section through two case studies.

4.1. FIRST CASE STUDY

With reference to the academic robotic task considered in this paper given by the desired path (9) and the desired velocity profile (10), the curve Γ in the state space is given by (11):

$$\Gamma = \left\{ \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix} \in \mathbb{R}^4: \mathbf{q} = \mathbf{q}_d(s) = \begin{bmatrix} r_0 \cos(s) \\ r_0 \sin(s) \end{bmatrix} \text{ and} \right. \\ \left. \dot{\mathbf{q}} = \frac{d\mathbf{q}_d(s)}{ds} \vartheta(s) = \begin{bmatrix} -v_0 \sin(s) \\ v_0 \cos(s) \end{bmatrix} \forall s \in [s_0, \infty) \right\}. \quad (18)$$

Thus, the requested task of tracing a circle of radius r_0 centered at the origin of the joint configuration space at a given speed v_0 is achieved provided that

$$\lim_{t \rightarrow \infty} \text{dist} \left(\begin{bmatrix} \mathbf{q}(t) \\ \dot{\mathbf{q}}(t) \end{bmatrix}, \Gamma \right) = 0. \quad (19)$$

A criterion to quantify the accomplishment of the path control objective (19) is given by defining the following variables

$$\mathbf{e}_p(\sigma, \mathbf{q}) = \mathbf{q}_d(\sigma) - \mathbf{q} = \begin{bmatrix} r_0 \cos(\sigma) \\ r_0 \sin(\sigma) \end{bmatrix} - \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}, \quad (20)$$

$$\mathbf{e}_v(\sigma, \dot{\mathbf{q}}) = \frac{d\mathbf{q}_d(\sigma)}{d\sigma} \vartheta(\sigma) - \dot{\mathbf{q}} = \begin{bmatrix} -v_0 \sin(\sigma) \\ v_0 \cos(\sigma) \end{bmatrix} - \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix}, \quad (21)$$

where σ is a new path parameter obtained from the *extended motion plan* defined here by

$$\dot{\sigma} = \vartheta(\sigma) - k \mathbf{e}_p(\sigma, \mathbf{q})^T \frac{d\mathbf{q}_d(\sigma)}{d\sigma} \\ = \frac{v_0}{r_0} - k \mathbf{e}_p(\sigma, \mathbf{q})^T \begin{bmatrix} -r_0 \sin(\sigma) \\ r_0 \cos(\sigma) \end{bmatrix}, \quad (22)$$

where k is a strictly positive constant and $\sigma(0) = s_0$. It is worthwhile to note that if $k = 0$ in the extended motion plan (22), then the motion plan (12) is obtained.

It should be noticed that equations (20) and (21) achieve

$$\mathbf{e}_p(\sigma, \mathbf{q}) = \mathbf{0} \implies \mathbf{q} = \mathbf{q}_d(\sigma) \text{ and } \sigma(t) \in [s_0, \infty), \\ \mathbf{e}_v(\sigma, \dot{\mathbf{q}}) = \mathbf{0} \implies \dot{\mathbf{q}} = \frac{d\mathbf{q}_d(\sigma)}{d\sigma} \vartheta(\sigma). \quad (23)$$

Thus, as a consequence of (23), we have that the set Γ given by expression (18), can also be defined as

$$\Gamma = \left\{ \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix} \in \mathbb{R}^4, \sigma \in [s_0, \infty): \mathbf{e}_p(\sigma, \mathbf{q}) = \mathbf{0}, \mathbf{e}_v(\sigma, \dot{\mathbf{q}}) = \mathbf{0} \right\}.$$

Hence, the control objective (19) is attained provided that

$$\lim_{t \rightarrow \infty} \begin{bmatrix} \mathbf{e}_p(\sigma(t), \mathbf{q}(t)) \\ \mathbf{e}_v(\sigma(t), \dot{\mathbf{q}}(t)) \end{bmatrix} = \mathbf{0}. \quad (24)$$

Thus, given a desired path $\mathbf{q}_d(s)$ and a desired velocity profile $\vartheta(s)$, we proposed the following path controller, which is expressed in terms of the path parameter $\sigma(t)$,

$$\begin{aligned} \boldsymbol{\tau} = & M(\mathbf{q}) \left[\frac{d}{dt} \left[\frac{d\mathbf{q}_d(\sigma)}{d\sigma} \vartheta(\sigma) \right] + K_v \mathbf{e}_v(\sigma, \dot{\mathbf{q}}) + K_p \mathbf{e}_p(\sigma, \mathbf{q}) \right] \\ & + C(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{f}(\dot{\mathbf{q}}), \end{aligned} \quad (25)$$

where K_p and K_v are 2×2 diagonal positive definite matrices, i.e., $K_p = \text{diag}\{k_{p1}, k_{p2}\}$, and $K_v = \text{diag}\{k_{v1}, k_{v2}\}$. It is worth noticing that

$$\frac{d}{dt} \left[\frac{d\mathbf{q}_d(\sigma)}{d\sigma} \vartheta(\sigma) \right] = \frac{d}{dt} \begin{bmatrix} -v_0 \sin(\sigma) \\ v_0 \cos(\sigma) \end{bmatrix} = \begin{bmatrix} -v_0 \cos(\sigma) \dot{\sigma} \\ -v_0 \sin(\sigma) \dot{\sigma} \end{bmatrix}.$$

The path controller (25) is based in the inverse dynamics technique [13]. Figure 4 shows a block diagram of implementation of path controller (22) and (25).

Let us notice that the dynamics of $\sigma(t)$, defined by the extended motion plan (22), depends on the signal $\mathbf{e}_p(t)$. Furthermore, in the extended motion plan (22) can be observed that if $\mathbf{e}_p(t) \rightarrow \mathbf{0}$ as $t \rightarrow \infty$ then $\dot{\sigma}(t) \rightarrow v_0/r_0$ as $t \rightarrow \infty$, therefore the motion plan (12) will be recovered as $\mathbf{e}_p(t) \rightarrow \mathbf{0}$.

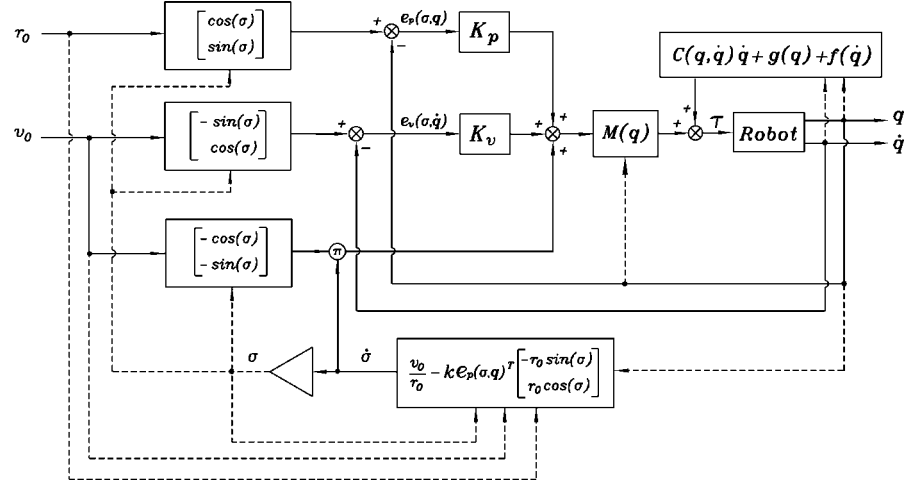


Figure 4. Path control system: first case study.

The closed-loop system is obtained by substituting the control law (25) in the robot model (1). This yields the following closed-loop equation

$$\frac{d}{dt} \begin{bmatrix} \mathbf{e}_p \\ \mathbf{e}_v \\ \sigma \end{bmatrix} = \begin{bmatrix} \mathbf{e}_v - k \begin{bmatrix} \mathbf{e}_p^T \begin{bmatrix} -r_0 \sin(\sigma) \\ r_0 \cos(\sigma) \end{bmatrix} \\ -K_v \mathbf{e}_v - \bar{K}_p \mathbf{e}_p \end{bmatrix} \begin{bmatrix} -r_0 \sin(\sigma) \\ r_0 \cos(\sigma) \end{bmatrix} \\ \frac{v_0}{r_0} - k \mathbf{e}_p^T \begin{bmatrix} -r_0 \sin(\sigma) \\ r_0 \cos(\sigma) \end{bmatrix} \end{bmatrix}. \quad (26)$$

System (26) is autonomous. In order to analyze the asymptotic behavior of the solutions of closed-loop system (26), we invoke the LaSalle's invariance principle [4]. Let us propose the nonnegative function

$$V(\mathbf{e}_p, \mathbf{e}_v, \sigma) = \frac{1}{2} \mathbf{e}_p^T \mathbf{e}_p + \frac{1}{2} \mathbf{e}_v^T K_p^{-1} \mathbf{e}_v.$$

The time derivative of $V(\mathbf{e}_p, \mathbf{e}_v)$ along of system trajectories (26) yields

$$\dot{V}(\mathbf{e}_p, \mathbf{e}_v, \sigma) = -\mathbf{e}_v^T K_p^{-1} K_v \mathbf{e}_v - k \left[\mathbf{e}_p^T \begin{bmatrix} -r_0 \sin(\sigma) \\ r_0 \cos(\sigma) \end{bmatrix} \right]^2.$$

We define the set

$$\begin{aligned} \Omega &= \left\{ \begin{bmatrix} \mathbf{e}_p \\ \mathbf{e}_v \\ \sigma \end{bmatrix} \in \mathbb{R}^5 : \dot{V}(\mathbf{e}_p, \mathbf{e}_v, \sigma) = 0 \right\} \\ &= \left\{ \begin{bmatrix} \mathbf{e}_p \\ \mathbf{e}_v \\ \sigma \end{bmatrix} \in \mathbb{R}^5 : \mathbf{e}_p^T \begin{bmatrix} -r_0 \sin(\sigma) \\ r_0 \cos(\sigma) \end{bmatrix} = 0, \mathbf{e}_v = \mathbf{0}, \text{ and } \sigma \in \mathbb{R} \right\}. \end{aligned}$$

The next step in the application of the LaSalle's invariance principle is to find the largest invariant set in Ω . We denote S the largest invariant set in Ω

$$S = \left\{ \begin{bmatrix} \mathbf{e}_p \\ \mathbf{e}_v \\ \sigma \end{bmatrix} \in \mathbb{R}^5 : \mathbf{e}_p = \mathbf{0}, \mathbf{e}_v = \mathbf{0}, \text{ and } \sigma \in [s_0, \infty) \right\}.$$

Therefore, according to the LaSalle's invariance principle, the solutions of the closed-loop system (26) approaches S as $t \rightarrow \infty$ for any initial condition $\mathbf{e}_p(\sigma(0))$, $\mathbf{q}(0) \in \mathbb{R}^2$, $\mathbf{e}_v(\sigma(0))$, $\dot{\mathbf{q}}(0) \in \mathbb{R}^2$, and $\sigma(0) = s_0$. Thus, the requested task is attained since we have demonstrated that

$$\lim_{t \rightarrow \infty} \begin{bmatrix} \mathbf{e}_p(\sigma(t), \mathbf{q}(t)) \\ \mathbf{e}_v(\sigma(t), \dot{\mathbf{q}}(t)) \end{bmatrix} = \mathbf{0}$$

is satisfied, then the control objective (19) is also achieved.

The path control objective (24) does not impose the time as an important ingredient to achieve the task since it is not necessary to use timed desired trajectories. Thus, without the use of timed desired trajectories, and with the aim of coping with

errors in the motion performed by the manipulator, path control can be achieved by controlling the dynamics of path parameter $\sigma(t)$. This goal is attained by the extended motion plan (22).

4.2. SECOND CASE STUDY

In this section we address the special case of previous controller in the limit when $k \rightarrow \infty$ into the extended motion plan (22). We begin by expressing in an equivalent way the set Γ in (11) as

$$\Gamma = \left\{ \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix} \in \mathbb{R}^4: \|\mathbf{q}\| = r_0 \text{ and } \|\dot{\mathbf{q}}\| = v_0 \right\}. \quad (27)$$

On the other hand, it is possible to rewrite the extended motion plan (22) in the following form

$$\frac{1}{k} \dot{\sigma} = \frac{1}{k} \frac{v_0}{r_0} - \mathbf{e}_p^T \begin{bmatrix} -r_0 \sin(\sigma) \\ r_0 \cos(\sigma) \end{bmatrix}. \quad (28)$$

Considering $k = \infty$ in the extended motion plan (28), we found that it degenerates into the algebraic equation

$$\mathbf{e}_p^T \begin{bmatrix} -r_0 \sin(\sigma) \\ r_0 \cos(\sigma) \end{bmatrix} = \left[\begin{bmatrix} r_0 \cos(\sigma) \\ r_0 \sin(\sigma) \end{bmatrix} - \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \right]^T \begin{bmatrix} -r_0 \sin(\sigma) \\ r_0 \cos(\sigma) \end{bmatrix} = 0. \quad (29)$$

After some algebra, it is possible to solve σ from the algebraic equation (29)

$$\bar{\sigma}(\mathbf{q}) = \arctan\left(\frac{q_2}{q_1}\right). \quad (30)$$

In view of

$$\begin{aligned} \cos\left(\arctan\left(\frac{q_2}{q_1}\right)\right) &= \frac{q_1}{\sqrt{q_1^2 + q_2^2}}, \\ \sin\left(\arctan\left(\frac{q_2}{q_1}\right)\right) &= \frac{q_2}{\sqrt{q_1^2 + q_2^2}}, \end{aligned}$$

then we have that \mathbf{e}_p and \mathbf{e}_v , given by equations (20) and (21) respectively, can be rewritten as

$$\mathbf{e}_p(\bar{\sigma}(\mathbf{q}), \mathbf{q}) = \mathbf{q}_d(\bar{\sigma}(\mathbf{q})) - \mathbf{q} = \begin{bmatrix} r_0 \frac{q_1}{\sqrt{q_1^2 + q_2^2}} \\ r_0 \frac{q_2}{\sqrt{q_1^2 + q_2^2}} \end{bmatrix} - \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}, \quad (31)$$

$$\mathbf{e}_v(\bar{\sigma}(\mathbf{q}), \dot{\mathbf{q}}) = \frac{d\mathbf{q}_d}{d\bar{\sigma}}(\bar{\sigma}(\mathbf{q})) \dot{\bar{\sigma}}(\mathbf{q}) - \dot{\mathbf{q}} = \begin{bmatrix} -v_0 \frac{q_2}{\sqrt{q_1^2 + q_2^2}} \\ v_0 \frac{q_1}{\sqrt{q_1^2 + q_2^2}} \end{bmatrix} - \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix}. \quad (32)$$

Let us notice that a measure of how far a point $[\mathbf{q}^T \ \dot{\mathbf{q}}^T]^T$ of the state space from the curve Γ given in (27) can be* the expressions given by (31), and (32), which are rewritten as

$$\mathbf{e}_p(\bar{\sigma}(\mathbf{q}), \mathbf{q}) = r_0 \frac{\mathbf{q}}{\|\mathbf{q}\|} - \mathbf{q}, \quad (33)$$

$$\mathbf{e}_v(\bar{\sigma}(\mathbf{q}), \dot{\mathbf{q}}) = v_0 \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \frac{\mathbf{q}}{\|\mathbf{q}\|} - \dot{\mathbf{q}}. \quad (34)$$

From equations (31) and (32), we can see that $\mathbf{e}_p(\bar{\sigma}(\mathbf{q}), \mathbf{q})$ and $\mathbf{e}_v(\bar{\sigma}(\mathbf{q}), \dot{\mathbf{q}})$ satisfy

$$\begin{aligned} \mathbf{e}_p(\bar{\sigma}(\mathbf{q}), \mathbf{q}) = \mathbf{0} &\implies \mathbf{q}_d(\bar{\sigma}(\mathbf{q})) = \mathbf{q} &\implies \|\mathbf{q}\| = r_0, \\ \mathbf{e}_v(\bar{\sigma}(\mathbf{q}), \dot{\mathbf{q}}) = \mathbf{0} &\implies \frac{d\mathbf{q}_d}{d\bar{\sigma}}(\bar{\sigma}(\mathbf{q}))\vartheta(\bar{\sigma}(\mathbf{q})) = \dot{\mathbf{q}} &\implies \|\dot{\mathbf{q}}\| = v_0. \end{aligned} \quad (35)$$

The requested task of tracing a circle of radius r_0 centered at the origin of the joint configuration space at a given speed v_0 is achieved provided that

$$\lim_{t \rightarrow \infty} \text{dist} \left(\begin{bmatrix} \mathbf{q}(t) \\ \dot{\mathbf{q}}(t) \end{bmatrix}, \Gamma \right) = 0, \quad (36)$$

where Γ is the set defined by (27). As a consequence of (35), we have that the curve Γ defined in (27) can also be redefined as

$$\begin{aligned} \Gamma &= \left\{ \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix} \in \mathbb{R}^4: \mathbf{e}_p(\bar{\sigma}(\mathbf{q}), \mathbf{q}) = \mathbf{0}, \text{ and } \mathbf{e}_v(\bar{\sigma}(\mathbf{q}), \dot{\mathbf{q}}) = \mathbf{0} \right\} \\ &= \left\{ \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix} \in \mathbb{R}^4: \mathbf{q} = \mathbf{q}_d(\bar{\sigma}(\mathbf{q})) \text{ and } \dot{\mathbf{q}} = \frac{d\mathbf{q}_d}{d\bar{\sigma}}(\bar{\sigma}(\mathbf{q}))\vartheta(\bar{\sigma}(\mathbf{q})) \right\}. \end{aligned} \quad (37)$$

Hence, if the limit

$$\lim_{t \rightarrow \infty} \begin{bmatrix} \mathbf{e}_p(\bar{\sigma}(\mathbf{q}(t)), \mathbf{q}(t)) \\ \mathbf{e}_v(\bar{\sigma}(\mathbf{q}(t)), \dot{\mathbf{q}}(t)) \end{bmatrix} = \mathbf{0} \quad (38)$$

is achieved, then the control objective (36) is fulfilled.

In order to satisfy (38), we propose the following path control law inspired from the inverse dynamics approach

$$\begin{aligned} \boldsymbol{\tau} &= M(\mathbf{q}) \left[\frac{d}{dt} \left[v_0 \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \frac{\mathbf{q}}{\|\mathbf{q}\|} \right] + K_v \mathbf{e}_v(\bar{\sigma}(\mathbf{q}), \dot{\mathbf{q}}) + k_p \mathbf{e}_p(\bar{\sigma}(\mathbf{q}), \mathbf{q}) \right] \\ &\quad + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{f}(\dot{\mathbf{q}}), \end{aligned} \quad (39)$$

* Notice that $\mathbf{e}_p(\mathbf{q})$ gives the smallest Euclidean distance in joint position space between \mathbf{q} and the set of points defining the circle $\|\mathbf{q}\| = r_0$.

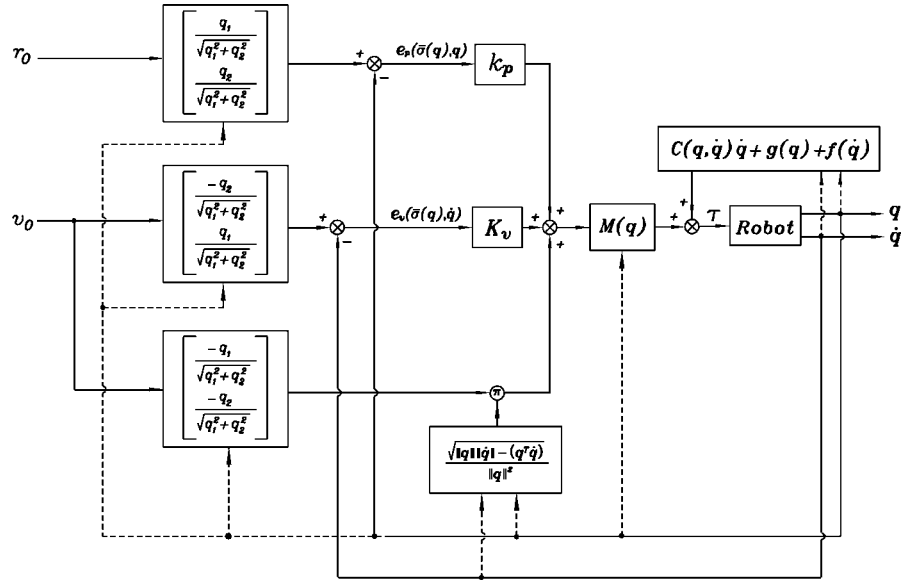


Figure 5. Second case study: path control system.

where $k_p > 0$ and K_v is a 2×2 symmetric positive definite matrix. It is valid to note that

$$\begin{aligned} \frac{d}{dt} \left[v_0 \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \frac{\mathbf{q}}{\|\mathbf{q}\|} \right] &= \frac{d}{dt} \begin{bmatrix} -v_0 \frac{q_2}{\sqrt{q_1^2 + q_2^2}} \\ v_0 \frac{q_1}{\sqrt{q_1^2 + q_2^2}} \end{bmatrix} \\ &= \begin{bmatrix} -v_0 \frac{q_1}{\sqrt{q_1^2 + q_2^2}} \\ -v_0 \frac{q_2}{\sqrt{q_1^2 + q_2^2}} \end{bmatrix} \frac{\sqrt{\|\mathbf{q}\|^2 \|\dot{\mathbf{q}}\|^2 - (\mathbf{q}^T \dot{\mathbf{q}})^2}}{\|\mathbf{q}\|^2}. \end{aligned}$$

Figure 5 depicts a block diagram for implementation of path controller (39).

The closed-loop system is obtained by substituting the control law (39) in the robot model (1). This yields

$$\frac{d}{dt} \underbrace{\left[v_0 \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \frac{\mathbf{q}}{\|\mathbf{q}\|} - \dot{\mathbf{q}} \right]}_{\mathbf{e}_v(\bar{\sigma}(q), \dot{q})} + K_v \mathbf{e}_v(\bar{\sigma}(q), \dot{q}) + k_p \mathbf{e}_p(\bar{\sigma}(q), q) = \mathbf{0} \quad (40)$$

which can be expressed as

$$\frac{d}{dt} \begin{bmatrix} \mathbf{e}_p \\ \mathbf{e}_v \end{bmatrix} = \begin{bmatrix} \mathbf{e}_v + S(\mathbf{q}, \dot{\mathbf{q}}) r_0 \frac{\mathbf{q}}{\|\mathbf{q}\|} \\ -K_v \mathbf{e}_v - k_p \mathbf{e}_p \end{bmatrix}, \quad (41)$$

where the skew-symmetric matrix $S(\mathbf{q}, \dot{\mathbf{q}})$ is given by

$$S(\mathbf{q}, \dot{\mathbf{q}}) = \left[\frac{\sqrt{\|\mathbf{q}\|^2 \|\dot{\mathbf{q}}\|^2 - (\mathbf{q}^T \dot{\mathbf{q}})^2}}{\|\mathbf{q}\|^2} - \frac{v_0}{r_0} \right] \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}.$$

In virtue of (35), and noting that $(d/dt)\|\mathbf{q}\| = (d/dt)r_0 = 0$ implies $\mathbf{q}^T \dot{\mathbf{q}} = 0$, then we have the conclusion that $[\mathbf{e}_p^T \ \mathbf{e}_v^T]^T = \mathbf{0}$ is an equilibrium point of the closed-loop equation (41).

To carry out the stability analysis, we propose the following Lyapunov function candidate:

$$V(\mathbf{e}_p, \mathbf{e}_v) = \frac{1}{2} \mathbf{e}_v^T \mathbf{e}_v + \frac{k_p}{2} \mathbf{e}_p^T \mathbf{e}_p. \quad (42)$$

The time derivative of (42) along of the system trajectories (41) yields

$$\dot{V}(\mathbf{e}_p, \mathbf{e}_v) = -\mathbf{e}_v^T K_v \mathbf{e}_v,$$

where we utilized the skew-symmetry of $S(\mathbf{q}, \dot{\mathbf{q}})$ to have

$$k_p \mathbf{e}_p^T S(\mathbf{q}, \dot{\mathbf{q}}) r_0 \frac{\mathbf{q}}{\|\mathbf{q}\|} = k_p r_0 \frac{r_0 - \|\mathbf{q}\|}{\|\mathbf{q}\|^2} \mathbf{q}^T S(\mathbf{q}, \dot{\mathbf{q}}) \mathbf{q} = 0.$$

The conclusion of stability of the origin $[\mathbf{e}_p^T \ \mathbf{e}_v^T]^T = \mathbf{0}$ of the closed-loop system state space is implicated from the fact that $\dot{V}(\mathbf{e}_p, \mathbf{e}_v)$ is a negative semidefinite function. Since the system (41) is autonomous, then it is possible to invoke the Krasovskii–LaSalle’s theorem [15] to study asymptotic stability. To this end, we define the set

$$\begin{aligned} \Omega &= \left\{ \begin{bmatrix} \mathbf{e}_p \\ \mathbf{e}_v \end{bmatrix} \in \mathbb{R}^4: \dot{V}(\mathbf{e}_p, \mathbf{e}_v) = 0 \right\} \\ &= \left\{ \begin{bmatrix} \mathbf{e}_p \\ \mathbf{e}_v \end{bmatrix} \in \mathbb{R}^4: \mathbf{e}_p \in \mathbb{R}^2, \text{ and } \mathbf{e}_v = \mathbf{0} \right\}. \end{aligned}$$

Since $k_p \mathbf{e}_p = \mathbf{0}$ has the unique solution $\mathbf{e}_p = \mathbf{0}$, hence the largest invariant set in Ω is $[\mathbf{e}_p^T \ \mathbf{e}_v^T]^T = \mathbf{0}$. Therefore, invoking the Krasovskii–LaSalle’s theorem, this is an asymptotically stable equilibrium. Among other things this implies (38), then also the control objective (36) which ensures that the requested task is attained.

With the path controller (39), the use of timed desired trajectories have been obviated again. Path control was achieved by the procedure of relating the encoding of requested task with the robot joint position.

5. Experimental Evaluation

The experiments presented in this section have been carried out on a robotic arm built at CICESE Research Center. This is a direct-drive vertical arm with two

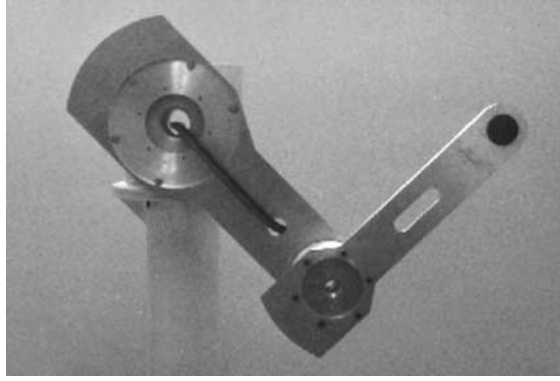


Figure 6. Experimental robot arm.

degrees-of-freedom whose rigid links are joined with revolute joints (see Figure 6). For a complete description and model of the set-up, the reader is referred to [9, 10].

Experiments showed that static and Coulomb friction at the joints are present and they depend in a complex manner on the joint position and velocity. We have decided to compensate only viscous friction. The remaining friction terms acts as disturbance of the closed-loop system.

The path controllers (22) and (25), and (39) have been implemented in the experimental system. The experiments were conducted using the following data: desired circle's radius $r_0 = 57.3$ [deg], desired speed $v_0 = 114.59$ [deg/s], and gains

$$k_p = 800.0 [1/s^2],$$

$$K_v = \text{diag}\{45.0, 100.0\} [1/s].$$

Path controller (22) and (25) was implemented using $K_p = \text{diag}\{k_p, k_p\}$, and $k = 20$ in the extended motion plan (22). The arm initial joint configuration was

$$\begin{bmatrix} q_1(0) \\ q_2(0) \end{bmatrix} = \begin{bmatrix} 40.0 \\ 70.0 \end{bmatrix} [\text{deg}].$$

5.1. FIRST CASE STUDY: EXPERIMENTS

The experimental results of using controller (22) and (25) are shown in Figures 7 and 8. The plot in Figure 7 shows q_2 versus q_1 . It is noted a natural behavior of the arm in the position space tracing a circle centered at the origin having a radius $r_0 = 57.3$ [deg], so performing the desired task. The time evolution of $\|\mathbf{e}_p\|$ and $\|\mathbf{e}_v\|$ is depicted in Figure 8. Both have a clear tendency to zero – as expected – but small oscillation persist. The norm $\|\mathbf{e}_p\|$ exhibits maximum peaks of 1.5 [deg] in steady state, then peak deviation of 2.6% from the nominal r_0 value – 57.3 [deg]. On the other hand, $\|\mathbf{e}_v\|$ in Figure 8 presents maximum peaks of 14.5 [deg/s],

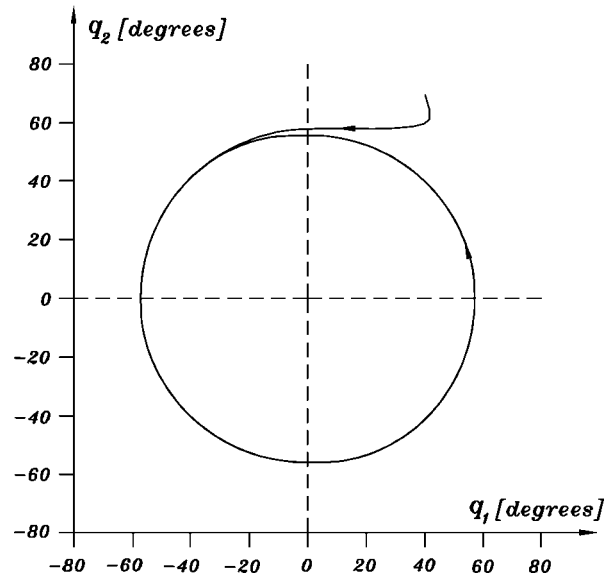


Figure 7. First case study: q_2 versus q_1 .

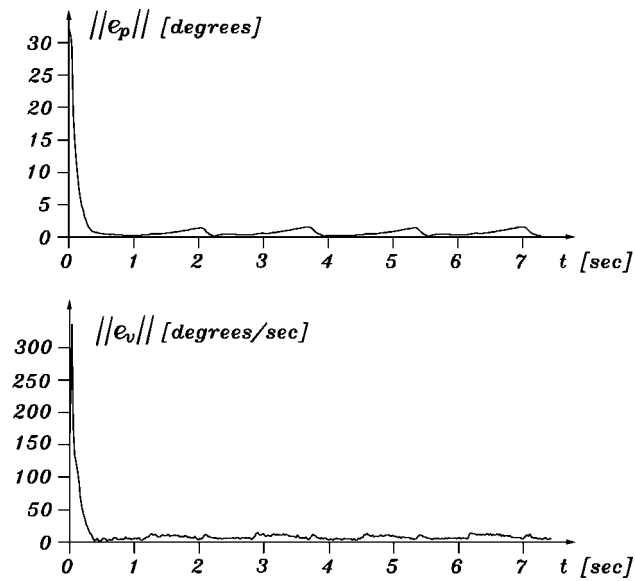


Figure 8. First case study: time evolution of $\|e_p\|$ and $\|e_v\|$.

leading to a peak deviation of 12.65% from the specified value of $v_0 = 114.59$ [deg/s].

Simulation evidences showed that the remaining persistent errors in both $\|e_p\|$ and $\|e_v\|$ are due mainly to uncompensated Coulomb friction effects in the arm

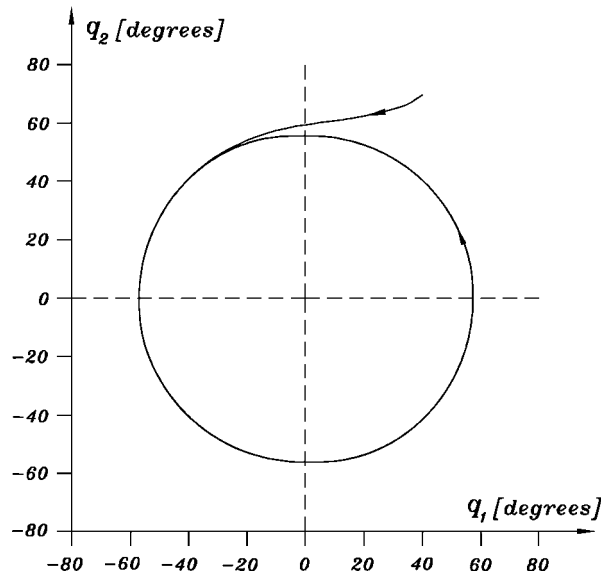


Figure 9. Second case study: q_2 versus q_1 .

joints. Notwithstanding, further reduction of these errors can be achieved utilizing higher gains, but at the price of exciting high frequency modes of the arm.

5.2. SECOND CASE STUDY: EXPERIMENTS

The second case study on path control refers to path controller (39). Experiments are presented in Figures 9 and 10. Figure 9 shows the path traced by the arm in the position space q_1 - q_2 . Like to performance of controller (22) and (25), a natural behavior of the arm in the position space is noted, performing successfully the requested task. The time evolution of $\|\mathbf{e}_p\|$ and $\|\mathbf{e}_v\|$ is shown in Figure 10. The time evolution of norm $\|\mathbf{e}_p\|$ presents maximum peaks of 1.4 [deg] in steady state, which give a peak deviation of 2.44% from the nominal r_0 value - 57.3 [deg]. On the other hand, time evolution of $\|\mathbf{e}_v\|$ presents maximum peaks of 17.2 [deg/s], guiding to a peak deviation of 15.01% from the specified value of v_0 - 114.59 [deg/s].

Similar to controller (22) and (25), improvement in performance can be obtained by increasing the gains, but some undesirable phenomenon can appear, such as enlargement of noise in joint velocity estimation, among others.

6. Concluding Remarks

Path control is an approach to perform a class of robotic tasks where motion coordination is more important than timing. We have formulated the path control problem specifying a desired path and a desired velocity profile. High level tasks can be

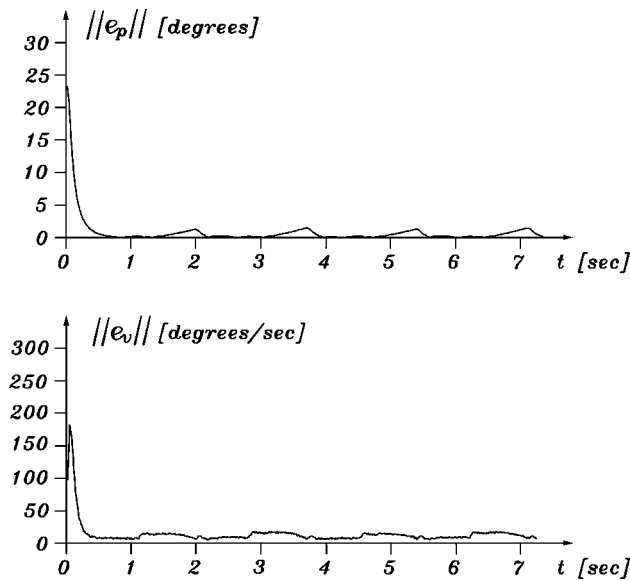


Figure 10. Second case study: time evolution of $\|\mathbf{e}_p\|$ and $\|\mathbf{e}_v\|$.

coded into the path control law, so avoiding the use of timed desired trajectories for encoding the task. The path control concept has been illustrated in this paper by means of the simple academic task of tracing a circle in joint space having constant tangent speed. Solutions based on trajectory tracking control and two proposed path control algorithms have been discussed. Experimental evaluation on a direct-drive arm has given evidence of the practical feasibility of the proposed approaches.

References

1. Abdallah, C., Dawson, D., Dorato, P., and Jamshidi M.: Survey of robust control for rigid robots, *IEEE Control Systems Mag.* **11**(2) (1991), 24–30.
2. Dahl, O. and Nielsen, L.: Torque-limited path following by on-line trajectory time scaling, *IEEE Trans. Robotics Automat.* **6**(5) (1990), 554–561.
3. Kang, W., Xi, N., and Tan, J.: Analysis and design of non-time based motion controller for mobile robots, in: *Proc. of the IEEE Internat. Conf. on Robotics and Automation*, Detroit, MI, 1999, pp. 2964–2969.
4. Khalil, H.: *Nonlinear Systems*, Prentice-Hall, Upper Saddle River, 1996.
5. Lewis, F. L., Abdallah, C. T., and Dawson, D.: *Control of Robot Manipulators*, Macmillan, New York, 1993.
6. Marsden, J. E. and Tromba, A. J.: *Vector Calculus*, Freeman, New York, 1988.
7. Paden, B. and Panja, R.: Globally asymptotically stable ‘PD+’ controller for robot manipulators, *Internat. J. Control* **47**(6) (1988), 1697–1712.
8. Pfeiffer, F. and Johanni, R.: A concept for manipulator trajectory planning, *IEEE J. Robotics Automat.* **3**(2) (1987), 115–123.
9. Reyes, F. and Kelly, R.: Experimental evaluation of identification schemes on a direct drive robot, *Robotica* **15** (1997), 563–571.

10. Reyes, F. and Kelly, R.: Experimental evaluation of model-based controllers on a direct-drive robot arm, *Mechatronics* **11** (2001), 267–282.
11. Sciavicco, L. and Siciliano, B.: *Modeling and Control of Robot Manipulators*, Springer, London, 2000.
12. Slotine, J. J. E. and Li, W.: *Applied Nonlinear Control*, Prentice-Hall, Englewood Cliffs, NJ, 1991.
13. Spong, M. W. and Vidyasagar, M.: *Robot Dynamics and Control*, Wiley, New York, 1989.
14. Tarn, T. J., Xi, N., and Bejczy, A. K.: Path-based approach to integrated planning and control of robotic manipulators, *Automatica* **32** (1996), 1675–1687.
15. Vidyasagar, M.: *Nonlinear Systems Analysis*, Prentice-Hall, Englewood Cliffs, NJ, 1993.