Suboptimal output regulation of robotic manipulators by iterative learning*

P. Lucibello

Sogin
Società gestione impianti nucleari
Via Torino, 6, 00184 Roma, Italy
lucibello@sogin.it

S. Panzieri

Dip. Informatica e Automazione
Università degli Studi "Roma Tre"
Via della Vasca Navale 79, 00146 Roma, Italy
panzieri@uniroma3.it

F. Pascucci

Dip. Informatica e Sistemistica
Università degli Studi "La Sapienza"
Via Eudossiana, 18, 00189 Roma, Italy
pascucci@dia.uniroma3.it

Abstract

The problem of tracking output trajectories generated by a finite dimensional exosystem using a finite dimensional iterative learning algorithm has been largely considered in literature for linear systems. In this paper, for a class of nonlinear systems, a piece-wise continuous reference is considered and a suitable norm of the output error is minimised rather then zeroed. An application to robotic manipulators concludes the paper.

1 Introduction

In a recent work [1] a new iterative learning algorithm for output tracking of linear, finite dimensional, dynamic systems has been presented. The control system used in that paper is defined by extending the plant with a finite dimensional linear exosystem from which depend both the plant control input and the output reference trajectory. An output zeroing problem is hence solved by means of a robust iterative learning algorithm, finding an initial condition for both plant state and exosystem state such that the difference between the plant output and the reference trajectory is zeroed.

The learning of the initial condition strongly differentiates the algorithm in [1] w.r.t. others available in the literature. The output tracking by iterative learning problems that can be found in most of the actual literature (see for example [2, 3, 4, 5, 6]) consist in requiring that a function of the plant state exactly tracks a desired function of the time belonging to some infinite-dimensional linear space. This amplitude of the set of desired output trajectories leads to learning algorithms that are infinite-dimensional. In contrast, the exosystem formulation above presented reduces the set of desired output trajectories to a finite-dimensional one, and then the problem is solved with a finite-dimensional learning algorithm.

The purpose of this paper is to present an extension of the algorithm introduced for linear systems in [1], and applied in [7] to a flexible link, to a class of nonlinear systems, namely quasi-linear systems, that will be described in Section 3. A piece-wise continuous reference that cannot be generated by a linear exosystem is considered and a suitable norm of the output error is minimised rather then zeroed. The novelty of this approach, i.e., the choice of simply minimising the output error, is forced to the consideration that the zeroing result is non possible for such systems.

In Section 4 an application to rigid robotic manipulators, transformed into quasi-linear systems via an appropriate feedback, is described and some experiments conclude the paper.

2 Problem setting

Let Y and W be the linear space of piece-wise continuous mappings $[0,T] \rightarrow \Re^m$ and $[0,T] \rightarrow \Re^w$, respectively. On these spaces consider the following inner product and norm

$$< \chi_1(t), \chi_2(t) > \;=\; \int_0^T \chi_1^{'}(t)\chi_2(t)dt \qquad (1)$$

$$\|\chi_1(t)\| \;=\; \frac{1}{2}\int_0^T \chi_1^{'}(t)\chi_1(t)dt. \qquad (2)$$

Denote by Z a neighborhood of the zero of $\Re^q, q \geq m$, and by D a neighborhood of the zero of W. Let's consider a *control system* as defined by a continuous mapping

$$\Gamma : Z \times D \rightarrow Y, \quad y = \Gamma(\zeta, d) \qquad (3)$$

where $\zeta \in Z$ is the control variable, $d \in D$ the disturbance, and $y \in Y$ the output of the system.

For system (3), we wish to solve, by means of an iterative learning algorithm, the following problem.

*Submitted for presentation at the 11th International Conference on Advanced Robotics, June 30 - July 3, 2003 University of Coimbra, Portugal

Problem 1 Optimal output regulation problem*: given the control system (3) and a fixed disturbance $d \in D$, find a control $\zeta \in Z$, such that $\|y\|$ is minimised.*

In a learning contest, a quite general approach to this problem would be assuming that the function Γ is not available and that the disturbance d is unknown. Under the hypothesis that the disturbance does not change from trial to trial, at each trial the value of the function Γ can be computed and the search for the minimum tentatively accomplished in the framework of nonlinear programming (see e.g.[8]). In other words, one may try to use one of the existing optimisation algorithms, that do not need the prior knowledge of the derivatives of the function to be minimised.

Here we do not discuss the implementation of nonlinear programming algorithms and we focus on an approach based upon the hypothesis that we have a partial knowledge of the control system to be regulated. In particular we restrict the analysis to a specific class of control systems, that we define as quasi-linear and for which we introduce a suboptimal algorithm. This analysis is developed in next Section, where the meanings of quasi-linear and suboptimal are made precise.

3 A suboptimal algorithm for quasi-linear systems

Suppose that the output of the control system (3) could be cast in the following form

$$y(t,\zeta) = \Psi(t)\zeta + \gamma(t) + \varepsilon\lambda(\zeta,t,\varepsilon) \quad \zeta \in \Re^q, t \in [0,T] \tag{4}$$

where $y(t,\zeta) \in \Re^m$, $\Psi(t)$ is a linear map between $\Re^q$ and $\Re^m$, $\gamma(t) \in \Re^m$ is the disturbance, and $\lambda(\zeta,t,\varepsilon)$ is a possibly nonlinear function, two times continuously differentiable with respect to ζ and continuously differentiable (smooth) with respect to the small parameter $\varepsilon \in \Re$. Functions $\Psi(t), \gamma(t)$, and $\lambda(\zeta,t,\varepsilon)$ are supposed to be piece-wise continuous with respect to $t \in [0,T]$.

A system whose output has the form (4) is referred to as quasi-linear, being the output $y(t,\zeta)$ linear with respect to ζ for $\varepsilon = 0$.

Given the output (4), the function $\varphi(\zeta)$ to be minimised, according to the defined norm, assumes the form

$$\varphi(\zeta) = \|y(t,\zeta)\| = \frac{1}{2}\zeta' N\zeta + \beta'\zeta + \mu + \varepsilon\alpha(\zeta,\varepsilon) \tag{5}$$

with

$$N = 2\|\Psi(t)\|, \quad \beta =< \Psi(t), \gamma(t) >, \quad \mu = \|\gamma(t)\|$$

$$\alpha(\zeta,\varepsilon) = \int_0^T \lambda'(\zeta,t,\varepsilon)(\Psi(t)\zeta + \gamma(t) + \frac{\varepsilon}{2}\lambda(\zeta,t,\varepsilon))dt$$

Lemma 1 *Assume that N is invertible, then for sufficiently small ε the function (5) has an isolated minimum.*

Proof. The stationary condition for (5) gives

$$\sigma = N\zeta + \beta + \varepsilon\frac{\partial}{\partial\zeta}\alpha(\zeta,\varepsilon) = 0 \tag{6}$$

with σ the gradient of the objective function. Due to invertibility of N and its positive definition an isolated minimum exists for $\varepsilon = 0$. Moreover, since $\frac{\partial}{\partial\zeta}\alpha(\zeta,\varepsilon)$ is smooth, by the Implicit Function Theorem [9] an isolated minimum, smoothly dependent on ε, exists also for sufficiently small ε.‡

From now on we assume that the matrix N is invertible.

A classic approach to solve the given optimisation problem could be using a quasi-Newton algorithm [8]. Suppose that the gradient of the objective function could be computed at the k-th trial and that N is known. Denote by ζ_k and σ_k the values of the control variable and of the gradient at the k-th trial, respectively. Since N is a good approximation of the Hessian for small ε, we could try to reach the minimum by using the following rule

$$\zeta_{k+1} = \zeta_k - N^{-1}\sigma_k. \tag{7}$$

Lemma 2 *For sufficiently small ε, the sequence (7) converges to a point ζ_∞ smoothly dependent on ε and $\sigma_k \to 0$ as $k \to \infty$.*

Proof. By simple algebraic manipulations, we get from eqs. (7) and (6)

$$\zeta_{k+1} = -N^{-1}\beta - \varepsilon N^{-1}\left.\frac{\partial}{\partial\zeta}\alpha(\zeta,\varepsilon)\right|_{\zeta=\zeta_k} = T(\zeta_k,\varepsilon). \tag{8}$$

Now, since $\lambda(\zeta,t,\varepsilon)$ is two-times continuously differentiable with respect to ζ, then for each ε smaller than a given $\bar{\varepsilon}$, the function $\frac{\partial}{\partial\zeta}\alpha(\zeta,\varepsilon)$ is continuous together with its first derivative and hence Lipshitzian:

$$\left\| \left.\frac{\partial}{\partial\zeta}\alpha(\zeta,\varepsilon)\right|_k - \left.\frac{\partial}{\partial\zeta}\alpha(\zeta,\varepsilon)\right|_{k-1} \right\| \leq M\|(\zeta_k - \zeta_{k-1})\|. \tag{9}$$

Then, the mapping $T(\zeta_k,\varepsilon)$ satisfies the following inequality

$$\|T(\zeta_k,\varepsilon) - T(\zeta_{k-1},\varepsilon)\| \leq -\varepsilon N^{-1}M\|(\zeta_k - \zeta_{k-1})\| \tag{10}$$

and, being ζ in a normed Euclidean space (a Banach space), the contraction mapping Theorem applies [9] with ε sufficiently small. This implies $\zeta_k \to \zeta_\infty$ while $\sigma \to 0$ as $k \to \infty$. In particular, for $\varepsilon = 0$, eq. (7) is globally convergent in one step to

$$\zeta_\infty = \bar{\zeta}_\infty = -N^{-1}\beta. \tag{11}$$

The smooth dependence of ζ_∞ on ε has already been shown in Lemma 1.‡

Consider now the more general case that the system to be regulated is only partially known. In particular assume that the term $\varepsilon\lambda(\zeta, t, \varepsilon)$ is unknown. In this case, the gradient of the objective function cannot be exactly computed and can either be numerically approximated by repeated trials on the system, or estimated on the basis of the nominal linear part of the system. As already said, the first of these options is covered by existing numerical routines [8], and than we focus on the second one. An approximate gradient of the objective function can be computed at each trial by using the formula

$$\psi = \int_0^T \Psi'(t) y(t) dt, \tag{12}$$

and instead of (7), we can use the following updating rule

$$\zeta_{k+1} = \zeta_k - N^{-1}\psi_k. \tag{13}$$

where ψ_k denotes the values of the approximate gradient at the k-th trial.

Theorem 1 *For sufficiently small ε, the sequence (13) converges to a point $\hat{\zeta}_\infty$ smoothly dependent on ε, while $\psi_k \to 0$ as $k \to \infty$. The point of convergence $\hat{\zeta}_\infty$, is suboptimal in the sense that it continuously approaches the optimal one $\bar{\zeta}_\infty$ defined in Lemma 2, as $\varepsilon \to 0$.*

Proof. By substituting the expression of ψ in (13), we get

$$\begin{aligned} \zeta_{k+1} &= \zeta_k - N^{-1}(N\zeta_k + \beta + \int_0^T \Psi'(t)\varepsilon\lambda(\zeta, t, \varepsilon)dt) \\ &= -N^{-1}\beta - \varepsilon N^{-1}\hat{\alpha}(\zeta_k, \varepsilon) = \hat{T}(\zeta_k, \varepsilon) \end{aligned} \tag{14}$$

with

$$\hat{\alpha}(\zeta_k, \varepsilon) = \int_0^T \Psi'(t)\lambda(\zeta_k, t, \varepsilon)dt. \tag{15}$$

Now, since $\hat{\alpha}(\zeta_k, \varepsilon)$ is smooth, $\forall\zeta_0$ and for sufficiently small ε, the same steps showed in Lemma 2 can be followed for $\hat{T}(\zeta, \varepsilon)$ and then the convergence of the sequence $\{\zeta_k\}$ can be deduced. This also implies that $\psi_k \to 0$ as $k \to \infty$.

Last, since ψ depends smoothly on ε, by applying the Implicit Function Theorem [9] to the equation $\psi = 0$ in a neighborhood of $\varepsilon = 0$, we see that the solution smoothly depends on ε and this proves its suboptimality.‡

4 Robotic manipulators

Consider the class of systems known as rigid robot arms described by the equation

$$\begin{aligned} B(q(t))\ddot{q}(t) + n(q(t), \dot{q}(t)) = f(t) \\ t \in [0, T], \quad q(t) \in \Re^n, \quad q(0) = q_0, \quad \dot{q}(0) = \dot{q}_0 \end{aligned} \tag{16}$$

where $q(t)$ is the vector of the joint positions, $B(q(t))$ is the positive definite matrix of inertia, $n(q(t), \dot{q}(t))$ the vector of centripetal and gravitational forces, $f(t)$ the vector of the n joint forces, and define its output as

$$y(t) = q(t) - q_d(t) \tag{17}$$

with $q_d(t)$ the vector of joint position to be tracked. Say $D(q(t))$ the inverse of $B(q(t))$. $D(q(t))$ and $n(q(t), \dot{q}(t))$ have continuous partial derivatives up through order $h \geq 2$. On the time interval $[0, T]$, the function $q_d(t)$ and its derivatives may have a finite number of discontinuities. In the sequel, for sake of clarity, we denote by $\varpi(t)$ the vector of joint velocities $\dot{q}(t)$.

Now, close a feedback around joint velocities and set

$$f(t) = -\frac{1}{\varepsilon}(\varpi(t) - v(t)) \quad \varepsilon > 0 \tag{18}$$

with $v(t)$ the new control input. We have

$$\dot{q}(t) = \varpi(t) \tag{19}$$

$$\varepsilon\dot{\varpi}(t) = -D(q(t))[\varpi(t) - v(t) - \varepsilon n((q(t), \varpi(t)))] \tag{20}$$

$$\varpi(0) = \dot{q}_0 \quad q(0) = q_0$$

Sending $\varepsilon \to 0$, i.e., applying an high gain feedback, the robot arm system is singularly perturbed [9] and splits in a fast and slow system. The slow system is obtained by setting $\varepsilon = 0$

$$\dot{q}(t) = \varpi(t) \qquad q(0) = q_0 \tag{21a}$$

$$\varpi(t) = v(t). \tag{21b}$$

Assume that $v(t)$ in (20) has continuous first derivative, and let $q^*(t)$ be the smooth solution of the slow system (19). Let $\varsigma(t)$ denote the fast transient of $\varpi(t)$, with τ the fast time, such that $\varepsilon\tau = t$. The dynamics of $\varsigma(\tau)$ is obtained by replacing $\varpi(t)$ with $\varsigma(t) + \dot{q}^*(t)$ in the system dynamics and setting $\varepsilon = 0$

$$\frac{d\varsigma(\tau)}{d\tau} = -D(q^*(t))\varsigma(\tau) \quad \varsigma(0) = \dot{q}_0 - \dot{q}^*(0) \tag{22}$$

Since $D(q(t))$ is positive definite and lower bounded on $[0, T]$, the fast system is exponentially stable, uniformly w.r.t. $t \in [0, T]$, and Tikhonov's theorem [9] applies. This theorem states that for sufficiently small ε and $\varsigma(0)$, the following approximation holds

$$q(t) = q^*(t) + \mathcal{O}(\varepsilon) = q_0 + \int_0^t v(\vartheta)d\vartheta + \mathcal{O}(\varepsilon) \quad t \in [0, T] \tag{23}$$

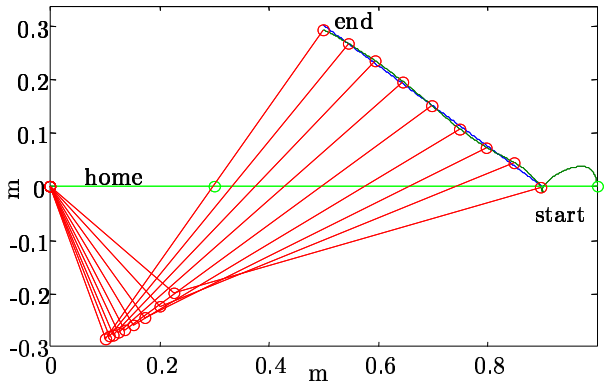

Figure 1: *Linear tip trajectory with* $s = 1$

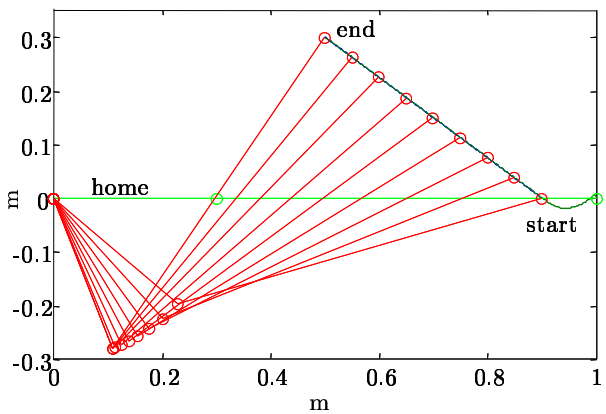

Figure 2: *Linear tip trajectory with* $s = 7$

where $\mathcal{O}(\varepsilon)$ is a term of order ε, i.e.,$\mathcal{O}(\varepsilon) < k|\varepsilon|$ for some positive constant k.

Now, to build the input $v(t)$ that minimise the norm of the output $y(t)$ defined in (17), i.e, able to steer the robotic system along the required output trajectory, and to transform the robotic system into the form (4), we dynamically extend system (21) by means of s integrators, letting

$$\frac{d^s v(t)}{dt^s} = 0 \quad \frac{d^i v(t)}{dt^i} = v^{(i)} \quad i = 0, ..., s-1. \tag{24}$$

Set

$$\zeta = (q_0, v_0^{(0)}, ..., v_0^{(s-1)}), \tag{25}$$

and consider equation (23): since

$$\int_0^t v(\vartheta) d\vartheta = \sum_{i=0}^{s-1} \frac{t^{i+1}}{i!} v^{(i)} \tag{26}$$

for sufficiently small $\varepsilon > 0$, the robot output takes the form of equation (4)

$$y(t, \zeta) = \Psi(t)\zeta - q_d(t) + \varepsilon\lambda(\zeta, t, \varepsilon) \tag{27}$$

with

$$\Psi(t)\zeta = q_0 + \sum_{i=0}^{s-1} \frac{t^{i+1}}{i!} v^{(i)}. \tag{28}$$

The function $\lambda(\zeta, t, \varepsilon)$ has continuous partial derivatives up through order $h > 2$ with respect to ζ and ε. Moreover it easy to see that the mapping

$$N = \int_0^T \Psi'(t)\Psi(t) dt \tag{29}$$

is invertible and then Theorem 1 applies.

Remark 1 The inversion of matrix N is a quite delicate task. Even if its invertibility is proven, increasing the dimension s of the exosystem leads rapidly to an ill-conditioned problem that can be only partially avoided making a different choice of the basic functions generated by the exosystem (now polinomials using (24)).

Remark 2 The procedure introduced requires the initialisation of the system to an arbitrary point ζ of some open set of the state space. To overcome difficulties that may arise for some physical systems, in particular when non zero initial conditions must be achieved for the first n components of vector ζ, an iterative steering algorithm for robotic systems can be implemented as in [10]. This task can be accomplished only through the control input $v(t)$ and, following the approach of [1], it is possible to combine the two learning processes in a single learning algorithm, implementing both initial state steering and search of the correct initial condition.

5 Simulations and experimental results

To show the feasibility of the proposed algorithm, some simulations have been performed using the model of an experimental open chain planar arm with two rigid links and two revolute joints. Moreover, some experiments have been carried on using the real arm. The physical data of the robot are respectively for the first and second link: lengths equal to 0.3 m and 0.7 m, moments of inertia $J_1 = 0.285\,\mathrm{Kg\,m^2}$ and $J_2 = 0.3033\,\mathrm{Kg\,m^2}$, static moments 0.9801 Kg m and 0.65 Kg m, and the mass of the second link is equal to 1.8571 Kg. Each joint is actuated by a direct drive DC motor and is equipped with an encoder and a tachometer. The encoders resolution is equal to $\pi/10000$ rad.

The robot is digitally controlled by means of a personal computer using a sampling frequency of 200 Hz for each signal. Analog feedbacks from the tachometers signals are closed at the joints and, in addition, a proportional derivative feedback has been digitally closed around the angular positions. An integrator

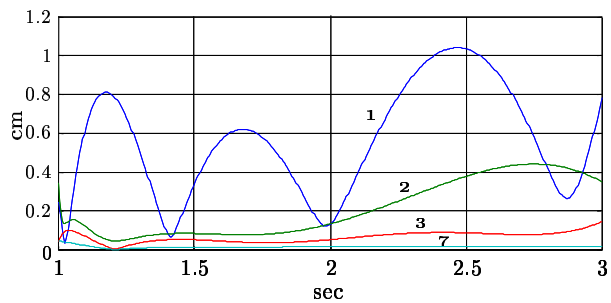

Figure 3: *Linear trajectory error for* $s = 1, 2, 3, 7$

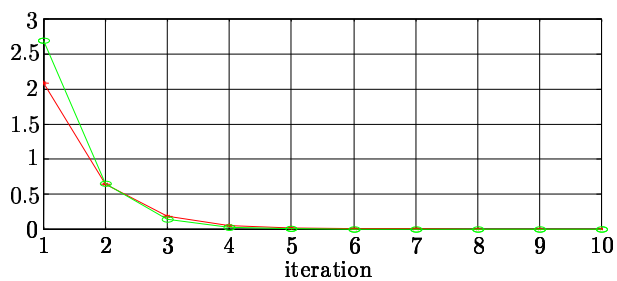

Figure 4: *Index error for linear tip trajectory and* $s = 7$

for each channel has been added to smooth the new control signal. The overall values of the PD controller gains are $K_p = (7, 2)$ Kg m for the proportional part and $K_d = (2, 4)$ Kg m sec for the derivative one. The arm has been tilted of approximately 10 degrees to introduce a nonlinear perturbation due to gravity effect that has been considered also for simulation. By means of the known robot inverse kinematics, two reference joint trajectories have been generated starting from a tip reference trajectory. For both links, the dynamic extension used is a chain of integrators.

In the first simulation, the robot has to track, with

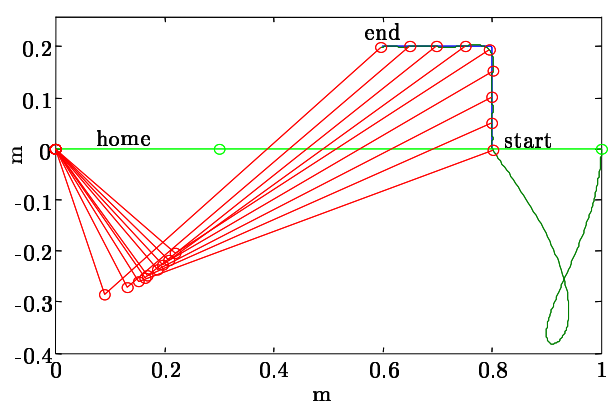

Figure 5: *Non-smooth tip trajectory with* $s = 9$

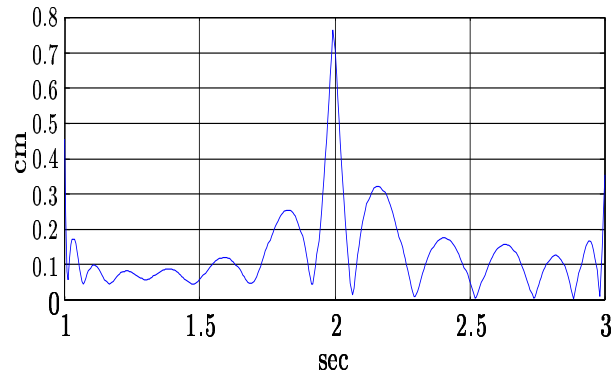

Figure 6: *Non-smooth tip trajectory error for* $s = 9$

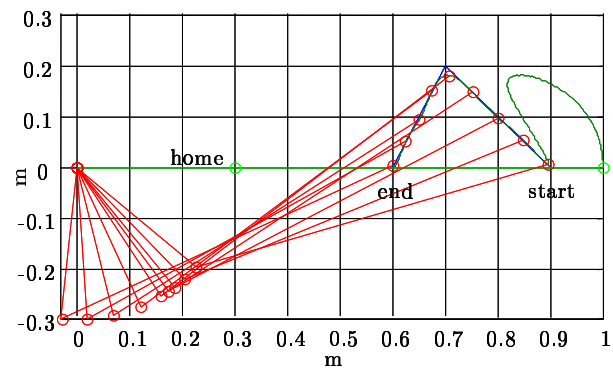

Figure 7: *Experiment with* $s = 7$

constant velocity, a linear reference tip trajectory over a time interval of 2 seconds. Robot's tip is requested to move between the points $(0.9, 0)$ and $(0.5, 0.3)$. An initial time interval of one second is allowed to steer the robot from home $(1, 0)$ to the correct initial state at $t = 0$. A different behavior of the algorithm is obtained according to the dimension of the dynamical extension (24). In Figs. 1 and 2 the trajectories obtained at the 10-th iteration with $s = 1$ and $s = 7$ respectively are shown. Note that increasing the dimension of the exosystem, i.e., the number of linear modes that can be internally generated, the algorithm shows a better performance that is confirmed by Fig. 3 where, for different values of s, the cartesian error for the tip trajectory is reported. In particular, to show that the convergence is geometric, referring to the $s = 7$ case, in Fig. 4 the norm of ψ, as defined in (12), is reported for each link along 10 iterations. Note that the trajectory, during the initialisation phase, can change in shape due to the different dimensions of the dynamic extensions.

Another simulation has been performed using a non-smooth tip reference trajectory build with two linear segments showed in Fig. 5. Velocity along the

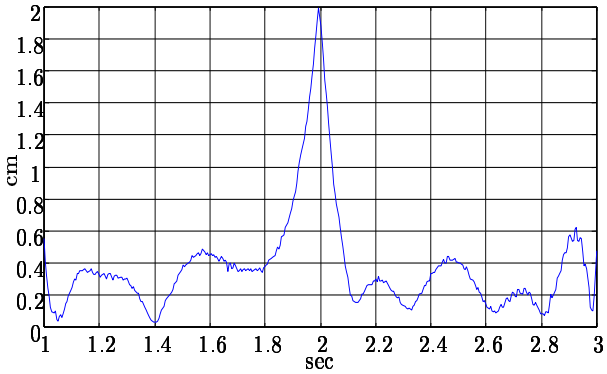

Figure 8: *Experiment trajectory error with* $s = 7$

path has constant absolute value of 0.2m/s and another second has been allowed to reach the initial condition starting from the tip home position $(1, 0)$. In the same Figure the robot trajectory at 10-th iteration with $s = 9$ is reported and, in Fig. 6, the trajectory error is shown.

Finally, by using the laboratory set up above described, the algorithm has been also experimentally tested. The robot performance while attempting to track the reference tip trajectory already introduced is illustrated in the graphs reported in Figs. 7 and 8. Note that, even in the real case, the tracking is better where the trajectory can be generated by the exosystem getting worse approaching the speed discontinuity.

6 Conclusions

The paper deals with quasi-linear systems and shows how, through an appropriate state extension, and the use of an iterative learning algorithm, it is possible to track a piece-wise continuous output trajectory minimising a suitable norm of the error. Future work will be devoted to analyse both the relation between state extension and error minimisation, and to better understand the condition under which sub-optimal procedure becomes optimal in the fully linear case. Finally, a deeper investigation will be devoted to the design of an opportune state extension to avoid ill-conditioning in the computation of mapping N.

References

[1] P. Lucibello. "Output Zeroing whit Internal Stability by Learning," *Automatica*, vol. 31, no. 11, pp. 1665–1672, 1995.

[2] S. Arimoto, S. Kawamura, F. Miyazaki, and S. Tamaki, "Learning control theory for dynamical systems," *Proc. 24th IEEE Conf. on Decision and Control,* Ft. Lauderdale, Florida, Dec. 1985.

[3] J. Hauser, "Learning control for a class of nonlinear systems," Proc of 26th Conf. on Decision and Control, Los Angeles, California, 1987, pp. 859-860.

[4] T. Sugie and T. Ono, "An iterative learning control law for dynamical systems," Automatica, vol. 27, pp. 729-732, 1991.

[5] G. Heinzinger, D. Fenwick, B. Paden, and F. Miyazaki, "Stability of learning control with disturbances and uncertain initial conditions," *IEEE Trans. Automatic Control,* vol. 37, no. 1, January 1992, pp. 110-114.

[6] H.S. Ahn, C.H. Choi, and K.B. Kim. "Iterative learning control for a class of nonlinear systems," *Automatica*, vol. 29, n. 6, pp. 1575-1578, 1993.

[7] P. Lucibello, S. Panzieri, "Experiments on output tracking with internal stability by learning for a one-link flexible arm," *Automatica*, vol. 33, n. 11, pp. 2065-2069, 1997.

[8] D.G. Luenberger, "Linear and nonlinear programming," Second edition, *Addison-Wesley Publishing Company, Inc.*, Reading, Massachusetts, 1984.

[9] H.K. Khalil, "Nonlinear systems," Third edition, *Prentice Hall*, NJ, 1996.

[10] P. Lucibello, S. Panzieri, G. Ulivi. "Repositioning control of a two-link flexible arm by learning," *Automatica*, vol. 33, n. 4, pp. 579–590, 1997.