Algebraic On-Line Identifier for Visual Servoing of Robot Manipulators.

†E.C. Dean-León †H. Sira-Ramírez †V. Parra-Vega
‡A.Espinosa-Romero
†Mechatronics Division - CINVESTAV, México.
Email: {hsira,vparra,edean}@mail.cinvestav.mx
‡Institute for Applied Mathematics and Systems - UNAM, México.
Email: arturoe@cic3.iimas.unam.mx.

Abstract

Differential kinematics model, of the CCD coordinates and generalized robot joint coordinates, has been usually considered to design position-based visual feedback kinematic controllers for kinematic tracking of planar manipulators under fixed-camera configuration. Nevertheless if the intrinsic camera parameters are unknown then some sort of calibration must be done (either on-line or off-line). In this paper, we propose an algebraic scheme to derive a visual servoing controller that ensures global tracking, even when camera orientation angle is $\theta = \pi/2$. The most important feature of this approach is the exact identifier of the unknown parameters of camera, which are used to achieve close-loop linearization, with exponential convergence. Thus the implementation does not need knowledge neither camera orientation nor depth of field parameters. The evaluation of the performance of the controller and the identifier is done with simulations of 2DoF serial open kinematic chain, and its dynamic model using state feedback joint PD. Robot and camera parameters ar taken from a real system.

1 Introduction

Visual servoing has been formally studied from different research areas, such as the control community, the computer science community and the robotics community, in particular. Lot of papers with quite different and novel schemes are proposed. However, only very few proposals produce formal stability results for the real, and relevant, case of tracking with uncalibrated camera with on-line calibration system. Formal and rigorous stability results are required to stand any test, that is to guarantee *a priori* a given closed-loop performance of robotic systems driven by visual information. it seems that the main problem to address on-line calibration is the fact that parameters enter nonlinearly into the kinematic model between robot and camera models. Thus standard adaptive techniques cannot be used.

In this paper, we propose a rather different approach based on an algebraic procedure for on-line uncertain parameter identification recently proposed, and completely justified from a module theory viewpoint, in the work by Fliess and Sira-Ramírez (see [2]). In this way, exact parameters are obtained and inverse kinematic control can be easily derived to obtain global exponential tracking.

To prove the performance of the proposed controller at the kinematic and dynamic level, we present simulations on a 2 DoF planar robot, which confirms the expected stability properties for trajectory tasks of planar kinematic serial chains. Then, we consider tracking for a (dynamical) robot manipulator using PD controller. Our controller shows superior performance with respect to the only known formal kinematic control scheme for uncalibrated camera [4], and in fact our controller stands as the first globally exponentially stable uncalibrated visual servoing controller for robot arms. The reason is that [4] is a regulator for autonomous tasks, not for tracking tasks and proves only local stability.

The paper is organized as follows. Section 2 introduces the differential kinematic model under consideration and the static visual mapping. Section 3 presents the control law synthesis and design. Section 4 shows the algebraic identifier of unknown parameters. The simulation conditions and comparative results are discussed in Section 5, and finally, conclusions are offered in Section 6.

2 The Robotic System Model and Vision System

The robotic system is modelled by a serial link kinematic robot manipulator equipped with joint position and velocity sensors, with a vision system in fixed (static) camera configuration as follows.

2.1 Join to Cartesian transformation

Denote by x_B the position of the robot end-effector with respect to the base frame. From the *forward kinematics*, we have

$$x_B = f\left(q\right) \tag{1}$$

where $q \in R^2$ denotes the generalized joint angles of the manipulator, and $f(q) : \Re^2 \rightarrow \Re^2$ stands for its direct kinematic mapping.

2.2 Kinematic of the robotic model

Considering a 2 DoF planar robot we denote tha plane where the motion of the robot end-effector takes place with a two dimensional Cartesian coordinate frame, called *the robot coordinate frame*, labeled R_1-R_2. The position of the robot end-effector on the plane with respect to the *robot coordinate frame* is denoted by $x_R = \left[\begin{array}{cc} x_{r_1} & x_{r_2} \end{array}\right]^T \in \Re$.

A Digital camera (CCD type) providing an imagen of the whole robot workspace is placed perpendicular to the plane where the robot evolves (see Fig.1) The optical center is located ata distance z with respect to the R_1-R_2 plane, and the intersection between optical axis and the R_1-R_2 robot workspace is denoted by $O_R = \left[\begin{array}{cc} O_{r_1} & O_{r_2} \end{array}\right]^T$. The orientation of the camera around the optical axis measured clockwise is denoted by θ (see Fig.1)

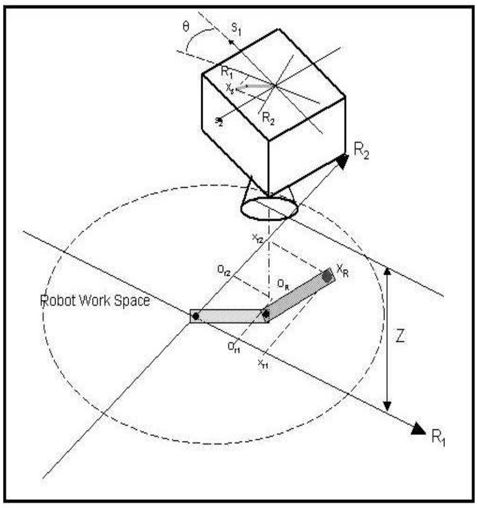

Figure 1: Cordinate Frames

The camera provides the image of the scene on the CCD discrete array of sensors elements called *the front image plane*; this image may have some deformation due to lens distorsion that we considered despicable. Finally, this image is later stored in the computer frame buffer being available on the computer screen. On this computer screen we define a two dimensional coordinate frame, called *screen coordinate frame* or *computer image coordinate system* labeled S_1-S_2. The description of robot end-effector in the screen coordinate frame is denoted by $x_S = \left[\begin{array}{cc} x_{s_1} & x_{s_2} \end{array}\right]^T$.

The description of the robot end-effector $x_R = \left[\begin{array}{cc} x_{r_1} & x_{r_2} \end{array}\right]^T$ in the robot coordinate frame, based on the *prespective projection model*, is given in terms of the screen coordinate frame as[1] (see Fig.1):

$$\left[\begin{array}{c} x_{s_1} \\ x_{s_2} \end{array}\right] = \alpha_0 h \cdot R\left(\theta\right) \left\{ \left[\begin{array}{c} x_{r_1} \\ x_{r_2} \end{array}\right] - \left[\begin{array}{c} O_{r_1} \\ O_{r_2} \end{array}\right] \right\} + \left[\begin{array}{c} c_x \\ c_y \end{array}\right] \tag{2}$$

where $\left[\begin{array}{cc} c_x & c_y \end{array}\right]^T$ is the image center[1], α_0 is the scale factor of length units in the front image plane

[1] For a detailed procedure to obtain the explicit relationship see for instance [5].

given in *pixels/m* which is assumed to be negative, h is the magnification factor defined as

$$h = \frac{\lambda}{\lambda - z} < 0$$

and, $R(x) \in SO(2)$ is the rotation matrix generated by clockwise rotating the camera about its optical axis by x radians:

$$R(x) = \begin{bmatrix} \cos(x) & -\sin(x) \\ \sin(x) & \cos(x) \end{bmatrix} \quad (3)$$

From equations (1) and (2) we have the description of the robot end-effector, with respect to the base frame, given in terms of the screen coordinate frame

$$x_{sB} = \alpha_0 h \cdot R(\theta)\, x_B - \alpha_0 h \cdot R(\theta)\, O_R + C$$

and defining

$$\alpha = \alpha_0 h$$
$$\beta = -\alpha_0 h \cdot R(\theta)\, O_R + C$$

we have

$$x_{sB} = \alpha R(\theta)\, f(q) + \beta \quad (4)$$

When any intrinsic parameters, such as the depth of field z, camera position offset O, and/or focal length λ are unknown, it is said that the camera is not calibrated (that is α, θ, and β are unknown), which is a more relevant problem for visual servoing since usually these parameters are hardly known in any practical implementation.

2.3 Differential Kinematics

By differentiating equation (1), we obtain

$$\dot{x}_B = J(q)\, \dot{q} \quad (5)$$

where $\dot{x}_B$ is the velocity of the end-effector with respect to the base frame, $J(q)$ is the analytic Jacobian matrix of the manipulator, and $\dot{q}$ is the (generalized) joint velocities. Thus, by using equation (5), the differential kinematic model of (4) becomes

$$\dot{x}_{sB} = \alpha R J(q)\, \dot{q} \quad (6)$$

where $\dot{y} \in R^2$ denotes the velocity of the end-effector with respect to the image frame.

2.4 Robot Dynamics

In the absence of friction or other disturbances, the dynamics of a serial n-link rigid, non-redundant, fully actuated robot manipulator can be written as follows[2].

$$H(q)\, \ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau$$

where $q \in \Re^n$ is the vector of join displacements, $\tau \in \Re^{n\times 1}$ stands for the vector applied joint torques, $H(q) \in \Re^{n\times n}$is the symmetric positive definite manipulator inertia matrix, $C(q, \dot{q}) \in \Re^n$ stands for the vector of centripetal and Coriolis torques, and finally $G(q) \in \Re^n$ is the vector of gravitational torques, in our case this vector is $G(q) = [0] \in \Re^n$.

3 Control Law Synthesis and Design

In the next section we explain the control law synthesis and its design.

3.1 Visual kinematic control paradigm

It is customary for kinematic control for robot arms to consider the differential kinematic model (5) to design a kinematic controller, in order to design the control law, equation (6) is rewritten in terms of a forced (controlled) input u as follows

$$\dot{x}_{sB} = \alpha R u \quad (7)$$

where $u \equiv J(q)\, \dot{q}$ such that the kinematic control problem is to design u to guarantee convergence of (7) to any smooth trajectory $x_d(t) \in C^2$. Notice that the following feedback linearization controller

$$u = R^{-1}\alpha^{-1}v, \quad (8)$$

for $v = \dot{x}_{sd} - \kappa\Delta x_s$, with $\Delta x_s = x_{sB} - x_{sd}$, $\kappa > 0$, yields

$$\Delta\dot{x}_s = -\kappa\Delta x_s, \quad (9)$$

which guarantee $x_{sB} \to x_d, \dot{x}_{sB} \to \dot{x}_{sd}$ globally exponentially. However, if α and θ are unknown, then (8) cannot be implemented.

[2] Without loss of generality, our controller can be applied with similar results if we consider dynamic friction, for instance the LuGre model.

3.2 Implementation on a dynamic robot

Once we obtain the controller (8), this kinematic controller becomes the desired joint trajectory, that is

$$\dot{q}_d = J(q)^{-1} u \qquad (10)$$

for the kinematic robot. At the dynamic level, we can design a fast velocity tracking controller τ for the real (dynamic) robot using (10) as the desired trajectories.

3.3 Control Law

A visual based control of a two link planar manipulator (using (3), (7)), partially controlled by a fast velocity feedback loop, can be formulated as follows (see Hsu and Aquino [1]):

Given the visual flow dynamics (7),

$$\begin{array}{rl} \dot{x}_{s1} = & \alpha\left(\cos(\theta)u_1 - \sin(\theta)u_2\right) \\ \dot{x}_{s2} = & \alpha\left(\sin(\theta)u_1 + \cos(\theta)u_2\right) \end{array} \qquad (11)$$

and a desired reference trajectory, $x_{sd}(t) = (x_{s1d}(t), x_{s2d}(t))$ *in the visual frame space, find the control input vector* $u = (u_1, u_2)$ *such that* $x_s(t) \rightarrow x_{sd}(t)$*, regardless of the unknown, but constant, values of the real parameters,* α *and* θ*.*

3.4 A certainty equivalence PI controller

Assuming that the parameters are known (certainty equivalence principle) for (11), a global trajectory tracking controller of the PI type for a two DoF robot armis is given by

$$\begin{array}{rl} u_1 = & \alpha^{-1}(\cos(\theta)v_1 + \sin(\theta)v_2) \\ u_2 = & \alpha^{-1}(-\sin(\theta)v_1 + \cos(\theta)v_2) \end{array} \qquad (12)$$

with

$$v_1 = \dot{x}_{s1d}(t) - k_{11}\Delta x_{s1} - k_{01}\int_{t_0}^{t} \Delta x_{s1}(\sigma))d\sigma$$

$$v_2 = \dot{x}_{s2d}(t) - k_{22}\Delta x_{s2} - k_{02}\int_{t_0}^{t} \Delta x_{s2}(\sigma))d\sigma$$

where $\Delta x_{si} = x_{si} - x_{sid}(t)$, $i = 1, 2$. The closed loop tracking error dynamics $\Delta x_s = (\Delta x_{s1}, \Delta x_{s2}) = (x_{s1} - x_{s1d}(t), x_{s2} - x_{s2d}(t))$ is seen to be given by the linear vector equation

$$\Delta\dot{x}_s + K_1\Delta x_s + K_0\int_{t_0}^{t} \Delta x_s(\sigma)d\sigma = 0 \qquad (13)$$

where $K_1 = \mathrm{diag}(k_{11}, k_{22})$ and $K_0 = \mathrm{diag}(\mathrm{k}_{01}, \mathrm{k}_{02})$ are diagonal matrices. The error vector trajectories can thus be made to globally exponentially asymptotically approach the origin of R^2 for a suitable set of design matrix coefficients, K_0, and K_1. To be able to implement the controller u of equation (12), now we propose an scheme to identify α and θ.

4 Calibration: Algebraic identifier for parameters θ, α

Notice that the differential kinematics model $\dot{x} = (\dot{x}_1, \dot{x}_2)$ of (7) can be written in terms of the complex variable $j = \sqrt{-1}$ as follows

$$\dot{z} = \alpha e^{j\theta} v \qquad (14)$$

for

$$z = y_1 + jy_2$$
$$v = u_1 + ju_2$$
$$e^{j\theta} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$$

Now, for convenience we write (14) as follows

$$\dot{z} = \xi v \qquad (15)$$

and let $\xi = \alpha e^{j\theta}$ be a vector with the components, $\xi_1 = a\cos(\theta)$ and $\xi_2 = a\sin(\theta)$. Knowledge of ξ implies that we can uniquely (modulo a multiple of 360^o degrees rotation of the camera) determine the unknown parameters α and θ as

$$\begin{array}{rl} \alpha = & \sqrt{\xi_1^2 + \xi_2^2}, \\ \theta = & \arctan(\frac{\xi_2}{\xi_1}) \end{array} \qquad (16)$$

In the next few lines, we follow an algebraic procedure for on-line uncertain parameter identification recently proposed, and completely justified from a module theory viewpoint, in the work by

Fliess and Sira-Ramírez (see [2]). This algorithm allows to obtain exactly ξ_1 and $\xi 2$, and using (16), we finally obtain α, θ. In this way, we can implement (12) without any knowledge of depth of field z, nor focal length λ, nor rotation angle θ.

4.1 Calibration algorithm [2]

We sketch the novel identification algorithm proposed by :

1. Take the Laplace transform in equation (15)

$$s\hat{z} - z_0 = \xi\hat{\nu} \tag{17}$$

2. Differentiate equation (17) once with respect to the complex variable s

$$s\frac{d\hat{z}}{ds} + \hat{z} = \xi\frac{d\hat{v}}{ds} \tag{18}$$

3. Find the inverse Laplace transform of the resulting expressions (equation 18)

$$-\frac{d}{dt}\left[(t-t_0)\,z\right] + z = -\xi\left[(t-t_0)\,v\right] \tag{19}$$

4. Integrate equation (19) once and rearranging we obtain the complex uncertain parameter ξ, which we now denote as $\hat{\xi}$

$$\hat{\xi} = \frac{(t-t_0)\,z - \int_{t_0}^{t} z(\sigma)\,d\sigma}{\int_{t_0}^{t}(\sigma - t_0)\,v(\sigma)\,d\sigma}$$

5. Obtain the uncertain parameter vector components ξ_1, ξ_2 in time domain

$$\begin{bmatrix} \hat{\xi}_1 \\ \hat{\xi}_2 \end{bmatrix} = \begin{bmatrix} \int_{t_0}^{t} u_1(\sigma)d\sigma & -\int_{t_0}^{t} u_2(\sigma)d\sigma \\ \int_{t_0}^{t} u_2(\sigma)d\sigma & \int_{t_0}^{t} u_1(\sigma)d\sigma \end{bmatrix}^{-1} * \begin{bmatrix} (t-t_0)y_1(t) - \int_{t_0}^{t} y_1(\sigma)d\sigma \\ (t-t_0)y_2(t) - \int_{t_0}^{t} y_2(\sigma)d\sigma \end{bmatrix} \tag{20}$$

6. Compute α and θ of (16)

7. Compute the controller (12)

8. Finally, exponentially stable equation (13) appears

Remark 1 *Once the vector components ξ_1 and ξ_2 are obtained by accurately evaluating the previous expressions after a time interval of the form, $[t_0, t_0 + \delta]$, has elapsed, with δ being an arbitrarily small strictly positive real number, we use such an on-line computed value, $\widehat{\xi}$, as the real $\widehat{\xi}$ to compute α and θ of equation (16) and substitute them into the proposed PI certainty equivalence controller (12).*

Remark 2 *Note that the constant value of the vector $\widehat{\xi}$ needs to be computed only* once. *Thus, after an accurate computation of this vector has been obtained, we may opt to "switch off" the identifier, right after time $t = t_0 + \delta$, and indefinitely use this value in the controller. However, in many other applications, the uncertain parameter may suddenly change to a new constant value. In such cases, the algebraic identifier may then be used on a repetitive fashion by re-initiating the computations with a new "initial time" t_0. In this manner, the certainty equivalence controller can be properly updated. If, on the other hand, the identifier is left "on" for an indefinite period of time, opting for a continuous updating of the controller uncertain parameters, then one should take provisions to avoid divisions by zero when the uncertain parameter factor crosses this singular value. See [3] for further details.*

5 Simulations

A serial-link rigid robot arm, being built at the institute, whose base frame is located at $(-0.666, -0.333)$ m with respect to vision frame (parameters are those of the Sony DWF-V500) and a thin, fixed iris, no focus World Optics CCTV lens without aberration are considered. The endpoint of the manipulator is requested to draw a circle of radius of 0.1 m in 6 s centered at $(-0.3, 0.1)$ m of R_1-R_2. One alternative controller, with rigorous stability proof, that deals with kinematic visual servoing without knowledge of α nor θ is the controller proposed recently by [4]. Therefore, we compare our controller versus [4] for tracking tasks, for two cases: the pure kinematic case (inverse kinematics) and the dynamic case, for the later we use a joint PD controller.

These parameters yield $\alpha = -402.14 pixel/m$, and $\beta = -60.2 pixel/m$.

	Parameter	Value
Link length 1	l_1	0.300m
Center of mass 1	l_{c1}	0.21m
Link length 1	l_2	0.210m
Center of mass 1	l_{c2}	0.18m
Moment of inertia 1	I_1	10.1kg
Moment of inertia 2	I_2	3.1kg
Mass 1	m_1	10.1kg
Mass 2	m_2	3.1kg
Depth of field	z	1.5m
Focal Length	λ	0.008m
Scaling	α_0	72,200 p/m
Camera offset	c_x, c_y	0, 0 pixel
Center offset	O_R	0.1 m
Angle	θ	30°

Table 1: Visual servo robot system

5.1 Results for the kinematic case

Figure 2 shows the position and velocity kinematic tracking errors, wherein exponential convergence for time $t \geq \delta$, for δ equal 3 integrations steeps of $\delta t = 0.001\ s$. Control input u is shown is figure 3. The performance with respect to kinematic tracking errors and kinematic control effort of [4] is not presented separately, because those can be seen in the next figures.

5.2 Results for the dynamic case, with PD control

The desired trajectories obtained in the previous subsection are now used, premultiplied by the J^{-1}, as desired trajectories at the joint level of a PD controller for the dynamic model of a two DoF robot manipulator, whose parameters are also given in Table 1. Figures 4 and 5 show the dynamic response of the PD controller, notice the transient response of [4], in contrast to the smooth, exponentially decrescent, response of our approach. Figures 6 and 7 plot the screen coordinates tracking, and 8 9 the tasks space coordinates.

6 Conclusions

In this note, we have presented a position-based visual servoing controller for kinematic robot manipulators under the assumption that all intrinsic parameters are unknown. An on-line algebraic identifier operator is proposed to find the exact vision parameters and thus, an inverse kinematic controller is proposed to achieve exponential tracking. The closed-loop performance is evaluated at the kinematic and dynamic level, and compared against another formal control scheme. The main characteristic of our approach is its simplicity of the control structure with the low computational cost and the fact that renders smooth trajectories, which are very relevant for dynamic control.

References

[1] Hsu, L., and Aquino, P., "Adaptive visual tracking with uncertain manipulator dynamics and uncalibrated camera" *Proc. 38th IEEE Conference on Decision and Control*, pp. 1248-1253, Phoenix, December 1999.

[2] M. Fliess and H. Sira-Ramírez, "An algebraic framework for linear identification" *ESAIM Contol and Calculus of Variation* (submitted for publication)

[3] H. Sira-Ramírez, E. Fossas and M. Fliess, " On the control of an uncertain, double bridge buck converter: An algebraic parameter identification approach" IEEE Conference on Decision and Control (submitted for publication)

[4] Astoli, A, Hsu, L., Ortega, R, and Netto, M. A Solution to the Adaptive Visual Servoing Problem, Proc. of the Int. Conference on Robotics and Automation, 742-77, 2002

[5] Kelly, R., Shirkey, P., and Spong, M.W., "Fixed Camera Visual Servo Control for Planar Robots", Proc. of the 1996 IEEE Int. Conf. on Robotics and Automation, Minnesota - April 1996.

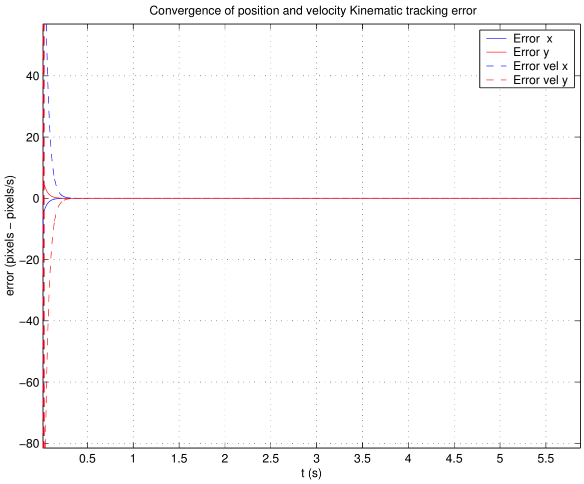

Figure 2: Position and velocity kinematic tracking errors using our controller: Global exponential convergence even for $\theta = \frac{\pi}{2}$.

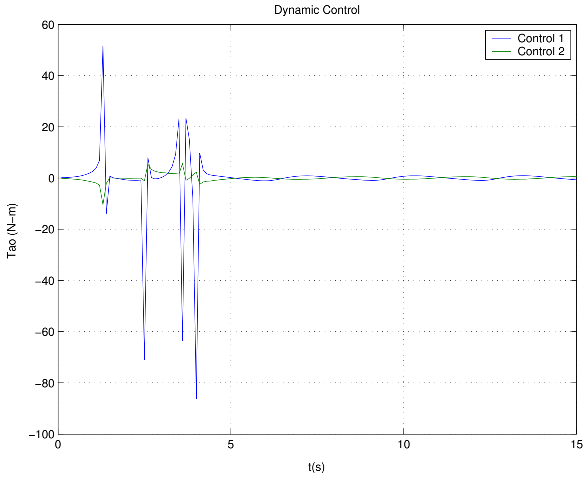

Figure 4: Dynamic performance of [4] with PD controller

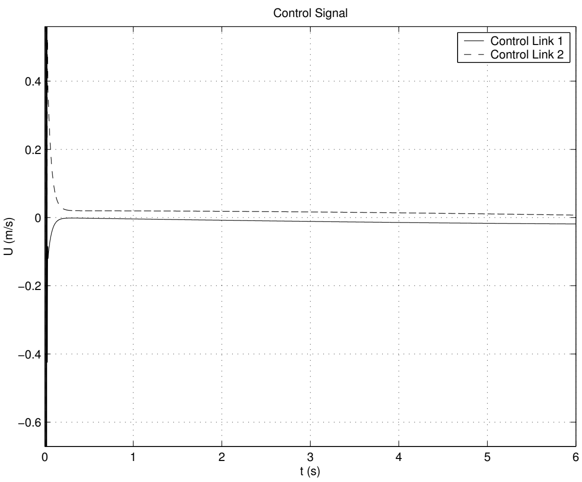

Figure 3: Control inputs for drawing a circle ($\frac{\mathrm{m}}{\mathrm{s}}$).

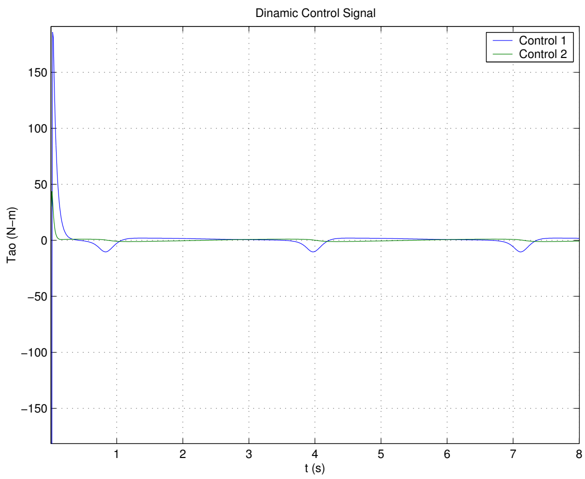

Figure 5: Dynamic performance of our approach with PD controller

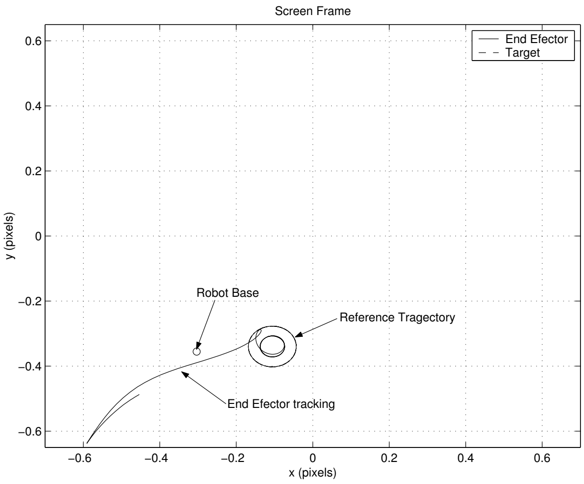

Figure 6: Tracking in screen coordinates with PD control: [4]. Scales are $\times 10^3$

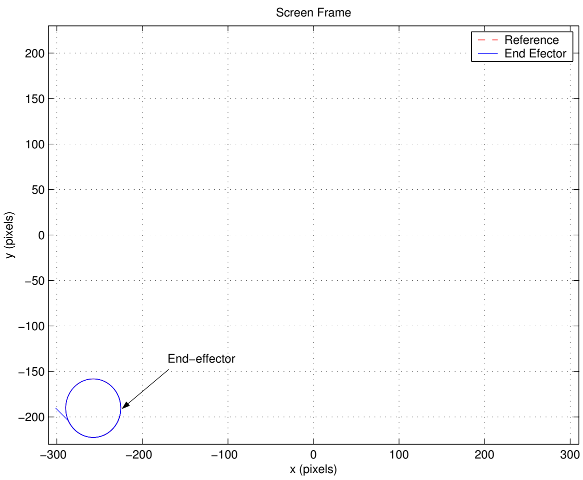

Figure 7: Tracking in screen coordinates with PD control: Our controller

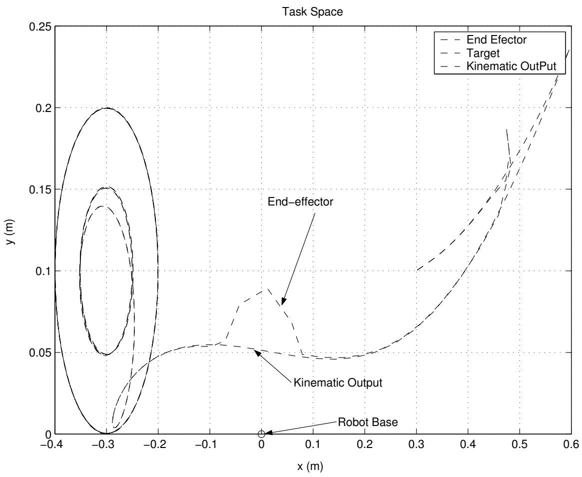

Figure 8: Tracking in task coordinates with PD control: [4].

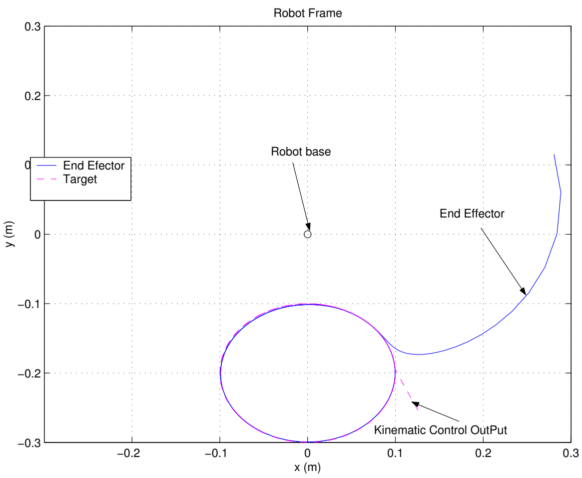

Figure 9: Tracking in task coordinates with PD control: our controlles