

Robust Path Following for Robot Manipulators

Rajan Gill Dana Kulić Christopher Nielsen

Abstract—Path following controllers make the output of a control system approach and traverse a pre-specified path with no *a priori* time-parametrization. This paper implements a path following controller, based on transverse feedback linearization (TFL), which guarantees invariance of the path to be followed. The coordinate and feedback transformation employed allows one to easily design control laws to generate arbitrary desired motions on the path for the closed-loop system. The approach is applied to an uncertain and simplified model of a robot manipulator for which none of the dynamic parameters are measured. The controller is made robust to modelling uncertainties using Lyapunov redesign. The robustified controller is tested on a 4-degree-of-freedom (4-DOF) manipulator with a combination of revolute and linear actuated links. The experimental results show a substantial improvement when using the robust controller for path following versus standard state feedback.

I. INTRODUCTION

The problem of requiring a robot to follow a desired motion can be broadly classified as either a trajectory tracking or path following problem. Trajectory tracking consists of tracking a time-indexed set of reference points. This approach may not be suitable for certain applications as it may be unnecessarily demanding or infeasible. Consider an aircraft tracking a desired reference trajectory. If at any point the aircraft falls behind in tracking the reference, due to wind, then the aircraft will need to increase its thrust in order to “catch up” to the time-indexed trajectory and this can lead to stall [1]. Similarly, for a robot tracing a desired contour, lag in following a desired reference signal could cause the end-effector to “cut through” an obstacle with excessive control input.

Path following seeks feedback controllers that drive the system’s output to a desired path and traverses it without *a priori* time parametrization. Thus, path following controllers stabilizes all trajectories whose associated output lies on the desired path whereas trajectory tracking controllers do this for a single trajectory. In general, path following control results in smoother convergence to the desired path compared to trajectory tracking control laws, and the control signals are less likely to saturate [2]. It is also shown in [3] that for systems that are non-minimum phase, path following controllers remove performance limitations as opposed to trajectory tracking controllers which have a lower bound on the achievable \mathcal{L}_2 -norm of the tracking error.

There are several approaches for implementing path following. The papers [4]–[6] used an approach of projecting

the current system state onto the desired path in real time. Then a tracking controller is used to stabilize this generated reference point. This allows the reference motion to be altered depending on the system state. These results have been applied to robot manipulators [7]. However, this approach does not guarantee that the desired path will be invariant, i.e., if for some time the output is on the path and has a velocity tangent to the path, then the output will remain on the path for all future time.

Virtual holonomic constraints can be used to guarantee invariance of a path via set stabilization. This methodology is used for orbital stabilization for a Furuta pendulum [8], helicopters [9], and walking/running of bipedal robots [10]. Also, [11] introduces conditions for which feedback linearization of the transverse dynamics associated with periodic orbits can be done. This inspired the work in [12] in which an arbitrary non-periodic path in the output space can be made attractive and invariant using transverse feedback linearization (TFL). This idea was further extended in [13] by providing conditions for which desired motions can be achieved on the path; as opposed to orbital stabilization where a limit cycle is forced in the closed loop system. This simplifies the controller design and decomposes the path following design problem into two independent sub-problems: staying on the path and moving along the path (summarized in Section II). However, in [13], perfect knowledge of the dynamic parameters of the system was assumed, and performance was demonstrated only for a linear, decoupled system. In this paper, the TFL approach is applied to a nonlinear, coupled manipulator system, and a robust controller is developed to overcome modelling inaccuracies. The experimental platform is modelled in Section III, with the control law described in Section IV. Experimental results are shown in Section V.

A. Notation

Let $\text{col}(x_1, \dots, x_k) := [x_1 \ \cdots \ x_k]^\top$ where \top denotes transpose and, given two column vectors a and b , we let $\text{col}(a, b) := [a^\top \ b^\top]^\top$. The notation $s \circ h : A \rightarrow C$ represents the composition of maps $s : B \rightarrow C$ and $h : A \rightarrow B$. The i th element of a vector is denoted by the subscript $[\cdot]_i$, and $[\cdot]_{i:j}$ denotes the sub-vector from the i th to j th element. The Euclidean norm of a vector and induced norm of a matrix are both denoted by $\|\cdot\|$. Let $\arg : \mathbb{C} \rightarrow (-\pi, \pi]$ represent the principal argument of a complex number $z = x + iy$.

II. PATH FOLLOWING FOR MECHANICAL SYSTEMS

In this section we review the path following control design methodology from [13] which applies to a large class of mechanical systems. In Section IV we show that our

This research is partially supported by the Natural Sciences and Engineering Research Council of Canada (NSERC).

The authors are with the Dept. of Electrical and Computer Engineering, University of Waterloo, Waterloo ON, N2L 3G1 Canada. {rjgill; dkulic; cnielsen}@uwaterloo.ca

experimental platform, a 4-DOF manipulator with revolute joints driven by linear actuators, falls into this class.

A. TFL for Mechanical Systems

Consider a mechanical system with N degrees of freedom. Let $(x_c, x_v) \in \mathbb{R}^N \times \mathbb{R}^N$ and $x := \text{col}(x_c, x_v) \in \mathbb{R}^{2N}$ where $n = 2N$. Then, under mild assumptions, the mechanical system can locally be expressed in the form:

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ &:= \begin{bmatrix} x_v \\ f_v(x) \end{bmatrix} + \begin{bmatrix} 0 \\ g_v(x_c) \end{bmatrix} u, \end{aligned} \quad (1)$$

where, $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$, $g : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times m}$ are smooth and $u \in \mathbb{R}^m$ is the vector of control inputs.

The output of system (1) is restricted to satisfy

$$y = h(x), \quad \frac{\partial h(x)}{\partial x_v} = 0, \quad y \in \mathbb{R}^p, \quad (2)$$

where $h : \mathbb{R}^n \rightarrow \mathbb{R}^p$ is smooth.

The objective in path following is to design a feedback control law so that the output (2) of the closed-loop system follows a given, smooth parametrized curve $\sigma : \mathbb{R} \rightarrow \mathbb{R}^p$. In [13] geometric restrictions on the allowable class of curves are imposed.

Assumption 1. The path $\sigma(\cdot)$ is a 1-dimensional embedded submanifold of \mathbb{R}^p .

Assumption 2. There exists a smooth map $s : \mathbb{R}^p \rightarrow \mathbb{R}^{p-1}$ such that 0 is a regular value of s and $\sigma(\mathbb{D}) = s^{-1}(0)$. Let $\gamma := s^{-1}(0)$. Moreover, the lift of γ to \mathbb{R}^n

$$\Gamma := (s \circ h)^{-1}(0) = \{x \in \mathbb{R}^n : s(h(x)) = 0\} \quad (3)$$

is a submanifold of \mathbb{R}^n .

Assumption 1 ensures that the path has no self-intersections or corners. Assumption 2 requires that the entire path be represented as the zero level set of a smooth function $s : \mathbb{R}^p \rightarrow \mathbb{R}^{p-1}$ in the output space of the system (2), and that its Jacobian is full rank at each point of the curve γ . Locally this is always possible if Assumption 1 holds [13].

Making the state x of (1) approach the set (3) is equivalent to making the system output (2) approach its desired path $\gamma = s^{-1}(0)$. Unfortunately, the set Γ is generally not invariant under the dynamics of (1) and as a result cannot be stabilized. Therefore, rather than stabilizing Γ , we instead stabilize the largest controlled invariant set contained in Γ . This set, denoted Γ^* , is called the path following manifold of (1) with respect to the path γ . The path following manifold consists of all the motions of the system (1) for which the output (2) remains on the curve γ by suitable choice of control input u . Let the dimension of Γ^* be $n^* > 0$. Equivalently Γ^* can be thought of as the zero dynamics manifold [14] of (1) with output $\lambda(x) := s \circ h(x)$. The existence of Γ^* is assured as long as the path γ is feasible for the system (1) [13].

If the virtual output $\lambda(x)$ yields a well-defined relative degree at some point x^* on Γ^* then, locally, input-output feedback linearization for non-square systems can be performed at x^* . In particular, this guarantees the existence of

a coordinate transformation $T : U \rightarrow T(U)$, $x \mapsto (\eta, \xi)$ where U is a neighbourhood of x^* , and a regular feedback transformation¹ $u = \alpha(x) + \beta(x)v$ such that $T(\Gamma^* \cap U) = \{(\eta, \xi) : \xi = 0\}$. Defining $v := \text{col}(v^\dagger, v^\parallel)$, in the new coordinates, system (1) becomes

$$\begin{aligned} \dot{\eta} &= f^0(\eta, \xi) + g^\dagger(\eta, \xi)v^\dagger + g^\parallel(\eta, \xi)v^\parallel \\ \dot{\xi} &= A^\dagger \xi + B^\dagger v^\dagger \end{aligned} \quad (4)$$

where $(\eta, \xi) \in \mathbb{R}^{n^*} \times \mathbb{R}^{n-n^*}$, the pair (A^\dagger, B^\dagger) is controllable and the various functions are all smooth [13]. The ξ -subsystem describes the motion off the set Γ^* and for this reason is called the transversal dynamics. The outer transversal control input v^\dagger can be used to stabilize the origin of this subsystem using linear control techniques. This ensures attractiveness of Γ^* and invariance of the path.

The zero dynamics of the system, $\dot{\eta} = f^0(\eta, 0) + g^\parallel(\eta, 0)v^\parallel$, are called the tangential dynamics and govern the motion of the system restricted to the desired path. The outer tangential control input v^\parallel can, in principle, be designed to achieve the desired motion on the path following manifold. However, since the tangential dynamics are in general nonlinear, designing the control law v^\parallel can be difficult [15] or impossible.

B. Tangential Control

In [13], the structure of mechanical systems (1) is leveraged to refine the normal form (4), so that the η -subsystem is partially linear, time-invariant and controllable. By Assumption 1, there exists a map

$$\begin{aligned} \sigma : \mathbb{R} &\rightarrow \mathbb{R}^p \\ \theta &\mapsto \sigma(\theta) \end{aligned} \quad (5)$$

where, in the case of closed paths, there exists a number $L > 0$ such that for all $\theta \in \mathbb{R}$, $\sigma(\theta + L) = \sigma(\theta)$. If we assume, without loss of generality, that σ is regular, then the smallest such $L > 0$ is the arc-length of the closed curve.

A projection operator is introduced that maps a point y , sufficiently close to the desired path, to a unique $\theta \in \mathbb{R}$ ($\theta \in [0, L)$ for closed-paths) such that $\sigma(\theta)$ is closest to y . In particular,

$$\begin{aligned} \varpi : \gamma_\epsilon &\rightarrow \mathbb{R} \\ y &\mapsto \arg \inf_{\theta \in \mathbb{R}} \|y - \sigma(\theta)\| \end{aligned} \quad (6)$$

where γ_ϵ is a tubular neighbourhood of the path γ . Using the map (6) and $s \circ h(x)$ from Assumption 2, a ‘‘virtual’’ output is defined as:

$$\hat{y} := \begin{bmatrix} \pi(x) \\ \lambda(x) \end{bmatrix} := \begin{bmatrix} \varpi \circ h(x) \\ s \circ h(x) \end{bmatrix}. \quad (7)$$

In [13, Theorem 3.2], it is shown that if for some $x^* \in \mathbb{R}^n$, $h(x^*) \in \gamma$,

$$\text{Im} \left(\frac{\partial h}{\partial x} g_v(x_c) \Big|_{x=x^*} \right) \simeq \mathbb{R}^p \quad (8)$$

¹The feedback transformation $u = \alpha(x) + \beta(x)v$ is regular if for all $x \in U$, $\beta(x)$ has rank m .

holds for the system (1)-(2), and for a path γ satisfying Assumptions 1 and 2, then via input-output feedback linearization on the virtual output (7), the system can be transformed [16] to a refined version of the the normal form (4) given by

$$\begin{aligned}\dot{\eta}_1 &= f^0(\eta, \xi) + g^{\text{h}}(\eta, \xi)v^{\text{h}} + g_1^{\parallel}(\eta, \xi)v_1^{\parallel} + g_2^{\parallel}(\eta, \xi)v_2^{\parallel} \\ \dot{\eta}_2 &= A^{\parallel}\eta_2 + B^{\parallel}v_2^{\parallel} \\ \dot{\xi} &= A^{\text{h}}\xi + B^{\text{h}}v^{\text{h}}.\end{aligned}\quad (9)$$

Here $\dim(\eta_1) = n^* - 2$, $\dim(\eta_2) = 2$, the pairs $(A^{\parallel}, B^{\parallel})$, $(A^{\text{h}}, B^{\text{h}})$ are controllable, and $v = \text{col}(v^{\parallel}, v^{\text{h}}) = \text{col}(v_1^{\parallel}, v_2^{\parallel}, v^{\text{h}}) \in \mathbb{R}^{m-p} \times \mathbb{R} \times \mathbb{R}^{p-1}$. Notice that a basic feasibility requirement for condition (8) is $m \geq p$. If this is true, then for a standard robotic manipulator, (8) will hold on its entire state space except at its singular configurations.

As before, stabilizing the path following manifold is equivalent to stabilizing the origin of the ξ -subsystem. On the path following manifold, the dynamics reduce to

$$\begin{aligned}\dot{\eta}_1 &= f^0(\eta, 0) + g_1^{\parallel}(\eta, 0)v_1^{\parallel} + g_2^{\parallel}(\eta, 0)v_2^{\parallel} \\ \dot{\eta}_2 &= A^{\parallel}\eta_2 + B^{\parallel}v_2^{\parallel}.\end{aligned}\quad (10)$$

At any two points on the path following manifold $(\hat{\eta}_1, \hat{\eta}_2)$ and $(\tilde{\eta}_1, \tilde{\eta}_2)$, the output (2) of (1) lies on the desired path γ . However, if the output lies on different points on the desired path then, by the definition of the projection operator (6), $\tilde{\eta}_2 \neq \hat{\eta}_2$. Hence, the η_2 -subsystem determines the motion of the output along the desired path. Thus, v_2^{\parallel} can be used to control the motion along the path. The η_1 -subsystem represents the dynamics on the path following manifold that do not produce motion along the path in the output space. Note that the η_1 -subsystem does not exist if $m = p = N$, thus, this subsystem captures the redundancy of the system (1).

Although this is a model based technique, we will see in Section IV-B and V that a robust controller may be used as the control law for v to account for modelling uncertainty.

III. MODELLING

Modelling the mechanical dynamics of a fully actuated robot manipulator can be done via the Euler-Lagrange formulation

$$\begin{aligned}\frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{q}_i} - \frac{\partial \mathcal{L}}{\partial q_i} &= \tau_i, \quad i \in \{1, \dots, N\} \\ \mathcal{L} &= \mathcal{K}(q, \dot{q}) - \mathcal{P}(q)\end{aligned}\quad (11)$$

where $q, \dot{q} \in \mathbb{R}^N$ are the joint angles and velocities, respectively, $\tau \in \mathbb{R}^N$ are the joint torques, $\mathcal{K} : \mathbb{R}^N \times \mathbb{R}^N \rightarrow \mathbb{R}$ is the system's kinetic energy and $\mathcal{P} : \mathbb{R}^N \rightarrow \mathbb{R}$ is the system's potential energy. The functions \mathcal{K} and \mathcal{P} depend on the robot's mechanical parameters such as the link mass, center of mass, and inertia for each link. Equation (11) can be rewritten in the familiar vector form

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau \quad (12)$$

where $D(q) \in \mathbb{R}^{N \times N}$ is the positive-definite, symmetric inertia matrix, $C(q, \dot{q}) \in \mathbb{R}^{N \times N}$ represents the centripetal

and Coriolis terms, and $G(q) = \nabla \mathcal{P}(q) = (d\mathcal{P}_q)^{\top} \in \mathbb{R}^N$ represents the gravitation effects [17].

The test platform is the Clearpath Robotic Manipulator (CPM) (see Figure 1) which is a four degree of freedom, fully actuated system. The shoulder, elbow, and wrist links are actuated by DC linear actuators.

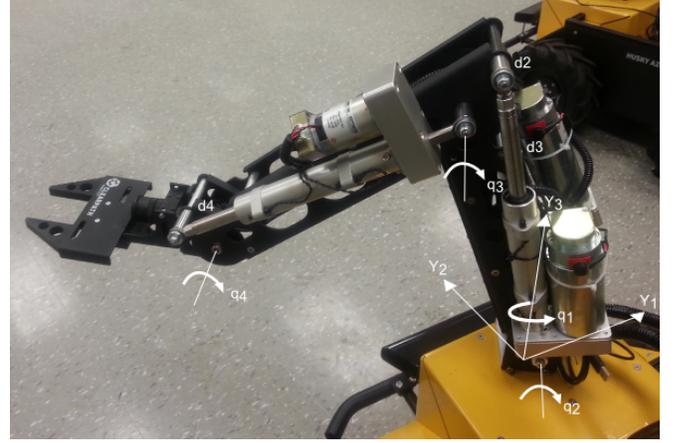


Fig. 1. The Clearpath Manipulator (CPM) with the joint coordinates and output space labelled.

The link center of mass and inertias are constant in the Denavit-Hartenberg (DH) coordinate frame attached to each link [17]. However, the linear actuator center of mass and inertias are not constant in the corresponding DH frame, as the linear actuator extension d_i and orientation vary with the joint angles. This varying center of mass and inertia of the linear actuators significantly complicates the dynamic model. To simplify the model an average is taken over the CPM workspace to yield constant parameters in each DH frame. In particular, let $d_i \in \mathbb{R}, i \in \{2, 3, 4\}$ be the linear actuator extension for the i th joint, and $I_{LA_i} : \mathbb{R} \rightarrow \mathbb{R}^{3 \times 3}, m_{cLA_i} : \mathbb{R} \rightarrow \mathbb{R}^3$ be the inertia tensor and center of mass location for the i th actuator, respectively, with respect to the DH frame of the link the actuator is attached to. Then the average is taken as

$$\begin{aligned}\bar{I}_{LA_i} &:= \frac{1}{d_i^{\text{max}}} \int_0^{d_i^{\text{max}}} I_{LA_i}(d_i) dd_i \\ \bar{m}_{cLA_i} &:= \frac{1}{d_i^{\text{max}}} \int_0^{d_i^{\text{max}}} m_{cLA_i}(d_i) dd_i\end{aligned}$$

where d_i^{max} is the maximum extension of linear actuator i . These values are used as constant approximations of the center of mass and inertia tensors of the actuators in their respective DH frame. All of the CPM mechanical parameters are expected to be inaccurate because they are generated by SolidWorks[©] and were not physically measured.

The waist actuator is a standard DC motor passing through a worm gearbox. Let the angle of this motor be q_{m1} . The linear actuators consists of a rotational brushed DC motor and a leadscrew. Based on a standard DC motor model, the

motor dynamics become

$$\begin{aligned} J_{m_1} \ddot{q}_{m_1} + B_1 \dot{q}_{m_1} &= K_1 u_1 - \frac{\tau_1}{r_1} \\ J_{m_i} \ddot{d}_i + B_i \dot{d}_i &= K_i u_i - \frac{F_i}{r_i \rho_i}, \quad i \in \{2, 3, 4\} \end{aligned} \quad (13)$$

where J_{m_i} , B_i , K_i are the motor parameters, r_i are the gearbox ratios, F_i are the linear actuator load forces, ρ_i comprises the lead screw parameters [18], and u_i are the motor voltages. The parameters were obtained from the actuator data sheets and were not physically measured. An invertible map $l_i : \mathbb{R} \rightarrow \mathbb{R}$, $i \in \{2, 3, 4\}$, is used to convert the joint angles q_i to the linear actuator extension d_i :

$$d_i = l_i(q_i) = \sqrt{a_i^2 + h_i^2 - 2a_i h_i \cos(q_i + \beta_i)} - L_i. \quad (14)$$

Here, for $i \in \{2, 3, 4\}$, a_i are the link lengths, h_i, β_i are parameters that depend on the motor mounting scheme, and L_i are the lengths of the linear actuator housing. The joint torques $\tau_i, i \in \{2, 3, 4\}$ in (12) are related to the linear actuator force F_i by a map $\zeta_i : \mathbb{R} \rightarrow \mathbb{R}$,

$$\zeta_i(q_i) := \frac{\sin(q_i + \beta_i)}{L_i + d_i} h_i,$$

such that $\tau_i = F_i \zeta_i(q_i)$, $i \in \{2, 3, 4\}$. Using these maps and the chain rule, (12) and (13) become

$$\begin{aligned} [D(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q)]_1 &= \tau_1 \\ r_1^2 J_{m_1} \ddot{q}_1 + r_1^2 B_1 \dot{q}_1 &= r_1 K_1 u_1 - \tau_1 \\ \left. \begin{aligned} [D(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q)]_i &= F_i \zeta_i(q_i) \\ r_i \rho_i J_{m_i} \frac{\partial^2 l_i}{\partial q_i^2}(q_i) \ddot{q}_i + r_i \rho_i \left(B_i \frac{\partial l_i}{\partial q_i}(q_i) + \right. \\ \left. J_{m_i} \frac{\partial^2 l_i}{\partial q_i^2}(q_i) \right) \dot{q}_i &= r_i \rho_i K_i u_i - F_i \end{aligned} \right\} i \in \{2, 3, 4\} \end{aligned}$$

which can be written compactly

$$M(q)\ddot{q} + \tilde{C}(q, \dot{q})\dot{q} + G(q) = A(q)u. \quad (15)$$

The dynamic model of the robot neglects friction and uses crude estimates of the parameters. This model uncertainty is addressed by the controller design in the subsequent section.

IV. CONTROL DESIGN

A. Transverse Feedback Linearization

In accordance with the discussion in Section II-A, let $x_c := q$, $x_v := \dot{q}$ where q and \dot{q} are the variables in the CPM model (15). The model falls into the class of systems (1)-(2) with $f_v(x) = -M^{-1}(x_c) \left(\tilde{C}(x)x_v + G(x_c) \right)$, $g_v(x_c) = M^{-1}(x_c)A(x_c)$, and

$$\begin{aligned} h(x_c) &:= [y_1 \quad y_2 \quad y_3 \quad y_4]^T \\ &= \begin{bmatrix} \cos(x_1) (a_2 \cos(x_2) + a_3 \cos(x_2 + x_3)) \\ \sin(x_1) (a_2 \cos(x_2) + a_3 \cos(x_2 + x_3)) \\ a_2 \sin(x_2) + a_3 \sin(x_2 + x_3) + d_1 \\ x_2 + x_3 + x_4 \end{bmatrix} \end{aligned}$$

where (y_1, y_2, y_3) denotes the position of the wrist with respect to the basis $\{Y_1, Y_2, Y_3\}$ (see Figure 1) and y_4 is

the angle the end-effector makes with the ground. Our main focus in this paper is to deal with robustness to modelling errors, thus we avoid dealing with the redundancy of the robot by augmenting the output y_4 . Thus $N = m = p$ and we expect no η_1 dynamics in (9).

The objective is to make the output y approach and traverse a path γ that satisfies Assumptions 1 and 2. One such path is a circle in a plane, which will be used for illustration, but more complicated paths such as splines can be used with the proposed approach. Without loss of generality, let the circle be centred at the coordinate $o_c \in \mathbb{R}^3$ with respect to the basis $\{Y_1, Y_2, Y_3\}$, have radius r , and be contained in the plane defined by the unit normal vector $Y_{c_3} \in \mathbb{R}^3$. Furthermore, let $Y_{c_3} = Y_{c_1} \times Y_{c_2}$, where $\|Y_{c_i}\| = 1$, $i \in \{1, 2\}$. The coordinate transformation $T_{Y_c}^Y : \mathbb{R}^3 \rightarrow \mathbb{R}^3$,

$$y = T_{Y_c}^Y(y_c) = [Y_{c_1} \quad Y_{c_2} \quad Y_{c_3}] y_c + o_c \quad (16)$$

transforms coordinates y_c in the frame $\{Y_{c_1}, Y_{c_2}, Y_{c_3}\}$ attached to the center of the circle to the output coordinate frame $\{Y_1, Y_2, Y_3\}$. Thus, for some desired end-effector angle $\psi \in [0, 2\pi)$ in the output space, the parametrized curve can be written as

$$\sigma(\theta) = \begin{bmatrix} T_{Y_c}^Y \begin{pmatrix} r \cos(\theta) \\ r \sin(\theta) \\ 0 \end{pmatrix} \\ \psi \end{bmatrix}. \quad (17)$$

The corresponding lift of the path (see Assumption 2) is given by (3), where

$$s \circ h(x) := \begin{bmatrix} \left(y_{c_1}^2 + y_{c_2}^2 - r^2 \right) \\ y_{c_3} \\ x_2 + x_3 + x_4 - \psi \end{bmatrix}_{y_c = (T_{Y_c}^Y)^{-1} \circ h(x)} \quad (18)$$

where the elements of s correspond to being on the desired cylinder, plane, and end-effector angle, respectively.

To get to the normal form (9), condition (8) needs to be satisfied. It is necessary that $m \geq p$ for this to hold, which is true for this example. Furthermore, the matrix $\frac{\partial h}{\partial x} g_v(x_c)$ is full rank everywhere except at the singularities of the manipulator (shoulder link lines up with the elbow link) and when $A(x_c)$ is singular, which by design of the CPM cannot occur. Therefore, the path can be positioned to avoid these singularities and thus will satisfy (8).

Next the projection operator (6) is required in the construction of the virtual output (7). In the case of a circle,

$$\varpi(y) = r \arg \left((y_{c_1} + iy_{c_2})_{y_c = (T_{Y_c}^Y)^{-1}(y)} \right) \quad (19)$$

is the arc length of the output on the circle, and thus will uniquely project points off the circle to $(-r\pi, r\pi] \subset \mathbb{R}$ provided that $(y_{c_1}, y_{c_2}) \neq (0, 0)$. The functions $\lambda(x) = s \circ h(x)$, $\pi(x) = \varpi \circ h(x)$ are obtained from (18) and (19), and the virtual output (7) has a well defined relative degree at each point on the circle. Thus the desired normal form

can be obtained by applying the coordinate transformation,

$$T : U \subseteq \mathbb{R}^8 \rightarrow T(U) \subseteq (-r\pi, r\pi) \times \mathbb{R}^7$$

$$x \mapsto (\eta, \xi) = \begin{bmatrix} \eta_2^1 \\ \eta_2^2 \\ \xi_i^1 \\ \xi_i^2 \end{bmatrix} = \begin{bmatrix} \pi(x) \\ L_f \pi(x) \\ \lambda_i(x) \\ L_f \lambda_i(x) \end{bmatrix}, \quad (20)$$

for $i \in \{1, 2, 3\}$. The states $\xi_1^1, \xi_2^1, \xi_3^1$ correspond to the error to the cylinder, plane, and end-effector angle, respectively. Although the above coordinate transformation is only guaranteed to be well-defined in a neighbourhood of a point x^* on the set Γ^* , it can be shown that in the case of the circular paths, U contains the entire set Γ^* . The dynamics in (η, ξ) -coordinates become

$$\begin{cases} \dot{\eta}_2^1 = \eta_2^2 \\ \dot{\eta}_2^2 = L_f^2 \pi(x)|_{x=T^{-1}(\eta, \xi)} + L_g L_f \pi(x)|_{x=T^{-1}(\eta, \xi)} u \\ \dot{\xi}_i^1 = \xi_i^2 \\ \dot{\xi}_i^2 = L_f^2 \lambda_i(x)|_{x=T^{-1}(\eta, \xi)} + L_g L_f \lambda_i(x)|_{x=T^{-1}(\eta, \xi)} u \end{cases} i \in \{1, 2, 3\}. \quad (21)$$

The feedback transformation, which is well defined in U ,

$$u = \begin{bmatrix} L_g L_f \pi(x) \\ L_g L_f \lambda(x) \end{bmatrix}_{x=T^{-1}(\eta, \xi)}^{-1} \left(- \begin{bmatrix} L_f^2 \pi(x) \\ L_f^2 \lambda(x) \end{bmatrix}_{x=T^{-1}(\eta, \xi)} + v \right) \quad (22)$$

$$=: \beta^{-1}(\eta, \xi)(-\alpha(\eta, \xi) + v)$$

yields the feedback linearized system

$$\begin{cases} \dot{\eta}_2^1 = \eta_2^2 \\ \dot{\eta}_2^2 = v_2^\parallel \\ \dot{\xi}_i^1 = \xi_i^2 \\ \dot{\xi}_i^2 = v_i^\flat \end{cases} i \in \{1, 2, 3\} \quad (23)$$

where $v = \text{col}(v_2^\parallel, v^\flat) \in \mathbb{R} \times \mathbb{R}^3$. This is precisely the desired normal form (9) with $A^\parallel = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$, $B^\parallel = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, $A^\flat = \text{diag}(A_1, A_2, A_3)$, $A_i = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$, $B^\flat = \text{diag}(B_1, B_2, B_3)$, $B_i = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$. The outer control law for v can be designed by linear control techniques to stabilize the origin of the ξ -subsystem in order to stabilize the manifold Γ^* , and to track desired motions along the path governed by the η_2 -subsystem. Note that the decoupling matrix $\beta(\cdot) \in \mathbb{R}^{4 \times 4}$ in (22) will be full rank since (8) holds and the path satisfies Assumptions 1 and 2, except at the center of the circle and at the singular configurations of the manipulator. For general paths, this result may only be true in a neighbourhood of a point on the path γ .

B. Robust Controller

To track a desired constant velocity $\eta_2^{2\text{ref}}$ along the path, the following outer control law may be used

$$v_2^\parallel(\eta, t) = K_{p_1} (\eta_2^{2\text{ref}} - \eta_2^2(t)) + K_{I_1} \int_0^t (\eta_2^{2\text{ref}} - \eta_2^2(\tau)) d\tau. \quad (24)$$

The above PI control law works well in practice for a majority of systems in the presence of disturbance [19]. It turns out that this controller works well for our tracking purposes for the tangential speed along the path. The ξ -subsystem in (21) with (22) can be rewritten as

$$\dot{\xi} = A^\flat \xi + B^\flat [\alpha(\eta, \xi) + \beta(\eta, \xi)u]_{2:4} \quad (25)$$

Indeed the feedback transformation (22) results in perfect cancellation of the nonlinear terms $\alpha(\cdot), \beta(\cdot)$ resulting in the linearized system $\dot{\xi} = A^\flat \xi + B^\flat v^\flat$. However, with the unmodelled dynamics and parameter uncertainties described in Section III, the resulting system takes the form

$$\begin{aligned} \dot{\xi} &= A^\flat \xi + B^\flat [\alpha(\eta, \xi) + \Delta_\alpha(\eta, \xi) + (\beta(\eta, \xi) + \Delta_\beta(\eta, \xi))u]_{2:4} \\ &=: A^\flat \xi + B^\flat [\alpha(\eta, \xi) + \beta(\eta, \xi)u]_{2:4} + \delta(\eta, \xi, u) \end{aligned}$$

where $\Delta_\alpha(\cdot) \in \mathbb{R}^4$, $\Delta_\beta(\cdot) \in \mathbb{R}^{4 \times 4}$ represent the modelling uncertainties. Thus we do not expect perfect cancellation of the nonlinear terms using the control input (22). The outer control input v^\flat can be designed to overcome the perturbation term $\delta(\eta, \xi, u)$. It is shown in [20] that if the uncertainty satisfies

$$\begin{aligned} \|(\Delta_\beta(\eta, \xi) + \beta(\eta, \xi))^{-1} \beta(\eta, \xi) - I_{4 \times 4}\| &\leq k_1 < 1 \\ \|(\Delta_\beta(\eta, \xi) + \beta(\eta, \xi))^{-1} \Delta_\alpha(\eta, \xi)\| &\leq k_2 \|\text{col}(\eta, \xi)\| \end{aligned} \quad (26)$$

then the inner control loop (22) and the outer controller of the form

$$v^\flat = (K + K_0) \xi + \begin{cases} K_1 \xi / \|\xi\| & : \|\xi\| \geq \mu > 0 \\ K_2 \|\xi\| \xi & : \|\xi\| < \mu. \end{cases} \quad (27)$$

where $(A^\flat + B^\flat K)$ is Hurwitz, $K_0, K_1, K_2, \mu \in \mathbb{R}$ are constants that depend on k_1, k_2 , then the origin of ξ will be asymptotically stable. In practice, finding k_1, k_2 is difficult, which implies difficulty in selecting μ and the gains K_i . However we treat μ and K_i as tuning parameters in the following section.

V. EXPERIMENTAL RESULTS

We implement the path following controller on the CPM via Labview Real-Time Module[©] to control the motor PWM amplifiers and to read the optical encoders. The encoders read the d_i of equation (13) (the linear actuator extension). Hence the inverse mapping of (14), L_i^{-1} , is used to retrieve the states x_c of the system. The derivative of x_c is numerically computed to retrieve the x_v states. The block diagram of the control system can be seen in Figure 2.

To check the validity of the robust controller (27), we examine how uncertain the nominal model is. Figure 3 compares two transversal controllers, one with simple state feedback $v^\flat = K\xi$ and the other with the robust controller (27). The feedback gain K is the same for both controllers, and thus, only the robustifying terms were tuned in the latter controller. Clearly the simple state feedback on ξ was not able to stabilize the transversal subsystem (ξ -subsystem) and hence the wrist position is far from the desired path,

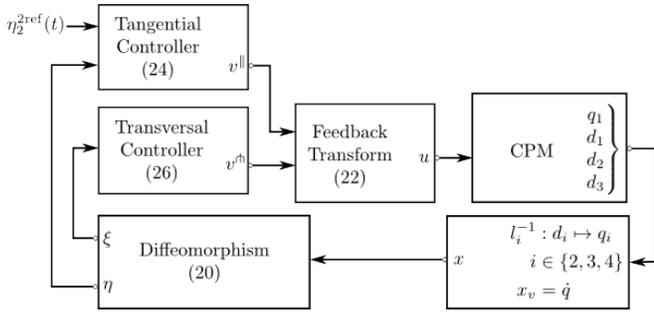


Fig. 2. Block diagram of the final control system.

showing the need and usefulness of the robust controller to overcome model uncertainty. There is still slight error due to the bounds of the perturbation terms $\Delta_\alpha, \Delta_\beta$ not satisfying (26). Nevertheless the error is significantly reduced.

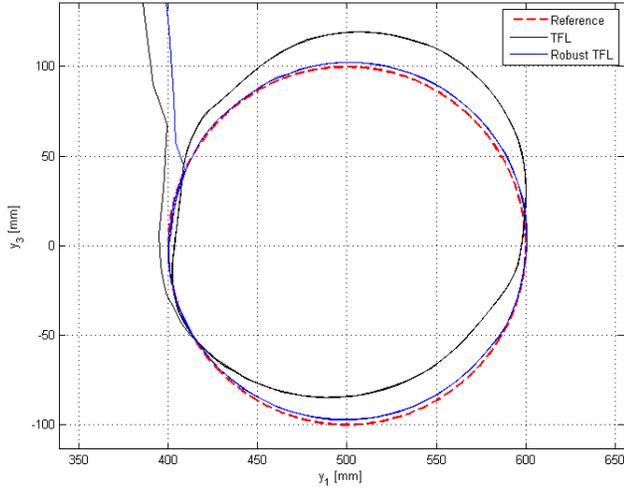


Fig. 3. Comparison of our path following controller with and without the robust component (27) on the transversal states. The path parameters in (16) and (17) are $Y_{c1} = [1 \ 0 \ 0]^T, Y_{c2} = [0 \ 0 \ 1]^T, Y_{c3} = [0 \ 1 \ 0]^T, \psi = 0, r = 100 \text{ mm}$, and $o_c = [500 \ 0 \ 0]^T$.

The robust control law is also compared with the simple trajectory tracking controller of independent-joint PID [17]:

$$u_i = K_{P_i} (x_{c_i}^{\text{ref}}(t) - x_{c_i}(t)) + K_{D_i} (\dot{x}_{c_i}^{\text{ref}}(t) - \dot{x}_{c_i}(t)) + K_{I_i} \int_0^t (x_{c_i}^{\text{ref}}(\tau) - x_{c_i}(\tau)) d\tau, \quad i \in \{1, 2, 3, 4\},$$

where the reference signal x_c^{ref} is generated by solving the inverse kinematics for the desired path (17) with the desired speed parametrization. Namely, the path parameter $\theta(t)$ is such that $\dot{\theta}(t) = \frac{\eta_2^{2\text{ref}}}{r}$. The joint velocities x_v^{ref} are generated by $x_v^{\text{ref}} = J^{-1}(x_c^{\text{ref}})\sigma'(t)$, where $J = \frac{\partial h(x_c)}{\partial x_c}$ is the manipulator Jacobian [17].

We test the path of a circle in the horizontal plane shown in Figure 4 with a constant reference velocity of $\eta_2^{2\text{ref}} = 100 \frac{\text{mm}}{\text{s}}$ tangent to the circle. The transformed state positions are shown in Figure 5. From (18) and (20), the states $\xi_1^1, \xi_2^1, \xi_3^1$ correspond to the error to the cylinder, plane, and end-effector angle, respectively.

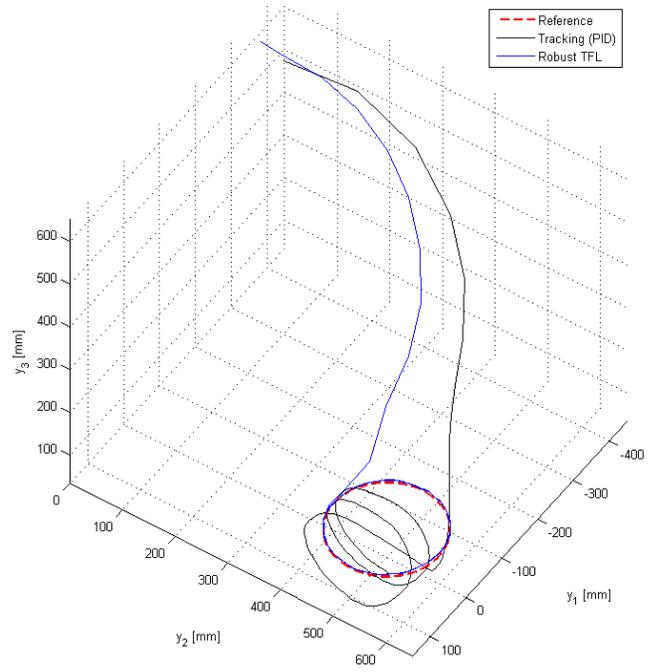


Fig. 4. Comparison of the robust path following controller with a trajectory tracking controller in the output space. The path parameters in (16) and (17) are $Y_{c1} = [1 \ 0 \ 0]^T, Y_{c2} = [0 \ 1 \ 0]^T, Y_{c3} = [0 \ 0 \ 1]^T, \psi = 0, r = 100 \text{ mm}$, and $o_c = [0 \ 500 \ 100]^T$.

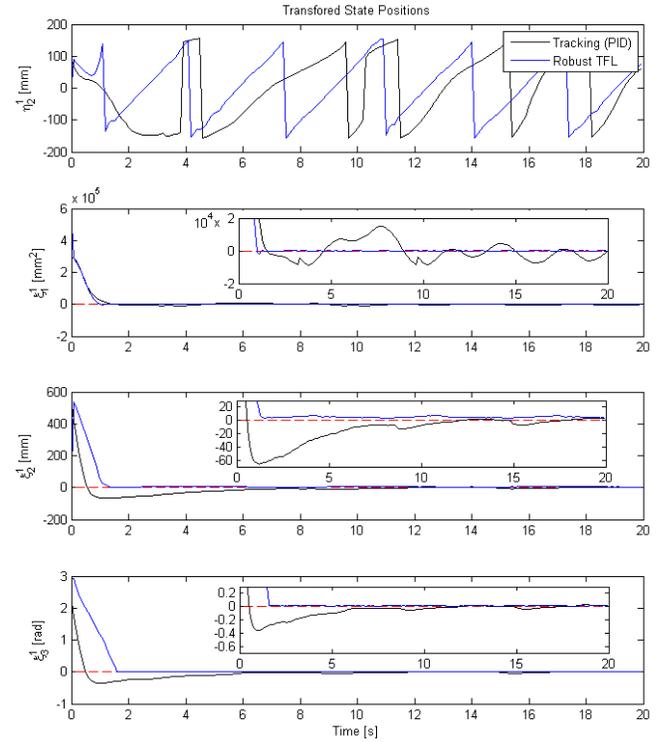


Fig. 5. Transformed state positions with the transformation T applied from (20).

These results demonstrate one of the main disadvantages of classical trajectory tracking control. The large initial error causes a poor transient response, and since the controller is

trying to catch up to the trajectory, forces the manipulator to actually cut across the cylinder. On the other hand, the path following controller naturally converges to the closest point on the path due to the explicit stabilization of the path following manifold Γ^* . The dynamics are being accounted for in a robust manner, thus gain-tuning for the outer loop controller on v becomes easier as the dynamics appear linear in this space.

The inputs are shown in Figure V. The two controllers are operating in a similar range of control effort.

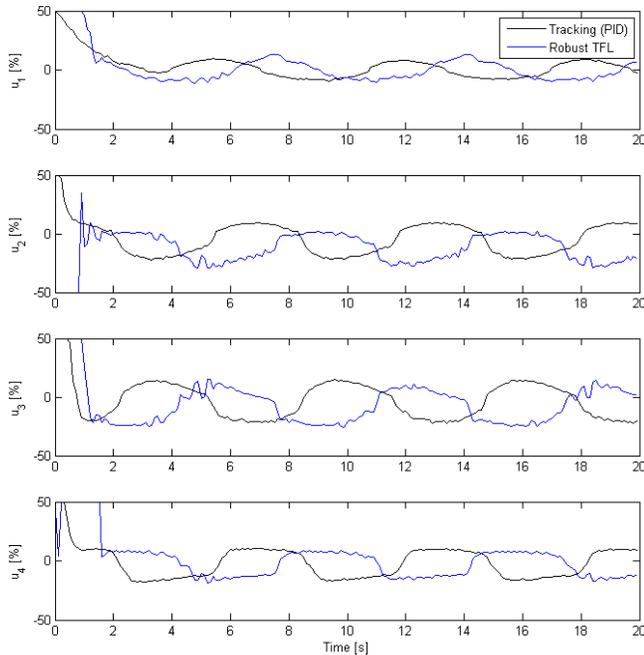


Fig. 6. Comparison of the control signal for the path following controller and the trajectory tracking controller.

VI. CONCLUSION AND FUTURE WORK

This paper proposes a robust path following controller design approach for articulated manipulators. The path following controller uses transverse feedback linearization to stabilize the path following manifold. Then, by Lyapunov redesign, a robust controller is used to enforce asymptotic stability of the transverse dynamics in the presence of modelling uncertainty. In addition, the controller uses a refinement of the normal form to facilitate desired motion on the path via a linear controller. The approach was validated on a 4-DOF robot manipulator using circles as the paths, and performed well despite the modelling inaccuracies.

We also fully linearized the system by defining a four dimensional output space. Had we removed the constraint ψ on the end-effector angle, the resulting system would have been redundant. Dealing with redundancy in the context of path following is an area for future work.

REFERENCES

[1] J. Hauser and R. Hindman, "Aggressive flight maneuvers," in *Decision and Control, 1997. IEEE, 1997*, pp. 4186–4191.

[2] L. Lapierre and D. Soetanto, "Nonlinear path-following control of an auv," *Ocean engineering*, vol. 34, no. 11, pp. 1734–1744, 2007.

[3] A. P. Aguiar, J. P. Hespanha, and P. V. Kokotovic, "Path-following for nonminimum phase systems removes performance limitations," *Automatic Control, IEEE Transactions on*, vol. 50, no. 2, pp. 234–239, 2005.

[4] J. Hauser and R. Hindman, "Maneuver regulation from trajectory tracking: Feedback linearizable systems," in *Proc. IFAC Symp. Non-linear Control Systems Design, 1995*, pp. 595–600.

[5] R. Skjetne, T. I. Fossen, and P. V. Kokotović, "Robust output maneuvering for a class of nonlinear systems," *Automatica*, vol. 40, no. 3, pp. 373–383, 2004.

[6] K. Kant and S. W. Zucker, "Toward efficient trajectory planning: The path-velocity decomposition," *The International Journal of Robotics Research*, vol. 5, no. 3, pp. 72–89, 1986.

[7] U. Mettin, P. X. La Hera, D. O. Morales, A. S. Shiriaev, L. B. Freidovich, and S. Westerberg, "Path-constrained trajectory planning and time-independent motion control: Application to a forestry crane," in *14th International Conference on Advanced Robotics (ICAR), Proceedings, 2009*.

[8] A. S. Shiriaev, L. B. Freidovich, A. Robertsson, R. Johansson, and A. Sandberg, "Virtual-holonomic-constraints-based design of stable oscillations of furuta pendulum: Theory and experiments," *Robotics, IEEE Transactions on*, vol. 23, no. 4, pp. 827–832, 2007.

[9] S. Westerberg, U. Mettin, A. S. Shiriaev, L. B. Freidovich, and Y. Orlov, "Motion planning and control of a simplified helicopter model based on virtual holonomic constraints," in *Advanced Robotics, International Conference on. IEEE, 2009*, pp. 1–6.

[10] C. Chevallereau, E. Westervelt, and J. Grizzle, "Asymptotic stabilization of a five-link, four-actuator, planar bipedal runner," in *Decision and Control*, vol. 1. IEEE, 2004, pp. 303–310.

[11] A. Banaszuk and J. Hauser, "Feedback linearization of transverse dynamics for periodic orbits," *Systems & control letters*, vol. 26, no. 2, pp. 95–105, 1995.

[12] C. Nielsen and M. Maggiore, "Output stabilization and maneuver regulation: A geometric approach," *Systems & control letters*, vol. 55, no. 5, pp. 418–427, 2006.

[13] A. Hladio, C. Nielsen, and D. Wang, "Path following for a class of mechanical systems," *Control Systems Technology, IEEE Transactions on*, Accepted November 2012. DOI: 10.1109/TCST.2012.2223470.

[14] A. Isidori, *Nonlinear control systems II*. Springer Verlag, 1999, vol. 2.

[15] L. Consolini, M. Maggiore, C. Nielsen, and M. Tosques, "Path following for the pvtol aircraft," *Automatica*, vol. 46, no. 8, pp. 1284–1296, 2010.

[16] C. Nielsen, C. Fulford, and M. Maggiore, "Path following using transverse feedback linearization: Application to a maglev positioning system," *Automatica*, vol. 46, no. 3, pp. 585–590, 2010.

[17] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot modeling and control*. John Wiley & Sons, Hoboken, NJ, 2006.

[18] R. L. Norton, *Machine design: an integrated approach*. Prentice-Hall New York, 1996.

[19] B. C. Kuo and M. F. Golnaraghi, *Automatic control systems*. John Wiley & Sons, 2003, vol. 4.

[20] F. Esfandiari and H. Khalil, "Output feedback stabilization of fully linearizable systems," *International Journal of Control*, vol. 56, no. 5, pp. 1007–1037, 1992.