Path Following for Quadrotors

Sant Kumar and Rajan Gill

***Abstract*— This paper presents a path following controller for quadrotors to follow splines in the output space. The control pipeline includes utilization of a cascaded control architecture, generation of paths using quintic spline interpolation, transformation of coordinates using Frenet-Serret (FS) frames, and decoupling controllers for the transversal and tangential sub-systems. The result is a time-invariant controller that renders the desired path invariant and attractive from arbitrary initial conditions, even when thrown into the air (shown in experiments), while maintaining a desired tangential velocity.**

I. INTRODUCTION

This paper builds on the body of work pertaining to path following of mechanical systems [1] [2] [3] [4] and references therein. Path following is a broad term, and in our context, we define it as the ability to drive a system's output to a desired path in space and traverse it at a required tangential velocity. Such an objective could be achieved by standard time varying trajectory tracking controllers (for which a plethora of studies have already been done for the quadrotor, see next paragraph), however properties such as attractiveness of the path from arbitrary initial conditions or invariance of the path (i.e. once on the path with velocity tangential to it, the system will remain on the path for all future time) can no longer be guaranteed [1] [2]. This is guaranteed, however, by a set stabilization approach [3] which, in this paper, yields a time-invariant controller. In our path following context, the set that is to be stabilized are the collection of all state trajectories whose associated outputs lie on the desired path; this set is defined as the path following manifold in [3]. Our approach towards stabilizing this path following manifold is based on a well studied theory of transverse feedback linearization [3] which, apart from guaranteeing path invariance, also divides the control design process into two separate, decoupled branches: make the output approach the path and stay on it (transversal) and another to achieve the desired motion on the path (tangential) [2]. This paper presents the experimental validation of employing this control approach on quadrotors.

Within the realm of quadrotors, there have been a number studies on trajectory tracking controllers since after mid 2000s. In [5][6], a nonlinear H∞ & backstepping controller was employed for tracking a helical trajectory. In [7], a feedback linearization law, linearizing the rotational and translational dynamics, was used for trajectory (and point) tracking. Additionally, other forms of trajectory tracking controllers based on adaptive control, sliding mode control, predictive and nonlinear robust control have been studied ([7] [8] and references therein). However, trajectory tracking involves a reference trajectory being parametrized in space as well as time which may not be suitable when spatial errors are more critical than temporal errors [1]. On the contrary, path following involves a time-free parametrization of the reference path. In [9], path following was achieved by allowing the quadrotor to pursue, without intercepting, a virtual target moving on the path. In this technique, the convergence towards the path was found to be highly sensitive to initial conditions and vehicle velocity. In fact, this approach, as well as other contour error mitigation strategies (typically used in the machining community), does not guarantee that the desired path will be invariant [4]. The approach in this paper is insensitive to initial conditions, and is shown in experiments by throwing the quadrotor in the air.

In [8] and [10], a path following controller for quadrotors based on transverse feedback linearization is presented. They utilize input dynamic extension, augmenting two additional states, in order for the system output to achieve a well defined vector relative degree, and was shown in simulation. Their approach requires the inversion of a $4x4$ matrix and is limited to a relatively simple class of paths such as circles. In this paper, this requirement is reduced to inverting a $3x3$ matrix (a marginal improvement in computation) owing to a cascaded control architecture [11], explained briefly in Section II-B, which avoids the need of employing input dynamic extension. This gives the added advantage of being able to integrate the proposed path following controller with standard flight controllers that have on-board attitude and acceleration control[1][2]. The path following routine presented here draws motivation from [4], where a transformation of the state coordinates is done using Frenet-Serret (FS) frames [12] to achieve a time-invariant, path-following controller (PFC) for a large class of paths: spline interpolation of way-points. In this paper, the PFC produces, as its output, the required body acceleration and body-rate feed-forward (parametrized by state, not in time) for the quadrotor in order to follow a reference path with a user defined velocity.

The remainder of this paper is organized as follows: Section II briefly explains the system model and the cascaded control architecture. Section III describes the path following problem and Section IV describes the PFC design. Section V show experimental results from various initial conditions.

This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Swiss National Science Foundation (SNSF).

The authors are with the Institute of Dynamic Systems and Control (IDSC) in the Mechanical and Process Engineering Deparatment, ETH Zurich, 8037 Zurich, Switzerland {santsa, rgill}@ethz.ch

[1] PX4

[2] Ardupilot

Finally, some conclusions and ideas for future works have been proposed in Section VI.

Notation

Let $\langle p,\ q\rangle$ denote the standard inner product of vectors p and q in $\mathbb{R}^n$. The Euclidean norm of a vector p is denoted by $\|p\|$. The derivative of p with respect to time is represented by $\dot{p}$ and the derivative with respect to path parameter is represented by p'.

II. System Model

A. Quadrotor Dynamics

The quadrotor is modelled as a rigid body with six degrees of freedom. The position of the center of mass of the quadrotor $x_p = (x_{p1}, x_{p2}, x_{p3})$ in 3D cartesian coordinates and the euler angles $q = (\phi, \theta, \psi) := (roll, pitch, yaw)$ represent the translational and rotational degrees of freedom [13]. The translational acceleration in the inertial frame is given by [14]:

$$a = \begin{bmatrix} \ddot{x}_{p1} \\ \ddot{x}_{p2} \\ \ddot{x}_{p3} \end{bmatrix} = R(q) \begin{bmatrix} 0 \\ 0 \\ T \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{1}$$

where $R(q)$ is the rotation matrix which rotates a vector in the body frame to inertial frame, $g = 9.81 m/s^2$ is the acceleration due to gravity and T is the mass normalized collective thrust of the four propellers. As for the body rate $\omega = (\omega_x, \omega_y, \omega_z)$, the commanded body rates ω_{cmd} are tracked by the onboard controller which feedback linearizes the nonlinear rotational dynamics [14]. The relation between angular angle rate $\dot{q}$ and body rate ω is given by:

$$J_r \dot{q} = \omega \tag{2}$$

$$\mathbf{J}_r = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{bmatrix} \tag{3}$$

B. Control Architecture and System Dynamics

A cascaded control architecture as shown in Fig. 1 was used for quadrotor control. The cascaded control structure is a commonly used concept and has already been studied for quadrotor control [11] [15] [16]. In this paper, the design of the PF controller is discussed (Section IV). It is assumed that dynamics of the inner control loops comprised of attitude controller [17], onboard controller and quadrotor [11] are fast enough such that the combined system plant behaves as a simple double integrator with acceleration a_{des} as direct input, along with body rate feed-forward input ω_{ff} and desired yaw ψ_{des}. The following state and output equations are considered for this system.

$$\text{State } x = [x_p \quad x_v]^T \in \mathbb{R}^6$$
$$\text{where } x_v := \dot{x}_p$$

With acceleration $a \in \mathbb{R}^3$ as control input, this results in the following linear dynamics:

$$\dot{x} = Ax + Ba \tag{4}$$

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The output of system (4) is defined as:

$$y = h(x) = x_p \in \mathbb{R}^3 \tag{5}$$

III. Problem Formulation

A. Reference Path

For the reference path, we consider a parametrized curve:

$$\sigma(\cdot) : [0, \lambda_{max}] \to \mathbb{R}^3,$$

Given a $\lambda_{max} \in \mathbb{R}$, this reference path can be represented by the set embedded in the output space:

$$\mathscr{P} := \sigma([0, \lambda_{max}]) \subset \mathbb{R}^3 \tag{6}$$

Assumption (framed curves) **:** For all $\lambda \in [0, \lambda_{max}]$, $\sigma'(\lambda)$, $\sigma''(\lambda)$, $\sigma'''(\lambda)$ are linearly independent (all derivatives are with respect to the parameter λ).
This assumption allows for the Frenet-Serret Frames of the path, discussed in Section IV-A, to be well defined [4] in the output space $\mathbb{R}^3$.
Remark: In practice, the path is generated by quintic spline interpolation (see Section V). The *framed curves* assumption is a relatively weak assumption since when quintic spline interpolating, the probability of the first three path derivatives being linearly dependent in a non-trivial sub-interval of $[0, \lambda_{max}]$ is zero (loosely speaking) [4].

B. Problem Statement

The problem investigated in this paper is to find a continuous time-invariant feedback control law for a_{des}, ω_{ff} and ψ_{des} for the quadrotor system (Fig. 1) resulting in the system output (5) exponentially approaching the given path $\mathscr{P}$ and traversing along the path with a user-defined constant reference tangential velocity η_2^{ref}. This control law also has to ensure path invariance (at any time instant $t = 0$ if output $y(0) \in \mathscr{P}$, then $y(t) \in \mathscr{P}\ \forall t > 0$) and path attractiveness ($y(t) \to \mathscr{P}$, $\forall y(0) \notin \mathscr{P}$). This will be done via set stabilization to yield the required PF controller block in Fig. 1.

IV. Path Following Controller Design

The proposed PF controller requires the path parameter λ^* which correspond to a point on the path closest to the current system output $y = h(x)$. This λ^* can be defined as:

$$\lambda^* = s(x) := \arg \min_{\lambda \in [0, \lambda_{max}]} \|h(x) - \sigma(\lambda)\| \tag{7}$$

At $t = 0$, this parameter is computed by quantizing the path and searching for the closest point in a brute-force fashion. For $t > 0$, the adaptive gradient descent algorithm in [4] is used. The algorithm implementation in [4] also works on self-intersecting curves.

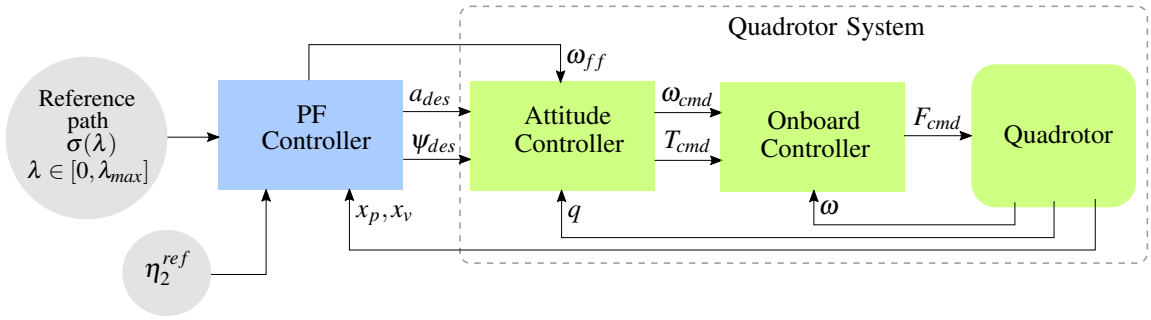

Fig. 1. Cascaded quadrotor control architecture

Frenet-Serret Frame

The Frenet-Serret Frame (FSF) at a point on the path is an orthonormal basis formed from tangent, normal and binormal unit vectors at that point of the path. Projections on these unit vectors defined on the reference path, as shown in Fig. 2, are used to represent transformed states discussed in Section IV-A. Interested readers are referred to [12] and [18] for further technical details about FSF. In this paper, the following expressions are used for these vectors. Here, the unit-tangent, unit-normal and unit-binormal FSF vectors are represented by t, p and b respectively [18]:

$$
\begin{aligned}
t(\lambda) &= \frac{\sigma'(\lambda)}{\|\sigma'(\lambda)\|} \\
p(\lambda) &= \frac{t'(\lambda)}{\|t'(\lambda)\|} \\
b(\lambda) &= t(\lambda) \times p(\lambda) \\
\frac{d}{d\lambda}\begin{bmatrix} t \\ p \\ b \end{bmatrix} &= \|\sigma'(\lambda)\| \begin{bmatrix} 0 & \kappa & 0 \\ -\kappa & 0 & \tau \\ 0 & -\tau & 0 \end{bmatrix} \begin{bmatrix} t \\ p \\ b \end{bmatrix}, \ \lambda \in [0,1] \\
\text{where, } \kappa &= \frac{\|\sigma'(\lambda) \times \sigma''(\lambda)\|}{\|\sigma'(\lambda)\|^3}, \ \tau = \frac{[\sigma'(\lambda), \sigma''(\lambda), \sigma'''(\lambda)]}{\|\sigma'(\lambda) \times \sigma''(\lambda)\|^2}
\end{aligned}
\tag{8}
$$

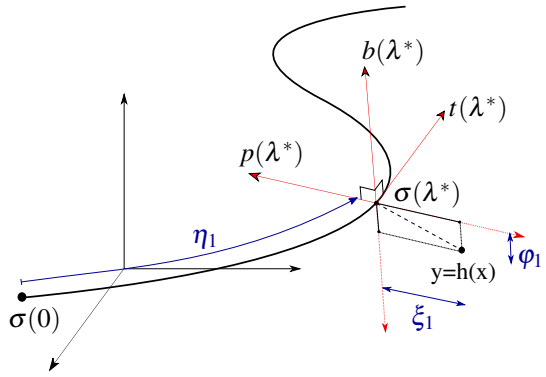

Fig. 2. Frenet-Serret Frames, tangential position η_1 and transversal positions $\{\xi_1, \varphi_1\}$

A. Coordinate Transformation

In the first step towards PF controller design, a coordinate transformation T is constructed that maps the state x of system (4) to a new set of well-separated coordinates representing the position and velocity of the quadrotor along the reference path (tangential) and towards the reference path (transversal).

1) Tangential States: The first tangential state η_1 (tangential position) is the arclength traversed along the path, i.e. path arclength between $\sigma(0)$ and $\sigma(\lambda^*)$ defined as:

$$\eta_1(x) := \int_0^{\lambda^*} \left\| \frac{d\sigma(r)}{dr} \right\| dr \tag{9}$$

The second tangential state η_2 (tangential velocity) is computed by taking time derivative of η_1. It turns out η_2 is the projection of the quadrotor velocity x_v onto unit-tangent FS vector eq(8) [4].

$$\eta_2(x) := \dot{\eta}_1 = \langle t(\lambda^*), x_v \rangle \tag{10}$$

2) Transversal States: As shown in Fig. 2, the transversal positions (ξ_1 and φ_1) are the projection of transversal distance from the reference path, that is $h(x) - \sigma(\lambda^*)$, on the unit-normal and unit-binormal FS vectors (8). Taking their time derivatives gives the other two transversal states (transversal velocities, ξ_2 and φ_2) [4]:

$$\xi_1(x) := \langle p(\lambda^*), h(x) - \sigma(\lambda^*) \rangle \tag{11}$$

$$\xi_2(x) := \dot{\xi}_1 = \frac{\eta_2}{\|\sigma'(\lambda^*)\|} \langle p'(\lambda^*), h(x) - \sigma(\lambda^*) \rangle + \langle p(\lambda^*), x_v \rangle \tag{12}$$

$$\varphi_1(x) := \langle b(\lambda^*), h(x) - \sigma(\lambda^*) \rangle \tag{13}$$

$$\varphi_2(x) := \dot{\varphi}_1 = \frac{\eta_2}{\|\sigma'(\lambda^*)\|} \langle b'(\lambda^*), h(x) - \sigma(\lambda^*) \rangle + \langle b(\lambda^*), x_v \rangle \tag{14}$$

B. Dynamics in the Transformed Coordinates

The dynamics in the transformed coordinates can be written as:

$$
\begin{aligned}
\dot{\eta}_1 &= \eta_2 \\
\dot{\eta}_2 &= \alpha_1(x) + \beta_1(x) a \\
\dot{\xi}_1 &= \xi_2 \\
\dot{\xi}_2 &= \alpha_2(x) + \beta_2(x) a \\
\dot{\varphi}_1 &= \varphi_2 \\
\dot{\varphi}_2 &= \alpha_3(x) + \beta_3(x) a
\end{aligned}
\tag{15}
$$

where

$$\alpha_1(x) = \frac{\eta_2}{\|\sigma'(\lambda^*)\|}\langle t'(\lambda^*), x_v\rangle$$
$$\beta_1(x) = t(\lambda^*)^\top$$
$$\alpha_2(x) = \frac{\eta_2}{\|\sigma'(\lambda^*)\|}\langle p'(\lambda^*), 2x_v - \eta_2(x)t(\lambda^*)\rangle + \left\langle h(x)-\sigma(\lambda^*), p''(\lambda^*)\left(\frac{\eta_2}{\|\sigma'(\lambda^*)\|}\right)^2 + p'(\lambda^*)\left(\frac{\alpha_1(x)}{\|\sigma'(\lambda^*)\|} - \eta_2^2(x)\frac{\langle\sigma'(\lambda^*),\sigma''(\lambda^*)\rangle}{\|\sigma'(\lambda^*)\|^4}\right)\right\rangle$$
$$\beta_2(x) = p(\lambda^*)^\top + (h(x)-\sigma(\lambda^*))^\top p'(\lambda^*)\frac{\beta_1(x)}{\|\sigma'(\lambda^*)\|}$$
$$\alpha_3(x) = \frac{\eta_2}{\|\sigma'(\lambda^*)\|}\langle b'(\lambda^*), 2x_v - \eta_2(x)t(\lambda^*)\rangle + \left\langle h(x)-\sigma(\lambda^*), b''(\lambda^*)\left(\frac{\eta_2}{\|\sigma'(\lambda^*)\|}\right)^2 + b'(\lambda^*)\left(\frac{\alpha_1(x)}{\|\sigma'(\lambda^*)\|} - \eta_2^2(x)\frac{\langle\sigma'(\lambda^*),\sigma''(\lambda^*)\rangle}{\|\sigma'(\lambda^*)\|^4}\right)\right\rangle$$
$$\beta_3(x) = b(\lambda^*)^\top + (h(x)-\sigma(\lambda^*))^\top b'(\lambda^*)\frac{\beta_1(x)}{\|\sigma'(\lambda^*)\|} \tag{16}$$

where $\lambda^* = s(x)$ from equation (7) and FS vectors derivatives are given in equation (8). The reader can refer to [4] for the derivation of above expressions[3].

C. Transverse Feedback Linearization (TFL)

Directly following from the definition of ξ_1 and φ_1, lift of the path σ to the state space $\mathbb{R}^6$ can be defined as:

$$\Gamma := \{x \in \mathbb{R}^6 : \xi_1(x) = 0, \varphi_1(x) = 0\} \tag{17}$$

The set Γ is the set of all $x \in \mathbb{R}^6$ which correspond to the system output y lying in the set $\mathscr{P}$ (6). Under the system dynamics (4) Γ is not invariant and hence cannot be stabilized. Thus we instead stabilize the largest controlled invariant set Γ^* contained in Γ. This set is called the path following manifold of (4) with respect to $\mathscr{P}$. This set consists of all the motions of system (4) for which the system output y remains in $\mathscr{P}$ by suitable choice of control input a. Equivalently Γ^* can be thought of as zero dynamics manifold [19] of (4) with output $(\xi_1(x), \varphi_1(x))$.

$$\begin{aligned}\Gamma^* &:= \{x \in \mathbb{R}^6 : \xi_1(x) = 0, \dot{\xi}_1(x) = 0, \varphi_1(x) = 0, \dot{\varphi}_1(x) = 0\} \\ &= \{x \in \mathbb{R}^6 : \xi_1(x) = 0, \xi_2(x) = 0, \varphi_1(x) = 0, \varphi_2(x) = 0\}\end{aligned} \tag{18}$$

Therefore, the following virtual output is considered:

$$\hat{y} := \begin{bmatrix}\eta_1(x)\\ \xi_1(x)\\ \varphi_1(x)\end{bmatrix} \tag{19}$$

[3]In [4], there is an error on page 1383: the term $\frac{L_g L_f \eta_1(x)}{\|\sigma'(\lambda^*)\|}$ in $L_f^2\xi_1^{j-1}(x)$ should be $\frac{L_f^2\eta_1(x)}{\|\sigma'(\lambda^*)\|}$.

It can be observed in (15) and (16) that this virtual output has a vector relative degree of $\{2,2,2\}$ for all $x \in \mathbb{R}^6$ [4]. Thus an input-output feedback linearization can be performed by defining the map T as:

$$\begin{aligned}T : U \subseteq \mathbb{R}^6 &\to T(U) \subseteq \mathbb{R}^6 \\ x &\mapsto (\eta_1(x), \eta_2(x), \xi_1(x), \xi_2(x), \varphi_1(x), \varphi_2(x))\end{aligned} \tag{20}$$

The dynamics in these new coordinates have been defined in Section IV-B. Observing the structure of the dynamics (15), an auxiliary input $u = \begin{bmatrix}u_\eta & u_\xi & u_\varphi\end{bmatrix}^\top \in \mathbb{R}^3$ is defined such that

$$u = \beta(x)a + \alpha(x) \tag{21}$$

where

$$\beta(x) = \begin{bmatrix}\beta_1(x)\\ \beta_2(x)\\ \beta_3(x)\end{bmatrix},\ \alpha(x) = \begin{bmatrix}\alpha_1(x)\\ \alpha_2(x)\\ \alpha_3(x)\end{bmatrix} \tag{22}$$

The new dynamics with auxiliary input are thus simplified into the following linearized system:

$$\left.\begin{aligned}\dot{\eta}_1 &= \eta_2\\ \dot{\eta}_2 &= u_\eta\end{aligned}\right\}\text{Tangential subsystem}$$

$$\left.\begin{aligned}\dot{\xi}_1 &= \xi_2\\ \dot{\xi}_2 &= u_\xi\\ \dot{\varphi}_1 &= \varphi_2\\ \dot{\varphi}_2 &= u_\varphi\end{aligned}\right\}\text{Transversal subsystem}$$

The ξ and φ subsystems describe the motion off the set Γ^* and thus represent transversal dynamics. Stabilizing this transversal subsystem is equivalent to enforcing Γ^* attractive and controlled invariant, thereby rendering the same for the path $\mathscr{P}$ in the output space. A simple linear controller for transversal control input (u_ξ, u_φ) can be used to stabilize the origin of this subsystem. Similarly a separate linear controller can be used for tangential control input u_η for the tangential subsystem to track a desired tangential velocity η_2^{ref}.

The well-defined relative degree of the chosen virtual output $\hat{y}$ (19) at all $x \in \mathbb{R}^6$ also guarantees the feedback transformation (21) to be regular [3], that is, $\beta(x)$ being non-singular for all $x \in \mathbb{R}^6$. The desired acceleration input a_{des} for the quadrotor system (Fig. 1) is then calculated:

$$a_{des} = \beta(x)^{-1}(u - \alpha(x)) \tag{23}$$

D. ω_{ff} and ψ_{des} Computation

In this subsection body rate feed-forward input ω_{ff} and desired yaw input ψ_{des} are computed to be sent to the inner control loops (Fig. 1). From equation (1), the total mass-normalized forces required by the quadrotor to follow the reference path is given by:

$$f := a + \begin{bmatrix}0\\0\\g\end{bmatrix} = R(q)\begin{bmatrix}0\\0\\T\end{bmatrix} \tag{24}$$

From [20], the body rates ω_x and ω_y are expressed as:

$$\begin{bmatrix}\omega_y\\ -\omega_x\\ 0\end{bmatrix} = R(q)^T\left(\frac{\dot{a}}{\|f\|} - \frac{ff^T\dot{a}}{\|f\|^3}\right) \tag{25}$$

In principle, when the system output is tracking the reference path at the desired rate η_2^{ref} while remaining on the path, then $x \in \Gamma^*$, $\eta_2 = \eta_2^{ref}$ and $u = 0$. In this case, taking the time derivative of equation (21) yields:

$$\begin{aligned} 0 &= \beta(x)\dot{a} + \frac{d\beta(x)}{dt}a + \frac{d\alpha(x)}{dt} \\ \Rightarrow \dot{a} &= -\beta(x)^{-1}\left(\frac{d\beta(x)}{dt}a + \frac{d\alpha(x)}{dt}\right) \end{aligned} \tag{26}$$

where $a = -\beta(x)^{-1}\alpha(x)$ from equation (23). Additionally, in this case, $\frac{d\beta(x)}{dt}$ and $\frac{d\alpha(x)}{dt}$ are found by differentiating (16) with respect to time:

$$\begin{aligned} \left.\frac{d\beta_1(x)}{dt}\right|_{x\in\Gamma^*} &= t'(\lambda^*)\frac{d\lambda^*}{dt} \\ \left.\frac{d\beta_2(x)}{dt}\right|_{x\in\Gamma^*} &= p'(\lambda^*)\frac{d\lambda^*}{dt} \\ \left.\frac{d\beta_3(x)}{dt}\right|_{x\in\Gamma^*} &= b'(\lambda^*)\frac{d\lambda^*}{dt} \end{aligned}$$

$$\begin{aligned} \left.\frac{d\alpha_1(x)}{dt}\right|_{x\in\Gamma^*} &= \eta_2\left(\frac{d\|\sigma'(\lambda^*)\|^{-1}}{d\lambda^*}\frac{d\lambda^*}{dt}\right)t'(\lambda^*)^T x_v \\ &\quad + \frac{\eta_2}{\|\sigma'(\lambda^*)\|}\left[\left(t''(\lambda^*)\frac{d\lambda^*}{dt}\right)^T x_v + t'(\lambda^*)^T a\right] \\ \left.\frac{d\alpha_2(x)}{dt}\right|_{x\in\Gamma^*} &= \eta_2\left(\frac{d\|\sigma'(\lambda^*)\|^{-1}}{d\lambda^*}\frac{d\lambda^*}{dt}\right)p'(\lambda^*)^T(2x_v \\ &\quad - \eta_2 t(\lambda^*)) + \frac{\eta_2}{\|\sigma'(\lambda^*)\|}\left[\left(p''(\lambda^*)\frac{d\lambda^*}{dt}\right)^T(2x_v\right. \\ &\quad \left. - \eta_2 t(\lambda^*)) + p'(\lambda^*)^T\left(2a - \eta_2 t'(\lambda^*)\frac{d\lambda^*}{dt}\right)\right] \\ \left.\frac{d\alpha_3(x)}{dt}\right|_{x\in\Gamma^*} &= \eta_2\left(\frac{d\|\sigma'(\lambda^*)\|^{-1}}{d\lambda^*}\frac{d\lambda^*}{dt}\right)b'(\lambda^*)^T(2x_v \\ &\quad - \eta_2 t(\lambda^*)) + \frac{\eta_2}{\|\sigma'(\lambda^*)\|}\left[\left(b''(\lambda^*)\frac{d\lambda^*}{dt}\right)^T(2x_v\right. \\ &\quad \left. - \eta_2 t(\lambda^*)) + b'(\lambda^*)^T\left(2a - \eta_2 t'(\lambda^*)\frac{d\lambda^*}{dt}\right)\right] \end{aligned} \tag{27}$$

where $\lambda^* = s(x)$ (7), $\frac{d\lambda^*}{dt} = \frac{\eta_2}{\|\sigma'(\lambda^*)\|}$ [4], and $\frac{d\|\sigma'(\lambda^*)\|^{-1}}{d\lambda^*} = -\frac{\langle\sigma'(\lambda^*),\ \sigma''(\lambda^*)\rangle}{\|\sigma'(\lambda^*)\|^3}$. Thus, by substituting (24), (26) and (27) in (25), ω_x and ω_y are computed considering that the system output is on the reference path ($x \in \Gamma^*$) while tracking the desired tangential velocity ($\eta_2 = \eta_2^{ref}$)[4]. Furthermore, we consider a user defined yaw specification which is parametrized by displacement on the path, i.e. η_1 [8]:

$$\begin{aligned} \psi_{des} &= \psi(\eta_1) \\ \Rightarrow \dot{\psi}_{des} &= \frac{\partial\psi(\eta_1)}{\partial\eta_1}\frac{d\eta_1}{dt} = \frac{\partial\psi(\eta_1)}{\partial\eta_1}\eta_2 \end{aligned}$$

[4]This procedure can be extended to any $x \in \mathbb{R}^6$, however the expressions for $\frac{d\beta(x)}{dt}$ and $\frac{d\alpha(x)}{dt}$ will become excessively complicated.

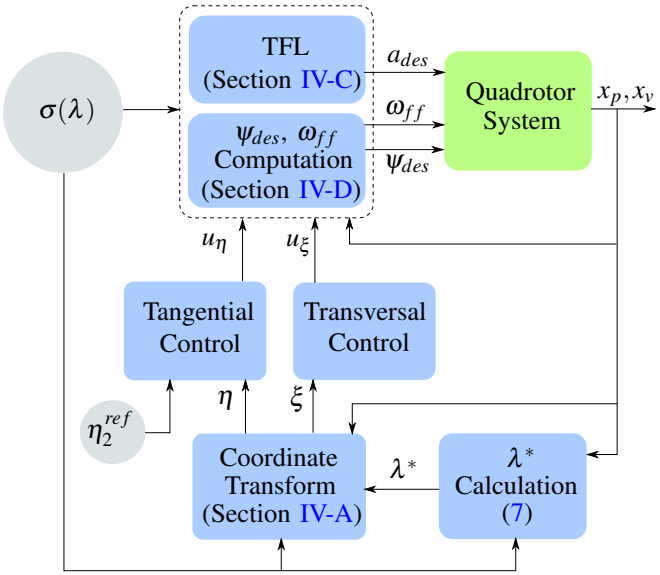

Fig. 3. Quadrotor path following controller block diagram

The euler angles (ϕ, θ and ψ) can be computed from (1), and ω_x, ω_y and $\dot{\psi}$ are computed as above. Thus, from (2) we get 3 linear equations and 3 unknown variables $\dot{\phi}$, $\dot{\theta}$ and ω_z $\Rightarrow$ ω_z is computed by solving this system of linear equations. Hence, the required body rate feed-forward input is obtained as a function of the state x:

$$\omega_{ff} = \begin{bmatrix}\omega_x \\ \omega_y \\ \omega_z\end{bmatrix} = \omega_{ff}(x), \quad \text{for } x \in \Gamma^*,\ \eta_2 = \eta_2^{ref} \tag{28}$$

A summary of the proposed path following control design is in Fig. 3.

V. EXPERIMENTAL RESULTS

In this section the experimental results obtained by implementing the proposed path following controller on a quadrotor are discussed. The experiments were performed in the Flying Machine Arena at ETH Zurich. A closed curve made from spline-interpolating 7 way-points is used as the reference path. The path also has a self intersection point as seen in Fig. 4. A constant yaw angle ($\psi_{des} = 0$) is considered. Using loop shaping techniques the band-width and damping ratio for the transversal controller are set to 6 Hz and 0.7, and that for the tangential controller are set to 4 Hz and 0.8 respectively.

A. Varying initial position (initial velocity = 0)

In Fig. 4, system output evolution with 5 different initial positions, all starting from rest (initial velocity = 0) are shown. This demonstrates the path attractiveness and path invariance nature of the proposed path following controller. Fig. 5 shows the evolution of tangential states η_1 and η_2 when the quadrotor moves along the closed reference path (PF1 in Fig. 4), starting from initial position $(0, -1.5, 2)$ at rest. It can be observed that the quadrotor tracks the desired tangential velocity of $-1.5m/s$ while following the

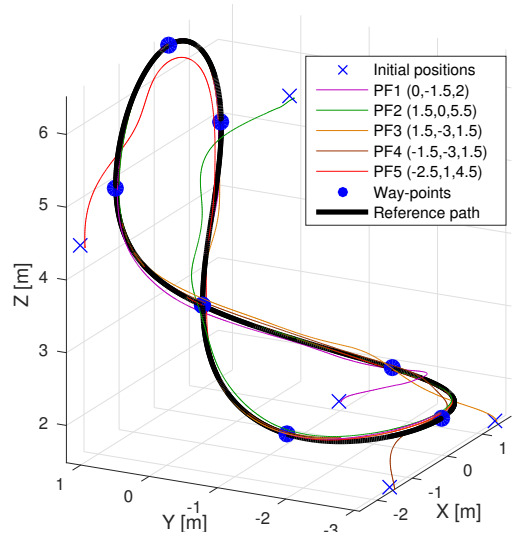

Fig. 4. Path following with different initial positions with zero initial velocity (coordinates shown in the legend are initial positions for the experiment), $\eta_2^{ref} = -1.5m/s^2$

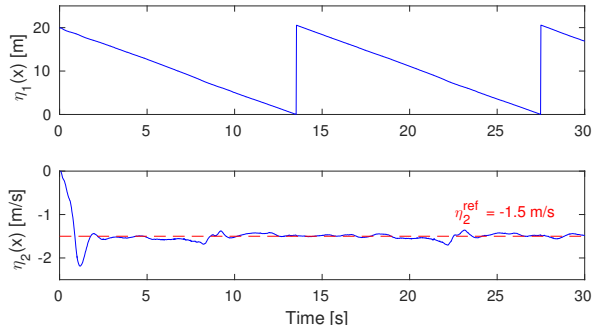

Fig. 5. Tangential states evolution for PF1 (Initial position = (0,-1.5,2), Initial velocity = 0)

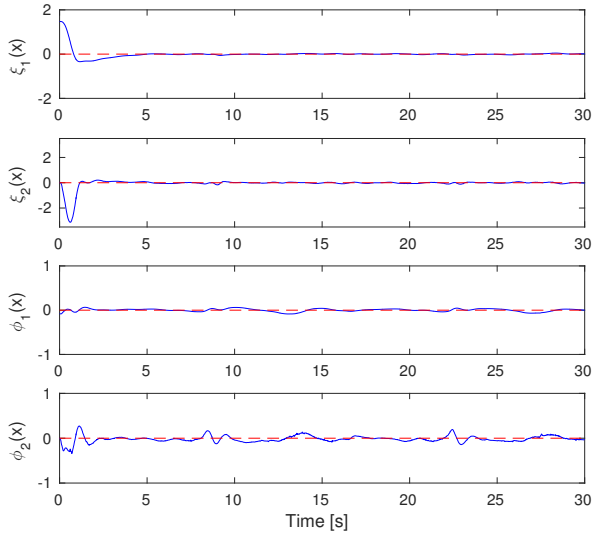

Fig. 6. Transversal states evolution for PF1 (Initial position = (0,-1.5,2), Initial velocity = 0)

reference path. In η_2 plot, small ripples are observed between $8-10s$ and $22-24s$ time spans. A potential cause for this is the unmodelled vortex ring state that occurs when the quadrotor moves downwards, leading to an inaccurate thrust model assumed by the inner control loop. Furthermore, the evolution of transversal states ξ and φ during the experiment can be seen in Fig. 6 in the same PF1 experiment initial conditions as mentioned above. As observed in this figure, the convergence of transversal states to 0 reaffirms the path attractiveness nature of this path following controller.

B. Varying initial position and velocity (thrown experiments)

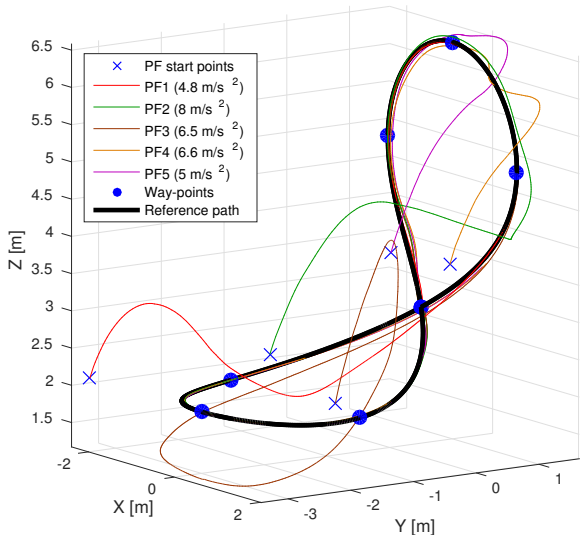

Fig. 7. Thrown experiments : Path following with different initial positions and different initial velocities (speed values shown in the legend are the magnitude of velocities when path following controller gets activated, i.e. quadrotor reaches the activation height of $2m$ for PF1, PF2 and PF3; $4m$ for PF4 and PF5), $\eta_2^{ref} = 1.5m/s^2$

In these experiments the quadrotor is randomly thrown in the air and the implemented path following controller is activated only when it reaches a user-defined height. Fig. 7 shows 5 such thrown experiments. Again, this demonstrates the path attractiveness and path invariance nature of the controller even with radically different initial conditions (initial position as well as initial velocity). Fig. 8 and

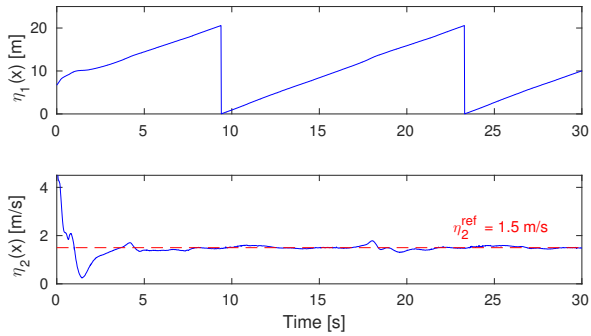

Fig. 8. Tangential states evolution for PF5 (thrown experiment)

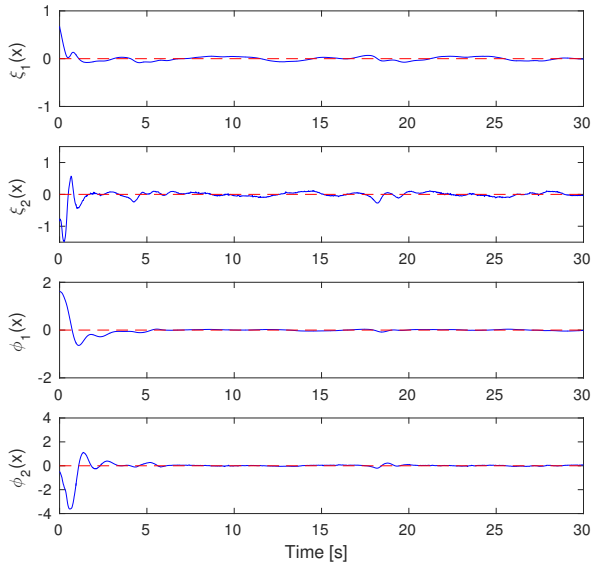

Fig. 9. Transversal states evolution for PF5(thrown experiment)

Fig. 9 show the evolution of tangential and transversal states respectively during PF5 thrown experiment (in Fig. 7). It can be observed from Fig. 8 that the quadrotor started with a non-zero tangential velocity ($\approx 3.5m/s$) but finally tracked the desired $\eta_2^{ref} = 1.5m/s$. Similarly in Fig. 9, it is observed that the transversal states are stabilized as expected. Again in this case some ripples can be seen in the η_2 plot in Fig. 8 possibly due to the similar reason as mentioned in the previous case.

VI. CONCLUSION

A path following controller is presented for quadrotors using a cascaded control architecture that stabilizes the set associated with the lift of the curve to the state space. The approach is shown for general paths that can be generated via spline interpolation. Associated body rate commands are also generated for when the vehicle is on the path and tracking the desired tangential reference speed. The controller is also time-invariant, as the commanded body rate feed forward signals and acceleration signals are parameterized by state and not in time.

The applications for such work can be in situations where spatial errors are more critical than temporal errors for an aerial vehicle. This could consist of surveillance and inspection applications.

In the future, one could extent the body rate signal generation to be valid for any state of the system, not just when path tracking is achieved. Furthermore, iterative learning can be done to improve the performance of the system. It would be interesting to generate a learned signal that is also parameterized by state and not in time, which can be cast into the path following framework presented in this work.

ACKNOWLEDGEMENT

This work is supported by and builds upon prior contributions by numerous collaborators in the Flying Machine Arena project. A list of past and present participants of the project is available at http://flyingmachinearena.org/people/.

REFERENCES

[1] J. Hauser and R. Hindman, "Maneuver regulation from trajectory tracking: Feedback linearizable systems," *3rd IFAC Symposium on Nonlinear Control Systems Design*, pp. 595–600, 1995.
[2] A. Hladio, C. Nielsen, and D. Wang, "Path following controller design for a class of mechanical systems," *18th IFAC World Congress*, vol. 44, no. 1, pp. 10 331–10 336, 2011.
[3] 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.
[4] R. Gill, D. Kulic, and C. Nielsen, "Spline path following for redundant mechanical systems," *IEEE Transaction on Robotics*, vol. 31, no. 6, pp. 1378–1392, 2015.
[5] G. V. Raffo, M. G. Ortega, and F. R. Rubio, "Backstepping/nonlinear h control for path tracking of a quadrotor unmanned aerial vehicle," *2008 American Control Conference*, pp. 3356–3361, 2008.
[6] S. Bouabdallah, P. Murrieri, and R. Siegwart, "Design and control of an indoor micro quadrotor," *2004 IEEE Conference on Robotics and Automation*, pp. 4393–4398, 2004.
[7] R. Bonna and J. F. Camino, "Trajectory tracking control of a quadrotor using feedback linearization," *International Symposium on Dynamic Problems of Mechanics*, 2015.
[8] A. Roza and M. Maggiore, "Path following controller for a quadrotor helicopter," *2012 American Control Conference (ACC)*, pp. 4655–4660, 2012.
[9] A. Manjunath, "Path following by a quadrotor using virtual target pursuit guidance," *All Graduate Theses and Dissertations, Utah State University*, no. Paper 4990, 2016.
[10] A. Akhtar, S. L. Waslander, and C. Nielsen, "Path following for a quadrotor using dynamic extension and transverse feedback linearization," *51st IEEE Conference on Decision and Control*, pp. 3551–3556, 2012.
[11] S. Lupashin, M. Hehn, M. W. Mueller, A. P. Schoellig, M. Sherback, and R. D'Andrea, "A platform for aerial robotics research and demonstration: The flying machine arena," *Mechatronics*, 2014.
[12] E. Cartan, *Elie Cartan (1869-1951) M.A. Akivis, B.A. Rosenfeld (transl. by V.V. Goldberg)*, 1993.
[13] J. Diebel, "Representing attitude: Euler angles, unit quaternions, and rotation vectors," *Matrix*, vol. 58, no. 1-35, 2006.
[14] M. Hehn and R. D'Andrea, "A flying inverted pendulum," in *2011 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2011, pp. 763–770.
[15] S. Bouabdallah and R. Siegwart, "Full control of a quadrotor," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2007, pp. 153–158.
[16] S. Lange, N. Sunderhauf, and P. Protzel, "A vision based onboard approach for landing and position control of an autonomous multirotor uav in gps-denied environments," in *International Conference on Advanced Robotics (ICAR)*. IEEE, 2009.
[17] D. Brescianini, M. Hehn, and R. D'Andrea, "Nonlinear quadrocopter attitude control," *Semester Thesis Technical Report, D-MAVT, ETH Zurich*, 2013.
[18] E. Kreyszig, *Differential Geometry*. Dover, 1991.
[19] I. A, *Nonlinear control systems (3rd edition)*. Springer.
[20] M. Hehn and R. D'Andrea, "Quadrocopter Trajectory Generation and Control," in *Proceedings of the 18th IFAC World Congress, 2011*, ser. IFAC proceedings series, vol. 18. Oxford, UK: International Federation of Automatic Control (IFAC), 2011.