2022 International Conference on Unmanned Aircraft Systems (ICUAS)
Dubrovnik, Croatia. June 21-24, 2022

Dynamic Feedback Linearization of a UAV Suspended Load System

Arash Mohammadhasani, Mohamed Al Lawati, Zifei Jiang, Alan F. Lynch, *Member, IEEE*

2022 International Conference on Unmanned Aircraft Systems (ICUAS) | 978-1-6654-0593-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICUAS54217.2022.9836223

***Abstract*— We present the dynamic state feedback linearization of an unmanned aerial vehicle (UAV) slung load system (SLS) that solves an output tracking problem for payload position and UAV yaw. The suspended payload is modelled as a single pendulum attached to the center of mass of the UAV with a spherical joint. We apply the Dynamic Extension Algorithm (DEA) to design an extended dynamics with full relative degree. A static state feedback linearization of the augmented and extended system provides LTI tracking error dynamics in the linearizing state coordinates. This dynamics is exponentially stable on a well-defined and practical region of state space. The DEA is implemented using symbolic computation software which can be applied to general nonlinear control-affine systems. Closed-loop performance is validated in simulation and compared with a state-of-the-art method from the literature.**

I. INTRODUCTION

There has been recent interest on using Unmanned Aerial Vehicles (UAVs) for load transport [1]. One approach uses gripper mechanisms that rigidly connect the UAV to the payload holding it close to vehicle [2], [3], [4]. Such an approach can make loading and unloading more challenging, especially in cluttered environments. Another approach uses a *slung load system* (SLS) where the load is suspended using one or more cables attached to the bottom of the UAV. SLSs separate the UAV from the payload and this improves safety and loading capability. As well, SLSs can be generalized to multiple cooperating UAVs for increased lifting capacity and full pose control of the load [5]. However, even for a single UAV SLS, motion control is a challenging problem given the coupling between the payload and vehicle which can create dangerous oscillations as described in [6] for full-sized helicopters. Designs must be robust, provide accurate load positioning, and be easy to tune. These characteristics are provided by the design proposed in this paper.

SLS motion control is based on a variety of models. For example, cable or rope flexibility is modelled with a PDE [7] or a chain of serially-connected links with spherical joints [8], [9], [10]. The model adopted in this paper is commonly used in the literature and consists of a single pendulum attached to the center of mass (CoM) of the UAV with a spherical joint. The payload is taken as a point mass. This model is accurate when tension in the cable is high and flexibility is negligible. A number of researchers have applied nonlinear control to this model and work can be divided based on the control output chosen: UAV position [11], [12], [13], [13] or payload position [8], [14]. The proposed method controls payload position and UAV yaw. Directly controlling payload position improves accuracy of load trajectory tracking. Control methods can also be categorized on control structure. A *full* design is based on the entire SLS dynamics consisting of the coupled UAV and pendulum subsystems [12]. An *inner-outer loop* design is performed on individual subsystems [5], [15], [13], [11]. The proposed method is a full design and benefits from a complete closed-loop stability analysis. The analysis of inner-outer loop designs often ignore coupling between the loops. In the following we describe existing approaches for nonlinear SLS control and compare them with the proposed method.

A design which considers UAV position as the controlled output is in [12] which initially considers the full SLS model. As the control problem is regulation of UAV position, load position is indirectly controlled by minimizing load swing. The design method uses an Interconnection and Damping Assignment-Passivity Based Control (IDA-PBC). This method solves a PDE for a desired closed-loop potential energy. To manage the complexity of solving this equation, designs are performed using dynamics restricted to longitudinal and lateral planes. Only stability (as opposed to asymptotic stability) of tracking error is proven. In [11] UAV position is the controlled output and an inner-outer loop design is proposed. The outer loop consists of translational UAV and pendulum angle dynamics. Two outer loop designs are proven to be asymptotically stable. The effect of loop coupling is ignored and, as in [12], the outer loop is split into lateral and longitudinal subsystems which are assumed decoupled. Work in [13] applies backstepping to obtain locally exponentially stable (ES) UAV position tracking error dynamics. As pendulum position is not directly tracked, an open-loop trajectory is planned to minimize pendulum oscillations. An inner-outer loop structure is adopted without analyzing the effect of loop coupling on stability.

Taking load position as the controlled output is another approach taken in the literature. For example, [14] proposes a 3-loop control structure where the inner loop tracks UAV attitude. The middle loop controls the load's attitude and UAV yaw. Finally, the outermost loop tracks load position. The entire tracking error dynamics is proven to be almost globally exponentially attractive. The approach is geometric as rotation matrices are used for UAV attitude and a 3D direction vector describes pendulum attitude. Exponential stability (ES) is on a local region which is difficult to

Arash Mohammadhasani, Mohamed Al Lawati, Zifei Jiang and Alan Lynch are with the Applied Nonlinear Control Lab (ANCL), Dept. Electrical & Computer Engineering, University of Alberta, AB, T6G 1H9, CA. Email: {arash.mhasani, maallawa, zifei.jiang, alan.lynch}@ualberta.ca

Al Lawati is also with Dept. Mechanical & Industrial Engineering, Sultan Qaboos University, Muscat, Oman.

determine and depends on controller gains. We remark that [14] proves differential flatness of the SLS model. Hence, the SLS is necessarily dynamic state feedback linearizable. However, this feedback control is not considered in [14]. The proposed method is the only known work which derives this linearizing control.

The contribution of this paper is to derive the dynamic state feedback linearization of the SLS model and to use it for output tracking for load position and UAV yaw. The *full* nonlinear system dynamics are considered in the design. Unlike most inner-outer loop methods where coupling between loops is neglected e.g., [11], [13], the motivation behind the proposed method is to compensate the entire dynamics. Since the extended dynamics is static state linearizable, the tracking error dynamics is exponentially stable on a well defined region of state space which is practical for most load transport scenarios (e.g., the magnitude of roll and pitch should be bounded by 90° which is expected for safe flight when carrying a load). Linearity of the error dynamics in the design coordinates simplifies gain tuning to meet transient specifications. The benefits of the proposed design are illustrated by comparing it with the approach in [14] in simulation. The Dynamic Extension Algorithm (DEA) used to compute the dynamic feedback linearization is implemented in Maple [16]. This code can be applied to general nonlinear control-affine systems.

This paper is organized as follows. Section II presents the SLS model. Section III presents the DEA and applies it to the SLS model. Section V provides simulation results for the proposed design. Section VI concludes the paper.

II. SLUNG LOAD SYSTEM (SLS) MODELLING

In this section, we provide a model for the UAV with suspended load in control-affine form

$$\begin{aligned} \dot{x} &= f(x) + \sum_{i=1}^{m} g_i(x) u_i \\ y_i &= h_i(x), \quad 1 \le i \le m \end{aligned} \tag{1}$$

with vector fields $f, g_i : \mathscr{M} \to \mathbb{R}^n$ and output functions $h_i : \mathscr{M} \to \mathbb{R}$ defined on an open subset $\mathscr{M} \subseteq \mathbb{R}^n$. In (1), $x \in \mathscr{M}$ is the state, and $u = [u_1, \ldots, u_m]^\top \in \mathscr{U} \subset \mathbb{R}^m$ is the control input. We choose a commonly used nonlinear rigid body model for the UAV and follow the presentation in [17]. We make the common assumption that the pendulum pivots about the UAV CoM (centre of mass) and therefore does not exert torque on the UAV. This implies the rotational dynamics of the UAV are decoupled from the pendulum dynamics. Two references frames (see Fig. 1) are used: a navigation frame $\mathscr{N}$ and body frame $\mathscr{B}$. Frame $\mathscr{N}$ is assumed inertial and has orthonormal basis vectors n_1, n_2, n_3 oriented north, east, and down, respectively. The origin of frame $\mathscr{B}$ is the UAV's CoM and its basis vectors b_1, b_2, b_3 oriented (relative to the vehicle) forward, right, and down, respectively. We denote the rotation matrix $R \in \mathrm{SO}(3) = \{R \in \mathbb{R}^{3\times 3} : RR^T = I, \det(R) = 1\}$ which describes the orientation of $\mathscr{B}$ with respect to $\mathscr{N}$. We use roll-pitch-yaw (ϕ-θ-ψ) or ZYX Euler

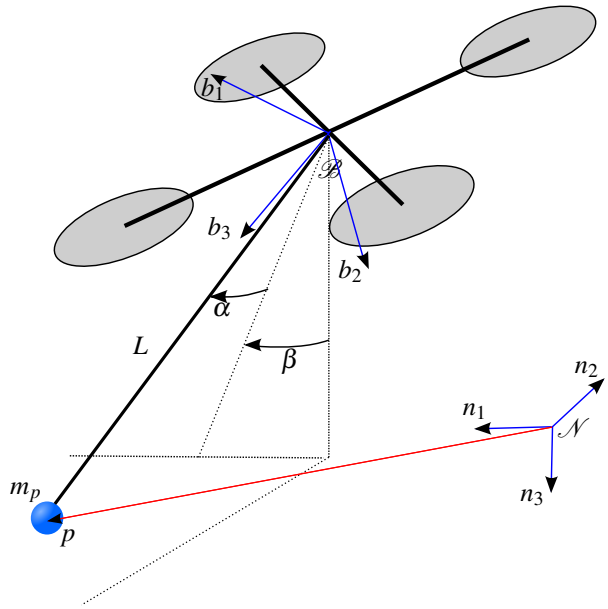


Fig. 1: The SLS system. The suspended load is modelled as a spherical pendulum attached to the UAV CoM.

angles to parametrize R:

$$R(\eta) = \begin{bmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\psi c_\phi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{bmatrix}$$

where $\eta = [\phi, \theta, \psi]^T$, $c_\theta = \cos\theta$, $s_\theta = \sin\theta$. The rotational dynamics for the quadrotor is

$$\dot{R} = RS(\omega) \tag{2a}$$

$$J\dot{\omega} = -S(\omega)J\omega + \tau \tag{2b}$$

where $\omega \in \mathbb{R}^3$ is angular velocity in $\mathscr{B}$, and $J \in \mathbb{R}^{3\times 3}$ is the inertia of the vehicle. To simplify presentation, the UAV mass is assumed distributed symmetrically and hence $J = \mathrm{diag}([J_1, J_2, J_3])$. The vector $\tau = [\tau_1, \tau_2, \tau_3]^T \in \mathbb{R}^3$ is the torque due to all propeller thrusts expressed in $\mathscr{B}$, the skew operator $S(\cdot) : \mathbb{R}^3 \to so(3)$ is

$$S(x) = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix}, \quad \text{where } x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

and $so(3) = \{M \in \mathbb{R}^{3\times 3} : M^T = -M\}$. The rotational kinematics (2a) expressed in η is

$$\dot{\eta} = W(\eta)\omega \tag{3}$$

with

$$W(\eta)=\begin{bmatrix}1 & s_\phi t_\theta & c_\phi t_\theta\\ 0 & c_\phi & -s_\phi\\ 0 & s_\phi/c_\theta & c_\phi/c_\theta\end{bmatrix}$$

where $t_\theta = \tan\theta$.

For a traditional quadrotor UAV, each propeller generates thrust in the $-b_3$ direction, and we denote the total thrust due to all propellers by the scalar input $\bar{u}\geq 0$, i.e., the total thrust vector is $F=-\bar{u}b_3=-R\bar{u}n_3$ and $\|F\|=\bar{u}$. The system input is $u=[\bar{u},\tau^\top]^\top\in\mathscr{U}\subset\mathbb{R}^4$. As is customary, design is performed for u which is algebraically related to the physical inputs which are the PWM inputs to the UAV Electronic Speed Controllers (ESCs).

The pendulum dynamics form a coupled subsystem whose configuration space is $\mathbb{R}^3\times\mathbb{S}^2$ for the pendulum position p and pendulum orientation parameterized by angles α,β. The angle of the pendulum relative to n_1 is α, and β is the angle relative to n_2 (see Fig. 1). For simplicity, the mass of the pendulum m_p is concentrated at its end and the rod of length L is assumed massless. As in [13], [12] we use the Euler-Lagrange equations to obtain the pendulum dynamics. This yields

$$M(q)\ddot{q}+C(q,\dot{q})\dot{q}+G(q)=B\bar{u} \tag{4}$$

where $q=[p^\top,\alpha,\beta]^\top$, M is a positive definite mass matrix, $C\dot{q}$ models the centrifugal and Coriolis effects, and G is the gravity vector. Expressions for M,C,G, and B are given in Appendix A where we have taken m as the UAV mass. Pendulum dynamics (4) can be combined with (2b) and (3) to obtain a control affine model with $m=4$ inputs and $n=16$ states. We obtain

$$\dot{x}=\begin{bmatrix} v\\ \gamma_\alpha\\ \gamma_\beta\\ W(\eta)\omega\\ -\frac{\mathrm{s}_\beta(\gamma_\alpha^2\mathrm{c}_\beta^2+\gamma_\beta^2)Lm}{m+m_p}\\ \frac{\mathrm{s}_\alpha\mathrm{c}_\beta(\gamma_\alpha^2\mathrm{c}_\beta^2+\gamma_\beta^2)Lm}{m+m_p}\\ g-\frac{\mathrm{c}_\alpha\mathrm{c}_\beta(\gamma_\alpha^2\mathrm{c}_\beta^2+\gamma_\beta^2)Lm}{m+m_p}\\ 2\gamma_\alpha\gamma_\beta\mathrm{t}_\beta\\ -\gamma_\alpha^2\mathrm{c}_\beta\mathrm{s}_\beta\\ -J^{-1}S(\omega)J\omega\end{bmatrix}+\begin{bmatrix}0_{8\times1} & 0_{8\times3}\\ \bar{g}(x) & 0_{5\times3}\\ 0_{3\times1} & J^{-1}\end{bmatrix}u \tag{5}$$

where $x=[p^\top,\alpha,\beta,\eta^\top,v^\top,\gamma_\alpha,\gamma_\beta,\omega^\top]^\top\in\mathbb{R}^{16}$, and we have defined $v=\dot{p},\gamma_\beta=\dot{\beta},\gamma_\alpha=\dot{\alpha}$. The expression for $\bar{g}\in\mathbb{R}^5$ is given in Appendix A. We remark that due to parametrization of the orientation of the UAV and pendulum, (5) includes singularities when $\mathrm{c}_\beta=\mathrm{c}_\theta=0$ and hence we define the domain $\mathscr{M}$ for the SLS state as a subset of $\mathbb{R}^{16}$ containing 0 and excluding these points. From a practical point of view, it is unlikely these singularities are of concern as they involve extreme motion not normally encountered during the safe operation of the SLS. Further, $\mathrm{c}_\beta=0$ (or $\mathrm{c}_\alpha=0$) is impossible without the pendulum colliding with the UAV. Our control objective is to achieve asymptotic tracking of a general smooth reference output y_d for pendulum position and yaw angle. Hence, the output is taken as

$$y=h(x)=[p^\top,\psi]^\top \tag{6}$$

III. SLS MOTION CONTROL

A. Dynamic Extension Algorithm (DEA)

In this subsection, we review the Dynamic Extension Algorithm (DEA) which is applied below in Section III-B to (5)-(6) to solve the output tracking problem with dynamic state feedback linearization.

The Lie derivative of a function $\lambda:\mathscr{M}\to\mathbb{R}$ along the vector field f is defined by $L_f\lambda(x)=\frac{\partial\lambda}{\partial x}f(x)$. When performing the DEA we make use of a vector of indices $r=[r_1,\dots r_m]$ such that r_i is the largest integer satisfying $L_{g_j}L_f^k h_i(x)=0, 1\leq j\leq m, k<r_i-1$, about some $x_0\in\mathscr{M}$. The existence of these indices does not imply the system has a well-defined relative degree about x_0 as this requires the decoupling matrix

$$A(x)=\begin{bmatrix}L_{g_1}L_f^{r_1-1}h_1(x) & \dots & L_{g_m}L_f^{r_1-1}h_1(x)\\ \vdots & \dots & \vdots\\ L_{g_1}L_f^{r_m-1}h_m(x) & \dots & L_{g_m}L_f^{r_m-1}h_m(x)\end{bmatrix} \tag{7}$$

to be nonsingular at x_0.

For the SLS model (5), (6) we compute $r=[2,2,2,2]$ and

$$A(x)=\begin{bmatrix}a_{11}(x) & 0 & 0 & 0\\ a_{21}(x) & 0 & 0 & 0\\ a_{31}(x) & 0 & 0 & 0\\ 0 & 0 & \frac{\mathrm{s}_\phi}{J_2\mathrm{c}_\theta} & \frac{\mathrm{c}_\phi}{J_3\mathrm{c}_\theta}\end{bmatrix} \tag{8}$$

where a_{11},a_{21},a_{31} are functions of state. On the region where

$$a_{31}(x)=-\frac{[\mathrm{s}_\beta,-\mathrm{s}_\alpha\mathrm{c}_\beta,\mathrm{c}_\alpha\mathrm{c}_\beta]\cdot Rn_3}{m+m_p}\neq 0 \tag{9}$$

we obtain $\mathrm{rank}(A(x))=2$. Clearly, relative degree does not exist at any point in $\mathscr{M}$. This implies the SLS cannot be input-output state feedback linearized with output (6). However, using the DEA we will construct an extended dynamics with full relative degree (i.e., $\sum_{i=1}^m r_i$ for the extended dynamics equals the dimension of the extended state).

Following [18] we outline the DEA for (1). Variables have superscript $\langle i\rangle$ to keep track of the algorithm iteration. We begin with Step 1 and assume the decoupling matrix $A^{\langle 0\rangle}=A$ given by (7) has constant rank less than m about $x_0\in\mathscr{M}$ where $r^{\langle 0\rangle}=[r_1,\dots,r_m]$. This implies there exists one or more rows of $A^{\langle 0\rangle}$ which can be expressed as a linear combination of the others. Let $a_i, 1\leq i\leq m$ denote the rows of $A^{\langle 0\rangle}$. We reorder (if necessary) the rows of $A^{\langle 0\rangle}$ such that its first $k-1$ rows are independent at x_0. The remaining $m-k+1$ rows are dependent and hence a_j for some $j\geq k$ can be written as a linear combination of the first $k-1$ rows. That is, there exists $k-1$ smooth functions $c_1(x),\dots,c_{k-1}(x)$ such that $a_j(x)=\sum_{i=1}^{k-1}c_i(x)a_i(x)$. Hence, there exists integers

$(i_0, j_0), 1 \le i_0 \le k-1$ such that c_{i_0} is not identically zero and for Step i we have

$$a_{i_0 j_0}^{\langle i-1\rangle}(x_0^{\langle i-1\rangle}) = L_{g_{j_0}^{\langle i-1\rangle}} L_{f^{\langle i-1\rangle}}^{r_{i_0}^{\langle i-1\rangle}-1} h_{i_0}(x_0^{\langle i-1\rangle}) \neq 0$$

where quantities with $^{\langle i-1\rangle}$ come from the previous step. That is, $f^{\langle i-1\rangle}(x^{\langle i-1\rangle}), g_j^{\langle i-1\rangle}(x^{\langle i-1\rangle}), 1 \le j \le m$ denote the system vector fields from the previous step with $g_j^{\langle 0\rangle} = g_j$ and $f^{\langle 0\rangle} = f$ are from the original dynamics (1). State $x^{\langle i-1\rangle}$ corresponds to the system from the previous step and $x^{\langle 0\rangle} = x$ (see (11) for $i > 1$). At Step i we define the dynamic state feedback $v^{\langle i\rangle}$ using

$$\begin{aligned} v_j^{\langle i\rangle} &= v_j^{\langle i-1\rangle}, \quad 1 \le j \le m, j \neq j_0 \\ v_{j_0}^{\langle i-1\rangle} &= \frac{1}{a_{i_0 j_0}^{\langle i-1\rangle}}(k^{\langle i\rangle} + s^{\langle i\rangle}\xi_i - \sum_{\substack{j=1 \\ j\neq j_0}}^{m} a_{i_0 j}^{\langle i-1\rangle} v_j^{\langle i-1\rangle}) \\ \dot{\xi}_i &= v_{j_0}^{\langle i\rangle} \end{aligned} \tag{10}$$

where $v^{\langle 0\rangle} = u$ is the input for (1), $\xi_i \in \mathbb{R}$ is a controller state, and $k^{\langle i\rangle}$ and $s^{\langle i\rangle}$ are any smooth functions such that $k^{\langle i\rangle}(x_0^{\langle i-1\rangle}) = 0$ and $s^{\langle i\rangle}(x_0^{\langle i-1\rangle}) = 1$. At Step i we create a new state vector $x^{\langle i\rangle}$ by concatenating the state of the previous iteration $x^{\langle i-1\rangle}$ with controller state ξ_i:

$$x^{\langle i\rangle} = [x^{\top\langle i-1\rangle}, \xi_i]^\top \in \mathscr{M} \times \mathbb{R}^i, i \ge 1 \tag{11}$$

Applying (10) to (1) defines new system vector fields $f^{\langle i\rangle}(x^{\langle i\rangle}), g_j^{\langle i\rangle}(x^{\langle i\rangle}), 1 \le j \le m$ defined on an extended state space $\mathscr{M} \times \mathbb{R}^i$ and with new input $v^{\langle i\rangle} = [v_1^{\langle i\rangle}, \ldots, v_m^{\langle i\rangle}]^\top$. That is,

$$\dot{x}^{\langle i\rangle} = f^{\langle i\rangle}(x^{\langle i\rangle}) + \sum_{j=1}^{m} g_j^{\langle i\rangle}(x^{\langle i\rangle}) v_j^{\langle i\rangle}$$

where, for $i > 0$, $f^{\langle i\rangle}$, and $g^{\langle i\rangle}$ are given by

$$\begin{aligned} f^{\langle i\rangle} &= [(f^{\langle i-1\rangle} + \frac{k^{\langle i\rangle} + \xi_i s^{\langle i\rangle}}{a_{i_0 j_0}^{\langle i-1\rangle}} g_{j_0}^{\langle i-1\rangle})^\top, 0]^\top \\ g_{j_0}^{\langle i\rangle} &= [0_{1\times(n+i-1)}, 1]^\top \\ g_j^{\langle i\rangle} &= [(g_j^{\langle i-1\rangle} - \frac{g_{j_0}^{\langle i-1\rangle} a_{i_0 j}^{\langle i-1\rangle}}{a_{i_0 j_0}^{\langle i-1\rangle}})^\top, 0]^\top, \quad 1 \le j \le m, j \neq j_0 \end{aligned}$$

The DEA iteration proceeds to Step $i+1$ if $A^{\langle i\rangle}$ has constant rank less than m at $[x_0^\top, 0_{1\times i}]^\top$. If rank of $A^{\langle i\rangle}$ is m, the extended dynamics has a well-defined relative degree $r^{<i>}$ and the algorithm terminates.

B. Dynamic Full State Feedback Linearization of the SLS

In this section, we apply the DEA to the SLS model (5)-(6). We show that after four iterations the algorithm converges to an extended dynamics with full relative degree of 20 which is the dimension of the extended state. In Section III-C we augment the state with the integral of load position and apply a static state feedback linearization to achieve the output tracking control objective.

We pick the point $x_0 = 0$ for practical reasons as we require an extended dynamics with well-defined relative degree in hover. The region on which relative degree is defined includes x_0 and allows for trajectory tracking is discussed further in Section III-D.

Due to the structure of the decoupling matrix, we obtain a unique $(i_0, j_0) = (3,1)$ at each DEA step. In order to simplify the extended system dynamics, the functions k and s are chosen as

$$k^{\langle i\rangle}(x^{\langle i-1\rangle}) = -L_{f^{\langle i-1\rangle}}^{r_3^{\langle i-1\rangle}} h_3(x^{\langle i-1\rangle}), \quad s^{\langle i\rangle}(x^{\langle i-1\rangle}) = 1 \tag{12}$$

We remark this choice is important as other choices lead to very large expressions for the extended dynamics which cannot be computed using symbolic math tools in a reasonable amount of time. Using (12) we obtain

$$\begin{aligned} f^{\langle i\rangle} &= [(f^{\langle i-1\rangle} + \frac{k^{\langle i\rangle} + \xi_i s^{\langle i\rangle}}{a_{31}^{\langle i-1\rangle}} g_1^{\langle i-1\rangle})^\top, 0]^\top \\ g_1^{\langle i\rangle} &= [0_{1\times(15+i)}, 1]^\top \\ g_j^{\langle i\rangle} &= [g_j^{\langle i-1\rangle\top}, 0]^\top, \text{for } j = 2,3,4 \end{aligned} \tag{13}$$

for $i = 1,2,3,4$, where $f^{\langle 0\rangle}(x) = f(x)$ and $g^{\langle 0\rangle}(x) = g(x)$ are from (5), and $a_{31}^{\langle i-1\rangle}$ is the $(3,1)$ entry of $A^{\langle i-1\rangle}$.

The decoupling matrices

$$A^{\langle i\rangle}(x^{\langle i\rangle}) = \begin{bmatrix} \frac{\mathrm{t}_\beta}{\mathrm{c}_\alpha} & 0 & 0 & 0 \\ -\mathrm{t}_\alpha & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & \frac{\mathrm{s}_\phi}{J_2 \mathrm{c}_\theta} & \frac{\mathrm{c}_\phi}{J_3 \mathrm{c}_\theta} \end{bmatrix}, \quad i = 1,2,3$$

The structure of $A^{\langle i\rangle}$ is consistent with $r^{\langle i\rangle} = [i+2, i+2, i+2, 2], 1 \le i \le 3$. At each iteration we delay the appearance of the thrust input by defining a new controller state. Therefore, it takes another time derivative of the output for the thrust to appear when using the extended state and extended dynamics.

Using (10) we obtain the dynamic state feedback

$$\begin{aligned} v_j^{\langle i\rangle} &= v_j^{\langle i-1\rangle} = \tau_j, \quad \text{for } j = 2,3,4 \\ v_1^{\langle i-1\rangle} &= \frac{1}{a_{31}^{\langle i-1\rangle}}(\xi_i - L_{f^{\langle i-1\rangle}}^{r_3^{\langle i-1\rangle}} h_3) \\ \dot{\xi}_i &= v_1^{\langle i\rangle} \end{aligned} \tag{14}$$

where $i = 1,2,3,4$. Hence, we observe that only the thrust component of the input is modified.

At Step 4, due to the appearance of τ in the time derivatives of y_1, y_2, we obtain

$$A^{\langle 4\rangle} = \begin{bmatrix} a_{11}^{\langle 4\rangle} & a_{12}^{\langle 4\rangle} & a_{13}^{\langle 4\rangle} & 0 \\ a_{21}^{\langle 4\rangle} & a_{22}^{\langle 4\rangle} & a_{23}^{\langle 4\rangle} & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & a_{43}^{\langle 4\rangle} & a_{44}^{\langle 4\rangle} \end{bmatrix} \tag{15}$$

with $\mathrm{rank}(A^{\langle 4\rangle}(x)) = 4$ on a subset of $\mathscr{M}$. Since $r^{\langle 4\rangle} = [6,6,6,2]$, Step 4 achieves full relative degree of 20 which is the dimension of $x^{\langle 4\rangle}$.

C. Static State Feedback Linearization for Robust Output Tracking

In this section we apply static state feedback linearization to the extended dynamics computed in the previous subsection. We augment the extended state with the integral of pendulum position to improve robustness of the closed-loop. For convenience we define the input of the extended dynamics as $\bar{\nu} = [\nu^{\langle 4\rangle}, \tau^\top]^\top$, $\bar{f} = f^{\langle 4\rangle}$, $\bar{\mathscr{M}} = \mathscr{M} \times \mathbb{R}^4 \subset \mathbb{R}^{20}$, $\bar{r} = r^{\langle 4\rangle}$, and $\bar{A} = A^{\langle 4\rangle}$.

First we define tracking error coordinates

$$\begin{aligned}\tilde{z} = [\int_0^t (h_1 - y_{d1})d\tau, h_1 - y_{d1}, L_{\bar{f}}h_1 - \dot{y}_{d1}, \ldots,\\ L_{\bar{f}}^5 h_1 - y_{d1}^{(5)}, \ldots, \int_0^t (h_3 - y_{d3})d\tau, \ldots,\\ L_{\bar{f}}^5 h_3 - y_{d3}^{(5)}, h_4 - y_{d4}, L_{\bar{f}}h_4 - \dot{y}_{d4}]^\top\end{aligned} \tag{16}$$

We have $\tilde{z} \in \mathbb{R}^{23}$, where 16 components come from the original system dynamics (5), four components are added as controller states in the DEA, and three from integral augmentation of the pendulum position. We can write the error dynamics in $\tilde{z}$-coordinates as

$$\dot{\tilde{z}} = A_c\tilde{z} + B_c(b + \bar{A}\bar{\nu}) \tag{17}$$

where

$$\begin{aligned}A_c &= \operatorname{diag}(A_1, A_2, A_3, A_4)\\ B_c &= \begin{bmatrix} e_7 & e_{14} & e_{21} & e_{23}\end{bmatrix}\\ b &= [L_{\bar{f}}^6 h_1, L_{\bar{f}}^6 h_2, L_{\bar{f}}^6 h_3, L_{\bar{f}}^2 h_4]^\top\end{aligned}$$

$e_i \in \mathbb{R}^{23}$ denotes the unit vector in the ith direction, and for $i = 1,2,3$ we have

$$A_i = \begin{bmatrix} 0_{6\times 1} & I_6 \\ 0 & 0_{1\times 6}\end{bmatrix} \in \mathbb{R}^{7\times 7}, \text{ and } A_4 = \begin{bmatrix} 0 & 1 \\ 0 & 0\end{bmatrix}$$

Applying the static state linearizing control

$$\bar{\nu} = \bar{A}^{-1}(K\tilde{z} - b + y_d^{(\bar{r})}) \tag{18}$$

with $y_d^{(\bar{r})} = [y_{d1}^{(6)}, y_{d2}^{(6)}, y_{d3}^{(6)}, y_{d4}^{(2)}]^\top$ to (17) gives

$$\dot{\tilde{z}} = (A_c + B_cK)\tilde{z} \tag{19}$$

where $K \in \mathbb{R}^{4\times 23}$ is a control gain chosen such that $A_c + B_cK$ is Hurwitz and suitable tracking error transient performance is obtained.

A related result given in [19] where an output tracking control for the SLS is derived using quasi-static state feedback linearization. This control also achieves LTI error dynamics, but without requiring controller states. Hence, the output tracking state feedback is a static function of the original system state.

D. Domain of the Output Tracking Controller

In this section we determine the domain of the linearizing control law (18). Since the $\tilde{z}$-coordinates (16) are defined using the Lie derivatives of h along $\bar{f}$, some of singular points of the control law are inherited from those in f. These points include $\theta = \pm 90°$, $\beta = \pm 90°$ which are seen in the original dynamics (5) and which arise due to the Euler angles. The points $\alpha = \pm 90°$ are singularities that appear in $\bar{f}$ due to the dynamic extension. The expression for $\bar{f}$ is omitted as is it too large.

The control is also singular at points where the Jacobian matrix of $\tilde{z}$-coordinates is singular. These points are the same as those where the distribution rank condition in [20] does not hold. Equivalently, we can obtain these points from where the decoupling matrix $\bar{A}$ is singular. These points are

$$\phi = \pm 90° \tag{20a}$$

$$[\mathrm{s}_\beta, -\mathrm{s}_\alpha \mathrm{c}_\beta, \mathrm{c}_\alpha \mathrm{c}_\beta] \cdot Rn_3 = 0 \tag{20b}$$

$$\xi_1 = g - \frac{\mathrm{c}_\alpha \mathrm{c}_\beta (\gamma_\alpha^2 \mathrm{c}_\beta^2 + \gamma_\beta^2) Lm}{m + m_p} \tag{20c}$$

$$\xi_1 = g \tag{20d}$$

We remark that the geometric interpretation of the LHS of (20b) (which from (9) is a scaling of a_{31}) is the inner product of the direction vector of the pendulum $[\mathrm{s}_\beta, -\mathrm{s}_\alpha \mathrm{c}_\beta, \mathrm{c}_\alpha \mathrm{c}_\beta]$ and the direction of the UAV thrust vector Rn_3 with both vectors expressed in $\mathscr{N}$. Thus, a physical singularity appears when the direction of pendulum is perpendicular to the thrust vector. In Step 1 of the DEA we define a controller state $\xi_1 = L_{\bar{f}}^2 h_3(x) + a_{31}(x)\bar{u}$ (see (14) for $i = 1$). Hence, condition (20c) is equivalent to $\bar{u} = 0$. The condition (20d) occurs when the downwards linear acceleration of the pendulum $\ddot{p}_3 = g$. This is another physical singularity which appears in many UAV motion controllers, e.g., [21]. Hence, we remark that the domain of the controller is a suitable subset of $\bar{\mathscr{M}}$ which excludes the abovementioned points. This domain is practically sensible in that singularities only occur in conditions that would not be typical of safe operation.

IV. DEA in Symbolic Math

Symbolic math software Maple [16] was used to implement the DEA for a general control-affine system (1). The code is available at [22] and was inspired by work in [23]. However, the approach in [23] did not implement the DEA correctly for a number of reasons. First, it only performed one DEA step. Second, the method for computing indices i_0, j_0 was incorrect. For example, the first and fourth inputs were extended simultaneously in Step 1 when applied to (5)-(6). Unlike the work of [23] where depreciated packages were used (e.g., `linalg`) our code uses only currently supported packages (e.g., `LinearAlgebra` and their data structures). We use the `CodeGeneration` package to optimize the control law expressions for Matlab simulation. This is important as it allows us to generate an expression for the control in less than a minute on a standard CPU (Intel i9-10900K, 3.70 GHz). As the approach [23] did not perform this optimization, it would yield large expressions which could not be used for simulation. Another feature of our work is that it allows user input at each DEA step so that functions k and s can be adjusted. This is important as we have observed that for certain common choices (e.g., $k(x) = 0$ and $s(x) = 1$)), controller expressions are so large Maple cannot provide a result in a reasonable amount of

time. Lastly, our simulations are performed from Maple using Matlab by code generating the simulation files in Maple. Matlab commands are run using Maple's `Matlab` package. Four example systems are provided including the SLS to demonstrate the DEA.

V. Simulations

In this section we validate the proposed control law in simulation and compare its performance with an existing method [14]. We consider simulations for stabilization and tracking of the pendulum position and the yaw angle. The system parameters used in the model and controller are in Table I.

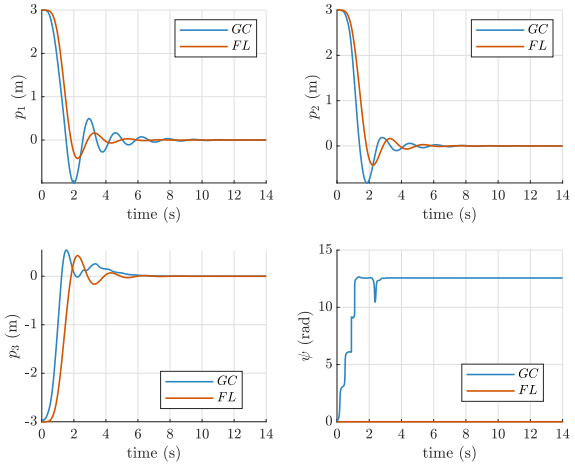
TABLE I: System parameters.

m	1.6 kg
m_p	0.16 kg
L	1 m
J_1	0.03 kg·m^2
J_2	0.03 kg·m^2
J_3	0.05 kg·m^2

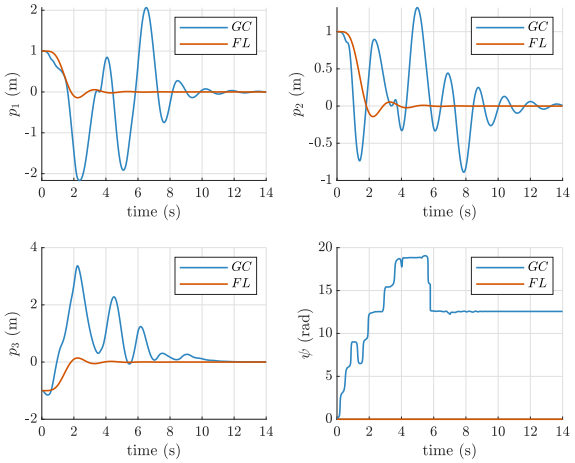
A. Position Stabilization Performance Comparison

We consider the problem of stabilizing $x = 0$ and choose reference output $y_d = 0$. We initialize the system with $p(0) \neq 0$ and all other states zero (i.e., the SLS is at rest). Since the error dynamics (19) is LTI, it is straightforward to design the transient performance for y. Using linear system theory we design K for a 10%-settling time of 3 s and an overshoot of 15%. Fig. 2(a) shows the response for p in red where the performance specifications are met as expected. To compare the performance with the proposed design, we consider the same problem using the geometric controller (GC) in [14]. We remark that the GC is a 3-level nested controller and guarantees almost-global *exponential attractiveness* [24, Def. 1], but ES occurs on a limited region. As well, the error dynamics is nonlinear and this makes gain tuning difficult and non-systematic. Hence, the GC gains were chosen by trail-and-error to yield a response that is as as close as possible to the desired transient specifications. The trajectories for p are in blue in Fig. 2(a).

Since for the proposed design $x = 0$ is ES, when the initial position is closer to the origin and controller gains are unchanged, we obtain a lower bound on $\|p\|$. Hence, when we initialize $p(0)$ closer to the 0 we expect a smaller transient. This is confirmed in simulation in Fig. 2(b). However, since ES is not guaranteed with the GC, the transient performance for y is severely degraded for smaller $p(0)$. We observe that although $\psi(0) = 0$, the UAV performs two full yaw rotations before it converges. As well, the transient performance in position is very oscillatory. The poor transient performance of the GC is attributed to its nonlinear error dynamics whose equilibrium is locally ES. Using [14, Eqns. (39), (45)], for the simulated initial conditions and controller gains, an estimated ES region of attraction (ROA) was computed which shows that ES is only locally ensured and a complex function of controller gain. In particular, the ES ROA does not include $\alpha = \beta = 0$ which implies ES is not ensured for the simulated conditions. This should be compared to the proposed controller whose ES ROA includes $\alpha = \beta = 0$.



(a) Initial position further from origin: $\|p(0)\| = 5.83$ m.



(b) Initial position closer to origin: $\|p(0)\| = 2.45$ m.

Fig. 2: Position trajectories for the proposed design (FL), and the design in [14] (GC). The same gains are used for both simulations.

B. Time-varying Trajectory Tracking

The proposed controller can achieve output tracking for complex trajectories. Fig. 3 illustrates a 3-D plot of the figure-8 reference trajectory for position. The reference involves a constant yaw. The simulated input trajectories are given in Fig. 4. The tracking error and configurations variables are in Figs 5, and 6. We observe that tracking error converges to 0 with acceptable transient performance and input magnitudes. The design for gain K was performed in a straightforward manner using LQR.

VI. Conclusions

This paper presented an output tracking controller for an SLS using dynamic state feedback linearization. Four

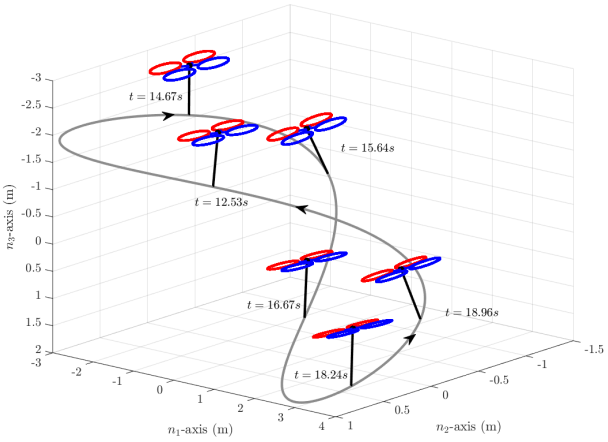


Fig. 3: 3-D plot of output tracking.

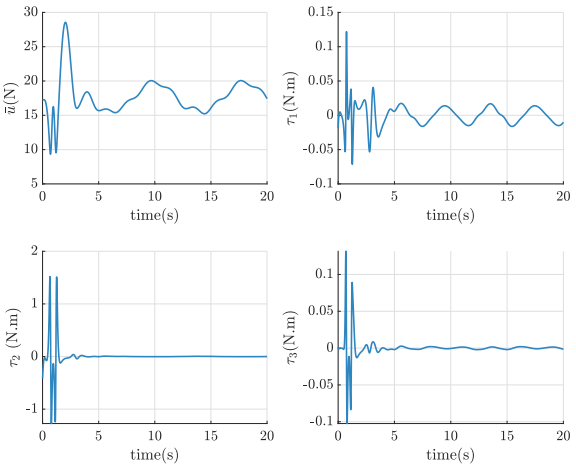


Fig. 4: Output tracking: Input trajectories.

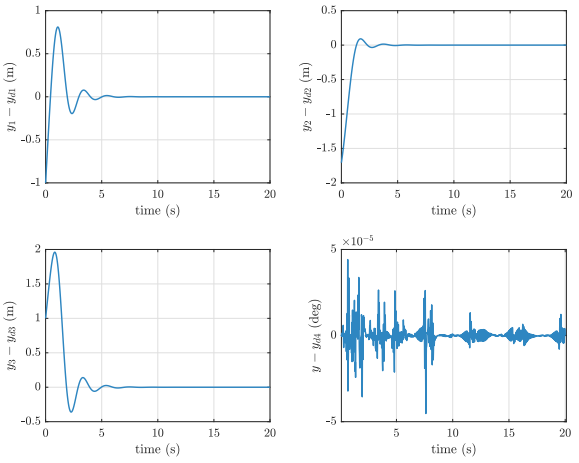


Fig. 5: Output tracking: Output error.

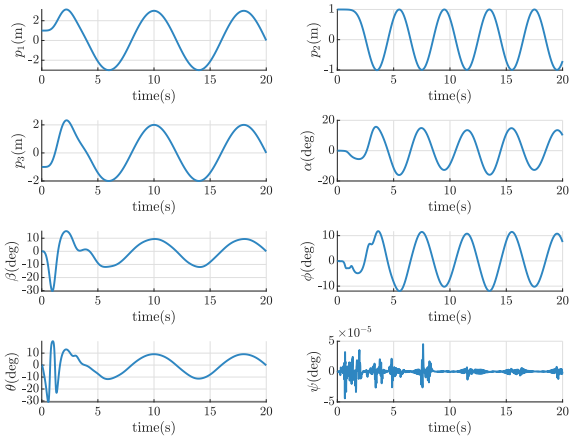


Fig. 6: Output tracking: Configuration variables.

iterations of the DEA are used to derive the control law. Hence, the dynamic state feedback introduces an additional 4 controller states. These states are related to UAV thrust and its time derivatives up to order 3. The DEA constructs an extended dynamics with a full relative degree and hence static state feedback linearizable with an auxiliary control. The extended state is augmented with the integral of pendulum position and static state feedback linearization achieves LTI ES error dynamics. Linearity is an important property of the design as it facilitates gain selection and stability analysis. Simulations confirm the design's ES property and ease of gain tuning. Performance is compared with an existing design [14]. The proposed design is implemented using symbolic math software which can be applied to generic nonlinear control-affine systems and available at [22].

Appendix A
Modelling

The mass matrix M in (4) is

$$M(q) = \left[\begin{array}{c|cc} & 0 & -Lmc_\beta \\ (m+m_p)I_3 & Lmc_\beta c_\alpha & -Lms_\beta s_\alpha \\ & Lms_\alpha c_\beta & Lms_\beta c_\alpha \\ \hline & L^2mc_\beta^2 & 0 \\ * & 0 & L^2m \end{array}\right]$$

where the submatrix denoted by * can be determined by symmetry. The (k,j) element of C is obtained from

$$C_{kj} = \sum_{i=1}^{5}\left(\frac{\partial M_{kj}}{\partial q_i} + \frac{\partial M_{ki}}{\partial q_j} + \frac{\partial M_{ij}}{\partial q_k}\right)\dot{q}_i$$

where M_{kj} is the (k,j) element of M. This yields

$$C(q,\dot{q}) = -Lm\left[\begin{array}{c|cc} & 0 & -s_\beta\dot{\beta} \\ 0 & c_\alpha s_\beta\dot{\beta} + c_\beta s_\alpha\dot{\alpha} & s_\beta c_\alpha\dot{\alpha} + c_\beta s_\alpha\dot{\beta} \\ & s_\beta s_\alpha\dot{\beta} - c_\beta c_\alpha\dot{\alpha} & s_\beta s_\alpha\dot{\alpha} - c_\beta c_\alpha\dot{\beta} \\ & Ls_\beta c_\beta\dot{\beta} & Ls_\beta c_\beta\dot{\alpha} \\ & -Lc_\beta s_\beta\dot{\alpha} & 0 \end{array}\right]$$

The gravity vector is

$$G(q) = -g\begin{bmatrix}0 & 0 & (m+m_p) & Lmc_\beta s_\alpha & Lms_\beta c_\alpha\end{bmatrix}^\top$$

and

$$B\bar{u} = -\frac{m_p}{m}Rn_3\bar{u} = \frac{m_p}{m}[F^\top, 0, 0]^\top$$

The components for the pendulum dynamics in (5) can be obtained from (4). The drift vector field components are

$$-M^{-1}(C(q,\dot{q})q + G(q)) = \begin{bmatrix} -\frac{s_\beta(\gamma_\alpha^2 c_\beta^2+\gamma_\beta^2)Lm}{m+m_p} \\ \frac{s_\alpha c_\beta(\gamma_\alpha^2 c_\beta^2+\gamma_\beta^2)Lm}{m+m_p} \\ g - \frac{c_\alpha c_\beta(\gamma_\alpha^2 c_\beta^2+\gamma_\beta^2)Lm}{m+m_p} \\ 2\gamma_\alpha\gamma_\beta t_\beta \\ -\gamma_\alpha^2 c_\beta s_\beta \end{bmatrix}$$

and the input vector field components are

$$\bar{g}(x) = M^{-1}B = \begin{bmatrix} -\frac{w_1+w_2 s_\beta^2}{m+m_p} \\ \frac{(w_3+w_4)c_\beta}{m+m_p} \\ \frac{(w_5-w_6)c_\alpha c_\beta}{m+m_p} \\ \frac{w_7}{Lmc_\beta} \\ \frac{w_8+w_9}{Lm} \end{bmatrix}$$

where

$$\begin{aligned}
w_1 &= c_\beta s_\beta\left(\left(c_\alpha c_\theta - s_\alpha s_\theta s_\psi\right)c_\phi + c_\psi s_\alpha s_\phi\right)\\
w_2 &= c_\phi s_\theta c_\psi + s_\phi s_\psi\\
w_3 &= c_\beta\left(\left(c_\alpha^2 s_\theta s_\psi + c_\alpha c_\theta s_\alpha - s_\psi s_\theta\right)c_\phi + c_\psi s_\phi s_\alpha^2\right)\\
w_4 &= s_\beta s_\alpha\left(c_\phi s_\theta c_\psi + s_\phi s_\psi\right)\\
w_5 &= c_\beta\left((s_\alpha s_\theta s_\psi - c_\alpha c_\theta)c_\phi - c_\psi s_\alpha s_\phi\right)\\
w_6 &= s_\beta\left(c_\phi s_\theta c_\psi + s_\phi s_\psi\right)\\
w_7 &= c_\psi c_\alpha s_\phi - c_\phi\left(s_\theta c_\alpha s_\psi + s_\alpha c_\theta\right)\\
w_8 &= c_\phi\left(s_\beta\left(s_\alpha s_\theta s_\psi - c_\alpha c_\theta\right) + c_\beta c_\psi s_\theta\right)\\
w_9 &= s_\phi\left(c_\beta s_\psi - c_\psi s_\alpha s_\beta\right)
\end{aligned}$$

REFERENCES

[1] D. K. D. Villa, A. S. Brandao, and M. Sarcinelli-Filho, "A survey on load transportation using multirotor UAVs," *J. Intell. Robot. Syst.*, vol. 98, no. 2, pp. 267–296, 2020.

[2] Q. Lindsey, D. Mellinger, and V. Kumar, "Construction of cubic structures with quadrotor teams," in *Robotics: Science & Systems VII*. MIT Press, 2011, pp. 177–184.

[3] P. E. I. Pounds, D. R. Bersak, and A. M. Dollar, "Grasping from the air: Hovering capture and load stability," in *Proc. IEEE Int. Conf. on Robotics and Automation*, Toronto, ON, 2011, pp. 2491–2498.

[4] A. Gawel, M. Kamel, T. Novkovic, J. Widauer, D. Schindler, B. P. von Altishofen, R. Siegwart, and J. Nieto, "Aerial picking and delivery of magnetic objects with MAVs," in *Proc. IEEE Int. Conf. on Robotics and Automation*, Singapore, 2017, pp. 5746–5752.

[5] M. Bernard and K. Kondak, "Generic slung load transportation system using small size helicopters," in *Proc. IEEE Int. Conf. on Robotics and Automation*, Kobe, Japan, 2009, pp. 3258–3264.

[6] D. Fusato, G. Guglieri, and R. Celi, "Flight dynamics of an articulated rotor helicopter with an external slung load," *J. Am. Helicopter Soc.*, vol. 46, no. 1, pp. 3–13, 2001.

[7] A. Irscheid, M. Konz, and J. Rudolph, "A flatness-based approach to the control of distributed parameter systems applied to load transportation with heavy ropes," in *Advanced Control Techniques in Complex Engineering Systems: Theory and Applications*. Springer, 2019, pp. 279–294.

[8] R. M. Murray, "Trajectory generation for a towed cable system using differential flatness," *IFAC Proceedings Volumes*, vol. 29, no. 1, pp. 2792–2797, 1996.

[9] F. A. Goodarzi, D. Lee, and T. Lee, "Geometric stabilization of a quadrotor UAV with a payload connected by flexible cable," in *Proc. American Control Conf*, Portland, OR, 2014, pp. 4923–4930.

[10] P. Kotaru, G. Wu, and K. Sreenath, "Differential-flatness and control of quadrotor(s) with a payload suspended through flexible cable(s)," in *Proc. Indian Control Conf.*, Kanpur, India, 2018, pp. 352–357.

[11] M. Guerrero-Sánchez, R. Lozano, P. Castillo, O. Hernández-González, C. García-Beltrán, and G. Valencia-Palomo, "Nonlinear control strategies for a UAV carrying a load with swing attenuation," *Appl. Math. Model.*, vol. 91, pp. 709–722, 2021.

[12] M. E. Guerrero, D. Mercado, R. Lozano, and C. García, "Swing-attenuation for a quadrotor transporting a cable suspended payload," *ISA T.*, vol. 68, pp. 433–449, 2017.

[13] K. Klausen, T. I. Fossen, and T. A. Johansen, "Nonlinear control with swing damping of a multirotor UAV with suspended load," *J. Intell. Robot. Syst.*, vol. 88, pp. 379–394, 2017.

[14] K. Sreenath, T. Lee, and V. Kumar, "Geometric control and differential flatness of a quadrotor UAV with a cable-suspended load," in *Proc. IEEE Int. Conf. on Decision and Control*, Firenze, Italy, 2013, pp. 2269–2274.

[15] K. Sreenath, N. Michael, and V. Kumar, "Trajectory generation and control of a quadrotor with a cable-suspended load-a differentially-flat hybrid system," in *Proc. IEEE Int. Conf. on Robotics and Automation*, Karlsruhe, Germany, 2013, pp. 4888–4895.

[16] "Maple," accessed May 13, 2022. [Online]. Available: https://www.maplesoft.com/products/maple/

[17] H. Xie, "Dynamic visual servoing of rotary wing unmanned aerial vehicles," Ph.D. dissertation, Dept. of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, 2016.

[18] H. Nijmeijer and W. Respondek, "Dynamic input-output decoupling of nonlinear control systems," *IEEE Trans. Automat. Contr.*, vol. 33, no. 11, pp. 1065–1070, 1988.

[19] Z. Jiang, A. Mohammadhasani, M. A. Lawati, and A. F. Lynch, "Flatness-based motion control of a UAV slung load system using quasi-static feedback linearization," in *Proc. IEEE Int. Conf. on Unmanned Aircraft Systems*, Dubrovnik, Croatia, 2022.

[20] H. Nijmeijer and J. Schumacher, "The regular local noninteracting control problem for nonlinear control systems," *SIAM J. Control Optim.*, vol. 24, no. 6, pp. 1232–1245, 1986.

[21] A. Moeini, A. F. Lynch, and Q. Zhao, "A backstepping disturbance observer control for multirotor UAVs: theory and experiment," *Int. J. Control*, pp. 1–15, 2021, online access.

[22] A. Mohammadhasani, "Maple codes for dynamic extension algorithm (DEA)," 2021, accessed May 13, 2022. [Online]. Available: https://github.com/ANCL/DSFBL.git

[23] G. Fischer, "Nonlincon: symbolic analysis and design package for nonlinear control systems," Master's thesis, Eindhoven University of Technology, Eindhoven, 1994.

[24] T. Lee, M. Leok, and N. H. McClamroch, "Geometric tracking control of a quadrotor UAV on SE(3)," in *Proc. IEEE Int. Conf. on Decision and Control*, Atlanta, GA, 2010, pp. 5420–5425.