Nonlinear Trajectory Control of Multi-body Aerial Manipulators

Marin Kobilarov
Laboratory for Computational Sensing and Robotics
Johns Hopkins University

Abstract

This paper studies trajectory control of aerial vehicles equipped with robotic manipulators. The proposed approach employs free-flying multi-body dynamics modeling and backstepping control to develop stabilizing control laws for a class of underactuated aerial systems. Two control methods are developed: coordinate-based and coordinate-free which are both generally applicable to aerial manipulation tasks. A simulated hexrotor vehicle equipped with a simple manipulator is employed to demonstrate the proposed techniques.

1 Introduction and Related Work

Motivated by recent progress in aerial robotics this paper considers the trajectory control of articulated flying mechanisms capable of performing aerial manipulation tasks. Aerial systems equipped with manipulator arms have a number of potential applications, e.g. to pick-up and transport vital supplies or to reach difficult-to-access locations and perform emergency repairs. The ability to grasp and transport objects has recently been explored using small autonomous helicopters [1, 2] operating outdoors and using multiple coordinated quadrotors [3] to assemble indoor structures. A related problem is balancing an inverted rigid mass [4]. Equipping aerial vehicles with more complex multi-degree of freedom manipulators remains challenging due their limited payload capacity and inherent flight instability. Such issues are currently being explored in the context of the Mobile Manipulating Unmanned Aerial Vehicle (MM-UAV) project [5, 6, 7] and are one of the main focii in the recently established Airobots project [8] and (Aerial Robotics Cooperative Assembly System) ARCAS project [9]. A related problem studied previously deals with the dynamics of helicopters with external slung loads (e.g. [10, 11]). Various aerial manipulation aspects are currently being considered, ranging from the ability to generate dynamic maneuvers mimicking avian grasping [12], specifically designing vehicles to exploit contact with the environment [13], or investigating haptic teleoperation [14]. Recent work more specifically focuses on the stability control of multi degree-of-freedom aerial manipulators using linear control techniques [15] or nonlinear variable parameter backstepping [16].

Motivated by these developments this work proposes a general nonlinear control strategy for aerial vehicles equipped with one or more articulated manipulators. A standard model simplification is to ignore rotational cross-coupling of lift forces and regard it as uncertainty during control [17, 18, 19]. Under such assumption our proposed methodology is applicable to any helicopter-type or any other multi-rotor-type vehicle. The paper develops a general trajectory control methodology with stability guarantees applicable to deterministic multi-body systems modeled as a tree-structure and controlled with lift and torque forces generated by propellers, and torques generated by the manipulator joint motors. While related to existing work on free-flying multi-body systems [20, 21, 22] the problem we consider poses a number of additional challenges arising from underactuation, gravity, and coupling between internal shape dynamics and overall system motion.

Standard methods for underactuated systems based on partial feedback linearization and *strong inertial coupling* [23] are not applicable, i.e. practically speaking there is no strong coupling between the uncontrolled accelerations in position space and the remaining degrees of freedom. On the other hand, it has been shown that controlling the position and the angle around the translation force input axis of a helicopter-like vehicle (modeled as single rigid body) is a choice that does not result in unstable *zero dynamics* [17]. Choosing these coordinates as outputs then renders the system differentially flat and feedback linearizable and appropriate virtual controls are found using dynamic decoupling [24] (or equivalently known as dynamic extension [25]). Such an approach is employed to control a number of quadrotor vehicles [26, 27, 28]. A number of methods have been recently proposed for controlling aerial vehicles using backstepping for better efficiency and disturbance rejection [19, 26, 29, 30, 31, 32, 33, 34]. A limitation of standard methods based on local coordinates is that the resulting controller is not globally valid and can result in singularities and unstable behavior, e.g. during inverted flight maneuvers. A method for tracking on manifolds [18] was proposed to overcome these limitations and achieve almost globally stable behavior. In addition, alternative methods for tracking on manifolds have been proposed [35, 36] that result in simpler control laws but rely on stronger assumptions such as a fixed upper bound on the maximum position or velocity error. Many of these methods have also been implemented successfully on a number of real vehicles.

The specific contributions of this work are to: 1) provide a general multi-body aerial vehicle modeling framework, 2) specify a coordinate change that enables tracking control with provable stability, 3) provide a tracking control law based on standard multi-body system models with minimum assumptions or simplifications, 4) provide an alternative coordinate-free geometric formulation which avoids singularities, 4) give guidelines for implementing tasks that require simultaneous tracking of the system center-of-mass and the manipulator tip position. The proposed method currently does not account for uncertainty and control input bounds saturation which are critical for applications on real vehicles.

We first briefly describe the basic system model in §2. A standard coordinate-based control strategy based on nonlinear backstepping control is developed in §3. A coordinate-free approach is developed in §4 which explicitly models the system as a composite free-flying rigid-body using rotation matrices rather than Euler angles. The methods are applied to a simulated hexrotor vehicle equipped with a manipulator and demonstrated by designing and tracking an agressive reaching maneuver §5.

2 System Dynamics

The free-flying vehicle is modeled as a mechanical system consisting of $n+1$ interconnected rigid bodies arranged in a tree structure. The configuration of body $\#i$ is denoted by $g_i \in SE(3)$, where

$$g_i = \begin{pmatrix} R_i & p_i \\ 0 & 1 \end{pmatrix}, \quad g_i^{-1} = \begin{pmatrix} R_i^T & -R_i^T p_i \\ 0 & 1 \end{pmatrix}.$$

where $p_i \in \mathbb{R}^3$ denotes the position of its center of mass and and $R_i \in \mathrm{SO}(3)$ denotes its orientation. Its body-fixed angular and linear velocities are denoted by $\omega_i \in \mathbb{R}^3$ and $v_i \in \mathbb{R}^3$. The pose inertia

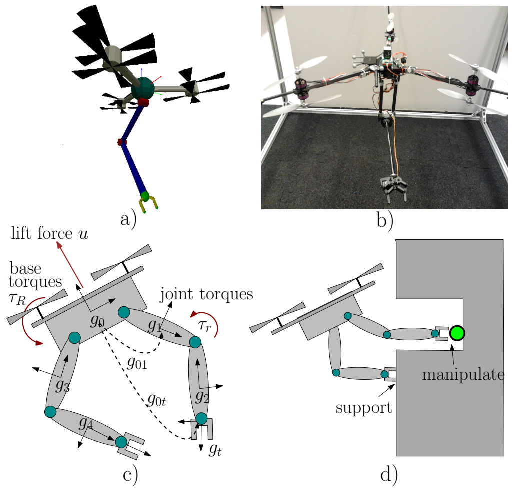

Figure 1: a) simulated model of hex-rotor vehicle, b) a prototype robot with 3-dof manipulator in development, c) diagram of a typical multi-body aerial system, d) an imaginary scenario where aerial agility could play a key role.

tensor of each body is denoted by the diagonal matrix $\mathbb{I}_i$ defined by

$$\mathbb{I}_i = \begin{pmatrix} \mathbb{J}_i & 0 \\ 0 & m_i I_3, \end{pmatrix}$$

where $\mathbb{J}_i$ is the rotational inertia tensor, m_i is its mass, and I_n denotes the n-x-n identity matrix. Each body is subject to potential energy, e.g. due to gravity, defined by the function $V : SE(3) \to \mathbb{R}$. Assume that the base body #0 is subject to forces from propellers that result in body-fixed torques $\tau_R \in \mathbb{R}^3$ and lift force $u > 0$ aligned with the constant body-fixed vertical axis $e_3 = (0, 0, 1)$.

The system has n joints described by parameters $r \in M$, where $M \subset \mathbb{R}^n$ is the *shape space*. Following standard notation [37], denote the relative transformation between the base body#0 and body#i by $g_{0i} : M \to \mathrm{SE}(3)$, i.e.

$$g_i = g_0 g_{0i}(r).$$

We assume that all joints are controlled using torque inputs denoted by $\tau_r \in \mathbb{R}^n$. Torques around the base and at the joints are combined in the torque vector

$$\tau = (\tau_R, \tau_r) \in \mathbb{R}^{3+n}.$$

Note that we assume a high-level form of the lift u and torques τ_R applied at the base body. In practice, they will be generated by actuators such as rotors or propellers with could be subject to internal dynamics as well as additional aerodynamic effects. For instance, a simplified quadrotor model is based on rotor speed inputs Ω_i, for $i = 1, ..., 4$, so that

$$\begin{aligned} u &= k_t(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2), \\ \tau_R &= \begin{bmatrix} l k_t(\Omega_4^2 - \Omega_2^2) \\ l k_t(\Omega_3^2 - \Omega_1^2) \\ k_m(\Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_4^2) \end{bmatrix}, \end{aligned} \tag{1}$$

where l, k_t, k_m are constant model parameters. We assume that there is a known mapping, such as (1) in the quadrotor case, between the high-level inputs u, τ_R and the actual physical actuator inputs.

3 Tracking control using standard coordinates

The system dynamics can be expressed in standard form (e.g. [37]) according to

$$M_s(q_s)\ddot{q}_s + C_s(q_s, \dot{q}_s)\dot{q}_s + N_s(q_s, \dot{q}_s) = B_s(q_s)u, \tag{2}$$

where $q_s \in \mathbb{R}^3 \times \mathbb{R}^3 \times M$ are the system coordinates

$$q_s = (p_0, \eta_0, r),$$

with $p_0 \in \mathbb{R}^3$ denoting the position of the base body #0 and $\eta_0 = (\alpha, \beta, \gamma) \in \mathbb{R}^3$ its three orientation angles. Standard algorithms exist for computing the matrix M_s as well as the so called *bias* terms $C_s(q_s, \dot{q}_s)\dot{q}_s + N_s(q_s, \dot{q}_s)$ by treating (p_0, η_0) as the parameters of a virtual six-dimensional joint connecting the base frame to a fixed inertial frame [38].

Let the matrices $M_{p\eta}, M_{pr}, M_{\eta\eta}, M_{\eta r}, M_{rr}$ be defined by partitioning the mass matrix (27) according to

$$M_s = \begin{bmatrix} M_{pp} & M_{p\eta} & M_{pr} \\ M_{\eta p} & M_{\eta\eta} & M_{\eta r} \\ M_{rp} & M_{r\eta} & M_{rr} \end{bmatrix}, \tag{3}$$

so that e.g. $M_{\eta p}$ pairs $\dot{\eta}$ and $\dot{p}$ in the expression for the kinetic energy $\frac{1}{2}\dot{q}^T M \dot{q}$. Since M_s is symmetric we have $M_{yx} = M_{xy}^T$. In addition, it is straightforward to verify that for systems operating in air we have $M_{pp} = mI_3$ where m is the total mass of the multi-body system defined by $m = \sum_{i=0}^n m_i$.

3.1 Center-of-mass Coordinate Change

The equations of motion (2) using standard coordinates q_s result in coupling between all degrees of freedom. For the aerial systems considered, the position p_0 is controlled by orienting the base body in order to properly direct the main lift vector $e_3 u$ in a desired direction. This becomes a non-trivial task when the manipulator is moving since the reference body is subject to additional rotational and translational forces arising from the joint motions. To deal with this coupling we transform the system by change of coordinates that diagonalize the mass matrix M_s with respect to the position. The rotation angle around the lift direction e_3 and the transformed position coordinates will thus become differentially flat outputs of the articulated multi-body system.

The first step is to combine the base and joint angles into the coordinates $q = (\eta, r) \in \mathbb{R}^3 \times M$ so that

$$q_s = (p_0, q).$$

New velocities $\dot{p}$ are then chosen according to

$$\dot{p} = \dot{p}_0 + S(q)\dot{q}, \tag{4}$$

where

$$S(q) = M_{pp}(q)^{-1}\left[M_{p\eta}(q), \quad M_{pr}(q)\right]$$

which correspond to the new position $p \in \mathbb{R}^3$ given by

$$p = \sum_{i=0}^{n} \frac{m_i}{m} p_i. \tag{5}$$

It is clear that the new position p is simply the instantaneous center of mass of the whole system.

With this transformation the dynamics can be written according to

$$m\ddot{p} = f(p, \dot{p}) + R(\eta)e_3 u, \tag{6}$$

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + N(q, \dot{q}) = B(q)\tau - S(q)^T e_3 u, \tag{7}$$

where $R(\eta)$ is the rotation matrix of the base body parametrized by the angles η and the mass matrix $M(q)$ is expressed as

$$M = [-S \quad I]^T M_s [-S \quad I] \tag{8}$$

$$= \begin{bmatrix} M_{\eta\eta} - M_{\eta p}M_{pp}^{-1}M_{p\eta} & M_{\eta r} - M_{\eta p}M_{pp}^{-1}M_{pr} \\ M_{r\eta} - M_{rp}M_{pp}^{-1}M_{p\eta} & M_{rr} - M_{rp}M_{pp}^{-1}M_{pr} \end{bmatrix}. \tag{9}$$

The term $f(p, \dot{p})$ in (6) denotes all other position forces. The simplest case is to assume that the only external force is gravity, so that $f = ma_g$ is constant.

The terms C, N, and B in (7) are computed using standard methods based on the new coordinates q and matrix M. The key point is that the position dynamics (6) now depends only on the rotation $R(\eta)$ of the base body, while the remaining rotational and joint dynamics are completely decoupled from the position p. The effect of lift forces now enters the base dynamics though through the additional term $S(q)R(\eta)e_3u$ in (7).

3.2 Trajectory Tracking Control

The tracking task is typically specified by a desired posture trajectory $q_{sd}(\cdot)$ given by

$$q_{sd}(t) = [p_{0d}(t), \eta_d(t), r_d(t)].$$

Due to underactuation it is actually not possible to independently achieve both a desired position and arbitrary desired orientation angles. In aerial tasks we are interested in tracking position while specifying only one rotational degree of freedom, i.e. the rotation around the body-fixed e_3-axis. Thus, a natural choice of rotational coordinates are XYZ Euler angles $\eta = (\alpha, \beta, \gamma)$ giving the rotation

$$R(\eta) = R_x(\alpha)R_y(\beta)R_z(\gamma), \tag{10}$$

where e.g. R_x denotes rotation around the body fixed x-axis. Note that the angles can be regarded as $yaw - pitch - roll$ angles where the yaw is performed first. This is in contrast to the more standard aircraft attitude convention where yaw is performed last.

For control design purposes, the given output trajectory is converted into an equivalent center-of-mass desired trajectory given by

$$[p_d(t), \gamma_d(t), r_d(t)]$$

which is accomplished in a straightforward manner using forward kinematics.

The center-of-mass transformation puts the system in a form suitable for building upon existing techniques (e.g. [19]) to handle the underactuated aerial base dynamics (6) and the fully actuated manipulator dynamics (7) using standard manipulator control [37] and constructing a unified and provably convergent methodology.

In order to simplify the control law design, the nominal dynamics are expressed according to

$$\dot{x} = Ax + B[f + b(\alpha, \beta, u)], \tag{11}$$

where $x \in \mathbb{R}^6$ denotes the state $x = (p, \dot{p})$, the control vector $b : \mathbb{R}^3 \to \mathbb{R}^3$ is defined by

$$b(\alpha, \beta, u) = Re_3 u = u \begin{bmatrix} \sin\beta \\ -\cos\beta\sin\alpha \\ \cos\alpha\cos\beta \end{bmatrix}$$

and the matrices A and B are given by

$$A = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix}, \qquad B = \begin{bmatrix} 0 \\ \frac{1}{m} I \end{bmatrix}. \tag{12}$$

Note that the choice of angles (10) removes dependence on the yaw angle γ from the position dynamics which enables a more straightforward control law derivation.

We next proceed by developing a backstepping approach for performing trajectory tracking control. The term $b(\alpha, \beta, u)$ is regarded as a virtual control input for the subsystem (11) with respect to the error

$$z_0(t) = x(t) - x_d(t).$$

The first step is to define the desired force b_d by

$$b_d(t, x) = m\ddot{p}_d(t) - K z_0(t) - f, \tag{13}$$

for a chosen gain matrix $K = [K_p, K_d]$ and an associated storage function

$$V_0(t, x) = \frac{1}{2} z_0^T P z_0 \geqslant 0, \tag{14}$$

where the positive definite matrix P satisfies the standard Lyapunov condition

$$P(A - BK) + (A - BK)^T P = -Q,$$

for some positive definite matrix Q. A typical choice is to employ

$$P = \begin{bmatrix} K_p & \epsilon\, m I_3 \\ \epsilon\, m I_3 & m I_3 \end{bmatrix}, \qquad Q = \begin{bmatrix} \epsilon K_p & \epsilon\, K_d \\ \epsilon\, K_d & K_v - \epsilon m I_3 \end{bmatrix},$$

where $\epsilon > 0$ is chosen sufficiently small to ensure $P, Q > 0$. The Lyapunov function then evolves according to

$$\dot{V}_0 = -\frac{1}{2} z_0^T Q z_0 + (B^T P z_0)^T (b - b_d). \tag{15}$$

At this point it is necessary to simultaneously achieve the orientation imposed by the force direction b_d as well as the remaining coordinates γ_d and r_d. We thus define the storage function

$$V_1 = V_0 + \frac{1}{2}\|z_1\|^2 \geqslant 0,$$

where the error z_1 is defined by

$$z_1(t,x,\eta,r,u) = \begin{bmatrix} b(\alpha,\beta,u) - b_d(t,x) \\ \gamma - \gamma_d(t) \\ r - r_d(t) \end{bmatrix}.$$

The evolution of V_1 is computed according to

$$\dot{V}_1 = \dot{V}_0 + z_1^T \begin{bmatrix} \dot{b} - mp_d^{(3)} + K\dot{z}_0 \\ \dot{\gamma} - \dot{\gamma}_d(t) \\ \dot{r} - \dot{r}_d(t) \end{bmatrix} \tag{16}$$

Next, define the vector $Y = (\dot{b}, \dot{\gamma}, \dot{r})$ and its *desired value* Y_d (i.e. the value which renders $\dot{V}_1$ negative definite) by

$$Y_d(t,x,\eta,r,u) = \begin{bmatrix} mp_d^{(3)} - K\dot{z}_0 - B^T P z_0 \\ \dot{\gamma}_d(t) \\ \dot{r}_d(t) \end{bmatrix} - K_1 z_1,$$

for some positive definite *diagonal* matrix K_1. After substituting Y_d in (16) we obtain

$$\dot{V}_1 = -\frac{1}{2} z_0^T Q z_0 - \frac{1}{2} z_1^T K_1 z_1 + z_1^T (Y - Y_d).$$

Next, define the storage function

$$V_2(t,x,\eta,u,\dot{\eta},\dot{u}) = V_1 + \frac{1}{2}\|z_2\|^2 \geqslant 0, \tag{17}$$

where the error z_2 is defined by

$$z_2 = Y - Y_d. \tag{18}$$

Taking its derivative we obtain

$$\dot{V}_2 = \dot{V}_1 + z_2^T \left(\dot{Y} - \dot{Y}_d\right) \tag{19}$$

The desired value of $\dot{Y}$ is defined by the vector

$$Z_d = \dot{Y}_d - z_1 - K_2 z_2 \tag{20}$$

for a chosen positive definite diagonal matrix K_2. Note that the actual expression for $\dot{Y}_d$ is obtained by substituting the dynamics of $\dot{x}$ to obtain

$$\dot{Y}_d = \begin{bmatrix} mp_d^{(4)} - K(ABg + B\dot{g} - \ddot{x}_d) - B^T P \dot{z}_0 \\ \ddot{\gamma}_d(t) \\ \ddot{r}_d(t) \end{bmatrix} - K_1 \dot{z}_1. \tag{21}$$

After substituting (21) in (19) we obtain

$$\dot{V}_2 = -\frac{1}{2} z_0^T Q z_0 - \frac{1}{2} z_1^T K_1 z_1 - \frac{1}{2} z_2^T K_2 z_2 + z_2^T (\dot{Y} - Z_d) \tag{22}$$

The relationship $\dot{Y} = Z_d$, or equivalently $(\dddot{b}, \ddot{\gamma}, \ddot{r}) = Z_d$, can now be satisfied directly based on the following relationship, obtained after straightforward algebra,

$$\dddot{b} = D \begin{bmatrix} \ddot{u} \\ \ddot{\alpha} \\ \ddot{\beta} \end{bmatrix} + 2\dot{u}C \begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \end{bmatrix} + F \begin{bmatrix} \dot{\alpha}^2 \\ \dot{\alpha}\dot{\beta} \\ \dot{\beta}^2 \end{bmatrix}, \tag{23}$$

where

$$C = \begin{bmatrix} 0 & -\sin\beta \\ -\cos\alpha\cos\beta & \sin\alpha\sin\beta \\ -\sin\alpha\cos\beta & -\cos\alpha\sin\beta \end{bmatrix}, \qquad D = \begin{bmatrix} b/u & uC \end{bmatrix},$$

$$F = \begin{bmatrix} 0 & 0 & -\cos\beta \\ \sin\alpha\cos\beta & 2\cos\alpha\sin\beta & \sin\alpha\cos\beta \\ -\cos\alpha\cos\beta & 2\sin\alpha\sin\beta & -\cos\alpha\cos\beta \end{bmatrix}.$$

It can be verified that as long as $\beta \neq \pi/2$ the matrix D is full rank. The requirement $\dot{Y} = Z_d$ is then satisfied by setting

$$\begin{bmatrix} \ddot{u} \\ \ddot{\alpha} \\ \ddot{\beta} \\ \hline \ddot{\gamma} \\ \ddot{r} \end{bmatrix} = \begin{bmatrix} D^{-1}\left\{ Z_{d(1:3)} - 2\dot{u}C \begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \end{bmatrix} - F \begin{bmatrix} \dot{\alpha}^2 \\ \dot{\alpha}\dot{\beta} \\ \dot{\beta}^2 \end{bmatrix} \right\} \\ \hline Z_{d(4:n+4)} \end{bmatrix} := \Gamma,$$

where $Z_{(i:j)}$ denotes a sub-vector with elements from index i to j, e.g. $Z_{(2:4)} = (Z_2, Z_3, Z_4)$.

In view of the dynamics (7) the desired acceleration values are achieved by setting the torques to

$$\tau = B^{-1}\left(M\Gamma_{(2:n+4)} + C\dot{q} + N + S^T b\right).$$

In summary, we have obtained conditions on the required lift vector $\dddot{b}$ which translate to conditions on $\ddot{u}, \ddot{\alpha}, \ddot{\beta}$. These conditions, combined with those on $\ddot{\gamma}, \ddot{r}$, are satisfied by setting the torques τ and lift $\ddot{u}$ so that the time-derivative of the Lyapunov function (17) becomes negative definite (22). As we will see this corresponds to asymptotic stability of the chosen output.

Proposition 1. *The control law*

$$\begin{aligned} \ddot{u} &= \Gamma_{(1)} \\ \tau &= B^{-1}\left(M\Gamma_{(2:n+4)} + C\dot{q} + N + S^T b\right). \end{aligned} \tag{24}$$

achieves asymptotic output tracking of the given bounded desired signals $p_d(t), \gamma_d(t), r_d(t)$ where $p_d(t)$ is at least four-times differentiable and has bounded derivatives while $\gamma_d(t)$ and $r_d(t)$ are at least twice-differentiable and have bounded derivatives. The following two assumptions must hold: 1.) the initial state and reference signals are such that $u(t) = e^T R(t)^T(m\ddot{p}(t) - f) > 0$ for all $t > 0$, 2.) $\beta(t) \neq \pi/2$ for all $t > 0$.

Proof. Applying the control law (24) results in

$$\dot{V}_2 = -\frac{1}{2}z_0^T Q z_0 - \frac{1}{2}z_1^T K_1 z_1 - \frac{1}{2}z_2^T K_2 z_2.$$

The first step is to establish boundedness of the extended state $(x, q, \dot{q}, u, \dot{u})$. Since $\dot{V}_2 \leqslant 0$ the error signals $z_0(t)$, $z_1(t)$ and $z_2(t)$ are uniformly bounded. Since z_0 is bounded and $x_d(t)$ is bounded we have that $x(t)$ is uniformly bounded. Thus, b_d is uniformly bounded and since γ_d, r_d, and z_1 are bounded then then b, γ, and r are uniformly bounded, and hence u is also bounded. Thus, $\dot{x}$ is bounded and since $\dot{x}_d$ is bounded we have that $\dot{z}_0$ is bounded. Therefore, Y_d is bounded. Since z_2 is bounded then Y is bounded which implies that $\dot{u}, \dot{\eta}$, and $\dot{r}$ are bounded.

Next we examine the second derivative

$$\ddot{V}_2 = -z_0^T Q \dot{z}_0 - z_1^T K_1 \dot{z}_1 - z_2^T K_2 \dot{z}_2,$$

where

$$\dot{z}_0 = \dot{x} - \dot{x}_d, \qquad \dot{z}_1 = \begin{bmatrix} \dot{b} - \dot{b}_d \\ \dot{\gamma} - \dot{\gamma}_d \\ \dot{r} - \dot{r}_d \end{bmatrix}, \qquad \dot{z}_2 = \dot{Y} - \dot{Y}_d \tag{25}$$

Since $\dot{Y}_d$ depends linearly on $\dot{u}, \dot{\eta}, \dot{r}$ then it is bounded. Furthermore, applying the control law we have that $\dot{Y} = Z_d$ which is also bounded since z_1 and z_2 are bounded. Thus, $\dot{z}_2$ is bounded and therefore $\ddot{V}$ is bounded. This implies that $\dot{V}$ is uniformly continuous function of time. Since V is lower bounded by zero, $\dot{V}$ is negative semi-definite and $\dot{V}$ is uniformly continuous, by the Lyapunov-like lemma [39] we have $\dot{V} \to 0$ and hence the tracking error dynamics are locally asymptotically stable. □

Note that we only guarantee stability when the control input u never approaches zero and when the vehicle attitude never approaches $\beta = \pi/2$. This proposition relies on the strong assumption that the initial conditions and reference signals are such that the resulting dynamics will not encounter the two singularities.

4 Coordinate-free formulation

The previous formulation is based on multi-body models which regard the base body configuration as a virtual joint motion described by six local coordinates. A more geometric approach is to consider the dynamics of free-flying system as a composite floating rigid body. This has two practical benefits: first, singularity at $\beta = \pi/2$ will be avoided and second, the resulting mass matrix will depend only on the joint angle coordinates r instead of q which reveals additional structure.

Let $q_s = (p_0, R, r)$ and $\xi_s = (v_0, \omega, \dot{r})$ denote the system configuration and velocity, respectively, where p_0 is the position of the base body, $R \equiv R_0$ is its orientation, v_0 and $\omega \equiv \omega_0$ are its body-fixed linear and angular velocities. Note that with a slight abuse of notation q_s was redefined from §3 to signify that it now contains a rotation matrix R rather than a specific choice of coordinates η.

The Lagrangian of the system is defined by

$$L_s(q_s, \xi_s) = \frac{1}{2} \xi_s^T \mathcal{M}_s(r) \xi_s - \sum_{i=0}^{n} m_i a_g^T p_i, \tag{26}$$

where the positions $p_1, ..., p_n$ are regarded as functions of q_s, and a_g denotes acceleration due to gravity. The mass matrix $\mathcal{M}_s$ is defined by (e.g. see [37, 22])

$$\mathcal{M}_s(r) = \left[\begin{array}{c|c} \mathbb{I}_0 + \sum_{i=1}^{n} A_i^T \mathbb{I}_i A_i & \sum_{i=1}^n A_i^T \mathbb{I}_i J_i \\ \hline \sum_{i=1}^n J_i^T \mathbb{I}_i A_i & \sum_{i=1}^n J_i^T \mathbb{I}_i J_i \end{array}\right] \tag{27}$$

using the adjoint notation $A_i := \mathrm{Ad}_{g_{0i}^{-1}(r)}$, and Jacobian $J_i := \sum_{j=1}^n [g_{0i}^{-1}(r)\partial_{r_j} g_{0i}(r)]^\vee$. Various efficient methods exist [38] to compute the Jacobians and the mass matrix recursively exploiting the tree structure of the multi-body system.

4.1 Center-of-mass Coordinate Change

Analogously to §3.1 the position dynamics can be factored out by diagonalizing the mass matrix with respect to the body-fixed linear velocity v_0. The first step is to combine the base and joint angles into the coordinates $q = (R, r) \in SO(3) \times M$ and $\xi = (\omega, \dot{r}) \in \mathbb{R}^3 \times \mathbb{R}^n$ so that

$$q_s = (p_0, q), \qquad \xi_s = (v_0, \xi).$$

The new position velocity v is then chosen according to

$$v = v_0 + \mathcal{S}(r)\xi, \tag{28}$$

where

$$\mathcal{S}(q) = \mathcal{M}_{pp}(q)^{-1} \left[\mathcal{M}_{p\eta}(q), \quad \mathcal{M}_{pr}(q)\right]$$

which correspond to the new center-of-mass position $p \in \mathbb{R}^3$, i.e by the relationship $\dot{p} = Rv$.

Proposition 2. *The equations of motion in coordinates* (p, v, q, ξ) *take the form:*

$$m\ddot{p} = ma_g + Re_3 u, \tag{29}$$

$$\dot{R} = R\widehat{\omega}, \tag{30}$$

$$\begin{bmatrix} \omega \\ \dot{r} \end{bmatrix} = \mathcal{M}(r)^{-1} \begin{bmatrix} \mu \\ \nu \end{bmatrix}, \tag{31}$$

$$\begin{bmatrix} \dot{\mu} \\ \dot{\nu} \end{bmatrix} = \begin{bmatrix} \mu \times \omega \\ \frac{1}{2}\xi^T \partial \mathcal{M}(r)\xi \end{bmatrix} + \tau - \mathcal{S}(r)^T e_3 u, \tag{32}$$

Proof. It can be verified that the Lagrangian

$$L(p, q, v, \xi) = \frac{1}{2} m v^T v + \frac{1}{2}\xi^T \mathcal{M}(r)\xi - m a_g^T p \tag{33}$$

satisfies the relationship $L_s(q_s, \xi_s) = L(p, q, v, \xi)$. In addition, the following relationship holds between the virtual work in $q_s = (p_0, q)$ coordinates and (p, q) coordinates:

$$\int \langle Re_3 u, \delta p_0 \rangle + \langle \tau_R, \eta \rangle + \langle \tau_r, \delta r \rangle = \int \langle Re_3 u, \delta p \rangle + \left\langle \tau - \mathcal{S}(r)^T e_3 u, (\eta, \delta r) \right\rangle, \tag{34}$$

with $\eta = (R^T \delta R)^\vee$ and where the variational relationship

$$R^T \delta p_0 = R^T \delta p + \mathcal{S}(r) \begin{bmatrix} \eta \\ \delta r \end{bmatrix}$$

was employed. The variational principle

$$\delta \int L(p, q, v, \xi) dt + \langle Re_3 u, \delta x \rangle + \big\langle \tau - \mathcal{S}(r)^T e_3 u, (\eta, \delta r) \big\rangle = 0, \tag{35}$$

then holds true and determines the system dynamics. The position dynamics is decoupled and results in the standard form (29). The relation (31) is the Legendre transform from momenta $\mu = \partial_\omega L$ and $\nu = \partial_{\dot r} L$ to velocities. The momenta evolution (32) is then derived by taking variations $(\delta R, \delta r)$ under the standard (e.g. [40]) rigid-body constraint $\delta\omega = \dot\eta + \omega \times \eta$. □

Note that the main body and joint dynamics (32) were obtained in a form which leaves the torques τ decoupled. The rotational coupling is instead at the momentum level through the Legendre transform (31).

Before deriving the control law in §4.3 it is necessary to introduce an approach for defining the error between two given rotation matrices intrinsically.

4.2 General Rotation Error

Our approach for treating the error in rotation without resorting to coordinates such as Euler angles follows the development in [18] and more generally [41]. For greater generality, we provide an abstract mapping with alternative choices for encoding this error. In particular, the Cayley map and its higher-order versions provide a simple approach that leads to even further expansion of the region of asymptotic stability.

More specifically, rrrors in rotation are encoded using a *retraction map.* The following definitions will enable us to obtain a tracking control law which avoids the singularity at $\beta = \pi/2$.

Definition 4.1. The retraction map $\vartheta : \mathbb{R}^3 \to \mathrm{SO}(3)$ is a smooth map around the origin such that $\vartheta(0) = I_3$.

The notion of *retraction* is used to approximate the difference between two given rotation matrices R_a and R_b using a single vector, say $\Delta \in \mathbb{R}^3$. One can visualize the manifold $SO(3)$ as a curved surface with R_a represented as a point at which the tangent vector Δ is attached. The vector can be "retracted" or "bent" onto to the surface until its tip reaches the surface at another point. The vector whose tip touches R_b is taken as the difference between R_a and R_b. While there can be arbitrary retractions for our purposes we are interested in maps which approximate the exponential map.

Next we define a matrix-value map C_ϑ which abstracts away the nonlinear terms in the retraction map. In the following definitions the "hat" notation $\widehat{\cdot} : \mathbb{R}^3 \to \mathfrak{so}(3)$ defined by

$$\widehat{\omega} = \begin{bmatrix} 0 & -w^3 & w^3 \\ w^3 & 0 & -w^1 \\ -w^2 & w^1 & 0 \end{bmatrix}, \tag{36}$$

and its inverse $\check{\cdot} : \to \mathfrak{so}(3) \to \mathbb{R}^3$ are employed.

Definition 4.2. The map $C_\vartheta : \mathbb{R}^3 \to L(\mathbb{R}^3, \mathbb{R}^3)$ is such that, for a given $R \in \mathrm{SO}(3)$, the following holds

$$R = I + \widehat{\rho} C_\vartheta(\rho),$$

where $\rho = \vartheta^{-1}(R) \in \mathbb{R}^3$.

There are three retraction map choices that we employ:

1. The *exponential map* $\vartheta = \exp$ and its inverse $\vartheta^{-1} = \log$ are defined by:

$$\exp(\rho) = \begin{cases} I_3, & \rho = 0 \\ I_3 + \frac{\sin\|\rho\|}{\|\rho\|}\widehat{\rho} + \frac{1-\cos\|\rho\|}{\|\rho\|^2}\widehat{\rho}^2, & \rho \neq 0 \end{cases} \tag{37}$$

$$\log(R) = \begin{cases} 0, & \theta = 0 \\ \frac{\theta}{2\sin\theta}\left(R - R^T\right)^{\vee}, & \theta \neq 0 \end{cases} \tag{38}$$

$$C_{\exp}(\rho) = \begin{cases} I_3, & \rho = 0 \\ \frac{\sin\|\rho\|}{\|\rho\|} I_3 + \frac{1-\cos\|\rho\|}{\|\rho\|^2}\widehat{\rho}, & \rho \neq 0 \end{cases}, \tag{39}$$

where $\theta = \arccos\frac{\mathrm{trace}(R)-1}{2}$.

2. The *Cayley map* $\vartheta = \mathrm{cay}$ and its inverse $\vartheta^{-1} = \mathrm{cay}^{-1}$ are defined by:

$$\mathrm{cay}(\rho) = \mathbf{I}_3 + \frac{4}{4+\|\rho\|^2}\left(\widehat{\rho} + \frac{\widehat{\rho}^2}{2}\right). \tag{40}$$

$$\mathrm{cay}^{-1}(R) = -2\left[(I_3 + R)^{-1}(I_3 - R)\right]^{\vee} \tag{41}$$

$$C_{\mathrm{cay}}(\rho) = \frac{4}{4+\|\rho\|^2}\left(I_3 + \frac{\widehat{\rho}}{2}\right). \tag{42}$$

3. Higher-order Rodriguez's parameters $\vartheta = \mathrm{rod}_2$ and $\vartheta^{-1} = \mathrm{rod}_2^{-1}$ (described in e.g. [42]).

The exponential and Cayley maps can represent rotation errors up to π radians. This range can be extended to 2π using the modified Rodriguez's parameters and to even larger ranges using higher-order Cayley mappings. In our implementation we employ $\vartheta = \mathrm{cay}$ since it has the simplest form, without any trigonometric functions or singularities at the origin.

4.3 Trajectory Tracking Control

Similarly to §3.2 assume that the tracking task is specified in terms of desired center-of-mass posture, i.e. by

$$[p_d(t), R_d(t), r_d(t)]$$

where the rotation matrix R_d must satisfy the conditions

$$R_d e_3 = b_d(x,t)/u_d, \qquad u_d = \|b_d\|, \tag{43}$$

where recall that (13)

$$b_d(t,x) = m\ddot{p}_d(t) - Kz_0(t) - f.$$

This condition leaves one additional degree of freedom in R_d that can be specified by the user. In practice, b_d/u_d serves as the third column of the matrix R_d while the other two columns can be freely chosen (subject to the standard unit orthogonality constraints), e.g. to obtain a matrix R_d that is closest to a given reference R_{ref} (see [18] for an example).

We start with the Lyapunov function V_0 already defined in (14) which evolves according to

$$\dot{V}_0 = -\frac{1}{2}z_0^T Q z_0 + (B^T P z_0)^T (Re_3 u - b_d). \tag{44}$$

Using Definition 4.2 the following relationship holds

$$b_d = R(R_d^T R)^T e_3 u_d = R\left[I - \widehat{\rho} C_\vartheta(-\rho)\right] e u_d,$$

where $\rho = \vartheta^{-1}(R_d^T R)$, which is substituted in (44) to obtain

$$\dot{V}_0 = -\frac{1}{2} z_0^T Q z_0 + (u - u_d) e^T R^T B^T P z_0 + \rho^T \left[(R^T B^T P z_0) \times (C_\vartheta(-\rho) e u_d)\right]. \tag{45}$$

Since the orientation error ρ is now part of the backstepping stage, it is time to also introduce the remaining coordinates, i.e. the joint angles r. We thus define the storage function

$$V_1 = V_0 + \frac{1}{2}\|z_1\|^2 \geqslant 0,$$

where the error z_1 is defined by

$$z_1 = \begin{bmatrix} u - u_d \\ \rho \\ r - r_d \end{bmatrix}$$

The evolution of V_1 is computed according to

$$\dot{V}_1 = \dot{V}_0 + z_1^T \begin{bmatrix} \dot{u} - \dot{u}_d(t) \\ \omega - R^T R_d \omega_d(t) \\ \dot{r} - \dot{r}_d(t) \end{bmatrix}, \tag{46}$$

where $\widehat{\omega}_d = R_d^T \dot{R}_d$. Next, define the vector $\mathcal{Y} = (\dot{u}, \omega, \dot{r})$ and its desired value by

$$\mathcal{Y}_d(t, x, R, r, u) = \begin{bmatrix} \dot{u}_d(t) - e^T R^T B^T P z_0 \\ R^T R_d \omega_d(t) - (R^T B^T P z_0) \times \left[C_\vartheta(-\rho) e u_d\right] \\ \dot{r}_d(t) \end{bmatrix} - K_1 z_1,$$

for some positive definite diagonal matrix K_1. After substituting $\mathcal{Y}_d$ in (46) we obtain

$$\dot{V}_1 = -\frac{1}{2} z_0^T Q z_0 - \frac{1}{2} z_1^T K_1 z_1 + z_1^T(\mathcal{Y} - \mathcal{Y}_d).$$

Next, define the storage function

$$V_2 = V_1 + \frac{1}{2}\|z_2\|^2 \geqslant 0, \tag{47}$$

where the error z_2 is defined by

$$z_2 = \mathcal{Y} - \mathcal{Y}_d. \tag{48}$$

Taking its derivative we obtain

$$\dot{V}_2 = \dot{V}_1 + z_2^T \left(\dot{\mathcal{Y}} - \dot{\mathcal{Y}}_d\right). \tag{49}$$

The desired value of $\dot{\mathcal{Y}}$ is defined by the vector

$$\mathcal{Z}_d = \dot{\mathcal{Y}}_d - z_1 - K_2 z_2 \tag{50}$$

for a chosen positive definite diagonal matrix K_2. After substituting (50) in (49) we obtain

$$\dot{V}_2 = -\frac{1}{2}z_0^T Q z_0 - \frac{1}{2}z_1^T K_1 z_1 - \frac{1}{2}z_2^T K_2 z_2 + z_2^T(\dot{\mathcal{Y}} - \mathcal{Z}_d) \tag{51}$$

The relationship $\dot{\mathcal{Y}} = \mathcal{Z}_d$, or equivalently $(\ddot{u}, \dot{\omega}, \ddot{r}) = \mathcal{Z}_d$, can now be satisfied directly using the dynamics (32). This is accomplished by substituting the relationship

$$\begin{bmatrix} \dot{\mu} \\ \dot{\nu} \end{bmatrix} = \dot{\mathcal{M}}(r)\begin{bmatrix} \omega \\ \dot{r} \end{bmatrix} + \mathcal{M}(r)\begin{bmatrix} \dot{\omega} \\ \ddot{r} \end{bmatrix}$$

into the dynamics (32) and setting the torque inputs to

$$\tau = \mathcal{M}(r)\mathcal{Z}_{d(2:n+4)} + \dot{\mathcal{M}}(r)\xi - \begin{bmatrix} \mu \times \omega \\ \frac{1}{2}\xi^T \partial\mathcal{M}(r)\xi \end{bmatrix} + \mathcal{S}(r)^T e_3 u, \tag{52}$$

The complete control law is summarized as follows.

Proposition 3. *The control inputs*

$$\begin{aligned} \ddot{u} &= \mathcal{Z}_{d(1)} \\ \tau &= \mathcal{M}(r)\mathcal{Z}_{d(2:n+4)} + \dot{\mathcal{M}}(r)\xi - \begin{bmatrix} \mu \times \omega \\ \frac{1}{2}\xi^T \partial\mathcal{M}(r)\xi \end{bmatrix} + \mathcal{S}(r)^T e_3 u, \end{aligned} \tag{53}$$

achieve asymptotic output tracking of given bounded desired signals $p_d(t), R_d(t), r_d(t)$ where $p_d(t)$ is at least four-times differentiable and has bounded derivatives while $R_d(t)$ and $r_d(t)$ are at least twice-differentiable and have bounded derivatives. In addition, the following two assumptions must hold: 1.) the initial state and reference signals are such that $u(t) = e_3^T R(t)^T (m\ddot{p}(t) - f) > 0$; 2.) the control law is not applied when the angle of the rotation $R_d^T R$ is exactly π.

Proof. The proof is very similar to the coordinate-based development in Proposition 1. The key point is the Lyapunov function V_2 is positive definite while the proposed control law renders its time-derivative (51) negative definite. □

Note that the two assumptions are natural and do not impose practical limitations: 1.) when $u = 0$ the vehicle looses controllability and, as expected, the vehicle enters free-fall; 2.) the rotation $R_d^T R$ has an angle exactly π almost never since the set $\{\pi\}$ is obviously measure-zero. Since the state is determined by an imperfect sensor and always has small variations, the ill-posedness of the retraction maps at π is not an issue in practice.

5 Application: hexrotor with a simple manipulator

The hexrotor shown in Figure 1 has three pairs of propellers fixed onto three spokes at 120 degrees. A two-link manipulator with a low-cost gripper is suspended from the vehicle and can extend forward between the two forward-facing spokes. Such an arrangement enables the manipulator tip to extend beyond the vehicle perimeter which enables interesting reaching maneuvers.

Ignoring the gripper motor, the manipulator has two degrees of freedom, i.e. $r = (r_1, r_2)$. The forward kinematics are given by

$$g_{01}(r) = \begin{pmatrix} c_1 & 0 & s_1 & -\frac{l_1}{2}s_1 \\ 0 & 1 & 0 & 0 \\ -s_1 & 0 & c_1 & -\frac{l_1}{2}c_1 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \tag{54}$$

$$g_{02}(r) = \begin{pmatrix} c_{12} & 0 & s_{12} & -\frac{l_2}{2}s_{12} - l_1 s_1 \\ 0 & 1 & 0 & 0 \\ -s_{12} & 0 & c_{12} & -\frac{l_2}{2}c_{12} - l_1 c_1 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \tag{55}$$

$$g_{0t}(r) = \begin{pmatrix} c_{12} & 0 & s_{12} & -l_2 s_{12} - l_1 s_1 \\ 0 & 1 & 0 & 0 \\ -s_{12} & 0 & c_{12} & -l_2 c_{12} - l_1 c_1 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \tag{56}$$

using the shorthand notation $c_i := \cos r_i$, $s_i := \sin r_i$ for $i = 1, 2$ and $c_{12} := \cos(r_1 + r_2)$, $s_{12} := \sin(r_1 + r_2)$.

We assume that a desired trajectory is specified using a desired center of mass $x_d(t) \in \mathbb{R}^3$ and manipulator tip position $y_d(t) \in \mathbb{R}^3$. Let $I_x : SE(3) \to \mathbb{R}^3$ and $I_R : SE(3) \to SO(3)$ extract the position and orientation of a given pose. The required orientation R_d and joint angles r_d to track y_d are chosen to satisfy

$$y_d = I_x\left(g_0(x_d, R_d) g_{0t}(r_d)\right). \tag{57}$$

The rotation R_d is chosen during closed-loop tracking so that $R_d e = b_d$. This leaves an extra degree of freedom in R_d, i.e. rotating the frame around the b_d axis. A desired y_d can be exactly achieved by setting

$$c_3 = \frac{b_d}{\|b_d\|} \tag{58}$$

$$c_2 = \frac{c_3 \times (y_d - x_d)}{\|c_3 \times (y_d - x_d)\|} \tag{59}$$

$$R_d = [\pm c_2 \times c_3 \mid \pm c_2 \mid c_3] \tag{60}$$

The required angles r_1, and r_2 are found through the relationship

$$I_x\left(g_{0t}(r)\right) = R_d^T (y_d - x_d), \tag{61}$$

where only the first and third elements of the vectors on both sides are non-zero. The relation (61) is solved in closed form using standard inverse kinematics techniques. There are a total of four solutions (R_d, r_d) for any given (x_d, y_d) since (58) gives two choices and (61) results in a quadratic equation with two roots.

Figures 2 and 3 show the simulated controller behavior during an aggressive reaching maneuver. The vehicle is required to track a path $(x_d(t), y_d(t))$ that extends the manipulator outside of the vehicle propeller range in order to reach a desired final point.

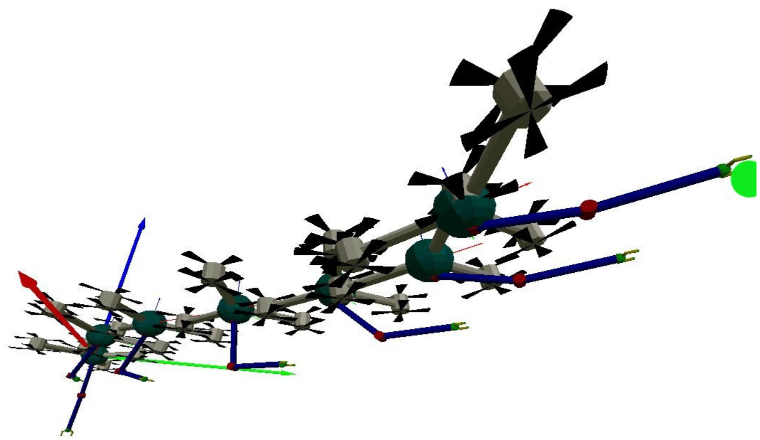

Figure 2: Several frames along the simulated hexrotor trajectory reaching a desired point in workspace.

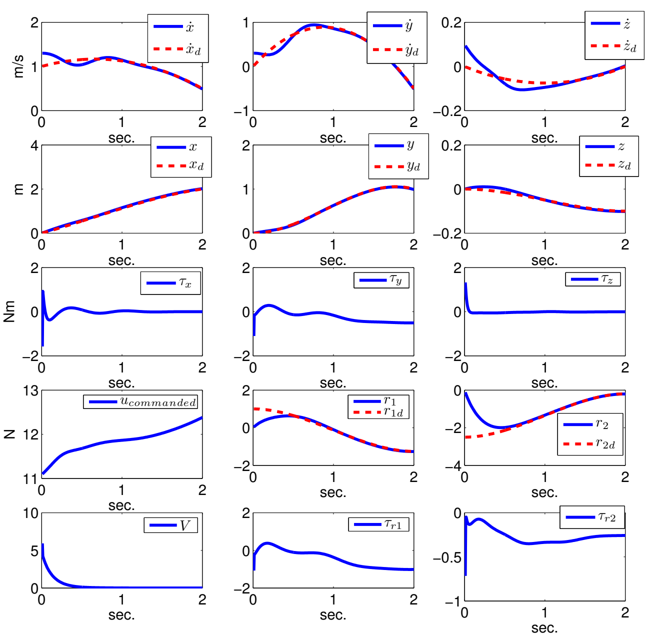

Figure 3: History of the states, control inputs, and Lyapunov function V of the scenario shown in Fig. 2

6 Conclusion

This paper studies trajectory tracking of articulated aerial systems. The developed controller employed existing results in free-flying multi-body system modeling, backstepping control of underactuated systems, and control on manifolds. The key contribution is to transform the system into a suitable form that enables dynamic extension and energy-based control, avoiding the use of rotational coordinates, proving stability for the coupled closed-loop system. Key issues that need to addressed is the sensitivity of the resulting controller to unmodeled disturbances in view of their effect on the higher-order derivatives present in the control law. Furthermore, bounds on the actuators need to be formally considered since aerial manipulation tasks would require inputs close to the vehicle operational envelope. Resolving these two issues is critical for rendering the proposed methods useful for practical applications.

References

[1] P. Pounds, D. Bersak, and A. Dollar, "The yale aerial manipulator: Grasping in flight," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, may 2011, pp. 2974 –2975.

[2] P. E. I. Pounds, D. R. Bersak, and A. M. Dollar, "Stability of small-scale uav helicopters and quadrotors with added payload mass under pid control," *Auton. Robots*, vol. 33, no. 1-2, pp. 129–142, 2012.

[3] Q. J. Lindsey, D. Mellinger, and V. Kumar, "Construction of cubic structures with quadrotor teams," *Robotics: Science and Systems*, June 2011.

[4] M. Hehn and R. D'Andrea, "A flying inverted pendulum," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, may 2011, pp. 763 –770.

[5] C. M. Korpela, T. W. Danko, and P. Y. Oh, "Mm-uav: Mobile manipulating unmanned aerial vehicle," *J. Intell. Robotics Syst.*, vol. 65, no. 1-4, pp. 93–101, Jan. 2012. [Online]. Available: http://dx.doi.org/10.1007/s10846-011-9591-3

[6] C. Korpela, M. Orsag, T. Danko, B. Kobe, C. McNeil, R. Pisch, and P. Oh, "Flight stability in aerial redundant manipulators," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, may 2012, pp. 3529 –3530.

[7] C. Korpela, T. Danko, and P. Oh, "Designing a system for mobile manipulation from an unmanned aerial vehicle," in *Technologies for Practical Robot Applications (TePRA), 2011 IEEE Conference on*, april 2011, pp. 109 –114.

[8] Airobots, "http://airobots.ing.unibo.it."

[9] ARCAS, "http://www.arcas-project.eu."

[10] D. Fusata, G. Guglieri, and R. Celi, "Flight dynamics of an articulated rotor helicopter with an external slung load," *Journal of the American Helicopter Society*, vol. 46, no. 1, pp. 3–14, 2001.

[11] I. Maza, K. Kondak, M. Bernard, and A. Ollero, "Multi-uav cooperation and control for load transportation and deployment," *Journal of Intelligent and Robotic Systems*, vol. 57, no. 1-4, pp. 417–449, 2010. [Online]. Available: http://dx.doi.org/10.1007/s10846-009-9352-8

[12] J. Thomas, J. Polin, K. Sreenath, and V. Kumar, "Avian-inspired grasping for quadrotor micro uavs," in *ASME International Design Engineering Technical Conference (IDETC)*, to appear, 2013.

[13] A. Torre, D. Mengoli, R. Naldi, F. Forte, A. Macchelli, and L. Marconi, "A prototype of aerial manipulator," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, 2012, pp. 2653–2654.

[14] A. Mersha, S. Stramigioli, and R. Carloni, "Bilateral teleoperation of underactuated unmanned aerial vehicles: The virtual slave concept," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, 2012, pp. 4614–4620.

[15] M. Orsag, C. Korpela, M. Pekala, and P. Oh, "Stability control in aerial manipulation," in *American Control Conference (ACC), 2013*, 2013, pp. 5581–5586.

[16] A. Jimenez-Cano, J. Martin, G. Heredia, R. Cano, and A. Ollero, "Control of an aerial robot with multi-link arm for assembly tasks," in *International Conference on Robotics and Automation*, 2013.

[17] T. Koo and S. Sastry, "Output tracking control design of a helicopter model based on approximate linearization," in *Decision and Control, 1998. Proceedings of the 37th IEEE Conference on*, vol. 4, dec 1998, pp. 3635 –3640 vol.4.

[18] E. Frazzoli, M. Dahleh, and E. Feron, "Trajectory tracking control design for autonomous helicopters using a backstepping algorithm," in *American Control Conference, 2000. Proceedings of the 2000*, vol. 6, 2000, pp. 4102–4107.

[19] R. Mahony and T. Hamel, "Robust trajectory tracking for a scale model autonomous helicopter," *International Journal of Robust and Nonlinear Control*, vol. 14, no. 12, pp. 1035–1059, 2004. [Online]. Available: http://dx.doi.org/10.1002/rnc.931

[20] S. Dubowsky and E. Papadopoulos, "The kinematics, dynamics, and control of free-flying and free-floating space robotic systems," *Robotics and Automation, IEEE Transactions on*, vol. 9, no. 5, pp. 531 –543, oct 1993.

[21] R. Mukherjee and D. Chen, "Control of free-flying underactuated space manipulators to equilibrium manifolds," *Robotics and Automation, IEEE Transactions on*, vol. 9, no. 5, pp. 561 –570, oct 1993.

[22] A. Jain, *Robot and Multibody Dynamics: Analysis and Algorithms.* Springer, 2011.

[23] M. W. Spong, "Underactuated mechanical systems," in *Control Problems in Robotics and Automation.* Springer-Verlag, 1998.

[24] J. Descusse and C. H. Moog, "Dynamic decoupling for right-invertible nonlinear systems," *Syst. Control Lett.*, vol. 8, no. 4, pp. 345–349, Mar. 1987. [Online]. Available: http://dx.doi.org/10.1016/0167-6911(87)90101-0

[25] J. Hauser, S. Sastry, and G. Meyer, “Nonlinear control design for slightly non-minimum phase systems: Application to v/stol aircraft,” *Automatica*, vol. 28, no. 4, pp. 665 – 679, 1992. [Online]. Available: http://www.sciencedirect.com/science/article/pii/000510989290029F

[26] E. Altug, J. P. Ostrowski, and R. E. Mahony, “Control of a quadrotor helicopter using visual feedback,” in *International Conference on Robotics and Automation*, 2002, pp. 72–77.

[27] S. Al-Hiddabi, “Quadrotor control using feedback linearization with dynamic extension,” in *Mechatronics and its Applications, 2009. ISMA ’09. 6th International Symposium on*, march 2009, pp. 1 –3.

[28] D. Lee, H. Jin Kim, and S. Sastry, “Feedback linearization vs. adaptive sliding mode control for a quadrotor helicopter,” *International Journal of Control, Automation and Systems*, vol. 7, pp. 419–428, 2009, 10.1007/s12555-009-0311-8. [Online]. Available: http://dx.doi.org/10.1007/s12555-009-0311-8

[29] S. Bouabdallah and R. Siegwart, “Backstepping and sliding-mode techniques applied to an indoor micro quadrotor,” in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, april 2005, pp. 2247 – 2252.

[30] T. Madani and A. Benallegue, “Control of a quadrotor mini-helicopter via full state backstepping technique,” in *Decision and Control, 2006 45th IEEE Conference on*, dec. 2006, pp. 1515 –1520.

[31] P. Adigbli, C. Grand, J. baptiste Mouret, and S. Doncieux, “Nonlinear attitude and position control of a micro quadrotor using sliding mode and backstepping techniques,” in *European Micro Air Vehicle Conference and Flight Competition (EMAV2007)*, 2007.

[32] G. Raffo, M. Ortega, and F. Rubio, “Backstepping/nonlinear h-infty control for path tracking of a quadrotor unmanned aerial vehicle,” in *American Control Conference, 2008*, june 2008, pp. 3356 –3361.

[33] J. Colorado, A. Barrientos, A. Martinez, B. Lafaverges, and J. Valente, “Mini-quadrotor attitude control based on hybrid backstepping amp; frenet-serret theory,” in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, may 2010, pp. 1617 –1622.

[34] I.-H. Choi and H.-C. Bang, “Adaptive command filtered backstepping tracking controller design for quadrotor unmanned aerial vehicle,” *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 226, no. 5, pp. 483–497, 2012. [Online]. Available: http://pig.sagepub.com/content/226/5/483.abstract

[35] R. Olfati-Saber, “Nonlinear control of underactuated mechanical systems with application to robotics and aerospace vehicles,” Ph.D. dissertation, Massachusetts Institute of Technology, 2001, aAI0803036.

[36] T. Lee, M. Leok, and N. McClamroch, “Geometric tracking control of a quadrotor uav on se(3),” in *Decision and Control (CDC), 2010 49th IEEE Conference on*, dec. 2010, pp. 5420 –5425.

[37] R. M. Murray, Z. Li, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation.* CRC, 1994.

[38] R. Featherstone, *Rigid Body Dynamics Algorithms.* Springer, 2008.

[39] J. J. E. Slotine and W. A. Li, *Applied Nonlinear Control.* Prentice Hall, 1991.

[40] J. E. Marsden and T. S. Ratiu, *Introduction to Mechanics and Symmetry.* Springer, 1999.

[41] F. Bullo and A. Lewis, *Geometric Control of Mechanical Systems.* Springer, 2004.

[42] P. Tsiotras, "Higher order cayley transforms with applications to attitude representations," *Journal of Guidance, Control, and Dynamics*, vol. 20, 1997.