Received February 13, 2013, accepted April 16, 2013, published May 10, 2013.

Digital Object Identifier 10.1109/ACCESS.2013.2260794

Robust Optimal Control of Quadrotor UAVs

AYKUT C. SATICI[1], HASAN POONAWALA[1], AND MARK W. SPONG[1,2]
[1]University of Texas at Dallas, Richardson, TX 75080, USA
[2]Erik Jonsson School of Engineering and Computer Science, University of Texas at Dallas, Richardson, TX 75080, USA

Corresponding author: A. C. Satici (acsatici@utdallas.edu)

This work was supported in part by the National Science Foundation under Grants ECCS 07-25433 and CMMI-0856368, the University of Texas STARs Program, and DGIST R&D Program of the Ministry of Education, Science and Technology of Korea under Grant 12-BD-0101.

ABSTRACT This paper provides the design and implementation of an L_1-optimal control of a quadrotor unmanned aerial vehicle (UAV). The quadrotor UAV is an underactuated rigid body with four propellers that generate forces along the rotor axes. These four forces are used to achieve asymptotic tracking of four outputs, namely the position of the center of mass of the UAV and the heading. With perfect knowledge of plant parameters and no measurement noise, the magnitudes of the errors are shown to exponentially converge to zero. In the case of parametric uncertainty and measurement noise, the controller yields an exponential decrease of the magnitude of the errors in an L_1-optimal sense. In other words, the controller is designed so that it minimizes the L_∞-gain of the plant with respect to disturbances. The performance of the controller is evaluated in experiments and compared with that of a related robust nonlinear controller in the literature. The experimental data shows that the proposed controller rejects persistent disturbances, which is quantified by a very small magnitude of the mean error.

INDEX TERMS Robust control, optimal control, quadrotor, feedback linearization, unmanned aerial vehicle.

I. INTRODUCTION

Quadrotors have become increasingly popular in recent years due to their ability to hover and maneuver in tight spaces. Stabilizing the quadrotor in its configuration space $\mathbf{SE}(3)$ is an interesting and challenging control problem. The challenge arises due to the availability of four control inputs to stabilize the system that lives in a six dimensional configuration space and possesses coupled, nonlinear and unstable open-loop dynamics.

The goal of most quadrotor control schemes is either to stabilize the center of mass (CoM) at a desired location or for the CoM to track a desired trajectory. Practical implementation of these nonlinear control schemes for robotic systems requires one to take into consideration the issues of robustness to parameter uncertainty, external disturbances, and sensor noise. It is well-known that the equations of motion of a quadrotor can be partially linearized [1] and decoupled by appropriate nonlinear feedback. known as the method of partial feedback linearization. It is very useful from a control viewpoint, since a part of the complex highly coupled nonlinear dynamics of the quadrotor are replaced by a simple set of second-order linear differential equations. The partial feedback linearization results in a controller that has the same order of complexity as the model of the system being controlled. Good control performance requires a high sampling rate at which the desired control can be computed. A desire to maximize quadrotor payloads and performance results in a constraint on computational power available for autonomous control. It is highly advantageous therefore to consider nonlinear feedback based upon simplified models and to quantify the performance of such controllers.

In this paper the partial feedback linearization method is applied to the nominal model. A linear compensator for the resulting partial linear model is then designed using the method of stable factorizations. The advantages of this control scheme can be summarized as follows.

- Any linear compensator that stabilizes a linear model can be specified using the method of stable factorization. As a consequence, one is able to conclude whether a compensator designed based on the linear model is able to stabilize the ''true'' nonlinear system as well, provided the deviation between the modeled and the actual plant satisfies certain bounds.
- If the deviation of the model from the plant does not exceed the maximum allowable, it is possible to achieve corresponding reductions in the effects on the closed-loop performance of the parameter uncertainty, measurement noise, available computational power, etc.

The simplest control method involves linearization of the quadrotor dynamics around the hover state and then using various linear control methods to stabilize the system, such as standard SISO tools in [2], or PID and LQR methods in [3]. In [4] the authors use a sliding mode control method to stabilize the linearized dynamics. The work includes a rate limited PID for control of the height. Controllers designed using such linear methods may be used to track slow motions. However, nonlinear control methods are needed to achieve higher performance and to execute more aggressive motions.

The quadrotor dynamics has a cascade structure. The four motor inputs are mapped to three independent torques acting on the quadrotor frame and one thrust force. The torque inputs directly control the orientation dynamics. The orientation then affects the thrust vector direction which drives the translation dynamics. This cascade structure can be elegantly exploited by using backstepping control techniques [5], [6] or inner-loop/outer-loop approaches [2], [3], [7]. The work in [5] modifies the backstepping approach by using sliding mode control for one of the cascaded variables, which introduces some robustness to disturbances.

While the position of the CoM is almost always expressed in cartesian coordinates, selection of a representation for the orientation is an important step. All the papers mentioned so far use an Euler-angle parametrization, which suffers from singularities. To address this, the work in [8] represents the orientation using the rotation matrix itself. This leads to an orientation tracking controller that is almost globally attractive. The desired orientation and thrust magnitude are designed such that the translation dynamics are linear. The paper gives a good summary of the various representations. The same method was implemented in the work of [9] in order to achieve aggressive motions of a quadrotor.

Feedback linearization [10] is often employed in order to facilitate the use of linear control methods over a large region of the state space. The choice for the outputs or states to linearize may differ. In [11] the position and heading are taken as the outputs to be linearized whereas in [12] the height and three Euler angles are linearized. The control inputs to the linearized dynamics are selected as PD control.

Most nonlinear control methods are highly dependent on the exact knowledge of the system parameters in order to achieve asymptotic tracking. Various researchers have attempted to address this issue using robust control [13]–[15] or adaptive control [16] and even a combination of the two [17]. The performance recovery generally trades off asymptotic stability with uniform ultimate boundedness.

The control method proposed in this paper also uses the idea of controlling the thrust direction to achieve position control. However, the control of orientation is prone to persistent disturbances stemming from uncertain inertial parameters, misalignment of blade axes, discrepancy between the measured quadrotor frame and the actual frame. The rejection of such persistent disturbances is the primary motivation for our control design. Methods addressing this issue were first developed in [18]–[21].

The method in [21] elegantly determines the optimal linear controller that stabilizes a given plant while rejecting persistent disturbances. The method relies on the stable factorization approach [22] which is applicable only to linear plants. This precludes the methods used in [8] to stabilize the quadrotor since the plant remains nonlinear. The key contribution of this work is the parametrization of the orientation and subsequent partial feedback linearization of the dynamics which paves the way for applying the L_1 optimal robust linear compensator design for nonlinear robotic systems similar to [20]. Furthermore, we have implemented this controller on a quadrotor in LARS[1] at the University of Texas at Dallas, and we have compared the performance of our controller with that in [15]. The design of the robust controller in [15] is not best suited for rejecting persistent disturbances, whereas our controller is. The results show that our controller can handle the effect of modeling errors such as blade misalignment and shifted center of mass location better than that in [15].

II. KINEMATICS AND DYNAMICS OF THE QUADROTOR

In the following few paragraphs, we derive the kinematic and dynamic differential equations of motion of the quadrotor, culminating in equations (5). The reader who is familiar with the development may skip to these equations immediately without losing any generality. The quadrotor is a simple mechanical system with configuration space $Q = SE(3) = \mathbb{R}^3 \ltimes SO(3)$, the semidirect product of the three-dimensional Euclidean space and the special orthogonal group. The first factor in the semidirect product represents the position of the center of mass of the quadrotor in the world frame. We select the center of mass as the origin of the body-fixed frame $\Sigma : \{o, x, y, z\}$ and denote the vector from the origin of the inertial frame $\Sigma_0 : \{o_0, x_0, y_0, z_0\}$ to o by $p \in \mathbb{R}^3$. Ideally, the vectors $\{x, y\}$ span the plane parallel to the rotor axes and z is orthogonal to this plane. We denote by $R \in SO(3)$, the orientation of the quadrotor body-fixed frame with respect to the world frame. The relevant axes and forces are illustrated on the Quanser Qball [23] in Figure 1. These four forces, generated by four rotors and propellers, induce a total thrust f, and a net torque $\tau = \{\tau_x, \tau_y, \tau_z\}$ around the three body axes $\{x, y, z\}$. In addition to these definitions, we will denote the linear and angular velocities of the quadrotor by the pair $(v, \Omega) \in \mathbb{R}^3 \ltimes \mathfrak{so}(3)$. Under the vector space isomorphism $\mathfrak{so}(3) \simeq \mathbb{R}^3$, the skew-symmetric matrix Ω will be identified with the 3-vector ω. Finally, we denote the (identical) distance from the center of mass of the quadrotor to the rotor axes by L, the total mass by m, and the inertia tensor represented in the body-fixed frame by $J = \mathrm{diag}(J_p, J_r, J_y)$.

Since the configuration space of the quadrotor is $Q = SE(3)$, it can be modeled as an underactuated rigid body. The velocity of the quadrotor is an element of the tangent space $T_{(p,R)}\mathbb{R}^3 \ltimes SO(3)$. As a Lie group, the tangent space of $\mathbb{R}^3 \ltimes SO(3)$ can be identified with the tangent space at the identity by the left or right translation to the identity. Since the

[1](L)aboratory for (A)utonomous (R)obotics and (S)ystems

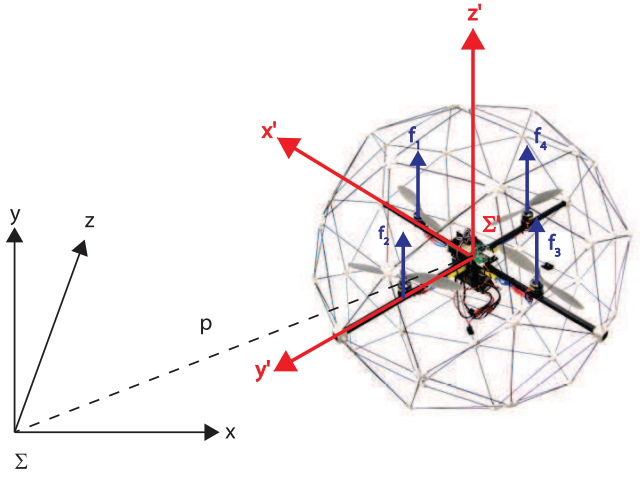


FIGURE 1. Quanser Qball model.

rotational equations of motion are easier when expressed in the body-fixed frame, we choose to left translate to the identity, which yields the body-fixed angular velocity

$$\Omega = R^T \dot{R} \tag{1}$$

Since the tangent space of $\mathbb{R}^3$ is again $\mathbb{R}^3$, the kinematics of the translational motion is simply $v = \dot{p}$.

The unforced translational dynamics of the quadrotor is the same as the dynamics of a particle (center of mass) under the action of the gravitational potential field. The total thrust $f = \sum_{i=1}^{4} f_i$ acts on the acceleration of this particle in the direction of the z-axis.

The unforced rotational dynamics of the quadrotor are given by the Lie-Poisson equations of motion of $\mathfrak{so}(3)^*$ relative to the rigid body bracket. Following [24], we define the angular momentum in the body frame $\Pi := J\Omega$ whence the kinetic energy function on $\mathfrak{so}(3)^*$ becomes

$$K(\Pi) = \frac{1}{2}\Pi^T J^{-1} \Pi \tag{2}$$

Define another set of functions $F_i(\Pi) := \Pi_i, i = \{1, 2, 3\}$. Set F to be the vector-valued function with components F_i and apply the Lie-Poisson equations,

$$\dot{F} = \{F, K\} \tag{3a}$$
$$\dot{\Pi} = -\Pi \cdot (\nabla F(\Pi) \times \nabla K(\Pi)) \tag{3b}$$
$$= \Pi \times J^{-1}\Pi \tag{3c}$$

Expressing these equations in body angular velocities and including the body torques applied by the rotors,

$$J\dot{\omega} = J\omega \times \omega + \tau \tag{4}$$

In summary, we have the following equations of motion for the quadrotor describing the states at each instant in time.

$$\dot{p} = v \tag{5a}$$
$$m\dot{v} = (-mg + f\,R)e_3 \tag{5b}$$
$$\dot{R} = R\Omega \tag{5c}$$
$$J\dot{\omega} = J\omega \times \omega + \tau \tag{5d}$$

where e_3 is the third standard unit vector in the Cartesian coordinate system and g is the constant gravitational acceleration. We have already mentioned that the total thrust is the sum of the individual forces of the propellers. Since the front, back and left, right propellers rotate in opposite directions, they generate a net torque around the z axis. While the moment arm L and the difference between the forces f_1 and f_3 generate a net torque around the y-axis, the difference between the forces f_2 and f_4 generate a net torque around the x-axis. This is summarized with the matrix equation given below

$$\begin{bmatrix} f \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & -L & 0 & L \\ L & 0 & -L & 0 \\ K_\tau & K_\tau & -K_\tau & -K_\tau \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix}. \tag{6}$$

III. CONTROL DESIGN

We parametrize the 12-dimensional state space of the quadrotor by selecting six position variables and their time derivatives. The first three position variables (x, y, z) specify the center of mass, p, of the quadrotor, while the remaining three position variables $\left(\frac{r_{13}}{r_{33}}, \frac{r_{23}}{r_{33}}, \arctan_2(-r_{12}, r_{22})\right)$ specify the orientation of the quadrotor with respect to the inertial frame. The numbers r_{ij} are the elements of the rotation matrix $R \in SO(3)$. The parametrization of the orientation is local and restricted to the connected open subset $U = \{R \in SO(3) : r_{33} > 0\}$ of $SO(3)$. The first two orientation variables are proportional to the x and y components of the thrust vector in the world frame determining the linear horizontal acceleration of the quadrotor in this frame. The last orientation variable corresponds to the heading of the body x-axis in the world frame, which can also be interpreted as the yaw angle in a Body312 Euler angle representation of the orientation. Unlike the traditional Euler-angle parametrization, the orientation variables in our parametrization above enter linearly in the translation dynamics, as shown below, which facilitates the corresponding controller development.

In this parametrization of the state space, a point is specified by the vectors

$$\eta = (\eta_1, \ldots, \eta_6)^T = (x, y, z, \dot{x}, \dot{y}, \dot{z})^T$$
$$\xi = (\xi_1, \ldots, \xi_6)^T$$

where

$$\xi_1 = \frac{r_{13}}{r_{33}}, \qquad \xi_2 = \frac{r_{13}}{r_{33}}$$
$$\xi_3 = \arctan_2(-r_{12}, r_{22}), \qquad \xi_4 = \frac{\dot{r}_{13}r_{33} - r_{13}\dot{r}_{33}}{r_{33}^2}$$
$$\xi_5 = \frac{\dot{r}_{23}r_{33} - r_{23}\dot{r}_{33}}{r_{33}^2}, \qquad \xi_6 = \frac{\dot{r}_{22}r_{12} - r_{22}\dot{r}_{12}}{r_{12}^2 + r_{22}^2}$$

whose dynamics can be obtained from the global set of equations (5d) as

$$\dot{\eta} = \begin{bmatrix} \eta_4 \\ \eta_5 \\ \eta_6 \\ 0 \\ 0 \\ -mg \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \xi_1 \\ \xi_2 \\ 1 \end{bmatrix} r_{33} f + d_\eta \tag{7a}$$

$$\dot{\xi} = \begin{bmatrix} \xi_4 \\ \xi_5 \\ \xi_6 \\ a_{1x}(R)\tau_x + b_{1x}(R)\tau_y + \sigma_1 \\ a_{2x}(R)\tau_x + b_{2y}(R)\tau_y + \sigma_2 \\ a_{3x}(R)\tau_x + b_{3z}(R)\tau_z + \sigma_3 \end{bmatrix} + d_\zeta \tag{7b}$$

where $a_{(\cdot)}$, $b_{(\cdot)}$ and $\sigma_{(\cdot)}$ are given in Appendix A. We have introduced the terms d_η and d_ζ as additional persistent disturbances on the system that may arise, for example, from misalignment of the quadrotor blades, as shown below in Section IV.

The control goal is to asymptotically stabilize the center of mass position and the heading of the quadrotor. In other words, if $(\eta_{1d}, \eta_{2d}, \eta_{3d}, \xi_{3d})$ denote the desired values for $(\eta_1, \eta_2, \eta_3, \xi_3)$, we would like to have a real number $T > 0$ such that given any $\epsilon > 0$

$$\begin{aligned} &\|\big(\eta_1, \eta_2, \eta_3, \xi_3, \dot{\eta}_1, \dot{\eta}_2, \dot{\eta}_3, \dot{\xi}_3\big) \\ &- \big(\eta_{1d}, \eta_{2d}, \eta_{3d}, \xi_{3d}, \dot{\eta}_{1d}, \dot{\eta}_{2d}, \dot{\eta}_{3d}, \dot{\xi}_{3d}\big)\| < \epsilon \end{aligned} \tag{8}$$

for all $t > T$.

The four available control inputs can be viewed as the thrust along body z-axis and a torque about each body axis. We use feedback linearization to facilitate later design of controllers based on L_1-optimal control. This is the one of the main reasons the parametrization mentioned at the beginning of this section is selected. The L_1-optimal controller that is going to be designed assumes that the nominal dynamics can be reduced to a double integrator; however, the global representation of the orientation kinematics (1) cannot be put in this form.

To begin the feedback linearization procedure, we choose the thrust f as

$$f = \frac{1}{r_{33}} h(\eta_3, \eta_6) \tag{9a}$$

$$h(\eta_3, \eta_6) := mg + u_z(\eta_3, \eta_6) \tag{9b}$$

where u_z is an auxiliary controller to be designed in the sequel. The torques $\tau_{(\cdot)}$ are chosen as the solution to the equations

$$\begin{bmatrix} a_{1x}(R) & b_{1y}(R) & 0 \\ a_{2x}(R) & b_{2y}(R) & 0 \\ a_{3x}(R) & 0 & b_{3z}(R) \end{bmatrix} \begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} + \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \end{bmatrix} = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} \tag{10}$$

Note that a unique solution exists as long as $r_{33} \neq 0$. The right-hand side of these equations are auxiliary control variables to be designed as L_1-optimal controllers that asymptotically stabilize ξ-dynamics. As a result of these manipulations, the dynamics (7) become

$$\dot{\eta} = f_\eta(\eta) + g(\eta)\xi + d_\eta \tag{11a}$$

$$\dot{\xi} = f_\xi(\xi) + v + d_\zeta \tag{11b}$$

where

$$f_\eta(\eta) = \begin{bmatrix} \eta_4 & \eta_5 & \eta_6 & 0 & 0 & u_z(\eta) \end{bmatrix}^T \tag{12a}$$

$$g(\eta) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ h(\eta) & 0 & 0 & 0 & 0 & 0 \\ 0 & h(\eta) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{12b}$$

$$f_\xi(\xi) = \begin{bmatrix} \xi_4 & \xi_5 & \xi_6 & 0 & 0 & 0 \end{bmatrix}^T \tag{12c}$$

$$v = \begin{bmatrix} 0 & 0 & 0 & v_x & v_y & v_z \end{bmatrix}^T \tag{12d}$$

We observe that the selected thrust (9) asymptotically stabilizes the height, z, of the quadrotor. In the remainder of this section, we are going to choose v in an L_1-optimal fashion so that the orientation is stabilized to its desired trajectory, which will, in turn, be constructed to asymptotically stabilize (x, y) and ξ_3.

A. BACKSTEPPING CONTROL

Consider the resulting system dynamics, given by (11). Concentrating on the first component (11a) of this system of differential equations, we notice that by setting $\xi = \phi(\eta)$ with ϕ chosen appropriately, we can asymptotically stabilize η to a desired value. With this observation in mind, let

$$\phi(\eta) = \begin{bmatrix} \frac{u_x}{h} & \frac{u_y}{h} & \xi_{3d} & 0 & 0 & \dot{\xi}_{3d} \end{bmatrix} \tag{13}$$

where u_x and u_y are going to be selected as the output of an L_1-optimal stabilizing linear controller. Consider the change of variables

$$\zeta = \xi - \phi(\eta) \tag{14}$$

We substitute this relation into (11) and consider the auxiliary control v as functions of ζ to obtain

$$\dot{\eta} = A\eta + Bu(\eta) + g(\eta)\zeta + d_\eta \tag{15a}$$

$$\dot{\zeta} = A\zeta + Bv(\zeta) - g_2(\eta, \dot{\eta}) + d_\zeta \tag{15b}$$

The various other terms are given by

$$A = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ I \end{bmatrix} \tag{16a}$$

$$g_2(\eta, \dot{\eta}) = \begin{bmatrix} \frac{\dot{u}_x h - \dot{h} u_x}{h^2} & \frac{\dot{u}_y h - \dot{h} u_y}{h^2} & 0 & 0 & 0 & 0 \end{bmatrix}^T \tag{16b}$$

The dynamics in (15b) and (15b) consists of the following nominal linear system, perturbed by the terms $g(\eta)\zeta$ and g_2 respectively.

$$\dot{\eta} = A\eta + Bu(\eta) \tag{17a}$$

$$\dot{\zeta} = A\zeta + Bv(\zeta) \tag{17b}$$

Proposition III.1. Assume that the auxiliary controllers u_x, u_y, u_z and v_x, v_y, v_z in Equation 17b are designed so that

the origin of (17b) is exponentially stable. Then, when the disturbance terms d_η and d_ζ are zero, the origin of the full system 15b is exponentially stable.

Proof: It suffices to show that the perturbation terms in (15) are locally Lipschitz and vanish at the origin $(\eta, \zeta) \equiv (0, 0)$. This is trivially true for the term $g(\eta)\zeta$ as $g(\eta)$ is bounded and is continuously differentiable. On the other hand, the second term $g_2(\eta, \dot{\eta})$, is comprised of the time derivative of $\frac{u_x}{h}$ and $\frac{u_y}{h}$. Notice that all of these functions are at least continuously differentiable functions of $(\eta, \dot{\eta})$ and therefore, are locally Lipschitz. Moreover, since u_x, u_y are linear functions and h is an affine function of η, the functions $\frac{u_x}{h}$ and $\frac{u_y}{h}$ are identically zero whenever η is identically zero. This, in particular, implies that under these circumstances, the time derivative of these functions are also identically zero.

From this analysis, we can conclude by [10, p. 341, Lemma 9.1] that the origin of the perturbed system (15) is exponentially stable.

Remark 1. In purely model inversion techniques, in order to feedback linearize equation (15) one would require knowledge of g_2. However, the computation of g_2 is not feasible since it requires acceleration level measurements, which are known to be extremely noisy. In contrast, proposition III-A means that to achieve exponential stability we do not need to use this malevolent signal in our control law.

B. L_1 OPTIMAL CONTROL

In this section, we present our main result, which is the design of the control inputs $u(\eta)$ and $v(\zeta)$ for the system (15) using the L_1-optimal control design procedure from [20] and [21].

We first use the stable factorization approach, given an exposition in [22], that allows one to parametrize the set of all controllers that stabilize a given (linear) system as well as the set of all stable transfer matrices that are achievable.

The first step of the approach is to factor the plant transfer matrix G into a "ratio" of the form

$$G(s) = N(s)\,[D(s)]^{-1} = \left[\tilde{D}(s)\right]^{-1} \tilde{N}(s) \tag{18}$$

where N, D, $\tilde{N}$, $\tilde{D}$ are stable rational matrices, i.e., every element of each matrix is proper and has all of its poles in the open left half-plane. Next, one solves the so-called Bezout identities:

$$\begin{bmatrix} Y(s) & X(s) \\ -\tilde{N}(s) & \tilde{D}(s) \end{bmatrix} \begin{bmatrix} D(s) & -\tilde{X}(s) \\ N(s) & \tilde{Y}(s) \end{bmatrix} = I \tag{19}$$

where $X, \tilde{X}, Y, \tilde{Y}$ are also stable rational matrices. It can then be shown that the set of all controllers that stabilize G is given by [22]

$$\left\{ -\left(Y - R\tilde{N}\right)^{-1} \left(X + R\tilde{D}\right) \right\} \tag{20}$$

where R is an arbitrary matrix of appropriate dimensions whose elements are stable rational functions.

We notice that the nominal plant $G(s)$ of equation (18) given by (17) is a system of double integrators. It is then routine to determine the various other matrices in (18) and (19) using the techniques of [25]. This gives

$$G(s) = \frac{1}{s^2} \begin{bmatrix} I \\ sI \end{bmatrix} \tag{21a}$$

$$N(s) = \tilde{N}(s) = \frac{1}{(s+1)^2} \begin{bmatrix} I \\ sI \end{bmatrix} \tag{21b}$$

$$D(s) = \frac{s^2}{(s+1)^2} \cdot I \tag{21c}$$

$$\tilde{D}(s) = \frac{1}{(s+1)^2} \begin{bmatrix} (s^2+2s)I & -2I \\ -sI & (s^2+1)I \end{bmatrix} \tag{21d}$$

$$X(s) = \tilde{X}(s) = \frac{1}{(s+1)^2} \begin{bmatrix} 1+2s & 2+4s \end{bmatrix} I \tag{21e}$$

$$Y(s) = \frac{s^2+4s+2}{(s+1)^2} \cdot I \tag{21f}$$

$$\tilde{Y}(s) = \frac{1}{(s+1)^2} \begin{bmatrix} (s^2+2s+2)I & 2I \\ sI & (s^2+4s+1)I \end{bmatrix} \tag{21g}$$

It is shown in [20] how to choose the "free" parameter R in (20) such that the system error rejects "uncertainties" due to system modeling. Thus, the only error that may remain will be due to the measurement noise. The procedure involves producing a sequence $\{R_k\}$ and examining the resulting controller

$$C_k = -\left(Y - R_k\tilde{N}\right)^{-1} \left(X + R_k\tilde{D}\right) \tag{22}$$

and the error dynamics

$$e = P_{1k}\zeta + P_{2k}w \tag{23}$$

where ζ stands for all the persistent disturbances exerted on the system (15), w denotes the measurement error, and the transfer matrices P_{1k} and P_{2k} are given by

$$P_{1k} = N(Y - R_k\tilde{N}) \tag{24a}$$

$$P_{2k} = I - D(Y - R_k\tilde{N}) \tag{24b}$$

The sequence $\{R_k\}$ in the linear compensator expression (22) can be chosen such that as $k \to \infty$, $\|P_{1k}\|_{\hat{A}} \to 0$ and $\|P_{2k}\|_{\hat{A}} \to 1$, with the $\hat{A}$ norm denoting the L_∞-gain of the system P_i. As $\|P_{1k}\|_{\hat{A}} \to 0$, the rejection of persistent disturbances gets better and better. Since minimizing the L_∞-gain of the operator P_i minimizes the L_1-gain of the input signal, this procedure amounts to designing an L_1-optimal controller for the quadrotor. This implies that in the absence of measurement noise, the mean value of the error over time will be minimized.

The linear controllers consist of two different transfer functions that accept position and velocity errors as inputs, respectively. Each yields control forces/torques that are added together to form the final linear controller. The forms of the resulting linear controllers derived using this robust control framework are given below.

$$\frac{U(s)}{E_p(s)} = \frac{2k_1s^2 + k_1k_2s + k_1^2}{s^2} \tag{25a}$$

$$\frac{U(s)}{E_v(s)} = \frac{2k_2s^2 + k_2^2s + k_1k_2}{s^2} \tag{25b}$$

where k_1 and k_2 positive are gains to be tuned. The tuning process is not so difficult to achieve once it is noted that k_1 acts as the proportional (spring) and k_2 as the derivative (damping) gain of the system. Each of the controllers $\{u_x, u_y, u_z, v_x, v_y, v_z\}$ mentioned in the Section III-A are of this form with appropriately chosen k_1, k_2 gains. Particular choices of these gains, suited for our quadrotor, are given in Section IV.

IV. SIMULATION RESULTS

This section presents simulations of the proposed (LARS) controller on a model of the Quanser Qball quadrotor [23] taking into account the various sources of modeling uncertainties. The controller is compared with that presented in [8], which we refer to as RGTC standing for ''Robust Geometric Tracking Controller''. The simulations are required in order to demonstrate performance for the case when only model uncertainty is present, whereas in experiment some measurement noise is present. We also present simulations which include measurement noise in the model. The Quanser Qball has the following properties as given in its user manual:

$$J = \text{diag}\left([0.03, 0.03, 0.03]\right)\ \text{kg}\cdot\text{m}^2 \tag{26a}$$

$$m = 1.4\,\text{kg},\quad \omega = 15\,\frac{\text{rad}}{\text{sec}} \tag{26b}$$

where ω is the actuator bandwidth of each propeller. That is, the force output of each propeller is achieved by passing the command through a low-pass filter with $\frac{15}{2\pi}$Hz. bandwidth. This, of course, is a huge limitation on the range fast of motions achievable. Therefore, we design a simple lead controller that moves the pole at $\omega = \frac{15}{2\pi}$Hz to $\frac{37.5}{2\pi}$Hz. This mini controller can be expressed in frequency domain as

$$\frac{F(s)}{U(s)} = C(s) = 2.5\frac{s+15}{s+37.5} \tag{27}$$

The quadrotor is subject to several kinds of uncertainties, the most straightforward of which are its mass and its inertia tensor. More subtle uncertainties include misalignment of the rotors with respect to the quadrotor frame. That is, although the rotors of the blades are supposed to be parallel to the frame of the quadrotor, in reality, they might be slightly tilted causing the application of unmodeled forces and moments on the quadrotor. Such rotor misalignment has been observed on the quadrotor in our laboratory. Moreover, if the center of mass of the quadrotor is not located exactly at the center, the manner in which the torques enter into the rotational dynamics is no longer given by (6). The visualization of the ''actual'' quadrotor in Figure 2 sheds light on the derivation of its ''actual'' dynamics. In this figure, we see that the plane of the quadrotor frame, P, has a normal vector N. In the ideal model, the forces of each of the propellers are along this normal. The direction of these forces deviate from this normal in reality, as depicted in Figure 2, giving rise to the dynamical equations of motion presented in (28)

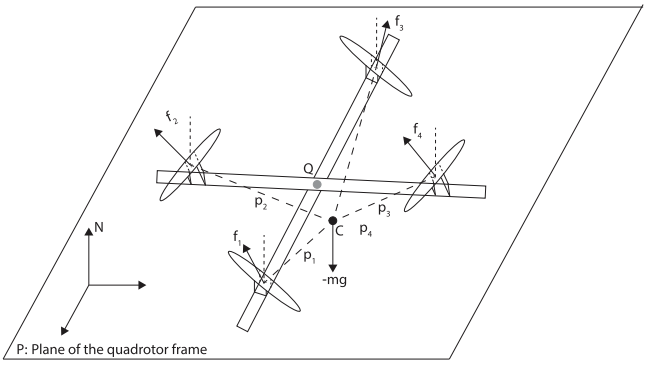


FIGURE 2. "Actual" Quadrotor Model: The direction of the thrust forces of each of the propellers deviate from the normal vector to the plane of the quadrotor frame. The CoM is not exactly situated at the center of the quadrotor frame.

$$J\dot{\omega} + \omega \times J\omega = \sum_{i=1}^{4} f_i \left[K_\tau \sigma(i) I + \hat{p}_i\right] R_i \cdot e_3 \tag{28a}$$

$$m\ddot{p} = \sum_{i=4}^{4} (-mgI + f_i R_i) \cdot e_3 \tag{28b}$$

where $R_i \in SO(3)$ is the orientation of the i^{th} propeller with respect to the quadrotor frame, e_3 is the third standard Cartesian unit vector, p_i is the position vector, in the quadrotor frame, from the center of mass C to the center of the i^{th} propeller. The hat operator $\hat{}$ is the usual isomorphism from $\mathbb{R}^3$ to $\mathfrak{so}(3)$. Lastly, I is the 3×3 identity matrix and $\sigma : \{1, 2, 3, 4\} \to \{-1, 1\}$ is defined by

$$\sigma(i) = \begin{cases} 1 & \text{if } i = 1, 2 \\ -1 & \text{if } i = 3, 4 \end{cases}$$

The experimentally determined values of $R_i e_3$ for the four rotors are given by

$$R_1 e_3 = \left[0.03\ 0.03\ 0.9991\right]^T \tag{29a}$$

$$R_2 e_3 = \left[0.06\ 0.03\ 0.9977\right]^T \tag{29b}$$

$$R_3 e_3 = \left[0.03\ 0.06\ 0.9977\right]^T \tag{29c}$$

$$R_4 e_3 = \left[0.09\ 0.09\ 0.9919\right]^T \tag{29d}$$

Finally, the measurements of the quadrotor pose and velocity are never perfect. In the simulations, they will be modeled as a Gaussian noise acting on both the linear and angular readings additively. In what follows, we are going to present simulation results of our proposed controller and compare it with one of the most prominent robust controllers in the literature provided by [15].

The value of gain parameters used in our controller is given in Table 1. The value of parameters defined in [8] are given in Table 2 where the robust control parameters are unchanged, however the proportional and derivative gains have been scaled to be comparable to the gains in Table 1. Finally, the statistics of the additive gaussian white noise used to corrupt the measurements are given in Table 3.

TABLE 1. Gains used in our controller.

Loop	k_1	k_2
x	3	3.4293
y	3	2.6944
z	3	2.6944
ξ_1	176.2000	26.5481
ξ_2	176.2000	26.5481
ξ_3	26.4300	10.2820

TABLE 2. Parameters from [8].

Parameter	Value
K_R	$5.3394\, I_3$
K_Ω	0.7
k_x	6
k_v	5.4
c_1	3.6
c_2	0.6
ϵ_x	0.04
ϵ_R	0.04

TABLE 3. Variances of additive zero mean gaussian white noise.

Parameter	Value
σ_p^2	10^{-6}
$\sigma_{\dot{p}}^2$	0.01
σ_R^2	10^{-4}
σ_ω^2	0.01

A. SET-POINT CONTROL

We command a constant desired point in the Cartesian space to which the quadrotor center of mass should converge while potentially suffering from uncertainties in parameters and measurement noise. The results of a simulation for the case when there is no measurement noise but parameter uncertainty exists is given in Figure 3. The steady state error in position is zero in the case of the LARS controller (Figure 3c). The orientations errors also have zero steady state values (Figure 3a). For the RGTC controller, the position errors reach a steady non-zero value (Figure 3d). The orientation also does not converge to the desired values (Figure 3b). Tables 4 and 5 give the errors in steady state for both controllers.

When noise is added, the results follow a similar trend (Figures 4). As seen in Figures 4a and 4c, the average value of the errors in orientation and position are nearly zero for the LARS controller. This is confirmed by the mean values of the errors in position for the experiment in Tables 4 and 5. In the case of the RGTC controller, we see that the mean values of the errors reach a constant value that is slightly offset from zero (See Figures 4b and 4d, and the mean values given in Table 4). This shows that the presence of measurement noise does not qualitatively change the relative behaviour of the two controllers.

TABLE 4. Mean values of errors obtained in simulations of LARS and RGTC controllers. S: Set-point regulation, C: Tracking a circular trajectory.

Task	Noise	Controller	$\mu_{e_x}[m]$	$\mu_{e_y}[m]$	$\mu_{e_z}[m]$
S	no	LARS	1.4096e-7	9.0064e-8	−1.7917e-7
S	yes	LARS	−7.4749e-4	−7.2466e-4	−7.4496e-4
C	no	LARS	2.01e-5	9.4342e-6	−1.33e-6
C	yes	LARS	−7.2708e-4	−7.1625e-4	−7.4137e-4
S	no	RGTC	0.0658	0.0288	0.0697
S	yes	RGTC	0.0704	0.0358	0.0743
C	no	RGTC	0.0791	0.0336	0.0754
C	yes	RGTC	0.0811	0.0356	0.0815

TABLE 5. Standard deviations of errors obtained in simulations of LARS and RGTC controllers. S: Set-point regulation, C: Tracking a circular trajectory.

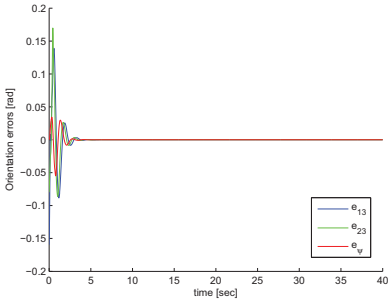
Task	Noise	Controller	$\sigma_{e_x}[m]$	$\sigma_{e_y}[m]$	$\sigma_{e_z}[m]$
S	no	LARS	7.05e-7	4.785e-7	6.509e-7
S	yes	LARS	0.0111	0.0081	0.0085
C	no	LARS	0.1434	0.1434	0.0001
C	yes	LARS	0.1430	0.1451	0.0086
S	no	RGTC	6.404e-4	2.503e-4	1.698e-4
S	yes	RGTC	0.0082	0.0067	0.0065
C	no	RGTC	0.1291	0.1183	0.0207
C	yes	RGTC	0.1249	0.1203	0.0134

B. TRAJECTORY TRACKING

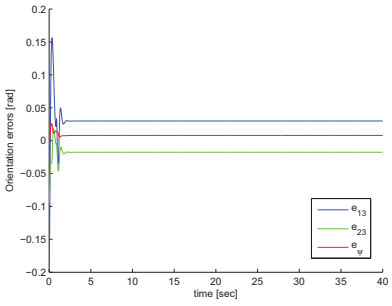
The quadrotor is commanded to follow a trajectory

$$\begin{bmatrix} x(t) \\ y(t) \\ z(t) \\ \xi_3(t) \end{bmatrix} = \begin{bmatrix} 0.5\sin(0.5t) \\ 0.5\cos(0.5t) \\ 1.5 \\ 0 \end{bmatrix} \tag{30}$$

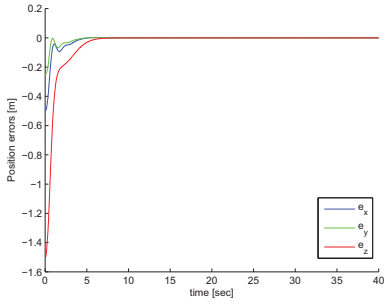
Simulation results are presented for the case when measurement noise is both absent (Figure 5) and present (Figure 7). Parameter uncertainty is always assumed to exist. First, we look at the case where measurement noise is not included in the simulation. As seen in Figures 5c and 5d, neither controller tracks the desired trajectory perfectly due to the actuator limitations and the prevention of using high gains due to noise. However, the LARS controller rejects the persistent disturbances due to modeling errors, as seen by the mean of the errors in position in Table 4. In the case of the RGTC controller, the mean values of the position errors are not zero. The resulting trajectories can be seen in Figures 6a and 6b. The LARS controllers results in a trajectory concentric with the desired circle, whereas the trajectory acheived by the RGTC controller has a center different from that of the desired trajectory. The desired orientation is tracked in the case of the LARS controller as seen in Figure 5a, whereas this does not occur for the RGTC controller (Figure 5b). Table 4 supports this conclusion, as seen from the mean of the errors. The



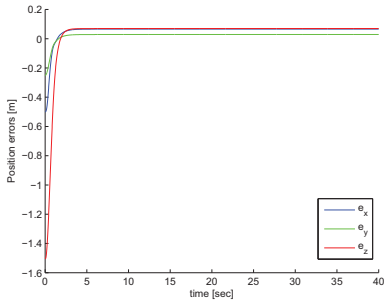
(a) Orientation errors (LARS)



(b) Orientation errors (RGTC)

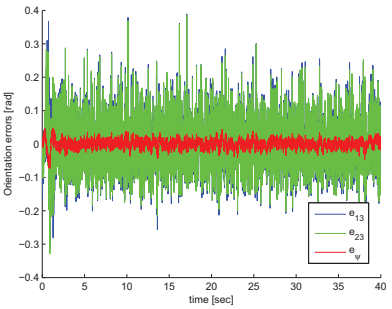


(c) Position errors (LARS)

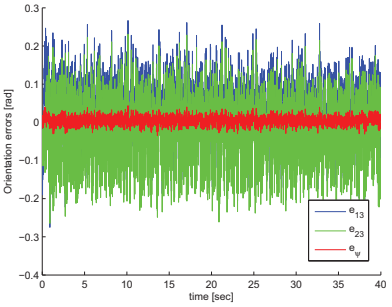


(d) Position errors (RGTC)

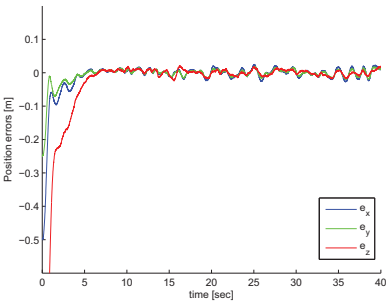
FIGURE 3. Simulation of set-point regulation with parameter uncertainty and no measurement noise. The orientation errors in (a) are zero for the LARS controller while there is a persistent error when using the LARS controller as seen in (b). A similar trend is observed for the position errors in (c) and (d).



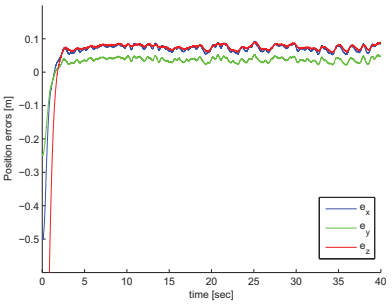
(a) Orientation errors (LARS)



(b) Orientation errors (RGTC)



(c) Position errors (LARS)

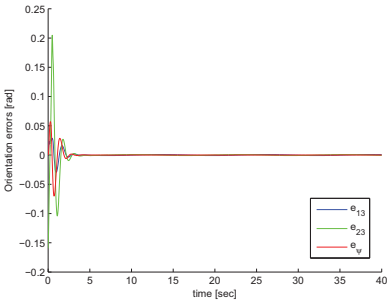


(d) Position errors (RGTC)

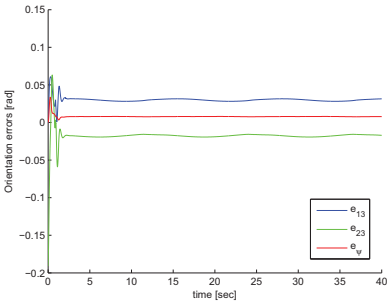
FIGURE 4. Simulation of set-point regulation with parameter uncertainty and measurement noise for the LARS and RGTC controllers. The mean values of the orientation errors tend to zero for the LARS controller (a) while they are not zero for the RGTC controller (b). A similar trend is observed for the position errors in (c) and (d).

standard deviations of the RGTC controller are lower. Note that this does not mean that the mean squared error is lower, due to the larger mean in the case of the RGTC controller.

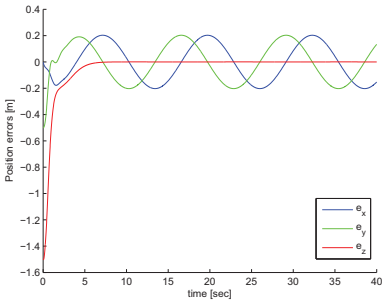
When measurement noise is added, we see that the plots do not change much qualitatively. The effect of measurement noise can easily be seen in Figures 7c and 7d. The statistics of the obtained trajectory in Tables 4 and 5 show that the conclusions from the simulations without noise still hold, that



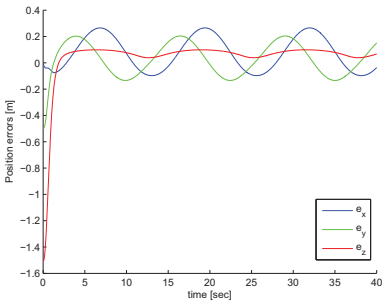
(a) Orientation errors (LARS)



(b) Orientation errors (RGTC)



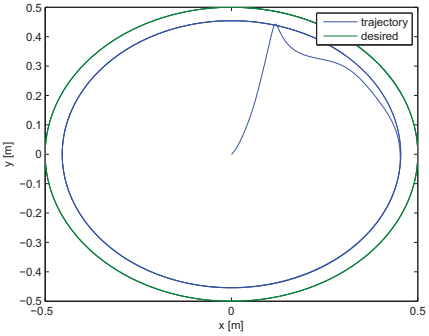
(c) Position errors (LARS)



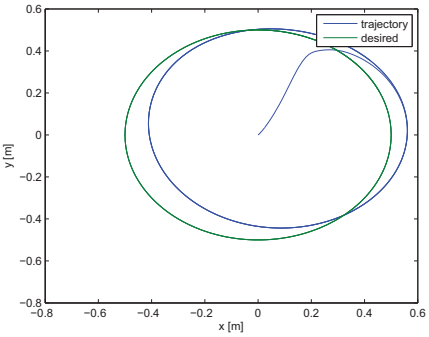
(d) Position errors (RGTC)

FIGURE 5. Simulation of trajectory tracking with parameter uncertainty and no measurement noise. The LARS controller rejects persistent distrubances (parameter uncertainties) yielding a zero-mean steady-state errors in orientation and position as seen in (a) and (c) respectively. In comparison, the same errors due to the RGTC controller have a persistent non-zero mean value as seen in (b) and (d).

is, the mean value of the trajectory for the LARS controller is still nearly zero, whereas for the RTGC controller it is non zero. Figures 7a and 7b show that the position errors appear



(a) Implicit plot (LARS)



(b) Implicit plot (RGTC)

FIGURE 6. Implicit plot of the trajectory tracking simulation for LARS and RGTC controllers. No measurement noise is included in the model. The persistent disturbances causes a deviation in the final path of the quadrotor when running the RGTC controller, whereas the LARS controller can reject these disturbances.

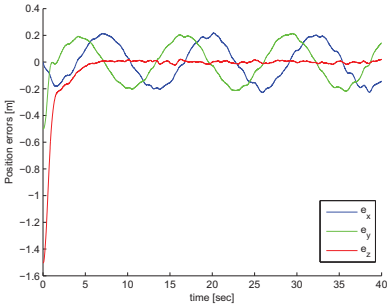
to have a zero mean in the case of the LARS controller but not for the RGTC controller. This is confirmed again in Table 4. Similar to the simulation with no measurement noise, we see that the resulting trajectory for the LARS controller is concentric with the desired one (Figure 8a) whereas that for the RGTC controller is offset (Figure 8b).

V. EXPERIMENTAL IMPLEMENTATION

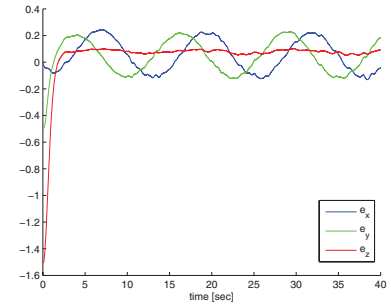
The results of the simulation sections are recreated in an experimental environment with an actual Quanser Qball quadrotor [23]. We also continue our foregoing comparison with the RTGC robust controller [8] by implementing this controller and showing the experimental results with that controller. The experimental setup is as follows. The Quanser Qball is assumed to have the same properties as was given at the start of the Section IV. The linear position feedback is acquired from an 8-camera Vicon vision system [26] working at a rate of 100 Hz. The linear velocity feedback is derived from this position reading by numerical differentiation coupled with a first-order filter with a cut-off frequency of 40 Hz.

$$H(s) = \frac{40s}{s + 40}$$

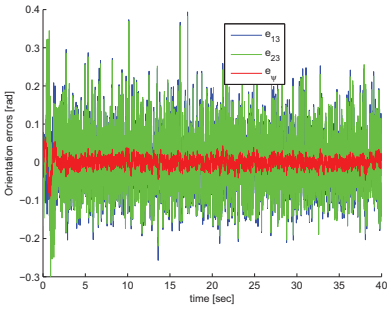
Since the motions of this particular quadrotor cannot really be faster than 20 Hz. primarily due to the restrictions of



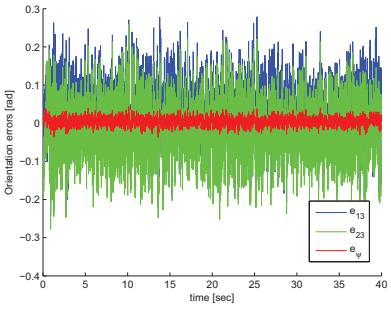
(a) Position errors (LARS)



(b) Position errors (RGTC)



(c) Orientation errors (LARS)

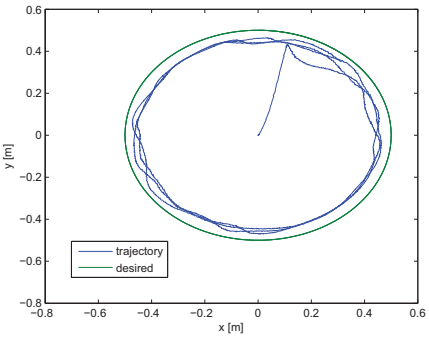


(d) Orientation errors (RGTC)

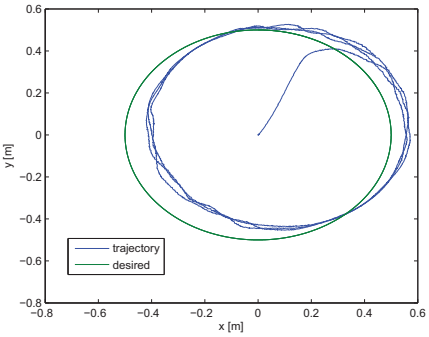
FIGURE 7. **Trajectory tracking with parameter uncertainty and measurement noise. The LARS controller rejects persistent distrubances (parameter uncertainties) yielding a zero-mean steady-state errors in orientation and position as seen in (a) and (c) respectively. In comparison, the same errors due to the RGTC controller have a persistent non-zero mean value as seen in (b) and (d).**

the actuators, the aforementioned velocity measurement is justified to work well.

The orientation of the quadrotor is also acquired using vision data from the Vicon system as follows. We start by



(a) Implicit plot (LARS)



(b) Implicit plot (RGTC)

FIGURE 8. **Implicit plots for the simulation of trajectory tracking task with parameter uncertainty and measurement noise. Again, The persistent disturbances causes a deviation in the final path of the quadrotor when running the RGTC controller, whereas the LARS controller can reject these disturbances.**

placing markers on both ends of the rods. The body x-axis of the quadrotor is then identified as the vector v_1 whose tail is the marker at the "back" and whose head is at the "front" of the quadrotor. We then form the vector v_2 from the "back" marker to the "left" marker and the vector v_3 from the "back" marker to the "top" marker (see Figure 9). Using the Gram-Schmidt orthonormalization process [27], we get an orthonormal body-frame for the quadrotor, whose components when expressed in the inertial frame, form the rotation matrix R.

The body angular velocity of the quadrotor, on the other hand, is acquired from the on-board gyroscope. The gyroscope data is downsampled to $100Hz$ to match the data rate of the Vicon system.

The controller runs on an embedded processor, Gumstix Verdex, stationed on the quadrotor. Its processing power is 600 MHz. The reference commands, control gains and parameters are broadcast to the gumstix via a wireless communication protocol. The software used to prepare the C-code run on the gumstix and to send the various parameters to the gumstix is Matlab/Simulink.

Two types of experiments are conducted using each controller:

- Set-point regulation
- Tracking a circular trajectory at 1 rad/s

TABLE 6. Desired coordinates in the world frame: (i) Set-point (ii) The moving trajectory is a circle in the horizontal plane. The steady state error for RGTC requires us to command a higher height for sufficient ground clearance.

Experiment	LARS	RGTC
Set-point	(0.4,0,0.5)	(0.4,0,0.75)
Circle 1	$(0.4 + \cos(t), \sin(t), 0.5)$	$(0.4 + \cos(t), \sin(t), 0.75)$

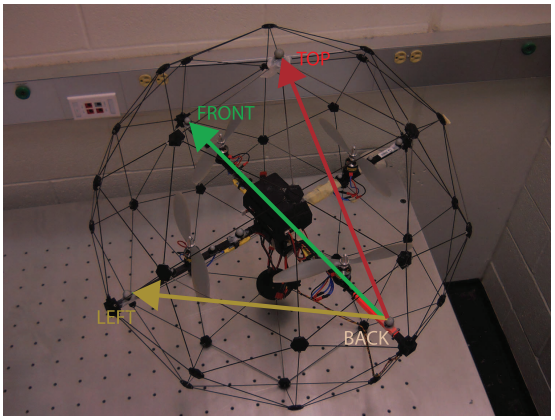


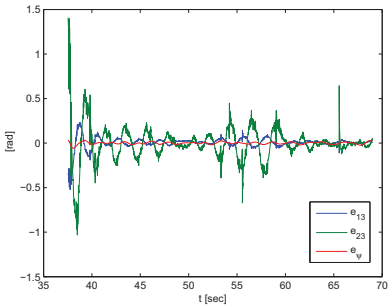
FIGURE 9. Rotation matrix construction: The vectors v_1 from BACK to FRONT, v_2 from BACK to LEFT and v_3 from BACK to TOP are orthonormalized and their components expressed in the inertial frame to form the rotation matrix R

The performance of the position controller is demonstrated in two steps. First, the Qball lifts off from the initial position at the desired height z_d. Once the height has settled, the robot is commanded to the point $(0.4, 0, z_d)$.

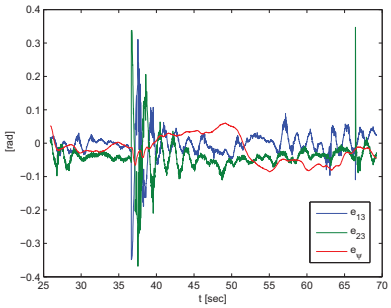
This is due to the fact that large step changes in all commanded variables will often hit the saturation limits, and in the closed lab space potentially leads to a crash. In the circle experiment, the command is switched from this point to the trajectory given in Table 6.

A. SET-POINT REGULATION RESULTS

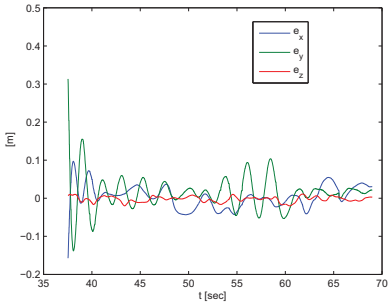
The results of the set-point regulation experiments for both controllers are seen in Figure 10. In Figure 10c we see that the errors decrease from the initial value and tend to a zero-mean behaviour in the case of the LARS controller. There appears to be a disturbance at around $t = 53$s, which is duly rejected. In the case of the RGTC control, after the Qball lifts off, the step position command is applied at around $t = 35$s. The errors decrease, however, e_y and e_z do not have a zero mean steady state behaviour, as seen in Figure 10d. The x-direction position error, e_x, appears to have a zero mean behaviour. The orientation error for the LARS controller follows a similar behaviour (Figure 10a): the orientation errors decrease to a mean value close to zero, even after the disturbance. The orientation errors for the RGTC controller are not close to zero, as seen in Figure 10b, except the e_{13} error. This is logical, since in the dynamics (5) the dynamics in x and y are directly affected by the r_{13} and r_{23} components of the rotation matrix R, and hence errors in those values will result in errors in x and y even with ideal thrust generation.



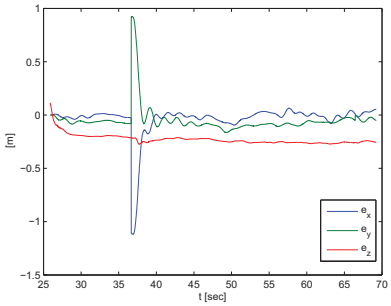
(a) Orientation errors (LARS)



(b) Orientation error (RGTC)



(c) Position errors (LARS)

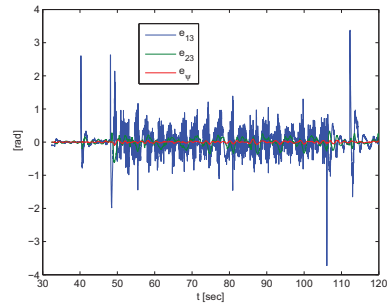


(d) Position errors (RGTC)

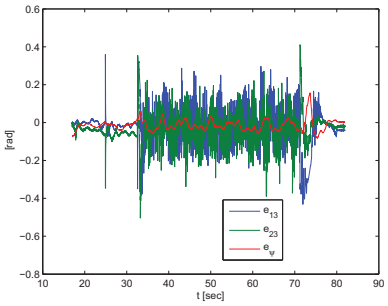
FIGURE 10. Set-point regulation experiment. The LARS controller results in orientation errors with a small mean value than those due to the RGTC as seen in (a) and (b). A similar trend is seen in the position errors as seen in (c) and (d).

B. TRAJECTORY TRACKING RESULTS

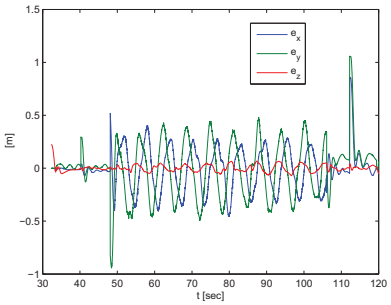
The Qball quadrotor is desired to track the trajectory in Table 6. Figure 12 gives the result of the experiment for both controllers. In figure 11c the quadrotor is commanded to hover at the inital position $(0.4, 0, 0.5)$ at $t = 31$s.



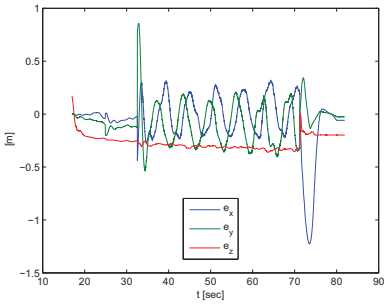
(a) Orientation errors (LARS)



(b) Orientation errors (RGTC)

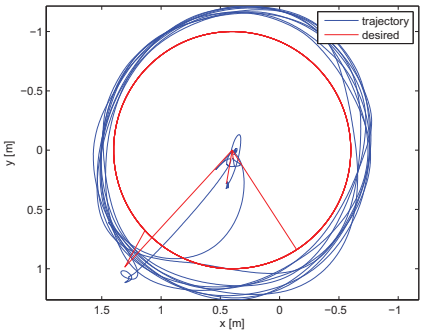


(c) Position errors (LARS)

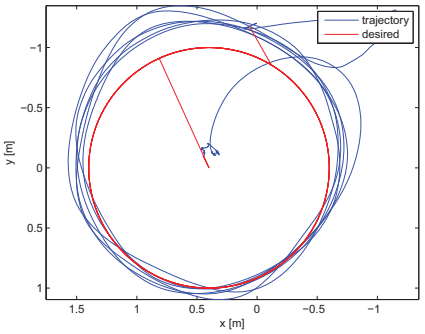


(d) Position errors (RGTC)

FIGURE 11. **Trajectory tracking experiment with a circular trajectory. The LARS controller yields nearly zero mean steady state errors in orientation and position. The RGTC controller yields errors which do not have zero mean behaviour.**



(a) Implicit plot (LARS)



(b) Implicit plot (RGTC)

FIGURE 12. **Trajectory tracking experiment with a circular trajectory. The LARS controller yields a correctly nearly centered circular trajectory, having a nearly zero mean translational steady state error. The circular trajectory of the RGTC controller has a shifted center, product of nonzero steady state errors.**

The command is then switched to the circular trajecteory given in Table 6 at approximately $t = 49$s. The step command results in the error shooting up, which reduces to the sinusoidal errors with a nearly zero mean, similar to the behaviour seen in Figure 10c. At $t = 105$s the desired trajectory is switched to a set-point along the tangent to the circle at that instant. At $t = 115$s the Qball is commanded to return to (0.4, 0, 0.5), resulting in a jump in errors due to the step command. We see that, in spite of the step commands in positions, resulting in step commands in the desired orientations (See Figure 11a) the height error does not increase, as expected due to the LARS control design. The orientation errors are seen to have a nearly zero mean behaviour in Figure 11a. The resulting trajectory is a distorted circle with a center almost identical to that of the desired trajectory (See Figure 12a).

Figure 11d shows the errors in position for a similar experiment using the RGTC controller. The Qball lifts off at $t = 18$s, and hovers at a point slightly offset from the desired point. Due to the steady state error in height always present in the RGTC controller, the command is higher ($z_d = 0.75$m). This yields sufficient clearance above the ground so as to minimize the ground effect. The Qball is then commanded to follow the circular trajectory in Table 6. The error in x appears to have a zero mean behaviour, whereas y and z have non-zero mean errors. This is reflected in the Figure 12b where the resulting trajectory is circular but shifted in the negative y direction with respect to the desired circular trajectory. The orientation plot in Figure 11b shows that there is some error in e_{23} at least in the set-point tracking part of the experiment.

Videos of experiments conducted with the quadrotor are detailed in Appendix B.

VI. CONCLUSION

We have presented an L_1-optimal robust controller for the quadrotor dynamical system that rejects persistent disturbances. Consequently, the only remaining errors in the steady-state response of the system are due to measurement noise. We have derived an "actual" model for the quadrotor under practical modeling uncertainties. We used this model to conduct simulation studies for both set-point and trajectory tracking applications and compared the performance of our controller with one of the more prominent robust controller in the literature. The same comparison has also been carried out experimentally and the performance of the system, predicted in the simulations, were seen to be very closely replicated.

APPENDIX A
EQUATIONS OF MOTION UNDER THE SELECTED PARAMETRIZATION

We use the quadrotor dynamic equations of motion (5d) to derive the equations governing the parametrization. We have $\eta = (x, y, z)$ and $\xi = \left(\frac{r_{13}}{r_{33}}, \frac{r_{23}}{r_{33}}, \arctan_2(-r_{12}, r_{22})\right)$. We differentiate with respect to time once and substitute from the kinematic equations (1) and differentiate again before substituting from the equations (5d). For the translational variables η, this gives

$$\ddot{\eta}_1 = \ddot{x} = \frac{r_{13}}{m} f \tag{31a}$$

$$\ddot{\eta}_2 = \ddot{y} = \frac{r_{23}}{m} f \tag{31b}$$

$$\ddot{\eta}_3 = \ddot{z} = -g + \frac{r_{33}}{m} f \tag{31c}$$

and for the rotational dynamics,

$$\ddot{\xi}_1 = a_{1x}(R)\tau_x + b_{1y}(R)\tau_y + \sigma_1(R, \omega) \tag{32a}$$

$$\ddot{\xi}_2 = a_{2x}(R)\tau_x + b_{2y}(R)\tau_y + \sigma_2(R, \omega) \tag{32b}$$

$$\ddot{\xi}_3 = a_{3x}(R)\tau_x + b_{3z}(R)\tau_z + \sigma_3(R, \omega) \tag{32c}$$

where the functions $a_{1x}, a_{2x}, a_{3x}, b_{1y}, b_{2y}, b_{3z}, \sigma_1, \sigma_2, \sigma_3$ are given as follows:

$$a_{1x}(R) = \frac{r_{13}r_{32} - r_{12}r_{33}}{J_r r_{33}^2}$$

$$a_{2x}(R) = \frac{r_{11}r_{33} - r_{13}r_{31}}{J_p r_{33}^2}$$

$$a_{3x}(R) = \frac{r_{12}r_{23} - r_{13}r_{22}}{J_r\left(r_{12}^2 + r_{22}^2\right)}$$

$$b_{1y}(R) = \frac{-r_{13}r_{31} + r_{11}r_{33}}{J_p r_{33}^2}$$

$$b_{2y}(R) = \frac{-r_{23}r_{31} + r_{21}r_{33}}{J_p r_{33}^2}$$

$$b_{3z}(R) = \frac{-r_{12}r_{21} + r_{11}r_{22}}{J_y\left(r_{12}^2 + r_{22}^2\right)}$$

$$\begin{aligned}
\sigma_1(R, \omega) = \frac{1}{J_p J_r r_{33}^3}\Big(&2J_p J_r r_{13} r_{32}^2 w_x^2 - 4J_p J_r r_{13} r_{31} r_{32} w_x w_y \\
&+ 2J_p J_r r_{13} r_{31}^2 w_y^2 \\
&+ 2J_p J_r r_{12} r_{33} w_x(-r_{32} w_x + r_{31} w_y) \\
&+ J_r(-J_r + J_y) r_{33}(-r_{13} r_{31} + r_{11} r_{33}) w_x w_z \\
&+ J_p(J_r + J_y) r_{12} r_{33}^2 w_y w_z + J_p^2 r_{33} \\
&\times (r_{13} r_{32} - r_{12} r_{33}) w_y w_z - J_p r_{13} r_{33} \\
&\times (J_r r_{31} w_x + (J_r + J_y) r_{32} w_y) w_z + J_p J_r r_{11} r_{33} \\
&\times \left(2r_{32} w_x w_y - 2r_{31} w_y^2 + r_{33} w_x w_z\right)\Big)
\end{aligned}$$

$$\begin{aligned}
\sigma_2(R, \omega) = \frac{1}{J_p J_r r_{33}^3}\Big(&2J_p J_r r_{23} r_{32}^2 w_x^2 - 4J_p J_r r_{23} r_{31} r_{32} w_x w_y \\
&+ 2J_p J_r r_{23} r_{31}^2 w_y^2 + 2J_p J_r r_{22} r_{33} w_x \\
&\times (-r_{32} w_x + r_{31} w_y) \\
&+ J_r(-J_r + J_y) r_{33}(-r_{23} r_{31} + r_{21} r_{33}) w_x w_z \\
&+ J_p(J_r + J_y) r_{22} r_{33}^2 w_y w_z + J_p^2 r_{33} \\
&\times (r_{23} r_{32} - r_{22} r_{33}) w_y w_z \\
&- J_p r_{23} r_{33}(J_r r_{31} w_x + (J_r + J_y) r_{32} w_y) w_z \\
&+ J_p J_r r_{21} r_{33}\left(2r_{32} w_x w_y - 2r_{31} w_y^2 + r_{33} w_x w_z\right)\Big)
\end{aligned}$$

$$\begin{aligned}
\sigma_3(R, \omega) = \frac{1}{J_r J_y\left(r_{12}^2 + r_{22}^2\right)^2}\Big(&J_r J_y r_{12}^3 r_{21} w_x w_y \\
&+ J_r J_y r_{12} r_{21} r_{22}^2 w_x w_y - J_r(-J_p + J_r) \\
&\times (r_{12} r_{21} - r_{11} r_{22})\left(r_{12}^2 + r_{22}^2\right) w_x w_y \\
&- (J_p + J_r) J_y r_{12}^2 r_{13} r_{22} w_y w_z - (J_p + J_r) J_y \\
&\times r_{13} r_{22}^3 w_y w_z + (J_p + J_r) J_y r_{12}^3 r_{23} w_y w_z \\
&+ (J_p + J_r) J_y r_{12} r_{22}^2 r_{23} w_y w_z - J_y^2\left(r_{12}^2 + r_{22}^2\right) \\
&\times (-r_{13} r_{22} + r_{12} r_{23}) w_y w_z - 2J_r J_y r_{12}^2 r_{13} w_x \\
&\times (r_{23} w_x - r_{21} w_z) 2J_r J_y r_{13} r_{22}^2 w_x \\
&-(-r_{23} w_x + r_{21} w_z) + 2J_r J_y r_{12} r_{22} \\
&\times (r_{13} w_x - r_{23} w_x - r_{11} w_z + r_{21} w_z) \\
&\times ((r_{13} + r_{23}) w_x - (r_{11} + r_{21}) w_z) - J_r J_y r_{11} r_{22}^2 \\
&\times \left(r_{22} w_x w_y + 2r_{23} w_x w_z - 2r_{21} w_z^2\right) \\
&- J_r J_y r_{11} r_{12}^2\left(r_{22} w_x w_y - 2r_{23} w_x w_z + 2r_{21} w_z^2\right)\Big).
\end{aligned}$$

APPENDIX B
VIDEOS OF EXPERIMENTS

The results of trajectory tracking experiments were recorded for some challenging trajectories: (*i*) Horizontal circle with sinusoidal height command [28], (*ii*) Vertical circle [29], and (*iii*) Lissajous curve [30].

The experiments consist of three phases. In the first phase, the quadrotor is commanded to hover at a height above the starting position, then commanded to move to a starting point at the same height. The second phase consists of trajectory tracking. During the third phase the following landing scheme is implemented. A predicted position is calculated based on

the quadrotor position and velocity at the end of the second phase and commanded as a set-point. The desired height is decreased in steps until the quadrotor can be powered down safely.

The first trajectory is a horizontal circle with a varying height command [28]. The varying height creates an increased demand on quadrotor actuators. This is further tested when the frequency of the height command is doubled during the motion of the quadrotor.

The second experiment demonstrates the controller performance when gravity acts as a disturbance non-trivially. This is achieved by requiring the quadrotor to track a vertical circle [29].

The third experiment shows the performance of the controller under a significantly aggressive motion: tracking a three dimensional Lissajous curve. The quadrotor motion is seen to be both stable and smooth throughout the experiment [30].

The reader is invited to view further videos on the UTD Robotics channel [31], not all of which use the L_1-optimal controller. It can be observed that the tracking performance is poorer and more sluggish when the quadrotor uses naive PID control.

REFERENCES

[1] M. Spong, "Partial feedback linearization of underactuated mechanical systems," in *Proc. IEEE/RSJ/GI Int. Conf. Intell. Robots Syst. Adv. Robot. Syst. Real World*, vol. 1. Sep. 1994, pp. 314–321.

[2] P. Pounds, R. Mahony, P. Hynes, and J. Roberts, "Design of a four-rotor aerial robot," in *Proc. Australasian Conf. Robot. Autom.*, Nov. 2002, pp. 145–150.

[3] S. Bouabdallah, A. Noth, and R. Siegwart, "PID vs LQ control techniques applied to an indoor micro quadrotor," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, vol. 3. Oct. 2004, pp. 2451–2456.

[4] R. Xu and U. Ozguner, "Sliding mode control of a quadrotor helicopter," in *Proc. 45th IEEE Conf. Decision Control*, Dec. 2006, pp. 4957–4962.

[5] S. Bouabdallah and R. Siegwart, "Backstepping and sliding-mode techniques applied to an indoor micro quadrotor," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2005, pp. 2247–2252.

[6] T. Madani and A. Benallegue, "Backstepping control for a quadrotor helicopter," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2006, pp. 3255–3260.

[7] S. Bouabdallah, P. Murrieri, and R. Siegwart, "Design and control of an indoor micro quadrotor," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 5. May 2004, pp. 4393–4398.

[8] T. Lee, M. Leoky, and N. McClamroch, "Geometric tracking control of a quadrotor UAV on SE(3)," in *Proc. 49th IEEE Conf. Decision Control*, Dec. 2010, pp. 5420–5425.

[9] D. Mellinger and V. Kumar, "Minimum snap trajectory generation and control for quadrotors," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2011, pp. 2520–2525.

[10] H. Khalil, *Nonlinear Systems*. Englewood Cliffs, NJ, USA: Prentice-Hall, 2002.

[11] V. Mistler, A. Benallegue, and N. M'Sirdi, "Exact linearization and noninteracting control of a 4 rotors helicopter via dynamic feedback," in *Proc. 10th IEEE Int. Workshop Robot Human Interact. Commun.*, Jan. 2001, pp. 586–593.

[12] A. Mokhtari and A. Benallegue, "Dynamic feedback controller of euler angles and wind parameters estimation for a quadrotor unmanned aerial vehicle," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 3. May 2004, pp. 2359–2366.

[13] G. V. Raffo, M. G. Ortega, and F. R. Rubio, "An integral predictive/nonlinear control H_∞ structure for a quadrotor helicopter," *Automatica*, vol. 46, no. 1, pp. 29–39, Jan. 2010.

[14] A. Mokhtari, A. Benallegue, and B. Daachi, "Robust feedback linearization and GH_∞ controller for a quadrotor unmanned aerial vehicle," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Aug. 2005, pp. 1198–1203.

[15] T. Lee, M. Leok, and N. H. McClamroch, "Nonlinear robust tracking control of a quadrotor UAV on SE(3)," *Asian J. Control*, vol. 15, no. 2, pp. 391–408, Mar. 2013.

[16] C. Diao, B. Xian, Q. Yin, W. Zeng, H. Li, and Y. Yang, "A nonlinear adaptive control approach for quadrotor uavs," in *Proc. 8th Asian Control Conf. (ASCC)*, May 2011, pp. 223–228.

[17] T. Fernando, J. Chandiramani, T. Lee, and H. Gutierrez, "Robust adaptive geometric tracking controls on SO(3) with an application to the attitude dynamics of a quadrotor uav," in *Proc. 50th IEEE Conf. Decision Control Eur. Control Conf.*, Dec. 2011, pp. 7380–7385.

[18] M. Spong and M. Vidyasagar, "Robust linear compensator design for nonlinear robotic control," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 2. Mar. 1985, pp. 954–959.

[19] M. Spong and M. Vidyasagar, "Robust nonlinear control of robot manipulators," in *Proc. IEEE Conf. Decision Control*, vol. 24. Dec. 1985, pp. 1767–1772.

[20] M. Spong and M. Vidyasagar, "Robust linear compensator design for nonlinear robotic control," *IEEE J. Robot. Autom.*, vol. 3, no. 4, pp. 345–351, Aug. 1987.

[21] M. Vidyasagar, "Optimal rejection of persistent bounded disturbances," *IEEE Trans. Autom. Control*, vol. 31, no. 6, pp. 527–534, Jun. 1986.

[22] M. Vidyasagar, *Control System Synthesis: A Factorization Approach* (Synthesis Lectures on Control and Mechatronics). San Rafael, CA, USA: Morgan & Claypool Publishers, 2011.

[23] Quanser Inc. (2013, Apr.). *Quanser Qball-X4*, Markham, Canada [Online]. Available: http://www.quanser.com/english/html/UVS_Lab/fs_Qball_X4.htm

[24] J. Marsden and T. Ratiu, *Introduction to Mechanics and Symmetry: A Basic Exposition of Classical Mechanical Systems* (Texts in Applied Mathematics). New York, NY, USA: Springer-Verlag, 1999.

[25] C. Nett, C. Jacobson, and M. Balas, "A connection between state-space and doubly coprime fractional representations," *IEEE Trans. Autom. Control*, vol. 29, no. 9, pp. 831–832, Sep. 1984.

[26] Vicon Motion Systems Ltd. (2012). *Vicon MX Giganet Gigabit Ethernet Controller*, Oxford, U.K. [Online]. Available: http://www.vicon.com/products/mxgiganet.html

[27] E. Kreyszig, *Differential Geometry* (Dover books on advanced mathematics). New York, NY, USA: Dover, 1991.

[28] Laboratory for Autonomous Robotics, UT Dallas. (2012). *Quadrotor Tracking a Horizontal Circle with Sinusoidal Height Command*, Cambridge, MA, USA [Online]. Available: http://www.youtube.com/watch?v=xZwQ3Uis4dc

[29] Laboratory for Autonomous Robotics, UT Dallas. (2012). *Quadrotor Tracking a Vertical Circle*, Cambridge, MA, USA [Online]. Available: http://www.youtube.com/watch?v=TqY4TcJXEcY

[30] Laboratory for Autonomous Robotics, UT Dallas. (2012). *Quadrotor Tracking a Lissajous Curve*, Cambridge, MA, USA [Online]. Available: http://www.youtube.com/watch?v=1lIGlwu3Cic

[31] Laboratory for Autonomous Robotics, UT Dallas. (2012). *UTD Robotics Channel on Youtube*, Cambridge, MA, USA [Online]. Available: http://www.youtube.com/UTDRobotics

AYKUT C. SATICI received the B.Sc. and M.Sc. degrees in mechatronics engineering from Sabanci University, Istanbul, Turkey, in 2008 and 2010, respectively, and the Masters degree in mathematics from the University of Texas, Dallas, TX, USA, in 2013. He is currently with the Electrical Engineering Department, University of Texas at Dallas. His current research interests include robotics, geometric mechanics, and cooperative control.

HASAN POONAWALA was born in Mumbai, India, on May 14, 1985. He received the B.Tech. degree in mechanical engineering from the National Institute of Technology, Surathkal, India, in 2007, and the M.S. degree in mechanical engineering from the University of Michigan, Ann Arbor, MI, USA, in 2009. He is currently pursuing the Ph.D. degree in electrical engineering with the University of Texas, Dallas, TX, USA. His current research interests include formation control and connectivity control of mobile robot networks, bipedal robotics and nonlinear control systems.

MARK W. SPONG (S'81–M'81–SM'89–F'96) received the B.A. degree (magna cum laude and Phi Beta Kappa) in mathematics and physics from Hiram College, Hiram, OH, USA, in 1975, the M.S. degree in mathematics from New Mexico State University, Las Cruces, NM, USA, in 1977, and the M.S. and D.Sc. degrees in systems science and mathematics from Washington University in St. Louis, MO, USA, in 1979 and 1981, respectively.

He was with the University of Illinois at Urbana-Champaign, Urbana, IL, USA, from 1984 to 2008. He is currently the Dean of the Erik Jonsson School of Engineering and Computer Science, University of Texas at Dallas, TX, USA, where he is a Professor of electrical engineering and holds both the Lars Magnus Ericsson Chair and the Excellence in Education Chair. He has authored or co-authored more than 250 technical papers in control and robotics and four books, and holds one patent. His current research interests include robotics, mechatronics, and nonlinear control theory.

Dr. Spong was the President of the IEEE Control Systems Society and was both the Editor-in-Chief and an Associate Editor of the IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY and an Associate Editor of the IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION, the IEEE Control Systems Magazine, and the IEEE TRANSACTIONS ON AUTOMATIC CONTROL. He was the Vice President for Publication Activities from 2000 to 2002. He was the recipient of several awards, including the first Intelligent Robots and Systems Fumio Harashima Award for Innovative Technologies, the IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY Outstanding Paper Award, the Senior Scientist Research Award from the Alexander von Humboldt Foundation, the Distinguished Member Award from the IEEE Control Systems Society, the John R. Ragazzini and O. Hugo Schuck Awards from the American Automatic Control Council, and the IEEE Third Millennium Medal.