Master of Science Thesis

Nonlinear Geometric Control of a Quadrotor with a Cable-Suspended Load

N.N. Vo

August 1, 2017

Delft Center for Systems and Control

Delft University of Technology

Nonlinear Geometric Control of a Quadrotor with a Cable-Suspended Load

Master of Science Thesis

For obtaining the degree of Master of Science in Mechanical Engineering at Delft University of Technology

N.N. Vo

August 1, 2017

Faculty of Mechanical, Maritime and Materials Engineering · Delft University of Technology

The work in this Master of Science Thesis was supported by Alten Nederland. Their cooperation is hereby gratefully acknowledged.

Delft University of Technology

Delft University Of Technology
Delft Center for Systems and Control

The undersigned hereby certify that they have read and recommend to the Faculty of Mechanical, Maritime and Materials Engineering for acceptance a thesis entitled **"Nonlinear Geometric Control of a Quadrotor with a Cable-Suspended Load"** by **N.N. Vo** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: August 1, 2017

Supervisor TU Delft: ____________________

dr.ir. T. Keviczky

Readers: ____________________

Dr. J. Alonso-Mora

____

A. Sharifi Kolarijani

Supervisor Alten: ____________________

ir. B. van Vliet

Abstract

A Quadrotor is a type of Unmanned Aerial Vehicle that has received an increasing amount of attention recently with many applications including search and rescue, surveillance, supply of food and medicines as disaster relief and object manipulation in construction and transportation. An interesting subproblem of load transportation, is the control of the position of a cable suspended load. The challenge is in the fact that the Quadrotor-Load system is highly nonlinear and under-actuated. The load cannot be controlled directly and has a natural swing at the end of each Quadrotor movement.

This thesis presents a Nonlinear Geometric Control approach for the position tracking of a cable suspended load. The focus lies on the subsystem of the total Quadrotor-Load system where the cable tension is non-zero, which is analogous to a system with a rigid link between the Quadrotor and Load.

First, an introduction is given on Geometric Mechanics, an approach that applies differential geometric techniques to systems modeling and control, based on the geometric properties of the dynamics of the system. It is shown how the configuration of the Quadrotor-Load system can be described on smooth nonlinear geometric configuration spaces. Analyzing these geometric structures with the principles of differential geometry allows modeling in an unambiguous coordinate-free dynamic fashion, while avoiding the problem of singularities that would occur on local charts. The geometric properties are utilized to define tracking error functions on these same spaces, making it possible to design almost globally defined controllers.

The main goal of this thesis is to investigate the possibilities and limitations of Nonlinear Geometric Control for the purpose of load trajectory tracking of a Quadrotor with a cable-suspended Load, by evaluating the stability of the system and the tracking performance on different load trajectories. The Quadrotor-Load system is modeled with the tools of differential geometry in order to make it suitable for Nonlinear Geometric Control. A backstepping approach is applied to generate a cascaded structure with multiple nonlinear Geometric controllers, allowing control of several flight modes that are responsible for the control of 1) Quadrotor attitude, 2) Load attitude and 3) Load position. A Linear Quadratic Regulator is derived to compare control performance. Simulations are generated to demonstrate the stability of the closed-loop system, and the tracking performance of both linear and nonlinear controllers are discussed.

Acknowledgements

I would like to thank my supervisors dr.ir. T. Keviczky from *Delft Center of Systems and Control*, and ir. B. van Vliet from *Alten Nederland B.V.* for their assistance during my research and the writing of this thesis. I would also like to thank all colleagues from Alten and TU Delft for their time and advice.

Delft, University of Technology
August 1, 2017

N.N. Vo

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

A Quadrotor (QR) is a type of Unmanned Aerial Vehicle (UAV) that has received an increasing amount of attention recently with many applications being actively investigated. Possible applications include search and rescue, surveillance, reliable supply of food and medicines as disaster relief and object manipulation in construction and transportation. It has already proven itself useful for many tasks like multi-agent missions, mapping, explorations, transportation and entertainment such as acrobatic performances.

The inspiration for this research is build upon the idea of creating a system of multiple autonomous QRs for a cooperative towing task. The advantage of such systems for object manipulation is the increased reach and the possibility to reduce complexity of the individual robot, decrease cost over traditional robotic systems and high reliability. One can think of examples in nature, where individuals coordinate, cooperate and collaborate to perform tasks that they individually can not accomplish. Redundancy makes development of fail safe control methods possible and can extend the capabilities of a single robot.

Considering a multi-agent task, one can think of multiple QRs assisting in the transportation of a common load. This cooperation can be executed in many ways, but this research focuses on QRs with a cable-suspended load in motion. The suspended object naturally continues to swing at the end of every movement. In case a residual motion can result in damage or in order to avoid obstacles and path following, an accurate positioning is required. Reducing the oscillation, or controlling the position of the suspended load might be necessary, but is challenging in the fact that this cable-suspended system is under-actuated. Possible objectives are minimizing the oscillations of the load during or after motion, minimizing the time to position the load, trajectory tracking, trajectory generation and obstacle avoidance.

1-1 Aim and Motivation

The aim is to control the position of a suspended load using a QR. Before considering multiple QRs, it is important to investigate the possibilities of a single QR with load system. Hence, in this research a single QR is considered for the transportation of a cable suspended load, which will exert additional forces and torques on the QR. This is a challenging control problem in the fact that the QR system is under-actuated. Adding a suspended load will add extra DOFs and oscillations of the load occur at the end of every movement.

The system can be divided into two subsystems. The first subsystem is where the cable tension is non-zero and the distance between the QR and the load is defined by the cable length. Both QR and load are coupled as one system. The second subsystem is where the cable tension is zero, such that the QR and load in free fall are two separate decoupled systems. This research focuses on the first subsystem, such that the cable tension is non-zero. In order to control both subsystems, hybrid control must be applied, which is considered out of the scope of this research.

Former work on attitude control of QR and/or load often relies on linear control methods such as PID, MPC and LQR control. The dynamics are linearized around an equilibrium point, describing the system dynamics by a set of linear differential equations. The control of a QR-Load system is a very specific case and scarcely investigated. Former work includes MPC [2] and LQR control approaches, where an optimal control strategy is used to minimize the swing of the load.

The reason that linear control near an equilibrium state is commonly applied, is partly to avoid difficulties that come with modeling and controlling the non-linearities of the system. However, linear control limits the system to small angle movements, as the optimization will not allow large angles that deviate to far from the linearized point. This type of modeling and control will not be sufficient for applications that require fast aggressive maneuvers. Nonlinear control systems are often governed by nonlinear differential equations and are able to represent the dynamics in a more realistic manner. Nonlinear control approaches to minimize the load swing includes a Model Based Algorithm controller applied by [3].

****

Bart: I think you might need some references here.
Nam: Duidelijk: is nog ToDo

****

Nonlinear Geometric Control is a nonlinear model based control technique based on a modeling approach involving the concepts of differential geometry. This results in a globally defined coordinate-free dynamical model, while preventing issues regarding singularities, and enabling the design of controllers that offer almost-global convergence properties.

Former work includes a nonlinear geometric control of a QR [4, 5] and nonlinear geometric control of the load position, load attitude and QR attitude of a QR-Load system [6, 7, 8]. Nonlinear Geometric Control for QR systems is rarely found in literature, despite the advantageous properties of differential geometry.

This motivates to investigate the potential and limitations of a rarely used nonlinear Geometric Control approach, and to investigate the performance of a load transportation maneuver, when compared to a commonly used linear control strategy.

Different aspects involving the modeling and control for the QR-Load system must be investigated, for it can be expected that the non-linearity will have a great influence in the representation of the dynamics and the stability, accuracy and type of the control design. It is possible to investigate which advantages or disadvantages this nonlinear approach has compared to a linear approach, in terms of stability and performance.

1-2 Organization of the Report

In this first chapter, a brief introduction of the subject is given and the problem is described. This is followed by discussing the aim, motivation and contributions of this thesis for this research. The organization of the report is as follows.

In Chapter 2 the dynamics of the QR-Load system is described by the laws of kinematics and the application of Newton's laws or Lagrangian mechanics. Geometric Mechanics is used to understand and derive the system's equations of motion in order to allow nonlinear geometric controller design and analysis. The system configuration space is described on a differentiable manifold using the tools of Differential geometry, instead of using the tools of Euclidean geometry, where the system dynamics evolve in a three dimensional space. In contrast with classical modeling techniques, geometric modeling results in a compact, unambiguous and coordinate-free model.

Describing the system dynamics on nonlinear manifolds allows the design of nonlinear geometric controllers on these same manifolds. The control design is presented in Chapter 3. The controller has a cascaded structure, allowing the control of several flight modes that are accountable for the control of different degrees of freedom.

Chapter **??** describes the experiments that are done to investigate the abilities and performance of a nonlinear Geometric Control design. Different tracking objectives are defined in order to compare the performance between an LQR control design and a nonlinear Geometric Control design. The results are presented and findings are discussed.

In the final chapter a summary of the thesis is given, followed by the conclusions that were made based on the results of the research. Finally, recommendations are given which could serve as an starting point for future work.

Chapter 2

Dynamic Model

A mathematical model of the system needs to be derived in order to simulate and study the effects of nonlinear Geometric Control. In Section 2-1, an introduction is given about Geometric Mechanics. This is a modern description of the classical mechanics from the perspective of Differential Geometry, which is a discipline in mathematics that studies manifolds and their geometric properties, using the tools of calculus.

The assumptions that are applied to simplify the model are shortly discussed in Section 2-2. Next, in Section 2-4 a dynamical model of the QR-Load system is obtained with Geometric Mechanics, resulting in a compact, coordinate-free, unambiguous representation of the dynamics, described on nonlinear manifolds.

2-1 Geometric Mechanics

For the derivation of the equations of motions, traditional modeling methods often parameterize the rotations in a local coordinate system. This can be done with Euler Angles, and despite this parametrization might result in singularities, this is a commonly used method to describe rotations. There are 24 possible sets of Euler angles and many different conventions are used, which introduces ambiguity. The definition of Euler angles is not unique and a sequence of rotations is not commutative. Therefore, Euler angles are never expressed in terms of the external frame, or in terms of the co-moving rotated body frame, but in a mixture.

An other disadvantage of Euler angles, is that the transformation from their time rates of change to the angular velocity vector is not globally defined. Furthermore, when angular errors are large, the difference in Euler angles is no longer a good metric to define the orientation error. Hence, the error is rather written as the required rotation to get from the current to a desired orientation, which can be achieved by considering geometric properties of the system.

In Geometric Mechanics the configuration space of systems is a *group manifold* instead of a Euclidean space. The kinetic and potential energies are expressed in terms of this configuration space and their tangent spaces. It explores the geometric structure of a Lagrangian-or Hamiltonian system through the concepts of vector calculus, linear algebra, differential geometry, and nonlinear control theory. Geometric mechanics provides fundamental insights into the nonlinear system mechanics and yields useful tools for dynamics and control theory.

An example is given of a simple 2-link arm, to illustrate different representations of a configuration space, see Figure 2-1. Let the configuration of the arm be defined by two coordinates in a Cartesian coordinate system, which is a local representation. This can be seen in Figure 2-1b, where the colored edges illustrate singularities, because the definition of one point has multiple solutions.
Next, the configuration space is represented as a geometric shape called a *torus*, as shown in Figure 2-1c. It is a smooth manifold where each configuration is mapped uniquely, which allows the configuration to be defined globally.

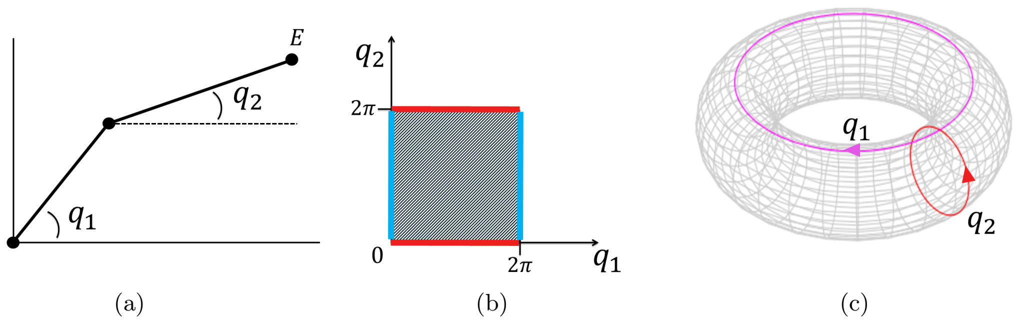


Figure 2-1: Configuration Space of a 2-link arm

Manifolds The fundamental object of differential geometry a manifold. A manifold is a mathematical space, a collection of points, that locally resembles Euclidean space near each point. Examples are a plane, a ball, a torus and a sphere. Manifolds are important objects in mathematics and physics because they allow more complicated structures to be expressed and understood in terms of the relatively well-understood properties of simpler spaces. Each point of an n-dimensional manifold has a neighborhood that is homeomorphic to the n-dimensional Euclidean space, meaning that there is a continuous function describing the relation between these spaces, illustrated in Figure 2-2.

****

Bart: Link coordinate free to singularity
Nam: ToDo

****

A *differentiable manifold* is a smooth and continuous manifold and is locally similar enough to a linear space to allow to do calculus. One can define directions, tangent spaces, and differentiable functions on such a manifold.
Taking the derivative at a point on a manifold is equivalent to a *tangent vector* at that point. Meaning that derivatives are conceptually equivalent to an infinitesimally short tangent vector. Each point of an n-dimensional differentiable manifold has a tangent space, which is

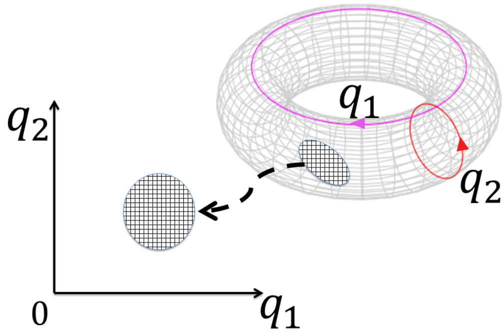
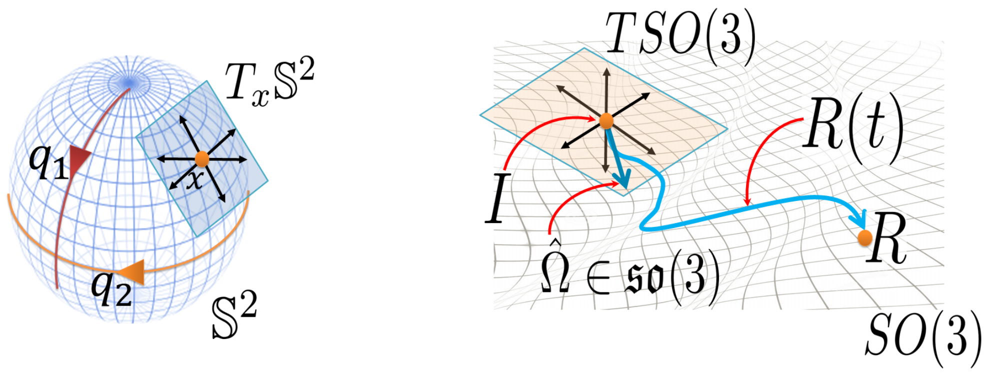


Figure 2-2: A manifold locally resembles a Euclidean space

an n-dimensional Euclidean space consisting of all the tangent vectors of *all* curves that pass through that point.
A tangent space describes a relationship between a position and a velocity at that position. This property is of importance for the determination of configuration error functions, which give a measure of the error between a desired state and an actual state. The configuration errors will be described in Section 3-1.
To illustrate a tangent space, a point x is chosen on a *2-sphere*, which is a manifold denoted by $\mathbb{S}^2$ and defined by a sphere of dimension 2. The tangent space at point x is the collection of all tangent vectors at point x and is denoted by $T_x\mathbb{S}^2$, see Figure 2-3a.



(a) Representation of a manifold with a tangent space

(b) Identity map of manifold $SO(3)$ with Lie Algebra $\mathfrak{so}(3)$

Figure 2-3: Tangent Spaces on different manifolds

Geometric Configuration Spaces Several methods exist to describe rotations, such as *Euler Angles*, quaternions or rotation matrices. The main disadvantages of Euler angles are that some functions have singularities and they are a less accurate measure for the integration of incremental changes in attitude over time, compared to other methods. To avoid these problems, in Geometric Mechanics rotations are expressed as rotation matrices to provide a global representation of the attitude of a rigid body.

The QR attitude is expressed as a rotation matrix R in the Special Orthogonal Group $SO(3)$, which describes the rotation of a body frame relative to the spatial frame. The manifold $SO(3)$ is defined as

$$SO(3) \triangleq \left\{R \in \mathbb{R}^{3\times 3} | RR^T = I_{3\times 3}, det(R) = 1\right\} \tag{2-1}$$

where $SO(3)$ is the group of all rotations about the origin of a 3-D Euclidean space, which preserves the origin, Euclidean distance and orientation. [9, 10]

Every rotation has a unique inverse rotation and the identity map satisfies the definition of a rotation. The elements of *Lie Algebra* $\mathfrak{so}(3)$, a property associated with $SO(3)$, are the elements of the *tangent space* of $SO(3)$ at the identity element, see Figure 2-3b. These elements define the relation between the rotation R and its derivative $\dot{R}$, such that

$$\dot{R} = R\hat{\Omega} \tag{2-2}$$

For $n \in \mathbb{N}$, $\mathfrak{so}(n)$ is is the vector space of skew-symmetric matrices in $\mathbb{R}^{n\times n}$ and defined as

$$\mathfrak{so}(n) \triangleq \left\{S \in \mathbb{R}^{n\times n} | S^T = -S\right\} \tag{2-3}$$

The hat map $\wedge : \mathbb{R}^3 \to \mathfrak{so}(3)$ is an isomorphism between $\mathbb{R}^3$ and the set of 3×3 skew symmetric matrices, such that $\hat{x}y = x \times y$ for any $x, y \in \mathbb{R}^3$. The vee map $\vee : \mathfrak{so}(3) \to \mathbb{R}^3$, and is the inverse isomorphism of the hat map. Several properties of the hat map are

$$\hat{x}y = x \times y = -y \times x = -\hat{y}x, \tag{2-4}$$
$$tr[A\hat{x}] = \frac{1}{2}tr[\hat{x}(A - A^T)] = -x^T(A - A^T)^\vee, \tag{2-5}$$
$$\hat{x}A + A^T\hat{x} = (\{tr[A]I_{3\times 3} - A\}x)^\wedge, \tag{2-6}$$
$$R\hat{x}R^T = (Rx)^\wedge, \tag{2-7}$$

for any $x, y \in \mathbb{R}^3$, $A \in \mathbb{R}^{3\times 3}$, and $R \in SO(3)$. The mapping between the body angular velocity vector $\Omega \in \mathbb{R}^3$ and $\hat{\Omega} \in \mathfrak{so}(3)$ is written as

$$\hat{\Omega} = \begin{bmatrix} 0 & -\Omega_3 & \Omega_2 \\ \Omega_3 & 0 & -\Omega_1 \\ -\Omega_2 & \Omega_1 & 0 \end{bmatrix}, \quad \begin{bmatrix} 0 & -\Omega_3 & \Omega_2 \\ \Omega_3 & 0 & -\Omega_1 \\ -\Omega_2 & \Omega_1 & 0 \end{bmatrix}^\vee = \Omega \tag{2-8}$$

The load attitude is expressed as a unit vector q, which points from $\{\mathcal{B}\}$ to the load. The configuration space is a *two-sphere* $\mathbb{S}^2$ defined as

$$\mathbb{S}^2 \triangleq \left\{q \in \mathbb{R}^3 | q \cdot q = 1\right\} \tag{2-9}$$

The plane tangent to the sphere at q is the tangent space

$$T_q\mathbb{S}^2 \simeq \left\{\omega \in \mathbb{R}^3 | q \cdot \omega = 0\right\} \tag{2-10}$$

where ω is the angular velocity of the suspended load.

2-2 Modeling Assumptions

Table 2-1 shows the assumptions that are used for modeling the QR-Load system, simplifying the complexity of the model.

Modeling assumptions Quadrotor model
• The structure of the QR is rigid and symmetric. Elastic deformations and shock (sudden accelerations) of the QR are ignored. • The mass distribution of the QR is symmetrical in the x-y plane. • The inertia matrix is time-invariant. • Aerodynamic effects acting on the QR are neglected. Blade flapping, Turbulence, Ground Effects. • The air density ρ around the QR is constant. • The propellers are rigid $\Rightarrow$ The thrust produced by rotor i is parallel to the axis of rotor i. • Drag factor d and thrust factor b are approximated by a constant. Thrust force F_i and moment M_i of each propeller is proportional to the square of the propeller speed.
Modeling assumptions Quadrotor-Load model
• The cable is modeled as a rigid and massless cable. • The cable is connected to a friction-less joint at the origin of the body-fixed. • The tension in the cable is considered to be non-zero. This implies that the QR-Load subsystem that consists of a QR and a Load in free fall, is disregarded. • Aerodynamic effects acting on the load are neglected.

Table 2-1: Modeling assumptions

2-3 Quadrotor Model

The QR model representation is shown in Figure 2-4. Two Cartesian coordinate frames are defined:

- The body-fixed reference frame $\{\mathcal{B}\}$ (Body Frame)

  with unit vectors $\{\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3\}$ along the axes
- The ground-fixed reference frame $\{\mathcal{I}\}$ (Inertial Frame)

  with unit vectors $\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$ along the axes

such that $\{\mathcal{I}\}$ is fixed to earth and the axis of $\{\mathcal{B}\}$, $\mathbf{b}_1$ and $\mathbf{b}_2$ coincide with the arms of the QR.

The QR is described as a rigid body with six degrees of freedom, driven by the system inputs: the total upward force f and the moments $M = \begin{bmatrix} M_\phi & M_\theta & M_\psi \end{bmatrix}^T$ around the body axes. The configuration of the QR can be described by 1) the location of the QR's Center of

Mass (com), $x_Q \in \mathbb{R}^3$, described in the Euclidean space w.r.t. $\{\mathcal{I}\}$, and 2) the *attitude* which is the orientation of $\{\mathcal{B}\}$ w.r.t. $\{\mathcal{I}\}$ evolving on a nonlinear space, described by a rotation matrix $R \in SO(3)$. The dynamics of a rigid body can be expressed on the manifold $SE(3)$, which is the group of *rigid displacements* in $\mathbb{R}^3$. A rigid displacement describes both the rotation and the position of $\{\mathcal{B}\}$ relative to $\{\mathcal{I}\}$.

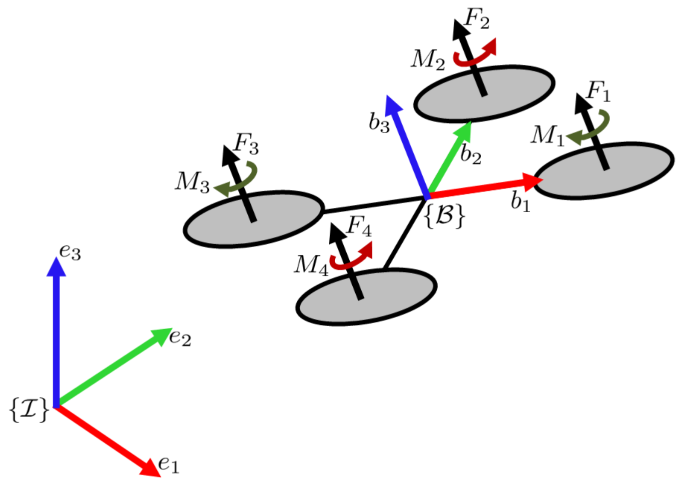


Figure 2-4: Quadrotor model representation

Rotor dynamics The complex dynamics of the rotors and their interactions with drag and thrust forces are represented by a simplified model. The angular speed ω_i of rotor i, for $i = 1, 2, 3, 4$, generates a force F_i parallel to the direction of the rotor axis of rotor i, given by

$$F_i = \left(\frac{K_v K_\tau \sqrt{2\rho A}}{K_t} \omega_i \right)^2 \simeq b\omega_i^2 \tag{2-11}$$

where K_v, K_t are constants related to the motor properties, ρ is the density of the surrounding air, A is the area swept out by the rotor, K_τ is a constant determined by the blade configuration and parameters, and b is the thrust factor.
The torque around the axis of rotor i, generated due to drag is given by

$$M_i = \frac{1}{2} R\rho C_D A(\omega_i R)^2 \simeq d\omega_i^2 \tag{2-12}$$

where R is the radius of the propeller, C_D is a dimensionless constant, and d is the drag constant.

The required rotor speeds ω_i can be calculated for a given desired total thrust f and total moment $M = \begin{bmatrix} M_\phi & M_\theta & M_\psi \end{bmatrix}^T$, by solving the following equation

$$\begin{bmatrix} f \\ M_\phi \\ M_\theta \\ M_\psi \end{bmatrix} = \begin{bmatrix} b & b & b & b \\ 0 & -lb & 0 & lb \\ lb & 0 & -lb & 0 \\ -d & d & -d & d \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} \tag{2-13}$$

where l is the distance from the rotor to the QR's com, d is the drag factor and b is the thrust factor.

2-4 Quadrotor-Load Model

The total Quadrotor-Load model consists of two subsystems, 1) where the cable tension is zero, and 2) where the cable tension is non-zero. In this research, the focus is only on the subsystem where the cable tension is non-zero. The QR-Load model is shown in Figure 2-5. The unit vector $q \in \mathbb{S}^2$ gives the direction from the QR to the Load expressed in $\{\mathcal{B}\}$. The position of the QR and Load are related by

$$x_Q = x_L - Lq \tag{2-14}$$

where $x_Q \in \mathbb{R}^3$ is the position of the QR's com expressed in $\{\mathcal{I}\}$, $x_L \in \mathbb{R}^3$ is the position of the load expressed in $\{\mathcal{I}\}$, and L is the length of the cable.

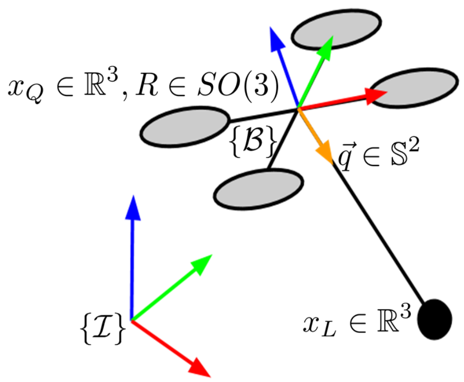


Figure 2-5: Quadrotor with Load model representation

The configuration of the load can be described by its location $x_L \in \mathbb{R}^3$ w.r.t. $\{\mathcal{I}\}$, evolving in Euclidean space, and the load attitude evolving on a nonlinear space $\mathbb{S}^2$, described by the unit vector $q \in \mathbb{S}^2$.
For a study on rigid body dynamics and optimal control problems, where geometric features are incorporated, one can refer to [11].

Euler-Lagrange To develop the Euler-Lagrange equations for mechanical systems that evolve on manifolds, an approach developed by [11, 12, 13, 14] is applied. The basic idea is to express the variations of the curves evolving on $\mathbb{S}^2$ and $SO(3)$. This approach is based on Hamilton's principle, which states that the evolution of a physical system is a solution of the functional equation given by

$$\frac{\delta S}{\delta \mathbf{x}(t)} = 0 \tag{2-15}$$

where $\mathbf{x}$ defines the configuration space. S is the action integral, defined as

$$S = \int_{t_1}^{t_2} \mathcal{L} dt \tag{2-16}$$

where $\mathcal{L} = \mathcal{T} - \mathcal{U}$ is the Lagrangian of the system, and $\mathcal{T}$,$\mathcal{U}$ are the kinetic and potential energy, respectively.

Hamilton's principle of least action states that the path a conservative mechanical system takes between two states $\mathbf{x}_1$ and $\mathbf{x}_2$ at time t_1 and t_2, is the one for which Equation 2-16 is a stationary point, resulting in

$$\delta S = \int_{t_1}^{t_2} \delta \mathcal{L} dt = 0 \tag{2-17}$$

where $\delta\mathcal{L}$ is the variation of the Lagrangian. For systems with non-conservative forces and moments, Equation 2-17 is extended to

$$\delta S = \int_{t_1}^{t_2} (\delta W + \delta \mathcal{L}) dt = 0 \tag{2-18}$$

where δW is the virtual work. Equation 2-18 is applied to the QR-Load system, where the configuration manifold is $\mathbb{R}^3 \times \mathbb{S}^2 \times SO(3)$. With the following states

$$\mathbf{x} = \begin{bmatrix} x_L & \dot{x}_L & q & \omega & R & \Omega \end{bmatrix}^T \tag{2-19}$$

where ω is the angular velocity of the load and Ω denotes the angular velocity of the body-fixed frame.

The kinetic energy $\mathcal{T}$ and the potential energy $\mathcal{U}$ for the system are denoted as

$$\begin{aligned} \mathcal{T} &= \frac{1}{2} m_Q \dot{x}_Q \cdot \dot{x}_Q + \frac{1}{2} m_L \dot{x}_L \cdot \dot{x}_L + \frac{1}{2} \Omega \cdot J \cdot \Omega \\ \mathcal{U} &= m_Q g x_Q \cdot e_3 + m_L g x_L \cdot e_3 \end{aligned} \tag{2-20}$$

where $J \in \mathbb{R}^{3\times 3}$ is the inertia tensor of the QR, and g is the gravity constant.
The energy can be rewritten in terms of q and x_L, by substituting Equation 2-14, giving

$$\mathcal{T} = \frac{1}{2}(m_Q + m_L)\dot{x}_L \cdot \dot{x}_L - m_Q L \dot{x}_L \cdot \dot{q} + \frac{1}{2} m_Q L^2 \dot{q} \cdot \dot{q} + \frac{1}{2} \Omega \cdot J \cdot \Omega \tag{2-21}$$

$$\mathcal{U} = (m_Q + m_L) g x_L \cdot e_3 - m_Q g L q \cdot e_3 \tag{2-22}$$

Variations The variations of $\mathcal{T}$ and $\mathcal{U}$ are approximated by a first-order Taylor approximation, which results in

$$\begin{aligned} \delta\mathcal{T} &\approx \frac{\partial \mathcal{T}}{\partial \dot{x}_L}\delta \dot{x}_L + \frac{\partial \mathcal{T}}{\partial \dot{q}}\delta \dot{q} + \frac{\partial \mathcal{T}}{\partial \Omega}\delta\Omega \\ &= ((m_Q + m_L)\dot{x}_L - m_Q L \dot{q}) \cdot \delta \dot{x}_L + (-m_Q L \dot{x}_L + m_Q L^2 \dot{q}) \cdot \delta \dot{q} + (J\Omega) \cdot \delta\Omega \\ \delta\mathcal{U} &\approx \frac{\partial \mathcal{U}}{\partial x_L}\delta x_L + \frac{\partial \mathcal{U}}{\partial q}\delta q \\ &= ((m_Q + m_L) g e_3) \cdot \delta x_L - (m_Q g L e_3) \cdot \delta q \end{aligned} \tag{2-23}$$

The first term of virtual work is obtained from f acting on the QR and is given by the following term,

$$\begin{aligned} \delta W_1 &= f R e_3 \cdot \sum_{j=1}^{3} \frac{\partial x_Q}{\partial \mathbf{q}_j} \delta \mathbf{q}_j \\ &= f R e_3 \cdot (\delta x_L - L \delta q) \end{aligned} \tag{2-24}$$

where $\mathbf{q}_j = x_L, q, R$ and x_Q is substituted by Equation 2-14. The second term of virtual work is obtained from M acting on the QR. This gives the following term

$$\begin{aligned} \delta W_2 &= M \cdot \sum_{j=1}^{3} \frac{\partial \Omega}{\partial \dot{\mathbf{q}}_j} \delta \dot{\mathbf{q}}_j \\ &= M \cdot (R^T \delta R) \end{aligned} \tag{2-25}$$

The variations in energy and the virtual work can be substituted into Equation 2-18, such that

$$\delta S = \int_{t_1}^{t_2} (\delta W_1 + \delta W_2 + \delta \mathcal{T} - \delta \mathcal{U}) dt \tag{2-26}$$

While $x_L, \dot{x}_L$ vary on $\mathbb{R}^3$, Equation 2-26 is also a function of variations on manifolds, where δR is a variation on $SO(3)$ and δq is a variation on $\mathbb{S}^2$. These so called infinitesimal variations define how the curves on the manifold "vary", and are obtained as shown in [1, 15, 12, 14, 16].

$$\begin{aligned} \delta R = R\hat{\eta} \in T_R SO(3), \text{ where } \eta \in \mathbb{R}^3, \hat{\eta} \in \mathfrak{so}(3) \\ \delta q = \xi \times q \in T_q \mathbb{S}^2, \text{ where } \xi \in \mathbb{R}^3, \xi \cdot q = 0 \end{aligned} \tag{2-27}$$

The following variations follow from differentiation,

$$\begin{aligned} \delta \dot{q} &= \dot{\xi} \times q + \xi \times \dot{q}, \\ \delta \dot{R} &= \dot{R}\hat{\eta} + R\dot{\hat{\eta}}, \\ \delta \hat{\Omega} &= \delta(R^T \dot{R}) \\ &= \delta R^T \dot{R} + R^T \delta \dot{R} \\ &= (R\hat{\eta})^T \dot{R} + R^T (\dot{R}\hat{\eta} + R\dot{\hat{\eta}}) \\ &= \hat{\eta}^T \hat{\Omega} + \hat{\Omega}\hat{\eta} + \dot{\hat{\eta}} \\ &= (\hat{\Omega}\eta)^\wedge + \dot{\hat{\eta}}, \\ \delta \Omega &= (\hat{\Omega}\eta) + \dot{\eta} \end{aligned} \tag{2-28}$$

These variations are substituted into Equation 2-26, allowing it to be a function of variations in each generalized coordinate.

$$\begin{aligned} \delta S &= \int_{t_1}^{t_2} (\delta W_1 + \delta W_2 + \delta \mathcal{T} - \delta \mathcal{U}) dt \\ &= \int_{t_1}^{t_2} (((m_Q + m_L)\dot{x}_L - m_Q L \dot{q}) \cdot \delta \dot{x}_L + (fRe_3 - (m_Q + m_L) g e_3) \cdot \delta x_L) dt \\ &+ \int_{t_1}^{t_2} ((m_Q L^2 \dot{q} - m_Q L \dot{x}_L) \cdot \delta \dot{q} + (m_Q g L e_3 - fLRe_3) \cdot \delta q) dt \\ &+ \int_{t_1}^{t_2} (\Omega^T J \cdot \delta \Omega + M \cdot (R^T \delta R)) dt \end{aligned} \tag{2-29}$$

After rearranging and setting each variation to 0, the following equations of motion for the

QR-Load system are found.

$$\frac{d}{dt}x_L = \dot{x}_L \tag{2-30}$$
$$(m_Q + m_L)(\ddot{x}_L + ge_3) = (q \cdot fRe_3 - m_Q L(\dot{q} \cdot \dot{q}))q \tag{2-31}$$
$$\dot{q} = \omega \times q \tag{2-32}$$
$$m_Q L\dot{\omega} = -q \times fRe_3 \tag{2-33}$$
$$\dot{R} = R\hat{\Omega} \tag{2-34}$$
$$J\dot{\Omega} + \Omega \times J\Omega = M \tag{2-35}$$

where Equations 2-31 and 2-33 are the load position and attitude dynamics, and Equation 2-35 represents the QR attitude dynamics.

The dynamics of the complete QR-Load system can be globally expressed on the Special Orthogonal Group $SO(3)$, *two-sphere* $\mathbb{S}^2$ and Special Euclidean Group $SE(3)$, which are all smooth manifolds. This results in a compact notation of the equations of motion, making the large amount of trigonometric functions unnecessary, that normally are introduced by Euler angles.

Summary

In this chapter, the dynamical model of the Quadrotor-Load system was derived. The motivation to use Geometric Mechanics and a basic understanding of its concepts are given in order to understand the difference between a Nonlinear Geometric model and a model obtained with classical modeling approaches.

With the tools of differential geometry, the system dynamics are expressed on nonlinear configuration manifolds, which results in a globally defined, compact, unambiguous representation of the model. This dynamical model is used for a nonlinear geometric control approach, which is discussed in the next chapter.

Chapter 3

Control Design

Section 3-1 introduces Nonlinear Geometric Control and concepts of geometric properties that are used for analysis and control design. In the previous chapter, the configuration spaces of the system dynamics were expressed on nonlinear manifolds,. In Section 3-1-1, error functions and geometric mappings are defined on these same nonlinear manifolds in order to measure the error between current and desired states.

In order to deal with the under-actuated nature of the system, a backstepping control approach is applied to stabilize the system . This control design allows multiple controllers to operate in a cascaded structure, which results in the possibility to track a load position, while stabilizing the system. The control design with its different flight modes and the corresponding controllers, are discussed in Section 3-2.

3-1 Nonlinear Geometric Control

Many control systems are developed for the standard form of ordinary differential equations

$$\dot{x} = f(x, u) \tag{3-1}$$

where x is the state and u the control input. It is assumed that the state and the control input lie in Euclidean spaces, and the system equations are defined in terms of smooth functions between Euclidean spaces. However, for many mechanical systems, the configuration space can only be expressed locally as a Euclidean space. A nonlinear space is required to express the configuration space globally, which is discussed in the previous chapter.

Geometric Control Theory is the study on how geometry of the state space influences control problems. In control systems engineering, the underlying geometric features of a dynamic system are often not considered carefully. Differential Geometric control techniques utilize these geometric properties for control system design and analysis. The objective is to express both the system dynamics and control inputs on nonlinear manifolds instead of local charts. In contrast to locally defined linear control, nonlinear geometric control can be defined almost globally, avoiding singularities that would occur in the representation of large angles and complex maneuvering.

The design of the controllers for the QR attitude can be found in [4], and the controllers of load attitude- and position this can be found in [1]. Thorough stability analyses are presented in these references. For a deeper understanding of Lyapunov stability analysis in geometric control, the reader can refer to [16]. Other control systems that are able to switch between control modes, such as the hybrid control described in [17], require complicated reachability set analysis to guarantee safe switching between different flight modes. Nonlinear Geometric Control does not requires such analysis, as the region of attraction for each flight mode covers the configuration space almost globally. A study on global nonlinear dynamics of various classes of closed loop attitude control systems can be found in [18].

****

Bart: Stability analysis goed in relatie gebracht met de scope van jouw thesis. Hier heb je wat aan en weet je wat je te wachten staat als je hybrid gaat toevoegen in further research.
Nam: Het gaat eigenlijk over hybrid control zonder differential geometry. Aangepast.

****

****

Bart: Bedoel je hier This is in contrast to ....

Met andere woorden volgens mij klopt die zin niet helemaal toch?

En waarom haal je hier aan dat hybrid control geen complicated analysis nodig heeft?
Nam: Doel is om duidelijk te maken dat die hybrid controller -zonder geometric control- júist wel complicated analysis nodig heeft, om te switchen naar andere flight modes. En de Geometric Controller niet. Zin veranderd: duidelijker?

****

3-1-1 Error Functions

The control of a trajectory tracking problem requires state feedback to define tracking errors, a measure of the difference between the current states and the desired states. Since the closed-loop system dynamics evolve on nonlinear manifolds, which describe the configuration space of the QR attitude $\in SO(3)$ and the load attitude $\in \mathbb{S}^2$, error functions are defined on these same manifolds [16]. These functions play a role in the definition of the potential function for the closed-loop system and form the basis for both stabilizing and tracking controllers of the QR-Load system.

Quadrotor Attitude Error

Recall that R is the rotation matrix to describe the QR attitude, and R_d is the desired rotation matrix. To describe the relative rotation from the body frame to the desired frame, an *attitude error* is defined as $R_d^T R$. Note that $R_d^T R$ is again a rotation matrix itself. Based on this attitude error, the *tracking error function* Ψ_R on $SO(3)$ is chosen to be

$$\Psi_R(R, R_d) = \frac{1}{2} tr \left[I - R_d^T R \right] \tag{3-2}$$

such that Ψ_R is locally positive-definite about $R_d^T R = I$ within the region where the rotation angle between R and R_d is less than 180°. It can be shown that this region where $\Psi_R < 2$ almost covers $SO(3)$ [19].
Using Equation 2-5 and 2-27, the derivative of the tracking error function Ψ_R with respect to R along the direction of $\delta R = R\hat{\eta}$ for $\eta \in \mathbb{R}^3$ is given by

$$\begin{aligned} \mathbf{D}_R \Psi(R, R_d) \cdot R\hat{\eta} &= -\frac{1}{2} tr[R_d^T R \hat{\eta}] \\ &= \frac{1}{2}(R_d^T R - R^T R_d)^\vee \cdot \eta \end{aligned} \tag{3-3}$$

where the *vee map* $^\vee : \mathfrak{so}(3) \to \mathbb{R}^3$ is the inverse of the *hat map* defined in Section 2-1. From this equation, the QR *attitude tracking error* $e_R \in \mathbb{R}^3$ is chosen as follows

$$e_R = \frac{1}{2}(R_d^T R - R^T R_d)^\vee \tag{3-4}$$

It is important to note that the velocities $\dot{R}$ and $\dot{R}_d$ cannot be compared directly, since they do not lie in the same space. At time $t = t_0$, assume that $R(t_0) = q$ and $R_d(t_0) = r$, then $\dot{R}$ and $\dot{R}_d$ lie in their own tangent spaces, denoted by $T_q SO(3)$ and $T_r SO(3)$, respectively. For this reason, $\dot{R}_d$ must be transformed into a vector on $T_q SO(3)$ to allow a meaningful comparison with $\dot{R}$. Defining a velocity error can be achieved with a mathematical object called a *transport map*, which enables the comparison of tangent vectors living in different spaces.
In Figure 3-1, two curves $R(t)$ and $R_d(t)$ evolve on manifold $SO(3)$. Transport map $\mathcal{T}(q, r)$: $T_r SO(3) \mapsto T_q SO(3)$ allows comparison of the velocity curves $\dot{R}$ and $\dot{R}_d$.

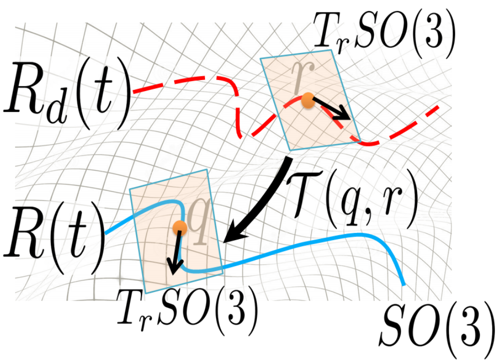


Figure 3-1: Transport map $\mathcal{T}(q, r)$

The *velocity error* $\dot{e}$ is a vector field along R corresponding to the transport map. It defines the velocity error between the curves R and R_d, and is defined as

$$\dot{e} = \dot{R} - \dot{R}_d(R_d^T R) \tag{3-5}$$

This equation is rewritten to obtain the angular velocity tracking error, as follows

$$\begin{aligned} \dot{R} - \dot{R}_d(R_d^T R) &= R\hat{\Omega} - R_d\hat{\Omega}_d(R_d^T R) \\ &= R(\Omega)^\wedge - (RR^T)R_d\hat{\Omega}_d R_d^T R \\ &= R(\Omega)^\wedge - R(R^T R_d \Omega_d)^\wedge \\ &= R(\Omega - R^T R_d \Omega_d)^\wedge \end{aligned} \tag{3-6}$$

The angular *velocity tracking error* e_Ω expressed in $\{\mathcal{B}\}$ is defined as

$$e_\Omega = \Omega - R^T R_d \Omega_d \tag{3-7}$$

Similar to the form of Equation 2-34, e_Ω represents the angular velocity vector of the relative rotation matrix $R_d^T R$, represented in $\{\mathcal{B}\}$. Hence, it can be shown that the following equation holds

$$\frac{d}{dt}(R_d^T R) = (R_d^T R)\hat{e}_\Omega \tag{3-8}$$

****

Bart: Wat is het verschil tussen de velocity error between curves R en Rd en angular velocity tracking error?

Nam: velocity error $\dot{e}$ is velocity error tussen $\dot{R}$ en $\dot{R}_d$. Angular velocity error e_Ω is de angular velocity vector van $R_d^T R$, dus verschil tussen Ω en Ω_d.

****

The values of the QR attitude tracking error e_R and the QR angular velocity tracking error e_Ω are used later on to design control for the QR attitude.

Load Attitude Error

The load attitude dynamics evolve on $\mathbb{S}^2$ and its tangent space $T\mathbb{S}^2$, where the error of the load attitude is described in a similar approach. The error between the load attitude q and the desired load attitude q_d is defined by the error function $q_d^T q$. Based on the error function, the tracking error function Ψ_q on $\mathbb{S}^2$ is chosen to be

$$\Psi_q = 1 - q_d^T q \tag{3-9}$$

The derivative of the tracking error function Ψ_q is given by

$$d_1\Psi_q(q, q_d) = \hat{q}^2 q_d \tag{3-10}$$

From this equation, the load attitude error function e_q is defined as follows

$$e_q = \hat{q}^2 q_d \tag{3-11}$$

Again, a *transport map* is used for a comparison between the tangent vectors on different tangent spaces. Using the tracking error Ψ_q and the transport map $\mathcal{T}_{\mathbb{S}^2}$, a closed-loop energy function evolving on $\mathbb{S}^2$ is derived [16, 11.3.2]. From this energy function, the load angular velocity error function is defined as

$$e_{\dot{q}} = \dot{q} - (q_d \times \dot{q}_d) \times q \tag{3-12}$$

The values of the load attitude tracking error e_q and the load angular velocity tracking error $e_{\dot{q}}$ are used later on to design control for the load attitude.

Load Position Error

The tracking errors for the load position and load velocity are defined as

$$e_x = x - x_d \tag{3-13}$$

$$e_v = v - v_d \tag{3-14}$$

where $v_d = \dot{x}_d$. Furthermore, $x_d(t) \in \mathbb{R}^3$ must be a smooth twice-differentiable load trajectory, such that functions are well defined. The values of the position error e_x and the velocity error e_v are used later on to design control for the load position.

3-2 Backstepping Control

A backstepping approach will be used in this research for the control of the load trajectory tracking problem, and is commonly used QR control[20]. Backstepping control is a Lyapunov based control technique for the stabilization of nonlinear dynamical systems developed by [21]. The method relies on a *triangular structure* of the system in a certain set of coordinates. The system is split into subsystems in a cascaded structure and recursive techniques allow a systematically design of feedback control laws and corresponding Lyapunov functions.
The control structure is created by starting with a stable system as the most inner subsystem. By "stepping back" from this subsystem, a control loop can be added around it containing a control law that defines a change of coordinates, which allows control of a new state, while still stabilizing the inner structure. The control law is designed by using states as virtual control inputs, such that each loop computes a virtual command signal for the adjacent inner loop. This is repeated until the final external control is reached.
The backstepping approach determines how to stabilize the QR with the control inputs f and M, while several controllers are able to track different states, see Figure 3-2. The inner controller determines what the required control inputs are, driven by R_c. The next controller calculates how to drive the computed rotation matrix R_c based on q_c, such that the QR is stabilized. And the last controller determines which load attitude q_c is required, such that the desired load position $x_{L,d}$ is tracked.

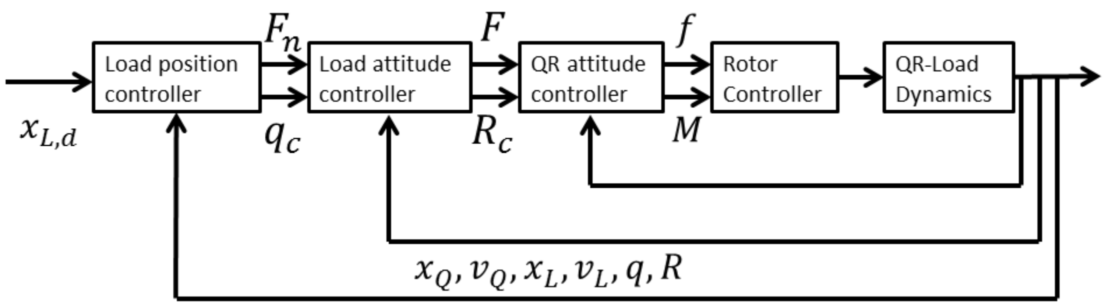


Figure 3-2: Nonlinear Geometric Control Loop of the QR-Load system [1]

Since the QR has only four actuators, it is not possible to control all DOFs of the QR-Load system simultaneously. The backstepping approach allows control of different flight modes in which a combination of DOFs are controlled. The flight modes and their functions are defined below in order, from the most inner loop to the most outer loop.

- QR Attitude Controlled Mode
  - Track a desired QR attitude $R_d(t)$ or commanded signal $R_c(t)$. Optional tracking of a desired heading direction $b_{1_d}(t)$, the first column of $R_d(t)$.
  - Calculate the control input M for the QR-Load system
- Load Attitude Controlled Mode
  - Track a desired load attitude $q_d(t)$ or commanded signal $q_c(t)$
  - Calculate a computed QR attitude R_c for the QR attitude controller

  - Calculate the control input f for the QR-Load system

- Load Position Controlled Mode
  - Track a desired load position $x_{L,d}(t)$
  - Calculate a computed load attitude q_c for the load attitude controller
  - Calculate the control input f for the QR-Load system

where the subscript d denotes a desired tracking reference, and the subscript c denotes a computed tracking reference, calculated by the controllers. More detailed derivations of the equations in the following sections can be found in Section A-2.

3-2-1 Quadrotor Attitude Tracking

The QR Attitude Controlled Mode is designed to control the QR attitude by tracking a smooth desired QR attitude $R_d(t)$. Analysis of the error dynamics e_R and e_Ω requires the calculation of their time derivatives.
Using Equations 3-4 and 3-7, the derivative of the attitude tracking error e_R can be written as

$$\dot{e}_R = \frac{1}{2}(R_d^T R \hat{e}_\Omega + \hat{e}_\Omega R^T R_d)^\vee \tag{3-15}$$

The derivative of the angular velocity tracking error e_Ω, follows from Equations 2-34, 3-7 and the fact that $\hat{\Omega}_d \Omega_d = 0$, such that

$$\dot{e}_\Omega = \dot{\Omega} + \hat{\Omega} R^T R_d \Omega_d - R^T R_d \dot{\Omega}_d \tag{3-16}$$

Recall from Equation 2-34, that the kinematics equation for the desired attitude can be written as

$$\dot{R}_d = R_d \hat{\Omega}_d \text{ and } \hat{\Omega}_d = R_d^T \dot{R}_d \tag{3-17}$$

From this follows that the desired angular acceleration $\dot{\Omega}_d$ can then be defined as follows

$$\begin{aligned} \dot{\hat{\Omega}}_d &= (\dot{R}_d^T \dot{R}_d) + (R_d^T \ddot{R}_d) \\ &= (R_d \hat{\Omega}_d)^T (R_d \hat{\Omega}_d) + (R_d^T \ddot{R}_d) \\ &= -\hat{\Omega}_d \hat{\Omega}_d + R_d^T \ddot{R}_d, \\ \dot{\Omega}_d &= (-\hat{\Omega}_d \hat{\Omega}_d + R_d^T \ddot{R}_d)^\vee \end{aligned} \tag{3-18}$$

By substituting Equation 2-35 into Equation 3-16, the following is obtained

$$\dot{e}_\Omega = J^{-1}(-\Omega \times J\Omega + M) + \hat{\Omega} R^T R_d \Omega_d - R^T R_d \dot{\Omega}_d \tag{3-19}$$

From this, the control input M is defined [19], and consists of a proportional term, a derivative term and a canceling term, as follows

$$M = -k_R e_R - k_\Omega e_\Omega + \Omega \times J\Omega - J(\hat{\Omega} R^T R_d \Omega_d - R^T R_d \dot{\Omega}_d) \tag{3-20}$$

Rapid exponential convergence of the attitude error function and angular velocity error function can be achieved by adding the parameter ϵ to Equation 3-20, where $0 < \epsilon < 1$, as done in [1]

$$M = -\frac{1}{\epsilon^2}k_R e_R - \frac{1}{\epsilon}k_\Omega e_\Omega + \Omega \times J\Omega - J(\hat{\Omega}R^T R_d \Omega_d - R^T R_d \dot{\Omega}_d) \tag{3-21}$$

Substituting Equation 3-21 into Equation 3-19 results in

$$\dot{e}_\Omega = J^{-1}(-\frac{1}{\epsilon^2}k_R e_R - \frac{1}{\epsilon}k_\Omega e_\Omega) \tag{3-22}$$

for any positive constants k_R, k_Ω.
Equation 2-6 is used to rewrite the time derivative of e_R as follows

$$\begin{aligned} \dot{e}_R &= \frac{1}{2}(R_d^T R \hat{e}_\Omega + \hat{e}_\Omega R^T R_d)^\vee \\ &= \frac{1}{2}(tr[R^T R_d]I - R^T R_d)e_\Omega \equiv C(R_d^T R)e_\Omega \end{aligned} \tag{3-23}$$

where $\| C(R_d^T R) \|_2 \leq 1$, such that $\| \dot{e}_R \| \leq \| e_\Omega \|$ for all $R_d^T R \in SO(3)$, guaranteeing that $\dot{e}_R$ will be bounded if e_Ω is. Equations 3-22 and 3-23 are used in a stability analysis of the controller, and it is proven in [4] that the zero equilibrium of the closed loop tracking error $(e_R, e_\Omega) = (0, 0)$ is exponentially stable, if the initial conditions satisfy

$$\Psi_R(R(0), R_d(0)) < 2 \tag{3-24}$$

$$\| e_\Omega(0) \|^2 < \frac{2}{\lambda_M(J)} \frac{k_R}{\epsilon^2}(2 - \Psi_R(R(0), R_d(0))) \tag{3-25}$$

where $\lambda_M(\cdot)$ denotes the maximum eigenvalue.
Furthermore, there exist constants $\alpha_R, \beta_R > 0$ such that

$$\Psi_R(R(t), R_d(t)) \leq min\left\{2, \alpha_R e^{-\beta_R t}\right\} \tag{3-26}$$

Equations 3-24 and 3-25 determine the domain of attraction, which is the region in which the trajectory of the system is able to converge to an asymptotically stable equilibrium point. The domain of attraction almost covers $SO(3)$, this is referred to as almost-global exponential attractiveness.

Note that the tracking of the QR attitude does not require any specification of the thrust magnitude f. During this flight mode, the translational motion can only be controlled partially, which makes this flight mode suitable for attitude maneuvers with short time periods.

3-2-2 Load Attitude Tracking

The Load Attitude Controlled Mode is designed to track a desired load attitude q_d. In order to influence the load dynamics, see Equation 2-33, the load attitude controller calculates a computed QR attitude R_c for the QR attitude controller, such that R_d is replaced by R_c, where R_c is defined as

$$R_c = \left[b_{1c}; b_{3c} \times b_{1c}; b_{3c}\right] \tag{3-27}$$

And Ω_d is replaced by Ω_c, where Ω_c is defined by

$$\hat{\Omega}_c = R_c^T \dot{R}_c \tag{3-28}$$

which will influence the QR attitude dynamics, see Equation 2-35.
The unit vector b_{1c} is the first column of R_c and is constructed by normalizing the projection of a desired heading angle $b_{1d} \in \mathbb{S}^2$ onto the plane normal to b_{3c}, see Figure 3-3. Defining

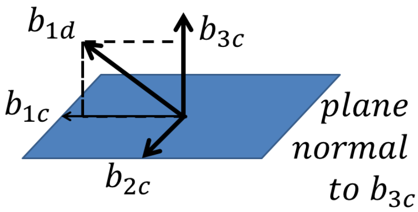


Figure 3-3: Representation of R_c

$b_{1c} \in \mathbb{S}^2$ orthogonal to b_{3c} guarantees that $R_c \in SO(3)$ [19]. This is defined as

$$b_{1c} = -\frac{1}{||b_{3c} \times b_{1d}||}(b_{3c} \times (b_{3c} \times b_{1d})) \tag{3-29}$$

such that b_{1d} is chosen, not parallel to b_{3c}.
$b_{3c} \in \mathbb{S}^2$ is the third column of R_c and is defined by a normalization of F,

$$b_{3c} = \frac{F}{||F||} \tag{3-30}$$

where F is defined by a normal component F_n, a proportional-derivative component F_{pd} and feedforward control force F_{ff}

$$F = F_n - F_{pd} - F_{ff} \tag{3-31}$$

The inclusion of F_n ensures that b_{3c} is always well defined. F_n is defined as

$$F_n = -(q_d \cdot q)q \tag{3-32}$$

The control forces F_{pd} and F_{ff} are defined for trajectory tracking in [16, 11.2.5] as follows

$$\begin{aligned} F_{pd} &= -k_P \hat{q}^2 q_d - k_D(\dot{q} - (q_d \times \dot{q}_d) \times q) \\ &= -k_q e_q - k_\omega e_{\dot{q}} \end{aligned} \tag{3-33}$$

for positive constants k_q, k_ω.

$$F_{ff} = m_Q L \langle\langle q, q_d \times \dot{q}_d \rangle\rangle_{\mathbb{R}^3} (q \times \dot{q}) + m_Q L (q_d \times \ddot{q}_d) \times q \tag{3-34}$$

This controller affects the control input f, which is defined as

$$f = F \cdot Re_3 \tag{3-35}$$

and the control input M will be defined as

$$M = -\frac{1}{\epsilon^2} k_R e_R - \frac{1}{\epsilon} k_\Omega e_\Omega + \Omega \times J\Omega - J(\hat{\Omega} R^T R_c \Omega_c - R^T R_c \dot{\Omega}_c) \tag{3-36}$$

It is proven in [1] and [16, Lemma 11.23] that the zero equilibrium of the closed loop tracking error $(e_q, e_{\dot{q}}, e_R, e_\Omega) = (0,0,0,0)$ is exponentially stable, if the initial conditions satisfy

$$\Psi_q(q(0), q_d(0)) < 2 \tag{3-37}$$

$$\| e_{\dot{q}}(0) \|^2 < \frac{2}{m_Q L} k_R (2 - \Psi_q(q(0), q_d(0))) \tag{3-38}$$

The domain of attraction is defined by Equations 3-24, 3-25, 3-37 and 3-38. Equation 3-37 states that the initial load attitude error should be less than 180°, which means that the controller achieves almost-global exponential convergence for load attitude q. Furthermore, there exist constants $\alpha_q, \beta_q > 0$ such that

$$\Psi_q(q(t), q_d(t)) \leq min \left\{2, \alpha_q e^{-\beta_q t}\right\} \tag{3-39}$$

3-2-3 Load Position Tracking

The Load Position Controlled Mode is designed to track a desired load position $x_{L,d}$. Analysis of the error dynamics e_x and e_v requires the calculation of their time derivatives.
The derivative of the load position error e_x is given by

$$\dot{e}_x = e_v \tag{3-40}$$

and from Equation 2-31 and given that $\dot{e}_v = \ddot{x}_L - \ddot{x}_{L,d}$ follows

$$(m_Q + m_L)\dot{e}_v = -(m_Q + m_L)(ge_3 + \ddot{x}_{L,d}) - m_Q L(\dot{q} \cdot \dot{q})q + (q \cdot fRe_3)q \tag{3-41}$$

Equations 3-40 and 3-40 are used in a stability analysis of the controller. The load position controller calculates a computed load attitude q_c for the load attitude controller. R_d and q_d are replaced by R_c and q_c, respectively. In order to stabilize the error dynamics, it is proven in [1] that the required computed load attitude is defined as

$$q_c = -\frac{A}{||A||} \tag{3-42}$$

where

$$A = -k_x e_x - k_v e_v + (m_Q + m_L)(\ddot{x}_{L,d} + ge_3) + m_Q L(\dot{q} \cdot \dot{q})q \tag{3-43}$$

Furthermore, Equation 3-32 is redefined as

$$F_n = (A \cdot q)q \tag{3-44}$$

which is substituted in Equation 3-31, resulting in a new control input f.

This controller ensures that the zero equilibrium of the closed loop tracking error $(e_x, e_v, e_q, e_{\dot{q}}, e_R, e_\Omega) = (0,0,0,0,0,0)$ is exponentially attractive, if the initial conditions satisfy

$$\Psi_q(q(0), q_c(0)) < \psi_1 < 1 \tag{3-45}$$

$$\| e_x(0) \|^2 < e_{x_{max}} \tag{3-46}$$

where $e_{x_{max}}$ and ψ_1 are fixed design depended constants.

The domain of attraction is defined by Equations 3-24, 3-25, 3-45 and the following equation

$$\| e_{\dot{q}}(0) \|^2 < \frac{2}{m_Q L} k_q(\psi_1 - \Psi_q(q(0), q_d(0))) \tag{3-47}$$

Summary

In this chapter, control design based on Nonlinear Geometric Control is discussed. What is particular in this control technique, is the fact that error functions are defined on non-Euclidean manifolds, similar to the manifolds that describe the configuration space of the system. Since these manifolds are locally Euclidean, local stability properties of a closed-loop equilibrium solution can be determined by using standard Lyapunov methods. Based on these error functions, controllers are designed in a backstepping approach, enabling both load position tracking and stabilization of the system. Using the geometric properties of the system allows the design of globally defined controllers that ensure almost-global convergence of the QR attitude and load attitude. In order to test the control performance of a load position tracking objective, experiments are defined in the next chapter.

Chapter 4

Experiment

The experimental procedure is explained in Section 4-1. It is discussed what experiments can be done in order to investigate the tracking performance of Nonlinear Geometric Control. In addition, a comparison will be made between the performances of the Nonlinear Geometric Controller and a linear LQR controller.

The controllers are tested on their ability to track a desired load trajectory. Section 4-2 presents desired trajectories that create situations with different challenges, and it is discussed what could be expected from these experiments.

In Section 4-3 the experimental setup is discussed. The model parameters for the QR-Load system are presented, as well as the controller parameters for both Nonlinear Geometric controller and LQR controller. The notion of a backstepping command filter is made to explain a mathematical simplification in the experiments.

Finally, in Section 4-4 the results that are obtained from the load trajectory tracking experiments are presented and discussed. The stability of the closed-loop system is demonstrated for the Nonlinear Geometric Controller and the differences in linear- and nonlinear controller performance are discussed.

4-1 Procedure

Performance of both Nonlinear Geometric Control and LQR control can be evaluated by comparing their ability to track a load trajectory with minimal error. In linear control however, a linearized model is obtained by assuming small angles of both load and QR around an equilibrium point. The model is obtained by assuming the system in equilibrium when the QR is in hover position with the load hanging directly underneath it. As a result, the linearized model does not allow direct reference tracking of the load position.

The LQR cost function allows control of the inputs f and M, and the states which define the QR position, QR attitude and load attitude. Therefore, full load position control is not possible. It can be approached by QR position control or minimization of the load swing.

This limitation illustrates an important difference between the use of a linear and a nonlinear model.

The experiments describe a smooth desired load trajectory $x_{L,d}(t)$ in order to get well-defined control functions. In this work the desired load paths are generated by hand. The associated required velocity and acceleration are calculated by a *command filter*, which is explained in more detail in Section 4-3.

The experiments done with the LQR controller will apply reference tracking of the QR position, which is based on the desired load trajectories that are used for the nonlinear Geometric controller. When assuming small angles and minimal load swing, the QR position should be a cable length above the predefined desired load position. Note that this will not allow a direct comparison of the load trajectory tracking performance, nevertheless this will illustrate important differences between the controllers. For the purpose of load transportation, both controllers can be used. The difference is in the approach of the problem.

4-2 Trajectories

This section discusses a number of cases that describe different load trajectories for the QR-Load system. A description is given of desired trajectory and the challenges that are involved.

The stability of the closed-loop system is investigated by observing whether the controller is able to bring the system to a stationary final state. The behavior of the error dynamics is observed through the error functions, as described in Chapter 3.

Case A

In this first case a smooth step-like trajectory is In a regular step function the system is subjected to a sudden input.

The step response is a common analysis tool used to obtain information about the stability of a dynamical system. A step function can be used to investigate the effects of a sudden input to the desired load position. However, since the desired load trajectory is required to be twice-differentiable, a smooth step-like function is generated. The goal is to transport the load from a starting position along the direction of the x-axis to a final position. It can be investigated whether the system responds with overshoot or undershoot, whether the desired steady-state is reached and the needed settling time.

Figure 4-1 shows the desired trajectory over time, and a three dimensional representation.

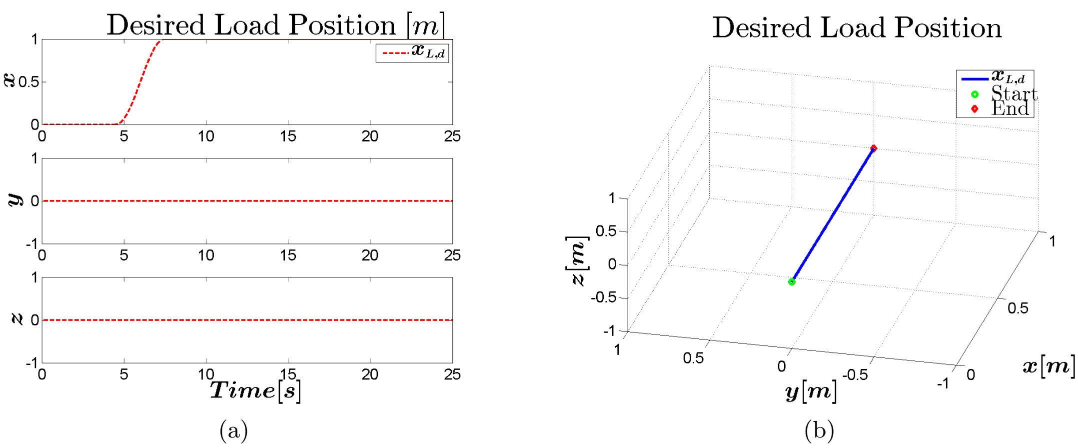
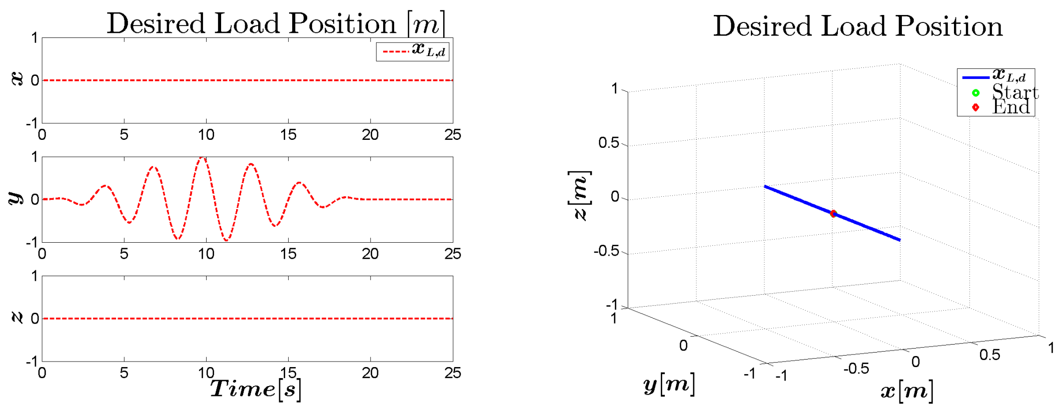


Figure 4-1: Desired Load Position Case A

Case B

The next trajectory is described by a sine wave in one direction which grows in amplitude over time. The load pos It can be expected that the QR attitude

Figure 4-2 shows the desired trajectory over time, and a three dimensional representation.



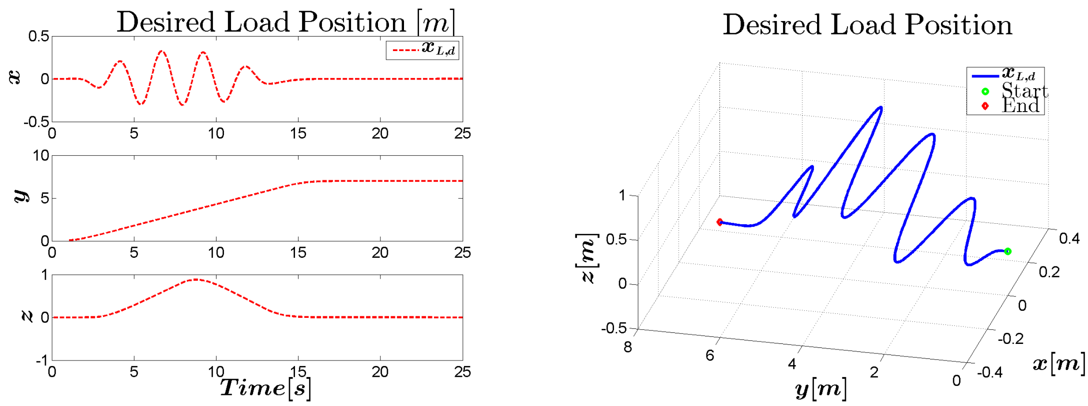
(a) (b)

Figure 4-2: Desired Load Position Case B

Case C

For this case a trajectory is generated to test multiple disciplines. The trajectory has the shape of a sine wave that moves along the y-axis and varies in amplitude in the direction of the x-axis, while going up and down in the direction of the z-axis. The changing amplitude of the trajectory that moves from side to side, requires varying velocities to 'keep up' with the trajectory. It can be expected that the nonlinear geometric control allows large QR angles, whereas the LQR will possible fail to deviate far from the equilibrium point.

Figure 4-3 shows the desired trajectory over time, and a three dimensional representation.



(a) (b)

Figure 4-3: Desired Load Position Case C

4-3 Setup

Model parameters The simulations are developed using Matlab and Simulink, using the system parameters found in Table 4-1.

Parameter	Value		Description
m_Q	4.34	kg	Quadrotor Mass
m_L	0.1	kg	Load Mass
l	0.315	m	Arm length from QR com to rotor
L	0.7	m	Cable Length
I_{xx}	0.0820	kgm^2	Quadrotor Inertia about x-axis
I_{yy}	0.0845	kgm^2	Quadrotor Inertia about y-axis
I_{zz}	0.1377	kgm^2	Quadrotor Inertia about z-axis

Table 4-1: Modeling Parameters

Geometric Control The chosen controller gains in Equations 3-20,3-33,3-43 can be found in Table 4-2.

Gain	Case A	Case B	Case C
k_R			
k_Ω			
k_q			
k_ω			
k_x			
k_v			

Table 4-2: Controller Gains Nonlinear Geometric Controller

LQR Control Linear Quadratic Regulator (LQR) control uses an algorithm to obtain a state-feedback controller, minimizing a cost function depending on the states and weight factors. An LQR design is shown in Figure 4-4

Figure 4-4: LQR control design

LQR control is based on a small angle assumption. Therefore, a traditional modeling method may represent the rotation matrix with a local coordinate system, for example with an Euler Angle parameterization. A continuous time linearized model of the system used in this controller is represented in the following form

$$\dot{\mathbf{x}} = A\mathbf{x} + Bu \tag{4-1}$$
$$y = C\mathbf{x} + Du \tag{4-2}$$

where $\mathbf{x}$ is the state vector and u is the input vector, defined as follows

$$\mathbf{x} = \begin{bmatrix} x & y & z & \phi & \theta & \psi & \phi_L & \theta_L & \dot{x} & \dot{y} & \dot{z} & \dot{\phi} & \dot{\theta} & \dot{\psi} & \dot{\phi}_L & \dot{\theta}_L \end{bmatrix}^T \tag{4-3}$$
$$u = \begin{bmatrix} f & M_\phi & M_\theta & M_\psi \end{bmatrix}^T \tag{4-4}$$

where ϕ_L and θ_L are the angle of rotation of the load about the x-axis and y-axis in $\{\mathcal{B}\}$, respectively. The derivation of A, B, C, D can be found in Section A-3.

Using `Matlab` command `lqr(A,B,Q,R)`, an optimal gain matrix K is calculated, such that the state-feedback law $u = -K\mathbf{x}$ minimizes the quadratic cost function defined as

$$J(u) = \int_0^\infty (\mathbf{x}^T Q\mathbf{x} + u^T Ru)dt \tag{4-5}$$

The weight matrices Q and R that define the effects of the states and inputs in the cost function, and the calculated gain matrix K can be found in Section A-3.

Command Filtering A consequence of a backstepping control approach, is that it also increases the order of the states. The inner control loops become a function of the commanded signals and their higher derivatives, which are generated by the controllers in outer loops. In the control design, discussed in Chapter 3, the load attitude controller generates a commanded QR attitude R_c and its derivative $\dot{R}_c$. In the same fashion, the load position controller generates a commanded load attitude q_c and its derivative $\dot{q}_c$. Instead of analytic differentiation of these terms, which can be tedious and could result in high computational costs, these values can be obtained with the use of a Command Filter, which is explained in more detail in [22].

The basic idea is that the command signal is pre-filtered by a low pass filter and generates an estimation of the derivatives of the commanded signal. In this thesis a backstepping command filter of third order is applied to compute $\dot{R}_c, \ddot{R}_c, \dot{q}_c, \ddot{q}_c$. The transfer function of the original commanded input signal X_c^o and the filtered output X_c has the form

$$\frac{X_c(s)}{X_c^o(s)} = H(s) = \frac{\omega_{n1}}{s + \omega_{n1}} \cdot \frac{\omega_{n2}^2}{s^2 + 2\zeta\omega_{n2}s + \omega_{n2}^2} \tag{4-6}$$

Where ζ is the damping ratio and ω_n the undamped natural frequency. The filter has the following state space representation

$$\dot{x}_1 = x_2 \tag{4-7}$$
$$\dot{x}_2 = x_3 \tag{4-8}$$
$$\dot{x}_3 = -(2\zeta\omega_{n2} + \omega_{n1})x_3 - (2\zeta\omega_{n1}\omega_{n2} + \omega_{n2}^2)x_2 - (\omega_{n1}\omega_{n2}^2)(x_1 - x_c^o) \tag{4-9}$$

where $x_1 = x_c$, $x_2 = \dot{x}_c$ and $x_3 = \ddot{x}_c$, such that x_c. The command filter parameters that were chosen, are shown in Table 4-3.

Parameter	R	q
ω_{n1}		
ω_{n2}		
ζ		

Table 4-3: Command Filter Parameters

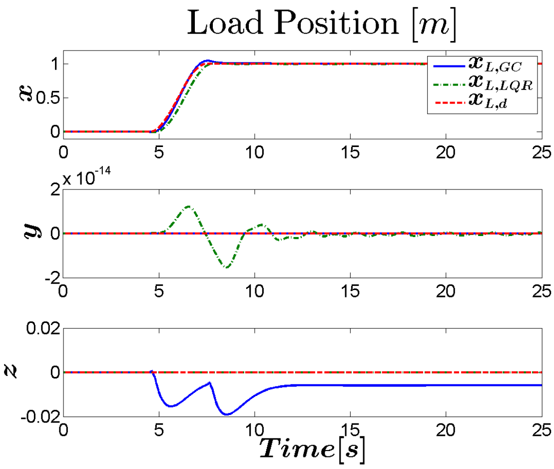
4-4 Results

In Figures 4-6, 4-8 and 4-10 the load tracking performance is shown for the Nonlinear Geometric Controller. From these figures the tracking errors of the QR attitude, load attitude and load position, and the stability of the tracking errors can be analyzed.

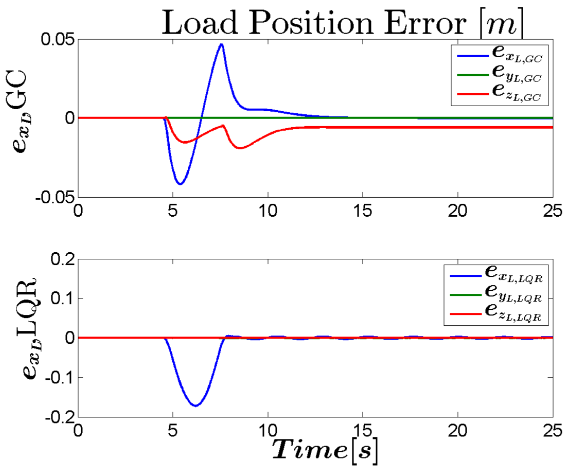
4-4-1 Case A

Figure 4-5a shows the load position along the desired load position $x_{L,d}$ for both control approaches.
Figure 4-5b shows the load position error for both control approaches.

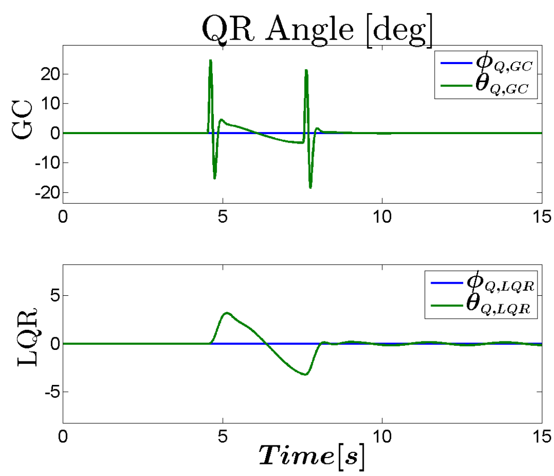
Figure 4-5c shows the QR attitude with respect to $\{\mathcal{I}\}$.
In Figure 4-5d the load angle with respect to $\{\mathcal{B}\}$ is shown.
Observations:



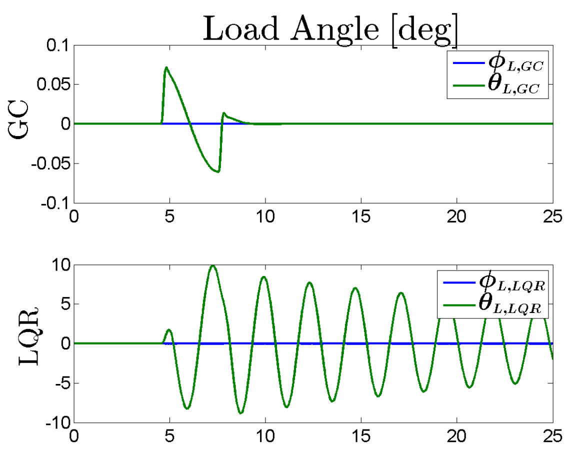
(a) Load tracking



(b) Load Position Error



(c) QR Attitude



(d) Load Attitude

Figure 4-5: Controller Comparison Case A

The desired and actual load trajectory, and the position error are shown in Figure 4-6a and Figure 4-6b, respectively. From this can be seen that a small steady state error remains in the z-direction. However, $(e_x, e_v) = (0, 0)$ is exponentially attractive.

Figure 4-6c and 4-6d show the tracking errors of the QR attitude and load attitude, respectively.
Observations: $(e_x, e_v, e_q, e_{\dot{q}}, e_R, e_\Omega) = (0, 0, 0, 0, 0, 0)$ is exponentially stable

Figure 4-6e and 4-6f show the tracking error functions of the QR and load, respectively. Observations: there exist constants $\alpha_q, \beta_q > 0$ such that

$$\Psi_q(q(t), q_d(t)) \leq min\left\{2, \alpha_q e^{-\beta_q t}\right\} \tag{4-10}$$

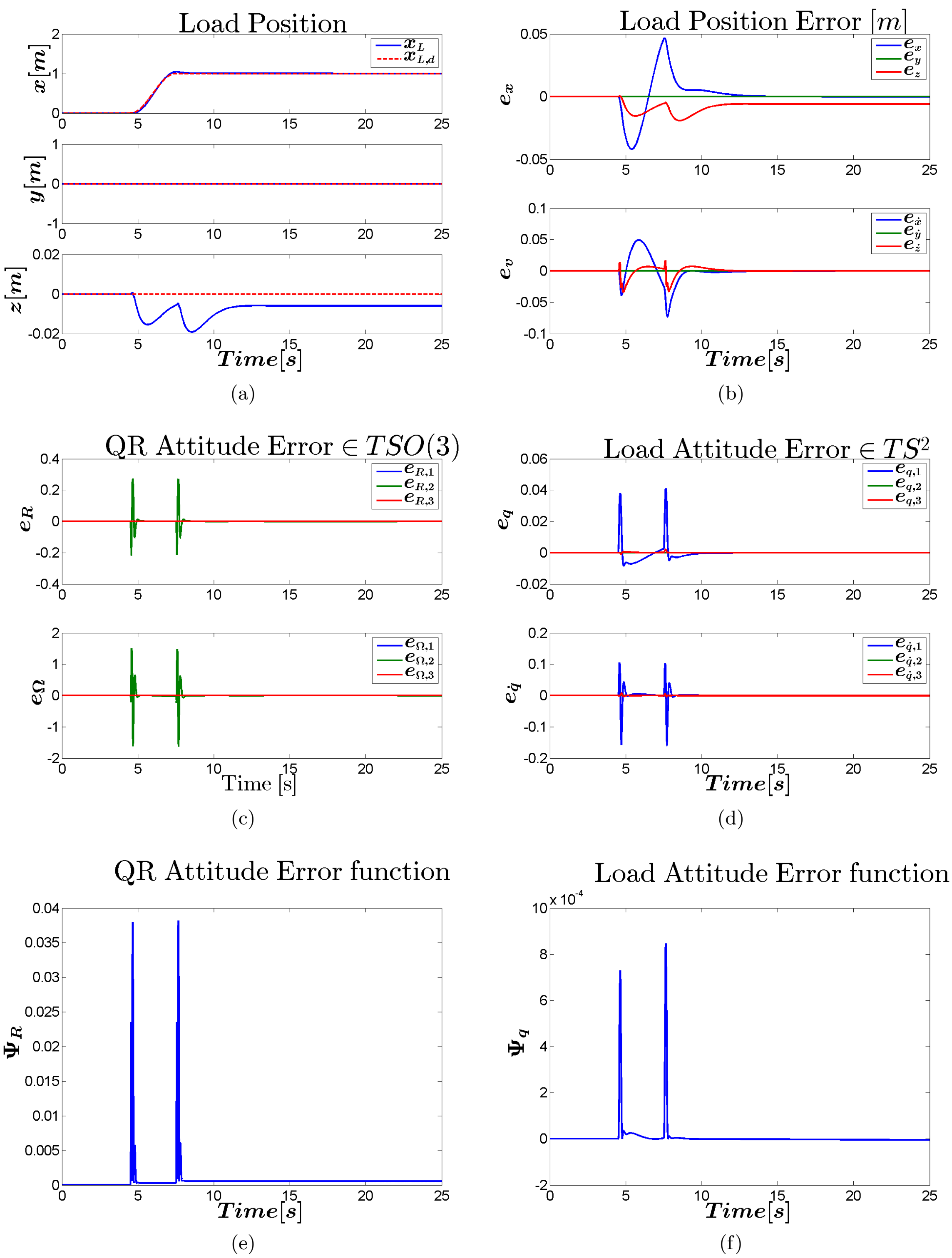


Figure 4-6: Results Nonlinear Geometric Control Case A

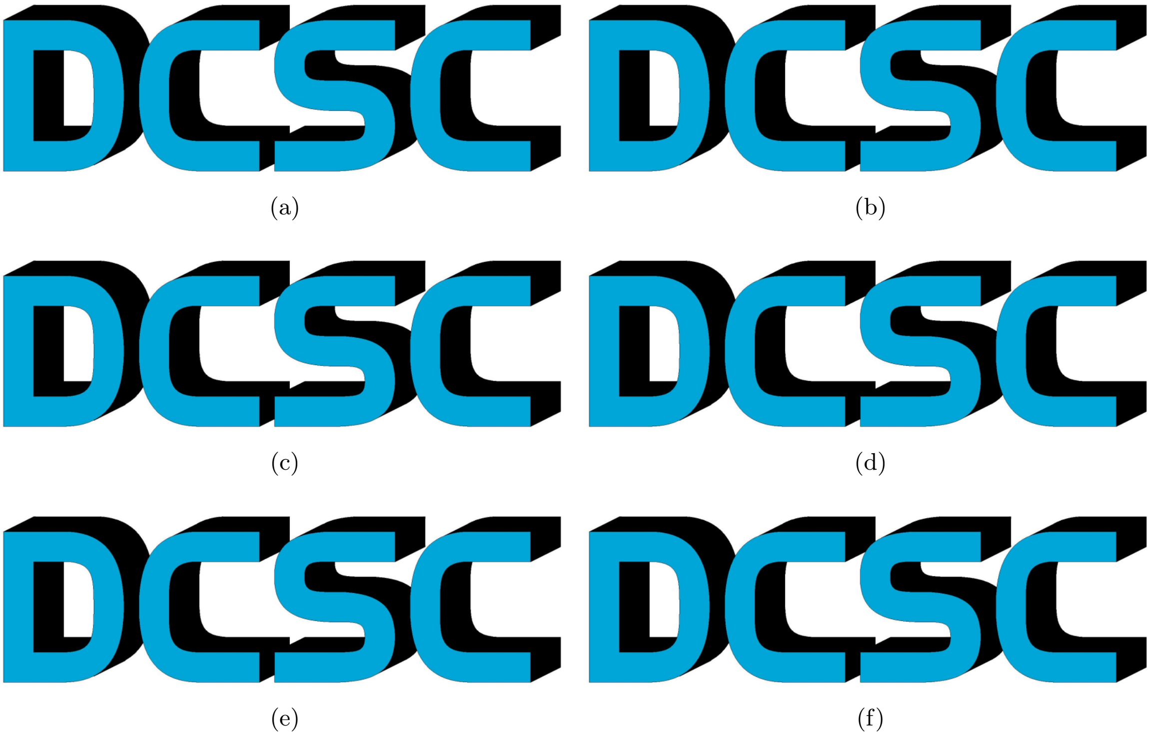
4-4-2 Case B

Figure 4-7a shows the load position along the desired load position $x_{L,d}$ of both controllers.

(a) Load tracking

(b) Load Position Error

Figure 4-7: Controller Comparison Case B



(a) (b)

(c) (d)

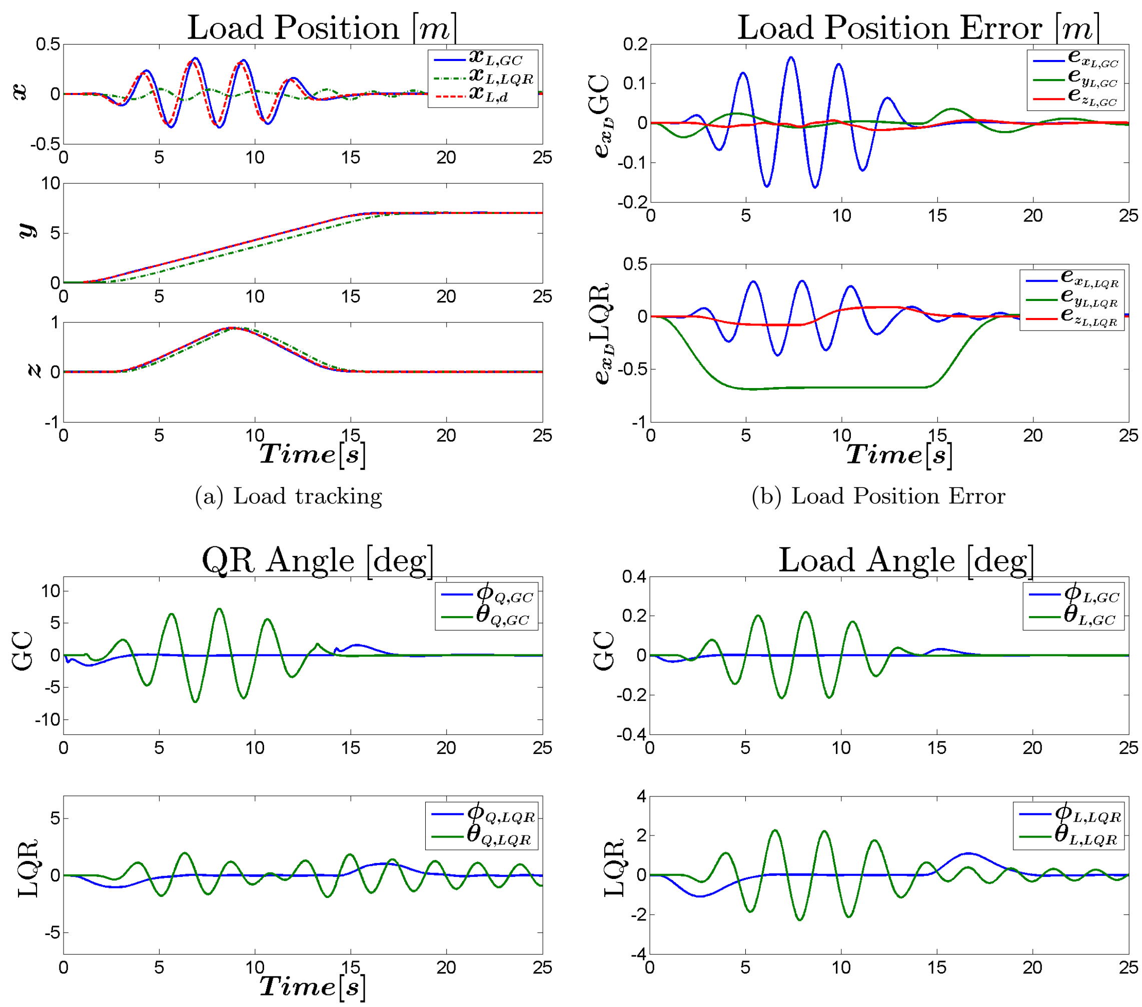
(e) (f)

Figure 4-8: Results Nonlinear Geometric Control Case B

4-4-3 Case C

Figure 4-9a shows the load position along the desired load position $x_{L,d}$ of both controllers.
Figure 4-9b shows the load position error for both control approaches.
Observations: fact that LQR can not control load position is obvious.
OTHER GAINS FOR LQR!
Very small penalty on load angle results in swinging load; decreasing load position error, but very bad anti-swing.

Figure 4-9c shows the QR attitude with respect to $\{\mathcal{I}\}$.
In Figure 4-9d the load angle with respect to $\{\mathcal{B}\}$ is shown.
Observations: Load angles are huge, check results



(a) Load tracking (b) Load Position Error

(c) QR Attitude (d) Load Attitude

Figure 4-9: Controller Comparison Case C

While tracking the required QR attitude, which tilts the QR to reach the desired velocities in the right direction, it can be seen that the system has difficulties to also maintain the desired height, which can be explained by the fact that the total force will not point upwards if the

QR is tilted. Despite the fact that the QR is moving from side to side, the upward force is still controlled to track the desired height.

Figure 4-10a shows the desired load position, and Figure 4-10b shows that the error is mainly the overshoot in the x-direction, due to the fast desired swinging motion.

Figure 4-10c and 4-10d show the tracking errors of the QR attitude and load attitude, respectively.
Observations: $(e_x, e_v, e_q, e_{\dot{q}}, e_R, e_\Omega) = (0, 0, 0, 0, 0, 0)$ is exponentially stable

Figure 4-10e and 4-10f show the tracking error functions of the QR and load, respectively.
Observations: there exist constants $\alpha_q, \beta_q > 0$ such that

$$\Psi_q(q(t), q_d(t)) \leq min \left\{ 2, \alpha_q e^{-\beta_q t} \right\} \tag{4-11}$$

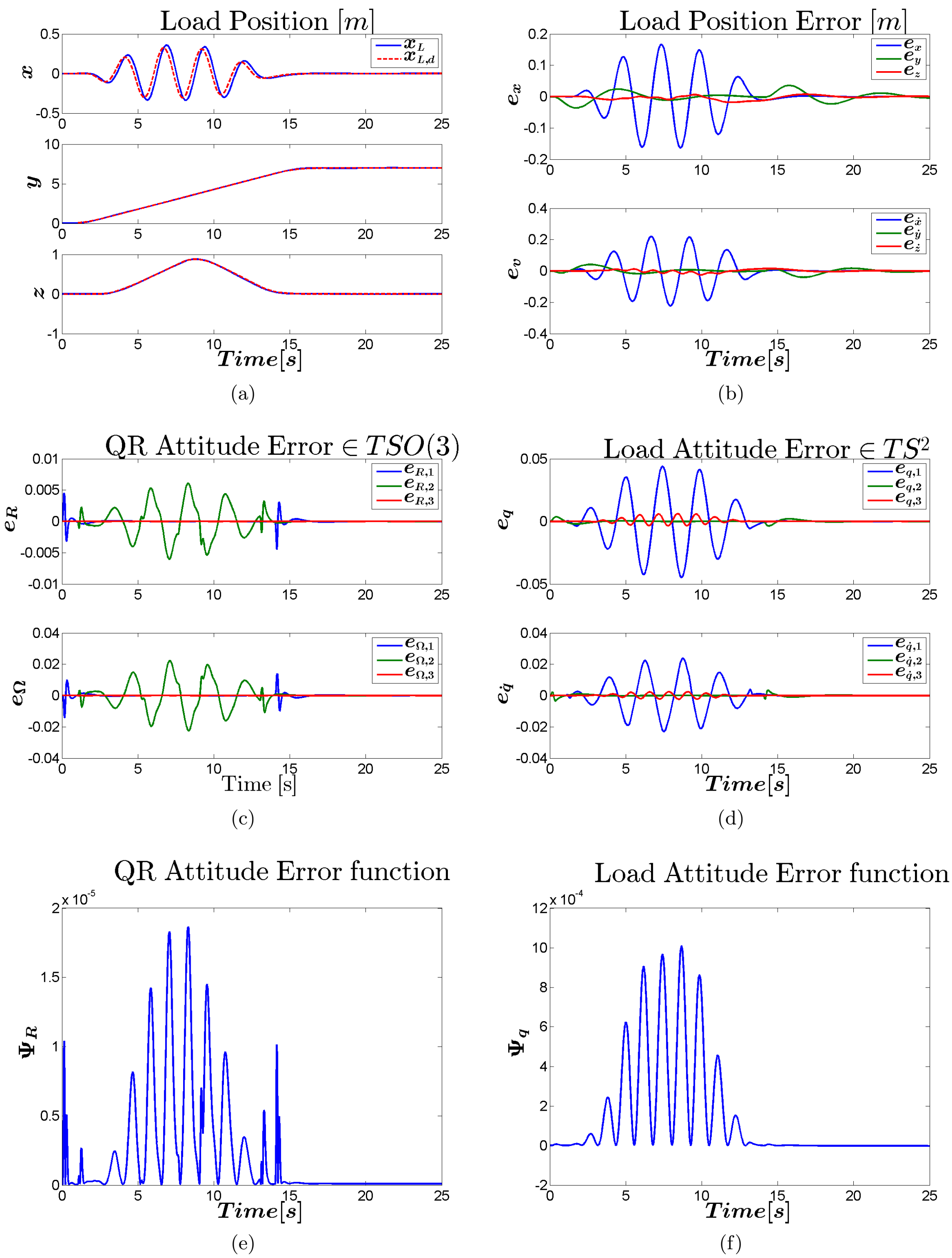


Figure 4-10: Results Nonlinear Geometric Control Case C

4-5 Conclusion

Near the equilibrium configuration, the LQR controller is able to reduce the swing. In fast trajectories however, the shortcomings of the LQR controller become evident.

The nonlinear geometric controller depends on feed forward terms that are obtained from the desired trajectories. Trajectory generation approaches exist that are able to generate the required desired position, velocity and acceleration by however it is possible to compute these with trajectory generating algorithms too.

The controllers are functions of the computed tracking references q_c, R_c and their derivatives. These terms are approximated by a command filter, which means that the accuracy decreases because high frequency terms are filtered.

Chapter 5

Conclusions and Future Work

5-1 Summary and Conclusions

This report starts with an introduction to subject. The aim is described and the motivation for this research is given. This chapter ends with a description of the organization of the report.

After the main introduction, the concepts of Geometric Mechanics are introduced. Differential Geometry is a mathematical discipline that is used to study geometric problems. Instead of employing Euclidean spaces that are defined by Cartesian coordinates, the configuration space of the system is described on nonlinear manifolds. With the tools of differential geometry, differential calculus and integral calculus, a globally defined model is obtained.

Based on the geometric model, a nonlinear geometric control design is discussed. A backstepping approach allows different DOFs of the under-actuated system to be controlled in a cascaded structure. The controllers are driven by error functions, which are defined on the geometric spaces through matrix operations that arise from linear algebra.

The geometric control is based on are functions of error functions defined on nonlinear manifolds. by Differential Geometry.

Next, the experiment is defined. The nonlinear controller is used to track desired load trajectories in different situations. The nonlinear performance is compared to an LQR control Testing the nonlinear Geometric Controller To compare with a common linear controller, LQR control is used to compare

Results are,

This research The conclusions that can be extracted from the experiments is that

5-2 Recommendations for Future Work

5-2-1 Investigate Implementation

Digital Control The concept of Geometric Control is shown under the assumption of continuous-time control. However, an analysis must be done in the discrete-time domain for the implementation of a real-time application. This must verify whether it is feasible to run the controller on an on-board processor on a QR. The control performance could be limited by the bandwidth of either the discretized control system or the wireless communication. It must be investigated whether the control system is still able to deal with the fast dynamics that are required for aggressive maneuvering. Continuous-time Euler-Lagrange equations could be found by minimizing the action integral, which is a function of the Lagrangian. In a similar procedure the discrete-time Euler-Lagrange can be obtained, by minimizing te summation of a discrete Lagrangian, which is demonstrated in [11].

Model identification and validation In this thesis the model parameters are either obtained from examples in literature or arbitrarily chosen. In practice, identification and validation of the QR model and rotor dynamics is required. The mathematical model requires inclusion of the masses, inertia matrix and lengths of the QR, as well as the drag and thrust constants of the rotors, that are very unlikely to be identical.

As a theoretical extension the influence of model mismatches could be simulated.

Robustness The control in this thesis assumes perfect state feedback. In practice the controller depends on visual feedback or data obtained from an on-board inertial measurement unit. Unlike in simulations, this data will contain noise, uncertainties and possibly drift. Based on a nonlinear geometric approach for a QR without load, [5] includes uncertainties in the translational dynamics and rotational dynamics to prove the robustness against unstructured uncertainties. This work could be extended to a QR-Load system to investigate the effects of non-perfect state feedback.

Due to uncertainties In what way would parameter choice in the controllers affect robustness? To test the controller for

How to estimate states?

Parameter Estimation can be done by

State Estimation can be done by

Drawback: assumes all states to be known

Model based. What if analytical model is not accurate?

What parameters must be

5-2-2 Minimum Snap Trajectory Generation

The trajectories described in Section 4-2 were arbitrarily generated by hand to test the performance of the controller in different situations. Whenever more complex trajectories are desired, or when an optimal trajectory is required, this approach is no longer efficient and too complex to solve by hand. A recommended extension to this thesis is the automatic generation of a trajectory. This approach is presented by [23] and applied in [8, 24]. A QP problem is solved by minimizing the second derivative of the acceleration (snap), which guarantees a smooth optimal trajectory. The QP allows inclusion of constraints, such as maximum inputs and checkpoints in trajectories, by formulating these as constraints of the QP problem. Furthermore, it is proven that the system is *differential flat*, meaning that all states and inputs can be expressed in terms of only four states and their derivatives. This property is used to transform the high-dimensional optimization problem into a four-dimensional problem.

5-2-3 Hybrid System

This thesis is only focused on the subsystem where the tension in the cable is non-zero. A possible extension is to apply hybrid control, such that the controller is able to switch between two subsystem models whenever the cable tension switches between zero and non-zero. A trajectory generation method that accounts for the switching dynamics of the hybrid system is presented in [8]. In [1, 8, 24] both subsystems are expressed in the form of one hybrid model.

Appendix A

Appendix

A-1 Derivation of Equations of motion

A-1-1 Load Dynamics

Let x_{CM} denote the position of the center of mass of the combined Quadrotor-Load system, expressed in $\{\mathcal{I}\}$. Which can be found by

$$\begin{aligned} m_Q(x_Q - x_{CM}) + m_L(x_L - x_{CM}) = 0 \\ (m_Q + m_L)x_{CM} = m_Q x_Q + m_L x_L \end{aligned} \tag{A-1}$$

Applying Newton's laws of motion to (A-1) and inserting (2-14) gives the

$$\begin{aligned} (m_Q + m_L)\ddot{x}_{CM} = fRe_3 - (m_Q + m_L)ge_3 \\ (m_Q + m_L)(\ddot{x}_L + ge_3) = fRe_3 + m_Q L\ddot{q} \end{aligned} \tag{A-2}$$

Here comes the derivation of $\ddot{q}$, obtained by geometric mechanics.

A-2 Derivation Error dynamics

A-2-1 Quadrotor Attitude

From the angular velocity tracking error e_Ω follows

$$\begin{aligned} e_\Omega &= \Omega - R^T R_d \Omega_d, \\ \hat{e}_\Omega &= \hat{\Omega} - R^T R_d \hat{\Omega}_d R_d^T R \end{aligned} \tag{A-3}$$

The attitude tracking error is given by

$$e_R = \frac{1}{2}(R_d^T R - R^T R_d)^\vee \tag{A-4}$$

where its time derivative is given by

$$
\begin{aligned}
\dot{e}_R &= \frac{1}{2}(\dot{R}_d^T R + R_d^T \dot{R} - \dot{R}^T R_d - R^T \dot{R}_d)^\vee \\
&= \frac{1}{2}((R_d\hat{\Omega}_d)^T R + R_d^T(R\hat{\Omega}) - (R\hat{\Omega})^T R_d - R^T(R_d\hat{\Omega}_d))^\vee \\
&= \frac{1}{2}(-\hat{\Omega}_d R_d^T R + R_d^T R\hat{\Omega} + \hat{\Omega} R^T R_d - R^T R_d \hat{\Omega}_d)^\vee \\
&= \frac{1}{2}(R_d^T R(\hat{\Omega} - R^T R_d \hat{\Omega}_d R_d^T R) + (\hat{\Omega} - R^T R_d \hat{\Omega}_d R_d^T R) R^T R_d)^\vee
\end{aligned}
\tag{A-5}
$$

which can be rewritten by substituting Equation A-3 as follows

$$
\dot{e}_R = \frac{1}{2}(R_d^T R \hat{e}_\Omega + \hat{e}_\Omega R^T R_d)^\vee \tag{A-6}
$$

The time derivative of the angular velocity tracking error e_Ω is given by

$$
\begin{aligned}
\dot{e}_\Omega &= \dot{\Omega} - \dot{R}^T R_d \Omega_d - R^T \dot{R}_d \Omega_d - R^T R_d \dot{\Omega}_d \\
&= \dot{\Omega} - (R\hat{\Omega})^T R_d \Omega_d - R^T (R_d \hat{\Omega}_d) \Omega_d - R^T R_d \dot{\Omega}_d \\
&= \dot{\Omega} + \hat{\Omega} R^T R_d \Omega_d - R^T (R_d \hat{\Omega}_d) \Omega_d - R^T R_d \dot{\Omega}_d
\end{aligned}
\tag{A-7}
$$

where $\hat{\Omega}_d \Omega_d = \Omega_d \times \Omega_d = 0$. Substituting Equation 2-35 results in

$$
\dot{e}_\Omega = J^{-1}(-\Omega \times J\Omega + M) + \hat{\Omega} R^T R_d \Omega_d - R^T R_d \dot{\Omega}_d \tag{A-8}
$$

A-2-2 Load Attitude

For the load attitude tracking problem on manifold $\mathbb{S}^2$, the error function E_{cl} is defined as follows [16, p.]

$$
E_{cl}(v_q, w_r) = \Psi(q, r) + \frac{1}{2} \parallel v_q - \mathcal{T}(q, r) \cdot w_r \parallel_{\mathbb{G}}^2 \tag{A-9}
$$

where $\parallel \cdot \parallel_{\mathbb{G}}$ denotes the norm on tensors, induced by a Riemannian metric. The error function and transport map are defined for $\mathbb{S}^2$ as follows

$$
\Psi_{\mathbb{S}^2}(\mathbf{x}, \mathbf{r}) = k_P(1 - \langle\langle \mathbf{r}, \mathbf{x} \rangle\rangle_{\mathbb{R}^3}) \tag{A-10}
$$

and

$$
\mathcal{T}_{\mathbb{S}^2}(\mathbf{x}, \mathbf{r}) \cdot \mathbf{v} = (\mathbf{r} \times \mathbf{v}) \times \mathbf{x} \tag{A-11}
$$

where $\mathbf{r}(t)$ is the reference trajectory, and $\mathbf{x} \in \mathbb{S}^2$ and $\mathbf{v} \in T_\mathbf{x}\mathbb{S}^2$. Substituting Equations A-10 and A-11 into Equation A-9 results in the following closed-loop energy function

$$
E_{cl}(t, \mathbf{x}, \mathbf{v}) = k_P(1 - \langle\langle \mathbf{r}, \mathbf{x} \rangle\rangle_{\mathbb{R}^3}) + \frac{1}{2} m \parallel \mathbf{v} - (\mathbf{r}(t) \times \dot{\mathbf{r}}(t)) \times \mathbf{x} \parallel_{\mathbb{R}^3}^2 \tag{A-12}
$$

A-3 LQR controller

A-3-1 Modeling

The linearized model is written into a first order ODE of the form

$$\dot{\mathbf{x}} = A\mathbf{x} + Bu \tag{A-13}$$
$$y = C\mathbf{x} + Du \tag{A-14}$$

with the following state- and input vectors

$$\begin{aligned} \mathbf{x} &= \begin{bmatrix} x & y & z & \phi & \theta & \psi & \phi_L & \theta_L & \dot{x} & \dot{y} & \dot{z} & \dot{\phi} & \dot{\theta} & \dot{\psi} & \dot{\phi}_L & \dot{\theta}_L \end{bmatrix}^T \\ u &= \begin{bmatrix} f & M_\phi & M_\theta & M_\psi \end{bmatrix}^T \end{aligned} \tag{A-15}$$

The model is linearized about the hovering flight mode. All translational and rotational velocities are zero during hover. The positional states and the yaw angle do not affect the dynamics, and are set equal to zero. A thrust input $u_1 = g(mQ+mL)$ is required to maintain hover, and all other control inputs are set equal to zero. The states and inputs in the equations of motion are substituted by an initial condition and a perturbation

$$\dot{\mathbf{x}} \to \dot{\mathbf{x}}_0 + \delta\dot{\mathbf{x}}, \quad \mathbf{x} \to \mathbf{x}_0 + \delta\mathbf{x}, \quad u \to u_0 + \delta u \tag{A-16}$$

$$\begin{aligned} \mathbf{x}(0) &= \mathbf{0} \\ u(0) &= \begin{bmatrix} g(m_Q + m_L) & 0 & 0 & 0 \end{bmatrix}^T \end{aligned} \tag{A-17}$$

The linearized equations of motion are rearranged into Equation A-18 and substituted in Equation A-13.

$$\begin{bmatrix} content... \end{bmatrix} \begin{bmatrix} \delta\ddot{x} \\ \delta\ddot{y} \\ \delta\ddot{z} \\ \delta\ddot{\phi} \\ \delta\ddot{\theta} \\ \delta\ddot{\psi} \\ \delta\ddot{\phi}_L \\ \delta\ddot{\theta}_L \end{bmatrix} + \begin{bmatrix} content... \end{bmatrix} \begin{bmatrix} \delta x \\ \delta y \\ \delta z \\ \delta\phi \\ \delta\theta \\ \delta\psi \\ \delta\phi_L \\ \delta\theta_L \end{bmatrix} = \begin{bmatrix} content... \end{bmatrix} \begin{bmatrix} \delta u_1 \\ \delta u_2 \\ \delta u_3 \\ \delta u_4 \end{bmatrix} \tag{A-18}$$

```
LQRA =
```

```
LQRB =
```

The tuning parameters of the LQR controller a chosen as follows

$$\begin{aligned} Q &= diag\begin{pmatrix} 10 & 10 & 100, & 1 & 1 & 1, & 1 & 1, & 1 & 1 & 1, & 1 & 1 & 1, & 1 & 1 \end{pmatrix} \\ R &= diag\begin{pmatrix} 0.044, & 1.56, & 1.56, & 1.56 \end{pmatrix} \end{aligned} \tag{A-19}$$

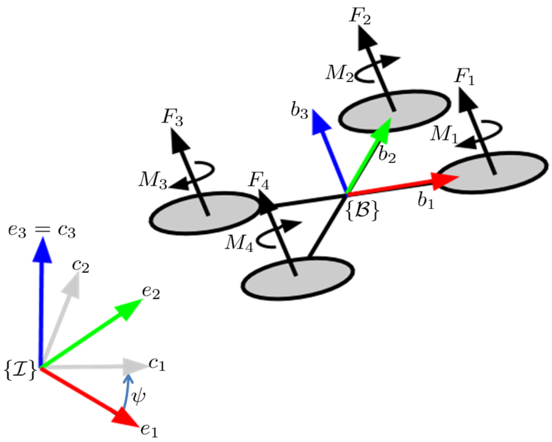
`Matlab` command `lqr(LQRA,LQRB,Q,R)` generates the following gain matrix K

```
K =
```

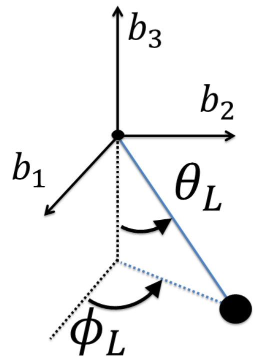
A-4 Classical Modeling

This section describes the derivation of the model by using classical modeling techniques.

When assuming small angle maneuvers, *Euler-angles* can be used to locally parameterize the orientation of the body-fixed reference coordinate frame with respect to the inertial reference coordinate frame. Simple linear controllers are often based on a linearized dynamical model, applying this small angles assumption.



(a) Quadrotor model representation



(b) Load angles

Figure A-1: Model representation

The following equations of motion follow from Newton's law.

$$\begin{aligned} \dot{x}_Q &= v_Q \\ m_Q \dot{v}_Q &= fRe_3 - m_Q g e_3 - Tq \\ \dot{x}_L &= v_L \\ m_L \dot{v}_L &= -m_L g e_3 + Tq \end{aligned} \tag{A-20}$$

where $x_Q = x_L - Lq$. T is the cable tension, defined by $T = |f|q$, where $|f| = m_L \dot{v}_L$ is the magnitude of the force.

Because Euler-Angles are used, a function is required that maps a vector of the Z-X-Y Euler angles to its rotation matrix $R \in SO(3)$, which is denoted as [20]

$$R_{ZXY}(\phi, \theta, \psi) = \begin{bmatrix} c_\psi c_\theta - s_\phi s_\psi s_\theta & -c_\phi s_\psi & c_\psi s_\theta + c_\theta s_\phi s_\psi \\ c_\theta s_\psi + c_\psi s_\phi s_\theta & c_\phi c_\psi & s_\psi s_\theta - c_\psi c_\theta s_\phi \\ -c_\phi s_\theta & s_\phi & c_\phi c_\theta \end{bmatrix} \tag{A-21}$$

The Z-X-Y Euler angles rotate $\{\mathcal{B}\}$, as can be seen in Figure A-1a. The first rotation by yaw angle ψ is around the z-axis of $\{\mathcal{I}\}$. Next is the rotation by roll angle ϕ, and the last rotation is by pitch angle θ.

The unit vector q from the QR to the load is represented in $\{\mathcal{B}\}$. Define ϕ_L as the rotation of the load around the z-axis, measured from $\vec{b}_1$, and θ_L is the angle between the cable and the z-axis of $\{\mathcal{B}\}$, see Figure A-1b. The Cartesian coordinates can be retrieved through

$$x_L = x_Q + qL \tag{A-22}$$

where

$$q = \begin{bmatrix} s_{\theta_L} c_{\phi_L} \\ s_{\theta_L} s_{\phi_L} \\ -c_{\theta_L} \end{bmatrix} \tag{A-23}$$

Differentiating Equation A-22 and A-23 gives

$$\begin{aligned} \ddot{x}_L &= \ddot{x}_Q + \ddot{q}L \\ \ddot{q} &= \begin{bmatrix} \ddot{\theta}_L c_{\theta_L} c_{\phi_L} - \ddot{\phi}_L s_{\theta_L} s_{\phi_L} - \dot{\phi}_L^2 s_{\theta_L} c_{\phi_L} - \dot{\theta}_L^2 s_{\theta_L} c_{\phi_L} - 2\dot{\theta}_L \dot{\phi}_L c_{\theta_L} s_{\phi_L} \\ \ddot{\theta}_L c_{\theta_L} s_{\phi_L} + \ddot{\phi}_L s_{\theta_L} c_{\phi_L} - \dot{\phi}_L^2 s_{\theta_L} s_{\phi_L} - \dot{\theta}_L^2 s_{\theta_L} s_{\phi_L} + 2\dot{\theta}_L \dot{\phi}_L c_{\theta_L} c_{\phi_L} \\ \ddot{\theta}_L s_{\theta_L} + \dot{\theta}_L^2 c_{\theta_L} \end{bmatrix} \end{aligned} \tag{A-24}$$

$$\begin{aligned} \ddot{x}_Q &= \frac{1}{m_Q}(f(c_\psi s_\theta + c_\theta s_\phi s_\psi) - T s_{\theta_L} c_{\psi_L}) \\ \ddot{y}_Q &= \frac{1}{m_Q}(f(s_\psi s_\theta - c_\psi c_\theta s_\phi) - T s_{\theta_L} s_{\psi_L}) \\ \ddot{z}_Q &= \frac{1}{m_Q}(f(c_\phi c_\theta) - T c_{\theta_L}) - g \end{aligned} \tag{A-25}$$

A-5 Figures

Figure A-2: Simulink Command Filter

Bibliography

[1] K. Sreenath, Taeyoung Lee, and V. Kumar, "Geometric control and differential flatness of a quadrotor UAV with a cable-suspended load," in *52nd IEEE Conference on Decision and Control*, pp. 2269–2274, IEEE, dec 2013.

[2] R. P. K. Jain and T. Keviczky, "MSc Thesis: Transportation of Cable Suspended Load using Unmanned Aerial Vehicles: A Real-time Model Predictive Control approach," 2015.

[3] S. Sadr, S. A. A. Moosavian, and P. Zarafshan, "Dynamics Modeling and Control of a Quad-rotor Helicopter," *Journal of Robotics, Hindawi Publishing Corporation*, vol. 2014, no. August 2016, 2014.

[4] T. Lee, M. Leok, and N. H. Mcclamroch, "Geometric Tracking Control of a Quadrotor UAV on SE(3)," *49th IEEE Conference on Decision and Control*, no. 3, pp. 5420–5425, 2010.

[5] F. Goodarzi, D. Lee, and T. Lee, "Geometric nonlinear PID control of a quadrotor UAV on SE (3)," *Control Conference (ECC), 2013 European*, pp. 3845–3850, 2013.

[6] K. Sreenath, N. Michael, and V. Kumar, "Trajectory generation and control of a quadrotor with a cable-suspended load - A differentially-flat hybrid system," in *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 4888–4895, 2013.

[7] K. Sreenath, T. Lee, and V. Kumar, "Geometric control and differential flatness of a quadrotor UAV with a cable-suspended load," in *Proceedings of the IEEE Conference on Decision and Control*, pp. 2269–2274, 2013.

[8] S. Tang, "Aggressive Maneuvering of a Quadrotor with a Cable-Suspended Payload," tech. rep., University of Pennsylvania Philadelphia, Pennsylvania, 2014.

[9] N. Chaturvedi, "Rigid-Body Attitude Control," *IEEE Control Systems*, vol. 31, no. 3, pp. 30–51, 2011.

[10] R. M. Murray, Z. Li, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*, vol. 29. 1994.

[11] T. Lee, *Computational Geometric Mechanics and Control of Rigid Bodies.* PhD thesis, The University of Michigan, 2008.

[12] T. L. T. Lee, N. McClamroch, and M. Leok, “A lie group variational integrator for the attitude dynamics of a rigid body with applications to the 3D pendulum,” *Proceedings of 2005 IEEE Conference on Control Applications, 2005. CCA 2005.*, pp. 962–967, 2005.

[13] T. Lee, M. Leok, and N. H. McClamroch, “Lagrangian mechanics and variational integrators on two-spheres,” *International Journal for Numerical Methods in Engineering*, vol. 79, no. 9, pp. 1147–1174, 2009.

[14] T. Lee, M. Leok, and N. H. McClamroch, “Stable Manifolds of Saddle Points for Pendulum Dynamics on Sˆ2 and SO(3),” p. 9, 2011.

[15] T. Lee, M. Leok, and N. H. McClamroch, “Lie group variational integrators for the full body problem in orbital mechanics,” *Celestial Mechanics and Dynamical Astronomy*, vol. 98, no. 2, pp. 121–144, 2007.

[16] F. Bullo and A. D. Lewis, *Geometric control of mechanical systems: modeling, analysis, and design for simple mechanical control systems.* Springer, 2005.

[17] J. H. Gillula, H. Huang, M. P. Vitus, and C. J. Tomlin, “Design of guaranteed safe maneuvers using reachable sets: Autonomous quadrotor aerobatics in theory and practice,” in *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 1649–1654, 2010.

[18] N. A. Chaturvedi, T. Lee, M. Leok, and N. H. McClamroch, “Nonlinear dynamics of the 3D pendulum,” *Journal of Nonlinear Science*, vol. 21, no. 1, pp. 3–32, 2011.

[19] T. Lee, M. Leok, and N. McClamroch, “Control of complex maneuvers for a quadrotor UAV using geometric methods on SE (3),” *arXiv*, 2010.

[20] R. Mahony, V. Kumar, and P. Corke, “Multirotor Aerial Vehicles: Modeling, Estimation, and Control of Quadrotor,” *IEEE Robotics & Automation Magazine*, vol. 19, no. 3, pp. 20–32, 2012.

[21] I. Kanellakopoulos, P. V. Kokotovic, and A. S. Morse, “Systematic Design of Adaptive Controllers for Feedback Linearizable Systems,” *IEEE Transactions on Automatic Control*, vol. 36, no. 11, pp. 1241–1253, 1991.

[22] J. A. Farrell, M. Polycarpou, M. Sharma, and W. Dong, “Command Filtered Backstepping,” pp. 1923–1928, 2008.

[23] D. Mellinger and V. Kumar, “Minimum snap trajectory generation and control for quadrotors,” in *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 2520–2525, 2011.

[24] S. Tang and V. Kumar, “Mixed Integer Quadratic Program Trajectory Generation for a Quadrotor with a Cable-Suspended Payload,” *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2216–2222, 2015.

Nomenclature

ϵ	Tuning parameter to enable rapid exponential convergence of e_R, e_Ω
$\lambda_M(\cdot)$	Maximum eigenvalue
ω_i	Angular speed of rotor i
$\{\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3\}$	Unit vectors along the axes of $\{\mathcal{B}\}$
$\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$	Unit vectors along the axes of $\{\mathcal{I}\}$
$\{\mathcal{B}\}$	Body Frame
$\{\mathcal{I}\}$	Inertial World Frame
b	Thrust factor
d	Drag factor
f	Total thrust in direction of $\mathbf{b}_3$, expressed in $\{\mathcal{B}\}$. $f = \sum_{i=1}^{4} F_i$
F_i	Force generated by rotor i
g	Gravitation constant
$J \in \mathbb{R}^{3\times 3}$	Inertia tensor of QR
L	Length of the cable
l	Distance from the rotor to the QR CM
M	Total moment around axes of $\{\mathcal{B}\}$, expressed in $\{\mathcal{B}\}$. $M = \begin{bmatrix} M_\phi & M_\theta & M_\psi \end{bmatrix}^T$
M_i	Drag moment generated by each propellor
$q \in \mathbb{S}^2$	Unit vector from QR to Load
$x_L \in \mathbb{R}^3$	Position of the load

$x_Q \in \mathbb{R}^3$	Position of the QR CM
x_{CM}	Position CM of QR-Load system

Acronyms

QR	Quadrotor
UAV	Unmanned Aerial Vehicle
com	Center of Mass
DOF	Degree of Freedom
PID	Proportional-Integral-Derivative (Controller)
MPC	Model Predictive Control
LQR	Linear Quadratic Regulator
QP	Quadratic Programming