Master of Science Thesis

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

N.N. Vo

July 23, 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

July 23, 2017

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

The work in Master of Science Thesis was supported by Alten. 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: July 23, 2017

Supervisor: dr.ir. T. Keviczky

Readers: Dr. J Alonso-Mora

A. Sharifi Kolarijani

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 presented and the results of both the linear and nonlinear controller 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 — N.N. Vo
July 23, 2017

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 [?] 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 [?].

****

Bart: I think you might need some references here.
Nam:

****

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 [?, ?] and nonlinear geometric control of the load position, load attitude and QR attitude of a QR-Load system [?, ?, ?]. 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 5 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

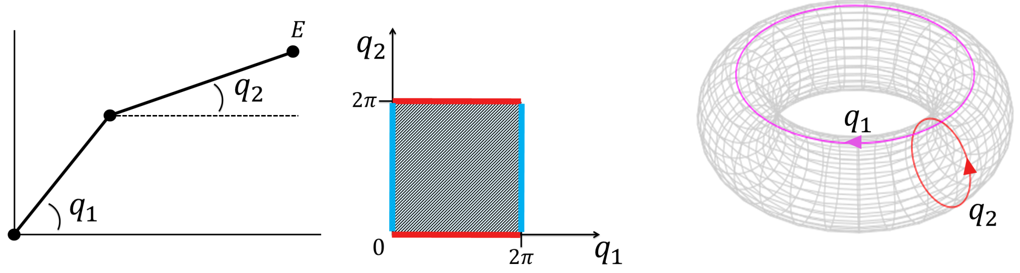
For the derivation of the equations of motions, traditional modeling methods often parameterize the rotations in a local coordinate system. This can be done with Euler Angles, and despite this parametrization might result in singularities, this is a commonly used method to describe rotations. There are 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.



(a) (b) (c)

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:

****

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

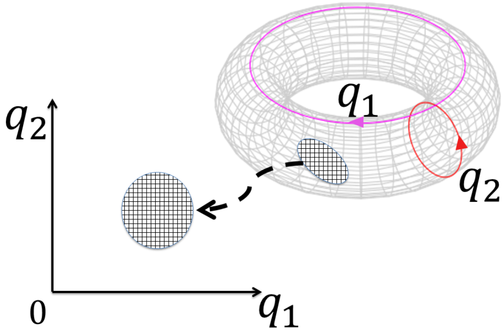
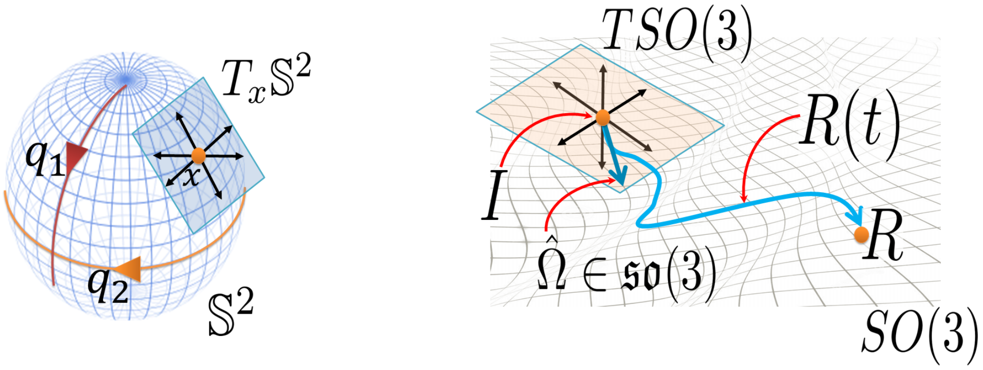


Figure 2-2: A manifold locally resembles a Euclidean space

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

$$SO(3) \triangleq \left\{R \in \mathbb{R}^{3\times3} | RR^T = I_{3\times3}, det(R) = 1\right\} \tag{2-1}$$

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. [?, ?]

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

$$\hat{x}y = x \times y = -y \times x = -\hat{y}x, \tag{2-4}$$

$$tr[A\hat{x}] = \frac{1}{2}tr[\hat{x}(A - A^T)] = -x^T(A - A^T)^\vee, \tag{2-5}$$

$$\hat{x}A + A^T\hat{x} = (\{tr[A]I_{3\times3} - A\}x)^\wedge, \tag{2-6}$$

$$R\hat{x}R^T = (Rx)^\wedge, \tag{2-7}$$

for any $x, y \in \mathbb{R}^3, A \in \mathbb{R}^{3\times3}$, 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.

Considering the properties of the system, the QR is described as a rigid body with six degrees of freedom, driven by forces and moments. The configuration of the QR can be described by the location of the QRs CM, $x_Q \in \mathbb{R}^3$, described in the Euclidean space w.r.t.

$\{\mathcal{I}\}$ and by 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 the QR can be expressed on $SE(3)$, 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 $\{\mathcal{I}\}$.

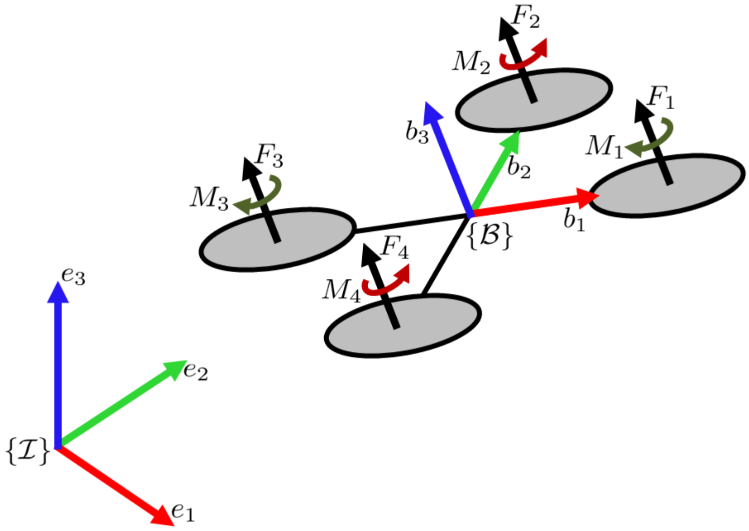


Figure 2-4: Quadrotor model representation

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 Center of Mass (CM) and M_ϕ, M_θ, M_ψ denote the moments around the x, y, z-axis in $\{\mathcal{B}\}$, respectively.

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 CM 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.

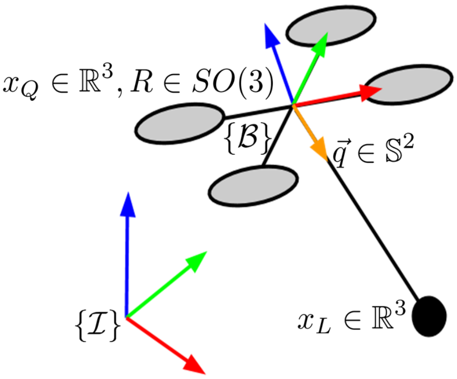

$x_Q \in \mathbb{R}^3, R \in SO(3)$
$\{\mathcal{B}\}$
$\vec{q} \in \mathbb{S}^2$
$\{\mathcal{I}\}$
$x_L \in \mathbb{R}^3$

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 [?].

Euler-Lagrange To develop the Euler-Lagrange equations for mechanical systems that evolve on manifolds, an approach developed by [?, ?, ?, ?] 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 $J \in \mathbb{R}^{3\times3}$ 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 are obtained as shown in [**?**, **?**, **?**, **?**].

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

The following variations follow from differentiation,

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

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 are 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 the concepts of Nonlinear Geometric Control, and the control design structure is discussed. The required control inputs are calculated by defining the tracking errors on nonlinear manifolds similar to the configuration spaces of the system dynamics.

A backstepping control approach is applied for the control of the load position tracking problem, allowing different controllers to operate in a cascaded structure. The controllers that are designed for each flight mode, are discussed in Section 3-2.

****

bart: Descibe just as in chapter 2 what we are going to read in the next sections
nam: please check

****

3-1 Nonlinear Geometric Control

Many control systems are developed for the standard form of ordinary differential equations, namely $\dot{x} = f(x, u)$, where the state and the control input are denoted by x and u. 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. In order to express the configuration space globally, a nonlinear space is required. A dynamical model on this nonlinear space is obtained in the previous chapter.

Geometric Control Theory is the study on how geometry of the state space influences controls 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 manifolds instead of local charts. In contrast to locally defined linear control, nonlinear geometric control can be defined almost globally, avoiding singularities that occur in the representation of large angles and complex maneuvering.

Global nonlinear dynamics of various classes of closed loop attitude control systems have been studied in recent years [?]. In contrast to hybrid control systems [?], complicated reachability set analysis is not required to guarantee safe switching between different flight modes, as the region of attraction for each flight mode covers the configuration space almost globally.

Backstepping Control A backstepping approach, or cascade control, is a Lyapunov based technique to design the control of nonlinear dynamical systems and ensures Lyapunov stability. This approach is commonly used for the control of QRs [?] and will also be used in this research for the control of the load trajectory tracking problem.
The basic principle is to create a cascaded structure by starting with a stable system as a base, then "stepping back" from this base to add a control loop around it that stabilizes the new system and enables the control of another state. This is repeated until the final external control is reached, see Figure 3-1. The control law is designed by using states as virtual control signals. Each loop computes a virtual command signal for the adjacent inner loop. This backstepping approach determines how to stabilize the QR with the control inputs f and M. The inner controller determines what the required control inputs are, driven by R_c. The next controller calculates how to drive R_c based on q_c, such that the QR is stabilized. And the last controller determines which q_c is needed to follow the desired load position $x_{L,d}$.

Because 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 parts of the DOFs are controlled. The flight modes and their functions are defined in order, from the most inner loop to the most outer loop, as follows

- QR Attitude Controlled Mode
  - Track a desired QR attitude $R_d(t)$ and a heading direction $b_{1_d}(t)$
  - Give a desired input M for system
- Load Attitude Controlled Mode
  - Track a desired load attitude command $q_d(t)$
  - Give a computed QR attitude R_c for the QR attitude controller (instead of $R_d(t)$)
- Load Position Controlled Mode
  - Track a desired load position $x_{L,d}(t)$
  - Give a computed load attitude q_c for the load attitude controller (instead of $q_d(t)$)

where the subscript d denotes a desired tracking reference, and the subscript c denotes a computed value that is calculated as a tracking reference. The difference in this notation is whether the signal is a predefined desired signal, or a signal computed by a controller.

The design of the controllers for the QR attitude can be found in [?], and the controllers of load attitude- and position this can be found in [?]. Thorough stability analyses are presented in either references. For a deeper understanding of Lyapunov stability analysis in geometric control, the reader can refer to [?].

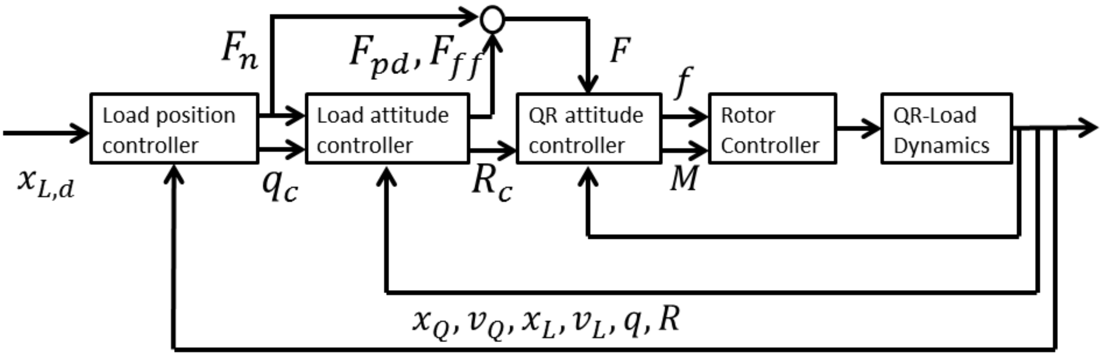


Figure 3-1: Nonlinear Geometric Control Loop of the QR-Load system [?]

Configuration Errors The control system for a trajectory tracking problem requires state feedback and a measure of errors defined by a difference between the desired and current states. The system dynamics evolve on nonlinear manifolds that describe the configuration spaces for the QR attitude $\in SO(3)$ and the load attitude $\in \mathbb{S}^2$. Tracking errors functions are defined on these same manifolds, as shown in [?], and form the basis of the controllers that must stabilize the QR-Load system.

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, the attitude error is denoted as $R_d^T R$. The QR attitude error function Ψ_R on $SO(3)$ is chosen to be [?]

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

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 $SO(3)$. Equation 2-27 states that the variation of the rotation matrix is expressed as $\delta R = R\hat{\eta}$ for $\eta \in \mathbb{R}^3$. Such that with Equation 2-5, the derivative of Equation 3-1 is given by

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

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 derivative, the attitude tracking error e_R is obtained

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

It is important to note that the tangent vectors $\dot{R}$ and $\dot{R}_d$ cannot be compared directly, since they do not lie in the same space. $\dot{R}$ and $\dot{R}_d$ are expressed in their own tangent spaces, denoted by $T_R SO(3)$ and $T_{R_d} SO(3)$, respectively. For this reason, $\dot{R}_d$ is transformed into a vector on $T_R SO(3)$ to compare it with $\dot{R}$. This is done by an mathematical object called a *transport map*, which enables the comparison of velocities living in different spaces. See

Figure 3-2, where two curves $R(t), R_d(t)$ evolve on manifold $SO(3)$, such that transport map $\mathcal{T}(R, R_d) : T_{R_d}SO(3) \mapsto T_R SO(3)$ allows comparison of $\dot{R}$ and $\dot{R}_d$.

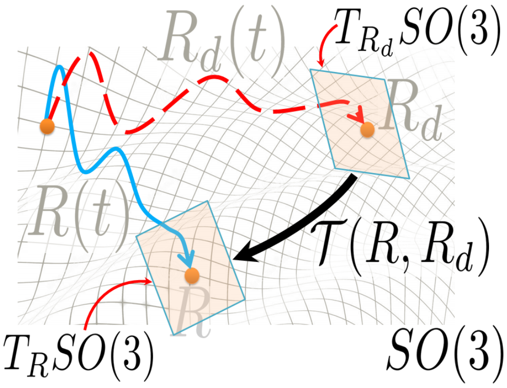


Figure 3-2: Transport map $\mathcal{T}(R, R_d)$

The comparison of the two tangent vectors is needed to calculate the error of body angular velocity Ω. This is derived from the velocity error that corresponds to the transport map $\mathcal{T}(R, Rd)$, which is defined as

$$\dot{e} = \dot{R} - \dot{R}_d(R_d^T R) \tag{3-4}$$

This can be rewritten as follows

$$\begin{aligned} \dot{R} - \dot{R}_d(R_d^T R) &= R\hat{\Omega} - R_d\hat{\Omega}_d(R_d^T R) \\ &= R(\Omega)^\wedge - (RR^T)R_d\hat{\Omega}_d R_d^T R \\ &= R(\Omega)^\wedge - R(R^T R_d\Omega_d)^\wedge \\ &= R(\Omega - R^T R_d\Omega_d)^\wedge \end{aligned} \tag{3-5}$$

From this follows the angular velocity tracking error e_Ω in $\{\mathcal{B}\}$, which is defined as

$$e_\Omega = \Omega - R^T R_d \Omega_d \tag{3-6}$$

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

$$\frac{d}{dt}(R_d^T R) = (R_d^T R)\hat{e}_\Omega \tag{3-7}$$

Next, the load attitude error function is expressed on $\mathbb{S}^2$ and represents the distance from the direction q to the desired direction q_d. This is given by

$$\Psi_q = 1 - q_d^T q \tag{3-8}$$

In the same fashion a *transport map* is used for a comparison between the tangent vectors on tangent spaces $T_q\mathbb{S}^2$ and $T_{q_d}\mathbb{S}^2$. This results in the following error functions on $T\mathbb{S}^2$

$$e_q = \hat{q}^2 q_d \tag{3-9}$$
$$e_{\dot{q}} = \dot{q} - (q_d \times \dot{q}_d) \times q \tag{3-10}$$

The tracking errors for position and velocity are defined as

$$e_x = x - x_d \tag{3-11}$$

$$e_v = v - v_d \tag{3-12}$$

where $v_d = \dot{x}_d$ and $x_d(t) \in \mathbb{R}^3$ is a smooth desired load position.

3-2 Tracking modes

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 command $R_d(t)$. This is done by controlling the earlier obtained error dynamics of e_R and e_Ω. The derivations of the equations in this section can be found in Section A-4.

From Equations 3-3 and 3-6, the derivative of the attitude tracking error e_R can be written as

$$\dot{e}_R = \frac{1}{2}(R_d^T R \hat{e}_\Omega + \hat{e}_\Omega R^T R_d)^\vee \tag{3-13}$$

Similar to Equation 2-34, the kinematics equation for the desired attitude can be written as

$$\dot{R}_d = R_d \hat{\Omega}_d, \text{ and so } \hat{\Omega}_d = R_d^T \dot{R}_d \tag{3-14}$$

The definition of the desired angular acceleration $\dot{\Omega}_d$ can then be defined as follows

$$\begin{aligned} \dot{\hat{\Omega}}_d &= (\dot{R}_d^T \dot{R}_d) + (R_d^T \ddot{R}_d) \\ &= (R_d \hat{\Omega}_d)^T (R_d \hat{\Omega}_d) + (R_d^T \ddot{R}_d) \\ &= -\hat{\Omega}_d \hat{\Omega}_d + R_d^T \ddot{R}_d, \\ \dot{\Omega}_d &= (-\hat{\Omega}_d \hat{\Omega}_d + R_d^T \ddot{R}_d)^\vee \end{aligned} \tag{3-15}$$

From previous equations and Equations 2-34, 3-6 and the fact that $\hat{\Omega}_d \Omega_d = 0$, follows that the derivative of the angular velocity tracking error e_Ω can be written as

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

By substituting Equation 2-35 follows

$$\dot{e}_\Omega = J^{-1}(-\Omega \times J\Omega + M) + \hat{\Omega} R^T R_d \Omega_d - R^T R_d \dot{\Omega}_d \tag{3-17}$$

Now the control input M can be defined as a proportional term, a derivative term and a canceling term, as follows [**?**]

$$M = -k_R e_R - k_\Omega e_\Omega + \Omega \times J\Omega - J(\hat{\Omega} R^T R_d \Omega_d - R^T R_d \dot{\Omega}_d) \tag{3-18}$$

such that Equation 3-16 is reduced to

$$J \dot{e}_\Omega = -k_R e_R - k_\Omega e_\Omega \tag{3-19}$$

for any positive constants k_R, k_Ω. In [?], Equation 3-18 is defined as

$$M = -\frac{1}{\epsilon^2}k_R e_R - \frac{1}{\epsilon}k_\Omega e_\Omega + \Omega \times J\Omega - J(\hat{\Omega}R^T R_d \Omega_d - R^T R_d \dot{\Omega}_d) \tag{3-20}$$

where ϵ is a parameter to enable rapid exponential convergence of the attitude- and angular velocity error functions, such that $0 < \epsilon < 1$.

A stability analysis of the controller is presented in [?] and it is proven that the zero equilibrium of the closed loop tracking error $(e_R, e_\Omega) = (0,0)$ is exponentially stable, if the initial conditions satisfy

$$\Psi_R(R(0), R_d(0)) < 2 \tag{3-21}$$

$$\parallel e_\Omega(0) \parallel^2 < \frac{2}{\lambda_M(J)}\frac{k_R}{\epsilon^2}(2 - \Psi_R(R(0), R_d(0))) \tag{3-22}$$

where $\lambda_M(\cdot)$ denotes the maximum eigenvalue. The domain of attraction is defined by Equations 3-21 and 3-22.
Furthermore, there exist constants $\alpha_R, \beta_R > 0$ such that

$$\Psi_R(R(t), R_d(t)) \leq min\left\{2, \alpha_R e^{-\beta_R t}\right\} \tag{3-23}$$

$$J\dot{e}_\Omega = -\frac{1}{\epsilon^2}k_R e_R - \frac{1}{\epsilon}k_\Omega e_\Omega \tag{3-24}$$

3-2-2 Load Attitude Tracking

The Load Attitude Controlled Mode tracks a desired load attitude q_d by calculating a command signal for the QR attitude, defined as

$$R_c = \left[b_{1c}; b_{3c} \times b_{1c}; b_{3c}\right] \tag{3-25}$$

where $b_{3c} \in \mathbb{S}^2$ is defined by

$$b_{3c} = \frac{F}{||F||} \tag{3-26}$$

Such that F in Equation 3-26 is defined by a normal component F_n, F_{pd} and F_{ff}

$$F = F_n - F_{pd} - F_{ff} \tag{3-27}$$

Control forces for a system evolving on $\mathbb{S}^2$, are derived in [?]. This results in a proportional-derivative force F_{pd} and a feed forward force F_{ff}, that are functions of Equations 3-9 and 3-10. The following terms are obtained

$$\begin{aligned} F_{pd} &= -k_P \hat{q}^2 q_d - k_D(\dot{q} - (q_d \times \dot{q}_d \times q) \\ &= -k_q e_q - k_\omega e_{\dot{q}} \end{aligned} \tag{3-28}$$

$$F_{ff} = m_Q L \langle\langle q, q_d \times \dot{q}_d \rangle\rangle_{\mathbb{R}^3}(q \times \dot{q}) + m_Q L(q_d \times \ddot{q}_d) \times q \tag{3-29}$$

The unit vector b_{1c} is defined as [?]

$$b_{1c} = -\frac{1}{||b_{3c} \times b_{1d}||}(b_{3c} \times (b_{3c} \times b_{1d})) \tag{3-30}$$

where $b_{1d} \in \mathbb{S}^2$ is chosen, not parallel to b_{3c}. The total upward thrust is defined as

$$f = F \cdot Re_3 \tag{3-31}$$

It is proven in [?] 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

$$\Psi_q(q(0), q_d(0)) < 2 \tag{3-32}$$

$$\| e_{\dot{q}}(0) \|^2 < \frac{2}{m_Q L} k_R(2 - \Psi_q(q(0), q_d(0))) \tag{3-33}$$

The domain of attraction is defined by Equations 3-21, 3-22, 3-32 and 3-33. Furthermore, there exist constants $\alpha_q, \beta_q > 0$ such that

$$\Psi_q(q(t), q_d(t)) \leq min\left\{2, \alpha_q e^{-\beta_q t}\right\} \tag{3-34}$$

3-2-3 Load Position Tracking

Tracks load position reference. Outputs load attitude reference.

$$q_c = -\frac{A}{||A||} \tag{3-35}$$

where

$$A = -k_x e_x - k_v e_v + (m_Q + m_L)(\ddot{x}_{L,d} + g e_3) + m_Q L(\dot{q} \cdot \dot{q})q \tag{3-36}$$

with $e_x = x_L - x_{L,d}$ and $e_v = \dot{x}_L - \dot{x}_{L,d}$. Furthermore, F_n is redefined as

$$F_n = (A \cdot q)q \tag{3-37}$$

It is proven in [?] 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 stable, if the initial conditions satisfy

$$\Psi_q(q(0), q_c(0)) < \psi_1 < 1 \tag{3-38}$$

$$\| e_x(0) \|^2 < e_{x_{max}} \tag{3-39}$$

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

The domain of attraction is defined by Equations 3-21, 3-22, 3-38 and the following equation

$$\| e_{\dot{q}}(0) \|^2 < \frac{2}{m_Q L} k_q(\psi_1 - \Psi_q(q(0), q_d(0))) \tag{3-40}$$

Furthermore, there exist constants $\alpha_q, \beta_q > 0$ such that

$$\Psi_q(q(t), q_d(t)) \leq min\left\{2, \alpha_q e^{-\beta_q t}\right\} \tag{3-41}$$

3-3 Stability Analysis

Normally Lyapunov Analysis is

Lyapunov Analysis on SO3 x R3 and S2 x R3

Closed-loop full-attitude dynamics evolve on the non-Euclidean manifold $SO(3) \times \mathbb{R}^3$. Since these manifolds are locally Euclidean, local stability properties of a closed-loop equilibrium solution can be assessed using standard Lyapunov methods. In addition, the LaSalle invariance result and related Lyapunov results apply to closed-loop vector fields defined on these manifolds.

However, since the manifolds $SO(3)$ and $\mathbb{S}^2$ are compact, the radial unboundedness assumption cannot be satisfied; consequently, global asymptotic stability cannot follow from a Lyapunov analysis on Euclidean spaces [40], and therefore must be analyzed in alternative ways [19]–[23].[**?**, p.43]

[40]: [19]: [20]: [21]: [22]: [23]:

[**?**] summarizes global results on attitude control and stabilization for a rigid body using continuous time- invariant feedback. The analysis uses methods of geometric mechanics based on the geometry of the special orthogonal group $SO(3)$ and the two-sphere $\mathbb{S}^2$.

****

Bart: Ok, but what are you doing with this? Does this relate to backstepping?
Nam: Dit gaat over analyse die je normaal doet mbv Lyapunov om de stabiliteit aan te tonen.

****

****

Bart: Does this include a part about tuning of the controller?
Nam: De stabiliteit hangt wel samen met de control parameters. De keuze hiervan is op dit moment arbitrair. De 'juiste' gains kiezen is wellicht zoals eerder besproken overbodig

****

Summary

In this chapter the control design based on Nonlinear Geometric Control was discussed.

The main difference with other control techniques is that the tracking errors are also defined on manifolds, which allows the design of almost global defined controllers.

Stability analysis is different from a Lyapunov analysis on Euclidean spaces.

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 potential 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 trajectories that create situations with different challenges. It is discussed what could be expected from these experiments, and the differences in performance of the controllers is discussed.

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.

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, load control is only possible in means of minimizing the load swing. This fact illustrates an important difference between the use of a linear and a nonlinear model.

The experiments describe a desired load trajectory $x_{L,d}(t)$, which is required to be smooth for Geometric Control, such that feed forward terms can be generated and implemented. In this thesis the desired load paths are generated by hand, and the required velocity and acceleration is calculated by a command filter, of which the details are described 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 is approximately a cable length above the predefined desired load position. Note that this will not allow a direct comparison of the load trajectory tracking, nevertheless this will illustrate the main differences between the controllers with the same purpose of load transportation, but with a different approach.

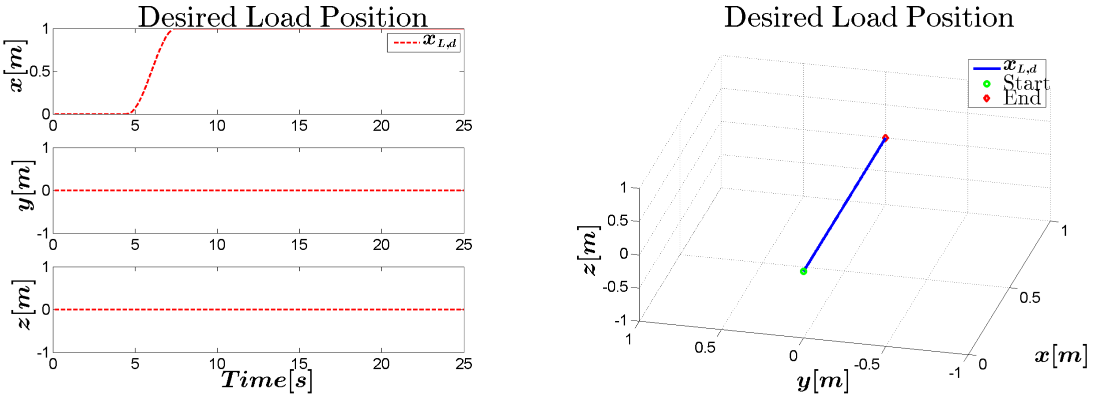
4-2 Trajectories

What observations can be made in order to adapt the controller properties that improve performance of the test cases. Description of tests that apply on all cases.

4-2-1 Case A

In this case a smooth step-like trajectory is generated to transport the load from a starting position along the direction of the x-asis to the final position. In a regular step function the system is subjected to a sudden input. The stability of the system can be investigated by observing whether it is able to reach a stationary final state, and how fast this can be reached. It can be seen whether the system responds with an overshoot and how fast the response is, when the controller tries to track the trajectory.

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



(a) (b)

Figure 4-1: Desired Load Position Case A

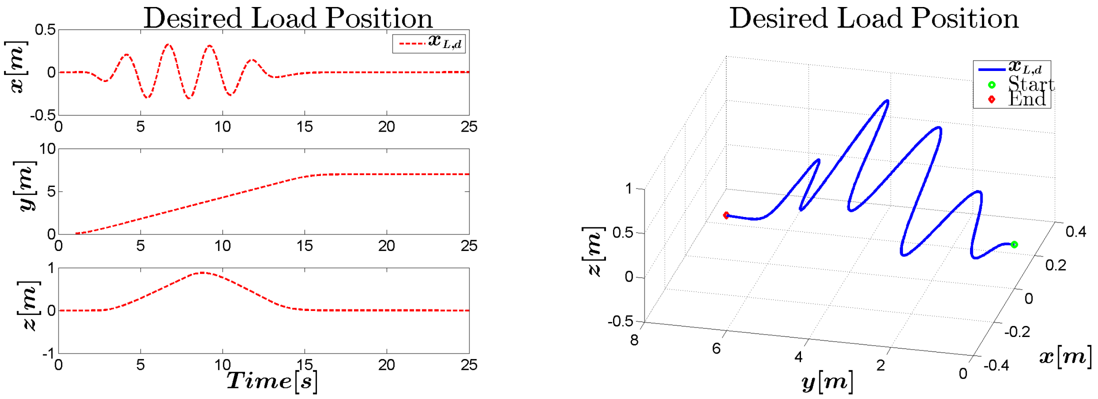
4-2-2 Case B

PLANNING: make case to test limits on QR angles while tracking load trajectory. Is nonlinear GC useful for such aggressive maneuvers?

4-2-3 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 4-2 shows the desired trajectory over time, and a three dimensional representation.



(a)

(b)

Figure 4-2: 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 CM 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

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 4-3

Figure 4-3: 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-2.

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-2.

Geometric Control The chosen controller gains in Equations 3-18,3-25,3-35 can be found in Table 4-2.

Gain	Value
k_R	
k_Ω	
k_q	
k_ω	
k_x	
k_v	

Table 4-2: Controller Gains

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 an outer loop. In the earlier presented control design, 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 require high computational costs, these values can be obtained with the use of a Command Filter, which is explained in more detail in [**?**].

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. See Figure **??** and A-2. 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 $x_3 = \ddot{x}_c$.

Chapter 5

Results

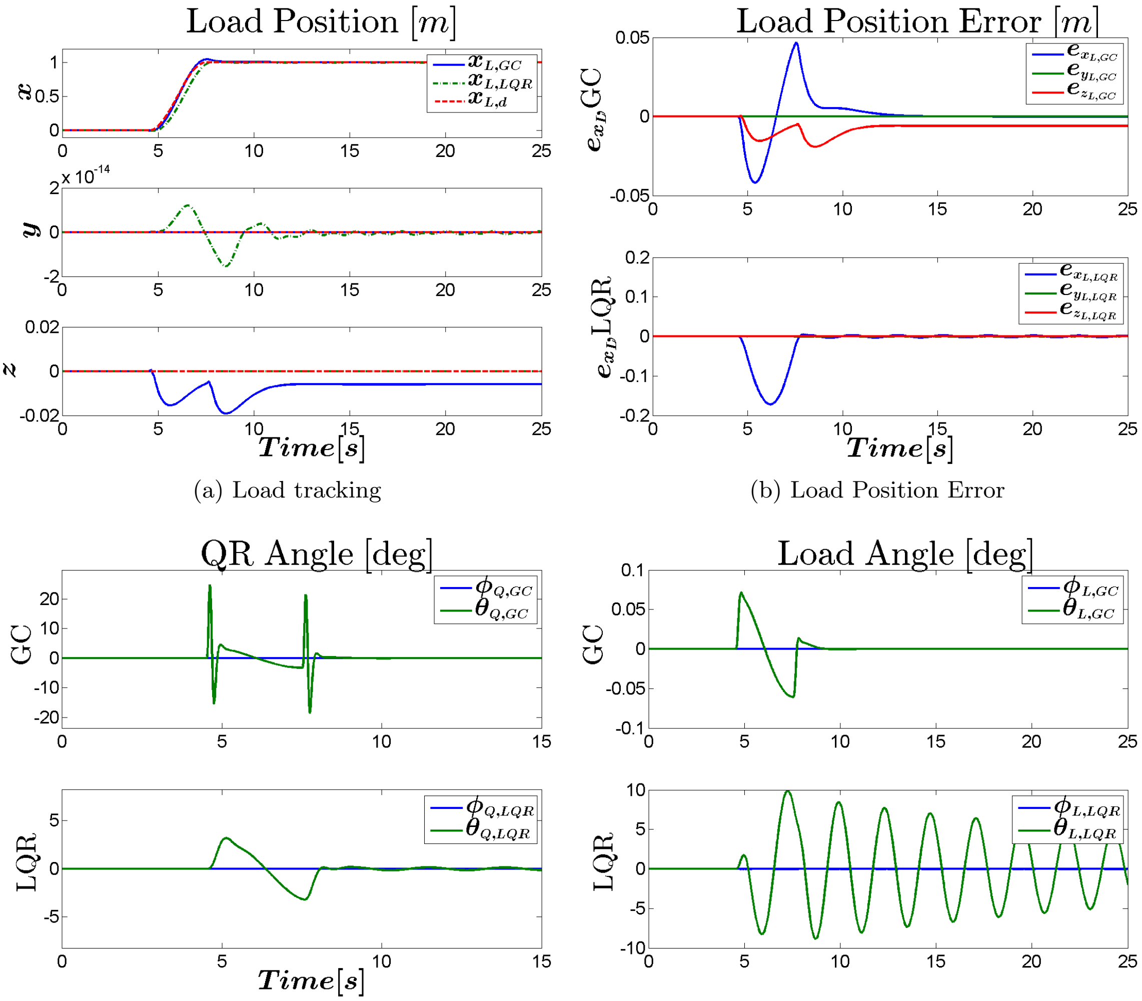
The sections in this chapter discuss the results that are obtained from the load trajectory tracking experiments.

In Figures 5-2, 5-4 and 5-6 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.

5-1 Case A

Figure 5-1a shows the load position along the desired load position $x_{L,d}$ for both control approaches.
Figure 5-1b shows the load position error for both control approaches.

Figure 5-1c shows the QR attitude with respect to $\{\mathcal{I}\}$.
In Figure 5-1d 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 5-1: Controller Comparison Case A

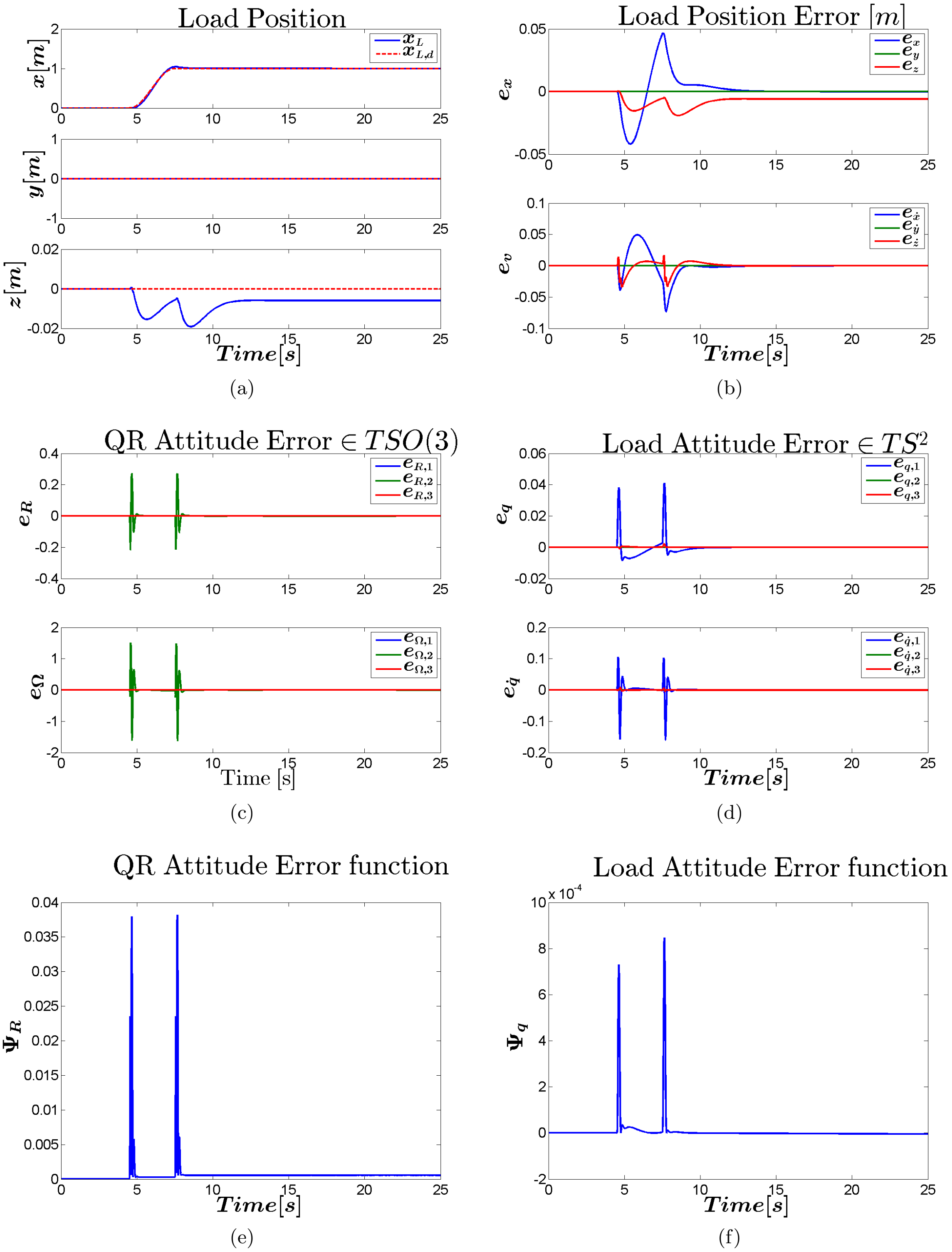
The desired and actual load trajectory, and the position error are shown in Figure 5-2a and Figure 5-2b, respectively. From this can be seen that a small steady state error remains in the z-direction. However, $(e_x, e_v) = (0, 0)$ is exponentially attractive.

Figure 5-2c and 5-2d 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 5-2e and 5-2f show the tracking error functions of the QR and load, respectively. Observations: there exist constants $\alpha_q, \beta_q > 0$ such that

$$\Psi_q(q(t), q_d(t)) \leq min\left\{2, \alpha_q e^{-\beta_q t}\right\} \tag{5-1}$$



(a)

(b)

(c)

(d)

(e)

(f)

Figure 5-2: Results Nonlinear Geometric Control Case A

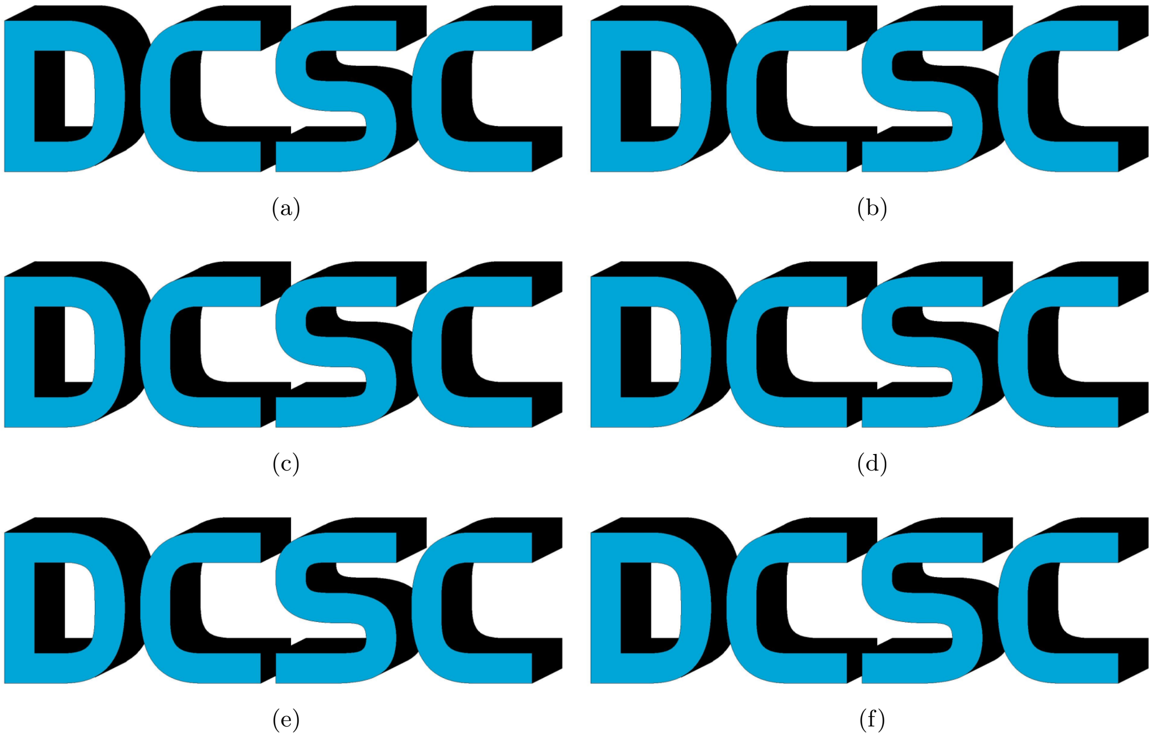
5-2 Case B

Figure 5-3a shows the load position along the desired load position $x_{L,d}$ of both controllers.

(a) Load tracking

(b) Load Position Error

Figure 5-3: Controller Comparison Case B



(a) (b)

(c) (d)

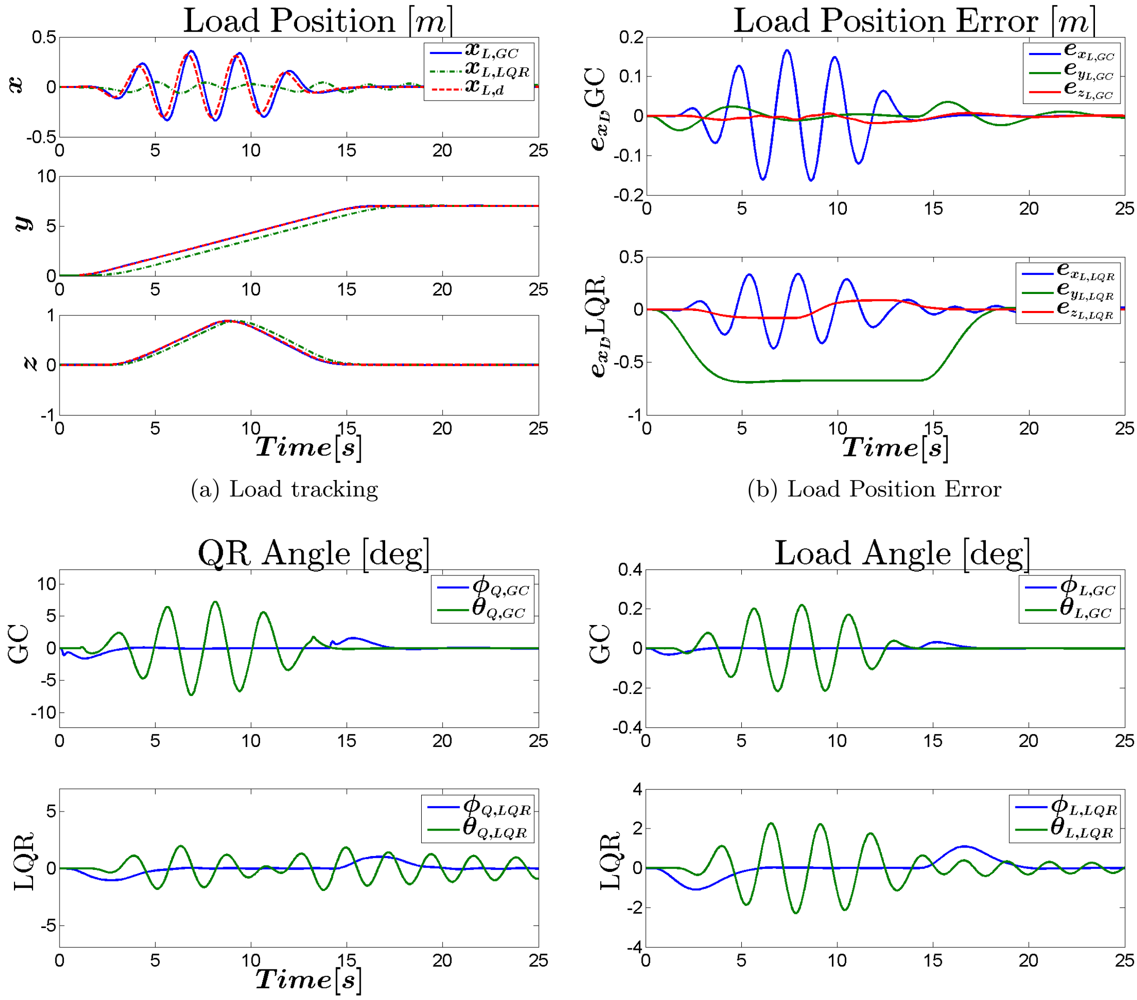
(e) (f)

Figure 5-4: Results Nonlinear Geometric Control Case B

5-3 Case C

Figure 5-5a shows the load position along the desired load position $x_{L,d}$ of both controllers.
Figure 5-5b 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 5-5c shows the QR attitude with respect to $\{\mathcal{I}\}$.
In Figure 5-5d 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 5-5: Controller Comparison Case C

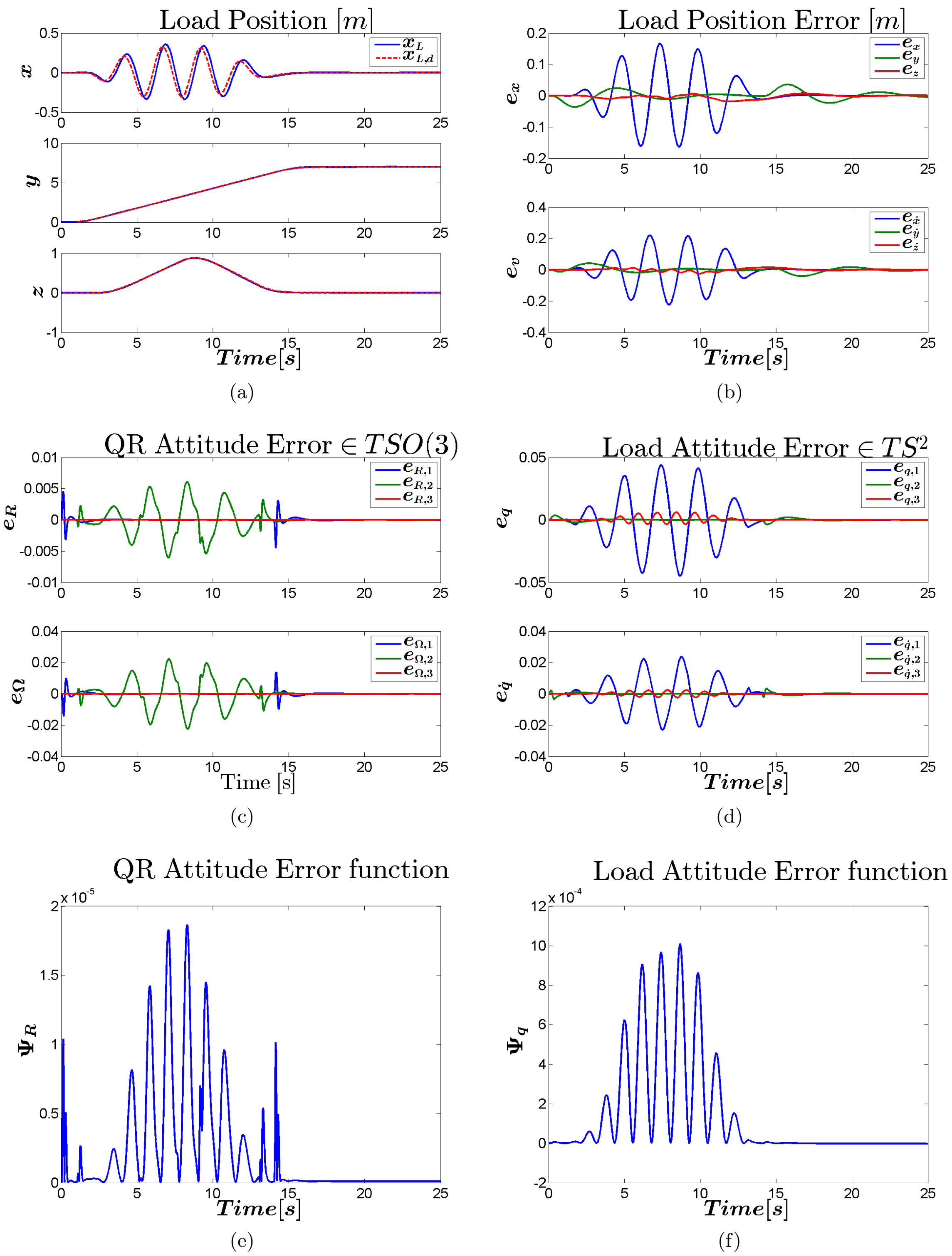
While tracking the required QR attitude, which tilts the QR to reach the desired velocities in the right direction, it can be seen that the system has difficulties to also maintain the desired height, which can be explained by the fact that the total force will not point upwards if the QR is tilted. Despite the fact that the QR is moving from side to side, the upward force is still controlled to track the desired height.

Figure 5-6a shows the desired load position, and Figure 5-6b shows that the error is mainly the overshoot in the x-direction, due to the fast desired swinging motion.

Figure 5-6c and 5-6d 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 5-6e and 5-6f show the tracking error functions of the QR and load, respectively.
Observations: there exist constants $\alpha_q, \beta_q > 0$ such that

$$\Psi_q(q(t), q_d(t)) \leq min\left\{2, \alpha_q e^{-\beta_q t}\right\} \tag{5-2}$$



(a)

(b)

(c)

(d)

(e)

(f)

Figure 5-6: Results Nonlinear Geometric Control Case C

5-4 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 6

Conclusions and Future Work

6-1 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

6-2 Recommendations for Future Work

6-2-1 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 [?].

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, [?] 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

6-2-2 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 [?] and applied in [?, ?]. 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.

6-2-3 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 [?]. In [?, ?, ?] 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 LQR controller

A-2-1 Modeling

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

$$\dot{\mathbf{x}} = A\mathbf{x} + Bu \tag{A-3}$$

$$y = C\mathbf{x} + Du \tag{A-4}$$

with the following state- and input vectors

$$\begin{aligned} \mathbf{x} &= \begin{bmatrix} x & y & z & \phi & \theta & \psi & \phi_L & \theta_L & \dot{x} & \dot{y} & \dot{z} & \dot{\phi} & \dot{\theta} & \dot{\psi} & \dot{\phi}_L & \dot{\theta}_L \end{bmatrix}^T \\ u &= \begin{bmatrix} f & M_\phi & M_\theta & M_\psi \end{bmatrix}^T \end{aligned} \tag{A-5}$$

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

$$\dot{\mathbf{x}} \to \dot{\mathbf{x}}_0 + \delta\dot{\mathbf{x}}, \quad \mathbf{x} \to \mathbf{x}_0 + \delta\mathbf{x}, \quad u \to u_0 + \delta u \tag{A-6}$$

$$\begin{aligned} \mathbf{x}(0) &= \mathbf{0} \\ u(0) &= \begin{bmatrix} g(m_Q + m_L) & 0 & 0 & 0 \end{bmatrix}^T \end{aligned} \tag{A-7}$$

The linearized equations of motion are rearranged into Equation A-8 and substituted in Equation A-3.

$$\begin{bmatrix} content... \end{bmatrix} \begin{bmatrix} \delta\ddot{x} \\ \delta\ddot{y} \\ \delta\ddot{z} \\ \delta\ddot{\phi} \\ \delta\ddot{\theta} \\ \delta\ddot{\psi} \\ \delta\ddot{\phi}_L \\ \delta\ddot{\theta}_L \end{bmatrix} + \begin{bmatrix} content... \end{bmatrix} \begin{bmatrix} \delta x \\ \delta y \\ \delta z \\ \delta\phi \\ \delta\theta \\ \delta\psi \\ \delta\phi_L \\ \delta\theta_L \end{bmatrix} = \begin{bmatrix} content... \end{bmatrix} \begin{bmatrix} \delta u_1 \\ \delta u_2 \\ \delta u_3 \\ \delta u_4 \end{bmatrix} \tag{A-8}$$

```
LQRA =
```

```
LQRB =
```

The tuning parameters of the LQR controller a chosen as follows

$$\begin{aligned} Q &= diag\begin{pmatrix}10 & 10 & 100, & 1 & 1 & 1, & 1 & 1, & 1 & 1 & 1, & 1 & 1 & 1, & 1 & 1\end{pmatrix} \\ R &= diag\begin{pmatrix}0.044, & 1.56, & 1.56, & 1.56\end{pmatrix} \end{aligned} \tag{A-9}$$

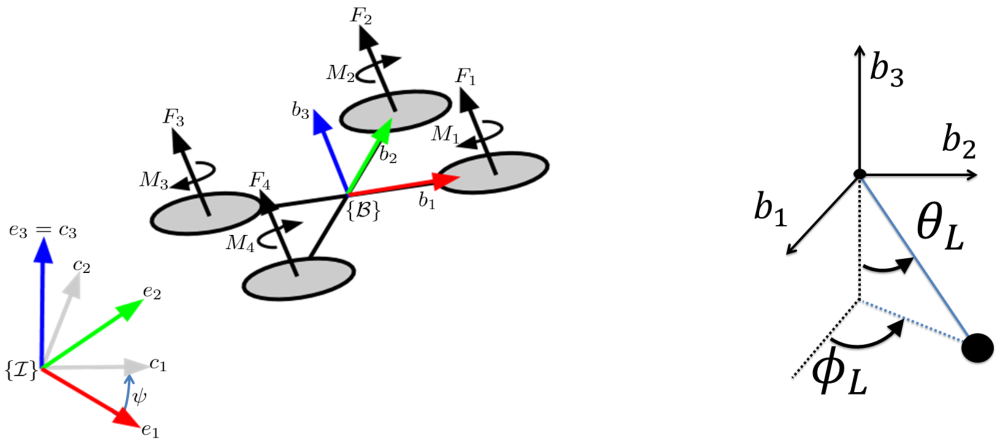
`Matlab` command `lqr(LQRA,LQRB,Q,R)` generates the following gain matrix K

```
K =
```

A-3 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.

$$\begin{aligned} \dot{x}_Q &= v_Q \\ m_Q \dot{v}_Q &= fRe_3 - m_Q g e_3 - Tq \\ \dot{x}_L &= v_L \\ m_L \dot{v}_L &= -m_L g e_3 + Tq \end{aligned} \tag{A-10}$$

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 [**?**]

$$R_{ZXY}(\phi,\theta,\psi) = \begin{bmatrix} c_\psi c_\theta - s_\phi s_\psi s_\theta & -c_\phi s_\psi & c_\psi s_\theta + c_\theta s_\phi s_\psi \\ c_\theta s_\psi + c_\psi s_\phi s_\theta & c_\phi c_\psi & s_\psi s_\theta - c_\psi c_\theta s_\phi \\ -c_\phi s_\theta & s_\phi & c_\phi c_\theta \end{bmatrix} \tag{A-11}$$

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

$$x_L = x_Q + qL \tag{A-12}$$

where

$$q = \begin{bmatrix} s_{\theta_L} c_{\phi_L} \\ s_{\theta_L} s_{\phi_L} \\ -c_{\theta_L} \end{bmatrix} \tag{A-13}$$

Differentiating Equation A-12 and A-13 gives

$$\begin{aligned}
\ddot{x}_L &= \ddot{x}_Q + \ddot{q}L \\
\ddot{q} &= \begin{bmatrix} \ddot{\theta}_L c_{\theta_L} c_{\phi_L} - \ddot{\phi}_L s_{\theta_L} s_{\phi_L} - \dot{\phi}_L^2 s_{\theta_L} c_{\phi_L} - \dot{\theta}_L^2 s_{\theta_L} c_{\phi_L} - 2\dot{\theta}_L \dot{\phi}_L c_{\theta_L} s_{\phi_L} \\ \ddot{\theta}_L c_{\theta_L} s_{\phi_L} + \ddot{\phi}_L s_{\theta_L} c_{\phi_L} - \dot{\phi}_L^2 s_{\theta_L} s_{\phi_L} - \dot{\theta}_L^2 s_{\theta_L} s_{\phi_L} + 2\dot{\theta}_L \dot{\phi}_L c_{\theta_L} c_{\phi_L} \\ \ddot{\theta}_L s_{\theta_L} + \dot{\theta}_L^2 c_{\theta_L} \end{bmatrix}
\end{aligned} \tag{A-14}$$

$$\begin{aligned}
\ddot{x}_Q &= \frac{1}{m_Q}(f(c_\psi s_\theta + c_\theta s_\phi s_\psi) - T s_{\theta_L} c_{\psi_L}) \\
\ddot{y}_Q &= \frac{1}{m_Q}(f(s_\psi s_\theta - c_\psi c_\theta s_\phi) - T s_{\theta_L} s_{\psi_L}) \\
\ddot{z}_Q &= \frac{1}{m_Q}(f(c_\phi c_\theta) - T c_{\theta_L}) - g
\end{aligned} \tag{A-15}$$

A-4 Derivation Error dynamics

A-4-1 Quadrotor Attitude

From the angular velocity tracking error e_Ω follows

$$\begin{aligned}
e_\Omega &= \Omega - R^T R_d \Omega_d, \\
\hat{e}_\Omega &= \hat{\Omega} - R^T R_d \hat{\Omega}_d R_d^T R
\end{aligned} \tag{A-16}$$

The attitude tracking and its time derivative is given by

$$\begin{aligned}
e_R &= \frac{1}{2}(R_d^T R - R^T R_d)^\vee, \\
\dot{e}_R &= \frac{1}{2}(\dot{R}_d^T R + R_d^T \dot{R} - \dot{R}^T R_d - R^T \dot{R}_d)^\vee \\
&= \frac{1}{2}((R_d \hat{\Omega}_d)^T R + R_d^T (R\hat{\Omega}) - (R\hat{\Omega})^T R_d - R^T (R_d \hat{\Omega}_d))^\vee \\
&= \frac{1}{2}(-\hat{\Omega}_d R_d^T R + R_d^T R \hat{\Omega} + \hat{\Omega} R^T R_d - R^T R_d \hat{\Omega}_d)^\vee \\
&= \frac{1}{2}(R_d^T R(\hat{\Omega} - R^T R_d \hat{\Omega}_d R_d^T R) + (\hat{\Omega} - R^T R_d \hat{\Omega}_d R_d^T R) R^T R_d)^\vee
\end{aligned} \tag{A-17}$$

Equation A-16 is substituted and Equation 2-6 is used to rewrite the time derivative of e_R as follows

$$\begin{aligned}
\dot{e}_R &= \frac{1}{2}(R_d^T R \hat{e}_\Omega + \hat{e}_\Omega R^T R_d)^\vee \\
&= \frac{1}{2}(tr[R^T R_d]I - R^T R_d)e_\Omega \equiv C(R_d^T R)e_\Omega
\end{aligned} \tag{A-18}$$

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

where $\hat{\Omega}_d \Omega_d = \Omega_d \times \Omega_d = 0$.

A-5 Figures

Figure A-2: Simulink Command Filter

Nomenclature

ϵ	Tuning parameter to enable rapid exponential convergence of e_R, e_Ω
$\lambda_M(\cdot)$	Maximum eigenvalue
$\mathbb{I} \in \mathbb{R}^{3\times 3}$	Inertia tensor of QR
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
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
CM	Center of Mass
DOF	Degree of Freedom
PID	Proportional-Integral-Derivative (Controller)
MPC	Model Predictive Control
LQR	Linear Quadratic Regulator
QP	Quadratic Programming