Implementation of an Explicit Reference Governor in a Nonlinear Quadcopter System

A Thesis Presented
by

Oliver de Quadros

to

The Department of Electrical and Computer Engineering

in partial fulfillment of the requirements
for the degree of

Master of Science

in

Electrical and Computer Engineering

Northeastern University
Boston, Massachusetts

December 2015
To my family and friends who have made it all possible through your unconditional support and encouragement.
Contents

List of Figures iv
List of Tables v
Acknowledgments vi
Abstract of the Thesis vii
Introduction 1
Current State of Quadcopter Control 3
  1.1 Current Solutions 4
    1.1.1 Model Predictive Control 4
    1.1.2 Command Governor 4
    1.1.3 Reference Governor 5
Explicit Reference Governor Theory 6
  2.1 System and Problem Description 6
  2.2 General Explicit Reference Governor 7
  2.3 Explicit Reference Governor Applied to Linear System 10
Qball X4 Model 12
  3.1 Actuator Dynamics 13
  3.2 Roll/Pitch Model 14
  3.3 Height Model 15
  3.4 X-Y Position Model 16
  3.5 Yaw Model 16
  3.6 Qball X4 Linear Model, Command inputs, and Parameters. 17
  3.7 Quadcopter System Identification 19
Results 22
Conclusion and Future Work 27
  5.1 Future Work 27
Appendix A 28
Bibliography 30
List of Figures

Figure 1   Quanser Qball X4                             2
Figure 2   Explicit Reference Governor Scheme [4]     9
Figure 3   Qball X4 axes and sign conventions        12
Figure 4   model of the yaw axis with propeller direction shown 13
Figure 5   Model of the roll/pitch axis              14
Figure 6   Step Response of Qball X4 simulation Without Saturation 19
Figure 7   Unit Step in X Simulation with Saturation  22
Figure 8   Unit Step in X Simulation without Saturation 22
Figure 9   Unit Step in X Simulation without Saturation with ERG 23
Figure 10  Unit Step in X Simulation with ERG, internal states, and 14 state model 23
Figure 11  Step with FV 5 in X Simulation with Saturation 24
Figure 12  Step with FV 5 in X Simulation without Saturation 24
Figure 13  Step with FV 5 in X Simulation with ERG, internal states, and 14 state model 24
Figure 14  Step with FV 5 in X Simulation without Saturation with ERG 24
Figure 15  Unit step in Y Simulation with Saturation 25
Figure 16  Unit step in Y Simulation without Saturation 25
Figure 17  Unit Step in Y Simulation without Saturation with ERG 26
Figure 18: x response of 11 state estimate           28
Figure 19: pitch response of 11 state estimate      28
Figure 20: pitch response of 12 state estimate      28
Figure 21: x response of 12 state estimate          28
Figure 22: x response of 14 state estimate          29
Figure 23: pitch response of 14 state estimate      29
Figure 24: x response of 15 state estimate          29
Figure 25: pitch response of 15 state estimate      29
List of Tables

Table 1: Computation Time and Settling Time Comparisons 5
Table 2 Operations Performed by the ERG 11
Table 3 Table of Physical Parameters of the Qball X4 18
Acknowledgments

I wish to thank Dr. Mario Sznaier for all the support, help, and understanding he has provided to me during the course of this thesis.

I wish to also thank Dr. Bahram Shafai and Dr. Rifat Sipahi for agreeing to take part in my committee at such short notice.

To all the friends and family who have made this possible just know I couldn’t have done it without your support.
Abstract of the Thesis

Implementation of an Explicit Reference Governor in a Nonlinear Quadcopter System

by

Oliver de Quadros

Master of Science in Electrical and Computer Engineering

Northeastern University, December 2015

Dr. Mario Sznaier, Adviser

Quadcopters are inherently nonlinear systems but can be linearized over a small range of pitch (θ) and roll (φ) angles and therefore are able to use standard linear feedback controllers to minimize steady state errors and improve tracking performance. However, these controllers are usually subjected to saturation in order to keep the system in the linear region. The nonlinearities arising from these saturations are generally not acceptable since they can cause errors which do not converge and long settling times. In this thesis, we discuss and implement an Explicit Reference Governor (ERG) that attenuates the XY-references to the Quadcopter when constraint violations may occur in order to maintain stability. This is done by creating upper bounds on the value of the Lyapunov function from system constraints. These bounds are then enforced by changing the velocity of the reference command. This ERG is preferable due to its low computation times and because it provides a consistent closed-form control law regardless of system. The theory of the ERG is presented along with the nonlinear and linear models of the Quadcopter in which the constraints of the system are established. The ERG is then implemented in a simulation platform of the Quadcopter and is tested and compared to the Quadcopter model with and without saturation. The implemented ERG improved the settling time of the system by 15% compared to the saturated system and was able to stay within the allowable limits unlike the unsaturated system.
Introduction

Quadcopters and other unmanned aerial vehicles (UAVs) are becoming more generally used in public dynamic environments where the surroundings are changing at high rates and have tight turns. In these environments, UAVs need to react quickly to any changes in the environment by minimizing computation times. Most quadcopters nowadays, such as the Qball X4 from Quanser, are modeled to be linear and have implemented PID controllers and feedback control in order to minimize errors and improve tracking performance. This means that the quadcopters will be able to track reference commands such as position and orientation closely with minimal error when they are within the limits which keep the model linear \[1\]. In order to keep these commands within certain constraints, the output of the closed loop systems are usually subjected to saturations \[1\]. The nonlinearity effect of these saturations due to large outputs may cause the error of the system to diverge or cause instability in the system or the system may take too long to settle \[1\].

Some traditional strategies to deal with these problems is to reduce the performance of the existing linear controllers and putting limits on the reference commands\[1\]. This however has many negative effects such as increasing oscillations, increasing settling times, and decreasing agility of the system.

In order to tackle these problems, several nonlinear solutions have become more and more popular. One popular choice is to replace the linear controllers to fit into the model predictive control framework \[2\]. However if the system already responds nominally to a small reference with the existing linear controller it might also be prudent to augment the closed loop system with an additional controller which handles constraint enforcement. Examples of such systems are reference/command governors, anti-windup compensation, and Lyapunov controllers with barrier functions \[3\].

In this thesis we will be focusing on implementing an Explicit Reference Governor (ERG) developed by Garone and Nicotra in \[4\] on the Qball X4. A reference governor is a controller that
modifies the reference command given to the pre-compensated closed loop system in order to enforce state and controller constraints based on the current state of the closed-loop system and the velocity of the reference [4]. The reference governor essentially attempt to produce a modified reference command as close to the original reference command as possible without violating the constraints [3]. Implementation of the ERG was done in Matlab Simulink and testing was carried out using the Quanser Matlab toolbox Quarc that allows for realistic graphical testing of the system. The results were then compared to the saturated and unsaturated closed loop responses of the Qball X4 system.

The thesis is outlined as follows: Chapter 1 reviews some of the different solutions currently available and justifies the use of the ERG. Chapter 2 presents the theory of the implemented ERG for the general and the case as it applies to linear systems. Chapter 3 presents the modeling of the Qball X4 and its closed-loop characterization. Chapter 4 outlines how the ERG was implemented on the Qball X4. Finally, Chapter 5 presents the results from testing of the Qball X4 with the ERG before presenting the conclusions and future work in Chapter 6.

Figure 1  Quanser Qball X4
Chapter 1

Current State of Quadcopter Control

The controls laboratory at Northeastern University has several drones such as the Qball-X4 and four Parrot AR Drones that are being used in research. These drones, as of this writing, are being used in various realms of research such as obstacle avoidance, object detection, and surveillance. The research itself seems very promising but the drones have limited uses and are only good in very controlled and static environments. They have slow response times when being autonomously controlled since the Qball X4 has many saturation points within the onboard controller to dampen the reference signals being provided to the Qball X4 so they do not pass a certain limit. This negatively affects performance by introducing delays. Therefore, implementing a nonlinear reference governor into these drones will open up several avenues of research in the future by preemptively attenuating the reference signals in order to eventually increasing the online speed of operations. The reference governor that we implement in this paper drastically decreases computation time compared to Model Predictive Control (MPC) and therefore would be an asset to the research pursuits of the controls lab.

In this section, we will discuss some different methods currently being employed in the control of quadcopters and other Unmanned Aerial Vehicles UAVs with fast dynamics and introduce the Explicit Reference Governor we have chosen to implement.
1.1 Current Solutions

1.1.1 Model Predictive Control

Model Predictive Control (MPC) is a popular choice for control and is greatly studied in many fields mainly the petro-chemical industries[5], UAV control [6], and process control. There have been several reviews published by several authors on MPC which describe the process in great detail [7], [5], [8], [9], [10]. The basic idea behind MPC is the idea of receding horizons [11] and it is an optimal control based algorithm. In order to optimize the future plant output to the reference points, the algorithm repeatedly solves an online optimal control problem [12]. This optimization aims to minimize the cost function subject to constraints over a future horizon, after which the control signal is fed back into the controller and the process is repeated when the new output is available [12]. However, due to its online optimization nature, MPC is generally applied to slow processes that can be applied in real time [4]. In order to overcome this, researchers are developing faster and more efficient MPC algorithms, such as in [13] and [14]. However, creating fast and reliable MPC schemes is still being researched and is an open problem [15].

1.1.2 Command Governor

The command governor is an add-on nonlinear control scheme that has seen a wide variety of applications. Command governors have been proposed in papers such as [16][17][18]. It is similar to the reference governor in that it attempts to change the reference signal by minimizing a cost function subject to constraints [3]. There have been a few variants of the command governor such as the Prioritized Reference Governor[19], the Extended Command Governor [17], and the Hybrid
Command Governor [16]. Command governors are able to provide a larger constrained domain of attraction and faster response times but at the cost of increasing computational complexity [3], but this is not desirable for UAV applications.

1.1.3 Reference Governor

The reference governor is also an add-on scheme for enforcing pointwise in time state and control constraints by modifying the reference command to a well-designed closed loop system [3]. There have been different kinds of reference governors proposed, but their common purpose is to preserve whenever possible the response of the closed loop system [3]. The reference governor is especially attractive because of the simple implementations for both linear and nonlinear systems with disturbances and parameter uncertainties [3]. Many different reference governors have been proposed and designed including some based on model linearization [20] and nonlinear models [21][22][23], robust reference governors [24], and reduced order reference governors [25]. In this work we will be focusing on the Explicit Reference Governor developed by Emanuel Garone and Marco Nicotra as it is recently developed and shows good performance in [4]. The main advantages shown in the paper over regular reference governors and model predictive control is that the computation time for each iteration is much smaller. This is demonstrated in Table 1 taken from [4].

<table>
<thead>
<tr>
<th></th>
<th>Settling Time [s]</th>
<th>CPU Time [μs/iteration]</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPC</td>
<td>0.26</td>
<td>329</td>
</tr>
<tr>
<td>Std RG</td>
<td>0.33</td>
<td>189</td>
</tr>
<tr>
<td>ERG ( v \neq 0 )</td>
<td>0.67</td>
<td>3.7</td>
</tr>
<tr>
<td>ERG ( v = 0 )</td>
<td>0.67</td>
<td>3.8</td>
</tr>
</tbody>
</table>

Table 1: Computation Time and Settling Time Comparisons
Chapter 2

Explicit Reference Governor Theory

This section presents the general theory behind the ERG and then goes on to specialize it to linear systems. The ERG is a nonlinear controller that is able to enforce constraints without having to perform any online optimizations by applying the static nature of Lyapunov level sets in order to create a maximum admissible value of the Lyapunov function from state space constraints [4]. These constraints are then imposed on the velocity of the reference signal. Main goals of the ERG is to modify the reference signal so that the modified reference signal stays within the bounded limits.

The equations and conclusions contained here are a summary of the paper written by Emanuele Garone and Marco Nicotra entitled “Explicit Reference Governor for Constrained Nonlinear systems.” For further reading please see [4].

2.1 System and Problem Description

Let

\[ \dot{x} = f(x(t), g(t)) \]  

(1)

describe the closed loop dynamics of a pre-compensated system that is subjected to constraints

\[ c_i \left( x_g(t), g(t) \right) \geq 0, \quad i = 1, \ldots, n_c, \forall t \]  

(2)
We are assuming that the system has been compensated so that for any applied reference \( g(t) \in \mathbb{R}^n \) it is Globally Asymptotically Stable (GAS) where the steady state is given by \( x_g \in \mathbb{R}^n \) and is continuous in \( g \). It is also assumed that the constraints are concave functions of \( g \) implying that allowed steady state references in the set:

\[
\mathcal{G} = \{ g : c_i(x_g, g) \geq 0, \quad i = 1, \ldots, n_c \} \quad (3)
\]

are convex [4].

Assuming this system, the ERG aims to create a reference at every time instant that will not violate the constraints and simultaneously approximates the original reference as closely as possible.

### 2.2 General Explicit Reference Governor

This ERG is based on the observation that the Lyapunov function defines an invariant level set centered in the steady state \( x_g \). Therefore, it is sufficient to manipulate \( g \) so that the invariant level set is wholly contained in the constraints at all times in order to adhere to the constraints. This approach is conservative as seen in Table 3 since the reference signal is attenuated, but this is reasonable in terms of nonlinear systems since the tradeoff is reduced computation times due to no predictions or online optimizations [4].

To describe the basic principle of the ERG we can consider a Lyapunov function satisfying

\[
V(x, x_g) > 0, \forall x \neq x_g \quad \text{and} \quad V(x_g, x_g) = 0 \quad \text{such that [4]}:
\]

\[
\dot{V}(x, x_g) < 0 \quad \text{for} \quad \dot{g} = 0, \forall x \in \mathbb{R}^n, \forall g \in \mathbb{R}^m \quad (4)
\]

Now with this \( V(x, x_g) \), define a set of smooth functions \( \Gamma_i(g) \) such that [4]

\[
V(x, x_g) \leq \Gamma_i(g) \Rightarrow c_i(x_g, g) \geq 0 \quad (5)
\]

\[
\forall \delta > 0, \exists \epsilon > 0 : \quad c_i(x_g, g) \geq \delta \Rightarrow \Gamma_i(g) \geq \epsilon \quad (6)
\]
From equations (4) and (5), for a constant applied reference $g$, the set $\{x: V(x, x_g) \leq \Gamma_i(g)\}$ is an invariant level set completely containing $x_g$ and completely within the $i$-th constraint. Also from (6), whenever $x_g$ is strictly within the constraints, then $\Gamma_i(g) > 0$ [4].

From this the constraints from (2) can be enforced at each time instant by making sure that

$$V(x, x_g) \leq I_j(g)$$

(7)

Where $J = \{argmin_i(\Gamma_i(g))\}$ [4]

If we define a $v(x, g, r)$ such that

$$v(x, x_g, \mu \frac{\partial}{\partial ||g||}) \leq \min_{\mu \in J} \Gamma_i \left( g, \mu \frac{\partial}{\partial ||g||} \right), \quad \forall \mu: 0 \leq \mu \leq v$$

(8)

then if initial conditions satisfy $V(x(0), x_{g(0)}) \leq I_j(g(0))$ and if $||\dot{g}||$ is such that

$$\begin{cases} 
0 \leq ||\dot{g}|| \leq v & \text{if } V(x, x_g) = I_j(g) \\
0 \leq ||\dot{g}|| < \infty & \text{if } V(x, x_g) < I_j(g)
\end{cases}$$

(9)

constraints (2) are never violated [4].

Additionally, if $I_j(g(0)) > 0$, and $g(t)$ is such that

$$\dot{g} = v(x, g, r) \frac{r-g}{||r-g||}$$

(10)

with $v(x, g, r)$ satisfying

$$\begin{cases} 
||v(x, g, r)|| = 0 & \text{if } g = r \\
||v(x, g, r)|| \leq v & \text{if } g \neq r \land V(x, x_g) = I_j(g) \\
||v(x, g, r)|| > 0 & \text{if } g \neq r \land V(x, x_g) < I_j(g)
\end{cases}$$

(11)

and $v \geq 0$ satisfying (8), then for any reference the constraints are never violated and for any constant reference $r(t) \in G$, $g(t)$ asymptotically tends to $r(t)$ [4].
In order to ensure integrability of the differential equations, the ERG is designed such that $\dot{g}$ is continuous as follows[4]:

$$\gamma(x, g, r) = [\phi(x, g) + \nu(x, g, r)]\sigma(g, r) \quad (12)$$

where

- $\phi(x, g)$ is a proportional non-negative feedback term that ensures continuity in $\Gamma_1(g) - V(x, x_g)$:

$$\phi(x, g) = \kappa \left( \Gamma_1(g) - V(x, x_g) \right) \quad (13)$$

with $\kappa > 0$ an arbitrary large scalar. Due to the possibility of numerical errors in implementation, $\phi(x, g)$ should also have a lower bound of zero to guarantee $\phi(x, g) \geq 0$.

- $\nu(x, g, r)$ is a non-negative feedforward term

- $\sigma(g, r)$ is a term ensuring continuity in $r - g$

$$\sigma(g, r) = \text{sat}_1 \left( \frac{\|r - g\|}{\eta} \right) \quad (14)$$

with $\eta > 0$ being an arbitrary small scalar.

This structure is proposed in Figure 2.
2.3 Explicit Reference Governor Applied to Linear System

It will now be shown how to specialize and implement the ERG in linear systems. Consider a pre-stabilized linear system of the form

\[ \dot{x} = Ax(t) + Bg(t) \]  

subjected to the affine constraints

\[ c_{xi}^T x(t) + c_{gi}^T g(t) + d_i \geq 0, \quad \forall t, i = 1, \ldots, n_c \]  

Given a constant \( g \), the steady state is given by [4]

\[ x_g = -A^{-1}Bg. \]  

There also exist a \( P > 0 \) given any \( Q > 0 \) such that

\[ A^T P + PA = -Q \]  

so that the Lyapunov equation is given by

\[ V(x, x_g) = (x - x_g)^T P (x - x_g). \]  

Then the maximum \( \Gamma_i(g) \) such that \( V(x, x_g) \leq \Gamma_i(g) \) implies (16) is [4]

\[ \Gamma_i(g) = \frac{(c_{xi}^T x_g + c_{gi}^T + d_i)^2}{c_{xi}^T P^{-1} c_{xi}}. \]

Therefore,

\[ \nu(x, x_g) = \max \left\{ 0, \min_{i \in J} \frac{1}{2} \left( x - x_g \right)^T Q \left( x - x_g \right) \left[ \left( x - x_g \right)^T P + \frac{c_{xi}^T x_g + c_{gi}^T + d_i}{c_{xi}^T P^{-1} c_{xi}} (c_{xi}^T + c_{gi}^T) A^{-1} B \right] \right\} \]

is the largest \( \nu \) satisfying (8) [4].

So in summary, the operations performed by the ERG are collected in Table 2.
<table>
<thead>
<tr>
<th><strong>Formula</strong></th>
<th><strong>Direction Vector</strong></th>
<th>[ \hat{\rho} = \frac{r - g}{|r - g|} ]</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Current Lyapunov</strong></td>
<td>[ V = (x - x_g)^T P (x - x_g) ]</td>
<td></td>
</tr>
<tr>
<td><strong>Distance from Constraints</strong></td>
<td>[ \Gamma_i(g) = \frac{(c_i^T x_g + c_i^T + d_i)^2}{c_{ji}^T P^{-1} c_{ji}} ]</td>
<td></td>
</tr>
<tr>
<td><strong>Feedback term</strong></td>
<td>[ \phi = \kappa \left( \Gamma_j(g) - V(x, x_g) \right) ]</td>
<td></td>
</tr>
<tr>
<td><strong>Feedforward term</strong></td>
<td>[ v(x, x_g) = \max \left{ 0, \min_{\omega \in \Omega} \left{ \frac{1}{2} (x - x_g)^T Q (x - x_g) \right} \right} ]</td>
<td></td>
</tr>
<tr>
<td><strong>Smoothing term</strong></td>
<td>[ \sigma = \min \left( 1, \frac{|r - g|}{\eta} \right) ]</td>
<td></td>
</tr>
</tbody>
</table>

\[ \frac{\dot{x}}{\|x\|} = \left[ \phi(x, g) + v(x, g, r) \right] \sigma(g, r) \hat{\rho} \]

*Table 2 Operations Performed by the ERG*
Chapter 3

Qball X4 Model

In this section we will outline both the nonlinear and linear models of the Qball X4 as it is in [26] and in [16]. For the purposes of this thesis, we will be defining the axes of the body frame of the vehicle as \((x,y,z)\) as they are shown in Figure 2. We will also need to define the global workspace axes as \((X, Y, Z)\) which are defined in the same orientation as when the Qball X4 is sitting upright on the ground. Along with these axes, we need to define the roll, pitch and yaw angles of rotation about the body frame axes that are also shown in Figure 3. The Qball X4 is a quadcopter, which means it has four rotors. Each rotor is made up of a Park-400 Brushless DC motor onto which a 10-inch propeller is fitted. The motors are mounted along the X and Y axes of the frame. The basic setup of the rotors is shown in Figure 2. As can be seen directly opposing rotors are spinning in the same direction whereas adjacent rotors are spinning in opposite directions i.e. rotors 1 and 2 are spinning clockwise while 3 and 4 spin counterclockwise.

*Figure 3  Qball X4 axes and sign conventions*
3.1 Actuator Dynamics

The thrust generated by each of the four brushless motors is modeled by the following first order system [26], [16]:

\[ F_i = K \frac{\omega}{s + \omega} u_i \]  \hspace{1cm} (20)

where \(u_i\) is the PWM input to the \(i\)th actuator, \(\omega\) is the actuator bandwidth and \(K\) is a positive gain.

Hereinafter we will use the state variable \(v\) to represent the actuator dynamics as follows [26], [16]:

\[ v = \frac{\omega}{s + \omega} u_i \]

such that:

\[ F_i = Kv \]  \hspace{1cm} (21)
3.2 Roll/Pitch Model

We assume that the rotations about the \( x \) and \( y \) axes are decoupled in order to allow us to model the roll and pitch models as shown in Figure 5. In this figure, we can see that two propellers contribute to the motion on each axis. The thrust generated from each motor can be calculated using equation (1) and using its corresponding PWM input. The motion about the center of gravity is due to the difference in thrust of the two rotors. The pitch and roll angles (\( \theta \) and \( \phi \), respectively) can be formulated using the following first order dynamics [26], [16]:

\[
\begin{align*}
J_{roll} \ddot{\phi} &= \Delta F_{3,4} L, \\
J_{pitch} \ddot{\theta} &= \Delta F_{1,2} L
\end{align*}
\]

where \( J_{roll} = J_{pitch} \) denote the rotational inertia of the quadcopter in the roll and pitch axes. They are equal since a quadrotor is generally symmetrical about each axis. \( L \) is the distance from the center of each axis to the center of the DC motors or the center of the propellers. \( \Delta F_{1,2} \) and \( \Delta F_{3,4} \) each represent the differences in thrust between the motor pairs (1,2) and (3,4), respectively, as described in Figure 4 model of the yaw axis with propeller direction shown [26], [16].

![Figure 5 Model of the roll/pitch axis](image)
By combining the actuator dynamics and the roll/pitch dynamical models we come up with the following state space descriptions [16]:

\[
\begin{bmatrix}
\hat{\theta} \\
\hat{\phi} \\
\hat{v}
\end{bmatrix} =
\begin{bmatrix}
0 & 1 & 0 \\
0 & 0 & \frac{KL}{J_y} \\
0 & 0 & -\omega
\end{bmatrix}
\begin{bmatrix}
\theta \\
\phi \\
v
\end{bmatrix} +
\begin{bmatrix}
0 \\
0 \\
\Delta u_{1,2}
\end{bmatrix}
\] (23)

\[
\begin{bmatrix}
\hat{\theta} \\
\hat{\phi} \\
\hat{v}
\end{bmatrix} =
\begin{bmatrix}
0 & 1 & 0 \\
0 & 0 & \frac{KL}{J_z} \\
0 & 0 & -\omega
\end{bmatrix}
\begin{bmatrix}
\theta \\
\phi \\
v
\end{bmatrix} +
\begin{bmatrix}
0 \\
0 \\
\Delta u_{3,4}
\end{bmatrix}
\] (24)

Where \( \Delta u_{i,j} = u_i - u_j \).

3.3 Height Model

The vertical motion (along the Z-axis) of the Qball X4 is affected by all of the 4 rotors. The dynamic model of the Qball X4 can be written as

\[
M \ddot{Z} = 4F \cos(\theta) \cos(\phi) - Mg
\] (25)

where \( F \) is the thrust from each rotor, \( M \) is the total mass of the Qball X, \( Z \) is the height, \( g \) is the acceleration due to gravity, and \( \theta \) and \( \phi \) are the pitch and roll angles, respectively [26], [16]. Here we can see that when \( \theta \) and/or \( \phi \) are nonzero, the overall thrust vector will not be perpendicular to the X-Y plane. If we combine equation (20) with (25) we get the following state space [26], [16]:

\[
\begin{bmatrix}
\dot{Z} \\
\dot{\omega}
\end{bmatrix} =
\begin{bmatrix}
0 & 1 & 0 \\
0 & 0 & \frac{4K \cos(\theta) \cos(\phi)}{M} \\
0 & 0 & -\omega
\end{bmatrix}
\begin{bmatrix}
Z \\
\dot{Z} \\
v
\end{bmatrix} +
\begin{bmatrix}
0 \\
0 \\
\Delta u_{z,6}
\end{bmatrix}
\] (26)
3.4 X-Y Position Model

Assuming the yaw angle does not change,

$$\psi = 0$$

The X-Y dynamical position model is given as:

$$M\ddot{X} = 4F \sin(\theta) \quad (27)$$

$$M\ddot{Y} = -4F \sin(\phi) \quad (28)$$

Then using (20),(27), and (28), we get the following state space [26], [16]:

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{4K \sin(\theta)}{M} \\ 0 & 0 & \frac{-4K \sin(\phi)}{M} \end{bmatrix} \begin{bmatrix} X \\ \dot{X} \\ \dot{Y} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} u_z \quad (29)$$

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{4K \sin(\theta)}{M} \\ 0 & 0 & \frac{-4K \sin(\phi)}{M} \end{bmatrix} \begin{bmatrix} X \\ \dot{X} \\ \dot{Y} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} u_z \quad (30)$$

3.5 Yaw Model

By assuming a linear relationship between the motor torque ($\tau_i$) and the PWM signal ($u_i$):

$$\tau_i = K_z u_i, \forall K_z \in \mathbb{R}^+, i = 1, ..., 4 \quad (31)$$

the motion about the z-axis is given by the difference in torques exerted by the clockwise and counter-clockwise pairs of rotors as shown in Figure 4 [26], [16].
The rotation about the yaw axis can be modeled by the following equation [26], [16]:

\[ J_{yaw} \ddot{\psi} = \Delta \tau \]  

(32)

where \( J_{yaw} \) is the rotational inertia about the z-axis, \( \psi \) is the yaw angle, and \( \Delta \tau \) is the resultant torque of the motors. Therefore by replacing with [torque equation] we get the resultant torque to be [26], [16]:

\[ \Delta \tau = K(u_i + u_2 - u_3 - u_4) = K_x \Delta u \]  

(33)

The state space of the yaw dynamical model is given as follows [26], [16]:

\[
\begin{bmatrix}
\dot{\psi} \\
\dot{\psi}
\end{bmatrix} =
\begin{bmatrix}
0 & 1 \\
0 & 0
\end{bmatrix}
\begin{bmatrix}
\psi \\
\dot{\psi}
\end{bmatrix}
+ \begin{bmatrix}
0 \\
\frac{K}{J_{yaw}}
\end{bmatrix} \Delta u
\]  

(34)

3.6 Qball X4 Linear Model, Command inputs, and Parameters.

In order to linearize the model, we simply need to assume the roll/pitch angles are small:

\[ \because \sin(\theta) \approx \theta \quad , \quad \cos(\theta) \approx 1 \quad \text{where } |\theta| \leq 0.3 \text{ rads} \]

Therefore our linear state space models for (26), (29), and (30) will be as follows [26], [16]:

**Height**:

\[
\begin{bmatrix}
\dot{Z} \\
\dot{\nu}
\end{bmatrix} =
\begin{bmatrix}
0 & 1 & 0 \\
0 & 0 & \frac{4K}{M} \\
0 & 0 & -\omega
\end{bmatrix}
\begin{bmatrix}
Z \\
\dot{\nu}
\end{bmatrix}
+ \begin{bmatrix}
0 \\
0 \\
0
\end{bmatrix} u_z
+ \begin{bmatrix}
0 \\
0 \\
-g
\end{bmatrix}
\]  

(35)

**X position**:

\[
\begin{bmatrix}
\dot{X} \\
\dot{\nu}
\end{bmatrix} =
\begin{bmatrix}
0 & 1 & 0 \\
0 & 0 & \frac{4K}{M} \phi \\
0 & 0 & -\omega
\end{bmatrix}
\begin{bmatrix}
X \\
\dot{\nu}
\end{bmatrix}
+ \begin{bmatrix}
0 \\
0 \\
0
\end{bmatrix} u_z
\]  

(36)

**Y position**:

\[
\begin{bmatrix}
\dot{Y} \\
\dot{\nu}
\end{bmatrix} =
\begin{bmatrix}
0 & 1 & 0 \\
0 & 0 & -\frac{4K}{M} \phi \\
0 & 0 & -\omega
\end{bmatrix}
\begin{bmatrix}
Y \\
\dot{\nu}
\end{bmatrix}
+ \begin{bmatrix}
0 \\
0 \\
0
\end{bmatrix} u_z
\]  

(37)
Note that the effective PWM inputs to the motors are as follows [16]:

\[
\begin{aligned}
    u_1 &= u_x + \Delta u_\theta + \Delta u_\phi \\
    u_2 &= u_x - \Delta u_\theta + \Delta u_\phi \\
    u_3 &= u_x + \Delta u_\theta - \Delta u_\phi \\
    u_4 &= u_x - \Delta u_\theta - \Delta u_\phi
\end{aligned}
\]

Table 3 collects all the physical parameters of the Qball X4 [26].

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K$</td>
<td>120 N</td>
</tr>
<tr>
<td>$\omega$</td>
<td>15 rad/sec</td>
</tr>
<tr>
<td>$J_x$</td>
<td>0.03 Kgm$^2$</td>
</tr>
<tr>
<td>$J_y$</td>
<td>0.03 Kgm$^2$</td>
</tr>
<tr>
<td>$J_z$</td>
<td>0.04 Kgm$^2$</td>
</tr>
<tr>
<td>$M$</td>
<td>1.53 Kg</td>
</tr>
<tr>
<td>$K_z$</td>
<td>4 Nm</td>
</tr>
<tr>
<td>$L$</td>
<td>0.2 m</td>
</tr>
</tbody>
</table>

Table 3  Table of Physical Parameters of the Qball X4
3.7 Quadcopter System Identification

In order to design the ERG it is required to get a full system state space model. Since the system has been pre-compensated, there are already PID controllers present that achieve desired performance in the linear region. These controllers have filters that complicate their dynamics and therefore it was elected to go through a system identification process in order to estimate the complete closed-loop dynamics of the system. Quanser provided a simulation model of the Qball X4 that was used to do the data gathering of the system. For this thesis, we will discuss how the X position closed loop dynamics were designed since the dynamics of the X and Y models differ only in sign and therefore the ERG found using the X position model can be easily adapted to the Y position model.

Before the system identification process can begin, it was necessary to remove all saturation blocks from the Qball X4 Simulink model so that we can get the natural response of the system from any reference input. Next, a unit step input was given as a reference to the system and the system’s pitch and x-position responses were recorded to the workspace (as can be seen in Figure 6).

![Figure 6 Step Response of Qball X4 simulation Without Saturation](image)

Using this data, we used the System Identification Toolbox in Matlab to obtain possible State Space Models. After several state space orders were tested the state space with 14 states seemed to be the
closest match to the Qball X4 simulation test data (see Appendix A). The state space was then reduced to 5th order using balanced truncations to the following:

\[
A = \begin{bmatrix}
-0.0381 & -5.3253 & -0.2092 & 0.1320 & -0.0272 \\
5.4360 & -2.1104 & -1.8431 & 6.5320 & 0.4601 \\
-0.2104 & 2.5755 & -1.7747 & -7.1344 & -0.3545 \\
0.6791 & -6.7197 & 14.3102 & -18.2676 & -5.5197 \\
-0.1061 & 0.8152 & -1.3923 & 6.5062 & -2.4648 \\
\end{bmatrix}, \quad B = \begin{bmatrix}
0.4184 \\
-2.9862 \\
1.4543 \\
-3.8615 \\
0.5875 \\
\end{bmatrix}
\]

\[
C = \begin{bmatrix}
0.3885 & 0.8105 & 1.3896 & 0.0486 & 0.3645 \\
-0.1553 & -2.8741 & -0.4290 & 3.8611 & 0.4608 \\
\end{bmatrix}, \quad D = \begin{bmatrix}
0.0051 \\
0.0387 \\
\end{bmatrix}
\]

to be used in a state space of the form:

\[
\dot{x} = Ax + Bu \\
y = Cx + Du
\]

Where the \( u \) is the reference command and \( y = [r_{out} \quad \theta_{out}]^T \)

Since our ERG needs to use the states of system in order to perform its control action, and since we cannot directly access the states of the system, we built a standard Luenberger observer to estimate the states of the system with

\[
L = \begin{bmatrix}
46.6330 & -7.6179 \\
-12.9714 & -2.7132 \\
6.1817 & 7.5563 \\
-30.8132 & 1.9955 \\
51.9487 & 4.8553 \\
\end{bmatrix}
\]

and

\[
(A - LC) = \begin{bmatrix}
10.0541 & 0.6048 & 15.0182 & 17.6388 & 6.4381 \\
-1.4384 & 19.2832 & -7.1231 & -36.6111 & -6.0898 \\
12.9603 & 23.9896 & 57.9848 & -24.4739 & 4.7911 \\
\end{bmatrix}
\]

It is also necessary to formalize our constraints on the system. For this implementation, we focused on keeping the pitch and roll angles small since the Qball X4 has a Watchdog system that
will turn off the Qball if either angle exceeds 40°. The saturation point in the Qball for the pitch command was 0.3 rads so we will use this as the constraint. We formalize this for pitch as follows:

\[ |\theta| \leq \theta_{\text{max}} = 0.3 \]

\[ |C_{\theta}x_{\theta} + D_{\theta}u| \leq \theta_{\text{max}} \]

Where \( C_{\theta} \) and \( D_{\theta} \) are the second rows of the \( C \) and \( D \) matrices, respectively, in the state description. Therefore, we have an upper and lower bound for the pitch:

\[-C_{\theta}x_{\theta} - D_{\theta}u + \theta_{\text{max}} \geq 0\]

\[C_{\theta}x_{\theta} + D_{\theta}u + \theta_{\text{max}} \geq 0\]

It is now possible to implement the ERG described in Chapter 2.
Chapter 4

Results

The ERG was implemented following the plan of Figure 2 using the formulas in Table 2 and this was inserted into the simulation environment. Simulations were run for both a unit step and a step of height 5 on three different simulations. The simulations were of the system with saturations, the system with all saturations removed, and the unsaturated system with the ERG. The results are given below. For completeness sake, the results of the simulation using the ERG with the internal states and the 14 state model were also included. This also serves the purpose to show the effects of model reduction and estimating the state using the observer.

Figure 7 Unit Step in X Simulation with Saturation
Figure 8 Unit Step in X Simulation without Saturation
The results of the unit step input in the $x$ direction are given in Figure 7, Figure 8, Figure 9, and Figure 10 for the different systems. Figure 7 is of a unit step input to the Qball X4 Simulation with the saturations in place and as we can see this system settles in about 3.5 seconds, has no overshoot, and has a max pitch response of about 0.1 rads. Figure 8 on the other hand, is the result of removing all the saturation points from the Qball X4 simulation and as we can see it settles in about 2.5 seconds, has no overshoot, and its max pitch response is about 1.2 rads. Recall that the Watchdog in the physical system shuts the system down when either pitch or roll exceed about 0.7 rads so in real life this system would have shut down. Figure 9 shows the simulation results of a step input into the system augmented with the ERG. There is an overshoot of about 20%, the settling time is about 3 seconds, and the max pitch response is about 0.3 rads, which is within the constraints. The ERG augmented system is not as smooth as the saturated system but its settling time is about 15% faster while remaining within the constraints. Figure 10 shows us the result of implementing the ERG with the 14 state model determined in Section 3.7. From here we can see that the system is much more conservative than the others as it takes about 12 seconds to settle but it never violates the constraints as its max angle is about 0.19 rads. This shows us that once we start estimating states and reducing the model, the results we get turn out to be slightly different than expected.
Next, we try a step input with a final value of 5.

Figure 11 shows the simulation of the system with the saturations in place and the settling time is about 11.5 seconds, there is no overshoot, and the max pitch response was about 0.3 rads, due to the saturation blocks. Figure 12 is the result of the simulation with the saturation blocks removed and we can see that the system did not settle at the reference set point. In fact, in the simulation the system went completely unstable and crashed since the max pitch response was about 6.5 rads.

Figure 14 is the result of the simulation of the system augmented with the ERG and we can see that the settling time was about 7 seconds, the overshoot was about 20%, and the max pitch response...
was almost 1.5 rads. This violates the constraints and would have shut down the Qball X4 in real life. This is however, a great improvement over the system without the saturation blocks as it improved the pitch response by a factor of about 5. Figure 13 demonstrates the response of the system when the states are taken directly from the system without the use of an observer. The settling time is about 16 second, there is no overshoot, and the max pitch response is about 0.3 rads which does not violate the constraints. From this simulation, you can see the conservative nature of the system as there are various ‘knees’ in the x response and g. These are where the ERG ‘tells’ the system to slow down or else it will violate its constraints.

For completeness sake, we extended the ERG to act on the Y position closed loop system to see if it acts the same way as the X position closed loop system. For this simulation, the same state space that was used for the X position closed loop system was used for the Y position closed loop system. The rationale behind this is that the X and Y position closed loop models should be identical as the system is symmetrical. The results follow for a unit step.

![Figure 15 Unit step in Y Simulation with Saturation](image1)

![Figure 16 Unit step in Y Simulation without Saturation](image2)
Figure 17 Unit Step in Y Simulation without Saturation with ERG

Figure 15 shows us the result of the simulation with the original saturations still in place. We can see it is actually the same as Figure 7 except that the roll response is the negative of the pitch response. This is the same as Figure 8 and Figure 16. This shows us that the two models are similar. Now if we look at Figure 17 we can see that the settling time is about 4 seconds, the overshoot is about 50%. The max roll response is about 0.3 rads that is within the constraints. Therefore, if we compare this to Figure 9 we can see that the overshoot is bigger and the settling time is a bit greater but the response is similar because the two systems have very similar dynamics.
Chapter 5

Conclusion and Future Work

In this thesis an Explicit Reference Governor was studied, implemented, and tested in the Qball X4 quadcopter. The main objective of the Explicit Reference governor was to attenuate any reference signals into a pre-compensated linear system subject to constraints such that the constraints were not violated. The ERG improved the performance of the system when compared to the system without any saturation blocks, but was unable to handle larger reference commands without violating constraints. Any reference commands up to one unit in the x-direction were handled just fine but any higher and it would violate the constraints. When it performed as desired, the system with the ERG was able to settle about 15% faster than the saturated system. It also improved on the performance of the unsaturated system by a factor of 4. Due to having to estimate the states of the system using the observer, the theory of the ERG gets distorted as we are estimating the states of a five state system that is the product of a 14 state system reduction. This can be seen in the simulations that did not use the observer. They were slower but they showed the true action of the ERG if the states are accurately estimated.

5.1 Future Work

For future work, the ERG will need to be improved in order to better handle constraints and to reduce the likelihood of constraint violation. After this has been done, the ERG should be implemented on the actual Qball X4. A more accurate system identification will possibly improve performance of the ERG. ERG should be extended to the height system.
Appendix A

Different System Identification with different states.

11 State Estimate:

Figure 18: x response of 11 state estimate

Figure 19: pitch response of 11 state estimate

12 State Estimate:

Figure 20: x response of 12 state estimate

Figure 21: pitch response of 12 state estimate
14 State Estimate:

Figure 22: $x$ response of 14 state estimate

Figure 23: Pitch response of 14 state estimate

15 State Estimate

Figure 24: $x$ response of 15 state estimate

Figure 25: Pitch response of 15 state estimate
Bibliography


command governors for control of turbocharged gasoline engines based on linear models,”


