11-1-2013

Dexterous Hexrotor UAV Platform

Guangying Jiang
University of Denver

Dexterous Hexrotor UAV Platform

A Thesis

Presented to

the Faculty of the Daniel Felix Ritchie School of Engineering and Computer Science

University of Denver

In Partial Fulfillment

Of the Requirements for the Degree of

Master of Science

By

Guangying Jiang

November 2013

Advisor: Prof. Richard Voyles

Author: Guangying Jiang
Title: Dexterous Hexrotor UAV Platform
Advisor: Richard Voyles
Degree Date: November 2013

Abstract

Mobile manipulation is a hot area of study in robotics as it unites the two classes of robots: locomotors and manipulators. An emerging niche in the field of mobile manipulation is aerial mobile manipulation. Although there has been a fair amount of study of free-flying satellites with graspers, the more recent trend has been to outfit UAVs with graspers to assist various manipulation tasks. While this recent work has yielded impressive results, it is hampered by a lack of appropriate testbeds for aerial mobile manipulation, similar to the state of ground-based mobile manipulation a decade ago. Typical helicopters or quadrotors cannot instantaneously resist or apply an arbitrary force in the plane perpendicular to the rotor axis, which makes them inadequate for complex mobile manipulation tasks. Based on the concept of force closure (a term from the dexterous manipulation community), this thesis introduces the new type of dexterous, 6-DoF UAV which provides the unique capability of being able to resist any applied wrench, or generalized force-torque. In this thesis, we describe the importance of force closure for mobile manipulation, explain why it is lacking in current UAV platforms, and describe how our hexrotor provides this important capability as well as exhibiting holonomic behavior.

Acknowledgements

First of all, I would like to express my appreciation to my adviser Professor Richard Voyles for his invaluable guidance in my two years of graduate studies and research. I also appreciate the support of Professor and Professor for serving as the committee members in my final oral examination.

Secondly, I thank all my lab mates a lot, especially, Sam Povilus, Yanzhe Cui, and Justin Huff. Sam did the initial design of Dexterous Hexrotor. Yanzhe worked with me on the project at the beginning. Justin helped me fabricated some parts in the machine shop.

Thirdly, I thank my school, University of Denver, for the great education.

Last but not least, I would like to give my special thanks to my family, their encouragement drives me until the end of my study. Also, thanks so much to my friends, they make me not lonely here. I enjoy all the happy time with them.

Support for my work was generously provided by the National Science Foundation through grants CNS-1138XXX and the NSF Safety, Security and Rescue Research Center.

Table of Contents

List of Figures

List of Tables

Chapter One: Introduction

A multicopter is a UAV (Unmanned Aerial Vehicle) that is lifted and manipulated by a number of rotors. Rotors of contemporary multicopters are mostly installed on the frame such that all their thrust vectors are parallel and vertical as in Fig. 1.1. Control of vehicle motion is achieved by alternating the pitch or roll rotation rates.

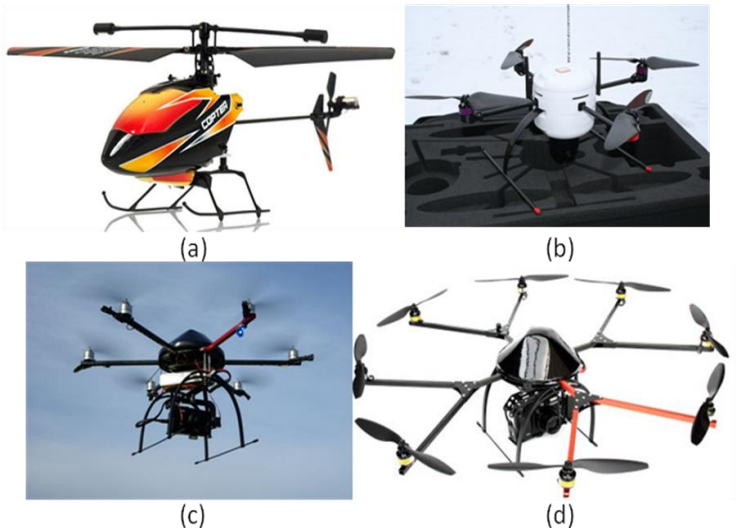


Fig.1.1 Current UAV platforms (a) Helicopter (b) Linkquad Quadrotor (c) Holger Buss' MK Hexrotor (d) Octocopter.

Numerous multicopter systems have been developed as custom research platforms, custom teaching platforms, toys, and commercially available systems (e.g. [1, 2, 3]). There even exist a number of open source projects [4, 5]. Most of these multicopters are like the ones used in [6] or [7], having a number of fixed pitch propellers with parallel and vertical thrust vectors.

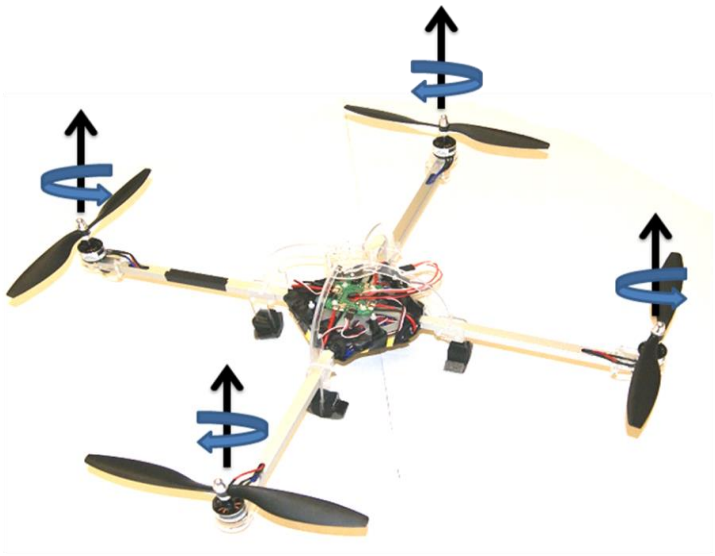


Fig.1.2 Quadrotor illustrated with thrust vectors.

Depending on how these N thrust actuators are physically arranged, various combinations of thrust magnitudes result in net force/torque vectors that span a subset of the 6-D Cartesian space of generalized forces. Coriolis forces induced by the spinning rotors can even be used to augment the net force vector on the multirotor body. Mathematically, a matrix can be constructed that provides a mapping from N-D actuator space to 6-D Cartesian force space, but this matrix can have rank no greater than four when the thrusters are all parallel. In fact, the standard quadrotor configuration results in rank of exactly four as the four thrusters provide linear force along Z axis, and torques around X and Y, while Coriolis effects provide torque around the Z axis. (This is what makes tri-rotors infeasible and conventional single-rotor helicopters also have four actuators: main rotor, tail rotor and two actuators on the swash plate.) Instantaneous exertion of linear forces along X and Y are impossible with these configurations, they can

only lift, pitch, roll and yaw. They cannot move laterally without first rolling or pitching an angle so these systems are nonholonomic.

The multicopter system is dynamically unstable but allows for high maneuverability [8]. Some attempts have been made to change this system like quadrotors, to increase controllability [9], stability [10], or maneuverability, but they still only have four motors. This instability lends itself to surveillance where quick movement is key and inability to maintain a stable pose is not an issue, but not much else can be done with these platforms.

Several hexrotor platforms have been developed including a miniature version from Airbotix and robust six- and eight-rotor versions from Aerobot [4] and Draganfly [1]. Recently, the U.S. Air Force announced a solicitation for hexrotor platforms for military procurement. All these known hexrotor configurations employ parallel thrust vectors (identical to the quadrotors) which result in rank-4 actuator-to-Cartesian mappings incapable of xerting arbitrary forces in 6-D space. These six-rotor helicopters have been created simply for the increased lift of six rotors, redundancy, and smoother operation.

1.1 Motivation

Manipulation is making a come-back in robotics. Through most of the 90's and early 2000's, mobile robots dominated research and application, such as the DARPA grand challenge and autonomous helicopters. Now in kind of mid 2000's and later up to now, there has been research in manipulation again. Manipulation is about agility, dexterity, immobilizing a part when grasping.

In harkens back in manufacturing, there is an emerging field of mobile manipulation, combining the growth of mobile robots with robots we can manipulate. The field of

mobile manipulation [17, 18, 19] combines two broad classes of robots, locomotors and manipulators, yet the potential impact is greater than the sum of the parts. Aerial mobile manipulation is an emerging field within mobile manipulation for which the locomotor is a UAV (unmanned aerial vehicle) [21, 34, 36, 37]. The popular quadrotor has become the main UAV of choice in robotics research, due to its ease of control and low cost. The added mobility and access that quadrotors provide brings a new dimension to the study of mobile manipulation and new challenges, as well.

One of the greatest challenges that UAVs, in general, introduce, and quadrotors in particular, is the inability to instantaneously exert forces in the plane. Quadrotors are non-holonomic; in order for them to move forward or sideways, they first have to pitch the entire body of the quadrotor to direct the thrust vector in the desired direction. What this means for aerial mobile manipulation is that the quadrotor cannot resist an arbitrary generalized force/torque instantaneously. In the parlance of the dexterous manipulation community, it lacks "force closure". In fact, in one of the first attempts to use a UAV to interact physically with its environment, Albers et al had to add an auxiliary actuator to maintain contact so Newton' Third Law of equal and opposite reaction would not immediately push the UAV away [38].

1.2 Related Works

1.2.1 Force Closure

Force closure and form closure [22] are concepts from dextrous manipulation that long pre-date the field of robotics. Reuleaux [23] and his contemporaries analyzed mobility under constraint in the late 1800's. Since those early days, force closure and form closure

have received significant attention, yet definitions have not always been consistent within the robotics literature. Historically, force closure has been the more mature research area with a well defined theory and set of definitions revolving around screw theory. Form closure, on the other hand, historically, has been more imprecise. Rimon and Burdick published a seminal work in robotics [24] that rigorously defined first and second order force and form closure and showed their equivalence. We base our discussion on this analysis and use their definition of the term “force closure” in this paper to reinforce the relevance to arbitrary wrenches.

Force closure, as defined by Rimon and Burdick, is the ability of a mechanism to directly resist any arbitrary wrench (combination of force and torque) applied to it. A "force closure grasp" in the dexterous grasping literature [25, 26, 27], is a grasp of an object that can resist an arbitrary wrench applied to the object. This class of grasps is important to the dexterous manipulation community but is often ignored by the mobile manipulation community because of the large mass of the mobile base and other issues have greater priority.

1.2.2 Aerial Mobile Manipulation

Aerial mobile manipulation is a newly emerging field even though it has existed for decades. Fig. 1.3 shows a ski resort being assembled in Canada. This is actually a ski resort being assembled in Canada. There are a group of workers doing the adjustment as this big helicopter doing the heavy lifting.

Fig. 1.3 A ski resort being assembled in Canada (Reprinted with permission from Judy Shellabarger Gosnell)

Bring this apart, the helicopter can't actually do the assembly. Due to Newton's third law of equal and opposite reaction, as soon as the manipulated part comes into contact with the environment, it would balance away. Because the helicopter cannot exert forces instantaneously in the plane. It can only exert forces up and down. It can do some pitch and yaw, but it can't exert forces instantaneously in the plane.

Fig. 1.4 Aerial Mobile Manipulation: (a) Yale Aerial Manipulator capturing a block in hover (Three helicopters transporting a load (Reprinted with permission from Paul E.I. Pounds and Ivan Maza)

Through most of the 90's and early 2000's, mobile robots dominated research and application. And one of those things was quadrotors, helicopters with four blades. So people have started to apply helicopters and quadrotors to mobile manipulation.

Arron Dollar has studied unstable dynamics of the vehicle and coupled object-aircraft motion while grasping objects during flight [34]. They also demonstrate grasping and retrieval of variety of objects while hovering, using Yale Aerial Manipulator. Paul Oh equipped the quadrotor with a gripper and studied about contact forces and external disturbances acted on the gripper and the entire manipulation system [37]. Anibal Ollero [39] and the GRASP lab at the University of Pennsylvania [21] both have worked on using multiple collaborative UAVs in order to perform transportation tasks. They did research on the interactions between UAVs, physical couplings in the joint transportation of the payload and stabilizing the payload along three-dimensional trajectories.

The helicopter or quadrotor approach is limited though, only the top of objects can be grasped by the under-hung grasper and oddly balanced objects must be lifted by multiple UAVs. The UAV teams rigidly clamp to the manipulated object but, due to their design, they can only apply limited forces and torques. These piloted helicopters and quadrotors manipulate objects by hanging them from a line and/or employ fewer than six vehicles, severely limiting the range of wrenches that can be applied to the object. In generalized force space, they are effectively degenerate.

The floating nature of aerial platforms and the highly compliant nature of their positioning bring the issue of force closure to the fore. With fixed-base manipulators, determining the degree of force closure of a manipulation system is simplified to determining the degree of force closure of the gripper or end effector. In mobile

manipulation, the manipulator base is not fixed to the ground so determining the set of wrenches that can be resisted is not strictly limited to the capabilities of the end effector. But due to the large differences in mass of the mobile base and end effector, it is generally safe to assume the degree of force closure is limited by the end effector and not by the ability of the mobile base to remain motionless. Therefore, for aerial mobile manipulation, the concept of force closure of the entire manipulation system needs to be considered.

Conventional aerial platforms are not able to resist an arbitrary wrench so an end effector carried by such a vehicle will not be able to exhibit force closure. Force closure for arbitrary rigid objects in 3-D space requires six controllable degrees of freedom in force-torque space to truly accomplish. Current quad rotors lack both the number of degrees of freedom but also independence of the degrees of freedom due to the fact that the force vectors are all parallel.

Furthermore, it is not sufficient to simply attach a 6-DoF manipulator to the bottom of a quadrotor or other degenerate aerial platform, as this does not guarantee force closure. While a 6-DoF manipulator can exert arbitrary wrenches when grounded, if the base upon which it is mounted cannot resist an arbitrary wrench, the combination remains degenerate.

Finally with the multi-quadrotor approach force closure has not been obtained, lateral forces must be compensated for by torques, not direct opposition. We believe that a single UAV can have more manipulation abilities, chiefly force closure, all forces put on the grasped object can be opposed directly. Also a single UAV has numerous advantages

over a system that requires multiple UAV's including simplicity, communications, cost and available poses.

1.2.3 Nonparallel Multicopter

Besides force closure, an interesting side effect of a UAV with control over six degrees of freedom is the ability to accomplish 3D translational flight (hover at any orientations and translate in any directions). It would be much easier for a 6 degrees of freedom UAV to achieve unusual body attitudes as Mellinger et al did [31]. They developed elaborate dynamic control methodologies to achieve unusual body attitudes during aggressive flight maneuvers to allow their quadrotor UAVs to pass through narrow windows and other hazards. Using high precision external sensing of the pose of the vehicle and an accurate dynamic model, their quadrotor is capable of passing through the diagonal of a rectangular orifice, presumably to enter a damaged building through a narrow window in response to a disaster. The window presents an orifice of which the horizontal width is not sufficient to allow passage of the UAV, but the diagonal distance is. Since a typical quadrotor cannot hover at an arbitrary orientation (the result of a mapping from actuator space to Cartesian generalized force space with rank less than six, indicative of incomplete force closure), an aggressive dynamic maneuver is required to achieve entry.

Therefore, as a way to gain full controllability over the 6-Dof robot pose and the ability of tracking an arbitrary trajectory over time (e.g., hover on one spot with an angled pose), some UAV platforms with tilting propellers have been designed. M. Ryll proposed a quadrotor design with tilting propellers [32]. In this work, to solve the problem of limited mobility of standard quadrotor, rather than four fixed rotors, four variable-pitch rotors are

used to provide an additional set of control inputs. Because of standard quadrotor's inherent under actuation for 6-DoF parameterizing the quadrotor position/orientation in space, Ryll claims that they can gain full controllability over the quadrotor position/orientation by means of these four additional actuated degrees of freedom.

Developed about the same time as ours, a similar non-planar hexrotor design was introduced by D. Langkamp [33]. Using six fixed-pitch/variable-speed or variable-pitch/fix-speed rotors, the "Tumbleweed" is designed to achieve full flight envelope (their saying of 3D translational flight).

A prototype is proposed with fixed-pitch/variable-speed rotors and all rotors are pitched at 45°. But as they claimed, it cannot achieve full flight envelope because of limited forces it can generate in frame plane. So they abandoned this design and shifted to a variable-pitch/fix-speed rotors design. And by the use of high thrust/weight electric variable pitch units and a low airframe mass fraction, the Tumbleweed can achieve full flight envelop if the robot can be lifted by only one pair of rotors.

In these two works, the actuation concept of tilting propellers during flight actually makes it possible to access all 6 degrees of freedom of the robot. But for aerial manipulation tasks, full flight envelop is not important. What is important in resisting any arbitrary wrench is fast response of exerting forces. Tilting propellers during flight using servos may not be fast enough to act on outside wrench.

1.3 Contribution

We propose a new approach in aerial vehicles as the Dexterous Hexrotor UAV that can instantaneously resist arbitrary forces -- in other words, it provides force closure. To

perform precise and effective mobile manipulation, this is a property that any locomotor must have, be it ground-based, water-based, or air-based. To achieve this, the thrusters of our hexrotor are canted so that the combined thrust vectors span the space of Cartesian forces and torques. This adds little cost or complexity to a conventional hexrotor.

With decomposed forces and torques, we derived a thrust mapping mapping from actuator space to force/torque space for the Dexterous Hexrotor.

Since each rotor is rotated a cant angle around its radius, we also developed a metric for the optimization of Dexterous UAV performance for manipulation-based tasks based on the condition number of the conversion matrix.

With an attitude controller created to stabilize the Dexterous Hexrotor for human controlled flight and a control system based on the conversion matrix, a flight-capable prototype with translational force control has been built and tested.

1.4 Thesis Organization

Chapter 2 is the heart of this work and describes the concepts of the Dexterous Hexrotor, the structure design, force/torque decomposition, and cant angle selection.

Chapter 3 describes the control system and the attitude controller we built for stable flight.

Chapter 4 gives some technical details of the Dexterous Hexrotor prototype.

Chapter 5 presents the experiment results from the platform, while Chapter 6 concludes and mentions future directions that could be explored.

Chapter Two: Dexterous Hexrotor Theory

The basic idea behind hexrotor is we are going to provide 6 actuators. So the input is 6 degrees of freedom. If we want to control 6 degrees of freedom, we know we have to have a minimum of 6 actuators. Or else our matrix is going to deficient. If we have 4 actuators as a quadrotor, we cannot possibly have independent control over 6 degrees of freedom.

There are many hexrotors exist in the commercial world today. They all have six parallel thrust propellers spaced evenly around the circumference of a circle. All its thrusters are vertical. Therefore, they still result in rank deficient matrices, in other words, there are no components from parallel thrusts that act in the plane perpendicular to the rotor axis. In this kind of configuration, these hexrotors work like quadrotors.

Fig. 2.1 Typical Hexrotor

Because all these thrusters are parallel and vertical, they can only provide linear force along Z axis, and torques around X, Y axes. Torque around Z is achieved indirectly

through Coriolis forces resulting from differential angular velocities of the counter-rotating propellers.

If we spin all the rotors at a particular same speed, the robot hovers. If we vary speed of these rotors, the robot can roll, pitch and yaw.

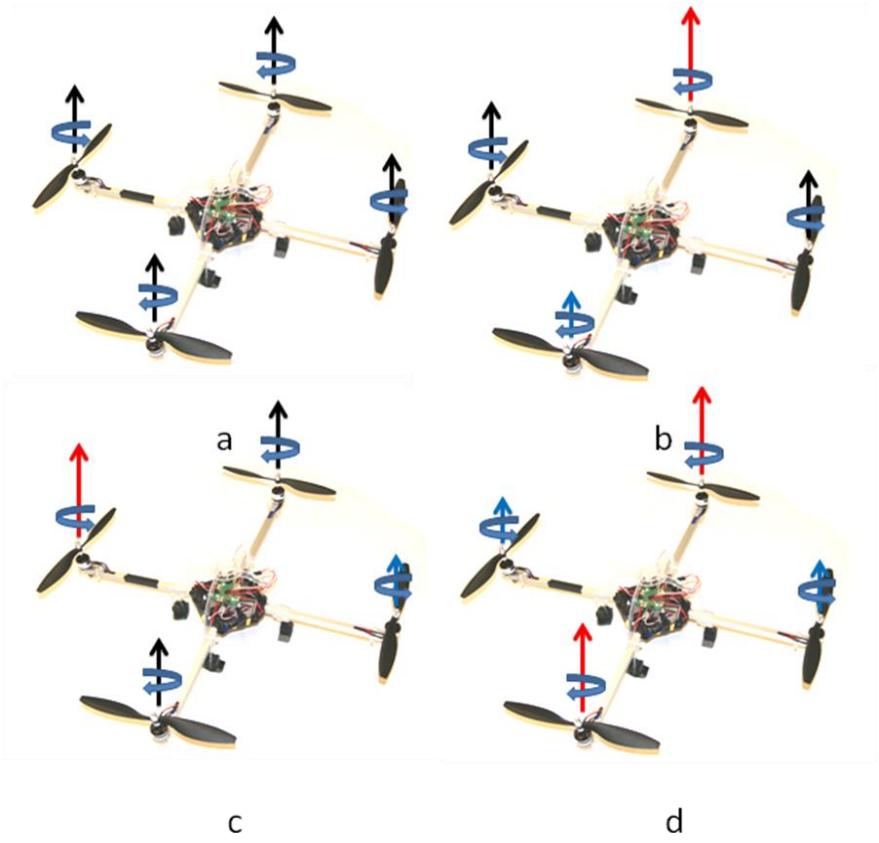


Fig. 2.2 How quadrotor moves (a) Hover, (b) Roll, (c) Pitch, (d) Yaw.

This configuration only has controllability over 4 degrees of freedom parameterizing the robot position and orientation in space. But it still can move arbitrarily in 6 degrees of freedom space, because translational acceleration in the plane perpendicular to the rotor axis can be achieved by maintaining a non-zero pitch or roll angle. So with these four degrees of freedom, quadrotors can do hover, roll, pitch, and yaw. If the robot wants to

move in the plane perpendicular to the rotor axis, it has to maintain a non-zero pitch or roll angle to get the translational acceleration.

To achieve 6 degrees of freedom, we first provide 6 actuators as a typical hexrotor design. Then we just make these parallel thrusters nonparallel to explore full entirely 6D space of forces and torques.

Fig. 2.3 The Dexterous Hexrotor, note the rotors are nonparallel.

As Fig. 2.3 shows, the Dexterous Hexrotor has 6 independently controlled rotors arranged in pairs on 3 planes. Each rotor is rotated a cant angle around its radius. Therefore in-plane components result while still maintaining a symmetric basis of vectors, and forces and torques around each axis can be produced.

Since the Dexterous Hexrotor can span the force-torque space. Therefore, by detaching and combining forces and torques produced by each rotor, forces and torques acting on the UAV around each axis can be accomplished and controllability over full 6 degrees of freedom can be achieved. By varying the speed and choosing the direction of the rotation

of the rotors, the Dexterous Hexrotor can get not only forces along Z axis, torques around X, Y, Z axes, but also forces along X, Y axes.

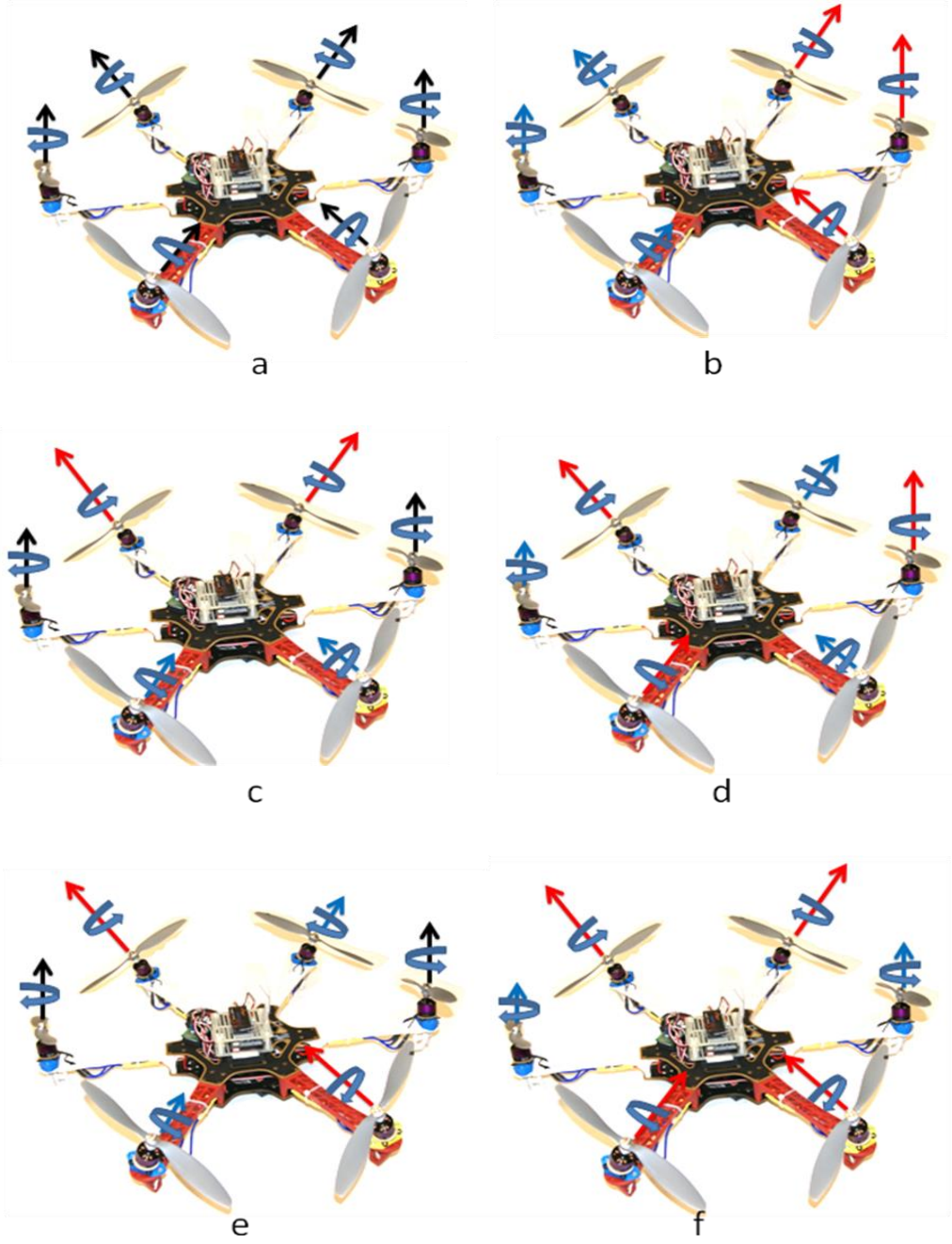


Fig. 2.4 How the Dexterous Hexrotor moves: (a) Hover, (b) Roll, (c) Pitch, (d) Yaw, (e) Translational acceleration along X axis (f) Translational acceleration along Y axis

With the force along Z axis and torques around X, Y, Z axes, the Dexterous Hexrotor can hover, roll, pitch and yaw just like typical quadrotors do. But instead of pitching or rolling an angle like typical quadrotors, the Dexterous Hexrotor can get translational acceleration acceleration by simply varying speed of these rotors. It can truly control its 6 degrees of freedom mobility.

2.1 Dexterous Hexrotor Structure

There are many ways to use six motors to create six independent degrees of freedom, but to create a design that was easy to fabricate, a disk design was used as seen in Fig.2.5. To get linear independence for all six degrees of freedom, the thrust vectors could not be on the same axis, nor pointed along one axis. This fact leads us to arrange these six rotors in pairs of three planes and rotate each thruster a cant angle φ around its radius as seen in Fig.2.6. We lay these rotors out along the edge of the disk canted tangentially to the edge of the disk alternating clockwise, counterclockwise, clockwise and so on as seen in Fig. 2.7.

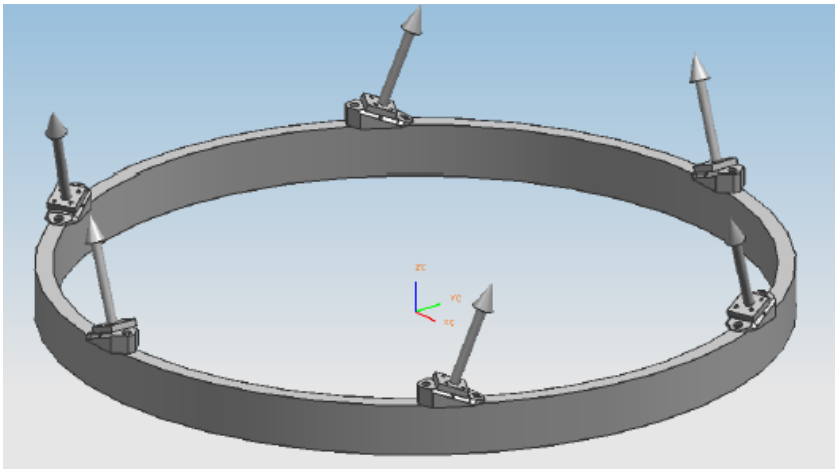


Fig. 2.5 A CAD model showing Dexterous Hexrotor thrust vectors in place of the motors.

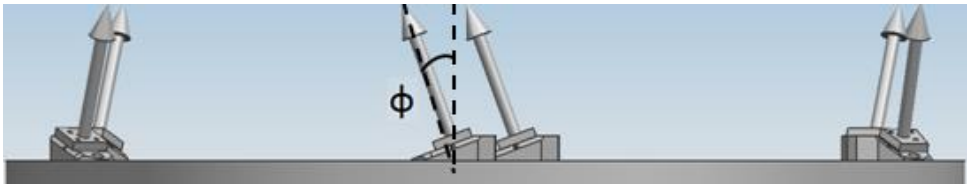


Fig. 2.6 A CAD model of the Dexterous Hexrotor thrust vector giving a clear picture of cant angle φ.

Position of six motors and their rotation are defined in Fig.2.7. X configuration of multicopter configuration is chosen, so X axis aligns with Motor3 and Motor6.

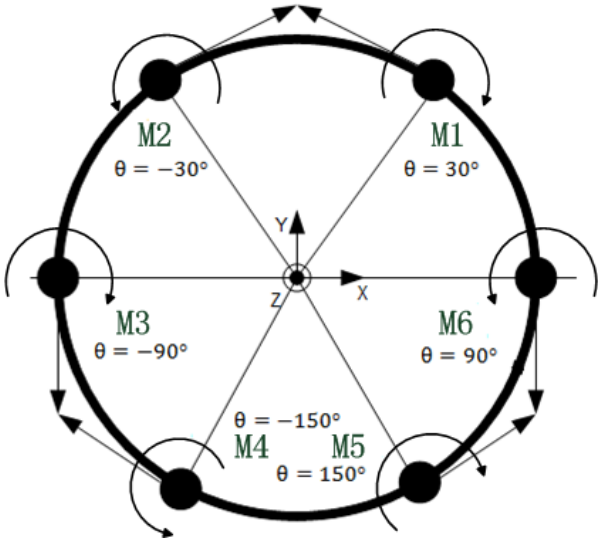


Fig. 2.7 Motor definition of the Dexterous Hexrotor, where n is the motor number, θ represents angular displacements of the motors.

To keep the direction of each rotor's torque same with its force's in-plane components, Motor1, Motor3, Motor5 are rotating clockwise, Motor2, Motor4, Motor6 are rotating counter-clockwise. And to minimize the effects of the torques from the motors themselves, pusher and puller propellers are used, Motor1, Motor3, Motor5 use one type while Motor2, Motor4, Motor6 use the other.

2.2 Force Decomposition

Each rotor produces a force and a torque. The force is a thrust point out along the rotor axis. The torque is generated by the drag of the propellers and acts on the body of the robot. The direction of the torque will be in the opposite direction of the motion of the propeller. This torque serves as yawing torque in typical quadrotors. But for **the** Dexterous Hexrotor, it will contribute to torque around all three axes.

The lift and drag produced by the propellers is proportional to the square of angular velocity. And we assume the square of angular velocity is proportional to the pulse width modulation command sent to the motors. Therefore, force and torque produced by each motor can be expressed as equation (2.1).

$$\begin{aligned} F_{motor} &= K_1 * PWM_{motor} \\ \tau_{motor} &= K_2 * PWM_{motor} \end{aligned} \qquad \text{(Eqn. 2.1)}$$

where F_{motor} and τ_{motor} are force and torque produced by the motors. K1 and K2 are motor-dependent parameters and can be determined experimentally. PWM_{motor} is the pulse width modulation command sent to the motor.

To compute the net force/torque acting on the UAV from all thrusters, we first decompose each motor's thrust and torque into X, Y, and Z components on the body frame. The components of Cartesian generalized forces from Motor1 can be put into a matrix as in equations (2.2) and (2.3).

$$\begin{bmatrix} F_{1fx} \\ F_{1fy} \\ F_{1fz} \\ F_{1\tau x} \\ F_{1\tau y} \\ F_{1\tau z} \end{bmatrix} = \begin{bmatrix} -K_1 * PWM_1 * cos(\theta_1) * sin(\phi) \\ K_1 * PWM_1 * sin(\theta_1) * sin(\phi) \\ K_1 * PWM_1 * cos(\phi) \\ d * K_1 * PWM_1 * cos(\theta_1) * cos(\phi) \\ -d * K_1 * PWM_1 * cos(\theta_1) * cos(\phi) \\ d * K_1 * PWM_1 * sin(\phi) \end{bmatrix} \qquad \text{(Eqn. 2.2)}$$

$$\begin{bmatrix} T_{tx1} \\ T_{ty1} \\ T_{tz1} \end{bmatrix} = \begin{bmatrix} -K_2 * PWM_1 * cos(\theta_1) sin(\phi) \\ K_2 * PWM_1 * cos(\theta_1) sin(\phi) \\ K_2 * PWM_1 * cos(\phi) \end{bmatrix} \quad \text{(Eqn. 2.3)}$$

where K_1 and K_2 is the constant between force and torque produced by Motor1 and PWM_1(the pulse width modulation command sent to Motor1), φ is the cant angle from vertical, θ represents the rotor's position, d is the distance from rotor 1 to the central axis (the radius of the Dexterous Hexrotor). $[F_{1fx}\ F_{1fy}\ F_{1fz}\ F_{1\tau x}\ F_{1\tau y}\ F_{1\tau z}]^T$ are forces and torques decomposed from F_1 (the force produced by Motor1) and $[\tau_{1\tau x}\ \tau_{1\tau y}\ \tau_{1\tau z}]^T$ are torques decomposed from τ_1 (the torque produced by Motor1).

Then we can get the total force/torque $[F_{1x}\ F_{1y}\ F_{1z}\ \tau_{1x}\ \tau_{1y}\ \tau_{1z}]^T$ acting on the Dexterous Hexrotor from Motor1.

$$\begin{aligned} \begin{bmatrix} F_{1x} \\ F_{1y} \\ F_{1z} \\ \tau_{1x} \\ \tau_{1y} \\ \tau_{1z} \end{bmatrix} &= \begin{bmatrix} F_{1fx} \\ F_{1fy} \\ F_{1fz} \\ F_{1\tau x} + \tau_{1\tau x} \\ F_{1\tau y} + \tau_{1\tau y} \\ F_{1\tau y} + \tau_{1\tau z} \end{bmatrix} \\ &= \begin{bmatrix} -K_1 * PWM_1 * cos(\theta_1) * sin(\phi) \\ K_1 * PWM_1 * sin(\theta_1) * sin(\phi) \\ K_1 * PWM_1 * cos(\phi) \\ PWM_1 * cos(\theta_1) * (d * K_1 * cos(\phi) - K_2 * sin(\phi)) \\ PWM_1 * cos(\theta_1) * (-d * K_1 * cos(\phi) + K_2 * sin(\phi)) \\ PWM_1 * (d * K_1 * sin(\phi) + K_2 * cos(\phi)) \end{bmatrix} \\ &= PWM_1 * \begin{bmatrix} -K_1 C\theta_1 S\phi \\ K_1 S\theta_1 S\phi \\ K_1 C\phi \\ C\theta_1 (dK_1 C\phi - K_2 S\phi) \\ C\theta_1 (-dK_1 C\phi + K_2 S\phi) \\ dK_1 S\phi + K_2 C\phi \end{bmatrix} \end{aligned} \quad \text{(Eqn. 2.4)}$$

Once one rotor is decomposed, we can follow the same pattern and decompose forces and torques produced by other motors and get equations (2.5, 2.6, 2.7, 2.8, 2.9)

$$\begin{bmatrix} F_{2x} \\ F_{2y} \\ F_{2z} \\ \tau_{2x} \\ \tau_{2y} \\ \tau_{2z} \end{bmatrix} = PWM_2 * \begin{bmatrix} K_1 C\theta_2 S\phi \\ -K_1 S\theta_2 S\phi \\ K_1 C\phi \\ C\theta_2(dK_1 C\phi - K_2 S\phi) \\ S\theta_2(dK_1 C\phi - K_2 S\phi) \\ -dK_1 S\phi - K_2 C\phi \end{bmatrix} \quad \text{(Eqn. 2.5)}$$

$$\begin{bmatrix} F_{3x} \\ F_{3y} \\ F_{3z} \\ \tau_{3x} \\ \tau_{3y} \\ \tau_{3z} \end{bmatrix} = PWM_3 * \begin{bmatrix} K_1 C\theta_3 S\phi \\ K_1 S\theta_3 S\phi \\ K_1 C\phi \\ C\theta_3(dK_1 C\phi + K_2 S\phi) \\ S\theta_3(-dK_1 C\phi + K_2 S\phi) \\ dK_1 S\phi + K_2 C\phi \end{bmatrix} \quad \text{(Eqn. 2.6)}$$

$$\begin{bmatrix} F_{4x} \\ F_{4y} \\ F_{4z} \\ \tau_{4x} \\ \tau_{4y} \\ \tau_{4z} \end{bmatrix} = PWM_4 * \begin{bmatrix} K_1 C\theta_4 S\phi \\ -K_1 S\theta_4 S\phi \\ K_1 C\phi \\ C\theta_4(dK_1 C\phi - K_2 S\phi) \\ S\theta_4(-dK_1 C\phi + K_2 S\phi) \\ -dK_1 S\phi - K_2 C\phi \end{bmatrix} \quad \text{(Eqn. 2.7)}$$

$$\begin{bmatrix} F_{5x} \\ F_{5y} \\ F_{5z} \\ \tau_{5x} \\ \tau_{5y} \\ \tau_{5z} \end{bmatrix} = PWM_5 * \begin{bmatrix} -K_1 C\theta_5 S\phi \\ K_1 S\theta_5 S\phi \\ K_1 C\phi \\ C\theta_5(dK_1 C\phi - K_2 S\phi) \\ S\theta_5(-dK_1 C\phi + K_2 S\phi) \\ dK_1 S\phi + K_2 C\phi \end{bmatrix} \quad \text{(Eqn. 2.8)}$$

$$\begin{bmatrix} F_{6x} \\ F_{6y} \\ F_{6z} \\ \tau_{6x} \\ \tau_{6y} \\ \tau_{6z} \end{bmatrix} = PWM_6 * \begin{bmatrix} K_1 C\theta_6 S\phi \\ -K_1 S\theta_6 S\phi \\ K_1 C\phi \\ C\theta_6(dK_1 C\phi + K_2 S\phi) \\ S\theta_6(-dK_1 C\phi + K_2 S\phi) \\ -dK_1 S\phi - K_2 C\phi \end{bmatrix} \quad \text{(Eqn. 2.9)}$$

2.3 Mapping from actuator space to force/torque space

After decomposition of force/torque produced by each motor, now we can compute the net force/torque of the Dexterous Hexrotor as equation (2.10).

$$\begin{bmatrix} F_x \\ F_x \\ F_x \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = \begin{bmatrix} F_{1x} \\ F_{1y} \\ F_{1z} \\ \tau_{1x} \\ \tau_{1y} \\ \tau_{1z} \end{bmatrix} + \begin{bmatrix} F_{2x} \\ F_{2y} \\ F_{2z} \\ \tau_{2x} \\ \tau_{2y} \\ \tau_{2z} \end{bmatrix} + \begin{bmatrix} F_{3x} \\ F_{3y} \\ F_{3z} \\ \tau_{3x} \\ \tau_{3y} \\ \tau_{3z} \end{bmatrix} + \begin{bmatrix} F_{4x} \\ F_{4y} \\ F_{4z} \\ \tau_{4x} \\ \tau_{4y} \\ \tau_{4z} \end{bmatrix} + \begin{bmatrix} F_{5x} \\ F_{5y} \\ F_{5z} \\ \tau_{5x} \\ \tau_{5y} \\ \tau_{5z} \end{bmatrix} + \begin{bmatrix} F_{6x} \\ F_{6y} \\ F_{6z} \\ \tau_{6x} \\ \tau_{6y} \\ \tau_{6z} \end{bmatrix} \quad \text{(Eqn. 2.10)}$$

Where $[F_x\ F_y\ F_z\ \tau_x\ \tau_y\ \tau_z]^T$ is the net force/torque act on the body of the Dexterous Hexrotor.

From equations (2.4 2.5, 2.6, 2.7, 2.8, 2.9), the right part of equation equals this 6×6 conversion matrix multiplied by $[PWM_1\ PWM_2\ PWM_3\ PWM_4\ PWM_5\ PWM_6]^T$.

$$\begin{bmatrix} -K_1C\theta_1S\phi & K_1C\theta_2S\phi & K_1C\theta_3S\phi & K_1C\theta_4S\phi & K_1C\theta_5S\phi & K_1C\theta_6S\phi \\ K_1S\theta_1S\phi & -K_1S\theta_2S\phi & K_1S\theta_3S\phi & -K_1S\theta_4S\phi & K_1S\theta_5S\phi & -K_1S\theta_6S\phi \\ K_1C\phi & K_1C\phi & K_1C\phi & K_1C\phi & K_1C\phi & K_1C\phi \\ C\theta_1(dK_1C\phi - K_2S\phi) & C\theta_2(dK_1C\phi - K_2S\phi) & C\theta_3(dK_1C\phi + K_2S\phi) & C\theta_4(dK_1C\phi - K_2S\phi) & C\theta_5(dK_1C\phi - K_2S\phi) & C\theta_6(dK_1C\phi + K_2S\phi) \\ S\theta_1(-dK_1C\phi + K_2S\phi) & S\theta_2(dK_1C\phi - K_2S\phi) & S\theta_3(-dK_1C\phi + K_2S\phi) & S\theta_4(-dK_1C\phi + K_2S\phi) & S\theta_5(-dK_1C\phi + K_2S\phi) & S\theta_6(-dK_1C\phi + K_2S\phi) \\ dK_1S\phi + K_2C\phi & -dK_1S\phi - K_2C\phi & dK_1S\phi + K_2C\phi & -dK_1S\phi - K_2C\phi & dK_1S\phi + K_2C\phi & -dK_1S\phi - K_2C\phi \end{bmatrix}$$

Express this matrix as M_φ. We can conclude this equation as

$$\begin{bmatrix} F_x \\ F_y \\ F_z \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = M_\phi \cdot \begin{bmatrix} PWM_1 \\ PWM_2 \\ PWM_3 \\ PWM_4 \\ PWM_5 \\ PWM_6 \end{bmatrix} \quad \text{(Eqn. 2.11)}$$

In the Dexterous Hexrotor, $[F_x\ F_y\ F_z\ \tau_x\ \tau_y\ \tau_z]^T$ is the desired force/torque vector. Values of this vector will decide the robot's orientation and position. $[PWM_1\ PWM_2\ PWM_3\ PWM_4\ PWM_5\ PWM_6]^T$ are six independent controlled inputs for the robot. Based on this equation, we can establish a relationship between inputs (PWM commands) and outputs

(force/torque) of the Dexterous Hexrotor and build a mapping from UAV's actuator space to Cartesian force/torque space. Therefore, to control the Dexterous Hexrotor and get desired force/torque vector, we multiply the inversed conversion matrix by the desired force/torque vector, get PWM commands and sent them to the motors. The force/torque control equation can be derived as

$$\begin{bmatrix} PWM_1 \\ PWM_2 \\ PWM_3 \\ PWM_4 \\ PWM_5 \\ PWM_6 \end{bmatrix} = [M_\phi]^{-1} \cdot \begin{bmatrix} F_x \\ F_y \\ F_z \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} \quad \text{(Eqn. 2.12)}$$

The conversion matrix is the mapping from actuator space to force/torque space. With K1 and K2 determined for our motors, if the cant angle is zero. The conversion matrix is M_{0°:

$$M_{0^\circ} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 5.7 & 5.7 & 5.7 & 5.7 & 5.7 & 5.7 \\ 1.33 & 1.3 & 0 & -1.33 & -1.33 & 0 \\ -0.769 & 0.769 & 1.54 & 0.769 & -0.769 & -1.54 \\ 0.13 & -0.13 & 0.13 & -0.13 & 0.13 & -0.13 \end{bmatrix}$$

which would conform to a typical hexrotor design. M_{0° only has a rank of 4. We have no ability to instantaneously control forces in X and Y axes through this matrix.

If we cant the thrusters at an angle, for example, at 20°, the thrust mapping becomes M_{20°:

$$M_{20^\circ} = \begin{bmatrix} -1.69 & 1.69 & 0 & -1.69 & 1.69 & 0 \\ 0.975 & 0.975 & -1.95 & 0.975 & 0.975 & -1.95 \\ 5.36 & 5.36 & 5.36 & 5.36 & 5.36 & 5.36 \\ 1.21 & 1.21 & 0 & -1.21 & -1.21 & 0 \\ -0.701 & 0.701 & 1.4 & 0.701 & -0.701 & -1.4 \\ 0.649 & -0.649 & 0.649 & -0.649 & 0.649 & -0.649 \end{bmatrix}$$

which would conform to a nonparallel design. This matrix provides a mapping from 6-D actuator space to 6-D force/torque space and has a rank of 6, indicating we have 6 independent controlled degrees of freedom in Cartesian force/torque space.

2.4 Dexterous Hexrotor Performance Optimization

The cant angle decides how much force we can put into $[F_z\ \tau_x\ \tau_y]^T$ or $[F_x\ F_y\ \tau_z]^T$. If $\varphi = 0^\circ$, the matrix becomes:

$$M_{0^\circ} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 5.7 & 5.7 & 5.7 & 5.7 & 5.7 & 5.7 \\ 1.33 & 1.3 & 0 & -1.33 & -1.33 & 0 \\ -0.769 & 0.769 & 1.54 & 0.769 & -0.769 & -1.54 \\ 0.13 & -0.13 & 0.13 & -0.13 & 0.13 & -0.13 \end{bmatrix}$$

Clearly the matrix becomes degenerate as there are no forces in X or Y from rotor force/torque decomposition, this would conform to a fairly typical quad-rotor design. There's no mapping from actuator space to forces in X and Y. It's rank deficient. The opposite happens when φ is 90°:

$$M_{90^\circ} = \begin{bmatrix} -4.94 & 4.94 & 0 & -4.94 & 4.94 & 0 \\ 2.85 & 2.85 & -5.7 & 2.85 & 2.85 & -5.7 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -0.11 & -0.11 & 0 & 0.11 & 0.11 & 0 \\ 0.065 & -0.065 & -0.13 & -0.065 & 0.065 & 0.13 \\ 1.54 & -1.54 & 1.54 & -1.54 & 1.54 & -1.54 \end{bmatrix}$$

Force can be applied in X and Y but no lift and the matrix becomes deficient again. With these examples in mind it is obvious that the closer to $\varphi = 0^\circ$ the more the Dexterous Hexrotor can lift, but the closer to $\varphi = 90^\circ$ the more lifting force can be obtained while tilted, or the more the Dexterous Hexrotor can tilt.

A cant angle has to be chosen somewhere between 0° and 90° when we build the Dextrous Hexrotor. So a metric for optimization of the Dextrous Hexrotor UAV's performance has been developed.

2.4.1 Engineering Requirement

The performance of the Dextrous Hexrotor at different cant angles is affected by several variables, including the thrust of the motors, the diameter of the UAV, and the load we want to carry. For manipulation tasks, the engineering requirement for the Dextrous Hexrotor is the desired payload. We will set the desired payload before each task.

So for the Dextrous Hexrotor engineering prototype, the thrust of the motors and the diameter of the UAV are given based on the motors and frame we chosen. Given the system's own mass and desired payload, the load we want to carry are also set. The only thing undecided is the cant angle.

Table I. Parameters of the Dextrous Hexrotor engineering prototype

K1	5.7
K2	1.3
Diameter	550 cm
Mass of UAV	2 kg
Mass of Manipulator	0.5 kg
Desired Payload	0.3 kg

With these parameters, we can plot maximum forces and torques the Dexterous Hexrotor can achieve near hover condition at different cant angles in Fig. 2.8.

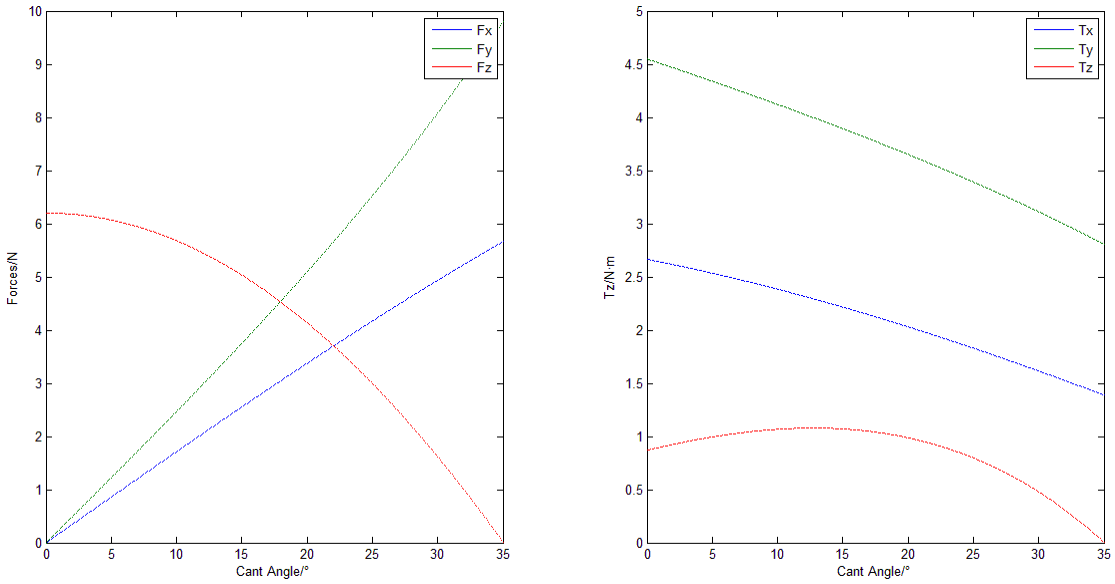


Fig.2.8 Maximum forces the Dexterous Hexrotor can achieve at different cant angles

Clearly at φ = 0° , there are no forces in X or Y and not much torques around Z. The opposite happens when φ = 35° , where all forces in Z are used to provide lift, but more force can be applied in X and Y. This means the Dexterous Hexrotor engineering

prototype can operate from 0° to 35° . So a cant angle between 0° to 35° should be chosen.

2.4.2 Cant Angle Optimization

To optimize the cant angle for the performance of our UAV, we adapt Yoshikawa's concept of ``manipulability" to ours. As defined by Yoshikawa, ``manipulability measure" is a quantitative measure of manipulating ability of robot arms in positioning and orienting the end-effectors, by looking at the isotropism of manipulator's motion in linear dimensions X, Y, Z and angles roll, pitch, yaw [40]. So for our UAV, we consider the combination of forces and torques as a similar measure of mobility. We are going to look at the isotropism of the forces and torques, not just how strong is it. To visualize how isotropic are the forces and torques, we plot these force/torque ellipsoids as shown in Fig. 2.9.

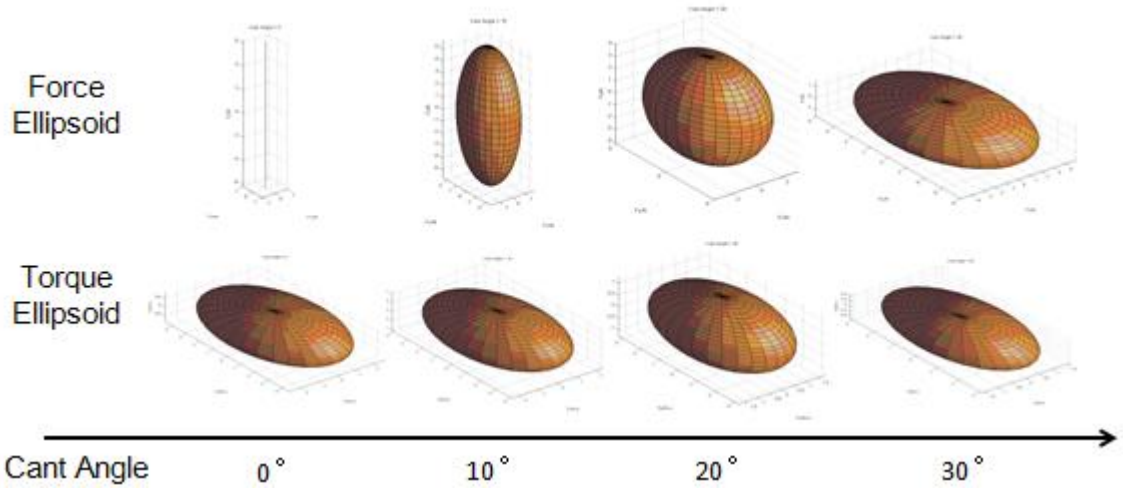


Fig. 2.9 Force/Torque ellipsoids at different cant angles

The radius of these ellipsoids represents the magnitude of maximum force/torque the Dexterous Hexrotor can achieve around X, Y, Z axes at near hover condition. When the cant angle is 0° , we get no ability to control F_x, F_y, but a lot of ability to control F_z. We also have ability to control τ_x τ_y τ_z, but not too much control over yaw because coriolis effect is weak. Then we cant the motor a little bit, we get a little bit control over F_x, F_y, but still very strong control in F_z. And likewise we get a little more control over yaw. At 20° , the force ellipsoid becomes almost equal and the torque ellipsoid also gets more round. At 30° , both force ellipsoid and torque ellipsoid start to squash down again. Therefore we can see from 0° to 30° and beyond, the force/torque ellipsoid can get very isotropic at some point, which is good in our mobility measure.

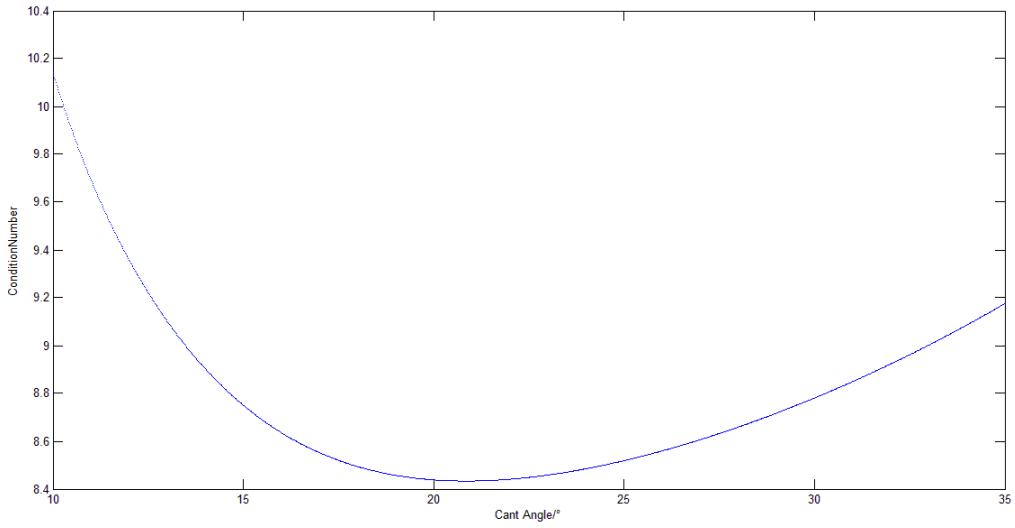


Fig. 2.10 Condition Number of the conversion matrix

We can also look at the condition number of the conversion matrix. Condition number is a mathematic term from linear algebra, which is the ratio of maximum eigenvalue to the minimum eigenvalue of the matrix. We can see it as the rate at which one side of the

equation will change with respect to the other side. If the condition number is large, even a small error in one side may cause a large error in the other side.

At 0° , when we have no control over Fx and Fy, the condition number would be infinite, because two of the eigenvalues are 0. Then it can get smaller and smaller when we increase the cant angle. It is hard getting to 1 since force and torque are measured under different scale. Eventually it gets higher again and it will be infinite at 90° , because one eigenvalue is 0.

We can combine these two metrics on a same plot as shown in Fig. 2.11. We plot the condition number of the conversion matrix with force/torque ellipsoids at different cant angles, giving us a measure of the isotropism of our UAV.

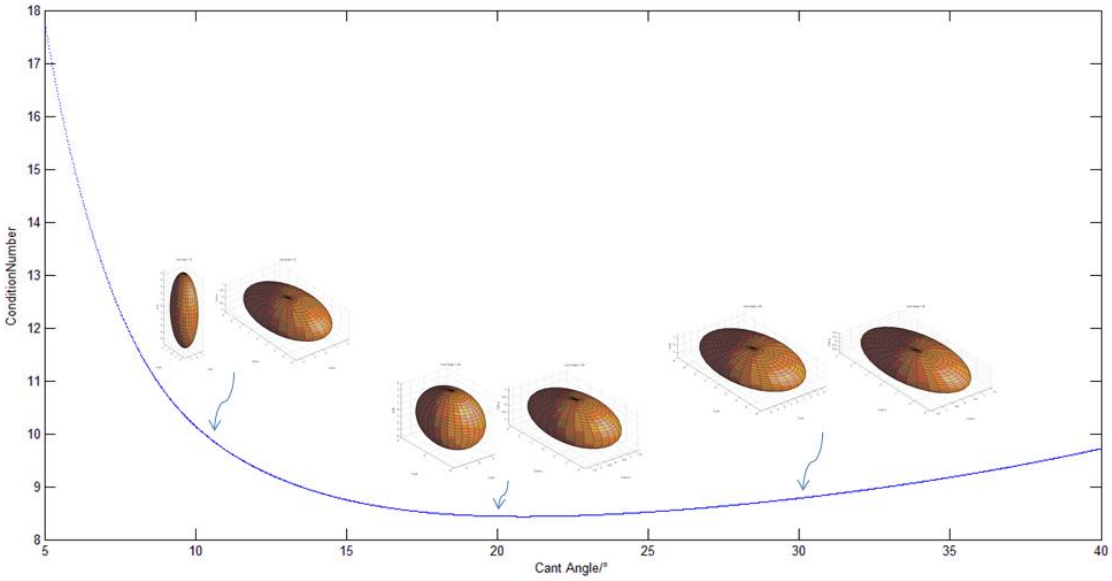


Fig. 2.11 Condition number with force/torque ellipsoids

Therefore, dependent on the motors we chosen, particular load of our manipulator, diameter of the UAV frame, we optimize the cant angle at 20° for this Dextrous Hexrotor engineering prototype. If we change the load or any other parameters, we would

optimize this for a different cant angle. And for tasks other than manipulation, we can optimize its performance with other parameters as well as the cant angle.

Chapter Three: Dexterous Hexrotor Control

For human in the loop flight, the pilot controls the Dexterous Hexrotor's orientation and position by manipulating the forces and torques acting on the platform. For the Dexterous Hexrotor, there are six independent controlled forces and torques acting on itself. Therefore, control systems to help the pilot control the Dexterous Hexrotor's orientation and position have been developed and are described here.

For simple flight in unstructured environments, the control system of the Dexterous Hexrotor is only implemented with an attitude controller and a low level translational force controller.

For indoor 3D translational flight, a position controller using an 8-camera Vicon MXT40 motion capture system providing real-time position feedback has been implemented to augment the same attitude controller.

3.1 Dexterous Hexrotor Control System

Forces and torques acting on the Dexterous Hexrotor are generated by the thrust of each rotor. And the thrust is proportional to the PWM value put into the motor. Based on the force/torque control equation, to generate desired force/torque, PWM values which are calculated from desired force/torque vector $[Fx\ Fy\ Fz\ \tau x\ \tau y\ \tau z]^T$ through thrust mapping, need to be sent to each motor.

Multicopters are inherently unstable in the air. Therefor an attitude controller is needed to stabilize the UAV during flight. A common control system used in quadrotors is as shown in Fig. 3.1.

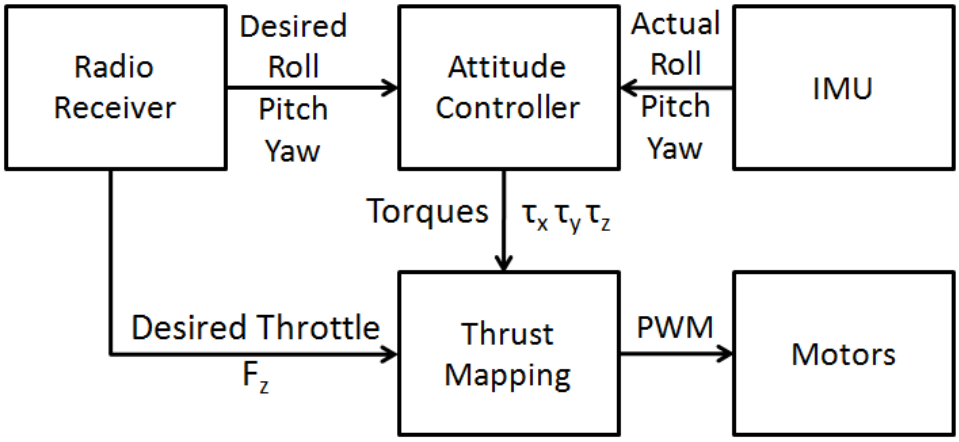


Fig. 3.1 Typical Quadrotor control system

This system is constructed by radio receiver, attitude controller, IMU(inertial measurement unit), thrust mapping and motors. With help of the attitude controller stabilizing the UAV, the pilot only needs to control its attitude to change the orientation and position (e.g. pitching for moving forward).

The difference between controlling the Dexterous Hexrotor and a quadrotor is the thrust mapping. Thrust mapping of quadrotor only maps $[Fz\ \tau x\ \tau y\ \tau z]^T$ to motors. Dexterous Hexrotor's thrust mapping needs to map both force and torque around each of three axes to the motors. So the control system should be as it is shown in Fig. 3.2.

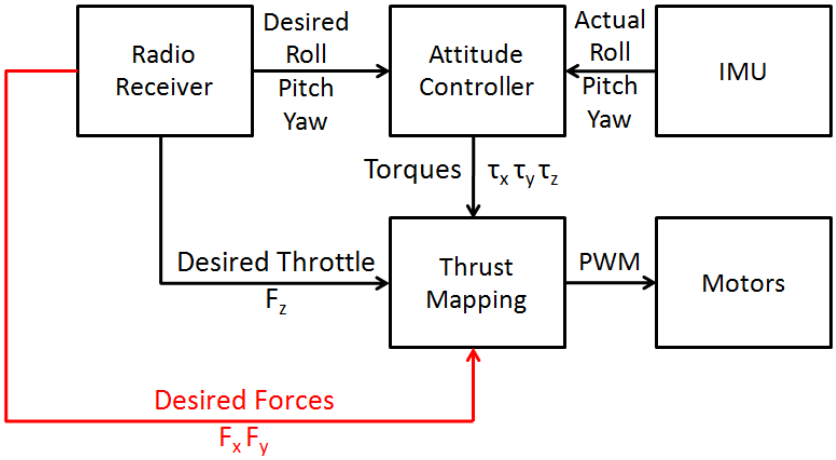


Fig. 3.2 Dexterous Hexrotor simple flight control system

Thrust mapping in this system maps [Fx Fy Fz τx τy τz]^T to [PWM1 PWM2 PWM3 PWM4 PWM5 PWM6]^T. It is a mapping from 6D actuator space to 6D force/torque space.

The user interface to the control system of the Dexterous Hexrotor has two more channels controlling Fx and Fy than a quadrotor. These forces are in the plane perpendicular to the rotor's axis. They can provide horizontal acceleration without pitching the body. These forces can be used to exert forces immediately in the plane and they also give the pilot another option of moving the Dexterous Hexrotor. When the pilot wants to move a quadrotor, it needs to pitch or roll an angle in order to get the translational acceleration. Dexterous Hexrotor can simply generate the translational acceleration by Fx or Fy.

The difference between control system of Dexterous Hexrotor and typical quadrotor is as shown in Table II.

Table II. Difference between control system of the Dexterous Hexrotor and typical quadrotors

	quadrotor	Dexterous Hexrotor
Pilot controlled variables	Roll, Pitch, Yaw, Fz	Roll, Pitch, Yaw, Fx, Fy, Fz
Number of Pilot controlled variables	4	6
Inputs to thrust mapping	F_z τ_x τ_y τ_z	F_x F_y F_z τ_x τ_y τ_z
Number of Inputs to thrust mapping	4	6
Outputs of thrust mapping	PWM_1 PWM_2 PWM_3 PWM_4	PWM_1 PWM_2 PWM_3 PWM_4 PWM_5 PWM_6

For indoor 3D translational flight, a control system implemented with both attitude controller and position controller has been developed with an 8-camera Vicon MXT40 motion capture system.

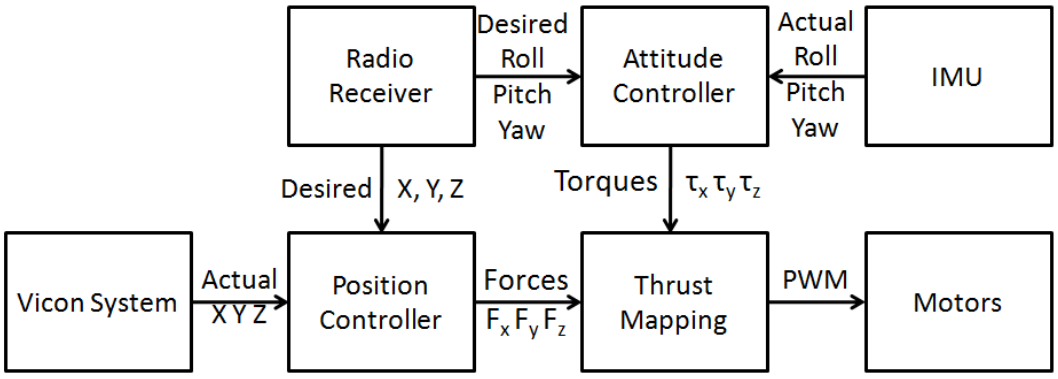


Fig. 3.3 Dexterous Hexrotor indoor 3D translational flight control system

In this control system, a Vicon motion capture system is used to provide real-time position of the Dexterous Hexrotor. Then a position controller uses this information to adjust the UAV's position by controlling the forces acted on the platform. Since the Dexterous Hexrotor's orientation and position is controlled by two separate controllers, now theoretically the Dexterous Hexrotor can achieve 3D translational flight, means

rotation without translation and translation without rotation. This is what typical quadrotors cannot do, because they can only control 4 DoFs in force/torque space.

3.2 Hexrotor Attitude Controller

The attitude controller is designed to stabilize Dexterous Hexrotor for hunman controlled flight. It helps Dexterous Hexrotor maintaining desired attitude during flight by controlling Dexterous Hexrotor roll, pitch and yaw. This is also a commonly used control structure in quadrotor commutity.

For stability and also fast response, three double loop PID controllers for roll, pitch and yaw have been developed and they share same structure as it is shown in Fig. 3.3.

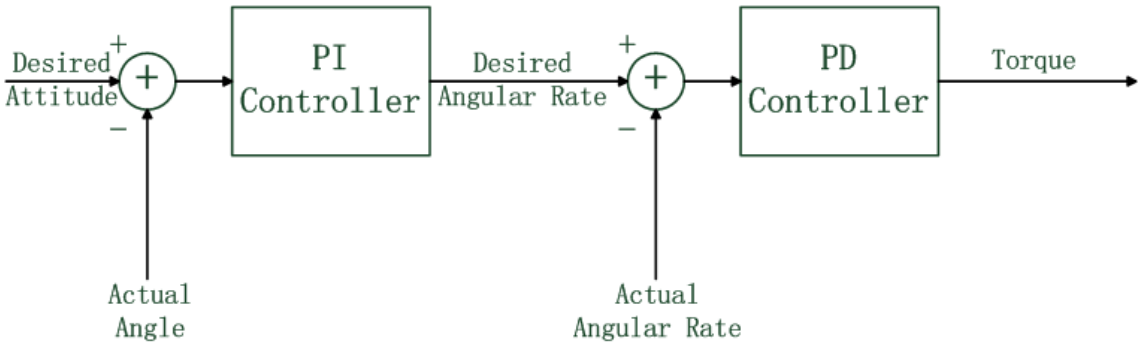


Fig. 3.4 Double loop PID controller

The outer loop is using a PI controller for controlling the angle. It takes desired angle as input, actual IMU angle as feedback, and outputs desired angular rate to the inner loop.

The inner loop is using a PD controller for controlling angular rate. It takes angular rate as input, actual IMU angular rate as feedback, and outputs torques to adjust the attitude of the Dexterous Hexrotor.

The PI, PD combination is chosen for a good reason. A single traditional PID controller would work, but slow. An outer loop controlling angle and an inner loop controlling rate

would be much better for its fast response. P item is necessary in both loops for fast response. A D term in the inner loop is for improving response time. An I term in the outer loop is for helping Dexterous Hexrotor dealing with persistent external forces, like wind or incorrect center of gravity.

Similar control structure is also implemented in the position controller. Actual position and its change rate will be provided by the Vicon motion capture system and sent to the Dexterous Hexrotor.

Chapter Four: Dexterous Hexrotor Prototype

4.1 Design of Dexterous Hexrotor Prototype

A prototype that uses off the shelf components, preferably those commonly used by modern research quadrotors has been built improve cost, compatibility and simplicity. The prototype of Dexterous Hexrotor design can be seen in Fig. 4.1.

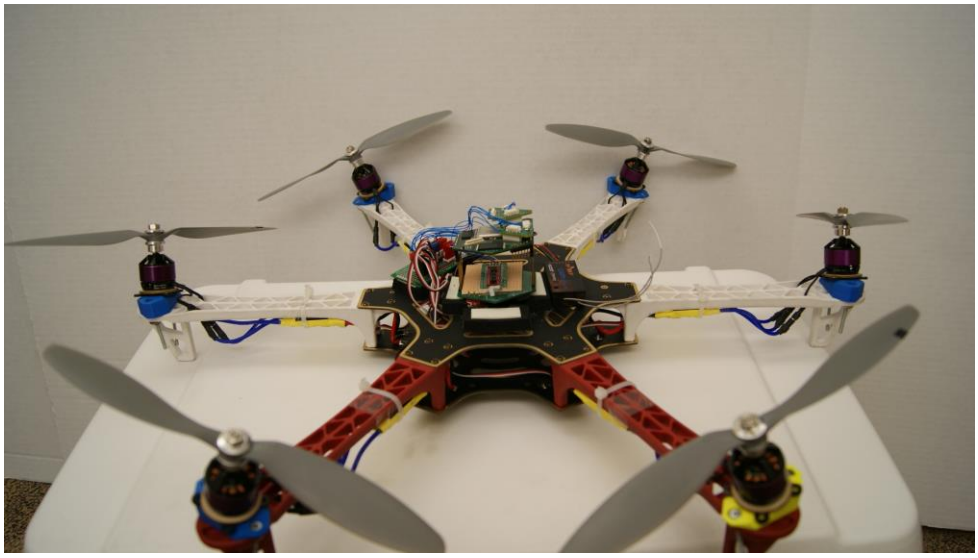


Fig. 4.1 Dexterous Hexrotor prototype built on a commercial frame

The motors are mounted on in house designed and fabricated ABS plastic adapters that canted 20° tangentially to the edge. These plastic adapters are then mounted on the end of each arm.

Fig. 4.2 Motor mounted on a 20° ABS plastic adapter

Standard 11.1V lithium polymer battery packs are used to power the 30A ESCs (electronic speed controller). Each ESC drives an 1130KV brushless DC motor with a three phase signal converted from the PWM command. The motors are installed with 9"x4.7" pusher or puller propellers.

A 2.4G Hz radio transmitter and receiver set are used for human controlled flight. Four main joysticks are used for roll, pitch, yaw and throttle. Two side knobs are used to set F_x and F_y values.

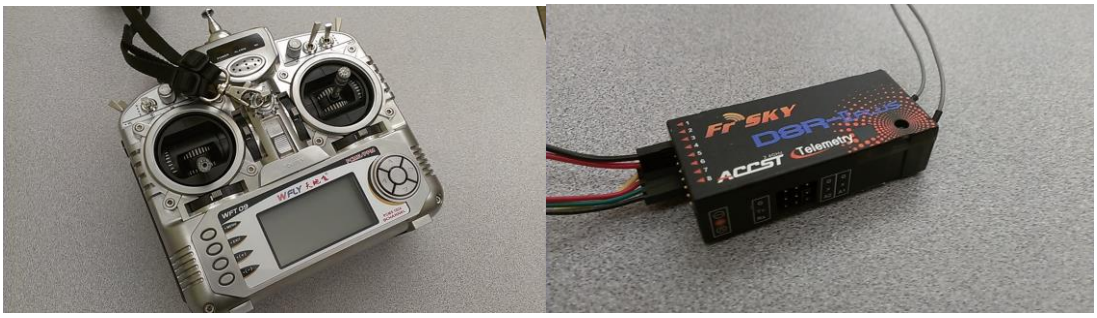


Fig. 4.3 WFly09 R/C controller and receiver

The controller we used is the RecoNode platform developed in our lab. A universal PCB board is used for routing signals between on board electronics, the RecoNode Platform, IMU, radio receiver, and ESCs.

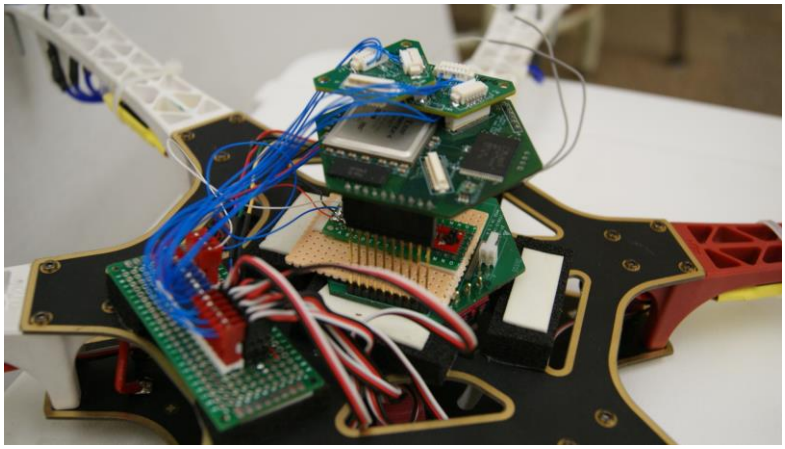


Fig. 4.4 On board electronics of Dexterous Hexrotor

A Sparkfun 9 DoF sensor stick is used as IMU. It includes an ADXL345 accelerometer, a HMC5883L magnetometer, and an ITG-3200 gyro. It communicates with CPU by I2C interface.

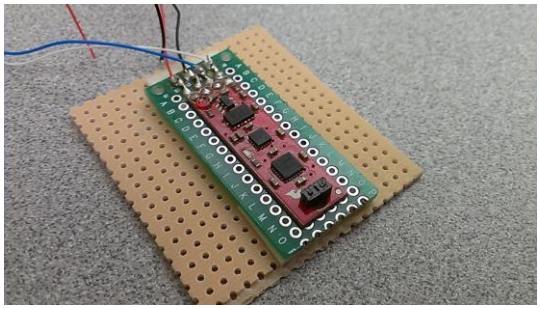


Fig. 4.5 Sparkfun 9 DoF sensor stick

An 8-camera Vicon MXT40 motion capture system is used as indoor real-time position feedback. This system contains Infrared reflective markers, MX-F40 cameras, MX Ultranet and a host computer. With markers on the UAV, this system can measure UAV's position and even attitude. All information can be sent to the UAV through a WIFI module.

Fig. 4.6 MX-F40 Camera

4.2 RecoNode

The RecoNode is a multi-processor architecture based on the Virtex 4 FPGA with multiple, hardcore PowerPCs [35]. Capable of up to 1600 MIPS from four PowerPCs plus hundreds of additional MIPS of special-purpose coprocessing from the FPGA fabric itself, this computational node is very high performance compared to conventional wireless sensor nodes - roughly 100 times greater computational throughput. The RecoNodes we have built are dual-processor models running at 400 MIPS with a power budget of 0.9 mW/MIPS.

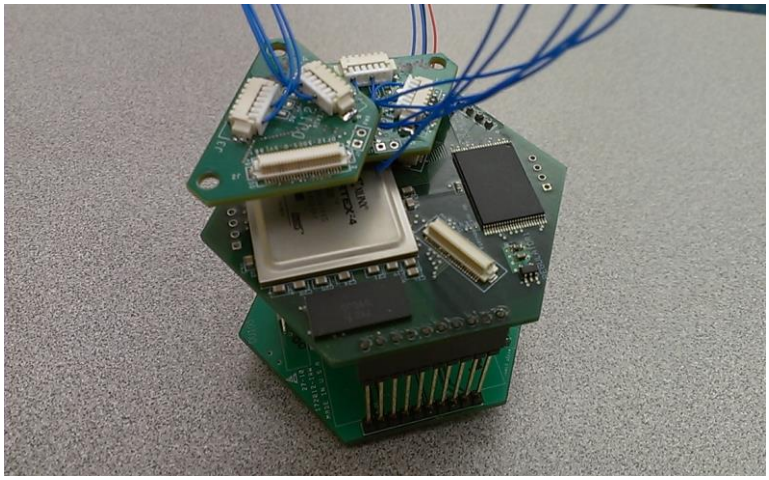
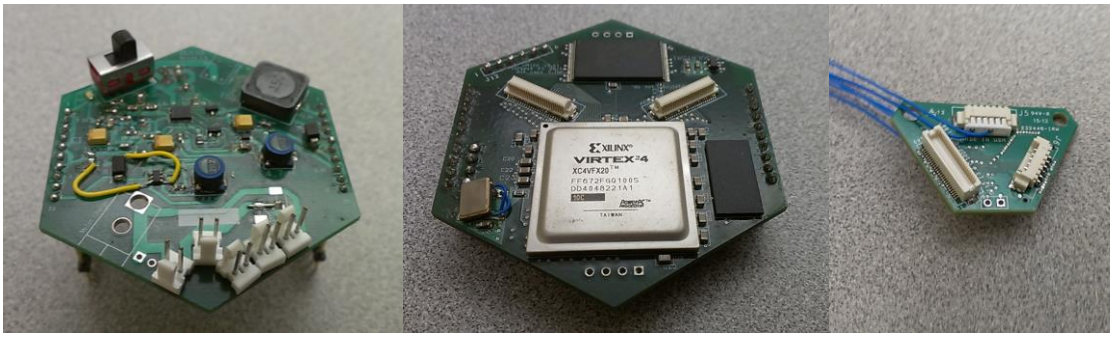


Fig. 4.7 The RecoNode stack withVirtex4 FX20 CPU and two single-width wedges

The RecoNode we used in Dexterous Hexrotor has two base boards and two same wedges, from bottom to top: DU105 power board, providing 1.2V, 2.5V, 3.3V and 5V for the system; DU100 CPU board, carrying a Virtex4 FX20 CPU running at 100M Hz; DU121 servo wedge, transferring the PWM commands from CPU to ESCs and input signals from radio receiver and IMU to CPU.



(a) (b) (c)

Fig. 4.8Wedges of RecoNode (a) DU105 power board, (b) DU100 CPU board, (c) DU121 servo wedge

Chapter Five: Experiments and Results

In order to verify the functionality of the Dexterous Hexrotor and its ability to achieve force closure, several experiments and tests have been done.

5.1 Force/Torque Test

As force closure in 3-D space requires six controllable degrees of freedom in force/torque space to truly accomplish. To prove the Dexterous Hexrotor can apply force/torque in all six degrees of freedom or only one degree of freedom without affecting the others. We bolted the Dexterous Hexrotor on an ATI 6 DoF force sensor and tried to produce forces and torques in all three dimensions. The test setup and ATI force sensor software are as shown in Fig. 5.1.

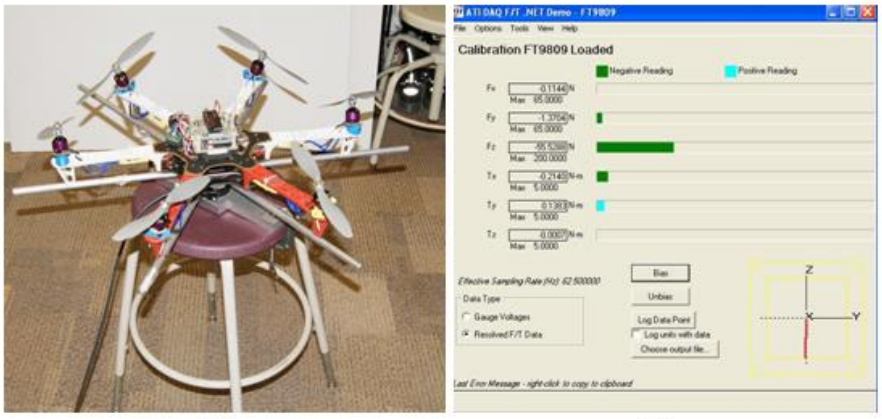


Fig. 5.1 Force/torque test setup: (a) The Dexterous Hexrotor bolted on an ATI 6 DoF force sensor. (b) ATI 6 DoF force sensor software.

During the test, unit forces and torques in the force/torque vector $[F_x\ F_y\ F_z\ \tau_x\ \tau_y\ \tau_z]^T$ were commanded sequentially for around five seconds each at near hover condition.

For five seconds with a five second off after, each the script requests a force in X, then Y, then Z axis. Then it requests a torque about X, then Y, and then Z axis. The result of the test is as shown in Fig 5.2 and tabulated in table I.

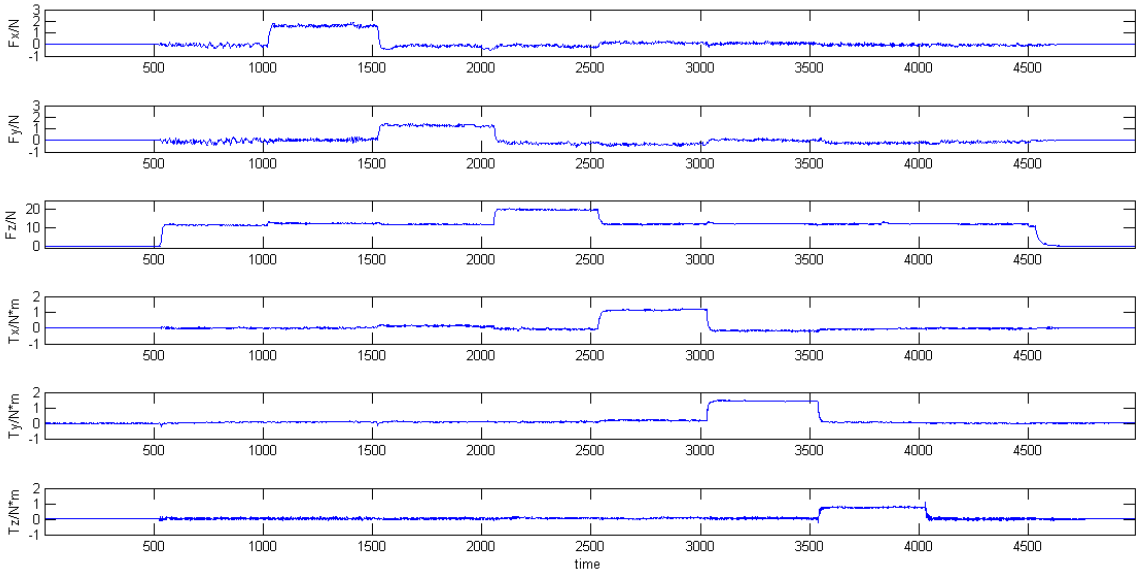


Fig. 5.2 Force/Torque plot recorded by an ATI 6-DOF force sensor

In this plot, the first 5 seconds all motors are off. It is apparent from F_z that at roughly 5 seconds, the Dexterous Hexrotor "takes off", then the sequence begins. Data was recorded by an ATI 6-DOF force sensor at 10kSamp/s. Time is in milliseconds.

Table III. Average value of each period in the Force/Torque plot.

	T1	T2	T3	T4	T5	T6	T7
F_x(N)	-0.08	1.59	-0.07	-0.09	0.06	0.03	-0.03
F_y(N)	-0.04	0.02	1.30	-0.06	-0.05	0.03	-0.04
F_z(N)	11.56	11.96	11.93	19.94	11.67	11.43	11.59
$T_x(N \cdot m)$	-0.01	0	0.06	0	1.13	-0.07	-0.05
$T_y(N \cdot m)$	0.04	0.08	0.08	0.08	0.09	1.44	0.07
$T_z(N \cdot m)$	0.04	0.04	0.04	0.03	0.03	0.05	0.75

In table II, T1 is from 5s to 10s, which shows half throttle. T2 is from 10s to 15s, when the Dexterous Hexrotor is generating a force in x axis. T3 is from 15s to 20s, when the Dexterous Hexrotor is generating a force in y axis. T4 is from 20s to 25s, when the Dexterous Hexrotor raises its throttle. T5 is from 25s to 30s, when the Dexterous Hexrotor is generating a torque around x axis. T6 is from 30s to 35s, when the Dexterous Hexrotor is generating a torque around y axis. T7 is from 35s to 40s, when the Dexterous Hexrotor is generating a torque around z axis. This data was produced by taking the time average of the central 3 second of each period.

Table IV. Percentage error of each force compared to the magnitude of force vector.

	T1	T2	T3	T4	T5	T6	T7
$F_x(\%)$	0.69	-	0.58	0.45	0.51	0.26	0.26
$F_y(\%)$	0.34	0.17	-	0.30	0.43	0.26	0.35
$F_z(\%)$	-	-	-	-	-	-	-

In this table, we compare error of each force to the magnitude of force vector we required. For example, during T1, we require a force vector in Z axis with a magnitude of 11.56 N. F_x and F_y should be zero during T1, but they are not. So we compare values of F_x and F_y to the magnitude of the force vector and get percentage error of 0.08/11.56 = 0.69% and 0.04/11.56 = 0.34%. It's same in T4, T5, T6, and T7, but different in T2 and T3. During T2 and T3, we require a force vector point along a direction between Z axis and X or Y

axis. Since we can observe error no more than 1%, it proves we can actually control force in each axis accurately with no coupling to the other axes.

Table V. Percentage error of each torque compared to its peak value.

	T1	T2	T3	T4	T5	T6	T7
$T_x(\%)$	0.8	0	5.3	0	-	6.2	4.4
$T_y(\%)$	2.7	5.4	5.4	5.4	6.2	-	4.8
$T_z(\%)$	5.3	5.3	5.3	4.0	4.0	6.6	-

We can also see error of each torque compared to its peak value. For example, T_x during T1 should be 0, as we are giving a force/torque command of $[0\ 0\ F_z\ 0\ 0\ 0]^T$, producing only a force vector in Z axis. So we compare the value of T_x during this period to its peak value happened in T5 and get its error 0.01/1.13 = 0.8%. Values of torques are much smaller than the magnitude of force vector, so their percentage error looks much larger. But still we can observe at most 6.6% error of torques, proving that we can actually control torque in each axis with no coupling to the other axes.

With the ability of controlling force/torque in each axis with no coupling to the other axes, it shows full controllability over six degree of freedoms of the Dexterous Hexrotor and linear independence between them.

5.2 PID Controller Test

Since we need a functional attitude controller help the Dexterous Hexrotor maintaining stable during flight, so we use the PID controller test to decide its PID parameters while we are building the Dexterous Hexrotor.

There are three double-loop PID controllers controlling roll, pitch, yaw and they share the same configuration only with different PID parameters. So we show pitch PID

controller test as an example. First we bolted the Dexterous Hexrotor on a free rolling stick and keep the desired angle as zero. Then we disturbed the Dexterous Hexrotor's attitude by tilting it an angle by hand and release it.

Fig.5.3 PID controller test set up

As soon as we release it, the Dexterous Hexrotor would come back to horizontal under the control of pitch PID controller. But the response may not be good at first time. So we change the PID parameters to make its response better. After the adjustments, response of the Dexterous Hexrotor is plotted in Fig. 5.4. Data of its attitude, the pitch angle, is recorded every 10ms.

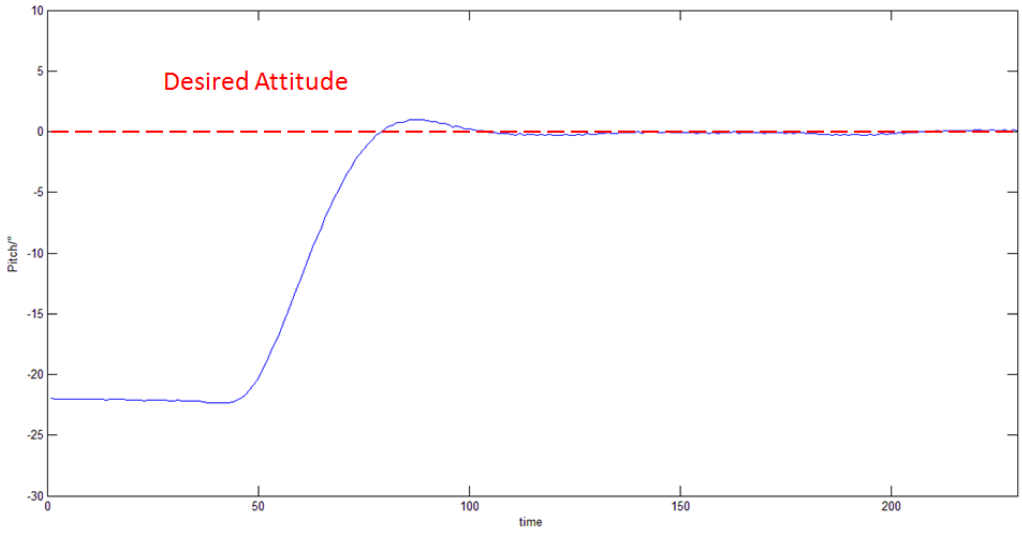


Fig. 5.4 Pitch PID controller response.

As we can see from the plot, the Dexterous Hexrotor tilts at -22° at the beginning. When we release it, the hexrotor comes back to 0° with an overshoot of 0.8° within half second. Then it kept stable around 0° . This would conform a good response of the attitude controller.

5.3 Exert Forces Test

To test the response time of the Dexterous Hexrotor corresponding to external forces, a staged peg-in-hole task is presented with the Dexterous Hexrotor and a normal quadrotor. A frame diagram as Fig. 5.5 shows the experiment's setup. The peg was held rigidly by the UAV. With the peg trapped half-way in a hole which is attached to the 6 DoFs ATI force sensor, the UAV starts to take off and increase throttle until it can carry its own weight. Then we command it to exert a horizontal force perpendicular to the axis of the peg.

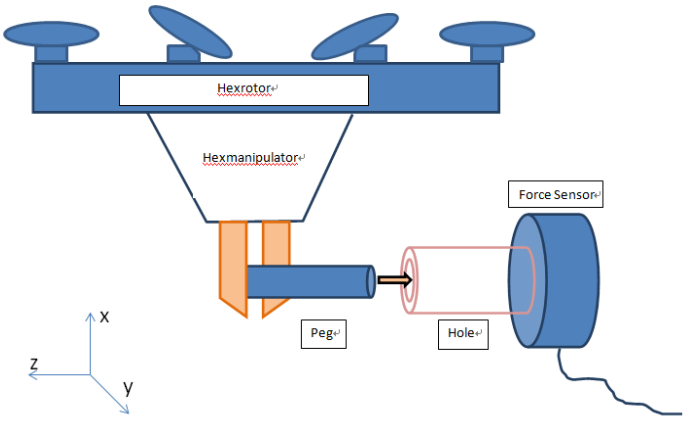


Fig. 5.5 Peg-in-hole setup diagram, with force sensor's coordinate system indicated.

Fig.5.6 Peg-in-hole setup with the Dexterous Hexrotor.

The result is shown in Fig. 5.7 and Fig. 5.8. Changes in force sensor measurements of F_y are direct measurements of the force which the Dexterous Hexrotor applied on the hole. A positive and a negative pulse are detected around 600ms and 1600ms. At the meantime, attitude of the Dexterous Hexrotor was recorded and plot in the same figure.

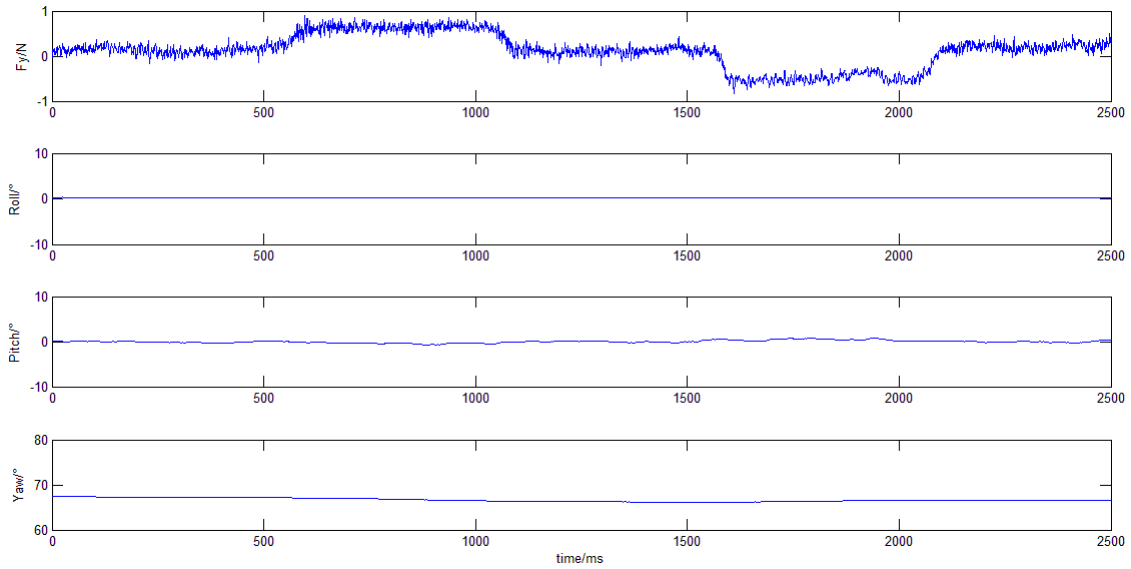


Fig.5.7 The Dexterous Hexrotor measurements of F_y and roll, pitch, yaw.

When the Dexterous Hexrotor is exerting a force in Y axis, its attitude, especially pitch angle, does not change more than 1° . This proves the Dexterous Hexrotor is exerting a force in Y axis without pitching.

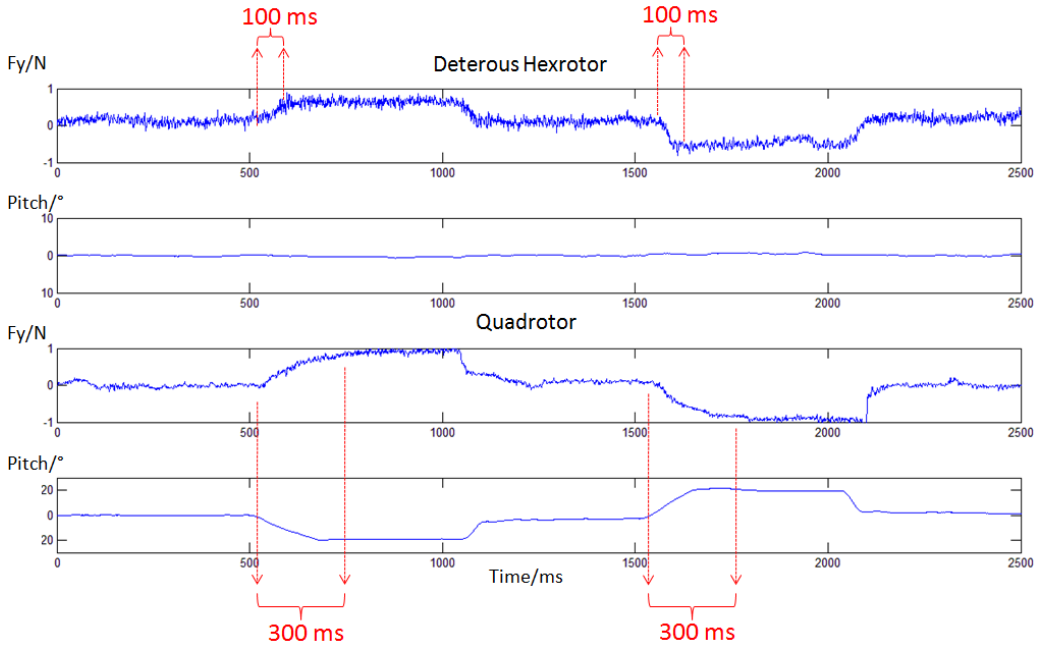


Fig. 5.8 The Dexterous Hexrotor and quadrotor's measurements of F_y and pitch.

First two plots in Fig. 5.8 shows that Dexterous Hexrotor is exerting a force in Y axis without pitching. Last two plots shows when the quadrotor is trying to apply the same force in Y axis, it pitches about -20° and 20°. This is because a quadrotor can only generate a horizontal force by tilting an angle and our hexrotor can do it by simply changing the rotational velocity of the rotors without tilting.

Therefore, when we need a horizontal force, it is obvious that the lag varying speeds of propellers of the hexrotor will be much smaller than that caused by tilting the entire body of a quadrotor.

5.4 Flight Test

Flight test indoor and outdoor has been done. As the result, we can easily fly the Dexterous Hexrotor with the help of attitude controller.

Fig. 5.9 The Dexterous Hexrotor flying indoor.

Fig. 5.10 The Dexterous Hexrotor flying over the University of Denver campus

Translational flight has been demonstrated. In this flight, the Dexterous Hexrotor was moving horizontally without pitching or rolling any angle. Translational acceleration was achieved by F_x and F_y forces produced.

Chapter Six: Conclusion

This thesis introduces a truly holonomic aerial rotorcraft that provides force closure for controlled interaction with external structures. The key contribution of this thesis is to derive the mapping from actuator space to force/torque space based on decomposed net force/torque and develop a metric for the optimization of Dexterous UAV's performance for manipulation-based tasks. A flight-capable prototype with translational force control has been built and tested.

Table VI. Difference between the Dexterous Hexrotor and typical quadrotors

	Quadrotor	Dexterous Hexrotor
Number of Actuators	4	6
Cant Angle of Each Actuator	0°	0° - 90°
DoFs in Force/Torque Space	4	6
Controllable Dofs	4	6
Ability to Achieve Force Closure	No	Yes
Mobility in 3D space	Nonholonomic	Holonomic
Response of Exert Horizontal Force	Slow	Fast

It should be noted that we claim our Dexterous Hexrotor can ``instantaneously resist arbitrary forces." This is not strictly true as the Dexterous Hexrotor can only change the

torque of its motors instantaneously. The required change in the thrust magnitude is not dependent on torque, for a propeller, but on speed. Therefore, the independent thrust magnitudes and the resulting net force/torque experience a lag due to the inertia of the thrusters. The lag due to the rotor inertia is much smaller than that due to pitching the entire vehicle, as for a conventional quadrotor, and we believe is smaller than the pitching of the variable cant angle concept of the Tumbleweed (which also has to overcome the gyroscopic action of the rotors).

To achieve force closure, we cant all thrusters of the Dexterous Hexrotor at an angle and make all thrust vectors nonparallel and vertical. This would definitely influence the power efficiency of the UAV. When a typical quadrotor hovers, all its power is used to combat gravity. But for the Dexterous Hexrotor, some of its power will be used to cancel out each rotor's thrust. This is the main drawback of nonparallel design.

Having functionality proved and force closure achieved, a broad range of research is opened up. Optimizing for tasks from manipulation to surveillance can be tested by varying the cant angle. To be an effective aerial mobile manipulation platform, some sort of gripper can be added. A design that allows for access the edge of the hexrotor, how to counter balance the gripper and its load must be studied. These force tests were worked out low speeds, the dynamic forces should be worked out. A controller has been created to stabilize the hexrotor for human controlled flight and allow for autonomous operation and feedback control over Fx and Fy.

References

[1] Draganfly Innovations Inc. Draganflyer X6, URLwww.draganfly.com/uav-helicopter/draganflyer-x6.

[2] UAS Technologies Sweden AB. Innovative UAV Aircraft & Aerial Video Systems. URL http://uastech.com..

[3] Aerobot. Mikrokopter. aerobot.com.au/hexacopter/mikrokopter-hexakopter.html.

[4] Multicopter Wiki. URL www.multicopter.org.

[5] ArduCopter . URLhttp://code.google.com/p/arducopter/wiki/ArduCopter.

[6] Steven L. Waslander Claire J. Tomlin Gabriel M. Hofmann, Haomiao Huang. Quadrotor helicopter flight dynamics and control: Theory and experiment. In AIAA Guidance, Navigation and Control Conference and Exhibit, aug 2007. doi: 10.1109/ROBOT.2002.1013341.

[7] E. Nice. Design of a four rotor hovering vehicle. Master's thesis, Cornell University, 2004.

[8] E. Altug, J.P. Ostrowski, and R. Mahony. Control of a quadrotor helicopter using visual feedback. In Proceedings, IEEE International Conference on Robotics and Automation, 2002.

[9] P Pounds, Mahony R., and Corke P. Modelling and control of a quad-rotor robot. In Australasian Conference on Robotics and Automation 2006, 2006.

[10] P Pounds, Mahony R., and Gresham J. Towards Dynamically-Favourable Quad-Rotor Aerial Robots, 2010.

[11] Rodney Brooks. A robust layered control system for a mobile robot. IEEE Journal of Robotics and Automation, 2(1):14–23, 1986.

[12] H. Durrant-Whyte, D. Rye, and E. Nebot. Localisation of automatic guided vehicles. In Robotics Research: The 7th International Symposium (ISRR'95), pages 613–625. Springer Verlag, 1996.

[13] P. Akella, M. Peshkin, E. Colgate, W. Wannasuphoprasit, N. Nagesh, J. Wells, S. Holland, T. Pearson, and B. Peacock. Cobots for the automobile assembly line. In Proc. of the IEEE Intl. Conf. on Robotics and Automation, 1999.

[14] K. Hagan, M. Hillman, S. Hagan, and J. Jepson. The design of a wheelchair mounted robot. In Computers in the Service of Mankind: Helping the Disabled (Digest No.: 1997/117), IEE Colloquium on, pages 6/1 –6/6, March 1997.

[15] Guangming Xiong, Jianwei Gong, Junyao Gao, Tao Zhao, Dongxue Liu, and Xijun Chen. Optimum design and simulation of a mobile robot with standing-up devices to assist elderly people and paraplegic patients. In Robotics and Biomimetics, 2006. ROBIO '06. IEEE International Conference on, pages 807 –812, 2006.

[16] Jonathan Fink, Nathan Michael, Soonkyum Kim, and Vijay Kumar. Planning and control for cooperative manipulation and transportation with aerial robots. Volume 30, pages 324–334, March 2011.

[17] Robert Holmberg and Oussama Khatib. Development and control of a holonomic mobile robot for mobile manipulation tasks. International Journal of Robotics Research, 19:1066–1074, Nov 2000.

[18] Yuandong Yang and Oliver Brock. Elastic roadmaps: Globally task-consistent motion for autonomous mobile manipulation in dynamic environments. In Proceedings of Robotics: Science and Systems, August 2006.

[19] Morgan Quigley, Siddharth Batra, Stephen Gould, Ellen Klingbeil, Quoc Le, Ashley Wellman, and Andrew Y. Ng. High-accuracy 3d sensing for mobile manipulation: Improving object detection and door opening. In Proceedings, IEEE International Conference on Robotics and Automation, volume 1, pages 2816 – 2822, May 2009.

[20] Paul Hendricks. Aerial manipulation of voles by common ravens. In Northwestern Naturalist, volume 89, pages 190–192, Winter 2008.

[21] D. Mellinger, M. Shomin, N. Michael, and V. Kumar. Cooperative Grasping and Transport using Multiple Quadrotors. In Proceedings of the International Symposium on Distributed Autonomous Robotic Systems, Nov 2010.

[22] Matthew T. Mason and J.K. Salisbury. Robot Hands and the Mechanics of Manipulation. MIT Press, January 1985.

[23] F. Reuleaux. The Kinematics of Machinery. Macmillan Publishers, 1876, republished by Dover, NY, 1963.

[24] Elon Rimon and Joel Burdick. On force and form closure. In Proceedings, IEEE International Conference on Robotics and Automation, volume 1, pages 1795–1800, April 1996.

[25] V.-D. Nguyen. Constructing force-closure grasps. International Journal of Robotics Research, 7:3–16, June 1988.

[26]Antonio Bicchi. On the closure properties of robotic grasping. International Journal of Robotics Research, 14:319–334, August 1995.

[27]Cong Bang Pham, Song Huat Yeo, Guilin Yang, Mustafa Shabbir Kurbanhusen, and I-Ming Chen. Forceclosure workspace analysis of cable-driven parallel mechanisms. Mechanism and Machine Theory, 41:53–69, 2006.

[28] J Nathan Michael, Jonathan Fink, and Vijay Kumar. Cooperative manipulation and transportation with aerial robots. Autonomous Robots, 30:73–86, 2011. ISSN 0929-5593.

[29] Nathan Michael, Soonkyum Kim, Jonathan Fink, and Vijay Kumar. Kinematics and statics of cooperative multi-robot aerial manipulation with cables. In ASME International Design Engineering Technical Conference and Computers and Information in Engineering Conference, 2009.

[30] D. Mellinger, M. Shomin, N. Michael, and V. Kumar. Cooperative Grasping and Transport using Multiple Quadrotors. In Proceedings of the International Symposium on Distributed Autonomous Robotic Systems, Nov 2010.

[31] D. Mellinger, N. Michael, and V. Kumar. Trajectory Generation and Control for Precise Aggressive Maneuvers with Quadrotors. In Proceedings of the International Symposium on Experimental Robotics, Dec 2010.

[32] Markus Ryll, Heinrich H. B ¨ulthoff, and Paolo Robuffo Giordano. Modeling and Control of a Quadrotor UAV with Tilting Propellers. In IEEE International Conference on Robotics and Automation, 2012.

[33] Langkamp, D. ·Roberts, G. ·Scillitoe, A. ·Lunnon, I. ·Llopis-Pascual, A. ·Zamecnik, J. ·Proctor, S. ·Rodriguez-Frias, M. ·Turner, M. ·Lanzon, A. ·Crowther, W. An engineering development of a novel hexrotor vehicle for 3D applications. In International Micro Air Vehicle conference and competitions, 2011.

[34] A. M. Dollar D. R. Bersak P. E. Pounds. Grasping From the Air: Hovering Capture and Load Stability. In IEEE International Conference on Robotics and Automation, 2011.

[35] R. M. Voyles, S. Povilus, R. Mangharam and K. Li, RecoNode: A Reconfigurable Node for Heterogeneous Multi-Robot Search and Rescue. In IEEE International Workshop on Safety, Security, and Rescue Robotics, 2010.

[36] S. Stramigioli A. Keemink, M. Fumagalli and R. Carloni. Mechanical design of a manipulation system for nmanned aerial vehicles. In Proceedings, IEEE International Conference on Robotics and Automation, 2012.

[37] C. Korpela and P.Y. Oh. Designing a mobile manipulator using an unmanned aerial vehicle. In IEEE International Conference on Technologies for Practical Robot Applications, April 2011.

[38] A. Albers, S. Trautmann, T. Howard, T. Nguyen, M. Frietsch, and C. Sauter. Semi-autonomous flying robot for physical interaction with environment. In Proceedings, IEEE International Conference on Robotics Automation and Mechatronics, pages 441–446, June 2010.

[39] M. Bernard A. Ollero I. Maza, K. Kondak. Mechanical design of a manipulation system for unmanned aerial vehicles. In Journal of Intelligent and Robotic Systems, volume 57, pages 417–449, Jan 2010.

[40] T. Yoshikawa. Manipulability of robotic mechanisms. In Journal of Robotics Research, volume 4, 1985.