

2023（令和5）年度 修士学位論文

## 4 発テイルトウイング機の制御システム開発

### Control system for Quad-Tilt Wing aircraft

2024年2月13日

高知工科大学大学院 工学研究科基盤工学専攻  
航空宇宙工学コース

1265034 森本 拓馬

指導教員 岡 宏一教授

# 目次

1.1	研究背景	- 1 -
1.2	期待されている活用事例	- 1 -
1.3	斜面着陸	- 2 -
1.4	研究目的	- 2 -
2.	機体の改良（ハードウェア）	- 3 -
2.1	ブラシレスモータ用バッテリーの変更と増設	- 3 -
2.2	各基盤の再作成	- 4 -
3.	機体の改良（ソフトウェア）	- 5 -
3.1	ROS 2 システム	- 5 -
3.1.1	GCS ノードの削除	- 5 -
3.1.2	パラメータサーバーの導入	- 5 -
3.2	姿勢推定システム	- 6 -
3.3	センサーモジュール, アクチュエータモジュールの再開発	- 7 -
4.	斜面推定システムの開発	- 8 -
4.1	SLAM を用いた外界情報の取得	- 8 -
4.2	深度情報を用いた斜面の傾斜推定	- 8 -
4.3	斜面に対する機体の回転について	- 9 -
4.3.1	前方 2 点の深度の差分から機体を回転させる手法	- 10 -
4.3.2	3 つの深度情報を用いた回転角を推定する手法	- 10 -
4.4	各手法のシミュレーションを用いた比較	- 13 -
4.4.1	各座標での深度と斜面のモデル化	- 13 -
4.5	シミュレーションによる 2 手法の比較	- 14 -
4.6	深度カメラを用いた実装	- 15 -
4.6.1	使用した深度カメラとソフトウェア	- 15 -
4.6.2	深度情報取得位置	- 16 -
4.6.3	深度情報の取得	- 16 -
4.6.4	深度情報取得位置の関係	- 17 -
4.7	実機を用いた斜面傾斜推定及び回転角推定実験	- 18 -
4.7.1	実験方法	- 18 -
4.7.2	実験結果	- 18 -
4.7.3	考察	- 19 -
5.	小型モデルを用いた最適制御検証	- 20 -
5.1	目的	- 20 -
5.2	小型モデル作成	- 20 -
5.2.1	機体設計	- 20 -
5.3	使用機材	- 21 -
5.3.1	フレーム	- 21 -
5.3.2	プロペラ駆動用モータ及び ESC	- 21 -
5.3.3	バッテリー	- 22 -
5.3.4	プロペラ	- 22 -
5.3.5	フライトコントローラ	- 22 -
5.3.6	姿勢推定用 IMU	- 23 -
5.3.7	完成機体	- 23 -
5.4	運動モデル	- 23 -
5.4.1	座標系	- 24 -
5.4.2	非線形モデル	- 24 -
5.4.3	線形モデル	- 24 -
5.5	システム同定	- 26 -
5.5.1	機体重量と慣性モーメントについて	- 26 -

5.5.2	推力及びプロペラの回転に伴う反トルクについて .....	- 27 -
5.6	制御器設計 .....	- 28 -
5.6.1	LQR コントローラ設計 .....	- 28 -
5.6.2	シミュレーション .....	- 28 -
5.7	小型モデルを用いた 1 自由度実験 .....	- 30 -
5.7.1	1 実験方法 .....	- 30 -
5.7.2	実験結果 .....	- 31 -
5.8	6 自由度飛行試験 .....	- 32 -
5.8.1	実験方法 .....	- 32 -
5.8.2	実験結果 .....	- 32 -
5.8.3	考察 .....	- 33 -
6.	実機の制御系設計 .....	- 35 -
6.1	運動モデルの再検討 .....	- 35 -
6.2	運動モデル .....	- 35 -
6.2.1	座標系 .....	- 35 -
6.2.2	並進運動モデル .....	- 37 -
6.2.3	回転運動モデル .....	- 37 -
6.2.4	プロペラ後流による揚力のモデリング .....	- 39 -
6.2.5	機体のモデリング .....	- 41 -
6.2.6	ティルト角の微小な揺動 .....	- 41 -
6.3	シミュレーション .....	- 42 -
6.3.1	制御系 .....	- 42 -
6.3.2	シミュレーション結果 .....	- 43 -
6.4	飛行試験 .....	- 43 -
6.4.1	実験 1 .....	- 43 -
6.4.2	実験 2 .....	- 44 -
6.4.3	飛行試験の考察 .....	- 45 -
7.	結論 .....	- 47 -
	謝辞 .....	- 48 -
	参考文献 .....	- 48 -

## 1. 序論

### 1.1 研究背景

本研究の対象機体である4発ティルトウイング機(Quad Tilt Wing aircraft : QTW)はVTOL(Vertical Take Off and Landing)機の一つであり、以下の図 1.1, 1.2 に示すように4枚の可動主翼を持った機体である。この機体の特徴は翼角度を変化させることによって機体の飛行特性を変えることができる点である。図 1.1 は翼を地面と垂直に固定したものであり、この状態をマルチコプターモードと呼称する。この状態ではマルチコプターの様に4枚のプロペラを用いて機体の浮上、姿勢制御を行うことで垂直離着陸や空中でのホバリングなどが可能となる。

一方、図 1.2 は翼を地面と平行に固定したものであり、この状態を飛行機モードと呼称する。この状態では主に主翼から揚力を発生させ浮上し、プロペラを回転させることで推進力を得て飛行する。そのため、固定翼機のように高速度での巡行や、長距離の巡行が可能となる。



図 1.1 マルチコプターモード



図.1.2 飛行機モード

### 1.2 期待されている活用事例

本機で期待されている活用事例として災害時の監視が挙げられる。以下の図 1.3 にはその概要を示す。地震や津波などの災害が発生した場合、物資供給や災害の監視のために現場から離れた基地局から素早く現場へ向かう必要がある。そのため、基地局から現場までは飛行機モードで移動する。現場到着後はマルチコプターモードに遷移し、ホバリングを行いながら災害の監視を行う。また、災害地付近の滑走路等が地震や津波などで使用不可となっている場合にも、付近に機体大のスペースがあれば、そこに着陸することができる。

機体の着陸について、災害現場では瓦礫や家屋の倒壊によりスペースはあるものの、平坦ではないという状況が多く見られる。そこで次節では QTW が従来の回転翼機と比較して、斜面への着陸といった点について優位であることを述べる。

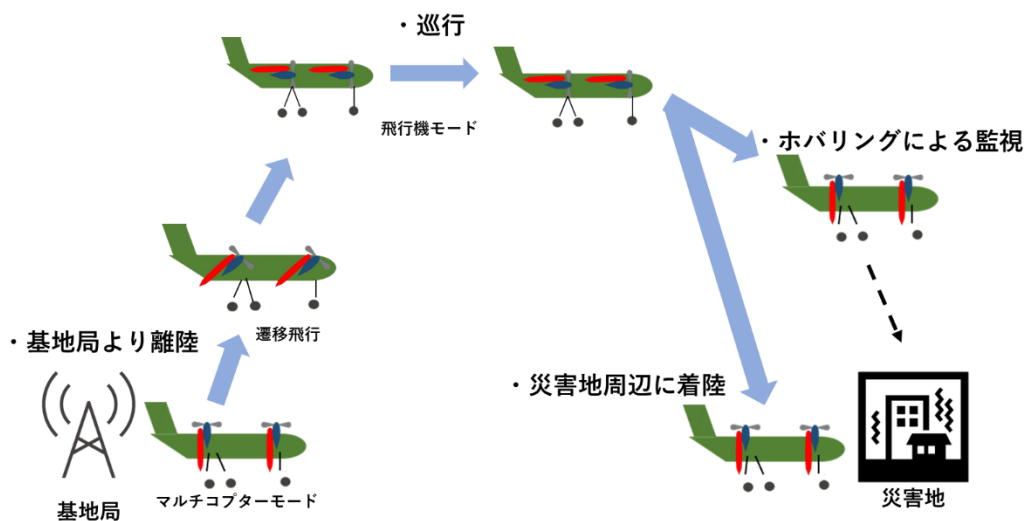


図 1.3 災害時の QTW 活用方法と飛行スケジュール

### 1.3 斜面着陸

以下の図 1.4 には従来回転翼機の代表としてヘリコプタが斜面着陸を行う様子を示している。斜面に着陸する場合、機体を斜面と平行に傾ける必要があるが、ヘリコプタは常に推力は機体と垂直に生じているため、機体を地面と平行に傾けると斜面から離れる方向に推力が向いてしまい斜面への着陸が困難である。

一方、図 1.5 には QTW が斜面に着陸の様子を示している。QTW はヘリコプタと異なりティルト角を変化させることで推力の向きを変化させることができるため、図に示すように機体は斜面と平行にし、推力は重力と釣り合う方向に向けることで斜面着陸が容易になると考えられている。

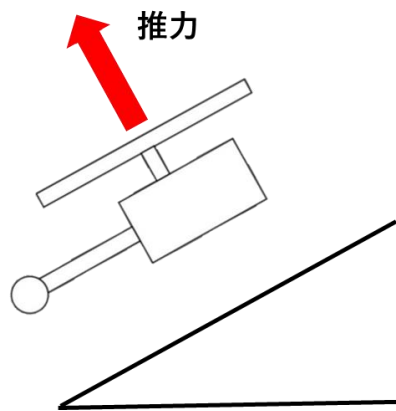


図 1.4 ヘリコプタの斜面着陸の様子

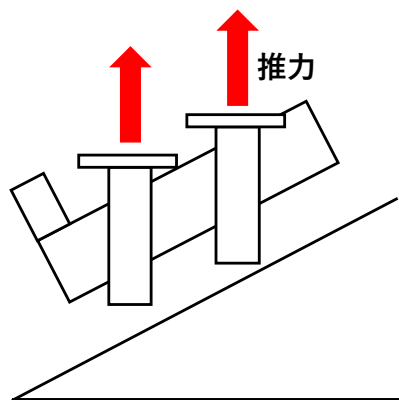


図 1.5 QTW の斜面着陸の様子

### 1.4 研究目的

QTW の斜面への着陸及びホバリング中のティルト角の遷移については丹羽<sup>(1)</sup>らによって最適制御を用いた機体制御によって可能であることがシミュレーション上で示されている。本研究では機体の斜面着陸について、当研究室で作成された機体<sup>(2)</sup>を用いて実証する事を研究の目的としている。

## 2. 機体の改良（ハードウェア）

本章，次章では先行研究<sup>(1)</sup>にて行われた飛行試験（以下，前年度試験）を踏まえて，いくつか改良すべき点が見つかったため機体の改良について示す．

### 2.1 ブラシレスモータ用バッテリーの変更と増設

以下の表 2.1 に変更前のバッテリーと変更後のバッテリー緒言について示す．前年度試験ではプロペラ駆動に使われたブラシレスモータの出力が 70%程度で機体が持ち上がっていることが確認できた．しかし，本機体に用いられている ESC 側で安全装置が働き，80%以上の推力は制限されていることから，推力の余剰が不足していると判断しバッテリーを変更した．

表 2.1 変更前と変更後のバッテリーの比較

	変更前	変更後
バッテリータイプ	Li-Po	Li-Po
セル数 [Cell]	3	4
バッテリー容量 [mAh]	4200	6500
C レート [C]	35	80

以下の図 2.1 には変更前(3Cell)と変更後(4Cell)の推力の比較を示す．変更前のバッテリーでは一基当たり必要な推力である約 10N を得るためには約 65%の PWM 入力が必要であったが，変更後は得られる得られる出力が向上しており，約 50%の PWM 入力で約 10N を得る事ができていることがわかる．

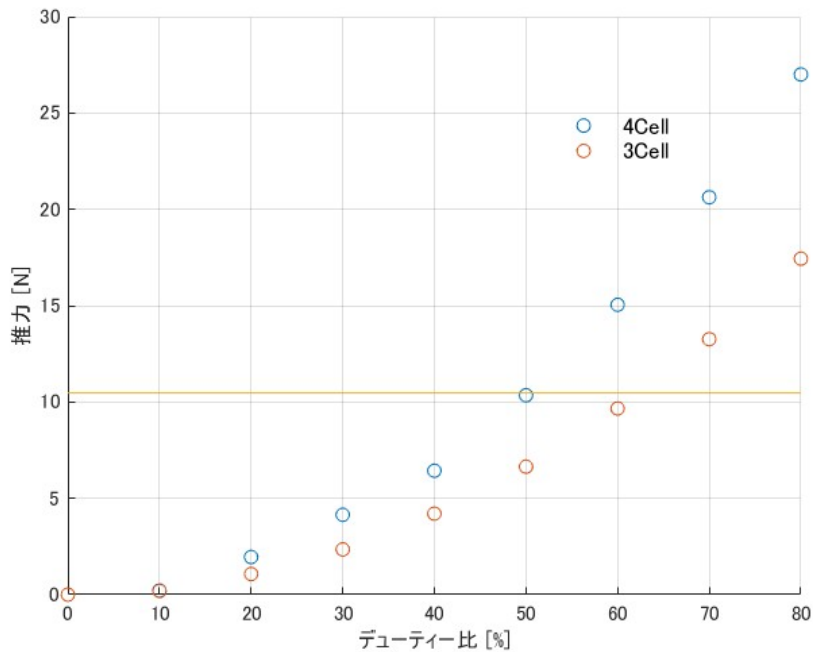


図 2.1 変更前バッテリー(3cell)と変更後バッテリー(4cell)での推力比較

また，前年度試験ではフライトコントローラとして使用していた，RaspberryPi4B とティルト，フラップ，ラダーに用いられているサーボモータを全て共通化していた．しかし，この設計では前年度試験時，バッテリー消費が非常に早く，15分程度しか連続して実験を行う事が出来なかった．加えて，フライトコントローラとして用いられている RaspberryPi4B は電力が低下すると，処理速度の低下がみられることから

フライトコントローラ用のバッテリーとサーボモータ用バッテリーの分離，増設した．以下の表 2.2 には各用途ごとの使用バッテリーを示す．

表 2.2 使用したバッテリー

	フライトコントローラ	サーボモータ用
バッテリータイプ	Li-Po	Li-Po
セル数 [Cell]	2	2
バッテリー容量 [mAh]	1000	1000
C レート [C]	30	50

## 2.2 各基盤の再作成

前節の通りバッテリーの増設を行ったことから，各基盤の再作成を行う必要がある．基盤の再作成に当たり，前年度試験時に作成された基盤はプリント基板で作成されており，新たなセンサ等の増設のハードルが高かった．このことから，ユニバーサル基盤で再作成し，新規のセンサ増設しやすくした．以下の図 2.2 にはその再作成した基盤の外観を示す．

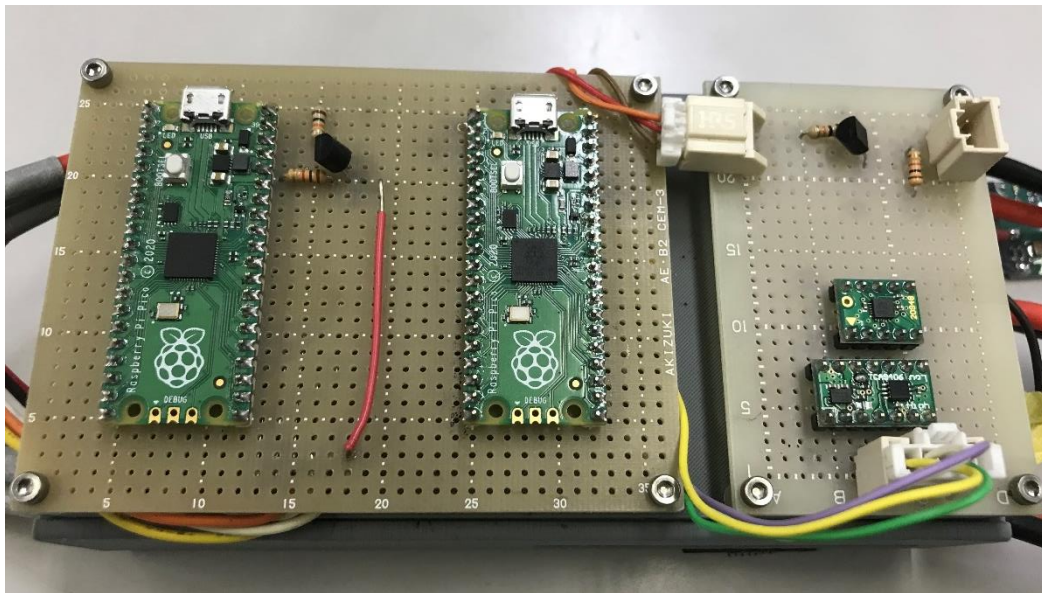


図 2.2 再作成基盤の外観

また，次節でも述べるが，前年度実験ではセンサ用及び各モータ駆動に Tiny2040 を使用している．しかし，このマイコンは本体サイズが非常に小さい事がメリットであるが，サイズが小さいことでピン数が限られており，センサ及び他のモジュールの増設に制限があった．そこで，これらのマイコンを Raspberry Pi Pico に変更することで増設可能なモジュール数を増やしている．以下の図 2.3,2.4 にはそれらの外観，表 2.3 にはそれらの比較を示している．





図 2.3 Tiny2040

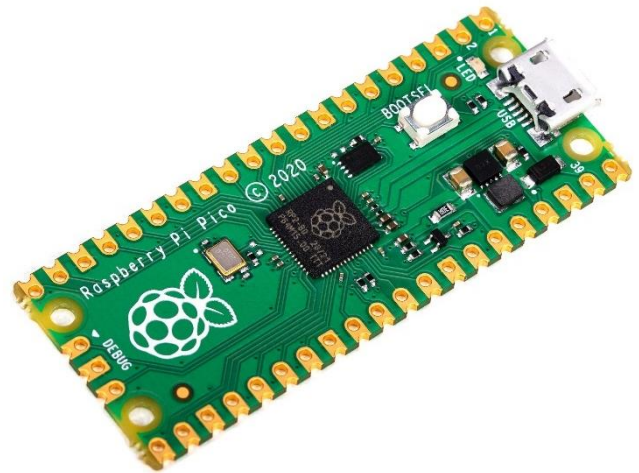


図 2.4 Raspberry Pi Pico

表 2.3 マイコンの比較

	Tiny2040	Raspberry Pi Pico
寸法[mm]	22.9 × 18.2	51 × 21
動作電圧[V]	3-5.5	1.8-5.5
搭載チップ	RP2040	RP2040
コア	ARM Cortex M0+ × 2	ARM Cortex M0+ × 2
ピン数	12(SPI:1 I2C:2 UART:2 ADC: 4)	26(SPI:2 I2C:2 UART:2 ADC:3)

### 3. 機体の改良（ソフトウェア）

#### 3.1 ROS 2 システム

前年度試験時に構成された ROS 2 を用いた分散処理という設計思想，および各ノードの役割などを踏襲しつつ，プログラムの簡素化を行い，より開発，実験が進めやすいように改良した．以下の図には今回使用したシステムの概要を示す．

##### 3.1.1 GCS ノードの削除

システムの簡素化のため，前年度実験で不使用だった地上局ノードは削除した．地上局ノードに割り当てられていた，機体のモード変更はプロポからのスティック操作で変更できるように設定した．

##### 3.1.2 パラメータサーバーの導入

前年度実験ではコントローラのゲインを変更する際は，ゲインを手動で入力に再度ビルドしていた．しかし，この方法ではゲイン調整をするためにビルドする時間がかかってしまうため，Control ノードにパラメータサーバーを導入し，プログラム実行中にゲインを変更できるように設定した．



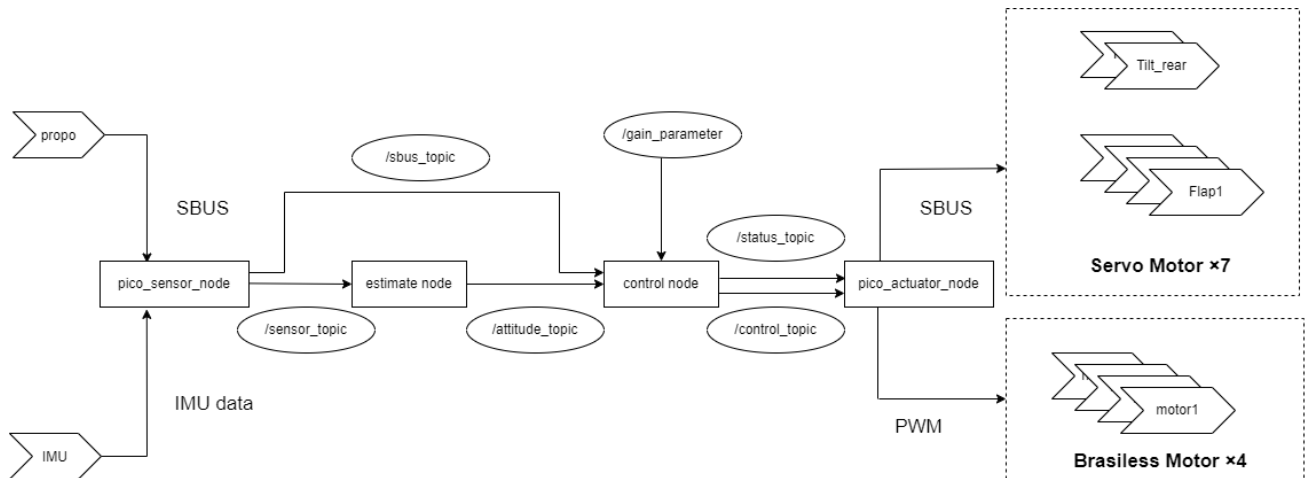


図 3.1 ROS 2 システム概要

### 3.2 姿勢推定システム

前年度試験では相補フィルタを用いて姿勢推定を行っていたが、本研究では Madgwick filter<sup>(3)</sup>を用いて姿勢推定を行った。この姿勢推定アルゴリズムはクォータニオンがベースの姿勢推定アルゴリズムであり、カルマンフィルタと同程度の精度を持ち、かつカルマンフィルタよりも高速で処理することがメリットとして挙げられる。ROS 2 ではこのアルゴリズムのプラグイン<sup>(4)</sup>が公開されており、本研究ではこのプラグインを使用し、姿勢推定を行う。

この姿勢推定アルゴリズムは相補フィルタと同様に、角速度の積分による推定値と加速度による推定値の混合比率をゲインで設定することができる。以下の図 3.2 はモーションキャプチャで計測された姿勢角を真値とし、ゲインを 0.2, 0.6 の 2 パターンで推定し比較したものである。

ゲインを 0.2 に設定した場合は推定値の高周波ノイズが抑えられていることがわかるが、真値と比較して位相が大きく遅れていることがわかる。一方ゲインを 0.6 に設定した場合は推定値にノイズが乗っており、真値より大きく推定値が算出されているが、真値に対して大きな位相遅れなく追従していることがわかる。本研究対象である機体はマルチコプターのように飛行することがあることから、位相遅れ無く推定することに重きを置き、試行錯誤の結果、最終的にはゲインを 0.8 に設定した。

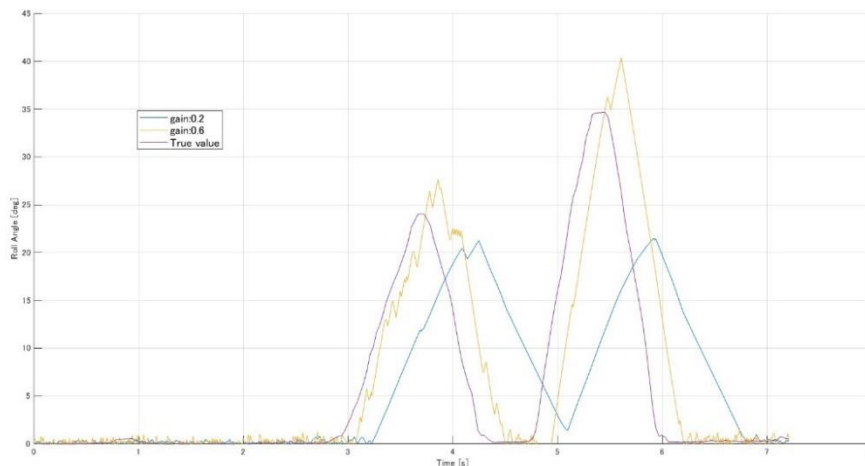


図 3.2 ゲインによる推定結果の比較

### 3.3 センサーモジュール、アクチュエータモジュールの再開発

前章で述べたように、基盤の再設計に伴いセンサモジュール、アクチュエータモジュールとして用いられていたマイコンを変更した。この変更に伴い、それぞれ内部のプログラムも軽微な変更を行った。参考文献より SBUS を受信及び送信するためには数 ms の待機時間が必要であることから、前年度試験では FreeRTOS を用いて並列処理を行い、処理の高速化を図っていた。しかし、欠点としてプログラムが非常に複雑になることが挙げられ、開発が非常に困難であった。そこで、マルチコアによる並列処理を用いて SBUS 信号の処理を完全に別にしたことで複雑な処理が不要となり、その結果今後の開発の難易度が大きく下がったと考えている。

## 4. 斜面推定システムの開発

先行研究<sup>(4)</sup>で行われた数値検証では斜面の角度は既知であるものとして検証していた。本章では深度情報を用いた斜面推定について検討し、深度カメラを用いた実機での斜面推定実験の結果まで示す。

### 4.1 SLAM を用いた外界情報の取得

近年、マルチコプター型 UAV や自動運転などで用いられる外界情報取得の手段としては主に、SLAM といった技術が挙げられる。SLAM とは Simultaneous Localization and Mapping の略語であり、自己位置推定と環境地図作成を同時に行うといった技術である。SLAM には大きく分けて、レーザスキャナである LiDAR を用いた LiDAR SLAM, HD カメラを用いた Visual SLAM, 距離画像カメラを用いた Depth SLAM の3手法が存在する。それぞれ、異なるセンサを用いた SLAM 手法であり、以下表 4.1 にそれぞれの特徴を示す。

表 4.1 SLAM 手法の比較

	LiDAR	HD カメラ	距離画像カメラ
データ	2D/3D 点群	2D 画像	深度画像
測距原理	Time-of-Flight 三角法	三角法	Time-of-Flight 三角法
最大測定距離	数十～数百メートル	基線長に依存	数メートル
視野角	水平：～360 度 垂直：数十度	水平：～120 度程度 垂直：数十度	水平：数十度 垂直：数十度
備考	高精度、低密度	低精度、高密度、	中精度、高密度 太陽光下では困難

出典：友納・原(2020)<sup>(5)</sup>を参考に作成

以上のような SLAM 技術に共通して当てはまる特徴としては高精度化を実現しようとするると計算処理が非常に多くなる事である。本研究ではフライトコントローラに RaspberryPi を用いており飛行中にいずれかの SLAM 技術を高精度でリアルタイムに処理することは非常に困難であると考えられる。そこで、次節では飛行中の機体からの深度情報 3 点のみと、それらの幾何学的関係から斜面推定を行うシステムについて提案する。

### 4.2 深度情報を用いた斜面の傾斜推定

本節では以下の図 4.1 に示すような斜面の傾斜角 $\alpha$ の推定原理について示す。示された斜面において斜面座標系を斜面上部方向を $X^s$ 軸、斜面右手方向を $Y^s$ 軸、斜面に垂直な方向を $Z^s$ 軸として定義する。また、簡単のために本研究では斜面の $X^s$ 軸方向の傾きは常に一定で $Y^s$ 軸方向の傾きは 0 である単純な斜面について検討する。また、以下の図 4.2 に示すような機体前方を底辺とした二等辺三角形の頂点の位置で深度情報を取得する。図中の $H_A, H_B, H_C$ は各点での深度を表している。

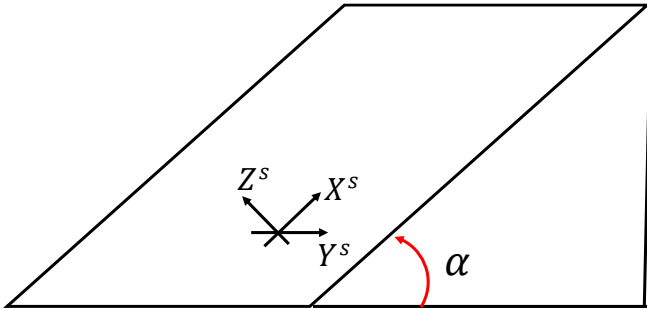


図 4.1 推定対象の斜面

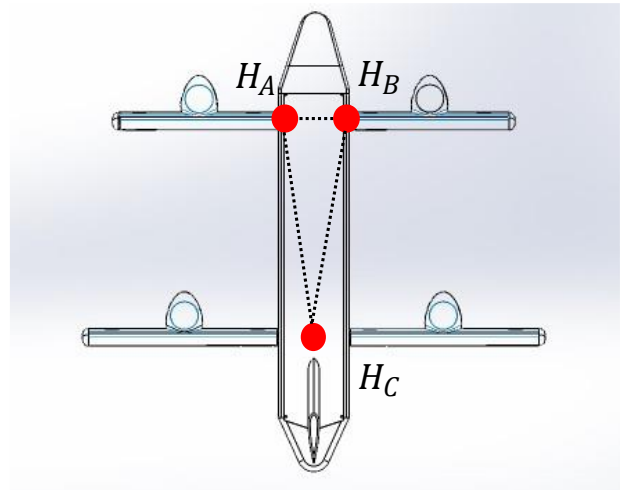


図 4.2 取得する深度情報の位置関係

以上のような斜面の傾斜 $\alpha$ は図 4.3 の様に前後の深度情報とそれらの距離から以下の式(4.1)を用いてに非常に簡単に求めることができる。

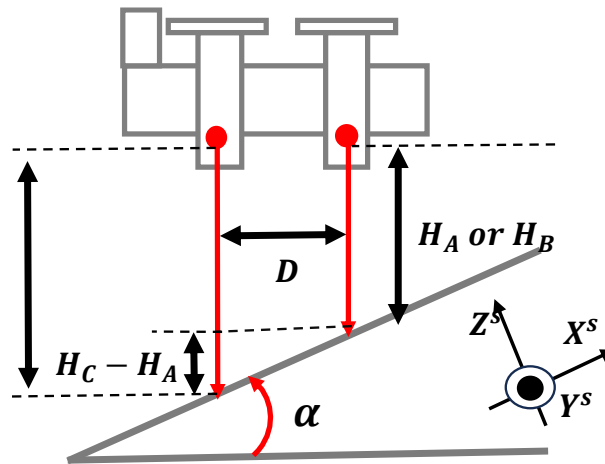


図 4.3 斜面傾斜の推定

$$\alpha = \tan^{-1} \left( \frac{H_C - H_A}{D} \right) \quad (4.1)$$

式中の $D$ は前部深度測定位置から後部深度測定位置までの距離である。また、対象斜面は $Y^s$ 軸方向に傾きが無い事から、左右の深度情報は等しく $H_A = H_B$ であるので右側深度情報 $H_2$ を用いても同様の結果を得る事ができる。

#### 4.3 斜面に対する機体の回転について

前節では前後の深度情報から斜面の傾斜推定を行う方法について検討したが、以下の図 4.4 右図の様に機体が斜面に対して $\beta$ 回転してしまっている場合に以上の傾斜推定を行ってしまうと、赤線上の傾斜を

求めていることになり、正しい傾斜推定とは言えない。そこで、斜面に対して $\beta$ 回転している場合にはその回転角を認識し、 $\beta = 0$ となるようなシステムについて検討する。本研究では前方2つの深度情報の偏差から機体を回転させる手法と、3つの深度情報を用いて回転角 $\beta$ を推定し機体を回転させる2種類の手法について述べる。

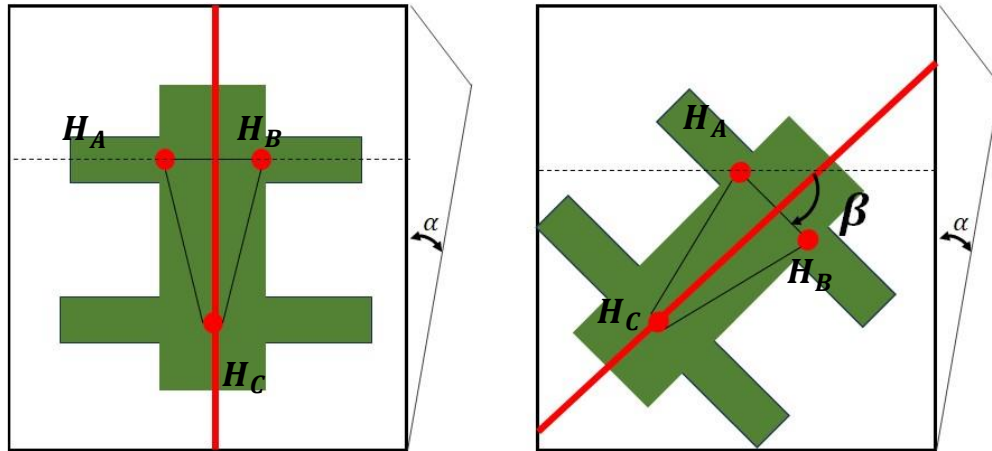


図 4.4 斜面に対する機体の回転角 $\beta = 0$  (左図)と $\beta \neq 0$  (右図)

#### 4.3.1 前方2点の深度の差分から機体を回転させる手法

簡単な斜面では $Y^s$ 軸方向に傾きは無いことから、式(4.2)を満たすことができれば機体の回転角である $\beta$ は0になったと言える。この関係を用いて、以下の図4.5にこの手法のブロック線図を示す。このシステムは深度センサから取得した前方2点の $H_A$ と $H_B$ に偏差があった場合、コントローラを介してヨー軸周りに $\psi$ の目標値を与え回転させるといったものである。本稿では以下この手法を手法1と呼ぶ。

$$H_A - H_B = 0 \quad (4.2)$$

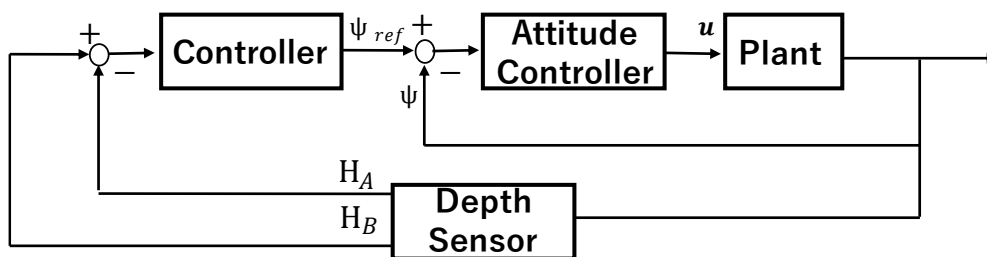


図 4.5 前方2点の偏差を用いた回転角制御(手法1)

ただし、以上の制御方法については $\beta = 180deg$ の場合も式(4.2)を満たすため、注意が必要である。

#### 4.3.2 3つの深度情報を用いた回転角を推定する手法

この手法は回転角 $\beta$ を推定し推定した角度をそのまま姿勢角目標値として与える手法である。前項の手法とは異なりコントローラの数をもつ減らすことができるのがこの手法の利点である。本稿ではこの手法の事を手法2と呼ぶ。

以下の図 4.6 は図 4.4 右図を幾何的にモデル化した物である。赤点の位置がそれぞれ深度情報取得位置であり  $H_A, H_B, H_C$  取得位置をそれぞれ点  $A, B, C$  として表している。また、 $l_{AB}, l_{BC}$  はそれぞれ、点  $A$  と点  $B$  の距離、点  $B$  と点  $C$  の距離、 $\omega$  は  $l_{AB}, l_{BC}$  がなす角である。

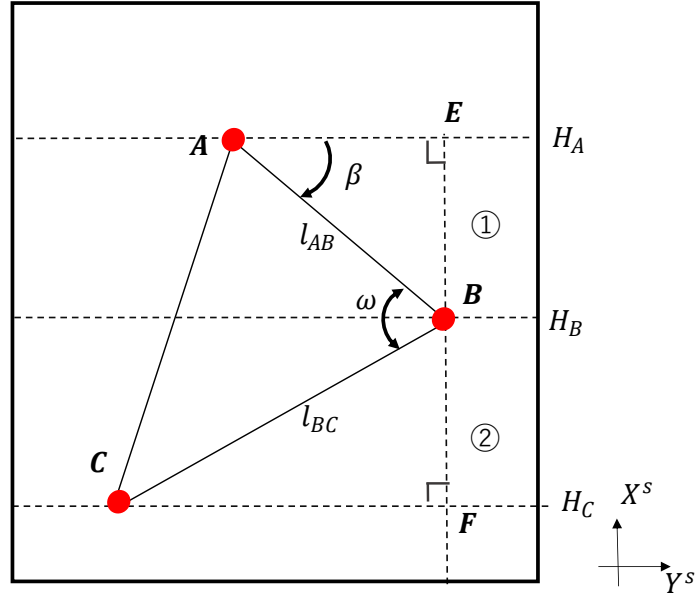


図 4.6 各深度情報取得位置とそれらの幾何的關係

簡単な斜面の仮定から  $X_s$  軸方向に傾きは等しい事から、①の区間の傾き  $\alpha_{EB}$  と②の区間の傾き  $\alpha_{BF}$  の関係は式(4.3)の様に表すことができる。

$$\alpha_{EB} = \alpha_{BF} \quad (4.3)$$

次に①の区間の傾き  $\alpha_{EB}$  について求めていく。簡単な斜面の仮定から  $Y_s$  軸方向に傾きは無いので、点 E での深度は  $H_A$ 、点 B での深度は  $H_B$  である。 $\angle ABE$  は式(4.4)の様に表され、

$$\angle ABE = 90 - \beta \quad (4.4)$$

EB 間の距離  $l_{EB}$  は  $l_{AB}$  を用いて

$$l_{EB} = l_{AB} \cos(\angle ABE) = l_{AB} \cos(90 - \beta) = l_{AB} \sin \beta \quad (4.5)$$

すなわち①区間の傾き  $\alpha_{EB}$  は(4.5)と  $H_1$  と  $H_2$  の差から求めることができるので、

$$\alpha_{EB} = \tan^{-1} \left( \frac{H_B - H_A}{l_{EB}} \right) = \tan^{-1} \left( \frac{H_B - H_A}{l_{AB} \sin \beta} \right) \quad (4.6)$$

と表すことができる。同様に②の区間についての傾き  $\alpha_{BF}$  についても求める。 $\angle CBF$  は式(4.7)の様に表される

$$\angle CBF = 180 - (\omega + \angle ABE) \quad (4.7)$$

従って  $\angle CBF$  は

$$\angle CBF = 180 - (\omega + 90 - \beta) = 90 - \omega + \beta \quad (4.8)$$

BF 間の距離  $l_{BF}$  は  $l_{23}$  を用いて

$$l_{BF} = l_{BC} \cos(\angle CBF) = l_{BC} \cos\{90 - (\omega - \beta)\} = l_{23} \sin(\omega - \beta) \quad (4.9)$$

すなわち②区間の傾き $\alpha_{BF}$ は(4.9)と $H_B$ と $H_C$ の差から求めることができるので、

$$\alpha_{BF} = \tan^{-1}\left(\frac{H_C - H_B}{l_{BF}}\right) = \tan^{-1}\left\{\frac{H_C - H_B}{l_{BC} \sin(\omega - \beta)}\right\} \quad (4.10)$$

(4.3)より

$$\tan^{-1}\left(\frac{H_B - H_A}{l_{AB} \sin \beta}\right) = \tan^{-1}\left\{\frac{H_C - H_B}{l_{BC} \sin(\omega - \beta)}\right\} \quad (4.11)$$

上式の左辺と右辺を比較すると

$$\frac{H_B - H_A}{l_{AB} \sin \beta} = \frac{H_C - H_B}{l_{BC} \sin(\omega - \beta)} \quad (4.12)$$

$$(H_B - H_A)l_{BC} \sin(\omega - \beta) = (H_C - H_B)l_{AB} \sin \beta \quad (4.13)$$

$(H_B - H_A)l_{BC} = A$ ,  $(H_C - H_B)l_{AB} = B$ と置くと

$$A(\sin \omega \cos \beta - \cos \omega \sin \beta) = B \sin \beta \quad (4.14)$$

$\beta$ について整理すると

$$\frac{\sin \beta}{\cos \beta} = \tan \beta = \frac{A \sin \omega}{A \cos \omega + B} \quad (4.15)$$

従って回転角 $\beta$ は

$$\beta = \tan^{-1}\left(\frac{A \sin \omega}{A \cos \omega + B}\right) \quad (4.16)$$

ただし

$$A = (H_B - H_A)l_{BC} \quad (4.17)$$

$$B = (H_C - H_B)l_{AB} \quad (4.18)$$

である。

従って、求められた式(4.16),(4.17),(4.18)を用いて図 4.6 の様に赤点の地点の深度を取得すると斜面に対する回転角 $\beta$ を求めることができる。これらの式を用いた斜面に対する回転角制御のブロック線図を以下の図 4.7 に示す。

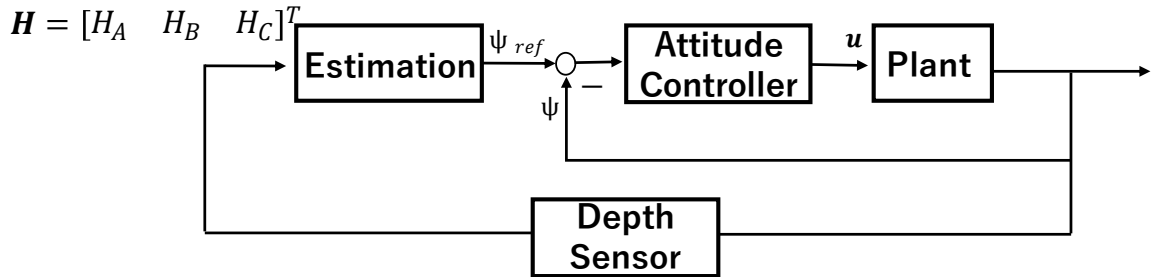


図 4.7 回転角推定を用いた制御(手法 2)



#### 4.4 各手法のシミュレーションを用いた比較

##### 4.4.1 各座標での深度と斜面のモデル化

本節では以上で提案した2手法についてシミュレーションを用いて、性能の比較検討を行う。まずは斜面と機体の深度モデルを構築する。機体の深度モデルを構築するには、機体が $\beta$ 回転した際の深度取得位置の座標を決定する必要がある。以下の図 4.8 は各深度取得位置の関係を示した図である。図 4.8 に示す $X_{BD}Y_{BD}$ 座標は左前方の深度取得位置である点 A を原点と定義した機体固定深度座標であり図 4.9 の様に機体に固定した座標系である。また、図 4.8 の $X'_sY'_s$ 座標は機体斜面に対する回転角 $\beta$ だけ点 A 周りで回転させた座標である。

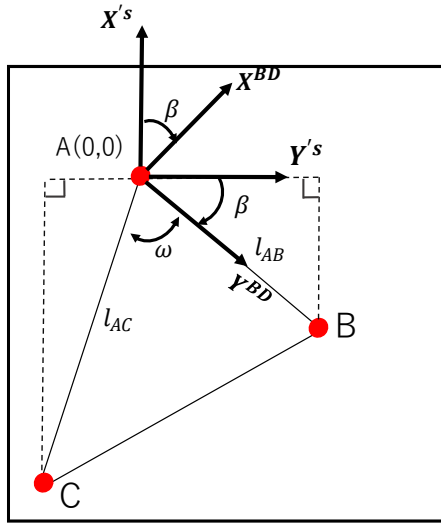


図 4.8 各深度情報取得位置の関係

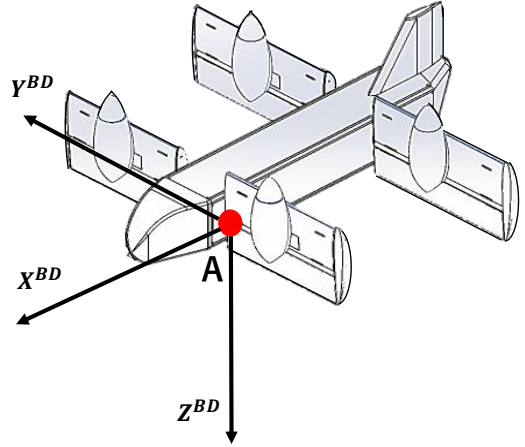


図 4.9 機体固定深度座標系

図 4.8 から B の座標 $B(x_B^{BD}, y_B^{BD})$ は

$$x_B^{BD} = 0 \quad (4.19)$$

$$y_B^{BD} = l_{AB} \quad (4.20)$$

同様に C の座標 $C(x_C^{BD}, y_C^{BD})$ は

$$x_C^{BD} = -l_{AC} \sin \omega \quad (4.21)$$

$$y_C^{BD} = l_{AC} \cos \omega \quad (4.22)$$

で表すことができる。

次にこの座標系を斜面に対する回転角 $\beta$ だけ回転させた $X'^sY'^s$ 座標に座標変換すると、点 B の座標 $B(x_B'^s, y_B'^s)$ と点 C の座標は $C(x_C'^s, y_C'^s)$ は以下の様に表すことができる

$$x_B'^s = -l_{AB} \sin \beta \quad (4.23)$$

$$y_B'^s = l_{AB} \cos \beta \quad (4.24)$$

$$x_C'^s = -l_{AC} \sin(\beta + \omega) \quad (4.25)$$

$$y_C'^s = l_{AC} \cos(\beta + \omega) \quad (4.26)$$

以下の図 4.10 は斜面及び、機体を斜面側面からみた図である。先ほど座標変換した、 $X'^sY'^s$ 座標を斜面座標 $X^sY^s$ 座標系に座標変換すると点 B $(x_B^s, y_B^s)$ , 点 C $(x_C^s, y_C^s)$ の座標は

$$x_B^s = -l_{AB} \frac{\sin \beta}{\cos \alpha} \quad (4.27)$$

$$y_B^s = l_{AB} \cos \beta \quad (4.28)$$

$$x_c^s = -l_{AC} \frac{\sin(\beta + \omega)}{\cos \alpha} \quad (4.29)$$

$$y_c^s = l_{AC} \cos(\beta + \omega) \quad (4.30)$$

次に斜面の深度モデルについて検討する．図 4.10 に示すように斜面の傾きは $\alpha$ であり，各点での深度は点 A での深度を $H_0$ とすると各点での深度は $H = H_0 + \Delta H_i (i = B, C)$ として表される．図 4.10 から各 $\Delta H_B, \Delta H_C$ は以下の式(4.31),(4.32)で求める事ができる．ただし深度 $\Delta H_i$ は下向きを正として扱う．

$$\Delta H_B = -x_B^s \sin \alpha = l_{AB} \sin \beta \tan \alpha \quad (4.31)$$

$$\Delta H_C = -x_C^s \sin \alpha = l_{AC} \sin(\beta + \omega) \tan \alpha \quad (4.32)$$

以上の 2 式を用いてシミュレーションを行う．

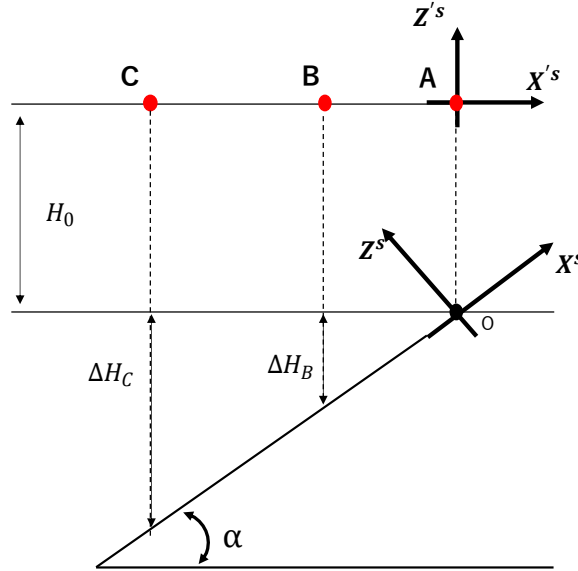


図 4.10 斜面座標と深度モデル

#### 4.5 シミュレーションによる 2 手法の比較

以下の図 4.11 には手法 1，図 4.12 には手法 2 のシミュレーション結果を示している．両検証共に，機体が斜面に対して $\beta = 60deg$ だけ回転しており $\beta = 0$ となるような制御を行っている．また，それぞれ斜面の傾斜角 $\alpha$ が 10deg, 20deg, 30deg, 40deg の 4 つのパターンの斜面について検証を行った．

手法 1 の結果について，斜面の傾斜角が 10deg, 20deg の場合はオーバーシュートなく $\beta = 0$ に収束しているが，30deg, 40deg になると収束の際にオーバーシュートが起きていることがわかる．また，収束時間に関しても，斜面の傾斜角が大きくなるにつれ収束に時間がかかっていることから，この手法では異なる斜面に対して，応答特性が異なってしまう事がわかる．

対して，手法 2 については全ての傾斜角について，およそ 2.5 秒で定常値に収束しており，オーバーシュートなく収束していることがこの結果からわかる．未知の斜面に対して斜面の傾斜及び回転角を推定する際には，斜面によって制御性能が異なることは好ましくないと考え，本研究では傾斜角によらず，同様の制御性能を出すことができる手法 2 を用いる．

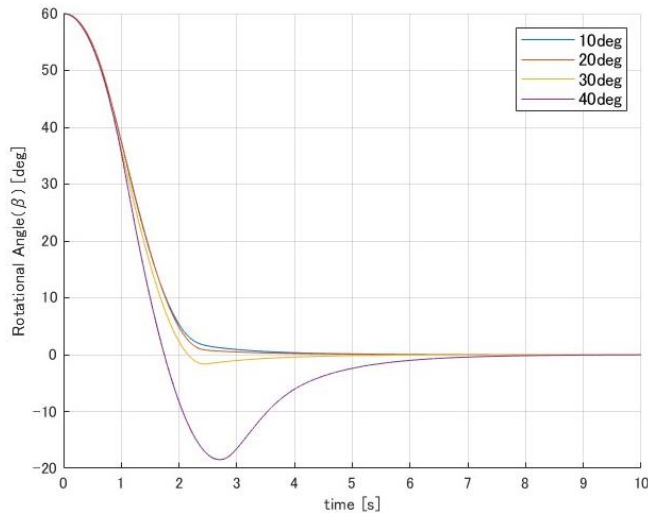


図 4.11 手法 1 を用いたシミュレーション結果

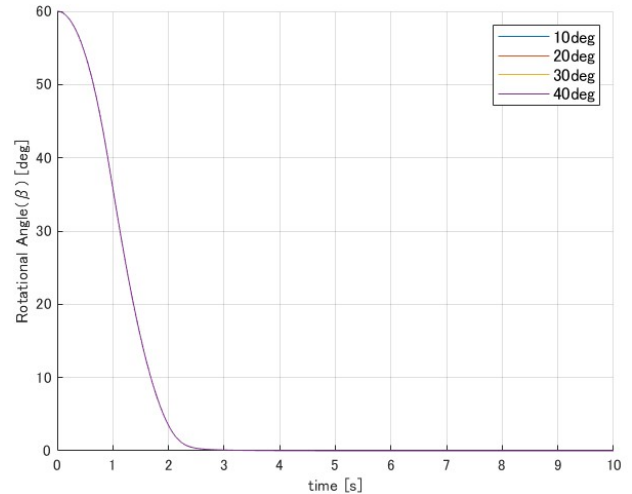


図 4.12 手法 2 を用いたシミュレーション結果

## 4.6 深度カメラを用いた実装

### 4.6.1 使用した深度カメラとソフトウェア

本研究で用いた深度カメラ Intel®RealSense™Depth Camera D435i について、外観を図 4.13 に諸元を以下の表 4.2 に示す。この深度カメラは図 4.13 にある様に 2 か所のステレオカメラの視差から深度情報を求めるものである。



図 4.13 使用した深度カメラの外観

表 4.2 Intel®RealSense™Depth Camera D435i の諸元

最大視野角(長手方向) [deg]	69
最大視野角(短手方向) [deg]	42
解像度	848×480
フレームレート [Hz]	60

出典：Intel®RealSense™Depth Camera D435i Datasheet<sup>(6)</sup>

本研究では ROS 2 を用い、Madgwick フィルタを使用した際と同様に公式によって公開されているプラグイン<sup>(7)</sup>を用いて深度情報を取得する。

#### 4.6.2 深度情報取得位置

深度カメラによる深度情報取得位置を以下の図 4.14 に示す。取り付け位置としては、機体の重心位置に、取り付け方向としては、機体の長手方向とカメラの長手方向を一致する様に取り付けることで、機体の長手方向に視野を広く取ることができるように取り付けた。図の黒点の位置がカメラ取付場所、赤点位置が深度情報取得位置である。

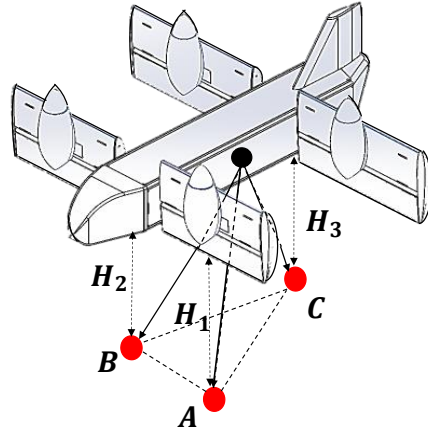


図 4.14 深度カメラ取付位置と深度情報取得位置

#### 4.6.3 深度情報の取得

ROS2 プラグインでは深度情報は特定のピクセルを指定することで取得することができる。そこで本研究では以下の図 4.15 に示すような位置関係とピクセルを指定し、さらに指定ピクセルに小石や雑草などの障害物があった場合には、正確な深度を取得することができないと考え、指定位置周辺のピクセルの深度情報も取得しそれらを平均することでその位置での深度情報としている。

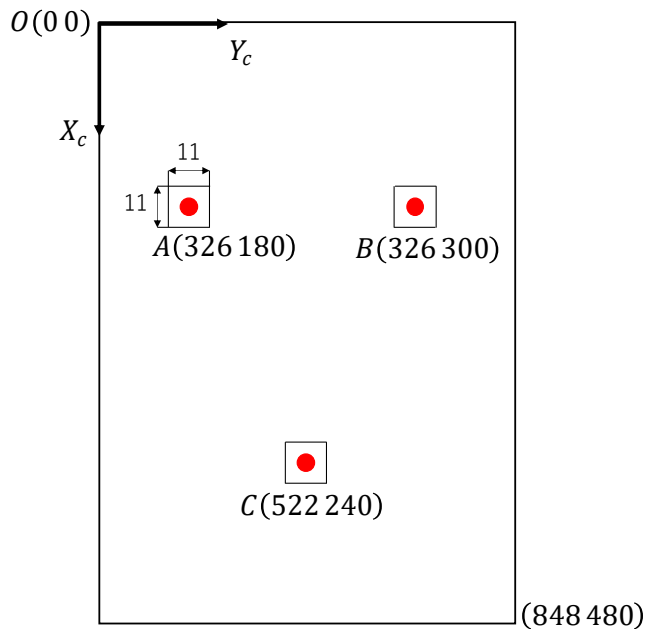


図 4.15 深度情報取得位置 (ピクセル)

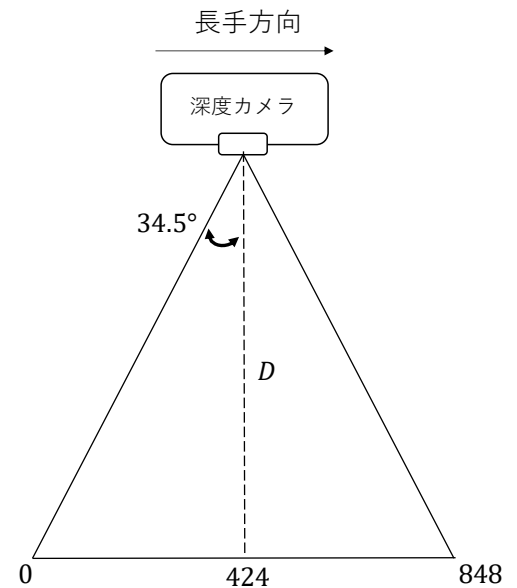


図 4.16 画角とピクセル間の距離の関係

上で示したの図 4.16 について、まずは画角から 1 ピクセルの長さ $L_0$ を計算する。 $D$ [mm]はカメラからの深度、 $34.5^\circ$  は深度カメラの最大画角(長手方向)の半分の数値として式(4.33)の様に表される。

$$L_0 = D \times \frac{\tan(34.5)}{424} \quad (4.33)$$

例として $D = 1000\text{mm}$ の時 $11 \times 11$ の長手方向の長さ $L_x$ を求めるには 1 ピクセルの長さを 11 倍すればいいので、

$$L_x = 1000 \times \frac{\tan(34.5)}{424} \times 11 = 17.8304[\text{mm}] \quad (4.34)$$

同様に、短手方向の長さ $L_y$ は

$$L_y = 1000 \times \frac{\tan(21.0)}{240} \times 11 = 17.5938[\text{mm}] \quad (4.35)$$

よって図 4.15 に示すような  $11 \times 11$  の範囲はおよそ  $17.8\text{mm} \times 17.6\text{mm}$  であることがわかる。

#### 4.6.4 深度情報取得位置の関係

4.1 及び 4.2 で述べた斜面の傾斜角推定と斜面に対する回転角の推定には、深度情報取得位置の関係性が必要である。ここではそれぞれの位置関係について述べる。以下の図 4.17 は図 4.15 に示した点 A, B の関係である。点 A, B はそれぞれ中心から等距離に配置してあるのでそれぞれの点 A での短手方向画角  $FOV_{AS}$  と点 B での画角  $FOV_{BS}$  は等しくなる。これらの画角  $FOV_A, FOV_B$  を求める。

点 A 及び点 B はそれぞれ、中心から 60 ピクセルずつ離れているので、中心から点 A までの距離  $L_A$  を計算すると、中心カメラ深度を  $D$ [mm] とおくと

$$L_A = L_B = D \times \frac{\tan(21.0)}{240} \times 60 \quad (4.36)$$

$L_A$  と  $D$  との距離の比から

$$FOV_{AS} = FOV_{BS} = \text{atan}\left(\frac{L_A}{D}\right) = \text{atan}\left(\frac{\tan(21.0)}{4}\right) = 5.48166 \approx 5.4817[\text{deg}] \quad (4.37)$$

と計算できる。回転角の推定には点 AB 間の距離  $l_{AB}$  が必要であるので、図 4.17 に示すような少し傾いた面（赤線）での距離  $l_{AB'}$  を考えると、

$$l_{AB} = L_A + L_B \quad (4.38)$$

$$L_A = H_A \tan(FOV_{AS}) \quad (4.39 - 1)$$

$$L_{B'} = H_{B'} \tan(FOV_{BS}) \quad (4.39 - 2)$$

よって

$$l_{AB'} = (H_A + H_{B'}) \tan(5.48166) \quad (4.40)$$

となり、傾いた面の場合でも画角とそれぞれの深度から距離  $l_{AB}$  求めることができる。

同様に点 BC 間の距離  $l_{BC}$  についても同様に長手方向の画角  $FOV_{BL}, FOV_{CL}$  は

$$FOV_{BL} = FOV_{CL} = \text{atan}\left(\frac{L_B}{D}\right) = \text{atan}\left(\frac{\tan(34.5)}{424} \times 98\right) = 9.02617 \approx 9.0262[\text{deg}] \quad (4.41)$$

$$l_{BC} = (H_B + H_C) \tan(9.02617) \quad (4.42)$$

求めることができる。また、 $l_{AB}$  と  $l_{BC}$  がなす角  $\omega$  は

$$\omega = \text{acos}\left(\frac{l_{AB}}{2l_{BC}}\right) = \text{acos}\left(\frac{(H_A + H_B) \tan(5.48166)}{2\{(H_B + H_C) \tan(9.02617)\}}\right) \quad (4.43)$$

で求めることができる。

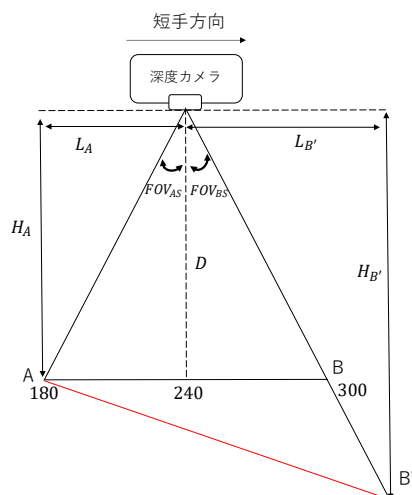


図 4.17 深度情報取得位置と画角の関係

## 4.7 実機を用いた斜面傾斜推定及び回転角推定実験

### 4.7.1 実験方法

前節で求めた関係式を用いて実機を用いて実験を行った。実験方法は以下の図 4.18 に示すように、三脚を用いて深度カメラを水平に固定する。その下では斜面に仮定した黒色の板を動かすことによって、疑似的に斜面の傾斜角 $\alpha$ と $\beta$ を生成する。板の角度については図右下のモーションキャプチャを用いて、板のオイラー角を取得することで、それぞれ傾斜角 $\alpha$ 、回転角 $\beta$ の真値としてしている。得られた真値と深度カメラを用いて算出された推定値を比較し推定精度について比較、検証する。

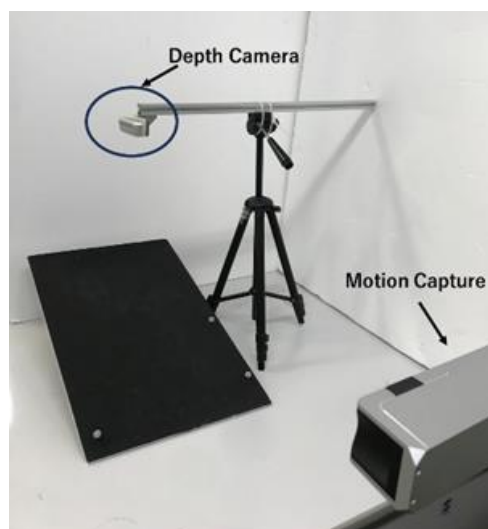


図 4.18 実験の様子

### 4.7.2 実験結果

以下の図 4.19 に斜面の傾斜角推定の結果、図 4.20 には回転角推定の結果を示す。それぞれ、赤線がモーションキャプチャを用いた斜面の真値、青線が深度カメラを用いた推定値である。

図 4.19 について、斜面の傾斜角推定は推定の立ち上がり時間におよそ 0.4[s]の遅れがみられるものの、精度良く追従できていることがわかる。また、推定値の精度に関してはピーク時に最大 2[deg]程の推定誤

差が見られたが、これらは三角関数の計算時に生じる誤差の範囲内であると考え、傾斜角の推定に関しては実用に値する精度を得る事ができた。

図 4.20 について、斜面の回転角推定は真値に対して、推定結果+5~10[deg]程度オフセットされていることがわかったこの結果に関しては次項にて考察を行う。

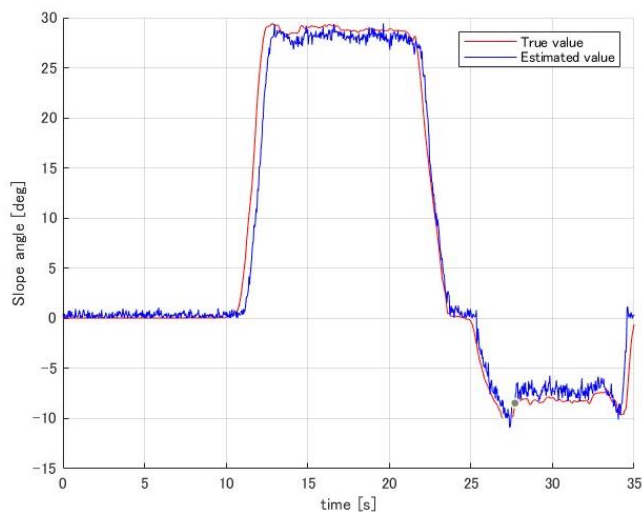


図 4.19 斜面傾斜角推定の結果

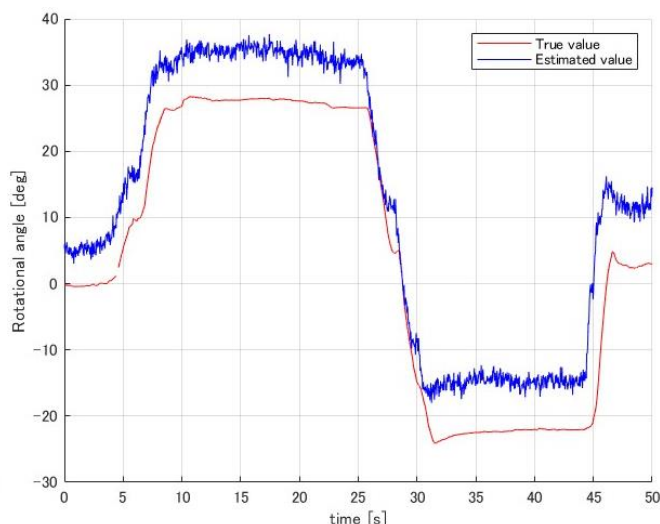


図 4.20 回転角推定の結果

### 4.7.3 考察

図 4.20 の結果について考察する. 以下の図 4.21 にこの実験の推定に用いられた 3 点の深度情報を示す. 真値では 0~4[s]まで 0[deg]であるが、実験結果では 5[deg]の推定結果を算出している. これは図 4.21 に示す 0~4[s]での前方 2 点の深度情報に差があることでこの誤差が生じていると考察した. そこで以下の図 4.22 には 1 回目のデータを基に深度データをキャリブレーションした推定結果を示した. オフセット後はピーク時の推定誤差が緩和されたものの、それでもなお、5[deg]程度推定誤差が現れる結果となった. この推定誤差は、斜面の傾きが一方方向で無い事が原因の一つであると考えられ、今後の課題としては、斜面座標 $Y_g$ 方向にも傾きがある場合の、誤差の補償方法について検討していく必要があると考えている。

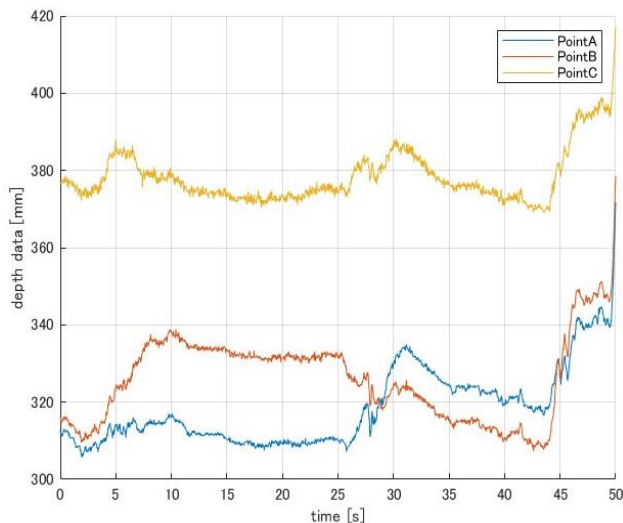


図 4.21 推定に使用した深度データ

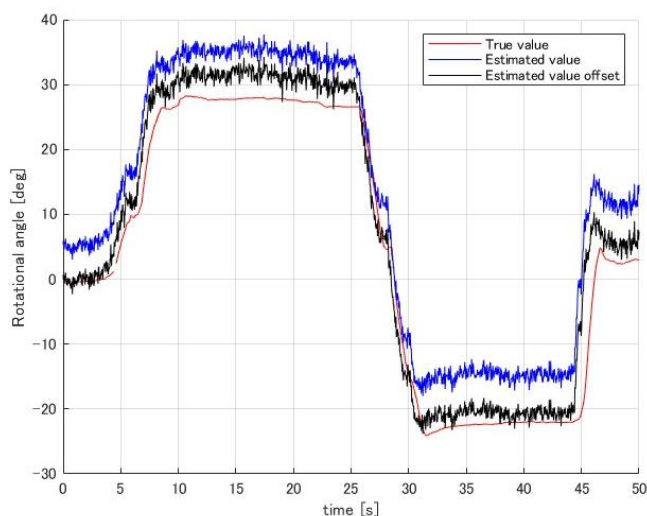


図 4.22 深度データオフセット後の推定データ



## 5. 小型モデルを用いた最適制御検証

### 5.1 目的

前年度試験ではホバリング状態を平衡状態として非線形システムを線形化し、線形化したモデルに対して積分型サーボ系<sup>(8)</sup>を適用しコントローラを設計していた。しかし、実機を用いた試験では離陸することができなかった。そこで、本章では実機を模した小型モデルを作成し、同様のコントローラを適用する事で最適制御の有用性について再度検証することを目的としている。

### 5.2 小型モデル作成

#### 5.2.1 機体設計

この機体作成の目的は離陸およびホバリング時の検証であるため、ティルト角は 90 度に固定した状態で機体を小型化する。以下の図 5.1 は実機の寸法を示した図である。この実機は前翼の後流の影響による後翼の揚力減少を補償するため、後翼の長さが前翼より少し長くなっている。そのため、プロペラの位置も前後で差がある。また、機体中心から前翼と後翼までは等しい距離にある、という特徴を継承し、以下の図 5.2 に示すような機体を設計した。

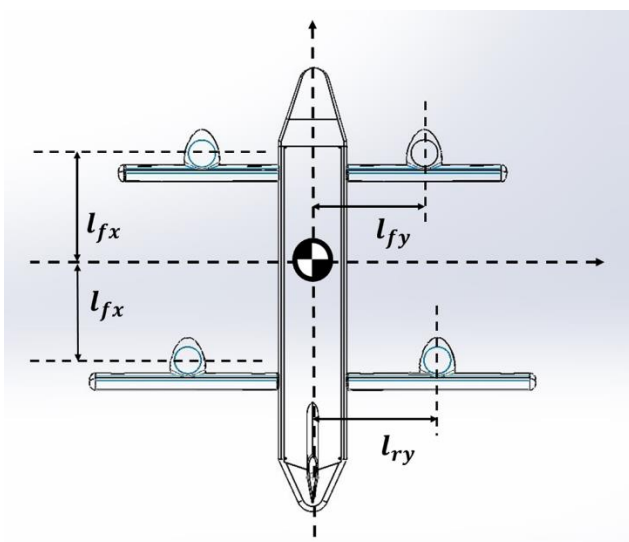


図 5.1 実機のワイヤフレーム図

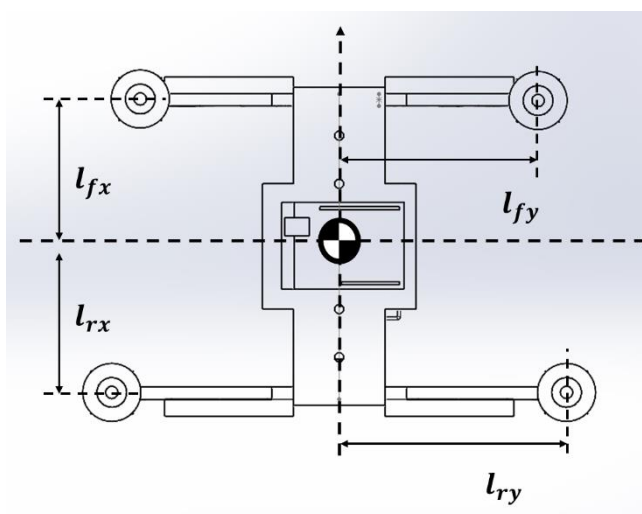


図 5.2 小型モデルのワイヤフレーム図

また以下の表 5.1 にはそれぞれの機体の寸法を示す。実機の前翼プロペラの位置 $l_{fy}$ と $l_{ry}$ の長さの比は表 5.1 より $l_{fy}:l_{ry} = 210:240$ である。この寸法比から小型機の前後の腕の長さを決定した。

表 5.1 各機体の寸法比較

	実機	小型機
$l_{fx}$ [mm]	200	100
$l_{rx}$ [mm]	200	100
$l_{fy}$ [mm]	210	130
$l_{ry}$ [mm]	240	148.6

## 5.3 使用機材

### 5.3.1 フレーム

フレームは Solidworks を用いて 3D モデルを作成し、作成したモデルをパーツごとに分割して 3D プリンタで印刷し、印刷後に組み立てる。以下の図 5.3 には完成したフレームの外観を示す。

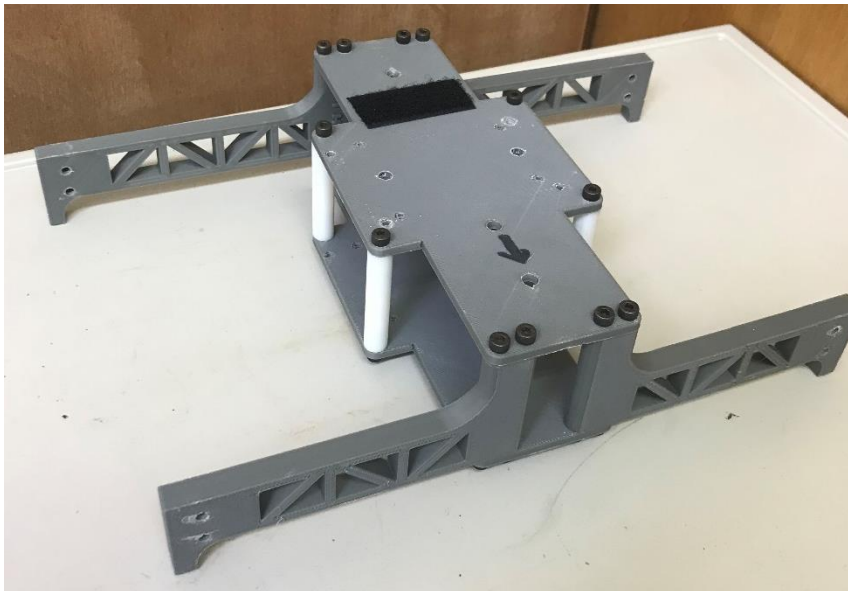


図 5.3 小型モデルフレーム

### 5.3.2 プロペラ駆動用モータ及び ESC

プロペラの駆動モータには先行研究<sup>2)</sup>で用いられていたモータを用いる。以下の図 5.4 には示す。また、モータの制御には昨年度実験で用いられたものと同様の ESC を用いる。



図 5.4 使用モータ

### 5.3.3 バッテリ

プロペラ駆動用モータには Li-Po バッテリを使用した。また、フライトコントローラ駆動用にも Li-Po バッテリを使用している。以下の表 5.2 にはそれぞれのバッテリの諸元を示す。

表 5.2 搭載バッテリの諸元

	ブラシレスモータ用	フライトコントローラ用
バッテリタイプ	Li-Po	Li-Po
セル数 [Cell]	4	2
バッテリ容量 [mAh]	5000	1000
C レート [C]	40	50

### 5.3.4 プロペラ

以上のフレーム、プロペラ駆動用モータ、ESC、バッテリの重量から、プロペラのサイズおよびピッチを検討し、最終的には逆ピッチ用のプロペラが存在するサイズである7×5のプロペラに決定した

### 5.3.5 フライトコントローラ

フライトコントローラにはモータの駆動に必要な、PWM 信号を専用ライブラリを用いることで生成しやすいことや、専用 IDE を用いることで比較的簡単にプログラムを実装できる Arduino シリーズを用いて行う。製作初期段階では Arduino Uno を用いていたが、メモリが不足してしまう事が判明し、Arduino Mega に変更した。以下の図 5.5 には使用した Arduino Mega の外観、表 5.3 には Arduino Uno と Arduino Mega の比較を示す。



図 5.5 Arduino Mega

表 5.3 Arduino Mega と Arduino Uno の比較

	Arduino Mega	Arduino Uno
寸法[mm]	101.52 × 53.3	68.6 × 53.4
動作電圧[V]	5	5
搭載チップ	ATmega2560	ATmega328P
メモリ[KB]	256	32KB
ピン数	54	14

出典：Arduino 公式ストア<sup>(9)</sup>より引用

### 5.3.6 姿勢推定用 IMU

姿勢推定用の IMU ユニットは MPU6050 を使用している. このユニットから加速度情報, 角速度情報を取得し, 本研究で用いられている Madgwick フィルタを用いて姿勢推定を行う. 以下の図 5.6 には MPU6050 の外観を示す.



図 5.6 使用した IMU ユニット

### 5.3.7 完成機体

以下の図 5.7 には作成した小型モデルの外観を示す.



図 5.7 作成した小型機体

## 5.4 運動モデル

本節で使用する記号について, あらかじめ以下の表 5.4 にまとめる

表 5.3 使用記号

$I_{xx} I_{yy} I_{zz}$	:	機体座標軸周りの慣性モーメント[N/m <sup>2</sup> ]	$B$	:	推力:反トルク比率[-]
$P Q R$	:	機体座標軸での角速度[rad/s]	$B'$	:	PWM:反トルク比率[-]
$\phi \theta \psi$	:	ZXY 系でのオイラー角[rad]	$u_{fr}$	:	右前方プロペラ PWM 周波数[ $\mu$ s]
$T_{fr}$	:	右前方プロペラからの推力[N]	$u_{rr}$	:	右後方プロペラ PWM 周波数[ $\mu$ s]
$T_{rr}$	:	右後方プロペラからの推力[N]	$u_{rl}$	:	左後方プロペラ PWM 周波数[ $\mu$ s]
$T_{rl}$	:	左後方プロペラからの推力[N]	$u_{fl}$	:	左前方プロペラ PWM 周波数
$T_{fl}$	:	左前方プロペラからの推力[N]			
$A_0$	:	記号 A の平衡点での値[-]			

### 5.4.1 座標系

機体固定座標系，地面固定座標系は参考文献(1)と同様に，機体前方方向に $X^B$ とり，地面固定座標系 $XYZ$ についてはEND座標を用いる．以下の図5.8にはそれぞれの座標系を示す．

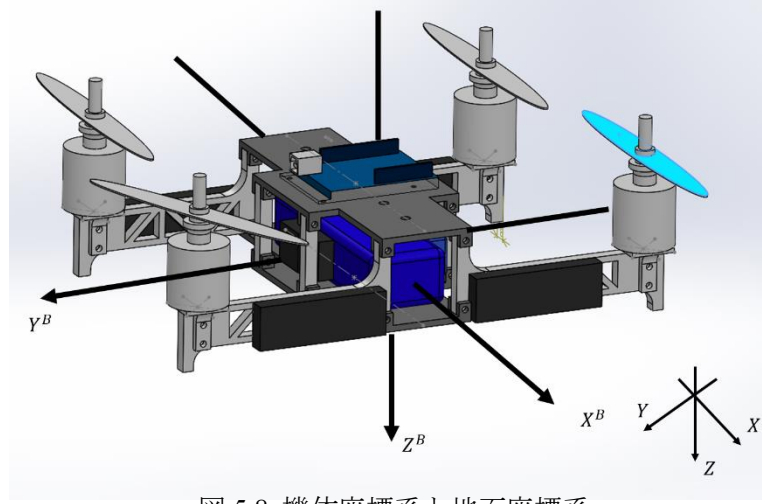


図 5.8 機体座標系と地面座標系

### 5.4.2 非線形モデル

小型機体の非線形運動モデルについても参考文献(1)を引用する．ただし，実機モデルとは異なりテールト角は常に 90 度であるのでその点については留意する．以下の式(5.1)には機体の回転運動に関する非線形モデルを示す．

$$\begin{bmatrix} \dot{P} \\ \dot{Q} \\ \dot{R} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \frac{1}{I_{xx}} \left( -(I_{zz} - I_{yy})QR + (-l_{fy}(T_{fr} - T_{fl}) - l_{ry}(T_{rr} - T_{rl})) \right) \\ \frac{1}{I_{yy}} \left( -(I_{xx} - I_{zz})RP + (l_{fx}(T_{fr} + T_{fl}) - l_{rx}(T_{rr} + T_{rl})) \right) \\ \frac{1}{I_{zz}} \left( -(I_{yy} - I_{xx})PQ - B(-T_{fr} + T_{fl} + T_{rr} - T_{rl}) \right) \\ P \cos \theta + R \sin \theta \\ P \tan \phi \sin \theta + Q - R \tan \phi \cos \theta \\ -P \frac{\sin \theta}{\cos \phi} + R \frac{\cos \theta}{\cos \phi} \end{bmatrix} \quad (5.1)$$

### 5.4.3 線形モデル

前項で示された非線形運動モデル(5.1)を最適制御に適用するため線形化を行う．平衡点は $P = Q = R = \Phi = \theta = \psi = 0[\text{deg}]$ のホバリング状態を平衡点とする．前項の(5.1)式の各変数を平衡点での値と微小偏差の形に変換すると一行目から

$$\frac{d}{dt}(P_0 + \delta P) = \frac{1}{I_{xx}} \left( (I_{yy} - I_{zz})(Q_0 + \delta Q)(R_0 + \delta R) - (l_{fy}(T_{fr} - T_{fl}) + l_{ry}(T_{rr} - T_{rl})) \right) \quad (5.2-1)$$

$$\frac{d}{dt}(Q_0 + \delta Q) = \frac{1}{I_{yy}} \left( (I_{zz} - I_{xx})(R_0 + \delta R)(P_0 + \delta P) + (l_{fx}(T_{fr} + T_{fl}) - l_{rx}(T_{rr} + T_{rl})) \right) \quad (5.2-2)$$

$$\frac{d}{dt}(R_0 + \delta R) = \frac{1}{I_{zz}} \left( (I_{xx} - I_{yy})(P_0 + \delta P)(Q_0 + \delta Q) - B(-T_{fr} + T_{fl} + T_{rr} - T_{rl}) \right) \quad (5.2-3)$$

$$\frac{d}{dt}(\Phi_0 + \delta \Phi) = (P_0 + \delta P) \cos(\theta_0 + \delta \theta) + (R_0 + \delta R) \sin(\theta_0 + \delta \theta) \quad (5.2-4)$$

$$\frac{d}{dt}(\theta_0 + \delta \theta) = (P_0 + \delta P) \tan(\phi_0 + \delta \phi) \sin(\theta_0 + \delta \theta) + (Q_0 + \delta Q) - (R_0 + \delta R) \tan(\phi_0 + \delta \phi) \cos(\theta_0 + \delta \theta) \quad (5.2-5)$$

$$\frac{d}{dt}(\psi_0 + \delta\psi) = -(P_0 + \delta P) \frac{\sin(\theta_0 + \delta\theta)}{\cos(\phi_0 + \delta\phi)} + (R_0 + \delta R) \frac{\cos(\theta_0 + \delta\theta)}{\cos(\phi_0 + \delta\phi)} \quad (5.2-6)$$

以上の式における微小偏差の三角関数について，平衡点でテーラー展開して，2次以上を近似すると

$$\sin(\delta\theta) \cong \delta\theta - \frac{\delta\theta^3}{3!} + \frac{\delta\theta^5}{5!} + \dots \cong \delta\theta \quad (5.3-1)$$

$$\cos(\delta\theta) \cong 1 - \frac{\delta\theta^2}{2!} + \frac{\delta\theta^4}{4!} - \dots \cong 1 \quad (5.3-2)$$

(5.2-1)~(5.2-6)について微小項の積は無視すると，

$$\frac{d}{dt} \delta P = \frac{1}{I_{xx}} \left( - (l_{fy}(T_{fr} - T_{fl}) + l_{ry}(T_{rr} - T_{rl})) \right) \quad (5.4-1)$$

$$\frac{d}{dt} \delta Q = \frac{1}{I_{yy}} \left( l_{fx}(T_{fr} + T_{fl}) - l_{rx}(T_{rr} + T_{rl}) \right) \quad (5.4-2)$$

$$\frac{d}{dt} \delta R = \frac{1}{I_{zz}} \left( -B(-T_{fr} + T_{fl} + T_{rr} - T_{rl}) \right) \quad (5.4-3)$$

$$\frac{d}{dt} \delta\phi = \delta P \quad (5.4-4)$$

$$\frac{d}{dt} \delta\theta = \delta Q \quad (5.4-5)$$

$$\frac{d}{dt} \delta\psi = \delta R \quad (5.4-6)$$

となる．次に制御入力である推力 $T_i$ についても平衡点の値と微小偏差の形に変換する．ただし，平衡状態では以下の式が成り立つ．

$$l_{fx}T_{fr0} + l_{fy}T_{fl0} - l_{rx}T_{rr0} - l_{ry}T_{rl0} = 0 \quad (5.5-1)$$

$$-l_{fy}T_{fr0} + l_{fy}T_{fl0} - l_{ry}T_{rr0} + l_{ry}T_{rl0} = 0 \quad (5.5-2)$$

以上の式より式(5.4-1)~(5.4-3)は以下の様に変形できる．

$$\frac{d}{dt} \delta P = \frac{1}{I_{xx}} \left( - (l_{fy}(\delta T_{fr} - \delta T_{fl}) + l_{ry}(\delta T_{rr} - \delta T_{rl})) \right) \quad (5.6-1)$$

$$\frac{d}{dt} \delta Q = \frac{1}{I_{yy}} \left( l_{fx}(\delta T_{fr} + \delta T_{fl}) - l_{rx}(\delta T_{rr} + \delta T_{rl}) \right) \quad (5.6-2)$$

$$\frac{d}{dt} \delta R = \frac{1}{I_{zz}} \left( -B(-\delta T_{fr} + \delta T_{fl} + \delta T_{rr} - \delta T_{rl}) \right) \quad (5.6-3)$$

従って(5.6-1)~(5.6-3)と(5.4-4)~(5.4-6)を行形式に直すと，

$$\frac{d}{dt} \begin{bmatrix} \delta P \\ \delta Q \\ \delta R \\ \delta\phi \\ \delta\theta \\ \delta\psi \end{bmatrix} = \begin{bmatrix} \frac{1}{I_{xx}} \left( - (l_{fy}(\delta T_{fr} - \delta T_{fl}) + l_{ry}(\delta T_{rr} - \delta T_{rl})) \right) \\ \frac{1}{I_{yy}} \left( l_{fx}(\delta T_{fr} + \delta T_{fl}) - l_{rx}(\delta T_{rr} + \delta T_{rl}) \right) \\ \frac{1}{I_{zz}} \left( -B(-\delta T_{fr} + \delta T_{fl} + \delta T_{rr} - \delta T_{rl}) \right) \\ \delta P \\ \delta Q \\ \delta R \end{bmatrix} \quad (5.7)$$

この式を状態空間表現で表すと

$$\begin{bmatrix} \delta P \\ \delta Q \\ \delta R \\ \delta \phi \\ \delta \theta \\ \delta \psi \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta P \\ \delta Q \\ \delta R \\ \delta \phi \\ \delta \theta \\ \delta \psi \end{bmatrix} + \begin{bmatrix} -\frac{l_{fy}}{I_{xx}} & \frac{l_{fy}}{I_{xx}} & \frac{l_{ry}}{I_{xx}} & -\frac{l_{ry}}{I_{xx}} \\ \frac{l_{fx}}{I_{yy}} & \frac{l_{fx}}{I_{yy}} & -\frac{l_{rx}}{I_{yy}} & -\frac{l_{rx}}{I_{yy}} \\ \frac{B}{I_{zz}} & -\frac{B}{I_{zz}} & \frac{B}{I_{zz}} & -\frac{B}{I_{zz}} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta T_{fr} \\ \delta T_{rr} \\ \delta T_{rl} \\ \delta T_{fl} \end{bmatrix} \quad (5.8)$$

となる。

式(5.8)の入力値は単位が[N]であるが、実際に制御を行う際には PWM 信号を入力するため、入力の次元を PWM 信号のデューティ比に変換する必要がある。推力に関しても、平衡点周りで線形に変化するものと仮定し、以下の様に平衡点と平衡点周りの微小変化に変換する。

$$T_{i(i=fr,rr,rl,fl)} = u_0 + \delta u \quad (5.9)$$

よって

$$\begin{bmatrix} \delta P \\ \delta Q \\ \delta R \\ \delta \phi \\ \delta \theta \\ \delta \psi \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta P \\ \delta Q \\ \delta R \\ \delta \phi \\ \delta \theta \\ \delta \psi \end{bmatrix} + \begin{bmatrix} -\frac{l_{fy}}{I_{xx}} & \frac{l_{fy}}{I_{xx}} & \frac{l_{ry}}{I_{xx}} & -\frac{l_{ry}}{I_{xx}} \\ \frac{l_{fx}}{I_{yy}} & \frac{l_{fx}}{I_{yy}} & -\frac{l_{rx}}{I_{yy}} & -\frac{l_{rx}}{I_{yy}} \\ \frac{B}{I_{zz}} & -\frac{B}{I_{zz}} & \frac{B}{I_{zz}} & -\frac{B}{I_{zz}} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta u_{fr} \\ \delta u_{rr} \\ \delta u_{rl} \\ \delta u_{fl} \end{bmatrix} \quad (5.10)$$

となり運動モデルの線形化が完了した。

## 5.5 システム同定

### 5.5.1 機体重量と慣性モーメントについて

機体重量及び、慣性モーメントについては以下の表 5.4 に示す。慣性モーメントは以下の図 5.9 の様に 2 点釣り法を用いて求めた。

表 5.5 小型モデル機の重量と慣性モーメント

重量[kg]	1.9
$I_{xx}$ [kg m <sup>2</sup> ]	0.01511
$I_{yy}$ [kg m <sup>2</sup> ]	0.01039
$I_{zz}$ [kg m <sup>2</sup> ]	0.02403





図 5.9 二点釣法を用いた慣性モーメント計測

### 5.5.2 推力及びプロペラの回転に伴う反トルクについて

プロペラを取り付けたモータを 6 軸力覚系に取り付け、特定のデューティ比で駆動させた後、10 秒間力覚系で計測し、得られた値の平均値をその回転数での推力としている。また、本研究で用いられている ESC はブラックボックス化されており、ESC に入力した PWM 周波数とそれに対応した推力と反トルクでモデリングを行う。以下の図 5.10 に推力の計測結果を示す。図 5.11 に反トルクの計測結果を示す。

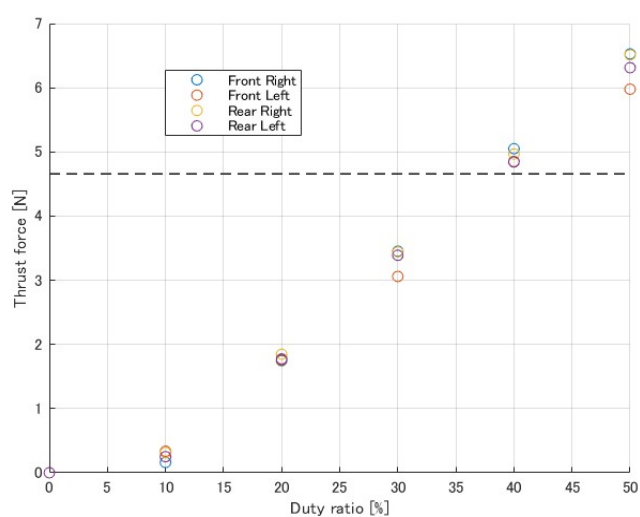


図 5.10 各モータの推力計測結果

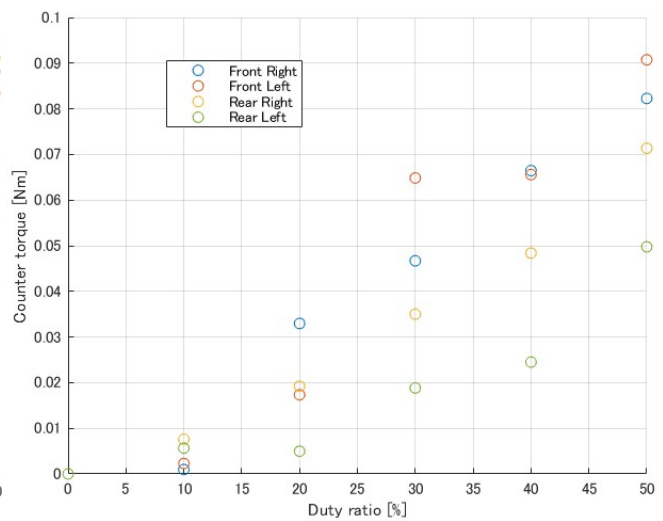


図 5.11 各モータの反トルク計測結果

以上の結果について、図 5.10 中の黒線は機体重量から算出されたモーター機あたりに必要な推力である。計測結果からそれぞれのモータに大きなばらつきは無く、各モータ 36%程度のデューティ比で平衡点を取ることがわかった。

一方、反トルクの測定結果にはモータ毎にばらつきが現れることがわかった。この結果はモータの回転による振動が原因であり、制御を行う際にそれほど大きな影響は無いと考えている。以下の表 5.5 には推力と反トルクに関するシステム同定結果をまとめた物を示す

表 5.6 システム同定結果

	右前方	左前方	右後方	左後方	平均
平衡点(推力) [%]	36.063	39.38	36.51	37.17	37.28
傾き(推力)	0.01594	0.01445	0.01552	0.01507	0.01525
傾き(反トルク)	$1.776 \times 10^{-4}$	$1.975 \times 10^{-4}$	$1.414 \times 10^{-4}$	$0.9119 \times 10^{-4}$	$1.519 \times 10^{-4}$
分散(反トルク)	0.0011	0.0015	0.00072	0.00034	

コントローラを設計する場合は推力の測定結果には大きなバラツキは見られなかったことから、平衡点及び傾きには平均値を用いる。反トルクに関しては結果にバラツキが見られたことから、平均ではなく、最も分散が小さかった右後方のデータを使用する。

## 5.6 制御器設計

### 5.6.1 LQR コントローラ設計

前節で得られた線形モデルを基に LQR コントローラを設計する。LQR コントローラは状態方程式の評価関数  $J$  を最小にする最適なゲイン  $K$  を得る手法である。以下の式(5.11)には用いた評価関数を示す

$$J = \int_0^{\infty} (\mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{u}^T \mathbf{R} \mathbf{u}) dt \quad (5.11)$$

ここで、 $\mathbf{x}, \mathbf{u}$  はそれぞれ以下の式(5.12-1)(5.12-2)の様に示される。

$$\mathbf{x} = \begin{bmatrix} \delta P \\ \delta Q \\ \delta R \\ \delta \phi \\ \delta \theta \\ \delta \psi \end{bmatrix} \quad (5.12 - 1)$$

$$\mathbf{u} = \begin{bmatrix} \delta u_{fr} \\ \delta u_{rr} \\ \delta u_{rl} \\ \delta u_{fl} \end{bmatrix} \quad (5.12 - 2)$$

また、コントローラの設計は Matlab の `lqr` コマンドを用いて行う。また、重み行列  $\mathbf{Q}, \mathbf{R}$  については次項でシミュレーションを行い試行錯誤的に決定する。

### 5.6.2 シミュレーション

下図 5.12 で示された制御系を Simulink を用いて検証し、重み行列  $\mathbf{Q}, \mathbf{R}$  を決定する。シミュレーション方法としては、まずはロール軸、またはピッチ軸のみでそれぞれ非線形シミュレーションを行いながら重み行列を調整していき、各軸周りで十分な結果を得る事ができれば、3 軸でのシミュレーションを行い、最終的な重み行列を決定する。

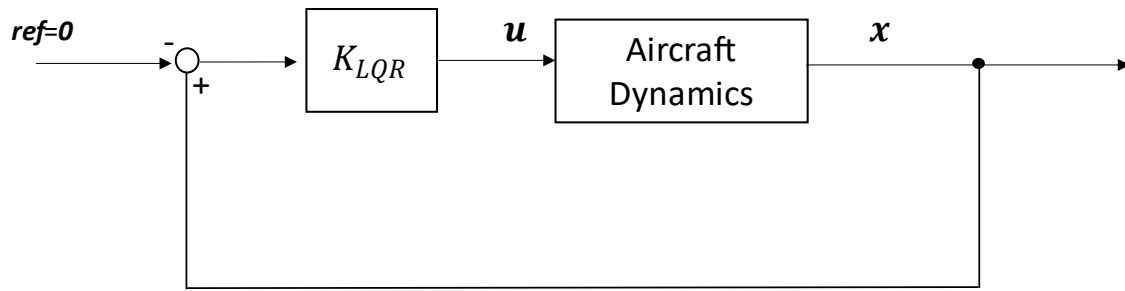


図 5.12 使用したブロック線図

以下の図 5.13 にはロール軸のみでのシミュレーション結果，図 5.14 にはピッチ軸のみでのシミュレーション結果，図 5.15,5.16 には 3 軸でのシミュレーション結果，図 5.17 では 3 軸と 1 軸のみでの結果を比較したものを示している

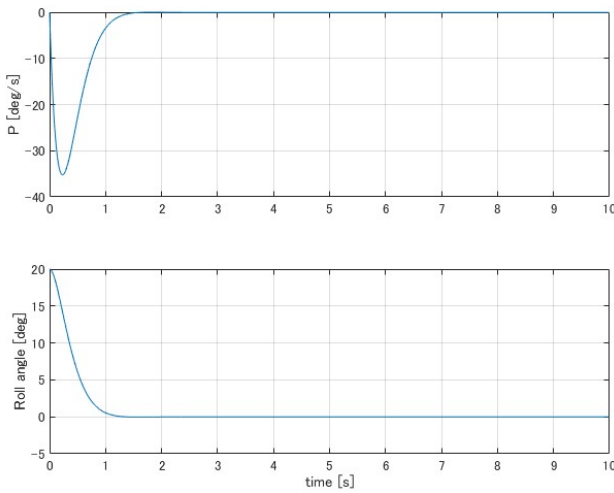


図 5.13 ロール軸周りのシミュレーション結果

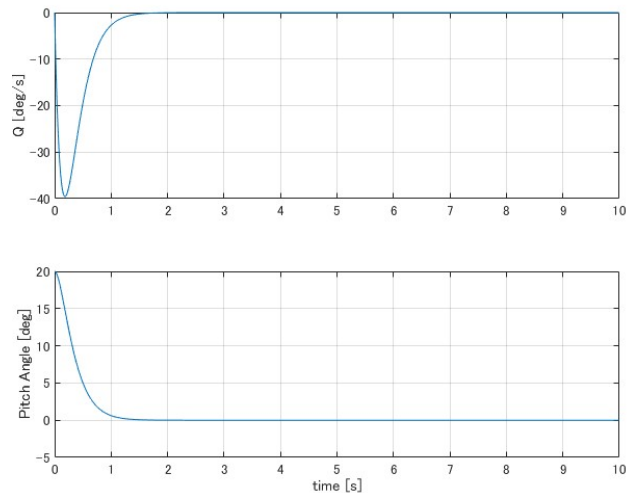


図 5.14 ピッチ軸周りのシミュレーション結果

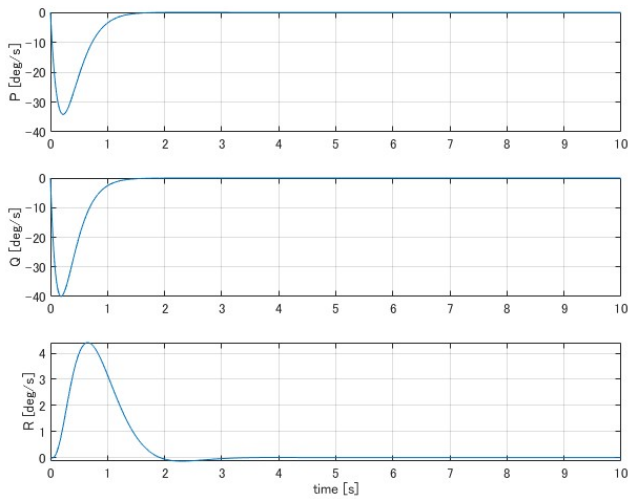


図 5.15 自由度角速度シミュレーション結果

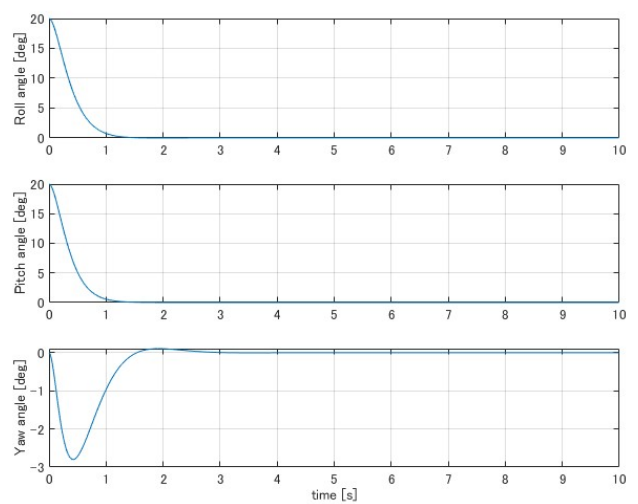


図 5.16 自由度姿勢角シミュレーション結果

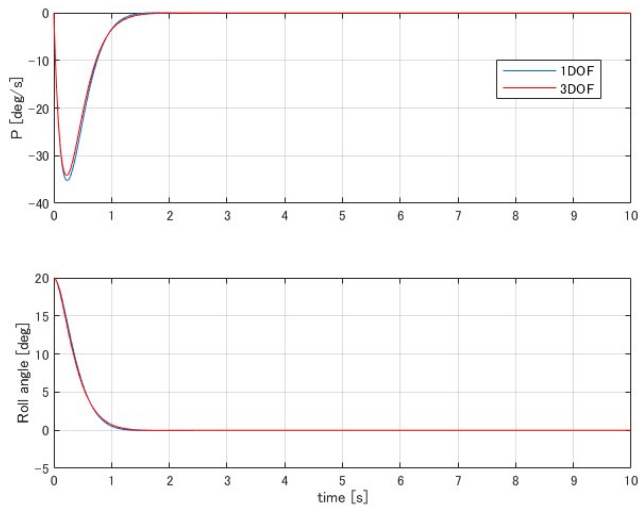


図 5.17 1 軸と 3 軸での比較(ロール軸)

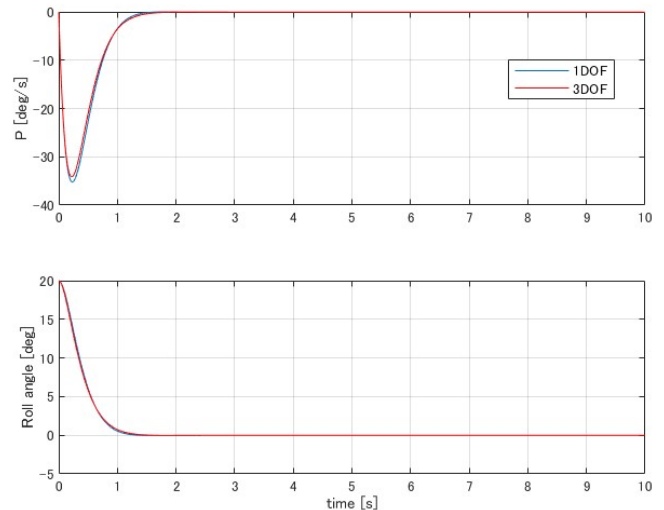


図 5.18 1 軸と 3 軸での比較(ピッチ軸)

図 5.13, 5.14 ではロール軸, ピッチ軸, 両軸とも与えられた初期値 20deg に対して 1 秒以内に収束しており, 良好な制御性能を得る事ができた。

図 5.15, 5.16 に関して, ロール軸, ピッチ軸に関する重み行列はそのまま, ヨー軸に関する重みだけを調整しながらシミュレーションを行った。ロール軸, ピッチ軸は両軸とも 1 自由度でシミュレーションした時と同様に 1 秒以内に収束している。ヨー軸に関してはロール軸, ピッチ軸の変動に伴い変化しているが, 1 秒以内に 0 に収束することができている。

図 5.17, 5.18 について, 1 自由度と 3 自由度でそれぞれ, シミュレーション結果を比較したが結果はほぼ一致していることからシミュレーションでは, 線形モデルで設計したコントローラは非線形モデルにも対応可能であると考えられる。

## 5.7 小型モデルを用いた 1 自由度実験

### 5.7.1 1 実験方法

前節のシミュレーションを用いて設計されたコントローラを用いて, 実機を用いて 1 自由度での実験を行った。実験方法は以下の図 5.19 に示すようなアルミフレームで作成された実験台に小型機の前方と後方をワイヤーで結束することで自由度を減らし 1 自由度での検証を行った。以下の実験ではロール軸のみの検証を行っており, プロペラ回転中に機体に結ばれたコードを引っ張り, 約 20 度機体を傾けた状態で手を離し, ロール角 0deg の目標値を与え機体の挙動を確認する。



図 5.19 小型モデルを用いたロール軸周りでの検証

### 5.7.2 実験結果

以下の図 5.20 はロール軸周りの実験結果, 図 5.21 はその入力である. ただし, 角速度 $P$ 及び各モータへの入力値には高周波ノイズが多く含まれていたため, ローパス処理を行ったデータを示している

図 5.20, 図 5.21 について, 与えられた初期値(20deg)に対して, 右側のプロペラ( $T_{fr}$ ,  $T_{rr}$ )を強く, 左側のプロペラ( $T_{rl}$ ,  $T_{rr}$ )を弱くすることで徐々に徐々に振動が減衰できていることが結果からわかる. また, 制御入力に関しても徐々に定常値に向かっていることから, プログラムやシステムそのものには問題が無いと考えられる. 振動の減衰は確認できた一方, 角速度 $P$ , ロール角 $\phi$ は目標値である 0deg 近辺でおよそ, +3~0.3 deg 程度の揺動が見られた. しかし, この揺動がきっかけで, 機体が不安定化することは無かったため, 機体の安定化には成功していると考えられる.

以下の図 5.22 にはシミュレーションと小型モデル機での結果の比較を示す.

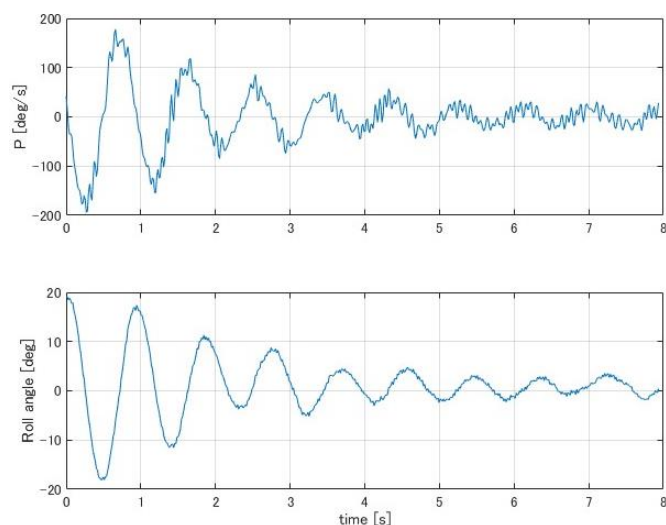


図 5.20 実験結果

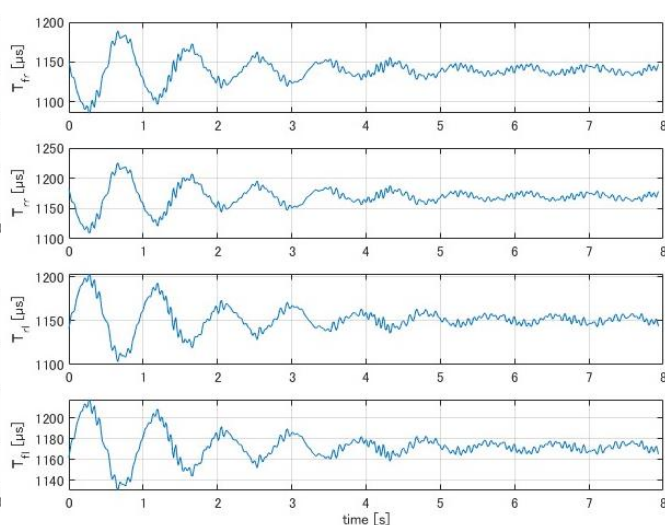


図 5.21 制御入力

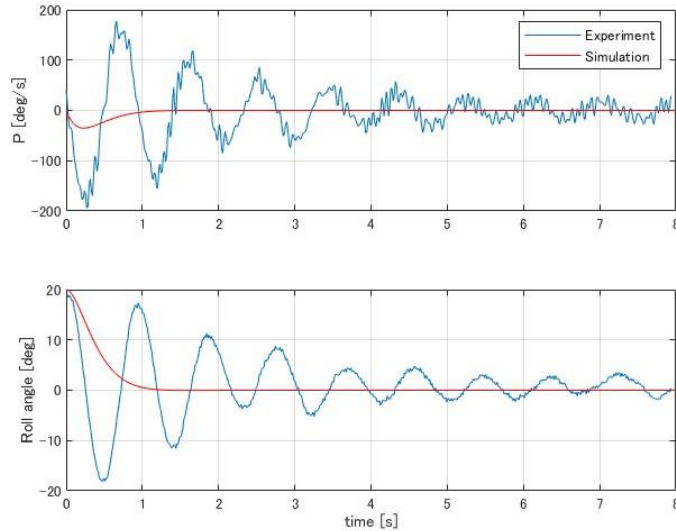


図 5.22 シミュレーションとの比較

図 5.22 より、シミュレーションと実機での実験では大きな差が出たことがわかる。角速度に関しては、20deg の初期値を減衰するために最大でも、-35deg/s 程度だが実験では最大-200deg/s 程度角速度が出ている。また、姿勢角についてもシミュレーションではおよそ 1s 程度で収束しているが、実験ではおよそ 3.5s 程度収束に要していることがわかる。このことから、現時点ではシミュレーションと実験の差が、主にシステム同定時におけるモデル化誤差の影響により大きくなってしまっており、シミュレーションベースで最適制御の重み行列を決定していくのは難しいと判断した。

そこで、ロール軸、ピッチ軸の重み行列をこの実験から試行錯誤的に決定する手法を用いて、次節の 6 自由度での実験に移る。

## 5.8 6 自由度飛行試験

### 5.8.1 実験方法

本節では前節で行った、1 軸での姿勢安定化実験を基にロール軸及びピッチ軸の重み行列を決定し小型モデルでの飛行試験を行う。

この実験ではヨー軸を除く回転運動にのみ制御を行っており、並進運動に関しては制御をしていない。浮上には手元のプロポからの信号を受け、徐々に平衡点を上昇させることで、全機推力を上昇させ浮上を試みた。

### 5.8.2 実験結果

以下の図 5.23 に角速度の実験結果、図 5.24 に姿勢角の実験結果、図 5.25 に制御入力の実験結果を示す。

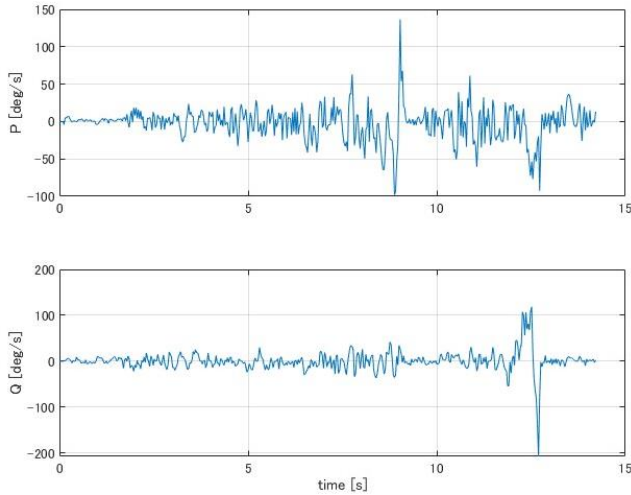


図 5.23 角速度応答の結果

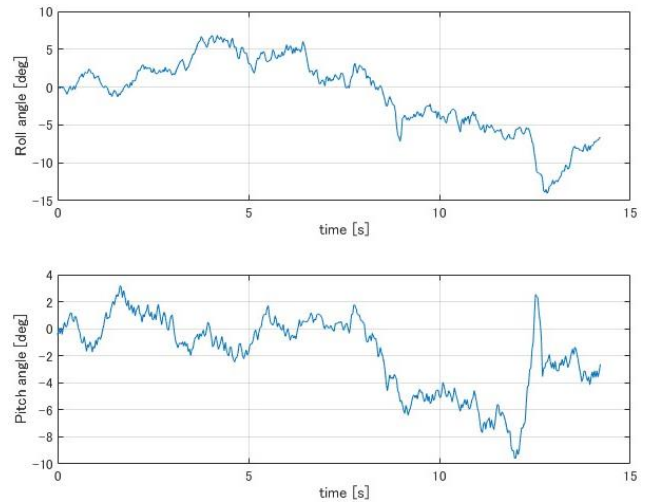


図 5.24 姿勢角応答の結果

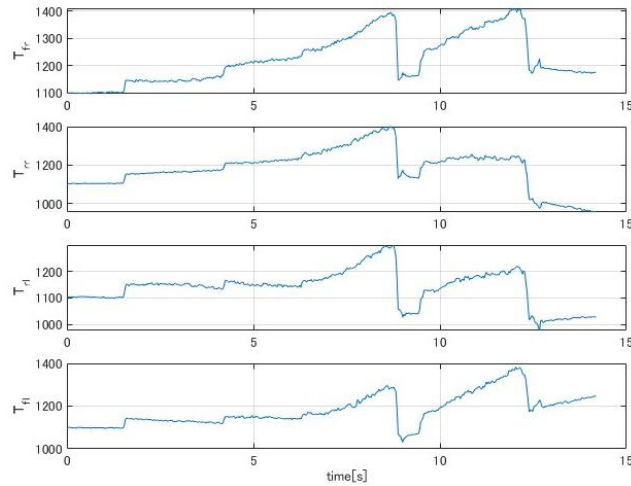


図 5.25 制御入力の結果

姿勢角応答の結果について、実験開始からおよそ数秒はロール角、ピッチ角共に 10 度以内に抑制することができているが、平衡点を上昇させていくにつれて、徐々に両軸とも姿勢角が大きくなり始め、-5deg 程度で一度姿勢角が収束していることがわかる。また、制御入力に関して、10s の時点でモータへの入力 が  $T_{fr} > T_{rr} \approx T_{fl} > T_{rl}$  の様に推力の差は出ているものの、姿勢角に反映されていない事がわかる。

### 5.8.3 考察

この実験では、1 自由度でそれぞれ設計したコントローラで試験を行ったが、地面から少し浮き上がることはできたものの、姿勢を安定化し空中でホバリングすることができなかった。この原因について考察する。

まず、システムの非線形性について考察する。シミュレーションでは 1 自由度でシミュレーションしたものと、3 自由度でシミュレーションしたものでは結果に大差は出なかった。しかし、実際には 1 自由度で設計したコントローラでは 3 自由度の安定化をすることができなかった。このことから、実際のシステムの非線形性を最適制御で抑える事は困難であると考えられる。



次に最適制御の適用について考察する。最適制御理論はシステムのパラメータから最適な制御ゲインを計算する理論でありことからシステム同定の時点で、モデル化誤差が出た場合については最適なゲインは求めることができない。今回のシステム同定については、特に慣性モーメントの測定の時点で大きな誤差が出たと考えられ、今後最適制御理論を適用する場合には、システム同定についてより理解と実験の精度を上げる必要があると考えた

モデル化方法について考察する。今回はシミュレーションを行う際のモデル化について、ニュートン法を用いてホワイトボックスモデリングを行い、それを基にシミュレーション及びコントローラ的设计を行った。しかし、シミュレーション上の挙動と実際の挙動にはかなり差があった事から、別のモデル化方法についても検討する必要があると考える。例としては、ホワイトボックスモデリングの場合は、モデルのエネルギーに着目したラグランジュ法や、各入力からインパルス入力を送ることで、得られる応答から伝達関数を求める方法なども挙げられる。

## 6. 実機の制御系設計

本章では 3 章での小型モデルを用いた最適制御検証の結果から、最適制御理論を用いた制御系ではなく、PID 制御をベースとした制御系を実機での試験を中心に設計を進めていく。

### 6.1 運動モデルの再検討

本節では先行研究でシミュレーション及び LQR 制御系の設計で用いた運動モデルについて再検討する。

#### 6.1.1 プロペラ後流による翼の揚力発生

先行研究では以下の図 6.1 の様にプロペラの回転により発生した後流によって生じる翼からの揚力発生について、その影響は小さく無視できるものとしてモデリングを行っていた。しかし、実験ではプロペラの回転速度を上昇させていくと、機体が図 6.2 の様に後退している様子が確認できた。このことから、プロペラ後流による翼の揚力は無視できないものと考え、本研究ではこの影響を加味しモデリングを行う。

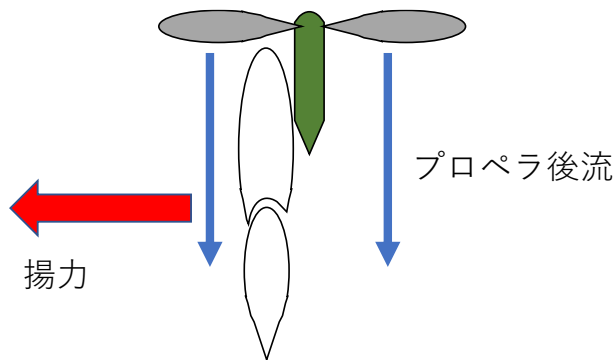


図 6.1 プロペラの後流による揚力発生

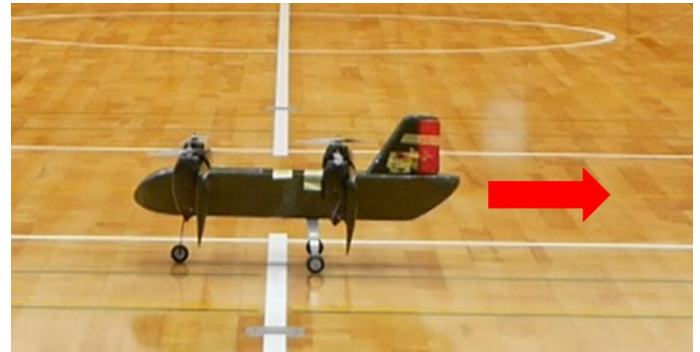


図 6.2 実験で見られた機体の後退

## 6.2 運動モデル

### 6.2.1 座標系

このモデルで用いている座標系について以下の図 6.3, 6.4 に示す。

それぞれ右手系で表現され、機体固定座標系  $X^B$  軸を機体前方方向に、各軸周りに右ねじ回りを正とする、角速度  $PQR$  を定義する。

同様に、地面に固定した座標系を慣性座標系とし、END 座標として定義する。また、各軸周りに Z-X-Y の回転順序のオイラー角を右ねじ回りを正として  $\phi\theta\psi$  を定義する。

図 6.4 の翼座標系について、機体に対して前側にティルトさせた場合を正とし、翼が地面と垂直の場合を  $\xi = 0$  として定義する

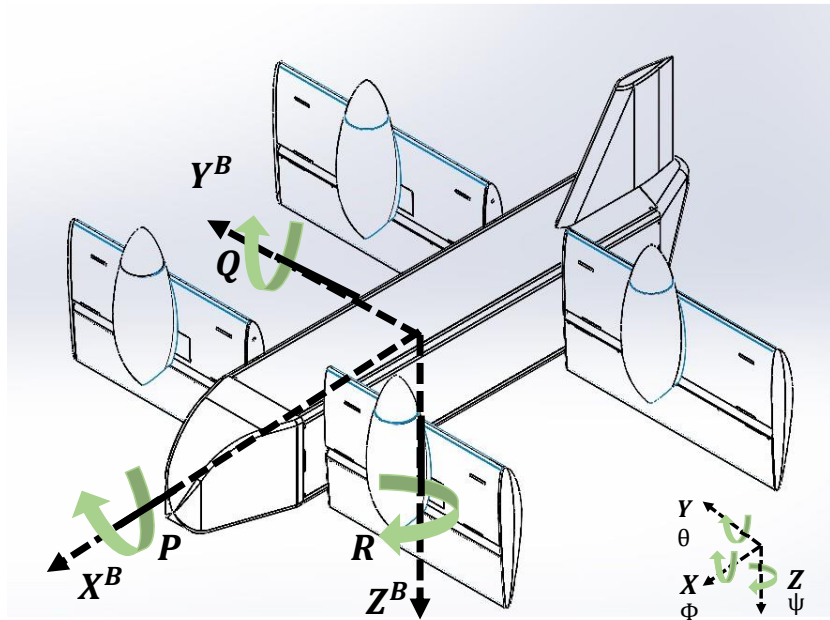


図 6.3 機体固定座標系 $X^B Y^B Z^B$ と地面固定座標系  $XYZ$

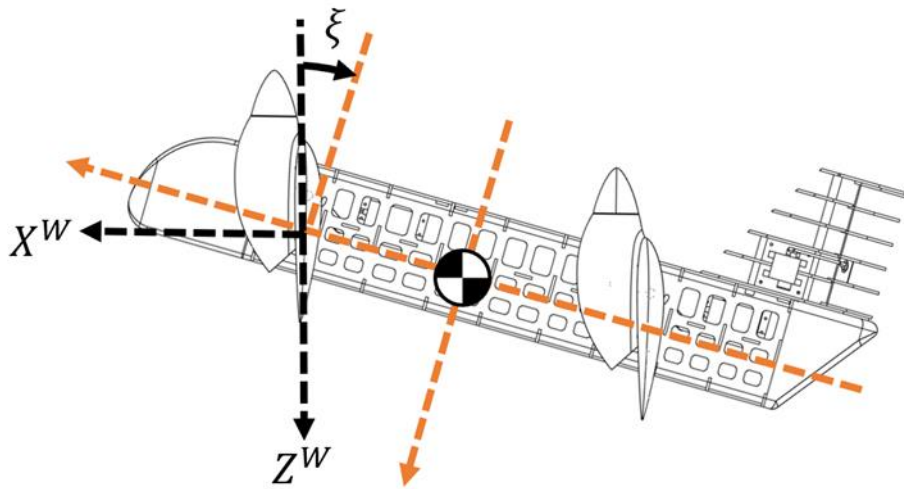


図 6.4 翼座標系 $X^W Z^W$

次にそれぞれの座標間の回転行列について、

機体座標系 $X^B Y^B Z^B$ から慣性座標系 $XYZ$ までの回転行列は

$$[R]^{EB} = \begin{bmatrix} \cos \theta \cos \psi - \sin \phi \sin \theta \sin \psi & -\cos \phi \sin \psi & \sin \theta \cos \psi + \sin \phi \cos \theta \sin \psi \\ \cos \theta \sin \psi + \sin \phi \sin \theta \cos \psi & \cos \phi \cos \psi & \sin \theta \sin \psi - \sin \phi \cos \theta \cos \psi \\ -\cos \phi \sin \theta & \sin \phi & \cos \phi \cos \theta \end{bmatrix} \quad (6.1)$$

翼座標系 $X^W Z^W$ から機体座標系 $X^B Y^B Z^B$ までの回転行列は

$$[R]^{BW} = \begin{bmatrix} \cos \xi & 0 & \sin \xi \\ 0 & 1 & 0 \\ -\sin \xi & 0 & \cos \xi \end{bmatrix} \quad (6.2)$$

また翼座標 $X^W Z^W$ から、機体座標 $X^B Y^B Z^B$ を経由して、慣性座標系 $XYZ$ までの回転行列は

$$[R]^{EW} = [R]^{EB} [R]^{BW} \quad (6.3)$$

よって

$$[R]^{EW} = \begin{bmatrix} \cos(\xi + \theta) \cos \psi - \sin \phi \sin(\xi + \theta) \sin \psi & -\cos \phi \sin \psi & \sin(\xi + \theta) \cos \psi + \sin \phi \cos(\xi + \theta) \sin \psi \\ \cos \theta(\xi + \theta) \sin \psi + \sin \phi \sin(\xi + \theta) \cos \psi & \cos \phi \cos \psi & \sin(\xi + \theta) \sin \psi - \sin \phi \cos(\xi + \theta) \cos \psi \\ -\cos \phi \sin(\xi + \theta) & \sin \phi & \cos \phi \cos(\xi + \theta) \end{bmatrix} \quad (6.4)$$

で表すことができる。

### 6.2.2 並進運動モデル

プロペラからの推力と翼からの揚力は機体固定座標系では以下のような力 $\mathbf{F}$ で表すことができる

$$\mathbf{F}_{Th} = \begin{bmatrix} 0 \\ 0 \\ -(T_{fr} + T_{fl} + T_{rr} + T_{rl}) \end{bmatrix} \quad (6.5)$$

$$\mathbf{F}_w = \begin{bmatrix} -(L_{fr} + L_{fl} + L_{rr} + L_{rl}) \\ 0 \\ 0 \end{bmatrix} \quad (6.6)$$

$$\mathbf{F} = [R]^{BW}(\mathbf{F}_{Th} + \mathbf{F}_w) \quad (6.7)$$

従って

$$\begin{aligned} \mathbf{F} &= \begin{bmatrix} -(T_{fr} + T_{fl} + T_{rr} + T_{rl}) \sin \xi - (L_{fr} + L_{fl} + L_{rr} + L_{rl}) \cos \xi \\ 0 \\ -(T_{fr} + T_{fl} + T_{rr} + T_{rl}) \cos \xi - (L_{fr} + L_{fl} + L_{rr} + L_{rl}) \sin \xi \end{bmatrix} \\ &= \begin{bmatrix} -T_{all} \sin \xi - L_{all} \cos \xi \\ 0 \\ -T_{all} \cos \xi - L_{all} \sin \xi \end{bmatrix} \end{aligned} \quad (6.8)$$

ただし $T_{fr} + T_{fl} + T_{rr} + T_{rl} = T_{all}$   $L_{fr} + L_{fl} + L_{rr} + L_{rl} = L_{all}$ と書き換えている

慣性座標系での並進の運動方程式は式(6.3)の回転行列と重力項 $\mathbf{G} = [0 \ 0 \ g]^T$ を用いて

$$m \frac{d^2}{dt^2} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = [R]^{EB} \mathbf{F} + \mathbf{G} \quad (6.9)$$

$$\frac{d^2}{dt^2} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} -\frac{1}{m} \{(c\psi s\theta + c\theta s\phi s\psi)(T_{all} c\xi - L_{all} s\xi) + (c\psi c\theta - s\phi s\theta s\psi)(T_{all} s\xi + L_{all} c\xi)\} \\ -\frac{1}{m} \{(c\theta s\psi + s\phi s\theta c\psi)(T_{all} s\xi + L_{all} c\xi) + (s\theta s\psi - s\phi c\theta c\psi)(T_{all} c\xi - L_{all} s\xi)\} \\ \frac{1}{m} (c\phi s\theta(T_{all} s\xi + L_{all} c\xi) - c\phi c\theta(T_{all} c\xi - L_{all} s\xi) + g) \end{bmatrix} \quad (6.10)$$

と表すことができる。ただし、 $\sin \theta \rightarrow s\theta$ や $\cos \theta \rightarrow c\theta$ のような書き換えを行っている。

### 6.2.3 回転運動モデル

機体の回転運動を生み出すモーメントの要素は、プロペラからの推力 $\mathbf{M}_{Th}$ 、プロペラの回転に伴う反トルク $\mathbf{M}_R$ 、そして機体の翼から生み出されるトルク $\mathbf{M}_w$ である。

プロペラの推力が生み出すモーメントは各プロペラの推力と重心からの距離から求めることができるので、

$$\begin{aligned}
\mathbf{M}_{Th} &= \begin{bmatrix} -T_{fr}l_{fy} \cos \xi + T_{fl}l_{fy} \cos \xi - T_{rr}l_{ry} \cos \xi + T_{rl}l_{ry} \cos \xi \\ T_{fr}l_{fx} \cos \xi + T_{fl}l_{fx} \cos \xi - T_{rr}l_{rx} \cos \xi - T_{rl}l_{rx} \cos \xi \\ T_{fr}l_{fy} \sin \xi - T_{fl}l_{fy} \sin \xi + T_{rr}l_{ry} \sin \xi - T_{rl}l_{ry} \sin \xi \end{bmatrix} \\
&= \begin{bmatrix} \{(-T_{fr} + T_{fl})l_{fy} + (-T_{rr} + T_{rl})l_{ry}\} \cos \xi \\ \{(T_{fr} + T_{fl})l_{fx} + (-T_{rr} - T_{rl})l_{rx}\} \cos \xi \\ \{(T_{fr} - T_{fl})l_{fy} + (T_{rr} - T_{rl})l_{ry}\} \sin \xi \end{bmatrix} \quad (6.11)
\end{aligned}$$

反トルクが生み出すモーメントを $\Omega_i (i = fr, fl, rr, rl)$ とすると

$$\mathbf{M}_R = \begin{bmatrix} (-\Omega_{fr} + \Omega_{fl} + \Omega_{rr} - \Omega_{rl}) \sin \xi \\ 0 \\ (-\Omega_{fr} + \Omega_{fl} + \Omega_{rr} - \Omega_{rl}) \cos \xi \end{bmatrix} \quad (6.12)$$

翼からの揚力が生み出すモーメントについて、先行研究<sup>(2)</sup>より、プロペラは翼の中心に位置しており、揚力はこの一点から生成されると仮定し、

$$\begin{aligned}
\mathbf{M}_w &= \begin{bmatrix} -L_{fr}l_{fy} \sin \xi + L_{fl}l_{fy} \sin \xi - L_{rr}l_{ry} \sin \xi + L_{rl}l_{ry} \sin \xi \\ L_{fr}l_{fx} \sin \xi + L_{fl}l_{fx} \sin \xi - L_{rr}l_{rx} \sin \xi - L_{rl}l_{rx} \sin \xi \\ -L_{fr}l_{fy} \cos \xi + L_{fl}l_{fy} \cos \xi - L_{rr}l_{ry} \cos \xi + L_{rl}l_{ry} \cos \xi \end{bmatrix} \\
&= \begin{bmatrix} \{(-L_{fr} + L_{fl})l_{fy} + (-L_{rr} + L_{rl})l_{ry}\} \sin \xi \\ \{(L_{fr} + L_{fl})l_{fx} + (-L_{rr} - L_{rl})l_{rx}\} \sin \xi \\ \{(-L_{fr} + L_{fl})l_{fy} + (-L_{rr} + L_{rl})l_{ry}\} \cos \xi \end{bmatrix} \quad (6.13)
\end{aligned}$$

の様に表すことができる。

よって全機のモーメント $\mathbf{M}$ は

$$\begin{aligned}
\mathbf{M} &= \mathbf{M}_{Th} + \mathbf{M}_R + \mathbf{M}_w \\
&= \begin{bmatrix} \{(-T_{fr} + T_{fl})l_{fy} + (-T_{rr} + T_{rl})l_{ry}\} c \xi + \{((-\Omega_{fr} + \Omega_{fl} + \Omega_{rr} - \Omega_{rl})) + (-L_{fr} + L_{fl})l_{fy} + (-L_{rr} + L_{rl})l_{ry}\} s \xi \\ \{(T_{fr} + T_{fl})l_{fx} + (-T_{rr} - T_{rl})l_{rx}\} c \xi + \{(L_{fr} + L_{fl})l_{fx} + (-L_{rr} - L_{rl})l_{rx}\} s \xi \\ \{(T_{fr} - T_{fl})l_{fy} + (T_{rr} - T_{rl})l_{ry}\} s \xi + \{((-\Omega_{fr} + \Omega_{fl} + \Omega_{rr} - \Omega_{rl})) + (-L_{fr} + L_{fl})l_{fy} + (-L_{rr} + L_{rl})l_{ry}\} c \xi \end{bmatrix} \quad (6.14)
\end{aligned}$$

と表すことができる。以下では上式は冗長なのでそれぞれ、

$$\mathbf{M} = [M_1 \quad M_2 \quad M_3]^T$$

と省略することにする。

オイラーの回転運動方程式より、

$$\begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \frac{d}{dt} \begin{bmatrix} P \\ Q \\ R \end{bmatrix} = - \left( \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix} \right)^T \times \begin{bmatrix} P \\ Q \\ R \end{bmatrix} + \mathbf{M} \quad (6.15)$$

$$\begin{bmatrix} \dot{P} \\ \dot{Q} \\ \dot{R} \end{bmatrix} = \begin{bmatrix} \frac{1}{I_{xx}} (- (I_{zz} - I_{yy}) QR) + M_1 \\ \frac{1}{I_{yy}} (- (I_{xx} - I_{zz}) RP) + M_2 \\ \frac{1}{I_{zz}} (- (I_{yy} - I_{xx}) PQ) + M_3 \end{bmatrix} \quad (6.16)$$

ただし、

$$M_1 = \{(-T_{fr} + T_{fl})l_{fy} + (-T_{rr} + T_{rl})l_{ry}\} c \xi + \{((-\Omega_{fr} + \Omega_{fl} + \Omega_{rr} - \Omega_{rl})) + (-L_{fr} + L_{fl})l_{fy} + (-L_{rr} + L_{rl})l_{ry}\} s \xi \quad (6.17-1)$$

$$M_2 = \{(T_{fr} + T_{fl})l_{fx} + (-T_{rr} - T_{rl})l_{rx}\} c \xi + \{(L_{fr} + L_{fl})l_{fx} + (-L_{rr} - L_{rl})l_{rx}\} s \xi \quad (6.17-2)$$

$$M_3 = \{(T_{fr} - T_{fl})l_{fy} + (T_{rr} - T_{rl})l_{ry}\} s \xi + \{((-\Omega_{fr} + \Omega_{fl} + \Omega_{rr} - \Omega_{rl})) + (-L_{fr} + L_{fl})l_{fy} + (-L_{rr} + L_{rl})l_{ry}\} c \xi \quad (6.17-3)$$

である

機体座標系の角速度 $PQR$ はオイラー角の時間微分であるので、

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ \sin \phi \sin \theta & 1 & -\frac{\sin \phi}{\cos \phi} \cos \theta \\ \cos \phi \sin \theta & 0 & \frac{\cos \theta}{\cos \phi} \\ -\frac{\sin \theta}{\cos \phi} & 0 & \frac{\cos \theta}{\cos \phi} \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix} \quad (6.18)$$

と表すことができる。

#### 6.2.4 プロペラ後流による揚力のモデリング

プロペラ後流による翼の揚力発生について、翼の揚力は風速に影響を受ける事から本来ならば、翼の回転速度から得られる後流の風速を用いてモデリングを行う事が好ましいが、本研究で用いる ESC では回転速度をフィードバックする機能を持ち合わせていないため、モータに入力する PWM 信号のデューティ比 $u$ と発生する揚力 $L$ の関係性を用いる。

以下の図 6.5 にはプロペラ後流による揚力モデリングの同定実験の様子を示す。図のように、6 軸力覚系に機体を固定し、ティルト角を $\xi = 0\text{deg}$  に設定し実験を行う。

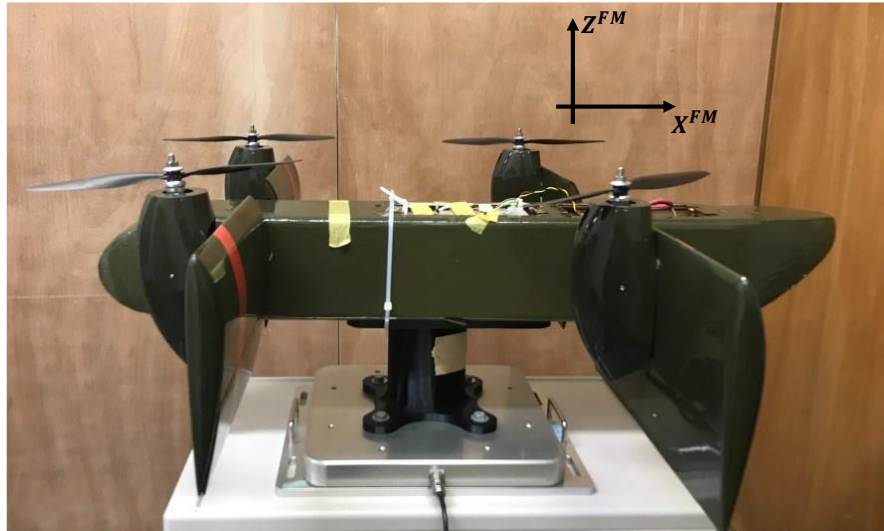


図 6.5 実験の様子と 6 軸力覚系座標 $X^{FM}Z^{FM}$

実験方法は全てのプロペラを同じデューティ比で駆動させ、図 6.5 の 6 軸力覚系座標の $X^{FM}$ 軸方向の力を測定し、プロペラ後流揚力 $L_{all}(= L_{fr} + L_{fl} + L_{rr} + L_{rl})$ を得る。

得られた揚力は 4 つの翼から得られた揚力の合計であるので、各翼ごとの揚力を計算する。翼から得られる揚力 $L$ は以下の式(6.19)様に、空気密度 $\rho$ 、流速 $v$ 、翼面積 $S$ 、揚力係数 $C_L$ の関数である。

$$L = \frac{1}{2} \rho v^2 S C_L \quad (6.19)$$

ここで、流速に関して各翼は同じデューティ比で駆動させているので各翼で流速は等しく、揚力係数に関して、各翼で翼型は等しく、同じようにプロペラが取り付けられていることから、揚力係数に関しても各翼で等しい。空気密度に関しても同様に各翼で等しいので、翼の揚力 $L$ は翼面積のみの関数と考えることができる。

本機体に搭載された翼について以下の表 6.1 にまとめる

表 6.1 機体搭載翼

	スパン [m]	翼弦長 [m]	翼面積 [m <sup>2</sup> ]
前翼	0.2	0.72	1.44
後翼	0.2	0.82	1.64

なお、左右の翼に関しては前後共に同じ形状であるので前翼から得られる揚力を $L_{front} = L_{fr} = L_{fl}$ 、後翼から得られる揚力を $L_{rear} = L_{rr} = L_{rl}$ とおくと以上の表より、

$$L_{front} : L_{rear} = 1.44 : 1.64 \quad (6.20)$$

すなわち

$$L_{front} = \frac{1.44}{1.66} L_{rear} \quad (6.21)$$

であるので、

$$L_{all} = 2L_{front} + 2L_{rear} \quad (6.22)$$

$$L_{all} = 2 \left( 1 + \frac{1.44}{1.64} \right) L_{rear} \quad (6.23)$$

$$L_{rear} = \frac{L_{all}}{2 \left( 1 + \frac{1.44}{1.64} \right)} = 0.26623 L_{all} \quad (6.24)$$

と求めることができる。(6.21)より $L_f$ は

$$L_{front} = 0.23376 L_{all} \quad (6.25)$$

で求めることができる。

以下の図 6.6, 6.7 にはそれぞれ前翼 1 枚から得られた揚力 $L_{front}$ と近似曲線、後翼 1 枚から得られた揚力 $L_{rear}$ と近似曲線を示す。

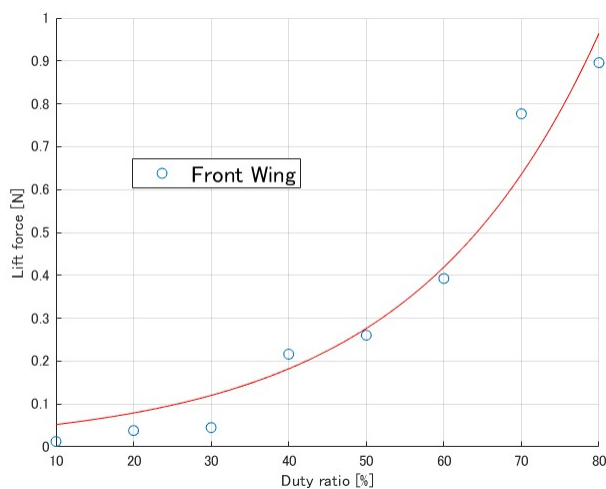


図 6.6 前翼から得られる揚力

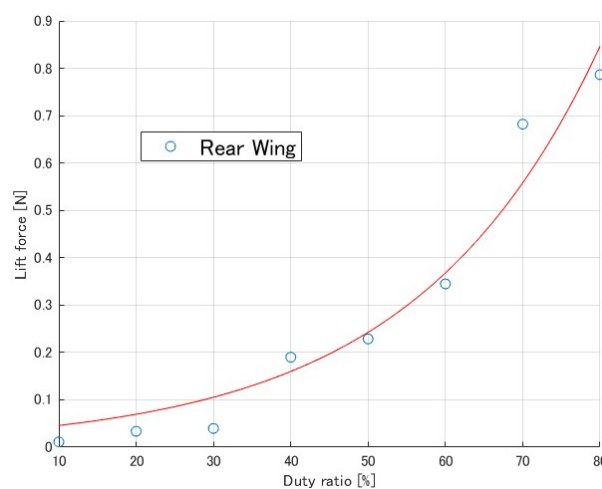


図 6.7 後翼から得られる揚力

それぞれの近似曲線は Matlab の Curve Fitting Toolbox を用いてフィッティングを行った。近似曲線は

$$L = a \exp(bu) \quad (6.26)$$

として近似している。以下の表 6.2 に得られた係数をそれぞれ示す

表 6.2 各翼の近似係数

	$a$	$b$
前翼	0.0344	0.04167
後翼	0.0302	0.04167

### 6.2.5 機体のモデリング

本項では先行研究で行われた、機体のシステム同定法に倣い、機体の重量 $m$ 、機体の重心位置から得られる各モーターまでのモーメントアームの長さ $l_{fx}, l_{fy}, l_{rx}, l_{ry}$ 、機体の慣性モーメント $I_{xx}, I_{yy}, I_{zz}$ 、モータの推力モデルの4点を行った。以下の表 6.3 にはそれぞれのシステム同定結果をまとめる。

表 6.3 システム同定の結果まとめ

機体重量 [kg]	$m$	4.66
モーメントアーム長 [m]	$l_{fx}$	0.3137
	$l_{fy}$	0.21
	$l_{rx}$	0.0863
	$l_{ry}$	0.24
慣性モーメント [kg m <sup>2</sup> ]	$I_{xx}$	0.3372
	$I_{yy}$	0.6079
	$I_{zz}$	0.9141
モータ	推力モデル	$T = 9.679 \times 10^{-9}u^2 + 2.86 \times 10^{-6}u + 0.0057$
	反トルクモデル	$\Omega = 1.504 \times 10^{-10}u^2 + 1.208 \times 10^{-6}u + 7.723 \times 10^{-4}$

モータモデルは本研究では最適制御を用いないことから、線形モデルではなく2次曲線モデルを用いて近似している。また、PWM周波数を16bit整数型で表現するために正規化している。

### 6.2.6 ティルト角の微小な揺動

前項の実験中にプロペラを30%以上のデューティ比で駆動させると以下の図 6.8 の様にティルト角が微小な変化を起こす事が実験中に判明した。

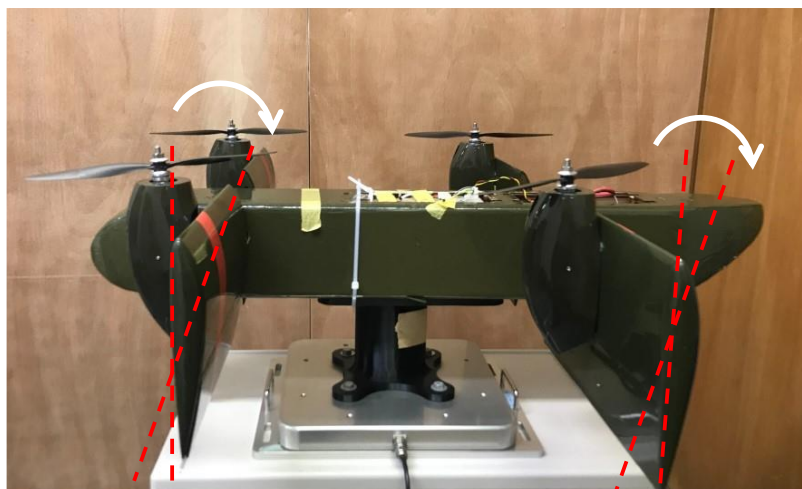


図 6.8 ティルト角の微小揺動



原因は2つあると考えられる。一つはティルト機構に用いられているギアのバックラッシュである。ティルト機構に用いられているギア及びサーボモータ内のバックラッシュにより、ティルト角が固定されていないことが原因の一つとして考えられる。この原因に加えて、翼の揚力によるピッチングモーメントがもう一つの原因であると考えられる。揚力が発生することにより、翼の頭上げ方向にモーメントが掛かり、ティルト角が固定されていないことも相まって結果的に翼が後ろ方向に回転してしまったと考えられる。

対策として、デューティー比毎に揺動する角度を計測し、その分だけティルト角を前側にオフセットするという手法を取った。以下の図 6.9 にはデューティー比とティルト角の変化量の関係を示す。

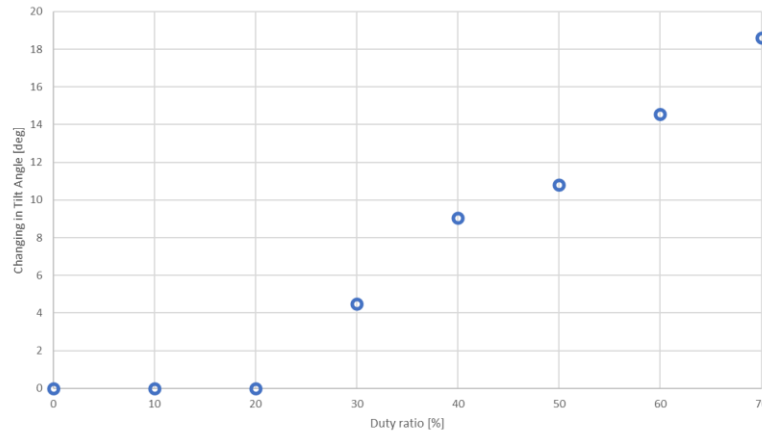


図 6.9 デューティー比とティルト角の揺動の関係

以上の結果からデューティー比が 20%以上の時のティルト角の揺動は線形に増加していることがわかる。

### 6.3 シミュレーション

#### 6.3.1 制御系

以下の図 6.10 には今回用いたカスケード型 PID コントローラを示す。

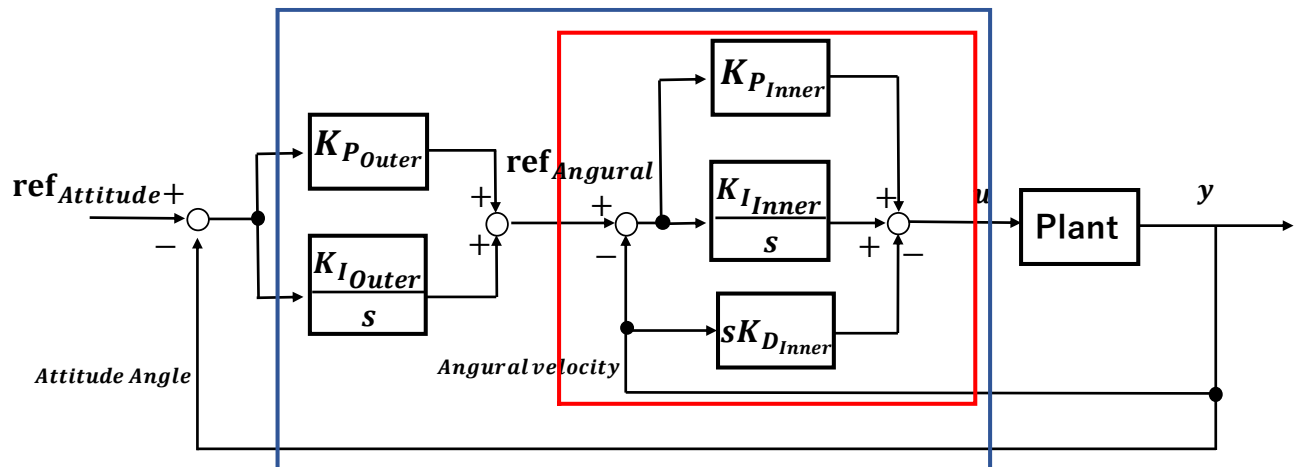


図 6.10 カスケード型 PID 制御のブロック線図

図 6.10 に示す制御系は青線で囲まれた、姿勢角 $\phi, \theta, \psi$ を FB しているアウターループと、赤線で囲まれたアウターループによって生成された角速度の目標値に対して機体角速度 $P, Q, R$ を FB しているインナー

ループ部の2つのFBシステムを掛け合わせた制御系である。インナーループのD制御の部分には、姿勢角目標値から算出される角速度目標値に対して微分によるキックが起きる事を防ぐため、微分先行型を採用している。

### 6.3.2 シミュレーション結果

以下の図 6.10, 6.11 にはそれぞれ角速度 $P, Q, R$ のシミュレーション結果, オイラー角 $\phi, \theta, \psi$ のシミュレーション結果を示す。

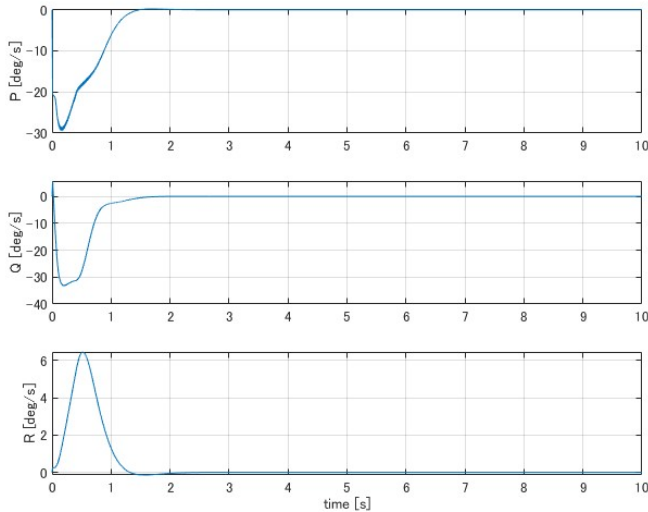


図 6.10 角速度のシミュレーション結果

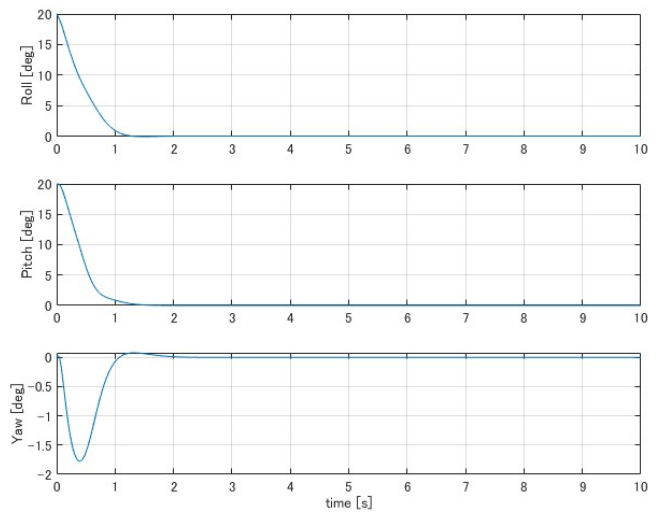


図 6.11 オイラー角のシミュレーション結果

以上のシミュレーションはロール角, ピッチ角にそれぞれ 20deg の初期値を与え, 減衰する様子を示したものである。制御系の設計方針として, 姿勢角が 1 秒以内に 80%以上収束し, オーバーシュートしないことを目標に制御系を設計した。結果より, 各軸とも制御目標を達していることが結果からわかる。

## 6.4 飛行試験

シミュレーションを用いて設計されたコントローラを基準にゲインを調節しながら実験を行った。結論としては, 機体の安定した離陸を成功させることはできず, 有用なデータを取ることはできなかった。ここでは, その実験の結果をいくつか抜粋し, その結果から考察を行う。

### 6.4.1 実験 1

この実験の結果を以下の図 6.12, 6.13, 6.14 それぞれ, 機体姿勢角データ, 角速度データ, 制御入力データのログを示す。

この実験では角速度データに注目すると, ロール軸, ピッチ軸, ヨー軸に全ての軸周りの回転に対して減衰できていることがわかる。しかし, 発散はしていないものの 0 に収束する前にオーバーシュートが見られる。

次にオイラー角に注目する。実験開始直後は比較的機体が安定していることがデータからわかるが, 推力を上昇させていくと徐々に機体が振動的になり, 特にロール角ピッチ角が連成して現れる場合にはロール角, ピッチ角共に大きな値が見て取れる。また, 実験終了時の姿勢角を見ると 0 度から少しオフセットしていることから, 実験後は毎回オフセットが必要であると考えられる。

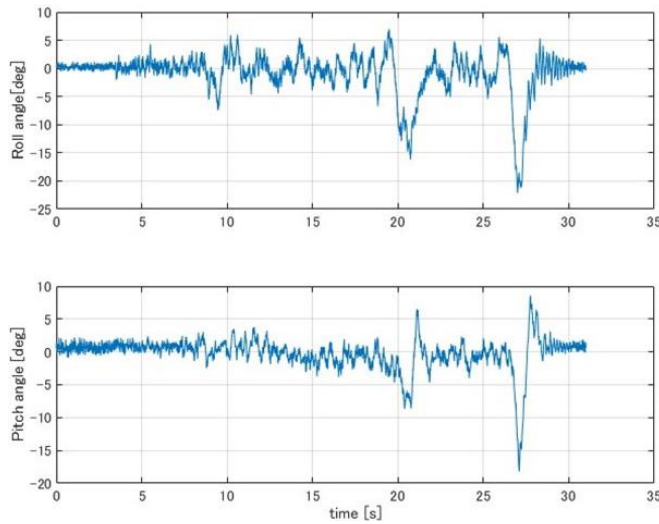


図 6.12 姿勢角

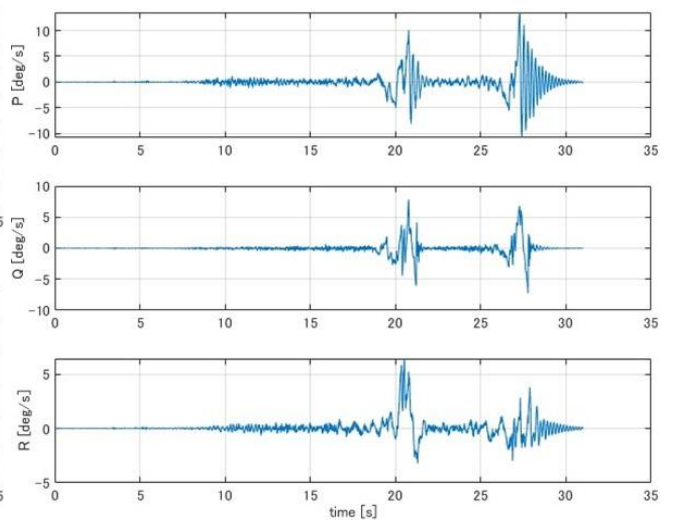


図 6.13 角速度

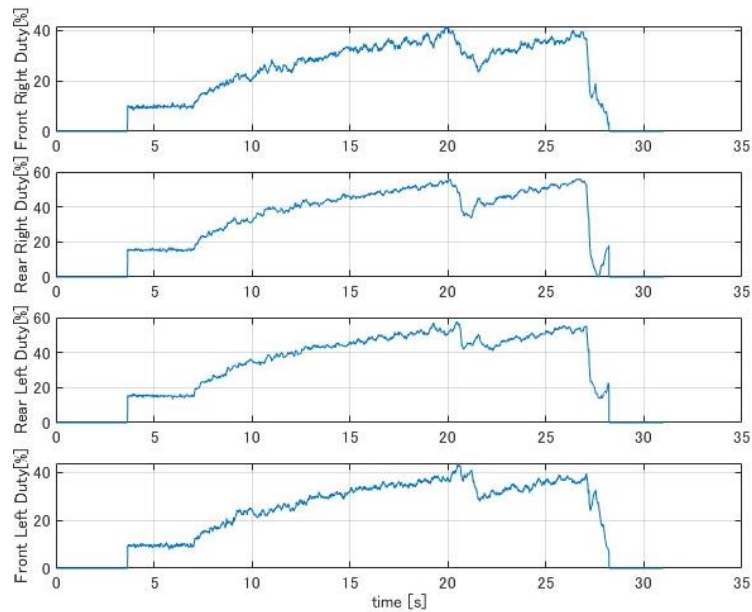


図 6.14 制御入力

### 6.4.2 実験 2

以下の図 6.15, 6.16, 6.17 にもそれぞれ姿勢角, 角速度, 制御入力の実験結果を示す.

この実験でも同様に, 推力を徐々に大きくしていくにつれて, 機体が不安定になる傾向があった. また, オイラー角が 10 度以上になってしまった場合には機体が大きく乱れ, その場合に機体の姿勢角を戻すことができずに, そのまま離陸せずに流れて行ってしまったといった現象が起きていた.

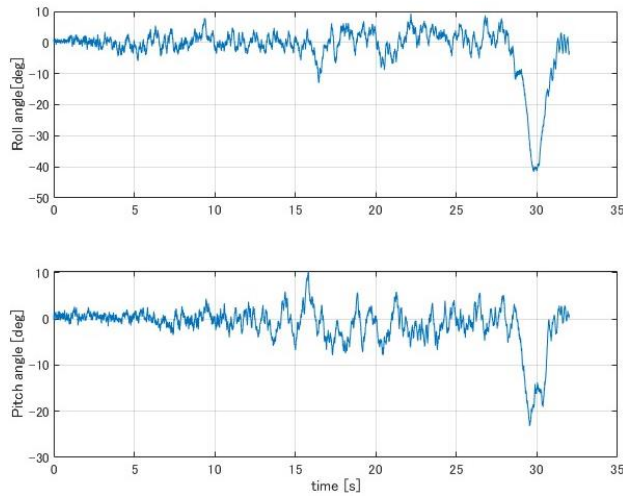


図 6.15 姿勢角

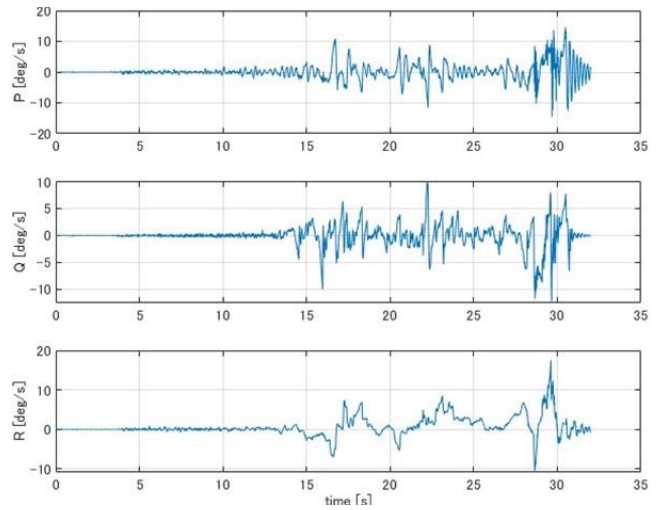


図 6.16 角速度

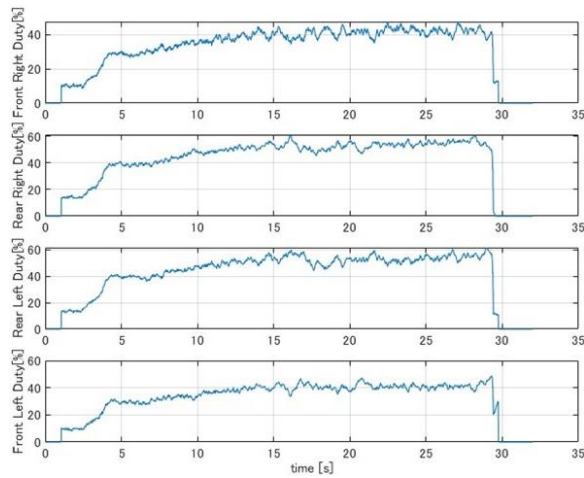


図 6.17 制御入力

### 6.4.3 飛行試験の考察

先ず、2つの試験に共通する結果として、機体のオイラー角が10度を超えると、機体が制御できなくなるという現象が起きた。以下の図 6.18 に青線にロール角速度、ピッチ角速度のセンサ値、赤線はアウトーループによって算出された角速度の目標値を示す。

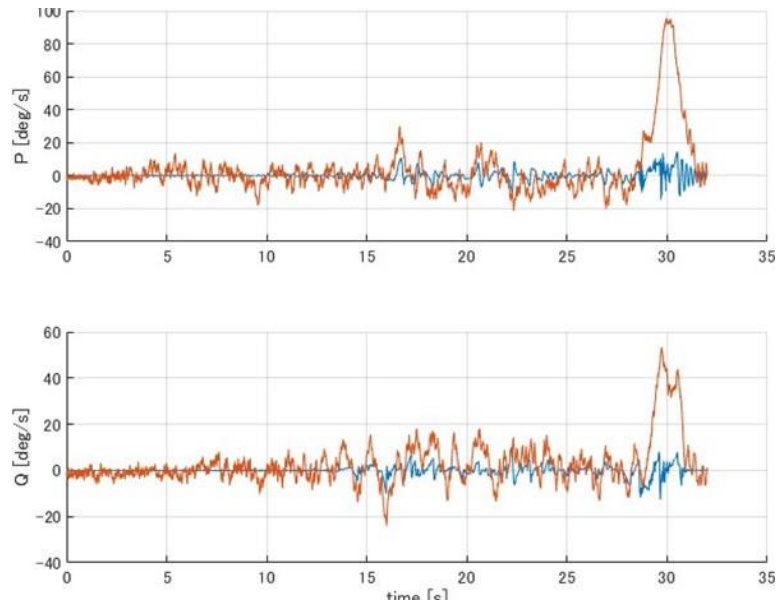


図 6.18 角速度目標値と実験値

この図から、機体の角速度指令値が 0.5 を超えると機体角速度が追従できなくなっていることがわかる。これは、インナーループのゲイン不足かコードにミスがあるかの原因が考えられる。

次に、姿勢推定について、今回の実験では Madgwick フィルタを用いて姿勢推定を行っており、フィルタ内のゲインを姿勢角の位相遅れの抑制に重きを置き 0.8 で実験を行った。しかし、この状態ではプロペラが全く回転していない状態でも、姿勢角にノイズが乗っていることから、もう少しゲインを下げる必要があると考えられる。

また、今回の実験では地磁気センサーを使用しておらず、ヨー軸方向の制御は角速度のみ行っており、ヨー軸の姿勢角制御は行っていなかったが、プロペラの回転速度を上昇させていくと、機体が回転してしまっていたことから、地磁気によるヨー軸の制御も行う必要があると考える。

ヨー軸の制御については本研究ではフラップの動作に問題があった事から、フラップを入力としたプロペラ後流による揚力の制御については行わず、プロペラの反トルクのみでヨー軸の制御を行った。プロペラの反トルクを用いると、必然的に前後左右の推力のバランスが崩れる事から今後はフラップを使ったヨー軸制御についても検討していくべきである。

次に高さ方向の制御について、今回の実験では推力を操縦者手元のリモコンを操作し、平衡点を上昇させることで離陸を試みた。しかし、平衡点を高めていくと少しの手元の狂いで、大きく推力が変わってしまうので今後は気圧センサを導入し高さ方向のセンシングを行う事で手元のリモコンからは高さの目標値設定が行えるようにするべきだと考える。

## 7. 結論

本稿では主に

- ・先行研究で開発されたシステムの改良
- ・深度情報を用いた斜面の傾斜推定及び斜面との相対的な回転角の推定
- ・小型モデルを用いた最適制御の検証
- ・実機の制御系設計

以上の4テーマについて述べた。

深度情報を用いた斜面推定では、深度カメラを用いて斜面との距離を計測することで斜面の傾斜及び相対的な斜面との回転角を推定できることを幾何的に示した後、実装し実験を行った。傾斜が1方向のみの簡単な斜面では比較的高精度で、それぞれ推定出来ていたが、2方向に傾斜がある場合には誤差が生じるなど、現実の斜面での推定にはやや課題が残る結果となった。

小型モデルを用いた最適制御検証では、シミュレーションと1自由度での実験、1自由度での実験と3自由度での実験をそれぞれ比較することによって、実機の制御系を設計する場合の方針を立てた。

実機の制御系設計について、前年度試験では影響がないと判断し、無視していたプロペラの後流による揚力発生についてモデリングを行い、実機での試験をベースにP-PID制御器を用いて実機の飛行試験を行った。しかし、機体は離陸することができず、有用なデータを取ることができなかった。今後の方針としては、実機試験で用いられたプログラムについて再度見直しを行い、飛行試験を継続し離陸、飛行中の斜面推定、斜面への着陸といった流れで、研究を進めていきたい。

## 謝辞

本研究を行うにあたり、多くの助言・サポートを頂いた指導教官の岡宏一教授と研究室の友人達に深く感謝申し上げます。ありがとうございました。

## 参考文献

- (1) 丹羽拓巳, “4 発ティルトウイング機の傾斜地に対する着陸の検討”, 高知工科大学附属情報図書館, 2022 年度修士論文, 2023
- (2) 三田侑弥, “4 発ティルト翼 VTOL 機のシステム同定と制御則設計”, 高知工科大学附属情報図書館, 2020 年度修士論文, 2021
- (3) Sebastian Madgwick, “An efficient orientation filter for inertial and inertial/magnetic sensor arrays” Report x-io and University of Bristol(UK), Volume 25, 2010
- (4) Ivan Dryanovski, “imu\_filter\_madgwick”, ROS Wiki, [http://wiki.ros.org/imu\\_filter\\_madgwick](http://wiki.ros.org/imu_filter_madgwick), (参照日 2024/2/1)
- (5) 友納正裕 原祥堯, “SLAM の現状と今後の展望”, システム/制御/情報 Vol. 64, No.2 pp.45-50, 2020
- (6) Intel®, “Intel®RealSense™ Product Family D400 Series Datasheet”, September 2023 (参照日 2024/2/10)
- (7) SamerKhshiboun , “ Intel(R) RealSense(TM) ROS Wrapper for Depth Camera ”, Github , <https://github.com/IntelRealSense/realsense-ros>, (参照日 2024/2/10)
- (8) 池田 雅夫, 須田 信英, “積分型最適サーボ系の構成”, 計測自動制御学会論文集, 1988, 24 巻, 1 号,p. 40-46
- (9) Arduino Foundation, “Arduino – Home”, Arduino, <https://www.arduino.cc/>, (参照日 2024/2/1)