

クアッドコプターの飛行安定化制御システムの開発

2015SC018 林美咲 2015SC061 宮野峻
2015SC072 西田裕貴 2015SC109 米川翔太
指導教員：中島明

1 はじめに

近年ドローンの市場規模は急激に拡大しており今後ドローンの活躍をする場は広がっていくと考えられている。

本研究では姿勢角制御器及び高度制御器を作成し、目標姿勢及び目標高度に追従させることを目的として制御を行う。本稿の構成としてまずドローンのモデリングを行う際に必要な物理量、座標の定義を行い、その定義に基づき運動方程式の導出を行う。次に制御器の設計と説明を行い、その制御器を用いたシミュレーション結果を示す。さらにドローンの慣性モーメントのパラメータ推定の方法と結果を説明する。そして、ドローンの制御を行う制御基板の概要及び利用方法を紹介し、現在搭載している制御基板の説明を行う。その後実験の結果を示し、最後に本論文の結論を述べる。

2 ドローンのモデリング

2.1 ドローンの座標系とパラメータ

3次元空間にあるドローンの空間表現を行うためには位置と姿勢角が必要である。これらの状態量を表現するためには、2つの直交座標系の定義を行う必要がある。1つ目が基準となる直交座標系である基準座標系 (Σ_r)、2つ目がドローンに固定された機体座標系 (Σ_b) である。これら2つの座標系はともに右手座標系である。また文字の左上添え字は基準となる座標系、右下添え字は表現される座標系を示しており、 b は機体座標系、 w は基準座標系を示す。以下の表1にドローンの状態パラメータを表し、図1はドローンの座標系 (Σ_r)、(Σ_b) の関係図とドローン機体における状態パラメータを図示したものである。ここで U_f を次のように定義をする。

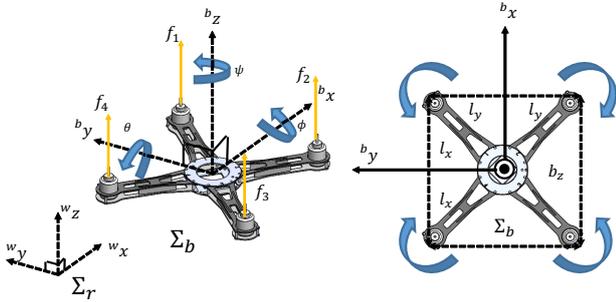


図1 ドローンの座標系

$$U_f = \sum_{i=1}^4 f_i \quad (1)$$

表1 ドローンのモデリングに関するパラメータの定義

記号	名称及び単位
m_b	機体の質量 [kg]
x	機体の x 方向への位置座標 [m]
y	機体の y 方向への位置座標 [m]
z	機体の z 方向への位置座標 [m]
ϕ	機体の姿勢角 (roll 角)[rad] (x 軸周り回転)
θ	機体の姿勢角 (pitch 角)[rad] (y 軸周り回転)
ψ	機体の姿勢角 (yaw 角)[rad] (z 軸周り回転)
J_{xx}	機体の x 軸慣性モーメント [kgm ²]
J_{yy}	機体の y 軸慣性モーメント [kgm ²]
J_{zz}	機体の z 軸慣性モーメント [kgm ²]
l_x	ロータと y 軸間の距離 [m]
l_y	ロータと x 軸間の距離 [m]
f_i	ロータ i 番の推力 [N]

2.2 ドローンモデルの導出

一般化座標 q は、基準座標系から見たドローンの位置、ドローンの姿勢角であり、6次元の縦ベクトルで表すことができる。ドローンの位置ベクトルを ${}^w P_b = [x, y, z]^T$ とし、ドローンの姿勢角を $\eta = [\phi, \theta, \psi]^T$ とすると

$$q = [{}^w P_b^T \quad \eta^T]^T \in \mathbb{R}^6$$

が得られる。次に ω を機体座標系での回転速度ベクトルとする。回転速度ベクトル ω と姿勢角の時間微分 $\dot{\eta}$ との関係は以下のように表される。

$$\omega = T\dot{\eta}, \quad T = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi \cos\theta \\ 0 & -\sin\phi & \cos\phi \cos\theta \end{bmatrix} \quad (2)$$

また、ラグランジュ関数 L_b は運動エネルギー $T_b(q, \dot{q})$ とポテンシャルエネルギー $U_b(q)$ から以下のように表される。

$$L_b(q, \dot{q}) = T_b(q, \dot{q}) - U_b(q) \quad (3)$$

$$T_b(q, \dot{q}) = \frac{1}{2}(m_b {}^w \dot{P}_b^T {}^w \dot{P}_b + \omega^T {}^b J_b \omega) \quad (4)$$

$$U_b(q) = m_b g {}^w e_z^T {}^w P_b \quad (5)$$

ただし、 $e_z = [0, 0, 1]^T$ は各座標系での z 軸方向の単位ベクトルである。また、 ${}^b J_b = \text{diag}(J_{xx}, J_{yy}, J_{zz})$ である。

次に、機体座標系の各軸周りに生じるモーメント τ_x, τ_y, τ_z について説明する。各軸周りのモーメントはアームの長さ l とロータの生み出す推力の積で書き表せるので以下のよう記述出来る。

$$\begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = B_b u, B_b = \begin{bmatrix} l_y & -l_y & -l_y & l_y \\ -l_x & -l_x & l_x & l_x \\ 1 & -1 & 1 & -1 \end{bmatrix}, u = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} \quad (6)$$

次に一般化力 F について考える。一般化力は並進運動では力、回転運動ではモーメントである。以上から一般化力 F は以下のように表せる。

$$F = B_f u, B_f = \begin{bmatrix} {}^w R_b^b e_z c^T \\ T^T B_b \end{bmatrix}, c = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

ここで ${}^w R_b$ は ZYX オイラー角を用いた機体座標系から基準座標系への回転行列である。

2.3 ラグランジュの運動方程式から状態方程式の導出

ラグランジュの運動方程式は (3) 式を用いて次の (7) 式となる。

$$\frac{d}{dt} \left(\frac{\partial L_b(q, \dot{q})}{\partial \dot{q}} \right) - \frac{\partial L_b(q, \dot{q})}{\partial q} = B_f u \quad (7)$$

ここで慣性行列 $M(q)$ と $N(q, \dot{q})$ を用いて定義した $N(q, \dot{q}) = \frac{d}{dt}(M(q)\dot{q}) - \frac{\partial T_b(q, \dot{q})}{\partial q} + \frac{\partial U_b(q)}{\partial q}$ から (7) 式は

$$M(q)\ddot{q} + N(q, \dot{q}) = B_f u \quad (8)$$

と変形できる。状態変数 x を $x = [q^T, \dot{q}^T]^T$ として非線形状態方程式

$$\dot{x} = f(x) + g(x)u \quad (9)$$

が得られる。ただし、

$$f(x) = \begin{bmatrix} \dot{q} \\ -M^{-1}(q)(N(q, \dot{q})) \end{bmatrix}, g(x) = \begin{bmatrix} O_{7 \times 4} \\ M^{-1}(q)B_f \end{bmatrix} \quad (10)$$

である。[1][2]

3 制御システム

3.1 roll 角, pitch 角の角度の P 制御と角速度の PID 制御と yaw 角の角度の PID 制御

3.1.1 カスケード型制御について

カスケード型制御 [3] は以下の図 2 のような多重ループ構造になっている。

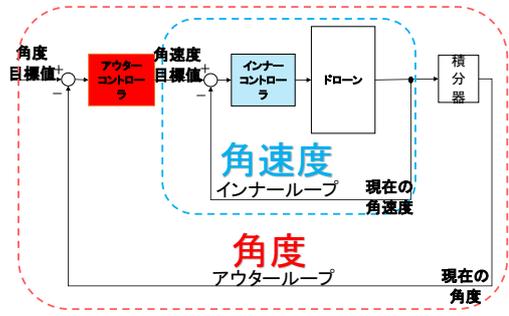


図 2 カスケード制御の概要

制御対象の一部であるドローンに対してインナーループでフィードバック制御を施しドローンの出力値である角速度と角速度目標値の次元を揃えた上で、積分器を含む制御対象の残りの角度に対してアウターループでフィードバック制御を行う。このようにカスケード型制御は制御対象が複雑で高次元であっても特性ごとに制御が可能である。

3.1.2 ドローンに対するカスケード型 P-PID 制御

この節ではドローンの姿勢角に対しカスケード型の P-PID 制御を行った。roll 角 ϕ , pitch 角 θ に関しては角度の P 制御と角速度の PID 制御器を用いた。yaw 角 ψ に関しては角度の PID 制御を用いた。この制御器を作成した理由としては姿勢角の PID 制御と比較して、角速度の制御を加えることで安定性が向上する事が挙げられる。今後この制御器はカスケード P-PID 制御と呼称する。以下の図 3 にブロック線図を示す。ここで $\phi_d, \theta_d, \psi_d, z_d, \dot{\phi}_d, \dot{\theta}_d, \dot{\psi}_d, \dot{z}_d$ はセンサから取得できる角度, 高度, 角速度, z 方向の速度であり, $\phi_{ref}, \theta_{ref}, \psi_{ref}, z_{ref}$ は各状態量の目標値である。

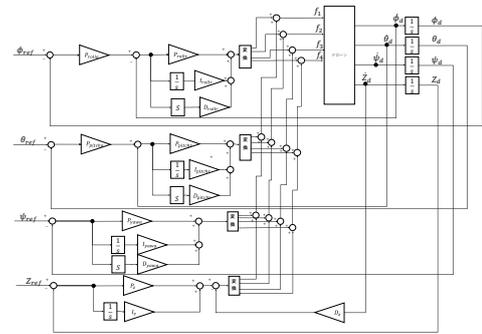


図 3 カスケード P-PID 制御のブロック線図

またドローンのカスケード P-PID 制御に用いたパラメータを以下の表 2 に示す。

3.2 姿勢角制御の詳細

カスケード P-PID 制御を用いた結果, 入力 (11) 式となる。ただし $e_\phi = \phi_{ref} - \phi_d$, $e_\theta = \theta_{ref} - \theta_d$, $e_\psi = \psi_{ref} - \psi_d$, $e_{\omega\phi} = P_{roll}e_\phi - \dot{\phi}_d$, $e_{\omega\theta} = P_{pitch}e_\theta - \dot{\theta}_d$ である。

表 2 ドローンのカスケード P-PID 制御に関するパラメータ

記号	名称	ゲイン値
P_{rolla}	roll 角度制御の P ゲイン	10
P_{pitcha}	pitch 角度制御の P ゲイン	10
P_{yawa}	yaw 角度制御の P ゲイン	1
P_z	高度制御の P ゲイン	100
P_{rollr}	roll 角速度制御の P ゲイン	0.5
P_{pitchr}	pitch 角速度制御の P ゲイン	0.5
I_{rollr}	roll 角速度制御の I ゲイン	0.1
I_{pitchr}	pitch 角速度制御の I ゲイン	0.1
I_{yawa}	yaw 角度制御の I ゲイン	0.002
I_z	高度制御の I ゲイン	5.2
D_{rollr}	roll 角速度制御の D ゲイン	0.001
D_{pitchr}	pitch 角速度制御の D ゲイン	0.001
D_{yawa}	yaw 角度制御の D ゲイン	1
D_z	高度制御の D ゲイン	5

$$\begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = \begin{bmatrix} P_{rolla}e_\phi + P_{rollr}e_{\omega\phi} + I_{rollr} \int_0^t e_{\omega\phi} dt + D_{rollr} \dot{e}_{\omega\phi} \\ P_{pitcha}e_\theta + P_{pitchr}e_{\omega\theta} + I_{pitchr} \int_0^t e_{\omega\theta} dt + D_{pitchr} \dot{e}_{\omega\theta} \\ P_{yawa}e_\psi + I_{yawa} \int_0^t e_\psi dt + D_{yawa} \dot{e}_\psi \end{bmatrix} \quad (11)$$

ここで roll 角のカスケード P-PID により生じたトルク τ_x は,

$$\tau_x = L_x \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix}, L_x = [l_y \quad -l_y \quad -l_y \quad l_y] \quad (12)$$

となる。よって roll 方向に回転させるために必要なトルクから各ロータの推力への変換は

$$\begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} = L_x^T (L_x L_x^T)^{-1} \tau_x \quad (13)$$

$$\begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} = \begin{bmatrix} \frac{1}{4l_y} \\ -\frac{1}{4l_y} \\ -\frac{1}{4l_y} \\ \frac{1}{4l_y} \end{bmatrix} \tau_x \quad (14)$$

となる。 τ_y, τ_z に関するモーメントから各ロータへの推力の変換は省略する。

3.2.1 高度制御の詳細

高度制御に関しても、微分先行型 PID 制御を用いて制御器の設計を行う。高度に関する運動は、回転運動には影響を及ぼさず、 z 方向の並進運動の入力を U_f とすると z 方向の入力は

$$U_f = m_b g + P_z(z_{ref} - z_d) + I_z \int_0^t (z_{ref} - z_d) dt - D_z \dot{z}_d \quad (15)$$

となる。ここで高度の PI 制御及び速度のフィードバックにより生じた推力 U_f は (1) 式から、

$$U_f = [1 \quad 1 \quad 1 \quad 1] \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} \quad (16)$$

となる。したがって高度制御のために必要な各ロータの推力への分配を行う変換は以下ようになる。

$$\begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} = \begin{bmatrix} \frac{1}{4} \\ \frac{1}{4} \\ \frac{1}{4} \\ \frac{1}{4} \end{bmatrix} U_f \quad (17)$$

3.3 目標姿勢及び目標高度追従のための積分型最適サーボ系制御器設計

3.1 節では古典制御の代表的なものである PID 制御を用いてドローンの制御を行ってきたが、評価関数によるコントローラの評価を行うことはできない。最適制御の分野からの制御器の設計も目指し最適サーボ系による制御器の作成手法を示す。

3.3.1 積分型最適サーボ系 [4]

ドローンの roll 角, pitch 角に関する回転運動の状態方程式, 出力方程式を以下の (18) 式として定義をする。

$$\begin{cases} \dot{\alpha}(t) = A\alpha(t) + Bu(t) \\ y(t) = C\alpha(t) \end{cases} \quad (18)$$

ただし, $\alpha = [\phi, \theta, \dot{\phi}, \dot{\theta}]^T$, $A \in \mathbb{R}^{4 \times 4}$, $B \in \mathbb{R}^{4 \times 4}$, $C \in \mathbb{R}^{2 \times 4}$ である。また, A, B は roll 角 ϕ , pitch 角 θ に関する状態量を平衡点近傍で線形化し, 導出されたものである。

出力 $y(t)$ と目標値 $y_{ref}(t)$ の偏差を $e(t)$ とすると

$$e(t) = y_{ref}(t) - y(t) \quad (19)$$

となる。また偏差 $e(t)$ を 0[s] から t [s] の区間で積分したものを $\xi(t)$ とする。

ドローンの roll 角, pitch 角に関する拡大偏差システムは状態変数を $\alpha_e = [\tilde{\alpha}, \tilde{\xi}]^T$ とすると

$$\begin{cases} \dot{\alpha}_e(t) = A_e \alpha_e(t) + B_e \tilde{u}(t) \\ e(t) = C_e \alpha_e(t) \end{cases} \quad (20)$$

と表される。ただし、

$$A_e = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix}, B_e = \begin{bmatrix} B \\ 0 \end{bmatrix}, C_e = [-C \quad 0]$$

$$\tilde{\alpha} = \alpha - \alpha_\infty, \quad \tilde{\xi} = \xi - \xi_\infty, \quad \tilde{u} = u - u_\infty$$

である。また $\alpha_\infty, \xi_\infty, u_\infty$ はそれぞれ定常値である。

次に (20) 式の拡大系における評価関数は

$$J_{servo} = \int_0^\infty \alpha_e^T \begin{bmatrix} C^T Q_{a1} C & 0 \\ 0 & Q_{a2} \end{bmatrix} \alpha_e + \tilde{u}^T R_s \tilde{u} dt \quad (21)$$

と定義される。ここで Q_{a1} , Q_{a2} , R_s は正定行列であり、一般に Q_{a1} , Q_{a2} , R_s は重み行列と呼ばれる。今回重み行列の値はそれぞれ

$$Q_{a1} = \text{diag}(100, 100) \quad (22)$$

$$Q_{a2} = \text{diag}(20000, 20000) \quad (23)$$

$$R_s = \text{diag}(5, 5, 5, 5) \quad (24)$$

とした。

最適レギュレータ理論に基づき制御量を目標値に追従させる制御器設計を行った結果積分型コントローラは [5] から、

$$u(t) = -R_s^{-1}B^T P_{11} \alpha(t) - R_s^{-1}B^T P_{12} \int_0^t e(t) dt + [R_s^{-1}B^T P_{11} - 2R_s^{-1}B^T P_{12} P_{22}^{-1} P_{12}^T \quad I] \begin{bmatrix} A & B \\ C & 0 \end{bmatrix}^+ \begin{bmatrix} 0 \\ I \end{bmatrix} y_r + 2R_s^{-1}B^T P_{12} P_{22}^{-1} P_{12}^T \alpha_0 \quad (25)$$

となる。

3.3.2 積分型最適サーボ系の概要とブロック線図

roll 角 ϕ , pitch 角 θ について最適サーボ系による制御を行い, yaw 角と高度である ψ , z は 3.3 節と同様に PID 制御による制御器設計を行った。作成した制御器のブロック線図を図 4 に示す。

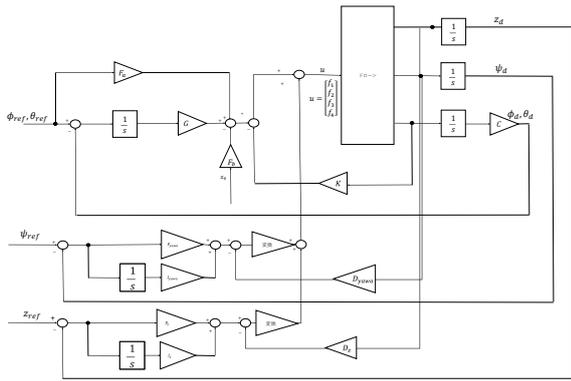


図 4 積分型最適サーボ系の制御系

4 シミュレーション

3 節で説明した 2 つの制御器についてのシミュレーションを行う。

4.1 シミュレーションの状況設定

シミュレーションの状況設定は 2 つある。まず 1 つ目がドローンをホバリングさせ、その後 roll 角と pitch 角に姿勢角目標値を与える。2 つ目が 1 つ目の状況に加えて定値外乱を加える。以上 2 つのシミュレーションを考える。またドローンの初期値は位置, 姿勢角, 速度, 角速度はすべて 0 であるとする。目標値は高度 2.5[m], roll 角 ϕ は $\frac{\pi}{6}$ [rad], pitch 角 θ は $\frac{\pi}{6}$ [rad] としている。またシミュレー

ションのゲインは表 2, 重み行列は (22) 式 ~ (24) 式に基づいている。

4.2 外乱なしのシミュレーション結果と考察

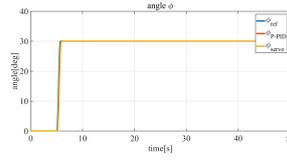


図 5 姿勢角 ϕ

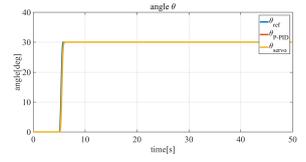


図 6 姿勢角 θ

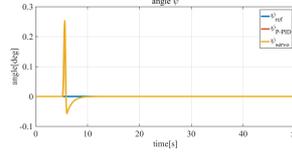


図 7 姿勢角 ψ

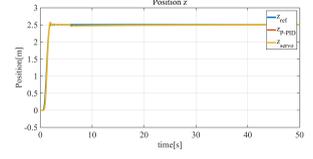


図 8 高度 z

外乱がない場合のシミュレーションでは姿勢角 ϕ , θ , ψ はすべての制御器で目標値に追従していることが図 5, 図 6, 図 7 を見て分かる。

4.3 外乱ありのシミュレーション結果と考察

定値外乱がある場合では制御器によって応答が異なる。今回印加した定値外乱は roll 角 ϕ に大きく影響することが図 9 から分かる。また積分器を持っている制御器では定値外乱に対して目標値に修正するように制御器が働いていることが見て取れる。この結果から外部からの外乱を受ける可能性がある飛行条件においては制御器として 2 つの制御器は適切である。

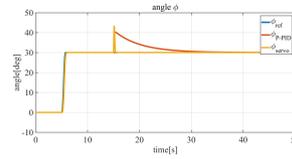


図 9 姿勢角 ϕ

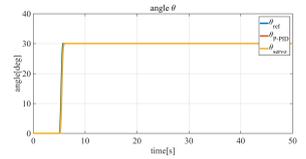


図 10 姿勢角 θ

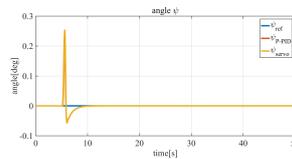


図 11 姿勢角 ψ

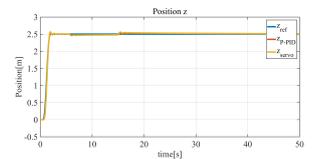


図 12 高度 z

5 慣性モーメント同定実験 (J_{xx} , J_{yy})

5.1 システム同定

システム同定とは, 対象システムの入出力データの測定値から制御モデルのパラメータを決定することである。ドローンの制御を行う上で, x 軸周り, y 軸周り, z 軸周りの慣性モーメントのパラメータを得る必要がある。

5.2 最小二乗法によるパラメータ推定

基板とセンサ類をドローンに搭載し、 x 軸から平行に離れた、ドローンの一端を回転軸にし、その軸周りの慣性モーメントを推定する。その後平行軸の定理を用いて x 軸周りの慣性モーメントの導出をした。以下の図 13 は機体座標系の x 軸に対して正面から見た図であり、図 14 はドローンを横から見たものである。

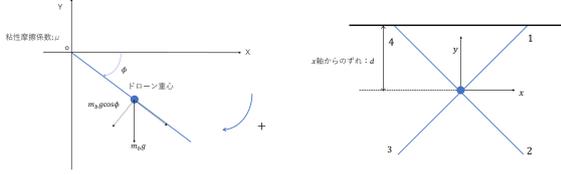


図 13 前から見たドローン 図 14 横から見たドローン

ここで運動方程式は

$$J_r \ddot{\phi} = m_b g d \cos \phi - \mu \dot{\phi} \quad (26)$$

となる。 J_r が回転軸周りの慣性モーメント、 m_b がドローンの質量、 d が回転軸と x 軸間の距離、 μ が粘性摩擦係数である。式 (26) を式変形すると

$$\begin{bmatrix} \ddot{\phi} & \dot{\phi} \end{bmatrix} \begin{bmatrix} J_r \\ \mu \end{bmatrix} = m_b g d \cos \phi \quad (27)$$

となる。慣性計測装置で計測した roll 角 ϕ から中心差分を用いることで角速度 $\dot{\phi}$ 、角加速度 $\ddot{\phi}$ の時系列データが得られる。したがって (27) 式は

$$\begin{bmatrix} \ddot{\phi}_1 & \dot{\phi}_1 \\ \ddot{\phi}_2 & \dot{\phi}_2 \\ \vdots & \vdots \end{bmatrix} \begin{bmatrix} J_r \\ \mu \end{bmatrix} = m_b g d \begin{bmatrix} \cos \phi_1 \\ \cos \phi_2 \\ \vdots \end{bmatrix} \quad (28)$$

となる。

$$V = \begin{bmatrix} \ddot{\phi}_1 & \dot{\phi}_1 \\ \ddot{\phi}_2 & \dot{\phi}_2 \\ \vdots & \vdots \end{bmatrix}, X = \begin{bmatrix} J_r \\ \mu \end{bmatrix}, Z = m_b g d \begin{bmatrix} \cos \phi_1 \\ \cos \phi_2 \\ \vdots \end{bmatrix} \text{ と定義し、まとめると} \quad V X = Z \quad (29)$$

となる。ここで、 $V \in \mathbb{R}^{n \times 2}$ は観測値行列、 $X \in \mathbb{R}^{2 \times 1}$ が同定するパラメータの行列、 $Z \in \mathbb{R}^{n \times 1}$ が入力値行列であり、 n はデータ数である。

よって X は、

$$X = (V^T V)^{-1} V^T Z \quad (30)$$

となる。回転軸周りの慣性モーメント J_r が得られた後、平行軸の定理を用いて x 軸周りの慣性モーメント J_{xx} の導出を行った。 y 軸周りの慣性モーメント J_{yy} の同定も同様の方法を用いて行った。

5.3 同定結果

5.2 節に従って同定実験を行った。その結果を以下に示す。

表 3 慣性モーメント J_{xx} と J_{yy} 同定実験

試行回数	慣性モーメント $J_{xx} [\text{kgm}^2]$	慣性モーメント $J_{yy} [\text{kgm}^2]$
1	7.572×10^{-3}	6.600×10^{-3}
2	7.345×10^{-3}	6.678×10^{-3}
3	7.392×10^{-3}	6.574×10^{-3}
4	6.901×10^{-3}	6.696×10^{-3}
5	7.514×10^{-3}	6.613×10^{-3}
6	6.905×10^{-3}	6.629×10^{-3}
7	7.379×10^{-3}	6.679×10^{-3}
8	7.464×10^{-3}	6.683×10^{-3}
平均	7.309×10^{-3}	6.644×10^{-3}

表 3 から各試行回数の同定結果がどれも近いものであったことが分かる。よって表 3 の 8 回の試行データの平均値を実験機の各軸周りの慣性モーメントとして用いることとした。

6 実験機器

6.1 制御用基板

本研究では制御基板として岩手大学の佐藤淳准教授の研究室で開発された ANS1PIC 基板 [6] (図 15) を使用する。ANS1PIC 基板は制御系の研究者、学生向けに開発された小型 Simulink リアルタイムターゲット (MATLAB/Simulink からの利用を前提とした計測・制御用機器) である。市販されているリアルタイムターゲット製品は比較的大型・高価なものが多く、据え置きでの使用には有用だが、小型機に搭載するには不便な点が多い。対して ANS1PIC 基板は小型軽量化・低コスト化を実現しており、ドローン等の小型機への搭載にも極めて有用である。これが本研究にこの基板を使用する理由の 1 つである。



図 15 ANS1PIC 基板

ANS1PIC 基板には他にも様々な特徴があり、特に我々が研究をする上で大きなメリットが MATLAB/Simulink との親和性の高さである。先にも述べたように、ANS1PIC 基板は MATLAB/Simulink の利用を前提に開発されている。そのため MATLAB の自動コード生成パッケージ等を利用すれば、自分で設計したフィルタや制御系を容易に実

験機に実装することができる仕様になっている。これにより、制御系の開発を効率的に進めることができるのである。

さらに本基板は、マルチコア構成によって各コアに役割を分担し、高精度かつ高速な処理の実現している。マルチコア構成にはソフトウェア開発の面でも利点がある。それはセンサ関係の開発とアクチュエータ関係の開発とを別のコアに分離して行えるため、より簡単にソフトウェアの開発を進めることができるということである。ANS1PIC 基板の持つ3つのコアの名称とその特徴およびソフトウェア開発環境を下記の表4に示す。

表4 ANS1PIC 基板の各コアの名称と特徴及び開発環境

名称	使用マイコンと特徴	開発環境
FC コア	PIC24F マイコン 制御系処理・信号処理	Simulink (C 自動生成)
SE コア	mbed LPC1114 マイコン 計測機器の情報処理 データの物理量変換	C, C++
SV コア	mbed LPC1768 マイコン RC 入力, モータ出力処理 通信・データ管理 飛行モード切り替え	C, C++

ANS1PIC 基板は本来航空機への搭載を目的に開発されているため各コアのプログラムも当然航空機仕様になっているが、必要箇所に変更を加えてソフトウェア開発を行うことによりドローンに対しても使用することができる。

以下に示す図16はANS1PIC 基板をドローンに搭載する際の周辺機器との接続及び信号の流れを表している。

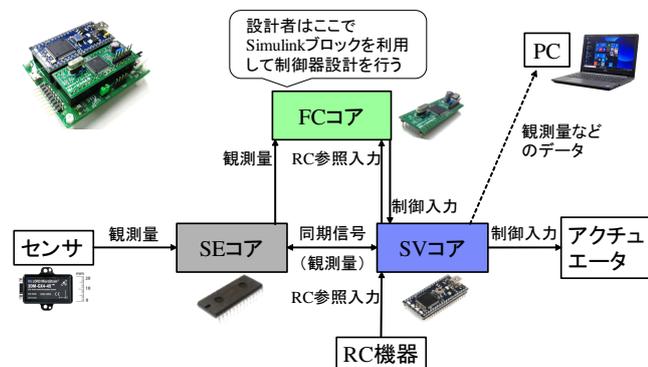


図16 ANS1PIC 基板と周辺機器の接続の概要

本研究ではRC 機器からの入力を目標姿勢角として受信し、センサから得た機体の状態量との偏差を取ってフィードバック制御を行うことを目的としている。そのため制御系の処理を担うFC コアにプロポからの目標値とセンサからの状態量の情報を与える必要がある。そこでANS1PIC 基板では図16に示した通り、RC 機器から与えられた入力はRC 参照入力としてSV コアからFC コアに送信し、センサから取得した機体の状態量は観測量としてSE コア

を通してFC コアに送信するようになっている。なお、ここでRC 入力とセンサ値はそれぞれFC コア内で扱いやすいように正規化されている。

6.2 慣性計測装置 (IMU)

本研究ではドローンの姿勢制御を行うにあたって慣性計測装置 (以下IMU) を使用する。IMU とは3 軸ジャイロ・3 軸加速度を計測することで、3 次元の角速度・加速度を計測する装置である。今回用いるIMU として、LORD 社の3DM-GX4-45 (図17) を搭載した。3DM-GX4-45 は3 軸磁気センサ・拡張カルマンフィルタを搭載しており、ドローンの位置 x, y, z , 速度 $\dot{x}, \dot{y}, \dot{z}$, 姿勢角 ϕ, θ, ψ , 姿勢角速度 $\dot{\phi}, \dot{\theta}, \dot{\psi}$ を観測可能な状態変数とすることができる。

3DM-GX4-45 は観測データをRS-232C レベルのシリアル通信で送るが、ANS1PIC 基板が対応している通信方式はI²C 通信であるため変換が必要である。本研究では岩手大学が作成した変換基板を使用してこれを行っている。



図17 3DM-GX4-45

6.3 通信機器

ドローンの姿勢制御を行うにあたって目標姿勢角を与えるためにRC 機器が必要である。本研究ではRC 機器として、Futaba 6K 送信機及びFutaba R3006SB 受信機を使用する。これらは7ch に対応しており、本研究では機体のroll 角, pitch 角, yaw 角速度, スロットルの指令と飛行モードの切り替えスイッチの計5ch の信号を使用する。

また、RC 機器とは別に、実験中のドローンの状態量等の飛行データをPC に送信するために、データリンク用無線モジュールも搭載する必要がある。本研究ではIEEE802.15.4 規格準拠の2.4GHz 帯無線通信モジュール、Xbee-Pro を採用している。本機はUART(3.3V) 通信を無線化することができる通信機器で、飛行中もリアルタイムで機体の状態を確認、記録することができる。データはログとして保存でき、実験結果の検証にも有用である。

7 ソフトウェア開発

7.1 ANS1PIC 基板のプログラムの仕様変更

先に述べたように、ANS1PIC 基板のプログラムはデフォルトの状態では航空機に合わせた仕様になっているが、本研究ではドローンに搭載する。そこで各コアの仕様

をドローンに合わせたものに改変する必要がある。クアドロータ型ドローンは4つのロータから生み出される推力によって機体を浮上させ、 x 軸周りの回転、 y 軸周りの回転、 z 軸周りの回転を発生させる。したがって我々は、RC 機器の信号を元に機体の高度、及び姿勢を操作できるようにしなければならない。そこで今回は RC 機器から送られてくる信号のうち、スティックで操作できる4つの信号をそれぞれ throttle, roll 角, pitch 角, yaw 角速度の指令値とし、その値に応じて各ロータの回転数を変化させられるようにプログラムを作成した。

7.2 FC コアのプログラム

7.2.1 概要

先の実験機器紹介でも簡単に触れたが、FC コアには Matlab/Simlink で開発された制御器が搭載されている。本研究の姿勢制御もそこに記述されることになる。

本研究のドローンには複数の飛行モードを設定しており、制御器を用いてドローンを操作するモードを Auto モード、制御器を介さず直接プロポの指令をモータに反映させるモードを Manual モードと定義している。この2つのモードは、たとえドローンが飛行中であってもプロポのスイッチ操作で操縦者が自由に切り替えることができるように FC コア内のプログラムで設定している。

7.2.2 実装している制御器

本研究ではドローンに IMU を固定し、機体の傾きに応じてモータの回転数を変化させることに成功した。制御器は RC 機器から送られてきた信号を目標値としたカスケード P-PD 制御を搭載している。ブロック線図を図 18 に示す。なお、使用したゲインは 8 章に記載する。

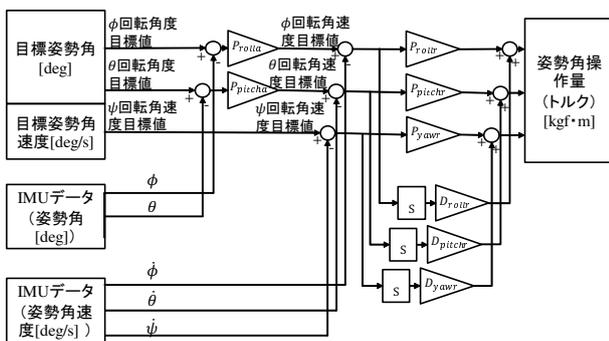


図 18 カスケード P-PD 制御による姿勢制御

7.3 SE コアのプログラム

SE コアでは各種センサ機器からのデータの収集と物理量への変換、及びそのデータの各コアへの共有を行っている。記述する処理が多いので初期設定に手間がかかるが、基本的にその後はセンサ類の構成が変わらない限り変更を加えずに使うことができる。

7.4 SV コアのプログラム

7.4.1 概要

SV コアでは RC 機器から送られてきた PWM 信号の正規化と FC コアへの転送、さらに FC コアで計算された制御入力のロータへの送信を行っている。他にも地上局との通信・データ管理も行い、飛行中の機体の情報を記録、送信している。このとき SV コアでは ANS1PIC 基板の各コアの同期状態も観測しており、基板の動作に不具合が生じた際には地上局でも即座に確認できるようになっている。

7.4.2 緊急運転モード

ANS1PIC 基板を機能拡張する場合、搭載するセンサの増設等によって基板に要求される処理が増えると、各コア間の通信や処理に大きなギャップが生じることがある。すると、パイロットの操縦と機体の応答との間にタイムラグが生じ、正常な操作が行えなくなる可能性がある。

そこで SV コアには緊急用の手動操作プログラムが実装してある。これは飛行中のコア間通信のギャップを観測し、ギャップが基準値を超えた場合は緊急運転モードに切り替えるというものである。このモードは RC 機器の信号を SV コアから直接各ロータへ送って機体を操縦する。これによって SV, FC コア間での通信を介さずに機体の操縦が行えるため、ギャップによる遅延を回避することができる。

ただ、SV コアと FC コアでは開発環境がまるで違うため、当然 FC コアで設計されているような制御器は搭載することができない。そのため本研究では緊急時の予防策として、完全な操縦不能を回避するために RC 機器のスロットル指令値を直接4つのロータへ送信し、機体をホバリングさせる機能を実装している。

8 実機実験

8.1 x 軸周りの 1 軸実験

8.1.1 モデリング

この 1 軸実験は 2 章、5.2 節で行ったモデリングとは異なった運動特性に従うものであるため、表 1 で定義された変数を用いて以下のように定義した。

$$J_{xx}\ddot{\phi} = l_y(-f_1 + f_2 + f_3 - f_4) - \mu\dot{\phi} + m_b g l_z \sin \phi \quad (31)$$

に表される。ここで μ は 5 節で同定した粘性摩擦係数であり、 l_z は回転軸と重心の距離、 g は重力加速度である。

8.1.2 シミュレーション

3.1 節で記述したカスケード型制御器のうち roll 角に関係のある制御を用いて 1 軸安定化実験のシミュレーションを行った。ただし、実機実験ではモーターのキャリブレーションなど実験の前準備を行っている間に積分項が溜まり続けてしまうおそれがあるため I 制御は省いた P-PD 制御を用いている。シミュレーションで用いた各ゲインは $P_{rolla} = 2$, $P_{rollr} = 1.5$, $D_{rollr} = 0.005$ としている。

シミュレーション結果の角度を図 19, 角速度を図 20 に示す。角度, 角速度ともに定常偏差を残しつつも収束していることが分かる。また, 角速度 $\dot{\phi}$ の目標値が 0[rad/s] に収束していないのはアウターループである角度 ϕ に定常偏差が残っているためその偏差を打ち消すための角速度が目標値として与えられていると考えられる。

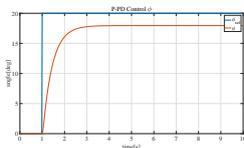


図 19 roll 角度

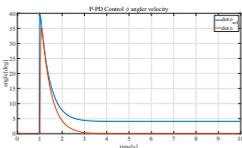


図 20 roll 角速度

8.1.3 実験結果

実験から得られた roll 角を以下の図 21 に示す。目標値としてプロポから ± 20 [deg] を印加し, そのときのドローンの角度, 角速度のデータを IMU から得た。

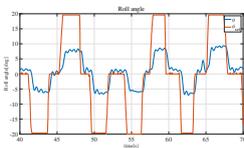


図 21 roll 角度結果

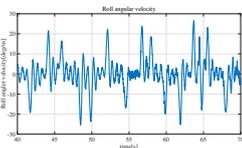


図 22 roll 角速度結果

図 21 を見ると, シミュレーションよりも定常偏差が大きく残っていることが分かった。また ± 20 [deg] を印加しても不安定化せず, 定常偏差を残しつつも安定化されていることが分かる。また平衡点である 0[deg] の周りでは不安定化せず安定していることが確認できた。

8.2 飛行実験

1 軸実験での安定化に成功したので, 次に x 軸と y 軸の安定化を目的としたドローンの飛行実験を行った。この実験において, モデリングは 2 章で行ったものであり, 制御思想は 3.1 章で述べたものを用いる。ただし, 1 軸実験と同様に I 制御を除いたカスケード P-PD 制御である。

8.2.1 実験結果

実験に用いた制御器である 7.2.2 項の図 18 におけるゲインを表 8.2.1 に示す。実験から得られた roll, pitch それぞれの角度と角速度, プロポから送られているスロットル信号を以下の図 23~27 に示す。図 23 のスロットル信号見ると 72[s] から 88[s] あたりにかけてドローンが飛行していることが分かる。また, 図 26 の pitch 角度をみるとドローンが 72[s] 付近で離陸した際に機体が $+5$ [deg] ほど傾いているのを操縦者が -5 [deg] の目標値をプロポから送ることで機体の pitch 角度を 0[deg] 付近で保ったまま飛行していることが確認できる。88[s] あたりで機体の角度, 角速度が大きく振動しているのはドローンが着陸した際の振動である。

表 5 実験時の PD ゲイン

記号	ゲイン
P_{rolla}	2
P_{pitcha}	2
P_{rollr}	1.5
P_{pitchr}	1.5
P_{yawr}	1
D_{rollr}	0.005
D_{pitchr}	0.005

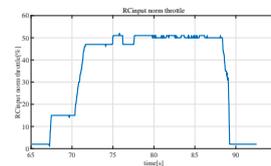


図 23 スロットル入力

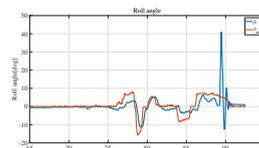


図 24 roll 角度実験結果

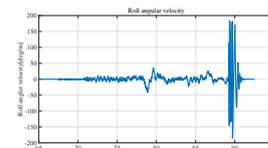


図 25 roll 角速度実験結果

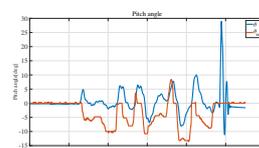


図 26 pitch 角度実験結果

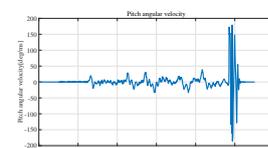


図 27 pitch 角速度実験結果

9 最後に

本研究ではシミュレーション上での複数の制御器作成し, ドローンの姿勢角制御を行った。またドローンの慣性モーメントを同定実験により求めた。さらに ANS1PIC 制御基板のドローンでの運用に向けたソフトウェア開発を行い, 実機に搭載して飛行実験を行うことに成功した。今後は yaw 軸周りの安定化を行い, 実用に足る精度での飛行を実現するとともに更なる機能拡張を目指す。

参考文献

- [1] 小出昭一郎. 物理入門コース [新装版]—解析力学. 岩波書店, 2017.
- [2] 平手貴大. ドローンによる倒立振子の安定化制御. 2017 年卒業学士論文, 南山大学理工学部機械電子制御工学科坂本・中島研究室, 2017.
- [3] 松原 厚. 精密位置決め・送り系設計のための制御工学. 森北出版株式会社, 2017.
- [4] 川田昌克. Matlab/Simulink による現代制御入門. 森北出版株式会社, 2011.
- [5] 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, Vol. 24, No. 1, pp. 40–46, 1988.
- [6] 岩手大学理工学部 システム創成工学科 機械科学コース 航空宇宙システム研究部門佐藤研究室. ANS1 PIC 制御基板 ユーザーマニュアル.