

# ドローンの飛行のロバスト性向上を目指したシミュレーション研究

2020SC036 児玉薫

指導教員：坂本登 中島明

## 1 はじめに

近年、ドローンというものは映画やMVの撮影や田畑への農業散布、災害調査や救助活動など、様々な分野で活用されている。現在では法整備やガイドラインの制定も進んでおり、これから利用が増えることが日本国内でも見通されている。[1]

ドローンの主な活躍場所について想定すると、先ほど挙げた例で用いられるのはほとんどが屋外である。当然、屋外では風など外乱の影響をドローン本体が受けるため、それを抑制するのが外での飛行においては重要となる。そこで本研究では「不確かな要素や数値の変動に強い」という特徴に注目し、 $H_\infty$  制御を用いてドローンのロバスト制御を試みる。

## 2 $H_\infty$ 制御

$H_\infty$  制御とは、最初にも軽く述べたように、外乱や摂動（制御対象が持つさまざまな不確かさや変動）の影響を抑制する制御手法である。不確かな要素を持つ制御対象に対して、想定される最悪の場合でも安定した結果が得られるように制御をするのが目的だ。[2]

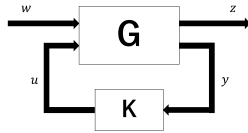


図1  $H_\infty$  制御の図 [2]

図に示した  $w$  は外乱入力、 $z$  は評価出力、 $y$  は制御出力、 $u$  は制御入力である。

$H_\infty$  制御とは  $w$  から  $z$  までの伝達関数のゲインの最大値、 $H_\infty$  ノルムをとある値  $\gamma$  以下にすること、すなわち  $\|G_{zw}\|_\infty < \gamma$  を満たす制御器  $K(s)$  を求めることが必要である。とある値  $\gamma$  はなるべく小さいものであることが求められており、先ほど述べた条件を満たす中で  $\gamma$  を求める際には「 $\gamma$  イタレーション」と呼ばれる二部探索法が用いられる。

本研究では matlab 内で使用できる hinfsyn というコマンドを用いて  $\gamma$  と制御器  $K$  を求めた。

## 3 ドローンのシミュレーション

この章では私が作成したドローンのシミュレーションについての解説を示していく。研究期間と難易度の問題から、ドローンの運動を一部に限定した状況を想定し、その場合でのシミュレーションを作成した。

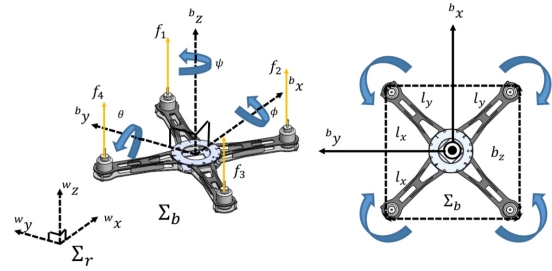


図2 ドローン機体（三次元）

図2にて示されているドローン本体に対し、 $x$  軸を貫く形で棒を通して固定することでドローンの  $x$  軸まわりの回転運動のみに絞る。それを  $x$  軸の正の向き側から見た図が図3である。本研究ではこのようにしてドローンの二次元での回転運動について考える。

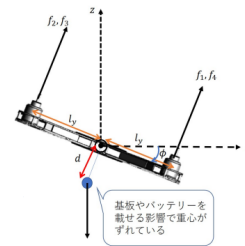


図3 ドローン機体（二次元）

本研究では図3に示すドローンの機体を想定し、シミュレーションを作成した。シミュレーションの目標は回転角  $\phi$  の制御で、目標値は5秒で0度から10度に立ち上がるステップ関数である。

また、機体の各パラメータは表1に示す。 $m_b$  はドローン

表1 ドローンのシミュレーションにおけるパラメータ

記号	値および単位
$m_b$	1.20[kg]
$J_{xx}$	$7.309 \times 10^{-3}$ [kgm <sup>2</sup> ]
$l_y$	0.17[m]
$d$	0.04[m]
$\mu$	$5.85 \times 10^{-3}$ [Nms/rad]

の質量で、 $J_{xx}$  は慣性モーメント、 $l_y$  がドローンの中心からモータまでの長さで、 $d$  がドローン機体の中心と重心までの距離、そして  $\mu$  が軸の粘性摩擦係数である。

ただし、表1で示すパラメータに関して  $J_{xx}$  は  $\pm 80\%$ 、 $\mu$

は±80%の誤差を持つものとして想定している。この数値によるぶれを抑制しつつ、目標値を満たすように設計することがこのシミュレーションの目的となる。

次に、図3で示したドローンの回転運動方程式は以下に表す。

$$\dot{\phi} = \frac{m_b g d}{J_{xx}} \phi - \frac{\mu}{J_{xx}} \dot{\phi} + \frac{l_y}{J_{xx}} (f_1 - f_2 - f_3 + f_4)$$

これを状態方程式で表すと、

$$\frac{d}{dt} \begin{bmatrix} \phi \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ \frac{m_b g d}{J_{xx}} & -\frac{\mu}{J_{xx}} \end{bmatrix} \begin{bmatrix} \phi \\ \dot{\phi} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{l_y}{J_{xx}} \end{bmatrix} (f_1 - f_2 - f_3 + f_4)$$

という形になり、この形式でmatlab内での入力に用いる。

ここからはmatlabを用いて作成したシミュレーターについての解説と、その実行結果について述べる。

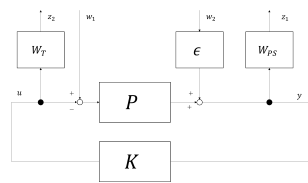


図4 一般化プラント

図4は今回の研究で作成した一般化プラントである。まずPはドローンの運動方程式を入力して作成した本研究での制御対象で、Kはその制御器である。

そしてW\_T, W\_PS, εは「重み関数」と呼ばれるもので、これらの値を適切に調整することで摂動を抑制したロバスト制御を目指す。

本研究では重み関数の値をそれぞれ、

$$W_T = \frac{10 \times s^2}{s^2 + 30s + 15^2}, \quad W_{PS} = \frac{1}{1 \times 10^{-4}s + 1 \times 10^{-9}}, \quad \epsilon = 0.9$$

という値に設定した。

以上のような準備を行ったうえでシミュレーションを実行し、その応答を示したものが図5である。比較のために同じ制御対象に対してPID制御を行った際の結果も隣に示す。

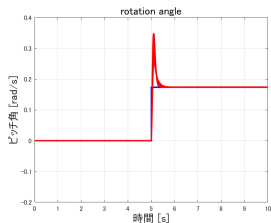


図5 H<sup>∞</sup>制御の目標値応答

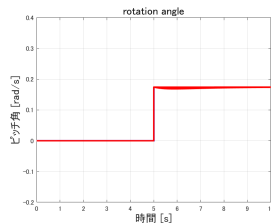
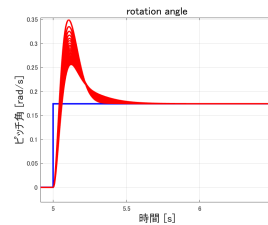


図6 PID制御の目標値応答

二つのグラフを見比べてみれば一目瞭然の通り、H<sup>∞</sup>制御の目標値応答ではオーバーシュートが大きく出てしまっている。これは推力を10[N]前後に制限するという条件の

中で何度も重み関数の値を調整した末に得られた結果であり、これよりさらに良い結果を求めたとしても、オーバーシュートを抑えるには限界があると考えている。

目標値追従という点で見ればH<sup>∞</sup>制御よりもPID制御の方が優れていることは明らかであり、シミュレーションの作成難易度においてもゲインの調整の容易さに関してもこちらの方が非常に簡単である。



)

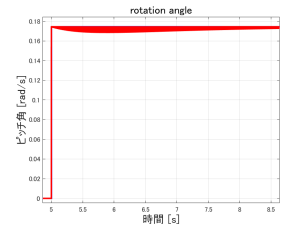


図7 H<sup>∞</sup>制御の目標値応答 (拡大図) 図8 PID制御の目標値応答 (拡大図)

しかし図7と図8を見てわかる通り、H<sup>∞</sup>制御を用いた場合の方が、パラメータ摂動による差が目標値に収束するのが早いという強みがある。

以上の結果から私は、H<sup>∞</sup>制御とPID制御を組み合わせることができれば、摂動を抑制できるというH<sup>∞</sup>制御の強みと、容易に目標値追従が成せるというPID制御の強みをそれぞれ活かし、ただ別々に一つの制御手法を以て制御を行うよりも、更に優れた結果が得られるのではないかと考察した。

#### 4 おわりに

本研究では3章で示したように、H<sup>∞</sup>制御を用いて不確かな要素の抑制を目指した制御を行った。

今回の目的であったドローンの制御への有用性の確認という点においては、H<sup>∞</sup>制御の強みを実際に試してそれを確認できたものの、目標値追従の困難さという問題があり100%達成できたとは言えない。

今後の課題として挙げられるのはH<sup>∞</sup>制御を用いたシミュレーションでのオーバーシュートの低減、H<sup>∞</sup>制御とPID制御の両方を組み合わせたシミュレーションの実装でその性能を確かめること、そして大きな目標である南山大学のドローンのシミュレーターにH<sup>∞</sup>制御を組み込むことの3つであり、本研究が二次元での回転運動に限定したH<sup>∞</sup>制御のみで終わってしまったことは非常に残念である。

#### 参考文献

- [1] 内閣官房：ドローン情報共有プラットフォーム、
- [2] 平田 光男：“実践ロバスト制御”，コロナ社 2017.
- [3] 野波 健蔵：“ドローン工学入門 モデリングから制御まで”，コロナ社 2020.