

不変多様体の学習によるドローンの極低推力変化 運動制御手法の確立

名古屋大学 大学院工学研究科航空宇宙工学専攻

椿野 大輔

1. はじめに

現在、マルチロータ型ドローン（以降、単にドローンと記載）は様々な用途で利用されており、また新たな利用も検討されている。代表的なものでも、橋梁などのインフラ点検、農業、物流、災害対策など多岐に及ぶ¹⁾。ドローンによって実現可能なことが実証されれば、その後考えるべきことは、いかに効率的に飛行させるかということになる。特にドローンは、ロータが発生する推力で重力への対抗と推進・姿勢制御の全てを行う必要があり、固定翼航空機に比べエネルギー効率に劣る。効率的な飛行には、機体構成などのハード面や飛行計画の決定など、様々な要素が影響するが、機体の運動制御は最も基本的な要素である。本研究は効率的な機体の運動制御のために入力量の削減を目指す。特に、ホバリング状態を維持するために最低限必要な入力量からの差が極めて小さくなるような運動制御を目指す。

ドローンの運動制御については、求めるレベルに応じて多くのものが提案されている。ドローンの動特性は非線形であり、制御入力となる推力や姿勢角などの状態に制約が存在する。制約条件を考慮可能な制御手法としてモデル予測制御²⁾がよく知られている。モデル予測制御は、現在時刻から将来のある時刻までの制約条件下での有限時間最適制御問題を解き、得られる最適入力の最初の値を現在の制御入力として使用し、それを制御周期で繰り返す手法である。最適制御問題を解くためのライブラリがよく整備されていることから、モデル予測制御は様々なシステムの制御に広く使用されている。ドローンの運動制御への適用も多くなされている³⁾。モデル予測制御は優れた制御手法であるが、制御周期毎に最適制御問題を解く必要があるなど、実装時には計算量が課題となる。また、制約条件が増えれば、そもそも最適制御問題の求解が困難となり、制御周期毎の最適化計算の実施が不可能となる。また、有限時間の最適制御はその区間で最も評価関数が低くなるものを求めるのであり、それが実際の制御目的の達成を意味するわけではない。

本研究では、制御周期毎の最適化計算を必要としない陽なフィードバック制御則による制御を考える。そのために、入力量の制約を扱うことができる非線形制御則設計手法として、フォーワードイング⁴⁾に注目する。この手法は、システムの状態変数間の影響関係に、あるカスケード構造が存在する場合に使用できるものである。すでに述べたように入力制約を直接扱えるだけではなく、閉ループ系の安定性を保証することができる。しかしながら、制御則の設計においてある不変多様体を表す関数とその偏導関数を用いる必要があり、その関数を得るためにはある偏微分方程式を解析的に解く必要がある。この解析的な求解は簡単なシステムの場合を除いて一般に不可能であることから、これまで現実的なシステムの制御にあまり使用されてこなかった。本研究ではこの問題を解決するために、求めるべき関数をニューラルネットワークにより近似することを提案する。さらに、ニューラルネットワークの学習では、データだけでは

なく、Physics Informed Neural Network (PINN)⁵⁾の考え方を導入し、本来満たすべき偏微分方程式の情報も用いる。これにより、制御則設計で必要となる不変多様体を表す関数の偏導関数も適切に近似できるようにすることを目指す。そして、ドローンの運動制御へ適用し、その有効性を検証する。

2. フォワードイング設計

ここでは、本研究で基礎となる非線形制御則設計法であるフォワードイング⁴⁾について説明し、課題について述べる。フォワードイングは、以下のような厳密フィードフォワード構造をもつシステムを対象とするものである。

$$\begin{aligned}\dot{x} &= f_1(\xi) + g_1(\xi)u \\ \dot{\xi} &= f_2(\xi) + g_2(\xi)u\end{aligned}$$

ただし、 $x(t) \in \mathbb{R}^{n_1}$, $\xi(t) \in \mathbb{R}^{n_2}$ ($n_1, n_2 \in \mathbb{N}$) は状態、 $u(t) \in \mathbb{R}^m$ は制御入力である。また、方程式中の各関数 f_1, f_2, g_1, g_2 は十分滑らかであるとし、適当な状態変数の平行移動により、 $f_1(0) = 0, f_2(0) = 0$ となっているとする。一つの方程式をサブシステムの方程式と考えると、二つのサブシステムは図1に示すようなカスケード接続の構造をしている。特長として、第2サブシステムは第1サブシステムへ影響を与えるが、その逆の情報の流れが存在しない。フォワードイングは、この構造を利用して安定化制御則を設計する手法である。

フォワードイング設計では、独立している第2サブシステムに対して何らかの方法で安定化制御則が得られているとする。ここではそれを、 $u = \alpha_2(\xi)$ と表し、この制御則のもと $\xi(t) \rightarrow 0 (t \rightarrow \infty)$ が成り立っているとする。この時の第1サブシステムの挙動を考える。制御則 $u = \alpha_2(\xi)$ は、第2サブシステムの安定化しか行わないため、一般に第1サブシステムは安定化されない。しかしながら、前述のカスケード構造のため、第1サブシステムの状態 x はある定常値に収束する。さらにある関数 $\beta: \mathbb{R}^{n_2} \rightarrow \mathbb{R}^{n_1}$ が存在し、この定常値は

$$\lim_{t \rightarrow \infty} x(t) = x(0) + \beta(\xi(0)) \quad (1)$$

と表せる。この関数 β を用いて、状態変数 x を次式のように変換する。

$$z = x + \beta(\xi) \quad (2)$$

すると、 z は後で示す関数 $w: \mathbb{R}^{n_2} \rightarrow \mathbb{R}^m$ を含む以下の方程式を満たす。

$$\dot{z} = w(\xi)(u - \alpha_2(\xi))$$

そこで $u = \alpha_2(\xi) + v$ のように制御入力を拡大し、これを代入した表現 $\dot{z} = w(\xi)v$ に基づいて第1サブシステムに対する制御則 $v = \alpha_1(z, \xi)$ を設計する。このようにして、システム全体の制御則を $u = \alpha_1(z, \xi) + \alpha_2(\xi)$ と構成するのが、フォワードイングによる設計手法である。システム全体の制御則の設計を、よりサイズの小さなサブシステムごとの制御則設計に分解し、各制御則を足し合わせるため、制御則に制約を付加しやすい。実際、 α_1, α_2 ともに上下限の制約を持つようなものを設計すれば、全体の制御則も上下限制約をもつものとなる。これが本研究でフォワードイングに注目する理由である。

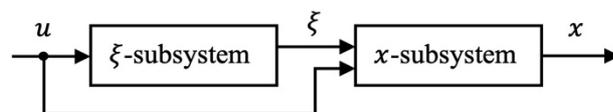


図 1. 厳密フィードフォワード構造

この手法で問題となるのは、関数 β を求める過程である。実は集合

$$\{(x, \xi) \in \mathbb{R}^{n_1+n_2} \mid x + \beta(\xi) = c, c \in \mathbb{R}^{n_1}\}$$

は制御則 $u = \alpha_2(\xi)$ を用いた場合の閉ループ系の不変多様体になっている。そのため、変換(2)は、この不変多様体を座標として用いることを意味している。不変多様体を表すことから、関数 β は偏微分方程式

$$f_1(\xi) + g_1(\xi)\alpha_2(\xi) + \frac{\partial \beta}{\partial \xi}(\xi)(f_2(\xi) + g_2(\xi)\alpha_2(\xi)) = 0, \beta(0) = 0 \quad (3)$$

の解として求められる。しかしながら、この方程式の解析的な求解は一般に不可能である。また、(2)のように変換に用いる必要があるため、偏微分方程式(3)を空間離散化して得られる離散点毎の数値解も望ましくない。この偏微分方程式の求解を含むことが、フォーワードイングがこれまであまり使用されてこなかった理由である。なお、偏微分方程式(3)より、 z が満たす方程式中の w は

$$w(\xi) = g_1(\xi) + \frac{\partial \beta}{\partial \xi}(\xi)g_2(\xi)$$

であることがわかる。すなわち、 β の偏導関数を含んでいる。

3. 提案手法: ニューラルネットワークによる不変多様体の学習

前節で述べたフォーワードイングにおける課題に対して、本研究では不変多様体を表す関数 β をニューラルネットワーク $\hat{\beta}$ で近似することを提案する。ニューラルネットワークは学習したい関係式を入力変数に関する関数として近似することから、そのまま変換(2)に用いることができる。このニューラルネットワークの学習のためのデータは、数値シミュレーションを繰り返すことで取得する。制御則 $u = \alpha_2(\xi)$ を用いたシミュレーションを行う際、 ξ の初期値 $\xi(0)$ を β の値が知りたい点に選ぶことで、その点での β 値を(1)に基づいて x の定常値として計算することができる。これを複数回繰り返して取得したものを $\hat{\beta}$ の訓練データとして学習に用いる。

実際の制御則設計では、 β そのものだけではなく、その偏導関数が必要となる。これは、前章の最後に述べた関数 w に β の偏導関数が含まれるためである。特定の点とそこでの値による学習だけでは、偏導関数が振動的になるなどの問題が発生する。この問題を解決するために、Physics Informed Neural Network (PINN) ⁵⁾ の考え方を利用して、近似した関数 $\hat{\beta}$ に対する(3)式左辺の値も評価に追加して学習を行う。このように、数値シミュレーションによるデータだけではなく、モデルから定まる偏微分方程式の情報も学習に使用することが本研究の特徴である。これを図2にまとめる。

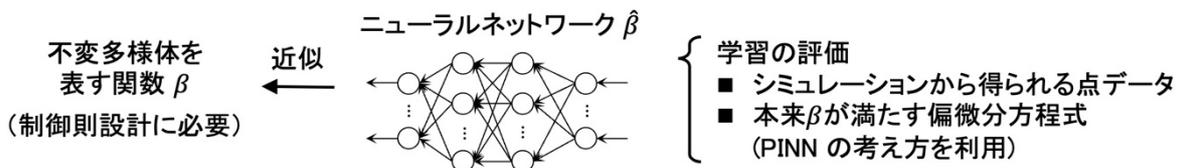


図 2. 提案手法の概要

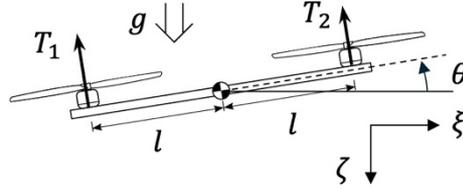


図 3. マルチロータ型ドローンの平面運動

4. マルチロータ型ドローン制御：モデルとフォワーディング設計

ここでは、図 3 に示すマルチロータ型ドローンの平面運動を考える．状態方程式は、状態を $x = (x_1, x_2, x_3, x_4, x_5, x_6)^T := (\xi, \dot{\xi}, \zeta, \dot{\zeta}, \theta, \dot{\theta})^T$ と選ぶと運動方程式より次式で与えられる．

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \end{bmatrix} = \begin{bmatrix} x_2 \\ -g \sin x_5 \\ x_4 \\ g(1 - \cos x_5) \\ x_6 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ -(1/M) \sin x_5 & 0 \\ 0 & 0 \\ -(1/M) \cos x_5 & 0 \\ 0 & 0 \\ 0 & 1/J \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (4)$$

ただし、 $M, J > 0$ はそれぞれ質量と重心周りの慣性モーメントである．制御入力として、総推力 $T_1 + T_2$ からホバリングに必要な重力に対抗するだけの推力 mg を引いた $u_1 = T_1 + T_2 - mg$ 、および総トルク $u_2 = l(T_2 - T_1)$ を選ぶ．また、位置の制御は行わず並進速度と姿勢の制御を考える．ホバリング状態に必要な入力量からの差が極めて小さくなるようなフィードバック制御則を、フォワーディングにより設計する．詳細は本研究に関する成果をまとめた論文⁶⁾を参照されたい．

フォワーディングを適用するために、システム(4)を 2 章で述べた厳密フィードバック形で表現する．まず、以下の角度、角速度部分を 2 章で述べたシステムとみなしてフォワーディングを適用する．

$$\begin{aligned} \dot{x}_5 &= x_6 \\ \dot{x}_6 &= [0 \quad 1/J] \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \end{aligned}$$

ここでは、角速度を安定化する制御則を $u_1 = 0, u_2 = \alpha_6(x_6)$ として、 $\alpha_6(x_6)$ を設計者が任意に定めることができる定数 $M_6 > 0$ に関して $|\alpha_6(x_6)| \leq M_6$ が成り立つように設計する．つぎにフォワーディングに従い、姿勢角 x_5 を現時点では未知の不変多様体を与える関数 β_5 が求められたとして変換 $z_5 = x_5 + \beta_5(x_6)$ により変換する．そして、角度も安定化する制御則として、 $u_1 = 0, u_2 = \alpha_5(z_5, x_6) + \alpha_6(x_6)$ を設計する．ここでも同様に、設計者が任意に与えることができる定数 $M_5 > 0$ を用いて、 $|\alpha_5(z_5, x_6)| < M_5$ が成り立つように設計する．これで、形式的にはホバリング状態を維持するために必要な入力量からの変化が指定値 $M_5 + M_6$ 以下となる姿勢制御則が得られる．

つぎは並進運動制御のための制御則を設計する．そのために、並進速度の方程式と回転運動の方程式を以下のように 2 章で述べた厳密フィードバック形で表現する．

$$\begin{bmatrix} \dot{x}_2 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} -g \sin x_5 \\ g(1 - \cos x_5) \end{bmatrix} + \begin{bmatrix} -(1/M) \sin x_5 & 0 \\ -(1/M) \cos x_5 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

$$\begin{bmatrix} \dot{z}_5 \\ \dot{x}_6 \end{bmatrix} = \begin{bmatrix} -w_5(x_6) \alpha_6(x_6) \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & w_5(x_6) \\ 0 & 1/J \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

この表現をもとにフォワードイングを適用する．姿勢運動の制御則 (z_5, x_6 を安定化する制御則) はすでに設計している．これをもとにその制御則下での並進速度に関する不変多様体を与える関数 β_2, β_4 が求められたとして変換 $z_2 = x_2 + \beta_2(x_5, x_6)$, $z_4 = x_4 + \beta_4(x_5, x_6)$ を導入し, 制御則 $u_1 = \alpha_{24,1}(z_2, z_4, x_5, x_6)$, $u_2 = \alpha_{24,2}(z_2, z_4, x_5, x_6) + \alpha_5(z_5, x_6) + \alpha_6(x_6)$ を設計する．姿勢制御の場合と同様に, 設計者が任意に定める定数 $M_{24,1}, M_{24,2} > 0$ に対して, $|\alpha_{24,1}(z_2, z_4, z_5, x_6)| < M_{24,1}$, $|\alpha_{24,2}(z_2, z_4, z_5, x_6)| < M_{24,2}$ が成り立つように設計する．

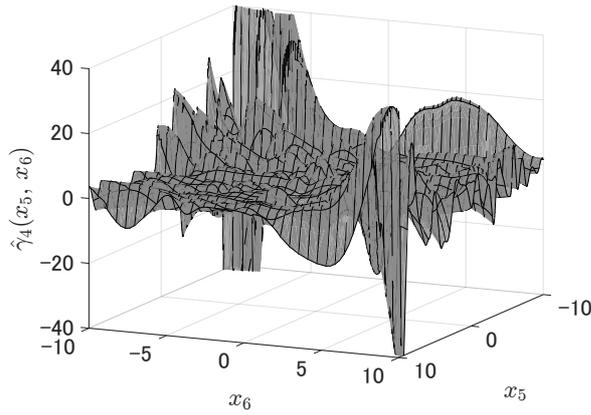
以上より, ホバリング状態を維持するために最低限必要な入力量から, 総推力については $M_{24,1}$ だけ, 総トルクについては $M_{24,2} + M_5 + M_6$ だけ変化させる, 並進速度および回転運動を安定化する制御則が設計される．ただし, 設計で用いる β_2, β_4 は解析的に求めることができないため, 4章で述べた手法によりニューラルネットワークで近似しながら設計を行う．なお, 設計においては空気力を無視しているため, 提案制御則で実際に達成可能な並進速度には上限値 $M_{(\cdot)}$ によって決まる限界がある．

5. 数値例

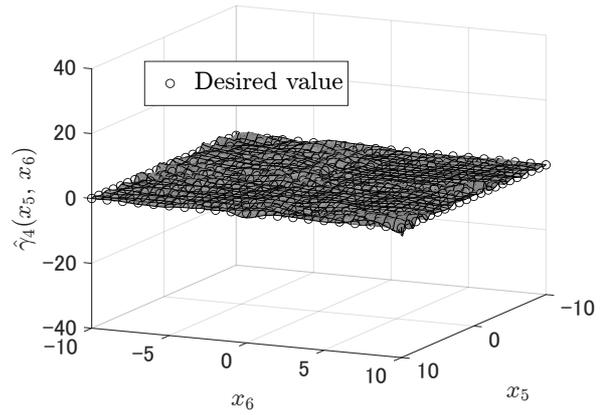
前章で述べた制御則設計において, 3章で述べた不変多様体のニューラルネットワークによる学習手法を用いた結果を示す．

まず, シミュレーションによるデータだけではなく, 偏微分方程式の情報も使用することの必要性を示す．第4章で述べた手順のうち, 並進速度 \dot{x}_4 を安定化する制御則を設計する際に必要となる不変多様体を表す β_4 を, ある学習条件でニューラルネットワーク $\hat{\beta}_4$ により近似した際の結果について示す．偏微分方程式(3)に対応する β_4 が本来満たすべき偏微分方程式左辺の点 (x_5, x_6) での値を $\hat{\gamma}_4(x_5, x_6)$ で表す．正しい β_4 であればすべての点で0になるものである．図4(a)にシミュレーションデータのみを使用して学習した場合の $\hat{\gamma}_4$ の値を, 図4(b)に偏微分方程式の値も評価に組み込んで学習した場合の $\hat{\gamma}_4$ の値をそれぞれ示す．データだけを用いた場合(a)では大きな値の振動が見られる．これらは全ての点で0になるべきものであることから, いくら点データを合わせたからといっても, 本来の関数が満たすべき方程式を満たすとは限らないことがわかる．一方, (b)の方では, 偏微分方程式の値も学習で評価したことから, 0に近い値をとっている．

この優劣自体は, 学習に用いた評価が異なることから当然の結果といえる．重要なのは, この分布が0に近いことが, 関数 $\hat{\beta}_4$ を用いた場合の $x_4 + \hat{\beta}_4(x_5, x_6)$ が定ベクトルとなる集合の不変性に必要ということである．すなわち, 不変多様体に基づいた制御則設計法であるフォワードイングにおいて, その不変多様体を十分に近似できていないことは, どれだけ点データを十分に再現できていたとしても, 得られる制御則が本来の性能を発揮でき

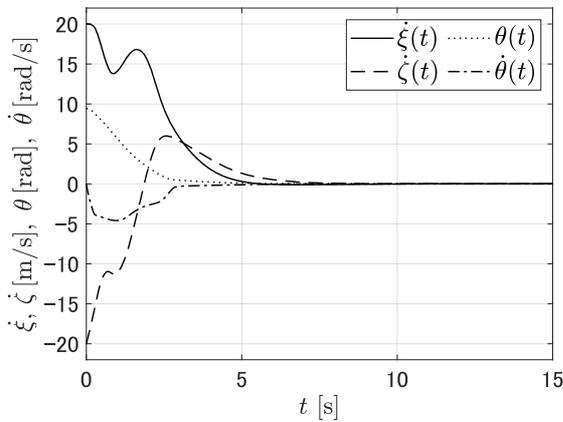


(a) 点データのみを用いた場合

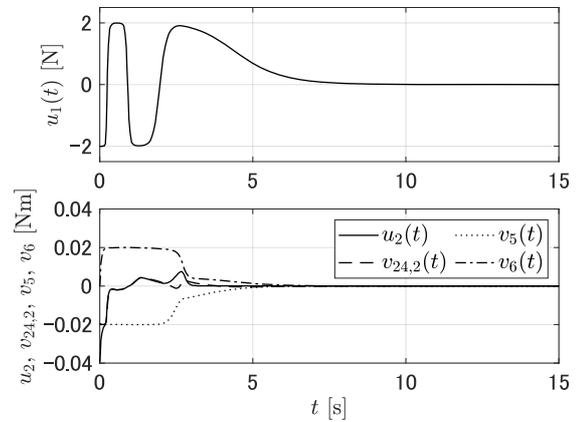


(b) 偏微分方程式の値も評価する場合

図 4. 近似関数 $\hat{\beta}_4$ に対する偏微分方程式の値分布



(a) 状態変数の時間変化



(b) 入力量の時間変化

図 5. シミュレーション結果

ないことを意味する．そのため，近似的なフォーディングを目指す本手法にとって，ニューラルネットワークの学習において，モデルから定まる情報である偏微分方程式値を導入したことは，重要な意義をもっている．

最後に，提案手法によって得られた制御則を用いた場合の数値シミュレーション結果について示す．機体のパラメータは $m = 5.5 \times 10^{-1}$ [kg], $J = 1.5 \times 10^{-3}$ [kg · m²] とした．入力の上限值は $M_{24,1} = 2.0$ [N], $M_{24,2} + M_5 + M_6 = 6.0 \times 10^{-2}$ [N · m] としして制御則を設計した．初期条件を $x_2(0) = \dot{\xi}(0) = 20$ [m/s], $x_4(0) = \dot{\zeta}(0) = -20$ [m/s], $x_5(0) = \theta(0) = 3\pi$ [rad], $x_6(0) = \dot{\theta}(0) = 0$ [rad/s] とししてシミュレーションを行った場合の結果を図 5 に示す．各状態変数の変化を(a)に，制御に必要な入力の値を(b)に示している．図 5(a)より，全ての状態変数が時間とともに 0 に収束していることが確認できる．すなわち，設計した制御則により漸近安定化が達成できている．つぎに図 5(b)により安定化に要した入力量を確認する．まず，総推力である u_1 について，(b)の上段の図より，設定した上下限值 2.0 の範囲内で変化していることがわかる．また，総トルクである u_2 を示す下段の図では， u_2 全体と設計の各段階で求めた制御則を $v_6 = \alpha(x_6), v_5 = \alpha_5(z_5, z_6), v_{24,2} = \alpha_{24,2}(z_2, z_4, z_5, x_6)$ としして記載している．図より総トルク入力が設定した上下限值である 6.0×10^{-2} の範囲で変化していることが

確認できる。なお、本シミュレーションでは、角度の初期値を大きくとっており、非線形性の影響が大きい範囲内のシミュレーションとなっていることに注意されたい。そのように非線形領域においても、提案手法により事前に設計しておいたフィードバック制御則により、入力制約内での漸近安定化が確認され、提案手法の有効性を示している。

6. まとめ

本研究では、マルチロータ型ドローンに対して、ホバリング状態を維持する入力からの変化が極めて小さくなるようなフィードバック制御則の設計について扱った。入力の変化を小さくするために、入力制約付きの非線形制御問題として扱い、フォワードイングを用いた設計法を提案した。フォワードイングの適用で問題となる不変多様体を表す関数の偏微分方程式からの求解を、ニューラルネットワークによる学習に置き換えることを提案した。この学習において、単に点データのみで学習するのではなく、不変性に関連した偏微分方程式の値も学習の評価に加えることで、フォワードイングに適した近似を行うことができた。平面運動に対する数値シミュレーションにより、その有効性を確認した。

当初は実機を用いた実験まで行う予定であった。実機の準備も行なっていたものの、3次元運動に対する提案手法による不変多様体の学習の基礎検討に時間を要してしまい、実機実証は今後の課題に残されることになった。

謝辞

本研究を遂行するにあたり、公益財団法人 天野工業技術研究所から多大なご支援を頂きました。ここに記して謝意を示します。また、本研究を共同で行った名古屋大学大学院工学研究科航空宇宙工学専攻大学院生伊藤黎氏に謝意を示します。

参考文献

- 1) 野波健蔵 (編), “続・ドローン産業応用のすべて—進化する自律飛行が変える未来—”, オーム社, 2020.
- 2) D. Q. Mayne, J. B. Rawlings, C. V. Rao, P.O.M. Scokaert, “Constrained model predictive control: Stability and optimality,” *Automatica*, vol. 36, no. 8, pp. 789-814, 2000.
- 3) H. Nguyen, M. Kamel, K. Alexis and R. Siegwart, “Model predictive control for micro aerial vehicles: A survey,” *Proceedings of the 2021 European Control Conference (ECC)*, pp. 1556-1563, 2021.
- 4) R. Sepulchre, M. Jankovic, P. V. Kokotovic, “Integrator forwarding: A new recursive nonlinear robust design,” *Automatica*, vol. 33, no. 5, pp. 979-984, 1997.
- 5) M. Raissi, P. Perdikaris, G. E. Karniadakis, “Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations,” *Journal of Computational Physics*, vol. 378, pp. 686-707, 2019.
- 6) 伊藤, 三田, 椿野, “フォワードイング設計におけるデータとモデルの情報を用いた安定多様体の近似手法,” 計測自動制御学会論文集, vol. 61, no. 2, 2025. (採録決定済)