

〈一般研究課題〉 車輪付きドローンに対するモデル予測経路積分制御
の設計と実験検証
助成研究者 名古屋工業大学 仲野 聰史



車輪付きドローンに対するモデル予測経路積分制御 の設計と実験検証

仲野 聰史
(名古屋工業大学)

Model Predictive Path Integral Control for Two-Wheeled Drones: A Control Framework and Experiments

Satoshi Nakano
(Nagoya Institute of Technology)

Abstract :

This research presents a novel motion planning approach for two-wheeled drones capable of both ground driving and aerial flight. Existing methods for such hybrid systems often rely on gradient-based optimization and assume that obstacle shapes are differentiable, which limits their applicability in complex or unstructured environments. To address this issue, we propose a motion planning framework based on Model Predictive Path Integral (MPPI) control. This approach enables navigation through arbitrarily shaped obstacles by dynamically switching between driving and flight modes. To deal with the instability and discontinuities caused by mode transitions, the method introduces a mode-dependent control space and employs an auxiliary controller to guide MPPI sampling. Simulation results validate that the proposed method enables robust navigation and effective obstacle avoidance through seamless mode switching.

1. はじめに

近年、ドローンは監視、輸送、探索といったさまざまなタスクで重要な役割を果たしている [1, 2]。中でも、地上走行と飛行の両モードを切り替えて移動可能なドローンは、エネルギー効率と移動能力の両立という観点から注目を集めている。二輪ドローン(図1)はその代表例であり、車輪による地上走行によりエネルギー消費を抑えつつ、必要に応じて飛行によって地形に依存しない移動

を実現する [3]. こうした特性を併せ持つハイブリッドな移動ロボットは、効率性と柔軟性を兼ね備えた高性能なアプローチといえる。

しかしながら、地上走行モードと飛行モードでは動力学や制約条件が不連続に変化するため、両モードを同一の制御枠組みで統合的に扱うことは容易ではない。とくに、地面との接触や障害物回避を伴うような非構造的な環境では、従来の勾配ベースの逐次最適化手法では適切な計画が得られないことが多い。

これまで、二輪ドローンの運動計画にはモデル予測制御(Model Predictive Control, MPC)がよく用いられてきた [4, 5]. MPCは、システムの動力学と制約を考慮しつつ、有限の時間範囲でのコスト関数を最小化する手法である。しかし、勾配に基づくMPCでは、非微分可能なコストや動的制約を扱うことが難しいため、地面接触時の力学や任意形状の障害物回避といった課題に対応するのは困難である。一方、モデル予測経路積分制御(Model Predictive Path Integral control, MPPI)は、サンプリングベースの最適化手法であり、非微分可能な要素を扱えるため、ロボットの運動計画において有望な手法として注目されている [6]. MPPIは確率的なサンプリングに基づき、リアルタイムで非微分可能なコスト関数の最小化を実現することができる。

ドローンに対するMPPIの応用に関する研究は進んでおり、適応制御との統合や、衝突リスクを考慮した経路計画、さらにはコストマップを利用した非凸・非微分可能な障害物制約の扱いなど、多様なアプローチが提案されている [7]. しかし、これらはいずれも一般的な飛行ドローンを対象としたものであり、車輪を備えた二輪ドローンに適用された例はこれまでにない。

一方で、従来の二輪ドローンに関する研究では、主に非線形のカスケード制御によって、地上と飛行それぞれのモードに個別に対応した制御器を切り替えるアプローチが採用されていた。このような手法は計算コストが低く、軌道追従には有効であるものの、障害物回避やモード統合といった柔軟な運動計画には対応していない。また、勾配ベースの非線形MPCを用いた手法では、補完制約を用いることでモード切り替えを最適化問題に含めることができるが、コストマップや地面接触時の反力といった非微分可能な要素を扱うには限界がある。

本研究では、MPPIを二輪ドローンに適用し、地上走行と飛行を動的に切り替えながら移動可能な運動計画手法を提案する。提案手法では、不連続なコストを扱うことが可能な MPPI の特長を活かし、ペナルティコストによる障害物制約の定式化を行う。この定式化によって、従来手法では困難であった任意形状の障害物に対応可能な障害物回避制御を実現する。

2. 記号の定義

MPPIフレームワークにおける状態遷移モデルを構築するために必要な記号の定義を行う。慣性座標系を $\Sigma_w = (i_x, i_y, i_z)$ 、機体座標系を $\Sigma_b = (b_x, b_y, b_z)$ とする。ここで、 b_x, b_y はロータの中心を通る平面上にあり、 b_y は車軸方向、 b_z はその平面に垂直であり、ロータの推力方向と一致する。機体

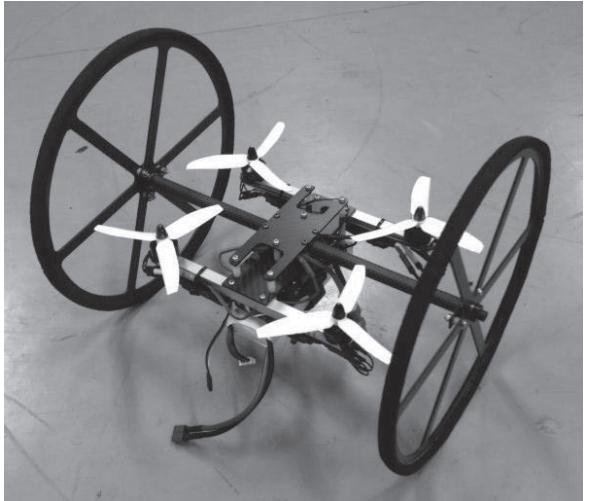


図1. 車輪付きドローン

位置を

$$\xi = \begin{bmatrix} \xi_x \\ \xi_y \\ \xi_z \end{bmatrix} \in R^3$$

とする。機体の姿勢はZYXオイラー角で表され、

$$\eta = \begin{bmatrix} \psi \\ \theta \\ \phi \end{bmatrix}$$

とする。回転行列 $R(\eta) \in SO(3)$ は次式で与えられる。

$$R(\eta) = \begin{bmatrix} s_\theta c_\psi s_\phi + c_\phi s_\psi & c_\phi s_\theta c_\psi - s_\phi s_\psi & c_\theta c_\psi \\ s_\theta s_\psi c_\phi - c_\phi c_\psi & c_\phi s_\theta s_\psi + s_\phi c_\psi & c_\theta s_\psi \\ -c_\theta s_\phi & -c_\theta c_\phi & s_\theta \end{bmatrix}$$

ここで、 $s=\sin(\cdot)$, $c=\cos(\cdot)$ 。また、標準基底 e_x, e_y, e_z を用いれば、 $[b_x, b_y, b_z] = R[e_x, e_y, e_z]$ である。運動学は以下で表される。

$$\dot{\xi} = Rv$$

$$\dot{\eta} = \Phi\Omega$$

ここで、 $v=[v_x, v_y, v_z]^T$, $\Omega=[\Omega_x, \Omega_y, \Omega_z]^T$ はそれぞれボディ速度、ボディ角速度である。変換行列 $\Phi \in R^{3 \times 3}$ は次式で得られる。

$$\Phi(\eta) = \begin{bmatrix} 0 & -\sin\phi/\cos\theta & \cos\phi/\cos\theta \\ 0 & \cos\phi & -\sin\phi \\ 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \end{bmatrix}$$

物理パラメータとして、質量 m , 慣性行列 $J=\text{diag}(J_x, J_y, J_z)$, 車輪直径 d , 車軸長さ l , 推力 f , トルク $\tau \in R^3$ を定義する。

3. 制御系設計

3.1. 制御目標および制御構造

車輪付きドローンが任意形状の障害物がある環境下で、目標位置まで走行または飛行する状況を考える。制御目標は、障害物を回避しつつ機体位置 ξ を目標位置 $\xi_d \in R^3$ に追従させることである。また、車輪付き

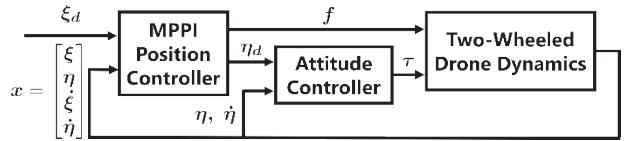


図2. 提案制御系

ドローンに対する制御入力は、全推力 $f \in R$ およびトルク $\tau \in R^3$ とする。本研究では、障害物回避と目標位置追従のための位置制御を MPPI (Model Predictive Path Integral) で行う。まず、全推力 f と目標姿勢 η_d を MPPI の制御入力

$$u = \begin{bmatrix} f \\ \eta_d \end{bmatrix}$$

として算出する。さらに、目標姿勢 η_d に対して現在の姿勢 η を追従させるための姿勢制御則によって、トルク入力 τ を算出する。制御系全体は、図2に示すカスケード構造となる。

3.2. MPPIによる位置制御則

位置制御では、2通りの制御入力を決定する MPPI を並列に実行し、高度に応じて適切な制御入

力に切り替える。2つの MPPI のアルゴリズムは、入力系列の初期化則以外は共通であり、概要を Algorithm 1 に示す。

Algorithm 1: MPPI for Position Control

Require:

- K : Number of rollouts
- N : Time steps
- F_{model} : Transition model
- Δt : Time step
- Σ, λ : Parameters

While task not completed:

1. Initialize nominal control:

$$u_j^{\text{nom}} = \text{Initialize}()$$

2. Get current state:

$$x_0 = \text{CurrentState}()$$

3. **For** $k = 1, \dots, K$:

- a. Set initial state:

$$x_0^k = x_0$$

- b. Sample noise:

$$\delta^k = (\delta u_0^k, \dots, \delta u_{N-1}^k),$$

where $\delta u_j^k \sim N(0, \Sigma)$

- c. **For** $j = 0, \dots, N - 1$:

- o Perturb control:

$$u_j^k = u_j^{\text{nom}} + \delta u_j^k$$

- o Simulate forward:

$$x_{j+1}^k = F_{\text{model}}(x_j^k, u_j^k)$$

- d. Compute trajectory cost:

$$S_k = \text{ComputeCost}(x^k, u^k)$$

4. Compute baseline cost:

$$\rho = \min(S_1, \dots, S_K)$$

5. Compute normalization factor:

$$\mu = \sum_{k=1}^K \exp\left(-\frac{1}{\lambda}(S_k - \rho)\right)$$

6. **For** $k = 1, \dots, K$:

Compute weight:

$$w_k = \frac{1}{\mu} \exp\left(-\frac{1}{\lambda}(S_k - \rho)\right)$$

7. For $j = 0, \dots, N - 1$:

Compute optimal control input:

$$u_j^{\text{opt}} = \sum_{k=1}^K w_k u_j^k$$

8. Send first control input to low-level controller:

$$\text{SendToLLController}(u_0^{\text{opt}})$$

End While

まず、2つの MPPI においてそれぞれ、初期化則によって入力系列を初期化する。つぎに、初期化した入力系列に基づき、平均0、分散 Σ のガウスノイズによって入力系列のサンプリングを行う。サンプリングした入力系列に対して、状態遷移モデルとコスト関数によって、各サンプルのコストを計算する。最後に、計算したコストと入力系列の重み付け和によって最適入力系列を計算し、その最初のステップを MPPI の制御入力とする。

3.3. コスト関数

制御目標である目標位置への追従と障害物回避のためのコスト関数を設計する。本研究では、状態偏差の二次形式を含む終端コストとランニングコスト、および障害物との衝突に対するペナルティコストによって、制御目標を達成する。コスト関数は次式で表される。

$$S_k = \|\tilde{x}_N^k\|_{C_t}^2 + \sum_{j=0}^{N-1} (\|\tilde{x}_j^k\|_{C_r}^2 + \|u_j^k\|_{C_u}^2 + C_{\text{obs}} s_{\text{obs},j}^k)$$

ここで、 $s_{\text{obs},j}^k$ は障害物と衝突した場合に 1、それ以外では 0 をとるバイナリ変数であり、次のように定義される。

$$s_{\text{obs},j}^k = \begin{cases} 1 & , \text{if collision} \\ 0 & , \text{otherwise} \end{cases}$$

また、 \tilde{x}_N^k は状態偏差を表し、

$$\tilde{x}_N^k = x_N^k - x_d, x_d = [\xi_d^\top \quad 0^{1 \times 9}]^\top$$

と定義される。さらに、 $C_t \in R^{12 \times 12}$, $C_r \in R^{12 \times 12}$, $C_u \in R^{4 \times 4}$ は半正定値の重み行列である。ノルム記号 $\|\cdot\|_p$ は、重み付きユークリッド内積

$$\|u\|_p^2 = u^\top P u$$

を意味する。また、 C_{obs} は障害物衝突時に課されるペナルティの重み係数である。

4. 数値実験

本節では、前節で設計した制御則によって制御目標が達成されていることを、数値シミュレーションを用いて確認する。車輪付きドローンの初期状態は、初期位置： $\xi_0 = [0 \ 0 \ 0]^\top$ 、初期速度： $\dot{\xi}_0 = [0 \ 0 \ 0]^\top$ 、初期姿勢： $\eta_0 = [0 \ 0 \ 0]^\top$ 、初期角速度： $\dot{\eta}_0 = [0 \ 0 \ 0]^\top$ と与える。問題設定として、目標位置 $\xi_d = [2.4 \ 0 \ 0]^\top$ が与えられており、障害物としては高さ方向が i_y に一致する、半径 $r_{\text{obst}} = 0.05 \text{ m}$ の円柱が、 $(i_x, i_z) = (1.2, -0.14)$ の位置に存在するとする。車輪付きドローンの物理パラメータは、機体質量： $m = 0.938 \text{ kg}$ 、慣性テンソル： $J = \text{diag}(0.00933, 0.00285, 0.01130) \text{ kg} \cdot \text{m}^2$ 、車輪の直径： $d = 0.28 \text{ m}$ 、車軸長さ： $l = 0.35 \text{ m}$ 、地面との反発係数： $e = 0$ とする。障害物との衝突判定は、機体の重心位置 ξ が、障害物領域から $\sqrt{d^2 + l^2}/2$ だけオフセットした範囲内にあるかどうかによって判

断する。

制御パラメータとして、MPPIにおける重み行列は以下のように設定する。

$$\begin{aligned} C_t &= 10^3 \cdot \text{diag}(6, 12, 6, 0, 0, 0, 2.4, 4.8, 2.4, 0, 0, 0), \\ C_r &= 10^2 \cdot \text{diag}(3, 6, 3, 0, 0, 0, 1.2, 2.4, 1.2, 0, 0, 0), \\ C_u &= \text{diag}(1.6, 1.6, 1.6, 1.6), \\ C_{\text{obs}} &= 10^6 \end{aligned}$$

その他のシミュレーション条件は以下の通りである。

- ・ サンプル時間 : $\Delta t = 0.02 s$
- ・ モード切替高さ : $\xi_{z,\text{switch}} = 0.0841 m$
- ・ サンプル数 : $K = 700$
- ・ 補助制御器に基づくサンプル数 : $K_{\text{aux}} = 300$
- ・ 温度パラメータ : $\lambda = 10$
- ・ 補助制御器ゲイン : $K_{\xi} = [\text{diag}(1, 1, 1) \text{ diag}(1, 1, 1)]$
- ・ 姿勢制御の比例・微分ゲイン : $k_{\eta} = 1, k_{\dot{\eta}} = 1$
- ・ ノイズの分散 :
 - 地上付近 : $\Sigma = \text{diag}(2.25, 0.03, 0.03)$
 - 飛行用 : $\Sigma = \text{diag}(2.25, 0.03, 0.03, 0.03)$

図3に車輪付きドローンの軌道を示す。黒実線が機体重心の軌跡であり、薄い灰色の領域が障害物の判定領域を表している。ここから、目標値への収束と障害物の回避が実現できていることが確認できる。

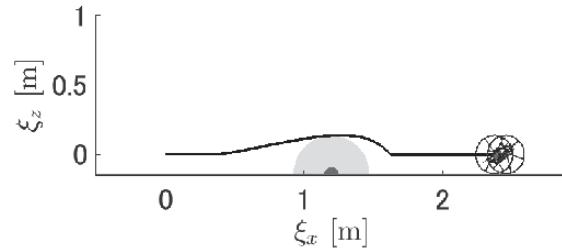


図3. 数値実験結果

5. おわりに

本研究では、二輪ドローンのための新しいMPPIベースのナビゲーションフレームワークを提案した。提案した制御フレームワークは、地上走行と飛行を状況に応じて動的に切り替えながら障害物を回避し、目的地に向かって移動することが可能であることを確認した。今後の課題は、周囲環境の観測やプランニングを含むアーキテクチャへの拡張が挙げられる。

参考文献

- [1] 野波健蔵, "ドローン技術の現状と課題およびビジネス最前線," 情報管理, Vol. 59, No. 11, pp. 755–763, 2017.
- [2] K. Nonami, F. Kendoul, S. Suzuki, W. Wang and D. Nakazawa, "Autonomous Flying Robots," Springer, 2010.
- [3] N. Takahashi, S. Yamashita, Y. Sato, Y. Kutsuna and M. Yamada, "All-Round Two-Wheeled Quadrotor Helicopters with Protect-Frames for Air-Land-Sea Vehicle (Controller Design and Automatic Charging Equipment)," Advanced Robotics, Vol. 29, No. 1, pp. 69–87, 2015.

- [4] R. Zhang, J. Lin, Y. Wu, Y. Gao, C. Wang, C. Xu, Y. Cao and F. Gao, "Model-Based Planning and Control for Terrestrial-Aerial Bi-modal Vehicles with Passive Wheels," In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1070–1077, 2023.
- [5] T. Wu, Y. Zhu, L. Zhang, J. Yang and Y. Ding, "Unified Terrestrial/Aerial Motion Planning for HyTAQs via NMPC," *IEEE Robotics and Automation Letters*, Vol. 8, No. 2, pp. 1085–1092, 2023.
- [6] G. Williams, P. Drews, B. Goldfain, J. M. Rehg and E. A. Theodorou, "Information-Theoretic Model Predictive Control: Theory and Applications to Autonomous Driving," *IEEE Transactions on Robotics*, Vol. 34, No. 6, pp. 1603–1622, 2018.
- [7] M. Minarik, R. Penicka, V. Vonasek and M. Saska, "Model Predictive Path Integral Control for Agile Unmanned Aerial Vehicles," In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 13144–13151, 2024.