

小型飛行船のPD制御

2006MI198 山田 浩之

指導教員 陳 幹

1 はじめに

本研究では、情報処理学会ソフトウェア工学研究会主催の2009MDD ロボットチャレンジに向け小型飛行船の制御を行う。組込みソフトウェア開発の技術力向上のためには、ソフトウェア工学で知られている各種のモデリング技術を適切に応用することが必要であり、その応用力を身につけることが大会の目的となる。飛行船にはモーター(上昇用1つ、旋回・直進用2つ)が3つ搭載されており、各モーターを制御し、飛行船を出発地から離陸させ、いくつかの規定動作を行った後、立ち寄り点を通りながら目的地に着陸させる事が目標となる。大会には規定が複数存在し、それらを満たすような動作をさせる飛行船の制御モデルの設計が必要となる。直進・旋回・上昇用のPDコントローラをそれぞれ独立に作成し、それら3つの動作をうまく組み合わせ航行を行う。目標指令は各目標地点に到着した時点で変更し、到着時の飛行船の状態は考慮しない。よって、最適な軌道での航行は不可能であるが、目標地点の1m以内を通過させるような精度の航行を期待する。

2 運動方程式

座標系 x - y と、鉛直座標 z をとり、重心を回転軸としたときの飛行船の前方向と x 軸のなす角度を θ とすると、運動方程式は以下のように表わされる [1]。

$$\begin{aligned}m\ddot{x}(t) &= -\mu\dot{x}(t) + F \cos \theta(t) \\m\ddot{y}(t) &= -\mu\dot{y}(t) + F \sin \theta(t) \\m\ddot{z}(t) &= -D_z\dot{z}(t) + F_z(t) + T_z - mg \\J\ddot{\theta}(t) &= -\psi\dot{\theta} + N(t)r_f \\(F(t) &= F_R + F_L, N(t) = F_R - F_L)\end{aligned}$$

上記は、3次元での飛行船の運動方程式である。ただし、 m [kg]; 飛行船の質量、 r_f [m]; 重心とプロペラの距離、 μ [kg/s]; 直進方向の摩擦係数、 ψ [kg・m²/s]; 回転方向の摩擦係数、 D_z [kg/s]; 垂直方向の摩擦係数、 J [kg/m²]; 回転モーメント、 F_R ; 右プロペラの推力、 F_L ; 左プロペラの推力、 F_z ; 中心プロペラの推力、 T_z ; 浮力である。また、 $T_z \approx mg$ とし、中心プロペラの推力が無い状態では多少落下運動をするように調整しておく。

3 微分先行型 PD 制御

高度の制御を例にとると、PID 制御では偏差の微分値 $\dot{e}_z(t) = \dot{r}_z(t) - \dot{z}(t)$ を用いるため、目標値 $r_z(t)$ が急激に変化する場合、大きな操作量 $u(t)$ を必要としてしまう。その状況を避けるため、偏差の微分値 $\dot{e}_z(t) = \dot{r}_z(t) - \dot{z}(t)$

の代わりに $-\dot{z}(t)$ を用いる。このような PD 制御を微分先行型制御といい、特に積分ゲインを 0 とした

$$u(t) = k_p e(t) - k_d \dot{z}(t)$$

を用いるとき、微分先行型 PD 制御と呼ぶ [2]。

4 航行における評価基準

大会における評価には、モデルの評価と航行の評価の2つがあり、特に航行においては以下のような基準で審査員により判断される [3]。

- 1) 指定された動作で規定動作を行った。
- 2) 立ち寄り地点近傍 1m 以内を通過した。
- 3) 立ち寄り地点に機体が接触した。
- 4) 目標値の円形内に降下して着地、静止した。
- 5) 着地はできたが、静止できずバウンドした。
- 6) 飛行中の情報を審判が理解できる形でスクリーンに実時間表示できた。

5 上昇運動シミュレーション (規定動作)

コンテストには [上昇・下降・直進・後退・旋回・ホバリング] といった規定動作が存在し、指定された動作を指定された順序で行わなければならない。今回のコンテストの規定動作は、3m 上昇 + ホバリングであるので、その動作を航行シミュレーションにより確認する。ただし、初期状態は $z_0 = 0$ 、目標指令は $r_z = 3$ のように設定する。図 1 は上昇 + ホバリングのシミュレーション結果である。

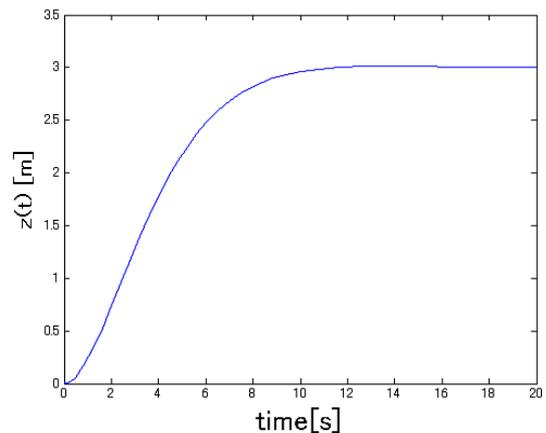


図 1 上昇 + ホバリングのシミュレーション結果

図 1 から、約 10 秒で目標値 3 に追従しているのがわかる。この結果より、小型飛行船は $z = 0$ から $z = 3$ まで上昇し、その後ホバリングが可能であると予測できる。つまり、規定動作が可能であると考えられる。

6 直進・旋回運動シミュレーション

6.1 旋回命令

直進・旋回運動シミュレーションでは、 x 、 y に目標指令を与えることで航行を行うが、旋回をする以上 θ への目標指令が必要となる。しかし、 θ への目標指令は航行中の小型飛行船の位置により常に変動してしまう。そのため θ への目標指令を

$$r_\theta = \arctan \frac{e_y}{e_x}$$

によって求め、常に目標指令を変化させることにより飛行船が目標地点に向くように設計した。ただし、 e_x : x 座標における目標地点と現在地点との差、 e_y : y 座標における目標地点と現在地点との差である。

6.2 シミュレーション結果

上昇後の航行を行う上で最も必要となる動作が、直進・旋回である。それぞれの動作を上手く組み合わせ、目標地点を目指す。これらの動作を xy 平面上での航行シミュレーションにより確認する。初期状態は $(x_0, y_0) = (0, 0)$ 、 $\theta = 0$ 、 $\frac{\pi}{4}$ 、 $\frac{\pi}{2}$ 、 π 、 $\frac{3\pi}{2}$ 、目標指令は $(r_x, r_y) = (5, 5)$ のように設定する。図 2 は直進・旋回のシミュレーション結果である。

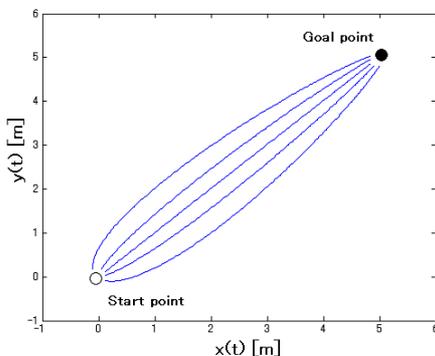


図 2 直進・旋回のシミュレーション結果

図 2 により目標座標 (5,5) に到着しているのがわかり、到着までの時間は約 60 秒であった。この結果より、小型飛行船は初期状態に影響されことなく目標地点に到着することができるかと予測できる。

7 コンテスト航行シミュレーション

最終的には小型飛行船を二つの立ち寄り点 (Way point) の 1m 以内を通過させ、ゴールポイントへ到着させるよう航行させなければならない。その航行が実現可能なモデルであるかを実証するため、コンテスト本番と同じような

ポイントを指定してシミュレーションを行った。図 3 はコンテスト航行シミュレーションの結果である。

Lift off point : $(x,y)=(0,0)$
way point1 : $(x,y)=(6,4)$
way point2 : $(x,y)=(-1,7)$
goal point : $(x,y)=(3,16)$

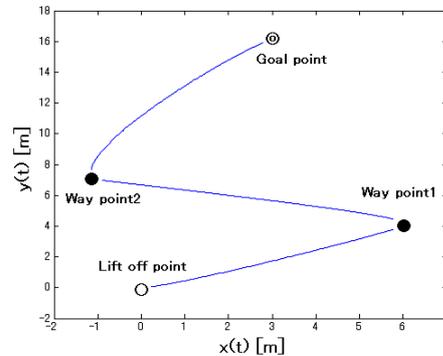


図 3 コンテスト航行シミュレーション

航行時間は、Lift off point から Goal point まで約 3 分 20 秒であった。この結果からわかるように、離陸と着陸の時間を含めても 5 分以内で到着することが可能であると予測できる。

8 考察と今後の課題

小型飛行船の制御モデルを PID 制御理論を基に作成し、航行シミュレーションを行った。最適な航路とまではいかなかったが、当初の目的であったコンテストの内容を満たすような制御モデルが設計できた。今後の課題としては、しっかりとしたシステム同定を行って制御モデルの精度を上げなくてはならないと考える。また、今回のコンテストで上位となったチームは、実践に近い実験が行える環境を作っていたので、見習って環境を整えなければならない。加えて、今回のモデル設計で小型飛行船のシステム同定の難しさを痛感したため、ロバスト性を取り入れ、ある程度のモデル誤差に対応出来るような制御モデル設計が必要になると考える。

参考文献

- [1] 石川繁樹：モデル予測制御による飛行船システムの制御系設計に関する研究 (2005)
(http://www.fl.ctrl.titech.ac.jp/paper/2005/Thesis/M_shigeki_proc.pdf)
- [2] 井上和夫, 川田昌克, 西岡勝博：MATLAB/Simulink によるわかりやすい制御工学, 森北出版 (2001)
- [3] MDD ロボットチャレンジ 2009 実行委員会：競技仕様書, 情報処理学会 (2009)