

目 次

まえがき

1. Pythonの設定と基本操作

1.1 Pythonの環境設定	3
1.2 Pythonの基本操作	8
1.3 Pythonのモジュール	11
1.4 Python-Controlの設定	19
1章のポイント	21
章末問題	21

2. 状態空間表現と伝達関数表現

2.1 状態空間表現	25
2.2 状態空間表現と伝達関数表現の関係	31
2.3 状態変数線図とブロック線図	38
2.4 可制御性	44
2.5 可観測性	51
2.6 非線形システムの線形化	54
2章のポイント	57
章末問題	57

3. 状態変換と正準形式

3.1 状態変換	63
3.2 対角正準形式	68
3.3 可制御正準形式	76
3.4 可観測正準形式	82
3章のポイント	87
章末問題	87

4. システムの応答と安定性

4.1	状態方程式の解	91
4.2	状態空間表現での応答計算	95
4.3	一次遅れ系の応答	101
4.4	二次遅れ系の応答	109
4.5	安定性	116
	4章のポイント	120
	章末問題	120

5. 時間応答に着目した制御系設計

5.1	時間応答に着目した制御性能	125
5.2	極と応答の関係	127
5.3	状態フィードバック制御と極配置	134
5.4	アッカーマンの方法による極配置	140
5.5	オブザーバ	142
5.6	オブザーバを併合したレギュレータ	146
5.7	最適レギュレータ	150
5.8	ロバスト安定度指定法	154
5.9	最適1型サーボ系	158
5.10	2自由度制御	170
	5章のポイント	180
	章末問題	180

6. 周波数応答に着目した制御系設計

6.1	目標値追従性と積分要素	185
6.2	外乱抑制性	192
6.3	感度関数と目標値追従性	199

6.4	ロバスト安定性と相補感度関数	205
6.5	\mathcal{H}_∞ 制御	212
	6章のポイント	223
	章末問題	223

7. デジタル制御

7.1	デジタル制御系の概要	227
7.2	デジタル再設計による制御器の離散化	231
7.3	離散時間系の応答と安定性	242
	7章のポイント	255
	章末問題	255

8. 平行二輪車の制御

8.1	平行二輪車の概要	259
8.2	運動方程式の導出	261
8.3	近似線形化による状態空間表現の導出	263
8.4	最適レギュレータの設計	267
8.5	制御対象の離散化と離散時間最適レギュレータ	270
	8章のポイント	274
	章末問題	274

付録

1	アッカーマンの方法の導出	279
2	本文中で省略した Python スクリプト	283
3	本書で使用した関数のリスト	290