

ま え が き

実時間で非線形最適制御問題を解いて制御入力を決めるフィードバック制御手法は、実現できれば適用範囲がきわめて広いものの、多大な計算量のため、前世紀には不可能とされていた。しかし、今世紀に入ると、コンピュータの急速な発展により実現の可能性が高まり、実時間最適化に特化した数値解法が活発に研究されはじめた。そして、近年、実時間最適化によるフィードバック制御が実問題に適用されつつあり、幅広い分野への波及効果が期待される。

本書は、実時間最適化による制御として特にモデル予測制御 (receding horizon 制御) を取り上げ、その問題設定と数値計算アルゴリズム、プログラミングツールを概説し、自動操船システムにおける実用化事例、航空機誘導の飛行実験事例の他、自動車、ロボットなどへの適用事例を解説する。特に、制御目的をどのように最適化問題として定式化するか、計算環境と計算時間、評価関数の調整方法など、実応用におけるポイントに重点を置き、実問題に携わる技術者にとって有用な情報を提供する。さらに、偏微分方程式で記述される熱流体システムへの適用、他の問題設定など、今後の発展の可能性についても解説し、研究者に研究動向の情報を提供する。

本書は、大塚敏之「非線形最適制御入門 (システム制御工学シリーズ)」コロナ社 (2011) の発展編と位置づけられるが、ある程度の数学に関する知識 (ベクトルと行列、多変数の微積分、常微分方程式) があれば単独で読み進められるよう、最適制御と実時間最適化アルゴリズムについても概説している。各章の担当は以下のとおりである。

- 1 章, 2 章: 大塚 敏之 (京都大学)
- 3 章: 浜松 正典 (川崎重工業株式会社)
- 4 章: 永塚 満 (川崎重工業株式会社)

5 章：川邊 武俊（九州大学），向井 正和（工学院大学），M.A.S. カマル（株式会社豊田中央研究所）

6 章：西羅 光（日産自動車株式会社）

7 章：山北 昌毅（東京工業大学），李 俊黙（Hyundai Autron 株式会社）

8 章：橋本 智昭（大阪大学）

9 章：大塚 敏之（京都大学）

本書で紹介する制御手法は数値計算に大きく依存しており，その性質上，どのような問題でも確実に解けるとは限らない。適用のノウハウについてできる限り述べるよう努めたが，産業応用においては，適用する問題との相性や問題固有の工夫など，いくつかのハードルは避けられないと思われる。とはいえ，一昔前には想像もできなかったほど複雑な問題が現在では実時間で解けるようになりつつあり，今後，実時間最適化の適用範囲はますます広がっていくと考えられる。本書に記した適用事例が技術者を鼓舞し，実時間最適化に限らず新しい手法への挑戦によって価値を生み出す一助になれば，執筆者一同にとって大きな喜びである。

末筆ながら，本書の実現に関わったすべての方々に感謝の意を表したい。まず，各章で紹介する事例は，各執筆者の学生や同僚をはじめとする関係者の貢献があってこそ実現したものである。特に，2章で紹介する自動コード生成システム（Maple 版）は，サイバネットシステム（株）の石塚真一氏，松永奈美氏，郭 蕾氏によって開発された。編著者の要望を実現してくださったことに深く感謝したい。また，コロナ社には本書の企画実現にご支援をいただいた。京都大学の河野 佑氏と大阪大学の湯野剛史氏には草稿に対して貴重なコメントをいただいた。そして，執筆者各位には，本書の企画に快く賛同しご寄稿いただくとともに，編著者のさまざまなお願いと作業遅れにも寛容に対応していただいた。ここに記して心からお礼申し上げたい。

2014 年 11 月

編著者しるす

目 次

1. 問題設定とアルゴリズム

1.1 本章の概要	1
1.2 最適制御問題	1
1.3 停留条件	4
1.4 モデル予測制御	6
1.5 実時間最適化アルゴリズム	10
1.5.1 フィードバック制御における最適化の特徴	10
1.5.2 離散化された停留条件	11
1.5.3 時間変化する解の追跡	13
1.5.4 連立1次方程式の解法	15
1.5.5 実時間最適化アルゴリズムのまとめ	18
1.6 本章のまとめ	20
引用・参考文献	20

2. 自動コード生成

2.1 本章の概要	22
2.2 Maple版 AutoGenU	24
2.2.1 概要	24
2.2.2 各セクションの機能	28
2.2.3 Cソースファイル	50

2.3 Mathematica 版 AutoGenU	54
2.3.1 概 要	54
2.3.2 入力ファイル	56
2.3.3 ノートブック AutoGenU.nb	59
2.3.4 プログラムのコンパイルと実行	63
2.3.5 シミュレーション結果のグラフ描画	63
2.4 パラメータの調整	65
2.4.1 C/GMRES 法のパラメータ	65
2.4.2 評価関数の調整	67
2.5 本章のまとめ	70
引用・参考文献	70

3. 自動操船システム

3.1 本章の概要	72
3.2 システムの概要	74
3.3 モデルと評価関数	78
3.3.1 状態推定オブザーバと船体位置制御	78
3.3.2 推力配分	85
3.4 適用結果	87
3.5 本章のまとめ	92
引用・参考文献	94

4. 航空機の衝突回避

4.1 本章の概要	95
4.2 システムの概要	96

4.3	モデルと評価関数	97
4.3.1	領域回避誘導則の制御目的	97
4.3.2	ロール角制御	97
4.3.3	速度制御	98
4.3.4	ピッチ角制御	100
4.4	適用結果	100
4.4.1	MuPAL- α を用いた飛行実証	100
4.4.2	飛行実験結果	103
4.4.3	シミュレーション結果	110
4.5	本章のまとめ	113
	引用・参考文献	113

5. 自動車の省燃費運転

5.1	本章の概要	115
5.2	道路情報予測を用いた自動車の省燃費運転	117
5.2.1	車両モデル	117
5.2.2	燃料消費モデル	118
5.2.3	評価関数と拘束条件	119
5.2.4	適用結果	120
5.3	道路勾配情報を用いた自動車の省燃費運転	124
5.3.1	車両モデル	124
5.3.2	燃費モデル	125
5.3.3	評価関数と拘束条件	128
5.3.4	適用結果	128
5.4	先行車停止挙動予測を用いた自動車の省燃費運転	136
5.4.1	モデルと評価関数	136

5.4.2 適用結果	140
5.5 本章のまとめ	144
引用・参考文献	145

6. 自動車の経路生成

6.1 本章の概要	146
6.2 障害物操舵回避支援システムの概要	148
6.2.1 研究の背景	148
6.2.2 システムのコンセプト	149
6.2.3 検証用システムのハードウェア構成	150
6.3 実時間最適化問題としての定式化	151
6.3.1 問題設定	151
6.3.2 システムモデル	152
6.3.3 評価関数	156
6.4 実時間最適化アルゴリズムの改良	159
6.4.1 障害物回避支援システムにおけるモデル予測制御系の構成	159
6.4.2 C/GMRES法を適用した際に遭遇した課題	160
6.4.3 GMRES法を取り入れたニュートン法	162
6.4.4 ヤコビ行列の正則化を加えたアルゴリズム	164
6.5 アルゴリズムの動作検証結果	167
6.5.1 ニュートン法の検証と反復演算回数設計	167
6.5.2 ヤコビ行列正則化の効果検証	168
6.5.3 車両実験結果	170
6.6 本章のまとめ	172
引用・参考文献	173

7. 衝突現象を含むロボットの制御

7.1 本章の概要	174
7.2 状態ジャンプを含むモデル予測制御問題	174
7.2.1 モデル予測制御問題	174
7.2.2 状態ジャンプを伴う拘束条件付き非線形モデル予測制御の最適性条件	175
7.2.3 状態ジャンプを伴う拘束条件付き非線形モデル予測制御問題に対する離散化された最適性条件	177
7.3 状態ジャンプを伴う拘束条件付きモデル予測制御に対する C/GMRES 法の拡張	179
7.3.1 制御入力系列の更新方法	180
7.3.2 前進差分近似における問題	182
7.4 状態数変化を伴うシステムの切替え	183
7.4.1 積分器による制御入力の数値的安定化	183
7.4.2 制御入力安定化と入力拘束導入のためのシステムの切替え	185
7.5 適用結果	187
7.5.1 着地制御問題	187
7.5.2 シミュレーション結果	191
7.6 本章のまとめ	198
引用・参考文献	198

8. 熱流体システムの制御

8.1 本章の概要	200
8.2 熱流体システムの概要	200

8.3 熱流体システムの最適制御問題	203
8.4 停留条件の数値解法	210
8.4.1 数値流体計算とSMAC法	211
8.4.2 最適解の更新と縮小写像法	215
8.5 数値シミュレーション	222
8.6 本章のまとめ	228
引用・参考文献	229

9. 他の応用と展開

9.1 本章の概要	230
9.2 さまざまな応用	230
9.2.1 非線形機械システム	230
9.2.2 複雑な非線形システム	231
9.2.3 大規模な非線形システム	232
9.3 実時間最適化の展開	234
9.3.1 アルゴリズムの改良	234
9.3.2 問題設定の拡張	235
9.4 本章のまとめ	238
引用・参考文献	238

索引	242
----	-----

1

問題設定とアルゴリズム

1.1 本章の概要

本書では、**実時間最適化** (real-time optimization) による制御の手法として、**モデル予測制御** (model predictive control, **MPC**) を扱う。モデル予測制御とは、各時刻で有限時間未来までの**最適制御問題** (optimal control problem) を解いて制御入力を決定するフィードバック制御手法であり、最適制御問題を十分速く数値的に解くことさえできれば、さまざまなシステムに対してさまざまな目的や拘束を考慮した制御が行える。モデル予測制御を適切に設計するには最適制御問題の知識が必要であり、モデル予測制御を実現するには最適制御問題をフィードバック制御のサンプリング周期内に解くための数値計算アルゴリズムが必要である。そこで、本章では、まず最初に最適制御の問題設定と最適制御が満たすべき条件を概説する。続いて、モデル予測制御の問題設定を述べ、モデル予測制御を実現するための実時間最適化アルゴリズムについて概説する。

1.2 最適制御問題

最適制御問題とは、**ダイナミカルシステム**が何らかの意味で最適な応答を達成するよう制御入力 (操作量) を決定する問題である。一般に、ダイナミカルシステムは、次のような**状態方程式** (state equation) で表される。

$$\dot{x}(t) = f(x(t), u(t), p(t)) \quad (1.1)$$

ここで、 $x(t)$ は状態ベクトルと呼ばれ、システムの振る舞いを表す変数（状態変数）を要素に持つベクトルである。また、 $u(t)$ は制御入力ベクトルと呼ばれ、システムに対する制御入力（操作量）を要素に持つベクトルである。さらに、 $p(t)$ はシステムに含まれる時変パラメータのベクトルである。システムの応答を最適化するという目的は、しばしば次のような評価関数（performance index）を最小化することに帰着される。

$$J = \varphi(x(t_f), p(t_f)) + \int_{t_0}^{t_f} L(x(t), u(t), p(t)) dt \quad (1.2)$$

ここで、 t_0 は制御を開始する初期時刻、 t_f は制御を終了する終端時刻であり、初期時刻から終端時刻までの時間区間を評価区間（horizon）という。評価関数におけるスカラー値関数 φ と L を適切に設定することで、さまざまな目的を表現できる。例えば、状態が追従すべき参照軌道を時変パラメータ $p(t)$ に含めれば、軌道追従問題を扱うこともできる。制御にあたっては、システムの初期状態 $x(t_0)$ が与えられている場合や、終端時刻において到達すべき終端状態 $x(t_f)$ が与えられている場合などがある。より一般には、初期時刻において成り立つべき初期条件 $\chi(x(t_0), p(t_0)) = 0$ や終端時刻において成り立つべき終端条件 $\psi(x(t_f), p(t_f)) = 0$ が課される場合もある。

状態方程式に加えて、制御を行うすべての時刻において守らなければならない条件が、拘束条件（constraint）として

$$C(x(t), u(t), p(t)) = 0 \quad (t_0 \leq t \leq t_f) \quad (1.3)$$

と与えられることもある。ここで、 C は、0 に拘束されるべき関数を要素に持つベクトルである。なお、拘束条件が等式ではなく不等式 $C(x(t), u(t), p(t)) \leq 0$ (C はスカラー) の場合は、仮想的な制御入力（ダミー入力） $v(t)$ を導入して、 $C(x(t), u(t), p(t)) + v^2(t) = 0$ という等価な等式拘束条件に変換することができる。もしくは、拘束条件が破られると発散するバリア関数 $-\log(-C(x(t), u(t), p(t)))$ や $-1/C(x(t), u(t), p(t))$ を評価関数に加える方法もある。

例 1.1 (セミアクティブダンパ) ばね質点ダンパシステムの振動を減衰係数によって制御することを考えよう。外力でなく減衰係数を（非負の範囲で）操作して振動を減衰させる装置をセミアクティブダンパという。セミアクティブダンパは、例えば自動車のサスペンションやビルの制振装置などに使われている。図 1.1 のように、質点の変位を $y(t)$ 、質点の質量を m 、ばねのばね係数を k とし、制御入力である減衰係数を $u(t)$ で表すと、運動方程式は

$$m\ddot{y}(t) + u(t)\dot{y} + ky(t) = 0 \quad (1.4)$$

となる。ここで、減衰係数の大きさには制約があり

$$0 \leq u(t) \leq u_{max} \quad (u_{max} > 0) \quad (1.5)$$

を満たさなければならないものとする。

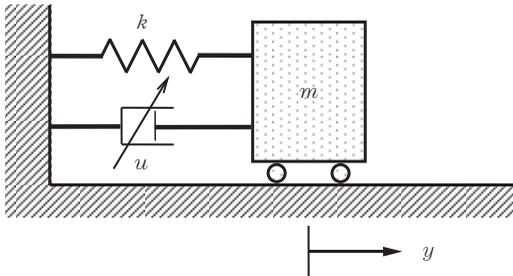


図 1.1 セミアクティブダンパ

状態ベクトルを $x = [x_1 \ x_2]^T = [y \ \dot{y}]^T$ と定義すると、運動方程式から次のような状態方程式が得られる。

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} x_2(t) \\ -\frac{k}{m}x_1(t) - \frac{1}{m}x_2(t)u(t) \end{bmatrix}$$

この式の右辺が、式 (1.1) の f に相当する。これは、状態変数 $x_2(t)$ と制御入力 $u(t)$ の積が含まれる非線形システムになっている。また、式 (1.5) の不等式拘束条件は

$$C(x(t), u(t), v(t)) = \left(u(t) - \frac{u_{max}}{2}\right)^2 + v^2(t) - \frac{u_{max}^2}{4} = 0$$

という等式拘束条件に書き替えることができる。上式は、 u - v 平面において、中心が $(u_{max}/2, 0)$ で半径が $u_{max}/2$ の円を表しており、その円周上の点 (u, v) は、必ず $0 \leq u \leq u_{max}$ を満たす。

制御の目的を、できるだけ小さい入力で速やかに変位を 0 に収束させることとすると、評価関数は、例えば

$$J = \frac{1}{2}x^T(t_f)S_f x(t_f) + \frac{1}{2} \int_{t_0}^{t_f} (x^T(t)Qx(t) + ru^2(t)) dt$$

のように選ぶことが考えられる。ここで、 S_f と Q は準正定な 2×2 行列、 r は正のスカラであり、いずれも重みと呼ばれる。2 次形式 $x^T(t_f)S_f x(t_f)$ と $x^T(t)Qx(t)$ は状態ベクトル x が原点からどれだけ離れているかを表し、 $ru^2(t)$ は入力の大きさを表している。変位、速度、入力それぞれの大きさをどれくらい重視するかに応じて重みを設定する。ただし、重みと応答の関係は定性的なものなので、最適制御による応答を見ながら重みを調整する必要がある。

1.3 停留条件

最適制御問題の評価関数 J は、状態 $x(t)$ と制御入力 $u(t)$ という時間関数によって値が決まる汎関数と見なすことができる。したがって、最適制御問題は、状態方程式と等式拘束条件の下で汎関数を最小化する変分問題になっている。例えば、初期状態 $x(t_0) = x_0$ が与えられていて終端状態 $x(t_f)$ が自由な場合、変分法によって汎関数の停留条件 (stationary condition) を求めると、最適制御が満たすべき必要条件が次のように得られる^{1), 2)}。

$$\dot{x}(t) = f(x(t), u(t), p(t)) \tag{1.6}$$

$$x(t_0) = x_0 \tag{1.7}$$

$$\dot{\lambda}(t) = - \left(\frac{\partial H}{\partial x} \right)^T (x(t), u(t), \lambda(t), \mu(t), p(t)) \quad (1.8)$$

$$\lambda(t_f) = \left(\frac{\partial \varphi}{\partial x} \right)^T (x(t_f), p(t_f)) \quad (1.9)$$

$$\frac{\partial H}{\partial u}(x(t), u(t), \lambda(t), \mu(t), p(t)) = 0 \quad (1.10)$$

$$C(x(t), u(t), p(t)) = 0 \quad (1.11)$$

ここで

$$H(x, u, \lambda, \mu, p) = L(x, u, p) + \lambda^T f(x, u, p) + \mu^T C(x, u, p)$$

はハミルトン関数 (Hamiltonian) と呼ばれるスカラー値関数であり, λ は随伴変数 (adjoint variables) または共状態 (costate) と呼ばれる変数のベクトル, μ は拘束条件 $C(x, u, p) = 0$ に対応するラグランジュ乗数のベクトルである。式 (1.6)~(1.11) はオイラー・ラグランジュ方程式 (Euler-Lagrange equations) と呼ばれ, 最適制御問題において基本的な条件である。また, 随伴変数の微分方程式 (1.8) を随伴方程式 (adjoint equation) という。

オイラー・ラグランジュ方程式 (1.6)~(1.11) を見ると, 式 (1.6), (1.7), (1.11) はもともと与えられた状態方程式, 初期条件, 拘束条件そのものである。式 (1.10) は制御入力 u と同じ数の代数方程式を与えており, 式 (1.11) と併せて, $(u(t), \mu(t))$ に対する代数方程式を与えている。したがって, 適切な問題設定がなされていれば, $(u(t), \mu(t))$ は $(x(t), \lambda(t), p(t))$ によって定まる。その結果, 式 (1.6), (1.8) は $(x(t), \lambda(t))$ の連立常微分方程式になる。ただし, $x(t)$ に対しては初期条件 (1.7) が与えられているのに対し, $\lambda(t)$ に対しては終端条件 (1.9) が与えられているため, いわゆる **2点境界値問題** (two-point boundary-value problem) になっている。常微分方程式の初期値問題であればルンゲ・クッタ法などさまざまな数値解法によって解くことができるが, 2点境界値問題の場合は, 初期条件と終端条件の両方を満たすような解を何らかの方法で探索する必要がある。そのため, 最適制御問題の数値解法は複雑となり, 計算量も多くなる。最適制御問題の数値解法としては, 例えば, 勾配法やニュートン法がある。いずれも解の修

正を反復する方法であり、コンピュータの急速な進歩をもってしても、与えられた初期状態に対する最適制御を即座に計算することは一般に困難である。また、数値解法によって求められる最適制御は時刻 t の関数であり、フィードフォワード制御になる。したがって、システムの状態に応じて制御入力を決めるフィードバック制御に比べて外乱やモデル誤差の影響を受けやすい。仮に、すべての状態と時刻に対する最適制御をあらかじめ計算して保存しておけば、状態フィードバック制御が実現できるが、状態の次元が大きい場合には膨大な計算量と記憶量が必要となり現実的ではない。

以上で述べた停留条件や数値解法の詳細については、最適制御に関する成書^{1),2)}を参照されたい。現実的な最適制御問題の例については、文献^{3),4)}が参考になる。

1.4 モデル予測制御

式 (1.2) の評価関数を最小にする最適制御問題は、たとえわずかな計算量で解けるとしても、フィードバック制御の用途には向かない。なぜなら、フィードバック制御は、通常、継続的に行われるものであって、開始時刻や終了時刻が決まっていないのに対し、初期時刻と終端時刻の固定された最適制御問題では評価区間以外の時刻における制御入力が決まらないからである。最適制御問題によって継続的なフィードバック制御を実現するためには、しばしば無限評価区間が用いられるが、特殊な場合を除いて解くことができない。

そこで、有限な評価区間の最適制御問題を数値的に解くことで継続的なフィードバック制御を実現する方法を考える必要がある。そのような問題設定として、有限な評価区間が時間とともに移動していく最適制御問題が考えられる。すなわち、各時刻 t における評価関数が

$$J = \varphi(x(t+T), p(t+T)) + \int_t^{t+T} L(x(\tau), u(\tau), p(\tau)) d\tau \quad (1.12)$$

で与えられる最適制御問題である。ここで、 T はどれだけ未来までを最適化する

かを表す評価区間長さである。この評価関数を最小にする最適制御 $u_{opt}(\tau)$ は、現在時刻 t から有限時間未来 $t+T$ までの評価区間[†]にわたる時間関数として定められ、その時間関数は現在の時刻 t と状態 $x(t)$ にも依存するので、 $u_{opt}(\tau; t, x(t))$ と表せる。したがって、各時刻における実際の制御入力 $u(t)$ を、その時刻で数値的に求めた最適制御の初期値によって

$$u(t) = u_{opt}(t; t, x(t)) \quad (1.13)$$

と与えることにすれば、状態フィードバック制御が実現できることになる。図 1.2 のように、ある時刻 t_1 における最適化で決定された評価区間上の制御入力の時刻 t ($t_1 \leq t \leq t_1 + T$) における値 $u_{opt}(t; t_1, x(t_1))$ は、あくまでその時点での予測値であって、実際の制御入力である $u(t) = u_{opt}(t; t, x(t))$ とは一般に異なることに注意されたい。また、制御入力は、その時刻の状態 $x(t)$ から数値計算によって決定されるので、状態フィードバック制御則 $u_{opt}(t; t, x(t))$ は陽に求められない。これは、オンラインでの最適化を前提としない従来のフィードバック制御系との大きな違いである。このように、各時刻で有限時間未来までの応答を最適化してフィードバック制御を実現する手法を、モデル予測制御ないし **receding horizon** 制御という。

本書では、実時間最適化による制御としてこのモデル予測制御を扱う。なぜ

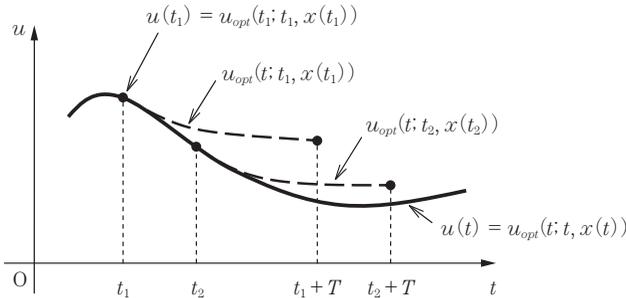


図 1.2 各時刻における有限時間未来までの最適化によって決まる制御入力

[†] モデル予測制御における評価区間を予測ホライズン (prediction horizon) ともいう。

索引

【あ】			
アジマススラスト	75	コルモゴロフの前向き 方程式	232
【う】		【さ】	
ウォームスタート	44	最適制御問題	1
運転支援	146	サンプリング周期	10
【え】		【し】	
エコロジカル ドライビング	115	実時間最適化	1
【お】		縮小写像法	218
オイラー・ラグランジュ 方程式	5	障害物回避経路	149
オブザーバ	237	状態推定	237
【か】		状態方程式	1
外乱オブザーバ	236	省燃費運転	115
カルマンフィルタ	237	【す】	
【き】		随伴変数	5
共状態	5	随伴方程式	5
【く】		推力配分機能	76
クリロフ部分空間	165	スタッガード格子	213
【け】		【せ】	
撃力	189	セミアクティブダンパ	3
【こ】		【そ】	
拘束条件	2	操舵反力トルク	149
コールドスタート	44	【た】	
		ダミー入力	2
		【つ】	
		追跡回避問題	236
		【て】	
		停留条件	4
		適応モデル予測制御	236
		デッドレコニング機能	77
		電動パワーステアリング	151
		【と】	
		特異値分解	235
		【な】	
		内点拘束条件	176
		ナッシュ均衡	237
		ナッシュ微分ゲーム 問題	237
		【に】	
		2点境界値問題	5
		入出力線形化	235
		ニュートン法	162
		ニュートン流体	200
		ニュートン・ GMRES法	164
		二輪モデル	153
		【は】	
		ハミルトン関数	5
		バリア関数	2
		【ひ】	
		非圧縮性流体	201
		飛行実験	103
		微分ゲーム問題	236
		評価関数	2

評価区間	2	ホモトピー法	14	【ら】	
【ふ】		【も】		ラグランジュ乗数	5
フィードバック制御	6	モデル予測制御	1	【り】	
フィードフォワード制御	6	モンテカルロ近似	233	粒子モデル予測制御	233
ブシネスク近似	201	【や】		【れ】	
【へ】		ヤコビ行列の正則化	164	連続変形法	14
変分原理	206	【よ】			
【ほ】		予測ホライズン	7		
ポテンシャルフィールド	156				

【A】		【G】		【R】	
AutoGenU	22	Gipps モデル	120	receding horizon 制御	7
【C】		GMRES 法	15	receding horizon ナッシュ	
condensing	234	【I】		微分ゲーム問題	237
C/GMRES 法	18	ILQ 設計法	235	receding horizon	
【D】		【M】		微分ゲーム問題	237
DPS	72	Magic Formula	153	【S】	
【E】		MaTX	191	SMAC 法	210
EPS	151	MHE	237	【T】	
【F】		moving horizon 推定	237	TCAS	95
FBW	100	MPC	1	【Z】	
		MuPAL- α	100	ZMP	194

—— 編著者・著者略歴および執筆分担(執筆順) ——

おおつか としゆき

大塚 敏之 (1章, 2章, 9章)
1990年 東京都立科学技術大学工学部航空宇宙システム工学科卒業
1992年 東京都立科学技術大学大学院工学研究科修士課程修了(力学系システム工学専攻)
1995年 東京都立科学技術大学大学院工学研究科博士課程修了(工学システム専攻)
博士(工学)
1995年 筑波大学講師
1999年 大阪大学講師
2003年 大阪大学助教授
2007年 大阪大学教授
2013年 京都大学教授
現在に至る

はままつ まさのり

浜松 正典 (3章)
1991年 大阪大学基礎工学部制御工学科卒業
1991年 川崎重工株式会社勤務
現在に至る

ながつか

みつる

永塚 満 (4章)
1995年 早稲田大学理工学部電気工学科卒業
1997年 早稲田大学大学院理工学研究科修士課程修了(電気工学専攻)
1997年 川崎重工株式会社勤務
現在に至る

かわべ たけとし

川邊 武俊 (5章)
1982年 早稲田大学理工学部応用物理学科卒業
1984年 早稲田大学大学院理工学研究科修士課程修了(物理学及応用物理学専攻)
1984年 日産自動車株式会社勤務
1994年 工学博士(東京大学)
2005年 九州大学大学院教授
現在に至る

むかい

まさかず

向井 正和 (5章)
2000年 金沢大学工学部電気・情報工学科卒業
2002年 金沢大学大学院自然科学研究科博士前期課程修了(電子情報システム専攻)
2005年 金沢大学大学院自然科学研究科博士後期課程修了(機能開発科学専攻)
博士(工学)
2005年 九州大学助手
2007年 九州大学助教
2014年 工学院大学准教授
現在に至る

モハマド アブドス

サマド

カマル

Md. Abdus Samad Kamal (5章)
1997年 バングラデシュ・クルナ工業大学電気電子工学科卒業
2003年 九州大学大学院システム情報工学修士課程修了(電気電子システム工学専攻)
2006年 九州大学大学院システム情報工学博士課程修了(電気電子システム工学専攻)
博士(学術)
2006年 マレーシア・国際イスラム大学助教授
2008年 (財)福岡県産業科学技術振興財団研究員
2011年 科学技術振興機構・東京大学生産技術研究所研究員
2014年 株式会社豊田中央研究所客員研究員
現在に至る

にしら ひかる
西羅 光 (6章)

- 1997年 東京大学工学部計数工学科卒業
- 1999年 東京大学大学院工学系研究科修士課程修了(計数工学専攻)
- 1999年 日産自動車株式会社勤務
- 2012年 日産自動車株式会社総合研究所モビリティ・サービス研究所主任研究員
現在に至る

やまきた まさき
山北 昌毅 (7章)

- 1984年 東京工業大学工学部制御工学科卒業
- 1986年 東京工業大学大学院理工学研究科修士課程修了(制御工学専攻)
- 1989年 東京工業大学大学院理工学研究科博士課程修了(制御工学専攻)
工学博士
- 1989年 東京工業大学助手
- 1993年 豊橋技術科学大学講師
- 1995年 東京工業大学助教授
- 2000年 東京工業大学大学院助教授
- 2007年 東京工業大学大学院准教授
現在に至る

い じゅんむく
李 俊黙 (7章)

- 2004年 東京工業大学工学部制御システム工学科卒業
- 2006年 東京工業大学大学院理工学研究科修士課程修了(機械制御システム専攻)
- 2006年 Samsung Electronics 株式会社勤務
- 2012年 Hyundai Autron 株式会社勤務
現在に至る

はしもと ともあき
橋本 智昭 (8章)

- 2003年 東京都立科学技術大学工学部航空宇宙システム工学科卒業
- 2004年 東京都立科学技術大学大学院工学研究科博士前期課程修了(航空宇宙工学専攻)
- 2007年 東京都立科学技術大学大学院工学研究科博士後期課程単位取得退学(航空宇宙工学専攻)
博士(工学)
- 2007年 理化学研究所研究補助員
- 2008年 信州大学助教
- 2009年 大阪大学助教
現在に至る

実時間最適化による制御の実応用

Practical Applications of Control by Real-Time Optimization

© Toshiyuki Ohtsuka 2015

2015年1月26日 初版第1刷発行

★

検印省略

編著者 大塚敏之
著者 浜松正典
永塚満
川邊武俊
向井正和
M. A. S. Kamal
西羅光
山北昌毅
李俊黙
橋本智昭
発行者 株式会社 コロナ社
代表者 牛来真也
印刷所 三美印刷株式会社

112-0011 東京都文京区千石 4-46-10

発行所 株式会社 コロナ社

CORONA PUBLISHING CO., LTD.

Tokyo Japan

振替 00140-8-14844・電話(03)3941-3131(代)

ホームページ <http://www.coronasha.co.jp>

ISBN 978-4-339-03210-9 (森岡) (製本:愛千製本所)

Printed in Japan



本書のコピー、スキャン、デジタル化等の無断複製・転載は著作権法上での例外を除き禁じられています。購入者以外の第三者による本書の電子データ化及び電子書籍化は、いかなる場合も認めておりません。

落丁・乱丁本はお取替えいたします