

モンテカルロモデル予測制御の非線形システム  
に対する性能向上と適用可能性に関する  
研究

2023 年 3 月

仲谷 真太郎

モンテカルロモデル予測制御の非線形システム  
に対する性能向上と適用可能性に関する  
研究

仲谷 真太郎

理工情報生命学術院  
システム情報工学研究群  
筑波大学

2023 年 3 月

# 目次

目次	III
第1章 序論	1
1.1 研究背景	1
1.2 本研究の研究課題	4
1.2.1 研究課題1：性能向上	5
1.2.1.1 サンプル型 MPC の最適化精度と近似誤差の影響	5
1.2.1.2 収束性改善手法を検討する必要性	9
1.2.1.3 研究課題1に対する本研究の取り組み	10
1.2.2 研究課題2：適用可能性について	11
1.3 本論文の構成	13
第2章 モンテカルロモデル予測制御の収束性解析	14
2.1 はじめに	14
2.2 モデル予測制御	15
2.3 モンテカルロモデル予測制御	16
2.3.1 入力列の生成	18
2.3.2 並列シミュレーション	19
2.3.3 推定値の計算	19
2.3.4 各種制約の処理要領	21
2.4 MCMPC の線形系における収束性解析	21
2.4.1 線形システムに対する有限時間最適制御問題	22
2.4.2 主結果:同一制御周期内反復による制御則の収束	23
2.5 数値例	26
2.6 2章のまとめ	27

第 3 章	各種冷却法を用いたモンテカルロモデル予測制御の収束特性改善	30
3.1	はじめに	30
3.2	幾何冷却法	31
3.3	双曲線型冷却法	31
3.4	非線形系への適用可能性	33
3.5	数値シミュレーション	34
3.5.1	検証内容	36
3.5.2	<b>Problem 1</b> での検証結果	36
3.5.3	<b>Problem 2</b> での検証結果	37
3.6	MCMPC のパラメータ設計指針	40
3.7	3 章のまとめ	42
第 4 章	分散共分散行列適応進化戦略を用いたサンプリング効率化	44
4.1	はじめに	44
4.2	$(\mu/\mu_w, \lambda)$ -CMA-ES 型分散共分散行列更新則	46
4.3	$(\mu/\mu_w, \lambda)$ -CMA-ES を用いた MCMPC	47
4.3.1	アルゴリズム	47
4.3.2	パラメータの設計指針	49
4.4	評価実験	49
4.4.1	最適制御問題の問題設定	50
4.4.2	結 果	51
4.5	4 章のまとめ	55
第 5 章	サンプルベースニュートン型最適化手法を用いた収束特性の改善	57
5.1	はじめに	57
5.2	Sample-based Newton-Like 法	58
5.3	並列黄金分割探索法	61
5.4	数値シミュレーション	62
5.4.1	ベンチマーク問題の問題設定	64
5.4.2	<b>Problem 5</b> への適用結果	68
5.4.3	<b>Problem 6</b> への適用結果	72
5.4.4	<b>Problem 7</b> への適用結果	72



5.4.5	<b>Problem 6</b> 及び <b>Problem 7</b> におけるサンプルベース・ニュートン型 最適化法が最適化に失敗した要因分析 . . . . .	74
5.5	5 章のまとめ . . . . .	74
<b>第 6 章</b>	<b>不連続現象を含む非線形システムへの適用</b>	<b>77</b>
6.1	はじめに . . . . .	77
6.2	MPC における最適制御問題の解法と本章例題との関係 . . . . .	78
6.3	衝突を含んだ台車型倒立振子の予測モデル . . . . .	81
6.4	検証要領 . . . . .	82
6.5	数値シミュレーション . . . . .	84
6.6	実機実験 . . . . .	85
6.7	6 章のまとめ . . . . .	87
<b>第 7 章</b>	<b>総 括</b>	<b>91</b>
7.1	本研究の成果 . . . . .	92
7.1.1	研究課題 1 : 「性能向上」に対する本研究の成果 . . . . .	92
7.1.2	研究課題 2 : 「適用可能性」に対する本研究の成果 . . . . .	94
7.2	今後の課題 . . . . .	94
	<b>謝 辞</b>	<b>97</b>
	<b>参考文献</b>	<b>99</b>
	<b>付 録 A 重み付き確率 (2.32) の期待値の導出</b>	<b>105</b>

# 第1章 序論

## 1.1. 研究背景

実世界に存在するすべての動的システムは、厳密には何かしらの非線形性を含んだ非線形システムである [1]。実際にそれらのシステムに制御技術を適用する際には、アクチュエータの性能や制御にかかるコストについての制約、システムや利用者の安全性確保に起因する制約など、さまざまな制約を考慮した上で制御系を設計する必要がある。近年、システムの非線形性や各種の制約を考慮した上で性能限界を追求したいという需要の高まりから、実時間非線形最適制御手法であるモデル予測制御 (model predictive control; MPC) が非常に注目されている [2,3]。

最適制御手法は、制御対象の制御量の挙動をその数理モデルなどを用いて予測し、制御量や操作量に係る各種の制約を満たす操作量の中から、現在時刻から予測区間 (horizon; ホライズン) 分だけ未来までの時刻における制御目的を定量的に表した制御評価関数が、最小となる操作量を決定する最適制御問題の解を、実際にシステムに投入する制御入力とする制御手法である。また、MPC は、与えられた最適制御問題を、予測区間をずらしながら繰り返し解き直す Moving Horizon Controller [4] の考え方に基づく最適制御手法であり、これにより、時々刻々と変化する状況に応じた制御を行うことができるという特長を有している。制御対象の非線形性、制御量や操作量に課せられる各種の制約を制御アルゴリズムの中で陽に考慮した制御が実現できる制御手法であるため、ロボティクス、自動車工学 [5]、航空宇宙工学 [6–8] などの幅広い分野への応用に関する研究が盛んに行なわれている。今後も、制御対象の性能限界を追求した制御に対する需要の増加が予想されており、MPC はますます重要な役割を担う制御技術になると考えられている [2,9]。

MPC を実現する上で、拘束条件付きの非線形最適化問題を実時間で解くことができる汎用的な最適化ソルバーの開発は、重要な研究課題の 1 つに位置付けられている [10,11]。表 1.1 に示す通り、これまでに数多くの MPC のための実時間最適化アルゴリズムが考案されている。また、それらの中には、さまざまな問題への実装手順を自動化したソフトウェアツールとして公開されているものも存在する [12–19]。

また、表 1.1 に示す通り MPC は、その最適化手法の違いにより決定論的最適化手法を用いる勾配型 MPC と確率・統計的最適化手法を用いるサンプル型 MPC の 2 種類に分類することができる [20]。これまでに公開されている MPC のためのソフトウェアツール [12–19] の多くは、決定論的最適化法を用いた勾配型 MPC である。決定論的最適化手法は、勾配法、シューティング法、ニュートン法、拡張ラグランジュ法、逐次 2 次計画法などのように、目的関数が減少するように、勾配やヘッセ行列などの微分情報を手ばかりとして、繰り返し解の修正を行う反復法を用いる最適化手法である。特定の条件下では、十分な反復のもと、所望の最適解に対して許容誤差の範囲に収まる解に収束することが保証されており、最適化の精度という観点では優れた手法である。一方、いざ実際の問題を解こうとすると、例えば、厳しい制約が課されているなど制御評価関数が高い非線形性を有している場合は実時間で反復が終了しない場合がある [9]。さらに、目的関数に微分不可や不連続となる点が存在する場合や目的関数の一部又は全部がブラックボックス化されている問題には、適用することができない。

これとは別に、表 1.1 右側にまとめた、モンテカルロ最適化のような確率・統計的最適化手法を用いて MPC を実現する方法も存在する。これらの確率・統計的最適化手法に基づく MPC は、サンプル型 MPC と呼ばれている [20]。Fig. 1.1 に示す通り、サンプル型 MPC のアイデア自体は MPC の誕生とほぼ同時期から存在するが、過大な計算量により、特に機械系などの短いサンプリング周期をもつシステムへの適用については、実現困難なアイデアに長く留まっていた。近年ようやく、並列演算を得意とするマルチコア CPU や graphics processing unit (GPU) などのデバイスの性能向上とそれらの普及に伴い、5 分の 1 サイズのラリーカー [21] やクアッドコプター [22,23] などの機械系への適用例が報告されるようになってきている。このことは、サンプル型 MPC そのものが、今まさに黎明期から発展期への転換期を迎えようとしている研究分野であり、そこには、多くの研究課題が残されている。そこで、本研究では、サンプル型 MPC に着目している。

確率・統計的最適化手法にもまた、モンテカルロ最適化法に限定しても、粒子群最適化 (particle swarm optimization; PSO) や焼きなまし法 (Simulated annealing; SA) などの探索をベースとする手法、マルコフ連鎖モンテカルロ (Markov chain Monte Carlo; MCMC) や経路積分最適制御法、Prior Feedback 法などの確率分布の期待値の近似をベースとする手法のように多数のバリエーションが存在する [24,25]。探索をベースとするアルゴリズムであれば、探索に十分な時間を掛ければ、幅広いクラスの問題において最適解への収束が保証されている [24]。期待値の近似を用いるアルゴリズムについても十分なサンプル

数と近似の精度が担保されれば、最適解の近似解を得ることができるとされている。しかしながら、これらの条件は、MPC に課される“実時間性”の制約とは相反しており、確率・統計的最適化手法を用いた MPC の最適化の精度は、勾配型 MPC に比べ劣るとされており、ひいては、MPC としての性能もまた劣ることになる。この最適化の精度の問題について、その原因を明らかにし、具体的な改善手法について検討することで、精度（性能）の面において勾配型 MPC に引けを取らないサンプル型 MPC の最適化アルゴリズムを構築するというのが本研究における 1 つ目の重要な研究課題である。

他方、これまでに多くの実システムへの適用例において、サンプル型 MPC が、制御性能の面で勾配型 MPC を上回った、あるいは、勾配型 MPC では困難とされていた課題に適用できた、という内容が報告されている [22,26–30]。これらの多くは、最適化の際に勾配やヘシアンなどの解析的な事前知識が不要であるという確率・統計的最適化手法の特長が効果を発揮した結果であると考えられる。また、この特長により確率・統計的最適化手法は、例えば、目的関数が部分的に不連続又は微分不可な場合や目的関数の一部又は全部がブラックボックス化されたような最適化問題へも適用可能であるため [24]、決定論的最適化手法を用いた勾配型 MPC より広範なクラスの制御対象に適用できる可能性がある。以上により、サンプル型 MPC はさまざまな分野においてブレイクスルーをもたらす重要な制御技術になることが期待できる。一方で、これまで行なわれてきたサンプル型 MPC に関する研究では、適用範囲を拡大するという観点での検証が十分行なわれていない。この点について、これまでの勾配型 MPC が適用困難あるいは不可とされていた問題に対して、サンプル型 MPC がどの程度効果を発揮するかについて、その適用可能性を明らかにしていくというのが本研究 2 つ目の重要な研究課題である。

本研究の目的は、上述した 2 つの研究課題について、改善が必要な点は、その改善手法を検討し、有効活用できる点についてはそれを最大限活用した、より実用的な MPC の制御アルゴリズムを構築することである。そこで、本研究では、後述する“探索型”と“近似型”の 2 つの側面を持ちあわせた確率・統計的最適化手法に基づくサンプル型 MPC であるモンテカルロモデル予測制御（Monte Carlo model predictive control; MCMPC）を対象とし、上述した研究課題に関する検証及び評価を行った。1 つ目の研究課題については、MCMPC を用いた際の収束性解析を行った上で、その結果に基づく収束特性改善手法を提案し、提案法の有効性を確認した。2 つ目の研究課題については、従来の勾配型 MPC が適用困難／不可な問題に対する MCMPC の適用可能性を、数値シミュレーション・実機での実験結果に基づき評価し、本研究で提案する方法が、従来の MPC より多く

の優れた性質をもつ、より実用的な制御アルゴリズムであることを確認した。

本章では、以降、1.2 節において、ここで示した 2 つの研究課題の詳細と本研究の目的との関係性や本論文で示す結果と各研究課題との対応関係について述べ、1.3 節に本論文全体の構成を記述する。

## 1.2. 本研究の研究課題

1.1 節で述べた通り、本研究の目的は、以下に示す研究課題に取り組み、そのうちの克服すべき課題について改善要領を検討し、有効利用できるものについてはそれを積極的に取り込んだ、より実用的な制御アルゴリズムを構築することである。

研究課題 1 性能向上（サンプル型 MPC の最適化の精度改善）

研究課題 2 適用可能性（勾配型 MPC が適用困難／不可な問題への適用可能性の検証）

そこで、本節では、1.1 節及び上に列挙した 2 つの研究課題についてその必要性を詳述するとともに、これらの研究課題の解決と研究目的である実用的な MPC の制御アルゴリズム構築との関係性について述べる。

Table 1.1. Classification of optimization method used in MPC and their characteristic.

		Gradient-based MPC	Sample-based MPC
Optimization method		Deterministic	Probabilistic / statistical
Gradient information		Required	Not required
Optimization performance	Multimodality	C	A
	Discontinuity	C	A
	Tight constraint	B	A
Affinity with parallel computing		B	A
Accuracy of optimization		A	C
Typical algorithms		<b>C/GMRES</b> [12,31] <b>MUSCOD-II</b> [13] <b>ACADO</b> [14] <b>FiOrdOs</b> [15] <b>VIAOC</b> [16] <b>ParNMPC</b> [17] <b>GRAMPC</b> [10] <b>CasADi</b> [19]	<b>MCMC-MPC</b> [32] <b>PF-MPC</b> [33] <b>PSO-MPC</b> [34] <b>MPPIC</b> [21,35,36] <b>MCMPC</b> [28,37] <b>IT-MPC</b> [27] <b>SA-MPC</b> [38]

<sup>A</sup> Excellent (Demonstrate excellent performance.)

<sup>B</sup> Good (Can be handled, but the performance is a slightly inferior)

<sup>C</sup> Poor (Difficult to handle / Significantly inferior performance.)

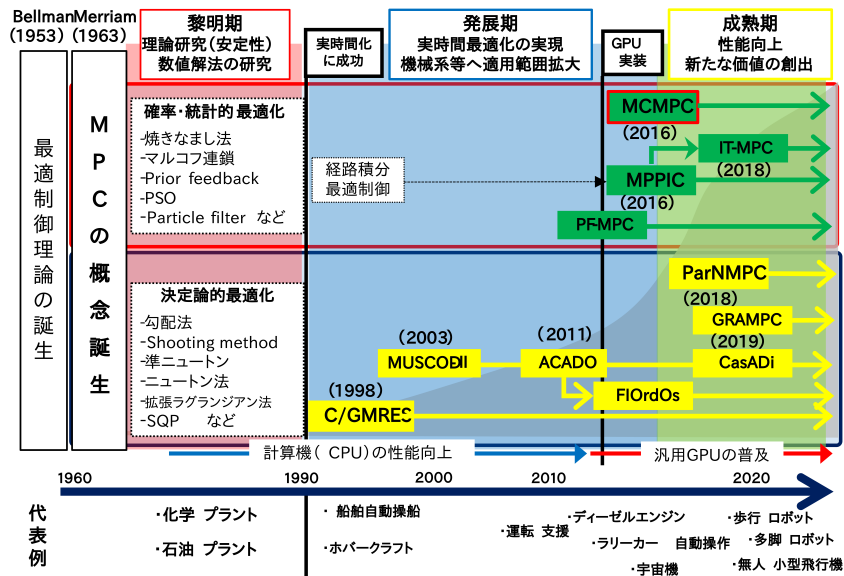


Fig. 1.1: Chronology of MPC Research History

### 1.2.1 研究課題 1：性能向上

本小節では、本研究の1つ目の研究課題である性能向上について記述する。はじめに、1.2.1.1において、MPCの性能と深い関わりのある最適化の精度について、確率・統計的最適化手法を用いる場合に留意する必要がある問題点について記述する。ついで、1.2.1.2において、1.2.1.1で記述した問題がサンプル型MPC（特に、MCMPC）を用いて制御を行う上での問題点について述べ、MCMPCの性能向上が必要不可欠であることを示す。さらに、本研究の取り組みについても記述する。

#### 1.2.1.1 サンプル型MPCの最適化精度と近似誤差の影響

MPCが実時間非線形最適制御手法に基づく制御手法であることは、既に、1.1節で述べた通りである。そのため、最適化に係る部分のアルゴリズムの収束特性、中でも収束速度や最適化の精度は、制御性能そのものに影響を与える重要な要素となる。

例えば、表1.1左側に示す勾配型MPCでよく用いられている、拡張ラグランジュ関数法に対する勾配法（[10,16]で使用）や逐次2次計画法（[14]で使用）、その他のニュートン型最適化法では、十分な反復のもとに、推定値が予め指定した許容誤差範囲に収まるような解に収束することが保証されている[9]。また、サンプル型MPCについても、推定値や粒子に確率的な摂動を与えながら“探索”するアプローチで最適化を行うSAやPSOであれば、前者の場合はアルゴリズム固有のパラメータである温度を適切に更新するこ

とで、後者の場合であれば、粒子間の引力や斥力に相当するパラメータを適切に更新することで、大域的な最適解に収束することが保証されている。

一方で、確率・統計的最適化手法に基づくサンプル型 MPC には、上述した“探索”型のアルゴリズムとは別に、最適解をモンテカルロ近似により“近似”する方法も存在する。代表的なアルゴリズムとしては、モデル予測経路積分制御 (Model predictive path integral control; MPPIC), Information-Theoretic model predictive control (IT-MPC) などが挙げられる。Williams らが提案した MPPIC [21,36] は、最適制御問題の解を計算する際に必要な偏導関数が特殊な重みをもつ確率分布 (目的分布) の期待値として表現できるという非線形確率最適制御と経路積分最適制御 [39] に着想を得た“確率システム”を対象とした手法である。MPPIC は、経路積分最適制御の考え方を取り込む上で、制御対象や評価関数に特別な条件が必要である。IT-MPC [27] は、目的分布の重みの付け方を工夫することで、MPPIC をより一般的な制御対象や評価関数をもつ最適制御問題に適用できるように拡張したものである。

本稿で着目している MCMPC は、“確定システム”を対象とした手法であるが、モンテカルロ近似により目的分布を“近似”し、その分布のピークとなる点 (最適解) を“探索”するという発想に基づく最適化手法を用いたサンプル型 MPC である。つまり、MCMPC については、基本的な戦略は“近似”に基づいているものの、“探索”型の性質も持ちあわせた手法であると言える。

ここで、“探索”型 [40,41]、“近似”型 [42–44] 双方の多くの確率・統計的最適化手法の収束性解析に用いられる Pincus の定理 [45, Theorem] について触れておく。

**定理 1 ([45, Theorem]).**  $S$  を  $n$  次元ユークリッド空間  $\mathbb{R}^n$  の有界の領域の閉包とし、 $F(\mathbf{X})$  を  $S$  上で連続な関数とする。また、 $\mathbb{R}^n$  上の点  $\mathbf{X} = (x_1, \dots, x_n)$  に対して  $F(\mathbf{X})$  は唯一の点  $\mathbf{X}^*$  において大域的な最小値をとるとする。この時、

$$x_i^* = \lim_{\eta \rightarrow 0} \frac{\int_S x_i \exp\left(\frac{-F(\mathbf{X})}{\eta}\right) d\mathbf{X}}{\int_S \exp\left(\frac{-F(\mathbf{X})}{\eta}\right) d\mathbf{X}}. \quad (i = 1, \dots, n) \quad (1.1)$$

定理中の式 (1.1) は、極限の操作を除けば、MPPIC や IT-MPC で用いられている期待値演算と重みの付け方やその重みの下での重み付き期待値であるという点で一致している。また、図 1.2 は、上で示した Pincus の定理 [45, Theorem] の内容を簡単な為 1 次元で図示したものである。図 1.2 に示す通り、この定理の基本的なアイデアは、 $\exp(\cdot)$  の重みが、 $\eta \rightarrow 0$  となる極限でデルタ関数  $\delta(\mathbf{X} - \mathbf{X}^*)$  となる性質を利用したものである。

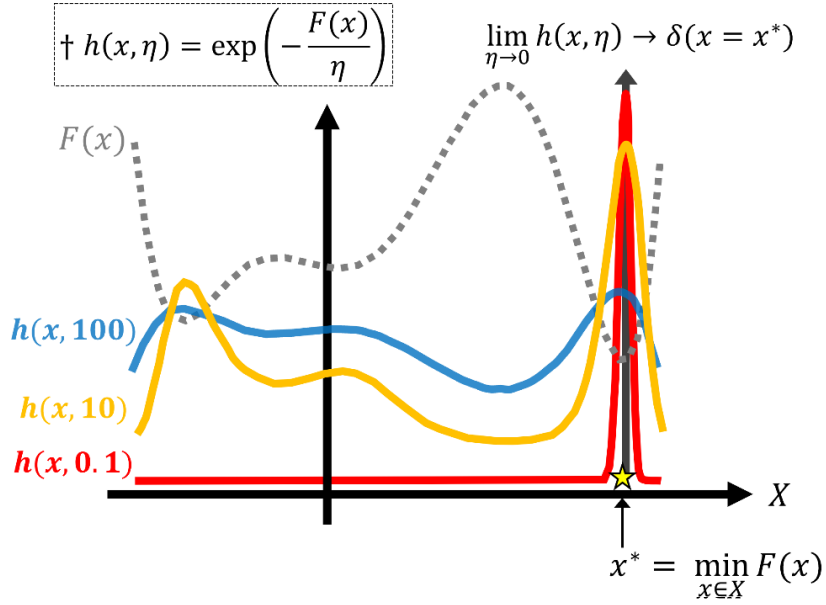


Fig. 1.2: Schematic diagram of Pincus' theorem [45, Theorem]

SA [38,40] や MCMC [32] などの“探索”型の最適化では、図 1.2 中の  $h(x, \cdot)$  で示す目的分布の密度にそって採択率を定め、温度パラメータに対応する  $\eta$  を徐々に絞り込みながら逐次的にサンプルを生成することで、極限状態に近づくにつれ  $\mathbf{X}^*$  の周辺のサンプルが生成・採用されやすくするアルゴリズムである。これらの“探索”型の最適化手法は、勾配型 MPC が処理困難な最適化問題においても収束性が保証されている反面、その収束速度が非常に緩慢であることが知られており、我々が想定する機械系などの非常に短いサンプリング周期を持つ制御対象には不向きな手法である。

一方、SA や MCMC のように逐次的にサンプルを生成するのとは別に、同時並列的に生成した有限個のサンプルを用いて (1.1) 式をモンテカルロ近似する方法を採用しているのが MCMPC [28,37,46] や Population Monte Carlo [44], Dynamic Importance Sampling [42] である。これらの手法は、“近似”型の性質も併せ持っているので、その最適化性能は、モンテカルロ近似の精度に大きく影響を受けることになる。

図 1.3 は、“近似”型の確率・統計的最適化手法を用いる MCMPC の最適化要領を簡略化して図示したものである。図に示す通り、推定値と最適解の間に誤差があることが分かる。この誤差には、いくつかの異なる要因が内在している。

1 つ目の要因は、提案分布 (図 1.3 中、青色実線で示す分布) と目的分布 (図 1.3 中、緑色実線で示す分布) が乖離している (2 つの分布間の KL 情報量が大きい。) ことにより生じる誤差である。この誤差については、2 つの分布間の密度比をキャンセルする操作である重点サンプリングなどのサンプリング手法を用いることである程度改善すること



ができる。

2つ目の要因は、重み付き期待値の形で与えられる式 (1.1) 右辺を有限個のサンプルを用いた以下に示す標本平均で近似していることによるものである。

$$\frac{\int_{\mathbf{s}} x_i \exp\left(-\frac{F(\mathbf{X})}{\eta}\right) d\mathbf{X}}{\int_{\mathbf{s}} \exp\left(-\frac{F(\mathbf{X})}{\eta}\right) d\mathbf{X}} \approx \frac{\sum_{\mathbf{X}} x_i \exp\left(-\frac{F(\mathbf{X})}{\eta}\right)}{\sum_{\mathbf{X}} \exp\left(-\frac{F(\mathbf{X})}{\eta}\right)} = \tilde{x}_i, \quad (i = 1, \dots, n) \quad (1.2)$$

これにより、推定値である  $\tilde{\mathbf{X}}$  は確率変数となる。つまり、同一アルゴリズムで同一の初期値から最適化を行ったとしても、(1.2) 右辺の演算に用いるサンプルが異なれば結果にばらつきが生じることを意味している。これに起因する誤差は、ガウス雑音とみなすことができ、中心極限定理により、サンプル数  $N$  を増加させてもその収束が  $O(1/\sqrt{N})$  と緩慢であることが分かる。また、この誤差を完全に排除するには、理論上、無限個のサンプルが必要になるが、その要求が非現実的なものであることは明らかである。

3つ目の要因は、コンピュータを用いて実装する上での有効桁数の問題である。これは、 $\eta$  がある値より小さくなると、 $\exp(-\frac{F(\mathbf{X})}{\eta})$  が有効桁の範囲で 0 となり、(1.2) において適切な演算が行えなくなる問題のことである。そこで、実装上は、少なくとも 1 つ以上のサンプルの重み  $\exp(-\frac{F(\mathbf{X})}{\eta})$  が有効な値を持つ範囲で  $\eta$  を決定する必要がある。これにより、興味の対象である目的分布が最大値をとる点と (1.1) で与えられる点が一致しないケースも想定される。

本研究における 1 つ目の研究課題の狙いは、上に列挙したような要因により発生するサンプル型 MPC、中でも MCMPC の最適化時に発生する誤差の影響を軽減する方法について検討し、その最適化の精度を向上することで制御性能を向上させることである。

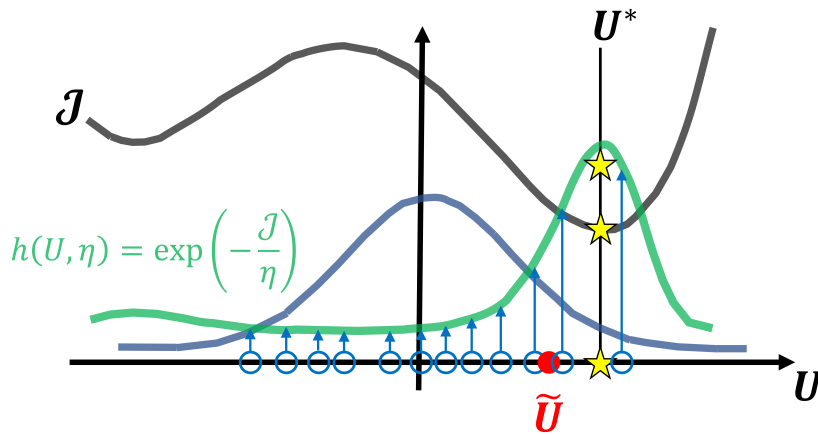


Fig. 1.3: Schematic diagram of the optimization principle of probabilistic/statistical optimization methods

### 1.2.1.2 収束性改善手法を検討する必要性

1.2.1.1 では、確率・統計的最適化手法に基づくサンプル型 MPC の中でも、特に、“近似” 型の最適化手法を用いるタイプの MPC について考える時に、その最適化アルゴリズム由来の誤差の影響について考慮する必要性とその誤差の要因について簡潔に記述した。1.2.1.1 で指摘した誤差は、確定論的最適化手法における収束条件判定に用いられる許容誤差の範囲に収まる場合もあれば、システムを制御する上で好ましくない影響を与える場合もある。例えば、この誤差が原因で、実際にシステムに投入する入力の小刻みに振動するガウス雑音のような入力となる [27,29,36,47] ことが知られている。そのような入力は、定常状態における整定性の悪化や入力として消費されるエネルギーの増加、アクチュエータの劣化を早めるなど、制御にとって好ましくない影響を与えることになる。このような問題に対処するためにも、確率・統計的最適化手法に基づく最適化アルゴリズム由来の誤差を軽減する収束性改善手法について検討する必要がある。

以下では、1.2.1.1 で示した誤差を軽減する方法として、関連研究で採られている方法について簡単に記述する。1.2.1.1 で示した 1 つ目の要因に対する対策としてよく用いられるのは、重点サンプリング [27,48,49] や MCMC のように、サンプリングが容易な適当な提案分布から抽出したサンプルが目的分布からのサンプルとみなせるように、サンプルの重みや採択率を調整する方法である。しかしながら、これらの手法は、目的分布と提案分布の形状が乖離している（2 つの分布間の KL 情報量 [50] が大きくなる）場合には十分な効果が期待できないという問題がある [43]。また、これとは別の方法で、1.2.1.1 で示した 1 つ目及び 2 つ目の双方の要因を解決し得る方法として、佐藤ら [51] のように特殊な反復規則を用いて、目的分布と提案分布の乖離により生じる誤差（要因 1）と標本平均で近似していることにより生じる誤差（要因 2）の双方を逐次的に減少させる方法も提案されている。

本研究で着目している MCMPC においても、“同一制御周期内反復” と呼ばれる反復規則 [28,37] が用いられている。同一制御周期内反復は、提案分布に係る各種のパラメータのうちユーザーが自由に変更可能なパラメータを反復の都度、逐次的に更新することで、近似精度を向上させようという考え方に基づく手法である。しかしながら、具体的にどのパラメータをどのように更新すれば、近似誤差を軽減できるのかについては明らかにされていないという問題がある。

関連研究では、上記のような反復規則を用いる方法以外の方法もいくつか検討されて

いるので、簡単に紹介する。Homanga らの提案した MPC は、大域的な最適化に確率・統計的最適化手法であるクロスエントロピー法を、局所的な最適化に決定論的最適化手法である最急降下法を利用している [52]。同様のアプローチは、多点局所探索法 [53,54] とも呼ばれ、解析的な勾配やヘシアンを計算するのは比較的容易なものの目的関数の多峰性などにより、局所解に陥ってしまうようなケースにおいて有力な最適化手法として知られている。実際に Homanga らの例でも、決定論的最適化手法のみを用いた MPC との性能比較でより優れた結果を残したと報告されている。しかしながら、この手法は、勾配やヘシアンが解析的に計算できることが必須要件となる。この要件は、目的関数の一部または全部がブラックボックス化された最適化問題への適用可能性というサンプル型 MPC の特長と相反するため、適用範囲がやや限定されるという問題がある。

他の取り組みとしては、実際にシステムに投入される入力に小刻みに振動する非常にノイズの入力となるという問題にのみ焦点をあてた手法である Savitzky-Golay filter [55] のようなフィルターを取り入れ、入力を平滑化する方法がよく用いられている [27,36,47,56]。入力が振動的になるという問題に対してはある程度効果を発揮するが、そういった現象を完全には排除出来ない。さらに、平滑化の前後で最適性を保証する枠組みがないなどの問題がある。

以上を踏まえると、1.2.1.1 で示した誤差が制御にとって好ましくない影響を与える問題に対して、既にいくつかの先行研究が存在するものの、いずれのアプローチにおいても改善の余地が多大にあると言える。そこで本研究では、収束性改善手法が満たすべき要件として、次に示すような要件を掲げた。

- 1) 実時間実装可能な範囲で十分な効果が期待できる。
- 2) 改善手法のアルゴリズムが、サンプル型 MPC の特長に干渉しない。
- 3) 改善手法適用後の結果が、適用前の結果より評価値で劣ることがない。

#### 1.2.1.3 研究課題 1 に対する本研究の取り組み

ここまで、1.2.1.1 では、確率・統計的最適化手法において、最適化の際に発生するそのアルゴリズム由来の誤差の問題と具体的な要因について述べ、1.2.1.2 では、そういった誤差が実際に制御器として利用する際に及ぼす負の影響と関連研究での取り組みを紹介した。

ここまでの考察の内容を踏まえ、本小々節では、本研究における研究課題1に対する具体的な取り組み方について記述する。本研究では、確率・統計的最適化手法のみを用いる方法と、確率・統計的最適化と決定論的最適化手法双方の性質を持ち合わせた手法を用いる方法の2つのアプローチに基づき、以下に示す3つの解決策を提案し、それぞれの有効性の検証を行なった。

- 幾何・双曲線型冷却法を用いた収束特性改善手法（3章）
- 分散共分散行列適応進化戦略を用いた収束特性改善手法（4章）
- サンプルベース・ニュートン型最適化法を用いた収束特性改善手法（5章）

上に示した3つのうち、3章及び4章で示す内容が、確率・統計的最適化手法のみを用いる方法に対応している。これについては、1.2.1.2で挙げた未解決課題の1つである、同一制御周期内反復におけるMCMPCの提案分布に係るパラメータと誤差の関係について、2章において、特に提案分布の形状情報に相当する分散が重要なファクターであることを明らかにした上で、その結果に基づく分散の設計・更新要領を提案し、その有効性についてまとめている。

確率・統計的最適化と決定論的最適化手法双方の性質を持ち合わせた手法については、5章で示すサンプルベース・ニュートン型最適化法がそれに対応する。5章では、勾配やヘシアンを事前に解析的な形で明らかにしておく必要がない“勾配レス”な手法であるという点で確率・統計的最適化手法の性質を有しながら、同時に、決定論的最適化手法であるニュートン法の幾何学的解釈を利用しており、その最適化の精度については決定論的最適化手法と同程度であることが期待できるサンプルベース・ニュートン型最適化法を提案し、その有効性などについてまとめている。

## 1.2.2 研究課題2：適用可能性について

冒頭の1.1節でも述べたとおり、確率・統計的最適化手法は最適化計算に目的関数に関する勾配やヘシアンと呼ばれる事前情報を必要としないという特長がある。確率・統計的最適化手法の大部分を占めるモンテカルロ最適化法についてまとめたRobertら[43]の文献によると、この特長により、決定論的最適化が適用困難／不可とされるような以下のような問題にも容易に適用できることこそが、確率・統計的最適化手法を用いる最大の利点となり得るとされている。

- 非線形性が強く、勾配やヘシアンが数値的に不安定になる。
- 目的関数の一部又は全部がブラックボックス化されている。
- 目的関数などに不連続・微分不可な関数が含まれる。

ここに挙げた以外にも、決定論的最適化に比べ、初期値依存性が低く、多峰性があるような問題においても、局所解からの脱出が容易であることなどを特長に挙げる文献 [27,39] も存在する。しかしながら、以上の性質は、あくまでも理想的な条件下における確率・統計的最適化法の一般的な性質として知られているものである。

サンプル型 MPC のアイデアが誕生して以来、多数の文献が、上述した確率・統計的最適化手法の性質を積極的に利用した制御の実現可能性について言及している [20,27,36,39,51]。一方で、前段落で述べた理想的な条件下とは、最適化に十分な時間を費やすことが可能であるなどであり、“実時間性” が求められる MPC の問題設定に対しては、非現実的な条件である。そのため、実際には、先ほど列挙したような決定論的最適化が適用困難／不可とされる問題において、確率・統計的最適化手法の恩恵をどの程度享受可能であるかなどについては、目的関数の一部または全部がブラックボックス化された問題への適用とその結果についてまとめた文献 [57,58] が存在するほかは、そのほとんどが明らかにされていない。例えば、目的関数に微分不可能となる点が含まれる例題や、強い非線形性などでヘシアンが悪条件となり決定論的なニュートン型最適化法がうまく機能しないといった例題にサンプル型 MPC を適用したという例は、筆者らが文献を調査した限りでは確認できていない。

そこで、本研究では、これらの性質を持つ問題に MCMPC を適用し、その効果を検証することで、これまで従来の勾配型 MPC では適用困難／不可とされていた問題に対してどの程度まで適用可能かについて考察する。本研究課題の意義は下記に示す通りである。例えば、前者の微分不可能な点が存在する目的関数を持つシステムへの適用可能性が明らかになれば、二足歩行ロボットが外界と衝突した場合、ドローンが壁面等に衝突した場合、倒立振り子型移動体が段差に衝突した場合など、不連続現象を含む非線形システムにおいてブレイクスルーとなることが予想される。また、強い非線形性については、不連続性の他に、厳しい制約条件が課されていることに起因するものが想定される。これについても、近年、さまざまな分野において、システムの非線形性や各種の制約を考慮した上で性能限界を追求したいという需要が高まりつつあることから、こういった要求に対するブレイクスルーとなることが予想される。

### 1.3. 本論文の構成

本論文は全7章から構成される。1章では、研究背景、研究目的、研究課題及び論文の構成について述べた。2章では、モデル予測制御全般の説明、MCMPCのアルゴリズム、MCMPCにおける同一制御周期内反復と呼ばれる反復規則とその収束解析の結果などについて記述する。収束解析では、最適解が解析的な形で与えられる離散時間線形システムに対する制約なし最適制御問題を考え、同一制御周期内反復の適用により、MCMPCの推定値の期待値が真の最適解に収束することを明らかにする。さらに、その際の結果から、ユーザーパラメータの1つである提案分布の分散と収束特性の関係を明らかにし、当該パラメータが収束特性を特徴づける重要なパラメータであることを明らかにする。

3章及び4章では、2章で明らした提案分布の分散と収束特性の関係を利用し、同一制御周期内反復の過程で、提案分布の分散共分散行列を更新する手法について検討し、収束特性改善やサンプル生成の高効率化についても評価を行う。3章では、最もサンプル生成が容易な分散共分散行列が対角行列、かつ、すべての要素が同じ値である場合の更新則を考えた。また、その仮定のもとで、双曲線型冷却法と呼ばれる手法が、推定値と真の最適値の誤差、推定値のばらつきの双方を0に収束させることができる手法であることを明らかにする。4章では、効率的にサンプルを生成する方法として、対角項のみでなく非対角項についても考慮できる分散共分散行列適応進化戦略 (Covariance Matrix adaptation-Evolutional strategy; CMA-ES) を組み入れた方法について検討を行っている。

5章では、最適化時に、決定論的最適化手法であるニュートン法の幾何学的解釈に基づき、ランダムサンプリングと2次曲面回帰により、勾配情報、具体的には、勾配ベクトルとヘッセ行列を近似するサンプルベース・ニュートン型最適化法について記述している。ここでは、勾配型MPCでは対処困難な問題におけるMCMPC及びサンプルベース・ニュートン型最適化法を用いたMPCの有効性についても検証している。

以上の2章から5章は主に、性能向上に資する本研究の取り組みとその成果に対応している。

6章は、MCMPCの適用可能性について、勾配型MPCが適用不可な問題の1つと考えられる、不連続現象を考慮した最適制御問題への適用とその結果についてまとめている。

7章は、本研究の成果と今後の課題についてまとめている。

## 第2章 モンテカルロモデル予測制御の収束性解析

### 2.1. はじめに

実時間非線形最適制御手法に基づく MPC において、その最適化の精度は、制御性能そのものに影響を与える重要な要素である。他方、本研究で着目している MCMPC などの確率・統計的最適化手法に基づくサンプル型 MPC の最適化の精度が、従来の勾配型 MPC に比べ劣るという問題があることについては、既に 1.2.1.1 や 1.2.1.2 で述べたとおりである。この問題については、これまでに多くの先行研究で改善策が検討されている [27,36,37,51,52,56]。文献 [27] では、重点サンプリングを用いた改善手法が提案されているが、目的分布と提案分布の形状が乖離している場合には十分な効果が期待できないといった問題がある。文献 [36,56] では、最適化の精度が与える影響のうち、実際にシステムに投入する入力の小刻みに振動するガウス雑音のようになる点に着目し、Savitzky-Golay filter のようなフィルターにより入力を平滑化することで改善を試みる方法が提案されている。ただし、この方法には、平滑化後の入力が、平滑化前の入力より最適化という観点で優れていることを保証できないという問題がある。また、[52] や [51] では、前者であれば直接的に、後者の場合も間接的に、制御評価関数についての微分情報が必要になるという点で、サンプル型 MPC の重要な特長と相反するという問題がある。以上のような特性をふまえ、本章では、[37] で採用されている“同一制御周期内反復”と呼ばれる特殊な反復規則に基づく改善策にフォーカスを当てる。

同一制御周期内反復 [28,37] のアイデアは、MCMPC のアルゴリズムにおいて、サンプル生成に使用する提案分布の平均値や分散などのパラメータを反復の都度更新することで、目的分布と提案分布の形状の乖離により生じる誤差を減少させ、近似精度を向上させようとするものである。しかしながら、これらの研究 [28,37] では、具体的にどのパラメータをどのような要領で設計・更新すれば近似精度が向上するかについてまでは言及されていない。この同一制御周期内反復について、その収束解析を行い、提案分布のパラメータと収束特性の関係を明らかに出来れば、収束特性については MCMPC の制御性能

を向上する最適化アルゴリズムが構築できる可能性がある。

そこで、本章では、目的分布が解析的な形で表現可能な離散時間線形システムに対する最適制御問題において、同一制御周期内反復の反復規則 1 回あたりの推定値と最適解との誤差の関係を解析的な形で明らかにする。さらに、その結果から MCMPC の収束性を決める重要な要素である提案分布のパラメータのうち、どのパラメータが近似精度の向上に最も寄与するかについても考察する。

本章の構成は、以下に示す通りである。MCMPC の収束性解析に先立ち、2.2 節では、MPC の概要及び一般的な MPC の問題設定（最適制御問題）の定式化について述べる。次いで、2.3 節では、2.2 節で定式化した最適制御問題に対する MCMPC の最適化手順、制御量や操作量に係る制約の処理要領などについて記述する。本章の主要な結果である、離散時間線形システムに対する最適制御問題における同一制御周期内反復適用時の収束性解析の結果については、2.4 節に示す。また、収束性解析の結果についての数値例を交えた考察は、2.5 節にまとめる。最後に、2.6 節にて、本章の内容を総括するとともに、収束性解析の結果から得られる収束特性改善手法についても考察する。

## 2.2. モデル予測制御

まず、本節では、MPC の概要及び一般的な MPC の問題設定について述べる。

MPC は、制御対象の数値モデルなどを用いて、予測ホライズンと呼ばれる一定時間未来までの時刻区間における制御量の将来挙動を予測し、制御目的を定量的に表した制御評価関数が最小となるように、(1) 予測ホライズン分の操作量（入力）を、最適化により決定する。(2) 制御時刻が進むたびに、予測ホライズンを後退させて、最適化し直す。という 2 つの考え方に基づく制御手法である [59]。(1) で解く最適化問題は、最適制御問題とよばれ、現時点から一定時間未来までの時間区間（予測ホライズン）を定め、目標状態からの誤差や入力の大きさ、入力や制御量にかかる制約を考慮してユーザが設計する。また、(2) に示した方策は *receding horizon policy* と呼ばれ、制御時刻の進行にあわせ、ホライズンを後退させながら有限時間の最適制御問題を解き直し、得られた入力のうち初期時刻のものを印加し続けることを意味している [9]。

以下では、一般的な MPC の問題設定について記述する。制御対象として、以下に示すようなオイラー近似された離散時間非線形システムを考える。

$$x_{k+1} = x_k + f(x_k, u_k)\Delta t. \quad (2.1)$$



ここで、 $k$  は現在の制御時刻、 $\Delta t$  は制御対象のサンプリング周期、 $f(x_k, u_k)$  は制御対象の状態遷移を表す非線形関数である。また、 $x_k \in \mathbb{R}^n$  及び  $u_k \in \mathbb{R}^m$  は、それぞれ状態及び入力を表すベクトルである。次に最適制御問題として、現在時刻  $\{k|0\}$  から有限時間  $T$  だけ未来までの予測ホライズン  $[\{k|0\}, \{k|T\}]$  における制御評価関数  $J$ （以下、評価関数という。）を最小化する、次式のような非線形制約付きの最適化問題を考える。

$$\begin{aligned}
& \underset{\mathbf{u}_k}{\text{minimize}} && J = \varphi(x_{\{k|T\}}) + \sum_{\tau=0}^{T-1} S(x_{\{k|\tau\}}, u_{\{k|\tau\}}, \tau) \\
& \text{subject to} && x_{\{k|0\}} = x_k \\
& && x_{\{k|\tau+1\}} = x_{\{k|\tau\}} + f(x_{\{k|\tau\}}, u_{\{k|\tau\}})\Delta t \\
& && H(x_{\{k|\tau\}}, u_{\{k|\tau\}}) = 0 \\
& && G(x_{\{k|\tau\}}, u_{\{k|\tau\}}) \geq 0
\end{aligned} \tag{2.2}$$

ここで、終端コスト  $\varphi(x_{\{k|T\}})$  及びステージコスト  $S(x_{\{k|\tau\}}, u_{\{k|\tau\}}, \tau)$  は制御目的に合わせユーザが設計するスカラー値関数である。また、制約条件として、初期値条件、状態方程式、入力や状態などに関する等式制約  $H(x_{\{k|\tau\}}, u_{\{k|\tau\}})$  および不等式制約  $G(x_{\{k|\tau\}}, u_{\{k|\tau\}})$  が課されている。上記の最適制御問題を何らかの方法で解いて最適制御入力

$$\mathbf{u}_k^* = \underset{\mathbf{u}_k}{\operatorname{argmin}} J, \tag{2.3}$$

を求め、そのうちの最初の時刻のものである  $u_{\{k|0\}}^*$  を実際の制御入力として制御対象に投入する。以上の手順を制御時刻が進むごとに予測ホライズンをずらしながら繰り返していくのが MPC の一般的な問題設定である。

## 2.3. モンテカルロモデル予測制御

MCMPC は、以下に示す 3 つの主要なステップで構成されている。

手順 1 入力列の生成

手順 2 並列シミュレーション

手順 3 推定値の計算

手順 1 ～ 3 は、サンプル型 MPC の標準的な構成であり、多くのサンプル型 MPC（例えば、[21,22,27,36,60].）で採用されている。Fig. 2.2 は、手順 1 ～ 3 を簡略化し図示したものである。Fig. 2.2 に示す通り、手順 1 ～ 3 のほとんどの部分は、並列演算可能であり、

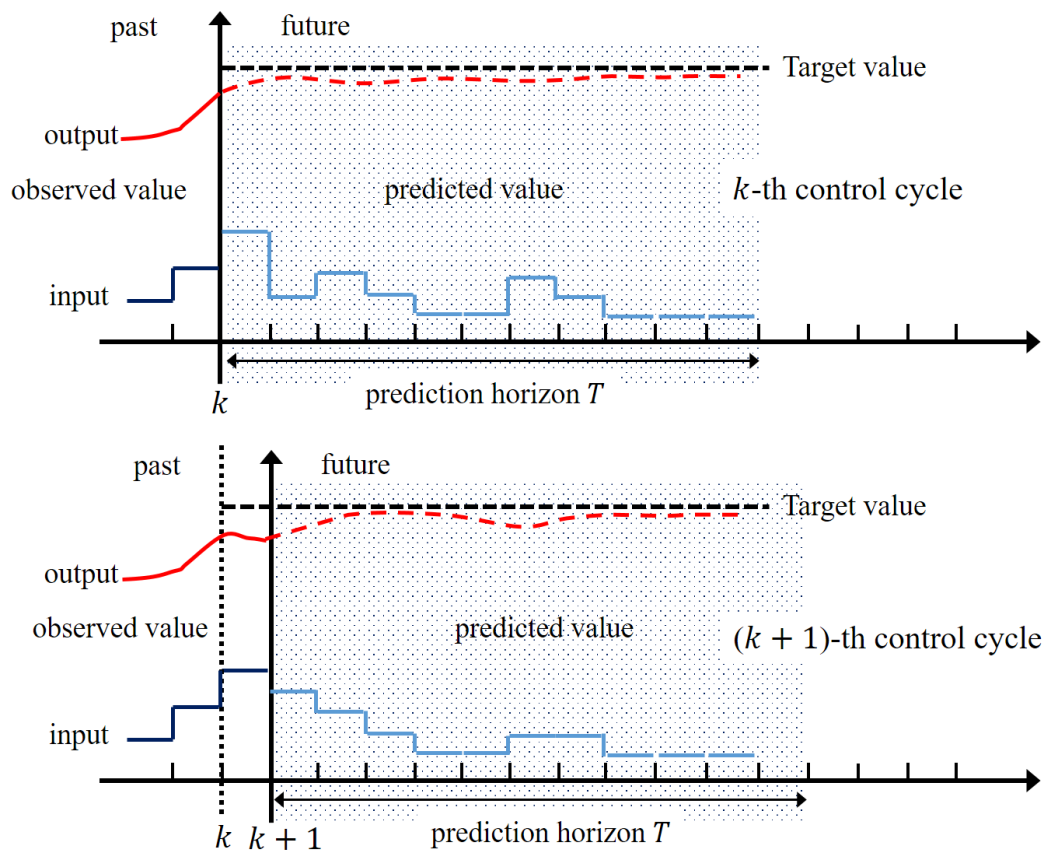


Fig. 2.1: Concept of model predictive control

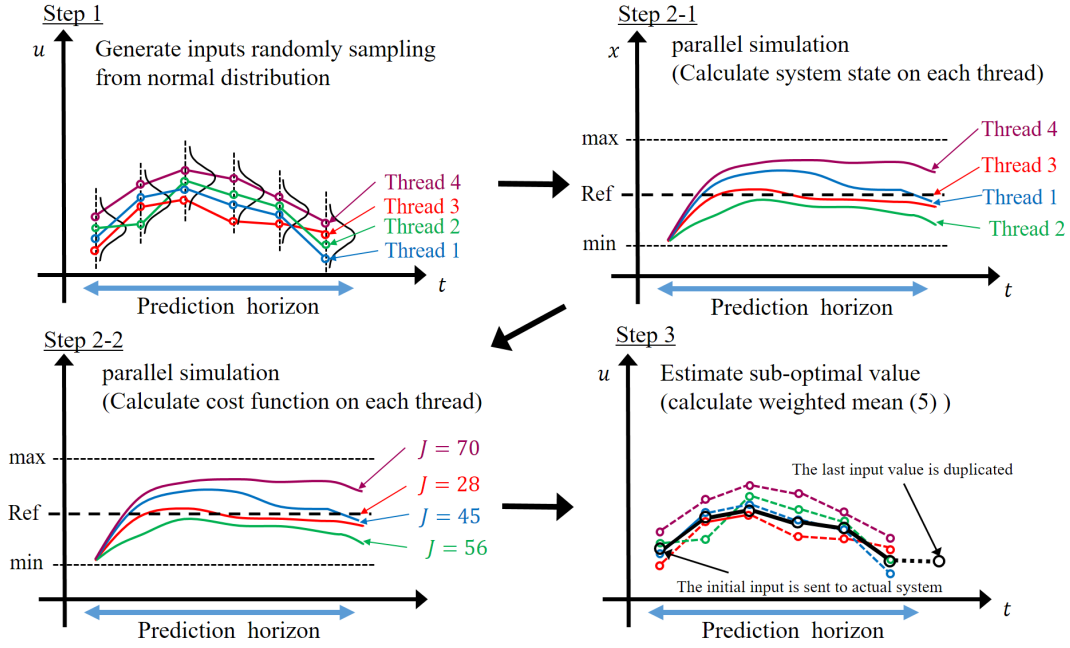


Fig. 2.2: Flowchart of the algorithm for Monte Carlo model predictive control.

GPU などの並列演算デバイスを用いることで計算時間を大幅に短縮することができる。これにより、例えば、ラリーカー（サンプリング周期：60 Hz）[21,27,36] やクアッドコプター [22,23]（サンプリング周期：20 Hz）のようなサンプリング周期が短い制御対象にも実時間で適用できるとされている。以降では、文献 [28,37,46,61] に沿って、各ステップの概要とアルゴリズムについて説明する。

### 2.3.1 入力列の生成

MCMPC における入力列の生成には、次のような平均ベクトル  $\bar{\mathbf{u}}_k$  と分散共分散行列  $\Sigma_k$  をもつ多変量正規分布からランダムサンプリングを用いるのが一般的である [23,28,37,46, 61,62].

$$\hat{\mathbf{u}}_k^l \sim \mathcal{N}(\bar{\mathbf{u}}_k, \Sigma_k) \quad (2.4)$$

ここで、 $l$  は、全  $L$  個のサンプル中、 $l$  番目のサンプルであることを示す添字である。また、平均ベクトル  $\bar{\mathbf{u}}_k$  には、直前の制御時刻  $k-1$  における推定値ベクトル  $\hat{\mathbf{u}}_k$  を、予測ホライズンを 1 ステップ分だけ後退させた

$$\bar{\mathbf{u}}_k \leftarrow [\tilde{u}_{\{k-1|1\}}, \dots, \tilde{u}_{\{k-1|T-1\}}, \tilde{u}_{\{k-1|T-1\}}] \quad (2.5)$$

を用いる。このとき、 $T$  番目の予測ステップにおける推定値が空になるので、 $\tilde{\mathbf{u}}_{\{k-1|T-1\}}$  をコピーし、使用している。制御開始直後など、推定値  $\tilde{\mathbf{u}}_k$  についての情報がない時は、 $\tilde{\mathbf{u}}_k = [0, \dots, 0]$  とするのが一般的である。分散共分散行列  $\Sigma_k$  については、 $\sigma_k^2$  を対角成分にもつ対角行列を用いる方法 [23,28,37,46,61,62] の他に、分散共分散行列適応進化戦略 (Covariance matrix adaptation evolution strategy; CMA-ES) [63,64] を用いて制御時刻ごとに更新していく方法 [65] などが提案されている。

### 2.3.2 並列シミュレーション

並列シミュレーションでは、(2.4) に従いサンプリングした入力列を用いて、システムモデルである状態方程式 (2.1) などを用いて、システムの将来挙動を予測する。予測シミュレーションは、(2.4) で生成した  $L$  個の入力列ごとに独立して実行することができる。そのため、GPU などの並列演算デバイスを用いることで、入力列ごとに少しずつ異なる多数のシナリオを並行して予測することができ、計算時間を大幅に短縮することができる。

次に、予測ステップ  $0 \sim T$  までの予測シミュレーションの結果を用いて、各入力列に対応する評価値をユーザが設計した評価関数  $J$  を用いて計算し、入力列ごとの重み（経路コストなどとも呼ばれる [27].）を計算する。

### 2.3.3 推定値の計算

MCMPC では、次に示すような重み付き期待値を計算し、推定値を導出する。

$$\tilde{\mathbf{u}}_{\{k|\tau\}} = \frac{\sum_{l \in \hat{L}} \hat{\mathbf{u}}_{\{k|\tau\}}^l \exp\left(-\frac{J^l}{\eta}\right)}{\sum_{l \in \hat{L}} \exp\left(-\frac{J^l}{\eta}\right)}. \quad (2.6)$$

ここで、 $\eta > 0$  であり、 $\hat{L}$  は、(1.2) に対応する (2.6) 式に示す重み付き平均の計算に使用するサンプルのサイズを表している。MCMPC は、(2.6) の計算を行う際に用いるサンプルのサイズの違いによって 3 つのパターンに分類 [46] できる。以下はその 3 つのパターンの分類と特徴を示している。

#### Normal type MCMPC

##### 定義

(2.6) の計算に使用するサンプルのサイズを  $\hat{L} = L$  とする方法

## 特徴

評価関数が多峰性である場合などに最適化の性能が低下する。

## Elite sample MCMPC

### 定義

全  $L$  個のサンプルを評価値  $J$  の昇順にソートし、そのうち上位  $\hat{L}(\leq L)$  個までを使用して (2.6) の演算を行う方法

### 特徴

**Normal type MCMPC** より最適化性能（とくに、収束速度）が良い。**TOP 1 sample MCMPC** よりスムーズな入力をシステムに印加できる。もっともよく用いられている方法である [23,27,28,46,62]。

## TOP 1 sample

### 定義

全  $L$  サンプル中、もっとも良い評価値を持つサンプルを推定値とする方法

### 特徴

評価関数に複数の峰がある場合、評価関数に微分不可又は不連続となる点が存在する場合などに有効な方法である。システムに印加する入力がノンスムーズな入力になる。

また、式 (2.6) で使用する、評価値  $J$  を用いた  $\exp(\cdot)$  型の重みを用いた最適化については、理想的なサンプル生成が出来るという仮定のもとで、 $\eta \rightarrow 0$  としたときの収束性が示されている [45]。

推定値 (2.6) の計算まで終わると、推定最適制御入力列

$$\tilde{\mathbf{u}}_k = [\tilde{u}_{\{k-1|0\}}, \dots, \tilde{u}_{\{k-1|T-1\}}, \tilde{u}_{\{k-1|T\}}] \quad (2.7)$$

のうち、 $\tilde{u}_{\{k-1|0\}}$  を実際にシステムに加え、残りの入力列は、(2.5) に示すように、次制御時刻の入力列生成に使用する正規分布の平均値としてコピー・保存する。

以上が、MCMPC のある制御時刻  $k$  における同一制御周期内反復を行わない場合の主要な 3 つの手順となる。同一制御周期内反復では、手順 1 における平均値として (2.7) をそのまま利用し、手順 1 ～手順 3 を所定の回数反復する。

手順1～手順3において、制御対象 (2.1) や評価関数 (2.2) に対する微分や偏微分といった操作は不要である。そのため、状態に依存して生起するタイミングが異なる衝突などの自然発生型のジャンプ現象 [66] を含むような最適制御問題にも容易に適用することができる。

### 2.3.4 各種制約の処理要領

ここでは、MCMPC における一般的な各種の制約の処理要領について記述する。入力  
の上下限に関する制約については、(2.4) でランダムサンプリングした入力列に対して、  
次に示すような飽和処理を施す。

$$\hat{u}_{\{k|\tau\}} = \begin{cases} u_{\max} & (\hat{u}_{\{k|\tau\}} \geq u_{\max}) \\ \hat{u}_{\{k|\tau\}} & (u_{\min} \leq \hat{u}_{\{k|\tau\}} \leq u_{\max}) \\ u_{\min} & (\hat{u}_{\{k|\tau\}} < u_{\min}) \end{cases} \quad (2.8)$$

上記のような簡単な処理で、その後の並列シミュレーション、推定値の計算の際に、少なくとも箱型制約で与えられる入力制約を満足する入力列のみを利用することができる。また、簡単な類推から、上記の処理によって (2.6) に示す推定値も、箱型制約で与えられる入力制約を違反しないことが確認できる。

その他の状態制約など、不等式制約として与えられる制約については、評価関数を計算する際に、次に示すような許容領域  $\mathcal{S}$  の境界付近で無限大に発散する非負のバリア関数

$$B(x_{\{k|\tau\}}) = \begin{cases} \geq 0 & (x_{\{k|\tau\}} \in \mathcal{S}) \\ \rightarrow \infty & (x_{\{k|\tau\}} \notin \mathcal{S}) \end{cases} \quad (2.9)$$

をステージコスト  $S(x_{\{k|\tau\}}, u_{\{k|\tau\}}, \tau)$  に加えた

$$\hat{S}_C(x_{\{k|\tau\}}, u_{\{k|\tau\}}, \tau) = S(x_{\{k|\tau\}}, u_{\{k|\tau\}}, \tau) + \frac{1}{r} B(x_{\{k|\tau\}}) \quad (2.10)$$

を新たなステージコストとして使用する。これにより、全  $L$  個のサンプル中から  $\hat{L}$  個のサンプルを選び抜きする過程でこれらの制約に違反するサンプルが淘汰されやすくなる。

以上のような単純な実装ではあるものの、耐制約性能の比較においては、サンプル型 MPC が勾配型 MPC より優れたパフォーマンスを発揮したという報告もある [27,29]。

## 2.4. MCMPC の線形系における収束性解析

本節では、MCMPC を一般的な離散時間線形システムに適用した際の収束性に関する解析結果を示す。次いで、その内容を応用して、収束性と MCMPC 固有の各種のユーザーパラメータとの関係について考察する。

### 2.4.1 線形システムに対する有限時間最適制御問題

ここでは、最適解が解析的に与えられる最適制御問題である、離散時間線形システムに対する2次形式の評価関数を最小化するLQ制御問題 (Linear quadratic control problem) について考える。制御対象は、 $n$ 次元 $m$ 入力離散時間線形システム

$$x_{\{k|i+1\}} = Ax_{\{k|i\}} + Bu_{\{k|i\}} \quad (2.11)$$

とし、次の状態・入力について2次形式となる評価関数

$$J(x_{\{k|i\}}, u_{\{k|i\}}, k) = \frac{1}{2} x_{\{k|T_d\}}^T Q_f x_{\{k|T_d\}} + \frac{1}{2} \sum_{\tau=0}^{T_d-1} \left( x_{\{k|\tau\}}^T Q x_{\{k|\tau\}} + u_{\{k|\tau\}}^T R u_{\{k|\tau\}} \right) \quad (2.12)$$

を最小化する最適制御入力  $\mathbf{u}_k^*$  を決定する問題を考える。ここで、 $Q_f$  及び  $Q$  はともに  $n \times n$  準正定対称行列、 $R$  は  $m \times m$  正定行列とする。予測ホライズン分の状態・入力ベクトルをまとめて

$$X = \begin{bmatrix} x_{\{k|0\}}^T, \dots, x_{\{k|T_d\}}^T \end{bmatrix}^T \quad (2.13)$$

$$U = \begin{bmatrix} u_{\{k|0\}}^T, \dots, u_{\{k|T_d-1\}}^T \end{bmatrix}^T \quad (2.14)$$

というベクトルを定義すると、評価関数 (2.12) は

$$J(X, U, k) = \frac{1}{2} X^T \bar{Q} X + \frac{1}{2} U^T \bar{R} U \quad (2.15)$$

$$\bar{Q} := \text{block-diag}[Q, \dots, Q, Q_f] \in \mathbb{R}^{n(T_d+1) \times n(T_d+1)} \quad (2.16)$$

$$\bar{R} := \text{block-diag}[R, \dots, R] \in \mathbb{R}^{mT_d \times mT_d} \quad (2.17)$$

と表すことができる。また、状態方程式 (2.11) から

$$X = \bar{A}x_{\{k|0\}} + \bar{B}U \quad (2.18)$$

$$\bar{A} := \begin{bmatrix} E \\ A \\ \vdots \\ A^{T_d} \end{bmatrix} \in \mathbb{R}^{n(T_d+1) \times n} \quad (2.19)$$

$$\bar{B} := \begin{bmatrix} 0 & & & \dots & 0 \\ B & 0 & & \dots & \vdots \\ AB & B & \ddots & & \\ \vdots & & \ddots & & \vdots \\ A^{T_d-2}B & A^{T_d-3}B & \dots & B & 0 \\ A^{T_d-1}B & A^{T_d-2}B & \dots & AB & B \end{bmatrix} \in \mathbb{R}^{n(T_d+1) \times mT_d} \quad (2.20)$$

と表すことができる．ここで， $E$  は  $n \times n$  単位行列である．さらに，(2.18) の  $X$  は  $U$  によって陽に表されているので消去することができ

$$J(U) = \frac{1}{2} x_{\{k|0\}}^T \hat{A} x_{\{k|0\}} + x_{\{k|0\}}^T \hat{B} U + \frac{1}{2} U^T \hat{Q} U \quad (2.21)$$

$$\hat{A} := \bar{A}^T \bar{Q} \bar{A} \quad (2.22)$$

$$\hat{B} := \bar{A}^T \bar{Q} \bar{B} \quad (2.23)$$

$$\hat{Q} := \bar{B}^T \bar{Q} \bar{B} + \bar{R} \quad (2.24)$$

という  $U$  に関する 2 次の評価関数を最小化する問題に帰着することができる．ここで，仮定より  $Q$  と  $Q_f$  がともに準正定であるので  $\bar{Q}$  も準正定となること， $R$  が正定であるので  $\bar{R}$  も正定となることより，(2.24) 式の  $\hat{Q}$  は正定となる．このとき，(2.21) は凸関数であり，(2.3) を満足する  $\mathbf{u}^*$  を次式のように一意に決定することができる．

$$\mathbf{u}^* = -\hat{Q}^{-1} \hat{B}^T x_{\{k|0\}} \quad (2.25)$$

以上の結果は，離散時間線形システムに対する LQ 制御問題 (2.12) における一般論である [9,46]．

#### 2.4.2 主結果:同一制御周期内反復による制御則の収束

本項では，前項で述べた LQ 制御問題 (2.12) に MCMPC(Normal type) を適用した際の収束性解析の結果を示す．

はじめに，以下の仮定をおく．

**仮定 1.** 多変量正規分布 (2.4) の分散共分散行列を

$$\Sigma = \text{block-diag} [\varsigma_{\{ \cdot | 0 \}}, \dots, \varsigma_{\{ \cdot | T_d - 1 \}}] \in \mathbb{R}^{mT_d \times mT_d} \quad (2.26)$$

のように定める．ここで，

$$\varsigma_{\{ \cdot | \cdot \}} := \text{diag} [\sigma^2, \dots, \sigma^2] \in \mathbb{R}^{m \times m} \quad (2.27)$$

である．

MCMPC の手順 3（推定値の計算）で使用する，(2.4) に従いサンプリングされた  $\hat{\mathbf{u}}^\beta$  ごとの重みを表す関数  $w(\hat{\mathbf{u}}^\beta)$  は，前項で示した結果 (2.21) 及び (2.25) を用いて，

$$w(\hat{\mathbf{u}}^\beta) = \exp \left( -\frac{J(\hat{\mathbf{u}}^\beta)}{\eta} \right) \quad (2.28)$$



$$= \exp \left( -\frac{1}{2\eta} (\hat{\mathbf{u}})^T \hat{\mathbf{Q}} \hat{\mathbf{u}} - \frac{1}{\eta} x_{\{k|0\}}^T \hat{\mathbf{B}} \hat{\mathbf{u}} - \frac{1}{2\eta} x_{\{k|0\}}^T \hat{\mathbf{A}} x_{\{k|0\}} \right) \quad (2.29)$$

$$= \exp \left( -\frac{1}{2\eta} (\hat{\mathbf{u}} - \mathbf{u}^*)^T \hat{\mathbf{Q}} (\hat{\mathbf{u}} - \mathbf{u}^*) + \text{const.} \right) \quad (2.30)$$

とすることができる。ここで、 $\hat{\mathbf{u}}$  は、(2.14) の定義に従って、(2.4) でランダムサンプリングした入力列を整列したベクトルである。また、(2.29) および (2.30) では、煩雑化を避けるために、サンプルの番号を示す上添字  $\beta$  を省略していることに注意されたい。

次に、 $\hat{\mathbf{u}}^\beta$  の密度関数について考える。(2.4) に従い多変量正規分布からランダムサンプリングされた  $\hat{\mathbf{u}}^\beta$  は、連続型の確率変数とみなすことができ、その密度関数は、

$$p(\hat{\mathbf{u}}^\beta) = \frac{1}{\sqrt{(2\pi)^{mT_d} |\Sigma|}} \exp \left( -\frac{1}{2} (\hat{\mathbf{u}}^\beta - \bar{\mathbf{u}})^T \Sigma^{-1} (\hat{\mathbf{u}}^\beta - \bar{\mathbf{u}}) \right) \quad (2.31)$$

である。このとき、(2.6) 左辺の  $\bar{\mathbf{u}}$  も確率変数となるから、その期待値は (2.28) と (2.31) を用いて

$$\mathbb{E}(\bar{\mathbf{u}}) := \int w(\hat{\mathbf{u}}) p(\hat{\mathbf{u}}) d\hat{\mathbf{u}} \quad (2.32)$$

$$= (\sigma^2 \hat{\mathbf{Q}} + \eta \mathbf{E}_{\mathbf{u}})^{-1} (\sigma^2 \hat{\mathbf{Q}} \mathbf{u}^* + \eta \bar{\mathbf{u}}) \quad (2.33)$$

とできる（詳細な導出仮定は付録 A に示す。）。ここで、 $\mathbb{E}(\cdot)$  は期待値を意味しており、 $\mathbf{E}_{\mathbf{u}}$  は、 $mT_d \times mT_d$  単位行列とする。(2.32) から (2.33) の導出には、確率変数の関数の期待値の定義 [67] を用いた。さらに、(2.32) から (2.33) を導出する過程において、MCMPC の推定入力列  $\bar{\mathbf{u}}$  の標本分散を

$$\Sigma_S = \frac{\eta \sigma^2}{L} (\sigma^2 \hat{\mathbf{Q}} + \eta \mathbf{E}_{\mathbf{u}})^{-1} \quad (2.34)$$

のように解析的に導出することができる（詳細な計算は、付録 A に示す通りである。）。

次に同一制御時刻内に、 $\bar{\mathbf{u}}_{(d+1)} \leftarrow \bar{\mathbf{u}}_{(d)}$  の要領で (2.4) の多変量正規分布の平均値を更新しながら、手順 1 から手順 3 を反復実行する処理（同一制御周期内反復）について考える。ここで  $(d)$  は反復処理の回数を表している。このとき、式 (2.33) から

$$\bar{\mathbf{u}}_{(d+1)} = (\sigma^2 \hat{\mathbf{Q}} + \eta \mathbf{E}_{\mathbf{u}})^{-1} (\sigma^2 \hat{\mathbf{Q}} \mathbf{u}^* + \eta \bar{\mathbf{u}}_{(d)}) \quad (2.35)$$

となり、最適制御入力列  $\mathbf{u}^*$  と  $d$  回目の推定結果  $\bar{\mathbf{u}}_{(d)}$  の間の誤差ベクトルを

$$\mathbf{e}_{(d)} := \bar{\mathbf{u}}_{(d)} - \mathbf{u}^* \quad (2.36)$$

と定義すると、誤差ベクトルについて次の漸化式 (2.37) で示す関係が成立する。

$$\mathbf{e}_{(d+1)} = \left( \frac{\sigma^2}{\eta} + \mathbf{E}_{\mathbf{u}} \right)^{-1} \mathbf{e}_{(d)} \quad (2.37)$$

以上の考察から、以下に示す重要な結果を得ることができる。

**定理 2.** LQ 制御問題 (2.12) に MCMPC を適用した際の  $d$  回目の同一制御周期内反復における推定結果  $\bar{\mathbf{u}}_{(d)}$  の“期待値”は、 $d \rightarrow \infty$  としたとき、LQ 制御問題 (2.12) の唯一の最適解  $\mathbf{u}^*$  に収束する。

*Proof.* 誤差ベクトル  $\mathbf{e}_d$  が  $d \rightarrow \infty$  で 0 に漸近収束するための必要十分条件は、行列

$$\Omega := \left( \frac{\sigma^2}{\eta} + \mathbf{E}_{\mathbf{u}} \right)^{-1} \quad (2.38)$$

の全ての固有値の絶対値が 1 未満となることである。

任意の実正定値行列  $M_A, M_B$  について、ある行列  $Z$  の  $i$  番目の固有値を  $\lambda_i(Z)$  と記述することにとすると、

$$\forall i \in \mathbb{R}: \quad \lambda_i(M_A + M_B) > \lambda_i(M_A) \quad (2.39)$$

となる (証明は省略) ことを利用すると、仮定より  $\hat{Q}$  は実正定対称行列であるから

$$\lambda_i \left( \frac{\sigma^2}{\eta} + \mathbf{E}_{\mathbf{u}} \right) > \lambda_i(\mathbf{E}_{\mathbf{u}}) = 1 \quad (2.40)$$

となる。ここで、正則行列の逆行列の固有値がもとの行列の固有値の逆数となる定理から

$$\lambda_i(\Omega) < \lambda_i(\mathbf{E}_{\mathbf{u}}^{-1}) = 1 \quad (2.41)$$

となる。以上の結果から行列  $\Omega$  の全ての固有値の絶対値が 1 未満となる。このとき、

$$\lim_{d \rightarrow \infty} \mathbf{e}_{(d)} = 0 \quad (2.42)$$

となり、誤差ベクトル  $\mathbf{e}_{(d)}$  の定義から

$$\lim_{d \rightarrow \infty} (\bar{\mathbf{u}}_{(d)} - \mathbf{u}^*) = 0 \quad (2.43)$$

となる。(2.43) は、 $\bar{\mathbf{u}}_{(d)}$  が  $d \rightarrow \infty$  としたとき、 $\mathbf{u}^*$  に収束することを示している。  $\square$

さらに、ここまでの考察から以下に示す MCMPC の収束性に関する 2 つの重要な系が得られる。

**系 1.** 重み付き確率の計算に使用するサンプル数  $\mathcal{B}$  と (2.6) の推定値  $\bar{\mathbf{u}}$  の標本分散 (2.34) との間に

$$\lim_{L \rightarrow \infty} \Sigma_S = 0 \quad (2.44)$$

の関係が成立する。また、簡単な類推からサンプル数  $L$  を大きくすれば、 $O(\frac{1}{\sqrt{L}})$  で標本分散を小さくできることが分かる。

系 2. 推定値  $\tilde{\mathbf{u}}$  の標本分散 (2.34) と誤差ベクトル  $\mathbf{e}_{(0)}$  に対する漸化式 (2.37) における係数行列  $\Omega$  の固有値について，提案分布の分散  $\sigma$  の大小関係から以下の関係式が導かれる．

$\sigma$  が大きいとき：

$$\forall i: \lim_{\sigma \rightarrow \infty} \lambda_i(\Omega) = 0 \iff \tilde{\mathbf{u}}_{(0)} = \mathbf{u}^* \quad (2.45)$$

$$\lim_{\sigma \rightarrow \infty} \Sigma_S = \frac{\lambda \hat{Q}^{-1}}{L} \quad (2.46)$$

$\sigma$  が小さいとき：

$$\forall i: \lim_{\sigma \rightarrow 0} \lambda_i(\Omega) = 1 \iff \lim_{d \rightarrow \infty} \tilde{\mathbf{u}}_{(d)} = \bar{\mathbf{u}}_{(0)} \quad (2.47)$$

$$\lim_{\sigma \rightarrow 0} \Sigma_S = 0 \quad (2.48)$$

系 2 の (2.45)～(2.48) は，収束速度と推定精度の間のトレードオフ関係を示している．つまり，収束速度を早くするには  $\sigma$  を大きく設定する必要があるが，大きく設定し過ぎると推定値  $\tilde{\mathbf{u}}$  の標本分散は一定値となり推定精度はあまり良くならない．反対に  $\sigma$  を小さく設定すれば，収束速度は遅くなるが，推定精度は高くなる，ということを意味している．次の節では，この内容について簡単な数値例を示す．

## 2.5. 数値例

ここでは，2.4.2 項系 2 で示した内容を確認する数値例を示し，収束速度と推定精度のトレードオフ関係について考察する．検証については，**Problem 1** に示す離散時間線形システムのレギュレート問題で行った．

### Problem 1. 離散時間線形システム

本問題における離散時間線形システムの状態方程式は，

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k \quad (2.49)$$

$$\mathbf{A} = \begin{bmatrix} 1.0 & 0.0 & 0.01 & 0.0 \\ 0.0 & 1.0 & 0.0 & 0.01 \\ 0.0 & -0.017 & 0.881 & 0.0 \\ 0.0 & 0.430 & 0.448 & 1.0 \end{bmatrix} \quad (2.50)$$

$$\mathbf{B} = \begin{bmatrix} 0.0 \\ 0.0 \\ 0.094 \\ -0.354 \end{bmatrix} \quad (2.51)$$

である．ここで  $x_k \in \mathbb{R}^2$ ,  $u_k \in \mathbb{R}^1$  である．評価関数は，(2.12) と同じ状態・入力について 2 次形式の形で記述されるものを採用する．また，予測ステップ数  $T_d = 30$ ，状態・入力に関する重み行列を

$$Q = \begin{bmatrix} 1.75 & 0.0 & 0.0 & 0.0 \\ 0.0 & 1.75 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.04 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.05 \end{bmatrix} \quad (2.52)$$

$$R = 1, \quad (2.53)$$

システムの初期状態を  $x_0 = [2.5, 0.25, 0.5, 3.0]^T$  とした．

Fig. 2.3 は，**Problem 1** に示した最適制御問題に MCMPC を適用した際の結果である．横軸は，同一制御周期内反復の反復回数，縦軸は，各反復における推定誤差  $e_{\{0|0\}}^{(d)} = \tilde{u}_{\{0|0\}}^{(d)} - u_{\{0|0\}}^*$  である．それぞれのグラフにおける赤色の実線は，全 50 回の試行における予測区間上の初期時刻における予測入力と最適値との誤差  $e_{\{0|0\}}^{(d)} = \tilde{u}_{\{0|0\}}^{(d)} - u_{\{0|0\}}^*$  の平均値，オレンジの領域は，大きさ  $3\sigma$  に相当する標準偏差，黒色の破線は全 50 回の試行中のある試行の結果をプロットしたものである．**Problem 1** に対する最適制御入力  $u^*$  は，(2.25) を用いて解析的に導出することができ， $u_{\{0|0\}}^* = 2.196$  となる．

Fig. 2.3 を見て明らかなように，対角行列である提案分布の分散共分散行列  $\Sigma$  の各要素の値（実質的には，(2.26) 及び (2.27) の仮定より  $\sigma$  の値に相当する．）が大きくなるほど，誤差  $e_0$  の平均値が 0 近傍に収束するまでに要する同一制御周期内反復の回数が少なくなっていることが分かる．一方で  $\sigma$  の値が大きいほど，推定結果のばらつきを示すオレンジの領域（標準偏差）が大きくなり，黒色破線で確認できるように収束後に 0 近傍で大きく振動している様子が確認できる．逆に， $\sigma$  を小さく設定した場合 Fig. 2.3(b) の結果では，Fig. 2.3(d) の結果と比べ，誤差が 0 となるまでの反復回数は 2 倍近くになるが，推定値のばらつきは非常に小さく抑えることができている．

以上の結果より，2.4.2 項系 2 で示したとおり，MCMPC 固有のユーザパラメータである  $\sigma$  の与え方により，推定値の精度と収束までに要する反復回数の間トレードオフの関係があることが明らかになった．

## 2.6. 2 章のまとめ

2 章では，準備として，冒頭 2.2～2.3 節にかけて，一般的な MPC の問題設定である最適制御問題の定式化を行い，MCMPC のアルゴリズムにより最適制御問題を解く実際の

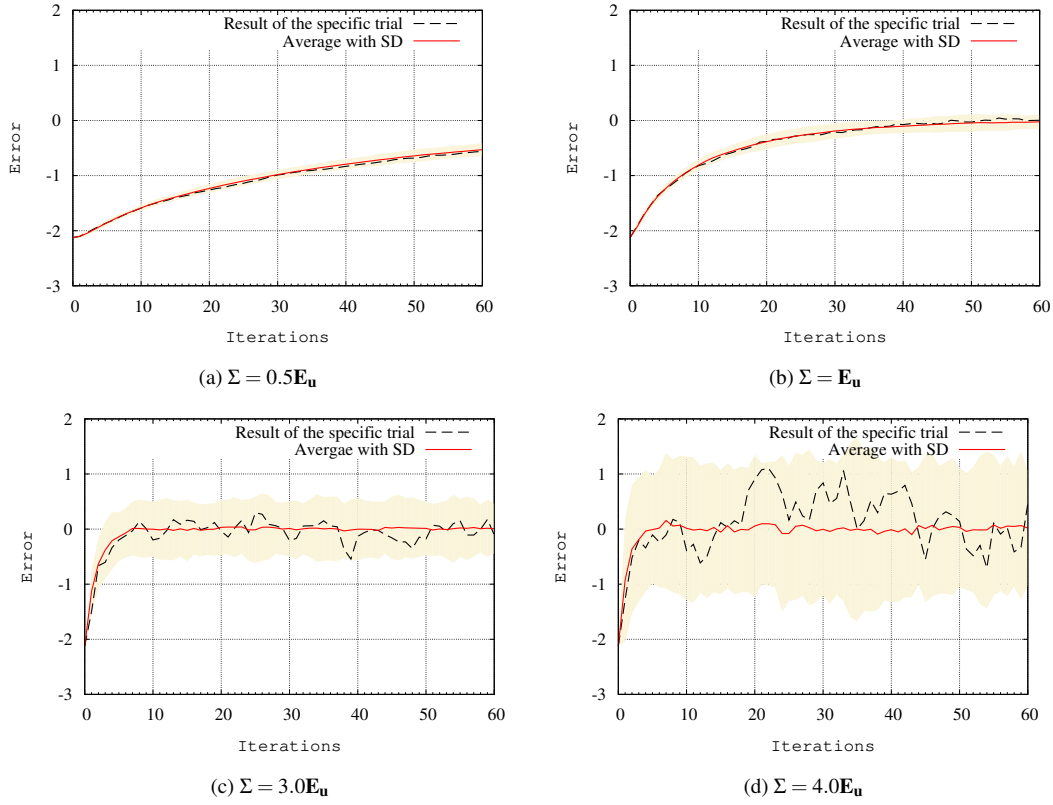


Fig. 2.3: Effect of  $\sigma$  on estimation error  $\mathbf{e}_0 = \tilde{u}_0 - u_0^*$  in the numerical simulation on **Problem 1**.

手順について簡潔に記述した。

その上で、2.4 節では、本章の主要な結果である、離散時間線形システムに対する有限時間最適制御問題を対象とした際の、同一制御周期内反復の反復規則 1 回あたりの推定値と最適解との誤差の関係を解析的な形で明らかにした収束性解析の結果を示した。以下では、収束性解析の結果に基づく MCMPC の収束特性とその改善手法を検討する上で留意すべき内容について示す。2.4 節の収束性解析のうち、MCMPC の収束特性について理解する上で、とくに重要な要素となる内容は、系 1 及び系 2 にまとめた内容である。系 1 は、(2.6) で示した  $\exp(\cdot)$  で重み付けされた分布の標本平均の形で与えられる MCMPC の推定値が確率変数となることに起因する誤差（推定値のばらつき；標本分散）を、(2.6) の計算に使用するサンプルのサイズで抑える方法を採用した際の効果を示したものである。許容できる誤差の大きさ、(2.34) 右辺に示す標本分散が予め与えられれば、サンプル数が見積もれる反面、その効果はサンプル数  $L$  の増加に対して  $O(1/\sqrt{L})$  と非常に緩慢であることが分かる。さらに、このとき考慮している誤差は、推定値のばらつきに起因するものだけであり、サンプル数を増やしたところで、(2.37) の漸化式に示す推定値と最適値の誤差の収束速度には全く影響がないことに注意が必要である。

一方、系 2 は、推定値のばらつきに起因する誤差、(2.37) の漸化式で示す誤差の双方についての関係を述べたものになっている。系 2 の主張は、対角行列で与えられる分散共分散行列をもつ提案分布を考えた時に、対角要素に相当する  $\sigma$  を大きくすれば、(2.37) の漸化式で示す誤差の収束速度が速くなる一方、推定精度に相当する推定値のばらつきが大きくなることを意味している。これを数値例として示したのが、2.5 節に示す結果である。

最後に、本章の結果を踏まえた収束特性改善手法について考察する。系 1 及び系 2 の結果を援用すると、サンプルサイズをいたずらに増やすのは、収束速度にほとんど影響しないのみでなく、1 反復あたりの計算時間の増加にも繋がるので、反復により推定精度を向上させるというアイデアには適さないと考えられる。他方、提案分布の分散共分散行列は、収束速度と推定精度の双方に影響のあるパラメータであり、対角行列を用いる限り、その大小により計算時間が増加することはないので、反復に基づく戦略をとるのであれば、この分散共分散行列を適切に設計・更新するのが最も効果的な改善手法になり得ると考えられる。そこで、以降の章では、系 2 で示したトレードオフ関係を活用した MCMPC の収束特性改善手法を示し、その有効性を確認する。

## 第3章 各種冷却法を用いたモンテカルロモデル予測制御の収束特性改善

### 3.1. はじめに

第2章では、最適解が解析的な形で与えられる離散時間線形システムに対する有限時間最適制御問題において、特殊な反復規則を用いることで、MCMPCの推定値の期待値が真の最適解に一致することを示した。さらに、MCMPCのパラメータのうち、提案分布(2.4)の分散共分散行列が収束速度や推定精度を決める重要なパラメータであることを明らかにした(系2)。系2は、 $\infty$ 回目の反復において、推定値  $\hat{\mathbf{u}}_{(d)}$  を  $\mathbf{u}^*$  に完全に一致させるためには、分散  $\sigma$  を適切にコントロールする必要があることを意味している。

上述した内容は、1.2.1.1 で示した Pincus の定理と深い関わりがある。提案分布の分散を  $\sigma \rightarrow 0$  とすると、提案分布は、その平均値におけるデルタ関数  $\delta(\hat{\mathbf{u}} - \mathbf{u})$  となる。つまり、間接的にはあるものの、Pincus の定理において、 $\eta \rightarrow 0$  とするのと同じ効果が期待できるのである。一方、収束速度との両立を考えると、初期の  $\sigma$  を少し大きめに設定し、徐々に絞り込んで行くというのが現実的かつ効果的な手法になると予想できる。SA は、ここで述べた方法と同様のアイデアで、解の更新、すなわち反復の度に温度と呼ばれるパラメータを更新しながら最適化を行う“探索”ベースの確率・統計的最適化手法である。SA における温度と呼ばれるパラメータは、 $\exp(\cdot)$  で与えられる重みに含まれる  $\eta$  に対応しているが、 $\eta$  を“冷却”して  $\eta \rightarrow 0$  とするのと、提案分布の分散  $\sigma$  を絞り込む操作に同じ効果が期待できるのは既に述べた通りである。そこで、SA における温度冷却法に着想を得た方法で、分散  $\sigma$  を更新する手法を提案し、その有効性を検証する。

本章の構成は次に示す通りである。3.2 節及び 3.3 節にて、MCMPC の収束特性を改善するために新たに導入する冷却法の概要について説明する。特に、3.3 節では、双曲線型冷却法が、2.4 節で考えた問題設定においては、 $\infty$  回反復時の推定値のばらつきと推定値と最適解との誤差の双方を 0 にすることができる手法であることを解析的な形で示す。ついで、3.5 節では、3.2 節及び 3.3 節で提案した各冷却法による収束特性改善手法の有効性について、離散時間線形システム、MPC の性能評価によく用いられる非線形システムであ

る台車型倒立振子の振り上げ安定化制御の2つの例題への適用結果を示し、検証・考察する。また、ここまでの内容をまとめ、提案分布に分散共分散行列が対角行列となる多変量正規分布を用いる場合のパラメータ設計・更新の指針を3.6節に示す。最後に、3.7節に本章のまとめを記述する。

### 3.2. 幾何冷却法

幾何冷却法を用いた分散更新則を以下に示す。

$$\sigma_{(d)} = \gamma^{(d)} \sigma_{(0)}. \quad (3.1)$$

ここで、 $\sigma_{(0)}$  は、同一制御周期内反復を行う際の分散  $\sigma$  の初期値であり、 $\gamma$  は、冷却スケジュールと呼ばれる定数パラメータである。SA の分野では、正規分布を提案分布として用いた場合に対数型アニーリングと呼ばれる、

$$\sigma_{(d)} = \frac{\sigma_{(0)}}{\log(d+2)}, \quad (3.2)$$

のような分散更新則を用いることで、推定解が最適化に到達できることが理論的に保証されている [68,69]。一方で、対数型アニーリング (3.1) では、初期温度から最終温度にもっていくには非常に膨大な反復回数が必要になるため、現実の応用には適さないとされている。そこで、SA では折衷案として、最適解への収束は犠牲にして、(3.1) で  $\gamma \in [0.8, 1.0)$  に設定した更新則が一般的に使用されている [68]。本論文でも、上記の考え方にに基づき、(3.1) で示す更新則を導入する。

冷却係数  $\gamma$  は、文献 [46] などによると、実時間のうちに実行可能な反復回数  $D$  と提案分布の分散の初期値  $\sigma_{(0)}$  及び同一制御周期内反復  $D$  回実行時における所望の分散の大きさ  $\sigma_{(D)}$  をもとに、

$$\gamma = \exp \left( \frac{1}{D} \log \left( \frac{\sigma_{(D)}}{\sigma_{(0)}} \right) \right) \quad (3.3)$$

とするのが良いとされている。

### 3.3. 双曲線型冷却法

以下に示すのが、双曲線型冷却法と呼ばれる分散更新則である。

$$\sigma_{(d)} = \frac{\sigma_{(0)}}{\sqrt{d+1}} \quad (3.4)$$



ここで、2.4.1 項で定義した最適制御問題に対し (3.4) で示す更新則を導入した MCMPC を適用した場合に得られる重要な定理を以下に示す。

**定理 3.** 離散時間線形システムに対する有限時間最適制御問題 (2.4.1 節) に対して、同一制御周期内反復時に (3.4) に示す分散更新則を導入した MCMPC を適用した際の誤差ベクトル  $\mathbf{e}_{(d)}$  及び標本分散  $\Sigma_S$  について考える。このとき、任意の  $\sigma_{(0)} \in \mathbb{R}^+$  に対して、反復回数  $d \rightarrow \infty$  としたとき、誤差ベクトル  $\mathbf{e}_{(d)}$  及び  $\Sigma_S$  はともに 0 となる。

*Proof.* 以下では、任意の行列  $X$  又はベクトル  $\mathbf{v}$  に対するユークリッドノルムをそれぞれ  $\|X\|$ ,  $\|\mathbf{v}\|$  と定義する。  $D$  回目の反復における誤差ベクトルのユークリッドノルムについて

$$\|\mathbf{e}_{(D)}\| \leq \prod_{d=0}^D \left\| (\hat{\sigma}_{(d)}^2 \hat{Q} + \mathbf{E}_u)^{-1} \right\| \|\mathbf{e}_{(0)}\| \quad (3.5)$$

$$\hat{\sigma}_{(D)} = \frac{\sigma_{(0)}}{\sqrt{D+1}} \quad (3.6)$$

の関係が成立する。このとき、 $\hat{Q}$  が正定対称行列であることから、

$$\frac{1}{\hat{\sigma}_{(d)} \lambda_{\max}(\hat{Q}) + 1} \leq \|(\hat{\sigma}_{(d)}^2 \hat{Q} + \mathbf{E}_u)^{-1}\| \leq \frac{1}{\hat{\sigma}_{(d)} \lambda_{\min}(\hat{Q}) + 1} \quad (3.7)$$

という不等式が成立する。ここで、 $\lambda_{\max}(\tilde{X}) \in \mathbb{R}^+$  及び  $\lambda_{\min}(\tilde{X}) \in \mathbb{R}^+$  はそれぞれ任意の正定対称行列  $\tilde{X}$  の最大固有値又は最小固有値を表している。このとき、(3.4) 及び (3.7) より、(3.5) は、

$$\|\mathbf{e}_D\| \leq \|\mathbf{e}_0\| \prod_{d=0}^D \left( \frac{1}{\frac{\sigma_0^2 \lambda_{\min}(\hat{Q})}{d} + 1} \right) \quad (3.8)$$

とすることができる。さらに (3.8) はガンマ関数  $\Gamma(\cdot)$  を用いて

$$\Xi_D = \prod_{d=0}^D \left( \frac{1}{\frac{\sigma_0^2 \lambda_{\min}(\hat{Q})}{d+1} + 1} \right) = \frac{\Gamma(D+2)}{(\alpha+1) \frac{\Gamma(\alpha+D+2)}{\Gamma(\alpha+2)}}. \quad (3.9)$$

のように記述できる。ただし、 $\alpha = \hat{\sigma}_0^2 \lambda_{\min}(\hat{Q})$  である。ここで、0 と負の整数を除く任意の実数  $\beta$  と  $s \in \mathbb{N}$  に対し、

$$\lim_{s \rightarrow \infty} \frac{\Gamma(\beta+s)}{s^\beta \Gamma(s)} = 1 \quad (3.10)$$

となるガンマ関数の性質を利用すると、 $\alpha$  の定義及び (3.10) から (3.9) は

$$\begin{aligned} \lim_{\hat{D} \rightarrow \infty} \Xi_{\hat{D}} &= C_{\hat{D}} \lim_{\hat{D} \rightarrow \infty} \frac{\Gamma(\hat{D})}{\Gamma(\alpha + \hat{D})} \\ &= C_{\hat{D}} \lim_{\hat{D} \rightarrow \infty} \frac{\Gamma(\hat{D})}{\Gamma(\alpha + \hat{D})} \frac{\Gamma(\alpha + \hat{D})}{\hat{D}^\alpha \Gamma(\hat{D})} \end{aligned}$$

$$\begin{aligned}
&= C_{\hat{D}} \lim_{\hat{D} \rightarrow \infty} \frac{1}{\hat{D}^\alpha} \\
&= 0
\end{aligned} \tag{3.11}$$

となる．ただし， $\hat{D} := D + 2$  であり，

$$C_{\hat{D}} := \frac{\Gamma(\alpha + 2)}{\alpha + 1}.$$

としている．(3.5)～(3.11) に示す結果より，双曲線形冷却法 (3.4) を導入した MCMPC の同一制御周期内反復の結果，反復回数  $D \rightarrow \infty$  とした時に，誤差ベクトル  $\mathbf{e}_{(D)}$  の期待値が 0 となることを示した．

次に，同様の状況で，標本分散  $\Sigma_S$  が 0 となることを示す．このとき，反復回数  $D \rightarrow \infty$  とした時の標本分散は，

$$\lim_{D \rightarrow \infty} \Sigma_S = \lim_{D \rightarrow \infty} \frac{\sigma_D^2 \eta}{L} (\sigma_D^2 \hat{Q} + \eta \mathbf{E}_u)^{-1} = 0. \tag{3.12}$$

となる．ここで (3.12) では，(3.4) より

$$\lim_{D \rightarrow \infty} \sigma_D = 0 \tag{3.13}$$

となることを利用している．(3.12) は，誤差ベクトル  $\mathbf{e}_{(D)}$  が 0 となるタイミング，すなわち  $D \rightarrow \infty$  の時に，推定値  $\tilde{\mathbf{u}}_{(D)}$  のばらつきを表す標本分散  $\Sigma_S$  が 0 となることを示している．以上の結果より，双曲線型冷却法を用いた MCMPC では，任意の  $\sigma_0 \in \mathbb{R}^+$  に対して，反復回数  $d \rightarrow \infty$  としたとき，誤差ベクトル  $\mathbf{e}_d$  と標本分散  $\Sigma_S$  はともに 0 となる．□

### 3.4. 非線形系への適用可能性

ここまで，2 章や本章 3.2 節及び 3.3 節で述べた内容は，MCMPC を用いた非線形系の制御においても有効であると考えられる．なぜなら，非線形系における収束性解析には，文献 [27,45,51] などさまざまなバリエーションがあるが，いずれにおいても (2.6) に示す加重平均の形で推定値が得られるという点で，今回我々が考察しているものと共通しているからである．また，文献 [51] では特殊な反復規則に従って，経路コストの期待値を逐次的に更新していくと，最適経路コストに収束するという事実に基づく，非線形系に対する (2.6) を用いた推定値計算方法の収束性解析の結果が示されている．この結果は，推定入力値の期待値ベースで考えるか，経路コストの期待値ベースで考えるかの違いはあるものの本質的には等価であると考えられる．

そこで、本章では、2.4.2 節の収束性解析の結果の非線形系への適用可能性、非線形系に適用した際の提案分布のパラメータが収束性に与える影響についても数値シミュレーションの結果をもとに考察する。

### 3.5. 数値シミュレーション

ここでは、(3.1) 及び (3.4) で示した冷却法を適用した際の効果について、線形系・非線形系への適用結果を示し考察する。以下では、検証に用いたモデル及び最適制御の問題設定について簡単に記述する。

なお、線形系での検証に用いたモデル及び最適制御の問題設定は、2.5 節の **Problem 1** で示したものと同様である。

#### Problem 2. 台車型倒立振子の振り上げ安定化制御

次に、非線形系への適用可能性を確認するための数値シミュレーションに用いるモデルと評価関数について記述する。制御対象は、次式に示す運動方程式で記述される台車型倒立振子とする [70]。

$$(m_c + m_p)\ddot{z} + m_p l \ddot{\theta} \cos \theta = -\mu_c \dot{z} + m_p l \dot{\theta}^2 \sin \theta + f_c \quad (3.14)$$

$$m_p l \ddot{z} \cos \theta + (J_p + m_p l^2) \ddot{\theta} = -\mu_p \dot{\theta} + m_p g l \sin \theta \quad (3.15)$$

ここで、 $z$ ,  $\dot{z}$ ,  $\ddot{z}$  はそれぞれ台車位置、速度、加速度であり、 $\theta$ ,  $\dot{\theta}$ ,  $\ddot{\theta}$  はそれぞれ振子角度、振子角速度、振子角加速度を表している。その他の (3.14) 及び (3.15) 中の定数または変数の定義については、Fig. 3.1 及び Table 3.1 に示す通りである。

状態ベクトルは、

$$x_{\{|\cdot\}}^T = \begin{bmatrix} z & \dot{z} & \theta & \dot{\theta} \end{bmatrix}$$

とし、制御の目的は、初期状態

$$x_{\{0|0\}}^T = \begin{bmatrix} 0 & 0 & \pi & 0 \end{bmatrix}$$

から目標状態（倒立状態）を示す、

$$x_{obj}^T = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}$$

に状態を安定させることとする。

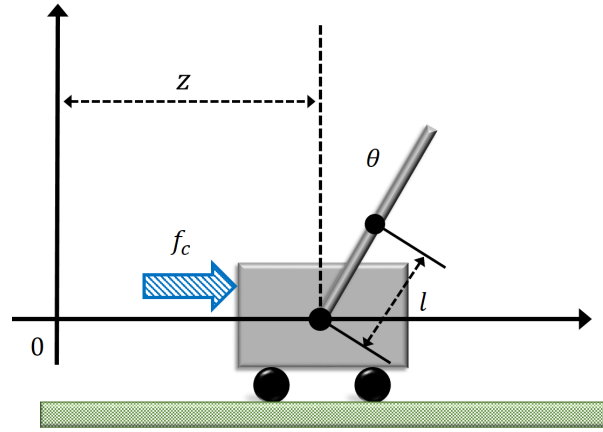


Fig. 3.1: Schematic diagram of the inverted pendulum on a cart

Table 3.1. Parameters of cart type inverted pendulum (Fig. 3.1)

Name	Symbol	Value
Length of the pendulum	$l$	0.2 m
Mass of cart	$m_c$	0.1 kg
Mass of pendulum	$m_p$	0.024 kg
Viscous friction of the cart	$\mu_c$	1.45
Viscous friction of the pendulum	$\mu_p$	0.000001
Gravitational acceleration	$g$	9.81 m/s <sup>2</sup>
Moment of inertia	$J_p$	$\frac{m_p l^2}{3}$ kg/m <sup>2</sup>

評価関数は, (2.12) と同じ状態・入力についての 2 次形式のものを使用し, 状態・入力それぞれに関する重み行列を

$$Q = \text{diag}(1.75, 0.03, 1.75, 0.04)$$

$$R = 1.0$$

とした。ただし, 評価関数の演算に使用する状態ベクトルは, 振子角度  $\theta$  の“巻き戻し現象 [71]’”を防止する観点から,  $\theta \in [-\pi, \pi)$  となる多価関数を用いて

$$x_{\{\cdot\}}^T = \begin{bmatrix} z & \dot{z} & \sin(\frac{\theta}{2}) & \dot{\theta} \end{bmatrix}$$

とした。また, サンプルング周期は 100 Hz, 予測ホライズン (ステップ数) 1.0 s ( $T_d = 100$ ) である。

### 3.5.1 検証内容

本章では、以下に示す3つの異なる条件のMCMPC間での最適化及び制御性能の比較検証を行なった。

**Normal** 分散更新則適用なし

**Geometric** 分散更新則として幾何冷却 (3.1) を適用

**Hyperbolic** 分散更新則として双曲線型冷却法 (3.4) を適用

また、最適化性能については、次式に示す指標を基に比較を行なった。

$$e_{\{k|0\}} = u_{\{k|0\}}^* - \tilde{u}_{\{k|0\}} \quad (3.16)$$

ここで、 $\tilde{u}_{\{k|0\}}$  及び  $u_{\{k|0\}}^*$  はそれぞれ、当該制御時刻  $k$  において実際にシステムに投入する入力 of MCMPC を用いた推定値及び最適値である。なお、最適値  $u_{\{k|0\}}^*$  は線形系の場合は解析的に導出出来るのでそれを用い、非線形系においては、文献 [46] に倣って、ある状態から TOP1 sample 法を用いた MCMPC に、冷却係数  $\gamma = 0.99$  に設定した幾何冷却法を適用し 10000 回の同一制御周期内反復を行なった時の推定値としている。

### 3.5.2 Problem 1 での検証結果

本項での検証内容は、線形系への適用における各種冷却法の有効性についてである。Fig. 3.2 及び Fig. 3.3 はそれぞれ、提案分布の分散の初期値を  $\sigma_{(0)} = 0.5$  と  $\sigma_{(0)} = 1.0$  に設定し、**Problem 1** で述べた問題設定のもと制御時刻  $k = 0$  において同一制御周期内反復のみを繰り返し 80 回行う試行を 1 試行とし、計 50 試行行なったときの予測ステップ  $\tau = 0$  における推定値  $\tilde{u}_{\{0|0\}}$  と最適値  $u_{\{0|0\}}^*$  との間の誤差  $e_{\{0|0\}} = u_{\{0|0\}}^* - \tilde{u}$  の標本平均（青色実線）と標本標準偏差  $3\sigma$ （水色の塗りつぶし）、ある 1 回の試行における反復ごとの誤差の推移（黒色破線）をプロットしたものである。

Fig. 3.2 では、冷却なし (Fig. 3.2(a))、冷却係数を  $\gamma = 0.95$  とした幾何冷却法を適用 (Fig. 3.2(c)) した結果がおおよそ 10 回の反復で、誤差 0 近傍に収束している。一方、解のばらつきに対応する水色の領域の大きさを見ると Fig. 3.2(c) の方が小さく抑えることができています。双曲線型冷却法 (Fig. 3.2(b)) は、分散の初期値を小さくしすぎた影響で収束が遅くなり、反復 70 回程度まで収束しておらず、ばらつきについても何もしていないものと同程度となった。また、冷却係数を  $\gamma = 0.8$  に設定した幾何冷却法 (Fig. 3.2(d)) で

は、収束する前に解の更新が行なわれなくなっており、誤差が残ったままとなった。併せて、解のばらつきについても最も大きくなった。

初期分散を大きく設定した Fig. 3.3 では、冷却係を  $\gamma = 0.95$  と設定した幾何冷却法 (Fig. 3.3(c)) 及び双曲線型冷却法 (Fig. 3.3(b)) の双方で、約 10 回程度の反復で誤差と解のばらつきの双方をほぼ 0 とすることができている。一方で、冷却係を  $\gamma = 0.8$  と設定した幾何冷却法 (Fig. 3.3(d)) では、僅かにではあるが誤差が生じており、解のばらつきについても上述した 2 例より大きくなっている。また、冷却法適用なしについては、誤差の標本平均は 0 近傍にあるが、解のばらつきは非常に大きくなっている。

本例題において、サンプル数を  $L = 3000$ 、制御周期を 100 Hz とすると、実時間のうちに反復可能な回数は 15 17 回程度であり、反復回数を 15 回に固定したとすると、Fig. 3.3(c) 及び Fig. 3.3(b) の結果が最も良いと言える。

以上を踏まえると、2 章で示した系 2 の関係も考慮し、初期分散を大きめに設定し、冷却係数を  $\gamma = 0.95$  程度に設定した幾何冷却を用いるか、同じく初期分散を大きめに設定し、双曲線型冷却を適用するのが制御性能向上の観点で良い結果を与えることになると言える。また、幾何冷却については、初期分散と冷却係数の決め方次第で所望の結果が得られなくなるので注意が必要である。

### 3.5.3 Problem 2 での検証結果

Fig. 3.4(a) は、**Problem 2** に、冷却法適用なしの MCMPC、幾何冷却を適用した MCMPC、双曲線型冷却法を適用した MCMPC について、冷却法適用なしの MCMPC でシミュレーションした Fig. 3.5 における制御時刻  $t = 0.69\text{s}$  ( $k = 69$ ) 時点の状態量を初期値として、同一制御周期内反復を行なった際の誤差  $e_{\{69|0\}} = \tilde{u}_{\{69|0\}} - u_{\{69|0\}}^*$  について、異なる条件ごとに 50 試行行なった際の標本平均（各色実線）と標本標準偏差（各色の塗りつぶし）をプロットしたものである。シミュレーションに用いた MCMPC のパラメータは Table 3.2 に示す通りである。非線形系への適用時は、冷却法なしの MCMPC では、誤差を伴ったまま解の更新が行なわれなくなっているのが分かる。これは、冷却を行なわない場合は、2 章で示したような収束性は期待できないことを示唆していると言える。また、幾何冷却法を適用したものについても、解のばらつきが大きく、誤差の標本平均も 0 に収束していない。これは、冷却係数を  $\gamma = 0.84$  に設定したため、冷却が急速に行なわれすぎたためであると考えられる。一方、双曲線型冷却法では、反復回数 500 回あたりで、誤差の標本平均が 0 に収束し、それ以降の標本標準偏差の大きさもほとんど 0 となっている。

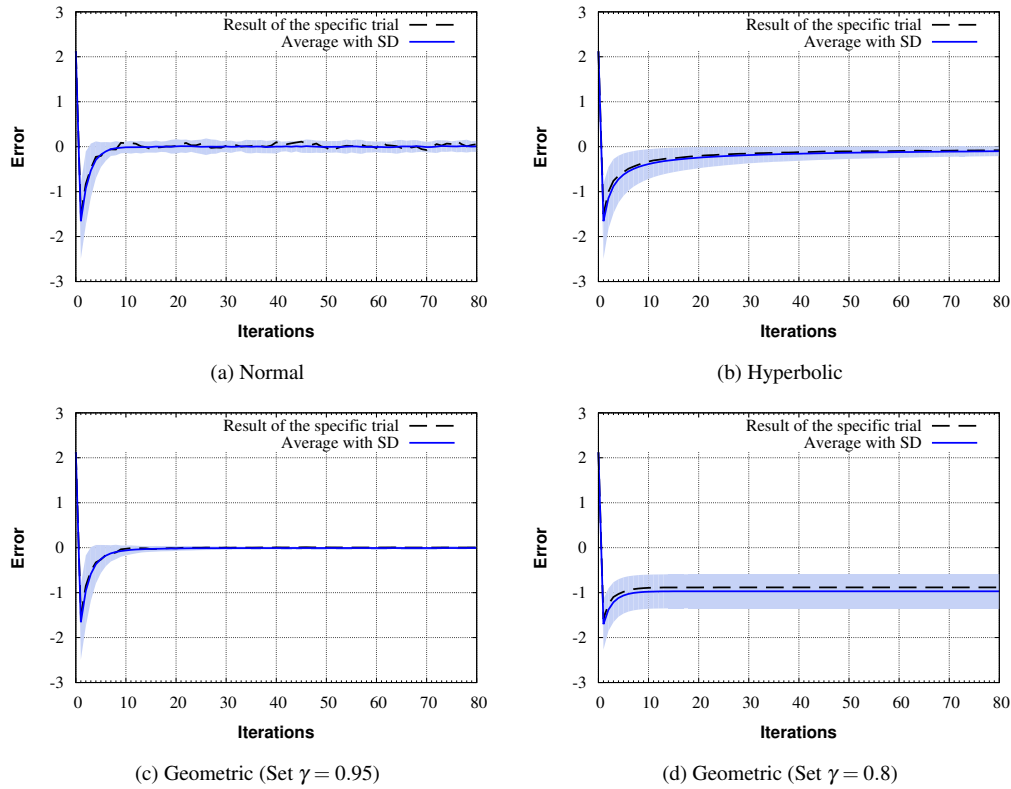


Fig. 3.2: Comparison of estimation errors  $e_{\{0|0\}} = u_{\{0|0\}}^* - \tilde{u}_{\{0|0\}}$  in **Problem 1** with or without cooling method (Set  $\sigma = 0.5$ )

Fig. 3.4(b) は 50 試行の内の誤差が最大となった（最悪ケース）をプロットしたものである．拡大図が示すように，反復 30 回目までは，幾何冷却法の方が，誤差が小さくなっている．一方で，幾何冷却法の方は，30 回目以降ほとんど解の更新が行なわれておらず，最終的な誤差は，双曲線型冷却法より大きくなっている．

Table 3.3 は，**Problem 2** において，振子角度が  $|\theta| \leq 0.1 \text{ rad}$  または  $|\theta - 2\pi| \leq 0.1 \text{ rad}$  となり，かつ，評価値が  $J \leq 60$  となった状態が 0.5 s 以上継続したうちの最初の時刻を整定時刻と定義し，冷却法の有無及び Elite sample MCMPC におけるサンプルの採択率（上位何 % を採択したか）ごとの整定時刻の全 50 回試行の平均値と最悪値を示している．採択率を 0.01～0.20 としたときは，幾何冷却法がいずれのケースにおいても平均値，最悪値ともに最短時間となっている．また，いずれの採択率においても，冷却法適用なしが，もっとも遅くなっている．この結果は，冷却法適用により，最適化の性能が向上したことを示唆していると言える．

以上の結果から，系 2 で考察した結果を考慮し，分散を冷却法により更新する本章の提案手法は，制御性能向上に寄与する方法であることが確認できた．これは，誤差の収束に合わせ，提案分布の分散を小さくすることで，より真の最適化に近い点に集中的に

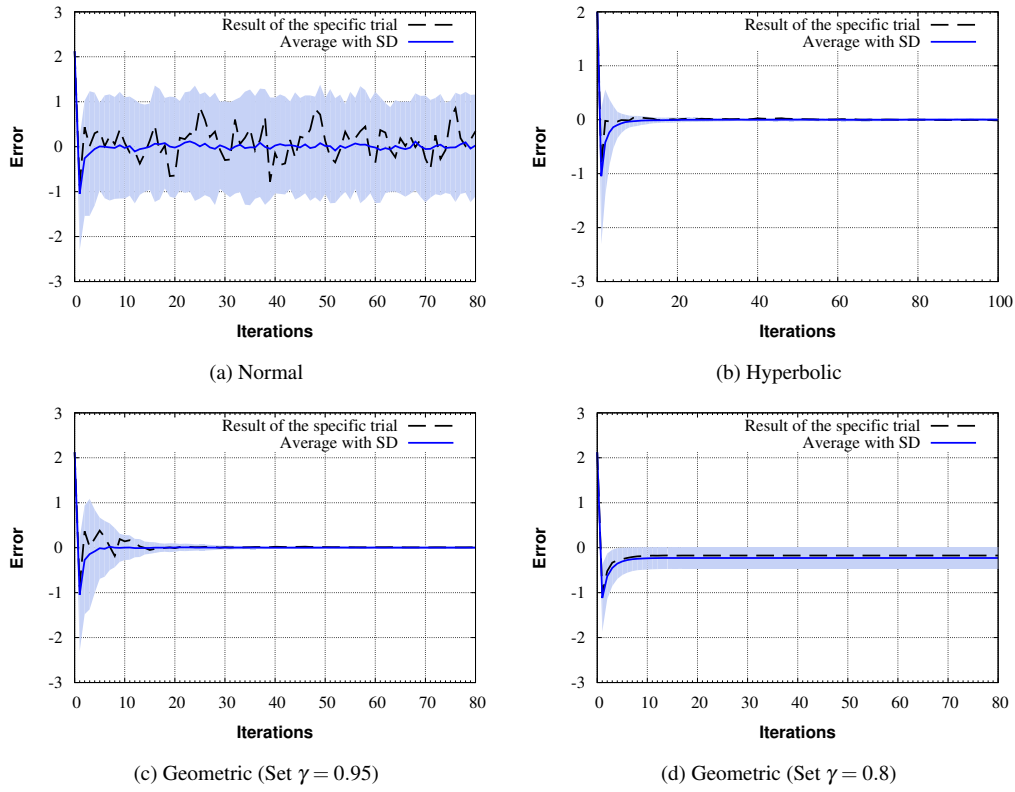


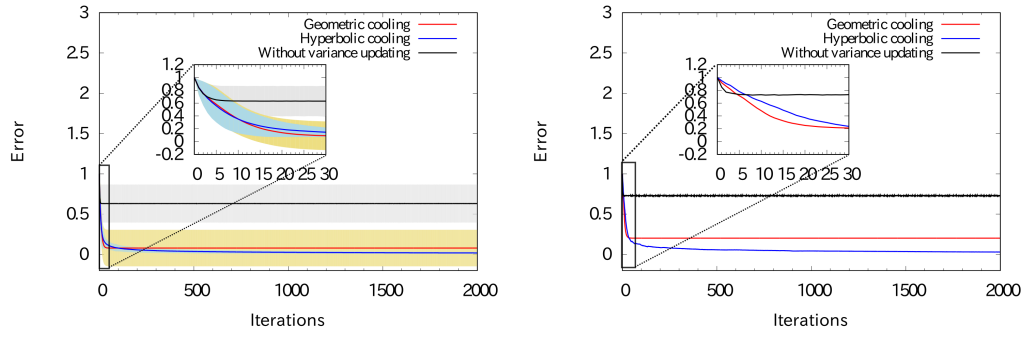
Fig. 3.3: Comparison of estimation errors  $e_{\{0|0\}} = u_{\{0|0\}}^* - \tilde{u}_{\{0|0\}}$  in **Problem 1** with or without cooling method (Set  $\sigma = 1.0$ )

サンプルが行えるようになったためであると考えられる。一方で、幾何冷却法及び双曲線型冷却法の効果については、十分な反復が行える場合は、双曲線型冷却法がより優れた手法であり、実時間性を考慮した 10 回程度の反復では、冷却係数を適切に定めた幾何冷却法と双曲線型冷却法であまり差がないという結果になった。

Table 3.2. Design parameters for MCMPC in **Problem 2**

Simulation time	10 s
Control cycle	100 Hz
Prediction horizon (Steps)	1.0 s(100steps)
Number of samples	3,000
Number of iteration	10
$\sigma_{(0)}$	1.2
$\eta$	6000
$\gamma$	0.95





(a) Sample mean (solid line) and sample standard deviation  $3\sigma$  (fill) of 50 trials

(b) Comparison of worst-case among 50 trails

Fig. 3.4: Comparison of convergence rate and estimation accuracy by using the  $e_{\{0.69|0\}}$  (Without variance updating vs. with geometric cooling vs. with hyperbolic cooling)

Table 3.3. Comparison of settling times according to acceptance rates of elite samples and different cooling patterns (average and worst case on 50 trials).

		Cooling method					
		Without cooling		Geometric cooling		Hyperbolic cooling	
		Average	Worst	Average	Worst	Average	Worst
Acceptance rate	0.01	3.59 s	4.34 s	2.84 s	3.01 s	2.86 s	3.06 s
	0.05	3.75 s	4.98 s	2.94 s	3.06 s	3.24 s	3.43 s
	0.10	5.92 s	7.65 s	3.16 s	3.36 s	3.38 s	3.92 s
	0.20	6.79 s	7.51 s	3.38 s	3.92 s	3.63 s	4.06 s
	0.25	7.72 s	9.52 s	4.19 s	4.41 s	3.83 s	4.33 s
	0.50	No data	No data	6.06 s	7.04 s	6.50 s	8.17 s

### 3.6. MCMPCのパラメータ設計指針

本節では、2章及び本章ここまでにおいて示した結果及び文献 [46] の内容をふまえた、MCMPC のユーザパラメータの設計指針について記述する。MCMPC を実装するのにあたって、一般的な MPC の問題設定とは別に、設計すべきパラメータは、大きく分けると、提案分布から抽出する総サンプル数  $L$  及び選り抜き標本数  $\hat{L}$ 、提案分布の分散共分散行列  $\Sigma$ 、冷却法適用の有無と種類、同一制御周期内反復の総反復回数  $D$  である。

このうち、サンプル数  $L$  と反復回数  $D$  については、 $D$  がなるべく大きくなる範囲で、かつ、1 制御周期のうちに所望の反復回数の演算ができる範囲でサンプル数を決めることになる。また、同一制御周期内反復 1 回あたりの計算時間は、状態・入力の次元やホライズンを分割した数  $T_d$  の大小にも大きく影響を受けることにも留意する必要がある。その他、サンプル数を増やすと、選り抜き標本を決定する際のソートの処理なども 1 反復あたりの処理時間が増加する要因となる。他方、2.6 節で示したように、サンプル数は、

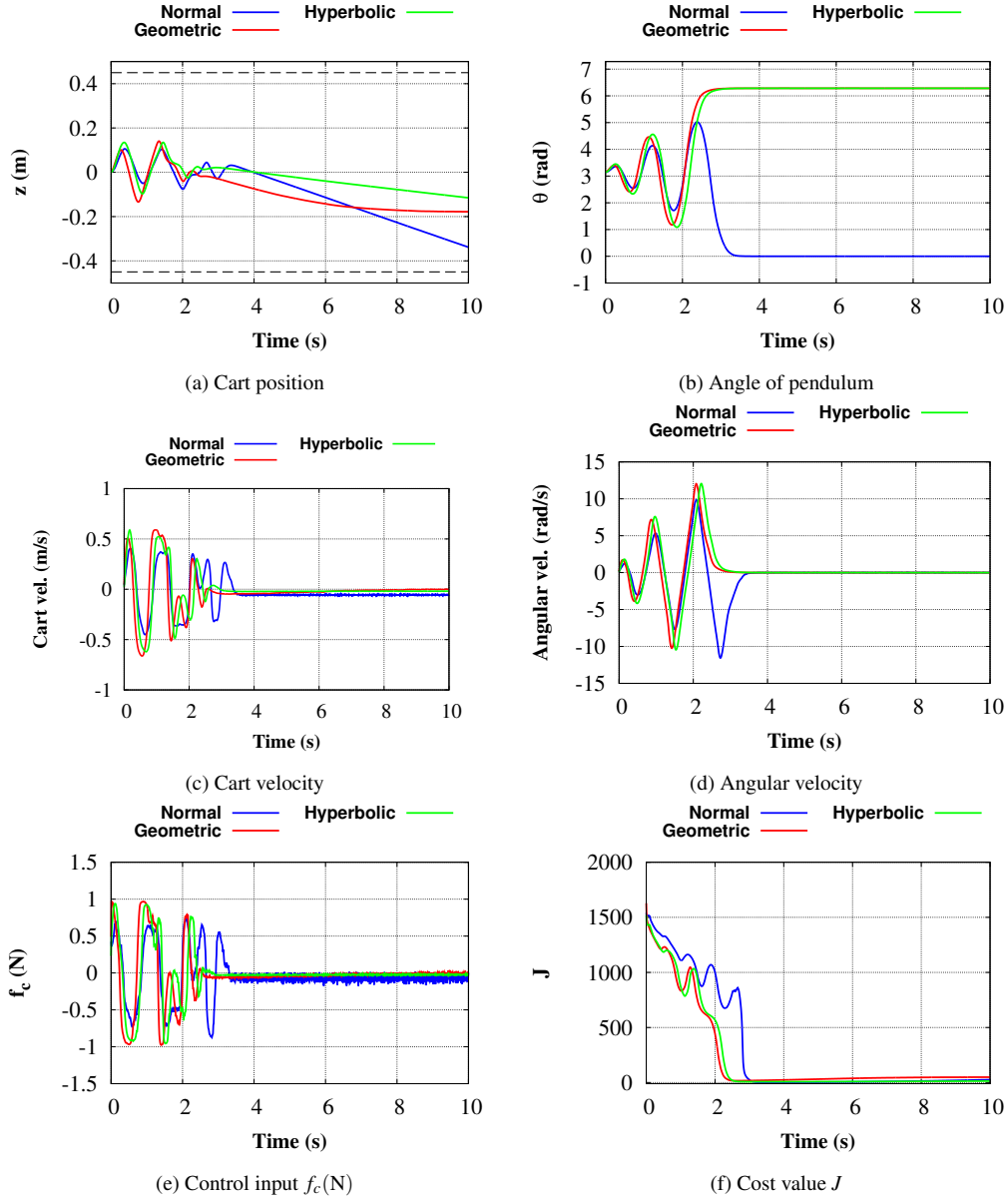


Fig. 3.5: MPC trajectories for **Problem 2**

推定値のばらつきを抑える効果はあるものの、推定値と最適解との誤差の収束速度には影響を与えない。これらを勘案し、MCMPCが適用された事例 [23,28,37,62,72] などを踏まえると、予測に用いる総サンプル数  $L$  は、1,000 ~ 10,000 の範囲で定めるのが一般的であると言える。

分散共分散行列  $\Sigma$  については、入力制約が、 $|u| \leq u_{max}$  で与えられ、対角行列を用いる場合であれば、文献 [46] のように、

$$\sigma_0 \geq \frac{u_{max}}{3} \quad (3.17)$$

とするのが良いとされる。(3.17)の意味する所は、提案分布の平均値が、ちょうど入力上

下限の中央値にあるとき、提案分布の  $3\sigma$  となる裾野が、実行可能入力が存在する領域をカバーするように設計するのが良いということである。また、系 2 より、大きく設定すると、収束速度が速くなるという性質があるので、(3.17) を参考に、少し大きめに設計するのが良い。

冷却法の有無については、数値例 3.5 で示したように、幾何冷却、双曲線型冷却のいずれを適用しても、適用なしの場合に比べ性能が向上するので、適用しないという選択肢は除外される。ついで、幾何冷却、双曲線型冷却のいずれを選択するかについてであるが、十分な同一制御周期内反復（例えば、20 回以上）が実行可能であれば、双曲線型冷却法に軍配が上がると言える。一方、十分な反復回数が実行できないのであれば、(3.3) を参考に、冷却係数  $\gamma$  を適切に定めた幾何冷却法を用いるのが良い。

以上を踏まえた、MCMPC のパラメータ設計手順の一例を以下に示す。

- 1) 使用する GPU のコア数などをもとに 1,000 ~ 10,000 の範囲でサンプル数  $L$  を決める。
- 2) 1 制御周期内に実行可能な同一制御周期内反復の回数を計測し、 $D$  とする。
- 3) (3.17) を参考に、分散の初期値  $\sigma_0$  を定める。
- 4)  $D$  に応じて、冷却法を選択する（幾何冷却法選択時は、(3.3) に従い、冷却係数  $\gamma$  を定める.）。

### 3.7. 3 章のまとめ

本章では、2 章系 2 で示した分散  $\sigma$  の設計による収束速度と推定精度のトレードオフ関係を解消する方法として、同一制御周期内反復において、分散  $\sigma$  を逐次更新する 2 種類の異なる分散更新方法（冷却法）を提案し、3.5 節でその有効性を確認した。特に、3.3 節では、双曲線型冷却法を MCMPC の同一制御周期内反復に導入した際に、(2.37) の漸化式で与えられる誤差と推定値のばらつき（標本分散）の双方を 0 に漸近収束させることができることを解析的に明らかにした。また、数値例では、一般的な MPC で対象とする非線形システムへの適用とその効果についての検証を行うため、ベンチマーク問題としてよく利用されている台車型倒立振子の振り上げ安定化制御（**Problem 2**）への適用結果を示した。幾何冷却法、双曲線形冷却法いずれを適用した場合でも、制御性能（**Problem 2** においては、整定までに要する時間を指標とした。）が向上していることを確認した。ま

た，十分な反復回数が得られる場合であれば，対象が線形／非線形いずれの場合においても双曲線形冷却法が最適化の面で最も優れていることを確認した．

一方で，“実時間”最適制御手法であるモデル予測制御では，限られたサンプリング時刻内に同一制御周期内反復を反復できる回数には上限がある．そのような反復回数の上限の問題まで考慮すると，最適解への収束性は犠牲になるが，3.2 節で提案した幾何冷却法であっても，双曲線型冷却法と遜色ない最適化性能を実現できることを確認した．また，3.6 節では，2 章の考察及びこれらの結果を踏まえた，MCMPC のパラメータ設計指針と，設計手順の一例を示した．

本章で提案した収束特性改善手法はいずれも，提案分布のパラメータのみに着目した手法であり，目的分布（評価値  $J$  を  $\exp(\cdot)$  で重み付けした分布）の形状情報を一切考慮することなく，推定値の精度を向上させることができるという利点がある．一方で，現実的な反復回数での実行となると，目的分布の形状を情報を活用したサンプリングなど，効率的なサンプル生成手法についても検討する必要があると考えられる．

## 第4章 分散共分散行列適応進化戦略を用いたサンプリング効率化

### 4.1. はじめに

3章では、2.4節の結果をもとに、MCMPC固有のパラメータのうち、入力列の生成に使用する正規分布の分散を更新することで、収束特性の改善を図る手法を提案した。一方、3章で提案した方法は、いずれも、2.4節の仮定1を満たす分散共分散行列(2.26)をもつ多変量正規分布を用いて入力列生成を行うMCMPCに対してのみ有効な手法である。仮定1を満たす分散共分散行列(2.26)を持つ正規分布からの入力列生成は容易である反面、提案分布と目的分布の形状に大きな隔たりがある場合には、最適化にほとんど寄与しない“無駄”なサンプルが多数生成される恐れがある。例えば、文献[49,73]などは、上述の仮定を満たすようなサンプル生成方法をナチュラル・サンプラーと呼び、最適化や関数近似にそのまま適用する際には、非常に非効率なサンプル生成手法であることに注意する必要があると指摘している。実際に、これらの文献では、ナチュラル・サンプラーは、目的関数の大まかな形状を知る手がかりとしてのみ利用し、その形状情報に基づき設計した補助サンプラーを用いて再度サンプリングを行うことで精度を向上させる方法が採用されている。

一方、サンプル型MPCの分野では、サンプル抽出自体は、仮定1を満たす、あるいは入力ごとに $\sigma$ の大きさだけを変えた対角行列を分散共分散行列にもつ多変量正規分布を用いて行い、サンプル毎の重みの設計[27,35]や重みによるサンプルの取捨選択[27,46]によって推定値の精度向上を図っているものが大多数を占めている。しかしながら、いずれの手法も、形状情報に基づくサンプルの再抽出は行なっておらず、生成したサンプルの良し悪しによって制御性能が大きく左右されるといった問題がある。また、MCMCやSAを用いたMPCのように、目的関数の形状情報に基づくサンプリングが行うものも存在するが、“徐々に”サンプルの精度を上げていくこれらの手法では、機械系などサンプリング周期が短いシステムへの適用を考えた際に、その緩慢な収束特性から実時間性に乏しいという問題がある。MCMPCなどのサンプル型MPCにおいて、実時間性を考慮

した上で、目的関数の形状情報を何らかの形で使用した効率的なサンプル生成が実現できれば、制御性能向上に大きく寄与することが容易に想像できる。

上述した要求を満たすサンプリング手法の1つに、分散共分散行列適応進化戦略 (Covariance matrix adaptation evolutionary strategy; CMA-ES) がある。CMA-ES は、候補となるサンプルを生成する正規分布の分散共分散行列をある規則に従い更新することで、悪スケール、多峰性などの種々の困難さをもつ連続最適化問題に対して効率的な探索ができる多点探索型の進化計算アルゴリズムである [64]。自然勾配法 [74] としても解釈することができ、逐次更新して得た分散共分散行列が目的関数の幾何的形状 (勾配などの情報) と深い関わりがあるとされている。例えば、秋本ら [75] は、CMA-ES による分散共分散行列の更新が、新たに生成するサンプルが Newton 方向に集中するように、サンプリングを行う (多次元) 正規分布を適切にスケールリングする処理に対応していることを明らかにしている。このような性質を上手く利用すれば、MCMPC などのサンプル型 MPC の大幅な性能向上が見込まれる。そこで、本章では、MCMPC などのサンプル型 MPC におけるモンテカルロ近似により生じる誤差を軽減するために、CMA-ES で用いられる分散共分散行列更新則を適用したアルゴリズムを提案し、その有効性を確認する。

本章の主な貢献は、2 章及び 3 章までの結果と考察を踏まえて、MCMPC の収束特性と最も深い関わりのある提案分布の分散共分散行列の更新要領として、CMA-ES を用いたアルゴリズムを提案し、提案法による高効率なサンプリングの実現について検証を行ったことである。また、本章で提案する方法は、1.2.1.2 で挙げた、サンプル型 MPC におけるそのアルゴリズム由来の最適化における誤差が原因となる、制御入力が高周波雑音のような波形となる問題の抑制効果については、十分な効果が得られなかったという課題も指摘し、それについての解決策についても検討・考察を行なった。

本章の構成は、以下に示す通りである。はじめに、4.2 節にて、本章で提案する手法において、分散共分散の更新に用いる CMA-ES のアルゴリズムについて記述する。ついで、4.3 節では、CMA-ES による分散共分散行列の更新と MCMPC のアルゴリズムを統合した本章の提案アルゴリズムについて記述する。4.4 節の評価実験では、異なる 2 種類の制御対象に対し、提案手法と他のサンプル型 MPC の代表的なアルゴリズムを適用した結果を用いて比較および評価を行う。ここでの主な評価項目は、効率的なサンプリングができているか、制御入力が高周波雑音のような振動的な波形になる問題に対して十分な抑制効果が得られているかの 2 項目である。最後に、4.5 節に、本章のまとめと今後検討が必要な課題などを記述する。

## 4.2. $(\mu/\mu_w, \lambda)$ -CMA-ES 型分散共分散行列更新則

以下では、CMA-ES 型分散共分散行列更新則を導入した MCMPC の同一制御周期内反復  $d$  回目のサンプル生成に利用する多変量正規分布を  $\mathcal{N}(\bar{\mathbf{u}}^{(d)}, (\zeta^{(d)})^2 \mathbf{C}^{(d)})$  と記述することにする．ここで、 $\zeta > 0$  はステップサイズと呼ばれるパラメータであり、 $\mathbf{C}^{(d)}$  は分散共分散行列である．

はじめに、初期パラメータとして  $\bar{\mathbf{u}}_{(d)}$ ,  $\zeta_{(d)}^2$ ,  $\mathbf{C}_{(d)}$  を探索領域の大きさに応じて定め、進化パスと呼ばれる  $\mathbf{p}_c^{(0)}$  及び  $\mathbf{p}_\zeta^{(0)}$  を  $\mathbf{p}_c^{(0)} = \mathbf{p}_\zeta^{(0)} = \mathbf{0}$  のように初期化する．以降、所定の反復回数だけ、次に示す Step 1 から Step 4 を繰り返し実行する．

**Step 1** 正規分布  $\mathcal{N}(\mathbf{0}, \mathbf{E})$  から  $\lambda$  個の独立なサンプル  $(\mathbf{z}_l)_{l=1, \dots, \lambda}$  を生成し、

$$\hat{\mathbf{u}}_l^{(d)} = \bar{\mathbf{u}}_l^{(d)} + \zeta^{(d)} \mathbf{y}_l \quad (4.1)$$

$$\mathbf{y}_l = \sqrt{\mathbf{C}^{(d)}} \mathbf{z}_l \quad (4.2)$$

に従い、サンプル  $\hat{\mathbf{u}}_{1, \dots, \lambda}^{(d)}$  を生成する．ここで、 $\mathbf{E}$  は単位行列である．

**Step 2** サンプルごとに並列シミュレーション (2.3.2 項) を実行し、評価値  $J(\hat{\mathbf{u}}_{1, \dots, \lambda}^{(d)})$  を計算し、上位  $\mu$  個を昇順に  $\hat{\mathbf{u}}_{1: \mu}^{(d)}$ ,  $\mathbf{z}_{1: \mu}$  及び  $\mathbf{y}_{1: \mu}$  のように並べ替える．

**Step 3**  $(\mathbf{y}_{1: \mu})$  の加重平均

$$\mathbf{dy} = \sum_{l=1}^{\mu} \mathbf{W}_l \mathbf{y}_l \quad (4.3)$$

を計算し、進化パスを

$$\mathbf{p}_\zeta^{(d+1)} = (1 - c_\zeta) \mathbf{p}_\zeta^{(d)} + \sqrt{(c_\zeta(2 - c_\zeta)\mu_w)} \sqrt{\mathbf{C}^{(d)}}^{-1} \mathbf{dy} \quad (4.4)$$

$$\mathbf{p}_c^{(d+1)} = (1 - c_c) \mathbf{p}_c^{(d)} + \sqrt{(c_c(2 - c_c)\mu_w)} \mathbf{dy} \quad (4.5)$$

に従い更新する．ここで、 $c_\zeta$  と  $c_c$  は進化パスの学習率、 $\sqrt{\mathbf{C}^{(d)}}$  は  $\mathbf{C}^{(d)} = \left(\sqrt{\mathbf{C}^{(d)}}\right)^2$  を満たす対称行列、(4.3) 中の重み  $\mathbf{W}_l$  及び進化パスの学習率  $\mu_w$  はそれぞれ

$$\mathbf{W}_l = \frac{\exp\left(\frac{-J_l}{\lambda}\right)}{\sum_{l=1}^{\mu} \exp\left(\frac{-J_l}{\lambda}\right)} \quad (4.6)$$

$$\mu_w = \frac{1}{\sum_{l=1}^{\mu} \mathbf{W}_l^2} \quad (4.7)$$

と記述できるものとする．本研究で使用する重み (4.6) は、一般的な CMA-ES で用いられる重み [63,64,75] とは異なるが、重みに課される条件 [63,75] はすべて満足している．

**Step 4** パラメータ  $\mathbf{u}^{(d)}$ ,  $\zeta^{(d)}$  及び  $\mathbf{C}^{(d)}$  をそれぞれ次に示すように更新する.

$$\zeta^{(d+1)} = \zeta^{(d)} \exp \left( \frac{c_\zeta}{d_\zeta} \left( \frac{\|\mathbf{p}_\zeta^{(d+1)}\|}{\chi} - 1 \right) \right) \quad (4.8)$$

$$\mathbf{C}^{(d+1)} = \mathbf{C}^{(d)} + c_1 \left[ OP(\mathbf{p}_c^{(d+1)}) - \mathbf{C}^{(d)} \right] + c_\mu \sum_{l=1}^{\mu} \left[ OP(\mathbf{y}_l) - \mathbf{C}^{(d)} \right] \quad (4.9)$$

ここで,  $OP(\mathbf{v}) = \mathbf{v}\mathbf{v}^T$  は, ベクトル  $\mathbf{v}$  とそれ自身のテンソル積を表わしている. なお,  $\mathbf{u}^{(d)}$  の更新式は (2.6) と同様である. また,

$$\chi = \mathbb{E}[\mathcal{N}(\mathbf{0}, \mathbf{E})] \quad (4.10)$$

であり, 近似値として,

$$\chi \approx (mT)^{1/2} \left( 1 - \frac{1}{4(mT)} + \frac{1}{21(mT)^2} \right) \quad (4.11)$$

を用いるのが一般的である [64]. ただし,  $m$  は (2.1) における入力次元,  $T$  は予測ステップ数である. また, (4.8) 及び (4.9) 中の変数  $c_1$  及び  $c_\mu$  はそれぞれ共分散行列の rank-one 更新, rank- $\mu$  更新の学習率を示している.

Step 1 から Step 4 に示したアルゴリズムは, CMA-ES の中でも  $(\mu/\mu_w, \lambda)$ -CMA-ES と呼ばれる手法を MCMPC に導入するために独自にアレンジした手法である. ある条件下では, 上記 4 つの Step を反復することで, 3 つのパラメータからなる正規分布が, 最適解  $\mathbf{u}^*$  における Dirac-Delta 分布に収束することが文献 [75] で示されている.

### 4.3. $(\mu/\mu_w, \lambda)$ -CMA-ES を用いた MCMPC

ここでは, 4.2 節で紹介した分散共分散行列更新則を用いた MCMPC のアルゴリズムについて詳述する.

#### 4.3.1 アルゴリズム

Algorithm 1 は,  $(\mu/\mu_w, \lambda)$ -CMA-ES 型分散共分散行列更新則を用いた MCMPC のアルゴリズムの擬似コードである. MPC\_GPU 関数や分散共分散行列  $\mathbf{C}^{(d)}$  の固有値分解の処理など, 計算負荷が大きい処理は, GPU などの並列演算デバイスを用いることで十分高速化でき, 100 Hz 程度のサンプリング周期のうちに 5 ~ 6 回程度の反復処理が可能である.



---

**Algorithm 1** Monte Carlo Model Predictive Control with  $(\mu/\mu_w, \lambda)$ -CMA-ES
 

---

**Given:**  $\lambda$  : Number of random samples.  $n/m$  : Dimension of state / input.

$k$  : Current control cycle.  $x_{\{k|0\}}$  : Observed current state.

$l$  : Thread ID.  $T$  : Prediction steps.  $\mathbf{E}$  : Identity matrix.

$J$  : Cost value.  $\zeta$  : Evolutional variance.  $D$  : Number of iterations.

$\hat{\mathbf{u}}$  : Randomly sampled input sequences generated by (4.1)

$\bar{\mathbf{u}}$  : Copied estimation.  $\tilde{\mathbf{u}}$  : Estimation computed by (2.6).

$\mathbf{p}_\zeta^{(d)}, \mathbf{p}_c^{(d)}$  : Evolutional path.  $\mathbf{C}^{(d)}$  : Evolutional covariance matrix.

```

1:  $k \leftarrow 0$ 
2: Initialize  $\zeta^{(0)}, \mathbf{C}^{(0)}, \mathbf{p}_\zeta^{(0)}, \mathbf{p}_c^{(0)}, \bar{\mathbf{u}}^{(0)}$ 
3: while task not completed, do
4:   for  $d = 0$  to  $D$  do
5:     // Get  $\lambda$  simulation results calculated in parallel on GPU.
6:      $(\hat{\mathbf{u}}_{1:\lambda}, J_{1:\lambda}, \mathbf{y}_{1:\lambda}) \leftarrow \text{MPC\_GPU}(\lambda, x_{\{k|0\}}, \bar{\mathbf{u}}^{(d)}, \zeta^{(d)}, \mathbf{C}^{(d)})$ 
7:     // Sort up to  $\mu$ th in ascending order of  $J_{1:\lambda}$ .
8:      $(\hat{\mathbf{u}}_{1:\mu}, J_{1:\mu}, \mathbf{y}_{1:\mu}) \leftarrow \text{sort}(\hat{\mathbf{u}}_{1:\lambda}, J_{1:\lambda}, \mathbf{y}_{1:\lambda})$ 
9:      $\tilde{\mathbf{u}} \leftarrow \frac{\sum_{l=1}^{\mu} \hat{\mathbf{u}}_l \exp\left(-\frac{J_l}{\eta}\right)}{\sum_{l=1}^{\mu} \exp\left(-\frac{J_l}{\eta}\right)}$ 
10:     $\mathbf{dy} \leftarrow \frac{\sum_{l=1}^{\mu} \mathbf{y}_l \exp\left(-\frac{J_l}{\eta}\right)}{\sum_{l=1}^{\mu} \exp\left(-\frac{J_l}{\eta}\right)}$ 
11:    // Update  $[\mathbf{p}_\zeta^{(d)}, \mathbf{p}_c^{(d)}, \zeta^{(d)}, \mathbf{C}^{(d)}]$  using (4.4) to (4.9).
12:  end for
13:  Send  $\tilde{\mathbf{u}}_{\{k|0\}}$  to actuators. Copy  $\bar{\mathbf{u}}^{(d+1)} \leftarrow \tilde{\mathbf{u}}$ 
14:  Update observed state  $x_{\{k+1|0\}}$  after receiving feedback.
15:  Check for task completion.
16:   $k \leftarrow k + 1$ 
17: end while

```

// Function executed in parallel by  $\mathbf{I}$  GPU threads.

**function** MPC\_GPU( $\lambda, x_{\{k|0\}}, \bar{\mathbf{u}}^{(d)}, \zeta^{(d)}, \mathbf{C}^{(d)}$ )

// Execute random sampling.

$\mathbf{z}_l \sim \mathcal{N}(\mathbf{0}, \mathbf{E})$

$\mathbf{y}_l \leftarrow \sqrt{\mathbf{C}^{(d)}} \mathbf{z}_l$

$\hat{\mathbf{u}}_l \leftarrow \bar{\mathbf{u}}^{(d)} + \zeta^{(d)} \mathbf{y}_l$

// Execute forward-time-simulation

**for**  $\tau = 0$  to  $T$  **do**

$x_{l, \{k|\tau+1\}} = x_{l, \{k|\tau\}} + f(x_{l, \{k|\tau\}}, u_{l, \{k|\tau\}}) \Delta t$

**end for**

$J_l \leftarrow \phi(x_{l, \{k|T\}}) + \sum_{\tau=0}^{T-1} S(x_{l, \{k|\tau\}}, \hat{u}_{l, \{k|\tau\}}, \tau)$

**return**  $(\hat{\mathbf{u}}_l, J_l, \mathbf{y}_l)$

**end function**

---

Table 4.1. Recommended parameters for  $(\mu_{\mathbf{w}}, \lambda)$ -CMA-ES

Name	Symbol	formula	Value (For $mT_d = 100$ )	Value (For $mT_d = 110$ )
Minimum sample size	$\lambda_{\min}$	$4 + \lfloor 3 \ln(mT) \rfloor$	18	19
Number of elite samples	$\mu$	$\frac{4 + \lfloor 3 \ln(mT) \rfloor}{2}$	9	10
Learning rate	$c_\zeta$	$\frac{4}{mT + 4}$	0.038	0.035
	$c_c$	$\frac{4}{mT + 4}$	0.038	0.035
	$c_1$	$\sqrt{c_\zeta (2 - c_\zeta)}$	0.27	0.26
Damping parameter	$d_\zeta$	$c_\zeta^{-1} + 1$	27.32	29.57
$\mathbb{E}[\mathcal{N}(\mathbf{0}, \mathbf{E})]$	$\chi$	Eq.(4.11)	9.96	10.46

#### 4.3.2 パラメータの設計指針

$(\mu_{\mathbf{w}}, \lambda)$ -CMA-ES のユーザパラメータの推奨値を Table 4.1 に示す．Table 4.1 に示すパラメータについては，文献 [64] に示されているほか，文献 [63] では，それぞれのパラメータの意味や推奨値の根拠となる収束解析の結果が示されている．Table 4.1 を見てあきらかなように，推奨パラメータは，探索空間の次元数（=MPC の問題設定における入力次元と予測ステップ数の積  $mT$ ）を基に全て定めることができる．

また，探索領域に合わせた調整が必要な  $\bar{\mathbf{u}}^{(d)}$ ， $\zeta^{(d)}$ ， $\mathbf{C}^{(d)}$  などのパラメータの初期値については，2.6 節の考察を基に設定することができる．

#### 4.4. 評価実験

本章の目的は，サンプル型 MPC における効率的なサンプル生成方法を提案し，その効果を明らかにすることにある．そこで，本評価実験においては， $(\mu_{\mathbf{w}}, \lambda)$ -CMA-ES を適用しない IT-MPC [27] 及び MCMPC [46]，提案法であるサンプル生成に  $(\mu_{\mathbf{w}}, \lambda)$ -CMA-ES を用いた MCMPC の 3 つのパターンでの制御性能の比較を行なった．ここで，IT-MPC と通常の MCMPC との違いは，(2.6) を計算する際に，目的分布の重みを考慮した重点サンプリングを行っているか否かである．前者が重点サンプリングを行う方法，後者が重点

サンプリングを行わない手法である。つまり、本評価実験では、提案法であるサンプル生成に  $(\mu_w, \lambda)$ -CMA-ES を用いる方法が、重点サンプリングを用いた方法より優れた効果を発揮できるかについても確かめていることになる。

評価実験では、台車型倒立振子 (**Problem 3**) 及び台車型直列二重倒立振 (**Problem 4**) を制御対象とする最適制御問題に 3 つのパターンの制御手法を適用し、上述した内容について評価・検証を行う。

#### 4.4.1 最適制御問題の問題設定

はじめに、各対象に対する最適制御問題の問題設定について記述する。

##### **Problem 3.** 台車型倒立振子の振り上げ安定化制御

**Problem 3** の問題設定は、3 章 **Problem 2** と同様の台車型倒立振子振子の振り上げ安定化制御である。数理モデルにおける物理パラメータ等の設定も **Problem 2** と同様としている。

##### **Problem 4.** 台車型直列二重倒立振子の振り上げ安定化制御

Fig. 4.1 に示す台車型の二重倒立振子の運動方程式は、

$$I_\theta \ddot{\theta} + M_1 \cos(\theta - \phi) \ddot{\phi} + M_1 \dot{\phi}^2 \sin(\theta - \phi) + (c_1 + c_2) \dot{\theta} - c_2 \dot{\phi} - M_2 g \sin \theta + M_2 \ddot{z} \cos \theta = 0 \quad (4.12)$$

$$I_\phi \ddot{\phi} + M_1 \cos(\theta - \phi) \ddot{\theta} - M_1 \dot{\theta}^2 \sin(\theta - \phi) - c_2(\dot{\theta} - \dot{\phi}) - M_3 g \sin \phi + M_3 \ddot{z} \cos \phi = 0 \quad (4.13)$$

で表すことができる [76]。ここで、各係数は以下のように定めている。

$$I_\theta = I_1 + J_{n_1} + m_2 L^2 + n_2 L^2$$

$$I_\phi = I_2 + J_{n_2}$$

$$M_1 = m_2 l_2 L$$

$$M_2 = m_1 l_1 + m_2 L + n_2 L$$

$$M_3 = m_2 l_2$$

状態ベクトルは、

$$x_{\{1\}}^T = [z \quad \dot{z} \quad \theta \quad \dot{\theta} \quad \phi \quad \dot{\phi}]$$

である。ここで、 $z$  及び  $\dot{z}$  は台車の変位及び速度、 $\theta$  及び  $\dot{\theta}$  は第 1 リンクの鉛直上向きに対する角度及び角速度、 $\phi$  及び  $\dot{\phi}$  は、第 2 リンクの鉛直上向きに対する角度及び角速度

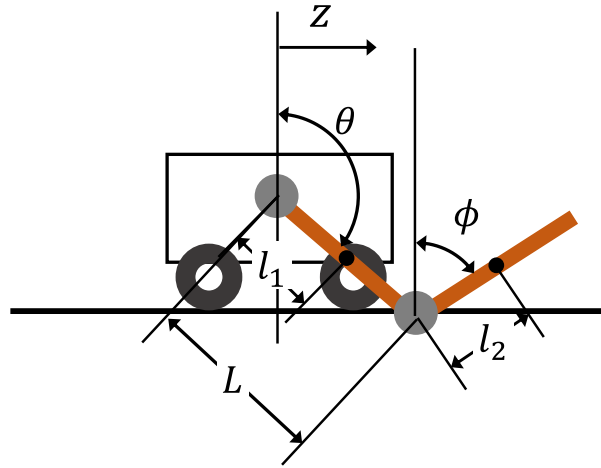


Fig. 4.1: Schematic diagram of the serial double inverted pendulum on a cart

を表している．その他の変数及びパラメータは，第 1 リンクと台車の間のジョイントの質量を  $n_1$ ，第 1 リンクと第 2 リンクの間のジョイントの質量を  $n_2$ ，各リンクの質量を  $m_1$  及び  $m_2$ ，各リンクの軸・重心間の距離を  $l_1$  及び  $l_2$ ，各リンクの重心回りの慣性モーメントを  $J_{n_1}$  及び  $J_{n_2}$ ，各リンクの回転軸回りの慣性モーメントを  $I_1$  及び  $I_2$ ，各リンクの回転摩擦係数を  $c_1$  及び  $c_2$ ，第 1 リンクの全長を  $L$ ，重力加速度を  $g$  としている．また，制御入力 は台車への印加加速度として  $u := \ddot{z}$  とする．

評価関数は (2.12) 同様，状態・入力について 2 次形式のものを採用し，

$$Q = \text{diag}(2.0, 0.01, 2.0, 0.01, 2.0, 0.01)$$

$$R = 0.0001$$

とした．また，状態及び入力に関する制約として

$$|z| \leq 0.45$$

$$|u| \leq 20.0$$

を与えた．サンプリング周期は 100 Hz，予測ステップ数  $T_d = 110$  とした．

#### 4.4.2 結果

Table 4.2 及び Table 4.3 は，それぞれ，異なるサンプルサイズにおいて **(Problem 3)** の台車型倒立振子及び **(Problem 4)** の台車型直列二重倒立振子の振り上げ安定化制御の数値シミュレーションを 30 回試行したときのタスクの成功率が初めて 100 % となったときのサ

サンプルのサイズを示している。ここで、本研究におけるタスクの成功とは、それぞれの制御対象において、振子が倒立状態となった状態が5 s以上継続することを意味している。また、Fig. 4.2～Fig. 4.7は、IT-MPC、MCMPC、及びAlgorithm 1に示す $(\mu/\mu_w, \lambda)$ -CMA-ESによる分散共分散行列更新則を導入したMCMPC (CMA-MCMPC)のそれぞれの手法で、タスク成功率が100 %となった条件のうち、最小のサンプルサイズを用いて行なった数値シミュレーションのある一試行の結果を示している。

Table 4.2 及び Table 4.3 双方の結果から明らかなように、CMA-MCMPCがいずれのタスクにおいても最小のサンプルサイズでタスク成功率100 %を実現することができている。特に、**Problem 3**での検証結果 (Fig. 4.2～Fig. 4.4 及び Table 4.2) では、この最適制御問題における最小サンプルサイズ (18 サンプル) でのタスク達成率100 %が実現できている。さらに、Fig. 4.4の振子角度の時間応答の結果に示す通り、18 サンプルしか使用していないCMA-MCMPC (Fig. 4.4(c)) が最も高速で振り上げ安定化に成功している。これは、評価関数の形状情報を考慮した効率的なサンプリングが行なわれた結果、(1.2) や (2.6) で示すモンテカルロ近似に利用されるサンプルの質が向上したためであると考えられる。

**Problem 4**での検証結果において、CMA-MCMPCの理論上の最小サンプルサイズと実験値が一致しなかったのは、**Problem 3**と比べ**Problem 4**のタスクの難易度が高く、評価関数の形状が複雑化したことが主な要因と考えられる。また、そのような状況において、実時間性を実現するために収束させるのに必要な十分な回数の反復を行うことができなかったことも影響していると考えられる。

Fig. 4.3 及び Fig. 4.6 に示す台車位置の制御結果においては、IT-MPC (Fig. 4.3(a) 及び Fig. 4.6(a)) 及び MCMPC (Fig. 4.3(b) 及び Fig. 4.6(b)) を用いた場合では、台車位置を目標値である0 m 近傍に収束させることができているが、CMA-MCMPC (Fig. 4.3(c) 及び Fig. 4.6(c)) の結果では、状態制約近傍に留まり続けている。この原因として、反復回数が十分でなかった他に、(4.8) や (4.9) で用いた $c_1$  や  $c_\mu$  などのパラメータを固定値としたことが影響していると考えられる。サンプリング周期ごとに状態量などの初期値が変化する最適制御問題を解き直すことになるMPCでは、時々刻々と評価関数 $J$ の形状や性質も変化していくと推察される。 $c_1$  や  $c_\mu$  は、評価関数が多峰性関数や稜線構造を有した関数であるなど悪条件となる場合に、局所解からの脱出や大域的な最適解への収束を手助けするパラメータであるとされている [64]。また、文献 [77] では、これらのパラメータについても、評価関数の性質に合わせ適応的に変化させる必要があることに言及されて

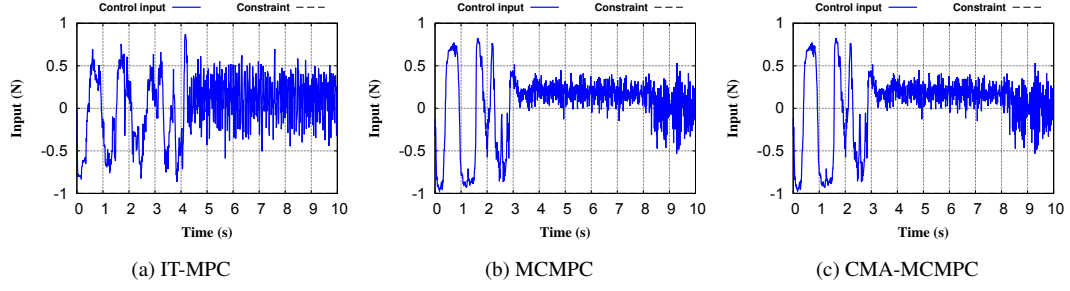


Fig. 4.2: Comparison of time response of control inputs in **Problem 3** when using the following different sample size; (a) and (b) used 50 samples, (c) used 18 samples.

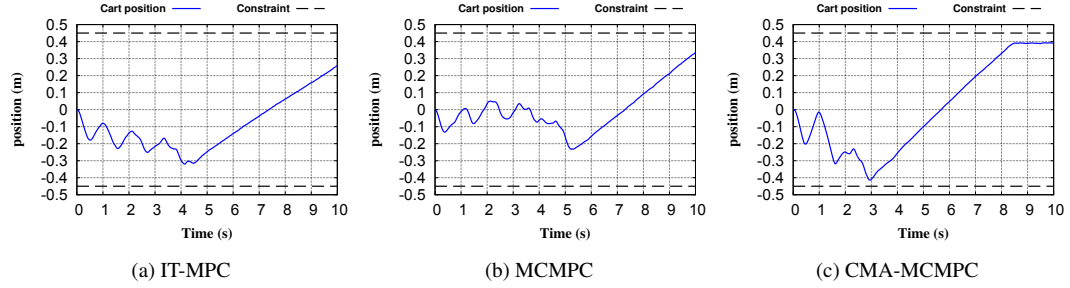


Fig. 4.3: Comparison of time response of cart position in **Problem 3** when using the following different sample size; (a) and (b) used 50 samples, (c) used 18 samples.

いる．CMA-MCMPC を用いる場合には，このような考え方を取り入れることも今後検討していく必要がある．

また，1.2.1.2 で挙げた，制御入力がガウス雑音のような振動的な波形になる現象の抑制効果について見ると，CMA-ES を適用しても (Fig. 4.2(c) 及び Fig. 4.5(c))，大きな効果は得られなかったことが分かる．これについても，ここまでの考察同様，過少な反復回数， $c_1$  や  $c_\mu$  などのパラメータの扱い方が原因であると考えられる．

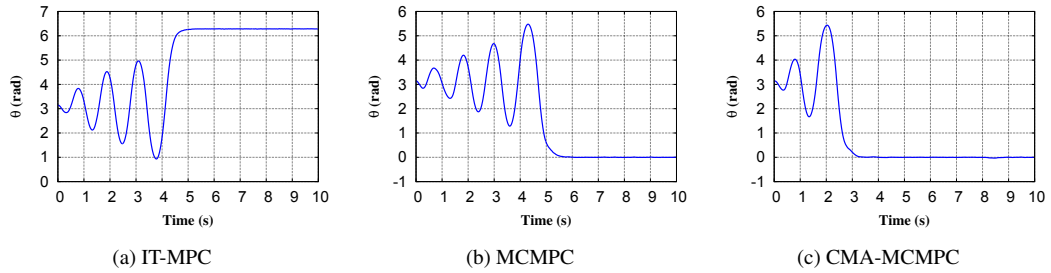


Fig. 4.4: Comparison of time response of pendulum angle in **Problem 3** when using the following different sample size; (a) and (b) used 50 samples, (c) used 18 samples.

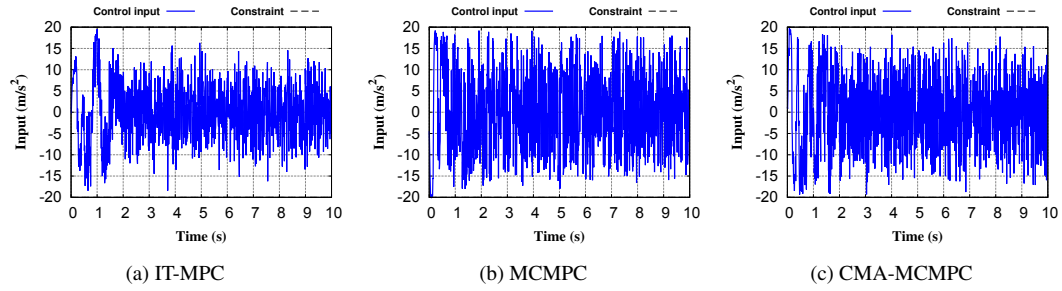


Fig. 4.5: Comparison of time response of control input in **Problem 4** when using the following different sample size; (a) used 3,300 samples, (b) used 550 samples, and (c) used 220 samples.

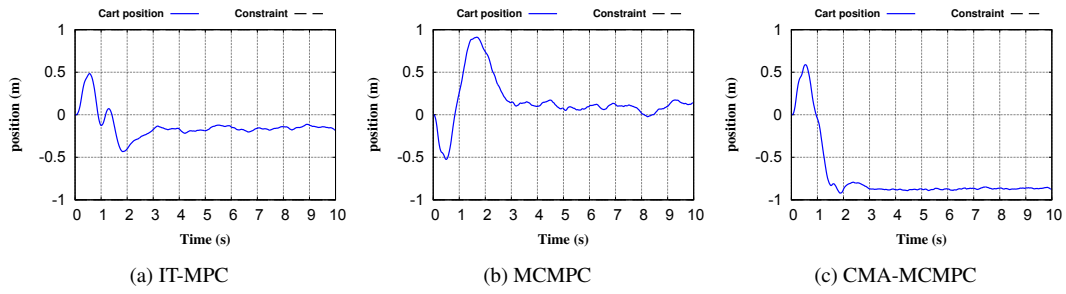


Fig. 4.6: Comparison of time response of cart position in **Problem 4** when using the following different sample size; (a) used 3,300 samples, (b) used 550 samples, and (c) used 220 samples.

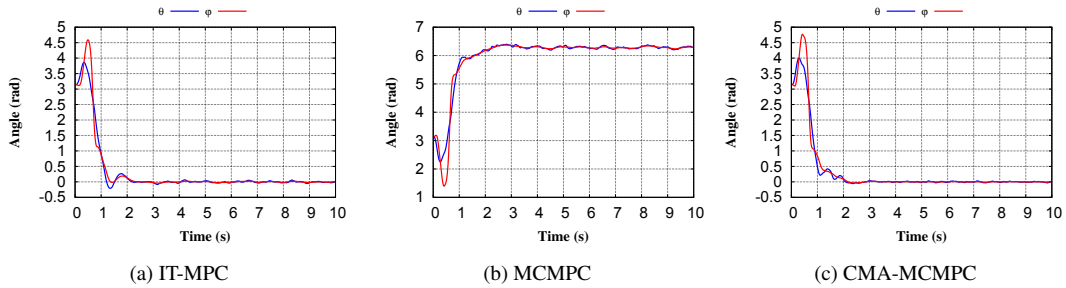


Fig. 4.7: Comparison of time response of pendulum angles  $\theta$  and  $\phi$  in **Problem 4** when using the following different sample size; (a) used 3,300 samples, (b) used 550 samples, and (c) used 220 samples.

Table 4.2. Task success rate by sample size (For **Problem 3**)

	Sample size		
	18 (= theoretical minimum)	30 ( $0.3mT_d$ )	50 ( $0.5mT_d$ )
IT-MPC	0 ( $\frac{0}{30}$ )	0 ( $\frac{0}{30}$ )	100 ( $\frac{30}{30}$ )
MCMPC	0 ( $\frac{0}{30}$ )	0 ( $\frac{0}{30}$ )	100 ( $\frac{30}{30}$ )
CMA-MPC	100 ( $\frac{30}{30}$ )	100 ( $\frac{30}{30}$ )	100 ( $\frac{30}{30}$ )

Table 4.3. Task success rate by sample size (For **Problem 4**)

	Sample size		
	220 ( $2mT_d$ )	550 ( $5mT_d$ )	3300 ( $33mT_d$ )
IT-MPC	0 ( $\frac{0}{30}$ )	0 ( $\frac{0}{30}$ )	100 ( $\frac{30}{30}$ )
MCMPC	0 ( $\frac{0}{30}$ )	17 ( $\frac{5}{30}$ )	100 ( $\frac{30}{30}$ )
CMA-MPC	100 ( $\frac{30}{30}$ )	100 ( $\frac{30}{30}$ )	100 ( $\frac{30}{30}$ )

#### 4.5. 4章のまとめ

本章では、3章で課題とした実時間性を考慮した上での近似精度改善手法として、目的関数の形状情報を考慮した分散共分散行列の更新則を取り入れたサンプル生成手法による高効率サンプリングについて検討した。

本研究で提案した自然勾配法と深い関わりのある CMA-ES を、分散共分散行列の更新に用いる本手法では、従来法と比べ、少数のサンプルであっても同様のタスクを実現できるなど、効率的なサンプリングを行う上では、有効な手法であることが確認できた。一方で、制御性能向上の観点では、特に制御入力が高ス雑音のような振動的な波形になる現象の抑制効果が小さいことや状態量が目標値以外のところに収束してしまうケースがあるなど、十分な効果を得ることが出来なかった。これについては、4.4.2でも考察したとおり、実時間性の制約のため、反復回数が過少であったことや、CMA-ES に係るパラメータの更新要領などが不十分であったためであると考えられる。今後、本章の提案手法である CMA-ES を用いた MCMPC による更なる制御性能向上を実現していくためには、MPC で扱う最適制御問題の性質に沿った CMA-ES のパラメータ更新則などについて先行研究 [77] を参考に検討するほか、重点サンプリングの考え方を取り入れている IT-MPC [27] に本章で示した方法を適用することなども検討する必要があると考えられる。

一方、ここまで、2章、3章及び本章の結果より、確率・統計的最適化手法のみを用いての制御性能向上には限界があると考えられることができる。これは、MPC に課される“実時間



性”の制約が主な原因であり，これにより，反復回数やサンプル数などが過少となり，十分な精度の解が得られないためである．この問題については，関連研究である Homanga ら [52] のように，決定論的最適化手法と確率・統計的最適化手法を組み合わせる手法が一つの解決策となり得る．しかしながら，勾配やヘシアンなどの解析的な微分情報が必須となる決定論的最適化手法をそのまま取り入れるのは，我々の研究目的である，より実用的な MPC の制御アルゴリズムの構築とは相反することになる．そこで，この点に留意して，文献 [78,79] のように，何らかの確率・統計的な手法を用いて，勾配やヘシアンなどの微分情報を近似する手法について検討する必要がある．

## 第5章 サンプルベースニュートン型最適化 手法を用いた収束特性の改善

### 5.1. はじめに

確率・統計的最適化手法を用いて最適制御問題を解くサンプル型 MPC には、そのアルゴリズム由来のランダム性の影響により、実際にシステムに加える制御入力が高周波雑音のような振動的な波形となるといった欠点が存在する。実際に、3 章や 4 章でみたように、特に定常状態においてその影響が顕著になる。

この問題を解決するために、Homanga ら [52] は、大域的な探索に、確率・統計的最適化手法を用い、局所的な最適化には決定論的最適化手法である最急降下法を使用する方法を提案し、勾配型 MPC やサンプル型 MPC 単体での制御結果より優れた性能を発揮したと報告している。このような手法は、多点局所探索法 [53,54] と呼ばれ、目的関数（最適制御問題においては評価関数に相当する。）が多峰性を有する場合や、悪スケールな場合などに、有効な手法として知られている。一方で、これらの手法を適用するには、目的関数に関する先見情報（最適制御問題においては、制御対象の数値モデルや評価関数についての偏導関数に相当する。）が必須になるので、それらがブラックボックス化されたような問題には適用できないなど、サンプル型 MPC の利点と相反するという問題点が存在する。また、本論文 3 章及び 4 章で提案した手法も、十分時間をかけることが許されるのであれば強力な解決策となり得るが、そのような要求は、MPC に求められる実時間性の要求と相反するという課題があった。

そこで、本章では、決定論的最適化手法の 1 つであるニュートン法の幾何学的解釈に着想を得た、2 次超曲面回帰に基づくサンプル型のニュートン型最適化法を提案し、これらの問題を解決する。提案法は、2 次超曲面回帰を用いて、勾配及びヘシアンを近似する手法であるので、先見情報としてそれらを保持しておく必要がない。そのため、ブラックボックス化された最適制御問題にも適用可能であるなど、取り扱い可能な最適化問題のクラスについては、従来のサンプル型 MPC のものと同様であるといった利点がある。また、Fig. 5.1 に示すように、提案法は、ある 1 点近傍の勾配上を利用する従来の

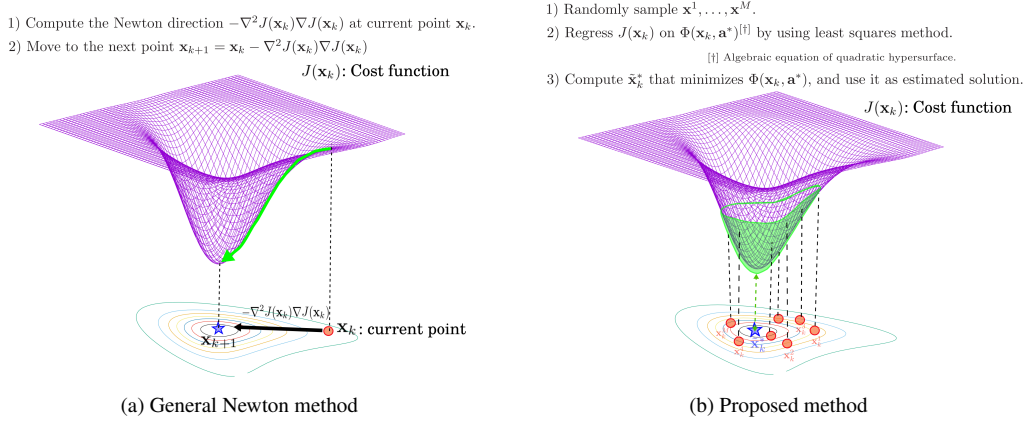


Fig. 5.1: Illustration of difference between the geometric interpretation of general Newton's method and proposed method.

ニュートン法 [80] と比べ、より大域的な情報を用いた手法であるので、従来法と同程度以上の収束速度を期待することができる。

本章の構成は以下に示す通りである。5.2 節では、一般的なニュートン法の幾何学的解釈から出発し、本章で提案するサンプルベース・ニュートン型最適化法のアイデアの核心部分と実装方法について記述する。5.3 節では、サンプルベース・ニュートン型最適化法が最適化に失敗する事例について考察し、そのような事態における解決策の 1 つとして並列黄金分割探索法を提案する。5.4 節では、従来の勾配型 MPC では処理困難な例題における適用事例を示し、本提案手法の有効性について検証する。最後に、5.5 節において、本章のまとめを述べる。

## 5.2. Sample-based Newton-Like 法

ここでは、提案法である 2 次超曲面回帰を用いたニュートン型最適化手法のアルゴリズムについて詳述する。

はじめに、一般的なニュートン法の幾何学的解釈 (Fig. 5.1 (a)) について考える。目的関数  $F(\mathbf{z})$  を近似解  $\mathbf{z}^{(d)}$  のまわりでテーラー展開し、2 次の微小量まで考慮した 2 次近似関数は、

$$F(\mathbf{z}_k^{(d)} + \delta \mathbf{z}) \approx F(\mathbf{z}^{(d)}) + \left( \frac{\partial F(\mathbf{z}^{(d)})}{\partial \mathbf{z}} \right)^T \delta \mathbf{z} + \frac{1}{2} \delta \mathbf{z}^T \nabla^2 F(\mathbf{z}_k^{(d)}) \delta \mathbf{z}, \quad (5.1)$$

と書くことができる。ここで、勾配ベクトルとヘッセ行列を

$$\mathbf{g} := \frac{\partial F(\mathbf{z}^{(d)})}{\partial \mathbf{z}} \quad (5.2)$$

$$\mathbf{H} := \nabla^2 F(\mathbf{z}^{(d)}) \quad (5.3)$$

と書くことにする．このとき， $\mathbf{H}$  が正定ならば， $\mathbf{z}^{(d)}$  近傍において  $F(\cdot)$  は凸関数となるので，

$$\delta \mathbf{z} = -\mathbf{H}^{-1} \mathbf{g} \quad (5.4)$$

のように探索方向を定め，

$$\mathbf{z}^{(d+1)} = \mathbf{z}^{(d)} - \mathbf{H}^{-1} \mathbf{g} \quad (5.5)$$

とすれば， $\mathbf{z}^{(d+1)}$  は， $\mathbf{z}^{(d)}$  近傍における  $F(\mathbf{z})$  の停留点となる．(5.1) から (5.5) で示したように，一般的なニュートン法の探索方向（以降，ニュートンステップという．）は目的関数  $F(\mathbf{z})$  を 2 次近似し，目的関数が最小となる停留点を求める方法と言える．一般的な最適化においては，Fig. 5.1(a) に示すように，(5.1) から (5.5) の手順を反復することで真の最適解  $\mathbf{z}^*$  収束させることができる．ニュートン法は，反復により生成した点列  $\{\mathbf{z}^{(d)}\}$  が最適解  $\mathbf{z}^*$  に 2 次収束することが知られており，決定論的最適化手法の中でも，収束が速いアルゴリズムである．一方で，目的関数に関する先見情報として勾配 (5.2) やヘッセ行列 (5.3) を予め用意する必要がある，その要求が，サンプル型 MPC の性質と相反することになる．

そこで，本研究では，(2.2) の評価関数  $J$  を 2 次超曲面の代数方程式  $\Phi(\hat{\mathbf{u}}_k^l, \mathbf{a})$  で回帰し，近似的に勾配やヘッセ行列を導出することで，それらを先見情報として保持しておく必要のない，勾配レスな最適化手法を提案する．以降では，提案法の具体的な手順について記述する．

回帰に用いる代数方程式はサンプル  $\hat{\mathbf{u}}_k^l$  ごとに，

$$\begin{aligned} \Phi(\hat{\mathbf{u}}_k^l, \mathbf{a}) &= \sum_{i,j=1}^T \hat{u}_{\{k|i-1\}}^l \hat{H}_{i,j} \hat{u}_{\{k|j-1\}}^l + \sum_{i=1}^T \hat{g}_i \hat{u}_{\{k|i-1\}}^l + \hat{C} \\ &= \sum_{i=1}^p a_i \hat{\phi}_i(\hat{\mathbf{u}}_k^l) \end{aligned} \quad (5.6)$$

であり， $\mathbf{a} \in \mathbf{R}^p$  は， $mT(mT+1)/2$  番目までの要素が，回帰によるヘシアン  $\hat{\mathbf{H}}$  の上（または，下）三角部分の要素， $(mT(mT+1)/2)+1$  番目から  $(mT(mT+1)/2)+T$  番目の要素が回帰による勾配ベクトル  $\hat{\mathbf{g}}$  の全要素，最後の要素が定数項  $\hat{C}$  に対応する要素をもつベクトルとする．

提案法では，はじめに，最小二乗法を用いて，以下に示す残差平方和

$$S_E = \sum_{l=1}^{\hat{L}} E_l^2 \quad (5.7)$$

が最小となるように、代数方程式 (5.6) の係数ベクトル  $\mathbf{a}$  を決定する。ここで、

$$w_l = \exp\left(-\frac{J(\hat{\mathbf{u}}_k^l)}{\tilde{\eta}}\right) \quad (5.8)$$

$$E_l = J(\hat{\mathbf{u}}_k^l) - \Phi(\hat{\mathbf{u}}_k^l, \mathbf{a}) \quad (5.9)$$

である。ここで、 $\hat{L}$  については、回帰に利用するサンプルの数を表している。また、(5.6)、(5.8) 及び (5.9) 中の  $\hat{\mathbf{u}}_k^l$  は、MCMPC のアルゴリズムうち入力列生成 (2.3.1 項) と同様の手続きでランダムサンプリングした入力列であり、2.3.1 項の手順で生成したものを再利用することもできる。

以下では、2 次超曲面回帰を用いて、勾配ベクトルとヘシアンを近似する方法について説明する。式 (5.6) に示す通り、 $\Phi(\hat{\mathbf{u}}_k^l, \mathbf{a})$  は、決定すべき変数である係数  $\mathbf{a}$  についての線形結合の形で記述できるので、

$$\mathbf{a}^* = (X^T W X)^{-1} X^T W Y, \quad (5.10)$$

のような簡単な行列演算を用いて、最適解  $\mathbf{a}^*$  を求めることができる。ここで、

$$Y := \left[ J(\hat{\mathbf{u}}_k^0), \dots, J(\hat{\mathbf{u}}_k^{\hat{L}}) \right]^T \quad (5.11)$$

$$W := \text{diag}(w_0, \dots, w_{\hat{L}}) \quad (5.12)$$

$$X := \begin{bmatrix} \hat{\phi}_1(\hat{\mathbf{u}}_k^1) & \cdots & \hat{\phi}_p(\hat{\mathbf{u}}_k^1) \\ \vdots & \ddots & \vdots \\ \hat{\phi}_1(\hat{\mathbf{u}}_k^{\hat{L}}) & \cdots & \hat{\phi}_p(\hat{\mathbf{u}}_k^{\hat{L}}) \end{bmatrix} \quad (5.13)$$

である。 $\mathbf{a}^*$  が求まったあとは、 $\mathbf{a}$  の定義に従って、勾配ベクトル及びヘシアンのそれぞれの要素を抽出し、適切に並べることで、2 次超曲面回帰により近似した勾配ベクトル  $\hat{\mathbf{g}}$  とヘシアン  $\hat{\mathbf{H}}$  を得ることができ、

$$\mathbf{u}_k^* = \tilde{\mathbf{u}} - (2\hat{\mathbf{H}})^{-1} \hat{\mathbf{g}} \quad (5.14)$$

のような形で、生の評価値  $J(\mathbf{u})$  を、ランダムサンプリングした入力列  $\hat{\mathbf{u}}_k^1, \dots, \hat{\mathbf{u}}_k^{\hat{L}}$  とその入力列ごとの評価値  $J(\hat{\mathbf{u}}_k^1), \dots, J(\hat{\mathbf{u}}_k^{\hat{L}})$  を用いて回帰した 2 次超曲面である  $\Phi(\hat{\mathbf{u}}_k^l, \mathbf{a})$  の極値を求めることができる。以降では、(2.6) 及び (5.14) を用いて計算した推定値をそれぞれ、 $\mathbf{u}_k^{MCMPC}$  及び  $\mathbf{u}_k^{Newton}$  と記述する。

ここまで見てきたように、提案法は、その実装にあたって、評価関数に関する勾配やヘシアンを事前に解析的に求めておく必要がない“勾配レス（あるいは、勾配フリーという。）”な手法である。よって、1.2.1.2 で挙げた、収束特性改善手法に求められる要件

のうち、サンプル型 MPC をの特長に干渉しないという特長を満たしている。さらに、2 次超曲面回帰の精度が十分高く、 $\hat{\mathbf{H}}$  が正定であるならば、その最適化精度は、一般的なニュートン法と同程度であることが容易に想像できる。さらに、回帰に際して、一般的なニュートン法と比較して、より大域的な形状情報を用いた 2 次近似を行っているので、一般的なニュートン法と同程度以上の収束速度が期待できる。

一方で、次項で想定するようなケースに於いては、回帰の精度が不十分となり、提案法である Sample-based Newton-Like 法が期待した通り動作しないことがある。

### 5.3. 並列黄金分割探索法

ここでは、5.2 節で説明したサンプルベース・ニュートン型最適化法が最適化に失敗するケースについて考察する。また、そのようなケースに対する対処法として“並列黄金分割探索法”を提案する。

Fig. 5.2 は、2 次超曲面回帰の結果得たヘシアン  $\hat{\mathbf{H}}$  が正定でない、又は特異となる、以外の理由によりサンプルベース・ニュートン型最適化法が最適化に失敗するケースを図示したものである。Fig. 5.2(a) は、箱型制約として与えられる入力制約を (2.8) に示す飽和演算により処理した結果、サンプルベース・ニュートン型最適化法による推定値  $\mathbf{u}_k^{Newton}$  の評価値  $J(\mathbf{u}_k^{Newton})$  が、MCMPC の推定値  $\mathbf{u}_k^{MCMPC}$  の評価値より劣る（最大化の場合小さく、最小化の場合大きくなる。）ケースを図示している。また、Fig. 5.2(b) は、サンプルベースニュートン型最適化法の推定値である回帰 2 次超曲面の極が、バリア関数 (2.9) が大きな値をもつ領域又は制約違反領域にあることにより最適化に失敗するケースを図示したものである。いずれのケースにおいても、回帰に利用するサンプルの中に制約を逸脱しているものが含まれていないため、制約境界に関する情報が欠如していることが原因であると考えられる。以上をまとめると、Fig. 5.2 に示す通り、提案法であるサンプルベース・ニュートン型最適化法において、各種の制約を処理する際に何らかの手立てが必要になる。以降では、最急降下法や準ニュートン法などの決定論的最適化手法でよく用いられる直線探索法から着想を得た、上述した問題を解決するための方法について記述する。

Fig. 5.2 に示した 2 つのケースにおいても、近似ヘシアン  $\hat{\mathbf{H}}$  が正定で、かつ、回帰の精度が良い場合には、(5.14) 右辺第 2 項に適切なステップ幅を導入することで、降下方向上で評価値が最小となる点を発見できる可能性がある。また、この際  $\mathbf{u}_k^{MCMPC}$  と  $\mathbf{u}_k^{Newton}$  を結ぶ線分上だけでなく、他のランダムサンプリングされた入力列  $\hat{\mathbf{u}}_k^{1, \dots, \hat{L}}$  と  $\mathbf{u}_k^{Newton}$  を結ぶ

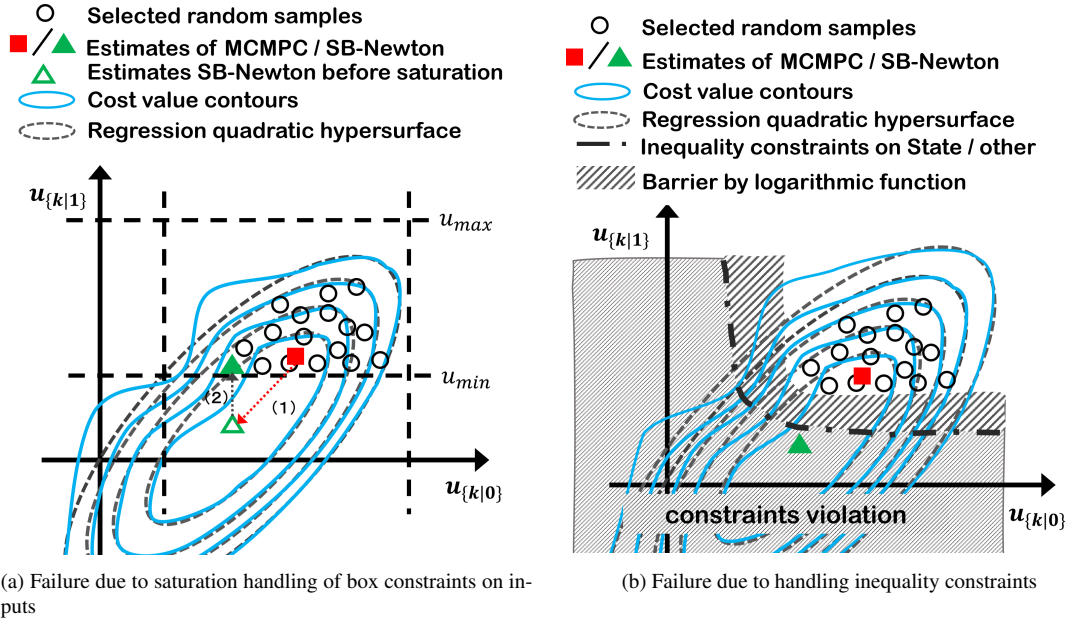


Fig. 5.2: Two cases where the Sample-based Newton-like method fails to optimize (expect when the Hessian is not positive definite or singular.)

線分上も探索することで限なく効率的に探索することができる。Alg. 2 は、この考え方に基づく並列黄金分割探索法のアルゴリズムの擬似コードである。Alg. 2 に示す通り、関数  $PGSS < \hat{L} >$  は、探索の結果、探索区間において  $\mathbf{u}^{MCMPC}$  の評価値より優れた評価値をもつ評価点があれば、それを  $\mathbf{u}^{Best}$  とし、なければ当該ブロックの最良値として  $\mathbf{u}^{MCMPC}$  を返す関数となっている。この処理は、探索区間上で、評価関数が単峰関数とならない場合などにも対応できる（一般的な黄金分割探索は、探索区間上で評価関数が単峰関数である仮定が前提となる。）ようにしたものである。これにより、少なくとも、元の  $\mathbf{u}^{MCMPC}$  より評価値の意味で劣る入力列が選択されないようにしている。また、Alg. 2 では、入力列ごとに block 単位で並列実装する擬似コードを示したが、block 内の処理も並列化できるので、すべての処理が GPU によって高速化できる。

## 5.4. 数値シミュレーション

本節の目的は、5.2 節～5.3 節で示した並列黄金分割探索付きサンプルベース・ニュートン型最適化を用いた提案並列 MPC の有効性を確認することである。5.3 節の考察から、特に Fig. 5.2 に列挙した各ケースにおける提案手法の有効性を検証する必要があると考える。そのため、[10] のベンチマーク問題の中から入力制約とその他の不等式制約の両方を有する問題も含めた 3 つを選び、数値シミュレーションの題材とした。以下では、3

---

**Algorithm 2** Parallel golden-section-search (PGSS) function
 

---

**Given:**  $\hat{L}$  : Number of samples used for regression.       $T$  : Number of prediction time steps.

$D_{PGSS}$  : Number of PGSS iterations.

block.id = 0, ...,  $\hat{L}$       ratio =  $(-1 + \sqrt{5})/2$

```

1: function PGSS( $\hat{L}$ )( $x_{\{k|0\}}$ ,  $\mathbf{u}_k^{Newton}$ ,  $\mathbf{u}_k^{MCMPC}$ ,  $\hat{\mathbf{u}}_k^{1, \dots, \hat{L}}$ ,  $J^{Newton}$ ,  $J^{MCMPC}$ )
2:   Initialize  $d_{PGSS} \leftarrow 0$ 
3:   Set  $\mathbf{u}_{0, \dots, \hat{L}}^{Newton / MCMPC} \leftarrow \mathbf{u}_k^{Newton / MCMPC}$  and  $J_{0, \dots, \hat{L}}^{Newton / MCMPC} \leftarrow J^{Newton / MCMPC}$ 
4:   Set  $\mathbf{u}_{0, \dots, \hat{L}}^{Best} \leftarrow \mathbf{u}_k^{MCMPC}$  and  $J_{0, \dots, \hat{L}}^{Best} \leftarrow J^{MCMPC}$ 
5:   while  $d_{PGSS} < D_{PGSS}$  do
6:     // Generate evaluating points in parallel using  $\hat{L}$  blocks on GPU.
7:     if block.id = 0 then
8:        $\hat{\mathbf{u}}_0^{Left} \leftarrow \text{ratio} * \mathbf{u}_k^{Newton} + (1 - \text{ratio}) * \mathbf{u}_k^{MCMPC}$ 
9:        $\hat{\mathbf{u}}_0^{Right} \leftarrow (1 - \text{ratio}) * \mathbf{u}_k^{Newton} + \text{ratio} * \mathbf{u}_k^{MCMPC}$ 
10:    else
11:       $\hat{\mathbf{u}}_{\text{block.id}}^{Left} \leftarrow \text{ratio} * \mathbf{u}_k^{Newton} + (1 - \text{ratio}) * \mathbf{u}_k^{\text{block.id}}$ 
12:       $\hat{\mathbf{u}}_{\text{block.id}}^{Right} \leftarrow (1 - \text{ratio}) * \mathbf{u}_k^{Newton} + \text{ratio} * \mathbf{u}_k^{\text{block.id}}$ 
13:    end if
14:    // Execute parallel simulation
15:    for  $\tau = 0$  to  $T$  do
16:       $x_{\text{block.id}, \{k|\tau+1\}}^{(Left/Right)} = x_{\text{block.id}, \{k|\tau\}}^{(Left/Right)} + f \left( x_{\text{block.id}, \{k|\tau\}}^{(Left/Right)}, u_{\text{block.id}, \{k|\tau\}}^{(Left/Right)} \right) \Delta t$ 
17:    end for
18:     $J_{\text{block.id}}^{(Left/Right)} \leftarrow \phi \left( x_{\text{block.id}, \{k|T\}}^{(Left/Right)} \right) + \sum_{\tau=0}^{T-1} S_C \left( x_{\text{block.id}, \{k|\tau\}}^{(Left/Right)}, \hat{u}_{\text{block.id}, \{k|\tau\}}^{(Left/Right)}, \tau \right)$ 
19:    if  $J^{Left} \leq J^{Right}$  then
20:       $\mathbf{u}_{\text{block.id}}^{MCMPC}, J_{\text{block.id}}^{MCMPC} \leftarrow \mathbf{u}_{\text{block.id}}^{Right}, J_{\text{block.id}}^{Right}$ 
21:      if  $J^{Left} \leq J^{MCMPC}$  then
22:         $\mathbf{u}_{\text{block.id}}^{Best}, J_{\text{block.id}}^{Best} \leftarrow \mathbf{u}_{\text{block.id}}^{Left}, J_{\text{block.id}}^{Left}$ 
23:      end if
24:    else
25:       $\mathbf{u}_{\text{block.id}}^{Newton}, J_{\text{block.id}}^{Newton} \leftarrow \mathbf{u}_{\text{block.id}}^{Left}, J_{\text{block.id}}^{Left}$ 
26:      if  $J^{Right} \leq J^{MCMPC}$  then
27:         $\mathbf{u}_{\text{block.id}}^{Best}, J_{\text{block.id}}^{Best} \leftarrow \mathbf{u}_{\text{block.id}}^{Right}, J_{\text{block.id}}^{Right}$ 
28:      end if
29:    end if
30:     $d_{PGSS} \leftarrow d_{PGSS} + 1$ 
31:  end while
32:  return  $(\mathbf{u}_{1, \dots, \hat{L}}^{Best}, J_{1, \dots, \hat{L}}^{Best})$ 
33: end function

```

---



つのベンチマーク問題で用いた制御対象の数値モデルと各ベンチマーク問題の問題設定について記述するとともに、それぞれに対する適用結果を示す。

#### 5.4.1 ベンチマーク問題の問題設定

1つ目の問題設定は、近年様々な分野で活躍しているクアッドコプタの位置姿勢制御の問題である。

##### Problem 5. クアッドコプタ

Fig. 5.3 は数値シミュレーションに使用したクアッドコプタのモデルを図示したものである。文献 [23] を参考に、姿勢表現には四元数を用い、高さ方向の推力  $v_z$  および各軸周りの回転方向の角速度  $v_{\omega_x}, v_{\omega_y}, v_{\omega_z}$  を制御入力とする 4 入力システム (5.15)~(5.20) としてモデル化した。

$$I_{xx}\dot{\omega}_x - (I_{yy} - I_{zz})\omega_y\omega_z - \frac{1}{2}T_{max}l \frac{u_{ccw1}|u_{ccw1}| - u_{ccw2}|u_{ccw2}|}{u_{max}^2} = 0 \quad (5.15)$$

$$I_{yy}\dot{\omega}_y + (I_{xx} - I_{zz})\omega_x\omega_z + \frac{1}{2}T_{max}l \frac{u_{cw1}|u_{cw1}| - u_{cw2}|u_{cw2}|}{u_{max}^2} = 0 \quad (5.16)$$

$$I_{zz}\dot{\omega}_z - (I_{xx} - I_{yy})\omega_x\omega_y + \tau_{max}u_r = 0 \quad (5.17)$$

$$m\ddot{x} - 2T_{max}(e_0e_2 + e_1e_3)u_r = 0 \quad (5.18)$$

$$m\ddot{y} + 2T_{max}(e_0e_1 - e_2e_3)u_r = 0 \quad (5.19)$$

$$m(g + \ddot{z}) - T_{max}(-1 + 2e_0^2 + 2e_3^2)u_r = 0 \quad (5.20)$$

ここで、式 (5.15)~(5.20) 中の  $u_r$  は、次式 (5.21) に示すとおりである。

$$u_r = \frac{u_{cw1}|u_{cw1}| + u_{cw2}|u_{cw2}|}{u_{max}^2} + \frac{u_{ccw1}|u_{ccw1}| + u_{ccw2}|u_{ccw2}|}{u_{max}^2} \quad (5.21)$$

また、ロータの回転数には、(5.22)~(5.25) の計算から得たものを使用した。

$$u_{cw1} = u_g + v_z - v_{\omega_y} + v_{\omega_z} \quad (5.22)$$

$$u_{cw2} = u_g + v_z + v_{\omega_y} + v_{\omega_z} \quad (5.23)$$

$$u_{ccw1} = u_g + v_z + v_{\omega_x} - v_{\omega_z} \quad (5.24)$$

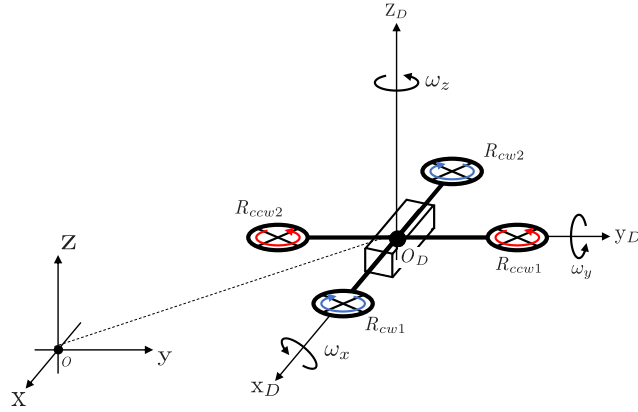


Fig. 5.3: Coordinate system for the quadcopter

$$u_{ccw2} = u_g + v_z - v\omega_x - v\omega_z \quad (5.25)$$

その他の式 (5.15)～(5.20) 中の定数パラメータは，Table 5.1 に示すとおりである．  
制御評価関数は，式 (5.26) に示す状態・入力に関する 2 次形式のものを用いた．

$$J = \sum_{\tau=0}^{T_d-1} \mathbf{x}_{\{k|\tau+1\}}^T \mathbf{Q} \mathbf{x}_{\{k|\tau+1\}} + \mathbf{u}_{\{k|\tau\}}^T \mathbf{R} \mathbf{u}_{\{k|\tau\}} \quad (5.26)$$

ここで，状態ベクトル  $\mathbf{x}$  は，目標位置と現在位置の誤差，並進速度，四元数  $[e_0, e_1, e_2, e_3]$  の虚部 3 変数  $e_1, e_2, e_3$ ，角速度の 12 次元のベクトル (5.27) である．

$$\mathbf{x} = [\Delta x, \Delta y, \Delta z, \dot{x}, \dot{y}, \dot{z}, e_1, e_2, e_3, \omega_x, \omega_y, \omega_z]^T \quad (5.27)$$

また，状態量および入力に関する重み行列は，

$$\begin{aligned} \mathbf{Q} &= \text{diag}(10, 10, 20, 1, 1, 2, 100, 100, 100, 10, 10, 10) \\ \mathbf{R} &= \text{diag}(0.1, 0.1, 0.1, 0.1) \end{aligned} \quad (5.28)$$

とした．予測ホライズン長は，0.98 s とし，予測ステップ数は， $T_d = 14$  とした．制御周期は，50 Hz である．

制御問題としては，飛行中に何らかの物体と衝突した等の不測事態を想定した姿勢回復制御を扱った．

#### Problem 6. 厳しい制約が課された台車型倒立振子の振り上げ安定化制御

2 つ目の問題設定は，3.5 節で示した **Problem 2** と同様のモデルを用いた台車型倒立振子の振り上げ安定化制御が題材である．そのため数理モデルにおける物理パラメータ等の

Table 5.1. Parameters of quadcopter

$m$	Mass of quadcopter	1.0 kg
$I_{xx}$	Moment of inertia around $x_D$ axis	$0.0085 \text{ kg} \cdot \text{m}^2$
$I_{yy}$	Moment of inertia around $y_D$ axis	$0.008 \text{ kg} \cdot \text{m}^2$
$I_{zz}$	Moment of inertia around $z_D$ axis	$0.0165 \text{ kg} \cdot \text{m}^2$
$u_{max}$	Maximum rotor speed	230.0 rps
$\tau_{max}$	Maximum anti-torque	$0.25 \text{ N} \cdot \text{m}$
$u_g$	Rotor speed when gravity and thrust balance	175.0 rps

設定値は **Problem 2** と同様のものを採用した．ただし，予測ホライズン長を  $0.6 \text{ s}$  とし，予測ステップ数を (30 steps) としている．本例題最大の特徴は，台車位置の制約として，

$$|z| \leq 0.05, \quad |f_c| \leq 0.8$$

というより厳しい制約が課せれている点である．これにより，最適化の際にヘシアンを使用する勾配型 MPC では処理困難な問題設定となっている．

3 つ目の問題設定を以下に示す．

### Problem 7. 2D-Crane

本例題の問題設定は，[10] を参考にした．Fig. 5.4 に示す 2 D-Crane の状態ベクトルは  $\mathbf{x} = [S_C, \dot{S}_C, S_{RL}, \dot{S}_{RL}, \Phi, \dot{\Phi}]$  であり，動的モデルは以下に示す通りである．

$$\ddot{S}_C = a_C \tag{5.29a}$$

$$\ddot{S}_{RL} = a_{RL} \tag{5.29b}$$

$$\ddot{\Phi} = -\frac{1}{S_{RL}} \left( g \sin(\Phi) + a_C \cos(\Phi) + 2S_{RL} \dot{\Phi} \right) \tag{5.29c}$$

ここで， $g$  は重力加速度， $S_C$  は台車位置， $S_{RL}$  はロープ長， $\Phi$  は鉛直下向きを原点としたときのロープの角度を表している．評価関数は，(2.12) と同様，状態・入力について 2 次形式のものを使用し，状態及び入力に関する重み行列は

$$Q = \text{diag}(1.0, 2.0, 2.0, 1.0, 1.0, 4.0), R = \text{diag}(0.05, 0.05),$$

とした．制約条件として，入力制約  $|a_C| \leq 2.0$ ， $|a_{RL}| \leq 2.0$ ，状態制約  $|\Phi| \leq 0.3$  及び (5.30) に示す非線形不等式制約が課されている難易度の高い問題設定となっている．

$$h(\mathbf{x}) = S_{RL} \cos(\Phi) - 0.2 (S_C + S_{RL} \sin(\Phi))^2 - 1.25 \leq 0 \tag{5.30}$$

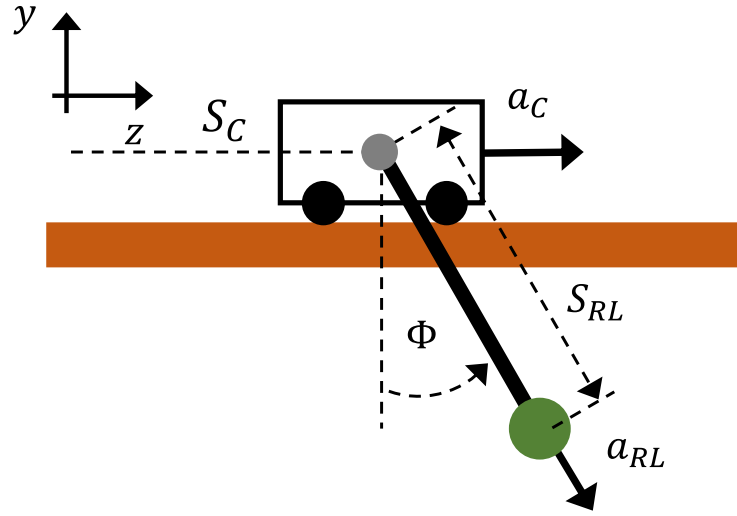


Fig. 5.4: Schematic representation of 2D-Crane system (**Problem 7**).

Table 5.2. Design parameters for proposed algorithm

Name		Problem 5	Problem 6	Problem 7
Prediction horizon		0.98s	0.6 s	1.2 s
Prediction steps		14	30	15
Dimensions of search space		56	30	30
Control cycle		50 Hz		
Num. of random samples		9000		
Num. of elite samples	MCMPC	200	150	100
	quadratic regression	2000	950	450
Variance	MCMPC	1.56	0.64	0.45
	SB-Newton method	0.06		0.36
Number of PGSS iteration		10	4	8

ここで,  $h(\mathbf{x})$  は, ロープ先端の安全動作範囲に関する制約である. 制御の目的は, 初期状態:

$$\mathbf{x}_{\{0|0\}} = [-2.0, 0.0, 2.0, 0.0, 0.0, 0.0], \quad (5.31)$$

から終端状態:

$$\mathbf{x}_{\{k|\tau\}}^{obj} = [2.0, 0.0, 2.0, 0.0, 0.0, 0.0], \quad (5.32)$$

へとクレーンの台車及びロープの先端を移動させることである.

**Problem 5**~**Problem 7** に対する提案法の設計パラメータは, Table 5.2 に示す通りである.

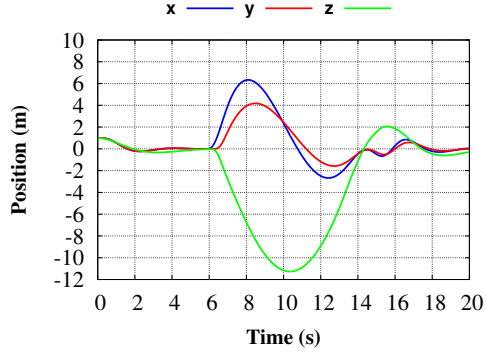
### 5.4.2 Problem 5 への適用結果

本数値シミュレーションでは、 $t = 6.0$  s に瞬発的な外力により速度  $\dot{x}$  や角速度  $\omega_y$  に対し、急激な変化 ( $\Delta \dot{x} = 1.0$  および  $\Delta \omega_y = 9.0$  or  $20.0$ ) を加えることで不測事態を再現し、そこからの姿勢回復制御を行った。

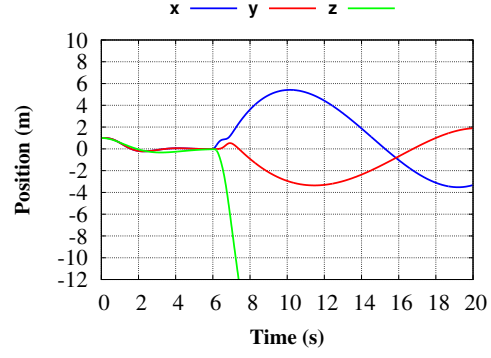
Fig. 5.5～Fig.5.7 は、それぞれ位置・姿勢・制御入力の時応答をプロットしたものである。外乱として  $\omega_y = 9.0$  を加えたときの位置の変化 (Fig. 5.5(a), Fig. 5.5(c) 及び Fig. 5.5(e)) 及び姿勢の変化 (Fig. 5.6(a), Fig. 5.6(c) 及び Fig. 5.6(e)) では提案法を用いたものが、最も速く、正確に目標位置・姿勢に回復することができている。次に良い結果を示したのは、MCMPC だけを用いたものであるが、14 s 以降にも  $y$  方向の位置制御の結果に偏差が残り続けている。提案法と MCMPC のみを用いた場合の結果において 6.0 s～14.0 s の制御の結果に大きな違いが見られないのは、5.3 節で示した黄金分割探索を行った結果、MCMPC に近い解が選択されているためであると考えられる。

外乱として  $\omega_y = 20.0$  を加えたときの位置の変化 (Fig. 5.5(b), Fig. 5.5(d) 及び Fig. 5.5(f)) 及び姿勢の変化 (Fig. 5.6(b), Fig. 5.6(d) 及び Fig. 5.6(f)) の結果では、提案法の性能の良さがより際立って確認できる。特に、この条件下では、実時間性を考慮した範囲で十分反復を行った勾配型 MPC の一種である ParNMPC を用いた制御結果では、位置・姿勢（特に、 $z$  軸方向の位置制御が破綻してしまっている。）を回復できていないことに注意する必要がある。これは、MCMPC などのサンプル型 MPC の特長の 1 つとして考えられている、レジリエンス制御 [26] への適用可能性を表していると考えられる。ParNMPC において、姿勢回復に失敗しているのは、急激な状態量の変化により、解くべき最適制御問題の初期値が前時刻のものから大きく変化したことで、前時刻の解を初期値としてニュートン法などの反復法を繰り返す勾配型 MPC のポリシーではこの変化に対応しきれなかったためであると考えられる。

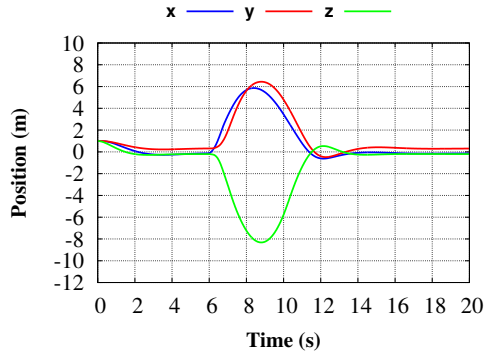
以上の結果から、提案法は、サンプル型 MPC の欠点である（特に、状態量が整定する直前において顕著である）最適化精度が勾配型 MPC に劣るという問題を克服し、目標値追従性能では勾配型 MPC と同程度以上の性能を発揮できる手法であることを確認した。また、勾配型 MPC との制御の結果の比較から、レジリエンス制御という観点において、サンプル型 MPC の利点を持ち合わせている制御手法であることも確認できた。



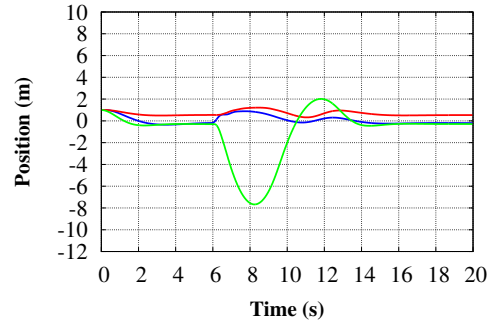
(a) ParNMPC (Add  $\Delta\omega_y = 9.0$ )



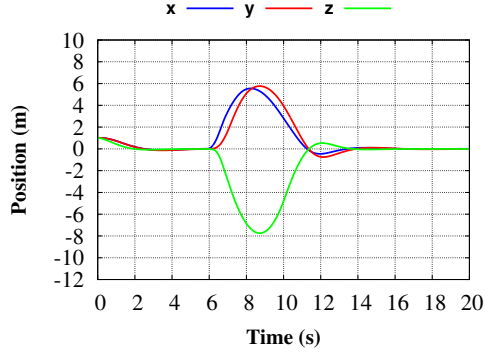
(b) ParNMPC (Add  $\Delta\omega_y = 20.0$ )



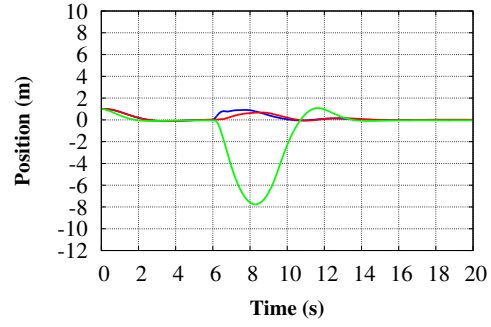
(c) MCMPC (Add  $\Delta\omega_y = 9.0$ )



(d) MCMPC (Add  $\Delta\omega_y = 20.0$ )



(e) Proposed method (Add  $\Delta\omega_y = 9.0$ )



(f) Proposed method (Add  $\Delta\omega_y = 20.0$ )

Fig. 5.5: Time response of positional parameters in **Problem 5**

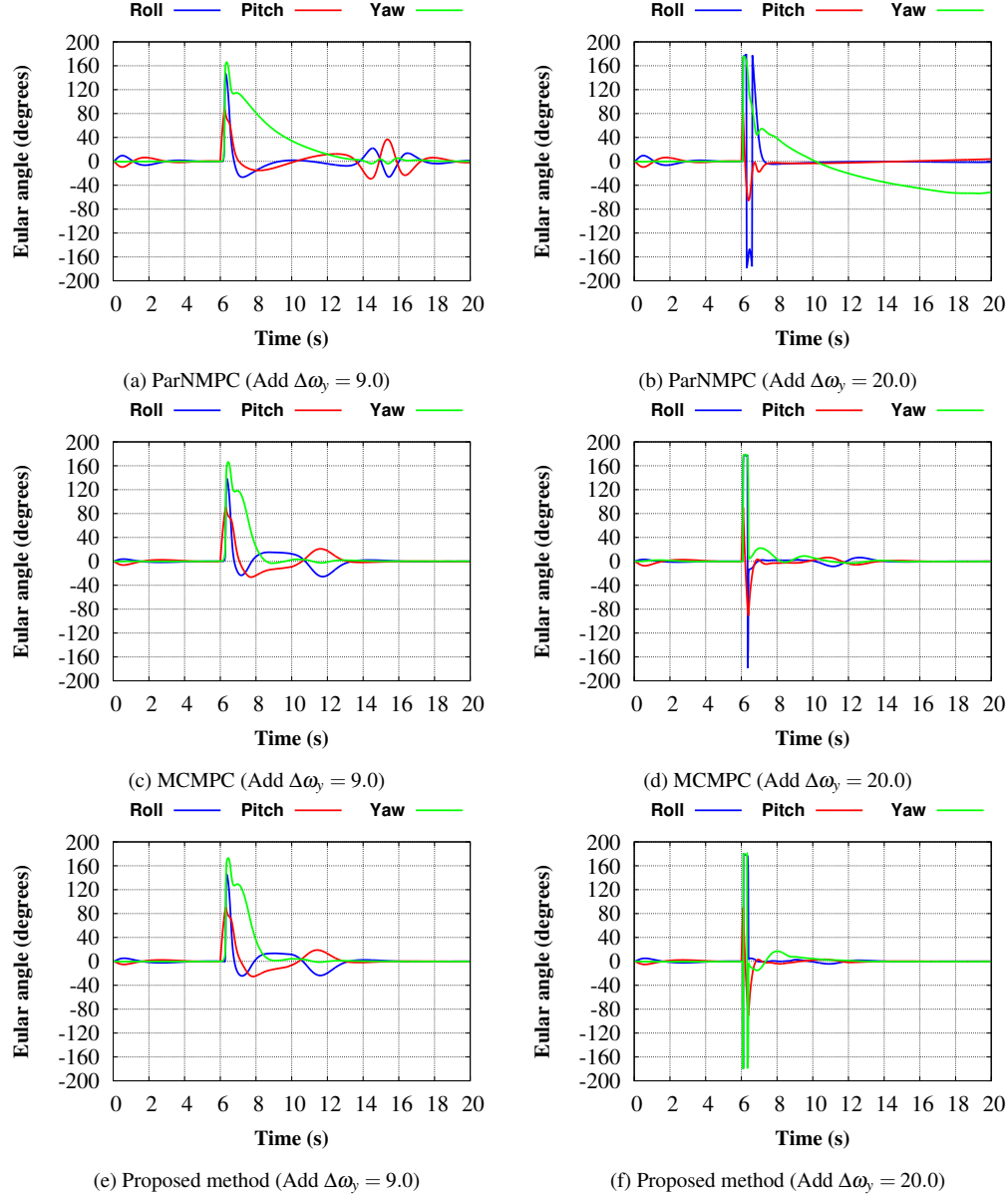


Fig. 5.6: Time response of attitude parameters in **Problem 5**

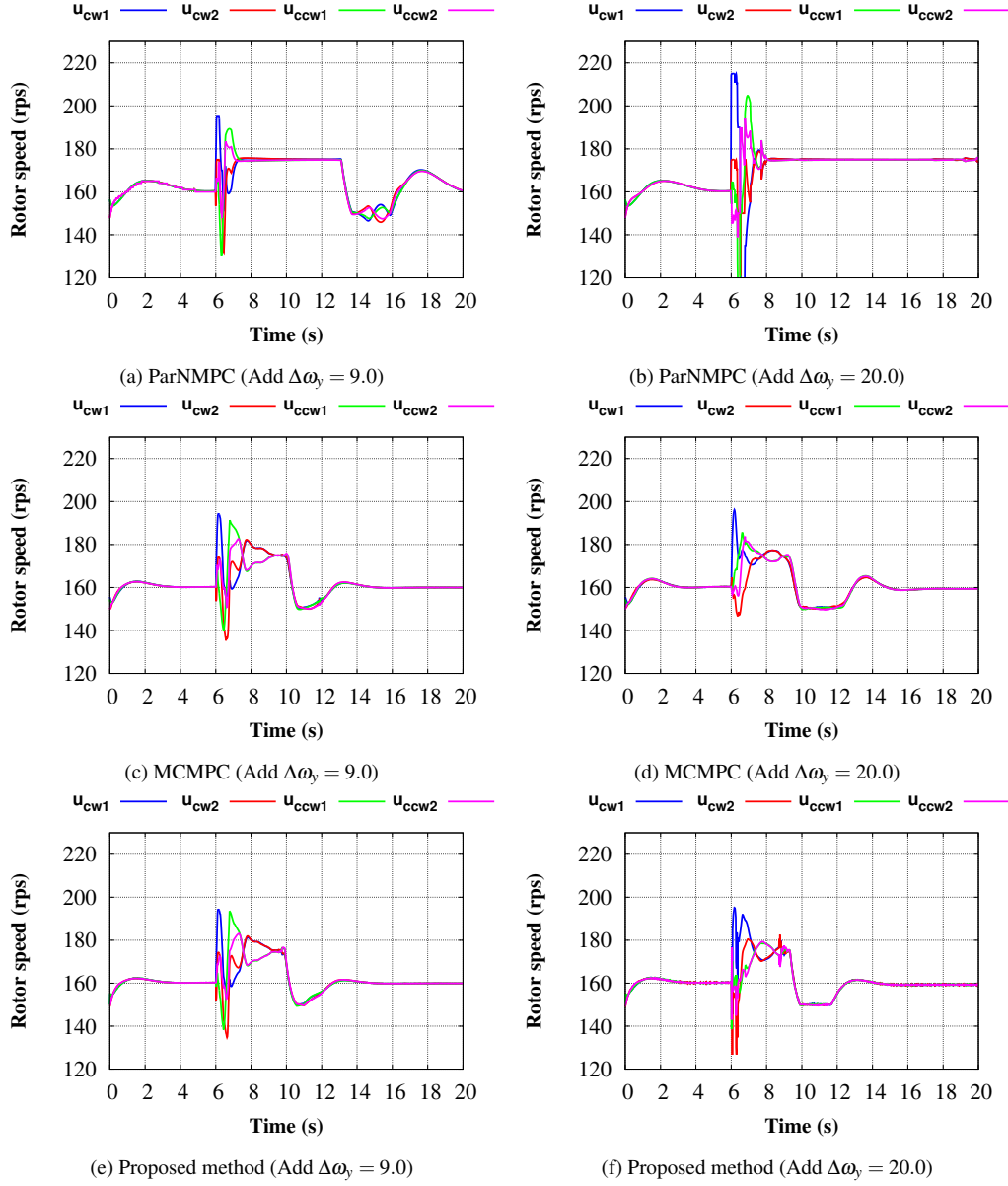


Fig. 5.7: Time response of control input (rotor speed) in **Problem 5**



### 5.4.3 Problem 6 への適用結果

Fig. 5.8 及び Fig.5.9 は, **Problem 6** に勾配型 MPC である GRAMPC, サンプル型 MPC である MCMPC 及び提案手法を適用した際の制御入力, 台車位置, 振子角度の時間応答を示している. またそれぞれ図の最下部に位置するプロットは, サンプルベース・ニュートン型最適化法が最適化に失敗した要因を色分けしてプロットしたものである. 白色の場合は, サンプルベース・ニュートン型最適化法による最適化が成功した例を示している.

勾配型 MPC には, GRAMPC の他に ParNMPC, ACADO, C/GMRES などもあるが, 本例題において, 制約違反は無視してタスクを完了できたのは GRAMPC だけであった. これは, GRAMPC 以外のソルバーがニュートン型の最適化法を利用しており, 本例題のような厳しい制約が課された最適化の際には, ヘシアンが正定でない, 又は, 悪条件となることが原因であると考えられる. 実際に, ParNMPC では, “ヘシアンが悪条件” であるという旨の警告が出て, それ以降の最適化に失敗し, 制御が破綻した. また, Fig. 5.9(a) に示す通り, GRAMPC ですら, 制約を満たす解を見つけることが出来ず, 当初設定した台車位置の制約に違反していることが分かる. これについては, 実時間性を無視して 1,000 ~ 10,000 回程度反復を行なっても同様の結果となった.

Fig. 5.8 において MCMPC のみを用いた場合 (2 段目のプロット) と提案法を用いた場合の入力波形を比べると明らかなように, サンプル型 MPC の欠点である入力波形の振動的現象が抑制されていることが分かる. また, この効果として, Fig. 5.9(a) に示すように, 提案法による台車位置の制御結果は, 勾配型 MPC である GRAMPC と同様に, 目標値に追従しているのに対して, MCMPC のみの場合は, 偏差が残っているという違いが生まれた. これは, 最適化の精度由来の入力の振動的な現象を抑制できたことで, 制御性能 (特に, 目標値追従特性) が向上したためである.

以上の結果から分かるように, 提案法は, 厳しい制約が課されている非線形性の強い問題であっても容易に処理できるというサンプル型 MPC の特長と, 勾配型 MPC が処理できる問題においては勾配型 MPC と同程度の最適化精度を発揮できるという特長を併せ持った手法であると言える.

### 5.4.4 Problem 7 への適用結果

Fig. 5.10 及び Fig. 5.11 は, **Problem 7** に, GRAMPC (Fig 5.11 においては青線), MCMPC (同じく, 赤線), 提案手法 (同じく, 緑線) を適用した際の制御入力, 台車位置, ロープ

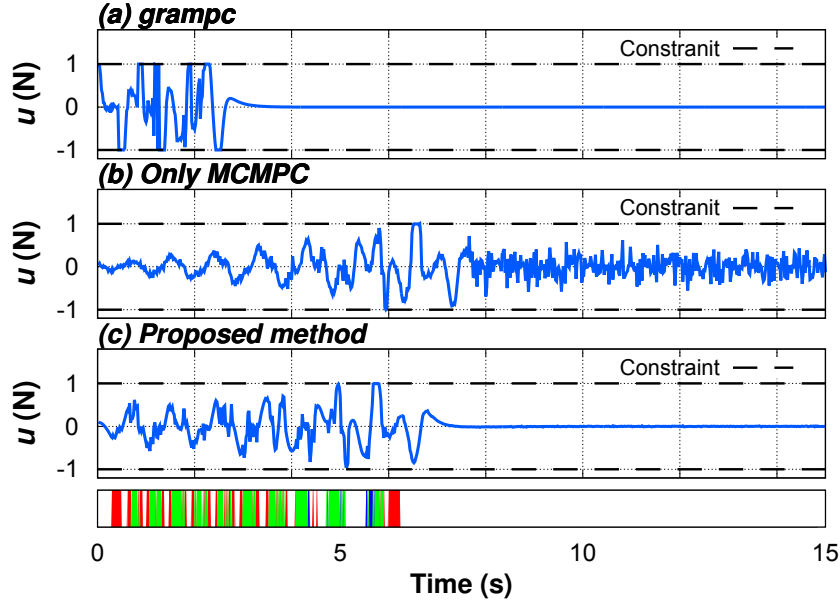


Fig. 5.8: Time response of control input in **Problem 6**. The bottom color bar shows failure case of SB-Newton (blue: Input constraint, red: State constraint, green: both cases, white: no failure).

長，ロープの角速度及び非線形不等式制約式の時間応答をプロットしたものである．

先ほどと同様，入力制約以外の制約であるロープの角速度（Fig. 5.11(c））及び非線形不等式制約  $h(x)$  の時間応答において，GRAMPC で制御を行った結果が制約違反を犯している．一方，MCMPC 及び提案手法では，そのような制約違反は発生していない．

また，台車位置 (Fig 5.11(a)) 及びロープ長 (Fig. 5.11) の制御結果における目標値までの到達順序は，台車位置が MCMPC，提案法，GRAMPC，ロープ長が提案法，GRAMPC，MCMPC であり，いずれにおいても，速応性の観点で提案法が GRAMPC より優れた結果であった．また，目標値追従特性で見ると，GRAMPC 及び提案法の両者が目標値に追

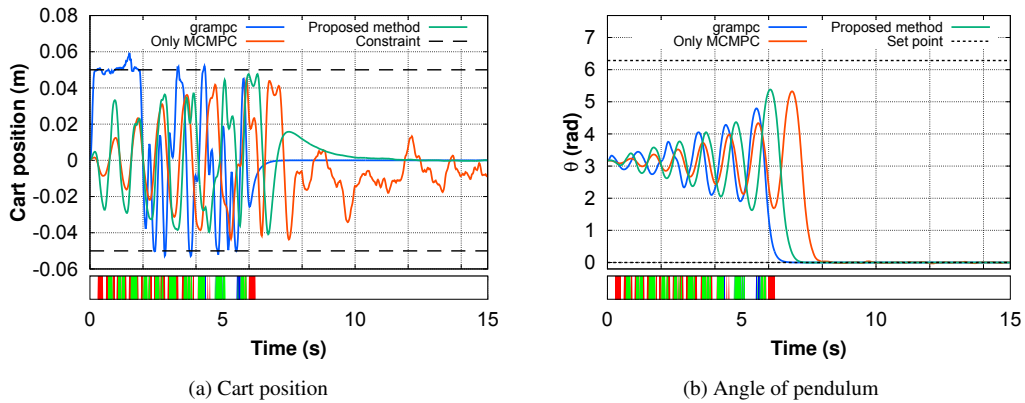


Fig. 5.9: MPC trajectories in **Problem 6**. The bottom color bar shows failure case of SB-Newton (blue: Input constraint, red: State constraint, green: both cases, white: no failure). the lines are colored with blue (GRAMPC), red (MCMPC), and green (proposed method).

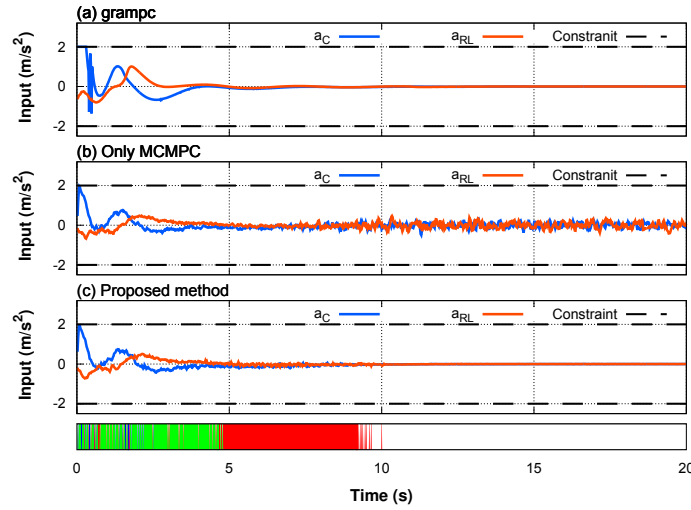


Fig. 5.10: Time response of control inputs;  $a_c$ (blue line),  $a_{RL}$ (red line) in **Problem 7**.

従しているのに対して，MCMPC の結果は目標値近傍で振動的であり，偏差が残る結果となっている．これは，Fig. 5.10 の MCMPC の入力波形における，時刻 9.0 s 以降の振動的な現象が影響しているためであると考えることができる．

以上をまとめると，**Problem 7** においても，提案法が勾配型 MPC の良い性質とサンプル型 MPC の良い性質の両面を併せ持った手法であることが分かる．

#### 5.4.5 Problem 6 及び Problem 7 におけるサンプルベース・ニュートン型最適化法が最適化に失敗した要因分析

Table 5.3 は，**Problem 6** 及び **Problem 7** 双方の問題へ提案法を適用した際に

$$J^{MCMPC} \leq J^{Newton}$$

となり，並列黄金分割探索法を適用する前段階でサンプルベース・ニュートン型最適化法が最適化に失敗した要因をまとめたものである．Table 5.3 が示すように，いずれの問題への適用結果においても，Fig. 5.2 で考察した，入力制約違反，状態制約違反，入力及び状態制約を同時に違反したケースが全体の 90% 以上を占めていることが分かる．これは，並列黄金分割法の適用について述べた 5.3 節での考察が概ね正しかったことを示唆している．

### 5.5. 5章のまとめ

本章では，確率・統計的最適化手法により最適制御問題を解くサンプル型 MPC における，そのアルゴリズム由来のランダム性の影響から最適化の精度が悪く，実際にシ

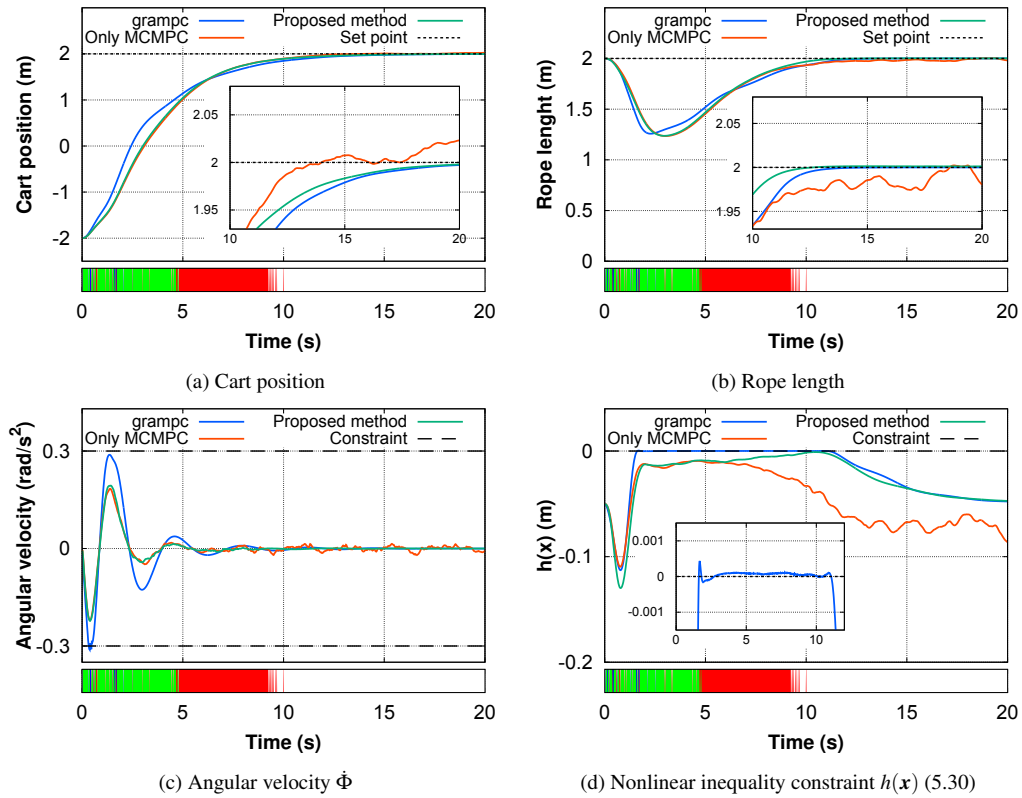


Fig. 5.11: Time response in **Problem 7**: (a) cart position, (b) rope length, (c) angular velocity, (d) inequality constraints.

Table 5.3. Frequency of failure of SB-Newton

<i>Factor</i>	<i>Problem 6</i>	<i>problem 7</i>
Hessian $\hat{H} \leq 0$	5.4 % (12/221)	2.7 % (13/482)
Input const. (Fig. 5.2(a))	5.4 % (12/221)	5.4 % (39/482)
State const. (Fig. 5.2 (b))	36.7 % (81/221)	55.0 % (265/482)
Input & state const.	52.5 % (116/221)	36.9 % (178/482)

システムに加える制御入力が高ス雑音のような不規則な振動的波形となるといった欠点を克服する方法について検討した。

本章で提案したサンプルベース・ニュートン型最適化法は、ランダムサンプリングと2次超曲面回帰に基づき、ニュートン法などで用いられる勾配やヘシアンに相当するベクトルや行列を近似的に導出する手法であり、事前情報としての勾配情報が不要であるという“勾配レス”なアルゴリズムである。この特長は、サンプル型 MPC の特長である、勾配情報不要であるという点で一致しており、サンプル型 MPC の一種と考えることができる。一方、最適化におけるアイデアの中心は、2次収束が期待できる勾配法であるニュートン法の幾何学的解釈に基づくものであり、2次超曲面回帰の精度が十分高けれ

ば，ニュートン法と同程度以上の収束速度と解の精度が期待できる手法である．

数値シミュレーションでは，勾配型 MPC が処理困難な例題である，サンプル型 MPC の特長であるレジリエンス性が求められるクアッドコプタの姿勢回復制御 (**Problem 5**)，厳しい制約が課された非線形性の強い最適問題の処理が求められる台車型倒立振子の振り上げ安定化制御 (**Problem 6**)，状態・入力に関する箱型制約以外の非線形不等式制約が課されている 2 D-Crane の制御 (**Problem 7**) の問題において，提案法の有効性について検証した．いずれの問題においても，提案法が，勾配型 MPC が処理困難である問題でも処理できるというサンプル型 MPC の利点を有しており，さらに，サンプル型 MPC の欠点であった入力がガウス雑音のように小刻みに振動する現象を抑制できていることを確認した．これらの結果は，提案法が勾配型 MPC とサンプル型 MPC の両者の長所を併せ持った手法であることを示唆している．

以上の結果より，本提案法は，当初研究課題として挙げた，制御性能を向上して，勾配型 MPC が処理可能な問題では，勾配型 MPC と同程度の性能を発揮し，勾配型 MPC が処理困難な問題へも適用できる，より実用的な MPC の制御アルゴリズムであると言える．

## 第6章 不連続現象を含む非線形システムへの適用

### 6.1. はじめに

本章は、1章において、研究課題2として示した適用可能性のうち、衝突のような不連続現象を含む制御対象への適用可能性に関する検証結果をまとめたものである。確率・統計的最適化手法を用いたサンプル型 MPC には、従来型の決定論的最適化手法を用いた勾配型 MPC と比較して、より広範なクラス最適化問題に適用できる可能性があることについては、1.2.2 で述べた通りである。Table 1.1 で比較した内容に基づき考えると、システムのモデルや評価関数において多峰性や不連続性、微分不可性、厳しい制約が課されるなど強い非線形性により勾配やヘッセ行列が発散するまたは悪条件となるような勾配型 MPC が対処困難な問題への適用可能性について議論する必要があると言える。5章では簡単な数値例で、厳しい制約が課されるなどの強い非線形性を有する最適化問題において、ヘッセ行列が悪条件となるなど、勾配型 MPC が処理困難とする問題におけるサンプル型 MPC の有効性について議論した。一方、多峰性、不連続性や微分不可性を伴う問題への適用については、確率・統計的最適化手法と決定論的最適化の適用可能範囲を分ける分水嶺とも言える重要な性質であるにも関わらず、著者らが調査した範囲でそういった例題への適用や比較検証は行なわれていない。

そこで、本章では、不連続な現象を伴う制御系への適用とその有効性について検証を行う。本章で題材としたのは、台車型倒立振子の振り上げ安定化制御において、台車が走路上に存在する壁などの障害物に衝突した際に発生する撃力由来のエネルギーを活用して振り上げ高速化を実現する制御問題である。当該問題設定は、衝突の前後でシステムの予測モデルに状態ジャンプが生じるので、一般的な勾配型 MPC では、処理困難な問題とされている。例えば、文献 [81] では、不連続現象を考慮した最適制御問題の定式化および勾配法によるその解法が示されているが、本章の問題設定のように、不連続現象を積極的に活用する制御則を導出すると、事前に候補となる不連続現象の発生タイミングを計画しておく必要がある。上記に関する詳細は、不連続現象（状態ジャン

プ)を含む最適制御問題の勾配型 MPC による数値解法とその問題点として、6.2 節で述べる。いずれにしても、本章で扱う課題は、従来の勾配型 MPC では困難な問題設定であり、MCMPC の不連続性を伴う非線形系への適用可能性を議論するのに十分有用な課題設定である。

本章の構成は以下に示す通りである。はじめに、6.2 節にて、従来型の勾配型 MPC を用いた MPC のにおける最適制御問題の一般的な解法と、本章の例題のように不連続現象を含む制御対象への適用時における問題点について関連研究の動向を含め記述する。6.3 節において、台車型倒立振子の不連続性を振り上げ安定化制御の問題において、台車が壁と衝突した際に発生する撃力を考慮した予測モデルの構築について述べる。次いで、6.4 節で、具体的な検証内容及び評価要領について記述する。検証については、6.5 節の数値シミュレーションだけでなく 6.6 節で示す実機実験も行った。最後に、6.7 節で本章の内容についてまとめる。

## 6.2. MPC における最適制御問題の解法と本章例題との関係

ここでは、状態ジャンプを伴う制御対象に対する MPC の最適制御問題の定式化及びその解法について記述する。ここで、状態ジャンプとは、本章で扱う衝突現象のように、その現象の前後において  $x_{\{k|i-\}} \neq x_{\{k|i+\}}$  のようにホライズン中のあるステップ  $i$  において、状態が不連続に変化する現象のことである。一般的な MPC [9,11] では、実際に最適制御問題を解く際には、それを適当なステップ数にホライズンを分割し離散近似したものを考えるため、ここでも、(2.2) 同様、離散時間システムの場合における解法について記述する。以下は、予測ホライズン長  $T$  を  $N$  ステップに分割した場合の最適制御問題である。

$$x_{\{k|0\}} = x_k \quad (6.1)$$

$$x_{\{k|i+1\}} = x_{\{k|i\}} + f(x_{\{k|i\}}, u_{\{k|i\}}) \Delta \tau := \bar{f}(x_{\{k|i\}}, u_{\{k|i\}}) \quad (6.2)$$

$$J = \phi(x_{\{k|N\}}) + \sum_{i=0}^{N-1} S(x_{\{k|i\}}, u_{\{k|i\}}) \quad (6.3)$$

ここで、 $x_{\{k|\cdot\}} \in \mathbb{R}^n$  は状態ベクトル、 $u_{\{k|\cdot\}} \in \mathbb{R}^m$  は入力ベクトルであり、(6.1) は最適制御問題を解く際の状態に関する初期条件、(6.2) は、状態方程式であり、本最適化問題に課される等式拘束条件である。このとき、最適制御問題は、(6.3) が最小となるように有限個の変数である  $u_{\{k|0\}}, \dots, u_{\{k|N-1\}}$  を決定する等式拘束条件付きの最適化問題となる。等式拘束条件 (6.2) を、

$$\bar{f}(x_{\{k|i\}}, u_{\{k|i\}}) - x_{\{k|i+1\}} = 0$$

のように変形し、ラグランジュの未定乗数として  $\lambda_{\{k|i+1\}} \in \mathbb{R}^n$  を導入すると、次に示すラグランジュ関数が得られる。

$$\bar{J} = \phi(x_{\{k|N\}}) + \sum_{i=0}^{N-1} \{S(x_{\{k|i\}}, u_{\{k|i\}}) + \lambda_{\{k|i+1\}}^T \bar{f}(x_{\{k|i\}}, u_{\{k|i\}}) - \lambda_{\{k|i+1\}}^T x_{\{k|i+1\}}\} \quad (6.4)$$

ここで、次に示すようなハミルトン関数 [9,11] と呼ばれるスカラー値関数

$$H(x_{\{k|i\}}, u_{\{k|i\}}, \lambda_{\{k|i+1\}}) = S(x_{\{k|i\}}, u_{\{k|i\}}) + \lambda_{\{k|i+1\}}^T \bar{f}(x_{\{k|i\}}, u_{\{k|i\}}) \quad (6.5)$$

を導入する。

このとき、ラグランジュ関数 (6.4) に対する停留条件、元の拘束条件 (6.2) などを併せてまとめると、次に示す最適制御が満たすべき必要条件を得ることができる [9].

**定理 4** ([9, 定理 3.1]). (離散時間のオイラー・ラグランジュ方程式) 離散時間システム (6.2) の初期状態が (6.1) で与えられているとする。制御入力時系列  $u_{\{k|0\}}, \dots, u_{\{k|N-1\}}$  が評価関数 (6.3) を最小にする最適制御入力であるとき、随伴変数時系列  $\lambda_{\{k|0\}}, \dots, \lambda_{\{k|N\}}$  が存在して、すべての  $i = 0, \dots, N-1$  において以下が成り立つ。

$$x_{\{k|i+1\}} = \bar{f}(x_{\{k|i\}}, u_{\{k|i\}}) \quad (6.6)$$

$$\lambda_{\{k|N\}} = \left( \frac{\partial \phi}{\partial x} \right)^T (x_{\{k|N\}}) \quad (6.7)$$

$$\lambda_{\{k|i\}} = \left( \frac{\partial H}{\partial x} \right)^T (x_{\{k|i\}}, u_{\{k|i\}}, \lambda_{\{k|i+1\}}) \quad (6.8)$$

$$\frac{\partial H}{\partial u}(x_{\{k|i\}}, u_{\{k|i\}}, \lambda_{\{k|i+1\}}) = 0 \quad (6.9)$$

定理の内容、特に、式 (6.9) より、ある条件のもとで、 $u_{\{k|i\}}$  は、 $x_{\{k|i\}}$ 、 $\lambda_{\{k|i+1\}}$  の関数として決まるという性質がある。また、 $x_{\{k|i\}}$  と  $\lambda_{\{k|i\}}$  についても、一方が決まればもう一方が自動的に決定するという性質がある。ただし、状態  $x_{\{k|i\}}$  は (6.1) のように初期条件が与えられるのに対し、随伴変数  $\lambda_{\{k|i\}}$  は (6.7) のように終端条件が課されている点に注意が必要である。上記のような初期時刻および終端時刻の両方で境界条件が与えられている方程式は 2 点境界値問題と呼ばれ、(6.6) や (6.8) が非線形な場合には、直接解くのが困難とされている。そこで、多くの勾配型 MPC では、式 (6.6)~(6.9) で示すオイラー・ラグランジュ方程式を満たすように解の修正を繰り返す、何らかの反復解法を用いてこれを解いている [9,11,82].

一方、式 (6.6)~(6.9) の構造から、一旦いずれかの未知関数が決定すると、その未知関数から他の未知関数が決まってしまうという性質が、本章で扱う例題のように、状態量



などに不連続現象が含まれる場合には弊害となることがある．例えば，最初に決定する未知関数を  $x_{\{k|i\}}$  とすると，本章の例題では，6.3 節で示すような衝突方程式などを用いれば，不連続現象（衝突や状態ジャンプ）を含む状態の時系列  $x_{\{k|0\}}, \dots, x_{\{k|N\}}$  を順時間方向に一応は予測することができる．ただし，不連続現象が発生した時刻を  $i_c$  とすると時刻  $i_c$  においては，(6.5) や (6.6) が成立していないことに注意する必要がある．この時，時刻  $N$  から  $i_c + 1$  までの  $\lambda_{\{k|i_c+1\}}, \dots, \lambda_{\{k|N\}}$  は  $x_{\{k|i\}}$  の関数として，上の段落述べた性質から (6.7) 及び (6.8) を用いて導出できるが，時刻  $i_c$  においては，(6.5)，(6.6) 及び (6.8) が成立しないのでこれ以降の時刻の  $\lambda_{\{k|i\}}$  を逆時間方向に正確に辿ることは出来ない．従来の勾配型 MPC では，このような問題は扱えないので，問題設定そのものを修正する方法が考えられている．

文献 [81,83] では，上述した問題に対し，不連続現象が発生した時刻の前後における未知関数  $x_{\{k|i\}}$ ， $\lambda_{\{k|i\}}$ ， $u_{\{k|i\}}$  に等式制約（内点拘束条件などという．）を課すことで，評価関数を分割し，分割後のそれぞれの時間区間において式 (6.6)～(6.9) が満たされるように解の更新を行う方法が提案されている．また，Borrelli ら [84] は，状態量  $x_{\{k|i\}}$  を適当に分割した上で，分割後の各状態量を初期値として，上述した方法を用いてオフラインで最適解を計算しておき，実際の制御を行う際には，観測した状態量と最も近い状態量に対するオフラインで計算した最適解を初期値としてニュートン法などを用いる explicit MPC と呼ばれる方法を用いる手法を提案している．しかし，いずれの方法も，実際に何らかの問題に適用するとすると，制御目的に最も適合する不連続現象が生起するタイミングを何らかの方法で予めオフラインで計画しておく必要がある．また，上記で述べたような不連続現象が生起するタイミングを含めた最適化問題は，組合せ最適化問題を伴う NP 困難な問題となることが知られている [85,86]．

本章で扱う例題も本質的には，上述したような性質を有する最適制御問題であり，オンラインで不連続現象の発生タイミングまでを含んだ組合せ最適化問題を解く必要がある非常に難易度の高い課題設定となっている．前の段落で述べた通り，本質的には，NP 困難な問題であるとされるので，本研究では，不連続現象に関する事前知識の有無による制御結果の違いから MCMPC の不連続性を伴う非線形系への適用可能性および有効性について議論する．

### 6.3. 衝突を含んだ台車型倒立振子の予測モデル

以下では、台車型倒立振子の不連続性を振り上げ安定化制御の問題において、台車が壁と衝突した際に発生する撃力を考慮した予測モデルの構築と本章で扱う問題設定について記述する。

#### Problem 8. 衝突現象を有効活用した台車型倒立振子の振り上げ安定化制御

衝突現象を考慮しない一般的な台車型倒立振子の力学モデルは、3.5 節 **Problem 2** で示した通りである。ここでは、台車が走路上にある壁と衝突した際のエネルギーの変化に着目した衝突を含む台車型倒立振子の力学モデルについて記述する。衝突は、その前後で、物体の変形、熱の発生、内部エネルギーの変化などにより何らかのエネルギーの損失を伴う非段弾性衝突と、前後で力学的エネルギーが保存される完全弾性衝突に分類される。衝突によるエネルギーの変化は、物体の速度に関する項の急激な変化として記述できる。本例題で扱う台車型倒立振子の場合、以下に示す関係式を用いて衝突現象をモデル化することができる。

$$q = \begin{bmatrix} z \\ \theta \end{bmatrix} \quad (6.10)$$

$$\mathbf{M}\dot{q}_+ = \mathbf{M}\dot{q}_- + \mathbf{t}\Phi\lambda_c \quad (6.11)$$

$$\mathbf{M} = \begin{bmatrix} m_p + m_c & m_p l_p \cos \theta \\ m_p l_p \cos \theta & J_p + m_p l_p^2 \end{bmatrix} \quad (6.12)$$

ここで、 $z$  は台車位置、 $\theta$  は振子角度とし、一般化座標  $q$  の 1 回の時間微分を  $\dot{q}$ （一般化速度）、 $q$  の 2 回微分を  $\ddot{q}$  と定義している。また、 $\dot{q}_-$ 、 $\dot{q}_+$  の下添字  $-$  及び  $+$  はそれぞれ、衝突の前及び後の物理量であることを意味している。式 (6.11) は衝突方程式であり、 $\lambda_c$  は衝突時のエネルギー損失と関係のある撃力に相当する項である。

本例題のように、台車型倒立振子の振り上げ安定化において、台車のみが壁と衝突する状況では、台車の速度に関する自由度のみが拘束を受けることになる。そのことを考慮すると、運動量保存則より

$$-e\dot{z}_+ = \dot{z}_- + \mathbf{t}\Phi\lambda_c \quad (6.13)$$

$$\Phi = \begin{bmatrix} 1 & 0 \end{bmatrix} \quad (6.14)$$

となる。ここで、 $e$  は台車と壁の間の反発係数である。さらに撃力に相当する  $\lambda_c$  は、(6.10)～(6.13) の関係を用いて、

$$\lambda_c = -(1+e)(\Phi\mathbf{M}^t\Phi)^{-1}\Phi\dot{q}_- \quad (6.15)$$

と導出することができる。さらに、(6.11)、(6.14) 及び (6.15) の関係を用いて、衝突の瞬間振子に働く角力積による振子角速度の瞬間的な変化量は、衝突前後の台車速度及び振子角速度などを用いて、

$$\dot{\theta}_+ = \dot{\theta}_- + (1+e) \frac{m_p l_p \cos \theta}{J_p + m_p l_p^2} \dot{z}_- \quad (6.16)$$

のように導出することができる。

予測シミュレーションにおける衝突現象のモデル化では、以下に示す要領で台車速度や振子角速度をの急激な変化を再現する。

台車位置 衝突した側の壁との接触面の座標

振子角速度 (6.11) を用いて更新

台車速度  $\dot{z}_+ = -e\dot{z}_-$  のように更新

## 6.4. 検証要領

ここでは、サンプル型 MPC である MCMPC が不連続現象を効果的に扱うことができるかを検証するために、台車型倒立振子の振り上げ安定化制御において以下に示す4つの異なるシナリオを準備して、数値シミュレーション及び実機実験を行った。

### Scenario 1 (Fig. 6.1(a))

予測モデル 衝突に関する予測モデルなし

シナリオ 台車の走路上に壁が存在しない

### Scenario 2 (Fig. 6.1(b))

予測モデル 衝突に関する予測モデルあり

シナリオ 台車の走路上に壁が存在する

### Scenario 3 (Fig. 6.1(c))

予測モデル 衝突に関する予測モデルなし

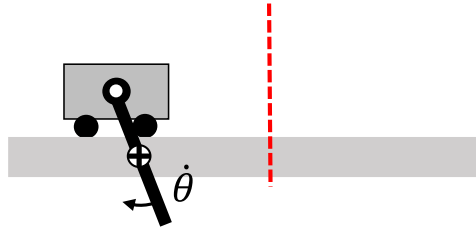
シナリオ 台車の走路上に壁が存在する

### Scenario 4 (Fig. 6.1(d))

予測モデル 壁の位置を既知とし、位置制約として処理する

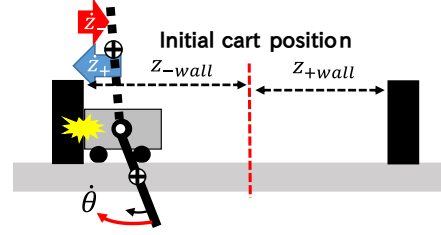
### Normal situation ( Without walls)

#### Initial cart position



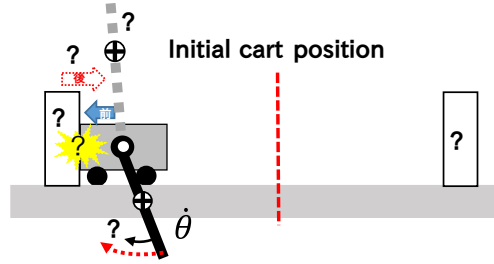
(a) Scenario 1

### Exploiting collision of cart for swing- up



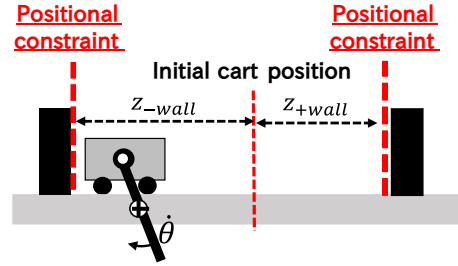
(b) Scenario 2

### Unexpected collision



(c) Scenario 3

### Collision as penalty



(d) Scenario 4

Fig. 6.1: Schematic diagram of the four scenarios used to verify the feasibility of control that actively utilizes a collision phenomenon for the pendulum swing.

### シナリオ 台車の走路上に壁が存在する

以降、具体的な検証要領について示す。Scenario 2 での制御の結果が、各制御時刻における評価値  $J$  が他の Scenario のものと比較した時に、常に最短時刻で 0 近傍に収束し、なおかつ、振り角度が倒立安定状態を示す  $0 \text{ rad}$  または  $\pm 2k\pi \text{ rad}$  に最短時刻で到達していれば、それは、衝突を有効に利用し、振り上げ高速化が実現できていることになる。Scenario 3 は、対照実験であり、Scenario 3 での制御の結果が、Scenario 2 のものと常に同じで最短となる場合には、Scenario 2 における衝突が、予測に基づく意図的なものではなく、偶然生起した衝突により振り上げの高速化ができていないのに他ならないことを示している。一方で、Scenario 3 での振り上げの結果、Scenario 2 の結果との比較において、劣るようなケースが大多数である場合、Scenario 2 は、意図的な衝突により高速化を実現出来ると言える。最後の Scenario 4 は、壁面近傍で急減速させる必要があるシナリオで、振り上げ高速化が撃力に由来するものか否かを検証するための対照実験となっている。

Table 6.1. Design parameters of MCMPC (for 8)

Name	Value
Prediction interval	1.0 s
Num. of prediction steps	100 steps
Control cycle	100 Hz
Num. of samples	5120
Num. of elite sample	50 (Elite sample MCMPC) 1 (TOP1 sample)
Num. of iterations	4

## 6.5. 数値シミュレーション

Table 6.1 は、本章における数値シミュレーション及び実機実験に用いた Elite sample 及び TOP1 sample の MCMPC のパラメータを表している。また、Fig. 6.2～Fig. 6.6 は、数値シミュレーションにおける台車位置、台車速度、振子角度、振子角速度及び各制御時刻における予測入力時系列  $\hat{\mathbf{u}}_k$  (MCMPC の推定値) を用いて計算した評価値  $J(\hat{\mathbf{u}}_k)$  の値を図示したものである。左右の壁は、それぞれ左側  $-0.14$  m、右側  $0.18$  m の位置に設置している (Fig. 6.2 の黒色の破線)。また、反発係数は、実機実験 (6.6) で用いた実験機材で計測した値  $e = 0.35$  としている。

Fig. 6.2 に示す Elite sample 及び TOP1 sample の双方の結果において、Scenario 1 (黒色実線)、Scenario 4 (緑色実線) 以外の全ての Scenario において、台車が黒色の破線で示す壁の位置に到達し、衝突が生起していることが分かる。また、Elite sample を用いた結果である Fig. 6.3 (a) 及び Fig. 6.5 (a) では、Scenario 2 (青色実線) では、 $0.3$  s 及び  $0.88$  s の計 2 回、Scenario 3 (赤色実線) では、 $0.3$  s 時に 1 度だけ壁との衝突が発生し、台車速度や振子角速度が急激に変化するジャンプ現象が確認できる。TOP1 sample を用いた結果である Fig. 6.3 (b) 及び Fig. 6.5 (b) では、Scenario 2 では、 $1.4$  s 時のみ、Scenario 3 では、 $0.4$  s 及び  $1.5$  s に衝突が発生している。ただし、Scenario 3 における衝突による状態量のジャンプの幅は、Scenario 2 のものと比較すると小さいことが分かる。

さらに、振り上げに直前に振子角度がピークに到達する瞬間に着目すると、意図的な衝突が可能な Scenario 2 では、衝突により発生した撃力を利用したことで最も速いタイミングで振子角速度がピークを迎えている。併せて、Fig. 6.4 や Fig. 6.6 の結果より、振り上げに要している時間や評価値  $J$  が収束するまでに要している時間については、衝突現象を予測モデルに含んでいる Scenario 2 が最短となっている。このことは、Table 6.2

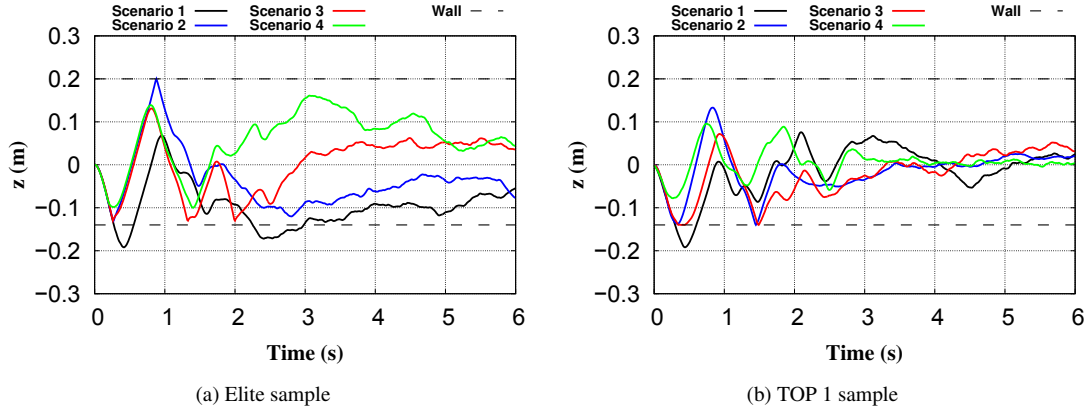


Fig. 6.2: Time response of cart position  $z$  (in numerical simulation of **Problem 8**)

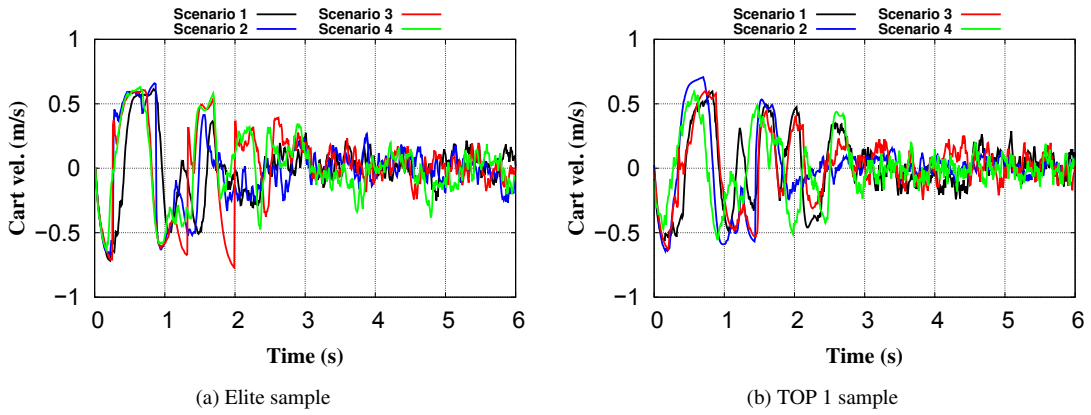


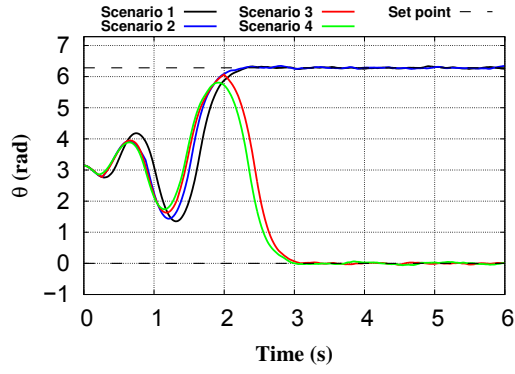
Fig. 6.3: Time response of cart velocity  $\dot{z}$  (in numerical simulation of **Problem 8**)

に示す, Scenario 及び Elite または TOP1 の制御器の種類ごとに, 振り子角度が倒立状態を表す  $0 \pm 0.1$  rad または  $2\pi \pm 0.1$  rad になり, かつ推定入力時系列を用いて計算した評価関数値  $J$  が 50 以下である状態が 0.5 s 以上継続した時間のうちの最初の時刻とする整定時刻の結果を見ても明らかである.

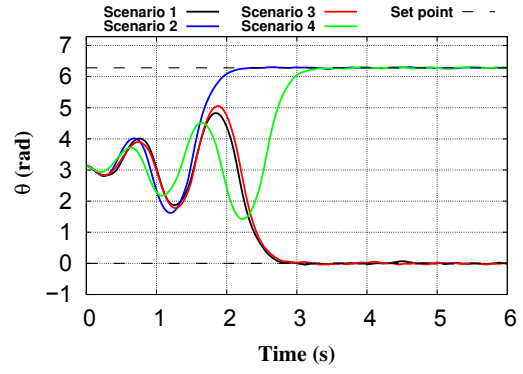
これらの結果より, 数値シミュレーションにおいて, 衝突を有効活用した制御が実現できていることを確認することが出来た.

## 6.6. 実機実験

Fig. 6.7~Fig. 6.11 は, 6.5 節同様に, それぞれ, Elite sample または TOP1 sample の MCMPC を用いた実機実験における台車位置, 台車速度, 振り子角度, 振り子角速度及び各制御時刻における推定入力時系列  $\hat{\mathbf{u}}_k$  を用いて演算した評価値  $J$  の時間応答をプロットしたものである. Fig. 6.7 中, 黒色破線は壁の位置を示しており, それぞれ左側の壁が台車の初期位置を原点とし  $-0.14$  m の位置, 右側の壁が  $0.18$  m の位置にある. 反発係数につ

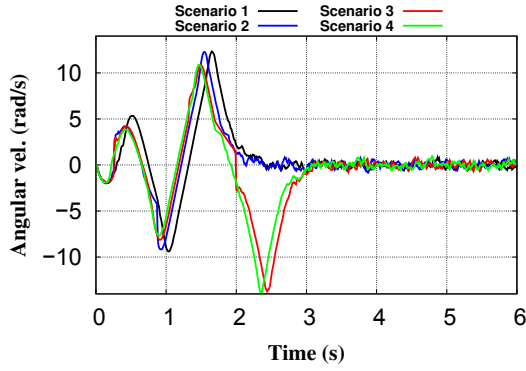


(a) Elite sample

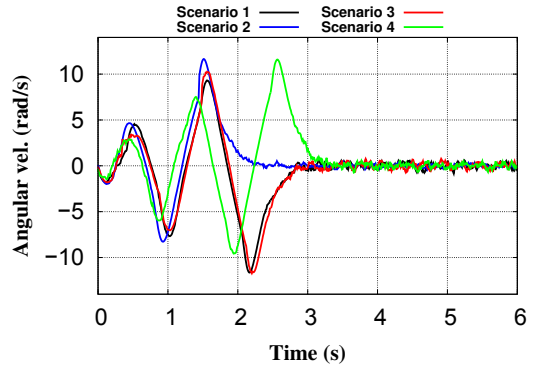


(b) TOP 1 sample

Fig. 6.4: Time response of angle of the pendulum  $\theta$  (in numerical simulation of **Problem 8**)

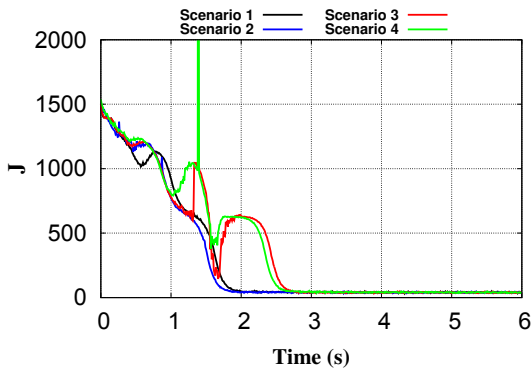


(a) Elite sample

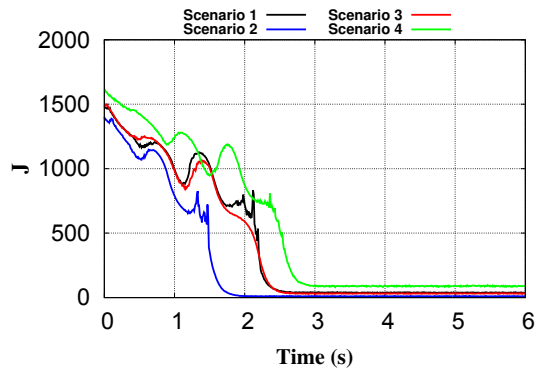


(b) TOP 1 sample

Fig. 6.5: Time response of angular velocity  $\dot{\theta}$  (in numerical simulation of **Problem 8**)



(a) Elite sample



(b) TOP 1 sample

Fig. 6.6: Cost value  $J(\bar{\mathbf{u}}_k)$  (in numerical simulation of **Problem 8**)

いては、(6.16)に示す通り、反発係数 $e$ が大きくなるほど衝突時に振子が受け取る撃力相当のエネルギーが大きくなるので、台車と壁の接触面にバネを挟むことで高反発を実現している。反発係数を求めるために行った事前実験の結果より $e = 0.3 \sim 0.4$ 程度であったので、Scenario 2においてはその中央値である $e = 0.35$ を用いて衝突に関する予測を行っている。

Elite sample 及び TOP1 sample いずれを用いた結果においても、Scenario 2の結果（青線でプロット）が最も速く制御目的である倒立安定状態に遷移している様子が確認できた。壁の位置については、ここに示した例以外の数値でも多くの例でScenario 2が最速で振り上げ安定化を実現できるといった結果を得た。また、実機実験における再現性という観点では、今回与えた問題設定の条件下でScenario ごとに20回の試行を行ったところ、ほとんどすべての試行において同様の結果を得た。

実機実験では、高反発を再現するのにバネを利用しているため、グラフ（例えば、Fig. 6.8やFig. 6.10）からは、衝突により大きなジャンプ現象が発生している様子は推察しづらいが、衝突を予測モデルに含んでいるScenario 2では、振子角速度がピークに差し掛かる直前で壁と接触しており、ピークに到達する時刻が他のScenarioの結果と異なっていることが分かる。

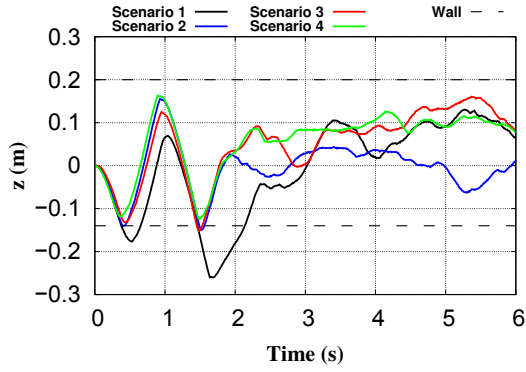
Table 6.2の結果では、整定時刻について、Scenario 3が最短となるケースも存在している。これは、予期せぬ衝突により、偶然、振り上げ安定化が上手くいったケースであると考えられる。また、Scenario 2において最長となるケースでさえも、Scenario 3を除く他の2つのScenarioと比べると高速での振り上げが実現できていることが確認できる。

以上の結果から、サンプル型MPCの一種であるMCMPCでは、評価関数や制御対象のダイナミクスに衝突などの不連続現象が内在している予測モデルを用いた最適制御問題であっても容易に扱うことができ、そのような現象を積極的に制御に取り入れたこれまでにない制御が実現可能であることを確認できた。

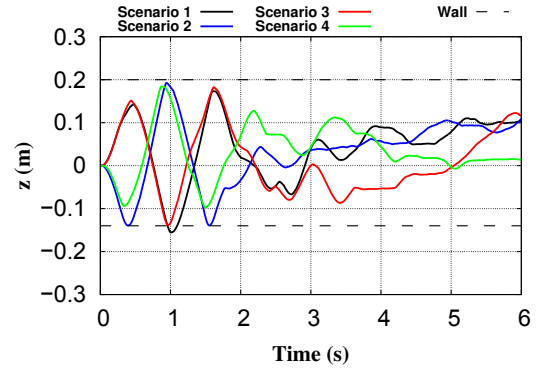
## 6.7. 6章のまとめ

本章では、サンプル型MPCの一種であるMCMPCの適用可能性について、その特長の1つである、最適化問題を解く際に勾配情報が不要であるという点に着目し、その特長を利用し不連続現象を積極的に利用した制御が実現できるかについて、台車型倒立振子の振り上げ安定化制御の問題において、台車が壁と衝突するエネルギーを活用可能な状況を作為し制御問題を考え、検証を行った。



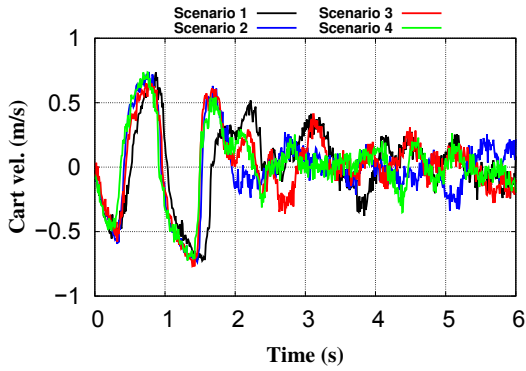


(a) Elite sample

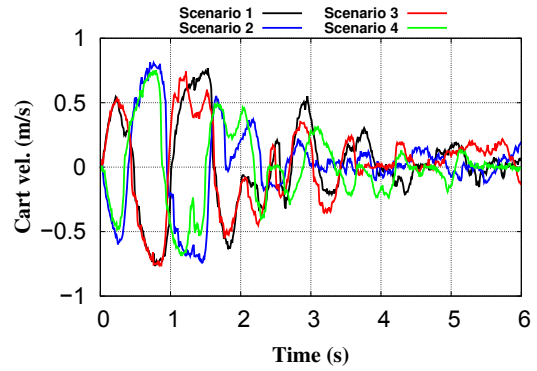


(b) TOP 1 sample

Fig. 6.7: Time response of cart position  $z$  (in the laboratory experiment of **Problem 8**)

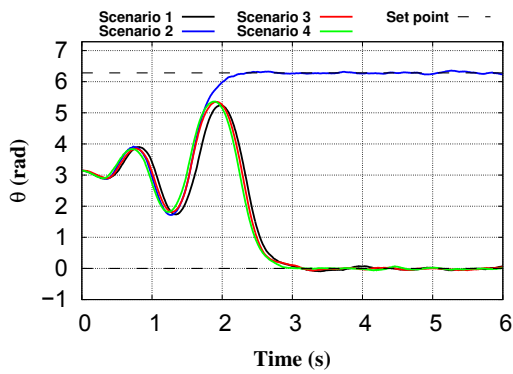


(a) Elite sample

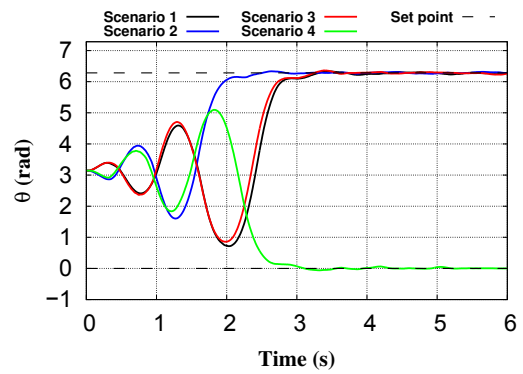


(b) TOP 1 sample

Fig. 6.8: Time response of cart velocity  $\dot{z}$  (in the laboratory experiment of **Problem 8**)



(a) Elite sample



(b) TOP 1 sample

Fig. 6.9: Time response of angle of the pendulum  $\theta$  (in the laboratory experiment of **Problem 8**)

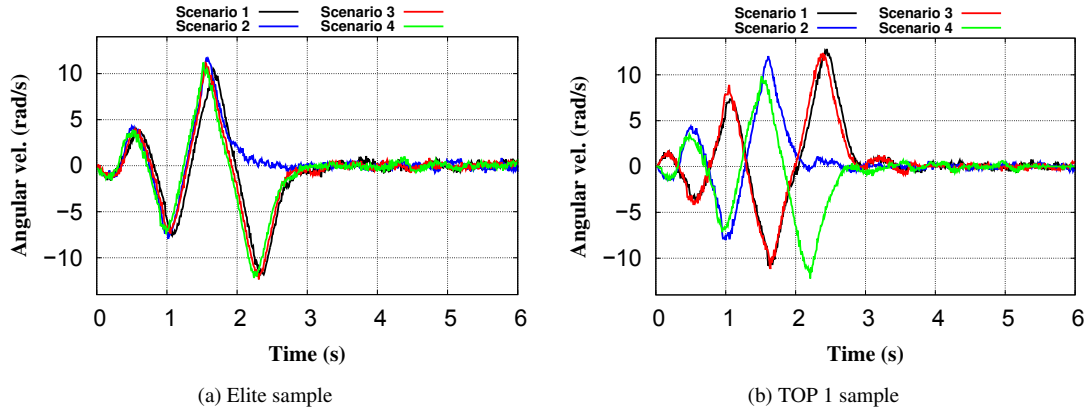


Fig. 6.10: Time response of angular velocity  $\dot{\theta}$  (in the laboratory experiment of **Problem 8**)

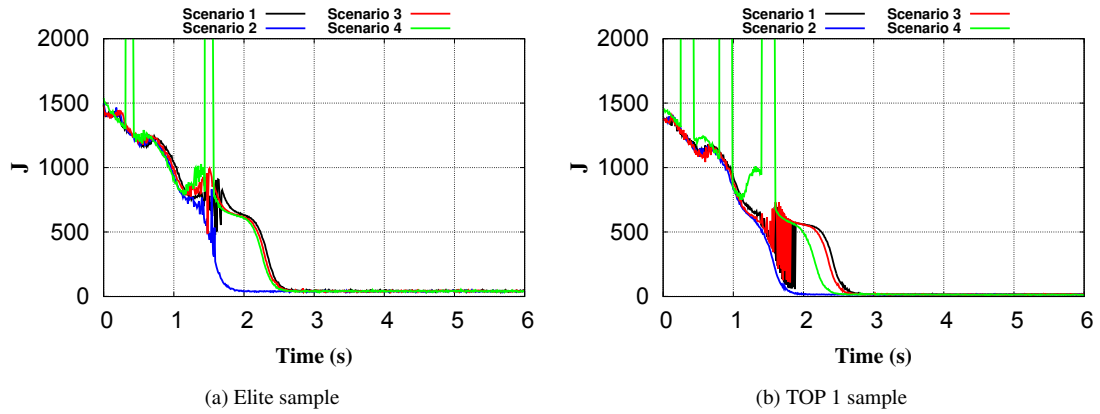


Fig. 6.11: Cost value  $J(\mathbf{u}_k)$  (in the laboratory experiment of **Problem 8**)

Table 6.2. Comparison of settling time (Results of 20 trails)

		Controller pattern	
		Elite sample	TOP1 sample
Simulation	Scenario 1	2.35 s	2.14 s
	Scenario 2	2.25 s	1.75 s
	Scenario 3	2.97 s	2.41 s
	Scenario 4	2.91 s	2.82 s
Laboratory experiment	Scenario 1	2.85 ~ 3.01 s	2.77 ~ 2.86 s
	Scenario 2	2.20 ~ 2.30 s	1.99 ~ 2.50 s
	Scenario 3	2.24 ~ 2.87 s	2.25 ~ 4.02 s
	Scenario 4	2.70 ~ 2.85 s	3.33 ~ 4.30 s

6.5 節及び 6.6 節で示したいずれの結果においても、衝突に関する予測モデルを保持している Scenario 2 が最速で評価値  $J$  を収束させ、最短時間で振り上げ安定化を実現できることを確認した。これは、予測モデルとして衝突に関する知識を保持することで、衝突を積極的に利用する挙動を予測し、その挙動を最適解として選択することができているためであると考えられる。また、MCMPC において、このような効果が確認できたのは、6.2 節で述べたような不連続性を有する問題ではあるものの、入力の変動に対する評価関数の変動のみに着目すると、剣山や稜線のような構造は有していない比較的扱いやすい構造であったのではないかと考えられる。これについては、今後何らかの方法で検証する必要がある。

本章で示した結果は、いくつかの先行研究 [51,87] で指摘されているサンプル型 MPC の利点の 1 つである、不連続な現象を含む系において、評価関数が局所的に微分不可となるような最適制御問題に対しても適用可能であるという内容を、実機実験の結果をベースに確認したものである。このような実例研究は、本章掲載の例題が初めてである。

さらに、本章で紹介した結果は、クアッドコプタの制御問題において、壁面との衝突が避けられないような状況で、衝突が姿勢変化に与える影響を小さくする制御を実現した加藤らの研究 [23] や、二足歩行ロボットにおける ZMP 制約下での重心軌道生成において、壁面接触を伴う不連続なシステムの制御などの問題 [72] に応用されているなど、サンプル型 MPC の適用範囲拡大に大きく貢献している。

## 第7章 総 括

非線形システムに対する一般的かつ統一的な制御系設計手法を与えるのは困難であるとされている。そこで、ある評価関数最小の意味で合理的な制御系を決定する枠組みである最適制御，中でも実時間最適制御に基づく MPC が注目されている [9]。MPC にあっても，特に確率・統計的最適化手法をベースとするサンプル型 MPC は，その最適化手法の汎用性の高さから，従来型の MPC では扱い困難とされていた様々な問題へ適用できると考えられている。一方で，サンプル型 MPC は，そのアイデア自体は古くから存在するものの，勾配型 MPC と比較したときの計算量の多さがネックとなり，実システム（特に，サンプリング周期の短い機械系など）への適用は近年（著者らが調査した限りでは 2016 年 [21,37] が最も古い機械系への適用例である。）やっとその見通しがたって来たというのが現状である。そのため，確率・統計的最適化手法が持つ最適化における汎用性の高さがどこまで MPC の問題設定に有効なのか？ 実時間実行という制約の中で十分な最適化精度が期待できるのか？ 期待できないとすれば，それにより制御を行う上で問題は生じないのか？ など多くの課題について十分な検討及び検証がなされていなかった。

本研究では，そのような状況を鑑み，サンプル型 MPC の一種である MCMPC を中心に，その収束特性，最適化の精度が制御に与える悪影響，逆に，確率・統計的最適化手法ベースの MPC ならではの長を生かした制御問題への適用とその効果などについて検討及び検証を行った。検証に用いた MPC の問題設定及び本研究の研究課題との関連などを，Table 7.1 及び Table 7.2 にまとめた。両表に示す通り，MPC のベンチマーク問題が纏められている文献 [10] と比較しても遜色のない多種多様な制御対象・問題設定において検証を行っている。具体的な成果については，後の 7.1 節で纏めることにするが，全体を俯瞰すると，現在，黎明期から発展期への転換期にあるサンプル型 MPC の課題である収束特性の問題を改善し，その性能向上に寄与するとともに，これまでの MPC の枠組みでは処理困難な問題に対する適用可能性を明らかにし，その適用可能範囲を拡大するきっかけを与えるなど，サンプル型 MPC の発展はもとより，MPC 研究の発展に十分貢献したものであると考える。

## 7.1. 本研究の成果

ここでは、当初定めた研究課題である、“性能向上”と“適用可能性の検討”の大きく2項目に分け、本研究の成果を纏める。

### 7.1.1 研究課題1：「性能向上」に対する本研究の成果

解析的な最適解が得られる離散時間線形システムの最適制御問題における MCMPC の同一制御周期内反復適用時の収束解析を行い、ユーザが設計可能なパラメータである、仮説的な入力をサンプルする分布である提案分布の分散共分散が MCMPC の収束特性と深く関わりのパラメータであることを明らかにした。その上で、当該パラメータを同一制御周期内反復の過程で更新する方法を検討し、その手法が線形系のみならず、非線形系の制御においても制御性能向上に寄与する方法であることを明らかにした。また、それらの関係をもとに、冷却法適用を前提とした際の MCMPC のパラメータ設計指針を示し

Table 7.1. Overview of the MPC benchmark problem addressed in this study and its relevance to the Research Questions.

Problem	Dimensions			Constraints			Verification items	Chapter
	x	u	$T_d$	x	u	other		
<b>Problem 1</b> Linear system	4	1	30	No	No	No	Research Question 1	Chap. 2
<b>Problem 2</b> ( <b>Problem 3</b> ) Cart & Pole (Single)	4	1	100	Yes	Yes	No		Chap. 3 (Chap. 4)
<b>Problem 4</b> Cart & Pole (Double)	6	1	110	Yes	Yes	No		Chap. 4
<b>Problem 5</b> Quadcopter	12	4	14	No	Yes	No	Research Question 2 ( <b>Resilient control</b> )	Chap. 5
<b>Problem 6</b> Cart & Pole (Single)	4	1	30	Yes	Yes	No	Research Question 2 ( <b>Tight constraint</b> )	
<b>Problem 7</b> 2D-Crane	6	2	15	Yes	Yes	<b>Yes</b>	Research Question 2 ( <b>Tight constraint</b> )	
<b>Problem 8</b> Cart & Pole (Single)	4	1	100	Yes	Yes	No	Research Question 2 ( <b>Discontinuity</b> )	Chap. 6

Table 7.2. Overview of MCMPC applications in related studies and its relevance to the Research Question 2 of this study.

Problem	Dimensions			Constraints			Relevance	Reference
	x	u	$T_d$	x	u	other		
Quadcopter	12	4	75	Yes	Yes	No	Discontinuity Discontinuity	[23]
Spacecraft (Single rotor control)	4	1	120	No	Yes	No		[62]
Mobile robot (Obstacle avoidance)	3	2	40	Yes	Yes	No	Discontinuity Multimodality	[88]
Biped robot	6	2	50	Yes	Yes	Yes	Discontinuity	[72]

た。これらの成果により，新たな制御対象に MCMPC を適用する際にユーザーが設計すべき MCMPC 固有のパラメータについて，その設計指針の大枠を示すことができたと言え，制御性能改善に資する有益な情報を提供できたと考えられる。

また，実際の制御においては，特に制御量が整定する直前において，最適化の精度（近似誤差）の問題が顕著になり，目標値追従性能などを毀損する恐れがあること明らかにした。これについて，現状取りうる最良の手段の1つとして，サンプル型 MPC の勾配不要の特長と，勾配型 MPC の最適化の精度を併せ持つ，サンプルベース・ニュートン型最適化法を提案した。当該手法は，問題の規模が大きい場合には，計算時間の観点で問題はあものの，勾配型 MPC が扱う問題と同程度の規模であれば，十分実時間で動かすことができ，その精度も，勾配型 MPC と同程度であることを確認した。さらには，勾配型 MPC が適用困難な問題にも適用でき，従来のサンプル型 MPC より優れた性能を発揮できるなど，性能向上の目標として掲げた内容を十分に満たした手法であることも明らかにした。本手法は，精度を保ちつつ計算量軽減が出来れば，より高度な制御に適用可能な実用的な制御アルゴリズムになりうると考えられる。

以上をまとめると，当初研究課題に挙げた，性能向上の課題について，十分その目標を達成できた考える。

### 7.1.2 研究課題2：「適用可能性」に対する本研究の成果

適用可能性については、従来の勾配型 MPC が処理可能かどうか 1 つの指標として、様々な問題において評価を行なった。それらの中でも、本研究における最も大きな貢献は、6 章で示した衝突などの不連続現象を積極的に活用した制御の実現可能性について実機実験レベルで検証したことである。これにより、Table 7.2 に示すように、これまで従来の勾配型 MPC が適用が困難とされていた問題への適用が試みられるようになるなど、適用可能範囲の拡大に貢献した。Table 7.2 に示す文献では、例えば、[62] では、評価関数そのものが微分不可関数で与えられる場合、[88] では、制約条件が微分不可関数で与えられる場合、[72] では、着地衝撃という不連続現象に加え、条件分岐を伴う状態制約が課されている場合などのように、本論文 6 章で扱った内容とは、少し毛色の異なる不連続性を有する MPC の問題設定に対する MCMPC の有効性が報告されている。

また、これまで意識されてこなかった勾配型 MPC の限界についても、その全貌までとは行かないものの個別の事象単位で垣間みることが出来た。例えば、突風などの不測の事態により状態量が大きく乱された直後からもとの軌道へと回復するためのレジリエンス制御 [26]、厳しい制約が課された非線形性の強い問題への適用などである。これらについては、勾配型 MPC では、ヘッセ行列が悪条件となる、ステップサイズの計算が間に合わないなど種々の要因で対処困難であると考えられるが、サンプル型 MPC では、適当な実行可能解を即座に求めることができるため対応可能であることを明らかにした。本研究においては、制約条件近傍でのモデル化誤差の影響などは考慮していないが、これまでの MPC では取り扱い困難な問題での有効性を確認したことで、適用可能範囲の拡大に貢献したと考えられる。

## 7.2. 今後の課題

ここでは、本研究の残された課題と今後の展望について議論する。

### 非線形系の最適制御問題における収束性解析

本研究での収束性解析は、解析解が得られる離散時間線形システムが対象であった。非線形については、3 章で簡単な数値例での検証を行なったのみで、解析的な結果を与えるには至らなかった。一方、文献 [45] を参考にすれば、ある極限を考えた際の収束性を議論する下地は既に存在していることが分かっている。文献 [45] は、焼きなまし法や MCMC

などの収束性解析にも応用されており、その考え方を上手く MCMPC や IT-MPC の枠組みに取り入れられれば、非線形系における収束性解析も行えるのではないかと考えている。いずれにしても、実時間最適化に基づく MPC において、その収束性を確かめるというのは、最も重要なタスクであるので、非線形系における収束性解析は、本研究で挙げる今後の課題の中でも最も優先度・重要度ともに高いタスクであると考えている。

## サンプル型 MPC の安定性解析

安定性解析は、MPC 研究の重要な研究課題の 1 つである。決定論的最適化をベースとする勾配型 MPC においては、例えば文献 [89,90] において、特殊な終端制約条件を課すことで評価関数自体にリアプノフ関数の性質を持たせることで、その制約が守られる範囲では、安定性が保証される枠組みなど多数の安定性解析に関する結果が纏められている。現実的な計算時間で、そのような制約を満たす解が得られるかなど課題はあるものの、終端制約条件を満たす解さえ得られれば、安定性を保証出来るというのは魅力的である。

一方、サンプル型 MPC においては、上述した方法をそのまま当てはめるのはかなり困難なアイデアであると言える。これは、実用上、その推定値が確率変数となるためである。上述した安定性解析の方策の多くは、終端制約条件と等式制約を課すものであり、有限個のサンプルからその終端制約条件を満たす解を引き当てるのは現実的に不可能と言える。

一方、サンプル型 MPC の性質を考慮すると文献 [91,92] などの考え方の方が適合していると考えられる。両者はいずれも“実行可能解”をもって、線形系の制御則で安定性が保証出来る集合まで状態量を遷移させることが可能か否かで安定性を議論する枠組みである。実行可能解の集合の内、所望の集合まで到達可能な実行可能解が存在する領域をサンプルで近似する方法を採っているため、現状、特定の時刻内に到達可能な実行可能解の領域を事前に計算し、各時刻で常にそれを満たす解を採用する方策が取られているが、実時間で領域（あるいは領域を定義する関数みたいなもの）を評価できるようになれば、安定性を確かめながらサンプル型 MPC を利用できるようになると考えられる。

これらの手法を参考にして、サンプル型 MPC の安定性解析について今後さらに議論を進めていく必要があると考えている。



## サンプルベース・ニュートン型最適化の計算量削減

5章で提案した手法は、ヘッセ行列の構造などは特に意識せずに、すべての要素について2次超曲面回帰を用いて導出している。そのため、2次超曲面回帰に必要な行列演算の計算量だけでみても入力の変数×予測ステップ数の3乗のオーダーとなる。実際に100 Hz程度で最適化を行うとなると、計算可能な問題の規模は、入力の変数と予測ステップ数の積が30～35程度が上限となる。そのため、より大規模な問題への適用を考える上では、ヘッセ行列の構造などを利用した計算量の削減が必要になる。

例えば、入力デカップリングを適切に施せば、各入力間の相関関係が解消され、ヘッセ行列のうちほとんどの要素が0になると考えられる。このような性質を利用して、真に必要な要素についてのみ2次超曲面回帰により導出するというのが1つのアイデアとして考えられる。また、評価関数に特別な工夫を凝らして、ヘッセ行列が対角行列になるようにして2次超曲面回帰を行う方法も考えられる。いずれにしても、問題の構造を上手く利用し、2次超曲面回帰で求める要素を削減するのが重要なアイデアになると考えられる。以上のような方向性を参考に、今後さらに改善していく必要がある。

## 適用可能性に関する更なる検証

本研究では、適用可能性について、従来の勾配型 MPC が処理困難／不可であるとされる問題というのを1つの指標として、Table 7.1 に示すような MPC のベンチマーク問題を中心にその検証を行った。当然のことながら、Table 7.1 や Table 7.2 に示す制御対象や問題設定が、実世界に実在する全ての非線形システムを網羅している訳ではない。また、本研究では、MCMPC や他のサンプル型 MPC の適用限界を適用可能性の指標とした評価も行っていない。

上記をふまえると、Table 7.1 や Table 7.2 に挙げた制御対象以外の実在する非線形システムに対する、本研究で掲げた意味での適用可能性についても今後議論していく必要があると言える。さらに、それらのうち、勾配型 MPC が処理可能でサンプル型 MPC が処理困難な問題、勾配型 MPC、サンプル型 MPC 双方が処理困難な問題のように、サンプル型 MPC の“適用限界”についても議論していく必要があると言える。

## 謝 辞

本論文は、筑波大学理工情報生命学術院システム情報工学研究群知能機能システム学位プログラムにおいて筑波大学システム情報系准教授 伊達央 先生のご指導のもと、著者が同学位プログラム博士課程学生として在籍した間に行なった研究成果をまとめたものです。同准教授には、本研究を進めるにあたり、大変ご多忙であるにも関わらず、終始変わらぬご指導・ご鞭撻を賜りました。ここに謹んで厚く感謝を申し上げます。

筑波大学システム情報系教授 坪内孝司 先生、並びに、筑波大学システム情報系教授 古賀弘樹 先生のお二人には、博士後期課程在籍時の副指導教員として、大学院セミナーなどの機会を通じて研究に向かう姿勢や研究に関する困難克服のための具体的な方策まで丁寧にご助言を戴くとともに本論文の細部にわたりご指導を戴きました。ここに深くお礼申し上げます。筑波大学システム情報系教授 河辺徹 先生、並びに、筑波大学システム情報系助教 澁谷長史 先生には、副査としてご助言を戴くとともに本論文の細部にわたりご指導を戴きました。ここに深くお礼申し上げます。

筑波大学システム情報系教授 大矢晃久 先生、並びに、筑波大学システム情報系助教 萬礼応 先生には、著者が、筑波大学知能ロボット研究室に在籍している間に、同研究室が開催するセミナーやシンポジウムなどの機会を通じて、研究開発に向かう姿勢を厳しくご指導を戴いたとともに研究の進め方や悩みについて親身になって相談にのって戴きました。ここに深くお礼申し上げます。また、同研究室の各位には研究遂行にあたり日頃より有益なご討論ご助言を戴いた。ここに感謝の意を表します。

本研究は、防衛省陸上自衛隊国内大学院研修生の研修業務の一環として行なったものであり、本研修の機会を与えて下さった防衛省陸上幕僚監部防衛部防衛課開発室の皆様、に、深謝の意を表します。また、本研修を遂行するにあたり、陸上自衛隊霞ヶ浦駐屯地業務隊総務科の皆様にはさまざまな面でお世話になりました。ここに深謝の意を表します。併せて、防衛大学校本科学生在籍時の私に、研究の楽しさや難しさ、やりがいなどを教えて戴いたとともに本研修に進むきっかけを与えてくださりました防衛大学校電気電子工学科教授 板宮敬悦 先生に深くお礼申し上げます。

最後に、これまで私をあたたく応援してくれた母，義両親，私を明るく励まし続けてくれた妻 あゆみ，娘 茉莉花，そして新しく生まれてこようとしているわが子に心から感謝します．

## 参考文献

- [1] 甲斐健也. 非線形制御 (発展編,< 特集> 初学者のための図解でわかる制御工学 ii). システム/制御/情報, Vol. 56, No. 6, pp. 306–309, 2012.
- [2] 大塚敏之. 非線形モデル予測制御の研究動向. システム/制御/情報, Vol. 61, No. 2, pp. 42–50, 2017.
- [3] 大嶋正裕, 小河守正. モデル予測制御-i: 基礎編: 発展の歴史と現状. システム/制御/情報, Vol. 46, pp. 286–293, 2002.
- [4] AI Propoi. Use of linear programming methods for synthesizing sampled-data automatic systems. *Automn. Remote Control*, Vol. 24, No. 7, pp. 837–844, 1963.
- [5] Jan Albersmeyer, Dörte Beigel, Christian Kirches, Leonard Wirsching, Hans Georg Bock, and Johannes P Schlöder. Fast nonlinear model predictive control with an application in automotive engineering. In *Nonlinear Model Predictive Control*, pp. 471–480. Springer, 2009.
- [6] Reurison Rodrigues, Andre Murilo, Renato Vilela Lopes, and Luiz Carlos Gadelha De Souza. Hardware in the loop simulation for model predictive control applied to satellite attitude control. *IEEE Access*, Vol. 7, pp. 157401–157416, 2019.
- [7] Atsushi Sakamoto, Yuichi Ikeda, Isao Yamaguchi, and Takashi Kida. Nonlinear model predictive control for large angle attitude maneuver of spacecraft with rw and rcs. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pp. 3202–3209. IEEE, 2016.
- [8] Utku Eren, Anna Prach, Başaran Bahadır Koçer, Saša V Raković, Erdal Kayacan, and Behçet Açıkmeşe. Model predictive control in aerospace systems: Current state and opportunities. *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 7, pp. 1541–1566, 2017.
- [9] 大塚敏之. 非線形最適制御入門. コロナ社, 2011.
- [10] Tobias Englert, Andreas Völz, Felix Mesmer, Sönke Rhein, and Knut Graichen. A software framework for embedded nonlinear model predictive control using a gradient-based augmented lagrangian approach (grampc). *Optimization and Engineering*, Vol. 20, No. 3, pp. 769–809, 2019.
- [11] 大塚敏之, 浜松正典, 永塚満, 川辺武俊, 向井正和, 西羅光, 山北昌毅, 李俊默. 実時間最適化による制御の実応用. コロナ社, 2015.
- [12] Toshiyuki Ohtsuka and Akira Kodama. Automatic code generation system for nonlinear receding horizon control. *Transactions of the Society of Instrument and Control Engineers*, Vol. 38, No. 7, pp. 617–623, 2002.
- [13] Daniel B Leineweber, Andreas Schäfer, Hans Georg Bock, and Johannes P Schlöder. An efficient multiple shooting based reduced sqp strategy for large-scale dynamic process optimization: Part ii: Software aspects and applications. *Computers & chemical engineering*, Vol. 27, No. 2, pp. 167–174, 2003.
- [14] Boris Houska, Hans Joachim Ferreau, and Moritz Diehl. Acado toolkit—an open-source framework for automatic control and dynamic optimization. *Optimal Control Applications and Methods*, Vol. 32, No. 3, pp. 298–312, 2011.

- [15] Colin N Jones, Alexander Domahidi, Manfred Morari, Stefan Richter, Fabian Ullmann, and Melanie Zeilinger. Fast predictive control: Real-time computation and certification. *IFAC Proceedings Volumes*, Vol. 45, No. 17, pp. 94–98, 2012.
- [16] Jouko Kalmari, Juha Backman, and Arto Visala. A toolkit for nonlinear model predictive control using gradient projection and code generation. *Control Engineering Practice*, Vol. 39, pp. 56–66, 2015.
- [17] Haoyang Deng and Toshiyuki Ohtsuka. A parallel code generation toolkit for nonlinear model predictive control. In *2018 IEEE Conference on Decision and Control (CDC)*, pp. 4920–4926. IEEE, 2018.
- [18] Markus Gifftthaler, Michael Neunert, Markus Stäuble, and Jonas Buchli. The control toolbox—an open-source c++ library for robotics, optimal and model predictive control. In *2018 IEEE International Conference on Simulation, Modeling, and Programming for Autonomous Robots (SIMPAN)*, pp. 123–129. IEEE, 2018.
- [19] Joel AE Andersson, Joris Gillis, Greg Horn, James B Rawlings, and Moritz Diehl. Casadi: a software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*, Vol. 11, No. 1, pp. 1–36, 2019.
- [20] Thomas Neve, Tom Lefebvre, and Guillaume Crevecœur. Comparative study of sample based model predictive control with application to autonomous racing. In *2022 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 1632–1638, 2022.
- [21] Grady Williams, Paul Drews, Brian Goldfain, James M. Rehg, and Evangelos A. Theodorou. Aggressive driving with model predictive path integral control. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1433–1440, 2016.
- [22] Riccardo Spica, Eric Cristofalo, Zijian Wang, Eduardo Montijano, and Mac Schwager. A real-time game theoretic planner for autonomous two-player drone racing. *IEEE Transactions on Robotics*, Vol. 36, No. 5, pp. 1389–1403, 2020.
- [23] 加藤幹也, 伊達央, 仲谷真太郎, 大矢晃久. モンテカルロモデル予測制御による壁面との衝突を考慮したクアッドコプタの制御. 計測自動制御学会論文集, Vol. 57, No. 9, pp. 379–390, 2021.
- [24] Christian P Robert and George Casella. *Monte Carlo statistical methods*, Vol. 2. Springer, 1999.
- [25] Dirk P Kroese and Reuven Y Rubinstein. Monte carlo methods. *Wiley Interdisciplinary Reviews: Computational Statistics*, Vol. 4, No. 1, pp. 48–58, 2012.
- [26] H. Date, S. Nakatani, and M. Kato. Resilient control: Toward overcoming unexpected. In *2019 58th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*, pp. 1089–1092, 2019.
- [27] Grady Williams, Paul Drews, Brian Goldfain, James M Rehg, and Evangelos 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.
- [28] Shintaro Nakatani and Hisashi Date. Swing up control of inverted pendulum on a cart with collision by monte carlo model predictive control. In *2019 58th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*, pp. 1050–1055. IEEE, 2019.
- [29] Shintaro Nakatani and Hisashi Date. Improving the accuracy of sample-based model predictive control via sample-based newton-like method with approximated hessian and gradient by quadratic regression. In *2021 60th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*, pp. 85–92, 2021.
- [30] Hanchen Lu, Qun Zong, Shupeng Lai, Bailing Tian, and Lihua Xie. Real-time perception-limited motion planning using sampling-based mpc. *IEEE Transactions on Industrial Electronics*, Vol. 69, No. 12, pp. 13182–13191, 2022.

- [31] Toshiyuki Ohtsuka. Time-variant receding-horizon control of nonlinear systems. *Journal of Guidance, Control, and dynamics*, Vol. 21, No. 1, pp. 174–176, 1998.
- [32] Elilini Siva, Paul Goulart, Jan Maciejowski, and Nikolas Kantas. Stability of model predictive control using markov chain monte carlo optimisation. In *2009 European Control Conference (ECC)*, pp. 2851–2856. IEEE, 2009.
- [33] Dominik Stahl and Jan Hauth. Pf-mpc: Particle filter-model predictive control. *Systems & Control Letters*, Vol. 60, No. 8, pp. 632–643, 2011.
- [34] Fang Xu, Hong Chen, Xun Gong, and Qin Mei. Fast nonlinear model predictive control on fpga using particle swarm optimization. *IEEE Transactions on Industrial Electronics*, Vol. 63, No. 1, pp. 310–321, 2015.
- [35] Grady Williams, Andrew Aldrich, and Evangelos Theodorou. Model predictive path integral control using covariance variable importance sampling. *arXiv preprint arXiv:1509.01149*, 2015.
- [36] Grady Williams, Andrew Aldrich, and Evangelos A. Theodorou. Model predictive path integral control: From theory to parallel computation. *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 2, pp. 344–357, 2017.
- [37] Shimpei Ohyama and Hisashi Date. Parallelized nonlinear model predictive control on gpu. In *2017 11th Asian Control Conference (ASCC)*, pp. 1620–1625, 2017.
- [38] M Mallaiah, K Rama Rao, and Ch Venkateswarlu. A simulated annealing optimization algorithm based nonlinear model predictive control strategy with application. *Evolving Systems*, Vol. 12, No. 1, pp. 225–231, 2021.
- [39] Hilbert J Kappen. Path integrals and symmetry breaking for optimal control theory. *Journal of statistical mechanics: theory and experiment*, Vol. 2005, No. 11, p. P11011, 2005.
- [40] Debasis Mitra, Fabio Romeo, and Alberto Sangiovanni-Vincentelli. Convergence and finite-time behavior of simulated annealing. *Advances in applied probability*, Vol. 18, No. 3, pp. 747–771, 1986.
- [41] Harold Szu and Ralph Hartley. Fast simulated annealing. *Physics letters A*, Vol. 122, No. 3-4, pp. 157–162, 1987.
- [42] Wing Hung Wong and Faming Liang. Dynamic weighting in monte carlo and optimization. *Proceedings of the National Academy of Sciences*, Vol. 94, No. 26, pp. 14220–14224, 1997.
- [43] Christian P Robert and Caroline Soubiran. Estimation of a normal mixture model through gibbs sampling and prior feedback. *Test*, Vol. 2, No. 1, pp. 125–146, 1993.
- [44] Olivier Cappé, Arnaud Guillin, Jean-Michel Marin, and Christian P Robert. Population monte carlo. *Journal of Computational and Graphical Statistics*, Vol. 13, No. 4, pp. 907–929, 2004.
- [45] Martin Pincus. -a closed form solution of certain programming problems. *Operations Research*, Vol. 16, No. 3, pp. 690–694, 1968.
- [46] Shintaro Nakatani and Hisashi Date. Sampling variance update method in monte carlo model predictive control. *IFAC-PapersOnLine*, Vol. 53, No. 2, pp. 1274–1281, 2020. 21st IFAC World Congress.
- [47] Thomas Neve, Tom Lefebvre, and Guillaume Crevecoeur. Comparative study of sample based model predictive control with application to autonomous racing. In *2022 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 1632–1638. IEEE, 2022.
- [48] Charles J Geyer. Estimation and optimization of functions. pp. 241–258, 1996.

- [49] Roman Liesenfeld and Jean-Francois Richard. Univariate and multivariate stochastic volatility models: estimation and diagnostics. *Journal of empirical finance*, Vol. 10, No. 4, pp. 505–531, 2003.
- [50] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, Vol. 22, No. 1, pp. 79–86, 1951.
- [51] 佐藤訓志. 経路積分解析と統計的サンプリングを用いた非線形確率最適制御問題の解法. システム/制御/情報, Vol. 62, No. 5, pp. 181–187, 2018.
- [52] Homanga Bharadhwaj, Kevin Xie, and Florian Shkurti. Model-predictive control via cross-entropy and gradient-based optimization. In *Learning for Dynamics and Control*, pp. 277–286. PMLR, 2020.
- [53] 金光秀雄, 今野英明, 高橋伸幸ほか. 多点局所探索に基づく大域的最適化アルゴリズム (1). 北海道教育大学紀要, 自然科学編, 2008.
- [54] 金光秀雄. 多点局所探索に基づく大域的最適化アルゴリズム (2): 上下制限付き連続最適化問題における直線探索法と局所探索法の検討. 北海道教育大学紀要. 自然科学編, 2018.
- [55] Abraham Savitzky and Marcel JE Golay. Smoothing and differentiation of data by simplified least squares procedures. *Analytical chemistry*, Vol. 36, No. 8, pp. 1627–1639, 1964.
- [56] David Pérez-Morales and Vincent Fremont. Information-theoretic sensor-based predictive control for autonomous vehicle navigation: A proof of concept. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pp. 879–884. IEEE, 2021.
- [57] Grady Williams, Nolan Wagener, Brian Goldfain, Paul Drews, James M Rehg, Byron Boots, and Evangelos A Theodorou. Information theoretic mpc for model-based reinforcement learning. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1714–1721. IEEE, 2017.
- [58] Yunduan Cui, Shigeki Osaki, and Takamitsu Matsubara. Autonomous boat driving system using sample-efficient model predictive control-based reinforcement learning approach. *Journal of Field Robotics*, Vol. 38, No. 3, pp. 331–354, 2021.
- [59] 児島晃, 大塚敏之. モデル予測制御の考え方. 計測と制御, Vol. 42, No. 4, pp. 310–312, 2003.
- [60] Omar Khan and Chandra Mouli R Madhuranthakam. Optimization of robust model predictive controllers for second-order biological processes using markov chain monte carlo techniques. *Industrial & Engineering Chemistry Research*, Vol. 58, No. 47, pp. 21569–21583, 2019.
- [61] 仲谷真太郎, 伊達央. 不連続な変化を伴う制御対象へのモンテカルロモデル予測制御の適用—台車の衝突を考慮した台車型倒立振子の振り上げ高速化と安定制御—. 計測自動制御学会論文集, Vol. 56, No. 3, pp. 116–123, 2020.
- [62] 竹崎大輔, 伊達央. モンテカルロ経済モデル予測制御による 1 ロータ宇宙機の太陽光発電量最大化. 計測自動制御学会論文集, Vol. 57, No. 3, pp. 195–202, 2021.
- [63] Nikolaus Hansen. The cma evolution strategy: a comparing review. *Towards a new evolutionary computation*, pp. 75–102, 2006.
- [64] 秋本洋平. Evolution strategies による連続最適化-cma-es の設計原理と理論的基盤. システム/制御/情報, Vol. 60, No. 7, pp. 292–297, 2016.
- [65] 仲谷真太郎, 伊達央. Mcmpc における目的関数の形状を考慮した分散共分散行列の更新. 第 8 回計測自動制御学会制御部門マルチシンポジウム (2021 年 3 月 1 日～4 日・オンライン), 2021.
- [66] 井村順一. ハイブリッドシステムの制御理論構築を目指して: いかに捉えるか? システム/制御/情報, Vol. 43, No. 12, pp. 651–659, 1999.

- [67] 大宮真弓, 松島正知. 統計解析法入門: 推定・検定・分散分析の考え方. 森北出版, 2015.
- [68] 中野良平. シミュレーテッドアニーリング: 基礎と最新技術. 人工知能, Vol. 9, No. 3, pp. 365–372, 1994.
- [69] Stuart Geman and Donald Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on pattern analysis and machine intelligence*, No. 6, pp. 721–741, 1984.
- [70] 川田昌克. 物理法則に基づくモデリング (基礎編,< 特集> 初学者のための図解でわかる制御工学 i). システム/制御/情報, Vol. 56, No. 4, pp. 166–169, 2012.
- [71] Sanjay P Bhat and Dennis S Bernstein. A topological obstruction to continuous global stabilization of rotational motion and the unwinding phenomenon. *Systems & control letters*, Vol. 39, No. 1, pp. 63–70, 2000.
- [72] 安藤日出海, 仲谷真太郎, 伊達央. 二足歩行ロボットにおける上体の壁面接触と zmp 制約を考慮したモンテカルロモデル予測制御を用いた重心軌道生成. 計測自動制御学会論文集 (in press).
- [73] 野澤勇樹, 中村信弘. 確率的依存構造をもつコピュラモデル. 統計数理, Vol. 68, No. 1, pp. 87–106, 2020.
- [74] Shun-Ichi Amari. Natural gradient works efficiently in learning. *Neural computation*, Vol. 10, No. 2, pp. 251–276, 1998.
- [75] Youhei Akimoto, Yuichi Nagata, Isao Ono, and Shigenobu Kobayashi. Bidirectional relation between cma evolution strategies and natural evolution strategies. In *International conference on parallel problem solving from nature*, pp. 154–163. Springer, 2010.
- [76] 逸見知弘, 明聡, 井上昭, 植木信幸, 平嶋洋一. 台車型直列二重倒立振子の振り上げ制御. 日本機械学会論文集 C 編, Vol. 71, No. 704, pp. 1269–1275, 2005.
- [77] Ilya Loshchilov, Marc Schoenauer, Michele Sebag, and Nikolaus Hansen. Maximum likelihood-based online adaptation of hyper-parameters in cma-es. In *International Conference on Parallel Problem Solving from Nature*, pp. 70–79. Springer, 2014.
- [78] James C Spall. Adaptive stochastic approximation by the simultaneous perturbation method. *IEEE transactions on automatic control*, Vol. 45, No. 10, pp. 1839–1853, 2000.
- [79] Siti Mahani Marjugi and Wah June Leong. Diagonal hessian approximation for limited memory quasi-newton via variational principle. *Journal of Applied Mathematics*, Vol. 2013, , 2013.
- [80] Magnus Rudolph Hestenes. *Newton’s Method and the Gradient Method*, pp. 1–80. Springer New York, New York, NY, 1980.
- [81] Sotaro Katayama, Masahiro Doi, and Toshiyuki Ohtsuka. A moving switching sequence approach for nonlinear model predictive control of switched systems with state-dependent switches and state jumps. *International Journal of Robust and Nonlinear Control*, Vol. 30, No. 2, pp. 719–740, 2020.
- [82] David Q Mayne. Model predictive control: Recent developments and future promise. *Automatica*, Vol. 50, No. 12, pp. 2967–2986, 2014.
- [83] Junmuk Lee and Masaki Yamakita. Nonlinear model predictive control for constrained mechanical systems with state jump. In *2006 IEEE Conference on Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control*, pp. 585–590. IEEE, 2006.
- [84] Francesco Borrelli, Mato Baotić, Alberto Bemporad, and Manfred Morari. Dynamic programming for constrained optimal control of discrete-time linear hybrid systems. *Automatica*, Vol. 41, No. 10, pp. 1709–1721, 2005.



- [85] 片山想太郎, 大塚敏之. スイッチドシステムの数値最適制御と歩行ロボットへの応用. 自動制御連合講演会講演論文集 第 64 回自動制御連合講演会, pp. 329–336. 自動制御連合講演会, 2021.
- [86] Pietro Belotti, Christian Kirches, Sven Leyffer, Jeff Linderoth, James Luedtke, and Ashutosh Mahajan. Mixed-integer nonlinear optimization. *Acta Numerica*, Vol. 22, pp. 1–131, 2013.
- [87] Ziyi Wang, Grady Williams, and Evangelos A Theodorou. Information theoretic model predictive control on jump diffusion processes. In *2019 American Control Conference (ACC)*, pp. 1663–1670. IEEE, 2019.
- [88] Tatsuya Toda, Shintaro Nakatani, and Hisashi Date. Clustering method for monte carlo model predictive control. *IFAC-PapersOnLine*, Vol. 54, No. 14, pp. 251–256, 2021.
- [89] David Q Mayne, James B Rawlings, Christopher V Rao, and Pierre OM Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, Vol. 36, No. 6, pp. 789–814, 2000.
- [90] 大嶋正裕, 関宏也. モデル予測制御-v: 非線形モデル予測制御. システム/制御/情報, Vol. 47, No. 1, pp. 52–57, 2003.
- [91] Ruxandra Valentina Bobiti. *Sampling-driven stability domains computation and predictive control of constrained nonlinear systems*. PhD thesis, PhD thesis, 2017. URL <https://pure.tue.nl/ws/files/78458403/...>, 2017.
- [92] Pierre OM Scokaert, David Q Mayne, and James B Rawlings. Suboptimal model predictive control (feasibility implies stability). *IEEE Transactions on Automatic Control*, Vol. 44, No. 3, pp. 648–654, 1999.

## 付録A 重み付き確率(2.32)の期待値の導出

ここでは, (2.32) から (2.33) を導出する具体的な計算過程について記述する. 以下は, (2.32) の指数部を, 前反復時に得た推定値  $\tilde{\mathbf{u}}$  と提案分布から生成したサンプル  $\hat{\mathbf{u}}$  を用いて  $\hat{\mathbf{u}}$  についての 2 次形式に変形する計算過程を示している.

$$\begin{aligned}
 \mathbb{E}(\tilde{\mathbf{u}}) &= \int w(\hat{\mathbf{u}}) p(\hat{\mathbf{u}}) d\hat{\mathbf{u}} \\
 &= \frac{1}{\sqrt{(2\pi)^{mT_d} |\Sigma|}} \int \exp \left( -\frac{1}{2\eta} (\hat{\mathbf{u}} - \mathbf{u}^*)^T \hat{Q} (\hat{\mathbf{u}} - \mathbf{u}^*) - (\hat{\mathbf{u}} - \tilde{\mathbf{u}})^T \Sigma^{-1} (\hat{\mathbf{u}} - \tilde{\mathbf{u}}) \right) d\hat{\mathbf{u}} \\
 &= C_1 \int \exp \left( -\frac{1}{2\eta} (\hat{\mathbf{u}} - \mathbf{u}^*)^T \hat{Q} (\hat{\mathbf{u}} - \mathbf{u}^*) - \frac{1}{2\sigma^2} (\hat{\mathbf{u}} - \tilde{\mathbf{u}})^T \mathbf{E}_{\mathbf{u}} (\hat{\mathbf{u}} - \tilde{\mathbf{u}}) \right) d\hat{\mathbf{u}} \\
 &= C_2 \int \exp \left( -\hat{\mathbf{u}}^T \left( \frac{1}{2\eta} \hat{Q} + \frac{1}{2\sigma^2} \mathbf{E}_{\mathbf{u}} \right) \hat{\mathbf{u}} + \frac{1}{\eta} (\mathbf{u}^*)^T \hat{Q} \hat{\mathbf{u}} + \frac{1}{\sigma^2} \tilde{\mathbf{u}}^T \hat{\mathbf{u}} \right) d\hat{\mathbf{u}} \\
 &= C_3 \int \exp \left( \frac{1}{2\eta\sigma^2} \hat{\mathbf{u}}^T (\sigma^2 \hat{Q} + \eta \mathbf{E}_{\mathbf{u}}) \hat{\mathbf{u}} + \frac{\eta}{\sigma^2} (\mathbf{u}^*)^T \hat{Q} \hat{\mathbf{u}} + \frac{\sigma^2}{\eta\sigma^2} \tilde{\mathbf{u}}^T \hat{\mathbf{u}} \right) d\hat{\mathbf{u}} \quad (\text{A.1}) \\
 &= C_4 \int \exp \left( -\frac{1}{2\eta\sigma^2} (\hat{\mathbf{u}} - \tilde{\mathbf{u}})^T (\sigma^2 \hat{Q} + \eta \mathbf{E}_{\mathbf{u}}) (\hat{\mathbf{u}} - \tilde{\mathbf{u}}) \right) d\hat{\mathbf{u}} \quad (\text{A.2})
 \end{aligned}$$

$C_1$  から  $C_4$  は,  $\hat{\mathbf{u}}$  に依存しない項, すなわち, 積分の演算を行う際には定数とみなせる項をまとめたものである. このとき, (A.1) の指数部を  $G$  とし,  $G$  の停留条件として  $\hat{\mathbf{u}} = \tilde{\mathbf{u}}$  における偏微分を考えると,

$$\left. \frac{\partial G}{\partial \hat{\mathbf{u}}} \right|_{\hat{\mathbf{u}}=\tilde{\mathbf{u}}} = (\sigma^2 \hat{Q} + \eta \mathbf{E}_{\mathbf{u}}) \tilde{\mathbf{u}} - (\sigma^2 \hat{Q} \mathbf{u}^* + \eta \tilde{\mathbf{u}}) = 0 \quad (\text{A.3})$$

となり, (A.3) を  $\tilde{\mathbf{u}}$  について解くと, (2.33) となる.

また, (A.2) は,  $\tilde{\mathbf{u}}$  を平均,  $(\sigma^2 \hat{Q} + \eta \mathbf{E}_{\mathbf{u}})^{-1}$  を分散共分散行列に持つ多変量正規分布の密度関数と同じ構造となっていることが分かる. このとき, 推定値である  $\tilde{\mathbf{u}}$  のばらつきを表す分散共分散行列  $\Sigma_S$  は,

$$\Sigma_S = \eta \sigma^2 (\sigma^2 \hat{Q} + \eta \mathbf{E}_{\mathbf{u}})^{-1} \quad (\text{A.4})$$

で与えられることが分かる.