修士論文

実時間制御に向けた
サンプリングベースモデル予測制御の効率化

福本 晃汰

奈良先端科学技術大学院大学
先端科学技術研究科
情報理工学プログラム

主指導教員: 杉本 謙二 教授
知能システム制御 研究室（情報科学領域）

令和 4 年 3 月 15 日提出

本論文は奈良先端科学技術大学院大学先端科学技術研究科に
修士 (工学) 授与の要件として提出した修士論文である.

福本 晃汰

審査委員：

主査	杉本 謙二	（情報科学領域 教授）
	池田 和司	（情報科学領域 教授）
	小林 泰介	（情報科学領域 助教）

実時間制御に向けた
サンプリングベースモデル予測制御の効率化*

福本 晃汰

内容梗概

近年，機械技術の発展により自動運転や多脚ロボットといった，モデルや拘束条件が複雑で制御周期の短いシステムに対する制御手法の研究が盛んに行われている．0 次最適化手法のクロスエントロピー法を援用したサンプリングベースのモデル予測制御は，対象システムの制限が最も緩いモデル予測制御法であり，様々な研究で活用されている．しかしながらクロスエントロピー法は計算コストが極めて高く，計算時間が十分でない場合は制御が困難であるか，危険な状態へ遷移する可能性があり，実時間制御には適さないとされてきた．既存研究のクロスエントロピー法では，行動を確率変数とする現在の方策分布と，最適な方策分布との Kullback-Leibler ダイバージェンス（KL ダイバージェンス）を最小化するように最適化を図るが，このとき最小化される KL ダイバージェンスが Forward KL と呼ばれる引数順であることに着目する．Forward KL の最小化は複数のモードに対して包括的にフィッティングを行う．しかしその性質は実時間制御において，探索を優先しすぎている．

そこで本研究では，Forward KL と引数順を逆転させた，Reverse KL を最小化することで単一のモードに対してフィッティングを行い，不要なサンプルを排除する．この新たな定式化の中で，サンプリング軌道に対して負の重みを与えることで，更に不要なサンプルを避けるように更新する．また，得られる更新則が鏡像降下法となることに着目し，加速鏡像降下法へ発展させ更新を効率化する．これらの改良により，サンプリングベースモデル予測制御の実時間における制御性能向上を

*奈良先端科学技術大学院大学 先端科学技術研究科 修士論文, 令和 4 年 3 月 15 日.

図る．

本稿ではこの提案手法の有効性検証として，シミュレーション上での高速道路の実時間の走行制御において，成功率上昇による制御性能の向上を確認した．

キーワード

モデル予測制御法，クロスエントロピー法，鏡像降下法，自動運転

Improving Efficiency of Sampling-based Model Predictive Control for Real-time Control*

Fukumoto Kota

Abstract

In recent years, the development of mechanical technology has led to a lot of research on control methods for systems with complex models, constraints, and short control periods, such as autonomous driving and multi-legged robots. Sampling-based model predictive control using cross entropy method (CEM), which is a zero-order optimization method, has been used in many studies because it has the least restriction on the target model. However, CEM has been considered unsuitable for real-time control due to high computational cost. In the existing CEM, optimization is performed to minimize the Kullback-Leibler (KL) divergence between the current policy distribution and the optimal distribution, where the action is a stochastic variable. The KL divergence to be minimized is in the order of arguments called the forward KL. The minimization of the forward KL is mean-seeking behavior such as a comprehensive fitting for multiple modes. Therefore, this property gives too much priority to search in real time control.

In this study, we accelerate the convergence of the variance for a single mode by minimizing the reverse KL, and exclude useless samples. With this new formulation, we update the sampling trajectories by giving negative weights to them in order to avoid the samples. In addition, by focusing on the fact that the obtained update rule is a mirror descent algorithm, we improve it to an accelerated mirror descent algorithm with configurations suitable for the problem in order to make the update speed more efficient. With these improvements,

*Master's Thesis, Graduate School of Science and Technology, Nara Institute of Science and Technology, March 15, 2022.

we aim to improve the real-time control performance of sampling-based model predictive control. In this paper, we confirmed the improvement of the control performance by the success rate increase in the highway driving real-time control on the simulation.

Keywords:

Model Predictive Control, Cross Entropy Method, Mirror Descent, Autonomous Driving

目次

図目次

表目次

第 1 章　序論

1.1　はじめに

近年，自動運転や多脚ロボットといった，モデルや拘束条件が複雑で制御周期の短いシステムの制御が求められている．このようなシステムに対して有効な制御手法の一つにモデル予測制御（Model Predictive Control; MPC）がある [1, 2]. MPC はオートラリーの高速走行制御 [3, 4] やヘリコプターの自律曲芸飛行 [5] などの高度な制御目標に対して特に有効な制御手法である．

MPC はその特徴として拘束条件を考慮しながら最適行動を決定することができる．そのため，拘束条件に漸近する状態が最も収益が高いとされる場合に特に有効と言える．例えば，MPC はプラント産業の現場で生まれたプロセス制御の技術であるが，プラントにおける温度制御では，拘束条件に漸近する温度で操業することで最も収益が高い場合が多いことから，MPC が有効活用されてきた [6, 7].

このような特徴を活用するため，近年では電機システムのような計算時間が十分に取れないようなシステムに対しても適用可能な MPC の研究がされている．

また，MPC は最適制御問題をどのように解くかによって幾つかに大別できるが，その中でも対象システムの制限が最も緩く汎用的なクロスエントロピー法（Cross Entropy Method; CEM） [8] を援用したサンプリングベースの手法が増えている [9–11]．これは勾配ベースの MPC と比べ，システムのモデルや最適化したいコスト関数が微分不可能な場合においても適用可能であり，なおかつ大域最適解への収束性も優れるなどの多くの利点が挙げられる．一方で，最適解への収束性に関しては，十分な（理論上は無限大の）サンプル数と更新回数（イテレーション）が要求されるため，計算コストが極めて高くなってしまう．そのため，サンプリングベースの MPC を用いて実時間制御を行うためには，高効率なアルゴリズムへと改良する必要がある．

また，このような更新ごとに解が徐々に最適解に収束するようなアルゴリズムの性質を，Anytime 性と呼ぶ [12]．このような性質を持つアルゴリズムを用いて実時間制御を行う場合，つまりイテレーション回数が十分でない場合，MPC で扱うソフトな拘束条件を満たさない可能性もあり，対象のシステムが危険な状態に推移

する可能性をはらむ．このような危険性は実機を用いる場合に，特に留意すべきである．

1.2 関連研究

1.2.1 モデル予測制御

MPC は，現在の時刻からホライゾンの時刻までの区間の状態や行動に関するコスト関数を最適化するような行動系列を導出し，現在の時刻の行動を採用する制御手法である．特に非線形なモデルや非凸なコスト関数に対する MPC のコスト関数の最適化手法として一般的に用いられるものとして，逐次線形二次レギュレータ (iterative Linear Quadratic Regulator; iLQR) [13] や，iterative Linear Quadratic Gaussian(iLQG) [14]，連続最小二乗残差 (Continuation/Generalized Minimum Residual; C/GMRES) 法 [15] 等があげられる．

iLQR/iLQG は，ベルマン最適性の原理に基づいた再帰的なコスト関数に関する式を定義することで，制御則を導出する手法である．先行研究 [16] では iLQG を用いて MuJoCo と呼ばれる物理シミュレータにおいて，任意の姿勢からの立ち上がりや，大きな外乱から回復するなどのタスクを実行するヒューマノイドロボットの実時間制御に適用されている．しかし，iLQG では最適化の際にコスト関数のヘシアンを必要とし，なおかつその行列が正定値でなければならないため，対象とするシステムが限定されている．

C/GMRES 法はコスト関数に関するハミルトニアンの最適性必要条件から定義される非線形連立偏微分方程式に Continuation method [17] を適用して漸近的に条件を満たす解を求められる形に式変形し，GMRES 法 [18, 19] を用いてそれを解くことで最適行動を高速に導出する手法である．先行研究ではこの手法を用いて 4 輪車両の衝突回避 [20] や，実時間でのホバークラフトの位置制御 [21] など，様々な手法に応用されている．しかし，C/GMRES 法では，ハミルトニアンを偏微分した値を必要とするため，微分不可能なコスト関数を持つシステムに適用することは難しい．

このように，サンプリングベースではない MPC において，いくつかの手法では実時間制御へ応用されているが，どれも適用する対象は限定されてしまう [22].

1.2.2　クロスエントロピー法によるモデル予測制御

最適化に CEM を援用したサンプリングベースの MPC では，イテレーション毎に方策分布から行動をサンプリングし，ダイナミクスモデルとコスト関数に通した値を用いて方策分布を更新する．この更新を十分な回数行うことで，方策分布はコスト関数を最適化するような行動を出力する確率分布に収束する．この手法では，微分不可能で非凸なモデルやコスト関数を持つシステムであっても容易に適用が可能である．

先行研究のいくつかではこの CEM を用いた実時間制御を目標とした研究が行われている．

例えば，動的環境に対応した鏡像降下法である動的鏡像降下法 (Dynamic Mirror Descent) [23] を用いた MPC の研究 [24] では，実際にオートラリーを用いて実時間での走行制御を行っている．この先行研究では CEM を一般化したアルゴリズムを提案しており，時系列に関するパラメータの関係性に着目した改良を行っている．

また，CEM に対して改善を施すことで，実時間制御性能の向上を図った研究 [25] では，いくつかのアドホックな改良を CEM に施した improved CEM を用いて，シミュレーション上でのいくつかのタスクにおいて制御性能の向上が認められている．しかしながらこの先行研究におけるアルゴリズム改善はアドホックな改良であり，理論的な保証をもたない．従って，実験的に性能向上が示されたタスク以外に対しても有効とは限らない．

1.3　研究目的

本研究では，CEM による MPC を用いて実時間制御を行うことを目指す．数理的な改良方針として，具体的には以下のような観点に焦点を当てている．

- 時間あたりの収束効率の向上：CEM はイテレーション毎に解が最適解に収束する．そこで時間による制約下において，最適解への収束効率が上昇するような改良を施す必要がある．
- 実時間制御における安全性能の向上：ロボット等の実機へ適用する場合，危険な状態に遷移することは避ける必要がある．MPC において状態の拘束条

件を考える場合，コスト関数内でペナルティ項を用いるが，これはソフトな拘束条件として扱われるため，実時間制御で導出した解では状態の拘束条件を満たさない可能性もある．そこでこの状態の拘束条件を可能な限り満たすような改良を施す必要がある．

以上から，実時間制御において安全な解に効率よく収束するような数理的な改善を行うことを目標とする．

本研究では，CEM において Kullback-Leibler ダイバージェンス（KL ダイバージェンス）の最適化問題を扱うことに着目する．KL ダイバージェンスは 2 つの確率分布間の乖離度を表す尺度として広く用いられるが，厳密には距離のような尺度とは異なり非対称性がある．この非対称性から KL ダイバージェンスの引数である２つの確率分布を入れ替える事で，最適化問題における更新の挙動が変化する性質に着目して，CEM のアルゴリズムを再導出する．また，これにより導かれた最適化問題を解析すると，CEM のアルゴリズム中でサンプリングした行動により生成されたサンプリング軌道に対する重みとなる関数が負値を取りうるようになる．このサンプリング軌道に対する重みを適切に設計することで，性能の悪いサンプリング軌道から積極的に逃れて性能の良い軌道への収束を優先することが可能となるような手法，Reverse KL CEM(RKL-CEM) を提案する．この RKL-CEM におけるパラメータの更新手法は，現パラメータとの KL ダイバージェンスを正則化項に持つ鏡像降下法を採用することになる．この点に着目し，鏡像降下法より更に収束効率の高いとされる加速鏡像降下法を本問題に合うよう適用した手法，Accelerated Mirror Descent CEM(AMD-CEM) を提案する．

本稿ではこれらの有効性検証として，シミュレーション上での高速道路の実時間の走行制御において，成功率改善とそれによる制御性能の向上を実験的に示す．

1.4 本論文の構成

本論文の構成を述べる．第 2 章では，本研究で扱う MPC，CEM について説明する．また RKL-CEM で用いるために，パラメータ最適化手法である鏡像降下法について説明する．第 3 章では，提案する高効率に安全な行動へと収束する RKL-CEM，AMD-CEM について，その導出とアルゴリズムの説明をする．第 4

章では，提案手法を用いて数値シミュレーションにより有効性の検証を行う．最後に第５章で本論文のまとめと今後の課題について述べる．

第 2 章　準備

2.1　非線形モデル予測制御

本研究では非線形モデル予測制御 (Non-linear MPC; NMPC) を用いて制御する．NMPC は最適制御の一種であり，状態 x と行動 u に基づく非線形モデル $f(x,u)$ と，状態と行動に対する拘束条件 $C(x,u)$ によって定義されるシステムを対象とする．具体的にサンプリング時刻 t では，時刻 t における状態 x_t を起点に，ホライゾン H までの行動系列 $U_t = \{u_t, \ldots, u_{t+H}\}$ を以下の最適化問題 (2.1.1) を通じて最適化する．なお，本研究では状態に対する予測ホライゾンと，制御入力に対する制御ホライゾンは同値であるとする．

$$
\begin{aligned}
U_t^* &= \arg\min_U J(X_t, U_t),\ (X_t = \{x_t, \ldots, x_{t+H+1}\}) \\
s.t.\ J(X_t, U_t) &= L_f(x_{t+H+1}) + \sum_{\tau=t}^{t+H} L(x_\tau, u_\tau) + C_x(X_t) \\
x_{\tau+1} &= f(x_\tau, u_\tau) \\
C_u(u_\tau) &\leq 0
\end{aligned}
\tag{2.1.1}
$$

式 (2.1.1) において，コスト関数 $J(X,U)$ は終端コスト $L_f(x)$ とステージコスト $L(x,u)$，ペナルティ項 $C_x(X)$ に分けられ，各時刻の状態や入力を評価する関数である．この最適行動系列 U_t^* から現時刻の最適行動 u_t^* をシステムに入力する．そして次の時刻でも，ホライゾンの時刻を 1 時刻分後方に移行させて同様の計算を行う．NMPC では，最適化問題をサンプリング時刻毎に解くことで閉ループ形式で制御を行い，ロバスト性の高い制御ができる．また，NMPC では拘束条件を考慮した解を導出可能であるが，一般的には解が存在せず，実行不可能にならないために，最適化変数である行動 U にはハードな制約 $C_u(u_\tau) \leq 0$ を設け，行動 U に依存する状態 X にはペナルティ項 $C_x(X)$ をコスト関数に設けることでソフトな制約をかけ，拘束条件を考慮することが一般的である．

NMPC はこのようにサンプリング時刻毎に H ステップ分の再帰的な計算を行うため，その計算コストの高さが課題とされる．

NMPC は最適化問題を解く手法によって大別されることが多く，例えば，最適な行動系列を求める手法として iLQR/iLQG [13, 14] や C/GMRES 法 [15] 等が存在するが，これらは最適化計算に勾配が必要な一次最適化手法である．そのため微分不可能なシステムであったり，微分可能性が不明のブラックボックスなモデルに対して適用が困難である．そこで本研究では，多様なモデルに対して適用可能な 0 次最適化手法である，CEM を用いたサンプリングベースの NMPC を考える．

2.2 クロスエントロピー法による最適化

2.2.1 クロスエントロピー法

クロスエントロピー法（Cross Entropy Method; CEM） [8] は，重点サンプリング法で導入した確率分布 $p(x;\theta)$ を最適分布 $p(x;\theta^*) = p^*(x)$ に近づける手法である．ここで $x \in \mathbb{R}^n$ で，$H(x)$ は $\mathbb{R}^n \to \mathcal{X}(\mathcal{X} \in \mathbb{R}, \mathcal{X} > 0)$ の正のスカラー関数である．確率分布 $p(x;\psi)$ におけるスカラー関数 $H(x)$ の期待値は重点サンプリング法により以下で与えられる．

$$\ell_{CEM} = \mathbb{E}_{p(x;\psi)}\left[H(x)\right] = \mathbb{E}_{p(x;\theta)}\left[H(x)\frac{p(x;\psi)}{p(x;\theta)}\right] \tag{2.2.1}$$

$p^*(x)$ は，期待値の分散において，$Var_{p^*(x)}[\ell_{CEM}] = 0$ となるよう次式で与えられる．

$$p^* = \frac{H(x)p(x;\psi)}{\ell_{CEM}} \tag{2.2.2}$$

CEM では $p(x;\theta) \to p^*(x)$ を達成するよう，2 つの分布間の乖離度を最小化する．この乖離度として KL ダイバージェンスが採用されており，次式のように展開される．

$$
\begin{aligned}
\theta^* &= \arg\min_{\theta} \mathrm{KL}(p^*(x) \| p(x;\theta)) \\
&= \arg\min_{\theta} \int_{\mathbb{R}^n} p^*(x) \ln p^*(x) dx - \int_{\mathbb{R}^n} p^*(x) \ln p(x;\theta) dx \\
&= \arg\min_{\theta} - \int_{\mathbb{R}^n} p^*(x) \ln p(x;\theta) dx \\
&= \arg\min_{\theta} - \int_{\mathbb{R}^n} \frac{H(x)p(x;\psi)}{\ell_{CEM}} \ln p(x;\theta) dx \\
&= \arg\min_{\theta} - \frac{1}{\ell_{CEM}} \int_{\mathbb{R}^n} H(x)p(x;\psi) \ln p(x;\theta) dx \\
&= \arg\min_{\theta} - \mathbb{E}_{p(x;\psi)}[H(x) \ln p(x;\theta)]
\end{aligned} \tag{2.2.3}
$$

このようにクロスエントロピーの最小化問題に帰着されることが CEM の名前の由来である．最終的な期待値は $p(x;\psi)$ からサンプリング可能であることから，サンプリングベースのモンテカルロ近似により計算可能であり，確率分布のパラメータ θ に関する勾配を 0 とする極大値を求めれば良い．

2.2.2　クロスエントロピー法による再帰的最適化

CEM を用いてコスト関数 $J(x)$ の最小化問題を解くことを考える．このとき，以下の確率が 1 となるような最適確率密度関数 $p^*(x)$ が最適解といえる．

$$
\begin{aligned}
\ell^*_{CEO} &= \mathbb{E}_{p^*(x)} \left[\mathbb{1}_{\{J(x)=\min_x J(x)\}} \right] \\
&= \mathbb{P}\left(J(x) = \min_x J(x) \right) = 1
\end{aligned} \tag{2.2.4}
$$

ここで $\mathbb{1}_{\{\cdot\}}$ は，$\{\cdot\}$ が真なら 1，偽なら 0 を返す指示関数である．なお，ここでは数理上スカラー関数 $H(x)$ は正のため，指示関数が偽の場合に返す値は限りなく 0 に近い値とする．しかしながら最適解をサンプリングする確率が限りなく低い場合，一度のサンプリングから $p^*(x)$ のパラメータ θ^* に最適化することは難しい．そこで以下の期待値を求めることを考える．

$$
\ell_{CEO} = \mathbb{E}_{p(x;\psi)}[\mathbb{1}_{\{J(x)\leq\gamma\}}] \tag{2.2.5}
$$

この期待値は $J(x)$ が q 分位数である閾値 γ 以下を取る確率を表す．これにより，評価値が閾値 γ よりも良いエリートデータに重み 1 を，他のサンプルに重み 0 を与えて更新することで，エリートデータをサンプルしやすくなるように確率分布 $p(x;\theta)$ を更新する．この更新の際，パラメータ θ は以下のような微分法，モンテカルロ法によって近似的に導出される．

$$\begin{aligned}
&\theta = \arg\min_{\theta} -\mathbb{E}_{p(x;\psi)}[\mathbb{1}_{\{J(x)\leq\gamma\}} \ln p(x;\theta)] \\
&-\mathbb{E}_{p(x;\psi)}[\mathbb{1}_{\{J(x)\leq\gamma\}} \nabla_\theta \ln p(x;\theta)] = 0 \\
&-\frac{1}{N}\sum_{i=1}^{N}[\mathbb{1}_{\{J(x_i)\leq\gamma\}} \nabla_\theta \ln p(x;\theta)] = 0
\end{aligned} \tag{2.2.6}$$

ただし，$x_i \sim p(x;\psi)$ であり，N はサンプル数である．

このように緩和した条件で最適化された $p(x;\theta)$ を次のサンプリング用の分布 $p(x;\psi)$ として用い，パラメータ更新を繰り返すことで，徐々に $p(x;\theta)$ を $p^*(x)$ へ近づけることができる．

2.2.3　クロスエントロピー法の制御応用

上記の CEM による最適化を最適制御問題に適用する．具体的には, 方策 $\pi(U_t;\theta)$ を次式の期待値 ℓ_{MPC} を正確に求めるために，最適化することを考える．

$$\ell_{MPC} = \mathbb{E}_{\pi(U_t;\psi)}\left[\mathbb{1}_{\{J(X_t,U_t)\leq\gamma\}}\right] \tag{2.2.7}$$

式 (2.2.3), (2.2.4), (2.2.7) を用いることで，方策分布 $\pi(U_t;\theta)$ のパラメータ θ は次式で更新される．

$$\hat{\theta} = \arg\min_{\theta} -\mathbb{E}_{\pi(U_t;\psi)}[\mathbb{1}_{\{J(X_t,U_t)\leq\gamma\}} \ln \pi(U_t;\theta)] \tag{2.2.8}$$

確率分布 $p(x;\psi), p(x;\theta)$ を方策分布 $\pi(U;\theta_j), \pi(U;\theta_{j+1})$ に置き換えると，CEM を援用したサンプリングベース NMPC のアルゴリズムの全容は Alg. 1 で示される．

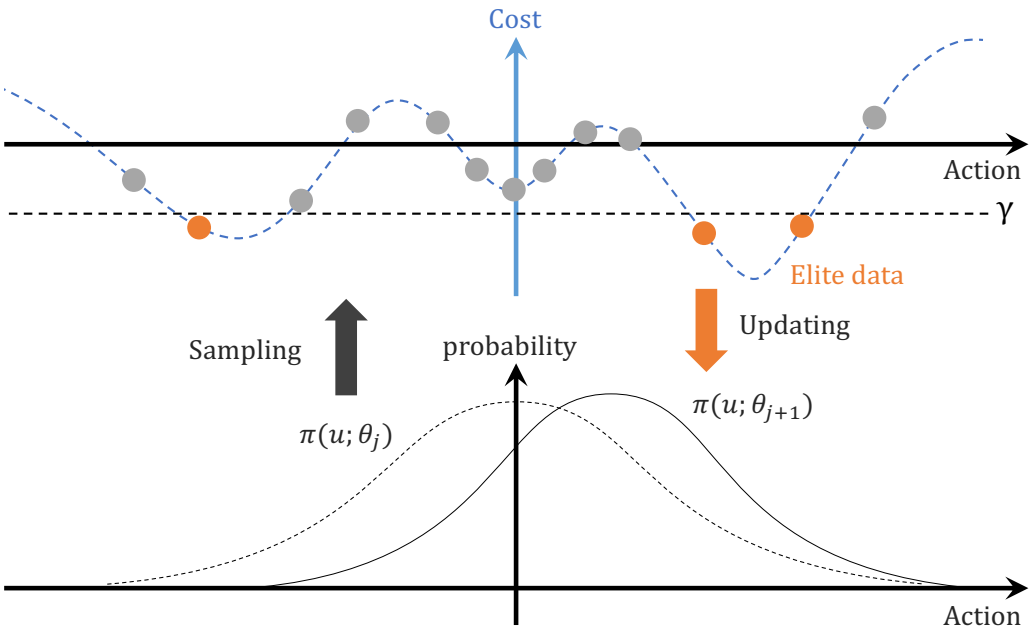


図 2.1　CEM による確率分布の最適化

Algorithm 1 CEM を用いた非線形モデル予測制御

Set $x_t,\ H,\ N,\ \alpha,\ q,\ \theta_0$

Initialize $\hat{x}_0 = x_t$

for $j \in \mathbb{N}$ untill the sampling period comes **do**

$U_i = \{u_0, \ldots, u_H\}_i \sim \pi(U;\theta_j)\ (\theta_j = \{\theta_0, \ldots, \theta_H\}_j,\ i = 1, \ldots, N)$

for $\tau = 0, \ldots, H$ **do**

$\hat{x}^i_{\tau+1} = f(\hat{x}^i_\tau, u^i_\tau)$

end for

$J_i = J(X_i, U_i)\ (X_i = \{\hat{x}_0, \ldots, \hat{x}_{H+1}\}_i)$

Setting γ from J's q quantile

$\hat{\theta} = \arg\min_\theta -\mathbb{E}_{\pi(U;\theta_j)}[\mathbb{1}_{\{J \le \gamma\}} \ln \pi(U;\theta)]$

$\theta_{j+1} = \alpha\theta_j + (1-\alpha)\hat{\theta}$

end for

$U_t = \mathbb{E}_{\pi(U;\theta_{j+1})}[U]$

return u_t from $U_t = \{u_t, \ldots, u_{t+H}\}$

ここで，j はイテレーション回数，N はサンプル数である．なお，本研究では局所解へ陥る可能性を防止するために，パラメータ θ の平滑化 [26](Alg. 1 10 行目) を用いる．この平滑化処理は勾配降下法と対応していることから，平滑化係数 α もステップサイズと対応している．

このようにパラメータ θ_j を繰り返し更新することで，サンプリングによる探索を最適解周辺に狭めていく (図 2.1).

2.3 鏡像降下法

鏡像降下法 (Mirror Descent Algorithm) [27, 28] はよく知られている勾配降下法 (Gradient Descent) を含んだ汎用的な一次最適化アルゴリズムであり，Beck ら [28] が，原始空間と双対空間を用いた降下法としての解釈を与えた一次最適化手法である．鏡像降下法は拘束条件下での最適化を陽に行え，場合によっては効率的に収束させることができる．

以下のような最適化問題を考える．

$$x^* \in \arg\min_{x \in C} f(x) \tag{2.3.1}$$

このとき，鏡像降下法は以下の式で与えられる．

$$x_{k+1} \in \arg\min_{x \in C} \eta \langle \nabla f(x_k), x - x_k \rangle + B_\phi(x, x_k) \tag{2.3.2}$$

$$B_\phi(x, x_k) := \phi(x) - \phi(x_k) - \langle \nabla \phi(x_k), x - x_k \rangle \tag{2.3.3}$$

ここで η はステップサイズを表す．式 (2.3.3) は Bregman ダイバージェンスと呼ばれる 2 つの引数間の乖離度を表す一般化指標である．Bregman ダイバージェンスはポテンシャル関数 ϕ を適切に設計することで二乗ユークリッド距離や KL ダイバージェンスなどの，二つの引数の乖離度を表す関数を表現できる．つまり式 (2.3.2) は，Bregman ダイバージェンスの意味で更新前と更新後の目的変数 x が

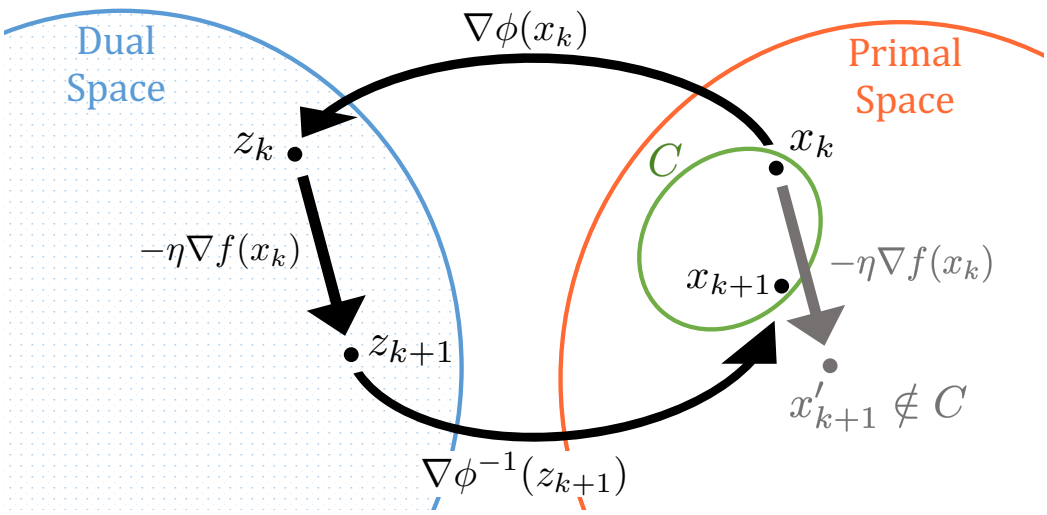


図 2.2　鏡像降下法による双対空間での降下

離れないような正則化を与えながら，目的関数 x の劣勾配を用いて降下法を行う更新則と解釈できる．式 (2.3.2) から，目的変数 x の勾配を求めることで，更新則は以下のような閉形式で与えられる．

$$\begin{aligned} \eta\nabla f(x_k) + \nabla\phi(x_{k+1}) - \nabla\phi(x_k) = 0 \\ x_{k+1} = \nabla\phi^{-1}\left(\nabla\phi(x_k) - \eta\nabla f(x_k)\right) \end{aligned} \tag{2.3.4}$$

Beck ら [28] の解釈では式 (2.3.4) から，適切な Bregman ダイバージェンスのポテンシャル関数の導関数 $\nabla\phi$ を用いて，目的変数 x を原始空間から双対空間へと射影し，降下法を行ってからポテンシャル関数の導関数の逆関数 $\nabla\phi^{-1}$ によって双対空間から原始空間へと射影する更新則だと解釈できる (図 2.2)．この解釈を用いてハードな拘束条件を持つ目的変数 x の更新を行うことを考える．具体的には，目的変数 x を拘束条件のある原始空間 $C(x \in C)$ から，双対変数 $z(z = \nabla\phi(x), z \in \mathbb{R})$ へと射影し，降下法を行ってから原始空間へと射影することでハードな拘束条件を考慮した更新を行うことができる．

第 3 章　提案法

本章では本研究で提案する新たな MPC のための最適化アルゴリズムである，Reverse KL CEM(RKL-CEM) と，RKL-CEM に改良を施した手法，Accelerated Mirror Descent CEM(AMD-CEM) について解説する．これらの手法は実時間のロボット制御において重要な，時間あたりの収束効率の向上および実時間制御における安全性能の向上について，既存手法よりも数理的に妥当な形で改善することを目的としている．

3.1　Reverse KL CEM

既存手法である CEM からの改善点は主に三点である．一つ目は CEM の最適化問題が KL ダイバージェンスの最小化を目的としていることに着目し，改良を行った点である．次に，一つ目の改良により，サンプリング軌道に対する重みの役割を果たす関数が負値を取りうるようになったことから，評価の悪いサンプリング軌道を積極的に避けるように重みの定義を行った点である．最後に，一つ目の改良によって定式化されたパラメータの拘束条件を正則化項として考慮した，鏡像降下法によって更新を行うことで最適化を図った点である．本節ではこれらの改良点に関して詳細に述べる．

3.1.1　カルバック・ライブラ ダイバージェンスの非対称性

KL ダイバージェンスは次式の定義からわかるように 2 つの分布間の乖離度を示すものの，非対称性を有している．

$$KL(p(x)\|q(x)) := \int_x p(x) \ln \frac{p(x)}{q(x)} dx \tag{3.1.1}$$

KL ダイバージェンスを用いて確率分布をターゲットとなる確率分布に近づける場合，式 (3.1.1) 中の確率分布 p と q のどちらがターゲットなのかによって挙動が大きく異なることが知られている (図 3.1)．一般的に p がターゲットの場合を

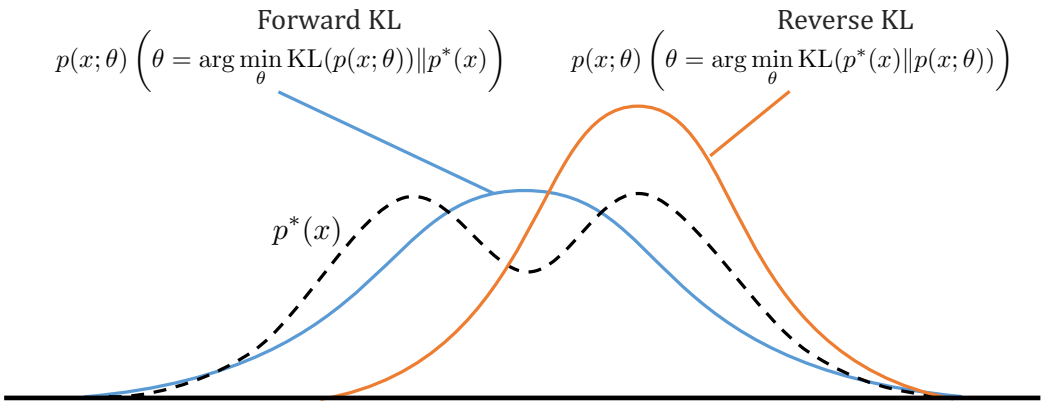


図 3.1　Forward KL と Reverse KL の最適化挙動

Forward KL と呼び，ターゲットの確率分布に対して包括的にフィッティングされる．一方で，q がターゲットの場合は Reverse KL と呼ばれ，ターゲットの確率分布のモード（のいずれか）以外は無視するよう，排他的にフィッティングされる．式 (2.2.3) からわかるように CEM は Forward KL を用いており，その挙動は探索を重視している一方で効率の悪化に繋がることが懸念される．

そこで，我々は CEM の最小化問題で Reverse KL を用いることで，不要なサンプリングを効率良く減らすことを狙う．式 (2.2.3) を参考に，KL ダイバージェンスの項を Reverse KL に変更して解くと以下のようになる．

$$
\begin{aligned}
\hat{\theta} &= \arg\min_\theta \mathrm{KL}(p(x;\theta)\|p^*(x)) \\
&= \arg\min_\theta \int_x p(x;\theta)\ln p(x;\theta)dx - \int_x p(x;\theta)\ln\frac{H(x)p(x;\psi)}{\ell}dx \\
&= \arg\min_\theta \int_x p(x;\theta)\ln p(x;\theta)dx - \int_x p(x;\theta)\ln p(x;\psi)dx - \int_x p(x;\theta)\ln H(x)dx \\
&= \arg\min_\theta \mathrm{KL}(p(x;\theta)\|p(x;\psi)) - \mathbb{E}_{p(x;\psi)}\left[\frac{p(x;\theta)}{p(x;\psi)}\ln H(x)\right] \qquad (3.1.2)
\end{aligned}
$$

ここで，第一項の KL ダイバージェンスは θ の更新を制限する正則化項とみなせる．この場合，上記の最小化問題は次式のような拘束付き最小化問題にラグラン

ジュの未定乗数法を介して変換できる．

$$\hat{\theta} = \arg\min_{\theta} -\mathbb{E}_{p(x;\psi)}\left[\frac{p(x;\theta)}{p(x;\psi)}\ln H(x)\right] \tag{3.1.3}$$

$$\text{s.t.KL}(p(x;\theta)\|p(x;\psi)) \leq \delta \tag{3.1.4}$$

ここで，$\delta \geq 0$ は KL ダイバージェンスの所望の上界を示す．

3.1.2 失敗方策を考慮した方策分布の更新

上記の問題は後述する鏡像降下法で解くことが可能であるが，ひとまず最小化したい目的関数の θ に関する勾配を計算すると，次式を得る．

$$-\nabla_\theta \mathbb{E}_{p(x;\psi)}\left[\frac{p(x;\theta)}{p(x;\psi)}\ln H(x)\right] = -\mathbb{E}_{p(x;\psi)}\left[\ln H(x)\nabla_\theta \ln p(x;\theta)\right] \tag{3.1.5}$$

これは式 (2.2.3) の勾配を計算した場合と比較すると，差分は $H(x) \to \ln H(x)$ となった点のみであり，計算コストが変わらないことは明白である．

スカラー関数 $H(x)$ は正で定義されていたため，従来の CEM では対数尤度への正の重みとして解釈された．しかし，Reverse KL の最小化を通じて得られた項には $\ln H(x)$ が含まれており，$0 < H(x) < 1$ のときに負値を，$H(x) = 1$ のときに厳密に 0 を，$H(x) > 1$ のときに正値を取ることができる．この特徴を活用すれば，MPC におけるコスト関数 $J(X,U)$ が q_1 分位数である閾値 γ_1 以下である価値の高いサンプルであるエリートデータを優先しながら，$J(X,U)$ が q_2 分位数である閾値 γ_2 以上の不要なサンプル，ドロップデータを排他するように θ を更新することが可能となる (図 3.2)．これらの特徴を踏まえ，提案する RKL-CEM ではスカラー関数を次式で定義する．

$$\ln H(x) = \mathbb{1}_{\{J(X,U)\leq\gamma_1\}} - \mathbb{1}_{\{J(X,U)\geq\gamma_2\}} \tag{3.1.6}$$

本稿では，CEM と可能な限り計算コストを同様に抑えるため，ドロップデータに対する重みを -1 とする．

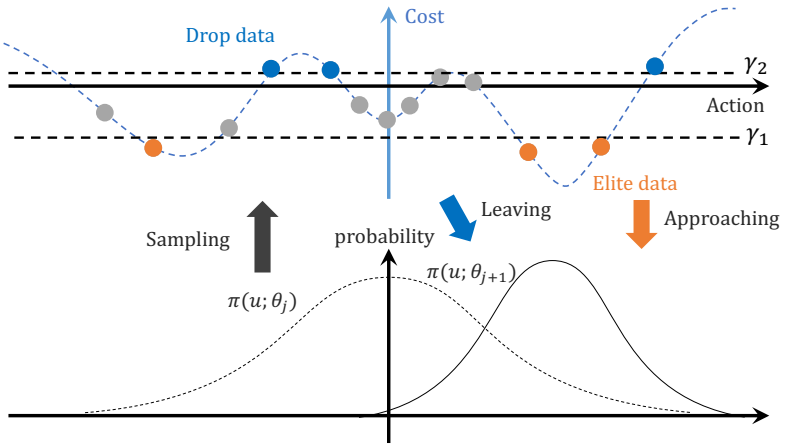


図 3.2　RKL-CEM による確率分布の最適化

3.1.3　鏡像降下法によるパラメータの更新

式 (3.1.3) で与えられた拘束付き最小化問題を解くために，鏡像降下法 [24,28] によりパラメータを更新する．KL ダイバージェンスによる拘束条件は確率分布としてのものであるが，実際に更新されるのは確率分布のパラメータであるため，パラメータごとの拘束条件に分離することで，鏡像降下法の Bregman ダイバージェンスに反映する必要がある．

ここで本研究では，簡単のため方策分布を正規分布として考える．他の確率分布を扱う場合でも，KL ダイバージェンスの拘束条件をパラメータごとの閉形式で与えられるような式変形が可能な確率分布を用いる必要があることに留意したい．方策分布がパラメータに平均 μ，標準偏差 σ を持つ正規分布であるとすると，拘束条件は以下のように展開できる．

$$KL[p(x;\theta)\|p(x;\psi)] = \frac{1}{2}\left[\ln\frac{\sigma_\psi^2}{\sigma_\theta^2} + \frac{\sigma_\theta^2}{\sigma_\psi^2} + \frac{1}{\sigma_\psi^2}\left(\mu_\psi - \mu_\theta\right)^2 - d\right] \leq \delta \qquad (3.1.7)$$

ここで式 (3.1.7) 中の確率分布 $p(x;\psi)$ のパラメータである平均，標準偏差は μ_ψ, σ_ψ，確率分布 $p(x;\theta)$ のパラメータの平均，標準偏差は $\mu_\theta, \sigma_\theta$ であり，d はパラメータの次元数である．δ を各パラメータごとの拘束 $\delta_\mu, \delta_\sigma$ に分けて考えると以下の通りとなる．

$$\begin{cases} \frac{1}{\sigma_\psi^2}(\mu_\psi - \mu_\theta)^2 \leq \delta_\mu \\ \frac{\sigma_\theta^2}{\sigma_\psi^2} - \ln\frac{\sigma_\theta^2}{\sigma_\psi^2} \leq \delta_\sigma \end{cases} \qquad (3.1.8)$$

鏡像降下法における Bregman ダイバージェンスが式 (3.1.8) となるようにポテンシャル関数 ϕ を設定することで，鏡像降下の正則化項として拘束条件を考慮する．そこで，ポテンシャル関数の導関数 $\nabla\phi$ を以下のように定義する．

$$\begin{cases} \nabla\phi_\mu(\mu) = \frac{2\mu}{\sigma_\psi^2} \\ \nabla\phi_\sigma(\sigma) = 2\left(\frac{\sigma}{\sigma_\psi^2} - \frac{1}{\sigma}\right) \end{cases} \qquad (3.1.9)$$

このポテンシャル関数の導関数 $\nabla\phi$ を元に，Bregman ダイバージェンスを構成すると以下の通りとなる．

$$\begin{cases} B_{\phi_\mu}(\mu, \mu_\psi) = \frac{1}{\sigma_\psi^2}(\mu_\psi - \mu)^2 \\ B_{\phi_\sigma}(\sigma, \sigma_\psi) = \frac{\sigma^2}{\sigma_\psi^2} - \ln\frac{\sigma^2}{\sigma_\psi^2} - 1 \end{cases} \qquad (3.1.10)$$

式 (3.1.10) のパラメータ σ の Bregman ダイバージェンスの定数項を除けば，これは式 (3.1.8) の左項と一致する．よって式 (3.1.9) のポテンシャル関数の導関数を使った Bregman ダイバージェンスを鏡像降下法に用いることで，式 (3.1.3) を解きパラメータを更新できる．また，このポテンシャル関数の導関数の逆関数 $\nabla\phi^{-1}$ に以下のような関数を用いることで，標準偏差 σ がハードな拘束条件 ($\sigma > 0$) を満たしながら更新する．

$$\begin{cases} \nabla\phi_{\mu}^{-1}(\mu) = \frac{\sigma_{\psi}^2\mu}{2} \\ \nabla\phi_{\sigma}^{-1}(\sigma) = \frac{1}{4}\left(\sigma_{\psi}^2\sigma + \sigma_{\psi}\sqrt{\sigma_{\psi}^2\sigma^2 + 16}\right) \end{cases} \tag{3.1.11}$$

これらを踏まえて，確率分布 $p(x;\psi), p(x;\theta)$ を方策分布 $\pi(U;\theta_j), \pi(U;\theta_{j+1})$ に置き換え，RKL-CEM を構築する．RKL-CEM は以下の Alg. 2 により現在の最適行動 u_t を導出する．

Algorithm 2 RKL-CEM を用いた非線形モデル予測制御

Set $x_t,\ H,\ N,\ \eta,\ q_1,\ q_2,\ \theta_0,\ \nabla\phi_\theta,\ \nabla\phi_\theta^{-1}$
Initialize $\hat{x}_0 = x_t$
for $j \in \mathbb{N}$ untill the sampling period comes **do**
  $U_i = \{u_0, \ldots, u_H\}_i \sim \pi(U;\theta_j)$ $(\theta_j = \{\theta_0, \ldots, \theta_H\}_j,\ i = 1, \ldots, N)$
  for $\tau = 0, \ldots, H$ **do**
    $\hat{x}_{\tau+1}^i = f(\hat{x}_\tau^i, u_\tau^i)$
  end for
  $J_i = J(X_i, U_i)$ $(X_i = \{\hat{x}_0, \ldots, \hat{x}_{H+1}\}_i)$
  Setting γ_1, γ_2 from J's q_1, q_2 quantile
  $\nabla_\theta L_j = -\mathbb{E}_{\pi(U;\theta_j)}[\mathbb{1}_{\{J\le\gamma_1\}}\nabla_\theta \ln\pi(U;\theta) - \mathbb{1}_{\{J\ge\gamma_2\}}\nabla_\theta \ln\pi(U;\theta)]$
  $\theta_{j+1} = \nabla\phi_\theta^{-1}\left(\nabla\phi_\theta(\theta_j) - \eta\nabla_\theta L_j\right)$
end for
$U_t = \mathbb{E}_{\pi(U;\theta_{j+1})}[U]$
return u_t from $U_t = \{u_t, \ldots, u_{t+H}\}$

3.2 Accelerated Mirror Descent CEM

本研究ではこれまでアルゴリズム改善の際に，１度の更新による最適解への収束効率を高めることを重視して RKL-CEM を開発し，その過程で RKL-CEM では鏡像降下法によるパラメータの更新が導出された．そこで，この鏡像降下法の１度の更新による収束効率を高めるために，Nesterov の加速法 [29] を再構成するような改良を施した加速鏡像降下法 [30] に着目した．本節では，この加速鏡像降下法を実

時間制御に向けた MPC に適した形となるよう調整を施した，AMD-CEM について解説する．

基本的な加速鏡像降下法は以下のアルゴリズムに従って，パラメータ x の更新を行う．

Algorithm 3 Accelerated Mirror Discent with Adaptive Restart

1: Set $x_0,\ s,\ \gamma\ (\geq 1),\ r\ (\geq 3)$
2: Initialize $l = 0,\ \tilde{x}_0 = x_0,\ \tilde{z}_0 = x_0,\ \left(\text{ or } z_0 \in (\nabla\phi)^{-1}(x_0)\right)$
3: **for** $k \in \mathbb{N}$ **do**
4: $\quad x_{k+1} = \lambda_l \tilde{z}_k + (1-\lambda_l)\,\tilde{x}_k$ with $\lambda_l = \frac{r}{r+l}$
5: $\quad \tilde{z}_{k+1} = \arg\min_{\tilde{z}\in\mathcal{X}} \frac{ks}{r}\left\langle \nabla f(x_{k+1}), \tilde{z}\right\rangle + B_\phi(\tilde{z}, \tilde{z}_k)$
6: $\quad \tilde{x}_{k+1} = \arg\min_{\tilde{x}\in\mathcal{X}} \gamma s \left\langle \nabla f(x_{k+1}), \tilde{x}\right\rangle + R(\tilde{x}, x_{k+1})$
7: $\quad l \leftarrow l+1$
8: $\quad$ **if** Restart Condition **then**
9: $\quad\quad l \leftarrow 0,\ \tilde{z}_{k+1} = \tilde{x}_{k+1}$
10: $\quad$ **end if**
11: **end for**

本研究では, Alg.3 の 5 行目の変数 $\tilde{z}$ の更新では, RKL-CEM と同様に式 (3.1.10) の Bregman ダイバージェンスを用いた鏡像降下法によりパラメータの更新を行う．一方，Alg.3 の 6 行目の変数 $\tilde{x}$ の更新では，射影勾配降下法 (Projected Gradient Descent) [31] により，パラメータのハードな拘束条件のみを考慮した勾配降下法による更新を行う．本稿ではこれら各更新方法について，便宜上前者を鏡像降下部，後者を勾配降下部と呼称する．この双対空間下での鏡像降下部による更新値と勾配降下部による更新値の線形加重和により，Nesterov の加速法を再構成し，収束効率を向上させる．この Alg.3 は，従来の鏡像降下法の収束率 $\mathcal{O}(1/k)$ から，文献 [30] において収束率 $\mathcal{O}(1/k^2)$ で収束することが保証されている．

また，加速鏡像降下法における，Adaptive Restart は，任意の条件に応じて鏡像降下部と勾配降下部の重み調整を担う変数 l のカウントを初期化することで，収束効率を高めるヒューリスティックな改善手法である．Restart Condition には，パラメータの変化量に着目した Speed Restart$\left(\left\|x^{(k+1)} - x^{(k)}\right\| < \left\|x^{(k)} - x^{(k-1)}\right\|\right)$ と，Gradient Restart$\left(\left\langle x^{(k+1)} - x^{(k)}, \nabla f\left(x^{(k)}\right)\right\rangle > 0\right)$ が提案されており，どちらも性能の向上が報告されている [32].

これらを踏まえて，パラメータ x_j を方策分布パラメータ θ_j に，ステップサイズ s を η 置き換え，鏡像降下部と勾配降下部の更新則を閉形式に式変形することで,

AMD-CEM を構築する．AMD-CEM では以下のアルゴリズムにより現在の最適行動 u_t を導出する．

Algorithm 4 AMD-CEM を用いた非線形モデル予測制御

Set $x_t,\ H,\ N,\ \eta,\ q_1,\ q_2,\ \theta_0,\ r(\geq 3),\ \gamma(\geq 1),\ \nabla\phi_\theta^z,\ \nabla{\phi_\theta^z}^{-1},\ \nabla\phi_\theta^R,\ \nabla{\phi_\theta^R}^{-1}$

Initialize $\hat{x}_0 = x_t,\ \theta_0^z, \theta_0^R = \theta_0,\ l = 0$

$\theta_1 = \lambda_l\theta_0^z + (1-\lambda_l)\theta_0^R,\ \lambda_l = \frac{r}{r+l}$

for $j \in \mathbb{N}$ untill the sampling period comes **do**

  $U_i = \{u_0, \ldots, u_H\}_i \sim \pi(U;\theta_j)\ (\theta_j = \{\theta_0, \ldots, \theta_H\}_j,\ i = 1, \ldots, N)$

  for $\tau = 0, \ldots, H$ **do**

    $\hat{x}_{\tau+1}^i = f(\hat{x}_\tau^i, u_\tau^i)$

  end for

  $J_i = J(X_i, U_i)\ (X_i = \{\hat{x}_0, \ldots, \hat{x}_{H+1}\}_i)$

  Setting γ_1, γ_2 from J's q_1, q_2 quantile

  $\nabla_\theta L_j = -\mathbb{E}_{\pi(U;\theta_j)}[\mathbb{1}_{\{J\leq\gamma_1\}}\nabla_\theta \ln\pi(U;\theta) - \mathbb{1}_{\{J\geq\gamma_2\}}\nabla_\theta \ln\pi(U;\theta)]$

  $\theta_{j+1}^z = {\nabla\phi_\theta^z}^{-1}\left(\nabla\phi_\theta^z(\theta_j^z) - \frac{j\eta}{r}\nabla_\theta L_j\right)$

  $\theta_{j+1}^R = {\nabla\phi_\theta^R}^{-1}\left(\nabla\phi_\theta^R(\theta_j^R) - \gamma\eta\nabla_\theta L_j\right)$

  $\theta_{j+1} = \lambda_l\theta_{j+1}^z + (1-\lambda_l)\theta_{j+1}^R,\ \lambda_l = \frac{r}{r+l}$

  if Restart Condition **then**

    $l = 0,\ \theta_{j+1}^z = \theta_{j+1}^R$

  end if

end for

$U_t = \mathbb{E}_{\pi(U;\theta_{j+1})}[U]$

return u_t from $U_t = \{u_t, \ldots, u_{t+H}\}$

第 4 章　数値シミュレーション

提案法の有効性を検証するため，失敗する危険性と高い報酬のトレードオフが存在するタスクを用意し，提案法と既存手法との比較を行った．

4.1　シミュレーション環境

本章では OpenAI Gym [33] の環境を基に作成された Highway-env [34] における Highway-v0（図 4.1）と呼称される環境を用い，実時間制御に近い計算時間制約を設けた上での制御性能評価を行った．Highway-v0 は二輪モデル [35] に従って動作する自車を制御し，Intelligent Driver Model(IDM) [36] と Minimizing Overall Braking Induced by Lane change(MOBIL) モデル [37] に従って行動する他車に，衝突しないよう高速道路を走行するタスクである．

本実験では自車の目標速度を 30 m/s（他車は 20–25 m/s）とし，500 step（シミュレーション内部時間の 50 s）の間で衝突しないように走行可能範囲内の 3 車線で回避もしくは減速を行うことを目標とする．観測値は自車と周囲の他車 3 台の座標 (x, y)，速度 (v_x, v_y)，機首方位の 20 次元，入力値は自車の加速度と操舵角の 2 次元である．

留意事項として，本実験において，MPC 内部のモデルは計算コスト削減のため，観測値から他車の入力値が 0 であるものと仮定して，予測を行った．この仮定では，他者が自車に合わせた速度で走行したり，自車の進路妨害になるような車線変更は行わない，といった衝突を回避するような行動を想定しないため，悪意のある走行例を除けば一般的には最悪な場合を想定した予測であると考えられる．

また，MPC のコスト関数では，各車線の中央を目標軌道とし，目標速度に近づくほど小さいコストを与えた．また走行可能範囲外の走行，あるいは他車との衝突，18 m/s 以下の速度に達した際には失敗とみなして大きなコストを与えるよう設計した．具体的に各サンプリング時刻のコストは以下のように定義した．

$$J_j = -0.5 \times L_{lane} - 0.5 \times L_{velocity} + 10 \times C_{penalty} \tag{4.1.1}$$
$$L_{lane} = 1 - 2\frac{\|y - y_{lane}\|}{l_{lane}},\ (L_{lane} \in (-\infty, 1])$$
$$L_{velocity} = \frac{1}{30}\min\{30, \|v_x + v_y\|\},\ (L_{velocity} \in [0, 1])$$
$$C_{penalty} = \mathbb{1}_{\{\text{ 衝突判定 }\}} + \mathbb{1}_{\{\text{ 車線外走行判定 }\}} + \mathbb{1}_{\{18 \geq \|v_x + v_y\|\}}$$

ここで，y_{lane} は走行中のレーン中央の y 座標 [m]，l_{lane} はレーン幅 [m] である．

また，本稿では OS が Ubuntu 20.04 LTS の Intel Core i9-9900K(3.60GHz), GeForce RTX 2060 SUPER(1470MHz) を搭載するコンピュータを用いた．

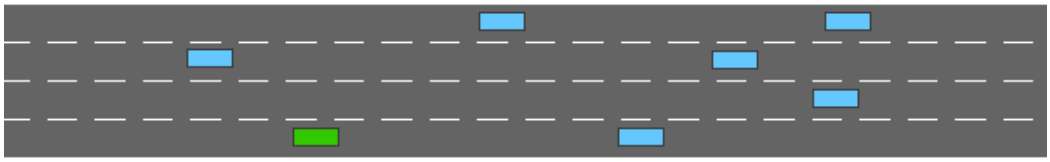


図 4.1　シミュレーション環境: Highway-v0

4.2　シミュレーション条件

実験に用いたパラメータは表 4.1 に示す．また，各 CEM 固有のパラメータは表 4.2，4.3，4.4 の通りである．なお，エリートデータ数 N_{elite}，ドロップデータ数 N_{drop} は，エリートデータ，ドロップデータとして用いられるサンプル数であり，各 CEM における q 分位数と対応したパラメータである．

表 4.3，4.4 におけるステップサイズ η' について，CEM ではパラメータ更新の際，モンテカルロ法を用いて近似計算を行うが，その際にエリートデータ（及びドロップデータ）以外を無視した総和を用いて計算を行う．そのため，近似計算により導出された目的関数の勾配が極めて小さくなってしまう．本稿では勾配の小さくなる要因を相殺して更新を行うために，鏡像降下法におけるステップサイズ η の代わりに，$\frac{\eta' N}{N_{elite}}$ を用いた．

本実験では，AMD-CEM における鏡像降下部のステップサイズ $\frac{j\eta}{r}$ のイテレーション数 j を表 4.4 の通り，4 からカウントを始めるものとした．これは実時間制

御下では平均イテレーションが少ないために，鏡像降下部における更新量が少なく，値の収束が極めて遅かったためである．また，本検証では Adaptive Restart は用いない．

表 4.1　シミュレーションパラメータ

パラメータ名	値
エピソード数	1000
制御周波数	$\{10,\ 20,\ 30\}$
ホライゾン H	15
サンプル数 N	10000
エリートデータ数 N_{elite}	100
ドロップデータ数 N_{drop}	$\{0,\ 25,\ 50\}$
初期方策分布パラメータ $\theta_0 = [\mu_0, \sigma_0]$	$[0,\ 1]$

表 4.2　CEM のパラメータ

パラメータ名	値
平滑化係数 α	0.4

表 4.3　RKL-CEM のパラメータ

パラメータ名	値/関数
ステップサイズ η'	0.6
射影関数 $\nabla\phi_\mu(\mu)$	$\frac{2\mu}{\sigma_\psi^2}$
射影関数 $\nabla\phi_\sigma(\sigma)$	$2\left(\frac{\sigma}{\sigma_\psi^2} - \frac{1}{\sigma}\right)$
射影関数の逆関数 $\nabla\phi_\mu^{-1}(\mu)$	$\frac{\sigma_\psi^2\mu}{2}$
射影関数の逆関数 $\nabla\phi_\sigma^{-1}(\sigma)$	$\frac{1}{4}\left(\sigma_\psi^2\sigma + \sigma_\psi\sqrt{\sigma_\psi^2\sigma^2 + 16}\right)$

表 4.4　AMD-CEM のパラメータ

パラメータ名	値/関数
ステップサイズ η'	0.8
初期イテレーション	4
r	3
γ	1
射影関数 $\nabla\phi_{\mu}^{z}(\mu)$	$\frac{2\mu}{\sigma_{\psi}^{2}}$
射影関数 $\nabla\phi_{\sigma}^{z}(\sigma)$	$2\left(\frac{\sigma}{\sigma_{\psi}^{2}}-\frac{1}{\sigma}\right)$
射影関数の逆関数 $\nabla{\phi_{\mu}^{z}}^{-1}(\mu)$	$\frac{\sigma_{\psi}^{2}\mu}{2}$
射影関数の逆関数 $\nabla{\phi_{\sigma}^{z}}^{-1}(\sigma)$	$\frac{1}{4}\left(\sigma_{\psi}^{2}\sigma+\sigma_{\psi}\sqrt{\sigma_{\psi}^{2}\sigma^{2}+16}\right)$
射影関数 $\nabla\phi_{\mu}^{R}(\mu)$	μ
射影関数 $\nabla\phi_{\sigma}^{R}(\sigma)$	σ
射影関数の逆関数 $\nabla{\phi_{\mu}^{R}}^{-1}(\mu)$	μ
射影関数の逆関数 $\nabla{\phi_{\sigma}^{R}}^{-1}(\sigma)$	$\max\{10^{-5},\ \sigma\}$

4.3　シミュレーション結果

既存手法である CEM と，提案手法である RKL-CEM，AMD-CEM について制御性能の比較を行った．制御性能を測る指標として，以下の 3 点に着目した．

- 成功率：走行可能範囲外の走行，あるいは他車との衝突した場合を失敗とした場合の各エピソードの成功率である．
- MPC Score：各 step の状態に対するコスト関数値の反数の平均値である．なお，エピソード途中で失敗した場合，以降エピソードの終了 (500 step 目) まで各 step の Score はコスト関数内で用いたペナルティ値と同じ-10 とした．
- 平均走行速度：自車の走行速度の平均値である．

また，実験結果における各手法の，step 毎のイテレーション数の平均を表 4.5 に示す．制御周波数ごとに着目すると，各 CEM はどれも平均イテレーションはほぼ変化が無く，計算コストがほぼ同等であることがわかる．

表 4.5　各手法の平均イテレーション

手法	平均イテレーション
10Hz	
CEM	10.0
RKL-CEM(N_{drop}=0)	9.99
RKL-CEM(N_{drop}=50)	10.0
AMD-CEM(N_{drop}=0)	9.51
AMD-CEM(N_{drop}=50)	9.74
20Hz	
CEM	5.00
RKL-CEM(N_{drop}=0)	4.99
RKL-CEM(N_{drop}=25)	5.00
RKL-CEM(N_{drop}=50)	4.95
AMD-CEM(N_{drop}=0)	4.95
AMD-CEM(N_{drop}=25)	4.61
AMD-CEM(N_{drop}=50)	4.44
30Hz	
CEM	3.00
RKL-CEM(N_{drop}=0)	3.00
RKL-CEM(N_{drop}=50)	3.00
AMD-CEM(N_{drop}=0)	3.00
AMD-CEM(N_{drop}=50)	3.00

4.3.1 各手法の制御性能について

ここでは制御周波数 20 Hz における制御性能に着目する．比較結果を 図 4.2 に示す．なお，手法名横の括弧内の#drop はドロップデータ数 N_{drop} を表す．

まず，MPC スコア（ 図4.2(a)）と，成功率（ 図4.2(b)）に着目する．CEM と提案手法を比較すると，提案手法の全ての MPC スコア，成功率が CEM よりも高いことから，本タスクにおいて提案手法では制御性能が向上していることがわかる．また，RKL-CEM，AMD-CEM のドロップデータ数 N_{drop} を変化させた場合の結果から，ドロップデータ数 N_{drop} を増やすことで成功率が向上していることから，サンプリングの早期段階から失敗を回避する方策を獲得できたことが示唆される．

次に，平均走行速度（ 図4.2(c)）に着目すると，ドロップデータ数 N_{drop} を増やすことで平均速度が減少傾向にある．これは，速度を上げることによる衝突の危険性の回避を優先し，安全性の高い準最適解に収束していると考えられる．

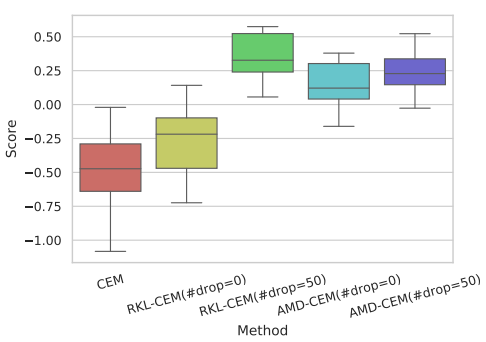
4.3.2 制御周波数を変化させた場合の制御性能の推移

続いて制御周波数を変化させた場合の制御性能の推移に着目する．比較結果を 図 4.3 に示す．

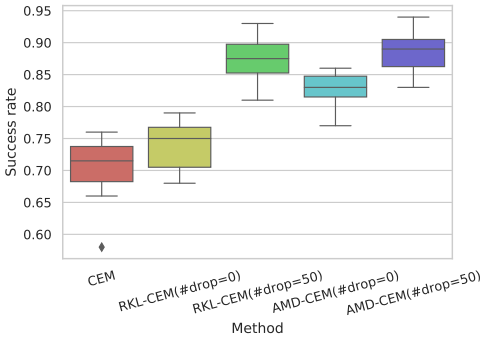
制御周波数を変化させると，表 4.5 のように制御周波数ごとに平均イテレーションが大きく変わってしまう．このような平均イテレーションの差は，解の質に影響を及ぼし，制御性能にも大きく関わる．

まず，MPC スコア（ 図4.3(a)）の結果から，制御周波数が小さくなった場合においても CEM に比べて提案手法は高い制御性能が保たれていることが確認できる．

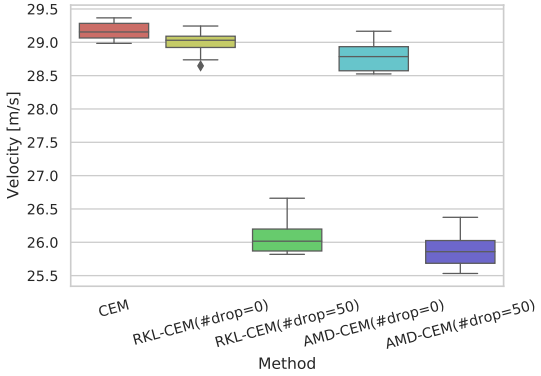
また，今度は MPC スコア（ 図4.2(a)）と，成功率（ 図4.2(b)）のドロップデータを考慮した場合に着目する．ドロップデータを考慮した手法は，考慮しない手法と比べ，制御周波数が小さくなると大きくその性能が下がっていることがわかる．これは最も初期段階のサンプリングで得たドロップデータの情報は分散が大きく，更新方向が一定ではないことから収束の悪化を招いている可能性があると考えられる．



(a) MPC スコア



(b) 成功率



(c) 平均走行速度

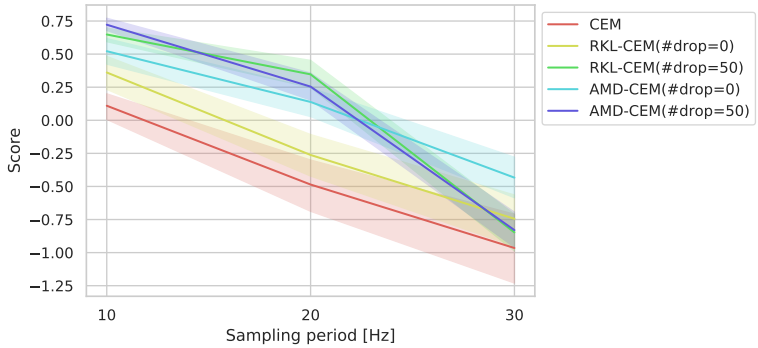
図 4.2　シミュレーション結果（制御周波数 20 Hz）

4.3.3　ドロップデータ数を変化させた場合の制御性能の推移

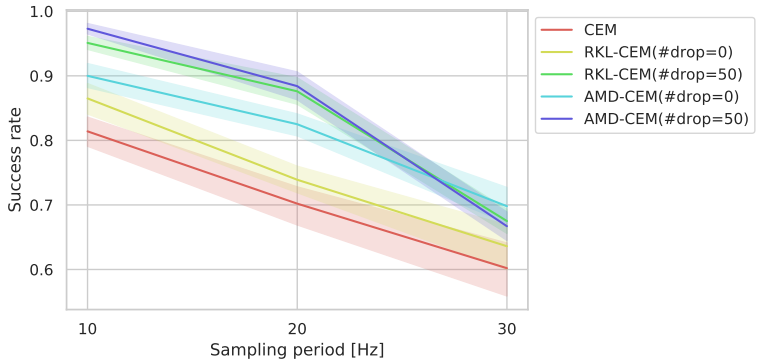
最後に制御周波数 20 Hz において，提案手法のドロップデータ数を変化させた場合の制御性能の推移に着目する．比較結果を 図 4.4 に示す．

先述のように， 図 4.4(a)， 4.4(b) から，ドロップデータ数 N_{drop} を増やすことで失敗を早期に回避するようになり成功率を上昇させることが確認できる．加えて AMD-CEM においては，ドロップデータを考慮しない場合でも制御性能が高いことが示されている．また，RKL-CEM と AMD-CEM を比較すると，AMD-CEM よりも RKL-CEM の方がドロップデータ数 N_{drop} が MPC スコア (図 4.4(a)) に及ぼす影響が大きいことがわかる．この原因について，先述の制御周波数を変化

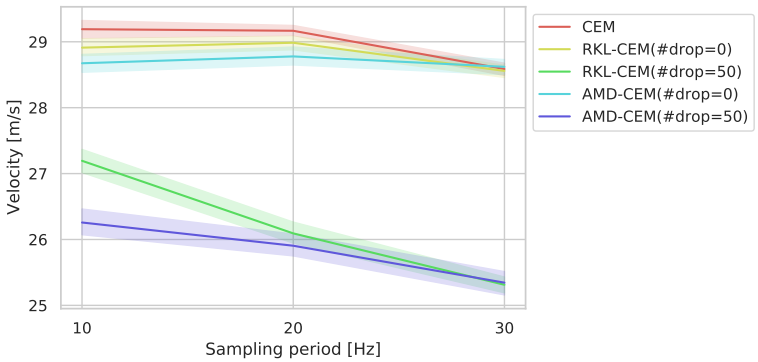
させた場合の制御性能比較において，ドロップデータを考慮した場合，制御周波数が小さくなるにつれて，考慮しない場合よりも制御性能の低下が著しいことや，表 4.5 の平均イテレーションから，AMD-CEM のドロップデータを考慮した場合の平均イテレーションの低さが制御性能に現れている可能性がある．これは，制御周波数が 30 Hz の MPC スコア（ 図4.3(a)）を見ると，RKL-CEM と AMD-CEM のドロップデータを考慮した場合の結果がほぼ等しく，表 4.5 から平均イテレーションがほぼ同じであることからも同様のことが推察される．他の原因として，AMD-CEM ではその収束性能が高いためにドロップデータによる恩恵が薄い可能性も考えられる．



(a) MPC スコア

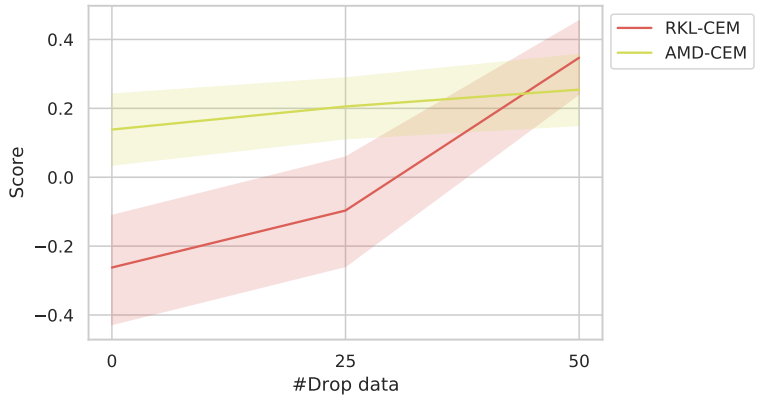


(b) 成功率

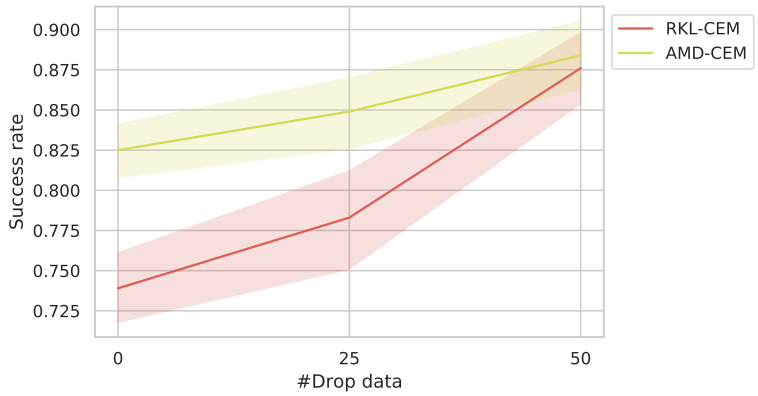


(c) 平均走行速度

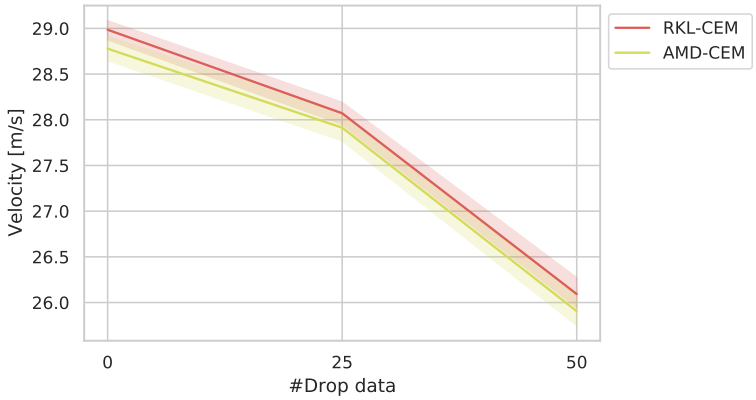
図 4.3　制御周波数別シミュレーション結果



(a) MPC スコア



(b) 成功率



(c) 平均走行速度

図 4.4　ドロップデータ数別シミュレーション結果

第 5 章　結論

5.1　まとめ

本稿では，最適化手法として CEM を用いたサンプリングベース MPC に対し，実時間制御に向けてアルゴリズムを導出から改良した RKL-CEM，AMD-CEM を提案した．既存手法の CEM では，Forward KL の最小化を行うことで，コスト関数を最小化するような，方策分布にフィッティングを行っていた．しかし，実時間制御下では複数のモードに対して包括的にフィッティングしてしまうために分散が大きくなってしまい，最適解周辺の探索効率の悪化が懸念された．そこで Reverse KL の最小化を行うことで，単一のモードに対して早急にフィッティングさせ，最適解周辺への探索を促し，探索効率の向上を図った．加えて，このような Reverse KL の最適化を通じて，負値を取りうる重みと鏡像降下法を導入した RKL-CEM を提案した．また，更なる収束効率の向上のため，パラメータの最適化手法に加速鏡像降下法を適用した AMD-CEM を提案した．

これらは数値シミュレーションより，実時間制御を想定した制御周期下において，CEM よりも総合的な性能が向上することを確認した．また，この制御性能の向上が，いくつかの制御周期下においても同様であることを確認した．さらに，ドロップデータの考慮により，成功率が上昇し，総合的な性能が向上することを確認した．

5.2　今後の課題

今後の課題は大きく二つ挙げられる．

一つ目の課題は，明確な失敗が無いようなタスクにおける検証の必要性である．本稿では提案手法の有効性の検証方法として高速道路の走行制御を用いたが，これは明確な失敗と高い報酬のトレードオフがあるようなタスクである．このような環境下において，提案手法の特徴を十全に活用できたため，パラメータ次第で提案手法はどちらの場合も CEM より，総合的な性能が大きく向上したと考えられる．そのため，明確な失敗が無いようなタスクにおいても制御性能の向上が見込まれるかどうか確認する必要があると考えられる．

二つ目の課題は，次元数の多いタスクにおける検証の必要性である．本稿の数値シミュレーションで用いたタスクは行動の次元が高々 2 次元の最適化対象が低次元なタスクであったため，最適化対象の次元数の多い実機や別のタスクにおいても有効な成果が得られるか確かめる必要があると考えられる．

謝辞

研究室生活を送るにあたり，制御工学の基礎知識，ラボミーティングや発表練習での研究や発表資料に関するご意見，研究に取り組む姿勢など終始熱心なご指導を賜りました知能システム制御研究室の杉本謙二教授に心より御礼申し上げます．

ご多用の中，副指導教員を引き受けていただき，発表会でコメントやご助言をいただきました数理情報学研究室の池田和司教授に深く御礼申し上げます．

本研究の遂行にあたり，小林泰介助教には本研究の基礎となる理論や，研究の方向性など本研究に必要な知識を多くご教授いただきました．また，論文執筆の方法や発表資料の作り方，発表の仕方など多くのご指導をしていただきました．心より御礼申し上げます．

ラボミーティングや発表練習で親身になって研究や発表資料に関するご意見をしていただき，丁寧なご指導をしていただきました花田研太助教に深く御礼申し上げます．

PBL 科目で研究に関するご意見，ご指導をいただきましたロボットラーニング研究室の松原崇充特任准教授に深く御礼申し上げます．

さまざまな事務手続きを丁寧かつ迅速に行なっていただきました秘書の林英子さんに深く感謝いたします．

そして，研究生活において様々なご助言やご意見，ご指導をくださった研究室の先輩方，苦楽を共にし，切磋琢磨し合った研究室の同期達，今後の活躍が期待される研究室の後輩達，気分転換に付き合ってくれた旧来の友人達，ここまで支えてくれた私の家族に心より感謝いたします．

参考文献

[1] E. F. Camacho and C. B. Alba, *Model predictive control*. Springer science & business media, 2013.

[2] D. Q. Mayne, "Model predictive control: Recent developments and future promise," *Automatica*, vol. 50, no. 12, pp. 2967–2986, 2014.

[3] G. Williams, P. Drews, B. Goldfain, J. M. Rehg, and E. A. Theodorou, "Aggressive driving with model predictive path integral control," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 1433–1440.

[4] G. Williams, B. Goldfain, P. Drews, J. M. Rehg, and E. A. Theodorou, "Autonomous racing with autorally vehicles and differential games," *arXiv preprint arXiv:1707.04540*, 2017.

[5] P. Abbeel, A. Coates, and A. Y. Ng, "Autonomous helicopter aerobatics through apprenticeship learning," *The International Journal of Robotics Research*, vol. 29, no. 13, pp. 1608–1639, 2010.

[6] J. M. Maciejowski, *Predictive control: with constraints*. Pearson education, 2002.

[7] 足立修一 and 管野政明, **モデル予測制御: 制約のもとでの最適制御**. 東京電機大学出版局, 2005.

[8] R. Rubinstein, "The cross-entropy method for combinatorial and continuous optimization," *Methodology and computing in applied probability*, vol. 1, no. 2, pp. 127–190, 1999.

[9] Z. I. Botev, D. P. Kroese, R. Y. Rubinstein, and P. L' Ecuyer, "The cross-entropy method for optimization," in *Handbook of statistics*. Elsevier, 2013, vol. 31, pp. 35–59.

[10] A. Nagabandi, G. Kahn, R. S. Fearing, and S. Levine, "Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 7559–7566.

[11] D. Hafner, T. Lillicrap, I. Fischer, R. Villegas, D. Ha, H. Lee, and J. Davidson, "Learning latent dynamics for planning from pixels," in *International Conference on Machine Learning*. PMLR, 2019, pp. 2555–2565.

[12] S. Zilberstein, "Using anytime algorithms in intelligent systems," *AI magazine*, vol. 17, no. 3, pp. 73–73, 1996.

[13] W. Li and E. Todorov, "Iterative linear quadratic regulator design for nonlinear biological movement systems." in *ICINCO (1)*. Citeseer, 2004, pp. 222–229.

[14] E. Todorov and W. Li, "A generalized iterative lqg method for locally-optimal feedback control of constrained nonlinear stochastic systems," in *Proceedings of the 2005, American Control Conference, 2005.* IEEE, 2005, pp. 300–306.

[15] T. Ohtsuka, "A continuation/gmres method for fast computation of nonlinear receding horizon control," *Automatica*, vol. 40, no. 4, pp. 563–574, 2004.

[16] Y. Tassa, T. Erez, and E. Todorov, "Synthesis and stabilization of complex behaviors through online trajectory optimization," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 4906–4913.

[17] S. L. Richter and R. A. Decarlo, "Continuation methods: Theory and applications," *IEEE Transactions on Systems, Man, and Cybernetics*, no. 4, pp. 459–464, 1983.

[18] C. T. Kelley, *Iterative methods for linear and nonlinear equations*. SIAM, 1995.

[19] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. Van der Vorst, *Templates for the solution of linear systems: building blocks for iterative methods*. SIAM, 1994.

[20] 朴達 and 大塚敏之, "C/gmres アルゴリズムによる非線形モデル予測制御を用いた四輪車両の衝突回避," in **自動制御連合講演会講演論文集 第** *51* **回自動制**

御連合講演会. 自動制御連合講演会, 2008, pp. 182–182.

[21] H. Seguchi and T. Ohtsuka, "Nonlinear receding horizon control of an underactuated hovercraft," *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, vol. 13, no. 3-4, pp. 381–398, 2003.

[22] 大塚敏之, "実時間最適化の考え方と応用," in **横幹連合コンファレンス予稿集 第 *8* 回横幹連合コンファレンス**. 横断型基幹科学技術研究団体連合 (横幹連合), 2017, pp. B–4.

[23] E. Hall and R. Willett, "Dynamical models and tracking regret in online convex programming," in *International Conference on Machine Learning*. PMLR, 2013, pp. 579–587.

[24] N. Wagener, C.-A. Cheng, J. Sacks, and B. Boots, "An online learning approach to model predictive control," *arXiv preprint arXiv:1902.08967*, 2019.

[25] C. Pinneri, S. Sawant, S. Blaes, J. Achterhold, J. Stueckler, M. Rolinek, and G. Martius, "Sample-efficient cross-entropy method for real-time planning," *arXiv preprint arXiv:2008.06389*, 2020.

[26] D. Chaudhuri, M. Mukherjee, M. H. Khondekar, and K. Ghosh, "Simple exponential smoothing and its control parameter: A reassessment," in *Recent Trends in Signal and Image Processing*. Springer, 2019, pp. 63–77.

[27] A. S. Nemirovskij and D. B. Yudin, "Problem complexity and method efficiency in optimization," 1983.

[28] A. Beck and M. Teboulle, "Mirror descent and nonlinear projected subgradient methods for convex optimization," *Operations Research Letters*, vol. 31, no. 3, pp. 167–175, 2003.

[29] Y. E. Nesterov, "A method for solving the convex programming problem with convergence rate o (1/kˆ 2)," in *Dokl. akad. nauk Sssr*, vol. 269, 1983, pp. 543–547.

[30] W. Krichene, A. Bayen, and P. Bartlett, "Accelerated mirror descent in continuous and discrete time," *Advances in neural information processing systems*, vol. 28, pp. 2845–2853, 2015.

[31] Y. Chen and M. J. Wainwright, "Fast low-rank estimation by projected gradient descent: General statistical and algorithmic guarantees," *arXiv preprint arXiv:1509.03025*, 2015.

[32] W. Su, S. Boyd, and E. Candes, "A differential equation for modeling nesterov' s accelerated gradient method: Theory and insights," *Advances in neural information processing systems*, vol. 27, pp. 2510–2518, 2014.

[33] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "Openai gym," *arXiv preprint arXiv:1606.01540*, 2016.

[34] E. Leurent:, "An environment for autonomous driving decision-making," 2018, https://github.com/eleurent/highway-env.

[35] P. Polack, F. Altché, B. d'Andréa Novel, and A. de La Fortelle, "The kinematic bicycle model: A consistent model for planning feasible trajectories for autonomous vehicles?" in *2017 IEEE intelligent vehicles symposium (IV)*. IEEE, 2017, pp. 812–818.

[36] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Physical review E*, vol. 62, no. 2, p. 1805, 2000.

[37] A. Kesting, M. Treiber, and D. Helbing, "General lane-changing model mobil for car-following models," *Transportation Research Record*, vol. 1999, no. 1, pp. 86–94, 2007.