



博士論文

冗長マニピュレータによる

高速な軌跡追従手先定速作業の動作計画に関する研究

筑波大学 システム情報工学研究科 知能機能システム専攻

岡部 弘佑

Kousuke OKABE

目次

第1章	序論	1
1.1	研究背景	2
1.2	研究目的	4
1.3	論文の構成	5
第2章	問題定式化と動作計画概要	8
2.1	始めに	9
2.2	冗長マニピュレータ定式化	10
2.3	軌跡追従手先定速作業定式化	11
2.4	動作計画システム概要	12
2.5	終わりに	15
第3章	状態空間定式化	16
3.1	はじめに	17
3.2	冗長パラメータの定式化	18
3.3	状態空間上の制限	22
3.3.1	駆動トルク制限による冗長加速度制限	22
3.3.2	関節速度制限による冗長速度制限	23
3.4	終わりに	25
第4章	高速動作が困難な区間の経路探索	26
4.1	始めに	27
4.2	難区間経路探索の概要	28
4.3	難区間経路探索	29

4.4	評価関数の設計	31
4.5	経路探索例	33
4.6	終わりに	40
第5章	高速動作が容易な区間の経路探索	41
5.1	始めに	42
5.2	易区間経路探索の概要	43
5.3	易区間経路探索	43
5.3.1	切換面の生成	43
5.3.2	駆動トルク最小となる冗長加速度の導出	45
5.3.3	易区間経路探索アルゴリズム	46
5.4	易区間経路探索例	53
5.5	終わりに	59
第6章	区間ごとの動作計画を用いた高速動作の計画	60
6.1	始めに	61
6.2	動作計画アルゴリズム	62
6.3	難区間と易区間へ手先軌跡の分割	66
6.4	最高手先速度の比較による有効性検証	69
6.5	動作計画の有効性	76
6.6	終わりに	80
第7章	動作計画適用例	81
7.1	始めに	82
7.2	直動関節を手首に有する水平マニピュレータへの適用	83
7.3	リニアスライダ上に設置された垂直多関節マニピュレータへの適用	90
7.4	手先軌道回転半径の最小化	98
7.5	終わりに	103

第 8 章 結論	104
8.1 まとめ	105
8.2 今後の展望	107
謝辞	108
参考文献	109
研究業績	114

第1章 序論

Contents

1.1	研究背景	2
1.2	研究目的	4
1.3	論文の構成	5

1.1 研究背景

産業界ではマニピュレータを用いた自動車ボディのアーク溶接や塗装，エンジン等の気密のためのシール材塗布，液晶ディスプレイ固定のための接着剤塗布など手先速度一定で手先位置指令軌跡を追従する軌跡追従手先定速作業は多く存在する．これら軌跡追従手先定速作業には手先速度の一定性の他にも，ばらつきを小さくするための繰り返し手先位置精度やタクトタイム短縮のための手先速度高速化，ワークの位置ズレに対応する手先軌跡の自動補正機能などが要求されている [1–3,45]．これらの要求に対してイメージセンサやレーザーレンジセンサによるビジュアルフィードバックを用いた手先軌跡の自動補正機能 [4–9] やスライディングモード制御や外乱オブザーバを利用したパラメータ変動やパラメータ誤差，外乱等にロバストな軌跡追従制御手法 [10–13] が提案されている．

軌跡追従手先定速作業動作は手先位置指令軌跡を一定手先速度で追従するため，指令軌跡の角部では他の箇所より大きな駆動トルクが必要となり，指令軌跡がマニピュレータの特異姿勢に近い箇所では他の箇所より大きな関節速度が必要となる．通常マニピュレータの各関節には駆動トルク制限や関節速度制限が存在するため，指令軌跡の角部や特異姿勢に近い箇所では追従に必要な駆動トルクや関節速度が制限に収まるような遅い手先速度を用いて指令軌跡を追従する必要がある．その結果，軌跡追従手先定速作業では指令軌跡の角部や特異姿勢付近を駆動トルク制限や関節速度制限内で追従可能な遅い手先速度を用いて全軌跡を追従することになるため，タクトタイムが大幅に長くなるという問題が存在する．そのためこの作業の高速化には，指令軌跡の角部や特異姿勢に近い箇所などの，大きな出力が必要となる箇所の高速化が非常に有効である．

軌跡追従手先定速作業の高速化を考える場合，指令軌跡のうち駆動トルクや関節速度が他の箇所に比べて大きな出力が必要となる箇所が，より小さい出力で追従可能な位置に来るように指令軌跡を配置する，つまりより速い手先速度で追従可能な位置に指令軌跡を配置する配置問題を解く方法が考えられる．配置問題を解くための手先位置や姿勢ごとのマニピュレータの性能を表す指標として，駆動トルクと手先加速度の関係を動的可操作性楕円体 [14]，特定の作業に対して最適な姿勢を決定するための *compatibility index* [15]，最もトルク伝達効率の高い姿勢を選択するためのインピーダンスマッチング楕円 [16] など

が提案され、他にも様々な性能を表す指標 [17-19] が提案されている。しかし配置問題を解く方法は、指令軌跡が複雑になることで遅い手先速度で追従する原因となる指令軌跡の角部等の数が多くなると、全ての箇所を速い手先速度で追従可能な位置に配置するのが困難になり、配置問題の効果は薄れていく。

マニピュレータの動作高速化手法としては、手先位置指令で与えられる2点を最短時間で移動する最適軌道計画手法 [20,21] や、与えられた手先位置指令軌跡を最短時間で追従する時間最適軌跡追従制御手法 [22-25] などが提案されている。しかし、最適軌道計画手法は Point to Point を最短時間で移動する経路を計画する手法であるため指令軌跡を追従することができず、時間最適軌跡追従制御手法では手先軌跡上の初端手先位置から終端手先位置までを最短時間で動作させることを目的として Bang-Bang 制御で手先軌跡を追従させるため、手先速度の一定性を満たすことができない。しかし、Bobrow らの提案する時間最適軌跡追従制御手法 [22] では指令軌跡上のマニピュレータ手先位置と手先軌跡上の手先速度からなる位相平面上にマニピュレータの駆動トルク制限を射影することで、駆動トルク制限と手先軌跡上における手先の運動との関係がわかりやすい表現手法を用いており、手先軌跡を追従する動作の計画にはとても有効な手法である。

一方、産業界では様々な作業にマニピュレータを用いており、必ずしもその作業専用のマニピュレータが用いられているわけではない。そのため、産業界では市販の汎用マニピュレータが広く使用されており、シーリング作業や溶接作業、塗装作業といったツールの向きに対するロール軸方向の姿勢が自由な作業において自由度が一つ余る。自由度が余っているマニピュレータは冗長マニピュレータと呼ばれ、この余った自由度、つまり冗長自由度を利用して手先は指令軌跡を追従しつつ、駆動トルクや関節速度の最小化 [26,27] や、障害物回避や特異姿勢の回避 [28,29] を行うなど、様々な冗長性を利用した手法 [30-32] が提案されている。このように、冗長マニピュレータは手先で指令軌跡を追従するというメインタスクを実行しながら、別のサブタスクを実行する能力を有しており、軌跡追従手先定速作業においても冗長性を利用することで、指令軌跡を追従するのに必要な駆動トルクや関節トルクを各関節に分散させることでより速い手先速度での追従が可能となる。前述した、Ma らや Galicki らの提案した時間最適軌跡追従制御手法 [24,25] では冗長マニピュレータに対して時間最適軌跡追従制御を適用しているが、これらの手法は冗長性を利

用することでより短い時間で追従することを目指した最適化手法ではなく、制御対象が冗長性を有することでマニピュレータの内部機構による動作が一意に決定しない問題を解決することで、冗長マニピュレータに対して時間最適軌跡追従制御を適用した手法である。また Ma らの論文 [24] では、冗長マニピュレータは手先の運動とは独立してマニピュレータの内部機構による動作が可能であるため、Bobrow らの手法 [22] のように位相平面上に駆動トルク制限を射影できないという問題が報告されている。

本論文では軌跡追従手先定速作業で要求される手先速度の一定性を満たしつつ軌跡追従手先定速作業動作の高速化を実現する動作計画システムの構築を行う。軌跡追従手先定速作業動作の高速化実現のため、冗長マニピュレータの冗長性を利用して各関節の駆動トルク制限・関節速度制限を回避することで高速化する手法を提案する。冗長マニピュレータの状態と駆動トルク制限・関節速度制限の関係は、Bobrow らの手法 [22] を位相平面上への射影から状態空間上への射影に拡張することで駆動トルク制限を射影できない問題を解決する。これにより動作計画問題を状態空間上に射影された駆動トルク制限、関節速度制限を回避する経路を計画する経路計画問題に帰着することでマニピュレータの動作計画を行う。

1.2 研究目的

本論文では手先位置指令軌跡を一定手先速度で追従する軌跡追従手先定速作業を対象として冗長自由度 1 を有する冗長マニピュレータの冗長性を利用した手先速度高速化を目的としたマニピュレータの動作計画システムの構築を行う。動作計画システムでは、まず手先軌跡の高手先速度動作が難しい区間について動作計画を行ない、次に高手先速度動作が容易な区間の動作計画を行う。動作計画システムは冗長マニピュレータの冗長性に関する状態空間上の経路探索問題として動作計画を行ない、高手先速度動作が困難な区間では状態空間上に射影された制限を回避する経路の探索を行ない、高手先速度動作が容易な区間では困難な区間で計画した経路を繋ぐ経路の探索を行う。

1.3 論文の構成

2章では冗長自由度1を有する冗長マニピュレータの定式化と軌跡追従手先定速作業の定式を行ない、マニピュレータの冗長性を利用した軌跡追従手先定速作業高速化のための動作計画システムの概要について述べる。まず冗長自由度1を有する冗長マニピュレータの順運動学関数から手先速度や手先加速度、関節変位、関節速度、関節加速度に関する幾何学的特性の定式化を行う。次にマニピュレータの動力学的特性についてもマニピュレータの運動方程式より定式化する。軌跡追従手先定速作業を行うことによる制限を示し、軌跡追従手先定速作業の定式化を行う。最後に、本論文で述べるマニピュレータの冗長性を利用した軌跡追従手先定速作業の高速化を行う動作計画システムの概要とこの動作計画システムで取り扱う問題設定について述べる。

3章では冗長性による動作をスカラー値の冗長パラメータで決定する式に定義し直し、新たに定義した冗長パラメータを状態量とした状態方程式の導出と、冗長パラメータを状態量とした状態空間上に駆動トルク制限と関節速度制限の射影を行う。まず、一般的な冗長マニピュレータの関節速度、関節加速度に関する式において関節空間と同次元のベクトルを用いて決定される冗長性による動作を冗長性による動作を表す項の正規基底ベクトルを用いることでスカラー値で決定される式とすることで、冗長性による動作をスカラーパラメータとして定義する。そして、手先位置に影響を与えない冗長性による動作と手先速度に影響を与えない冗長性による動作を状態量、手先加速度に影響を与えない冗長性による動作を入力変数とした状態方程式の導出を行う。最後に手先軌跡上の位置と手先位置に影響を与えない冗長性による動作、手先速度に影響を与えない冗長性による動作を軸とした状態空間を考え、この状態空間上に駆動トルク制限と関節速度制限を射影することで動作計画において経路が侵入してはいけない違反領域の導出を行う。

4章では手先軌跡の高手先速度動作が困難な区間の動作計画手法について述べる。まず、手先軌跡の高手先速度動作が困難な区間の動作計画を状態空間上の経路探索問題として考える難区間経路探索の概要について述べる。次に難区間経路探索において探索を行う、区間の初端状態の探索と初端状態から違反領域に進入することなく区間の終端まで到達する経路の探索アルゴリズムについて述べる。その後、区間の初端状態から違反領域に進入

することなく区間の終端まで到達する経路の探索に用いる経路の評価関数の設計を行う。最後に難区間経路探索の一例を示し、難区間経路探索を行うことで違反領域を回避するような経路が探索されることを示す。

5章では手先軌跡の高手先速度動作が容易な区間の動作計画手法について述べる。まず、難区間経路探索によって計画された状態空間上の経路を繋ぐ経路を計画する高手先速度動作が容易な区間の動作計画手法である易区間経路探索の概要について述べる。次に、易区間経路探索において重要な概念となる切換面という必ず区間の終端状態に到達する経路の存在する面の導出と、切換面の導出に必要な駆動トルクベクトルのノルムが最小となる関節加速度に影響を与えない冗長性による動作の導出を行う。そして、切換面を用いて難区間経路探索により計画された経路を繋ぐ経路を計画する易区間経路探索アルゴリズムについて述べる。最後に易区間経路探索の一例を示し、易区間経路探索を行うことで高手先速度動作が容易な区間で区間の初端状態と終端状態を繋ぐ経路が探索されることを示す。

6章では4章と5章で述べる難区間経路探索、易区間経路探索を用いて高速な軌跡追従手先定速作業動作を計画する動作計画システムとその有効性について述べる。まず、難区間経路探索と易区間経路探索を用いて高速な軌跡追従手先定速作業動作を計画する動作計画システムの動作計画アルゴリズムについて述べる。次に、手先軌跡を高手先速度動作が困難な区間と容易な区間に分割する手先軌跡の分割手法について述べる。その後、シミュレーションを用いて一般的な冗長マニピュレータの動作制御手法である関節加速度最小となる動作と駆動トルク最小となる動作の2手法を用いた場合の最大手先速度と動作計画システムを用いた場合の最大手先速度を比較することで動作計画システムの有効性を示す。最後に関節加速度最小となる動作と駆動トルク最小となる動作は動作計画システムの探索空間内に含まれることを示すことで、これらの一般的な冗長マニピュレータの動作制御手法により得られる最大手先速度より動作計画システムを用いることで必ず高速化されることを示す。

7章では動作計画システムを様々な構成のマニピュレータに対して適用することで、動作計画システムが冗長自由度1を有するマニピュレータを用いて手先位置の軌跡指令を追従する軌跡追従手先定速作業に対して一般化されていることを示し、動作計画システムを応用した丸角四角軌跡の角部円弧軌跡半径最小化手法について述べる。まず、水平多関

節マニピュレータの手首部に直道関節を有する冗長マニピュレータに対して動作計画システムを適用し、様々な関節構造のマニピュレータに対して動作計画システムが適用可能であることを示す。次にリニアスライダ上に垂直多関節マニピュレータが設置された構成の冗長マニピュレータに対して動作計画システムを適用し、3次元の作業空間を有するマニピュレータに対しても動作計画システムが適用可能であることを示す。最後に動作計画システムを応用した軌跡追従手先定速作業において丸角四角軌跡を追従する場合の角部円弧軌跡半径を最小化する手法について述べる。

最後に8章で本論文のまとめと今後の展望について述べる。

第2章 問題定式化と動作計画概要

Contents

2.1	はじめに	9
2.2	冗長マニピュレータ定式化	10
2.3	軌跡追従手先定速作業定式化	11
2.4	動作計画システム概要	12
2.5	終わりに	15

2.1 始めに

マニピュレータを用いた作業の動作計画を行うためにはマニピュレータと対象作業について把握しておく必要がある。本章では作業を行うマニピュレータと対象作業である軌跡追従手先定速作業をモデル化しこれらの定式化を行う。

研究目的であるマニピュレータの冗長性を利用した軌跡追従手先定速作業の高速化について考える場合、マニピュレータの動力学を無視できなくなるためマニピュレータの幾何学的特性だけでなく動力学的特性についてもモデル化して定式化する必要がある。

この章では、2.2節で幾何学的冗長性を1つ有するシリアルリンクマニピュレータの定式化を行う。2.3節では、対象作業である軌跡追従手先定速作業の概要と問題点、作業における制約について述べる。2.4節では、マニピュレータの冗長性を利用した軌跡追従手先定速作業の高速化のために本研究で解決すべき問題の設定と、その解となる動作を計画する動作計画システムの概要について述べる。

2.2 冗長マニピュレータ定式化

本項では吉田の著書 [42] や花房らの論文 [33] を参考にして、動作計画対象となる冗長自由度 1 を有する n 自由度冗長マニピュレータの定式化を行う。

マニピュレータ関節変位は $\mathbf{q} = [q_1, q_2, \dots, q_n]^T \in \mathbb{R}^n$ で表し、関節速度を $\dot{\mathbf{q}} = [\dot{q}_1, \dot{q}_2, \dots, \dot{q}_n] \in \mathbb{R}^n$ 、関節加速度を $\ddot{\mathbf{q}} = [\ddot{q}_1, \ddot{q}_2, \dots, \ddot{q}_n] \in \mathbb{R}^n$ で表す。また、マニピュレータは $m = n - 1$ 次元の作業空間を有し、マニピュレータの関節空間から作業空間への変換は次式で与えられる。

$$\mathbf{r} = P(\mathbf{q}) \quad (2.1)$$

但し、 $\mathbf{r} \in \mathbb{R}^m$ はマニピュレータの手先位置ベクトルを表し、 $P(\mathbf{q})$ はマニピュレータの順運動学関数で m 個のスカラー関数からなるベクトル関数である。式 (2.1) を 1 階、2 階時間微分することでマニピュレータの手先速度ベクトル $\dot{\mathbf{r}} \in \mathbb{R}^m$ と手先加速度ベクトル $\ddot{\mathbf{r}} \in \mathbb{R}^m$ が次式のように得られる。

$$\dot{\mathbf{r}} = \mathbf{J}\dot{\mathbf{q}} \quad (2.2)$$

$$\ddot{\mathbf{r}} = \mathbf{J}\ddot{\mathbf{q}} + \dot{\mathbf{J}}\dot{\mathbf{q}} \quad (2.3)$$

但し、 $\mathbf{J} = \mathbf{J}(\mathbf{q}) = \frac{\partial \mathbf{r}}{\partial \mathbf{q}^T} \in \mathbb{R}^{m \times n}$ はマニピュレータのヤコビ行列、 $\dot{\mathbf{J}} = \dot{\mathbf{J}}(\mathbf{q}, \dot{\mathbf{q}}) = \frac{d}{dt} \mathbf{J}$ である。

本研究では冗長自由度 1 を有するマニピュレータを対象としているため、このマニピュレータは作業空間に影響を及ぼさない内部機構に 1 自由度を有する。そのため、作業空間から関節空間への変換には、この冗長性による動作を表すパラメータを導入する必要がある。本研究では逆運動学を解くために冗長な姿勢を表すパラメータ $R_x \in \mathbb{R}$ を用いて作業空間から関節空間への変換を次式のようにした。

$$\mathbf{q} = P^{-1}(\mathbf{r}, R_x) \quad (2.4)$$

$$\dot{\mathbf{q}} = \mathbf{J}^+ \dot{\mathbf{r}} + (\mathbf{I} - \mathbf{J}^+ \mathbf{J}) \mathbf{k}_1 \quad (2.5)$$

$$\ddot{\mathbf{q}} = \mathbf{J}^+(\ddot{\mathbf{r}} - \dot{\mathbf{J}}\dot{\mathbf{q}}) + (\mathbf{I} - \mathbf{J}^+\mathbf{J})\mathbf{k}_2 \quad (2.6)$$

但し, $\mathbf{P}^{-1}(\mathbf{r}, R_x)$ は逆運動学関数で n 個のスカラー関数からなるベクトル関数, $\mathbf{J}^+ = \mathbf{J}^T(\mathbf{J}\mathbf{J}^T)^{-1} = \mathbf{J}^+(\mathbf{q}) \in \mathbb{R}^{n \times m}$ は \mathbf{J} の疑似逆行列, $\mathbf{I} \in \mathbb{R}^{n \times n}$ は単位行列, $\mathbf{k}_1 = [k_{11}, k_{12}, \dots, k_{1n}]^T \in \mathbb{R}^n$, $\mathbf{k}_2 = [k_{21}, k_{22}, \dots, k_{2n}]^T \in \mathbb{R}^n$ は任意の n 次元ベクトルである.

マニピュレータの駆動トルクは $\boldsymbol{\tau} = [\tau_1, \tau_2, \dots, \tau_n]^T \in \mathbb{R}^n$ で表し, ラグランジュの運動方程式よりマニピュレータの動力学方程式は次式で与えられる. ラグランジュの運動方程式を用いたマニピュレータの動力学方程式の導出方法は文献 [42,43] で詳しく述べられている.

$$\boldsymbol{\tau} = \mathbf{M}\ddot{\mathbf{q}} + \mathbf{H} + \mathbf{g} + \mathbf{d} \quad (2.7)$$

但し $\mathbf{M} = \mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n \times n}$ は慣性行列, $\mathbf{H} = \mathbf{H}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^n$ は遠心力やコリオリ力, $\mathbf{g} = \mathbf{g}(\mathbf{q}) \in \mathbb{R}^n$ は重力, $\mathbf{d} = \mathbf{d}(\dot{\mathbf{q}}) \in \mathbb{R}^n$ は摩擦等による外乱である.

マニピュレータの各関節には駆動トルク制限 Ω_τ と関節速度制限 Ω_v が存在し, 次式で与えられる.

$$\Omega_\tau = \{\boldsymbol{\tau} \mid -\tau_{limi} \leq \tau_i \leq \tau_{limi}; i = 1, 2, \dots, n\} \quad (2.8)$$

$$\Omega_v = \{\dot{\mathbf{q}} \mid -\dot{q}_{limi} \leq \dot{q}_i \leq \dot{q}_{limi}; i = 1, 2, \dots, n\} \quad (2.9)$$

2.3 軌跡追従手先定速作業定式化

本研究の対象作業である軌跡追従手先定速作業は, マニピュレータの作業空間上の手先位置指令軌跡を全軌跡において一定の手先速度で追従する作業である. 産業界で行われている作業としてはシーリング作業や塗装作業, 溶接作業などが軌跡追従手先定速作業にあたる. 軌跡追従手先定速作業は主に以下の問題を有する.

- Bobrow らの論文 [22] で動作計画に用いられている Fig.2.1 に示すような手先位相平面と平面上に射影された駆動トルク制限を考えると, 研究対象作業は手先定速作業

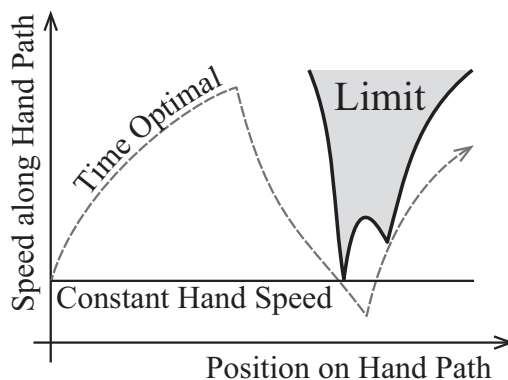


Fig. 2.1: 手先位相平面

であるため制限内で動作可能な最大手先速度が最も低速になる地点の手先速度で全手先軌道を動作させなければならない。そのため、局所的な動作可能な最大手先速度の低速化が全軌道における動作に影響し、大幅にタクトタイムが長くなる。

Fig.2.2 のようにマニピュレータの手先軌道は、軌跡に沿った初期手先位置からの距離 s の関数である軌跡指令関数 $\mathbf{f}(s)$, ($s_0 \leq s \leq s_1$) を用いて式 (2.10)-(2.12) のように与えられる。 $\mathbf{f}(s_0)$ は手先初期位置, $\mathbf{f}(s_1)$ は手先終端位置である。

$$\mathbf{r} = \mathbf{f} \quad (2.10)$$

$$\dot{\mathbf{r}} = \mathbf{f}' \dot{s} \quad (2.11)$$

$$\ddot{\mathbf{r}} = \mathbf{f}'' \dot{s}^2 \quad (2.12)$$

但し, $\mathbf{f} = \mathbf{f}(s)$, $\mathbf{f}' = \frac{d}{ds}\mathbf{f}(s)$, $\mathbf{f}'' = \frac{d^2}{ds^2}\mathbf{f}(s)$ で, 軌跡追従手先定速作業であるため $\dot{s} = 0$ としている。

2.4 動作計画システム概要

本研究で提案する動作計画システムでは軌跡追従手先定速作業高速化のための冗長性による動作の計画を, 状態空間上の経路探索問題に帰着することで行う。

動作計画システムは, Fig.2.3 に示す概要図のようにマニピュレータの冗長性による動作を, 状態空間上での経路探索問題として動作計画を行う。状態空間は軌跡上の位置 s と

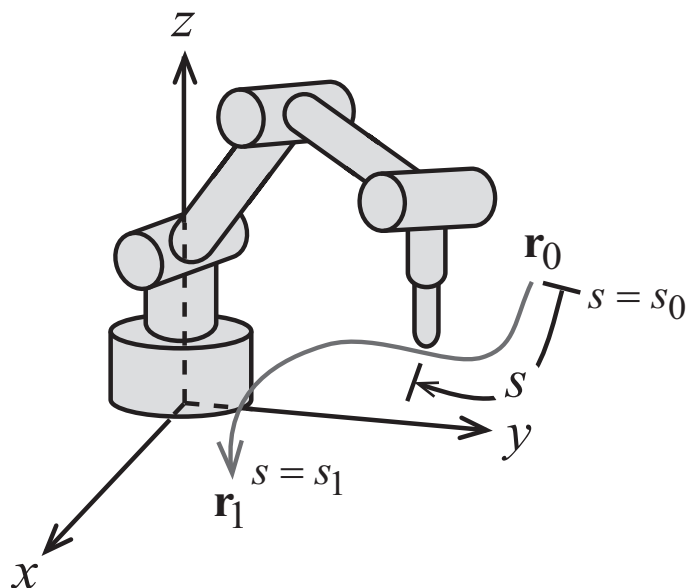


Fig. 2.2: 軌跡追従手先定速作業

マニピュレータの手先位置に影響を及ぼさない冗長性による姿勢のパラメータ（冗長姿勢）、手先速度に影響を及ぼさない冗長性による関節速度のパラメータ（冗長速度）を軸として持つ3次元空間で、軌跡追従手先定速動作は状態空間上の経路として表すことができる。そのため、本研究では軌跡追従手先定速作業における冗長性による動作の計画を状態空間上の経路探索問題に置き換えることができる。このとき、状態空間上の経路を左右する制御入力の手先加速度に影響を及ぼさない関節加速度のパラメータ（冗長加速度）となる。状態空間上にはマニピュレータの有する駆動トルク制限と関節角速度制限が射影され、状態空間上でこれらの制限を回避する経路を探索することで、駆動トルク制限と関節速度制限を満たす動作が計画される。本研究では状態空間上に射影された駆動トルク制限と関節速度制限を超過する領域を違反領域と呼ぶ。経路探索では Fig.2.3 のように、指令軌跡を高手先速度動作が困難な難区間と高手先速度動作が容易な易区間に区間分割し、それぞれの区間の状態空間上で難区間経路探索と易区間経路探索を行う。そして動作計画を行う手先速度 s を順次高速化していき、経路探索により経路が見つからなくなるまで s を高速化することで高速な軌跡追従手先定速作業の動作を計画する。この難区間経路探索は違反領域を回避する経路を求めることに重点を置いた手法で、易区間経路探索は難区間経路探索により得られた経路を繋ぐ経路を探索することで難区間の動作の予備動作を計

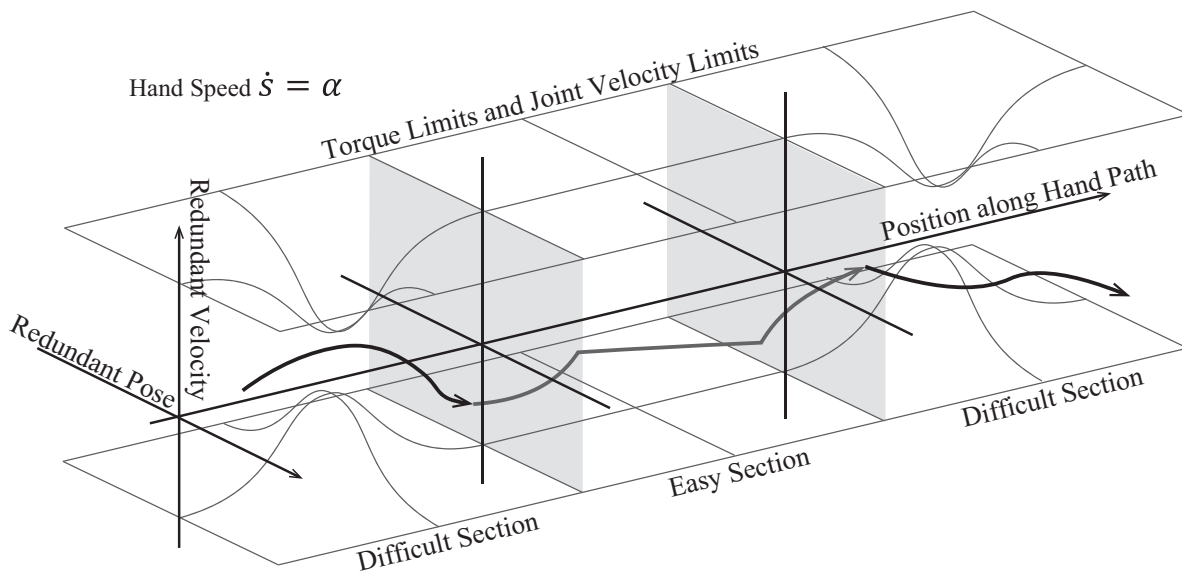


Fig. 2.3: 動作計画概要図

画する手法である。易区間経路探索手法はあくまで難区間のための予備動作の計画であるため、易区間で経路が見つからない場合はその区間の探索方法を難区間経路探索に切り替え、難区間経路探索により経路が見つからなくなるまで探索を行う。

本研究の目的は「マニピュレータの冗長性を利用した軌跡追従手先定速作業の高速化であるため、本研究で取り扱う問題は次のようになる。

問題設定

冗長自由度1のシリアルリンクマニピュレータを対象として、 f, f', f'' で与えられる連続した手先位置の指令軌跡を、手先速度一定 ($\dot{s} = 0$) かつ駆動トルク制限 Ω_τ 、関節速度制限 Ω_v 内で追従できる、可能な限り高速な手先速度 s とその時の動作 (q, \dot{q}, \ddot{q}) を求める。

2.5 終わりに

2章では, 冗長自由度を1つ有する n 自由度マニピュレータの定期化と, このマニピュレータに行わせる軌跡追従手先定速作業の定式化を行ない, 本研究の動作計画システムの概要について述べた. マニピュレータの幾何学的性質は式 (2.1) より定式化され, 動力学的性質はラグランジュの運動方程式より式 (2.7) のように定式化された. また, 軌跡追従手先定速作業を行うために必要な制約を式 (2.10), (2.12) のように定式化した. これらの式を満たす動作を2.4節で述べたような動作計画システムを用いて計画することで, マニピュレータの冗長性を利用した高速な軌跡追従手先定速作業動作が得られる.

第3章 状態空間定式化

Contents

3.1	はじめに	17
3.2	冗長パラメータの定式化	18
3.3	状態空間上の制限	22
3.3.1	駆動トルク制限による冗長加速度制限	22
3.3.2	関節速度制限による冗長速度制限	23
3.4	終わりに	25

3.1 はじめに

本研究の動作計画システムは状態空間上の経路探索問題として高速な軌跡追従手先定速作業動作の計画を行う。本章では動作計画システムで用いる冗長性による動作を状態量とした状態空間の定義と、駆動トルク制限、関節速度制限の状態空間上への射影を行う。

マニピュレータの冗長性による動作は式(2.4)の冗長姿勢 R_x 、式(2.5)、(2.6)の右辺第2項で定式化されている。式(2.5)、(2.6)の右辺第2項は1つの冗長自由度による運動に対してそれぞれ n 個のパラメータを定める必要があり、探索空間が不必要に広がっている。そのため、本章では式(2.5)、(2.6)右辺第2項の冗長性による動作を、右辺第2項の基底ベクトルを用いてスカラー値で表す式を用いて定義し直す必要がある。

本章では、3.2節でマニピュレータの冗長性による動作をスカラー値で定義し、冗長性による動作を状態量とした状態空間を構成する。また、新たに定義した冗長性による動作を表す冗長パラメータを状態量とした状態方程式の導出を行う。3.3節では構成した状態空間上へのマニピュレータの駆動トルク制限と関節速度制限の射影を行う。

3.2 冗長パラメータの定式化

本研究で対象としている軌跡追従手先定速作業では作業空間上の運動は s によって決定されるため、マニピュレータの関節に関するパラメータ $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$ では拘束条件が存在し、動作計画を行うには適していない。そのため、本研究では作業空間上の動作とは独立した冗長性による動作に対して動作計画を行うが、式(2.5),(2.6)では手先速度とは独立した冗長性による関節速度と手先加速度とは独立した冗長性による関節加速度にあたる右辺第2項が n 次元のベクトルによって決定する。1冗長自由度の動作に対して n 次元ベクトルを用いて動作計画を行うのは非効率的であるため、本章では手先の運動とは独立した冗長性による動作の姿勢、関節速度、関節加速度をスカラー値の冗長姿勢 R_x 、冗長速度 R_v 、冗長加速度 R_a を用いて定義し、これら変数を用いた状態方程式の導出を行う。

マニピュレータの冗長性による動作を表す式(2.5),(2.6)の右辺第2項は文献[42,44]によると式(3.1)のように特異値分解することができるため、式(2.5),(2.6)は式(3.2),(3.3)のように変換することができ、スカラー値 R_v, R_a によって $\dot{\mathbf{q}}, \ddot{\mathbf{q}}$ が得られる式とすることができる。

$$(I - \mathbf{J}^+ \mathbf{J}) = \mathbf{U} \mathbf{D}_\lambda \mathbf{V}^T \quad (3.1)$$

$$\dot{\mathbf{q}} = \mathbf{J}^+ \dot{\mathbf{r}} + \mathbf{U} R_v \quad (3.2)$$

$$\ddot{\mathbf{q}} = \mathbf{J}^+ (\ddot{\mathbf{r}} - \dot{\mathbf{J}} \dot{\mathbf{q}}) + \mathbf{U} R_a \quad (3.3)$$

但し、 $\mathbf{U} \in \mathbb{R}^n$ は \mathbf{J}^+ の張る空間の直交補空間の正規基底ベクトル、 $\mathbf{D}_\lambda \in \mathbb{R}$ は $(I - \mathbf{J}^+ \mathbf{J})$ の特異値を大きいものから対角に並べた行列で、本研究では冗長自由度1としているためスカラー値となる。 $\mathbf{V} \in \mathbb{R}^n$ は正規基底ベクトルである。また、冗長速度は $R_v = \mathbf{D}_\lambda \mathbf{V}^T \mathbf{k}_1 \in \mathbb{R}$ 、冗長加速度は $R_a = \mathbf{D}_\lambda \mathbf{V}^T \mathbf{k}_2 \in \mathbb{R}$ の任意のスカラー値となる。

ここで手先作業空間の次元を $m = 2$ とすると、Fig.3.1のように関節速度空間上ではヤコビ行列の擬似逆行列 \mathbf{J}^+ の1列目の列ベクトル \mathbf{J}_1^+ と、2列目の列ベクトル \mathbf{J}_2^+ によって張られる平面に直交するように正規基底ベクトル \mathbf{U} が与えられる。ベクトル $\mathbf{J}_1^+, \mathbf{J}_2^+$ は手先速度ベクトル $\dot{\mathbf{r}}$ の成分ごとの基底ベクトルであるため、ベクトル \mathbf{U} を基底とする冗長速

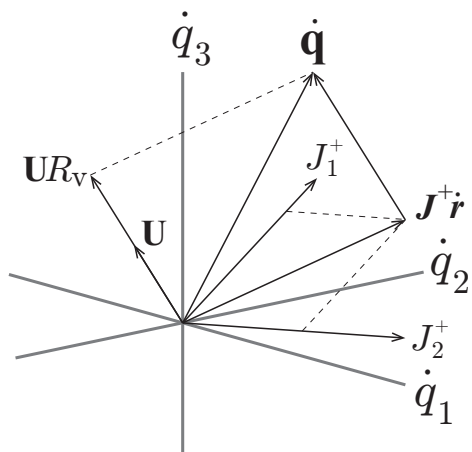


Fig. 3.1: 関節速度空間上の冗長速度

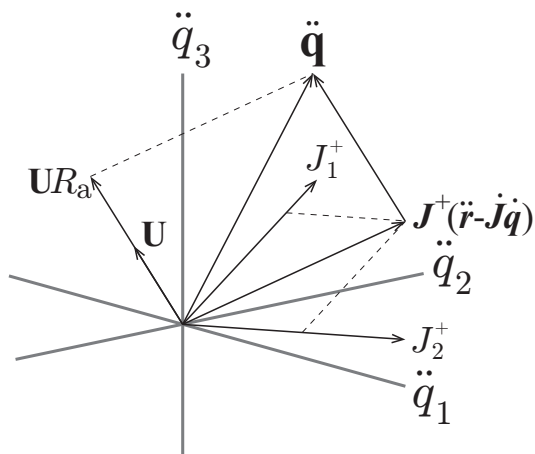


Fig. 3.2: 関節加速度空間上の冗長加速度

度 R_v の関節速度空間上のベクトル $\mathbf{U}R_v$ は手先速度に影響を与えないマニピュレータの内部機構の動作速度を表しており、 $R_v = 0$ の時に関節速度ベクトルのノルム $\|\dot{\mathbf{q}}\|$ が最小になる。また、Fig.3.2 のように関節加速度空間上でも正規基底ベクトル \mathbf{U} はヤコビ行列の擬似逆行列の各列ベクトル \mathbf{J}_i^+ ($i = 1, 2, \dots, m$) と直交したベクトルとなる。そのため、冗長加速度 R_a はベクトル $\ddot{\mathbf{r}} - \dot{\mathbf{J}}\dot{\mathbf{q}}$ に影響を与えない内部機構の加速度を表しており、 $R_a = 0$ の時に関節加速度ベクトルのノルム $\|\ddot{\mathbf{q}}\|$ は最小となる。

次にマニピュレータの冗長性による動作について状態量を R_x, R_v 、入力を R_a とした状態方程式について述べる。状態方程式を求めるためには、冗長姿勢 R_x に関する式が必要となる。冗長マニピュレータの逆運動学問題はマニピュレータの構成によって様々であ

るが、Leeらは冗長マニピュレータの冗長自由度数だけマニピュレータの関節を冗長関節と定義し、冗長関節の関節変位を冗長パラメータ（本研究で言う冗長姿勢 R_x ）とする手法 [36] を提案している。しかし、マニピュレータの構成によっては単一の関節変位を冗長パラメータとするより関節変位の和を用いた方が冗長パラメータによるマニピュレータの姿勢を理解しやすい。そのため著者らは冗長姿勢 R_x を式 (3.4) のように定義した。

$$R_x = \Sigma q \quad (3.4)$$

但し、 $\Sigma \in \mathbb{R}^{1 \times n}$ は $[\mathbf{J}^T, \Sigma^T]^T$ が正則、つまり $\Sigma \mathbf{U} \neq 0$ を満たし、マニピュレータの逆運動学問題が解きやすいように選択される任意の定数ベクトルである。

冗長姿勢 R_x の時間微分は式 (3.2),(3.4) より次式となる。

$$\dot{R}_x = \Sigma \mathbf{J}^+ \dot{\mathbf{r}} + \Sigma \mathbf{U} R_v \quad (3.5)$$

冗長速度 R_v の時間微分は \mathbf{U} が \mathbf{J}^+ の張る空間の直行補空間の正規基底ベクトルであることから $\mathbf{U}^T \mathbf{J}^+ = 0$ であることを用いると式 (3.2),(3.3) より次式が得られる。

$$\dot{R}_v = R_a + \dot{\mathbf{U}}^T \mathbf{J}^+ \dot{\mathbf{r}} + \dot{\mathbf{U}}^T \mathbf{U} R_v \quad (3.6)$$

式 (3.5) と式 (3.6) を冗長姿勢 R_x と冗長速度 R_v を状態量、冗長加速度 R_a を入力として状態方程式の形にまとめると次式のようなになる。

$$\frac{d}{dt} \begin{bmatrix} R_x \\ R_v \end{bmatrix} = \begin{bmatrix} 0 & \Sigma \mathbf{U} \\ 0 & \dot{\mathbf{U}}^T \mathbf{U} \end{bmatrix} \begin{bmatrix} R_x \\ R_v \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} R_a + \begin{bmatrix} \Sigma \\ \dot{\mathbf{U}}^T \end{bmatrix} \mathbf{J}^+ \dot{\mathbf{r}} \quad (3.7)$$

式 (3.7) を用いることで、初期状態と各時刻の冗長加速度 $R_a(t)$ より各時刻の状態が得られる。また、状態の更新則は次式のようなになる。

$$\begin{bmatrix} R_x(k+1) \\ R_v(k+1) \end{bmatrix} = \begin{bmatrix} R_x(k) \\ R_v(k) \end{bmatrix} + \begin{bmatrix} \dot{R}_x(k) \\ \dot{R}_v(k) \end{bmatrix} T_s \quad (3.8)$$

$$\begin{bmatrix} \dot{R}_x(k) \\ \dot{R}_v(k) \end{bmatrix} = \begin{bmatrix} 0 & \mathbf{\Sigma} \mathbf{U}(k) \\ 0 & \dot{\mathbf{U}}^T(k) \mathbf{U}(k) \end{bmatrix} \begin{bmatrix} R_x(k) \\ R_v(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} R_a(k) + \begin{bmatrix} \mathbf{\Sigma} \\ \dot{\mathbf{U}}^T(k) \end{bmatrix} \mathbf{J}^+(k) \mathbf{f}'(k) \dot{s}$$

但し, $\mathbf{f}(k) = \mathbf{f}(s(k))$, $\mathbf{f}'(k) = \mathbf{f}'(s(k))$, $\mathbf{f}''(k) = \mathbf{f}''(s(k))$ で, カッコ内の k は k サンプル目の値を表す.

正規基底ベクトル \mathbf{U} は MATLAB の SVD 関数を用いて求める方法や Park ら [34] が提案している手法, Chang [35] が提案している手法等があるが, Park らの手法は MATLAB の SVD 関数と比べて 5~10 倍高速に演算可能であると述べられている. また, 本稿では状態方程式に正規基底ベクトルの時間微分 $\dot{\mathbf{U}}$ を必要とするため, \mathbf{U} と同時に $\dot{\mathbf{U}}$ が得られる Park らの手法を用いて正規基底ベクトルを求める. 正規基底ベクトルには次式が成り立つ.

$$\mathbf{J} \mathbf{U} = 0 \quad (3.9)$$

$$\mathbf{U}^T \mathbf{U} = 1 \quad (3.10)$$

これらを時間微分すると

$$\mathbf{J} \dot{\mathbf{U}} = -\mathbf{J} \mathbf{U} \quad (3.11)$$

$$\mathbf{U}^T \dot{\mathbf{U}} = 0 \quad (3.12)$$

式(3.11),(3.12)をまとめると

$$\dot{\mathbf{U}} = \begin{bmatrix} \mathbf{J} \\ \mathbf{U}^T \end{bmatrix}^{-1} \begin{bmatrix} -\mathbf{J} \mathbf{U} \\ 0 \end{bmatrix} \quad (3.13)$$

式(3.13)より正規基底ベクトル \mathbf{U} の更新則はオイラー法を用いて次式のように得られる.

$$\mathbf{U}(k) = \mathbf{U}(k-1) + \dot{\mathbf{U}}(k) T_s \quad (3.14)$$

$$\dot{\mathbf{U}}(k) = \begin{bmatrix} \mathbf{J}(k-1) \\ \mathbf{U}^T(k-1) \end{bmatrix}^{-1} \begin{bmatrix} -\mathbf{J}(k)\mathbf{U}(k-1) \\ 0 \end{bmatrix}$$

但し, T_s はサンプリングタイムを表す.

この Park らの手法は $\dot{\mathbf{U}}$ を用いて逐次的に正規基底ベクトル \mathbf{U} を求めるため, 第1サンプル目に対しては適用することができない. そのため, 本研究では最初のサンプリングのみ MATLAB の SVD 関数を用いて正規基底ベクトル \mathbf{U} を求めている. また, \mathbf{U} は \mathbf{J}^+ の張る空間の直交補空間の正規基底ベクトルであるため, 反対方向を向く2通りのベクトルを取ることができ, 一意に定まらない. 従って, 本研究では $\Sigma\mathbf{U} > 0$ となる方向のベクトルを正規基底ベクトル \mathbf{U} としている.

3.3 状態空間上の制限

3.3.1 駆動トルク制限による冗長加速度制限

マニピュレータの運動方程式 (2.7) は (3.2), (3.3), (2.11), (2.12) から次式のように書き換えられる.

$$\boldsymbol{\tau} = \mathbf{c}_1 \mathbf{R}_a + \mathbf{c}_2 \quad (3.15)$$

$$\mathbf{c}_1 = \mathbf{M}\mathbf{U}$$

$$\mathbf{c}_2 = \mathbf{M}\mathbf{J}^+ \{ \mathbf{f}'' \dot{s}^2 - \dot{\mathbf{J}}\mathbf{J}^+ \mathbf{f}' \dot{s} - \dot{\mathbf{J}}\mathbf{U}\mathbf{R}_v \} + \mathbf{H} + \mathbf{g} + \mathbf{d}$$

式 (2.8) の駆動トルク制限は式 (3.15) より次式のようになる.

$$-\tau_{ilim} - c_{2i} \leq c_{1i} R_a \leq \tau_{ilim} - c_{2i} \quad (i = 1, 2, \dots, n) \quad (3.16)$$

但し, 添え字 i は各ベクトルの行番号を表している. ここで $c_{1i} \neq 0$ とすると, 冗長加速度 R_a の制限は次式のようになる.

$$F_{\tau i} \leq R_a \leq G_{\tau i} \quad (i = 1, 2, \dots, n) \quad (3.17)$$

$$G_{\tau i} = \begin{cases} (\tau_{ilim} - c_{2i})/c_{1i}, & (c_{1i} > 0) \\ (-\tau_{ilim} - c_{2i})/c_{1i}, & (c_{1i} < 0) \end{cases}$$

$$F_{\tau i} = \begin{cases} (-\tau_{ilim} - c_{2i})/c_{1i}, & (c_{1i} > 0) \\ (\tau_{ilim} - c_{2i})/c_{1i}, & (c_{1i} < 0) \end{cases}$$

式(3.17)は第*i*関節の駆動トルク制限 $\Omega_{\tau i}$ による冗長加速度 R_a の制限を表している。よって R_a の制限は次式のようになる。

$$F_{\tau} \leq R_a \leq G_{\tau} \quad (3.18)$$

$$G_{\tau} = \min\{G_{\tau i}; i = 1, 2, \dots, n\}$$

$$F_{\tau} = \max\{F_{\tau i}; i = 1, 2, \dots, n\}$$

3.3.2 関節速度制限による冗長速度制限

マニピュレータの関節速度に関する式(3.2)は(2.11)から次式のように書き換えられる。

$$\dot{\mathbf{q}} = \mathbf{c}_3 R_v + \mathbf{c}_4 \quad (3.19)$$

$$\mathbf{c}_3 = \mathbf{U}$$

$$\mathbf{c}_4 = \mathbf{J}^+ \mathbf{f}' \dot{\mathbf{s}}$$

式(2.9)の関節速度制限は式(3.19)より次式のようになる。

$$-\dot{q}_{ilim} - c_{4i} \leq c_{3i}R_v \leq \dot{q}_{ilim} - c_{4i} \quad (i = 1, 2, \dots, n) \quad (3.20)$$

ここで $c_{3i} \neq 0$ とすると、冗長速度 R_v の制限は次式のようになる。

$$F_{vi} \leq R_v \leq G_{vi} \quad (i = 1, 2, \dots, n) \quad (3.21)$$

$$G_{vi} = \begin{cases} (\dot{q}_{ilim} - c_{4i})/c_{3i}, & (c_{3i} > 0) \\ (-\dot{q}_{ilim} - c_{4i})/c_{3i}, & (c_{3i} < 0) \end{cases}$$

$$F_{vi} = \begin{cases} (-\dot{q}_{ilim} - c_{4i})/c_{3i}, & (c_{3i} > 0) \\ (\dot{q}_{ilim} - c_{4i})/c_{3i}, & (c_{3i} < 0) \end{cases}$$

式(3.21)は第 i 関節の関節速度制限 Ω_{vi} による冗長速度 R_v の制限を表している。よって R_v の制限は次式のようになる。

$$F_v \leq R_v \leq G_v \quad (3.22)$$

$$G_v = \min\{G_{vi} ; i = 1, 2, \dots, n\}$$

$$F_v = \max\{F_{vi} ; i = 1, 2, \dots, n\}$$

本稿では式(3.18),(3.22)を満たす領域を動作可能領域と呼び、それ以外の領域を違反領域と呼ぶ。

3.4 終わりに

3章では、マニピュレータの冗長性による動作を冗長姿勢 R_x 、冗長速度 R_v 、冗長加速度 R_a の冗長パラメータで定義し、手先軌道上の位置 s と手先速度 \dot{s} 、冗長姿勢 R_x 、冗長速度 R_v を状態量とし冗長加速度 R_a を入力とした状態方程式を求めた。また、冗長性を状態量とした状態空間上に駆動トルク制限と関節速度制限を射影し、冗長加速度制限と冗長速度制限からなる違反領域を導出した。

マニピュレータは冗長性を有するため、手先位置影響を与えない内部機構で無限の姿勢を取ることができる。この冗長性による手先位置に影響を与えない姿勢を本研究では式 (3.4) のように定義した。同様にマニピュレータの手先速度に影響を与えない冗長性による関節速度を式 (3.14) で導出される正規基底ベクトル \mathbf{U} を用いて式 (3.2) で定義し、マニピュレータの手先加速度に影響を与えない冗長性による関節加速度を正規基底ベクトル \mathbf{U} を用いて式 (3.3) で定義した。状態空間は軌跡上の手先位置 s と冗長姿勢 R_x 、冗長速度 R_v を軸とする空間であるため、状態空間上の経路はこれら状態量の関係式を制約条件として満たす必要がある。冗長性による動作を状態量とした状態方程式は式 (3.7) のように導出される。動作計画を状態空間上での経路探索として行うために、状態空間上に駆動トルク制限と関節速度制限を射影する必要がある。状態空間上に射影された駆動トルク制限は式 (3.18) のように表され、状態空間上に射影された関節速度制限は式 (3.22) のように表される。

第4章 高速動作が困難な区間の経路探索

Contents

4.1	始めに	27
4.2	難区間経路探索の概要	28
4.3	難区間経路探索	29
4.4	評価関数の設計	31
4.5	経路探索例	33
4.6	終わりに	40

4.1 始めに

軌跡追従手先定速作業においては、指令軌跡上の位置によって動作に必要な駆動トルクや関節速度は異なる。そのため、軌跡追従手先定速作業の高速化を考える場合は大きな駆動トルク/関節速度が必要となる箇所では動作計画が重要になる。本章ではそんな手先速度での追従が困難な区間に対するマニピュレータの冗長性を利用した高速動作の計画手法について述べる。

指令軌跡は6章6.3節で述べる軌跡分割手法を用いて、手先速度での追従が困難な難区間、と手先速度での追従が容易な易区間に区間分割される。冗長マニピュレータの動作は状態空間上での1本の経路で表されるため、指令軌跡の難区間における動作は難区間状態空間上で3章で導出した違反領域を回避する経路を探索することで計画される。

本章では、4.2節で高速動作が困難な区間の経路探索手法の概要について説明し、4.3節で高速動作が困難な区間の経路探索手法の詳しい経路探索アルゴリズムについて述べる。4.4節では、経路探索に用いる経路の評価関数の設計を行う。最後に4.5節で高速動作が困難な区間の経路探索結果の一例を示す。

4.2 難区間経路探索の概要

難区間経路探索では難区間に分類された指令軌跡の手先速度 s における状態空間上で、難区間の初期手先位置から違反領域に侵入することなく難区間の終端手先位置まで到達する経路の探索を状態空間上で行なう。マニピュレータの冗長性による動作の計画問題は状態空間上の経路探索問題に帰着することができる。経路探索手法は比留川 [37] の論文にまとめられているようにロードマップ法やセル分割法、ポテンシャル法などの手法があるが、本研究で帰着する経路探索空間は軌跡上の位置 s 軸が時間軸と同等の意味を持ち、さらには状態方程式という経路の制約条件も存在するため、軌跡追従手先定速作業動作の計画は有田らの飛行機などの定速移動体の軌道計画 [38] に近い経路探索手法が必要となる。本研究ではある一つの難区間初端状態から違反領域に侵入せずに難区間の終端まで到達する経路の探索と、難区間経路探索を難区間初端状態についての探索の2つの探索問題として分けて考える。

まずは、ある一つの難区間初端状態からの経路探索の概要について述べる。状態空間上の経路はマニピュレータの状態方程式 (3.7) によって冗長加速度 R_a によって決定する。そのため、難区間初端状態からサンプリングごと冗長加速度制限内の冗長加速度を $R_a(k)$ を選択し、逐次的に経路を求めていく。 k サンプリング目の冗長加速度 $R_a(k)$ は冗長加速度制限の範囲を h 段階に分割した冗長加速度を用いて $s_e[\text{m}]$ 先まで動作させ、その時の h 本の経路の先端の状態を評価関数を用いて評価し、最も高い評価値となった冗長加速度を k サンプリング目の冗長加速度 $R_a(k)$ とする。本研究の難区間経路探索手法は評価関数を用いて難区間の初端状態から順次経路を決定していくため、初端状態から違反領域に侵入することなく難区間の終端まで到達可能な経路が存在したら必ず発見できる手法ではない。また、本難区間経路探索により違反領域に侵入しない経路が発見できない場合は、その初端状態からの経路は存在しないものとした。

次に難区間の初端状態探索の概要について述べる。本研究の動作計画では十分低速な手先速度から経路探索を開始し、徐々に手先速度を高速にしていくことで最も高速な手先速度での動作を求める。この時、手先速度 s における状態空間上の違反領域は手先速度 $s + \delta s$ における違反領域に含まれるものと仮定した。この仮定を置くことにより、手先速

度 s の状態空間上で違反領域に侵入しない経路が発見されない難区間初端状態からは、手先速度 $s + \delta s$ の状態空間上でも経路が発見されないと考えられるため、手先速度が更新されるたびに難区間初端状態に対して全探索を行う必要がなくなる。そのため本研究の難区間初端状態探索では、動作計画の最初の十分低速な手先速度における経路探索時に各難区間初端の広い範囲から経路探索を行い、手先速度を順次速くしていきながら更新前の手先速度において経路が発見された初端状態から経路探索を行う。また、手先速度を微小手先速度 δs 増加させることで違反領域の形状が急激に変化するとは考えにくいいため、手先速度が遅い状態空間上では違反領域に侵入しない経路の存在する難区間初端状態に関する探索の間隔を密にして探索にしなくても、経路の存在する難区間初端状態の分布する範囲を把握することができる。手先速度の増加に伴い経路の存在する難区間初端状態の分布範囲は狭くなるため、発見できた経路の存在する難区間初端状態数が閾値以下となった時に、経路の存在する難区間初端状態付近に対して経路探索を行う難区間初端状態を追加し経路探索を行う初端状態の間隔を密にしていく。ある一定以上難区間初端状態の間隔を密にしても経路の存在する難区間初端状態が発見できない場合は、その手先速度における難区間の経路は存在しないものとする。

4.3 難区間経路探索

難区間経路探索ではある手先速度 s の状態空間上の難区間初端状態から違反領域に侵入することなく難区間の初端手先位置から終端手先位置まで到達する経路の探索を行う。但し、 N_d を経路探索を行う難区間の全サンプル数とし、難区間の範囲は $k = 1, 2, \dots, N_d$ とする。

まずある1つの難区間初端状態からの違反領域に侵入しない経路の計画方法について述べる。状態空間上の経路は式(3.7)の状態方程式より冗長加速度 R_a によって決定するため、サンプリングタイムごとに式(3.18)の冗長加速度制限内から k サンプル目の冗長加速度 $R_a(k)$ を選択して経路の探索を行う。以下に経路探索アルゴリズムについて述べる。

Step.1 計画を行うサンプル番号を $k = 1$ とし、難区間の初端状態を1サンプル目の経路の

状態 $(R_x(1), R_v(1))$ とする.

Step.2 冗長加速度制限を $h \in \mathbb{N}$ 段階に分割し, 各冗長加速度を用いて k サンプル目から $s_e[\text{m}]$ 先までの h 本の経路を予想する. この時, 予想した経路が違反領域に進入する場合は違反領域に進入したサンプル以降の経路は求めない. 経路の予想には次式の冗長加速度を用いる.

$$R_a(j, i) = F_\tau(j, i) + \frac{i}{h-1} (G_\tau(j, i) - F_\tau(j, i)) \quad (4.1)$$

但し, $i = 0, 1, \dots, h-1$, $j = k, k+1, \dots, k + \lceil s_e / (\dot{s} T_s) \rceil$, $G_\tau = G_\tau(s(j), \dot{s}, R_x(j, i), R_v(j, i))$, $F_\tau = F_\tau(s(j), \dot{s}, R_x(j, i), R_v(j, i))$ で, $(R_x(j, i), R_v(j, i))$ は i 段目の冗長加速度を用いて予想された経路の j サンプル目の状態である.

Step.3 予想した h 本の経路の先端の状態を次節で設計する評価関数を用いて評価する. この時, 予想した h 本の経路全てが違反領域内に侵入している場合は, 難区間の初端状態 $(R_x(1), R_v(1))$ からの違反領域に侵入しない経路は存在しないものとして経路探索を終了する.

Step.4 最も高い評価地が得られた経路の k サンプル目の冗長加速度 $R_a(k, i)$ を k サンプル目の冗長加速度 $R_a(k)$ とし, それにより得られる状態 $(R_x(k+1), R_v(k+1))$ を $k+1$ サンプル目の経路の状態とする.

Step.5 $k \neq N_d$ ならば $k = k+1$ として Step.2 に戻る.

次に違反領域に侵入しない経路の存在する難区間初端状態の探索方法について述べる. 本研究の動作計画では難区間の経路探索は優先的に行われるため難区間経路探索は易区間経路探索結果に影響されることなく行われる. そのため, 違反領域に侵入しない経路の存在する難区間初端状態の探索は難区間の経路探索結果だけを考慮して行えばよい. 難区間の初端状態探索は上記した経路探索手法を用いて行われる. 以下に難区間の初端状態探索アルゴリズムについて述べる.

- Step.1 十分低速な手先速度 \dot{s} における状態空間上の難区間初端手先位置に初端状態の間隔を $w = (\delta R_x, \delta R_y)$ として難区間初端状態を均等に配置する。この時、初端状態の間隔 w は比較的広く設定する。
- Step.2 難区間の各初端状態から違反領域に侵入しない経路の探索を行う。
- Step.3 違反領域に侵入しない経路数がゼロだった場合は、初端状態の間隔を $w = w/2$ として Step.2 で経路探索を行った初端状態の R_x - R_y 平面上の 8 近傍に初端状態を追加配置し、追加した初端状態から違反領域に侵入しない経路の探索を行う。この時、初端状態の間隔 w が最小の場合、初端状態の追加配置を行わず手先速度 \dot{s} における違反領域に侵入しない経路は存在しないものとして難区間経路探索を終了する。
- Step.4 違反領域に侵入しない経路数が閾値より少ない場合は、初端状態の間隔を $w = w/2$ として手先速度 \dot{s} の状態空間上において違反領域に侵入しない経路が発見された初端状態の R_x - R_y 平面上の 8 近傍に初端状態を追加配置し、追加した初端状態から違反領域に侵入しない経路の探索を行う。この時、初端状態の間隔 w が最小の場合、初端状態の追加配置を行わない。
- Step.5 手先速度 \dot{s} を $\dot{s} = \dot{s} + \delta \dot{s}$ とし、Step.2 へ戻る。

上記した難区間の経路探索手法と難区間の初端状態探索手法を合わせて難区間経路探索と呼ぶ。

4.4 評価関数の設計

本項では難区間経路探索の経路決定に用いる評価関数の設計を行う。

評価関数の設計指針としては次のことを考える。

- 難区間経路探索では状態空間上の違反領域に侵入しない経路が得られればよく、経路が違反領域をかすめるような経路を通ったとしても何ら不都合はない。また、状態空間上の経路はそれまでの経路が以降の経路計画に影響することはなく、状態 (R_x, R_y)

と冗長加速度入力 R_a によって経路が決定する．そのため評価関数では経路に対してではなく状態空間上の各状態に対して評価を行い，その状態から違反領域に侵入しない経路が多く存在しそうな状態の評価値が高くなるように設計する．

- 状態空間上の違反領域は，駆動トルク制限を状態空間上に射影した冗長加速度制限と，関節速度制限を状態空間上に射影した冗長化速度制限から構成される．そのため，評価関数では単位の異なるこれらの2つの制限を考慮して状態の評価を行い，かつ一方の制限による影響が支配的な評価とならない評価関数にしなければならない．
- 評価関数は難区間経路探索の一部として用いているため，全探索ののような以降の経路を全て計算しその状態を評価するような方式ではなくリーズナブルな計算量の評価関数とする．

これらのことから本研究では，状態が違反領域から離れていればそれだけ以降は違反領域に進入しない経路が得やすいものと仮定し，状態が違反領域から離れているほど評価の高くなる評価関数 E を次式のように設計した．

$$E = E_v + E_\tau \quad (4.2)$$

$$E_v = 1 - \exp(-\min(G_v(j^*, i) - R_v(j^*, i), R_v(j^*, i) - F_v(j^*, i)))$$

$$E_\tau = 1 - \exp(-G_\tau(j^*, i) + F_\tau(j^*, i))$$

但し， $j^* = k + \lceil s_e / (sT_s) \rceil$ である．式(4.2)は状態から冗長速度制限までの距離を評価する E_v 項と状態から冗長化速度制限までの距離を評価する E_τ 項の和で各状態の評価を行う．冗長速度制限は軌道上の手先位置 s と手先速度 \dot{s} ，冗長姿勢 R_x より求められるため， E_v 項では状態から冗長速度の上限・加減までの差が小さい方の値を用いて状態から冗長速度制限までの距離を評価している． E_τ 項では状態から冗長加速度制限までの距離を，その状態の冗長加速度制限の幅を用いて評価している．状態空間上の冗長加速度制限は冗長加速度制限の下限が上限を上回る状態であることを表しており，評価する状態から冗長加速度制限による違反領域との境界までの距離を求めるのは困難である．そのため，本研究では冗長加速度制限の幅が状態空間上の冗長加速度制限により違反領域と状態との距離

に比例すると考え、 E_τ のような関数を用いて状態から冗長加速度制限までの距離を評価した。また、冗長加速度と冗長速度では単位が異なるため、単純に比較し評価することが出来ない。そのため本研究では式(4.2)のように、状態と冗長加速度制限、冗長速度制限までの距離を $1 - e^{-\alpha}$ の形の関数を用いて評価することで、 E_v もしくは E_τ の一方が支配的になることを回避しつつ違反領域に近い状態は評価値が下がる評価関数となっている。

4.5 経路探索例

本節では Fig.4.1 に示すモデルを有する回転関節型水平3関節冗長マニピュレータに対して難区間経路探索を手先要した経路探索の一例を示す。

手先軌跡指令は 270[deg] 回転する半径 5[mm] の円弧軌跡と、その接線方向に伸びる直線軌跡を接続した Fig.4.2 のような手先軌跡を用いる。マニピュレータのパラメータは文献 [39, 40] のような手法を用いて実機より同定可能であり、本論文ではこれらの手法を用いて同定した Tab.4.1 に示す実験機のパラメータを用いた。ここで回転関節型平面マニピュレータの関節変位の和はマニピュレータの手先姿勢となることから、冗長姿勢 R_x は $\Sigma = [1, 1, 1]^T$ と定めた。難区間経路探索は全手先軌跡を難区間として、冗長加速度分割段階数 $h = 11$ 段階、経路を予想する距離 $s_e = 3$ [mm]、初端状態の探索範囲を $(0 \leq R_x \leq 2\pi)$, $(-20 \leq R_x \leq 20)$ の各9分割として経路探索を行ない、最後に得られた手先速度 $\dot{s} = 0.4$ [m/s] における状態空間上の経路を難区間経路探索結果の一例として示す。

Fig.4.3 に計画された経路の冗長姿勢 R_x と冗長速度 R_v 、冗長加速度 R_a を、Fig.4.4 に状態空間上の違反領域と計画された経路を示す。ここで、Fig.4.4 に示す状態空間は計画された経路が見やすいように冗長姿勢 R_x 軸の描画範囲を状態空間-A と状態空間-B で変更している。Fig.4.5 と Fig.4.6 に計画された経路によるマニピュレータ動作の駆動トルクと関節速度を示す。

Fig.4.3 では各冗長パラメータが制限の範囲内で計画されている。また、冗長加速度 R_a は 0.015[s] から 0.04[s] 付近で常に冗長加速度制限の下限 F_τ の値を取っており、0.03[s] 付近では冗長加速度制限の上限 G_τ と下限 F_τ が等しい値となっている。そのため、Fig.4.5 に示す冗長加速度制限の基となっている駆動トルクでは、 τ_1 が 0.015[s] から 0.04[s] 付近で

駆動トルク制限値での動作となっており、また冗長加速度制限の上限 G_{τ} と下限 F_{τ} が等しくなる 0.03[s] 付近では τ_1 と τ_2 が共に駆動トルク制限値での動作となっている。これは冗長加速度制限が駆動トルク制限より生じていることからこのような結果となっている。また、冗長加速度制限の上限と下限が等しくなる 0.03[s] 付近での駆動トルクが全て制限値での動作となっていないのは、冗長加速度制限は式 (3.18) のように各関節の駆動トルク制限による冗長加速度制限の最小と最大を取っているため、冗長加速度制限の幅がゼロになるのはある関節の上限 G_{τ_i} と他の関節の下限 F_{τ_j} が一致する時だからである。そのためマニピュレータの関節数が増加しても、冗長加速度制限の上限と下限が一致する時は2つの関節の駆動トルクが駆動トルク制限値となっている時である。Fig.4.5 の τ_1 は 0.05[s] 付近で振動的な波形となっている。これは難区間経路探索の冗長加速度 R_a は式 (4.1) のように離散的な値から選択して計画を行っているためである。Fig.4.3 の冗長速度 R_v が 0.03[s] 付近で冗長速度制限に近い値となっている。そのため、Fig.4.6 に示す関節速度も 0.03[s] 付近で関節速度制限に近い値となっている。これらの結果より、難区間経路探索を行うことで状態空間上の違反領域を回避するような経路が計画されていることがわかる。

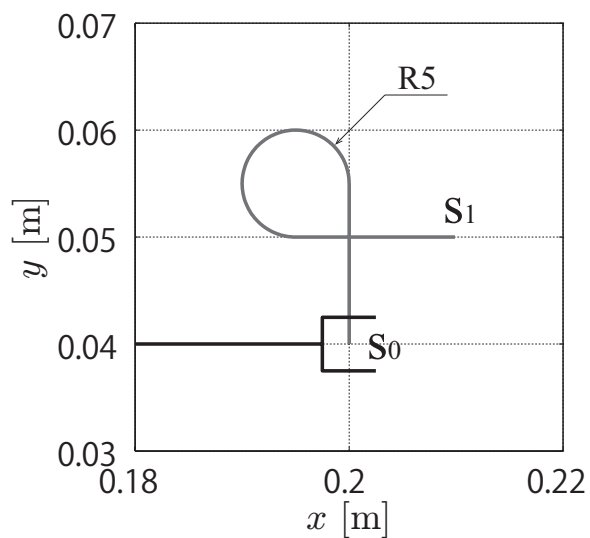
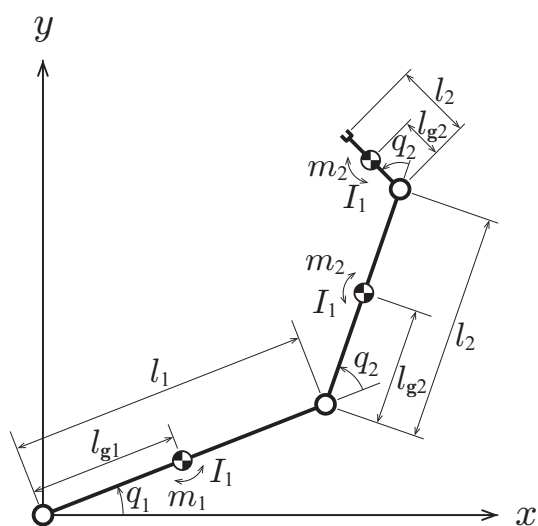


Fig. 4.1: 水平3関節マニピュレータモデル Fig. 4.2: 難区間経路探索を行なう手先軌跡

Tab. 4.1: 水平3関節冗長マニピュレータのパラメータ

	Link1	Link2	Link3	unit
Length of links	200	150	50	[mm]
Mass of links	1235	380	110	[g]
Length to center of gravity	-27.88	0.24	2.81	[mm]
Inertia of links	793.5	88.2	1.7	$\times 10^{-5} [kgm^2]$
Joint torque limits	0.90	0.55	0.31	[Nm]
Joint velocity limits	8.4	8.4	14.0	[rad/s]
Vector Σ			1 1 1	

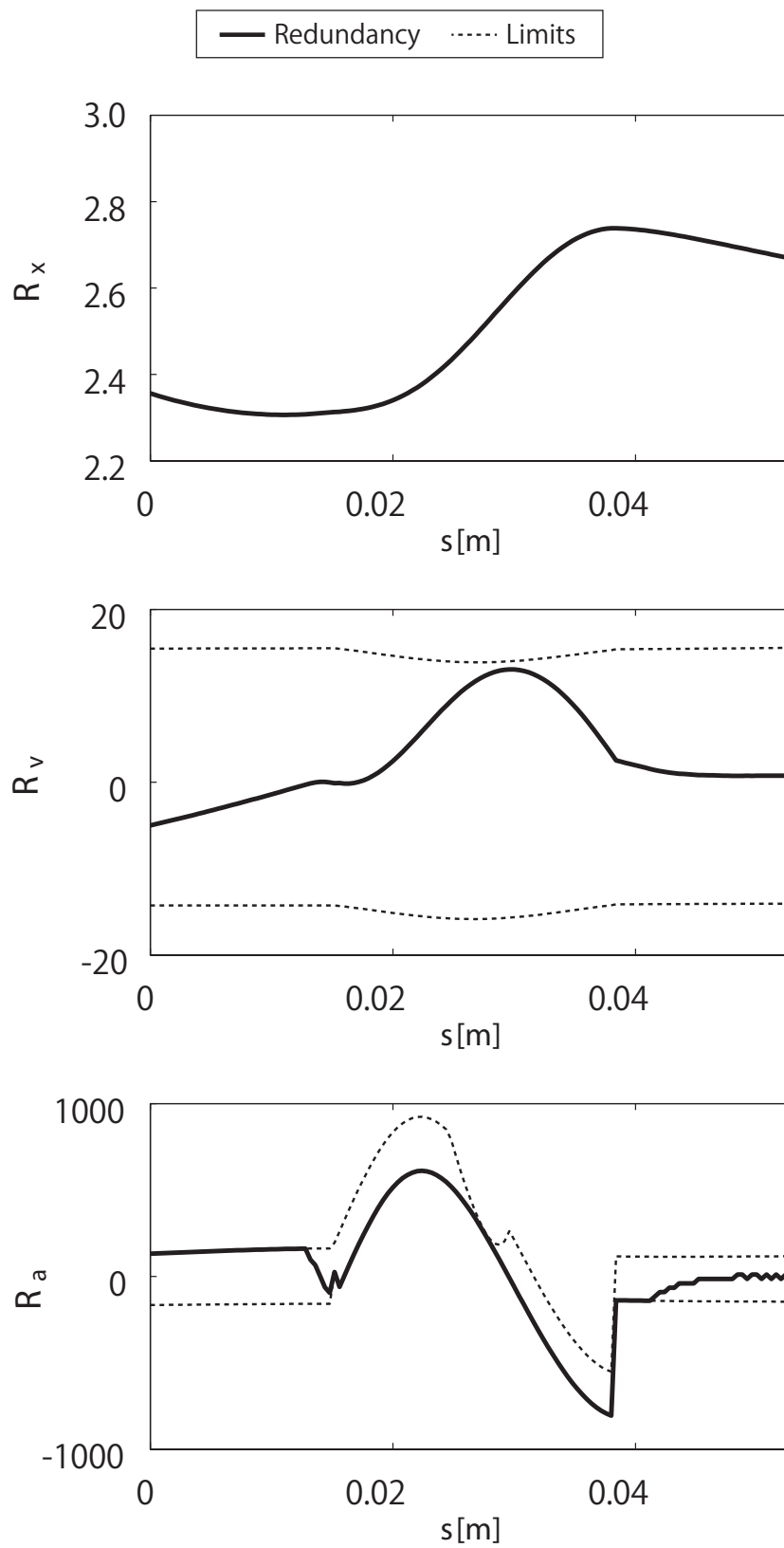


Fig. 4.3: 動作計画により得られた冗長パラメータ

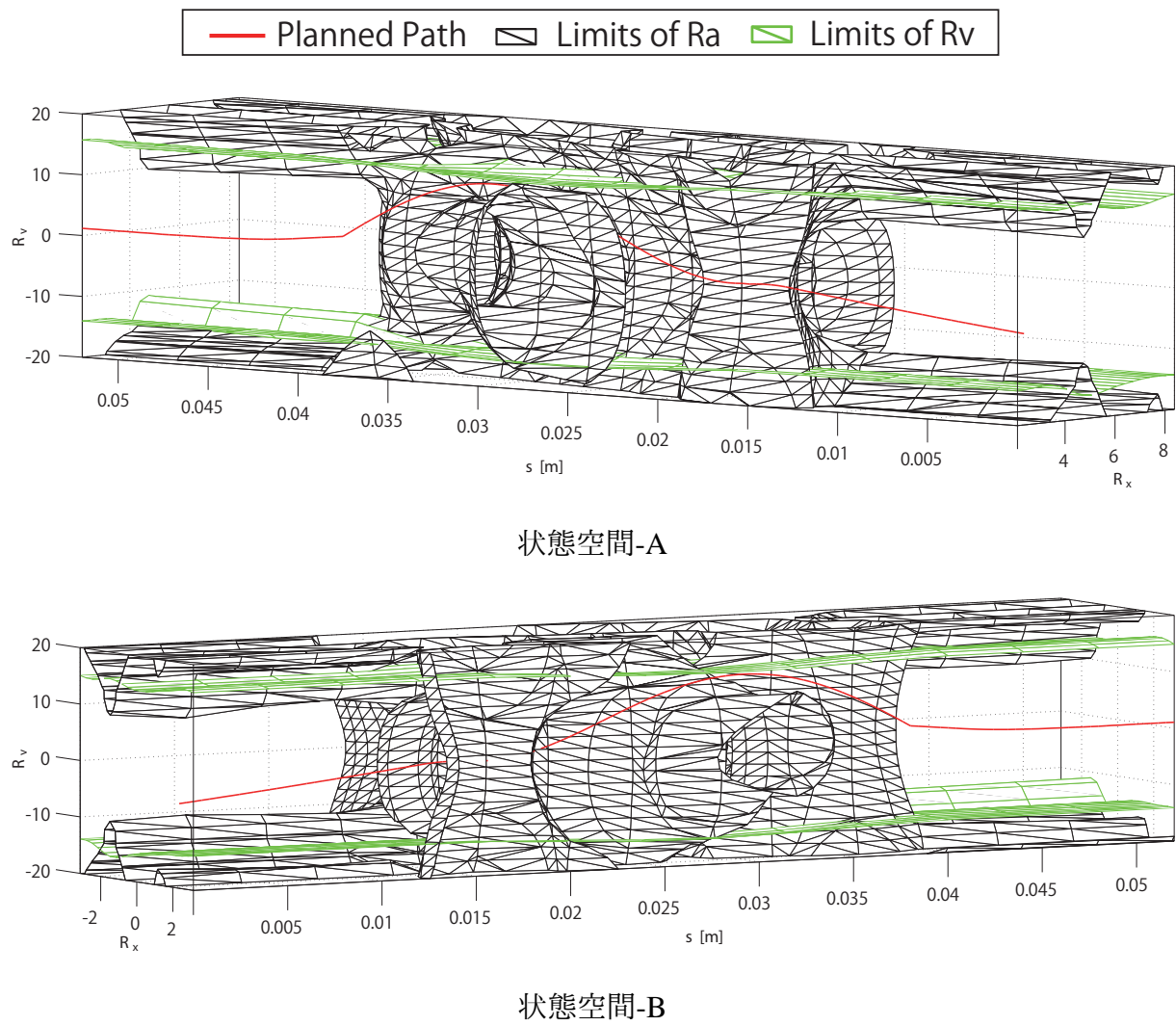


Fig. 4.4: 状態空間上の経路と違反領域

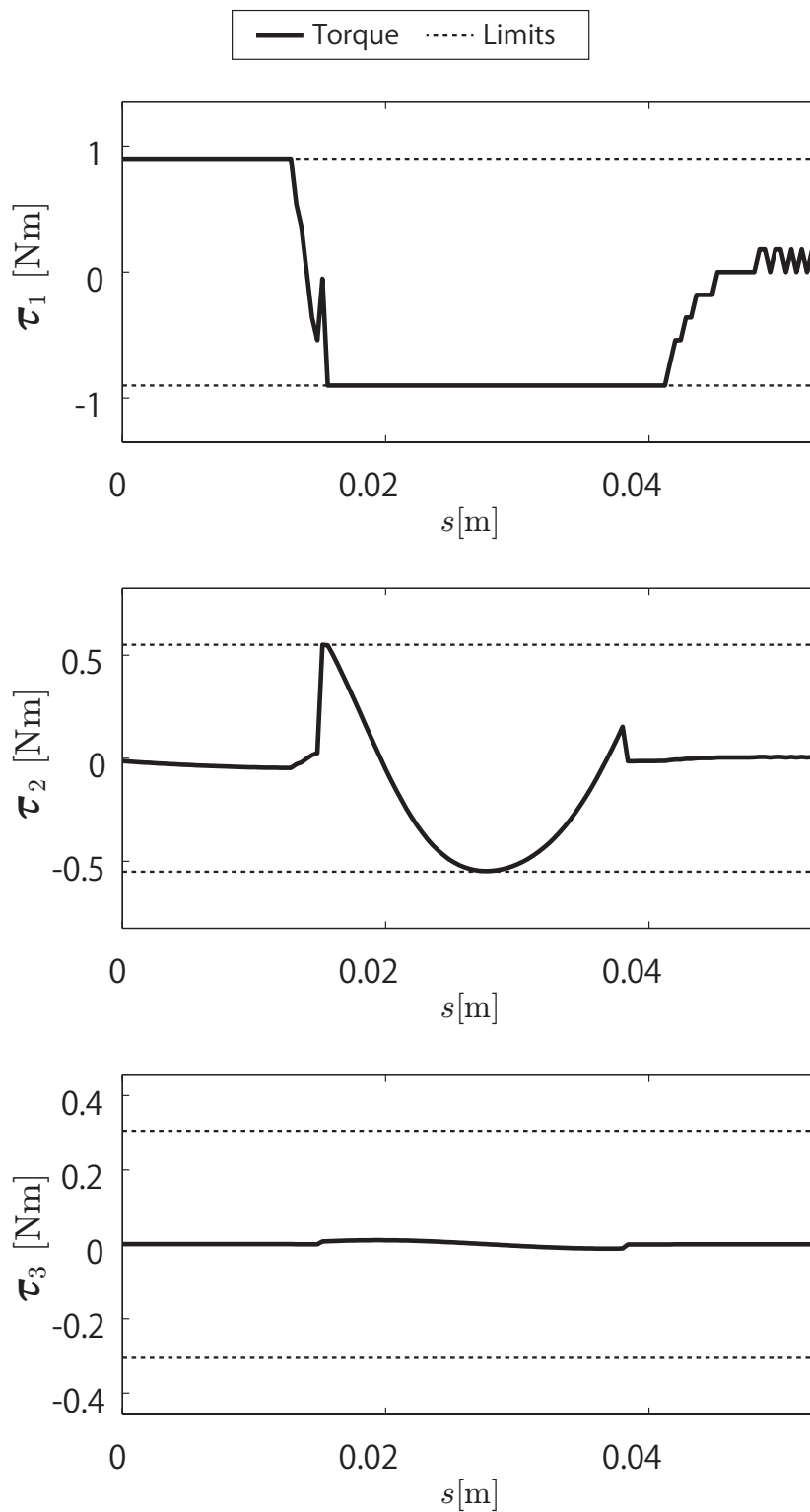


Fig. 4.5: 計画された動作の駆動トルク

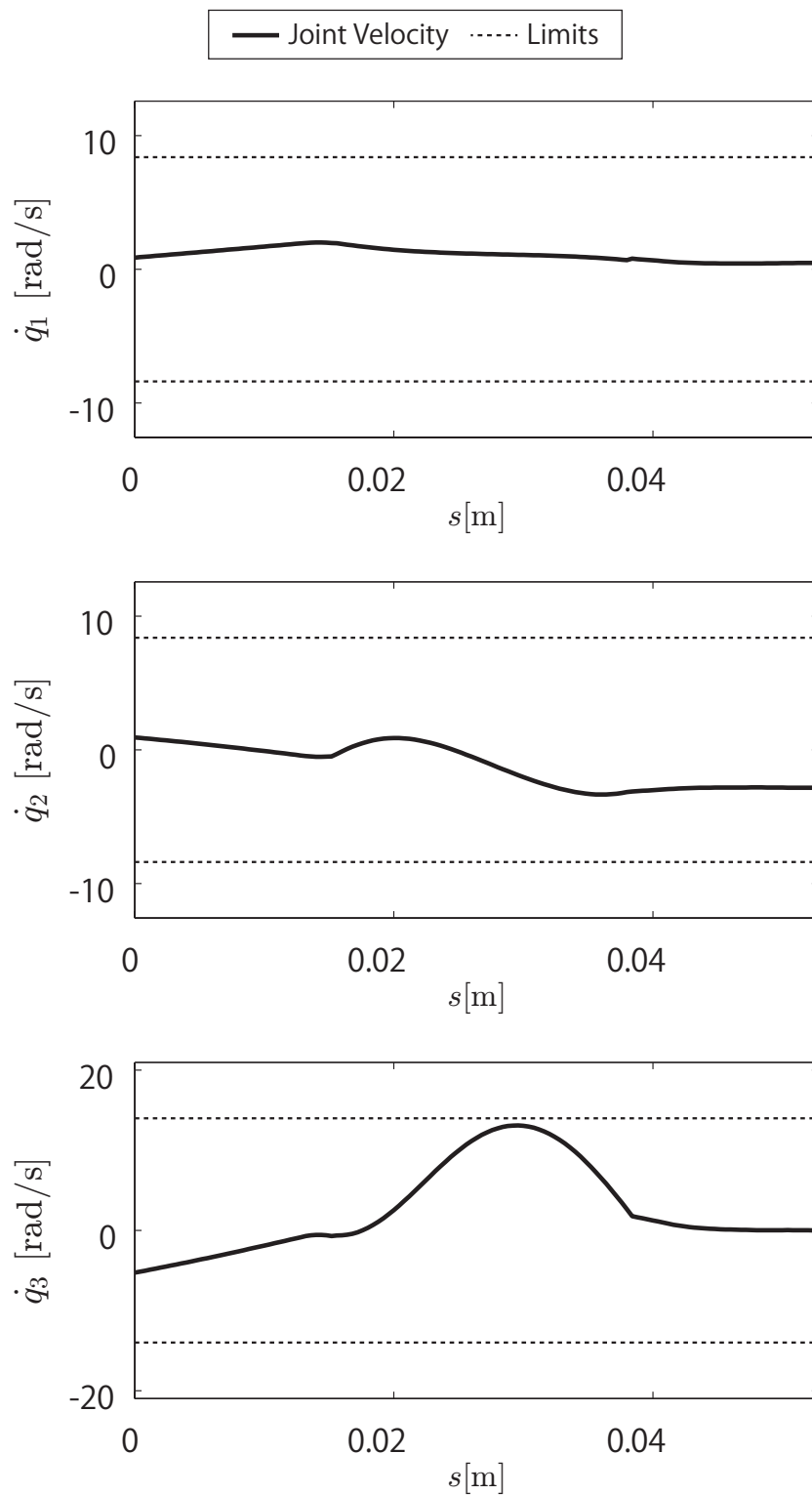


Fig. 4.6: 計画された動作の関節速度

4.6 終わりに

4章では、手先軌跡の高手先速度動作が困難な区間に対する状態空間上の経路探索手法について述べた後、水平3関節冗長マニピュレータに対して全手先軌跡を難区間とした動作計画例を示した。

高手先速度動作が困難な区間の動作計画手法である難区間経路探索は、ある1つの難区間初端状態からの経路探索と難区間の初端状態に関する探索の2手法から成り立つ。ある1つの難区間初端状態からの経路探索は、難区間初端状態 (R_x, R_y) から、サンプリングタイムごとに計画された経路の先端から h 個の冗長加速度入力を用いた場合の経路を $s_e[m]$ 先まで予想し、予想された経路を式(4.2)の評価関数として逐次的に求めていくことで、違反領域に侵入しない経路を探索する。難区間の初端状態に関する探索は、手先速度を増加させながら違反領域に進入することなく難区間終端手先位置まで到達する経路の存在する難区間初端状態を絞り込んでいくことで、高手先速度における違反領域に進入することなく難区間終端手先位置まで到達可能な初端状態の存在する範囲を求める。これによりサンプリングごとに難区間初端手先位置の $R_x - R_y$ 平面上の全ての状態に対して経路探索を行うことなく違反領域に進入することなく難区間終端手先位置まで到達可能な初端状態を求めることができる。

このように難区間経路探索は十分低速な手先速度から、違反領域に侵入しない経路の存在する初端状態の範囲を絞りこみながら、高手先速度における状態空間上の難区間初端状態と違反領域に侵入しない経路を求める。

第5章 高速動作が容易な区間の経路探索

Contents

5.1	始めに	42
5.2	易区間経路探索の概要	43
5.3	易区間経路探索	43
5.3.1	切換面の生成	43
5.3.2	駆動トルク最小となる冗長加速度の導出	45
5.3.3	易区間経路探索アルゴリズム	46
5.4	易区間経路探索例	53
5.5	終わりに	59

5.1 始めに

軌跡追従手先定速作業の高速化問題において重要なのは高手先速度動作が困難な区間の動作計画で、高手先速度動作が困難な区間の間に存在する高手先速度動作が容易な区間では駆動トルク制限や関節速度制限に比較的余裕がある。本章では高手先速度動作が困難な区間での動作に自由度を持たせるため、高手先速度動作が困難な区間の予備動作となる高手先速度動作が困難な区間で計画された経路を繋ぐ経路を探索する易区間経路探索手法について述べる。

本章では5.2節で易区間経路探索手法の概要について説明する。5.3節では、まず易区間経路探索手法において重要な考え方である切換面の生成方法について述べ、次に切換面を用いた易区間の初端状態と終端状態を繋ぐ易区間経路探索アルゴリズムについて述べる。最後に、5.4節で切換面を用いた易区間経路探索結果の一例を示す。

5.2 易区間経路探索の概要

易区間経路探索は前後の難区間経路探索結果により与えられる易区間の初端状態 (R_{x0}, R_{v0}) と、易区間の終端状態 (R_{x1}, R_{v1}) を繋ぐ経路の探索を状態空間上で行う。動作計画では前後の難区間で計画された経路の各組み合わせに対して易区間経路探索を行うが、本章では1つの易区間初端状態と1つの易区間終端状態を繋ぐ易区間経路探索手法について述べる。状態空間上の経路はマニピュレータの状態方程式 (3.7) を満たす必要があり、初端状態と終端状態を任意の経路で繋ぐことは出来ない。そのため、本研究では

可変構造系と呼ばれる制御系 [41] で用いられる切換線という概念を応用した切換面という終端状態に到達可能な面を考える。切換面とは終端状態から s 軸とは逆方向に向かって最大冗長加速度 G_τ と最小冗長加速度 F_τ を用いた経路を生成し、その経路上の各点から駆動トルク最小となる冗長加速度を用いた経路を生成してできる面で、切換面上からの経路は最大冗長加速度、最小冗長加速度、駆動トルク最小となる冗長加速度を用いて切換面上を辿ることで終端状態までの経路を得ることができる。易区間経路探索では、この切換面を用いて初端状態から切換面までの経路を探索することで初端状態と終端状態に繋ぐ経路の探索を行う。

5.3 易区間経路探索

5.3.1 切換面の生成

切換面とはその面上からは必ず終端状態に到達可能な曲面のことで、最大冗長加速度による経路と最小冗長加速度による経路、駆動トルク最小となる冗長加速度による経路から生成される。

切換面は以下の手順で生成される。また切換面は Fig.5.1 のような離散的な点群によって構成される。

Step.1 Fig.5.1-a,b のように、易区間の終端状態から初期手先位置に向かって最大冗長加速度 $R_a = G_\tau$ を用いて得られる経路（以降、加速線）と最小冗長加速度 $R_a = F_\tau$ を用

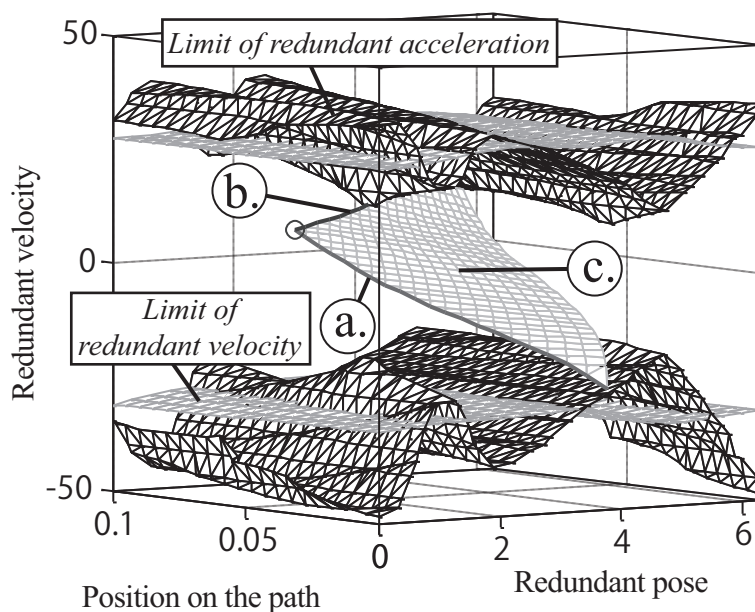


Fig. 5.1: Switching surface on the state space

いて得られる経路（以降，減速線）を生成する．但し，経路が違反領域に侵入した場合は進入した経路の生成はそこで終了する．この時，状態空間上の経路はバックトレースとなり，式 (5.11) では経路を求めることが出来ない．そのため，次式を用いて終端状態からの経路を求める．

$$\begin{bmatrix} R_x(k-1) \\ R_v(k-1) \end{bmatrix} = \begin{bmatrix} R_x(k) \\ R_v(k) \end{bmatrix} - \begin{bmatrix} \dot{R}_x(k) \\ \dot{R}_v(k) \end{bmatrix} T_s \quad (5.1)$$

$$\begin{bmatrix} \dot{R}_x(k) \\ \dot{R}_v(k) \end{bmatrix} = \begin{bmatrix} 0 & \Sigma U(k) \\ 0 & \dot{U}^T(k) U(k) \end{bmatrix} \begin{bmatrix} R_x(k) \\ R_v(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} R_a(k) + \begin{bmatrix} \Sigma \\ \dot{U}^T(k) \end{bmatrix} J^+(k) f'(k) s$$

Step.2 Fig.5.1-c のように，生成された加速線と減速線上の全ての点から駆動トルク最小となる冗長加速度を用いて経路を生成する．但し，経路が違反領域に侵入した場合は進入した経路の生成はそこで終了する．この駆動トルク最小となる冗長加速度については次項で述べる．

5.3.2 駆動トルク最小となる冗長加速度の導出

マニピュレータの動力学方程式を冗長加速度の式とした式 (3.15) を再度示す.

$$\boldsymbol{\tau} = \mathbf{c}_1 R_a + \mathbf{c}_2 \quad (5.2)$$

$$\mathbf{c}_1 = \mathbf{M}\mathbf{U}$$

$$\mathbf{c}_2 = \mathbf{M}\mathbf{J}^+(\mathbf{f}''\dot{s}^2 - \mathbf{J}\mathbf{J}^+\mathbf{f}'\dot{s} - \mathbf{J}\mathbf{U}\mathbf{R}_v) + \mathbf{H} + \mathbf{g} + \mathbf{d}$$

マニピュレータには各関節に駆動トルク制限 Ω_τ が存在するため Ω_τ で正規化する必要がある. そのため式 (5.2) は次式のようになる.

$$\mathbf{W}\boldsymbol{\tau} = \mathbf{W}\mathbf{c}_1 R_a + \mathbf{W}\mathbf{c}_2 \quad (5.3)$$

但し,

$$\mathbf{W} = \begin{bmatrix} \frac{1}{\tau_{lim1}} & & & 0 \\ & \frac{1}{\tau_{lim2}} & & \\ & & \ddots & \\ 0 & & & \frac{1}{\tau_{limn}} \end{bmatrix} \quad (5.4)$$

この重み付き駆動トルク $\mathbf{W}\boldsymbol{\tau}$ は Fig.5.2 のように, $\mathbf{W}\mathbf{c}_1$ ベクトルを基底ベクトルとした $\mathbf{W}\mathbf{c}_1 R_a$ ベクトルと, 冗長加速度 R_a とは独立した $\mathbf{W}\mathbf{c}_2$ ベクトルの線形和である. そのため, 重み付き駆動トルク $\mathbf{W}\boldsymbol{\tau}$ が最小となるのは $\mathbf{W}\boldsymbol{\tau}$ ベクトルと $\mathbf{W}\mathbf{c}_1 R_a$ ベクトルが直交するときである. 従って,

$$(\mathbf{W}\mathbf{c}_1 R_a)^T \mathbf{W}\boldsymbol{\tau} = 0 \quad (5.5)$$

$$R_a \mathbf{c}_1^T \mathbf{W}^T \mathbf{W} (\mathbf{c}_1 R_a + \mathbf{c}_2) = 0 \quad (5.6)$$

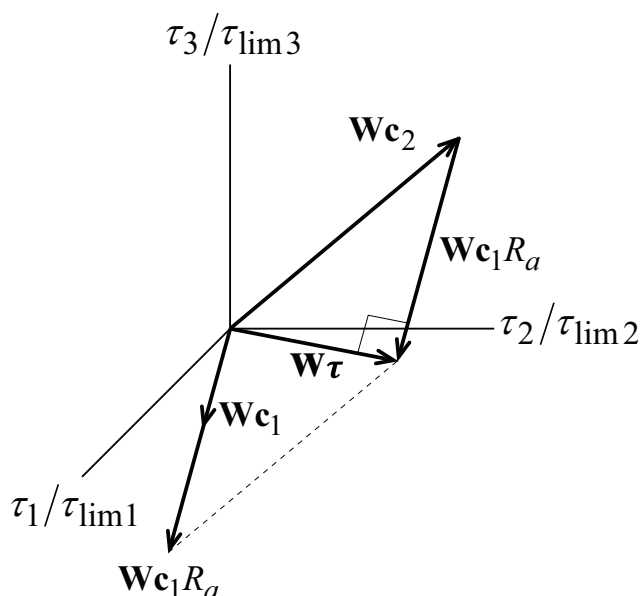


Fig. 5.2: 駆動トルク最小となる冗長加速度

但し \mathbf{W}, \mathbf{M} は対称行列であるため，駆動トルクが最小となる冗長加速度は次式のように導出される．

$$R_a = -(\mathbf{c}_1^T \mathbf{W}^2 \mathbf{c}_1)^{-1} \mathbf{c}_1^T \mathbf{W}^2 \mathbf{c}_2 \quad (5.7)$$

5.3.3 易区間経路探索アルゴリズム

易区間経路探索は次のような条件のもと，与えられた初端状態と終端状態を繋ぐ易区間の経路探索を行う．

- 手先速度 \dot{s} と易区間の初端状態 $[R_{x0}, R_{v0}]^T$ ，終端状態 $[R_{x1}, R_{v1}]^T$ が与えられる
- 経路探索を行なう易区間の手先目標軌道 $\mathbf{r}, \dot{\mathbf{r}}, \ddot{\mathbf{r}}$ が与えられる

易区間経路探索は3つのフェーズに分かれており，易区間の始めから終わりまでの全てのサンプル $k = 1, 2, \dots, N_s$ に対して以下の操作を行なう．フェーズ1では，まず終端状態から切換面を生成し，その後切換面と k サンプリング目の経路 $[R_x(k), R_v(k)]^T$ の位置関係を基に切換面までの経路を計画する．一般的に Fig.5.1 に示すように切換面は曲面によっ

て構成されるため、位置関係の比較が少々困難である。そのため、本研究では k サンプル目の経路 $[R_x(k), R_v(k)]^T$ に近い切換面を構成する2点 $[R_{xs1}, R_{vs1}]^T, [R_{xs2}, R_{vs2}]^T$ より、 k サンプル目の切換面を1次関数で近似した近似切換面 f_α で表す。 k サンプル目の近似切換面は式(5.8)のようになる。

$$f_\alpha(k) = \alpha(R_x - R_{xs1}) + R_{vs1} - R_v \quad (5.8)$$

$$\alpha = \frac{R_{vs1} - R_{vs2}}{R_{xs1} - R_{xs2}}$$

易区間経路探索では経路と切換面の位置関係によって経路を計画していくため、必ずしも切換面を構成する全ての点が必要になるわけではない。そのため、近似切換面を構成するのに必要な2点 $[R_{xs1}, R_{vs1}]^T, [R_{xs2}, R_{vs2}]^T$ を探索するとき同時に切換面の必要な部分だけを計算する。経路に近い切換面を構成する点 $[R_{xs1}, R_{vs1}]^T$ は2分探索を用いて探索する。

2分探索を行なうために切換面を Fig.5.3 のような配列に格納する。この配列は $2N_s - 1 \times N_s \times 2$ の配列で、配列の幅 k は k サンプル目の値であることを表し、配列の奥行きにはそれぞれの状態の冗長姿勢 R_{xik} と冗長速度 R_{vik} を表し、Fig.5.3 のように N_s 行 N_s 列目に終端状態が位置し、そこから斜めに加速線と減速線の状態を格納する。加速線と減速線から駆動トルク最小となる経路により構成される切換面の各状態は、駆動トルク最小となる経路の始点となる加速線・減速線の状態が格納された行に Fig.5.3 のように格納される。このような配列に格納された切換面を用いて経路に近い切換面を構成する点 $[R_{xs1}, R_{vs1}]^T$ の探索を行なう。2分探索は次の手順で行なう。

-
- Step.1 終端状態 $[R_{x1}, R_{v1}]^T$ から加速線と減速線を求め、Fig.5.3 のように配列に格納する。次に減速線の先端が格納された行番号を i_{MAX} 、減速線先端の冗長速度を R_{vMAX} とし、加速線の先端が格納された行番号を i_{MIN} 、加速線先端の冗長速度を R_{vMIN} とする。
- Step.2 i_{MAX} と i_{MIN} の中央値を求めて i_{MID} とする。この時、中央値が整数でない場合は整数に丸める。また、 i_{MID} 行の切換面が計算されていなければ、経路を計画するサンプリング k までの切換面を計算して Fig.5.3 のように格納する。

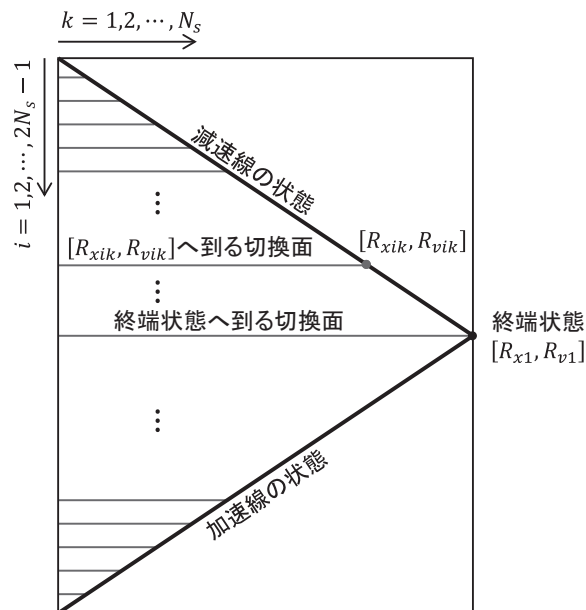


Fig. 5.3: 切換面の配列への格納方法

Step.3 i_{MID} 行 k 列目の冗長速度 $R_{v_{i_{MID}k}}$ を $R_{v_{MID}}$ とする. この時, 切換面を構成する駆動トルク最小となる経路が途中で違反領域に侵入してしまい k サンプル目まで経路を計算できていない場合は, 計算された経路の先端の冗長速度を $R_{v_{MID}}$ とする.

Step.4 計画している経路先端の冗長速度 $R_v(k)$ と $R_{v_{MID}}$ を比較し, $R_{v_{MID}} < R_v(k) < R_{v_{MAX}}$ なら i_{MID} を i_{MIN} に, $R_{v_{MID}}$ を $R_{v_{MIN}}$ にする. $R_{v_{MIN}} < R_v(k) < R_{v_{MID}}$ なら i_{MID} を i_{MAX} に, $R_{v_{MID}}$ を $R_{v_{MAX}}$ にする. $R_v(k) < R_{v_{MIN}}$, $R_{v_{MAX}} < R_v(k)$ なら経路は切換面とは交差ししないものとして易区間経路探索を終了する.

Step.5 $i_{MAX} - i_{MIN} = 1$ なら, 経路に最も近い切換面を構成する点を $[R_{x_{i_{MID}k}}, R_{v_{i_{MID}k}}]^T$, $[R_{x_{i_{MID}+1,k}}, R_{v_{i_{MID}+1,k}}]^T$ と $[R_{x_{i_{MID}-1,k}}, R_{v_{i_{MID}-1,k}}]^T$ から冗長速度が $R_v(k)$ に近い方を経路に 2 番目に近い切換面を構成する点として探索終了. $i_{MAX} - i_{MIN} \neq 1$ なら Step.2 へ戻る.

初端状態から近似切換面までの経路は式 (5.9) より k サンプル目の冗長加速度 $R_a(k)$ を決定し, 1 サンプリングの経路を求める.

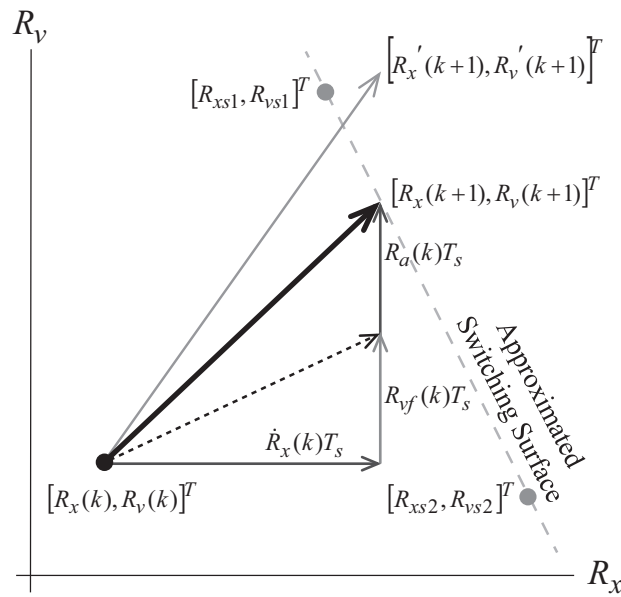


Fig. 5.4: 経路を近似切換面上に位置させるための冗長加速度

$$R_a(k) = \begin{cases} G_\tau(k) & (f_\alpha(k) < 0) \\ F_\tau(k) & (f_\alpha(k) < 0) \end{cases} \quad (5.9)$$

次式に式 (3.8) の状態更新則を再記する.

$$\begin{bmatrix} R_x(k+1) \\ R_v(k+1) \end{bmatrix} = \begin{bmatrix} R_x(k) \\ R_v(k) \end{bmatrix} + \begin{bmatrix} \dot{R}_x(k) \\ \dot{R}_v(k) \end{bmatrix} T_s \quad (5.10)$$

$$\begin{bmatrix} \dot{R}_x(k) \\ \dot{R}_v(k) \end{bmatrix} = \begin{bmatrix} 0 & \Sigma \mathbf{U}(k) \\ 0 & \dot{\mathbf{U}}^T(k) \mathbf{U}(k) \end{bmatrix} \begin{bmatrix} R_x(k) \\ R_v(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} R_a(k) + \begin{bmatrix} \Sigma \\ \dot{\mathbf{U}}^T(k) \end{bmatrix} \mathbf{J}^+(k) \mathbf{f}'(k) \dot{s}$$

このように初端状態からの経路を求めていき、経路が切換面上に達したらフェース2に移行する。このとき、経路はサンプリングタイムごとに離散的に計算しているため、Fig.5.4のように経路が $[R_x(k), R_v(k)]^T$ から $[R'_x(k+1), R'_v(k+1)]^T$ に更新され、近似切換面上への経路ではなく近似切換面を行き過ぎる経路が計算されることがほとんどである。そのため、経路が近似切換面を行き過ぎる場合は経路が近似切換面上に来る冗長加速度を用いて経路を計画する。

式(5.10)より、 $k+1$ サンプル目の冗長姿勢 $R_x(k+1)$ は k サンプル目の冗長速度 $R_v(k)$ によって決定し、 k サンプル目の冗長加速度 $R_a(k)$ とは独立である。そのため Fig.5.4 のように $k+1$ サンプル目の冗長姿勢 $R_x(k+1)$ は $\dot{R}_x(k)T_x$ により既に決定しており、 $k+1$ サンプル目の冗長速度 $R_v(k+1)$ が $R_x(k+1)$ における近似切換面の冗長速度と一致する冗長加速度とすればよい。

$R_x(k+1)$ における近似切換面の冗長速度は次式のように表される。

$$R_v = \alpha(R_x(k+1) - R_{xs1}) + R_{vs1} \quad (5.11)$$

但し、 $[R_{xs1}, R_{vs1}]^T$ は経路に最も近い切換面を構成する点、 $[R_{xs2}, R_{vs2}]^T$ は経路に2番目に近い切換面を構成する点、 $\alpha = (R_{vs1} - R_{vs2}) / (R_{xs1} - R_{xs2})$ である。

式(5.11)と式(5.10)の $R_v(k+1)$ が一致すると経路が近似切換面上に位置することになるため、 k サンプル目の冗長加速度 $R_a(k)$ は次のように求められる。

$$R_v(k) + \{R_a(k) + \dot{U}^T(k)U(k)R_v(k) + \dot{U}^T(k)J^+(k)f'(k)\dot{s}\}T_s = \alpha(R_x(k+1) - R_{xs1}) + R_{vs1} \quad (5.12)$$

$$R_a(k) = \{\alpha(R_x(k+1) - R_{xs1}) + R_{vs1} - R_v(k) - R_{vf}(k)\} / T_s \quad (5.13)$$

但し、

$$R_{vf}(k) = \dot{U}^T(k)\{J^+(k)f'(k)\dot{s} + U(k)R_v(k)\}T_s$$

また、フェーズ1の操作により切換面に到達する経路が得られない場合はそこで易区間経路探索を終了する。

フェーズ2では、切換面上を辿ることで加速線もしくは減速線、終端状態までの経路を計画する。このとき駆動トルク最小となる冗長加速度を用いて計算することで切換面上を辿る経路が得られるはずだが、経路は連続した曲面の切換面ではなく1次関数で近似した近似切換面上にあるため、駆動トルク最小となる冗長加速度を用いて経路を計算すると切換面から離れていくことが予想される。そのため、フェーズ2では切換面上を辿る経路

を求めるために式 (5.13) で導出した近似切換面上に到達する冗長加速度を用いることで、加速線もしくは減速線、終端状態への経路を計画する。経路が加速線もしくは減速線に到達する、もしくは行き過ぎたらフェーズ3へ移行する。

フェーズ3では加速線もしくは減速線を辿り終端地点へ到達する経路を求める。経路が加速線もしくは減速線上に位置している場合は最大冗長加速度もしくは最小冗長加速度を用いて加速線もしくは減速線を辿る経路を求める。しかし、ほとんどの場合はフェーズ2で加速線や減速線を行き過ぎた経路が計画される。そのため、加速線や減速線を越える1サンプル前から経路計画をやり直し、最大冗長加速度もしくは最小冗長加速度を用いた経路を計画する。このとき、冗長加速度は次式のように終端状態と経路の位置関係を比較することで決定する。

$$R_a(k) = \begin{cases} G_\tau(k) & (R_{v1} > R_v(k)) \\ F_\tau(k) & (R_{v1} < R_v(k)) \\ 0 & (R_{v1} = R_v(k)) \end{cases} \quad (5.14)$$

易区間経路探索のフローチャートを Fig.5.5 に示す。

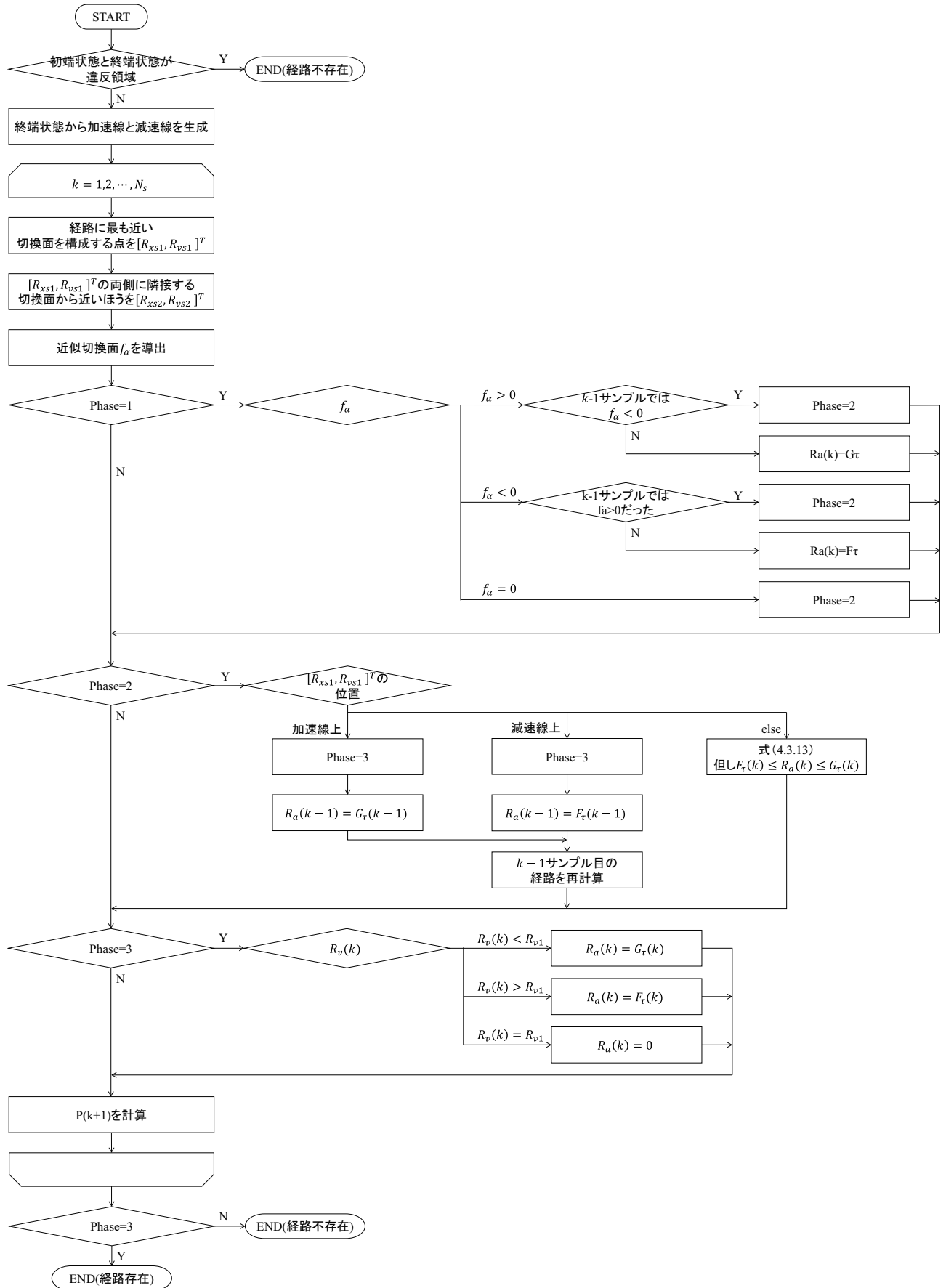


Fig. 5.5: 易区間経路探索手法のフローチャート

5.4 易区間経路探索例

本節では易区間経路探索による計画結果の一例を示す。

易区間経路探索は Fig.4.1 に示すモデルを有する水平3関節冗長マニピュレータに対して、終端状態 $(R_{x1}, R_{v1}) = (\pi, 0)$ と初端状態 $(R_{x0}, R_{v0}) = (\pi/2, 10), (\pi/2, -10), (3\pi/2, 10), (3\pi/2, -10)$ を繋ぐ経路の探索を行なう。この時、冗長マニピュレータの手先は Fig.5.6 に示す軌跡を追従するものとした。また冗長マニピュレータのパラメータは Tab.4.1 に示す値を用いた。Fig.5.7-5.10 にそれぞれの状態を繋ぐ易区間経路探索結果を示す。

これらの易区間経路探索結果より、Fig.5.7, 5.8, 5.9 では切換面を辿ることで初端状態と終端状態を繋ぐ経路が計画されている。Fig.5.10 では初端状態から切換面までの経路の途中で違反領域に侵入し、初状態と終端状態を繋ぐ経路が計画されずに易区間経路探索が終了している。

Fig.5.7 の初端状態 $(R_{x0}, R_{v0}) = (\pi/2, 10)$ からの易区間経路探索では $s = 0.003[m]$ 地点まで $R_a = F_\tau$ で初端状態から切換面上までの経路が計画され、 $s = 0.058[m]$ 地点から減速線を辿ることで終端状態への経路を計画している。また、フェーズ2の始まりである $s = 0.003[m]$ 地点付近では、近似切換面上に経路が位置するように冗長加速度を調整しているためピークのような波形が現れている。

Fig.5.8 の初端状態 $(R_{x0}, R_{v0}) = (\pi/2, -10)$ からの経路探索でも同様に $s = 0.007[m]$ 地点まで $R_a = G_\tau$ で初端状態から切換面上までの経路が計画され、 $s = 0.58[m]$ 地点から減速線を辿り終端状態へ達する経路となっている。Fig.5.8 の冗長加速度入力は $s = 0.007[m]$ から $s = 0.01[m]$ にかけて振動的な冗長加速度入力が計画されている。これは、 $s = 0.009[m]$ 地点の $R_a = G_\tau$ となっている地点で、経路から2番目に近い切換面を構成する点が最も近い経路の下側の点から最も近い経路の上側の点に切り替わり、その結果近似切換面が変化しこのような波形となっている。しかし、フェーズ2では常に経路が近似切換面上に来るように冗長加速度入力を計算しているため、このような近似切換面の変化があったとしても問題なく切換面上を辿る経路が計画される。

Fig.5.9 の初端状態 $(R_{x0}, R_{v0}) = (3\pi/2, 10)$ からの経路探索は、 $s = 0.005[m]$ 地点まで $R_a = F_\tau$ で初端状態から切換面までの経路が計画され、 $s = 0.058[m]$ 地点から加速線を辿り終端

状態へ到達する経路が計画されている。

Fig.5.9 の初端状態 $(R_{x0}, R_{y0}) = (3\pi/2, 10)$ からの経路は、経路が切換面に到達した $s = 0.005[m]$ 地点付近の切換面形状が複雑な形状となっている。そのため、切換面を辿るためにこのような冗長加速度入力が必要となり、また切換面の形状が複雑であるため近似切換面を導出する2点の変化による影響が大きくなりこのような波形が計画される。

Fig.5.10 の初端状態 $(R_{x0}, R_{y0}) = (3\pi/2, -10)$ からの経路探索では、経路が計画されずに易区間経路探索が終了している。これは経路に最も近い切換面の部分が複雑な形状をしているため近似切換面を構成する2点の Fig.5.3 の配列で下に格納されている点の冗長速度が、配列の上に格納されている点の冗長速度を上回ってしまい、近似切換面の傾きが負の無限大から正の無限大に変化し、式(5.9)の切換面のどちら側に経路が位置するかを示す判定式が反転した結果、このように切換面から離れる方向に冗長加速度が計画されている。

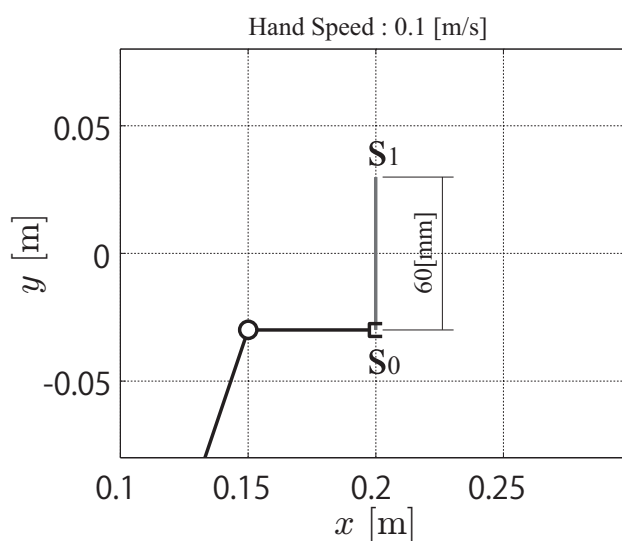
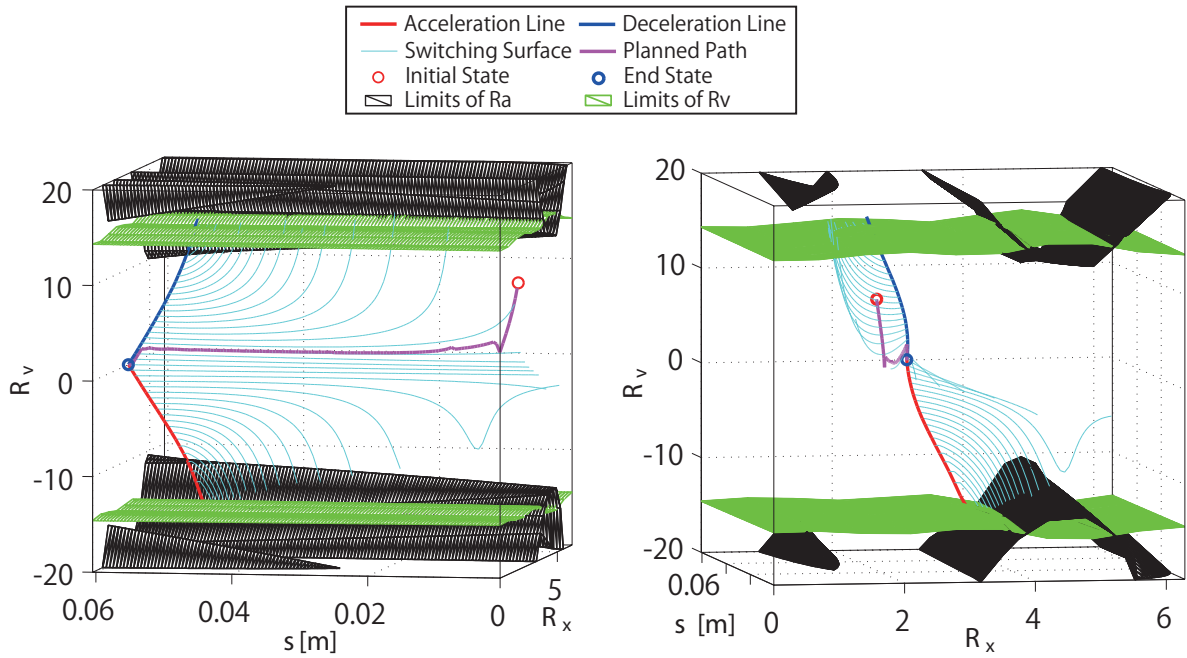
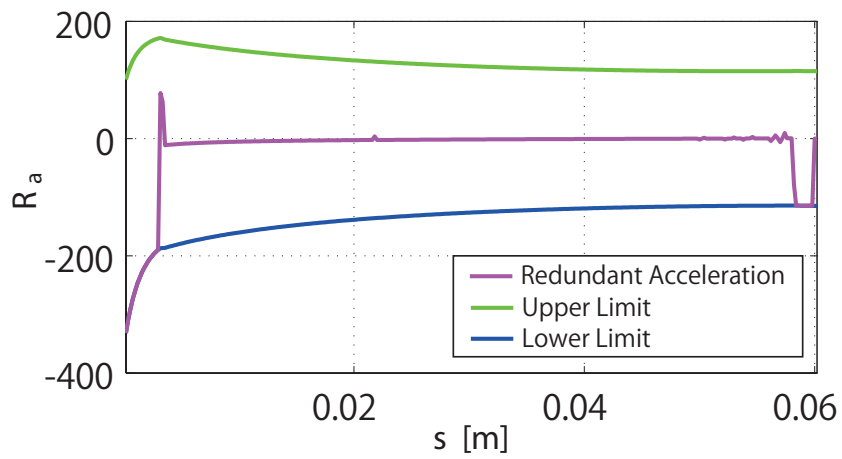


Fig. 5.6: 易区間経路探索を行なう手先軌跡指令



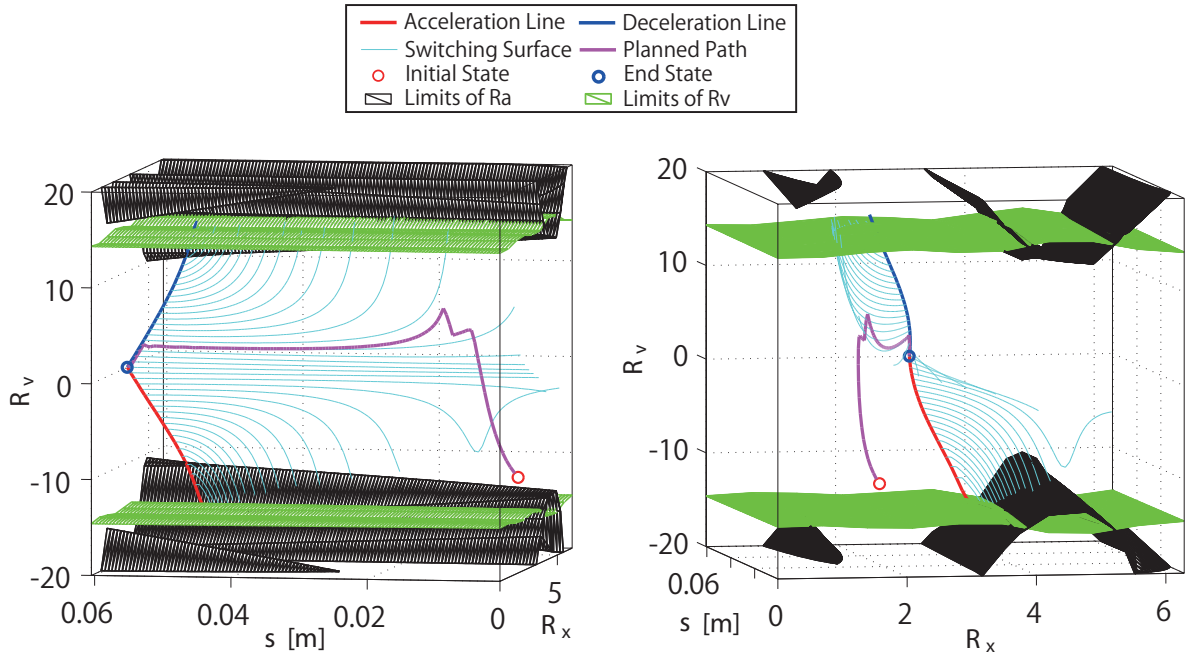
状態空間 (Side View)

状態空間 (Front View)



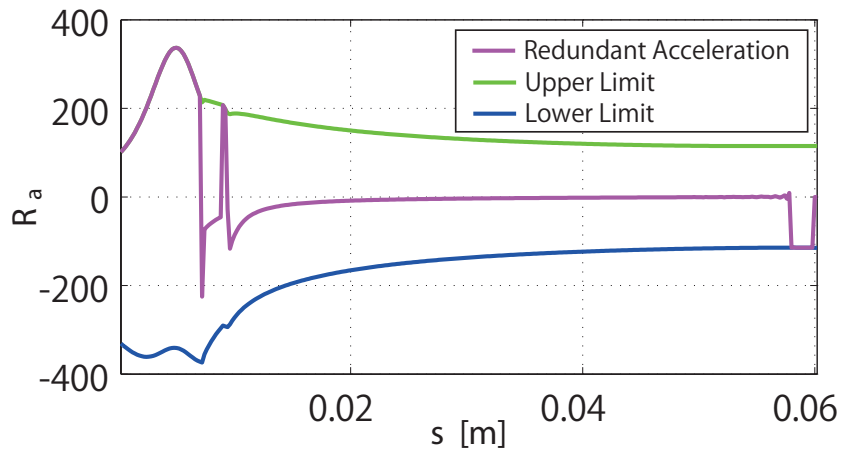
冗長加速度入力

Fig. 5.7: 初端状態 $(R_{x0}, R_{v0}) = (\pi/2, 10)$ からの易区間経路探索結果



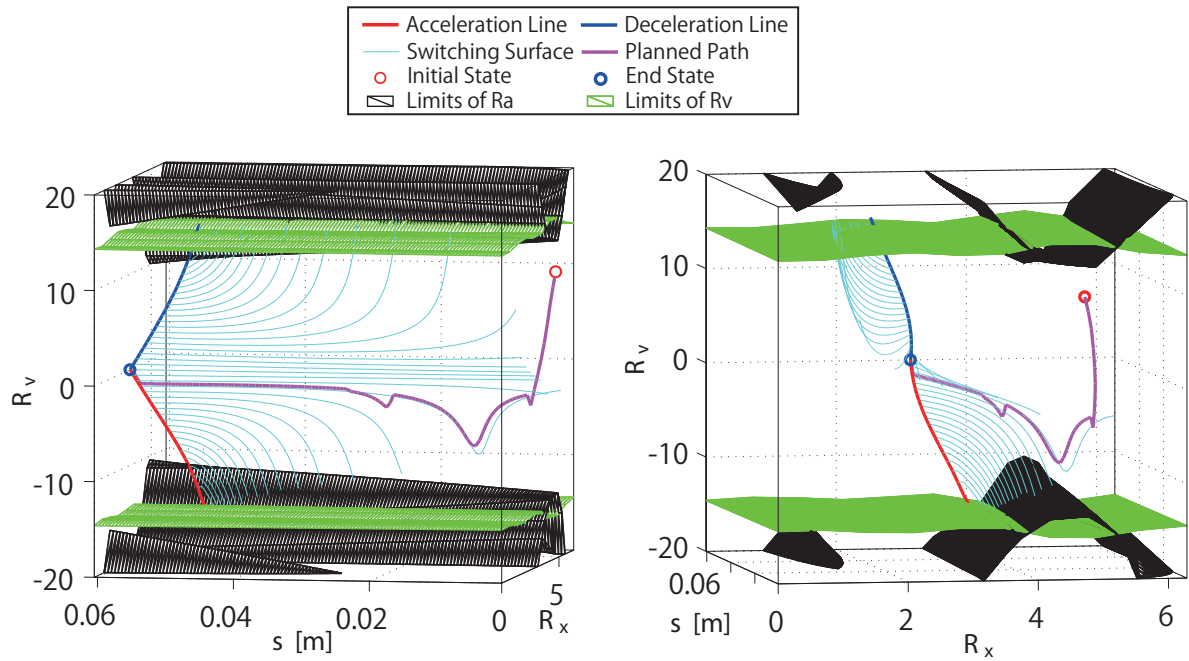
状態空間 (Side View)

状態空間 (Front View)



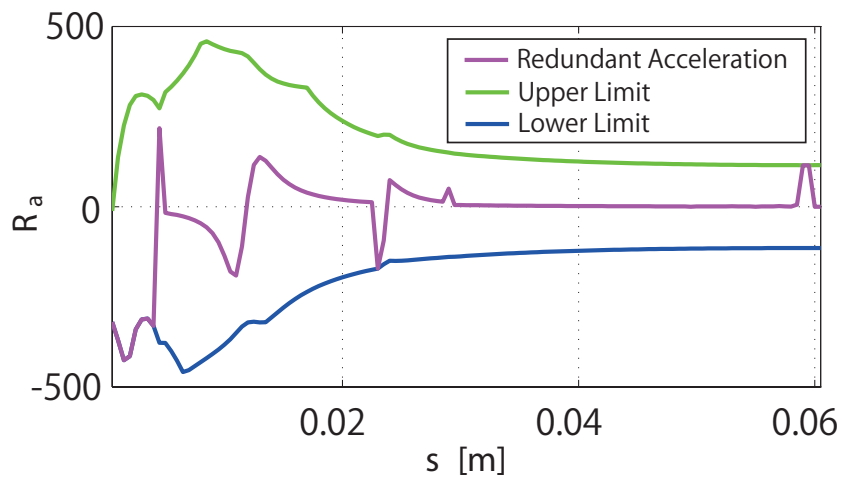
冗長加速度入力

Fig. 5.8: 初端状態 $(R_{x0}, R_{v0}) = (\pi/2, -10)$ からの易区間経路探索結果



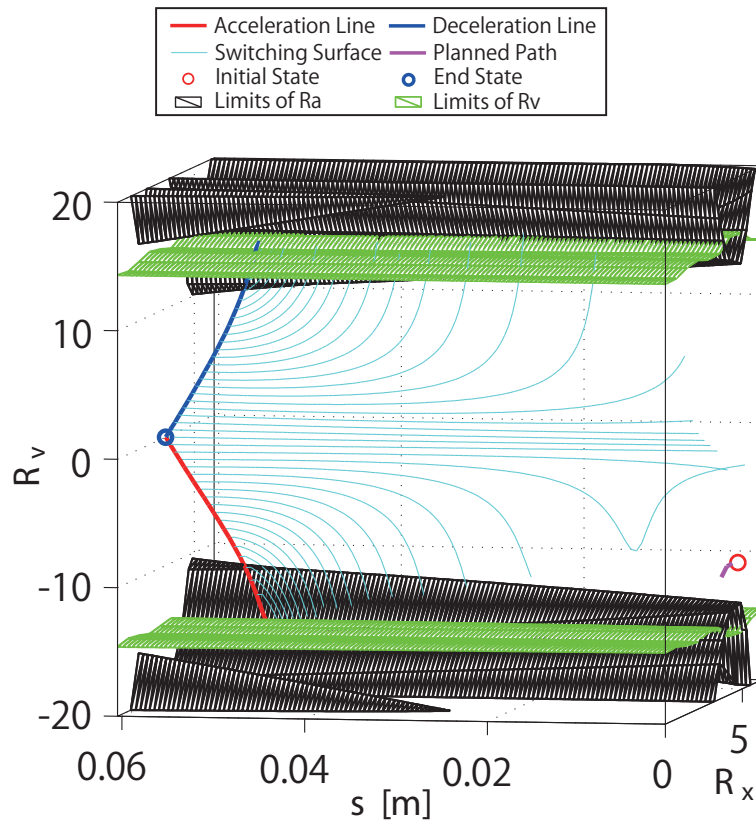
状態空間 (Side View)

状態空間 (Front View)

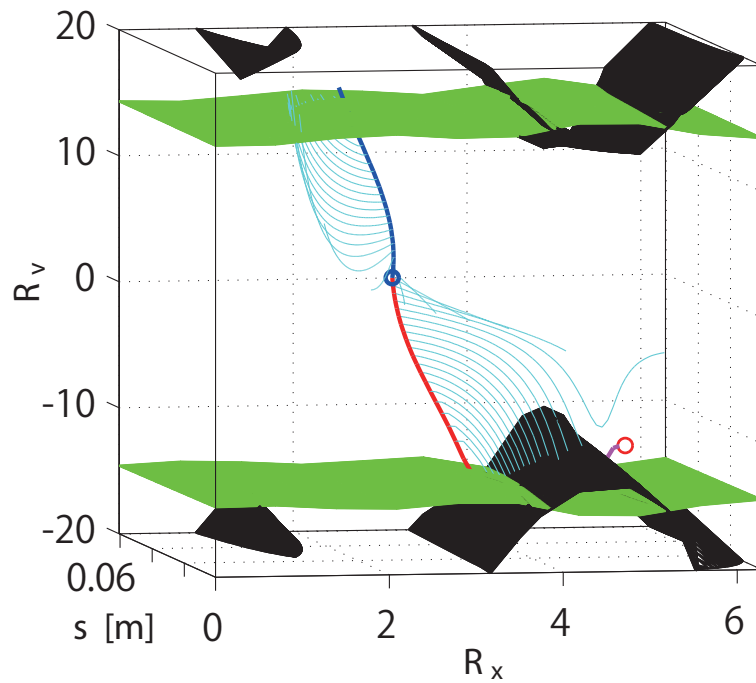


冗長加速度入力

Fig. 5.9: 初端状態 $(R_{x0}, R_{v0}) = (3\pi/2, 10)$ からの易区間経路探索結果



状態空間 (Side View)



状態空間 (Front View)

Fig. 5.10: 初端状態 $(R_{x0}, R_{v0}) = (3\pi/2, -10)$ からの易区間経路探索結果

5.5 終わりに

5章では、手先軌跡の高速動作が容易な易区間の経路探索として易区間前後の難区間で計画された経路を繋ぐ経路の探索手法について述べた。また、5章の最後には水平3関節マニピュレータを対象として、易区間経路探索の一例を示した。

易区間経路探索は易区間前後の難区間で計画された経路を繋ぐ経路を計画するために、必ず易区間終端状態到達する経路で構成される切換面を考え、易区間初端状態から切換面までの経路を探索することで易区間の初端状態と終端状態を繋ぐ経路の探索を行う。切換面は、易区間終端状態に最大冗長加速度と最小冗長加速度で到達する経路を式(5.1)を用いて求められる加速線・減速線と、これら加速線・減速線上の各点に式(5.7)の駆動トルクベクトルノルム最小となる冗長加速度で到達する経路で構成される。そのため、易区間の経路は最大もしくは最小冗長加速度、次に駆動トルクベクトルノルム最小となる冗長加速度、最後に最大もしくは最小冗長加速度により求められる経路となる。

このように易区間経路探索はこの切換面を用いて Fig.5.5 のようなアルゴリズムで易区間の初端状態と終端状態を繋ぐ経路を探索する。

第6章 区間ごとの動作計画を用いた高速 動作の計画

Contents

6.1	始めに	61
6.2	動作計画アルゴリズム	62
6.3	難区間と易区間へ手先軌跡の分割	66
6.4	最高手先速度の比較による有効性検証	69
6.5	動作計画の有効性	76
6.6	終わりに	80

6.1 始めに

4章では高手先速度動作が困難な区間に対して違反領域に侵入しない経路の探索手法について、5章では高手先速度動作が困難な区間で計画された経路を繋ぐ高手先速度動作が容易な区間の経路探索手法について述べてきた。本章ではこれら区間ごとの経路探索手法を用いた高速な軌跡追従手先定速作業動作を計画する動作計画システムについて述べる。

動作計画システムは2章で概要を述べたように、手先軌跡を高手先速度動作が困難な難区間と高手先速度動作が容易な易区間に分割し、それぞれの区間に対して状態空間上で難区間経路探索と易区間経路探索を行う。難区間経路探索は優先的に探索が行われ、易区間経路探索では難区間経路探索により得られた経路を繋ぐ経路の探索を行う。そして、探索を行う状態空間の手先速度を徐々に高速な手先速度に更新していきながら経路探索を行ない、初期手先位置から終端手先位置まで連続な経路が得られなくなるまで手先速度を高速化していく。最後に、最も高速な手先速度の状態空間上で得られた初期手先位置から終端手先位置まで連続な経路を高速な軌跡追従手先定速作業動作とするといった方法で高速な動作の計画を行う。

本章では6.2節で動作計画システムの詳しいアルゴリズムについて述べる。6.3節では経路探索手法の切換基準となる高手先速度動作が困難な区間と容易な区間への区間分割手法について述べる。最後に6.4節と6.5節では、冗長マニピュレータの一般的な動作制御手法である関節加速度ベクトルのノルム $\|\ddot{\mathbf{q}}\|$ が最小となる動作（以降、MA手法）と、マニピュレータの動作高速化を考えた場合に最も簡単に実装できる駆動トルクベクトルのノルム $\|\boldsymbol{\tau}\|$ が最小となる動作（以降、MT手法）と実験的、理論的に比較し、本研究の動作計画システムが有効であることを示す。

6.2 動作計画アルゴリズム

動作計画システムは、手先位置指令軌跡を高手先速度動作が困難な難区間と高手先速度動作が容易な易区間に区間分割し、それぞれの区間の状態空間上で難区間経路探索と易区間経路探索を行う。そして s を徐々に高速にしていき、経路探索により経路が見つからなくなるまで s を高速化することで高手先速度な軌跡追従手先定速作業の動作を計画する。この難区間経路探索は違反領域を回避する経路を求めることに重点を置いた手法で、易区間経路探索は難区間経路探索により得られた経路を繋ぐ経路を探索することで難区間の動作の予備動作を計画する手法である。易区間経路探索手法はあくまで難区間のための予備動作の計画であるため、易区間で経路が見つからない場合はその区間の探索方法を難区間経路探索に切り替え、難区間経路探索により経路が見つからなくなるまで探索を行う。

動作計画システムは以下の手順で動作を計画する。但し、 $nCell$ は分割された手先軌跡の区関数、 $j = 1, 2, \dots, nCell$ は区間番号を表し、区間 $j = 1$ は最も初端手先位置 s_0 側の区間で、区間 $j = nCell$ は最も終端手先位置 s_1 側の区間である。

Step.1 微小手先速度 δs からMA手法を用いて十分低速な手先速度を求め、経路探索を行なう状態空間の手先速度 s とし、次節で述べる手先軌跡分割手法を用いて手先軌跡を難区間と易区間に分割する。最後に、各難区間の初期手先位置に初端状態を配置する。区間 j の初端状態間隔は $w_j = (\delta R_{xj}, \delta R_{vj})$ とし、経路探索を進めるにつれて間隔を短く変更していくのでここでは広めに設定する。

Step.2 全ての難区間に対して4章の難区間経路探索を行なう。難区間経路探索は手先速度 s の状態空間上で各難区間の初端状態から探索を行い、違反領域に侵入しない経路数が閾値以上になるように難区間初端状態を追加して難区間経路探索を行なう。

このとき、違反領域に侵入しない経路数がゼロの場合はStep.8へ。

Step.3 手先軌跡の終端手先位置 s_1 に近い易区間から順($j = nCell, nCell - 1, \dots, 1$)に、全易区間に対して5章の易区間経路探索を手先速度 s の状態空間上で行う。易区間 j の経路探索は前後の難区間 $j + 1, j - 1$ の経路探索により得られた経路の端状態を易区

間の初端・終端状態として易区間経路探索を行なうが、全ての初端・終端状態の組み合わせに対して経路探索を行なうのではなく、終端手先位置 s_1 から連続した経路が存在する終端状態と全ての初端状態の組み合わせに対してのみ易区間経路探索を行う。

Step.4 各易区間（区間 j ）の経路探索終了後に終端手先位置 s_1 から連続する経路が最低1本存在するかを確認し、終端手先位置 s_1 から連続した経路が存在するならば次の易区間 $j-2$ の経路探索へ進む。

このとき $j-2 < 1$ の場合、手先速度 s を微小手先速度高速化して $s + \delta s$ とし、Step.2 へ戻る。

Step.5 終端手先位置 s_1 から連続する経路が1本も存在せず、かつ前後難区間のどちらか片方でも端状態間隔 w が最小でない場合は、区間 $j-1$ から区間 $nCell$ の間の各難区間に対して端状態間隔 $w_i = w_i/2$ ($i = j-1, j, \dots, nCell$ | 区間 i は難区間) として難区間初端状態の冗長姿勢 R_x と冗長速度 R_v を変化させた8近傍に初端状態を追加し、追加した全初端状態から難区間経路探索を行なう。

Step.6 終端手先位置 s_1 から連続する経路が1本も存在せず、かつ前後難区間両方の端状態間隔 w が最小の場合、易区間 j と前後の難区間 $j-1, j+1$ を結合して1つの難区間とし、結合前の区間 $j-1$ の初端状態を結合した区間の初端状態に、結合前の区間 $j+1$ の終端状態を結合した区間の終端状態とする。また、結合した区間以外の区間の端状態を全難区間に対する経路探索により得られた端状態、つまり Step.3 開始時の端状態に戻す。

Step.7 結合し難区間となった区間 j に対して4章の難区間経路探索を行なう。難区間経路探索は手先速度 s の状態空間上で探索を行い、違反領域に侵入しない経路数が閾値以上になるように難区間初端状態を追加して難区間経路探索を行なう。

このとき、違反領域に侵入しない経路数がゼロの場合は Step.8 へ。違反領域に侵入しない経路数が閾値以上の場合は区間 $j+1$ から Step.3 の易区間経路探索へ戻る。また、端状態の間隔 w を最小としても違反領域に侵入しない経路数が閾値以下の場合

も，区間 $j+1$ から Step.3 の易区間経路探索へ戻る．

Step.8 端状態間隔 w を最小とした難区間経路探索を用いても違反領域に侵入しない経路が得られない場合は，手先速度 $s - \delta s$ の状態空間上で得られた各区間の経路を初期手先位置 s_0 から終端手先位置 s_1 に向かってトレースし，全手先軌跡において連続な経路を求める．最後に全手先軌跡において連続な経路から式 (2.4),(3.2),(3.3) を用いてマニピュレータの動作に変換し，動作計画を終了する．

Fig.6.1 に動作計画アルゴリズムのフローチャートを示す．

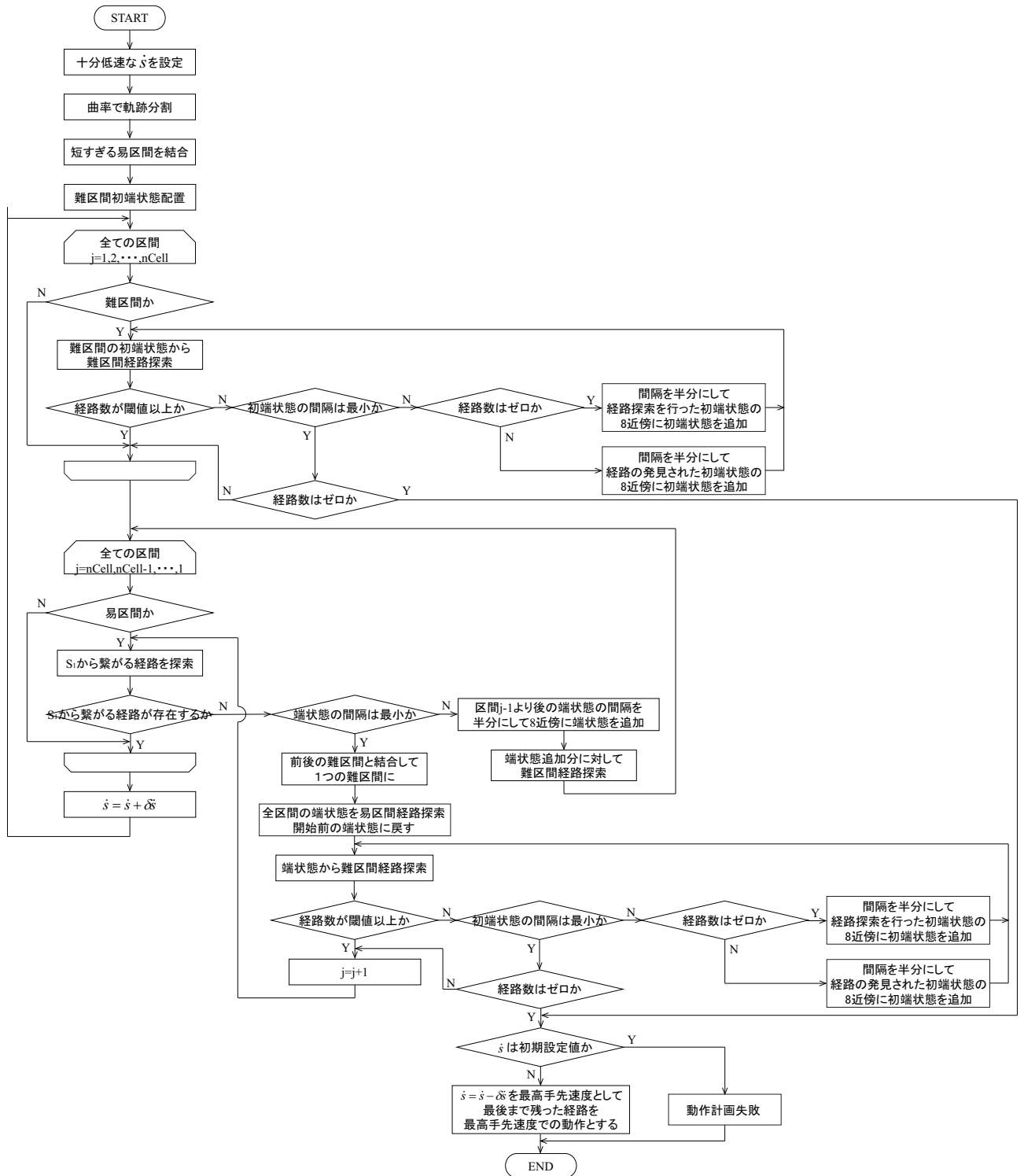


Fig. 6.1: 動作計画システムのフローチャート

6.3 難区間と易区間へ手先軌跡の分割

本研究の動作計画システムは手先軌跡を高速な手先速度での動作が困難な難区間と高速な手先速度での動作が容易な易区間に分割し、各区間で異なる経路探索手法を用いて動作計画を行う。そのため動作計画を開始する前に $\mathbf{f}, \mathbf{f}', \mathbf{f}''$ やマニピュレータのパラメータから手先軌跡を難区間と易区間に分割する必要がある。

手先軌跡の区間分割基準を議論するにあたり、まず指令軌跡に対する違反領域の傾向について述べる。まず式(3.17),(3.21)の関節ごとの制限について見てみる。制限の幅は式(6.1),(6.2)のように、制限の中央値は式(6.3),(6.4)のようになる

$$w_{ai} = G_{\tau i} - F_{\tau i} = \begin{cases} 2\tau_{ilim}/c_{1i}, & (c_{1i} > 0) \\ -2\tau_{ilim}/c_{1i}, & (c_{1i} < 0) \end{cases} \quad (6.1)$$

$$w_{vi} = G_{v_i} - F_{v_i} = \begin{cases} 2\dot{q}_{ilim}/c_{3i}, & (c_{3i} > 0) \\ -2\dot{q}_{ilim}/c_{3i}, & (c_{3i} < 0) \end{cases} \quad (6.2)$$

$$a_{ai} = \frac{G_{\tau i} + F_{\tau i}}{2} = c_{2i}/c_{1i} \quad (6.3)$$

$$a_{vi} = \frac{G_{v_i} + F_{v_i}}{2} = c_{4i}/c_{3i} \quad (6.4)$$

但し、

$$\mathbf{c}_1 = \mathbf{M}\mathbf{U}$$

$$\mathbf{c}_2 = \mathbf{M}\mathbf{J}^+ \{ \ddot{\mathbf{r}} - \mathbf{J}\mathbf{J}^+ \dot{\mathbf{r}} - \mathbf{J}\mathbf{U}\mathbf{R}_v \} + \mathbf{H} + \mathbf{g} + \mathbf{d}$$

$$\mathbf{c}_3 = \mathbf{U}$$

$$\mathbf{c}_4 = \mathbf{J}^+ \dot{\mathbf{r}}$$

$$w_{ai} = w_{ai}(\mathbf{r}(s), \mathbf{R}_x), \quad w_{vi} = w_{vi}(\mathbf{r}(s), \mathbf{R}_x), \quad a_{ai} = a_{ai}(\mathbf{r}(s), \dot{\mathbf{r}}(s, \dot{s}), \ddot{\mathbf{r}}(s, \dot{s}), \mathbf{R}_x, \mathbf{R}_v),$$

$a_{vi} = a_{vi}(\mathbf{r}(s), \dot{\mathbf{r}}(s, \dot{s}), \mathbf{R}_x)$ である。

関節ごとの制限の幅 w_{ai}, w_{vi} は状態量 s, \mathbf{R}_x の関数であるため、状態空間上の状態を $s - \mathbf{R}_x$ 平面に射影した時の位置によって制限の幅は決定し、また手先速度 \dot{s} とは独立である。そ

のため、手先速度 \dot{s} が高速になることによる動作の難しさの基準となる R_v, R_a の制限は関節ごとの制限の中央値 a_{ai}, a_{vi} によって変化する。関節ごとの制限の中央値 a_{ai}, a_{vi} は手先速度 \dot{s} と状態 s, R_x, R_v の関数であり、これらの値は動作計画により求める状態量であるため区間分割に用いることはできない。しかし、中央値 a_{ai}, a_{vi} は指令軌跡関数 f, f', f'' の関数でもある。従って式 (2.11), (2.12) より、手先速度 \dot{s} による関節ごとの制限への影響の大きさを決定する f', f'' を用いて区間分割を行うことが妥当である。特に f'' は手先速度 \dot{s} の2乗で効いてくるため、区間分割の基準とするのに適している。

一方、手先軌跡の曲率 ρ は次式で表される。

$$\rho = \left\| \frac{\partial^2 \mathbf{r}}{\partial s^2} \right\| = \sqrt{\mathbf{f}''^T \mathbf{f}''} \quad (6.5)$$

上式より曲率 ρ は f'' ベクトルの長さを表すスカラー値であるため、高速な手先速度での動作の困難さを測る良い評価関数となる。また、この評価関数を用いて高速な手先速度での動作が困難と判断される手先軌跡の曲率の高い地点は、感覚的な高速な手先速度での動作の困難さと一致する。そのため本研究では手先軌跡の難区間と易区間への分割を式 (6.5) の手先軌跡の曲率を用いて、曲率が閾値より高い区間を難区間、曲率が閾値より低い区間を易区間とする。この評価関数を用いて手先軌跡を分割する場合、マンピュレータの姿勢の変化による影響は評価関数に反映されない。そのためマンピュレータに対する指令にあたるベクトル \mathbf{r} を手先位置ベクトルとし、手先姿勢成分を含まないベクトルとしている。つまり、本研究の動作計画システムは手先姿勢が制約として与えられる軌跡追従手先定速作業の動作は計画することができない。手先姿勢が制約として与えられる軌跡追従手先定速作業の動作計画を行うためには、手先軌跡の区間分割を人間の手で分割し、与えてやる必要がある。

上述したように手先軌跡の曲率を基に区間分割を行うとチャタリングや数サンプルの長さの区間が生じるなどして、区間長が短すぎるのが原因で易区間経路探索がうまく計画できないといった問題が生じる。そのため、本研究では Fig.6.2 のように難区間に対して膨張・縮小処理を行い、短すぎる易区間を難区間に含める処理を行う。

Step.1 手先軌跡の曲率に基づいて閾値を用いて手先軌跡を易区間と難区間に分割する。

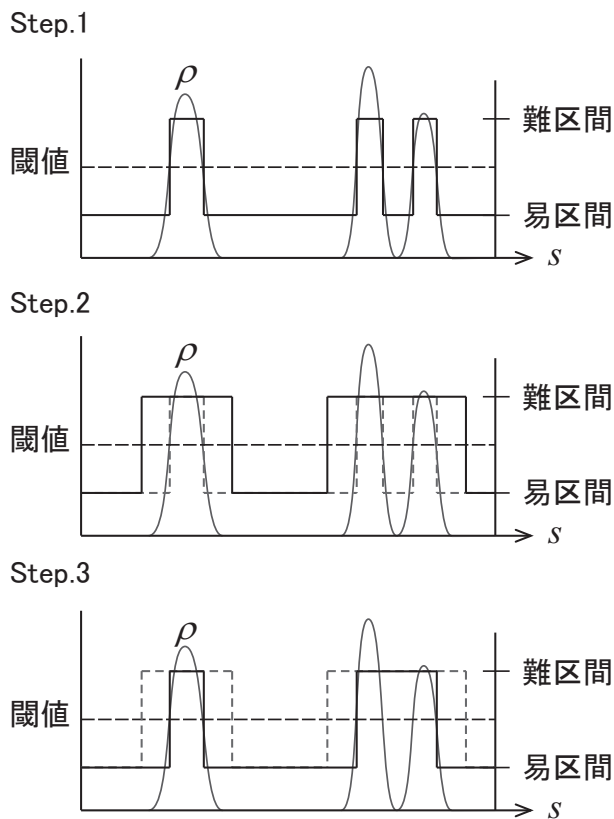


Fig. 6.2: 難区間に対する膨張・縮小処理

Step.2 各難区間に対して区間膨張処理を行い, 各難区間の前後 s_w [m] の易区間を難区間とする. このとき, 膨張した難区間同士が重なった場合は1つの難区間とする. 但し, $s_w \in \mathbb{R} \leq 0$ である.

Step.3 区間膨張処理により得られた難区間に対して s_w [m] の区間縮小処理を行う.

上記の区間膨張・縮小処理を行うことで, チャタリングによる細かい区間の切り替わりや, 易区間経路探索が上手くいかないような数サンプルの長さしかない易区間が生成されることを防ぐことができる.

6.4 最高手先速度の比較による有効性検証

本節では本動作計画システムと一般的な冗長マニピュレータの動作制御手法である MA 手法，動作高速化を考えた場合に簡単に実装可能な MT 手法を用いた場合の軌跡追従手先定速作業動作の最高手先速度の比較を行ない，本動作計画システムの有効性を示す．また，動作計画に必要とする計算量を比較するために本動作計画システムと MA 手法，MT 手法，さらに本動作計画システムにおいて区間を分割して計画することによる計算量の違いを比較するために全手先軌跡を難区間として計画を行った時の演算時間を示す．

MA 手法と MT 手法は初期状態によって動作が決定する手法であるため，これら 2 つの動作制御手法は初期状態に関して全探索を行ない最も高速な手先速度動作を求め本動作計画システムとの最高手先速度の比較を行う．マニピュレータは Fig.4.1 のモデルを有する水平 3 関節冗長マニピュレータを対象として動作計画を行う．また，冗長マニピュレータのパラメータは Tab.4.1 に示すパラメータとした．軌跡追従手先定速作業の手先軌跡は Fig.6.3 に示す半径 3[mm] の円弧軌跡と長軸 10[mm]，短軸 6[mm] の楕円軌跡を直線軌跡で繋いだ手先軌跡とした．区間分割を行うための曲率閾値を 200 とし，半径 5[mm] の円弧軌跡を閾値として分割されるようにした．この閾値を用いた場合の区間分割結果は Fig.6.3 のようになり，この区間から動作計画を開始する．動作計画には CPU: Intel Core i7 3.4GHz，メモリ: 8GB の PC で MATLAB を用いて演算を行った．

各動作計画により得られた最高手先速度と動作計画に要した演算時間を Tab.6.1 に示す．本動作計画システムにより計画された動作を Fig.6.5 に，この時の手先軌跡の易区間と難区間を Fig.6.4 に，冗長パラメータを Fig.6.6 に，状態空間上の違反領域と計画された経路を Fig.6.7 に示す．また，動作計画システムにより計画された動作の駆動トルクを Fig.6.8 に，関節速度を 6.9 に示す．

Tab.6.1 より，本動作計画システムを用いてマニピュレータの冗長性を利用した動作を計画することで高速な軌跡追従手先定速作業動作が計画されていることがわかる．しかし，MA 手法や MT 手法と比較して動作計画を行うことにより演算時間は大幅に増加している．また，動作計画システムにおいて区間分割を行う場合と全手先軌跡を難区間とした場合の最大手先速度を比較すると，全手先軌跡を難区間とした場合の方が最大手先速度が

Tab. 6.1: 動作計画により得られた最大手先速度

	Maximum Hand Speed	Calculation Time
Proposal Method	0.28 [m/s]	1 h 24 m
Motion by Path Finding for Difficult Sections	0.29 [m/s]	3 h 55 m
Motion by Minimum Joint Torque	0.26 [m/s]	34 s
Motion by Minimum Joint Acceleration	0.14 [m/s]	32 s

高速な動作が計画されている。これは区間を分割して動作計画する場合は、各難区間の経路探索を離散的に難区間の初端状態を与えているため、手先速度高速化に伴い減少する違反領域に侵入しない経路の存在する難区間初端状態の範囲が経路探索を行う離散化された難区間初端状態の隙間に存在したためであると考えられる。そのため計算時間は増加するが、各難区間初端状態の間隔を密にして動作計画を行うことで全手先軌跡を難区間とした場合の最大手先速度と同等の手先速度での動作が得られる。Fig.6.3, 6.4の区間では動作計画前の区間では易区間とされていた初期手先位置から3番目の区間が難区間に変更されている。これはFig.6.3からも分かるように、易区間から難区間に変更された区間は区間長が短く易区間経路探索を行うには短すぎるため難区間に変更されている。

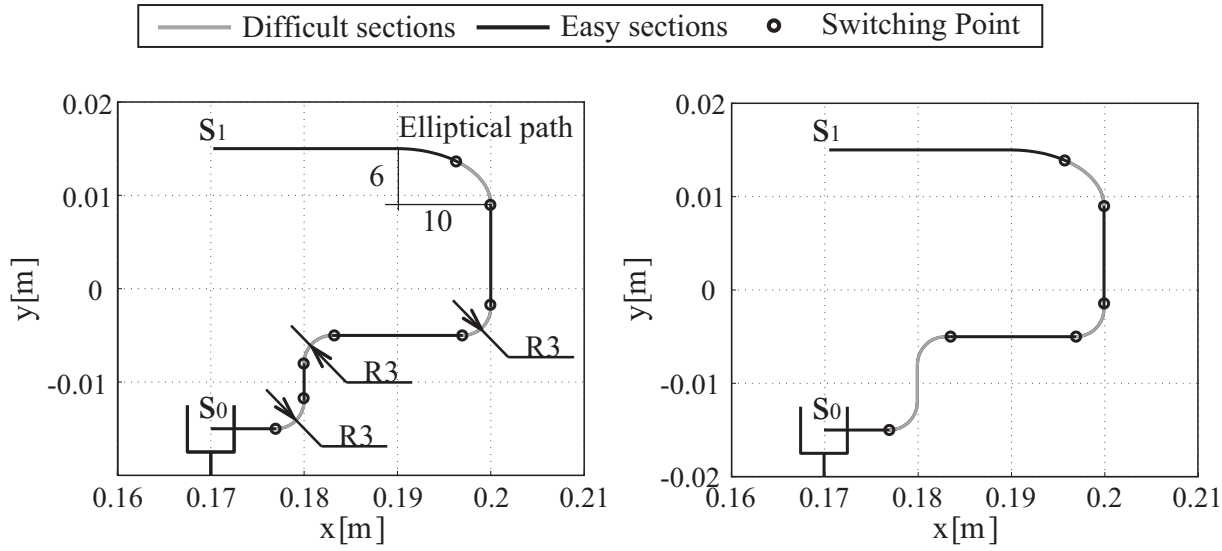


Fig. 6.3: 動作計画前の易・難区間

Fig. 6.4: 動作計画後の易・難区間

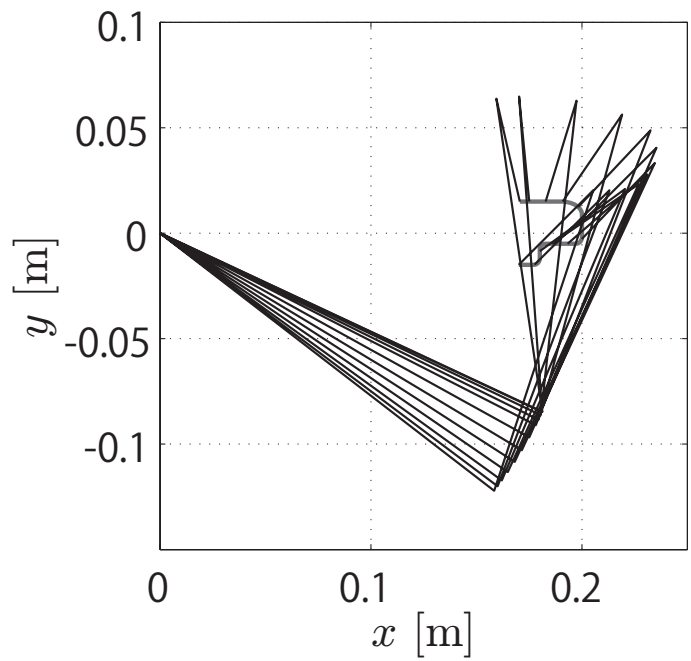


Fig. 6.5: 動作計画により得られた動作

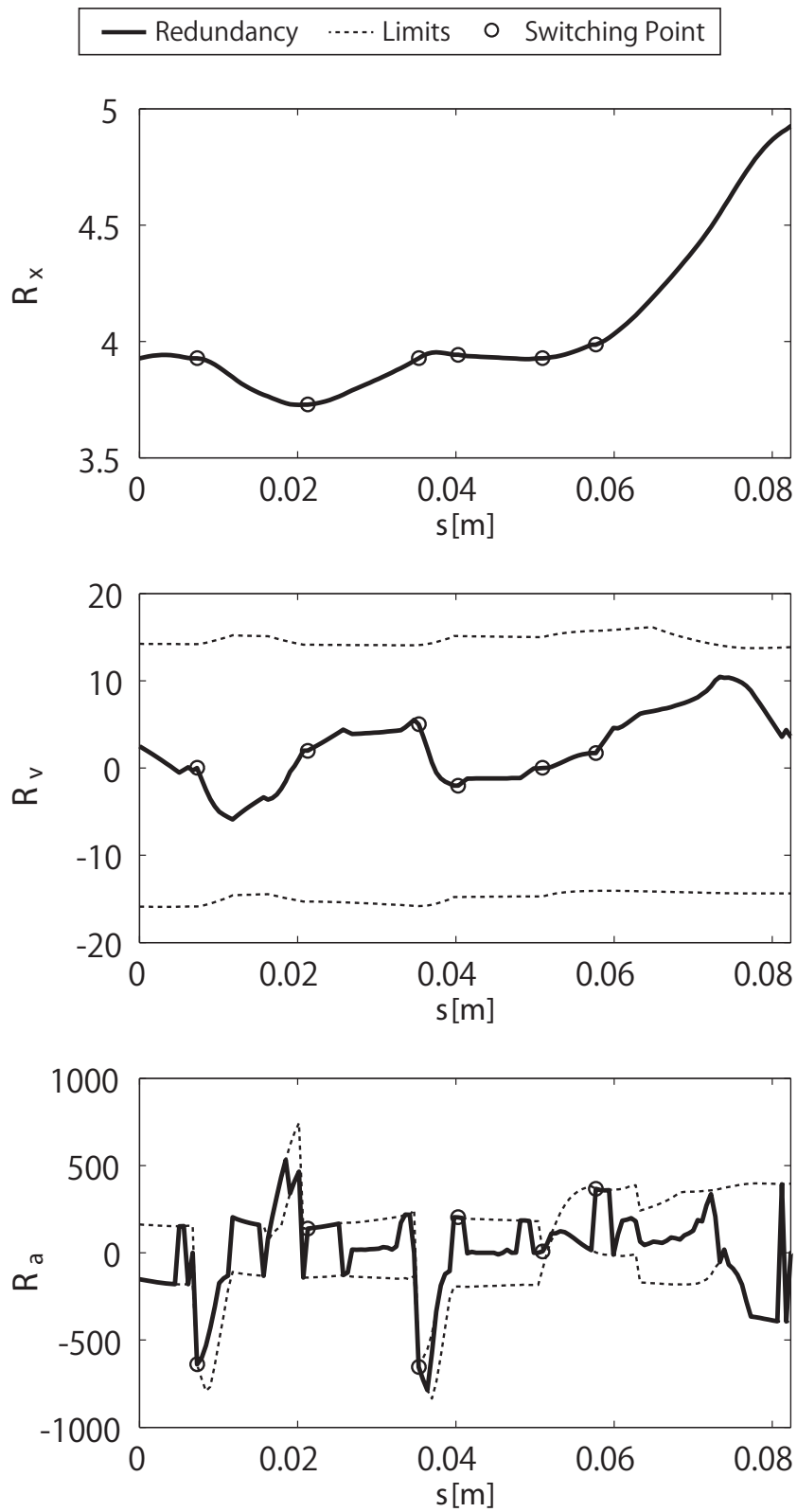
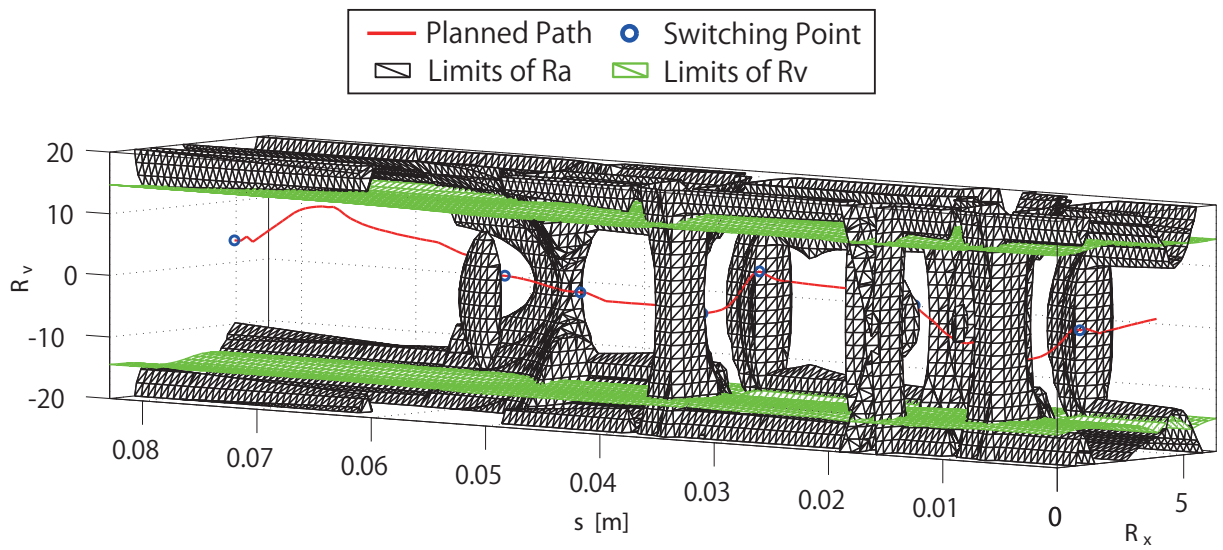
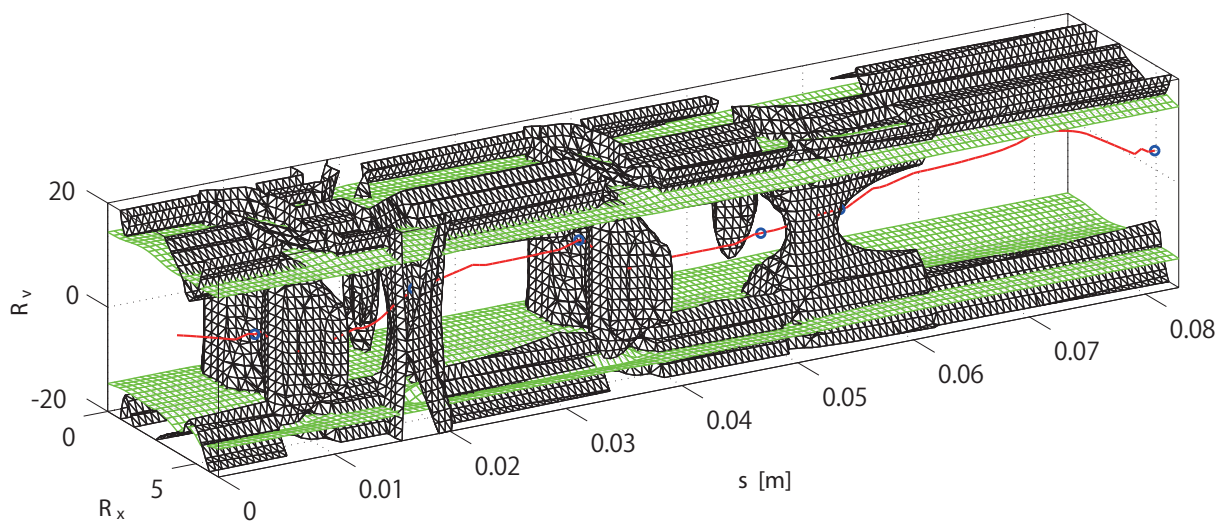


Fig. 6.6: 動作計画により得られた冗長パラメータ



状態空間-A



状態空間-B

Fig. 6.7: 状態空間上の経路と違反領域

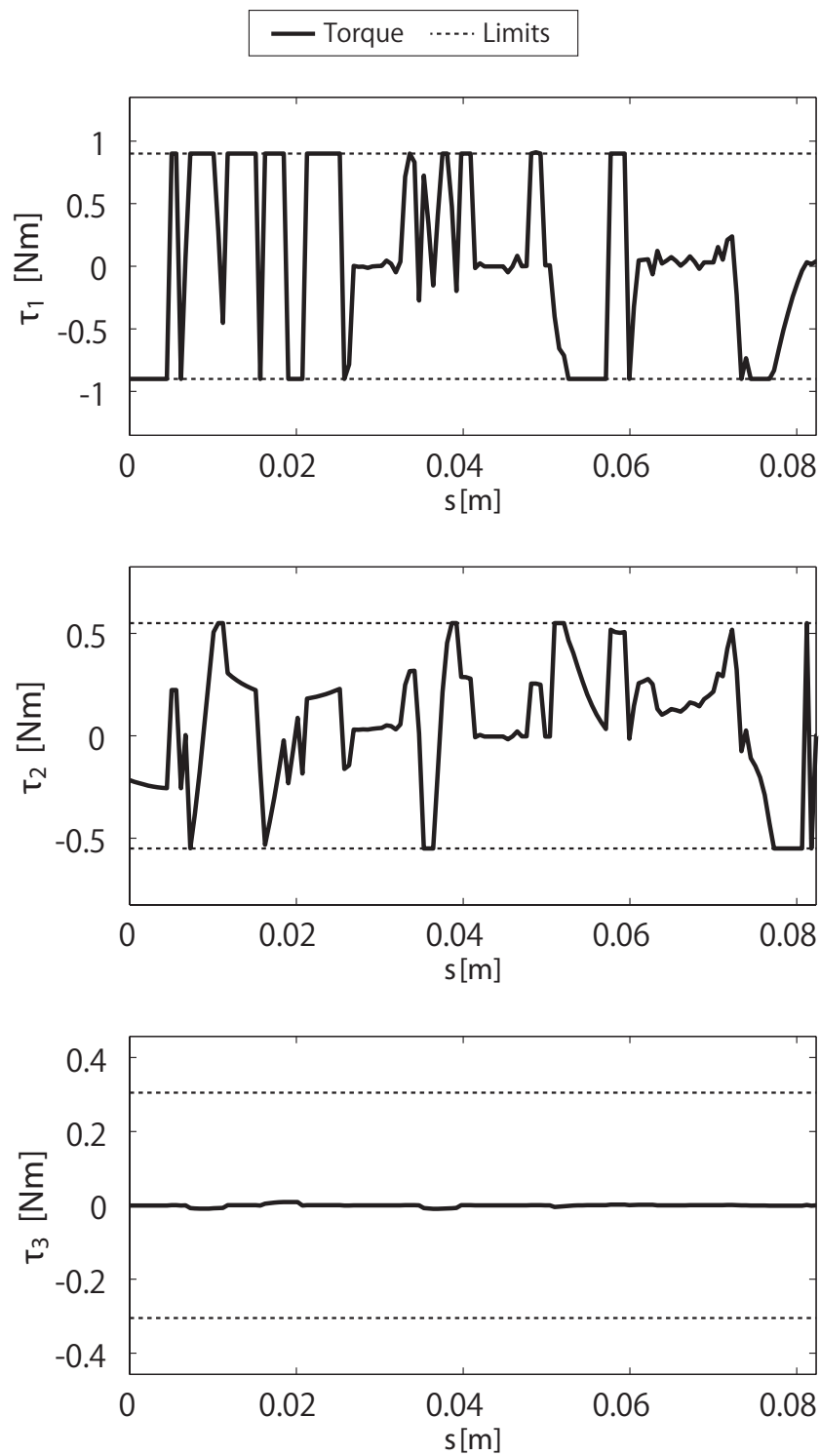


Fig. 6.8: 計画された動作の駆動トルク

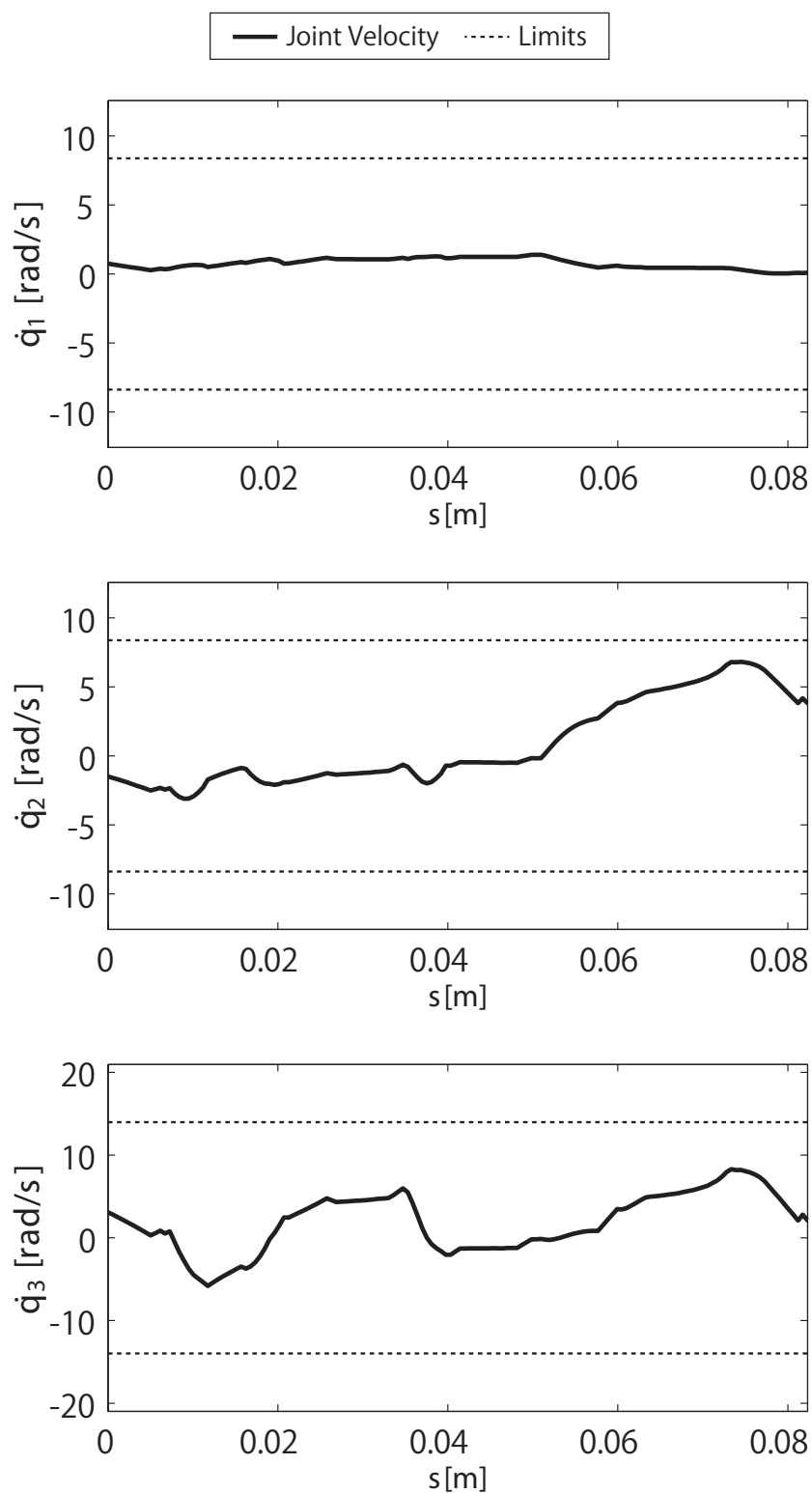


Fig. 6.9: 計画された動作の関節速度

6.5 動作計画の有効性

本節では一般的な冗長マニピュレータの動作制御手法である関節角加速度ベクトルのノルム $\|\ddot{\mathbf{q}}\|$ が最小となる動作（以降，MA手法）や，マニピュレータの動作高速化を考えた場合に最も簡単に実装できる駆動トルクベクトルのノルム $\|\boldsymbol{\tau}\|$ が最小となる動作（以降，MT手法）と比較し，本動作計画を用いた場合は必ずこれらの手法を用いて動作可能な最大手先速度以上の動作が計画されること理論的に示す。

まずMA手法と比較について述べる．MA手法は次式に再記する式(2.6)を式(6.7)のように右辺第2項を零として手先加速度 $\ddot{\mathbf{r}}$ と初期状態により決定される動作である。

$$\ddot{\mathbf{q}} = \mathbf{J}^+(\ddot{\mathbf{r}} - \dot{\mathbf{J}}\dot{\mathbf{q}}) + (\mathbf{I} - \mathbf{J}^+\mathbf{J})\mathbf{k}_2 \quad (6.6)$$

$$\ddot{\mathbf{q}} = \mathbf{J}^+(\ddot{\mathbf{r}} - \dot{\mathbf{J}}\dot{\mathbf{q}}) \quad (6.7)$$

式(6.7)で表されるMA手法による動作は冗長加速度 $R_a = 0$ による動作と等しく，Fig.3.2からもMA手法を用いた動作は関節加速度ベクトルノルム $\|\ddot{\mathbf{q}}\|$ が最小となることがわかる．また，Fig.6.10に前節で計画したMA手法を用いて計画した最大手先速度動作時の冗長加速度を示す．Fig.6.10からもMA手法を用いた動作は冗長加速度 R_a は常に零として計画される動作であることがわかる。

そのため，冗長加速度制限が $G_\tau \geq 0$ かつ $F_\tau \leq 0$ の場合は $R_a = 0$ の動作は難区間経路探索の探索空間内に含まれ， $G_\tau < 0$ もしくは $F_\tau > 0$ の場合は $R_a = 0$ の動作は違反領域内であるため駆動トルクが駆動トルク制限を超過する動作となる．従って，本研究の動作計画システムを用いて高速な手先速度動作を計画した場合は必ずMA手法を用いた最大手先速度動作以上の手先速度動作が得られる。

次にMT手法との比較について述べる．MT手法は式(5.7)の冗長加速度を用いた動作で，MA手法と同様にその動作は手先加速度 $\ddot{\mathbf{r}}$ と初期状態により決定する．Fig.6.11に前節で計画したMT手法を用いた最大手先速度動作時の冗長加速度を示す。

Fig.6.11のMT手法を用いた動作計画結果より，駆動トルクノルム最小となる冗長加速度 R_{at} は冗長加速度制限の中央値に近い値をとることが予想される．第 i 関節の駆動トルク制限による冗長加速度制限の中央値は次式となる。

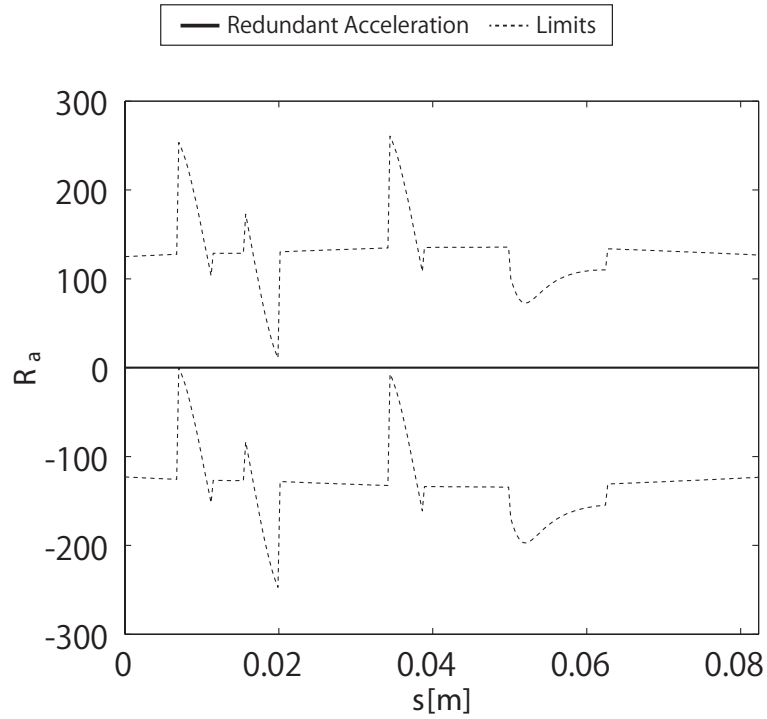


Fig. 6.10: MA 手法を用いて計画した最大手先速度動作時の冗長加速度

$$\frac{G_{\tau i} + F_{\tau i}}{2} = -c_{2i}/c_{1i} \quad (6.8)$$

ここで式 (6.8) と駆動トルクノルム最小となる冗長加速度 $R_{a\tau}$ が近しい値となることを示す. 次式に駆動トルクノルムが最小となる冗長加速度 $R_{a\tau}$ を再記する.

$$R_{a\tau} = -(\mathbf{c}_1^T \mathbf{W}^2 \mathbf{c}_1)^{-1} \mathbf{c}_1^T \mathbf{W}^2 \mathbf{c}_2 \quad (6.9)$$

但し,

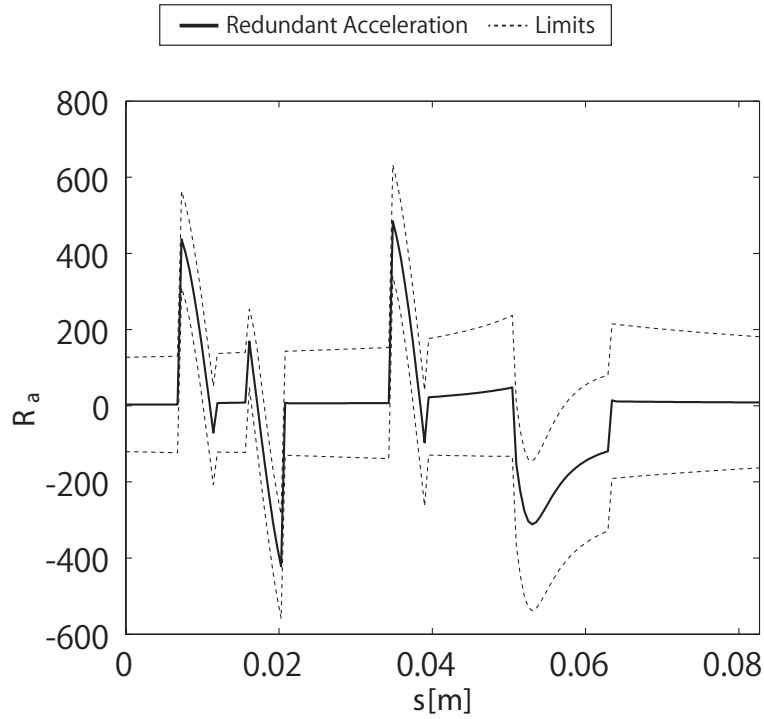


Fig. 6.11: MA 手法を用いて計画した最大手先速度動作時の冗長加速度

$$\mathbf{W} = \begin{bmatrix} 1/\tau_{lim1} & & & 0 \\ & 1/\tau_{lim2} & & \\ & & \ddots & \\ 0 & & & 1/\tau_{limn} \end{bmatrix}$$

$$\mathbf{c}_1 = \mathbf{M}\mathbf{U}$$

$$\mathbf{c}_2 = \mathbf{M}\mathbf{J}^+ (\mathbf{f}'' \mathbf{s}^2 - \mathbf{J}\mathbf{J}^+ \mathbf{f}' \dot{\mathbf{s}} - \mathbf{J}\mathbf{U}\mathbf{R}_v) + \mathbf{H} + \mathbf{g} + \mathbf{d}$$

式 (6.9) の行列式を展開する.

$$\sum_{i=1}^n \frac{c_{1i}^2}{\tau_{limi}^2} R_{a\tau} = - \sum_{i=1}^n \frac{c_{1i}}{\tau_{limi}^2} c_{2i} \quad (6.10)$$

上式に式 (6.8) を代入すると以下の式が得られる.

$$R_{a\tau} = \frac{\sum_{i=1}^n \frac{c_{1i}^2}{\tau_{limi}^2} (G_{\tau i} + F_{\tau i})}{2 \sum_{i=1}^n \frac{c_{1i}^2}{\tau_{limi}^2}} \quad (6.11)$$

式(6.11)より, 各関節の駆動トルク制限による冗長加速制限の中央値が等しくなると仮定すると

$$R_{a\tau} = \frac{G_{\tau} + F_{\tau}}{2} \quad | \quad G_{\tau_1} + F_{\tau_1} = G_{\tau_2} + F_{\tau_2} = \dots = G_{\tau_n} + F_{\tau_n} \quad (6.12)$$

従って各関節の駆動トルク制限による冗長加速度制限の中央値が等しい場合, 駆動トルクノルム最小となる冗長加速度 $R_{a\tau}$ は冗長加速度制限の中央値を取ることがわかる.

このことから MT 手法の冗長加速度は冗長加速度制限の中央値かそれに近い値を取ると言えるため, MT 手法による動作は難区間経路探索の探索空間内に含まれる. 従って, 本研究の動作計画を用いて高速な手先速度動作を計画した場合は必ず MT 手法を用いた場合の最大手先速度動作以上の手先速度動作が計画される.

6.6 終わりに

6章では、4章と5章で述べた難区間経路探索手法と易区間経路探索手法を用いて高速な軌跡追従手先定速作業動作を計画する動作計画システムについて述べた。また、動作計画に必要な手先速度動作が困難な区間と容易な区間への分割手法についても述べた。最後に本動作計画システムの有効性を、シミュレーションによる動作計画結果の比較と動作計画システムの探索空間に関節加速度ベクトル最小動作と駆動トルクベクトル最小動作が含まれることを示すことで動作計画システムの有効性を示した。

易区間経路探索と難区間経路探索の適用範囲を決定する区間分割は手先軌跡の曲率を用いて分割する。この分割した区間を用いて、動作計画システムでは十分低速な手先速度から経路探索を開始し、徐々に手先速度を高速な手先速度に増加させていき全手先軌跡において連続な経路が得られなくなるまで手先速度を増加させる。このとき、最後まで残っていた全手先軌跡において連続な経路を最も高速な手先速度における軌跡追従手先定速作業動作とする。また、易区間の経路探索により全手先軌跡において連続な経路が得られなくなった場合はその易区間を前後の難区間と結合して1つの難区間とし全手先軌跡において連続な経路の探索を続ける。

このような動作計画システムを用いて高速な軌跡追従手先定速作業動作の計画を行う。

第7章 動作計画適用例

Contents

7.1	始めに	82
7.2	直動関節を手首に有する水平マニピュレータへの適用	83
7.3	リニアスライダ上に設置された垂直多関節マニピュレータへの適用	90
7.4	手先軌道回転半径の最小化	98
7.5	終わりに	103

7.1 始めに

本動作計画はマニピュレータの手先位置の関数で与えられる手先軌道指令を追従する冗長自由度1を有するマニピュレータに対して一般化されている。そのため本章では様々な構成のマニピュレータに対して動作計画システムを適用し、本動作計画は様々なマニピュレータに対して適用可能であることを示す。

本章では7.2節で手首部に直道関節を有する水平マニピュレータに対する動作計画適用結果を示し、7.3節ではリニアスライダ上に設置された垂直多関節マニピュレータに対する動作計画適用結果を示す。7.4節では本動作計画システムを応用し、軌跡追従手先定速作業において丸角四角軌跡を追従する場合の角部円弧軌跡の円弧半径をマニピュレータの冗長性を利用することで最小化する手法について述べる。

7.2 直動関節を手首に有する水平マニピュレータへの適用

本節では Fig.7.1 のようなモデルで表される水平 2 関節マニピュレータの手首部に直道アクチュエータが取り付けられた構造のマニピュレータに対して動作計画を行う。マニピュレータのパラメータは Tab.7.3 に示す値を用いた。冗長姿勢 R_x は $\Sigma = [0, 0, 1]^T$ とし、手首部の直動関節変位を冗長姿勢 R_x とした。動作計画は微小手先速度 $\delta s = 0.01[\text{m/s}]$ とし、探索範囲は $R_x | 0 \leq R_x \leq 0.15, R_v | -15 \leq R_v \leq 15$ を 17 段階分割とした。軌跡追従手先定速作業の手先軌跡は Fig.7.2 に示す半径 3[mm] の円弧軌跡と長軸 10[mm] と短軸 6[mm] の楕円軌跡を直線軌跡で繋いだ軌跡とした。

本動作計画システムを用いて計画された動作の最大手先速度と動作計画システムの効果を示すために MA 手法を用いた場合の最大手先速度と MT 手法を用いた場合の最大手先速度を Tab.7.2 に示す。計画した高速な軌跡追従手先定速作業動作を Fig.7.4 に、動作計画終了時の手先軌跡の易区間と難区間への分類を Fig.7.3 に、計画された冗長パラメータを Fig.7.5 に、状態空間上の経路と違反領域を Fig.7.6 に、計画された動作の駆動トルクを Fig.7.7 に、関節速度を Fig.7.8 に示す。

本動作計画システムを用いて冗長性を利用した動作を計画することで Tab.7.2 に示すように、軌跡追従手先定速作業の高速化がなされている。また、これらの結果より本動作計画システムは直動関節を含む冗長マニピュレータに対しても適用可能であることが分かる。

Tab. 7.2: 動作計画により得られた最大手先速度

	Maximum Hand Speed
Proposal Method	0.26 [m/s]
Motion by Minimum Joint Torque	0.25 [m/s]
Motion by Minimum Joint Acceleration	0.23 [m/s]

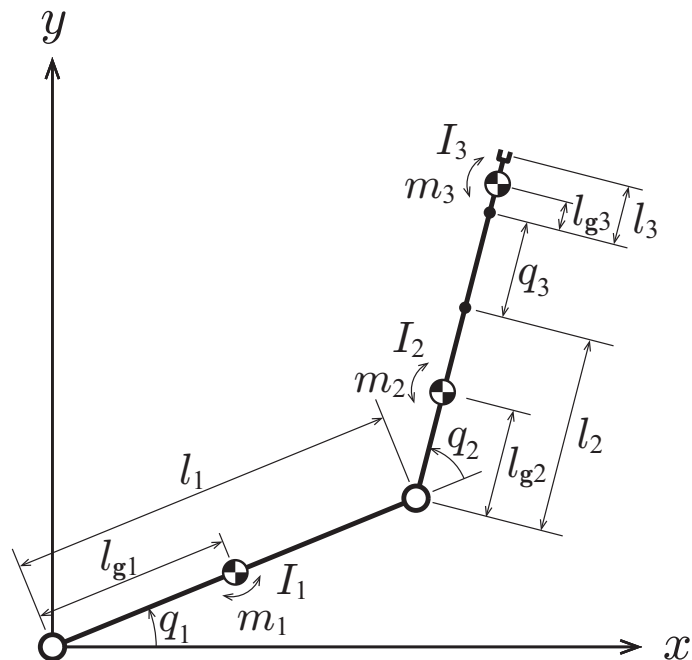


Fig. 7.1: 直動関節を手首に有する水平マニピュレータのモデル

Tab. 7.1: 直動関節を有する水平マニピュレータの諸パラメータ

	Link1	Link2	Link3
Length of links	200 [mm]	100 [mm]	30 [mm]
Mass of links	1235 [g]	380 [g]	110 [g]
Length to center of gravity	-27.88 [mm]	0.24 [mm]	2.81 [mm]
Inertia of links	7935 [mgm ²]	882 [mgm ²]	17 [mgm ²]
Joint torque limits	0.90 [Nm]	0.55 [Nm]	2 [N]
Joint velocity limits	8.4 [rad/s]	8.4 [rad/s]	1.3 [m/s]
vector Σ	[0 0 1]		

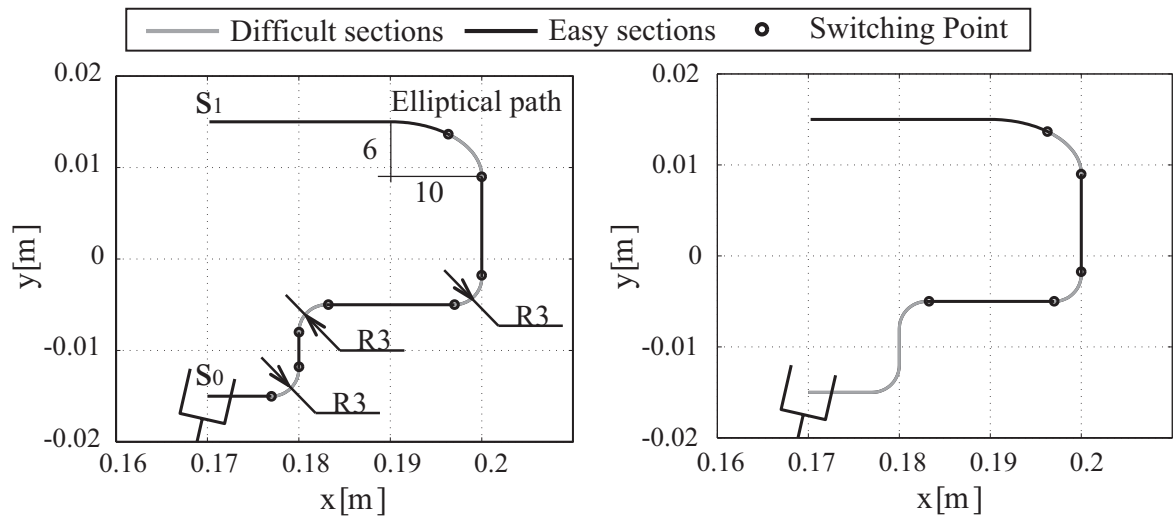


Fig. 7.2: 動作計画前の易区間と難区間

Fig. 7.3: 動作計画後の易区間と難区間

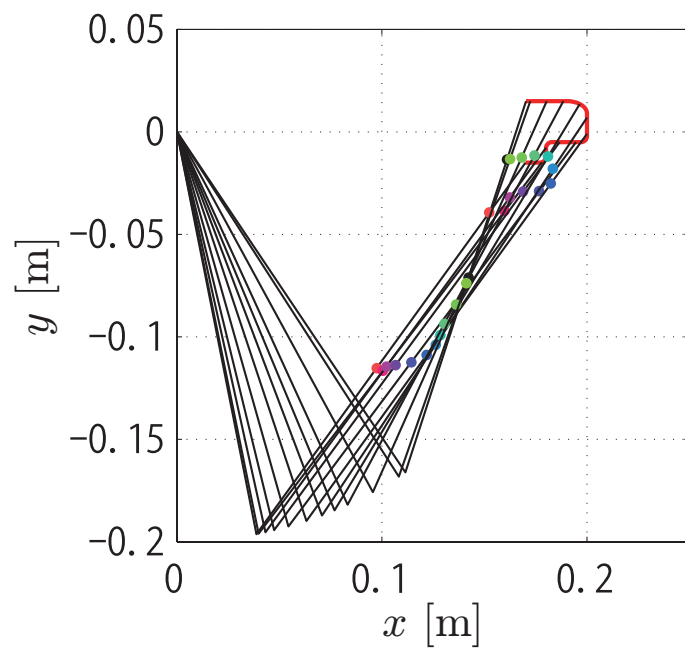


Fig. 7.4: 動作計画により得られた動作

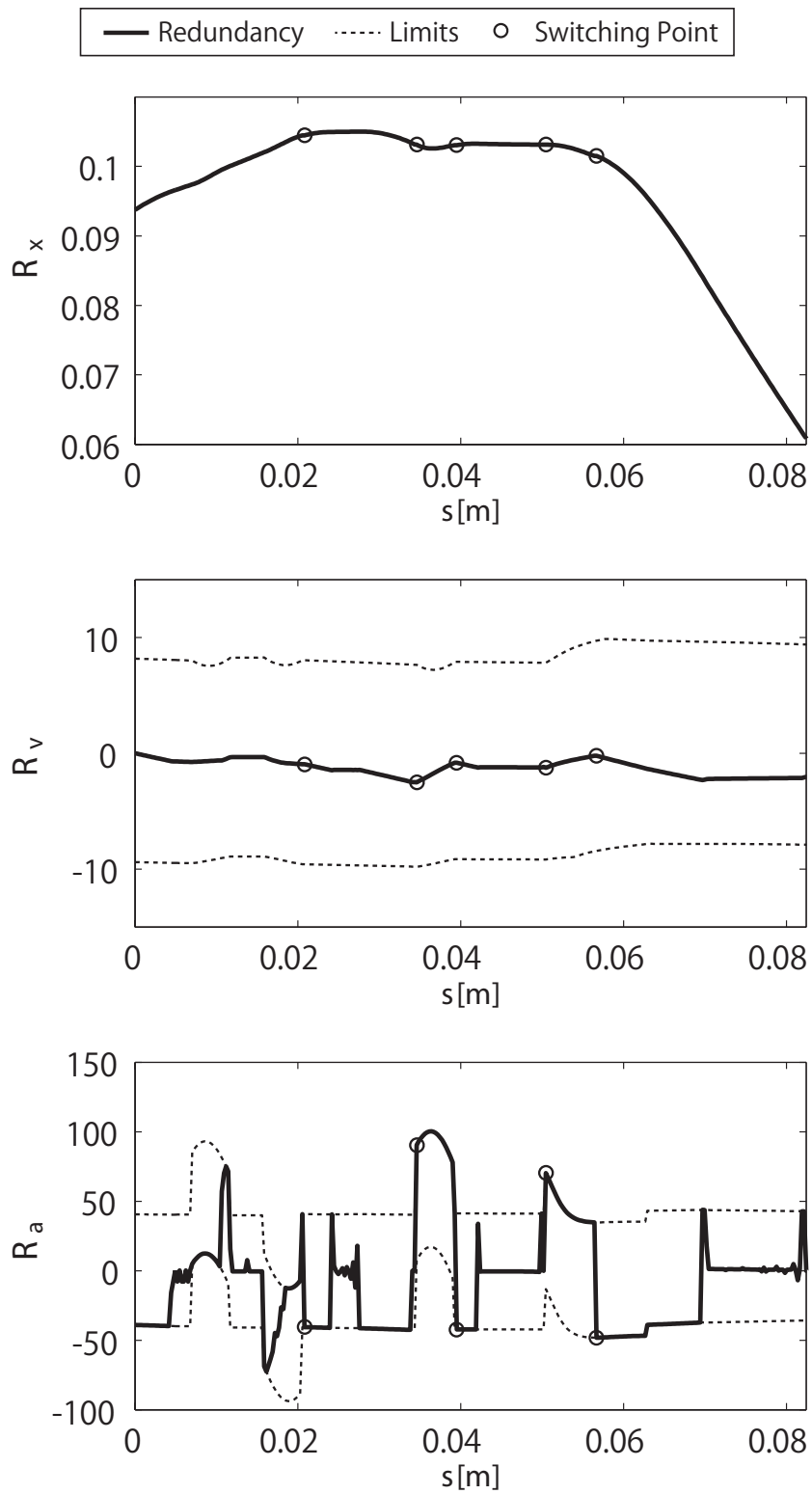
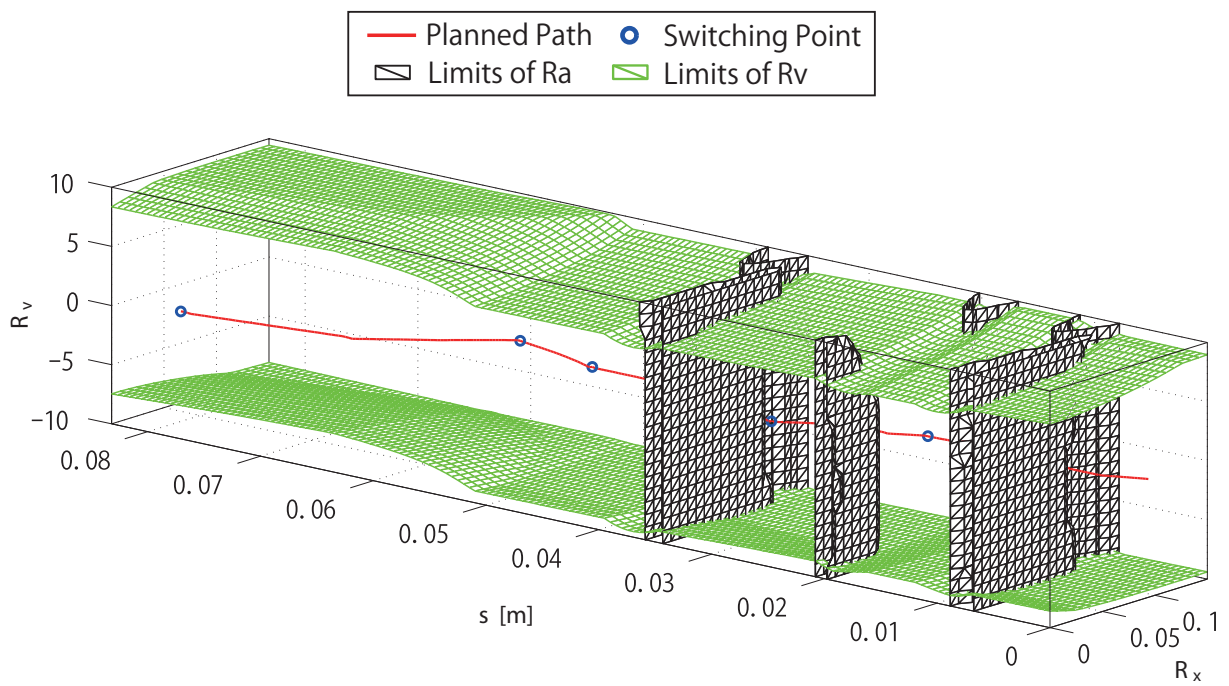
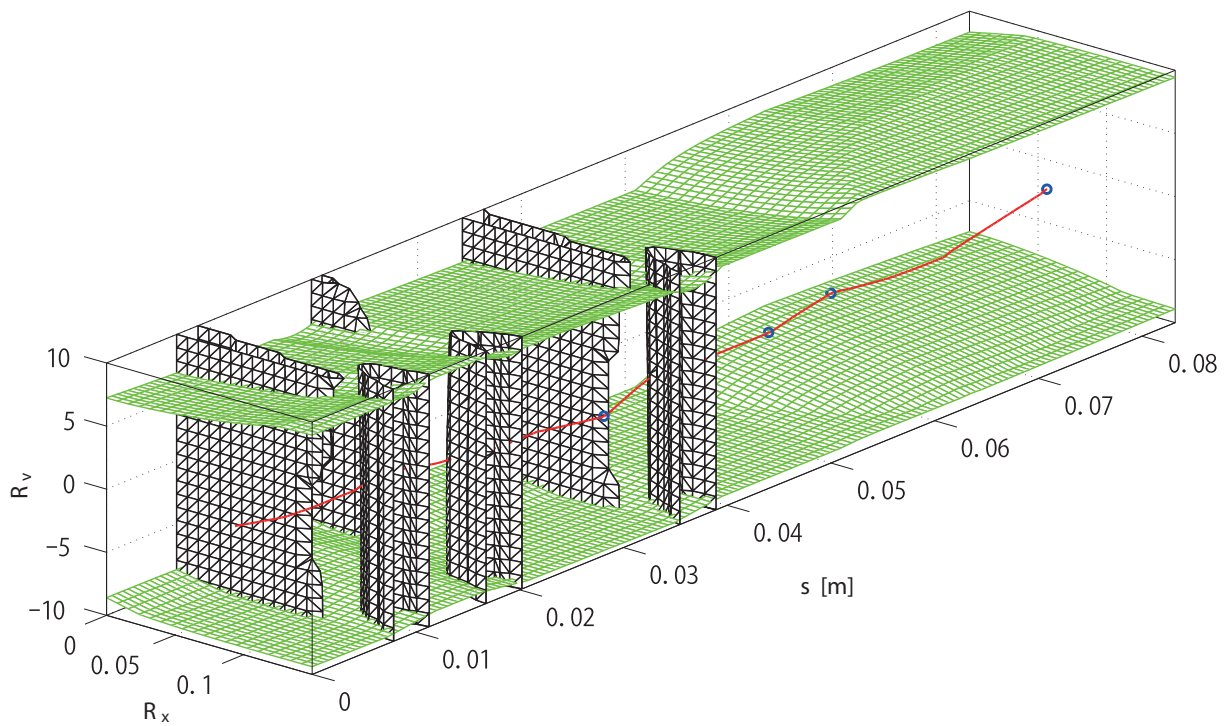


Fig. 7.5: 動作計画により得られた冗長パラメータ



状態空間-A



状態空間-B

Fig. 7.6: 状態空間上の経路と違反領域

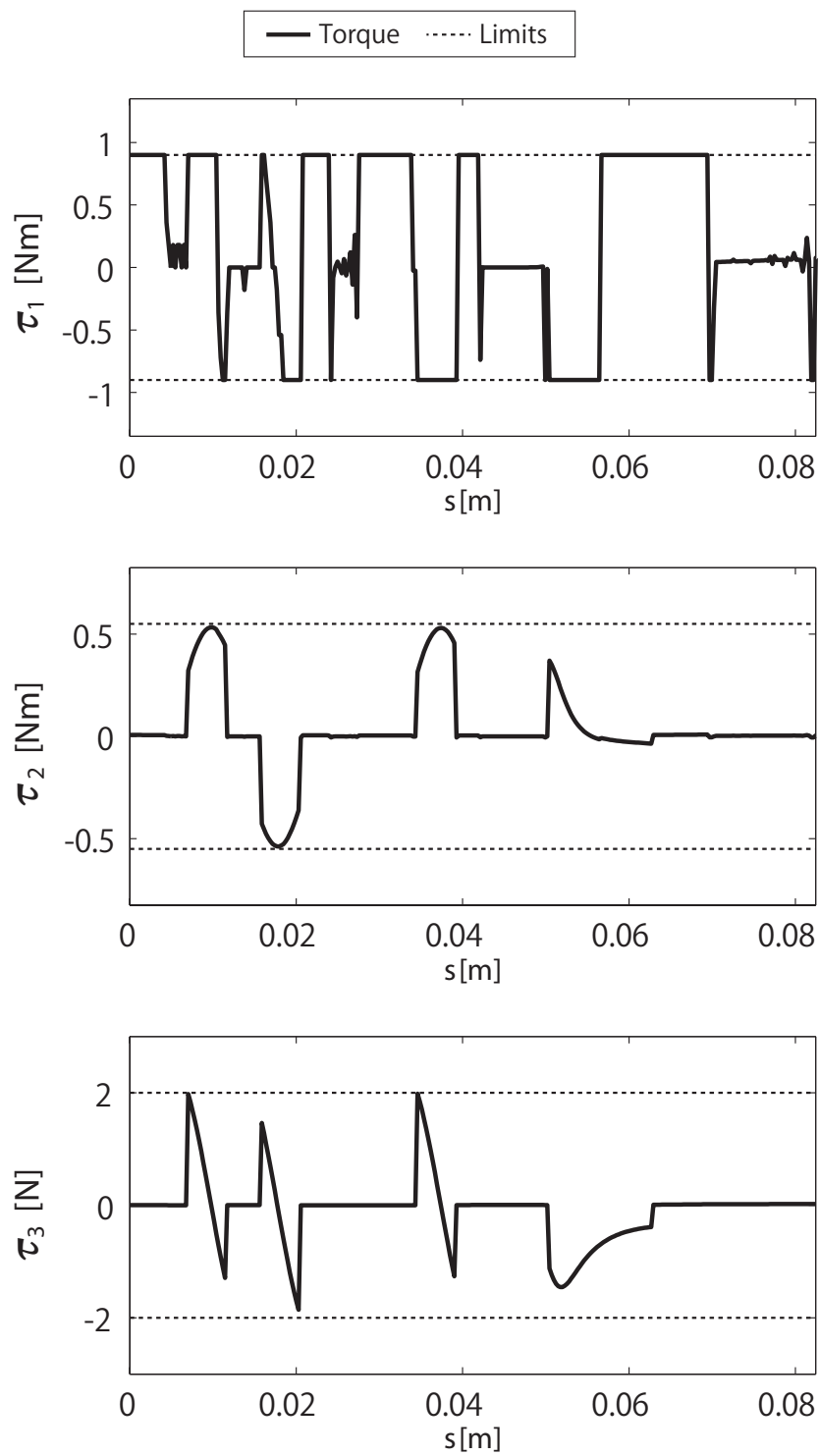


Fig. 7.7: 計画された動作の駆動トルク

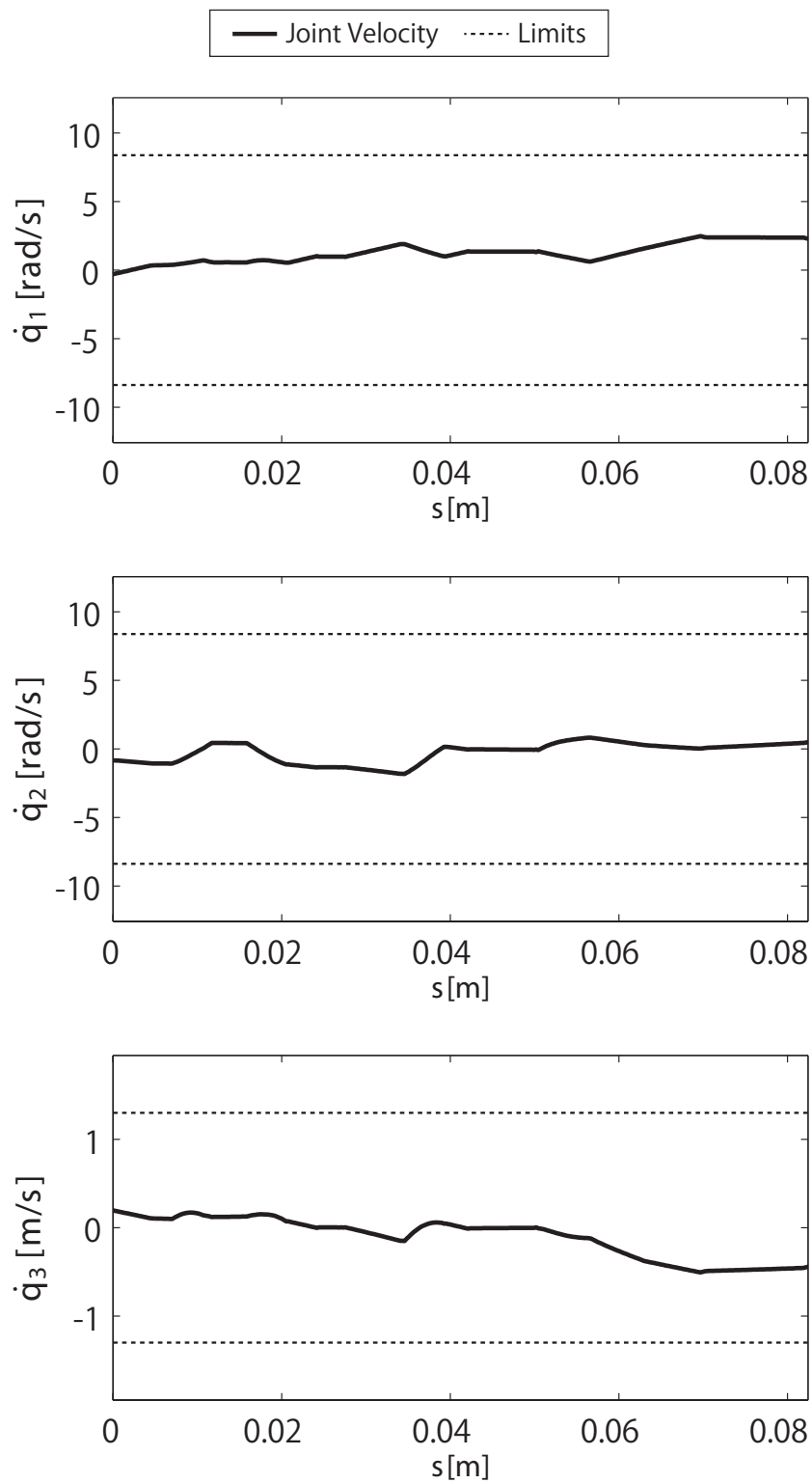


Fig. 7.8: 計画された動作の関節速度

7.3 リニアスライダ上に設置された垂直多関節マニピュレータへの適用

本節では Fig.7.9 で表されるモデルを有する垂直多関節型マニピュレータがリニアスライダ上に設置されたような構造を持つマニピュレータに対して動作計画を行う。マニピュレータのパラメータは Tab.7.3 に示す値を用いた。重力は $-z$ 軸方向に重力加速度 $9.8[m/s^2]$ とした。冗長姿勢 R_x は $\Sigma = [0, 1, 0, 0]^T$ とし、リニアスライダ上に設置された垂直多関節マニピュレータの旋回軸関節変位を冗長姿勢 R_x とした。動作計画は微小手先速度 $\delta s = 0.01[m/s]$ 、探索範囲 $R_x | 0 \leq R_x \leq \pi$, $R_v | -25 \leq R_v \leq 25$ の 17 段階として計画を行った。軌跡追従手先定速作業の手先軌跡は Fig.7.10 に示す軌跡で、Fig.7.11 でマニピュレータが追従している軌跡とした。

動作計画システムを用いて計画された最大手先速度と、MA 手法・MT 手法を用いた場合の最大手先速度を Tab.7.4 に、計画された動作を Fig.7.11 に、計画された冗長パラメータを Fig.7.12 に、状態空間上の経路と違反領域を Fig.7.13 に、計画された動作の駆動トルクを Fig.7.14 に関節速度を Fig.7.15 に示す。

Tab.7.4 の結果より、本動作計画を用いて冗長性を利用した動作を計画することで軌跡追従手先定速作業が高速化されていることが分かる。また、これらの結果より平面多関節マニピュレータだけでなく 3 次元の作業空間を持つマニピュレータに対しても本動作計画システムが適用可能で、重力による影響も考慮されることが分かる。

Tab. 7.4: 動作計画により得られた最大手先速度

	Maximum Hand Speed
Proposal Method	0.36 [m/s]
Motion by Minimum Joint Torque	0.23 [m/s]
Motion by Minimum Joint Acceleration	0.18 [m/s]

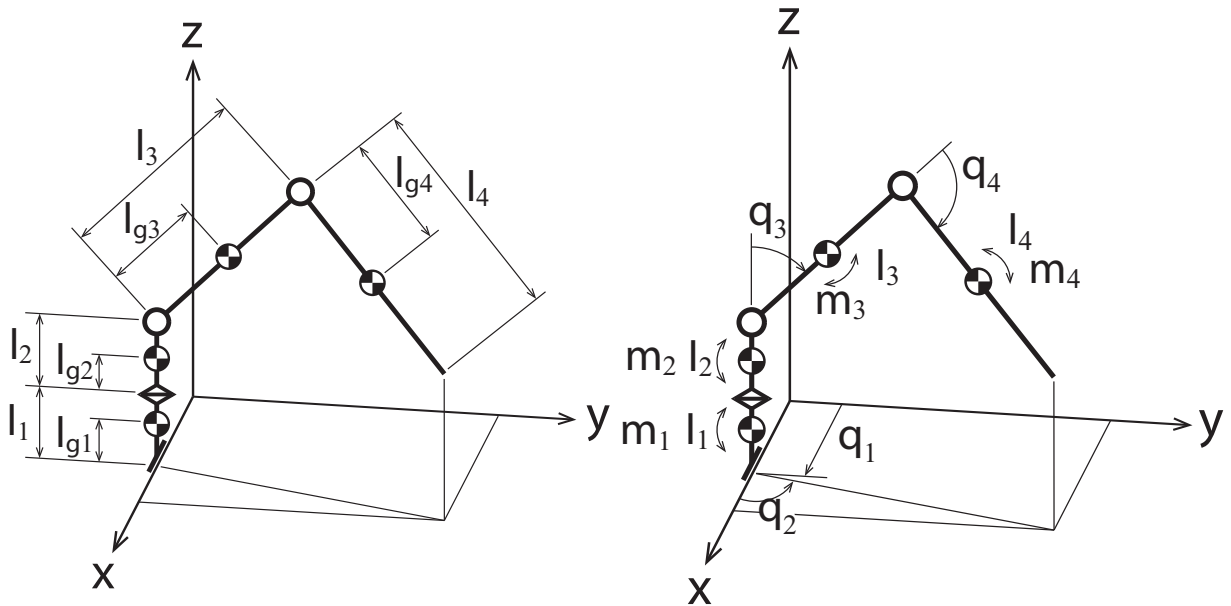


Fig. 7.9: リニアスライダ上に設置された垂直多関節マニピュレータのモデル

Tab. 7.3: 直動関節を有する水平マニピュレータの諸パラメータ

	Link1	Link2	Link3	Link3
Length of links	50 [mm]	50 [mm]	150 [mm]	150 [mm]
Mass of links	0.40 [kg]	1.18 [kg]	0.70 [kg]	0.70 [kg]
Length to center of gravity	25 [mm]	25 [mm]	75 [mm]	75 [mm]
Inertia of links	0 [mgm ²]	4200 [mgm ²]	3900 [mgm ²]	3900 [mgm ²]
Joint torque limits	80 [N]	2.03 [Nm]	2.03 [Nm]	1.03 [Nm]
Joint velocity limits	2.6 [m/s]	16.5 [rad/s]	16.5 [rad/s]	33.0 [rad/s]
Vector Σ	[0 1 0 0]			

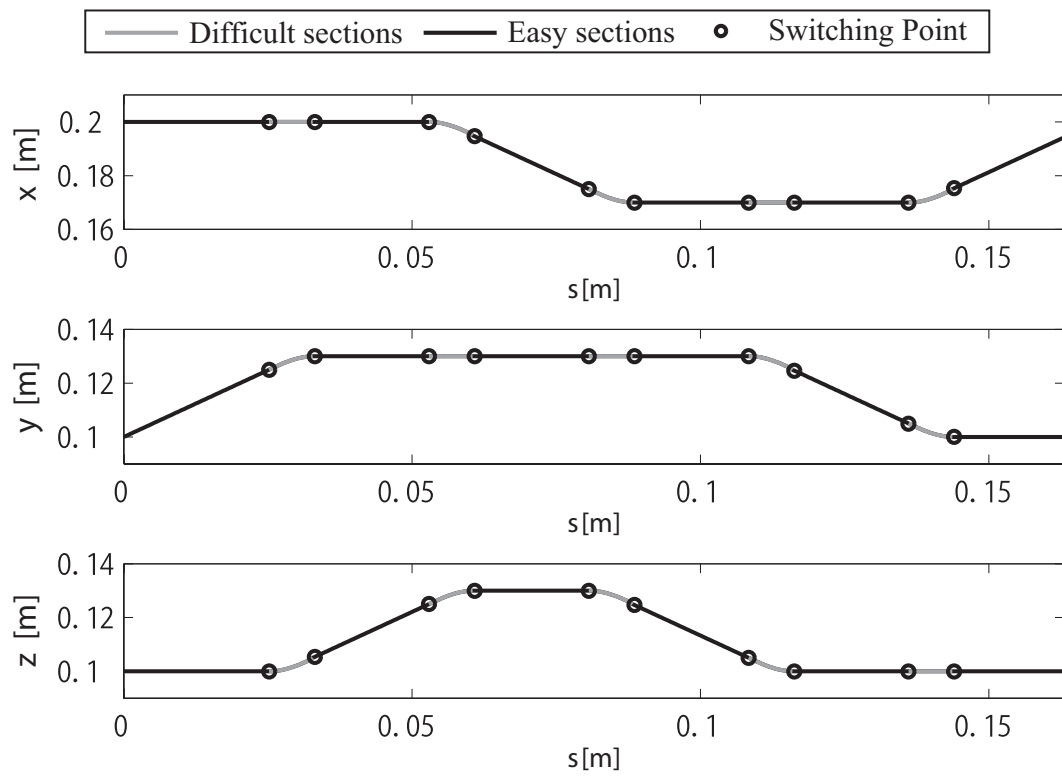
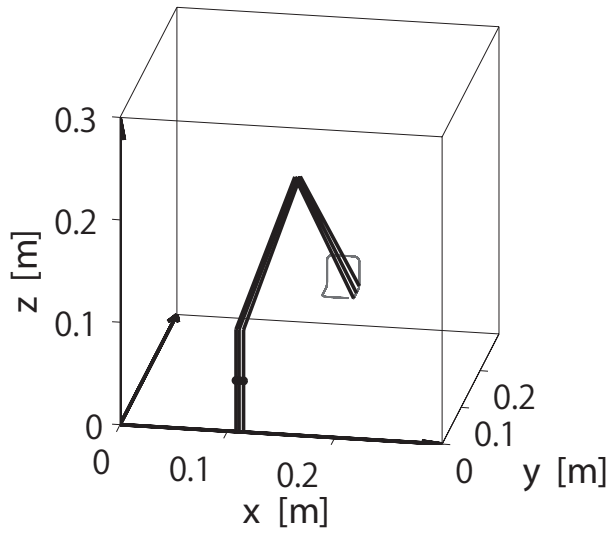
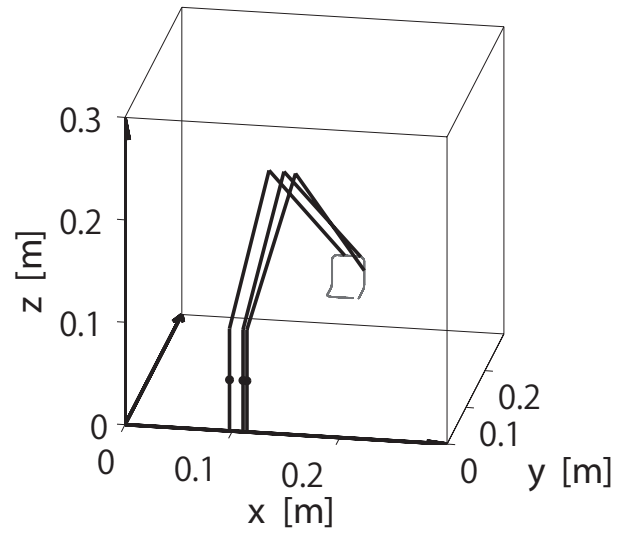


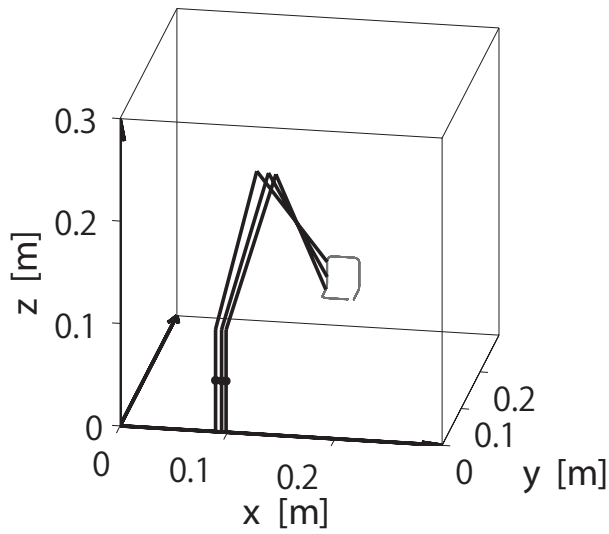
Fig. 7.10: 手先軌跡指令



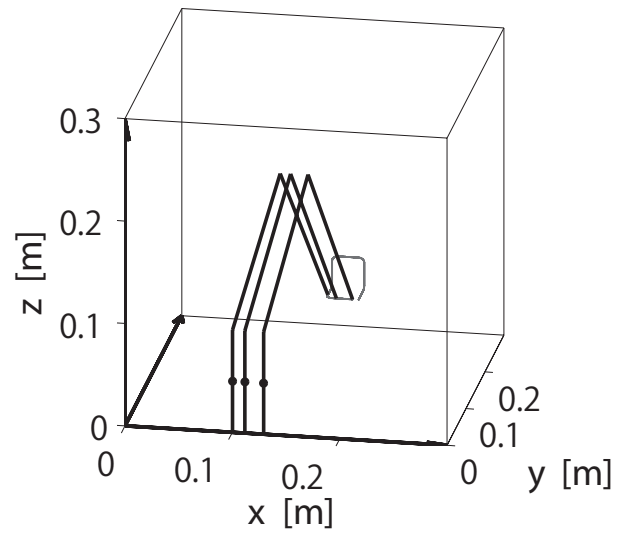
$s = 0, s = 0.030, s = 0.015$



$s = 0.044, s = 0.059, s = 0.074$



$s = 0.089, s = 0.104, s = 0.118$



$s = 0.133, s = 0.148, s = 0.163$

Fig. 7.11: 動作計画により得られた動作

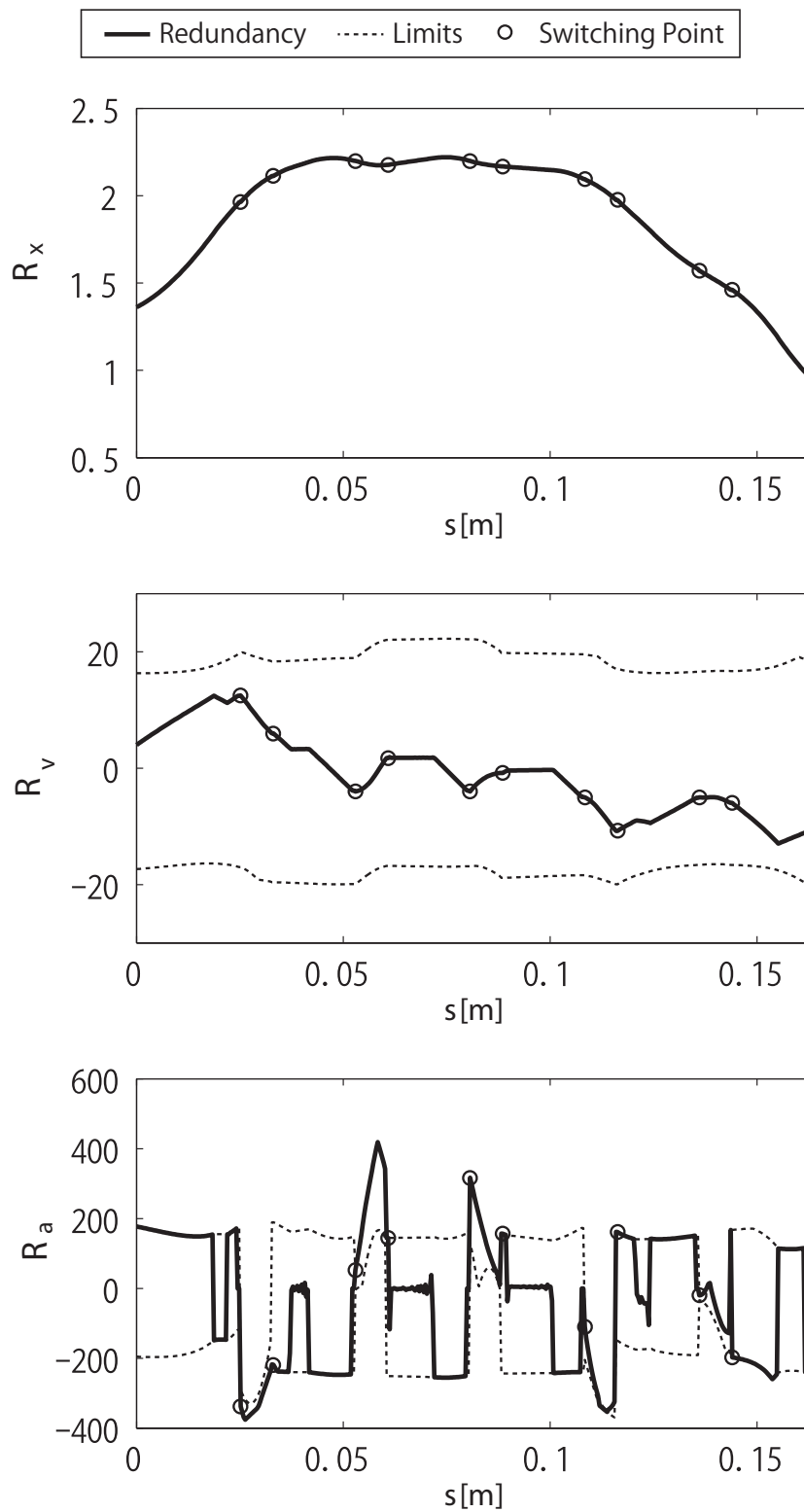
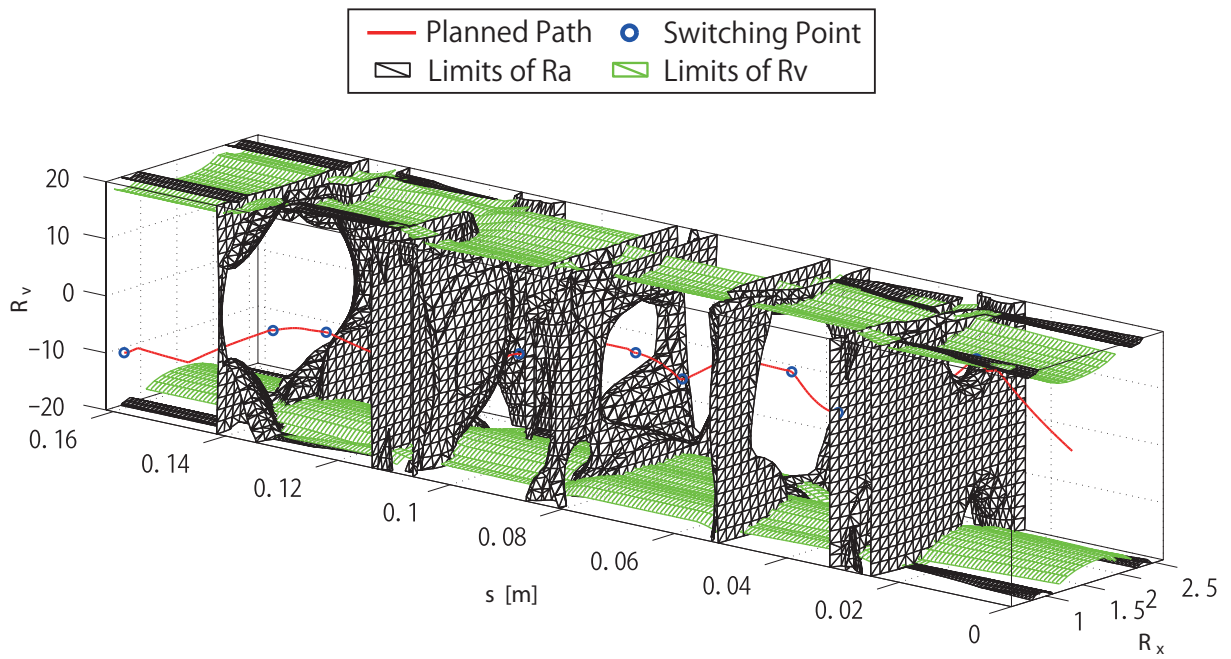
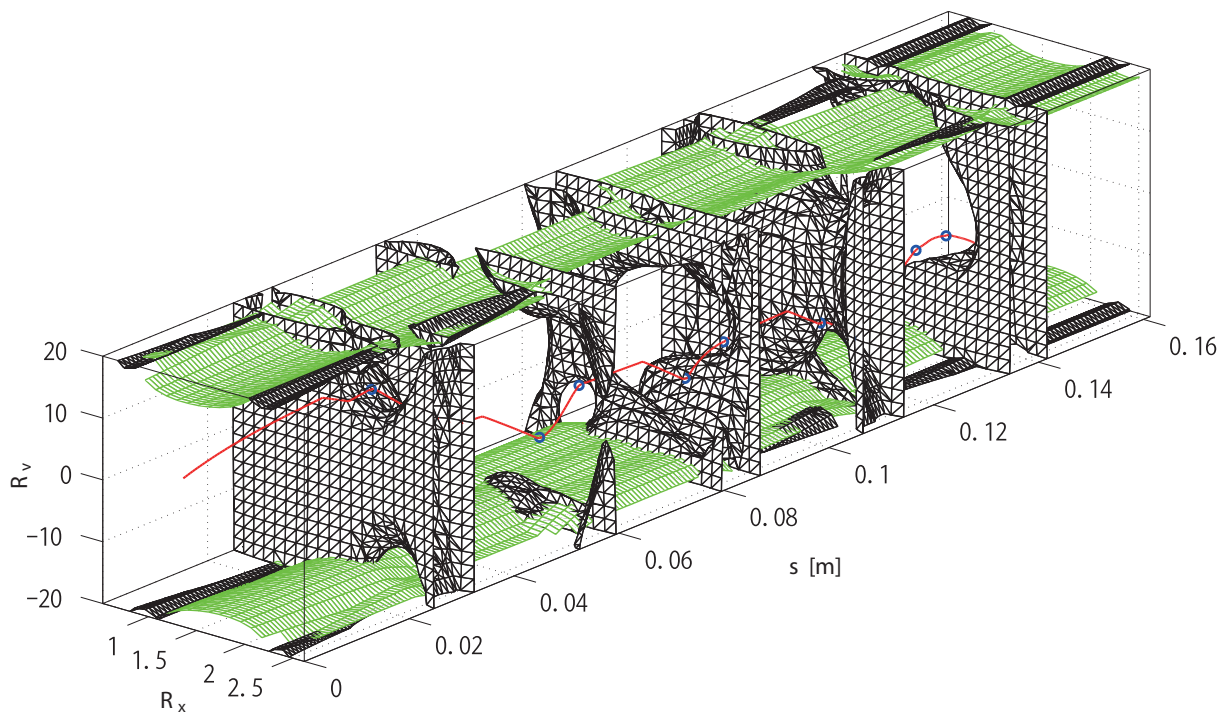


Fig. 7.12: 動作計画により得られた冗長パラメータ



状態空間-A



状態空間-B

Fig. 7.13: 状態空間上の経路と違反領域

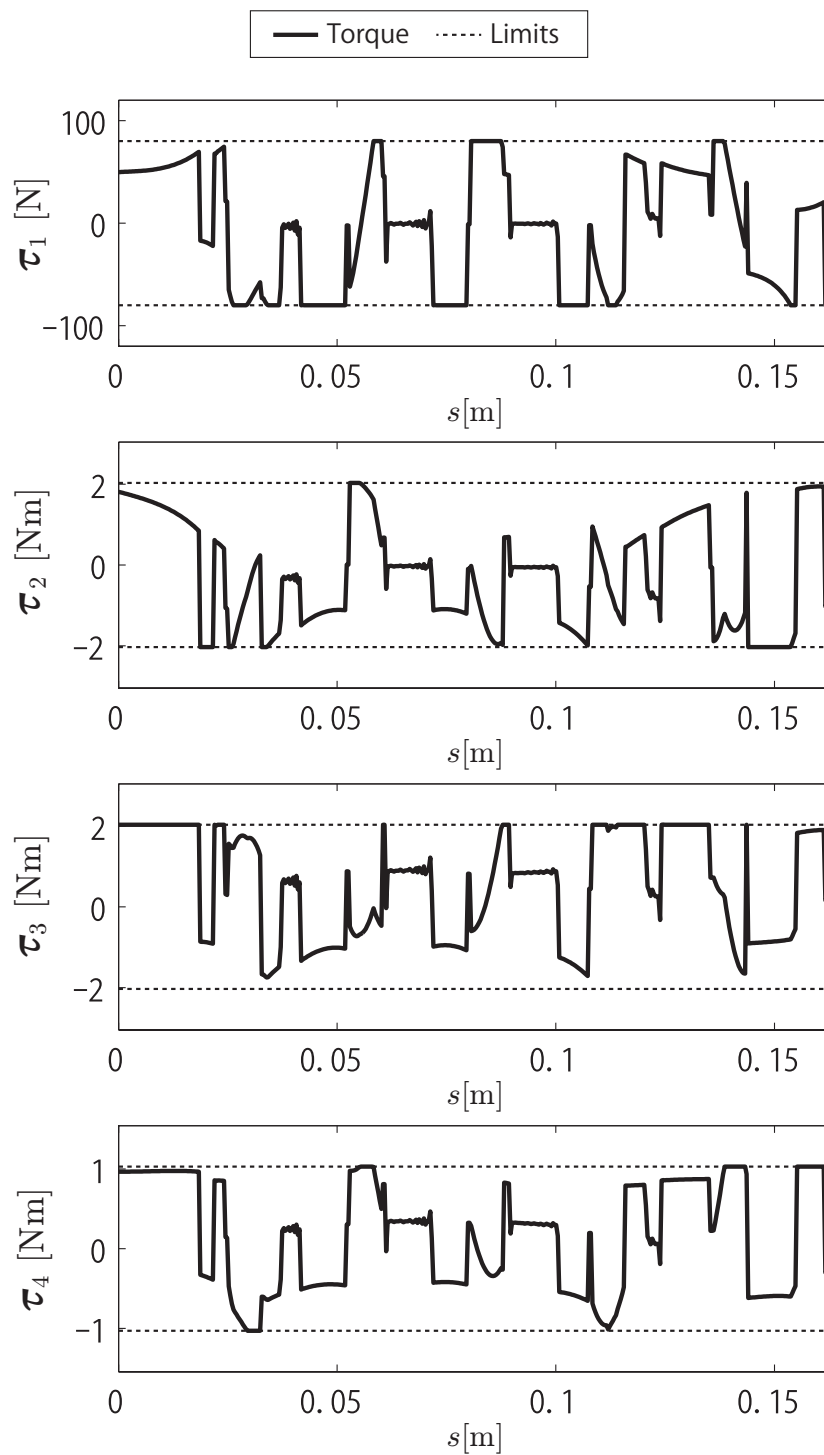


Fig. 7.14: 計画された動作の駆動トルク

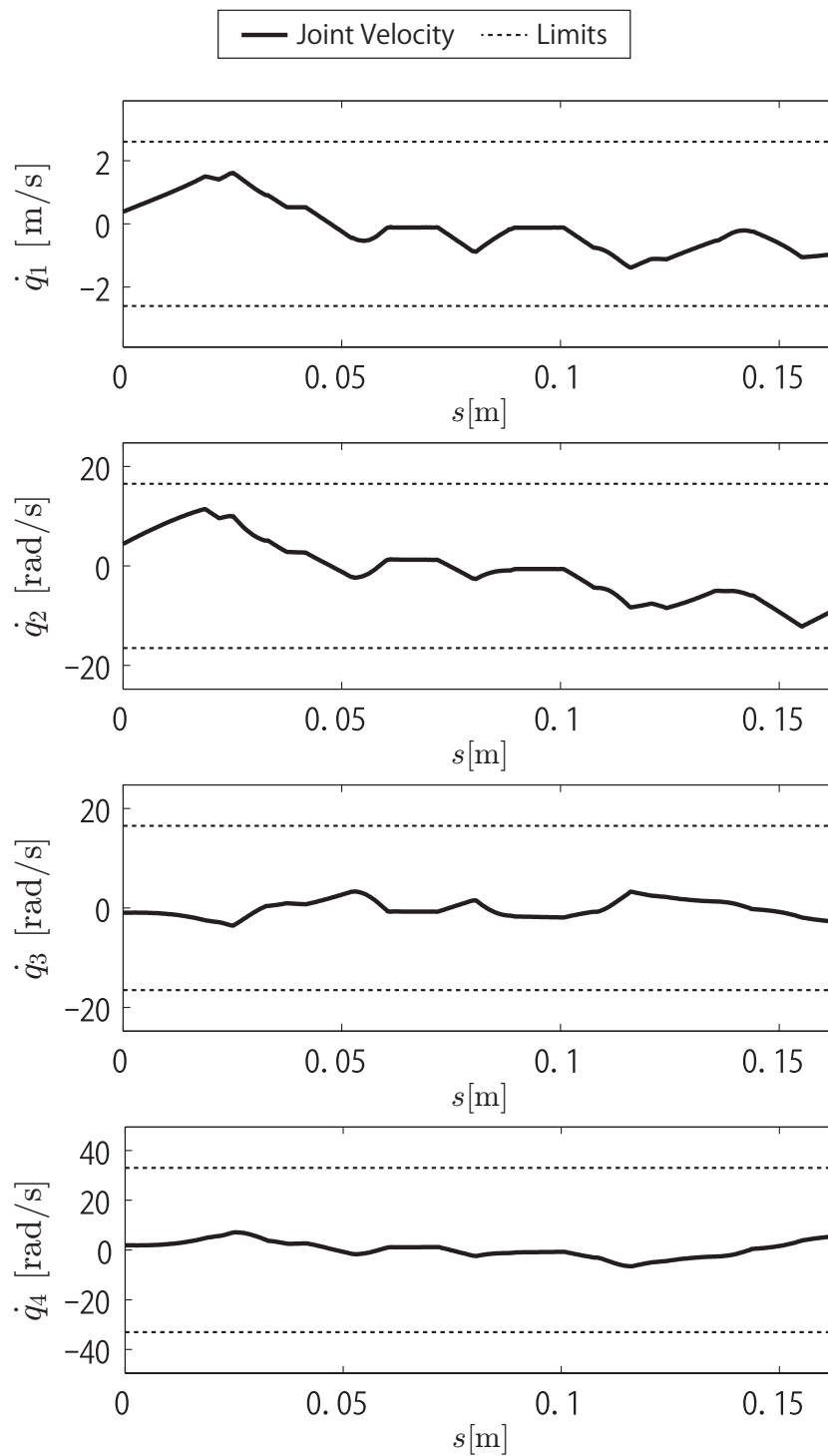


Fig. 7.15: 計画された動作の関節速度

7.4 手先軌道回転半径の最小化

軌跡追従手先定速作業では手先速度が一定で動作させる必要があるため、四角軌跡を描く場合に駆動トルク等の制限により角部を円弧軌跡等で補間した軌跡を用いて動作させる。産業界では四角軌跡を追従する作業は多く存在し、四角軌跡の角部はできるだけ直角に近い軌跡としたほうが好ましい。

本節では本動作計画システムを応用し、ある手先速度における軌跡追従手先定速作業を行う丸角四角軌跡角部の円弧軌跡の半径最小化を行う。円弧軌跡半径最小化は Fig.6.1 に示す動作計画アルゴリズムにおいて、手先速度 s を微小手先速度 δs 高速化する代わりに円弧軌跡半径 r^* を微小半径 δr^* 小さくする。具体的な動作計画アルゴリズムの違いは以下の点が最大手先速度を求める動作計画アルゴリズムと異なる。

1. Fig.6.1 の初期処理で行う“十分低速な s を設定”の代わりに軌跡追従手先定速作業を行う手先速度 s を設定する。そして、十分円弧半径 r^* の大きな丸角四角軌跡を手先軌跡とする。
2. Fig.6.1 のメインループ最後にある“ $s = s + \delta s$ ”の代わりに $r^* = r^* - \delta r^*$ として手先軌跡を円弧半径 r^* の丸角四角軌跡とする。

このような丸角四角軌跡の円弧半径を最小化するアルゴリズムを用いて、手先速度 0.30[m/s] で丸角四角軌跡を描く軌跡追従手先定速作業の角部軌跡をマニピュレータの冗長性を利用して最小化する。円弧半径最小化は Fig.4.1 に示すモデルを有する水平3関節冗長マニピュレータを対象として、1辺 50[mm] の正方形の角部を円弧軌道で補間した手先軌跡に対して円弧軌跡最小化を行う。マニピュレータのパラメータは Tab.4.1 に示す値とし、微小半径 $\delta r^* = 0.25$ [mm] とした。手先速度 $s = 0.30$ [m/s] における円弧半径最小化の結果、最小円弧半径は $r^* = 3.75$ [mm] となった。その時の手先軌跡を Fig.7.16 に示す。また、計画された動作の冗長マニピュレータを Fig.7.18 に、駆動トルクを Fig.7.19 に、関節速度を Fig.7.20 に示す。冗長性を利用した円弧軌跡最小化の効果を示すために、最小駆動トルクでマニピュレータを動作させる MT 手法を用いた場合の手先速度 $s = 0.30$ [m/s] での最小円弧半径は $r^* = 18.50$ [mm] となった。この時の手先軌跡を Fig.7.17 に示す。

Fig.7.16 と Fig.7.17 より，丸角四角軌跡の角部円弧軌跡はマニピュレータの冗長性を利用して小さい半径で動作可能であることが分かる．

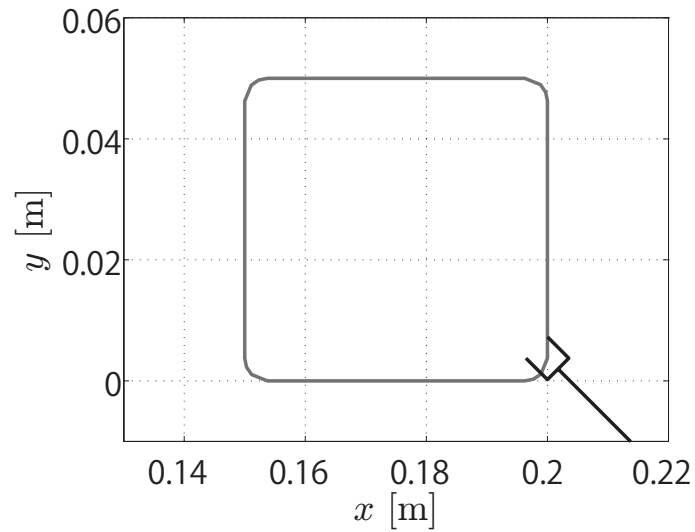


Fig. 7.16: 動作計画を用いた最小半径円弧軌跡

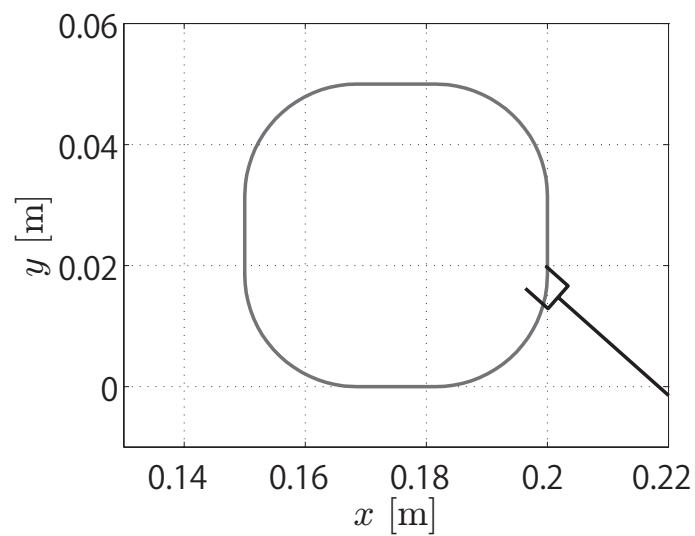


Fig. 7.17: MT 手法を用いた最小半径円弧軌跡

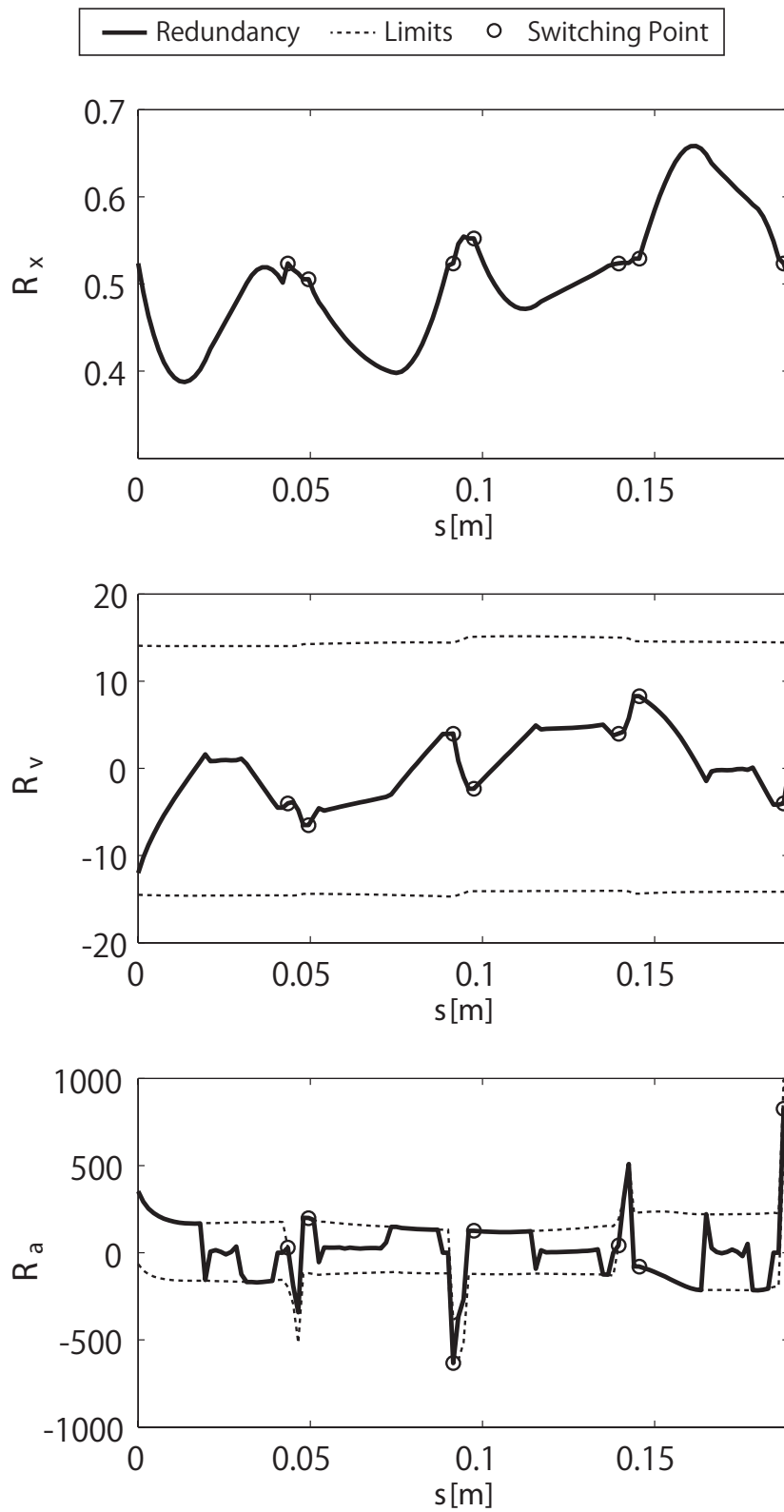


Fig. 7.18: 動作計画により得られた冗長パラメータ

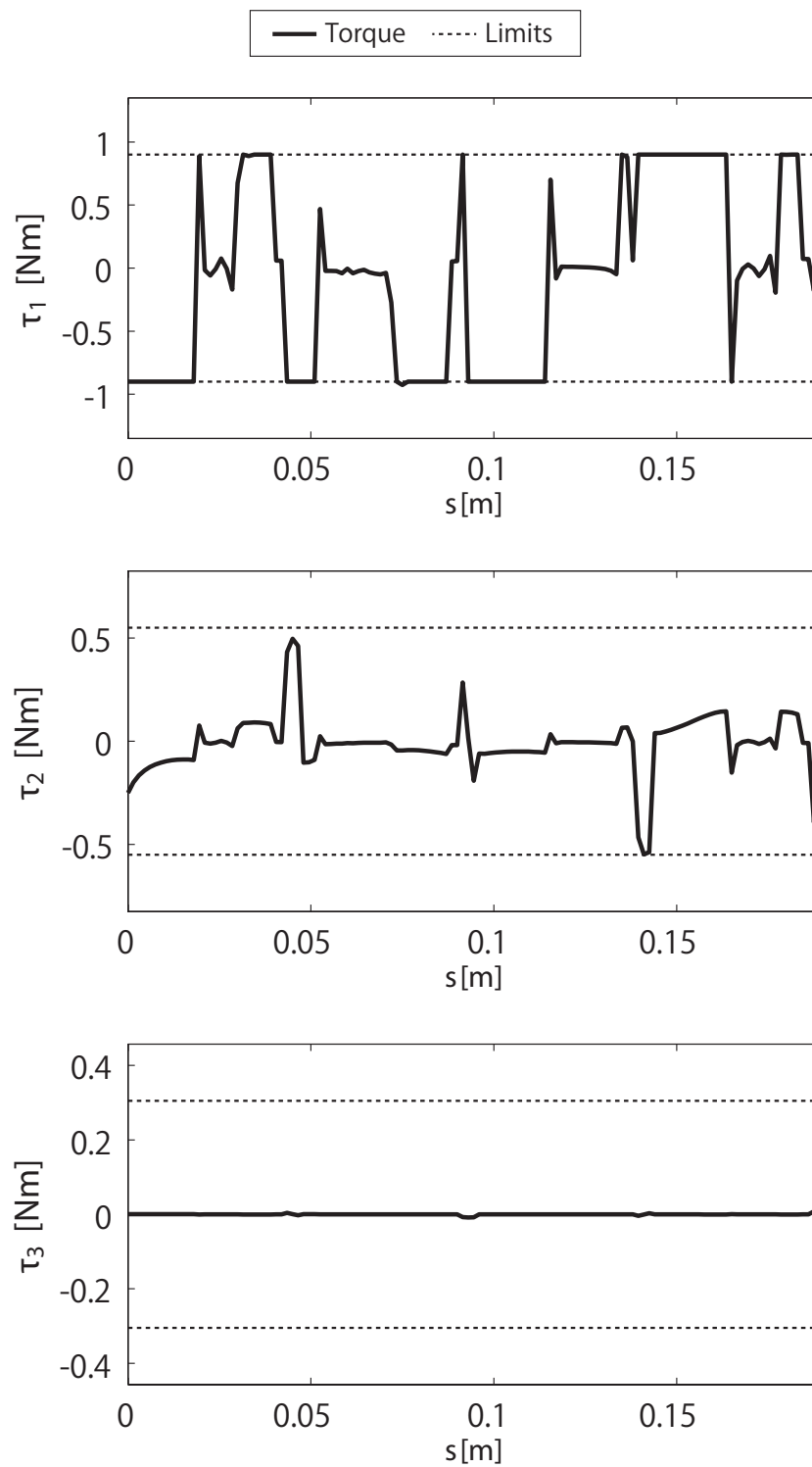


Fig. 7.19: 計画された動作の駆動トルク

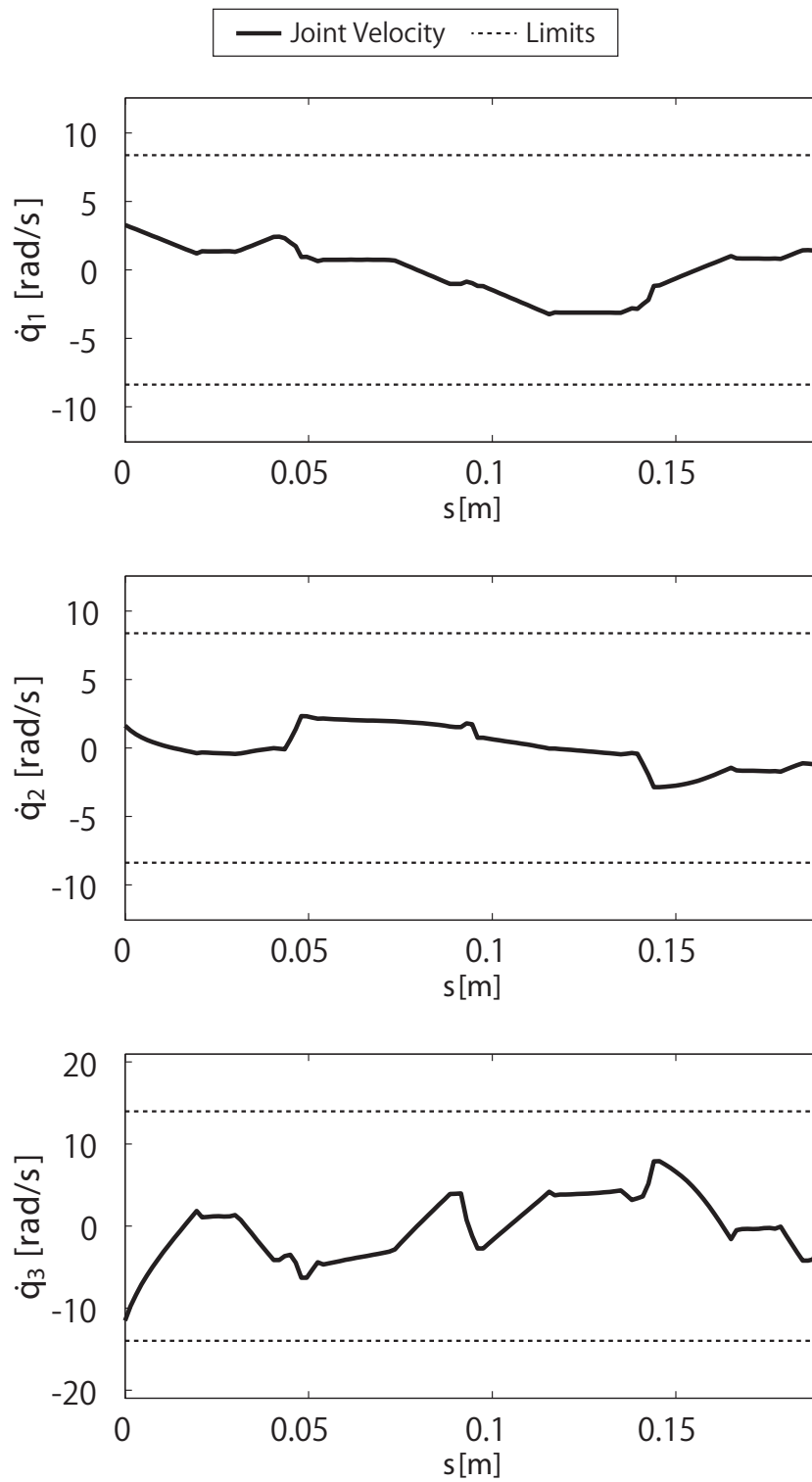


Fig. 7.20: 計画された動作の関節速度

7.5 終わりに

7章では、本動作計画システムを水平マニピュレータの手首部に直動関節を有する水平3関節冗長マニピュレータと垂直多関節型マニピュレータをリニアスライダ上に設置する構成のマニピュレータに対して適用し本動作計画システムの適用範囲を示した。また本動作計画システムを応用し、マニピュレータの冗長性を利用することで丸角四角軌跡を描く軌跡追従手先定速作業の角部円弧半径を最小化する動作計画システムについて述べた。

本研究では冗長自由度1を有する冗長マニピュレータを用いて与えられた手先位置指令を追従する作業に対して一般化しており、本章で示す動作計画システム適用例により平面関節型マニピュレータだけでなく直動関節を有する冗長マニピュレータや作業空間が3次元空間のマニピュレータに対しても適用可能であることを示した。また、本動作計画システムで計画しているマニピュレータの冗長性は、動作計画システムのメインループで更新する変数を変更することで軌跡追従手先定速作業の手先速度高速化以外にも、丸角四角軌跡の角部円弧軌跡半径最小化等にも応用可能になる。

第8章 結論

Contents

8.1	まとめ	105
8.2	今後の展望	107

8.1 まとめ

本論文では軌跡追従手先定速作業の高速化のために、冗長マニピュレータの有する冗長性を利用して駆動トルク制限，関節速度制限を回避する動作を計画する動作計画システムの構築と動作計画システムの有効性検証を行った．実行タスクとしては冗長自由度1を有する冗長マニピュレータを用いて手先位置指令軌跡を一定手先速度で追従する軌跡追従手先定速作業を対象とした．

2章では冗長マニピュレータの冗長性を利用した動作を計画するために，冗長自由度1を有する冗長マニピュレータの定式化を行ない，マニピュレータの順運動学関数から手先の運動に関する式と冗長性による動作の項を含む関節の運動に関する式を導出した．また，マニピュレータの動力学方程式の定式化と各種制限の定式化を行った．次に，軌跡追従手先定速作業におけるマニピュレータの満たすべき条件を示した．最後に冗長マニピュレータの有する冗長性を利用して駆動トルク制限，関節速度制限を回避する動作を計画する動作計画システムの概要について述べた．

3章では，2章で定式化した冗長マニピュレータの関係式より，マニピュレータの冗長性による動作を手先位置に影響を与えない冗長性に姿勢パラメータである冗長姿勢，手先速度に影響を与えない冗長性による関節速度パラメータである冗長速度，手先加速度に影響を与えない冗長性による関節加速度パラメータである冗長加速度の3つのスカラー値で定義する式の導出を行ない，冗長姿勢と冗長速度を状態量，冗長加速度を入力とした状態方程式の導出を行った．次に，手先軌跡上の手先位置と冗長姿勢，冗長速度を軸とする状態空間を考え，状態空間上に駆動トルク制限と関節速度制限の射影を行ない状態空間上の制限を表す違反領域の導出を行った．これにより，冗長マニピュレータの手先動作とは独立した冗長性による動作を表す状態空間が構成され，状態空間上にマニピュレータの制限を表す違反領域が射影されたことにより動作計画問題を状態空間上の経路探索問題に帰着することができた．

4章では動作計画システムの一部である手先軌跡の高手先速度動作が困難な区間における状態空間上の経路探索手法である難区間経路探索について述べた．難区間経路探索は区間の初端状態に関する探索と，初端状態から違反領域に進入することなく終端手先位置

まで到達する経路の探索の2つで構成され、初端状態からの経路探索は評価関数を用いて初端状態から逐次的に経路を決定していく探索を行ない、初端状態の探索は十分低速な手先速度から違反領域に進入することなく終端手先位置まで到達する経路の存在する初端状態の範囲を絞り込むことで探索を行う。そして難区間経路探索で用いる評価関数の設計を行った。最後に難区間経路探索の一例を示し、難区間経路探索により状態空間上の違反領域を回避する経路が計画されることを示した。これにより、軌跡追従手先定速作業の高速化において高速化の妨げとなるクリティカルポイントを含む区間の高速化が可能となった。

5章では、4章で述べた難区間経路探索により計画された経路を繋ぐ経路を探索する易区間経路探索について述べた。易区間経路探索では区間の終端状態に必ず到達する経路で構成される切換面を用いて探索を行うため、5章ではまず切換面の導出と、切換面の構成に必要な駆動トルクベクトルノルムが最小となる冗長加速度の導出を行った。次に、切換面を用いた易区間経路探索アルゴリズムについて説明し、最後に易区間経路探索の一例を示すことで易区間経路探索により3章で求めた状態方程式を満たしつつ区間の初端状態と終端状態を繋ぐ経路が計画されることを示した。これにより、高手先速度動作が困難な区間を優先的に計画し、高手先速度動作が容易な区間では困難な区間で計画された動作の準備動作を行うことが可能となった。

6章では、4章と5章で説明した経路探索手法を用いた高速な軌跡追従手先定速作業動作を計画する動作計画システムについて説明した。次に、シミュレーションを用いて一般的な冗長マニピュレータの動作制御手法である関節加速度最小となる動作を用いた最大手先速度動作と駆動トルク最小動作を用いた最大手先速度動作の2つの手法と最大手先速度を比較することで、動作計画システムを用いることによりこれら一般的な動作制御手法より高速な軌跡追従手先定速作業動作が計画されていることを示した。最後に、関節加速度最小となる動作の冗長加速度と駆動トルク最小となる動作の冗長加速度は動作計画システムの探索空間内に含まれることを示すことで、理論的に動作計画システムを用いることでこれら一般的な動作制御手法より高速な軌跡追従手先定速作業動作が計画されることを示した。これにより、高速な軌跡追従手先定速作業動作の計画が可能となった。

7章では、6章で説明した動作計画システムを平面多関節マニピュレータの手首部に直

道関節を有する冗長マニピュレータや垂直多関節マニピュレータがリニアスライダ上に設置された構成の冗長マニピュレータに対して適用することで、動作計画システムが冗長自由度1を有する冗長マニピュレータを用いて手先位置指令軌跡を追従する軌跡追従手先定速作業について一般化されていることを示し、動作計画システムを応用した軌跡追従手先定速作業における丸角四角軌跡の角部円弧軌跡半径を最小化する動作計画手法とその計画例について述べた。

以上のような手先位置指令軌跡を追従する高速な軌跡追従手先定速作業動作を計画する動作計画システムは一般的な動作制御手法より高速な動作が計画されることを理論的、数値解的に示した。また、動作計画システムは軌跡追従手先定速作業高速化のような Bang-Bang 制御を用いて高速動作が計画できない作業に対して応用可能であることを示した。

8.2 今後の展望

本動作計画システムは冗長自由度1を有する冗長マニピュレータを用いて手先位置指令軌跡を追従する軌跡追従手先定速作業について一般化されているが、手先位置指令軌跡と共に手先姿勢指令が与えられる指令は動作計画システムの一般化の範囲に含まれていない。そのため、手先位置指令軌跡と共に手先姿勢指令が与えられる指令に対しても一般化した動作計画システムに拡張する余地が残っている。また、実際のマニピュレータにはマニピュレータによって関節変位制限が存在するため、関節変位制限を状態空間上に射影し経路探索を行うことも必要になってくると考えられる。

謝辞

本論文の審査の主査を務めてくださった、

筑波大学 システム情報系 相山康道 准教授

に心より感謝申し上げます。先生の存在無くして本研究の成果、本論文の完成はありませんでした。私の提案する難解な手法を理解し、論理の矛盾や考察漏れを的確に指摘していただき誠にありがとうございました。先生からは研究に対する姿勢や心構え、重要視すべき点など極めて多くのことを教わりました。

また、副査を勤めてくださった、

筑波大学 システム情報系 堀憲之 教授

筑波大学大学院 システム情報工学研究科 望山洋 准教授

筑波大学 システム情報系 藪野浩司 教授

中央大学 理工学部 大隅久 教授

からは厳しい御指摘や貴重な御助言を頂きました。謹んで感謝の意を表します。各分野の第一人者である先生方からのご意見は大変参考になりました。特に中央大学理工学部の大隅久教授には、研究室の御縁で本研究の最初期段階から何度も相談に乗っていただき、誠にありがとうございました。

博士前期課程においてご指導いただいた前・九州工業大学情報工学部制御システム工学科教授の黒龍一先生からは、応用制御に関する知識と実際にものを制御する技術を頂いただけでなく、私が研究者となるきっかけを下さいました。誠にありがとうございました。

また、研究に関する様々な問題や提案手法の相談に乗っていただいた、筑波大学大学院システム情報工学研究科知能機能システム専攻のマニピュレーション・システム研究室の皆様へ深く感謝いたします。

最後に、本研究を進めるにあたりご理解とご協力をしてくださった皆様、自分のやりたいようにやりなさいと温かく見守って下さった両親に深く感謝するとともに謝辞とさせていただきます。

参考文献

論文

- [1] 富山 和照, “ディスペンサに見る最新技術動向”, エレクトロニクス実装学会誌, Vol.7, No.6, pp.501-505, (2004)
- [2] 島野 一郎, “高密度実装に求められるディスペンス技術”, エレクトロニクス実装学会誌, Vol.11, No.6, pp.451-455, (2008)
- [3] 白濱 和人, 蟹谷 清, “溶接ロボット”, 日本ロボット学会誌, Vol.27, No.3, pp.274-277, (2009)
- [4] 張 曉林, 藤堂 勇雄, “距離情報と濃淡情報の統合に基づく継目位置検出法を用いたシーリングロボットの制御”, 日本機械学会論文集, Series C, Vol.60, No.577, pp.3130-3137, (1994)
- [5] 荒川 賢一, 柿崎 隆夫, 大名 新治, “トラッキングロボットによるシーリング作業のための作業目標点高速検出法”, 日本ロボット学会誌, Vol.16, No.3, pp.376-383, (1998)
- [6] 柿崎 隆夫, 柳原 義正, 荒川 賢一, 梅野 光, 望月 研二, 大名 新治, “車体シーリングロボットのための高速倣い視覚センシングシステムの開発”, 日本機械学会論文集, Series C, Vol.64, No.628, pp.4737-4744, (1998)
- [7] 黄 健, 藤堂 勇雄, 常磐 孝徳, “シーリングロボットの目標軌道の実時間生成法と制御”, 日本機械学会論文集, Series C, Vol.66, No.648, pp.2669-2676, (2000)
- [8] 清水 湧一, “センサ利用によるアーク溶接の自動化”, 日本ロボット学会誌, Vol.9, No.3, pp.374-378, (1991)

- [9] H.Chen, J.Li, G.Xing, J.Xing and H.Sun, “Trajectory Tracking Control of a Macro-Micro Welding Robot Based on the Vision Navigation”, *Robotics and Biomimetics (ROBIO), 2010 IEEE International Conference on*, pp.667-672, (2010)
- [10] C.S.Liu and H.Peng, “Disturbance Observer Based Tracking Control”, *Journal of Dynamics Systems, Measurement, and Control*, Vol.122, Issue.2, pp.332-335, (2000)
- [11] Y.Tang, “Terminal Sliding Mode Control for Rigid Robots”, *Automatica*, Vol.34, No.1, pp.51-56, (1998)
- [12] 陳 芸峰, 池田 弘行, 美多 勉, 涌井 伸二, “スライディングモード制御によるロボットアームの軌道制御と実験結果”, *日本ロボット学会誌*, Vol.7, No.6, pp.706-711, (1989)
- [13] Q.P.Ha, D.C.Rye and H.F.Durrant-Whyte, “Fuzzy moving sliding mode control with application to robotic manipulators”, *Automatica*, Vol.35, Issue.4, pp.607-616, (1999)
- [14] 吉川 恒夫, “ロボットアームの動的可操作性”, *計測自動制御学会論文集*, Vol.21, No.9, pp.970-975, (1985)
- [15] S.L.Chiu, “Task Compatibility of Manipulator Postures”, *The International Journal of Robotics Research*, Vol.7, No.5, pp.13-21, (1988)
- [16] 倉爪 亮, 長谷川 勉, “シリアルリンクマニピュレータのインピーダンスマッチング”, *日本ロボット学会誌*, Vol.23, No.2, pp.1-9, (2005)
- [17] 見浪 護, 内藤 康弘, 朝倉 俊行, “冗長マニピュレータの回避可操作性”, *日本ロボット学会誌*, Vol.17, No.6, pp.887-895, (1999)
- [18] 浅田 晴比古, “ロボットアーム動特性の幾何学的解析法”, *計測自動制御学会論文集*, Vol.19, No.6, pp.500-505, (1983)
- [19] H.Tanaka, M.Minami and Y.Nae, “Trajectory Tracking of Redundant Manipulators Based on Avoidance Manipulability Shape Index”, *Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on*, pp.4083-4088, (2005)

- [20] J.E.Bobrow, "Optimal Robot Path Planning Using the Minimum-Time Criterion", *IEEE JOURNAL OF ROBOTICS AND AUTOMATION*, Vol.4, No.4, pp.443-450, (1988)
- [21] 加島 正, 石動 善久, "動作時間を考慮した消費エネルギー最小規範によるマニピュレータの軌道計画", 日本ロボット学会誌, Vol.15, No.7, pp.1012-1018, (1997)
- [22] J.E.Bobrow, S.Dubowsky and J.S.Gibson, "Time-Optimal Control of Robotic Manipulators Along Specified Paths", *The International Journal of Robotics Research*, Vol.4, No.3, pp.3-17, (1985)
- [23] K.G.Shin and N.D.Mckay, "Minimum-Time Control of Robotic Manipulators with Geometric Path Constraints", *IEEE TRANSACTION ON AUTOMATIC CONTROL*, Vol.AC-30, No.6, pp.531-541, (1985)
- [24] S.Ma and M.Watanabe, "Time Optimal Path-Tracking Control of Kinematically Redundant Manipulators", *JSME International Journal, Series C*, Vol.47, No.2, pp.582-590, (2004)
- [25] M.Galicki, "Time-Optimal Controls of Kinematically Redundant Manipulators with Geometric Constraints", *IEEE Transactions on Robotics and Automation*, Vol.16, No.1, pp.89-93, (2000)
- [26] S.Ma, "A Balancing Technique to Stabilize Local Torque Optimization Solution of Redundant Manipulators", *Journal of Robotic Systems*, Vol.13, No.3, pp.177-185, (1996)
- [27] Y.Zhang, D.Guo and S.Ma "Different-Level Simultaneous Minimization of Joint-Velocity and Joint-Torque for Redundant Robot Manipulators", *Journal of Intelligent and Robotic Systems*, Vol.72, Issue.3-4, pp.301-323, (2013)
- [28] 吉川 恒夫, "冗長性を有するロボットの制御", 日本ロボット学会誌, Vol.2, No.6, pp.587-592, (1984)

- [29] T.Yoshikawa, "Analysis and Control of Robot manipulators with Redundancy", *Proc. of 1st International Symposium of Robotics Research*, Cambridge, MA, MIT Press, pp.735-747, (1983)
- [30] 馬 書根, 広瀬 茂男, 吉灘 裕, "冗長マニピュレータの効率的な冗長制御法", 日本ロボット学会誌, Vol.14, No.5, pp.703-709, (1996)
- [31] 中村 仁彦, 花房 秀郎, "関節型ロボットアームの最適冗長制御", 計測自動制御学会論文集, Vol.21, No.5, pp.501-507, (1985)
- [32] O.Egeland, "Task-Space Tracking with Redundant Manipulators", *IEEE JOURNAL OF ROBOTICS AND AUTOMATION*, Vol.RA-3, No.5, pp.471-475, (1987)
- [33] 花房 秀郎, 吉川 恒夫, 中村 仁彦, "多関節ロボットアームの冗長性の解析とその優先順位を有する作業への応用", 計測自動制御学会論文集, Vol.19, No.5, pp.421-426, (1983)
- [34] K.C.Park, P.H.Chang and S.Lee, "Analysis and control of redundant manipulator dynamics based on an extended operational space", *Robotica*, Vol.19, Issue.6, pp.649-662, (2001)
- [35] P.H.Chang, "A Closed-Form Solution for Inverse Kinematics of Robot Manipulators with Redundancy", *IEEE JOURNAL OF ROBOTICS AND AUTOMATION*, Vol.RA-3, No.5, pp.393-403, (1987)
- [36] S.Lee, and A.K.Bejczy, "Redundant Arm Kinematic Control Based on Parameterization", *Proceeding of the 1991 IEEE International Conference on Robotics and Automation*, pp.458-465, (1991)
- [37] 比留川 博久, "経路探索問題-ロボットの動作計画-", 情報処理, Vol.35, No.8, pp.751-760, (1994)
- [38] 有田 俊作, 上野 誠也, "定速移動体の最適軌道計画と軌道追従に関する研究", 航空宇宙技術, Vol.13, pp.25-32, (2014)
- [39] 大須賀 公一, 前田 浩一, "マニピュレータの動特性同定法", 計測自動制御学会論文集, Vol.22, No.6, pp.637-643, (1986)

- [40] 村上 俊之, 大西 公平, “外乱オブザーバによる多自由度ロボットの動特性同定法”, 日本ロボット学会誌, Vol.11, No.1, pp.131-139, (1993)
- [41] 橋本 秀紀, “VSS によるモーションコントロール”, 日本ロボット学会誌, Vol.11, No.4, pp.515-520, (1993)

書籍

- [42] 吉川 恒夫, 『ロボット制御基礎論』, コロナ社, (1988)
- [43] 田辺 行人, 品田 正樹, 『理・工基礎 解析力学』, 裳華房, (1988)
- [44] 柳井 晴夫, 竹内 啓, 『射影行列・一般逆行列・特異値分解』, 東京大学出版, (1983)

その他

- [45] ヘイシンロボティクスペンサー 吐出の羅針学, <http://www. robo-dispenser.com/compass/>

研究業績

本論文に関連するもの

査読あり学術論文

- Okabe Kousuke and Aiyama Yasumichi, 「Fast Motion of Path Tracking Task with Constant Hand Speed using Redundant Manipulator」, 『Journal of Robotics and Mechatronics』, (Vol.27 No.1 に掲載予定)

査読あり国際会議における発表

- ○ Kousuke Okabe, Yasumichi Aiyama, “Efficient Motion Planning for Fast Motion of Path Tracking Task with Constant Hand Speed using Redundant Manipulator”, 『Proceeding of 2014 IEEE/SICE International Symposium on System Integration』 (SII2014), pp.40-45, December, 2014
- ○ Kousuke Okabe, Yasumichi Aiyama, “Motion Planning of a Redundant Manipulator for the Purpose of Speed-Up of the Hands Constant Speed Tasks”, 『Proceeding of 2013 IEEE/SICE International Symposium on System Integration』 (SII2013), pp.539-544, December, 2013

査読あり国内会議における発表

- ○岡部弘佑, 相山康道, 「冗長マニピュレータによる高速な軌跡追従手先定速作業の効率的な動作計画」, 『第20回ロボティクスシンポジウム』, 軽井沢プリンスホテルウエスト, 3月2015年, (採択済み)

口頭発表

- ○岡部弘佑, 相山康道, 「冗長マニピュレータによる高速な軌跡追従手先定速作業の効率的な動作計画 第1報, 一軌道分割を用いた経路探索と実機による検証実験」, 『第32回日本ロボット学会 学術講演会』, 3 N 2-04, 九州産業大学, 9月2014年
- ○岡部弘佑, 相山康道, 「冗長マニピュレータによる高速な軌跡追従手先定速作業の効率的な動作計画 第2報, 一曲率による軌道分割と分割された軌道に対する経路計画」, 『第32回日本ロボット学会 学術講演会』, 3 N 2-05, 九州産業大学, 9月2014年
- ○岡部弘佑, 相山康道, 「冗長マニピュレータによる高速な軌跡追従手先定速動作の軌道分割を用いた動作計画」, 『ロボティクス・メカトロニクス講演会2014 in Toyama』, 1P1-Q01, 富山市総合体育館, 5月2014年
- ○岡部弘佑, 相山康道, 「冗長マニピュレータのトルク・角速度制限を考慮した高速な手先定速作業の動作計画」, 『第31回日本ロボット学会 学術講演会』, 1H1-04, 首都大学東京 南大沢キャンパス, 9月2013年
- ○岡部弘佑, 相山康道, 「手先定速作業の高速化を目的とした冗長マニピュレータの動作計画」, 『ROBOMECH2013 in TSUKUBA』, 2P1-J04, つくば国際会議場, 5月2013年

本論文に直接関連しないもの

査読あり学術論文

- L.Lan, R.Oguro and K.Okabe, 「A Simple Interpolation Methodology of Position Command in High-speed and High-accuracy Control for Industrial Robots」, 『高速信号処理応用技術学会誌』, 第 14 巻, 第 2 号, pp.23-30, (2011)

査読あり国際会議における発表

- ○ K.Okabe, R.Oguro, and L.Lan, 「A Control Proposal for Circle Contour Manufacturing in Rotary Coordinate System」, 『Proceeding of the 4th International Conference on Intelligent Network and Intelligent Systems (ICINIS2011)』, The Intelligent Networks and Systems Society (INASS), pp.29-32, 2011

口頭発表

- ○岡部弘佑, 小黒龍一, 蘭林峰, 「オブザーバを用いたイナーシャ誤差, 粘性摩擦係数, クーロン摩擦係数の推定手法」, 『平成 24 年電気学会産業応用部門大会 (JIASC2012)』, 2-21, 千葉工業大学 津田沼キャンパス, 8 月 2012 年
- ○岡部弘佑, 小黒龍一, 蘭林峰, 「オブザーバを用いたイナーシャ誤差及び摩擦外乱の同時推定に関する研究」, 『平成 23 年電気学会産業応用部門大会 (JIASC2011)』, 2-32, 琉球大学 千原キャンパス, 9 月 2011 年
- ○蘭林峰, 小黒龍一, 岡部弘佑, 「モデル追従制御を用いた 2 リンクマニピュレータの非干渉制御」, 『平成 23 年電気学会産業応用部門大会 (JIASC2011)』, 2-27, 琉球大学 千原キャンパス, 9 月 2011 年
- ○岡部弘佑, 小黒龍一, 「パラメータ推定オブザーバに関する研究」, 『第 29 回計測自動制御学会九州支部学術講演会』, 202A1, 宮崎大学工学部 木花キャンパス, 12 月 2010 年

- ○岡部弘佑, 小黒龍一, 「力覚センサレス力制御に関する基礎研究」, 『ROBOMECH2010 in ASAHIKAWA』, 1P1-F13, 旭川大雪アリーナ, 6月2010年