

冗長系逆運動学マップ計算の並列分散アーキテクチャ

Parallel Decentralized Computation Architecture for Calculating an Inverse Kinematics Map of a Redundant Robot System

安藤 英由樹¹⁾ 羅 志偉²⁾ 平松 誠治 加藤 厚生

Hideyuki ANDO¹⁾, Zhi-Wei LUO²⁾, Seiji HIRAMATSU, Atsuo KATO

Abstract: A high-speed computation algorithm and a developed hard-ware architecture are described in this paper which are used for solving the inverse kinematics method of a redundant robot by a nonlinear mapping method. The nonlinear mapping algorithm computes the nonlinear map between fingertip positions and joint angles of the robot by a diffusion-based parallel dispersed method. However, to compute the high density nonlinear map by this algorithm, there are two problems, that is enormous computing time and many processors are required. In this paper, we propose one way of computing high density map with the reasonable number of processors and computing time by the divided parallel dispersed method. Also, we have designed a hard-ware architecture and composed parallel CPU computer system and then made some experiments to evaluate the performance of this system.

1. はじめに

ロボットアームの逆運動学解を求める問題は、ロボットの制御において重要課題の一つとして研究されている。この分野では、アームの正確な幾何学モデルを前提に、方程式を解くことによってその解を求める方法が検討されてきた。また、次世代ロボットとして自律型ロボット¹⁾が考えられている。自律型ロボットとは、変化する環境の中で自ら判断しながら目的を遂行するための行動を行うロボットである。このようなロボットでは、環境の変化や障害物に対応する柔軟性を要求されるため、人間のように関節数に冗長性が必要であると考えられている¹⁾。ところが冗長性を有するロボットアームの逆運動学解を求める問題は一般的な幾何学モデルを前提とした方程式では解くことが困難である。この問題に関し、冗長ロボットに与えられた作業に優先順位を付けたサブタスクに

分解し順次実行する方法^{2) 3)} や、逆運動計算を解く必要のないインピーダンス制御を用いる方法^{4) 5)} が検討されている。しかしいずれも、計算時間がかかる問題をかかえている。また、一方で正確な幾何学モデルを持っていても学習によって適応させるニューラルネットワークを用いた学習制御の応用による解法が報告されているが^{6) 7) 8)}、各ニューロン間の大域的な結合が必要で、現実的なハードウェア技術では実現不可能となっている。

これらに対して、拡散方程式と誤差修正式を併用した非線形マッピングの並列分散計算方式⁹⁾ は、マップの中の各格子点は隣接する点だけの情報で計算を進めるため、各々の格子点は大域的な結合を必要としない。また、誤差収束についても非常に良好な結果が得られている。本研究ではこの方式を採用した。

ところがこの方式を、実際のロボットに適用するためには、格子間隔の密なマップを必要とする。格子点の数だけプロセッサを配置すると、格子間隔が密になるほど膨大なプロセッサが必要となり現実性に乏しい。また、

1) 愛知工業大学 電気電子工学専攻 (豊田市)

2) 理化学研究所 バイオミメティク
コントロール研究センター (名古屋市)

拡散にかかる時間も格子数に比例して増大するため, 計算に膨大な時間がかかる。

本研究では, 冗長性を有するロボットに実用できる高密度な非線形マップを並列分散方式で構成するための計算アーキテクチャについて考察する。

ここでは, 格子の疎分割と区域分割をおこない, 実現可能な台数のプロセッサによる並列分散処理についての設計と製作を行った結果を報告する。

2. 冗長ロボットの定義

本研究では冗長ロボットを次のように定義する。すなわち, ロボットの作業に必要な空間ベクトル $\mathbf{x} = (x_1, x_2, \dots, x_m)^T$ の次元数 m とロボットの関節ベクトル $\theta = (\theta_1, \theta_2, \dots, \theta_n)^T$ の次元数 n の関係が $m < n$ となる場合に冗長である。

3. マップ学習アルゴリズム

ここでは, 拡散方程式と誤差修正式を併用した非線形マッピング並列分散方式について述べる。この方法は拡散方程式を計算する拡散ステップと誤差修正式を計算する誤差修正ステップからなり, その特徴として, 1) オフラインでマップ計算をおこなうため制御時には計算時間がかからない。2) わずかな教師情報のみで学習が可能である。3) 拡散ステップ時に逆ヤコビ行列の計算が可能である。4) 拡散ステップの計算には隣接した局所情報のみを必要とし大域的な情報を必要としない, が挙げられる。以下では2次元平面上でのマップ作成の例を挙げる。

図1に示される2次元平面に3つのリンク (L_1 , L_2 , L_3) と3つの回転関節 θ をもつロボットがある。この平面上に $(0, 0)$ から (m, n) までのマップを計算する。このマップ上にロボットの手先位置と重なる格子点位置に関する関節角度を収納する。たとえば, 図1の格子点 (i, j) には, 関節角度 θ_1 , θ_2 , θ_3 が収納されることになる。

このマップ条件は1) 空間的に姿勢が連続である。2) 手先位置と目標位置である格子点との間に誤差がないこ

とである。

マップ計算の始めとして, 4隅の格子点に教師情報となる関節角度を与える。この教師情報から拡散ステップによって関節角度情報(姿勢)を均等に拡散していくため, 姿勢は教師情報から一意に決まる。

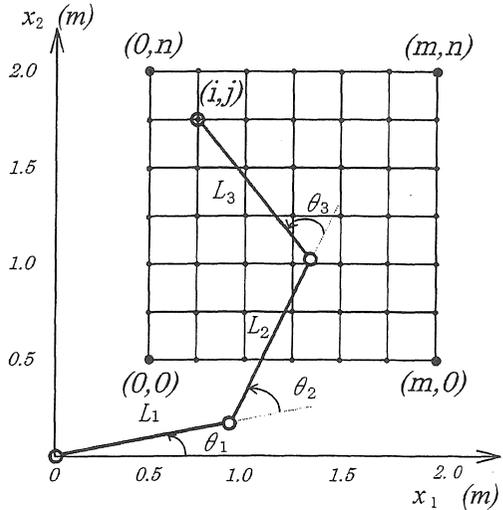


図1. 関節角度のマッピング

3.1 拡散ステップ

拡散方程式は次式で与えられる。

$$\nabla_x^2 \theta = 0 \quad (3.1)$$

ここで, $\theta = (\theta_1, \theta_2, \theta_3)^T$ はロボットの関節角度で, $\mathbf{x} = (x_1, x_2)^T$ は作業空間である。この式により, あらかじめ作業空間に与えられた教師情報でかこまれた空間で関節角度(姿勢)は均一に拡散され, 連続的な分布が得られる。(3.1)式を離散的に書き換えると次式になる。

$$\theta_{i,j}^{t+1} = \frac{1}{4} \{ \theta_{i-1,j}^t + \theta_{i+1,j}^t + \theta_{i,j-1}^t + \theta_{i,j+1}^t \} \quad (3.2)$$

ただし, t は学習ステップで i, j は作業空間における, 1つの格子が占める位置を表わす。

また拡散過程における各学習ステップの差 $\Delta \tilde{\theta}^t = \theta^t - \theta^{t-1}$, $\Delta \tilde{\mathbf{x}}^t = \mathbf{x}^t - \mathbf{x}^{t-1}$ には次式の関係があることから,

$$\dot{\theta} = A \dot{\mathbf{x}} \quad (3.3)$$

ここで, A は逆ヤコビ行列と呼ばれ, 次式により求めることができる。

$$A_{i,j}^{t+1} = A_{i,j}^t + \frac{\varepsilon}{\|\Delta\tilde{x}_{i,j}^t\|^2} (\Delta\tilde{\theta}_{i,j}^t - A_{i,j}^t \Delta\tilde{x}_{i,j}^t) \Delta\tilde{x}_{i,j}^{tT} \quad (3.4)$$

この A は、次の誤差修正ステップで使用できる。ここで、 ε は A の形成を制御する係数である。

以上により、拡散方程式から連続的な姿勢を得ることはできた。しかし拡散は、線型的な空間補間に過ぎず、 t の極限でも手先の位置は目標位置に近づくだけで、目標に一致するわけではない。そこで、次の誤差修正ステップで誤差を収束させる。

3.2 誤差修正ステップ

誤差を小さくする為の角度の修正式は次式で与えられる。

$$\theta_{i,j}^{t+1} = \theta_{i,j}^t + \beta A_{i,j}^t \{x_{i,j}^d - f(\theta_{i,j}^t)\} \quad (3.5)$$

この式において、拡散ステップで計算した逆ヤコビ行列 A を用いることで、連続した姿勢を変化させないで誤差をなくすることができる。ここで、 $x_{i,j}^d$ は目標位置（格子点の位置）で、 $f(\theta_{i,j}^t)$ は関節角度から求めた手先位置を示す。また、 β は修正の重み係数である。

この2つのステップにより、空間的に連続でありかつ、誤差がないマップを得ることができる。

この過程を例示すれば図2となる。①始めに4隅に教師情報を与える。②拡散ステップにより連続的な姿勢が求まっているが、手先位置と目標位置には誤差が残っている。③誤差修正ステップにより連続な姿勢で誤差のないマップが求まる。

以上の手法の特徴を以下に挙げる。拡散方程式から、各格子点上の関節角度は、隣接する格子情報のみで学習が可能である。拡散を行う際に、誤差修正式使用可能な逆ヤコビ行列 A を学習する。

この手法より例示した点では良好な結果を得ることができる。しかし、実際のロボットに適用するためにはさらに高密度の高いマップを必要とする。そのため、この手法をそのまま用いると、膨大な個数のプロセッサを必要とし、また、膨大な計算時間がかかり、現実的ではない。

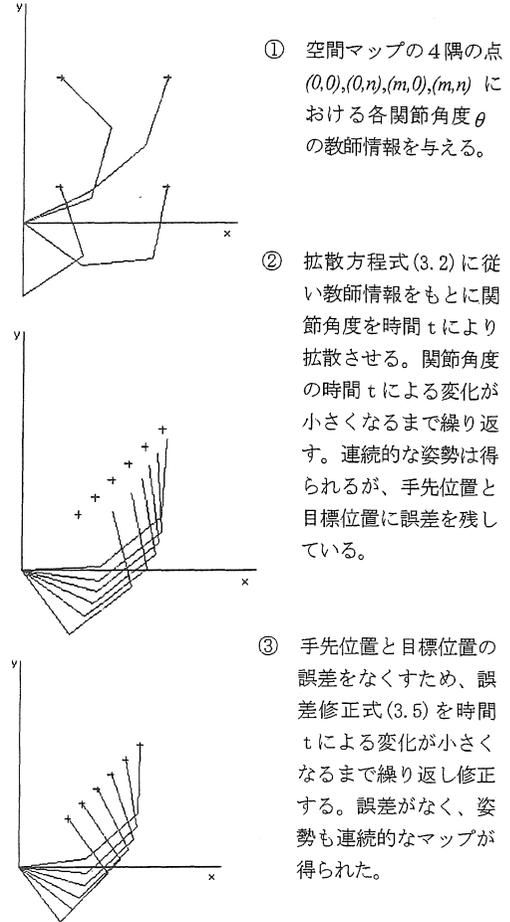


図2. 計算過程での様子

4. 並列分散アーキテクチャ

前章のマッピング学習アルゴリズムでは高密度なマップを得るためには膨大な時間がかかり、また大量のプロセッサを必要とした。本稿では、この問題解決に、プロセッサ数と計算時間を現実的な値とする方法について、学習区域の分割化及び、分割後それぞれを並列計算する並列化の2側面から検討を行った結果について報告する。

4.1 分割による計算時間短縮

ここでは、分割化と計算時間短縮について実験による検討を行った結果について述べる。計算時間短縮のためにまず、マップ計算においてどの行程が一番時間がかかるか検討を行った。その結果、拡散ステップにおいて、

変化が収束するまでにかかる繰り返し回数とその計算時間が計算時間の多くの割合を占めていることがわかった。格子数が増えるにしたがって繰り返し回数はべき乗で増えていくことから、計算時間も同等の増加傾向をもつことがわかる(図3参照)。つまり、格子数が増えるほど繰り返し回数が増えるため、粗なマップに分ける方がよい。

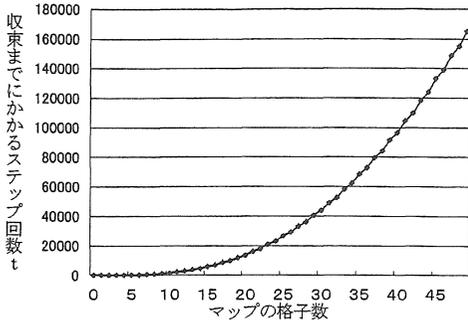
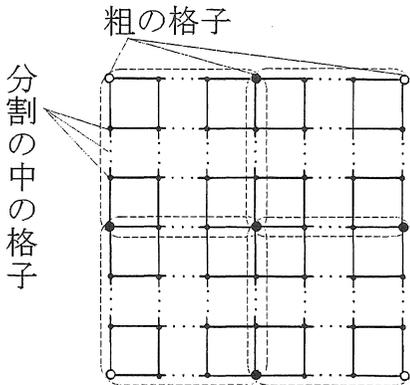


図3. 拡散ステップにおいて関節角度変化の収束間までの計算回数と格子の関係

一般に領域を分割するときは境界での連続性が問題となる。ここでも境界でいかに連続性を保って拡散を行うかが問題となる。そこで、本稿では次に示す分割方式を提案する。

4.2 格子数分割アルゴリズム

マップ作成の計算時間短縮のために考案した、格子数分割アルゴリズムについて説明する。



1. 始めに教師情報○から粗な格子●からなるマップを計算する
2. 1.で計算したマップ●を教師情報にして分割の中の格子・を計算する

図4. 分割による手法(4分割の例)

図4に例示する4分割マップについて、まず教師情報を白丸のように与える。そして前述の方法で黒丸大を格子点とする荒く粗なマップを計算する。そして、得られた結果を新しい教師情報として、これを境界とした小さいマップに分割

する。分割したマップを各々計算し、最終的に結合し大きなマップに復元する。このとき、境界の格子点は重なり合うことになる。

このアルゴリズムにより格子数41×41(区間40×40)の分割マップ計算を行った結果を表1に示す。ここでは、並列化は行っておらず、1台のコンピュータを用いて、分割計算後、各々小分けしたマップを逐次計算し、最終的にかかった時間と繰り返し回数を示した。

前述した分割手法は4分割を例示したが、各辺を均等区間に分割する場合、16分割、25分割、64分割、100分割、400分割の6通りの分割方法があり、それによっても計算時間が変わってくる。表から、格子数41×41のマップの場合は、格子数64分割を行ったとき最も計算時間が短くなることから、最適な分割方法であり、分割なしと比べて、1/100以下程度の計算時間で終了することがわかる。

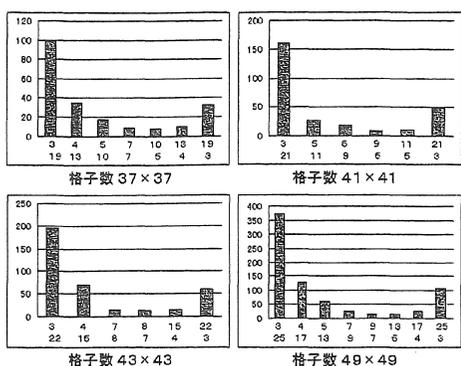
表1. 分割の方法の違いによる計算時間と計算回数

粗の格子数	分割の中の格子数	計算時間[s]	計算回数[回]
2(分割なし)	41×41	974.76	176673100
3(4分割)	21×21	166.91	29110500
5(16分割)	11×11	28.12	5048600
6(25分割)	9×9	17.80	3263400
9(64分割)	6×6	8.79	1627200
11(100分割)	5×5	9.61	1814600
21(400分割)	3×3	50.75	9076500

41×41マップを作成するための分割のシミュレーション

つぎに、他の格子数サイズのマップについても、同様に最適な分割数についての検討を行った。このときの計算時間について図5に示す。この図5より、分割の方法で計算時間が変わり、どの格子数サイズのマップについても最適な分割数が存在することがわかる。小さなマップに分割すればするほど計算時間が短くなるわけではない。この理由として、1) 分割数が増えるにつれ分割にかかる時間が多くなる。2) 分割したマップの計算は、

境界の格子点は2重に計算しているので、分割が多くなるほど無駄な計算が多くなるためであると考えられる。



縦軸 一時間 (s)
横軸 上段一分割格子数 下段一分割中格子数
図5. 分割方法と計算時間

分割化による計算時間短縮について結果を図6に示す。分割を行った場合は、図5から最良な分割方法を採用した。

分割を行うことで計算時間が短縮されることが分かる。以上により、マップの分割化は計算時間短縮に有効である。

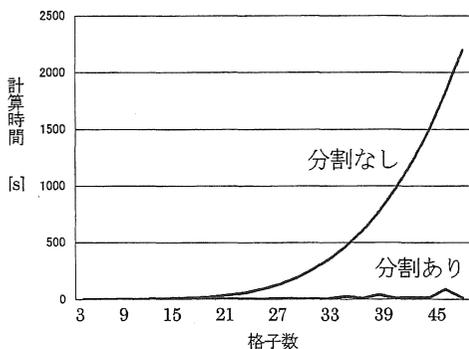


図6. 1台の計算機で分割計算した場合の効果

4.3 分割後のマップの並列計算

前項では、1台のプロセッサを用いた場合の分割の有効性について述べた。そこで次に、ここではさらに分割後の小さなマップについて、複数のプロセッサで並列に計算した結果に述べる。

アルゴリズムについて図7の4分割の例を挙げて説明

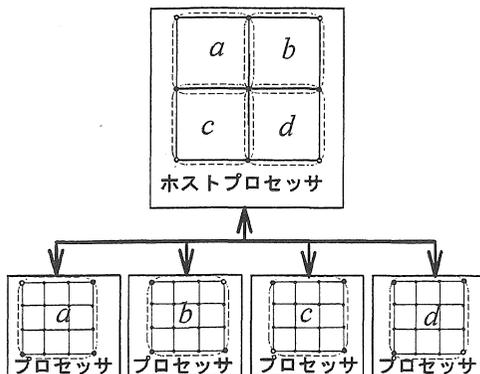


図7. マップの並列計算

まず、ホストプロセッサで荒い粗のマップを計算し、得られた結果を教師情報として密な分割を行う。分割された密なマップを下位のプロセッサで計算する。このとき、通信を必要とする情報は最初に与える教師情報のみでよく、下位のプロセッサ間では通信を行う必要がない。この方法は通信時間短縮に有利である。

4.4 並列計算アーキテクチャ

以上のアルゴリズムで計算をおこなう並列計算機のハードウェアのアーキテクチャについて設計、製作を行った。その模式図を図8に示す。

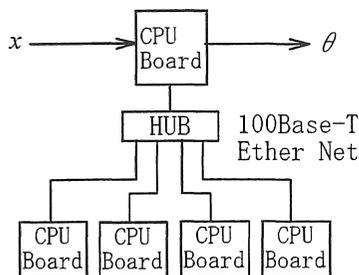


図8. ハードウェアのアーキテクチャ

計算機のCPU、マザーボードは市販品を用いた。通信には100base/TのEthernetを用いた。なお、模式図ではホストプロセッサを加えて5台のプロセッサを必要としているが、ホストプロセッサも分割計

算の後には下位のプロセッサとして用いることができるので、実際には4台のプロセッサを用いた。

並列計算機を用いた場合と1台の計算機で計算した場合の計算時間の比較を行った。マップの一边の格子数とマップ生成までの計算時間の関係は図9となった。どちらも最適な分割を行っている。

並列計算により、さらに計算時間短縮ができることが分かる。

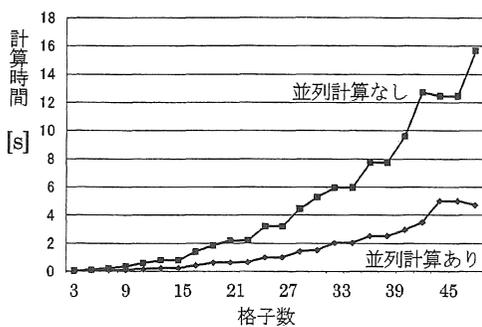


図9. 4台の並列計算による計算時間短縮

図9の結果から1台のプロセッサと4台のプロセッサでの計算時間の比を求めた結果を図10に示す。計算時間の比はマップの一边の格子数により変化があるものの3から3.5倍の計算時間短縮性が認められる。4倍にならない理由として、最初に必要とする分割のための計算は1台でしか行えないこと、通信時間がかかることが原因である。

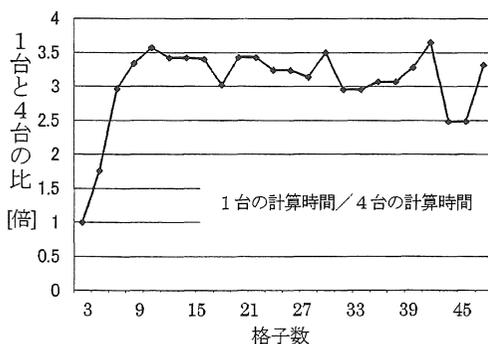


図10. 1台と4台の計算時間の比較

5. まとめ

本稿では、拡散方式を用いた冗長系逆運動学マップ計算の並列分散アーキテクチャについて、分割化と並列化の両面から検討をし、これを実現するハードウェアアーキテクチャについて設計と製作を行った。その結果、提案方式の高速な計算への有効性を示し、計算時間の短縮を実現した。

参考文献

- 1) 特集：計測と制御, vol. 29, No. 10, 1990
- 2) 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, 2-6, pp83-88, 1984
- 3) 花房, 吉川, 中村,：“関節形ロボットのアームの冗長性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
- 4) Neville Hogan：“An Approach Impedance to Manipulation, Part I, II, III”, ASME Journal of Dynamic Systems, Measurement, and Control, 107, pp.1-24, 1985
- 5) 辻 敏夫 アハマド・ヤジディ：“冗長マニピュレータのインピーダンス制御；冗長性を利用した関節インピーダンスの調節法”, 日本ロボット学会誌, 12, 4, 609-615, 1994
- 6) 川人光男：脳の計算理論, 産業図書
- 7) 大山英昭, 館 暉：“神経回路より出力誤差のフィードバックを行う逆運動学問題の解法”, 日本ロボット学会誌, 13-1, pp89-99, 1995
- 8) 田中 健一, 清水 優子, 土屋 和雄：“神経回路網による冗長マニピュレータの逆キネマティクス問題の解法”, 計測自動制御学会論文集 Vol.27, No.7, pp815-821, 1991
- 9) Zhiwei Luo, Masami Ito：“Diffusion-based learning theory for organizing visuo-motor coordination”, Bio. Cybernetics, 79, 279-289, 1998

(受理 平成11年3月20日)