

Second SICE Symposium on Computational Intelligence

September 28, 2012, Okayama

第2回コンピューターショナル・インテリジェンス研究会

「CIにおける高次元化、高次元表現」

最近のコンピューターショナル・インテリジェンス (CI) に関する研究・技術の発展には著しいものがあります。このような状況を鑑み、計測自動制御学会システム・情報部門では、新たな研究成果の発表、研究交流の場として「コンピューターショナル・インテリジェンス研究会」を立ち上げました。第2回となる今回は、テーマを「CIにおける高次元化、高次元表現」として、広くこの分野の学生・研究者・実務者の交流をはかりたいと存じます。

この研究会の議論が、この分野の新たなパラダイムを切り拓くことにつながることを期待いたしております。

趣旨：

第1回の研究会ではテーマを「クリフォードニューロコンピューティングを中心として」として開催しました。これは最近活発に研究されている、実数で表現されたニューロコンピューティングを、複素数、四元数などを一般化したクリフォード代数を用いて表現した高次元表現のニューロコンピューティングを取り上げたものです。高次元化することにより表現能力を柔軟かつ豊かにでき、高度な計算知能を実現できるものとして活発に研究されています。第2回の今回はこのテーマを継続するとともに、議論を活発にするため、次のように「高次元化、高次元表現」を切り口とします。

CIにおいては、クリフォード代数による高次元化だけでなく、様々な高次元化、高次元表現があります。たとえば、ニューラルネットワークの結合の高次元化、サポートベクターマシーンにおける特徴空間の高次元化があります。GAやPSOなど多点探索型の最適化は探索点の高次元化と捉えることができ、多点探索型の強化学習の研究も始まっています。また量子表現を用いた強化学習、PSOなどの研究もあり、これらは量子化による高次元表現です。カオスを用いた知能化の研究もあり、これはアトラクターの高次元化と捉えることができます。本研究会では、以上の例をはじめ、CIにおける様々な高次元化、高次元表現を広くとり上げ、その能力、可能性を、理論から応用まで幅広く議論します。

企画・実行委員：

荒井 幸代 (千葉大学)、黒江 康明 (京都工芸繊維大学)、新田 徹 (産業技術総合研究所)、松野 隆幸 (岡山大学)、見浪 護 (岡山大学)、森 禎弘 (京都工芸繊維大学)、矢納 陽 (岡山大学)、渡辺 桂吾 (岡山大学)

目 次

9月28日(金)

[挨拶] 8:50-9:00 ニューラルネットワーク部会主査 見浪 護

[高次元ニューラルネット1] 9:00-10:20 司会：橋 完太 (工学院大学)

- [1] バイアス項をもったカオス複素連想記憶
○鈴木 陽三、小林 正樹 (山梨大学) (1)
- [2] 複素連想記憶における局所逐次学習による記銘パターンの安定性評価
○山本 大貴、磯川 悌次郎、西村 治彦、上浦 尚武、松井 伸之 (兵庫県立大学) (7)
- [3] BFGS法と可約性写像を用いた複素多層パーセプトロン探索法
○鈴木 真矢、中野 良平 (中部大学) (13)
- [4] 階層構造に基づいた危点を持たないニューラルネットワークの構成
○新田 徹 (産業技術総合研究所) (19)

[ニューラルネットと機械学習] 10:25-12:05 司会：村田 純一 (九州大学)

- [5] ニューラルネットワークとモジュラー型ファジィ推論モデルの融合
○関 宏理 (関西学院大学) (23)
- [6] An Analysis on Ensembles of Neural Networks: a Motivation behind Low Dimensional Networks
○Danilo Vasconcellos Vargas、高野 浩貴、村田 純一 (九州大学) (31)
- [7] ニューロ様断熱的量子計算のための量子ビットデバイスの検討
○金城 光永、比嘉 孝昌、諸見里 貴徳、島袋 勝彦 (琉球大学) (37)
- [8] ニューラルネットワークによる離散事象システムの実現と同定法 – 高次結合表現とスパース実現 –
○黒江 康明、森 禎弘 (京都工芸繊維大学) (41)
- [9] 機械学習を用いたユーザ意図モデリング
○金田 祐也、趙 強福、渡會 恭平 (会津大学) (49)

[運動 – 特徴抽出と制御 –] 12:50-14:10 司会：前田 裕 (関西大学)

- [10] 人体運動データの特徴抽出についての一考察
○橋 完太 (工学院大学) (53)
- [11] Visual Lifting Stabilization に基づく二足歩行の動的形状変更能力
○小林 洋祐、見浪 護、矢納 陽、前場 友秀 (岡山大学) (57)
- [12] Dual Nature of Constraint Motion and Lyapunov-stable Position/Force Control Method
○于 福佳、見浪 護、矢納 陽 (岡山大学) (65)
- [13] 肘付拘束冗長性を利用したハンド軌道追従と拘束運動の非干渉化制御 – 消費エネルギーを基準にした肘付き位置と抗力の最適化
○糸島 道之、尾崎 壮紘、前場 友秀、見浪 護、矢納 陽 (岡山大学) (73)

[高次元ニューラルネット2] 14:15-15:35 司会：新田 徹 (産業技術総合研究所)

- [14] 複素ニューラルネットワークを用いた広帯域ビームフォーミングによる音響イメージング
○寺林 光太郎、廣瀬 明 (東京大学) (81)
- [15] 複素ニューラルネットワークのハードウェア化
○前田 裕、安藤 充弥、伊藤 秀隆 (関西大学) (87)
- [16] 高次元ネットワークインバージョンによる逆問題解法
○井浦 翼、小川 毅彦 (拓殖大学) (91)
- [17] Fourier-Mellin Transform with Quaternions
○Eckhard Hitzer (University of Fukui) (95)

[最適化] 15:40-17:40 司会：小野 功（東京工業大学）

[18] 複素信号処理と多様体上の最適化

○西森 康則（産業技術総合研究所）（103）

[19] 高次元関数最適化への DX-NES の適用

○福島 信純、永田 裕一、小野 功（東京工業大学）（107）

[20] 粒子群最適化の高次元関数最適化への適用に関する考察

安田 大悟、○畠中 利治、内種 岳詞（大阪大学）（115）

[21] 量子粒子群最適化法の基本性能評価

○田附 浩一郎、村本 憲幸、松井 伸之、磯川 悌次郎（兵庫県立大学）（121）

[22] ACO を用いたキーボード配置の適応的最適化

○石谷 彰斗、榊原 一紀、西川 郁子（立命館大学）（125）

[23] 任意軌道追従のための画像に基づくファジィ制御器の GA による最適化

○加藤 達也、渡辺 桂吾、前山 祥一（岡山大学）（131）

18:00-20:00 懇親会 岡山大学 ピーチユニオン レストラン

バイアス項をもったカオス複素連想記憶

○鈴木陽三 小林正樹 (山梨大学)

Chaotic Complex-valued Associative Memory with Bias Terms

*Yoza Suzuki Masaki Kobayashi (University of Yamanashi)

Abstract— Complex-valued Associative Memory(CAM) can store multi-level patterns. Chaotic Complex-valued Associative Memory(CCAM) can recall all stored patterns. The CAM stores not only the training patterns but also rotated patterns. So, CCAM recall all training patterns and rotated patterns in the recall process. We introduce strong bias terms to avoid recalling the rotated patterns. By computer simulations, we can see that strong bias terms can avoid recalling the rotated patterns unlike simple bias terms.

Key Words: complex-valued associative memory, chaos neuron, rotated patterns, bias terms

1 まえがき

ニューラルネットワークは、人の脳内神経における情報処理を擬似神経回路を用いて模倣する試みである。このニューラルネットワークによる例として連想記憶がある。これまでにホップフィールド型連想記憶 (Hopfield Associative Memory:HAM) ^{1), 2)} を代表として、多くのモデルが提案された。そのHAMを拡張したモデルの一つに複素連想記憶 (Complex-valued Associative Memory:CAM) ³⁾ がある。HAMは2値情報の処理のみしか行えないが、CAMは、状態が複素数で表された複素ニューロンで構成されるため、多値情報の処理が行える。

連想記憶における問題点の一つとして、偽記憶の問題がある。連想記憶が偽記憶を学習していると、雑音の存在するパターンを入力した際に、その出力パターンは偽記憶に引き寄せられ、正しく想起が行えない可能性がある。HAMは与えられた学習パターンのみでなく、その反転したパターンも記憶することが知られている。しかしCAMの場合は更に深刻であり、学習パターンのみでなく回転パターンも記憶してしまう。複素ニューロンが K の状態数を持つとき、一つの学習パターンに対して $K-1$ 個の回転パターンが存在する。また他の偽記憶としては重畳パターンがあげられる。HAMの重畳パターンは、学習パターンと反転パターンの組み合わせから生じるが、CAMの場合は、学習パターンと回転パターンの組み合わせによって生じる。CAMでは回転パターンの数が複素ニューロンの状態数に比例するため、その数は膨大な数になる。

本論文では中田、長名の提案したカオス複素連想記憶 (Chaotic Complex-valued Associative Memory:CCAM)⁴⁾ に注目する。CCAMはカオスニューロンモデルを複素数値化することで、多値パターンの動的想起を実現している。しかし、動的想起において偽記憶の問題は解決されておらず、回転パターンや重畳パターンが想起されている。我々はCCAMにおける偽記憶の問題解決手法として、バイアス項の導入を提案する。連想記憶への単純なバイアス項の導入は、Lee⁵⁾ と Muezzinoglu⁶⁾ らによって提案されていたが、CCAMにおいて単純なバイアス項では偽記憶に引きつけられてしまうため、より大きなバイアス項を与える必要がある。我々は、バイアス項の大きさによるCCAMの想起の変化を、計算機シ

ミュレーションによって確認した。

2 複素連想記憶

まず複素ニューロンについて定義する。 $K(> 2)$ のとき、複素平面上の原点を中心とする単位円は K 等分され、その K 等分点は $s_k(k = 1, 2, \dots, K-1)$ と定義する。このとき s_k は

$$\theta_K = \frac{\pi}{K}, \quad (1)$$

$$s_k = \exp(2k\theta_K\sqrt{-1}). \quad (2)$$

である。また複素ニューロンの状態の集合は $\{s_k\}$ である。ニューロンの入力を I とすると、出力は関数 $f(\cdot)$ によって求められる。関数 $f(\cdot)$ は

$$f(I) = \frac{\eta I}{\eta - 1.0 + |I|} \quad (3)$$

と定義される。ここで η は $\eta > 1$ を満たす定数である。

次にCAMについて説明する。ニューロン i からニューロン j への結合荷重を複素数 w_{ji} 、ニューロン i の状態を z_i とすると、ニューロン j への入力 S_j は、

$$S_j = \sum_{i \neq j} w_{ji} z_i. \quad (4)$$

と定義できる。このとき結合荷重は $w_{ji} = \bar{w}_{ij}$ を満たす必要がある。

ニューロンの数を N のとき、ニューロンの状態ベクトルは $z = (z_1, z_2, \dots, z_N)$ となり、 z が安定しているとするとき、このときすべての j で次の式が成り立つ。

$$z_j = f(S_j). \quad (5)$$

次に z の回転パターンについて考える。回転パターンは s_k を用いて、 $s_k z = (s_k z_1, s_k z_2, \dots, s_k z_N)$ と表せる。この回転パターンを式 (5) に与えると次の等式が得られる。

$$f\left(\sum_{i \neq j} w_{ji} s_k z_i\right) = s_k f\left(\sum_{i \neq j} w_{ji} z_i\right) = s_k z_j. \quad (6)$$

従って、回転パターンも安定することがわかる。

最後に学習則について説明する。学習パターン p のニューロンの状態ベクトルを、 $z_p = (z_{p1}, z_{p2}, \dots, z_{pN})$ とする。この時のニューロン i からニューロン j への結合荷重は、

$$w_{ji} = \sum_p z_{pj} \bar{z}_{pi}. \quad (7)$$

となり、これは結合荷重に対する要求を満たす。また \bar{z} は共役複素数を表す。

3 カオスニューラルネットワーク

カオスニューロンモデルは、従来のニューロンモデルにカオスを導入したものである。カオスニューロンモデルを相互に結合したネットワークはカオスニューラルネットワークと呼ばれている。 N 個のニューロン間の相互結合を持つネットワークの n 番目のカオスニューロンのダイナミクスは、

$$z_n(t+1) = f\left(\sum_{j=1}^N w_{nj} \sum_{d=0}^t k_m^d z_j(t-d) - \alpha \sum_{d=0}^t k_r^d z_n(t-d)\right) \quad (8)$$

と定義される。ここで、 $z_n(t)$ は時刻 t における n 番目のニューロンの出力、 α は不応性に関するスケールリングファクタ ($\alpha > 0$)、 k_m 、 k_r は時間減衰定数 ($0 \leq k_m, k_r < 1$)、である。カオス的な動作をさせるためにはパラメータ α や k_m 、 k_r を適切に決定する必要がある。

式 (8) の第 2 項は安定状態が維持されるとそれに伴って、現在の状態とは逆方向に移行しようとする働きが大きくなる。この働きにより、安定状態から抜け出し学習したパターンを動的に想起することが出来るようになる。

4 カオス複素連想記憶

カオス複素連想記憶は、複素ニューロンモデルにカオスを導入したカオス複素ニューロンモデルが相互結合することで構成されたホップフィールドネットワークによって実現する。学習には式 (7) のヘブ則を用いる。またカオス複素連想記憶における n 番目のニューロンのダイナミクスは式 (8) から、

$$z_n(t+1) = f\left(\sum_{j=1}^N w_{nj} \sum_{d=0}^t k_m^d z_j(t-d) - \alpha \sum_{d=0}^t k_r^d z_n(t-d)\right) \quad (9)$$

と表される。ここで、 $z_n(t)$ は時刻 t における n 番目のニューロンの出力、 w_{nj} は j 番目のニューロンから n 番目のニューロンへの結合荷重である。またこれらの値は複素数で表される。また k_m 、 k_r は時間減衰定数、 α は不応性に関するスケールリングファクタでこれらは実数値をとる。

5 バイアス項

我々は CCAM における回転パターンの想起の改善に対して、バイアス項の導入を提案する。バイアス項は常に 1 を出力するニューロンからの結合荷重であると考えられる。このときのバイアス項からニューロン j への結合荷重を考える。従って、ニューロン j への入力 S_j は、

$$S_j = \sum_{i \neq j} w_{ji} z_i + w_{j0} \quad (10)$$

と定義される。またヘブ則から、

$$w_{j0} = \sum_p z_{pj} \quad (11)$$

が得られる。しかし、このバイアス項では単純に一つのニューロンを加えているだけなので、効果は少ない。そのため、バイアス項の大きさのパラメータ C とし、これによってバイアス項の大きさを変化させる。パラメータを加えた結合荷重の式は、

$$w_{j0} = C \sum_p z_{pj} \quad (12)$$

と表される。このとき $C = 0$ はバイアス項がない場合、 $C = 1$ は単純なバイアス項を加えた場合と等しくなる。

6 計算機シミュレーション

バイアス項の効果を確認するために計算機シミュレーションを行った。CCAM の計算機シミュレーションにおける、各パラメータの値は次のようになる。

1. $K = 4$
2. $N = 400$
3. $\alpha = 35.0$
4. $k_m = 0.1$
5. $k_r = 0.92$
6. $\eta = 1.1$
7. $t < 200$

学習したパターンは Fig.1 に示す。CCAM の単純な出力は、複素平面上の $|z_n| < \eta$ を満たす点をとるが、想起結果の出力は各ニューロンの状態を次のように量子化を行なっている。

$$g(z) = \begin{cases} s_0 & 0 \leq z < \theta_K \\ s_1 & \theta_K \leq z < 3\theta_K \\ s_2 & 3\theta_K \leq z < 5\theta_K \\ \vdots & \\ s_{K-1} & (2K-3)\theta_K \leq z < (2K-1)\theta_K \\ s_K & (2K-1)\theta_K \leq z < 2\pi \end{cases} \quad (13)$$

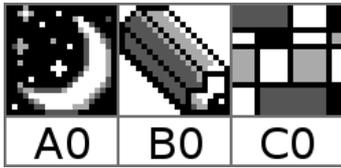


Fig. 1: Training patterns

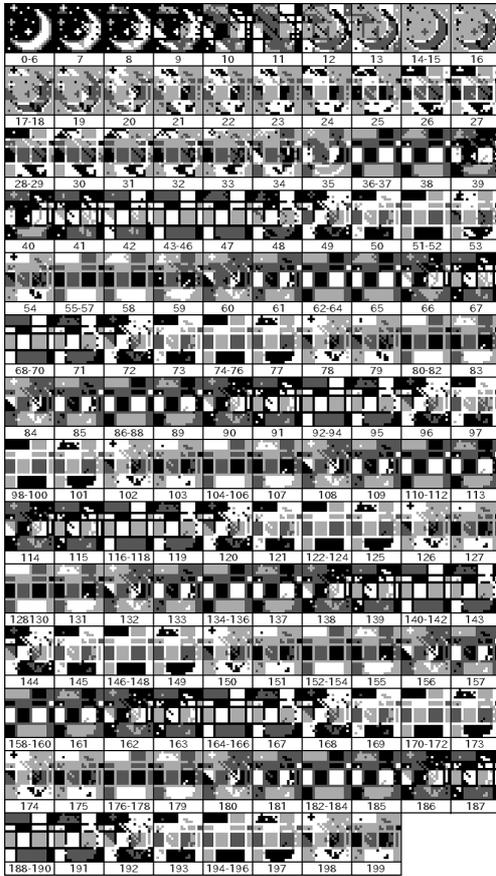


Fig. 2: The simulation result of $C = 0$

想起結果は Fig.2 のようになる。

量子化によってニューロンは K 個の状態を出力として取る。従って今回の場合では、 $K = 4$ であるため回転パターンは 3 パターン存在し、Fig. 1 の学習パターン A0 を例に取ると Fig. 3 のような回転パターンが各学習パターンで存在する。A1 は A0 が $\pi/2$ 回転したパターンで、A2 は A0 が π 回転したパターン、A3 は A0 が $3\pi/2$ 回転したパターンとなる。想起の結果としては Fig. 2 のようになる。Fig. 2 は $C = 0$ のときの結果であり、パターン下の数字は想起された時間を表す。

計算機シミュレーションでは、バイアス項のパラメータである C を 0, 1, 100, 200 と順に大きくしていきそれぞれの想起結果を比較する。

$C = 0$ の場合、想起結果は次の Table 1 のようになった。表における othres は重畳パターンである。 $C = 0$ のときいくつかの学習パターンが想起されているが、想起された学習パターンの回転パターンの方が多く想起されている。また重畳パターンは時刻 t の早い時のみみられた。

$C = 1$ の場合、想起結果は Table 2 のようになった。この結果は $C = 0$ のときとあまり変わらない結果となった。このことから単純に大きき 1 のバイアス項を

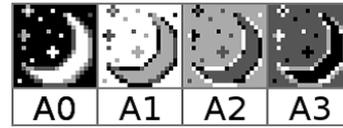


Fig. 3: Rotated patterns of A0

与えても、あまり回転パターンの想起削減に効果がないことがわかる。

次に $C = 100$ の場合、想起結果は Table 3 のようになった。バイアス項を大きくしてみると、A0 の回転パターンは想起されなくなった。また他の学習パターンの回転パターンも安定回数が減少した。しかし、バイアス項が小さい場合と比較して、重畳パターンの安定回数が増加してしまった。

$C = 200$ の場合、想起結果は Table 4 のようになった。更にバイアス項を大きくすると、すべての学習パターンのみが想起され、回転パターンは想起されなくなった。このことからバイアス項が回転パターンの安定回数を抑える効果があることがわかる。しかし、重畳パターンは今までの結果で最も多く安定してしまっている。

ここで、更に想起時間を $t < 200$ から $t < 1000$ に長くしてシミュレーションを行い、各条件下におけるパターンごとの安定回数を Table 5 にまとめた。

想起時間を長くした場合でも、強いバイアス項のある $C = 200$ では回転パターンは全く想起されなかった。また $C = 100$ の場合でも $C = 0, 1$ の場合に比べて回転パターンの安定回数が減少していることから、回転パターンの安定を減少させる効果がバイアス項にあると考えられる。

7 まとめ

中田、長名の提案した CCAM は学習パターンだけでなく、回転パターンや重畳パターンを想起してしまっていた。しかし、今回我々の提案した強いバイアス項を導入することで、CCAM は回転パターンを想起しなくなった。また重畳パターンは、バイアス項を強くするにつれ安定回数が増加してしまった。

シミュレーション結果にみられた問題である重畳パターンの安定回数の増加については、スケールングファクタや時間減衰定数を適切に設定することで、減少させることが可能であると考えられる。しかし、これらのパラメータは扱いが難しく、 $C = 0$ の実験結果のようにすべての学習パターンを想起しないパラメータも存在する。従って、適切なパラメータを設定するためには、多くのシミュレーションや実行者の経験による推測が必要となる。この問題の解決には、パラメータの容易な設定方法の確立や、パラメータに依存しない重畳パターンの安定回数を減少させる手法を提案する必要がある。

参考文献

- 1) Hopfield, J., J.: Neural networks and physical systems with emergent collective computational abilities. In: Proceedings of the National Academy of Sciences of the United States of America, vol.79, no.8, 2554/2558, (1982)
- 2) Hopfield, J., J.: Neurons with graded response have collective computational properties like those of two-state neurons. National Academy of Sciences of the

Table 1: The simulation result of $C = 0$

patterns	periods						
A0	0 - 6						
B0							
C0	43 - 46	68 - 70	92 - 94	116 - 118	140 - 142	164 - 166	188 - 190
A1							
B1							
C1	51 - 52	74 - 76	98 - 100	122 - 124	146 - 148	170 - 172	194 - 196
A2	14 - 15						
B2							
C2	55 - 57	80 - 82	104 - 106	128 - 130	152 - 154	176 - 178	
A3							
B3							
C3	36 - 37	62 - 64	86 - 88	110 - 112	134 - 136	158 - 160	182 - 184
others	17 - 18	28 - 29					

Table 2: The simulation result of $C = 1$

patterns	periods							
A0	0 - 6							
B0								
C0	38 - 42	64 - 66	88 - 90	112 - 114	136 - 138	160 - 162	184 - 186	
A1								
B1								
C1	47 - 48	70 - 72	94 - 96	118 - 120	142 - 144	166 - 168	190 - 192	
A2	14 - 16							
B2								
C2	27 - 30	52 - 53	76 - 78	100 - 102	124 - 126	148 - 150	172 - 174	196 - 198
A3								
B3								
C3	33 - 34	58 - 60	82 - 84	106 - 108	130 - 132	154 - 156	178 - 180	
others	17 - 18	25 - 26						

United States of America, vol.81, no.10, 3088/3092, (1984)

- 3) Jankowski, S., Lozowski, A., Zurada, J., M.: Complex-valued multistate neural sssociative memory. IEEE Transaction on Neural Networks, vol.7, no.6, 1491/1496 (1996)
- 4) Nakada, M., Osana, Y.: Chaotic complex-valued associative memory. Proc. International Symposium on Nonlinear Theory and its Applications, 493/496 (2007)
- 5) Lee, D., L.: Improvements of complex-valued hopfield associative memory by using generalized projection rules. IEEE Transaction on Neural Networks, vol.17, no.5, 1341/1347 (2006)
- 6) Muezzinoglu, M., K., Guzelis, C., Zurada, J., M.: A new design method for the complex-valued multi-state hopfield associative memory. IEEE Transaction on Neural Networks, vol.14, no.4, 891/899 (2003)

Table 3: The simulation result of $C = 100$

patterns	periods
A0	0 - 1
B0	152 - 154 177 - 181
C0	11 - 15 36 - 37 64 - 65 87 - 88
A1	
B1	147 - 148 195 - 197
C1	69 - 71
A2	
B2	143 - 144 190 - 192
C2	25 - 27 77 - 78
A3	
B3	
C3	30 - 31 81 - 82
others	4 - 7 58 - 59 95 - 96 108 - 109 126 - 127 128 - 130 134 - 135 157 - 158 162 - 163

Table 4: The simulation result of $C = 200$

patterns	periods
A0	0 - 1 20 - 22 44 - 47 81 - 83 164 - 167
B0	34 - 36 68 - 71
C0	54 - 58 91 - 93 114 - 118 142 - 145 175 - 178
A1	
B1	
C1	
A2	
B2	
C2	
A3	
B3	
C3	
others	2 - 16 26 - 27 50 - 51 78 - 79 94 - 95 120 - 121 124 - 125 134 - 135 147 - 148 157 - 158 171 - 172 179 - 180 187 - 188

Table 5: The simulation results of $t \downarrow 1000$

	patterns												
C	A0	B0	C0	A1	B1	C1	A2	B2	C2	A3	B3	C3	others
0	1	0	40	0	0	40	1	0	40	0	0	41	2
1	1	0	40	0	0	40	1	0	41	0	0	41	2
100	1	36	4	0	18	1	0	18	2	0	0	2	76
200	14	12	31	0	0	0	0	0	0	0	0	0	81

複素連想記憶における局所逐次学習による 記銘パターンの安定性評価

○山本大貴 礪川悌次郎 西村治彦 上浦尚武 松井伸之 (兵庫県立大学)

Stability Evaluation on Complex-Valued Associative Memory by Local Iterative Learning Scheme

* Hiroki Yamamoto, Teijiro Isokawa, Haruhiko Nishimura,

Naotake Kamiura, and Nobuyuki Matsui

(University of Hyogo)

Abstract—Stability of embedded patterns on associative memory is investigated in this paper. The associative memory is composed of complex-valued Hopfield neural network, in which the state of a neuron is encoded by a phase value on a unit circle of complex plane. Local iterative learning scheme and Projection rule are used for embedding the patterns onto the network. Retrieval performances for embedded patterns are evaluated through storing randomly generated patterns and gray-scaled images and randomly generated patterns with various resolution of neuron state.

Key Words: Hopfield Neural Network, Complex-valued Neural Network, Iterative Learning, Projection Rule

1 まえがき

複素ニューラルネットワーク(Complex-Valued Neural Network, 以下複素NN)とは, 入力, 出力, および結合荷重などのニューロンに関するパラメータが複素数により表現されるニューラルネットワークである. これまでに様々な複素NNが提案され, 基礎理論から応用に至るまで広く研究が展開されている^{1, 2)}. 複素NNにおける利点の一つは, 単一のニューロンにより2次元情報を取り扱うことができるということであり, これにより振幅・位相情報や2次元座標情報などを自然に表現することが可能である. このような特徴を持つ複素NNは, レーダー波を用いた地雷探知³⁾ や, 通信チャンネルにおける等化問題⁴⁾ などの工学的問題にも適用されている.

複素多値ニューラルネットワーク(Complex-Valued Multistate Neural Network, 以下多値NN)は複素NNの一種であり^{5, 6, 7, 8)}, ニューロンの状態が複素平面における単位円上の点の位置情報により表現される. すなわち, ニューロン状態は位相として表現することが可能であり, 単位円上に配置する点の数により任意の離散的な情報を表現することが可能である. これまでに, 多値NNについてはいくつかのモデルが提案され, パターンの記銘手法ならびにその安定性が理論的および実験的に解析が行われている^{9, 10, 11, 12)}.

Hebb学習則は, パターンを記銘するための基本的かつ簡単な学習則であるが, 記銘すべきパターンベクト

ルは互いに直交している必要がある. しかしながら, 全ての記銘パターンが互いに直交していることは現実的には考え難く, 実用的な学習則ではない. 射影学習則は, 擬逆行列を用いて非直交パターンを直交パターンに射影することにより, Hebb学習則における制限を克服した学習則である. この学習則では, 記銘パターンの擬逆行列を計算する必要があり, パターンの大きさに応じて, その計算量は増大する.

そこで, 複素数並びに四元数の多値NNにおける局所逐次学習法¹³⁾ が提案されている¹²⁾. これは, 射影学習則の一実装であり, この学習則では, 擬逆行列を計算する代わりに, 誤差逆伝搬法と同様に記銘パターンを繰り返し提示することによりネットワークの結合荷重を更新してゆく方法である. またこの学習則では, 各パターンの提示回数を操作することにより, 記銘するパターンにおける引き込み領域の大きさを制御することも可能である. 提案学習法により記銘パターンが引き込み領域となるように結合荷重の更新が確かに行われることが理論的に示されているが¹²⁾, 実際の画像パターンなどを用いた場合における記銘性能については未だ検証されていない. 本稿では, 多値NNにおいて局所逐次学習法を用いてパターンを記銘することにより, 記銘されたパターンの安定性を定量的に評価する. また, 使用する記銘パターンとして濃淡実画像ならびに様々な階調を持つランダムパターンを使用し, 性能比較として射影学習則を用いた記銘・想起実験も行う.

2 準備

2.1 複素多値ニューラルネットワーク

まず、文献^{6, 8)}において用いられている多値NNモデルについて説明する. このモデルは、入出力値、閾値、結合荷重全てが複素数で表現される複素数ニューロンから構成され、ニューロンの値は複素平面の単位円上の点で表される. つまり、このモデルにおけるニューロンの値は位相のみで表現される. Fig. 1では、量子化単位 K の値が6の場合、すなわち6状態を取り得るニューロンの状態値を示す.

次に、 N 個の複素ニューロンから構成される相互結合型ネットワークについて考える. 時刻 t におけるニューロン p の内部ポテンシャル $h_p(t)$ は次式により定義される.

$$h_p(t) = \sum_q^N w_{pq} u_q(t) \quad (1)$$

ここで、 w_{pq} はニューロン q から p への結合荷重である. また、活性化関数 $c\text{sign}$ を用いることにより、時刻 $(t+1)$ におけるニューロン p の更新状態は次式で定義する.

$$u_p(t+1) = c\text{sign}(h_p(t) \cdot z^{1/2}) \quad (2)$$

$$c\text{sign}(u) = \begin{cases} z^0 & 0 \leq \arg(u) < \varphi_0 \\ z^1 & \varphi_0 \leq \arg(u) < 2\varphi_0 \\ \vdots & \vdots \\ z^{K-1} & (K-1)\varphi_0 \leq \arg(u) < K\varphi_0 \end{cases} \quad (3)$$

ここで φ_0 は量子化単位であり、 $\varphi_0 = 2\pi/K$ を満たす. この活性化関数により、ニューロンの状態は K 分割された量子化状態のうちの一つを取ることになる.

このモデルから構成される相互結合型のネットワークの安定性は、 $|\Delta\varphi| < \varphi_0/2$ という条件下においてネットワークのエネルギー E が単調減少することにより証明されている⁸⁾. ここで、 $\Delta\varphi$ は状態更新を行ったニューロンの位相変化であり、 $\Delta\varphi = u(t+1) - u(t)$ として定義される. またネットワークのエネルギー E は次式で定義される.

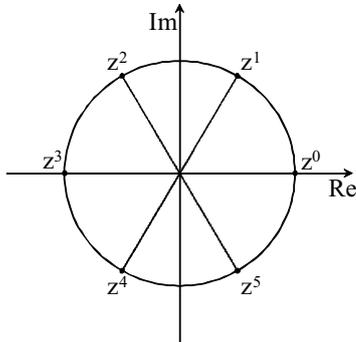


Fig. 1: An example of output points in the complex-valued multistate neuron model

$$E(t) = -\frac{1}{2} \sum_p \sum_q w_{pq} u_p^* u_q \quad (4)$$

ここで、結合荷重 $W = \{w_{pq}\}$ はエルミート行列 ($w_{pq} = w_{qp}^*$) である.

2.2 パターン記録における安定条件

多値 NN における学習則を説明するために、まずネットワークに記録されるパターンが安定に記録されるための条件について説明する. μ 番目の記録パターンベクトルを $\xi^\mu = \{\xi_p^\mu\}$ ($p=1, \dots, N; \mu=1, \dots, M$) とする. ここで、 ξ_p^μ は 0 から $(K-1)$ の間の整数値である ($\xi_p^\mu \in \{1, \dots, K-1\}$). これを用いて記録パターンは、複素平面における単位円上の点として次式で表される.

$$\varepsilon_p^\mu = z^{\xi_p^\mu} = e^{i\xi_p^\mu \varphi_0} \quad (5)$$

記録パターン ε_p^μ においてネットワークが安定しているとき、

$$u_p(t+1) = u_p(t) = \varepsilon_p^\mu \quad (6)$$

が各ニューロン p ($p=1, \dots, N$) において成り立つ. このとき、ニューロン p の位相は次の関係を持つ.

$$\arg(\varepsilon_p^\mu) \leq \arg(u_p(t+1)) < \arg(\varepsilon_p^\mu) + \varphi_0 \quad (7)$$

(2)式により、この関係は次のようになる.

$$|\arg(h_p(t)) - \arg(\varepsilon_p^\mu)| < \frac{\varphi_0}{2} \quad (8)$$

また、所望の記録パターンにおいてより高い安定性を得るために、閾値パラメータ κ を導入し、次式の関係を満たすように結合荷重の設定を行う.

$$|\arg(h_p(t)) - \arg(\varepsilon_p^\mu)| < \kappa < \frac{\varphi_0}{2} \quad (9)$$

この κ を導入することにより、各パターンにおける引き込み領域を制御することができる.

2.3 Hebb 学習則

後述する射影学習則の基礎となる Hebb 学習則について説明する. 記名パターン $\{\varepsilon_\mu\}$ が与えられた時、多値 NN におけるニューロン p と q の間の結合荷重を次式により決定する.

$$w_{pq} = \frac{1}{N} \sum_{\mu=1}^M \varepsilon_p^\mu \tilde{\varepsilon}_q^\mu \quad (10)$$

ここで、 ε_p^μ は記名パターン μ におけるニューロン p の状態であり、 M は記名パターン数である. この式で計算される結合荷重は、 $w_{pq} = w_{pq}^*$ および $w_{pp} \geq 0$ の条件を満たし、さらに各記録パターンが安定点となる.

2.4 射影学習則

前節の Hebb 学習則においては、各記録パターンベクトルが互いに直交関係となる必要がある。射影学習則^{10, 14, 15)}では、非直交パターンを直交パターンに射影し、Hebb 学習則により記録を行うことにより非直交パターンの記録を行う。多値 NN における射影学習則は、以下に示す行列 $Q_{\mu\nu}$ を導入することにより射影を行う。

$$Q_{\mu\nu} = \frac{1}{N} \sum_p \tilde{\varepsilon}_p^\mu \varepsilon_p^\nu \quad (11)$$

これにより、ネットワークの結合荷重 \tilde{w}_{pq} は、次式で計算される。

$$\tilde{w}_{pq} = \frac{1}{N} \sum_{\nu, \mu} \varepsilon_p^\mu \tilde{\varepsilon}_q^\nu (Q^{-1})_{\mu\nu} \quad (12)$$

この学習則により形成される結合荷重を持つネットワークにおいても、Hebb 学習則と同様に各記録パターンが安定点となる。記録パターン ε^σ が入力された場合におけるニューロン p の内部状態 \tilde{h}_p は、

$$\begin{aligned} \tilde{h}_p &= \sum_{q=1}^N \tilde{w}_{pq} \varepsilon_q^\sigma \\ &= \frac{1}{N} \sum_p \varepsilon_p^\mu (Q^{-1})_{\mu\nu} \sum_q \varepsilon_q^{\nu*} \varepsilon_q^\sigma \\ &= \sum_{\mu, \nu} \varepsilon_p^\mu (Q^{-1})_{\mu\nu} Q_{\nu\sigma} \\ &= \sum_{\mu, \nu} \varepsilon_p^\mu (Q^{-1} Q)_{\mu\sigma} \\ &= \sum_{\mu} \varepsilon_p^\mu \delta_{\mu\sigma} \\ &= \varepsilon_p^\sigma \end{aligned} \quad (13)$$

と表される。これにより、記録パターンが入力された場合、ニューロン状態が更新されても同じ記録パターン状態になることから、記録パターンはネットワークにおける安定点であることがわかる。

2.5 局所逐次学習法

局所逐次学習法は、前節で説明した射影学習則の一実装である。この学習則では誤差逆伝搬法などの学習則と同様に、記録パターンを繰り返し提示することにより結合荷重を徐々に更新し形成してゆく。局所逐次学習法では、射影学習則において必要とされた擬逆行列 Q^{-1} を計算する必要がなく、また各記録パターンに対する引き込み領域の大きさを制御できるという特徴がある¹³⁾。

多値 NN における局所逐次学習法においては、ネットワークの結合荷重は以下の式により更新される。

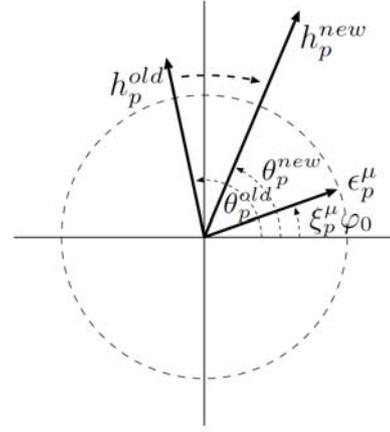


Fig. 2: Behavior of vectors in a complex plane due to the application of iterative learning

$$w_{pq}^{new} = w_{pq}^{old} + \delta w_{pq} \quad (14)$$

$$\delta w_{pq} = \frac{1}{N} \varepsilon_p^\mu \varepsilon_q^{\mu*} \quad (15)$$

結合荷重を 1 回更新した後のニューロンの内部状態は次式で表される。

$$h_p^{new} = \sum_q w_{pq}^{new} u_q = h_p^{old} + \sum_q \frac{1}{N} \varepsilon_p^\mu \varepsilon_q^{\mu*} u_q \quad (16)$$

結合荷重の更新による記録状態 ε_q^μ と内部状態の関係は、 $u_q = \varepsilon_q^\mu$ とすることにより次式のように確認することができる。

$$h_p^{new} = h_p^{old} + \sum_q \frac{1}{N} \varepsilon_p^\mu \varepsilon_q^{\mu*} \varepsilon_q^\mu = h_p^{old} + \varepsilon_p^\mu \quad (17)$$

これらのベクトルを図示したものが Fig.2 である。これらのベクトルの関係から、結合荷重を更新するたびに h_p^{new} の向きは ε_p^μ の向きに近づいていることがわかる。すなわち、式(8)あるいは式(9)を満足するように結合荷重を更新することにより、記録パターンは安定化することになる。

3 実験結果

3.1 ノイズを含む入力からの画像想起

まず局所逐次学習法による連想記憶の動作を確認するために、ノイズを含む入力からのパターン想起能力を調査する。この実験において記録される 3 種類の画像¹⁶⁾を Fig. 3 に示す。

これらの画像はその大きさが $90 \times 90 = 8100$ 画素となるように調整され、各画素値は 5 ビット(32 階調)により表現されている。入力に用いる画像は、これらの記録パターンにノイズを重畳したものを使用する。本実験でのノイズの重畳は、各画素に対し、一定の確率(r_n)により一様分布に従うランダムな値により画素値を書き換える操作により行う。Fig. 3(a)の画像に対してノイズを重畳した例を Fig. 4 に示す。この図より r_n が増加

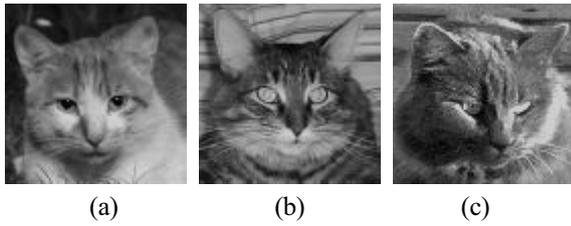


Fig. 3: Three stored images for the image retrieval experiment

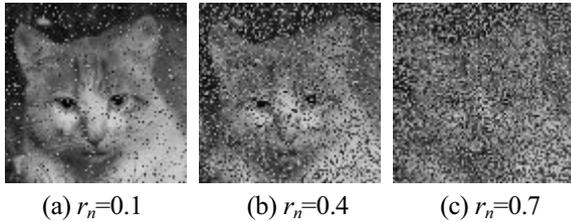


Fig. 4: Input images with various noise probabilities r_n

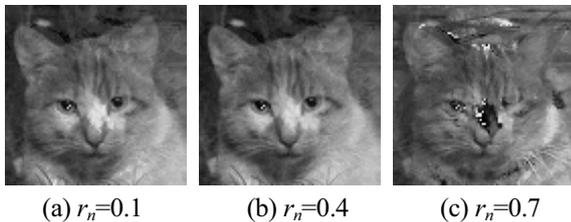


Fig. 5: Output images from the network when the images in Fig. 4 are used for input

すると、元画像が徐々に欠落してゆくことが分かる。記録を行うネットワークにおけるニューロン数は画像の大きさと同じ、すなわち $N=8100$ とし、量子化単位 $K=32$ とする。学習段階において、Fig. 3(a), 3(b), 3(c)の画像をそれぞれ1回ずつ提示し、この操作を10回繰り返すことで結合荷重を再形成する。その後、ネットワークの初期状態としてテスト画像を与え、ネットワーク状態が収束するまでニューロンの更新を行う。上記手順によるネットワークの出力結果をFig. 5に示す。Fig. 5(a), 5(b), 5(c)は、初期状態としてFig. 4(a), 4(b), 4(c)をそれぞれ与えた場合に対応している。この結果より、低い r_n においては、記録パターンは正しく想起されていることがわかる。これにより、局所逐次学習則により記録パターンを提示することによりネットワーク内には記録パターンを中心とした引き込み領域が確かに形成できていることが示された。

3.2 記録パターンの安定性

次に、様々な記録パターン数・量子化単位に対して記録パターンの安定性について検討する。この実験では、記録パターンにおける各要素は一様分布する乱数値に設定される。パターン(ネットワーク内のニューロン数)の大きさは 10×10 に設定し、量子化単位 K は $K=4, 8, 16, 32, 64$ とする。また、記録するパターン数は $M=2 \sim 50$ とする。

パターンの安定性は次のようにして検証する。まず、 K と M の値を設定し、これらの値を元にパターンを生

成し、ネットワークに記録を行う。記録においては、各記録パターンをそれぞれ10回提示し、結合荷重を更新する。記録を行った後に入力パターンを与えネットワークの状態更新を行うことにより想起を行う。入力として記録パターンのいずれかをネットワークの初期状態に設定する。想起を行った結果、ネットワークの状態が安定し初期状態と同一であれば、入力に用いたパターン(記録パターンの一つ)は正しく埋め込まれている。また、想起の結果ネットワークの状態が初期状態と異なる状態に落ち着くのであれば、パターンは正しく記録されていない。想起実験については、記録に用いた全てのパターンを入力として用い、これら全ての入力に対して正しく想起できた場合のみ、想起に成功したとみなす。

局所逐次学習則を用いて想起実験を行った結果をFig. 6に示す。この結果は、記録パターン数 M に対する想起成功率を表しており、各 M については異なる100種類の記録パターンセットを用いた場合の平均想起成功率を算出している。この結果より、同一規模のネットワークの場合においては、記録パターン数が多くなるほど記録は困難になっているということがわかる。また、正しく記録できるパターン数は、 K にも依存していることが分かる。これは、 K の値が大きくな

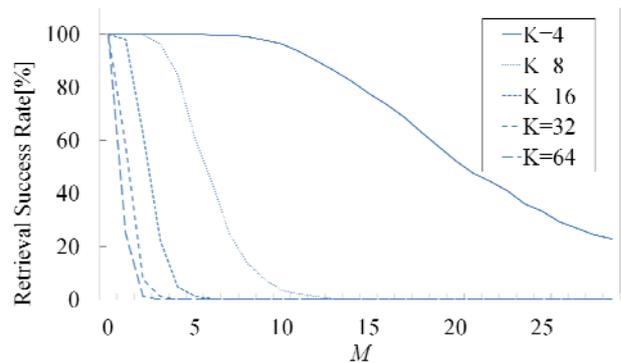


Fig. 6: M dependencies of the retrieval success rates by local iterative learning

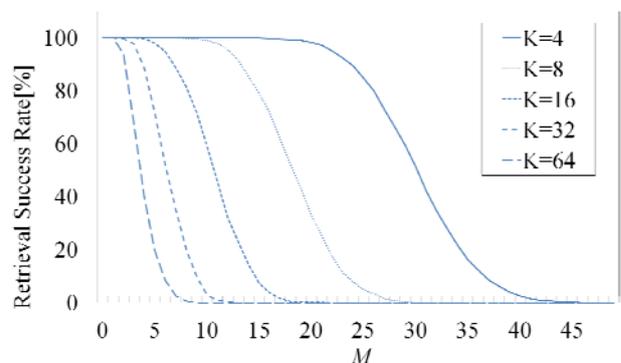


Fig. 7: M dependencies of the retrieval success rates by Projection rule

ることにより、ニューロンの持つ情報量も増加し、これは M の値の増加に相当するものとも考えることもできる。また、同様の実験を射影学習則により行った想起成功率を Fig. 7 に示す。この結果から、局所逐次学習法に比べると記録性能は高いが、こちらも K や M の値の変化に対して局所逐次学習則と同様の傾向が見られるということがわかる。

4 まとめ

本稿では、複素連想記憶における記録パターンの安定性を評価した。この複素連想記憶は複素多値ホップフィールドネットワークにより構成されており、パターンを記録させる手法として局所逐次学習法を用いている。実画像パターンならびにランダムパターンを用いたパターン記録・想起実験の結果より、局所逐次学習法を用いることにより非直交パターンの記録が確かに可能であることが示された。また、記録能力は記録パターン数 M ならびに量子化単位 K に依存することも示された。

また、局所逐次学習法では射影学習則と比較すると計算コストを大幅に削減することが可能であるが、パターン記録性能について見ると射影学習則には及ばないのが現状である。しかしながら、記録パターンを提示する過程において、パターン毎の提示回数や提示順序を操作することにより記録能力を向上できる可能性がある。また、より高次元の連想記憶、例えば四元数に基づく多値連想記憶¹⁷⁾における局所逐次学習則の性能評価も重要な課題であると考えられる。

謝辞

本研究の一部は日本学術振興会の科学研究補助金(若手研究(B) 24700227 および基盤研究(C) 23500286)の支援により行われた。

参考文献

- 1) A. Hirose, Ed.: *Complex-Valued Neural Networks: Theories and Application*, ser. Innovative Intelligence. Singapore, World Scientific Publishing, vol. 5 (2003)
- 2) T. Nitta, Ed.: *Complex-Valued Neural Networks: Utilizing High-Dimensional Parameters*, Hershey, New York: Information Science Reference (2009)
- 3) Y. Nakano and A. Hirose: Improvement of Plastic Landmine Visualization Performance by Use of Ring-CSOM and Frequency-Domain Local Correlation, *IEICE Transactions*, vol. 92-C, no. 1, 102/108 (2009).
- 4) R. Pandey: Complex-Valued Neural Networks for Blind Equalization of Time-Varying Channels, *International Journal of Signal Processing*, vol. 1, no. 1, 1/8 (2004)
- 5) A. J. Noest: Associative Memory in Sparse Neural Networks, *Europhysics Letters*, vol. 6, no. 6, 469/474 (1988)
- 6) N. N. Aizenberg and I. N. Aizenberg: CNN Based on Multi-Valued Neuron as a Model of Associative Memory for Gray-Scale Images, *Proceedings of the 2nd IEEE International Workshop on Cellular*

- Neural Networks and their Applications (CNNA-92)*, 36/41 (1992)
- 7) I. N. Aizenberg, N. N. Aizenberg, and J. Vandewalle: *Multi-Valued and Universal Binary Neurons – Theory, Learning and Applications* –, Boston/Dordrecht/London: Kluwer Academic Publishers (2000)
- 8) S. Jankowski, A. Lozowski, and J. M. Zurada: Complex-Valued Multistate Neural Associative Memory, *IEEE Transactions on Neural Networks*, vol. 7, no. 6, 1491/1496 (1996)
- 9) M. K. Müezzinoğlu, C. Güzeliş, and J. M. Zurada: A New Design Method for the Complex-Valued Multistate Hopfield Associative Memory, *IEEE Transactions on Neural Networks*, vol. 14, no. 4, 891/899 (2003)
- 10) D.-L. Lee: Improvements of complex-valued hopfield associative memory by using generalized projection rules, *IEEE Transaction on Neural Networks*, vol. 17, no. 5, 1341/1347 (2006)
- 11) M. Kobayashi: Pseudo-relaxation learning algorithm for complex valued associative memory, *International Journal of Neural Systems*, vol. 18, no. 2, 147/156 (2008)
- 12) T. Isokawa, H. Nishimura, and N. Matsui: An Iterative Learning Scheme for Multistate Complex-Valued and Quaternionic Hopfield Neural Networks, *Proceedings of International Joint Conference on Neural Networks (IJCNN2009)*, 1365/1371 (2009)
- 13) S. Diederich and M. Opper; Learning of Correlated Patterns in Spin-Glass Networks by Local Learning Rules, *Phys. Rev. Lett.*, vol. 58, 949/952 (1987)
- 14) T. Kohonen: *Self-Organization and Associative Memory*, Berlin, Heidelberg: Springer (1984)
- 15) L. Personnaz, I. Guyon, and G. Dreyfus; Collective Computational Properties of Neural Networks: New Learning Mechanisms, *Phys. Rev. A*, vol. 34, 4217/4228 (1986)
- 16) F. Flueret and D. Geman: Stationary Features and Cat Detection, *Journal of Machine Learning Research*, vol. 9, 2549/2578 (2008)
- 17) T. Isokawa, H. Nishimura, A. Saitoh, N. Kamiura, and N. Matsui: On the Scheme of Quaternionic Multistate Hopfield Neural Network, *Proceedings of Joint 4th International Conference on Soft Computing and Intelligent Systems and 9th International Symposium on advanced Intelligent Systems (SCIS & ISIS 2008)*, 809/813 (2008)

BFGS法と可約性写像を用いた複素多層パーセプトロン探索法

○鈴木 真矢 中野 良平 (中部大学)

Complex-Valued Multilayer Perceptron Search Utilizing BFGS Method and Reducibility Mapping

* S. Suzumura and R. Nakano (Chubu University)

Abstract— This paper proposes a new search method for complex-valued MLP, which utilizes the BFGS method and the reducibility mapping. The complex-valued version of the BFGS method can be formalized by using Wirtinger calculus. We compared the performance of the proposed method with those of the steepest descent and the BFGS method.

Key Words: complex-valued multilayer perceptron, steepest descent, BFGS, reducibility mapping

1 はじめに

実数の指数関数 e^x は単調増加関数であるが、複素関数 e^{x+iy} へ拡張することにより、周期的運動が表現できるようになる。また、 $y=0$ のとき e^{x+iy} は単調増加関数となるように、実数上での特性が損なわれるわけではない。複素ニューラルネットワーク (NN) を利用することにより、実 NN では獲得できなかった汎化特性が得られと考えられ、画像処理や通信の分野にて多彩な応用が展開されている^{9, 4, 2)}。

複素 NN の学習法として、バックプロパゲーション学習法 (BP 法) や最急降下法が知られる。本稿では、複素多層パーセプトロンの高速学習法として、BFGS (Broyden-Fletcher-Goldfarb-Shanno) 法と可約性写像を用いた探索法を提案する。BFGS 法の複素数への拡張は、Wirtinger 微分¹⁾ を用いて行う。なお、一般に、BP 法は学習率を一定として、1 サンプル毎に学習を進めるため、リアルタイム学習に適している。ただし、本稿では、特異点を有する活性化関数を扱うため、BP 法では収束性に難が残る。したがって、直線探索付きの最急降下法を扱う。計算機実験では、最急降下法、BFGS 法、そして提案法を比較し、提案法の有効性を示す。

2 複素多層パーセプトロンモデル

本稿が対象とする複素 3 層パーセプトロンを図 1 に示す。隠れユニットおよび出力ユニットの計算式を以

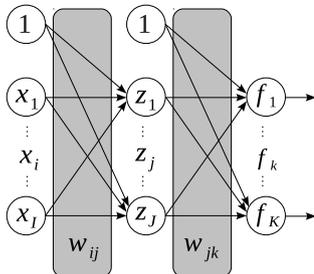


Fig. 1: 複素 3 層パーセプトロンモデル

下に示す。ただし、 $x_0^\mu = z_0^\mu = 1$ とし、右肩の μ はサンプル番号を示す。

$$f_k^\mu = \sum_{j=0}^J w_{jk} z_j^\mu, \quad z_j^\mu = g(h_j^\mu), \quad h_j^\mu = \sum_{i=0}^I w_{ij} x_i^\mu \quad (1)$$

ここで、 $g(h)$ は以下の活性化関数とする。

$$g(h) = \frac{1}{1 + \sqrt{-1} + e^{-h}} \quad (2)$$

$g(h)$ は正則関数であるが、周期的に特異点を有する。

3 Wirtinger 微分と探索法

複素最急降下法は、目的関数の勾配の計算を必要とする。勾配の計算には、Wirtinger 微分¹⁾ が利用できる。Wirtinger 微分における多変量の記法には、 c 表現、 z 表現および r 表現がある。各表現のベクトルは以下である。

$$c^H = (z^H \bar{z}^H), \quad z = (I \ iI)r, \quad r^T = (x^T \ y^T)$$

ここで、右肩の H, T は共役転置および転置を示し、 \bar{z} は z の複素共役を示す。 i は虚数、 I は単位行列を示し、 x, y は実ベクトルである。

目的関数をスカラー実関数 E としたとき、勾配 $\frac{\partial E}{\partial r}$ と $\frac{\partial E}{\partial \bar{z}}$ には、以下の関係が成り立つ。

$$\frac{\partial E}{\partial r^T} = 2 \left(\text{Re} \left[\frac{\partial E}{\partial \bar{z}^T} \right] \quad \text{Im} \left[\frac{\partial E}{\partial \bar{z}^T} \right] \right) \quad (3)$$

勾配 $\frac{\partial E}{\partial \bar{z}}$ は、 $\frac{\partial E}{\partial r}$ よりも導出が容易である場合が多い。

3.1 最急降下法

最急降下法は、目的関数の勾配の計算を必要とする。目的関数は以下のスカラー実関数とする。

$$E = \frac{1}{N} \sum_{\mu} \sum_k^K \delta_k^\mu \bar{\delta}_k^\mu, \quad \delta_k^\mu = f_k^\mu - y_k^\mu \quad (4)$$

ここで、 y_k^μ は出力ユニット k のサンプル μ における教師信号を示し、 N はサンプル数とする。以下、勾配の各成分を示す。

$$\frac{\partial E}{\partial w_{ij}} = \frac{1}{N} \sum_{\mu} \sum_k^K \delta_k^\mu \overline{w_{jk} g'(h_j^\mu) x_i^\mu} \quad (5)$$

$$\frac{\partial E}{\partial \bar{w}_{jk}} = \frac{1}{N} \sum_{\mu} \delta_k^\mu z_j^\mu \quad (6)$$

重みの更新は、上記で求めた勾配に直線探索で求めた探索幅を乗じて実行する。

学習の終了は、以下の条件式を満たすか、任意指定の最大反復回数 T_{max} に達したら終了とする。

$$E_{t-1} - E_t \leq \theta E_t \quad (7)$$

ここで、 E_{t-1}, E_t は重み更新前後の目的関数を示す。 θ は終了判定係数とする。

3.2 BFGS 法

BFGS 法の複素モデルへの拡張は、Wirtinger 微分の r 表現を用いることで、実モデルと同様に論じられる。 r 表現における BFGS 公式は以下である^{5, 3)}。

$$\mathbf{G}_{t+1} = \mathbf{G}_t + \left(1 + \frac{\Delta \mathbf{g}_t^T \mathbf{G}_t \Delta \mathbf{g}_t}{\Delta \mathbf{g}_t^T \Delta \mathbf{r}_t}\right) \frac{\Delta \mathbf{r}_t \Delta \mathbf{r}_t^T}{\Delta \mathbf{g}_t^T \Delta \mathbf{r}_t} - \frac{\Delta \mathbf{r}_t \Delta \mathbf{g}_t^T \mathbf{G}_t + \mathbf{G}_t \Delta \mathbf{g}_t \Delta \mathbf{r}_t^T}{\Delta \mathbf{g}_t^T \Delta \mathbf{r}_t} \quad (8)$$

ここで、 \mathbf{G}_{t+1} は Hesse 逆行列の近似を示す。 \mathbf{g}_t は勾配であり、 \mathbf{r}_t は全重みを列挙した実ベクトルである。 $\Delta \mathbf{g}_t = \mathbf{g}_{t+1} - \mathbf{g}_t$ 、 $\Delta \mathbf{r}_t = \mathbf{r}_{t+1} - \mathbf{r}_t$ とする。

複素モデルでは、 \mathbf{g}_t を以下とする。

$$\mathbf{g}_t^T = 2 \left(\text{Re} \left[\frac{\partial E}{\partial \mathbf{z}_t^T} \right] \quad \text{Im} \left[\frac{\partial E}{\partial \mathbf{z}_t^T} \right] \right) \quad (9)$$

ここで、 \mathbf{z}_t は z 表現の重みベクトルである。 勾配 $\partial E / \partial \mathbf{z}_t$ としては、式 (5) および式 (6) が利用できる。 重みの更新は、探索方向 ($= -\mathbf{G}_t \mathbf{g}_t$) に直線探索で求めた探索幅を乗じて実行する。 学習の終了は、式 (7) を満たすか、任意指定の最大反復回数に達したら終了とする。

3.3 Levenberg-Marquardt 法

入力層から隠れ層、隠れ層から出力層の各重みをそれぞれ $\mathbf{w}^{(1)}, \mathbf{w}^{(2)}$ と表記する。 本稿で扱う複素 MLP は、出力層が線形であるため、 $\mathbf{w}^{(2)}$ だけを最適化するとき、目的関数を最も減少させる $\mathbf{w}^{(2)}$ は一意に決まる。 後述する可約性写像探索法では、 $\mathbf{w}^{(2)}$ の最適化法に Levenberg-Marquardt 法 (以下、LM 法) を用いる。 隠れ層から出力ユニット k への重みベクトル $\mathbf{w}_k^{(2)}$ の更新式を以下に示す。

$$\mathbf{w}_k^{(2)new} \leftarrow \mathbf{w}_k^{(2)} + \Delta \mathbf{w}_k^{(2)} \quad (10)$$

LM 法から求まる $\Delta \mathbf{w}_k^{(2)}$ は、以下の極値条件を満たす $\Delta \mathbf{w}_k^{(2)}$ と等価である。

$$\frac{\partial E}{\partial \mathbf{w}_k^{(2)new}} = \mathbf{0} \quad (11)$$

すなわち、 $\Delta \mathbf{w}_k^{(2)}$ は以下となる。

$$\Delta \mathbf{w}_k^{(2)} = -(\mathbf{Z}^H \mathbf{Z} + \lambda \mathbf{I})^{-1} (\mathbf{Z}^H \mathbf{d}_k) \quad (12)$$

ここで、 \mathbf{Z} は隠れユニットの出力であり、 $N \times (J+1)$ 行列である。 \mathbf{d}_k は出力ユニット k における複素 MLP の出力と教師信号の残差を示すベクトルであり、各成分は式 (4) の δ から求まる。 λ は正則化係数であり、 \mathbf{I} は単位行列である。 $\mathbf{Z}^H \mathbf{Z}$ の対角要素が非負であるため、 λ は正とする。 本稿では、 $\lambda = 10^{-8}$ とする。

4 可約性写像探索法

隠れユニット数 J の複素 MLP (J) から複素 MLP ($J+1$) への出力等価な写像を考えることができる⁸⁾。 図 2 の写像により、隠れユニット z_{J+1} を追加して、良質の解を探索する方法を可約性写像探索法 (reducibility mapping search method) と呼ぶ。 ここで、隠れユニツ

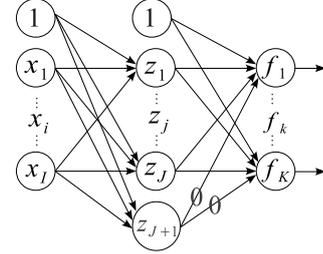


Fig. 2: 可約性写像によって得られる MLP ($J+1$)

ト $J+1$ から出力層への各重みは 0 に初期化する。 このとき、入力層から隠れユニット $J+1$ への各重みは自由となる。

可約性写像探索法は鈴村ら^{7, 6)} によって 2011 年に提案されたが、新たに LM 法を用いたものをバージョン 2 とする。 可約性写像探索法 (ver. 2) では、自由な重み $\{w_{i,J+1} \text{ for } i\}$ をランダムに初期化し、目的関数を最も減少させる $\mathbf{w}^{(2)}$ を LM 法によって求める。 式 (5) から分かるように、 $\mathbf{w}^{(2)}$ の変化に伴って $\mathbf{w}^{(1)}$ に関する勾配が変化するため、MLP ($J+1$) から再学習すれば、目的関数のさらなる減少が見込める。 以下に可約性写像探索法のアルゴリズムを示す。

可約性写像探索法 (ver. 2)

- 1: 隠れユニット数を $J = 0$ とする。
- 2: **while** $J < J_{max}$ **do**
- 3: コスト (目的関数の値) を $E(J+1) = \infty$ とする。
- 4: **for** $n = 1, 2, \dots, N_{max}$ **do**
- 5: 重み $\mathbf{w}(J)$ から $\mathbf{w}(J+1)$ を作成する。
- 6: $\{w_{i,J+1} \text{ for } i\} \in \mathbf{w}(J+1)$ をランダムに初期化する。
- 7: LM 法を実行する。
- 8: BFGS 法を実行する。
- 9: 学習後のコスト並びに重みを E', \mathbf{w}' とする。
- 10: **if** $E' < E(J+1)$ **then**
- 11: $E(J+1) \leftarrow E', \mathbf{w}(J+1) \leftarrow \mathbf{w}'$
- 12: **end if**
- 13: **end for**
- 14: $J \leftarrow J+1$
- 15: **end while**

ここで、 J_{max} は最大隠れユニット数であり、 N_{max} は可約性写像探索法の最大探索回数である。

【活性化関数と可約性写像探索法の相性】

本稿で扱う活性化関数とは直接関係ないが、値域が正に限定された活性化関数を複素 MLP に用いたとき、可約性写像探索法において、隠れユニットを追加しても訓練誤差を減少させられないことがある。 活性化関数の出力に負の値が存在しないとき、訓練誤差を減少

さらされる経路が発見できないと考えられる。その原因は今後追求しなければならない。

5 汎化性能の向上

重みの正則化や、早期停止 (early stopping) などの汎化性能向上化法が知られる。本稿では、各探索法において複素 MLP への入力 x_i^μ の振幅 $|x_i^\mu|$ を範囲 $[-1, 1]$ へスケール (scaling) することで、汎化性能を向上させる。

$$x_i^\mu \leftarrow x_i^\mu / \underset{\mu}{\operatorname{argmin}}(|x_i^\mu|) \quad (13)$$

入力のスケールは、歪んだ探索空間を補正する目的で用いられることが多いが^{5, 3)}、巨大な入力を抑えられるため、汎化性能を向上させる効果もある。なお、入力のスケールは学習の前処理であり、学習後は、 $1/\underset{\mu}{\operatorname{argmin}}(|x_i^\mu|)$ を入力層から隠れ層の重みに乗しておく。未知入力はスケールしない。

6 計算機実験

計算機実験を通して最急降下法、BFGS 法ならびに可約性写像探索法を汎化性能と計算時間の面で比較する。以下、3つの探索法をそれぞれ SD, BFGS, RM と表記する。本稿では、2種の人工問題を用いて実験する。実験条件は表 1 として、人工問題 1 と 2 で同じ設定値を用いる。最急降下法および BFGS 法では、乱数

Table 1: 実験条件

項目	設定値
最大隠れユニット数 J_{max}	20
最大反復回数 T_{max}	20000
最大探索回数 N_{max}	10
終了判定係数 θ	10^{-8}
重みの初期化範囲	$[-10, 10]$

のシード番号を変更して学習を N_{max} 回繰り返し、訓練誤差の最小値を求める。なお、可約性写像探索法は、内部処理として学習を N_{max} 回繰り返し、訓練誤差の最小値を求めている。

6.1 人工問題 1

複素 MLP は、周期性を含む問題に対して高い汎化性能を発揮すると期待される。ここでは、以下の人工問題を用いて実験する。

$$f(r, \theta) = re^{i\theta} \quad (14)$$

r は円の半径、 θ は角度を示す。学習データおよびテストデータは以下のように設定する。このとき、学習データの教師信号は図 3 となり、テストデータは図 4 となる。

学習データ：範囲 $[1, 5]$ にて r を $\frac{1}{2}$ の等間隔刻みに設定。範囲 $[-\pi, \frac{\pi}{2}]$ にて θ を $\frac{\pi}{10}$ の等間隔刻みに設定。

テストデータ：範囲 $[\frac{1}{2}, 10]$ にて r を $\frac{1}{4}$ の等間隔刻みに設定。範囲 $[-\pi, \pi]$ にて θ を $\frac{\pi}{20}$ の等間隔刻みに設定。

上記の学習データを 2 入力 1 出力の複素 MLP で学習する。学習後は、テストデータに対する複素 MLP の出力と、関数 $f(r, \theta)$ の真値との平均 2 乗和誤差 (以下、テスト誤差と呼ぶ) を評価する。

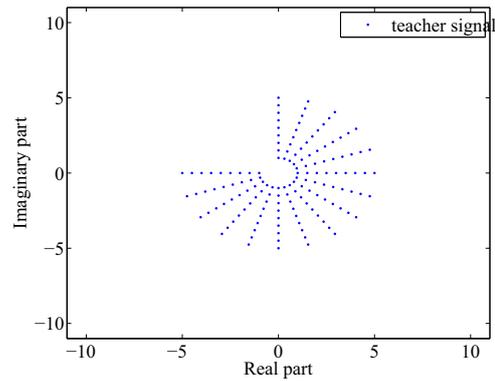


Fig. 3: 教師信号

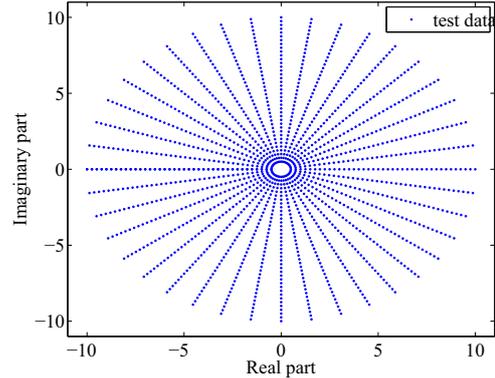


Fig. 4: テストデータ

Table 2: 計算時間の総計 (単位: 秒)

SD	BFGS	RM
6090.8	1707.4	38.525

【実験結果】各隠れユニット数における訓練誤差、テスト誤差ならびに計算時間を図 5, 図 6, 図 7 に示す。計算時間の総計を表 2 に示す。

【考察】図 5 に着目すると、隠れユニット数 $J=1$ から $J=6$ の間で訓練誤差を最小にした探索法は、BFGS 法であると分かる。一方、 $J=7$ 以降は、一貫して可約性写像探索法が最も訓練誤差を最小にした。図 6 に着目すると、 $J=1$ から $J=5$ の間で、テスト誤差を最小にした探索法は、BFGS 法であると分かる。しかし、 $J=6$ 以降は、BFGS 法のテスト誤差は非常に不安定となった。例えば、 $J=7$ において、真値を予測した結果の第 2 象限をそれぞれ図 8 (最急降下法)、図 9 (BFGS 法)、図 10 (可約性写像探索法) に示したが、BFGS 法は教師信号として与えていない部分が大きく欠損していた。重みの初期値を全てランダムとする BFGS 法では、学習の初期段階では複素 MLP の出力に歪を含んでいるため、歪を含んだまま教師信号を近似してしまい、テスト誤差が増加してしまったと考えられる。BFGS 法では、重みに適切な正則化を課す必要があると考えられる。一方、可約性写像探索法は、テスト誤差のゆらぎが比較的少ないことが分かる。可約性写像探索法では、自由度が小さいとき (歪が少ないとき) の重みを引き継ぐから、複素 MLP の出力が比較的滑らかになったと考えられる。なお、最急降下法では、全ての隠れユニット数において、真値の予測が不十分であった。

図 7 の計算時間に着目すると、最も高速に学習を進めた探索法は可約性写像探索法であると分かる。可約

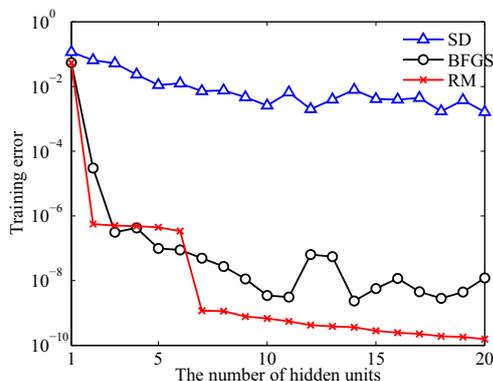


Fig. 5: 各隠れユニット数における訓練誤差

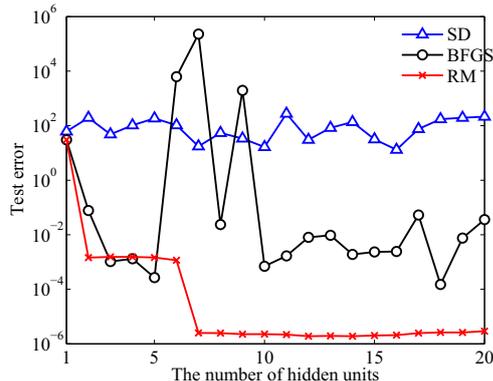


Fig. 6: 各隠れユニット数におけるテスト誤差

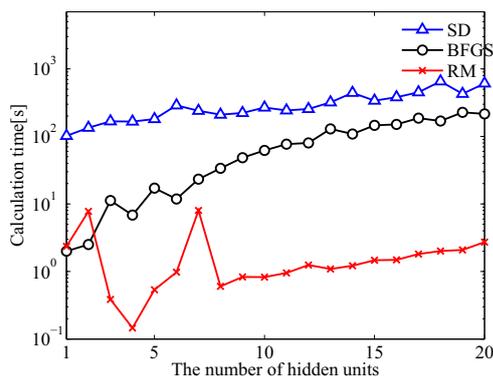


Fig. 7: 各隠れユニット数における計算時間 (単位: 秒)

性写像探索法では、訓練誤差を大きく改善した隠れユニット数 ($J=2$ と $J=7$) において、計算時間が大きくなる傾向があった。表 2 の計算時間の総計に着目すると、全体を通して可約性写像探索法は最急降下法に対して約 158 倍速く学習が進行し、BFGS 法に対しては、約 44 倍速く学習が進行したと分かる。可約性写像探索法は最適化された重みを再利用するから、高速な学習が可能であると考えられる。

7 人工問題 2

複素平面 $z = x + iy$ 上の点を補間する実験を行う。ここで、媒介変数 t を用いて $x = x(t), y = y(t)$ とする。 t は $t = 1, 2, \dots, 150$ として、 x, y を図 11 のように与える。なお、図上の * が教師信号として与える点であり、点を結ぶ線分はデータの並びを示すために描画した。 t を入力、 z を教師信号として、1 入力 1 出力の複素 MLP で学習する。学習後は、 $t = 1, 1.1, 1.2, \dots, 150$ のように、教師信号に比べて 10 倍細かく刻んだ t を複

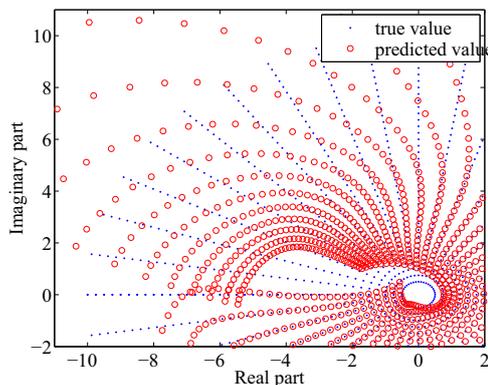


Fig. 8: 真値の予測結果 (SD, $J=7$)

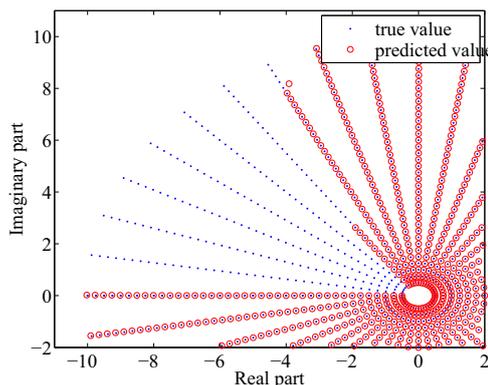


Fig. 9: 真値の予測結果 (BFGS, $J=7$)

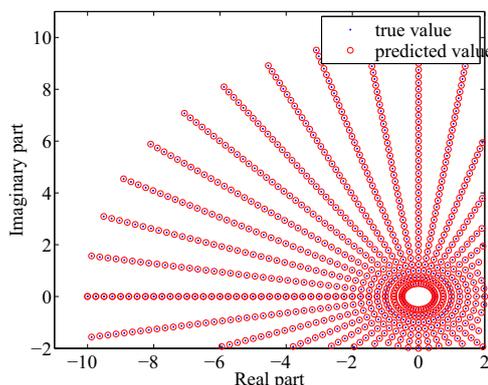


Fig. 10: 真値の予測結果 (RM, $J=7$)

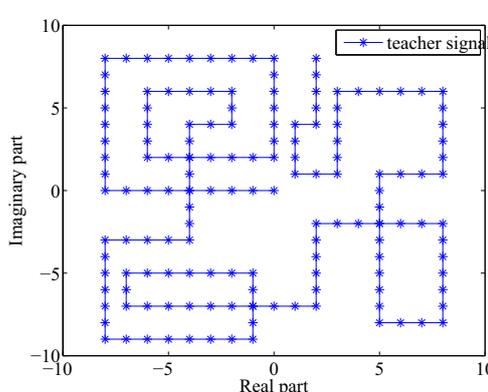


Fig. 11: 教師信号

素 MLP へ与え、その出力を可視化する。

【実験結果】各隠れユニット数における訓練誤差ならびに計算時間を図 12, 図 13 に示す。計算時間の総計を

表 3 に示す。各探索法の補間結果をそれぞれ図 14, 図 15, 図 16 に示す。ただし, 上から順に隠れユニット数 $J = 5, 10, 15, 20$ の補間結果を示した。

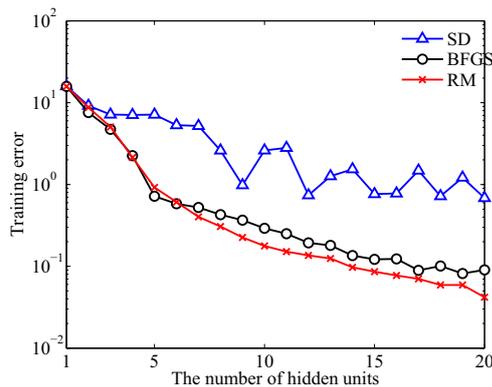


Fig. 12: 各隠れユニット数における訓練誤差

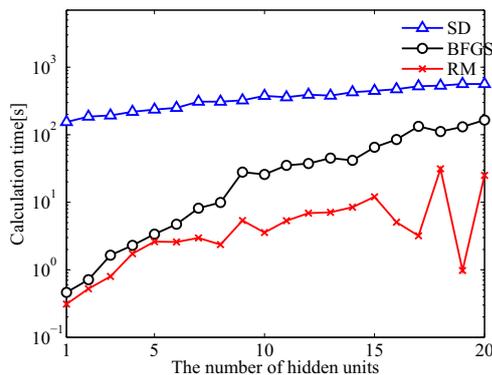


Fig. 13: 各隠れユニット数における計算時間 (単位: 秒)

Table 3: 計算時間の総計 (単位: 秒)

SD	BFGS	RM
7209.2	934.50	128.04

【考察】図 12 の訓練誤差に着目すると, $J = 1$ から $J = 5$ においては, BFGS 法と可約性写像探索法の訓練誤差の収束曲線の傾きが比較的急であると分かる。一方, $J = 6$ 以降は, 両収束曲線の傾きは比較的平坦であると分かる。図 13 の計算時間に関しては, $J = 1$ から $J = 5$ の間では, BFGS 法と可約性写像探索法に大差はなかった。しかし, $J = 6$ 以降は, 可約性写像探索法の計算時間が最も短いことが分かる。BFGS 法であれ可約性写像探索法であれ, 訓練誤差が大きく改善された隠れユニット数において, 計算時間が長くなる傾向があった。可約性写像探索法では, 訓練誤差がある程度小さくなった地点から探索を再開するから, 訓練誤差を大きく改善する労力が削減され, 計算時間が短くなったと言える。具体的にどれほど高速化されたかは, 表 3 の計算時間の総計に着目すると, 全体を通して可約性写像探索法は最急降下法に対して約 56 倍速く学習が進行し, BFGS 法に対しては, 約 7 倍速く学習が進行したと分かる。

図 14, 図 15, 図 16 の補間結果に着目すると, 最急降下法では補間曲線が硬質的であると分かる。BFGS 法と可約性写像探索法では, 補間曲線が軟質的であった。補間点にノイズを含めた場合, BFGS 法と可約性写像探索法では, 隠れユニット数を少なめに選ぶ必要があると考えられる。なお, 図は省略するが, 隠れユニッ

ト数 $J = 1$ から $J = 20$ まで全ての補間結果可視化したとき, 人工問題 1 の最急降下法や BFGS 法のように, 内挿した部分が欠損したり, 大きく歪むことはなかった。補間を滑らかに行うには, サンプル数が十分であったと考えられる。

8 まとめ

本稿では, BFGS 法と可約性写像を用いた可約性写像探索法を提案し, 計算機実験によって提案法の有効性を評価した。実験結果から, 可約性写像探索法は汎化性能に優れ, 格段に速い学習が行えることが分かった。可約性写像探索法の内部処理を並列化すれば, さらに高速化が見込める。今後の課題は, 可約性写像探索法と活性化関数の相性に関する考察, 枝刈り法の適応, そして実ニューラルネットワークとの比較である。

参考文献

- 1) K. K. Delgado. The complex gradient operator and the CR-calculus. ECE275A-Lecture Supplement (2006)
- 2) A. Hirose. *Complex-Valued Neural Networks*. Springer, 2nd edition (2012)
- 3) D. G. Luenberger and Y. Ye. *Linear and Nonlinear Programming*. Springer-Verlag, 3rd edition (2008)
- 4) T. Nitta, editor. *Complex-valued Neural Networks: Utilizing High-dimensional Parameters*. Information Science Reference (2009)
- 5) J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer-Verlag, 2nd edition (2006)
- 6) 鈴木真矢, 中野良平. 固有ベクトル降下法と可約性写像を用いた複素多層パーセプトロン探索法. 信学技報, NC2011-88, 25/30 (2011)
- 7) 鈴木真矢, 中野良平. 複素多層パーセプトロンの探索空間と探索法. 第 1 回コンピューターショナル・インテリジェンス研究会, 85/91 (2011)
- 8) 新田徹. 複素ニューラルネットワークにおける一意性定理とパラメータの冗長性. 信学論, J85-D-II(5), 796/804 (2002)
- 9) 廣瀬明. 複素ニューラルネットワーク. サイエンス社 (2005)

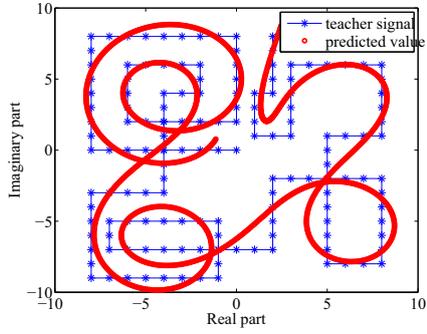
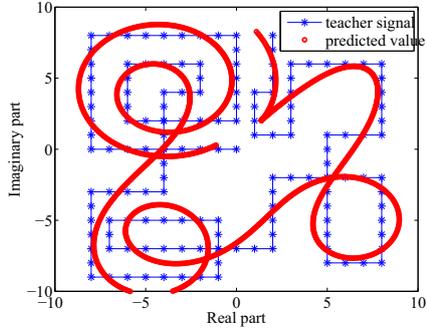
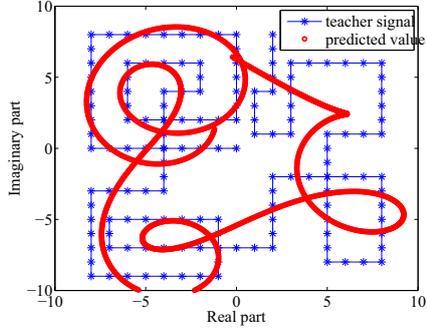
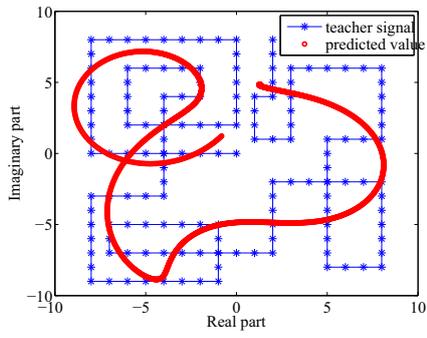


Fig. 14: 補間結果 (SD)

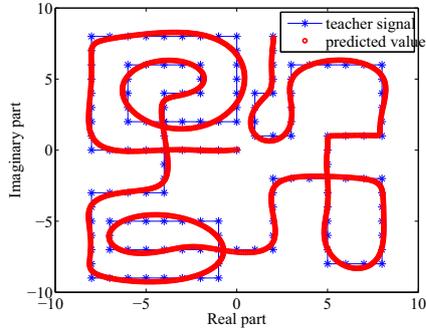
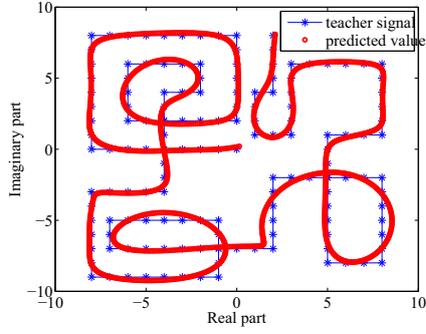
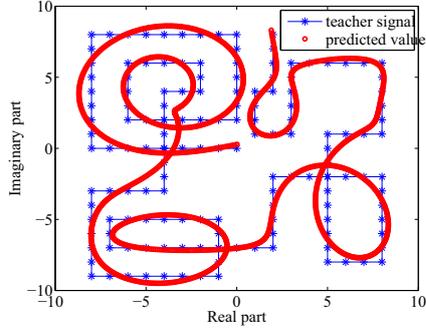
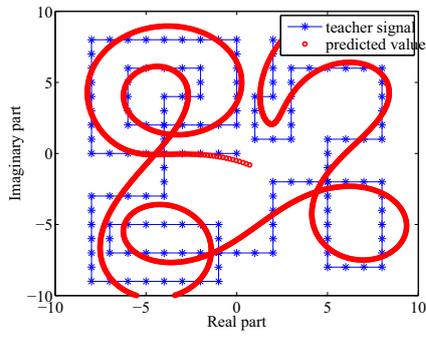


Fig. 15: 補間結果 (BFGS)

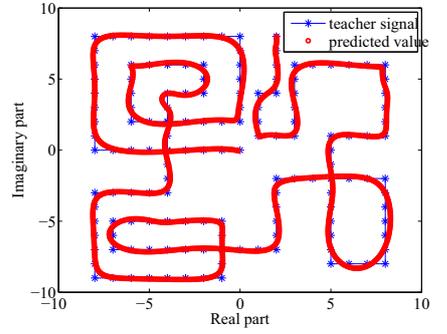
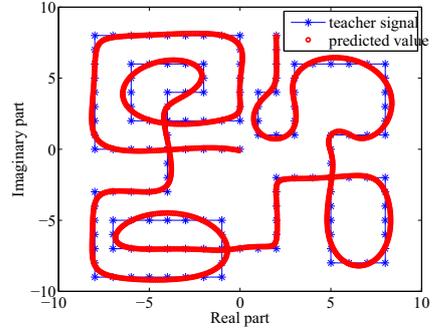
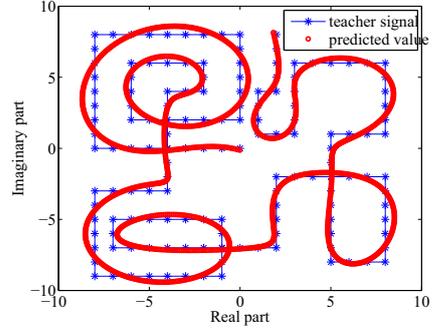
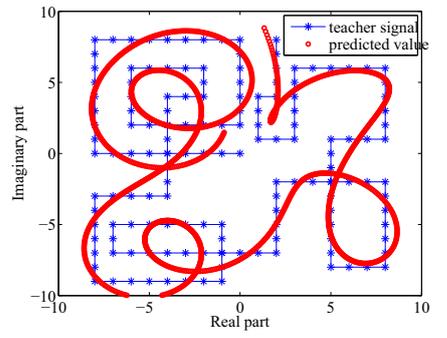


Fig. 16: 補間結果 (RM)

階層構造に基づいた危点を持たないニューラルネットワークの構成

○新田 徹 (産業技術総合研究所)

Construction of Neural Networks without Critical Points based on Hierarchical Structure

*Tohru Nitta (National Institute of Advanced Industrial Science and Technology, AIST)

Abstract - A critical point is a point on which the derivatives of an error function are all zero. It was proved in 2000 that the critical points caused by the hierarchical structure of the real-valued neural network could be local minima or saddle points. In the case of complex-valued neural networks, most of the critical points are saddle points unlike those of the real-valued neural network. Several studies have demonstrated that those kind of singularity has a negative effect on learning dynamics in learning machines. In this paper, the decomposition of high-dimensional NNs into low-dimensional NNs equivalent to the original NNs yields the NNs which do not have critical points based on the hierarchical structure.

Key Words: Neural network, Critical point, Singular point, Redundancy, Complex number, Quaternion

1 はじめに

一般に、ニューラルネットワーク (NN) の学習の目的は、誤差関数のグローバルミニマムを求めることである。もし、 ω_* が誤差関数 $E(\omega)$ のグローバルミニマムならば、 $\partial E(\omega_*)/\partial \omega = \mathbf{0}$ が成り立つ。しかし、 $\partial E(\omega_*)/\partial \omega = \mathbf{0}$ が成り立つからといって、 ω_* がグローバルミニマムであるとは限らない。 $\partial E(\omega_*)/\partial \omega = \mathbf{0}$ を満たす点 ω_* は E の危点と呼ばれる。危点は、ローカルミニマム (極小点)、ローカルマキシマム (極大点)、鞍点のいずれかになり得る。

福水らは、実 NN (実数値信号を扱う通常の NN) における階層構造に起因するローカルミニマムの存在を数理的に証明した¹⁾。つまり、 $H-1$ 個の中間ニューロンを持つ 3 層実 NN における危点は、 H 個の中間ニューロンを持つ 3 層実 NN の危点になり、それらはローカルミニマムかまたは鞍点になることを示した。この種の危点は実 NN の特異点となり、学習の停滞をもたらす。

複素ニューラルネットワーク (複素 NN) は、通常の NN の重みや閾値といった (実数値) パラメータを複素数に拡張したもので、複素データや 2 次元データを情報処理するのに適している^{2,3)}。3 層複素 NN の危点に関しても、3 層実 NN の危点と同様のことが示されている⁴⁾： $H-1$ 個の中間ニューロンを持つ 3 層複素 NN における危点は、 H 個の中間ニューロンを持つ 3 層複素 NN の危点になり、それら (のほとんど) は鞍点である。

近年、こういった特異点が調べられつつある^{5,6,7,8)}。階層型 NN や混合正規分布といった、階層構造や結合荷重の交換に関する対称性を持った学習モデルは概ね特異点を持っている。特異点は学習モデルの学習ダイナミクスに影響を与え、学習の停滞を招く原因であることなどが分かっている。

本稿では、高次元 NN をそれと等価な低次元の

NN に分解することによって、階層構造に基づいた危点を持たない NN を構成できることを示す。

2 解析

本章では、高次元 NN をそれと等価な低次元 NN に分解することによって、階層構造に基づいた危点を持たない NN を構成する。

2.1 2-2-2 実ニューラルネットワークの構成

1-1-1 複素 NN から、階層構造に基づく危点を持たない 2-2-2 実 NN を構成する。

次のような 1-1-1 複素 NN を考える (ネット 1 と呼ぶ)。入力ニューロンと中間ニューロンの間の重みを $a+ib \in \mathbf{C}$ 、中間ニューロンと出力ニューロンの間の重みを $v+iw \in \mathbf{C}$ 、ただし、 $a+ib \neq 0, v+iw \neq 0$ と仮定する。ここで、 \mathbf{C} は複素数全体の集合を表す。中間ニューロンの閾値を $c+id \in \mathbf{C}$ 、出力ニューロンの閾値を $p+iq \in \mathbf{C}$ 、入力を $x+iy \in \mathbf{C}$ 、出力を $X+iY \in \mathbf{C}$ とする。中間ニューロンの活性化関数を

$$f_C(z) = \tanh(x) + i \tanh(y), \quad z = x + iy, \quad (1)$$

出力ニューロンの活性化関数を

$$g_C(z) = z, \quad z = x + iy \quad (2)$$

とする。ネット 1 は、図 1 に示す 2-2-2 実 NN (ネット 2 と呼ぶ) に等価であることが容易にわかる⁹⁾。

命題 1 ネット 2 には、階層構造に基づいた危点が存在しない。□

(証明) ネット 2 から中間ニューロン 1 を取り除いた 2-1-2 実 NN (ネット 3 と呼ぶ) を考える (図 2)。ネット 3 の学習パラメータは危点であり、写像

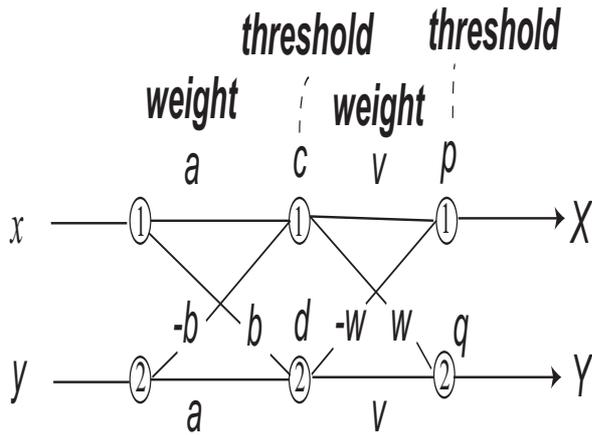


Fig. 1: ネット 2. ネット 1 と等価な 2-2-2 実 NN. 重みの間に制約がある. $x, y, a, b, c, d, v, w, p, q, X, Y$ はすべて実数値.

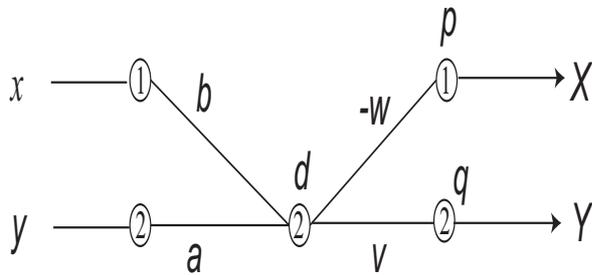


Fig. 2: ネット 3. ネット 2 から中間ニューロン 1 を取り除いた 2-1-2 実 NN.

$F(x, y)$ を実現していると仮定する. ネット 3 に, 取り除いた中間ニューロン 1 を再び加えることによって, 同じ写像 $F(x, y)$ を実現できるのは次の 3 つの場合に限られる¹⁾.

1. 付け加える中間ニューロン 1 と出力ニューロン (1 および 2) との間の 2 つの重みがともに 0 の場合.
この場合, $v = w = 0$ でなければならない. しかし, これは仮定 $v + iw \neq 0$ に反する.
2. 付け加える中間ニューロン 1 と入力ニューロン (1 および 2) との間の重みベクトル w_1 が 0 の場合.
この場合, $a = b = 0$ でなければならない. しかし, これは仮定 $a + ib \neq 0$ に反する.
3. 付け加える中間ニューロン 1 と出力ニューロン 1 との間の重みを ν_{11} , 付け加える中間ニューロン 1 と出力ニューロン 2 との間の重みを ν_{12} , 中間ニューロン 2 と出力ニューロン 1 との間の重みを ν_{21} , 中間ニューロン 2 と出力ニューロン 2 との間の重みを ν_{22} とする. また, 付け加える中間ニューロン 1 と入力ニューロン (1 および 2) との間の重みベクトルを w_1 , 中間ニューロン 2 と入力ニューロン (1 および 2) との間の重みベクトルを w_2 とする. このとき,

$w_1 = \pm w_2$ であり, かつ, $\nu_{11} + \nu_{21} = -w$, $\nu_{12} + \nu_{22} = v$ の場合.

この場合, $a = b = 0$ でなければならない. しかし, これは仮定 $a + ib \neq 0$ に反する.

故に, ネット 3 に元の間ニューロン 1 を付け加えて, ネット 2 の重みの構造を持つ形で, 写像 $F(x, y)$ を実現することはできない.

以上のことは, 中間ニューロン 1 を取り除くという操作をした場合であったが, 中間ニューロン 2 を取り除いた場合も同様にして同じ結論が得られる. 従って, ネット 2 は, (実 NN としての) 階層構造に基づいた危点を持たない. (証明終)

2.2 4-4-4 実ニューラルネットワークの構成

1-1-1 四元数 NN から, 階層構造に基づく危点を持たない 4-4-4 実 NN を構成する.

次のような 1-1-1 四元数 NN を考える (ネット 4 と呼ぶ). 入力ニューロンと中間ニューロンの間の重みを $A = a + ib + jc + kd \in \mathcal{Q}$, 中間ニューロンと出力ニューロンの間の重みを $B = \alpha + i\beta + j\gamma + k\delta \in \mathcal{Q}$, ただし, $A \neq 0, B \neq 0$ と仮定する. ここで, \mathcal{Q} は四元数全体の集合を表す. 中間ニューロンの閾値を $C = p + iq + jr + ks \in \mathcal{Q}$, 出力ニューロンの閾値を $D = \mu + i\nu + j\rho + k\sigma \in \mathcal{Q}$, 入力を $I = v + iw + jx + ky \in \mathcal{Q}$, 出力を $O = V + iW + jX + kY \in \mathcal{Q}$ とする. 中間ニューロンの活性化関数を

$$f_{\mathcal{Q}}(u) = \tanh(u_1) + i \tanh(u_2) + j \tanh(u_3) + k \tanh(u_4),$$

$$u = u_1 + iu_2 + ju_3 + ku_4, \quad (3)$$

出力ニューロンの活性化関数を

$$g_{\mathcal{Q}}(u) = u, \quad u \in \mathcal{Q} \quad (4)$$

とする. ここで, 四元数は乗法に関して非可換なので, 入力値と重みの掛け算の順番によって, 計算結果が異なる ($IA \neq AI$). そのため, 正規四元数ニューロン (normal quaternary neuron, AI を計算) と逆四元数ニューロン (inverse quaternary neuron, IA を計算) の 2 つの種類四元数ニューロンが存在する^{10, 11)}. 本稿では, 手始めに, 逆四元数ニューロンだけから構成される四元数 NN を扱う.

ネット 4 は, 図 3 に示す 4-4-4 実 NN (ネット 5 と呼ぶ) に等価であることが容易にわかる.

命題 2 ネット 5 には, 階層構造に基づいた危点が存在しない. \square

(証明) ネット 5 から中間ニューロン 1 を取り除いた 4-3-4 実 NN (ネット 6 と呼ぶ) を考える. ネット 6 の学習パラメータは危点であり, 写像 $F(v, w, x, y)$ を実現していると仮定する. ネット 6 に, 取り除いた中間ニューロン 1 を再び加えることによって, 同じ写像 $F(v, w, x, y)$ を実現できるのは次の 3 つの場合に限られる¹⁾.

1. 付け加える中間ニューロン 1 と出力ニューロン (1 ~ 4) との間の 4 つの重みがすべて 0 の場合.
この場合, $B = 0$ でなければならない. しかし, これは仮定 $B \neq 0$ に反する.

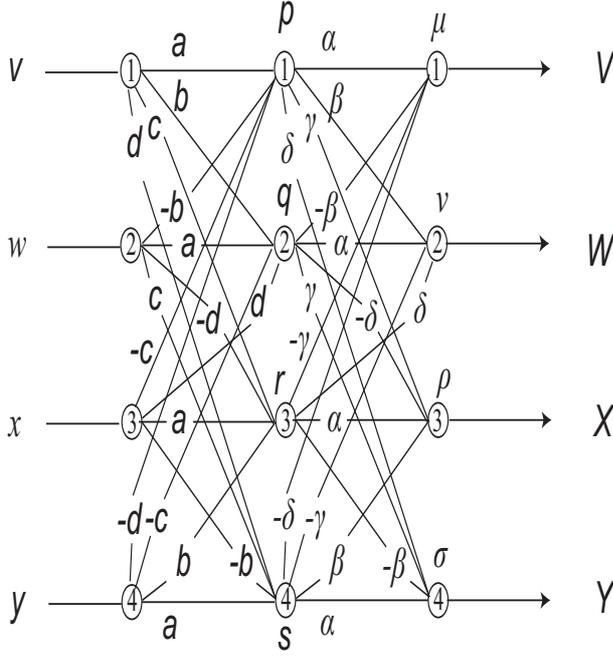


Fig. 3: ネット5. ネット4と等価な4-4-4実NN. 重みの間に制約がある. $v, w, x, y, a, b, c, d, \alpha, \beta, \gamma, \delta, p, q, r, s, \mu, \nu, \rho, \sigma, V, W, X, Y$ はすべて実数値.

2. 付け加える中間ニューロン1と入力ニューロン(1~4)との間の重みベクトル w_1 が $\mathbf{0}$ の場合.

この場合, $A = 0$ でなければならない. しかし, これは仮定 $A \neq 0$ に反する.

3. 中間ニューロン2~4から一つ選び, それを中間ニューロン j と記すことにする. 付け加える中間ニューロン1と入力ニューロン(1~4)との間の重みベクトルを w_1 , 中間ニューロン j と入力ニューロン(1~4)との間の重みベクトルを w_j とする. このとき, $w_1 = \pm w_j$ の場合(中間層と出力層の間の重みに関する条件もあるが, 証明に使わないので省略した). この場合, $A = 0$ でなければならない. しかし, これは仮定 $A \neq 0$ に反する.

故に, ネット6に元の間ニューロン1を付け加えて, ネット5の重みの構造を持つ形で, 写像 $F(v, w, x, y)$ を実現することはできない.

以上のことは, 中間ニューロン1を取り除くという操作をした場合であったが, 中間ニューロン2~4のいずれかを取り除いた場合も同様にして同じ結論が得られる. 従って, ネット5は, (実NNとしての) 階層構造に基づいた危点を持たない. (証明終)

2.3 2-2-2 複素ニューラルネットワークの構成

1-1-1 四元数NNから, (複素NNとしての) 階層構造に基づく危点を持たない2-2-2複素NNを構成する.

2.2節において定義した1-1-1四元数NN(ネット4)に, 次の仮定をおいたものをネット7と呼ぶ:

$$a + ib \neq \pm i(c + id), \quad (5)$$

$$C = 0. \quad (6)$$

式(5)には次のような意味がある. ネット7の入力ニューロンと中間ニューロンの重み A は次のように書ける (Cayley-Dickson notation):

$$\begin{aligned} A &= a + ib + jc + kd \\ &= x_1 + x_2j. \end{aligned} \quad (7)$$

ここで, $x_1 = a + ib, x_2 = c + id \in \mathbb{C}$ とおいた. 式(7)により, 式(5)は, $x_1 \neq \pm ix_2$ と書ける. x_1 と x_2 をそれぞれ2次元ベクトルと考えると, それらは直交していないことになる. また, $x_1 = \pm ix_2$ ならば, $A = x_1 \pm x_2j = x_1(1 \pm k)$ だから, 重み A は x_1 の情報だけ持つことになる. つまり, 式(5)は, このような情報量の少ない特別な場合は除外することを意味する. また, 式(6)は, 中間ニューロンの閾値が0であることを意味するが, これは‘複素NNが可約(reducible)であるための条件’を適用するために必要である¹²⁾.

ネット7は, Cayley-Dickson notation を使うと, 図4に示す2-2-2複素NN(ネット8と呼ぶ)に等価であることがわかる. ここで, $v' = v + iw, x' = x + iy, a' = a + ib, c' = c + id, \alpha' = \alpha + i\beta, \gamma' = \gamma + i\delta, \mu' = \mu + i\nu, \rho' = \rho + i\sigma, V' = V + iW, X' = X + iY$ とおいた. 活性化関数は, 式(1), (2)のとおりである.

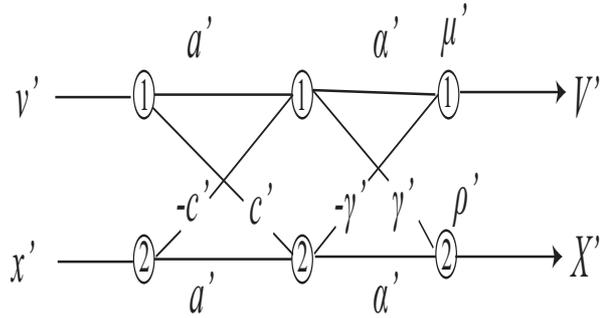


Fig. 4: ネット8. ネット7と等価な2-2-2複素NN. 重みの間に制約がある. $v', x', a', c', \alpha', \gamma', \mu', \rho', V', X'$ はすべて複素数値. 中間ニューロンの閾値はすべて0なので省略してある.

命題3 ネット8には, (複素NNとしての) 階層構造に基づいた危点が存在しない. \square

(証明) ネット8から中間ニューロン1を取り除いた2-1-2複素NN(ネット9と呼ぶ)を考える. ネット9の学習パラメータは危点であり, 複素写像 $F(v', x')$ を実現していると仮定する. ネット9に, 取り除いた中間ニューロン1を再び加えることによって, 同じ複素写像 $F(v', x')$ を実現できるのは次の3つの場合に限られる¹²⁾.

1. 付け加える中間ニューロン1と出力ニューロン(1~2)との間の2つの重みがすべて0の場合. この場合, $\alpha' = \gamma' = 0$ でなければならない. これは仮定 $B = \alpha' + \gamma'j \neq 0$ に反する.

2. 付け加える中間ニューロン1と入力ニューロン(1~2)との間の重みベクトル \mathbf{w}_1 が $\mathbf{0}$ の場合.
この場合, $a' = c' = 0$ でなければならない. これは仮定 $A = a' + c'j \neq 0$ に反する.
3. 付け加える中間ニューロン1への入力の総和 φ_1 と中間ニューロン2への入力の総和 φ_2 が回転等価である場合.
 $\varphi_1(v', x') = v'a' - x'c'$, $\varphi_2(v', x') = v'c' + x'a'$ であるから,
 - (a) $\varphi_1 \equiv \pm\varphi_2$ の場合
 $a' = c' = 0$ であることが必要. これは, 仮定 $A = a' + c'j \neq 0$ に反する.
 - (b) $\varphi_1 \equiv \pm i\varphi_2$ の場合
 $a' = \pm ic'$ であることが必要. これは, 仮定 $a + ib \neq \pm i(c + id)$ (式 (5)) に反する.

故に, ネット9に中間ニューロン1を付け加えて, ネット8の重みの構造を持つ形で, 写像 $F(v', x')$ を実現することはできない.

以上のことは, 中間ニューロン1を取り除くという操作をした場合であったが, 中間ニューロン2を取り除いた場合も同様にして同じ結論が得られる. 従って, ネット8は, (複素NNとしての) 階層構造に基づいた危点を持たない. (証明終)

本稿では, 1-1-1 四元数NNの中間ニューロンの閾値は0と仮定した(式 (6)). これは, '複素NNが可約(reducible)であるための3つの条件'(証明中の項番1~3)を適用するために必要であった¹²⁾. その結果, 得られた2-2-2複素NNの中間ニューロンの閾値はすべて0である. 中間ニューロンの閾値が必ずしも0でない1-1-1四元数NNを考えることによって, 中間ニューロンの閾値が必ずしも0でない2-2-2複素NNが得られる可能性がある. 中間ニューロンの閾値が必ずしも0でない3層複素NNが可約であるためには, 上記の3つの条件に加えて, 例外的可約性(exceptional reducibility)なる条件が必要になる¹³⁾.

3 おわりに

本稿では, 高次元NNをそれと等価な低次元のNNに分解することによって, 階層構造に基づいた危点を持たないニューラルネットワークを構成できることを示した. 具体的には, 次の3つの場合について示した. (a) 1-1-1複素NNから, 2-2-2実NNを構成. (b) 1-1-1四元数NNから, 4-4-4実NNを構成. (c) 1-1-1四元数NNから, 2-2-2複素NNを構成. 今後は, 次のことに取り組む予定である.

1. 本稿では, 逆四元数ニューロンだけから成る四元数NNを用いたが, 正規四元数ニューロンをも用いた場合についても調べる.
2. 例外的可約性¹³⁾を用いた, 中間ニューロンの閾値が必ずしも0でない一般の複素NNの構成について検討する.
3. 一般の $Cl_{p,q}$ 上のクリフォードNN¹⁴⁾をそれと等価な低次元のクリフォードNNに分解することによって, 階層構造に基づいた危点を持たない $m - n - l$ NNを構成する.

参考文献

- 1) Fukumizu, K. and Amari, S.: Local Minima and Plateaus in Hierarchical Structures of Multilayer Perceptrons, *Neural Networks*, **13-3**, 317/327 (2000).
- 2) 廣瀬明: 複素ニューラルネットワーク, SGCライブラリ 38, サイエンス社 (2005).
- 3) T. Nitta (Ed.): *Complex-Valued Neural Networks: Utilizing High-Dimensional Parameters*, Information Science Reference, Pennsylvania, USA (2009).
- 4) Nitta, T.: On the Critical Points of the Complex-valued Neural Network, *Proc. International Conference on Neural Information Processing, ICONIP'02-Singapore*, Nov. 19-22, Vol.3, 1099/1103 (2002).
- 5) 渡辺澄夫, 福水健次, 萩原克幸, 甘利俊一: 特異モデルの学習理論, *信学論 (D-II)*, Vol.J88-D-II, No.2, 159/169 (2005).
- 6) Amari, S., Park, H. and Ozeki, T.: Singularities Affect Dynamics of Learning in Neuromanifolds, *Neural Computation*, **18-5**, 1007/1065 (2006).
- 7) 渡辺澄夫: 代数幾何と学習理論, 森北出版 (2006).
- 8) 新田徹: 複素ニューロンの特異性について, *信学論 (D)*, Vol.J93-D, No.8, Aug. 1614/1621 (2010).
- 9) Nitta, T.: An Analysis of the Fundamental Structure of Complex-valued Neurons, *Neural Processing Letters*, **12-3**, 239/246 (2000).
- 10) Nitta, T.: A Quaternary Version of the Back-propagation Algorithm, *Proc. IEEE Int. Conf. on Neural Networks, ICNN'95-Perth*, Nov. 27-Dec. 1, Vol.5, 2753/2756 (1995).
- 11) Nitta, T.: A Solution to the 4-bit Parity Problem with a Single Quaternary Neuron, *Neural Information Processing - Letters and Reviews*, **5-2** (2005).
- 12) 新田徹: 複素ニューラルネットワークにおける一意性定理とパラメータの冗長性, *電子情報通信学会論文誌 DII*, Vol.J85-D-II, No.5, 796/804 (2002).
- 13) Kobayashi, M: Exceptional Reducibility of Complex-Valued Neural Networks, *IEEE Trans. Neural Networks*, **21-7**, 1060/1072 (2010).
- 14) 黒江康明, 新田徹, E. Hitzer: クリフォード代数の応用 - サーベイ -, *計測と制御*, **51-4**, 328/337 (2012).

ニューラルネットワークとモジュラー型ファジィ推論モデルの融合

○関宏理 (関西学院大学)

Union between Neural Networks and Modular Type Fuzzy Inference Model

*Hirosato Seki (Kwansei Gakuin University)

Abstract— This paper firstly surveys neuralnet work-type SIRMs fuzzy inference model which unifies neural network and SIRMs fuzzy inference model, and its applicability. Moreover, the neuralnet work-type SIRMs model is shown to be superior to the conventional model by applying it to identification of two nonlinear functions, medical data and Iris data. Finally, this paper proposes a *neuralnet work-type modular fuzzy model* which generalize single-input rulemodules of the neuralnetwork-type SIRMs model to multiple-input rule modules.

Key Words: Approximate Reasoning, Neural Networks, Modular type fuzzy inference model

1 はじめに

Mamdani がファジィ推論の概念をスチームエンジン実験装置の制御へ適用して以来、様々な分野でファジィ推論の研究と応用が行われてきた¹⁾。

Mamdani のファジィ推論モデル¹⁾ や高木-菅野のファジィ推論モデル (T-S 推論モデル)²⁾ などの if-then 形ファジィ推論法では、システムの全ての入力項目が if の前件部にセットされ、全ての出力項目が then の後件部にセットされる。そのため、規則数が膨大になり、ファジィ規則の設定や調整が困難となるなどの問題が生じてしまう。一方、湯場崎ら^{3, 4, 5, 6, 7, 8, 9)} により提案された、1 入力型の if-then 形式のファジィルール群からの推論出力を統合する単一入力ルール群 (SIRMs, Single Input Rule Modules) 結合型ファジィ推論モデルは、ファジィ規則数を大幅に削減でき、1 次遅れ+無駄時間系の制御や未拘束物体の軌道追跡制御、倒立振り子システムの安定化制御などへ応用され、良好な結果が得られている。

また、Seki ら^{10, 11)} は、SIRMs 推論法の後件部は定数であるが、これを関数に一般化した関数型単一入力ルール群ファジィ推論法を提案しており、非線形同定に適用し、良好な結果が得られている。

しかし、これらの SIRMs 推論法も排他的論理和 (XOR) のような非線形分離問題には対応できないことが明らかにされている¹⁰⁾。

そこで Seki ら^{12, 13)} は SIRMs 推論法とニューラルネットワークを融合した“ニューラルネットワーク型単一入力ルール群ファジィ推論モデル” (以後、簡単にニューラルネットワーク型 SIRMs 推論モデル) を提案し、排他的論理和の実現が可能であることを示した。また Watanabe ら^{14, 15, 16, 17)} は別のアプローチとして、少数の入力組み合わせ対応したモジュラー型ファジィ推論モデルを提案し、排他的論理和の実現や実システムにおいて良好な結果を示している。

本稿では、ニューラルネットワーク型 SIRMs 推論モデルを紹介し、文献^{12, 13)} で示された非線形関数の同定、医療データ、あやめデータに適用した結果も紹介する。また、上記よりニューラルネットワークとモジュラー型ファジィ推論モデルの融合の一般化についても言及する。

2 ニューラルネットワークとモジュラー型ファジィ推論モデルの融合

本章では SIRMs ファジィ推論モデル、ニューラルネットワーク型 SIRMs ファジィ推論モデル、モジュラー型ファジィ推論モデルについて復習する。

2.1 SIRMs 結合型ファジィ推論法

従来のファジィ推論法ではファジィ規則数は入力項目数が増えると指数関数的に増加する。これは、全ての入力項目が規則の前件部にセットされることにある。これに対して、湯場崎らは、各入力項目に対してその入力項目だけを前件部変数とする 1 入力型の if-then 形式のファジィルール群を定義して、各ルール群の推論結果の重み付き総和を最終推論結果とする単一入力ルール群 (SIRMs) 結合型ファジィ推論法 (以後、SIRMs 推論法) を提案し、規則数を大幅に減らすことができるとを示した。

以下に、 n 入力 1 出力に対して、入力項目数 n と同じ個数の 1 入力 1 出力型のルール群から構成される SIRMs 推論モデルを式 (1) に示す^{3, 4, 5, 6, 7, 8, 9)}。

$$\begin{aligned} \text{Rules-1} : \{x_1 = A_j^1 \longrightarrow y_1 = y_j^1\}_{j=1}^{m_1} \\ \vdots \\ \text{Rules-}i : \{x_i = A_j^i \longrightarrow y_i = y_j^i\}_{j=1}^{m_i} \\ \vdots \\ \text{Rules-}n : \{x_n = A_j^n \longrightarrow y_n = y_j^n\}_{j=1}^{m_n} \end{aligned} \quad (1)$$

ここで、 x_i は前件部変数、 y_i は後件部変数である。 A_j^i は前件部変数 x_i のファジィ集合、 y_j^i は後件部の実数出力値を意味する。ただし、ルール群の番号は $i = 1, 2, \dots, n$ を、 i 番目のルール群 Rules- i の中の規則番号は $j = 1, 2, \dots, m_i$ である。

各入力 x_i^0 が入力されたとき、ルール群 Rules- i における j 番目の規則の前件部の適合度 h_j^i は式 (2) で与えられ、その規則からの推論結果 y_i^0 は合成重心法により

式 (3) のように求められる。

$$h_j^i = A_j^i(x_i^0) \quad (2)$$

$$y_i^0 = \frac{\sum_{k=1}^{m_i} h_k^i y_k^i}{\sum_{k=1}^{m_i} h_k^i} \quad (3)$$

各入力項目 (すなわち, ルール群) の重視度を w_i ($i = 1, 2, \dots, n$) に設定した場合, SIRM_s 推論法の最終出力 y^0 は以下のように各ルール群の推論結果 y_i^0 の重視度 w_i 付き総和として定義される^{3, 4, 5, 6, 7, 8, 9}。

$$y^0 = \sum_{i=1}^n w_i y_i^0 \quad (4)$$

2.2 モジュラーファジィモデル^{14, 15, 16, 17}

2.2.1 ルール群に対する入力の定義

モジュラーファジィモデルを用いる際に, 予め入力の組み合わせを決めておく必要がある^{14, 15, 16, 17}。

今, モジュラーファジィモデルに対して, 以下の4入力の場合の入力ベクトル \mathbf{x} を考えよう。

$$\mathbf{x} = [x_1, x_2, x_3, x_4]^T \quad (5)$$

各ルール群の入力には入力ベクトル \mathbf{x} から2入力の組み合わせを用いるように設定したとする。すなわち,

$$\{P_1(\mathbf{x}), P_2(\mathbf{x}), P_3(\mathbf{x}), P_4(\mathbf{x}), P_5(\mathbf{x}), P_6(\mathbf{x})\} = \{(x_1, x_2), (x_1, x_3), (x_1, x_4), (x_2, x_3), (x_2, x_4), (x_3, x_4)\} \quad (6)$$

であり, $P_1(\mathbf{x}), \dots, P_6(\mathbf{x})$ が本モデルのルール群に対する入力となる。同様に, 3入力の組み合わせは以下のようになる。

$$\{P_1(\mathbf{x}), P_2(\mathbf{x}), P_3(\mathbf{x}), P_4(\mathbf{x})\} = \{(x_1, x_2, x_3), (x_1, x_2, x_4), (x_1, x_3, x_4), (x_2, x_3, x_4)\} \quad (7)$$

2.2.2 モジュラーファジィモデルの構造

以下に, l 入力 1 出力 k 個の入力の組み合わせに対して, ルール群の数 n ($= {}_l C_k$) と同じ個数の 1 入力 1 出力型のルール群から構成されるモジュラーファジィモデルを式 (8) に示す^{14, 15, 16, 17}。

$$\begin{aligned} \text{Rules-1} : \{P_1(\mathbf{x}) = A_j^1 \longrightarrow y_1 = f_j^1(P_1(\mathbf{x}))\}_{j=1}^{m_1} \\ \vdots \\ \text{Rules-}i : \{P_i(\mathbf{x}) = A_j^i \longrightarrow y_i = f_j^i(P_i(\mathbf{x}))\}_{j=1}^{m_i} \quad (8) \\ \vdots \\ \text{Rules-}n : \{P_n(\mathbf{x}) = A_j^n \longrightarrow y_n = f_j^n(P_n(\mathbf{x}))\}_{j=1}^{m_n} \end{aligned}$$

ここで, \mathbf{x} は予め決められた組み合わせの入力ベクトル, y_i は後件部変数である。 A_j^i は前件部変数 x_i のファジィ集合, y_j^i は後件部の実数出力値を意味する。ただ

し, ルール群の番号は $i = 1, 2, \dots, n$ を, i 番目のルール群 Rules- i 中の規則番号は $j = 1, 2, \dots, m_i$ である。

各入力 x_i^0 が入力されたとき, ルール群 Rules- i における j 番目の規則の前件部の適合度 h_j^i は式 (9) で与えられ, その規則からの推論結果 y_i^0 は合成重心法により式 (10) のように求められる。

$$h_j^i = A_j^i(P_i(\mathbf{x}^0)) \quad (9)$$

$$y_i^0 = \frac{\sum_{j=1}^{m_i} h_j^i f_j^i(P_i(\mathbf{x}^0))}{\sum_{j=1}^{m_i} h_j^i} \quad (10)$$

ルール群の重視度を w_i ($i = 1, 2, \dots, n$) に設定した場合, 従来の SIRM_s 推論モデルと同様に, モジュラーファジィモデルの最終出力 y^0 は以下のように各ルール群の推論結果 y_i^0 の重視度 w_i 付き総和として定義される^{14, 15, 16}。

$$y^0 = \sum_{i=1}^n w_i y_i^0 \quad (11)$$

2.3 ニューラルネットワーク型 SIRM_s ファジィ推論モデル

今日までファジィ推論とニューラルネットワークの融合に関する研究が盛んに行われている^{18, 19, 20, 21}。本節では SIRM_s 推論モデルとニューラルネットワークを融合した“ニューラルネットワーク型単一入力ルール群ファジィ推論法,” (以後, ニューラルネットワーク型 SIRM_s 推論法) を提案する。 n 入力 1 出力に対して, 入力項目数 n と同じ個数の 1 入力 1 出力型のルール群から構成されるニューラルネットワーク型 SIRM_s 推論法は従来の SIRM_s 推論法と同様に式 (1) で与えられる。また, 各入力 x_i^0 が入力されたとき, ルール群 Rules- i における j 番目の規則の前件部の適合度 h_j^i と, その規則からの推論結果 y_i^0 もそれぞれ同様に式 (2), (3) で与えられる。

従来の SIRM_s 推論法では, 最終推論値は式 (3) の重視度付き総和で求められたが, 本推論法では最終推論値を求める前に中間層を導入する。 S 個の素子からなる中間層を導入したとき, s 番目の中間層において各ルール群出力値の重み付き総和からなる入力層における閾値 θ_1^s を引いた値 T_s は以下のように与えられる。

$$T_s = \sum_{i=1}^n v_i^s y_i^0 - \theta_1^s \quad (12)$$

ここで, $s = 1, 2, \dots, S$ であり, v_i^s は s 番目の素子における y_i^0 に対する結合荷重, θ_1^s は s 番目の素子における閾値を表す。

s 番目の素子の出力 H_s はシグモイド関数を用いて以下のように与えられる。

$$H_s = \frac{1}{1 + e^{-T_s}} \quad (13)$$

各素子の出力 H_s の重視度 w_i 付き総和から中間層における閾値 θ_2 を引いた値 U は以下のように与えられる。

$$U = \sum_{s=1}^S w_s H_s - \theta_2 \quad (14)$$

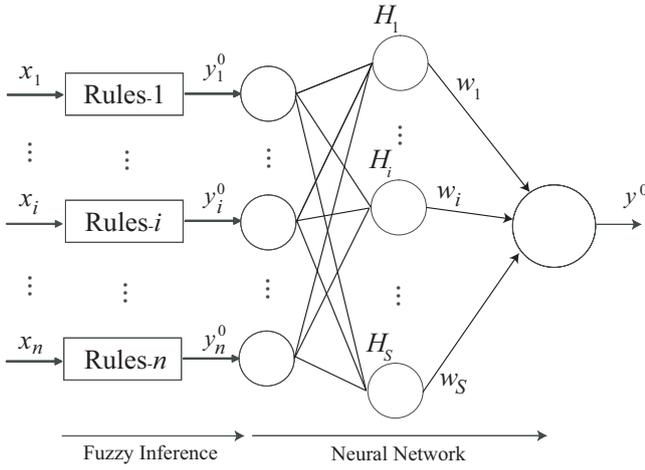


Fig. 1: ニューラルネットワーク型 SIRM 推論法の構造

本推論法の最終推論値 y^0 は式 (3), (12)~(14) とシグモイド関数を用いて以下のように求めることができる。

$$y^0 = \frac{1}{1 + e^{-U}} \quad (15)$$

ニューラルネットワーク型 SIRM 推論法の構造を図 1 に示す。

3 学習アルゴリズム

システムが複雑になるにつれて、ニューラルネットワークのパラメータやファジィ推論のメンバーシップ関数とファジィルールを設定することが難しくなる。また、たとえ試行錯誤でメンバーシップ関数とファジィルールが設定できるとしても、それは必ずしも最適とは限らない。そこで、システムの入出力データに基づいてニューラルネットワークの各々のパラメータやメンバーシップ関数とファジィルールを自動的に最適に作成することが望まれる。このことから、これらの学習アルゴリズムが数多く提案されている^{23, 24, 25, 26, 27}。以下では文献^{12, 13}で述べられるような本モデルに関する最急降下法を用いた学習アルゴリズムを示す。本モデルで学習するパラメータは前件部変数のメンバーシップ関数のパラメータ、後件部実数出力値、および入力項目の重視度である。

r 番目の入力パターン $(x_{r1}, \dots, x_{ri}, \dots, x_{rp})$ に対する真の出力を y^{Tr} とし、その教師パターンに対する本モデルの出力が y^{0r} であったとき、評価関数 E_r

$$E_r = \frac{1}{2}(y^{Tr} - y^{0r})^2 \quad (16)$$

により、真の出力と実際の出力の自乗誤差として定義する。

文献^{12, 13}では、ファジィ集合 $A_j^i(x_i)$ として三角形およびガウス型の 2 つを用いられている。これら 2 種類のファジィ集合のパラメータである中心 a_j^i 、幅 b_j^i と、後件部実数値の y_j^i 、および重視度 w_i は最急降下法により以下のように求められる。

三角形ファジィ集合の場合:

ファジィ集合 $A_j^i(x_i)$ が三角形の場合、

$$A_j^i(x_i) = \begin{cases} 1 - |x_i - a_j^i|/b_j^i; & a_j^i - b_j^i \leq x_i \leq a_j^i + b_j^i \\ 0; & \text{otherwise} \end{cases} \quad (17)$$

のように表される。ここで a_j^i は中心、 b_j^i は幅である。このことから、各パラメータの $t+1$ 回目における学習アルゴリズムは以下のように求められる。

$$\begin{aligned} a_j^i(t+1) &= a_j^i(t) + \alpha \cdot (y^T - y^0(t)) \cdot y^0(t) \\ &\quad \cdot (1 - y^0(t)) \cdot \sum_{q=1}^S w_q(t) H_q(t) (1 - H_q(t)) v_i^q(t) \\ &\quad \cdot \frac{y_j^i(t) - y_i^0(t)}{\sum_{k=1}^{m_i} h_k^i(t)} \cdot \frac{\text{sgn}(x_i - a_j^i(t))}{b_j^i(t)} \end{aligned} \quad (18)$$

$$\begin{aligned} b_j^i(t+1) &= b_j^i(t) + \beta \cdot (y^T - y^0(t)) \cdot y^0(t) \\ &\quad \cdot (1 - y^0(t)) \cdot \sum_{q=1}^S w_q(t) H_q(t) (1 - H_q(t)) v_i^q(t) \\ &\quad \cdot \frac{y_j^i(t) - y_i^0(t)}{\sum_{k=1}^{m_i} h_k^i(t)} \cdot \frac{|x_i - a_j^i(t)|}{(b_j^i(t))^2} \end{aligned} \quad (19)$$

$$\begin{aligned} y_j^i(t+1) &= y_j^i(t) + \gamma \cdot (y^T - y^0(t)) \cdot y^0(t) \cdot (1 - y^0(t)) \\ &\quad \cdot \sum_{q=1}^S w_q(t) H_q(t) (1 - H_q(t)) v_i^q(t) \cdot \frac{h_j^i(t)}{\sum_{k=1}^{m_i} h_k^i(t)} \end{aligned} \quad (20)$$

$$\begin{aligned} w_s(t+1) &= w_s(t) + \delta \cdot (y^T - y^0(t)) \cdot y^0(t) \cdot (1 - y^0(t)) \\ &\quad \cdot H_s(t) \end{aligned} \quad (21)$$

$$\begin{aligned} v_i^s(t+1) &= v_i^s(t) + \epsilon \cdot (y^T - y^0(t)) \cdot y^0(t) \cdot (1 - y^0(t)) \\ &\quad \cdot w_s(t) \cdot H_s(t) \cdot (1 - H_s(t)) \cdot y_i^0(t) \end{aligned} \quad (22)$$

ここで、 sgn は次の符号関数を表している。

$$\text{sgn}(x) = \begin{cases} 1 & : x > 0 \\ 0 & : x = 0 \\ -1 & : x < 0 \end{cases} \quad (23)$$

ガウス型ファジィ集合の場合:

ファジィ集合 $A_j^i(x_i)$ がガウス型の場合、

$$A_j^i(x_i) = \exp\left(-\frac{(x_i - a_j^i)^2}{b_j^i}\right) \quad (24)$$

のように表され、 a_j^i は中心、 b_j^i は分散である。このことから、各パラメータの学習アルゴリズムは以下の式

で求められる。

$$\begin{aligned}
a_j^i(t+1) &= a_j^i(t) + \alpha \cdot (y^T - y^0(t)) \cdot y^0(t) \\
&\quad \cdot (1 - y^0(t)) \cdot \sum_{q=1}^S w_q(t) H_q(t) (1 - H_q(t)) v_i^q(t) \\
&\quad \cdot \frac{y_j^i(t) - y_i^0(t)}{\sum_{k=1}^{m_i} h_k^i(t)} \cdot \frac{2(x_i - a_j^i(t))}{b_j^i(t)} \cdot h_j^i(t) \quad (25)
\end{aligned}$$

$$\begin{aligned}
b_j^i(t+1) &= b_j^i(t) + \beta \cdot (y^T - y^0(t)) \cdot y^0(t) \\
&\quad \cdot (1 - y^0(t)) \cdot \sum_{q=1}^S w_q(t) H_q(t) (1 - H_q(t)) v_i^q(t) \\
&\quad \cdot \frac{y_j^i(t) - y_i^0(t)}{\sum_{k=1}^{m_i} h_k^i(t)} \cdot \left(\frac{x_i - a_j^i(t)}{b_j^i(t)} \right)^2 \cdot h_j^i(t) \quad (26)
\end{aligned}$$

$$\begin{aligned}
y_j^i(t+1) &= y_j^i(t) + \gamma \cdot (y^T - y^0(t)) \cdot y^0(t) \cdot (1 - y^0(t)) \\
&\quad \cdot \sum_{q=1}^S w_q(t) H_q(t) (1 - H_q(t)) v_i^q(t) \cdot \frac{h_j^i(t)}{\sum_{k=1}^{m_i} h_k^i(t)} \quad (27)
\end{aligned}$$

$$\begin{aligned}
w_s(t+1) &= w_s(t) + \delta \cdot (y^T - y^0(t)) \cdot y^0(t) \cdot (1 - y^0(t)) \\
&\quad \cdot H_s(t) \quad (28)
\end{aligned}$$

$$\begin{aligned}
v_i^s(t+1) &= v_i^s(t) + \epsilon \cdot (y^T - y^0(t)) \cdot y^0(t) \cdot (1 - y^0(t)) \\
&\quad \cdot w_s(t) \cdot H_s(t) \cdot (1 - H_s(t)) \cdot y_i^0(t) \quad (29)
\end{aligned}$$

ここで、 t は学習回数、 α , β , γ , δ , ϵ は学習係数である。

4 数値実験

3では本モデルの学習アルゴリズムを示した。4では、本推論モデルと SIRM_s 推論モデルを用い、XOR に対して非線形分離を試みる¹²⁾。次に2入力1出力の非線形関数同定¹²⁾に適用することにより、本手法が強い非線形性を持つシステムにも対応できることを示す。さらに、医療診断システム¹³⁾とあやめ判別分析¹³⁾にも適用することにより、その適用可能性をも示す。この際、SIRM_s 推論モデルとの比較、検討を行う。

4.1 排他的論理和の実現

XORの同定において、入力項目の台集合をすべて5等分し、三角型メンバーシップ関数を隣同士が値0.5で交差するように設定する。さらに各入力項目の重視度の初期値をそれぞれ0.5に設定する。

XORに対する教師パターンとして表1を用いる。

次に、用いるテストパターンとして、表2に示すような、入力変数 x_1, x_2 を0から1まで0.05間隔で増加させて生成した441(21×21)個のデータを用いる。

Table 1: XOR に対する教師パターン

	x_1	x_2	y
d_1	0	0	0
d_2	0	1	1
d_3	1	0	1
d_4	1	1	0

Table 2: 評価パターン

	x_1	x_2
d_1	0	0
d_2	0	0.05
\vdots	\vdots	\vdots
d_{20}	0	0.95
d_{21}	0	1
d_{22}	0.05	0
d_{23}	0.05	0.05
\vdots	\vdots	\vdots
d_{440}	1	0.95
d_{441}	1	1

Table 3: XOR に対する評価誤差

	SIRM _s	NN-SIRM _s
d_1	0.5	1.60×10^{-2}
d_2	0.5	1.56×10^{-2}
d_3	0.5	1.23×10^{-2}
d_4	0.5	0.77×10^{-2}

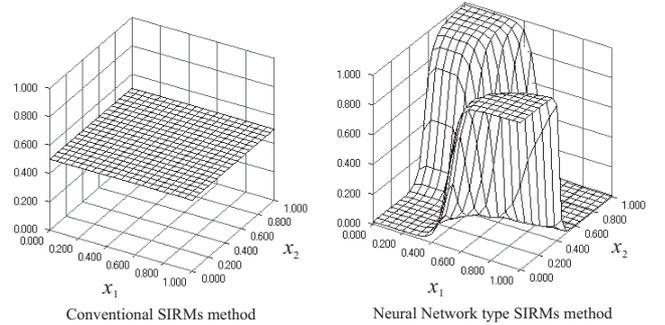


Fig. 2: XOR に対する各推論法による推論曲面¹²⁾

学習係数 α , β , γ , δ , ϵ はそれぞれ試行錯誤で0.001, 0.0001, 0.01, 0.01, 0.01に設定した。テストデータに対する平均自乗誤差を評価誤差として計算し、さらに理想の出力と実際の出力との絶対値の最大値を最大誤差として求める。

ニューラルネットワーク型 SIRM_s 推論法(表中では NN-SIRM_s と記す)と SIRM_s 推論法に対して、表1を入力値とし、それぞれ学習を行った。結果を表3に示す。

また、SIRM_s 推論法と本推論法の推論曲面を図2に示す。

以上の結果より、従来の SIRM_s 推論法で XOR を実現できないことに対し、本推論モデルでは理想に近い結果が得られることが分かった。

4.2 非線形関数の同定への適用

本章では、ニューラルネットワーク型 SIRM_s 推論モデルと上記の学習アルゴリズムを用いて、以下のような2種類の非線形関数の同定を行い、従来の SIRM_s 推

論モデルとの比較・検討を行う¹²⁾。

非線形関数 1: $y = (2x_1 + 4x_2^2 + 0.1)^2 / 37.21$

非線形関数 2: $y = (2 \sin(\pi x_1) + \cos(\pi x_2) + 3) / 6$

各入力項目の範囲をすべて $[-1, 1]$ とし、出力の範囲は $[0, 1]$ となる。

非線形関数の同定において、入力項目の台集合をすべて 5 等分し、三角型、ガウス型共にそれぞれ 1 つのメンバーシップ関数を隣同士が値 0.5 で交差するように設定する。さらに各入力項目の重視度の初期値をそれぞれ 0.5 に設定する。

教師パターンとしてランダムに生成したデータ 49 組を用いて、学習を 1000 回行う。そして、各入力項目を 0.04 刻みで順次変えて得た 2601 (51×51) 個の入出力データをテストパターンとする。学習係数 $\alpha, \beta, \gamma, \delta, \epsilon$ はそれぞれ試行錯誤で 0.001, 0.0001, 0.01, 0.01, 0.01 に設定した。テストデータに対する平均自乗誤差を評価誤差として計算し、さらに理想の出力と実際の出力との絶対値の最大値を最大誤差として求める。

非線形関数 1 に対して、学習回数を 1000 回に設定し、それぞれ 10 回のシミュレーションを行った。評価誤差、最大誤差に対する結果を表 4, 5 に示す。

これらの結果から、ニューラルネットワーク型 SIRMs 推論法の精度の良さがうかがえる。

非線形関数 2 に対して学習回数を 1000 回に設定し、それぞれ 10 回のシミュレーションを行った。評価誤差、最大誤差に対する結果を表 6, 7 に示す。

Table 4: 非線形関数 1 に対する評価誤差¹²⁾

Case	三角型		ガウス型	
	SIRMs	NN-SIRMs	SIRMs	NN-SIRMs
1	0.0849	0.0312	0.0889	0.0289
2	0.0687	0.0351	0.0776	0.0344
3	0.0733	0.0280	0.0900	0.0296
4	0.0701	0.0320	0.0802	0.0295
5	0.0884	0.0365	0.0896	0.0320
6	0.0677	0.0370	0.0688	0.0339
7	0.0675	0.0376	0.0705	0.0452
8	0.0728	0.0269	0.0791	0.0274
9	0.0614	0.0291	0.0702	0.0276
10	0.0799	0.0265	0.0845	0.0274
平均	0.0735	0.0320	0.0799	0.0316

Table 5: 非線形関数 1 に対する最大誤差¹²⁾

Case	三角型		ガウス型	
	SIRMs	NN-SIRMs	SIRMs	NN-SIRMs
1	0.4676	0.1949	0.3382	0.1429
2	0.3001	0.1516	0.3308	0.2219
3	0.2632	0.1295	0.2777	0.1135
4	0.2448	0.1660	0.2991	0.1411
5	0.4666	0.2801	0.3146	0.2198
6	0.6039	0.4339	0.5798	0.3707
7	0.5556	0.4404	0.5606	0.4683
8	0.4082	0.2122	0.4465	0.1985
9	0.4214	0.3859	0.5387	0.3534
10	0.3954	0.1839	0.3579	0.1804
平均	0.4127	0.2578	0.4044	0.2410

Table 6: 非線形関数 2 に対する評価誤差¹²⁾

Case	三角型		ガウス型	
	SIRMs	NN-SIRMs	SIRMs	NN-SIRMs
1	0.0354	0.0271	0.0302	0.0295
2	0.0357	0.0333	0.0296	0.0280
3	0.0335	0.0279	0.0263	0.0274
4	0.0375	0.0312	0.0299	0.0295
5	0.0354	0.0287	0.0281	0.0279
6	0.0383	0.0329	0.0301	0.0289
7	0.0349	0.0298	0.0277	0.0284
8	0.0348	0.0282	0.0291	0.0272
9	0.0349	0.0300	0.0279	0.0282
10	0.0329	0.0305	0.0276	0.0290
平均	0.0353	0.0300	0.0287	0.0284

Table 7: 非線形関数 2 に対する最大誤差¹²⁾

Case	三角型		ガウス型	
	SIRMs	NN-SIRMs	SIRMs	NN-SIRMs
1	0.1163	0.1163	0.1644	0.1190
2	0.1392	0.1493	0.1437	0.1305
3	0.1156	0.1258	0.1180	0.1144
4	0.1049	0.1124	0.1455	0.1025
5	0.1295	0.1229	0.0776	0.1159
6	0.1510	0.1169	0.0920	0.1218
7	0.1193	0.1069	0.1208	0.1116
8	0.1174	0.1409	0.0908	0.1350
9	0.1453	0.1157	0.1230	0.1098
10	0.1119	0.1472	0.1119	0.1228
平均	0.1250	0.1254	0.1188	0.1183

以上の結果から、両手法に対して関数 1 と関数 2 では関数 2 の方がよいシミュレーション結果を得た。これは両手法が関数 1 のような両入力項目が乗法的に干渉しているシステムに対しては表現能力が弱く、関数 2 のような両入力項目が加法的に干渉しているシステムには表現能力が強いということが関係している。また関数 2 に関しては両手法で理想に近い結果が得られていることに対し、関数 1 に関しては従来の SIRMs 推論法に関しては良い結果が得られなかったが、本推論法では理想値に近い結果が得られた。このことから本推論法は乗法的にも加法的にも精度良く対応することが明らかになった。

以上の結果から、従来法よりも本推論法の方が優位であることが分かる。

4.3 医療診断システムへの適用

本章ではニューラルネットワーク型 SIRMs 推論法の性能を実際に評価するため、医療診断システムに適用し、評価を行った結果を示す¹³⁾。医療データは、医師が実際に診断した 145 個の糖尿病データを用いる²⁸⁾。糖尿病データは 5 入力 1 出力となっており、各入力項目は相対体重、空腹時血糖値、ブドウ糖値、インシュリン値、SSPG 値の 5 つで、範囲をすべて $[0, 1]$ に正規化する。

なお、出力は 3 群に判別し、出力値 0.75 以上を 1 群 (臨床的糖尿)、0.25 以上 0.75 未満を 2 群 (化学的糖尿)、0.25 未満を 3 群 (正常) とする。

初期ルール作成において、糖尿病診断データに基づ

Table 8: 糖尿病データに対する評価誤差¹³⁾

Case	三角型		ガウス型	
	SIRMs	NN-SIRMs	SIRMs	NN-SIRMs
1	0.1180	0.1172	0.1204	0.0970
2	0.1131	0.0985	0.1124	0.0885
3	0.1154	0.0423	0.0975	0.0659
4	0.1254	0.0904	0.1229	0.0757
5	0.1045	0.0746	0.1099	0.0787
6	0.1079	0.0686	0.1134	0.0746
7	0.1155	0.0665	0.1203	0.0963
8	0.1161	0.0583	0.1152	0.0730
9	0.1201	0.0634	0.1150	0.0662
10	0.1179	0.0631	0.1104	0.0607
平均	0.1154	0.0743	0.1137	0.0777

Table 9: 糖尿病データに対する正答率(%)¹³⁾

Case	三角型		ガウス型	
	SIRMs	NN-SIRMs	SIRMs	NN-SIRMs
1	86.3	84.9	90.4	84.9
2	91.8	82.2	93.2	87.7
3	95.9	97.3	95.9	100.0
4	86.3	83.6	87.7	86.3
5	93.2	90.4	93.2	94.5
6	87.7	94.5	91.8	93.2
7	91.8	93.2	90.4	90.4
8	87.7	93.2	90.4	94.5
9	91.8	91.8	94.5	91.8
10	90.4	91.8	94.5	94.5
平均	90.3	90.3	92.2	91.8

いて5入力1出力で各入力項目の台集合を5等分したメンバーシップ関数を用意する。そして、初期の三角型、ガウス型メンバーシップ関数は隣同士が0.5で交差するように、各入力項目の重視度の初期値を0.5に設定する。

5入力1出力の糖尿病診断データ145個を教師データ72個、評価データ73個にランダムで振り分ける。三角型、ガウス型メンバーシップ関数それぞれを用いて推論を行い、3の学習アルゴリズムを元に各種パラメータを逐次調整する。学習係数は試行錯誤の上決定した。学習回数を1000回に設定し、三角型、ガウス型共に10回シミュレーションを行った。結果を表8,9に示す。

表8,9より、本推論法は正答率が90%以上であり、評価誤差に関しては本推論法が従来法よりも良い結果を得られていることから、本システムの同定能力の高さがうかがえる。

4.4 あやめデータの判別分析

4.3では医療診断データに適用し、その精度の検証を行なった。本章ではニューラルネットワーク型SIRMs推論法をあやめ判別分析に適用し、評価した結果を示す¹³⁾。あやめデータは、Irisデータを用いる²⁹⁾。Irisデータは以下のIris Setosa, Iris Versicolor, Iris Virginicaの3クラスに分類されており、推論に用いられる理想出力は下記の範囲で、Iris Setosaを0.1667, Iris Versicolorを0.5, Iris Virginicaを0.8333としている。

$$\left\{ \begin{array}{l} 0.1667 \text{ (Iris Setosa)} : y^0 < \frac{1}{3} \\ 0.5 \text{ (Iris Versicolor)} : \frac{1}{3} \leq y^0 < \frac{2}{3} \\ 0.8333 \text{ (Iris Virginica)} : \frac{2}{3} \leq y^0 \end{array} \right.$$

Table 10: あやめデータに対する評価誤差¹³⁾

Case	三角型		ガウス型	
	SIRMs	NN-SIRMs	SIRMs	NN-SIRMs
1	0.0465	0.0389	0.0584	0.0447
2	0.0334	0.0310	0.0460	0.0359
3	0.0359	0.0290	0.0457	0.0334
4	0.0488	0.0380	0.0529	0.0424
5	0.0488	0.0333	0.0567	0.0455
6	0.0582	0.0325	0.0585	0.0448
7	0.0371	0.0357	0.0534	0.0428
8	0.0360	0.0342	0.0550	0.0427
9	0.0371	0.0384	0.0631	0.0535
10	0.0387	0.0387	0.0600	0.0513
平均	0.0420	0.0350	0.0550	0.0437

Table 11: あやめデータに対する正答率(%)¹³⁾

Case	三角型		ガウス型	
	SIRMs	NN-SIRMs	SIRMs	NN-SIRMs
1	94.7	94.7	96.0	93.3
2	96.0	96.0	97.3	96.0
3	100.0	100.0	100.0	100.0
4	97.3	97.3	98.7	97.3
5	98.7	97.3	98.7	98.7
6	98.7	97.3	98.7	98.7
7	97.3	100.0	98.7	97.3
8	96.0	96.0	97.3	97.3
9	93.3	93.3	93.3	92.0
10	93.3	93.3	94.7	96.0
平均	96.5	96.5	97.3	96.7

また、1クラス50個のサンプルを持つ4次元データであり、範囲をすべて[0, 1]に正規化する。

初期ルール作成において、糖尿病診断データに基づいて4入力1出力で各入力項目の台集合を5等分したメンバーシップ関数を用意する。そして、初期の三角型、ガウス型メンバーシップ関数は隣同士が0.5で交差するように、各入力項目の重視度の初期値を0.5に設定する。

三角型、ガウス型メンバーシップ関数それぞれを用いて推論を行い、3の学習アルゴリズムを元に各種パラメータを逐次調整する。学習係数は試行錯誤の上決定した。学習回数を1000回に設定し、三角型、ガウス型共に10回シミュレーションを行った。結果を表10,11に示す。

表10,11より、本推論法の正答率は従来の推論法と大きな差は見られなかったが、90%以上と高い正答率を得ており、評価誤差に関しては医療データと同様に、本推論法が従来法よりも良い結果を得られていることから、本推論法が優位であることが分かる。

5 モジュラー型ファジィ推論モデルとニューラルネットワークの融合

2ではニューラルネットワークとSIRMs推論モデルの融合として、ニューラルネットワーク型SIRMs推論モデルとモジュラー型ファジィ推論モデルを紹介した。SIRMs推論モデルとモジュラー型ファジィ推論モデルの対応から、ニューラルネットワーク型SIRMs推論モデルの一般化として、ニューラルネットワーク型モジュラーファジィ推論モデルを提案することができる。すなわち、本推論モデルのモデルは図3のように与えられる。

6 むすび

本稿では、ニューラルネットワークとモジュラー型ルールを持つファジィ推論モデルの融合したモデルと

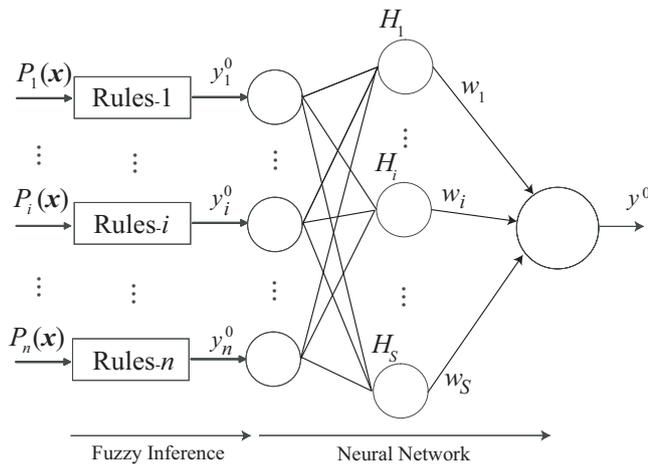


Fig. 3: ニューラルネットワーク型モジュラーファジィモデルの構造

して、ニューラルネットワーク型 SIRMs 推論モデルを紹介した。本モデルは様々な非線形関数の同定、医療診断、あやめ判別分析に応用され、すべての問題に対して良好な結果を得ることができている。

また、最後に本モデルの一般化モデルの一つとしてニューラルネットワーク型モジュラーファジィモデルを提案した。今後の課題としては、ニューラルネットワーク型モジュラーファジィモデルを実システムへ応用することにより、その精度を検証し、その性質を解明することや最適なパラメータ設定の考案などが挙げられる。

参考文献

- 1) E. H. Mamdani: Application of fuzzy algorithms for control of simple dynamic plant, Proc. IEE, **121** 1585/1588 (1974).
- 2) T. Takagi and M. Sugeno: Fuzzy identification of systems and its Applications to modeling and control, IEEE Trans. Syst., Man, Cybern., **SMC-15-1**, 116/132 (1985).
- 3) N. Yubazaki, J. Yi and K. Hirota: SIRMs (Single Input Rule Modules) connected fuzzy inference model, Journal of Advanced Computational Intelligence and Intelligent Informatics, **1**, 23/30(1997).
- 4) J. Yi, N. Yubazaki, and K. Hirota: Stabilization control of seriestype double inverted pendulum systems using the SIRMs dynamically connected fuzzy inference model, Artificial Intelligence in Engineering, **15**, 297/308 (2001).
- 5) J. Yi, N. Yubazaki, and K. Hirota: Upswing and stabilization control of inverted pendulum system based on the SIRMs dynamically connected fuzzy inference model, Fuzzy Sets and Systems, **122** 139/152 (2001).
- 6) J. Yi, N. Yubazaki, and K. Hirota: A new fuzzy controller for stabilization of parallel-type double inverted pendulum system, Fuzzy Sets and Systems, **126**, 105/119 (2002).
- 7) J. Yi, N. Yubazaki, and K. Hirota: A proposal of SIRMs dynamically connected fuzzy inference model for plural input fuzzy control, Fuzzy Sets and Systems, **125** 79/92 (2002).
- 8) J. Yi, N. Yubazaki, and K. Hirota: Anti-swing and positioning control of overhead traveling crane, Inf. Sci. **155** 19/42 (2002).
- 9) H. Seki and M. Mizumoto: On the equivalence of fuzzy inference methods—part 1: basic concept and definition, IEEE Trans. Fuzzy Systems, **19-6**, 1097/1106 (2011).
- 10) H. Seki, H. Ishii and M. Mizumoto: On the generalization of single input rule modules connected type fuzzy reasoning method, IEEE Trans. Fuzzy Systems, **16-5**, 1180/1187 (2008).
- 11) H. Seki, H. Ishii and M. Mizumoto: On the monotonicity of fuzzy-inference methods related to T-S inference method, IEEE Trans. Fuzzy Systems, **18-3**, 629/634 (2010).
- 12) H. Seki, S. Watanabe, H. Ishii, and M. Mizumoto: Realization of XOR by SIRMs connected fuzzy inference method, Lecture Notes in Artificial Intelligence, **5579**, 252/261, Springer-Verlag (2009).
- 13) S. Watanabe, H. Seki, and H. Ishii: Discriminant analysis by neural network-type SIRMs connected fuzzy inference method, Proc. 2010 IEEE International Conference on Industrial Informatics, 93/97 (2010).
- 14) T. Watanabe and Y. Takahashi: Hierarchical reinforcement learning using a modular fuzzy model for multi-agent problem, Proc. 2007 IEEE Int. Conf. Syst., Man, Cybern., 1681/1686 (2007).
- 15) T. Watanabe and R. Fujioka: Collaborative filtering based on modular fuzzy model concept, Proc. Joint 4th Int. Conf. on Soft Computing and Intelligent Systems and 9th Int. Symp. on advanced Intelligent Systems (SCIS&ISIS 2008), 135/1354 (2008).
- 16) T. Watanabe and T. Wada: Reinforcement learning based on modular fuzzy model with gating unit, Proc. 2008 IEEE Int. Conf. Syst., Man, Cybern., 1806/1811 (2008).
- 17) T. Watanabe and H. Seki: A study on modeling approach based on modular fuzzy model, Proc. The 1st International Workshop on Aware Computing (IWAC09), 743/748 (2009).
- 18) J. R. Jang: ANFIS: adaptive-network-based fuzzy inference system, IEEE Trans. Syst., Man Cybern., **23**, 665/685 (1993).
- 19) F. L. Chung and J. C. Duan: On multistage fuzzy neural network modeling, IEEE Trans. Fuzzy Syst., **8-2**, 125/142 (2000).
- 20) M. Y. Chen and D. A. Linkens: A systematic neuro-fuzzy modeling framework with application to material property prediction, IEEE Trans. Syst., Man, Cybern., **31-5**, 781/790 (2001).
- 21) J. S. Wang and C. S. George Lee: Self-adaptive neuro-fuzzy inference systems for classification applications, IEEE Trans. Fuzzy Syst., **10-6**, 790/802 (2002).
- 22) H. Iyatomi and M. Hagiwara: Adaptive fuzzy inference neural network, Pattern Recognition, **37-10**, 2049/2057 (2004).
- 23) H. Ichihashi: Iterative fuzzy modeling and a hierarchical network, Proc. 4th IFSA Congress of Engineering, 49/52 (1991).
- 24) L. X. Wang and J. M. Mendel: Back-Propagation fuzzy system as nonlinear dynamic system identifiers, Proc. 1992 IEEE Int. Conf. Fuzzy Syst., 1409/1416 (1992).
- 25) X. Cui and K. G. Shin: Direct control and coordination using neural networks, IEEE Trans. Syst., Man, Cybern., **23-3**, 686/697 (1993).
- 26) C. F. Juang: A TSK type recurrent fuzzy network for dynamic systems processing by neural network and genetic algorithms, IEEE Trans. Fuzzy Syst., **10-2**, 155/170 (2002).
- 27) W. Yu and X. Li, "Fuzzy identification using fuzzy neural networks with stable learning algorithms, IEEE Trans. Fuzzy Syst., **12-3**, 411/420 (2004).
- 28) D. F. Andrews and A. M. Herzber: Data: A collection of problems from many fields for the students and research worker, Springer (1985).
- 29) D. J. Newman, S. Hettich, C. L. Blake, and C. J. Marz: UCI repository of machine learning databases," University of California, Irvine (1998). <http://www.ics.uci.edu/~mllearn/MLRepository.html>

An Analysis on Evolutionary Ensembles of Neural Networks: a Motivation behind Low Dimensional Networks

*Danilo Vasconcellos Vargas, Hirotaka Takano and Junichi Murata (Kyushu University)

Abstract— The accuracy of ensembles have been shown to depend on the base classifier’s accuracy and diversity. However, choosing good base classifiers to compose an ensemble is still a difficult task. The base classifier’s accuracy-diversity trade-off composes a complex space to search. Moreover, searching for all possible classifiers is naturally unfeasible and some assumptions have to be taken. In this context, we present a motivation behind low dimensional classifiers. The experiments are conducted on varying levels of difficulty (time constraints and addition of irrelevant variables) and different datasets. Based on these experiments, it is verified that an evolutionary ensemble made of a certain portion of low dimensional neural networks can be more robust and accurate than high dimensional ones. Throughout the experiments the influences of diversity, base classifier’s accuracy as well as the effects of the ensemble and evolution are analyzed and empirically evaluated. Thus, the use of low dimensional base classifiers is justified as a reasonable assumption for the construction of ensembles, which enables ensembles to achieve the maximum observed accuracy as well as surpass high dimensional ones in robustness and learning speed.

Key Words: Network Evolution, Ensembles, Homogeneous Ensembles, Classification, Multilayer Perceptron.

1 Introduction

The research on ensembles of classifiers has been motivated by the idea that weak learners can produce a strong learner (1), (2), (3). But how can we choose weak learners to create a better ensemble? Are some weak learners better than others?

In the literature the effect of joining the classifiers by a majority vote was shown to relate strongly to the accuracy of the base classifiers and diversity (4), (5), (6). Therefore, using a multiobjective search for classifiers with both high accuracy and diversity was justified. But using both the base classifier’s accuracy and diversity to choose or search for classifiers has turned out to be non trivial because there is some complex trade-off (constraint) between accuracy and diversity (7). Moreover, searching through all possible settings of classifiers is naturally unfeasible, therefore some restriction is imposed over the search in some form, such as choosing a particular classifier type or limiting the number of dimensions¹.

This paper presents an answer to these questions. In the context of homogeneous evolutionary ensembles, we show that some “low dimensional weak learners” can produce more robust and accurate results than other weak learners. The “low dimensional weak learners” are classifiers made of a low number of dimensions and with a relatively low accuracy when evaluated alone, but present high accuracy and low variation of accuracy when joined in an ensemble. This happens even when the number of maximum iterations in the learning process are steeply decreased (say, from 1000 to 20 iterations) or a large number of irrelevant variables are added to the problem (for example, twenty irrelevant variables added to four relevant variables).

In summary, these ensembles of low dimensional learners have important advantages when contrasted with high dimensional ones. We highlight the following:

- Fast Learning Process - When compared to ensembles of high dimensional networks, their

search space is smaller, allowing them to reach good solutions in a small amount of time;

- Robustness against Irrelevant Variables - They can overcome the existence of irrelevant variables better than higher dimensional learners;
- Robustness against Number of Learning Iterations - Since they are less affected by overfitting and underfitting, decreasing the number of learning iterations has only minor effects;
- Flexibility - They keep the same behavior over a set of different datasets;
- Accuracy - When compared to high dimensional models, they have equivalent or better accuracy.

To complement the study, an analysis is drawn based on the empirical tests which summarizes and explain the results. Lastly, the effect of the evolution and ensemble are evaluated separately, explaining their individual influences over the experiments.

2 Evolutionary Ensemble of Multilayer Perceptrons

This paper uses for the analysis an evolutionary ensemble of multilayer perceptrons (8) with a majority voting mechanism. The reason for the evolutionary choice instead of more typical ensemble settings is that the evolutionary ensemble is less sensitive and therefore presents also less variation in accuracy (for details see Section 5). Actually, the simple evolutionary ensemble used here is similar to removing the less accurate base classifiers. Therefore, the analysis conveyed here can be extended to the typical ensemble settings with minor modifications (see Section 5).

Having said that, the evolutionary process is described in detail in Table 1 which consists of the simple selection of a number of classifiers, hereby called “number of predated” (*pred*), to be excluded for each evolutionary cycle (iteration). For the selection process, the accuracy of the single neural networks are compared on an unseen 20% of the training samples, using the remaining 80% of the training samples for the actual training. And the excluded classifiers (“predated classifiers”) are the ones which achieved

¹Dimensions will be used throughout this article to refer to the number of variables used by a classifier

Table 1: Evolutionary Ensemble Construction Algorithm

1. Generate an initial ensemble with $nmod$ classifiers each having m randomly chosen variables.
2. Separate the training dataset into two parts: one with 20% and the other with 80% of the samples.
3. For each evolutionary cycle until the max_{cycles} is reached do the following:
 - (a) Train all the classifiers using 80% of the training samples.
 - (b) Evaluate the classifiers using the unseen 20% of the training samples.
 - (c) Select the $pred$ least accurate classifiers and remove them from the ensemble.
 - (d) Add to the ensemble new $pred$ classifiers each with m randomly chosen variables.
4. Final ensemble.

Table 2: Parameters

	Multilayer Perceptron
Initial weights	$[-0.1, 0.1]$
Weight decay	$5e - 4$
Hidden nodes	5
Maximum iterations	1000
Output units	logistic function
	Evolutionary Ensemble
Number of models ($nmod$)	10
Evolutionary cycles (max_{cycles})	20
Number of predated ($pred$)	1

the worst accuracy on the unseen 20% of the training dataset. The same samples were used to train all neural networks, therefore no increase in diversity is caused by the evolution. At the same time, the accuracy measured over the 20% of the training samples is not biased by the samples (they do not vary for all classifiers) and therefore the selection is also not dependent on the samples.

3 Experiments

3.1 Settings

All experiments, when not stated otherwise, use evolutionary ensembles as described previously. The ensembles have a fixed number of neural networks and all the neural networks have the same number of variables m ($< M$, where M is the total number of inputs) but the inputs themselves may differ from network to network (they are chosen randomly). The tests will be conducted by varying the number of dimensions m (inputs) and every test will be repeated over 100 trials with different 80/20 train/test splits. Moreover, throughout the tests the parameters are kept mostly the same, they are described in Table 2. For the implementation we used the `nnet` package from R (9) which is a feed-forward neural network having a single hidden layer and back-propagation as learning algorithm.

The experiments were conducted over the following datasets from the UCI machine learning repository (10): Glass, Wine, Iris, Ionosphere and ‘‘Pima Indians Diabetes’’ (which we will refer to as ‘‘Diabetes’’). All datasets were divided into 80% for training and 20% for testing. Therefore, taking the evolutionary selection into account, which also separates the train-

ing dataset into a portion to evaluate the networks and another portion to train them. Each single neural network was trained in an effective 64% of the total samples.

3.2 Relation of the Accuracy of Ensembles with the Dimension

Here, the objective is to identify the behavior of the ensemble accuracy when we vary the dimension used by the single neural networks (‘‘weak learners’’) as well as to make a comparison between the accuracies of the ensemble and the single networks. Figure 1 show the results for some datasets, where the first and the second rows contrast the accuracy of the averaged individual neural networks with that of the ensemble.

All the datasets observe the same effect, the individual neural networks accuracy increases slowly over the increase of dimensions while the ensemble accuracy increases steeply. Therefore, the use of an ensemble reduces the importance of the number of dimensions. Note that the dimensions used in the ensemble is potentially the same as the original dataset. However, the number of dimensions required for searching for a good hypothesis (learning process) for each low dimension classifier is smaller and therefore the combination of low dimension classifiers present in the ensemble make the number of dimensions in the learning process relatively less important. Moreover, the accuracy decreases on the high dimension portion (specially on the Wine and Ionosphere dataset). This effect is directly related with the diversity of the ensembles as we will show in the next section.

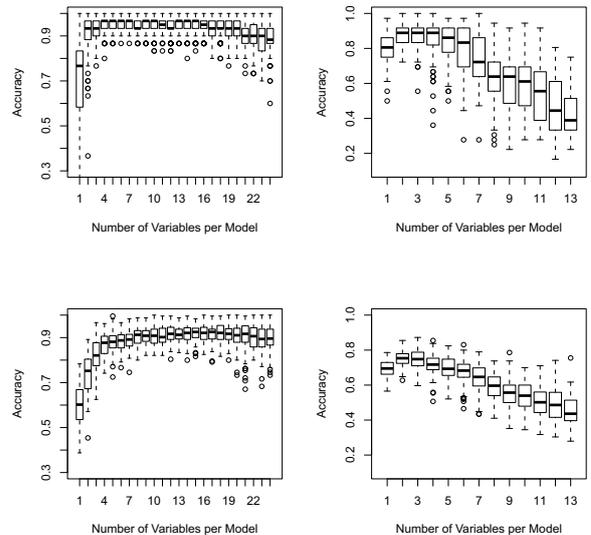


Fig. 2: Behavior over the Iris (with additional 20 irrelevant variables) and Wine (using single neural networks with only 20 iterations and 3 hidden nodes) datasets (respectively the first and second column). The first row shows the ensemble accuracy and the second shows the average accuracy of the same individual neural networks evaluated alone over the testing dataset.

Although Figure 1 brings some overall behavior, it still lack some insight of how the accuracy would behave under irrelevant attributes and time constraints (low number of iterations). That is, in real world problems not rarely the classifiers face a high number of irrelevant attributes and/or have to give reasonably good solution under tight time constraints such as real

time problems. To test these situations, in Figure 2 we increase the difficulty of the datasets by adding 20 irrelevant variables in the Iris dataset and limiting the number of iterations to 20 and the number of hidden nodes to three for the Wine dataset.

Figure 2 shows the behavior of a certain low dimensional portion agreeing with Figure 1 in that some low dimensional ensembles are the most robust and accurate. The reason why the high dimensional part performs poorly can be justified based on underfitting. Still, the behavior of the low dimensional ensembles with as low as two variables is unexpected. For example, ensembles made of two or three variables can achieve around 93% accuracy in the Iris with 20 additional irrelevant variables while in the Wine dataset they obtained the maximum accuracy observed.

Since the unique factor which changes from the first row to the second row of all the figures is the ensemble effect of adding all networks in a voting ensemble, the explanation for this behavior can only be linked to this effect. As for the evolution, it was applied for both the first and the second row, being at first glance not relevant for the behavior that makes the low dimensional networks a promising alternative. But before analyzing this effect further it is important to consider the influence of diversity.

3.3 The Influence of Diversity

Diversity and accuracy has been regarded as the most important characteristics to be considered when choosing classifiers to compose an ensemble. Therefore, we analyze in this section the behavior of diversity with relation to the accuracy.

There are a vast number of measures of diversity in the literature (11), (12). From those, this work will use Shannon’s diversity index D_{sha} (13) (exactly the same as the entropy). That is, given a set of classifiers Pop with a set of classes Cl , any sample can be classified into different classes by different individual classifiers in Pop . Let us denote the number of classifications of a sample into class i by c_i . Let us define the fraction of classifications as $p_i = \frac{c_i}{|Pop|}$. Then it follows that:

$$D_{sha} = \sum_{i=1}^{|Cl|} p_i \ln(p_i). \quad (1)$$

Shannon’s diversity was computed for every sample and then the diversities were summed over all samples and divided by the number of samples.

Using the above diversity measure, Figure 3 was drawn over the Diabetes and Wine datasets. From Figure 3, it can be seen that diversity decreases with the increase in the number of variables used. This was expected, since the number of possible combinations of variables for a fixed dimensionality increases as the dimensionality of a neural network decreases. Moreover, the final portion of the diversity in both figures differ. In the Diabetes dataset the diversity decreases to a minimum amount and stays constant, however, in the Wine dataset the diversity continues decreasing towards zero with the increase in the number of variables. This explains the decrease in the ensemble accuracy for the Wine dataset in the highest dimensional ensembles.

The diversity of the lower dimensional classifiers are one of the reasons for their advantages. This property is, however, inherent to their inferior accuracy and lower number of dimensions. It should be easy to see that higher dimensional classifiers will have less diversity, but sometimes it is forgotten that the diversity

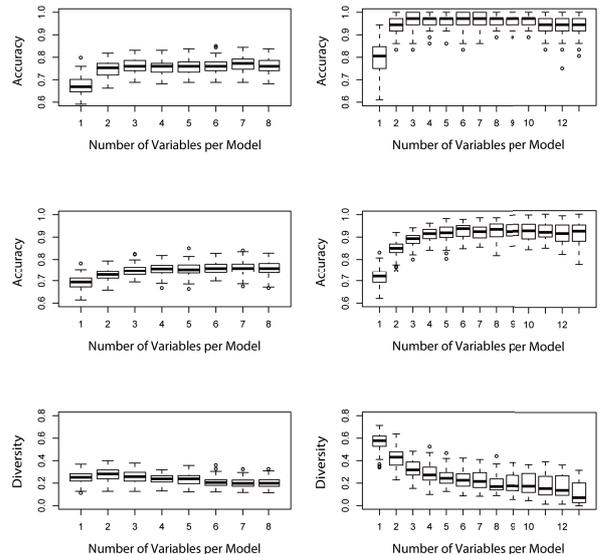


Fig. 3: Diversity behavior over the Diabetes and Wine datasets (respectively the first and second column) and its relation with accuracy and number of variables per model. The first row shows the ensemble accuracy, the second shows the average accuracy of the same individual neural networks evaluated alone over the testing dataset and the third shows the diversity.

is closely related to the accuracy and high accurate models will have similar output, differing minimally in the diversity. Actually, this fact makes the lower accurate individuals more prone to receiving a higher benefit from using them in an ensemble.

4 Empirical Analysis of the Ensemble’s Accuracy

Throughout the experiments it was observed that in the accuracy curve the number of variables per model can be divided into three parts: A, B and C. Considering the problem dependent decrease in accuracy of part C, the empirical diagram shown on Figure 4 can be drawn.

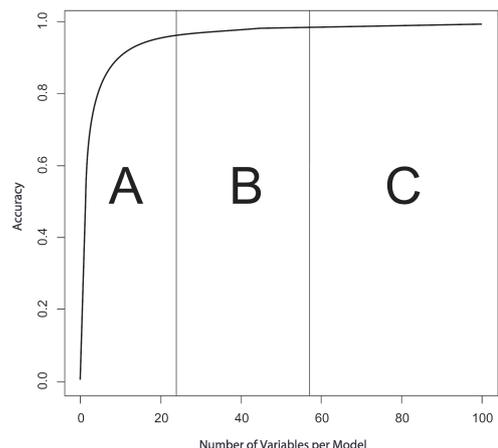


Fig. 4: Empirical diagram showing the three conceptual divisions.

With the exception of Part B, all of the parts have some sort of disadvantage. In one hand Part A has the following problem:

- Few Information Available - The information available through their few inputs are not suf-

Table 3: Change of Evolutionary’s Parameters

Evolutionary Ensemble	
Number of models	10
Evolutionary cycles	20
Number of predated	1
Modified Evolutionary Ensemble	
Number of models	20
Evolutionary cycles	40
Number of predated	4

ficient for the classifier to learn.

On the other hand, Part C has:

- More Susceptible to Overfitting;
- More Susceptible to Underfitting;
- Slower Learning Process - Their higher complexity (high number of weights) needs more iterations to arrive at the same level of accuracy as that of lower complexity ones;
- Require High Classifier Complexity.

Therefore, part B benefits from a suitable complexity which enables them to learn the problem at hand when joined in an ensemble at the same time their relative simplicity allows for a more robust classifier (overcoming overfitting/underfitting and irrelevant variables). Moreover, their lower complexity speeds up the learning process.

5 Evaluating Evolutionary and Ensemble Effects

Two effects take place in the analyzed evolutionary ensemble:

- Ensemble Effect - The effect of adding single networks together by a voting mechanism.
- Evolutionary Effect - The influence of evolving networks together. In this context, the evolution process is described by generations of evaluating and predated some of the less fitted networks.²

The question we want to study here is what role does the removal of some of the lowest accurate classifiers plays on the ensemble. Previously, the figures were divided between the accuracy of the ensemble and the accuracy of the individual classifiers. Therefore it was possible to note the ensemble effect. This time, we separate the figures between the accuracy of the ensemble with and without applying evolution. In this way, the effect of evolution and ensemble are shown separately.

In these tests, the ensemble setting without evolution possess the same number of trained neural networks used throughout the evolutionary method which is 30 (number of models + number of predated * evolutionary cycles). In other words, the ensemble without evolution use 30 neural networks in the majority voting mechanism while the evolutionary ensemble uses 30 neural networks to evolve itself, however, the final ensemble is composed of only 10 neural networks.

It can be observed in Figure 5 that by adding evolution to the ensemble scenario it may decrease the variation in the accuracy while increasing the accuracy in some portions of the spectrum, i.e., in comparison

²Naturally, the simple evolutionary effect spotted here is somewhat similar to just removing the less accurate base classifiers as mentioned previously.

with evolutionary ensemble some of the boxplots without evolution had a high variation of the accuracy and presented behaviors which differ slightly from dataset to dataset. Although the differences are small and the overall behavior is preserved, the evolution effect still plays a role in decreasing the sensitivity of the number of dimensions used and consequently from dataset to dataset.

The reason behind the variation in accuracy of the ensemble without evolution can be spotted in Figure 6. As expected, the diversity of the typical ensemble is higher than the evolutionary ensemble, because the evolutionary ensemble decreases the diversity when it removes the lower accurate classifiers. In these experiments, it is interesting to note the relation between the diversity over the Wine dataset (lower right of Figure 6) and its accuracy (lower left of Figure 5) for the typical ensemble. In the Wine dataset the diversity remains at a very high value which seems to be connected with the unusual high accuracy variation and many outliers at the high dimensional portion of the spectrum.

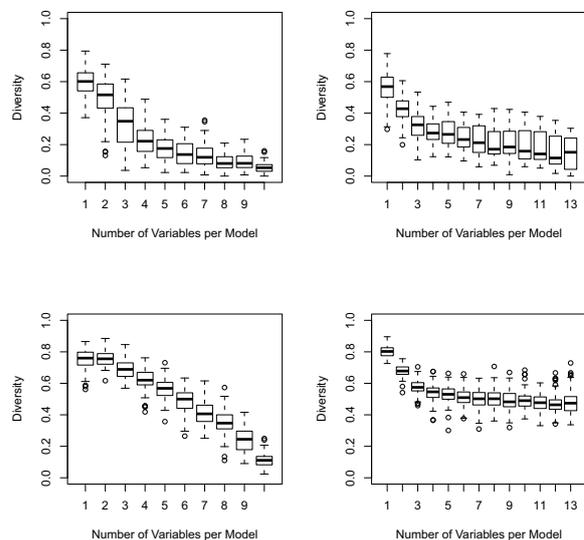


Fig. 6: Effects of evolution and ensemble on the diversity. Tests made over the Glass (left) and Wine (right) Datasets with the upper row having the original results from the evolutionary ensemble and the lower row having a typical ensemble settings without evolution. The number of neural networks trained by both methods are the same and equal to 30.

6 Conclusions

This paper concerns with the question of how to choose “weak learners” to compose an ensemble. To answer this question, experiments were conducted on different datasets and varying levels of difficulty (time constraints and addition of irrelevant variables). From these experiments we observed:

- Regarding Base Classifier’s Accuracy - By analyzing the behavior of the base classifier’s accuracy, the high dimensional ones was seen to possess the highest accuracy as a base classifier. But when analyzed as an ensemble, a part of the low dimensional ensembles showed more robustness and at least the same accuracy as the high dimensional ones (even surpassing them in some scenarios).

- Regarding Diversity - A series of tests showed that the diversity increased with the decrease in the base classifiers' dimensions. Therefore, it was related to the decrease in accuracy for the high dimensional ensembles. Moreover, recall that the diversity is closely related to the accuracy and high accurate models will have similar output, differing minimally in the diversity. In other words, the smaller the accuracy the higher the gain of using an ensemble.
- Regarding Evolution and Ensemble Effects - The evolution and ensemble effects were analyzed separately. Evolution was seen to relate strongly with the decrease in the ensemble's accuracy variation, while the diversity added by the low accurate base classifiers (not present in the evolutionary ensemble since they were predated) in the typical ensemble increased its sensibility while not conferring benefits in accuracy.

Thus, it was shown that if the number of dimensions of an ensemble is varied, a certain portion of the low dimensional neural networks appears as the most promising one. This knowledge can be used to aid the construction of ensembles. In other words, it can shrink the search space of the possible classifiers to a smaller search space of promising low dimensional classifiers. Therefore, we conclude that low dimensional base classifiers are better weak learners, in the sense that ensembles made of them benefit from higher robustness, accuracy and learning speed when compared to ones using high dimensional base classifiers.

References

- [1] R.E. Schapire. The strength of weak learnability. *Machine learning*, 5(2):197/227 (1990).
- [2] R.E. Schapire. The boosting approach to machine learning: An overview. *Lecture Notes in Statistics*, pages 149/172 (2003).
- [3] L. Breiman. Bagging predictors. *Machine learning*, 24(2):123/140 (1996).
- [4] T.J. Hastie and R.J. Tibshirani. *Generalized additive models*. Chapman Hall/CRC,(1990).
- [5] G. Valentini and F. Masulli. Ensembles of learning machines. *Neural Nets*, pages 3/20 (2002).
- [6] K. Tumer and J. Ghosh. Error correlation and error reduction in ensemble classifiers. *Connection Science*, 8(3-4):385/404, (1996).
- [7] A. Chandra, H. Chen, and X. Yao. Trade-off between diversity and accuracy in ensemble generation. *Multi- objective machine learning*, pages 429/464, (2006).
- [8] S.S. Haykin. *Neural networks and learning machines*, volume 3. Prentice Hall, (2009).
- [9] W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S*. Springer, New York, fourth edition, (2002). ISBN 0-387-95457-0.
- [10] A. Frank and A. Asuncion. *UCI machine learning repository*, (2010).
- [11] S.T. Hadjitodorov, L.I. Kuncheva, and L.P. Todorova. Moderate diversity for better cluster ensembles. *Information Fusion*, 7(3):264/275, (2006).
- [12] L.I. Kuncheva and C.J. Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine learning*, 51(2):181/207, (2003).
- [13] C.E. Shannon. The mathematical theory of communication. 1963. *MD computing computers in medical practice*, 14(4):306/317, (1948).

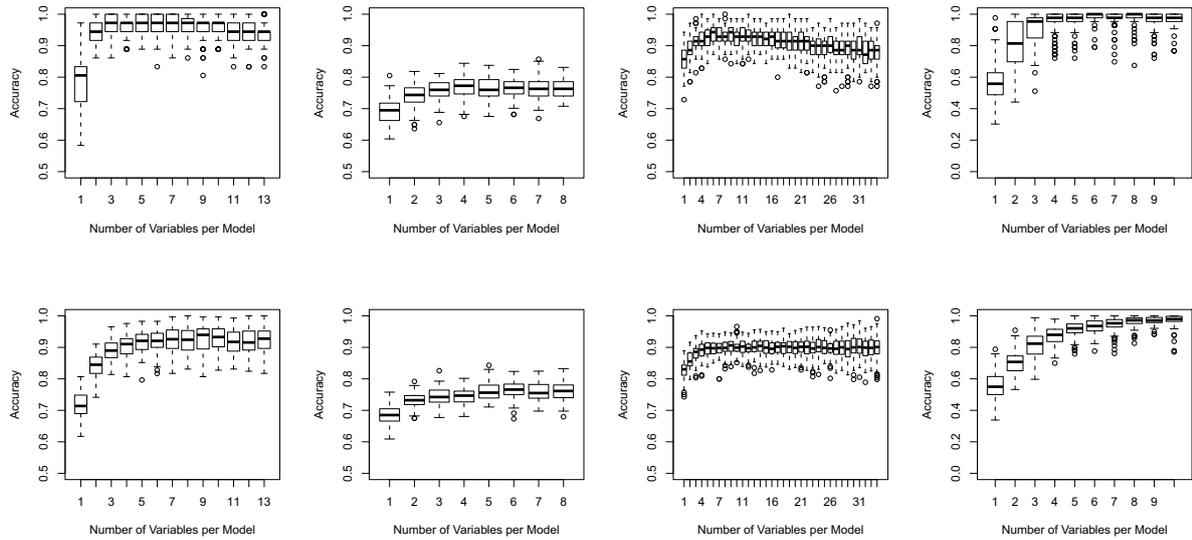


Fig. 1: Behavior over the Wine, Diabetes, Ionosphere and Glass datasets (respectively the first, second, third and fourth column). The first row shows the ensemble accuracy and the second shows the average accuracy of the same individual neural networks evaluated alone over the testing dataset.

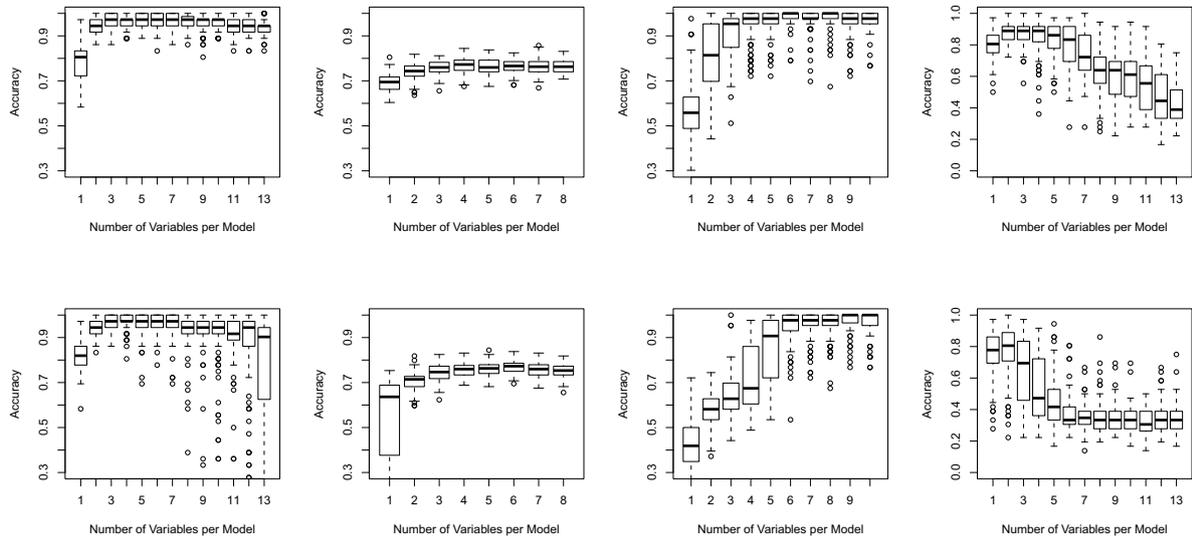


Fig. 5: Effects of evolution and ensemble on the accuracy. Tests made over the (from the left to the right) Wine, Diabetes, Glass and a modified Wine (using single neural networks) Datasets with the upper row having the original results from the evolutionary ensemble and the lower row having a typical ensemble settings without evolution. The number of neural networks trained by both methods are the same and equal to 30.

ニューロ様断熱的量子計算のための量子ビットデバイスの検討

○金城光永[†] 比嘉孝昌^{††} 諸見里貴徳^{††} 島袋勝彦[†]

[†] (琉球大学)

^{††} (琉球大学大学院理工学研究科)

A Study on Quantum Bit Devices for the Neuromorphic Adiabatic Quantum Computation

*M. Kinjo[†], T. Higa^{††}, T. Moromizato^{††}, and K. Shimabukuro[†]

[†] (University of the Ryukyus)

^{††} (Graduate School of Engineering and Science, University of the Ryukyus)

Abstract— The adiabatic quantum computation (AQC) and the neuromorphic AQC (NAQC) can be applied to the prime factorization problem, database search problem, and various combinatorial optimization problems. Especially, the NAQC, the AQC with efficient energy dissipation and design method for a final Hamiltonian in consideration of the analogy with an artificial neural network, can be applied to optimization problems if its cost function can be expressed in a quadratic form. However, there are few reports that have successful operations on real quantum bit devices. In this paper, we study possibilities of executing the NAQC on two real quantum bit devices based on Si and Diamond.

Key Words: Quantum bit, Adiabatic quantum computation, Neural network, Nuclear Spin, N- center

1 はじめに

量子コンピュータは、量子力学的重ね合わせを用いた量子ビット (qubit) が複数の状態を持つことができる性質を利用して、超並列計算による高速情報処理の実現が期待されている^{1, 2)}。具体的な量子計算アルゴリズムの例として、特定用途向けの Shor の素因数分解アルゴリズム³⁾ および Grover のデータベース検索アルゴリズム⁴⁾、また、特定用途向けではない量子計算アルゴリズムとして Adiabatic Quantum Computation (AQC)⁵⁾ および Neuromorphic AQC (NAQC)^{6, 7)}、などが挙げられる。

AQC は Hamiltonian を断熱変化させることにより、組み合わせ最適化問題を解く量子計算アルゴリズムである。その際、最終 Hamiltonian はその基底状態が対象問題の解となるように構成する必要がある。したがって、断熱定理⁸⁾により、断熱的变化中の量子状態が常に初期状態 (通常は基底状態) 近辺に停滞していることで最適解を得ることができる。ただしその際、エネルギー散逸が無く、かつエネルギー準位の交差が無いという制約が満たされている必要がある。この AQC をより汎用性の高い計算手法とするために、微小なエネルギー散逸の導入と qubit の相互作用を基にした Hamiltonian の構成法を取り入れた、ニューロ様量子計算手法 (NAQC) が提案された。これは、qubit とその相互作用を人工ニューラルネットワーク (ANN) のニューロンとそのシナプス結合と見立て、シナプス結合の興奮性と抑制性を利用した量子計算アルゴリズムである。これにより、ANN と同様な応用を期待できるが、その具体的なデバイス実現方法が課題となっている。

一方、量子効果デバイスとして Si ベース、超伝導体ベース、またはダイヤモンドベースなど様々なデバイスが提案されている。特に、室温で量子効果を観測可能なダイヤモンド中の単一炭素-空孔複合中心 (N-V 中心) が、次のデバイスの有力候補として注目されている。その他にイオントラップを利用した手法や、特定

デバイスの集合体をまとめて取り扱う NMR を利用した手法等が挙げられる。実際に、量子アルゴリズムの実行例として、Shor のアルゴリズムの実行が多数の核スピン量子ビットと NMR を利用して行われた研究報告がある。なお、比較的低温下の報告例がほとんどである。

そこで本研究では、現在提案されている Si 系の P 原子核スピンデバイスやダイヤモンド系の N-V 中心を用いたデバイスを考慮して、組み合わせ最適化問題を高速で解くことを目指し、Hamiltonian の構成や解析、制御可能な空間的配置の考察などを行うことによって、NAQC の実行可能性を考察する。

2 NAQC

NAQC においてその基本となる AQC とは、時間に依存する Schrödinger 方程式における Hamiltonian の時間変化、

$$H(t) = \left(1 - \frac{t}{T}\right) H_I + \frac{t}{T} H_F \quad (1)$$

を活用した演算法である。ここで、 T は実行時間、 H_I および H_F はそれぞれ初期および最終 Hamiltonian を示し、断熱定理に沿うよう十分な大きな T を選択する必要がある。また、最終 Hamiltonian は対象問題に応じて構成 (すなわち qubit 数もこの時に考慮) され、時間 $t = T$ の段階で量子状態を観測することにより、対象問題の解を得る。その際、Hamiltonian の時間変化中に、エネルギーの交差や縮退が存在しないことが条件となる。

一方 NAQC は、AQC の断熱定理による制約を改善するため、極微少のエネルギー散逸を取り入れ、かつ人間の脳を工学的に模倣した ANN に基づく Hamiltonian 構成法を用いた演算法である。その構成法の核となる ANN は、神経回路を構成する素子であるニューロン同士をつなぐシナプス結合に興奮性と抑制性を持ち、そ

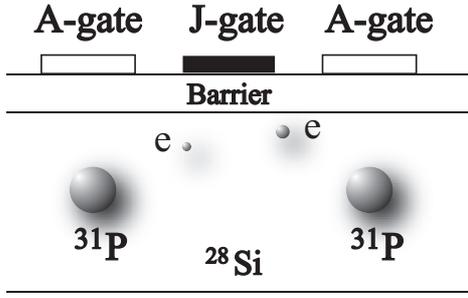


Fig. 1: A qubit-device structure with two ^{31}P nuclear spins in ^{28}Si by proposed Kane.

これらの複雑な組み合わせにより信号処理を行う。なお、興奮性の結合は結合先のニューロンの膜電位を上昇させ、逆に抑制性の結合はこの現象の発生を抑制する。このシナプス結合の特徴を、qubit の相互作用間に活用することで、qubit で構成するネットワークを、ANN と同種の応用に利用できる可能性があり、汎用性のさらなる向上が期待できると考えられる。

3 NAQC を実行可能な候補デバイス

ニューロンをスピン系の qubit に置き換えて考えるとき、スピンの状態 (Z 軸に沿って上向きを $|0\rangle$ 、下向きを $|1\rangle$) をニューロンの応答 (それぞれ非発火および発火) に対応させる。この際、2 つの qubit が $|00\rangle$ ($= |0\rangle|0\rangle$) または $|11\rangle$ の状態のとき、興奮性の結合を持つ 2 つのニューロン同士の場合の応答に対応する。逆に $|01\rangle$ または $|10\rangle$ の状態が抑制性の結合を持つ場合に相当する。

これらの状態を基底状態として持つように qubit 間の相互作用をうまく制御できることが、NAQC を実行可能な qubit デバイスの必要条件となる。

3.1 ^{28}Si 中の ^{31}P 原子

Kane により提案された qubit デバイス (Fig. 1 参照) の Hamiltonian は、 ^{31}P 原子核スピンを情報の担い手とし、結合に過剰な電子スピンひとつも含めた 1-qubit をふたつ考慮した場合、

$$H = H_1 + H_2 + J_{12}(\sigma^{e_1} \cdot \sigma^{e_2}) + A_{12}(\sigma^{n_1} \cdot \sigma^{e_2}) + A_{21}(\sigma^{n_2} \cdot \sigma^{e_1}), \quad (2)$$

で与えられる⁹⁾。ただし、

$$H_i = \mu_B B \sigma_z^{e_i} - g_n \mu_n B \sigma_z^{n_i} + A_{ii}(\sigma^{n_i} \cdot \sigma^{e_i}), \quad (3)$$

である。ここで、 σ^x は粒子 x に対する Pauli のスピン行列を表し、粒子 n_i および e_i は i 番目の P 原子核および電子、 B は磁束密度を表す。また、 A_{ij} および J_{ij} は i 番目と j 番目の核・電子スピン間および電子・電子スピン間の相互作用強度を表す。

式 (2) を数値シミュレーションの簡便のため無次元化し、静磁場 ($B = 0$) において、組となる核スピンと電子スピン、および電子スピン間のみ相互作用が生じる ($A_{12} = A_{21} = 0$) として、 $J = J_{12}$ の変化に対する各量子状態の観測確率を Fig. 2 に示す。これより、 J を大きくすることで $|01\rangle$ および $|10\rangle$ の観測確率のみが高くなっており、すなわち抑制性の結合を示唆する結果が得られた。これらは先行研究である中宮ら¹⁰⁾ と同様の結果である。

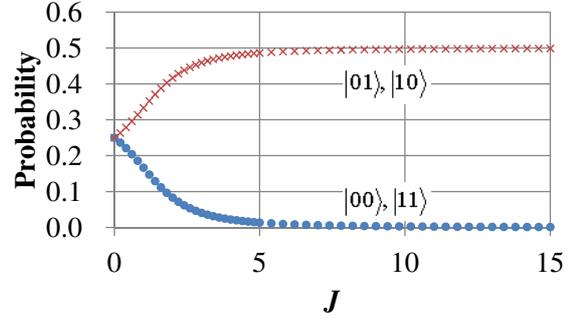


Fig. 2: Probability of the states of a 2-qubit device in Si as a function of $J = J_{12}$.

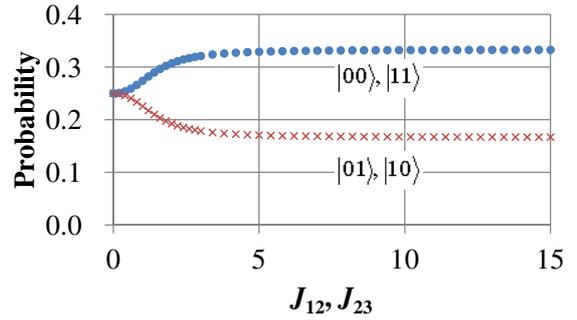


Fig. 3: Probability of the states $|0x0\rangle$, $|0x1\rangle$, $|1x0\rangle$, and $|1x1\rangle$ of a 3-qubit device in Si as a function of $J = J_{12} = J_{23}$.

これを拡張して興奮性の結合を表す Hamiltonian の構成法を考える。この際、抑制性の結合のみが許されるという限定条件を持つ ANN を想定し、3 つのニューロンを隣接配置したモデル (ただし両端のニューロン同士は結合を持たない) において、両端のニューロンだけを見る (すなわち中間のニューロンは隠れニューロンとして見なす) と、それらは興奮性の結合を持つものとして考えることができる。一方、3-qubit デバイスの Hamiltonian は、

$$H = H_1 + H_2 + H_3 + J_{12}(\sigma^{e_1} \cdot \sigma^{e_2}) + J_{23}(\sigma^{e_2} \cdot \sigma^{e_3}) + J_{31}(\sigma^{e_3} \cdot \sigma^{e_1}) + A_{12}(\sigma^{n_1} \cdot \sigma^{e_2}) + A_{13}(\sigma^{n_1} \cdot \sigma^{e_3}) + A_{21}(\sigma^{n_2} \cdot \sigma^{e_1}) + A_{23}(\sigma^{n_2} \cdot \sigma^{e_3}) + A_{31}(\sigma^{n_3} \cdot \sigma^{e_1}) + A_{32}(\sigma^{n_3} \cdot \sigma^{e_2}), \quad (4)$$

で与えられるので、前述の ANN の場合と同様に考えると、興奮性の結合を持つためには両端の量子ビット間の相互作用は無いので、 $J_{31} = 0$ および $A_{13} = A_{31} = 0$ の場合に相当する。

前述の数値シミュレーション結果と同様に式 (4) を無次元化し、静磁場 ($B = 0$) において、組となる核スピンと電子スピン、および電子スピン間のみ相互作用が生じる ($A_{ij} = 0$) とした、 $J = J_{12} = J_{23}$ に対する各量子状態の観測確率を Fig. 3 に示す。これより J を大きくすることで $|00\rangle$ および $|11\rangle$ の観測確率が高くなっており、すなわち興奮性の結合を示唆する結果が得られた。ただしこの場合、前述した 2-qubit デバイスと

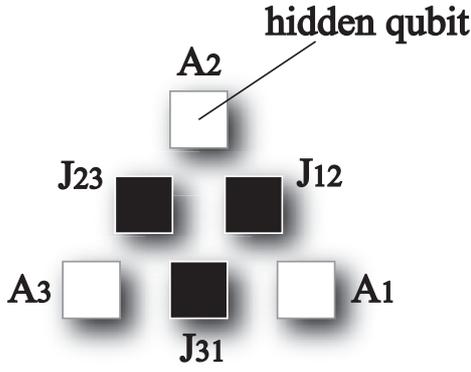


Fig. 4: Topview of a proposed qubit-device structure with three ^{31}P nuclear spins in ^{28}Si based on Kane's device.

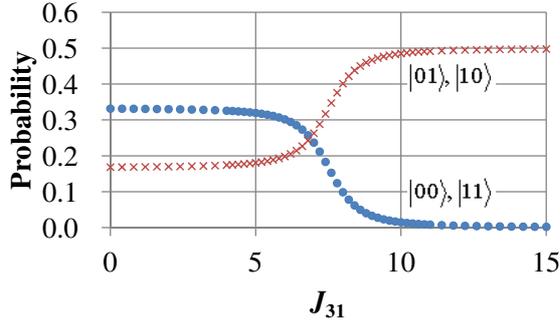


Fig. 5: Probability of the states of a proposed 3-qubit device in Si as a function of J_{31} ($J_{12} = J_{23} = 7.5$).

同様の抑制性を示す状態 (誤った状態) の確率 ($|01\rangle$ および $|10\rangle$ の観測確率) が残ってしまっているが、興奮性の結合を示す状態より低い確率となっている。

これまで述べた 2-および 3-qubit デバイスは、qubit 数やその性質が異なるため、そのまま集積化した際の結合の性質 (抑制性または興奮性) を、学習などによって後天的に変更することはできない。そこで、単純な操作で前述の抑制性と興奮性の両者を表現できることを前提に、3-qubit デバイスの構造を工夫することで、その実現を検討する。提案するデバイス構造を Fig. 4 に示す。これは各 qubit (A_i の直下に配置された ^{31}P 原子) を正三角形の各頂点に配置し、それぞれの相互作用 J_{ij} を制御可能とした構造である。すなわち前述の数値シミュレーション条件において、 $J_{31} \neq 0$ とした場合に相当する。ここで、式 (4) において、 $J_{12} = J_{23}$ を固定し J_{31} を変化させた際の各状態の観測確率を Fig. 5 に示す。これよりひとつの制御端子 J_{31} を制御するだけで抑制性の結合 ($J_{31} > J_{12}$) と興奮性の結合 ($J_{31} < J_{12}$) を表現できることがわかった。ただし、興奮性を示す部分では誤る確率が一定の割合で残っていることに注意が必要である。

3.2 ダイヤモンド中の N-V 中心

Fig. 6 はダイヤモンドの中で起こる N-V 中心と呼ばれる格子欠陥 (いわゆる不純物) を模式的に表したものである。これは、ピンクダイヤモンドなどの着色の原因となるが、qubit として利用可能なスピンを持ち、スピン状態を電磁波により操作できる等の特徴を持っている。量子状態を持続できる時間 (コヒーレンス時

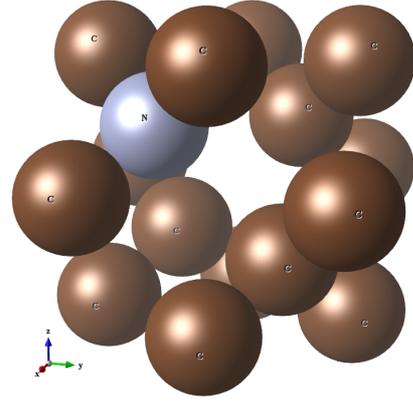


Fig. 6: Illustration of the Nitrogen-Vacancy(N-V) center in diamond.

間) が長く、室温動作も期待できる。室温動作可能な qubit デバイスとして注目されるダイヤモンド中の N-V 中心の Hamiltonian は、

$$H = D_{es}S_z^2 + \mu_B g_{es} \mathbf{B} \cdot \mathbf{S} + E_{es}(S_x^2 - S_y^2) - A_{es} \mathbf{S} \cdot \boldsymbol{\sigma}, \quad (5)$$

で与えられる^{11, 12)}。ここで、 D_{es} は励起状態 (ES) の縦方向のゼロ磁場分裂、 g_{es} は ES の g 因子、 A_{es} は超微細分 E_{es} はひずみ誘起横方向異方性の係数である。また、 \mathbf{S} および $\boldsymbol{\sigma}$ はスピン 1 およびスピン 1/2 のスピン演算子である。

この N-V 中心を 2 つ作用させた場合における Hamiltonian を考えると、

(a) ゼロ磁場分裂、

$$H_a = D_{es}S_z^2 + E_{es}(S_x^2 - S_y^2), \quad (6)$$

(b) 1 番目および 2 番目の電子スピンのゼーマン相互作用、

$$H_b = \mu_B g_{es} \mathbf{B} \cdot \mathbf{S}_1 + \mu_B g_{es} \mathbf{B} \cdot \mathbf{S}_2, \quad (7)$$

(c) 1 番目および 2 番目の ^{15}N による超微細相互作用、

$$H_c = -A_{11} \mathbf{S}_1 \cdot \boldsymbol{\sigma}_1 - A_{22} \mathbf{S}_1 \cdot \boldsymbol{\sigma}_1, \quad (8)$$

(d) ^{15}N による 1 番目の核スピン 1/2 とその隣に存在する別の核スピン 1/2 との間に働く相互作用、

$$H_d = J_{12} \boldsymbol{\sigma}_1 \cdot \boldsymbol{\sigma}_2, \quad (9)$$

(e) 1 番目の N-V 中心によるスピン 1 と 2 番目の ^{15}N による核スピン 1/2 間に働く相互作用、

$$H_e = J_{o1} \mathbf{S}_1 \cdot \boldsymbol{\sigma}_2, \quad (10)$$

(f) 2 番目の N-V 中心によるスピン 1 と 1 番目の ^{15}N による核スピン 1/2 間に働く相互作用、

$$H_f = J_{o2} \mathbf{S}_2 \cdot \boldsymbol{\sigma}_1, \quad (11)$$

より、2-qubit における Hamiltonian はそれらの和、

$$H = H_a + H_b + H_c + H_d + H_e + H_f, \quad (12)$$

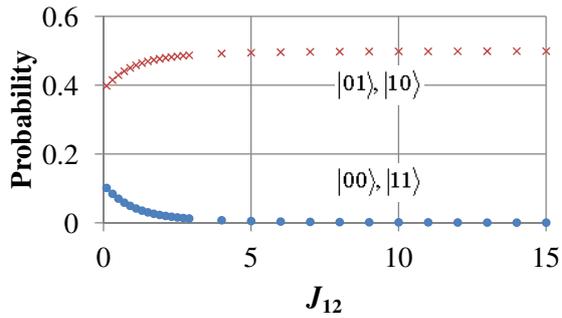


Fig. 7: Probability of the states of a 2-qubit device in diamond as a function of J_{12} .

となる。ここで、 J_{ij} は i 番目と j 番目の ^{15}N による核スピンの相互作用の大きさであり、 A_{ii} は i 番目内の N-V 中心のスピンの大きさである。また、 J_{o1} は 1 番目の N-V 中心によるスピンの大きさであり、 J_{o2} は 2 番目の N-V 中心によるスピンの大きさである。

ここで 1 番目および 2 番目の N-V 中心における ^{15}N による核スピン 1/2 を情報の担い手として着目した際、 J_{12} を変化させた 2-qubit の Hamiltonian による状態の観測確率の変化を Fig. 7 に示す。ただし、2-qubit における Hamiltonian 中のゼロ磁場分裂、ゼーマン相互作用は固定とし、また J_{o1} および J_{o2} はゼロとして無視した。Fig. 7 より $|01\rangle$ および $|10\rangle$ の観測確率のみが高くなっており、すなわち抑制性の結合を示唆する結果が得られた。これにより、抑制性のそれを組み合わせて興奮性を実現することで、NAQC 実現のためのデバイスの候補として考えることができるが、他のパラメータなどについてさらなる検討が必要である。

4 考察

Si 中の ^{31}P 原子核スピンを利用したデバイスについては、デバイスの構造を工夫することで、ひとつのデバイスで抑制性や興奮性の結合を制御できることを示した。具体的には 3 つの J-gate の電圧制御において、2 つを固定バイアスとし、残りひとつを可変とすることで、実現できることを示した。ただし、興奮性の結合を示す領域で、誤り（抑制性）の確率 32% 程度を含むことに注意しなければならない。この特性が、大規模応用に与える影響は今後検討していかなければならない。

また、ダイヤモンド中の N-V 中心を利用したデバイスについては、まだ十分に検討が進んでいないものの、 ^{15}N 原子核スピンを情報の担い手として 2 つ隣接させた場合、前述の Si でのデバイスと同様に抑制性の性質を持つことを示した。ただし、それを並べて興奮性の結合を持つデバイスを構成する際、qubit 間のそれぞれの相互作用をどのように制御するのか具体的な方法は現在検討中であり、その結論も踏まえた検討が必要である。

ここで、もし興奮性のみの結合しか見いだせない場合、そのデバイスと NAQC を利用した ANN 分野への応用はかなり限定されることとなる。しかしながら、Si

およびダイヤモンドを用いたいずれのデバイスにおいても抑制性の結合が見いだせたので、これらデバイスは NAQC の実行可能性が高いと言える。

5 まとめと今後の課題

ニューロ様断熱的量子計算の実行に適した qubit デバイスとして、Si 中の P 原子核スピンを利用したデバイスとダイヤモンド中の N-V 中心を利用したデバイスを検討した。前者においては、3-qubit デバイスで制御を工夫することにより、興奮性および抑制性を示す量子状態を作り出すことができた。しかし、一定の誤り率も存在するため、その改善を検討する必要がある。後者においては、ダイヤモンド中で起こる格子欠陥の量子力学的効果を、qubit の状態として考慮した結果、2 つの N-V 中心における ^{15}N による核スピン 1/2 を qubit として着目した際、2-qubit の Hamiltonian により、抑制性を示す量子状態と見なせる結果が得られた。以上の結果を考慮しつつ、今後大規模化に向けて、それらデバイスの実現も見据えた詳細な検討を行う。

参考文献

- 1) Gennady P Berman, et al. : Introduction to Quantum Computer, World Scientific, Singapore(1998).
- 2) M. A. Nielsen and I. L. Chuang : Quantum Computation and Quantum Information, Cambridge University Press, Cambridge (2000).
- 3) P. W. Shor : Polynomial-time algorithm for prime factorization and discrete logarithms on a quantum computer, SIAM J. Comput., **26**, 1484/1509 (1997).
- 4) L. K. Grover : A fast quantum mechanical algorithm for database search, Proc. Twenty-Eighth Annual ACM Symp. on the Theory of Computing, 212/219 (1996).
- 5) E. Farhi, J. Goldstone, S. Gutmann, J. Lapan, A. Lundgren, and D. Preda : A Quantum Adiabatic Evolution Algorithm Applied to random Instances of an NP-Complete Problem, Science, **292**, 472/476 (2001).
- 6) S. Sato, M. Kinjo, and K. Nakajima : An Approach for Quantum Computing using Adiabatic Evolution Algorithm, Jpn. J. Appl. Phys., **42**, 7169/7173 (2003).
- 7) M.Kinjo et al. : Neuromorphic quantum computation with energy dissipation, Physical review A, **72**, 052328-1/052328-6 (2005).
- 8) A. Messiah : Quantum Mechanics, 747/755, Dover, New York (1999).
- 9) B.E.Kane: A silicon-based nuclear spin quantum computer, Nature, **393**, 133/137 (1998).
- 10) 中宮裕希ほか: Kane の量子ビットを用いた量子ニューラルネットワークの基本特性の検証, 信学技報, NLP2005-4 (2005).
- 11) J. Wrachtrup, et al. : Optics and Spectroscopy, **91**-3 (2001).
- 12) G.D. Fuchs, et al. : Physical review Letter, **101**, 117601 (2008).

ニューラルネットワークによる離散事象システムの実現と同定法 —高次結合表現とスパース実現—

○黒江康明 森禎弘 (京都工芸繊維大学)

Identification and Realization of A Class of Discrete Event Systems by Neural Networks – High Order Connections and Sparse Realization

*Y. Kuroe and Y. Mori (Kyoto Institute of Technology)

Abstract— In analysis and design of a system, the first target is to obtain an appropriate model of the system. Because of the complex dynamics of discrete event systems (DESs), it is difficult to obtain a model of unknown DESs from given input and output data. This paper discusses an identification and realization method of a class of DESs by neural networks. We consider a class of DESs which is modeled by using finite state automata. In identification and realization of systems by using neural networks, it is essentially important to develop a suitable architecture of neural networks. We already proposed two neural network architectures: one is a class of recurrent neural networks and the other is a class of recurrent high-order neural networks, which are capable of representing DESs with the network size being smaller than the existing neural network models. In this paper we present an identification method of DESs, which makes it possible to obtain sparse realization, that is, to obtain networks with simpler structure. It is shown through numerical experiments that presented method makes it possible to obtain simpler neural networks which can exactly simulate target DESs.

Key Words: discrete event system, finite state automaton, recurrent neural network, high-order neural network, identification, realization

1 はじめに

近年、人間の脳の情報処理機能を模倣したニューラルネットワーク(以下、NN)を利用したオートマトンの実現、学習に関係する研究が盛んに行われている^{1, 2, 3, 4, 5, 6}。たとえば正則言語を識別する機械の学習として、ある入力文字列の受理/非受理を判定するNNを求める問題がある。このようなオートマトンの学習問題は言語の認識装置の実現の方法として様々な点から研究されている。

一方、現実の多くの制御システムにおいて離散事象システムとして扱うべきシステムがあらゆるところで見られるようになってきた。また、現実の多くのシステムは何らかの意味でハイブリッドシステムとなっている、すなわち連続変数を扱う連続システムと離散変数を扱う離散事象システムが結合したシステムとなっており、そのようなシステムの解析、設計、制御に関する研究が非常に盛んになっている。

一般にシステムを解析、設計する際にまずやるべきことは、対象とするシステムのモデルを構築すること、すなわちモデリングである。離散事象システムにおいては、そのダイナミクスの複雑性より与えられたデータよりその数学モデルを求めるのが非常に困難になっており、未知なる離散事象システムを実現、同定する方法の開発が望まれている。

本稿では、有限オートマトン(Finite State Automata (FSA))で表される離散事象システムを対象として、NNを用いて実現あるいは同定する方法について議論する。その際、問題となるのは対象の離散事象システムを実現あるいは同定するNNのアーキテクチャーをどのように導くかである。筆者らはすでに、このためのNNのモデルを2つ提案している⁷。すなわち、1つはあるクラスのリカレントNN(Recurrent Neural Networks (RNN))であり、他の1つはその結合に高次結合を含

むりカレント高次NN(Recurrent High-Order Neural Networks (RHONN))である。これら2つのモデルは有限オートマトンで表される離散事象システムを厳密に表現できる。また従来提案されているNNモデルよりニューロン数が少ないNNで表現することができ、決定すべき未知パラメータが少なくできるため、実現や同定問題にとって都合が良い。本稿では、これら2つのモデルの導出を示すとともに、これらを用いて入力データより未知なる離散事象システムを実現あるいは同定する方法について議論する。特に、より構造の簡単な(パラメータ数が少ないスパースな)ネットワークを得るための実現、同定法を示す⁸。またいくつかの数値実験を行った結果を示し、その性能を評価する。

2 問題設定

2.1 対象とする離散事象システムのモデル

本稿では、次式で定義される決定性有限オートマトンで表される離散事象システム(Discrete Event Systems, 以下DESとする)を対象とする。

$$\mathcal{M} = (Q, q_0, \Sigma, \Delta, \delta, \varphi) \quad (1)$$

ここで、 Q は状態シンボルの集合: $Q = \{q^{(1)}, q^{(2)}, \dots, q^{(r)}\}$, r は状態シンボルの数, $q_0 \in Q$ は初期状態, Σ は入力シンボルの集合: $\Sigma = \{i^{(1)}, i^{(2)}, \dots, i^{(m)}\}$, m は入力のシンボルの数, Δ は出力シンボルの集合: $\Delta = \{o^{(1)}, o^{(2)}, \dots, o^{(l)}\}$ で、 l は出力シンボルの数である。また $\delta: Q \times \Sigma \rightarrow Q$ は状態遷移関数, $\varphi: Q \times \Sigma \rightarrow \Delta$ は出力関数である。

このDES \mathcal{M} はある単位時間ごとに動作するものとし、ある時刻 t での入力、出力、状態をそれぞれ $i(t) \in \Sigma$, $o(t) \in \Delta$, $q(t) \in Q$ とするとDES \mathcal{M} は次式のような離散時間ダイナミカルシステムとして表現できる。

$$\mathcal{M}: \begin{cases} q(t+1) = \delta(q(t), i(t)), & q(0) = q_0 \\ o(t) = \varphi(q(t), i(t)) \end{cases} \quad (2)$$

2.2 DESの実現, 同定問題

本稿では, (1) 式あるいは (2) 式で表される未知なる DES を対象とし, NN により実現, 同定する問題を考える. そのため次の仮定をおく.

仮定 1 状態シンボルの集合 Q および初期状態 q_0 は未知.

仮定 2 状態遷移関数 δ および出力関数 φ は未知.

仮定 3 入力シンボルの集合 Σ および出力シンボルの集合 Δ は既知.

仮定 4 入力データ列 $\{i(t)\}$ とこれに対応する出力データ列 $\{o(t)\}$ は既知あるいは観測可能.

これらの仮定は, 実際の DES の実現や同定問題において自然な仮定となっている. これらの仮定のもと, NN による DES の実現や同定問題は次のようになる. “未知なる対象の DES の入出力データ列 $\{i(t)\}$ および $\{o(t)\}$ が与えられたとし, 入力データ列 $\{i(t)\}$ を NN の入力に入力したとき, その出力が出力データ列 $\{o(t)\}$ に一致するような NN を求める”. この問題を適切に解決するためには, 次の二つの問題を検討することが必要となる. 一つ目の問題は, DES の実現や同定に相応しい NN のアーキテクチャをどのように決めるかである. また二つ目の問題は, 与えられた対象の DES の入出力データより, 決定したアーキテクチャの NN のパラメータ (結合荷重や閾値) の値をどのように決めるかである. 特に, DES の実現において, できるだけ簡素な NN として実現することが望まれることが多く, このことを実現する方法を開発することも重要となる. 本稿ではまず前者の問題について議論する. 次に後者の問題, 特に零の結合係数が出来るだけ多い NN として実現, すなわちスパースな NN を実現する方法について議論する.

3 高次結合を持つニューラルネットワークのモデル

ここでは, DES の実現および同定問題にふさわしい NN のアーキテクチャを導くために, 一般的なクラスの NN のモデルを示す. ここで考える NN は Fig. 1 示すようなネットワークで, 通常の結合に加えて高次結合を含む^{9, 10}. このネットワークは, ニューロンユニット, 結合ユニットおよび外部入力, 外部出力からなり, ニューロンユニットは2種類のニューロン, すなわちダイナミクスを持つ動的ニューロンと持たない静的ニューロンを考える. すべてのニューロンと結合ユニットは任意に結合しており, また結合ユニットは通常の結合だけでなく高次結合を含む.

ここで, N_d , N_s , N_c , M および L をそれぞれ動的ニューロン, 静的ニューロン, 結合ユニット, 外部入力, 外部出力の数とする. 動的ニューロンの数学モデルは次のように表わされる.

$$\begin{aligned} v_i^d(t+1) = & \sum_{j=1}^{N_d} w_{ij}^{dd} h_j^d(t) + \sum_{j=1}^{N_s} w_{ij}^{ds} h_j^s(t) \\ & + \sum_{j=1}^{N_c} w_{ij}^d z_j(t) + \sum_{j=1}^M w_{ij}^{dI} I_j(t) \end{aligned}$$

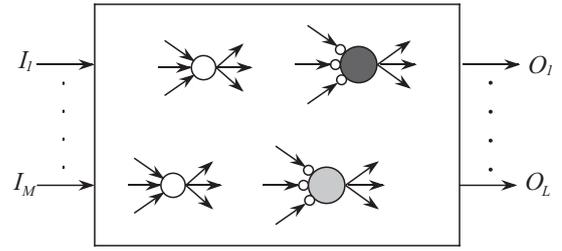
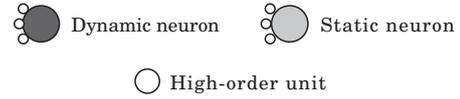


Fig. 1: Recurrent neural networks with high-order connection units.

$$+ \theta_i^d, \quad v_i^d(0) = v_{i0}^d \quad (3)$$

$$h_i^d(t) = f_i^d(v_i^d(t)), \quad (i = 1, 2, \dots, N_d) \quad (4)$$

また, 静的ニューロンの数学モデルは次のようにあらわされる.

$$\begin{aligned} u_i^s(t) = & \sum_{j=1}^{N_s} w_{ij}^{ss} h_j^s(t) + \sum_{j=1}^{N_d} w_{ij}^{sd} h_j^d(t) + \sum_{j=1}^{N_c} w_{ij}^s z_j(t) \\ & + \sum_{j=1}^M w_{ij}^{sI} I_j(t) + \theta_i^s \end{aligned} \quad (5)$$

$$h_i^s(t) = f_i^s(u_i^s(t)), \quad (i = 1, 2, \dots, N_s) \quad (6)$$

ここで, $v_i^d(t)$, v_{i0}^d , $h_i^d(t)$ および θ_i^d はそれぞれ, 動的ニューロン i の状態, 初期状態, 出力, 閾値であり, $u_i^s(t)$, $h_i^s(t)$ および θ_i^s はそれぞれ, 静的ニューロン i の状態, 出力, 閾値である. $z_j(t)$ は, 結合ユニット j の出力, $I_j(t)$ は外部入力 j である. また, w_{ij}^{dd} は動的ニューロン j から動的ニューロン i , w_{ij}^{ds} は静的ニューロン j から動的ニューロン i , w_{ij}^d は結合ユニット j から動的ニューロン i , w_{ij}^{dI} は外部入力 I_j から動的ニューロン i , w_{ij}^{ss} は静的ニューロン j から静的ニューロン i , w_{ij}^{sd} は動的ニューロン j から静的ニューロン i , w_{ij}^s は結合ユニット j から静的ニューロン i , w_{ij}^{sI} は外部入力 I_j から静的ニューロン i への結合加重である. また, $f_i^d(\cdot)$, $f_i^s(\cdot)$ はそれぞれ動的ニューロン i , 静的ニューロン i の出力関数である.

結合ユニットは, 各ニューロンの間および外部入力との間の結合を決めるユニットで, 通常の結合に加えて高次の結合を含む. \mathbf{g} を次のように定義される $N_d + N_s + M$ 次元のベクトルとする.

$$\mathbf{g} = [h_1^d, h_2^d, \dots, h_{N_d}^d, h_1^s, h_2^s, \dots, h_{N_s}^s, I_1, I_2, \dots, I_M]^T. \quad (7)$$

結合ユニットの数学モデルは,

$$z_j(t) = G_j(\mathbf{g}(t)), \quad (j = 1, 2, \dots, N_c) \quad (8)$$

と表わされる. ここで, $G_j(\mathbf{g}(t))$ は次のように定義される.

$$G_j(\mathbf{g}(t)) := \prod_{k \in J_j} g_k^{d_k(j)}(t). \quad (9)$$

この式において、 $g_k(t)$ は $\mathbf{g}(t)$ の第 k 要素、 J_j はインデックス集合 $\{1, 2, \dots, N_d + N_s + M\}$ の部分集合、また $d_k(j)$ は非負の整数である。ここで $d_k(j)$ は指数をあらわし、上付き記号ではないことに注意されたい。

外部出力 O_i は、次のように表わされる。

$$O_i(t) = \sum_{j=1}^{N_d} \delta_{ij}^d h_j^d(t) + \sum_{j=1}^{N_s} \delta_{ij}^s h_j^s(t) \quad (10)$$

$$(i = 1, 2, \dots, L)$$

ここで δ_{ij}^d および δ_{ij}^s は 1 あるいは 0 の値をとるパラメータで、もし動的ニューロン j (静的ニューロン j) が外部出力 O_i に接続されている場合は $\delta_{ij}^d = 1$ ($\delta_{ij}^s = 1$)、接続されていない場合は $\delta_{ij}^d = 0$ ($\delta_{ij}^s = 0$) とする。

4 DES のニューラルネットワーク表現

これまで、NN による FSA の表現あるいは学習に関する研究は、特に言語の認識装置の実現の分野においていくつかの進展がなされているが、その代表的なモデルは 2 次の結合を含むリカレント 2 次結合 NN である^{3, 5)}。これらの NN モデルにおいては、FSA における入力、状態、出力の各シンボルは、基底ベクトルを用いて表されている。たとえば、各状態シンボル $q^{(i)}$ は、 $q^{(1)} = (1, 0, \dots, 0)$ 、 $q^{(2)} = (0, 1, \dots, 0)$ 、 \dots 、 $q^{(r)} = (0, 0, \dots, 1)$ のように、 r 次元の単位基底ベクトルで表される。また FSA の各状態は、それぞれ一つのニューロンが割り当てられて表現される。このように導かれたリカレント 2 次結合 NN においては、FSA の各状態ごとに 1 つのニューロンを割り当てて構成されるため、対象とする FSA の状態の数が多くなると、FSA を表現するのに必要なニューロン数は多くなる。そのため、FSA で表される DES の実現や同定において決定すべき未知パラメータが多くなり問題となる。これに対して筆者らはすでに、FSA で表される DES に対し、この問題を解決した NN モデルを二つ提案している。一つはあるクラスのリカレント NN (RNN) であり、他の 1 つは高次結合を含むリカレント高次 NN (RHONN) であり、より少ないニューロンを持つネットワークで DES を表現できる。以下ではそれらを導出する。

4.1 リカレントニューラルネットワーク表現

まず、DES \mathcal{M} の実現や同定のための高次結合のない通常の RNN のモデルを導出する。DES \mathcal{M} のすべての状態 $q^{(i)}$ ($i = 1, 2, \dots, r$)、入力シンボル $i^{(i)}$ ($i = 1, 2, \dots, m$) および出力シンボル $o^{(i)}$ ($i = 1, 2, \dots, l$) を 2 進符号化して表わす。たとえば、DES \mathcal{M} の状態シンボルの数を 4、すなわち $r = 4$ とすると、 $q^{(1)} = (0, 0)$ 、 $q^{(2)} = (0, 1)$ 、 $q^{(3)} = (1, 0)$ 、 $q^{(4)} = (1, 1)$ のように表す。したがって、(2) 式における $q(t)$ 、 $i(t)$ および $o(t)$ は次のように表現される。

$$\begin{aligned} q(t) &= (s_1(t), s_2(t), \dots, s_\alpha(t)) \quad (s_i(t) \in \{0, 1\}) \\ i(t) &= (x_1(t), x_2(t), \dots, x_\beta(t)) \quad (x_i(t) \in \{0, 1\}) \\ o(t) &= (y_1(t), y_2(t), \dots, y_\gamma(t)) \quad (y_i(t) \in \{0, 1\}) \end{aligned} \quad (11)$$

ここで、 α 、 β 、 γ はそれぞれ r 、 m 、 l に依存して決まる自然数である。すなわち、 α 、 β 、 γ はそれぞれ、 $r \leq 2^\alpha$ 、 $m \leq 2^\beta$ 、 $l \leq 2^\gamma$ を満たす最小の自然数である。このよ

うに各変数を 2 進符号化することにより、DES の状態遷移関数と出力関数は論理関数で表わされる。

ここで、 z_i 、 $i = 1, 2, \dots, n$ ($n := \alpha + \beta$) なる新たな変数を導入し、 $z_1 = s_1, z_2 = s_2, \dots, z_\alpha = s_\alpha, z_{\alpha+1} = x_1, z_{\alpha+2} = x_2, \dots, z_n = x_\beta$ とする。また、(i) いかなる論理関数も標準展開定理より基本積の論理和で表わすことができる、(ii) 真の値を 1、偽の値を -1 とすると、基本論理演算 AND, OR, NOT は通常の算術演算と

$$S(x) = \begin{cases} 1 & \text{for } x \geq 0 \\ -1 & \text{for } x < 0 \end{cases}$$

と定義される符号関数 $S(\cdot)$ を用いて表わすことができ、という事実を用いることにより、(2) 式の DES \mathcal{M} のモデルは次のように変換することができる。

$$\mathcal{M}: \begin{cases} s_i(t+1) = S\left(\sum_{j=1}^{2^n} a_{ij} Z_j(t) + n_i^s - 1\right) \\ \quad (i = 1, 2, \dots, \alpha) \\ y_i(t) = S\left(\sum_{j=1}^{2^n} b_{ij} Z_j(t) + n_i^y - 1\right) \\ \quad (i = 1, 2, \dots, \gamma) \end{cases} \quad (12)$$

ここで

$$\begin{aligned} Z_1(t) &= S(z_1(t) + \dots + z_{n-1}(t) + z_n(t) - (n-1)) \\ Z_2(t) &= S(z_1(t) + \dots + z_{n-1}(t) - z_n(t) - (n-1)) \\ &\vdots \\ Z_{2^n}(t) &= S(-z_1(t) - \dots - z_{n-1}(t) - z_n(t) - (n-1)) \end{aligned} \quad (13)$$

であり、また n_i^s 、 n_i^y はそれぞれ $\{a_{ij} : a_{ij} = 1, i = 1, 2, \dots, \alpha, j = 1, 2, \dots, 2^n\}$ 、 $\{b_{ij} : b_{ij} = 1, i = 1, 2, \dots, \gamma, j = 1, 2, \dots, 2^n\}$ なる要素の数である。(12)(13) 式より、DES \mathcal{M} を表現できる NN のモデルを次のように導くことができる。

(3)、(4)、(5)、(6)、(7)、(8)、(9)、(10) 式で表わされる NN のモデルを考え、 $N_d = \alpha$ 、 $N_s = 2^{\alpha+\beta} + \gamma$ 、 $N_c = 0$ 、 $M = \beta$ 、 $L = \gamma$ 、 $f_i^d(\cdot) = S(\cdot)$ 、 $f_i^s(\cdot) = S(\cdot)$ とする。このネットワークにおいて、動的ニューロンの状態ベクトル $\mathbf{v}^d = (v_1^d, v_2^d, \dots, v_\alpha^d)$ 、外部入力ベクトル $\mathbf{I} = (I_1, I_2, \dots, I_\beta)$ および外部出力ベクトル $\mathbf{O} = (O_1, O_2, \dots, O_\gamma)$ をそれぞれ、DES \mathcal{M} の 2 進符号化された状態 q 、入力 i および出力 o に対応させる。(12) 式の第 1 式を実現するために、(3)(4) 式で表わされる α 個の動的ニューロンを割り当て、 $w_{ij}^{dd} = 0$ for $i, j = 1, 2, \dots, \alpha$ 、 $w_{ij}^{ds} = 0$ for $i = 1, 2, \dots, \alpha$ 、 $j = 2^{\alpha+\beta} + 1, 2^{\alpha+\beta} + 2, \dots, 2^{\alpha+\beta} + \gamma$ 、 $w_{ij}^{dl} = 0$ for $i = 1, 2, \dots, \alpha$ 、 $j = 1, 2, \dots, \beta$ とする。ここで、 θ_i^d ($i = 1, 2, \dots, \alpha$) の値は、(12) 式の n_i^s の定義より、 w_{ij}^{ds} ($i = 1, 2, \dots, \alpha$ 、 $j = 1, 2, \dots, 2^{\alpha+\beta}$) の値より一意に決まることに注意されたい。

(13) 式を実現するために、 $2^{\alpha+\beta}$ 個の (5)(6) 式で表わされる静的ニューロンを割り当て、 $w_{ij}^{ss} = 0$ for $i, j = 1, 2, \dots, 2^{\alpha+\beta}$ とする。 w_{ij}^{sd} ($i = 1, 2, \dots, 2^{\alpha+\beta}$ 、 $j = 1, 2, \dots, \alpha$)、 w_{ij}^{sl} ($i = 1, 2, \dots, 2^{\alpha+\beta}$ 、 $j = 1, 2, \dots, \beta$) および θ_i^s ($i = 1, 2, \dots, 2^{\alpha+\beta}$) の値は (13) 式から決め

ることができ、 w_{ij}^{sd} と w_{ij}^{sl} は '1' あるいは '0' の値をとり、 $\theta_i^s = -(n-1)$ ($i = 1, 2, \dots, 2^{\alpha+\beta}$) なる値をとる。(12) 式の第 2 式を実現するために、さらなる γ 個の (5)(6) 式で表わされる静的ニューロンを割り当て、 $w_{ij}^{ss} = 0$ for $i = 2^{\alpha+\beta} + 1, 2^{\alpha+\beta} + 2, \dots, 2^{\alpha+\beta} + \gamma$, $j = 2^{\alpha+\beta} + 1, 2^{\alpha+\beta} + 2, \dots, 2^{\alpha+\beta} + \gamma$, $w_{ij}^{sd} = 0$ for $i = 2^{\alpha+\beta} + 1, 2^{\alpha+\beta} + 2, \dots, 2^{\alpha+\beta} + \gamma$, $j = 1, 2, \dots, \alpha$, $w_{ij}^{sl} = 0$ for $i = 2^{\alpha+\beta} + 1, 2^{\alpha+\beta} + 2, \dots, 2^{\alpha+\beta} + \gamma$, $j = 1, 2, \dots, \beta$ とする。 θ_i^s ($i = 2^{\alpha+\beta} + 1, 2^{\alpha+\beta} + 2, \dots, 2^{\alpha+\beta} + \gamma$) の値は (12) 式の n_i^y の定義より、 w_{ij}^{ss} ($i = 2^{\alpha+\beta} + 1, 2^{\alpha+\beta} + 2, \dots, 2^{\alpha+\beta} + \gamma$, $j = 1, 2, \dots, 2^{\alpha+\beta}$) の値を用いて一意に決まることに注意されたい。さらに、(10) 式で表わされる外部出力を割り当て、 $\delta_{ij}^d = 0$ for $i = 1, 2, \dots, \gamma$, $j = 1, 2, \dots, \alpha$, $\delta_{ij}^s = 0$ for $i = 1, 2, \dots, \gamma$, $j = 1, 2, \dots, 2^{\alpha+\beta}$ とする。

Fig. 2 に、以上のように構築された RNN を示す。この RNN は α 個の動的ニューロンと $2^{\alpha+\beta} + \gamma$ 個の静的ニューロンからなり、(12) 式と (13) 式を実現しており、DES \mathcal{M} を厳密に表現できる能力を有している。

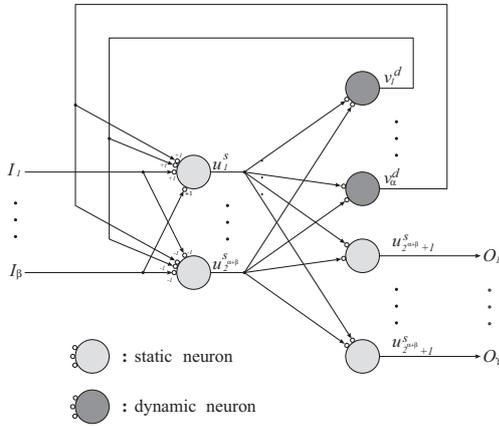


Fig. 2: Recurrent neural network for representing the DES

4.2 リカレント高次結合ニューラルネットワーク表現

ここでは DES \mathcal{M} の実現と同定のための、高次結合を含むリカレントニューラルネットワークのアーキテクチャを導く。前節と同様に、すべての状態シンボル $q^{(i)}$ ($i = 1, 2, \dots, r$)、入力シンボル $i^{(i)}$ ($i = 1, 2, \dots, m$) および出力シンボル $o^{(i)}$ ($i = 1, 2, \dots, \ell$) を、(11) 式のように 2 進符号化して表わす。ここで、(i) いかなる論理関数も標準展開定理より基本積の論理和で表わすことができる、(ii) 真の値を 1、偽の値を 0 とすると、基本論理演算 AND, OR, NOT は通常の算術演算と

$$H(x) = \begin{cases} 1 & \text{for } x \geq 0 \\ 0 & \text{for } x < 0 \end{cases}$$

と定義される Heviside 関数 $H(\cdot)$ を用いて表わすことができる、という事実を用いることにより、(2) 式の

DES のモデルは次のように変換することができる。

$$\mathcal{M} : \begin{cases} s_i(t+1) = H\left(\sum_{j=1}^{2^n-1} a_{ij}^* z_j^*(t) + a_{i2^n} - \gamma_i^q\right), \\ s_i(0) = s_{i0} \quad (i = 1, 2, \dots, \alpha) \\ y_i(t) = H\left(\sum_{j=1}^{2^n-1} b_{ij}^* z_j^*(t) + b_{i2^n} - \gamma_i^y\right) \\ \quad (i = 1, 2, \dots, \gamma) \end{cases} \quad (14)$$

ここで

$$\begin{cases} z_1^*(t) = z_1(t)z_2(t)z_3(t)\cdots z_n(t) \\ z_2^*(t) = z_2(t)z_3(t)\cdots z_n(t) \\ z_3^*(t) = z_3(t)z_4(t)\cdots z_n(t) \\ \vdots \\ z_{2^n-1}^*(t) = z_n(t) \end{cases} \quad (15)$$

であり、 a_{ij}^* , b_{ij}^* は適当な整数、 γ_i^q , γ_i^y はそれぞれ $0 < \gamma_i^q \leq 1$, $0 < \gamma_i^y \leq 1$ なる実数である。(14)(15) 式より、DES \mathcal{M} を表現できる NN のモデルを次のように導くことができる。

(3), (4), (5), (6), (7), (8), (9), (10) 式で表わされる NN のモデルを考え、 $N_d = \alpha$, $N_s = \gamma$, $N_c = 2^n - 1$ ($n = \alpha + \beta$)、 $M = \beta$, $L = \gamma$, $f_i^d(\cdot) = H(\cdot)$, $f_i^s(\cdot) = H(\cdot)$ とする。また、(7) 式における \mathbf{g} を次のように定義する。

$$\mathbf{g} = [h_1^d, h_2^d, \dots, h_{\alpha}^d, I_1, I_2, \dots, I_{\beta}]^T. \quad (16)$$

動的ニューロンにおいて、 $w_{ij}^{dd} = 0$ for $i, j = 1, 2, \dots, \alpha$, $w_{ij}^{ds} = 0$ for $i = 1, 2, \dots, \alpha$, $j = 1, 2, \dots, \gamma$, $w_{ij}^{dl} = 0$ for $i = 1, 2, \dots, \alpha$, $j = 1, 2, \dots, \beta$, とし、静的ニューロンにおいて $w_{ij}^{ss} = 0$ for $i, j = 1, 2, \dots, \gamma$, $w_{ij}^{sd} = 0$ for $i = 1, 2, \dots, \gamma$, $j = 1, 2, \dots, \alpha$, $w_{ij}^{sl} = 0$ for $i = 1, 2, \dots, \gamma$, $j = 1, 2, \dots, \beta$. とする。さらに (8) 式で表わされる結合ユニットにおいては、次のようにする。(15) 式で定義される各 z_j^* に対し、添え字の集合 J_j^* を、 $J_1^* = \{1, 2, 3, \dots, n\}$, $J_2^* = \{2, 3, \dots, n\}$, \dots , $J_{2^n-1}^* = \{n\}$ のように定義し、これらを用いて (9) 式の J_j , $d_k(j)$ を、 $J_j = J_j^*$ for $j = 1, 2, \dots, 2^n - 1$, $d_k(j) = 1$ for $j = 1, 2, \dots, 2^n - 1$ とする。さらに、(10) 式で表わされる外部入力を割り当て、 δ_{ij}^d および δ_{ij}^s の値は、 $\delta_{ij}^d = 0$ for $i = 1, 2, \dots, \gamma$, $j = 1, 2, \dots, \alpha$, $\delta_{ij}^s = 1$, for $j = i$, $\delta_{ij}^s = 0$ for $j \neq i$, $i = 1, 2, \dots, \gamma$ のように選ぶ。

Fig. 3 に以上のように構築された RHONN を示す。この RHONN は、その結合ユニットにおいて n 次の積結合を含み、(14) 式および (15) 式より、動的ニューロンの状態 $v_i^d(t)$ 、外部入力 $I_i(t)$ 、外部出力 $O_i(t)$ をそれぞれ、DES \mathcal{M} の状態 $q_i(t)$ 、入力 $i_i(t)$ 、出力 $o_i(t)$ に対応させることにより、DES \mathcal{M} を厳密に表現できる。

なお、この RHONN に含まれる高次結合のモデルは、NN だけでなく、たとえば遺伝子ネットワークにおける遺伝子間の相互作用関数のモデルとしてもよく用いられる¹¹⁾。

5 DES の実現および同定法と数値実験

以上のように導かれた RNN あるいは RHONN を用いて、未知の DES を実現あるいは同定するために、ま

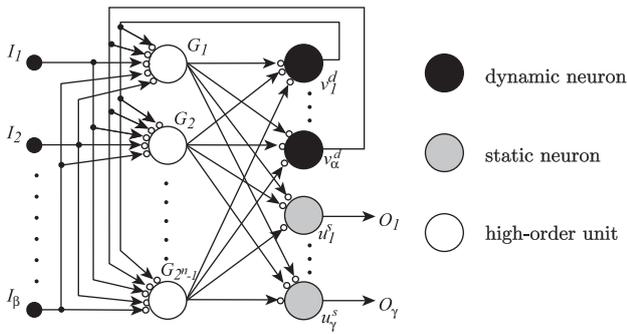


Fig. 3: Recurrent high-order neural network for representing the DES

ずRNNあるいはRHONNを、前節で述べた方法で構成することが必要になる。ところが、仮定より対象のDESの入力と出力の数は既知なので β と γ の値は一意に決めることができるが、状態の数は未知なので α の値は決めることはできない。すなわちRNNにおいては、外部入力および外部出力の数はそれぞれあらかじめ $M = \beta$, $L = \gamma$ と決めることができるが、動的ニューロンの数($N_d = \alpha$)と静的ニューロンの数($N_s = 2^{\alpha+\beta} + \gamma$)はあらかじめ決めておくことが出来ない。またRHONNにおいては、静的ニューロン、外部入力と外部出力の数はそれぞれあらかじめ $N_s = \gamma$, $M = \beta$ and $L = \gamma$ と決めることができるが、動的ニューロンの数 $N_d = \alpha$ と結合ユニットの数 $N_c = 2^n - 1$, ($n = \alpha + \beta$)はあらかじめ決めておくことはできない。そのため、DESの実現あるいは同定において、対象のDESの α より N_d が大きくなるようネットワークの動的ニューロンの数を十分多くすることにする(したがってRNNにおいては静的ニューロンの数、またRHONNにおいては結合ユニットの数も十分多くすることになる)。

以上のようにして構成したNNを用いて、対象とするDESの同定問題は次のようにして解くことができる。DESの入力データ列 $\{i(t) : t = 0, 1, 2, \dots, t_f\}$ とこれに対応する出力データ列 $\{o(t) : t = 0, 1, 2, \dots, t_f\}$ が与えられたとする。これに対し、構成したNNの入力 $I_j(t)$ に入力データ列 $\{i(t) : t = 0, 1, 2, \dots, t_f\}$ を入力し、その出力 $O_i(t)$ が出力データ列 $\{o(t) : t = 0, 1, 2, \dots, t_f\}$ に一致するようNNを学習させればよい。すなわちDESの同定問題は、次のような誤差関数を最小とするNNのパラメータを決める問題となる。

$$\frac{1}{2} \sum_{t=0}^{t_f} \sum_{i=1}^L |o_i(t) - O_i(t)|^2.$$

ここで注意すべきことは、前節で示したDESの実現及び同定のNNモデルには、零の値を持つパラメータが多く含まれ、これらを適切に決める必要があるということである。すなわち、可能な限り零の値を多く含むスパースな実現として対象のDESを実現、同定するNNを求めることが本質的に重要となる。そのため同定問題を、ネットワークの複雑性を考慮にいたした最適化問題として、NNの学習問題を定式化することにする。そのため、次のような評価関数を定義する。

$$J = J_1 + \rho J_2 \quad (17)$$

$$J_1 = \frac{1}{2} \sum_{t=0}^{t_f} \sum_{i=1}^L |o_i(t) - O_i(t)|^2$$

$$J_2 = \|\mathbf{w}\|_1 = \sum_{k,\ell=d,s} \sum_{i,j} |w_{ij}^{k\ell}| + \sum_{k=d,s} \sum_{i,j} |w_{ij}^k|$$

ここで、 $\rho > 0$ は適当な重み係数である。上式の評価関数において、 J_2 はNNのパラメータベクトルに関する ℓ^1 ノルムであり、これがネットワークの複雑度の尺度となりうる。石川はNNの学習問題において、ネットワークの複雑度を表す尺度としてパラメータベクトルのユークリッドノルム $\|\mathbf{w}\|_2$ より ℓ^1 ノルムをとる方が、零でないパラメータ数をより減らせるネットワークが得られることを示している¹²⁾。したがって、評価関数にネットワークのパラメータベクトルの ℓ^1 ノルムである J_2 を導入することによりスパースな実現ができることが期待できる。

以上により、DESの同定問題は(17)を最小化するNNの学習問題に帰着される。この評価関数を最小化するパラメータを求めるため、本稿では最急降下法、共役勾配法などの勾配法に基づく方法を用いる。勾配法を適用する際、RNNあるいはRHONNにおける非線形関数 $S(\cdot)$, $H(\cdot)$ は微分可能でないので、これらを精度よく近似できる滑らかな関数であるシグモイド関数に置き換えることにする。また、仮定1よりDESの初期状態 q_0 は未知なので、動的ニューロンの初期状態 $v_i^d(0)$ の値をあらかじめ決定することはできない。そのためこの初期値も最適化すべきパラメータに加えることにする。従って、この最適化問題における決定変数すなわち学習パラメータは、RNNの場合、結合重み w_{ij}^{ds} ($i = 1, 2, \dots, \alpha$, $j = 1, 2, \dots, 2^{\alpha+\beta}$), w_{ij}^{ss} ($i = 2^{\alpha+\beta} + 1, 2^{\alpha+\beta} + 2, \dots, 2^{\alpha+\beta} + \gamma$, $j = 1, 2, \dots, 2^{\alpha+\beta}$)と動的ニューロンの初期値 v_{i0}^d ($i = 1, 2, \dots, \alpha$)で、その総数は $2^{(\alpha+\beta)} \times (\alpha + \gamma) + \alpha$ となる。またRHONNの場合、学習パラメータは w_{im}^d , θ_i^d , w_{im}^s , θ_i^s , $v_i^d(0)$ で、その総数は $2^{(\alpha+\beta)} \times (\alpha + \gamma) + \alpha$ となる。一方、これまで提案されているリカレント2次NNの場合は、学習パラメータは w_{im}^d , w_{im}^s , $v_i^d(0)$ であり、その総数は $rm(r+l) + r$ となる。ここで、 α および β は、 $r \leq 2^\alpha$ および $m \leq 2^\beta$ を満たす最小の自然数なので、RNNおよびRHONNの学習パラメータの総数は2次NNに比べ少なくでき、DESの状態の数が多くなるほどこの傾向は顕著となることがわかる。

以下では、RNNおよびRHONNを用いて3つDESを同定した数値実験結果を示し、両モデルを比較する。最初の例は、状態遷移図がFig. 4で表わされる簡単なDESである。これは'1'のみからなる系列を受理するDESで、状態シンボルの数は1、また $\Sigma = \Delta = \{0, 1\}$ である。2番目の例は、Fig. 5のように状態遷移図が表わされるDESで、(10)*なる系列を受理する。状態シンボルの数は3、また $\Sigma = \Delta = \{0, 1\}$ である。また、3番目の例はFig. 6のように状態遷移図が表わされるDESで、'000'を含まない系列を受理する。状態シンボルの数は4、また $\Sigma = \Delta = \{0, 1\}$ である。

これらのDESの実現あるいは同定のため、§4.1および§4.2で議論したようにRNNあるいはRHONNを構成する。問題設定の仮定より入力シンボルの集合 Σ および出力シンボルの集合 Δ は既知なので、これら3つの例

Table 1: Results of identification of DES by RNN

the number of dynamic neurons	Example 1			Example 2			Example 3		
	the minimum value of			the minimum value of			the minimum value of		
	J	J_1	J_2	J	J_1	J_2	J	J_1	J_2
$N_d = 1$	0.2036	0.01288	19.07						
$N_d = 2$	0.1971	0.01451	18.26	0.2417	0.01112	23.06	0.6266	0.03517	59.15
$N_d = 3$	0.2169	0.01423	20.26	0.2378	0.01208	22.57	0.6293	0.03682	59.25
$N_d = 4$	0.4296	0.03044	39.91	0.2860	0.01078	27.52	0.8281	0.0400	78.81

Table 2: Results of identification of DES by RHONN

the number of dynamic neurons	Example 1			Example 2			Example 3		
	the minimum value of			the minimum value of			the minimum value of		
	J	J_1	J_2	J	J_1	J_2	J	J_1	J_2
$N_d = 1$	0.324	0.02367	30.03						
$N_d = 2$	0.3827	0.02668	35.61	0.8075	0.07141	73.61	0.6211	0.04945	57.17
$N_d = 3$	0.4429	0.02757	41.54	0.7719	0.05963	71.23	0.7317	0.04947	68.22
$N_d = 4$	0.4999	0.0289	47.10	0.8101	0.06173	74.84	0.8297	0.05250	77.72

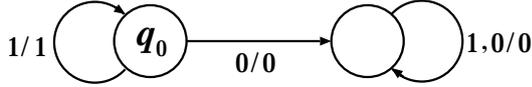


Fig. 4: Example 1: DES accepting the sequences consisting of only '1'

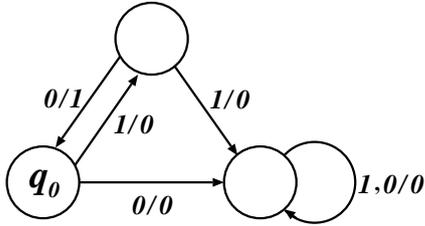


Fig. 5: Example 2: DES accepting the sequence (10)*.

とも $M = \beta = 1$, $L = \gamma = 1$ とすればよい. ところが, 仮定より対象の DES の状態シンボルの集合 Q および初期状態 q_0 は未知なので, 動的ニューロンの個数 $N_d = \alpha$ はあらかじめ決められない. さらに RNN においては静的ニューロンの数 ($N_s = 2^{\alpha+\beta} + \gamma$), また RHONN においては結合ユニットの数 $N_c = 2^n - 1$, ($n = \alpha + \beta$) もあらかじめ決めることはできない (RHONN における静的ニューロンの数は $N_s = \gamma = 1$ とあらかじめ決めることができる). したがって実験においては, 動的ニューロンの数を $N_d = \alpha$ として, α をいくつか変化させて実験を行った (したがって, これに応じて RNN においては静的ニューロンの数 ($N_s = 2^{\alpha+\beta} + \gamma$), また RHONN においては結合ユニットの数 $N_c = 2^n - 1$, ($n = \alpha + \beta$) を変化させている).

また, RNN および RHONN の非線形関数を十分精度よく近似できる微分可能な関数として, $S(x)$ に対しては $\frac{2}{1 - \exp(-ax)} - 1$ を, $H(x)$ に対しては $\frac{1}{1 - \exp(-ax)}$ を用いた. なお, DES \mathcal{M} に対し, 状態数が r の DES は長さ $2r - 1$ のすべての系列に対する出力によって一意に決定できることが知られている. 従って同定用入出力データとしては長さ $2r - 1$ のすべての系列を使用した. また (17) 式の評価関数 J を最小化する方法としては, 共役勾配法を用いた.

以上のような条件で実験した結果を Table 1 と 2 に示す. Table 1 は RNN による結果, Table 2 は RHONN による結果を示し, 動的ニューロン数をいくつか変え

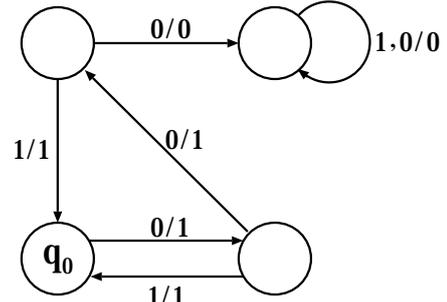


Fig. 6: Example 3: DES accepting the sequences that do not include '000'

た場合に対して得られた J , J_1 , J_2 の値を示している. ただし, $a = 1$, $\delta = 0.01$ とした. これらの表より, いずれの場合においても目的関数が十分小さくできていることが分かる. また以上により得られた NN が, 対象の DES の動作と同じ動作をするかどうかを, 実際に NN を動作させて確認した. その結果, いずれの場合においても, NN の閾値関数の閾値 (RNN の場合 $S(x)$ の 1 と -1 の値, RHONN の場合 $H(x)$ の 1 と 0 の値) に適当な許容誤差を考慮することにより, 対象の DES とまったく同じ動作をすることを確認した. Table 3 と 4 にそれぞれ学習によって得られた RNN と RHONN における閾値の許容誤差の絶対値の最大値を示す. 表中の値は次のことを意味している. たとえば, Table 3 の Example 1, $N_d = 4$ の場合, 得られた RNN の閾値関数を $S(x) = 1 - 0.01910$ for $x \geq 0$, $S(x) = -1 + 0.01910$ for $x < 0$ とすることにより, 対象の DES の動作とまったく同じ動作をすることを意味している. また, Table 4 の Example 1, $N_d = 4$ の場合, 得られた RHONN の閾値関数を $H(x) = 1 - 0.03456$ for $x \geq 0$, $H(x) = 0 + 0.03456$ for $x < 0$ とすることにより, 対象の DES の動作とまったく同じ動作をすることを意味している.

次に, 評価関数として J_2 を付加することの効果を検討した. その結果, J_2 を付加することにより零の値をもつ結合係数がより多く含むスパースなネットワークが実現できることが確認できた. その RNN の場合の結果の例を Table 5 に, RHONN の場合の結果の例を Table 6 に示す. これらの表は, Example 2 の例題に $N_d = 4$ として, 評価関数における J_2 の重み係数 ρ の

Table 3: The maximum absolute values of the tolerances to the threshold values obtained for RNN

	Example 1	Example 2	Example 3
the number of dynamic neurons	the maximal absolute value of the tolerances	the maximal absolute value of the tolerances	the maximal absolute value of the tolerances
$N_d = 1$	0.01652		
$N_d = 2$	0.01899	0.03851	0.05948
$N_d = 3$	0.02226	0.1316	0.05345
$N_d = 4$	0.01910	0.04568	0.03796

Table 4: The maximum absolute values of the tolerances to the threshold values obtained for RHONN

	Example 1	Example 2	Example 3
the number of dynamic neurons	the maximal absolute value of the tolerances	the maximal absolute value of the tolerances	the maximal absolute value of the tolerances
$N_d = 1$	0.02408		
$N_d = 2$	0.02895	0.1618	0.06824
$N_d = 3$	0.03020	0.1458	0.07415
$N_d = 4$	0.03456	0.1339	0.08534

値をいくつか変えて同定実験を行った結果を示したもので、各 ρ の値に対し得られたネットワークの零の値を持つ結合係数の数と閾値関数の許容誤差の絶対値の最大値を示している。これらの表より、評価関数に J_2 を付加しない場合 ($\rho = 0$ の場合) に比べ J_2 を付加した場合 ($\rho > 0$ の場合) は、零の値を持つ結合係数の数を大きく増加させることができ、スパースなネットワークが得られていることが分かる。なお、この表における閾値関数の許容誤差の絶対値の最大値の意味は、Table 3 と 4 の値の意味と同じで、得られた RNN および RHONN における閾値関数の閾値にこれらの誤差を考慮すると、得られたネットワークは対象とする DES とまったく同じ動作をするということを意味している。

Table 5: The number of zero-valued parameters obtained for RNN in Example 2

the value of ρ	the maximal absolute value of the tolerances	the number of zero valued parameters
0	0.000004519	0
0.00001	0.001022	150
0.0001	0.003291	150
0.01	0.1570	150

Table 6: The number of zero-valued parameters obtained for RHONN in Example 2

the value of ρ	the maximal absolute value of the tolerances	the number of zero valued parameters
0	0.000004162	0
0.00001	0.003898	104
0.0001	0.01285	110
0.01	0.1382	104

6 おわりに

本稿では、NN を用いて離散事象システムを実現、同定する方法について議論した。実現、同定のためには、まずどのようなアーキテクチャの NN を用いるかが問題となるが、2種類のアーキテクチャの NN のモデルを導いた。1つは通常の RNN、また他の一つはその結合に高次結合を含む RHONN である。また、導いた NN を用いて DES を実現、同定する方法について議論した。特に、より構造の簡単な (パラメータ数が少ないス

パースな) ネットワークを得るための実現、同定法を示した。

参考文献

- 1) M. L. Minsky: *Computation: Finite and Infinite Machines*, Prentice-Hall, New York (1967).
- 2) N. Alon, A. K. Dewdney and T. J. Ott: Efficient Simulation of Automata by Neural Nets, *Journal of Association for Computing Machinery*, vol.38, No.2, April, 495/514, (1991).
- 3) C. L. Giles, C. B. Miller, D. Chen, H. H. Chen, G. Z. Sun and Y. C. Lee: Learning and Extracting Finite State Automata with Second-Order Recurrent Neural Networks, *Neural Computation*, 4, 393/405 (1992).
- 4) Z. Zegn, R. M. Goodman and P. Smyth: Learning Finite State Machines with Self-Clustering Recurrent Networks, *Neural Computation*, 5, 976/990 (1993).
- 5) C. L. Giles, D. Chen, G. Sun, H. Chen, Y. Lee and W. Goudreau: Constructive Learning of Recurrent Neural Networks: Limitations of Recurrent Cascade Correlation and a Simple Solution, *IEEE Trans. on Neural Networks*, Vol.6, No.4, 829/836 (1995).
- 6) Y. Kuroe: Representation and Identification Method of Finite State Automata by Recurrent High-Order Neural Networks, *ICANN 2005 Proceedings*, W. Duch et al (Eds.), *Lecture Notes in Computer Science*, 3697, 181/190, Springer-Verlag (2005).
- 7) Y. Kuroe and Y. Mori: Neural network models for identification and realization of a class of discrete event systems. *Proc. of IEEE International Conference on Systems, Man, and Cybernetics*, 1363/1369 (2007).
- 8) Y. Kuroe and Y. Mori: Identification of a Class of Discrete Event Systems by Neural Networks - Sparse Realization -, *Proceedings of the 17th IFAC World Congress*, 4618/4624 (2008)
- 9) E. B. Kosmatopoulos, M. M. Polycarpou, M. A. Christodoulou, and P. A. Ioannos, "High-Order Neural Network Structures for Identification of Dynamical Systems," *IEEE Trans. Neural Networks*, vol.6, no.2, 422/431 (1995).
- 10) Y. Kuroe, H. Ikeda and T. Mori, "Identification of Nonlinear Dynamical Systems by Recurrent High-Order Neural Networks," *Proc. of IEEE International Conference on Systems, Man, and Cybernetics*, Vol.1, 70/75 (1997).
- 11) 森禎弘, 黒江康明, 森武宏: 発現パターンに基づく遺伝子ネットワークの学習による設計法, 計測自動制御学会論文集, Vol.44, No.11, 936/945 (2008)
- 12) M. Ishikawa: Structural learning with forgetting, *Neural Networks*, volume 9, No.3, 509/521 (1996).

機械学習を用いたユーザ意図モデリング

○金田祐也 趙強福 渡會恭平 (会津大学)

User Intention Modeling Based on Machine Learning

* Yuya Kaneda, Zhao Qiangfu and Kyohei Watarai (The University of Aizu)

Abstract— Nowadays, internet is becoming more and more popular, and it provides a convenient way for different users to post information. Therefore, enormous information is added everyday. If the user searches information, the result includes unnecessary information not only necessary information. So, it is difficult for users who are not good at searching information to find necessary information. For solving this problem, we make User intention model based on Machine learning. If we use this model for retrieval system, we get good results. In this paper, we try to make this model, and investigate a proposed method that is good or not.

Key Words: User intention modeling, Machine learning, Support vector machine

1 緒言

近年、インターネットの普及に伴い、多くの人々がインターネットを活用するようになった。また、ユーザが自由に、且つ簡単に情報発信を行える為、日々多くの情報が生まれている。それ故、多くの情報が蓄積されている為、ユーザが情報検索を行った際に、たくさんの情報を含んだ検索結果が表示される。この検索結果全てが、ユーザが欲している情報ならば何も問題は無い。しかし、実際にはユーザが予期していないデータが多く含まれている場合がほとんどである。その為、検索した後、さらに検索結果の中から欲しい情報を探し出さなければならぬという手間が発生している。勿論、情報検索に長けた者ならば、さらに検索結果を絞り出す等の方法を取ることができ、容易に情報を探し出す事が出来る。しかしながら、ユーザの多くはそうではない。その為、莫大な情報の中から1つ1つ確認する等をし、探していた情報を見つけ出そうとする。このような方法を取ると、欲しかった情報を見つけ出すまでに長い時間を要する事となってしまう、ユーザにとってはストレスとなる。

検索結果の性能を向上させる為、検索履歴等を用いた方法等は提案されている¹⁾。しかし、検索結果から、ユーザが欲しい情報を抽出する方法として、我々はユーザの好みを判別する事によってデータを抽出する方法を考えた。ユーザの好みを判別できるモデルを使う事により、特にニュース記事の検索時等において、ユーザが欲しい情報を検索する手助けとなるはずである。そこで、テキストマイニングを通して機械学習を用いてユーザの好みを判別するモデルを生成する事を考えた。

機械学習は人工知能の1分野であり、その中でも教師有り学習のモデルを用いた手法がある。ユーザの好みを判別する為、この手法を適用させると良い結果が得られるのではないかと考えた。教師有り学習とは、学習データを与え、データについて学習をしてモデルを生成する。最適な学習データを与える事により、高い性能を示す事が出来る手法である。今回の問題に対しては、ユーザにより、同じ話題についてでも欲する情報が異なるケースがある。その為、ユーザについて学習したモデルを用いる事により、この問題にも対応出来るはずである。

機械学習において、最も性能が良いとされているのがサポートベクターマシン (SVM: Support Vector Machine) である²⁾。SVMは、学習データに対して2クラス間のマージンを最大とする判別境界を求める。判別時に、この



Fig. 1: 文書データから SVM モデルを生成するまで

判別境界を用いてデータを識別する。この手法により、高い汎化性能を示す為、SVMが今回の問題に適していると考えた。

今回、我々はテキストマイニングを用いて、ユーザ意図モデルを生成できるか実験をした。また、より少ない学習データで性能の良いモデルが出来事を期待して、対話的手法を適用させた。

本論文の構成は、以下の通りである。第2章では、基礎知識としてテキストマイニングにおける文書分類とSVMについて説明をする。第3章では、対話的手法を用いたユーザ意図モデリングについて説明をする。第4章では、実験結果と考察を述べる。第5章は、まとめである。

2 基礎知識

2.1 文書分類

ここでは、文書分類について説明する。始めに、文書を単語レベルにまで分解する。そして、次にそれぞれの単語の頻出数を用いて特徴ベクトルを生成し、正規化を行う。最後に、生成された特徴ベクトルによって機械学習モデルを生成する。

2.1.1 形態素解析

日本語の文章は、英語とは異なり単語毎に区別されていない。その為、単語の頻出数を用いて特徴ベクトルを作る為に、日本語の文章を単語毎に分ける必要がある。その際、我々は形態素解析を用いている。形態素解析は、文法と辞書情報を用いて文章を単語に分ける手法である。本論文においては、形態素解析のツールの中でも、有名なMeCab³⁾を使用する。MeCabを使用すると、日本語の文章，“私はシステム知能学講座の一員である”は，“私”，“は”，“システム”，“知能”，“学”，“講座”，“の”，“一員”，“で”，“ある”に分けられる。

2.1.2 単語選別

形態素解析によって、文書からたくさんの単語が切り分けられる。文書の数が多ければ多い程、多種の単語が得られる。しかし、単語の中にはあまり意味を成さない単語もあり、全ての単語を使うと言う事は、冗長である。そこで、我々はテキストマイニングの為に使う単語を品詞で選別している。例えば、先ほどの例に対して“名詞”，“動詞”についてのみ選別するとすると，“私”，“システム”，“知能”，“学”，“講座”，“一員”のみの単語が得られるという事である。

2.1.3 特徴ベクトル

単語選別を経て得られた単語(d個の単語)を使って、特徴ベクトルを生成する。i番目の文書から作られる特徴ベクトルは、以下の様になる。

$$x_i = (w_{i1}, \dots, w_{id})$$

ここで、 w_{ij} はi番目の文書のj番目の単語の頻出数の事である。

また、我々は特徴ベクトルを生成した後、正規化を行って、データを同じ空間で扱う様にしている。正規化については、ユークリッドノルムを用いて行われ、以下の様になる。

$$x_i^{normalized} = (w_{i1}/\|x_i\|, \dots, w_{id}/\|x_i\|)$$

2.2 サポートベクターマシン

サポートベクターマシン(SVM: Support Vector Machine)は、機械学習の手法の1つであり、主に教師有り学習に分類される。SVMは、判別境界を用いてデータを識別する。判別境界を作成する際、学習データのそれぞれのクラスと判別境界とのマージンを最大化させて作成する、マージン最大化という手法を用いて作成する。マージン最大化という手法により、SVMは高い汎化性能を示す⁴⁾。

従来のSVMは、線形問題にのみ高い性能が保証されていた。その為、もし非線形問題を取り扱う時は、性能が著しく低下する可能性があった。非線形問題に適用する為には、高次元空間に写像する事で線形問題に置き換える方法がある。しかし、カーネルトリックという手法を用いる事により、高次元空間に写像する事無く、非線形問題に対して高い性能を示す事が出来る。カーネルトリックは、カーネル関数という関数を用いてSVMを拡張している(非線形SVM, Fig. 2参照)。カーネル関数には、多項式関数や、放射基底関数(RBF: Radial Basis Function)などがある。

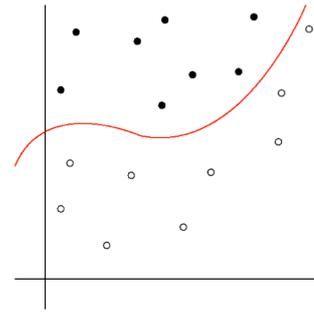


Fig. 2: 非線形 SVM の概念図

3 ユーザ意図モデリング

ユーザ意図モデリングとは、ユーザの意図情報を用いてモデルを生成する事である。ユーザには、趣味や感情など、実に様々な情報がある。これらの情報を元に、ユーザの意図するモデルを生成する。モデルを生成する際、我々は機械学習の手法を用いてモデルを生成している。

学習データとしては、先の章で述べた文書が一例として挙げられる。他には、画像や音声等のメタデータを使って、特徴ベクトルを生成する事により、幅広いデータを使う事が出来る。

3.1 対話的ユーザ意図モデリング

ユーザ意図モデルを生成する際、学習データとなるデータをランダムに選び出して、ラベリングしていた。しかし、実際に実験した所、良い結果が得られなかった。我々が考察した結果、既存のままでは多くの学習データを用意しなければ、良いモデルを生成できないのでは無いかという結論が出た。そこで、我々はより少ないデータでより良いモデルを生成する為に、対話的的手法を用いる事を考えた。従来の方法では、 N_d 個(N_d =学習データの数)のデータ全てについて、ユーザによってラベリングされてからユーザ意図モデルを生成した。しかし、対話的的手法を取り入れる事で、 N_s 個($N_s \ll N_d$)のデータについてラベリングされる毎に、ラ

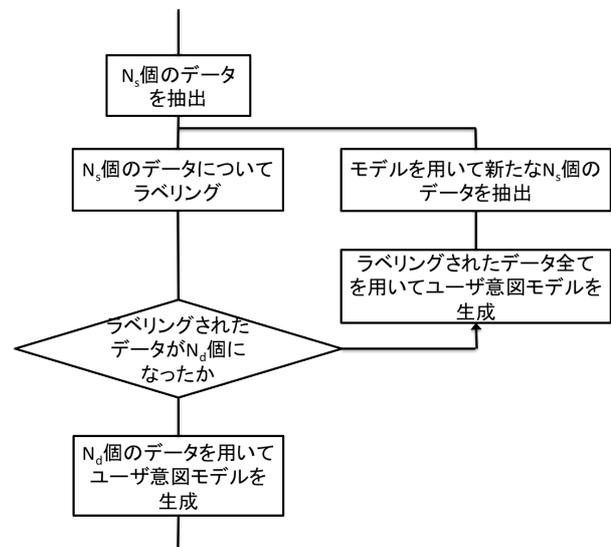


Fig. 3: 対話的ユーザ意図モデリングのプロセス

ベリリングされたデータ全てを用いてモデルを生成し、モデルによってユーザが意図しているであろう、新たな N_s 個の学習データが抽出される。この手順を繰り返し、 N_d 個のデータ全てがラベリングされた時、最終的なユーザ意図モデルを生成する。モデルにより、ユーザの意図したデータのみを抽出するため、今回の対話的手法の場合は、クラスが偏ったデータのみになる可能性もある。しかし、片方のクラスについてのみ判断していく事で、よりユーザの曖昧な部分の判別が出来るはずである。

4 実験

4.1 実験の設定

本稿では、提案した手法の有効性を確認する為、テキストデータベースを用いて実験を行った。使用したデータベースは、ブログ、ニュースの記事を集めた物であり、それぞれのパラメータは Table 1 の通りである。また、SVM については、Joachims による SVM^{Light} を使用し、RBF カーネルを用いた非線形 SVM を用いて実験を行った⁵⁾。提案手法であるユーザ意図モデリングについては、名詞、動詞、形容詞を用いて特徴ベクトルを生成し、10 個($N_s=10$)の文書についてラベリングした際に、新たなテストデータを 10 個抽出し、規定数までラベリングしてもらい実験を行った。規定数はそれぞれのデータベースにおいて 50 個($N_d=50$)とし、

	文書数
Blog	4,813
News	20,125

Table 1: データベースのパラメータ

	Blog (%)				News (%)			
	1 st	2 nd	3 rd	4 th	1 st	2 nd	3 rd	4 th
User 1	60	60	50	80	70	80	80	100
User 2	90	100	50	80	70	100	80	70
User 3	60	50	30	30	70	60	90	80
User 4	60	40	40	10	70	70	70	40
User 5	50	70	80	70	60	50	70	70
User 6	50	40	50	60	70	40	60	50
User 7	100	90	80	70	80	80	90	70
User 8	60	50	70	70	10	30	40	60
User 9	70	30	50	70	90	90	80	30
User10	40	30	10	30	10	20	50	20

Table 2: 実験結果

最初の 10 個の文書については、それぞれランダムに抽出している。ラベリングについては、その文書が好きか嫌いかのどちらかを選んでもらい、好き嫌いをラベルしてユーザ意図モデルを生成し、どのくらいの性能が得られるかを確認する事にした。また、抽出されるデータについては、ユーザが読みたい文書のみを最終的に抽出出来るようにするために、ユーザが好きであると思われるデータを抽出する事とした。これらの条件において、10 名の人に実験を行ってもらった。

4.2 評価方法

3 章で触れたように、 N_s 個のデータについてラベリングする毎に、ランダムに、モデルを用いて好きであると判別された、新しい N_s 個のデータを抽出する。今回、この抽出されたデータが、本当にユーザが好きであったかを評価する。 $N_d=50$ であり、 $N_s=10$ であるため、各データベースについて、実験者 1 人に対して 4 回の評価を行い、少ないデータでも良い結果が得られるかを確認する。精度については、以下の式により評価する。

$$N_c(\text{正解数}) / N_s \times 100$$

4.3 実験結果と考察

実験結果は Table 2 の通りとなった。

最低精度は 10% であり、最高精度は 100% である。全体の特徴として、増加の傾向は見られた。しかし、増加の傾向は見られたが、その中でも 3 つの傾向があった。1 つ目は、徐々に増加して行くケースである。2 つ目は、減少してから増加するケース、そして 3 つ目は、増加した後、減少してしまうケースである。

1 つ目のケースについては、我々が予期していたケースであり、対話的手法が有効的に働いた結果であると考えられる。

2 つ目のケースは、精度が減少した後、また増加した。これは、最初にある程度の精度が得られるモデル

を生成したが、その後学習データが増え、データが複雑になってしまい、精度が減少したのでは無いかと考えられる。他に、本手法においては、先に述べたように、ユーザの好きなデータが多く集まる傾向にある。その為、正しい境界よりも、SVMの判別境界がユーザの嫌いであるデータ側に寄ってしまい、エラー率が増加する要因となっていると考えられる。しかし、その後学習データが増え、複雑になってしまった学習データを、偏ってしまった判別境界を正しく学習し直せた為に、精度が上がり、良いモデルが生成出来たのだと推測できる。

3つ目のケースに関しては、2つ目のケースに見られた、データが複雑になってしまった事、判別境界が偏ってしまった事が要因として考えられる。この場合、もう少しラベリングされたデータが増える事で、精度の向上が見込める。他の原因としては、過学習による機械学習モデルの劣悪化がある。このケースの場合、学習時におけるパラメータを調整するか、または不必要である学習データを削減する必要があると思う。

各実験者の精度を見比べると、人によっては良い結果が得られたが、人によってはあまり良い性能が得られなかった。人によっては、すぐ有効なモデルを生成出来る為であると考えられる。しかし、精度がよく無かった人については、学習データが足りない、または何かしらの処理を加えなければならない可能性がある。人によっては、単語により大きく好き嫌いの影響を受ける可能性がある。その為、それ以外の単語を特徴として扱わなくする事で、有効な学習データを生成する事が出来る可能性がある。または、影響を受けないと思われる特徴について、1つだけの特徴だけでなく、他の特徴の関係等を考慮する事により、一見すると悪影響を与えていた特徴が重要な特徴と成りうる可能性もある。この場合、これらの関係等の情報を持つ特徴を追加する等、高次元化を施す事により、より性能の良いユーザ意図モデルを生成できるのではないかと我々は考えた。今回の実験結果を踏まえ、どうすれば良いモデルを作れるか、さらに実験を重ねる必要があると思う。

5 結言

本稿では、機械学習を用いたユーザ意図モデリングとして、ユーザの好き嫌いを意図情報として用い、且つ対話的手法を用いた方法を提案した。結果から、性能の良いモデルが得られる、または性能が良くなる傾向がある事が分かった。しかし、一部のユーザについては性能があまり良くなく、学習データを増やせば良くなるのか、充分検証する事が、これからの課題の1つである。他に、今回は少ないデータで結果が得られるのかを確認する事が目的であった。その中で、大きく分けて3つの傾向がある事が分かった。この傾向が、もっと多くの学習データがある際にも現れるか、確認していきたい。

また、考察の中で触れたが、学習データの特徴の中に、そのままでは悪影響を及ぼしている可能性のある特徴がある。この特徴について、削るべきか、または特徴の関係などから新たな特徴を生成して有効活用すべきなのか、さらに検証を重ねていきたい。

参考文献

- 1) J. Teevan, ST Dumais and E. Horvitz: “Personalizing Search via Automated Analysis of Interests and Activities”, In SIGIR 2005
- 2) T. Joachims: “Text categorization with support vector machines: Learning with many relevant features”, Proceedings of 10th European Conference on Machine Learning (ECML-98), 1998
- 3) T. Kudo: “MeCab: Yet another part-of-speech and morphological analyzer”, <http://mecab.sourceforge.net/>, 2007
- 4) V. Vapnik: “Statistical Learning Theory”, Wiley, New York, 1998
- 5) T. Joachims: “SVM-Light Support Vector Machine”, <http://svmlight.joachims.org/>, 2008

人体運動の特徴抽出についての一考察

○橋 完太 (工学院大学)

A discussion on feature extraction of human motion

* Kanta Tachibana (Kogakuin University)

Abstract— This paper discusses feature extraction of human motion in daily life through our two progressing works. The first one aims to support successful aging of elder people by e-Rowing. In this study we extract Fourier coefficients from time series force and power data of rowing machine. This helps to compress data and visualize exercise history of the person. The second study aims to reduce physical and mental cost in office. Conventional manipulation of computers with a mouse and a keyboard is not very natural. Toward manipulation with gesture, we identify gestures suitable to computer manipulation. Vector features are extracted from motion data captured with Kinect.

Key Words: complex Fourier coefficients, 3d vector features

1 はじめに

計測機器の高性能化と廉価化に伴い、日常における人体運動を従来よりも高精度かつ頻繁に計測することが可能となった。計測した運動データを利用することで、健康維持・増進に役立てたり、人体にとって自然とは言い難いコンピュータ操作を軽減したりといった応用が可能である。本稿では、家庭や職場での実応用を目指して行っている人体運動の計測を活用する二つの研究について、計測データの特徴抽出に絞って考察する。

一つ目の研究は、ボート漕ぎ運動の力発揮パターンを屋内ローイングマシンで計測して筋力や有酸素能力の変遷を記録し、健康維持・増進に役立てることを目指す。Fig. 1 に示すようなボート漕ぎ運動では、(a)の位置から(b)の位置へ脚伸展と腕屈曲でハンドルを引っ張る。本稿ではこの相をドライブと呼ぶ。(b)の位置から(a)の位置へ戻るときにはハンドルには力がかか

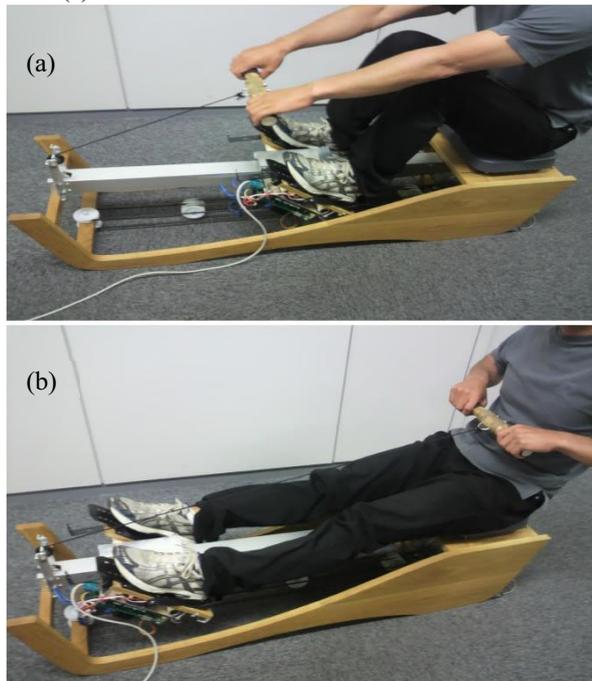


Figure 1 : e-Rowing, an indoor rowing machine

らない。この相はリカバリーと呼ばれる。筆者らが設計・開発を進めている e-Rowing¹⁾は、ロータリエンコーダと発電機を搭載し、足部に対するハンドルの相対位置とハンドルを引く力を計測できる。相対位置と力はどちらも前後方向の1次元で計測する。ボート漕ぎ運動はランニングや自転車漕ぎなどと同様にリズムカルで長時間継続可能であり、また、他のリズムカルな周期的運動に比べて全身の多くの大筋群が用いられる特徴がある²⁾。

Fig. 2 に e-Rowing システムにおける情報の流れを示す。年齢などを考慮して生活習慣、神経系、骨格筋、心肺器官の機能向上を目的とし、主に骨格筋機能と心肺機能の経年変化を可視化するため、ボート漕ぎ運動を計測する。多くの場合ボート漕ぎ運動の周期は2秒から3秒であり、e-Rowing はドライブ中のハンドルの力と位置のデータを16ミリ秒毎に計測するため、一漕ぎで100前後の時点についてデータが観測される。一般に健康のためのボート漕ぎ運動は数10分間継続し、漕ぐ運動を周期的に数100回繰り返すため、一回の運動セッションで数万の時点についてデータが計測される。このような生データをそのまま全て保管すると、データ量が膨大となり、筋力や有酸素能力の経年変化を記録する目的に対しては、明らかに無駄な部分が多い。生データのうち、筋パワーや持久力の運動の特徴を捉えた情報を見出し、その本質的な情報だけを持つデータへ圧縮する必要がある。本稿では周期的データの複素 Fourier 係数への圧縮手法について検討する。圧縮されたデータはデータベースに保存され、必要に応じて運動者本人の過去の運動履歴や同年代の記録を参照したり、それらを加工して視覚的なフィードバックを与えたりし、運動者の意欲を高める。

計測した生データは、16ミリ秒ごとに計測されるため膨大である。また、周期的ではあるが厳密な周期関数ではない。ひと漕ぎ毎の周期や漕ぐ力には振れ幅があり、計測誤差や欠測値も起こりうる。周期的な運動データから特徴を抽出した過去の文献では、石川ら³⁾は、モーションキャプチャで取得した舞踏時の下肢の関節角度の平均、分散、歪度を特徴量とした。吉本ら⁴⁾は、腰部のマーカ位置に対する Fourier 解析も行ったが、振幅のみを使用し位相は使用しなかった。

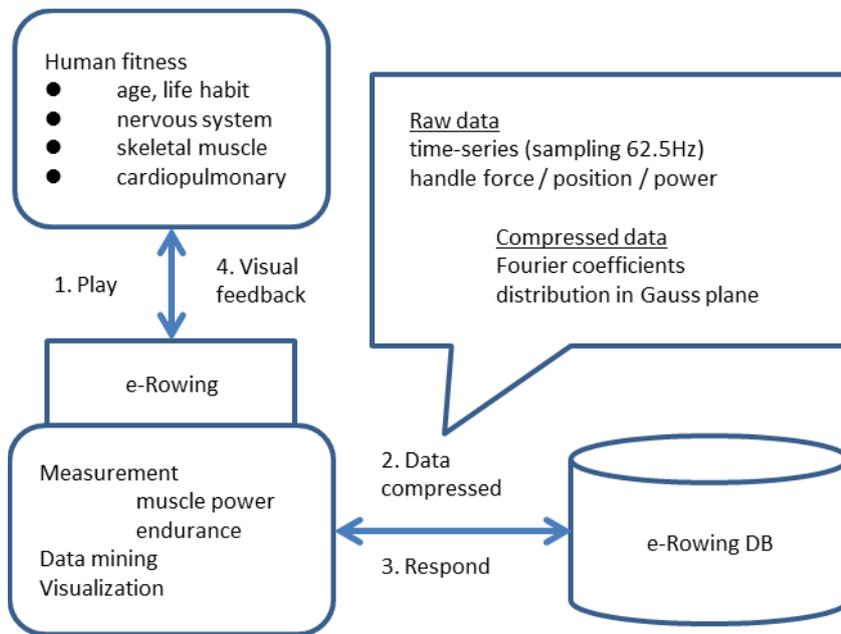


Figure 2 : information flow in e-Rowing system

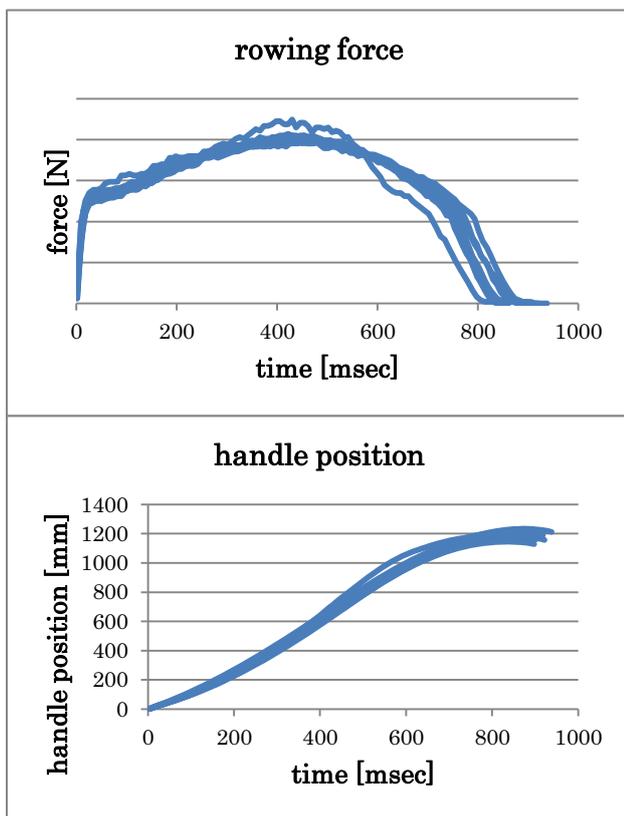


Figure 3 : force and position measured by e-Rowing

二つ目の研究は、ジェスチャによってコンピュータを操作することを目指す。Microsoft社のKinectはRGBカメラと深度センサを搭載し、人体の動作を追跡できる安価なデバイスとして注目されている。Kinectの登場により、コンピュータへのジェスチャ入力が身近なものとなりつつある。Kinectなどモーションキャプチャシステムを用いて、特定のジェスチャを認識する場

合、そのジェスチャに類似した動作が偽陽性(false positive)に誤認識されてしまう可能性がある。作業中、誤認識によって意図しない操作が起こると、作業効率が低下し、使用者へ余計な負担がかかる。そのため、操作を意図した時以外の普段の動作で頻発し誤認識されやすいジェスチャは、コンピュータ操作を行うためのジェスチャとして適切ではない。

この研究では、予め準備した様々なジェスチャに対し、意図したコンピュータ操作以外の普段の動作での起こりやすさ、つまり、誤認識されやすさを評価する。本稿では、その手法について述べる。

2 周期的データの特徴抽出

周期的運動データの特徴抽出法として、過去の研究では舞踏データを対象にモーションキャプチャで取得した関節角度の基本的統計量(平均, 分散, 歪度)を用いる研究³⁾や、身体重心に近い腰部位置をFourier解析する研究⁴⁾があった。しかし、文献4)では振幅のみに着目しており、位相は無視されていた。

モーションキャプチャで関節位置や関節角度を計測するのではなく、本研究ではe-Rowingによって力と位置の情報を計測する。本研究では、これらの周期的な生データを圧縮するため、Fourier変換で導出される複素Fourier係数を利用する。Fig. 3にe-Rowingで計測したボート漕ぎ運動データの例を示す。ある運動者が続けて漕いだ9漕ぎ分のドライブ中の力とハンドル位置をそれぞれ重ねて示す。同一の運動者であるにもかかわらず、力にもハンドル位置にもばらつきがあることが分かる。

これらの計測データから、横軸にハンドル位置[m]を縦軸に力[N]をプロットして時系列につなぐ(Fig. 4上側)と、そのグラフにより囲まれた領域の面積は仕事[J]を表す。このようなグラフは、運動者への視覚的なフィードバックとして有効なもののひとつである。また、ハンドル位置の差分(ハンドル速度)[m/s]と力[N]の積がパワー[W]であり、パワーの時系列(Fig. 4下側)も漕ぎ方の特徴をよく表すと考える。

ひと漕ぎ毎の力、ハンドル位置、パワーの時系列データ $f(t)$ をそれぞれFourier変換し、複素Fourier係数

$$c_n = \frac{a_n - jb_n}{2} = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(t) \exp(-jnt) dt \in \mathbb{Z}$$

を得る。一漕ぎ毎に $\{a_n, b_n \in \mathbb{R} \mid n = 0, 1, \dots, N\}$ を一組得て、一定の回数を漕ぎ終えた後、もしくは、運動セッション終了後、波数 n ごとに、漕いだ回数分だけ複素平面内のプロットを得る。複素Fourier係数の平均とばらつきを圧縮データとしてデータベースに保存しておけば、必要となった際、Fourier逆変換によって平均的な時系列データとそのばらつきを復元できる。

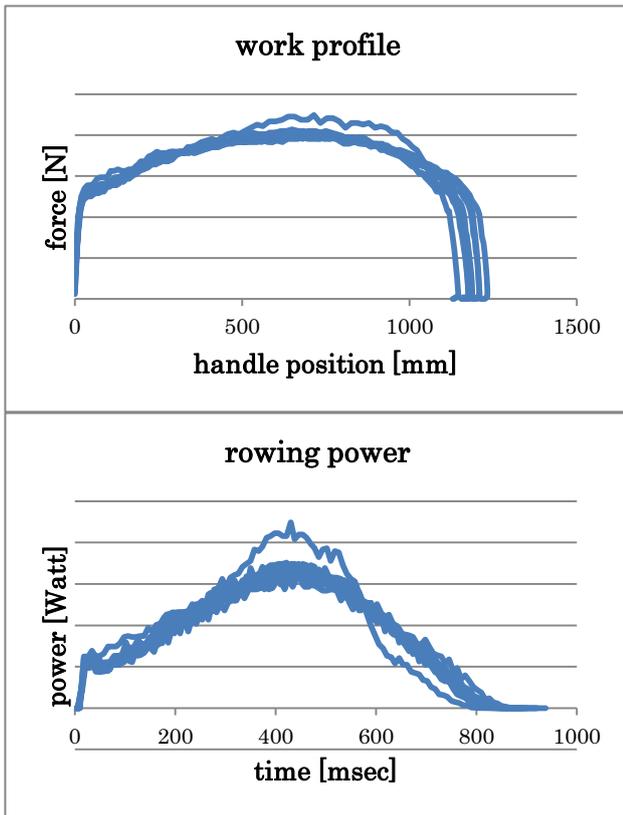


Figure 5 : force profile and rowing power

3 ジェスチャの特徴抽出

Kinectによるモーションキャプチャ

Kinect によって計測される情報は RGB カメラによる二次元のカラー情報と、深度センサによるデプスマップである。Kinect はこれらの情報を統合して、人間が画像内にいれば検出し、その人物の各関節位置座標を3次元座標として得る。

本研究におけるジェスチャ認識には、Kinect で算出された関節座標から、各体節の方向ベクトルを導出し特徴パラメータとして用いる。

デスクワークにおけるジェスチャ認識

本研究ではデスクワークにおけるコンピュータ操作を行うためのジェスチャを対象とする。本研究でデスクワークは、ディスプレイとマウス、キーボードを机の上に設置し、操作者がディスプレイ正面の椅子に座って作業を行うことを意味する(Fig. 5)。

デスクワークにおけるジェスチャでは、机によって隠れる下半身をジェスチャの特徴として用いることはできない。そこで、ジェスチャは全て腕の動きのみで表現することにする。Kinect によって得られる関節位置座標の内、肩からひじに向くベクトルを上腕、ひじから手首に向くベクトルを前腕、手首から指の先に向くベクトルを手の向きとする。片腕の上腕、前腕、手の向きの3本の3次元単位ベクトルを特徴パラメータとしてジェスチャの認識に用いる。

Kinect のカメラが操作者に対して正面に向いていな

い場合もあることを考慮し、カメラの ZX 平面に平行で、両肩を通る直線に垂直な方向を人の正面方向とし、各腕の特徴パラメータはカメラの ZX 平面と人の正面方向を基準とする。

ジェスチャの認識手法

ジェスチャの認識手法に DP マッチング法⁵⁾を用いる。吉本ら⁴⁾は、運動動作の個人性把握のためにモーションキャプチャで取得した関節座標に対して、特定の関節のマークに着目し、DP マッチング法を用いている。DP マッチング法では入力動作の時間軸 t とモデル動作の時間軸 τ を区別する。入力動作の時刻 t における特徴パラメータを u_t ($0 \leq t \leq T_u$) で、モデル動作の時刻 τ における特徴パラメータを z_τ ($0 \leq \tau \leq T_z$) で表す。両者の姿勢の局所距離を以下の $d(t, \tau)$ で表す。

$$d(t, \tau) = \sum_{k=1}^N (u_t(k) - z_\tau(k))^2$$

累積距離 $S(t, \tau)$ を

初期条件 : $S(1, 1) = 2d(1, 1)$

$$\text{漸化式 : } S(t, \tau) = \min \begin{cases} S(t-1, \tau) + d(t, \tau) \\ S(t-1, \tau-1) + 2d(t, \tau) \\ S(t, \tau-1) + d(t, \tau) \end{cases}$$

によって求める。ジェスチャモデル ℓ に対して入力動作のパターン長 T_u とモデル動作のパターン長 T_z によって正規化した評価値 E_ℓ は次のように定義される。

$$E_\ell = \frac{S_\ell(T_u, T_z)}{T_u + T_z}$$

評価値 E_ℓ が閾値 h 以下であり、用意したモデルの中で最小である場合にジェスチャモデル ℓ の動作を行ったと判定する。

連続動作中でジェスチャ動作を認識する際には、ジェスチャの開始時点と終了時点を決めなければならない。村尾ら⁶⁾は通常動作の中には動きの加速度に定常性があり、その定常性が崩れた点を動作開始点と定義している。

ここでは動作開始時点において腕の各体節の加速度が急増すると考え、一定以上の加速度で腕が動くとその時点を実験開始時点として認識を開始する手法を取る。

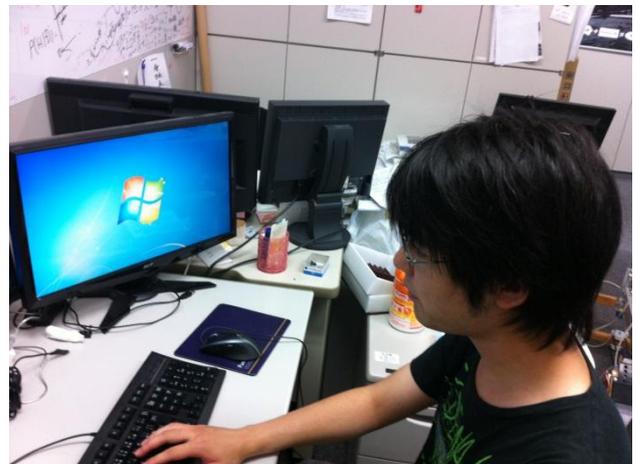


Figure 4 : snapshot of a person in deskwork

認識終了を表す時刻 T_u については、各ジェスチャモデルのパターン終了時間 T_z と入力動作時間 t までの累積距離が一定の閾値以下になった時とする。ただし、入力するジェスチャ動作は1秒以内で行うものとし、時間 t が1秒を超えても認識が終了しない場合は該当のジェスチャモデルと一致しないと判定する。

ジェスチャ認識実験

デスクワークにおけるジェスチャとして Fig. 6 に示す計 10 項目のジェスチャをジェスチャモデルとして定義した。なお、手の片方のみを動作認識対象とする場合、逆の手の動作はそのパラメータは用いない。ここでいう利き手とは、普段マウスを使用する手のことを指す。

- 1: 利き手を前に突き出す
- 2: 利き手を内側から外側へ動かす
- 3: 利き手を外側から内側へ動かす
- 4: 利き手で時計回りに円を描く
- 5: 利き手で反時計回りに円を描く
- 6: 利き手を上から下へ動かす
- 7: 利き手を下から上へ動かす
- 8: 両手をからだの正面で1回叩く
- 9: 利き手で逆の肩を1回叩く
- 10: 利き手で側頭部を1回叩く

実験は各被験者に対して Kinect を設置したコンピュ

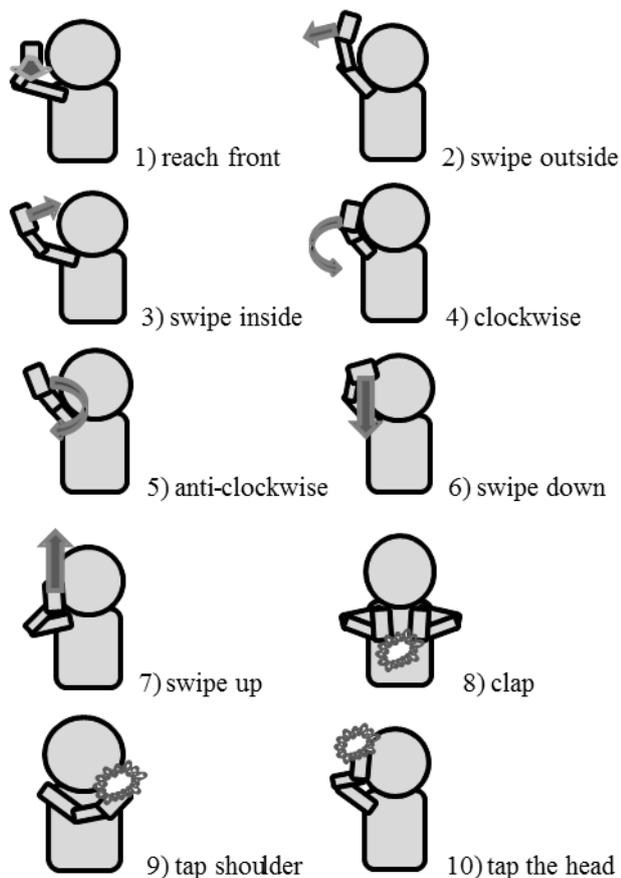


Figure 6 : gestures to be tested

ータ前で普段の作業を 15 分間行ってもらい、その動作があらかじめ設定したジェスチャ動作として誤認識した回数をそれぞれのジェスチャモデルについて計数する。実験を行う上で被験者には普段通りの作業を行ってもらう。

実験後、今度はジェスチャ内容を提示しその動作を各ジェスチャに対し 10 回ずつ行ってもらい、正しくそのジェスチャが認識できるかどうかを調べる。実験結果は研究会当日に発表する。

4 まとめ

本稿では、日常生活での応用を目指し身体運動計測を活用して行っている二つの研究について、特に、特徴抽出に焦点を絞り、その概要を述べた。時系列のデータに対して複素Fourier変換を抽出する方法、および、三次元座標で表されるモーションキャプチャデータに対して、体節の方向ベクトルを抽出する方法を述べた。

参考文献

- 1) 竹内, 樋口, 橋, 堀田 : e-Rowing フェイスブックページ, <http://www.facebook.com/ERowing>, 2012/8/26
- 2) 樋口 : ローイング (ボート漕ぎ) 運動の健康スポーツ科学 体力科学 体力科学 60-1, 78, (2011)
- 3) 石川, 神里, 星野 : 舞踊における身体運動の特徴抽出と印象との関連性: 下肢運動に関する検討 映像情報メディア学会技術報告 映像情報メディア学会技術報告 25-29, 79/84 (2001)
- 4) 吉村, 酒井, 甲斐, 吉村 : 日本舞踊の「振り」部分抽出とその特性の定量化の試み, 電子情報通信学会論文誌. D-II, 情報・システム, II-パターン処理 J84-D-II-12, 2644/2653, (2001)
- 5) 内田 : DP マッチング概説 : 基本と様々な拡張, 電子情報通信学会技術研究報告. PRMU 106-428, 31/36 (2006)
- 6) 村尾, 寺田 : 加速度センサの定常性判定による動作認識手法, 情報処理学会論文誌 52-6, 1968/1979 (2011)

Visual-Lifting Stabilizationに基づく二足歩行の動的形状変更能力

○小林洋祐 見浪護 矢納陽 前場友秀 (岡山大学)

Dynamic Reconfiguration Capability of Bipedal Walking Based on Visual-Lifting Stabilization

*Y. Kobayashi, M. Minami, A. Yanou and T. Maeba (Okayama University)

Abstract— In this paper, we propose a new index of dynamic manipulability for humanoid robot to estimate dynamic ability to change configuration by using remaining redundancy, while prior task is being controlled, e.g., face and eyes being directed to some object. Several indexes have been proposed so far to measure static and dynamical capability of robot manipulator. For example, Dynamic Manipulability (DM) ellipsoid describes a distribution of hand acceleration produced by normalized joint torque. On the other hand, Reconfiguration Manipulability (RM) ellipsoid denotes a distribution of each link velocity produced by joint angular velocity. This paper shows new index of Dynamic Reconfiguration Manipulability (DRM) combined DM and RM, and we have adopted the DRM to a humanoid robot, exhibiting how the DRM indicates directly the configuration-changeability of walking humanoid robot.

Key Words: Dynamic Reconfiguration Manipulability, Humanoid robot, Bipedal walking

1 緒言

ヒューマノイドのダイナミクスは強い非線形性や複雑な干渉を有するため取り扱いが難しい。そのため、線形倒立振り子などへ近似するなど計算を単純化して制御器の設計を容易にすることによって所望の運動を実現することが多い^{1)~8)}。ヒューマノイドの歩行に関する研究も幅広く行われており、その中でも Zero-Moment Point (ZMP) を参照することによる歩行制御は有力かつ現実的な手法であることが知られている。ZMP を支持多角形の凸包内^{9), 10)}に留めようとする制御によって多くのヒューマノイドの実機が現実世界における歩行を達成した。ZMP を用いない手法としては、引き込み領域の内側へ歩行軌跡を留めることに重点を置いた手法^{11)~13)}があり、これは入力トルクを決定するためにリミットサイクルを参照する手法¹⁴⁾を含む。これらの議論において、モデル化や制御器設計は単純化された二足歩行モデルが対象であり、足 (foot) を含むモデルの作成や足の滑りなどが歩行に与える影響の議論は避けられる傾向にある。その一方で、文献¹⁵⁾は足 (foot) の存在によって多様な歩容が生成されることを明確に指摘し、様々な歩容を含む歩行モデルを作成している。また、足が接地している状況は拘束運動としてモデル化が可能であり、文献¹⁷⁾は代数方程式に基づいて物体が環境と接触しながら運動を行っている状態を滑り摩擦を含めて表現し、ヒューマンフィギュアへの応用を提案している。これらの文献と同様に、著者らは可能な限り詳細に導出されたダイナミクスに基づいた議論を行う。本研究と文献^{15), 17)}の観点は共通しているが、各歩容のダイナミクスを状態変数の次元の変化に基づいて導出しているという点が異なっている。つまり、状態変数の次元が運動の結果に応じて変化する(足の拘束状態や路面上の滑り運動に影響を与える)。このような運動の次元が時変である系に対する制御器の設計や安定性の判別を議論している報告はない。

そして、本報では ZMP に依存しない歩行の実現を考える。すなわち、ZMP が支持多角形の境界上に存在

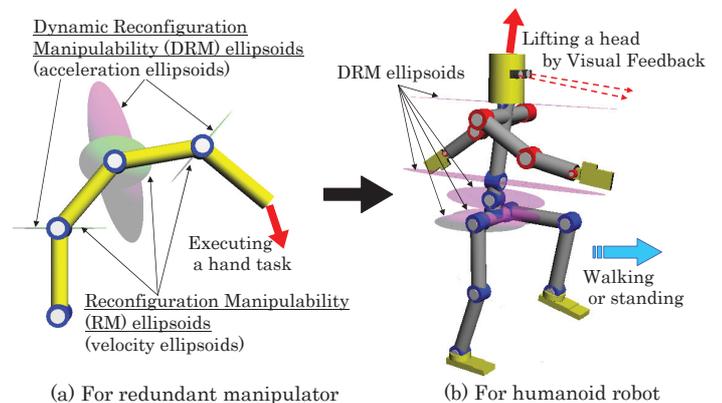


Fig. 1: Applications of dynamic reconfiguration manipulability

し、転倒状態となる歩容を含んだ人間らしい「自然」な歩行である。しかし、ヒューマノイドが転倒状態となっているときの歩容は不安定となる。このような問題に対して、著者らはビジュアルフィードバックによる姿勢安定化戦略をインピーダンス制御²³⁾の概念に基づいて行う。この戦略は文献^{24), 25)}において提案されている“visual pose estimation”を利用しており、ヒューマノイドが固定目標物体を実時間で認識し、その結果得られる頭部の位置/姿勢の目標値と実際の位置/姿勢の偏差をフィードバックすることにより、直立および歩行状態における姿勢安定化を行う。

また人間は、歩行中に頭部の姿勢を固定したまま腰を動かすことができる。つまり、前方にある対象物を見ながら障害物を避けることができる。この人間の振る舞いは、歩行や注視といった第1のタスクを実行した結果から生じる冗長性利用である。それゆえに、本報ではヒューマノイドの動的な冗長性について議論するための概念として動的回避可操作性 (Dynamic Reconfiguration Manipulability, DRM) を提案する。DRMは、マニピュレータの手先にどの程度自由に加速度を発生させることができるかを表す動的可操作性 (Dynamic

Manipulability)²⁶⁾ の概念と冗長マニピュレータの手先に速度タスクが与えられた場合の中間リンクにどの程度自由に速度を発生させることができるかを表す回避可操作性 (Reconfiguration Manipulability)²⁷⁾ の概念を結びつけたものである。この概念により、手先に加速度タスクが与えられた場合に入力トルクによって中間リンクにどの程度自由に加速度を発生させることができるかを指標化できる。この指標をヒューマノイドに適用することにより、頭部の高さを保持した状態で腰部の動的な形状変更のしやすさを表すことができ冗長性利用の目安となる。そこで、本報ではビジュアルフィードバックによる姿勢安定化戦略を用いることで、姿勢安定化や支持多角形の凸包内に ZMP を留めない二足歩行の実現に一定の効果があること、及びビジュアルフィードバックを含む歩行において頭部に与える力の大きさの違いによる動的変更能力の違いについてシミュレーションにより示す。

2 動的回避可操作性

2.1 リンク動的可操作性

リンク動的可操作性楕円体は、マニピュレータのハンドにタスクが与えられていない場合、あるリンクの先端がどの方向にどれだけ加速度を出せるのかを表すものである。各リンクの動きやすさを関節トルクを用いて考える。マニピュレータの運動方程式は一般的に

$$M(q)\ddot{q} + h(q, \dot{q}) + g(q) + D\dot{q} = \tau \quad (1)$$

で表わされる。ここで、 $M(q) \in R^{n \times n}$ は慣性行列、 $h(q, \dot{q}) \in R^n$ は遠心力、コリオリ力を表す項、 $g(q) \in R^n$ は重力を表す項、 $D = \text{diag}[d_1, d_2, \dots, d_n]$ は粘性抵抗行列であり、 $\tau \in R^n$ は入力トルクベクトルである。一方、第 i リンク先端の速度 $\dot{r}_i \in R^m$ と角速度 $\dot{q} \in R^n$ の関係は次式のように表される。

$$\dot{r}_i = J_i(q)\dot{q} \quad (i = 1, 2, \dots, n) \quad (2)$$

ここで、 $J_i(q) \in R^{m \times n}$ は 0 の成分を含んだヤコビ行列 $J_i = [\tilde{J}_i, \mathbf{0}]$ として表される。式 (2) を微分することで、

$$\ddot{r}_i = J_i(q)\ddot{q} + a_{ri}(q, \dot{q}) \quad (3)$$

$$a_{ri}(q, \dot{q}) = \dot{J}_i(q)\dot{q} \quad (4)$$

が得られる。 $a_{ri}(q, \dot{q})$ は q と r_i を表す 2 つの座標系空間の関係が非線形であることに起因する加速度と解釈できる。ここで、 a_{ri} というリンク先端の加速度の一部を関節トルクによって実現できる加速度成分とできない成分に分解すると、

$$\begin{aligned} a_{ri} &= J_i J_i^+ a_{ri} + (I - J_i J_i^+) a_{ri} \\ &= J_i M^{-1} M J_i^+ a_{ri} + (I_n - J_i J_i^+) a_{ri} \end{aligned} \quad (5)$$

が得られる。式 (5) において J_i^+ はヤコビ行列 J_i の擬似逆行列であり、 I_n は $n \times n$ 単位行列である。ここで、

$$\tilde{\tau} \triangleq \tau - h(q, \dot{q}) - g(q) - D\dot{q} + M J_i^+ a_{ri} \quad (6)$$

$$\ddot{r}_i \triangleq \ddot{r}_i - (I_n - J_i J_i^+) a_{ri} \quad (7)$$

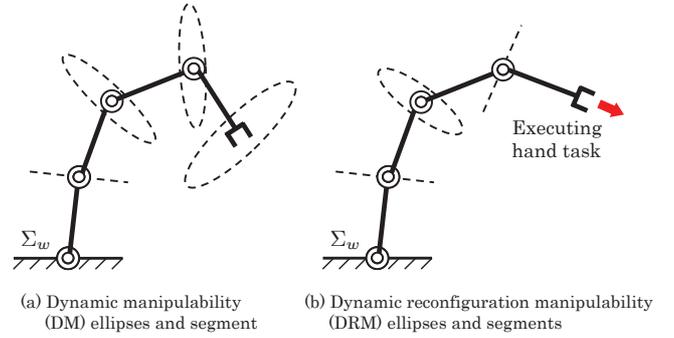


Fig. 2: Dynamic manipulability and dynamic reconfiguration manipulability

によって新たな変数 $\tilde{\tau}$ と \ddot{r}_i を導入すると式 (1), (3), (5), (6), (7) より

$$\ddot{r}_i = J_i M^{-1} \tilde{\tau} \quad (i = 1, 2, \dots, n) \quad (8)$$

という関係が得られる。ここで、ユークリッドノルム $\|\tilde{\tau}\| \leq 1$ と制約をすると、次式のように表せる。

$$\ddot{r}_i^T [J_i (M^T M)^{-1} J_i^T] + \ddot{r}_i \leq 1, \text{ and } \ddot{r}_i \in R(J_i) \quad (9)$$

これは、 $\|\tilde{\tau}\|$ が $\|\tilde{\tau}\| \leq 1$ を満足するような関節トルク $\tilde{\tau}$ を用いて実現し得る各リンクの先端加速度 \ddot{r}_i の全てもとらなる集合を考えると、それは式 (9) で表され、 $J_i(q)$ の値域空間の次元を持つユークリッド空間内の楕円体となる。

2.2 動的回避可操作性

マニピュレータのハンドにタスクが与えられた場合 ($i = n$) の形状変更能力の良し悪しを考える。その場合のリンク動的可操作性楕円体のことを動的回避可操作性楕円体と呼ぶ。ハンド目標加速度 \ddot{r}_{nd} がタスクとして与えられる場合、 \ddot{r}_{nd} を実現するための $\tilde{\tau}$ は式 (8) より次式のように求まる。

$$\ddot{r}_{nd} = J_n M^{-1} \tilde{\tau} \quad (10)$$

式 (10) の $\tilde{\tau}$ について解くと、

$$\begin{aligned} \tilde{\tau} &= (J_n M^{-1})^+ \ddot{r}_{nd} \\ &\quad + [I_n - (J_n M^{-1})^+ (J_n M^{-1})] \mathbf{l} \end{aligned} \quad (11)$$

$\mathbf{l} \in R^n$ は任意ベクトルである。式 (10) の右辺第一項は \ddot{r}_{nd} を実現する $\tilde{\tau}$ の中で $\|\tilde{\tau}\|$ を最小にする解を与える。また第二項は、第一項による \ddot{r}_{nd} の実現には無関係にマニピュレータの形状を変更する関節トルクを与える。 $(J_n M^{-1})^+ \ddot{r}_{nd}$ による形状変化が原因となって障害物との干渉が発生する場合、障害物回避を行いつつ \ddot{r}_{nd} を実現できるかどうかは、任意ベクトル \mathbf{l} の $(J_n M^{-1})$ の零空間への直交射影 $[I_n - (J_n M^{-1})^+ (J_n M^{-1})] \mathbf{l}$ に依存して決まる。

目標タスクの次に優先する動的回避タスクを、第一動的回避サブタスクと呼び、左肩添え字の“1”で表す。冗長自由度が多い場合には、複数のサブタスクを実行できる可能性がある。第一サブタスクが第 i リンクに与えられる場合、その回避要求加速度を ${}^1\ddot{r}_{id}$ で表す。式 (10) と式 (12) より次式を得る。

$$\begin{aligned} {}^1\ddot{r}_{id} &= J_i M^{-1} (J_n M^{-1})^+ \ddot{r}_{nd} \\ &\quad + J_i M^{-1} [I_n - (J_n M^{-1})^+ (J_n M^{-1})] \mathbf{l} \end{aligned} \quad (12)$$

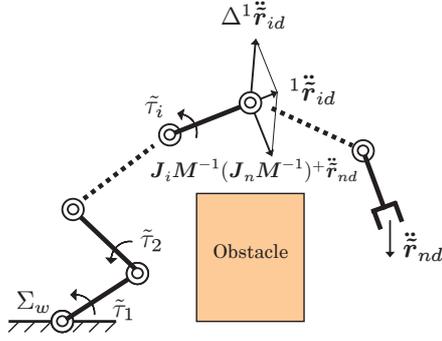


Fig. 3: Obstacle avoidance of intermediate links

ここで、以下のように定義する。

$$\Delta^1 \ddot{\mathbf{r}}_{id} \triangleq \ddot{\mathbf{r}}_{id} - \mathbf{J}_i \mathbf{M}^{-1} (\mathbf{J}_n \mathbf{M}^{-1})^+ \ddot{\mathbf{r}}_{nd} \quad (13)$$

$${}^1 \mathbf{\Lambda}_i \triangleq \mathbf{J}_i \mathbf{M}^{-1} [\mathbf{I}_n - (\mathbf{J}_n \mathbf{M}^{-1})^+ (\mathbf{J}_n \mathbf{M}^{-1})] \quad (14)$$

$\Delta^1 \ddot{\mathbf{r}}_{id}$ を第一動的回避ベクトル, ${}^1 \mathbf{\Lambda}_i$ を第一回避行列と呼ぶ。式 (13) の関係を Fig.3 に示す。 $\ddot{\mathbf{r}}_{nd}$ により第 i リンクに発生する加速度 $\mathbf{J}_i \mathbf{M}^{-1} (\mathbf{J}_n \mathbf{M}^{-1})^+ \ddot{\mathbf{r}}_{nd}$ に対して第一回避要求加速度 $\ddot{\mathbf{r}}_{id}$ を実現するには, $\Delta^1 \ddot{\mathbf{r}}_{id}$ を $\tilde{\tau}_1, \tilde{\tau}_2, \dots, \tilde{\tau}_i$ により発生させる必要がある。そこで, 式 (13) は次式の様に表せる。

$$\Delta^1 \ddot{\mathbf{r}}_{id} = {}^1 \mathbf{\Lambda}_i {}^1 \mathbf{l} \quad (15)$$

${}^1 \mathbf{\Lambda}_i$ は ${}^1 \mathbf{l}$ の射影行列である。 $\ddot{\mathbf{r}}_{nd}$ が与えられるとき, $\Delta^1 \ddot{\mathbf{r}}_{id}$ を通して $\forall \ddot{\mathbf{r}}_{id} \in R^m$ を実現できるかどうかは, ${}^1 \mathbf{\Lambda}_i$ に依存しており, ${}^1 \mathbf{\Lambda}_i$ により $\forall \Delta^1 \ddot{\mathbf{r}}_{id}$ の実現の可能性が判定できる。式 (15) より $\Delta^1 \ddot{\mathbf{r}}_{id}$ を実現する ${}^1 \mathbf{l}$ を求めると次式となる。

$${}^1 \mathbf{l} = {}^1 \mathbf{\Lambda}_i^+ \Delta^1 \ddot{\mathbf{r}}_{id} + (\mathbf{I}_n - {}^1 \mathbf{\Lambda}_i^+ {}^1 \mathbf{\Lambda}_i) {}^2 \mathbf{l} \quad (16)$$

${}^2 \mathbf{l}$ は ${}^1 \mathbf{l}$ と同様に ${}^2 \mathbf{l} \in R^n$ を満たす任意ベクトルである。ここで新たな入力 ${}^1 \mathbf{l}$ が $\|{}^1 \mathbf{l}\| \leq 1$ なる制約のもとに, $\Delta^1 \ddot{\mathbf{r}}_{id}$ が動ける範囲は次式で表される。

$$\Delta^1 \ddot{\mathbf{r}}_{id}^T ({}^1 \mathbf{\Lambda}_i^+)^T {}^1 \mathbf{\Lambda}_i^+ \Delta^1 \ddot{\mathbf{r}}_{id} \leq 1 \quad (17)$$

$\text{rank}({}^1 \mathbf{\Lambda}_i) = m$ のとき, 式 (17) は m 次元の楕円体を表す。そのときの必要十分条件は

$$\Delta^1 \ddot{\mathbf{r}}_{id} = {}^1 \mathbf{\Lambda}_i^+ {}^1 \mathbf{\Lambda}_i^+ \Delta^1 \ddot{\mathbf{r}}_{id}, \quad \Delta^1 \ddot{\mathbf{r}}_{id} \in R^m, \quad (18)$$

が成り立つことである。一方, $\text{rank}({}^1 \mathbf{\Lambda}_i) = p < m$ のとき, $\Delta^1 \ddot{\mathbf{r}}_{id}$ は R^m を満たさないで, その場合は $\Delta^1 \ddot{\mathbf{r}}_{id}$ は $\Delta^1 \ddot{\mathbf{r}}_{id}^*$ として表され, 式 (17) は次のように示される。

$$\begin{aligned} (\Delta^1 \ddot{\mathbf{r}}_{id}^*)^T ({}^1 \mathbf{\Lambda}_i^+)^T {}^1 \mathbf{\Lambda}_i^+ \Delta^1 \ddot{\mathbf{r}}_{id}^* &\leq 1 \\ (\Delta^1 \ddot{\mathbf{r}}_{id}^* = {}^1 \mathbf{\Lambda}_i^+ {}^1 \mathbf{\Lambda}_i^+ \Delta^1 \ddot{\mathbf{r}}_{id}^*) &\quad (19) \end{aligned}$$

式 (19) は r 次元の楕円体である。これらの楕円体は Fig.2(b) のように示される。

2.3 動的回避可操作形状値 (DRMSI)

この章では DRM を比較するための指標について考える。行列 $\mathbf{\Lambda}$ の特異値分解により,

$${}^1 \mathbf{\Lambda}_i = {}^1 \mathbf{U}_i^1 \Sigma_i^1 \mathbf{V}_i^T \quad (20)$$

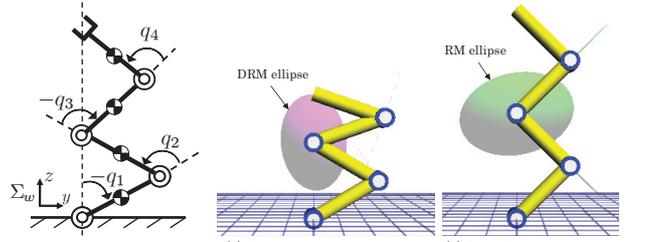


Fig. 4: 4-link manipulator

Fig. 5: Shapes of manipulator

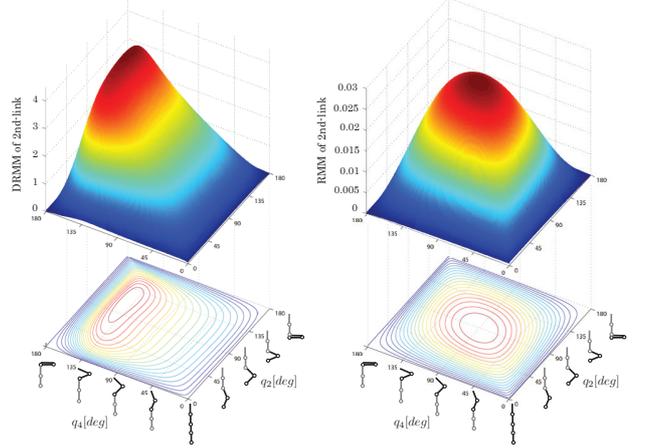


Fig. 6: 2nd-link DRMM and RMM distribution

$${}^1 \Sigma_i = \begin{matrix} & r & n-r \\ r & \begin{bmatrix} {}^1 \sigma_{i1} & & 0 \\ & \ddots & \\ 0 & & {}^1 \sigma_{ir} \end{bmatrix} & \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \\ m-r & \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} & \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \end{matrix} \quad (21)$$

ただし, ${}^1 \mathbf{U} \in R^{m \times m}$, ${}^1 \mathbf{V} \in R^{n \times n}$ は直交行列であり, r は ${}^1 \mathbf{\Lambda}_i$ の特異値に 0 を持たない数であり, かつ $\sigma_{i,1} \geq \dots \geq \sigma_{i,r} > 0$ である。第 i リンクの動的形状変更能力は次の式で表される。

$${}^1 w_i = {}^1 \sigma_{i1} \cdot {}^1 \sigma_{i2} \cdots {}^1 \sigma_{ir} \quad (22)$$

本論文では, 関節トルクによって第 i リンク先端に任意の方向へ加速度を発生できる度合を w_i 動的回避可操作度 (Dynamic Reconfiguration Manipulability Measure, DRMM) として定義する。第 i リンク先端の楕円体の体積は以下のように求められる。

$${}^1 V_{DRi} = c_m \cdot {}^1 w_i \quad (23)$$

$$c_m = \begin{cases} 2(2\pi)^{(m-1)/2} / [1 \cdot 3 \cdots (m-2)m] & (m : \text{odd}) \\ (2\pi)^{m/2} / [2 \cdot 4 \cdots (m-2)m] & (m : \text{even}) \end{cases} \quad (24)$$

マニピュレータ全体のリンクについて動的形状変更能力を考慮するために動的回避可操作形状値 (Dynamic Reconfiguration Manipulability Shape Index, DRMSI) を次のように定義する。

$${}^1 W_{DR} = \sum_{i=1}^{n-1} a_i {}^1 V_{DRi} \quad (25)$$

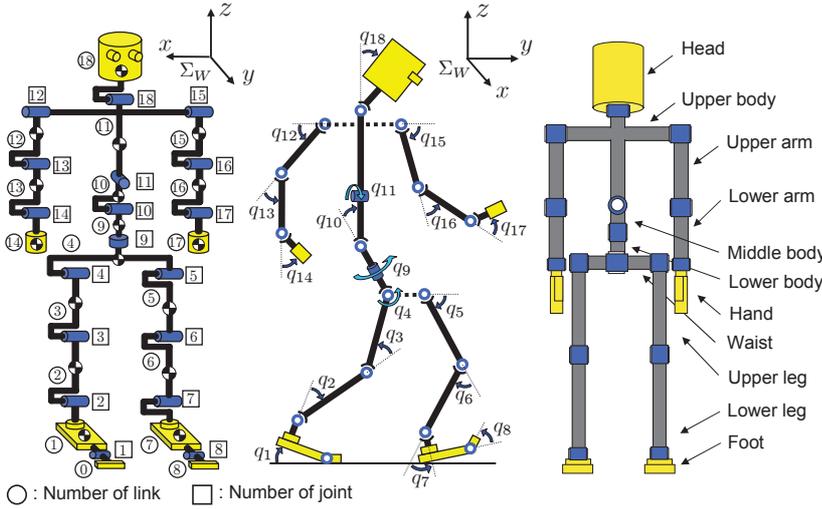


Fig. 7: Definition of humanoid's link, joint and angle number

ちなみに、線分の長さや楕円の面積に比べ楕円体の体積を大きくするため、特異値を100倍した。\$a_i\$はFig. 2(b)のような2次元平面のロボットでは次のように表される。

$$a_1 = a_{n-1} = 1[m^{-1}], a_{2,3,\dots,(n-2)} = 1[m^{-2}] \quad (26)$$

2.4 計算例

この節ではFig. 4の4リンクマニピュレータに対して提案するDRMを適用する。各リンクの長さ、質量はそれぞれ0.3[m], 1.0[kg]とし、粘性抵抗は2.0[N·m·s/rad]とする。このシミュレーションでは、第2リンク先端と手先が常に\$y=0\$にあるものとする。つまり、\$q_2\$と\$q_4\$が与えられたときに\$q_1 = -q_2/2.0\$, \$q_3 = -(q_2+q_4)/2.0\$となる。Fig. 5(a), (b)はFig. 6においてDRMMまたは回避可操作度RMMが最大値となったときのマニピュレータ形状と、それぞれの形状に対するDRMEとRMEを示したものである。Fig. 6において、第2リンクのRMMは\$q_2 = 90^\circ\$, \$q_4 = 90^\circ\$のとき最大となり、DRMMは\$q_2 = 118^\circ\$, \$q_4 = 141^\circ\$のとき最大となっている。

3 ヒューマノイドの動力学モデル

ヒューマノイドの関節構成はFig. 7のようになっており、文献¹⁸⁾を基に寸法(リンクの長さ\$l_i\$ [m]・リンクの質量\$m_i\$ [kg]・関節の粘性摩擦係数\$d_i\$ [N·m·s/rad])はTable 1の値とする。本報では、文献¹⁹⁾と同様にヒューマノイドを直列リンク接続のマニピュレータと見なして、Newton-Euler法^{20), 21)}を利用したダイナミクスの数値計算を行った。しかし、Fig. 7で示すモデルは分岐を持つという点において文献¹⁹⁾と異なっている。本報のモデルはつま先を含んだ足、腕、胴体などから構成される18自由度の剛体モデルである。脚がサジタル面に限定されながらも、その平らな足裏と蹴り力によって様々な歩容が現れる。本論文では、リンク1とリンク2を含んだ脚を支持脚、リンク7とリンク8を含んだ脚を遊脚または接地脚と呼ぶ。

3.1 片足接地モデル

片足接地の運動方程式は式(1)と同様である。ここで、支持脚にすべりが発生しない面接地であれば関節角度は\$\mathbf{q} = [q_2, q_3, \dots, q_{18}]^T\$のようになることを考える。この歩

Table 1: Physical parameters

Link	\$l_i\$	\$m_i\$	\$d_i\$
Head	0.24	4.5	0.5
Upper body	0.41	21.5	10.0
Middle body	0.1	2.0	10.0
Lower body	0.1	2.0	10.0
Upper arm	0.31	2.3	0.03
Lower arm	0.24	1.4	1.0
Hand	0.18	0.4	2.0
Waist	0.27	2.0	10.0
Upper leg	0.38	7.3	10.0
Lower leg	0.40	3.4	10.0
Foot	0.07	1.1	10.0
Total	1.7	63.8	

容はFig. 8 (I)に示されている。遊脚が接地する前に支持脚のかかどが地面から離れたときはFig. 8 (II)のようになり変数\$q_1\$が\$\mathbf{q}\$に加えられ、\$\mathbf{q} = [q_1, q_2, \dots, q_{18}]^T\$として表される。この状態でのつま先のトルク\$\tau_1\$は常に0である。つまり、非ホロノミック拘束であり、転倒状態である。

3.2 拘束状態を含んだモデル

遊脚が地面に接触した場合、Fig. 8 (III), (IV)のような状態が現れる。これは地面に対する接地脚の位置\$z_h\$または角度\$q_e\$が拘束されていることを示す。足は垂直方向にのみ拘束されるので\$y\$方向への運動をもつ。つまり、接地した場合の足の速度に依存して接地脚は前または後ろ方向に滑る。足の位置拘束と角度拘束は\$C_1, C_2\$としてそれぞれ定義する。これらの拘束は接地脚のかかともしくはつま先の位置を意味する\$\mathbf{r}(\mathbf{q})\$を含む式(27)によって示される。

$$\mathbf{C}(\mathbf{r}(\mathbf{q})) = \begin{bmatrix} C_1(\mathbf{r}(\mathbf{q})) \\ C_2(\mathbf{r}(\mathbf{q})) \\ C_3(\mathbf{r}(\mathbf{q})) \end{bmatrix} = \mathbf{0} \quad (27)$$

ここで、\$C_3\$は接地脚が滑らないための拘束(\$y_h\$が一定)であり、Fig. 8(III), (IV)に示す。この拘束は足の滑り速度\$\dot{y}_h\$が十分小さい値\$\varepsilon\$よりも小さくなった場合\$|\dot{y}_h| < \varepsilon\$に起こる。これは、摩擦力が動摩擦から静止摩擦力に変わったことを意味する。一方、接地脚に作用する力\$f_y\$が\$f_t\$を超える、つまり\$|f_y| > |f_t|\$となると静止摩擦力から動摩擦力に変わり、滑り始める。そのとき、外力\$f_{n_z}, f_{n_y}\$、摩擦力\$f_t\$と外トルク\$\tau_n\$を含んだロボットの運動方程式は\$C_1, C_2, C_3\$に対応して次のように表される。

$$\begin{aligned} \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) + \mathbf{D}\dot{\mathbf{q}} \\ = \boldsymbol{\tau} + \mathbf{j}_{c_z}^T f_{n_z} - \mathbf{j}_t^T f_t + \mathbf{j}_r^T \tau_n + \mathbf{j}_{c_y}^T f_{n_y} \end{aligned} \quad (28)$$

ここで\$\mathbf{j}_{c_z}, \mathbf{j}_t, \mathbf{j}_r, \mathbf{j}_{c_y}\$は次のように定義される。

$$\begin{aligned} \mathbf{j}_{c_z}^T &= \left(\frac{\partial C_1}{\partial \mathbf{q}^T} \right)^T \left(1 / \left\| \frac{\partial C_1}{\partial \mathbf{r}^T} \right\| \right), \quad \mathbf{j}_t^T = \left(\frac{\partial \mathbf{r}}{\partial \mathbf{q}^T} \right)^T \frac{\dot{\mathbf{r}}}{\|\dot{\mathbf{r}}\|} \\ \mathbf{j}_r^T &= \left(\frac{\partial C_2}{\partial \mathbf{q}^T} \right)^T \left(1 / \left\| \frac{\partial C_2}{\partial \mathbf{q}^T} \right\| \right), \quad \mathbf{j}_{c_y}^T = \left(\frac{\partial C_3}{\partial \mathbf{q}^T} \right)^T \left(1 / \left\| \frac{\partial C_3}{\partial \mathbf{r}^T} \right\| \right) \end{aligned}$$

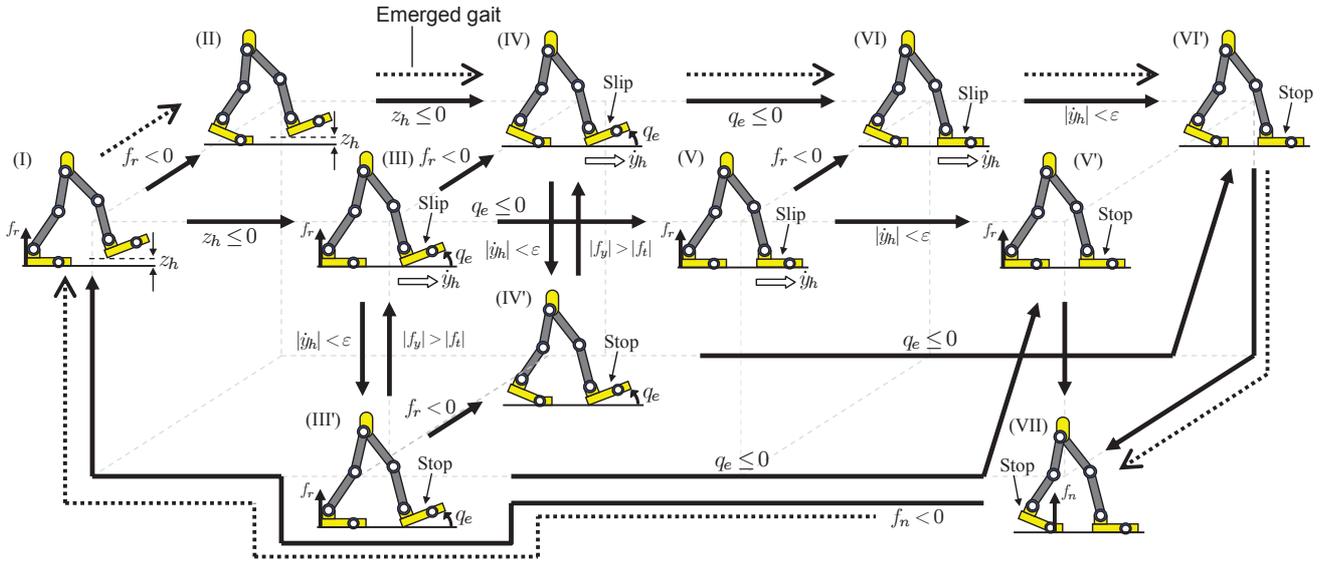


Fig. 8: State, gait's transition and emerged walking gait

Table 2: Possible states for humanoid's walking

State	State variables	Constraint
(I)	$\mathbf{q} = [q_2, q_3, \dots, q_{18}]^T$	Nothing
(II)	$\mathbf{q} = [q_1, q_2, \dots, q_{18}]^T$	Nothing
(III)	$\mathbf{q} = [q_2, q_3, \dots, q_{18}]^T, f_{n_z}$	$C_1 = 0$
(III')	$\mathbf{q} = [q_2, q_3, \dots, q_{18}]^T, f_{n_z}, f_{n_y}$	$C_1, C_3 = 0$
(IV)	$\mathbf{q} = [q_1, q_2, \dots, q_{18}]^T, f_{n_z}$	$C_1 = 0$
(IV')	$\mathbf{q} = [q_1, q_2, \dots, q_{18}]^T, f_{n_z}, f_{n_y}$	$C_1, C_3 = 0$
(V)	$\mathbf{q} = [q_2, q_3, \dots, q_{18}]^T, f_{n_z}, \tau_n$	$C_1, C_2 = 0$
(V')	$\mathbf{q} = [q_2, q_3, \dots, q_{18}]^T, f_{n_z}, \tau_n, f_{n_y}$	$C_1, C_2, C_3 = 0$
(VI)	$\mathbf{q} = [q_1, q_2, \dots, q_{18}]^T, f_{n_z}, \tau_n$	$C_1, C_2 = 0$
(VI')	$\mathbf{q} = [q_1, q_2, \dots, q_{18}]^T, f_{n_z}, \tau_n, f_{n_y}$	$C_1, C_2, C_3 = 0$
(VII)	$\mathbf{q} = [q_2, q_3, \dots, q_{18}]^T, f_{n_z}, f_{n_y}$	$C_1, C_3 = 0$

次の2点は簡単な物理法則である。(i) f_{n_z} (地面に対する拘束力), f_t (地面に対する摩擦力) が直交している。(ii) f_t の値が f_{n_z} に依存している $f_t = K f_{n_z}$ (K は定数: $0 < K \leq 1$). さらに, 式 (27) を時間 t で2階微分すると $\ddot{\mathbf{q}}$ についての拘束式が得られる。

$$\left(\frac{\partial C_i}{\partial \mathbf{q}^T} \right) \ddot{\mathbf{q}} + \dot{\mathbf{q}}^T \left[\frac{\partial}{\partial \mathbf{q}} \left(\frac{\partial C_i}{\partial \mathbf{q}^T} \right) \right] \dot{\mathbf{q}} = 0 \quad (i = 1, 2, 3) \quad (29)$$

式 (27) の拘束の下, 式 (28), (29) での $\ddot{\mathbf{q}}$ は同一の時間で一致する。そして接触が続いている間, $\ddot{\mathbf{q}}, f_{n_z}, \tau_n, f_{n_y}$ が維持される¹⁷⁾。Fig. 8 に示すように接触パターンが状態変数の次元の変化を作り, 次元の変化が接触パターンを変える。本報では, Fig. 8 に示す歩行中に起こりうる11種類の歩容を想定し, 条件に応じて切り替えている。これらの全ての歩容は式 (28), (29) のダイナミクスにおいて表現可能である。それぞれの歩容変数と次元の組み合わせを Table 2 に示す。

3.3 衝突

遊脚が地面に接触したとき衝突を考慮する必要がある。ここでは, 完全非弾性衝突によって表される衝突

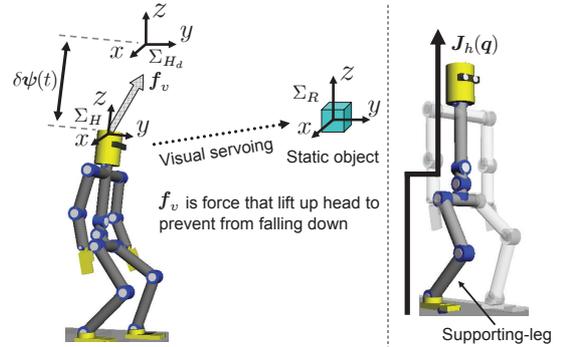


Fig. 10: Concept of Visual Lifting Stabilization

15) であると想定する。

4 Visual Lifting Stabilization

4.1 フィードバックによるトルク生成

本節では, 視覚情報に基づくフィードバックを用いたヒューマノイドの姿勢安定化戦略 (Fig. 10) を提案する。まず, ヒューマノイドの頭部に固定された座標系 Σ_H に対する固定目標物体 Σ_R の位置/姿勢をビジュアルサーボにより計測する。この両者の関係を同次変換行列 ${}^H\mathbf{T}_R$ として定義する。そして, 目標物体と任意に与えた頭部の目標値 Σ_{Hd} の関係 ${}^R\mathbf{T}_{Hd}$ を用いることによって, Σ_{Hd} と Σ_H の偏差 ${}^H\mathbf{T}_{Hd}$ が以下の式によって計算できる。

$${}^H\mathbf{T}_{Hd}(\psi_d(t), \psi(t)) = {}^H\mathbf{T}_R(\psi(t)) \cdot {}^{Hd}\mathbf{T}_R^{-1}(\psi_d(t)) \quad (30)$$

ここで, ${}^H\mathbf{T}_R$ は文献^{24), 25)} で提案されている手法によって計測可能であるが, 本報では ${}^H\mathbf{T}_R$ を画像認識から得るのではなく, 真値が計測されたことを仮定している。そして, $\delta\psi(t)$ を最小とするために, 関節に入力すべきトルクを以下の式で与える。

$$\boldsymbol{\tau}_h(t) = \mathbf{J}_h(\mathbf{q})^T \mathbf{K}_p \delta\psi(t) \quad (31)$$

ここで, $\mathbf{J}_h(\mathbf{q})$ は Fig. 10 に示すように支持脚から頭部までのヤコビ行列, \mathbf{K}_p は比例ゲインを表している。このトルクによって頭部が目標値に引っ張り上げられ, 頭部・重心位置の低下や転倒を防止する。ここで, $\boldsymbol{\tau}_h(t)$

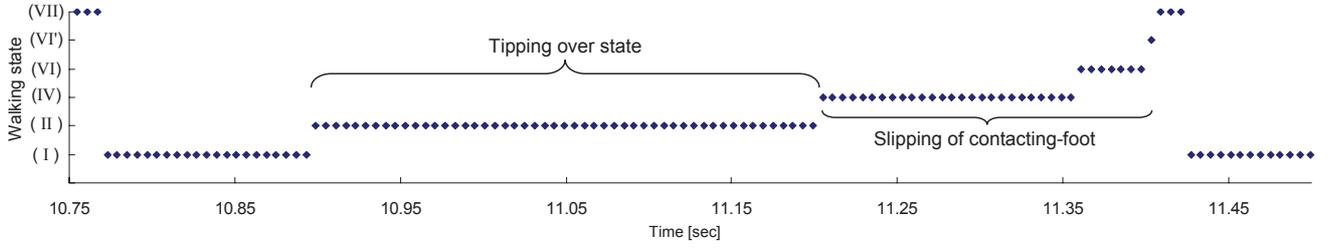


Fig. 9: State transition of walking in one step

中の支持脚つま先のトルクは0とし、つま先立ち状態では転倒状態を意味する。遊脚が前方で接地しても接地脚の進行方向の滑りが増大する場合には転倒する。

4.2 フィードフォワードによるトルク生成

$\tau_h(t)$ の入力トルク以外に 2 つのトルクを加える。遊脚 (joint-5) を前に振り出すために $\tau_t(t) = [0, \dots, 0, \tau_{t5}, 0, \dots, 0]^T$ を与え、ロール角への回転を生じるために腰関節 (joint-11) へ $\tau_w(t) = [0, \dots, 0, \tau_{w11}, 0, \dots, 0]^T$ を与える。 τ_{t5} と τ_{w11} は人間の歩行周期に近くなるよう次のように設定した。

$$\tau_{t5} = 20 \cos \{2\pi(t - t_1)/1.85\}, \quad (32)$$

$$\tau_{w11} = \begin{cases} 50 \sin \{2\pi(t - t_1)/1.85\} & (\text{if Right leg}) \\ -50 \sin \{2\pi(t - t_1)/1.85\} & (\text{if Left leg}) \end{cases} \quad (33)$$

ここで t_1 は、状態 (V) または状態 (VII) から状態 (VII) へ歩容が遷移し、支持脚と接地脚が入れ替わる時刻である。

4.3 統合されたコントローラ

式 (31), (32) と (33) によって表されるトルク入力を 1 つにまとめて、 $\tau(t) = \tau_h(t) + \tau_t(t) + \tau_w(t)$ をコントローラとする。

5 二足歩行の例

シミュレーション環境はプログラム作成のため “Borland C++ Builder Professional Ver. 5.0” を用い、表示には “OpenGL Ver. 1.5.0” を用いた。尚、サンプリングタイムは 3.0×10^{-3} [sec]、地面の摩擦係数は $f_t = 0.7f_{n_z}$ である。

5.1 人間らしい安定な歩行例

この章では、安定な歩行を実現するために試行錯誤的に比例ゲイン $\mathbf{K}_p = \text{diag}[20, 290, 1100]$ と設定した。そのときの歩行の様子を Fig. 13(a) に示す。歩行結果は、平均歩幅 0.43[m] であり、平均時速 2.15 [km/h] である。ヒューマノイドのダイナミクスによって発生する歩容の遷移を Fig. 9 に示す。本シミュレーションではヒューマノイドは Fig. 8 の “Emerged gait” に示す状態遷移 (I) \rightarrow (II) \rightarrow (IV) \rightarrow (VI) \rightarrow (VI') \rightarrow (VII) \rightarrow (I) \rightarrow ... のように歩行した。この遷移は Fig. 8 に示す全ての起こりうる状態の中から閉ループのダイナミクスによって選択されたものである。ヒューマノイドの首部分 (link-18 の起点) の軌跡を 3 次元表示したものを Fig. 11 に示す。ただし、過渡応答部分と考えられる 20 歩目までを除いており、地面との衝突による影響は軌道上に表れているが、定常状態においてはリ

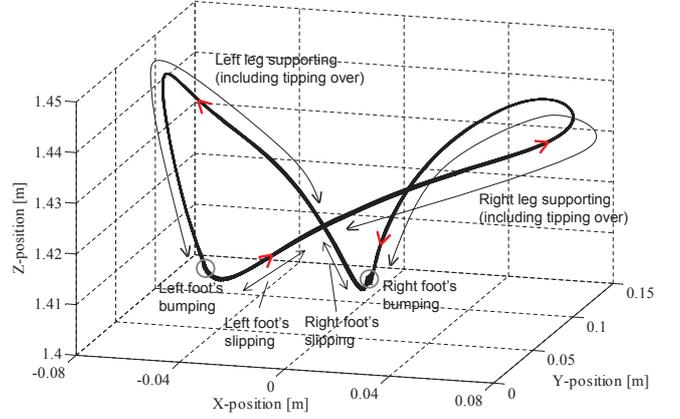


Fig. 11: Emerged limit cycle [(x, y, z) trajectory of neck]

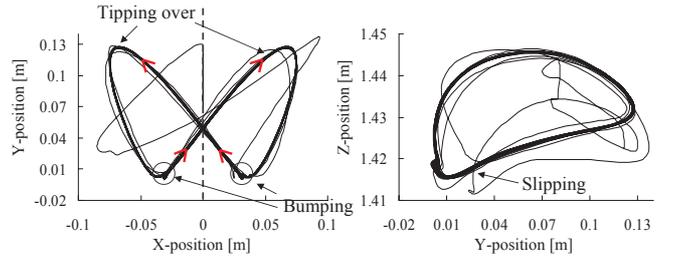


Fig. 12: Motion trajectory of neck

ミットサイクルに収束したと考えられる。そこで、初期形状から 200 歩目までの首の軌跡を x - y 平面と y - z 平面に分けたものを Fig. 12 に示す。左側のグラフでは、破線から右側と左側が線対称になっており、上半身のロール方向への回転により首と肩が y 軸方向にも振動していることが見て取れる。右側のグラフは y - z 平面での首の軌跡であるが、衝突を含む歩容によって首は上下方向に加えて前後方向に振動していることが分かる。これらの図はビジュアルフィードバックによって、転倒状態、衝突、滑りを含む歩行が安定化されていることを意味している。

5.2 動的回避可操作性に基づく分析

この節では 3 種類の比例ゲインによる歩行の違いを動的回避可操作性を適用して考察する。高ゲイン ($\mathbf{K}_p = \text{diag}[20, 290, 1100]$)、中ゲイン ($\mathbf{K}_p = \text{diag}[20, 290, 950]$)、低ゲイン ($\mathbf{K}_p = \text{diag}[20, 290, 900]$) とする。ヒューマノイドは Fig. 13(a), (b), (c) のようにそれぞれ歩行した。楕円体は見やすさのためにスケールアップしている。Fig. 13 において、(a)-1, (b)-1 (c)-1 では支持脚が面接地しているため膝関節に DRM 楕円体が見られない (厳密には線分は存在している)。つまり、変数の次元は $\mathbf{q} = [q_2, q_3, \dots, q_{18}]^T$ となっている。それ以外では、点

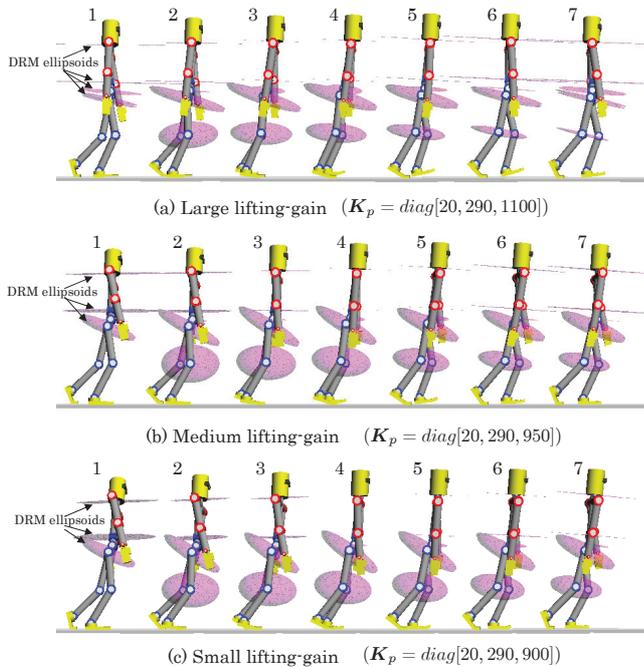


Fig. 13: Screen-shot of bipedal walking with DRM ellipsoids

接地であり $\mathbf{q} = [q_1, q_2, \dots, q_{18}]^T$ となっているため膝関節にも楕円体が見られる（厳密には楕円である）。また、13(c)の形状は(a)と比べ腰を落とした形状であるので、DRM楕円体の体積は(c)の方が大きい。Fig. 14とFig.16からも、比例ゲインが大きいほど楕円体の体積の総和を表したDRMSIが小さくなっていることが確認できる。これは、(c)の形状変更能力が(a)の形状より大きいことを意味しているが、(c)の歩行は人間らしい歩行とは言い難い。Fig.15とFig.17は回避可操作性楕円体の体積の総和を表すRMSIを表しているが、比例ゲインが大きいほど一歩におけるRMSIの最大-最小の幅が大きくなっている。

6 結言

本論文では、冗長性利用のための動的な形状変更能力を評価する指標（動的回避可操作性）を提案し、4リンクマニピュレータでの簡単な数値例を示し、ヒューマノイドロボットの二足歩行に適用した。滑り動作を含む二足歩行の状態遷移は7種類の歩容で記述され、それぞれの状態でのダイナミクスはN.E.法によって計算された。そして、Visual Lifting Stabilizationがヒューマノイドの姿勢安定化に一定の効果を示すことを確認した。歩行結果に表れた歩容遷移はヒューマノイドの動作に依存して選択された。さらに、人間のように腰の位置を高く保つ歩行よりも、腰を落とした二足歩行の方が高い動的形状変更能力を持っていることが確認できた。

参考文献

- 1) S. Kajita, M. Morisawa, K. Miura, S. Nakaoka, K. Harada, K. Kaneko, F. Kanehiro and K. Yokoi: Biped Walking Stabilization Based on Linear Inverted Pendulum Tracking, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 4489/4496 (2010)
- 2) H. Dau, C. Chew and A. Poo: Proposal of Augmented Linear Inverted Pendulum Model for Bipedal

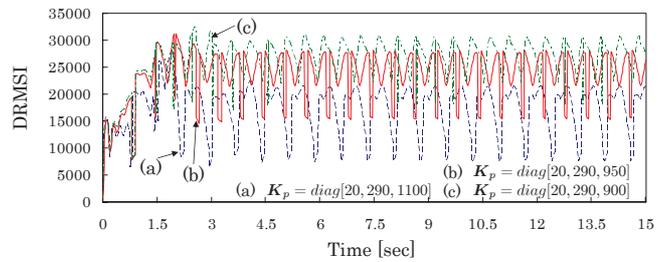


Fig. 14: DRMSI until $t = 15.0[s]$

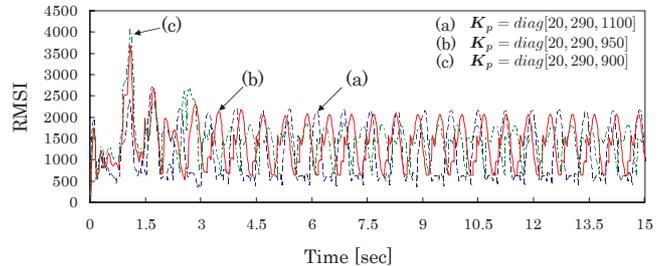


Fig. 15: RMSI until $t = 15.0[s]$

- Gait Planning, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 172/177 (2010)
- 3) S. Kajita and K. Tani: Study of Dynamic Locomotion on Rigid Terrain-Derivation and Application of the Linear Inverted Pendulum Mode, *Proceedings of IEEE International Conference on Robotics and Automation*, 1405/1411 (1991)
 - 4) S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi and H. Hirikawa: The 3D Linear Inverted Pendulum Mode: A simple modeling for a biped walking pattern generation, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 239/246, (2001)
 - 5) J.H. Park and K.D. Kim: Biped walking robot using gravity-compensated inverted pendulum mode and computed torque control, *Proceedings of IEEE International Conference on Robotics and Automation*, Vol. 4, 3528/3593 (1998)
 - 6) P.B. Wieber: Trajectory free linear model predictive control for stable walking in the presence of strong perturbations, *Proceedings of International Conference on Humanoid Robotics*, 137/142, (2006)
 - 7) P.B. Wieber: Viability and predictive control for safe locomotion, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1103/1108, (2008)
 - 8) A. Herdt, N. Perrin and P.B. Wieber: Walking without thinking about it, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 190/195 (2010)
 - 9) M. Vukobratovic, A. Frank and D. Juricic: On the Stability of Biped Locomotion, *IEEE Transactions on Biomedical Engineering*, Vol. 17, No. 1, 25/36, (1970)
 - 10) M. Vukobratovic and J. Stepanenko: On the Stability of Anthropomorphic Systems, *Mathematical Biosciences*, Vol. 15, 1/37, (1972)
 - 11) S. Collins, A. Ruina, R. Tedrake and M. Wisse: Efficient Bipedal Robots Based on Passive-Dynamic Walkers, *Science*, Vol. 307, 1082/1085 (2005)
 - 12) J. Pratt, P. Dilworth and G. Pratt: Virtual Model Control of a Bipedal Walking Robot, *Proceedings of IEEE International Conference on Robotics and Automation*, 193/198 (1997)
 - 13) R.E. Westervelt, W.J. Grizzle and E.D. Koditschek: Hybrid Zero Dynamics of Planar Biped Walkers, *IEEE Transactions on Automatic Control*, Vol. 48, No. 1, 42/56 (2003)

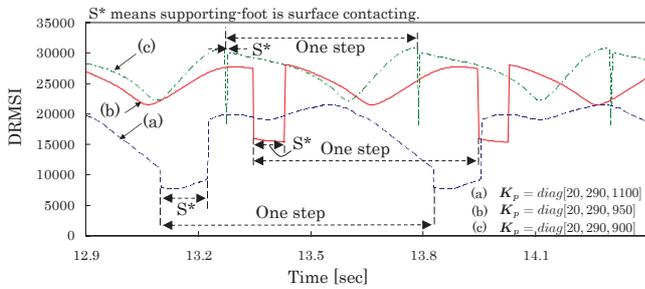


Fig. 16: DRMSI around $t = 13.5[s]$

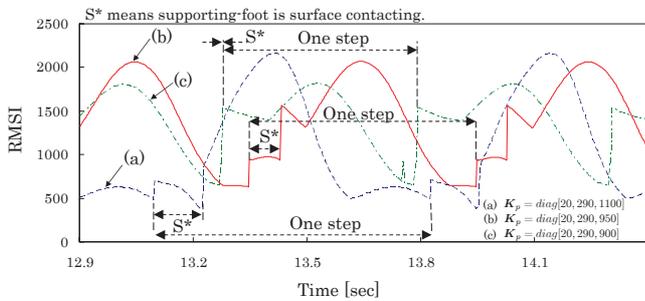


Fig. 17: RMSI around $t = 13.5[s]$

- 14) Y. Harada, J. Takahashi, D. Nenchev and D. Sato : Limit Cycle Based Walk of a Powered 7DOF 3D Biped with Flat Feet, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 3623/3628 (2010)
- 15) Y. Huang, B. Chen, Q. Wang, K. Wei and L. Wang : Energetic efficiency and stability of dynamic bipedal walking gaits with different step lengths, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 4077/4082 (2010)
- 16) T. Wu, T. Yeh and B. Hsu : Trajectory Planning of a One-Legged Robot Performing Stable Hop, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 4922/4927 (2010)
- 17) 中村, 山根 : 拘束条件が不連続に変化するリンク系の動力学—環境と接触しながら運動するヒューマンフィギュアへの応用—, *日本ロボット学会誌*, Vol. 18, No. 3, 435/443 (2000)
- 18) M. Kouchi, M. Mochimaru, H. Iwasawa and S. Mitani : Anthropometric database for Japanese Population 1997-98, Japanese Industrial Standards Center (AIST, MITI), (2000)
- 19) Y. Fujimoto and A. Kawamura : Three Dimensional Digital Simulation and Autonomous Walking Control for Eight-Axis Biped Robot, *Proceedings of IEEE International Conference on Robotics and Automation*, 2877/2884 (1995)
- 20) J.Y.S. Luh, M.W. Walker and R.P.C. Paul : On-Line Computational Scheme for Mechanical Manipulators, *ASME Journal of Dynamics Systems, Measurement, and Control*, Vol. 102, No. 2, 69/76 (1980)
- 21) M.W. Walker and D.E. Orin : Efficient Dynamic Computer Simulation of Robotic Mechanisms, *ASME Journal of Dynamics Systems, Measurement, and Control*, Vol. 104, 205/211 (1982)
- 22) T. Mita and K. Osuka : Introduction to Robot Control, CORONA PUBLISHING CO., LTD., (1989) (in Japanese).
- 23) N. Hogan : Impedance Control; An Approach to Manipulation, Parts I-III, *ASME Journal of Dynamics Systems, Measurement, and Control* Vol. 107, No. 1, 1/24 (1985)
- 24) W. Song, M. Minami, F. Yu, Y. Zhang and A. Yanou : 3-D Hand & Eye-Vergence Approaching Visual Ser-

voing with Lyapunov-Stable Pose Tracking, *Proceedings of IEEE International Conference on Robotics and Automation*, 5210/5217 (2011)

- 25) F. Yu, W. Song and M. Minami : Visual Servoing with Quick Eye-Vergence to Enhance Trackability and Stability, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 6228/6233 (2010)
- 26) T. Yoshikawa : Dynamic Manipulability of Robot Manipulators, *Proceedings of the IEEE International Conference on Robotics and Automation*, Vol. 2, No. 1, 113/124 (1985)
- 27) 見浪, 内藤, 朝倉 : 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, 887/895 (1999)

Dual Nature of Constraint Motion and Lyapunov-stable Position/Force Control Method

*Fujia Yu, Mamoru Minami and Akira Yanou (Okayama University)

Abstract— In the model of the constrained dynamic system of a robot, constrained forces are included and expressed as an algebraic function of states (instantaneous process) and a redundancy existing in constraint dynamics (constraint redundancy). Using these results, a force and position control law is proposed by taking the advantages of the redundancy of input generalized forces to the constrained forces and instantaneous process without involving any force sensor. Then proof confirming by Lyapunov method that the exerting force equals instantaneously and constantly to desired one and that the motion of robot's hand in a movable space converges to desired hand's position. The effectiveness has been confirmed by a 2-link robot in simulation.

Key Words: Position/Force control, Lyapunov stability, Constraint condition

1 Introduction

It is well known that robots, particularly articulated types, are very dexterous and have large operable space. Hence, it will have a promising future to introduce such kinds of robots more extensively into manufacturing. For example, the tasks of the grinding or cutting of the deeply located surfaces within a cabinet might be too difficult to machine. Furthermore, for some auxiliary machining operations, it may cost too much for an expensive machining center to do. Therefore, employing robots in such areas will be a satisfactory alternative. On the other hand, comparing with a machine tool, the characteristics of robots on stiffness, damping and vibration-proofing are somewhat poor. In order to take the advantage of the dexterity of robots, much sophisticated design and control strategies have to be developed.

We think that a paper ¹⁾ had classified contact tasks of robots practically. The following classification is along with the statements in ¹⁾. Robot force control method can be largely classified into impedance control and hybrid control. In impedance control, a prescribed dynamic relation is sought to be maintained between the robot end-effector's force exerting to an object constraining the end-effector and position displacement toward the direction vertical to the object's surface ¹¹⁾. In hybrid control, the end-effector's force is explicitly controlled in selected directions and the end-effector's position is controlled in the remaining (complementary) directions ¹³⁾.

The hybrid control approaches can be further classified into three main categories ¹⁾: (A) explicit (model based) hybrid control of rigid robots in elastic contact with a compliant environment, e.g., ¹⁴⁾-¹⁵⁾, in which the end-effector force is controlled by directly commanding the joint torques of the robot based on the sensed force error; (B) implicit (position/velocity based) hybrid control of rigid robots in elastic contact with a compliant environment, e.g., ¹⁶⁾, in which the end-effector force is controlled indirectly by modifying the reference trajectory given into an inner loop joint

position/velocity controller based on the sensed force error; and (C) explicit (model based) hybrid control of rigid robots in hard contact with a rigid environment, e.g., ¹³⁾, ¹⁷⁾.

Nakamura have used the following matrix equation below in ⁶⁾, so both constraint condition and the dynamics can be represented simultaneously. This equation means that determining the constraint force F_n does not include time integration like $\ddot{\mathbf{q}}$. In this equation $\ddot{\mathbf{q}}$ express the angular acceleration, and F_n express the constraint force. Furthermore, he extended the dynamics representation method into the concept of dynamics filter ⁷⁾, ⁸⁾.

$$\begin{aligned} & \begin{bmatrix} \mathbf{M} & -\mathbf{J}_c^T \\ \frac{\partial C}{\partial \mathbf{q}^T} & 0 \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ F_n \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{\tau} - \frac{1}{2} \dot{\mathbf{M}}\dot{\mathbf{q}} - \mathbf{N}\dot{\mathbf{q}} - \mathbf{G} - \mathbf{J}_t^T F_t \\ -\dot{\mathbf{q}}^T \left[\frac{\partial C}{\partial \mathbf{q}} \left(\frac{\partial C}{\partial \mathbf{q}^T} \right) \right] \dot{\mathbf{q}} \end{bmatrix} \quad (1) \end{aligned}$$

Furthermore, many researches have discussed on the constraint-combined force/position hybrid control method. To ensure the stabilities of the constrained motion, those force and position control methods have utilized Lyapunov's stability analysis under the inverse dynamic compensation where force control strategies have been explained intelligibly in papers ¹⁾-³⁾. But these stability proofs are trying to divide the procedure into two different parts ⁹⁾, ¹⁰⁾: $\lim_{t \rightarrow \infty} \mathbf{F}_n = \mathbf{F}_{nd}$ and $\lim_{t \rightarrow \infty} \mathbf{r} = \mathbf{r}_d$, here \mathbf{F}_n and \mathbf{F}_{nd} are the actual constraint force and the desired constraint force, while \mathbf{r} and \mathbf{r}_d are the actual hand position of the manipulator and the desired one.

In ¹²⁾ it is written that "If contact is modeled by means of geometric constraints, then the contact forces cannot be expressed as algebraic functions of the state variables \mathbf{q} , $\dot{\mathbf{q}}$." The \mathbf{q} , $\dot{\mathbf{q}}$ express the angle and angular velocity of the joints. We do not think it is right, because the contact force has been calculated in (2). (2) is derived from (1), which has been pointed out by Hemami ¹⁸⁾ in the analysis of biped walking robot, denotes clearly the algebraic relation between

the input torque τ of the robot and exerting force to the working object F_n have algebraic relation, when robot's end-effector being in touch with a surface in 3-D space:

$$F_n = a(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{A}(\mathbf{q})\mathbf{J}_r^T F_t - \mathbf{A}(\mathbf{q})\tau. \quad (2)$$

Where $a(\mathbf{q}, \dot{\mathbf{q}})$ and $\mathbf{A}(\mathbf{q})$, \mathbf{J}_r are scalar function and vectors defined in following section. (2) exhibits vector τ determining F_n has a redundancy against constraint force F_n since F_n is scalar.

In this paper, the third category (C) of contacting situation that assumes rigid link manipulator and hard contacting with nonelastic environment. From (2) we can know that the force transmission process is an immediately finished process for a rigidly structured manipulator just as the acceleration being determined immediately by state variables and input generalized forces. Exploiting (2), we design a new controller whose stability is guaranteed by Lyapunov method, which exerting force $F_n(t) = F_{nd}$ and $\lim_{t \rightarrow \infty} \mathbf{r} = \mathbf{r}_d$. The effectiveness has been confirmed by a 2-link grinding robot model in simulation.

2 Analysis of Grinding Task

There are four kinds of grinding processes in common use, called respectively vertical surface grinding, horizontal surface grinding, internal grinding and cylindrical grinding. A grinding machine usually can only perform one or two of these processes because of kinematic limitation. However, all of the four kinds of tasks can be finished by a single robot manipulator for its dexterity in movement. To do so, the grinding wheel has to contact with the workpiece. A set of contacting surfaces, especially the surfaces being machined, will form constraints to the motions of the grinding wheel. As for vertical surface grinding operation shown in Fig.1(a), the grinding wheel in contact with a surface of the work-piece is not free to move through that surface, which forms a position constraint. And also, the wheel cannot freely apply arbitrary force tangent to the surface in case of no disturbing force like friction existing, which forms a set of force constraint. Situations of constraints for other kinds of grinding tasks are shown in Fig.1(b), (c) and (d).

In general, the desired grinding position trajectory is given by processing drawings for each grinding procedure, which the grinding allowance is considered. As for grinding forces F_n , F_t and F_s , the desired values should be determined carefully for different grinding conditions. Generally speaking, the grinding power is related to the metal removal rate (weight of metal being removed within unit time), which is determined by the depth of cut, the width of cut, the linear velocity of the grinding wheel, the feed rate and so on. There are many empirical formula available for the determination of grinding power, and the desired force trajectory can then be planned according to the

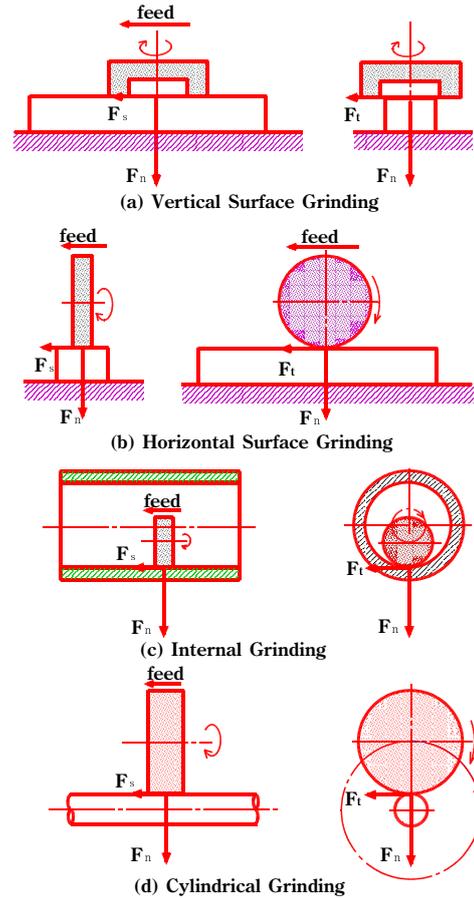


Fig. 1: Types of Grinding Tasks

power. The normal grinding force F_n is exerted in the perpendicular direction of the surface. It is a significant factor that affects ground accuracy and surface roughness of workpiece. The value of it is also related to the grinding power or directly to the tangential grinding force as

$$F_t = K_t F_n, \quad (3)$$

where, K_t is an empirical coefficient, F_t is the tangential grinding force. This relation gives us an estimated value of F_t given that K_t and F_n are known.

The axial grinding force F_s is proportional with the feed rate, and is much smaller than the former force.

Eq. (3) is based on the situation that position of the grinding cutter is controlled like currently used machining center. But when a robot is used for the grinding task, the exerting force to the object and the position of the grinding cutter should be controlled simultaneously. The F_n is generally determined by the constrained situation, and it is not suitable to apply (3) to grinding motion by the robots.

3 Modeling

Hemami and Wyman have addressed the issue of control of a moving robot according to constraint condition and examined the problem of the control of

the biped locomotion constrained in the frontal plane. Their purpose was to control the position coordinates of the biped locomotion rather than generalized forces of constrained dynamic equation involved the item of generalized forces of constraints. And the constrained force is used as a determining condition to change the dynamic model from constrained motion to free motion of the manipulators. In this paper, the grinding manipulator shown in Fig. 1, whose end-point is in contact with the constrained surface, is modeled according with Lagrangian equations of motion in term of the constraint forces, referring to what Hemami¹⁸⁾ and Arimoto⁹⁾ have done:

$$\begin{aligned} & \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\mathbf{q}}} \right) - \left(\frac{\partial L}{\partial \mathbf{q}} \right) \\ &= \boldsymbol{\tau} + \left(\frac{\partial C}{\partial \mathbf{q}^T} \right)^T / \left\| \frac{\partial C}{\partial \mathbf{q}} \right\| F_n - \left(\frac{\partial \mathbf{r}}{\partial \mathbf{q}^T} \right)^T \dot{\mathbf{r}} / \left\| \mathbf{r} \right\| F_t \\ &= \boldsymbol{\tau} + \left(\frac{\partial \mathbf{r}}{\partial \mathbf{q}^T} \right)^T \left(\frac{\partial C}{\partial \mathbf{r}} \right) / \left\| \frac{\partial C}{\partial \mathbf{q}} \right\| F_n - \frac{\dot{\mathbf{r}}}{\left\| \mathbf{r} \right\|} F_t \end{aligned} \quad (4)$$

where, \mathbf{J}_c and \mathbf{J}_r are defined as:

$$\begin{aligned} \mathbf{J}_c &= \frac{\frac{\partial C}{\partial \mathbf{q}^T}}{\left\| \frac{\partial C}{\partial \mathbf{q}^T} \right\|} = \frac{\partial C}{\partial \mathbf{r}^T} \mathbf{J} \left\| \frac{\partial C}{\partial \mathbf{r}} \right\|, \\ \mathbf{J} &= \frac{\partial \mathbf{r}}{\partial \mathbf{q}^T}, \quad \mathbf{J}_r^T = \mathbf{J}^T \frac{\dot{\mathbf{r}}}{\left\| \dot{\mathbf{r}} \right\|}, \end{aligned}$$

\mathbf{r} is the position vector of the hand and can be expressed as a kinematic equation,

$$\mathbf{r} = \mathbf{r}(\mathbf{q}). \quad (5)$$

\mathbf{q} is $n(\geq 2)$ generalized coordinates. Then this manipulator does not have kinematic redundancy. In this research we only discuss the problem under only one constraint condition, so C is a scalar function of the constraint, and is expressed as an equation of constraints,

$$C(\mathbf{r}(\mathbf{q})) = 0, \quad (6)$$

F_n is the scalar express the value of the constrained force associated with C and F_t is the scalar express the value of tangential friction force.

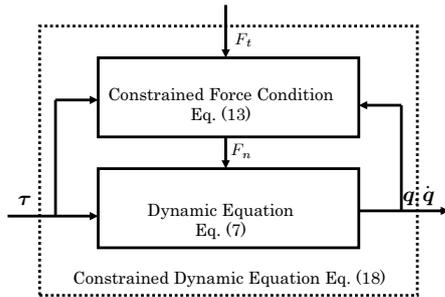


Fig. 2: Model of Constraint Dynamic System

In⁹⁾ (4) can be derived into :

$$\begin{aligned} \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \frac{1}{2}\dot{\mathbf{M}}(\mathbf{q})\dot{\mathbf{q}} + \mathbf{N}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) \\ = \boldsymbol{\tau} + \mathbf{J}_c^T(\mathbf{q})F_n - \mathbf{J}_r^T(\mathbf{q})F_t, \end{aligned} \quad (7)$$

here we express $\mathbf{M}(\mathbf{q})$ as \mathbf{M} and $\mathbf{N}(\mathbf{q}, \dot{\mathbf{q}})$ as \mathbf{N} for short. \mathbf{M} is an $n \times n$ matrix, \mathbf{N} is a $n \times n$ skew-symmetrical matrix. \mathbf{G} is a n row vectors. $\boldsymbol{\tau}$ is n inputs.

From the constraint condition (6) we can get

$$\frac{\partial C}{\partial \mathbf{q}^T} \ddot{\mathbf{q}} = -\dot{\mathbf{q}}^T \left[\frac{\partial C}{\partial \mathbf{q}} \left(\frac{\partial C}{\partial \mathbf{q}^T} \right) \right] \dot{\mathbf{q}}. \quad (8)$$

The equation (7) and (8) can be combined as follows which is the same equation of (1):

$$\begin{aligned} & \begin{bmatrix} \mathbf{M} & -\mathbf{J}_c^T \\ \frac{\partial C}{\partial \mathbf{q}^T} & 0 \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ F_n \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{\tau} - \frac{1}{2}\dot{\mathbf{M}}\dot{\mathbf{q}} - \mathbf{N}\dot{\mathbf{q}} - \mathbf{G} - \mathbf{J}_t^T F_t \\ -\dot{\mathbf{q}}^T \left[\frac{\partial C}{\partial \mathbf{q}} \left(\frac{\partial C}{\partial \mathbf{q}^T} \right) \right] \dot{\mathbf{q}} \end{bmatrix} \end{aligned} \quad (9)$$

This equation is also used by Nakamura in^{6), 7)} and⁸⁾, it is easy to see that when the matrix on the left side is invertible, there exist a $\boldsymbol{\tau}$ which determines the $\ddot{\mathbf{q}}$ and F_n satisfying (7) and (8) respectively. And as below, the inertia matrix combined with constraint condition is guaranteed to be non-singular.

$$\begin{aligned} & \det \begin{bmatrix} \mathbf{M} & -\mathbf{J}_c^T \\ \frac{\partial C}{\partial \mathbf{q}^T} & 0 \end{bmatrix} \\ &= \det \mathbf{M} \cdot \det \left(0 - \frac{\partial C}{\partial \mathbf{q}^T} \mathbf{M}^{-1} (-\mathbf{J}_c^T) \right) \\ &= \frac{1}{\left\| \frac{\partial C}{\partial \mathbf{q}^T} \right\|} \det \mathbf{M} \cdot \det \left(\frac{\partial C}{\partial \mathbf{q}^T} \mathbf{M}^{-1} \left(\frac{\partial C}{\partial \mathbf{q}^T} \right)^T \right) > 0 \end{aligned} \quad (10)$$

which means that the matrix is invertible since $\partial C(\mathbf{r}(\mathbf{q}))/\partial \mathbf{q}^T \neq \mathbf{0}$. We define:

$$m_c \triangleq \left(\frac{\partial C}{\partial \mathbf{q}^T} \right) \mathbf{M}^{-1} \left(\frac{\partial C}{\partial \mathbf{q}^T} \right)^T. \quad (11)$$

The inverse matrix can be calculated as follow:

$$\begin{aligned} & \begin{bmatrix} \mathbf{M} & -\mathbf{J}_c^T \\ \frac{\partial C}{\partial \mathbf{q}^T} & 0 \end{bmatrix}^{-1} \\ &= \begin{bmatrix} \mathbf{M}^{-1} \{ \mathbf{I} - \mathbf{J}_c^T m_c^{-1} \left\| \frac{\partial C}{\partial \mathbf{r}} \right\| \frac{\partial C}{\partial \mathbf{q}^T} \mathbf{M}^{-1} \} \\ -m_c^{-1} \left\| \frac{\partial C}{\partial \mathbf{r}} \right\| \frac{\partial C}{\partial \mathbf{q}^T} \mathbf{M}^{-1} \\ m_c^{-1} \left\| \frac{\partial C}{\partial \mathbf{r}} \right\| \mathbf{M}^{-1} \mathbf{J}_c^T \\ m_c^{-1} \left\| \frac{\partial C}{\partial \mathbf{r}} \right\| \end{bmatrix}. \end{aligned} \quad (12)$$

From (9) and (12) the constraint force F_n being identical to (2) can be solved like:

$$F_n = \left[\left(\frac{\partial C}{\partial \mathbf{q}^T} \right) \mathbf{M}^{-1} \left(\frac{\partial C}{\partial \mathbf{q}^T} \right)^T \right]^{-1} \left\| \frac{\partial C}{\partial \mathbf{r}} \right\|$$

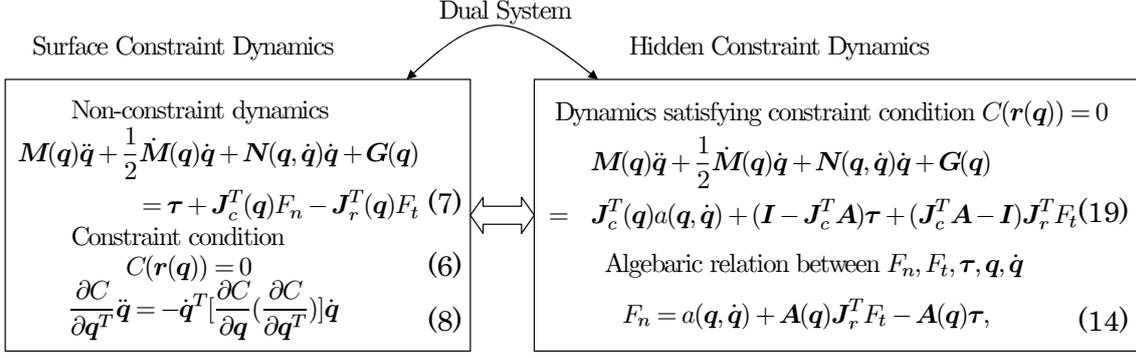


Fig. 3: Dual nature of manipulator under constraint condition

$$\left\{ -\left[\frac{\partial}{\partial \mathbf{q}} \left(\frac{\partial C}{\partial \mathbf{q}^T} \right) \dot{\mathbf{q}} + \left(\frac{\partial C}{\partial \mathbf{q}^T} \right) M^{-1} \left(\frac{1}{2} \dot{M} \dot{\mathbf{q}} + N \dot{\mathbf{q}} + G(\mathbf{q}) + \mathbf{J}_r^T F_t \right) \right] \right. \\ \left. - \left[\left(\frac{\partial C}{\partial \mathbf{q}^T} \right) M^{-1} \left(\frac{\partial C}{\partial \mathbf{q}^T} \right)^T \right]^{-1} \parallel \frac{\partial C}{\partial \mathbf{r}} \parallel \left\{ \left(\frac{\partial C}{\partial \mathbf{q}^T} \right) M^{-1} \right\} \boldsymbol{\tau} \right. \\ \triangleq a(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{A}(\mathbf{q})\mathbf{J}_r^T F_t - \mathbf{A}(\mathbf{q})\boldsymbol{\tau}, \quad (13)$$

where, $a(\mathbf{q}, \dot{\mathbf{q}})$ is a scalar representing the first term in the expression of F_n , and $\mathbf{A}(\mathbf{q})$ is an n line vector. As shown clearly in (13) that dimension of $\boldsymbol{\tau}$ is larger than the dimension of F_n , and F_n can be realized in the range space of $\mathbf{A}(\mathbf{q})$. This means $\boldsymbol{\tau}$ has a kind of redundancy against F_n . We named this redundancy appearing always in constraint dynamics of manipulator as constraint redundancy. $a(\mathbf{q}, \dot{\mathbf{q}})$ and $\mathbf{A}(\mathbf{q})$ are defined concretely as follow:

$$a(\mathbf{q}, \dot{\mathbf{q}}) \triangleq m_c^{-1} \parallel \frac{\partial C}{\partial \mathbf{r}^T} \parallel \left\{ -\left[\frac{\partial}{\partial \mathbf{q}^T} \left(\frac{\partial C}{\partial \mathbf{q}} \right) \dot{\mathbf{q}} \right] \right. \\ \left. + \left(\frac{\partial C}{\partial \mathbf{q}^T} \right) M^{-1} \left(\frac{1}{2} \dot{M} \dot{\mathbf{q}} + N \dot{\mathbf{q}} + G \right) \right\}, \quad (14)$$

$$\mathbf{A}(\mathbf{q}) \triangleq m_c^{-1} \parallel \frac{\partial C}{\partial \mathbf{r}^T} \parallel \left\{ \left(\frac{\partial C}{\partial \mathbf{q}^T} \right) M^{-1} \right\}, \quad (15)$$

(13) is written as follow for short:

$$F_n = F_n(\mathbf{q}, \dot{\mathbf{q}}, \boldsymbol{\tau}, F_t). \quad (16)$$

From (9) and (12), we can get that:

$$\ddot{\mathbf{q}} = M^{-1} \left(\boldsymbol{\tau} - \frac{1}{2} \dot{M} \dot{\mathbf{q}} - N \dot{\mathbf{q}} - G - \mathbf{J}_t^T F_t + \mathbf{J}_c^T F_n \right). \quad (17)$$

Inserting F_n , (9) into (17), the state equation of the system excluding the constrained force (as $F_n > 0$) can be rewritten as

$$M(\mathbf{q})\ddot{\mathbf{q}} + \frac{1}{2}\dot{M}(\mathbf{q})\dot{\mathbf{q}} + N(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + G(\mathbf{q}) \\ = \mathbf{J}_c^T(\mathbf{q})a(\mathbf{q}, \dot{\mathbf{q}}) + (\mathbf{I} - \mathbf{J}_c^T \mathbf{A})\boldsymbol{\tau} + (\mathbf{J}_c^T \mathbf{A} - \mathbf{I})\mathbf{J}_r^T F_t \quad (18)$$

which is denoted as a model of the constraint dynamic system in Fig. 2.

Solutions of these dynamic equation always satisfy the constrained condition (6). The forward description of contacting dynamics has been represented by (7) and (6). The fact that the solution \mathbf{q} of (7) have to satisfy (6) make us anticipate that F_d should be satisfied simultaneously and instantly regardless of the motion \mathbf{q} , $\dot{\mathbf{q}}$, and any $\boldsymbol{\tau}$. The algebraic solution has been derived as (13). Then the dynamics of the manipulator whose solution \mathbf{q} , $\dot{\mathbf{q}}$ always satisfy the constraint condition (8) derived from (6) has been translated into (18). In the Fig. 3, the backward relation of (13) and (18) are described in the right hand half. (13) exhibits clearly the comment ‘‘If contact is modeled by means of geometric constraints, then the contact forces cannot be expressed as algebraic functions of the state variables \mathbf{q} , $\dot{\mathbf{q}}$,’’ in pp.55¹²⁾ is not correct and contradicting to. The backward description of constraint dynamics has been long ignored by robotic researchers, but we had proposed the force sensorless position/force control based on using this backward description directly⁴⁾. Here two descriptions on left and right side are equivalent, then it can be called a ‘‘dual system.’’ In this paper, we propose a new controller with Lyapunov -stability over non-constraint motion and with instantaneous achievement of desired contacting force.

4 Force and position controller

In this paper, we propose a controller whose convergence is guaranteed by Lyapunov method and introduce the calculate method of it.

4.1 Controller

Let \mathbf{S} be a column full rank matrix spanning the null space of $\partial C / \partial \mathbf{q}$, we can get $\mathbf{S}^T \left(\partial C / \partial \mathbf{q}^T \right)^T = 0$, i.e.

$$\mathbf{S}^T \mathbf{J}_c^T = \mathbf{0}. \quad (19)$$

It is possible to find an auxiliary vector \mathbf{p} satisfies

$$\dot{\mathbf{q}} = \mathbf{S}\dot{\mathbf{p}}, \quad (20)$$

in the manipulator situation, \mathbf{p} is the end-effector position except the constraint direction, and

$$\ddot{\mathbf{q}} = \dot{\mathbf{S}}\dot{\mathbf{p}} + \mathbf{S}\ddot{\mathbf{p}}. \quad (21)$$

From the definition of \mathbf{A} in (15) and \mathbf{S} in (19) we know that $[\mathbf{A}^T, \mathbf{S}^T]^T$ is reversible, i.e. there certainly exist \mathbf{B} ($n \times 1$ vector) and \mathbf{D} ($n \times (n-1)$ matrix) satisfies

$$\begin{bmatrix} \mathbf{A} \\ \mathbf{S}^T \end{bmatrix} \mathbf{B} = \begin{bmatrix} 1 \\ \mathbf{0} \end{bmatrix} \quad (22)$$

$$\begin{bmatrix} \mathbf{A} \\ \mathbf{S}^T \end{bmatrix} \mathbf{D} = \begin{bmatrix} \mathbf{0}^T \\ \mathbf{I}_{n-1} \end{bmatrix} \quad (23)$$

respectively. Here \mathbf{I}_{n-1} is a $(n-1) \times (n-1)$ identify matrix. \mathbf{B} means selection matrix of range space of $\partial C/\partial \mathbf{q}$, which corresponds directly to the range space of $\mathbf{A}(\mathbf{q})$ as shown in (15) and null space of $\partial C/\partial \mathbf{q}^T$. And \mathbf{D} is vice versa.

Before propose the controller we will put forward three assumptions:

(a) The constraint condition is known and expressed by $C(\mathbf{r}(\mathbf{q})) = 0$.

(b) The tangential grinding force can be calculated by (3).

(c) The dynamic parameters of the system are known.

The following is a controller guaranteeing that the closed loop satisfies the exerted constrained force F_n be identical to the desired force F_{nd} regardless of time and the robot's motion along with the free motion directions.

$$\boldsymbol{\tau} = \mathbf{B}(F_{nd} - a) + \mathbf{D}[k_p(\mathbf{p}_d - \mathbf{p}) + k_d(\dot{\mathbf{p}}_d - \dot{\mathbf{p}})] + \mathbf{J}_r^T F_t \quad (24)$$

Here on the right side, the first term is to realize the desired constrained force, the second term is to control the pose of the manipulator, while the third item is to compensate the friction, with an assumption of F_t being able to be gotten correctly. This assumption can be materialized by using F_n and (3). The block diagram of the system is given in Fig. 4.

Because (2) is a algebraic function of the input torque, when we substitute (24) into (13), we can get

$$\begin{aligned} F_n &= a(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{A}\mathbf{B}(F_{nd} - a) \\ &+ \mathbf{A}\mathbf{D}[k_p(\mathbf{p}_d - \mathbf{p}) + k_d(\dot{\mathbf{p}}_d - \dot{\mathbf{p}})] \end{aligned} \quad (25)$$

from the definition we know that $\mathbf{A}\mathbf{B} = 1$ and $\mathbf{A}\mathbf{D} = \mathbf{0}$ so

$$F_n = F_{nd}, \quad (26)$$

here (26) does not include the variable of time t as a time differential manner, meaning the output force always equals the desired one.

4.2 Calculation of the variables

From definition of Jacobian matrix \mathbf{J} we can get

$$\dot{\mathbf{r}} = \mathbf{J}\dot{\mathbf{q}} \quad (27)$$

there exists \mathbf{P} which satisfies:

$$\dot{\mathbf{p}} = \mathbf{P}\mathbf{J}\dot{\mathbf{q}} = \tilde{\mathbf{J}}\dot{\mathbf{q}} \quad (28)$$

Define $\mathbf{P}\mathbf{J}$ as $\tilde{\mathbf{J}}$, taking (28) into (20) we can get

$$\tilde{\mathbf{J}}\mathbf{S}\dot{\mathbf{p}} = \dot{\mathbf{p}}. \quad (29)$$

$\tilde{\mathbf{J}}$ is a row full rank matrix of $(n-1) \times n$, so it is possible to find a \mathbf{S} which satisfying $\tilde{\mathbf{J}}\mathbf{S} = \mathbf{I}_{n-1}$ is a solution of (29) i.e. $\tilde{\mathbf{J}}\mathbf{S} = \mathbf{I}$ satisfies (20), and $\tilde{\mathbf{J}}\mathbf{S} = \mathbf{I}_{n-1}$ and (19) can be combined into one matrix equation:

$$\begin{bmatrix} \tilde{\mathbf{J}} \\ \mathbf{J}_c \end{bmatrix} \mathbf{S} = \begin{bmatrix} \mathbf{I}_{n-1} \\ \mathbf{0} \end{bmatrix} \quad (30)$$

Here we define $\hat{\mathbf{J}} = [\tilde{\mathbf{J}}^T, \mathbf{J}_c^T]^T$. $\tilde{\mathbf{J}}$ expresses range space of robot's hand motion not being constrained by object, and \mathbf{J}_c represents constraint direction, which is orthogonal to row vectors of $\tilde{\mathbf{J}}$. Then $\hat{\mathbf{J}}$ is reversible, so \mathbf{S} can be calculated as follow:

$$\mathbf{S} = \hat{\mathbf{J}}^{-1} \begin{bmatrix} \mathbf{I}_{n-1} \\ \mathbf{0} \end{bmatrix}. \quad (31)$$

By using \mathbf{S} , we can calculate \mathbf{B} and \mathbf{D} in (22) and (23), and calculate the input $\boldsymbol{\tau}$ in (24).

5 Stability Analysis

Premultiply \mathbf{S}^T we can get

$$\begin{aligned} \mathbf{S}^T \mathbf{M} \dot{\mathbf{S}}\dot{\mathbf{p}} + \mathbf{S}^T \mathbf{M} \mathbf{S}\ddot{\mathbf{p}} + \mathbf{S}^T \left(\frac{1}{2} \dot{\mathbf{M}} + \mathbf{N} \right) \mathbf{S}\dot{\mathbf{p}} \\ = \mathbf{S}^T \boldsymbol{\tau} + \mathbf{S}^T \mathbf{J}_c^T F_n - \mathbf{S}^T \mathbf{J}_r^T F_t \end{aligned} \quad (32)$$

Substituting (24) into (32) we can get

$$\begin{aligned} \mathbf{S}^T \mathbf{M} \dot{\mathbf{S}}\dot{\mathbf{p}} + \mathbf{S}^T \mathbf{M} \mathbf{S}\ddot{\mathbf{p}} + \mathbf{S}^T \left(\frac{1}{2} \dot{\mathbf{M}} + \mathbf{N} \right) \mathbf{S}\dot{\mathbf{p}} \\ = k_p(\mathbf{p}_d - \mathbf{p}) + k_d(\dot{\mathbf{p}}_d - \dot{\mathbf{p}}) \end{aligned} \quad (33)$$

here we set the desired end-effector is static which means $\dot{\mathbf{p}}_d = 0$, so closed loop dynamics is

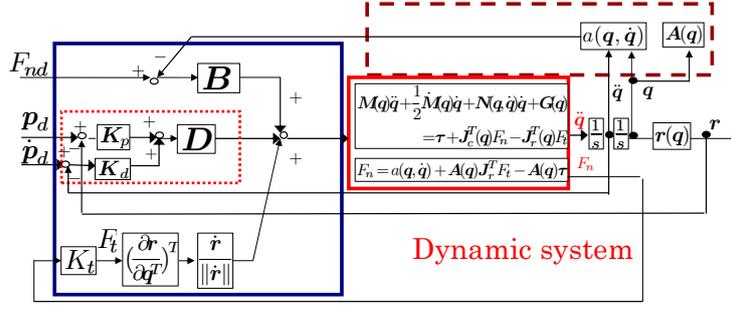
$$\begin{aligned} \mathbf{S}^T \mathbf{M} \dot{\mathbf{S}}\dot{\mathbf{p}} + \mathbf{S}^T \mathbf{M} \mathbf{S}\ddot{\mathbf{p}} + \mathbf{S}^T \frac{1}{2} \dot{\mathbf{M}} \mathbf{S}\dot{\mathbf{p}} - k_p(\mathbf{p}_d - \mathbf{p}) \\ = -\mathbf{S}^T \mathbf{N} \mathbf{S}\dot{\mathbf{p}} - k_d \dot{\mathbf{p}} \end{aligned} \quad (34)$$

set Lyapunov argument as:

$$V = \frac{1}{2} \dot{\mathbf{p}}^T \mathbf{S}^T \mathbf{M} \mathbf{S}\dot{\mathbf{p}} + \frac{1}{2} (\mathbf{p}_d - \mathbf{p})^T k_p (\mathbf{p}_d - \mathbf{p}) \quad (35)$$

so

$$\begin{aligned} \dot{V} &= \dot{\mathbf{p}}^T \mathbf{S}^T \mathbf{M} \dot{\mathbf{S}}\dot{\mathbf{p}} + \dot{\mathbf{p}}^T \mathbf{S}^T \mathbf{M} \mathbf{S}\ddot{\mathbf{p}} \\ &+ \dot{\mathbf{p}}^T \mathbf{S}^T \frac{1}{2} \dot{\mathbf{M}} \mathbf{S}\dot{\mathbf{p}} - k_p \dot{\mathbf{p}}^T (\mathbf{p}_d - \mathbf{p}) \end{aligned} \quad (36)$$



Position and Force Controller

Fig. 4: Grinding position / force control system

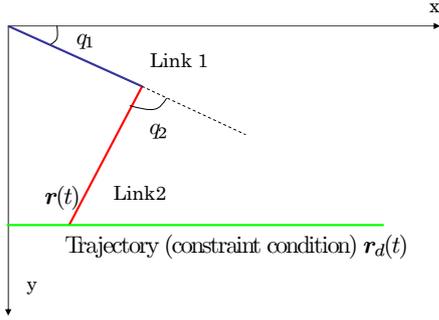


Fig. 5: 2-link grinding robot

from (34), (36) can be transformed to

$$\dot{V} = -\dot{\mathbf{p}}^T \mathbf{S}^T \mathbf{N} \mathbf{S} \dot{\mathbf{p}} - k_d \dot{\mathbf{p}}^T \dot{\mathbf{p}} \quad (37)$$

because \mathbf{N} is a skew symmetrical matrix, $\dot{\mathbf{p}}^T \mathbf{S}^T \mathbf{N} \mathbf{S} \dot{\mathbf{p}} = 0$, so

$$\dot{V} = -k_d \dot{\mathbf{p}}^T \dot{\mathbf{p}} \quad (38)$$

because $\dot{V} \leq 0$ and $V \geq 0$, from (38) we can see if and only if $\dot{\mathbf{p}} = \mathbf{0}$, $\dot{V} = 0$, which means that submitting $\dot{\mathbf{p}} = \mathbf{0}$ into (35) we can get $\mathbf{p}_d - \mathbf{p} = \mathbf{0}$ from Lasalle theorem we know that

$$\lim_{t \rightarrow \infty} \mathbf{p} = \mathbf{p}_d, \lim_{t \rightarrow \infty} \dot{\mathbf{p}} = \mathbf{0} \quad (39)$$

Because the constraint system satisfies that $C(\mathbf{r}(\mathbf{q})) = 0$,

$$\lim_{t \rightarrow \infty} \mathbf{r} = \mathbf{r}_d. \quad (40)$$

So the system will converge to the desired pose at last.

6 simulation

In this section I will introduce some simulations have been done to check the controller in 2-link condition as Fig. 5. To a 2-link manipulator the variables in (18) can be calculated as follow:

$$\mathbf{M} = \begin{bmatrix} J_1 + J_2 + 2\beta \cos q_2 & J_2 + 2\beta \cos q_2 \\ J_2 + 2\beta \cos q_2 & J_2 \end{bmatrix} \quad (41)$$

$$\frac{1}{2} \dot{\mathbf{M}} \dot{\mathbf{q}} + \mathbf{N} \dot{\mathbf{q}} = \begin{bmatrix} -(2\dot{q}_1 \dot{q}_2 + \dot{q}_1) \beta \sin q_2 \\ \dot{q}_1^2 \beta \sin q_2 \end{bmatrix} \quad (42)$$

here $J_1 = I_1 + (m_1 + 4m_2)l_1^2$, $J_2 = I_2 + m_2 l_2^2$ and $\beta = 2m_2 l_1 l_2$ and I, m, l are the initial moment, mass and length of the links. Jacobian matrix \mathbf{J} is.

$$\mathbf{J} = \begin{bmatrix} -\sin q_1 - \sin(q_1 + q_2) & -\sin(q_1 + q_2) \\ \cos q_1 + \cos(q_1 + q_2) & \cos(q_1 + q_2) \end{bmatrix} \quad (43)$$

In the simulation we set the constraint condition as:

$$C(\mathbf{r}(\mathbf{q})) = y - 0.6 = 0 \quad (44)$$

so

$$\frac{\partial C}{\partial \mathbf{r}} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad \dot{\mathbf{r}} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (45)$$

so

$$\mathbf{J}_c = \begin{bmatrix} \cos(q_1 + q_2) \\ \cos q_1 + \cos(q_1 + q_2) \end{bmatrix} \quad (46)$$

$$\mathbf{J}_r = \begin{bmatrix} -\sin(q_1 + q_2) \\ -\sin q_1 - \sin(q_1 + q_2) \end{bmatrix} \quad (47)$$

from the variables above we can calculate a, \mathbf{A} and m_c defined in (14) (15) and (11) and calculate F_n by (18).

For 2-link manipulator, \mathbf{S} and $\partial C / \partial \mathbf{q}$ in (19) are both 2×1 vectors, we can get $\mathbf{S}^T (\partial C / \partial \mathbf{q}^T) = 0$, i.e.

$$\mathbf{S}^T \mathbf{J}_c^T = \mathbf{0} \quad (48)$$

This \mathbf{S} also satisfies the follow equation,

$$\dot{\mathbf{q}} = \mathbf{S} \dot{p}_x \quad (49)$$

where \dot{p}_x is the end-effector position on x-axis, here we define the two elements of \mathbf{S} as $[S_1, S_2]$ and $\mathbf{J}_c = [J_{c1}, J_{c2}]$, also the two angles of the joints are define as q_1 and q_2 respectively, so (49) can be written as

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = \begin{bmatrix} S_1 \\ S_2 \end{bmatrix} \dot{p}_x \quad (50)$$

from the definition of Jacobian matrix we can know

$$[J_{11}, J_{12}] \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = \dot{p}_x \quad (51)$$

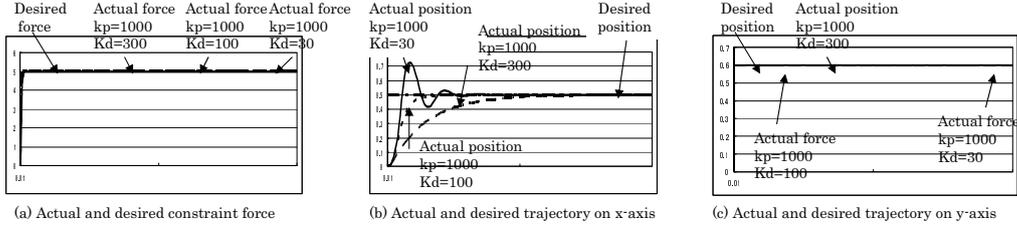


Fig. 6: Simulation result when $k_p = 1000$, $x_d = 0.5[m]$ and $F_{nd} = 6 + \text{sint}[N]$

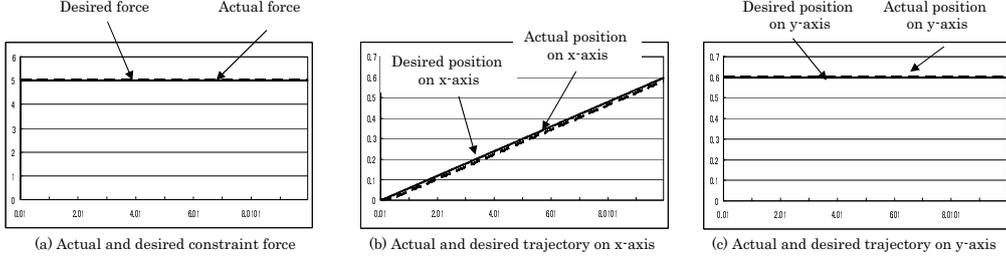


Fig. 7: Simulation result when $k_p = 1000$, $k_d = 300$, $x_d = 0.06t[m]$ and $F_{nd} = 5[N]$

here $[J_{11}, J_{12}]$ is the first line of Jacobian matrix and also the \hat{J} in (28). To get S satisfies $\hat{J}S = I$, take (50) into (51), we can get

$$[J_{11}, J_{12}] \begin{bmatrix} S_1 \\ S_2 \end{bmatrix} = 1 \quad (52)$$

so \hat{J} in (31) is

$$\hat{J} = \begin{bmatrix} J_{11} & J_{12} \\ J_{c1} & J_{c2} \end{bmatrix} \quad (53)$$

and

$$S = \hat{J}^{-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{J_{c2}J_{11} - J_{c1}J_{12}}{J_{c2}J_{11} - J_{c1}J_{12}} \\ -\frac{J_{c1}}{J_{c2}J_{11} - J_{c1}J_{12}} \end{bmatrix}. \quad (54)$$

Because there are only 2 links, B and D in (24) can be simplified as follow

$$\begin{bmatrix} A \\ S^T \end{bmatrix} B = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (55)$$

$$\begin{bmatrix} A \\ S^T \end{bmatrix} D = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad (56)$$

Then we did some simulations to check the controller, first we set the desired as a step input, in this simulation $x_d = 0.5[m]$, $F_{nd} = 5[N]$, $k_p = 1000$ and $k_d = 300, 100, 30$ respectively, the result is shown in Fig. 6, the constraint force and y -position are coincide and all x -position can converge to the desired position.

In the second simulation I will show the data of the simulation when $k_p = 1000$, $k_d = 300$, $x_d = 0.06t[m]$ and $F_{nd} = 5[N]$. From Fig. 7 (a) we can see that the system will output the desired force. And from (b)

and (c) we can see that the controller can control the end-effector move along the desired trajectory.

In the third simulation we make the desired force as a function of time, in this simulation $k_p = 1000$, $k_d = 300$, $x_d = 0.06t[m]$ and $F_{nd} = 6 + \text{sint}[N]$ the result is shown in Fig. 8, from the simulation result we can see that however the force changes, the end-effector tracks the desired position on x -axis while moves along the constraint line on y -axis. Just as it is explained, the surface constraint dynamics express the system by the equation of dynamics and constraint condition, but this is only a condition but not a limitation. In the hidden constraint dynamics, the constraint condition is combined into the dynamics equation, so no matter how much the input τ is, it just effect the constraint force, and the end-effector motion on the tangent direction, but the end-effector always moves along the constraint line of plane. This is the reason why we call the two dynamics dual system.

7 Conclusions

In this paper we designed a constraint-combined force/position controller for the continuous shape-grinding system, and prove the convergence of the controller in a new way by Lyapunov method, the output force of the system can always equal to the desired one. At last we did some simulations to confirm the controller. In the future we will apply it into experiment.

- 1) Jaydeep. Roy and Louis L. Whitcomb: Adaptive Force Control of Position/Velocity Controlled Robots: Theory and Experiment, IEEE Transactions on Robotics and Automation, Vol.18, No.2, April, (2002).

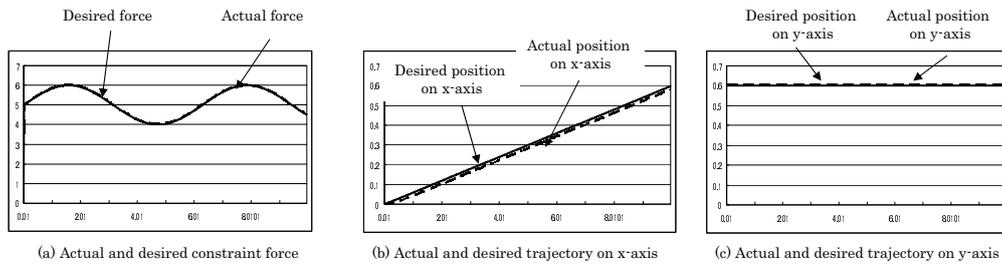


Fig. 8: Simulation result when $k_p = 1000$, $k_d = 300$, $x_d = 0.06t[m]$ and $F_{nd} = 6 + \sin t[N]$

- 2) Z. X. Peng and N. Adachi: Position and Force Control of Manipulators without Using Force Sensors(in Japanese), Trans. of JSME(C), Vol.57, No.537, 1625/1630 (1991).
- 3) D.Wang and N.H. McClamroch: Position and Force Control for Constrained Manipulator Motion: Lyapunov's Direct Approach, IEEE Trans. Robotics and Automation, Vol.9, 308/313(1993).
- 4) M.Minami, T.Asakura, L.X.Dong, Y.M.Huang. "Position Control and Explicit Force Control of Constrained Manipulators for Accurate Grinding Tasks" Advanced Robotics, Int. J. of RSJ, Vol.11, No.3, 285/230(1997).
- 5) M. Yamaguchi, "Feed Rate Control of Deburring Robot System(in Japanese)", JRSJ, Vol.9, No.3, 89/93(1991).
- 6) Katsu Yamane, Yoshihiko Nakamura. "O(N) Forward Dynamics Computation of Open Kinematic Chains Based on the Principle of Virtual Work", 2824/2831.
- 7) K.Yamane and Y.Nakamura. "Dynamics Filter - Concept and Implementation of On-Line Motion Generator for Human Figures" IEEE Transactions on Robotics and Automation, vol. 19, no. 3, 421/432(2003).
- 8) Yamane,K.,Nakamura,Y. "Dynamics Filter - Concept and Implementation of On-Line Motion Generator for Human Figures",ICRA2000 688/695 (2000).
- 9) S. Arimoto, Mechanics and Control of Robot(in Japanese), Asakura Publishing Co., Ltd., Tokyo, Japan, (1990).
- 10) McClamroch, N.H., and Wang, Danwei, Feedback stabilization and tracking of constrained robots, IEEE Transaction on Automatic Control, Vol.33, No.5, May 1988, 419/426 (1988).
- 11) N. Hogan: Impedance control: An approach to manipulation: Part 1-Theory, J.Dynamic Syst., Measurement Contr., Vol.107, 1/7(1985).
- 12) Miomir Vukobratovic, Veljko Potkonjak, Vladimir Matijevic "Dynamics of robots with contact tasks", 55(2003).
- 13) M.H.Raibert and J.J.Craig: Hybrid Position/Force Control of Manipulators, Trans. of the ASME, J. of Dynamic Systems, Measurement and Control, Vol.102, 126/133(1981).
- 14) B. Siciliano and L. Villani: A passivity-based approach to force regulation and motion control of robot manipulators, Automatica, Vol.32, No.3, 443/447(1996).
- 15) L. Villani, C.C. de Wit and B. Brogliato: An exponentially stable adaptive control for force and position tracking of robot manipulators, IEEE Trans. Automat. Contr., Vol44, 778/802(1999).
- 16) J. De Schutter and H.Van Brussel: Compliant robot motion 2. A control approach based on external control loops, Int. J. Robot. Res., Vol.7, No.4, 18/33(1988).
- 17) T. Yoshikawa: Dynamic Hybrid Position/Force control of Robot Manipulators — Description of Hand Constraints and Calculation of Joint Driving Force, IEEE J. on Robotics and Automation, Vol.RA-3, No.5, 386/392(1987).
- 18) Hooshang Hemami and Bostwick F. Wyman: Modeling and Control of Constrained Dynamic Systems with Application to Biped Locomotion in the Frontal Plane, IEEE Trans. on Automatic Control, Vol.AC-24, No.4, 526/535(1979).
- 19) Takeshi Ikeda and Mamoru Minami: Research of Grinding Robot without Force Sensor by Using Algebraic Equation(in Japanese), Transactions of the Japan Society of Mechanical Engineers(C), Vol.71, No.702, 270/277(2005).

肘付拘束冗長性を利用したハンド軌道追従と拘束運動の 非干渉化制御

—消費エネルギーを基準にした肘付き位置と抗力の最適化—

○糸島道之 尾崎壮紘 前場友秀 見浪護 矢納陽 (岡山大学)

Decoupling Control of Hand Trajectory Tracking and Constraint Motion using Redundancy with Bracing Elbow

-Optimization of reaction force and position of bracing elbow based on energy consumption-

*M. Itoshima, T. Ozaki, T. Maeba, M. Minami and A. Yanou
(Okayama University)

Abstract— Considering that human can do accurate task with small powers by contacting elbow or wrist on a table, we thought that manipulators can save energy and do a task more precisely like human by bracing itself. Bracing redundancy exists in the motion of the manipulator with contacting of its elbows. Although dimensions of motion are decreased by constraint, these dimensions can change to the dimensions that control force. Based on this fact, we propose a method to control position/force of the manipulator by using the bracing redundancy. That is, this paper shows that the dimension of position and force can be used effectively, simultaneously can do hand's trajectory tracking and control reaction force of the elbow by simulations.

Key Words: Constraint motion, Redundancy with bracing elbow, Position/Force control method

1 はじめに

人間は字を書く動作において手首や肘を机につくことで、より少ない力でペンを持ち、より正確に字を書けることを経験的に知っている。これは人間の習熟した行動の一例である。ロボットがこのような環境との拘束接地による反力を利用して重力の影響を抑え、より少ない消費エネルギーで精度の高い作業を実行できるのではないかと考え、マニピュレータの肘付き動作を提案する。

本論文で複数のリンクがテーブルや床などの環境に接触するマニピュレータについて考える。前述したように、このような状況は Fig.1 で示されるように字を書くなどの人間の一般的な動作で見られる。冗長マニピュレータは冗長自由度を使うことで複雑な作業を行うことができる。しかし冗長自由度が増加するにつれてマニピュレータのリンク数や重量が増加し、その結果マニピュレータを制御することが難しくなる。この問題を解決するためには字を書いたり、いすに座るなどの人間の行動を模倣した制御方法が有効であると考えられる。

運動学に基づいた冗長マニピュレータは幅広く研究され、その効果は数十年前に Chirikjian と Burdick¹⁾ により紹介されている。しかし、高い冗長性に反比例してマニピュレータは大きな重量を持つ。そのためマニピュレータのハンドは三次元空間で動くことが出来る一方で、ハンドへの負荷は微小なものに制限されていた。多くの研究では障害物回避^{2),5)} や形状の最適化^{6),7)} などの冗長性を利用する方法⁸⁾⁻¹¹⁾ が議論されているが、現段階では重量の問題が工場での使用に役立つレベルには達していない。

これまでも環境に接触している冗長マニピュレータの有効性と精度を議論した研究例は存在する。West と Asada¹²⁾ は拘束されたマニピュレータの位置/力同



Fig. 1: Human's writing motion

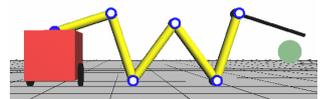


Fig. 2: Simulation screen

時制御コントローラを設計するための一般的な運動学的接触モデルを提案した。そしてマニピュレータの接触動作を制御するために複数点接触の運動学モデルが Oussama Khatib^{13),14)} (接触面がばねモデルとして仮定されている) によって提案された。しかし接触面は硬いものとして考えることが妥当である。一例として、鉄のばね定数は約 10^6 [N/m] であり環境を表すばねモデルは自然でないと考えられる。つまりこの接触モデルは現実的ではない。そのため、本論文で私たちは環境の接触による変形がない硬い接触モデルを扱う。そして、接触力は明確な代数方程式として拘束式と運動方程式から得られる。ロボティクスの分野では Peng¹⁵⁾ により最初に適用された式 (1) はロボットの手先が拘束されているときの入力トルクと抗力の代数関係を表している。

$$A\mathbf{f}_n = \mathbf{a} - B\boldsymbol{\tau} \quad (1)$$

\mathbf{f}_n は抗力、 A と \mathbf{a} 、 B は次章で定義されるベクトルと行列、 $\boldsymbol{\tau}$ は入力ベクトルである。

本論文で提案する位置と力を同時に制御するコントローラは式 (1) に基づいている。Peng の方法では力センサーとして式 (1) を用いているが、本論文では目標抗力 \mathbf{f}_{nd} を達成するための入力トルク $\boldsymbol{\tau}$ を計算するために用いた。式 (1) の存在は 2 つの重要な事実を表す。最

初の事実は代数方程式である式 (1) により入力トルクが決定された瞬間に抗力が計算できる。これに反して、角速度及び角度は積分によって計算されるため、入力トルクの決定から積分時間遅れた後に決まる。この代数方程式を用いることにより時間遅れなく目標の抗力に制御することが可能である。もう一つの重要な事実は入力トルク τ は拘束運動において抗力 f_n に対して冗長性を持つことである。式 (1) に関して $n \times 1$ のベクトル τ に対して $p (p < n)$ 点拘束時の p 個の抗力について考えると f_{nd} を達成するための τ には冗長性がある。すなわち、拘束を行うことで運動に関する次元は減少するが、この減少した次元は力を制御するための次元に変化する。本論文ではこれを”Bracing Redundancy”と名付ける。この冗長性と B の零空間を利用して、拘束している関節の抗力と位置、手先の軌道追従を同時かつ非干渉に制御できる新しいコントローラを提案する。

本報告では提案したコントローラの有効性をシミュレーションにより確認する。また、マニピュレータの肘付き位置と抗力を最適化することにより更なる消費エネルギーの低減につながると考えている。そのためシミュレーション結果から消費エネルギーに基づく肘付き位置と抗力の最適化について考察する。

2 拘束運動のモデリング

2.1 ハンド拘束運動

ここでは拘束面に接触しつつリンク先端のハンドで接触作業を行うロボットについて考える。 $\mathbf{q} \in R^n$ を一般化座標、 $\boldsymbol{\tau} \in R^n$ を一般化入力、 f_n を抗力、 f_t を摩擦力とすると、ハンド拘束状態の運動方程式は、

$$\begin{aligned} M(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) + D\dot{\mathbf{q}} \\ = \boldsymbol{\tau} + \left\{ \left(\frac{\partial C}{\partial \mathbf{q}^T} \right)^T / \left\| \frac{\partial C}{\partial \mathbf{r}^T} \right\| \right\} f_n - \left(\frac{\partial \mathbf{r}}{\partial \mathbf{q}^T} \right)^T \frac{\dot{\mathbf{r}}}{\|\dot{\mathbf{r}}\|} f_t \end{aligned} \quad (2)$$

と表される。ただし、 M は $n \times n$ の慣性行列、 \mathbf{h} と \mathbf{g} はコリオリ力・遠心力と重力の影響を表す $n \times 1$ のベクトル、 D は関節の粘性摩擦係数を表す $n \times n$ の対角行列 $D = \text{diag}[D_1, D_2, \dots, D_n]$ である。ここで、手先の位置・姿勢ベクトル $\mathbf{r} \in R^n$ と超曲面を表すスカラー関数の拘束条件 C は、順運動学関係式より、

$$\mathbf{r} = \mathbf{r}(\mathbf{q}) \quad (3)$$

$$C(\mathbf{r}(\mathbf{q})) = 0 \quad (4)$$

と表される。ここで拘束されるハンドは作業座標系の中で非拘束方向に運動できる自由度が残されていることから $n > 1$ である。また、本論文では一般化座標 \mathbf{q} は関節角度、一般化入力 $\boldsymbol{\tau}$ は入力トルクを表す。

2.2 肘つき拘束運動

n リンクマニピュレータの中間リンクが p 個の拘束

$$C_i(\mathbf{r}_i(\mathbf{q})) = 0 \quad (i = 1, 2, \dots, p) \quad (5)$$

を受けているときの運動方程式について考える。ここで \mathbf{r}_i は拘束を受けている第 i リンクの位置/姿勢を表す運動学方程式であり、式 (3) と同様、

$$\mathbf{r}_i = \mathbf{r}_i(\mathbf{q}) \quad (6)$$

$$\dot{\mathbf{r}}_i = \mathbf{J}_i(\mathbf{q})\dot{\mathbf{q}} \quad (7)$$

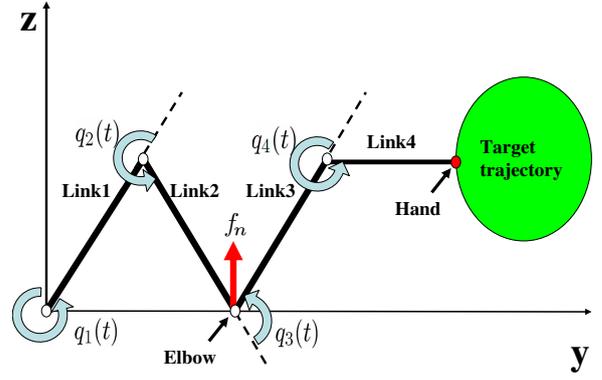


Fig. 3: Model

である。

式 (2) は、ハンド拘束運動を表している。拘束条件が複数存在する場合には、式 (2) の右辺第 2 項の抗力の係数ベクトルと右辺第 3 項の摩擦力の係数ベクトルはそれぞれ、

$$\left(\frac{\partial C_i}{\partial \mathbf{q}^T} \right)^T / \left\| \frac{\partial C_i}{\partial \mathbf{r}^T} \right\| = \mathbf{j}_{ci}^T \quad (8)$$

$$\left(\frac{\partial \mathbf{r}_i}{\partial \mathbf{q}^T} \right)^T \frac{\dot{\mathbf{r}}_i}{\|\dot{\mathbf{r}}_i\|} = \mathbf{j}_{ti}^T \quad (9)$$

の $n \times 1$ のベクトルとなる。さらに、

$$\mathbf{J}_c^T = [\mathbf{j}_{c1}^T, \mathbf{j}_{c2}^T, \dots, \mathbf{j}_{cp}^T] \quad (10)$$

$$\mathbf{J}_t^T = [\mathbf{j}_{t1}^T, \mathbf{j}_{t2}^T, \dots, \mathbf{j}_{tp}^T] \quad (11)$$

$$\mathbf{f}_n = [f_{n1}, f_{n2}, \dots, f_{np}]^T \quad (12)$$

$$\mathbf{f}_t = [f_{t1}, f_{t2}, \dots, f_{tp}]^T \quad (13)$$

と定義する。 \mathbf{J}_c^T 、 \mathbf{J}_t^T は $n \times p$ 行列であり、 \mathbf{f}_n 、 \mathbf{f}_t は $p \times 1$ のベクトルである。これらを考慮すると、 p 個の点で中間リンクが拘束されているマニピュレータの運動方程式は、

$$\begin{aligned} M(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) + D\dot{\mathbf{q}} \\ = \boldsymbol{\tau} + \sum_{i=1}^p (\mathbf{j}_{ci}^T f_{ni}) - \sum_{i=1}^p (\mathbf{j}_{ti}^T f_{ti}) \\ = \boldsymbol{\tau} + \mathbf{J}_c^T \mathbf{f}_n - \mathbf{J}_t^T \mathbf{f}_t \end{aligned} \quad (14)$$

となる。また、式 (5) を時間 t で 2 回微分し、 $\ddot{\mathbf{q}}$ の拘束条件を求めると、

$$\dot{\mathbf{q}}^T \left[\frac{\partial}{\partial \mathbf{q}} \left(\frac{\partial C_i}{\partial \mathbf{q}^T} \right) \right] \dot{\mathbf{q}} + \left(\frac{\partial C_i}{\partial \mathbf{q}^T} \right) \ddot{\mathbf{q}} = 0 \quad (15)$$

が得られる。マニピュレータが常に拘束面に拘束されるためには、式 (14) の解 $\mathbf{q}(t)$ が時間 t に無関係に式 (5) を満たさなければならない。式 (5) の時間微分によって得られた式 (15) を満たす $\ddot{\mathbf{q}}$ と式 (14) の $\ddot{\mathbf{q}}$ が同じ値をとるとき、式 (14) の $\mathbf{q}(t)$ は式 (5) を満たすことになる。ここで抗力 \mathbf{f}_n と摩擦力 \mathbf{f}_t について

$$\mathbf{f}_t = \mathbf{K} \mathbf{f}_n, \quad \mathbf{K} = \text{diag}[K_1, K_2, \dots, K_p] \quad (16)$$

とおくと、式 (14) は以下のようになる。

$$\begin{aligned} M(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) + D\dot{\mathbf{q}} \\ = \boldsymbol{\tau} + (\mathbf{J}_c^T - \mathbf{J}_t^T \mathbf{K}) \mathbf{f}_n \end{aligned} \quad (17)$$

式 (15), (17) を行列形式で表すと以下のようになる。

$$\begin{bmatrix} M(\mathbf{q}) & -(\mathbf{J}_c^T - \mathbf{J}_t^T \mathbf{K}) \\ \frac{\partial C_t}{\partial \mathbf{q}^T} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \mathbf{f}_n \end{bmatrix} \quad (18)$$

$$= \begin{bmatrix} \boldsymbol{\tau} - \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{g}(\mathbf{q}) - D\dot{\mathbf{q}} \\ -\dot{\mathbf{q}}^T \left[\frac{\partial}{\partial \mathbf{q}} \left(\frac{\partial C_t}{\partial \mathbf{q}^T} \right) \right] \dot{\mathbf{q}} \end{bmatrix} \quad (19)$$

2.3 肘付き時の抗力の導出

本節では \mathbf{f}_n の導出方法について述べる。まず、式 (14), (15) から $\ddot{\mathbf{q}}$ を消去すると

$$\begin{aligned} \left(\frac{\partial \mathbf{C}}{\partial \mathbf{q}^T} \right) M^{-1} \left(\frac{\partial \mathbf{C}}{\partial \mathbf{q}^T} \right)^T \frac{\mathbf{f}_n}{\left\| \frac{\partial \mathbf{C}}{\partial \mathbf{r}^T} \right\|} \\ = \left(\frac{\partial \mathbf{C}}{\partial \mathbf{q}^T} \right) M^{-1} \left(\mathbf{J}_t^T \mathbf{K} \mathbf{f}_n + D\dot{\mathbf{q}} + \mathbf{h} + \mathbf{g} - \boldsymbol{\tau} \right) \\ - \dot{\mathbf{q}}^T \left[\frac{\partial}{\partial \mathbf{q}} \left(\frac{\partial \mathbf{C}}{\partial \mathbf{q}^T} \right) \right] \dot{\mathbf{q}} \end{aligned} \quad (20)$$

が得られる。

ここで、 $(\partial \mathbf{C} / \partial \mathbf{q}^T) M^{-1} (\partial \mathbf{C} / \partial \mathbf{q}^T)^T = M_c$ と置くことにより、

$$\begin{aligned} M_c \mathbf{f}_n = \left\| \frac{\partial \mathbf{C}}{\partial \mathbf{r}^T} \right\| \left(\frac{\partial \mathbf{C}}{\partial \mathbf{q}^T} \right) M^{-1} (\mathbf{J}_t^T \mathbf{K} \mathbf{f}_n + D\dot{\mathbf{q}} \\ + \mathbf{h} + \mathbf{g} - \boldsymbol{\tau}) - \dot{\mathbf{q}}^T \left[\frac{\partial}{\partial \mathbf{q}} \left(\frac{\partial \mathbf{C}}{\partial \mathbf{q}^T} \right) \right] \dot{\mathbf{q}} \end{aligned} \quad (21)$$

となり、また

$$\mathbf{B} = \left\| \frac{\partial \mathbf{C}}{\partial \mathbf{r}^T} \right\| \left(\frac{\partial \mathbf{C}}{\partial \mathbf{q}^T} \right) M^{-1} \quad (22)$$

と置くと式 (21) は以下のようになる。ただし、 $p \times n$ 行列の \mathbf{B} は $\text{rank}(\mathbf{B}) = p$ で行フルランクの行列である。

$$\begin{aligned} M_c \mathbf{f}_n = \mathbf{B} \mathbf{J}_t^T \mathbf{K} \mathbf{f}_n - \mathbf{B} \boldsymbol{\tau} + \mathbf{B} \{ D\dot{\mathbf{q}} + \mathbf{h} + \mathbf{g} \} \\ - \dot{\mathbf{q}}^T \left[\frac{\partial}{\partial \mathbf{q}} \left(\frac{\partial \mathbf{C}}{\partial \mathbf{q}^T} \right) \right] \dot{\mathbf{q}} \end{aligned} \quad (23)$$

となる。ここで

$$\mathbf{a} = \mathbf{B} \{ D\dot{\mathbf{q}} + \mathbf{h} + \mathbf{g} \} - \dot{\mathbf{q}}^T \left[\frac{\partial}{\partial \mathbf{q}} \left(\frac{\partial \mathbf{C}}{\partial \mathbf{q}^T} \right) \right] \dot{\mathbf{q}} \quad (24)$$

とすると

$$M_c \mathbf{f}_n = \mathbf{B} \mathbf{J}_t^T \mathbf{K} \mathbf{f}_n - \mathbf{B} \boldsymbol{\tau} + \mathbf{a} \quad (25)$$

となる。さらに

$$\mathbf{A} = M_c - \mathbf{B} \mathbf{J}_t^T \mathbf{K} \quad (26)$$

と置くことにより、

$$\mathbf{A} \mathbf{f}_n = \mathbf{a} - \mathbf{B} \boldsymbol{\tau} \quad (27)$$

となり、拘束点の抗力 \mathbf{f}_n は入力トルク $\boldsymbol{\tau}$ との代数方程式から求めることができる。

ここで、式 (17), (27) から \mathbf{f}_n を消去し、 $|\mathbf{A}| \neq 0$ と仮定し、 $\mathbf{C} = \mathbf{J}_c^T - \mathbf{J}_t^T \mathbf{K}$ と置いて整理すると

$$\begin{aligned} M\ddot{\mathbf{q}} + \mathbf{h} + \mathbf{g} + D\dot{\mathbf{q}} = \boldsymbol{\tau} + \mathbf{C} \mathbf{A}^{-1} (\mathbf{a} - \mathbf{B} \boldsymbol{\tau}) \\ = (\mathbf{I} - \mathbf{C} \mathbf{A}^{-1} \mathbf{B}) \boldsymbol{\tau} + \mathbf{C} \mathbf{A}^{-1} \mathbf{a} \end{aligned} \quad (28)$$

となる。式 (28) は任意のトルク $\boldsymbol{\tau}$ を与えたときに常に拘束条件式 (4) を満たしながら運動することを示している。

2.4 モータを含めたロボットダイナミクス

本研究では、マニピュレータの関節を動かすのに使用するモータを直流サーボモータの等価な電子回路とする。ここで、モータの端子電圧 v_i , 抵抗 R_i , インダクタンス L_i , 回路を流れる電流 i_i , モータの角変位 θ_i , 発生トルク τ_{gi} , 負荷トルク τ_{Li} , 逆起電力 v_{gi} , モータの慣性モーメント I_{mi} , 逆起電力定数 K_{Ei} , トルク定数 K_{Ti} , 減速機の粘性摩擦係数 d_{mi} とする。このときモータについて電圧方程式, 逆起電力, 運動方程式, 発生トルクの関係式は以下のように表現できる。

$$v_i(t) = L_i \dot{i}_i + R_i i_i(t) + v_{gi}(t) \quad (29)$$

$$v_{gi}(t) = K_{Ei} \dot{\theta}_i(t) \quad (30)$$

$$I_{mi} \ddot{\theta}_i = \tau_{gi}(t) - \tau_{Li}(t) - d_{mi} \dot{\theta}_i \quad (31)$$

$$\tau_{gi}(t) = K_{Ti} i_i(t) \quad (32)$$

ただし、磁界と各定数との関係から直流モータでは、 $K_{Ti} = K_{Ei} (= K_i)$ であることが知られている。

ここで、式 (30) を式 (29) に、式 (32) を式 (31) にそれぞれ代入すると、

$$v_i = L_i \dot{i}_i + R_i i_i + K_i \dot{\theta}_i \quad (33)$$

$$I_{mi} \ddot{\theta}_i = K_i i_i - \tau_{Li} - d_{mi} \dot{\theta}_i \quad (34)$$

となる。また、モータを減速比が k_i の歯車列を介してマニピュレータに取り付けた場合、

$$\theta_i = k_i q_i \quad (35)$$

$$\tau_{Li} = \frac{\tau_i}{k_i} \quad (36)$$

となり、式 (33), 式 (34) を \dot{i}_i と τ_i の式にすると、

$$L_i \dot{i}_i = v_i - R_i i_i - K_i k_i \dot{q}_i \quad (37)$$

$$\tau_i = -I_{mi} k_i^2 \ddot{q}_i + K_i k_i i_i - d_{mi} k_i^2 \dot{q}_i \quad (38)$$

となる。式 (37), 式 (38) をベクトル, 行列で表記する場合、

$$L \dot{\mathbf{i}} = \mathbf{v} - \mathbf{R} \mathbf{i} - \mathbf{K}_m \dot{\mathbf{q}} \quad (39)$$

$$\boldsymbol{\tau} = -\mathbf{J}_m \ddot{\mathbf{q}} + \mathbf{K}_m \mathbf{i} - \mathbf{D}_m \dot{\mathbf{q}} \quad (40)$$

と置く。ここに、

$$\begin{aligned} \mathbf{v} &= [v_1, v_2, \dots, v_s]^T \\ \mathbf{i} &= [i_1, i_2, \dots, i_s]^T \end{aligned}$$

であり、また、

$$\begin{aligned} \mathbf{L} &= \text{diag}[L_1, L_2, \dots, L_s] \\ \mathbf{R} &= \text{diag}[R_1, R_2, \dots, R_s] \\ \mathbf{K}_m &= \text{diag}[K_{m1}, K_{m2}, \dots, K_{ms}] \\ \mathbf{J}_m &= \text{diag}[J_{m1}, J_{m2}, \dots, J_{ms}] \\ \mathbf{D}_m &= \text{diag}[D_{m1}, D_{m2}, \dots, D_{ms}] \\ K_{mi} &= K_i k_i, J_{mi} = I_{mi} k_i^2, D_{mi} = d_{mi} k_i^2 \end{aligned}$$

と定義し、すべて正の要素を持つ。

式(40)を式(17)に代入して整理すると、

$$\begin{aligned} (\mathbf{M}(\mathbf{q}) + \mathbf{J}_m)\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) + (\mathbf{D} + \mathbf{D}_m)\dot{\mathbf{q}} \\ = \mathbf{K}_m \mathbf{i} + (\mathbf{J}_c^T - \mathbf{J}_t^T \mathbf{K}) \mathbf{f}_n \end{aligned} \quad (41)$$

となる。式(14)と式(15)の関係と同様に、式(15)を満たす $\ddot{\mathbf{q}}$ と式(41)の $\ddot{\mathbf{q}}$ が同じ値をとるとき、式(41)の $\mathbf{q}(t)$ は式(5)を満たすことになる。

2.5 複数拘束条件付ロボット/モータの連立方程式

式(41)と式(17)の $\ddot{\mathbf{q}}$ が同じ値をとるため、抗力 \mathbf{f}_n は従属的に決定される。この関係を表すため、式(41)、(17)を以下のように変形する。

$$\begin{aligned} (\mathbf{M}(\mathbf{q}) + \mathbf{J}_m)\ddot{\mathbf{q}} - (\mathbf{J}_c^T - \mathbf{J}_t^T \mathbf{K}) \mathbf{f}_n \\ = \mathbf{K}_m \mathbf{i} - \mathbf{h} - \mathbf{g} - (\mathbf{D} + \mathbf{D}_m)\dot{\mathbf{q}} \end{aligned} \quad (42)$$

$$\left(\frac{\partial C_i}{\partial \mathbf{q}^T}\right) \ddot{\mathbf{q}} = -\dot{\mathbf{q}}^T \left[\frac{\partial}{\partial \mathbf{q}} \left(\frac{\partial C_i}{\partial \mathbf{q}^T}\right)\right] \dot{\mathbf{q}} \quad (43)$$

式(43)の左辺の $\partial C_i / \partial \mathbf{q}^T$ は $1 \times s$ の横ベクトルであり、右辺の $\dot{\mathbf{q}}^T \left[\frac{\partial}{\partial \mathbf{q}} \left(\frac{\partial C_i}{\partial \mathbf{q}^T}\right)\right] \dot{\mathbf{q}}$ はスカラーである。式(42)、式(43)、式(39)を合わせて以下の様な式に表す。

$$\begin{aligned} \begin{bmatrix} \mathbf{M} + \mathbf{J}_m & -(\mathbf{J}_c^T - \mathbf{J}_t^T \mathbf{K}) & \mathbf{0} \\ \frac{\partial C}{\partial \mathbf{q}^T} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{L} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \mathbf{f}_n \\ \mathbf{i} \end{bmatrix} \\ = \begin{bmatrix} \mathbf{K}_m \mathbf{i} - \mathbf{h} - \mathbf{g} - (\mathbf{D} + \mathbf{D}_m)\dot{\mathbf{q}} \\ -\dot{\mathbf{q}}^T \left[\frac{\partial}{\partial \mathbf{q}} \left(\frac{\partial C}{\partial \mathbf{q}^T}\right)\right] \dot{\mathbf{q}} \\ \mathbf{v} - \mathbf{R}\mathbf{i} - \mathbf{K}_m \dot{\mathbf{q}} \end{bmatrix} \end{aligned} \quad (44)$$

ただし、

$$\mathbf{C} = [C_1, C_2, \dots, C_p]^T \quad (45)$$

は $p \times n$ 行列である。ここで、慣性項 $(\mathbf{M} + \mathbf{J}_m)$ は $n \times n$ 行列、抗力の係数ベクトル $(\mathbf{j}_c^T - \mathbf{j}_t^T \mathbf{K})$ は $n \times p$ の縦ベクトル、 $\partial C / \partial \mathbf{q}^T$ は $p \times n$ の横ベクトル、インダクタンス \mathbf{L} は $n \times n$ の対角行列である。さらに、

$$\mathbf{M}^* = \begin{bmatrix} \mathbf{M} + \mathbf{J}_m & -(\mathbf{J}_c^T - \mathbf{J}_t^T \mathbf{K}) & \mathbf{0} \\ \frac{\partial C}{\partial \mathbf{q}^T} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{L} \end{bmatrix} \quad (46)$$

$$\mathbf{c} = \begin{bmatrix} \mathbf{K}_m \mathbf{i} - \mathbf{h} - \mathbf{g} - (\mathbf{D} + \mathbf{D}_m)\dot{\mathbf{q}} \\ -\dot{\mathbf{q}}^T \left[\frac{\partial}{\partial \mathbf{q}} \left(\frac{\partial C}{\partial \mathbf{q}^T}\right)\right] \dot{\mathbf{q}} \\ \mathbf{v} - \mathbf{R}\mathbf{i} - \mathbf{K}_m \dot{\mathbf{q}} \end{bmatrix} \quad (47)$$

とおくと、式(44)は、

$$\mathbf{M}^* \begin{bmatrix} \ddot{\mathbf{q}} \\ \mathbf{f}_n \\ \mathbf{i} \end{bmatrix} = \mathbf{c} \quad (48)$$

と表される。式(48)を未知である $\ddot{\mathbf{q}}$ 、 \mathbf{f}_n 、 \mathbf{i} の式にすると、

$$\begin{bmatrix} \ddot{\mathbf{q}} \\ \mathbf{f}_n \\ \mathbf{i} \end{bmatrix} = \mathbf{M}^{*-1} \mathbf{c} \quad (49)$$

となり、未知である $\ddot{\mathbf{q}}$ 、 \mathbf{f}_n 、 \mathbf{i} を同時に求めることができる。

3 コントローラ

拘束条件 $\mathbf{C}(\mathbf{r}(\mathbf{q}))$ と等しい次元にあり、環境と拘束された複数のリンクの間に働く目標抗力 \mathbf{f}_{nd} を得られる抗力と位置を分離制御する制御則は式(27)から直接得ることができる。

$$\boldsymbol{\tau} = \mathbf{B}^+(\mathbf{a} - \mathbf{A}\mathbf{f}_{nd}) + (\mathbf{I} - \mathbf{B}^+\mathbf{B})\mathbf{l} \quad (50)$$

$\partial \mathbf{C} / \partial \mathbf{q}$ は行フルランクの行列、 \mathbf{M} は常に正則であるので $p \times n$ 行列の \mathbf{B} は $\text{rank}(\mathbf{B}) = p$ で行フルランクの行列である。そこで、 $n \times 1$ のベクトル $\boldsymbol{\tau}$ は抗力を制御したあとに $n - p$ の冗長性があり、 $\text{rank}(\mathbf{I} - \mathbf{B}^+\mathbf{B}) = n - p$ となる。したがって、 $\boldsymbol{\tau}$ の残っている制御入力の手先目標軌道 \mathbf{r}_d を追従するためや他の目的のために任意のベクトル \mathbf{l} として使うことができる。

式(27)は拘束条件の数 p の仮定から得られる。 \mathbf{l} は拘束位置 p で位置制御に用いることができ、下式で与える。

$$\begin{aligned} \mathbf{l} &= \sum_{i=1}^p \mathbf{J}_i^T [K_{pi}(\mathbf{r}_{di} - \mathbf{r}_i) + K_{di}(\dot{\mathbf{r}}_{di} - \dot{\mathbf{r}}_i)] \\ &+ \mathbf{J}_n^T [K_{pn}(\mathbf{r}_{dn} - \mathbf{r}_n) + K_{dn}(\dot{\mathbf{r}}_{dn} - \dot{\mathbf{r}}_n)] \end{aligned} \quad (51)$$

右辺第1項は拘束点の位置制御に、右辺第2項は手先の位置制御にそれぞれ用いる。 $\text{rank}(\mathbf{I} - \mathbf{B}^+\mathbf{B}) = n - p$ の \mathbf{B} のゼロ空間に注目すると式(51)に含まれる仕事の次元を式(51)の仕事と関係しない他の仕事に使うことは出来ない。

$p \times n$ の行列 $\mathbf{B}(\mathbf{q})$ が行フルランクと仮定し、式(27)への入力である式(50)の $\boldsymbol{\tau}$ が閉ループであることを考えれば下式を得られる。

$$\begin{aligned} \mathbf{A}\mathbf{f}_n &= \mathbf{a} - \mathbf{B}\{\mathbf{B}^+(\mathbf{a} - \mathbf{A}\mathbf{f}_{nd}) + (\mathbf{I} - \mathbf{B}^+\mathbf{B})\mathbf{l}\} \\ &= \mathbf{A}\mathbf{f}_{nd} \end{aligned} \quad (52)$$

$p \times p$ の行列 \mathbf{A} が逆行列を持つとき、 $\mathbf{f}_n = \mathbf{f}_{nd}$ となる。

式(50)は直接トルクを発生させることができる直接駆動モータのロボットでは使うことができるが、普通のモータの入力は電圧である。この場合、式(50)のコントローラは下式に書き換えることができる。

$$\mathbf{v} = \mathbf{K}_v [\mathbf{B}^+(\mathbf{a} - \mathbf{A}\mathbf{f}_{nd}) + (\mathbf{I} - \mathbf{B}^+\mathbf{B})\mathbf{l}] \quad (53)$$

ここで \mathbf{v} はモータの入力電圧、 \mathbf{K}_v は正定値の係数行列である。また、Fig. 4は式(27)が表す入力トルク $\boldsymbol{\tau}$ と抗力 \mathbf{f}_n の代数方程式と抗力と位置を分離制御する方法を表す。

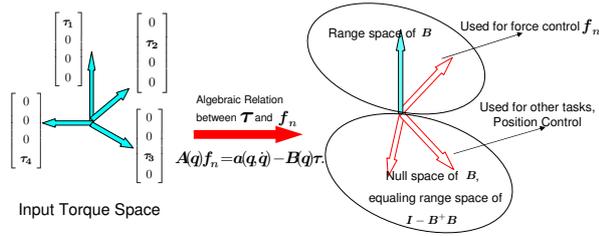


Fig. 4: Algebraic relation

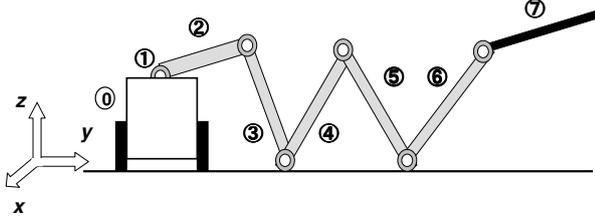


Fig. 5: Simulation model

4 シミュレーション

4.1 コントローラの有効性の検証

この節では、提案するコントローラの式 (53) が位置と力を同時かつ非干渉に制御できることをシミュレーションにより示す。Fig.5のような7リンクマニピュレータでシミュレーションを行い、数字はリンクの数を表す。台車をリンク0とし、台車とリンク2の結合点にあるz軸まわりの長さのない関節をリンク1、他のx軸まわりの関節の先にあるリンクを手前からリンク2, 3, ..., 7とする。ここで、物理パラメータを次のように設定した。それぞれのリンクの質量を $m_i = 5.0[\text{kg}]$ 、長さを $l_i = 0.6[\text{m}]$ 、関節の粘性摩擦係数を $D_i = 2.0[\text{N} \cdot \text{m} \cdot \text{s}/\text{rad}]$ 、トルク定数を $K_i = 0.203[\text{N} \cdot \text{m}/\text{A}]$ 、抵抗を $R_i = 1.1[\Omega]$ 、インダクタンスを $L_i = 0.0017[\text{H}]$ 、モータの慣性モーメントを $I_{mi} = 1.64 \times 10^{-4}[\text{kg} \cdot \text{m}^2]$ 、減速比を $k_i = 3.0$ 、減速器の粘性摩擦係数を $d_{mi} = 0.01[\text{N} \cdot \text{m} \cdot \text{s}/\text{rad}]$ とし、パラメータは実際のモータの仕様書から得ている。第4リンクの関節と第6リンクの関節を拘束し。以下では肘1、肘2と呼ぶ。肘1と肘2の目標抗力を $f_{4d} = f_{6d} = 125[\text{N}]$ 、手先の目標軌道 $(y_d, z_d)[\text{m}]$ と肘1の目標軌道 $y_{4d}[\text{m}]$ 、肘2の目標軌道 $y_{6d}[\text{m}]$ を以下のように与える。

$$y_d = 0.1 \cos \frac{2\pi}{T} t + 2.0 \quad (54)$$

$$z_d = 0.1 \sin \frac{2\pi}{T} t + 0.3 \quad (55)$$

$$y_{4d} = 0.1 \sin \frac{2\pi}{T} t + 0.55 \quad (56)$$

$$y_{6d} = 0.1 \cos \frac{2\pi}{T} t + 1.2 \quad (57)$$

ただし、周期を $T = 5[\text{sec}]$ 、シミュレーション時間を $10[\text{sec}]$ 、サンプリングタイムを $5[\text{msec}]$ とし、リンクの初期関節角度をそれぞれ $(q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7) = (0.0, 0.0, -0.4\pi, -0.5\pi, 0.7\pi, -0.6\pi, 0.7\pi, -0.5\pi)[\text{rad}]$ とする。このときのトルク入力の手先の抗力を Fig.6, 肘

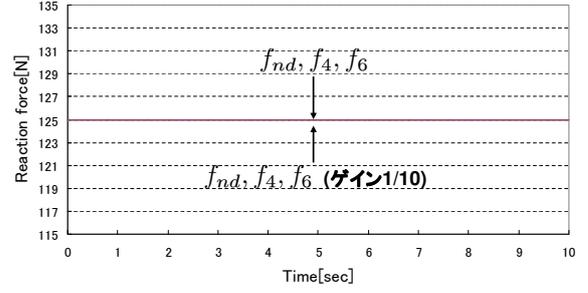


Fig. 6: Reaction force of the elbow(torque input)

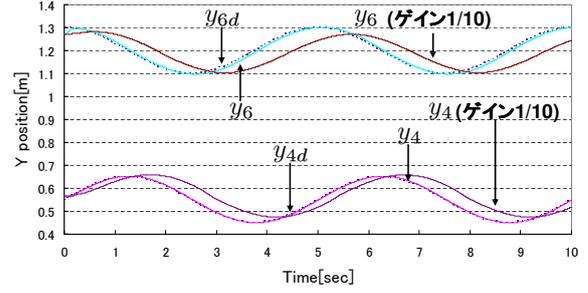


Fig. 7: Position of the elbow(torque input)

の位置を Fig.7, 手先の位置を Fig.8 に示し、電圧入力の手先の抗力を Fig.9, 肘の位置を Fig.10, 手先の位置を Fig.11 に示す。Figs.6-11 より適切なゲイン (手先までのゲイン $K_p = 8000[\text{N}], K_d = 300[\text{N} \cdot \text{s}]$ 、肘1までのゲイン $K_p = 9000[\text{N}], K_d = 500[\text{N} \cdot \text{s}]$ 、肘2までのゲイン $K_p = 5000[\text{N}], K_d = 300[\text{N} \cdot \text{s}]$ を与えたときに位置、抗力ともに目標値に追従している。つまり、位置と抗力は同時に制御できることが分かる。また、適切なゲインの1/10倍のゲインを与えると位置の追従精度は悪くなるが抗力は目標値に追従している。つまり、位置と抗力は互いに干渉していないことが分かる。

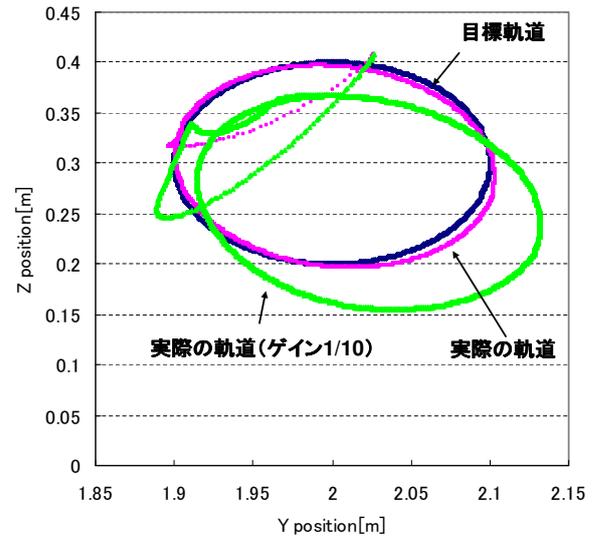


Fig. 8: Position of the hand(torque input)

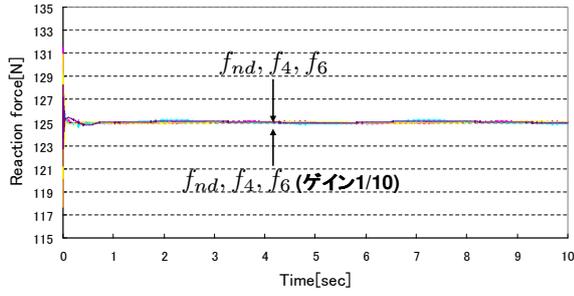


Fig. 9: Reaction force of the elbow(voltage input)

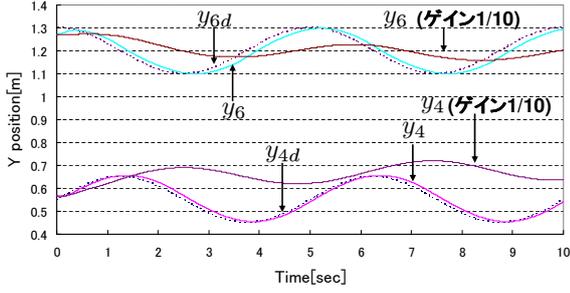


Fig. 10: Position of the elbow(voltage input)

4.2 肘付き抗力の最適化

この節では抗力の最適化について述べる. 式 (53) の右辺第 1 項が 0 となるような抗力を設定することにより抗力によるエネルギーの消費量を抑えることができると考えられる. そこで目標抗力 f_{nw} は以下の式で得られる.

$$f_{nw} = \frac{a}{A} \quad (58)$$

$$a = B \left\{ (D + D_m) \dot{q} + h + g - \dot{q}^T \left[\frac{\partial}{\partial q} \left(\frac{\partial C}{\partial \dot{q}^T} \right) \right] \dot{q} \right\} \quad (59)$$

$$B = \left\| \frac{\partial C}{\partial r^T} \right\| \left\{ \left(\frac{\partial C}{\partial \dot{q}^T} \right) (M + J_m)^{-1} \right\} \quad (60)$$

$$A = M_c - B J_t^T K \quad (61)$$

ただし, 最適化の評価関数として消費エネルギーを用いているためモータのダイナミクスを含む. ここで消費エネルギーの導出方法を以下に示す. マニピュレータの i リンクでの消費エネルギーは

$$E_i(t) = \int_0^t v_i(t) i_i(t) dt \quad (62)$$

となり, 全リンクでの消費エネルギーは以下のように求めることができる.

$$E_{sum}(t) = \sum_{i=1}^4 E_i(t) \quad (63)$$

シミュレーションは Fig.3 で表す 4 リンクマニピュレータで行う. 物理パラメータを次のように設定し

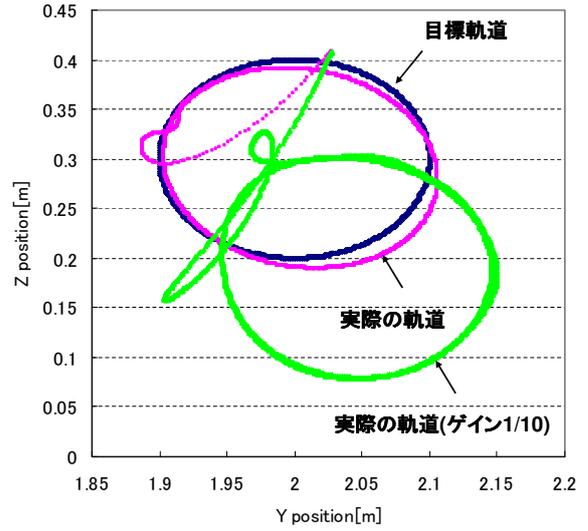


Fig. 11: Position of the hand(voltage input)

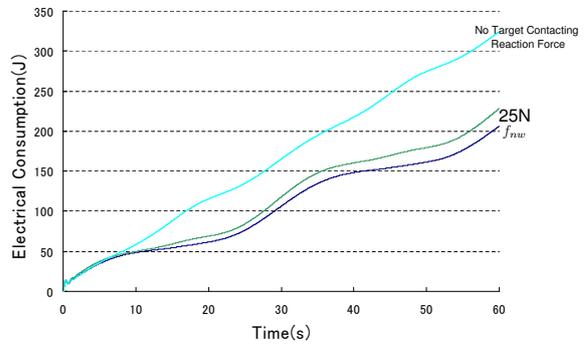


Fig. 12: Electrical consumption comparison of simulation of following trajectory

た. それぞれのリンクの質量 $m_{1,2,3} = 1.0[kg]$, $m_4 = 0.5[kg]$, 長さ $l_i = 0.5[m]$, 関節の粘性摩擦係数 $D_i = 2.9[N \cdot m \cdot s/rad]$, 手先の比例ゲイン $k_{pi} = 600[N]$, 手先の微分ゲイン $k_{di} = 100[N \cdot s]$, 肘位置の比例ゲイン $k_{pei} = 600[N]$, 肘位置の微分ゲイン $k_{dei} = 100[N \cdot s]$, トルク定数 $K_i = 0.2[N \cdot m/A]$, 抵抗 $R_i = 0.6[\Omega]$, インダクタンス $L_i = 0.17[H]$, モータの慣性モーメント $Im_i = 1.64 \times 10^{-4}[kg \cdot m^2]$, 減速比 $k_i = 3.0$, 減速機の粘性摩擦係数 $dm_i = 0.1[N \cdot m \cdot s/rad]$. また, 手先の目標軌道 $(y_d, z_d)[m]$ と肘の目標位置 $y_{2d}[m]$ を下式で与える.

$$y_{2d} = 0.6 \quad (64)$$

$$y = 0.15 \cos\left(\frac{\pi}{15}t\right) + 1.1 \quad (65)$$

$$z = 0.15 \sin\left(\frac{\pi}{15}t\right) + 0.4 \quad (66)$$

Fig.12 は定常値で最も消費エネルギーが少なかった目標抗力 25[N]¹⁶⁾, 時変の目標抗力 f_{nw} , 目標抗力を与えない場合の消費エネルギーを示している. Fig.12 より, 時変の目標抗力 f_{nw} を与えたときに消費エネルギーが最も少なくなっていることから提案した手法の有効性が確認できた.

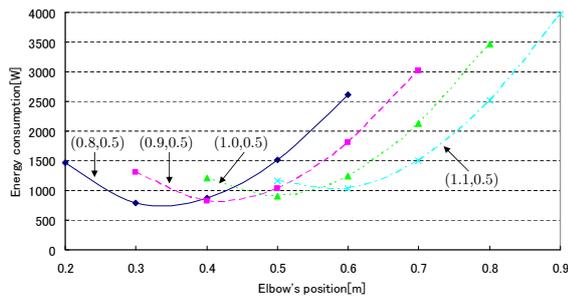


Fig. 13: Evaluation of energy consumption

4.3 肘付き位置の最適化

この節では肘付き位置の最適化について述べる。手先と肘の目標値を以下のように与える。ただし、各リンクの初期関節角度は肘と手先の初期位置から逆運動学を解いて与えている。

$$y_{2d} = \text{const.} \quad (67)$$

$$y_d = 0.2 \cos\left(\frac{2\pi}{10}t\right) + y_c \quad (68)$$

$$z_d = 0.2 \sin\left(\frac{2\pi}{10}t\right) + 0.5 \quad (69)$$

手先の比例ゲイン $k_{pi} = 300[\text{N}]$ 、手先の微分ゲイン $k_{di} = 50[\text{N}\cdot\text{s}]$ 、肘位置の比例ゲイン $k_{pei} = 200[\text{N}]$ 、肘位置の微分ゲイン $k_{dei} = 40[\text{N}\cdot\text{s}]$ とし、他のパラメータは前節のシミュレーションと同じとする。Fig.13は手先目標軌道の中心座標を $(y_c, z_c)[\text{m}] = (0.8, 0.5)$, $(0.9, 0.5)$, $(1.0, 0.5)$, $(1.1, 0.5)$ と変化させたときに $0.1[\text{m}]$ 刻みで与えた複数の肘付き位置で50秒間の消費エネルギーを示している。Fig.13より、手先目標軌道の中心座標からY軸負の方向に $0.5[\text{m}]$ 離れている肘付き位置が消費エネルギーが最小になることが分かった。

5 おわりに

本論文では肘付拘束冗長性を利用してハンド軌道追従と拘束運動を同時かつ非干渉に制御できるコントローラの有効性をシミュレーションにより確認した。また、マニピュレータの肘付き位置と抗力を最適化することにより更なる消費エネルギーの低減を目指し、シミュレーション結果から消費エネルギーに基づく肘付き位置と抗力の最適化について考察した。今後の方針としては肘付き抗力と位置の最適化を組み合わせて更なるエネルギーの最適化を図ることが挙げられる。

参考文献

- 1) G.S. Chirikjian, J.W. Burdick: "A Hyper-Redundant Manipulator", *IEEE Robotics & Automation Magazine*, December 1994, 22/29(1994).
- 2) K. Glass, R. Colbaugh, D. Lim, H. Seraji: "Real-time collision avoidance for redundant manipulators", *IEEE Transactions on Robotics and Automation*, Vol.11, 448/457(1995).
- 3) H. Seraji, B. Bon: "Real-Time Collision Avoidance for Position-Controlled Manipulators", *IEEE Transactions on Robotics and Automation*, Vol.15, No.4, 670/677(1999).
- 4) L. Zlajpah, B. Nemeč: "Kinematic Control Algorithms for On-line Obstacle Avoidance for Redundant Manipulator", *International Conference on Intelligent Robots and Systems*, Lausanne, Vol.2, 1898/1903(2002).

- 5) K.K. Lee, M. Buss: "Obstacle Avoidance for Redundant Robots Using Jacobian Transpose Method", *International Conference on Intelligent Robots and Systems*, San Diego, 3509/3514(2007).
- 6) R. Kurazume, T. Hasegawa: "Impedance Matching for Free Flying Robots", (in Japanese) *In The 20th Annual Conference of the Robotics Society of Japan*, 3J16(2002).
- 7) R. Kurazume, T. Hasegawa: "A New Index of Serial-Link Manipulator Performance Combining Dynamic Manipulability and Manipulating Force Ellipsoid", *IEEE Transactions on Robotics*, Vol.22, No.5, 1022/1028(2006).
- 8) F.Fahimi, H.Ashrafuon, C.Nataraj: "Obstacle Avoidance for Spatial Hyper-Redundant Manipulators", *The First Asian Conference on Multibody Dynamics 2002*, 247/254(2002).
- 9) S. Ma, M. Watanabe, H. Kondo: "Dynamic control of curve-constrained hyper-redundant manipulators", *Computational Intelligence in Robotics and Automation*, 2001. Proceedings 2001 IEEE International Symposium, 83/88(2001).
- 10) N. Oda, T. Murakami, K. Ohnishi: "A force based motion control strategy for hyper-redundant manipulator", *Industrial Electronics, Control and Instrumentation*, IECON 97. 23rd International Conference, vol.3, 1385/1390(1997).
- 11) S. Hirose, R. Chu: "Development of a light weight torque limiting M-Drive actuator for hyper-redundant manipulator Float Arm", *Robotics and Automation*, 1999. Proceedings. 1999 IEEE International Conference, Vol.4, 2831/2836(1999).
- 12) H. West and H. Asada: "A Method for the Design of Hybrid Position/Force Controllers for Manipulators Constrained by Contact with the Environment", *Proc. of 1985 IEEE International Conference on Robotics and Automation*, 251/260(1985).
- 13) J. Park and O. Khatib: "Multi-Link Multi-Contact Force Control for Manipulators", *Proc. of 2005 IEEE International Conference on Robotics and Automation*, 3624/3629(2005).
- 14) A. Petrovskaya, J. Park and O. Khatib: "Probabilistic Estimation of Whole Body Contacts for Multi-Contact Robot Control", *Proc. of 2007 IEEE International Conference on Robotics and Automation*, 568/573(2007).
- 15) Z. X. Peng, N. Adachi: "Position and Force Control of Manipulators without Using Force Sensors(in Japanese)", *Trans. of JSME(C)*, Vol.57, 1625/1630(1991).
- 16) T. Ozaki: "Relations between Consumed Energy of Constraint Motion and Contacting Reaction Force(in Japanese)", a graduation thesis of Department of Systems Engineering, Faculty of Engineering, Okayama University, (2012).

複素ニューラルネットワークを用いた広帯域ビームフォーミングによる音響イメージング

○寺林光太郎 廣瀬明 (東京大学)

Acoustic imaging using wide-band beamformer based on Complex-Valued Neural Networks

*Kotaro Terabayashi and Akira Hirose (University of Tokyo)

Abstract— In traditional adaptive array acoustic imaging, only narrowband signal has been used because of the narrowband characteristics of beamformers based on the Huygens' principle. In this paper, we propose a new acoustic imaging method which is suitable for ultra wideband signals. We use a wideband beamforming technique based on complex-valued neural networks to expand power inversion adaptive array (PIAA) method for high resolution and low speckle noise. We show computer simulation results.

Key Words: Wideband, Beamformer, Acoustic imaging, Power inversion adaptive array (PIAA), Complex-valued spatio-temporal neural network (CVSTNN)

1 背景

物体の画像を得る, イメージングを行う手法としては, 光学的手法がもっとも一般的といえよう. 光学的イメージング手法はカメラなど我々にとって大変身近に利用されており, 得られた画像の処理手法と共に盛んに研究も行われている. 一方, 光の代わりに音波を用いて画像を得る手法が, 音響イメージングと呼ばれる手法である. 光学的イメージング手法では物体の表面で反射した光を検出することでイメージを得る. 音響イメージングにおいても, 物体で反射散乱した音波を計測することによって物体を検出し, 画像を得る.

音響イメージングの応用例として, 医療画像診断, 非破壊検査, 海洋探査や, 魚群探査などがよく知られている. これらは水中や固体中など, 光学的手法では画像化が難しいか, 全くできないような場所で用いられている. 特にエコー診断などは我々にとっても大変身近に利用されている技術である. また水中で3次元画像を取得する装置が一般向けの製品として販売されており, 例えば OmniTech 社の EchoScope1600¹⁾ などがある.

音響イメージングでは, 音響ビームを何らかの方法を用いて空間上で走査することにより画像を得る, というのが最も基本的な考え方である. 走査方法として最もシンプルなのは, 指向性を持ったトランスデューサを回転させることである. しかし, この方法では角度分解能が単一のトランスデューサの指向性能に依存してしまうことや, 物理的な回転を行うため時間分解能が低いことなどの問題がある. そのため一般的には, 適応的トランスデューサアレイを用いることで音響ビームの走査を電氣的に行う, ビームフォーミングと呼ばれる技術が用いられる.

ビームフォーミング手法には多くの種類があり, 最も基本的なものである DAS(Delay and Sum) 法や, CAPON 法²⁾ などが広く知られている. CAPON 法では所望方向へのビーム (メインローブ) 以外の方向に発生してしまうビーム (サイドローブ) の寄与を最小化する. この CAPON 法を利用したイメージング手法に関する研究も行われている³⁾⁴⁾⁵⁾.

DAS や CAPON はビームを走査することでイメージ

ングを行う手法である. それに対し近年, MUSIC 法⁶⁾ や, ESPRIT などの超解像と呼ばれる技術を用いることで高い角度分解能を示すイメージング手法の研究も盛んに行われている⁷⁾. これら手法ではヌル (Null: ビームとビームの間で感度が落ち込む点) を走査することでイメージングを行う. ヌルはビームよりも鋭いため, ビーム走査より高い解像度を得ることができる.

しかしこれら全てのビームフォーミング手法において, ビームやヌルのステアリング角は, 音波の波長に依存している. そのため, イメージングに用いることができる音響信号は, 狭帯域に限られてしまう.

そこで本研究では, 広帯域ビームフォーミング手法を用いた音響イメージングを提案する. 広帯域な短パルス信号を用いることにより, レンジ方向の高分解能化とスペckルノイズの低減を図ることができる. これによって, 角度分解能も含め, 高分解能で低雑音の超音波イメージングを実現することができる.

2 従来手法

一般的な音響イメージングでは, イメージング対象に向けて超音波を送信し, その反射波を測定, 解析することで画像を得る.

音波伝搬時間と音速から, 対象までの距離が推定できる. そして送信特性か受信特性, あるいはその両方に空間的選択性を持たせ, それを走査することによって, ターゲットの空間的分布や, 3次元形状を推定することができる.

走査はビームフォーミングにより, 電氣的に行われる.

2.1 ビームフォーミング

最も基本的なビームフォーミング手法である DAS Beamforming について説明する. DAS では複数の受信センサを用いる. ここではリニアアレイを考え, 簡単のため, ターゲットがアレイサイズに比べ十分遠方にあり, アレイへの到来波が平面波に近似できるとする.

ここで, リニアアレイの解析モデルを定義する. アレイの構成を Fig.1 に示す. 素子数を K とし, アレイへの入力信号ベクトルを $\mathbf{x}(t) = [x_1(t) \ x_2(t) \ \dots \ x_K(t)]^T$ とする. また入力信号への位相変換を行う複素ウェイトベクトルを $\mathbf{w} = [w_1 \ w_2 \ \dots \ w_K]^T$ とすると, 入力 \mathbf{x} に

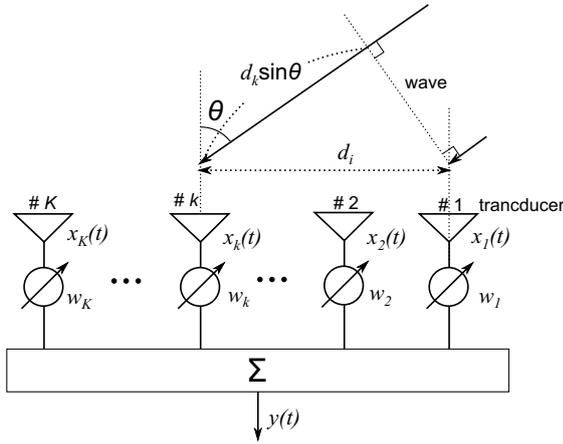


Fig. 1: Linear sensor array

複素ウェイト w をかけたものがアレイ出力 $y(t)$ となり、以下のように表せる。

$$y(t) = \mathbf{w}^* \mathbf{x}(t) \quad (1)$$

ただし、 $*$ は複素共役転置を表す。また入力信号の自己相関行列を $\mathbf{R}_{xx} = E[\mathbf{x}(t)\mathbf{x}^*(t)]$ と定義すると、センサアレイの出力電力 P_{out} が以下のように表せる。

$$P_{out} = \frac{1}{2} E[|y(t)|^2] = \frac{1}{2} \mathbf{w}^* \mathbf{R}_{xx} \mathbf{w} \quad (2)$$

Fig.1 のように音波を受信した場合、素子 $\#k$ と $\#1$ で受信される信号の位相差は、 $2\pi f d_k \sin \theta / c$ となる。ここで f は受信波の周波数、 c は音速、 θ は到来方向を表す。これより各センサの受信信号に対して適切な位相変換を施すことで、受信信号全ての位相を揃えることが可能である。上記のセンサアレイモデルにおいて、 θ 方向から到来する信号に対してアレイ出力を高めるには、複素ウェイトベクトル w を以下のように定めればよい。

$$\mathbf{w} = \begin{bmatrix} \exp(-j2\pi f \frac{d_1}{c} \sin \theta) \\ \exp(-j2\pi f \frac{d_2}{c} \sin \theta) \\ \vdots \\ \exp(-j2\pi f \frac{d_K}{c} \sin \theta) \end{bmatrix} \equiv \mathbf{a}(\theta) \quad (3)$$

このように、 θ 方向からの入力に対し出力パワーを最大化するウェイトベクトルを $\mathbf{a}(\theta)$ と定義し、モードベクトルと呼ぶ。

ここで $\mathbf{a}(\theta)$ は、アレイ形状と、ビームステアリング角 θ 、そして波長 $\lambda = f/c$ にも依存している。このため互いに異なる周波数を持つ信号に対して同じステアリング角 θ を持つようなモードベクトルをつくることはできない。帯域幅をもった受信信号に対し、中心波長にあわせてモードベクトル $\mathbf{a}(\theta)$ を設定すると、帯域の広がりに応じてビームも θ の周囲に広がってしまう。このため、DAS ビームフォーミングを用いてイメージングを行うばあい、高い解像度を得るためには狭帯域な信号を用いることが必要となる。このことは DAS にかぎらない。CAPON 法や、MUSIC 法においても、DAS と同様にモードベクトルが必要になるため、従来の音響イメージングでは、広帯域信号は扱えなかった。

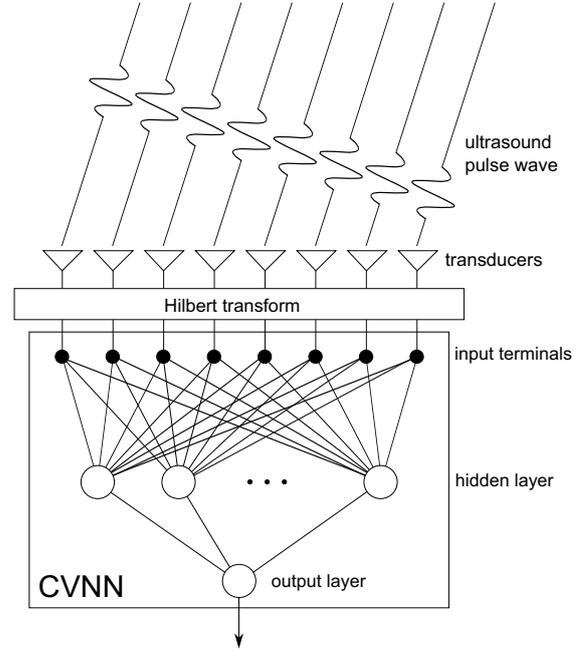


Fig. 2: Complex-valued neural network to realize beamforming of ultra wideband signals

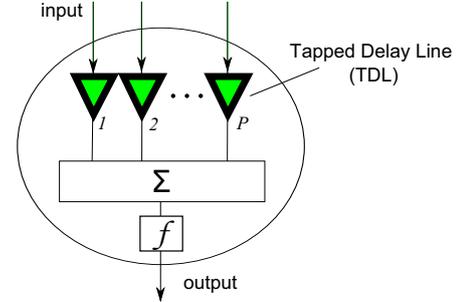


Fig. 3: the complex-valued neuron consisting of tapped delay line.

3 複素ニューラルネットワーク (CVNN) を用いた広帯域ビームフォーミング

提案手法で用いる広帯域音響ビームフォーミング手法と、それを用いた音響イメージングについて説明する。

かつて我々の研究グループにおいて、電波通信用に階層型複素ニューラルネットワークを用いた広帯域なビームフォーミング手法を提案した。そこでは、複素時空間多層ニューラルネットワーク (Complex-valued Spatio-temporal multilayer Neural Network : CVSTNN) が提案されている⁸⁾。ここでは、それを音波に適用し、さらにイメージングへの応用を提案する。

提案手法で用いる複素ニューラルネットワーク (CVNN) の概要を Fig.2 に示す。ニューラルネットワークは入力層、中間層、出力層の3層からなる階層構造になっており、各ニューロンを円で示している。

センサアレイで受信された信号はまずヒルベルト変換により解析信号へと変換される。ヒルベルト変換は、入力波形の正周波成分の位相を -90 度、負周波成分の位相を 90 度進ませる。ヒルベルト変換 \mathcal{H} 、受信信号を \mathbf{x} と表すと、解析信号は $(1 + j\mathcal{H}) \mathbf{x}$ で得られる。

こうして得られた解析信号がニューラルネットワークに入力され、入力層、中間層、出力層と順に伝搬する。

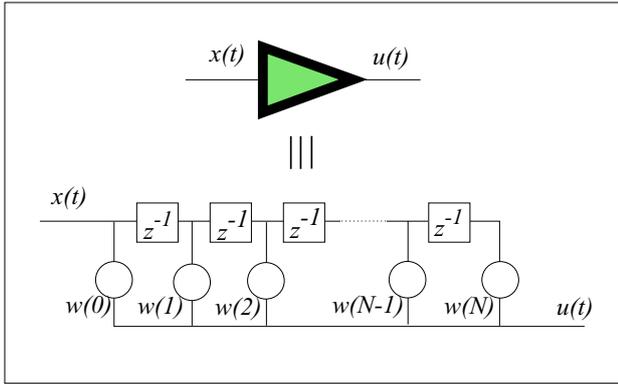


Fig. 4: Tapped delay line(TDL).

Fig.2 内に円で示した各ニューロン内の処理を図示したものが Fig.3 である. 各ニューロンで, すべての入力信号にそれぞれ振幅, 位相変調をかけて (ウェイトをかけて) 足しあわせ, 最後に活性化関数 $f(x) = \tanh(|x|) \exp(i \arg(x))$ を通して -1 から 1 の範囲に制限された出力を出力する.

ニューロン内で位相振幅変調を行う三角形で示した素子を, Tapped delay line(TDL) とよぶ. TDL の概要を Fig.4 に示す. TDL 内の z^{-1} は遅延素子を表す. $w(n)$ は位相振幅ウェイトであり, $w(n) = a(n) \exp(j\theta(n))$ と表せる. TDL の出力は, TDL に含まれる遅延素子による過去から現在までの入力の重み付け和であり, 過去の入力を記憶し, 反映させた出力を得ることができる.

$$y(t) = \sum_{n=0}^{N-1} x(t-n)w(n) \quad (4)$$

この TDL 内のウェイト $w(n)$ をニューロン学習によって更新し, 各センサからの受信信号に対して, 遅延, 振幅, 位相を適応的に変調することで, 広帯域信号のビームフォーミングを実現する.

3.1 ニューロン学習プロセス

CVNN 内のニューロン学習のプロセスについて説明する. ニューロン学習プロセスの概要を Fig.5 に示す.

はじめに学習に用いる教師信号を用意する. 教師信号は, CVNN への入力となる教師入力信号と, それに対する CVNN の理想的な出力値である教師出力信号からなる. 教師入力信号を CVNN に入力して, 順方向へ伝搬させ, 同時に教師出力信号をバックプロパゲーション則 (5) にのっとして, 出力層側から入力層へ向けて逆伝搬させる. 第 l 層の出力を y_l , 教師出力信号を \hat{y}_l , ウェイト $\mathbf{W}_l = [w_1 \ w_2 \ \dots \ w_P]$ に対し, $v = w/|w|^2$ とするとき, \hat{y}_l を逆伝搬させた, $l-1$ 層の教師出力信号 \hat{y}_{l-1} は

$$\hat{y}_{l-1}^* = f(\hat{y}_l^* \mathbf{V}_l) \quad (5)$$

とすることができる. 各層の出力部分において, 順方向に伝搬してきた信号と, 逆伝搬させた教師信号の誤差をとり, 次の更新則に従ってその層のウェイトを更新する⁹⁾.

$$w_{ji}(t+1) = |w_{ji}(t+1)| \exp(j\theta_{ji}(t+1)) \quad (6)$$

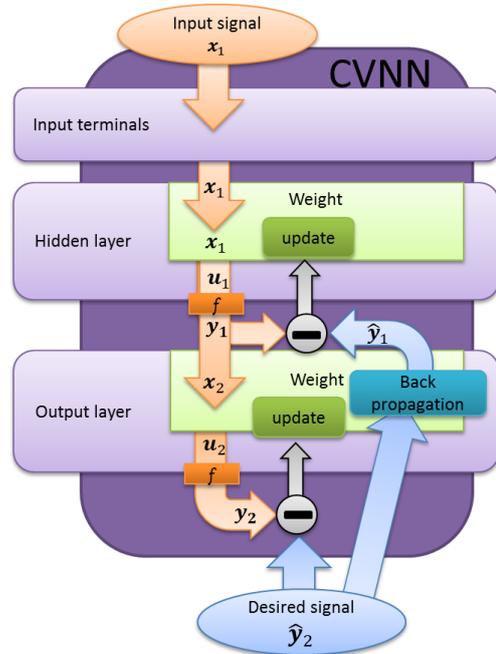


Fig. 5: Complex-valued backpropagation learning.

$$\begin{aligned} |w_{ji}(t+1)| &= |w_{ji}(t)| \\ &\quad - K \left((1 - |y_j|^2) \right. \\ &\quad \left. (|y_j| - |\hat{y}_j| \cos(\theta_j - \hat{\theta}_j)) |x_i| \cos \theta_{ji}^{\text{rot}} \right. \\ &\quad \left. - |y_j| |\hat{y}_j| \sin(\theta_j - \hat{\theta}_j) \frac{|x_i|}{|u_j|} \sin \theta_{ji}^{\text{rot}} \right) \end{aligned} \quad (7)$$

$$\begin{aligned} \theta_{ji}(t+1) &= \theta_{ji}(t) \\ &\quad - K \left((1 - |y_j|^2) \right. \\ &\quad \left. (|y_j| - |\hat{y}_j| \cos(\theta_j - \hat{\theta}_j)) |x_i| \sin \theta_{ji}^{\text{rot}} \right. \\ &\quad \left. + |y_j| |\hat{y}_j| \sin(\theta_j - \hat{\theta}_j) \frac{|x_i|}{|u_j|} \cos \theta_{ji}^{\text{rot}} \right) \end{aligned} \quad (8)$$

このようなウェイトの更新を繰り返し行うことで, CVNN の学習が進み, 教師入力信号に対する CVNN の出力は, 理想値である教師出力信号に近づいていく. 任意方向 θ から到来する広帯域信号 (短パルス波) を教師入力信号とし, 教師出力信号を入力と同様のパルス波としたり, 無音 (0 で一定) としたりすることで, θ 方向に対してビームを向けたり, ヌルを向けたりすることができる.

4 CVSTNN-PIAA 信号到来方向推定

第3章で示したビームフォーミング手法を用いた信号到来方向推定手法による音響イメージの生成方法を説明する. 特に本報告では, 離散的な対象物体を仮定し, ヌルを適応的に向ける事によるイメージング手法を提案する. これは, いわゆるパワーインバージョン法 (Power Inversion Adaptive Array : PIAA)¹⁰⁾ の考え方を, CVSTNN を利用することにより拡張したものである.

空間上に M 個の超音波散乱体が離散的に分布しているとす. その空間中に超音波超短パルスを送信すると, パルスは各散乱体により散乱され, それぞれセンサ

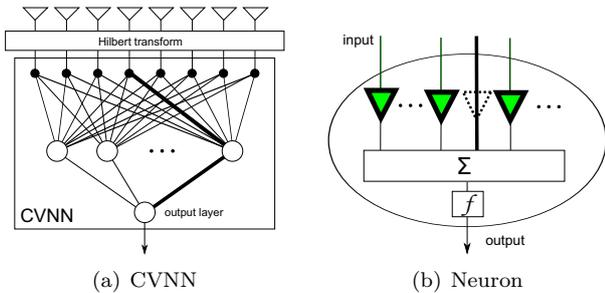


Fig. 6: Direct connection without TDLs along the thick lines

アレイへと到達する.すると,アレイで受信される時系列の信号には,最大 M 個のパルスが現れる.こうして得られた信号に対して,到来方向を推定することで,空間全体の離散散乱体の位置を推定し,画像化する.

受信波形を教師入力信号とし,それに対する教師出力信号は,常に 0 で一定とする.こうして M 個のパルス全てについて CVNN にヌル生成を学習させる.

このように学習が進められると, CVNN の指向特性は, M 個の散乱体の方向に対してヌルが向けられるようになる.学習後の CVNN の指向特性の逆数をとった時,ピークとなる方向(ヌル方向)が散乱体の位置を推定したものとなる.

ここで学習の自明な収束点として,ウェイトの絶対値がすべてが 0 となる場合,つまりどのような入力に対しても 0 を掛けて,出力が必ず 0 になってしまうような点に学習が収束しないようにする必要がある.そこでニューロン内の TDL を一部取り除き, TDL 処理を加えないようにすることで,自明な収束点を回避する.具体的には, Fig.6(a) および (b) に太線で示した伝搬路を伝わる信号の処理をしていた TDL を取り除き,ニューロンへの入力を加算器に直結する.こうすると, CVNN の出力には,入力信号のうち一つにのみ依存する項が現れる.よって自明な収束点がなくなり,学習はこの項を打ち消すように進められる.

5 シミュレーション

提案手法を用いた水中での広帯域音響イメージングのシミュレーション実験を行った.広い帯域幅を持つパルス入力信号を用意し,それを教師入力信号として CVNN の学習をおこない,学習後の CVNN の指向特性を評価することで,入力信号の到来方向を推定し,イメージングを行う.

5.1 実験方法

無限に広く,音速が一定 (1500m/s) の水中を考える.センサアレイには,64 個の無指向センサを 8×8 の等間隔に配置した 2 次元正方センサアレイを用いる.このセンサアレイ中の素子の配置を Fig.7 に示す.また, Fig.8 に示すように M 個の散乱体 $S_m (1 \leq m \leq M)$ が配置されている.散乱体は無限遠にあり,アレイに到達する信号は平面波であると考え.また,減衰,雑音もないものとする.

パルス波形は 500kHz の sin 波に,2 周期分の幅をもったハン窓をかけたものとする. Fig.9 にこのパルスの波形を示す.またこの波形のパワースペクトル密度を Fig.10 に示す.中心周波数 500kHz で,比帯域が 74% であり,広い帯域を持った信号であることがわかる. 500kHz で

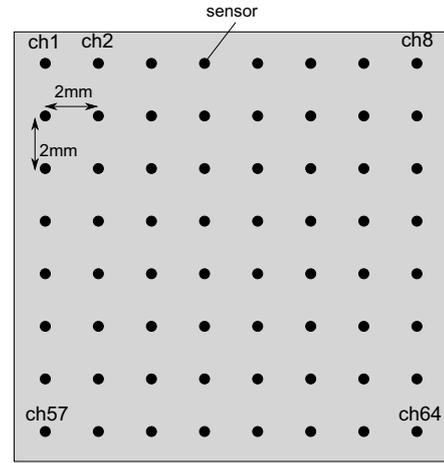


Fig. 7: Sensor layout.

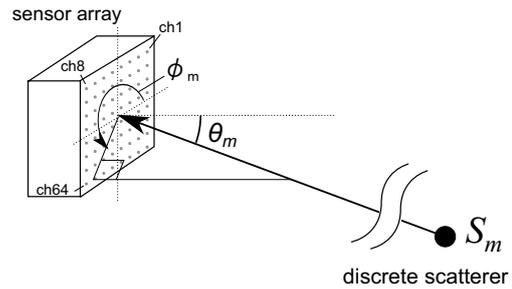


Fig. 8: Schematic representation of sensor array and scatterers.

の波長 λ は, $\lambda = 1/500 \times 1500 = 3\text{mm}$ である.

このような短パルス信号を空間に 1 つ送信する.パルスは各散乱体で散乱され,その散乱波をアレイで受信できる.アレイの各センサで得られる計 64 個の受信波形にはそれぞれ M 個のパルスが含まれる.またこの M 個のパルスはそれぞれ,その到来方向に応じて,素子ごとにずれた時刻にあらわれている.このような受信波形の中から,パルス信号部分のみを一つ一つ切り出し, M 組の教師入力信号とする.また教師出力信号は常に 0 としてニューラルネットワークに提示する.同一の CVSTNN に対して,この M 組の教師信号を 1 回ずつ順番に用いて計 M 回学習させる.この学習サイクルを繰り返すことにより, M 個の散乱体の方向全てにヌルが向くように学習をすすめる.

ここでは各散乱体からの散乱波はセンサアレイ上で互いに干渉せず,受信波形から M 個のパルスが全てを重なりなく切り出せると仮定している.

今回は点散乱体が一箇所, $(\phi, \theta) = (0, 10)\text{deg}$ 方向にある場合と, $(0, 15), (180, 15)\text{deg}$ の二箇所にある場合についてシミュレーションを行い,それぞれ 3000 サイクルの学習を行った.

5.2 結果

はじめに,散乱体がただ一つ $(\phi, \theta) = (0, 10)\text{deg}$ 方向にある場合についてのシミュレーション結果を示す.まず, Fig.11 にニューラルネットワークの学習曲線を示す.学習サイクルごとに,教師入力信号に対する CVSTNN の出力と,教師出力信号 (= 0) との平均二乗誤差をプロットしたものである.回数を重ねると誤差は小さくなり,低い値で安定することから,学習がうまく収束していることがわかる. Fig.12 は学習済み CVSTNN の指

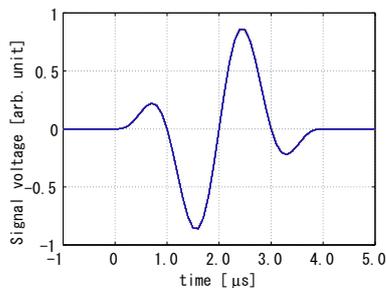


Fig. 9: Input pulse wave form.

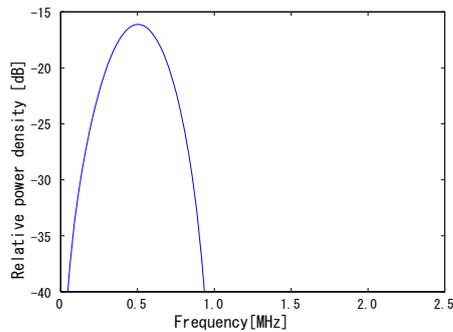


Fig. 10: Input pulse's power spectrum.

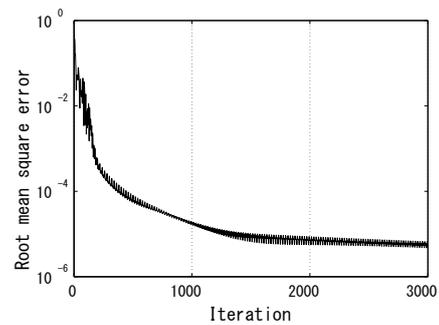


Fig. 11: Learning curve

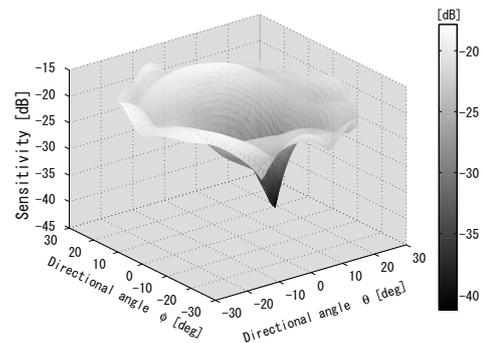


Fig. 12: Directional characteristic(single object).

向特性を表している。学習済みの CVSTNN に対して、学習に用いたものと同様のパルス信号をあらゆる到来方向から入力する。その時出力される信号の振幅の最大値を、到来角ごとにプロットしたものである。これを見ると、およそ $(\phi, \theta) = (0, 10)$ 方向で CVNN の出力が最も低くなっており、到来方向へヌルがうまく向けられていることが確かめられる。

指向特性の逆数をとった反転指向特性を Fig.13 に示す。この指向特性のピーク位置が、パルス到来方向の推定結果、つまり散乱体の位置推定結果を示している。

この指向特性を輝度変換し、極座標にマッピングしたものが Fig.14 である。設定した位置 $(0, 10)$ deg に像が得られている、また同時に、設定した方向以外にはどこにも像が現れていない。以上のことから、ただ 1 つの散乱体の像の取得ができたといえる。

次に、散乱体が $(\phi, \theta) = (0, 15), (180, 15)$ deg 方向の 2 つある場合のシミュレーション結果について述べる。学習済み CVSTNN の反転指向特性を Fig.15 に示す。これを見ると、設定した 2 つの方向にのみピークが現れている。散乱体が 2 つの場合でも、正しく方向推定することができ、像が得ることが確かめられた。

Fig.16 は、 $(\phi, \theta) = (0, 15)$ deg 周辺を拡大した画像である。輝度が最大となる点は $(\phi, \theta) = (-1.5, 15.2)$ deg である。 $\phi = -1.5$ deg で一定の時の CVSTNN の θ 指向特性を表したものが Fig.17 である。同様に、 $\theta = 15.2$ deg で一定の時の ϕ 特性を表したものが Fig.18 である。それぞれにおいて、ピークからの電力半減幅 $\Delta\theta, \Delta\phi$ を見ると、 $\Delta\theta = 1.2$ deg、 $\Delta\phi = 5.2$ deg であり、これらが角度分解能である。これらの分解能の値を決める要因についても、発表で議論する。

6 むすび

本稿では、複素ニューラルネットワークを用いたビームフォーミング手法を用い、PIAA による到来方向推定手法を拡張したイメージング手法を提案した。また提

案手法を用いてシミュレーションを行い、広帯域なパルス信号でのイメージングができることを確かめた。

参考文献

- 1) M.Vittorio and T.Andrea: "Three-dimensional image generation and processing in underwater acoustic vision," Proceeding of the IEEE, 1903/1946, (2000)
- 2) J.Capon: "High-resolution frequency-wavenumber spectrum analysis," Proceedings of the IEEE, vol.57, no.8, 1408/1418, (1969)
- 3) Z.Wang, J.Li, P.Stoica, T.Nishida, and M.Sheplak: "Constant-beamwidth and constant powerwidth wideband robust Capon beamformers for acoustic imaging," The Journal of the Acoustical Society of America, vol.116, 1621/2631, (2004)
- 4) I.K.Holfort, F.Gran and J.A.Jensen: "Minimum variance beamforming for high frame-rate ultrasound imaging," IEEE International Ultrasonics Symposium, 1541/1544, (2007)
- 5) J.Synnevag, A.Austeng and S.Holm: "Benefits of minimum-variance beamforming in medical ultrasound imaging," IEEE transactions on ultrasonics, ferroelectrics, and frequency control, vol.56, no.9, 1868/1879, (2009)
- 6) R.Schmidt: "Multiple emitter location and signal parameter estimation," Antennas and Propagation, IEEE Transactions on, vol.34, no.3, 276/280, (1986)
- 7) Y.Yoon and M.G.Amin:"High-Resolution Through-the-Wall Radar Imaging Using BeamSpace MUSIC," IEEE Transactions on Antennas and Propagation, vol.56, 1763/1774, (2008)
- 8) A.B.Sukusmono, A.Hirose : "Beamforming of ultra-wideband pulses by a complex-valued spatio-temporal multilayer neural network," International Journal of Neural Systems vol.15, Nos. 1 & 2 (2005)
- 9) A.Hirose: "Complex-Valued Neural Networks," Springer (2006), 2nd Edition (2012)
- 10) R.T.Compton : "The Power Inversion Adaptive Array : Concept and Performance," IEEE Transactions on Aerospace and Electronic Systems, vol.AES-15, No.6, 803/814, (1979)

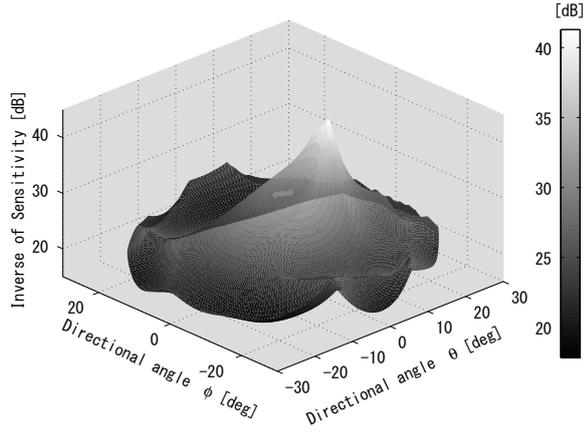


Fig. 13: Inverse of directional sensitivity (single object).

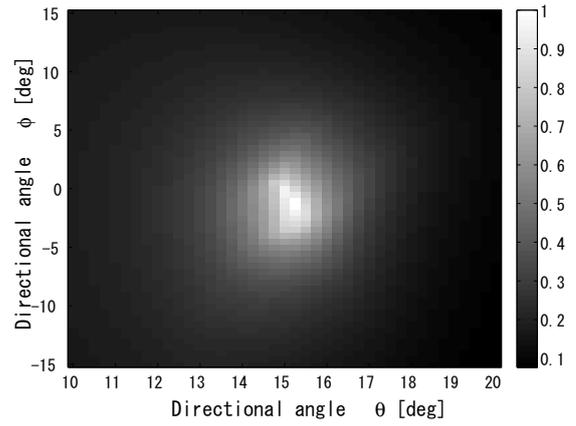


Fig. 16: Obtained image at around (0, 15)deg.

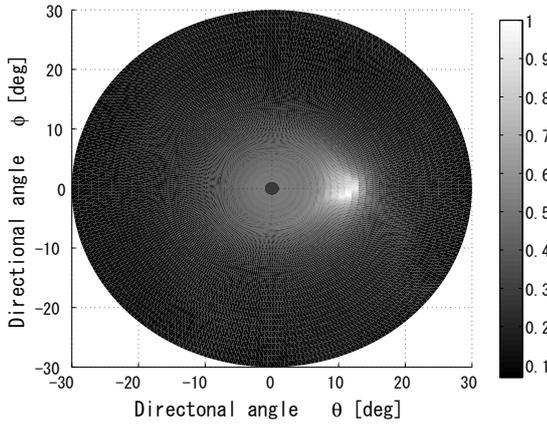


Fig. 14: Obtained image (single object).

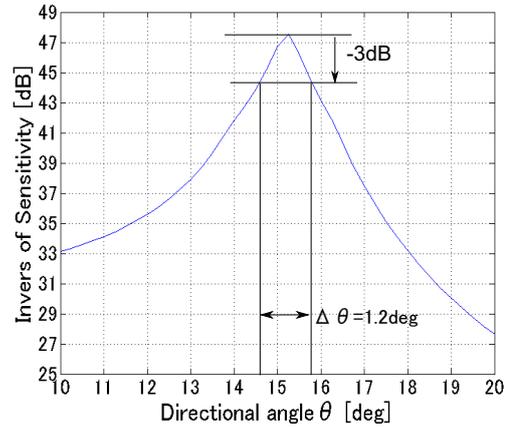


Fig. 17: Directional characteristics ($\phi = -1.5$ deg).

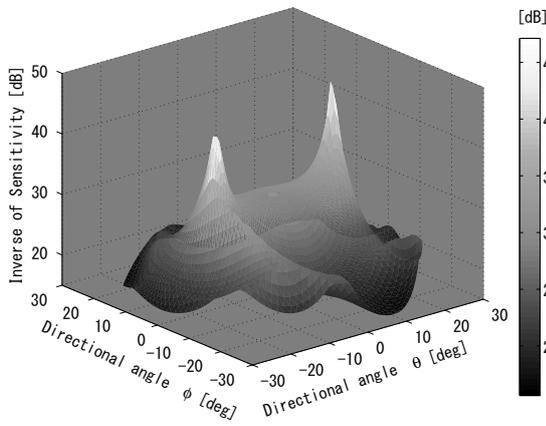


Fig. 15: Inverse of directional sensitivity(2 objects).

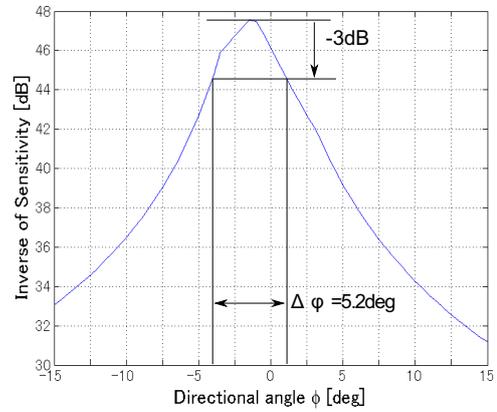


Fig. 18: Directional characteristics ($\theta = 15.2$ deg).

複素ニューラルネットワークのハードウェア化

○前田裕 安藤充弥 伊藤秀隆 (関西大学)

Hardware Implementation of complex-valued Neural Network

* Y. Maeda, M. Ando and H. Ito(Kansai University)

Abstract—Recently, complex-valued neural networks draw the attention because of their rich capability for information processing. Then implementation methodology including learning scheme is important and interesting. This paper describes a Field Programmable Gate Array (FPGA) implementation of a pulse density complex-valued neural network system with learning capability using the simultaneous perturbation method. Details of the system design with learning scheme are explained.

Key Words: FPGA, Complex-valued neural networks, Simultaneous Perturbation, Learning scheme

1 はじめに

近年、複素ニューラルネットワークは、幅広く研究され、通信方式や画像処理への応用やハードウェアでの実現も期待されている^{1),2)}。一般的に複素ニューラルネットワークの実現はソフトウェアによることが多い。しかし、ソフトウェアによる実現では信号の並列処理ができず、動作速度にも制限がある^{3),5)}。複素ニューラルネットワークの幅広い応用を考える場合、ハードウェアによる実現が望まれる。

一方、ハードウェアによる実現においても、アナログ実現かデジタル実現化などいくつかの選択肢がある。さらに、学習機能も含めた実現を考えると、ハードウェアによる実現には困難が多い。ハードウェア化に適した学習則とその実現法の検討が必要である。

パルス密度による数値表現は、回路化に際して、アナログ量をデジタル回路により実現できるという利点を持つ^{4),6)}。四則演算がANDやORなどの論理演算で実現できることにより、回路の簡素化が図れる。近年のFPGAの集積度の向上やEDAの発展と相まって、パルス密度によるニューラルネットワークの実現の容易性が増してきている^{4),6)}。

さらに、学習則についても、一般に用いられる複素バックプロパゲーションをハードウェア化することは回路の実現上大きな困難を伴う^{3),5)}。

以上の観点から、本研究では、同時摂動最適化法⁷⁾に基づいた学習機能を有するパルス密度型複素ニューラルネットワークのFPGAによるハードウェア化を想定してHDLを用いた設計を行った。

2 複素ニューラルネットワーク

階層型のニューラルネットワークを考える。複素ニューラルネットワークは、複数列の複素ニューロンから構成されており、入力信号、重み、しきい値及び出力信号はすべて複素数である。複素ニューロン l の内部ポテンシャル Z_l は

$$z_l = \sum_{k=1}^n x_k w_{lk} - \theta_l \quad (1)$$

と定義する。ここで、 Z_l は、ある層の l 番目のニューロンへの出力値、 w_{lk} は全層の k 番目のニューロンとこの l 番目のニューロン間の荷重、 x_k は前層の k 番目のニ

ューロンからの入力値、 θ_l は閾値である。出力信号 f は次のよう定義される^{1),2)}。 i は虚数単位である。

$$f(z_l) = f(x_1) + f(x_2)i \quad (2)$$

$$Z_l = x_1 + x_2i \quad (3)$$

$$f(x_l) = \frac{1}{1 + \exp(-x_l)} \quad (4)$$

3 パルス密度ニューラルネットワーク

パルス密度方式では、ある特定区間を定め、その区間に発生できるパルスの最大本数に対する実際の本数の割合によって値を表現する。パルス密度による表現は、乗算や加算などの演算をANDとORの演算素子のみで実現できる。したがって、ニューロンの数や、入力数が多くなっても高速かつ簡便な回路実現が可能となる^{3),6)}。

また、アナログ値をデジタル回路により取り扱うため、耐ノイズ性の向上や、システムが大規模化した場合でも安定動作するという利点があり、EDAを用いて容易に設計を行うことができる。

さらに、他の外部システムとの接続を考えた場合、これが、PCに代表されるように、デジタル回路である場合が多い。この場合、インターフェースの容易性を考えると外部システムとの親和性も高い。

本研究では、Fig.1に示されるように、ひとつの複素数の実部と虚部に対して、それぞれ、値そのものをパ

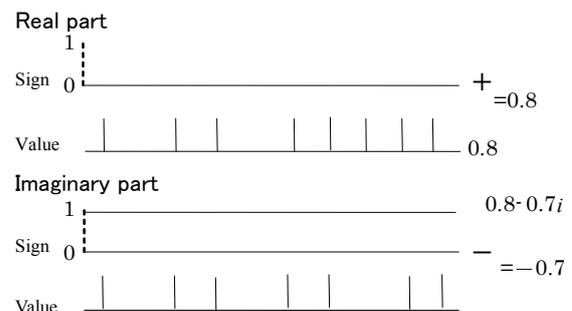


Fig. 1: Pulse density representation.

ルス密度表現した信号と、その符号を表す信号を割り当てる。このように表現された複素数の演算は、後で述べるような処理を行うことで実現する。

4 同時摂動学習則

ニューラルネットワークにおいて、学習機能は重要である。一方、その回路の実現には、困難を伴う。そこで、本研究では、同時摂動を用いたニューラルネットワークの学習則を用い、学習機能も含めて回路化する。

同時摂動最適化法は、差分近似の拡張として、パラメータの次元が高い場合でも、評価関数に対する観測回数を増やすことなく、勾配を推定する手法として考案された確率的な勾配法である。

一般にニューラルネットワークの学習則はバックプロパゲーション法を使用する。しかし、この手法をハードウェア的に実現する場合、回路の複雑性、配線の輻輳など多くの問題が生じる。そこで、同時摂動最適化法を使用し回路の簡素化を行う。同時摂動による学習法をパルス密度方式と併用することにより、学習機能の実現における回路の簡素化などの点でも利点が生じる³⁾⁻⁵⁾。

また、符号ベクトルによる同時摂動学習則では、学習機構を乱数回路と加算器のみから構成することができ、ハードウェア化がさらに容易になる。本研究で用いた符号ベクトル型同時摂動学習則は、つぎのように表される。

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \alpha \Delta \mathbf{w}_t \quad (5)$$

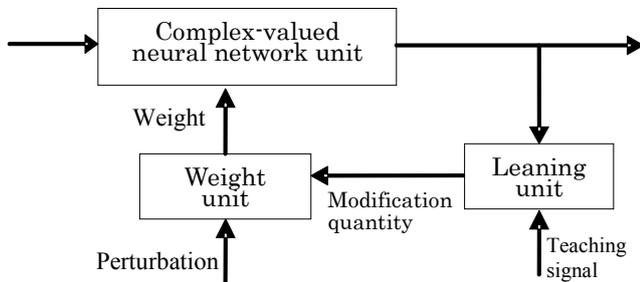


Fig. 2: System configuration.

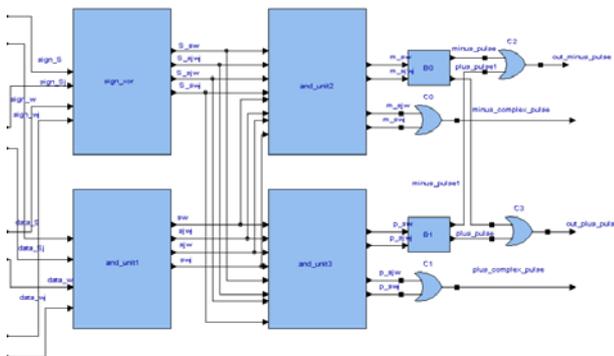


Fig. 3: Multiplication of complex values in pulse density.

$$\Delta \mathbf{w}_t = \frac{J(\mathbf{w}_t + c\mathbf{s}_t) + J(\mathbf{w}_t)}{c} \mathbf{s}_t \quad (6)$$

ここで、 \mathbf{w} は荷重、 $J(\cdot)$ は評価関数、 c は摂動の大きさ、 \mathbf{s}_t は符号ベクトルで、その要素はランダムに生成される+1あるいは-1の値である。 $\Delta \mathbf{w}_t$ は荷重の修正量で、評価関数の勾配の推定値を表している。

5 システム構成

設計した本システムは、複素ニューラルネットワーク部、学習部、荷重部の三つのユニットから構成され

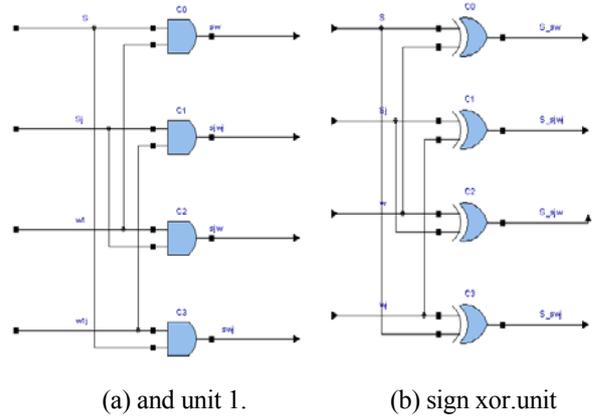


Fig. 4: Components of the system 1.

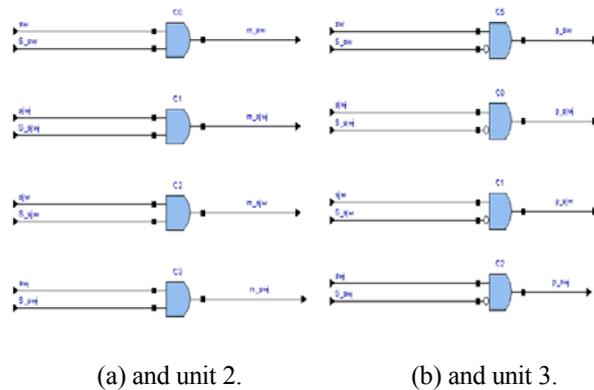


Fig. 5: Components of the system 2.

```

architecture TEST of B0 is
begin
  process(sw,sjwj)
  begin
    if(sw >= sjwj)then
      output_pulse 1 <= '0';
      output_pulse <= sw xor sjwj;
    else
      output_pulse 1 <= sjwj;
      output_pulse <= '0';
    end if;
  end process;
end;

```

Fig. 6: Subtraction of pulse.

ている。Fig.2 にシステム全体の構成を示す。RTL の設計、論理合成および配置配線には、それぞれ、Visual Elite、Precision_Synthesis RTL および Quartus II を用いた。

評価関数の設定は、学習則としての観点からだけでなく、システム的设计上重要である。パルス密度による表現の特徴を十分に生かすためには、通常用いられる実部と虚部のそれぞれの目標値に対する二乗誤差の和を用いるのではなく、絶対値の和を用いるのが有効である。これにより、カウンタを用いることでパルス密度表現されたニューラルネットワークの出力から誤差関数の値を容易に求めることができる。

$$J(\mathbf{w}) = \sum_{j=1}^n |t_{rj} - o_{rj}| + \sum_{j=1}^n |t_{cj} - o_{cj}| \quad (7)$$

ここで、 o_{rj} と o_{cj} は、それぞれ、複素ニューラルネットワークの j 番目の出力の実部と虚部を表し、 t_{rj} と t_{cj} は、それに対応する教師信号を表す。本研究では、式(7)を誤差関数として用いた。

5.1 複素ニューラルネットワーク部

複素ニューラルネットワーク部は、外部からの複素入力を受け取り、ニューラルネットワークによる前向き計算を行い、この結果を出力している。この前向き計算は、式(1)の複素荷重と複素入力の積和演算と、式(2)-(4)の入出力関数処理を行うことからなる。

Fig.3 に、パルス密度形式で表現された複素数の乗算の構成図を示す。Fig.4 と Fig.5 に Fig.3 で用いられている and unit 1、and xor unit、and unit 2 および and unit 3 の構成を示す。

この図において、入力は符号とパルス列で表現された複素数である。すなわち、入力 s と荷重 w の実部、虚部の四つのパルス入力と、それぞれに対応する四つの符号信号が入力である。

荷重 w と入力 s の実部、虚部のパルス密度信号は、それぞれ、and unit 1 で AND 演算され、乗算の処理が行われる(Fig.4(a)参照)。また、このときの符号は、xor unit で XOR 演算され、決定される(Fig.4(b)参照)。

つぎに、これらの信号は、セクタである and unit 2 および and unit 3 で正のパルスと負のパルスに分けられる(Fig.5(a)および Fig.5(b)参照)。

ここで、これらの信号は、C0、C1 の OR 演算器では虚部の計算が実行される。B0、B1、C2、C3 の演算器では、実部の計算が行われる。このとき、B0 および B1 については、Fig.6 で示されるようなパルスの減算の計算が実行されている。

出力は、正のパルス列と負のパルス列で表された複素数の実部と虚部である。この正と負に分けられた出力を、実部、虚部別にカウンタのアップカウントとダウンカウントに入力することにより、その値を求めることができる。さらに、複数の複素入力がある場合は、Fig.3 の構成の回路を複数準備し、その出力を OR 演算素子で接続する。

入出力関数の実現は、ニューラルネットワークのハードウェア化においては困難な問題のひとつである。しかしながら、パルス密度による表現では、その表現自身が飽和特性を有しており、OR 演算におけるパルスの加算時の誤差特性を考慮すると、特段の処理を行

うことなく、近似的にシグモイド関数を実現していることになる。このことも、パルス密度による表現を用いる利点のひとつである。

このような構成を用いることで、パルス密度表現された複素数の演算を実行する。

5.2 学習部

先に述べたように、複素バックプロパゲーションが学習則としては一般的である。しかしながら、これをハードウェアにより実現することは、特に、誤差の逆伝搬の実現には困難を伴う。このため、ハードウェア化に適した学習則を用いることが必要である。

この学習部は、複素ニューラルネットワーク部の出力と教師信号を比較し誤差を求め、これをもとに同時摂動学習則を用いて荷重修正量を計算し、荷重部へ出力する。摂動がある場合とない場合の複素ニューラルネットワーク部からの出力から、式(7)による誤差関数の値を計算する。つぎに、式(6)に基づいて荷重の修正量の計算を行う。

式(7)の誤差関数の計算では、複素ニューラルネットワーク部からのパルス信号がアップカウンタに送られ、対応する教師信号のパルスがダウンカウントされることにより絶対値誤差を求める。

式(5)および式(6)からわかるように、各荷重の修正量のうち $\alpha(J(\mathbf{w}+cs) - J(\mathbf{w}))/c$ の部分は共通である。

また、 α/c を 2 の乗数で表現できる量に制限することにより、 α/c の乗除算をビットシフトの操作に置き換えることができ、演算回路の簡素化と計算速度の向上が期待できる。

これらの学習部の計算の流れを Fig.7 に示す。この

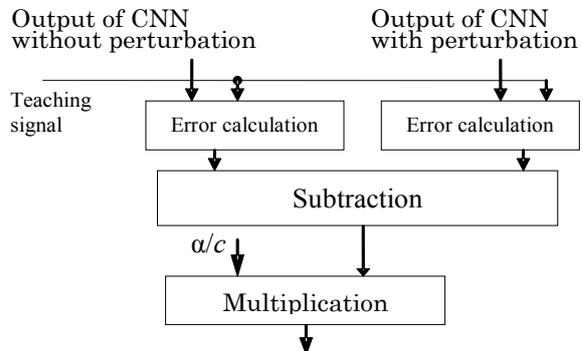


Fig. 7: Learning unit.

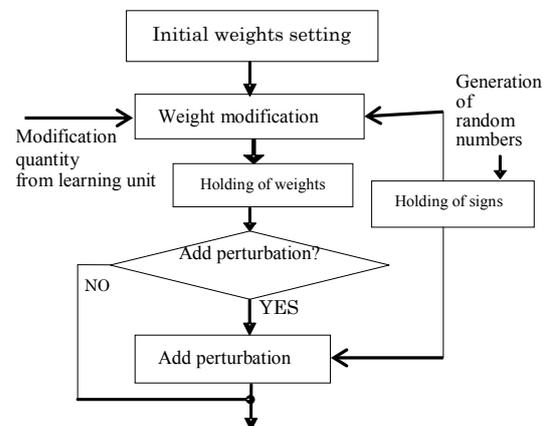


Fig. 8: Weight unit.

高次元ネットワークインバージョンによる逆問題解法

○井浦 翼 小川毅彦 (拓殖大学)

Solution of Inverse Problems Using High-Dimensional Network Inversion

* T. Iura and T. Ogawa (Takushoku University)

Abstract—Recently, the neural network model for handling the high dimensional numbers has been proposed and studied in the field of artificial neural networks. The quaternion is focused in a number of fields such as computer graphics, because we easily express the algebra in three-dimensional space using quaternion. On the other hand, the problem which searches for a cause from an observed result is called an inverse problem, and has been studied in various fields. The network inversion has been studied as a neural network based solution for inverse problems. In this paper, the extension of the network inversion to high-dimensional neural networks is examined by the computer simulations of the complex-valued and quaternion network inversions.

Key Words: high-dimensional numbers, network inversion, inverse problems.

1 はじめに

近年、複素数をはじめ高次元数を扱うためのニューラルネットモデルや学習法が提案されている¹⁾²⁾。高次元ニューラルネットワークは、高次元数入出力間の関係を学習・推定することができる。高次元数の中でも特に四元数は、三次元空間における代数表現を容易に行うことができるため、コンピュータグラフィックスなどの分野で注目されている。

一方、観測結果から原因を求める問題は逆問題と呼ばれ、さまざまな分野で研究が行われており³⁾、そのニューラルネットワーク解法としてネットワークインバージョンの方法が研究されている⁴⁾。ネットワークインバージョンによる逆問題解法として、画像処理や逆運動学問題への応用が検討されてきた⁵⁾⁶⁾。また、ネットワークインバージョンの複素領域への拡張が行われた⁷⁾。

本研究では、高次元空間における逆問題解法のために、複素数や四元数などの高次元数に拡張した高次元ネットワークインバージョン法を示す。高次元空間における逆写像問題およびロボットアーム逆運動学問題のシミュレーションによって動作を示す。

2 高次元数とニューラルネットワーク

四元数は W.R.Hamilton によって考案された高次元数で、実部と 3 つの虚部から成る。3 つの虚数単位を i, j, k とすると四元数は $x = x_1 + ix_2 + jx_3 + kx_4$ と表現される。 x_1, x_2, x_3, x_4 は実数であり四元数の各成分を表す。また $i^2 = j^2 = k^2 = ijk = 1, ij = -ji = k, jk = -kj = i, ki = -ik = j$ である。四元数は積に対して結合法則を、和に対して分配法則を満たす。

多層型ニューラルネットワークおよびその学習法の高次元数への拡張が行われている。複素数及び四元数ニューロンモデルについてはさまざまなものが検討されているが、本研究では簡単化のため、複素数や四元数の入力荷重和の各要素に独立にシグモイド関数を適用するモデルを考える。

まず複素数への拡張として、複素ニューラルネットワークとその学習法を考える。このモデルでは、複素荷重および複素ニューロンによって複素入出力関係を表現する。複素荷重については複素数に拡張された勾

配法に基づく誤差逆伝搬法で、

$$\begin{aligned}w_R(n+1) &= w_R(n) - \varepsilon_t \left(\frac{\partial E_R}{\partial w_R} + \frac{\partial E_I}{\partial w_R} \right) \\w_I(n+1) &= w_I(n) - \varepsilon_t \left(\frac{\partial E_I}{\partial w_I} - \frac{\partial E_R}{\partial w_I} \right)\end{aligned}\quad (1)$$

として繰り返し修正を行う。ここでは複素数ニューロンの実部および虚部に独立に変換関数を適用するモデルを考え、出力誤差を $E = E_R + iE_I$ 、荷重を $w = w_R + iw_I$ 、学習係数を ε_t とする。

さらに四元数への拡張として、四元数ニューラルネットワークとその学習法を考える。このモデルでは、四元数荷重および四元数ニューロンを用いて四元数入出力関係を表現する。誤差逆伝搬学習法の四元数への拡張として、

$$\begin{aligned}w_1(n+1) &= w_1(n) - \varepsilon_t \left(\frac{\partial E_1}{\partial w_1} - \frac{\partial E_2}{\partial w_1} - \frac{\partial E_3}{\partial w_1} - \frac{\partial E_4}{\partial w_1} \right) \\w_2(n+1) &= w_2(n) - \varepsilon_t \left(\frac{\partial E_1}{\partial w_2} + \frac{\partial E_2}{\partial w_2} + \frac{\partial E_3}{\partial w_2} - \frac{\partial E_4}{\partial w_2} \right) \\w_3(n+1) &= w_3(n) - \varepsilon_t \left(\frac{\partial E_1}{\partial w_3} - \frac{\partial E_2}{\partial w_3} + \frac{\partial E_3}{\partial w_3} + \frac{\partial E_4}{\partial w_3} \right) \\w_4(n+1) &= w_4(n) - \varepsilon_t \left(\frac{\partial E_1}{\partial w_4} + \frac{\partial E_2}{\partial w_4} - \frac{\partial E_3}{\partial w_4} + \frac{\partial E_4}{\partial w_4} \right)\end{aligned}\quad (2)$$

として繰り返し修正を行うことで、四元数荷重を修正する。ここでは四元数ニューロンの各部に独立に変換関数を適用するモデルを考え、出力誤差を $E = E_1 + iE_2 + jE_3 + kE_4$ 、荷重を $w = w_1 + iw_2 + jw_3 + kw_4$ 、学習係数を ε_t とする。

3 逆問題とネットワークインバージョン

順問題が原因から結果を導くものであるのに対し、逆問題は結果から原因を求める、あるいは出力から入力を求めるものである。

通常の多層型ニューラルネットワークは、入力から出力への方向で学習を行い、学習によって得られた順

方向の関係を用いて入力から出力を推定しているため、順問題の解法であると言える。これに対し、学習によって得られた順方向関係を逆に用いて出力から入力を推定することによって、多層型ニューラルネットワークを用いて逆問題を解く方法がネットワークインバージョンである。

ネットワークインバージョンでは、学習時に得られた荷重を固定することによって、与えられた出力から入力を、

$$x(n+1) = x(n) - \varepsilon_e \frac{\partial E}{\partial x} \quad (3)$$

のように勾配法を用いて繰り返し修正する。ここで、 x および E 、 ε_e はそれぞれ入力、出力誤差、入力修正の微小係数を表す。これによって、学習した関係を逆に用いて出力から入力を推定することができる。

4 高次元ネットワークインバージョン

オリジナルのネットワークインバージョンは、実数型の多層ニューラルネットワークを用いて逆問題を解くものであるが、複素数などの高次元数ニューラルネットに拡張することで、高次元数間の逆問題を解く枠組みを考えることができる。

本研究では、複素ニューラルネットを用いて逆問題を解くための複素ネットワークインバージョンと、四元数ニューラルネットを用いて逆問題を解くための四元数ネットワークインバージョンを考える。

4.1 複素ネットワークインバージョン

ネットワークインバージョンを複素多層型ネットワークに拡張したものが複素ネットワークインバージョンである。

複素ネットワークインバージョンでは、通常のネットワークインバージョンの入力修正の原理を複素領域に拡張し、学習済みの複素多層型ニューラルネットワークを用いて、複素出力から複素入力を推定する。実際には、学習済みの複素多層型ニューラルネットワークに対して複素ランダム入力を与え、その複素出力と目標複素出力の誤差を逆伝搬し、複素入力の修正を繰り返すことによって、複素出力から複素入力を推定する。

複素ネットワークインバージョンによる逆推定時は、学習時に得られた関係を固定したまま、複素入力 $x = x_R + ix_I$ に仮の入力を与え、出力 $y = y_R + iy_I$ から複素出力誤差 $E = E_R + iE_I$ を計算する。そして、複素入力の実部と虚部をそれぞれ、

$$\begin{aligned} x_R(n+1) &= x_R(n) - \varepsilon_e \left(\frac{\partial E_R}{\partial x_R} + \frac{\partial E_I}{\partial x_R} \right) \\ x_I(n+1) &= x_I(n) - \varepsilon_e \left(\frac{\partial E_I}{\partial x_I} - \frac{\partial E_R}{\partial x_I} \right) \end{aligned} \quad (4)$$

の式に基づき更新する。 ε_e は入力更新の係数を意味する。この更新手順を繰り返すことによって、複素入力は出力の値を満たすように修正されていく。結果として、学習によって得られた複素荷重分布を使用して、複素出力から複素入力を繰り返し法によって逆推定することができる。

4.2 四元数ネットワークインバージョン

ネットワークインバージョンを四元数ニューラルネットワークに拡張する。ネットワークインバージョンの入力修正の原理を四元数領域に拡張し、学習済みの四元数ニューラルネットワークを用いて、四元数出力から四元数入力を推定する。実際には、学習済みのネットワークに対して四元数ランダム入力を与え、その四元数出力と目標四元数出力の誤差を逆伝搬し、四元数入力の修正を繰り返すことによって、四元数出力から四元数入力を推定する。

具体的には、学習時に得られた関係を固定したまま、四元数入力 $x = x_1 + ix_2 + jx_3 + kx_4$ に仮の入力を与え、出力 $y = y_1 + iy_2 + jy_3 + ky_4$ から複素出力誤差 $E = E_1 + iE_2 + jE_3 + kE_4$ を計算する。そして、四元数入力の各部をそれぞれ、

$$\begin{aligned} x_1(n+1) &= x_1(n) - \varepsilon_e \left(\frac{\partial E_1}{\partial x_1} - \frac{\partial E_2}{\partial x_1} - \frac{\partial E_3}{\partial x_1} - \frac{\partial E_4}{\partial x_1} \right) \\ x_2(n+1) &= x_2(n) - \varepsilon_e \left(\frac{\partial E_1}{\partial x_2} + \frac{\partial E_2}{\partial x_2} + \frac{\partial E_3}{\partial x_2} - \frac{\partial E_4}{\partial x_2} \right) \\ x_3(n+1) &= x_3(n) - \varepsilon_e \left(\frac{\partial E_1}{\partial x_3} - \frac{\partial E_2}{\partial x_3} + \frac{\partial E_3}{\partial x_3} + \frac{\partial E_4}{\partial x_3} \right) \\ x_4(n+1) &= x_4(n) - \varepsilon_e \left(\frac{\partial E_1}{\partial x_4} + \frac{\partial E_2}{\partial x_4} - \frac{\partial E_3}{\partial x_4} + \frac{\partial E_4}{\partial x_4} \right) \end{aligned} \quad (5)$$

の式に基づき更新する。 ε_e は入力更新の係数を意味する。この更新手順を繰り返すことで、結果として、学習によって得られた四元数荷重分布を使用して、四元数出力から四元数入力を繰り返し法によって逆推定できる。

5 シミュレーション

本研究では、高次元領域にわたる逆問題として、高次元写像の逆推定問題と、ロボットアームの逆運動学問題を取り上げる。以下で問題について説明し、さらに、複素および四元数ネットワークインバージョンによる逆推定シミュレーションを行い、それぞれの方法の動作を示す。

5.1 高次元逆写像問題

高次元領域にわたる簡単な逆問題の例として、複素写像および四元数写像の逆推定問題を取り上げる。

まず、複素ネットワークインバージョンによる複素写像の逆推定問題を考える。複素ニューラルネットワークによって複素平面上の点間に与えられた回転移動を学習し、さらに複素平面上に任意に与えられる点の逆写像を、学習時に得られた順方向関係を逆に用いて推定する。ここでは、原点周りの回転移動を考える。すなわち、原点周り θ の回転移動を行う入出力データ間の対応を学習し、与えられた出力データに対応する入力データの逆推定を試みる。学習データは $x_{nl} = x_{nR}$ 、 $(x_n = \{-0.7, -0.6, \dots, 0.7\})$ の直線上の点を入力とし、これらの点を $\theta = \pi/6$ だけ回転させた点を出力として学習する。推定時には、 $x_{nR}^2/a^2 + x_{nI}^2/b^2 = 1$ 、 $(x_{nR} = \cos\phi_n, x_{nI} = \sin\phi_n, \phi_n = \{0, \pi/6, \dots, 11\pi/6\}, a=0.9, b=0.3)$ の楕円上の点を出力として与え、対応する入力の推定を行う。

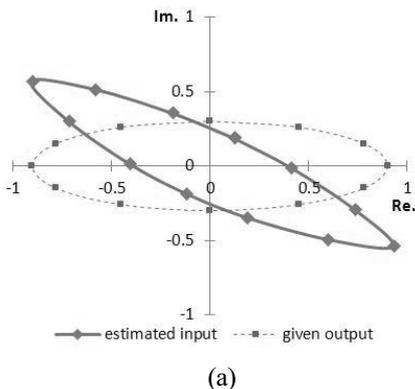
次に、四元数ネットワークインバージョンによる四元数写像の逆推定問題を考える。四元数ニューラルネットワークによって三次元空間内の点間に与えられた移動を学習し、さらに三次元空間内に任意に与えられる点の逆写像を、学習時に得られた順方向関係を逆に用いて推定する。ここでは、 x_2 軸周りの回転移動と x_3 軸周りの回転移動を考える。すなわち、 x_2 周りに θ_1 回転移動しさらに x_3 周りに θ_2 回転移動した入出力データ間の対応を学習し、与えられた出力データに対応する入力データの逆推定を試みる。

学習データは $x_{n2} = x_{n3} = x_{n4} = t_n$, ($t_n = \{-0.6, -0.5, \dots, 0.6\}$) の直線上の点を入力とし、これらの点を x_2 軸周りに $\theta_1 = \pi/6$, x_3 軸周りに $\theta_2 = -\pi/6$ 回転させた点を出力として学習する。推定時には、 $x_{n2}^2/a^2 + x_{n3}^2/b^2 + x_{n4}^2/c^2 = 1$, ($x_{n2} = \sin\phi_{n1}\cos\phi_{n2}$, $x_{n3} = \sin\phi_{n1}\sin\phi_{n2}$, $x_{n4} = \cos\phi_{n1}$, $\phi_{n1} = \phi_{n2} = \{0, \pi/6, \dots, 11\pi/6\}$, $a = b = 0.9$, $c = 0.5$) の楕円体上の点を出力として与え、対応する入力の推定を行う。

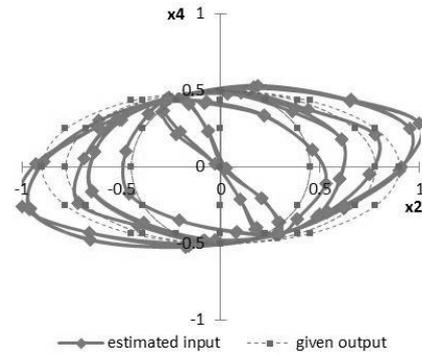
Table 1 に示すパラメータをもつ複素ネットワーク (CVNN) および四元数ネットワーク (QNN) を用いて、二次元平面および三次元空間内の写像の学習・逆推定を行った。Fig. 1 が逆推定結果である。複素ネットワークインバージョンの結果によると、出力として与えた楕円を $-\theta$ だけ回転移動した楕円が推定されていることがわかる。また、四元数ネットワークインバージョンの結果については、 x_2 - x_4 平面への投影図で示す。これによると、やや誤差があるものの、出力として与えた楕円体を x_3 軸周りに $-\theta_2$ だけ回転移動した楕円体が推定されていることがわかる。結果より、複素ネットワークインバージョンおよび四元数ネットワークインバージョンによって、学習した関係を逆に用いて入力を逆推定していることがわかる。

Table 1 Network parameters for inverse mapping problem.

Network	CVNN	QNN
Number of input neurons	1	1
Number of hidden neurons	5	10
Number of output neurons	1	1
Training rate ϵ_t	0.01	0.01
Input correcting rate ϵ_e	0.01	0.01
Max. number of training epoch	10000	10000
Max. number of estimating epoch	10000	10000
Training error to be attained	0.001	0.001
Estimation error to be attained	0.001	0.001



(a)



(b)

Fig. 1 Estimated results of inverse mapping, (a) by complex-valued network, and (c) by quaternion network.

5.2 ロボットアーム逆運動学問題

多関節ロボットアームの目標先端座標を実現する関節角を求める問題は逆問題であり、動作空間を二次元および三次元と考えると、複素領域および四元数領域にわたる逆問題が構成できる。本研究では Fig. 2 に示す二次元平面および三次元空間で動作する 2 自由度ロボットアームの先端座標からの関節角の推定を考える。

まず、複素ネットワークインバージョンによる二次元 2 自由度ロボットアーム関節角の逆推定を考える。複素ニューラルネットワークによって 2 つの関節角入力と先端座標出力間の対応を学習し、さらに任意に与えられる先端座標に対応する関節角を、学習時に得られた順方向関係を逆に用いて推定する。学習データは、 $\theta_1 = \{0, \pi/6, \pi/3, \pi/2\}$, $\theta_2 = \{0, \pi/6, \pi/3, \pi/2\}$ とし、その一部を推定対象データとした。

続いて、四元数ネットワークインバージョンによる三次元 2 自由度ロボットアーム関節角の推定を考える。四元数ニューラルネットワークによって 2 つの関節角入力と先端座標出力間の対応を学習し、さらに任意に与えられる先端座標に対応する関節角を、学習時に得られた順方向関係を逆に用いて推定する。学習データは、 $\theta_1 = \{0, \pi/6, \pi/3, \pi/2\}$, $\theta_2 = \{0, \pi/6, \pi/3, \pi/2, 2\pi/3, 5\pi/6\}$ とし、その一部を推定対象データとした。

Table 2 に示すパラメータをもつ複素ネットワークおよび四元数ネットワークを用いて、2 自由度ロボットアームの関節角と先端座標の関係の学習および逆推定を行った。複素ネットワークによる関節角の推定結果を Table 3 に、四元数ネットワークによる関節角の推定結果を Table 4 に示す。結果によると、複素ネットワークと四元数ネットワークともに、やや誤差が生じているものの、与えられた先端座標に対応する関節角を概ね推定していることがわかる。

これらの結果により、複素ネットワークおよび四元数ネットワーク上で、学習した関係を逆に用いて入力を逆推定していることがわかる。しかし、推定結果の精度はまだ不十分であり、この原因として、入出力間の多対一対応などの不良設定性などが考えられる。今後実際の使用に当たっては検討が必要である。

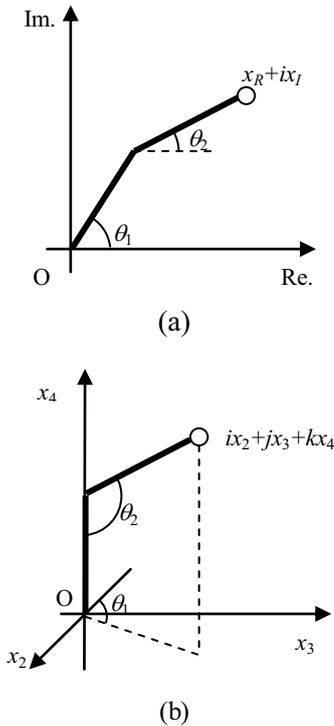


Fig. 2. 2DOF robot arm for inverse estimation of angles, (a) 2D model and (b) 3D model.

Table 2. Network parameters for inverse kinematics.

Network	CVNN	QNN
Number of input neurons	2	2
Number of hidden neurons	10	20
Number of output neurons	1	1
Training rate ε_t	0.01	0.001
Input correcting rate ε_e	0.01	0.001
Max. number of training epoch	100000	50000
Max. number of estimating epoch	20000	10000
Training error to be attained	0.0001	0.001
Estimation error to be attained	0.00001	0.00001

Table 3. Estimated angles by complex-valued network.

correct angles (deg.)		estimated angles (deg.)	
θ_1	θ_2	θ_1	θ_2
0	0	0.00	0.00
0	30	11.36	24.42
0	60	14.50	49.11
0	90	35.93	60.50
30	30	17.14	40.86
30	60	39.97	48.16
30	90	48.99	65.58
60	60	53.64	65.84
60	90	65.51	76.65
90	90	90.00	90.00

Table 4. Estimated angles by quaternion network.

correct angles (deg.)		estimated angles (deg.)	
θ_1	θ_2	θ_1	θ_2
30	30	32.77	29.43
60	30	49.01	26.11
90	30	72.77	26.97
30	60	35.36	55.85
60	60	57.52	55.75
90	60	83.73	52.34
30	90	35.18	80.97
60	90	55.79	84.79
90	90	82.11	84.86
30	120	33.59	110.24
60	120	58.61	123.21
90	120	90.00	135.95
30	150	32.08	134.88
60	150	62.37	137.72
90	150	89.82	140.81

6 おわりに

本研究では高次元空間における逆問題解法のために、複素数や四元数などの高次元数に拡張した高次元ネットワークインバージョン法を示した。また、これらの動作を示すために、高次元空間における逆写像問題およびロボットアーム逆運動学問題のシミュレーションを行った。今後の課題として、複素領域や四元数領域にわたる実際の逆問題への応用が挙げられる。また、逆問題の不良設定性緩和のための正則化法の検討⁸⁾や、推定精度の向上のための方策の検討などが考えられる。

参考文献

- 1) Nitta, T.: An extension of the backpropagation algorithm to complex numbers, *Neural Networks*, vol.10, no.8, 1392/1415 (1997)
- 2) Nitta, T.: A quaternary version of the back-propagation algorithm, *Proc. IEEE Int. Conf. on Neural Networks*, vol. 5, 2753/2756 (1995)
- 3) Grötsch, C. W.: *Inverse problems in the mathematical sciences*, Informatica International (1993)
- 4) Linden, A. and Kindermann, J.: Inversion of multilayer nets, *Proc. Int. Joint Conf. on Neural Networks*, 425/430 (1989)
- 5) Valova, I., Kameyama, K. and Kosugi, Y.: Image decomposition by answer-in-weights neural network, *IEICE Trans. on Information and Systems*, vol. E78-D, no. 9, 1221/1224 (1995)
- 6) Lu, B. and Ito, K.: Regularization of inverse kinematics for redundant manipulators using neural network inversions, *Proc. of IEEE Int. Conf. on Neural Networks*, 2726/2731 (1995)
- 7) 小川, 金田: 複素逆問題の解法のための複素ネットワークインバージョン, *信学論 D-II*, vol. J88-D-II, No.9, 1954/1962 (2005)
- 8) Fukami, S., Ogawa, T. and Kanada, H.: Regularization for complex-valued network inversion, *Proc. SICE Annual Conf.*, 1237/1242 (2008)

Fourier-Mellin Transform with Quaternions

* Eckhard Hitzer (University of Fukui)

Abstract— In this contribution we generalize the classical complex Fourier Mellin transform [3], which transforms functions f representing, e.g., a gray level image defined over a compact set of \mathbb{R}^2 . The quaternionic Fourier Mellin transform (QFMT) applies to functions (e.g. color a image) $f : \mathbb{R}^2 \rightarrow \mathbb{H}$, for which $|f|$ is summable over $\mathbb{R}_+^* \times \mathbb{S}^1$ under the measure $d\theta \frac{dx}{r}$. \mathbb{R}_+^* is the multiplicative group of positive and non-zero real numbers. We investigate application relevant properties of the QFMT.

Keywords: Fourier-Mellin transform, quaternion algebra, invariant shape recognition, color image processing

1 Introduction

The Fourier-Mellin transform is an excellent tool in order to achieve translation, rotation and scale invariant shape recognition. It has recently been extended to the algebra of quaternions \mathbb{H} in the form of the quaternionic Fourier-Mellin Transform [11], which in principle also allows a non-marginal processing of color images. A further extension to Clifford algebras $Cl(p, q)$, $p + q = 2$ has also recently been completed [12]. In this contribution we discuss the quaternionic Fourier-Mellin Transform following [11].

This contribution has the following structure. We begin with a brief introduction to the algebra of quaternions \mathbb{H} , and explain the two-dimensional orthogonal planes split of quaternions together with its geometric interpretation. Then the quaternionic Fourier Mellin transform (QFMT) is introduced with brief anecdotal comments on Robert Hjalmar Mellin's biography, basics of quaternion-valued functions, the QFMT definition, and an overview of its main properties: inverse QFMT transform, linearity, quasi-complex forms of the QFMT, modulus and magnitude identities, modulation, symmetry properties, derivatives and power scaling, as well as Plancherel and Parseval theorems. Finally the symmetry and kernel structures of the following four two-dimensional transforms are compared: complex FMT, complex Fourier transform, quaternionic Fourier transform, and the QFMT. Color versions of the figures (showing maxima and minima in red and blue colors, respectively), can be found in the electronic copy of this contribution on [5].

2 Quaternions

Gauss¹, Rodrigues and Hamilton introduced the 4D quaternion algebra \mathbb{H} over \mathbb{R} with three imaginary

¹By reading this paper you agree to the terms of the *Creative Peace License* [10].

units:

$$\begin{aligned} ij &= -ji = k, \quad jk = -kj = i, \quad ki = -ik = j, \\ i^2 &= j^2 = k^2 = ijk = -1. \end{aligned} \quad (1)$$

Every quaternion

$$q = q_r + q_i i + q_j j + q_k k \in \mathbb{H}, \quad q_r, q_i, q_j, q_k \in \mathbb{R} \quad (2)$$

has *quaternion conjugate* (reversion in $Cl_{3,0}^+$)

$$\tilde{q} = q_r - q_i i - q_j j - q_k k, \quad (3)$$

This leads to *norm* of $q \in \mathbb{H}$, and an inverse of every non-zero $q \in \mathbb{H}$

$$\begin{aligned} |q| &= \sqrt{q\tilde{q}} = \sqrt{q_r^2 + q_i^2 + q_j^2 + q_k^2}, \quad |pq| = |p||q|, \\ q^{-1} &= \frac{\tilde{q}}{|q|^2} = \frac{\tilde{q}}{q\tilde{q}}. \end{aligned} \quad (4)$$

The scalar part of quaternions is symmetric

$$Sc(q) = q_r = \frac{1}{2}(q + \tilde{q}), \quad Sc(pq) = Sc(qp). \quad (5)$$

The *inner product* of quaternions defines *orthogonality*

$$Sc(p\tilde{q}) = p_r q_r + p_i q_i + p_j q_j + p_k q_k \in \mathbb{R}. \quad (6)$$

2.1 The (2D) orthogonal planes split (OPS) of quaternions

We consider an arbitrary pair of pure quaternions f, g , $f^2 = g^2 = -1$. The orthogonal 2D planes split (OPS) is then defined with respect to a pair of pure unit quaternions f, g as

$$q_{\pm} = \frac{1}{2}(q \pm f g q). \quad (7)$$

We thus observe, that

$$f g q = q_+ - q_-, \quad (8)$$

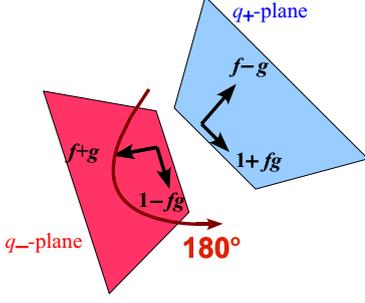


Figure 1: The map $f()g$ rotates the q_- plane by 180° around the q_+ axis plane. (Basis for $g \neq \pm f$.)

i.e. under the map $f()g$ the q_+ part is invariant, but the q_- part changes sign. Both parts are two-dimensional, and span two completely orthogonal planes. For $f \neq \pm g$ the q_+ plane is spanned by the orthogonal quaternions $\{f - g, 1 + fg\}$, and the q_- plane is spanned by $\{f + g, 1 - fg\}$. Vice versa, in general any two 2D orthogonal planes in \mathbb{H} determine a corresponding pair f, g .

Lemma 2.1 (Orthogonality of two OPS planes). *Given any two quaternions q, p and applying the OPS with respect to two linearly independent pure unit quaternions f, g we get zero for the scalar part of the mixed products*

$$\begin{aligned} Sc(p_+ \tilde{q}_-) &= 0, & Sc(p_- \tilde{q}_+) &= 0 \\ \implies |q|^2 &= |q_+|^2 + |q_-|^2. \end{aligned} \quad (9)$$

2.2 Geometric interpretation of map $f()g$

The map $f()g$ rotates the q_- plane by 180° around the q_+ axis plane, see Fig. 1. This interpretation of the map $f()g$ is in perfect agreement with Coxeter's notion of half-turn in [2].

We obtain the following important identities:

$$\begin{aligned} e^{\alpha f} q_{\pm} e^{\beta g} &= q_{\pm} e^{(\beta \mp \alpha)g} = e^{(\alpha \mp \beta)f} q_{\pm}, \\ q &\in \mathbb{H}, \quad \alpha, \beta \in \mathbb{R}. \end{aligned} \quad (10)$$

For $g \neq \pm f$ the set $\{f - g, 1 + fg, f + g, 1 - fg\}$ forms an orthogonal basis of \mathbb{H} interpreted as \mathbb{R}^4 . We can therefore use the following representation for every $q \in \mathbb{H}$ by means of four real coefficients $q_1, q_2, q_3, q_4 \in \mathbb{R}$

$$\begin{aligned} q &= q_1(1 + fg) + q_2(f - g) + q_3(1 - fg) + q_4(f + g), \\ & \quad (11) \\ q_1 &= Sc(q(1 + fg)^{-1}), \quad q_2 = Sc(q(f - g)^{-1}), \\ q_3 &= Sc(q(1 - fg)^{-1}), \quad q_4 = Sc(q(f + g)^{-1}). \end{aligned}$$



Figure 2: Simplex/perplex split of [15] with gray line $f = g = (\mathbf{i} + \mathbf{j} + \mathbf{k})/\sqrt{3}$. *Top left*: Original. *Top right*: q_- -part (luminance). *Bottom right*: q_+ -part ($q_i \leftrightarrow q_j$) (chrominance). *Bottom left*: Sum.

In the case of $f = \mathbf{i}, g = \mathbf{j}$ we obtain the coefficients

$$\begin{aligned} q_1 &= \frac{1}{2}(q_r + q_k), & q_2 &= \frac{1}{2}(q_i - q_j), \\ q_3 &= \frac{1}{2}(q_r - q_k), & q_4 &= \frac{1}{2}(q_i + q_j). \end{aligned} \quad (12)$$

The OPS with respect to a single pure unit quaternion, e.g., $f = g = \mathbf{i}$ gives

$$\begin{aligned} q_{\pm} &= \frac{1}{2}(q_{\pm} \pm \mathbf{i}q_{\pm}), & q_+ &= q_j \mathbf{j} + q_k \mathbf{k} = (q_j + q_k \mathbf{i}) \mathbf{j}, \\ q_- &= q_r + q_i \mathbf{i}, \end{aligned} \quad (13)$$

where the q_+ plane is 2D and manifestly orthogonal to the 2D q_- plane. The above corresponds to the simplex/perplex split of [4], see an application in Fig. 2 from [15].

3 The Quaternionic Fourier Mellin transformations (QFMT)

3.1 Robert Hjalmar Mellin (1854–1933)

Robert Hjalmar Mellin (1854–1933) [13], Fig. 3, was a Finnish mathematician, a student of G. Mittag-Leffler and K. Weierstrass. He became the director of the Polytechnic Institute in Helsinki, and in 1908 first professor of mathematics at Technical University of Finland. He was a fervent fennoman with fiery temperament, and co-founder of the Finnish Academy of Sciences. He became known for the *Mellin transform* with major applications to the evaluation of integrals, see [14], which lists 1624 references. During his last



Figure 3: Robert Hjalmar Mellin (1854–1933). Image: Wikipedia.

10 years he tried to refute Einstein's theory of relativity as logically untenable.

Definition 3.1 (Classical Fourier Mellin transform (FMT)).

$$\begin{aligned} \forall(v, k) \in \mathbb{R} \times \mathbb{Z}, \quad \mathcal{M}\{h\}(v, k) &= \\ &= \frac{1}{2\pi} \int_0^\infty \int_0^{2\pi} h(r, \theta) r^{-iv} e^{-ik\theta} d\theta \frac{dr}{r}, \end{aligned} \quad (14)$$

where $h : \mathbb{R}^2 \rightarrow \mathbb{R}$ denotes a function representing, e.g., a gray level image defined over a compact set of \mathbb{R}^2 .

Well known applications are to shape recognition (independent of rotation and scale), image registration, and similarity.

3.2 Inner product, symmetric part, norm of quaternion-valued functions

The quaternion \mathbb{H} -valued inner product for quaternion-valued functions $h, m : \mathbb{R}^2 \rightarrow \mathbb{H}$ is given by

$$(h, m) = \int_{\mathbb{R}^2} h(\mathbf{x}) \tilde{m}(\mathbf{x}) d^2 \mathbf{x}, \quad \text{with} \quad d^2 \mathbf{x} = dx dy, \quad (15)$$

It has a symmetric real scalar part

$$\langle h, m \rangle = \frac{1}{2} [(h, m) + (m, h)] = \int_{\mathbb{R}^2} \text{Sc}(h(\mathbf{x}) \tilde{m}(\mathbf{x})) d^2 \mathbf{x}, \quad (16)$$

which allows to define a $L^2(\mathbb{R}^2; \mathbb{H})$ -norm

$$\begin{aligned} \|h\|^2 &= (h, h) = \langle h, h \rangle = \int_{\mathbb{R}^2} |h(\mathbf{x})|^2 d^2 \mathbf{x} \\ \implies \|h\|^2 &= \|h_+\|^2 + \|h_-\|^2. \end{aligned} \quad (17)$$

A quaternion module can be defined as $L^2(\mathbb{R}^2; \mathbb{H})$ by

$$L^2(\mathbb{R}^2; \mathbb{H}) = \{h|h : \mathbb{R}^2 \rightarrow \mathbb{H}, \|h\| < \infty\}. \quad (18)$$

3.3 The quaternionic Fourier Mellin transformation and its properties

We now define the generalization of the FMT to quaternionic signals.

Definition 3.2 (Quaternionic Fourier Mellin transform (QFMT)). Let $f, g \in \mathbb{H} : f^2 = g^2 = -1$ be any pair of pure unit quaternions. The quaternionic Fourier Mellin transform (QFMT) is given by

$$\begin{aligned} \forall(v, k) \in \mathbb{R} \times \mathbb{Z}, \quad \hat{h}(v, k) &= \mathcal{M}\{h\}(v, k) = \\ &= \frac{1}{2\pi} \int_0^\infty \int_0^{2\pi} r^{-fv} h(r, \theta) e^{-gk\theta} d\theta \frac{dr}{r}, \end{aligned} \quad (19)$$

where $h : \mathbb{R}^2 \rightarrow \mathbb{H}$ denotes a function from \mathbb{R}^2 into the algebra of quaternions \mathbb{H} , such that $|h|$ is summable over $\mathbb{R}_+^* \times \mathbb{S}^1$ under the measure $d\theta \frac{dr}{r}$. \mathbb{R}_+^* is the multiplicative group of positive and non-zero real numbers.

For $f = i, g = j$ we have the special case

$$\begin{aligned} \forall(v, k) \in \mathbb{Z} \times \mathbb{R}, \quad \hat{h}(v, k) &= \mathcal{M}\{h\}(v, k) = \\ &= \frac{1}{2\pi} \int_0^\infty \int_0^{2\pi} r^{-iv} h(r, \theta) e^{-jk\theta} d\theta \frac{dr}{r}, \end{aligned} \quad (20)$$

Note, that the \pm split and the QFMT commute:

$$\mathcal{M}\{h_\pm\} = \mathcal{M}\{h\}_\pm.$$

Theorem 3.3 (Inverse QFMT). The QFMT can be inverted by

$$\begin{aligned} h(r, \theta) &= \mathcal{M}^{-1}\{\hat{h}\}(r, \theta) = \\ &= \frac{1}{2\pi} \int_{-\infty}^\infty \sum_{k \in \mathbb{Z}} r^{fv} \hat{h}(v, k) e^{gk\theta} dv. \end{aligned} \quad (21)$$

The proof uses

$$\begin{aligned} \frac{1}{2\pi} \sum_{k \in \mathbb{Z}} e^{gk(\theta - \theta')} &= \delta(\theta - \theta'), \quad r^{fv} = e^{fv \ln r}, \\ \frac{1}{2\pi} \int_0^{2\pi} e^{fv(\ln(r) - s)} dv &= \delta(\ln(r) - s). \end{aligned} \quad (22)$$

We now investigate the basic properties of the QFMT. First, left linearity: For $\alpha, \beta \in \{q \mid q = q_r + q_f f, q_r, q_f \in \mathbb{R}\}$,

$$\begin{aligned} m(r, \theta) &= \alpha h_1(r, \theta) + \beta h_2(r, \theta) \\ \implies \hat{m}(v, k) &= \alpha \hat{h}_1(v, k) + \beta \hat{h}_2(v, k). \end{aligned} \quad (23)$$

Second, right linearity: For $\alpha', \beta' \in \{q \mid q = q_r + q_g g, q_r, q_g \in \mathbb{R}\}$,

$$\begin{aligned} m(r, \theta) &= h_1(r, \theta) \alpha' + h_2(r, \theta) \beta' \\ \implies \hat{m}(v, k) &= \hat{h}_1(v, k) \alpha' + \hat{h}_2(v, k) \beta'. \end{aligned} \quad (24)$$

The linearity of the QFMT leads to

$$\begin{aligned}\mathcal{M}\{h\}(v, k) &= \mathcal{M}\{h_- + h_+\}(v, k) \\ &= \mathcal{M}\{h_-\}(v, k) + \mathcal{M}\{h_+\}(v, k),\end{aligned}\quad (25)$$

which gives rise to the following theorem.

Theorem 3.4 (Quasi-complex FMT like forms for QFMT of h_{\pm}). *The QFMT of h_{\pm} parts of $h \in L^2(\mathbb{R}^2, \mathbb{H})$ have simple quasi-complex forms*

$$\begin{aligned}\mathcal{M}\{h_{\pm}\} &= \frac{1}{2\pi} \int_0^{\infty} \int_0^{2\pi} h_{\pm} r^{\pm gv} e^{-gk\theta} d\theta \frac{dr}{r} \\ &= \frac{1}{2\pi} \int_0^{\infty} \int_0^{2\pi} r^{-fv} e^{\pm fk\theta} h_{\pm} d\theta \frac{dr}{r}.\end{aligned}\quad (26)$$

Theorem 3.4 allows to use discrete and fast software to compute the QFMT based on a pair of complex FMT transformations.

For the two split parts of the QFMT, we have the following lemma.

Lemma 3.5 (Modulus identities). *Due to $|q|^2 = |q_-|^2 + |q_+|^2$ we get for $f : \mathbb{R}^2 \rightarrow \mathbb{H}$ the following identities*

$$\begin{aligned}|h(r, \theta)|^2 &= |h_-(r, \theta)|^2 + |h_+(r, \theta)|^2, \\ |\mathcal{M}\{h\}(v, k)|^2 &= |\mathcal{M}\{h_-\}(v, k)|^2 + |\mathcal{M}\{h_+\}(v, k)|^2.\end{aligned}\quad (27)$$

Further properties are *scaling* and *rotation*: For $m(r, \theta) = h(ar, \theta + \phi)$, $a > 0$, $0 \leq \phi \leq 2\pi$,

$$\widehat{m}(v, k) = a^{fv} \widehat{h}(v, k) e^{gk\phi}.\quad (28)$$

Moreover, we have the following magnitude identity:

$$|\widehat{m}(v, k)| = |\widehat{h}(v, k)|,\quad (29)$$

i.e. the magnitude of the QFMT of a scaled and rotated quaternion signal $m(r, \theta) = h(ar, \theta + \phi)$ is identical to the magnitude of the QFMT of h . Equation (29) forms the basis for applications to rotation and scale invariant shape recognition and image registration. This may now be extended to color images, since quaternions can encode colors RGB in their $\mathbf{i}, \mathbf{j}, \mathbf{k}$ components.

The reflection at the unit circle ($r \rightarrow \frac{1}{r}$) leads to

$$m(r, \theta) = h\left(\frac{1}{r}, \theta\right) \implies \widehat{m}(v, k) = \widehat{h}(-v, k).\quad (30)$$

Reversing the sense of sense of rotation ($\theta \rightarrow -\theta$) yields

$$m(r, \theta) = h(r, -\theta) \implies \widehat{m}(v, k) = \widehat{h}(v, -k).\quad (31)$$

Regarding radial and rotary modulation we assume

$$m(r, \theta) = r^{fv_0} h(r, \theta) e^{gk_0\theta}, \quad v_0 \in \mathbb{R}, k_0 \in \mathbb{Z}.\quad (32)$$

Then we get

$$\widehat{m}(v, k) = \widehat{h}(v - v_0, k - k_0).\quad (33)$$

3.4 QFMT derivatives and power scaling

We note for the logarithmic derivative that $r \frac{d}{dr} = r \partial_r$,

$$\mathcal{M}\{(r \partial_r)^n h\}(v, k) = (fv)^n \widehat{h}(v, k), \quad n \in \mathbb{N}.\quad (34)$$

Applying the angular derivative with respect to θ we obtain

$$\mathcal{M}\{\partial_{\theta}^n h\}(v, k) = \widehat{h}(v, k) (gk)^n, \quad n \in \mathbb{N}.\quad (35)$$

Finally, power scaling with $\ln r$ and θ leads to

$$\mathcal{M}\{(\ln r)^m \theta^n h\}(v, k) = f^m \partial_v^m \partial_k^n \widehat{h}(v, k) g^n, \quad m, n \in \mathbb{N}.\quad (36)$$

3.5 QFMT Plancherel and Parseval theorems

For the QFMT we have the following two theorems.

Theorem 3.6 (QFMT Plancherel theorem). *The scalar part of the inner product of two functions $h, m : \mathbb{R}^2 \rightarrow \mathbb{H}$ is*

$$\langle h, m \rangle = \langle \widehat{h}, \widehat{m} \rangle.\quad (37)$$

Theorem 3.7 (QFMT Parseval theorem). *Let $h : \mathbb{R}^2 \rightarrow \mathbb{H}$. Then*

$$\|h\| = \|\widehat{h}\|, \quad \|h\|^2 = \|\widehat{h}\|^2 = \|\widehat{h}_+\|^2 + \|\widehat{h}_-\|^2.\quad (38)$$

4 Symmetry and kernel structures of 2D FMT, FT, QFT, QFMT

The QFMT of real signals analyzes symmetry. The following notation will be used². The function h_{ee} is *even* with respect to (w.r.t.) $r \rightarrow \frac{1}{r} \iff \ln r \rightarrow -\ln r$, i.e. w.r.t. the reflection at the unit circle, and *even* w.r.t. $\theta \rightarrow -\theta$, i.e. w.r.t. reversing the sense of rotation (reflection at the $\theta = 0$ line of polar coordinates in the (r, θ) -plane). Similarly we denote by h_{eo} even-odd symmetry, by h_{oe} odd-even symmetry, and by h_{oo} odd-odd symmetry.

Let h be a real valued function $\mathbb{R}^2 \rightarrow \mathbb{R}$. The QFMT of h results in

$$\widehat{h}(v, k) = \underbrace{\widehat{h}_{ee}(v, k)}_{\text{real part}} + \underbrace{\widehat{h}_{eo}(v, k)}_{f\text{-part}} + \underbrace{\widehat{h}_{oe}(v, k)}_{g\text{-part}} + \underbrace{\widehat{h}_{oo}(v, k)}_{fg\text{-part}}.\quad (39)$$

The QFMT of a real signal therefore automatically separates components with different combinations of symmetry w.r.t. reflection at the unit circle and reversal of the sense of rotation. The four components of the QFMT kernel differ by radial and angular phase shifts, see Fig. 4. The symmetries of $r \rightarrow 1/r$

²In this section we assume $g \neq \pm f$, but a similar study is possible for $g = \pm f$.

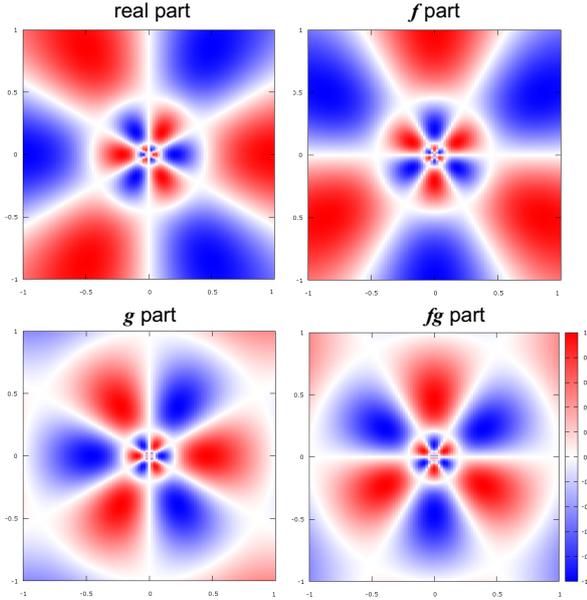


Figure 4: Four components of the QFMT kernel ($v = 2, k = 3$).

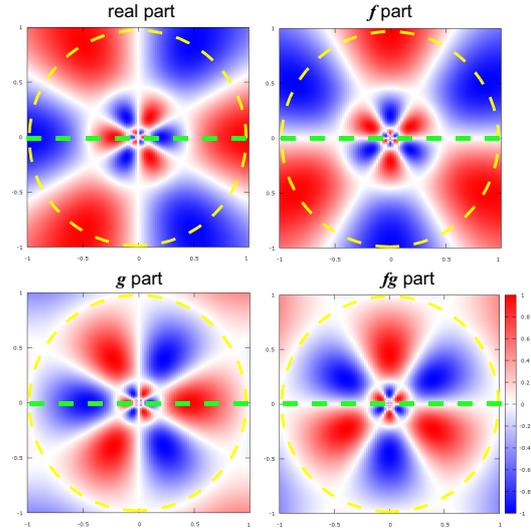


Figure 5: Symmetries of four components of the QFMT kernel ($v = 2, k = 3$).

(reflection at *yellow unit circle*), and $\theta \rightarrow -\theta$ (reflection at *green line*) can be clearly seen in Fig. 5.

Figure 6 shows real the component of the QFMT kernel for various values of v, k , demonstrating various angular and radial resolutions. Figure 7 shows the real component of the QFMT kernel for $v = k = 4$ at three different scales. Similar patterns appear at all scales. Figure 8 shows the kernels of complex 2D Fourier transform (FT) $e^{-i(ux+vy)}$, $i \in \mathbb{C}$, and the QFT $e^{-iux}e^{-jvy}$, $i, j \in \mathbb{H}$, taken from [1], which treats applications to 2D gray scale images. Corresponding applications to color images can be found in [15]. The 2D FT is intrinsically 1D, the QFT is intrinsically 2D, which makes it superior in disparity estimation and 2D texture segmentation, etc.

Figure 9 compares the kernels (real parts) of 2D complex FMT and the QFMT. Obviously the 2D QFMT can analyze genuine 2D textures better than the 2D complex FMT. Finally, Fig. 10 compares the kernels of the QFT (top four) and the QFMT (bottom four). The scale invariant feature of the QFMT is obvious. Compared with the left side of Fig. 9, the QFMT is obviously the linear superposition of two quasi-complex FMTs with opposite winding sense, as shown in Theorem 3.4.

5 Conclusion

The algebra of quaternions allows to construct a variety of quaternionic Fourier-Mellin transformations (QFMT), dependent on the choice of $f, g \in \mathbb{H}$,

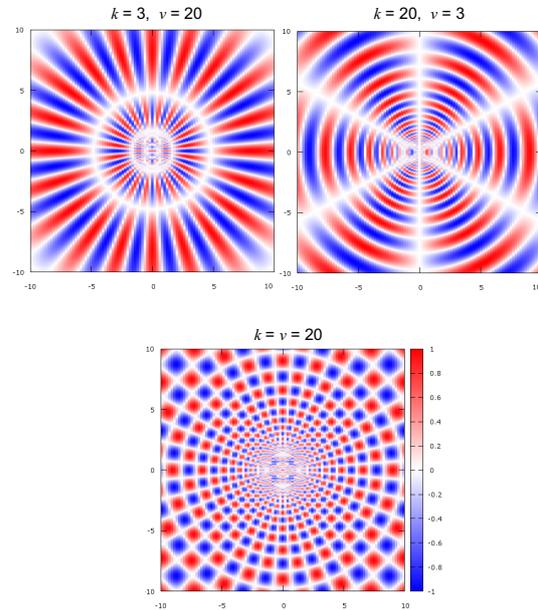


Figure 6: *Top left*: High angular resolution. *Top right*: High radial resolution. *Bottom*: High radial and angular resolution.

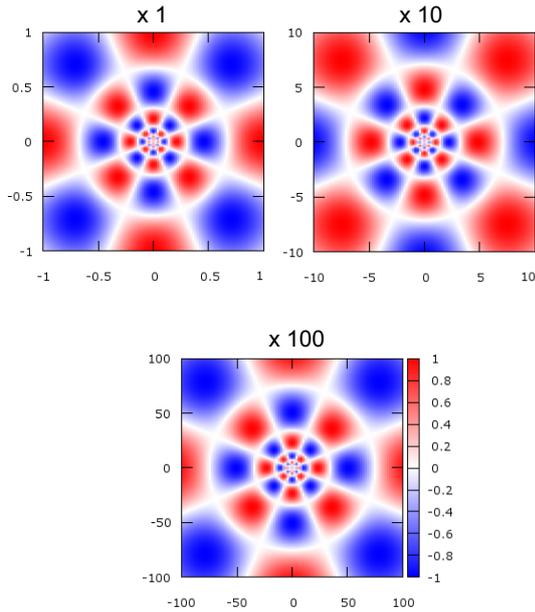


Figure 7: Illustration of QFMT scaling.

$f^2 = g^2 = -1$. According to Figure 9, the QFMT is to be expected to lead to a better analysis of scale invariant patterns than the FMT, similar to the better performance in pattern analysis of the QFT compared to the complex 2D FT [1]. Further variations of the QFMT would be to place both kernel factors initially at the left or right of the signal $h(r, \theta)$. The whole QFMT concept can easily be generalized to Clifford algebras $Cl(p, q)$, based on the general theory of square roots of -1 in $Cl(p, q)$, compare, e.g., [12] for the case of $p + q = 2$.

The modulus of the transform is scale and rotation invariant. Preceded by 2D FT or by QFT, this allows translation, scale and rotation invariant object description. A diverse range of applications can therefore be imagined: Color object shape recognition, color image registration, application to evaluation of hypercomplex integrals, etc.

Future research may be on extensions to higher dimensional Clifford algebras $Cl(p, q)$, to windowed and wavelet transforms, discretization, and numerical implementations.

Acknowledgment

I want to acknowledge God [16]:

Do you not know that in a race all the runners run,
but only one gets the prize?
Run in such a way as to get the prize.
Everyone who competes in the games
goes into strict training.

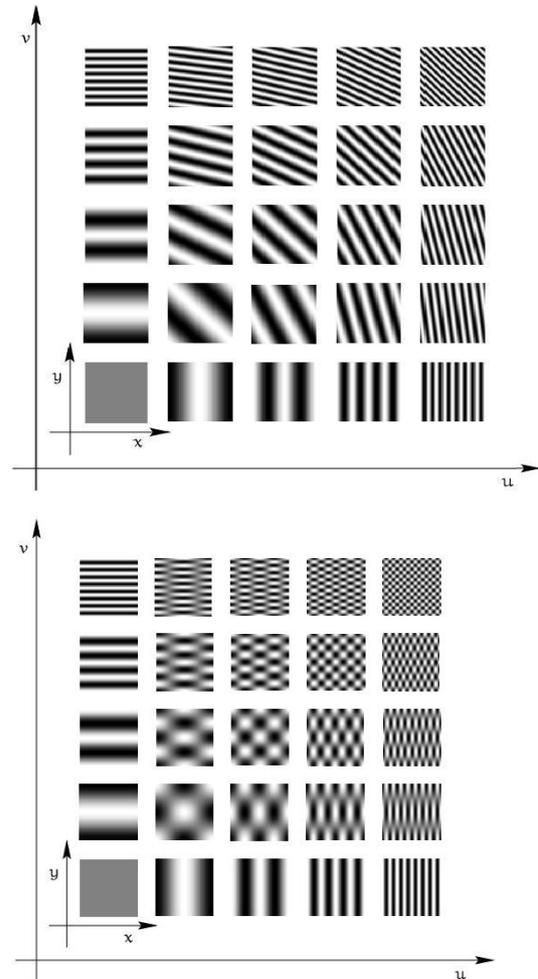


Figure 8: Top: 2D FT is intrinsically 1D. Bottom: QFT is intrinsically 2D. Source: [1].

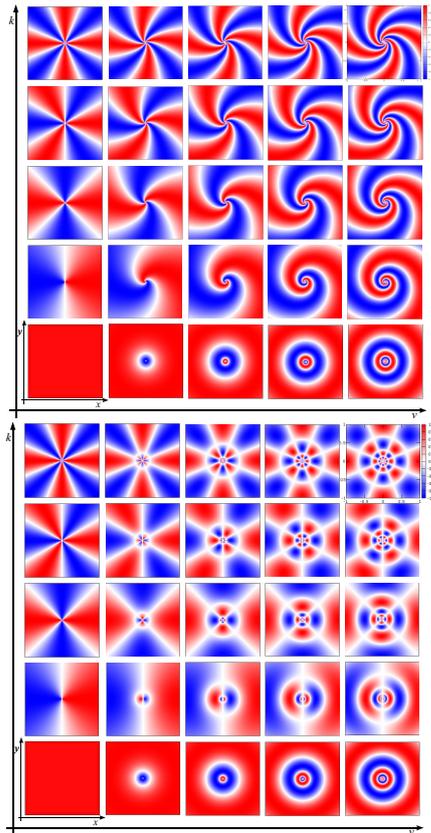


Figure 9: *Top four*: Kernel of FMT. *Bottom four*: Kernel of QFMT. $k, v \in \{0, 1, 2, 3, 4\}$.

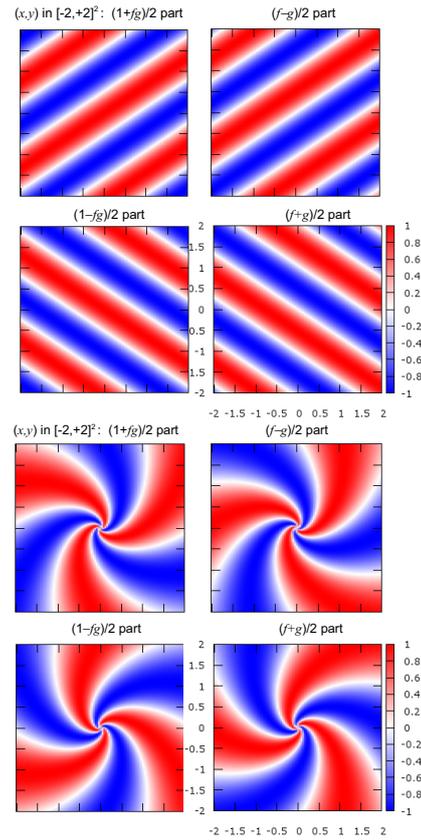


Figure 10: *Top four*: QFT kernel, *bottom four*: QFMT kernel. *First row*: q_+ parts: $1 + fg$ and $f - g$ components. *Second row*: q_- parts: $1 - fg$ and $f + g$ components.

They do it to get a crown that will not last,
but we do it to get a crown that will last
forever.

I want to thank my dear family, as well as T. Nitta
and Y. Kuroe.

References

- [1] T. Bülow, *Hypercomplex Spectral Signal Representations for the Processing and Analysis of Images*, Ph. D. thesis, University of Kiel (1999).
- [2] H. S. M. Coxeter, *Quaternions and Reflections*, The American Mathematical Monthly, **53**(3) (Mar. 1946), 136/146 (1946).
- [3] S. Dorrode and F. Ghorbel, *Robust and efficient Fourier-Mellin transform approximations for gray-level image reconstruction and complete invariant description*, Computer Vision and Image Understanding, **83**(1), 57/78 (2001). DOI 10.1006/cviu.2001.0922.
- [4] T. A. Ell, *Quaternionic-Fourier Transform for Analysis of Two-dimensional Linear Time-Invariant Partial Differential Systems*. in Proceedings of the 32nd IEEE Conference on Decision and Control, December 15-17, **2**, 1830/1841 (1993).
- [5] E. Hitzer, electronic versions of publications available for download, <http://sinai.mech.fukui-u.ac.jp/gcj/pubs.html>
- [6] E. Hitzer, *Quaternion Fourier Transform on Quaternion Fields and Generalizations*, Advances in Applied Clifford Algebras, **17**(3), 497/517 (2007).
- [7] E. Hitzer, *Directional Uncertainty Principle for Quaternion Fourier Transforms*, Advances in Applied Clifford Algebras, **20**(2), 271/284 (2010), online since 08 July 2009.
- [8] E. Hitzer, S. Sangwine, *The orthogonal planes split of quaternions*, Proceedings of ICCA9, Weimar 2011.
- [9] E. Hitzer, *OPS-QFTs: A new type of quaternion Fourier transforms based on the orthogonal planes split with one or two general pure quaternions*, AIP Conf. Proc. Volume 1389, 280/283 (2011), NUMERICAL ANALYSIS AND APPLIED MATHEMATICS ICNAAM 2011: International Conference on Numerical Analysis and Applied Mathematics; Doi: 10.1063/1.3636721
- [10] E. Hitzer, *Creative Peace License*, <http://gaupdate.wordpress.com/2011/12/14/the-creative-peace-license-14-dec-2011/>
- [11] E. Hitzer, *Quaternionic Fourier-Mellin Transform*, accepted for T. Sugawa (ed.), Proceedings of the The 19th International Conference on Finite or Infinite Dimensional Complex Analysis and Applications (ICFIDCAA), 11–15 December 2011, Hiroshima, Japan, (2012).
- [12] E. Hitzer, *Fourier-Mellin transform, Clifford algebra, square roots of -1* , submitted to: S. Georgiev, J. Morais, W. Sprössig (session eds.), Proceedings of ICNPAA 2012, 10–15 July 2012, TU Vienna, Vienna, AIP Conference Proceedings, (2012).
- [13] Robert Hjalmar Mellin (1854–1933). Biography. Homepage of St. Andrews University, Scotland.
- [14] A.P. Prodnikov, Y.A. Brychkov, O.I. Marichev, *Evaluation of integrals and the Mellin transform*. Transl. from Itogi Nauki i Techniki, Seriya Matematicheskii Analiz, Vol. 27, 3/146 (1989).
- [15] T. Ell, S. Sangwine, *Hypercomplex Fourier Transforms of Color Images* IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 16, NO. 1, JAN. 2007, 22/35 (2007).
- [16] The Bible, New International Version (NIV), Paul's first letter to the Corinthians, chapter 1, verses 24 and 25, <http://www.biblegateway.com/>

複素信号処理と多様体上の最適化

○西森 康則 (産業技術総合研究所)

Complex Signal Processing and Optimization on Manifolds

*Yasunori Nishimori

(National Institute of Advanced Industrial Science and Technology (AIST))

Abstract— Complex independent subspace analysis task can be cast as an optimization problem on complex flag manifolds, and we adapt the Riemannian optimization method such as Riemannian gradient descent and conjugate gradient method to this class of complex manifolds. Riemannian optimization method was originally proposed for real manifolds, and yet update rules for optimization on complex homogeneous spaces are directly obtained from the real counterparts by a correspondence between real and complex matrices.

Key Words: Complex independent subspace analysis, Natural gradient, Conjugate gradient method, Complex flag manifolds

1 はじめに

複素ニューラルネットの研究が90年代から進められてきたのに比べて遅れた感はあるが、近年、複素数値信号の統計的信号処理の研究が活発に行われている¹¹⁾¹⁷⁾。一方、拘束条件付きの最適化問題を、多様体上の拘束無しの最適化問題と読みかえて解く、いわゆる多様体上の最適化の研究が、Absil-Mahony-Sepulchreによる教科書²⁾の出版もあり、近年、機械学習や信号処理、コンピュータビジョン、科学計算など多くの分野で認知され、応用されてきている。しかしながら、現在のところ、多様体上の最適化の研究は、実多様体上の実関数の最適化が主であって、複素信号処理で必要となる、複素多様体上の実関数の最適化の研究は、まだそれほど進められていない。

本稿では、複素数値の統計的信号処理の問題として複素独立部分空間分析をとりあげる。そして、その解が複素旗多様体上の実関数の最適化によって得られること、複素旗多様体のようなユニタリー群の等質空間では、リーマン幾何的最適化法(最急降下法、共役勾配法)の更新式は、実旗多様体の場合の更新式から一定の手続きで導出できることを示す。(文献(15)で発表されたものに基づく。)

2 複素独立部分空間分析

複素独立部分空間分析は、以下のように実の独立部分空間分析⁹⁾を複素に拡張することにより定義される。まず源信号 $s \in \mathbb{C}^N$ が非退化な行列 $A \in \mathbb{C}^{N \times N}$ によって線形に混合された信号が観測される。

$$x = As \quad (1)$$

ここに、 s は d_j 個 ($j = 1, \dots, r+1$) ごとのクラスターに分かれ、

$$s = (s_1, \dots, s_{d_1}, s_{d_1+1}, \dots, s_n, s_{n+1}, \dots, s_N)^T \quad (2)$$

$$\sum_{j=1}^{r+1} d_j = N \quad (3)$$

異なるクラスターに属する任意の2つの信号は統計的に独立な信号で、同じクラスターに属する2つの信号

は統計的に従属な信号であるという確率的生成モデルを仮定する。さらに、 $d_{r+1} = N - n$ で、 $(r+1)$ 番目のクラスターに含まれる信号がガウシアンノイズであることを仮定すると、複素独立部分空間分析(cISA)のタスクは、観測された混合信号から、 $(N - n)$ 個のガウシアンノイズを除いた n 個の源信号 s_1, \dots, s_n を、信号のラベルや、パワーの不定性を除いて復元するという問題になる。最もよく扱われるのが $N = n$ の場合である。

複素独立部分空間分析を解くために、実独立部分空間分析の場合と同様に、まず前処理として混合信号の平均値を0に揃えて白色化しておく。

$$z = Bx, \quad E_z [zz^H] = I_N \quad (4)$$

すると複素独立部分空間分析を解くことは、

$$y = W^H z = W^H B A s \quad (5)$$

で表される出力ベクトルが、最初に仮定した確率的生成モデルの条件を満たすようなユニタリー行列 W を求めることに帰着する。

$$W = [W_1, W_2, \dots, W_r] \in \mathbb{C}^{N \times n} \quad (6)$$

$$W^H W = I_n, W_j \in \mathbb{C}^{N \times d_j} \quad (7)$$

従って、複素独立部分空間分析を解くことは、複素シュティーフエル多様体 $\text{St}(N, n; \mathbb{C})$ 上の実関数の最適化問題に還元される。

$$\text{St}(N, n; \mathbb{C}) = \{W \in \mathbb{C}^{N \times n} | W^H W = I_n\} \quad (8)$$

複素シュティーフエル多様体で $N = n$ の場合がユニタリー群 $U(n)$ である。

コスト関数としては、Hyvärinen, Hoyer が実独立部分空間分析を最尤推定に基づいて解くために導入したコスト関数⁹⁾を拡張した以下の関数を用いる。

$$f(W) = \sum_{j=1}^r E_z \left[\log p \left(\sum_{l=1}^{d_j} |w_{jl}^H z|^2 \right) \right] \quad (9)$$

ここに、 E_z は z についての経験平均で、 p は、前処理後の信号 z を部分空間 w_{jl} 達で張られる ($W_j =$

w_{j1}, \dots, w_{jd_j}) 部分空間へ正射影したもののパワーに関する確率密度関数である。

このコスト関数は、以下のように各 W_j ($j = 1, \dots, r$) へのユニタリー変換に関しての不変性を持っていることに注意する。

$$W \mapsto W \text{diag}[U_1, U_2, \dots, U_r], U_j \in U(d_j) \quad (10)$$

従ってコスト関数の最適化は、複素シュティーフエル多様体 $\text{St}(N, n; \mathbb{C})$ を次の同値関係 \sim で割って得られる多様体上で行われることになる。

$$W_1 \sim W_2$$

$$\iff \exists U_i \in U(d_i) \text{ s.t. } W_2 = W_1 \text{diag}[U_1, U_2, \dots, U_r] \quad (11)$$

この商空間は複素旗多様体と呼ばれ、以下では $\text{Fl}(n, d_1, \dots, d_r; \mathbb{C})$ と表すことにする。 $\text{Fl}(n, d_1, \dots, d_r; \mathbb{C})$ は等質空間としては $U(N)/U(d_1) \times \dots \times U(d_r) \times U(N - n)$ と同型で、 $r = 1$ の場合が複素グラスマン多様体となる。実、複素のグラスマン多様体は信号処理における部分空間追跡や、パターン認識における部分空間法等様々な所で用いられている。

3 リーマン幾何的最適化

多様体上の最適化問題を解くために、ターゲットとなる多様体に適切なリーマン計量を導入し、通常のユークリッド空間上で考案されてきた種々の反復的最適化法を、リーマン構造を反映させて拡張した最適化法をリーマン幾何的最適化法²⁾と呼ぶ事にする。

リーマン計量とは、多様体 M の各点の接空間 $T_p M$ 上で滑らかに定義される非退化な正定値内積 $g_p: T_p M \times T_p M \rightarrow \mathbb{R}$ のことであった。ニューラルネット学習においては、ニューラルネットの集合のなす多様体にどのリーマン計量を選ぶかによって学習の性能が変わることが良く知られており、リーマン幾何の意味での M 上の関数 f の勾配を自然勾配⁴⁾と呼ぶことが多い。自然勾配 $\text{grad}_p f(p)$ は直観的には次のように理解できる。多様体上の各点 p の近傍で、 p を中心とする微小な半径 dr の円 C を考える。この円周上で、中心 p からの関数の変化が最も激しい方向、つまり $\Delta = f(q) - f(p)$ が最大となる方向 \vec{pq} を勾配ベクトルと定義する。この M 上の微小半径 dr を測るためにリーマン計量が必要になるわけである。

以下では、滑らかな多様体 M 上の実可微分関数 f を最小化するためにリーマン構造を利用した最急降下法と共役勾配法を考えるが、その際に自然勾配と並んで必要となるものが測地線の式である。

$W \in M$ を始速度 V で出発する測地線の式を時間パラメータ t として $\varphi_M(W, V, t)$ と表すことにする。

$$\varphi_M(W, V, 0) = W, \quad \frac{d}{dt} \varphi_M(W, V, 0)|_{t=0} = V \quad (12)$$

測地線は、ユークリッド空間上の直線を多様体の場合に拡張したもので、十分近い2点を結ぶ最短線となっている。局所最短線を求めるためには多様体上に距離が定められる必要があり、これもリーマン計量から決定

される。そして測地線は計量テンソルについての微分方程式を解くことによって求められるが、リーマン幾何的最適化で使われるのは、ほとんどが、微分方程式を綺麗に解くことができ、測地線が陽に求められる場合である。多様体上の測地線の方程式を数値的に解く研究もあるが¹⁹⁾、最適化に用いられている例はまだないようである。また、測地線の表示としてよく使われる行列の指数関数の計算量は大きくなるため、行列の指数関数をより少ない計算量で求める試みや、厳密な測地線ではなく、測地線に近い曲線に沿った写像(レトラクション)²⁾を用いることにより計算量を減らす試みもさかに行われているが、本稿では触れない。

3.1 実シュティーフエル多様体上の最適化

実シュティーフエル多様体 $\text{St}(N, n)$ 上に以下のリーマン計量 g を導入する。

$$g_W(u, v) = \text{tr} u^\top \left(I - \frac{1}{2} W W^\top \right) v, u, v \in T_W \text{St}(N, n) \quad (13)$$

この計量に関して $\text{St}(N, n)$ の測地線は行列の指数関数を用いて陽に表せる。

$$\varphi_{\text{St}(N, n)}(W, V, t) = \exp(t(DW^\top - WD^\top))W \quad (14)$$

$$D = \left(I - \frac{1}{2} W W^\top \right) V \quad (15)$$

この測地線の表示式を用いて、 $M = \text{St}(N, n)$ 上のリーマン幾何的最急降下、共役勾配法の更新式は以下で与えられる。(Fig.1 参照)

- リーマン幾何的最急降下法

$$G_k = -\text{grad}_{W_k} f(W_k) \quad (16)$$

$$W_{k+1} = \varphi_M(W_k, G_k, t_k), t_k \text{ は学習係数} \quad (17)$$

- リーマン幾何的共役勾配法

$$H_0 = -\text{grad}_{W_0} f(W_0), \quad (18)$$

$$H_k = -\text{grad}_{W_k} f(W_k) + \gamma_k \tau H_{k-1} \quad (19)$$

$$t_k^{\min} = \arg \min_t f(\varphi_M(W_k, H_k, t)) \quad (20)$$

$$W_{k+1} = \varphi_M(W_k, H_k, t_k^{\min}) \quad (21)$$

$$\gamma_k = \frac{g_{W_k}(v_k - \tau v_{k-1}, v_k)}{g_{W_{k-1}}(v_{k-1}, v_{k-1})}, v_k = \text{grad}_{W_k} f(W_k)$$

ここで、 τH_{k-1} は H_{k-1} の測地線 $\varphi_M(W_{k-1}, H_{k-1}, t)$ に沿う W_k への平行移動とする。曲線に沿う平行移動もリーマン計量によって定められる。共役勾配法において γ_k を定める複数の方法が提案されているが、ここでは Fletcher-Reeves による方法を用いる。その際正確には上式右辺分子において v_{k-1} の代りに τv_{k-1} が用いられるべきであるが、 $\tau v_{k-1} = \tau \text{grad}_{W_{k-1}} f(W_{k-1})$ を低い計算量で求める方法がみつからないので、Edelman-Arias-Smith⁶⁾ が用いたのと同様の近似をここで用いる。

$\varphi_{\text{St}(N, n)}(W_k, H_k, t)$ は測地線であるから、測地線に沿う始速度の平行移動ベクトル τH_k は、平行移動の終

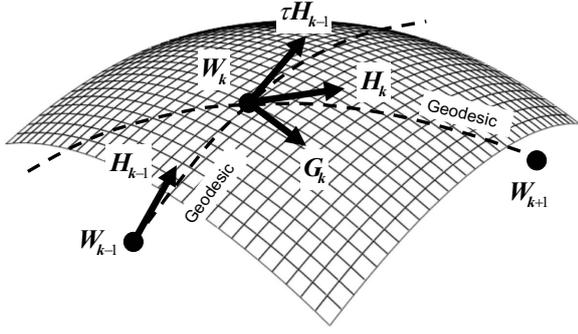


Fig. 1: リーマン幾何的共役勾配法の更新ステップ点における測地線の速度ベクトルになるので、

$$\begin{aligned} \frac{d}{dt} \varphi_{\text{St}(N,n)}(W_k, H_k, t) \\ = (D_k W_k^\top - W_k^\top D_k) \exp(t(D_k W_k^\top - W_k^\top D_k)) \end{aligned} \quad (22)$$

において $t = t_k^{\min}$ において、

$$\tau H_k = (D_k W_k^\top - W_k^\top D_k) W_{k+1} \quad (23)$$

を得る. (ここに $D_k = (I - \frac{1}{2} W_k W_k^\top) H_k$)

3.2 複素旗多様体上の最適化

まず複素シュティーフエル多様体上の実関数 F の最適化を考える.

$$\text{St}(N, n; \mathbb{C}) \ni W = W^{\Re} + iW^{\Im} \mapsto F(W) \in \mathbb{R} \quad (24)$$

前節の実シュティーフエル多様体上の最適化手法を適用するために、 $\text{St}(N, n; \mathbb{C})$ を実多様体として捉える. 定義方程式

$$W^H W = (W^{\Re\top} - iW^{\Im\top})(W^{\Re} + iW^{\Im}) = I_n \quad (25)$$

から、 $\text{St}(N, n; \mathbb{C})$ は $\mathbb{R}^{2N \times n}$ 内の部分多様体

$$M := \{W' = \begin{pmatrix} W^{\Re} \\ W^{\Im} \end{pmatrix} \in \mathbb{R}^{2N \times n}\}$$

$$W^{\Re\top} W^{\Re} + W^{\Im\top} W^{\Im} = I_n, W^{\Re\top} W^{\Im} - W^{\Im\top} W^{\Re} = O_n \quad (26)$$

とみなせるが、これをさらに次の埋め込み写像によって $\mathbb{R}^{2N \times 2n}$ 内の部分多様体として捉える.

$$v: W' \mapsto \tilde{W} = \begin{pmatrix} W^{\Re} & -W^{\Im} \\ W^{\Im} & W^{\Re} \end{pmatrix} \quad (27)$$

$$\begin{array}{ccccc} \text{St}(N, n; \mathbb{C}) & \rightarrow & M & \xrightarrow{v} & \tilde{M} & \rightarrow & \text{St}(2N, 2n; \mathbb{R}) \\ W & \mapsto & W' & \mapsto & \tilde{W} & & \end{array} \quad (28)$$

関係式 (26) から、 $\tilde{W}^\top \tilde{W} = I_{2n}$ が成り立つ、つまり、 $\tilde{M} = v(M)$ は $\text{St}(2N, 2n; \mathbb{R})$ の部分多様体になる. さらに強く全測地的部分多様体 (あるいは自己平行部分多様体) になっていることが示せる. 即ち、 \tilde{M} の測地線は必ず $\text{St}(2N, 2n; \mathbb{R})$ の標準計量に関する測地線になっている. これは埋め込み写像 $u: W \mapsto \tilde{W}$ が以下の性質を持つことに基づいている.

1. $u(I_m) = I_{2m}$
2. $u(W_1 W_2) = u(W_1) u(W_2)$, where $W_1 \in \mathbb{C}^{l \times m}, W_2 \in \mathbb{C}^{m \times n}$
3. $u(W_1 + W_2) = u(W_1) + u(W_2)$, where $W_1, W_2 \in \mathbb{C}^{m \times n}$
4. $u(W^H) = u(W)^\top$, where $W \in \mathbb{C}^{m \times m}$
5. $u(\exp(W)) = \exp(u(W))$, where $W \in \mathbb{C}^{m \times m}$

従って、 $\text{St}(N, n; \mathbb{C})$ 上の実関数 F の最適化は、実多様体 \tilde{M} 上の実関数 $f(\tilde{W}) := F(W)$ の最適化に還元され、前節の実シュティーフエル多様体上の最適化法が適用できる. そこで、リーマン幾何的最急降下法、共役勾配法のアルゴリズムを書き下すのに必要な、コスト関数の自然勾配と測地線を求める.

まず前者であるが、 \tilde{M} は $\text{St}(2N, 2n; \mathbb{R})$ の部分多様体だから、 \tilde{M} 上のコスト関数 f の自然勾配は、 f の $\text{St}(2N, 2n; \mathbb{R})$ 上 \tilde{W} での自然勾配を、 \tilde{M} の \tilde{W} での接空間に正射影することによって得られる. $\text{St}(2N, 2n; \mathbb{R})$ 上の \tilde{W} における f の自然勾配は、常に \tilde{M} の \tilde{W} での接空間に含まれるので、結局

$$\text{grad}_{\tilde{W}}^{\tilde{M}} f(\tilde{W}) = \text{grad}_{\tilde{W}}^{\text{St}(2N, 2n; \mathbb{R})} F(\tilde{W}) \quad (29)$$

$$= \tilde{\nabla} F - \tilde{W}^\top \tilde{\nabla} F \tilde{W}^\top \quad (30)$$

を得る. ここに、

$$\tilde{\nabla} F = v \left(\left(\frac{\partial F}{\partial W^{\Re}}, \frac{\partial F}{\partial W^{\Im}} \right)^\top \right) \quad (31)$$

$$= \begin{pmatrix} \frac{\partial F}{\partial W^{\Re}} & -\frac{\partial F}{\partial W^{\Im}} \\ \frac{\partial F}{\partial W^{\Im}} & \frac{\partial F}{\partial W^{\Re}} \end{pmatrix} \quad (32)$$

である.

後者は、 \tilde{M} が $\text{St}(2N, 2n; \mathbb{R})$ の全測地部分多様体であることから、

$$\varphi_{\tilde{M}}(\tilde{W}, \tilde{V}, t) = \varphi_{\text{St}(2N, 2n; \mathbb{R})}(\tilde{W}, \tilde{V}, t) \quad (33)$$

$$= \exp(t(\tilde{D} \tilde{W}^\top - \tilde{W} \tilde{D}^\top)) \tilde{W} \quad (34)$$

$$\tilde{D} = (I - \frac{1}{2} \tilde{W} \tilde{W}^\top) \tilde{V} \quad (35)$$

となる.

このようにして、複素多様体 $\text{St}(N, n; \mathbb{C})$ 上の実関数 f の最適化を実多様体上 $\text{St}(2N, 2n; \mathbb{R})$ で反復的に行うアルゴリズムが得られるが、埋め込み写像の性質を使ってこの更新式を元の複素多様体 $\text{St}(N, n; \mathbb{C})$ 上に引き戻すことができる.

$\text{St}(2N, 2n; \mathbb{R})$ 上の測地線を $\text{St}(N, n; \mathbb{C})$ に引き戻した曲線は、実の共役 \top をエルミート共役 H に置き換えれば得られる.

$$\varphi_{\text{St}(N, 2n; \mathbb{C})}(W, V, t) = \exp(t(DW^H - W D^H)) W \quad (36)$$

$$D = (I - \frac{1}{2} W W^H) V, V = u^{-1} \tilde{V} \quad (37)$$

同様に、 $\text{St}(2N, 2n; \mathbb{R})$ 上の自然勾配を $\text{St}(N, n; \mathbb{C})$ に引き戻すには、複素勾配を用いる.

$$\frac{\partial F}{\partial W} := \frac{1}{2} \left(\frac{\partial F}{\partial W^{\Re}} + i \frac{\partial F}{\partial W^{\Im}} \right) \quad (38)$$

この複素勾配を用いる演算はヴィルティンガー・カリキュラス、あるいは $\mathbb{C}-\mathbb{R}$ カリキュラスと呼ばれ、複素多様体上の最適化を考察する際に多くの研究者に用いられている¹⁾³⁾⁵⁾⁷⁾⁸⁾¹⁰⁾¹²⁾¹⁸⁾.

$$\tilde{\nabla} F \mapsto 2 \frac{\partial F}{\partial \bar{W}}, \quad \text{並びに} \quad \top \mapsto H \quad (39)$$

との置き換えで、

$$\text{grad}_W^{\text{St}(N,n;\mathbb{C})} F = 2 \left(\frac{\partial F}{\partial \bar{W}} - W \frac{\partial F}{\partial \bar{W}}^H W \right) \quad (40)$$

を得る。ただし、複素多様体上の自然勾配は定義されていないので、 $\text{St}(2N, 2n; \mathbb{R})$ 上の自然勾配の $\text{St}(N, n; \mathbb{C})$ への引き戻しを形式的に左辺のように表した。

今までの複素シュティーフエル多様体の議論と同様に、以下の埋め込み写像を用いると、複素旗多様体 $\text{Fl}(n, d_1, \dots, d_r; \mathbb{C})$ を実旗多様体 $\text{Fl}(n, 2d_1, \dots, 2d_r; \mathbb{R})$ の全実部分多様体とみなすことができる。

$$\begin{aligned} v: W = [W_1, W_2, \dots, W_r] &\mapsto \tilde{W} \\ &= \begin{pmatrix} W_1^{\Re} - W_1^{\Im} & W_2^{\Re} - W_2^{\Im} & \dots & W_r^{\Re} - W_r^{\Im} \\ W_1^{\Im} & W_1^{\Re} & W_2^{\Im} & W_2^{\Re} & \dots & W_r^{\Im} & W_r^{\Re} \end{pmatrix} \end{aligned} \quad (41)$$

ただし $W_k = W_k^{\Re} + iW_k^{\Im} \in \mathbb{C}^{N \times d_k}$ 、 $\sum_{k=1}^r d_k = n$ 。

従って、実の自然勾配を複素の自然勾配に、共役演算をエルミート共役演算に置き換えることで、同様にして実旗多様体の場合の自然勾配と測地線を表す結果¹⁵⁾を複素旗多様体の自然勾配と測地線を表す結果に変換できる。

$$\text{grad}_W^{\text{Fl}(n,d_1,\dots,d_r;\mathbb{C})} F = [V_1^{\mathbb{C}}, V_2^{\mathbb{C}}, \dots, V_r^{\mathbb{C}}] \quad (42)$$

$$V_j^{\mathbb{C}} = X_j^{\mathbb{C}} - (W_j W_j^H X_j^{\mathbb{C}} + \sum_{l \neq j} W_l X_l^{\mathbb{C}H} W_j) \quad (43)$$

ここに

$$X_j^{\mathbb{C}} = \frac{\partial F}{\partial W_j^{\Re}} + i \frac{\partial F}{\partial W_j^{\Im}}. \quad (44)$$

$$\varphi_{\text{Fl}(n,d_1,\dots,d_r;\mathbb{C})}(W, V, t) = \exp(t(DW^H - WD^H))W \quad (45)$$

ただし、 $D = (I - \frac{1}{2}WW^H)V$ 。これらを(16)–(21)に代入して複素旗多様体上の更新則を得る。数値実験結果は文献(15)を参照されたい。

4 おわりに

複素信号処理の問題が複素多様体上の実関数の最適化を通して解かれる例として、複素独立部分空間分析をとりあげた。埋め込み写像を用いることにより、ユニタリ群に関する複素多様体上の実関数のリーマン幾何的最適化法の更新式が、直交群に関する実多様体上のリーマン幾何的最適化法の更新式から一定の手続きで導かれることを示した。正定置エルミート行列のなす複素多様体等、他の複素多様体の場合にも実の場合とアナログカルなアルゴリズムの導出が予想され、今後の複素信号処理、複素ニューラルネットにおける複素多様体上のリーマン幾何的最適化の応用が期待される。四元数値の行列の場合にも類似のことが可能であるかもしれない。

参考文献

- 1) T. Abrudan, J. Eriksson, and V. Koivunen, Steepest Descent Algorithms for Optimization under Unitary Matrix Constraint, *IEEE Transactions on Signal Processing*, Vol. 56, No. 3, 1134/1147(2007).
- 2) P. A. Absil, R. Mahony, R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*, Princeton University Press, 2007.
- 3) T. Adali and H. Li, A practical formulation for computation of complex gradients and its application to maximum likelihood, *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing (ICASSP)*, Vol.2, 633/636(2007).
- 4) S. Amari, Natural gradient works efficiently in learning, *Neural Computation*, **10**, 251/276(1998).
- 5) D.H. Brandwood, A Complex Gradient Operator and its Application in Adaptive Array Theory, *IEE Proceedings H (Microwaves, Optics, and Antennas)*, Vol. 130, No. 1, 11/16(1983).
- 6) A. Edelman, T.A. Arias, and S.T. Smith, The geometry of algorithms with orthogonality constraints, *SIAM Journal on Matrix Analysis and Applications*, **20** (2), 303/353(1998).
- 7) S. Fiori, A Study on Neural Learning on Manifold Foliations: The Case of the Lie Group $\text{SU}(3)$, *Neural Computation*, **4**, 1091/1197(2008).
- 8) A. Hjørungnes, *Complex-Valued Matrix Derivatives: With Applications in Signal Processing and Communications*, Cambridge University Press, 2011.
- 9) A. Hyvärinen and P.O. Hoyer, Emergence of phase and shift invariant features by decomposition of natural images into independent feature subspaces. *Neural Computation*, **12**(7), 1705/1720(2000).
- 10) Kenneth Kreutz-Delgado, The Complex Gradient Operator and the CR-Calculus, technical report, University of California, San Diego, (2003).
- 11) D. Mandic, V. Goh, *Complex Valued Nonlinear Adaptive Filters*, Wiley, 2009
- 12) J. Manton, Optimization algorithms exploiting unitary constraints, *IEEE Transactions on Signal Processing*, Volume 50, Issue 3, 635/650(2002).
- 13) Y. Nishimori and S. Akaho, Learning algorithms utilizing quasi-geodesic flows on the Stiefel manifold, *Neurocomputing*, **67** 106/135(2005).
- 14) Y. Nishimori, S. Akaho and M.D. Plumbley, Riemannian optimization method on the flag manifold for independent subspace analysis, *Proceedings of ICA2006*, 295/302(2006).
- 15) Y. Nishimori, S. Akaho, S. Abdallah, and M.D. Plumbley, Flag manifolds for subspace ICA problems, *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2007)*, Vol IV, 1417/1420(2007).
- 16) Y. Nishimori, S. Akaho, and M.D. Plumbley, Natural Conjugate Gradient on Complex Flag Manifolds for Complex Independent Subspace Analysis, *Proceedings of the International Conference on Artificial Neural Networks (ICANN 2008)*, Part I, 165/174(2008).
- 17) P. J. Schreier, L. L. Scharf, *Statistical Signal Processing of Complex-Valued Data: The Theory of Improper and Noncircular Signals*, Cambridge University Press, 2010.
- 18) A. van den Bos, Complex Gradient and Hessian, *IEE Proc.-Vis. Image Signal Processing*, **141**(6),380/82(1994).
- 19) L. Ying, E. J. Candès, Fast geodesics computation with the phase flow method, *Journal of Computational Physics*, Vol.220(6), 6/18(2006).

高次元関数最適化へのDX-NESの適用

○福島信純 永田裕一 小野功 (東京工業大学)

Applying DX-NES to High Dimensional Function Optimization

*N. Fukushima, Y. Nagata and I. Ono (Tokyo Institute of Technology)

Abstract— The Distance-weighted Exponential Natural Evolution Strategy (DX-NES) is a powerful method for black-box optimization where no algebraic expressions of objective functions are given explicitly. DX-NES consists of a natural gradient method and some heuristic techniques. DX-NES shows high performance on a set of several benchmark functions that includes ill-conditioned, non-separable and multimodal functions. However, these results were obtained in relatively low-dimensional objective functions with less than 100 decision variables. In this paper, we modify some internal parameters of DX-NES in order to apply DX-NES to higher-dimensional objective functions and investigate the increasing rate of the number of evaluations and the wall-time as the dimension of benchmark functions increases.

Key Words: black-box optimization, evolutionary computation, natural gradient

1 はじめに

与えられた制約条件の下で所要の目的を最も良く満たす解を発見するという問題は、科学研究や産業応用のさまざまな場面において日常的に現れる。このような問題は関数最適化(数理計画)とよばれる。関数最適化は、制約条件を満たす定義域 $\Omega \subset \mathbb{R}^n$ の下で、目的関数 f (の評価値 $f(\mathbf{x})$) を最小化する解 $\mathbf{x} \in \Omega$ (最適解) の発見を目標とする問題として定式化される。

実世界に現れる関数最適化の目的関数は、代数的表現さえ得られない black-box な関数であり、局所的な最適解を多く有することが多い。このような black-box な目的関数を最適化するために、進化計算の枠組みの下で遺伝的アルゴリズム (Genetic Algorithm), 進化戦略 (Evolution Strategies), 差分進化 (Differential Evolution) や粒子群最適化 (Particle Swarm Optimization) など多くの方法論が提案されている。著者らが提案した Distance-weighted Exponential Natural Evolution Strategy (DX-NES)³⁾ は進化戦略に属する手法であり、進化戦略の中で state-of-the-art な手法の一つとして知られる CMA-ES^{5, 6, 7)} と比べて良好な性能を示すことを実験的に確認している³⁾。しかしながら、これまでの多くの進化計算に関する研究と同様、DX-NES の有効性は 100 次元以下の比較的低次元の目的関数に対してのみでしか実験的に確認されていなかった。

本論文では、いくつかのベンチマーク関数で次元数を 2000 まで増やしていったときに、DX-NES による最適化が完了するまでに必要な解の評価回数と実行時間がどのように増加するかを調査する。また、これに併せて目的関数が低次元の場合に対して調整されていた DX-NES の内部パラメータの推奨値を、高次元の場合でも DX-NES が効率的に動作するように再設定する。

本論文の構成は以下の通りである。まず、第2節で関数最適化について述べる。次に、第3節では DX-NES のアルゴリズムについての説明を行い、第4節で DX-NES の内部パラメータの推奨値を高次元の目的関数に対しても効果的な設定となるように再設定する。そして、第5節で高次元問題に DX-NES を適用するいくつかの計算機実験を行う。最後に、第6節で本論文のとりまとめを行う。

2 関数最適化

関数最適化は、制約条件を満たす定義域 $\Omega \subset \mathbb{R}^n$ の下で、目的関数 f (またはその評価値 $f(\mathbf{x})$) を最小化する問題として次式のように定式化される。

$$\begin{aligned} & \text{minimize } f(\mathbf{x}) \\ & \text{where } \mathbf{x} \in \Omega, f: \Omega \rightarrow \mathbb{R}. \end{aligned} \quad (1)$$

このとき、 $f(\mathbf{x}^*) \leq f(\mathbf{x}), \forall \mathbf{x} \in \Omega$ を満たす解 \mathbf{x}^* は (大域) 最適解とよばれ、 \mathbf{x}^* の適当な近傍 $N(\mathbf{x}^*)$ に対して $f(\mathbf{x}^*) \leq f(\mathbf{x}), \forall \mathbf{x} \in N(\mathbf{x}^*)$ を満たす解 \mathbf{x}^* は局所最適解とよばれる。

実問題に現れる関数最適化の目的関数は、その代数的表現が $f(x) = x^2 + x + 1$ のように明示的に与えられない black-box 関数であることが多い。

black-box 関数の最適化を困難にする性質としては、悪スケール性 (ill-condition), 変数間依存性 (non-separability), 多峰性 (multimodality) がある。悪スケール性は目的関数に対する感度が解の変数ごとに大きく異なる性質である (Fig. 1)。変数間依存性は解の変数間に非線形な依存関係が存在するため、目的関数を効率的に改善するために複数の変数を同時に変化させることが必要な性質である (Fig. 2)。多峰性は定義域内に複数の局所最適解が存在する性質である (Fig. 3, 4)。

悪スケール性関数や変数間依存性を持つ目的関数を効率的に探索するためには、最適化手法が解空間のアフィン変換に対して不変性を有することが望ましい。また、多峰性関数については多点探索法やマルチスタート法などにより局所最適解を避ける必要がある。

3 DX-NES

Distance-weighted Exponential Natural Evolution Strategy (DX-NES)³⁾ は black-box な目的関数を最適化するための多点探索法であり、Exponential NES (xNES)⁴⁾ とよばれる最適化手法の改良法として著者らが提案した手法である。これらの手法はともに NES¹⁰⁾ とよばれる最適化の枠組みに属する。

以下では、NES の枠組みと xNES について説明した後、DX-NES における xNES からの改良点を説明し、最後に DX-NES のアルゴリズム全体を示す。

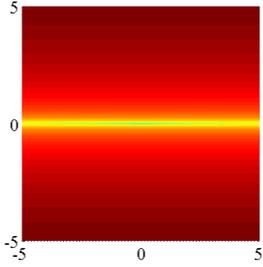


Fig. 1: An example of ill-conditioned function. (1-tablet function)

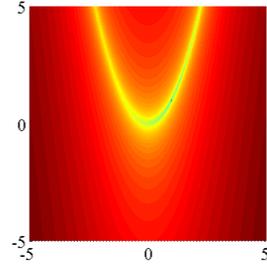


Fig. 2: An example of non-separable function. (Rosenbrock function)

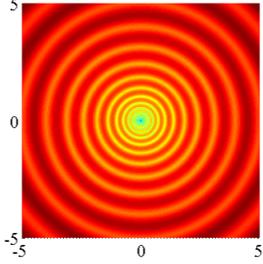


Fig. 3: An example of multimodal function. (Schaffer function)

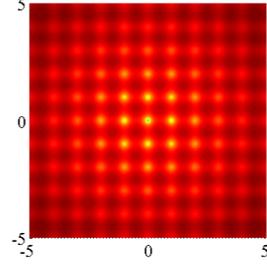


Fig. 4: An example of multimodal function. (Rastrigin function)

3.1 NES : Natural Evolution Strategy

NES は目的関数の評価値 $f(\mathbf{x})$ を直接最小化する代わりに、 m 個のパラメータ θ によって決まる確率密度関数 $p(\mathbf{x} | \theta)$ を導入して得られる期待評価値 $J(\theta)$ を最小化する。すなわち、NES は (1) の代わりに以下の関数最適化

$$\begin{aligned} \text{minimize } J(\theta) &= \int_{\Omega} f(\mathbf{x}) p(\mathbf{x} | \theta) d\mathbf{x} \\ \text{where } \theta &\in \mathbb{R}^m, \mathbf{x} \in \Omega, f: \Omega \rightarrow \mathbb{R} \end{aligned} \quad (2)$$

を行うことで black-box な目的関数の最適化を行う。

NES は、新たな関数最適化問題 (2) を解くにあたって、自然勾配法²⁾ を利用する。自然勾配法は最急降下法の改良法としてよく知られている Newton 法に類似した手法であり、そのアルゴリズムは次のようになる。

1. 初期パラメータ $\theta^{(0)}$ を決め、反復数 $g = 0$ とする。
2. 現在のパラメータ $\theta^{(g)}$ 周りで期待評価値の勾配 $\nabla_{\theta} J(\theta)|_{\theta=\theta^{(g)}}$ を計算する。
3. パラメータを次のように更新する。

$$\theta^{(g+1)} = \theta^{(g)} - \eta \mathbf{F}^{-1}(\theta) \nabla_{\theta} J(\theta) \Big|_{\theta=\theta^{(g)}} \quad (3)$$

ここで、 $\eta > 0$ は学習率、 $\mathbf{F}(\theta)$ は Fisher 情報行列

$$\mathbf{F}(\theta) = \int_{\Omega} \nabla_{\theta} p(\mathbf{x} | \theta) (\nabla_{\theta} p(\mathbf{x} | \theta))^T p(\mathbf{x} | \theta) d\mathbf{x}$$

であり、 $\mathbf{F}^{-1}(\theta) \nabla_{\theta} J(\theta)$ は自然勾配とよばれる。

4. 終了条件が満たされれば停止、そうでなければ $g = g + 1$ としてステップ 2 へ戻る。

このとき、目的関数 f が black-box なので、ステップ 3. の (3) 式の期待評価値の勾配 $\nabla_{\theta} J(\theta)$ は解析的に

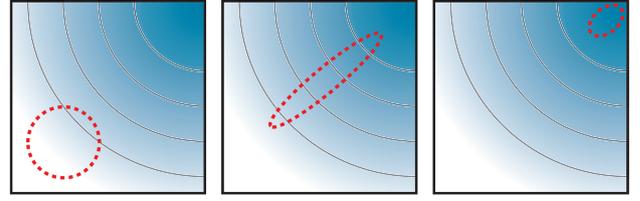


Fig. 5: A typical behavior of xNES(DX-NES). The shape of normal distribution (red dashed circle) changes from the left figure to right ones. Note that the deeper color the area has, the better objective function value the area has.

計算できない。しかし、期待評価値の勾配を

$$\begin{aligned} \nabla_{\theta} J(\theta) &= \int_{\Omega} f(\mathbf{x}) \nabla_{\theta} p(\mathbf{x} | \theta) d\mathbf{x} \\ &= \int_{\Omega} f(\mathbf{x}) p(\mathbf{x} | \theta) \nabla_{\theta} \ln p(\mathbf{x} | \theta) d\mathbf{x} \\ &= \mathbb{E}_{\mathbf{x}} [f(\mathbf{x}) \nabla_{\theta} \ln p(\mathbf{x} | \theta)] \end{aligned}$$

と期待値の形に式変形できるので、モンテカルロ法により次のように推定することができる。

$$\nabla_{\theta} J(\theta) \approx \frac{1}{\lambda} \sum_{i=1}^{\lambda} f(\mathbf{x}_i) \nabla_{\theta} \ln p(\mathbf{x}_i | \theta) \quad (4)$$

ここで、 \mathbf{x}_i は $p(\mathbf{x} | \theta)$ に従う λ 個のサンプルされた解である。

後述する xNES や DX-NES は確率分布として多変量正規分布を用いている。このとき、確率分布は NES の最適化の過程において Fig. 5 のように、分布の中心 (平均ベクトル) が最適解へ近づいていくと共に分布が最適解方向に伸び、その後最適解へ収束するという挙動を示す。

3.2 xNES : Exponential NES

xNES は確率分布として多変量正規分布を用いる NES の一つとして提案された手法である。多変量正規分布を用いることにより、xNES は解空間に対する回転や拡大・縮小に対して不変性を有し、悪スケール性や変数間依存性を持つ関数において比較的良好な性能を示す。

3.2.1 Exponential Parametrization

多変量正規分布の素直なパラメータ化は平均ベクトル $\mathbf{m} \in \mathbb{R}^n$ と共分散行列 $\mathbf{C} \in \mathbb{R}^{n \times n}$ を使ったものであるが、xNES では共分散行列に関する部分のパラメータ化については exponential parametrization⁴⁾ とよばれるパラメータ化を行う。

exponential parametrization では、 \mathbf{C} の代わりに $\mathbf{C} = \exp(\mathbf{M})$ を満たす対称行列 \mathbf{M} をパラメータとして (3) 式の更新を行う。ここで、 $\exp(\cdot)$ は行列指数関数であり、 $\exp(\mathbf{M}) = \sum_{k=0}^{\infty} \mathbf{M}^k / k!$ である。このようなパラメータ化を行うと、パラメータ更新の前後で共分散行列の半正定値対称性が保たれる^{*1}。また、共分散行列 \mathbf{C} を分布の拡がりやを制御するステップサイズ $\sigma \in \mathbb{R}$ と分布の形状を制御する正規化変換行列 $\mathbf{B} \in \mathbb{R}^{n \times n}$ に分けて独立に更新することができる。このとき、 $\mathbf{C} = \sigma \mathbf{B} (\sigma \mathbf{B})^T$ かつ $\det(\mathbf{B}) = 1$ である。

*1 $\mathbf{C} = \mathbf{C} - \eta \Delta \mathbf{C}$ のように更新すると、更新の前後で \mathbf{C} の半正定性が崩れることがある。

Fisher 情報行列と期待評価値の勾配の推定式 (4) に、 $\theta = [\mathbf{m}^T \text{vec}(\mathbf{M})^T]^T$ *2 と正規分布の確率密度関数

$$p(\mathbf{x} | \mathbf{m}, \mathbf{C}) = \frac{1}{\sqrt{(2\pi)^n}} \exp\left(-\frac{(\mathbf{x} - \mathbf{m})^T \mathbf{C}^{-1} (\mathbf{x} - \mathbf{m})}{2}\right)$$

を代入した結果を整理すると、ステップ 3. の更新則 (3) は以下ようになる⁴⁾。

$$\begin{aligned} \mathbf{m}^{(g+1)} &= \mathbf{m}^{(g)} + \eta_{\mathbf{m}} \sigma^{(g)} \mathbf{B}^{(g)} \mathbf{G}_{\delta}, \\ \sigma^{(g+1)} &= \sigma^{(g)} \exp(\eta_{\sigma} G_{\sigma}/2), \\ \mathbf{B}^{(g+1)} &= \mathbf{B}^{(g)} \exp(\eta_{\mathbf{B}} \mathbf{G}_{\mathbf{B}}/2). \end{aligned}$$

ここで、 $\eta_{\mathbf{m}}$, η_{σ} , $\eta_{\mathbf{B}}$ はそれぞれの添字のパラメータに対応する学習率を示す。また、

$$\begin{aligned} \mathbf{G}_{\delta} &= -\sum_{i=1}^{\lambda} \frac{f(\mathbf{x}_i)}{\lambda} \mathbf{z}_i, \quad \mathbf{G}_{\mathbf{M}} = -\sum_{i=1}^{\lambda} \frac{f(\mathbf{x}_i)}{\lambda} (\mathbf{z}_i \mathbf{z}_i^T - \mathbf{I}), \\ G_{\sigma} &= \text{tr}(\mathbf{G}_{\mathbf{M}})/n, \quad \mathbf{G}_{\mathbf{B}} = \mathbf{G}_{\mathbf{M}} - G_{\sigma} \mathbf{I}. \end{aligned}$$

であり、 $\mathbf{z}_i = (\mathbf{C}^{(g)})^{-1/2} (\mathbf{x}_i - \mathbf{m}^{(g)})$ は解 \mathbf{x}_i を正規分布で標準化したベクトル、 \mathbf{I} は $n \times n$ 単位行列である。

3.2.2 fitness shaping

fitness shaping¹⁰⁾ は、目的関数の評価値に対して単調増加関数 g を用いて $f_{\text{new}}(\mathbf{x}) = g(f(\mathbf{x}))$ のように変換を加えたときでも、性能が変化しないようにするために導入されている。

fitness shaping では、(4) 式において $-f(\mathbf{x}_i)/\lambda$ を評価値 $f(\mathbf{x}_i)$ の順位 i_{ord} のみに応じて決まる重み関数 $w_{i_{\text{ord}}}$ へと置き換える変更が行われる。これにより、評価値の増加率の非線形性が勾配に反映されなくなるので、上記の目的関数への変換に対して xNES は不変性を有する。

重み関数は、 $\sum_{i_{\text{ord}}=1}^{\lambda} w_{i_{\text{ord}}}^{\text{rank}} = 1$ と $f(\mathbf{x}_i) \leq f(\mathbf{x}_j) \Leftrightarrow w_{i_{\text{ord}}}^{\text{rank}} \geq w_{j_{\text{ord}}}^{\text{rank}}$ を満たし、 $w_{i_{\text{ord}}}^{\text{rank}} \geq 0$ ($i_{\text{ord}} < \lambda/2$) かつ $w_{i_{\text{ord}}}^{\text{rank}} = 0$ ($i_{\text{ord}} \geq \lambda/2$) となるように設計すると良い性能が得られることが経験的にわかっている^{4, 6)}。xNES や DX-NES では Table 1 に示す重み関数が用いられる。

3.2.3 fitness baseline

fitness baseline^{8, 9)} は期待評価値の勾配を推定する際に重み関数から定数を引くことで、推定量の平均を変えずに分散を減らす仕組みである。xNES では、重み関数から $\frac{1}{\lambda}$ を引くことで $w_{i_{\text{ord}}}^{\text{rank}} - \frac{1}{\lambda}$ としている。

3.3 DX-NES

DX-NES は xNES の改良法として著者らが提案した手法である。以降では、DX-NES で追加されている改良点の詳細について述べる。

3.3.1 distance-weight

期待評価値を改善するとき、xNES では分布を縮めることで悪い評価値を持つ領域での確率密度を減少させる挙動と平均ベクトルを移動させることでより良い評価値を持つ領域での確率密度を増加させる挙動が発生する。xNES では前者の挙動の効果が強くなりやすく、分布の拡がりが増えすぎると同時に平均ベクトルが移動しづらくなる現象が発生しやすい³⁾。

distance-weight は、Fig. 1 左図や中央図のように分布の平均ベクトルと (局所) 最適解の位置が離れている

*2 $\text{vec}(\mathbf{M})$ は $n \times n$ 行列 \mathbf{M} の列ベクトルを全て縦に並べた $n^2 \times 1$ 列ベクトルである。

状況下で、この現象が起きたときに分布の再拡大と移動を加速させる仕組みである。

平均ベクトルと最適解が離れている状況下で、正規分布が過剰に縮小すると平均ベクトルの更新量のマハラノビス距離は大きくなっていく。そこで、次節で述べる移動判定において、分布が過剰に縮小しており再拡大と移動を加速すべきと判定されたときには、fitness shaping で導入された重み関数 w_i^{rank} の代わりに以下に示す重み関数 (distance-weight) を用いる。

$$w_{i_{\text{ord}}}^{\text{dist}} = \frac{w_{i_{\text{ord}}}^{\text{rank}} \exp(\alpha \|\mathbf{z}_{i_{\text{ord}}}\|)}{\sum_{j_{\text{ord}}=1}^{\lambda} w_{j_{\text{ord}}}^{\text{rank}} \exp(\alpha \|\mathbf{z}_{j_{\text{ord}}}\|)} \quad (5)$$

ここで、拡張係数 $\alpha > 0$ は distance-weight の非線形性の強さを制御する内部パラメータであり、 $\|\mathbf{z}_i\|$ はそれぞれの解の平均ベクトルからのマハラノビス距離 $\|\mathbf{z}_i\| = \sqrt{(\mathbf{x}_i - \mathbf{m}^{(g)})^T (\mathbf{C}^{(g)})^{-1} (\mathbf{x}_i - \mathbf{m}^{(g)})}$ である。

$\exp(\alpha \|\mathbf{z}_i\|)$ の値はマハラノビス距離の意味で平均ベクトルから離れた位置に生成された解ほど大きくなる。そのため、期待評価値の勾配を推定する際にはそれらの解が重視されやすくなり、分布は現在の平均ベクトルの位置から遠く離れた方向へと変化しやすくなる。

このとき、前節で述べたように w_i^{rank} が $w_i^{\text{rank}} = 0$ ($i \geq \lambda/2$) を満たしていると、評価値の悪い解の重みは常に 0 となり、distance-weight により評価値の悪い解の重みは増加しない。

3.3.2 移動判定

前節で述べたように、正規分布が過剰に縮小すると平均ベクトルの更新量のマハラノビス距離は大きくなる。そのため、distance-weight の切り替えのための移動判定には、平均ベクトルの更新量のマハラノビス距離と閾値との比較を行い、その距離が閾値を超えたときに distance-weight を用いることが自然である。

しかし、平均ベクトルの更新量はサンプルから推定される値であり誤差が含まれているので、代わりに進化パス \mathbf{p}_{σ} ⁶⁾ とよばれるベクトルを利用する。

進化パスは正規分布で標準化された平均ベクトルの更新量を複数反復に渡って加重平均したベクトルであり、各反復において以下のように更新される。

$$\mathbf{p}_{\sigma}^{(g+1)} = (1 - c_{\sigma}) \mathbf{p}_{\sigma}^{(g)} + \sqrt{c_{\sigma}(2 - c_{\sigma})} \mu_{\text{eff}} \sum_{i=1}^{\lambda} w_{i_{\text{ord}}}^{\text{rank}} \mathbf{z}_i$$

ここで、 c_{σ} は進化パスの学習率である。また、 $\mu_{\text{eff}} = 1 / \sum_{i=1}^{\lambda} (w_{i_{\text{ord}}}^{\text{rank}})^2$ は、標準化された平均ベクトルの更新量 $\sum_{i=1}^{\lambda} w_{i_{\text{ord}}}^{\text{rank}} \mathbf{z}_i$ の大きさが $w_{i_{\text{ord}}}^{\text{rank}}$ の設定により変化するため、それを正規化するために導入されている。このとき、distance-weight によって進化パスが振動することを防ぐため、パラメータ更新量の計算のときと違って重み関数は $w_{i_{\text{ord}}}^{\text{rank}}$ に固定される。

閾値 ϵ には目的関数が定数関数のときの進化パスの長さ $\|\mathbf{p}_{\sigma}\|$ の期待値⁶⁾

$$\mathbb{E}\|\mathcal{N}(\mathbf{0}, \mathbf{I})\| = \sqrt{2} \frac{\Gamma((n+1)/2)}{\Gamma(n/2)} \sim \sqrt{n} \left(1 - \frac{1}{4n} + \frac{1}{21n^2}\right)$$

を用いる。

3.3.3 学習率の切り替え

DX-NES では、次に示す 2 つの考え方に基づいて、探索の状況に応じた学習率の切り替えを行っている。

Algorithm 1 The DX-NES Algorithm

```

1: Input:  $n \in \mathbb{N}$ ,  $\lambda \in \mathbb{E}$ ,  $f: \mathbb{R}^n \rightarrow \mathbb{R}$ ,  $\mathbf{m}^{(0)} \in \mathbb{R}^n$ ,  $\sigma^{(0)} \in \mathbb{R}$ ,  $\mathbf{B}^{(0)} \in \mathbb{R}^{n \times n}$ 
2:  $\mathbf{p}_\sigma^{(0)} \leftarrow \mathbf{0}$ ,  $g \leftarrow 0$ 
3: while stopping condition not met do
4:   for  $j \in 1, \dots, \lambda/2$  do
5:      $\mathbf{z}_{2j} \leftarrow \mathcal{N}(\mathbf{0}, \mathbf{I})$ ,  $\mathbf{z}_{2j+1} \leftarrow -\mathbf{z}_{2j}$ ,  $\mathbf{x}_{2j} \leftarrow \mathbf{m}^{(g)} + \sigma^{(g)}\mathbf{B}^{(g)}\mathbf{z}_{2j}$ ,  $\mathbf{x}_{2j+1} \leftarrow \mathbf{m}^{(g)} + \sigma^{(g)}\mathbf{B}^{(g)}\mathbf{z}_{2j+1}$ 
6:   end for
7:   sort  $\{(\mathbf{z}_i, \mathbf{x}_i)\}$  with respect to  $f(\mathbf{x}_i)$ 
8:    $\mathbf{p}_\sigma^{(g+1)} \leftarrow (1 - c_\sigma)\mathbf{p}_\sigma^{(g)} + \sqrt{c_\sigma(2 - c_\sigma)\mu_{\text{eff}} \sum_{i=1}^{\lambda} w_i^{\text{rank}} \mathbf{z}_i}$ 
9:   if  $\|\mathbf{p}_\sigma\| \geq \epsilon$  then  $w_i^{(g)} = w_i^{\text{dist}}$  else  $w_i^{(g)} = w_i^{\text{rank}}$  end if
10:  if  $\|\mathbf{p}_\sigma\| \geq \epsilon$  then  $\eta_\sigma^{(g)} = \eta_\sigma^{\text{move}}$ ,  $\eta_{\mathbf{B}}^{(g)} = \eta_{\mathbf{B}}^{\text{move}}$ 
11:  else if  $\|\mathbf{p}_\sigma\| \geq 0.1\epsilon$   $\eta_\sigma^{(g)} = \eta_\sigma^{\text{stag}}$ ,  $\eta_{\mathbf{B}}^{(g)} = \eta_{\mathbf{B}}^{\text{stag}}$  then
12:  else  $\eta_\sigma^{(g)} = \eta_\sigma^{\text{conv}}$ ,  $\eta_{\mathbf{B}}^{(g)} = \eta_{\mathbf{B}}^{\text{stag}}$  end if
13:   $\mathbf{G}_\delta \leftarrow \sum_{i=1}^{\lambda} (w_i^{(g)} - 1/\lambda) \mathbf{z}_i$ ,  $\mathbf{G}_M \leftarrow \sum_{i=1}^{\lambda} (w_i^{(g)} - 1/\lambda) (\mathbf{z}_i \mathbf{z}_i^T - \mathbf{I})$ ,  $G_\sigma \leftarrow \text{tr}(\mathbf{G}_M)/n$ ,  $\mathbf{G}_B \leftarrow \mathbf{G}_M - G_\sigma \mathbf{I}$ 
14:   $\mathbf{m}^{(g+1)} \leftarrow \mathbf{m}^{(g)} + \eta_m \sigma^{(g)} \mathbf{B}^{(g)} \mathbf{G}_\delta$ ,  $\sigma^{(g+1)} \leftarrow \sigma^{(g)} \exp\left(\eta_\sigma^{(g)} G_\sigma / 2\right)$ ,  $\mathbf{B}^{(g+1)} \leftarrow \mathbf{B}^{(g)} \exp\left(\eta_{\mathbf{B}}^{(g)} \mathbf{G}_B / 2\right)$ 
15:   $g \leftarrow g + 1$ 
16: end while

```

第一の切り替えは、分布が移動しているときとそうでないときに関するものである。分布の移動中には移動方向に対応する部分空間を探索すればよい。この部分空間は一般に解空間全体の次元よりも低い次元の空間となるため、分布が移動していないときよりも期待評価値の勾配を精度よく推定できると考えられる。そのため、DX-NES では分布の移動中に推定された勾配を信頼して、学習率を大きくしている。

第二の切り替えは、分布の拡がりに関するものである。分布の平均ベクトルがマハラノビス距離の意味でほとんど移動していないときには、分布の拡がりが増えすぎていると考えられる。よって、分布の収縮を加速させるために、ステップサイズ(分布の拡がり)の学習率を大きくしている。

以上の2つの学習率の切り替えには、distance-weightの切り替えのために用いた進化パスを利用する。第一の切り替えは前節の移動判定と同様に、 $\epsilon \leq \|\mathbf{p}_\sigma\|$ (移動)の場合と $\|\mathbf{p}_\sigma\| < \epsilon$ の場合に区切って行う。第二の切り替えは $\|\mathbf{p}_\sigma\| < 0.1\epsilon$ (収束)の場合と $0.1\epsilon \leq \|\mathbf{p}_\sigma\| < \epsilon$ (停滞)の場合に区切って行う。

3.3.4 対称変量法

期待評価値の勾配をモンテカルロ法で推定するとき、その推定量の分散を減少させるため、正規分布に従う解 \mathbf{x}_i を生成する際に対称変量法を用いる。

対称変量法は正規分布の平均ベクトルに対して点対称な位置にある2つの解を一組にして解を生成する方法であり、生成された解の奇数次の統計量(平均, 歪度, ...) は元の正規分布と同じ値となる。

3.3.5 DX-NESの全体

以上の内容をまとめたDX-NESのアルゴリズムの全体を Algorithm. 1 に示す。

ユーザーが最適化したい目的関数と分布の初期パラメータ、一反復で生成する解のサンプル数を入力した(1行目)後、その情報を基にアルゴリズムの内部状態が初期化される(2行目)。アルゴリズムのメインループ内(3-16行目)では、終了条件が満たされるまで以下の手順が反復される。まず、対称変量法を用いて正規

Table 1: The default settings of internal parameters given at ³⁾. All internal parameters, except for sample size λ , are determined from the dimension of objective function n and sample size λ .

index	default setting
w_i^{rank}	$\frac{\max(0, \log(\frac{\lambda}{2} + 1) - \log(i))}{\sum_{j=1}^{\lambda} \max(0, \log(\frac{\lambda}{2} + 1) - \log(j))}$
α	$(0.9 + 0.15 \log(n)) \min\left(1.0, \sqrt{\frac{\lambda}{n}}\right)$
η_m	1.0
$\eta_\sigma^{\text{move}}$	1.0
$\eta_\sigma^{\text{stag}}$	$0.5 \left(1.0 + \frac{\lambda}{\lambda + 2n}\right)$
$\eta_\sigma^{\text{conv}}$	$1.0 + \frac{\lambda}{\lambda + 2n}$
$\eta_{\mathbf{B}}^{\text{move}}$	$\frac{\lambda + 2n}{\lambda + 2n^2 + 100} \min\left(1.0, \sqrt{\frac{\lambda}{n}}\right)$
$\eta_{\mathbf{B}}^{\text{stag}}$	$\frac{\lambda}{\lambda + 2n^2 + 100}$
c_σ	$\frac{1}{\sqrt{n} n + \mu_{\text{eff}} + 5.0}$

乱数に従う解が生成(4-6行目)され、解の評価値が計算された後に、評価値の昇順に従って解がソートされる(7行目)。次に、進化パスを更新(8行目)し、進化パスのL2ノルムを基に重み関数と学習率それぞれの切り替えを行う(9-12行目)。最後に、パラメータの更新量を計算(13行目)して、パラメータを更新(14行目)し、反復数をインクリメントする(15行目)。

4 内部パラメータの推奨値

DX-NESでは、一反復における解のサンプル数 λ 以外の全ての内部パラメータに推奨値が与えられている³⁾。ここで、内部パラメータの推奨値は、次元数 n が100以下の目的関数で評価回数をできるだけ少なくするように設定されている。そのため、500次元を超

える目的関数に対して、いくつかの内部パラメータに関して妥当な推奨値が与えられていない。

本節では、予備実験の結果を用いて内部パラメータの推奨値に変更を加える。なお、変更を加える前や変更のなかった内部パラメータについては Table 1 に示す。

4.1 distance-weight の拡張係数

distance-weight を加える前後で、一変量標準正規分布の分散が何倍変わるかに着目して拡張係数 α を決定する。distance-weight は各反復におけるサンプル数が有限の場合で定義されていたが、自然にサンプル数が無限大の場合に対して拡張できる。サンプル数を無限大として distance-weight を加えたときの標準正規分布の分散 σ_{dist}^2 は、 $\mathbb{E}[z \exp(\alpha|z|)] = 0$ より、

$$\begin{aligned} \sigma_{\text{dist}}^2 &= \frac{\mathbb{E}[z^2 \exp(\alpha|z|)]}{\mathbb{E}[\exp(\alpha|z|)]} \\ &= \frac{\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} z^2 \exp\left(-\frac{z^2}{2} + \alpha|z|\right) dz}{\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \exp\left(-\frac{z^2}{2} + \alpha|z|\right) dz} \\ &= 1 + \alpha^2 + \sqrt{\frac{2}{\pi}} \frac{\alpha \exp\left(-\frac{\alpha^2}{2}\right)}{1 + \operatorname{erf}\left(\frac{\alpha}{\sqrt{2}}\right)} \end{aligned} \quad (6)$$

となる。このとき、分散 σ_{dist}^2 に $\exp(\alpha^2/2)$ を掛けて得られる関数 $g(\alpha) = \sigma_{\text{dist}}^2 \exp(\alpha^2/2)$ に、予備実験で得られた拡張係数の設定 α_{opt} を入力すると、各次元数 n において $y = g(\alpha_{\text{opt}}) \approx 0.024n$ となった。

以下では、次元数 n における拡張係数の設定として $\alpha = g^{-1}(0.024n)$ を用いる。ここで、逆関数 g^{-1} は閉形式で定義できないので、 $g(\alpha) - 0.024n = 0$ となる α を Newton 法を使って数値的に求めて用いる。

4.2 正規化変換行列の学習率

これまでの知見より正規化変換行列の学習率 η_{B} は、サンプル数 λ に関して単調増加しつつ次元数 n に関して単調減少する関数 $h(n, \lambda)$ とシグモイド状の関数 $\operatorname{sig}(\cdot)$ の合成関数 $\operatorname{sig}(h(n, \lambda))$ を利用すると良い性能を示すことがわかっている。従来の学習率の設定はこの原則に従っており $\lambda \rightarrow \infty$ において妥当であったが、次元数 n が大きくなったときの $\lambda = n$ 付近での値が性能を十分に引き出す設定になっていないことを予備実験により確認した。そのため、正規化変換行列の学習率 η_{B} の推奨値を以下に示す値へと修正する。

$$\begin{aligned} \eta_{\text{B}}^{\text{move}} &= \tanh\left(\frac{1}{6} \frac{\lambda}{0.1n^2 + 2n}\right) + \frac{1.6}{n + 50}, \\ \eta_{\text{B}}^{\text{stag}} &= \tanh\left(\frac{1}{2} \frac{\lambda}{n^2 + 6n}\right). \end{aligned}$$

4.3 進化パスの学習率

高次元において、進化パスの学習率 c_{σ} が過剰に小さくなっていたため以下の式に変更した。

$$c_{\sigma} = \frac{1}{2 \log(n+1)} \frac{\mu_{\text{eff}} + 2.0}{n + \mu_{\text{eff}} + 5.0}.$$

5 実験

本節では、いくつかの基本的な実験設定について述べた後、ベンチマーク関数を用いた3つの実験を行う。

Table 3: The characteristics of benchmark functions.

	ill-conditioned	non-separable	multimodal
Sphere	x	x	x
k-tablet	o	x	x
Ellipsoid	o	x	x
Rosenbrock	o	o	x
Schaffer	x	x	o
Rastrigin	x	x	o

5.1 DX-NES の GPU 実装

目的関数の評価値計算を除いた DX-NES 自体の計算量は $\mathcal{O}(n^3 + \lambda n^2)$ となっており、高次元問題において大量の計算時間が必要となる。ただし、DX-NES における支配的な計算量を持つ部分は行列の積と行列指数関数の計算なので並列計算との親和性は高い。そこで、実験を行うにあたって、ベンチマーク関数の評価値計算を含めた全ての処理を GPU 上で行う。

GPU プログラミング環境としては、東京工業大学の TSUBAME2.0 の G キュー計算ノードと MATLAB 2012a と Parallel Computing Toolbox (CUDA) を利用した。計算ノードのスペックは次のようになる¹²⁾。

- OS : SUSE Linux Enterprise Server 11 SP1
- CPU : Xeon 5670 2.93GHz 4Core/8Thread *3
- MEMORY : 21GB *4
- GPU : Tesla M2050
515Gflops(倍精度), GDDR5 3GB

なお、行列指数関数の実装としては、Pade 近似^{*5}を用いた高速な実装である expm 関数¹¹⁾ が MATLAB に標準で用意されている。しかし、MATLAB 2012a の expm 関数は GPU をサポートしていないので、Algorithm. 1 の 13 行目における $\exp\left(\eta_{\text{B}}^{(g)} \mathbf{G}_{\text{B}}/2\right)$ の計算を GPU に実装する際には別の方法を使う必要がある。

本論文では、 \mathbf{G}_{B} が対称行列であることを利用して、固有値分解 $\eta_{\text{B}}^{(g)} \mathbf{G}_{\text{B}}/2 = \mathbf{V} \mathbf{D} \mathbf{V}^{\text{T}}$ と対角行列 \mathbf{D} の対角成分のみに指数関数を適用して得られる $\hat{\mathbf{D}}$ を用いて、 $\exp\left(\eta_{\text{B}}^{(g)} \mathbf{G}_{\text{B}}/2\right) = \mathbf{V} \hat{\mathbf{D}} \mathbf{V}^{\text{T}}$ とする方法を用いている。このとき、 \mathbf{G}_{B} が数値的に対称にならないことがあるため、実装上は 13 行目の直前で $\mathbf{G}_{\text{B}} \leftarrow (\mathbf{G}_{\text{B}} + \mathbf{G}_{\text{B}}^{\text{T}})/2$ として対称行列化を行っている。

5.2 ベンチマーク関数と初期分布の設定

実験には Table 2 に示す 4 種類の単峰性関数 (Sphere, Rosenbrock, Ellipsoid, k-tablet) と 2 種類の多峰性関数 (Schaffer, Rastrigin) を用いる。これらの関数の中には、Table 3 に示すように悪スケール性、変数間依存性、多峰性を持つ関数が含まれている。Rosenbrock 関数の最適解は $\mathbf{x}_{\text{opt}} = \mathbf{1}$ であり、それ以外の関数では $\mathbf{x}_{\text{opt}} = \mathbf{0}$ である。全ての関数において最適解における評価値は $f(\mathbf{x}_{\text{opt}}) = 0$ となる。なお、k-tablet 関数、Rosenbrock 関数、Schaffer 関数、Rastrigin 関数については、2 次元での関数景観を Fig. 1-4 で示している。

Table 2 に示すように、分布の初期パラメータは Rosenbrock 関数を除いて平均ベクトルが最適解から離れた位置となるように設定した。なお、Rosenbrock

*3物理的には 12Core/24Thread。ジョブスケジューラが制限。

*4物理的には 54GiB。ジョブスケジューラが制限。

*5有理関数を用いて関数を近似する方法であり、Taylor 展開の拡張と見なすことができる。

Table 2: The definition of benchmark functions and initial parameters of distribution. $\mathbf{0}, \mathbf{3}$ and $\mathbf{55}$ are the n -dimensional vectors in which all elements are 0, 3 and 55 respectively. \mathbf{I} is n -dimensional unit matrix.

Unimodal	The definition of objective function	Initial parameters
Sphere	$f(\mathbf{x}) = \sum_{j=1}^n x_j^2$	$\mathbf{m} = \mathbf{3}, \sigma = 2, \mathbf{B} = \mathbf{I}$
k-tablet(k=n/4)	$f(\mathbf{x}) = \sum_{j=1}^k x_j^2 + \sum_{j=k+1}^n (100x_j)^2$	$\mathbf{m} = \mathbf{3}, \sigma = 2, \mathbf{B} = \mathbf{I}$
Ellipsoid	$f(\mathbf{x}) = \sum_{j=1}^n \left(1000 \frac{j-1}{n-1} x_j\right)^2$	$\mathbf{m} = \mathbf{3}, \sigma = 2, \mathbf{B} = \mathbf{I}$
Rosenbrock	$f(\mathbf{x}) = \sum_{j=1}^{n-1} [100(x_{j+1} - x_j^2)^2 + (x_j - 1)^2]$	$\mathbf{m} = \mathbf{0}, \sigma = 2, \mathbf{B} = \mathbf{I}$
Schaffer	$f(\mathbf{x}) = \sum_{j=1}^{n-1} (x_j^2 + x_{j+1}^2)^{0.25} \langle \sin^2 [50(x_j^2 + x_{j+1}^2)^{0.1}] + 1.0 \rangle$	$\mathbf{m} = \mathbf{55}, \sigma = 45, \mathbf{B} = \mathbf{I}$
Rastrigin	$f(\mathbf{x}) = 10n + \sum_{j=1}^n [x_j^2 - 10 \cos(2\pi x_j)]$	$\mathbf{m} = \mathbf{3}, \sigma = 2, \mathbf{B} = \mathbf{I}$

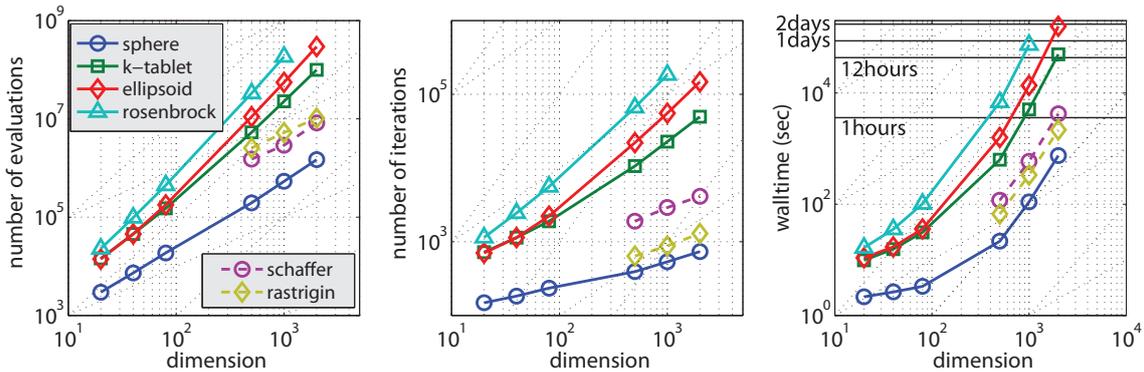


Fig. 6: The average numbers of evaluations (left), iterations (center) and wall-time (right) on unimodal functions for dimension $n = 20 - 2000$ and on multimodal functions for $n = 500 - 2000$. The sample sizes are set to $\lambda = n$ on unimodal function, $\lambda = 800, 1000, 2000$ on schaffer function and $\lambda = 4000, 6000, 8000$ on rastrigin function. Note that the number of iterations equals to the number of evaluations over each sample size.

関数では最適解に比較的近い位置に平均ベクトルを初期化しているが、DX-NESの最適化の過程で分布は原点付近に一度収束し、その後最適解 $\mathbf{x}_{\text{opt}} = \mathbf{1}$ へ向かって移動するため、実質的には平均ベクトルを最適解から離して初期化していることになる。

5.3 終了条件の設定と実行時間の定義

解の評価回数が(一反復における解のサンプル数 λ) \times (目的関数の次元数 n) $\times 10^3$ 回に到達する前に、サンプルされた解の中の最良評価値 f_{best} が $f_{\text{stop}} = 10^{-10}$ 以下になったとき、DX-NESが最適解を発見したとみなし最適化を終了する。

実行時間については、GPUデバイスやアルゴリズムの最初に内部パラメータ等の初期化にかかる時間を除く、メインループが開始されて終了するまでにかかった wall-time を指すこととする。なお、初期化時間は6,7秒程度かかっており、その90%以上はGPUデバイスの初期化に費やされている。

5.4 実験：次元数と評価回数, 反復数, 実行時間の関係

5.4.1 目的と実験計画

本節では、次元数を大きくしていったときに、DX-NESが最適化を完了するまでに必要とする解の評価回数, 反復数と実行時間がどのように推移していくかを調査する。

そのために、4つの単峰性関数において、次元数を $n = 20, 40, 80, 500, 1000, 2000$ としてそれぞれ実験を行う。ただし、Rosenbrockについては計算時間の問題より $n = 2000$ での実験を省略する。また、2つの多峰性関数についても $n = 500, 1000, 2000$ として実験を行う。

サンプル数 λ は単峰性関数で $\lambda = n$ とし、多峰性関数では10試行を行って10試行とも最適解を発見できる設定とした。具体的には、 $n = 500, 1000, 2000$ に対してSchaffer関数では $\lambda = 800, 1000, 2000$ とし、Rastrigin関数では $\lambda = 4000, 6000, 8000$ としている。

5.4.2 結果と考察

単峰性関数における5試行中と多峰性関数における10試行中の平均評価回数, 平均反復数と平均実行時間の次元数に対する推移をFig. 6に示す。

Sphere関数や二つの多峰性関数では、次元数の増加に対して平均評価回数と平均反復数はそれぞれ $\mathcal{O}(n) \sim \mathcal{O}(n^2)$ と $\mathcal{O}(\sqrt{n}) \sim \mathcal{O}(n)$ のオーダーで増加している。その一方で、k-tablet, Ellipsoid, Rosenbrock関数では $\mathcal{O}(n^2) \sim \mathcal{O}(n^3)$ と $\mathcal{O}(n) \sim \mathcal{O}(n^2)$ のオーダーで平均評価回数と平均反復数は増加する。

後者の目的関数においてより多くの平均評価回数や平均反復数が必要となる理由は、後者の目的関数が全て悪スケール性を有しており、効率的に最適解を発見するためには共分散行列の固有値・固有ベクトルを学習する必要がある点にある。例として、Fig. 1に示すk-tablet関数の場合では、二つの固有ベクトルがそれぞれ図の水平方向と垂直方向を向くようにしつつ、水平方向に対応する固有値が垂直方向に対応する固有値よりも相対的に大きくなるように、共分散行列を学習する必要がある。Rosenbrock関数ではFig. 2に示すように評価値の良い領域が湾曲しているため、常に固有ベクトルを学習し続ける必要がある。そのために、次元数に対する増加率はRosenbrock関数が一番大きく

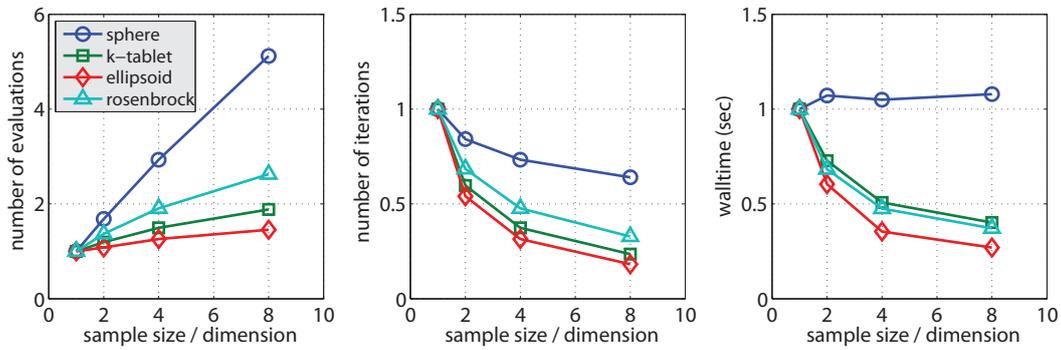


Fig. 7: The ratios of average numbers of evaluations (left), iterations (center) and wall-time (right) compared with those at $\lambda = n$ on the 500-dimensional unimodal functions. The sample sizes are set to $\lambda = 500, 1000, 2000, 4000$. Note that the number of iterations equals to the number of evaluations over each sample size.

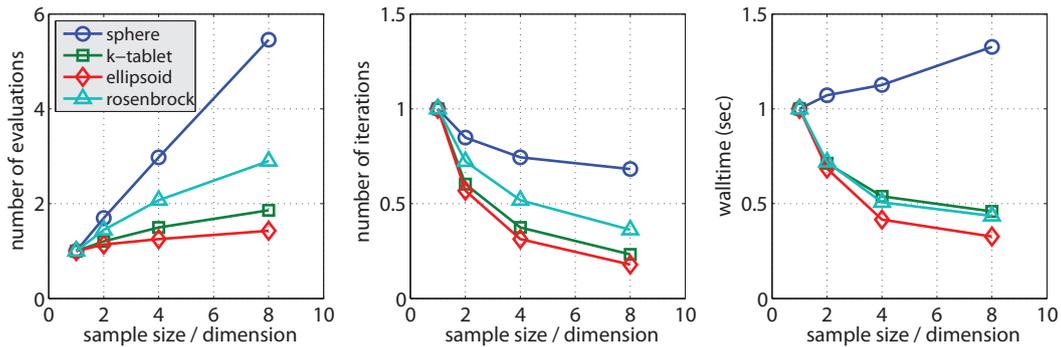


Fig. 8: The ratios of average numbers of evaluations (left), iterations (center) and wall-time (right) compared with those at $\lambda = n$ on the 1000-dimensional unimodal functions. The sample sizes are set to $\lambda = 1000, 2000, 4000, 8000$. Note that the number of iterations equals to the number of evaluations over each sample size.

なっていると考えられる。

実行時間に関しては、一反復の実行時間が次元数が大きくなるにつれて増加するので、平均評価回数と平均反復数よりも高いオーダーで増加していく。

5.5 実験：サンプル数と評価回数、反復数、実行時間の関係

5.5.1 目的と実験計画

本節では、高次元の目的関数において一反復における解のサンプル数を増加させたときの評価回数、反復数、実行時間の推移を調査する。このために、解のサンプル数を $\lambda = n, 2n, 4n, 8n$ と変化させたときの平均評価回数、平均反復数、平均実行時間が、 $\lambda = n$ のときの値に比べてどのように変化するかを実験する。

多峰性関数では局所最適解を避けるために大きなサンプル数を利用することが一般的であり、サンプル数を変化させると局所最適解を発見できなくなることがあるため、この実験は局所最適解のない単峰性関数のみを用いて行う。

5.5.2 結果と考察

次元数を $n = 500, 1000$ としたときの平均評価回数、平均反復数、平均実行時間をそれぞれ Fig. 7, Fig. 8 に示す。ここで、縦軸は $\lambda = n$ で 1 となるように正規化されており、横軸はサンプル数と次元数の比率 λ/n としていることに注意されたい。

平均評価回数は、どの関数においても一反復のサンプル数を増やしていくと大きくなっていくが、Sphere 関数の増加率は他の関数に比べて大きくなっている。この理由は、前節で述べた悪スケール性関数と違って、

Sphere 関数では共分散行列の固有値・固有ベクトルの学習をしなくても最適解を効率的に発見できることにある。いいかえれば、Sphere 関数では共分散行列の学習に関して多少の誤差が許容されるため、多くのサンプル数を用いて共分散行列の更新の精度を向上させるメリットが小さいということになる。

平均反復数については、平均評価回数と同様に目的関数によって差があるものの、サンプル数を k 倍すると概ね $1/\sqrt{k}$ 程度のオーダーで減少していくことが見てとれる。なお、学習率をシグモイド状の関数により決定しているので、サンプル数を無限大にしていくと、学習率の変化率が微小になり、サンプル数を増やしても反復数が減少しなくなっていくと予想される。この予想は実験結果と直感的に一致している。

平均実行時間に関しては、ベンチマーク関数の評価値計算を並列化しているため、一反復あたりの実行時間はサンプル数を増やしてもそれほど変化しない。そのため、平均実行時間は平均反復数と似た減少傾向を示しており、サンプル数を増やすことの利点となっている。ただし、評価値計算の並列化限度に達していたり、サンプル数が目的関数の次元数に比べて極端に大きく反復数が減少しづらくなっているときには、反復数が減少しても計算時間は増えてしまうことに注意が必要である。サンプル数を大きくする別の利点としては、多峰性関数において多くの場合に最適解を発見する確率が上昇することが挙げられる。

以上のことから、評価値計算のコストが比較的軽くかつ並列化することが可能ならば、実行時間の観点から並列化できる範囲で可能な限り一反復におけるサン

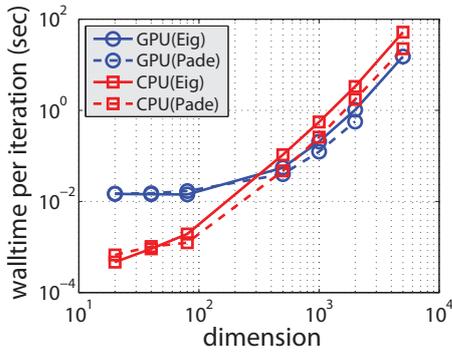


Fig. 9: The average numbers of wall-time per iteration on sphere function for dimension $n = 20 - 2000$. The sample sizes are set to $\lambda = n$.

ブル数を増やすことが望ましいといえる。

5.6 実験：一反復あたりの実行時間

5.6.1 目的と実験計画

本節では、一反復あたりの実行時間が次元数の増加に従って、どのように推移するかを調査する。

評価計算をできる限り除いた部分での比較を行うため、関数評価にかかる時間が最も少ない Sphere 関数において $\lambda = n$ かつ $n = 20, 40, 80, 500, 1000, 2000, 5000$ として 100 回反復を行ったときの一反復あたりの実行時間を計測する。また、比較のために GPU における実行時間だけでなく CPU での実行時間を示す。なお、CPU の実装では MATLAB により自動的にマルチスレッド動作が行われている。

行列指数関数の計算には、固有値分解を利用する方法と MATLAB に標準で用意されている Pade 近似に基づく方法 (expm 関数) を用いる。ただし、expm 関数については GPU がサポートされていないので、コード内の行列演算部分を GPU 向けに書き直して利用する。このとき、Pade 近似を使った GPU の実装の $n = 5000$ における実行時間については GPU のメモリが不足するため省略する。

5.6.2 結果と考察

行列指数関数の計算に固有値分解を用いた実装 (Eig) と Pade 近似を用いた実装 (Pade) における、一反復あたりの実行時間の次元数に対する推移を Fig. 9 に示す。

次元数が $n < 500$ と比較的低いとき、CPU の実装は GPU の実装に比べて最大で 30 ~ 40 倍高速である。第 5.3 節で述べたように GPU デバイスの初期化に 6 秒程度の時間がかかること、GPU へ実装するコストやメモリの制限などを考えると、低次元の問題では CPU での実装を推奨する。

一方で、 $n > 500$ においては GPU による実装が高速であり、CPU と GPU の計算時間の比率は $n > 500$ の範囲で次元数に比例して大きくなっていく。このとき、本論文で実験を行った次元数の範囲で、GPU の実装は CPU の実装に比べて最大で 3 倍程度高速となった。

Table 4 に示すように行列指数関数の計算は Pade 近似を用いる方法を利用したとしても、DX-NES の一反復の計算時間の 60% 程度を占めている。本論文では、GPU へ行列指数関数を実装する際に固有値分解と Pade 近似を用いたが、より高速な GPU 向けの行列指数関

Table 4: The breakdown of wall-time per iteration on 2000-dimensional sphere function with using Pade approximant to the matrix exponential.

	all	expm	sphere	other
CPU	1.723	0.999	0.00854	0.71546
GPU	0.580	0.365	0.00674	0.20826

数の計算法を導入することにより、さらに一反復あたりの実行時間を減らすことも可能である。

6 おわりに

本論文では、black-box な目的関数を最適化する手法として著者らが提案した DX-NES において、いくつかの内部パラメータの推奨値を高次元の目的関数でも効果的な設定となるように変更した。DX-NES を 500 ~ 2000 次元ベンチマーク関数の最適化に適用した結果、DX-NES は全てのベンチマークで最適解の発見に成功し、そのときに必要となる目的関数の評価回数や反復数、実行時間の増加率についていくつかの知見が得られた。

今後の課題は、最適解が制約境界上にあるときに DX-NES の性能が低下する問題が確認されているため、その問題への対処を行うこととなる。

参考文献

- 1) Akimoto, Y., Nagata, Y., Ono, I. and Kobayashi, S.: Bidirectional Relation between CMA Evolution Strategies and Natural Evolution Strategies, Parallel Problem Solving From Nature (PPSN) XI, 154/163 (2011).
- 2) Amari, S.: Natural gradient works efficiently in learning, Neural Computation, volume 10, Issue 2, 251/276 (1998).
- 3) Fukushima, N., Nagata, Y., Kobayashi, S. and Ono, I.: Proposal of Distance-weighted Exponential Natural Evolution Strategies, IEEE Congress on Evolutionary Computing (CEC) 2011, 164/171 (2011).
- 4) Glasmachers, T., Schaul, T., Sun, Y., Wierstra, D. and Schmidhuber, J.: Exponential Natural Evolution Strategies, Genetic and Evolutionary Computation Conference (GECCO) 2010, 393/400 (2010).
- 5) Hansen, N. and Kern, S.: Evaluating the CMA Evolution Strategy on Multimodal Test Functions, Parallel Problem Solving from Nature (PPSN) VIII, 282/291 (2004).
- 6) Hansen, N.: The CMA Evolution Strategy: A Comparing Review, Towards a new evolutionary computation. Advances on estimation of distribution algorithms, 75/102, Springer, (2006).
- 7) Hansen, N.: The CMA Evolution Strategy: A Tutorial, November 24, (2010).
- 8) Sun, Y., Wierstra, D., Schaul, T. and Schmidhuber, J.: Efficient Natural Evolution Strategies, Genetic and Evolutionary Computation Conference (GECCO) 2009, 539/545 (2009).
- 9) Sun, Y., Wierstra, D., Schaul, T. and Schmidhuber, J.: Stochastic Search using the Natural Gradient, In International Conference on Machine Learning (ICML), 1161/1168 (2009).
- 10) Wierstra, D., Schaul, T., Peters, J. and Schmidhuber, J.: Natural Evolution Strategies, IEEE Congress on Evolutionary Computing (CEC) 2008, 3381/3387 (2008).
- 11) MATLAB R2012a Documentation: expm function <http://www.mathworks.co.jp/help/techdoc/ref/expm.html>
- 12) TSUBAME2.0 利用の手引き 2012.08 ver 1.4 <http://tsubame.gsic.titech.ac.jp/user-guides>

粒子群最適化の高次元関数最適化への適用に関する考察

安田大悟 ○ 畠中利治 内種岳詞 (大阪大学)

A Study on Particle Swarm in High Dimension Function Optimization

D. Yasuda, *T. Hatanaka and T. Uchitane (Osaka University)

Abstract— We consider a stagnation in Particle Swarm Optimization due to a premature convergence and a lack of diversity in particle position or velocity. A stagnation is often observed in search process of PSO especially in high dimensional function optimization, even if there are not any local optima. Several approaches have been proposed to prevent or break such stagnation, however almost of them treats a stagnation as an event that particles are trapped in local optima. In this paper, we discuss a stagnation in PSO search process and propose a monitoring method for a stagnation and a simple technique to prevent a stagnation. Numerical examples are presented to show the proposed methods work well.

Key Words: Particle swarm, high dimension function, stagnation, local optima

1 はじめに

粒子群最適化 (PSO (Particle Swarm Optimization))¹⁾ は、その高い探索能力と粒子の比較的簡単な動作式に基づくアルゴリズムであることから注目を集め、さまざまな問題に適用されるようになってきている。GA などの進化計算と同様に、多目的最適化へも展開されている。しかし、一方で、PSO は高い探索能力を有する反面、初期収束に陥りやすいとされ²⁾、再初期化や外乱などにより粒子の速度を回復する方法、動作モデルの係数の与え方³⁾、グローバルベストの共有範囲をトポロジーを用いて制限する方法⁴⁾ および、探索の状況をフィードバックし粒子の動作モデルのパラメータを適応的に与える方法⁵⁾ などが研究されている。このような、局所解に収束することなく大域的最適解を与えるようにアルゴリズムのパラメータや実行条件を設定する問題は、PSO に限らず GA などの進化計算に共通の課題であり、一般に、探索と探索あるいは、大域探索と局所探索のバランスの取り方の問題とされている³⁾。例えば、序盤では大域的に振る舞い有望な領域を発見し、終盤では局所的に大域最適解に収束するようにパラメータを調節することが実際の問題に適用する際に望ましいと考えられ、PSO では、慣性項 w を探索ステップの増加に対して、線形に小さくしていくことで速度ベクトルの収束を序盤では遅く、終盤では速くするなどの方法がよく知られている。

ところで、PSO における初期収束には、局所解に粒子が集まることだけでなく、粒子の速度における多様性が喪失することにより、粒子の位置が収束することなく探索が停滞する場合がある。こういった、探索の停滞する状況をまとめて“stagnation”という。これまでにその解析に関する報告^{6, 7)} がなされているが、PSO を実際に最適化問題に適用する場合には、stagnation を検出することと、その解消法が重要である。そこで、本研究では、局所解が存在しない Sphere 関数を例に、stagnation の例を示すとともに、その検出といくつかの解消法について検討を行う。

以下、本論文では、2章で標準的な PSO のモデルを示し、Sphere 関数を例に stagnation について述べるとともに、要因について議論する。3章で摂動および回転による stagnation の解消法を示し、4章ではその検出

に関して考察する。

2 探索の stagnation

いま、 D を問題の次元、 N を探索に用いる粒子の個数とする。粒子は位置と速度 (探索の各ステップにおける粒子の移動を与えるベクトル) を有し、位置が、最適化問題における説明変数であり、 D 次元のベクトル $\mathbf{x}_i^t = (x_{i,1}^t, x_{i,2}^t, \dots, x_{i,D}^t)^T, i = 1, 2, \dots, N$ と表される。速度は $\mathbf{v}_i^t = (v_{i,1}^t, v_{i,2}^t, \dots, v_{i,D}^t)^T$ で与えられるステップごとの移動量である。ただし、 t は探索のステップ数、 i は粒子番号、 d は次元にそれぞれ対応する添え字であり、 $x_{i,d}^t, v_{i,d}^t$ は、それぞれ、 t ステップ目における粒子 i の位置ベクトルと速度ベクトルの d 次元目の成分を示す。

粒子の要素に対して、一般に、PSO の動作式は以下のように与えられる。

$$v_{i,d}^t = wv_{i,d}^{t-1} + c_1r_1(p_{i,d}^t - x_{i,d}^t) + c_2r_2(g_d^t - x_{i,d}^t) \quad (1)$$

$$x_{i,d}^{t+1} = x_{i,d}^t + v_{i,d}^t \quad (2)$$

ここで、 w, c_1, c_2 は適切に設定すべきパラメータであり、 r_1, r_2 はそれぞれ $[0, 1], i = 1, 2$ の範囲の乱数である。 p_i^t は、 i 番目の粒子が t ステップ目で記憶している粒子自身の履歴における最良の位置、すなわちパーソナルベスト (以下では $pbest$ と表記) を、 g^t は、 t ステップ目におけるすべての $pbest$ のうちで最良の位置すなわち、群全体で共有する最良の位置、グローバルベスト (以下では $gbest$ と表記) である。 $pbest, gbest$ は、それぞれその方向に粒子を向かわせる作用があることからガイドとも呼ばれる。なお、粒子のトポロジーにより、共有の範囲を制限するモデルを導入する場合は、ローカルベストが用いられる。

ここで、以下の議論のため、(1), (2) 式に対応する、粒子ごとのベクトル表記ならびに、群の位置ベクトル、速度ベクトルの行列表記を導入しておく。

$$\mathbf{v}_i^t = w\mathbf{v}_i^{t-1} + \phi_1(\mathbf{p}_i^t - \mathbf{x}_i^t) + \phi_2(\mathbf{g}^t - \mathbf{x}_i^t) \quad (3)$$

$$\mathbf{x}_i^{t+1} = \mathbf{x}_i^t + \mathbf{v}_i^t \quad (4)$$

ここで、 $\mathbf{p}_i^t, \mathbf{g}^t$ は、それぞれ、各粒子の $pbest$ と群の $gbest$

の位置であり,

$$\mathbf{p}_i^t = (p_{i,1}^t, p_{i,2}^t, \dots, p_{i,D}^t)^T$$

$$\mathbf{g}^t = (g_1^t, g_2^t, \dots, g_D^t)^T$$

である. なお, T は, 転置を表す.
また, ガイドの係数は行列で表現され,

$$\phi_i = \begin{pmatrix} \phi_{1,i} & & & & 0 \\ & \ddots & & & \\ & & \phi_{d,i} & & \\ & & & \ddots & \\ 0 & & & & \phi_{n,i} \end{pmatrix} \quad (5)$$

と書ける. ただし, $\phi_{1,i}, \dots, \phi_{n,i}$ は $(0, c_i), i = 1, 2$ の範囲の独立な一様乱数である.

以下の行列表現を導入すると,

$$\mathbf{V}^t = (\mathbf{v}_1^t \mathbf{v}_2^t \dots \mathbf{v}_N^t) \quad (6)$$

$$\mathbf{X}^t = (\mathbf{x}_1^t \mathbf{x}_2^t \dots \mathbf{x}_N^t) \quad (7)$$

群れの位置の更新式は, $\mathbf{X}^{t+1} = \mathbf{X}^t + \mathbf{V}^t$ と書ける.

ここではまず, 評価関数の次元 D に対する解の探索性能の変化をみるため, 局所解の存在しない, Sphere 関数

$$f_1(\mathbf{x}) = \sum_{i=1}^D x_i^2 \quad (8)$$

に対して PSO を適用する. 群れの個体数を 20 とし, その初期位置は $(-50, 50)^d$ の範囲の一様乱数で決定し, 各ステップごとの評価値 $f_1(\mathbf{g}^t)$ の値を調べた. ただし, 本研究では, (1) において $w = 0.7298$ を採用し, $c_1 = c_2 = 1.537$ とした⁸⁾.

目的関数の次元が $D = 10, 30, 60, 100$ の場合に対して, それぞれ 40 回の試行したときの g_{best} に対する目的関数の値の平均を Fig. 1, 2, 3, 4 に示す.

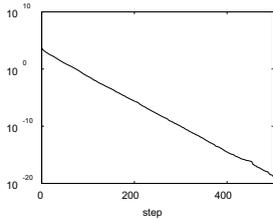


Fig. 1: $D = 10$

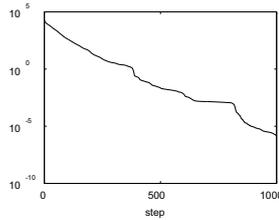


Fig. 2: $D = 30$

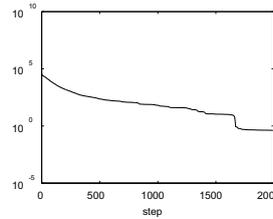


Fig. 3: $D = 60$

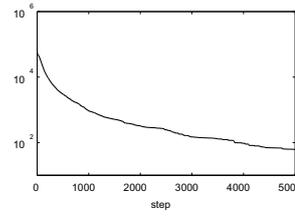


Fig. 4: $D = 100$

$D = 10$ の場合に比べ, $D = 30, 60, 100$ では, 問題の次元が大きくなるにつれて, 何らかの探索の停滞が確

認される. この現象は, 粒子数を問題の次元に応じて増加させれば, 解消できることが確認されるが, 問題の次元数に対して粒子数を単純に多くすることは必ずしも望ましくなく, 実際, PSO では, 20~100 程度の粒子を用いて探索を行うことが多い. また, 粒子数の目安として, Standard PSO2007⁹⁾ では, $10 + 2\sqrt{D}$ が, Standard PSO2011¹⁰⁾ では, ユーザによるとされているものの粒子数 40 が, それぞれ推奨されている.

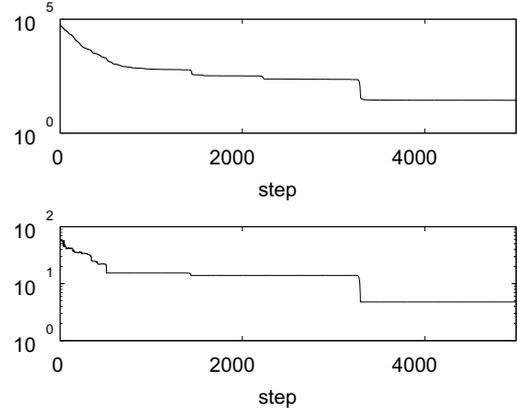


Fig. 5: 目的関数の評価値 $f_1(\mathbf{g}^t)$ (上) と $|g_m|$ (下)

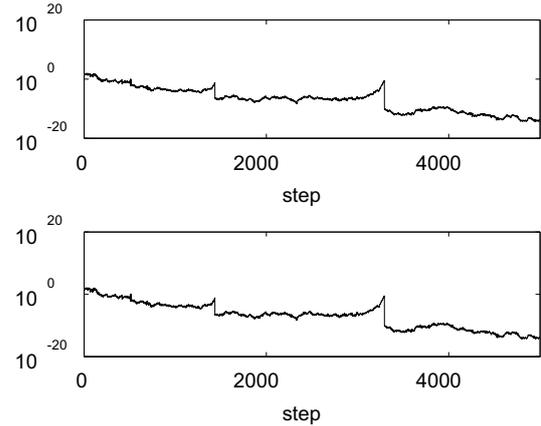


Fig. 6: 第 m 番目の座標の標準偏差 (上) と速度の標準偏差 (下)

ここで, $D = 100$ の場合の, ある一回の試行について各ステップごとに評価値と共に \mathbf{g}^t の成分の中で絶対値が最大の成分 g_m^t , 群全体の第 m 番目の座標情報 p_m, x_m, g_m の標準偏差, v_m の標準偏差を調べた. Fig. 5, 6 にその結果を示す. Fig. 5 から関数の評価値と $|g_m^t|$ の変化はほぼ連動しており, 評価値の更新が停滞している間, $|g_m^t|$ の値の変動も少ない. また, Fig. 6 から, 位置情報の標準偏差と速度の標準偏差の変動にも相関があることがわかる. さらに, Fig. 5 と Fig. 6 を比較すると, 評価値や $|g_m^t|$ が大きく変化するとき, Fig. 6 の標準偏差が大きくなっている. この傾向は, 同条件の下行った他の試行に関しても同様であった.

(1) のように速度 \mathbf{v}_i^t の第 d 成分 v_d は, $\mathbf{x}_i^t, \mathbf{p}_i^t, \mathbf{g}^t$ の第 d 成分のみによって決まる. g_d^t, p_d^t, x_d^t がある値に

収束すると、慣性の大きさを表わす w に従って v_d^t は減衰する。このことから、探索の停滞が停滞している間、群全体の g_d, p_d^t, x_d^t が真の解 $x_d^t = 0$ 以外で収束し、全ての個体の v_d^t のみが小さくなっている可能性が考えられる。

3 Stagnationの解消

3.1 Stagnation

Stagnation を発生させないことを目的とした、PSO の改良が数多く提案されている。代表的なものとして、進化計算の突然変異にならって何かの要素に外乱を導入するもの、交叉のように粒子間で情報を交換するもの、すべての粒子の挙動がそろってしまう状況を避けるために情報の共有範囲を制限するものが挙げられる。

一方で、探索の停滞には、粒子の位置の多様性が失われてしまっている場合と、前章で示したように全空間では多様であってもある部分空間での速度の多様性のみが失われている場合が考えられ、これらは区別して考えるべきであるが、高次元空間におけるこのような粒子の振る舞いをとらえることが困難なこともあり、多くの研究においてこれらは区別されていない。

3.2 速度に摂動与える改善法 11)

直接的に、前章で述べたような特定の次元における速度の喪失に対して、ここでは、 $\mathbf{p}_i^t - \mathbf{x}_i^t, \mathbf{g}^t - \mathbf{x}_i^t$ に直交する方向に速度変化を加えることを検討する。速度変化の向き \mathbf{h} は以下のように求める。

$$\mathbf{h}' = \mathbf{r} - \frac{(\mathbf{g}^t - \mathbf{x}_i^t)(\mathbf{r} \cdot (\mathbf{g}^t - \mathbf{x}_i^t))}{|\mathbf{g}^t - \mathbf{x}_i^t|^2} \quad (9)$$

$$\mathbf{s} = \mathbf{p}_i^t - \mathbf{x}_i^t - \frac{(\mathbf{g}^t - \mathbf{x}_i^t)((\mathbf{p}_i^t - \mathbf{x}_i^t) \cdot (\mathbf{g}^t - \mathbf{x}_i^t))}{|\mathbf{g}^t - \mathbf{x}_i^t|^2} \quad (10)$$

$$\mathbf{h}'' = \mathbf{h}' - \frac{\mathbf{s}(\mathbf{h}' \cdot \mathbf{s})}{|\mathbf{s}|^2} \quad (11)$$

$$\mathbf{h} = \frac{\mathbf{h}''}{|\mathbf{h}''|} \quad (12)$$

ただし、 \mathbf{r} は $(0, 1)^D$ の範囲の一様乱数からサンプリングした D 次元ベクトルである。 $|\mathbf{r}| < |\mathbf{h}''|$ なので、 $|g_d - x_d(t)|, |p_d - x_d(t)|$ が次元 d 以外の成分より小さければ、 h_d の絶対値は他の成分に比べて相対的に大きくなる確率が高くなる。

速度更新式 (3) は以下ようになる。

$$\mathbf{v}_i^t = w\mathbf{v}(t) + \phi_1(\mathbf{p}_i^t - \mathbf{x}_i^t) + \phi_2(\mathbf{g}^t - \mathbf{x}_i^t + \phi_3\mathbf{h}) \quad (13)$$

ここで、変化の大きさ ϕ_3 は、 $(-\alpha(|\mathbf{g}^t - \mathbf{x}_i^t| + |\mathbf{p}_i^t - \mathbf{x}_i^t|), \alpha(|\mathbf{g}^t - \mathbf{x}_i^t| + |\mathbf{p}_i^t - \mathbf{x}_i^t|))$ の範囲の一様乱数とする。以下の実験では、 $\alpha = 0.1$ とした。

3.3 回転による解消法 12)

本研究では、ここで座標軸の回転（選択した2つの軸の間での、速度の交換）による探索の停滞の解消を検討している。図7のように $|g_d - x_d|$ が大きな座標軸 l_p とごく小さな座標軸 l_q を θ だけ回転をさせると、もとの座標系では (3) の第 q 成分は常にごく小さな値しかと

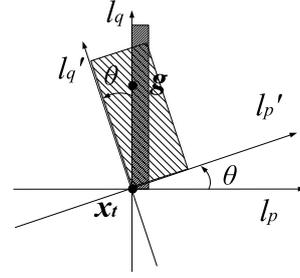


Fig. 7: 座標軸の回転による $\phi_1(g - x_t)$ の分布の変化

らないが、回転後の座標系では大きな値をとることができる。

このような座標軸を回転させる行列 A を導入すると (3) は

$$\mathbf{v}_i^t = w\mathbf{v}(t) + A^{-1}\phi_1A(\mathbf{p}_i^t - \mathbf{x}_i^t) + A^{-1}\phi_2A(\mathbf{g}^t - \mathbf{x}_i^t) \quad (14)$$

となる。ただし座標軸は A^{-1} でステップ毎にもとに戻している。

A は例えば $p = 2, q = 5$, 次元数 $d = 5$ の場合

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \cos \theta & 0 & 0 & -\sin \theta \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & \sin \theta & 0 & 0 & \cos \theta \end{pmatrix} \quad (15)$$

となるような $D \times D$ 行列である。以下の数値実験では、 $\theta = \pi/10$ とした。また、速度の絶対値の平均が大きい座標軸上位 5%、小さい座標軸上位 5% の座標軸の中から、それぞれ、 l_p, l_q をランダムに選ぶものとした。

3.4 数値実験

ここでは、オリジナルの PSO と 2 つの改善法 (a) 速度に摂動を加える方法および、 (b) 回転を作用させる方法を組み入れた PSO の探索性能の比較のため、次の 5 つのベンチマーク関数 (Sphere 関数 $f_1(\cdot)$, DejongF4 関数 $f_2(\cdot)$, Rosenbrock 関数 $f_3(\cdot)$, Griewank 関数 $f_4(\cdot)$ および、 Rastrigin 関数 $f_5(\cdot)$) に、それぞれ適用し PSO の個体の移動の各ステップごとに得られた関数の値を調べた。

Table 1: 初期位置の範囲

関数	範囲
Sphere	$(-50, 50)^d$
DejongF4	$(-20, 20)^d$
Rosenbrock	$(-100, 100)^d$
Griewank	$(-600, 600)^d$
Rastrigin	$(-5.12, 5.12)^d$

$$f_2(\mathbf{x}) = \sum_{i=1}^D ix_i^4 \quad (16)$$

$$f_3(\mathbf{x}) = \sum_{i=1}^{D-1} 100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2 \quad (17)$$

$$f_4(\mathbf{x}) = \frac{1}{4000} \sum_{i=1}^D x_i^2 - \prod_{i=1}^D \cos(x_i/\sqrt{i}) + 1 \quad (18)$$

$$f_5(\mathbf{x}) = \sum_{i=1}^D (x_i^2 + 10 - 10 \cos(2\pi x_i)) \quad (19)$$

関数の次元 D を 10, 30, 60, 100 に設定し, 終了条件はそれぞれ 200step, 500step, 1500step, 3000step, 5000step とした. また, 関数ごとに Table 1 のように設定した範囲の一樣乱数から初期位置を決める.

Sphere 関数に場合の, $d = 10, 30, 60, 100$ としたときのそれぞれの PSO の探索の状態を Fig. 8–11 に示す. また, Rosenbrock 関数に対する結果を, Fig. 12–15 に示す. Table 2 には, 5つの関数すべてに対して, 探索の最終世代に得られた評価値の平均値を示す.

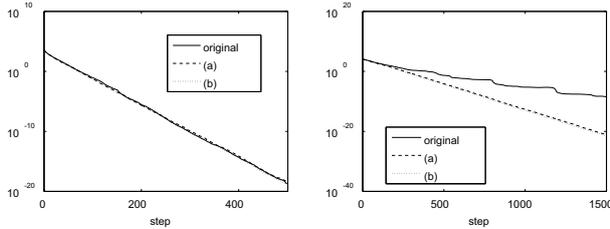


Fig. 8: $D = 10$

Fig. 9: $D = 30$

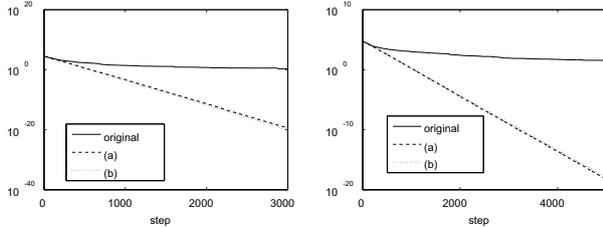


Fig. 10: $D = 60$

Fig. 11: $D = 100$

オリジナルの PSO が, $d = 3, d = 10$ の場合に比べて, 評価値の下がり方に鈍りを見せるのに対して, (a), (b) は, $d = 10$ の場合と同じよう下がり方をする. $d = 60, 100$ では, オリジナルの PSO の収束性能が悪化するが, (a), (b) は次元が低い場合と同じように収束していく.

DejongF4 においても同様の傾向が見られ, Sphere 関数や Dejong F4 関数のような, 変数間に依存関係を持たない単峰性の高次元関数において, 手法 (a), (b) は stagnation の解消に有用であると考えられる. また, このことは, 単峰性の場合の stagnation は, 3.1 で述べたように特定の部分空間における速度の多様性の喪失に

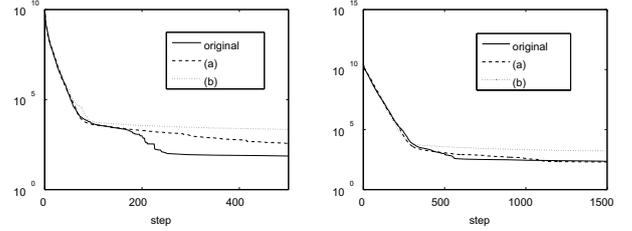


Fig. 12: $D = 10$

Fig. 13: $D = 30$

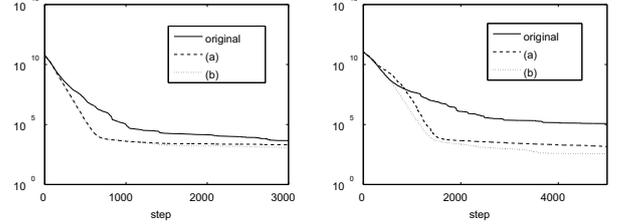


Fig. 14: $D = 60$

Fig. 15: $D = 100$

よるものという考察を支持するものと考えられる.

一方で, 多峰性関数である Griewank 関数や変数間に依存性がある Rosenbrock 関数における探索の停滞においても, Table 2. に示すように $d = 60, 100$ では, (a), (b) がオリジナルの PSO に比べて, 良い結果を残しており, Rastrigin 関数に関しても得られた関数の値に大きな差が見られないことから, 方法 (a), (b) を用いることによって, オリジナルの PSO と比べて大域的な探索能力を大きく失うことなく, 速度更新式 1 の各次元の座標及び速度の独立性を原因とする, 高次元関数における探索の停滞を防ぐことができると考えられる.

4 Stagnation の検出

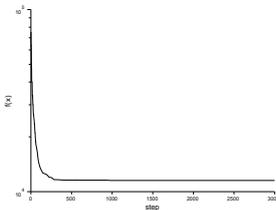
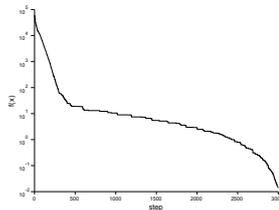
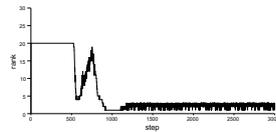
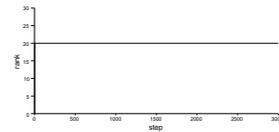
3章の考察と結果から探索の停滞の原因として, 局所解への初期収束だけではなく, 粒子の移動量を与える速度ベクトル \mathbf{v}_i^t の特定の成分における多様性の喪失があると考えられる. そこで, 本研究では, 速度ベクトルが多様性を維持しているか否かを, \mathbf{v}_i^t を列ベクトルとして, 全粒子について並べた行列 $V^t = (\mathbf{v}_1^t \mathbf{v}_2^t \cdots \mathbf{v}_N^t)$ の rank を考える.

V^t の rank は, V^t が $D \times N$ 行列であることから, $N < D$ である場合には, 最大は N である. そこで, $\text{rank} < N$ あるいは, $\text{rank} < K, \exists K < N$ をもって stagnation が起きている可能性があるかと判断することを考える.

Fig. 16, Fig. 17 に $D = 100, N = 20$ の場合に, Sphere 関数に対して, (1), (2) に従う PSO モデルで探索が停滞する場合と, 速度が小さくなったときに小さな外乱を与え探索の停滞を防ぐ処理を加えた場合の, g_{best} の値の推移を, また, Fig. 18, Fig. 19 にそれぞれの場合の rank の推移を示す. Fig. 17 では, 探索が進み $f_1(x)$ が十分に小さくなっており, 速度の行列の rank もフルランクを維持している. 一方で, stagnation がみられる Fig. 16 では, g_{best} の改善が進まなくなるとともに rank が落ちがみられる. 他の試行でも同様の傾向であり, これら結果は, rank による, stagnation の検出の可能性を示している.

Table 2: 最終的に得られた評価値

手法	次元	sphere	DjongF4	Rosenbrock	Griewank	Rastrigin
オリジナル	10	1.9669e - 019	2.4298e - 019	73.029	0.091786	8.8558
(a)		3.324e - 019	2.5029e - 019	364.73	0.12819	10.936
(b)		1.4229e - 019	5.7822e - 020	2196.7	0.13796	9.3226
オリジナル	30	3.8674e - 009	4.7338e - 012	239.62	0.045159	75.517
(a)		7.3776e - 022	6.6795e - 021	238.61	0.018373	85.448
(b)		2.3274e - 022	3.2715e - 022	115.32	0.014633	80.94
オリジナル	60	1.9026	0.63058	4601.6	0.84075	231.21
(a)		3.814e - 020	1.6208e - 018	2057.5	0.013792	246.47
(b)		8.0791e - 021	3.9024e - 019	1195.7	0.0097575	235.28
オリジナル	100	36.642	220.24	1.2053e + 005	2.837	456.28
(a)		5.4657e - 019	1.1666e - 016	1468.2	0.0063291	484.22
(b)		1.1922e - 018	1.5938e - 016	367.69	0.0049239	454.45

Fig. 16: *gbest* の挙動Fig. 17: *gbest* の挙動Fig. 18: *rank* の挙動Fig. 19: *rank* の挙動

なお、講演に際しては、他の結果および、*rank* による検出時のみ多様性の回復を試みる PSO の実行結果を示す予定である。

5 おわりに

本研究では、高次元関数の最適化における PSO の探索の停滞について議論した。PSO における初期収束は、一般に、局所解への収束であると捉えられているが、探索粒子数に比べ問題の変数の数が大きい高次元の最適化問題の場合には、局所最適解を有さない Sphere 関数においても探索が停滞することがある例を示すとともに、このような探索の停滞が速度ベクトルのある一部の要素における速度の多様性の喪失によって引き起こされる可能性を示した。また、そのような探索の停滞を振動および座標軸の回転によって回避する方法を示し、その結果が前述の可能性を支持することを述べた。さらに、数値実験により、探索が停滞するとき、集団の移動量をあたえる速度行列の *rank* が低下することを示し、このような探索の停滞を速度ベクトルから構成する行列の *rank* により検出することを提案した。

なお、本研究は科研費 (C) 22500201 の助成を受けている。ここに謝意を表する。

参考文献

- 1) J. Kennedy and R.C. Eberhart : “Particle Swarm Optimization,” *Proceedings of the 2005 IEEE International Joint Conference on Neural Networks*, 1942/1948 (1995)
- 2) J. Kennedy : “Some Issues and Practices for Particle Swarms,” *Proceedings of the 2007 IEEE Swarm Intelligence Symposium (SIS 2007)*, 162/169 (2007)
- 3) Y. Shi and R. C. Eberhart : “Parameter selection in particle swarm optimization” *Proceedings of the 1998 Annual Conference on Evolutionary Programming VII*, 591/600 (1998)
- 4) J. Kennedy, and R. Mendes : “Population structure and particle swarm performance,” *Proceedings of the IEEE Congress on Evolutionary Computation*, 1671/1676 (2002)
- 5) N. Iwasaki and K. Yasuda : “Adaptive Particle Swarm Optimization via Velocity Feedback,” *IEEE Transactions on Electronics, Information and Systems*, **125**-6, 987/988 (2005)
- 6) M. Clerc : “Stagnation Analysis in Particle Swarm Optimisation or What Happens When Nothing Happens,” *Technical Report CSM-460* (2006)
- 7) M. Jiang, Y. Luo and S. Yang : “Stagnation Analysis in Particle Swarm Optimization,” *Proceedings of the 2007 IEEE Swarm Intelligence Symposium (SIS 2007)*, 92/99 (2007)
- 8) M. Clerc and J. Kennedy : “The particle swarm: Explosion, stability, and convergence in a multidimensional complex space,” *IEEE Transactions on Evolutionary Computation*, **6**-1, 58/73 (2002)
- 9) D. Bratton and J. Kennedy : “Defining a Standard for Particle Swarm Optimization,” *Proceedings of the 2007 IEEE Swarm Intelligence Symposium (SIS 2007)*, 120/127 (2007)
- 10) M. Clerc : “Standard Particle Swarm Optimisation. From 2006 to 2011,” <http://clerc.maurice.free.fr/psol/> (2011)
- 11) 是永剛志 : 高次元関数最適化のための Particle Swarm Optimization, 大阪大学工学部応用自然科学科応用物理学科目卒業論文 (2006)
- 12) T. Korenaga, T. Hatanaka and K. Uosaki : “Performance Improvement of Particle Swarm Optimization for High-dimensional Function Optimization,” *Proceedings of the 2007 IEEE Congress on Evolutionary Computation*, 3288/3293 (2007)

量子粒子群最適化法の基本性能評価

○田附浩一朗 村本憲幸 松井伸之 礪川悌次郎 (兵庫県立大学)

Quantum-Inspired Particle Swarm Optimization and Its Performance

* K. Tazuke, N. Muramoto, N. Matsui and T. Isokawa (University of Hyogo)

Abstract—Quantum-inspired Particle Swarm Optimization (QPSO) is an extension of Particle Swarm Optimization method in which quantum mechanics concept is introduced. In the standard PSO, a state of each particle in multidimensional search space is represented by its velocity and position. By updating the state iteratively in referring to each best position and the global best position of the particle ensemble, we can obtain the optimum solution in the search space. In QPSO, instead of the velocity and position of each particle, a state of particle is represented by a wave function. Hence, the update rule in this method is different from standard ones and it uses a rule based on a potential field. In this paper, we report that QPSO outperforms PSO in solving the various minimum search problems.

Key Words: Quantum-inspired, PSO, QPSO, Minimum search

1 まえがき

Kennedy と Eberhart が粒子群最適化(Particle Swarm Optimization : PSO)¹⁾を提案して以来, PSO は連続問題における高い解探索性能を持つことから, さまざまな性能向上法が研究されてきた. 近年では, PSO に量子力学の概念を組み込んだ, 量子粒子群最適化(Quantum-Inspired PSO : QPSO)²⁾が Sun らによって提案され, PSO との性能比較から, QPSO の有用性が示されている.

本論文では, 最小値探索問題を通して, PSO と QPSO の基本性能を確認し, 精査した結果を報告する.

2 粒子群最適化法(PSO)

PSOは, 位置と速度を持つ粒子群が探索空間である多次元空間を移動しながら最適解を探索する手法であるが, 本節では, QPSOアルゴリズムを提示するために, まずこのPSOの概略を簡単に示しておく.

2.1 PSOの粒子群の構造

PSO の粒子群はいくつかの粒子からなる. n 個の粒子からなる粒子群 S^k は次式で表現される.

$$S^k = (s_1^k, s_2^k, \dots, s_n^k) \quad (1)$$

ここで, k は粒子群状態の更新回数である.

粒子群内の粒子 s_i^k は位置 X_i , 速度 V_i , 粒子自身の過去の位置に対する最適値 P_{besti} (以下, 粒子内最適位置とする), および粒子群の過去の位置に対する最適値 P_{gbest} (以下, 全体最適位置とする) を持っている. 粒子群の中の第 i 番目の粒子の位置 X_i は次式で表現される.

$$X_i = (x_{i1}, x_{i2}, \dots, x_{iD}) \quad (2)$$

同様に, 粒子の速度 V_i , 粒子内最適位置 P_{besti} , および全体最適位置 P_{gbest} はそれぞれ次式で表現される.

$$V_i = (v_{i1}, v_{i2}, \dots, v_{iD}) \quad (3)$$

$$P_{besti} = (p_{i1}, p_{i2}, \dots, p_{iD}) \quad (4)$$

$$P_{gbest} = (p_{g1}, p_{g2}, \dots, p_{gD}) \quad (5)$$

ここで, 式(2)から(5)の D は探索空間の次元を表す.

2.2 PSOのアルゴリズム

PSO は「粒子群の初期化」, 「粒子群の更新」, 「粒子の評価」により実行される. Fig. 1 に PSO のアルゴリズムを示し, 各処理の詳細な説明を以下(i)~(iii)にて行う.

(i) 粒子群の初期化

粒子群の初期化で各粒子の位置と速度を, 乱数を用いて初期化する. また, 各粒子で初期化された位置を粒子内最適位置とする. さらに, 各粒子の位置から算出された適応度の中で最良の適応度である位置を全体最適位置とする.

(ii) 粒子群の更新

粒子の更新は, はじめに, 現在の位置, 速度, 粒子内最適位置, および全体最適位置を用いて次世代の速度を更新する. 速度の更新式は次式で表現される.

$$v_{id}^{k+1} = wv_{id}^k + c_1r_1(p_{id} - x_{id}^k) + c_2r_2(p_{gd} - x_{id}^k) \quad (6)$$

ここで, w は慣性質量, c_1, c_2 は加速係数で任意に設定できるパラメータである. また, r_1, r_2 は(0,1]の一樣乱数, i は粒子の番号, k は粒子状態の更新回数である.

次に, 更新された速度と現在の位置を用いて位置の更新を行う. このときの更新式は次式で与えられる.

$$x_{id}^{k+1} = x_{id}^k + v_{id}^{k+1} \quad (7)$$

(iii) 粒子の評価

粒子の評価では, 各粒子の位置に対して, 適応度関数を適用して算出した適応度 f と, その粒子自身の粒子内最適位置から算出した適応度 f_{best} (以下, 粒子内最適値) を比較し, 適応度が良い方の位置を粒子内最適位置とする. さらに, 各粒子の適応度 f と全体最適位置から算出した適応度 f_{gbest} (以下, 全体最適値) の中で, 最も適応度の良い位置を全体最適位置とする.

3 量子粒子群最適化法(QPSO)

本節では, 量子挙動を組み込んだ粒子群最適化法(Quantum-Inspired PSO : QPSO)についての説明を行う. はじめに, QPSO の概要について説明を行い, 続いて QPSO のアルゴリズムを示す.

```

begin
  k←0
  initialize  $S^0$ 
  while (k < Max_Iteration) do
    begin
      i←0
      while (i < population) do
        begin
          update  $v_i^k$  by equation (6)
          update  $x_i^k$  by equation (7)
          if ( $f_i > f_{best}$ )  $p_{besti} \leftarrow x_i$ 
          if ( $f_i > f_{gbest}$ )  $p_{gbest} \leftarrow x_i$ 
          i←i+1
        end
      end
      k←k+1
    end
  end
end

```

Fig. 1: Procedure of PSO

```

begin
  k←0
  initialize  $S^0$ 
  while (k < Max_Iteration) do
    begin
      i←0
      calculate  $m_{best}$  using equation (10)
      while (i < population) do
        begin
          update  $x_i^k$  by equation (8)
          if ( $f_i > f_{best}$ )  $p_{besti} \leftarrow x_i$ 
          if ( $f_i > f_{gbest}$ )  $p_{gbest} \leftarrow x_i$ 
          i←i+1
        end
      end
      k←k+1
    end
  end
end

```

Fig. 2: Procedure of QPSO

3.1 QPSO の概要

QPSO は PSO の個々の粒子に量子力学の振る舞いを加えた手法である。PSO では粒子の軌道はその粒子の位置と速度によって決定されるが、量子力学では、粒子の位置と速度は不確定性原理により正確に決定できない。

そのため、QPSO における各粒子の状態は、位置と速度の代わりに波動関数 $\psi(x, t)$ によって描かれる。粒子がある位置に出現する確率は、その粒子が存在する空間のポテンシャル場に依存する確率密度関数 $|\psi(x, t)|^2$ で与えられる²⁾。

3.2 QPSO のアルゴリズム

QPSO は「粒子群の初期化」, 「更新」, 「粒子の評価」により実行され、処理の流れは PSO と同じである。しかし、QPSO では粒子の状態が波動関数 $\psi(x, t)$ によって描かれ、粒子の位置は空間のポテンシャル場に依存するため、更新の処理が PSO とは異なる。そこで更新処理の詳細を以下にて説明する。また Fig. 2 に QPSO のアルゴリズムを示す。

(i) 粒子群の更新

QPSO の粒子群の存在するポテンシャル場をデルタ井戸型ポテンシャルとし、これから得られる波動関数の確率密度関数をモンテカルロ手法で模擬すると、更新式は次式で表わされる²⁾。

$$x_{id}^{k+1} = p + \beta \times |m_{best} - x_{id}^k| \times \ln(1/u) \quad \text{if } j \geq 0.5$$

$$x_{id}^{k+1} = p - \beta \times |m_{best} - x_{id}^k| \times \ln(1/u) \quad \text{else} \quad (8)$$

ここで、 p はローカルアトラクタと呼ばれ式(9)で表わされる。また、 m_{best} はすべての粒子の粒子内最適位置の平均で定義され、式(10)で表わされる。

$$p = \frac{r_1 p_{id} + r_2 p_{gd}}{r_1 + r_2} \quad (9)$$

$$m_{best} = \frac{1}{M} \sum_{i=1}^M p_i$$

$$= \left(\frac{1}{M} \sum_{i=1}^M p_{i1}, \frac{1}{M} \sum_{i=1}^M p_{i2}, \dots, \frac{1}{M} \sum_{i=1}^M p_{id} \right) \quad (10)$$

ここで、 u, j, r_1 , および r_2 は(0,1]の乱数、 β は収縮膨張係数と呼ばれるパラメータ、 M は粒子数である。

4 実験結果

本節では、PSO と QPSO の性能評価を行う。4.1 節において、いくつかの評価関数で最小値探索を行い、PSO と QPSO の基本性能評価を行う。

4.1 最小値探索問題

PSO と QPSO において複数の評価関数を用いて最小値を探索することにより、どちらがより良い最適値を得るかを評価する。この実験では、各評価関数において、PSO および QPSO の粒子の位置は関数の定義域内に収まるように設定している。

適用する評価関数とその定義域を Table 1 に示す。なお、各評価関数の最小値は 0 である。また、それぞれの関数を x_1 - x_2 平面に射影したものを Fig. 3 に示す。Table 1 において F_1, F_2, F_4 そして F_5 が多峰性関数であり、それ以外は単峰性関数である。

4.2 実験方法、および実験条件

最小値探索では、PSO および QPSO の粒子数を 20、次元数を 10, 20, 30, 反復回数を 1000, 2000, 3000 と変化させて実験を行う。このとき、PSO の慣性係数 $w=0.9 \sim 0.4$, 加速係数 $c_1=c_2=2.05$ とし、QPSO の収縮膨張係数 $\beta=0.9 \sim 0.6$ とする。ただし w, β は反復ごとに式(11)に従って減少させる。

$$w^{k+1} = w^k - \frac{w_{start} - w_{end}}{Iteration} \quad (11)$$

ここで、 w_{start} は係数の初期値、 w_{end} は係数の終了値、 $Iteration$ は最大反復回数である。

Table 1: Benchmark functions

Benchmark function	Domain
$F_1(x) = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i) + 10)$	[-5.12, 5.12]
$F_2(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	[-600, 600]
$F_3(x) = \sum_{i=1}^n (100(x_{i+1} - x_i)^2 + (x_i - 1)^2)$	[-3.0, 3.0]
$F_4(x) = 418.9829n - \sum_{i=1}^n x_i \sin(\sqrt{ x_i })$	[-500, 500]
$F_5(x) = 20 + e - 20 \exp(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}) - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right)$	[-32, 32]
$F_6(x) = \sum_{i=1}^n \left(\sum_{j=1}^i x_j\right)^2$	[-32, 32]

4.3 実験結果

Table 2 に最小値探索の実験結果を示す。Table 2 において、func は関数、dim は次元数、itera は反復回数、avg、min、sd はそれぞれ 100 回の試行における反復終了時の全体最適値の平均、全体最適値の最小値、そして標準偏差である。

また、PSO と QPSO の反復回数および、次元数の変化に対する収束性の違いをみるために、多峰性関数 F_2 および単峰性関数 F_6 における全体最適値の平均と反復回数の関係をそれぞれ Fig. 4 から Fig. 7 に示す。

Table 2 において avg は、どの評価関数においても PSO より QPSO の方が小さい値となっている。sd も評価関数により差はあるが、PSO より QPSO の方が小さい値をとっている。

また、min においても関数 F_1 を除いて QPSO の方が小さい値をとっているが、関数 F_1 においては次元数と反復回数が増加することで、QPSO の方が良い結果を

得ている。さらに、Fig. 4, 5, 6, 7 から QPSO は少ない反復回数で適応度が収束し、PSO より小さい最小値を取っていることがわかる。

5 結論

本論文では、最小値探索問題により QPSO の基本的な性能を調査した。Table 2 により、関数の形状に関係なく QPSO が PSO よりも小さい最小値を得ることができた。また、最小値探索の精度を調べるために、標準偏差を求めると、PSO より QPSO の方が小さい値となっていた。このことから QPSO の方が、ばらつきが小さく、精度が高い探索ができることが分かった。

さらに、反復回数と適応度の関係を調べることにより、QPSO が PSO より少ない反復回数で探索できることも分かった。この実験により、QPSO の基本性能が PSO より優れていることを確認し得た。

QPSO のもたらすこれらの優れた結果は、QPSO 手法の方が PSO に比して局所解に留まらず大局的な探索を可能としていることによる。QPSO でのこの効果は、粒子の状態が波動関数で与えられ、ポテンシャル場から導かれる確率密度関数で位置が記述されるため粒子位置の確率的広がりをもつことに起因するものと考えられる。

これらの理論的基盤を明らかにするとともに、QPSO が様々な最適化問題において有望であることを示すことが今後の課題である。

謝辞

本研究の一部は科研費(基盤研究(C) 23500286)の助成を受けたものである。

参考文献

- 1) I. Kennedy, R. Eberhart : Particle Swarm Optimization, Pmc. IEEE int. Conf. On Neural Network, 1942/1948 (1995)
- 2) Sun J, Feng B, Xu W : Particle Swarm Optimization with Particles having Quantum Behavior, In Proc. of Congress on Evolutionary Computation, 325 /331 (2004)

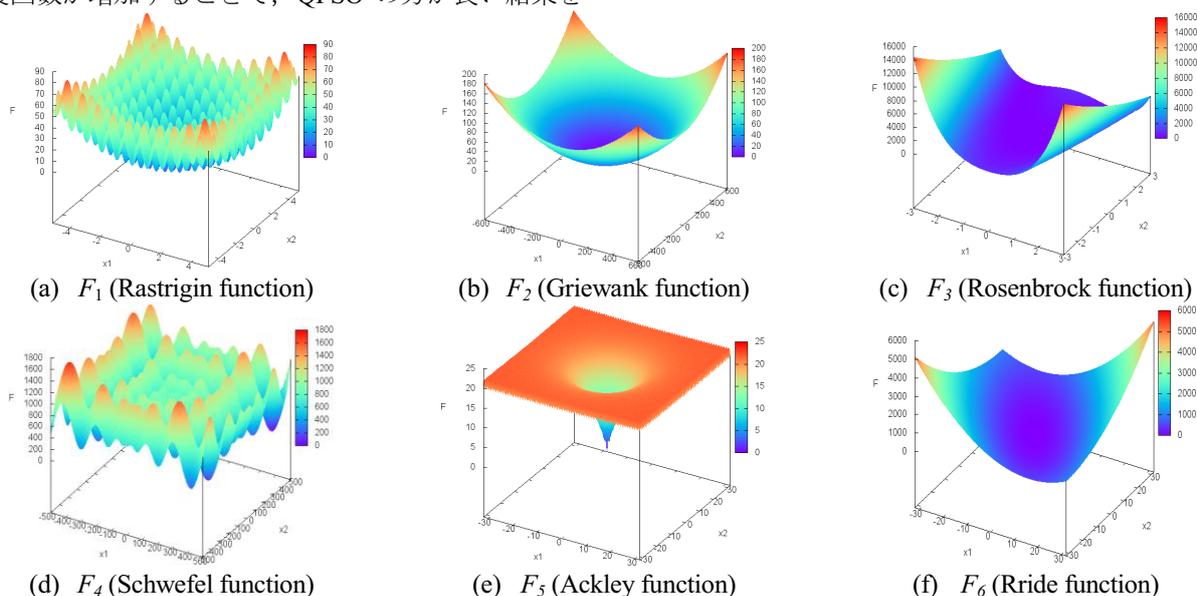


Fig. 3: Benchmark functions

Table 2: Fitness for minimum search problem

func	dim	itera	fitness					
			PSO			QPSO		
			avg	min	sd	avg	min	sd
F_1	10	1000	16.6318	0.9950	14.8661	10.3629	0.9950	5.9665
	20	2000	69.3211	9.5116	31.4196	26.8692	6.5923	18.5478
	30	3000	162.1034	63.7483	44.9918	50.7747	11.9398	21.6905
F_2	10	1000	3.7208	1.9690.E-02	15.4136	8.0063.E-02	1.0265.E-02	4.1311.E-02
	20	2000	51.4918	7.0499.E-14	56.3866	1.8505.E-02	0.0000	2.3688.E-02
	30	3000	139.0126	2.7381.E-11	99.9549	1.1405.E-02	0.0000	1.2737.E-02
F_3	10	1000	4.1635	0.3256	28.2484	0.5922	3.0847.E-04	0.9482
	20	2000	115.5930	0.6632	61.1348	2.1354	2.1415.E-05	1.3153
	30	3000	468.8171	4.6525	801.7907	16.0793	1.4917.E-03	26.1875
F_4	10	1000	927.8362	118.4385	332.5275	686.7333	118.4385	273.0858
	20	2000	2259.0931	833.6198	588.2512	1787.1090	473.7536	478.4056
	30	3000	3879.1649	2262.4652	862.2587	2861.5412	1763.8355	579.9196
F_5	10	1000	2.5332.E-09	3.8426.E-11	3.1941E-09	3.4284.E-15	3.1086.E-15	1.2986E-15
	20	2000	1.1261	1.7336.E-08	5.8066	6.7324.E-15	2.2204.E-15	2.8652E-15
	30	3000	6.5624	1.1165.E-06	8.2366	2.5278.E-14	6.6613.E-15	5.2028E-14
F_6	10	1000	10.2402	7.7810.E-07	87.3406	2.1000.E-06	1.5588.E-09	6.0510.E-06
	20	2000	881.8388	0.8119	662.2683	3.6012.E-02	7.6251.E-04	4.8087.E-02
	30	3000	2678.1946	814.5751	1175.5738	2.8487	0.3325	1.9508

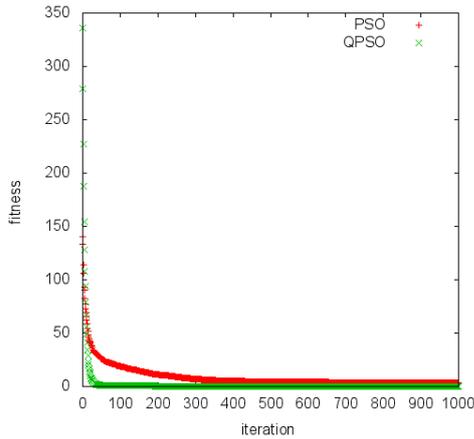


Fig. 4: Fitness values over iteration for minimum search problem (function: F_2 , dimension = 10, iteration = 1000)

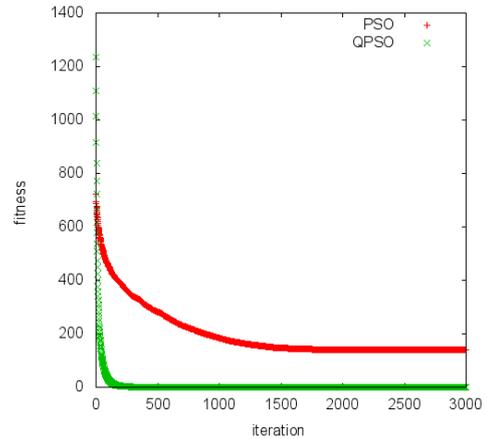


Fig. 5: Fitness values over iteration for minimum search problem (function: F_2 , dimension = 30, iteration = 3000)

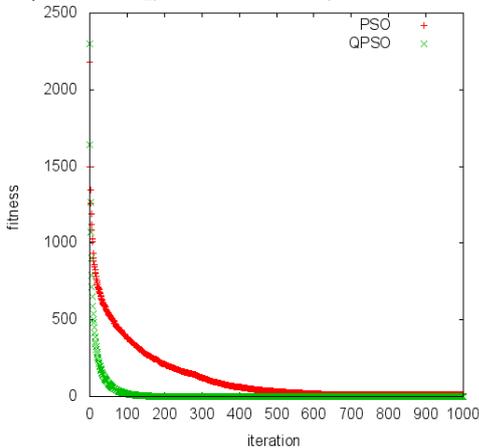


Fig. 6: Fitness values over iteration for minimum search problem (function: F_6 , dimension = 10, iteration = 1000)

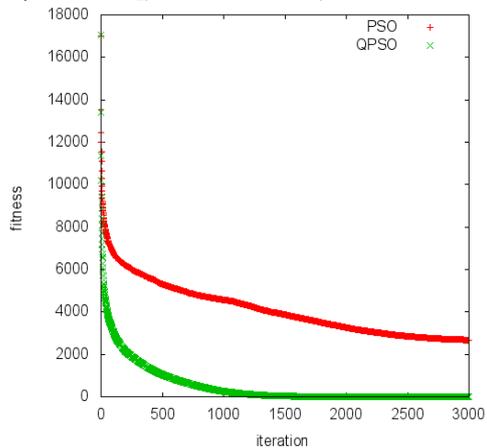


Fig. 7: Fitness values over iteration for minimum search problem (function: F_6 , dimension = 30, iteration = 3000)

ACOを用いたキーボード配置の適応的最適化

○石谷彰斗 榊原一紀 西川郁子 (立命館大学)

Keyboard Arrangement Problem for a Mobile Phone by Ant Colony Optimization

*A. Ishitani, K. Sakakibara and I. Nishikawa (Ritsumeikan University)

Abstract— Keyboard Arrangement Problem (KAP) for a mobile phone, where keyboard is optimally arranged depending on input documents by an ergonomic input load evaluation. Our goal is KAP for a software keyboard for a personalized use. First, we formulate KAP for a hardware keyboard on a mobile, and apply basic ACO. Extension to a software keyboard is straightforward by adding a load term corresponding to a flick motion. Next, ACO is improved by an adaptive pheromone evaporation for the balance between exploration and exploitation during the search. We also compared the obtained arrangements for different categories of the input document.

Key Words: Ant Colony Optimization(ACO), 最適配置, ソフトウェアキーボード, 適応的最適化, 携帯端末

1 はじめに

キーボード配置は、キーボード上の各キーに、文字を割り当てる組み合わせで表される。最適なキーボード配置を求める組合せ最適化問題をキーボード配置問題 (KAP:Keyboard Arrangement Problem) と呼ぶが、これは、複数の文字を複数のキーに、一文字ずつ割り当てる二次割当問題 (QAP:Quadratic Assignment Problem) の一種である。本研究では、KAPによって人間に使いやすく入力効率の高いキーボード配置を求めることを目指す。

従来より、ハードウェアキーボードに対して入力効率の向上を目指す研究が行われてきた。ハードウェア製のキーボードを対象としており、代表的な成果の例として、現在多くのPCで使用されているQWERTYキーボードや、従来型携帯端末であるフィーチャーフォンで使用されているテンキー型のキーボードなどがある¹⁾²⁾。また、人間工学の観点から開発されたものにDvorakキーボードがある。しかし、このキーボードは当時既に普及していたQWERTYキーボードに取って代わることはなかった。その理由として、ユーザが新たにキーボード配置を覚え直す必要性や、ハードウェアであるため生産時点でキーボードへの文字の印字を新配置に切り替える必要性が障害となったことが挙げられる。両者の相乗効果として、ユーザのまだ少ないキーボード配置に合わせた印字のキーボード生産が敬遠されたことが普及の大きな妨げとなった。

これに対して、近年普及したスマートフォンではソフトウェアキーボードが使用されている。このキーボードはタッチパネル上に表示され、タッチ操作によって入力を行うもので、ソフトウェア的な性質からQWERTYキーボードやテンキー型キーボードを模したものに切り替えて使うことができる。そのため、ハードウェアキーボードのように生産時点でキーボード配置に応じた文字を印字する必要がない。我々はそのに着目し、ユーザの要望や用途に柔軟に対応できるソフトウェアキーボードを主な対象として研究を進めている。

本稿ではKAPに対する最適化手法として、QAPでの有効性がしばしば報告されているAnt Colony Optimization(ACO)を用いる。ここで用いるACOのアルゴリズムは、ACOをKAPへ適用したEggerらの

報告²⁾を参考に、Ant Colony System(ACS)³⁾を基に構築した。さらに、アルゴリズムの改良も提案する。ここでは、改良アルゴリズムを巡回セールスマン問題 (TSP:Traveling Salesman Problem) に適用することで性能を評価した。最後に、異なる種類の入力文書に対して得られたキーボード配置を比較、検討した。

2 キーボード配置問題 (KAP)

本研究では、Fig. 1に示す携帯端末で使われているハードウェア (左) およびソフトウェア (右) キーボードに対するKAPを考える。図の左は、フィーチャーフォンで使用されているハードウェアキーボードの例、右はスマートフォンで使用されているソフトウェアキーボードの例である。これらの携帯端末は小さく、入力に使用できるボタンの数が限られている。そのため、テンキー型に数字が割り当てられた各ボタンに、最大五文字が配置される。つまり、一つのボタンが仮想的に複数のキーを持っており、ボタンを連続して押した回数などで入力するキーを判別している。なお以下では、キー入力には一指のみを使うものとする。



Fig. 1: Examples of mobile keyboard with 3x4 buttons.

2.1 ハードウェアキーボードに対するKAP

本節では、フィーチャーフォンなどの携帯端末で使用されているハードウェアキーボードに対するKAPを説明する。ここでは、Fig. 2に示すように、ひらがな、あるいは英語アルファベットの各文字を配置する。図は現在普及している一般的なキーボード配置である。

ひらがなは、図の左に示すように、テンキー0番から9番までの10ボタンに46文字を配置し、英字26文字は、図の右に示すように、テンキー2番から9番までの八ボタンに配置する。各ボタンは仮想的に五つのキーを持っており、各キーに最大一文字が配置され、そ

それぞれボタンの打鍵回数に応じて区別される。図では、同じボタンに割り当てられた五つのキーは、打鍵回数一回から五回までの昇順に、左から右に割り当てられた文字が示されている。例えば、「あ」を入力するにはテンキー1番のボタンを一回、「い」の入力には同じボタンを二回打鍵することを示す。

以上より、ここでのKAPはひらがな46文字を50キーに、あるいは、英字26文字を40キーに配置する組合せ最適化問題となる。なお、「*」はそのキーにどの文字も配置されていないことを示す。

あいうえお	かきくけこ	さしすせそ		abc**	def**
たちつと	なにぬねの	はひふへほ	ghi**	jkl**	mno**
まみむめも	やゆよ**	らりるれる	pqrs*	tuv**	wxyz*
	わをん**				

Fig. 2: Examples of hardware mobile keyboard.

これに対して、以下の二つの制約条件を課した¹⁾。一つ目は「各文字は必ずいずれかのキーに配置される」という制約、二つ目は「各キーには最大一文字が配置される」という制約である。

また、評価関数には人間工学的な評価指標を用い、文書の入力の際に指にかかる負荷を最小にすることによって、入力効率の向上をはかる。ここでは負荷として、入力に要する指の移動距離と打鍵回数を考えるが、それぞれ入力文書における隣接文字対と各文字の出現頻度に依存する。このように評価値が入力文書に依存することから、入力文書の種類に応じていかに最適化されたキーボード配置が得られるかにも着目する。

そこで指への負荷を負荷関数 V で表し、その最小化をはかる。 V は、以下の v_1 と v_2 の和で与える：

v_1 : ある文字を入力した後、次の文字を入力するための指の移動に要する負荷で、キー間の指の移動距離から求める：

$$v_1 = \sum_{i,j,p,q} F_{ij} \cdot D_{pq} \cdot x_{ip} \cdot x_{jq} \quad (1)$$

ここで、 i, j は文字、 p, q はキーを表し、 x_{ip} は文字 i をキー p に配置するか否かを表す二値の決定変数である：

$$x_{ip} = \begin{cases} 1, & \text{if letter } i \text{ is arranged to key } p, \\ 0, & \text{otherwise.} \end{cases}$$

この x_{ip} を用いると、先の二つの制約条件は以下の式で表される：

$$\begin{aligned} \sum_p x_{ip} &= 1 \text{ for } \forall i, \\ \sum_i x_{ip} &\leq 1 \text{ for } \forall p. \end{aligned}$$

また、 F_{ij} は隣接頻度であり、入力文書の中で文字 i の次に文字 j が隣接して出現した回数である。

D_{pq} は指の移動距離であり、キー p を持つボタンからキー q を持つボタンまでの直線距離によって以下で与える：

$$D_{pq} = \sqrt{(x_m - x_n)^2 + (y_m - y_n)^2}$$

m と n はそれぞれキー p と q を持つボタンであり、 x_m と y_m はボタン m の横および縦の座標である。

以上より、Eq. (1) で全てのキーと文字の組み合わせについて、入力文書に対する指の移動負荷を求める。

v_2 : 文字の入力に際して、その文字が配置されたキーの入力に必要なボタンの打鍵回数を負荷とし、以下の式で求める：

$$v_2 = \sum_{i,p} A_i \cdot P_p \cdot x_{ip} \quad (2)$$

ここで、 A_i は文字 i の入力文書中の出現頻度、 P_p はキー p の入力に必要なボタンの打鍵回数である。

負荷関数 V は Eqs. (1),(2) の重みつき線形和により、以下の式で与える：

$$V = \gamma_1 \cdot v_1 + \gamma_2 \cdot v_2 \quad (3)$$

ここで γ_1, γ_2 は、各評価指標に対する重みである。

2.2 ソフトウェアキーボードに対する KAP

次に、スマートフォンなどで使用されているタッチパネル上のソフトウェアキーボードに対する KAP を説明する。前節で示したハードウェアキーボードと同様に、Fig. 1 (右) のソフトウェアキーボードを対象に、ひらがな、あるいは英字の各文字を配置する。Fig. 3 は現在普及している一般的なキーボード配置である。ハードウェアキーボードとの違いは、一つのボタンが仮想的に持っている五つのキーを、一つはタップに、残りの四つはフリックに割り当てる点である。タップはボタンを一回押し操作を指し、ハードウェアキーボードでのボタンの一回打鍵に相当する（よってまた、連続タップによって、同じ文字をその回数分連続入力可能にもなる）。フリックは、ボタンに触れながら、そのまま上下左右のいずれか一方になぞる操作を指し、ハードウェアキーボードで二回～五回の打鍵で入力できるキーを、それぞれ順番に左、上、右、下に割り当てることにした。（なお、ひらがな配置に対しては、キー数を49とする。これはテンキー0番で下フリックをする指がタッチパネル領域からはみ出すため、このボタンのみキー数を四とするからである。）

図中では、タップに相当するキーはボタン中央に、フリックに相当するキーは各方向に該当する位置に示されている。入力の例として、「あ」を入力するにはテンキー1番のボタンをタップし、「い」の入力には同じボタンを左にフリックする。

ソフトウェアキーボードに対する KAP は、ハードウェアキーボードと比べてタップとフリックという相違があるため、評価関数には指の移動負荷である Eq. (1)

う いあえ お	く きかけ こ	す しさせ そ		c b a * *	f e d * *
つ ちたて と	ぬ になね の	ふ ひはへ ほ	i h g * *	l k j * *	o n m * *
む みまめ も	ゆ *や* よ	る りられ る	r q p s *	v u t * *	y x w z *
	ん をわ*				

Fig. 3: Examples of software mobile keyboard.

のみを使用する。また、この式における指の移動距離を以下に変更することで、フリック入力に対応する：

$$D_{pq} = \sqrt{(x_m - x_{m'})^2 + (y_m - y_{m'})^2} + \sqrt{(x_{m'} - x_n)^2 + (y_{m'} - y_n)^2}$$

ただし、 m' はキー p の入力に必要なタップまたはフリック操作によって移動した後の指位置である。ここで、タップの場合は指位置は変わらない。つまり D_{pq} は、キー p を入力するためにタップまたはフリック入力する指の移動距離と、その後、その指位置からキー q のボタンまでの移動距離の和である。

3 ACOによる探索とその改良

本章では、まず ACO を用いた KAP を説明し、探索結果を示す。以下に示すアルゴリズムは、Egger らが KAP の最適化に用いた ACO アルゴリズム²⁾ を参考にしており、Dorigo らが考案した Ant Colony System (ACS)³⁾ を基に、同じく Dorigo らの AS_{Elite}⁴⁾、Stützle らの MAX-MIN AS⁵⁾、Bullnheimer らの AS_{Rank}⁶⁾ を組み合わせたものである。各アルゴリズムを簡単に示すと、ACS は、局所的にフェロモンを更新する。AS_{Elite} は、各世代で評価値の高い解を得たアリにより多くフェロモンを堆積させる。MAX-MIN AS は、フェロモン濃度に上限値、下限値を設ける。そして AS_{Rank} は、フェロモンを更新するアリを解の評価値が高い数匹に限定する。

3.1 KAP への ACO の適用

KAP は、複数の文字を複数のキーへ一文字ずつ配置する QAP である。一般には NP 困難な問題であり、分枝限定法などの厳密解法では膨大な計算時間を要する。これに対して、ACO は QAP での有効性がたびたび報告されており、KAP の先行研究でも Egger らにより ACO の有効性が示されている²⁾。

以下に、ACO の KAP への適用法を示す。まず初めに N 匹のアリが文字の配置されていないキーボードを持つ。それぞれのキーボードには N_i 個のキーがあり、2.1 節で示した制約条件に従って N_c 個の文字を配置することで、各アリは一つの解（キーボード配置）を生成する。配置する文字の順番は、未配置の文字から等確率で選択され、どのキーにどの文字を配置するかは、以下の Eq. (4) の確率 p_{ip} に従って決められる：

$$p_{ip} = \frac{[\tau_{ip}]^\alpha [\eta_{ip}]^\beta}{\sum_{k \in \Omega} [\tau_{ik}]^\alpha [\eta_{ik}]^\beta} \quad (4)$$

p_{ip} は、文字 i のキー p への配置に対するフェロモン濃度 τ_{ip} と、その評価値 η_{ip} によって与えられる。 Ω は、

文字が未配置のキー集合である。つまり、 p_{ip} は文字がまだ配置されていないキー $p \in \Omega$ に、文字 i を配置する確率であり、 τ_{ip} と η_{ip} の各値が大きい程、その確率が上がる。また、 α, β はそれぞれ τ_{ip}, η_{ip} の優先度を定める。 η_{ip} は局所的な評価値であり、その時点で配置されているキーと文字の組み合わせに対してのみ、Eq. (3) で与えられる。

確率 p_{ip} に従って文字 i を配置するキー p を決定すると同時に、フェロモンの局所的な更新を行う。ここではこれを局所気化と呼び、フェロモン濃度 τ_{ip} を、

$$\tau_{ip} := \rho_{\text{local}} \cdot \tau_{ip}$$

に従って更新する。ここで $\rho_{\text{local}} (< 1)$ は局所気化係数であり、フェロモン濃度を減少させる。これによって、あるアリが一度選択したキーと文字の組み合わせは、その後、他のアリに選択されにくくなり、解集合の多様性が増すことになる。

全 N 匹のアリが、全 N_c 文字のキーへの配置を終え、各アリが生成した解を評価関数に従って評価する。その後、文字 i とキー p の組み合わせに対するフェロモン濃度 τ_{ip} を、以下により更新する。

気化：フェロモンの気化として、フェロモン濃度 τ_{ip} を

$$\tau_{ip} := \rho_{\text{global}} \cdot \tau_{ip}$$

に従って減少させる。 $\rho_{\text{global}} (\leq 1)$ は気化係数である。この気化は、前述の局所気化と異なり、全ての文字 i とキー p の組み合わせに対して行われる。

堆積：フェロモンの堆積は、評価値の高い解を生成した上位 M 匹のエリートアリに対してのみ行われる。上位 k 番目のエリートアリが生成したキーボード配置から、キー p とそこに配置された文字 i の組み合わせ全てに対して、そのフェロモン濃度を $\Delta\tau \cdot q(k)$ だけ増加させる。ここで、 $\Delta\tau$ はフェロモン堆積量を与えるパラメータであり、 $q(k)$ はエリートアリの順位に応じた係数で、順位の高いアリほど高い係数とする。

修正：フェロモン値の修正は、フェロモン濃度に上限値、下限値を設けることで生じる。あるフェロモン濃度が上限値または下限値を超えた場合、その濃度値は上限値または下限値に修正される。

上記の更新操作において、気化は、時間経過に伴ってフェロモン濃度を一様に減衰させる。堆積は、評価値の高い解を形成する部分解に高いフェロモン濃度を与えることで、その部分解が確率 (4) により選択されやすくなる。修正は、選択圧が強くなり過ぎるのを抑制する。

以上による、キーボード配置の生成、評価、フェロモンの更新を一つの世代とし、複数世代の繰り返しによって最適解を探索する。

3.2 ACO による計算機実験

前節で説明した ACO による KAP の計算機実験結果を示す。本節で扱う KAP は、ハードウェアキーボードに対して、ひらがな、または英語アルファベットの 2 種類の文字を配置する問題である。実験に用いた入力

文書は、Wall Street Journal 日本語版の記事から取得した。その記事に対して、漢字などを含む全ての文字をひらがな表記に変換した 7,321 文字、そこから更にローマ字表記に変換した 5,656 文字を、それぞれひらがな、または英語アルファベットの配置に使用する入力文書とする。なお、 γ_1, γ_2 値はいずれも 1 とし、評価指標に特に優先度は設定しなかった。

ACO の実験条件は、アリの数 $N = 30$ 、フェロモン初期値 = 0.09、フェロモン堆積量 $\Delta\tau = 0.20$ 、フェロモン気化係数 $\rho_{\text{global}} = 0.95$ 、フェロモン上限値 $\tau_{\text{max}} = 1.00$ 、フェロモン下限値 $\tau_{\text{min}} = 0.09$ 、確率 p_{ip} の設定に用いるパラメータ $\alpha = 3, \beta = 1$ 、エリートアリの数 $M = 4$ 、および、順位 k に対する係数 $q(k)$ は、 $q(1) = 1.00, q(2) = 0.50, q(3) = 0.20, q(4) = 0.10$ とした。キーと文字の数は、ひらがなを配置する場合は $N_t = 50, N_c = 46$ であり、アルファベットを配置する場合は $N_t = 40, N_c = 26$ となる。局所気化係数は $\rho_{\text{local}} = 1.00, 0.99, 0.98, \dots, 0.94$ の各値に設定して、それらの結果を比較する。なお、 $\rho_{\text{local}} = 1$ は局所気化がないことに相当する。探索は、一試行の世代数を 10,000 世代として、五試行行った。

計算結果を示す。Fig. 4 および Fig. 5 は、それぞれひらがなの配置、英語アルファベットの配置問題に対する最良評価値の五試行平均である。図の横軸は世代数、縦軸は最良評価値であり、各世代までに得られた最良解の評価値に対する五試行の平均値を示す。局所気化係数 ρ_{local} の値ごとに結果が示されている。Fig. 6 は、左はひらがな配置問題、右は英語アルファベット配置問題に対して得られた最良解である。いずれも $\rho_{\text{local}} = 0.98$ で得られ、各評価値は 7,488.7 および 3,506.7 であった。

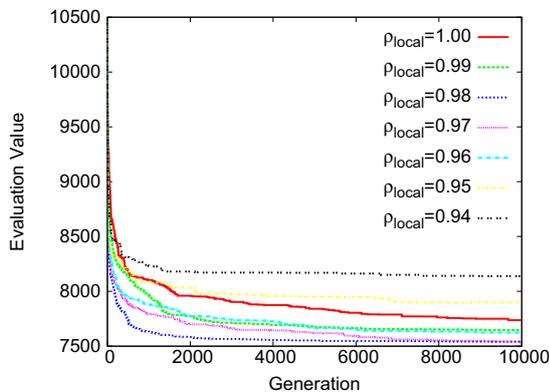


Fig. 4: Search process for Hirakana KAP by ACO.

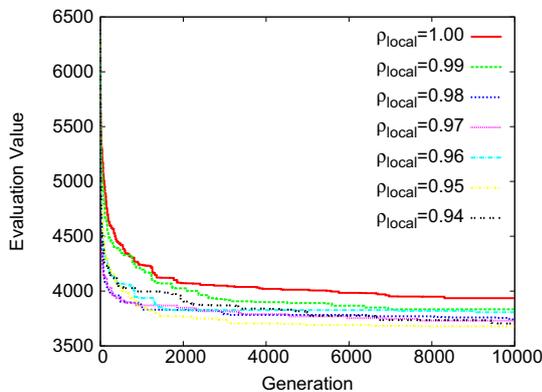


Fig. 5: Search process for English KAP by ACO.

すふるめあ	まれえはら	わむぬねへ		ihwbq	kpgvl
いさてつけ	んたのしか	りちよひゆ	r*mx*	estnc	oajud
きなおせを	とくこにう	もほそみや	*****	y*f**	*****
	る****				

Fig. 6: Optimal arrangements obtained for Hirakana(left) and English(right) KAP.

図より、ひらがな配置に対しては ρ_{local} 値が 1.00 から 0.98 に下がるにつれて、英字配置に対しても同じく 1.00 から 0.97 に下がるにつれて、いずれも探索性能が上がっている。 ρ_{local} 値がさらに低くなると、ひらがな配置では逆に徐々に効率が落ち、 $\rho_{\text{local}} = 0.95, 0.94$ では、局所気化がない ($\rho_{\text{local}} = 1$) 場合より性能が悪化している。英字配置でも、 ρ_{local} を上述の値よりさらに下げても性能の向上は見られない。以上より、探索効率局所気化係数に依存し、係数の適切な設定が必要なが分かる。

3.3 探索における適応的な局所気化

前 3.2 節の結果を受けて、局所気化係数を探索の過程で適応的に変化させるアルゴリズムを提案する。なお、ここでは提案アルゴリズムの性能評価のために、KAP と同じ QAP である TSP にまず適用して、計算機実験を行う。TSP に適用した ACO では、各アリは TSP に対する一つの経路を解として生成する。経路の一番目の都市は等確率で選ばれ、次に巡る都市を Eq. (4) の確率 p_{ip} に従って決定するが、ここで i は現在の都市、 p は次に巡る都市である。また、 τ_{ip} は都市 i から都市 p までの部分経路に対するフェロモン濃度を表し、 η_{ip} は都市 i, p 間の距離、 Ω は未巡回都市の集合である。本節では att48⁷⁾ として知られる 48 都市の TSP を対象とした。これに対する既知の最適評価値は 33,523.71 である。

3.3.1 TSP への ACO の適用

ACO の実験条件は、3.2 節と同じく、アリの数 $N = 30$ 、フェロモン初期値 = 0.09、フェロモン堆積量 $\Delta\tau = 0.20$ 、フェロモン気化係数 $\rho_{\text{global}} = 0.95$ 、フェロモン上限値 $\tau_{\text{max}} = 1.00$ 、フェロモン下限値 $\tau_{\text{min}} = 0.09$ 、確率 p_{ip} の設定に用いるパラメータ $\alpha = 3, \beta = 1$ 、エリートアリの数 $M = 4$ 、および、順位 k に対する係数 $q(k)$ は、 $q(1) = 1.00, q(2) = 0.50, q(3) = 0.20, q(4) = 0.10$ とした。また、局所気化係数は $\rho_{\text{local}} = 1.00, 0.99, 0.98, \dots, 0.94$ の各値に設定して比較する。ここで対象とした 48 都市の TSP では、3.1 節で定義した N_t, N_c はいずれも $N_t = N_c = 48$ となる。なお、本節での実験は一試行の結果である。

計算機実験の結果として、Fig. 7 に各世代の最良評価値を示す。局所気化係数 ρ_{local} を 1.00 から下げても、0.96 までは探索はほぼ安定し収束している。そのため、各世代でそれまでに得られた最良解やその近傍解が高確率で生成される必要がある。局所気化は、逆にそれを抑えることで多様性を増大し、大域的な最良解に導くことを目指すものである。よって、局所気化が強く働き過ぎると、 $\rho_{\text{local}} = 0.95$ の結果に見られる

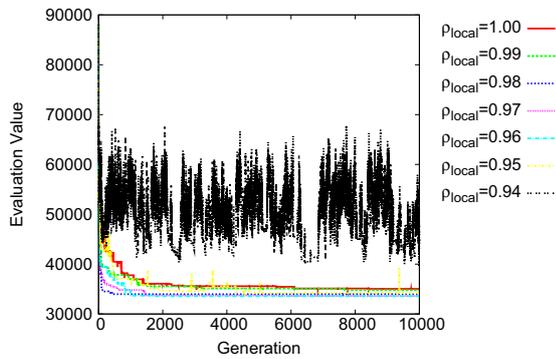


Fig. 7: Search process for TSP by ACO.

ように、世代交代により最良解が悪化することが目立ち始める。さらに $\rho_{local} = 0.94$ では、もはや安定した探索が行えていない。そこで以下では、探索の安定性と多様性の維持を適切にバランスさせるよう、局所気化係数を適応的に変化させるアルゴリズムを提案する。

3.3.2 適応的な局所気化による選択圧の制御

得られた中で最良の暫定最適解を保持しつつ、そこへの過度な集中の抑制を目的として、暫定最適解への収束度合いを制御すべく局所気化係数を変動させる。まず、現世代までに得られた最良解を暫定最適解とし、その解を生成したアリが何匹いるかを、暫定最適解への収束度を表す指標とする。その匹数 E と、フェロモン堆積の対象として優遇するエリートアリの匹数 M を比較し、以下の二状態に分類する：

状態 (A) $E > M$: 暫定最適解に集中しており、選択圧が高く、多様性が低い。

状態 (B) $E < M$: 暫定最適解への集中が少なく、選択圧が低く、多様性が高い。

状態 (A) のとき局所気化を強め (係数 ρ_{local} を減少させる)、状態 (B) のとき弱める (係数を増加する)。ここでは増減幅を各世代 0.001 とした。また同時にエリート保存 (暫定最適解アリ一匹の保存) を組み込んだ。

Fig. 8 に適応的な局所気化の実験結果を示す。図には、比較のために Fig. 7 で最も性能が良かった局所気化係数 $\rho_{local} = 0.96$ の結果も合わせて示した。適応的に気化係数を変えることで、さらに有効な探索が行えている。自動的にパラメータ値を獲得しつつ、より良い解が得られたことが分かる。

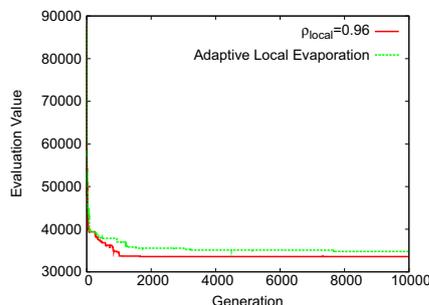


Fig. 8: Search process for TSP by ACO with adaptive local evaporation.

4 文書の種類に応じたキーボード配置

本章では、入力文書の特徴に応じたキーボード配置を検討する。入力文書として、日本語文 (ローマ字入

力)、英文、C++プログラムのソースコードの三つを用い、ソフトウェアキーボードへの英語アルファベットの最適配置として得られるキーボード配置を比較した。各一例ながら、以下に示す結果より、KAPによって、入力文書に応じたキーボード配置が見られた。

ACOの実験条件は、アリの数 $N = 30$ 、フェロモン初期値 $= 0.09$ 、フェロモン堆積量 $\Delta\tau = 0.20$ 、フェロモン気化係数 $\rho_{global} = 0.95$ 、フェロモン上限値 $\tau_{max} = 1.00$ 、フェロモン下限値 $\tau_{min} = 0.09$ 、確率 p_{ip} の設定に用いるパラメータ $\alpha = 3, \beta = 1$ 、エリートアリの数 $M = 4$ 、および、順位 k に対する係数 $q(k)$ は、 $q(1) = 1.00, q(2) = 0.50, q(3) = 0.20, q(4) = 0.10$ とした。キーと文字の数は、ソフトウェアキーボード上の $N_t = 40$ のキーに対して、英語アルファベット $N_c = 26$ 文字を配置する。なお、探索の世代数は 10,000 回とした。またここでは、3.3.2 節で導入した適応的な局所気化アルゴリズムを用いた。

4.1 ローマ字文書 (ローマ字綴りの日本語文書)

日本語文書のローマ字入力に対する KAP を考える。入力に用いたのは、Wall Street Journal 日本語版の記事をローマ字文に変換した英字 5,656 文字からなる文書である。このローマ字文書で出現頻度の高い文字は、母音を主として「a(738 回), i(632 回), o(624 回), u(534 回), n(485 回)」であり、頻度の高い隣接文字対は「o \rightarrow u(136 回), n \rightarrow o(127 回), k \rightarrow a(114 回), i \rightarrow n(119 回), a \rightarrow i(110 回)」であった。

Fig. 9 に ACO で得られた最良配置を示す。頻度が高い上記五つの隣接文字対に着目する。まず頻度が最大の「o \rightarrow u」は、「o」のタップの後、同一ボタンのフリックキーに「u」が配置された。これにより、指の移動にかかる負荷は 0 になる。「i \rightarrow n」は、「i」がフリックキーに配置され、フリック後に指が来るボタンに「n」が配置されている、また「a \rightarrow i」は、「a」のタップの後、指位置を上へ 1 移動させたボタンに「i」が配置されている。これらの場合、指の移動にかかる負荷は 1 である。このように、頻度の高い隣接文字対は入力しやすいキーに配置された。次に、残る二つの高頻度文字対「n \rightarrow o, k \rightarrow a」に関しては、他の文字との隣接頻度も合わせて見てみる。すると「n」は、「n \rightarrow i(99 回), n \rightarrow s(60 回)」であるため、「s, i」が配置されているテンキー 2 番のボタンへのフリックキーに、「k」は「k \rightarrow u(70 回), k \rightarrow i(68 回), k \rightarrow o(62 回)」であるため、「u, i, o」のどれにも指の移動負荷 1 で入力できるボタンに配置されている。このように、高出現頻度の子音文字に対して、高頻度で後続する母音文字が複数ある場合、それらを総合的に考慮した配置が得られていた。これは、五つの母音を用いるローマ字 (ローマ字綴りの日本語) に特徴的な結果だと思われる。

4.2 英語文書

英語文書の入力に対する KAP を考える。入力に用いたのは、New York Times の記事から取得した英字 5,656 文字からなる文書である。この文書で出現頻度の高い文字は「e(686 回), t(484 回), a(465 回), i(436 回), n(413 回)」であり、頻度の高い隣接文字対は「t \rightarrow h(141 回), h \rightarrow e(117 回), i \rightarrow n(107 回), e \rightarrow r(97 回), a \rightarrow n(88 回)」であった。

Fig. 10 に ACO で得られた最良配置を示す。頻度の

	*	q
	l s *	f k *
	i	y
*	n	z
* g h	m a r	u o *
*	d	*
j	e	w
* p x	c t *	* b v
*	*	*

Fig. 9: Optimal arrangement obtained by proposed ACO for a Romanized Japanese document.

高い隣接文字対「t → h, i → n, e → r」に対しては、「t, i, e」のタップの後、同一ボタンのフリックキーに「h, n, r」がそれぞれ配置された。「h → e, a → n」では、「h, a」がフリックキーに配置され、フリック後に指が来るボタンに「e, n」が配置されている。ここで特徴的な点は、頻度が高い隣接文字がさらに「t → h → e」と三つ連なっており、キーボード配置においても「t, h, e」を続けて入力しやすくなっていることである。これは入力文書において、「the」や、その綴りを含む単語「they, there」が多く使われていたため、このような結果が得られたと考えられる。

	*	*
	* u *	q b *
	g	j
*	d	*
* t h	s e r	p l *
x	a	y
z	n	w
* c k	v i f	* o *
*	*	*

Fig. 10: Optimal arrangement obtained by proposed ACO for an English document.

4.3 C++ソースコード

プログラムのソースコードの入力に対する KAP を考える。入力に用いた文書は、C++で作成した ACO による最適化探索プログラムで、英字 5,656 文字からなるソースコードである。出現頻度の高い文字は「e(503 回), t(462 回), i(365 回), a(341 回), o(331 回)」であり、頻度の高い隣接文字対は「i → n(148 回), n → t(135 回), l → e(92 回), o → u(84 回), e → y(81 回)」であった。

Fig. 11 に ACO で得られた最良配置を示す。出現頻度の高い文字は、全てタップキーに配置された。また、頻度の高い隣接文字対「i → n, o → u, e → y」に対しては、「i, o, e」のタップの後、同一ボタンのフリックキーに「n, u, y」がそれぞれ配置された。「n → t, l → e」では、「n, l」がフリックキーに配置され、フリック後に指が来るボタンに「t, e」が配置されている。ここで特徴的な点は、頻度が高い隣接文字がさらに「i → n → t」と三つ連なる頻度も非常に高く、キーボード配置においても「i, n, t」を続けて入力しやすくなっていることである。C++のソースコードでは、変数型である「int」の他、「print, fprint」などが多く使われていたた

め、このような結果が得られたと考えられる。

	*	*
	x k *	j b *
	z	p
*	m	*
* a l	h e y	r o *
*	d	u
g	c	f
* v w	* i n	s t *
*	q	*

Fig. 11: Optimal arrangement obtained by proposed ACO for an C++ source code.

5 おわりに

本稿では、まず、キーボード配置に対する研究開発の現状と背景を示し、ハードウェアおよびソフトウェアキーボードに対する KAP を説明した。次に KAP への ACO の適用を示し、ハードウェアキーボードに対して、計算機実験により最適配置を求めた。その中で ACO における局所気化の問題点を示し、その改善法として適応的に局所気化を変化させるアルゴリズムを提案した。それをベンチマーク問題としての TSP へ適用し、計算機実験により有効性を示した。さらに、入力文書に応じて最適配置を求める KAP の特徴を活かし、三種類の入力文書に対してそれぞれの特徴を反映した英字配置が得られることを確認した。

今後の課題として、ACO の性能向上に向けて、さらなる改良や、並列計算の実装を試みたい。また、ソフトウェアキーボードに対して、個人の間人間的な特性や嗜好、用途に応じたキーボード配置を検討したい。さらに、得られたキーボード配置の現実的な性能を、実験機を試作し被験者実験によって検証したい。

参考文献

- 1) Mauro Dell'Amico, José Carlos Díaz Díaz, Manuel Ioria, Roberto Montanaria : The single-finger keyboard layout problem, *Computer & Operations Research*, **36**, 3002/3012 (2009)
- 2) Jan Eggers, Dominique Feillet, Steffen Kehl, Marc Oliver Wagner, Bernard Yannou : Optimization of the keyboard arrangement problem using an Ant Colony algorithm, *European Journal of Operational Research*, **48**, 672/686 (2003)
- 3) Marco Dorigo, Luca Maria Gambardella : Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem, *The IEEE Transactions on Evolutionary Computation*, **1-1** (1997)
- 4) M.Dorigo, V. Maniezzo and A. Colorni : Ant System: Optimization by a colony of cooperating agents, *IEEE Transactions on Systems, Man, and Cybernetics-Part B*, **26-1**, 29/41 (1996)
- 5) T. Stützle and H. H. Hoos : MAX-MIN Ant System, *Future Generation Computer Systems*, **16-8**, 889/914 (2000)
- 6) B. Bullnheimer, R. F. Hartl and C. Strauss : A New Rank Based Version of the Ant System: A Computational Study, *Central European Journal for Operations Research and Economics*, **7-1**, 25/38 (1999)
- 7) <http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/>

任意軌道追従のための 画像に基づくファジィ制御器のGAによる最適化

○加藤達也 渡辺桂吾 前山祥一 (岡山大学)

Optimization of an Image-based Fuzzy Controller for Arbitrary Trajectory Tracking by a Genetic Algorithm

*T. Kato, K. Watanabe, and S. Maeyama (Okayama University)

Abstract— This paper proposes an image-based fuzzy controller for arbitrary trajectory tracking of a mobile robot, which determines control inputs from camera images captured by a camera mounted on the robot. Especially, variations of a target trajectory on an image plane are referred as state variables to improve the tracking performance. Then, a genetic algorithm is used to optimize the present fuzzy controller because the tuning of the controller becomes complicated due to the added state variables. The improvement of the tracking performance is confirmed by evaluating some simulation experiments.

Key Words: Image-based visual servo control, Fuzzy control, Genetic algorithm, Trajectory tracking

1 はじめに

従来の移動ロボットの制御の多くはあらゆる自己位置推定法によって求められたロボットの自己位置に基づいてそれらを制御してきた。一方で普段、人間は自分の正確な位置など気にすることなく、主に目で見えた情報を基に行動を決定している。視覚サーボは人間のように視覚情報に基づいてロボットの制御を行う手法であり、広く研究されている。その視覚サーボは位置に基づく視覚サーボ (Position-Based Visual Servo control, PBVS) と画像に基づく視覚サーボ (Image-Based Visual Servo control, IBVS) の2つの手法に大きく分けられる^{1, 2)}。

PBVSは三次元環境モデルなどとカメラ画像のマッチングによって推定されたロボットや物体の位置を基にロボットを制御する。位置を直接扱うため、多くの従来の制御理論を容易に適用可能であることが挙げられる。しかしながら、事前知識の量や三次元のマッチングの計算量が問題となる。Cherubiniら³⁾はカメラを搭載した非ホロノミック移動ロボットによる軌道追従を行った。

IBVSはロボットの位置を参照するのではなく、画像平面上の特徴量 (特徴点の座標や対象物の面積など) を基にロボットを制御する。Espiau⁴⁾はIBVSがPBVSに比べてロボットのモデルやカメラキャリブレーションの誤差に対してロバストであることを確かめた。IBVSの安定性についてはMalisら⁵⁾が議論している。

人間の行動は視覚情報に深く依存しているが、位置を気にするのではなく見た物そのものに反応していると思われる。これは「見え方」を制御するというIBVSのコンセプトと似ており、この手法は人間的な情報処理を行っていると考えられる。本研究の大きな目的は、IBVSを用いてロボットを制御することで、人間のような多様性をロボットに与えることである。我々はこれまで、移動ロボットの直線軌道追従を行うための画像に基づくファジィ制御器を提案し⁶⁾、これを応用した任意軌道追従法を提案してきた⁷⁾。しかし、この任意軌道追従法では画像座標上の目標軌道の変化量を参照していなかった。そこで、本稿では、その目標軌道の変化量を状態変数として参照するコントローラを設計す

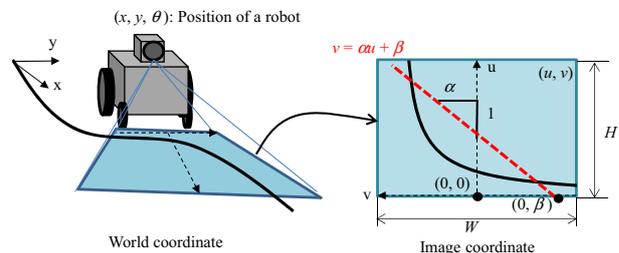


Fig. 1: World coordinate and image coordinate

ることで、さらなる追従性能の向上を図る。しかしながら、参照する状態変数の増加によりファジィ制御器の制御規則の数も増加するため、手作業による制御器の調整が困難となる。そこで、実数値GA⁸⁾によってファジィ制御器を最適化することとする。

本稿では、2章にて問題設定を述べ、3章で任意軌道追従を行うための画像に基づくファジィ制御器について説明する。4章ではそのファジィ制御器を実数値GAによって最適化する手法について述べる。5章では提案した最適化手法と、最適化されたファジィ制御器の性能をシミュレーションによって評価する。

2 問題設定

制御の目的は、カメラを搭載したロボットがIBVSによって地面に描かれた任意曲線軌道を追従することとする。Fig. 1は環境と座標の定義を表している。世界座標はx-y座標系とし、カメラ画像の座標はFig. 1に示すようにu-v座標系と定義する。また、画像のサイズは横幅Wで縦幅Hとする。制御対象は自動車のような前輪操舵型の4輪移動ロボットとする。ロボットの状態は位置と姿勢 (x, y, θ) で表現するものとし、前進速度を s 、前輪の操舵角を ϕ とする。画像上の目標軌道は最小二乗法によって直線 $v = \alpha u + \beta$ へと近似される。この直線のパラメータである α と β は次式よ

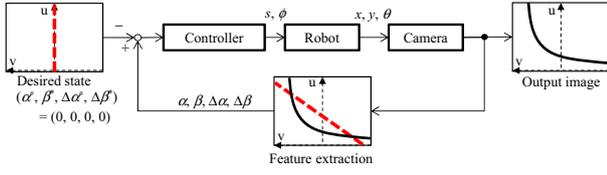


Fig. 2: Block diagram of the arbitrary trajectory tracking

り求められる。

$$\alpha = \frac{n \sum_{k=1}^n u_k v_k - \sum_{k=1}^n u_k \sum_{k=1}^n v_k}{n \sum_{k=1}^n u_k^2 - \left(\sum_{k=1}^n u_k \right)^2}$$

$$\beta = \frac{\sum_{k=1}^n u_k^2 \sum_{k=1}^n v_k - \sum_{k=1}^n u_k v_k \sum_{k=1}^n v_k}{n \sum_{k=1}^n u_k^2 - \left(\sum_{k=1}^n u_k \right)^2} \quad (1)$$

ただし、 (u_k, v_k) は画像上の目標軌道を含んでいる k 番目の画素の座標であり、 n はその総数を表している。また、これらのパラメータの変化量 $\Delta\alpha$ および $\Delta\beta$ は、現在のフレームより抽出したパラメータ値 α_t, β_t と 1 つ前のフレームより抽出した値 $\alpha_{t-1}, \beta_{t-1}$ との差分値とし、以下の式によって求めるものとする。

$$\begin{aligned} \Delta\alpha &= \alpha_t - \alpha_{t-1} \\ \Delta\beta &= \beta_t - \beta_{t-1} \end{aligned} \quad (2)$$

以降、これらのパラメータ $\alpha, \beta, \Delta\alpha, \Delta\beta$ を状態変数として扱うこととし、軌道追従を行うための制御器はこれらの状態変数より制御入力 s, ϕ を決定する。Fig. 2 は提案する任意軌道追従制御のブロック線図を示している。このブロック線図より、制御器がロボットの自己位置を参照することなく、画像座標上の情報だけを基に制御入力を決定していることが確かにわかる。

3 画像に基づくファジィ制御器

制御変数 $\alpha, \beta, \Delta\alpha, \Delta\beta$ と制御入力 s, ϕ との関係は画像の奥行きに依存しているためこれを正確にモデル化することは容易ではない。そこで、モデルフリーのファジィ制御器を用いることとした。ファジィ制御器の設計は主にメンバシップ関数の形状の調整と制御規則を制定することである。Fig. 3 にメンバシップ関数の形状を示す。ラベル N, ZO, P はそれぞれ Negative, Zero, Positive を意味しており、B, S はそれぞれ Big, Small を意味している。図中の $a_\alpha, a_\beta, a_{\Delta\alpha}, a_{\Delta\beta}, a_{vS}, a_{vB}, a_{\phi S}, a_{\phi B}$ はそれぞれのメンバシップ関数の広がりを表す変数である。なお、 β および $\Delta\beta$ に関するメンバシップ関数については、それぞれ画像の横幅 W によって正規化した値 $2\beta/W$ および $2\Delta\beta/W$ を用いている。

制御規則は 4 つの状態変数がそれぞれ 3 つのラベルを有することから 3^4 通り考えられ、それらは以下のように IF-THEN ルールとして定められる。

$$\begin{aligned} R_0 : \text{IF } \alpha = N, \beta = N, \Delta\alpha = N, \Delta\beta = N \\ \text{THEN } s = s_0, \phi = \phi_0 \end{aligned}$$

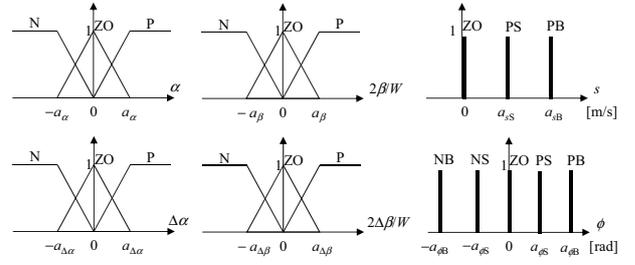


Fig. 3: Membership functions

$$\begin{aligned} R_{80} : \text{IF } \alpha = P, \beta = P, \Delta\alpha = P, \Delta\beta = P \\ \text{THEN } s = s_{80}, \phi = \phi_{80} \end{aligned}$$

ただし、規則の後件部の制御入力 s_i および ϕ_i ($i = 0 \dots 80$) にはそれぞれのメンバシップ関数が有するラベルが入るものとする。

制御規則の適合度 h_{Ri} ($i = 0 \dots 80$) は、その規則に含まれるそれぞれの状態変数のラベルに対応するメンバシップ関数値の積として次式より求められる。

$$\begin{aligned} h_{R0} &= \mu_{\alpha,N}(\alpha) \cdot \mu_{\beta,N}(\beta) \cdot \mu_{\Delta\alpha,N}(\Delta\alpha) \cdot \mu_{\Delta\beta,N}(\Delta\beta) \\ &\vdots \\ h_{R80} &= \mu_{\alpha,P}(\alpha) \cdot \mu_{\beta,P}(\beta) \cdot \mu_{\Delta\alpha,P}(\Delta\alpha) \cdot \mu_{\Delta\beta,P}(\Delta\beta) \end{aligned} \quad (3)$$

ただし、 $\mu_{\alpha,N}(\alpha)$ は $\alpha = N$ に対応するメンバシップ関数を表しており、他の表記についても同様のものとする。実数値としての制御入力は制御規則の適合度に基づく加重平均として次式によって求められる。

$$s = \frac{\sum_{i=0}^{80} h_{Ri} \hat{s}_i}{\sum_{i=0}^{80} h_{Ri}}, \quad \phi = \frac{\sum_{i=0}^{80} h_{Ri} \hat{\phi}_i}{\sum_{i=0}^{80} h_{Ri}} \quad (4)$$

ただし、 \hat{s}_i および $\hat{\phi}_i$ は i 番目の制御規則の後件部 s_i および ϕ_i のラベルに対応する実数値である。

4 GA を用いたファジィ制御器の最適化

提案したファジィ制御器の性能はメンバシップ関数の形状や制御規則によって決定する。しかしながら、それらは互いに影響しており、また 81 通りの規則を手作業で調整することは困難である。そこで、実数値 GA を用いてそれらを最適化することとする。調整が必要な値はメンバシップ関数の広がりを表す変数 $a_\alpha, a_\beta, a_{\Delta\alpha}, a_{\Delta\beta}, a_{vS}, a_{vB}, a_{\phi S}, a_{\phi B}$ および制御規則の後件部 $s_0, \phi_0, \dots, s_{80}, \phi_{80}$ である。これらをコード化した染色体を Fig. 4 に示す。メンバシップの広がりを表す変数にはそれぞれ図に示す範囲の実数値が設定され、制御規則の後件部にはそれぞれラベルと対応付けされた整数値が設定される。

個体の評価を行うための目標軌道 $y_{t1} = f_1(x)$ は次式で表されるように直線と正弦波からなるものとする。

$$y_{t1} = \begin{cases} 0, & 0 \leq x < 4 \\ 0.5 \sin\left(2\pi \cdot \frac{x}{4}\right), & 4 \leq x \leq 12 \end{cases} \quad (5)$$

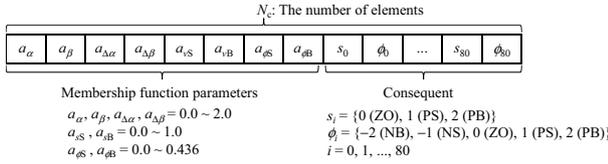


Fig. 4: Chromosome-encoded membership function parameters and consequent

また、個体の評価値 V は目標軌道を追従した際の y 方向の RMS 誤差 y_{RMS} と x 方向に進んだ距離 x_f および軌道の終端へと到達した時刻 t_f に基づく以下の評価関数によって求める。

$$V = \frac{k_1}{y_{\text{RMS}} + \varepsilon} + \frac{k_2}{t_f} + k_3 x_f \quad (6)$$

ただし、 k_1, k_2, k_3 は正のゲイン定数であり、 ε はゼロ除算を防ぐための値である。この評価関数の値が最大となるようなファジィ制御器を GA によって探索する。

5 シミュレーション

提案手法をシミュレーションによって評価する。ロボットの動作モデルとして次式に示す前輪操舵型の 4 輪移動ロボットをいることとした。

$$\begin{aligned} \dot{x} &= s \cos \theta \\ \dot{y} &= s \sin \theta \\ \dot{\theta} &= \frac{s \tan \phi}{L} \end{aligned} \quad (7)$$

ただし、 (x, y, θ) は世界座標上でのロボットの状態であり、 L は前輪と後輪の間の距離である。カメラは後輪から車体前方に向かって 0.24 [m] の地面から 0.13 [m] の位置に鉛直下向きより前方へ 60 [deg] 傾けた状態で搭載されているものとした。カメラのフレームレートは 20 [fps] とし、水平画角は 52 [deg]、垂直画角は 40 [deg] とした。画像のサイズは 320×240 [pixel] とした。目標軌道が画像の外に出た場合は 1 つ前の時刻の状態値をそのまま利用することとした。

実験 1 では GA を用いてファジィ制御器の最適化を行い、実験 2 では最適化したファジィ制御器によって任意軌道追従を行いその性能を評価する。なお、比較対象として、 α と β のみを状態変数とするファジィ制御器（規則数は 9）を用いる。以下では本稿にて提案したファジィ制御器を「制御器 1」、比較対象として用いる制御器を「制御器 2」と表記することとする。

5.1 実験 1: GA を用いたファジィ制御器の最適化

親の選択はルーレット選択によって行い、交叉は交叉率 0.9 の一様交叉とした。突然変異は染色体のある要素の値を、その要素がとり得る範囲内で生成したランダムな値に置き換えるものとし、その発生確率は要素数の逆数 $1/N_c$ とした。エリートは 1 体保存するものとした。1 世代の個体数は 20、1 回の試行は 30 [s] を制限時間とし、1000 世代で打ち切りとした。式 (6) の評価関数に用いられている定数はそれぞれ、 $k_1 = 1.0$ 、 $k_2 = 100$ 、 $k_3 = 5.0$ 、 $\varepsilon = 0.025$ とした。ロボットの初期状態は $(x, y, \theta) = (0$ [m], 0 [m], 0 [rad]) とした。

Table 1 に最適化されたファジィ制御器のパラメータ値を示す。なお、紙面の都合上 81 通りある制御規則は

Table 1: Optimized membership function parameters

Parameters	Controller 1	Controller 2
a_α	0.309	0.279
a_β	1.224	0.979
$a_{\Delta\alpha}$	1.42	-
$a_{\Delta\beta}$	1.09	-
a_{sS}	0.986	0.987
a_{sB}	0.989	0.991
$a_{\phi S}$	0.307	0.274
$a_{\phi B}$	0.428	0.346

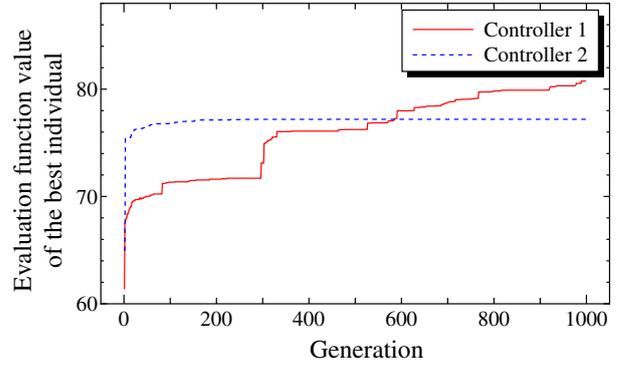


Fig. 5: History of the evaluation function values

割愛する。 a_{sS} と a_{sB} の値は両方の制御器でほとんど同じ値であるため、1 つの変数としてまとめることができると思われる。Fig. 5 に世代毎の最良個体の評価値を示す。制御器 2 はメンバシップ関数と制御規則の両方が制御器 1 と比べて少ないため、早い段階で評価値が飽和しており、これが制御器 2 の性能の限界であることが分かる。一方、制御器 1 はよりきめ細かい制御を行うことが可能であるため、600 世代付近までは、制御器 2 より評価値が劣るが最終的には制御器 2 を上回る評価値を得ていることがわかる。制御器 1 を用いたロボットの挙動について、300 世代付近までは初期位置から終端まで直進するロボットが最良個体として選ばれており、その後、目標軌道に沿って走行する個体が出現し少しずつ目標軌道との誤差が小さくなるように制御器が調整され、評価値が上昇している。計算機の性能の都合により 1000 世代で打ち切っているが、グラフの形状より制御器 1 の評価値はさらに上昇する可能性を秘めていると思われる。

5.2 実験 2: 任意軌道追従の性能

実験 1 で最適化したそれぞれの制御器によって行われる任意軌道追従の性能を評価する。目標軌道は次式 $y_{r2} = f_2(x)$ に示す正弦波形のものとした。

$$y_{r2} = 0.5 \sin \left(2\pi \cdot \frac{x}{4} \right), \quad 0 \leq x \leq 8 \quad (8)$$

ロボットの初期状態は $(x, y, \theta) = (-0.5$ [m], 0 [m], 0 [rad]) とした。

実験結果を Figs. 6, 7, 8 に示す。Fig. 6 は目標軌道とロボットの移動軌跡を示したものである。制御器 1 と制御器 2 の結果を比較すると制御器 1 を用いた場合の方がより目標軌道に近い位置を通っていることが分かる。しかしながら、目標軌道の波形の正のピーク付

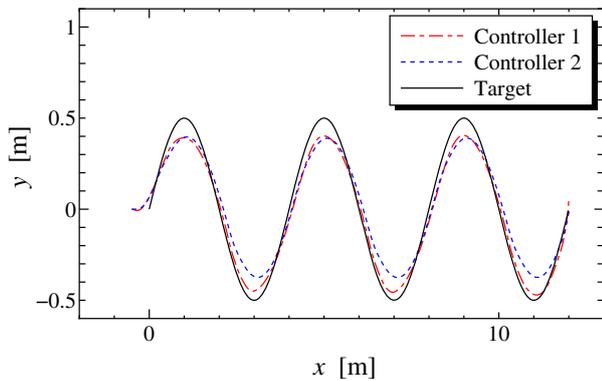


Fig. 6: Trajectory of the robot

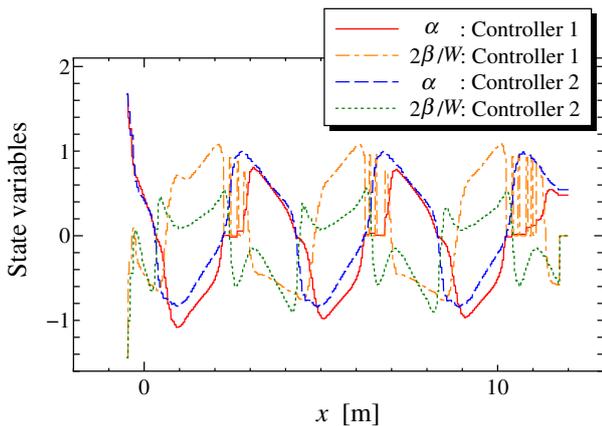


Fig. 7: Extracted state variables

近の誤差が、負のピーク付近の誤差に比べて大きいことがわかる。したがって、正のピーク付近の誤差はさらに減少させることが可能であると考えられる。Fig. 7はカメラ画像より抽出した状態変数の値を示している。全体を通して制御器1を用いた場合の方が値が大きいがわかる。これは、画像座標上で目標軌道が左右に大きく変動していることを意味し、部分的には α も β 同符号で ± 1 を超えていることから瞬間的に目標軌道を見失っていることがわかる。したがって、制御器1は目標軌道が画像の外に出た場合に、1つ前の時刻の値を用いるという取り決めを有効利用して、より目標軌道に接近しているものと思われる。しかしながら、目標軌道を見失うということは非常に危険なことであるため、本来はこれらのパラメータ値の変動を小さくすることが重要であり、評価関数にパラメータ値に関する項を追加する等の改善が必要であると思われる。Fig. 8は制御入力を示しており、制御器1を用いた場合は操舵角 ϕ が部分的に激しく変化していることがわかる。これでは実機に適用する場合にハードウェアに負担をかけてしまうため、制御入力の変動も可能な限り小さいほうが望ましい。したがって、上述の通り評価関数を見直して状態変数の値の変動を小さくするように改善することで、制御入力の変動も抑えることができると思われる。

6 まとめ

本稿では任意軌道追従のための画像に基づくファジィ制御器について、画像座標上の目標軌道の変化量 $\Delta\alpha$

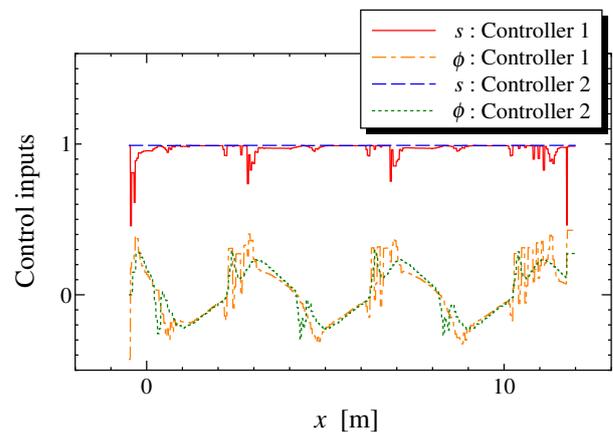


Fig. 8: Control inputs

および $\Delta\beta$ を状態変数として参照することで追従性能を向上させる手法を提案した。また、状態変数の増加に伴いファジィ制御器の調整が複雑になるという問題に対して、実数値GAによってファジィ制御器の最適化を行う手法を提案した。シミュレーションによる評価を行い、 $\Delta\alpha$ および $\Delta\beta$ を参照しない場合と比べて提案手法はより小さい誤差で目標軌道を追従できることが確認された。一方で、目標軌道を瞬間的に見失ってしまうという問題が発生した。したがって、評価関数に目標軌道の変動を抑えるための項を追加することが必要だと思われる。

参考文献

- 1) F. Chaumette, S. Hutchinson: Visual servo control, part I: basic approaches, IEEE Robotics and Automation Magazine, **13**-4, 82/90 (2006)
- 2) F. Chaumette, S. Hutchinson: Visual servo control, part II: advanced approaches, IEEE Robotics and Automation Magazine, **14**-1, 109/118 (2007)
- 3) A. Cherubini, F. Chaumette and G. Oriolo: A position-based visual servoing scheme for following path with nonholonomic mobile robots, IEEE/RSJ Int. Conf. on Intelligent Robotics and Systems, 1648/1654 (2008)
- 4) B. Espiau: Effect of camera calibration errors on visual servoing in robotics, Int. Symp. Exp. Robot., 182/192 (1993)
- 5) E. Malis, Y. Mezouar and P. Rivers: Robustness of image-based visual servoing with a calibrated camera in the presence of uncertainties in the three-dimensional structure, IEEE Trans. on Robotics, **26**-1, 112/120 (2010)
- 6) T. Kato, K. Watanabe and S. Maeyama: Image-based Trajectory Tracking with Fuzzy Control for Nonholonomic Mobile Robots, Proc. of the 37th Annual Conf. of the IEEE Industrial Electronics Society, 3180/3185 (2011)
- 7) T. Kato, K. Watanabe and S. Maeyama: Arbitrary Trajectory Tracking of a Mobile Robot by an Image-based Fuzzy Controller, Proc. of SICE Annual Conf., 2012, 160/165 (2012)
- 8) L. Davis: The Handbook of Genetic Algorithms, Van Nostrand Reinhold, New York, (1991)