

第6回 継続学習と知能の創発研究会

主催：日本知能情報ファジィ学会 継続学習と知能の創発研究会

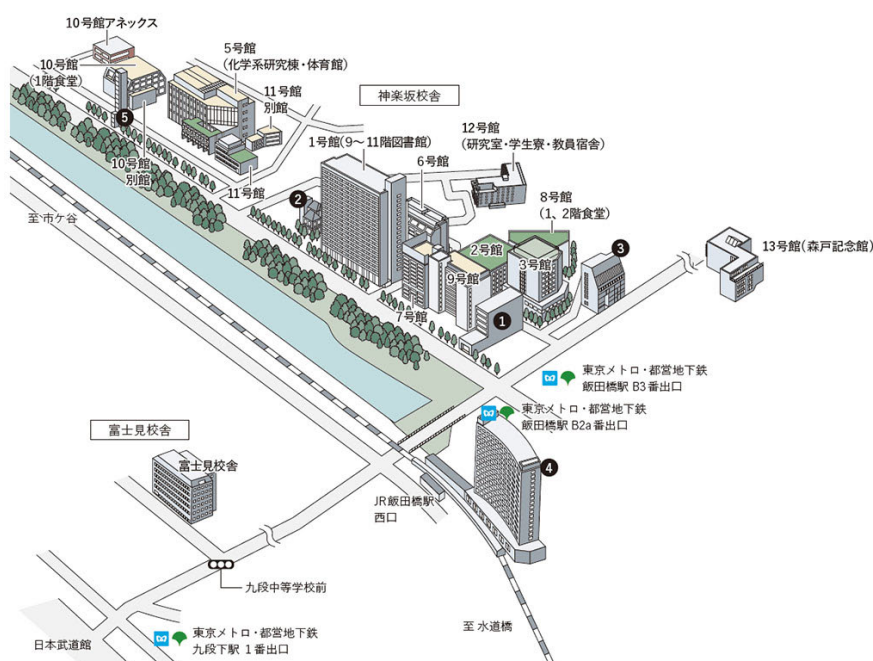
会期

2023年12月16日（土）、17日（日）

会場

東京理科大学 PORTA 神楽坂 7階 第3会議室(キャンパスマップの③)

<https://www.omu.ac.jp/isite/access/>



研究会 URL

https://www.cc.okayama-u.ac.jp/~surisys/eis/6th_eis.html



継続学習と知能の創発研究会
Continual Learning and Emergence of Intelligent Systems

プログラム

12/16 会場オープン (12:20~)

開会挨拶 (12:40~12:50)

第1セッション (12:50~14:05)

S1-01: パラメータ設定の不要な適応共鳴理論に基づく階層的クラスタリング手法の性能比較

○鳥越 大貴 (大阪府立大学), 増山 直輝 (大阪公立大学), 能島 裕介 (大阪公立大学)

S1-02: 適応共鳴理論に基づく階層的クラスタリング手法の比較検討

○田代 一貴 (大阪公立大学), 鳥越 大貴 (大阪公立大学),
増山 直輝 (大阪公立大学), 能島 裕介 (大阪公立大学)

S1-03: カーネル Ward 法に基づく HDBSCAN の検討

○大野 淳寛 (近畿大), 濱砂 幸裕 (近畿大)

第2セッション (14:15~15:55)

S2-01: PAMI: A New Python Library for Fuzzy Pattern Mining in Big Data

○RAGE Uday Kiran (The University of Aizu)

S2-02: 照度測定ロボットの自動走行における狭路通過のための制御系の検討

○坂本 倖規 (岡大), 戸田 雄一郎 (岡大), 松野 隆幸 (岡大)

S2-03: 3次元空間知覚システムのためのオドメトリを用いた Growing Neural Gas の補正手法の検討

○植野 有結 (岡大), 室本 達也 (岡大), 戸田 雄一郎 (岡大), 松野 隆幸 (岡大)

S2-04: 最小費用流問題の観点に基づくサイズコントロール機能を持つクラスタリングの検討

○青木 悠真 (近畿大学大学院), 濱砂 幸裕 (近畿大学情報学研究所)

第3セッション (16:05~17:45)

S3-01: Continual Learning with Fuzzy Inference Based on

Derivative-Free Independent Optimization of Each Fuzzy Rule: A Feasibility Study

○Kiyohiko Uehara (Ibaraki University)

S3-02: 筋電位を用いたソフトマニピュレータにおける姿勢推定手法の検討

○伊藤 裕太 (岡大), 須郷 賢 (岡大), 古田 優泰 (岡大), 戸田 雄一郎 (岡大),
脇元 修一 (岡大), 松野 隆幸 (岡大)

S3-03: 自己教師あり継続学習のためのクラスタリングによるリプレイバッファ選択方法の改良

○増山 直輝 (大阪公立大)

S3-04: ロボットの行動制御のためのトポロジカルインテリジェンス

○久保田 直行 (東京都立大学)

技術懇談会 (オンサイト) (18:15~)

12/17 会場オープン (10:00~)

第4セッション (10:30~11:45)

S4-01: ファジィ法線ベクトルとノードの信頼度に基づく領域成長法を用いた
ロバストなトポロジカルクラスタリング

○東海林 優也 (株式会社ロボティクス日本支店/東京都立産業技術高等専門学校/東京都立大学)

S4-02: 上・下限値が変動する時系列データに対する CLA の拡張

○張 強, 青木 健, 立川 智章, 藤井 孝藏 (東京理科大学)

S4-03: Study of Shape Recognition and Matching Based on Growing Neural Gas Generated Topology Structure

○Zhang Jiaqi (岡大), 戸田 雄一郎 (岡大), 松野 隆幸 (岡大)

第5セッション (13:00~14:15)

S5-01: Growing Neural Gas を用いた3次元空間知覚システムと2次元環境地図との統合

○久保平 大智 (岡大), 戸田 雄一郎 (岡大), 松野 隆幸 (岡大)

S5-02: 適応共鳴理論に基づく連合クラスタリング手法の比較検討

○上田 裕也 (大阪府立大学), 増山 直輝 (大阪公立大学), 能島 裕介 (大阪公立大学)

S5-03: 適応共鳴理論に基づくマルチラベル量質混在データのためのクラスタリング手法の改良

○西川 毅 (大阪公立大), 増山 直輝 (大阪公立大), 能島 裕介 (大阪公立大)

適応共鳴理論に基づく階層的クラスタリング手法の比較検討

Comparison of hierarchical clustering methods based on adaptive resonance theory

○ 田代 一貴¹, 鳥越 大貴², 増山 直輝¹, 能島 裕介¹

○ Kazuki Tashiro, Taiki Torigoe, Naoki Masuyama, Yusuke Nojima

¹ 大阪公立大学, ² 大阪府立大学

¹Osaka Metropolitan University, ²Osaka Prefecture University

カーネル Ward 法に基づく HDBSCAN の検討

○大野 淳寛¹, 濱砂 幸裕^{2,3}○ Atsuhiko Ohno¹, Yukihiro Hamasuna^{2,3}¹ 近畿大学大学院総合理工学研究科¹ Graduate School of Science and Engineering, Kindai University² 近畿大学情報学部² Faculty of Informatics, Kindai University³ 近畿大学情報学研究所³ Cyber Informatics Research Institute, Kindai University

1 はじめに

時系列データとは、時間経過と共に一定間隔で数値が変化するデータである [1]。時系列データを分析することで、株価の予測 [2] などが可能となる。時系列データに対する非類似度として、Dynamic Time Warping(DTW)[3] および Shape-based distance(SBD)[1] といった、時系列データの形状に着目した非類似度が知られている。

時系列データの分析手法の 1 つであるクラスタリングは、データ間の類似度もしくは非類似度に基づいて、データをクラスタと呼ばれるまとまりに分割する手法である。クラスタリング手法の 1 つに、階層的クラスタリングがある。階層的クラスタリングはクラスタを逐次的に結合する手法であり、クラスタの結合過程が樹形図として出力される。樹形図を用いることで、クラスタ生成の順序やデータのまとまりが可視化される。

階層的クラスタリングにおける計算手法の 1 つに Ward 法 [4] がある。Ward 法は、ユークリッド空間を仮定した手法であることが知られており、DTW および SBD といったユークリッド空間を仮定していない非類似度を扱うことは不適切である。この問題を解決するため、DTW および SBD より得られた非類似度を変換し、カーネル Ward 法 [4] に適用する手法が提案されている [5]。

階層的クラスタリングを用いた分析手法の 1 つに Hierarchical DBSCAN(HDBSCAN)[6] がある。HDBSCAN では、樹形図を基にデータを一定以上の個体を含むいくつかの領域に分割する。この領域のうち、データが多く存在する密度の高い領域をクラスタとみなす。密度の閾値を設けるのではなく領域同士を比較するため、密度の異なる領域が複数ある場合でもクラスタを発見しやすいことが知られている。ここで、HDBSCAN では最短距離法 [7] を用いて樹形図を生成しているが、最短距離法は 1 つのクラスタに 1 つずつ個体が結合されていく鎖効果が発生しやすい性質がある。鎖効果が発生している樹形図を基にデータの分割を行うと、個体が 1 つずつ分割されることで、領域が発見しづらくなる場合があるという問

題がある。

本稿では、この問題を解決するため、カーネル Ward 法を HDBSCAN に適用させる手法を提案する。Ward 法は最短距離法に比べて鎖効果を回避しやすい性質があるため、データを分割するときに一定以上の個体を含む領域を発見しやすくなる可能性がある。提案手法の有用性を検証するため時系列データに対して数値実験を行う。

2 カーネル Ward 法

カーネル Ward 法 [4] は、Ward 法における非類似度の計算にカーネル関数を用いる手法である。データ x, y の入力空間 R^p から高次元の特徴空間 H への写像 $\phi: R^p \rightarrow H$ を考える。このとき、 $k(x, y) = \langle \phi(x), \phi(y) \rangle$ を満たす ϕ が存在するとき、 $k(x, y)$ をカーネル関数という。

カーネル Ward 法では式 (1) に示すクラスタ内分散を用いる。式 (1) はクラスタ内分散を変形した式である。クラスタの集合を $\mathcal{G} = \{G_1, \dots, G_m\}$ としたとき、クラスタ $G_i, G_j \in \mathcal{G}$ の非類似度を、式 (2) のようにクラスタが結合したときのクラスタ内分散の増加分で定義する。カーネル Ward 法では $d(G_i, G_j)$ が最小のクラスタ同士を、クラスタ数が 1 つになるまで逐次的に結合する。

$$E(G) = \sum_{x \in G} k(x, x) - \frac{1}{|G|} \sum_{x \in G} \sum_{y \in G} k(x, y) \quad (1)$$

$$d(G_i, G_j) = E(G_i \cup G_j) - E(G_i) - E(G_j) \quad (2)$$

3 HDBSCAN

HDBSCAN[6] は、密度ベースのクラスタリング手法として知られる DBSCAN を階層的クラスタリングを用いて拡張した手法である。HDBSCAN のアルゴリズムを、データ $\mathbf{X} = \{x_1, \dots, x_n\}$ および 2 以上の整数で与えられるパラメータ m_{pts} を用いて説明する。まず、個体 $x_i \in \mathbf{X}$ と $x_j \in \mathbf{X}$ の非類似度を式 (3) で示す相互到達可能距離 $d_{mreach}(x_i, x_j)$ で定義する。ここで、 $d_{core}(x_i)$ は、 x_i から m_{pts} 番目の近傍点と x_i との非類似度である。

$$d_{mreach}(x_i, x_j) = \max(d_{core}(x_i), d_{core}(x_j), d(x_i, x_j)) \quad (3)$$

次に、相互到達可能距離を基に最短距離法を用いて樹形図を生成する。最短距離法は、最小の非類似度を持つクラスタ同士を逐次的に結合する手法である。生成された樹形図を入力として、**Algorithm 1** を用いて、図1のように樹形図をいくつかのクラスタに分割する。

Algorithm 1 HDBSCAN におけるクラスタの分割

Input: 生成された樹形図

Output: 分割された樹形図

- 1: $C_{parent} \leftarrow$ 親クラスタ
 - 2: $C_{left} \leftarrow$ 親クラスタが持つ左側の子クラスタ
 - 3: $C_{right} \leftarrow$ 親クラスタが持つ右側の子クラスタ
 - 4: $N(C_{left}) \leftarrow$ C_{left} が含む個体数
 - 5: $N(C_{right}) \leftarrow$ C_{right} が含む個体数
/* C_{parent} が子クラスタを持たないとき,
 $N(C_{left}) = 0, N(C_{right}) = 0$ とする*/
 - 6: **function** SPLIT(C_{parent})
 - 7: **if** $N(C_{left}) \geq m_{pts}$ and $N(C_{right}) \geq m_{pts}$ **then**
 - 8: C_{parent} が含む個体に新しいラベルを付ける
 - 9: C_{parent} を頂点とするクラスタを記録する
 - 10: SPLIT(C_{left}), SPLIT(C_{right})
 - 11: **else if** $N(C_{left}) \geq m_{pts}$ and $N(C_{right}) < m_{pts}$ **then**
 - 12: C_{right} が含む個体に外れ値ラベルを付ける
 - 13: SPLIT(C_{left})
 - 14: **else if** $N(C_{left}) < m_{pts}$ and $N(C_{right}) \geq m_{pts}$ **then**
 - 15: C_{left} が含む個体に外れ値ラベルを付ける
 - 16: SPLIT(C_{right})
 - 17: **else if** $N(C_{left}) < m_{pts}$ and $N(C_{right}) < m_{pts}$ **then**
 - 18: C_{parent} が含む個体に外れ値ラベルを付ける
 - 19: **end if**
 - 20: **end function**
 - 21: C_{top} = 最も大きい非類似度で結合しているクラスタ
 - 22: SPLIT(C_{top})
-

分割された樹形図では、大きなクラスタがいくつかの小さなクラスタを含んでいる。分割された樹形図から密度の高いクラスタを抽出するために、式(4)に示すクラスタ安定性 $S(C_m)$ を用いる。ここで、 $\lambda_p(x_i, C_m)$ は、個体

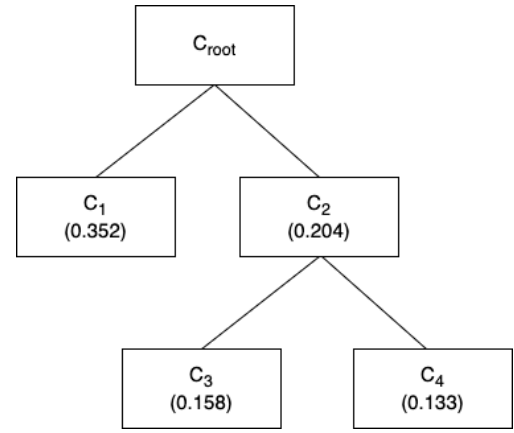


図1: 分割された樹形図 (数値はクラスタ安定性)

x_i がクラスタ C_m とは異なる新しいクラスタラベルを付けられたとき、もしくは外れ値ラベルを付けられたときの非類似度の逆数を表す。 $\lambda_{birth}(C_m)$ は、クラスタ C_m の親クラスタが分割されたときの非類似度の逆数を表す。

$$S(C_m) = \sum_{x_i \in C_m} \{\lambda_p(x_i, C_m) - \lambda_{birth}(C_m)\} \quad (4)$$

Algorithm 2 のようにクラスタ同士のクラスタ安定性を比較することでクラスタが抽出される。例えば、図1を入力として **Algorithm 2** を用いると、 C_1, C_3, C_4 がそれぞれクラスタとして抽出される。

4 カーネル Ward 法の適用

4.1 時系列データに対する非類似度のカーネル Ward 法への適用

時系列データ $\{x_1, \dots, x_n\}$ に対して DTW を用いて得られた非類似度を並べた行列 D_{DTW} を考える。 D_{DTW} の要素 $(D_{DTW})_{ij}$ は、 i 番目の個体 x_i と j 番目の個体 x_j の DTW による非類似度に対応する。Ward 法はユークリッド空間を仮定した手法であるため、DTW を用いて得られた D_{DTW} を Ward 法に適用することは不適切である。ここで、ユークリッド空間を仮定していない非類似度であっても、カーネル関数として利用可能であればカーネル Ward 法に適用できることが知られている [4]。この性質から、 D_{DTW} を変換することで、カーネル Ward 法に適用する手法が提案された [5]。

まず、 D_{DTW} の要素 $(D_{DTW})_{ij}$ を、式(5)のように類似度 $(S_\alpha)_{ij}$ に変換する。なお、 α は最大の $(D_{DTW})_{ij}$ より大きい数である。

Algorithm 2 HDBSCAN におけるクラスタの抽出

Input: 分割された樹形図

Output: 抽出されたクラスタ

- 1: $C_{parent} \leftarrow$ 親クラスタ
 - 2: $C_{left} \leftarrow$ 親クラスタが持つ左側の子クラスタ
 - 3: $C_{right} \leftarrow$ 親クラスタが持つ右側の子クラスタ
/* C_{parent} が子クラスタを持たないとき,
 $S(C_{left}) = 0, S(C_{right}) = 0$ とする*/
 - 4: **function** EXTRACT(C_{parent})
/* C_{parent} が C_{root} であるとき,
 $S(C_{parent}) = 0$ とする*/
 - 5: **if** $S(C_{parent}) \geq S(C_{left}) + S(C_{right})$ **then**
 - 6: C_{parent} をクラスタとして抽出する
 - 7: **else if** $S(C_{parent}) < S(C_{left}) + S(C_{right})$ **then**
 - 8: EXTRACT(C_{left}), EXTRACT(C_{right})
 - 9: **end if**
 - 10: **end function**
 - 11: $C_{root} \leftarrow$ 分割された樹形図における根のクラスタ
 - 12: EXTRACT(C_{root})
-

$$(S_{\alpha})_{ij} = \alpha - (D_{DTW})_{ij} \quad (5)$$

$(S_{\alpha})_{ij}$ を並べた行列 S_{α} は正定値行列になることが保証されていない。ここで、式 (6) のように十分大きな正の値 β を単位行列 I を用いて S_{α} の対角成分に加えることで、正定値行列 S_{β} に変換できる [4]。

$$S_{\beta} = S_{\alpha} + \beta I \quad (6)$$

S_{β} の要素 $(S_{\beta})_{ij}$ はカーネル関数 $k(\mathbf{x}_i, \mathbf{x}_j)$ として扱えるため、カーネル Ward 法に適用できる。

4.2 カーネル Ward 法に基づく HDBSCAN

HDBSCAN では、最短距離法を用いて樹形図を生成している。最短距離法は1つのクラスタに1つずつ個体が結合されていく鎖効果が発生しやすい性質がある。鎖効果が発生しているとき、Algorithm 1 において、 $N(C_{left})$ もしくは $N(C_{right})$ が1になり、 m_{pts} 未満になる状態が連続する場合がある。このとき、クラスタを分割して新しいラベルを付けることができないため、小さな領域が発見できない可能性がある。

この問題を解決するため、カーネル Ward 法を HDBSCAN に適用させる手法である K-W-HDBSCAN を提案する。Ward 法は鎖効果を生み出しにくい性質があるた

め、Ward 法と同様の樹形図を生成するカーネル Ward 法を用いることで、鎖効果を回避し小さな領域を発見することが期待できる。非類似度に DTW を用いた場合の K-W-HDBSCAN におけるクラスタリングの手順を Algorithm 3 に示す。

Algorithm 3 K-W-HDBSCAN

- STEP1:** DTW を用いて非類似度行列 D_{DTW} を計算する。
 - STEP2:** 式 (6) の変換を用いて D_{DTW} から S_{β} を得る。
 - STEP3:** S_{β} を基にカーネル Ward 法を用いて樹形図を生成する。
 - STEP4:** 生成された樹形図を基に Algorithm 1 を用いて分割された樹形図を得る。
 - STEP5:** 分割された樹形図を基に Algorithm 2 を用いて密度の大きいクラスタを抽出する。
-

5 数値実験

提案手法を用いて、UCR Time Series Classification Archive[8] で公開されている時系列データセットに対してクラスタリングを行った。実行結果は、Adjusted Rand Index(ARI)[9] を用いて評価した。ARI は2つのクラスタリング結果の類似性を評価する指標であり、結果が類似しているほど1に近い値を取る。本稿では、実験によるクラスタリング結果と、データセットの正解ラベルとの一致度が高いほど良好なデータ分割ができているとみなす。

5.1 実験条件

実験に使用したデータセットを、表1に示す。 C はクラスタ数を表す。クラスタ数とはデータセットの正解ラベルの種類である。

表1: 実験データの一覧

データセット名	C	データ数	データの長さ
BME	3	30	128
CBF	3	30	128
Plane	7	105	144

今回の実験では、提案手法である K-W-HDBSCAN に対して、非類似度をそれぞれ DTW, SBD, ユークリッド距離 (ED) としたものをを用いる。比較手法として、HDBSCAN に対して、非類似度をそれぞれ DTW, SBD, ユークリッ

ド距離 (ED) としたものをを用いる。また、各手法のパラメータ m_{pts} は全て 3 としている。

5.2 実験結果

表 2 に各データセットに対してクラスタリングを行った結果に対する ARI の評価を示す。同じデータに対するクラスタ分割で、最も ARI の高かった手法を太字で表している。

表 2: 実験結果の ARI 一覧

	BME	CBF	Plane
K-W-HDBSCAN(DTW)	0.466	0.744	0.925
K-W-HDBSCAN(SBD)	0.266	0.375	0.379
K-W-HDBSCAN(ED)	0.266	0.165	0.209
HDBSCAN(DTW)	0.397	0.519	0.231
HDBSCAN(SBD)	0.266	0.421	0.819
HDBSCAN(ED)	0.266	0.000	0.735

表 2 から、K-W-HDBSCAN(DTW) を用いた場合に比較的良好的なクラスタ分割ができる傾向にあることが示唆された。非類似度に注目すると、DTW、次いで SBD がそれぞれユークリッド距離よりも良好なクラスタ分割ができる傾向にあることから、時系列データの形状に着目した非類似度の方が、データの特徴を捉えられていることが示唆された。クラスタリング手法に着目すると、K-W-HDBSCAN の方が良好なクラスタ分割ができる傾向にあることから、カーネル Ward 法が最短距離法よりもクラスタを発見できていることが示唆された。

6 おわりに

DTW や SBD といった、時系列データの形状に着目した非類似度をカーネル Ward 法に適用させる手法を用いて、カーネル Ward 法に基づく HDBSCAN を提案した。実験結果より、カーネル Ward 法を用いることで、最短距離法を用いた手法よりも比較的良好的なクラスタ分割を行えることが示唆された。今後の課題として、カーネル Ward 法を用いたことによる鎖効果の軽減が良好なクラスタ分割に寄与したかどうかの検証が必要である。

参考文献

[1] J. Paparrizos, L. Gravano: k-Shape: Efficient and Accurate Clustering of Time Series, *Proc. of Special Interest Group on Management of Data*, pp. 1855–1870, 2015.

- [2] C. Guo, H. Jia, N. Zhang: Time series clustering based on ICA for stock data analysis, *Proc. of 4th International Conference on Wireless Communications, Networking and Mobile Computing*, pp. 1–4, 2008.
- [3] H. Sakoe, S. Chiba: Dynamic programming algorithm optimization for spoken word recognition, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. 26, pp. 43–49, 1978.
- [4] S. Miyamoto, R. Abe, Y. Endo, J. Takeshita: Ward method of hierarchical clustering for non-Euclidean similarity measures, *Proc. of 7th International Conference of Soft Computing and Pattern Recognition*, pp. 60–63, 2015.
- [5] 大野淳寛, 瀨砂幸裕: 時系列データに対する Ward 法の検討, 第 38 回ファジィシステムシンポジウム講演論文集 (FSS2022), TD3-1, 2022.
- [6] R. Campello, D. Moulavi, J. Sander: Density-Based Clustering Based on Hierarchical Density Estimates, *Proc. of 17th Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 160–172, 2013.
- [7] 宮本定明: 'クラスタ分析入門', 森北出版, 1999.
- [8] H. A. Dau, E. Keogh, K. Kamgar, C. M. Yeh, Y. Zhu, S. Gharghabi, C. A. Ratanamahatana, Y. Chen, B. Hu, N. Begum, A. Bagnall, A. Mueen, G. Batista, and Hexagon-ML: The UCR Time Series Classification Archive, 2018.
https://www.cs.ucr.edu/~eamonn/time_series_data_2018/
(最終閲覧日: 2023 年 11 月 19 日)
- [9] L. Hubert, P. Arabie: Comparing partitions, *Journal of Classification*, Vol. 2, No. 1, pp. 193–218, 1985.

連絡先

瀨砂 幸裕

〒 577-0813 大阪府東大阪市新上小阪 228-3

近畿大学情報学部

E-mail: yhama@info.kindai.ac.jp

照度測定ロボットの自動走行における狭路通過のための制御系の検討

○坂本倅規¹, 戸田雄一郎¹, 松野隆幸¹ (¹岡山大学)

**Examination of control system for narrow passage in automatic driving
of illumination measurement robot**

*K. Sakamoto¹, Y. Toda¹ and T. Matsuno¹ (¹Okayama University)

3次元空間知覚システムのためのオドメトリを用いた Growing Neural Gasの補正手法の検討

Investigation of a Correction Method of Growing Neural Gas Using Odometry
for 3D Spatial Perception Systems

○ 植野 有結¹, 室本 達也¹, 戸田 雄一郎¹, 松野 隆幸¹
○ Yui Ueno¹, Tatsuya Muromoto¹, Yuichiro Toda¹, Takayuki Matsuno¹

¹岡山大学

¹Okayama University

最小費用流問題の観点に基づく サイズ均等クラスタリングの検討

○ 青木 悠真¹, 瀨砂 幸裕^{2,3}○ Yuma Aoki¹, Yukihiro Hamasuna^{2,3}¹ 近畿大学大学院総合理工学研究科¹ Graduate School of Sci. and Eng., Kindai University² 近畿大学情報学部² Faculty of Informatics, Kindai University³ 近畿大学情報学研究所³ Cyber Informatics Research Institute, Kindai University

1 はじめに

k -means を代表とするクラスタリングは与えられたデータをデータ内の類似性尺度に基づき、データをクラスタと呼ばれるグループに分割する教師なし学習の手法である [1]. 各クラスタに含まれるデータ個体数を均等になるように制限するクラスタリング手法としてサイズ均等クラスタリング (Even-sized Clustering Based on Optimization; ECBO) がある [2]. ECBO では線形計画問題を解くことで帰属度の更新を行う。

クラスタリングの対象となるデータをクラスタへ割り当てることは二部マッチング問題と解釈することができる。また、二部マッチング問題を解く手法として最小費用流問題への変換が知られている。最小費用流問題はネットワークにフローを流した際のコストの総和が最小となるフローを求める問題である [3]. 最小費用流問題は負閉路消去法などの専用のアルゴリズムによって解くことができる。また、最小費用流問題のひとつである輸送問題と k -means や ECBO などの目的関数最適化に基づくクラスタリングの相関が示されている [4]. 本稿では ECBO における帰属度更新を最小費用流問題として捉えることを検討する。さらに帰属度の更新に最小費用流問題を解くパッケージを用いることで高速化が可能であるかを検討する。

2 準備

2.1 サイズ均等クラスタリング

ECBO は、 k -means にクラスタサイズを均等にする制約条件を加えたものである。以下の式においてデータセット $X = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ は p 次元の実ベクトルの集合であり、 n はデータ数である。 c はクラスタ数、 $\mathcal{C} = \{C_1, \dots, C_c\}$ をクラスタ集合とする。 K はクラスタサイズ、 u_{ki} は k 番目の個体のクラスタ i への帰属度、 \mathbf{v}_i は i 番目のクラスタ中心を表す変数である。クラスタサイズとはクラスタ内の個体数を指す。ECBO ではクラスタリングを行う前にクラスタサイズ K を算出し、 K

に基づいて帰属度を制限する。個体数 n とクラスタサイズ K の関係式を式 (1) に示す。

$$0 < n \leq (K + 1) \cdot \frac{n - (n \bmod K)}{K} \quad (1)$$

式 (1) を満たすとき、 c は式 (2) となる。

$$c = \frac{n - (n \bmod K)}{K} \quad (2)$$

ECBO の目的関数を式 (3)、制約条件を式 (4), (5) に示す。

$$\min. \quad J = \sum_{i=1}^c \sum_{k=1}^n u_{ki} \|\mathbf{x}_k - \mathbf{v}_i\|^2 \quad (3)$$

$$\text{s.t.} \quad \sum_{i=1}^c u_{ki} = 1, u_{ki} \in \{0, 1\}, \forall k \quad (4)$$

$$K \leq \sum_{k=1}^n u_{ki} \leq K + 1, \forall i \quad (5)$$

ECBO では整数計画問題を解いて帰属度の更新を行う。

ECBO におけるクラスタ中心の更新式は式 (6) で表される。

$$\mathbf{v}_i = \frac{\sum_{k=1}^n u_{ki} \mathbf{x}_k}{\sum_{k=1}^n u_{ki}} \quad (6)$$

ECBO のアルゴリズムを **Algorithm 1** に示す。

Algorithm 1 ECBO

STEP1 クラスタサイズ K を定めクラスタ数 c を算出し、 \mathbf{v}_i の初期値を個体からランダムに選択する。

STEP2 整数計画問題を解き、 u_{ki} を決定する。

STEP3 式 (6) に従い、 \mathbf{v}_i を更新する。

STEP4 収束条件を満たしていれば終了する。そうでなければ **STEP2** に戻る。

2.2 最小費用流問題

最小費用流問題はネットワークにフローを流す際のコストが最小となるフローを求める問題である。以下の式では頂点の集合を V , 枝の集合を A とする有向グラフ $G(V, A)$ において, 枝 a にフローを流す際のコストは $cost(a)$, 枝 a に流すフローを $\varphi(a)$, 枝 a のフローの下限を $\varphi^{lower}(a)$, 上限を $\varphi^{upper}(a)$, $\varphi(a)$, $\varphi^{lower}(a)$, $\varphi^{upper}(a)$ はそれぞれ0以上の整数値をとる. v をノードとし, $A^{in}(v)$ をノード v を終点とする枝の集合, $A^{out}(v)$ をノード v を始点とする枝の集合とし, 供給量を $b(v)$ とする. . 最小費用流問題の目的関数を式 (7), 制約条件を式 (8), (9) に示す.

$$\min. \quad J = \sum_{a \in A} cost(a)\varphi(a) \quad (7)$$

$$\text{s.t.} \quad \varphi^{lower}(a) \leq \varphi(a) \leq \varphi^{upper}(a), \forall a \in A \quad (8)$$

$$\sum_{a \in A^{out}(v)} \varphi(a) - \sum_{a \in A^{in}(v)} \varphi(a) = b(v), \forall v \in V \quad (9)$$

このとき $\sum_{a \in A^{out}(v)} \varphi(a)$ はノード v の流出量, $\sum_{a \in A^{in}(v)} \varphi(a)$ はノード v の流入量を表す.

3 最小費用流問題の観点に基づくサイズ均等クラスタリング

ECBO における帰属度の更新は以下のような最小費用流問題と考えることができる.

- データを表す n 個のノードとクラスタを表す c 個のノードからなる二部グラフに, 始点 s と終点 t を追加した図1のようなネットワークとする.
- 始点 s の供給量 $b(s)$ を n , 終点 t の供給量 $b(t)$ は $-n$ とする.
- 始点 s とデータを表すノード $k(k = 1, \dots, n)$ への枝 a_{sk} の容量 $\varphi^{upper}(a_{sk})$ を1, コスト $cost(a_{sk})$ を0, $\varphi(a_{sk}) = 1$ とする.
- データのノード k とクラスタのノード $i(i = 1, \dots, c)$ への枝 a_{ki} の容量 $\varphi^{upper}(a_{ki})$ を1, コスト $cost(a_{ki})$ を $\|\mathbf{x}_k - \mathbf{v}_i\|^2$, $\varphi(a) \in \{0, 1\}$ とする.
- クラスタのノード i から終点 t のノードへの枝 a_{it} の容量 $\varphi^{upper}(a_{it})$ を K , コスト $cost(a_{it})$ を0, $\varphi(a_{it}) = 1$ とするとする.

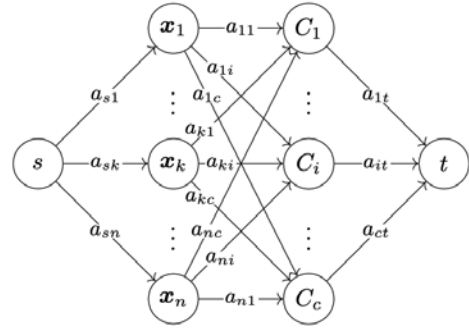


図 1: 帰属度の更新に用いるネットワーク

上記の最小費用流問題におけるデータのノード k からクラスタのノード i への流量をデータ k のクラスタ i への帰属度とする.

4 おわりに

本稿では k -means を発展させたクラスタリング手法である ECBO と最小費用流問題の関連性について検討した. 今後, クラスタサイズに下限と上限の制約を設けたサイズコントロールについても検討する.

参考文献

- [1] J. B. MacQueen, “Some methods for classification and analysis of multivariate observations”, *Proc. of the fifth Berkeley symposium on mathematical statistics and probability*, pp. 281–297, 1967.
- [2] Y. Endo, Y. Hamasuna, T. Hirano, and N. Kinoshita, “Even-Sized clustering based on optimization and its variants”, *Journal of Advanced Computational Intelligence and Intelligent Informatics (JACIII)*, Vol. 22, No. 1, pp. 62–69, 2018.
- [3] 繁野 麻衣子, ネットワーク最適化とアルゴリズム, 朝倉書店, 2010.
- [4] 遠藤 靖典, 輸送問題の観点からの目的関数最適化に基づくクラスタリングについて, 第38回ファジィシステムシンポジウム講演論文集, FD2-3, 2022.9.16.

連絡先

瀨砂 幸裕
〒577-0813
大阪府東大阪市新上小坂 228-3
近畿大学情報学部
TEL: 06-4307-3518
E-mail: yhama@info.kindai.ac.jp

Continual Learning with Fuzzy Inference Based on Derivative-Free Independent Optimization of Each Fuzzy Rule: A Feasibility Study

Kiyohiko Uehara

Ibaraki University, Hitachi, Ibaraki 316-8511, Japan

E-mail: kiyohiko.uehara.art@vc.ibaraki.ac.jp

1 Introduction

A method for continual learning with fuzzy inference is proposed on the basis of α -FUZZI-ES learning [1]. In α -FUZZI-ES learning, fuzzy rules can be evaluated independently of each other. This property is effective in continual learning because only fuzzy rules with low evaluations can be updated. The performance of the proposed method is demonstrated with simulations.

2 Formulation for Independent Evaluations of Each Fuzzy Rule

A method for evaluating fuzzy rules independently of each other has been proposed in [1], which is called α -FUZZI-ES. α -FUZZI-ES is briefly introduced in the following. Let X and Y be the input and the output spaces, respectively. α -FUZZI-ES is performed under the conditions that the input value $x \in X$ is numerical and $\sum_{j=1}^n \mu_{P_j}(x) = 1, \forall x \in X$. Here, n denotes the number of fuzzy rules and $\mu_{P_j}(x)$ is the membership function of the antecedent fuzzy set P_j . In α -FUZZI-ES, the evaluation value e_j of a fuzzy rule R_j is calculated by $e_j = \sum_{k=1}^{n_d} E_k \mu_{P_j}(\hat{x}_k)$, where n_d is the number of learning data given by numerical input-output pairs (\hat{x}_k, \hat{y}_k) ($k = 1, 2, \dots, n_d$). The symbol E_k denotes the evaluation value when \hat{x}_k is given as an input. In this paper, the small value of E_k means better performance. The mathematical derivation of e_j is shown in [1]. More efficient calculation of e_j is presented in [1]. An extended method of α -FUZZI-ES has been proposed in [2], which is called α -FUZZI-EX.

3 Continual Learning with α -FUZZI-ES Learning

The fuzzy-rule learning method with α -FUZZI-ES is called α -FUZZI-ES learning [1]. In α -FUZZI-ES learning, the consequent fuzzy sets Q_j ($j = 1, 2, \dots, n$) are tuned independently of each other so as to minimize e_j . Even when the evaluation functions are not differentiable, α -FUZZI-ES learning enables fuzzy rules to be optimized independently of each other. This property provides fast convergence in fuzzy rule learning [1].

Under the condition that $\sum_{j=1}^n \mu_{P_j}(x) = 1$, each fuzzy rule has its own coverage given by its antecedent fuzzy set. Although adjacent antecedent fuzzy sets overlap each other, α -FUZZI-ES can provide the way to evaluate fuzzy rules independently of each other. Thereby, α -FUZZI-ES learning makes it possible to update only fuzzy rules with the large value of e_j while maintaining the other fuzzy rules as previously tuned. This property of α -FUZZI-ES learning is effective in continual learning.

4 Numerical Results

In order to confirm the performance of α -FUZZI-ES learning in continual learning, α -FUZZI-ES learning is applied to interval prediction. As a cost function for evaluating fuzzy rules in learning, the coverage-width-based criterion (CWC) [4] is adopted. The numerical index of CWC is non-differentiable and thus it can demonstrate the effectiveness of α -FUZZI-ES learning with derivative-free optimization. In the interval prediction, a fuzzy inference method α -GEMII [3] is applied. α -GEMII mathematically proves that deduced fuzzy sets are convex and hence the supports of them can be used as prediction intervals.

Simulations are performed under the conditions shown in the following. The number of the fuzzy rules n and that of the learning data n_d are set to 21 and 501, respectively. The membership functions of P_j and Q_j ($j = 1, 2, \dots, 21$) are given by symmetric triangular functions in $[0, 1]$. The cores of P_j ($j = 1, 2, \dots, 21$) are placed at equal intervals in $[0, 1]$. The parameters for CWC are set the same as those shown in [1]. Each fuzzy rule is optimized with an

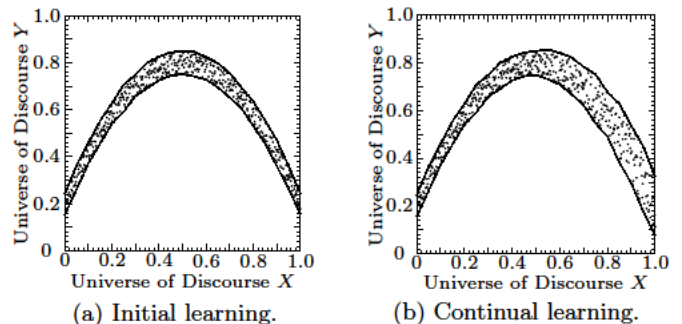


Fig. 1. Continual learning with α -FUZZI-ES learning.

artificial immune system (AIS) in the same way as in [1]. The number of antibodies is five and that of the clones for each antibody is four. The fuzzy rule learning with the AIS is performed iteratively for 500 generations.

The learning data are generated by the following equation: $\hat{y}_k = [-3.2(\hat{x}_k - 0.5)^2 + 0.9] + r_k$ ($k = 1, 2, \dots, 501$), where r_k denotes additional noise and \hat{x}_k ($k = 1, 2, \dots, n_d$) are regularly sampled values in $[0, 1]$. For initial learning, the value of r_k is given by a uniform random number in $[-0.05, 0.05]$. For continual learning, the range of r_k increases linearly from $[-0.05, 0.05]$ to $[-0.15, 0.15]$ as \hat{x}_k changes from 0.5 to 1.0; that is, the learning data for continual learning are different from those for initial learning data only when $\hat{x}_k \in [0.5, 1.0]$. Assuming that the cores of Q_j ($j = 1, 2, \dots, 21$) are ideally tuned, they are set by using the first term in the above equation for \hat{y}_k ; the core of Q_j is given by $-3.2(x_{P_j}^0 - 0.5)^2 + 0.9$, where $x_{P_j}^0$ denotes the core of P_j . Thereby, the effectiveness of CWC can be focused on in the fuzzy rule learning.

Figs.1 (a) and (b) show the results of the initial learning and the continual learning, respectively. As can be found in the figures, the prediction intervals are properly maintained in continual learning when $\hat{x}_k \in [0, 0.5]$. On the other hand, the prediction intervals are updated in continual learning when $\hat{x}_k \in [0.5, 1.0]$. It demonstrates that α -FUZZI-ES learning is feasible for continual learning.

5 Conclusion

A continual learning method with fuzzy inference has been proposed. The proposed method is based on α -FUZZI-ES learning. α -FUZZI-ES learning provides the way to evaluate fuzzy rules independently of each other. This property of α -FUZZI-ES learning makes it possible to maintain fuzzy rules with high evaluations same as in previous learning and to update only fuzzy rules with low evaluations in continual learning. Numerical results have shown the feasibility of α -FUZZI-ES learning for continual learning.

References

- [1] K. Uehara and K. Hirota, "A Fast Method for Fuzzy Rules Learning with Derivative-Free Optimization by Formulating Independent Evaluations of Each Fuzzy Rule," J. of Adv. Comput. Intell. Inform.(JACIII), Vol. 25, No. 2, pp. 213–225, 2021.
- [2] K. Uehara and K. Hirota, "Independent Evaluations of Each Fuzzy Rule for Derivative-Free Optimization of Fuzzy Systems: Toward Fast Fuzzy-Rules Learning for Fuzzy Inputs," Proc. of the 9th International Symposium on Computational Intelligence and Industrial Applications (ISChIA2020), 1A2-2-4, pp. 1–8, 2020.
- [3] K. Uehara, et al., "Inference with Governing Schemes for Propagation of Fuzzy Convex Constraints Based on α -Cuts," J. of Adv. Comput. Intell. Inform.(JACIII), Vol. 13, No. 3, pp. 321–330, 2009.
- [4] A. Khosravi, et al., "Prediction interval construction and optimization for adaptive neurofuzzy inference systems," IEEE Trans. Fuzzy Syst., vol. 19, no. 5, pp. 983-988, 2011.

筋電位を用いたソフトマニピュレータにおける 姿勢推定手法の検討

Study of pose estimation method for soft manipulator using EMG signals

○伊藤 裕太¹, 須郷 賢¹, 古田 優泰¹, 戸田 雄一郎¹, 脇元 修一¹, 松野 隆幸¹
○Yuta Ito¹, Ken Sugo¹, Yudai Furuta¹, Yuichiro Toda¹, Shuichi Wakimoto¹, Takayuki Matsuno¹

¹ 岡山大学

¹Okayama University

Recently, Japan's declining birthrate and aging population have increased the demand for robots in various industries. However, the rigidity of commonly developed robots makes it difficult to ensure safety for complex movements and objects. Therefore various types of soft manipulators characterized by mechanical softness have been developed. Since these manipulators are composed only of flexible materials, they are expected to play an active role in environments where safety is required, such as medical care and nursing care. In this study, we propose the method using machine learning with time-series data as input for classifying the posture estimation of soft manipulators based on myoelectric potentials. Specifically, we calculate several features from time-series data of muscle potentials, and verify their effectiveness through comparative experiments using Neural Networks (NN) and Recurrent Neural Networks (RNN).

1 緒言

近年、近年日本では少子高齢化が進み様々な業種でロボットの需要が高くなっている。しかし、一般的に開発されているロボットでは剛性が高いため複雑な動作や対象物への安全性の確保が難しいのが現状である。そこで機械的な柔らかさを特徴とする多様なタイプのソフトマニピュレータが開発されてきている。これらのマニピュレータは柔軟な材料のみで構成されるため、医療、介護などの安全性が要求される環境での活躍が期待できる。そのため本研究では、筋電位を用いたソフトマニピュレータの姿勢推定の分類に、時系列データを入力とした機械学習を用いる手法を提案する。具体的には、筋電位の時系列データからいくつかの特徴量を算出し、Neural Network(NN) や Recurrent Neural Network(RNN) を用いた比較実験を行うことでその有用性を確認する。

2 特徴量の算出

本研究では、姿勢推定分類の精度を向上させることを目的として、予め筋電位から複数の特徴量を算出している。これらの特徴量は組み合わせることで説明変数が増えるため、姿勢分類の正解率が上昇することが一般的に知られている。Tkachらはいくつかの特徴量を組み合わせる際、特徴量の組み合わせは4つが最も正解率が高くなると示している [1]。したがって本研究では、特徴量の組み合わせを4つまでに制限し実験を行う。以下に本研究で用いた特徴量をそれぞれ示す。ただし、 x_k はあるEMG センサにおける k 番目のデータを表している。

• Mean Absolute Value(mAV)

mAV は EMG センサにおける N 個の時系列データ

を用いた絶対値の平均に平均で表される。

$$mAV = \frac{1}{N} \sum_{k=1}^N |x_k| \quad (1)$$

ただし経験的に N は 250 とする。• Zero Crossings(ZC)

ZC は EMG センサの時系列データ内で正負が切り替わった数の合計である。具体的には、ZC は下記の条件を満たすときに 1 増加する。

$$\{x_k > 0 \text{ and } x_{k+1} < 0\} \\ \text{or } \{x_k < 0 \text{ and } x_{k+1} > 0\} \text{ and } |x_k - x_{k+1}| \geq 0 \quad (2)$$

• Waveform Length(waveLen)

waveLen は EMG センサにおけるある時刻と 1 時刻前のデータとの差分を合計したものである。

$$\text{waveLen} = \sum_{k=1}^N |x_k - x_{k-1}| \quad (3)$$

• Willson Amplitude(wAmp): 差分のバイナリ化

wAmp は EMG センサにおけるある時刻と 1 時刻前のデータとの差分の合計をバイナリ化したものである。

$$\text{wAmp} = \sum_{k=1}^N f(|x_k - x_{k+1}|) \quad (4)$$

$$f(x) = \begin{cases} 1 & \text{if } x > \epsilon \\ 0 & \text{otherwise} \end{cases}$$

ただし、経験的に $\epsilon = 0.0001$ とする。

以上で示した特徴量をいくつか組み合わせることで、実験の入力データに用いている。

3 RNN

本章では時系列データに強いRNNを用いて実験を行う。RNNは再帰型ニューラルネットワークと呼ばれており、数値の時系列データなどのパターンを認識するように設計された教師あり学習の一手法である。RNNは閉路上にコンテキストを持つ構造をしており、隠れ層が隠れ層自身に接続し、1時刻前の状態を次の入力値として扱う。今回用いたRNNはニューロン数300の中間層1層の3層構造である。入力データ数は4人の入力データを接続し1つのデータとしている。入力データ数は約260,000点で、サンプリングレートは1000[ms]とする。

ここで付録のTable1にRNNの学習時の実験結果、Table2にRNNの検証時の実験結果を示す。また、Fig. 1にRNNの各組み合わせ数の中でtest-Accuracyが最も高いもののAccuracyとF1-scoreを示す。

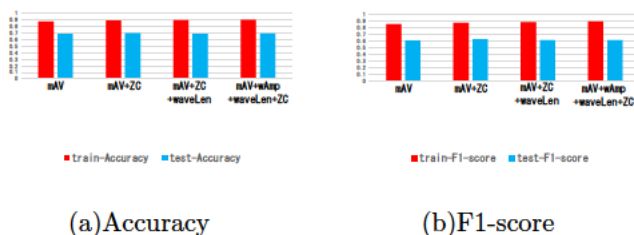


Fig 1: Results of RNN

Table1から、mAV+ZC+wAmp+waveLenの組み合わせがtrain-Accuracy・F1-scoreのどちらの値において最も高い結果が出るのが分かった。また、train-AccuracyはmAVを用いることや組み合わせる特徴量の数を増やすことで値が高くなっている。しかしTable2から、どの組み合わせでもtest-Accuracy:0.70前後、F1-score:0.60前後とあまり良い結果が得られないことが分かった。

4 NN

前章で示したRNNではtest-Accuracyは0.70前後でtrain-Accuracyと比べ低くなっており、F1-scoreは0.60前後とあまり高くなかった。これは今回用いたデータセットのデータ数が足りず、4人の入力データを接続することでデータ数を増やしたため、検証用データと学習用データでデータに偏りが生じたことにより起こった過学習だと考えられる。そのため本章では入力データをシャッフルすることが可能なNNを用いた姿勢推定分

類実験の結果を示す。今回用いたNNの構造はニューロン数300の中間層1層で構築された3層構造を用いた。また、RNNで起こった現象が入力データによるものか確認するためにシャッフルを用いたNNと用いていないNNの比較を行った。

ここでシャッフルを用いなかったNNの学習時の実験結果をTable3に検証時の実験結果をTable4に、シャッフルを用いたNNの学習時の実験結果をTable5に検証時の実験結果をTable6に示す。

Fig. 2にFig. 1と同じ特徴量の組み合わせでのシャッフルを用いなかったNNのAccuracyとF1-scoreを示す。

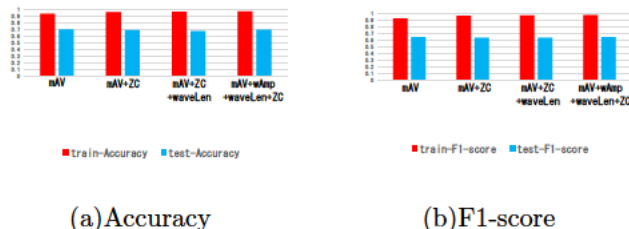


Fig 2: Results of NN without shuffling

Table3から、mAV+ZC+wAmp+waveLenの組み合わせにおいて、train-Accuracy・F1-scoreどの値も高い結果が出るのが分かった。また、train-AccuracyはmAVを用いることや組み合わせる特徴量の数を増やすことで高くなっている。しかしTable4から、どの組み合わせでも、test-Accuracy:0.70前後、F1-score:0.65前後とあまり良い結果は得られなかった。

Fig. 3にFig. 1と同じ特徴量の組み合わせでのシャッフル用いたNNのAccuracyとF1-scoreを示す。

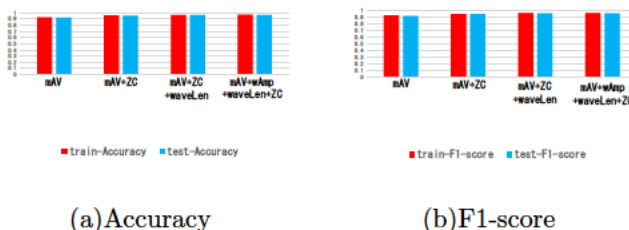
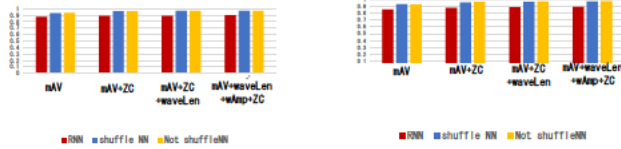


Fig 3: Results of NN with shuffling

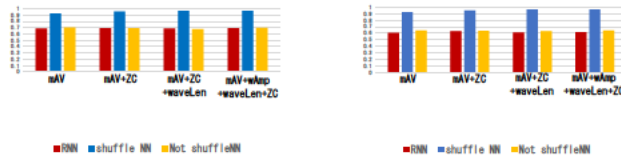
Table5,Table6から、mAV+ZC+wAmp+waveLenの組み合わせにおいてtrain-Accuracy・test-Accuracy・F1-

score のどの値も最も高い結果が出ることが分かった。また、RNN やシャッフルなし NN と同じく train-Accuracy が組み合わせる特徴量の数が増加するとともに高くなっているだけでなく、test-Accuracy・F1-score も高くなっている。



(a)train-Accuracy (b)train-F1-score

Fig 4: Results of each method with training data



(a)Test-Accuracy (b)Test-F1-score

Fig 5: Results of each method with training data

5 結言

本稿では、筋電位を用いて義手の姿勢推定分類を行う手法を提案した。実験結果から、特徴量を4つ組み合わせた場合が最もよい結果が得られることが分かった。また、シャッフルを用いた NN が train-Accuracy・test-Accuracy・F1-score が最も高い値を示した。しかし、今回の実験では4人の筋電位を接続した入力データを用いたため、シャッフルが不可能な RNN の結果やシャッフルを用いていない NN の結果が振るわなかったと考えられる。また train-Accuracy で RNN の値が最も低かった原因も前で述べた入力データが原因であると考えられる。これは RNN の特性である記憶保持において入力データの人が突然変わるためであると考えた。そこで今後は、十分なデータ量の1人の筋電位を用いて比較実験を行っていく予定である。

付録

Train-Accuracy(Tr A),test-Accuracy(Te A), train-loss(Tr L),test-loss(Te L), train- Precision(Tr P),test- Precision(Te P), train-recall(Tr R),test-recall(Te R), train-F1-score(Tr F1),test-F1-score(Te F1),mAV(M), ZC(Z), waveLen(W), wAmp(A) のように示す。

Table 1: Result of RNN

Input	TrA	TrL	TrP	TrR	TrF1
M	0.8721	1.2840	0.8632	0.8721	0.8524
Z	0.8394	0.7691	0.8074	0.8394	0.8147
W	0.8723	1.1880	0.8616	0.8723	0.8491
A	0.8525	0.9951	0.8306	0.8525	0.8335
M,Z	0.8887	1.2750	0.8864	0.8887	0.8739
M,W	0.8868	0.8099	0.8828	0.8886	0.8806
M,A	0.8772	1.2336	0.8720	0.8772	0.8725
Z,W	0.8784	1.4450	0.8617	0.8784	0.8632
Z,A	0.8723	1.0837	0.8677	0.8723	0.8566
W,A	0.8827	0.8377	0.8729	0.8827	0.8700
M,Z,W	0.8923	1.0840	0.8888	0.8923	0.8834
M,Z,A	0.8800	1.0548	0.8836	0.8880	0.8758
M,W,A	0.8719	1.0117	0.8600	0.8719	0.8618
Z,W,A	0.8790	1.4647	0.8725	0.8790	0.8701
M,Z,W,A	0.8986	0.9599	0.8958	0.8986	0.8914

Table 2: Result of RNN

Input	TeA	TeL	TeP	TeR	TeF1
M	0.6911	5.1725	0.6183	0.6911	0.6274
Z	0.6336	4.1420	0.5348	0.6336	0.5675
W	0.6768	4.2037	0.5788	0.6768	0.6011
A	0.6700	5.5387	0.5978	0.6700	0.6236
M,Z	0.6938	6.5192	0.6092	0.6938	0.6322
M,W	0.6819	4.6244	0.5841	0.6672	0.5975
M,A	0.6696	5.9474	0.5856	0.6697	0.6160
Z,W	0.6726	4.1938	0.5923	0.6726	0.6140
Z,A	0.6620	4.3851	0.5938	0.6620	0.6107
W,A	0.6884	5.3231	0.5949	0.6884	0.6316
M,Z,W	0.6868	5.3448	0.5827	0.6868	0.6142
M,Z,A	0.7039	4.4893	0.6038	0.7039	0.6414
M,W,A	0.6756	4.6152	0.5813	0.6756	0.6051
Z,W,A	0.6786	5.3785	0.5499	0.6786	0.5922
M,Z,W,A	0.6921	4.2898	0.5882	0.6921	0.6174

Table 4: Result of NN during test without shuffling

Input	TeA	TeL	TeP	TeR	TeF1
M	0.7011	5.8051	0.6352	0.7011	0.6441
Z	0.6345	4.7685	0.5543	0.6345	0.5870
W	0.7052	5.2447	0.5916	0.7052	0.6398
A	0.7101	4.8881	0.6396	0.7101	0.6635
M,Z	0.6924	5.6693	0.6002	0.6924	0.6373
M,W	0.7039	5.5362	0.6155	0.7039	0.6421
M,A	0.7182	6.2358	0.6237	0.7182	0.6568
Z,W	0.6879	6.1681	0.5856	0.6879	0.6276
Z,A	0.6809	6.5773	0.6034	0.6809	0.6326
W,A	0.7104	4.2494	0.6416	0.7104	0.6671
M,Z,W	0.6777	6.4124	0.6048	0.6777	0.6329
M,Z,A	0.6893	6.1824	0.6142	0.6893	0.6452
M,W,A	0.7110	6.3607	0.6244	0.7110	0.6583
Z,W,A	0.704	6.1961	0.6036	0.7045	0.6431
M,Z,W,A	0.6976	6.5130	0.6083	0.6976	0.6429

Table 5: Result of NN during training with shuffling

Input	TeA	TeL	TeP	TeR	TeF1
M	0.9305	0.1750	0.9270	0.9305	0.9281
Z	0.9128	0.2415	0.907	0.9128	0.9012
W	0.9227	0.1904	0.9154	0.9227	0.9082
A	0.9222	0.2106	0.9165	0.9222	0.9091
M,Z	0.9598	0.1005	0.9603	0.9598	0.9506
M,W	0.9571	0.1036	0.9550	0.9571	0.9557
M,A	0.9586	0.1003	0.9556	0.9586	0.9566
Z,W	0.9642	0.0887	0.9615	0.9642	0.9602
Z,A	0.9524	0.1189	0.9463	0.9524	0.9475
W,A	0.9584	0.1099	0.9578	0.9584	0.9519
M,Z,W	0.9678	0.0731	0.9687	0.9678	0.9627
M,Z,A	0.9720	0.0762	0.9707	0.9720	0.9696
M,W,A	0.9679	0.0756	0.9683	0.9679	0.9628
Z,W,A	0.9681	0.0763	0.9673	0.9681	0.9640
M,Z,W,A	0.9691	0.0712	0.9687	0.9691	0.9651

Table 3: Result of NN during training without shuffling

Input	TrA	TrL	TrP	TrR	TrF1
M	0.9366	0.1589	0.9346	0.9366	0.9222
Z	0.9175	0.2206	0.9130	0.9175	0.9031
W	0.9278	0.1778	0.9289	0.9278	0.9088
A	0.9257	0.2004	0.9297	0.9257	0.9033
M,Z	0.9597	0.0916	0.9563	0.9597	0.9551
M,W	0.9568	0.0980	0.9532	0.9568	0.9512
M,A	0.9564	0.1027	0.9526	0.9564	0.9536
Z,W	0.9606	0.0918	0.9574	0.9606	0.9575
Z,A	0.9533	0.1170	0.9480	0.9533	0.9472
W,A	0.9578	0.1087	0.9578	0.9578	0.9500
M,Z,W	0.9680	0.0749	0.9662	0.9680	0.9660
M,Z,A	0.9626	0.0895	0.9612	0.9626	0.9617
M,W,A	0.9688	0.0758	0.9675	0.9688	0.9679
Z,W,A	0.9666	0.0862	0.9672	0.9666	0.9622
M,Z,W,A	0.9694	0.0683	0.9684	0.9694	0.9688

Table 6: Result of NN during test with shuffling

Input	TeA	TeL	TeP	TeR	TeF1
M	0.9230	0.1891	0.9203	0.9230	0.9208
Z	0.8962	0.2694	0.8903	0.8962	0.8837
W	0.9148	0.2014	0.9078	0.9148	0.8989
A	0.9142	0.2287	0.9075	0.9142	0.9005
M,Z	0.9567	0.1070	0.9570	0.9567	0.9475
M,W	0.9560	0.1073	0.9540	0.9560	0.9547
M,A	0.9523	0.1075	0.9493	0.9523	0.9505
Z,W	0.9599	0.0961	0.9571	0.9599	0.9554
Z,A	0.9452	0.1299	0.9383	0.9452	0.9398
W,A	0.9531	0.1197	0.9525	0.9531	0.9464
M,Z,W	0.9647	0.0808	0.9652	0.9647	0.9597
M,Z,A	0.9707	0.0793	0.9694	0.9707	0.9687
M,W,A	0.9645	0.0827	0.9652	0.9645	0.9589
Z,W,A	0.9641	0.0845	0.9633	0.9641	0.9596
M,Z,W,A	0.9651	0.0806	0.9649	0.9651	0.9606

参考文献

- [1] Dennis Tkach, He Huang, Todd A Kuiken, “Study of stability of time-domain features for electromyographic pattern recognition”, Journal of NeuroEngineering and Rehabilitation, VOL. 7, NO. 21, 2010

上下限制約を持たない新しい大脳新皮質学習アルゴリズムの提案

張強[†], 青木健[†], 立川智章[†], 藤井孝藏[†]

東京理科大学[†]

1 はじめに

時系列予測とは, 未来のデータを予測することである. 予測結果を用いて効率的に意思決定できるため, リスク軽減や利益創出において時系列予測は重要な課題である. 実例として, 株価予測, 天気予測, 交通流量予測などが挙げられる. 従来の時系列予測方法は, 自己回帰モデル (AR), 移動平均モデル (MA), 自己回帰移動平均モデル (ARMA) や自己回帰積分移動平均モデル (ARIMA) などの統計的手法や, ニューラルネットワークに基づく RNN を拡張した長短期記憶 (Long Short-Term Memory, LSTM) [1] などの深層学習手法がある. 一方, 人間の脳の神経回路を模倣したアルゴリズム大脳新皮質学習アルゴリズム (Cortical Learning Algorithm, 以下 CLA) [2-4] が提案されている. CLA はデータを受け取りながらオンラインで予測器の内部構成を更新するところに利点があり, 有望な時系列予測法の一つであると考えられる [5].

しかし, 従来 CLA では, 時系列入力データの最小値と最大値を事前に設定する. 入力の値域が変化した場合, 新しい値域に合わせて再学習する必要があり, 学習コストが増大する問題がある. そこで, 本研究では動的に値域を変更しながら学習する CLA を提案する. これにより, 従来の最小値と最大値の制約を撤廃し, CLA の適用範囲の拡大および予測精度の改善を目指す.

2 大脳新皮質学習アルゴリズム

2.1 構成要素

CLA の構成要素を Fig. 1 に示す. CLA の予測器全体をリージョンと呼ぶ. リージョン内には, 複数のカラムを用意する. カラムは, 入力データ

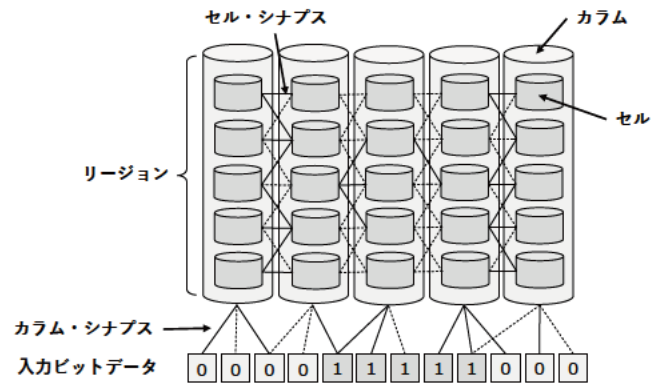


Fig. 1 CLA の予測器の構成

を学習器内の内部表現にするために使われる. それぞれのカラム内には, 複数のセルを用意する. セルは, 入力データの時系列データを内部表現するために使われる. 入力データはバイナリビット列で表現し, 各ビットとカラムの間にシナプスを配置する. また, セルとセルの間にもシナプスを配置する.

2.2 計算手順

CLA の全体像を Fig. 2 に示す. 最初にリージョン内に含まれる全てのカラム, セル, シナプスを初期化する. CLA は, 各時点 t においてエンコード, 空間プーリング, 時間プーリング, デコードを行う [2]. まず, 実数値である入力データ $X(t)$ を受け取り, それをバイナリ値であるビット列 $x(t)$

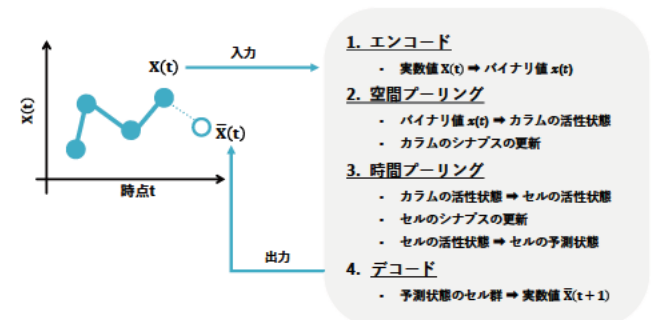


Fig. 2 CLA の全体像

Proposal of a new cortical learning algorithm without upper and lower bound constraints

[†] Qiang Zhang(4622522@ed.tus.ac.jp)

[†] Takeru Aoki(aoki@rs.tus.ac.jp)

[†] Tomoaki Tatsukawa(tatsukawa@rs.tus.ac.jp)

[†] Kozo Fujii(fujii@rs.tus.ac.jp)

Tokyo University of Science ([†])

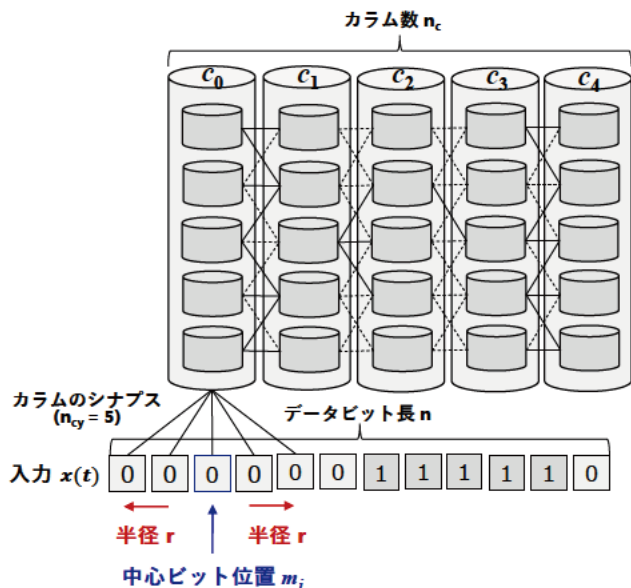


Fig. 3 CLA の初期化

なるカラムの部分集合として表現する。また、カラムのシナプスを更新する。そして、時間プーリングにおいて、入力データの文脈を全セル群の中から活性状態となるセルの部分集合として表現する。この手順では、セルのシナプスを更新する。また、次時点 $t+1$ に入力されるデータを予測状態となるセルの部分集合として表現する。最後に、予測状態のセル群を入力データと同じ形式にデコードし、予測値を出力する。

2.3 初期化

本研究では、青木らが提案した決定論的にシナプスを配置する初期化を用いる [5]。

Fig. 3 に示すように、各カラム c_i のシナプス配置中心ビット位置を m_i とする。入力データビット列の両端から半径 r ビットには、中心ビット位置 m_i を配置しない。カラム c_i の中心ビット位置 m_i は、以下の式で算出する。

$$m_i = i \cdot \frac{n - 2r}{n_c - 1} + r, \quad 0 \leq i < n_c \quad (1)$$

ここで、 n はデータビット長、 n_c はカラム数、 i は i 番目のカラムである。 n 、 n_c はパラメータである。

次に、半径 r 内の全てのデータビットに対してシナプスを配置する。半径 r は以下の式で求める。

$$r = \lfloor \frac{n_{cy} - 1}{2} \rfloor \quad (2)$$

ここで、 n_{cy} は各カラムのシナプス数であり、パラメータとして設定される。

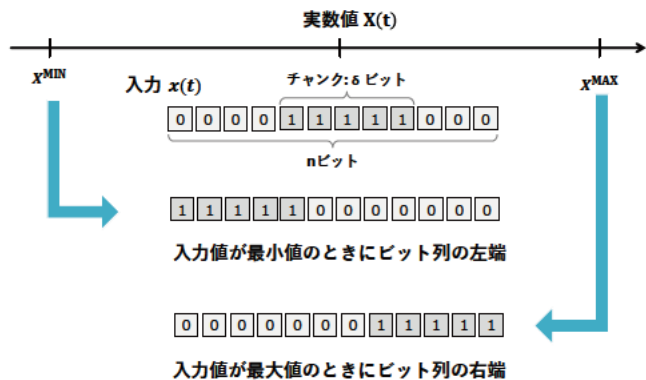


Fig. 4 従来の CLA エンコード

2.4 エンコード

CLA の予測器にビット列型に変換したデータ列を入力するため、実数値の入力をビット列型にエンコードする。各時点 t において、実数値である入力値 $X(t) \in [X^{min}, X^{max}]$ を受け取ることを仮定し、全体のデータビット長 n に対して、連続した δ ビットが 1 のチャンクとなるように変換する。

Fig. 4 に示すように、CLA は実数値 $X(t)$ をビット列 $x = (x_0, x_1, \dots, x_{n-1}) \in \{0, 1\}^n$ に変換する。チャンクの位置は、入力値が最小値 X^{min} の時にビット列の左端、最大値 X^{max} の時にビット列の右端となる。

各データビット $x_i (i = 0, 1, \dots, n-1)$ は、以下の式で決定される。

$$x_i = \begin{cases} 1, & \text{if } h(X(t)) < i \leq h(X(t)) + \delta, \\ 0, & \text{otherwise,} \end{cases} \quad (3)$$

ここで、 $h(X(t)) = \text{round}(\frac{(X(t) - X^{min}) \cdot (n - \delta)}{X^{max} - X^{min}})$ である。

2.5 カラムに基づくデコーダ

本研究では、青木らが提案したカラムに基づくデコーダを用いる [6]。カラムに基づくデコーダは、空間プーリングで構築された既存のシナプスネットワークを活用して予測ビット列を決定する。デコーダの概念図を Fig. 5 に示す。

カラムに基づくデコーダは、以下の手順でCLAの内部表現を予測値に変換する。

1. 各カラム $c_i (i = 0, 1, \dots, n_c - 1)$ について、予測状態のセル数を $c_i \cdot n_p$ として計数する。Fig. 5 において例を示す。カラム c_2 は予測状態のセルを二つ含んでいるため、2 となる。
2. 各予測データビット $x_j (j = 0, 1, \dots, n-1)$ について、接続状態のシナプスで紐づけら

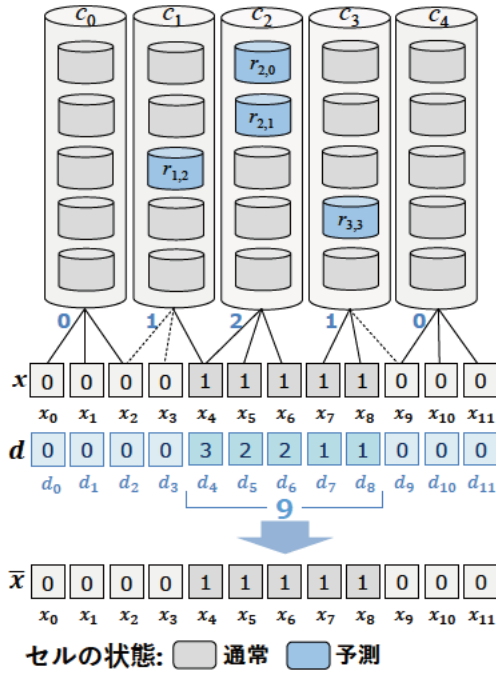


Fig. 5 カラムに基づくデコーダ

れたカラムに含まれる予測状態のセル数を $d_j = (j = 0, 1, \dots, n-1)$ として合計する。Fig. 5 において例を示す。データビット x_4 はカラム c_1 と c_2 との接続状態のシナプスを持っているため、 d_4 は 3 となる。

3. 現入力時点 t までに入力された入力ビット列集合 $\mathcal{X} = \{x^0, x^1, \dots\}$ について、以下の式を満たす予測ビット列候補集合 \mathcal{X}' を求める。

$$\mathcal{X}' = \arg \max_{x \in \mathcal{X}} \sum_{j=0}^n d_j x_j \quad (4)$$

4. 予測ビット列候補集合 \mathcal{X}' における中央値を求め、それを次時点 $t+1$ における予測ビット列 \bar{x} として出力する。式は以下ようになる。

$$\bar{x} = \text{median}(\mathcal{X}') \quad (5)$$

Fig. 5 において、 \mathcal{X} から最も $\sum_{j=0}^n d_j x_j$ が大きいビット列を選択し、 \bar{x} とする例を示す。ビット列 \bar{x} はカラムに基づくデコーダの出力となる。

3 CLA における現状の課題

従来の CLA では入力データの最小値と最大値を事前に設定し、その範囲内でデータを離散化し

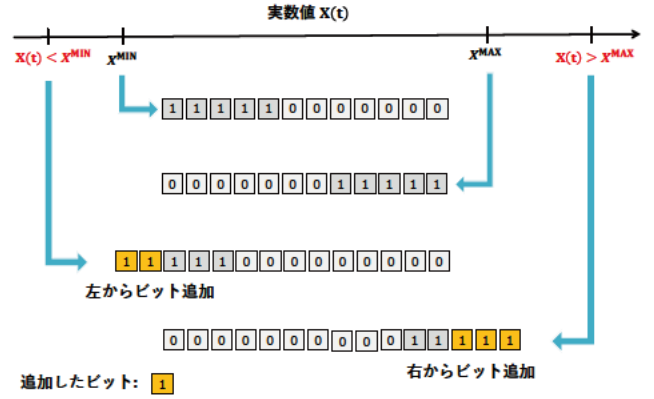


Fig. 6 ビット追加

て処理を行う。このとき、実世界の時系列データにおいて、事前にデータの最小値と最大値を知ることが難しい。設定された最小値と最大値を超えた場合、入力データビットとして表現できず、学習器の内部表現に変換できないため、予測が困難となる。一方、事前に最小値と最大値を大きく設定した場合、あるカラムが表現する入力データの値域が大きくなり、学習器が表現できる入力の解像度が落ちるため、予測誤差が増大する。

4 提案手法：上下制限約の撤廃

本稿では、従来 CLA の最小値と最大値の制限を撤廃し、CLA の適用範囲の拡大と予測精度の改善を目的として、入力データビットおよびカラムのシナプスを動的に追加する方法、それに合わせたカラムに基づくデコーダの拡張法を提案する。

4.1 入力データビット追加

実数値である入力値 $X(t)$ が最小値 X^{min} を下回る、または最大値 X^{max} を上回る場合、動的に入力データビットを追加する方法を提案する。各データビット $x_i (i = 0, 1, \dots, n-1)$ は、以下の式で決定される。

$$x_i = \begin{cases} 1, & \text{if } h(X(t)) < i \leq h(X(t)) + \delta, \\ 0, & \text{otherwise,} \end{cases} \quad (6)$$

ここで、 $h(X(t)) = \text{round}(\frac{(X(t)-X^{min}) \cdot (n-\delta)}{X^{max}-X^{min}})$ である。各時点 t において、チャンクの開始位置 $h(X(t))$ を計算する。ここで、Fig. 6 に示したように、

- $X(t) < X^{min}$ の場合、左からビットを追加する。 $X^{min} = X(t)$ に更新する。
- $X(t) > X^{max}$ の場合、右からビットを追加する。 $X^{max} = X(t)$ に更新する。

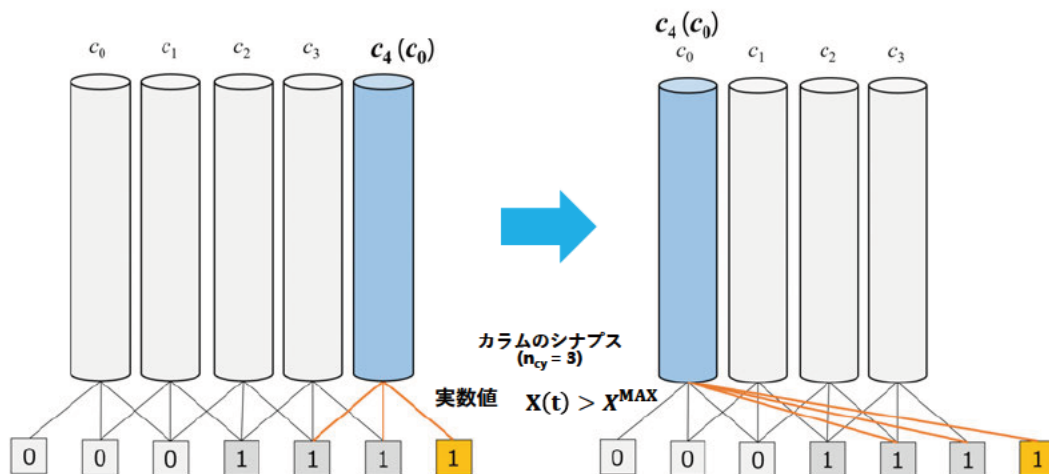


Fig. 7 カラムのシナプス追加

- $X(t) \in [X^{min}, X^{max}]$ の場合, ビットを追加しない.

例えば, 最小値と最大値はそれぞれ 0, 1 で, データビット長 $n = 421$, チャンクの長さ $\delta = 21$ に設定したとする. $t = 1$ 時点での入力値は -0.1 とした場合, $X(t) < X^{min}$ のため, 左から 40 ビットを追加する. そして, 左から 21 個のビットが 1 となり, それ以外の 440 個のビットは 0 となる.

4.2 カラムのシナプス追加

追加されたビットを CLA の内部表現に変換できるようにするため, カラムのシナプスを追加する方法を提案する. 従来 CLA では, 各カラムはビットの中心位置をもとにシナプスを生成している. これを活用し, 左端のカラム c_0 と右端のカラム c_{n_c} が周期境界を持つと仮定することで, 螺旋構造を作る.

カラムのシナプス追加の例を Fig. 7 に示す. カラム数 $n_c = 4$, カラムのシナプス $n_{cy} = 3$ とする. まず, 入力値 $X(t)$ が最大値 X^{max} より大きい際に, 1 ビットが追加される. 次に, 4 番目のカラムを仮定し, ビットの中心位置をもとに 3本のシナプスを生成する.

i 番目のカラム中心位置 m_i は以下の式で求める.

$$m_i = i \cdot \frac{n - 2r}{n_c - 1} + r, \quad i \in \mathbb{N} \quad (7)$$

ここで, i は i 番目のカラム, n はデータビット長, n_c はカラム数, r は半径である.

そして, 右端のカラム c_{n_c} が周期境界を持つと仮定し, カラム c_4 はカラム c_0 と同じカラムであるため, c_0 に c_4 で生成したシナプスを追加する.

他方, ビットの追加にあわせてカラムを追加することでも, 必要なカラムを確保できると考えられる. しかし, 大量のビットが追加された場合, それに合わせて大量のカラムを増やす必要がある. その結果, CLA の学習器が肥大化し, 計算コストが増大すると考えられる. そのため, 本研究ではカラムを増やさない方法を提案する.

4.3 カラムに基づくデコーダの拡張

ビットを追加した際, 螺旋構造によりあるカラムが複数の値を表現するため, デコードにおいて対応する複数の値の中でどれを予測値とすべきか判別できない問題点がある.

これに対して, 入力値を活用し, 入力値に近い値を選択する方法を提案する. Fig. 8 および Algorithm 1 に示したように, 提案法は入力値のチャンク位置を使って, 中心ビットから左右に設定した range 幅まで広げる. これにより, 予測値が入力値から連続的に変化すると仮定し, 連続値の特性にあった予測値をデコードすることができる.

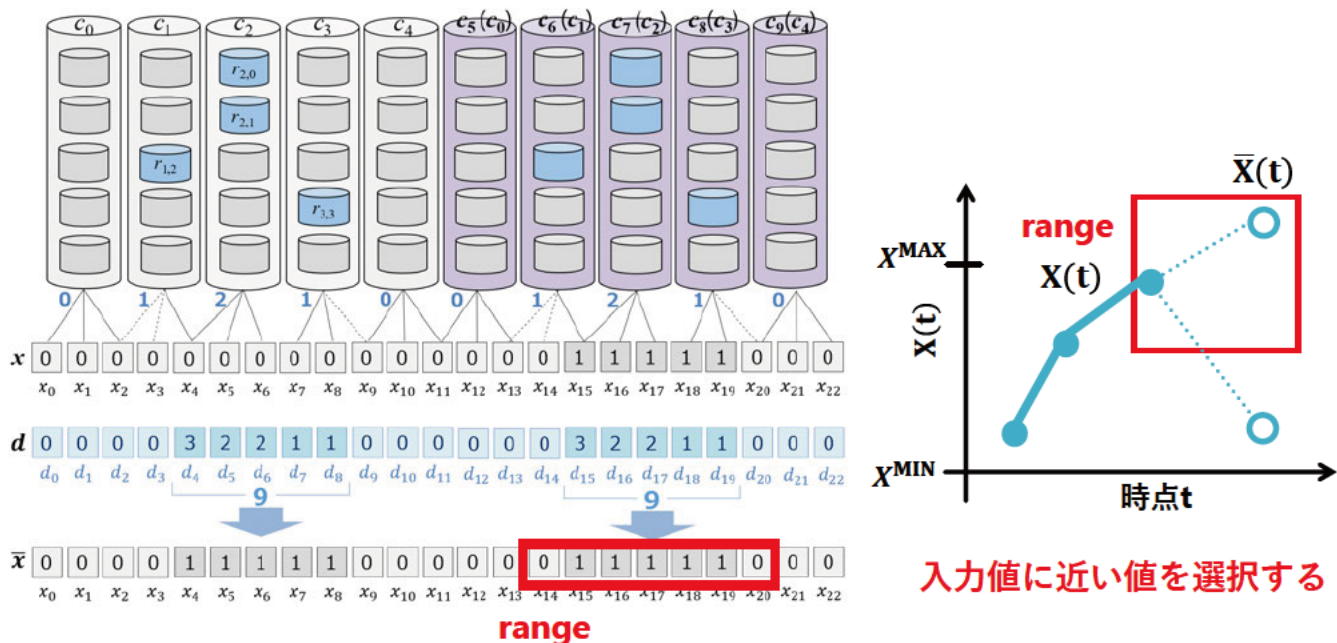


Fig. 8 カラムに基づくデコーダの拡張

Algorithm 1 Calculate Range Algorithm

Require: integer centerindex, unsigned integer rangeSize, vector(double) actvalue
Ensure: pair(unsigned int, unsigned int) representing start and end indices

- 1: $start \leftarrow centerindex - \frac{rangeSize-1}{2}$
- 2: $end \leftarrow centerindex + \frac{rangeSize-1}{2}$
- 3: **if** $start < 0$ **then**
- 4: $end \leftarrow end - start$
- 5: $start \leftarrow 0$
- 6: **end if**
- 7: **if** $end \geq size(actvalue)$ **then**
- 8: $overflow \leftarrow end - (size(actvalue) - 1)$
- 9: $end \leftarrow end - overflow$
- 10: $start \leftarrow start - overflow$
- 11: **end if**
- 12: **return** $make_pair(start, end)$

5 実験方法

提案法の効果を検証するため、様々な特徴を持つ人工時系列データを用いて従来の CLA と提案法を比較する。

5.1 入力データ

人工時系列データは周期的なデータである正弦波 $X_{sin}(t)$ を基に、正弦波、トレンドを持つ正弦波、ある時点で変化する正弦波、複数時点で変化する正弦波の4種類を用いる。各入力データは以下の式で表される。

$$X_{sin}(t) = \frac{1}{2} \cdot \sin\left(\frac{(t-1) \cdot \pi}{50}\right) + \frac{1}{2} \quad (8)$$

$$X_{trend}(t) = \frac{t}{20000} + X_{sin}(t) \quad (9)$$

$$X_{change}(t) = \begin{cases} X_{sin}(t) & \text{if } 0 \leq t \leq 20000, \\ 1 + X_{sin}(t) & \text{if } 20001 \leq t \leq 40000 \end{cases} \quad (10)$$

$$X_{chnages}(t) = \begin{cases} X_{sin}(t) & \text{if } 0 \leq t \leq 20000, \\ 1 + X_{sin}(t) & \text{if } 20001 \leq t \leq 40000, \\ X_{sin}(t) & \text{if } 40001 \leq t \leq 60000 \end{cases} \quad (11)$$

入力データセットの詳細はTab. 1 に示す. Eq. (8) に示したように, $X_{sin}(t)$ は単純な正弦波を用いる. 従来 CLA の予測の収束を考慮し, 正弦波 200 周期分にあたる 20,000 入力を全入力数として設定した. Eq. (9) に示したように, $X_{trend}(t)$ は時点が進むにつれて平均が増加していく正弦波を用いる. 増加量が多すぎると CLA での予測が困難となるため, 20,000 時点で合計 1 増加するように設定した. Eq. (10) に示したように, $X_{change}(t)$ はある時点で変化する正弦波を用いる. 学習のため, 前半と後半で 20,000 時点ずつ用意し, 20,000 時点において平均が 0.5 から 1.5 に変化するよう設定した. Eq. (11) に示したように, $X_{changes}(t)$ は複数時点で変化する正弦波を用いる. $X_{change}(t)$ に変化前の平均値に戻る部分を加えた正弦波であり, 学習のため, 各 20,000 時点で計 60,000 時点を用意した.

Table 1 入力データセット

データ名	最小値	最大値	時点数	範囲	中心
$X_{sin}(t)$	0	1	20000	[0, 1]	0.5
$X_{trend}(t)$	0	2	20000	[0, 2]	1
$X_{change}(t)$	0	3	40000	[0, 3]	1.5
$X_{chnages}(t)$	0	3	60000	[0, 3]	1.5

5.2 アルゴリズム

Table 2 実験設定

アルゴリズム	$[X^{min}, X^{max}]$	range
CLA(Fit)	[中心-範囲/2, 中心+範囲/2]	-
CLA(Wide)	[中心-5*範囲/2, 中心+5*範囲/2]	-
CLA(Narrow)	[0, 1]	-
CLA-DR	[0, 1]	201

従来 CLA として, 決定論的にシナプスを配置する初期化及びカラムに基づくデコーダを適用した CLA を用いる. また, 従来 CLA に提案手法を導入した CLA-DR を提案法とする.

本稿では, Tab. 2 に示すように, 最小値 X^{min} と最大値 X^{max} を変更した三つの従来 CLA と CLA-DR を比較する. CLA(Fit) では, 入力データ全体の最小値と最大値を X^{min} および X^{max} として使用した. CLA(Wide) では, X^{min} および X^{max} は事前に設定した最小値と最大値の 5 倍とした. CLA(Narrow) では, 入力値が設定した最小値もしくは最大値を超えた場合, それぞれを

最小値もしくは最大値に丸める. また, 提案法 CLA-DR では range を 201 に設定した.

5.3 パラメータ

CLA のパラメータは Tab. 3 に示す. また, 試行回数は 10 を設定し平均で比較する. 評価指標として, 入力値と予測値の差を予測誤差として算出する.

Table 3 CLA のパラメータ

データビット長	n	421
チャンクの長さ	δ	21
カラム数	n_c	2048
カラムのシナプス数	n_{cy}	22
カラムのシナプスの接続閾値	θ_c	0.1

6 結果および考察

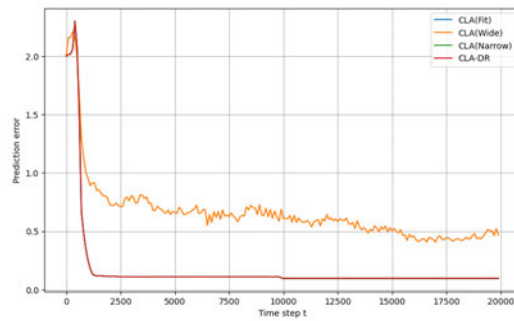
6.1 正弦波における結果

正弦波における結果を Fig. 9 に示す. Fig. 9(a) に示したように, CLA(Wide) は他手法よりも予測精度が悪化する. 実際に入力値よりも大きい範囲の最小値と最大値を設定したため, 予測値の解像度が落ちたと考えられる. また, CLA-DR は設定した最小値と最大値がデータビットの範囲内のため, ビットが追加されない. よって, CLA(Fit), CLA(Narrow) および CLA-DR の結果は同じとなる.

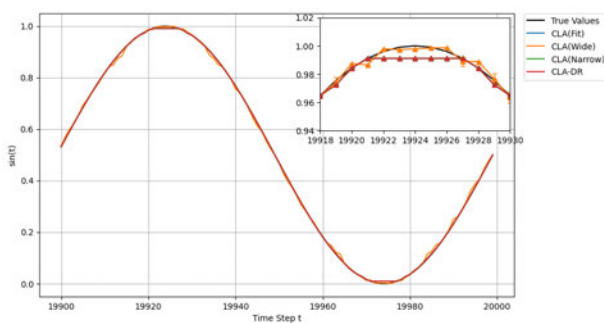
6.2 トレンドを持つ正弦波における結果

トレンドを持つ正弦波における結果を Fig. 10 に示す. Fig. 10 に示したように, CLA(Narrow) は他手法よりも予測精度が悪化する. Fig. 10(b) に示したように, CLA(Narrow) は設定した最小値と最大値を超えた際に丸めるため, 想定した範囲外の値を予測できないことがわかる.

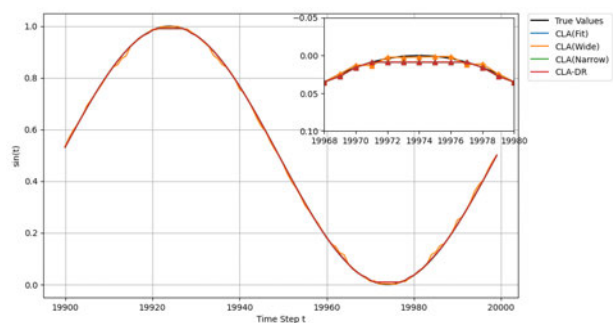
次に, Fig. 10(d) に示したように, 提案法の CLA-DR は CLA(Fit) と比較すると, 予測前期は予測誤差が増大するものの, 予測後期において予測誤差を低減することがわかる. これは, CLA-DR はトレンドによる値域の変化によりビットが追加されていくため, 最初から値域全体を網羅する CLA(Fit) よりも 1 ビットあたりの値域が細かく, 詳細な予測が表現できたと考えられる.



(a) 1 周期ごとの予測誤差合計



(b) 予測値と真値(拡大図 1)



(c) 予測値と真値(拡大図 2)

Fig. 9 正弦波における結果

6.3 ある時点で変化する正弦波における結果

ある時点で変化する正弦波における結果を Fig. 11 に示す. Fig. 11 に示したように, 提案法の CLA-DR は予測精度が最も良いことがわかる. これは, 変化前の値域に対して最小値と最大値を初期化し, その上で変化後の値域に対してビットを追加することで詳細な予測表現を維持できたと考えられる. 一方, CLA(Fit) では変化前後にかかわらずすべての値域を網羅するため, 入力が存在しない値域にもカラムを割り当て, 有効活用できないカラムが現れると考えられる.

6.4 複数時点で変化する正弦波における結果

複数時点で変化する正弦波における結果を Fig. 12 に示す. Fig. 12 に示したように, 提案法の CLA-DR では予測精度が最も良く, 従来法である CLA(Fit) を上回ることが分かった.

以上より, 様々な人工時系列データに対して, CLA-DR が良好な予測精度を示すことがわかった.

7 まとめ

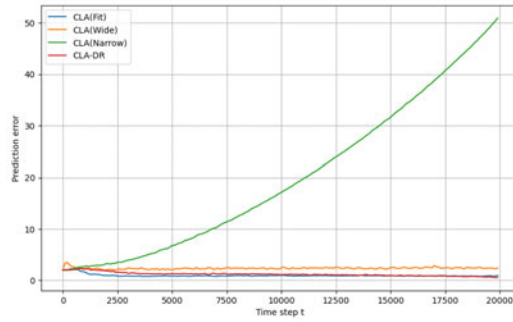
CLA における入力の値域の制限を撤廃するため, 入力データビットおよびカラムのシナプスを動的に追加する方法を提案した. カラムとシナプスの関係に周期境界を仮定することで, 類似した値におけるカラム表現の類似度を維持しながら, あるカラムが複数の値を表現できる.

時系列データを用いた実験の結果, 提案法である CLA-DR が値域の変化にあわせてシナプスを追加し, 設定した最小値・最大値を超えても予測できることがわかった. また, 入力の離散化粒度を維持したまま値域を拡張することで, 粒度を下げることで前もって値域を調整した従来 CLA よりも予測精度を改善できることを示した.

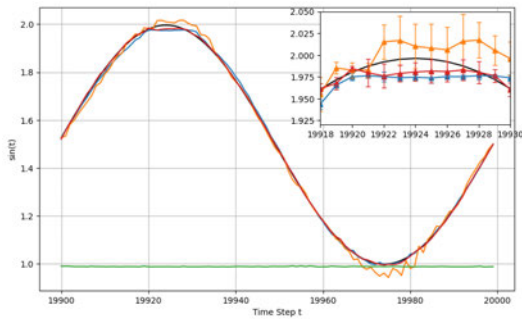
今後の予定として, 実世界の時系列データセットを用いて提案法を検証する.

参考文献

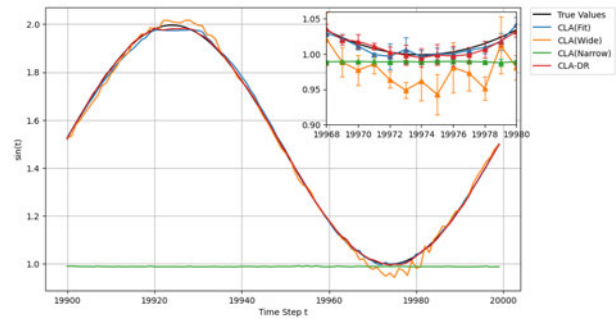
- 1) S. Hochreiter and J. Schmidhuber. "Long Short-Term Memory." *Neural Computation*, Vol.9, No.8, pp.1735-1780, 1997.



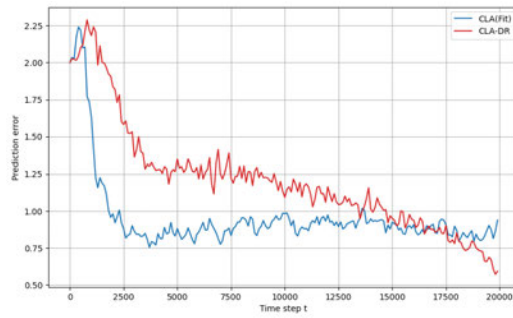
(a) 1周期ごとの予測誤差合計



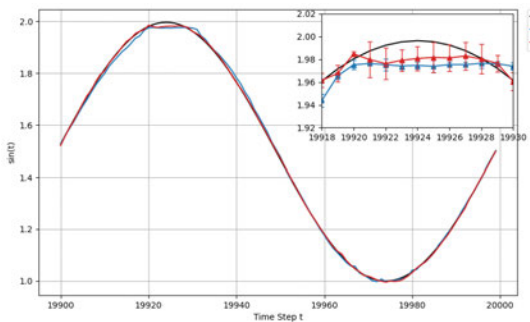
(b) 予測値と真値(拡大図1)



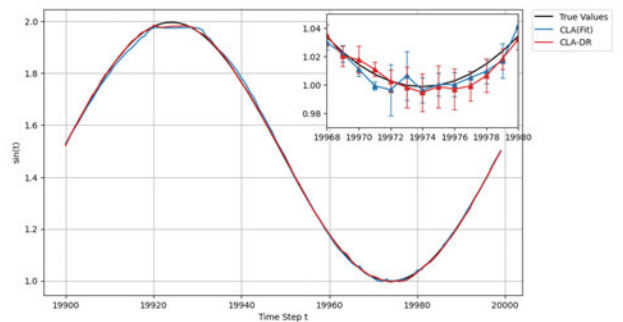
(c) 予測値と真値(拡大図2)



(d) 1周期ごとの予測誤差合計(一部抜粋)

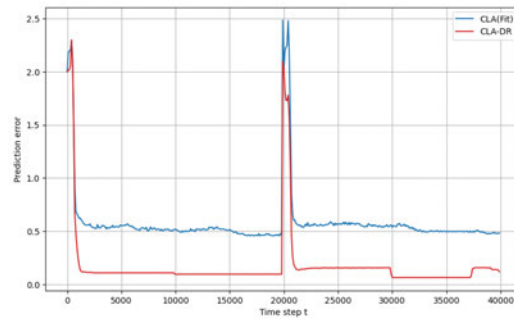


(e) 予測値と真値(一部抜粋・拡大図1)

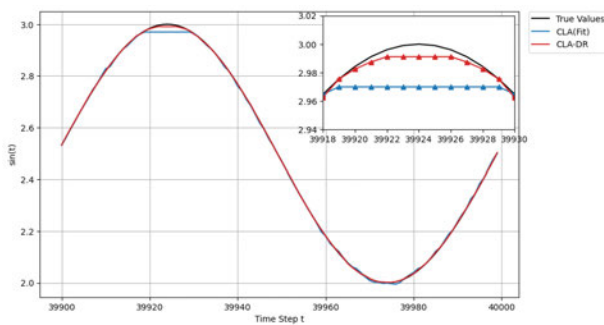


(f) 予測値と真値(一部抜粋・拡大図2)

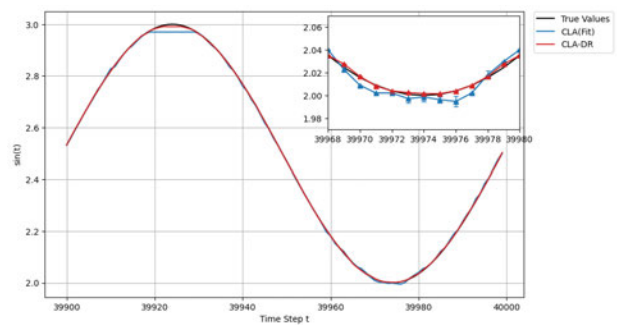
Fig. 10 トレンドを持つ正弦波における結果



(a) 1周期ごとの合計の誤差



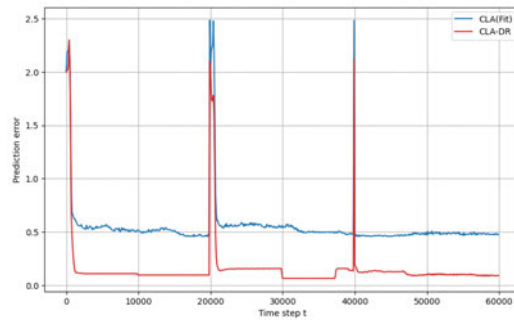
(b) 予測値と真値(拡大図 1)



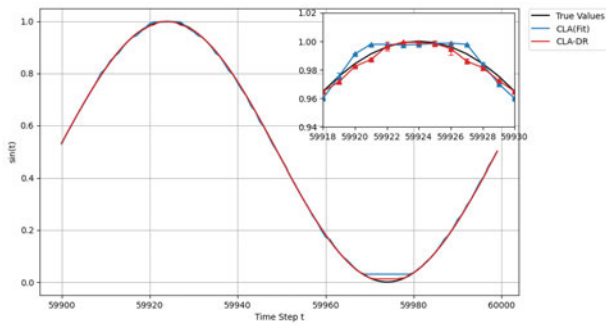
(c) 予測値と真値(拡大図 2)

Fig. 11 ある時点で変化する正弦波における結果

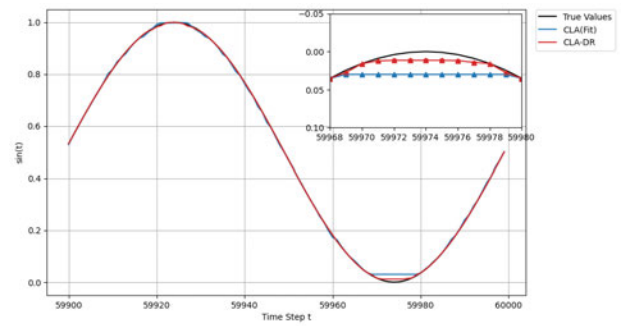
- 2) J. Hawkins, A. Subutai, and D. Dubinsky. “Hierarchical Temporal Memory including HTM Cortical Learning Algorithms.” *Technical report, Numenta, Inc.*, 2010.
- 3) J. Hawkins and A. Subutai. “Why Neurons Have Thousands of Synapses, a Theory of Sequence Memory in Neocortex.” *Frontiers in Neural Circuits*, Vol.10, pp.1-13, 2016.
- 4) NuPIC.
<https://github.com/numenta/nupic>
Accessed on 2023/09/02.
- 5) Takeru Aoki, Keiki Takadama, and Hiroyuki Sato. “Adaptive Synapse Arrangement in Cortical Learning Algorithm.” *Journal of Advanced Computational Intelligence and Intelligent Informatics (JACIII)*, Vol.25, No.4, pp.450-466, 2021.
- 6) Takeru Aoki, Keiki Takadama, and Hiroyuki Sato. “Column-Based Decoder of Internal Prediction Representation in Cortical Learning Algorithms.” *Joint 11th International Conference on Soft Computing and Intelligent Systems and 21st International Symposium on Advanced Intelligent Systems (SCIS&ISIS2020)*, pp.1-7, 2020.



(a) 1周期ごとの合計の誤差



(b) 予測値と真値(拡大図1)



(c) 予測値と真値(拡大図2)

Fig. 12 複数時点で変化する正弦波における結果

Study of Shape Recognition and Matching Based on Growing Neural Gas Generated Topology Structure

* Continual Learning and Emergence of Intelligent Systems

1st JIAQI ZHANG

2nd YUICHIRO TODA

3rd TAKAYUKI MATSUNO

OKAYAMA University

OKAYAMA University

OKAYAMA University

Abstract—Shape recognition is a main challenging problem in computer vision. Different approaches and tools are used to solve this problem. Most existing approaches to object recognition are based on pixels. Pixel-based methods are dependent on the geometry and nature of the pixels, so the destruction of pixels reduces their performance. In this paper, we construct a graph that captures the topological and geometrical properties of the object using growing neural gas(GNG). Then, using the coordinates and relation of its vertices, we extract features that are robust with respect to rotation and scale variation. We applied distortion and noise to some shapes and tried to match them to the original image, the results show that matching using this method is feasible.

Index Terms—Shape Recognition, GNG Graph, Graph-Based Features, Robot vision systems, Mobile robots

I. INTRODUCTION

Shape is a significant concept in image understanding. It includes considerable meaningful properties of an object and provides stability to different object deformations (such as articulation, occlusion, and noise) and transformations (such as rotation, translation, and scale). In this paper, we study object recognition from a graph theory viewpoint. Graphs are robust with respect to rotation, articulation, and noise, so they can be effective tools to capture image properties. We use the growing neural gas (GNG) algorithm to construct a GNG graph model of input data incrementally. The number of vertices of the graph does not depend on the size of the image and it is not sensitive to articulation and noise on the boundary. In this graph, every vertex has a coordinate, so we can use the geometrical properties of the graph as well. We use both topological and geometrical properties of this graph to extract meaningful features from the image. Both theoretic discussions and experimental results show that our method is invariant to articulation, noise, occlusion, rotation and scale.

II. METHODOLOGY

In this section, we introduce our proposed GNG method. We use a graph to approximate the object. The vertices are scattered almost uniformly inside the object. The main steps are:

- 1) Constructing a graph that models the object using the growing neural gas (GNG) algorithm. The constructed graph is called GNG graph.

- 2) Extracting the outer boundary of the GNG graph, using computational geometry approaches and extracting geometrical and topological features from this graph.
- 3) Measuring the similarity between objects using a dynamic programming algorithm.

The system consists of three main parts, the first part is floor guide map recognition and intelligence extraction, the second part is the matching of robot local map (generated by slam) and guide map information, and the third part is the path planning based on area distribution.

A. Advantages of GNG Graph

Our method is based on a GNG graph. This graph must have the following properties:

- The vertices are placed almost uniformly inside the object and the edges have almost equal lengths.
- The number of vertices is fixed and does not depend on the size of the object.
- The graph is robust with respect to noises. It ignores holes and cracks inside and on the boundary of the object.

Different approaches can be chosen to construct this graph. We used the GNG algorithm because it satisfies the mentioned properties, the running time is satisfiable, and can be extended to 3D object representation and object tracking. Also, since the structure is not fixed but adaptable, GNG can learn new evolving patterns in an online learning process and is able to adapt to dynamic changing operating conditions.

B. Boundary Extraction

In this phase, we extract the outer boundary of the GNG graph. If the graph is 2-connected, then the outer boundary is a cycle, otherwise, it is a closed walk. So, we can store its vertices in a cyclic array, C , in a clockwise order of appearance. The idea is similar to the idea of a convex hull algorithm. The full steps are like:

- 1) Find the rightmost vertex v and its neighbor u with the smallest clockwise angle with the vertical upward ray starting at v . Insert v and u in C .
- 2) Consider the two last vertices i and $i - 1$ in C , and for all vertices j , adjacent to i , compute the size of the clockwise angle between the edges $i, i - 1$ and i, j .

Select a vertex with a minimum angle as the next vertex on the boundary and insert it in C .

- 3) If the last two vertices in C are equal to initial vertices (u and v), exit otherwise go to step 3. (This helps pass the cut vertex if it exists.)

Fig. 1 shows an example of a GNG graph and its outer boundary.

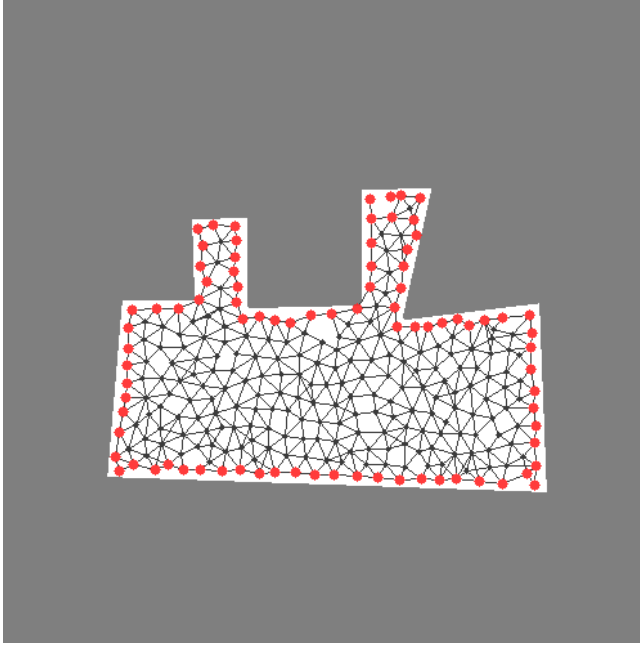


Fig. 1. GNG structure and contour detection on a binary image.

III. FEATURE EXTRACTION

Although the vertices on the outer boundary of a GNG graph capture special information about the object, the relation between the boundary and internal vertices can provide new topological features that lead to better recognition. We introduce features for each vertex on the outer boundary, then compute and combine these features to describe the global and local properties of the shape. The features are perimeter (P), boundary-in-disc (B) and convex hull-area (CH) for outer boundary vertices of GNG. Given the GNG graph of an object, let $U = u_1, u_2, \dots, u_n$ define the sequence of outer boundary vertices in clockwise order. A feature vector $F_i = (P_i, B_i, CH_i)$ is computed for every outer boundary vertex $u_i, i \in 1, \dots, n$. F_i consists of three features P_i, B_i, CH_i and C_i with lengths $|P_i| = m_1, |B_i| = m_2, |CH_i| = m_3$ and $\sum_{j=1}^3 m_j = m$. So F_i can be considered as a m -dimensional vector. We compute F_i for every outer boundary vertex $u_i, i \in 1, \dots, n$ and define $\mathbf{F} = \mathbf{F}(U) = (\mathbf{F}_1, \mathbf{F}_2, \mathbf{F}_3, \dots, \mathbf{F}_n)$ which is a $m \times n$ matrix. Let G be the GNG graph of an object and $V(G)$ be the set of vertices in G . For every two vertices $v, u \in V(G)$, $d(u, v)$ shows their distance in G . For every outer boundary vertex u_i and for every integer j , let $D_j(u_i) = \{v \in V(G) : d(v, u_i) \leq j\}$ be the discrete disc of radius j around u_i and $D_j(u_i) = \{v \in V(G) : d(v, u_i) = j\}$

be the boundary of the discrete disc. Figure 3 shows the discrete discs around a vertex in a sample image. We define four features P, B , and CH for a given GNG graph with outer boundary vertices $U = u_1, u_2, \dots, u_n$ as follows (shown as fig. 2):

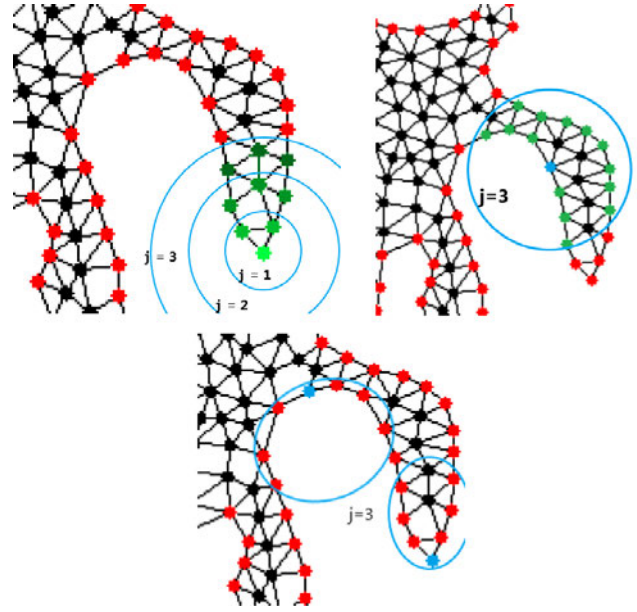


Fig. 2. Upper-left: feature P for a single node; Upper-right: feature B for a single node; Bottom: comparison between two nodes of feature CH , a larger value means a convex area.

- Perimeter (P): For each outer boundary vertex u_i , perimeter counts the number of vertices that are in $D_j(u_i)$ for $j \in 1, \dots, m_1$. Small values of j describe local properties, while larger values represent global properties of the shape. (See Fig.4a).
- Boundary-in-disc (B): The number of outer boundary vertices inside $D_j(u_i)$ for each outer boundary vertex u_i where $j \in 1, \dots, m_2$ is called boundary-in-disc. If the boundary has peaks and troughs inside the disc, this number is bigger, so this feature can keep the shape of the boundary if it is computed for different radii.
- Convex hull-area (CH): For each outer boundary vertex u_i and integer $j \in 1, 2, \dots, m_3$, let S be the convex hull of vertices in $D_j(u_i)$ that are on the outer boundary of G . The number of vertices of $D_j(u_i)$ that are inside S is called convex hull-area.

To determine the scale number m_i (maximum radius), they follow this condition: If the average difference of features between neighboring scales m_i and $m_i + 1$ is less than a threshold, the scale m is enough.

IV. MATCHING

To carry out matching between the two contour sets, we simply slide one feature sequence on the other and search for the point that minimizes the distance between the vectors. Let p_1, p_2, \dots, p_n and q_1, q_2, \dots, q_m be the boundary sequence of two objects A and B , respectively. A matching π from

A to B is defined as a mapping from p_1, p_2, \dots, p_n to q_1, q_2, \dots, q_m where p_i is mapped to $q_{\pi(i)}$ if $\pi(i) = 0$ and otherwise p_i remains unmapped. The cost of π is defined as $\sum_{i=1}^n c(p_i, q_{\pi(i)})$, where $c(p_i, q_{\pi(i)})$ equals the Euclidean distance between feature vectors p_i and $q_{\pi(i)}$. Fig. 3 shows part of the matching between two patterns, one of which is partially cut away and rotated by 180 degrees. Another matching in fig. 4 is made in a situation close to real conditions, where the pattern on the left represents one of the areas on the floor and another is a simulation of the metric map constructed by the robot. We consider this result acceptable as The area referred to can be marked on the map more or less correctly.

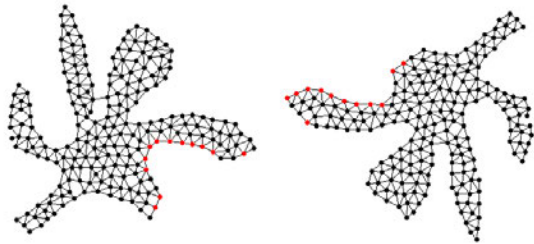
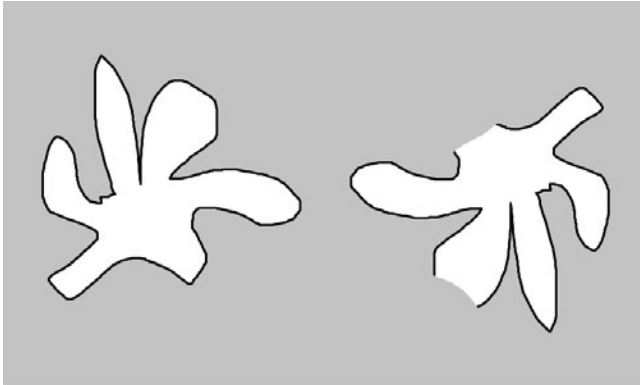


Fig. 3. A matching example

V. ATTEMPTS ON SCALE-INVARIANT

Considering that in real-world usage scenarios, we will sometimes need to match parts of the pattern to the whole, in addition, the scale of the input data (the scale of the image) is likely to be inconsistent. In order to maintain the consistency of the feature vector dimensions, the scale-invariance of the algorithm is very important. We have tried in several ways, but have not yet achieved the desired results. Here we give a brief description of the methods we have tried.

A. Density Control

Our initial idea was to control the generated GNG structure by setting an error parameter, the value of which is calculated

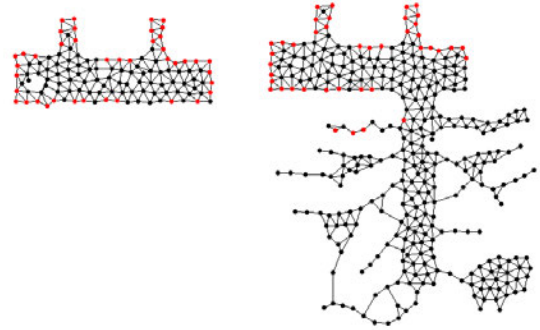
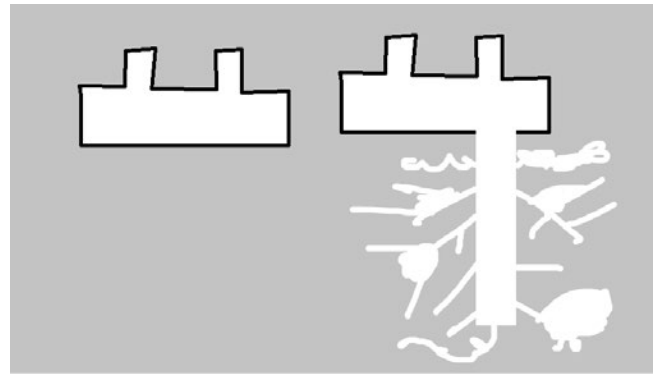


Fig. 4. A real-world condition simulation

by the following equation:

$$E = \sum_{i=0}^n I(x, y) \oplus M(x, y) \quad (1)$$

Where $I(x, y)$ stands for the pixel value at position (x, y) , and

$$M(x, y) = \begin{cases} 1, \forall n_i, |(x, y) - n_i| < (e_i/2), \\ 0, otherwise. \end{cases} \quad (2)$$

This method turns out to be useful while the shape of the input data is the same, which means it is only robust towards rotation and scaling, but shows a bad reaction to cutting and deformation. Experiments have been done between different scales while the shape of the input image remains unchanged, Fig. 5 shows the result between two different scales, where the data scale in the lower chart is 4 times the upper one. By controlling when to stop generating new nodes according to the error, we could get a similar amount of nodes as a result.

B. PCA of local point set

Another method we have tried is using PCA (Principal component analysis) to compare the similarity between local and global structures, the algorithm could be roughly summarized as follows: For a given node i , let n_i^k be the set of nodes in the k steps range. Perform a principal component analysis on the set n_i^k and set its first principal component vector to w_i^k , its

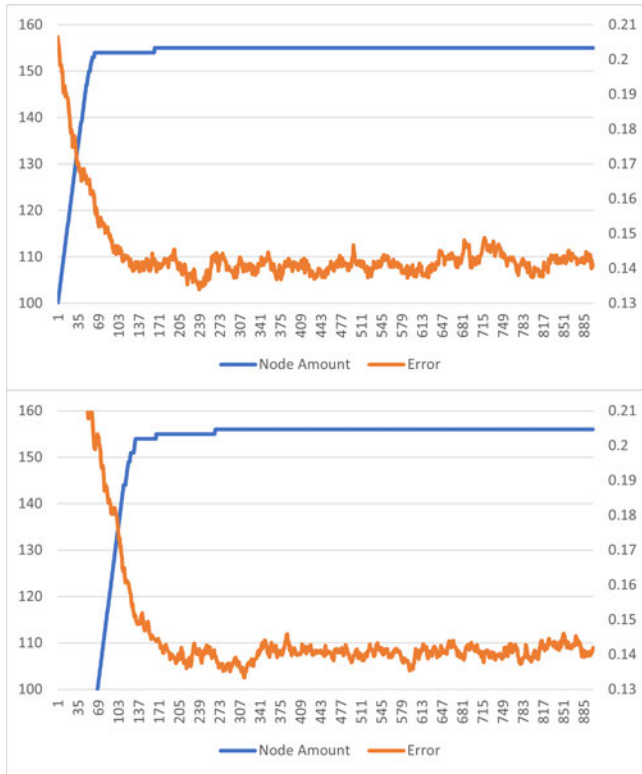


Fig. 5. Node number controlled by 'Error'

first and second eigenvalues to $\lambda_{i1}^k, \lambda_{i2}^k$ and its first principal component relatedness to

$$r_i^k = \frac{\lambda_{i1}^k}{\lambda_{i1}^k + \lambda_{i2}^k}$$

For each node, the evaluation factor k is calculated as follows. The result is shown in Fig.6) We have tried to use this method

Algorithm 1 Similarity calculation

```

k = 0
while True do
  if  $|r_i^k - r_i^0| > 0.1$  then
    break
  end if
  if  $(k + 1)dis_i^0 - dis_i^k > v_i$  then
    break
  end if
  if  $r_i^0 > 0.7$  and  $10^\circ < arccos(w_i^0 \cdot w_i^k) < 160^\circ$  then
    break
  end if
end while
    
```

to remove points that do not contribute much to the local structure, meaning that we would get fewer points in a sparse structure and more points in a dense structure. However, it proved difficult for us to reconstruct the topology between points in this process, and the new structure generated was difficult to use for matching.

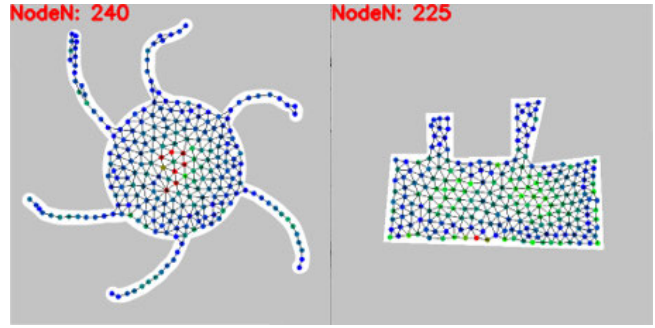


Fig. 6. Similarity between local and global structure, where redder nodes stand for higher similarity.

C. Image Descriptor

The last method was inspired by SPE(Screened Poisson Equation). We tried to build an image descriptor. Firstly we calculate the distance between the pixels and the contour and let the result be I' , then calculate $\Delta I'$. Some results are shown in Fig.7. Although we believe that this descriptor can describe the structure of shapes to some extent relatively well, we did not proceed further. The reason is mainly that it completely abandons the advantage of the fast response of GNG, and its computational effort is unacceptable in practical applications.

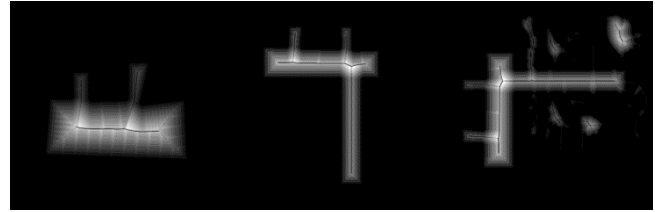


Fig. 7. Similarity between local and global structure, where redder nodes stand for higher similarity.

D. Summary of the attempt

Summarizing the research experience so far, we believe that constructing scale-space may be the most effective way to simultaneously satisfy the robustness of shearing, scaling, and deformation. In fact, the idea of scale-space is also implicit in our feature extraction algorithm, where the feature values contain both local and large-area features. Therefore, in the following research, we may try to construct a method that can assign the optimal scale to each feature point in order to realize scale invariance.

VI. CONCLUSION AND FUTURE WORK

In this paper, we proposed a method using both topological and geometrical properties of a graph to extract meaningful features and apply these features in the shape matching. Results have proved that this method is feasible. In future work, we want to apply this method in an in-door navigation based on building maps, with which we could carry out matching between the robot's local map generated by its

SLAM algorithm and However, to meet the requirements of our further work, some problems remain to be solved:

- 1) Dense of the nodes is controlled manually, an algorithm that could automatically control the density is needed.
- 2) The nodes are matched one by one, we need to consider the situation in which the numbers of the contour nodes are not the same.
- 3) The transformation matrix between the two figures, i.e., the relationship between the sample image and the image remaining to be matched is not clearly described with mathematics formulas.

REFERENCES

- [1] Toda, Yuichiro and Naoyuki Kubota. "Self-Localization Based on Multi-resolution Map for Remote Control of Multiple Mobile Robots." IEEE Transactions on Industrial Informatics 9 (2013): 1772-1781.
- [2] Y. Toda and N. Kubota. "Self-localization Based on Multi-resolution Map for Remote Control of Multiple Mobile Robots", Industrial Informatics, IEEE Transactions on, Vol.9, No.3, pp. 1772-1781, 2013.
- [3] B. Fritzke, "A growing neural gas network learns topologies," in Advances in Neural Information Processing Systems 7. MIT Press, pp. 625-632, 1995.
- [4] E. W. Dijkstra, "A note on two problems in connection with graphs." Numerische mathematik Vol.1 No.1, pp.269-271, 1959.
- [5] Mirehi, N., Tahmasbi, M. & Targhi, A.T. New graph-based features for shape recognition. Soft Comput 25, 7577-7592 (2021).
- [6] Anthony Stentz, "Optimal and Efficient Path Planning for Partially-Known Environments". The Robotics Institute; Carnegie Mellon University; Pittsburgh, PA 15213

適応共鳴理論に基づく連合クラスタリング手法の比較検討

A Comparative Study on Adaptive Resonance Theory-based Federated Clustering

○ 上田 裕也¹, 増山 直輝², 能島 裕介²

○ Yuya Ueda¹, Naoki Masuyama², Yusuke Nojima²

¹大阪府立大学

²大阪公立大学

¹Osaka Prefecture University ²Osaka Metropolitan University

適応共鳴理論に基づくマルチラベル量質混在データのための
クラスタリング手法の改良

Improvement of a Clustering Method using Adaptive Resonance Theory
for Multi-label Mixed Data

○ 西川 毅, 増山 直輝, 能島 裕介

○ Tsuyoshi Nishikawa, Naoki Masuyama, Yusuke Nojima

大阪公立大学

Osaka Metropolitan University