

第6回「コンピューターショナル・インテリジェンス研究会」

資料

期 日：2014年12月8日(月)～9日(火)

会 場：関西学院大学大阪梅田キャンパス

SICE®

主 催：計測自動制御学会 システム・情報部門

企 画：コンピューターショナル・インテリジェンス部会

共 催：日本知能情報ファジィ学会

協 賛：情報処理学会，システム制御情報学会，電子情報通信学会，
電気学会，日本神経回路学会，日本機械学会，人工知能学会，
ヒューマンインタフェース学会，進化計算学会，Japan Chapter of
IEEE Computational Intelligence Society，Japan Chapter of IEEE
Systems, Man, and Cybernetics Society，IEEE Circuits and Systems
Society Kansai Chapter

プログラム 12月8日(月)

11:00-12:30 セッション1 司会：半田 久志(近畿大学)

1. 画像センシングによる農作物の草丈推定に関する研究

○荒川 修平, 吉田 武史, 北園 淳, 小澤 誠一, 深尾 隆則, 大川 剛直(神戸大学), 村上 則幸, 辻 博之(農研機構)

2. 講義間の休憩時間における学生移動のシミュレーション —関西大学講義棟における混雑具合の検証—

○松崎 頼人, 花田 良子, 村中 徳明(関西大学)

3. ニューラルネットに基づく消費電力データの解析

○大桃 和樹, 金田 祐也, 小檜山 裕太, 趙 強福(会津大学), 多屋 優人(株式会社KDDI 研究所)

13:30-15:30 テーマセッション：ファジィとCI1 司会：関 宏理(関西学院大学)

4. ファジィ遺伝的機械学習における複数パターンを用いた新規ルール生成

○渡邊 一弘, 能島 裕介, 石渕 久生(大阪府立大学)

5. 多目的ファジィ遺伝的機械学習の並列分散実装

○高橋 佑治, 能島 裕介, 石渕 久生(大阪府立大学)

6. 段階的視覚を持つエージェントの実数値環境における追跡問題について —視覚を持ち Hunter から逃げる Prey の捕獲—

○山口 昂彦, 馬野 元秀(大阪府立大学)

7. カーレースゲームへの2つのQ表を用いたファジィQ学習の適用 —一次の目標を通過しやすい行動の学習—

○阪口 和馬, 馬野 元秀, 立野 宏樹(大阪府立大学)

15:40-17:40 フリーディスカッション 司会：黒江 康明(京都工芸繊維大学), 馬野 元秀(大阪府立大学)

「コンピューターショナル・インテリジェンス(CI)の現状とこれからの考える」

18:00-20:00 技術交流会

プログラム 12月9日(火)

11:00-12:30 テーマセッション：ファジィとCI2 司会：関 宏理（関西学院大学）

8. 全体的傾向と局所の特徴に基づく時系列データの言葉による表現 —振動を考慮した期間のファジィ集合の決定—

○岩浅 貴大, 高橋 勝稔, 馬野 元秀 (大阪府立大学)

9. 全体的傾向と局所の特徴に基づく時系列データの言葉による表現 —振動を取り出すための三角移動平均のパラメータの調整—

○島野 佳之, 高橋 勝稔, 馬野 元秀 (大阪府立大学)

10. 混合多項分布に基づくファジィ共クラスタリングにおける項目の排他的分割

○中野 剛也, 本多 克宏, 野津 亮 (大阪府立大学)

13:30-15:00 セッション2 司会：中島 智晴（大阪府立大学）

11. 探索点間の結合が探索に及ぼす影響についての検討

○酒井駿介, 畠中利治 (大阪大学)

12. 三次元点群データを利用した一般物体認識と重なった物体への対応

○細井 勇揮, 高橋 智一, 鈴木 昌人, 青柳 誠司 (関西大学)

13. 妥当性基準を用いた x-means について

○藏口 雄丈, 濱砂 幸裕 (近畿大学)

15:10-17:10 セッション3 司会：森 禎弘（京都工芸繊維大学）

14. 強化学習における word2vec を用いた転移学習

○武石 真登, 恒川 裕章, 鈴木 拓央, 濱上 知樹 (横浜国立大学)

15. 非エキスパートエージェント集団からのアンサンブル逆強化学習

○富田 真司, 恒川 裕章, 鈴木 拓央, 濱上 知樹 (横浜国立大学)

16. 帆走の強化学習において人間の操作ログを行動選択に用いる効果

○真鍋 秀朗, 橘 完太 (工学院大学)

17. フォーメーション形成問題に対する学習の確実性を向上させた群強化学習法とその評価

○飯間 等, 黒江 康明 (京都工芸繊維大学)

17:20-18:50 セッション4 司会：畠中利治（大阪大学）

18. 制約ペトリネット内でトークン進化が満たすべき必要な条件

○鈴木 秀明 (情報通信研究機構), 巖佐 庸 (九州大学)

19. ビジュアルサーボによる水中ロボットの位置・姿勢制御

○矢納 陽, 大西 祥太, 米森 健太, 石山 新太郎 (岡山大学), 藤本 勝樹 (広和株式

会社), 見浪護 (岡山大学)

20. セパトリクスを実現する発現パターンに基づく遺伝子ネットワークの設計 —方法と解の存在性—

○森 禎弘, 黒江 康明 (京都工芸繊維大学)

19:00 表彰式, 閉会

フリーディスカッション

テーマ：コンピューターショナル・インテリジェンス（CI）の現状とこれからの考える

企画者：黒江 康明（京都工芸繊維大学）

馬野 元秀（大阪府立大学）

村田 純一（九州大学）

関 宏理（関西学院大学）

画像センシングによる農作物の草丈推定に関する研究

○荒川修平 吉田武史 北園淳 小澤誠一 深尾隆則 大川剛直 (神戸大学)
村上則幸 辻博之 (農研機構)

A Study on the Estimation of Plant Height Using Image Sensing

*S. Arakawa, T. Yoshida, J. Kitazono, S. Ozawa, T. Fukao, T. Okawa (Kobe University),
N. Murakami and H. Tsuji (National Agriculture and Food Research Organization)

Abstract— In this work, we propose an image sensing method to estimate the height of agricultural plants. In the proposed method, several images are taken by moving a digital camera attached to a single-axis robot. Then, the two consecutive images with a plant tip are automatically selected and the plant height is estimated from the two images based on the triangulation. Here, plant tips are detected using SIFT keypoints. In the experiment, we estimate the heights of plants with several backgrounds. When the shape of plants and the backgrounds are relatively simple, the proposed image sensing method can measure plant height accurately. However, when plants and background have complex structure, the proposed method could have large estimation errors mainly due to the misdetection of plant tips.

Key Words: smart agriculture, image sensing, plant height estimation

1 はじめに

近年、我が国の農業は農業従事者の高齢化や後継者不足といった問題を抱えており、農作業の省力・軽労化が求められるようになってきた。他方でロボット技術やICT (Information and Communications Technology, 情報通信技術) などのはめざましい発展を遂げており、農業への活用も期待されている。こうした中で注目されはじめたのが、ICTを用いた農業技術「スマート農業」である。オランダでは、スマート農業を導入することにより少ない人口・小さい国土でありながら、農産物輸出額世界第2位となるまで成長している¹⁾。

画像センシングは、スマート農業のための有力なツールとして期待されており、野菜や果実、穀物などの選別²⁾であったり、ハイパースペクトルカメラを用いた画像センシングによる多くの植物・生態情報の取得³⁾等の研究が行なわれ、その有効性が示されてきた。その他、村上ら⁴⁾は画像センシング技術を用いて木になったままのナシの果実を計数することに成功し、また佐藤ら⁵⁾は単位面積あたりの植被領域の占有率を表す植被率が葉面積や窒素の含有率と相関が高いことを報告している等、画像センシングは様々な農作物の生育情報抽出手法として用いられている。

本研究では、農作物の様々な生育情報を画像センシングにより自動抽出し、収量との相関を調べることでより効率的な農作業を低コストで可能にすることを目的とし、その一環として農作物の草丈を画像から推定する手法を画像の撮影方法を含め提案する。

具体的には、単軸ロボットにカメラを取り付けることで等間隔での撮影が可能となり、それらの画像からSIFT特徴点抽出により被写体の頂点を求め、ピンホールカメラモデルの原理を利用し高さを推定する。しかし、実際の画像では背景からも特徴点が抽出されるため、1枚の画像からでは頂点を求めることは困難である。そこで、複数枚の画像から対応点探索を行ない、その移動量に基づき奥行ごとに特徴点を分類し頂点を求める。

実験では、複数種類の植物を用いて提案手法の有効

性を検証する。また、時刻や場所などの撮影状況を変えて検証を行い、日照条件や対象物の背景が推定結果に与える影響について議論する。

2 SIFT

Scale-Invariant Feature Transform (SIFT) はLowe⁶⁾⁷⁾によって提案された、画像中の特徴的な点(キーポイント)を検出しその特徴量を記述する手法であり、スケール変化、回転変化、輝度変化に対して不変である。

キーポイント検出はガウシアンピラミッドを用いた極値検出とキーポイントの絞り込みからなり、特徴記述はオリエンテーション割り当てによる方向の正規化と特徴ベクトルの作成からなる。

2.1 スケールスペース極値検出

スケール変化に対する不変性を得るため、スケール σ を考慮した画像空間を構築し、その中からキーポイント候補となる極値を検出する。

以下の式は位置 (x, y, σ) におけるスケール σ のガウス関数 $G(x, y, \sigma)$ と、画像 $I(x, y)$ と $G(x, y, \sigma)$ を畳み込んだ平滑化画像 $L(x, y, \sigma)$ を表す。

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} \exp\left(\frac{-(x^2 + y^2)}{2\sigma^2}\right) \quad (1)$$

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (2)$$

異なるスケールの平滑化画像の差分 $D(x, y, \sigma)$ をDoG (Difference-of-Gaussian) といひ (Fig. 1), 次式で表す。ただし、 k はスケールの増加率である。

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (3)$$

極値探索はDoG画像 $D(x, y, \sigma)$ から行い (Fig. 2), 注目画素が26近傍領域において極値となる場合キーポイント候補とする。

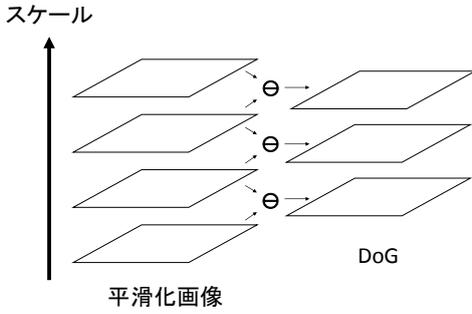


Fig. 1: Difference-of-Gaussian.

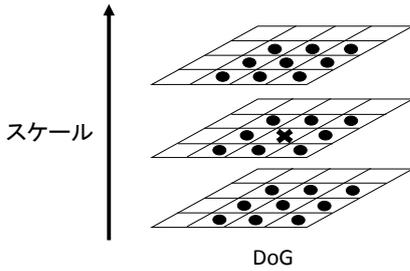


Fig. 2: Local extrema detection.

2.2 キーポイントのローカライズ

検出されたキーポイント候補点にはエッジ上の点やノイズも含まれるため、そのような点を削除し、より頑健な点をキーポイントとして選択する。

エッジ上の点は、DoG 出力値の二回偏微分 $D_{xx} = \frac{\partial^2 D}{\partial x^2}$, $D_{xy} = \frac{\partial^2 D}{\partial y \partial x}$, $D_{yy} = \frac{\partial^2 D}{\partial y^2}$, $D_{yx} = \frac{\partial^2 D}{\partial x \partial y}$ を計算して得られるヘッセ行列 \mathbf{H} から主曲率を算出して求める。以下にその式を示す。

$$\mathbf{H} = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{yx} & D_{yy} \end{bmatrix} \quad (4)$$

$$\frac{\text{Tr}(\mathbf{H})^2}{\det(\mathbf{H})} = \frac{(\lambda_1 + \lambda_2)^2}{\lambda_1 \lambda_2} = \frac{(\gamma + 1)^2}{\gamma} \quad (5)$$

ただし、 λ_1, λ_2 は \mathbf{H} の固有値 ($\lambda_1 > \lambda_2$) であり、 γ は、 λ_1 と λ_2 の比率 ($\lambda_1 = \gamma \lambda_2$) である。閾値 γ_{th} を設定し、以下の条件を満たす点をキーポイント候補とする。

$$\frac{\text{Tr}(\mathbf{H})^2}{\det(\mathbf{H})} < \frac{(\gamma_{th} + 1)^2}{\gamma_{th}}$$

また、DoG の出力値が小さい点はノイズの影響を受けやすいので、キーポイント候補点のサブピクセル位置での DoG 出力値の絶対値が閾値未満となる点を削除し、残った点をキーポイントとして選択する。

2.3 オリエンテーションの割り当て

回転不変性を得るため、検出された各キーポイントに対しオリエンテーションを割り当てる。キーポイントが検出された平滑化画像の各画素の勾配 $m(x, y)$ とその方向 $\theta(x, y)$ を以下の式より求める。

$$f_x(x, y) = L(x + 1, y) - L(x - 1, y)$$

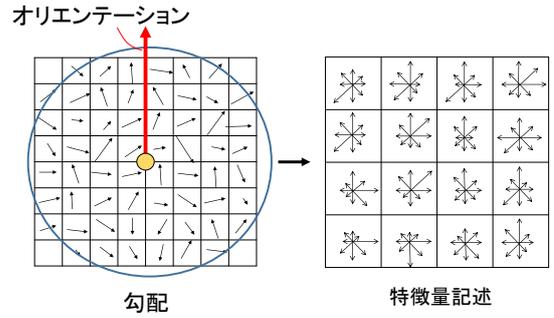


Fig. 3: A keypoint descriptor.

$$f_y(x, y) = L(x, y + 1) - L(x, y - 1)$$

$$m(x, y) = \sqrt{f_x(x, y)^2 + f_y(x, y)^2} \quad (6)$$

$$\theta(x, y) = \tan^{-1} \frac{f_y(x, y)}{f_x(x, y)} \quad (7)$$

m, θ から 36 方向に分割した重み付ヒストグラムを作成し、最大値の 80% 以上となる方向をキーポイントのオリエンテーションとして割り当てる。80% 以上となる方向が複数ある場合は、それらすべてをキーポイントのオリエンテーションとして割り当てる。

2.4 特徴量の記述

キーポイントの周辺領域をオリエンテーション方向に回転させて向きの正規化を行ない、領域ごとに方向ヒストグラムを作成する。さらに、領域を 4×4 の 16 に分割し、それぞれに 8 方向のヒストグラムを作成する (Fig. 3)。特徴量としては、 $4 \times 4 \times 8 = 128$ 次元のベクトルが得られる。

2.5 特徴マッチング

SIFT により得られた各キーポイントの特徴量を比較し、画像間の対応点を探索する。

画像 I から得られたキーポイント p_I の特徴量は 128 次元のベクトル $\mathbf{g}^{p_I} = (g_0^{p_I}, \dots, g_{127}^{p_I})$ で表される。画像 I_i, I_{i+1} 中のキーポイント $p_{I_i}, p_{I_{i+1}}$ の特徴量をそれぞれ $\mathbf{g}^{p_{I_i}}, \mathbf{g}^{p_{I_{i+1}}}$ としたとき、特徴量間のユークリッド距離 $d(\mathbf{g}^{p_{I_i}}, \mathbf{g}^{p_{I_{i+1}}})$ を次式にて求める。

$$d(\mathbf{g}^{p_{I_i}}, \mathbf{g}^{p_{I_{i+1}}}) = \sqrt{\sum_{j=0}^{127} (g_j^{p_{I_i}} - g_j^{p_{I_{i+1}}})^2} \quad (8)$$

Lowe⁷⁾ の手法では、あるキーポイント 1 点に対し、異なる画像中の全てのキーポイントとのユークリッド距離 d を算出し、最小となる点を対応点として選択する。このとき、最小値の次に小さい値に対する比率が一定以上となる組合せを棄却し、より効果的にマッチングを行なうことが可能となる。

3 SIFT 特徴量を用いた物体の高さ推定

連続画像に対して SIFT 特徴量を用いた対応点探索を行い、その移動量に応じてキーポイントを分類し、対象物体の特定及び高さ推定を行なう。



Fig. 4: Single-axis robot.

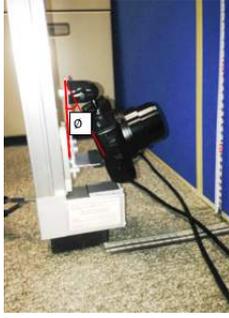


Fig. 5: Camera and the angle of installation.

3.1 高さ推定に用いる画像

本研究では、以下の方法で撮影された連続画像を用いることで、物体の高さ推定を可能にする。

Fig. 4はカメラを取り付ける単軸ロボットであり、初期位置 H_0 から $H_n (= H_0 + 1000)$ [mm] までカメラを等速で移動させることにより、等間隔 l で n 枚の連続画像を撮影する。 i 番目の画像の撮影位置 H_i は次式により求める。

$$H_i = H_0 + i \cdot l \quad (9)$$

また、茎など内部の様子を撮影する際、垂れ下がる農作物の葉に遮られにくくするため撮影は斜め下から行ない (Fig. 5)、カメラの取り付け角度を ϕ とする。

3.2 移動量に基づくキーポイント分類

特徴マッチングを行った後、画像 I_i, I_{i+1} 中の各対応点 $k_{I_i} = (u_{k_{I_i}}, v_{k_{I_i}})$, $k_{I_{i+1}} = (u_{k_{I_{i+1}}}, v_{k_{I_{i+1}}})$ に対し、画像空間内での水平方向・鉛直方向それぞれの移動距離 $\Delta u, \Delta v$ が次の条件を全て満たした場合に対象物体上のキーポイント候補とする。

- $|\Delta u| = |u_{k_{I_{i+1}}} - u_{k_{I_i}}| < \theta_u$
- $\Delta v = v_{k_{I_{i+1}}} - v_{k_{I_i}} < 0$
- $\theta_v^{low} < |\Delta v| < \theta_v^{high}$

ただし、 $\theta_u, \theta_v^{low}, \theta_v^{high}$ は水平方向・鉛直方向の閾値であり、実験では経験的に設定した。

被写体は静止していることを前提としているため、一つ目の条件を満たさないような、カメラの進行方向に垂直な方向に大きく動く対応点は誤対応とみなし、また二つ目の条件を満たさない対応点は、カメラの進行方向に向かってカメラより大きく動いていることになり、これについても同様に誤対応とみなし棄却する。さらに三つ目の条件により、動きの小さい対応点はカメ

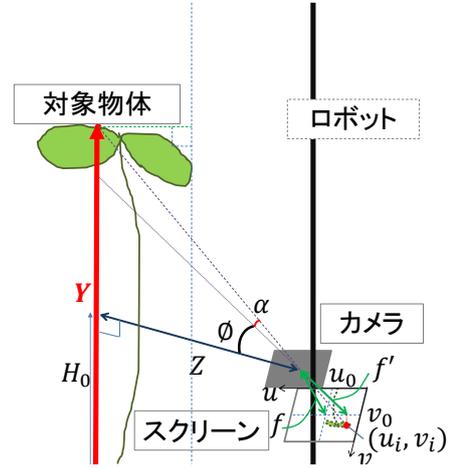


Fig. 6: Calculating the height.

ラから距離が一定以上離れている、すなわち背景であるとして棄却するとともに、あまりにも動きの大きいものを誤対応とする。

3.3 特徴点追跡

マッチングの誤対応削減のため、3枚の画像を用いてより信頼性の高いキーポイントのみを残す操作を行なう。具体的には、画像 I_i, I_{i+1} 間で残ったキーポイントと、画像 I_{i-1}, I_i 間で残ったキーポイントとの間で再度対応点探索及び3.2節の分類を行なう。ただし、閾値 θ_u は1回目の分類で用いた閾値 θ_u 以下の値をとるよう設定する。このようにして残ったキーポイントを高さ推定に用いる。画像 I_{i-1}, I_i 間でキーポイントが残らない場合、画像 I_i での高さ推定はスキップする。

3.4 高さ推定

画像空間内の座標を (u, v) としたとき、画像 I_i 中で残ったキーポイントのうち、 v が最大となる点とそれに対する画像 I_{i+1} 中での対応点を選択し、それぞれの座標を $(u_i, v_i), (u_{i+1}, v_{i+1})$ とする。

画像中心の座標を (u_0, v_0) とし、ピンホールカメラモデルの原理を利用し以下の式より高さ Y_i を推定する (Fig. 6)。

$$\begin{aligned} f' &= \sqrt{((u_i - u_0) \cdot s_u)^2 + f^2} \\ &= \sqrt{\left(\frac{(u_i - u_0)}{f_u}\right)^2 + 1} \cdot f^2 \end{aligned} \quad (10)$$

$$\begin{aligned} \alpha &= \tan^{-1} \left(\frac{(v_i - v_0) \cdot s_v}{f'} \right) \\ &= \tan^{-1} \left(\frac{(v_i - v_0)}{\sqrt{\left(\frac{(u_i - u_0)}{f_u}\right)^2 + 1} \cdot f_v^2} \right) \end{aligned} \quad (11)$$

$$Y_i = H_i + Z_i \cdot \tan(\phi + \alpha) \quad (12)$$

ここで、 ϕ, H_i は実測値から得られる既知の値であり、カメラの水平方向・鉛直方向の焦点距離 f_u, f_v は、焦点距離 f 、水平方向・鉛直方向の画素幅 s_u, s_v を用いて、

$$f_u = f/s_u$$

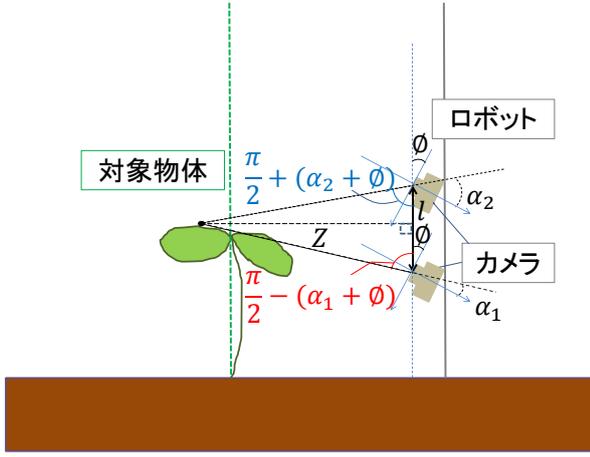


Fig. 7: Triangulation.

$$f_v = f/s_v$$

で表され、カメラキャリブレーション⁸⁾によってカメラの内部パラメータを推定することで得られる。

また、キーポイントの奥行 Z_i は、三角測量の原理を用いて次式により求められる (Fig. 7)。

$$Z_i = \frac{1}{\frac{1}{\tan(\frac{\pi}{2} - (\alpha_i + \phi))} + \frac{1}{\tan(\frac{\pi}{2} - (\alpha_{i+1} + \phi))}} \quad (13)$$

ただし、キーポイント数が0となった場合この操作は行わない。こうして求めた $Y_i (i = 1, \dots, n-1)$ のうち、 i が最大となるものを推定値 E とする。

4 実験

数種類の植物画像に対し、様々な背景パターンを用いて提案手法の有効性を検証する。

4.1 実験方法

3.1節で述べたように、等間隔で撮影された30枚の連続画像($n = 30$)を用いて高さ推定を行う。ただし、対象物体は単独で映っているものとする。このとき、式(9), (12), (13)の ϕ, H_0, l は実測値に基づき、以下の値を用いる。

$$\phi = \tan^{-1} \frac{1}{3} [\text{rad}]$$

$$H_0 = 160 [\text{mm}]$$

$$l = \frac{1000}{30} [\text{mm}]$$

まず、背景から特徴点が検出されにくい環境において撮影された植物 (Object 1, 高さ $L = 1050$ [mm]) の画像を用い、提案手法の推定精度を調べる。

次に、先程用いた植物を含む4種類の植物 (Fig. 8) に対し4方向 (Object X に対し Direction A_X, B_X, C_X, D_X) から撮影を行った画像を用い、撮影方向及び背景の変化に対する有効性を調べる。それぞれの高さ (対象物体に奥行がある場合は平均値) の真値 L は実測値を用いる。また、Object 1 ~ 3 に関しては、屋外でAM11, PM2, PM4の3回撮影を行い、照明の影響に対する有効性も調べる。

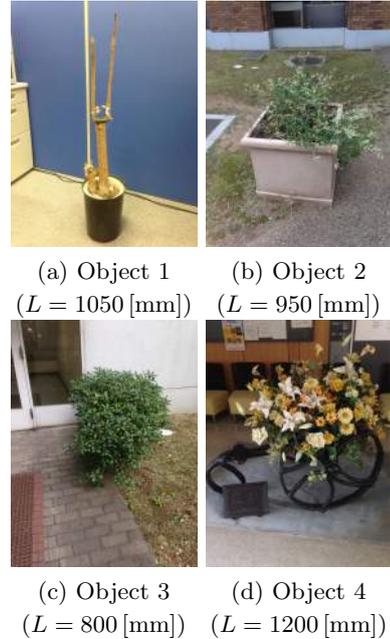


Fig. 8: Sample Objects and their heights.

Table 1: Error between the true value and the estimated value.

L [mm]	E [mm]	ϵ [mm]
1050	1041.5	8.5

4.2 実験結果

Table 1にObject 1の高さ推定値 E と真値 L との誤差 $\epsilon = |E - L|$ を示す。Fig. 9はそのときの結果画像であり、画像 I_i (右)と画像 I_{i+1} (左)を横に連結し、対応が取れたキーポイント同士を線で結んでいる。赤い円は検出されたすべてのキーポイントのスケールを表す。対象物体の頂点部分がうまく検出されており、誤差も小さく提案手法でほぼ正確に推定できた。

次に、Table 2に4種類の植物の高さ推定値を示す。赤色太字になっている値は、推定に用いたキーポイントが対象物体上にある場合 (推定成功) であり、そうでないものは、誤対応のため背景上のキーポイントを推定に用いた場合 (推定失敗) である。また、Table 3に各対象物体の推定成功率 $q = \frac{\text{推定成功回数}}{\text{全撮影回数}} \times 100$ [%]を示すが、環境によって成功率は大きく変化する結果となった。

Fig. 10に推定が成功した際の結果の例を示す。Object 1~4それぞれで、他仇市区頂点を検出できている

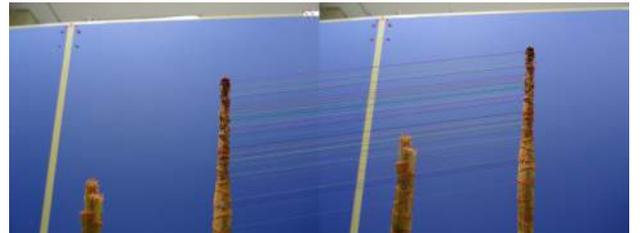


Fig. 9: Corresponding keypoints.

Table 2: Estimated value and the error of various plants.

(a) Object 1 ($L = 1050$ [mm]).				
Direction	A_1	B_1	C_1	D_1
E [mm](AM11)	1001.3	1042.5	1035.2	1048.4
ϵ [mm]	48.7	7.5	14.8	1.6
E [mm](PM2)	1017.2	1035.6	1036.9	1030.2
ϵ [mm]	32.9	14.4	13.1	19.8
E [mm](PM4)	1021.0	1038.3	1035.8	1043.1
ϵ [mm]	29.0	11.7	14.2	6.9
(b) Object 2 ($L = 950$ [mm]).				
Direction	A_2	B_2	C_2	D_2
E [mm](AM11)	896.9	1169.2	912.3	1244.0
ϵ [mm]	53.1	219.2	37.72	293.9
E [mm](PM2)	997.3	891.9	943.4	1029.9
ϵ [mm]	47.3	58.1	6.6	79.9
E [mm](PM4)	989.7	919.6	892.6	922.6
ϵ [mm]	39.7	30.4	57.4	27.4
(c) Object 3 ($L = 800$ [mm]).				
Direction	A_3	B_3	C_3	D_3
E [mm](AM11)	1073.0	1017.1	807.0	1177.9
ϵ [mm]	273.0	217.1	7.0	377.9
E [mm](PM2)	1026.1	1003.8	1143.5	1162.6
ϵ [mm]	226.1	203.8	343.5	362.6
E [mm](PM4)	1111.6	945.6	1209.6	1117.0
ϵ [mm]	311.6	145.6	409.6	317.0
(d) Object 4 ($L = 1200$ [mm]).				
Direction	A_4	B_4	C_4	D_4
E [mm]	1184.3	1114.0	1101.6	1175.5
ϵ [mm]	15.7	86.0	98.4	24.4

Table 3: The rate of success.

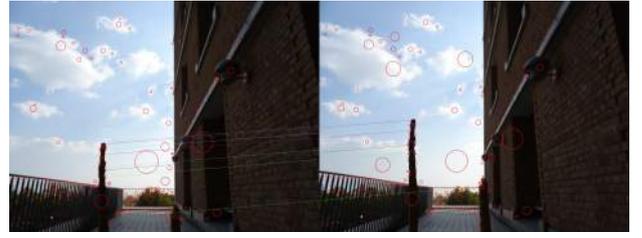
	Object 1	Object 2	Object 3	Object 4
q [%]	100	58.3	8.3	50

ことがわかる。

Object 1 では、撮影時刻と撮影方向において、真の高さを高い精度で推定できている。異なる撮影時刻で日照条件を変化させても頂点付近の対応点を抽出するのあまり影響は出ていない (Fig. 11)。

以下は誤対応による失敗例であり、結果画像を示しつつ原因を考察する。Fig. 12, Fig. 13 は 3 回の撮影のうち、2 回以上失敗した地点の例である。Fig. 12 の様な場合に共通する傾向として、スケールの小さな細かい点が多数検出され、その中で誤検出が生じている。式 (3) の DoG 出力値はスケール σ に比例して小さくなるためノイズの影響を受けやすく誤検出が生じやすい。

次に、Fig. 13 はいずれも背景に似たような形のタイルが多く使用された建物が映りこんでいる。このような場合、誤対応であっても 3.2 節で述べた条件を通過してしまうことが多くなり、結果失敗率が高くなると考えられる。しかし、Object 2 の Direction B_2 も同様の背景もつ (Fig. 14) が、成功率が高くなった。これは背景までの距離が大きくなったことでスケールが小さくなった結果、2.2 節のアルゴリズムによりキーポイント候補が削除されたためだと考えられる。



(a) Object 1 (Direction D_1 , AM11).



(b) Object 2 (Direction C_2 , AM11).



(c) Object 3 (Direction C_3 , AM11).



(d) Object 4 (Direction D_4)

Fig. 10: Difference of objects.

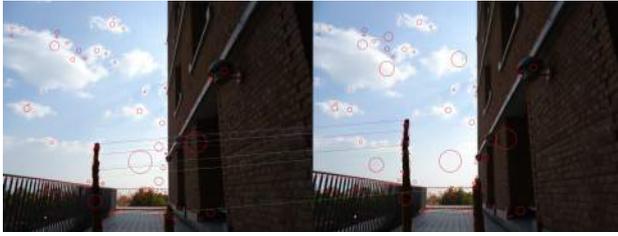
5 まとめ

本研究では単軸ロボットを用いた撮影により得られた連続画像から、画像センシングにより農作物の草丈を推定する手法を提案した。

提案手法は、対象物体や背景が単純なものに対しては誤差 10 [mm] 程度の精度で推定した。一方、対象物体や背景が複雑になると推定精度が低下する結果となった。精度が低下する主な要因は特徴点の誤対応であるが、小さいスケールの特徴点が多く検出される場合や背景にタイル張りの建物のような鉛直方向に似た特徴点が多く含まれる場合に起こりやすく、また背景との距離が小さい場合にこれらが多く検出されると考えられる。

今後、ノイズやエッジ上の点を可能な限り削除し、背景との距離が小さい場合においても誤対応が生じにくくかつ対象物体の頂点部分の特徴点はうまく検出できる最適な特徴検出手法及びパラメータを決定するよう改善する必要がある。

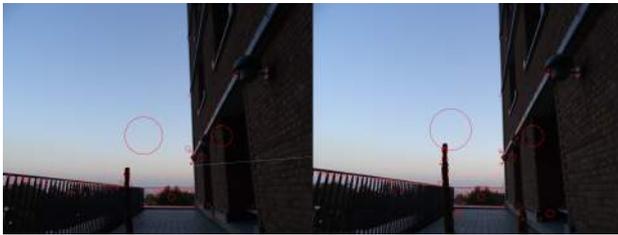
また、SFM(Structure from Motion)⁹⁾等のアルゴリ



(a) Object 1 (Direction D_1 , AM11).



(b) Object 1 (Direction D_1 , PM2).



(c) Object 1 (Direction D_1 , PM4).

Fig. 11: Effect of lighting conditions.



(a) Object 2 (Direction A_2 , PM2).



(b) Object 3 (Direction A_3 , AM11).

Fig. 12: Effect of small-scale noise.



(a) Object 2 (Direction D_2 , PM2).



(b) Object 3 (Direction D_3 , AM11).

Fig. 13: Effect of tiled patterns.



Fig. 14: Effect of depth of background.

- 4) 村上則幸, 吉田睦, 吉田武史: ナシ計数のための画像処理手法の研究, システム制御情報学会研究発表講演会講演論文集, **57** (2013)
- 5) 佐藤勉, 林恒夫, 岩田忠寿: 画像解析を利用した水稻の生育診断技術-2-葉面積指数及び植被率を指標とした生育診断と制御技術, 福井県農業試験場報告, 19/26 (1993)
- 6) D. G. Lowe: Object recognition from local scale-invariant features, International Conference on Computer Vision, 1150/1157 (1999)
- 7) D. G. Lowe: Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision, **60-2**, 91/110 (2004)
- 8) Z. Zhang: A flexible new technique for camera calibration, Pattern Analysis and Machine, **22-11**, 1330/1334 (2000)
- 9) C. Tomasi and T. Kanade: Shape and motion from image streams under orthography: a factorization method, International Journal of Computer Vision, **9-2**, 137/154 (1992)

ズームを実装し、カメラ位置等も自動抽出することも視野に入れる予定である。

参考文献

- 1) FAOstat, Food. "agriculture organization of the United Nations." Statistical database (2013).
- 2) 石川豊: ポストハーベストにおける画像計測の応用, 計測と制御, **37-2**, 91/94 (1998)
- 3) 大政謙次: プレシジョン・アグリカルチャーのための画像センシング, 農業情報研究, **11-3**, 213/229 (2002)

講義間の休憩時間における学生移動のシミュレーション — 関西大学講義棟における混雑具合の検証 —

○松崎頼人 花田良子 村中徳明 (関西大学)

Discussion of Crowd in Students' Flow during Intermission in Kansai University

*R. Matsuzaki, Y. Hanada and N. Muranaka (Kansai University)

Abstract— 平日の大学キャンパス内においては、多くの学生が講義棟内を往来するために群集が発生する。特に、休憩時間中の講義棟内の廊下や階段では学生の混雑が見られる。群集事故は主に混雑に由来するものであり、その危険性を低減するためには、過密状態を緩和する措置を議論するべきである。本研究では、学生の移動シミュレーションに基づいて、休憩時間中の教室間移動において群集事故が発生しやすい場所を特定する。また、それらの場所の学生の混雑度合いの時間的な推移から、混雑緩和の方策を検討することを目的とする。本稿では、関西大学の理工系学部の講義棟を対象としたモデル化および学生の移動シミュレーションの指針について述べる。

Key Words: 混雑シミュレーション, 大学講義棟

1 はじめに

古くから人々が盛んに利用する場所では、人々が特定の目的や指向性を持って集合するため、混雑が発生する。例えば、百貨店のバーゲン売り場を目指す移動や、災害時に建物内から屋外へ避難する際に出口を目指す移動などが挙げられる。このような状況では、他人との距離が切迫し、狭い範囲に多くの人が存在することになるため、各人が自由に動ける範囲が制限される。あるいは、歩行者の妨げになるような障害物が存在する場合、多方向からの群集同士の対流によって群集間の接触面に強い力が集まる場合には、事故が発生する可能性がある。群集によって発生する事故のことを群集事故と呼び、古くは約200年前に江戸の永代橋の崩落事故、近年では明石の歩道橋事故が発生し、多くの死傷者を出している。

群集事故の発生確率を減らすため、事故要因や群集内の特徴を分析し、理解することは重要である。群集の行動を分析する一般的な手法に、マルチエージェントシミュレーションがある。マルチエージェントシミュレーションでは群集内の個々のエージェントを適切にモデル化することで、現実空間における様々な環境を仮想的に表現する。これは、実験場所や参加者の準備によるコストおよび被験者に対する安全性への配慮の観点から、現実空間での実験による再現や検証が困難な場合でも短期間かつ手軽に群集の特徴を分析できることを指す。シミュレーションのシナリオとして、火災¹⁾、地震²⁾、津波³⁾などの自然災害からの避難に関する議論が多くなされている。また、避難以外では日常における購買客の移動の流れ⁴⁾や、鉄道の駅における乗客の流動⁵⁾などが取り上げられることもある。

本稿では、学生の移動に焦点を当てる。大学キャンパス内において、平日の休憩時間での講義棟では、非常に多くの学生が教室間を移動しており、廊下や階段において群集が発生する。このとき、群集事故を起こさないようにするためには、講義棟内で混雑する場所の評価を行い、そのリスクを分析することが不可欠である。大学講義棟内での群集混雑を考慮した教室間の移動に関する議論には、前川らによる時間割および教室配置の最適化のための教室間移動の混雑低減を図る

研究⁶⁾がある。また、災害時における大学講義棟の避難に関する議論には、北島らによる避難誘導による避難効率の影響を図る研究⁷⁾がある。

本研究では、関西大学の理工学系講義棟を対象に、平日休憩時間における学生の教室間移動による混雑をシミュレーションによって評価する。群集移動のシミュレータを構築するには、学生の移動則、移動空間のモデルを定義する必要がある。群集の移動はマルチエージェントシミュレーションにおいて記述され、学生は個々のエージェントとして扱われる。マルチエージェントシミュレーションにおける代表的な移動モデルは、2次元連続空間モデル、セルオートマトンモデル、ネットワークモデルの3つに分類される⁸⁾。2次元連続空間モデルは、1) 歩行者が2次元平面上で他の歩行者や障害物を回避したり、2) パーソナルスペースを確保したり、3) 目的方向への移動方向、移動距離や移動速度を算出している。そのため、エージェントの増加に伴って、計算量が膨大になる問題もある。セルオートマトンモデルは、歩行者が歩行可能な空間を格子状に分割しており、移動方向は制限される。歩行者（エージェント）は目的地方向の空のセルへ一定確率で移動したり、過去の歩行者の移動傾向を模倣するというような比較的単純なアルゴリズムが採用されている。ネットワークモデルは、歩行者の移動可能範囲が長さや幅を持ったリンクで表現され、部屋、廊下、階段などもリンクとして扱われる。歩行者（エージェント）の移動はノード間のリンク接続に依存しており、移動先の設定は柔軟に対応できるが、接続数が多くなると計算量が増加する。それぞれのモデルは、長所・短所が存在するため、シミュレーションの条件によって適切なモデルを利用する必要がある。

本研究では、講義終了と同時に学生が一斉に教室間移動を行うというシナリオのもとで、2次元連続空間モデルによるシミュレーションを構築し、大学講義棟内の混雑について解析・検討する。本シミュレーションでは、2次元連続空間モデルの中でもエージェントを粒子に見立て、エージェントや障害物に正の電荷、目的的に負の電荷を与えてクーロンの法則に基づきエージェントが目的地を目指すポテンシャルモデルを採用

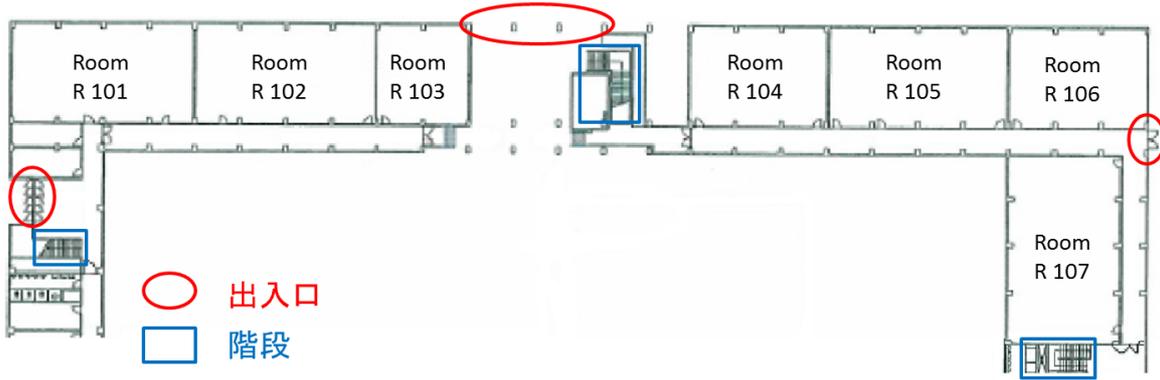


Fig. 1: The structure on the first floor in the school building.

Table 1: The number of the student that moves to the next class in each break time.

	月	火	水	木	金	土
1-2	2,247	2,324	2,187	1,915	1,971	799
3-4	2,202	1,689	2,015	1,810	1,780	810
4-5	1,506	1,494	809	1,232	1,256	382
5-6	777	588	324	774	658	64

する⁵⁾。学生の移動モデルについて、休憩時間における講義棟内での移動則および目的地は、実際の講義の履修データをもとに設定する。

2 大学講義棟の現状とそのモデル化

2.1 大学講義棟の現状

理工系講義棟では、主に理工系3学部を対象とした講義が行われている。さらに、理工系3学部以外の学部向けの講義も行われている。そのため、理工系講義棟には常時、人が多く集まり、混雑が発生しやすい状態におかれている。2014年度の理工系3学部の在籍者数は5,279人で、1学年あたり約1,300人である。

本研究では用いる学生のデータはすべて履修データをもとにしており、対象となる講義棟での履修データは全部で33,535件である。履修データには、学生を表すIDと履修している講義名、講義が行われる曜日と時限の情報がある。Table 1は、履修データから得られた2014年春学期における各休憩時間ごとの教室移動の総量である。2限と3限間のデータが表にないのは、昼休みで50分間の休憩時間があり、急ぐ必要のある教室移動とは異なるためである。

理工系講義棟は全部で5階建てとなっており、教室の数は全部で38室（授業を行っているのは31室）である。講義棟と外部との出入りは、1階の中央と両端の3箇所から可能である。各階間を移動する階段は各階の中央および両端に存在しており、エレベータは存在しない。Fig. 1に講義棟1階の構造図を示す。基本的に他の階も1階と同じ構造であるが、5階のみ構造が異なっている。5階の構造は通路が1本で繋がっており、教室間で直接移動可能であるなどの差異がある。

実際の教室移動の際には、特に中央の階段で混雑が発生している。講義棟1階の中央出口の目前には、生

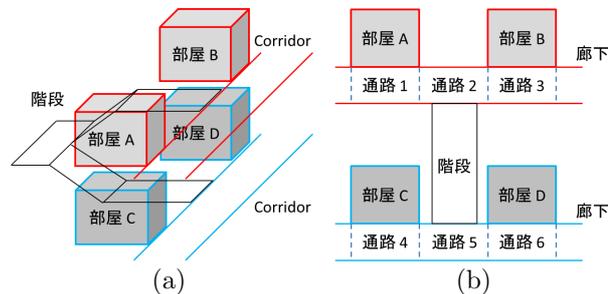


Fig. 2: Example of structure of a floor in school building.

協の食堂や自習が可能なスペースを備えた施設が存在する。そのため、一部の学生は休憩時間にこれらのスペースに移動し、講義が始まる前に戻ってくる姿が見られた。また、昼休みにはほとんどの学生が講義棟外へ出るため、群集の避難に似た動きが見られる。

2.2 大学講義棟のモデル化

本研究では、マルチエージェントシミュレーションを用いて理工系講義棟における学生の教室間移動による混在状況を解析することを目標とする。エージェントの移動モデルには、2次元空間上を自由に移動可能なモデルを採用する。ただし、本モデルは一本道のような単純な形状の領域には適しているが、大学講義棟のように教室間の移動に複数の経路が存在する場合には工夫が必要である。教室間の移動を実現するには、個々のエージェントに対し、ある教室から別の教室までの移動経路を設定する必要がある。そこで、複数ある教室間の移動経路を簡潔に表現するため、本モデルでは経路選択にのみネットワークを利用する。Fig. 2に建築物の立体構造を平面上に抽象化する流れを示す。Fig. 2(a)のような立体構造をモデル化するには、Fig. 2(b)のように平面上に簡略化する。このとき、廊下を教室や階段の壁の位置で区切り通路として分割することで、ネットワークを用いて簡単に経路を選択できるようにする。本研究の対象である理工系講義棟内には、1階から5階までの各階につながる階段が存在するが、各階をつなぐ階段（1-2階、2-3階、3-4階、4-5階の間）はそれぞれ独立して存在していると考えられる。

本モデルでは、講義棟の構造を教室、講義棟出入口、通路（廊下）、階段に分類し、これらを1つの区画と



Fig. 3: Example of gates.

してそれぞれ空間を表現する．特に，各階の通路は部分的な区画を複数連結することで，各階の廊下全体を Fig. 2 (b) のように分割されたように表現する．このとき，教室や講義棟出入口のドア部分および分割された廊下間の接続部分は通行できるようにする．教室間移動時の移動経路選択において，教室や講義棟出入口は開始地点あるいは目的地点，通路や階段はエージェントの移動経路の通過地点としての役割を持っている．

3 教室間移動のシミュレーション

本研究では，教室間移動のシミュレーションを行うため，講義棟内の構造モデルおよび学生の移動モデル，移動経路の選択モデルをそれぞれ構築する．講義棟内の構造モデルでは，建物内の各空間の構造について設定し，空間内を移動するエージェント（学生）の定義を行う．移動モデルでは，各エージェントの移動則をポテンシャルモデルによって速度ベクトルを決定し，単位時間ごとの位置座標を求める．なお，ポテンシャルモデルの各定式には，上水流らの研究⁵⁾を参考している．経路選択モデルでは，各エージェントの教室間移動における目的地を設定し，ネットワークを用いて教室間移動を行うための経路を選択する．

3.1 講義棟内における構造モデルの定義

3.1.1 講義棟のパラメータ

講義棟の構造パラメータは，教室間移動のシミュレーションにおける学生の移動に制約を与える役割を持つ．2章でも述べたとおり，教室，講義棟出入口，通路（廊下），階段の各空間を区画で定義し，複数の通路を組み合わせることで各階の廊下全体を表現する．空間を定義するために，以下のパラメータを与える．

- 空間を表す ID
- 空間の位置座標 $P_{space} = (x_s, y_s)$
- 空間を表す四角形のサイズ（幅，奥行き）
- 空間内のゲートの座標 $P_{gate} = (x_g, y_g)$
- ゲートのサイズ（長さ）

ここでゲートとは，Fig. 3 に示すような隣接する空間と接続するための領域であり，教室におけるドアや通路における接続部分を指す．空間やゲートの位置座標は最も左上の座標を選択し，それぞれのサイズはその位置座標からの幅や奥行きあるいは長さで表現する．

3.1.2 学生のパラメータ

学生のパラメータは，シミュレーション内でのエージェントを構成する役割を持つ．本研究では，エージェントをポテンシャルモデルによって移動則を決める．また，エージェントの進行方向や速度が限定されないように，開始地点から目的地点までの移動を微小時間ごとの速度ベクトルで決定する．学生を表すエージェントを定義するために，以下のパラメータを与える．

- 学生を識別する ID

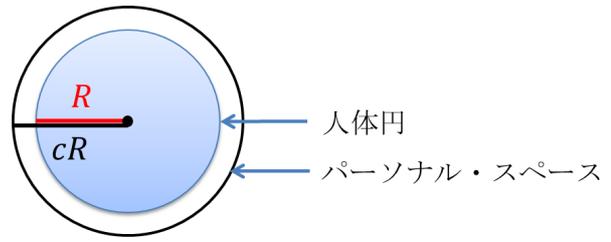


Fig. 4: Circle of agent and circle of personal space.

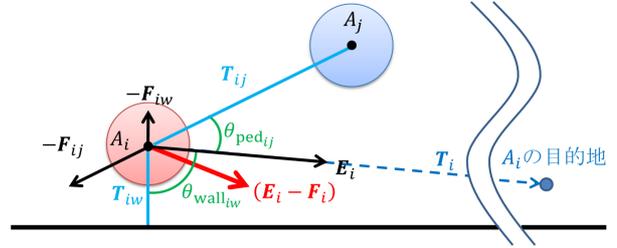


Fig. 5: How to determine gravitation and repulsion.

- 位置座標 $P = (x, y)$
- 人体幅を表す人体円の半径 R
- パーソナル・スペース比 c
- 速度ベクトル $V = (u, v)$
- 歩行速度 V_s
- 目的地点の ID（教室番号あるいは講義棟出入口の番号）

本モデルでは，シミュレーション時の計算量低減のため，人の体を円形で表わした人体円を利用する（Fig. 4）．しかし，人体円のみでの定義では，エージェント同士が完全に接触することを許すことになる．これは，非常に混雑している場所では問題ないが，周囲に人がほとんどいない場所で発生するのは現実的ではない．そこで，他者が侵入してくることに不快感を感じる領域であるパーソナル・スペースを人体円に導入する．パーソナル・スペースの形状は人体円と同じ円形とし，人体円の半径との比率 c （ただし， $c > 1.0$ ）で定義する．

3.2 移動モデルによる位置座標の決定

エージェントの移動モデルにはポテンシャルモデルを採用し，単位時間ごとの位置座標を速度ベクトルによって導出する．ポテンシャルモデルでは，クーロンの法則によって各エージェントへの引力や反発力を求め，速度ベクトルを導出する．本節では，エージェントと目的地にかかる引力，障害物（他のエージェントや壁）にかかる反発力の求め方を説明する．

3.2.1 引力および反発力の計算

Fig. 5 に，エージェント i における引力および反発力の例を示す．エージェントが目的地に向かって移動することを表現するため，エージェントには正の電荷が目的地には負の電荷が与えられており，互いの電荷間には引力が働く．エージェント i と目的地の電荷間にかかる引力を E_i とする．エージェント i の現在位置から目的地の位置までの距離が最短となるベクトルを T_i ，目的地に与える電荷を Q_e とすると，引力 E_i は，

$$E_i = Q_e \cdot \frac{T_i}{\|T_i\|} \quad (1)$$

である。このとき、エージェントの位置にかかわらず目的地に引き寄せられる引力の大きさは一定とする。また、本モデルでは2方向の歩行流（対向流）について考慮して、先頭エージェントへの追従の概念を取り入れ、追従ベクトルとして引力に加える。

現実世界におけるエージェント（学生）は、他のエージェントや壁などの障害物と接触しないように、障害物の位置や移動による影響を予測して行動する。この行動を表現するため、エージェントや壁には正の電荷が与えられており、互いの電荷間には反発力が働く。エージェント i と他のエージェント j との位置ベクトルを \mathbf{T}_{ij} 、エージェント i と壁 w との位置ベクトルを \mathbf{T}_{iw} とすると、他のエージェントからの反発力 $\mathbf{F}_{ped_{ij}}$ および壁からの反発力 $\mathbf{F}_{wall_{iw}}$ は、

$$\mathbf{F}_{ped_{ij}} = \frac{Q_{ped}}{\|\mathbf{T}_{ij}\|^2} \cdot \frac{\mathbf{T}_{ij}}{\|\mathbf{T}_{ij}\|} \cdot \cos\theta_{ped_{ij}} \quad (2)$$

$$\mathbf{F}_{wall_{iw}} = \frac{Q_{wall}}{\|\mathbf{T}_{iw}\|^2} \cdot \frac{\mathbf{T}_{iw}}{\|\mathbf{T}_{iw}\|} \cdot \cos\theta_{wall_{iw}} \quad (3)$$

である。 Q_{ped} 、 Q_{wall} は他のエージェントや壁の電荷の大きさ、 $\theta_{ped_{ij}}$ 、 $\theta_{wall_{iw}}$ はエージェント i にかかる引力 \mathbf{E}_i と各位置ベクトルのなす角である。式 (2)、式 (3) に余弦を乗ずることで、障害物に向かって進む場合に反発力が大きく、平行移動あるいは遠ざかる場合に反発力が小さくなるようにする。式 (2)、式 (3) から、対象となる他のエージェント数 n_{ped} と壁の数 n_{wall} を用いて、エージェント i にかかる全体の反発力 \mathbf{F}_i は、

$$\mathbf{F}_i = \sum_{j=1}^{n_{ped}} \mathbf{F}_{ped_{ij}} + \sum_{w=1}^{n_{wall}} \mathbf{F}_{wall_{iw}} \quad (4)$$

から求められる。ただし、すべての障害物に対して反発力を考慮すると計算量が多くなるので、各エージェントの視野に入る障害物のみを対象となるようにする。

3.2.2 速度ベクトルの導出

単位時間 Δt 後の速度ベクトル $\mathbf{V}_{i_{new}}$ は、現在の速度ベクトル \mathbf{V}_{old} および歩行速度 V_s 、式 (1)、式 (4) から、

$$\mathbf{V}'_i = \mathbf{V}_{old} + (\mathbf{E}_i - \mathbf{F}_i) \cdot \Delta t \quad (5)$$

$$\mathbf{V}_{i_{new}} = V_s \cdot \frac{\mathbf{V}'_i}{\|\mathbf{V}'_i\|} \quad (6)$$

で求められる。 \mathbf{V}_{i_0} は、障害物による反発力が存在しない場合での $\|\mathbf{V}'_i\|$ の最大値であり、これによって $\mathbf{V}_{i_{new}}$ は 0 から V_s までの大きさを持つ速度ベクトルとなる。

式 (6) から、単位時間 Δt 後の位置座標 $\mathbf{P}_{i_{new}}$ は、現在の位置座標 $\mathbf{V}_{i_{old}}$ を用いて、

$$\mathbf{P}_{i_{new}} = \mathbf{P}_{i_{old}} + \mathbf{V}_{i_{new}} \cdot \Delta t \quad (7)$$

となり、式 (7) から各エージェントの移動を決定する。

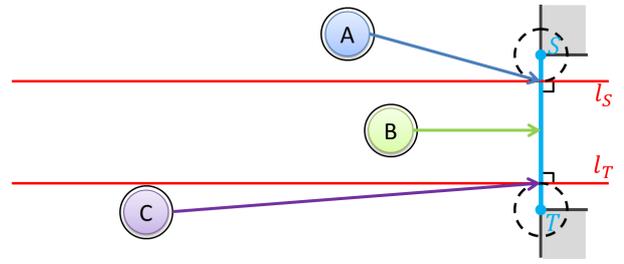


Fig. 6: Setting of destination.

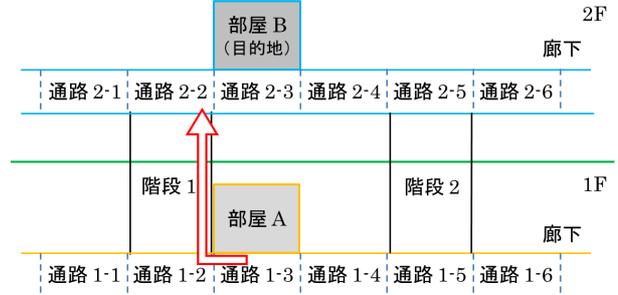


Fig. 7: Routing between classrooms.

3.3 経路選択モデル

3.3.1 目的地の設定

各エージェントは、ある地点を目指し移動しているが、エージェントはある1点ではなく、ある空間領域に向かっていていると考えられる。本研究では、目的地を点ではなく線分で表し、その線分上を通過したときに目的地に到達したものとす。また、ネットワークを利用して移動経路を決定するため、廊下の通路や階段などの通過地点を中間目的地として設定し、これらの目的地も線分で表す。

エージェントの移動経路は、障害物の影響がなければ通常目的地までの最短経路を通る。本モデルでは、各エージェントは単位時間ごとにおいて、目的地の線分上で最も近い点に向かうようにする。目的地に向かうエージェントの例を Fig. 6 に示しており、線分 ST を目的地として、エージェント A、B、C はそれぞれの矢印方向に進む。このとき、目的地の線分の端点 S および T からパーソナル・スペースも考慮した人体円半径の cR 分は障害物の接触のために目的地にすることができない。そのため、エージェント A、C は線分 l_s 、 l_T と線分 ST の交点に向かってそれぞれ進む。

3.3.2 移動経路の設定

本研究では、教室間移動の移動経路の選択にのみネットワークを利用する。各エージェントの移動経路は教室移動開始時に決定し、移動開始後に変更しないものとする。教室ごとに、目的地の各教室までの移動経路を事前に作成しておく。

教室間の移動において、開始地点と目的地が同じ階にある場合は、階段を利用する移動経路を利用するとは考えにくいので、移動経路は一意に決まる。開始地点と目的地が異なる階にある場合は、複数の移動経路が考えられるが、一般的には最短距離になる経路を選択する。Fig. 7 の例では、部屋 A から部屋 B までの移動に階段 1 を経由している。このとき、部屋 A を出て、通路 1-3、通路 1-2、階段 1、通路 2-2、通路 2-3 を経由して部屋 B にたどり着いており、それぞれの通路や階

段は中間目的地であると言える。つまり、エージェントがFig. 7のように移動するためには、目的地の部屋Bだけでなく中間目的地にも目的地の線分を示す必要がある。そこで、本シミュレーションでは3.1.1項において講義棟内の各空間に定義したゲートを目的地あるいは中間目的地として利用する。移動開始時には必ず教室か講義棟出入口に存在するため、次の中間目的地は通路であり、現在のエージェントの位置座標から最も近い中間目的地（通路）のゲートの線分を選択し移動する。中間目的地についての場合、次の中間目的地の通路や階段あるいは目的地の教室のゲートの線分を次の目的地として選択し移動する。これを繰り返し、目的地の教室に到着すれば、そのエージェントの移動は停止する。この行動をすべてのエージェントが完了することで、ある休憩時間における講義棟内での教室間移動は完了する。

4 シミュレーション実験

本シミュレーションモデルの妥当性を検証するため、実際に関西大学工学系講義棟の1つをモデル化し、実際の教室間移動のシミュレーションを行う。本シミュレーションでは、各教室、講義棟出入口から学生が一斉に移動を開始し、全員が目的地の教室に移動完了するまでの時間を測定する。シミュレーション実験で利用したシミュレーションモデルの各種パラメータおよびその結果については、当日の発表で議論する。

5 おわりに

本研究では大学講義棟内を対象に、休憩時間中の教室間移動において、群集事故が発生しやすい場所を特定するためのシミュレータについて述べた。学生の移動モデルには、エージェントを粒子に見立て、与えられた電荷をクーロンの法則に基づき速度ベクトルを導出するポテンシャルモデルを利用する。実際の講義棟の実測値および学生の履修データをもとに、実際の移動速度や障害物との接触を回避するような移動を設定することで、教室移動の様子を再現する。今後は、シミュレーションの結果をもとに具体的な混雑回避策について検討する。

参考文献

- 1) 鈴木達也, 村上達也, 藤本まなと, 和田友孝, 榎原博之, 岡田博美: 非常時緊急救命避難支援システム (ERESS) におけるリアルタイム協調型避難路探索方式の検討, 電子情報通信学会技術研究報告. IN, 情報ネットワーク, 112(209), pp.105-110 (2012).
- 2) 堀宗朗, 犬飼洋平, 小国健二, 市村強: 地震時の緊急避難行動を予測するシミュレーション手法の開発に関する基礎的研究, 社会技術研究論文集, Vol. 3, pp.138-145 (2005).
- 3) 北原武嗣, 岸祐介, 久保幸奨: 高低差を考慮した津波災害時の群衆避難における経路選択に関する一検討, 土木学会論文集 A1 (構造・地震工学), Vol.69, No.4, pp.1067-1075 (2013).
- 4) Purba Daru Kusuma and Azhari SN: Multi Agents based Traditional Market Customers' Behavior Design, 2013 International Conference on Computer, Control, Informatics and Its Applications, pp.331-336 (2013).
- 5) 上水流友望, 富井規雄: マルチエージェントモデルによる繁忙期における新幹線駅ホーム上の旅客流動シミュレーション, 電気学会論文誌 D (産業応用部門誌), Vol.134, No.8, pp. 750-759 (2014).

- 6) 前川廣太郎, 澤勢一史, 延原肇: 教室移動時間最適化のための群集団移動と多重解像ダイクストラ法を取り入れたマルチエージェントシステムと遺伝的アルゴリズムの開発, ファジィシステムシンポジウム講演論文集, Vol.28, pp.275-279 (2012).
- 7) 北島徹也, 伊津野和行, 八木康夫, 大窪健之: 大学の講義棟における災害時避難に関する一考察, 日本地震工学会論文集, Vol.11, No.1, pp.72-80 (2011).
- 8) 山下倫央, 野田五十樹: マルチエージェントシミュレーション:6. 避難シミュレーションの実社会への応用, 一般社団法人情報処理学会 情報処理, Vol.55, No.6, pp.572-578 (2014).

ニューラルネットに基づく消費電力データの解析

○大桃和樹 金田祐也 小檜山裕太 趙強福 (会津大学) 多屋優人 (株式会社 KDDI 研究所)

Analysis of Power Consumption Data by using Neural Network

*K. Omomo, Y. Kaneda, Y. Kobiyama, Q. Zhao (University of Aizu)
and M. Taya (KDDI R&D Laboratories)

Abstract— We want to establish a technology to estimate household structure or at-home situation, by analysis of power consumption data, indoor sensor data, and at-home situation data. If it was established, We would provide information related to saving power to the residents, or provide services for elderly persons. The first step of this research, here we confirm whether household structure can be classified based on only power consumption data. If it is possible, device costs are reduced.

Key Words: Power consumption, Neural network(NN)

1 緒言

近年、日本においては高齢者が増加し、日本の人口に対する高齢者の割合が増加している。高齢者の中には、家族で住んでいる人がいれば、家に一人で暮らしている事もある。一人暮らしの老人の中で問題となっているのが、孤独死である。文献¹⁾によれば、平成20年度には単身居住者で死亡から相当期間経過後に発見された件数が154件だったのに対し、平成24年度には220件に増加している。団塊世代の高齢化により、この問題はこれからさらに大きな問題と発展する可能性が高い。

我々は、宅内センサー及び宅内状況情報の分析による、世帯の情報を推定する技術について研究している。我々が対象とする情報は、宅内センサー、消費電力情報、及び宅内状況情報である。本技術が確立すれば、孤独死の早期発見、または異常を検知して自動的に救急車を呼ぶ等の方法を通して、孤独死を回避する事が可能であると考えている。さらには、効率的な電力消費を促す事で、資源の節約に貢献する事が可能となる。

本稿では、消費電力データのみに着目して、世帯の情報を検知する事が可能か検証する。もし消費電力データのみで検知可能であれば、導入コストを格段に抑える事ができ、より多くの家庭で導入を行う事が出来るようになる。検知方法として学習モデルであるニューラルネットワーク(NN)を基にした手法を提案し、検知可能かを確認する。

本稿の構成は、次の通りである。2章では、以前我々が研究していた内容を紹介し、本稿に至った経緯を説明する。3章では、本稿で提案する手法の関連手法について簡潔に紹介する。4章では、今回提案するニューラルネットワークを基にした推定方法について説明する。5章では、提案した手法を検証するための実験について説明する。6章では、5章で述べた実験の結果と、それについての考察を行う。7章は本稿のまとめであり、今後の活動について8章で紹介する。

2 前研究内容

我々は前研究の論文を、第24回インテリジェント・システム・シンポジウム(FAN2014)²⁾に投稿した。目的は本稿と同じで、消費電力データのみから、世帯情報を推定することである。前研究では、2つの手法を

提案し、実験を行った。

1つ目は、移動平均法を基にした手法である。消費電力データから住民の在宅を推定する方法として、我々は移動平均法を基にした手法を提案した。計測された各日の消費電力積算値だけではノイズが大きく、行動推定が難しいと判断したからである。そこで、我々は5日移動平均(5-day MA)と25日移動平均(25-day MA)を計算し、消費電力積算値と合わせて行動推定する方法を考えた。計算値を確認する為に、推定期間におけるグラフを準備し、行動推定の実験を行った。Fig. 1はこの実験でグラフ化したものの1つである。このグラフから、季節の変化に対応して消費電力が変化していることや、帰省の時期に大きな消費電力の変化があったことや、猛暑日に消費電力が増加していることなどが読み取れた。これらの結果から、居住者の冷暖房器具の使用や、居住者の不在などを推定できる。この実験から、居住者の行動が、消費電力データに表れていることがわかった。

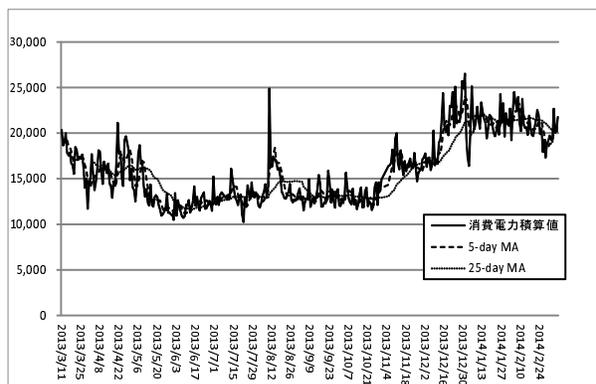


Fig. 1: Power consumption.

2つ目は、クラスタリング手法であるk平均法(k-means)を基にした手法である。消費電力データから家庭の世帯構成を推定する為に、我々はk-meansを基にした方法を考えた。k-meansを用いて、世帯の集合に対し、消費電力データを基にしたクラスタリングと、世帯情報を基にしたクラスタリングの2つを行い、それらの結果の類似度を測ることで、消費電力データからどの程度世帯情報を読み取ることができるかを確認した。2つのクラスタリング結果の類似度の計算には、調整ラ

ンド指数 (ARI) を用いた。また、各クラスタリング結果の特徴を確認する為に、クラスタリング結果について自己組織化写像 (SOFM) を用いて 2次元空間へ写像を行い、写像化された結果を基に推定を行った。まず、ARI の結果の考察を行った。ARI によって求めた類似度は 50%~60% という結果であった。これは、ランダムに判定した場合でも同等の数値が出る可能性があるため、精度がまだ足りないと考えられる。次に、SOFM の結果を考察した。Fig. 2 は、消費電力データを基にしたクラスタリング結果を SOFM で可視化したものである。消費電力データを基にしたクラスタリング結果からは、1つのクラスタを中心として放射状にマッピングされているという特徴が読み取れた。一方、世帯情報を基にしたクラスタリング結果からは、特定の順番で、クラスタが縞模様を描いてマッピングされているという特徴が読み取れた。このように、各クラスタリング結果はそれぞれ特徴を持っていたが、異なるものであった。消費電力データを基にしたクラスタリング結果の、放射状のマッピングの中心となっているクラスタに着目してみた。このクラスタに属する世帯の消費電力データについて調べてみたところ、共通する特徴を見つけた。それは、8時から16時までの消費電力は低く、それ以外の時間帯は高い、という特徴である。このことから、このクラスタに属する世帯は、8時から16時までは居住者は外出して、それ以外の時間帯は在宅してかつ電力を消費していると推測できる。このケースでは、居住者の行動のみでクラスタリングされた可能性があり、消費電力データと世帯情報の間には、相関性がない場合があると考えられる。

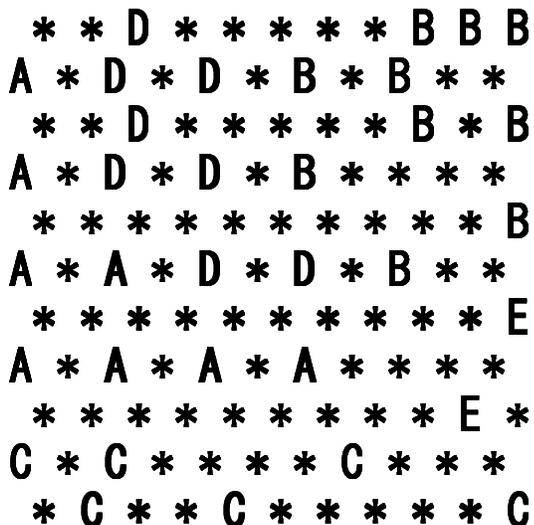


Fig. 2: SOFM

この研究から、消費電力データは世帯構成や在宅状況にある程度関与しているということがわかった。1つ目の手法では、帰省や冷暖房器具の使用などの、居住者の行動を推定することができた。2つ目の手法では、消費電力データは、居住者の外出のタイミングに大きく依存していることがわかった。そこから、推定の際にはこのことに気をつける必要があることがわかった。

以上が前研究の内容である。これは、人の目で行う推定手法であったり、教師なし学習を行う手法で

あった。しかし、この手法では将来的に自動化することが難しく、また、教師がないので推定の正しさが保証できないという欠点があることがわかった。その結果を参考にし、本稿では前研究とは逆に、ニューラルネットワークを基にした教師あり学習を行う手法を提案する。ニューラルネットワークを用いることにより、将来的に自動化が可能で、かつ実データを用いた教師あり学習であるので、推定の正しさが保証できると我々は考える。

3 基礎知識

3.1 ニューラルネットワーク

ニューラルネットワーク (NN) は、脳の神経回路の仕組みを計算機上で模倣したモデルである。NN は複数のニューロン (ノード) を持ち、それらをシナプス (重み付きエッジ) で結合し、ネットワークを形成する。学習によってシナプスの結合強度 (結合荷重) を変化させ、問題解決能力を持たせる。

3.2 多層パーセプトロン

多層パーセプトロン (MLP) は、複数の層にニューロンを配置させたフィードフォワード型ニューラルネットワークである。MLP は入力層、中間層 (隠れ層)、出力層の3つの層から構成される (Fig.3 参照)。入力として、特徴ベクトルと呼ばれる複数の特徴を表す数値からなるベクトルを与える。入力層、中間層、出力層の順にニューロンの出力を伝播していき、出力層の出力が最終的な MLP の出力となる。MLP は特徴ベクトルと教師信号を用いる教師あり学習が可能である。学習の際には誤差逆伝播法 (バックプロパゲーション) を用いる。これは、MLP の出力が教師信号と一致するように結合荷重を調整する方法である。

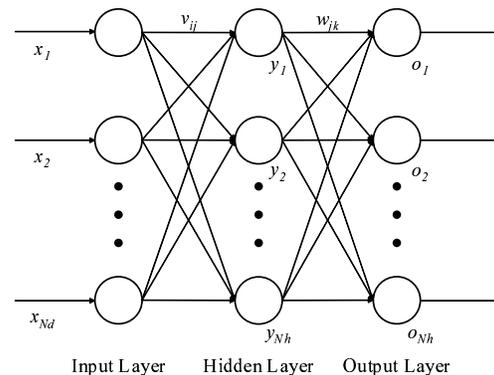


Fig. 3: Structure of MLP.

4 方法

消費電力データから世帯の情報を推定する為に、我々はニューラルネットワークを用いた方法を考えた。ニューラルネットワークである MLP を用いて、教師あり学習を行う。各世帯に対し、消費電力データを基に特徴ベクトルを作成する。そして、世帯情報を教師信号として MLP に学習させることで、世帯情報の推定能力を持たせる。特徴ベクトルの詳しい作成方法については 4.1 章で述べる。

4.1 1世帯の特徴ベクトルの作成法

以下の手順で1世帯を表す特徴ベクトルを作成する。

- Step 1: 一定期間内の消費電力データを用意する.
- Step 2: 各時間ごとに、一定期間内の消費電力の積算値を計算する.
- Step 3: 得られた 24 次元のベクトルを、その世帯の特徴ベクトルとする.

5 実験

5.1 実験設定

今回用いた消費電力データは、会津若松スマートシティ推進協議会から提供されたデータを用いた。データの収集には、Fig. 4にあるスマートメーターを用いた。消費電力データは 48 世帯分があり、15 分ごとの消費電力の積算値が記録されている。また、計測の対象となった世帯に関する情報も提供されている。提供された世帯の情報の内容は、Table 1の通りである。この情報の各項目を教師信号として用いた。これらを基に MLP を学習させ、推定能力の検証を行った。



Fig. 4: Smart meter

Table 1: Information of family

項目	区分	単位
居住者数	大人, 子供	人
世帯情報	単世帯, 複数世帯	世帯
延べ床面積		m^2
暖房機器	電気, ガス, 灯油	
住宅形態	戸建, マンション, アパート	
生活形態	定住, 一時滞在	

提案手法を検証するため、我々は 5 分割交差検定を用いた。居住者数、延べ床面積については、正答の値との誤差の平均値を算出した。居住者数については、大人の居住者数、子供の居住者数、大人と子供の居住者数の合計、の 3 つに分けた。それ以外の項目については、正答率を算出した。暖房機器については、電気とそれ以外の場合で消費電力データに大きな差が生じると考えたため、電気、ガス、灯油の 3 区分の場合以外にも、電気、その他の 2 区分の場合も検証した。全世

帯の特徴ベクトルの作成において、対象とする期間は、Table 2に示す 2 通りを検証した。この設定は、住人の行動には 1ヶ月、1週間ごとの周期性があると考えたためである。

Table 2: Terms of experiment.

期間	期間
期間 1(1ヶ月間)	2014年2月7日0時0分 ~ 2014年3月6日23時59分
期間 2(1週間)	2014年2月7日0時0分 ~ 2014年2月13日23時59分

5.1.1 MLP の設定

入力層、中間層、出力層の 3 層からなる MLP を用いた。入力層のニューロン数は特徴ベクトルの次元数と同じ 24 個に設定した。中間層のニューロン数は 5 個に設定した。出力層及び MLP の出力の設定は、教師信号の種類によって場合分けを行った。教師信号が居住者数などの数値の場合、出力層のニューロン数は 1 個とし、そのニューロンの出力を MLP の出力とした。教師信号が暖房機器などの数値以外の場合、出力層には各クラスラベルに対応するニューロンを設定し、出力層の中で出力値が最も大きいニューロンに対応するクラスラベルを MLP の出力とした。この時学習に用いる教師信号として、クラスラベルに対応する要素が 1 で、それ以外が 0 であるベクトルを用いる。学習率は 0.5、学習回数は 1000 回に設定し、実験を行った。

6 実験結果と考察

Table 3, 4 はそれぞれ 5 分割交差検定で計測した結果を表している。Table 3 は、居住者数、延べ床面積を教師信号とした時の、正答との誤差を表している。Table 4 は、それ以外の項目を教師信号とした時の、正答率を表している。

6.1 Table 3 に関する考察

大人の居住者数について、1~12 人の正答の幅に対し、約 1 人の誤差であった。このことから、大人の居住者数についてはある程度の検知ができているのではないかと我々は考える。期間の違いによる差については、あまり感じられなかったが、期間 1の方が少し誤差が小さかった。

子供の居住者数について、1 人未満の誤差であったが、正答の幅が 0~3 人なので、良い結果とは限らないと我々は考える。この項目についても期間 1の方が誤差が少し小さかった。

居住者数合計について、大人の居住者数の正答の幅と同じであったが、大人の居住者数における誤差より少し大きい結果となった。この項目についても期間 1の方が誤差が少し小さかった。

居住者数の項目についての結果全体を見ると、大人の居住者数が最も正確に検知できている結果であった。これは、消費電力データが、大人の居住者に大きく依存していて、逆に子供の居住者にあまり依存していないのではないかと我々は考える。

延べ床面積について、30~300 m^2 の正答の幅に対し、70 m^2 前後の誤差であった。これはかなり大きい誤差で

Table 3: Results of error.

	最小値	最大値	期間 1(1ヶ月)	期間 2(1週間)
大人の居住者数 (人)	1	12	1.24	1.36
子供の居住者数 (人)	0	3	0.69	0.87
居住者数合計 (人)	1	12	1.52	1.67
延べ床面積 (m^2)	30	300	67.82	75.02

Table 4: Results of accuracy.

	クラス数	期間 1(1ヶ月)	期間 2(1週間)
世帯情報 (%)	2	55.56	57.78
暖房機器 (電気, ガス, 灯油)(%)	3	51.11	57.78
暖房機器 (電気, その他)(%)	2	62.22	64.44
住宅形態 (%)	3	77.78	82.22
生活形態 (%)	2	91.11	88.89

はないかと我々は考える．この項目についても期間 1の方が誤差が少し小さかった．

6.2 Table 4 に関する考察

世帯情報について、クラス数 2 に対して 56%前後という結果であった．この結果は、あまり検知ができていないと我々は考える．この項目については期間 2の方が正答率が少し高かった．

暖房機器について、3 区分 (電気, ガス, 灯油) については 58%未満という結果であったが、2 区分 (電気, その他) については 63%前後という結果であった．しかし、これについてもクラス数 2 に対しての結果であるので、あまり検知ができていないと我々は考える．この項目についても、3 区分、2 区分どちらの場合も期間 2の方が正答率が少し高かった．

住宅形態については、クラス数 3 に対し、80%前後と高い正答率であった．戸建、マンション、アパートそれぞれを表す消費電力のパターンがあるのではないかと我々は考える．この項目についても期間 2の方が正答率が少し高かった．

生活形態については、クラス数 2 に対し、90%前後とこちらも高い正答率であった．こちらについても、定住、一時滞在それぞれを表す消費電力のパターンがあるのではないかと我々は考える．この項目については、期間 1の方が正答率が少し高かった．

6.3 期間に関する考察

Table 3 においては、期間 1の方が良い結果となった．逆に、Table 4 においては、生活形態以外の項目で、期間 2の方が良い結果となった．期間が長ければ長いほど、または短ければ短いほど良いという結果ではなかった．このことから、項目ごとに検知しやすい期間があるのではないかと我々は考える．例えば居住者数に関しては、データが 1 週間だけの場合、土日の 2 日間は休日であり、居住者が家にいる割合は高いが、それ以外の日の平日の 5 日間では、居住者が家にはいない割合が高いため、居住者数に関する情報が消費電力データに表れにくいのではないかと考える．

7 結論

本稿では、消費電力、宅内センサー情報、及び宅内状況情報データ分析による、世帯情報の推定する技術の確率を目指す最初の段階として、消費電力データのみに着目して検証した．MLP の教師あり学習を用いた

推定を、様々な項目について行ったが、精度は項目により様々であった．このことから、今回の手法には、扱う項目によって向き不向きがあることがわかった．大人の居住者数、生活形態(定住、一時滞在)の 2 つの項目については、良い結果であったので、これらの項目の検知については向いている手法なのではないかと考える．言い換えると、大人の居住者数や生活形態には、消費電力データのパターンと関連性があるのではないかと考えられる．それ以外の項目については、あまり良い結果は得られなかった．これらの項目には、消費電力データのパターンと関連性がないのではないかと考えられる．

8 今後の活動

本稿では、消費電力データのみを検証を行った．その結果を生かし、より詳細な推定を目指す研究をしたいと考えている．そのため、次に挙げる点について、これから検証していく．

1 つ目として、消費電力データから特徴ベクトルを作成する方法について、本稿で扱った方法以外にも検証することである．本稿では各時間ごとの消費電力積算値を計算し、24 次元のベクトルを作成したが、これ以外の方法も検証する必要があると考える．例えば、対象とする時間帯を、家族が全員家の中で活動している時間帯に絞ることで、消費電力データがより住人の行動に依存したものとなり、より良い精度が期待できるのではないかと考えている．

2 つ目として、消費電力データ以外に宅内センサー情報、及び宅内状況情報データも組み合わせた場合の推定の検証である．本稿では消費電力データのみを用いた手法であったので、さらなる精度向上が期待できる．しかし、扱うデータの種類が多く、複雑であるので、本稿で行った手法を参考に、なるべくシンプルな手法を考えていきたい．

参考文献

- 1) 内閣府, “平成 26 年版高齢社会白書 (全体版)”, http://www8.cao.go.jp/kourei/whitepaper/w-2014/zenbun/26pdf_index.html
- 2) FAN2014 実行委員会, “第 24 回インテリジェント・システム・シンポジウム (FAN2014)”, <http://www.cs.kitami-it.ac.jp/FAN2014/>

ファジィ遺伝的機械学習における 複数パターンを用いた新規ルール生成

○渡邊一弘 能島裕介 石渕久生 (大阪府立大学)

New Rule Generation from Multiple Patterns in Fuzzy Genetics-based Machine Learning

* K. Watanabe, Y. Nojima and H. Ishibuchi (Osaka Prefecture University)

Abstract— In genetics-based machine learning for classifier design, an if-then rule is often generated from the randomly-selected pattern in the training data. Since the generated rule is usually specified for the selected pattern, some attribute conditions are randomly replaced with “don't care” conditions in order to generalize the rule. To generate more appropriate rules that can cover several patterns, we propose a heuristic rule generation method from multiple patterns. We apply it to Michigan-style fuzzy genetics-based machine learning and examine the effect of the number of used patterns in the heuristic rule generation on the search performance.

Key Words: Fuzzy genetics-based machine learning, Michigan approach, Heuristic rule generation.

1 はじめに

ファジィ識別器の設計を行う手法の1つであるファジィ遺伝的機械学習(Genetics-based machine learning: GBML^{1), 2)}は、機械学習に進化型最適化手法を適用した学習手法であり、新しい知識の獲得や、データマイニングを行う方法として広く研究されている^{3), 4)}。本研究ではGBMLのアルゴリズムの1つであるMichigan型GBMLを用いる。

Michigan型ファジィGBMLにおける新規ルールの生成法の1つとして用いられているヒューリスティック生成法において、先行研究では1つのパターンからファジィルールを生成している。その後ルールの汎化性能を上げるため、あらかじめ設定した確率に基づいて条件部のファジィ集合を“don't care”に変更していた。しかし、この“don't care”適用確率は経験的に決定されているため、生成されるルールの条件部ファジィ集合の数が不適切である場合が考えられる。そのため本研究では、ルールの生成に複数のパターンを用い、それらのパターンが共通して覆われる属性のみにファジィ集合を割り当てることで、“don't care”適用確率を設定することなく、よりデータの特徴に合った条件部を持つルールの生成を行うことができるか調査する。

2 Michigan型ファジィGBML

2.1. ファジィ識別器

n 次元 M クラスのパターンを持つ m 個の学習用パターン $\mathbf{x}_p = (x_{p1}, \dots, x_{pn})$ が与えられているパターン識別問題を考える。 x_{pi} は p 番目の学習用パターンにおける i 番目の属性値 ($i = 1, 2, \dots, n$)を表す。この問題に対して以下に示されるIf-then形式のファジィルールを用いてファジィ識別器を設計する⁵⁾。

Rule R_q : If x_{p1} is A_{q1} and ... and x_{pn} is A_{qn}
then Class C_q with CF_q (1)

ここで R_q は q 番目のファジィルールを表すラベルであり、 x_{pi} は p 番目の学習用パターンにおける i 番目の属性値 ($i = 1, 2, \dots, n$)、 $A_q = (A_{q1}, \dots, A_{qn})$ は条件部ファジィ集合を表す。また、 C_q は結論部クラス、 CF_q はルール重みを表す。本研究では、三角型のメンバシップ関数

を持つファジィ集合を用いる。Fig. 1に示す2から5分割までの14種類のファジィ集合と、メンバシップ値を常に1とする“don't care”を合わせた合計15種類のファジィ集合を同時に用いる。

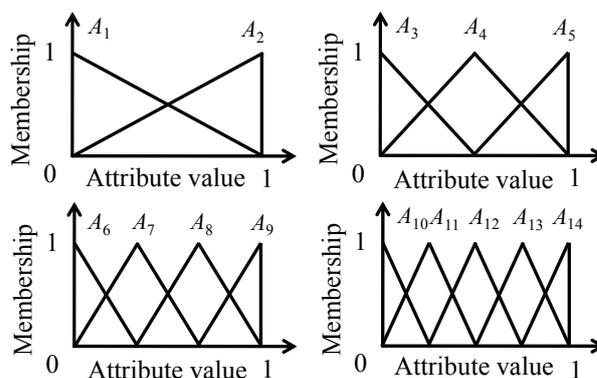


Fig. 1: Fuzzy membership functions used in this paper.

ファジィ識別器を生成する際には、各属性に対して条件部としてファジィ集合を決定した後、ルールの結論部クラス C_q を導出する⁶⁾。本研究では、結論部クラスの妥当性を高めるために、ルール R_q の信頼度から結論部クラスを導出する。ファジィ識別器は(1)式の形式で表されるファジィルールの集合である。入力 \mathbf{x}_p の識別には単一勝利ルールに基づく推論法を用いる。

2.2. 個体表現

Michigan型のファジィGBMLでは、1つのルールを1つの個体で表現する。次元数が n であるパターン識別問題を考える場合においては、1つのルールは n 個の条件部集合によって表されるため、1つの個体は長さ n の整数列として扱うことができる。

2.3. 初期個体群の生成

初期個体群として、個体群サイズ N_{pop} のファジィルールを生成する。1つのファジィルールの生成には同時適用パターン数 N_{combi} のパターンを用いる。

まず、学習用パターンからベースパターンを1つランダムに選択する。次に、サポートパターンとして、ベースパターンと同クラスのパターンを学習用パター

ンから $N_{\text{combi}} - 1$ 個選択する。選択した N_{combi} 個のパターンを用いて、属性ごとに各メンバシップ関数に対するメンバシップ値を算出する。 N_{combi} 個のパターンそれぞれのメンバシップ値の内、最小値をそのメンバシップ関数に対するメンバシップ値とする。このとき、メンバシップ値が 0.5 未満であれば 0 とする。全てのメンバシップ関数に対してメンバシップ値を割り当てた後、メンバシップ値が 0 以外のメンバシップ関数の中からルーレット選択を行い、属性のファジィ集合を決定する。全てのメンバシップ値が 0 の場合、その属性は “don't care” とする。この操作を全ての属性について行い、条件部ファジィ集合を決定する。

2.4. 個体の評価

個体の適応度には、各個体が正しく識別したパターン数を使用する。個体群であるルール集合を用いて学習用パターンを識別する際、単一勝利ルールによる推論法を用いるため、学習用パターン 1 つにつき 1 つの勝利ルールが決定される。この勝利ルールが学習用パターンを正しく識別できた回数を適応度とする。この時、個体群の中で全く同じファジィルールが存在する場合があるが、適応度を与えるのはその内 1 つのファジィルールのみである。

2.5. 新規ルールの生成方法

新規ルールの生成には、遺伝的生成法と複数パターンを用いたヒューリスティック生成法を用いる。

遺伝的生成法では、まず復元抽出によるバイナリトーナメント選択を用いて親個体を 2 つ選択する。そして、選択された親個体に一様交叉を適用し、2 つの子個体を生成する。2 つの個体から一方をランダムに選択し、突然変異操作を適用する。突然変異操作では、突然変異確率に従い条件部ファジィ集合をランダムに別のファジィ集合 (“don't care” を含む) に変更する。ファジィルールの結論部とルール重みはルールの条件部によって自動的に決定されるため、突然変異によりファジィ集合が変更された場合、学習用データを用いて再計算を行う。

複数パターンを用いたヒューリスティック生成法は、基本的には初期個体群の生成と同様であるが、ベースパターン 1 つは個体評価時に誤識別した学習用パターンの中からランダムに選択し、残り $N_{\text{combi}} - 1$ 個のサポートパターンの選択を以下の 3 種類の方法で行う。

- 誤識別したパターン集合からランダム
- 学習用パターン集合からランダム
- 学習用パターン集合の内、ベースパターンにユークリッド距離が近いものから順番

2.6. 新規ルールの生成と個体群の更新

生成される子個体の数 N_{rep} は、個体数によって決定される。本研究では、個体数の 20% (小数点以下切り上げ) とする。また、このとき生成法の組合せは以下の 2 種類のいずれかを用いる。

- 遺伝的生成法とヒューリスティック生成法
- ヒューリスティック生成法のみ

A) の場合、遺伝的生成法とヒューリスティック生成による新規ルールの生成は、生成数が同数になるように

行われる。ここで、子個体の生成数が奇数になる場合は、どちらの生成法で多く個体を作るかはランダムに決定する。

また、複数パターンを用いたヒューリスティック生成法では、誤識別パターンがルール生成数に満たない、または存在しない場合、ベースパターンを選択することはできない。この場合、ベースパターンも初期生成の場合と同様に学習用パターン全体から選択する。

新規ルールの生成後、個体群の更新を行う。このとき、個体群 N_{pop} 個のうち、適応度の低い順に N_{rep} 個を新しく作られた子個体と置き換える。

2.7. 識別器の獲得

本研究ではアルゴリズムの終了条件として世代数を用いる。また、最終的に獲得する識別器は全世代を通して学習用データに対する識別率が最も高かった個体群を選択する。つまり、最終世代の個体群ではなく最も学習用データに対する識別性能の高い個体群となる。これは、本研究で用いている Michigan 型のアプローチにおいて、進化がルール集合である識別器には関係なく個々のルールに基づいて行われるため、最終結果が必ずしも最良の識別器になるとは限らないからである。

3 数値実験

3.1. 数値実験設定

本研究の数値実験には、KEEL-data set repository⁷⁾で提供されているデータを用いた。本研究では、全ての属性値を [0, 1] で正規化して用いた。

Table 1: Four data sets used in this paper.

Data sets	#Patterns	#Attributes	#Classes
Glass	214	9	7
Heart	270	13	2
Pima	768	8	2
Vehicle	846	18	4

また、数値実験設定は次の通りである。

個体群サイズ (N_{pop}): 30
 ルール交換数 (N_{rep}): 6
 同時適用パターン数 (N_{combi}): 2, 3, 4, 5, 10
 交叉確率: 0.9
 突然変異確率: 1/n
 終了条件: 5000 世代経過
 試行回数: 300 回 (10CV × 30 回)

3.2. 数値実験結果

それぞれのデータに対する数値実験結果を Tables 2-9 に示す。サポートパターンの選択方法が 3 通り、新規個体の生成方法が 2 通り存在するため、計 6 通りの結果を示している。表内の Selection a-c はヒューリスティック生成法におけるサポートパターンの選択方法を表し、Type A, B は新規ルール生成時の生成法の組合せを表す。また、太字は従来の Michigan 型 GBML よりも優れていることを示す。Glass データ (Tables 2-3) では、学習用データに対する識別率が向上した設定と評価用データに対する識別率が向上した設定がほぼ一致していた。Heart データ (Tables 4-5) では学習用データに対する識別率が向上した設定については、評価用データに対して識別率が悪化していた。一方で、 N_{combi}

Table 2: The error rates on the training data of Glass data set.

N_{combi}	Selection a		Selection b		Selection c	
	Type A	Type B	Type A	Type B	Type A	Type B
Standard	25.39	25.24	25.39	25.24	25.39	25.24
2	24.95	22.35	24.57	21.84	24.25	21.71
3	26.00	24.60	25.12	23.20	25.30	24.33
4	26.55	27.45	25.87	26.08	25.70	26.28
5	26.87	31.11	26.51	30.99	26.07	28.52
10	27.15	40.07	27.17	42.30	27.13	39.74

Table 3: The error rates on the test data of Glass data set.

N_{combi}	Selection a		Selection b		Selection c	
	Type A	Type B	Type A	Type B	Type A	Type B
Standard	36.80	37.21	36.80	37.21	36.80	37.21
2	36.78	36.31	36.75	35.59	37.08	36.44
3	36.97	36.77	36.64	35.86	37.70	38.99
4	36.97	37.03	37.95	37.20	37.66	39.75
5	37.39	39.66	37.35	40.79	37.52	39.88
10	37.45	43.43	37.94	46.39	37.84	45.26

Table 4: The error rates on the training data of Heart data set.

N_{combi}	Selection a		Selection b		Selection c	
	Type A	Type B	Type A	Type B	Type A	Type B
Standard	9.44	8.91	9.44	8.91	9.44	8.91
2	10.20	9.38	10.30	9.69	10.85	29.70
3	9.80	9.86	10.05	11.70	10.78	19.12
4	9.76	10.57	10.08	12.91	10.50	12.79
5	9.71	11.01	10.12	13.99	10.29	11.74
10	9.76	12.86	10.39	17.42	10.16	14.09

Table 5: The error rates on the test data of Heart data set.

N_{combi}	Selection a		Selection b		Selection c	
	Type A	Type B	Type A	Type B	Type A	Type B
Standard	17.68	18.10	17.68	18.10	17.68	18.10
2	18.79	21.22	18.84	22.64	18.68	48.00
3	18.12	18.61	18.94	21.96	18.86	38.00
4	17.72	18.14	18.62	20.74	18.73	30.81
5	18.00	17.46	18.26	20.52	18.98	25.38
10	16.64	17.05	17.52	22.05	18.27	19.73

が 10 で遺伝的生成法とヒューリスティック生成法の両方を行う場合 (Type A) を見ると、それほど学習用データに対する性能は悪化していないにも関わらず、識別率用データに対する識別率が大きく向上していることがわかる。また、多くのデータにおいてベースパターンに近いものから順番に選択する設定はそれほど改善が見られなかったが、Pima データ (Tables 6-7) では、むしろ近いものから順番に選択する設定において圧倒的に高い性能を持つ識別器を獲得できた。しかし、Vehicle データ (Tables 8-9) のように学習用データに対して一部性能が向上している設定があるものの、評価用データに対してはむしろ悪化しているデータもあった。

以上より、正しく設定を行えば、多くのデータにおいて “don't care” 適用確率を用いなくても、従来の Michigan 型 GBML と同等か、それ以上の性能を持つ識別器を獲得することができると言える。

4 まとめ

本研究では、Michigan 型ファジィ GBML におけるヒューリスティック生成法において、複数のパターンを用いることでどのような有効性があるかを調査した。

数値実験結果より、複数のパターンが共通して覆われる属性値にのみファジィ集合を割り当てるのが有効であることが分かったが、一方で、データごとに有効な設定が異なることが分かった。また、多くのデータにおいて、“don't care” 適用確率を使用しなくても、正しく設定を選択すれば同程度かそれ以上の性能の識別器を得られることも分かった。

以上より、今後の課題としては、データごとに有効な設定を見つけることが挙げられる。また、本手法をハイブリッド型 GBML⁸⁾ に適用することで高い識別性能を持ち、複雑性の少ない識別器の獲得が期待できる。

Table 6: The error rates on the training data of Pima data set.

N_{combi}	Selection a		Selection b		Selection c	
	Type A	Type B	Type A	Type B	Type A	Type B
Standard	29.19	29.45	29.19	29.45	29.19	29.45
2	29.07	30.33	29.17	30.13	27.13	24.61
3	29.64	32.65	29.60	32.63	27.21	26.45
4	29.70	33.93	29.69	33.89	27.69	26.68
5	29.73	34.46	29.73	34.41	28.37	28.55
10	29.77	34.89	29.77	34.90	29.61	33.66

Table 7: The error rates on the test data of Pima data set.

N_{combi}	Selection a		Selection b		Selection c	
	Type A	Type B	Type A	Type B	Type A	Type B
Standard	30.08	30.46	30.08	30.46	30.08	30.46
2	30.03	32.18	30.02	31.93	29.30	28.17
3	30.39	33.63	30.47	33.49	29.47	29.82
4	30.42	34.30	30.44	34.30	29.79	30.00
5	30.53	34.64	30.49	34.59	29.98	31.54
10	30.51	34.89	30.56	34.89	30.48	34.53

Table 8: The error rates on the training data of Vehicle data set.

N_{combi}	Selection a		Selection b		Selection c	
	Type A	Type B	Type A	Type B	Type A	Type B
Standard	32.55	33.43	32.55	33.43	32.55	33.43
2	33.12	33.50	33.16	33.44	33.28	31.83
3	33.17	34.51	33.18	33.58	33.22	31.72
4	33.09	35.41	33.24	34.37	33.15	32.08
5	33.09	36.06	33.12	34.89	33.11	32.02
10	33.16	41.24	33.15	38.53	33.36	32.55

Table 9: The error rates on the test data of Vehicle data set.

N_{combi}	Selection a		Selection b		Selection c	
	Type A	Type B	Type A	Type B	Type A	Type B
Standard	35.06	36.17	35.06	36.17	35.06	36.17
2	35.74	38.12	35.80	37.99	37.57	38.56
3	35.70	37.96	35.57	37.59	37.46	38.24
4	35.63	38.41	35.92	37.49	36.91	38.21
5	35.42	38.70	35.59	38.08	36.70	37.80
0	35.35	43.16	36.00	40.33	35.95	36.60

参考文献

- 1) Cordón, O.: A historical review of evolutionary learning methods for Mamdani-type fuzzy rule-based systems: Designing interpretable genetic fuzzy systems. *International Journal of Approximate Reasoning* 52, 894/913 (2011)
- 2) Fernández, A., García, S., Luengo, J., Bernadó-Mansilla, E., Herrera, F.: Genetics-based machine learning for rule induction: State of the art, taxonomy, and comparative study. *IEEE Trans. on Evolutionary Computation* 14, 913/941, (2010)
- 3) Y. Jin, J. Branke : Evolutionary optimization in uncertain environments – A survey. *IEEE Trans. on Evolutionary Computation* 9, 3, 303/317 (2005)
- 4) L. B. Booker, D. E. Goldberg, J. H. Holland : Classifier system and genetic algorithms. *Artificial Intelligence* 40, 1-3, 235/282 (1989)
- 5) H. Ishibuchi, K. Nozaki, H. Tanaka : Distributed representation of fuzzy rules and its application to pattern classification. *Fuzzy Sets and Systems* 52, 1, 21/32 (1992)
- 6) H. Ishibuchi, T. Yamamoto : Rule weight specification in fuzzy rule-based classification systems. *IEEE Trans. on Fuzzy Systems* 13, 4, 428/435 (2005)
- 7) J. Alcalá-Fdez, A. Fernández, J. Luengo, J. Derrac, S. García, L. Sánchez, F. Herrera : KEEL data-mining software tool: data set repository. *Integration of Algorithms and Experimental Analysis Framework. Journal of Multiple-Valued Logic and Soft Computing* 17, 2-3, 255/287 (2011)
- 8) H. Ishibuchi, S. Mihara, Y. Nojima : Parallel distributed hybrid fuzzy GBML models with rule set migration and training data rotation. *IEEE Trans. on Fuzzy Systems* 21, 2, 355/368 (2013)

多目的ファジィ遺伝的機械学習の並列分散実装

○高橋佑治 能島裕介 石渕久生 (大阪府立大学)

Parallel Distributed Implementation of Multiobjective Fuzzy Genetics-based Machine Learning

* Y. Takahashi, Y. Nojima and H. Ishibuchi (Osaka Prefecture University)

Abstract— Fuzzy genetics-based machine learning is one of data mining techniques using evolutionary computation. By applying it to numerical datasets, we can obtain fuzzy rule-based classifiers, which are accurate and linguistically interpretable for human users. However, there are two major problems. One is that it is impossible to design the classifiers with the highest accuracy and interpretability due to their tradeoff relation. Evolutionary multiobjective optimization algorithms are often used to obtain a number of classifiers with the different tradeoff between accuracy and complexity. The other is the heavy computational load when this method is used for large datasets. In this paper, we apply parallel distributed implementation to the multiobjective fuzzy genetics-based machine learning to overcome the latter problem, and we examine the effects of our parallel distributed implementation on the search performance through computational experiments.

Key Words: Fuzzy genetics-based machine learning, Parallel distributed implementation, Multiobjective optimization

1 はじめに

近年、遺伝的ファジィシステム (Genetic Fuzzy Systems: GFS) に関する研究が盛んに行われている^{1),2)}. GFS によって設計されるファジィ識別器においては、複雑性が小さいこと、また、識別性能が高いことが重要である. しかし、これらの目的の間にはトレードオフの関係があり、識別性能が高く、かつ複雑性が小さい識別器を獲得することは困難である. そのため、複数の目的を同時に最適化し、そのトレードオフに沿った多数の非劣な識別器集合を獲得することのできる進化型多目的最適化 (Evolutionary Multiobjective Optimization: EMO) アルゴリズムを用いた多目的 GFS が提案されている^{3),4)}.

GFS は一般に高い探索性能を持つが、大規模データに適用した際の計算時間が問題点として挙げられている. この問題に対する解決策として、データの特徴選択や並列計算など様々な手法が提案されている^{5),6)}. 解決策の1つとして、先行研究において我々は単一目的の GFS に対する並列分散実装を提案した^{7),8)}. これは学習用データと個体群を部分学習用データと部分個体群へと分割し、複数の CPU コアを用い学習を行うものである. 部分個体群中の最良個体を別の部分個体群に移動させる移住操作や、部分データ集合の交換操作を一定間隔で行うことにより、単一目的の GFS において、学習性能に大きな悪影響を与えることなく計算時間を大幅に削減することができることを示した. 本研究では、並列分散実装を多目的ファジィ遺伝的機械学習 (Genetics-based Machine Learning: GBML⁴⁾) に対して適用し、個体の移住間隔や部分データの交換間隔の設定を変更することにより、得られる個体群や計算時間にどのような変化が見られるか調査する.

2 多目的ファジィ遺伝的機械学習

2.1 ファジィ識別器

n 次元 M クラスのパターン $\mathbf{x}_p = (x_{p1}, \dots, x_{pn})$ が m 個与えられているパターン識別問題について考える. x_{pi} は p 番目のパターンの第 i 次元 ($i = 1, 2, \dots, n$) における属性値を表す. この問題に対して、次に示す If-then ルールを用いてファジィ識別器を設計する⁹⁾.

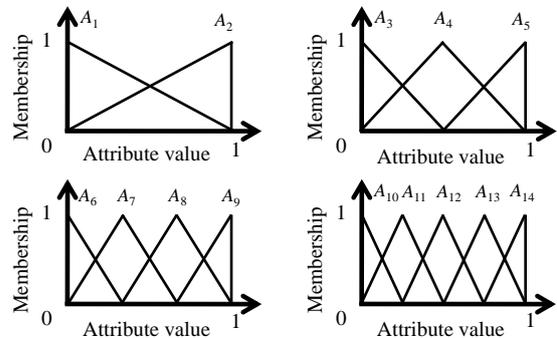


Fig. 1: Fourteen kinds of fuzzy membership functions.

Rule R_q : If x_{p1} is A_{q1} and ... and x_{pn} is A_{qn}
then Class C_q with CF_q (1)

ここで、 R_q は q 番目のファジィルールを表すラベルであり、 $\mathbf{A}_q = (A_{q1}, \dots, A_{qn})$ は条件部ファジィ集合を表す. 条件部ファジィ集合として、本研究では Fig. 1 に示す粒度の異なる 4 種類のファジィ集合を用いる. さらに、条件部に含まれる条件の数が少ないルールを生成するため、メンバシップ値が常に 1 となる“don't care”も条件部集合に含める. つまり、合計 15 種類のファジィ集合を同時に用いる. また、(1)式において、 C_q は結論部クラス、 CF_q はルールの重みを表す. ファジィ識別器は、(1)式で表されるファジィルールの集合により構成される. 未知パターンのクラスの決定には単一勝利ルールによる推論法を用いる¹⁰⁾.

2.2 多目的ファジィ遺伝的機械学習

本研究では、ファジィ識別器の設計のために多目的ファジィ GBML⁴⁾ を用いる. GBML には個体がルール集合を表現する Pittsburgh 型のアプローチと個体がルールを表現する Michigan 型のアプローチがある²⁾. 本研究で用いる多目的ファジィ GBML は、個体が識別器を表す Pittsburgh 型の GBML を基本として用い、そのルール集合の中のルールを個体として扱いルール単位の最適化を行う Michigan 型のアプローチもアルゴリズム内で行う. 多目的ファジィ GBML の手順を次に示す.

- Step 1: 学習用データ集合を用い、個体群の初期化、初期個体群の評価を行う。
- Step 2: Pittsburgh操作における遺伝的操作（選択、交叉、突然変異）を行い、子個体群を生成する。
- Step 3: 子個体に対して、Michigan操作適用確率に基づき、Michigan操作を行う。
- Step 4: 子個体の評価、世代更新を行う。
- Step 5: 終了条件を満たさない場合、Step 2へと戻る。
- Step 6: 個体群の中から非劣な個体を選択し、最終的な識別器集合とする。

本研究で用いる多目的ファジィGBMLでは、個体の評価として、識別器の誤識別率とルール数の2つの目的を用いる。

$$\text{Minimize } f_1(S) \text{ and minimize } f_2(S) \quad (2)$$

ここで、 $f_1(S)$ は個体 S の学習用データに対する誤識別率、 $f_2(S)$ はルール数を表す。

Pittsburgh操作における個体の選択操作や、個体群の更新は、代表的なEMOアルゴリズムであるNSGA-II¹¹⁾において用いられている非劣ランキングと混雑距離に基づいて行われる。

3 並列分散実装

ファジィ GBML を大規模なデータ集合へと適用した際、計算に膨大な時間がかかるという問題がある。そこで本研究では、多目的ファジィ GBML に対して並列分散実装^{7), 8)}を適用する。並列分散実装では、複数の CPU コアを用い、個体群とデータ集合を分割した部分集合により学習を行うことで計算時間の短縮を図る。GBML における並列分散実装のモデル図を Fig. 2 に示す。ここでは、個体群と学習用データ集合はそれぞれ7個の部分個体群、部分データ集合へと分割されている。部分個体群と部分データ集合のペアがそれぞれ1つの CPU コアへと割り当てられ、GBML によって部分個体群の最適化が行われる。個体群と学習用データがそれぞれ7個の部分集合へと分割される場合、計算時間は分割を行わない場合と比較して 1/49 となることが期待できる。

並列分散実装では、2つの操作を行う。1つは、部分データ集合の交換操作である。この操作では、部分個体群に対応する部分学習用データ集合を回転させるように交換する。もう1つは、個体の移住操作である。この操作は、部分個体群中で最も適応度の優れた個体を移住先の部分個体群へコピーし、移住先の部分個体群において最も適応度の劣った個体を除去する操作である。これら2つの操作により、学習用データに対する過学習を防ぎ、また、探索能力を向上させることができる。多目的 GBML において移住操作に関する個体の選択は (2)式における $f_1(S)$ を基準に行われる。つまり、各部分個体群で最も誤識別率の低い個体が移住先へコピーされ、移住先では最も誤識別率の高い個体の削除が行われる。

上記の並列計算は、2.2節の GBML の操作の流れのうち、Step. 2 から Step. 5 において行われる。各部分個体群において最終世代での学習が終了した後、全ての個体について、全ての学習用データを用いて(2)式の評価を行う。並列分散型多目的ファジィ GBML において得られる識別器の集合は、このときに非劣であった個体の集合となる。

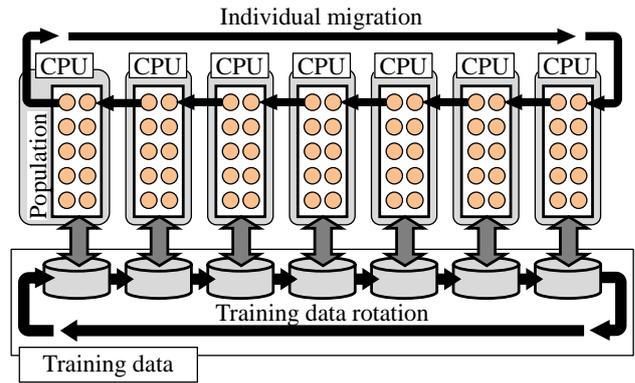


Fig. 2: Our parallel distributed implementation.

4 数値実験

数値実験において、多目的ファジィ GBML に対し並列分散実装を適用した際に得られる個体群の特徴や、その計算時間に与える影響を調査する。

4.1 数値実験設定

本研究の数値実験には、KEEL-data set repository¹²⁾により提供されている実世界データを用いた。使用データを Table 1 に示す。

Table 1: Three data sets used in this paper.

Data Sets	Patterns	Attributes	Classes
Phoneme	5404	5	2
Satimage	6435	36	7
Segment	2310	19	7

また、多目的ファジィ GBML において、本研究では以下の設定を用いた。

- 個体群サイズ: 210
- ルール数: 30 (初期), 1 (最小), 60 (最大)
- don't care 適用確率: $(n-5)/5$ (n は属性数)
- 終了条件: 50,000 世代
- 試行回数: 50 回 (10CV x 5 回)

Pittsburgh 操作:

- 交叉確率: 0.9
- 突然変異確率: $1 / (n \cdot |S|)$ ($|S|$ はルール数)

Michigan 操作:

- Michigan 操作適用確率: 0.5
- 交叉確率: 0.9
- 突然変異確率: $1 / n$

並列分散実装:

- 分割数: 7 分割
- 個体移住間隔: 1, 10, 50, 100, None
- 部分データ交換間隔: 1, 10, 50, 100, None

4.2 数値実験結果

それぞれのデータに対して多目的ファジィ GBML を適用したときに得られた識別器の集合を Figs. 3-5 に示す。ここで、各図において、得られた識別器の評価用データに対する誤識別率を識別器のルール数ごとに平均をとった値で示している。各図でのそれぞれのルール数における誤識別率は、結果の信頼性のため、50 回の試行のうち 25 回以上の試行において得られた結果のみを平均している。

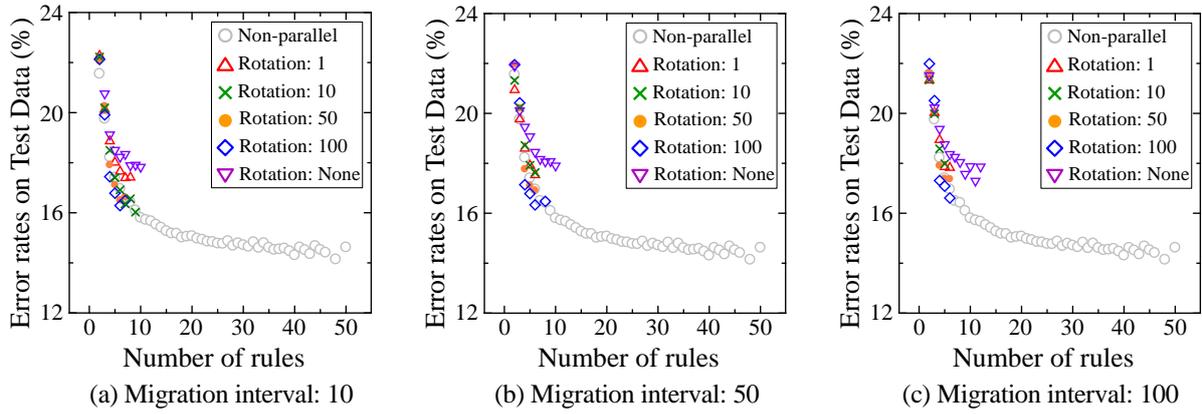


Fig. 3: The error rates on the test data of the Phoneme data set.

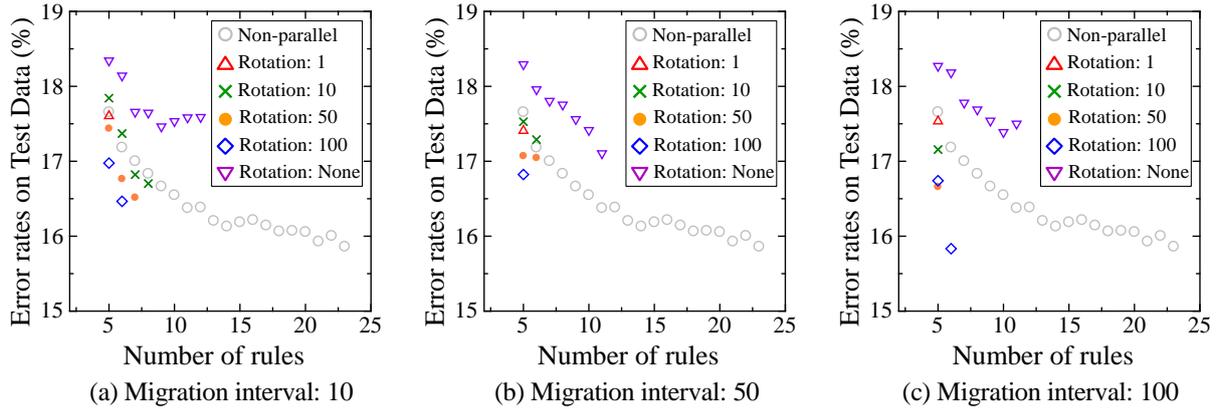


Fig. 4: The error rates on the test data of the Satimage data set.

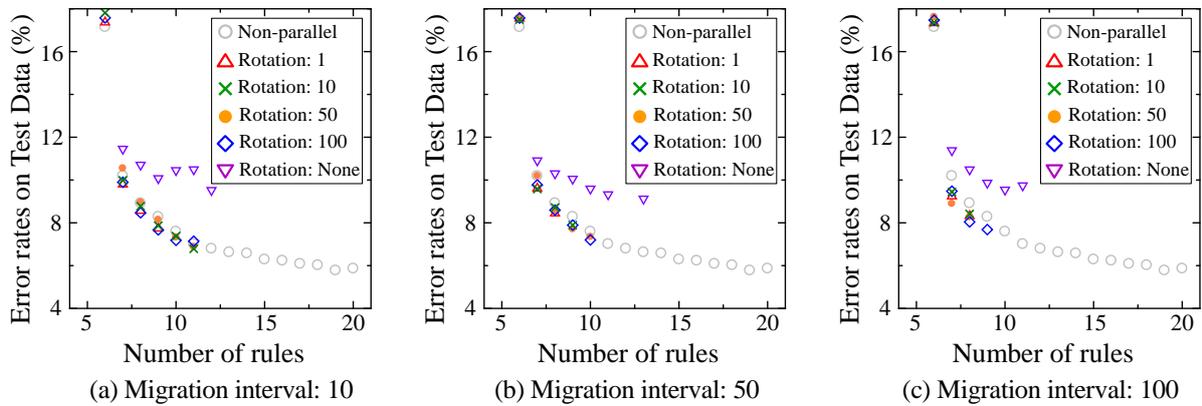


Fig. 5: The error rates on the test data of the Segment data set.

各図はそれぞれ、並列分散実装を行わない多目的ファジィ GBML の結果を灰色の丸印で表し、並列分散型多目的ファジィ GBML の結果は個体の移住間隔ごとに、データの交換間隔を変更した結果を1つの図にまとめている。図には、個体の移住間隔を10世代、50世代、100世代とした場合の結果を示している。

数値実験結果より、どのデータに対しても並列分散実装を適用することで得られる個体の数が少なくなり、ルール数の大きい個体が獲得できていないことが分かる。特に、個体の移住間隔を大きく設定した場合に、移住間隔が小さい場合よりも得られたルール数が少なくなる傾向が見られた。部分データの交換間隔に関しては、移住間隔ほど得られる個体数に影響はないことが分かる。しかし、データの交換間隔の設定によって、同じルール数の個体で比較すると、誤識別率に差が出

ていることが分かる。データ交換を行わなかった場合は、得られた個体の誤識別率は並列分散実装を行わない場合より高くなっているが、データ交換を行った場合、全体的に誤識別率は小さくなっていることが多い。特に Satimage データ (Fig. 4) では、移住間隔100世代、交換間隔100世代と設定した場合など、大きく誤識別率が低下している場合が見られた。これらの結果から、適度に移住操作や交換操作を行うことで、より識別性能の高い個体が獲得できると考えられる。

次に、計算時間の変化を Figs. 6-8 に示す。これらの図は、並列分散実装を行わない多目的ファジィ GBML の計算時間に対し、それぞれの個体移住間隔、データ交換間隔において何倍の高速化が得られたかを表している。また、各図において、7分割時の期待値である49倍以上の高速化が得られた設定を黒く記している。

Phoneme データでは, Fig. 6 より, ほとんどの設定において, 非常に大きく高速化ができていたことが分かる. Satimage, Segment データにおいても, Figs. 7, 8 より, 部分データの交換間隔を 10 世代, 50 世代, 100 世代とし, また個体移住間隔を大きく設定した場合に期待値以上の高速化ができていたことが分かる. Phoneme データでは, Fig. 3 より, 並列分散型多目的ファジィ GBML では, 非並列の場合と比べて得られた個体数が特に少なくなっていることが分かる. これが Phoneme データにおいて大きな高速化が得られた原因として考えられる. 他の 2 つのデータにおいても, 移住間隔を大きく設定し, 適度にデータの交換を行った場合に得られる個体数が少なくなり, 高速化率が高くなっていることが分かる.

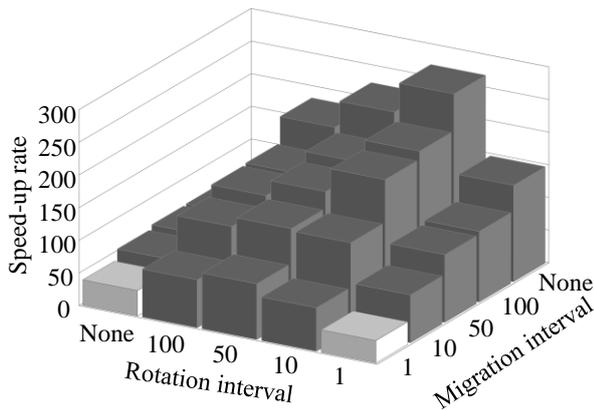


Fig. 6: Computation time of the Phoneme data set.

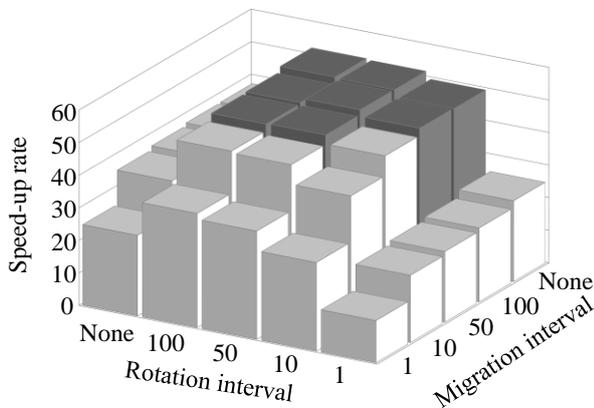


Fig. 7: Computation time of the Satimage data set.

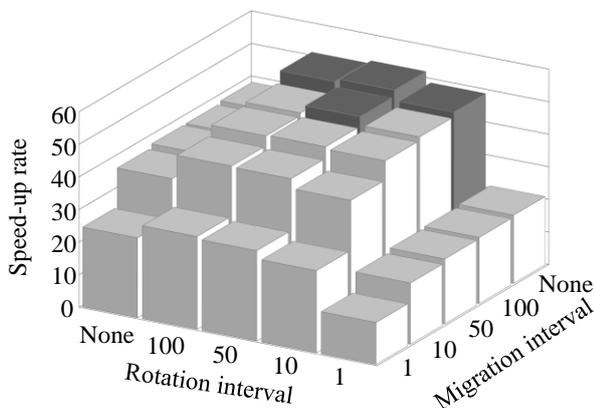


Fig. 8: Computation time of the Segment data set.

5 まとめ

本研究では, 並列分散実装を多目的ファジィ GBML に対して適用し, 個体の移住間隔や部分データの交換間隔の設定を変更することにより得られる個体群や計算時間にどのような変化が見られるかを調査した.

数値実験結果より, 個体群と学習用データを分割することで, 得られる個体の数が減少し, ルール数が大きい個体は獲得できなくなることが分かった. しかし, 個体の移住間隔や部分学習用データの交換間隔を適切に設定することにより, 非並列非分散の場合に得られたルール数と同じルール数の識別器と比較した際, より評価用データに対する誤識別率の低い個体を得ることができた. また, 並列分散実装により計算時間も大幅に削減できることも示した. 並列分散実装を行いつつ, より多くの個体を獲得することが今後の課題だといえる. そのため, より精度の高い識別器の獲得へとバイアスをかけた探索を行うことが考えられる.

参考文献

- 1) Cordon, O.: A historical review of evolutionary learning methods for Mamdani-type fuzzy rule-based systems: Designing interpretable genetic fuzzy systems. *International Journal of Approximate Reasoning* 52, 894/913 (2011)
- 2) Fernández, A., García, S., Luengo, J., Bernadó-Mansilla, E., Herrera, F.: Genetics-based machine learning for rule induction: State of the art, taxonomy, and comparative study. *IEEE Trans. on Evolutionary Computation* 14, 913/941 (2010)
- 3) Alcalá, R., Nojima, Y., Herrera, F., Ishibuchi, H.: Multiobjective genetic fuzzy rule selection of single granularity-based fuzzy classification rules and its interaction with the lateral tuning of membership functions. *Soft Computing* 15, 2303/2318 (2011)
- 4) Ishibuchi, H., Nojima, Y.: Analysis of interpretability-accuracy tradeoff of fuzzy systems by multiobjective fuzzy genetics-based machine learning. *International Journal of Approximate Reasoning* 44, 4/31 (2007)
- 5) Cano, J.R., Herrera, F., Lozano, M.: Stratification for scaling up evolutionary prototype selection. *Pattern Recognition Letters* 26, 953/963 (2005)
- 6) Fernández de Vega, F., Cantú-Paz, E. (eds): *Parallel and Distributed Computational Intelligence*, Springer (2010)
- 7) Nojima, Y., Ishibuchi, H., Kuwajima, I.: Parallel distributed genetic fuzzy rule selection. *Soft Computing* 13, 511/519 (2009)
- 8) Ishibuchi, H., Mihara, S., Nojima, Y.: Parallel distributed hybrid fuzzy GBML models with rule set migration and training data rotation. *IEEE Trans. on Fuzzy Systems* 21, 355/368 (2013)
- 9) Ishibuchi, H., Nakashima, T., Nii, M.: *Classification and Modeling with Linguistic Information Granules: Advanced Approaches to Linguistic Data Mining*. Springer, Berlin (2004)
- 10) Ishibuchi, H., Nakashima, T., Morisawa, T.: Voting in fuzzy rule-based systems for pattern classification problems. *Fuzzy Sets and Systems* 103, 223/238 (1999)
- 11) Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. on Evolutionary Computation* 6, 182/197 (2002)
- 12) <http://www.keel.es/>

段階的視覚を持つエージェントの 実数値環境における追跡問題について —視覚を持ち Hunter から逃げる Prey の捕獲—

○山口 昂彦 馬野 元秀 (大阪府立大学)

Pursuit Problem in Real Number Environment by Agents with a Staged View — Capturing the Escaping Prey with a Staged View

*Takahiko Yamaguchi Motohide Umano (Osaka Prefecture University)

Abstract A pursuit problem is a multi-agents' benchmark problem, where four hunters pursue and capture the prey in a cell environment. We have proposed a staged view that reflects a human's view where we can see an object easily in the neighborhood but more difficult in the longer distance and easily in the center direction but more difficult in the righter and lefter directions in both cell and real-number environments. We showed that hunters can capture the prey even with the staged view by fuzzy Q-learning. In the research, the prey selects actions randomly without any views. It is, however, natural that the prey with some view escapes from hunters. In this paper, we introduce the prey with a staged view selecting an escaping action by fuzzy Q-learning. We simulate this problem and compare the result with the previous one.

Key Words: Staged view, Pursuit problem, Fuzzy Q-learning

1. はじめに

マルチエージェントの標準問題に追跡問題がある [1]。これはマス目上で 4 つの Hunter が 1 つの Prey を追跡して捕獲する問題である。我々は、人間の視覚の「距離が遠くなると見えにくい」や「正面はよく見えるが左右方向は見えない」という特徴を表現した段階的視覚を提案し、マス目状の離散環境 [2][3] と、より現実に近い実数値環境 [4][5] において、段階的視覚を持つエージェントの学習について研究を行ってきた。その結果、両方の場合において段階的視覚でもファジィ Q-Learning により学習が可能であることが分かった [2][3][4][5]。

従来の研究では、Prey は視覚を持たず、Hunter との位置関係を考慮せずにランダムに移動していた。しかし、視覚を持ち視覚情報を基に Hunter から遠ざかるように移動をする方が自然である。そこで、本論文では Prey にも Hunter と同様の段階的視覚を持たせ、ファジィ Q-Learning [6] を用いて Hunter に捕まらないように行動させる。シミュレーションを行い、Prey が視覚を持たない場合の結果と比較する。

2. 実数値環境の追跡問題と ファジィ Q-learning

まず、実数値環境の追跡問題 [4][5] と Hunter の学習に用いたファジィ Q-learning [6] について説明する。

2.1 実数値環境の追跡問題

実数値環境の追跡問題では、一辺の長さが b の正方形でトーラス状の実数値空間上に、1 つの Prey と 4 つの Hunter がいる。Hunter の目標は Fig.1 のように Prey を囲むことである。どちらのエージェントも大きさが半径 r_1 の円形で、向きがあり、どちらも最大で前方に l_1 、後方に $l_1/2$ 、横に $2l_1/3$ の範囲内に移動できる。まず、移動する位置を Hunter が観測情報を用いて決定し、次に Prey がランダムに決定する。その後、各エージェントは決定した場所に小刻みに移動する (衝突すればそこで停止する)。これを 1 ターンと呼ぶ。

Hunter は半径 r の半円状の視覚範囲を持っている。視界内にエージェントがいれば、Fig.2 のようにその位置を相対距離 d と相対角度 θ で得ることができる。Hunter は Table.1 のような状態と行動の組合せからなる表をもとに行動の決定と学習を行う。

目標状態は、Fig.3 のように 2 ~ 4 つの Hunter の捕

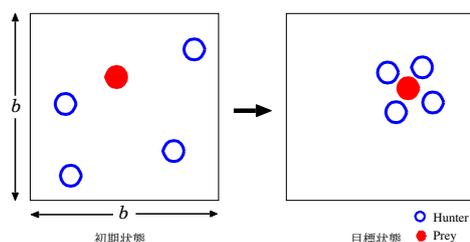


Fig. 1: 実数値環境の追跡問題

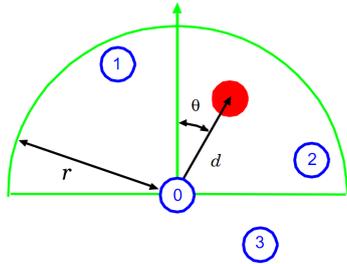


Fig. 2: Hunter の視覚範囲

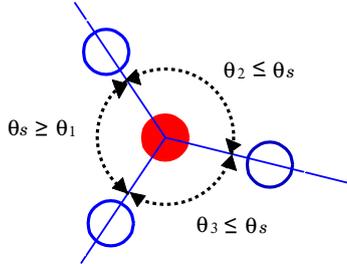


Fig. 3: 目標状態

Table. 1: 評価値表

状態		行動			
距離	角度	前	後	右	左
1 ~ 2	左	0.24	0.31	0.05	0.3
	前	0.01	0.45	1.12	0.35
	⋮	⋮	⋮	⋮	⋮
2 ~ 3	左	0.52	0.46	0.01	0.84
	前	0.74	0.31	0.23	0.7
	⋮	⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮	⋮	⋮

獲範囲 r_2 内に Prey がいる状態で、隣り合う Hunter との角度 (Prey の位置を角の頂点とする) が、全て θ_s 。以下のときであるとする。ターンの最後に盤面が目標状態であるかどうかを判定し、状態に応じて各 Hunter に報酬が与えられる。

我々はこの実数値環境の追跡問題にファジィ Q-learning を適用する研究を行ってきた [7][8]。

2.2 Q-learning とファジィ Q-learning

Q-learning [9] は強化学習の代表的な手法の一つであり、エージェントが行動を決定し行動した後に、前の状態に対する行動の Q 値 (評価値表の行動に付けられた値) を更新する。更新は、次の行動の中での最大の評価値を割引き、それに報酬を加えた値に近づけるという形で行う。ある状態で高い報酬を得た場合は、更新ごとにその状態に到達することが可能な状態にもその報酬が伝播することにより学習が行われる。

通常の Q-learning では、1 回の行動で 1 つの Q 値を更新するだけであるが、ファジィ Q-learning [6] では、状

態をファジィ集合にすることにより、1 回の行動で複数の Q 値を一致度に応じて更新する。ある時刻 t で一致度が μ である状態 s_t で行動 a_t を選び、状態が s_{t+1} に遷移したとする。このとき $Q(s_t, a_t)$ を次の式で更新する。

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \mu \alpha [r + \gamma \max_{a_{t+1} \in A} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)]$$

ここで、 α は学習率、 γ は割引率、 r は報酬を表す。 $\max_{a_{t+1} \in A} Q(s_{t+1}, a_{t+1})$ は各状態 s_{t+1} の最大の評価値に s_{t+1} との一致度を掛けた値の和で求める。なお、この更新式は文献 [6] のものとは少し異なっている。

評価値の初期値を 0 より大きい値にしておくと、正の報酬が得られるまで Q 値が下がるので、最大の評価値の行動を選択すれば、前に選んだ行動が選ばれなくなり、環境の探索が可能となる。

3. 段階的視覚

段階的視覚は 2003 年に山村ら [2][3] によって提案された追跡問題における粗視化の手法の一つで、正確な観測情報が得られる領域から大雑把な観測情報しか得られない領域へと段階的に変化する視覚である。

(1) 段階的視覚の構成

山村らによって提案された段階的視覚は、人間の視覚の「距離が遠くなると見えにくくなる」や「正面はよく見えるが左右方向は見えにくい」という特徴を表現したものであり、距離に応じて「近傍」「近距離」「中距離」「遠距離」という段階で構成されている。各段階でエージェントが得ることができる情報は、距離が大きくなるにつれて精度が悪くなっていく。文献 [4] における実数値環境の段階的視覚では、離散環境とは異なり各視覚段階の境界が明確でなくなるので、段階をファジィ的に構成することで視覚が徐々に変化していくことを表現している。人間の視野は両目で固視できる範囲 (両眼注視野)、両目で見える範囲 (両眼視野)、片目でのみ見える範囲 (片眼視野) に角度によって分かれている [10][11]。両眼注視野と片眼視野の境界線の角度は Fig.4 よりそれぞれ前方 60° 、 200° であり、両眼視野の境界線の角度は Fig.5 より前方 120° である。各段階の形状が同じではその特徴を表現できないので、文献 [5] では各段階の境界線を媒介変数表示の式で

$$\begin{aligned} \text{近傍} & \quad x = 0.7 \sin 2\theta, \quad y = 2 \sin \theta \\ \text{近距離} & \quad x = 3 \sin 2\theta, \quad y = 4 \sin \theta \\ \text{中距離} & \quad x = 3(1 + \cos \theta) \sin \theta, \\ & \quad y = 3(1 + \cos \theta) \cos \theta \end{aligned}$$

と定義した。これを図示すると Fig.6 のようになる。

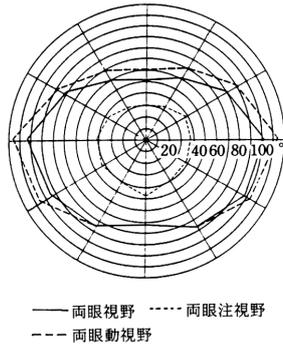


Fig. 4: 両眼注視野 [10]

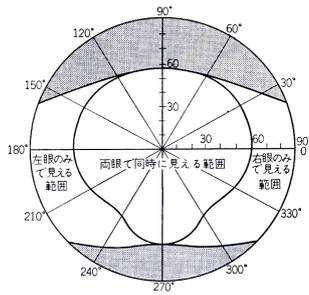


Fig. 5: 両眼視野 [11]

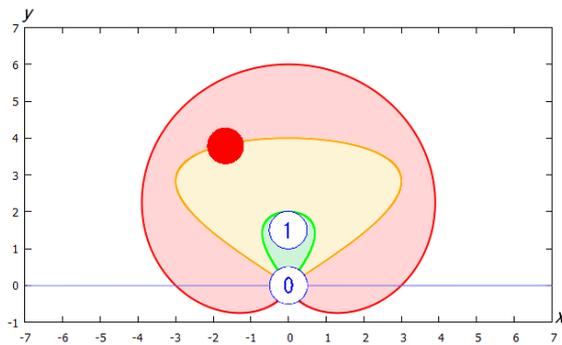


Fig. 6: 本研究の段階的視覚

(2) 段階的視覚で Hunter が得ることができる情報

各段階で Hunter が得ることができる情報は異なり、距離が大きくなるにつれて精度が悪くなっていく。各段階にいるエージェントについて次のような情報が得ることができる。

(a) 近傍：各エージェントの識別ができ (Hunter の場合は番号も分かる)、エージェントとの相対距離と相対角度が明確に分かる。

(b) 近距離：Hunter か Prey かの識別ができ (Hunter の番号は分からない)、エージェントが「近距離」にいるという情報と、相対方向が分かる。相対方向は Fig. 7 のファジィ集合のラベルとその一致度で与えられる。

(c) 中距離：エージェントの存在の情報を得ることができる (Prey か Hunter かも分からない)、エージェントが「中距離」にいるという情報と相対方向が分かる。相対方向は近距離と同様に Fig. 7 のファジィ集合のラベルと

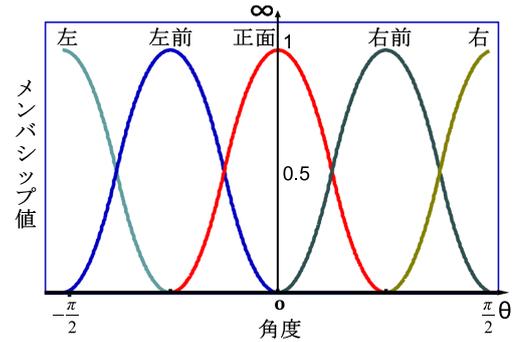


Fig. 7: 相対方向のファジィ集合

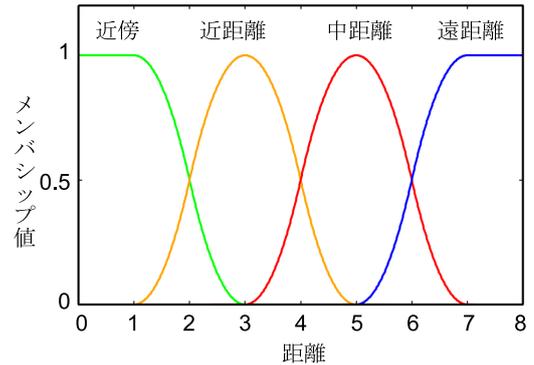


Fig. 8: 各段階のファジィ集合

Table. 2: 各段階で得られる情報

	観測対象の情報	位置情報	
近傍	個体の区別ができる	相対距離	相対角度
近距離	Hunter と Prey の区別	[近距離]	相対方向
中距離	エージェントの存在	[中距離]	相対方向
遠距離	なし	なし	

その一致度で与えられる。

(d) 遠距離：視覚外であるので情報は得られない。エージェントが後方にいる場合も含んでいる。

各段階で得られるこれらの情報をまとめたものが Table. 2 である。そして、これらの段階を Fig. 8 のようなファジィ集合で構成する。したがって、1つのエージェントに対して各段階の一致度とその段階で得られる情報の組が得られる。例えば、Fig. 6 のような状況の場合、Hunter0 に与えられる Prey と Hunter1 についての情報は

{0.6/〈 近距離, Prey, [近距離], {0.5/左前, 0.5/前} },
0.4/〈 中距離, Agent, [中距離], {0.5/左, 0.5/前} }
{0.7/〈 近傍, Hunter1, (相対距離 1.5), (相対角度 0) },
0.3/〈 近距離, Hunter, [近距離], (1.0/前) }

となる。これらが各段階ごとにまとめられて、

- 近傍 {0.7/⟨Hunter1, 相対距離, 1.5⟩,
0.7/⟨Hunter1, 相対角度, 0⟩}
- 近距離 {0.6 × 0.5/⟨Prey, 相対方向, 左前⟩,
0.6 × 0.5/⟨Prey, 相対方向, 前⟩,
0.3 × 1.0/⟨Hunter, 相対方向, 前⟩}
- 中距離 {0.4 × 0.5/⟨Agent, 相対方向, 左⟩,
0.4 × 0.5/⟨Agent, 相対方向, 前⟩}

となる。

(3) 段階的視覚における Hunter の Q 表の構成

一般の Q-learning では、状態に対応する情報が得られると考えるが、段階的視覚では、段階ごとに得られる情報が異なるので、1つの Q 表では表現できない。そこで、Hunter に「近傍」「近距離」「中距離」「遠距離」の4つの Q 表を持たせる。各段階での Q 表の状態は次のようになっている。

(a) 近傍

Prey に対しては相対距離についてのファジィ集合「極近い」と「近い」を Fig.9 のように定義し、相対角度は Fig.7 の左、左前、前、右前、右とする。したがって、Prey の状態数は $2 \times 5 = 10$ に不明を加えた 11 状態となる。Hunter に対しては相対距離の情報は用いないので、Hunter の状態数は相対方向に不明を加えた 6 状態となる。したがって、「近傍」の状態数は $11 \times 6^3 = 2376$ となる。Fig.9 のファジィ集合で 1 未満が定義されていないのは、エージェントの半径が 0.5 であり、エージェント間の距離は 1 以上であるからである。

(b) 近距離

「近距離」にいるという情報は「近距離」の Q 表を使用することで暗黙的に使用する。Prey と Hunter の区別が可能なので、Prey に対しては相対方向を状態としてそのまま使用する。Hunter に対しては相対方向を左と左前を左に、右と右前を右にまとめて、左、前、右で表現し、それぞれに Hunter が何体いるかで状態を表す。度合いが付いているのでその和を用い、0 体ぐらい、1 体ぐらい、2 体ぐらい、3 体ぐらいの 4 つの状態を表す。状態数は $6 \times 4^3 = 384$ である。しかし、左に Hunter が 3 体、前に Hunter が 3 体存在する等の状態は起こりえないので、実際に使用する状態数は 120 である。

(c) 中距離

「中距離」にいるという情報は「中距離」の Q 表を使用することで暗黙的に使用している。Prey と Hunter の区別がつかないので、左、前、右それぞれにエージェントが何体いるか (0 体ぐらい、1 体ぐらい、2 体ぐらい、3 体ぐらい) で状態を表す。状態数は $5^3 = 125$ であるが、「近距離」と同じように起こりえない状態が存

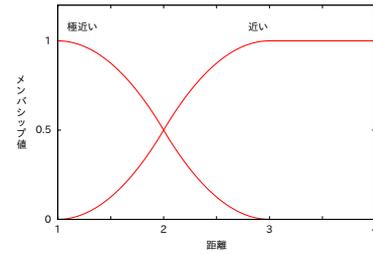


Fig. 9: 近傍での Prey に対する距離のファジィ集合

在するので、実際に使用する状態数は 35 である。

(d) 遠距離

遠距離は視覚外であり不明の 1 状態である。

(4) 段階的視覚における行動の選択と学習

ファジィ的な段階的視覚ではエージェントは隣り合う 2 つの段階に属することになるので、属する段階ごとに行動を決定し、それらを段階に属する一致度を考慮して合成したものを行動とする。

例として、Hunter0 から見たとき近傍において Prey の相対距離が「近い」で「右」にいて、他の全ての Hunter が「右」にいるという状態に対する一致度の計算方法を示す。Prey、Hunter1、Hunter2、Hunter3 の近傍に対する一致度をそれぞれ $\mu_p, \mu_1, \mu_2, \mu_3$ とし、相対距離をそれぞれ d_p, d_1, d_2, d_3 とし、相対角度をそれぞれ e_p, e_1, e_2, e_3 とする。近傍の Q 表の属性は Prey との相対距離と相対角度、Hunter1、Hunter2、Hunter3 の相対角度なので、このとき、 $G_{近}(d_i)$ を d_i の「近い」のファジィ集合に対する一致度、 $G_{右}(e_i)$ を e_i の「右」のファジィ集合に対する一致度として、状態の一致度 μ を次のように計算する。

$$\mu = (\mu_p \cdot G_{近}(d_p)) \cdot (\mu_p \cdot G_{右}(e_p)) \cdot (\mu_1 \cdot G_{右}(e_1)) \cdot (\mu_2 \cdot G_{右}(e_2)) \cdot (\mu_3 \cdot G_{右}(e_3))$$

この一致度が正の状態について、Q 値を基に行動を選択し、これを一致度により合成する。この段階ごとの行動の決定はファジィ Q-learning[7] と同じである。

属する全ての段階で行動の決定を行い、各段階から得られた行動を各段階の一致度の和で正規化して合成

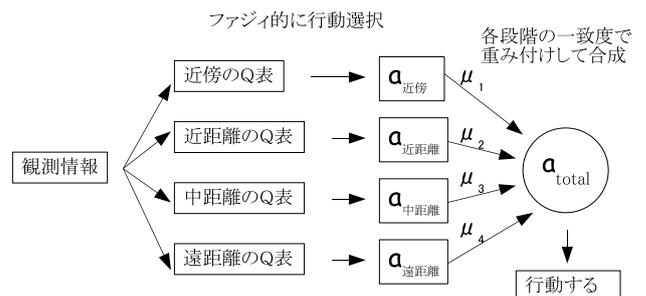


Fig. 10: 行動の選択と学習の流れ

し、全体の行動とする。Hunter の学習においては、行動の結果与えられた報酬は段階の一致度と状態の一致度の積の割合で分配される。この流れを図示したものが Fig.10 である。

(5) 視覚外に出た Prey の位置に対する推測

従来の研究では Hunter の視覚内にエージェントがない場合、直前ターンの状況に関わらず遠距離の Q 表を用いて行動選択していた。これでは直前ターンに Prey が近傍にいた場合と近距離にいた場合の区別をしていないが、人間であれば最後に Prey を観測したときの場所を基に現在の場所を推測して行動することができる。そこで、論文 [12] では推測時に異なる Q 表を用いることにより、Hunter の捕獲性能が向上するかを調べた。

(a) 推測を行う条件

推測を行う目的は Prey を見失ったときに、その状況に応じた行動をとることにより Prey を再発見することにある。そこで推測用の Q 表を用いて行動選択する条件を「1 ターン前に Prey が視覚内にいたが、現ターンでどのエージェントも視覚内にいない」状態とした。

観測したエージェントが Hunter であるか Prey であるかを区別できるのは、近傍と近距離のいずれかに属しているときなので、Prey が視覚内にいるという状態は、Prey の近傍・近距離のどちらかの一致度が 0 より大きいという状態である。

(b) 推測用の Q 表の構成

推測を行うターンでは全てのエージェントが視覚外に属しており、視覚情報がないので、1 ターン前の情報を基に行動を選択する。1 ターン前の Prey が近傍に属していたならば相対距離と相対角度がわかり、近距離に属していたならば相対方向が分かる。

しかし、推測は 1 ターン前の情報を基に行動選択するために、実際の視覚情報より信頼性が低くなる。そこで、近傍に属していた場合でも、相対距離を考慮せずに、左と前と右の 3 状態の相対方向のみを用いる。

Prey を見失った原因として、Prey が逃げただけでなく Hunter が Prey のいない方向に向いてしまった場合も考えられるので、1 ターン前に自分が動いた方向を前後左右の 4 状態からなる属性として用いて、Table.3 のような Q 表を定義する。状態数は $3 \times 4 = 12$ である。

(c) 推測の報酬

推測に基づく行動の結果 Prey を再発見したとき、すなわち Prey の近傍と近距離のどちらかの一致度が 0 より大きくなったとき、成功報酬 λ を与える。また、推測の目的は Prey を再発見することであり捕獲することではないので、捕獲に対する成功報酬と半成功報酬は

Table. 3: 推測用の Q 表

状態		行動			
直前の Prey の相対方向	直前の自分の行動	前	後	左	右
左	前	0.23	0.11	0.10	0.10
	後	0.24	0.11	1.11	0.11
	左	0.25	0.10	0.10	0.11
	右	0.1	0.1	0.1	0.1
前	前	0.24	0.31	0.05	0.3
	後	0.11	0.1	0.1	0.1
	左	0.10	0.1	0.1	0.1
	右	0.13	0.1	0.10	0.10
右	前	0.23	0.10	0.10	0.10
	後	0.24	0.11	0.11	0.11
	左	0.10	0.24	0.10	0.11
	右	0.1	0.1	0.1	0.1

与えない。

(d) 数値実験

従来 Hunter が行っていたファジィ Q-learning に推測用の Q 表を導入し実験を行った。その結果、従来の推測を行わない場合は捕獲成功率が 0.713、捕獲成功ターン数が 365.6 ターンであったのに対し、推測を行う場合は捕獲成功率が 0.894、捕獲成功ターン数が 227.6 ターンとなり、捕獲成功率と捕獲成功ターン数の両方で Hunter の性能の向上が見られた。それらが向上した理由としては、遠距離の Q 表は視覚情報が無いために状態が 1 つしかないが、視覚外の状態にも様々な場合がある。従来はそれら全てを 1 つの状態で表して、同じ行動をとっていたが、この実験では推測用の Q 表を用いることにより場合に応じた行動をとることができるようになったからである。

4. Prey の段階的視覚と Q 表

いままでの研究では、Prey は段階的視覚を持たず、状況に関わらず一定の範囲でランダムに行動を選択していた。しかし、Prey の行動としては視覚を持ち視覚情報を基に Hunter から遠ざかるような行動を選択する方が自然である。そこで、本論文では Prey にも Hunter と同じ段階的視覚を持たせ、Q 表を基に行動を選択させることにより、Hunter から逃げる行動を学習させる。

(1) 段階的視覚における Prey の Q 表の構成

段階的視覚で Prey が得ることができる情報は Hunter と同様であるが、本論文の追跡問題は Prey が 1 つなので、Prey の Q 表には Hunter に関する情報のみでよい。そこで、Prey の Q 表は Hunter の Q 表から Prey に関する状態を削除し、Hunter 1 つ分の状態を加えたものとなる。各段階での Q 表の状態は次のようになってい

る。

(a) 近傍

Hunter に対しては相対距離の情報は用いないので、Hunter の状態数は相対方向に不明を加えた 6 状態となる。したがって、「近傍」の状態数は $6^4 = 1296$ となる。

(b) 近距離

「近距離」にいるという情報は「近距離」の Q 表を使用することで暗黙的に使用する。Hunter に対しては相対方向を左と左前を左に、右と右前を右にまとめて、左、前、右で表現し、それぞれに Hunter が何体いるかで状態を表す。度合いが付いているのでその和を用い、0 体くらい、1 体くらい、2 体くらい、3 体くらい、4 体くらいの 5 つの状態を表す。状態数は $5^3 = 125$ である。しかし、左に Hunter が 4 体、前に Hunter が 4 体存在する等の状態は起こりえないので、実際に使用する状態数は 35 である。

(c) 中距離

Hunter の情報のみを使用するので、近距離とまったく同じ Q 表の構成になる。状態数は 35 である。

(d) 遠距離

遠距離は視覚外であり不明の 1 状態である。

(2) 段階的視覚における Prey の行動の選択と学習

Hunter には報酬として、目標状態になったときに成功報酬を、目標状態ではないが捕獲範囲に Prey がいるときに半成功報酬を与える。Prey にとっての目標状態はゲーム終了まで Hunter に捕獲されず逃げ切ったときの状態であるが、そのときの状態には Hunter から離れるように移動して逃げ切った場合や、Prey から Hunter の方向に移動しても捕獲されなかった場合などの様々な場合がある。必ずしも逃げるために適切な行動をとっていると限らないので、それら全ての状態に報酬を与えても Hunter から逃げる行動の学習が進むとは考えにくい。そこで、Prey が Hunter に捕獲されたときに負の報酬を与えることで、捕獲される行動を選択しないように学習させることにより、Hunter から逃げる行動を学習させる。Q 表の初期値を -0.1 とし、成功報酬を -1.0 、半成功報酬を -0.01 とする。

(3) 数値実験

実数値環境の段階的視覚を用いた追跡問題 [5] に、本論文で提案する段階的視覚を持ち Q 表を基に行動選択する Prey を用いて実験を行った。

[実験 1]

従来の実数値環境の追跡問題に、段階的視覚を持ち Q 表を基に行動選択する Prey を用いて実験を行った。この実験では Hunter は推測を行わない。1 ゲームを 1000

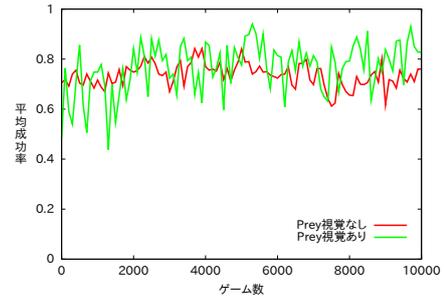


Fig. 11: 実験 1—平均捕獲成功率

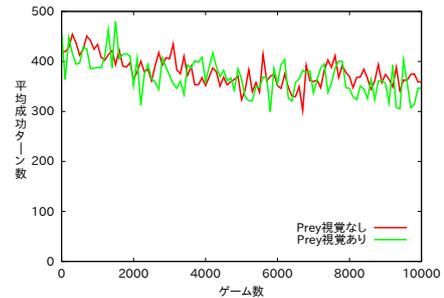


Fig. 12: 実験 1—平均捕獲成功ターン数

ターンとし、10000 ゲーム行ない、100 ゲーム毎に Q 値を更新せずに 100 ゲームの学習なしの試行を行った。これを 1 セットとして 10 セット行ない、その学習なしの試行 100 ゲームの平均の成功率と成功時ターン数の 10 セット平均を図示したのが Fig.11 と Fig.12 である。

このときの盤面の大きさは一辺 $b = 15$ で、エージェントのパラメータは、最大の移動距離 $l_1 = 1.0$ 、大きさの半径 $r_1 = 0.5$ 、Hunter の捕獲半径 $r_2 = 1.5$ とする。学習に使用するパラメータは、学習率 $\alpha = 0.0075$ 、割引率 $\gamma = 0.99$ 、Hunter の Q 値の初期値 0.1 、成功報酬を 1.0 、半成功報酬を 0.01 とし、Prey の Q 値の初期値 -0.1 、成功報酬を -1.0 、半成功報酬を -0.01 とし、実験を行った。学習時の行動の選択は ϵ -greedy 方策を用い、その探索率を $\epsilon = 0.01$ とし、学習なしの場合は Q 値が最大の行動を常にとるように $\epsilon = 0$ とした。8000 ~ 10000 ゲームの結果の平均は、従来の Prey が段階的視覚を持たずランダムに行動選択する場合の捕獲成功率が 0.713 、捕獲成功ターン数が 365.6 ターンであったのに対し、本論文の段階的視覚を持ち Q 表を基に行動選択する Prey の場合、捕獲成功率が 0.812 、捕獲成功ターン数が 341.9 ターンとなった。

実数値環境の追跡問題に、本論文で提案する段階的視覚を持ち Q 表を基に行動選択する Prey を用いると、Hunter にとっての捕獲成功率が上昇し、捕獲成功ターン数が小さくなるのが分かった。このような結果となった原因は、盤面の大きさが十分ではないため、Prey が逃げるスペースが空いていないからではないかと考えられる。

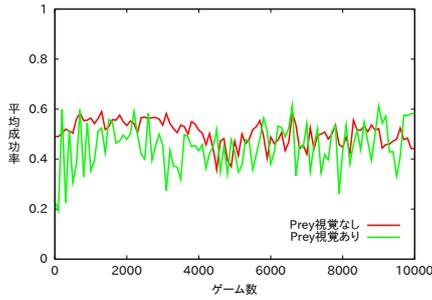


Fig. 13: 実験 2-平均捕獲成功率

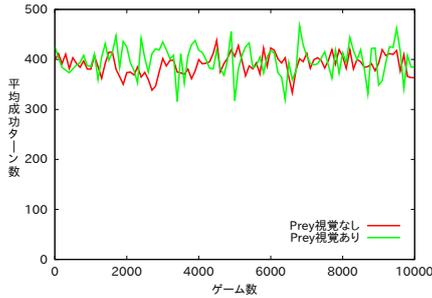


Fig. 14: 実験 2-平均捕獲成功ターン数

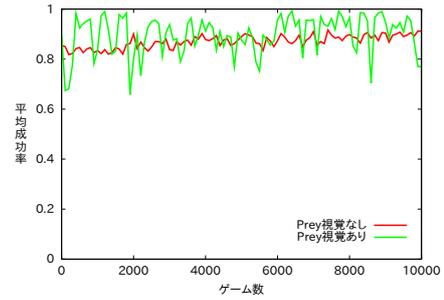


Fig. 15: 実験 3-平均捕獲成功率 (b=15)

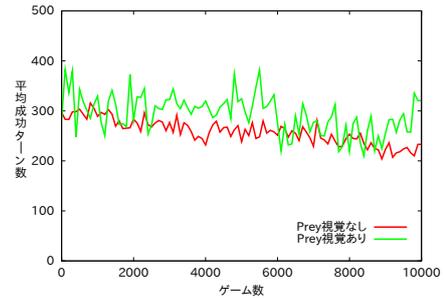


Fig. 16: 実験 3-平均捕獲成功ターン数 (b=15)

[実験 2]

実験 1 の考察より、盤面の大きさを一辺 $b = 45$ として実験を行った。その他の実験のパラメータはすべて実験 1 と同じである。8000 ~ 10000 ゲームの結果の平均は、従来の Prey が段階的視覚を持たずランダムに行動選択する場合の捕獲成功率が 0.490、捕獲成功ターン数が 394.9 ターンであったのに対し、本論文の段階的視覚を持ち Q 表を基に行動選択する Prey の場合、捕獲成功率が 0.487、捕獲成功ターン数が 404.4 ターンとなった。

盤面を大きくすると、Hunter にとっての捕獲成功率が極わずかに減少し捕獲成功ターン数が少し大きくなること分かった。このような結果となった理由は、実験 1 の考察が正しく、逃げるスペースがあれば Prey は Hunter から逃げるような行動を学習できているからではないかと考えられる。

[実験 3]

実験 1 と実験 2 それぞれについて、Hunter が推測を行う場合についてもシミュレーションを行う。推測の成功報酬 λ を 0.01 とする以外のパラメータはすべて実験 1、実験 2 と同じである。8000 ~ 10000 ゲームの結果の平均は、実験 1 の設定の場合 ($b = 15$)、Prey が段階的視覚を持たずランダムに行動選択する場合の捕獲成功率が 0.893、捕獲成功ターン数が 227.6 ターンであったのに対し、本論文の段階的視覚を持ち Q 表を基に行動選択する Prey の場合、捕獲成功率が 0.916、捕獲成功ターン数が 264.9 ターンとなった。実験 2 の設定の場合 ($b = 45$)、Prey が段階的視覚を持たずランダムに

行動選択する場合の捕獲成功率が 0.567、捕獲成功ターン数が 309.6 ターンであったのに対し、本論文の段階的視覚を持ち Q 表を基に行動選択する Prey の場合、捕獲成功率が 0.593、捕獲成功ターン数が 313.3 ターンとなった。

Hunter が推測を行う場合、どちらの場合でも Hunter にとっての捕獲成功率が上昇し捕獲成功ターン数が大きくなること分かった。捕獲成功率が上昇する原因として、Prey の行動がランダムでなく Q 表を基に ϵ -greedy 方策で行動を選択するようになったため、逃げる行動がパターン化したことで、Hunter が Prey の行動の規則性を推測により学習したからではないかと考えられる。捕獲成功ターン数が大きくなった理由は、Prey が視覚内の Hunter から逃げる行動を学習したため、その行動によって捕獲されるまでにターン数が多くかかるようになったからであると考えられる。

5. まとめ

実数値環境の段階的視覚を用いた追跡問題に段階的視覚を持ち Q 表を基に行動選択する Prey を導入し実験を行った。その結果、Hunter が推測を行わず盤面が十分に大きい場合、Hunter にとっての捕獲成功率、捕獲成功ターン数ともに性能が下降するが、Hunter が推測を行う場合、Hunter にとっての捕獲成功率は上昇するが、捕獲成功ターン数が大きくなること分かった。

今後の研究では、Prey への報酬の与え方や Q 表の構成を見直すことで Hunter にとっての成功率が低くなるようにしたい。また、Prey の段階的視覚を Hunter と

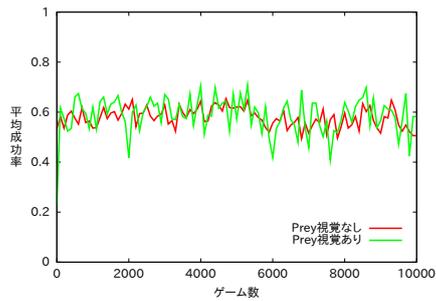


Fig. 17: 実験 3—平均捕獲成功率 ($b=45$)

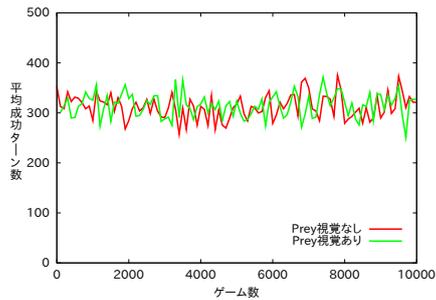


Fig. 18: 実験 3—平均捕獲成功ターン数 ($b=45$)

異なるものにする事で、捕獲することと逃げることにそれぞれに適した視覚をシミュレーションしたい。

参考文献

- [1] M. Benda, V. Jagannathan and R. Dodhiawalla: “On Optimal Cooperation of Knowledge Source,” *Technical Report BCS-G2010-28*, Boeing AI Center (1985)
- [2] 山村, 馬野, 瀬田, 岡田: “段階的な視覚をもつマルチエージェントにおける強化学習に関する研究”, 第 19 回ファジィシステムシンポジウム講演論文集, pp.73–76 (2002)
- [3] 山村, 馬野, 瀬田: “段階的な視界をもつマルチエージェントにおける強化学習について—追跡問題を例にして—”, 日本知能情報ファジィ学会誌, Vol.18, No.4, pp.561–570 (2006)
- [4] 馬野, 藤井, 小路, 細谷: “実数値環境の追跡問題におけるエージェントへの段階的視覚の適用”, 第 28 回ファジィシステムシンポジウム講演論文集, pp.1000–1005 (2012)
- [5] 馬野, 山口, 藤井: “実数値環境の追跡問題における段階的視覚の構成—実際の人間の視覚の特徴の表現—”, 第 29 回ファジィシステムシンポジウム講演論文集, pp.742–747 (2013)
- [6] P.Y. Glorennec and L. Jouffe: “Fuzzy Q-Learning,” *Proc. of the Sixth IEEE International Conference on Fuzzy Systems*, pp.659–662 (1997)
- [7] 馬野, 上森, 山村, 瀬田: “実数値環境下におけるマルチエージェントの強化学習—数値実験結果とその検討—”, 第 23 回ファジィシステムシンポジウム講演論文集, pp.295–298 (2007)
- [8] 馬野, 小路, 瀬田: “実数値環境下におけるマルチエージェントの強化学習—青エージェントに対する観測情報を用いた学習—”, 第 24 回ファジィシステムシンポジウム講演論文集, pp.897–898 (2008)
- [9] C.J.C.H. Watkins: *Learning from Delayed Rewards*, Ph.D. Thesis, Cambridge University (1989)
- [10] 橋本: 視覚情報とヒューマンエラー, 海文堂出版 (1996)
- [11] 樋渡: 視覚とテレビジョン, 日本放送出版協会 (1968)
- [12] 馬野, 山口: “実数値環境の追跡問題における段階的視覚の構成—視覚外に出た Prey の位置に対する推測機能の導入—”, 第 30 回ファジィシステムシンポジウム講演論文集, pp.544–549 (2014)

[問い合わせ先]

〒599-8531 堺市中区学園町 1-1
 大阪府立大学 大学院理学系研究科
 情報数理科学専攻
 山口 昂彦
 TEL: 072-254-9115 内線:3266
 FAX: 072-254-9163
 E-mail: yamaguchi@marron.cias.osakafu-u.ac.jp

カーレースゲームへの2つのQ表を用いたファジィQ学習の適用 – 次の目標を通過しやすい行動の学習 –

○阪口和馬 馬野元秀 立野宏樹 (大阪府立大学)

Application of Fuzzy Q-learning with Two Tables to Car Racing Game – The Learning Method to Pass Next Waypoints Efficiently

*K. Sakaguchi and M. Umamo and H. Tachino (Osaka Prefecture University)

Abstract Car racing game is a game of computer programs in IEEE CEC 2007 Car Racing Competition, where two car agents compete with each other for passing through waypoints in a two-dimensional real-number plane. In a competitive game, it is important for a car agent not only to pass through the current waypoint quickly but also to pass through the current waypoint so as to pass easily through the next one. We have applied a fuzzy Q-learning with weighted two Q-tables to a car racing game to learn such two different actions. We have simulations for various weights of Q-tables and show results.

Key Words: Fuzzy Q-learning, Car racing game, two Q-tables

1. はじめに

カーレースゲーム¹⁾とはIEEE2007 Car Racing Competitionのコンピュータプログラム同士の対戦ゲームで、2台のカーエージェントが2次元平面上に置かれた目標を目指しながら得点を競うものである。

このCompetitionでは、カーエージェントの行動選択を実現する方法として、ファジィ集合のメンバシップ関数を変化させながら、行動と状態に基づく学習を行う手法²⁾など様々な方法が用いられている。我々もファジィQ学習³⁾を用いて行動を学習する方法⁴⁾やファジィルールの調整に実数値遺伝的アルゴリズムを適用した手法⁵⁾などを提案してきた。

この対戦ゲームで相手に勝つための戦略として様々なものが考えられる。現在の目標に対する通過しやすさが相手と同程度の場合、現在の目標をとにかく早く通過することを優先し、現在の目標に対する通過しやすさが相手より明らかに大きい場合、現在の目標を早く通過することよりも、次の目標を通過しやすいように現在の目標を通過することを優先させる戦略などが考えられる。

我々は現在の目標を通過するための行動と次の目標を通過しやすいように回り込む行動という異なる性質をもつ行動を学習するために、単独走行のカーレースゲームに一定値で重み付けた2つのQ表を用いたファジィQ学習を適用した⁶⁾。次の目標を通過しやすいように現在の目標を通過する場合、目標を通過する直前の行動よりも目標を目指し始めた直後の行動の方が重要であると考えられる。

そこで、本論文では、このような学習を行うために

現在の状況に応じて2つのQ表の重みを変化させる手法を用いて、さまざまなシミュレーションを行い、その結果を報告する。

2. ファジィQ学習の適用

まず、カーレースゲームとファジィQ学習について説明する。

2.1 カーレースゲーム

カーレースゲームはFig. 1のように、2台のカーエージェントが2次元平面上に配置された目標を目指して行動し、得点を競うものである。目標には1、2、3と番号がつけられており、カーエージェントが目標1を通過すると、そのカーエージェントに得点が1点加算され、目標1は盤面から消える。そして目標2が目標1に、目標3が目標2に変更されて、新たな目標3が盤面上の決められた範囲内にランダムに出現する。盤面には壁などはなく、カーエージェントが場外に出た場合でもそのまま走り続ける。

各カーエージェントは1ステップごとに行動を選択し、既定の行動回数に達したとき、ゲームは終了する。ゲーム終了時点で得点が多い方のカーエージェントをそのゲームの勝者とする。

カーエージェントは1ステップごとに、次の情報を得ることができる。

- 自分の座標、速度、方向、角速度
- 対戦相手の座標、速度、方向、角速度
- 目標1の座標
- 目標2の座標

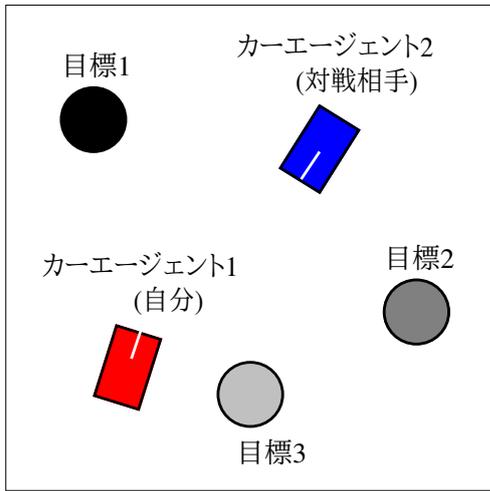


Fig. 1: カーレースゲームの概観

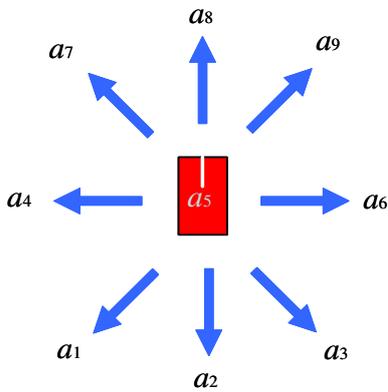


Fig. 2: カーエージェントの行動

- 自分と目標1までの距離、角度
- 自分と目標2までの距離、角度

座標と速度は2次元ベクトルであり、目標までの角度は $-\pi$ から π でカーエージェントが向いている方向を基準とした相対角度として与えられる。

カーエージェントの行動として、加速と操舵がある。加速は「正」、「そのまま」、「負」の3通りで、操舵は「右」、「そのまま」、「左」の3通りで、カーエージェントは Fig. 2 のような、それらの組み合わせ $3 \times 3 = 9$ 通りから1つを選択する。

ここでは、速さを速度ベクトルの大きさとする。速さが0ならば停止で、負ならばバックしていることになる。正で1回加速すると0.4増え、操舵すると、摩擦などを考慮した値となる。加速しないと減速していき、負で加速するとバックする。

2.2 ファジィQ学習

本研究では、対戦相手がいない状況でのゲームにおけるカーエージェントの行動選択を学習によって獲得することを目的とする。カーエージェントが目標を通

Table 1: Q表

		状態		行動				
D_i	A_j	S_k	s_{ijk}	a_1	a_2	...	a_9	
D_1	A_1	S_1	s_{111}	0.21	0.78	...	0.1	
		S_2	s_{112}	0.24	0.31	...	0.3	
		\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	
	A_2	S_1	s_{121}	0.5	0.21	...	0.9	
		S_2	s_{122}	0.01	0.45	...	0.35	
		\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots		
D_2	A_1	S_1	s_{211}	0.21	0.46	...	0.2	
		S_2	s_{212}	0.5	0.4	...	0.84	
		\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	
	A_2	S_1	s_{221}	0.25	0.3	...	0.14	
		S_2	s_{222}	0.7	0.3	...	0.7	
		\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots		

過した時に初めて、これまでの行動を評価することができる。したがって、学習方法は強化学習が適していると考えられ、その中でもQ学習⁷⁾を用いる。さらに、カーレースゲームは実数値環境であり、通常のQ学習では状態をうまく決められないので、ファジィQ学習³⁾を適用する。

ファジィQ学習では、Q学習の状態空間をファジィ集合で定義し、ファジィ集合と一致した状態で複数の行動が選択され、一致した度合いとその行動から全体的な行動を決定する。行動した結果から得られる報酬と次状態の最大Q値を用いて、一致した度合いに応じて複数のQ値を更新する。

(1) ファジィ状態空間とQ値

状態空間の属性として「目標1までの距離」、「目標1との角度」、「カーエージェントの速さ」の3つを用いて、それぞれのファジィ集合を Fig. 3、Fig. 4、Fig. 5 のように定義する。「目標1までの距離」では新たな目標が出現する正方形の範囲の対角線の長さを1としている。「目標1との角度」ではカーエージェントの正面を0、左側を正、右側を負としている。このように状態空間をファジィ集合で定義したものをファジィ状態空間と呼ぶ。

全ての状態において、それぞれの行動に対する価値を示すQ値を与え、その組み合わせを Table 1 のようなQ表で表す。なお、Q値の初期値はすべて0.1にしている。

(2) 行動決定

行動決定は、状態と行動の組合せに対して与えられているQ表を基に行う。状態は Fig. 3 から Fig. 5 で

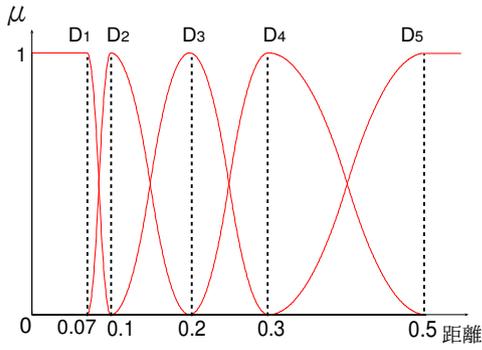


Fig. 3: 目標 1 との距離のファジィ集合

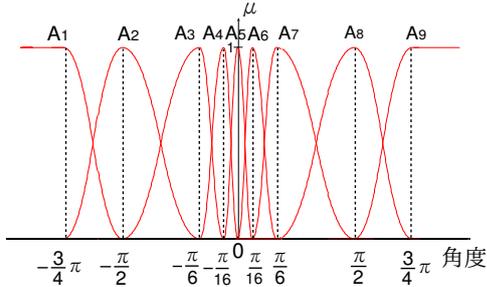


Fig. 4: 目標 1 との角度のファジィ集合

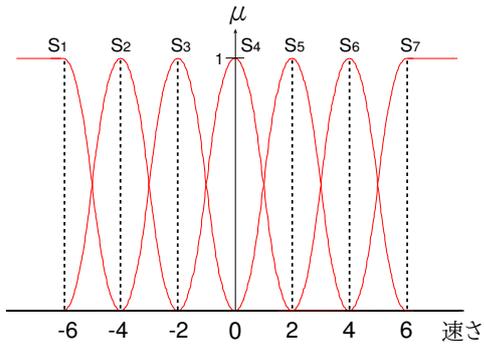


Fig. 5: カーエージェントの速さのファジィ集合

定義したファジィ集合 D_i , A_j , S_k の組合せとなるので、これを s_{ijk} で表す。

まず、環境から得られた情報を基に全ての状態 s_{ijk} の一致度 μ_{ijk} を求める。状態の一致度は、状態の各属性のファジィ集合のメンバーシップ値の積である。そして、正の一致度を持つ各状態について ϵ グリーディ法により行動を 1 つ選択する。各状態の一致度をベクトルの大きさと考えて複数の行動を合成し、そのベクトルに最も近い行動を $a_1 \sim a_9$ から選び、全体的な行動とする。

(3) Q 値の更新

次状態 s_{t+1} により報酬 r を得る。時刻 t の状態 s_t の行動 a_t に対する Q 値 $Q(s_t, a_t)$ は、次状態 s_{t+1} の報酬 r と次状態の最大 Q 値から現在の Q 値を引いた値を状態 s_t の一致度 μ_{s_t} に応じて分配するので、次の式を用いて更新する。

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \mu_{s_t} \alpha [r + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t)] \quad (1)$$

ここで、 α は学習率、 γ は割引率である。また、 $\max_{a'} Q(s_{t+1}, a')$ は、次状態 s_{t+1} の各状態 s'_{t+1} の最大 Q 値を一致度 $\mu_{s'_{t+1}}$ で重み付けたもので、次の式で与えられる。

$$\max_{a'} Q(s_{t+1}, a') = \sum_{s'_{t+1}} \mu_{s'_{t+1}} \max_{a'} Q(s'_{t+1}, a') \quad (2)$$

2.3 2つの Q 表を用いたファジィ Q 学習

自分の方が相手よりも明らかに現在の目標を通過しやすい場合、現在の目標を早く通過することよりも、次の目標を通過しやすいように現在の目標を通過することを優先する戦略が考えられる。現在の目標を通過するための行動と次の目標を通過しやすいように回り込む行動という性質の異なる行動を学習させるために、2種類の報酬に応じて2つの Q 表を用いるファジィ Q 学習⁶⁾を適用する(通常の Q 学習では、複数の目的を学習する Q 学習⁸⁾が提案されている)。2つの Q 表を用いるファジィ Q 学習では2つの Q 表に重みを定めておき、重みとそれぞれの Q 表から得られる行動から最終的な行動を決定する。行動した結果から得られる報酬を用いて、Q 表の重みに応じて2つの Q 表の Q 値を独立に更新する。

(1) ファジィ状態空間と Q 値

本研究では、2つの Q 表のファジィ状態空間のファジィ集合は共通とする。また、2つの Q 表を Q_1 表と Q_2 表と呼び、それぞれの Q 値を $Q_1(s_t, a_t)$, $Q_2(s_t, a_t)$ で表す。なお、それぞれの Q 表の Q 値の初期値はすべて 0.1 としている。

(2) 行動決定

それぞれの Q 表についてファジィ Q 学習と同じ方法で行動決定し、状態の一致度をベクトルの大きさとして合成した行動のベクトルを求める。Q 表の重みをベクトルの大きさとして、それらのベクトルをさらに合成し、そのベクトルに最も近い行動を $a_1 \sim a_9$ から選び、全体的な行動とする。

(3) Q 値の更新

次状態 s_{t+1} から Q_1 表、 Q_2 表についてそれぞれ報酬 r_1 , r_2 が与えられる。時刻 t の状態 s_t の行動 a_t に対する Q 値 $Q(s_t, a_t)$ は、次状態 s_{t+1} の報酬 r と次状態の最大 Q 値から現在の Q 値を引いた値を一致度 μ_{s_t} と重み (w_1, w_2) に応じて分配する。

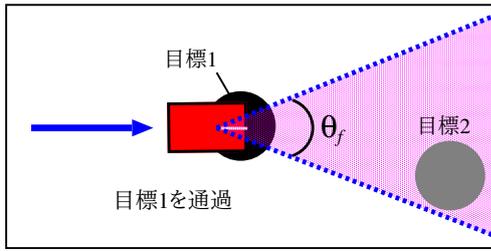


Fig. 6: 前方の定義

$$Q_i(s_t, a_t) \leftarrow Q_i(s_t, a_t) + w_i \mu_{s_t} \alpha_i [r_i + \gamma_i \max_{a'} Q_i(s_{t+1}, a') - Q_i(s_t, a_t)] \quad (3)$$

ここで、 $i = 1, 2$ で、 α_i 、 γ_i はそれぞれ Q_i 表に対する学習率、割引率であり、 $\max_{a'} Q_i(s_{t+1}, a')$ は、 Q_i 表について次状態 s_{t+1} の各状態 s'_{t+1} での最大 Q 値を一致度 $\mu_{s'_{t+1}}$ で重み付けたもので、次の式で与えられる。

$$\max_{a'} Q_i(s_{t+1}, a') = \sum_{s'_{t+1}} \mu_{s'_{t+1}} \max_{a'} Q_i(s'_{t+1}, a') \quad (4)$$

3. 次の目標を通過しやすいように現在の目標を通過する行動

目標 2 を通過しやすいように目標 1 を通過する行動を、目標 1 を通過したときに目標 2 がカーエージェントの前方に位置する行動であると考えた⁹⁾。本研究では、Fig. 6 のように目標 2 がカーエージェントの前方の一定の角度 θ_f 以内であれば前方とみなし、この論文では $\theta_f = \frac{\pi}{4}$ とする。

また、学習中に目標 2 の位置も考慮する必要があるため、ファジィ Q 学習に用いる属性に「目標 2 までの距離」と「目標 2 との角度」を追加し、用いるファジィ集合を Fig. 7、Fig. 8 のように定義した。

前の文献⁶⁾では、 Q_1 表を目標 1 を通過する行動を、 Q_2 表を目標 2 が前方にあるように回り込む行動を学習するための Q 表とし、2 つの Q 表に対する重み w_1 と w_2 を用いてそれぞれの Q 表から得られる行動を合成し、1 つの行動を決定した。このとき、 w_1 と w_2 は一定の値に設定したが、本論文では様々な状況に応じて重みを変えて数値実験を行う。

3.1 ステップ数に応じて重みを変化させる方法

目標 1 を通過する行動に関しては、目標を目指し始めてから目標を通過するまでのすべての行動が重要であると考えられるが、目標 2 が前方にあるように回り込む行動に関しては、目標を通過するまでの行動のうち、初めの方の行動が重要だと考えられる。そこで、Fig. 9 のように現在のステップ数に応じて、重み w_2 の値を線形に減少させる。

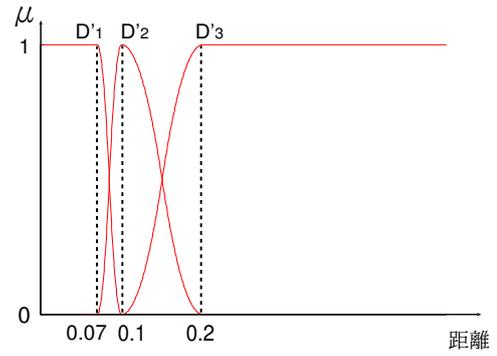


Fig. 7: 目標 2 との距離のファジィ集合

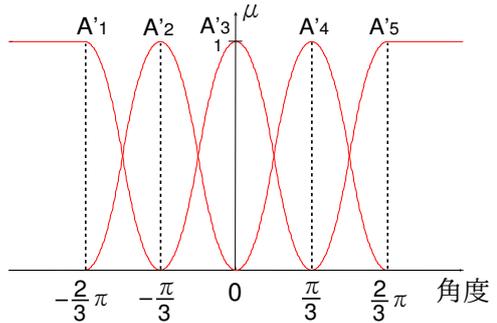


Fig. 8: 目標 2 との角度のファジィ集合

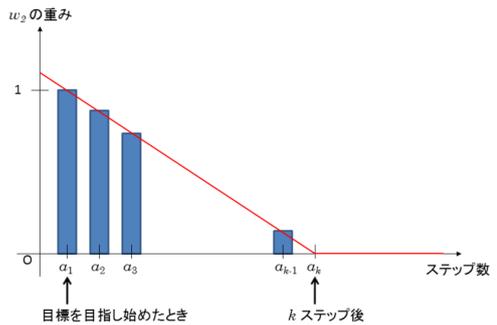


Fig. 9: ステップ数に応じた重みの値

[実験 1]

それぞれの Q 表に対して次のように報酬を与える。 Q_1 表では目標 1 を通過すれば通過報酬 $r_1 = 1$ を、そうでなければ失敗報酬 $r_1 = 0$ を与え、 Q_2 表では目標 1 を通過したときに目標 2 が前方にあれば向き報酬 $r_2 = 1$ を、そうでなければ失敗報酬 $r_2 = 0$ を与える。失敗報酬は目標を目指し始めてから通過するまでの間も両方の Q 表に対して常に与えられる。また、重み w_2 が 0 になる点 $k = 100, 300, 500$ として、学習を 5 万ゲーム行う。そして、学習の性能を調べるために学習を 100 ゲーム行うごとに学習を行わずに 100 ゲーム試行し、次の 3 つの値を調べる。

- 目標 1 の通過回数 (1 ゲームあたり)
- 前方に目標 2 が位置するように目標 1 を通過できた回数 (1 ゲームあたり)
- B の A に対する割合

以上の試行を1セットとして10セット行い、AとBのセット平均の推移を Fig. 10 ~ Fig. 12 に示す。赤のグラフが目標1の通過回数のセット平均を、青のグラフが前方に目標2が位置するように目標1を通過できた回数のセット平均を表している。また、BのAに対する割合の推移を Fig. 13 に示す。学習のパラメータとして、学習率 α_1 、 α_2 をともに0.1、割引率 γ_1 、 γ_2 をともに0.9、 ϵ を学習を行っているときは0.001、学習を行っていないときは0とし、ゲーム開始時のカーエージェントの初速度は0とする。

実験の結果、 k が大きくなるほど目標1の通過回数が大きくなっていることが分かる。これは k が大きくなるほど目標1を通過する行動に対する重み w_1 が全体的に大きくなるからだと考えられるので、妥当な結果であるといえる。また、 k が小さいとき、目標2が前方にあるように回り込む行動があまり学習できていないことが分かる。これは向き報酬が与えられるときの重み w_2 がとても小さくなっており、その結果、Q値の更新量が小さくなり、その影響が伝播していくのが遅いからであると考えられる。本来、目標を目指し始めたときの行動に報酬を与えるために重みを変化させたが、その学習がうまく行われていないといえる。

[実験 2]

向き報酬を与える機会を多くするために、 Q_1 表に関しては通常の更新を行い、 Q_2 表に関しては向き報酬 r_2 を獲得したとき、目標を目指し始めてから目標1を通過するまでの全ての経路に対して、次の式でQ値を更新する。

$$Q_2(s_t, a_t) \leftarrow Q_2(s_t, a_t) + w_2 \mu_{s_t} \alpha_2 r_2 \quad (5)$$

常にQ値は更新され続けているので、Q値の更新は完了していると考え、重みづけた報酬分のみをQ値に加えている。この方法を用いて実験を行った結果を Fig. 14 ~ Fig. 16 と Fig. 17 に示す。また、学習のパラメータや実験方法は実験1と同じである。

実験の結果、実験1の結果よりも目標1の通過回数も目標2が前方にあるように目標1を通過した割合も下がってしまった。これは目標1を通過したとき目標2が前方にあれば報酬が与えられるが、目標1を通過するまでの経路の良し悪しの評価がなされていないためによくない経路に関して大きくQ値が更新されてしまったからだと考えられる。

(注) 以上の実験では、目標1を通過していないときに2つのQ表に対して失敗報酬0を与えていた。目標1を通過する行動を学習するためにこの失敗報酬を与えるのは妥当であると考えられるが、目標2が前方に位置するように回り込む行動を学習するためにこの失敗

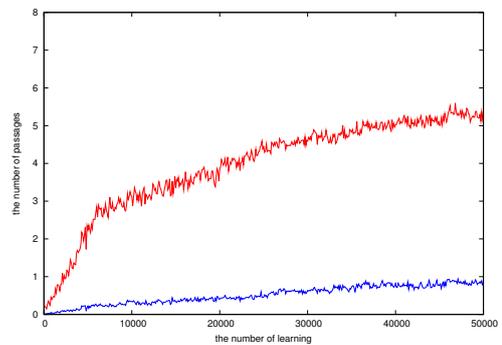


Fig. 10: 実験1の結果 ($k = 100$).

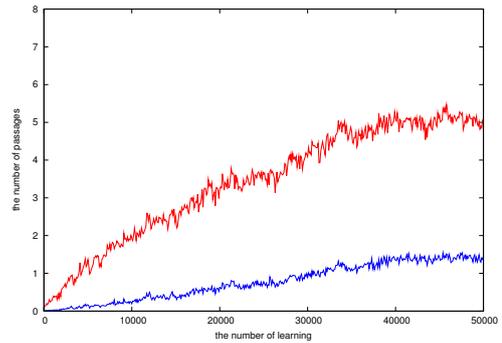


Fig. 11: 実験1の結果 ($k = 300$).

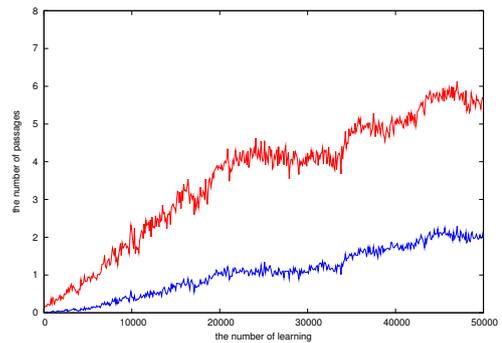


Fig. 12: 実験1の結果 ($k = 500$).

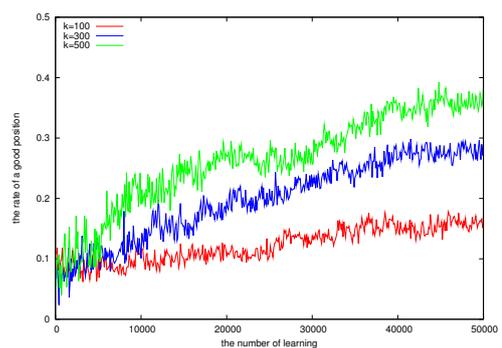


Fig. 13: 実験1の結果 (BのAに対する割合).

報酬を与える必要はないと考えられる。そこで、目標1を通過したときにのみ成功報酬か失敗報酬を与えるという方法が考えられる。しかし、Q学習では、常にQ値を更新しないと、Q値の更新の結果が伝播していない。報酬が与えられないときは報酬を現在のQ値

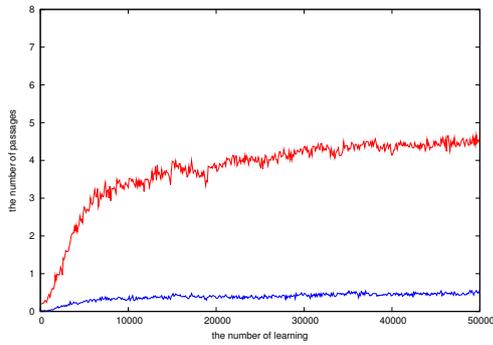


Fig. 14: 実験 2 の結果 ($k = 100$).

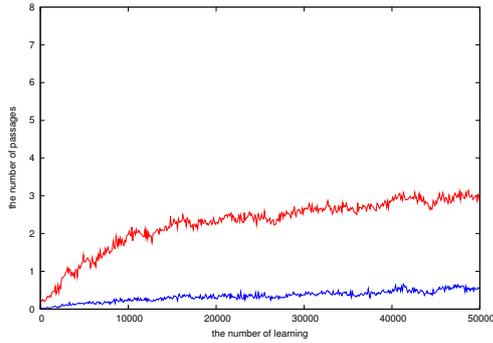


Fig. 15: 実験 2 の結果 ($k = 300$).

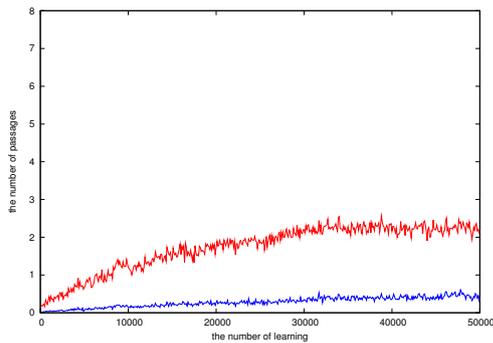


Fig. 16: 実験 2 の結果 ($k = 500$).

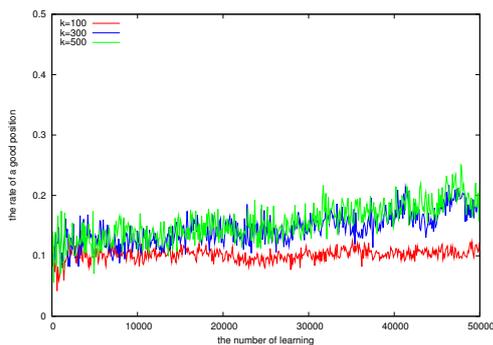


Fig. 17: 実験 2 の結果 (B の A に対する割合).

と考える、次のような更新式を用いて、次状態の Q 値だけで更新する方法が考えられる。

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \mu_{s_t} \alpha \gamma \max_{a'} Q(s_{t+1}, a') \quad (6)$$

これについては、今後シミュレーションを行い、検討していきたい。

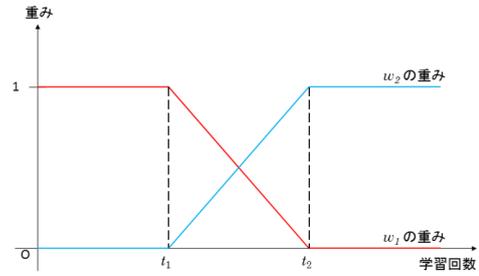


Fig. 18: 学習回数に応じた重みの量

3.2 学習回数に応じて重みを変化させる方法

目標 2 が前方にあるように回り込む行動を学習する前に、まず目標 1 を通過する行動という基本的な行動を学習すべきであると考えられる。すなわち、学習の序盤 ($0 \sim t_1$) では学習目標 1 を通過する行動のみを学習させ、中盤 ($t_1 \sim t_2$) で目標 2 が前方にあるように回り込む行動を徐々に学習させていき、終盤 ($t_2 \sim$) は目標 2 が前方にあるように回り込む行動のみを学習させていくような方法が考えられる。そこで、Fig. 18 のように学習回数に応じて重み w_2 の値を線形に変化させる。

[実験 3]

学習の序盤、中盤、終盤の割合を 1:1:1, 1:2:1, 1:2:2 としたとき、すなわち、学習回数 5 万回に対して、

$$(t_1, t_2) = \left(\frac{50000}{3}, \frac{100000}{3}\right), \left(\frac{50000}{4}, \frac{150000}{4}\right), \left(\frac{50000}{5}, \frac{150000}{5}\right)$$

としたときの結果を Fig. 19 ~ Fig. 21 と Fig. 22 に示す。このとき、その他のパラメータや実験方法は実験 1 と同じである。

実験の結果、学習の序盤では目標 1 の通過回数のみが大きく増加しており、目標 1 を通過できる学習が行われており、妥当な結果であるといえる。学習の中盤では目標 2 が前方にあるように目標 1 を通過できた割合が増加していることが分かるが、目標 1 の通過回数も大きく減少している。これは目標 2 が前方に位置するように回り込む行動と目標 1 を早く通過する行動とは大きく異なる (相反する) ためであると思われる。しかし、目標 1 を通過する行動を用いているにも関わらず、目標 1 の通過回数は下がりすぎているようにも思える。学習の終盤では目標 1 の通過回数が大きく回復しており、目標 2 が前方に位置するように回り込む割合の値も保たれている。この実験では基本的な行動から順にうまく学習できているといえそうである。

[実験 4]

Q_1 表について中盤や終盤に関しては学習をし終えたと考えて、Q 値の更新は行わずに (行動決定は前回と同様に行つて) 実験を行い、その結果を Fig. 23 ~ Fig. 25 と Fig. 26 に示す。また、その他のパラメータや実験

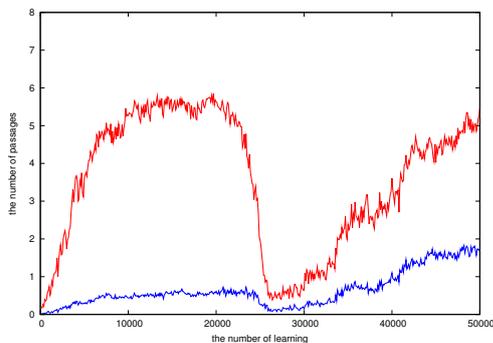


Fig. 19: 実験 3 の結果 $(t_1, t_2) = (\frac{50000}{3}, \frac{100000}{3})$.

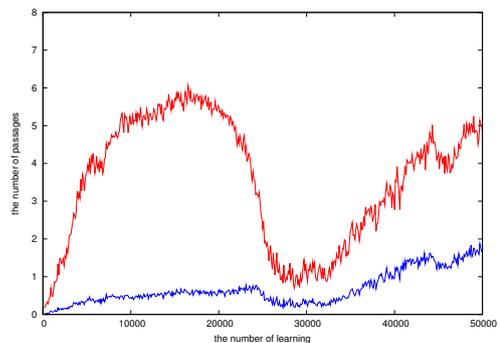


Fig. 23: 実験 4 の結果 $(t_1, t_2) = (\frac{50000}{3}, \frac{100000}{3})$.

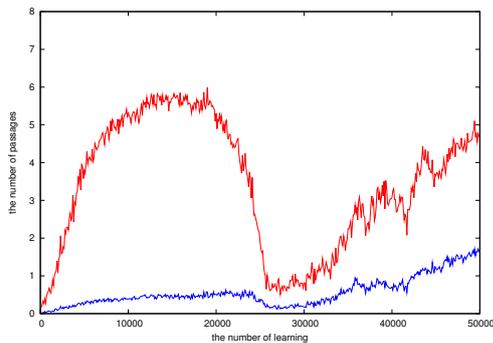


Fig. 20: 実験 3 の結果 $(t_1, t_2) = (\frac{50000}{4}, \frac{150000}{4})$.

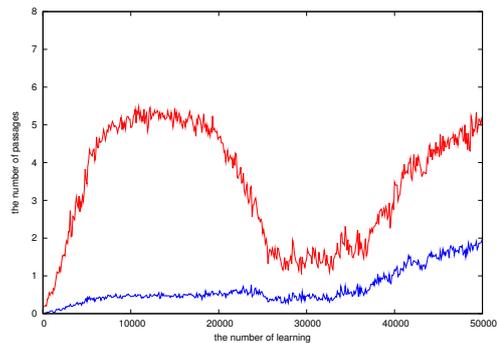


Fig. 24: 実験 4 の結果 $(t_1, t_2) = (\frac{50000}{4}, \frac{150000}{4})$.

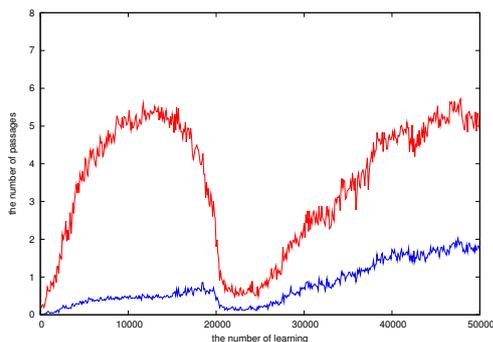


Fig. 21: 実験 3 の結果 $(t_1, t_2) = (\frac{50000}{5}, \frac{150000}{5})$.

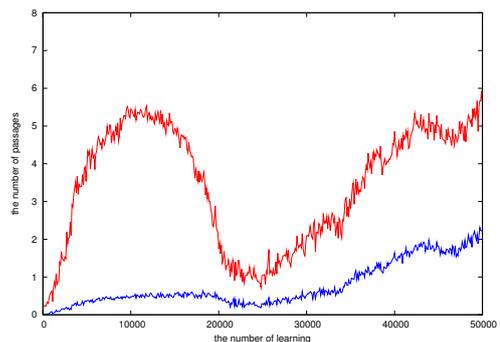


Fig. 25: 実験 4 の結果 $(t_1, t_2) = (\frac{50000}{5}, \frac{150000}{5})$.

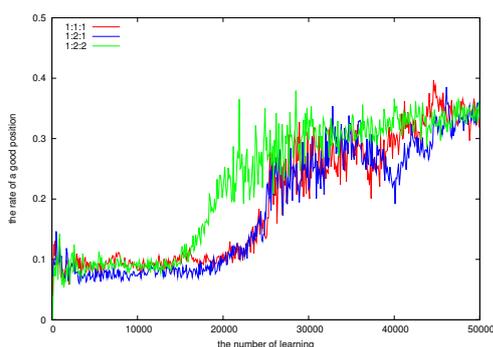


Fig. 22: 実験 3 の結果 (B の A に対する割合).

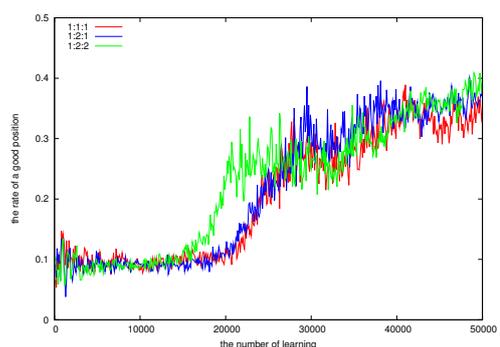


Fig. 26: 実験 4 の結果 (B の A に対する割合).

方法は実験 1 と同じである。

実験の結果、実験 3 の結果と比べて、学習の中盤と終盤の目標 1 の通過回数が少し多くなっていることが分かる。これは学習済みの Q_1 表を更新していないので、実験 3 よりも目標 1 を通過する行動の影響が大き

いからであると考えられ、目的どおりの学習が行えたといえる。また、学習の序盤で基本的な行動を学習する回数よりも、中盤や終盤で複雑な行動な行動を学習する回数を多くした方が学習が早く進んでいることが分かる。特に、今回の実験では学習の序盤、中盤、終盤

を、1:2:2としたときが目標1の通過回数と目標2が前方に位置するように回り込む割合の両方が高い結果となった。他の実験と比較して、目標2が前方にあるように回り込むという行動をうまく学習できたといえる。

4. おわりに

この論文では、カーレースゲームにおいて異なる性質をもつ行動を学習するために、状況に応じて重みを変化させる2つのQ表を用いたファジィQ学習を適用し、いくつかの状況に対して数値実験を行った。その結果、まず基本的な行動を学習して複雑な行動を学習する手法が有用であることが分かった。

今後はそれぞれのQ表で異なるファジィ状態空間を用いて学習を行ったり、ファジィ集合のチューニングを行うことで、2つのQ表を用いたファジィQ学習の手法を改善していきたい。また、単独走行では今回の実験で得られた行動の評価が十分にできないので、実際に対戦ゲームに適用していきたい。

参考文献

- 1) IEEE CEC 2007 Car Racing Competition: <http://julian.togelius.com/cec2007/>
- 2) D.T. Ho and J.M. Garibaldi: "A Fuzzy Approach for the 2007 CIG Simulated Car Racing Competition," *2008 IEEE Symposium on Computational Intelligence and Games*, pp.127-134 (2008)
- 3) P.Y. Glorennec and L. Jouffe: "Fuzzy Q-Learning," *Proc. of the Sixth IEEE International Conference on Fuzzy Systems*, pp.659-662 (1997)
- 4) 馬野, 岸本, 瀬田: 「カーレースゲームへのファジィQ学習の適用」, 第26回ファジィシステムシンポジウム講演論文集, pp.16-21, (2010)
- 5) 馬野, 伊瀬, 瀬田: 「カーレースゲームにおける実数値遺伝的アルゴリズムによるファジィルールのチューニングについて」, 第27回ファジィシステムシンポジウム講演論文集, pp.569-574 (2011)
- 6) 馬野, 阪口, 立野: 「カーレースゲームへのファジィQ学習の適用—2つのQ表を用いたファジィQ学習—」, 第30回ファジィシステムシンポジウム講演論文集, pp.556-561 (2014)
- 7) R.S. Sutton and A.G. Barto: *Reinforcement Learning*, A Bradford Book, 1998, 三上, 皆川 訳: 強化学習, 森北出版, (2000)
- 8) Z. Gabor, Z. Kalmar and C. Szepesvari: "Multi-criteria Reinforcement Learning," *Proc. of the Fifteenth International Conference on Machine Learning*, pp.197-205 (1998)
- 9) 馬野, 立野, 岸本, 伊瀬: 「カーレースゲームへのファジィQ学習の適用—一次の目標の通過のしやすさを優先した学習—」, 第29回ファジィシステムシンポジウム講演論文集, pp.1006-1011, (2013)

[問い合わせ先]

〒 599-8531 堺市中区学園町 1-1
大阪府立大学 大学院理学系研究科
情報数理科学専攻
阪口和馬
TEL: 072-254-9115 内線 3266
FAX: 072-254-9163
E-mail: sakaguchi@marron.cias.s.osakafu-u.ac.jp

全体的傾向と局所的特徴に基づく時系列データの言葉による表現 —振動を考慮した期間のファジィ集合の決定—

○岩浅 貴大 高橋 勝稔 馬野 元秀 (大阪府立大学)

Expressions of Time Series in Natural Language with Global Trend and Local Features

— Taking Oscillation into Account for Fuzzy Intervals
in the Temporal Axis

*T. Iwaasa and K. Takahashi and M. Umamo (Osaka Prefecture University)

Abstract— We have many kinds of data of time series, which are understood via linguistic expressions by humans. We have proposed a method to express their global trend and local features in a natural language. In the previous research, we proposed a method to define fuzzy intervals based on the big turning point by grouping the adjacent segments of the small difference of values. However, it may define fuzzy intervals that do not suit the tendency of oscillation of time series. In this paper, we propose a method to group the adjacent segments of the small weighted average of the difference of values and difference of amplitudes. We show that this method defines more appropriate fuzzy intervals.

Key Words: Time Series, Oscillation, Expressions in Natural Language

1. はじめに

時系列データとは時間順に並んだ数値データのこと
で、毎日の気温の変化や株価の変化などが身近な例と
してあげられる。人間はこのような時系列データを、「全
体的に中ぐらい増加」というような言葉で表現するこ
とによって理解していると思われる。

そこで、我々は与えられた時系列データを言葉によっ
て表現する方法を提案してきた¹⁾²⁾³⁾。文献¹⁾では時
系列データの全体的な傾向を言葉で表現し、元の時系
列データと全体的傾向を表す時系列データとの差が大
きい部分を局所的特徴として言葉で表現する方法を提
案した。

しかし、文献¹⁾の方法では、データの傾向と合わな
い表現になることがある。この理由の一つとして、全
ての時系列データに対して3つの期間(前期, 中期, 後
期)を同じ定義のファジィ集合を用いていることが考
えられる。そこで、時系列データに基づいて3つの期
間を調整する方法を提案した²⁾³⁾。文献²⁾では全体的
傾向を表す時系列データと元の時系列データとの差が
小さくなるように期間を調整するが、全体的傾向を表
す時系列データと元の時系列データの差を小さくする
ための計算に多くの時間を要する。文献³⁾では時系列
データの傾向(変化量)が前後で類似しているものをま
とめていくことで3つの期間(前期, 中期, 後期)のク
ラスタを作り、クラスタの境界の時刻を基に期間を定
義する方法を提案した。この方法では、振動回数の多

い時系列データに適応できなかった。

本論文では、文献³⁾の方法を拡張して、クラスタの
振動が似ているものも考慮することにより、データの
傾向(変化量と振動)が前後で類似しているものをま
とめることで、期間のファジィ集合を定義する方法を提
案する。これにより期間のファジィ集合がよりデータ
に基づいた分割になる。

2.1 時系列データの言葉による表現

時刻 t_1 から t_n までの時系列データ $x(t_i)$ を考える。
時系列データの全体的傾向と局所的特徴の言葉による
表現は以下のような手順で行う¹⁾。例として Fig. 1 の
黒色のグラフの時系列データを用いる。

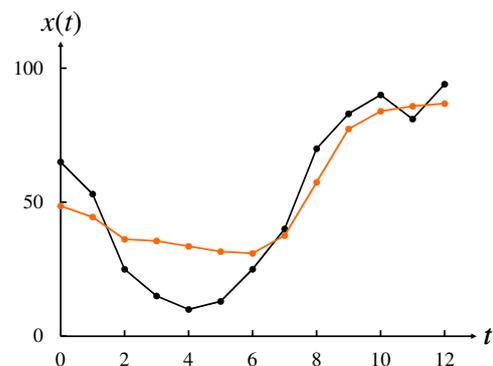


Fig. 1: 時系列データの例

Table 1: 全体的傾向を求める表

前→中	中→後	大きく減少	中ぐらい減少	少し減少	ほぼ一定	少し増加	中ぐらい増加	大きく増加
大きく減少	大きく減少	大きく減少	中ぐらい減少	中ぐらい減少	少し下に凸	下に凸	下に凸	下に凸
中ぐらい減少	大きく減少	大きく減少	中ぐらい減少	少し減少	少し下に凸	下に凸	下に凸	下に凸
少し減少	中ぐらい減少	中ぐらい減少	少し減少	ほぼ一定	少し下に凸	少し下に凸	少し下に凸	少し下に凸
ほぼ一定	中ぐらい減少	少し減少	ほぼ一定	ほぼ一定	ほぼ一定	少し増加	中ぐらい増加	中ぐらい増加
少し増加	少し上に凸	少し上に凸	少し上に凸	ほぼ一定	少し増加	中ぐらい増加	中ぐらい増加	中ぐらい増加
中ぐらい増加	上に凸	上に凸	少し上に凸	少し増加	中ぐらい増加	大きく増加	大きく増加	大きく増加
大きく増加	上に凸	上に凸	少し上に凸	中ぐらい増加	中ぐらい増加	大きく増加	大きく増加	大きく増加

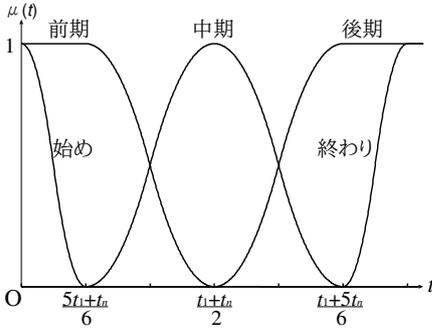


Fig. 2: 期間のファジィ集合

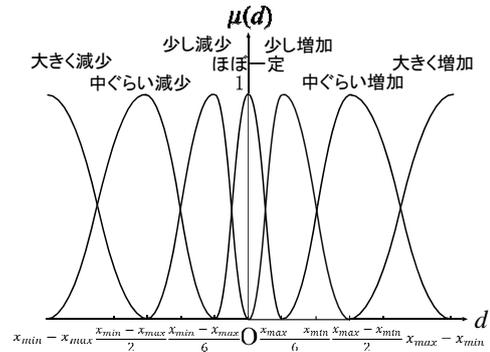


Fig. 3: 変化量のファジィ集合

(1) ファジィ集合による重み付き平均値の算出

まず, Fig. 2 のような期間のファジィ集合「前期」, 「中期」, 「後期」を用いて時系列データの各期間における代表値を求める. ここでは, 期間のファジィ集合による重み付き平均値を用い,

$$m_{\text{期間}} = \frac{\sum_{i=1}^n x(t_i) \mu_{\text{期間}}(t_i)}{\sum_{i=1}^n \mu_{\text{期間}}(t_i)} \quad (1)$$

で計算する. Fig. 1 の場合は, $m_{\text{前期}} = 47.667$, $m_{\text{中期}} = 24.652$, $m_{\text{後期}} = 82.400$ となる.

(2) 変化量のファジィ集合

前期から中期, 中期から後期への変化量を $d_{\text{前期} \rightarrow \text{中期}} = m_{\text{中期}} - m_{\text{前期}}$, $d_{\text{中期} \rightarrow \text{後期}} = m_{\text{後期}} - m_{\text{中期}}$ で計算し, Fig. 3 のような変化量のファジィ集合を用いて, 各変化量に対応する表現を求める. Fig. 3 で, x_{\min} と x_{\max} はそれぞれ時系列データのグラフの x 軸の最小値と最大値である. Fig. 1 の場合は $x_{\min} = 0$, $x_{\max} = 100$ で, $d_{\text{前期} \rightarrow \text{中期}} = -5.260$ で {0.185/少し減少, 0.815/ほぼゼロ} となり, $d_{\text{中期} \rightarrow \text{後期}} = 53.010$ で {0.988/中ぐらい増加, 0.012/大きく増加} となる.

(3) 全体的傾向

次に, Table 1 を用いて全体的傾向の表現を求める. Table 1 は縦に前期から中期への変化量のファジィ集合を, 横に中期から後期への変化量を表すファジィ集合をとり, 表の対応する要素に全体的傾向を表すファジィ

集合を入れてある. 変化量のファジィ集合のすべての組合せについて全体的傾向を求め, 各変化量のメンバシップ値の積をその全体的傾向のメンバシップ値とする. 同じ表現が得られた場合はそれらのメンバシップ値の和をとる¹⁾. Fig. 1 の場合は, {0.795/少し増加, 0.006/中ぐらい増加, 0.199/下に凸} となる.

(4) 全体的傾向を表す時系列データ

局所の特徴は, 時系列データと全体的傾向を比べ, 違いが顕著な部分であると考えられるので, 違いを見つけるために全体的傾向を表す時系列データをつくる.

時系列データは時間と値の組の集合で決まるので全体的傾向は期間と代表値の組で決まると考えられる. 今の例では {(前期, $m_{\text{前期}}$), (中期, $m_{\text{中期}}$), (後期, $m_{\text{後期}}$)} で, この各期間における代表値に Fig. 2 の期間のファジィ集合で重みをつけたものが全体的傾向の時系列データとなる. ここでは, グラフの両端を強調するために「始め」と「終わり」のファジィ集合を Fig. 2 のように定義している. 「始め」, 「前期」, 「中期」, 「後期」, 「終わり」の5つの期間を用いて,

$$x(t_i) = \frac{\sum_{\text{期間}} \mu_{\text{期間}}(t_i) \cdot m_{\text{期間}}}{\sum_{\text{期間}} \mu_{\text{期間}}(t_i)} \quad (2)$$

で計算する.

Fig. 1 の黒色のグラフの時系列データから得られる全体的傾向の時系列データは Fig. 1 の橙色のグラフと

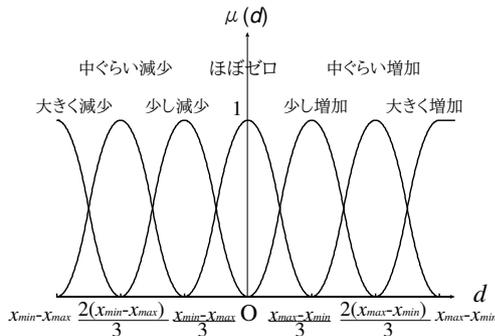


Fig. 4: 差のファジィ集合

なる。

(5) 局所的特徴

時系列データの局所的特徴は、時系列データと全体的傾向とを比べて差の絶対値が大きい部分を見つけ、Fig. 4の差のファジィ集合で表現する²⁾。Fig. 1では、 $t = 4$ すなわち {0.5/前期, 0.5/中期} において 0.173/ほぼゼロ, 0.827/少し減少となる。

(6) 言葉による表現

Fig. 1のデータの言葉による表現は、全体的傾向が {0.795/少し増加, 0.006/中ぐらい増加, 0.199/少し下に凸}, 局所的特徴が $t = 4$ で {0.173/ほぼゼロ, 0.827/少し減少} となる。

2.2 データの傾向に基づくファジィ集合の決定

これまで、どのような時系列データに対しても同じ定義の期間のファジィ集合を適用していたが、Fig. 1の場合のように、データの傾向と合わない期間のファジィ集合になっていることがあった¹⁾。そこで、データの変化量を基に時系列データの傾向が前後で類似しているものをまとめていくことにより前期、中期、後期の3つのクラスタを作り、クラスタの境界の時刻(これを変化点と呼ぶことにする)を基に期間のファジィ集合を定義する方法を提案した³⁾。

(1) データの傾向の概要

時系列データから、変化点 t_{r_1} と t_{r_2} (t_{r_1} は前期と中期のメンバシップ関数が0.5で交わる時刻, t_{r_2} は中期と後期のメンバシップ関数が0.5で交わる時刻とする)を見つけるために、データの傾向が時刻 t_i の前後で似ているものをまとめていく。文献³⁾では、データ x_i (以下、 $x(t_i)$ を x_i と略記する) について考えると、 t_i の前後の変化量が似ているものが傾向が似ているとした。このように考えると、データをまとめるというよりもデータ間の線分をまとめる方が自然である。そう

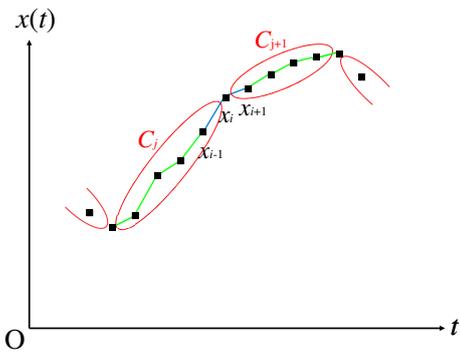


Fig. 5: クラスタの傾向の差

すると、最終的に得られる変化点はいずれかのデータ x_i の t 軸の値 t_i となる。

そこで、時系列データの隣接する点 x_i と x_{i+1} を結ぶ線分を $\langle x_i, x_{i+1} \rangle$ と表す。最初は $\langle x_1, x_2 \rangle, \langle x_2, x_3 \rangle, \dots, \langle x_{n-1}, x_n \rangle$ のそれぞれが1つのクラスタである状態から始める。これらのクラスタの変化量が前後で似ているクラスタから順にまとめていく。

(2) 変化量による類似性

t_i の前後の変化量が似ているもの、すなわち、 $x_i - x_{i-1}$ (厳密には $\frac{x_i - x_{i-1}}{t_i - t_{i-1}}$ であるが、ここでは $t_i - t_{i-1} = 1$ と考える) と $x_{i+1} - x_i$ の差が小さい (0に近いもの) を似ているものとする³⁾。

線分 $\langle x_{i-1}, x_i \rangle$ の変化量を $\Delta x_i = x_i - x_{i-1}$ とすると、時刻 t_i の前後の線分 $\langle x_{i-1}, x_i \rangle$ と $\langle x_i, x_{i+1} \rangle$ のそれぞれの変化量の差 $\Delta x_{i+1} - \Delta x_i$ が0に近いほど、時刻 t_i の前後での傾向の変化が小さいと考える。すなわち、線分の変化量の差の絶対値 $|\Delta x_{i+1} - \Delta x_i|$ により、時刻 t_i の前後での傾向の差を知ることができる。

クラスタ内の線分が複数になった場合、クラスタ同士の傾向の差を考える必要がある。線分 $\langle x_i, x_{i+1} \rangle, \langle x_{i+1}, x_{i+2} \rangle, \dots, \langle x_{i+k-1}, x_{i+k} \rangle$ からなるクラスタを $\langle x_i, x_{i+1}, \dots, x_{i+k} \rangle$ で表すことにする。

クラスタが大きくなって Fig. 5 のようになったとする。赤で囲んだ部分がクラスタで、Fig. 5ではデータ x_i の前にクラスタ C_j 、後に C_{j+1} がある状態を表している。このとき、文献³⁾ではクラスタの C_j と C_{j+1} の傾向の差を「クラスタの端の線分の変化量の差を利用する方法」と「クラスタ全体の変化量の平均を利用する方法」で提案した。以後の実験では、「クラスタ全体の変化量の平均を利用する方法」を用いている。

Fig. 5の緑色の線分がクラスタ全体の線分である。それらの変化量の平均がクラスタの傾向を表すと考え、変化量の平均の差が小さければ、前後のクラスタの傾向は似ていると考える。クラスタ同士をまとめる度に、クラスタの変化量の平均は変わる。

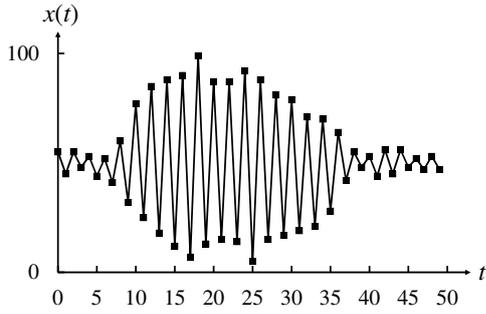


Fig. 6: 時系列データ 2

l 個の線分を持つクラスタを C_j , m 個の線分を持つクラスタを C_{j+1} とすると, これらのクラスタがまとまる場合, クラスタの変化量は

$$\frac{C_j \text{ の変化量} \times l + C_{j+1} \text{ の変化量} \times m}{l + m}$$

で求められる.

Fig. 1 の時系列データを考えると, 文献³⁾の手法では変化点が $t_{r_1} = 2, t_{r_2} = 10$ で, 全体的傾向が $\{0.927/\text{少し下に凸}, 0.073/\text{下に凸}\}$ となる. 文献¹⁾の手法では, どのような時系列データが与えられても固定の期間を用いていたが, データの傾向(変化量)に基づいて期間を適切に変更することで, より妥当な表現を得ることができるようになった.

しかし, Fig. 6 の時系列データの場合を考えてみると, 文献³⁾の手法では変化点が $t_{r_1} = 16, t_{r_2} = 32$ となる. この時系列データの場合, 大きな振動が始まる $t_{r_1} = 9$ の近辺と大きな振動が終わる $t_{r_2} = 34$ の近辺が変化点になるのが望ましい. 文献³⁾の手法では, データの傾向が似ているかどうかの基準としてクラスタの変化量のみを考えたので, 振動した時系列データには適応できていなかった. そこで, データの傾向が似ているかどうかの基準に, 振動を追加する.

3. 振動する時系列データの傾向に基づくファジィ集合の決定

データの変化量を基に時系列データの傾向が前後で類似しているものをまとめていく方法では, 振動した時系列データに対して, 適切な結果を得ることができなかった. そこで, データの振動が前後で類似しているかも考慮する.

(1) 振動による類似性

t_i の前後の「振動の仕方」が似ているものとして, 今回は振幅が似ているもの, すなわち, 振幅(各データの値と振動の中心との差)の差が小さいものを考える. 振動の中心は, 三角移動平均を用いて求める.

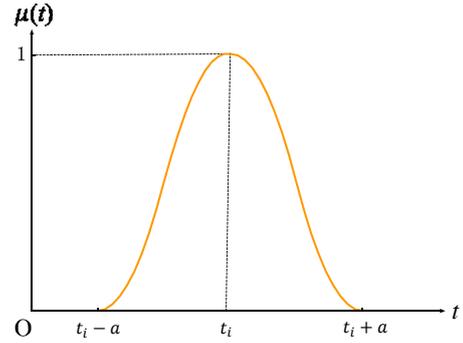


Fig. 7: 三角移動平均で用いる重み関数

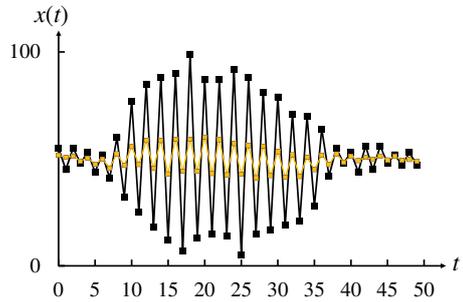


Fig. 8: Fig. 6 の時系列データと三角移動平均のデータのグラフ

三角移動平均とは, 中央の値に大きな重みを, 中央から離れるほど小さな重みを設定する重み付き平均のことで, 重みのグラフは通常は二等辺三角形のものを用いる. ここでは, 時刻 t_i で重みが 1 で時刻 t_i から a 離れている時刻 $t_i - a$ と時刻 $t_i + a$ での重みが 0 となる Fig. 7 のような π 関数(区分 2 次関数を基にしたもの)を用いる. したがって, 三角移動平均の値 $x_{\Delta}(t_i)$ は

$$x_{\Delta}(t_i) = \frac{\sum_{t_k \in [t_i - a, t_i + a]} \mu_{\pi}(t_k) \cdot x(t_k)}{\sum_{t_k \in [t_i - a, t_i + a]} \mu_{\pi}(t_k)}$$

となる. ここで, $\sum_{t_k \in [t_i - a, t_i + a]}$ は区間 $[t_i - a, t_i + a]$ の部分の時刻 t_k を変化させた時, 端付近では存在する点のみで考える. $a = 2$ の $x_{\Delta}(t_i)$ と元のデータ $x(t_i)$ のグラフは Fig. 8 のようになる. 以降の実験では全て $a = 2$ としている.

Fig. 9 の橙色の線分がクラスタ C_j の振動の中心線で, 青色は各データの振幅である. クラスタの振幅の差は, それらのデータの振幅の平均と考え, クラスタの振幅の差が小さければ, 前後のクラスタの傾向は似ていると考える. クラスタ C_j の振幅は

$$\frac{\sum |x(t_i) - x_{\Delta}(t_i)|}{m}$$

で求めることができる. ここで, m はクラスタ内のデー

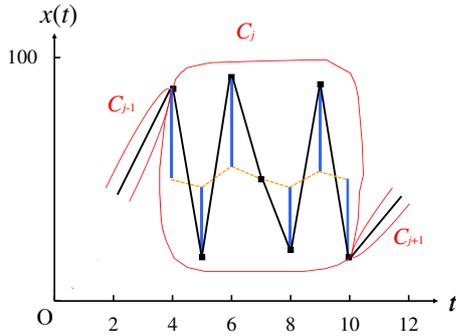


Fig. 9: クラスタの振幅

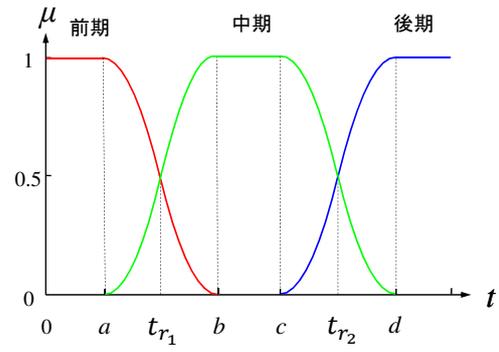


Fig. 11: 期間のファジィ集合の変更

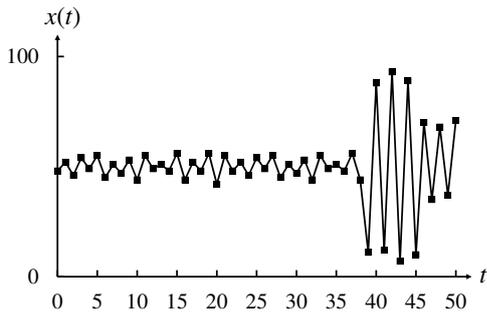


Fig. 10: 時系列データ 3

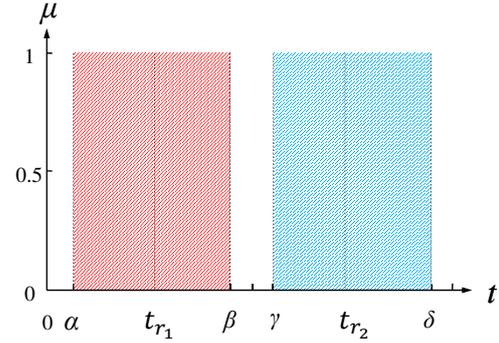


Fig. 12: 変化点 t_{r1} , t_{r2} に対する制約

タ数とする。

(2) クラスタの類似性

変化量と振動の類似性を組み合わせることにより、クラスタ同士が似ているかどうかを求める。重み w_1 と w_2 ($w_1 + w_2 = 1$) を用いて、「 $w_1 \times$ クラスタの変化量の差 + $w_2 \times$ クラスタの振幅の差」を計算し、この値が小さければ小さいほど、前後のクラスタの傾向は似ていると考える。似ているものから順にクラスタを1つにまとめていき、最終的に前期、中期、後期の3つのクラスタを作り、変化点を求める。Fig. 6の場合は、 $w_1 = 0.1$, $w_2 = 0.9$ と振動の様子を重視したパラメータ設定にすると、変化点は $t_{r1} = 9$ と、 $t_{r2} = 37$ となり、従来の手法に比べると、より人間の考え方に近い変化点が求められるようになる。言葉による表現は、全体的傾向が {1.00/ほぼ一定}, 局所の特徴が $t = 18$ で {0.565/少し増加, 0.435/中ぐらい増加} となる。

Fig. 1の時系列データの場合に対して、本手法を $w_1 = 0.1$, $w_2 = 0.9$ で適用すると、変化点が $t_{r1} = 2$, $t_{r2} = 9$ で、全体的傾向が {0.959/少し下に凸, 0.041/少し増加} となる。文献³⁾の手法では、変化点は $t_{r1} = 2$, $t_{r2} = 10$ となったが、本手法はデータの傾向(変化量と振動)が似ているものに注目して期間を決定するので、振動している部分 ($t = 9$ から $t = 12$) が1つのクラスタとな

り、振動が考慮された結果が得られる。

(3) 期間のファジィ集合の制限

期間のファジィ集合は、「前期」、「中期」、「後期」に分割すると、Fig. 11のような実数値 (a, b, c, d) の組合せで決まる。Fig. 2では、 $(a, b, c, d) = (\frac{5t_1+t_n}{6}, \frac{t_1+t_n}{2}, \frac{t_1+t_n}{2}, \frac{t_1+5t_n}{6})$ となっている ($b = c$ としている)。今回の手法では、変化点 t_{r1} と t_{r2} を基に (a, b, c, d) の値を定めることになる。

しかし、Fig. 10の時系列データに今回の手法を適用すると、 $t_{r1} = 38$, $t_{r2} = 46$ で「中期」、「後期」の期間が短く、グラフの半分より後半の期間まで「前期」になってしまう。これは不自然なので、Fig. 12のように実数値 $\alpha, \beta, \gamma, \delta$ を用いて、

$$t_1 \leq \alpha \leq t_{r1} \leq \beta \leq \gamma \leq t_{r2} \leq \delta \leq t_n$$

という制限を設ける。 α, β は t_{r1} を制限する値で、 γ, δ は t_{r2} を制限する値である。今回 Fig. 10の場合は $\alpha = 0$, $\beta = 24$, $\gamma = 26$, $\delta = 50$ とした。すると、Fig. 10の時系列データの変化点は $t_{r1} = 20$, $t_{r2} = 46$ となり、変化点の制限を設けることで不自然な期間の作成を防ぐことができる。

t_{r1}, t_{r2} が決まると (a, b, c, d) を Fig. 11のように定める。すなわち、 a を t 軸上の点 t_{r1} と t_1 の中点とし、 b を t_{r1} に関して a と対称な点とする ($t_{r1} - a = b - t_{r1}$)。そして、 d を t 軸上の点 t_{r2} と t_n の中点とし、 c を t_{r2}

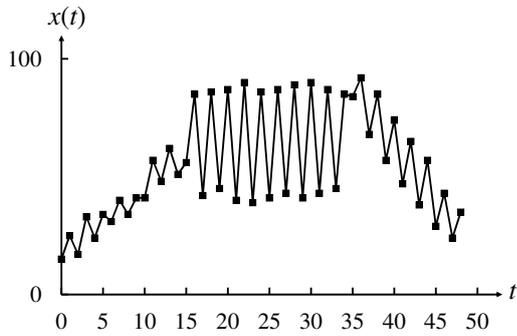


Fig. 13: 時系列データ 4

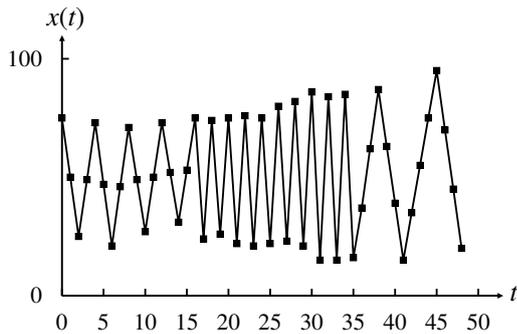


Fig. 14: 時系列データ 5

に関して d と対称な点とする $(t_{r_2} - c = d - t_{r_2})$. すると, (a, b, c, d) は次の式で定義することができる.

$$a = \frac{t_{r_1} + t_1}{2}, b = \frac{3t_{r_1} - t_1}{2}, c = \frac{3t_{r_2} - t_n}{2}, d = \frac{t_{r_2} + t_n}{2}$$

このとき, t_{r_1} と t_{r_2} の値によって $c < b$ となる場合は, (a, b, c, d) を再調整する. b は区間 $t_{r_1} t_{r_2}$ を $t_{r_1} - t_1 : t_n - t_{r_2}$ に内分する点とする ($b = c$). a は t_{r_1} に関して b と対称な点とし ($t_{r_1} - a = b - t_{r_1}$), d は t_{r_2} に関して c と対称な点とする ($t_{r_2} - c = d - t_{r_2}$). すなわち, (a, b, c, d) は次のような式で定義することができる.

$$a = 2t_{r_1} - b, b = \frac{t_{r_1}(t_n - t_{r_2}) + t_{r_2}(t_{r_1} - t_1)}{(t_{r_1} - t_1) + (t_n - t_{r_2})},$$

$$c = b, d = 2t_{r_2} - b$$

(4) 適用例

時系列データに応じて, 本手法の重み w_1 と w_2 をどのような配分にすれば良いかを検証する.

Fig. 13 の時系列データに対して, 本手法を適用すると $w_1 = 0.5, w_2 = 0.5$ のとき, 変化点は $t_{r_1} = 22$ と $t_{r_2} = 26$ で, データの傾向に合わない変化点となるが, $w_1 = 0.1, w_2 = 0.9$ のときは, 変化点は $t_{r_1} = 16$ と $t_{r_2} = 34$ で, データの傾向に合った変化点を求めることができる. この時系列データは振動しているので, クラスタの変化量の差の重み w_1 よりクラスタの振幅の差の重み w_2 を大きく設定することで, 振動の傾向を

重視した変化点になり, より妥当な期間を決定することができる.

Fig. 14 の時系列データの場合に対して, 本手法を適用すると, $w_1 = 0, w_2 = 1$ のとき, 変化点は $t_{r_1} = 1$ と $t_{r_2} = 35$ で, データの傾向に合わない変化点となるが, $w_1 = 0.05, w_2 = 0.95$ のとき, 変化点は $t_{r_1} = 16$ と $t_{r_2} = 35$ でデータの傾向に合った変化点を求めることができる. この結果から, クラスタが似ているかどうかの基準に振動のみを考慮した方法ではうまく機能しないので, 変化量と振動の両方の傾向に基づいて, 似ているクラスタを 1 つにまとめる必要がある.

このように適切な変化点を求めるためには, データに応じて適切な重みを設定する必要がある. 今回はユーザーが自由に定義できるようにしているが, 今後はデータの振動やデータ数等を解析してその状況に応じた適切な重みを考えていきたい.

4. おわりに

本論文では, 期間のファジィ集合の分割を時系列データの傾向に基づいて決定する方法について述べた. これにより期間の分割がよりデータの傾向に近いものとなった. 今後は, クラスタの類似値を求める際に用いる重みをデータに応じた適切な重みが自動的に決まるシステムを開発していきたい.

参考文献

- 1) 馬野 元秀, 小泉 尚之, 篠原 貴之, 瀬田 和久: 「全体的傾向と局所的特徴に基づく時系列データの言葉による表現」, 第 22 回ファジィ システム シンポジウム, pp.343-346, 2006.
- 2) 馬野 元秀, 高橋 勝稔, 瀬田 和久: 「全体的傾向と局所的特徴に基づく時系列データの言葉による表現-期間のファジィ集合の決定-」, 第 25 回ファジィ システム シンポジウム, pp.76-80, 2009.
- 3) 馬野 元秀, 岩浅 貴大, 高橋 勝稔: 「全体的傾向と局所的特徴に基づく時系列データの言葉による表現-データの傾向に基づく期間のファジィ集合の決定-」, 第 30 回ファジィ システム シンポジウム, pp.808-813, 2014.

[問い合わせ先]

〒599-8531 堺市中区学園町 1-1
大阪府立大学 大学院理学系研究科
情報数理科学専攻
岩浅 貴大
TEL: 072-254-9675
FAX: 072-254-9930
E-mail: iwaasa@marron.cias.osakafu-u.ac.jp

全体的傾向と局所的特徴に基づく時系列データの言葉による表現 —振動を取り出すための三角移動平均のパラメータの調整—

○島野佳之 高橋勝稔 馬野元秀 (大阪府立大学)

Expression of Time Series in Natural Language with Global Trend and Local Features

— Adjustment of Parameter of Triangular Moving Average for taking Oscillation

*Y. Shimano and K. Takahashi and M. Umamo (Osaka Prefecture University)

Abstract— We have many kinds of data of time series such as stock prices. We understand time series with a natural language. We have proposed a method to express a global trend and local features and oscillation of time series in a natural language. For large oscillations, we have proposed a method to express global trends and local features of local maximums and local minimums of differences between the data and their Triangular Moving Average (TMA). In order to express oscillation adequately, we need to adjust the parameters of TMA for the difference between the data and TMA to be symmetric with respect to t axis, we present expressing its outline more adequately by simulation.

Key Words: Time Series, Global Trend, Oscillation

1. はじめに

時系列データとは時間順に並んだデータのことです。我々人間はこのような時系列データを言葉で理解していると思われる。これまでに、時系列データの全体的傾向と局所的特徴を言葉で表現する方法を提案し¹⁾、さらに、標準偏差と増減回数を用いて振動を言葉で表現する方法²⁾を提案した。また、振動が多い場合の表現方法として、時系列データと三角移動平均との差のデータについて極大値と極小値の全体的傾向を求めることにより、振動している部分の概形を表現する方法を提案した⁴⁾。

振動している部分の概形を適切に表現するには、時系列データごとに三角移動平均のパラメータの調整が必要である。本研究では、そのパラメータを極大値と極小値の時系列データが三角移動平均に対して対称になるように調整することによって、より妥当な概形の表現が得られることをシミュレーションによって示す。

2. 時系列データの言葉による表現

まず、時系列データの全体的傾向と局所的特徴と振動の言葉による表現の方法について述べる¹⁾²⁾³⁾。

(1) 全体的傾向

時刻 t_1 から t_n までの時系列データ $x(t_i)$ を考える。ここでは時系列のグラフを考えているので、グラフの縦軸 ($x(t_i)$ 軸) の範囲が分かっているとします。具体的に縦軸の最小値を x_{min} 、最大値を x_{max} とする。

時系列データの全体的傾向を言葉で表現するには以下のような手順で行う¹⁾。まず、Fig. 1 のような前期、

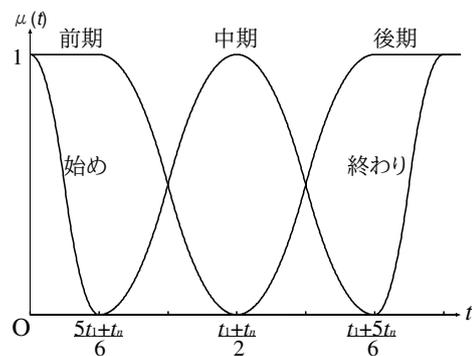


Fig. 1: 期間のファジィ集合

中期、後期の期間のファジィ集合を用いて時系列データをファジィ的に分割する。

次に各期間における代表値を求める。ここでは期間のファジィ集合による重み付き平均値を用い、

$$m_{\text{期間}} = \frac{\sum_{i=1}^n x(t_i) \mu_{\text{期間}}(t_i)}{\sum_{i=1}^n \mu_{\text{期間}}(t_i)} \quad (1)$$

で計算する。前期から中期、中期から後期への変化量を、 $d_{\text{前期} \rightarrow \text{中期}} = m_{\text{中期}} - m_{\text{前期}}$ 、 $d_{\text{中期} \rightarrow \text{後期}} = m_{\text{後期}} - m_{\text{中期}}$ で計算し、Fig. 2 のような変化量のファジィ集合を用いて、各変化量に対応する表現と度合いを求める。次に、Table 1 を用いて全体的傾向の表現を求める。Table 1 は縦に前期から中期への変化量のファジィ集合を、横に中期から後期への変化量のファジィ集合をとり、表の対応する要素に全体的傾向を表す表現を入れてある。変化量のファジィ集合の全ての組み合わせについて全体的傾向を求め、各変化量の一致度の積をその全体的傾向の一致度とする。同じ表現が得られた場合はそれら

Table 1: 全体的傾向を求める表

	大きく減少	中ぐらい減少	少し減少	ほぼ一定	少し増加	中ぐらい増加	大きく増加
大きく減少	大きく減少	大きく減少	中ぐらい減少	中ぐらい減少	少し下に凸	下に凸	下に凸
中ぐらい減少	大きく減少	大きく減少	中ぐらい減少	少し減少	少し下に凸	下に凸	下に凸
少し減少	中ぐらい減少	中ぐらい減少	少し減少	ほぼ一定	少し下に凸	少し下に凸	少し下に凸
ほぼ一定	中ぐらい減少	少し減少	ほぼ一定	ほぼ一定	ほぼ一定	増加	中ぐらい増加
少し増加	少し上に凸	少し上に凸	少し上に凸	ほぼ一定	少し増加	中ぐらい増加	中ぐらい増加
中ぐらい増加	上に凸	上に凸	少し上に凸	少し増加	中ぐらい増加	大きく増加	大きく増加
大きく増加	上に凸	上に凸	少し上に凸	中ぐらい増加	中ぐらい増加	大きく増加	大きく増加

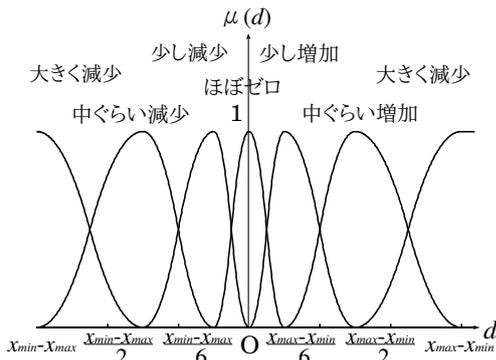


Fig. 2: 変化量のファジィ集合

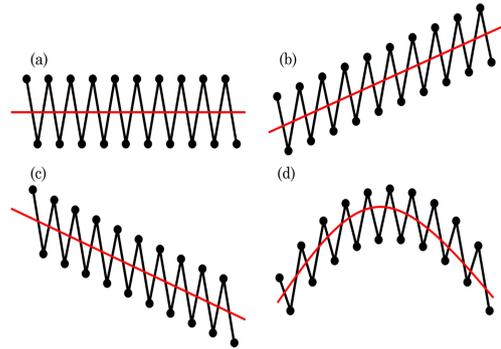


Fig. 3: 同じように見える振動

の一致度の和をとる.

(2) 局所の特徴

時系列データの局所の特徴は、時系列データと全体的傾向を比べ、差が顕著な部分であると考えられるので、まず全体的傾向を表す時系列データを作る。グラフの両端を強調するために「始め」と「終わり」のファジィ集合を定義し³⁾、 $m_{\text{始め}}$ と $m_{\text{終わり}}$ を求め、「始め」、「前期」、「中期」、「後期」、「終わり」の5つの期間を用いて、

$$x'(t_i) = \frac{\sum_{\text{期間}} \mu_{\text{期間}}(t_i) \cdot m_{\text{期間}}}{\sum_{\text{期間}} \mu_{\text{期間}}(t_i)} \quad (2)$$

で計算する。

そして、元の時系列データ $x(t_i)$ と全体的傾向の時系列データ $x'(t_i)$ とを比べて差の絶対値 $|x(t_i) - x'(t_i)|$ が大きい部分を見つけ、差 $x(t_i) - x'(t_i)$ をファジィ集合で表現する¹⁾。

(3) 振動の表現

振動している時系列データを、標準偏差と増減回数をを用いて言葉と度合いで表現する²⁾。

標準偏差により振幅の違いは区別できると思われるが、Fig. 3の各データは振動の様子だけを考えるとすべて同じように見える。したがって、振動の様子は全体的傾向からずれている部分だけを考える必要がある。

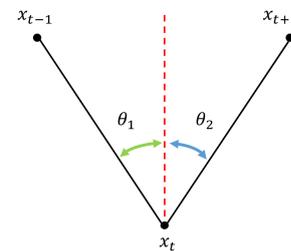


Fig. 4: 注目する角度

このことから、元の時系列データと全体的傾向を表す時系列データとの差の標準偏差 σ を用いる。

増減回数は、Fig. 4のように垂直線で2つに分けた角 θ_1 と θ_2 を用いて次のように定義されている。

$$v = \left\{ 2 - \frac{1}{90}(\theta_1 + \theta_2) \right\} \left\{ 1 - \frac{1}{180}|\theta_1 - \theta_2| \right\} \quad (3)$$

そして、標準偏差と増減回数をファジィ集合を用いて言葉と度合いで表現し、振動の表現を求める表²⁾を用いて振動の言葉による表現を得る。

3. 振動が多い時系列データの言葉による表現

我々は、Fig. 5のような振動の多いグラフに対して、振動している部分の上側の形(極大値の一部からなるグラフ)と下側の形(極小値の一部からなるグラフ)を見ることで、どのような振動であるかを表現できると考えている。そこで、振動が多い場合の表現方法として、

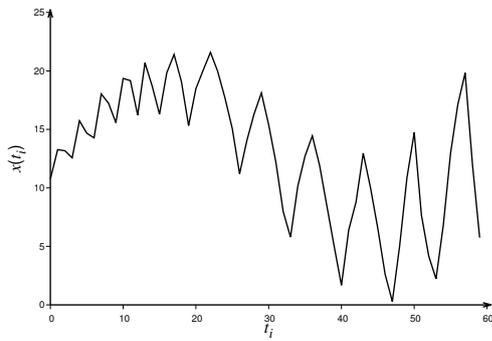


Fig. 5: 振動が大きい時系列データ (その 1)

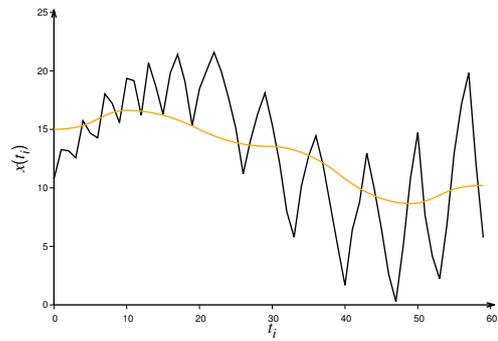


Fig. 6: Fig. 5 の時系列データとその全体的傾向のグラフ

時系列データの極大値と極小値の全体的傾向と局所の特徴を合わせて考える⁴⁾。

(1) 極大値と極小値

時系列データ $x(t_i)$ の極大値は増加から減少に変わる点, すなわち

$$x(t_{i-1}) < x(t_i) \text{ かつ } x(t_i) > x(t_{i+1}) \quad (4)$$

を満たす $x(t_i)$ である。また, 極小値は減少から増加に変わる点, すなわち

$$x(t_{i-1}) > x(t_i) \text{ かつ } x(t_i) < x(t_{i+1}) \quad (5)$$

を満たす $x(t_i)$ である。

(2) 振動の中心

2 節の振動の表現方法と同様に振動の様子のみを考えるために, 式 (2) による全体的傾向を表すデータ $x'(t_i)$ を考え, それとの差 $x_d(t_i) = x(t_i) - x'(t_i)$ を考える。

Fig. 5 の時系列データ $x(t_i)$ とその全体的傾向を表すデータ $x'(t_i)$ を付け加えたグラフが Fig. 6 である。そして, 全体的傾向との差 $x_d(t_i)$ のグラフは Fig. 7 のようになる。元の時系列データのグラフは縦の範囲を $[0, 25]$ とし, 差のデータのグラフは $[-15, 15]$ としていることに注意されたい。全体的傾向はその期間全体の平均値に近い値をとるために, 1 つの期間の中でデータが大きく変化している場合には元の時系列データの振動の中心から大きくはずれてしまう。Fig. 6 はその例で, 時系列データから正確に振動だけを取り出して, その振動を表現することができていない。もう少し狭い範囲の平均値との差を考える必要がある。そこで, 三角移動平均との差を考える。

三角移動平均とは, 中央の値に大きな重みを, 中央から離れるほど小さな重みを設定する重み付き平均のことで, 重みのグラフは通常は二等辺三角形のものをを用いる。ここでは, 時刻 t_i で重みが 1 で時刻 $t_i - a$ と時刻 $t_i + a$ での重みが 0 となる Fig. 8 のような π 関数 (区分 2 次関数を基にしたもの) をを用いる。 a は幅に対

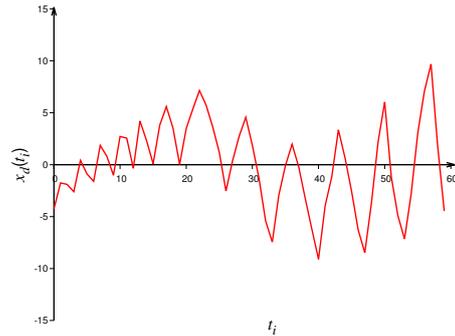


Fig. 7: Fig. 5 とその全体的傾向との差のグラフ

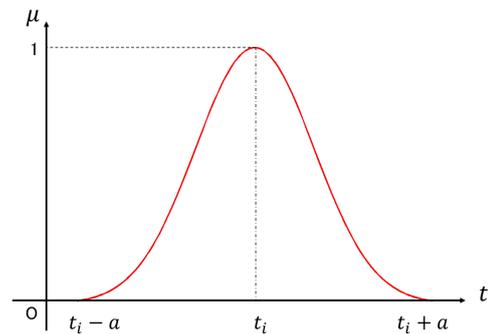


Fig. 8: 三角移動平均で用いる重み関数

応するパラメータである, したがって三角移動平均の値 $x_{\Delta}(t_i)$ は,

$$x_{\Delta}(t_i) = \frac{\sum_{t_k \in [t_i - a, t_i + a]} \mu_{\pi}(t_k) \cdot x(t_k)}{\sum_{t_k \in [t_i - a, t_i + a]} \mu_{\pi}(t_k)} \quad (6)$$

となる。ここで, $\sum_{t_k \in [t_i - a, t_i + a]}$ は区間 $[t_i - a, t_i + a]$ の部分の時刻 t_k を変化させたときの和とする。端付近では存在する点のみで考えるので, t_k は厳密には区間 $[\max(t_i - a, t_1), \min(t_i + a, t_n)]$ を変化させることになる。最後に, この三角移動平均 $x_{\Delta}(t_i)$ を使って $x_c(t_i) = x(t_i) - x_{\Delta}(t_i)$ を求める。文献⁴⁾では幅 a をデータ数に応じた一定の値にしていた。しかし, ここでは幅 a の適切な値を決める方法を提案する。

(3) 三角移動平均の幅の調整

a の値を変化させると, $x_c(t_i)$ が変化する。例えば, $a = 1.5$ のときは Fig. 9 のようになり, $a = 60$ のとき

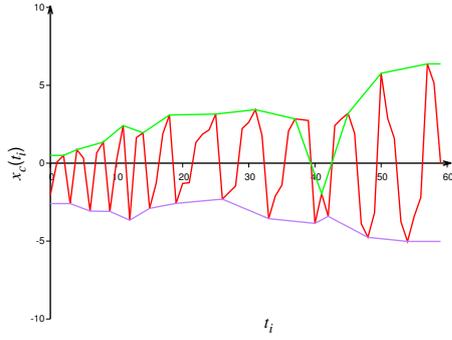


Fig. 9: Fig. 5 の $a = 1.5$ のときの x_c

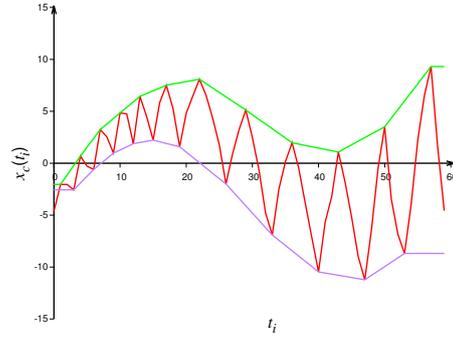


Fig. 10: Fig. 5 の $a = 60$ のときの x_c

は Fig. 10 のようになる．緑色の線が極大値をなぞった上側の線で，紫色の線が極小値をなぞった下側の線である．幅 a の適切な値は，三角移動平均との差の時系列データ $x_c(t_i)$ の極大値の点をなぞった上側の線と極小値の点をなぞった下側の線が全体として t 軸で対称になる場合があるので，上側の線のデータ $\overline{x}_c(t_i)$ と下側の線のデータ $\underline{x}_c(t_i)$ が全体として t 軸で対称となるような a の値を求めたい． $\overline{x}_c(t_i)$ と $\underline{x}_c(t_i)$ が t 軸で対称であれば， $\overline{x}_c(t_i) + \underline{x}_c(t_i) = 0$ となるので，対称でない程度は $|\overline{x}_c(t_i) + \underline{x}_c(t_i)|$ で表現できる．したがって，全体の対称でない程度は各 t_i の対称でない程度を平均として，

$$E = \frac{1}{n} \sum_{t_i \in [t_1, t_n]} |\overline{x}_c(t_i) + \underline{x}_c(t_i)| \quad (7)$$

で計算できる．このとき， $\overline{x}_c(t_i)$ については，極大値となる点は $x_c(t_i)$ に存在するが，それ以外は $x_c(t_i)$ に存在しないので，前後の極大値を使って補間する（ここでは線形で補間する）．したがって，上側のデータ $\overline{x}_c(t_i)$ は

$$\overline{x}_c(t_i) = \begin{cases} x_c(t_i) & x_c(t_i) \text{ が極大値のとき} \\ \frac{x_c(t_R) - x_c(t_L)}{t_R - t_L} (t_i - t_L) + x_c(t_L) & x_c(t_i) \text{ が極大値ではないとき} \end{cases} \quad (8)$$

となる．ここで， $x(t_L)$ は $x(t_i)$ の左側 ($t_L < t_i$) の最も近い極大値であり， $x(t_R)$ は $x(t_i)$ の右側 ($t_R > t_i$) の最も近い極大値である．下側のデータ $\underline{x}_c(t_i)$ も同様に考えて，

$$\underline{x}_c(t_i) = \begin{cases} x_c(t_i) & x_c(t_i) \text{ が極小値のとき} \\ \frac{x_c(t_R) - x_c(t_L)}{t_R - t_L} (t_i - t_L) + x_c(t_L) & x_c(t_i) \text{ が極小値ではないとき} \end{cases} \quad (9)$$

となる．ここで， $x(t_L)$ は $x(t_i)$ の左側 ($t_L < t_i$) の最も近い極小値であり， $x(t_R)$ は $x(t_i)$ の右側 ($t_R > t_i$) の最も近い極小値である．そして， E が最小となる a を見つけばよいので，まず， a を 1 からデータ数まで 0.1 ずつ変化させる．そのときの誤差 E のグラフが Fig. 11 であ

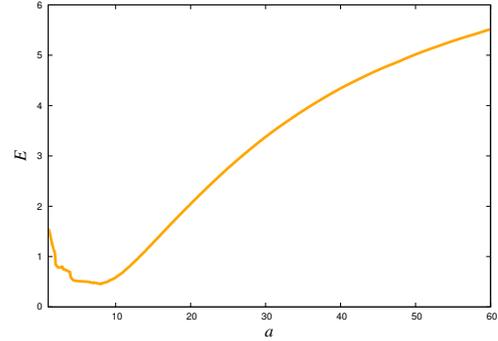


Fig. 11: Fig. 5 の $1.0 \leq a \leq 60.0$ のときの誤差

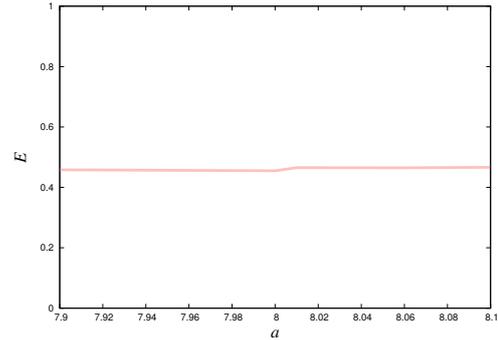


Fig. 12: Fig. 5 の $7.90 \leq a \leq 8.10$ のときの誤差

る．これを見ると $a = 8.0$ のとき誤差が最小になっている．そこで，この近くの区間 $[7.9, 8.1]$ の範囲で a を 0.01 ずつ変化させる．このときの誤差のグラフが Fig. 12 である．これを見ると $a = 8.00$ のとき誤差が最小になっている．したがって，Fig. 5 の時系列データに対して $a = 8.00$ のときに $\overline{x}_c(t_i)$ と $\underline{x}_c(t_i)$ が最も t 軸対称であるといえる．

このときの三角移動平均と元の時系列データのグラフは Fig. 13 のようになる． $a = 8.00$ のときの三角移動平均 $x_\Delta(t_i)$ は振動の中心にあることがわかる．

この $x_\Delta(t_i)$ を使って $x_c(t_i) = x(t_i) - x_\Delta(t_i)$ を求めると，Fig. 14 の赤色のグラフのようになる．そして， $x_c(t)$ の極大値と極小値の全体的傾向と局所の特徴を求めると，極大値の全体的傾向は {0.905/少し増加, 0.064/ほぼ一定, 0.032/中ぐらい増加} で，Fig. 14 の緑色のグラフが極大

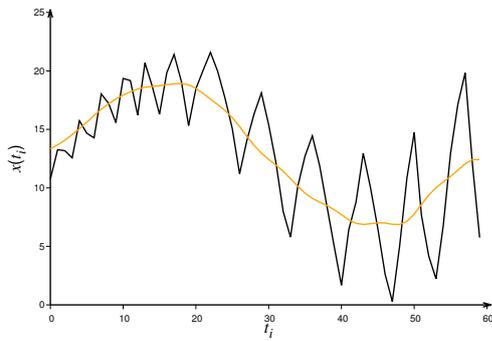


Fig. 13: Fig. 5 と $a = 8.00$ のときの三角移動平均

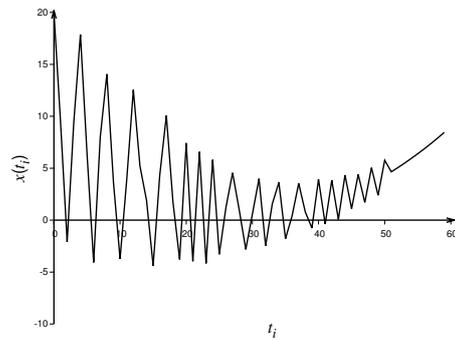


Fig. 15: 振動が大きい時系列データ (その 2)

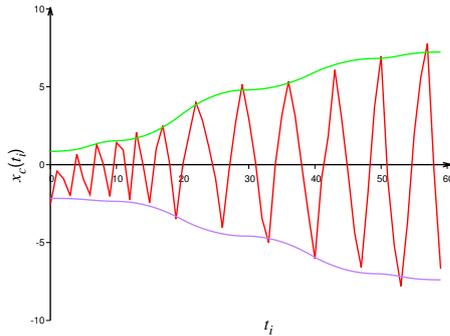


Fig. 14: Fig. 5 の差のデータとその極値の全体的傾向

値の全体的傾向のグラフとなり、局所の特徴は時刻 1 で {0.886/ほぼゼロ, 0.114/少し減少} となる。また、極小値の全体的傾向は {0.925/少し減少, 0.075/ほぼ一定} で、Fig. 14 の紫色のグラフが極小値の全体的傾向のグラフとなり、局所の特徴は時刻 53 で {0.973/ほぼゼロ, 0.027/少し減少} となる。

(3) 振動の表現

極大値と極小値の全体的傾向を用いて振動の表現を求める。そのための表が Table 2 で、縦が極小値で横が極大値の全体的傾向の言葉の表現である。極値の言葉の表現のすべての組み合わせについて、各言葉の表現の一致度の積を振動の言葉による表現の一致度とする。またこの表には — の部分があるが、これは全体的傾向が「一定」になるようにしたデータについて考えているので通常は選ばれないと考えている。小さい度合いで選ばれることがあるが、そのときは無視しているので、振動の一致度の和が 1 にならないことがある。Fig. 5 の時系列データの振動の表現は {0.866/振動の大きさが増加, 0.005/振動の大きさが一定} となる。

(4) 他の振動が多い時系列データ

別の振動が多い時系列データ Fig. 15 について同様に適切な a の値とそれを使って振動の表現を求めてみる。誤差は a を 0.1 きざみで変化させた場合 Fig. 16 のようになり、 $a = 3.9$ のときに誤差は最小になる。 a の値を [3.8, 4.0] の範囲で 0.01 きざみで変化させた場合 $a = 3.99$

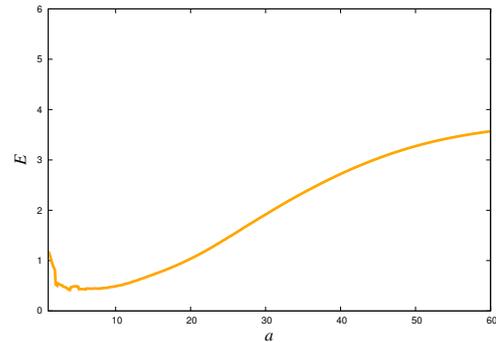


Fig. 16: Fig. 15 の $1.0 \leq a \leq 60.0$ のときの誤差

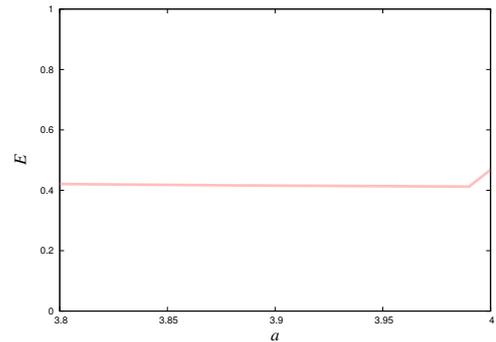


Fig. 17: Fig. 15 の $3.80 \leq a \leq 4.00$ のときの誤差

のときに誤差は最小になる。したがって、Fig. 15 の時系列データに対する a の値を 3.99 とする。このとき三角移動平均と元の時系列データが Fig. 18 のようになり、このデータに対しても、三角移動平均 $x_{\Delta}(t_i)$ は振動の中心にあることがわかる。このときの $x_{\Delta}(t_i)$ を用いて、差のデータ $x_c(t_i)$ を求めてその極値を考えると Fig. 19 となり、極大値の全体的傾向は {0.753/少し減少, 0.193/ほぼ一定, 0.054/中ぐらい減少} で、局所の特徴は時刻 4 で {0.920/ほぼゼロ, 0.080/少し増加} となる。極小値の全体的傾向は {0.649/少し増加, 0.315/ほぼ一定, 0.036/中ぐらい増加} で、局所の特徴は時刻 2 で {0.932/ほぼゼロ, 0.068/少し減少} となるので、振動の表現は {0.553/振動の大きさが減少, 0.061/振動の大きさが一定} となる。

Table 2: 振動の表現を求める表

	増加	ほぼ一定	減少	上に凸	下に凸
増加	—	—	振動の大きさが一定	—	—
ほぼ一定	—	振動の大きさが一定	—	—	—
減少	振動の大きさが増加	—	—	—	—
上に凸	—	—	—	—	真ん中で両側にくぼんでいる
下に凸	—	—	—	真ん中で両側に膨らんでいる	—

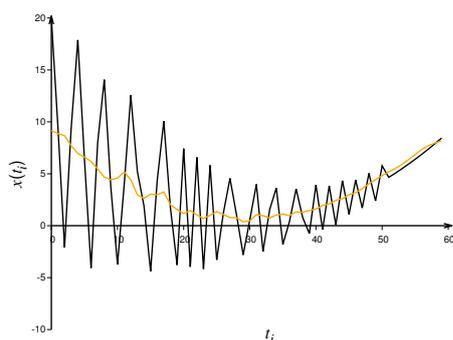


Fig. 18: Fig. 15 と $a = 3.99$ のときの三角移動平均

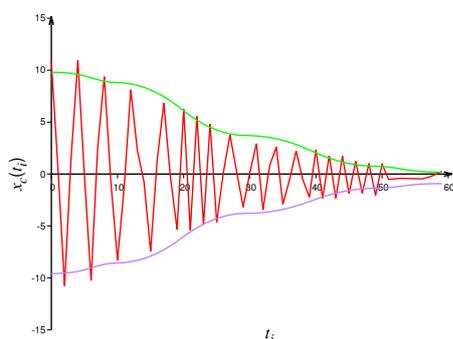


Fig. 19: Fig. 5 の差のデータとその極値の全体的傾向

4. おわりに

本論文では、振動回数が多い時系列データに対して振動の表現を三角移動平均値による中心線との差の極大値と極小値の全体的傾向と局所の特徴を用いて振動を表現する方法に対して、三角移動平均の重み関数の幅 a の適切な値を求める方法を提案した。今後の課題として、振動の表現を求めするための表の検討が考えられる。また、三角移動平均による中心線が振動しているとき、その中心線に対して本論文と同様のことを行えば、全体的な振動も表現できる。この振動の表現とその振動上にある振動の表現を組み合わせることさらに詳細な振動の表現をすることも可能になると思われる。このことについても今後検討していきたい。さ

らに、極大値と極小値の局所の特徴に関する処理の方法などについても考えていきたい。

参考文献

- 1) 馬野 元秀, 小泉 尚之, 篠原 貴之, 瀬田 和久:「全体的傾向と局所の特徴に基づく時系列データの言葉による表現」, 第 22 回ファジィシステムシンポジウム, pp.343-346, 2004.
- 2) 馬野 元秀, 岡村 光洋, 瀬田 和久:「全体的傾向と局所の特徴に基づく時系列データの言葉による表現—増減回数の拡張による振動の表現—」, 第 24 回ファジィシステムシンポジウム, pp.774-779, 2008.
- 3) 馬野 元秀, 高橋 勝稔, 瀬田 和久:「全体的傾向と局所の特徴に基づく時系列データの言葉による表現—期間のファジィ集合の決定—」, 第 25 回ファジィシステムシンポジウム, ページなし(講演番号 1F-204), 2009.
- 4) 馬野 元秀, 島野 佳之, 高橋 勝稔:「全体的傾向と局所の特徴に基づく時系列データの言葉による表現—振動回数が多い場合の振動表現—」, 第 30 回ファジィシステムシンポジウム, pp.814-819, 2014.

[問い合わせ先]

〒 599-8531

大阪府堺市中区学園町 1 - 1

大阪府立大学 大学院理学系研究科

情報数理科学専攻

島野 佳之

TEL: 072-254-9675

FAX: 072-254-9163

E-mail: shimano@marron.cias.osakafu-u.ac.jp

混合多項分布に基づくファジィ共クラスタリングにおける項目の排他的分割

中野 剛也 本多 克宏 野津 亮 (大阪府立大学)

Exclusive Item Partition in Fuzzy Co-clustering Based on Multinomial Mixture Concept

*T. Nakano, K. Honda and A. Notsu (Osaka Prefecture University)

Abstract— A fuzzy co-clustering model was constructed based on multinomial mixture concept and have been proved to be useful for such object-item cooccurrence information analysis as document-keyword analysis and purchase history analysis. However, the conventional fuzzy co-clustering model adopted exclusive partition penalty only for object memberships and could not consider exclusive item partition. In this paper, a novel clustering model is proposed, in which an additional exclusive partition penalty is utilized for item memberships, and the characteristic feature of item exclusive partition is discussed through several numerical experiments.

Key Words: Fuzzy clustering, Multinomial mixture models, Exclusive partition

1 はじめに

混合多項分布¹⁾の概念に基づくファジィ共クラスタリング²⁾は、文書解析や購買履歴分析などの個体-項目の共起情報データ解析において有効な手法であるが、個体に関しては排他的な分割となるものの項目については分割の排他性がない。

本研究では、項目の分割における排他性を強いるためのペナルティ項の有効性について議論する。クラスターを特徴づける項目を解釈する上で、各項目が唯一のクラスターに帰属することが望ましい場面もある。本研究では、新たな排他的項目分割の機構を提案し、数値実験を通して有効性を検証する。

2 FCM型ファジィ共クラスタリングと分割の制約条件

2.1 FCM型ファジィ共クラスタリング

n 個の個体と m 種類の項目の関連性を表す $n \times m$ 共起関係データ行列 $R = \{r_{ij}\}$ が与えられたときに、個体・項目の対からなる C 個の共クラスターに分割することを考える。 u_{ci} および w_{cj} を個体 i と項目 j のクラスター c へのファジィメンバシップとし、 i と j の共起度 r_{ij} が大きいときに u_{ci} と w_{cj} がともに大きくなる分割を求める。

FCM型の共クラスタリングモデルである FCCM法³⁾では、凝集度をクラスタリング基準とし、 u_{ci} と w_{cj} をFCM型アルゴリズムで同時推定する。

$$L_{fccm} = \sum_{c=1}^C \sum_{i=1}^n \sum_{j=1}^m u_{ci} w_{cj} r_{ij} - \lambda_u \sum_{c=1}^C \sum_{i=1}^n u_{ci} \log u_{ci} - \lambda_w \sum_{c=1}^C \sum_{j=1}^m w_{cj} \log w_{cj} \quad (1)$$

メンバシップのファジィ化のためには、エントロピー項による正則化の概念に基づくアプローチ⁴⁾が導入さ

れている。 λ_u および λ_w がメンバシップのファジィ度を調整する重みであり、大きいほどファジィ度が増す。繰り返しアルゴリズムでは、収束するまで u_{ci} と w_{cj} が交互最適化の枠組みで更新される。

個体のメンバシップ u_{ci} については、FCM法⁵⁾と同様に C 個のクラスターについての和が1となる排他的な制約が付加されており、個体のクラスターへの割り当てを表す。一方、項目のメンバシップ w_{cj} については、自明な解を避けるために、各クラスターにおいて m 個の項目についての和が1となる制約が用いられる。そのため、クラスター内での相対的な重要度を表す重みとなり、各項目は複数のクラスターに帰属したり、いずれにも属さなかったりすることがある。つまり、項目の割り当てとしての解釈には不適であった。

2.2 項目の排他的割り当てのためのペナルティ

項目がなるべく単一のクラスターに割り当てられるようにする改良として、逐次的なクラスター抽出⁶⁾におけるペナルティ項の概念を導入した以下の目的関数が提案された⁷⁾。

$$L_{fccm'} = \sum_{c=1}^C \sum_{i=1}^n \sum_{j=1}^m u_{ci} w_{cj} r_{ij} - \beta \sum_{c=1}^C \sum_{j=1}^m \sum_{t \neq c} w_{cj} \cdot w_{tj}^* - \lambda_u \sum_{c=1}^C \sum_{i=1}^n u_{ci} \log u_{ci} - \lambda_w \sum_{c=1}^C \sum_{j=1}^m w_{cj} \log w_{cj} \quad (2)$$

第2項が排他性を課すためのペナルティ項であり、重み β が排他性の度合いを調整する。クラスター c でのメンバシップ w_{cj} を算出する際、他のクラスターで帰属度が大きい場合には値が小さくなるようなペナルティとして、 $w_{cj} \cdot w_{tj}^*$ ($t \neq c$) が用いられている。他のクラスターに帰属し w_{tj}^* ($t \neq c$) が大きな値を持つ場合

は、 w_{cj} が小さくなる．ここで、 w_{tj}^* は FCM 型の繰り返しアルゴリズムにおいて、計算の簡略化のために前回更新時の w_{tj} の値を暫時固定して用いることを示している．

上記の目的関数から、2 種類のメンバシップの更新則は、おのこの以下のように求められる． u_{ci} については、従来の FCCM 法と同様の更新則となる．

$$u_{ci} = \frac{\exp\left(\lambda_u^{-1} \sum_{j=1}^m w_{cj} r_{ij}\right)}{\sum_{\ell=1}^C \exp\left(\lambda_u^{-1} \sum_{j=1}^m w_{\ell j} r_{ij}\right)} \quad (3)$$

w_{cj} については、ペナルティ項を考慮した更新則となる．

$$w_{cj} = \frac{\exp\left(\lambda_w^{-1} \left(\sum_{i=1}^n u_{ci} r_{ij} - \beta \sum_{t \neq c} w_{tj}^*\right)\right)}{\sum_{\ell=1}^m \exp\left(\lambda_w^{-1} \left(\sum_{i=1}^n u_{ci} r_{i\ell} - \beta \sum_{t \neq c} w_{t\ell}^*\right)\right)} \quad (4)$$

更新アルゴリズムでは、上記の二つの更新則を収束するまで繰り返し適用する．その際、新たに付加した排他性のペナルティ項の不適切な影響を避けるために、重み β を初期値では 0 とおき、徐々に増加させながら排他性を満たしていくものとする．これは、従来の FCCM 法で得られた共クラスター構造の明確化を目的として、項目のメンバシップの排他性を導入する意図に基づいている．

3 混合多項分布に基づくファジィ共クラスタリングと分割の制約条件

3.1 混合多項分布に基づくファジィ共クラスタリング

エントロピー正則化による FCM 法⁴⁾には、ガウス混合モデルとの密接な関係があり、ファジィ度の重みが要素分布の分散パラメータと対応することから、確率モデルとの比較でファジィ度を決定できる利点がある．一方、FCCM 法には対応する確率モデルが無いために、ファジィ度の調整における指標が無い．

FCCM 法の改良として、確率モデルに基づく共クラスタリング法である混合多項分布の概念に基づくファジィ共クラスタリング法²⁾が提案された．混合多項分布における疑似的な尤度関数が、クリスプな凝集度基準に K-L 情報量を付加することでソフト分割を実現していると解釈できることから、K-L 情報量による正則化を考慮した以下の目的関数が提案された．

$$L_{klfcm} = \sum_{c=1}^C \sum_{i=1}^n \sum_{j=1}^m u_{ci} r_{ij} \log w_{cj} + \lambda_u \sum_{c=1}^C \sum_{i=1}^n u_{ci} \log \frac{\pi_c}{u_{ci}} \quad (5)$$

π_c がクラスター c の容量を表し、K-L 情報量のペナルティ項により u_{ci} が π_c に近づく効果が得られるので、クラスターごとの容量を考慮した u_{ci} のファジィ化が

実現されている． λ_u は u_{ci} のファジィ化の重みであり、 $\lambda_u = 1$ のとき、(5) 式は混合多項分布における疑似的な尤度関数と等しい． $\lambda_u < 1$ のときには混合多項分布に比してクリスプに近い分割が、 $\lambda_u > 1$ のときにはよりファジィ度の強い分割が、それぞれ得られる．したがって、混合多項分布との比較においてファジィ度を調整することができる．

FCCM 法との対比においては、クラスター容量 π_c の追加のほかに、項目メンバシップ w_{cj} のファジィ化法が変更されている．FCCM 法では、線形関数となる凝集度にエントロピー項を付加することで非線形な目的関数が得られているが、K-L 情報量による正則化を用いる KLFCCM 法では、凝集度の算出部に非線形な $\log w_{cj}$ を導入することで、ペナルティ項の付加なしにファジィ化を行っている．ただし、項目メンバシップについてはファジィ度の調整機能が無く、混合多項分布と同じファジィ度に固定されている．

3.2 項目の排他的割り当てのためのペナルティ

FCCM 法と同様に、KLFCCM 法においても項目メンバシップはクラスター内での相対的な重要度の指標となっており、項目分割の明確化の観点からは、排他性の導入が好ましい場面もある．そこで本研究では、(5) 式の目的関数に項目メンバシップの排他性を導入することを考える．

FCCM 法において項目分割の排他性を導入するには、改良型の凝集度基準 $\sum_{c=1}^C \sum_{j=1}^m (\sum_{i=1}^n u_{ci} r_{ij} - \beta \sum_{t \neq c} w_{tj}^*) w_{cj}$ の中で、凝集性 $\sum_{i=1}^n u_{ci} r_{ij}$ と排他性 $\beta \sum_{t \neq c} w_{tj}^*$ のトレードオフを調整しながら、個体と項目のメンバシップが推定されていた．ここで、KLFCCM 法においても同様の概念を導入しようとする、 β が大きいときには改良型の凝集度基準が負の値を取りうる．FCCM 法の (4) 式の更新則では、 $\exp(\cdot)$ 関数を用いるために負の値を処理できるが、KLFCCM 法では負のクラスタリング基準を処理できない問題が生じる．

そこで本研究では、 $\sum_{i=1}^n u_{ci} r_{ij}$ と $\beta \sum_{t \neq c} w_{tj}^*$ のトレードオフを実現する手段として、積の形式で $(\sum_{i=1}^n u_{ci} r_{ij}) \times \exp(-\beta \sum_{t \neq c} w_{tj}^*)$ なる改良型の凝集度基準を用いる．もし、その時点のクラスター分割において項目 j が複数のクラスターに帰属しているならば、 $\sum w_{tj}^*$ が大きな値を持つため、凝集度 $\sum_{i=1}^n u_{ci} r_{ij}$ には小さなペナルティを掛けることで、分割の選好度を低下させる．

いま、 s_{cj} を項目 j のクラスター c における共有ペナルティとし、

$$s_{cj} = \exp\left(-\beta \sum_{t \neq c} w_{tj}^*\right) \quad (6)$$

のように求められるとすると、共起度 r_{ij} に共有ペナルティ s_{cj} を掛けることで選好度が低下される．もし、項目 j が唯一のクラスター c にのみ帰属するならば、 $\sum_{t \neq c} w_{tj}^* \rightarrow 0$ であることから共有ペナルティは $s_{cj} \rightarrow 1$ となる．一方、項目 j が複数のクラスターで大きなメンバシップを持つならば、 s_{cj} は小さな値を持つことになり、当該項目-個体の対については、メンバシップの更新時に影響が軽視される．

β は項目分割の排他性の程度を調整する重みであり、大きな値を与えると各項目は複数のクラスターでは大

きな帰属度を持つことができなくなり、排他性が強まる。一方、 $\beta = 0$ のときにはすべての s_{cj} が $s_{cj} = 1$ となり、項目の排他性を考慮しない従来の KLFCCM 法と一致する。改良型の凝集度を用いることで、KLFCCM 法の目的関数は、以下のように修正される。

$$L_{klfccm'} = \sum_{c=1}^C \sum_{j=1}^m \left(\sum_{i=1}^n u_{ci} r_{ij} s_{cj} \right) \log w_{cj} + \lambda_u \sum_{c=1}^C \sum_{i=1}^n u_{ci} \log \frac{\pi_c}{u_{ci}} \quad (7)$$

最適性の必要条件から、各々の更新則は、以下のよう求められる。まず、クラスター容量と個体のメンバシップは、それぞれ、

$$\pi_c = \frac{1}{n} \sum_{i=1}^n u_{ci} \quad (8)$$

$$u_{ci} = \frac{\pi_c \prod_{j=1}^m (w_{cj})^{(r_{ij} s_{cj}) / \lambda_u}}{\sum_{\ell=1}^C \pi_{\ell} \prod_{j=1}^m (w_{\ell j})^{(r_{ij} s_{\ell j}) / \lambda_u}} \quad (9)$$

となる。一方、項目のメンバシップは、

$$w_{cj} = \frac{\sum_{i=1}^n r_{ij} s_{cj} u_{ci}}{\sum_{\ell=1}^m \left(\sum_{i=1}^n r_{i\ell} s_{\ell j} u_{ci} \right)} \quad (10)$$

となる。いずれも、共有ペナルティを考慮した更新が行われることになるので、項目のクラスター重複を避けた分割が得られる。

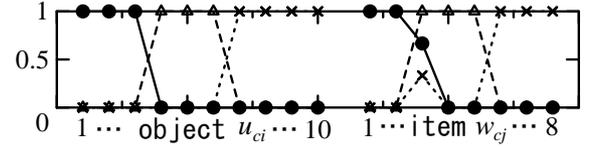
更新アルゴリズムでは、上記の三つの更新則を収束するまで繰り返し適用する。その際、新たに付加した排他性のペナルティ項の不適切な影響を避けるために、重み β を初期値では 0 とおき、徐々に増加させながら排他性を満たしていくものとする。これは、従来の FCCM 法で得られた共クラスター構造の明確化を目的として、項目のメンバシップの排他性を導入する意図に基づいている。

4 数値実験

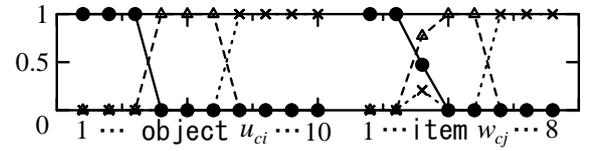
Fig. 1 に示す 10 個体と 8 項目に関する共起関係データを用いて、制約条件の影響を考察する。Fig. 2 に、最大値を 1 に基準化した個体と項目のメンバシップを比較する。クラスター数は $C = 3$ 、ファジィ度の重みは $\lambda_u = 1.0$ とした。三つのクラスターが 3 個の異なる記号 (●, ▲, ×) で示されている。また、提案法において排他性を強いるペナルティ項の重みは、 $\beta = \min\{0.1 \times (t - 1), \beta_{max}\}$ とおき、 $\beta_{max} = 1.0$ と $\beta_{max} = 10.0$ の結果も比較している。ただし、 t は繰り返し数である。KLFCCM 法と同様の排他性を考慮しない分割を初期分割とし、繰り返しアルゴリズムの中で徐々に排他性を強いる機構となっている。

item	1	2	3	4	5	6	7	8
object 1	1	1	1	0	0	0	0	0
object 2	1	0	0	0	0	0	0	0
object 3	1	0	0	0	0	0	0	0
object 4	0	0	1	1	1	0	0	0
object 5	0	0	1	1	1	0	0	0
object 6	0	0	0	0	0	1	1	1
object 7	0	0	0	0	0	1	1	1
object 8	0	0	0	0	0	1	0	1
object 9	0	0	0	0	0	1	0	1
object 10	0	0	0	0	0	1	0	1

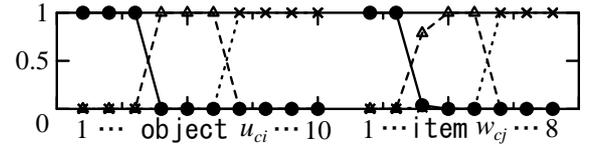
Fig. 1: Artificially generated cooccurrence matrix (black: $r_{ij} = 1$, white: $r_{ij} = 0$)



(a) memberships by KLFCCM



(b) memberships with exclusive penalty for item partition ($\beta_{max} = 1.0$)



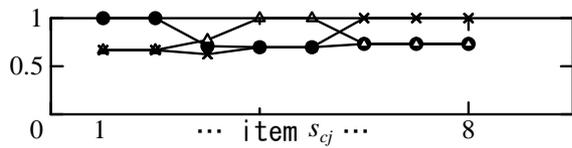
(c) memberships with exclusive penalty for item partition ($\beta_{max} = 10.0$)

Fig. 2: Comparison of memberships u_{ci} and w_{cj}

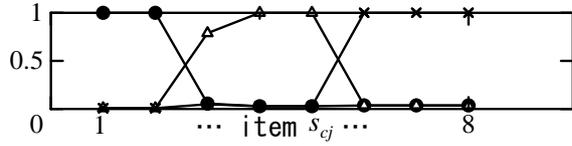
個体のみを排他的に割り当てる KLFCCM 法では、各個体がほぼクリスピーに分割されたが、共有度の高い項目 3 が複数のクラスターに値を持つことがわかる。つぎに、項目に対しても排他性を付加する提案法では、ペナルティ項の重み β が小さい時にはすべてのクラスターへのメンバシップが減少するのみであったが、大きくすることで唯一のクラスターでのみ値が大きくなり、共有度の高い項目 3 についても単一のクラスターに帰属するように改善されている。

つぎに、本研究で新たに付加された共有ペナルティ s_{cj} の挙動を考察する。Fig. 3 に、各項目の各クラスターでの共有ペナルティ s_{cj} を比較する。ペナルティ項の重みが小さい $\beta_{max} = 1.0$ では、すべての s_{cj} が比較的大きな値を持っていると同時に、共有度の大きい項目 3 の値がすべてのクラスターで似た値となっており、唯一のクラスターへの割り当てには寄与していないことが分かる。一方、 $\beta_{max} = 10.0$ では、すべての s_{cj} が 0 または 1 に近い値を持ち、唯一のクラスターへの割り当ての意味合いが強まっている。特に、共有度の大きい項目 3 でもクラスター間の相違を明確にする値となっており、排他性を強調できている。

このように、項目に対しても排他性を付加すること



(a) Sharing penalty of items ($\beta_{max} = 1.0$)



(a) Sharing penalty of items ($\beta_{max} = 10.0$)

Fig. 3: Comparison of sharing penalty of items

で、クラスターを特徴づける項目の解釈が明確となる。

5 おわりに

本研究では、混合多項分布の概念に基づくファジィ共クラスタリングにおける項目の排他的制約について議論した。明示的に排他性を持たせることで、クラスターを特徴づける項目の解釈に有効となる。今後の課題としては、個体と項目数のアンバランスに対応した重み付けや、ペナルティ項の重みの自動設定、妥当性尺度の活用⁸⁾などが挙げられる。また、文書解析⁹⁾や情報選別^{10, 7)}などへの応用展開も期待される。

なお、本研究の一部は文部科学省科学研究費補助金基盤研究(C) (#26330281) に基づくものである。

参考文献

- 1) L. Rigouste, O. Cappé and F. Yvon, "Inference and evaluation of the multinomial mixture model for text clustering," *Information Processing and Management*, **43-5**, 1260/1280, 2007

- 2) K. Honda, S. Oshio and A. Notsu, "FCM-type Fuzzy Co-clustering by K-L Information Regularization," *Proc. of 2014 IEEE International Conference on Fuzzy Systems*, 2505/2510, 2014
- 3) C.-H. Oh, K. Honda and H. Ichihashi, "Fuzzy clustering for categorical multivariate data," *Proc. of Joint 9th IFSA World Congress and 20th NAFIPS International Conference*, 2154/2159, 2001
- 4) S. Miyamoto, H. Ichihashi and K. Honda, *Algorithms for Fuzzy Clustering*, Springer, 2008
- 5) J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, 1981
- 6) K. Tsuda, M. Minoh and K. Ikeda: Extracting straight lines by sequential fuzzy clustering, *Pattern Recognition Letters*, **17**, 643/649, 1996
- 7) K. Honda, C.-H. Oh, Y. Matsumoto, A. Notsu and H. Ichihashi, "Exclusive partition in FCM-type co-clustering and its application to collaborative filtering," *International Journal of Computer Science and Network Security*, **12-12**, 52/58, 2012
- 8) K. Honda, M. Muranishi, A. Notsu and H. Ichihashi, "FCM-type cluster validation in fuzzy co-clustering and collaborative filtering applicability," *International Journal of Computer Science and Network Security*, vol.13, no.1, pp.24-29, 2013.
- 9) Y. Matsumoto, K. Honda, A. Notsu and H. Ichihashi, "FCM-type co-clustering of categorical multivariate data with exclusive partition," *Proc. of 6th International Conference on Soft Computing and Intelligent Systems and 13th International Symposium on Advanced Intelligent Systems*, pp. 1796-1800, 2012.
- 10) K. Honda, A. Notsu and H. Ichihashi, "Collaborative filtering by sequential user-item Co-cluster extraction from rectangular relational Data," *International Journal of Knowledge Engineering and Soft Data Paradigms*, vol.2, no.4, pp.312-327, 2010.

探索点間の結合が探索に及ぼす影響についての検討

○酒井駿介 畠中利治 (大阪大学)

A study on effect of search point network in evolutionary algorithms

*S. Sakai and T. Hatanaka (Osaka University)

Abstract— Recently, evolutionary computation families are used to find a solution of complex problems. Particle Swarm Optimization(PSO) is a member of such methods that is implemented in simple but has a powerful search ability. However, a theoretical aspect of search ability is not enough studied. In this study, a condition of reaching consensus in PSO particles is considered. From this viewpoint, we have examined an effect on search performance of a particles network for example a ring topology and a random graph of PSO.

Key Words: Particle Swarm Optimization, topology

1 はじめに

粒子群最適化 (Particle Swarm Optimization : PSO)¹⁾ は、比較的容易な実装と関数最適化における優れた性能から注目を集めており、様々な改良を施した新たな手法が提案され、ベンチマーク問題によってその有用性の検証がなされてきている²⁾。

しかしながら、PSOの探索性能については、単一の粒子の動作モデルの解析から粒子の挙動の安定条件が示されるなどの研究³⁾はあるが、群の働きとして探索性能を理解する立場からの研究はあまりなされていない。また、粒子同士の結合を定めるトポロジーが導入され、粒子が参照する近傍を定義することによって局所解への収束をおさえ、大域解への収束をうながすアプローチがはやくから導入されている⁴⁾⁵⁾⁶⁾が、トポロジー間の比較が行われるにとどまり、近傍範囲の条件の探索性能への影響は十分に調べられているとは言えない状況である。

そこで、本研究では、探索点同士の結合で形成される探索点をネットワークと考え、探索点ネットワークの特性がどのようにPSOの探索性能に影響を与えているかを調べることを目的としている。本報告では、そのための準備として、探索点同士のつながりが探索性能に与える影響を二つのベンチマーク問題で評価した結果を報告する。

2 PSO

PSOは多点探索による確率的最適化手法であり、複数の探索粒子が解空間内を動き回ることによって最適解を探索する。PSOの探索粒子はそれぞれの持つ位置と速度のベクトルによって表現される。探索粒子は次に示す更新式によって位置と速度の更新を行い、探索を進める。

$$v_{id}(t+1) = w(t) + c_1 R_1(p_{id} - x_{id}(t)) + c_2 R_2(l_d - x_{id}(t)) \quad (1)$$
$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (2)$$

w, c_1, c_2 は正の定数であり、 R_1, R_2 は $[0 \sim 1]$ の一様乱数である。また i は探索粒子の番号、 d は次元数、 t はステップ数を表している。実験でよく用いられているパラメータの設定値は $w = 0.7, c_1 = c_2 = 1.4$ であり、今回の実験でもこの設定値を用いる。 p_{id} (パーソナルベ

スト)は探索粒子が過去を含め一番最適な評価値を得たときの位置ベクトルである。また、 l_d (ローカルベスト)は近傍の p_{id} を参照して最適な評価値である位置ベクトルである。 l_b は探索粒子のトポロジーに依存して決定される。探索粒子のトポロジーが Fully connected であるとき、 l_b はすべての粒子の p_{ib} の最良の評価値となる。このとき l_b は g_b (グローバルベスト)と呼ばれる。

3 実験

探索点ネットワークの構造が探索性能にどのような影響を与えているかを調べるために計算機実験を行った。目的関数として、進化計算のテスト関数としてよく用いられる単峰性関数の Sphere 関数と、多峰性関数の Rastrigin 関数を用いた。

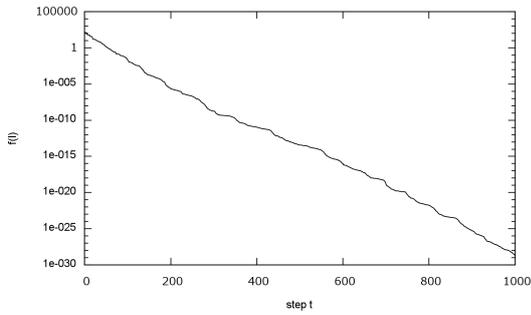
実験設定について述べる。目的関数の次元数は 20 で、探索粒子の数は 20、世代数を 1000 とした。実験を行ったトポロジーは Fully connected, Ring, Ransom の 3 種類であり、すべて双方向通信が可能な無向グラフとした。Random の接続は探索粒子間の接続確率を 0.05 とした。また、Random に関しては探索粒子の数を 40 と 60 に増やした場合の実験も行った。

本実験における各トポロジーの平均次数を Table 1 に示す。

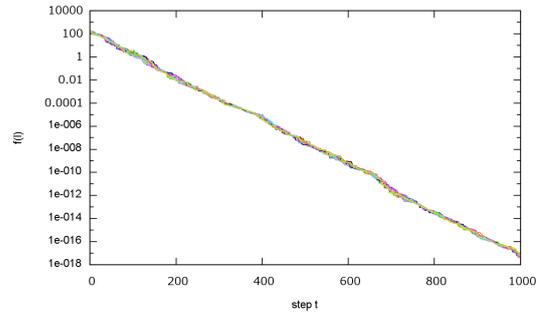
Table 1: Topology used for an experiment.

Topology	Particles	Average degree
Fully connected	20	19
Ring	20	2
Random(0.05)	20	0.7
	40	1.6
	60	2.57

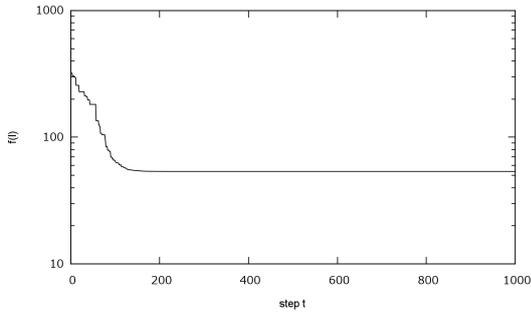
計算機実験の結果を Fig. 1~Fig. 5 に示す。Fig. 1 は Fully Connected, Fig. 2 は Ring, Fig. 3 は Random, Fig. 4 は Random の探索粒子が 40 粒子の場合、Fig. 5 は Random の探索粒子が 60 粒子の場合の結果である。グラフにプロットしている値は各世代における、(a) Sphere 関数の場合の $f(l)$, (b) Rastrigin 関数の場合の $f(l)$, (c) Sphere 関数の場合の $f(x)$, (d) Rastrigin 関数の場合の $f(x)$, (e) rastrigin 関数の場合の v である。今回の実験では、探索粒子の動きに着目する。



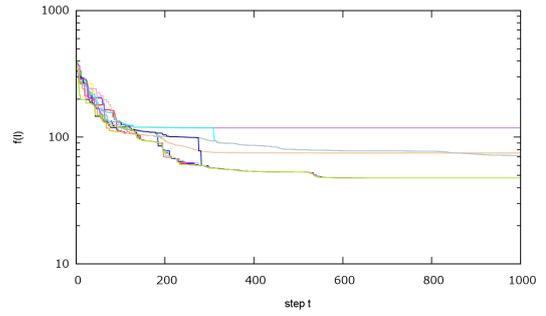
(a) The value of $f(l)$ in case of Sphere function.



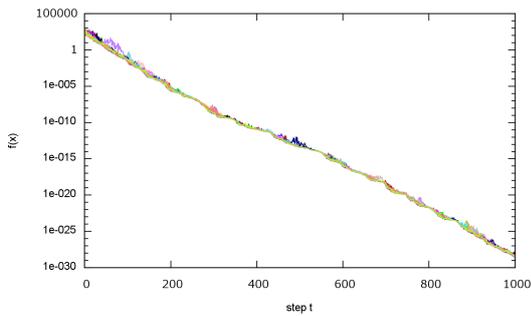
(a) The value of $f(l)$ in case of Sphere function.



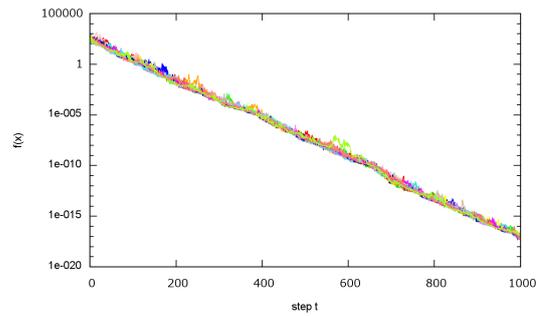
(b) The value of $f(l)$ in case of Rastrigin function.



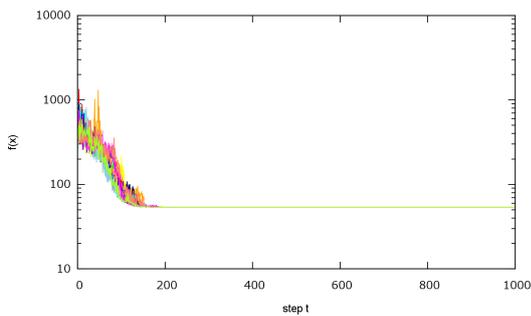
(b) The value of $f(l)$ in case of Rastrigin function.



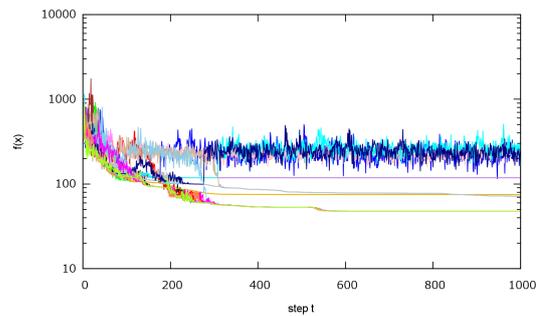
(c) The value of $f(x)$ in case of Sphere function.



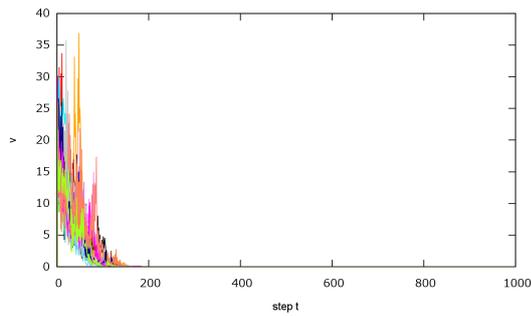
(c) The value of $f(x)$ in case of Sphere function.



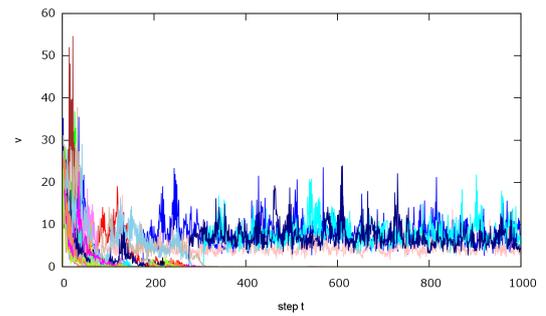
(d) The value of $f(x)$ in case of Rastrigin function.



(d) The value of $f(x)$ in case of Rastrigin function.



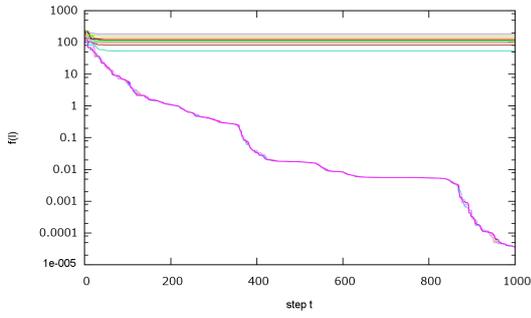
(e) The value of v in case of Rastrigin function.



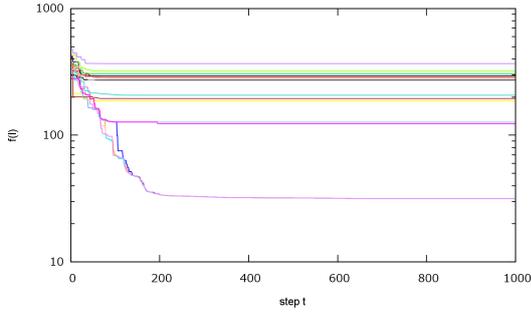
(e) The value of v in case of Rastrigin function.

Fig. 1: The $f(x)$ and $f(l)$ of PSO of the Fully connected.

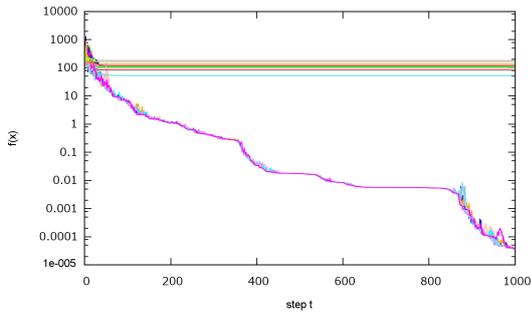
Fig. 2: The $f(x)$ and $f(l)$ of PSO of the Ring.



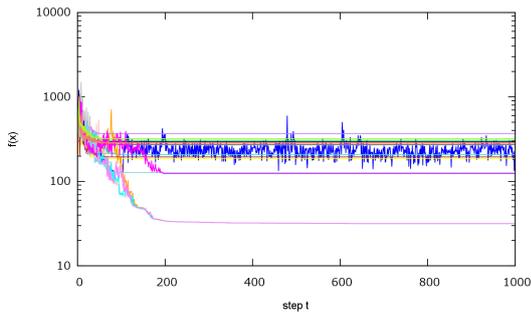
(a) The value of $f(l)$ in case of Sphere function.



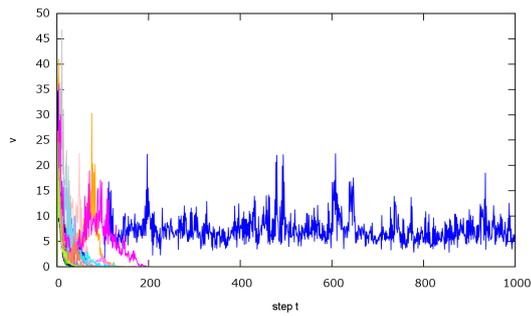
(b) The value of $f(l)$ in case of Rastrigin function.



(c) The value of $f(x)$ in case of Sphere function.

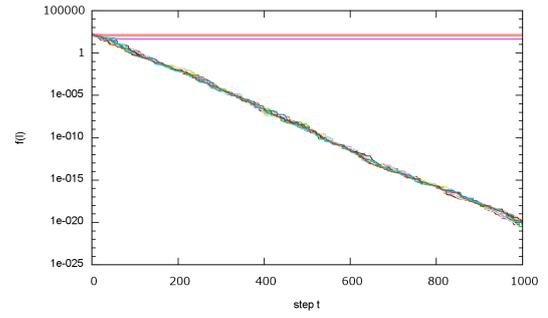


(d) The value of $f(x)$ in case of Rastrigin function.

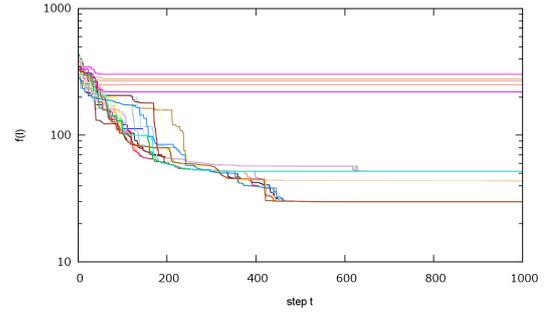


(e) The value of v in case of Rastrigin function.

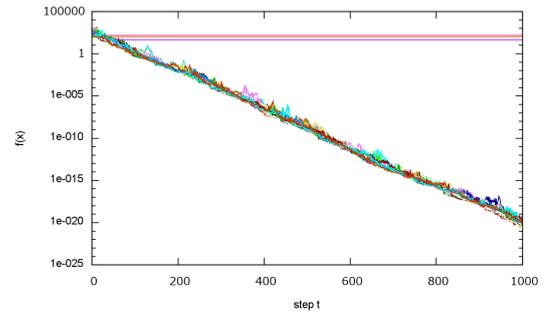
Fig. 3: The $f(x)$ and $f(l)$ of PSO of the Rand.



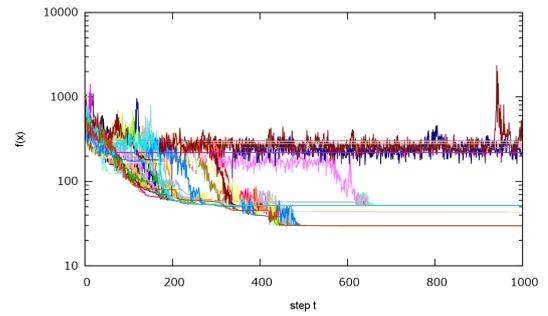
(a) The value of $f(l)$ in case of Sphere function.



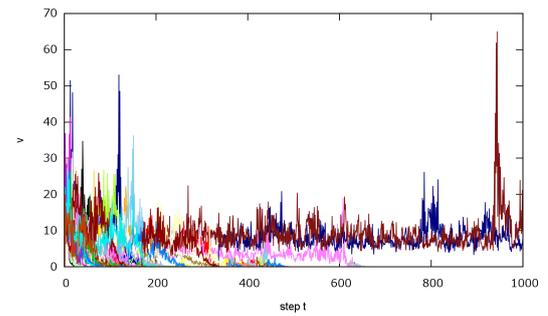
(b) The value of $f(l)$ in case of Rastrigin function.



(c) The value of $f(x)$ in case of Sphere function.

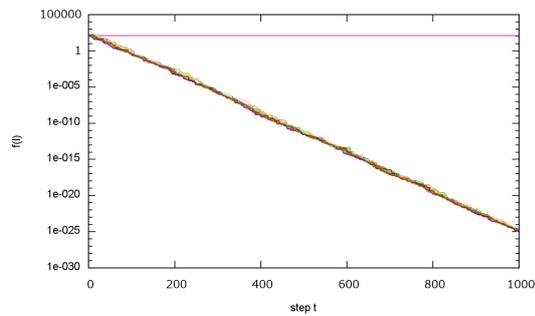


(d) The value of $f(x)$ in case of Rastrigin function.

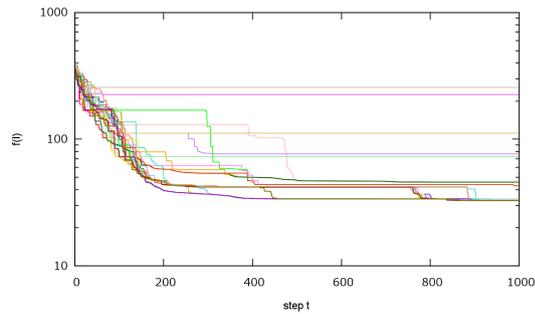


(e) The value of v in case of Rastrigin function.

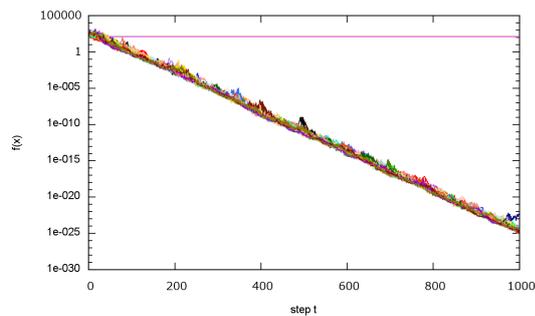
Fig. 4: The $f(x)$ and $f(l)$ of PSO(40 particles) of the Rand.



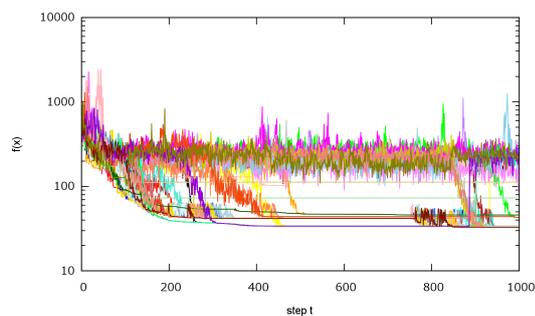
(a) The value of $f(l)$ in case of Sphere function



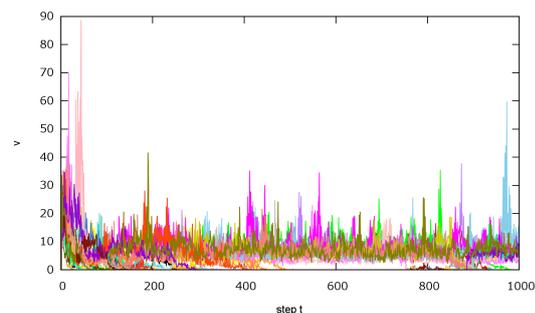
(b) The value of $f(l)$ in case of Rastrigin function



(c) The value of $f(x)$ in case of Sphere function



(d) The value of $f(x)$ in case of Rastrigin function



(e) The value of v in case of Rastrigin function

Fig. 5: The $f(x)$ and $f(l)$ of PSO(60 particles) of the Rand

4 結果

Fig. 1(a), Fig. 2(a), Fig. 3(a) より, 探索粒子の数が同じ条件であり, 目的関数が Sphere 関数の場合, 探索粒子の平均次数が高いほど性能がよいことが分かった. これは Sphere 関数は単峰性関数であるため l_d の共有範囲の広さがそのまま探索性能につながったものであると考えられる.

Random では他の探索粒子と一切結合がない探索粒子が現れることがある. Fig. 3(c), Fig. 3(d), Fig. 4(c), Fig. 4(d), Fig. 5(c), Fig. 5(d) より, 独立している探索粒子は自らの位置情報のみを用いて探索を進めるため, 早々に局所解へ陥ってしまう. また, 全体の探索粒子の数のうち独立している探索粒子の数の割合が大きくなるということは平均次数が低くなることを表している. そして, Fig. 3(d), Fig. 4(d), Fig. 5(d)において平均次数が低いものほど $f(x)$ の変化が小さくなる傾向があった. Fig. 3(e), Fig. 4(e), Fig. 5(e) の速度 v もまた平均次数が低いものほど値の変化が小さくなる. これは位置情報のやりとりをする探索粒子の数が少ないためだと考えられる. しかし, 平均次数が一番高い Fully Connected である Fig. 1(d) もまた $f(x)$ の変化が小さい. これは逆にすべての探索粒子と情報のやり取りをするためすべての探索粒子が一点を目指して移動していくことになる. したがって Fig. 1(d) の場合も $f(x)$ の変化が小さくなったと考えられる.

5 おわりに

PSO の探索点同士のつながりが探索性能に与える影響を計算機実験により, 二つのベンチマーク問題で評価した. その結果, トポロジーの平均次数が探索性能に影響を与える傾向があった. 今後は, トポロジーの平均次数に着目し, Star, Tree, Line などの他のトポロジーや, Random で作成する別の結合のものについて実験を進めていく. また, 探索点ネットワークを行列で表した時の固有値や rank が進化計算の探索性能にどのような影響を及ぼしているかを調べていく.

参考文献

- 1) J. Kennedy and R. Eberhart : Particle swarm optimization, Proceedings 1995 IEEE International Conference on Neural Networks, 1942/1948, (1995)
- 2) 齊藤利通 : 粒子群最適化と非線形システム, IEICE Fundamentals Review Vol.5 No.2, 155/161, (2011)
- 3) 若佐裕治, 田中幹也, 明石卓也 : Particle Swarm Optimization アルゴリズムの安定性解析, システム制御情報学会論文誌, Vol.23, No.1, 9/15, (2010)
- 4) R. Eberhart and J. Kennedy : A new optimizer using particle swarm theory, Proceedings 6th International Symposium on Micromachine Human Science, 39/43, (1995)
- 5) J. Kennedy and R. Mendes : Population Structure and Particle Swarm Performance, Proceedings of the 2002 IEEE Congress on Evolutionary Computation, 1671/1676, (2002)
- 6) R. Mendes, J. Kennedy and J. Neves : The Fully Informed Particle Swarm: Simpler, Maybe Better, IEEE transactions on evolutionary computation, Vol.8, No.3, 204/210, (2004)

三次元点群データを利用した一般物体認識と重なった物体への対応

○細井勇揮 高橋智一 鈴木昌人 青柳誠司 (関西大学)

Generic Object Recognition Using 3D Point Cloud and Its Application to Overlapped Objects

* Y. Hosoi, T. Takahashi, M. Suzuki and S. Aoyagi (University of Kansai)

Abstract— A method for generic object recognition to categorize objects such as desk, chair, etc., has been already proposed by authors, in which the point distribution data and the region size data of width, depth, and height are considered. However, two (or more) objects close to each other are likely segmented to one overlapping region. To address this problem, a new segmentation/recognition method is proposed in this article. This method imitates a human who seems to repeat segmentation and recognition complementally. As for overlapping chair and desk, segmentation and recognition were successfully conducted by the proposed method.

Key Words: Generic object recognition, Point cloud, AdaBoost classifier, Segmentation

1 緒言

ロボットが作業空間を自由に移動するためには、自己位置推定と地図作成を同時に行う SLAM (Simultaneous Localization and Mapping) と呼ばれる問題を解く必要があり、これに関して様々な研究が行われている[1-3]。これらの多くは、走行できる freespace の検出に主眼を置いており、室内の物体の種類 (カテゴリ) の認識については注目していない。一方人間は、部屋の中にある物体の種別を認識することでその部屋の種類 (会議室, オフィスルーム, 実験室等) を把握して, freespace を見つける効率を上げているものと考えられる。すなわち, 過去の経験から, その部屋のレイアウトではここに freespace が有るはずだ, といった推測をしているものと考えられる。

本研究室では既にオフィス環境を想定し, 距離画像センサのデータからオフィス機器を認識する手法を提案している[4]。ここでの認識は, 特定物体認識ではなく, 一般物体認識を想定する。この手法の処理手順は以下の通りである: 様々な種類の椅子および椅子以外の物体を様々な方向から捉えた距離画像を多数用意し, これらの三次元点群データの空間分布に基づき AdaBoost (Adaptive Boosting) 識別器[5]を学習して獲得する。AdaBoost は一個一個の識別器は認識率があまり高くないが, この識別器を多数用いることで, 最終的に高い認識率を得る手法である。学習済みの AdaBoost 学習器に未知の距離画像を入力すると, それが机であるか, 机でないかの判定を行うことができる。椅子, 人間等, 部屋に存在するその他の物体について同様に AdaBoost 学習器を獲得する。

実際の未知画像の処理は以下のように行う: 部屋の距離画像を取得し, これを距離に応じて領域分割 (セグメンテーション) する。各領域の点群データを AdaBoost 識別器に入力し, その領域のカテゴリ識別 (椅子であるか, 机であるか) を行う。

この手法では, 物体のセグメンテーションが正しく行えれば認識率は高いが, それに失敗すると認識が全くうまくいかないという問題点があった。これ

を解決するために, 本報では RGB 情報を用いた領域の小領域への再分割と, それら小領域の組み合わせを考えた領域統合を行い, 重なっている物体のセグメンテーションとカテゴリ識別を行う手法を提案する。具体的には, 1) 距離情報によりセグメンテーションされた領域を, RGB の情報を用いてさらに小さく再分割し, 2) 分割された小領域の全ての組み合わせを考え, 各組合せに対して領域を統合し, 3) 最も AdaBoost 学習器の出力スコアが高い組み合わせを採用する。

本報で提案する手法は以下の推察に基づいている: 人間は重なっている物体を識別するときに, エッジやカラー等の情報に基づいて領域のセグメンテーションを行い, 分割された領域を長い経験から獲得・記憶している種々の物体の概念情報と照らし合わせ, うまく照合できない場合はセグメンテーションをやり直すことを繰り返していると思われる。すなわち人間はセグメンテーションとリコグニションを相補的に繰り返すことにより, 重なっている物体の分離・認識を可能にしていると思われる。

2 提案手法

2.1 距離画像からのヒストグラム作成

本節では, 既に本研究室で提案している物体のカテゴリ識別法について説明する[4]。まず点群データの分布情報と物体の寸法 (幅, 奥行き, 高さ) の情報を AdaBoost によって学習する。Fig. 1(a), (b) に示すようなカラー画像と距離画像が得られた時, 距離に応じて Fig. 1(c) に示すように領域分割を行う。ここで分割された各領域を一つの物体と考える。この各領域に含まれている三次元点群データを Fig. 2 に示すように $5 \times 5 \times 5$ の小領域に分割し, 各小領域に含まれる点の数を正規化したヒストグラムを作成する。このヒストグラムに各領域の寸法 (Fig. 2 中の A, B, C. 正規化はせず実寸法) を加えた 128 次元のベクトル量を, AdaBoost の一つの入力データとする。

2.2 AdaBoost 学習の方法の変更 (正面から見るように回転変換した点群を学習データとする)

本研究では視線方向の変化に対応させるために,

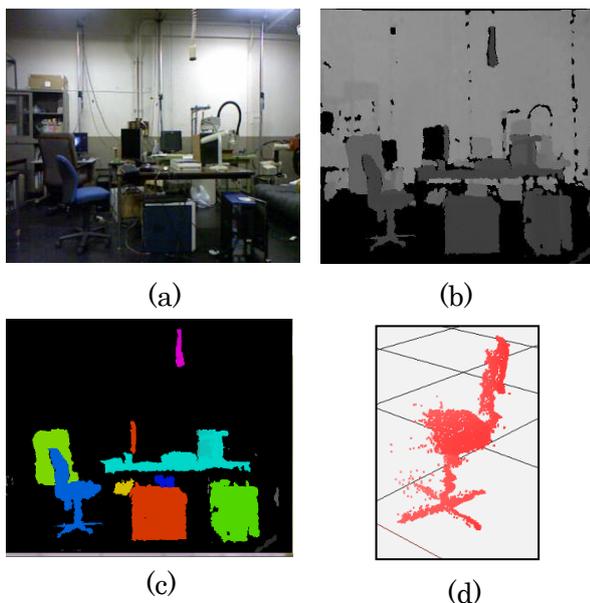


Fig. 1 Example image: (a) RGB image captured by CCD camera equipped with Kinect, (b) depth image obtained by Kinect, (c) extracted regions, each color of which corresponds to one region, and (d) extracted point cloud for “chair”.

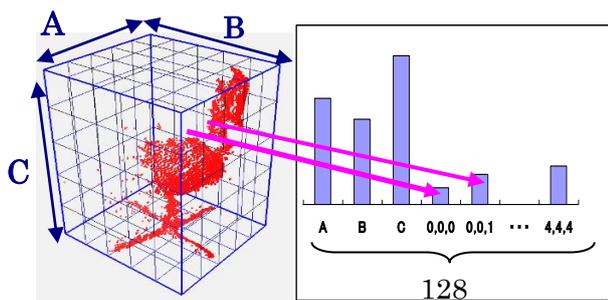


Fig. 2 Extracted region is divided to $5 \times 5 \times 5 = 125$ sub-regions. Width (A), depth (B), height (C), and number of measured points inside each sub-region are assembled to one histogram, i.e., the histogram consists of 128 ($=3+125$) bins. This histogram is an input data to Adaboost classifier.

様々な角度から点群データを取得し、これらをそのまま用いて Adaboost 学習器に学習させていた。この手法では、椅子の認識において、椅子の特徴的部分である背もたれに点群が集中している、等の情報が失われてしまい椅子の認識率が低下してしまっていた (Fig. 3).

本報告では、椅子の認識において、様々な視線方向から取得したデータをそのままの角度で学習させるのではなく、取得した点群をコンピュータ上で回転変換して椅子の向きを統一してから学習させる事によって、認識率を向上させる (Fig. 4).

具体的な手順は以下の通りである：点群を $5 \times 5 \times 5$ の小領域に分割した際の高さが一番上にある配列に注目する。椅子の場合その領域には背もたれの部分しか存在しないと考えられ、背もたれの角度によるパターンの配列を用意しておき、それに当てはまる角度だけ点群全体を回転させてから認識するように

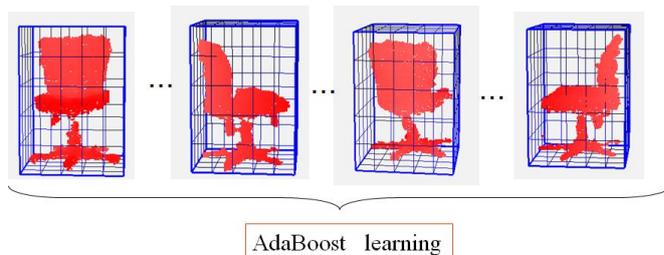


Fig. 3 Learning of point cloud data without considering object orientation (our previous method)

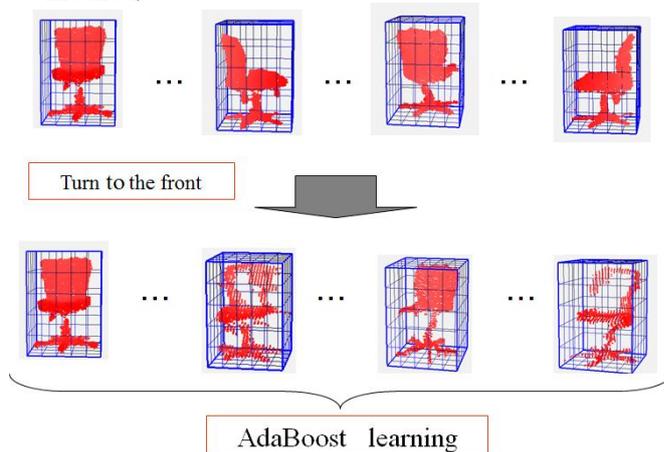
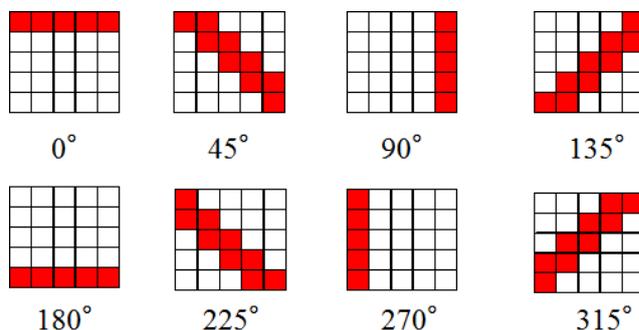
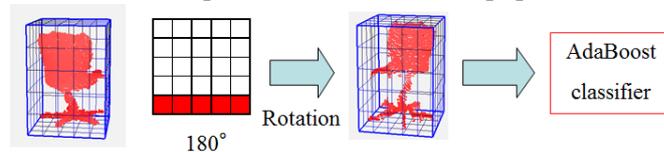


Fig. 4 Learning of point cloud data with considering object orientation, in which data is rotated so that the object front faces camera.



(a) Rotation patterns based on top points of



(b) Example of point cloud rotation for AdaBoost classifier

Fig.5 Rotation of point cloud data based on top points of chair back.

する(Fig. 5).

2.3 重なった領域の小領域への再分割, 小領域の組み合わせのうち AdaBoost 識別器のスコアが高いものを正解とする

距離画像において、隣接する点間距離が 30 mm 以下の場合に統合することでラベリングを行い、点群

データのセグメンテーション（領域分割）を行う。

ここで、距離情報による領域分割だけでは物体間の距離が近いと領域が重なってしまい、カテゴリの認識率が低下してしまっていた。そこで今回、人間が行っていると推察される、セグメンテーションと認識を相補的に行って物体を正確に分割・認識する手法を提案した。距離画像と RGB 画像の取得については、RGB-D センサである Kinect (Microsoft 社) を使用することにする。

具体的な手順を以下に示す。1) 距離画像が重なってしまった領域を、Kinect で同時に獲得される RGB 値（色情報）に基づき、k-means 法でさらに小さい複数の領域へ再分割する。ここで k-means 法のクラスター数は、様々な数値で試した結果により経験的に k=4 を採用した（色情報により 1 個の領域を 4 個の小領域に再分割する）、2) 分割された小領域の全ての組み合わせを考え、各組合せに対して小領域を統合し、この統合された領域について AdaBoost 学習器によるカテゴリ識別を行う、3) 最も AdaBoost 学習器の出力スコアが高い組み合わせを採用することで、重なっている椅子と机を分離して識別する。

3 実験

3.1 抽出された点群データのカテゴリ認識

Kinect で距離画像を撮影し、抽出したヒストグラムを用いて椅子と机について AdaBoost 学習を行った。学習器の精度を検証するため、学習には用いなかった未知データを識別器に入力した。具体的には、距離画像による領域分割後の椅子のデータ（画像から抽出された椅子の点群データ）120 個と椅子以外のデータ（画像から抽出された椅子以外の物体領域の点群データ）144 個を用いて学習を行い、検証用の未知のデータとして椅子のデータ 220 個と椅子以外のデ

Table 1 Experimental recognition result by AdaBoost (chair) without considering object orientation

(a) Learning results				(b) Verification results			
		output				output	
		chair	other			chair	other
input	chair	115	5	input	chair	208	12
	other	36	108		other	36	184

Table 2 Experimental recognition result by AdaBoost (chair) using point cloud data which is rotated so as that object front faces camera

(a) Learning results				(b) Verification results			
		output				output	
		chair	other			chair	other
input	chair	119	1	input	chair	210	10
	other	24	120		other	11	209

Table 3 Experimental recognition result by AdaBoost

(a) Learning results				(b) Verification results			
		output				output	
		desk	other			desk	other
input	desk	170	0	input	desk	149	0
	other	0	204		other	0	208

ータ 220 個を用いて検証を行った。

既報の椅子の姿勢を考慮しない学習方法での学習時の誤差と未知データの検証結果を Table 1 に示す。今回提案した椅子の正面がカメラ (Kinect) を向くように距離画像データを回転させたものを用いて学習器を作成した場合の結果を Table 2 に示す。これより、前者の方法では未知画像の認識において、 $(208+184)/440 \times 100 = 89\%$ であった椅子の認識率が、後者の方法では $(210+209)/440 \times 100 = 95\%$ まで向上した。これより物体の姿勢を揃えた距離画像データを用いて学習することの効果が確認できた。

机について、学習時に机 170 個、机以外 204 個、検証時に机 149 個、机以外 208 個のデータを用いて認識実験を行った結果を Table 3 に示す。机については、方向を揃えなくても高い認識率（今回の例では 100%）が得られた。これは机の大きさが部屋内の他の物体に比べて明らかに大きく、領域の寸法のデータ (Fig. 2 の A, B, C のパラメータ) が、AdaBoost 識別器を構成する際に特徴量として大きく効いているためだと思われる。

3.2 重なった領域の小領域への再分割と、小領域の組み合わせによるカテゴリ認識

Fig. 6 (a) に RGB-D 画像（距離画像に色情報を重ねたもの）を示すようなシーンにおいて、距離データをもとに領域を分割した結果を Fig. 6 (b) に示す。距離情報のみを用いた場合、物体の距離が近い場合、Fig. 6 (c) に示すように 2 物体（この例では机と椅子）が重なって同一の領域にセグメンテーションされてしまう。そこで、Fig. 6 (d) に示す RGB データ（色情報）に基づき、この重なった領域を k-means 法で 4 個の小領域に再分割した。結果を Fig. 6 (e) に示す。この図の 4 色の領域を label 0~3 と呼ぶこととする (Table 4 に対応を示す)。

ここで、構成点数が 200 個以下の、大きな領域から離れてしまった小さな領域については、経験的にノイズであると判断し、データから削除した。

その後、すべての label 0~3 の 4 領域の組み合わせを考え、この組み合わせの点群データを椅子と机の 2 個の AdaBoost 識別器に入力し、その出力値を評価した。出力値は 0~1 の値をとり、0.5 以上であれば正解（例えば“机”）、それ以下であれば不正解（例えば“机以外の物体”）であり、この値が高ければ高いほど（1 に近いほど）、その物体である確率が高くなる。

結果を Table 5 に示す。この表で網掛けを施した部分が、各椅子と机の AdaBoost 識別器において、出力値が最大となる組み合わせである。すなわち、椅子の場合は Label 0 と 2 の小領域の組み合わせが最も椅子らしく、机の場合は Label 1 と 3 の小領域の組み合わせが最も机らしい。ということがわかる。

この結果をもとに、複数の小領域を再統合した結果を Fig. 6 (f) に示す。これより重なっていた机と椅子の領域が、正しく 2 つの領域に分割され、それぞれ机と椅子と正常にカテゴリ識別できていることがわかる。

セグメンテーションに失敗した例を Fig. 7 に示す。原因としてノイズに対しての処理が甘いと、点群ヒストグラムの作成に影響してしまい、これを AdaBoost 識別器に入力すると間違ったカテゴリ識別

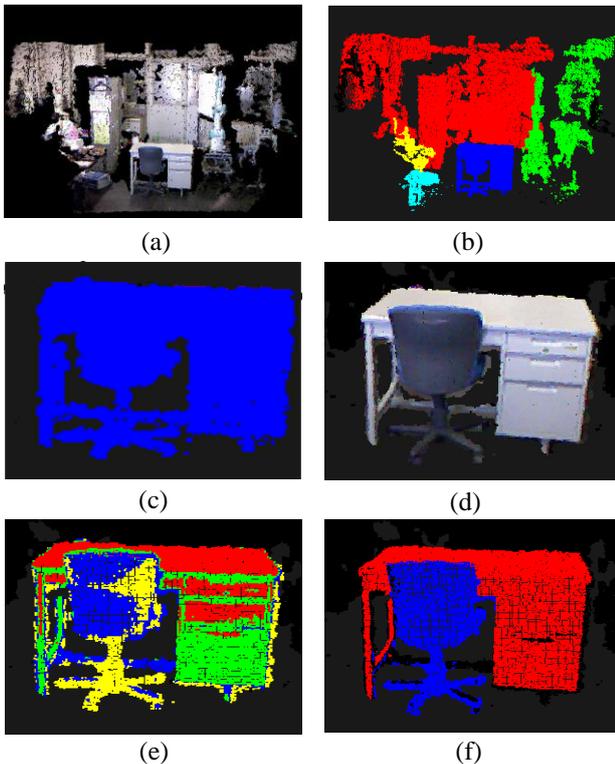


Fig. 6 Experimental result: (a) RGB-D image, (b) extracted regions based on depth information, each color of which corresponds to one region, (c) overlapping region, (d) RGB image of overlapping region, (e) segmentation based on RGB information using k-means method, and (f) segmentation and recognition result.

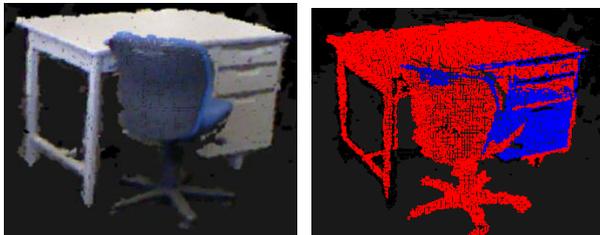


Fig. 7 Failure example of segmentation/recognition: (a) RGB image of overlapping region, (b) wrong segmentation/recognition result.

結果が出力されてしまうものと思われる。

4 結論

既報[4]の3次元点群分布を用いた物体のカテゴリ認識手法に対し、視線方向の変化を考慮して点群を学習することで、認識率を向上させることに成功した。

距離情報に基づき領域分割をした際に重なってしまった領域に対し、RGB情報を用いてさらに小領域に再分割し、すべての小領域の組み合わせに対してAdaBoost識別器を用いたカテゴリ認識を行い、最もスコアの高い組み合わせを用いて領域を再統合することで、セグメンテーションと物体のカテゴリ識別を精度良く行う手法を提案した。この手法は人間がセグメンテーションとリコグニションを相補的に繰り返し行っていることにインスパイアされている。

Table 4 Correspondence of color in Fig. 8(e) to label number

Label number	Color
0	red
1	green
2	blue
3	yellow

Table 5 Recognition result for combination of label, which is AdaBoost output value ranging from 0 (least probable) to 1 (most probable)

Label number	Chair	Desk
0&1	0.33	0.68
0&2	0.32	0.39
0&3	0.27	0.65
1&2	0.38	0.31
1&3	0.37	0.44
2&3	0.62	0.19
0&1&2	0.33	0.61
0&1&3	0.33	0.63
0&2&3	0.29	0.39
1&2&3	0.37	0.44
0&1&2&3	0.35	0.46

オフィスにある認識対象として椅子と机を採用し、高い認識率を得ることに成功した。今後はRGB画像のノイズ処理の性能を向上させ、領域の再分割の精度を上げることが課題である。

文献

- [1] M. Tomono, "3-D Object Map Building Using Dense Object Models with SIFT-based Recognition Features," in Proc. IROS 2006, pp. 1885-1890, 2006.
- [2] S. Ikeda and J. Miura, "3D Indoor Environment Modeling by a Mobile Robot with Omnidirectional Stereo and Laser Range Finder," in Proc. IROS 2006, pp. 3435-3440, 2006.
- [3] A. Davison, "Real-time Simultaneous Localization and Mapping with a Single Camera," in Proc. ICCV, pp. 1403-1410, 2003.
- [4] 石丸達也, 小泉智資, 高橋智一, 鈴木昌人, 青柳誠司, "Kinect から取得した濃淡画像と距離画像の併用による室内物体の抽出と認識, ROBOMECH2013, 2A2-J06
- [5] Y. Freund and R. E. Schapire: "Experiments with a new boosting algorithm", in Proc. Of the 13th Intl. Conf. Machine Learning, pp.148-156, 1996

妥当性基準を用いた x -means について

○藏口 雄丈 瀨砂 幸裕 (近畿大学)

On Cluster Validity Measure based x -means

*Y. Kuraguchi and Y. Hamasuna (Kinki University)

Abstract— Non-hierarchical clustering such as k -means and fuzzy c -means clustering need to be set the number of clusters before execution. On the other hand, x -means is known as the method which estimates the number of clusters automatically. The x -means is constructed with two steps: 1) partition process based on k -means, 2) evaluation process by using information criteria, e.g. AIC or BIC. Cluster validity measures are also known as the measure for evaluating the result of cluster partition. This paper proposes a novel x -means approach which uses cluster validity measures instead of information criteria. The effectiveness of cluster validity measure based x -means is verified through numerical examples.

Key Words: cluster validity measures, x -means, k -means, hard c -means

1 はじめに

今日, Web 情報検索, バイオインフォマティクス, マーケティングなどの分野で, データ解析技法の一つであるクラスタリングが用いられている. クラスタリングは大量のデータから, 未知の構造やパターンを発見するために用いられ, データ解析におけるクラスタリングとは, 与えられたデータ集合を外的基準を用いずに複数のグループに分類する技法を指しており, 分割した各グループはクラスタと呼ばれている. 同じクラスタに属するデータは似た性質を持ち, 違うクラスタに属するデータは異なる性質を持つ. クラスタリングには階層的な手法と非階層的な手法があり, 非階層的な手法の代表的な技法として k -means がある¹⁾. k -means は分割に曖昧さを認めないハードな分割であるため, 分割に曖昧さを認めるファジィ c -平均法²⁾³⁾ と対比してハード c -平均法 (HCM) とも呼ばれる. 本稿では, ハード c -平均法 (HCM) という呼称を用いる.

HCM は与えられたデータ集合に対して, ユーザがクラスタの分割数を指定する必要があるが, データ集合毎に最適なクラスタ数は異なる. そこで, 定量的に分割結果を評価し, 最適なクラスタ数を決定する指標として妥当性基準が知られている⁴⁾⁵⁾⁶⁾. 妥当性基準は, 各クラスタ毎のまとまりが良く, 各クラスタの分割がはっきりしている場合に良好な結果を示すように構成されている.

一方, クラスタ数を設定せずに分割を行う手法として x -means が提案されている⁷⁾⁸⁾. x -means は HCM を $c=2$ とし, 分割した各クラスタに再帰的に HCM を適用することでクラスタ分割を行う手法である. クラスタ分割の前後の状態に情報量基準¹⁰⁾¹¹⁾ を適用し, その評価結果により分割の要否を判断することで, 分割数を自動決定する点が x -means の特徴である.

従来の x -means では評価指標に情報量基準を用いるため, 対数尤度の計算と確率モデルの推定が必要であり, 計算コストが膨大となる. 特に高次元データに対して x -means を適応する場合, 分割処理は高速であるにも関わらず, 分割結果の評価に時間がかかり, 処理全体の高速性が失われるという問題がある. 本稿では評価指標に妥当性基準を用いることで高速な処理の実現を目的とする x -means を提案する.

以降 2 章で HCM, 妥当性基準, 従来の x -means について述べる. 次に 3 章で提案手法の x -means について述べ, 4 章で人工データ 2 つ, 実データ 2 つを用いた数値例を示し, 提案手法の有効性を検証する. 最後に 5 章で本稿のまとめを述べる.

2 準備

2.1 ハード c -平均法

HCM の目的関数および制約条件は式 (1), (2) で表される.

$$J_h(U, V) = \sum_{i=1}^c \sum_{k=1}^n u_{ki} \|\mathbf{x}_k - \mathbf{v}_i\|^2 \rightarrow \min \quad (1)$$

$$U_h = \{(u_{ki}) : u_{ki} \in \{0, 1\}, \sum_{i=1}^c u_{ki} = 1, \forall k\} \quad (2)$$

ここで, $X = \{\mathbf{x}_k | \mathbf{x}_k = (x_{k1}, \dots, x_{kp})^T \in \mathbb{R}^p, k = 1 \sim n\}$, $U = (u_{ki})$, $V = (\mathbf{v}_i)$ である. u_{ki} は帰属度であり, 個体がどのクラスタに属しているのかを表す. \mathbf{v}_i は p 次元ベクトルで, i 番目のクラスタのクラスタ中心を表す. c はクラスタ数, n は入力データ全体の個体数を表す.

帰属度およびクラスタ中心の更新式は式 (3), (4) で表される.

$$u_{ki} = \begin{cases} 1 & (i = \arg \min_{1 \leq j \leq c} \|\mathbf{x}_k - \mathbf{v}_j\|^2) \\ 0 & (\text{otherwise}) \end{cases} \quad (3)$$

$$\mathbf{v}_i = \frac{\sum_{k=1}^n u_{ki} \mathbf{x}_k}{\sum_{k=1}^n u_{ki}} \quad (4)$$

$u_{ki} \in \{0, 1\}$ であることは, 分割がハードであることを意味し, $u_{ki} = 1$ のとき, クラスタ i に属していることを表す. 以下に HCM のアルゴリズムを示す.

アルゴリズムの終了条件として, 目的関数あるいは各変数の収束, 最大繰り返し回数数の設定などがある.

Algorithm 1 Hard c-means

- HCM1 クラスタ数 c を定め、 V の初期値をランダムに設定する。
 HCM2 各データ \mathbf{x}_k を帰属度行列の更新式に従い各クラスタに割り当てる。
 HCM3 更新式に従いクラスタ中心 \mathbf{v}_i を更新。
 HCM4 終了条件を満たせば終了。そうでなければ Step2 に戻る。
-

2.2 妥当性基準

妥当性基準はこれまでに多数提案されているが、決定的なものについては、現在も議論は継続されている⁹⁾。本研究では Dunn's index⁶⁾, Davies-Bouldin's index⁵⁾, Xie-Beni index⁴⁾ の3つの妥当性基準を用いる。

2.2.1 Dunn's index

Dunn's index(DI) は式 (5) で表される。

$$DI = \min_{1 \leq i \leq c-1} \left\{ \min_{i+1 \leq j \leq c} \left\{ \frac{\text{dis}(G_i, G_j)}{\max_{1 \leq k \leq c} \text{dia}(G_l)} \right\} \right\} \quad (5)$$

また、 dia は各クラスタのまとまり度合いを指し、 dis はクラスタ間の分離度合いを表す指標であり、式 (6), (7) で定義される。 G_i, G_j, G_l は i, j, l 番目のクラスタを表す。

$$\text{dia}(G_l) = \max_{i, j \in G_l} d(\mathbf{x}_i, \mathbf{x}_j) \quad (6)$$

$$\text{dis}(G_i, G_j) = \min_{\mathbf{x} \in G_i, \mathbf{y} \in G_j} d(\mathbf{x}, \mathbf{y}) \quad (7)$$

\mathbf{x}, \mathbf{y} はそれぞれデータを表し、 G_i, G_j はクラスタを表す。この指標は大きい値であればあるほど良い分割結果であると判断される。式 (6) でクラスタ内で非類似度が最大のデータ間の組み合わせの値を求め、式 (7) でクラスタ間で非類似度が最小のデータ間の組み合わせの値を求める。そのため、クラスタ内データの凝集度が高く、クラスタ間の分離が良い場合に DI の値は大きくなり、良い結果になる。

2.2.2 Davies-Bouldin's index

Davies-Bouldin's index(DB) は以下の4式で構成される。

$$DB = \frac{1}{c} \sum_{i=1}^c R_i \quad (8)$$

$$R_i = \max_{j=j \dots c, i \neq j} (R_{ij}), i = 1 \dots c \quad (9)$$

$$R_{ij} = \frac{S_i + S_j}{\|\mathbf{v}_i - \mathbf{v}_j\|^2} \quad (10)$$

$$S_i = \frac{1}{|G_i|} \sum_{\mathbf{x} \in G_i} \|\mathbf{x} - \mathbf{v}_i\|^2 \quad (11)$$

式 (11) の S_i はクラスタ内分散度、式 (10) の R_{ij} はクラスタ間類似度を表している。式 (9) の R_i は式 (10) の条件式であり、ある i を固定した際、 i 以外のクラスタとのクラスタ間類似度の最大値を求める事を意味す

る。式 (9), (10), (11) を計算することで、式 (8) より、DB の値が求まる。この指標は小さい値を取れば取るほどよい結果である。

DB の式 (10) でクラスタ間類似度を計算する際、 $\|\mathbf{v}_i - \mathbf{v}_j\|^2$ でクラスタ中心間の非類似度を計算し、 $S_i + S_j$ でクラスタ内分散度の和を計算する。そのため、分割結果のクラスタ内のデータがクラスタ中心周辺に密集し、クラスタ間の分離が良ければ DB の値は小さくなり、良い結果になる。

2.2.3 Xie-Beni's index

Xie-Beni(XB) は式 (12) で表される。

$$XB = \frac{\sum_{k=1}^n \sum_{i=1}^c u_{ki} \|\mathbf{x}_k - \mathbf{v}_i\|^2}{n \min_{\substack{1 \leq i, j \leq c \\ i \neq j}} \|\mathbf{v}_i - \mathbf{v}_j\|^2} \quad (12)$$

$\sum_{k=1}^n \sum_{i=1}^c u_{ki} \|\mathbf{x}_k - \mathbf{v}_i\|^2 / n$ は全てのクラスタについて、クラスタ中心とクラスタ内のデータとの非類似度の平均値を意味し、クラスタ内の凝集度を表す。また $\min_{1 \leq i, j \leq c, i \neq j} \|\mathbf{v}_i - \mathbf{v}_j\|^2$ は各クラスタ中心間の非類似度の最小値を意味し、クラスタ間分離度を表す。XB ではクラスタ中心とクラスタ内データとの非類似度の平均が小さく、クラスタ間類似度が大きい場合 XB の値が小さくなり、良い値となる。

2.2.4 数値実験に用いるデータセット

本章では本稿で用いる数値データを示す。その後 Polaris データを HCM で分割し、その結果を各妥当性基準を用いて評価した結果を図示する。

本稿で使用するデータセットを Table. 1 に示す。

Table 1: Sets of objects.

データ名	個体数 (n)	次元 (p)	クラスタ数 (c)
人工データ 1	120	2	3
人工データ 2	125	2	5
Polaris データ	51	2	3
Iris データ	150	4	3

Iris データ¹²⁾ 以外のデータセットの散布図を Fig. 1, 2, 3 に示す。同一色、同一形の個体は同一のクラスタであることを意味する。

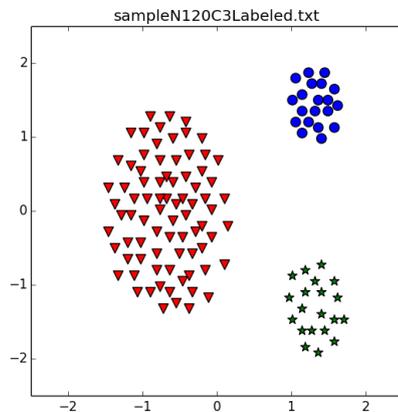


Fig. 1: Artificial data 1.

人工データ 1(Fig. 1) は、最も大きな集合が個体数

80, それ以外の集合が個体数各 20 のデータセットである。

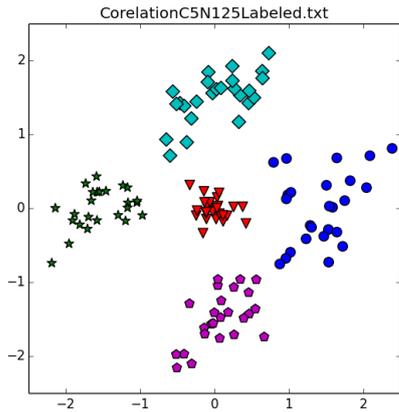


Fig. 2: Artificial data 2.

人工データ 2(Fig. 2) は, μ を平均, σ を分散, j を個体数, $\mu = [0, 0]$ 以外の集合の相関係数を 0.5 とし, 正規分布に従いランダムに生成した人工データである⁸⁾. 式で表したものが以下である.

$$\begin{aligned}
 x_j &\sim N(\mu = [0, 0], \sigma = [0.2, 0.2]), (j = 1, \dots, 25) \\
 x_j &\sim N(\mu = [-2, 0], \sigma = [0.3, 0.3]), (j = 26, \dots, 50) \\
 x_j &\sim N(\mu = [2, 0], \sigma = [0.3, 0.3]), (j = 51, \dots, 75) \\
 x_j &\sim N(\mu = [0, 2], \sigma = [0.4, 0.4]), (j = 76, \dots, 100) \\
 x_j &\sim N(\mu = [0, -2], \sigma = [0.4, 0.4]), (j = 101, \dots, 125)
 \end{aligned}$$

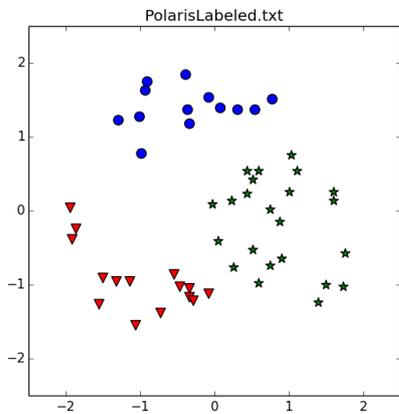


Fig. 3: Polaris data.

Polaris データは北極星付近の星の位置をプロットしたデータである.

Iris データは UCI Machine Learning の HP で公開されているベンチマークデータのの一つである¹²⁾.

2.2.5 妥当性基準を用いた数値例

Polaris データに対し HCM を $c = 2 \sim 6$ で初期値を変え 1000 回を実行し, 各妥当性基準を用いて評価した出力値の平均と分散を Fig. 4, 5, 6 に示す. 横軸の数値が分割数を意味し, グラフ中の「×」がその分割数での平均値を表す. 「×」を中心とした上下のバーが標準偏差を表す.

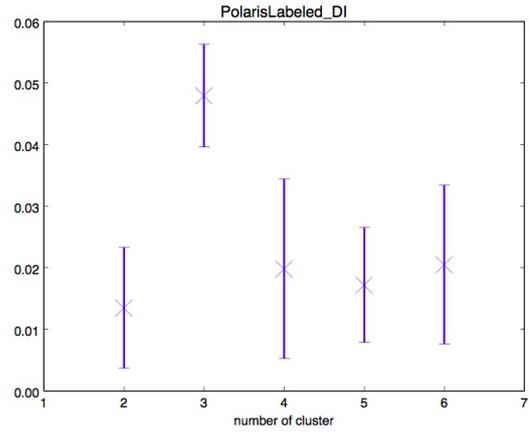


Fig. 4: Result of average and standard deviation by HCM and DI with Polaris data.

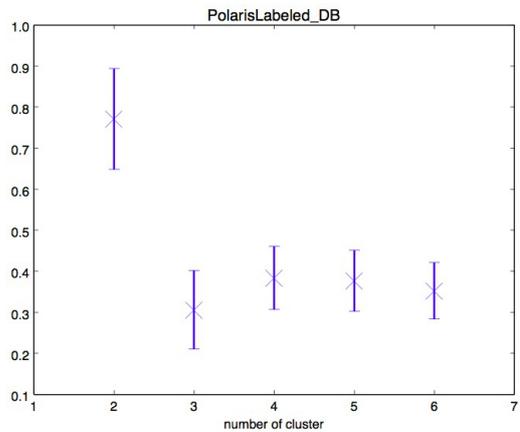


Fig. 5: Result of average and standard deviation by HCM and DB with Polaris data.

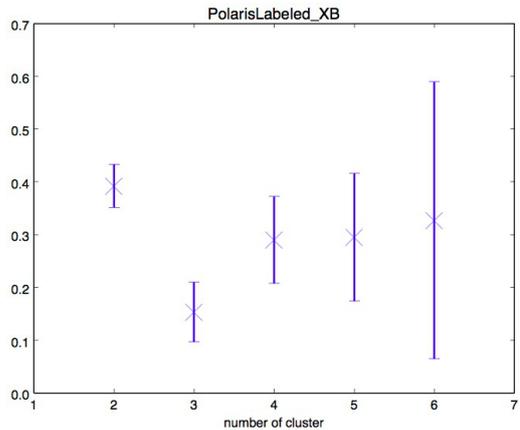


Fig. 6: Result of average and standard deviation by HCM and XB with Polaris data.

Figs. 4, 5, 6 より, Polaris データの分割結果に対しては全ての指標が最適な分割数 3 を最も良い平均値として出力した.

2.3 x-means

HCM をはじめとする従来の非階層的クラスタリング手法に関する問題点として, 以下が知られている:

1. 巨大な規模のデータでは計算時間が長くなる

2. クラスタ数をユーザが指定しなければならない
3. 局所最適解を採用しがちである

従来の x -means は特に 1, 2 の解決を目的に提案されている⁷⁾。 x -means のアルゴリズムは最初に十分に小さな分割数 (デフォルトで $c = 2$) で HCM を行った分割結果を情報量基準を用いて評価し, 分割した各クラスタに対して, HCM を再帰的に実行するという構成になっている。分割の可否および, 処理の継続を情報量基準を用いて評価する。

この処理を行うことで,

- 分割対象の個体が繰り返しごとに減少する
- クラスタ数を事前に設定する必要がない

というメリットが生じ, 1, 2 の問題を解消することが可能になる。従来手法では情報量基準として Bayesian Information Criteria(BIC)¹⁰⁾ や Akaike's Information Criteria(AIC)¹¹⁾ が評価指標として用いられている。

従来の x -means のアルゴリズムを以下に示す。

Algorithm 2 x -means⁷⁾

- Step1 c を十分に小さなクラスタ数 (デフォルトで $c = 2$) とし, HCM を実行。その分類結果に対し, 情報量基準の値を計算する。
- Step2 分割後の各クラスタに対し, $c = 2$ として HCM を実行。各クラスタの情報量基準の値を計算する。
- Step3 分割前と分割後の情報量基準の値を比較する。分割後の情報量基準の値が分割前の値より悪ければ終了。そうでなければ Step2 に戻る。
-

3 提案手法

従来の x -means は大規模・低次元のデータに対しての有用性が述べられている。しかし, 情報量基準は対数尤度を用いるため確率モデルの推定を行わなければならない。特に高次元データの場合, 計算に時間がかかる。そこで, 本研究ではその計算時間の長さを解消するために, 情報量基準の代替として妥当性基準を用いた x -means を提案する。

以下に本稿で提案する x -means のアルゴリズムを示す。

Algorithm 3 Cluster validity measure based x -means

- Step1 c を十分に小さなクラスタ数 (デフォルトで 2) とし, HCM を実行。その分類結果に対し, データ全体での妥当性基準の値を求める。
- Step2 分割後の各クラスタに対し, $c = 2$ として HCM を実行。分割後の状態に対してデータ全体での妥当性基準の値を求める。
- Step3 分割前と分割後の妥当性基準の値を比較する。分割後の妥当性基準の値が分割前の値より悪ければ終了。そうでなければ Step2 に戻る。
-

情報量基準は単一のクラスタに対しても評価可能な指標である。一方, 妥当性基準を用いた評価には, クラスタが複数必要である。妥当性基準は同一のデータセットに対して, 異なる分割数での指標の値を比較す

ることに意味があるため, 従来手法は再帰的に分割後のクラスタを評価するが, 提案手法ではデータセット全体で, 異なる分割数の状態の評価を行う。

4 数値例

本章の最初に, 提案手法の分割結果が正しいかどうかを外的基準を用いて評価するために使用する Rand Index¹³⁾ についての説明を行う。その後, 2.2.4 章で示した 4 つのデータに対して提案手法を実行し, 分割数と実行速度, Rand Index の値を示す。

入力データを平均 0, 分散 1 で正規化し, 提案手法を実行する。また, HCM におけるクラスタ中心の初期値選択はデータの中から重複無く c 個選択しその座標を初期値として用いる。

Table 3, 5, 7, 9 は各妥当性基準を x -means に適用し 1000 回実行した出力結果の分割数と実行時間を示した表である。一番左の列が用いた妥当性基準, 一番右の列が実行時間の合計, 2-5 の列は各列の分割数 (c) に分割された回数を表している。分割数「6 ≤」の列にはクラスタ数が 6 以上になった回数を示している。

Table 4, 6, 8, 10 は Rand Index の値を表した表である。一番左の列が用いた妥当性基準, 2-5 の列がその分割数 (c) の時の Rand Index の値を表している。分割数「6 ≤」の列には分割数が 6 以上になった時の Rand Index の値を示している。また, 各手法について Avg は各分割数の Rand Index の平均値, Std はその標準偏差を意味する。“-”を記録している部分は, 今回の実験では, 一度もその分割数を出力しなかった事を意味する。

4.1 Rand Index

Rand index は 2 つのグループの組み合わせの一致度を計算する評価する指標である。

Table 2: Combination table used in Rand Index.

U \ V	V ₁	V ₂	⋯	V _C	Sums
U ₁	n ₁₁	n ₁₂	⋯	n _{1C}	a ₁
U ₂	n ₂₁	n ₂₂	⋯	n _{2C}	a ₂
⋮	⋮	⋮	⋮	⋮	⋮
U _R	n _{R1}	n _{R2}	⋯	n _{RC}	a _R
Sums	b ₁	b ₂	⋯	b _C	∑ _{ij} n _{ij} = n

Rand Index はこのテーブルに従い 2 つの分割結果の組み合わせを計算する。本研究では, 2 つの分割結果の内の片方にラベルデータを用いるため, 以降の説明でも片方のグループをラベルデータとして説明する。Rand Index を計算するためには N_{11} , N_{10} , N_{01} , N_{00} を Table. 2 から求めなければならない。

- N_{11} : 同じラベル番号を持つデータが同じクラスタに分割されたデータ間の組み合わせの数。Table. 2 では n_{11}, \dots, n_{RC} の組み合わせの総和。
- N_{01} : 同じラベル番号を持つデータが, 分割後異なるクラスタに分割されたデータ間の組み合わせの数。Table. 2 では a_1, \dots, a_R の組み合わせの総和。
- N_{10} : 異なるラベル番号を持つデータが, 分割後同じクラスタに分割されたデータ間の組み合わせの数。Table. 2 では b_1, \dots, b_C の組み合わせの総和。

- N_{00} : ラベルが異なり、異なるクラスに分類されたデータ間の組み合わせの総和。実際には全てのデータの組み合わせから N_{11} , N_{10} , N_{01} を減算し求める。

そしてこの値を

$$RI = \frac{N_{00} + N_{11}}{NC_2} \quad (13)$$

として計算することで指標の値が求まる。この指標は1を取るとき最大であり、全てのデータが正しく分割されたことを意味する。反対に0を取る時最小であり、全てのデータが誤って分割されたことを意味する。

4.2 人工データ 1

各妥当性基準を x -means に適用し、人工データ 1 に対して実行した場合の分割数、実実行時間を Table. 3, Rand Index の評価結果を Table. 4 にそれぞれ示す。

Table 3: Results of obtained number of clusters and run time by proposed method with artificial data 1.

c	2	3	4	5	6 ≤	time(sec)
DI	0	764	212	12	12	1491.78
DB	1	455	522	14	8	139.7
XB	0	722	278	0	0	127.62

4.3 人工データ 2

各妥当性基準を x -means に適用し、人工データ 2 に対して実行した場合の分割数、実実行時間を Table. 5, Rand Index の評価結果を Table. 6 にそれぞれ示す。

Table 5: Results of obtained number of clusters and run time by proposed method with artificial data 2.

c	2	3	4	5	6 ≤	time(sec)
DI	2	515	349	99	35	1806.06
DB	0	5	247	334	414	198.99
XB	0	6	573	421	0	177.16

4.4 Polaris データ

各妥当性基準を x -means に適用し、Polaris データに対して実行した場合の分割数、実実行時間を Table. 7, Rand Index の評価結果を Table. 8 にそれぞれ示す。

Table 7: Results of obtained number of clusters and run time by proposed method with Polaris data.

c	2	3	4	5	6 ≤	time(sec)
DI	246	453	195	94	12	289.27
DB	0	8	36	171	785	92.92
XB	0	748	183	50	19	52.66

4.5 Iris データ

各妥当性基準を x -means に適用し、Iris データに対して実行した場合の実実行時間を Table. 9, Rand Index の評価結果を Table. 10 にそれぞれ示す。

4.6 各指標で見られた特徴

x -means の評価指標として、DI, DB, XB を用いた際、出力結果に見られた特徴について述べる。

はじめに Dunn's index について述べる。この指標に見られた特徴は以下の通りである。

Table 9: Results of obtained number of clusters and run time by proposed method with Iris data.

c	2	3	4	5	6 ≤	time(sec)
DI	998	1	1	0	0	1347.5
DB	995	0	0	2	3	130.59
XB	996	2	0	2	0	130.03

1. 処理速度が3つの指標の中で最も遅い
2. 分割数が少なくなる場合が多い

1. は式 (5) の分子で $\mathcal{O}(n^2)$ の計算が毎回必要になるため、処理に時間がかかる。2. はクラスタ間類似度を計算する式 (7) で全てのクラスタ間の中で、最も近いデータ同士を使用するため、特定のデータに対する依存度が高くなる。そのため、分離が悪いデータセットや、ノイズが含まれるデータセットを扱う場合、最も悪い組み合わせの部分が常に採用され、式 (6) が改善されない限り分割が継続されないため、分割数が少なくなると考えられる。

次に Davies-Bouldin's index この指標に見られた特徴は以下のとおりである。

1. 処理速度が DI に比べ高速である
2. 分割数が多くなる場合が多い
3. 最適な分割数でも正しく分類されていない場合がある

1. は計算量が $\mathcal{O}(n)$ であるためである。各クラスタで、クラスタに含まれるデータ全てと、クラスタ中心との非類似度の計算のみを行うため、DI より高速になる。2. は分離が悪いデータセットの場合、1つのクラスタが持つ個体数が極端に少なくなる場合がある。その場合、式 (11) の値がごく少数のデータに依存し、分割が最適であると評価されてしまうことがあり、分割が継続され分割数が増える傾向にある。3. は2. とは反対に、式 (11) を用いるため、1つのクラスタに多くのデータが含まれた分割が行われた場合、良い結果だと評価するのに十分な数のデータがクラスタ中心周辺に存在すれば、少数の誤った分割のデータは無視され分割結果が採用される。そのため、誤った分類が許容され、Rand Index の分散値が高くなり、また平均値も他の指標と比べ低くなる場合が多くなった。

最後に、Xie-Beni's index について述べる。この指標に見られた特徴は以下のとおりである。

1. 処理速度が3つの指標の中で最も高速である
2. 最適な分割数に近い分割数に分割が集中する

1. は DB と同様、式 (12) の分母の計算量は $\mathcal{O}(n)$ である。それにもかかわらず、DB と実行速度に差が存在するのは、プログラムの実装方法や計算量として表されていない部分の計算時間であると考えられる。2. については式 (12) は各データと、各クラスタ中心との非類似度の平均を求めているが、分割がハードである場合、分子は HCM の目的関数そのものである。HCM の目的関数は最小を目指す関数である。そのため、最適な分割数の時、あるいは最適な分割数に近い状態のとき値が最小に近づき、良い分割数に近いクラスタ数に分割が集中する。

Table 4: Results of average and standard deviation of Rand Index with artificial data 1.

c	2	3	4	5	$6 \leq$
DI(Avg \pm Std)	-	0.939 \pm 0.119	0.811 \pm 0.051	0.762 \pm 0.006	0.755 \pm 0.003
DB(Avg \pm Std)	0.776 \pm 0.000	0.988 \pm 0.035	0.790 \pm 0.025	0.870 \pm 0.015	0.876 \pm 0.014
XB(Avg \pm Std)	-	0.974 \pm 0.059	0.784 \pm 0.008	-	-

Table 6: Results of average and standard deviation of Rand Index with artificial data 2.

c	2	3	4	5	$6 \leq$
DI(Avg \pm Std)	0.613 \pm 0.054	0.792 \pm 0.024	0.893 \pm 0.014	0.944 \pm 0.013	0.935 \pm 0.009
DB(AVG \pm Std)	-	0.735 \pm 0.064	0.890 \pm 0.019	0.942 \pm 0.036	0.904 \pm 0.042
XB(Avg \pm Std)	-	0.701 \pm 0.064	0.893 \pm 0.012	0.962 \pm 0.014	-

Table 8: Results of average and standard deviation of Rand Index with Polaris data.

c	2	3	4	5	$6 \leq$
DI(Avg \pm Std)	0.703 \pm 0.048	0.869 \pm 0.075	0.914 \pm 0.047	0.860 \pm 0.056	0.823 \pm 0.038
DB(Avg \pm Std)	-	0.794 \pm 0.051	0.830 \pm 0.075	0.863 \pm 0.079	0.823 \pm 0.064
XB(Avg \pm Std)	-	0.870 \pm 0.082	0.888 \pm 0.073	0.818 \pm 0.029	0.812 \pm 0.029

Table 10: Results of average and standard deviation of Rand Index with Iris data.

c	2	3	4	5	$6 \leq$
DI(Avg \pm Std)	0.776 \pm 0.000	0.634 \pm 0.000	0.757 \pm 0.000	-	-
DB(Avg \pm Std)	0.776 \pm 0.000	-	-	0.738 \pm 0.003	0.723 \pm 0.004
XB(Avg \pm Std)	0.776 \pm 0.000	0.659 \pm 0.000	-	0.754 \pm 0.001	-

4.7 考察

本稿の数値実験では人工データ 1, 人工データ 2, Polaris データに対しては, XB を x -means に用いた場合が最も良い結果を出力した. しかし, Iris データに対してはどの指標を用いた場合も良い結果を出力することはほとんどなかった. Iris データは 3 つのクラスターのうちの, 2 つのクラスターが近接し, 類似性が高く評価されてしまう. そのため, 実際にはクラスター数 3 が本来の分割数であるにもかかわらず, クラスター数 2 が最も良いという出力結果が得られた.

5 おわりに

本稿では HCM, 妥当性基準, 従来の x -means について説明した. その後, 妥当性基準を用いた x -means を提案し, 数値例によりその有効性を示した.

今後の課題として計算時間の削減, ノイズに対するロバスト性の実現のため一般化 Dunn's index¹⁴⁾を用いた検証がある. また, KL 情報量正規化¹⁵⁾, カーネル法¹⁶⁾を用いて, より複雑な分類境界を持つデータセットへの拡張に取り組む.

参考文献

- 1) A. K. Jain : Data clustering: 50 years beyond K-means, Pattern Recognition Letters, Vol. 31, 651/666 (2010)
- 2) S. Miyamoto, H. Ichihashi, K. Honda : Algorithms for Fuzzy Clustering, Springer, Vol. 229, (2008)
- 3) J. C. Bezdek : Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum Press, (1981)
- 4) X. L. Xie, G. Beni : A Validity Measure for Fuzzy Clustering, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 13, 841/847 (1991)
- 5) D. L. Davies, D. W. Bouldin : Cluster Separation Measure, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 1, No. 2, 95/104 (1979)
- 6) J. C. Dunn : Well separated clusters and optimal fuzzy partitions, Journal of Cybernetics, Vol. 6, 95/104 (1978)
- 7) D. Pelleg, A. Moore : X-means: Extending K-means with Efficient Estimation of the Number of Clusters, ICML '00 Proceedings of the Seventeenth International Conference on Machine Learning, 727/734 (2000)
- 8) T. Ishioka : An Expansion Of X-means For Automatically Determining The Optimal Number Of Clusters - Progressive Iterations Of K-means And Merging Of The Clusters -, The IASTED International Conference on Computational Intelligence, Calgary Canada, 91/96 (2005)
- 9) 中村哲也, 橋本済, 宮本定明 : 様々なクラスター妥当性関数の比較 : カーネル化と直接最適化, 第 23 回ファジィシステムシンポジウム, 276/279 (2007)
- 10) G. Schwarz : Estimating the Dimension of a Model, Annales. Statist., Vol. 6, 461/464 (1978)
- 11) H. Akaike : A new look at the statistical model identification, IEEE Transactions on Automatic Control, Vol. 19, 716/723 (1974)
- 12) UCI Machine Learning Repository : <http://archive.ics.uci.edu/ml/>
- 13) L. Hubert, P. Arabie : Comparing Partitions, Journal of Classification, Vol. 2, 193/218 (1985)
- 14) J. C. Bezdek, N. R. Pal : Some New Indexes of Cluster Validity, IEEE Transactions on system, man, and cybernetics-Part B: Cybernetics, Vol. 28, 301/3015 (1998)
- 15) 宮岸 聖高, 市橋 秀友, 本多 克宏 : K-L 情報量正規化 FCM クラスタリング法, 日本ファジィ学会誌, Vol. 13, 406/417 (2001)
- 16) 赤穂 昭太郎 : カーネル多変量解析 - 非線形データ解析の新しい展開, 岩波書店

強化学習における *word2vec* を用いた転移学習

○武石真登 恒川裕章 鈴木拓央 濱上知樹 (横浜国立大学)

Transfer Learning for Reinforcement Learning by using *word2vec*

*M. Takeishi, H. Tsunekawa, T. Suzuki, and T. Hamagami (Yokohama National University)

Abstract— We propose a new transfer learning method for reinforcement learning using a semantics space which is trained by *word2vec*. To improve the efficiency of reinforcement learning, a lot of transfer learning methods are studied. The basic idea of transfer learning is to recall the same action among similar states. However, the method needs to define the similarity of state by heuristics. In order to overcome the limitation, we propose a new learning method which is able to estimate similarity by using context which can be evaluate semantic distance. Experimental results show that the method can construct the semantic space having similarity of states automatically. Moreover the results show that the method would improve other reinforcement learning algorithm.

Key Words: reinforcement learning, transfer learning, *word2vec*

1 はじめに

強化学習は、ロボットの自律移動など、未知環境で適切な動作が求められる用途に適した学習手法として広く研究が行われている。しかし、適切な行動獲得のために探索を多く必要とするため、学習に時間がかかるという課題がある。そのため、大規模な状態空間を扱う場合、エージェントに一から強化学習を行わせるのは現実的ではない。そこで探索を減らすために、事前知識を利用する学習法である転移学習を、強化学習に用いる研究が進められている¹⁾。

強化学習における転移学習法では、多くの場合事前タスクの状態空間で学習された行動を、目的タスクの状態空間に写像することにより、目的タスクで事前知識の利用を可能にしている。Reinaldoらは事前タスクから目的タスクへの写像をヒューリスティクスを用いて定めるQ-learningのアルゴリズムを提案している²⁾。また、筆者らは状態空間の位相類似性を用いて、学習済みの行動価値を未学習の価値空間へ写像する手法³⁾を提案した。これらの手法は、目的タスクの経験を事前タスクのどの状態に利用できるか明らかな場合を想定している。すなわち、状態間の類似性が経験的に明らかである場合に有効である。しかし、一般には状態間の類似性は明らかでない。よって、状態の類似性を自律的に獲得可能な手法が望ましい。

そこで本稿では、文章中に現れる単語の共起関係から、単語の類似性を学習可能な*word2vec*⁴⁾を用いて、エージェントの状態遷移から状態の類似性を学習可能なアルゴリズムを提案する。

2 状態の位相類似性を利用した転移学習

状態の位相類似性を利用した転移学習は、学習すべき状態空間 S から位相類似性を評価可能な状態空間 C へ式 (1) を用いて写像し、空間 C 上で転移学習を行う手法である。

$$C = F(S) \quad (1)$$

ただし、 F は $S \mapsto C$ の写像関数である。空間 C 上で行動を学習すべき部分空間 $C_a \subset C$ が存在するとき、式 (2) を用いて求められる空間 $C' \subset C$ に対して同時に行

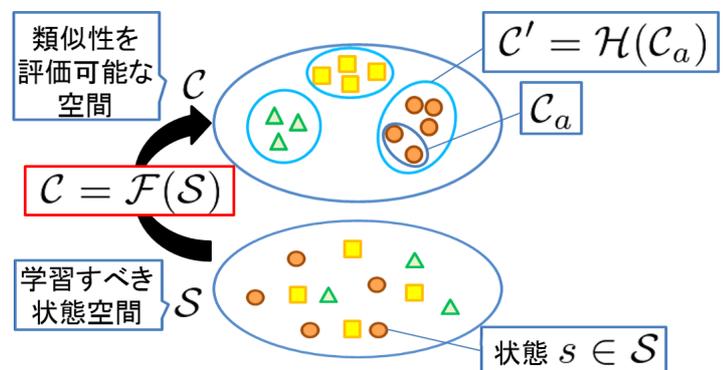


Fig. 1: 類似の位相構造を利用した転移学習

動を学習、すなわち C' に対して転移学習を行うことにより、学習の高速化を行う。

$$C' = H(C_a) \quad (2)$$

ただし、 H は空間 C 上で位相類似性を定義する関数である。Fig.1 に類似の位相構造を用いた転移学習の様子について示す。

写像関数 $C = F(S)$ が経験的に明らかなタスクにおいて、状態の位相類似性を用いた転移学習で強化学習の高速化が可能であることが確認された³⁾。しかし、一般に写像関数 F は明らかでないため、自律的に F を獲得可能なアルゴリズムが望ましい。

3 文脈からの位相類似性の学習

近年、文中に現れる単語の共起関係を用いることで、ニューラルネットワークを用いた教師なし学習で、単語の意味を学習可能であることが明らかになった⁴⁾。特に *word2vec* では、単語間の意味の近さをベクトル空間の位相類似性により表現することができる。よって、*word2vec* は単語を、位相類似性の評価可能な空間へ写像可能といえる。

この知見から本提案手法では、強化学習において、状態遷移中に現れる時系列の観測や行動の共起関係を *word2vec* で学習し、状態空間から意味空間への写像関数 $C = F(S)$ を学習する。

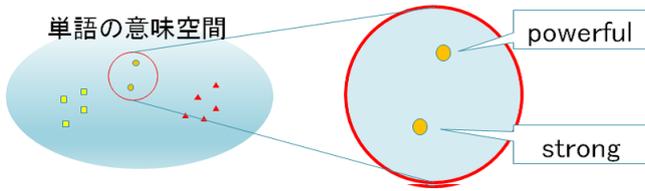


Fig. 2: 単語の意味空間

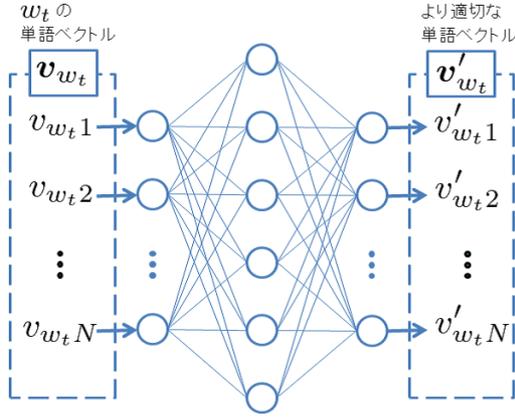


Fig. 3: 単語のベクトル学習

4 skip-gram モデル

`word2vec` では、skip-gram モデル⁴⁾ を用いることにより、単語の意味を学習する。

4.1 モデルの特徴

skip-gram モデルは、文章中における単語の共起関係、すなわち文脈を学習することにより単語を高次のベクトル空間へ写像する、教師なし学習のモデルである。Fig.2 に、skip-gram モデルを用いて単語の意味を学習した例を示す。skip-gram を用いると、Fig.2 のように、“powerful”、“strong”といった意味が近い単語ほど意味空間上での位相関係が近くなるような空間が作成される。

4.2 アルゴリズム

skip-gram モデルでは、コーパス中に現れた単語の意味が周辺の単語の共起関係により決定することを利用し、Fig.3 のように単語のベクトル表現をニューラルネットワークで学習する。今、コーパス中の t 番目に現れる単語を w_t 、それをベクトル表現したものを \mathbf{v}_{w_t} 、コーパスを w_1, w_2, \dots, w_T とすると、単語学習の目的関数は、式 (3) で表される。つまり、単語のベクトル \mathbf{v}_{w_t} が周辺の単語 $w_{t-C}, \dots, w_{t+C} \mid C \neq t$ のベクトル $\mathbf{v}_{w_{t-C}}, \dots, \mathbf{v}_{w_{t+C}}$ と距離が近くなるように、また、周辺の単語以外とは、なるだけベクトル間距離が遠くなるようにニューラルネットワークの重みを調整し、各単語に対する適切なベクトルを取得する。

$$\begin{aligned} \text{maximize } & \frac{1}{T} \sum_{t=1}^T \sum_{-C \leq j \leq C, j \neq 0} \log p(w_{t+j} | w_t) \quad (3) \\ p(w_0 | w_I) = & \frac{\exp(\mathbf{v}_{w_0}^T \mathbf{v}_{w_I})}{\sum_{w=1}^W \exp(\mathbf{v}_w^T \mathbf{v}_{w_I})} \end{aligned}$$

ただし、 \mathbf{v}_w はコーパス中に現れた単語のベクトルのうち w 番目に辞書登録された単語、 W は辞書に登録された単語数である。学習初期ではランダムに設定された \mathbf{v}_{w_t} を式 (3) を用いて学習することにより、文脈を考慮した単語の意味ベクトルを教師なしで獲得することができる。

5 skip-gram モデルを用いた写像関数の作成

提案手法では、単語が変わって状態の共起関係を skip-gram モデルを用いて学習する。これにより、状態を類似性を評価可能な空間 \mathcal{C} への写像関数する関数 \mathcal{F} を獲得する。

5.1 特徴空間の導入

事前タスクにおいて skip-gram モデルで状態の意味を学習しても、強化学習は未知環境において行動を獲得するアルゴリズムであることから、事前タスクで観測された状態が目的タスクのどの状態に対応するのか明らかでない。そこで、本提案手法では状態とは別に特徴 $m \in M$ を定義する。特徴 $m \in M$ とは目的タスクと事前タスクの状態において共通して観測可能な系列の空間である。そして、特徴 m を用いて状態の意味を推定する。

5.2 事前タスクにおける意味空間の作成

提案手法では、事前タスクで時刻 t において観測される特徴 m_t の意味ベクトルを、周辺の時刻 $t - C, t - C + 1, \dots, t + C \mid C \neq 0$ で観測された特徴 m_{t-C}, \dots, m_{t+C} の共起関係を利用し、skip-gram モデルで学習する。そして意味ベクトルを学習後、目的タスクにおいてある状態 s'_t に遷移したときに観測される特徴 m'_t の意味ベクトルを、 s'_t の意味ベクトルとして利用する。すなわち、 $C = \mathcal{F}(S)$ の代わりに、状態の特徴空間から意味空間への写像 $\mathcal{C} = \mathcal{F}(M)$ を用いることで、状態の意味を推定する。

6 行動の学習アルゴリズム

提案する強化学習では、目的タスクにおいて以下の3つの項目についてどのように行動すべきかを学習する。

- 遷移した状態 s_x
- 状態 s_x において観測された特徴 m_x
- 特徴 m_x と意味が類似している特徴 m_y

通常の強化学習では、各状態 $s \in S$ においてどのように行動を選択するかを学習するが、提案手法では各特徴 $m \in M$ が観測されたときに、どのように行動を選択するかを並行して学習する。さらに、事前学習により獲得された特徴空間中で、意味が類似している特徴に関しても同時に学習を行う。

提案手法では、観測された特徴 m_x が特徴空間 $M_a \subset M$ の要素、すなわち $m_x \in M_a$ であるとき、事前学習により獲得した特徴空間から意味空間への写像 $\mathcal{C}_a = \mathcal{F}(M_a)$ を用いて観測された特徴の意味を求め、その近傍の空間 $\mathcal{C}' = \mathcal{H}(\mathcal{C}_a)$ に含まれる $\mathcal{M}' = \mathcal{F}^{-1}(\mathcal{C}_a)$ の特徴空間の学習を行うことで、転移学習を行う。

6.1 更新式

強化学習の一般的な手法である Profit Sharing⁵⁾ では、エピソード終了時に各状態に対する行動の評価値 $w(s, a)$ を評価関数 f を用いて式 (4) で更新する。

$$w(s_x, a_x) \leftarrow w(s_x, a_x) + f(x) \quad (4)$$

$$f(x) = \begin{cases} r_s & (x = W - 1) \\ \frac{1}{S} f(x+1) & (x = W - 2, \dots, 0) \end{cases}$$

ただし、 W は1エピソードにかかったステップ数、 S は $S > 1$ の定数である。この更新式に加えて、提案手法では、式 (5) に示す特徴 $m \in M$ の学習を追加する。

$$w_m(m_x, a_x) \leftarrow w_m(m_x, a_x) + f_m(x) \quad (5)$$

$$f_m(x) = \begin{cases} r_m & (x = W - 1) \\ \frac{1}{S} f_m(x+1) & (x = W - 2, \dots, 0) \end{cases}$$

ただし、 w_m は特徴空間における行動の評価値、 f_m は特徴空間における評価関数である。さらに、skip-gram モデルを用いて求めた特徴の意味ベクトル \mathbf{v}_m を用いて、意味空間 C 上で m と m' が近傍に存在する場合、すなわちベクトル間距離 $\Delta = \mathbf{v}'_m - \mathbf{v}_m$ が Δ_{\max} よりも小さい場合に m' の評価値についても式 (6) を用いて更新を行う。

$$w_m(m'_x, a_x) \leftarrow w_m(m'_x, a_x) + f_m(x) \quad (6)$$

$$(\forall m'_x \mid \|\mathbf{v}'_{m'_x} - \mathbf{v}_{m_x}\| \leq \Delta_{\max})$$

6.2 行動選択

提案手法では、行動選択時に時刻 t における状態 s_t の評価値 $w(s_t, a)$ に加えて、時刻 t において観測される特徴 m_t の評価値 $w_m(m_t, a)$ を用いて行動選択を行う。 ϵ -greedy 方策を用いて行動選択を行った例を以下に示す。

$$\pi(s_t, a_t) = \begin{cases} 1 - \epsilon & (a_t = \max_a w'(s_t, a)) \\ \epsilon & (\text{otherwise}) \end{cases} \quad (7)$$

$$w'(s_t, a) = w(s_t, a) + w_m(m_t, a)$$

式 (7) を用いて行動選択を行うことにより、特徴空間と状態空間の双方を考慮した行動選択が可能になる。

7 シミュレーション実験

シミュレーション実験により、本提案手法の有効性を明らかにする。以下の2項目を確認するために、迷路タスク、ネットワークタスクの2つのタスクで実験を行った。

- skip-gram モデルを用いて、特徴の系列から適切な特徴の意味空間を作成可能かどうかの検証 (迷路タスク)
- 提案手法を用いた場合の学習速度の検証 (ネットワークタスク)

7.1 迷路タスク：意味空間の検証

skip-gram モデルを用いて、エージェントが観測可能な特徴系列 m_0, m_1, \dots から、適切な特徴の意味空間が作成可能かどうかを Fig.4 の迷路環境を用いて検証する。

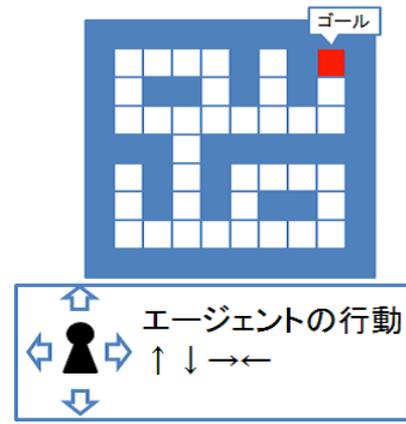


Fig. 4: 迷路環境

7.1.1 実験設定

エージェントは上下左右の行動を迷路上で行うことができる。また、状態は各マスの位置で定義される。まず、最適な行動を学習済みであるエージェントに。ランダムな初期位置からゴールまで行く行動を 10000 回実行させ、そこで観測された特徴系列 m_0, m_1, \dots を取得した。ただし今回の実験では、時刻 t における特徴を、5 ステップ前までに行った行動の履歴 $m_t = (a_{t-1}, a_{t-2}, a_{t-3}, a_{t-4}, a_{t-5})$ により定義した。次に、skip-gram モデルでそれぞれの特徴 m_0, m_1, \dots の意味ベクトル $\mathbf{v}_{m_0}, \mathbf{v}_{m_1}, \dots$ を作成する。実験パラメータを Table.1 に示す。

Table 1: 迷路環境における実験パラメータ

実験パラメータ	値
エージェントの行ったエピソード数	10000
意味ベクトルの次元数	100
ニューラルネットワークの中間層の数	100

作成された意味ベクトルの空間において、意味の近いベクトルほど空間上近い位置に配置されるかどうかを確認するために、SOM⁶⁾ を用いて描画を行った。

7.1.2 実験結果、考察

得られた SOM の空間を Fig.5 に示す。SOM の空間から、ベクトル間距離が近い特徴間ではエージェントが行うべき行動が一致している。例えば、Fig.5 において、迷路と空間の対応に着目すると、距離の近い4つの領域の表わす迷路上の状態(特徴)では、どれも、上に行く行動が最適な行動になっている。よって、意味空間上でベクトル間距離の近い特徴間で転移学習を行えば、学習速度を向上させることが可能である。

7.2 ネットワークタスク：学習速度の検証

本タスクでは、Fig.6 の環境において、スタートからゴールまで行く最短経路を求めるタスクを用いて、提案手法の学習速度について検証を行う。

7.2.1 実験設定

Fig.6 の環境では、3本のリンクを持ったノード 600 個が連結されており、エージェントは連結されたノード間をリンクを通して行き来できる。また、ノードには個別に固有の id と 4 種類のノードタイプが割り当て

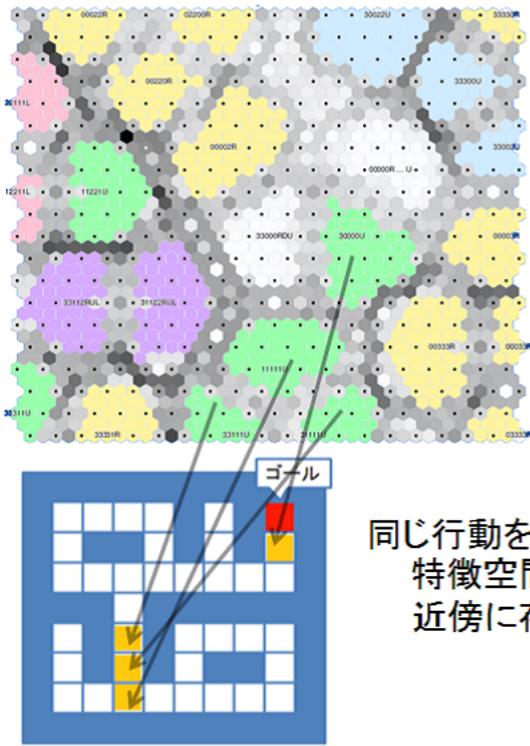


Fig. 5: skip-gram モデルで作成された意味空間

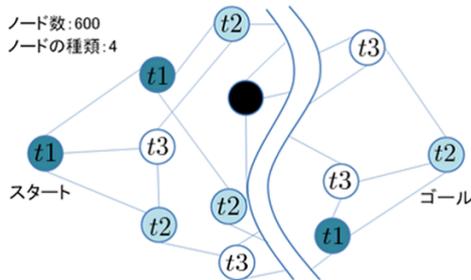


Fig. 6: 事前学習におけるタスク

られており、エージェントは観測を通し、id とノードタイプを知ることができる。

まず、Fig.6 の環境において最適な行動を行うことが可能なエージェントに、ランダムな初期位置からゴールノードまで行く行動 10000 エピソード動作させ、そこで観測される特徴系列を $m_0, m_1 \dots$ を取得した。ただし、今回の実験では特徴を、5 ステップ前までに観測されたノードタイプの履歴、すなわち $m_t = (t_t, t_{t-1}, t_{t-2}, t_{t-3}, t_{t-4}, t_{t-5})$ と定義した。次に、得られた特徴の系列 $m_0, m_1 \dots$ を用いて skip-gram モデルで特徴の意味ベクトル v_m を学習し、特徴の意味空間を作成した。実験パラメータを Table.2 に示す。最後に、Fig.6 に示されるネットワークを 2 つ連結した 1200 個のノードから成る環境を作成し、Profit Sharing を用いた場合、Profit Sharing に提案手法を追加した場合の 2 つの場合について比較を行った。Fig.7 に実験で用いた環境を示す。

$L = 300$ エピソードを 1 学習として、100 学習の平均を実験結果とした。また、行動は ϵ -greedy 方策を用

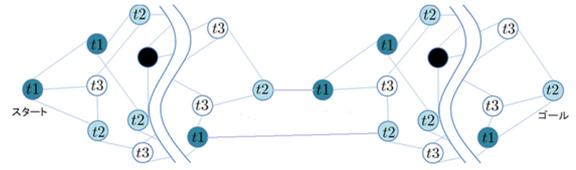


Fig. 7: 目的タスク

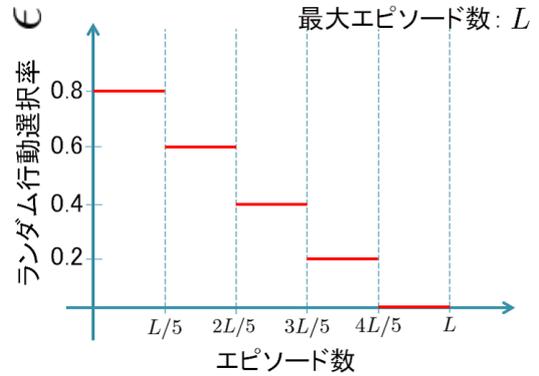


Fig. 8: スケジューリング

いて行動選択を行った。ただし、 ϵ は Fig.8 のようにスケジューリングした。また、今回は転移学習時に近傍の空間とみなす範囲を予備実験に基づき $\Delta_{\max} = 18$ とした。

事前学習においてネットワーク構造を考慮した意味空間が作成されているならば、本タスクでは既存手法と比較して少ないステップ数で適切な行動を獲得可能である。

Table 2: 事前タスクの実験パラメータ

実験パラメータ	値
エージェントが行ったエピソード数	10000
意味ベクトルの次元数	100
ニューラルネットワークの中間層の数	100
状態数	600
ノードの種類	4

Table 3: 目的タスクの実験パラメータ

実験パラメータ	値
学習率 α	0.3
S	4
ゴール達成時の報酬 r	1
ゴール達成時の特徴空間における報酬 r_m	1
Δ_{\max}	18

7.2.2 実験結果、考察

提案手法と、Profit Sharing の学習の収束の様子を比較した結果を、Fig.9 に示す。また、Fig.9 を短いエピソードの区間で区切って見た結果が Fig.10 である。横軸がエピソード数、縦軸がゴールまでのステップ数である。180 エピソードから、300 エピソードの間に着目すると、提案手法の方がやや少ないステップ数でゴールに達することができている。これは、提案手法における転移学習により、通常の Profit Sharing と比較して評価関数への経験の伝搬が早いことによる効果と考

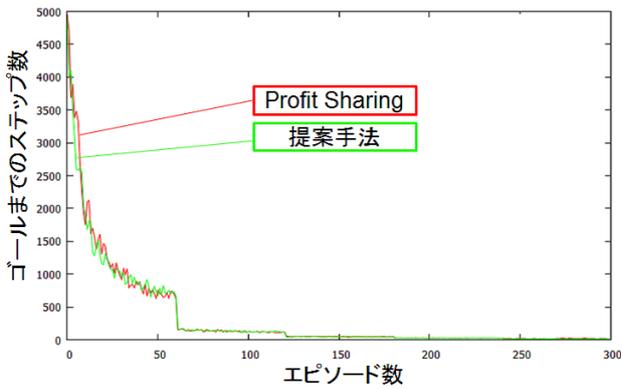


Fig. 9: 学習曲線

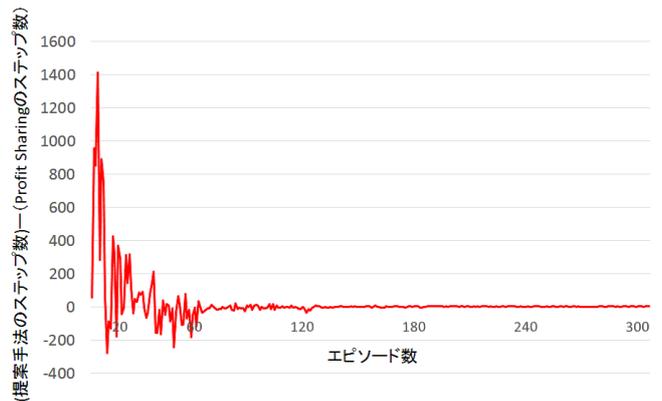


Fig. 11: 提案手法と Profit Sharing のステップ数の差

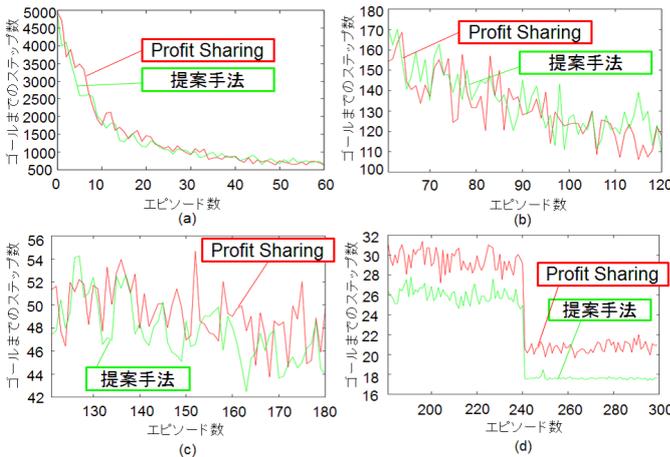


Fig. 10: 学習曲線の各部分

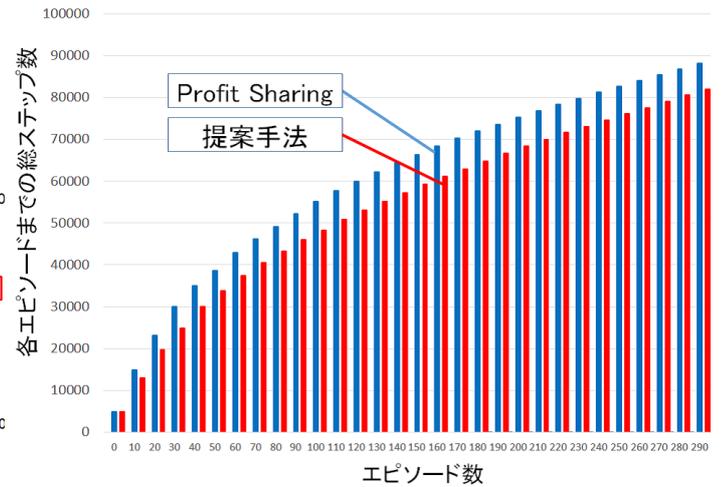


Fig. 12: 提案手法と Profit Sharing の各エピソードにおけるステップ数の総和

えられる。

Fig.11 に、各エピソードにおけるステップ数の提案手法と Profit Sharing の差を求めたものを示す。これを見ると、学習初期の段階において特に提案手法と Profit Sharing での探索回数に差がついていることが分かる。よって、学習初期段階の探索を減らす手法として、本手法が有効であると考えられる。さらに、Fig.12 に各エピソードに達するまでにエージェントが行ったステップの数の総和を提案手法と Profit Sharing で比較した結果を示す。これを見ると、60 エピソードまでは提案手法と Profit Sharing で総ステップ数の差が大きくなり、それ以降は差がほぼ変化していないことが分かる。このことから、学習初期段階において転移学習の効果が表れていることが分かる。また、300 エピソード目までに、提案手法の方が 3.82% 少ないステップ数で目的の行動を獲得可能であることが明らかになった。

7.2.3 課題

提案手法を用いて転移学習を行うことにより、通常の Profit Sharing と比較してやや少ないステップ数で適切な行動を獲得可能であることが明らかになった。しかしながら、期待よりもステップ数を減らすことができなかった。その原因として、以下の 2 点が挙げられる。

- 転移学習を考慮した方策のスケジューリング行われていない点

- 特徴空間における、近傍の定義が適切か明らかでない点

今回は ϵ -greedy 方策を用いて、エピソードごとにランダム行動選択確率 ϵ を決定するスケジューリング方法を用いた。この方法では、エピソードが同じであれば、学習が進んでいる状態、そうでない状態で、一律に同じ確率でランダム行動選択を行ってしまう。そのため転移学習を行って学習の進行を早くしたとしても、ランダム選択確率の高い学習初期においては転移の効果が陽に現れ辛い。そこで、学習進度に応じて ϵ を決定する方策を用いれば、転移学習の効果がより明確に現れると考えられる。

また、特徴空間における近傍の範囲 Δ_{\max} を予備実験に基づき決定しているが、この近傍の設定法が必ずしも適切であるとは言えない。そこで、近傍の範囲をより正確に決定することが可能ならば、さらに正確な転移学習が行われ、学習速度が向上すると考えられる。

8 まとめと今後の展望

skip-gram モデルを用いてタスク内で観測される特徴の共起関係を学習することにより、特徴を意味空間に写像可能であることを示した。また、その空間を用いて転移学習を行った結果、強化学習の学習速度が向上することを確認した。

今回は、近傍とみなすことができる空間の範囲 Δ_{\max} を予備実験に基づき決定したが、今後近傍の境界を自律的に獲得可能になれば、さらに本手法の適用範囲を広げることが可能であると考えられる。

参考文献

- 1) M.E.Taylor and P.Stone,: Transfer Learning for Reinforcement Learning domains: A Survey, Journal of Machine Learning Research, vol.10 pp1633-1685 (2009)
- 2) Reinaldo A. C. Bianchi, Carlos H. C. Ribeiro, and Anna H. R. Costa,: Heuristically Accelerated Q-learning: a new approach to speed up reinforcement learning, Lecture Notes in Artificial Intelligence, 3171, pp245-254 (2004)
- 3) M.Takeishi, T.Hamagami,: Transfer Learning for Q-learning by Using Similarity of Phase Structure in State Spaces, International Symposium on Artificial Life and Robotics (AROB 19th) (2014)
- 4) T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean,: Distributed Representations of Words and Phrases and their Compositionality, Accepted to NIPS (2013)
- 5) Gregmstette, J.J,: Credit Assignment in Rule Discovery Systems Based on Genetic Algorithms, Machine Learning, Vol.S pp225-245 (1988)
- 6) T.Kohonen,: The Self-Organizing Map, Proceedings of The IEEE, Vol. 78, No. 9, pp1464-1480 (1990)

非エキスパートエージェント集団からのアンサンブル逆強化学習

○富田真司 恒川裕章 鈴木拓央 濱上知樹 (横浜国立大学)

Ensemble Inverse Reinforcement Learning from Non-expert Agents

*S. Tomita, H. Tsunekawa, T. Suzuki, and T. Hamagami (Yokohama National University)

Abstract— Ensemble inverse reinforcement learning from non-expert agents' behavior is proposed. In many inverse reinforcement learning (IRL) problem, the expert agent which has ideal rewards for achieving the goal is supposed to be existing. However, in real world problem, the expert is not always observed. Moreover, the estimated reward function includes the bias depending on its inherent behavior if the reward for achieving the goal task is estimated from one agent. In order to overcome the limitation of IRL, we apply ensemble and boosting approach to IRL and integrate estimated reward functions from non-expert agents. To confirm the effectiveness of the proposed method, we compared the results of reinforcement learning using estimated reward functions and integrated reward function by simulation. The simulation result shows the proposed method can estimate the reward adaptively.

Key Words: inverse reinforcement learning, reinforcement learning, ensemble learning

1 はじめに

強化学習¹⁾(Reinforcement Learning, RL)は学習主体であるエージェントが報酬の設定された環境において自律的に行動を学習する手法である。人が立ち入ることが困難な環境における行動獲得や、複雑なパラメータ設定が必要なロボットの自律制御等への応用が期待されている。強化学習において適切な報酬の設定が難しいタスクでは、逆強化学習²⁾(Inverse Reinforcement Learning, IRL)を利用する。逆強化学習は、エキスパートの行動の軌跡や方策から報酬関数を推定する手法である。ここで、エキスパートは目的のタスクにおいて最適な振る舞いをするエージェントを指す。

しかし最適な振る舞いの定義が難しくエキスパートの発見が困難なタスクでは、振る舞いが最適ではないもののタスクを達成可能な非エキスパートエージェントから報酬を求めることになる。ここで、一体の非エキスパートから逆強化学習を行う場合、タスクの学習に適した報酬関数とは異なる報酬関数が推定される。これは逆強化学習に利用したエージェントが持つ固有の振る舞いに依存した報酬関数である。

そこで、よりタスクの学習に適した報酬関数を獲得するためにアンサンブル学習の応用を考える。アンサンブル学習は複数の弱学習器を組み合わせることで過学習の発生を抑制し、汎化性能を向上させる手法である。本稿では非エキスパートから逆強化学習によってより学習に適した報酬関数を得るためのアプローチとして、複数の非エキスパートから逆強化学習を行い、得られた報酬関数の統合を行うアルゴリズムを提案する。提案手法の概要を Fig.1 に示す。

2 見習い学習

本研究で用いている逆強化学習の手法である見習い学習³⁾について示す。

2.1 特徴ベクトルの設定

状態空間 S が持つ k 個の特徴を要素とした特徴ベクトル $\phi: S \rightarrow [0, 1]^k$ を考える。このとき、真の報酬関

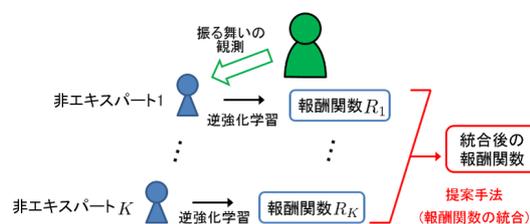


Fig. 1: 提案手法の概要

数 $R^*(s)$ は (1) 式のように表される。

$$R^*(s) = \mathbf{w}^* \cdot \phi(s), \quad \mathbf{w}^* \in \mathbb{R}^k \quad (1)$$

\mathbf{w}^* は各特徴の重みを表し、報酬の最大値を 1 以下に保つために、 $\|\mathbf{w}^*\|_1 \leq 1$ とする。

また、方策 π における状態価値関数 V^π の期待値 E は、初期状態を s_0 、割引率を γ として、(2) 式のように表される。

$$\begin{aligned} E_{s_0 \sim D}[V^\pi(s_0)] &= E[\sum_{t=0}^{\infty} \gamma^t R(s_t) | \pi] \\ &= \mathbf{w} \cdot E[\sum_{t=0}^{\infty} \gamma^t \phi(s_t) | \pi] \end{aligned} \quad (2)$$

方策 π のもとでの特徴ベクトルの期待値を特徴期待値 $\boldsymbol{\mu}(\pi)$ と呼び、(3) 式で定義する。

$$\begin{aligned} \boldsymbol{\mu}(\pi) &= E[\sum_{t=0}^{\infty} \gamma^t \phi(s_t) | \pi] \in \mathbb{R}^k \\ E_{s_0 \sim D}[V^\pi(s_0)] &= \mathbf{w} \cdot \boldsymbol{\mu}(\pi) \end{aligned} \quad (3)$$

エキスパートの方策を π_E とすると、 $s_0 \sim D$ からスタートし、 π_E に従って行動する状態遷移系列を観測することによって、真の報酬関数を推定することができる。

エキスパートの m 個の状態遷移系列 $\{s_0^i, s_1^i, \dots\}_{i=1}^m$ が与えられたとき、エキスパートの特徴期待値 $\hat{\boldsymbol{\mu}}_E = \boldsymbol{\mu}(\pi_E)$ は、(4) 式によって求められる。

$$\hat{\boldsymbol{\mu}}_E = \frac{1}{m} \sum_{i=1}^m \sum_{t=0}^{\infty} \gamma^t \phi(s_t^i) \quad (4)$$

2.2 エキスパートの方策の推定

$\mu(\pi_E)$ を他の特徴期待値を用いて (5) 式のように表すことができたとき, そのエキスパートの方策 π_E を (6) 式で推定する.

$$\begin{aligned}\mu(\pi_E) &= \lambda_1 \mu(\pi_1) + \lambda_2 \mu(\pi_2) + \cdots + \lambda_n \mu(\pi_n) \quad (5) \\ \pi_E &= \lambda_1 \pi_1 + \lambda_2 \pi_2 + \cdots + \lambda_n \pi_n \quad (6) \\ \sum_{i=1}^n \lambda_i &= 1, \forall_i \lambda_i \geq 0\end{aligned}$$

エキスパートの振る舞いに近い方策 $\tilde{\pi}$ を推定するために, $\|\mu_E - \mu(\tilde{\pi})\|_2 \leq \epsilon$ を満たす $\tilde{\pi}$ を求める必要がある. このとき,

$$\begin{aligned}\left| E \left[\sum_{t=0}^{\infty} \gamma^t R(s_t) | \pi_E \right] - E \left[\sum_{t=0}^{\infty} \gamma^t R(s_t) | \tilde{\pi} \right] \right| \\ = |\mathbf{w}^T \mu_E - \mathbf{w}^T \mu(\tilde{\pi})| \\ \leq \|\mathbf{w}\|_2 \|\mu_E - \mu(\tilde{\pi})\|_2 \\ \leq 1 \cdot \epsilon = \epsilon\end{aligned} \quad (7)$$

したがって, 問題は $\|\mu_E - \mu(\tilde{\pi})\|_2 \leq \epsilon$ となるような方策 $\tilde{\pi}$ を見つけることに帰着される. 以下, このような方策 $\tilde{\pi}$ を見つけるための計算手順を示す.

2.3 Projection Method

方策 $\tilde{\pi}$ を得るアルゴリズムの一つが Projection Method³⁾(PM) である. Projection Method で用いる式を以下に示す.

$$\bar{\mu}^{(i-1)} = \bar{\mu}^{(i-2)} + \frac{(\mu^{(i-1)} - \bar{\mu}^{(i-2)})^T (\mu_E - \bar{\mu}^{(i-2)})}{(\mu^{(i-1)} - \bar{\mu}^{(i-2)})^T (\mu^{(i-1)} - \bar{\mu}^{(i-2)})} \cdot (\mu^{(i-1)} - \bar{\mu}^{(i-2)}) \quad (8)$$

$$\mathbf{w}^{(i)} = \mu_E - \bar{\mu}^{(i-1)} \quad (9)$$

$$t^{(i)} = \|\mu_E - \bar{\mu}^{(i-1)}\|_2 \quad (10)$$

ただし $\bar{\mu}^{(0)} = \mu^{(0)}$ とする. Projection Method のアルゴリズムを以下に示す. またこのアルゴリズムが進行する様子を Fig.2 に示す.

1. ランダムに選んだ方策 π_0 のもとで $\mu^{(0)} = \mu(\pi_0)$ を計算し, $i = 1$ とする.
2. (8)~(10) 式により $\bar{\mu}^{(i-1)}$, $\mathbf{w}^{(i)}$, $t^{(i)}$ を求める.
3. $t^{(i)} \leq \epsilon$ のとき, アルゴリズムを終了する.
4. 報酬関数 $R = \mathbf{w}^{(i)} \cdot \phi$ のもとでの最適方策 π_i を, 強化学習により求める.
5. 方策 π_i のもとで $\mu^{(i)} = \mu(\pi_i)$ を計算する.
6. $i \leftarrow i + 1$ としてステップ 2 へ戻る.

2.4 報酬関数の推定

本研究では, 見習い学習を用いて得られた π_E から, その方策を得るための特徴ベクトルの重み $\hat{\mathbf{w}}_E$ を (11) 式のように推定し実験に用いている.

$$\hat{\mathbf{w}}_E = \lambda_1 \mathbf{w}_1 + \lambda_2 \mathbf{w}_2 + \cdots + \lambda_n \mathbf{w}_n \quad (11)$$

ここで, \mathbf{w}_i は π_i を獲得したときの特徴ベクトルに対する重みである.

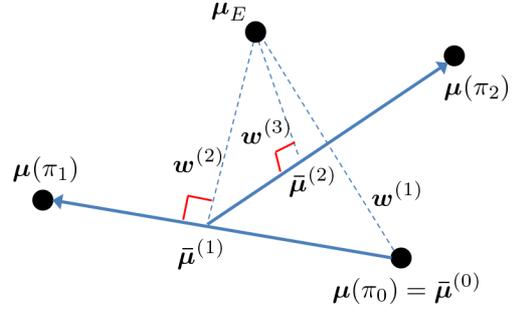


Fig. 2: $i = 3$ までの PM アルゴリズム

3 アンサンブル逆強化学習

提案手法はアンサンブル学習の一種である Adaboost⁴⁾ のアルゴリズムを応用している. Adaboost は前の弱学習器が誤判定した学習データの重みを増やすことで逐次的に重みを更新するアルゴリズムである. 学習データに対する判定精度から弱学習器の信頼度を求め, 全ての弱学習器を信頼度で重み付けして統合することでより判定精度の高い学習器を構成する.

Adaboost アルゴリズムを報酬関数の統合に応用した提案手法のアルゴリズムを以下に示す. Adaboost アルゴリズムにおける弱学習器を非エキスパートとする. また, 学習データを特徴期待値とする. 各ステップにおける計算は後述する.

観測される非エキスパートの数を K , 非エキスパート番号を $k = 1, 2, \dots, K$ とする. また, 設定した特徴数を L , 特徴番号を $l = 1, 2, \dots, L$ とする.

1. K 体の非エキスパートを観測しそれぞれの特徴期待値を獲得する. $k = 1$ とする.
2. 各特徴に対する重み β_l を更新する. 初期の重みは全て $1/L$ とする. (3.3 節)
3. k 番目の非エキスパートから逆強化学習を行い, 報酬関数 $R^{(k)}$ を獲得する.
4. k 番目の非エキスパートの各特徴の平均誤差を計算する. その誤差の閾値判定を行い, 重みを増やす特徴を決定する. (3.1 節)
5. 平均誤差の総和の逆数を取り非エキスパートの信頼度を計算する. (3.2 節)
6. $k = K$ のとき, 信頼度の正規化後に報酬関数を統合しアルゴリズムを終了する. (3.4 節)
7. $k \leftarrow k + 1$ とし, ステップ 2 へ戻る.

次節から, 提案手法の定式化を行う.

3.1 平均誤差の計算

l 番目の特徴の標準偏差 σ_l は (12) 式で表される.

$$\sigma_l = \sqrt{\frac{\sum_{k=1}^K \left(\mu_l^{(k)} - \bar{\mu}_l \right)^2}{K}} \quad (12)$$

特徴期待値と、観測された全ての非エキスパートの特徴期待値の平均値との差によって平均誤差を計算する。このとき、標準偏差を用いて正規化を行う。

特徴 ϕ_l に関する k 番目の非エキスパートの平均誤差 $E_l^{(k)}$ は、

$$E_l^{(k)} = \begin{cases} \frac{|\mu_l^{(k)} - \bar{\mu}_l|}{\sigma_l} & (\sigma \neq 0) \\ 0 & (\sigma = 0) \end{cases} \quad (13)$$

$E_l^{(k)}$ が閾値 T を上回ったとき、この特徴をエラーと判定する。

3.2 非エキスパートの信頼度の計算

特徴の重みを考慮し平均誤差の総和をとると、 k 番目の非エキスパートの平均誤差の総和 $E^{(k)}$ は、

$$E^{(k)} = \sum_{l=1}^L \beta_l E_l^{(k)} \quad (14)$$

$E^{(k)}$ の逆数を取り信頼度 $\alpha^{(k)}$ とする。

$$\alpha^{(k)} = \frac{1}{E^{(k)}} \quad (15)$$

3.3 特徴の重みの更新

k 番目の非エキスパートの特徴に対する重みを $\beta^{(k)}$ と表すこととする。 $k=1$ のとき、重みは均等である。

$$\begin{aligned} \beta^{(1)} &= \{\beta_1^{(1)}, \beta_2^{(1)}, \dots, \beta_L^{(1)}\} \\ &= \left\{ \frac{1}{L}, \frac{1}{L}, \dots, \frac{1}{L} \right\} \end{aligned} \quad (16)$$

ここで、 i 番目の特徴がエラーと判定されたとする。 k 番目の非エキスパートにおいてエラーと判定された特徴の重みの総和を $\beta_{\text{error}}^{(k)} = \sum_i \beta_i^{(k)}$ と表し、全ての l に対して (17) 式の更新を行う。

$$\beta_l^{(k+1)} = \begin{cases} \beta_l^{(k)} \cdot \frac{1}{\beta_{\text{error}}^{(k)}} & (E_l^{(k)} > T) \\ \beta_l^{(k)} & (E_l^{(k)} \leq T) \end{cases} \quad (17)$$

さらに $|\beta^{(k+1)}| = \sum_{l=1}^L \beta_l^{(k+1)}$ と表し、重みの総和を 1 に保つために (18) 式により正規化を行う。

$$\beta_l^{(k+1)} \leftarrow \beta_l^{(k+1)} \cdot \frac{1}{|\beta^{(k+1)}|} \quad (18)$$

3.4 信頼度による報酬関数の統合

$|\alpha| = \sum_{k=1}^K \alpha^{(k)}$ と表し、信頼度の総和が 1 となるように (19) 式により正規化を行う。

$$\alpha^{(k)} \leftarrow \frac{\alpha^{(k)}}{|\alpha|} \quad (19)$$

k 番目のエージェントが持つ報酬関数を $\mathbf{R}^{(k)}$ と表すと、これらの報酬関数を統合した報酬関数 $\tilde{\mathbf{R}}$ は (20) 式のように表される。

$$\begin{aligned} \tilde{\mathbf{R}} &= \alpha^{(1)} \mathbf{R}^{(1)} + \alpha^{(2)} \mathbf{R}^{(2)} + \dots + \alpha^{(K)} \mathbf{R}^{(K)} \quad (20) \\ \sum_{k=1}^K \alpha^{(k)} &= 1, \forall_k \alpha^{(k)} \geq 0 \end{aligned}$$

ここで、(1) 式より、(20) 式は (21) 式と同値である。

$$\tilde{\mathbf{R}} = (\alpha^{(1)} \mathbf{w}^{(1)} + \alpha^{(2)} \mathbf{w}^{(2)} + \dots + \alpha^{(K)} \mathbf{w}^{(K)}) \cdot \phi(s) \quad (21)$$

したがって、 $\tilde{\mathbf{R}}$ を得るための特徴の重みベクトル $\tilde{\mathbf{w}}$ は、(22) 式のように表される。

$$\tilde{\mathbf{w}} = \alpha^{(1)} \mathbf{w}^{(1)} + \alpha^{(2)} \mathbf{w}^{(2)} + \dots + \alpha^{(K)} \mathbf{w}^{(K)} \quad (22)$$

4 実験および考察

提案手法の有効性を確認するために、Fig.3 に示すドライビングタスクを用いてシミュレーション実験を行った。ドライビングタスクは、3 本の走行車線とその両側に車の走っていない路側帯が設けられている。自分の車の速度は他のどの車よりも速く、他の車を避けながら走行するタスクである。3 車線全てに他の車が走っている場合などは、路側帯を走行して衝突を回避する振る舞いが望まれる。

設定した特徴ベクトルは以下のようにになっている。走行車線に関する特徴が 5 つ、車間距離に関する特徴が 5 つであり、計 10 個の特徴を要素とした特徴ベクトルである。特徴ベクトルの要素は全て真なら 1、偽なら 0 を返す関数である。

特徴ベクトル $\phi = \{\phi_1, \phi_2, \dots, \phi_{10}\}$ の設定

- ϕ_1 : 左路側帯を走行中
- ϕ_2 : 左車線を走行中
- ϕ_3 : 中央車線を走行中
- ϕ_4 : 右車線を走行中
- ϕ_5 : 右路側帯を走行中
- ϕ_6 : 他の車と衝突している
- ϕ_7 : 先行車両との車間距離が短い
- ϕ_8 : 先行車両との車間距離が普通
- ϕ_9 : 先行車両との車間距離が長い
- ϕ_{10} : 現在の車線の前方に車なし

提案手法のアルゴリズムの妥当性を検証した実験 1、提案手法の有効性を確認した実験 2 の 2 つの実験を示す。

4.1 実験 1: パラメータ設定による信頼度変化の比較

提案手法ではパラメータとしてエラー判定を行うための閾値 T の設定が必要である。また、実験に利用す

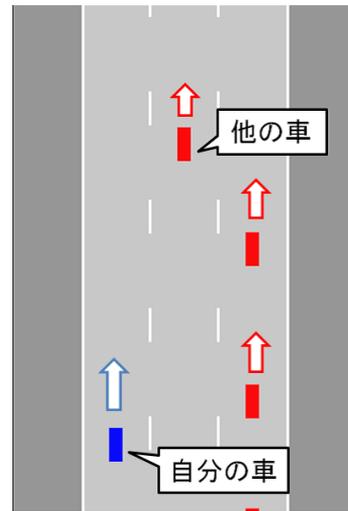


Fig. 3: ドライビングタスク

Table 1: 特徴期待値の観測結果

	ϕ_1	ϕ_2	ϕ_3	ϕ_4	ϕ_5
μ_1	0.0313	0.1785	0.3548	0.4356	0.0000
μ_2	0.0397	0.2921	0.3828	0.2853	0.0000
μ_3	0.0300	0.2030	0.4534	0.3136	0.0000
	ϕ_6	ϕ_7	ϕ_8	ϕ_9	ϕ_{10}
μ_1	0.0000	0.0651	0.2430	0.1818	0.5102
μ_2	0.0000	0.0112	0.0880	0.1708	0.7301
μ_3	0.0000	0.3543	0.2971	0.0860	0.2626

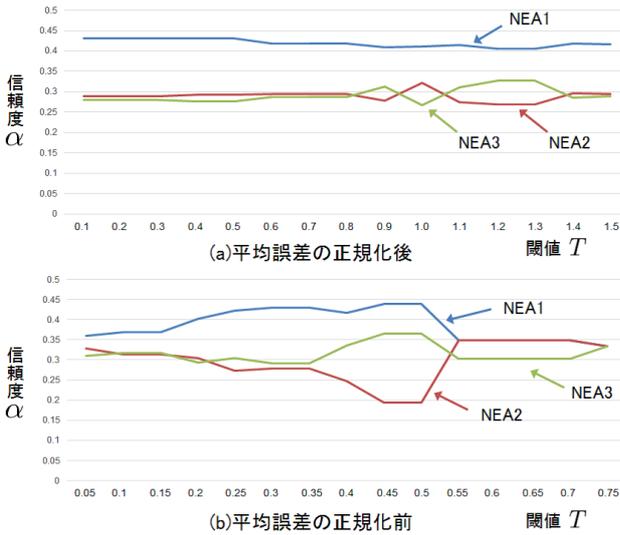


Fig. 4: 閾値 T による信頼度変化

非エキスパートの追加順序もパラメータの一つと言える。そこで、得られる信頼度のパラメータ設定による変化について検証を行った。

また、平均誤差の正規化の効果を確認するために、正規化後、正規化前のそれぞれの場合で結果の検証を行った。

4.1.1 実験設定

非エキスパート 3 体の振る舞いをそれぞれ 20000 ステップ観測した。観測した非エキスパートをそれぞれ NEA1, NEA2, NEA3 とする。全ての非エキスパートは、他の車との衝突回数が 0 であること、衝突が避けられない場合以外は路側帯に出ないことを共通の振る舞いとした。また個々の非エキスパートの振る舞いとして、

NEA1: なるべく右側車線を走ること
 NEA2: なるべく車間距離が長い車線を走行すること
 NEA3: 他の車の後ろに隠れるように走行することをそれぞれ設定した。NEA1, NEA2, NEA3 の特徴期待値をそれぞれ μ_1, μ_2, μ_3 として、観測結果を Table.1 に示す。Table.1 に示した特徴期待値を用いて、提案手法により 3 体の非エキスパートの信頼度を測定した。

4.1.2 結果と考察

閾値設定による信頼度の変化を Fig.4 に、非エキスパートの追加順序の変更による信頼度の変化を Fig.5 に示す。非エキスパートの追加順序は、

① NEA1 → NEA2 → NEA3

Table 2: 特徴期待値の観測結果

	ϕ_1	ϕ_2	ϕ_3	ϕ_4	ϕ_5
μ_4	0.1134	0.3130	0.3481	0.1443	0.0811
	ϕ_6	ϕ_7	ϕ_8	ϕ_9	ϕ_{10}
μ_4	0.0000	0.0202	0.1340	0.1642	0.6816

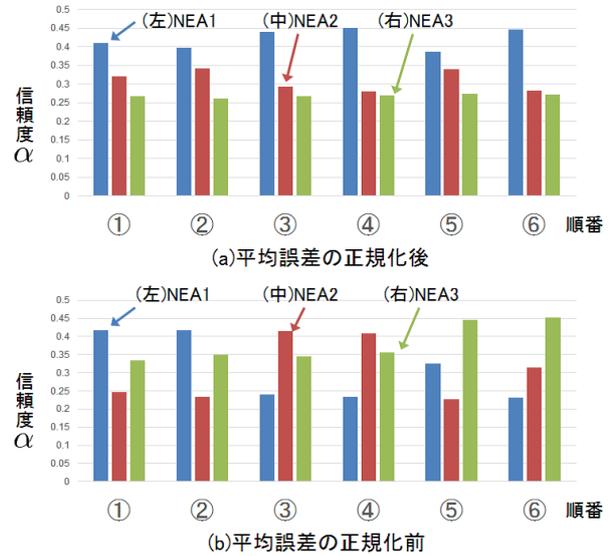


Fig. 5: 追加順序による信頼度変化

- ② NEA1 → NEA3 → NEA2
 - ③ NEA2 → NEA1 → NEA3
 - ④ NEA2 → NEA3 → NEA1
 - ⑤ NEA3 → NEA1 → NEA2
 - ⑥ NEA3 → NEA2 → NEA1
- として図に表している。

Fig.4 の (a) より、正規化後の場合はパラメータ変更の影響が小さく、設定値に依存しにくい結果が得られた。それに対して (b) の正規化前の場合では、得られた信頼度がパラメータ設定に強く依存した結果となった。Fig.5 でも同様に、正規化を行うことで非エキスパートの追加順序に依存しにくい結果が得られていることが分かる。

このことから、アルゴリズムの安定性の向上のために標準偏差による平均誤差の正規化が有効であると言える。

4.2 実験 2: 報酬関数の統合前後の学習精度の比較

統合前後の報酬関数を用いて、強化学習による学習精度の比較を行った。強化学習の価値更新は Q-learning を適用した。学習精度の指標は他の車との衝突率としている。衝突率は走行時間に対する衝突時間の割合であり、特徴 ϕ_6 に対する特徴期待値を表す。

4.2.1 実験設定

非エキスパート 4 体の振る舞いをそれぞれ 20000 ステップ観測した。NEA1, NEA2, NEA3 の振る舞いは実験 1 と同様である。NEA4 の振る舞いとして、NEA4: 2 車線以上の急な車線変更をしないを設定した。すなわち同じ方向へ続けて車線変更をしない振る舞いとなっている。NEA4 の特徴期待値を μ_4

として、観測結果を Table.2 に示す.

4 体の非エキスパートから得られた報酬関数と、それらを統合した報酬関数の計 5 つを用いて強化学習を行った. 見習い学習, 強化学習におけるパラメータを Table.3 に示す. ランダム行動率 ϵ は 20000 ステップごとに 0.1 として設定している. また, 非エキスパートは NEA1, NEA2, NEA3, NEA4 の順に逆強化学習を行い統合時に追加処理している.

Table 3: 実験 2 パラメータ

パラメータ	設定値
NEA の観測時間	20000 ステップ
終了条件	$i = 20$
強化学習の時間	200000 ステップ
学習率 α	0.3
割引率 γ	0.9
ランダム行動率 ϵ	1 ~ 0.1
エラー閾値 T	0.1

4.2.2 結果と考察

それぞれの非エキスパートから得られた特徴の重みベクトル w と統合後の重みベクトル \tilde{w} を Table.4 に示す. またそれらを用いて強化学習を行った際の衝突率を Table.5 に示す.

Table.4 において衝突に関する特徴である ϕ_6 に注目すると, 統合後の報酬関数よりも NEA3 の持つ報酬関数 w_3 の方がより小さな報酬となっている. しかし Table.5 より, 統合後の報酬関数を用いて強化学習を行った場合の衝突率が一番低くなっていることが分かる. NEA3 の衝突率が他の非エキスパートと比べて大きくなっているのは, 先行車両との車間距離が近いことを表す特徴である ϕ_7 に対して大きな報酬が設定されているためであると考えられる. このように, 衝突以外の特徴に対する報酬設定も衝突率に大きく関与していることが分かる. 統合後の報酬関数は, 各特徴の重み付けが改善し, よりタスクの学習に適した報酬関数と言える.

以上の結果より, 個々の非エキスパートから推定された報酬関数に比べて, 提案手法によりそれらを統合した報酬関数の方がタスクの学習に適していることを示し, 提案手法の有効性を確認した.

5 おわりに

本稿では, 非エキスパートエージェント集団からのアンサンブル逆強化学習について提案した. 提案手法は, 複数の非エキスパートエージェントから逆強化学習によって推定された報酬関数の統合を行うことで, 学習精度の向上を図る手法である. 報酬関数の統合は, アンサンブル学習の一つである Adaboost のアルゴリズムを応用している.

提案手法の有効性を確認するために, ドライビングタスクを用いたシミュレーション実験を行った. 実験 1 では提案手法のアルゴリズムの妥当性の検証, 実験 2 では報酬関数の統合による学習精度向上の検証を行い, 提案手法の有効性を示した.

Table 4: 統合前後の重みベクトル

	ϕ_1	ϕ_2	ϕ_3	ϕ_4	ϕ_5
w_1	-0.0976	-0.0044	0.0698	0.1496	-0.1174
w_2	-0.1391	0.0726	0.0766	0.1157	-0.1259
w_3	-0.0364	0.0141	0.0490	0.0417	-0.0684
w_4	-0.0743	0.1117	0.1205	-0.0230	-0.1350
\tilde{w}	-0.0975	0.0500	0.0793	0.0841	-0.1160
	ϕ_6	ϕ_7	ϕ_8	ϕ_9	ϕ_{10}
w_1	-0.1388	-0.1418	0.1062	0.1226	0.0517
w_2	-0.0836	-0.1153	-0.0365	0.1157	0.1190
w_3	-0.3539	0.3850	0.0102	-0.0010	-0.0403
w_4	-0.1060	-0.1617	0.0311	0.1347	0.1019
\tilde{w}	-0.1478	-0.0497	0.0241	0.1023	0.0709

Table 5: 統合前後の衝突率

重みベクトル	衝突率 (特徴期待値 ϕ_6)
ランダム行動 (参考)	0.0741
w_1	0.0200
w_2	0.0351
w_3	0.0380
w_4	0.0186
\tilde{w}	0.0112

5.1 参考文献

参考文献

- 1) Richard S. Sutton and Andrew G. Barto (三上貞芳, 皆川雅章 共訳): “強化学習”, 森北出版株式会社, 2003.
- 2) Andrew Y. Ng, Stuart Russell : “Algorithms for Inverse Reinforcement Learning”, In Proceedings of the 17th International Conference on Machine Learning, 2000.
- 3) Pieter Abbeel, Andrew Y. Ng : “Apprenticeship learning via Inverse Reinforcement Learning”, In Proceedings of the 21th International Conference on Machine Learning, 2004.
- 4) Y. Freund and R. E. Schapire: “Experiments with a New Boosting Algorithm”, In Proceedings of the 13th International Conference on Machine Learning, 1996.
- 5) Thomas G. Dietterich: “Ensemble Methods in Machine Learning”, In Proceedings of Multiple Classifier Systems, 2000.
- 6) 荒井幸代: “逆強化学習によるマルチエージェント系の報酬設定”, 計測と制御 第 52 巻 第 6 号 pp.534-539, 2013.

帆走の強化学習において人間の操作ログを行動選択に用いる効果

○真鍋秀朗, 橘完太 (工学院大学)

The Effect of Using Human's Operation Log to Select Action for Reinforcement Learning of Unmanned-Sailing

* Hideaki Manabe and Kanta Tachibana (Kogakuin University)

Abstract— About Sailing, propulsive force of sailing yacht changes by the relative velocity between boat and wind, controller of sailing yacht need to take future state and future action into consideration. Therefore, we use reinforcement learning that we can distribute rewards to effective combinations of state and action. In conventional reinforcement learning, it takes long time before agent obtains the first reward because action of agent is decided almost at random in early stage of learning. Expecting faster learning, we decide action as the same as human's operation log. This paper compares learning results of conventional method and our method which uses human's operation log.

Key Words: Sailing, Reinforcement Learning

1 はじめに

帆走は、帆に受けうる推進力が船と風との相対速度によって変化するため、将来にわたって状態と行動を予測する必要がある。そのため、状態と行動の組の系列へ報酬を分配できる強化学習が有効であると考え。

従来の強化学習においては、エージェントの初期行動はほぼランダムに決定されるため、エージェントが最初の報酬を得るまでの時間が長くなってしまふ。そこで、本研究では、あらかじめ人間が操作した行動を学習初期の行動に取り入れればより早い学習が期待できると考え、帆走の強化学習におけるエージェントの初期行動を人間が実際に操作したログから決定し、学習を進めていく。従来の手法で学習を進めた場合と人間の操作ログを行動選択に用いて学習を進めた場合を比較する。

2 実験内容

2.1 実験概要

本研究では帆船を自動操縦させることにより、効率の良い帆走を学習させることを目的とする。

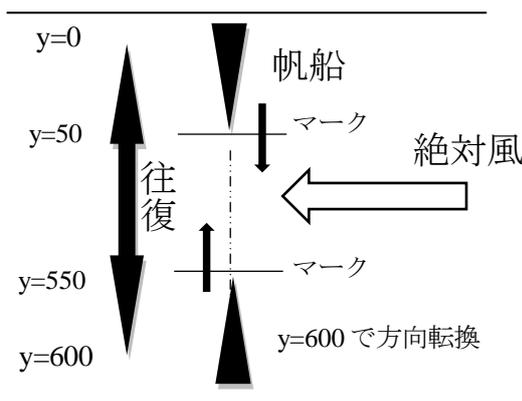


Fig.1: 帆船の自動操縦実験の概要

問題を単純化するため、画面の下向きを y 軸とし、 y 座標が 0 から 600 である 1 次元の空間を考えて $y=50$ と $y=550$ の区間の往復運動をエージェントに強化学習させる (Fig.1).

第 1 エピソードの最初のステップでは、帆船が $y=0$ の位置に速度 0 でいる。下方向に $y=550$ まで帆走してから上向きに方向転換して再び $y=50$ の位置に戻ると、報酬を得る。また、帆船が $y=0$, $y=600$ の位置に到達した場合、帆船が壁に衝突したと考え、速度を 0 にする。

帆走では、風向きと直交して走ると大きな加速度を得やすいため、画面の右側から一定の絶対風が吹く状況を設定する。帆船の速度は、推進力と水から受ける抵抗力により決まる。帆船が動くとき、帆船から見た見かけの風は、絶対風と帆船の速度の逆向きとを合成した風となる¹⁾。風が帆を押す力は帆が風を遮る面積に比例する；風が帆を押す力は風速の 2 乗に比例する；と仮定すると、帆船の進行方向に対する見かけの風の方向を ϕ 、見かけの風速を W とすると、帆船は風から $W^2(1 - \cos \phi)$ に比例する推進力を得る。帆船が水から受ける抵抗力は速さの 2 乗に比例すると仮定する。このような現実に近いシミュレーションのもとで、帆走の強化学習を行う。

2.2 人間の操作ログ

操作ログは Fig.2 で示す帆船操縦プログラムから取ってくるものとし、入力はキーボードで行う。キーを上に入力すると帆船が上に進み、キーを下に入力すると帆船が下に進む。キーを離した状態にすると現在の速度を維持する。

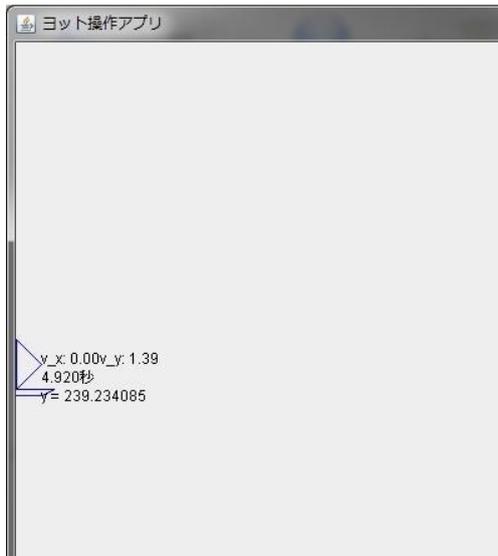


Fig.2 : 帆船操縦プログラム

1ステップは1ミリ秒とし、人間が操作した場合の1エピソードの平均はおよそ9,801ステップとなる。被験者は2人、1往復を1エピソードとして合計22往復分行った。分散は1,535,805、標準偏差は1239.28である。

2.3 帆走の強化学習

帆船の強化学習では、状態空間、行動空間ともに連続となる。連続な状態空間を次のように36の領域に離散化する。状態変数は帆船のy座標とy方向の速さ（以下、 v_y ）である。

y座標と v_y をそれぞれ6区分に区切り、全ての状態数は $6 \times 6 = 36$ である。

y座標による状態の区分は以下の通りである。

- y_1 : yが0以上50未満
- y_2 : yが50以上300未満（往路）
- y_3 : yが300以上550未満（往路）
- y_4 : yが550以上600未満
- y_5 : yが300以上550未満（復路）
- y_6 : yが50以上300未満（復路）

また、 v_y による状態の区分は以下の通りである。

- v_{y1} : v_y が-2未満
- v_{y2} : v_y が-2以上-1未満
- v_{y3} : v_y が-1以上0未満
- v_{y4} : v_y が0以上1未満
- v_{y5} : v_y が1以上2未満
- v_{y6} : v_y が2以上

比較対象の従来法として、Q学習を用い、行動決定

はepsilon-greedy法を利用する。従来のQ学習においては、Q表の各要素の初期値を一定値やランダムに設定することが一般的である。

学習におけるQ値の更新式は、

$$\hat{Q}(s_t, a_t) \leftarrow \hat{Q}(s_t, a_t) + \alpha \{r_t + \gamma \max_a \hat{Q}(s_{t+1}, a) - \hat{Q}(s_t, a_t)\}$$

である²⁾。ここで、 s_t は時刻tにおける状態、 a_t は時刻tにおける行動、 r_t は時刻tに獲得した報酬である。

学習パラメータは以下の通りに設定する。

学習係数 $\alpha = 0.1$

割引率 $\gamma = 0.9$

epsilon-greedy法における $\epsilon = 0.3$

Qの初期値 = 0

状態数 = y座標の状態6 × v_y の状態6 = 36

行動数 = 3（上に加速、速度維持、下に加速）

帆船がマーク地点を回った後に $y=50$ まで戻った時点で報酬 r_t を与え、それ以外の場合は報酬を与えない。 r_t は周回するまでにかかった時間に反比例し、

$$r_t = k/T_R$$

とする。なお、 k は報酬を設定する際の係数であり、本実験では $k=1,000,000$ とする。また、 T_R は帆船が往復するのに経過した時間である。 T_R が長ければ長いほど報酬が減少する。往復を完了した時点で1エピソードが終了する。1エピソードが終了する度に T_R を0に戻し、次のエピソードを開始する。

そのため、エージェントが最初の報酬を得るまでは、行動の望ましさをQ表で表すことができず、Q表に基づく行動選択はランダムに近いものとなる。報酬を得るたびに、この状況は改善されていくが、帆走のように複雑な対象を強化学習する際には、現実的な時間で学習が収束しない恐れがある。

2.4 本研究の提案手法

そこで、本研究では人間の操作ログから学習を進める手法を提案し、提案の効果を調べる。Fig.3 に示すように、まず、最初の何エピソードかの往復運動を人間が操作し、その間も学習を進める。何エピソードかを人間が操作した後はエージェントが学習したQ値に基づいて行動を選択する。本研究では、初期段階の人間による行動選択のエピソード数を変えた場合、学習にどのような違いが見られるのかを考察する。

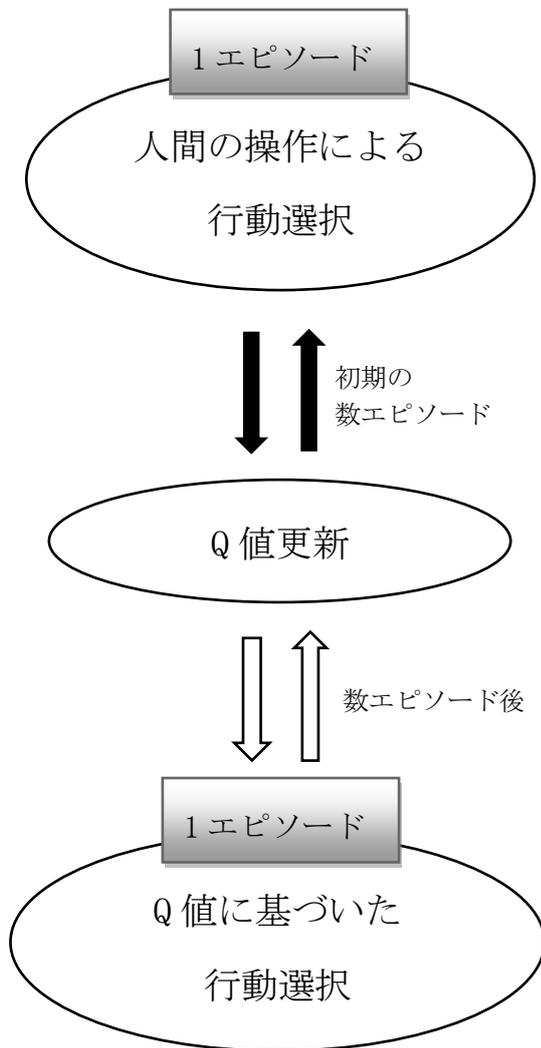


Fig.3 : 行動選択と学習の概要

3 実験結果

実験結果をそれぞれ Fig.4 - Fig.8 に示す。横軸はエピソード数、縦軸はエピソードにおけるステップ数（周回時間）である。

3.1 従来手法での実験結果

Fig.4 は人間の操作ログを用いなかった学習結果である。これを以後『試行 H0』とする。

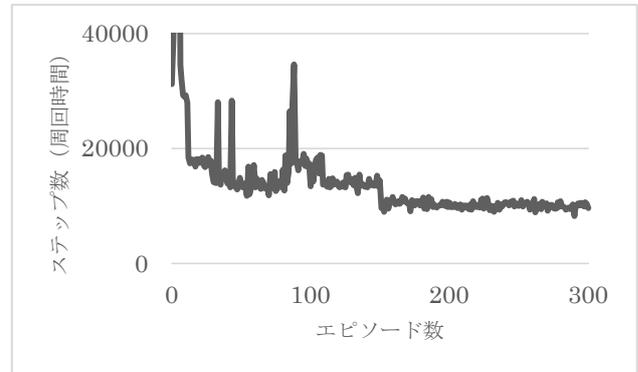


Fig.4 : 従来の Q 学習による学習の結果 (試行 H0)

3.2 提案手法での実験結果

Fig.5,6,7,8はそれぞれ最初の3,4,5,10エピソードを人間の操作ログによって行動選択しながら学習を行った結果である。これらをそれぞれ以後、『試行 H3』、『試行 H4』、『試行 H5』、『試行 H10』とする。Fig.5-Fig.8における点線より左側のエピソードは、人間による行動選択を行ったエピソードを示す。

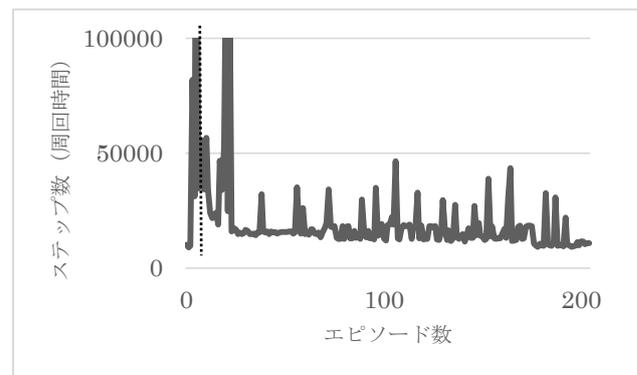


Fig.5 : 最初の3エピソードを人間による行動選択をして学習を進めた場合の結果 (試行 H3)

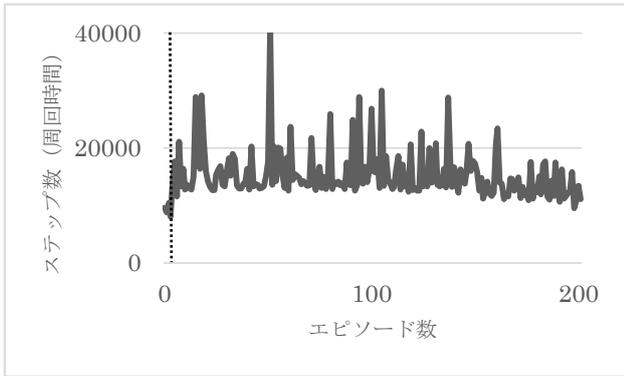


Fig.6：最初の4エピソードを人間による行動選択をして学習を進めた場合の結果（試行 H4）

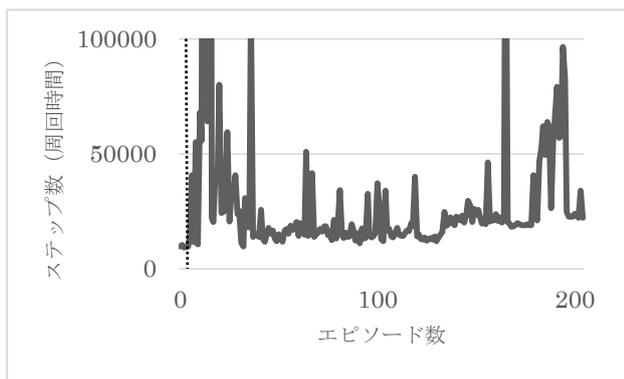


Fig.7：最初の5エピソードを人間による行動選択をして学習を進めた場合の結果（試行 H5）

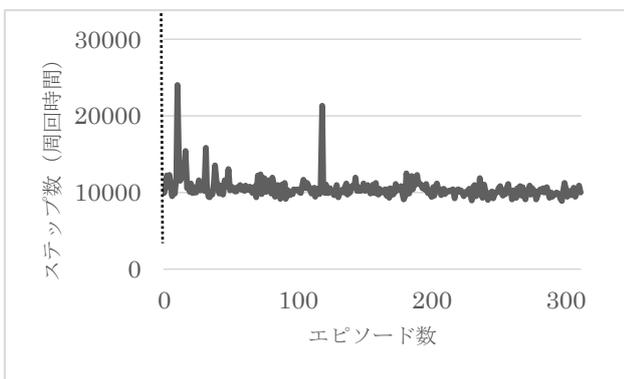


Fig.8：最初の10エピソードを人間による行動選択をして学習を進めた場合の結果（試行 H10）

Table 1 に、エピソード区間における累積ステップ数を 50 エピソードごとに算出した結果を示す。累積ステップ数は折れ線グラフを積分した値であり、そのエピソード区間におけるエージェントの学習回数である。

Table 1：ステップ数の累積値

エピソード区間	人間が行動選択したエピソード数				
	0(従来)	3	4	5	10
1-50	1,701,942	1,537,348	759,155	2,558,917	569,937
51-100	790,124	855,386	816,860	881,260	525,709
101-150	723,196	857,930	802,332	939,951	530,033
151-200	523,345	747,137	676,502	1,823,746	523,940

4 考察

試行 H0 の結果(Fig.4)からは、操作ログを用いなければ、学習が進むまでは周回に膨大な時間がかかることが分かる。

100 エピソード手前までは学習がほとんど進まず、ステップ数が減少しない。150 エピソード付近までには 15,000 ステップ前後へ減少する。150 エピソード以降はさらに学習が進み、約 10,000 ステップで収束する。

一方で、試行 H10 の結果(Fig.8)は、10 エピソード後も従来手法で収束した約 10,000 ステップを保持することが分かる。

Table 2 に、試行 H4 において 4 エピソードが終わった時点で Q 値の最も高かった行動を状態ごとに示す。また、Table 3 に、試行 H10 において 10 エピソードが終わった時点で Q 値の最も高かった行動を状態ごとに示す。また、空欄は Q 値が更新されなかった状態である。

Table 2：試行 H4 で Q 値の最も高かった行動 (4 エピソード後)

	v_{y1}	v_{y2}	v_{y3}	v_{y4}	v_{y5}	v_{y6}
y_1						
y_2						
y_3						
y_4	上	上	上	維持		維持
y_5	上					
y_6	上					

Table 3：試行 H10 で Q 値の最も高かった行動 (10 エピソード後)

	v_{y1}	v_{y2}	v_{y3}	v_{y4}	v_{y5}	v_{y6}
y_1	下	下	下	下	下	下
y_2						下
y_3					上	上
y_4	上	上	上	上	上	上
y_5	上	上				
y_6	下					

試行 H10 の場合、試行 H3、試行 H4、試行 H5 に比べて学習が早く進んだ理由は、Table 2、Table 3 が示すように、人間による行動選択が終わった時点でより多くの状態において Q 値の更新がされたためだと考えられる。

Table 4 に試行 H10 の 200 エピソード後に Q 値の最も高かった行動を状態ごとに示す。また、Table 5 に試行 H0 の 200 エピソード後に Q 値の最も高かった行動を状態ごとに示す。また、太字で示した場所は Table 4 と Table 5 において行動が異なる状態である（空欄は除く）。

Table 4 : 試行 H10 で Q 値の最も高かった行動 (200 エピソード後)

	v_{y1}	v_{y2}	v_{y3}	v_{y4}	v_{y5}	v_{y6}
y_1	維持	上	維持	下	下	下
y_2		維持			上	下
y_3				下	上	下
y_4		上	上	下	上	上
y_5	上	上	上	維持	下	下
y_6	下	維持	下			

Table 5 : 試行 H0 で Q 値の最も高かった行動 (200 エピソード後)

	v_{y1}	v_{y2}	v_{y3}	v_{y4}	v_{y5}	v_{y6}
y_1	維持	上	下	下	維持	
y_2	下	維持		下	下	下
y_3		下		下	上	下
y_4	維持	上	上	下	下	下
y_5	上	上	上	下	上	
y_6	下	維持	上			

Table 4 において、 (y_2, v_{y5}) の行動は試行 H10 では「上」、試行 H0 の場合では「下」となっている。これは人間が操作する場合は、 $y=600$ の地点に壁があるため、壁に衝突しないように減速する必要があるために「上」という行動が多くなったと考える。また、試行 H0 で学習した場合、あえて壁に衝突して速度を 0 にすることで逆方向への加速を素早くするという行動を学習したため、「下」という行動が多くなったと考える。

このように、機械学習により人間が操作しただけでは思い付かない行動をすることがあるが、実際のように壁に衝突することが推奨されないような状況では、機械学習のみならず、最初の数エピソードの行動選択を人間がする必要がある。また、壁に衝突した状態で学習を進めた場合と、壁に衝突せずに学習を進めた場合のどちらも、10,000 ステップというほぼ同じ値に収束する結果になったため、実際の壁に衝突せずに学習を進めることができる人間の方がより効率的な帆走ができると考える。

以下に Fig.9 は試行 H0、試行 H4、試行 H10 におけるそれぞれの学習回数とステップ数の変移である。また、Fig.10 は Fig.9 のグラフの縦軸を 40,000 ステップ以内で拡大した図である。

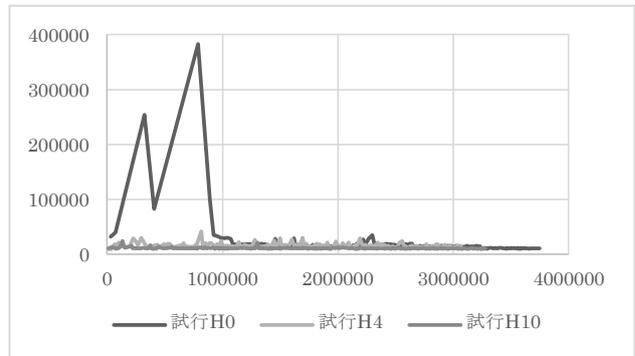


Fig.9 : 人間による行動選択のエピソード数を変えた際の学習回数とステップ数の変移

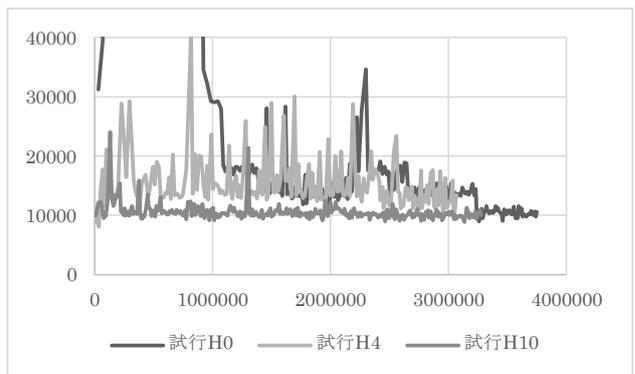


Fig.10 : Fig.9 を 40,000 ステップ以内に拡大した図

Fig.9, Fig.10 より、試行 H4 の累積ステップ数が試行 H0 より増加したのは、学習回数が不十分であったためにステップ数が収束しなかったためと考えられる。Fig.10 で分かるように、さらに学習回数を増やしたら従来の手法で収束したステップ数と同じ値に収束する可能性がある。また、試行 H10 の時に最も少ないステップで学習が進んでいることが分かり、学習回数 500,000 回あたりで 10,000 ステップに収束することが分かる。従来の手法で Q 学習を進めるのに比べて、人間による行動選択を加えた方が早く学習が進むということがいえる。

5 まとめ

本研究では、帆走の強化学習におけるエージェントの初期行動を人間が実際に操作したログから決定して学習を進める手法を提案し、従来の手法と比較した。その結果、人間による行動選択なしで学習を進めた場合は学習の収束に 150 エピソード（累積ステップにすると約 3,200,000 ステップ）ほど必要であるが、人間による行動選択を含めて学習を進めた場合はより少ない累積ステップ数で学習が進むことが分かった。特に、試行 H10 で学習を進めた場合の結果は、およそ 15 エピソードほどで収束し、従来の手法よりも早く学習が進んでいることが分かり、これは人間による行動選択が終わった時点で従来の手法に比べてより多くの行動において Q 値の更新がされたためだと考えた。

今回は試行 H0, 試行 H3, 試行 H4, 試行 H5, 試行 H10 と実験を行った。そのうち、試行 H4 の場合はステップ数が収束せず、学習回数が不十分であった可能性がある。試行 H3 と試行 H5 の場合は学習回数が十分あるにも関わらず、試行 H0 と比べて学習が進まない結果となったため、その原因を明らかにしていきたい。また、今後は人間による行動選択を x 座標にも拡張した 2 次元平面での学習についても研究を進めていきたいと考えている。

参考文献

- 1) Philip Jonathan Sterne : Reinforcement Sailing, 14-15, Master of Science Artificial Intelligence School of Informatics University of Edinburgh, (2004)
- 2) Richard S. Sutton and Andrew G. Barto : Reinforcement Learning, 159-161, 森北出版株式会社, (2000)

フォーメーション形成問題に対する 学習の確実性を向上させた群強化学習法とその評価

○ 飯間 等 黒江康明 (京都工芸繊維大学)

Swarm Reinforcement Learning Methods Improving Certainty of Learning for a Formation Problem and Their Evaluation

*H. Iima and Y. Kuroe (Kyoto Institute of Technology)

Abstract— In this paper, we treat a multi-robot formation problem. In this problem, multiple robots move from their respective initial positions, and they achieve to make a given target formation by reaching different goal positions each other. They learn their respective goal positions and the shortest routes to the goal positions. For solving the formation problem, we recently proposed swarm reinforcement learning methods. In the methods, multiple sets of the robots and environments, which are called learning worlds, are prepared and the robots in each learning world learn not only by performing a usual reinforcement learning method but also by exchanging information among the learning worlds. The methods, however, sometimes fail to find not only an optimal policy but also a policy which enables the robots to make the target formation, especially in the case where the target formation is large and complicated. In order to resolve this problem, this paper proposes swarm reinforcement learning methods in which the robots learn through taking actions based on the current policy in learning by the usual reinforcement learning method. The performance of the proposed methods is evaluated through numerical experiments.

Key Words: Reinforcement learning, Swarm intelligence, Formation control

1 はじめに

近年、マルチエージェントシステムのフォーメーション制御に関する研究が盛んに行われ、群ロボット、複数の無人飛行機、複数の人工衛星などへの応用が考えられている¹⁾。ところが、このようなシステムは非常に複雑で、その挙動を把握してモデルを作成し、それに基づき制御法を開発するのは一般に困難である。このようなシステムに対しては、エージェントが試行錯誤を通して学習を行う強化学習の導入が有効であると考えられる。なぜなら強化学習では環境に関する事前知識を必要としないからである。そこで、著者らは、フォーメーション制御の基礎的な問題として複数のロボットがフォーメーションを形成する強化学習問題を取り上げ、この問題に対する強化学習法に関する研究を進めている^{2, 3, 4)}。

フォーメーション形成問題は、複数のロボットが所望のフォーメーションを形成するために、各自の初期座標からロボットの台数分存在する目標座標までの最短経路を学習する問題である。各ロボットは互いに異なる目標座標に到達しなければならない、各自が到達すべき最適な目標座標を学習を通して獲得しなければならない。フォーメーション形成問題はマルチエージェント問題⁵⁾であり、一般にマルチエージェント問題においてはロボット間の相互作用が存在するので、特に学習が困難となる。また、ロボットは目標座標までの最短経路を学習したいので自身の初期座標に最も近い目標座標へ移動する方策を学習しようとするが、ロボット全体のことを考えると通常はその方策が最適ではない。このように全てのロボットにとって最適な方策を学習することは困難である。

フォーメーション形成問題に対して、著者らはエージェントと環境の組（これを学習世界と呼ぶ）を複数用意して学習を行う強化学習法を以前に提案しており、この学習法を群強化学習法と呼んでいる^{2, 3)}。群強化

学習法は、各学習世界で個別に通常の強化学習を用いて学習させるとともに、学習世界間の情報交換によっても学習させる方法である^{6, 7)}。個別学習で優れた学習を行った学習世界の情報を他の学習世界に受け渡して、各学習世界がその情報をもとに学習することにより、優れた方策を短い学習時間で得ることを期待している。優れた学習を行った学習世界の選別は各学習世界の個別学習結果を評価して行う。フォーメーション形成問題では各ロボットが目標座標までの最短経路を学習したいので、全ロボットの行動回数の総和に基づいて評価するのが妥当である。ところが、総和のみで評価すると、ロボットが自身の初期座標に最も近い目標座標に到達する方策を学習する可能性があり、そのような方策は通常最適ではない。最適な目標座標に最短経路で到達できている場合は全ロボットの行動回数のおぼろつきが小さいと考えられるので、文献^{2, 3)}では総和とおぼろつきの両方を用いて各学習世界の個別学習結果を評価する方法を提案している。数値実験において、目標とするフォーメーションが小さく複雑でない場合には、文献^{2, 3)}の学習法によって優れた方策を獲得できることを確認している。

ところが、文献^{2, 3)}の学習法では、目標とするフォーメーションが大きく複雑な場合には、最適な方策を求められないだけでなく、フォーメーションを形成する方策も求められない場合があった。そこで本稿では、最適な方策を求めることを多少犠牲にしても、フォーメーションを形成する方策を確実に求めることができるようにするための群強化学習法を提案する。提案方法をいくつかの例題に適用した数値実験を行い、提案方法の性能を評価する。

2 フォーメーション形成問題

Fig. 1 に示されるように、複数台存在する各ロボットが各自の与えられた初期座標から移動して、ロボットの

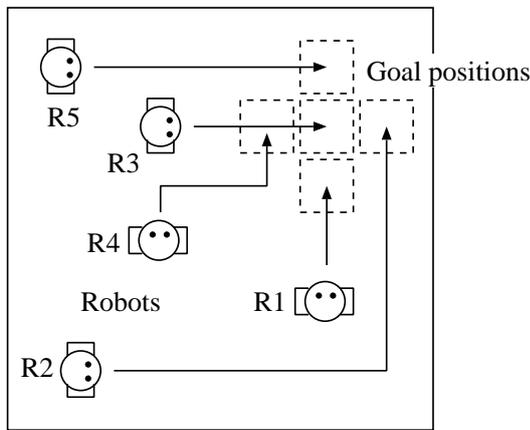


Fig. 1: Multi-robot formation problem

台数と同じ個数だけ存在する目標座標の一つに到達して何らかのフォーメーションを形成する問題を考える。各ロボットは互いに異なる目標座標に到達しなければならないものとする。このような状況は人文字のように複数のロボットが文字や絵を形成するような場合に現れる。このとき、全てのロボットが目標座標に到達してフォーメーションを形成するまでの時間が最短となる各ロボットの移動経路を求めたい。本稿では簡単のために、各ロボットが二次元格子平面上を移動するものとする。格子平面の大きさを $X \times Y$ とし、平面の左下の座標を $(1, 1)$ 、右上の座標を (X, Y) とする。ロボットの台数を I とすると、各ロボット R_i ($i = 1, 2, \dots, I$) は初期座標 S_i から出発し、 $\{G_1, G_2, \dots, G_I\}$ で表記される I 個の目標座標の一つに向かって移動する。各ロボットは上下左右への 1 マス分の移動および停止の中から一つの行動を選択できるが、二次元格子平面の外側には移動できず、そのような行動を選択した場合は移動せずに元の座標に留まるものとする。全てのロボットは同時に移動するが、他のロボットに接触しないようにするための処理を簡単のため次のように行うこととする。すなわち、全ロボットを番号 i の順に一台ずつ動かしてみたときに、別のロボットがいる座標に移動する行動を選択したロボットは移動せずに元の座標で留まるものとする。

以上に示したフォーメーション形成問題に対して、強化学習問題としての問題設定を説明する。ここでは、一台のロボットを一つの学習エージェントとみなす。各ロボット R_i は現在の自分の位置座標 (x_i, y_i) を観測することができるものとし、それを強化学習における状態とする。各ロボットの状態数は XY である。行動は上下左右への 1 マス分の移動および停止であり、各ロボットの行動数は 5 である。先に述べたように、全てのロボットは互いに異なる目標座標 $\{G_1, G_2, \dots, G_I\}$ に到達しなければならない。ロボット R_i が到達する目標座標を G_{O_i} ($O_i \in \{1, 2, \dots, I\}$) とするとき、 G_{O_i} はあらかじめ与えられず、各ロボットは学習を通して G_{O_i} を獲得するようにする。全てのロボットが目標座標に到達してフォーメーションを形成するまでの時間はロボットの行動回数に比例する。以上より、本強化学習問題は、全ロボットが互いに異なる目標座標に到達するまでの行動回数が最少となるように、各ロボッ

トの目標座標 G_{O_i} とそれに至る経路（行動手順）を学習する問題である。なお、全てのロボットが初期座標から出発してフォーメーションを形成するまでを 1 エピソードとする。

このようなフォーメーション形成問題はマルチエージェント問題であり、一般にマルチエージェント問題においてはロボット間の相互作用が存在するので、特に学習が困難となる。また、ロボットが初期座標に最も近い目標座標へ移動する利己の方策と他のロボットも最短経路をとることができる目標座標へ移動する利他的方策が異なるジレンマを有している。このことを Fig. 1 を用いて説明する。Fig. 1 において、初期座標が最も右に位置するロボット R_1 は自身に最も近い目標座標へ移動し、他の全てのロボットが目標座標へ到達するまで停止する行動をとっている。ロボット R_1 が自身に最も近い目標座標へ移動するので、左下のロボット R_2 は R_1 が移動する目標座標へは移動できず、より遠くの最も右に位置する目標座標へ移動している。ロボット R_2 は 5 台のロボットの中で最も経路が長いので、この場合のフォーメーションを形成するまでの時間は R_2 の移動時間に等しい。ロボット R_1 と R_2 の目標座標を逆にすると、先と同様に R_2 の経路が最も長いのでフォーメーションを形成するまでの時間は R_2 の移動時間に等しいが、その移動時間は Fig. 1 の場合より短くなる。したがって、ロボット R_1 と R_2 の目標座標を逆にした場合の方策はロボット全体の移動時間を考えた利他的方策であり、Fig. 1 の方策はロボット R_1 が自身の移動時間のことしか考えていない利己の方策である。これらの利他的方策と利己の方策が異なるジレンマによって利他的方策を学習することが困難となる。

3 フォーメーション形成問題に対する群強化学習法

前節のフォーメーション形成問題に対して、文献 2, 3) で提案した群強化学習法を概説する。

3.1 基本構成

強化学習では、エージェントが現在の状態で何らかの行動を実行して環境から次状態と報酬を受け取り、その受け取った報酬に基づいて最適な方策を学習する。群強化学習法では、エージェントと環境の組を学習世界と呼び、この学習世界を複数用意し、二種類の学習によって短時間に最適な方策を獲得しようとする。すなわち、各学習世界のエージェントが個別に通常の強化学習を行う学習と、他の学習世界との情報交換を通じた学習である。通常の強化学習法では一つの学習世界のみで学習を進めていくことに相当し、時として報酬が少ない無駄な行動をエージェントが選択する場合があります。一方、複数の学習世界を用意すると、いずれかの学習世界で何らかの意味で優れた学習が行われることが期待できる。その優れた学習を行った学習世界の情報を学習世界間で交換して、各学習世界のエージェントがその情報をもとに学習することにより、より短時間に最適な方策を得ることが期待できる。

群強化学習法の基本的な流れは次のようになる。まず各学習世界のエージェントが個別に通常の強化学習を行って学習する。次に、優れた学習を行った学習世界を選別するために、先の個別学習の結果に基づいて

各学習世界を評価する。その評価結果に基づいて優れた学習世界を選別し、優れた学習世界の何らかの情報を学習世界間で交換して各学習世界のエージェントが学習する。以上の個別学習、学習世界の評価および学習世界間の情報交換による学習を、エピソード数がある一定回数に達するまで繰り返す。

3.2 個別学習

各学習世界の個別学習で用いる学習法や行動選択法は基本的にはこれまでに提案されているどのような方法でも用いることができるが、以降では学習法として Q-learning 法⁸⁾を用い、行動選択法として ε -greedy 法を用いるものとして説明する。Q-learning 法は各状態 s 、各行動 a に対する価値 (Q 値) $Q(s, a)$ の推定値を用いて学習する方法であり、各学習世界 w の個別学習で用いる場合のアルゴリズムは次のようになる。

- Step 1 各ロボット R_i の現在の状態 s_{wi} を初期化する。
- Step 2 各ロボット R_i が後述する ε -greedy 法を用いて行動 a_{wi} を選択する。
- Step 3 各ロボット R_i が環境から次状態 s'_{wi} と報酬 r_{wi} を受け取る。
- Step 4 各ロボット R_i が状態 s_{wi} 、行動 a_{wi} に対する Q 値 $Q_{wi}(s, a)$ を次式で更新する。

$$Q_{wi}(s, a) \leftarrow Q_{wi}(s, a) + \alpha \{r_{wi} + \gamma \max_{a'} Q_{wi}(s', a') - Q_{wi}(s, a)\} \quad (1)$$

ここで、 α は学習率、 γ は割引率と呼ばれるパラメータである。

- Step 5 もしもエピソードの終了条件を満たしたならば、個別学習を終了する。そうでなければ、Step 2 へ戻る。

学習で得られた Q 値 $Q(s, a)$ より、ロボットの方策は各状態 s で行動

$$a^* = \arg \max_a Q(s, a) \quad (2)$$

をとるものとして与えられる。

ε -greedy 法は、確率 ε でランダムな行動を選択し、確率 $1 - \varepsilon$ で (2) 式で与えられる行動 a^* 、すなわち Q 値が最大の行動を選択する方法である。 ε は探査率と呼ばれるパラメータであり、その値はあらかじめ与えられる。Q 値が最大の行動を選択するのは現在の方策、すなわち個別学習が行われるエピソード時点における方策に関する Q 値をより正確に推定するためである。ところが、現在の方策に関する Q 値を推定してばかりいると他の Q 値を推定できず、それゆえ他の方策を探査することが困難となる問題を解決できない。この問題を解決する安易な方法として、ランダムに行動を選択することとしている。

3.3 学習世界の評価

2 節で述べたように、フォーメーション形成問題はジレンマを有しており、ロボット全体が最少の行動回数で目標座標に到達するためには、各ロボット R_i が向かうべき目標座標 G_{O_i} を適切に学習する必要がある。このために、各ロボットが適切な目標座標に向かっているかどうかを考慮して各学習世界を評価する。

学習世界の評価は文献 6) で提案した報酬割引和に基づいて行う。エピソード t における学習世界 w のロボット R_i の報酬割引和 $R_w(t)$ は次式で与えられる。

$$R_w(t) = \sum_{k=1}^{L_{wi}(t)} d^{L_{wi}(t)-k} r_{wik}(t) \quad (3)$$

ここで、 $L_w(t)$ と $r_{wik}(t)$ はそれぞれエピソード t の個別学習において学習世界 w のロボット R_i がフォーメーションを形成するまでに行動した回数と k 回目の行動で得られた報酬である。

報酬は目標座標までの行動回数が少ない場合になるべく多く与えられるように設定されるので、行動回数が少ない場合に報酬割引和が概ね大きくなる。したがって、全ロボットの報酬割引和の総和が大きい学習世界が優れていると評価するのが妥当である。ところが、総和のみで評価すると利己の方策、すなわち自身の初期座標に最も近い目標座標に到達する方策を学習する可能性がある。そこで、総和のみで評価を行うのではなく、各ロボットが利他的方策をとって適切な目標座標へ向かっているかどうかについても評価を行う。全てのロボットが利他的方策、すなわち最適な目標座標に最少回数の行動で到達できている場合は、各ロボットの報酬割引和のばらつきが小さいと考えられるので、報酬割引和の総和だけでなく、分散も用いて各学習世界を評価する。エピソード t の学習世界 w における全ロボットの報酬割引和の総和 $S_w(t)$ と分散 $V_w(t)$ は次式で与えられる。

$$S_w(t) = \sum_{i=1}^I R_{wi}(t) \quad (4)$$

$$V_w(t) = \frac{1}{I} \sum_{i=1}^I \left\{ \frac{1}{I} S_w(t) - R_{wi}(t) \right\}^2 \quad (5)$$

報酬割引和の総和 $S_w(t)$ と分散 $V_w(t)$ を用いて優れた学習世界を選別する方法を説明する。報酬割引和の総和 $S_w(t)$ が大きく、分散 $V_w(t)$ が小さい場合に、各ロボットが最適な目標座標に最少回数の行動で到達できていると考えられるので、まず $S_w(t)$ が全学習世界中で最も大きい学習世界と $V_w(t)$ が最も小さい学習世界を選別する。次に、これらの学習世界が同一であり、かつこの学習世界番号を w^* としたときに条件式

$$S_{w^*}(t) \geq \max_{t' < t, w'} S_{w'}(t'), \quad V_{w^*}(t) \leq \min_{t' < t, w'} V_{w'}(t') \quad (6)$$

を満たす場合に、学習世界 w^* が優れていると評価する。

3.4 学習世界間の情報交換に基づく学習

学習世界 w^* が優れていると評価されたときに、 w^* の Q 値 $Q_{w^*i}(s, a)$ を学習世界間で交換して各学習世界の Q 値 $Q_{wi}(s, a)$ を更新する。報酬割引和の総和 $S_w(t)$ が最も大きい学習世界と分散 $V_w(t)$ が最も小さい学習世界が異なっていたり、条件式 (6) を満たさなかったりする場合は、情報交換に基づく学習は行わない。

優れた学習世界の Q 値 $Q_{w^*i}(s, a)$ を用いて学習する方法として、BEST, AVE, PSO を文献 6) で提案している。BEST は優れた学習世界の Q 値で他の学習世界

の Q 値を置き換える方法であり、次式で Q 値 $Q_{wi}(s, a)$ を更新する。

$$Q_{wi}(s, a) \leftarrow Q_{w^*i}(s, a) \quad (7)$$

AVE は $Q_{wi}(s, a)$ と $Q_{w^*i}(s, a)$ の平均値で更新する方法であり、次式で Q 値 $Q_{wi}(s, a)$ を更新する。

$$Q_{wi}(s, a) \leftarrow \frac{Q_{wi}(s, a) + Q_{w^*i}(s, a)}{2} \quad (8)$$

PSO は多点探索最適化手法の一つである Particle Swarm Optimization⁹⁾ に基づく方法である。この方法では各学習世界が過去に学習してきた Q 値の中で最も優れた Q 値（自己最良 Q 値と呼ぶ）と、全ての学習世界が過去に学習してきた Q 値の中で最も優れた Q 値（全体最良 Q 値と呼ぶ）を記憶しておく。各学習世界 w の自己最良 Q 値 $P_{wi}(s, a)$ に関しては、現在の Q 値に対する報酬割引和の総和 $S_w(t)$ および分散 $V_w(t)$ を自己最良 Q 値に対する総和 S_w^P および分散 V_w^P とそれぞれ比較し、 $S_w(t) \geq S_w^P$ かつ $V_w(t) \leq V_w^P$ ならば自己最良 Q 値を現在の Q 値で置き換える。全体最良 Q 値 $G_i(s, a)$ は BEST や AVE で用いる優れた Q 値 $Q_{w^*i}(s, a)$ と同じである。全体最良 Q 値 $G_i(s, a)$ が置き換えられたとき、次式で Q 値 $Q_{wi}(s, a)$ を更新する。

$$U_{wi}(s, a) \leftarrow WU_{wi}(s, a) + C_1 R_1 \{P_{wi}(s, a) - Q_{wi}(s, a)\} + C_2 R_2 \{G_i(s, a) - Q_{wi}(s, a)\} \quad (9)$$

$$Q_{wi}(s, a) \leftarrow Q_{wi}(s, a) + U_{wi}(s, a) \quad (10)$$

ここで、 $U_{wi}(s, a)$ は $Q_{wi}(s, a)$ の変化量、 W , C_1 , C_2 は重みパラメータ、 R_1 , R_2 は 0 から 1 までの一様乱数である。

4 提案方法

3.2 項で述べたように、学習後に得られる方策では (2) 式で与えられる行動 a^* をとる。一方、個別学習中では確率 $1 - \varepsilon$ で方策に従った行動 a^* をとるが、確率 ε で方策とは無関係に選択した行動をとる。したがって個別学習において、現在の方策に従って行動するとフォーメーションを形成できない場合でも、方策とは無関係な行動をとることによりフォーメーションを形成できる場合がある。このような場合に、報酬割引和の総和 $S_w(t)$ が大きく、分散 $V_w(t)$ が小さくなると、このときの Q 値が優れた Q 値 $Q_{w^*i}(s, a)$ として選別され、全学習世界のロボットが情報交換を通して $Q_{w^*i}(s, a)$ を用いた学習を行ってしまう。これにより、最終的にフォーメーションを形成できない方策が得られる場合があり、目標とするフォーメーションを形成する方策を確実に獲得できる方法となっていないという問題がある。

この問題を解決するために、個別学習で常に現在の方策に従った行動を選択して学習を行うことにより、フォーメーションを形成できる方策を確実に学習する方法を提案する。ただし、常に全ての学習世界で現在の方策に従った行動を選択するようにすると、個別学習で他の方策を探索することが困難となる。そこで、現在の方策に従った行動を選択して学習する方法を部分

的に導入することとする。どのように部分的に導入するかについては 2 つの方法が考えられるので、それらの方法を用いた 2 つの群強化学習法（方法 1, 方法 2 とする）を以降で順に説明する。

4.1 方法 1

一部の学習世界のみで現在の方策に従った行動を選択して学習するようにする。すなわち、3.2 項で述べた ε -greedy 法における探索率を、現在の方策に従った行動を選択して学習する学習世界で $\varepsilon = 0$ とし、他の学習世界で $\varepsilon > 0$ とする。これにより、 $\varepsilon = 0$ とする学習世界の個別学習時にフォーメーションを形成できていて、報酬割引和の総和 $S_w(t)$ が大きく、分散 $V_w(t)$ が小さいなら、この学習世界の Q 値が優れていると正しく評価できる。この結果、最終的に得られる方策でもフォーメーションを形成できることが期待できる。

アルゴリズムを下記にまとめる。

- Step 1 一部の学習世界における探索率を $\varepsilon = 0$ とし、他の学習世界における探索率を $\varepsilon > 0$ とする。また、各学習世界の Q 値の初期値を与え、エピソード数を $t \leftarrow 1$ と初期化する。
- Step 2 学習世界 w ごとに、1 エピソードだけ個別に通常の強化学習法を実行して Q 値を更新する。
- Step 3 学習世界 w ごとに、Step 2 の個別学習で得た報酬に基づいて、割引報酬和の総和 $S_w(t)$ と分散 $V_w(t)$ を (4)(5) 式で求める。
- Step 4 Step 3 で求めた各学習世界の $S_w(t)$ と $V_w(t)$ を用いて優れた Q 値を選別し、その優れた Q 値を学習世界間で交換して全学習世界の全ロボットにおける全状態、全行動に対する Q 値を更新する。
- Step 5 $t \geq T$ ならばアルゴリズムを終了する。 $t < T$ ならば $t \leftarrow t + W$ として Step 2 へ戻る。ここで、 T はエピソードの総数、 W は学習世界の総数で、これらはあらかじめ与えられる。

4.2 方法 2

あるエピソード数より前のエピソードでは全ての学習世界の探索率を $\varepsilon > 0$ とし、そのエピソード数以降では全ての学習世界の探索率を $\varepsilon = 0$ として現在の方策に従った行動を選択して学習する。これにより、そのエピソード数以降の個別学習時にフォーメーションを形成できていれば、最終的に得られる方策でもフォーメーションを形成できることが大いに期待できる。

アルゴリズムを下記にまとめる。

- Step 1 各学習世界の探索率を $\varepsilon > 0$ とする。また、各学習世界の Q 値の初期値を与え、エピソード数を $t \leftarrow 1$ と初期化する。
- Step 2 方法 1 の Steps 2-4 を実行する。
- Step 3 $t \geq T^*$ ならば全ての学習世界の探索率を $\varepsilon \leftarrow 0$ とする。ここで、 T^* は探索率を 0 とするエピソード数で、あらかじめ与えられる。
- Step 4 方法 1 の Step 5 を実行する。

5 数値実験

前節で提案した群強化学習法を適用して提案方法の性能を評価する。

5.1 実験方法

ロボット数 I , 格子平面の縦横の長さ X, Y , ロボットの初期座標 S_i , 目標座標 G_i が下記のようになっている 4 種類の例題 1-4 を取り上げる.

・例題 1

$I = 5, X = Y = 6, S_1 = (4, 1), S_2 = (4, 2), S_3 = (4, 3), S_4 = (5, 1), S_5 = (6, 1), G_1 = (1, 4), G_2 = (1, 6), G_3 = (2, 5), G_4 = (3, 4), G_5 = (3, 6),$
最少行動回数 : 35

・例題 2

$I = 7, X = 5, Y = 8, S_1 = (3, 6), S_2 = (3, 7), S_3 = (3, 8), S_4 = (4, 6), S_5 = (4, 8), S_6 = (5, 6), S_7 = (5, 7), G_1 = (1, 2), G_2 = (1, 3), G_3 = (2, 1), G_4 = (3, 2), G_5 = (3, 3), G_6 = (3, 4), G_7 = (3, 5),$
最少行動回数 : 49

・例題 3

$I = 8, X = 7, Y = 6, S_1 = (5, 4), S_2 = (6, 4), S_3 = (7, 4), S_4 = (5, 5), S_5 = (6, 5), S_6 = (7, 5), S_7 = (5, 6), S_8 = (6, 6), G_1 = (1, 2), G_2 = (1, 3), G_3 = (1, 4), G_4 = (2, 1), G_5 = (3, 1), G_6 = (4, 2), G_7 = (4, 3), G_8 = (4, 4),$
最少行動回数 : 56

・例題 4

$I = 10, X = 8, Y = 7, S_1 = (6, 5), S_2 = (7, 5), S_3 = (8, 4), S_4 = (8, 5), S_5 = (6, 6), S_6 = (7, 6), S_7 = (8, 6), S_8 = (6, 7), S_9 = (7, 7), S_{10} = (8, 7), G_1 = (2, 1), G_2 = (2, 2), G_3 = (2, 3), G_4 = (2, 4), G_5 = (2, 5), G_6 = (3, 3), G_7 = (4, 3), G_8 = (3, 5), G_9 = (4, 5), G_{10} = (5, 5),$
最少行動回数 : 80

例題 1 は X 形, 例題 2 は J 形, 例題 3 は U 形, 例題 4 は F 形のフォーメーションを形成する. 例題 1, 2, 3, 4 の順にフォーメーションが大きくなり複雑となる.

報酬に関しては, 全てのロボットが目標座標に到達したときに各ロボットに正の報酬 10 を与える. それ以外の場合には負の報酬を与え, いずれかの目標座標にいるロボットには -1 , どの目標座標にもいないロボットには -2 を与える.

各例題に下記の 4 つの方法を適用して学習結果を比較する.

- ・方法 1 : 提案する群強化学習法 1
- ・方法 2 : 提案する群強化学習法 2
- ・方法 3 : 文献 2, 3) の群強化学習法
- ・QL : Q-learning 法

方法 1, 2 と方法 3 を比較することによって, 探査率 ε を 0 とする提案方法の性能を評価する. 群強化学習の個別学習は Q-learning を用いて行う. そこで, Q-learning 法と比較することで, 提案方法の群としての有用性を検証する.

学習中の各ロボットは ε -greedy 法を用いて行動し, 方法 1 で $\varepsilon = 0$ としない学習世界の探査率, 方法 3 および Q-learning 法の探査率は $\varepsilon = 0.2$ とした. 方法 2 で $\varepsilon = 0$ とするエピソード数は $T^* = \frac{3}{4}T$ とし, それ以前のエピソード数でも ε の値を 0.6 から 0 に徐々に減少させることとした. すなわち, エピソード t にお

Table 1: Cumulative frequency for the number of actions after the learning for Case 1

(a) $T = 1500$

Method	35	40	45	50	55	∞
1-BEST	16	92	99	99	100	0
2-BEST	8	76	89	96	100	0
3-BEST	20	83	96	99	100	0
1-AVE	17	90	100	100	100	0
2-AVE	12	68	91	99	100	0
3-AVE	24	93	100	100	100	0
1-PSO	19	85	95	100	100	0
2-PSO	16	84	94	99	100	0
3-PSO	24	93	100	100	100	0
QL	7	61	79	83	84	16

(b) $T = 3000$

Method	35	40	45	50	55	∞
1-BEST	16	90	100	100	100	0
2-BEST	11	66	87	98	100	0
3-BEST	21	89	100	100	100	0
1-AVE	21	91	100	100	100	0
2-AVE	17	80	93	98	100	0
3-AVE	26	93	99	100	100	0
1-PSO	28	93	96	100	100	0
2-PSO	22	88	96	100	100	0
3-PSO	29	93	100	100	100	0
QL	10	56	76	85	85	15

∞ : Number of policies under which robots cannot reach goal coordinates

ける探査率 $\varepsilon(t)$ は次式で与えられる.

$$\varepsilon(t) = \begin{cases} 0.6 \left(1 - \frac{4}{3T}t\right) & \left(t < \frac{3}{4}T\right) \\ 0 & \left(t \geq \frac{3}{4}T\right) \end{cases} \quad (11)$$

群強化学習法の学習世界数 W は $W = 3$ とし, 方法 1 では $\varepsilon = 0$ とする学習世界数を 1, $\varepsilon = 0.2$ とする学習世界数を 2 とした. (3) 式の割引率 d は $d = 0.9$ とした. 学習世界間の情報交換法は BEST, AVE, PSO の 3 種類を用い, PSO の (9) 式の重みパラメータ W, C_1, C_2 は $W = 0.2, C_1 = C_2 = 0.8$ と与えた.

学習を打ち切るエピソード数 T は 1500 および 3000 の 2 種類とした. 群強化学習法における学習を打ち切るエピソード数 T は全ての学習世界の総数である. 各学習法ではロボットの行動選択時に乱数が用いられる. 学習法の平均的な性能を評価するために, 各学習法を各例題に対して 100 回ずつ実行した.

5.2 実験結果と考察

Tables 1-4 は, 各学習法で得られた 100 個の方策を用いてロボットにフォーメーションを形成させるシミュレーションを行い, その 100 回分のシミュレーションにおいて全ロボットが目標座標に達するまでの行動回数の累積度数を示す. この表において, 例えば例題 1, 方法 1-BEST における行動回数 40 の累積度数は 92 と

Table 2: Cumulative frequency for the number of actions after the learning for Case 2

(a) $T = 1500$

Method	49	56	63	70	77	84	∞
1-BEST	11	51	79	82	83	83	17
2-BEST	6	40	78	97	100	100	0
3-BEST	11	46	70	74	77	77	23
1-AVE	10	48	76	77	78	78	22
2-AVE	8	34	81	97	99	100	0
3-AVE	13	48	66	70	72	72	28
1-PSO	11	34	61	84	93	94	6
2-PSO	7	29	75	97	100	100	0
3-PSO	12	45	60	63	64	64	36
QL	1	26	45	49	51	52	48

(b) $T = 3000$

Method	49	56	63	70	77	84	∞
1-BEST	14	70	86	97	98	98	2
2-BEST	11	49	87	99	100	100	0
3-BEST	14	69	90	93	93	94	6
1-AVE	14	78	95	97	97	97	3
2-AVE	13	51	89	100	100	100	0
3-AVE	16	72	93	94	94	94	6
1-PSO	14	75	92	97	98	98	2
2-PSO	13	48	86	100	100	100	0
3-PSO	16	72	88	90	91	91	9
QL	4	36	54	56	57	58	42

∞ : Number of policies under which robots cannot reach goal coordinates

なっているが、これは行動回数が40以下となる方策が100個中92個あることを示している。また、「 ∞ 」はフォーメーションを形成できなかった方策の数を示しており、1,500,000回移動してもロボットが目標座標に到達しないときにフォーメーションを形成できなかったと判定することとした。群強化学習法を用いる方法1-3では情報交換法としてBEST, AVE, PSOを用いるので、これらの方法の名称をハイフン(-)でつないだもので各学習法を表記している。例えば、方法1でAVEを用いる学習法を1-AVEと表記している。

まず、提案方法である方法1, 2と学習世界が一つの場合に相当するQ-learning法を比較する。各表の「 ∞ 」を除く行動回数の全ての列より、提案方法の方策数はQ-learning法のそれより概ね多くなっていることがわかり、それゆえ提案方法の方が行動回数の少ない優れた方策を学習できているといえる。また「 ∞ 」の列より、フォーメーションを形成できない方策数が、提案方法はQ-learning法より概ね少なくなっていることがわかる。

次に、方法1, 2と文献2, 3)の方法3を比較する。各表の「 ∞ 」の列より、方法3では、ロボット数が $I = 5$ とフォーメーションが小さく複雑でない例題1に対しては常にフォーメーションを形成できる方策が得られているが、ロボット数がこれより多い($I = 7, 8, 10$)のためにフォーメーションが大きくて複雑な例題2, 3, 4

Table 3: Cumulative frequency for the number of actions after the learning for Case 3

(a) $T = 1500$

Method	56	64	72	80	88	104	∞
1-BEST	4	33	64	76	78	78	22
2-BEST	2	21	66	89	98	100	0
3-BEST	5	33	66	73	75	75	25
1-AVE	4	30	60	66	68	69	31
2-AVE	2	27	69	93	99	100	0
3-AVE	4	27	55	67	67	67	33
1-PSO	2	24	55	75	82	94	6
2-PSO	1	23	66	86	98	100	0
3-PSO	4	21	48	60	62	62	38
QL	1	23	43	53	55	56	44

(b) $T = 3000$

Method	56	64	72	80	88	104	∞
1-BEST	9	56	87	96	98	98	2
2-BEST	5	35	78	96	100	100	0
3-BEST	11	59	87	92	95	95	5
1-AVE	14	59	92	95	96	97	3
2-AVE	5	33	81	96	100	100	0
3-AVE	15	65	87	93	95	95	5
1-PSO	10	60	91	95	97	97	3
2-PSO	6	40	85	99	99	100	0
3-PSO	15	55	82	91	91	91	9
QL	2	25	42	49	52	52	48

∞ : Number of policies under which robots cannot reach goal coordinates

に対してはフォーメーションを形成できる方策が得られない場合があることがわかる。特に $T = 1500$ の場合には、例題2, 3では20方策以上で、例題4では全ての方策でフォーメーションが形成できていない。これに対して提案方法では、例題1に対しては方法3と同じく常にフォーメーションを形成できる方策が得られており、例題2, 3, 4に対してはフォーメーションを形成できる方策数が方法3より概ね多くなっている。ただし、各表における行動回数の左端の列より、最適方策の数は方法3の方が概ね多いことがわかる。

次に、提案方法の中で方法1と方法2を比較する。各表の「 ∞ 」の列より、方法1では、例題2, 3に対してはフォーメーションを形成できない方策が得られる場合があり、例題4に対してはほとんどフォーメーションが形成できていない。これに対して方法2では、例題2, 3に対しては常にフォーメーションを形成できる方策が得られており、例題4に対しても $T = 3000$ の場合には常にフォーメーションを形成できる方策が得られている。ただし、各表における行動回数の左端の列より、例題1, 2, 3における最適方策の数は方法2より方法1の方が多。

以上をまとめると、方法1は方法3よりフォーメーションを確実に形成できる方法となっており、最適方策を獲得する回数も方法2より多い。方法2は最適方策を獲得する回数は方法1に劣るものの、必ずフォー

Table 4: Cumulative frequency for the number of actions after the learning for Case 4

(a) $T = 1500$

Method	80	90	100	110	120	180	∞
1-BEST	0	0	0	0	0	0	100
2-BEST	16	36	58	74	81	93	7
3-BEST	0	0	0	0	0	0	100
1-AVE	0	0	0	0	0	0	100
2-AVE	19	42	65	76	83	88	12
3-AVE	0	0	0	0	0	0	100
1-PSO	0	1	1	1	1	1	99
2-PSO	22	48	72	80	86	91	9
3-PSO	0	0	0	0	0	0	100
QL	0	0	0	0	0	0	100

(b) $T = 3000$

Method	80	90	100	110	120	180	∞
1-BEST	0	1	1	1	1	1	99
2-BEST	21	45	80	83	92	100	0
3-BEST	0	0	0	0	0	0	100
1-AVE	0	1	1	1	1	1	99
2-AVE	21	47	68	83	94	100	0
3-AVE	1	1	1	1	1	1	99
1-PSO	0	1	1	2	2	2	98
2-PSO	21	52	75	90	98	100	0
3-PSO	0	0	0	0	0	0	100
QL	0	0	0	0	0	0	100

∞ : Number of policies under which robots cannot reach goal coordinates

メーションを形成できる方法となっている。

6 おわりに

複数ロボットのフォーメーション形成問題に対して、所望のフォーメーションを形成できる方策を確実に獲得するための群強化学習法を提案した。提案方法では各学習世界が個別に行う学習において、一部の学習世界のロボットが現在の方策に従った行動のみを選択して学習するようになり、学習の後半で全ての学習世界のロボットが現在の方策に従った行動のみを選択して学習するようになり、フォーメーションを形成できる方策を確実に獲得しようとする。

数値実験結果より、提案方法によってフォーメーションを形成できる方策を獲得する回数が増えたことが確認された。一部の学習世界のロボットが現在の方策に従った行動のみを選択して学習する方法は、学習の後半で現在の方策に従った行動のみを選択して学習する方法より最適方策を獲得する回数は多いが、フォーメーションを形成できる方策を獲得できない場合があった。後者の方法は常にフォーメーションを形成できる方策を獲得できることが確認された。

後者の方法では現在の方策に従った行動のみを選択して学習し始めるエピソード数をあらかじめ与えておかなければならない。このエピソード数はある程度学習が収束したときとすればよいと考えられるので、TD誤差などの指標を用いて学習の収束を判定することに

より、このエピソード数をあらかじめ与えることなく適応的に決定できる可能性がある。このような群強化学習法を開発することが今後の課題である。

参考文献

- 1) 桜間, 東, 杉江: 分散的パターン選択によるマルチエージェントシステムのフォーメーション制御, システム制御情報学会論文誌, **25-3**, 45/53 (2012)
- 2) H. Iima and Y. Kuroe: Swarm Reinforcement Learning Method for a Multi-Robot Formation Problem, Proceedings of 2013 IEEE International Conference on Systems, Man and Cybernetics, 2298/2303 (2013)
- 3) 飯間, 黒江: フォーメーション形成問題に対する Particle Swarm Optimization に基づく群強化学習法, 第 57 回システム制御情報学会研究発表講演会講演論文集, 317-5 (2013)
- 4) 飯間, 黒江: フォーメーション形成問題に対する強化学習法—優れた状態行動価値の抽出に基づく方法—, 第 41 回知能システムシンポジウム資料, A13-1 (2014)
- 5) L. Busoniu and R. Babuska: Comprehensive Survey of Multiagent Reinforcement Learning, IEEE Transactions on Systems, Man, and Cybernetics - PART C, **38-2**, 156/172 (2008)
- 6) 飯間, 黒江: エージェント間の情報交換に基づく群強化学習法, 計測自動制御学会論文集, **42-11**, 1244/1251 (2006)
- 7) 黒江, 飯間: 群強化学習法, 計測と制御, **52-6**, 540/547 (2013)
- 8) C.J.C.H. Watkins and P. Dayan: Q-Learning, Machine Learning, **8-3**, 279/292 (1992)
- 9) J. Kennedy and R.C. Eberhart: Swarm Intelligence, Morgan Kaufmann Publishers (2001)

制約ペトリネット内でトークン進化が満たすべき必須な条件

鈴木 秀明 (情報通信研究機構)
巖佐 庸 (九州大学)

Prerequisite Conditions for Token Evolution in a Constraint Petri-net

*Hideaki Suzuki (NICT)
Yoh Iwasa (Kyushu University)

Abstract– Unification constraints of predicate logic or string rewriting rules of context free grammar can be represented by a Petri-net with the AND/OR tree structure. In this network, deductive inference or parsing can be formulated as a problem to find a solution subtree within which variables have consistent substitution. The paper focuses on a token-based heuristic method to solve such an ill-posed problem, and presents three prerequisite conditions to be satisfied by tokens which propagate and evolve in the network. Incorporating these requirements would enable us to design highly distributed algorithms for deduction or parsing.

Key Words: Constraint network, Deduction, Token, Evolution, Ill-posed problem

1 はじめに

演繹推論や構文解析等の問題でベースに用意される宣言的知識(論理プログラムや生成文法)は、通常、変数をノードに、制約をエッジに対応させることにより、制約ネットワーク (constraint network; CN)^{6,2)} の形に表わすことが出来る。その中で、最近鈴木ら^{11, 12, 15, 16, 18, 19)} によって提案された pKTN (ペトリネット型知識推移ネットワーク; Petri-net-based Knowledge Transitive Network) は、それまで考えられていた高レベルペトリネット^{4, 10, 7, 9, 5, 8, 3)} とは異なり、単一化制約をラベルなしアークで表現する方法となっており、次のような特徴を持つ:

- プレースとトランジションから成る2部グラフ構造を持ち、多変数制約も表現できる。
- プレースが、変数のみならず述語や関数記号をも明示的に表現する。
- ネットワーク自体が、述語プレースを OR ノードとする AND/OR 構造を持つ。

我々が通常、制約充足問題を制約プログラミング^{22,1)} で解く際、解は全ての制約(制約ネットワーク中の全てのエッジ)を満足するように決められる。しかしながら pKTN では、上の3番目の特徴のため、pKTN を展開して作られる半無限ネットワークの中の一部、部分ネットワークの中だけで制約を満たすことが要求される。

本論文はこのように制約ネットワークで「部分ネットワークの選り出し」と「辻褃合わせ」という2つの基準を同時に満たす解を求める問題に着目し、これに対処するための方法について考察する。特に、制約を表わすペトリネット上でトークン集団を進化させることにより、ヒューリスティックに解を求める ELISE (ELiminating Inconsistency by SElection; エリーゼ)^{13, 14, 12, 15)} という方法を取り上げ、上記の不良設定性に対処するため、トークンをどのように伝搬し、処理すべきかという問題について検討する。

以下まず、第2節で不良設定を含む制約ペトリネットの典型的な例として、pKTN、および生成文法を表わ

す AND/OR グラフについて述べた後、第3節で ELISE のアルゴリズムについてその概略を述べる。第4節でトークンが進化において従うべきいくつかの要件を洗い出し、第5節に結論を述べる。

2 制約ペトリネットの例 論理プログラム

[Fact] $Wife(Mary, Paul)$. (1a)

[Fact] $Child(Tom, Paul)$. (1b)

[Rule] $Child(x, y) \leftarrow Wife(y, z), Child(x, z)$. (1c)

[Goal] $\leftarrow Child(q, Mary)$. (1d)

を考える。ここに $Wife, Child$ は述語、 $Paul, Mary, Tom$ は定数、 x, y, z, q は変数である。このプログラムは節(1c)の頭部と本体に述語 $Child$ を共通に含むため、これを表わす pKTN (正確には、直接変換で得られる pKTN を展開したものの^{16, 18, 19)} は無限に続く木構造を持つ。Fig. 1(a) はその最初の3段を図示したもので、演繹解の有無は、あるシンボル代入(単一化)の下での展開 pKTN の真偽、即ち根を真とする部分グラフの存在の有無と等価であることが証明できる(健全性と完全性¹⁵⁾。

また別の例として、言語 $\{a^m b^n | m, n \geq 0\}$ を生成する文脈自由文法

$$\left. \begin{array}{l} S \rightarrow AB \\ A \rightarrow aA \\ A \rightarrow \epsilon \\ B \rightarrow bB \\ B \rightarrow \epsilon \end{array} \right\} \quad (2)$$

を考える。ここに S は初期記号、 A, B は非終端記号、 a, b は終端記号、 ϵ は空記号である。この文法による導出を表わす AND/OR 木は、非終端記号 A, B が生成規則の両辺にあるため、無限に続く木構造となる。Fig. 2(a) はその最初の3段に入力記号列 aab による制約を付加したもので、構文解析は、根から始まる部分木で、その上の非終端記号を無矛盾とするものを探索することと定式化される^{20, 21, 17)}。

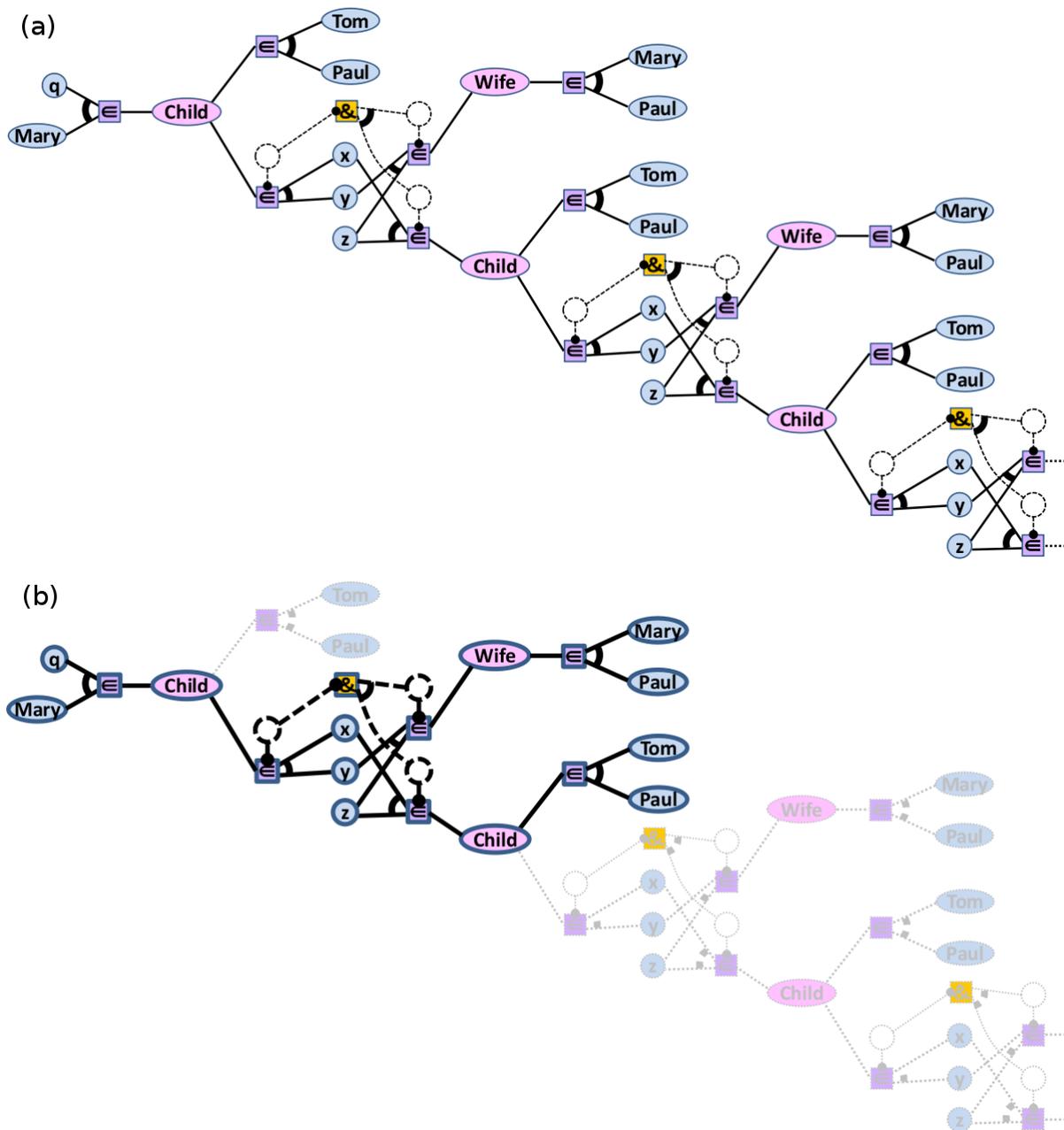


Fig. 1: 論理プログラム (1a)~(1d) から得られる展開 pKTN。無限に続く AND/OR 木の最初の 3 段を图示した。(b) の太線部分は無矛盾な制約を与える部分 (正解木、真枝)、またそれ以外の薄い点線部分は冗長な部分 (偽枝) である。図中、四角がトランジションを、円や楕円がプレースを表わす。円 / 楕円の中には定数名、変数名、述語名が表示されているが、定数名以外の名前は、推論には影響がない。トランジション中の \in は 'Split/Combine' を表わし、項をベクトル化したり分離したりする働きを持つ一方、 $\&$ は 'Logical AND' で、入力の論理積を出力する機能を持つ。

以上の例で、演繹推論 / 構文解析といった目標問題は、無限に続く展開 AND/OR 木の中で適切な部分木を見つけ、かつその中で無矛盾な記号割り当てを求めるといった不良設定問題に帰着する。次節では、このように定式化された問題を、ペトリネット上のトークン伝搬処理によって解く方法について議論する。

3 ELISE

ELISE (ELiminating Inconsistency by SElection; エリーゼ) は、データフロー・ネットワークやペトリネットといった分散ネットワークの中を流れるトークンを進化によって辻褃合わせし、埋め込まれた制約条件を満足する解を発見的に求める方法である。

周知のように、非同期処理のモデルとして見たペトリネットでは、ネットワーク中をトークンが流れ、そのトークンが運ぶ '値' が、各トランジションにおける

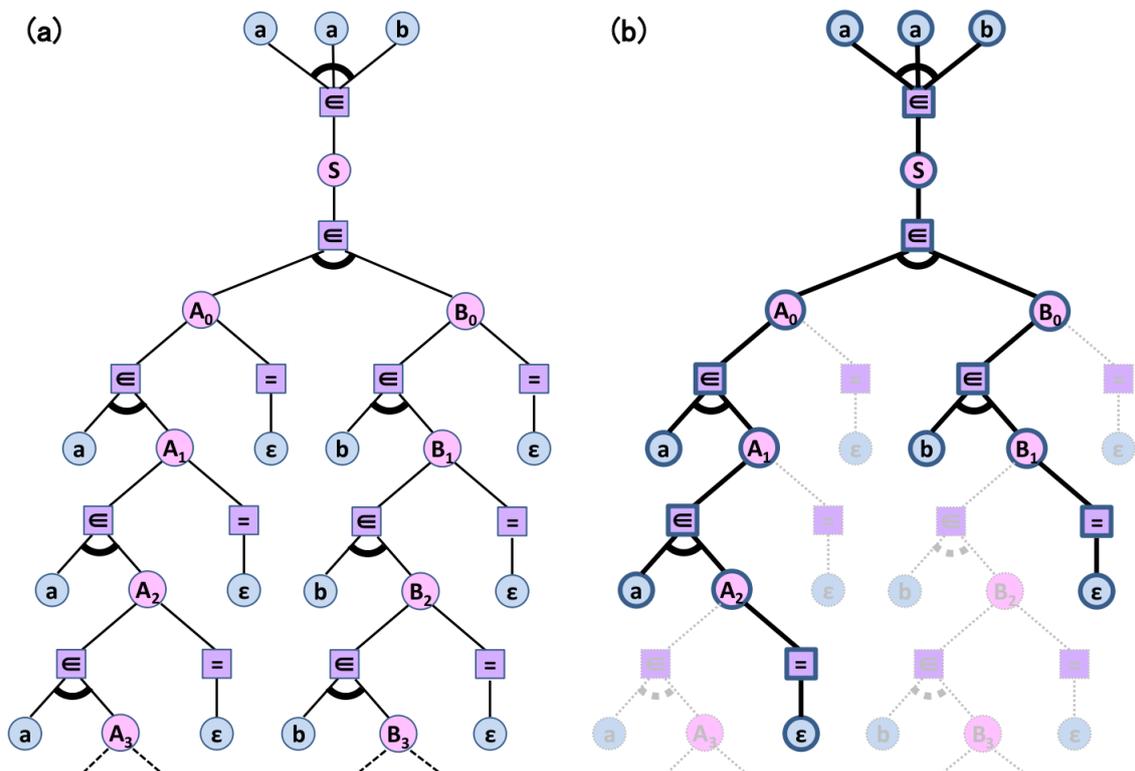


Fig. 2: 文脈自由文法 (2) を表わす AND/OR グラフ。無限に続く木の最初の 3 段と入力記号列 aab による初期記号への制約を図示した。(b) の太線部分は無矛盾な制約を与える部分 (正解木、真枝) で、またそれ以外の薄い点線部分は冗長な部分 (偽枝) である。グラフは四角をトランジション、円をプレースと考えることにより、ペトリネットと解釈できる。トランジションはそこから出るアークの本数により接続 (Concatenate; \in) または等価 ($=$) の機能を持つ。

発火で改変されることにより計算処理が進められる⁸⁾。ELISE ではこの‘値’ (x と表記) が表わす対象を拡張し、数値、シンボル、またはシンボルリスト (記号列) を表わすこととする。またそれと同時に、各トークンにはトークン間の比較で x に矛盾が無いほどより大きい値を持つように設定された実数値 r (reliability; 信頼度) が付与され、その r を適応度として、ネットワーク中の適当な箇所に設けられたプールの中でトークン進化が行なわれる。このようなトークンの生成・伝搬・進化を繰り返すことにより、トークン集団はネットワークに埋め込まれた制約条件に合致するものへと収束する。

これまで ELISE は数値連立方程式^{13,14)} やシンボル連立方程式^{12,15)} を表わすデータフロー・ネットワークに適用され、近似解法としての有効性が示されてきた。しかしながらこれらの実験で用いられた制約ネットワークには『冗長性』は含まれておらず、トークンは与えられた全ネットワークを制約の対象と仮定して、無矛盾性を追求しさえすればよかった。(数値連立方程式の場合は、いわゆる‘不能’の問題も扱われたが、条件の選び出しは対象とせず、近似解を得ることが目標とされた。) しかし、前節で与えた不良設定問題を ELISE で解く時、先の仮定は成り立たない。トークンは解を与える部分ネットワークを全ネットワークから選び出しながら、その部分ネットワーク内で値の辻褃合わせをしなければならない。もしこの抽出した部分ネットワークが大き過ぎたならば、値を無矛盾にすることは

できないだろうし、もしこの選ばれた部分ネットワークが小さ過ぎたならば、トークンは自明な解へと収束してしまうだろう。

4 トークン進化の要件

本節では、「無矛盾性」と「適切な制約の選び出し」という 2 つの要求を同時に求める問題を、ELISE のようなトークンベースの並列伝搬処理で解こうとする際に、トークン進化が満たすべき必要な要件を定式化する。

(1) 矛盾による $r =$ 小の情報、ネットワーク全体に知られること： トークン集団がネットワーク内に分散して存在する場合を考える。今あるプレース内のトークンにおいて、近傍トークンとの比較により x 値矛盾が検出され、そのトークンが小さな r 値を持つようになったとする。この時、もしトークン選択がローカルに設けられたプールの中で随時行なわれるとすると、このトークンは直ちに選択によって淘汰され、消失してしまう確率が高い。しかしながらこれはトークン集団全体の進化という観点からは都合が悪い。例えば Fig.1 の展開 pKTN の例で考える。このペトリネットでトークンの逆方向伝搬 (葉から根の向きの伝搬) が生じ、各 OR 分岐 (述語プレース) に、真枝と偽枝双方からのトークンが到着したとする。この内、偽枝由来のトークンは例えば図の z プレースの一つで矛盾を生むことになるが、その情報は最終解を与える q プレースに到着したトークンにとっても重要である。もしそ

の矛盾が生じたという情報が直ちに淘汰によってかき消されて、 q プレースのトークンに伝わらないとすると、 q プレースには偽枝由来のトークンがいつまでも残ることになりうる。このような事がないよう、ネットワーク内の一ヶ所で起きた x 値矛盾の情報は、そのトークンと無矛盾な x を持つネットワーク内の全てのトークンに伝えられ、共有される必要がある。

(2) 中一ヶ所矛盾と周辺矛盾が区別できること：前項の要件が満たされ、ネットワーク内の一ヶ所で起きた x 値矛盾をもとに、そのトークンと無矛盾な全てのトークンをネットワークから排除する仕組みが実現できたとする。しかしながらこの仕組みは、直ちに別の新たな問題を引き起こす。Fig.1(b) で太線表示された正解木について考える。この正解木上のプールが無矛盾な正解トークンで満たされた場合、その中の2つの *Child* プレースはそれぞれ、根元側は (*Tom, Mary*) のトークンで、根元から遠い側は (*Tom, Paul*) のトークンで占拠された状態になる。定数と矛盾する発火を抑制した場合、この前者のトークンが右上の偽枝上のトランジションの発火を引き起こすことはないのであるが、後者のトークンでは事情が異なる。後者のトークンの右下の偽枝上トランジションには、反対側に x, y という変数プレースが繋がっているだけであり、従ってこの偽枝上トランジションは、順方向発火で x プレースに *Tom* を、 y プレースに *Paul* を生成し続ける。これらは偽枝から生成されるトークンとの間に矛盾を生じさせ、それがもって正解トークンが排除される可能性が出てくる。このような正解木の周辺で起きる『周辺矛盾』と、前項に挙げた正解木内の『中一ヶ所矛盾』とを区別する仕組みがないと、正解トークンは安定して存続することができない。

(3) AND/OR 制約が加味できること：前項は正解木の周辺、特に OR 分岐で起きる問題であったが、この問題を含め、制約ペトリネットでは正解木が正しく抽出され安定に存続するためには、我々はペトリネットの持つ AND/OR 構造についての制約を正しく組み入れ、それに合わせてトークンの流れをコントロールしなければならない。ELISE のようなトークン伝搬がトランジション発火によって起こるモデルの場合、この要求はトランジション発火の制御（発火確率の調整）または個々のトークンが行なう伝搬経路の選択の形で実現される必要がある。

5 結論

述語論理知識や生成文法といった宣言的知識を表わす制約ペトリネットで、演繹推論/文脈自由文法の構文解析といった問題は、部分ネットワークの選出と、その部分ネットワーク上での記号の辻褃合わせを同時に行なう『不良設定問題』に定式化できることを述べた。また、このような問題を、多数のトークンを並列に伝搬させて進化させる方法でヒューリスティックに解くやり方について検討し、トークン進化に要求される必須な条件を定式化した。今後はこれら要件を満たす具体的方法を設計するとともに、実装実験を行なうのが課題となる。

参考文献

1) Barták, R.: Constraint Programming: In Pursuit of the Holy Grail. In: Proceedings of the Week of Doctoral Students

(WDS99), Part IV. MatFyzPress, Prague (1999) 555-564

2) Dechter, R.: From local to global consistency. Proceedings of the 8th CSCSI (Canadian AI Conference), Ottawa, Canada (1990) pp. 231-237 (best paper award) Artificial Intelligence **55**(1) (1992) 87-107

3) Jeffrey, J., Lobo, J., Murata, T.: A high-level Petri net for goal-directed semantics of Horn clause logic. IEEE Transactions on Knowledge and Data Engineering **8**(2) (1996) 241-259 DOI: 10.1109/69.494164

4) Jensen, K.: Coloured Petri Nets and the Invariant Method. Theoretical Computer Science **14** (1981) 317-336 North-Holland Publishing Company.

5) Lin, C., Chaudhury, A., Whinston, A.B., Marinescu, D.C.: Logical Inference of Horn Clauses in Petri Net Models. IEEE Transactions on Knowledge and Data Engineering **5**(3) (1993) 416-425

6) Montanary, U.: Networks of constraints fundamental properties and applications to picture processing. Information Sciences **7** (1974) 95-132

7) Murata, T., Zhang, D.: A predicate-transition net model for parallel interpretation of logic programs. IEEE Transactions on Software Engineering **14**(4) (1988) 481-497

8) 村田忠夫: ペトリネットの解析と応用. 近代科学社 (1992)

9) Peterka, G., Murata, T.: Proof procedure and answer extraction in Petri net model of logic programs. IEEE Transactions on Software Engineering **15**(2) (1989) 209-217 DOI: 10.1109/32.21746

10) Reisig, W.: Petri nets: an introduction. Springer-Verlag, New York, Inc. New York, NY, USA (1985)

11) 鈴木 秀明, 吉田 幹, 澤井 秀文: 演繹推論を実現するデータフローネットワークの提案. 人工知能学会研究会資料 人工知能基本問題研究会 (第 83 回) SIG-FPAI-B102 (2011) 1-7

12) Suzuki, H., Yoshida, M., Sawai, H.: A data-flow network that represents first-order logic for inference. In: Kuo, Y.H., Tseng, V.S.M., Kao, H.Y., Hong, T.P., Horng, M.F. (eds.): The 2012 Conference on Technologies and Applications of Artificial Intelligence TAAI, Proceedings (2012) 211-218 DOI: 10.1109/TAAI.2012.44

13) 鈴木 秀明: 逆算と進化により連立方程式を解くデータフロー・ネットワークの提案. 第 57 回システム制御情報学会研究発表講演会講演論文集 (SCI'13) (2013) 317-4

14) Suzuki, H., Iwasa, Y.: A network-based evolutionary method to solve inconsistent simultaneous equations approximately. AIP Conf. Proc. **1558** (11th International Conference of Numerical Analysis and Applied Mathematics - ICNAAM) (2013) 2486-2491 doi: 10.1063/1.4826045

15) Suzuki, H., Yoshida, M., Sawai, H.: A network representation of first-order logic that uses token evolution for inference. Journal of Information Science and Engineering (JISE) **30**(3) (2014) 669-686

16) 鈴木秀明, 吉田 幹: 述語を変数ノードに持つペトリネット型 KTN を用いた論理表現と演繹. 計測自動制御学会 (SICE) 第 41 回知能システムシンポジウム資料 (2014) B22-3

17) 鈴木秀明: 入力語にあいまい性を許す構文解析器の提案. 言語処理学会第 20 回年次大会発表論文集 (2014) 428-431

18) Suzuki, H., Yoshida, M.: Direct graphical representation of first-order logic for inference. In: Lambert, M.J. (ed.): Logic Programming: Theory, Practices and Challenges. Chap.4. Nova Science Publishers, Inc. (2014) 117-142 ISBN: 978-1-63117-853-5

19) 鈴木秀明: 制約ネットワークにおける不良設定問題を解く一手法. 計測自動制御学会 第 5 回コンピュータショナル・インテリジェンス研究会予稿 PG0008/14/0000-0037 (2014)

20) 白井良明, 辻井潤一: 人工知能. 岩波講座情報科学 **22** (1982)

21) 太原育夫: 人工知能の基礎知識. 近代科学社 (1988)

22) Van Hentenryck, P.: Constraint Satisfaction in Logic Programming. MIT Press, (1989)

ビジュアルサーボによる水中ロボットの位置・姿勢制御

○矢納陽 大西祥太 米森健太 石山新太郎 (岡山大学) 藤本勝樹 (広和株式会社) 見浪護 (岡山大学)

Position and orientation control of underwater vehicle system by visual servoing

*A. yanou S. Ohnishi K. Yonemori S. Ishiyama (Okayama University) K. Fujimoto (Kowa Corporation) M. Minami (Okayama University)

Abstract— A novel underwater vehicle system based on visual servoing technology with binocular wide-angle lens was developed to decontaminate radiation from mud in dam lake, river, regulation pond, sea and storage reservoirs in FUKUSHIMA. A real-time pose detection of target object through binocular camera system utilizing Genetic Algorithms (GA) for underwater vehicle is used, which is combined with servo controller, enabling to enhance stability performance of the vehicles pose relative to 3D marker. The following results were derived: (1) The proposed system kept the same high level of fitness value of GA as obtained at ground-based experiment. (2) The robustness of the proposed regulator system has been verified by exerting abrupt disturbance force. (3) A robust stability for 3D pose control in visual servoing process was confirmed.

Key Words: Autonomous underwater vehicle, Remotely operated vehicle, Visual servoing, Genetic algorithm

1 緒言

2011年3月11日の東北大震災以降、地震・津波・原子力発電事故が重なった福島県では、汚染地区での除染作業が進む中で、河川、ため池等の水中底泥の放射能汚染問題が新たに惹起されてきている。県内の農業用のダムやため池の底にたまる土や泥中の放射性物質の濃度を農林水産省や福島県が調べたところ約2,000か所のうち600か所で指定廃棄物の基準である1kg当たり8,000Bqを超える値が検出された¹⁾。そのため環境省や農水省では、これら指定廃棄物(原発事故に伴って発生した放射性物質を含む廃棄物のうち、濃度が1キログラム当たり8,000Bqを超えるもの)について国が責任を持って処理することとしている。しかし、これら水中底泥の除染回収においては、水の放射線遮蔽効果や水中での作業性の悪さなどから、その検出作業やダム池等100m以上の水深での除染回収作業は困難を極めるものと考えられる。そのため石山ら²⁾は、多機能センシング環境対応型制御知能MOS(Move on Sensing)の開発³⁾⁴⁾⁵⁾やMOS制御知能をROVに搭載してAUV(Autonomous Underwater Vehicle)化することより深水中の放射能汚染底泥の自動回収を目的としたMOS/AUV水中ロボット⁶⁾の開発を進めている。なお、本研究におけるMOS制御知能とは、カメラや放射線モニタなどのセンサから多元的に得た情報をもとに、Model-based-Matching法とGenetic Algorithmを利用して対象物をセンシングし、その結果に基づいてロボットを自律的に行動させるフィードバック制御方式と定義している。また、本方式で制御されるロボットをMOS制御知能化ロボットと称し、例えばMOS制御知能を搭載したAUVはMOS/AUVと表記している。このロボットで期待される性能は、(1)対象物の自動認識、(2)MOS制御知能ならびに無索化による自律航行⁷⁾⁸⁾、(3)高線量下での長期間連続的な除染回収作業、(4)高精度な放射線濃度分布計測、(5)自動給電、(6)複数のセンサー情報による深水環境情報(温度、濃度、流速等)の取得・分析などであり、著者らは地上における両眼カメラによるMOS搭載ロボット性能検証試験においてGA法(Genetic

Algorithm)を用いた3D Model-based-Matching法により、モデル対象物へのロボットによる物理的働きかけを0.5mmの精度で制御できることを示した³⁾。

これまで、水中ロボットの自律航行に関する研究はさまざまに行われている。吉田ら⁹⁾は水中ロボットの遠隔監視や制御に関する現状について報告しており、自律型水中ロボットは今なお発展途上の段階にあるとともに、今後の重要性について論じられている。例えば、鈴木¹⁰⁾、伊藤ら¹¹⁾は水中ロボットの開発について報告しているが、これらはいずれもロボットから送られてくる画像をもとに操作者が遠隔操縦することによってロボットの運動を制御するものである。また、柳ら¹²⁾はロボットが自律して行動できるようにするため人工水中ランドマークを利用したVision Systemによる手法を提案しているが、両眼カメラを利用した距離の計測はなされていない。

これらに対して、MOS/AUVにより水中でさまざまな作業(自動充電、生魚捕獲・観察等)を行わせる場合、対象物形状の正確な認識とともに作業環境中の外乱(流れ等)に抗するように対象物に対する正確な姿勢制御を自律的に行えなければならない。

そこで本研究ではROV水中ロボットに両眼カメラとMOS制御知能を搭載することによりAUV化することで水中における自律航行ならびに自律的水中作業を可能とするMOS/AUVの開発を行った。

本報では、両眼カメラから得られる画像とGA法から対象物との相対的位置・姿勢を認識する手法による、MOS/AUVと対象物との間の相対的位置・姿勢制御に関する実験結果について報告する。

2 実験方法

2.1 水中実験環境

2.1.1 水中ロボット系

水中実験で使用した遠隔操作型水中ロボット(広和(株)製ROV, 最大水深50m)をFig.1に示す。本ロボットは両眼視覚センサーとして、遠隔操縦用チルト機構付きカメラ1台(撮像素子CCD, 画素数38万画

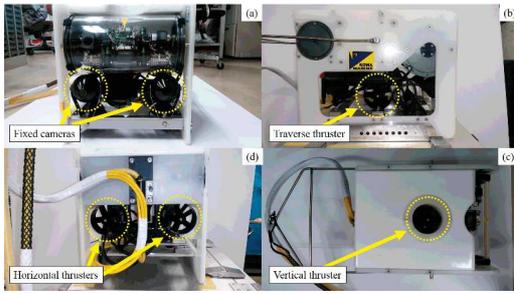


Fig. 1: Overview of ROV (a)Front view (b)Side view (c)Top view (d)Back view

素, 信号方式 NTSC, 最低被写体照度 1.5[lx], ズーム光学 10 倍) と制御用として前方 2 台および下方 2 台 (撮像素子 CCD, 画素数 38 万画素, 信号方式 NTSC, 最低被写体照度 0.8[lx], ズームなし) を搭載しており, MOS 制御で対象物の立体認識を行うため, 前方 2 台を同時使用した. 水中動力系では, 水平スラスト 2 基 (最大推力 9.8[N]), 垂直スラスト 1 基 (4.9[N]), 横スラスト 1 基 (4.9[N]) を搭載している. また, 照度確保のため LED ライト (5.8W) を 2 基搭載している.

2.1.2 水中実験系

水中実験では, 水道水を満たした簡易プール (縦×横×高さ, 2 m×3 m×0.75m) を実験槽として使用した. 上記 ROV を実験槽に水入させ, Fig.2 に示すようにテザーケーブル (200m) を介して ROV への給電ならびに実験槽脇に設置した PC を介して対象物の画像情報ならびに制御信号等の通信を行った. PC 側では ROV 搭載の 2 台のカメラから送られる対象物の立体像情報を基に, GA 法による 3D Model-based-Matching を行った.

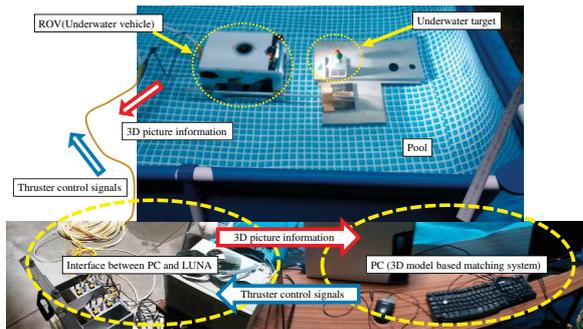


Fig. 2: Layout of under water experimental devices

また, 提案する制御系のブロック図を Fig.3 に示す. 提案するシステムでは, 両眼カメラから取得した画像を PC に転送し, Model-based-Matching 法と GA 法を組み合わせる対象物との相対位置・姿勢を認識させている. さらに, 認識結果と相対的目標位置・姿勢との誤差に基づく P 制御によって計算された指令電圧値を水中ロボットのスラストに与えることで, 推力を発生させている. なお, 本実験系における GA 法は, 33[ms] あたり 9 世代進化できることを予備実験で確認している.

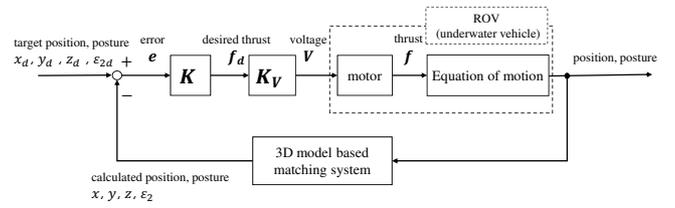


Fig. 3: Control logic for the proposed system

2.2 水中実験条件

2.2.1 GA 法による 3 次元対象物認識条件

両眼カメラを用いた GA の認識条件について, Fig.4 に示す. 赤・青・緑の三原色球 (直径 40mm) を取り付けた対象物を実験槽内任意位置に固定配置し, GA による認識を行う.

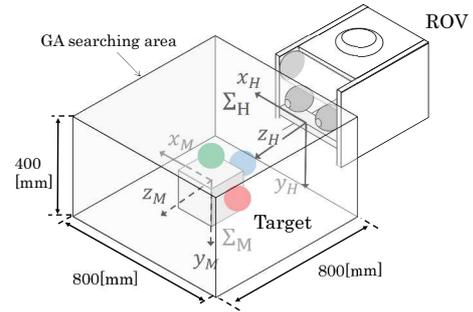


Fig. 4: Under water target and GA searching area

また, GA による 3D Model-based-Matching による対象物の位置・姿勢の認識情報は, 同図内に示す水中ロボットと対象物間で設定した GA 探索エリア内で獲得されると仮定している. GA の各条件を Table.1 に示す. ここで姿勢に関する変数 ϵ_2 は y 軸周りの回転角を意味している. また, 予備実験によって, 他の軸周りの回転は ROV の移動に依らず微小であったため, 本報ではこれらを GA による認識の対象としない.

2.2.2 水中制御安定性実験

任意位置に水中固定設置した対象物を GA により認識後, MOS 制御知能により水中ロボットに搭載した 4 基のスラストへ指令電圧値を送ることで, 水中ロボットと対象物間とで以下の相対的目標位置・姿勢 (x_d [mm], y_d [mm], z_d [mm], ϵ_{2d} [deg]) を保つように水中で制御を行い, その性能の検証を行った. Fig.5 に本研究の実験における座標系を示す.

$$\begin{aligned} x_d &= {}^H x_M = 0 \\ y_d &= {}^H y_M = -67 \\ z_d &= {}^H z_M = 600 \\ \epsilon_{2d} &= 0 \end{aligned}$$

また, x [mm], y [mm], z [mm], ϵ_2 [deg] を GA で認識した水中マーカーの位置・姿勢とおき, 相対的目標位置・姿勢に水中ロボットを一致させることを目指し, 以下の P 制御から計算される指令電圧値を各スラストへ与えた.

$$\text{奥行き方向} : v_1 = k_{p1}(z_d - z) + 2.5 \quad (1)$$

Table 1: parameters of GA

Number of genes	60
Target variables calculated by GA	Positions($x[mm], y[mm], z[mm]$), Posture($\epsilon_2[deg]$)
Search area (as centering the target in this area) [mm]	$\{x, y, z\} = \{\pm 400, \pm 200, \pm 400\}$
Control period [ms]	33
Number of generation evolution in the control period	9

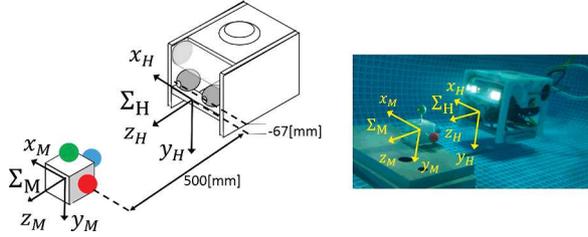


Fig. 5: Coordinate system provided in under water experiment

$$\text{鉛直軸回転} : v_2 = k_{p2}(\epsilon_2 d - \epsilon_2) + 2.5 \quad (2)$$

$$\text{鉛直方向} : v_3 = k_{p3}(y_d - y) + 2.5 \quad (3)$$

$$\text{右水平方向} : x_d - x < -5 \text{ (オン/オフ制御の条件式)}$$

$$\text{左水平方向} : 5 < x_d - x \text{ (オン/オフ制御の条件式)}$$

さらに、外乱に対するロバスト安定性を検証するため、制御中の水中ロボットに対して実験槽外部から物理力による外乱を加え、元の位置・姿勢へ復元できるかの確認も行った。

3 結果と考察

3.1 対象物の GA 法による認識精度

3.1.1 スラスト出力制御

遠隔操作型 ROV ではコントローラジョイスティック指示による作動操作時において、人間の指の動きによる誤動作を防止するためスラスト推進制御信号発信時にある程度の冗長性（不感帯）を持たせている。これに対し本研究では、mm 単位での高精度の移動ならびに姿勢制御性能の実現を目指している。そこで、ここでは各スラスト動力への推力-制御電圧変化特性における従来の不感帯の制御電圧に対して、推力の計算が容易にできるように制御電圧と推力の近似式を求めた。具体的には、予備実験で確認した ROV の不感帯特性を取り除き Fig.6 破線のように直線近似して推力の計算を行う。調整前はニュートラル状態（制御電圧 2.5V）近傍の電圧変化に対して出力応答に不感帯があったが、調整後は推力-制御電圧間の近似式に基づき、推力の計算が容易にできるようになった。なお、ROV の x_H 軸方向に関する推力は、正方向、負方向用にオン/オフ制御がそれぞれ実装されており、各方向ともオン時には 4.9[N] の推力が発生し、オフ時には推力は発生しない。

3.1.2 水中における GA 認識精度

Fig.7(a) に $x_d = 0, y_d = -67, z_d = 600, \epsilon_{2d} = 0$ を目標位置・姿勢とした水中ロボットの GA 認識時における適合度の時間変化を示す。これによると、認識開始から数秒以内で適合度が 1 以上となるとともに、

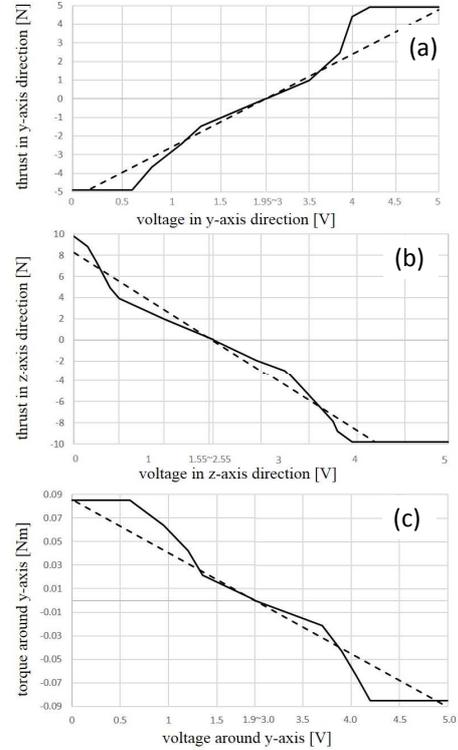


Fig. 6: Initial characteristics of thrust and torque control voltage and adjusted ones by removing dead-band and linearization: (a) characteristics removing dead-band (solid line) and adjusted one (broken line) in y-axis direction, (b) characteristics in z-axis direction and (c) characteristics around y-axis

適合度計測時間中において、ほぼ 0.8 以上の高い適合度を維持していることがわかる。一般的にビジュアルサーボにより精度よく対象物への物理的作業を行う場合、GA 認識精度は 0.5 以上必要であると考えられおり、宋薇ら¹³⁾¹⁴⁾¹⁵⁾¹⁶⁾¹⁷⁾は、本実験に使用した同一対象物を用いて室内における両眼認識精度の測定を行っている。これら両眼により陸上で得られた適合度の結果と今回水中で得られた結果を比較しても両者において同程度の適合度が得られていることがわかる。すなわち GA による水中での対象物認識精度は、気中における場合との比較においてもほぼ同程度の適合度が得られることがわかった。

3.1.3 位置・姿勢制御

実験槽外から物理的に外乱を与えなかった場合の制御結果を Fig.7 に示す。GA の認識誤差、ロボット移動時に生じるテザーケーブルからの反力、さらに移動時に発生する水圧変化による実験槽側面からの反射波を

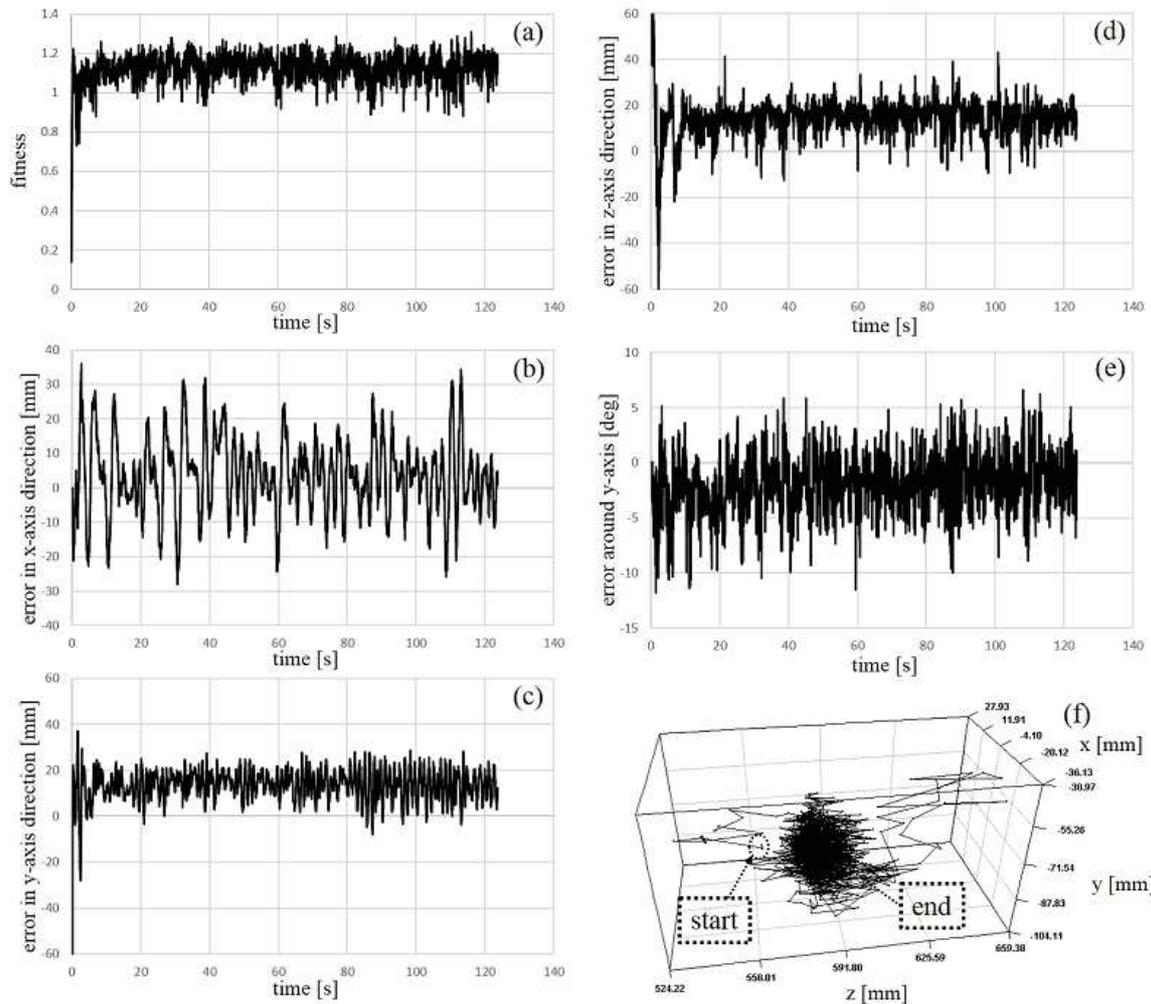


Fig. 7: Control result without additional disturbance: (a)fitness value, (b)error in x-axis direction, (c)error in y-axis direction, (d)error in z-axis direction, (e)error around y-axis, (f)3D trajectory of underwater vehicle

受け、相対的目標位置・姿勢からの誤差が定常的に現れるものの、対象物に対する相対的位置・姿勢を維持するMOS制御方式では4基のスラスタを常時同時稼働させることでこれら外乱要素をキャンセルし、相対的目標位置・姿勢近傍に水中ロボットの位置・姿勢を維持できることが確認できる。なお、Fig.7(b)に現れる誤差の緩慢な振る舞いは水中ロボット側面に設置されているスタビライザー板の抵抗による横スラスタの推力不足にその主要因があると考えられる。また、各目標値に対する定常誤差は、P制御によって与えられる推力と釣り合っている状態を意味している。

3.1.4 外乱に対するロバスト安定性

ここでは対象物を認識して位置・姿勢制御中の水中ロボットに対して、障害物との衝突や海流等の外乱が加わった場合のロバスト安定性を検証するため、実験槽外から棒(木製全長2m)を介した人力により制御中のロボットの位置を恣意的にずらすような外力を横方向(Fig.8(a))、下方向(Fig.8(b))、前方(Fig.8(c))、鉛直軸周り(Fig.8(d))にそれぞれ加え、水中ロボットが対象物との相対的目標位置・姿勢に復元するか確認を行った。

横方向、下方向、前方および鉛直軸周りに外乱を加えた場合の制御結果をFig.9~Fig.12に示す。各図とも

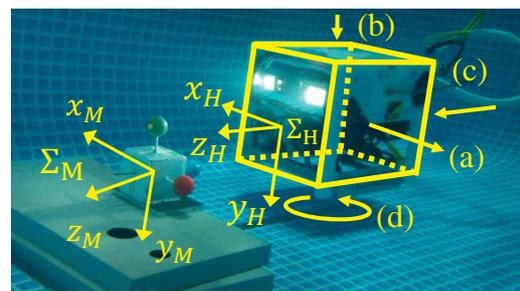


Fig. 8: Additional disturbance direction for Regulator performance test: (a)disturbance in x-axis direction, (b)disturbance in y-axis direction, (c)disturbance in z-axis direction and (d)disturbance around y-axis

(a)がGA認識の適合度、(b)は各変数についての水中ロボットが認識した対象物の位置(姿勢)、(c)は各変数についての相対的目標位置・姿勢と水中ロボットが認識した対象物の位置・姿勢との誤差、(d)はスラスタに与えた推力(トルク)を表している。各図とも実験開始から20[s]および60[s]あたりで外乱が加えられている。それぞれの図(a)(b)(c)(d)に記した(A)および(B)の区間において、外乱が加えられた際に適合度

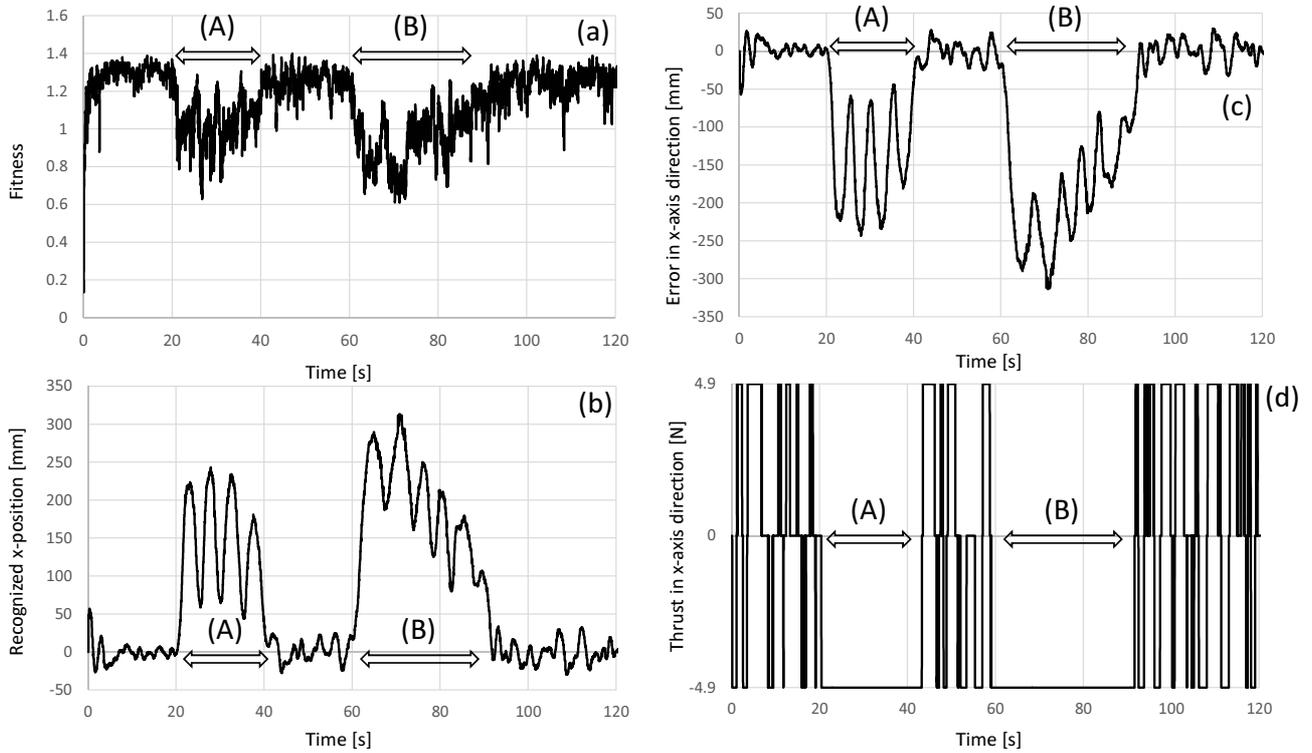


Fig. 9: Control result with disturbance in x-axis direction: (a)fitness, (b)recognized x-position, (c)error in x-axis direction and (d)thrust in x-axis direction

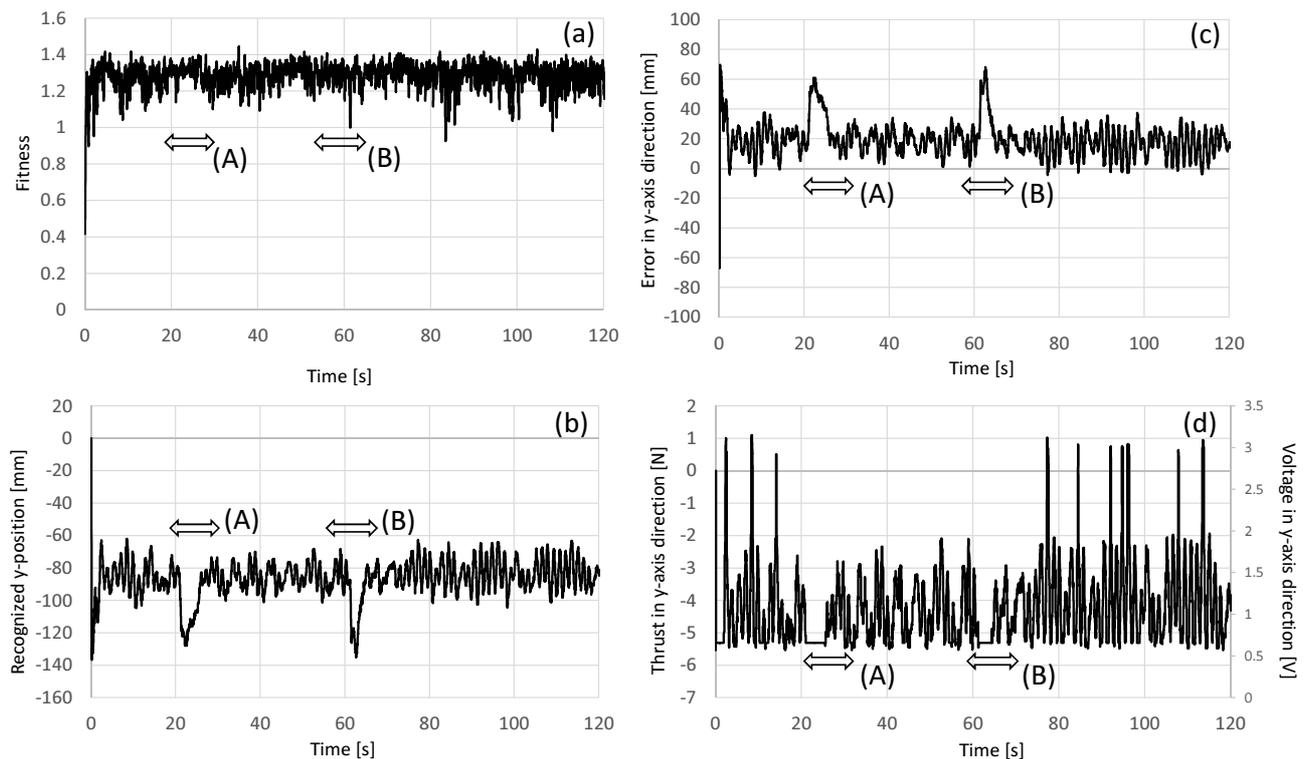


Fig. 10: Control result with disturbance in y-axis direction: (a)fitness, (b)recognized y-position, (c)error in y-axis direction and (d)thrust in y-axis direction

が一時的に低くなるが、ビジュアルサーボを維持しつつ、相対的目標位置・姿勢からの誤差に応じてスラストに与えられる推力（トルク）が変化することが分かる。すなわち、誤差を修正するための操作量が発生し、

その結果として相対的目標位置・姿勢に復元するよう水中ロボットの位置・姿勢が変化していくことを確認できる。以上より、提案システムではこれら全ての外乱に対して数秒～数10秒以内に元の位置に復元でき、

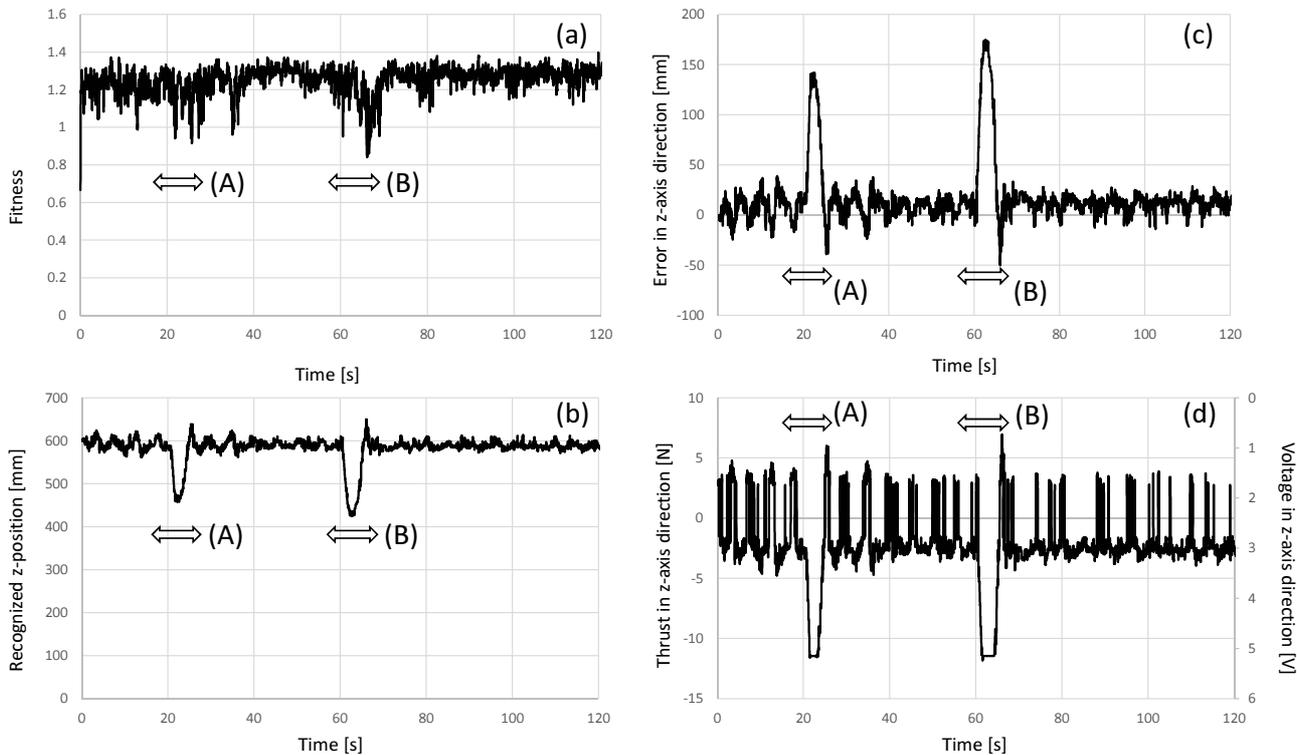


Fig. 11: Control result with disturbance in z-axis direction: (a)fitness, (b)recognized z-position, (c)error in z-axis direction and (d)thrust in z-axis direction

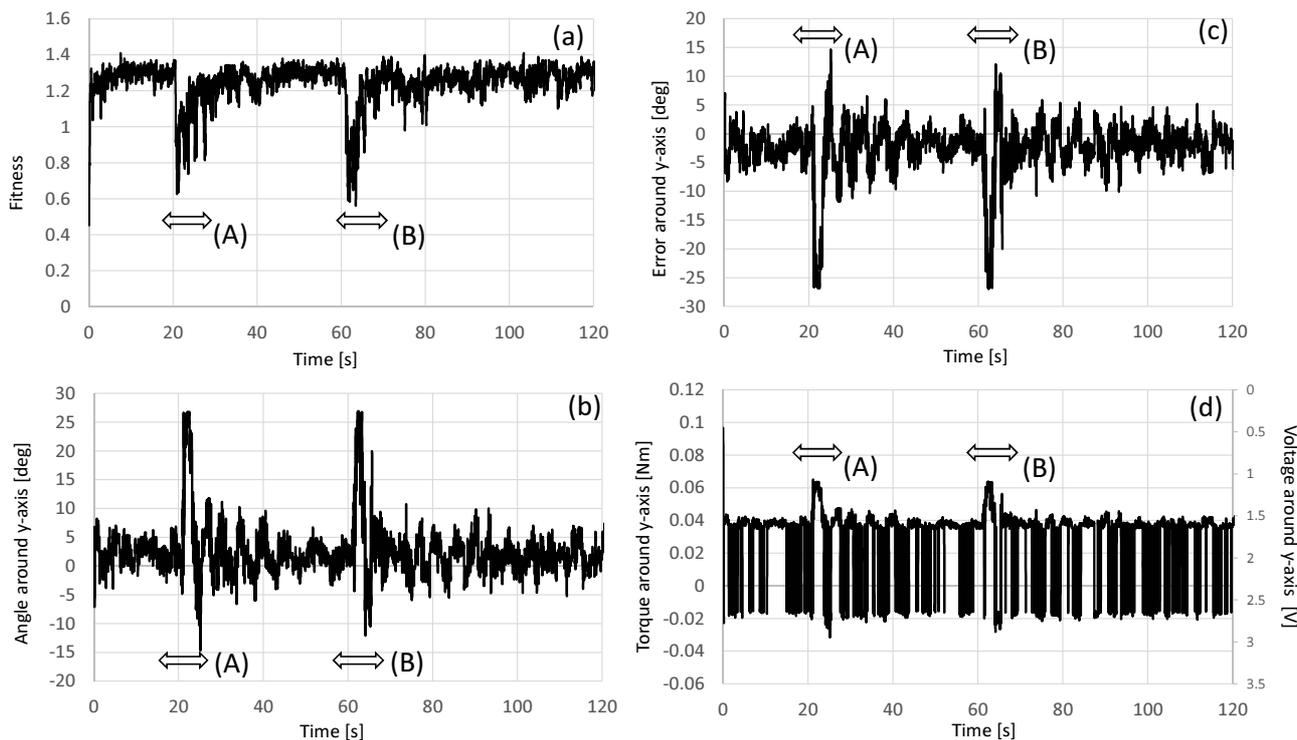


Fig. 12: Control result with disturbance around y-axis: (a)fitness, (b)recognized angle around y-axis, (c)error around y-axis and (d)torque around y-axis

外乱に対するロバスト安定性も有することを確認できた。ただし、両眼カメラの映像から対象物が完全に外れるような大きな変位を加えられた場合、ビジュアルサーボが切れてしまうため、水中ロボットが対象物との相対的目標位置・姿勢に制御出来なくなる可能性が

ある。

4 まとめ

MOS 制御知能を ROV に搭載することで、これまで人間が担っていた役割の代行を目指した水中ロボット

の研究開発を進めている。本報告では両眼カメラから取得した画像と GA 法を用いて、対象物に対して定位置・姿勢制御を行った結果を示した。また、提案するシステムは定位置・姿勢制御だけでなく、海流や障害物との衝突を想定した物理的外乱に対して、ロバスト性を有することも確認した。

参考文献

- 1) 農林水産省,「福島県内におけるため池中の放射性物質に関する実態と対策について～中間とりまとめ概要～平成 25 年 4 月農林水産省農村振興局」<http://www.maff.go.jp/j/kanbo/joho/saigai/pdf/tyukan.pdf> (2013).
- 2) 石山新太郎, 山口克巳, サーベイメータ搭載遠隔操作無人海中探査機による調整池汚泥の動態調査, 日本機械学会論文集 C 編 79 巻 799 号 718/725 (2013) .
- 3) 日本経済新聞, 汚染土壌ろ過用布効果を自動化, Science and Tech フラッシュ, (2014 年 7 月 8 日発刊) .
- 4) 石山新太郎, 山下学 (連載) 最新のセンシング/マニピュレーション融合化技術(2)放射能汚染土壌の洗浄再生化技術 (第 2 報), 「検査技術」(日本工業出版), Vol.19, No.4, 4月号, 45/49 (2014a) .
- 5) 石山新太郎ら, (連載) 最新のセンシング/マニピュレーション融合化技術(3) Move on Sensing 技術開発の現状 (第 3 報), 「検査技術」(日本工業出版) Vol.19, No.5, 5月号, 57/69 (2014b) .
- 6) 石山新太郎, (連載) 最新のセンシング/マニピュレーション融合化技術(1)水中サーベイメータ搭載遠隔操作無人海中探査機による水中汚泥の動態調査技術 (第 1 報), 「検査技術」(日本工業出版), Vol.19, No.3, 46/53 (2014c).
- 7) 柳善鉄, 浦環, 藤井輝夫, 水中ランドマークを利用した水中ロボットの航法制御, 生産研究, 52 巻, 5 号, 247/250 (2000) .
- 8) 近藤逸人, 浦環, 能勢義昭, 自律型海中ロボット“Tri-Dog 1”の設計とミッション, 生産研究, 52 巻, 5 号, 243/246 (2000) .
- 9) 吉田弘, 百留忠洋, 石橋正二郎, 越智寛, 志村拓也, 水中ロボットにおけるテレロボティクス, 日本ロボット学会誌, Vol.30, No.6, 578/581 (2012).
- 10) 鈴木正憲, 原子力発電プラント水中検査用 ROV の開発, 日本ロボット学会誌, Vol.22, No.6, 697/701 (2004).
- 11) 伊藤智之, 木村元比古, 小型水中点検ロボットの開発, 日本ロボット学会誌, Vol.22, No.6, 702/705 (2004).
- 12) 柳善鉄, 浦環, 藤井輝夫, 近藤逸人, 人工水中ランドマークと推測航法を利用した自律型水中ロボットの航法, 日本ロボット学会誌, Vol.20, No.3, 290/298 (2002).
- 13) 宋薇, 見浪護, 青柳誠司, クォータニオンを用いたフィードフォワードオンラインポーズ遺伝的認識法, 日本ロボット学会誌, Vol.28, No.1, 55/64 (2010a).
- 14) 宋薇, 見浪護, フィードフォワード遺伝的認識法を用いた 3-D ビジュアルサーボ, 日本ロボット学会誌, Vol.28, No.5, 591/598 (2010b).
- 15) Luca, A., Oriolo, G. and Giordano, P. R., On-line Estimation of Feature Depth for Image-Based Visual Servoing Schemes, Proceedings of 2007 IEEE Int. Conf. on Robotics and Automation (ICRA2007), 2823/2828 (2007).
- 16) 橋本浩一, 木村英紀, ビジュアルサーボイング-非線形オブザーバアプローチ-, 日本ロボット学会誌, Vol.13, No.7, 986/993 (1995).
- 17) Yu, F., Minami, M., Song, W., Zhu, J. and Yanou, A., ON-LINE HEAD POSE ESTIMATION WITH BINOCULAR HAND-EYE ROBOT BASED ON EVOLUTIONARY MODEL-BASED MATCHING, Journal of Computer and Information Technology, Vol.2, No.1, 43/54 (2012).

セパラトリクスを実現する 発現パターンに基づく遺伝子ネットワークの設計 —方法と解の存在性—

○森禎弘 黒江康明 (京都工芸繊維大学)

Expression Pattern Based Synthesis of Gene Regulatory Networks for Realizing Separatrices — A Method and Existence of Solutions—

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

Abstract— Recently, synthesis of gene regulatory networks having desired behavior has become of interest to many researchers and several studies have been done. Synthesizing simple gene regulatory networks having various behavior is expected for understanding functions of gene regulatory networks. In this paper we consider a synthesis problem of gene regulatory networks in which desired behavior are given by expression pattern sequences. In order to realizing various behavior, synthesis methods of gene regulatory networks whose dynamics has separatrices are needed. First, we show that there exists a solution of the synthesis problem for any desired behaviors requiring realization of separatrices. Second, we show a synthesis method based on the analysis of existence of solutions.

Key Words: Gene regulatory network, Synthesis problem, Separatrix

1 はじめに

遺伝子の発現機構を調べることは生物の仕組みを理解する上で重要であり、その調整機構である遺伝子ネットワークを対象とした研究が様々な観点から盛んに行われている。所望の機能を持つ遺伝子ネットワークを人工的に設計、実現する研究もその一つである^{1, 2, 3, 4, 5, 6, 7, 8, 9}。これには二つの背景がある。一つは、細胞の制御を実現するための最初のステップになることである。もう一つは、遺伝子発現機構の調整機能を解明するための構成論的アプローチとなることである。このような背景のもと、遺伝子ネットワークのどの遺伝子が発現しているかを表す発現パターンに着目し、所望の動作を発現パターンの変化を表す発現パターン遷移列とした遺伝子ネットワークの設計問題についていくつかの研究がなされている^{3, 4, 5, 6, 7, 8, 9}。

筆者らは、これまでに発現パターンに着目してその変化を表す発現パターン遷移列を所望の動作としたときの設計法を提案している⁶。発現パターン遷移列は発現パターンがどの順で遷移するかのみを表しており、遷移の時刻などの連続時間領域における情報は含まれていない。一方、設計に用いるモデルが区分的線形微分方程式モデルの場合、発現パターンの遷移時刻などを所望の動作として与えることができる。そこで、発現パターン遷移列に加えて発現パターンの所望の遷移時刻が与えられるとしたときの設計法⁸などを提案している。また、そのような遺伝子ネットワークの設計問題において任意の所望の動作をもつ遺伝子ネットワークのパラメータが存在することを示している⁹。

以上においては、所望の動作を表す発現パターン遷移列において、一つの発現パターンからの遷移先発現パターンは一つしか存在しないとしている。人工遺伝子ネットワークの構成や遺伝子ネットワークの機能理解に対する構成論的アプローチにおいて、より単純なモデルでネットワークの多彩な振舞いを実現できる方

が望ましいと考えられる。そのため、所望の発現パターン遷移列に一つの発現パターンからの遷移先が複数存在する場合の遺伝子ネットワークの設計法を提案している⁵。一つの発現パターンから複数の異なる発現パターンへの遷移が可能とすると、対象とする遺伝子ネットワークのモデルの解軌道を考えてときに、同じ発現パターンが現れる状態空間の領域が、次にどの発現パターンが現れる領域に解軌道が向かうかによって分割される。このような意味で、複数の発現パターンへ遷移可能な遺伝子ネットワークはセパラトリクスをもつといえる。本稿では、あるクラスの関数を相互作用関数として用いるとし、二つの異なる発現パターンへ遷移可能とするセパラトリクスを実現することで設計できるような二つの所望の発現パターン遷移列をもたせる遺伝子ネットワークの設計問題を考える。そして、任意の二つ所望の発現パターン遷移列に対して解が存在することを示す。設計問題の解の求め方を示すことで解の存在性を示しているため、解の存在性の証明が設計法にもなっている。

2 遺伝子ネットワークの設計問題

2.1 遺伝子ネットワークのダイナミクスとセパラトリクス

本稿で対象とする遺伝子ネットワークのモデルは、次式の区分的線形ネットワークである¹⁰。

$$\begin{aligned} \dot{x}_i(t) &= -d_i x_i(t) + f_i(w_{i1}, w_{i2}, \dots, w_{im_i}, \\ & y_1(t), y_2(t), \dots, y_n(t)), \quad x_i(0) = x_{(0,i)} \end{aligned} \quad (1)$$

$$y_i(t) = h(x_i(t)), \quad i = 1, 2, \dots, n \quad (2)$$

ここで、 h はしきい値関数で、

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

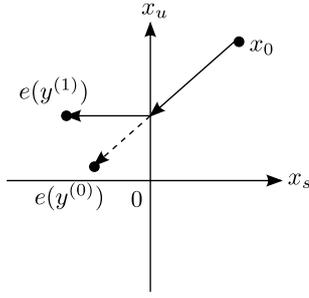


Fig. 1: A trajectory of the networks

である。以下ではこのモデルをベクトル形式で次式のように表す。

$$\dot{x}(t) = -Dx(t) + f(w, y(t)), \quad x(0) = x_0 \quad (3)$$

$$y(t) = H(x(t)) \quad (4)$$

ここで、 $x(t) = [x_1(t) \ x_2(t) \ \cdots \ x_n(t)]'$ であり、 $x_i(t)$ は i 番目の遺伝子の生成物の正規化された濃度を表す。 $x_0 = [x_{(0,1)} \ x_{(0,2)} \ \cdots \ x_{(0,n)}]'$ は初期状態である。 n は遺伝子の数である。 y は $y(t) = [y_1(t) \ y_2(t) \ \cdots \ y_n(t)]'$ であり、各遺伝子の発現レベルを表すベクトルである。ここではこの y を遺伝子ネットワークの発現パターンと呼ぶ。 $f = [f_1 \ f_2 \ \cdots \ f_n]'$ は遺伝子間の相互作用を表す関数である。 w は $w' = [w'_1 \ w'_2 \ \cdots \ w'_n]$ で、 $w_i = [w_{i1} \ w_{i2} \ \cdots \ w_{im_i}]'$ は f_i のパラメータを表すベクトル、 m_i は f_i のパラメータの個数を表す。 D は対角行列で、 $D = \text{diag}\{d_1, d_2, \dots, d_n\}$ であり、 $d_i > 0$ は i 番目の遺伝子の生成物の分解速度に関する正実数パラメータである。 $H(x)$ は、 $H(x) = [h(x_1) \ h(x_2) \ \cdots \ h(x_n)]'$ である。

以下では、発現パターン y に対して $e(y)$ を

$$e(y) = D^{-1}f(w, y)$$

と定義する。さらに、式 (3), (4) の遺伝子ネットワークが発現パターン y をとる x の n 次元実数空間における領域を Ω_y と定義する。すなわち、

$$\Omega_y = \{x \mid y = H(x), \forall i\}$$

とする。

つぎに、セパトリクスについて説明する。そのために、まず、式 (3), (4) の遺伝子ネットワークの動作を簡単に説明する。初期状態 x_0 が $x_0 \in \Omega_{y^{(0)}}$ とする。 $\Omega_{y^{(0)}}$ 内でモデルは線形ネットワーク $\dot{x}(t) = -Dx(t) + f(w, y^{(0)})$ となる。この線形ネットワークは D の定義より安定で、その解軌道は $e(y^{(0)})$ に向かう。このとき、発現パターンの変化の仕方は $e(y^{(0)})$ がどの領域 Ω_y に存在するかで決まる。たとえば、Fig. 1 のように $e(y^{(0)}) \in \Omega_{y^{(1)}}$ とし、 $y^{(0)}$ と $y^{(1)}$ で異なる要素は第 s 要素とする。この場合、 $\text{sign}(x_{(0,s)}) \neq \text{sign}(e_s(y^{(0)}))$ 、 $\text{sign}(x_{(0,i)}) = \text{sign}(e_i(y^{(0)}))$ 、 $i \neq s$ である。 $\Omega_{y^{(0)}}$ 内の遺伝子ネットワークの時刻 t での状態 $\phi(t, x_0)$ は

$$\phi_i(t, x_0) = x_{(0,i)} \exp(-d_i t) + e_i(y^{(0)}) \{1 - \exp(-d_i t)\} \quad (5)$$

となる。 $\text{sign}(x_{(0,i)}) = \text{sign}(e_i(y^{(0)}))$ 、 $i \neq s$ であることと $d_i > 0$ に注意すると、式 (5) より任意の $t > 0$ に対

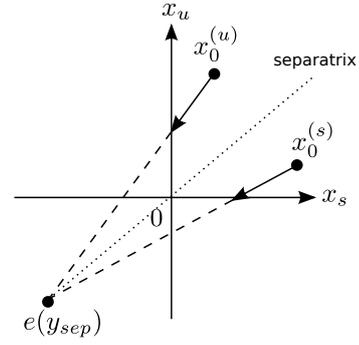


Fig. 2: An example of separatrix

して $\phi_i(t, x_0) \neq 0$ 、 $i \neq s$ である。一方、第 s 要素については、時刻 \hat{t}

$$\hat{t} = -\frac{1}{d_s} \log \left(\frac{e_s(y^{(0)})}{e_s(y^{(0)}) + x_{(s,0)}} \right) \quad (6)$$

で $\phi_s(\hat{t}, x_0) = 0$ となる。よって、発現パターンにおいて第 s 要素だけ変化することが可能で、 $\Omega_{y^{(1)}}$ 以外へ解軌道が向かうことはなく、必ず $y^{(0)}$ から $y^{(1)}$ への遷移が起こる。以下では、 $y^{(0)} \rightarrow y^{(1)}$ によって $y^{(0)}$ から $y^{(1)}$ への発現パターンの遷移を表す。また、複数回の発現パターンの遷移を $y^{(0)} \rightarrow y^{(1)} \rightarrow \cdots \rightarrow y^{(p)}$ のように表し、これを発現パターン遷移列と呼ぶ。また、遺伝子ネットワークが発現パターン遷移列をもつとは、そのような発現パターンの遷移が生じる解軌道をもつことである。

つぎに、セパトリクスをもつ場合の動作を説明する。ここでいうセパトリクスとは、同じ発現パターンが現れる領域 Ω_y を分割する超曲面で、分割された領域によって解軌道が次にどの発現パターンが現れる領域に向かうかが異なる。たとえば、 y_{sep} からそれぞれ第 s 要素と第 u 要素の異なる二つの発現パターン $y^{(s)}$ 、 $y^{(u)}$ にのみ遷移可能である場合を考える。このとき概念図を Fig. 2 に示す。初期値 x_0 を $x_0 \in \Omega_{y_{\text{sep}}}$ とし、 $e(y_{\text{sep}}) \in \Omega_{y_{\text{eq}}}$ とする。 $\Omega_{y_{\text{sep}}}$ 内の遺伝子ネットワークの時刻 t における状態は式 (5) と同様に求められる。このとき、 $\phi_s(t, x_0) = 0$ 、 $\phi_u(t, x_0) = 0$ となる時刻 $\hat{t}_s > 0$ と $\hat{t}_u > 0$ が存在することから、 x_0 と $e(y_{\text{sep}})$ の第 s 要素と第 u 要素で符合が異なることがわかる。つまり、 y_{sep} と y_{eq} で第 s 要素と第 u 要素の二つの要素が異なる値をもつ。また、発現パターンが変化する時刻は、 \hat{t}_s と \hat{t}_u の大小関係で決まる。たとえば、 $\hat{t}_s > \hat{t}_u$ であれば第 u 要素が変化し $y^{(u)}$ への遷移が起こる。解軌道がどちらの領域に向かうかは、 $x_s = 0$ かつ $x_u = 0$ の点を通る解軌道の集合が表す超曲面によって分割されたどの領域を通るかによって決まり、この超曲面がセパトリクスである。

2.2 セパトリクスを実現する設計問題

本稿で考える遺伝子ネットワークの設計問題を説明する。発現パターンに基づく遺伝子ネットワークの設計とは、所望の動作として発現パターン遷移列が与えられ、その発現パターン遷移列をもつ遺伝子ネットワークのパラメータ w を求めることである。前節で述べたように、ある発現パターンから複数の発現パターンへ遷移するにはセパトリクスを実現する必要がある。本稿では、二つの異なる発現パターンへ遷移可能とする

セパトリクスを実現することで所望の発現パターン遷移列をもつようにする、以下のような遺伝子ネットワークの設計問題を考える。

所望の動作として、二つの発現パターン遷移列が与えられ、遺伝子ネットワークにそれらをもたせるにはセパトリクスを実現しなければならないとする。すなわち、式 (3), (4) の遺伝子ネットワークにもたせたい所望の発現パターン遷移列を、 p_l を遷移回数として

$$y^{*(0,l)} \rightarrow y^{*(1,l)} \rightarrow \dots \rightarrow y^{*(p_l,l)}, \quad l = 1, 2 \quad (7)$$

とし、これらの発現パターンは相異なるとする。ただし、二つの発現パターン遷移列で共通な区間が存在するとする。すなわち、共通な区間の数を q 、それらにおける遷移回数を u_j とし、ある $s_j^{(l)}$, $j = 1, 2, \dots, q$, $l = 1, 2$ が存在して、

$$y^{*(s_j^{(1)}+r,1)} = y^{*(s_j^{(2)}+r,2)}, \quad r = 0, 1, \dots, u_j, j = 1, 2, \dots, q \quad (8)$$

$$y^{*(r,l)} \neq y^{*(s,l)}, r \neq s, r, s = 0, 1, \dots, p_l, l = 1, 2 \quad (9)$$

$$y^{*(r,l_1)} \neq y^{*(s,l_2)}, \quad r \neq s_j^{(l_1)}, s_j^{(l_1)} + 1, \dots, s_j^{(l_1)} + u_j, s = 0, 1, \dots, p_{l_2}, j = 1, 2, \dots, q, l_1 \neq l_2, l_1, l_2 = 1, 2 \quad (10)$$

を満たすとする。所望の発現パターン遷移列には、 $y^{*(s_j^{(1)}+u_j,1)} = y^{*(s_j^{(2)}+u_j,2)}$ から二つの異なる発現パターンである $y^{*(s_j^{(1)}+u_j+1,1)}$ と $y^{*(s_j^{(2)}+u_j+1,2)}$ へ遷移が存在する。よって、領域 $\Omega_{y^{*(s_j^{(1)}+u_j,1)}}$ にセパトリクスを実現する必要がある。本稿の遺伝子ネットワークの設計問題はつぎのようになる。

[セパトリクスを実現する遺伝子ネットワークの設計問題] 式 (3), (4) の遺伝子ネットワークに対して、所望の動作として式 (7) の二つの発現パターン遷移列が与えられるとする。このとき、所望の二つの発現パターン遷移列をもつ遺伝子ネットワークを設計せよ。

ここで、所望の発現パターン遷移列に次の仮定をおく。すなわち、 $y^{*(r,l)}$ と $y^{*(r+1,l)}$, $r = 0, 1, \dots, p-1$, $l = 1, 2$ で異なる要素は一つのみであることである。これは $x(t)$ の二つ以上の要素 $x_i(t)$ の符号が同時に変化することはまれであることを意味している。実際、二つ以上の要素の符号が同時に変化することはまれであるのでこの仮定をおいても問題はない。

以下では、二つの発現パターン遷移列で共通する区間の最後の発現パターンを $y_{sep}^{*(j)}$ と表記する。すなわち、 $y_{sep}^{*(j)}$ は、 $y^{*(s_j^{(1)}+u_j,1)}$ または $y^{*(s_j^{(2)}+u_j,2)}$ を表す。 $y^{*(r,l)}$ と $y^{*(r+1,l)}$ で異なる要素の添字を $i_r^{(l)}$ と表記する。すなわち、 $y_{i_r^{(l)}}^{*(r,l)} \neq y_{i_r^{(l)}}^{*(r+1,l)}$, $y_i^{*(r,l)} = y_i^{*(r+1,l)}$, $i \neq i_r^{(l)}$ である。 $y_{sep}^{*(j)}$ と第 $i_{s_j^{(1)}+u_j}^{(1)}$ 要素および第 $i_{s_j^{(2)}+u_j}^{(2)}$ 要素が異なるパターンを $y_{eq}^{*(j)}$ で表す。そして、所望の発現パターン遷移列に対して、 $\Omega_{y^{*(r-1,l)}}$ と $\Omega_{y^{*(r,l)}}$ の境界を $S_r^{(l)}$, $r = 1, 2, \dots, p_l$, $l = 1, 2$ と表記する。

3 セパトリクスを実現する設計問題の解の存在性

ここでは、[セパトリクスを実現する遺伝子ネットワークの設計問題] の解の存在性について議論する。そ

のために、まず、解軌道の初期値 x_0 , $e(y)$, 発現パターンが変化するまでの時間間隔 T とそのときの解軌道の座標 Ψ の間の関係について必要となる事実をいくつか示す。

補題 1 領域 $\Omega_{y^{*(r,l)}}$ における解軌道の進入点 $\Psi^{(r,l)} \in S_r^{(l)}$, 脱出点 $\Psi^{(r+1,l)} \in S_{r+1}^{(l)}$ と領域の通過時間間隔 $T^{(r+1,l)}$ が与えられたとき、 $\Psi^{(r,l)}$ から始まる解軌道が時間間隔 $T^{(r+1,l)}$ で $\Psi^{(r+1,l)}$ に到達するようにするための $e(y^{*(r,l)})$ が存在し、次式で与えられる。

$$e_i(y^{*(r,l)}) = \frac{\Psi_i^{(r+1,l)} - \Psi_i^{(r,l)} \exp(-d_i T^{(r+1,l)})}{1 - \exp(-d_i T^{(r+1,l)})} \quad (11)$$

証明: $\Psi^{(r,l)} \in S_r^{(l)}$, $\Psi^{(r+1,l)} \in S_{r+1}^{(l)}$ であることより、 $\text{sign}(\Psi_i^{(r,l)}) = \text{sign}(\Psi_i^{(r+1,l)})$, $i \neq i_r^{(l)}$, $i_{r+1}^{(l)}$, $\Psi_{i_r^{(l)}}^{(r,l)} = \Psi_{i_r^{(l)}}^{(r+1,l)} = 0$ である。このことから、 $\text{sign}(e_{i_{r+1}^{(l)}}(y^{*(r,l)})) \neq \text{sign}(\Psi_{i_{r+1}^{(l)}}^{(r,l)})$, $\text{sign}(e_i(y^{*(r,l)})) = \text{sign}(\Psi_i^{(r,l)})$, $i \neq i_{r+1}^{(l)}$ で、 $e(y^{*(r,l)}) \in \Omega_{y^{*(r+1,l)}}$ であることがわかる。また、 $\Omega_{y^{*(r,l)}}$ 内の解軌道は $e(y^{*(r,l)})$ へ向かい、時刻 t における状態は次式で与えられる。

$$\begin{aligned} \phi_i(t, \Psi^{(r,l)}) &= \Psi_i^{(r,l)} \exp(-d_i t) \\ &+ e_i(y^{*(r,l)}) \{1 - \exp(-d_i t)\}, \\ &i = 1, 2, \dots, n \end{aligned} \quad (12)$$

よって、 $t = 0$ のとき、

$$\phi(0, \Psi^{(r,l)}) = \Psi^{(r,l)}$$

で、 $t = T^{(r+1,l)}$ のとき、

$$\begin{aligned} \phi_i(T^{(r+1,l)}, \Psi^{(r,l)}) &= \Psi_i^{(r,l)} \exp(-d_i T^{(r+1,l)}) \\ &+ e_i(y^{*(r,l)}) (1 - \exp(-d_i T^{(r+1,l)})) \\ &= \Psi_i^{(r,l)} \exp(-d_i T^{(r+1,l)}) \\ &+ \frac{\Psi_i^{(r+1,l)} - \Psi_i^{(r,l)} \exp(-d_i T^{(r+1,l)})}{1 - \exp(-d_i T^{(r+1,l)})} \\ &\times \{1 - \exp(-d_i T^{(r+1,l)})\} \\ &= \Psi_i^{(r+1,l)}, i = 1, 2, \dots, n \end{aligned}$$

であるので、 $\Psi^{(r,l)}$ から領域 $\Omega_{y^{*(r,l)}}$ に入った解軌道 $\phi(t, \Psi^{(r,l)})$ は、 $T^{(r+1,l)}$ 経ったときに $\Psi^{(r+1,l)}$ に到達し、領域 $\Omega_{y^{*(r,l)}}$ から出て行くことがわかる。□

補題 2 解軌道の領域 $\Omega_{y^{*(r,l)}}$ からの脱出点 $\Psi^{(r+1,l)}$ と通過時間間隔 $T^{(r+1,l)}$ が与えられたとき、 $e(y^{*(r,l)})$ を

$$e(y^{*(r,l)}) \in \Omega_{y^{*(r+1,l)}} \quad (13)$$

を満たすように設定すると、 $y^{*(r,l)}$ から $y^{*(r+1,l)}$ への遷移が必ず生じ、与えられた通過時間間隔 $T^{(r+1,l)}$ で

脱出点 $\Psi^{(r+1,l)}$ に到達する解軌道の進入点の座標 $\Psi^{(r,l)}$ は、次式で与えられる。

$$\begin{aligned} \Psi_i^{(r,l)} &= \exp(d_i T^{(r+1,l)}) \\ &\times \left[\Psi_i^{(r+1,l)} - e_i(y^{*(r,l)}) \left\{ 1 - \exp(-d_i T^{(r+1,l)}) \right\} \right], \\ i &= 1, 2, \dots, n \end{aligned} \quad (14)$$

証明：解軌道の脱出側の交差点が $\Psi^{(r+1,l)}$, $l = 1, 2$ で、 $\Psi^{(r,l)}$ から始まる $\Omega_{y^{*(r,l)}}$ 内の解軌道の $T^{(r+1,l)}$ 後の座標は、式 (12) と式 (14) より

$$\begin{aligned} \phi_i(T^{(r+1,l)}, \Psi^{(r,l)}) &= \Psi_i^{(r,l)} \exp(-d_i T^{(r+1,l)}) \\ &+ e_i(y^{*(r,l)}) (1 - \exp(-d_i T^{(r+1,l)})), \\ &= \Psi_i^{(r+1,l)}, i = 1, 2, \dots, n \end{aligned} \quad (15)$$

と求められる。□

補題 3 所望の発現パターンの遷移として二つの遷移 $y_{sep}^{*(j)} \rightarrow y^{*(s_j^{(1)}+u_j+1,1)}$ と $y_{sep}^{*(j)} \rightarrow y^{*(s_j^{(2)}+u_j+1,2)}$, およびそれらの遷移が生じるまでの時間間隔 $T^{(s_j^{(l)}+u_j+1,l)}$, $l = 1, 2$ が与えられたとする。 $e(y_{sep}^{*(j)})$ を

$$e(y_{sep}^{*(j)}) \in \Omega_{y_{eq}^{(j)}} \quad (16)$$

を満たすように設定すると領域 $\Omega_{y_{sep}^{*(j)}}$ にセパトトリクスが生じ、所望の発現パターンの遷移が生じるような二つの解軌道の領域 $\Omega_{y_{sep}^{*(j)}}$ への進入点 $\Psi^{(s_j^{(1)}+u_j,1)}$ と $\Psi^{(s_j^{(2)}+u_j,2)}$ が存在し、 $\Psi^{(s_j^{(l)}+u_j,l)}$ から始まる解軌道の $\Omega_{y_{sep}^{*(j)}}$ からの脱出点 $\Psi^{(s_j^{(l)}+u_j+1,l)}$, $l = 1, 2$ を求めることができる。

証明：表記の簡単のため、 $y^{*(s_j^{(l)}+u_j+1,l)}$, $S_{s_j^{(l)}+u_j}^{(l)}$, $S_{s_j^{(l)}+u_j+1}^{(l)}$, $\Psi^{(s_j^{(l)}+u_j,l)}$, $\Psi^{(s_j^{(l)}+u_j+1,l)}$, $T^{(s_j^{(l)}+u_j+1,l)}$, $l = 1, 2$ をそれぞれ $y^{*(l)}$, $S_0^{(l)}$, $S_1^{(l)}$, $\Psi^{(0,l)}$, $\Psi^{(1,l)}$, $T^{(l)}$, $l = 1, 2$ と表記する。また、 $y_{sep}^{*(j)} = [1 \ 1 \ \dots \ 1]'$ とし、 $y_{sep}^{*(j)}$ と $y^{*(1)}$ では第 1 要素が異なり、 $y_{sep}^{*(j)}$ と $y^{*(2)}$ では第 2 要素が異なるとしても一般性は失われないので、以下ではこの様な場合について証明する。

$\Psi^{(0,1)}$ と $\Psi^{(0,2)}$ を

$$\Psi_1^{(0,1)} = e_1(y_{sep}^{*(j)}) \left\{ 1 - \exp(d_1 T^{(1)}) \right\} \quad (17)$$

$$\Psi_2^{(0,1)} > e_2(y_{sep}^{*(j)}) \left\{ 1 - \left(\frac{e_1(y_{sep}^{*(j)}) - \Psi_1^{(0,1)}}{e_1(y_{sep}^{*(j)})} \right)^{\frac{d_2}{d_1}} \right\} \quad (18)$$

$$\Psi_i^{(0,1)} > 0, \quad i = 3, 4, \dots, n \quad (19)$$

$$\Psi_1^{(0,2)} > e_1(y_{sep}^{*(j)}) \left\{ 1 - \left(\frac{e_2(y_{sep}^{*(j)}) - \Psi_2^{(0,2)}}{e_2(y_{sep}^{*(j)})} \right)^{\frac{d_1}{d_2}} \right\} \quad (20)$$

$$\Psi_2^{(0,2)} = e_2(y_{sep}^{*(j)}) \left\{ 1 - \exp(d_2 T^{(2)}) \right\} \quad (21)$$

$$\Psi_i^{(0,2)} > 0, \quad i = 3, 4, \dots, n \quad (22)$$

を満たすように設定する。 $\Psi^{(0,1)}$ と $\Psi^{(0,2)}$ がこのような条件を満たすと、 $\Psi^{(0,1)} \in \Omega_{y_{sep}^{*(j)}}$, $\Psi^{(0,2)} \in \Omega_{y_{sep}^{*(j)}}$ となる。以下では、このように設定すると $\Omega_{y_{sep}^{*(j)}}$ にセパトトリクスが存在し、 $\Psi^{(0,l)}$, $l = 1, 2$ から始まる解軌道において、 $y_{sep}^{*(j)} \rightarrow y^{*(l)}$, $l = 1, 2$ なるパターン遷移が生じることを示す。

$\Psi^{(0,1)}$ から始まる $\Omega_{y_{sep}^{*(j)}}$ 内の時刻 t での状態は次式で与えられる。

$$\begin{aligned} \phi_i(t, \Psi_i^{(0,1)}) &= \Psi_i^{(0,1)} \exp(-d_i t) \\ &+ e_i(y_{sep}^{*(j)}) \{ 1 - \exp(-d_i t) \} \\ i &= 1, 2, \dots, n \end{aligned} \quad (23)$$

まず、この解軌道が $S_1^{(1)}$ と交差することを示す。式 (16) を満たすことから、 $e_1(y_{sep}^{*(j)})$ は $\Psi_1^{(0,1)}$ と異符号で、 $e_2(y_{sep}^{*(j)})$ は $\Psi_2^{(0,2)}$ と異符号である。また、第 3 要素から第 n 要素については、 $e(y_{sep}^{*(j)})$ と $\Psi^{(0,1)}$ および $\Psi^{(0,2)}$ の要素は同符号である。そのため、先に示したように、 $\Psi^{(0,1)}$ から始まる解軌道において、ある時刻に値が 0 になるのは第 1 要素または第 2 要素のどちらかである。どちらの要素が先に 0 となるかを示すために、式 (23) において $\phi_1(\hat{T}, \Psi^{(0,1)}) = 0$, $\phi_2(\tilde{T}, \Psi^{(0,1)}) = 0$ となる $\hat{T} > 0$, $\tilde{T} > 0$ を求める。すると、それぞれ次のように得られる。 \hat{T} は、

$$0 = \Psi_1^{(0,1)} \exp(-d_1 \hat{T}) + e_1(y_{sep}^{*(j)}) \left\{ 1 - \exp(-d_1 \hat{T}) \right\}$$

を満たすことから、

$$\exp(-d_1 \hat{T}) = \frac{e_1(y_{sep}^{*(j)})}{e_1(y_{sep}^{*(j)}) - \Psi_1^{(0,1)}} = \frac{1}{\exp(d_1 T^{(1)})}$$

が成り立ち、これより

$$\hat{T} = T^{(1)} = \log_e \left(\frac{e_1(y_{sep}^{*(j)}) - \Psi_1^{(0,1)}}{e_1(y_{sep}^{*(j)})} \right)^{\frac{1}{d_1}} \quad (24)$$

と求められる。同様にして、 \tilde{T} は

$$\tilde{T} = \log_e \left(\frac{e_2(y_{sep}^{*(j)}) - \Psi_2^{(0,1)}}{e_2(y_{sep}^{*(j)})} \right)^{\frac{1}{d_2}} \quad (25)$$

と求められる。ここで、式 (18) から

$$\left(\frac{e_1(y_{sep}^{*(j)}) - \Psi_1^{(0,1)}}{e_1(y_{sep}^{*(j)})} \right)^{\frac{1}{d_1}} < \left(\frac{e_2(y_{sep}^{*(j)}) - \Psi_2^{(0,1)}}{e_2(y_{sep}^{*(j)})} \right)^{\frac{1}{d_2}}$$

であるので、式 (24) と式 (25) より $\hat{T} < \tilde{T}$ であることがわかる。よって、 $\Psi^{(0,1)}$ から始まる解軌道は $S_1^{(1)}$ と交差する。 $\Psi^{(0,2)}$ から始まる解軌道が $S_1^{(2)}$ と交差することも同様に示すことができ、それらの交差点は、次式で与えられる。

$$\begin{aligned} \Psi_i^{(1,l)} &= \Psi_i^{(0,l)} \exp(-d_i T^{(l)}) \\ &+ e_i(y_{sep}^{*(j)}) \left\{ 1 - \exp(-d_i T^{(l)}) \right\}, \\ i &= 1, 2, \dots, n, l = 1, 2 \end{aligned} \quad (26)$$

ため、そのような y に対する $e(y)$ は任意の実数ベクトルに設定することができる。

以上のようにしてすべての発現パターン y に対する $e(y)$ が求められると、それを実現するパラメータ w_y を

$$w_y = De(y), \quad \forall y \quad (30)$$

により求めることができるので、式 (28) の関数のパラメータが求められることがわかる。よって、それらを展開してまとめることで解 w が求められる。以上の議論は任意の発現パターン遷移列に対して成り立つので、任意の2つの所望の動作に対して、遺伝子ネットワークがそれらをもつようにするパラメータが存在する。□

4 セパラトリクスを実現する設計法

定理1の証明では、[セパラトリクスを実現する遺伝子ネットワークの設計問題]において定めるべきパラメータ w を解析的に求める方法を与えている。そこで、定理1の証明の考え方に基づいた設計法の手順を以下に示す。

[セパラトリクスを実現する遺伝子ネットワークの設計法]

Step 1. 所望の二つの発現パターン遷移列を式 (29) のように分割する。

Step 2. EPS_0 を実現するような $e(y^{*(0,l)})$, \dots , $e(y^{*(s_1^{(l)}+u_1+1,l)})$, $l = 1, 2$ を求める。

Step 2-1. $e(y_{sep}^{*(1)})$ を、式 (16) を満たすように定める。

Step 2-2. $T(s_1^{(l)}+u_1+1,l)$ を適当な正の数に選び、式 (17) から式 (22) の条件を満たすように $\Psi(s_1^{(l)}+u_1,l)$ を定め、式 (26) より $\Psi(s_1^{(l)}+u_1+1,l)$ を求める。

Step 2-3. $l = 1$ とする。

Step 2-4. $r = s_1^{(l)} + u_1 - 1$ とする。

Step 2-5. $e(y^{*(r,l)})$ が既に求められているなら、Step 2-7. へ進む。

Step 2-6. $T(r+1,l)$ を適当な正の数、 $e(y^{*(r,l)})$ を式 (13) を満たすように定め、式 (14) により $\Psi(r,l)$ を求める。

Step 2-7. $r = r - 1$ とし、 $r > 0$ ならば Step 2-5. へ、 $r = 0$ なら Step 2-8. へ進む。

Step 2-8. $l = l + 1$ とする。 $l = 2$ ならば Step 2-4. へ進む。 $l = 3$ なら Step 3. へ進む。

Step 3. $j = 2$ とする。

Step 4. EPS_{j-1} を実現するための $e(y^{*(s_{j-1}^{(l)}+u_{j-1}+1,l)})$, \dots , $e(y^{*(s_j^{(l)}+u_j,l)})$, $l = 1, 2$ を求める。

Step 4-1. $e(y_{sep}^{*(j)})$ を、式 (16) を満たすように定める。

Step 4-2. $T(s_j^{(l)}+u_j+1,l)$ を適当な正の数、式 (17) から式 (22) の条件を満たすように $\Psi(s_j^{(l)}+u_j,l)$ を定め、式 (26) より $\Psi(s_j^{(l)}+u_j+1,l)$ を求める。

Step 4-3. $l = 1$ とする。

Step 4-4. $r = s_j^{(l)} + u_j - 1$ とする。

Step 4-5. $e(y^{*(r,l)})$ が既に求められているなら、Step 4-7. へ進む。

Step 4-6. $T(r+1,l)$ を適当な正の数、 $e(y^{*(r,l)})$ を、式 (13) を満たすように定め、式 (14) により $\Psi(r,l)$ を求める。

Step 4-7. $r = r - 1$ とする。 $r > s_{j-1}^{(l)} + u_{j-1} + 1$ なら Step 4-5. へ、 $r = s_{j-1}^{(l)} + u_{j-1} + 1$ なら Step 4-8. へ進む。

Step 4-8. $l = l + 1$ とする。 $l = 2$ ならば Step 4-4. へ進む。 $l = 3$ なら Step 4-9. へ進む。

Step 4-9. $T(r,l)$, $l = 1, 2$ を適当な正の数に設定し、式 (11) より $e(y^{*(r,l)})$, $l = 1, 2$ を求める。

Step 5. $j = j + 1$ とする。 $j < q$ なら Step 4. へ、 $j = q$ なら Step 6. へ進む。

Step 6. EPS_q を実現するための $e(y^{*(s_q^{(l)}+u_q+1,l)})$, \dots , $e(y^{*(p_l-1,l)})$, $l = 1, 2$ を求める。

Step 6-1. $l = 1$ とする。

Step 6-2. $s_q^{(l)} + u_q + 1 = p_l$ ならば Step 6-8. へ進む。

Step 6-3. $T(s_q^{(l)}+u_q+1,l)$ を適当な正の数、 $\Psi(s_q^{(l)}+u_q+1,l)$ を適切な値に設定し、式 (14) により $\Psi(s_q^{(l)}+u_q,l)$ を求める。

Step 6-4. $r = s_q^{(l)} + u_q - 1$ とする。

Step 6-5. $e(y^{*(r,l)})$ が既に求められているなら、Step 6-7. へ進む。

Step 6-6. $T(r+1,l)$ を適当な正の数、 $e(y^{*(r,l)})$ を、式 (13) を満たすように定め、式 (14) により $\Psi(r,l)$ を求める。

Step 6-7. $r = r - 1$ とする。 $r > s_q^{(l)} + u_q + 1$ なら Step 6-5. へ、 $r = s_q^{(l)} + u_q + 1$ なら Step 6-8. へ進む。

Step 6-8. $l = l + 1$ とする。 $l = 2$ ならば Step 6-4. へ進む。 $l = 3$ なら Step 6-9. へ進む。

Step 6-9. $T(r,l)$, $l = 1, 2$ を適当な正の数に設定し、式 (11) より $e(y^{*(r,l)})$, $l = 1, 2$ を求める。

Step 7. 式 (30) より w_y をすべて求める。

Step 8. 式 (28) を展開して w を求め、終了する。

所望の発現パターン遷移列が周期的である場合、持続的な周期発現パターン遷移列を実現するためには $\Psi(0,l) = \Psi(p_l,l)$ とする必要がある⁷⁾。そのために、 $e(y^{*(0,l)})$ を求めることを最後に行う。すなわち、 $\Psi(p_l,l)$ が求められてから $T(0,l)$ を適当な正の数に設定し、 $\Psi(p_l,l)$ を $\Psi(0,l)$ として式 (11) より $e(y^{*(0,l)})$ を求める。

5 数値例

所望の発現パターン遷移列は、つぎの二つの周期的な発現パターン遷移列とする。

$$\begin{aligned} [0\ 1\ 0]' &\rightarrow [0\ 0\ 0]' \rightarrow [1\ 0\ 0]' \rightarrow [1\ 1\ 0]' \\ &\rightarrow [1\ 1\ 1]' \rightarrow [0\ 1\ 1]' \rightarrow [0\ 1\ 0]' \end{aligned} \quad (31)$$

$$\begin{aligned} [0\ 0\ 1]' &\rightarrow [0\ 0\ 0]' \rightarrow [1\ 0\ 0]' \rightarrow [1\ 0\ 1]' \\ &\rightarrow [1\ 1\ 1]' \rightarrow [0\ 1\ 1]' \rightarrow [0\ 0\ 1]' \end{aligned} \quad (32)$$

これらをもつ遺伝子ネットワークを提案法により設計する。

所望の発現パターン遷移列をみると、第1番目と第4の発現パターンが異なり、それら以外は二つの発現パターン遷移列で同じで、 $s_1^{(1)} = s_1^{(2)} = 1$, $u_1 = 1$, $s_2^{(1)} = s_2^{(2)} = 4$, $u_2 = 1$ である。そこで、これらを次のように分割する。

$$\begin{aligned} \text{EPS}_0 : &\begin{cases} [0\ 1\ 0]' \rightarrow [0\ 0\ 0]' \rightarrow [1\ 0\ 0]' \rightarrow [1\ 1\ 0]' \\ [0\ 0\ 1]' \rightarrow [0\ 0\ 0]' \rightarrow [1\ 0\ 0]' \rightarrow [1\ 0\ 1]' \end{cases} \\ \text{EPS}_1 : &\begin{cases} [1\ 1\ 0]' \rightarrow [1\ 1\ 1]' \rightarrow [0\ 1\ 1]' \rightarrow [0\ 1\ 0]' \\ [1\ 0\ 1]' \rightarrow [1\ 1\ 1]' \rightarrow [0\ 1\ 1]' \rightarrow [0\ 0\ 1]' \end{cases} \end{aligned}$$

EPS_0 を実現する $e(y^{*(r,l)})$, $r = 1, 2$ は, $T^{(2,l)} = \log_e 2$, $T^{(3,l)} = \log_e (\frac{5}{4})$ と設定して

$$e([1\ 0\ 0]') = [1\ 2\ 2]', \quad e([0\ 0\ 0]') = [1\ -1\ -1]'$$

と求まり、二つの解軌道の通過点は

$$\begin{aligned} \Psi^{(1,1)} &= [-1\ 0\ -1]', \Psi^{(1,2)} = [-1\ -1\ 0]', \\ \Psi^{(2,1)} &= [0\ -\frac{1}{2}\ -1]', \Psi^{(2,2)} = [0\ -1\ -\frac{1}{2}]', \\ \Psi^{(3,1)} &= [\frac{1}{5}\ 0\ -\frac{2}{5}]', \Psi^{(3,2)} = [\frac{1}{5}\ -\frac{2}{5}\ 0]' \end{aligned}$$

と求められる。

EPS_1 を実現する $e(y^{*(r,l)})$, $r = 3, 4, 5$ は, $T^{(4,l)} = \log_e 2$, $T^{(5,l)} = \log_e (\frac{3}{2})$, $T^{(6,l)} = \log_e (\frac{3}{2})$ と設定して

$$\begin{aligned} e([0\ 1\ 1]') &= [-1\ -2\ -2]', \quad e([1\ 1\ 1]') = [-1\ 3\ 3]' \\ e([1\ 1\ 0]') &= [\frac{4}{5}\ 3\ \frac{2}{5}]', \quad e([1\ 0\ 1]') = [\frac{4}{5}\ \frac{2}{5}\ 3]' \end{aligned}$$

と求まり、二つの解軌道の通過点は

$$\begin{aligned} \Psi^{(4,1)} &= [\frac{1}{2}\ \frac{3}{2}\ 0]', \Psi^{(4,2)} = [\frac{1}{2}\ 0\ \frac{3}{2}]', \\ \Psi^{(5,1)} &= [0\ 2\ 1]', \Psi^{(5,2)} = [0\ 1\ 2]', \\ \Psi^{(6,1)} &= [-\frac{1}{3}\ \frac{2}{3}\ 0]', \Psi^{(6,2)} = [-\frac{1}{3}\ 0\ \frac{2}{3}]' \end{aligned}$$

と求められる。

最後に、 $\Psi^{(6,1)}$, $\Psi^{(6,2)}$ をそれぞれ $\Psi^{(0,1)}$, $\Psi^{(0,2)}$ とし, $T^{(1,l)} = \log_e 2$, $l = 1, 2$ として $e([0\ 1\ 0]')$ と $e([0\ 0\ 1]')$ は

$$e([0\ 1\ 0]') = [-\frac{5}{3}\ -\frac{2}{3}\ -2]', \quad e([0\ 0\ 1]') = [-\frac{5}{3}\ -2\ -\frac{2}{3}]'$$

と求められる。

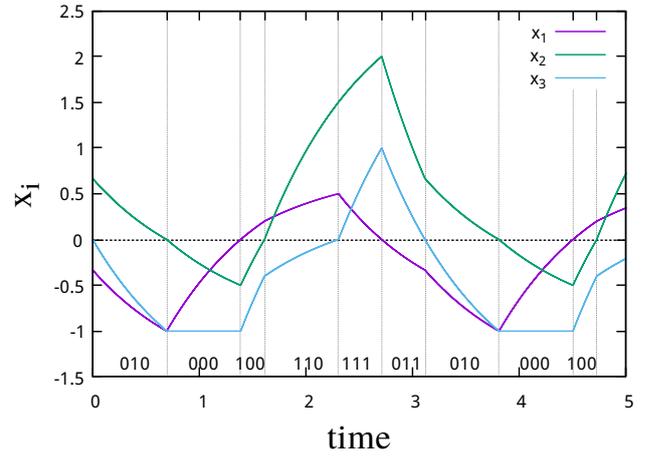


Fig. 3: A simulation result of the obtained gene regulatory network. The trajectory is periodic and the expression pattern sequence is same as the desired one (31)

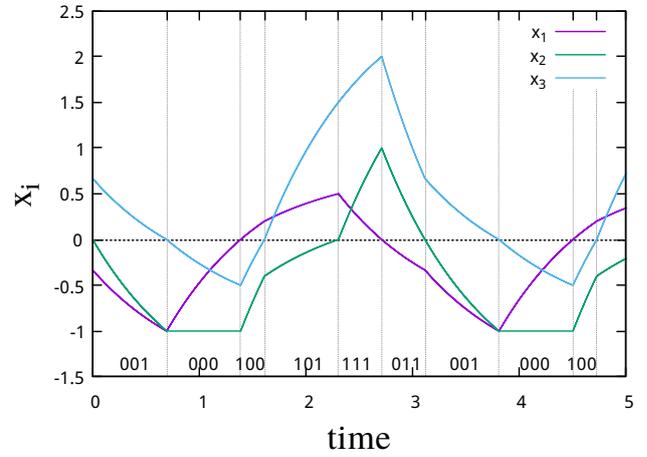


Fig. 4: A simulation result of the obtained gene regulatory network. The trajectory is periodic and the expression pattern sequence is same as the desired one (32)

提案法によって得られた遺伝子ネットワークのシミュレーション結果を Fig. 3 と Fig. 4 に示す。図の下部の数字は発現パターンを表し、縦の破線は発現パターンが変化した時刻を表す。Fig. 3, Fig. 4 より所望の二つの発現パターン遷移列をもつ遺伝子ネットワークが得られていることがわかる。

6 おわりに

所望の機能をもつ遺伝子ネットワークを設計、実現する研究が盛んに行われている。筆者らは、これまでに発現パターンに着目し、その変化を表す発現パターン遷移列を所望の動作としたときの設計法を提案している。より単純なモデルで多様な振舞いを実現したり、より多くの振舞いを同時に果たせたりできる方が望ましいと考えられる。そのためには、モデルの解軌道を考えたとき、同じ発現パターンが現れる領域を通るものの中に、つぎに現れる発現パターンが異なるものがある遺伝子ネットワークが実現可能となればよい。そこで、この意味で状態空間の領域を分割するセパトリクスを実現する必要がある設計問題に対して設計法を提案している。本稿では、この設計問題に対して領域

を二つに分割するセパトリクスを実現する必要のある場合について解が存在することを示した。また、その証明では所望の動作をもつ遺伝子ネットワークのパラメータを求めることで解の存在を示しているため、解法の提案も行っている。

参考文献

- 1) M. B. Elowitz and S. Leibler: A synthetic oscillatory network of transcriptional regulators, *Nature*, **403**, 335/338 (2000)
- 2) J. Hasty and F. Isaacs, F: Designer gene networks: Towards fundamental cellular control, *Chaos*, **11**-1, 207/220 (2001)
- 3) 市瀬, 合原: 遺伝子ネットワークモデルとその設計について, 第15回回路とシステム(軽井沢)ワークショップ, 589/593 (2002)
- 4) H. Nakayama, H. Tanaka and T. Ushio: The formulation of the control of an expression pattern in a gene network by propositional calculus, *J. Theor. Biol.*, **240**-3, 443/450 (2006)
- 5) 森, 門脇, 黒江, 森: 遺伝子ネットワークの学習による発現ベースの設計法—セパトリクスの実現, 第33回知能システムシンポジウム資料, (2006)
- 6) 森, 黒江, 森: 発現パターンに基づく遺伝子ネットワークの設計法, 計測自動制御学会論文集, **44**-11, 936/945 (2008)
- 7) Y. Mori and Y. Kuroe: Synthesis Method of Gene Regulatory Networks Having Desired Periodic Expression Pattern Sequences, Proc. of IEEE International Conference on SMC 1159/1164 (2012)
- 8) Y. Mori and Y. Kuroe, Synthesis Method of Gene Regulatory Networks Having Desired Expression-Pattern Transition Sequences, Proc. of 2013 9th Asian Control Conference, (2013)
- 9) 森, 黒江: 発現パターンに基づく遺伝子ネットワークの設計問題—解法と解の存在性—, 第56回自動制御連合講演会講演論文集, 1172/1177, (2013)
- 10) Glass, L., Classification of biological networks by their qualitative dynamics, *J. Theor. Biol.*, **54**, 85/107 (1975)

著作権 © 2014

公益社団法人計測自動制御学会(SICE)
〒113-0033 東京都文京区本郷 1-35-28-303

カタログ番号 14 PG 0014

著作権は、計測自動制御学会がもっている
ので、個人の使用のための複製以外の
目的で掲載の記事の一部または全文を
複製する場合には、著作権者に許可を求
め規定の複製料を支払うこと。

発行日：2014年12月8日

発行者：公益社団法人計測自動制御学会 システム・情報部門
コンピューターショナル・インテリジェンス部会