第11回コンピューテーショナル・インテリジェンス研究会
2017年6月26日-27日, 大津

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の解の存在性と解法

○森　禎弘　黒江康明（京都工芸繊維大学）

Existence of solutions to synthesis problem of gene regulatory networks for realizing separatrices

∗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 behavior 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，設計問題において，解の存在性は基本的な問題である．これまでに上述の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たときに，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複数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．この問題の場合，状態空間においてセパラトリクスおよび二つの解軌道の相対的な位置関係に関する条件を満たすようなパラメータおよび解軌道の初期値を求めることになる．本稿では，この条件の十分条件を導出し，その十分条件を満たすパラメータを求める手順を示すことで任意の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証明する．

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，次式の区分線形ネットワークである [10]．

$$\begin{aligned}\dot{x}_i(t) = -x_i(t) + f_i(&y_1(t), y_2(t), \cdots, y_n(t),\\ &w_{i1}, w_{i2}, \cdots, w_{im_i}),\quad x_i(0) = x_{(0,i)}\end{aligned} \tag{1}$$

$$y_i(t) = h(x_i(t)),\quad i = 1, 2, \cdots, n \tag{2}$$

ここで，h はしきい値関数で，

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

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．

$$\dot{x}(t) = -x(t) + f(y(t), w), \quad x(0) = x_0 \tag{3}$$

$$y(t) = H(x(t)) \tag{4}$$

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．$H(x)$ は，$H(x) = [h(x_1)\ h(x_2)\ \cdots\ h(x_n)]'$ である．

以下では，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．

相互作用関数としてよく用いられるものの一つに

$$f_i(y, a^{(i)}) = a_0^{(i)} + \sum_{j=1}^{n} a_j^{(i)} y_j + \sum_{j=1}^{n-1} \sum_{k=j+1}^{n} a_{jk}^{(i)} y_j y_k + \cdots + a_{12\cdots n}^{(i)} y_1 \cdots y_n, \tag{5}$$

$i = 1, 2, \cdots, n$ がある．本稿では，これを相互作用関数として用いることにする．また，パラメータ w を明記する必要がなければ，簡単のため相互作用関数を $f(y)$ と略記する．

つぎに，セパラトリクスが存在するときの遺伝子ネットワークの解軌道について説明する．初期状態 x_0 が $x_0 \in \Omega_{y^{(0)}}$ を満たすとする．解軌道は $f(y^{(0)})$ に向かう．このとき，発現パターンの変化の仕方は $f(y^{(0)})$ がどの領域 Ω_y に存在するかで決まる．$\Omega_{y^{(0)}}$ 内における遺伝子ネットワークの時刻 t での状態 $\phi(t, x_0)$ は

$$\phi_i(t, x_0) = x_{(0,i)} \exp(-t) + f_i(y^{(0)}) \{1 - \exp(-t)\} \tag{6}$$

となる．ここでいうセパラトリクスとは，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，つぎの仮定が成り立つ場合を考える．

[仮定] $f(y_{\rm sep}) \in \Omega_{y_{eq}}$ とする．ただし，$y_{\rm sep}$ と $y_{\rm eq}$ で第 s 要素と第 u 要素の二つの要素が異なる値をもつとする．

初期値 x_0 が $x_0 \in \Omega_{y_{\rm sep}}$ を満たすとし，このときの概念図を Fig. 1 に示す．$\Omega_{y_{\rm sep}}$ 内の遺伝子ネットワークの時刻 t における状態は式 (6) と同様に求められる．x_0 と $f(y_{\rm sep})$ の第 s 要素と第 u 要素で符合が異なることから，$\phi_s(\hat{t}_s, x_0) = 0$ あるいは $\phi_u(\hat{t}_u, x_0) = 0$ となる時刻 $\hat{t}_s > 0$ と $\hat{t}_u > 0$ がそれぞれ存在することがわかる．発現パターンが変化する時刻は，これら $\hat{t}_s$ 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領域のうち，どちらを通るかによって決まり，この超曲面がセパラトリクスである．セパラトリクス上の点から始まる解軌道では，ある時刻 $\hat{t}$ において $x_s(\hat{t}) = 0$ かつ $x_u(\hat{t}) = 0$ となることから，このセパラトリクスは次式で与えられる．

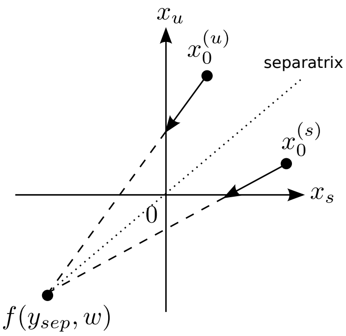


Fig. 1: An example of separatrix

$$x_u = \frac{f_u(y_{sep})}{f_s(y_{sep})} x_s \tag{7}$$

そして，Fig. 1 からただちにわかるように，解軌道の初期値 x_0 が

$$x_0 \in \Omega_{y_{sep}}, \quad \frac{x_{(0,u)}}{x_{(0,s)}} < \frac{f_u(y_{sep})}{f_s(y_{sep})} \tag{8}$$

を満たすならば，その解軌道において発現パターンは y_{sep} から $y^{(s)}$ へ，

$$x_0 \in \Omega_{y_{sep}}, \quad \frac{f_u(y_{sep})}{f_s(y_{sep})} < \frac{x_{(0,u)}}{x_{(0,s)}} \tag{9}$$

を満たすならば，その解軌道において発現パターンは y_{sep} から $y^{(u)}$ へ遷移する．

以上より，つぎの補題が成り立つことが分かる．

補題 1 [仮定] が満たされるとする．このとき，式 *(7)* のセパラトリクスが存在し，初期値が式 *(8)*(式 *(9)*) を満たすとき，その解軌道において発現パターンは y_{sep} から $y^{(s)}$ ($y^{(u)}$) へ遷移する．

以下では，$y^{(0)} \to y^{(1)}$ 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 $y^{(0)} \to y^{(1)} \to \cdots \to y^{(p)}$ 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．

2.2 セパラ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相互作用関数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ラトリクスを実現する必要がある．

本稿で考える設計問題は，もたせたい所望の発現パターン遷移列が複数あり，セパラトリクスを実現する必要があるような設計問題である．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，q を発現パターン遷移列の数，p_l を l 番目の発現パターン遷移列の遷移回数として

$$y^{*(0,l)} \to y^{*(1,l)} \to \cdots \to y^{*(p_l,l)}, \quad l = 1, 2, \cdots, q \tag{10}$$

とする．ただし，同じ発現パターン遷移列内の発現パターンは相異なるとする．すなわち，

$$y^{*(r,l)} \neq y^{*(\hat{r},l)},\ r \neq \hat{r} \tag{11}$$

とする．セパラトリクスを実現する必要があることから，異なる発現パターン遷移列に共通な発現パターンが存在するとする．すなわち，

$$y^{*(r,l)} = y^{*(\hat{r},\hat{l})},\ l \neq \hat{l} \tag{12}$$

となるような $r, l, \hat{r}$ と $\hat{l}$ が存在するとする．さらに，二つの発現パターン遷移列で共通なものが存在していたとしても，いずれ別々の発現パターンへ遷移するとする．すなわち，式 (12) を満たす $r, \hat{r}, l, \hat{l}$ に対して，ある s が存在し，

$$y^{*(r+s,l)} \neq y^{*(\hat{r}+s,\hat{l})} \tag{13}$$

となるとする．そのため，この所望の発現パターン遷移列を遺伝子ネットワークにもたせるには，セパラトリクスを実現する必要がある．ただし，高々二つの発現パターン遷移列でしか共通でない，すなわち，式 (12) が成り立つとき，

$$y^{*(\hat{r},\hat{l})} \neq y^{*(\bar{r},\bar{l})},\ \forall \bar{r} \neq \hat{r},\ \forall \bar{l} \neq \hat{l} \tag{14}$$

とする．

[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] 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 (10) 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 $y^{*(r+1,l)}$, $r = 0, 1, \cdots, p-1$, $l = 1, 2, \cdots, q$ で異なる要素は一つのみであることである．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，$y^{*(r,l)}$ 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，$y^{*(r,l)}_{i_r^{(l)}} \neq y^{*(r+1,l)}_{i_r^{(l)}}$, $y_i^{*(r,l)} = y_i^{*(r+1,l)}$, $\forall i \neq i_r^{(l)}$ である．所望の発現パターン遷移列に対して，$\Omega_{y^{*(r,l)}}$ と $\Omega_{y^{*(r+1,l)}}$ の境界を $S_r^{(l)}$, $r = 0, 1, \cdots, p_l - 1$, $l = 1, 2, \cdots, q$ と表記する．また，セパラトリクスを実現する必要がある領域における発現パターンの集合を $\mathcal{Y}_{sep}$ と表記する．すなわち，$y^{*(r,l)} \in \mathcal{Y}_{sep}$ であるならば，ある $\hat{r}, \hat{l}$ が存在し，$s = 1$ としたときに式 (12) と (13) が成り立つ．さらに，この場合，$y^{*(r+1,l)}$ と $y^{*(\hat{r}+1,\hat{l})}$ では，第 $i_r^{(l)}$ 要素と第 $i_{\hat{r}}^{(\hat{l})}$ 要素が異なり，$y^{*(r,l)}$ とこれら二つの要素が異なるパターンを $y_{eq}^{*(r,l)}$ と表記する．すなわち，

$$y_i^{*(r,l)} = y_{eq,i}^{*(r,l)},\ \forall i \neq i_r^{(l)}, i_{\hat{r}}^{(\hat{l})} \tag{15}$$

$$y_{i_r^{(l)}}^{*(r,l)} \neq y_{eq,i_r^{(l)}}^{*(r,l)}, \tag{16}$$

$$y_{i_{\hat{r}}^{(\hat{l})}}^{*(r,l)} \neq y_{eq,i_{\hat{r}}^{(\hat{l})}}^{*(r,l)} \tag{17}$$

である．また，$\Psi^{(r,l)}$ を次のように定義する．

$$\Psi_i^{(r+1,l)} := \frac{\Psi_i^{(r,l)} f_{i_r^{(l)}}(y^{*(r,l)}) - f_i(y^{*(r,l)}) \Psi_{i_r^{(l)}}^{(r,l)}}{f_{i_r^{(l)}}(y^{*(r,l)}) - \Psi_{i_r^{(l)}}^{(r,l)}},$$
$$i = 1, 2, \cdots, n,$$
$$r = 0, 1, \cdots, p_l, \quad l = 1, 2, \cdots, q \tag{18}$$

$y^{*(r,l)} \to y^{*(r+1,l)}$ なる発現パターンが生じるとき，この $\Psi^{(r+1,l)}$ は $\Psi^{(r,l)} \in S_r^{(l)} \cap \Omega_{y^{*(r,l)}}$ から始まる解軌道が $S_{r+1}^{(l)}$ と交差する点である．

3 セパラトリクスを実現する設計問題の解の存在性

ここでは，[セパラ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．

遺伝子ネットワークが所望の発現パターン遷移列をもつための十分条件は，以下のようになる．所望の発現パターン遷移列 (10) における $y^{*(r,l)}$ から $y^{*(r+1,l)}$ への遷移を遺伝子ネットワークがもつための十分条件は

$$f(y^{*(r,l)}) \in \Omega_{y^{*(r+1,l)}} \tag{19}$$

である [6]．ただし，$y^{*(r,l)} \notin \mathcal{Y}_{sep}$ である．また，所望の発現パターン遷移列には，同じ発現パターンから異なる二つの発現パターンへの遷移が存在する．すなわち，所望の発現パターン遷移列において，式 (12), (13) を満たす $r, \hat{r}, l, \hat{l}, s$ が存在するとしているため，式 (14) を考慮すると，すべての $y^{*(r,l)} \in \mathcal{Y}_{sep}$ に対して，領域 $\Omega_{y^{*(r,l)}}$ を二つに分割するセパラトリクスを実現しなければならない．そのための十分条件は，補題 1 より以下のように与えられる．すなわち，すべての $y^{*(r,l)} \in \mathcal{Y}_{sep}$ に対して，

$$f(y^{*(r,l)}) \in \Omega_{y_{eq}^{*(r,l)}} \tag{20}$$

が成り立ち，さらに，つぎの条件を満たす $\Psi^{(r,l)}$ と $\Psi^{(\hat{r},\hat{l})}$ が存在することである．

$$\left| \frac{\Psi_{i_{\hat{r}}^{(\hat{l})}}^{(\hat{r},\hat{l})}}{\Psi_{i_r^{(l)}}^{(\hat{r},\hat{l})}} \right| < \left| \frac{f_{i_{\hat{r}}^{(\hat{l})}}(y^{*(r,l)})}{f_{i_r^{(l)}}(y^{*(r,l)})} \right| < \left| \frac{\Psi_{i_{\hat{r}}^{(\hat{l})}}^{(r,l)}}{\Psi_{i_r^{(l)}}^{(r,l)}} \right| \tag{21}$$

$$\Psi^{(r,l)} \in S_r^{(l)} \tag{22}$$

$$\Psi^{(\hat{r},\hat{l})} \in S_{\hat{r}}^{(\hat{l})} \tag{23}$$

ここで，$y^{*(\hat{r},\hat{l})}$ は，$y^{*(r,l)} = y^{*(\hat{r},\hat{l})}, l \neq \hat{l}$ なるパターンである．以上より，つぎの補題を得る．

補題 2 *$y^{*(r,l)} \notin \mathcal{Y}_{sep}$ なるすべての $y^{*(r,l)}$ に対して式 (19) の条件を，$y^{*(r,l)} \in \mathcal{Y}_{sep}$ なるすべての $y^{*(r,l)}$ に対*

して式 *(20), (21), (22), (23)* の条件を満たす $\Psi^{(r,l)}$ が存在するならば，遺伝子ネットワークは所望の発現パターン遷移列をもつ．

この補題の条件を満たすパラメータ w を求めることができれば，それが本稿の設計問題の解となる．任意の所望の発現パターン遷移列に対して，この補題の条件を満たすパラメータ w の存在を示すことができ，それによってつぎの結果を得る．

定理 1 任意の所望の発現パターン遷移列に対して，所望の動作をもつ 遺伝子ネットワーク が存在する．

証明: 設計手順を示し，任意の所望の発現パターン遷移列に対してその手順で設計できることを示す．これによって本稿の設計問題に解が存在することを示す．証明は以下のようにして行う．

1. 設計手順を示すために，補題 2 の条件一つであるセパラトリクスを実現するための条件をより簡単な十分条件に置き換える．

2. 1. で示した十分条件を満たすようにする $\Psi^{(r,l)}$ と $f(y^{*(r,l)})$, $r = 0, 1, \cdots, p_l$, $l = 1, 2, \cdots, q$ の値の組を求める手順を示す．

3. すべての $y^{*(r,l)}$ に対して $f(y^{*(r,l)})$ の値が 2. で求めた値となるようにするパラメータ w の値を $f(y^{*(r,l)})$ の値から求める手順を示す．

以上の 2. および 3. で示す手順により，任意の発現パターン遷移列に対して w の値を決定することが可能で，設計問題の解が存在することがわかる．

まず，1. で述べた十分条件を導く．(a) セパラトリクスを実現する必要がないときの条件である式 (19) の条件は，そのままである．(b) セパラトリクスを実現する必要があるときは，式 (21) の要素間の比の不等式条件を満たす必要がある．この条件を，つぎのようにして要素間の等式条件に置き換えた十分条件にする．$y^{*(r,l)} \in \mathcal{Y}_{sep}$ となるすべての r, l に対して，

$$\left|\Psi^{(\hat{r},\hat{l})}_{i^{(\hat{l})}_{\hat{r}}}\right| = \alpha^{*(\hat{r},\hat{l},r,l)} \left|\Psi^{(\hat{r},\hat{l})}_{i^{(l)}_{r}}\right| \tag{24}$$

$$\left|f_{i^{(\hat{l})}_{\hat{r}}}(y^{*(r,l)})\right| = \beta^{*(\hat{r},\hat{l},r,l)} \left|f_{i^{(l)}_{r}}(y^{*(r,l)})\right| \tag{25}$$

$$\left|\Psi^{(r,l)}_{i^{(\hat{l})}_{\hat{r}}}\right| = \gamma^{*(r,l,\hat{r},\hat{l})} \left|\Psi^{(r,l)}_{i^{(l)}_{r}}\right| \tag{26}$$

としたとき，式 (21), (22), (23) が満たされるように $\alpha^{*(\hat{r},\hat{l},r,l)}$, $\beta^{*(\hat{r},\hat{t},r,l)}$, $\gamma^{*(r,l,\hat{r},\hat{t})}$ の値を決定する．任意の所望の発現パターン遷移列に対して，このような値の組が存在することは明らかである．

$\Psi^{(r,l)}$ と $\Psi^{(r+1,l)}$ の間には，式 (18) の関係がある．そのため，(b)-(2) 解軌道がセパラトリクスの存在する領域を連続して通過しなければならないとき，すなわち，$y^{*(r,l)}$, $y^{*(\hat{r},\hat{t})}$, $y^{*(r-1,l)}$, $y^{*(\bar{r},\bar{l})} \in \mathcal{Y}_{sep}$, $y^{*(r,l)} = y^{*(\hat{r},\hat{l})}$, $y^{*(r-1,l)} = y^{*(\bar{r},\bar{l})}$ である場合，$\Psi^{(r,l)}$, $f(y^{*(r,l)})$, $\Psi^{(r-1,l)}$, $f(y^{*(r-1,l)})$ は式 (24), (25), (26) の条件を同時に満たさなければならない．このことと式 (18) より，式 (26) の条件は次式に書き換えられる．

$$\begin{aligned}&\left|\frac{\Psi^{(r-1,l)}_{i^{(\hat{l})}_{\hat{r}}}}{\Psi^{(r-1,l)}_{i^{(l)}_{r-1}}} - \frac{f_{i^{(\hat{l})}_{\hat{r}}}(y^{*(r-1,l)})}{f_{i^{(l)}_{r-1}}(y^{*(r-1,l)})}\right| \\ &\quad = \alpha^{*(r,l,\hat{r},\hat{l})} \left|\frac{\Psi^{(r-1,l)}_{i^{(l)}_{r}}}{\Psi^{(r-1,l)}_{i^{(l)}_{r-1}}} - \frac{f_{i^{(l)}_{r}}(y^{*(r-1,l)})}{f_{i^{(l)}_{r-1}}(y^{*(r-1,l)})}\right|\end{aligned} \tag{27}$$

さらに，(b)-(2)-(ii) $i^{(\bar{l})}_{\bar{r}} = i^{(\hat{l})}_{\hat{r}}$ あるいは (b)-(2)-(iii) $i^{(l)}_{r} = i^{(\hat{l})}_{\hat{r}}$ が成り立つ場合，式 (27) の中に式 (24), (25), (26) の条件で比の絶対値が与えられているものが存在する．(b)-(2)-(ii) の場合，

$$\left|\frac{\Psi^{(r-1,l)}_{i^{(\bar{l})}_{\bar{r}}}}{\Psi^{(r-1,l)}_{i^{(l)}_{r-1}}}\right| = \left|\frac{\Psi^{(r-1,l)}_{i^{(\hat{l})}_{\hat{r}}}}{\Psi^{(r-1,l)}_{i^{(l)}_{r-1}}}\right| = \gamma^{*(r-1,l,\hat{r},\hat{l})} \tag{28}$$

$$\left|\frac{f_{i^{(\bar{l})}_{\bar{r}}}(y^{*(r-1,l)})}{f_{i^{(l)}_{r-1}}(y^{*(r-1,l)})}\right| = \left|\frac{f_{i^{(\hat{l})}_{\hat{r}}}(y^{*(r-1,l)})}{f_{i^{(l)}_{r-1}}(y^{*(r-1,l)})}\right| = \beta^{*(\hat{r},\hat{l},r-1,l)} \tag{29}$$

であるので，式 (27) の条件はつぎのように書き換えられる．

$$\begin{aligned}&\left|\gamma^{*(r-1,l,\bar{r},\bar{l})}_{\mathrm{sign}} - \beta^{*(\bar{r},\bar{l},r-1,l)}_{\mathrm{sign}}\right| \\ &\quad = \alpha^{*(r,l,\hat{r},\hat{l})} \left|\frac{\Psi^{(r-1,l)}_{i^{(l)}_{r}}}{\Psi^{(r-1,l)}_{i^{(l)}_{r-1}}} - \frac{f_{i^{(l)}_{r}}(y^{*(r-1,l)})}{f_{i^{(l)}_{r-1}}(y^{*(r-1,l)})}\right|\end{aligned} \tag{30}$$

ここで，

$$\gamma^{*(r,l,\bar{r},\bar{l})}_{\mathrm{sign}} := \mathrm{sign}\left(\frac{\Psi^{(r,l)}_{i^{(\bar{l})}_{\bar{r}}}}{\Psi^{(r,l)}_{i^{(l)}_{r}}}\right) \gamma^{*(r,l,\hat{r},\hat{l})} \tag{31}$$

$$\beta^{*(\hat{r},\hat{l},r,l)}_{\mathrm{sign}} := \mathrm{sign}\left(\frac{f_{i^{(\hat{l})}_{\hat{r}}}(y^{*(r,l)})}{f_{i^{(l)}_{r}}(y^{*(r,l)})}\right) \beta^{*(\hat{r},\hat{l},r,l)} \tag{32}$$

であり，$\mathrm{sign}(x)$ は x の符号を返す符号関数である．$\frac{\Psi^{(r,l)}_{i^{(\bar{l})}_{\bar{r}}}}{\Psi^{(r,l)}_{i^{(l)}_{r}}}$ と $\frac{f_{i^{(\hat{l})}_{\hat{r}}}(y^{*(r,l)})}{f_{i^{(l)}_{r}}(y^{*(r,l)})}$ の符号は，式 (22) と式 (20) の条件よりそれぞれ一意に定まる．一方，(b)-(2)-(iii) の場合，同様にして式 (27) の条件はつぎの条件書き換えることができる．

$$\begin{aligned}&\left|\frac{\Psi^{(r-1,l)}_{i^{(\hat{l})}_{\hat{r}}}}{\Psi^{(r-1,l)}_{i^{(l)}_{r-1}}} - \frac{f_{i^{(\hat{l})}_{\hat{r}}}(y^{*(r-1,l)})}{f_{i^{(l)}_{r-1}}(y^{*(r-1,l)})}\right| \\ &\quad = \alpha^{(r,l,\bar{r},\bar{l})} \left|\gamma^{(r-1,l,\bar{r},\bar{l})}_{\mathrm{sign}} - \beta^{(\bar{r},\bar{l},r-1,l)}_{\mathrm{sign}}\right|\end{aligned} \tag{33}$$

以上をまとめると，遺伝子ネットワークが所望の発現パターン遷移列をもつための十分条件としてつぎの条件が導かれる．すなわち，$y^{*(r,l)} \in \mathcal{Y}_{sep}$, $y^{*(\hat{r},\hat{l})}$ なるすべての r と l の組に対して，

$$0 < \alpha^{*(\hat{r},\hat{l},r,l)} < \beta^{*(\hat{r},\hat{l},r,l)} < \gamma^{*(r,l,\hat{r},\hat{l})} \tag{34}$$

を満たすように $\alpha^{*(\hat{r},\hat{l},r,l)}$, $\beta^{*(\hat{r},\hat{l},r,l)}$ および $\gamma^{*(r,l,\hat{r},\hat{l})}$ を決めたとき，以下の条件が成り立つ $\Psi^{(0,l)} \in \Omega_{y^{*(0,l)}}$, $l = 1, 2, \cdots, q$ が存在することである．

条件 (a) $y^{*(r,l)} \notin \mathcal{Y}_{sep}$ なる $y^{*(r,l)}$ に対して，式 (19) が成り立つ．

条件 (b)-(1) $y^{*(r,l)} \in \mathcal{Y}_{sep}$，$y^{*(\hat{r},\hat{l})} = y^{*(r,l)}$，かつ $y^{*(r-1,l)}$, $y^{*(\hat{r}-1,\hat{l})} \notin \mathcal{Y}_{sep}$ である $y^{*(r,l)}$ に対して，式 (20), (22), (23), (24), (25) および (26) の条件が成り立つ．

条件 (b)-(2)-(i) $y^{*(r,l)}$ $y^{*(r-1,l)} \in \mathcal{Y}_{sep}$，$y^{*(\hat{r},\hat{l})} = y^{*(r,l)}$, $y^{*(\bar{r},\bar{l})} = y^{*(r-1,l)}$, $i_{\bar{r}}^{(\bar{l})} \neq i_{\hat{r}}^{(\hat{l})}$，かつ $i_r^{(l)} \neq i_{\hat{r}}^{(\hat{l})}$ である $y^{*(r,l)}$ に対して，式 (20), (22), (23), (24), (25), (26)，および (27) の条件が成り立つ．

条件 (b)-(2)-(ii) $y^{*(r,l)}$ $y^{*(r-1,l)} \in \mathcal{Y}_{sep}$，$y^{*(\hat{r},\hat{l})} = y^{*(r,l)}$，$y^{*(\bar{r},\bar{l})} = y^{*(r-1,l)}$，$i_{\bar{r}}^{(\bar{l})} = i_{\hat{r}}^{(\bar{l})}$ である $y^{*(r,l)}$ に対して，式 (20), (22), (23), (24), (25), (26)，および (30) の条件が成り立つ．

条件 (b)-(2)-(iii) $y^{*(r,l)}$ $y^{*(r-1,l)} \in \mathcal{Y}_{sep}$，$y^{*(\hat{r},\hat{l})} = y^{*(r,l)}$，$y^{*(\bar{r},\bar{l})} = y^{*(r-1,l)}$，かつ $i_r^{(l)} = i_{\hat{r}}^{(\hat{l})}$ である $y^{*(r,l)}$ に対して，式 (20), (22), (23), (24), (25), (26)，および (33) の条件が成り立つ．

つぎに，先の 2. で述べたように，任意の所望の発現パターン遷移列に対してこの条件を満たすような $\Psi^{(r,l)}$, $f(y^{*(r,l)})$ の値を決定する手順を示す．まず，条件 (b)-(1) を考える．この条件を満たすように $\Psi_{i_{\hat{r}}^{(\hat{l})}}^{(\hat{r},\hat{l})}$, $\Psi_{i_r^{(l)}}^{(\hat{r},\hat{l})}$, $f_{i_{\hat{r}}^{(\hat{l})}}(y^{*(r,l)})$, $f_{i_r^{(l)}}(y^{*(r,l)})$, $\Psi_{i_{\hat{r}}^{(\hat{l})}}^{(r,l)}$，および $\Psi_{i_r^{(l)}}^{(r,l)}$ の値を決定することが可能であることは明らかである．なお，これらの値を決定すると，$\Psi^{(r+1,l)}$ の定義 (18) より $\Psi_{i_{\hat{r}}^{(\hat{l})}}^{(r+1,l)}$ の値が決まる．

つぎに，条件 (b)-(2)-(i), (b)-(2)-(ii), (b)-(2)-(iii) を考える．$f(y^{*(r-1,l)})$, $f(y^{*(\hat{r}-1,\hat{l})})$, $\Psi^{(r-1,l)}$，および $\Psi^{(\hat{r}-1,\hat{l})}$ のいくつかの要素の値を既に決定しているとする．例えば，$y^{*(r-1,l)}$ が (b)-(1) の条件を満たすパターンであるとすると，$f_{i_{\bar{r}-1}^{(\bar{l})}}(y^{*(r-1,l)})$, $f_{i_{r-1}^{(l)}}(y^{*(r-1,l)})$, $\Psi_{i_{\bar{r}-1}^{(\bar{l})}}^{(r-1,l)}$，および $\Psi_{i_{r-1}^{(l)}}^{(r-1,l)}$ の値を決定しておくことが出来る．よって，条件式の中で値を決定すべきものは，

- (b)-(2)-(i) の場合は，$\Psi_{i_{\hat{r}}^{(\hat{l})}}^{(r-1,l)}$, $f_{i_{\hat{r}}^{(\hat{l})}}(y^{*(r-1,l)})$, $\Psi_{i_r^{(l)}}^{(r-1,l)}$，および $f_{i_r^{(l)}}(y^{*(r-1,l)})$,
- (b)-(2)-(ii) の場合は，$\Psi_{i_r^{(l)}}^{(r-1,l)}$，および $f_{i_r^{(l)}}(y^{*(r-1,l)})$,
- (b)-(2)-(iii) の場合は，$\Psi_{i_{\hat{r}}^{(\hat{l})}}^{(r-1,l)}$，および $f_{i_{\hat{r}}^{(\hat{l})}}(y^{*(r-1,l)})$

である．これらについて，条件を満たすように値を決定できることは明らかである．これらの値を決定すると，$\Psi^{(r+1,l)}$ の定義 (18) より $\Psi_{i_{\hat{r}}^{(\hat{l})}}^{(r,l)}$ と $\Psi_{i_r^{(l)}}^{(r,l)}$ の値が決まり，これらの値は式 (26) の条件を満たす．$y^{*(\hat{r}-1,\hat{l})}$ も (b)-(1) の条件を満たすパターンである場合，同様にして $\Psi_{i_{\hat{r}}^{(\hat{l})}}^{(\hat{r},\hat{l})}$ と $\Psi_{i_r^{(l)}}^{(\hat{r},\hat{l})}$ が決まる．そうでなければこれらの値と残りの $f_{i_{\hat{r}}^{(\hat{l})}}(y^{*(r,l)})$ と $f_{i_r^{(l)}}(y^{*(r,l)})$ の値は，式 (24) と (25) を満たすように決定する．このように，$\Omega_{y^{*(r-1,l)}}$ にセパラトリクスが実現されるように $\Psi^{(\hat{r}-1,\hat{l})}$, $f(y^{*(r-1,l)})$, $\Psi^{(r-1,l)}$ の要素の一部の値が決定されているとしても，$\Omega_{y^{*(r,l)}}$ にもセパラトリクスが実現されるように $\Psi^{(\hat{r},\hat{l})}$, $f(y^{*(r,l)})$, $\Psi^{(r,l)}$ の要素の一部の値を決定することが可能である．このことから，セパラトリクスを実現すべき領域が連続している場合，その最初の領域は (b)-(1) の場合になるので，条件を持たすように必要な値を決定することができ，以降の領域については，前の領域で決めた値を用いて必要な値を決定していけばよい．まとめると，各発現パターン遷移列 l において，$y^{*(r,l)} \in \mathcal{Y}_{sep}$ なるすべての $y^{*(r,l)}$ に対して，r が小さいものから大きなものへと順に $\Psi^{(r,l)}$ と $f(y^{*(r,l)})$ の要素の一部を，先の条件を満たすように決定していくとができる．このステップを [手順 **(1)**] と呼ぶ．

つぎに，各発現パターン遷移列について以下のようにして $r = 0$ から $r = p_l - 1$ まで順に残りの $y^{*(r,l)}$ の要素の値を決定していく．すなわち，まず，$\Psi_{i_r^{(l)}}^{(r,l)}$ と $f_{i_r^{(l)}}(y^{*(r,l)})$ のうち未定のものを，$\Psi^{(r,l)} \in S_r^{(l)}$ と式 (19) あるいは (20) を満たすように決定する．つぎに，$\Psi_i^{(r,l)}$, $\Psi_i^{(r+1,l)}$ と $f_{i_r^{(l)}}(y^{*(r,l)})$ のうち，未定なものの組み合わせによって次のような (I) と (II) の場合分けを行ってこれらの値を決定する．

- (I) $f_{i_r^{(l)}}(y^{*(r,l)})$ が未定な場合，$\Psi_i^{(r,l)}$ と $\Psi_i^{(r+1,l)}$ で未定なものがあれば，$|\Psi_i^{(r,l)}| \geq |\Psi_i^{(r+1,l)}|$, $\Psi^{(r,l)} \in S_r^{(l)}$, $\Psi^{(r+1,l)} \in S_{r+1}^{(l)}$ を満たすように未定な $\Psi_i^{(r,l)}$ と $\Psi_i^{(r+1,l)}$ を決定し，式 (18) を変形した
$$f_i(y^{*(r,l)}) = \Psi_i^{*(r+1,l)} - \frac{\Psi_i^{*(r+1,l)} - \Psi_i^{*(r,l)}}{\Psi_{i_r^{(l)}}^{*(r,l)}} f_{i_r^{(l)}}(y^{*(r,l)}) \tag{35}$$
によって $f_{i_r^{(l)}}(y^{*(r,l)})$ を決定する．このように決定すると，$\Psi_i^{(r,l)}$, $\Psi_i^{(r+1,l)}$ と $f_{i_r^{(l)}}(y^{*(r,l)})$ の符号が一致し，式 (19) あるいは (20) が満たされる．
- (II) $f_{i_r^{(l)}}(y^{*(r,l)})$ の値が決定されている場合，$\Psi_i^{(r,l)}$ の値も既に決定されているので，式 (18) により $\Psi_i^{(r+1,l)}$ の値を決定する．

このステップを [手順 **(2)**] と呼ぶ．

つぎに，先の 3. でのべた [手順 **(2)**] で決定した値をもつような関数 f のパラメータの求め方を述べる．次

の関数を展開すると式 (5) となる．

$$\begin{aligned}f_i(y, w^{(i)}) =& w^{(i)}_{[0\ 0\ \cdots\ 0]'}(1-y_1)(1-y_2)\cdots(1-y_n)\\ &+ w^{(i)}_{[1\ 0\ \cdots\ 0]'}y_1(1-y_2)\cdots(1-y_n)\\ &+\cdots+w^{(i)}_{[1\ 1\ \cdots\ 1]'}y_1y_2\cdots y_n \end{aligned} \tag{36}$$

ここで，$w^{(i)}_y$ の下の添字 y は 2^n 個の発現パターンのうちの一つをとる．例えば，$n=2$ のときは

$$\begin{aligned}f_i(y, w^{(i)}) =& w^{(i)}_{[0\ 0]'}(1-y_1)(1-y_2)\\ &+ w^{(i)}_{[1\ 0]'}y_1(1-y_2)\\ &+ w^{(i)}_{[0\ 1]'}(1-y_1)y_2 + w^{(i)}_{[1\ 1]'}y_1y_2\end{aligned}$$

となる．この表現の場合，$f(y)$ は次式で表される．

$$f_i(y) = w^{(i)}_y,\ \ i=1,2,\cdots,n$$

そこで，$y^{*(p_l,l)}$, $l=1,2,\cdots,q$，および所望の発現パターン遷移列に含まれないパターン y に対して f の値を任意の値に設定する．例えば，すべて 0 とする．以上のようにして，2^n 個のすべての発現パターン y に対して $f(y)$ の値を決定する．すべての y に対して，先に決定した $f(y)$ の値を用いて

$$w^{(i)}_y = f_i(y),\ \ i=1,2,\cdots,n,\ \forall y$$

と設定し，式 (36) の式を展開することで所望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 が得られる．このステップを [手順 **(3)**] と呼ぶ．

以上の議論は任意の所望の発現パターン遷移列に対して成り立つので，設計問題の解が常に存在することが示された．□

4　設計例

所望の発現パターン遷移列が，

$$[1\ 1\ 1\ 1\ 1]' \to [0\ 1\ 1\ 1\ 1]' \to [0\ 0\ 1\ 1\ 1]' \to [0\ 0\ 0\ 1\ 1]' \tag{37}$$

$$[0\ 1\ 1\ 1\ 0]' \to [0\ 1\ 1\ 1\ 1]' \to [0\ 1\ 0\ 1\ 1]' \tag{38}$$

$$[1\ 0\ 1\ 1\ 0]' \to [0\ 0\ 1\ 1\ 0]' \to [0\ 0\ 1\ 1\ 1]' \to [0\ 0\ 1\ 0\ 1]' \tag{39}$$

$$[0\ 0\ 1\ 0\ 0]' \to [0\ 0\ 1\ 1\ 0]' \to [0\ 0\ 0\ 1\ 0]' \tag{40}$$

と与えられたとき，証明で示した手続きにより設計問題の解が得られることを示す．

これらの発現パターン遷移列の場合，$\mathcal{Y}_{sep}=\{y^{*(1,1)}, y^{*(2,1)}, y^{*(1,2)}, y^{*(1,3)}, y^{*(2,3)}, y^{*(1,4)}\}$, $i^{(1)}_1=1$, $i^{(1)}_2=2$, $i^{(1)}_3=3$, $i^{(2)}_1=5$, $i^{(2)}_2=3$, $i^{(3)}_1=1$, $i^{(3)}_2=5$, $i^{(3)}_3=4$, $i^{(4)}_1=4$, $i^{(4)}_2=3$ である．

まず [手順 **(1)**] を行う．

- $y^{*(1,1)}=y^{*(1,2)}$,$y^{*(0,1)}$, $y^{*(0,2)}\notin\mathcal{Y}_{sep}$ であるので，$\alpha^{*(1,2,1,2)}=1$, $\beta^{*(1,2,1,2)}=2$, $\gamma^{*(1,2,1,2)}=4$, $\Psi^{(1,1)}_2=1$, $f_2(y^{*(1,1)})=-1$, $\Psi^{(1,2)}_2=1$ と決定し，式 (22), (23), (24), (25), (26) より，$\Psi^{(1,2)}_3=1$, $f_3(y^{*(1,1)})=-2$, $\Psi^{(1,1)}_3=4$ と決定する．
- $y^{*(1,3)}=y^{*(1,4)}$, $y^{*(0,3)}$, $y^{*(0,4)}\notin\mathcal{Y}_{sep}$ であるので，同様に，$\alpha^{*(1,4,1,3)}=1$, $\beta^{*(1,4,1,3)}=2$, $\gamma^{*(1,3,1,4)}=4$, $\Psi^{(1,3)}_5=-1$, $f_5(y^{*(1,3)})=1$, $\Psi^{(1,4)}_5=-1$ と決定し，式 (22), (23), (24), (25), (26) より，$\Psi^{(1,4)}_3=1$, $f_3(y^{*(1,3)})=-2$, $\Psi^{(1,3)}_3=4$ と決定する．
- $y^{*(2,1)}=y^{*(2,3)}$, $y^{*(1,1)}$, $y^{*(1,3)}\in\mathcal{Y}_{sep}$ であるので，$\Psi^{(2,1)}$, $\Psi^{(2,3)}$ にすでに値を決定している要素がある．それに注意して，条件 (b)-(2)-(i), (b)-(2)-(ii), (b)-(2)-(iii) の該当する条件を満たすように $\alpha^{*(2,3,2,1)}=1$, $\beta^{*(2,3,2,1)}=2$, $\gamma^{*(2,1,2,3)}=4$, $\Psi^{(1,3)}_4=1$, $f_4(y^{*(1,3)})=1$, $\Psi^{(2,3)}_3=1$, $\Psi^{(2,3)}_4=1$ と決定する．

つぎに，[手順 **2**] を行う．各発現パターン遷移列に対して，$r=0$ から順に $r=p_l-1$ まで $y^{*(r,l)}$ の未定な要素の値を決定していき，以下のようになった．

$$\begin{aligned}
\Psi^{(0,1)} &= [1\ 1\ 2\ 1\ 1]',\\
\Psi^{(1,1)} &= [0\ 1\ 4\ 2\ 2]',\\
\Psi^{(2,1)} &= [-1\ 0\ 1\ 4\ 3]',\\
\Psi^{(3,1)} &= [-2\ \ -1\ 0\ 1\ 4]',\\
f(y^{*(0,1)}) &= [-1\ 1\ 6\ 3\ 3]',\\
f(y^{*(1,1)}) &= [-2\ \ -1\ \ -2\ 6\ 4]',\\
f(y^{*(2,1)}) &= [-3\ \ -2\ \ -1\ \ -2\ 5]',\\
\Psi^{(0,2)} &= [-1\ 1\ 1\ 1\ \ -1]',\\
\Psi^{(1,2)} &= [-2\ 1\ 1\ 2\ 0]',\\
\Psi^{(2,2)} &= [-2\ \frac{1}{3}\ 0\ \frac{10}{3}\ \frac{4}{3}]',\\
f(y^{*(0,2)}) &= [-3\ 1\ 1\ 3\ 1]',\\
f(y^{*(1,2)}) &= [-2\ \ -1\ \ -2\ 6\ 4]',\\
\Psi^{(0,3)} &= [1\ \ -1\ 3\ 1\ \ -1]',\\
\Psi^{(1,3)} &= [0\ \ -2\ 4\ 1\ \ -1]',\\
\Psi^{(2,3)} &= [-1\ \ -3\ 1\ 1\ 0]',\\
\Psi^{(3,3)} &= [-\frac{5}{3}\ \ -\frac{8}{3}\ \frac{1}{3}\ 0\ \frac{5}{3}]',\\
f(y^{*(0,3)}) &= [-1\ \ -3\ 5\ 1\ \ -1]',\\
f(y^{*(1,3)}) &= [-2\ \ -4\ \ -2\ 1\ 1]',\\
f(y^{*(2,3)}) &= [-3\ \ -2\ \ -1\ \ -2\ 5]',\\
\Psi^{(0,4)} &= [-1\ \ -1\ 1\ \ -1\ \ -1]',\\
\Psi^{(1,4)} &= [-2\ \ -2\ 1\ 0\ \ -1]',\\
\Psi^{(2,4)} &= [-2\ \ -\frac{8}{3}\ 0\ \frac{1}{3}\ \frac{1}{3}]',\\
f(y^{*(0,4)}) &= [-3\ \ -3\ 1\ 1\ \ -1]',\\
f(y^{*(1,4)}) &= [-2\ 4\ \ -2\ 1\ 1]'
\end{aligned}$$

最後に，[手順 **3**] を行い，設計問題の解の一つを求める．まず，$y^{*(p_l,l)}$, $l=1,2,3,4$ と発現パターン遷移列に現れていないパターン y に対して $f(y)$ の値を決定

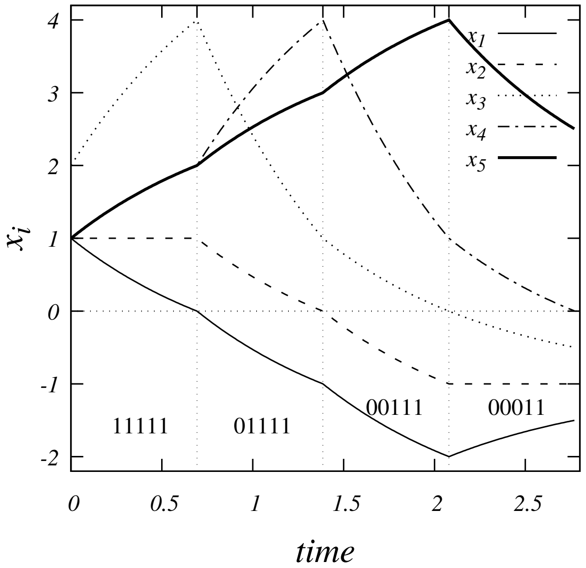


Fig. 2: A simulation result of the synthesized gene regulatory network: the first desired expression pattern sequence.

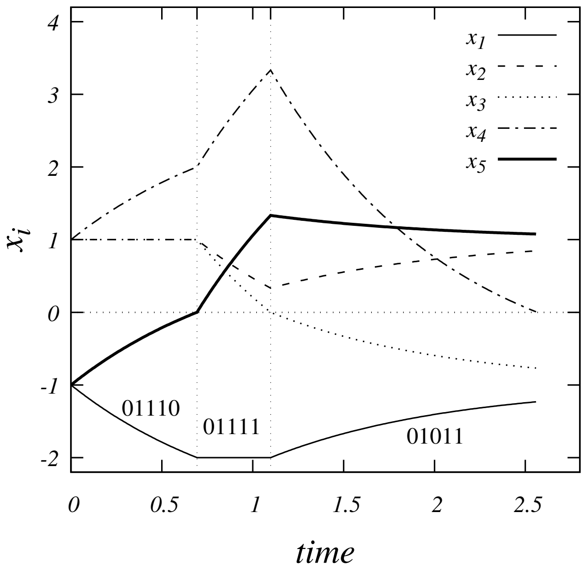


Fig. 3: A simulation result of the synthesized gene regulatory network: the second desired expression pattern sequence.

する．$y^{*(p_l,l)}$, $l = 1, 2, 3, 4$ については，

$$f(y^{*(3,1)}) = [-1\ \ -1\ \ -1\ \ -1\ 1]',$$
$$f(y^{*(2,2)}) = [-1\ 1\ \ -1\ \ -1\ 1]',$$
$$f(y^{*(3,3)}) = [-1\ \ -1\ \ -1\ \ -1\ 1]',$$
$$f(y^{*(2,4)}) = [-1\ 1\ \ -1\ 1\ 1]'$$

とし，残りのパターン y に対しては $f(y) = 0$ とした．このように決定して式 (36) を展開することにより w を求めた．求められたパラメータをもつ遺伝子ネットワークのシミュレーション結果を Fig. 2 から Fig. 5 に示す． 図中の縦の破線は発現パターンが遷移する時刻，図の下部の 0 または 1 の五桁の数値はその時間区間における発現パターンをそれぞれ表す．Fig. 2 から順にそれぞれ所望の発現パターン遷移列である式 (37) から (40) に対応していることがわかる．また，発現パターン

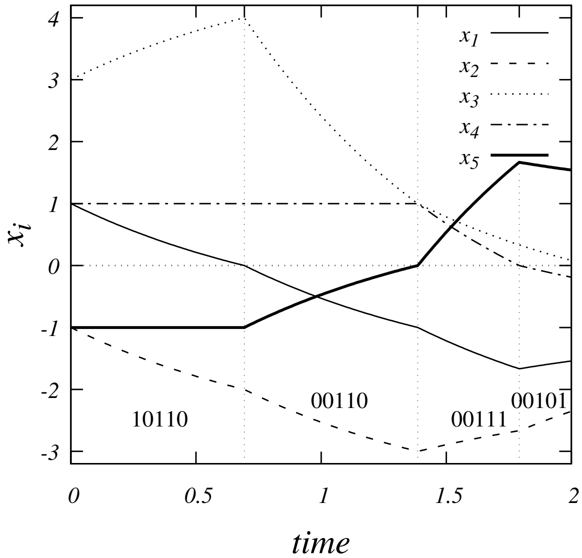


Fig. 4: A simulation result of the synthesized gene regulatory network: the third desired expression pattern sequence.

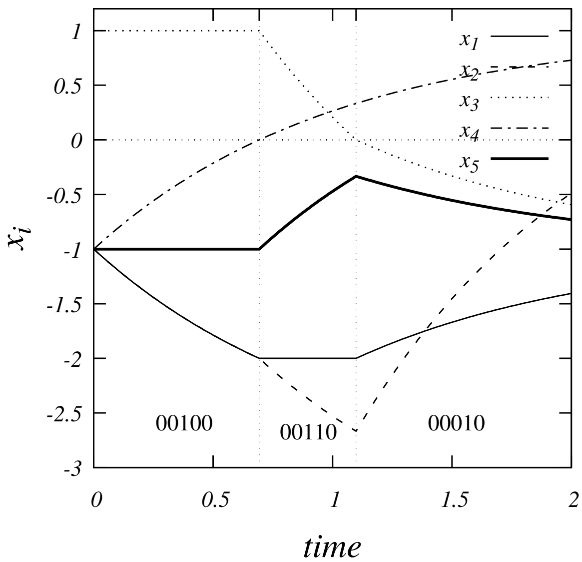


Fig. 5: A simulation result of the synthesized gene regulatory network: the forth desired expression pattern sequence.

が $y^{*(r,l)} \to y^{*(r+1,l)}$ の遷移をするとき，状態が $\Psi^{(r,l)}$ となっていることが確認できる．以上のことから，定理の証明で示した手順により設計問題の解の一つが得られていることが確認できる．

5 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も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ラト

リクスを実現する必要の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ラトリクスを実現する必要のある場合について，解を求める手順を示すことによって解が存在することを示した．

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—セパラトリクスの実現，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)

第11回コンピューテーショナル・インテリジェンス研究会
2017年6月26日-27日, 大津

状態跳躍を導入したMorris-Lecar型ニューロンモデルにおけるカオス誘起現象の検討

○信川創 吉田拓生 （千葉工業大学） 西村治彦 （兵庫県立大学） 山西輝也 （福井工業大学）

Investigation of Chaotic Spiking Activity in Morris-Lecar Type Neuron Model with State Dependent Jump

∗S. Nobukawa, T. Yoshida (Chiba Institute of Technology), H. Nishimura (University of Hyogo), and T. Yamanishi (Fukui University of Technology)

Abstract– Several hybrid spiking neuron models, which combine continuous spike-generation mechanisms and discontinuous resetting process after spiking, have been proposed. Izhikevich neuron model as this kind of model can reproduce many spiking patterns. It has also been revealed that this model has various kinds of bifurcation and routes to chaos under the effect of the state dependent jump in the resetting process. In response to this situation, we have further gotten interested in the relation between chaotic behaviors and the state dependent jump. In this paper, we approach the subject from the comparison of spiking neuron models without the resetting process and with it. We first adopt a continuous Morris-Lecar type spiking neuron model where the orbit at spiking state does not exhibit the divergent behavior and next insert the resetting process to it.

Key Words: chaos, hybrid spiking neuron model, saltation matrix, Lyapunov exponent

1 はじめに

これまで，脳・神経系においてはrateコーディングによって情報処理がなされているとされてきた．しかし，近年，temporalコーディングやpopulationコーディングといった多様な神経コーディングが存在し，これらが柔軟な情報処理を支えていることが示さている[1)]．このような情報処理である記憶や学習のメカニズムに関連した研究には，多様なコーディングを膜電位の発火活動によって実現できるスパイキングニューロンモデルが広く用いられている[2, 3, 4, 5)].

最も重要なスパイキングニューロンモデルとして知られているHodgkin-Huxley(HH)モデル[6)]は，細胞膜のキャパシタンスやイオンチャネルのレジスタンス特性を記述することで，ニューロダイナミクスを再現する．しかし，その方程式系の複雑さから，大規模なスパイキングニューラルネットワークでのシミュレーションにおいては計算コストが大きく，また，小規模なスパイキングニューラルネットワークであっても，解析的な分析方法が制限される．そこで，FitzHugh-NagumoニューロンモデルやHindmarsh-Roseモデルのように，発火活動に着目したHHモデルよりも単純な連続的な常微分方程式で記述されるモデルが提案されている[1)].

一方，ニューロンの発火状態から過分極状態への遷移をリセット動作(状態跳躍)とし，その跳躍と連続的なシステム挙動のダイナミクスを組み合わせたハイブリットなスパイキングニューロンモデルが提案されている[7)]．このモデルの1つであるIzhikevichニューロンモデルは，従来のFizHugh–Nagumoニューロンモデルのような連続的なダイナミクスのみに従うモデルと比較して，リセット動作を含む少数のパラメータ調整によって，多様なスパイクパターンを再現できることが知られている[8, 9, 10)]．更に，状態跳躍の効果により，スパイクパターンだけでなく，多様な分岐やカオスへのルートが存在することが明らかとなっている[11, 12, 13, 14, 15)]．このようなIzhikevichニューロンモデルの優れた特性は，連続なシステムに対する状態跳躍の導入による軌道への影響に着目することで分析できると考えられる．しかしながら，このモデルでは，リセット動作を除いた場合，休止解と発散解しか存在しないため，リセット動作の導入前後でのアトラクタの構造的な変化を評価することはできない．

我々はこれまでに，状態跳躍を除いてもHopf分岐により生成されたリミットサイクルが存在するFizHugh–Nagumoニューロンモデルに対して，状態跳躍を導入し，状態跳躍の連続軌道に対する影響を調べてきた[16)]．それに対して，本研究ではHopf分岐だけでなく，sadle-node分岐も許容するMorris-Lecar型ニューロンモデルにおいて，各分岐によって生成されたリミットサイクルに対する状態跳躍によって誘起されたカオスルートについての検討を行う．

2 モデルと評価指標

2.1 リセット動作を伴うスパイキングニューロンモデル

FitzHugh-Nagumoニューロンモデルは，(1),(2)式の2変数常微分方程式で表される[17, 18)].

$$\dot{v} = v(a-v)(v-1) - u + I \tag{1}$$
$$\dot{u} = bv - cu \tag{2}$$

ここで，vはニューロンの膜電位をuはmembrane recovery変数，Iは直流入力成分を表している．パラメータaはv-nullclineの形状決定を担う．b/c, cがuの感度と時定数をそれぞれ表している．神経系において，休止状態から発火状態の遷移はHopf分岐とsaddle-node分岐によって生じる[1, 10)]．これらの両方の分岐を実現するためには，$\dot{u}$として，(3)式のようなvのシグモイド関数が必要となる[10)].

$$\dot{u} = \alpha\left(\frac{1}{1+\exp(-(v-\beta)/\epsilon)} - u\right), \tag{3}$$

ここで，α は u の時定数を，β と ϵ はそれぞれシグモイド関数の形状決定を担う．本稿では $\dot{u}$ の関数に (2) 式ではなく，(3) 式を用いる．また，パラメータについては，$(a, \alpha, \epsilon) = (0.1, 0.1, 0.05)$ と設定する．

次に，このモデルに対して (4) 式で記述されるリセット動作による状態跳躍を導入する．

$$\text{if } v \geq v_{\text{peak}}, \text{ then} \begin{cases} v \leftarrow v_r \\ u \leftarrow u + d \end{cases} \tag{4}$$

ここで，(4) 式は v_{peak} を v の最大値と設定した場合，$v_r \to v_{\text{peak}}, d \to 0$ の極限で跳躍軌道が連続となる．また，パラメータ d は 0.01 と設定する．以下では，後退差分法で離散化を行い，跳躍点をニュートン法により特定する手法 [19] を用いて軌道の計算を行う．

2.2 評価尺度

システムのカオス性の評価と分岐解析のために，跳躍行列を考慮に入れた状態遷移行列を用いる．連続軌道上 (発火 (跳躍) と発火の間) における (1), (3) 式の変分方程式は状態遷移行列 $\Phi_i(t, t_{i-1})$ とヤコビアン $J(v, u, t)$ を用いて，(5), (6) 式のように表される．

$$\dot{\Phi}_{i+1}(t, t_i) = J(v, u, t)\Phi_{i+1}(t, t_i), \tag{5}$$

$$\Phi_{i+1}(t_i, t_i) = E \tag{6}$$

ここで，t_i は i 番目の発火が生じた時刻，E は単位行列を表している．跳躍前後の (v, u) 値をそれぞれ (v^+, u^+), (v^-, u^-) とすると，跳躍行列は (7) 式で表される [12]．

$$S_i = \begin{bmatrix} \frac{\dot{v}^+}{\dot{v}^-} & 0 \\ \frac{\dot{u}^+ - \dot{u}^-}{\dot{v}^-} & 1 \end{bmatrix} \tag{7}$$

$[T^k : T^{k+1}]$ で発火によるリセット動作により状態跳躍が生じた場合の状態遷移行列 $\Phi^k(t^{k+1}, T^k)$ $(k = 0, 1, \cdots, N-1)$ は跳躍行列による補正を考慮すると (8) 式のようになる．

$$\begin{aligned} \Phi^k(T^{k+1}, T^k) = \Phi_{i+1}(T^{k+1}, t_i)S_i\Phi_i(t_i, t_{i-1}) \\ \cdots S_2\Phi_2(t_2, t_1)S_1\Phi_1(t_1, T^k) \end{aligned} \tag{8}$$

(8) 式の評価時間間隔を τ とし，それぞれの評価時間間隔における状態遷移行列 $\Phi^k(T^{k+1}, T^k)$ の固有値を $l_j^k(k = 1, 2, \cdots, N)$ とすると，リアプノフスペクトル λ_j は (9) 式で表される．

$$\lambda_j = \frac{1}{T^N - T^0} \sum_{k=0}^{N-1} \log(|l_j^k|) \tag{9}$$

以下のシミュレーションでは，$T^{k+1} - T^k$ の周期を 20 回分の発火が生じる期間，もしくは 20 回の発火前に $T^{k+1} - T^k$ が 1000 に到達した場合は 1000 に設定する．

更に，$v = v_{\text{peak}}$ にポアンカレ断面を設定し，その断面上での u 値 $u_i(i = 1, 2, \cdots)$ の挙動についても評価を行う．u_i から l 回目にポアンカレ断面を通過する u 値は，ポアンカレ写像 ϕ により，$u_{i+l} = \psi^l(u_i)$ で表される．そして，条件 $u_0 = \psi^l(u_0)$ を満たす固定点の安定性は文献 [11] より，初期値 $_{\mathbf{0}} = (v_0, u_0)$ による軌道 (v, u) の微分から (10) 式によって計算される．

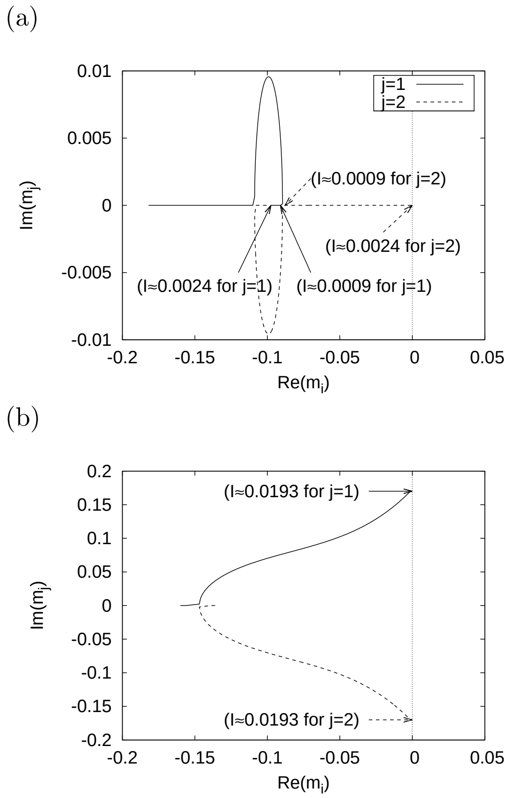


Fig. 1: Dependence of m_j on parameter I in regions #1 (a) and #2 (b) ($a = 0.1, \alpha = 0.1, \epsilon = 0.05$, $\beta =$ 0.5(region #1),0.3(region #2)).

$$\mu^l = \frac{\partial \phi^l}{\partial \ _0} = \begin{pmatrix} 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 \\ -\dot{v}/\dot{u} & 1 \end{pmatrix} \Phi(t_l, t_0) \begin{pmatrix} 0 \\ 1 \end{pmatrix} \tag{10}$$

ここで，t_0 は $_0$ の時刻とする．これにより，$|\mu^l| < 1$ が固定点の安定条件となり，$\mu^l = -1, 1$ で周期倍分岐と接線分岐の発生をそれぞれ判定できる．

3 結果と評価

3.1 連続的な is-L ca 型ニューロンモデルにおける分岐

リミットサイクルの生成における分岐を，固定点の近傍における (1), (3) 式のヤコビアン $J(v, u, t)$ の固有値 $m_j(j = 1, 2)$ を用いて調べる．Fig.1 (a) は領域#1 において I を $[-0.005 : 0.005]$ の区間で動かした場合の安定固定点周りの m_j である．$0.0009 \lesssim I \lesssim 0.0024$ において，m_2 は I が大きくなるにつれて実軸上を正の方向に移動し，$I \approx 0.0024$ で虚軸を横切り，安定固定点が不安定化する．一方，領域#2 において，I を $[-0.005 : 0.02]$ の区間で動かすと $I \approx 0.0193$ において，複素共役な $m_{1,2}$ が虚軸を横切り，安定固定点が不安定化している．前者の分岐は saddle-node 分岐，後者の分岐は Hopf 分岐と呼ばれる．

3.2 リセット動作を伴う is-L ca 型ニューロンモデルでの分岐とカオス

本節では，(1), (3) 式の連続的なスパイキングニューロンモデルの Fig. 1 での $I = 0.004$ (領域#1)，$I = 0.02$ (領域#2) の設定に対して，(4) 式によるリセット動作

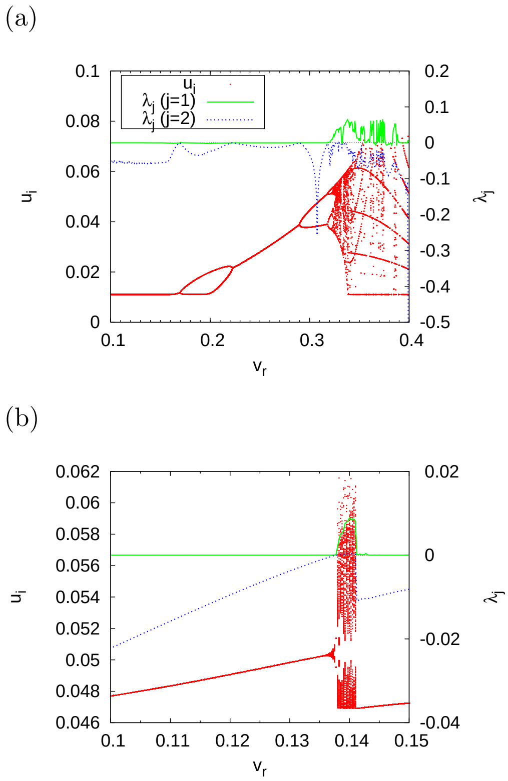


Fig. 2: Bifurcation diagram and Lyapunov exponents λ_j $(j = 1, 2)$ as functions of v_r around the chaotic region $(a = 0.1, \alpha = 0.1, \epsilon = 0.05)$. (a) Region #1 case $(\beta = 0.5, I = 0.004, v_{\mathrm{peak}} = 0.4, d = 0.01)$. (b) Region #2 case $(\beta = 0.3, I = 0.02, v_{\mathrm{peak}} = 0.225, d = 0.01)$.

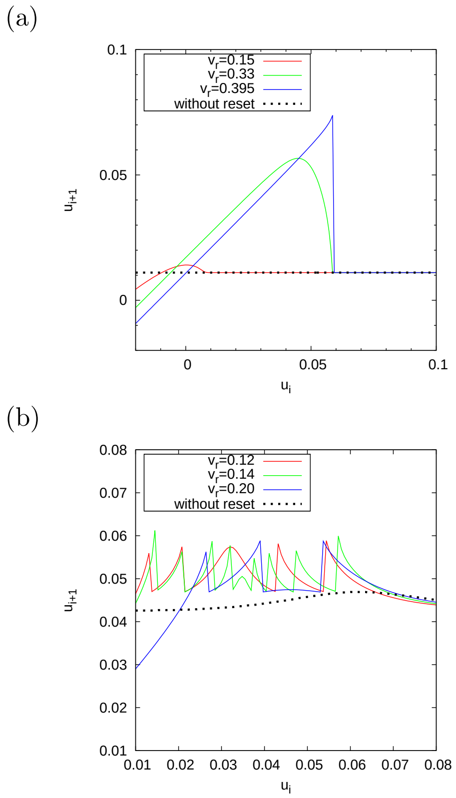


Fig. 3: Dependence of the return map $u_{i+1} = \psi(u_i)$ on parameter v_r in the cases of regions #1 (a) and #2 (b). $(a = 0.1, \alpha = 0.1, \epsilon = 0.05, d = 0.01$, region #1: $\beta = 0.5, I = 0.004, v_{\mathrm{peak}} = 0.4$, region #2: $\beta = 0.3, I = 0.02, v_{\mathrm{peak}} = 0.225)$

を導入したハイブリットな Morris-Lecar 型ニューロンモデルにおけるシステム挙動について調べる．Fig. 2 に，領域#1 において $v_{\mathrm{peak}} = 0.4$(Fig.(a))，領域#2 において $v_{\mathrm{peak}} = 0.225$(Fig.(b)) に設定した場合の u_i の分岐図と λ_j $(j = 1, 2)$ のパラメータ v_r への依存性を示す．これらの結果から，u_i が不規則な挙動を示し，$\lambda_1 > 0, \lambda_2 = 0$ となるカオス状態が，領域#1 においては $0.322 \lesssim v_r \lesssim 0.388$ で，領域#2 においては，$0.136 \lesssim v_r \lesssim 0.141$ で出現する様子が確認できる．更に，これらのカオスへのルートを μ^l を用いて調べると，領域#1 においては，周期倍分岐 $(\mu^l = -1)$ が $v_r \approx 0.288$ $(l = 1)$, 0.318 $(l = 2)$, 0.322 $(l = 4)$... で，接線分岐 $(\mu^l = 1)$ が $v_r \approx 0.388$ $(l = 5)$ で生じている．このことから，領域#1 で見られるカオスは，v_r が小さくなる，すなわちリセット動作による状態跳躍の距離が大きくなると $v_r \approx 0.388$ において，接線分岐によるルートによってカオス状態が生成される．そしてさらにその距離が大きくなると $v_r \approx 0.322$ において，周期倍分岐によるルートによって周期状態に移る．一方，領域#2 においては，接線分岐が $v_r \approx 0.141$ $(l = 1)$ と $v_r \approx 0.136$ $(l = 2)$ で生じている．このことから，この領域においても跳躍距離が大きくなることで，カオス状態が生成される．更に，その距離が大きくなると，領域#1 とは対照的に接線分岐によって周期状態に遷移することが確認された．

次に，ポアンカレリターンマップ $u_{i+1} = \psi(u_i)$ の跳躍パラメータ v_r への依存性について評価を行う．Fig.3 (a) に，領域#1 における $v_r = 0.15, 0.33, 0.395$ とリセット動作を除いた場合における ψ を示す．リセット動作のない状態 (点線) においては，$u_{i+1} = \psi(u_i) \approx 0.01$ となるが，リセット動作が導入されると区分的に線形な引伸しと折畳み構造が現れる ($v_r = 0.395$ (青線))．そして，v_r が小さくなる，すなわち跳躍距離が増加すると $u_i \approx 0.06$ での折畳み構造に非線形性が現れるようになる ($v_r = 0.33$ (緑線))．更に v_r が小さくなると折畳みの構造が消失する様子が確認できる ($v_r = 0.15$ (赤線))．一方，領域#2 においては (Fig.3 (b))，リセット動作のない状態 (点線) では，領域#1 と同様に $u_{i+1} = \psi(u_i) \approx 0.04$ となが．リセット動作を導入すると 3 つの引伸しと折畳み構造が現れる ($v_r = 0.2$(青線))．そして，その折畳みの回数は v_r の減少 (跳躍距離の増加) にともなって増加する ($v_r = 0.14$(緑線))．更に，v_r が減少すると $v_r = 0.12$(青線) のように折畳み回数は減少する．

更に，このポアンカレリターンマップの u-nullcline に関するパラメータ ϵ への影響についても調べる．Fig.4 (a) にパラメータ v_r を 0.33 に固定した場合の領域#1 の場合の結果を示す．ϵ が大きな場合，すなわち u-nullcline がステップ関数に近づくと，$\epsilon = 0.053$ (赤線) のように区分的に線形な引伸しと折畳みの構造が現れ，ϵ の減少に伴い，折畳み構造に非線形性が現れる様子が確認できる ($\epsilon = 0.05$(緑線), 0.03(青線))．一方，Fig.4 (b) にパラメータ v_r を 0.14 に固定した場合の領域#2 の場合の結果を示すと，$\epsilon = 0.05$ の場合は 9 回の折畳み構

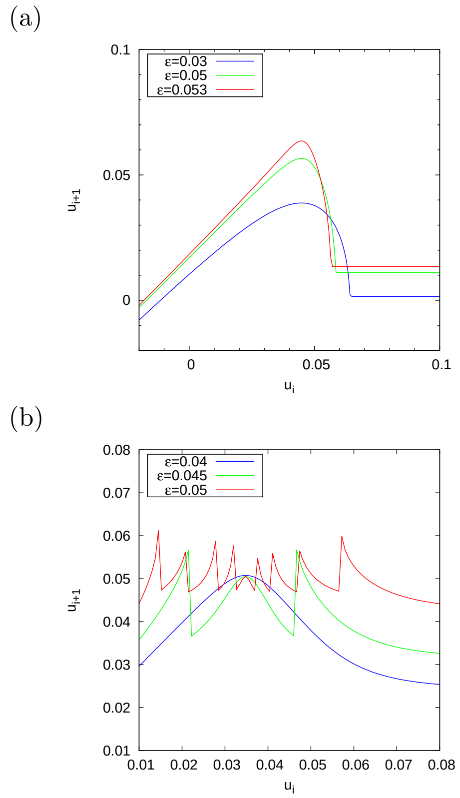


Fig. 4: Dependence of the return map $u_{i+1} = \psi(u_i)$ on parameter ϵ in the cases of regions #1 (a) and #2 (b). ($a = 0.1, \alpha = 0.1, d = 0.01$, region #1: $\beta = 0.5, I = 0.004, v_{\mathrm{peak}} = 0.4, v_r = 0.33$, region #2: $\beta = 0.3, I = 0.02, v_{\mathrm{peak}} = 0.225, v_r = 0.14$)

造が現れるが，ϵ の減少に伴って折畳み回数が減少する様子が確認できる ($\epsilon = 0.045$(緑線), 0.04(青線)).

4 おわりに

本稿では，連続的な Morris-Lecar 型のスパイキングニューロンモデルに対してリセット動作を導入し，状態跳躍の距離を大きくしていくことで誘起されたカオスルートについての検討を行った．そのために，まず連続的な MorrisLecar 型のスパイキングニューロンモデルの休止状態から発火状態への移行時の分岐について，安定固定点周りのヤコビアンの固有値を利用し解析した．その結果，saddle-node 分岐によって生じる領域#1 と Hopf 分岐によって生じる領域#2 が存在することが明らかとなった．

次に，ハイブリットな Morris-Lecar 型のスパイキングニューロンモデルに対して，跳躍行列を考慮に入れたリアプノフ指数 λ_j と固定点の安定性評価指標 μ^l を用いて，カオス性と分岐についての解析を行った．その結果，状態跳躍の距離が大きくなるにつれて，領域#1 では接線分岐からカオス状態が出現するが，更にその距離が大きくなると周期倍分岐によって，周期状態へ移行することが確認された．一方，領域#2 では，状態跳躍の距離が大きくなると領域#1 と同様に接線分岐からカオス状態に遷移するが，更にその距離大きくなると，周期倍分岐ではなく，接線分岐によって周期状態に移ることが観察された．

更に，このようなカオスへの分岐について，跳躍パラメータに対するポアンカレ写像の変化を評価したところ，領域#1 では跳躍効果によって引伸しと折畳み構造が追加され，跳躍距離が大きくなるにつれて，その構造に非線形性は付加されていくことを確認した．領域#2 においては，領域#1 と同様に跳躍効果によって引伸しと折畳み構造が付加されるが，跳躍距離が大きくなるにつれて，折畳み回数が増加していくことが明らかとなった．また，このようなアトラクタ構造の変化は回復変数の形状を決定するパラメータに対しても依存性を持つことを確認した．

以上の結果から，連続な Morris-Lecar 型のスパイキングニューロンモデルにおける発火状態への分岐の違いにより，リセット動作の導入によって，異なる特性を持ったカオス領域が出現することが明らかとなった．今後の課題としては，シナプスレベルでの状態跳躍を含んだニューラルシステムにおけるカオス・分岐解析が挙げられる．

謝辞

本研究は JSPS 科研費 若手研究（B）(15K21471) の助成を受けた．

参考文献

1) M. I. Rabinovich, P. Varona, A. I. Selverston, and H. D. Abarbanel, “Dynamical principles in neuroscience,” *Reviews of modern physics*, vol. 78, no. 4, pp. 1213–1265, 2006.

2) N. Schweighofer, K. Doya, H. Fukai, J. V. Chiron, T. Furukawa, and M. Kawato, “Chaos may enhance information transmission in the inferior olive,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101, no. 13, pp. 4655–4660, 2004.

3) N. Hiratani, J.-N. Teramae, and T. Fukai, “Associative memory model with long-tail-distributed hebbian synaptic connections,” *Frontiers in computational neuroscience*, vol. 6, 2012.

4) J. Mejias and A. Longtin, “Optimal heterogeneity for coding in spiking neural networks,” *Physical Review Letters*, vol. 108, no. 22, p. 228102, 2012.

5) S. Nobukawa and H. Nishimura, “Chaotic resonance in coupled inferior olive neurons with the Llinás approach neuron model,” *Neural Computation*, vol. 28, no. 11, pp. 2505–2532, 2016.

6) A. L. Hodgkin and A. F. Huxley, “A quantitative description of membrane current and its application to conduction and excitation in nerve,” *The Journal of physiology*, vol. 117, no. 4, pp. 500–544, 1952.

7) K. Aihara and H. Suzuki, “Theory of hybrid dynamical systems and its applications to biological and medical systems,” *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 368, no. 1930, pp. 4893–4914, 2010.

8) E. M. Izhikevich, “Simple model of spiking neurons,” *IEEE Transactions on neural networks*, vol. 14, no. 6, pp. 1569–1572, 2003.

9) ——, “Which model to use for cortical spiking neurons?” *IEEE transactions on neural networks*, vol. 15, no. 5, pp. 1063–1070, 2004.

10) ——, *Dynamical systems in neuroscience*. MIT press, 2007.

11) A. Tamura, T. Ueta, and S. Tsuji, “Bifurcation analysis of Izhikevich neuron model,” *Dynamics of continuous, discrete and impulsive systems, Series A: mathematical analysis*, vol. 16, no. 6, pp. 759–775, 2009.

12) F. Bizzarri, A. Brambilla, and G. S. Gajani, “Lyapunov exponents computation for hybrid neurons,” *Journal of computational neuroscience*, vol. 35, no. 2, pp. 201–212, 2013.

13) S. Nobukawa, H. Nishimura, T. Yamanishi, and J.-Q. Liu, "Analysis of chaotic resonance in Izhikevich neuron model," *PloS one*, vol. 10, no. 9, 2015.

14) 信川創, 西村治彦, 山西輝也, 劉健勤, "状態跳躍を伴うスパイキングニューラルシステムにおけるカオス評価法の検討," システム制御情報学会論文誌, vol. 29, no. 5, pp. 2010–215, 2016.

15) S. Nobukawa, H. Nishimura, and T. Yamanishi, "Chaotic resonance in typical routes to chaos in the Izhikevich neuron model," *Scientific Reports*, vol. 7, no. 1, p. 1331, 2017.

16) 信川創, 西村治彦, 山西輝也, "ハイブリット化された FitzHugh–Nagumo ニューロンモデルにおけるカオスルートの解析," システム制御情報学会論文誌, vol. 30, no. 5, pp. 167–174, 2017.

17) R. FitzHugh, "Impulses and physiological states in theoretical models of nerve membrane," *Biophysical journal*, vol. 1, no. 6, pp. 445–466, 1961.

18) J. Nagumo, S. Arimoto, and S. Yoshizawa, "An active pulse transmission line simulating nerve axon," *Proceedings of the IRE*, vol. 50, no. 10, pp. 2061–2070, 1962.

19) A. C. Hindmarsh, P. N. Brown, K. E. Grant, S. L. Lee, R. Serban, D. E. Shumaker, and C. S. Woodward, "Sundials: Suite of nonlinear and differential/algebraic equation solvers," *ACM Transactions on Mathematical Software (TOMS)*, vol. 31, no. 3, pp. 363–396, 2005.

第11回コンピューテーショナル・インテリジェンス研究会
2017年6月26日-27日, 大津

適合度を用いたRe-labeling Differential Evolutionの高効率化

○有木大悟　船木亮平　村田純一（九州大学）

Re-labeling Differential Evolution Using Fitness Value for Combinatorial Problem

* D. Ariki, R. Funaki and J. Murata (University of Kyushu)

Abstract－　Re-labeling Differential Evolution (RLDE) is one of Differential Evolution (DE) techniques extended to combinatorial optimization problems (COP). RLDE solves COP with re-labeling alleles, which are indices for distinguishing objects of the combination, by their frequency in all individuals. In this paper, the authors propose two techniques for re-labeling alleles using fitness value. Proposed RLDEs are expected to re-label alleles with accuracy by fitness value instead of frequency of the alleles in the case of problems that can calculate fitness value, and compared with original RLDE and DE.

Key Words:　Differential Evolution, Re-labeling Differential Evolution, Combinatorial Optimization

1　はじめに

進化化計算とは，生物の進化のメカニズムから着想を得た最適化手法で，選択，交叉，突然変異などをモデルとした演算を繰り返すことで最適解を求める多点探索手法である(1)．解の候補を個体とみなし，環境に適した個体の特徴を引き継いだ子個体を繰り返し生成することで，最適解を発見する．進化計算において探索の手掛かりとして必要な情報は，目的関数から計算される環境への適合度のみであり，目的関数が微分不可能，あるいは，目的関数が未知であっても最適化可能である．進化計算では，解のことを個体，解配列の要素を遺伝子座，解配列の値を遺伝子，遺伝子がとりうる値を対立遺伝子と呼ぶ．代表的な進化計算に遺伝的アルゴリズムや遺伝的プログラミング(2)，差分進化（Differential Evolution, DE)(3)などが挙げられる．

DEとは，1995年にR. StornとK. Priceによって提案された手法で進化計算の1つである．個体集団からランダムに選んだ2個体の差分ベクトルを用い，実数値空間を探索する数値最適化手法である．個体が広く分散している探索序盤では，個体間の差分ベクトルは大きくなるため，大域的探索を行う．個体が収束してくると差分ベクトルは小さくなり, 局所的探索を行う．そのため，DE は大域的探索から局所的探索へ円滑に移行するため，優れた探索性能を持つ．また，DE は対比較評価で個体の選択を行う．つまり，2 つの解候補のうちどちらが優れているかという情報のみで最適化を行うため，全個体の適合度を必要とせずに，最適化を行うことができる．この性質を利用して，人間が直接評価を行わなければならない最適化問題の場合には，ユーザは提示された2個体のうち良いと思う方を選択することを繰り返すだけで最適化を行うことが出来るため, ユーザの疲労を軽減することが可能である．これらの特性から， DEを様々な最適化問題に拡張させれば，効率よく解を求めることが期待できる．

実数値最適化問題向けの解法であるDEを組み合わせ最適化問題に適用する研究も行われている。組み合わせ最適化問題とは，探索空間が離散的であり，解が順序や割当のように組み合わせの構造を持つ問題のことである．代表的な問題の例として，最短経路問題や配送計画問題，施設配置問題，スケジューリング問題などが挙げられる．先行研究では，個体間差分を新たに適切に定義することによってDEを組み合わせ最適化問題に拡張した手法 Re-labeling Differential Evolution（RLDE）が提案されている(4)．この手法は，探索を行いながら，遺伝子の番号を割り振り直すことで効率的に解を求めることを目指した手法である．探索時に分かる遺伝子の出現回数が多い順に，遺伝子の番号を割り振り直すことにより，適した差分を定義している．

RLDEは対比較評価の対話型差分進化向けに作られているため，適合度が分からない問題に対して用いることを前提に作られている．そのため，出現回数の多い遺伝子ほど優れた適合度の要因になっているとみなし, 適合度の代わりに出現回数を用いている. しかし，適合度が計算から求められる問題に対しては，遺伝子の出現回数よりも，適合度を利用することで，より正確に対立遺伝子に番号を割り振り直すことが期待できる．

本研究では，RLDEの探索性能の向上を目的に，既存手法のRLDEに対して適合度を組み合わせた手法の提案を行う．そして，シミュレーションによって提案手法を従来手法と比較し，性能の評価を行う．結果を評価するためのテスト問題として，変数に依存関係がない最適化問題として簡単な問題と変数に依存関係のある複雑な最適化問題の2種類の問題を使用する．

2　差分進化（Differential Evolution, DE）

DE は，集団からランダムに選択した 2 個体の差分ベクトルを用いることで効果的な探索を実現できている．差分ベクトルは突然変異での変異の大きさとみなすことが出来る．個体が広く分散している探索序盤では，個体間の差分ベクトルは大きくなる傾向となり，

探索領域を広く探索する大域的探索を行う．個体が最適解に集中してくると，個体間の差分ベクトルは小さくなっていき, 局所的探索へと変化する. このように,探索の進行の段階によって自動的に変異の大きさが調整されるため，無駄のない効率的な探索を実現している．また，対話型進化計算では，評価時のユーザ疲労が課題となっており，対比較評価は従来主流だった全体比較に比べてユーザの疲労を軽減する効果があるとされて，注目されている．

近年，実数値空間での数値最適化手法として DE の研究が盛んに行われている．その主な理由としては,大域的探索から局所的探索へ円滑に移行し，優れた探索性能を持つこと，単純な算術演算に基づいているため，高速に動作すること，制御パラメータがスケーリングファクタ F，交叉率 CR，個体集団のサイズ N の 3 つのみと単純であることが挙げられる．

以下に DE の処理手順を示す.

1. ランダムに初期個体集団を生成する.
2. 親となる個体を集団から 1 個体選択し，target vector とする.
3. target vector を除いた集団の中から 3 個体を選択し，それぞれを base vector，Xr1，Xr2 とする.
4. base vector に F によってスケールされた Xr1 と Xr2 の差分ベクトルを加えることで，mutant vector を作成する．(式 1)

$$\text{mutant vector} = \text{base vector} + F \times (Xr1 - Xr2) \quad (式 1)$$

(スケーリングファクタ F は 0 よりも大きな実数値とする．)

5. mutant vector と target vector を交叉させ, trial vector を作成する.
6. trial vector と target vector を比較し，優れた適合度を持つ方を次の世代に残す.
7. 2〜6 までをすべての個体に対して実行する.
8. 世代が移り 2 の操作に戻る.

手順 3 での base vector，Xr1，Xr2 をどのようにして選択するかによって探索性能が変化する. 本研究では, base vector，Xr1，Xr2 の 3 個体ともランダムに選択することで大域探索性を高くする方法（random vector）と，base vector は集団の最良個体を選択し，Xr1，Xr2 はランダムに選択することで局所探索性を高くする方法（best vector）の 2 つの方法を用いた.

手順 5 での交叉は，Binomial Crossover と呼ばれる交叉の方法を使用した.

3 Re-labeling Differential Evolution（RLDE）

DE は差分ベクトルの大きさによって大域的探索，局所的探索かを調節する．大域的探索では，探索領域の広範囲を大まかに探索し，最適解がある可能性の高い領域を絞り込む．局所的探索では，その周辺を集中的に探索する．個体間の差分が小さくなると局所的探索に移行するため，DE は個体間の距離が近ければ,適合度も近くなるという前提が必要である．DE の個体間の距離はユークリッド距離で定義されているため,遺伝子がどの程度近いかという情報が距離の定義とされる.

RLDE は，探索を行いながら，遺伝子の番号を再割り振りすることで効率的に解を求めることを目標とした手法である．探索を行いながら，個体の分布情報を調べることで適合度が近い個体同士の差分が小さくなるように遺伝子の番号を変更する．DE では，良い適合度の個体は次の世代に引き継がれ，悪い適合度の個体は淘汰される．これを繰り返すことによって最適解周辺に集団が収束していき，遺伝子座に注目すると,良い適合度になりやすい遺伝子の割合が集団全体で高くなる．多くの個体に出現する遺伝子は，優れた適合度の要因となる遺伝子であり，出現回数が少なくなるに従って，劣った遺伝子ということになる．よって,集団での出現回数が多い順に対立遺伝子を並び替え,その順番を新しい番号とすることで，番号が近いと適合度も近くなる．この操作をすべての遺伝子座に対して行う.

以下に RLDE の処理手順を示す

1. ランダムに初期個体集団を生成する.
2. 親となる個体を集団から 1 個体選択し，target vector とする.
3. target vector を除いた集団の中から 3 個体を選択し，それぞれを base vector，Xr1，Xr2 とする.
4. mutant vector を作成する.
5. mutant vector と target vector を交叉させ, trial vector を作成する.
6. trial vector と target vector を比較し，優れた適合度を持つ方を次の世代に残す.
7. 2〜6 までをすべての個体に対して実行する.
8. 遺伝子座毎に集団内の対立遺伝子の出現回数を数え，多い順に対立遺伝子の識別番号を割り振る.
9. 世代が移り 2 の操作に戻る

手順 3 での個体の選択方法や手順 5 での交叉の方法は，DE と同じ方法で行った.

4 提案手法

4.1 提案手法 1：Re-labeling Differential Evolution fitness (RLDE fitness)

RLDE では，出現回数が多い遺伝子は，優れた遺伝子が含まれていると想定している．しかし，この手法では，良い適合度の個体には，優れた遺伝子が含まれていることを前提として考え，遺伝子座毎に，遺伝子

番号の再割り振りを行う．対立遺伝子が出現した個体の適合度を足し合わせていき，最終的にその合計値が大きい順番に識別番号を割り振る．そのため，この手法は，出現回数に加えて，適合度の情報を用いている．

						適合度
X1	0	1	2	1	・・・	f[X1] = 10
X2	0	2	1	1	・・・	f[X2] = 12
X3	2	1	2	1	・・・	f[X3] = 9
X4	2	1	0	2	・・・	f[X4] = 11
X5	2	2	1	1	・・・	f[X5] = 7

（□ の中の数字は遺伝子番号，f[X] は適合度を示す）

Fig.1: Genes and fitness values in one generation when the number of individuals is 5

Fig.1 を例にして考える．1 次元目の 0 の遺伝子番号には，$f(X_1)+f(X_2)=22$，1 の遺伝子番号には 0，2 の遺伝子番号には $f(X_3)+f(X_4)+f(X_5)=27$ の値を与え，それぞれの値が大きいものから順番に 0，1，2 と割り振っていく．Fig.1 の場合では 0 は 1 に替え，1 は 2 に替え，2 は 0 に替える．この操作を同様にすべての次元に対しても行い，遺伝子番号の割り振りを変えていく．

4.2 提案手法 2：Re-labeling Differential Evolution average (RLDE average)

上の提案手法と同様に，良い適合度の個体には，優れた遺伝子が含まれていることを前提として考える．対立遺伝子の出現した個体が持つ適合度の合計値を出現回数で割ることによって求められる平均値を用い，その平均値の大きい順に遺伝子番号を振り直す．そのため，この手法では出現回数の影響を受けない．先ほどの提案手法では適合度が低くても出現回数が多い遺伝子は，適合度の合計値が大きくなり，優れた遺伝子だとみなされる．しかし，この手法では，適合度が低くて出現回数が多い遺伝子の平均値は低くなり，遺伝子の番号を正確に振り直すことが出来る．よって，適合度が小さくても出現回数が多い遺伝子と適合度の大きな遺伝子を明確に区別したうえで，番号を割り振ることができ，出現回数を考慮せずに，各世代での適合度が大きい遺伝子を次世代に残すことが期待できる．

先ほどの Fig.1 では，1 次元目の 0 の遺伝子には$\{f(X_1)+f(X_2)\}/2=11$，1 の遺伝子には 0，2 の遺伝子には$\{f(X_3)+f(X_4)+f(X_5)\}/3=9$ を与え，それぞれの値が大きいものから順番に 0，1，2 と割り振っていく．Fig.1 の場合では 0 はそのままで，1 は 2 に替え，2 は 1 に替える．この操作を同様にすべての次元に対しても行い，遺伝子の番号の割り振りを変えていく．

5 問題と条件

5.1 シミュレーション問題

変数に依存性がない問題の代表的なベンチマーク問題の例が Sphere 関数である．

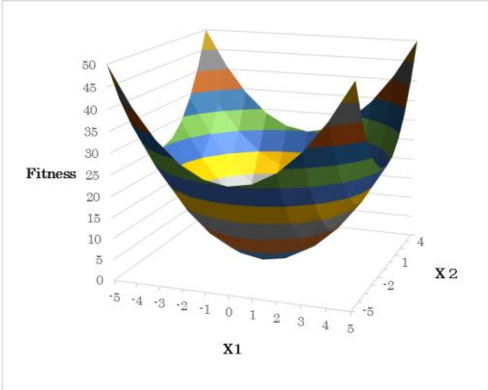


Fig.2: Sphere function

Sphere 関数の目的関数は，

$f(X_1, X_2, \cdots, X_n) = \sum X_i^2 \quad (-5 \leqq X_i \leqq 5)$ （式 2）

最小値は，

$f\min(0, \cdots, 0) = 0$ （式 3）

である．

この Sphere 関数は本来，実数値最適化問題である．そのため，組合せ最適化向けベンチマーク問題とするために次の変更を行った．Sphere 関数に対して，変数のラベルをランダムに入れ替え，それによってできるグラフ（Fig.3）に対して最適化を行った．Fig.4 は X_1 をある値に固定して X_2 を変化させたときの適合度の推移を示す．これから分かるように，X_1 の値が違っても，グラフの形状は変化しないことから，変数の依存関係を持たないことが分かる．

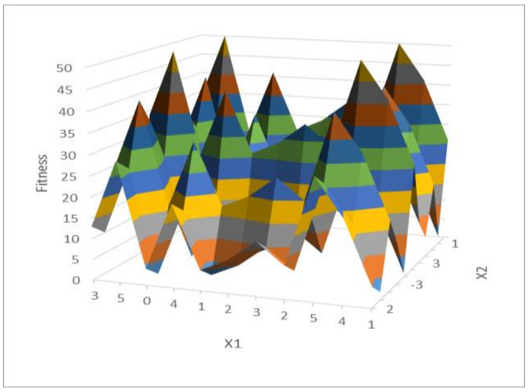


Fig.3: Graph after randomly exchanging label of Sphere function

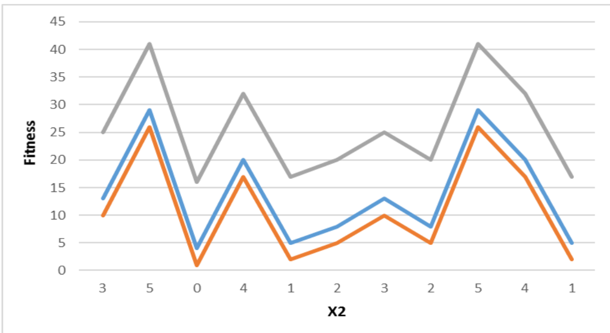


Fig.4: Transition of fitness when X_1 is fixed and X_2 is changed

また，変数の依存関係がある代表的なベンチマーク問題の例が Rosenbrock 関数である．Rosenbrock 関数の目的関数は，

$$f(X_1, X_2, \cdots, X_n) = \sum(100(X_{i+1} - X_i^{\,2})^2 + (X_i - 1)^2) \quad (-5 \leqq X_i \leqq 5) \qquad (式 4)$$

最小値は

$$f\ \mathrm{min}\ (1, \cdots, 1) = 0 \qquad (式 5)$$

である．

Rosenbrock 関数も実数値最適化問題であるので，Sphere 関数と同様の操作を行い，最適化を行った．Fig.8 は，X_1 をある値に固定して X_2 を変化させたときの適合度の推移を示す．X_1 の値が異なると，グラフの形状が変化していることから，変数の依存関係を持つことが分かる．

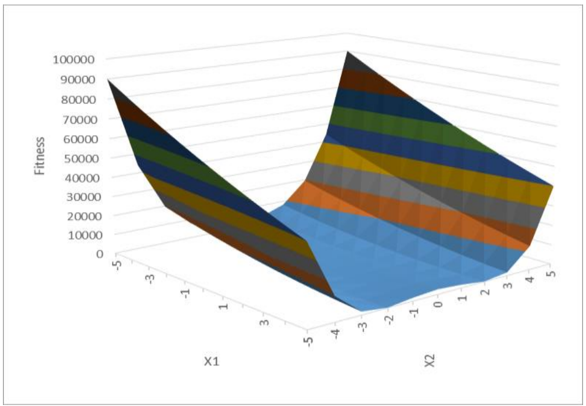


Fig.5:　Rosenbrock function

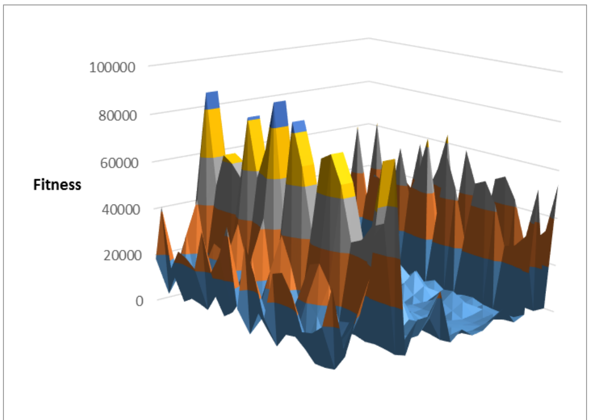


Fig.6: Graph after randomly exchanging label of Rosenbrock function

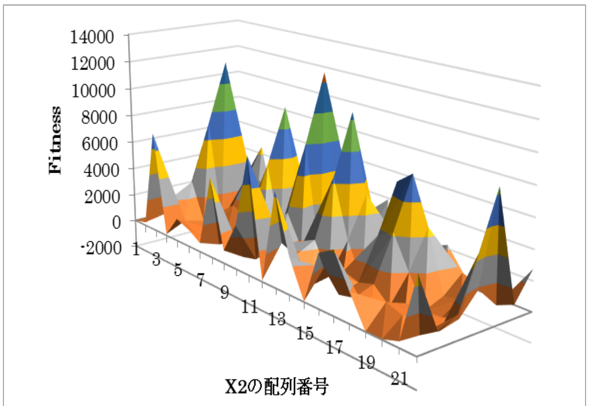


Fig.7: Graph near the optimal solution in Fig.6

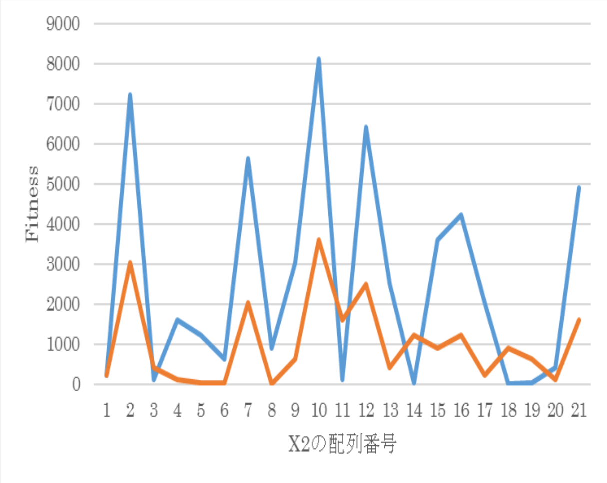


Fig.8: Transition of fitness when X_1 is fixed and X_2 is changed

5.2　シミュレーション条件

本研究では，DE，RLDE，RLDE fitness，RLDE average の性能比較を行う．2 つの問題に対して 2 つの条件でシミュレーションを行う．1 つ目は短時間で解を求められることを重視した条件で，個体数 10，世代数 20 までの探索を 100 回試行する．2 つ目は解を正確に求められることを重視した条件で，個体数 50，世代数 200 以上（Sphere 関数では 200 世代，Rosenbrock 関数では 500 世代）までの探索を 100 回試行する．

それぞれの Scaling Factor や交叉率は，個体数 10 では 15 世代，個体数 50 では 100 世代において最も優れた適合度となるように，Scaling Factor である F は 0.3, 0.5, 0.8, 1.0 から決定し，交叉率である CR は 0.3, 0.5, 0.8 から決定する．

6　シミュレーション結果・考察

6.1　シミュレーション結果

以下に，個体数 N が 10，50 のときの Sphere 関数から作られるテスト問題と Rosenbrock 関数から作られるテスト問題の結果を記す．（random，best は base vector の選択方法を示す．）

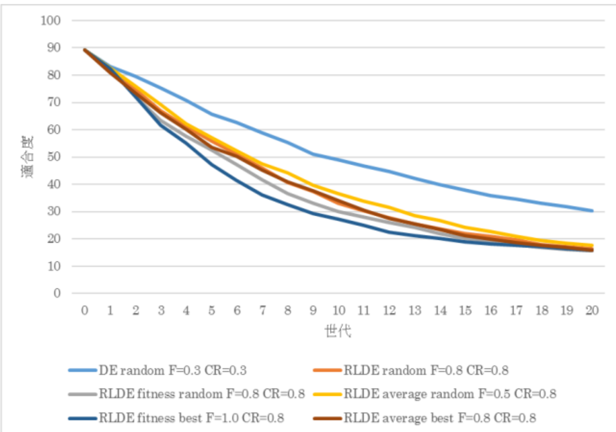


Fig.9:　Sphere function N=10
Average fitness of 100 trials

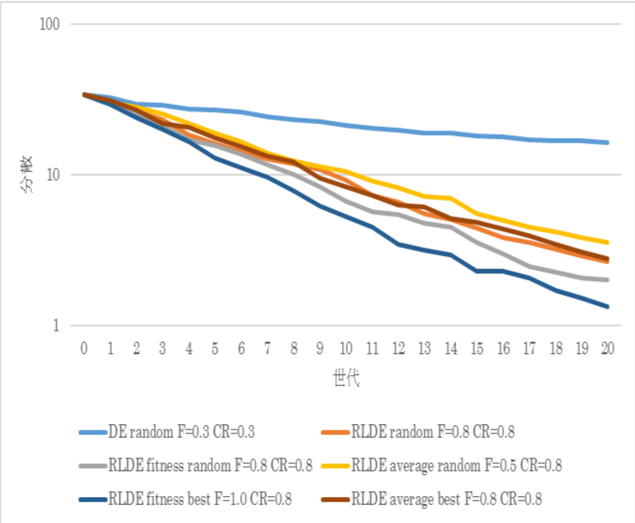


Fig.10: Sphere function N=10
Variance of parameters of the first dimension of 100 trials

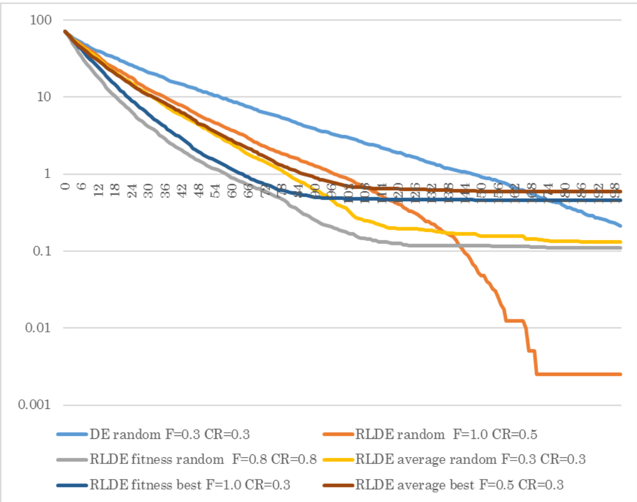


Fig.11: Sphere function N=50
Average fitness of 100 trials

Fig.12: Sphere function N=50
Variance of parameters of the first dimension of 100 trials

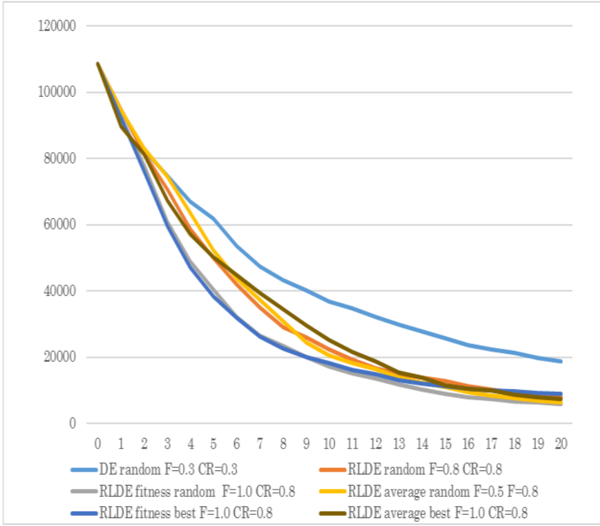


Fig.13: Rosenbrock function N=10
Average fitness of 100 trials

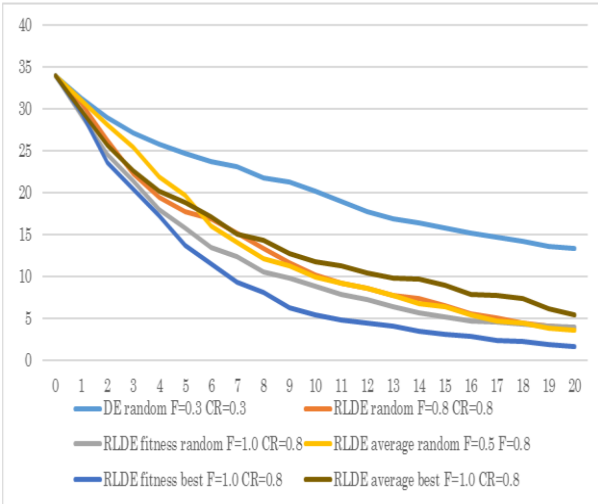


Fig.14: Rosenbrock function N=10
Variance of parameters of the first dimension of 100 trials

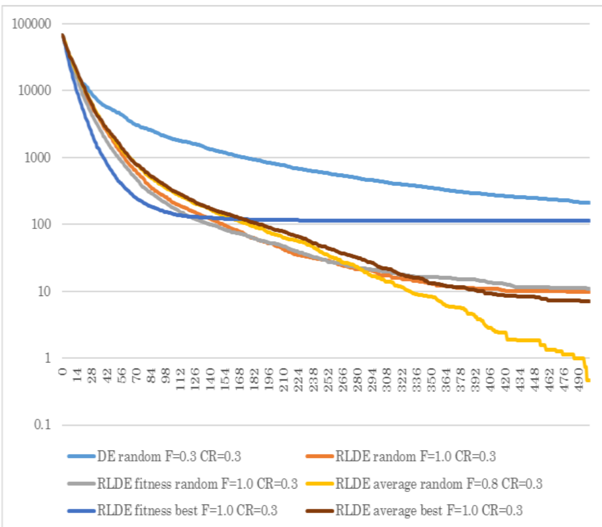


Fig.15: Sphere function N=50
Average fitness of 100 trials

Fig.16:　Rosenbrock function N=50
Variance of parameters of the first dimension of 100 trials

6.2　考察

Rosenbrock 関数のテスト問題では，best vector を用いるよりも，random vector を用いた方の探索性能が高かった．これは，変数の依存関係が大きく関わっていると考えられる．変数に依存関係がある問題は，1 つの変数の値が変化することによって，他の変数に影響を及ぼし，適合度が大きく変化する．その場合，best vector（局所的探索を重視した方法）を用いるよりも random vector（大域的探索を重視した方法）を用いる方が探索性能が向上したと考えられる．

RLDE fitness は，どのテスト問題においても探索初期での収束速度が速かった．100 試行の平均の 1 次元目のパラメータの分散がすべてのテスト問題で値が小さいことから，ある世代での最良解付近を中心に探索を行っていることがわかる．そのため，作成される差分ベクトルが小さくなっていくため，個体集団の収束が速い. しかし, Fig.15 から分かるように, RLDE fitness best は複雑に設定された問題の 100 世代以降では最適解を求められていなかった．これは，個体集団の収束により，個体の多様性が失われてしまったため，大域的探索が十分に行えなかったことが分かる．

RLDE average は世代数が多い場合（Fig.11 と Fig.15）に，最適解を求めることができていた．その理由は，新しく適合度が優れた個体が生成された時に，その遺伝子は出現回数が少ないにもかかわらず，出現回数は多いが劣った遺伝子よりも優れた遺伝子だと判断されるためである．そのため，出現回数が多い遺伝子に早期収束することなく，十分に大域的探索を行うことができる．

7　おわりに

先行研究では，出現回数が多い遺伝子は，優れた遺伝子が含まれていると想定している．本研究では，良い適合度の個体には，優れた遺伝子が含まれていることを前提として考える．対立遺伝子が出現した個体の適合度を足し合わせていき，最終的にその合計値が大きい順に遺伝子番号を振り直す手法と対立遺伝子の出現した個体が持つ適合度の平均値を求め，平均値の大きい順に遺伝子番号を振り直す手法を提案した．

シミュレーションでは, 従来手法と比較して, RLDE fitness（提案手法 1）は探索早期の収束速度が速かったが, 個体集団の多様性が失われていた. RLDE　average（提案手法 2）は，最適解を正確に求めることができていたが，個体集団の収束速度は遅かった．今後の展望として，提案手法それぞれの欠点を無くすため，遺伝子番号の新しい振り直し方，もしくは提案手法や先行手法の番号の振り直し部分を織り交ぜた手法を提案したい．

実際の対話型進化計算では，ユーザ固有の判断や好みが個体の評価に反映されるが，そのユーザの判断基準などは不明であり，個人によっても評価が異なる．そのため，実際に対話型進化計算の問題に適用し，問題を解決できるようなアルゴリズムの作成を行いたい.

参考文献

1)伊庭　斉志：C による探索プログラミングー基礎から遺伝的アルゴリズムまでー，株式会社オーム社,（平成 20 年 9 月）

2)棟朝　雅晴：遺伝的アルゴリズム　その理論と先端的手法，森北出版株式会社，（2008 年 7 月）

3) Price Kenneth, Storn Rainer M, Lampinen, Jouni A　: Differential evolution – A Practical Approach to Global Optimization，Springer，（2005）

4) Ryohei Funaki, Hirotaka Takano, Junichi Murata : Re-Labeling Differential Evolution for Combinatorial Optimization and Interactive Evolutionary Computation, SICE Journal of Control, Measurement, and System Integration, Vol. 9, No. 1, 18/25, (2016.01)

第11回コンピューテーショナル·インテリジェンス研究会
2017年6月26日-27日, 大津

ビジュアルリフティングアプローチによる二足歩行の安定性解析

○神 克礼 李 想 田 宏志 井澤 大時 見浪 護 松野 隆幸 (岡山大学)

Stability Analyses of Biped-walking by Visual-lifting Approach

*Keli Shen, Xiang Li, Hongzhi Tian, Daiji Izawa, Mamoru Minami and Takayuki Matsuno
(Okayama University)

Abstract– Biped walking control has been realized by Zero-Moment Point (ZMP). The efficiency of ZMP was well verified in keeping stable walking, but ZMP based walking cannot stop falling. Besides, dynamical walking can be used for walking that realizes kicks by toes, which does not depend on ZMP. Though the dynamical walking seems to be natural, robots tend to fall. Therefore, it is necessary for realization of human-like walking to keep dynamical walking stable. In our research, we have proposed a dynamical equation for walking derived by the Newton-Euler method including slipping, impact, line-touch and surface-touch of the foot. "Visual Lifting Approach" (VLA) is equipped to enhance the walking stability and stops the biped robot from falling without using ZMP. The VLA includes visual-lifting feedback and feedforward of walking gait generation. In this paper, we discuss how to realize the stable walking according to some measurements such as angle of ankle of floating foot, Center of Gravity (COG), waist angular velocity, height of head and waist and walking step length.

Key Words: Humanoid, Biped-walking, Visual-lifting Approach, Feedforward Inputs, Stability

1 Introduction

In many biped-walking control strategies of the humanoid, ZMP-based walking motion is considered as most efficient method, which has been certified to be useful in keeping stability of practical biped-walking, since it can make sure that humanoid robots can keep the balance of walking and standing by retaining the ZMP within the convex hull of supporting area[1, 2]. However, ZMP control makes the humanoid robots' waist lower and look like monkey while walking. Besides, other methods except ZMP are proposed to concentrate on keeping the biped-walking trajectories in side of a basin of attraction[3, 4, 5], including a way referring limit cycle to determining input torque[6].

These previous methods discussed are based on simplified biped models, which try to avoid discussing the effects of feet or slipping existing in real environment, Different from the above reference, one study[7] has pointed out the effect of foot having many walking gaits such as surface contacting (foot sole contacting with ground) and point contacting (heel contacting), changing the dimension of state variables. Our research has started from view point of[7] to describe the dynamics of gaits including point/surface-contacting state of foot, slipping of the foot and bumping as correctly as possible. It is called event-driven where walking gait transition would be determined by the past walking motion. The model in[7] only has foot model different from our model including the dynamics of whole-body humanoid with arms and head. And what the authors want to point out is that the dimension of equation of motion is changed by the varieties of the biped-walking introduced in[8] concerning one-legged hopping robot.

If the heel is detached from ground while its toe is contacting, a new state variable describing the rotation of foot will emerge, increasing the number of state variables. In fact, this kind of dynamics with dimension number of state variables changed by the result of its dynamical time profiles of motions are out of the area of control theory discussing how to control a system with fixed states' number. Further the tipping over motion has been called non-holonomic dynamics including a joint such as free joint without inputting torque.

At the same time, the heel or the toe of lifting foot in the air contacts with the ground geometrically. The referred paper[9] discussed the method of representing contacting with environment dealing constraint motion with friction by algebraic equation and applied it to human configuration[10]. According to these references, dynamics of 20 kinds of gaits were derived including slipping motion with both different constraint conditions and change of the dimension of state variables where the humanoid's dynamical model has been sufficient as much as possible[11].

In previous research on VLA in[12, 13, 14], the incomplete model of humanoid was applied in which head, arms and torso were neglected. Thus, there are some drawbacks, i.e., the model was too simple to consider the effect of dynamical coupling of arm and upper body. However, the new model proposed in this paper has been optimized concerning the above problem, and the discussion of slipping and effectiveness of the model have been proved in [11].

In this paper, Visual-lifting Approach(VLA) based on visual servo and visual feedback concept is examined to realize the human-like natural walking with slippage including toe-off state. Real-time position and orientation tracking method to observe a 3D object that is put near the humanoid to measure the robot's head relative pose has been proposed as visual pose estimation [15, 16]. The simulation result in-

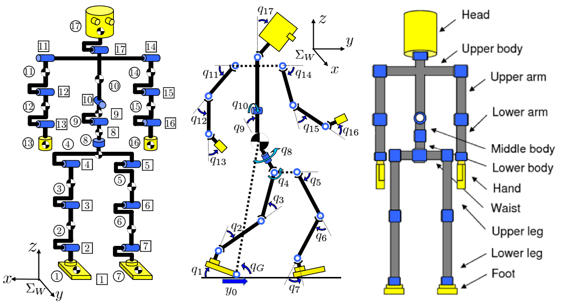


Fig. 1: Definition of biped-walking model, ①～⑰ represents link number, $\boxed{1}$～$\boxed{17}$ is joint number, q_1～q_{17} is joint angles.

Table 1: Physical parameters

Link	l_i[m]	m_i[kg]	d_i[Nms/rad]
Head	0.24	4.5	0.5
Upper body	0.41	21.5	10.0
Middle body	0.1	2.0	10.0
Lower body	0.1	2.0	10.0
Upper arm	0.31	2.3	0.03
Lower arm	0.24	1.4	1.0
Hand	0.18	0.4	2.0
Waist	0.27	2.0	10.0
Upper leg	0.38	7.3	10.0
Lower leg	0.40	3.4	10.0
Foot	0.07	1.3	10.0
Total weight [kg]	—	64.2	—
Total hight [m]	1.7	—	—

dicates that visual feedback control and feedforward inputs are useful to realize the stable biped-walking on the condition that humanoid's dynamics includes toe-off, slipping and bumping. Besides, this paper discuss how to realize the stable walking according to some measurements such as angle of ankle of floating foot, Center of Gravity (COG), waist angular velocity, height of head and waist and walking step length.

2 Biped-walking Model

The biped-walking robot in Fig.1 is discussed in this paper, Table 1 shows length l_i [m], mass m_i [kg] of links and coefficient of joints' viscous friction d_i [N·m·s/rad], which are determined by[17]. This model is simulated as a serial-link manipulator having branches and represents rigid whole-body such as feet including toe, torso, arms and so on and is up to 17 degree-of-freedom. Though motion of legs is limited in sagittal plane, it generates many walking gaits since the robot has flat-sole feet and kicking torque. In this paper, the foot named as link-1 is defined as "supporting-foot" and the other foot named as link-7 is defined as "free-foot" ("contacting-foot" when the free-foot contacts with ground) based on gaits. When the contacting-foot stopped slipping which indicated that static friction force is exerted to the foot, the contacting-foot is transferred into supporting-foot and the previous supporting-foot is changed to free-foot if it was isolated from floor.

3 Dynamical Calculations and Analyses

Equation of motion with one foot standing can be donated,

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

Here, $\boldsymbol{\tau} = [f_{y_0}, \tau_1, \tau_2, \cdots, \tau_{17}]$ is input torque, where f_{y_0} is always zero since the slipping motion has no actuators. $\boldsymbol{M}(\boldsymbol{q})$ is inertia matrix, $\boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ is the vector indicating Coriolis force and centrifugal one, and $\boldsymbol{g}(\boldsymbol{q})$ is gravity one. The μ_k in $\boldsymbol{D} = diag[\mu_k, d_1, d_2, \cdots, d_{17}]$ represents coefficient of friction, μ_k is the one between foot and ground. And $\boldsymbol{q} = [y_0, q_1, q_2, \cdots, q_{17}]^T$ includes the relative position between foot and ground y_0 generated by slipping and the angle of joints q_1～q_{17}.

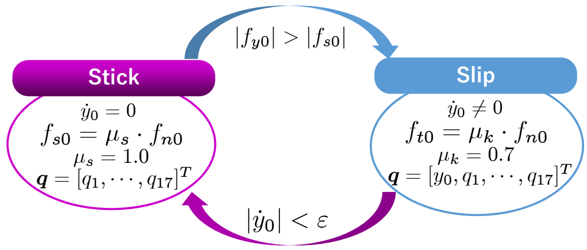


Fig. 2: Switch conditions of stick-slip motion

This stick motion state is described at left side of Fig.2. If $|\dot{y}_0| < \epsilon$ is satisfied, the degree of motion y_0 will disappear and the equation of motion will transfer to the equation of motion consisting of $\boldsymbol{q} = [q_1, q_2, \cdots, q_{17}]^T$. On this state, static friction coefficient $\mu_s = 1.0$ is employed, and static friction force $f_{s0} = \mu_s f_{n0}$ exerts to the lateral direction of foot.

However, when the supporting-foot (1-st link) is slipping (prismatic joint), the force exerting onto the 1-st link can be calculated by following equation.

$$f_{y_0} = \boldsymbol{e}_{z_0}^T \, {}^1\boldsymbol{f}_0 + \mu_k \dot{y}_0. \qquad (2)$$

where $\dot{y}_0$ is slipping velocity. The viscous friction force of y-axis (slipping axis) described as $\mu_k \dot{y}_0$ is shown in left-hand side of Eq.(2).

If the exerting lateral force f_{y0} generated by dynamical coupling of humanoid body calculated by Eq.(2) satisfies $|f_{y0}| > |f_{s0}|$, the slipping motion will start and the equation of motion, Eq.(2), will be changed into the one with variables of $\boldsymbol{q} = [y_0, q_1, q_2, \cdots, q_{17}]^T$ again, which is shown at the right state Fig.2.

4 Visual-lifting Approach

4.0.1 Feedback-lifting Torque Generator

This section proposes a visual-lifting feedback to improve biped standing/walking stability as shown in Fig.3. We apply a model-based matching method to

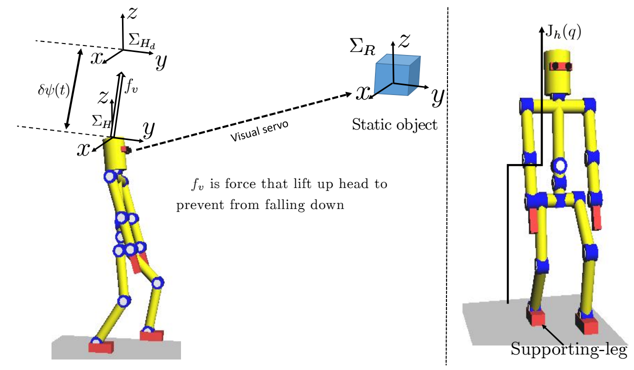


Fig. 3: Concept of Visual Lifting Stabilization.

evaluate posture of a static target object described by $\boldsymbol{\psi}(t)$ representing the robot's head based on Σ_H. The relatively desired posture of Σ_R (coordinate of reference target object) and Σ_H is predefined by Homogeneous Transformation as ${}^{H}\boldsymbol{T}_R$. The difference of the desired head posture Σ_{H_d} and the current posture Σ_H is defined as ${}^{H}\boldsymbol{T}_{H_d}$, it can be described by:

$$ {}^{H}\boldsymbol{T}_{H_d}(\boldsymbol{\psi}_d(t), \boldsymbol{\psi}(t)) = {}^{H}\boldsymbol{T}_R(\boldsymbol{\psi}(t)) \cdot {}^{H_d}\boldsymbol{T}_R{}^{-1}(\boldsymbol{\psi}_d(t)), \quad (3) $$

where ${}^{H}\boldsymbol{T}_R$ is calculated by $\boldsymbol{\psi}(t)$. $\boldsymbol{\psi}(t)$ can be measured by on-line visual posture evaluation proposed by [15, 16]. However, we assume that this parameter is set directly. Here, the force is considered to be directly proportional to $\delta\boldsymbol{\psi}(t)$, which is exerted on the head to minimize $\delta\boldsymbol{\psi}(t)(= \boldsymbol{\psi}_d(t) - \boldsymbol{\psi}(t))$ calculated from ${}^{H}\boldsymbol{T}_{H_d}$. The deviation of the robot's head posture is caused by gravity force and the influence of walking dynamics. The joint torque $\boldsymbol{\tau}_h(t)$ lifting the robot's head is donated:

$$ \boldsymbol{\tau}_h(t) = \boldsymbol{J}_h(\boldsymbol{q})^T \boldsymbol{K}_p \delta\boldsymbol{\psi}(t), \quad (4) $$

where $\boldsymbol{J}_h(\boldsymbol{q})$ in Fig.3 is Jacobian matrix of the head posture against joint angles including $q_1, q_2, q_3, q_4, q_8, q_9, q_{10}, q_{11}, q_{17}$, and $\boldsymbol{K}_p$ is proportional gain like impedance control. We apply this input to stop falling down caused by gravity or dangerous slipping gaits happened unpredictably during walking progress. We stress that the input torque for non-holonomic joint such as joint-1, τ_{h_1} in $\boldsymbol{\tau}_h(t)$ in (4) is zero for its free joint. $\delta\boldsymbol{\psi}(t)$ can show the deviation of the humanoid's position and orientation, however, only position is discussed in this study.

4.0.2 Foot and Body Motion Generator

Besides $\boldsymbol{\tau}_h(t)$, in order to make the floating-foot and supporting-foot step forward, added input torques $\boldsymbol{\tau}_t(t) = [0, \tau_{t2}, \tau_{t3}, 0, \tau_{t5}, \tau_{t6}, \tau_{t7}, 0, \cdots, 0]^T$ are used. And another kind of input torques $\boldsymbol{\tau}_w(t) = [0, \cdots, 0, \tau_{w8}, 0, \cdots, 0]^T$ is used to swing the roll angle of the waist (joint-8), which further realizes the arm swinging motion through dynamical coupling. Here, $\boldsymbol{\tau}_t(t)$ and $\boldsymbol{\tau}_w(t)$ are seen as feed-forward input torques. Here, t_2 means the time that supporting-foot and contacting-foot are switched. The elements $\boldsymbol{\tau}_t(t)$ and $\boldsymbol{\tau}_w(t)$ are shown below:

$$ \tau_{t5} = \begin{cases} 20cos(2\pi(t-t_2)/1.45), (t < 1.0[s]) \\ 15cos(2\pi(t-t_2)/1.85), (t \geq 1.0[s]), \end{cases} \quad (5) $$

$$ \tau_{w8} = \begin{cases} 50sin2\pi(t-t_2)/1.85), (\text{right foot is supporting}) \\ -50sin2\pi(t-t_2)/1.85), (\text{left foot is supporting}). \end{cases} \quad (6) $$

When time $t < 1.5[s]$, $\tau_{t2}, \tau_{t3}, \tau_{t6}, \tau_{t7}$ are set as feedback inputs.

$$ \tau_{t2} = 40(-0.2 - q_2), \quad (7) $$
$$ \tau_{t3} = 50(0.3 - q_3), \quad (8) $$
$$ \tau_{t6} = 100(-0.4 - q_6). \quad (9) $$

$$ \tau_{t7} = \begin{cases} 60(0.6 - q_7), (\text{the first step}) \\ 20(0.35 - q_7), (\text{others}). \end{cases} \quad (10) $$

When time $t > 1.5[s]$, $\tau_{t2}, \tau_{t3}, \tau_{t6}, \tau_{t7}$ are set as feed-forward inputs.

$$ \tau_{t2} = 10sin(2\pi(t-t_2)), \quad (11) $$
$$ \tau_{t3} = -10 + 10sin(2\pi(t-t_2)), \quad (12) $$
$$ \tau_{t6} = -20 + 20sin(\pi(t-t_2)). \quad (13) $$

$$ \tau_{t7} = \begin{cases} 60, (\text{floating and } q_7 \leq 0.6[rad]) \\ -40, (\text{point-contacting and } q_7 \geq 0.35[rad]) \\ 0, (\text{in other cases}). \end{cases} \quad (14) $$

4.1 Combined lifting/swinging controller

Combining three torque generators in Eqs.(4)∼(14), the controller for walking is derived,

$$ \boldsymbol{\tau}(t) = \boldsymbol{\tau}_h(t) + \boldsymbol{\tau}_t(t) + \boldsymbol{\tau}_w(t). \quad (15) $$

5 Simulation of biped-walking by VLA

In the environment that sampling time was set to $2.0 \times 10^{-4}[s]$ and coefficient of friction between the foot and the ground was set to $\mu_s = 1.0$ (static friction coefficient), $\mu_k = 0.7$ (viscous friction coefficient), the following simulation experiments were carried out. The desired position of head is set to $\boldsymbol{\psi_d} = [0, 0, 2.30[m]]$. Concerning simulation environment, we used "Borland C++ Builder Professional Ver. 5.0" to make simulation program and "OpenGL Ver. 1.5.0" to display humanoid's time-transient configurations.

In this section, some figures are obtained to analyze the stability of biped-walking in the simulation. In this simulation, we set lifting proportional gain $\boldsymbol{K}_p = diag[20, 290, 1010]$.

In Fig.4, X-axis represents the walking time, Y-axis represents the step length of walking. Biped walking includes three phases: initial phase, transient phase

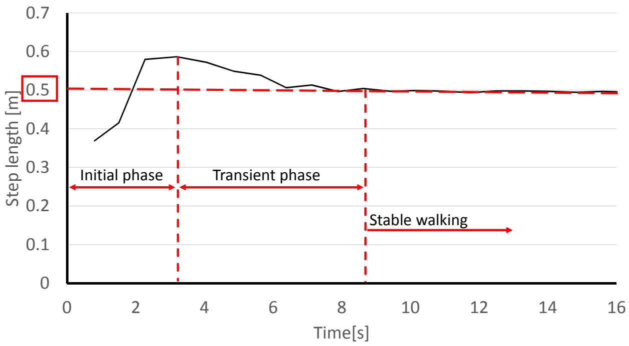


Fig. 4: Step length of Biped walking during 21 steps.

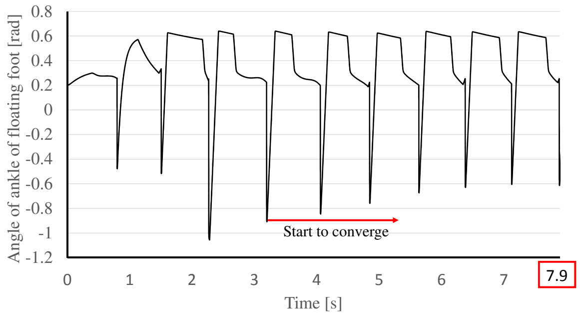


Fig. 5: Angle q_7 of ankle joint of floating foot (from the 1st step to the $11th$ step).

and stable walking phase.From Fig.4, the step length comes to convergence after $5th$ step. And biped robot walks as the same step length $0.5[m]$ after finishing $11th$ step.

Figures 5 and 6 show the change of the angle of ankle of the floating foot. Fig.5 shows that the angle of floating foot change irregularly before the $11th$ step. X-axis represents the time, Y-axis represents the angle of ankle joint of the floating foot. From this figure, the change shape of angle is different. After the $5th$ step, the change shape of angle starts to converge. After $11th$ step (after $8.69[s]$), the angle of ankle of floating foot change regularly in the certain range and Gait Cycle (time of finishing one step walking) changes in the limited range (from $0.77[s]$ to $0.79[s]$), which indicates that the gaits of floating foot change periodically. From the figure, the shape change of angle is similar from one step to another. The walking motion becomes stable after $11th$ step.

Figure 7 show the Center Of Gravity(COG) position during 100 step simulation. The upper part of Fig.7 shows the screen shot of the biped walking simulation. The point A means the initial posture. B and B' show the state before and after the switching of supporting foot in the $1st$ step. The points of C and C' show the second time of supporting foot switching. The lower two columns show the transition of position of COG from initial phase and transient phase to stable phase, which are depicted by coordinate Σ_{toe} that is fixed at the toe of the supporting foot. Fig.7 (b), (c)

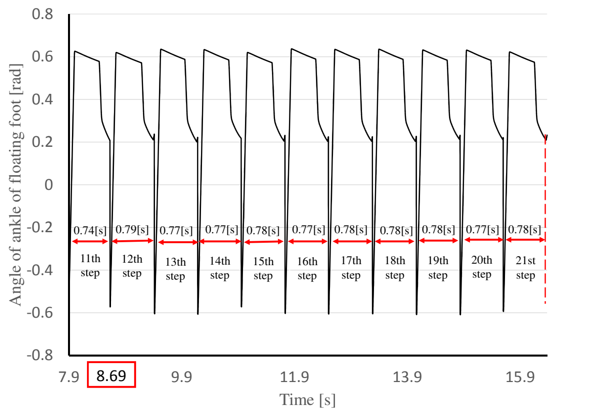


Fig. 6: Angle q_7 of ankle joint of floating foot (after $11th$ step).

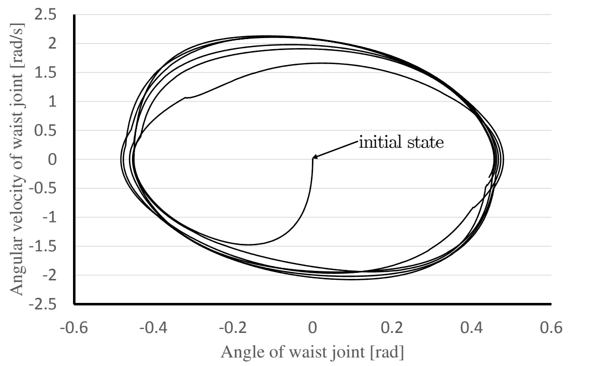
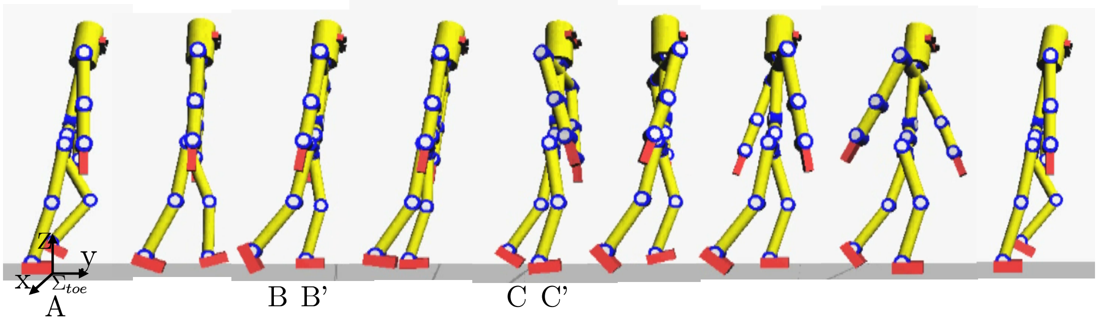


Fig. 8: Relation of angle q_8 and angular velocity $\dot{q}_8$ of waist joint in initial stage and convergence stage (from the 1st step to the $11th$ step).

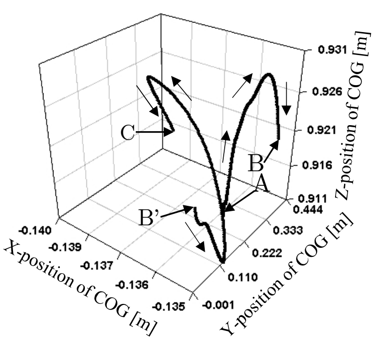
and (d) shows the initial phase and transient phase, the trajectory is complexed and no obvious similarity. In these figures, the position profile with A, B, B', C, C' corresponding to them in screen shots in Fig.7 (a). After entering stable walking shown in Fig.7 (e), (f) and (g), the trajectory of COG is converge to specific tendency, which is similar and along a narrow trajectory (the width of trajectory is less than $0.002[m]$).

Figure 8 and 9 represent the relation of angle q_8 and angular velocity $\dot{q}_8$ of waist joint during 100 steps. It is related to the stability of walking. Fig.8 shows the initial phrase and transient phase (from $1st$ step to $11th$ step). In this phase, the movement of the waist includes varieties and does not converge to one trajectory. When entering the stable state shown in Fig.9, the movement of the waist enters a limit cycle with a very small width.

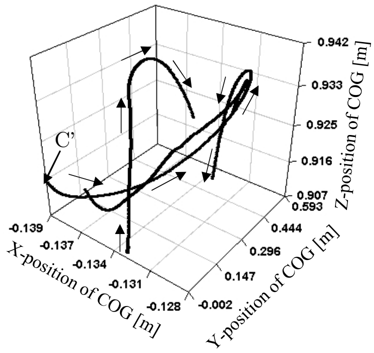
Figure 10 and 11 show the Z-axis position of head and waist based on the world coordinate system Σ_w during 100 steps walking. Fig.10 shows that the movement of both of head and waist has steady oscillations, which can be seen that the trajectory of motion is stable. Fig.11 is the expansion of Fig.10 in time. the height of head and waist before entering the stable state is described more obviously than Fig.10. Be-



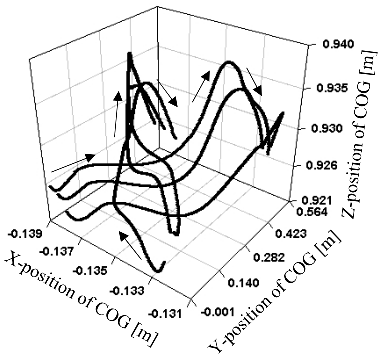
(a) Screen-shot of the biped-walking



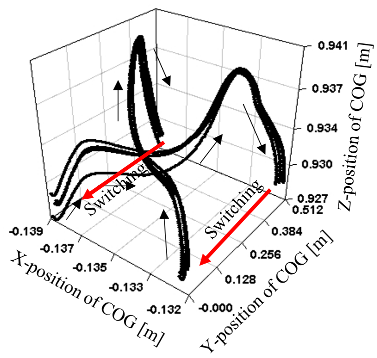
(b) COG position of 1st～2nd step



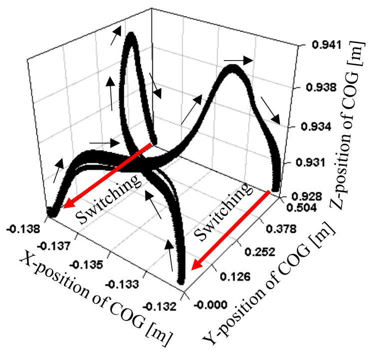
(c) COG position of 3rd～5th step



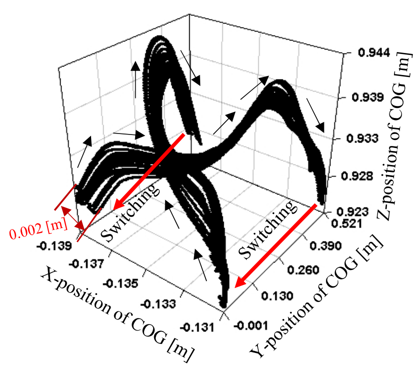
(d) COG position of 6th～11th step



(e) COG position of 12th～21st step



(f) COG position of 22nd～50th step



(g) COG position of 51st～100th step

Fig. 7: COG position during 100 steps walking simulation. The point of A means the initial posture, B and B' represent the state before and after the switching of supporting-foot in the first step. The C and C' show the second time of supporting foot switching. There are three states in the walking simulation. From $1st$ step to $5th$ step is the initial state, and from $6th$ step to $11th$ step is the transient state, after $11th$ step is the stable state.

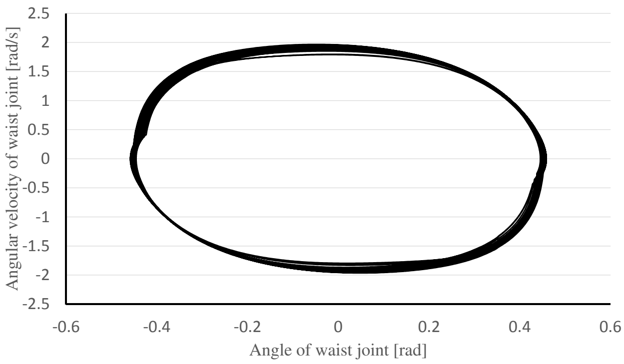


Fig. 9: Relation of angle q_8 and angular velocity $\dot{q}_8$ of waist joint in stable stage (after the $11th$ step).

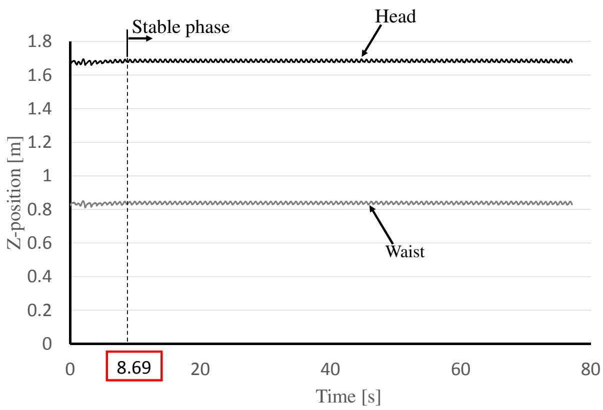


Fig. 10: Z-axis position (height) of head and waist joint based on the world coordinate system Σ_w during 100 steps walking. After the $11th$ step, the walking state is stable.

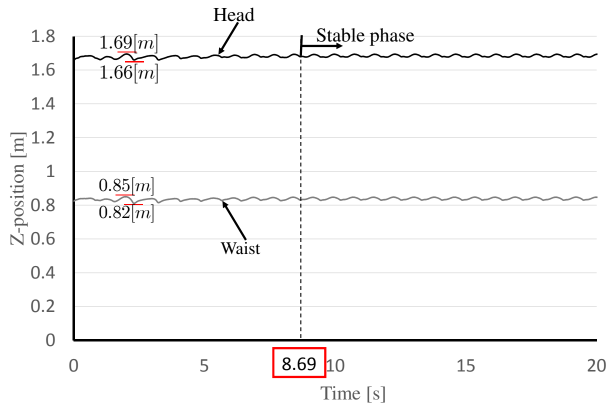


Fig. 11: Z-axis position (height) of head, waist joint before the stable state in Fig.10. After the $11th$ step, the vibrational motion of head and waist becomes steady

fore the $11th$ step, the height difference of waist and head is $0.03[m]$. After $11th$ step, difference becomes smaller and changes regularly. Therefore, the vibrational motion of head and waist become stable.

6 Conclusion

In this paper, the stability of walking is proved by some measurements such as step length, angle of ankle of floating foot and COG, waist angular velocity, height of head and waist. The results show Visual Feedback Control and Feedforward inputs based on the dynamical model that contains flat feet feet including toe, slipping and impact are effective to realize the stable walking, which is human-like natural walking. In the future work, we will adjust visual lifting gains to shorten the transient time and observe the versatility of feedforward inputs.

Reference

1) M. Vukobratovic, A. Frank and D. Juricic: On the Stability of Biped Locomotion, IEEE Transactions on Biomedical Engineering, Vol.17, No.1 (1970)
2) M. Vukobratovic and J. Stepanenko: On the Stability of Anthropomorphic Systems, Mathematical Biosciences, Vol.15, pp.1/37 (1972)
3) S. Colins, A. Ruina, R. Tedrake and M. Wisse: Efficient Bipedal Robots Based on Passive-Dynamic Walkers, Science, Vol.307, pp.1082/1085 (2005)
4) J. Pratt, P. Dilworth and G. Pratt: Virtual Model Control of a Bipedal Walking Robot, Proceedings of IEEE International Conference on Robotics and Automation, pp.193/198 (1997)
5) R.E. Westervelt, W.J. Grizzle and E.D. Koditschek: Hybrid Zero Dynamics of Planar Biped Walkers, IEEE Transactions on Automatic Control, Vol.48, No.1, pp.42/56 (2003)
6) Y. Harada, J. Takahashi, D. Nenchev and D. Sato: Limit Cycle Based Walk of a Powered 7DOF 3D Biped with Flat Feet, Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp.3623/3628 (2010)
7) Y. Huang, B. Chen, Q. Wang, K. Wei and L. Wang: Energetic efficiency and stability of dynamic bipedal walking gaits with different step lengths, Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp.4077/4082 (2010)
8) T. Wu, T. Yeh and B. Hsu: Trajectory Planning of a One-Legged Robot Performing Stable Hop, Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp.4922/4927 (2010)
9) Y. Nakamura and K. Yamane: Dynamics of Kinematic Chains with Discontinuous Changes of Constraints—Application to Human Figures that Move in Contact with the Environments—, Journal of RSJ, Vol.18, No.3, pp.435/443 (2000)
10) K.Yamane and Y.Nakamura: Dynamics Filter - Concept and Implementation of On-Line Motion Generator for Human Figures, IEEE Transactions on Robotics and Automation, vol.19, No.3, pp.421/432 (2003)
11) Xiang Li, Hiroki Imanishi, mamoru Minami, Takayuki Matsuno, Akira Yanou: Dynamical Model of Walking Transition Considering Nonlinear Friction with Floor, Journal of Advanced Computational Intelligence and Intelligent Informatics, Vol.20 No.6 (2016)
12) Wei Song, Mamoru Minami and Yanan Zhang: A Visual Lifting Approach for Dynamic Bipedal Walking, International Journal of Advanced Robotic Systems, Vol.9, pp.1-8 (2012)
13) Akira Yanou, Mamoru Minami, Tomohide Maeba and Yosuke Kobayashi: A First Step of Humanoid's Walking by Two Degree-of-freedom Generalized Predictive Control Combinedwith Visual Lifting Stabilization, Proceedings of the 39th Annual Conference of the IEEE Industrial Electronics Society (IECON2013), pp.6357/6362 (2013)

14) Wei Song, Mamoru Minami, Tomohide Maeba, Yanan Zhang and Akira Yanou: Visual Lifting Stabilization of Dynamic Bipedal Walking, Proceedings of 2011 IEEE-RAS International Conference on Humanoid Robots, pp.345/351 (2011)

15) W. Song, M. Minami, F. Yu, Y. Zhang and A. Yanou: 3-D Hand & Eye-Vergence Approaching Visual Servoing with Lyapunov-Stable Pose Tracking, Proceedings of IEEE International Conference on Robotics and Automation, pp.5210/5217 (2011)

16) F. Yu, W. Song and M. Minami: Visual Servoing with Quick Eye-Vergence to Enhance Trackability and Stability, Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp.6228/6233 (2010)

17) M. Kouchi, M. Mochimaru, H. Iwasawa and S. Mitani: Anthropometric database for Japanese Population 1997-98, Japanese Industrial Standards Center (AIST, MITI) (2000)

18) T. Maeba, M. Minami, A. Yanou and J. Nishiguchi: Dynamical Analyses of Humanoid's Walking by Visual Lifting Stabilization Based on Event-driven State Transition, 2012 IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics Proc, pp.7/14 (2012)

第11回コンピューテーショナル·インテリジェンス研究会
2017年6月26日-27日,大津

進化的実験計画法における実験数と感度分析精度の関係

○内種岳詞 （神戸大学） 周晨婷 畠中利治 （大阪大学）

The Relationship between The Number of Simulation Executions and Accuracy of Sensitivity Analysis in Evolutionary Design of Experiments

∗T. Uchitane (Kobe University), C. Zhou and T. Hatanaka (Osaka University)

Abstract– “Evolutionary Design of Experiments” (in short EDoE) is a framework to make designs of experiments dynamically. EDoE can be applied large scale social simulation and its analysis like Tsunami evacuation simulation. Here, the relationship between the number of simulation executions and accuracy of sensitivity analysis is discussed. From numerical simulation results, less mutation rate of EDoE may make the number of simulation executions less without making accuracy of sensitivity analysis worse.

Key Words: Design of experiments, Stochastic search algorithm, Sensitivity analysis

1 はじめに

大規模な自然災害への対策として，未知の災害が発生した場合の都市の機能を評価することが求められている．都市の機能評価には，過去のデータを集めて将来を予測する方法や都市システムをモデル化しシミュレーションから将来を予測する方法がよく用いられる．ここでは，シミュレーションを利用して都市機能を評価する手法におけるパラメータ数と実験数についての問題[1]，および実験数を削減する進化的実験計画法[2, 3, 4, 5]について説明する．そして，2 変数の交互作用を持つベンチマーク問題における進化的実験計画法の性能を評価する．評価手法として，全実験数の組み合わせに対する実施した実験数の比率の大小，および実験計画から得られた実験より推定されたモデルパラメータの誤差の大小を比較する．そして，2 変数の交互作用を持つベンチマーク問題における進化的実験計画法の有用性を示す．

2 進化的実験計画法

2.1 問題背景 [1]

集中豪雨や地震などの自然災害への対策は，さまざまな事態を想定しなければならないが，その対策を事前に評価することは困難である．このような防災の効果を評価することは困難であるが，膨大な現実データ蓄積と高度な計算機シミュレーションを利用した評価の推定への期待が高まっている．

このような背景のもと，金沢における地震発生後の津波避難シミュレーション[1]結果の一部が報告された．金沢市大野町 (Fig. 1) の津波避難において避難時間を推定するマルチエージェント歩行シミュレーションが実施された．この地域は，日本海沖地震による津波のリスクがあり，有事には迅速な避難が求められる．しかし，避難経路の橋が通れないことや，道が雪で覆われて移動が困難となることが予想される．Fig. 2 は金沢市大野町で実施された逃げ地図ワークショップ[6]で検討された通行不可能になる可能性のある橋 11 箇所と雪で移動が困難となると予想される道 11 通りが示されている．避難に要する時間は，利用可能な避難経路による影響を受ける．影響の大きさを見積もることができれば，避難時間に大きな影響を与える橋を補強したり，除雪対策を行い，迅速に避難を完了できる可能性がある．しかし，橋 11 本の通過の可否と雪道の除雪の有無の組み合わせ数は 2^{22} (約 419 万) 通りとなり，シミュレーション実験でさえ網羅的に実行することは難しい．ちなみに，市販の計算機 (Xeon 3.1GHz) のシングルプロセスでシミュレーション実験を行うと，1 組あたり約 4 分で，全組み合わせでは約 11650 日必要であった．なお，避難者が最短経路を予め知っているなど，単純化した人の移動モデルを利用したことで実行時間がすでに大幅に削減された上での見積り時間であることに留意されたい．

このように，これまでに推定できなかった規模の災害対策における評価が求められている．爆発するシナリオの組み合わせに対し，超並列計算機の利用によって，網羅的な計算を可能にする方法がある．一方で，網羅的なシナリオの組み合わせを網羅的に評価せず，重要なシナリオを効率よく発見する方法を開発することも重要な課題である．そのため，本研究では重要なシナリオをを探索する進化的実験計画法の開発を進めている．

2.2 2 変数交互作用を考慮したベンチマーク問題 [2, 3]

進化的実験計画法は，感度解析を適用できる適切なシナリオを効率よく選択する手法である．従来の実験計画法は，実験開始前に決定してから実験を実施するのに対し，本研究の手法では，実験を進め得られた結果により動的に計画を変更する．ここで *schema* は，$*0000000*0000000000000$ のように 0 と $*$ とで表現されるパターンとして定義される．要素は，0 か $*$ のどちらかで，要素数はイベントの数に等しい，たとえば，津

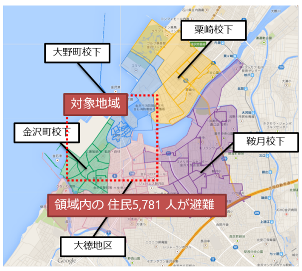


Fig. 1: 産業技術総合研究所人間情報研究部門において開発された NIGECHIZU SIMULATOR で対象とされた金沢市大野町．

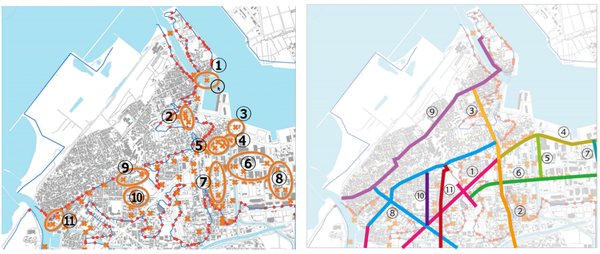


Fig. 2: 通行不可能となる可能性のある橋 11 本の配置と雪により移動が困難となる道 11 本の配置.

波避難シミュレーションでは，橋と道の合計数が 22 なので，要素数は 22 となる．ここで，$*$ はワイルドカードであり，0 と 1 の両方の状態を取る．すなわち，

$**000000**000000000000$

の *schema* からは，16 のシナリオ

1. 00 000000 00 000000000000
2. 10 000000 00 000000000000
3. 01 000000 00 000000000000
4. 11 000000 00 000000000000
5. 00 000000 10 000000000000
6. 10 000000 10 000000000000
7. 01 000000 10 000000000000
8. 11 000000 10 000000000000
9. 00 000000 01 000000000000
10. 10 000000 01 000000000000
11. 01 000000 01 000000000000
12. 11 000000 01 000000000000
13. 00 000000 11 000000000000
14. 10 000000 11 000000000000
15. 01 000000 11 000000000000
16. 11 000000 11 000000000000

が得られる．各シナリオの 0 または 1 は，たとえば，津波避難シミュレーションで橋や道の通行可否に対応している．そして，各シナリオは避難時間評価シミュレーションの入力となり，対応する 16 通りの避難時間がシミュレーション結果として得られる．よって，上記の *scema* が獲得できれば，第 1, 2, 9, 10 番目の主効果および，それらの主効果の交互作用の影響度を重回帰分析で評価できる．

最良の実験計画は感度解析の結果の確度が良く，かつ，実験数がより少ないものである．しかし，感度解析の確かさは，全シナリオに対する感度解析を実施してみるまで比較できない．よって，良い *schema* の集合を見つけるためには，なにかしらの方法で *schema* の良さを評価しなければならない．また，*schema* の $*$ の数が少ないことはシミュレーションを実施するシナリオ数が少なくてすみ，実験コストを削減できることを意味するが，$*$ の数が少なすぎると感度解析結果の確度は悪くなると考えられる．

進化的実験計画では，実験数の増加を抑え，感度解析結果の確度を良くするために，*schema* から得られるシナリオをシミュレーションで評価した結果の標本分散を評価する．なぜなら，*schema* から得られるシナリオでシミュレーションした結果が大きく変動すれば，感度解析の結果として大きな感度が得られるためである．よって，結果のばらつきを大きくする *schema* が実験計画に含まれるべきだと仮定し，結果の標本分散の大きさで *schema* を評価する．従って，発見すべき *schema* は，結果の標本分散を大きくし，かつ，含まれる $*$ の数が少ないものである．なお，ここでは *schema* に含まれる $*$ の数を *order* と呼ぶ．

簡易的なシステムモデルは式 (1) で与えられる．

$$
\begin{aligned}
y &= a_0 + \sum_{i=1}^{D} a_i x_i + \sum_{i=1}^{D} \sum_{j=i+1}^{D} b_{ij} x_i x_j \\
a_i &= a_{i+8} \\
b_{ij} &= b_{(i+8)(j+8)}
\end{aligned}
\tag{1}
$$

ここで，$a_0, a_1, \ldots, a_D, b_{ij}$ は定数のモデルパラーメタで x_i はモデルの入力変数 y はモデル出力，D はモデルの入力変数の次元である．x_i は 0 または 1 の値が入力される．D として，8 の倍数を与える．モデルパラメータの値は Table 1 に示す．式 (1) では，入力変数

Table 1: 簡易的なシステムのモデルパラメータ

a_0	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8
0	1500	1000	500	0	0	0	0	0

	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8
b_2	-1200	–	–	–	–	–	–	–
b_3	0	0	–	–	–	–	–	–
b_4	0	0	0	–	–	–	–	–
b_5	0	0	0	1000	–	–	–	–
b_6	0	0	0	0	0	–	–	–
b_7	0	0	0	0	0	0	–	–
b_8	0	0	0	0	0	0	0	–

の次元数 D を 8,16,24 と与えることにより，容易に次元数を増加させられる．また，2 種類の交互作用が明示的に与えられる．1 つは，主効果が存在する変数の組み合わせにおける交互作用で，$b_{12} = -1200$ が対応する．この交互作用は，x_1 と x_2 が同時に 1 になった時にのみ影響を及ぼす．そして，x_1 と x_2 には，それぞれ主効果として a_1 や a_2 が影響する．一方で，主効果が存在しない変数の組み合わせにおける交互作用で，$b_{45} = 1000$ が対応する．この交互作用は，x_4 と x_5 が同時に 1 になった時にのみ影響を及ぼす．そして，x_4 と x_5 には，それぞれ主効果の影響がない．

Fig. 3 に $D = 8$，Fig. 4 に $D = 16$ のときの *schema* に対応した標本分散の大きさを *order* 別にそれぞれ示す．主効果も交互作用もないダミー変数 x_6, x_7, x_8 が存在するため，それらの変数に $*$ がある *schema* の分散はそれらの変数に $*$ がない *schema* と同じ評価になる．シナリオ数を増やすことなく，より小さい *ordre* の *schema* を探すことが求められるので，たとえば，$D = 8$ において Fig. 3 における *order* が 4 で最大の分散を示す *schema*=“*0***000” を探すことが求められる．ちなみに，$D = 8$ のときに，感度分析を行うには，*schema*=“*****000” が発見されることが望ましいが，*schema*=“*****000” の分散は，係数の影響で *schema*=“*0***000” より小さくなる．

2.3 アルゴリズム

モデルに関する事前知識なしに，より良いデザインを獲得する最良の方法は存在しない．そこで，より良いデザインを探す一般的な手法が必要となる．ここでは，確率的な探索を繰り返すことにより，より良いデザインを獲得するアルゴリズムを提案する．提案したアルゴリズムは以下手順で実施される．

1. $*$ の数が 2 の初期パターンの生成 (集団サイズは ${}_DC_2$となる)
2. *schema* からシナリオを生成し結果の標本分散 $V(Y)$ を評価

Fig. 3: *order* と標本分散の関係 ($D = 8$).

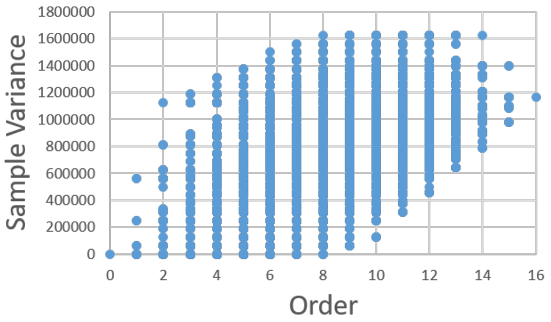


Fig. 4: *order* と標本分散の関係 ($D = 16$).

3. $V(Y)$ の値に基づき新しいユニークな *schema* を生成し，悪い *schema* を淘汰する

4. 終了条件を満たすまで，*schema* の評価・生成・淘汰を繰り返す

新しい *schema* は，2 つの親 *schema* の交叉と突然変異によって生成される．*schema* の分散の評価に基づき，親と成る *schema* をトーナメント選択 (トーナメントサイズ 2) する．生成された *schema* の要素の値は，それぞれ親の *schema* のどちらかの値から確率的に選択される．よって，生成された *schema* の *order* は，親の *order* より増減する．また，交差だけでは，どちらの親 *schema* にも $*$ の存在しない場所には新しく $*$ は生成されない．生成されたパターンの各遺伝子座には，突然変異がある確率 (MR) で起こり $*$ に変化する．

3　数値実験

問題の次元数 $D = 8, 16, 24$ において，突然変異確率 $MR = 500, 1000, 1500, 2000, 2500, 0$ を与えた．ここで，たとえば MR が 1000 であることは，生成される *schema* の要素 1000 個に 1 個が確率的に突然変異で $*$ へ変化することを意味する．よって，大きな MR ほど突然変異する数は少ないことを意味し，特に，$MR = 0$ は突然変異が起こらないことを意味する．なお，各実験設定において，乱数の初期値を変えた 5 試行を実施した．

3.1　進化的実験計画法で必要となった実験数と schema の関係 [5]

Figure. 5，Fig. 6，Fig. 7 に，$D = 8, 16, 24$ での利用されたシナリオ数の世代推移をそれぞれ示す．$D = 8$ の場合は MR の変化によるシナリオ数の増加速度の差は見えなかった．しかし，$D = 16, 24$ の場合，突然変異

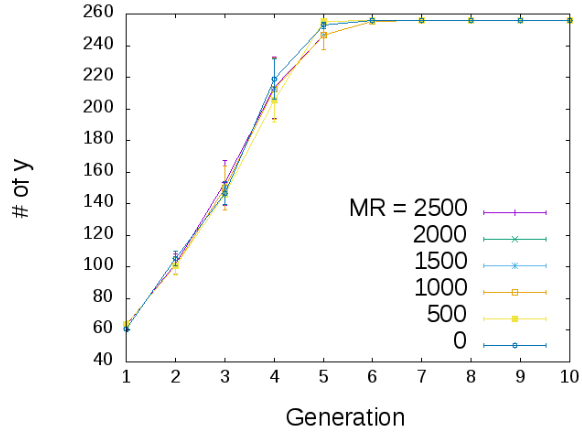


Fig. 5: 利用されたシナリオ数の推移 ($D = 8$).

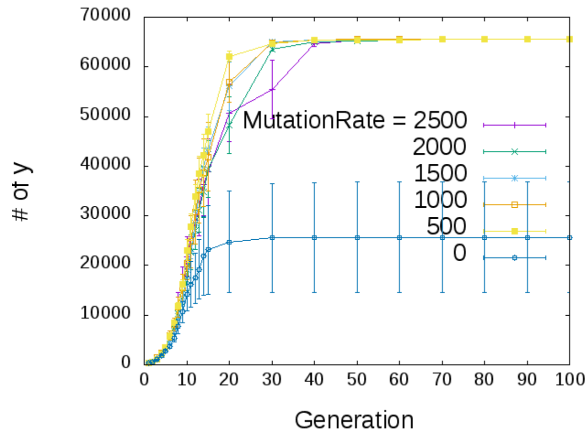


Fig. 6: [再実験データ] 利用されたシナリオ数の推移 ($D = 16$).

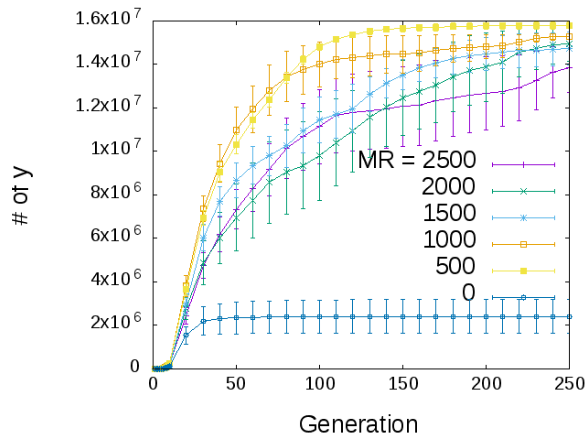


Fig. 7: 利用されたシナリオ数の推移 ($D = 24$).

が起こりやすいほど，シナリオ数の増加速度が速かった．さらに，突然変異が起こらない ($MR = 0$) 場合では，シナリオ数がより少ない数での増加しなくなった．なお，内種らによる報告 [5] における $D = 16$ の結果に誤りがあったため，ここでは Fig. 6 に再実験した結果を掲載している．

得られた *schema* の特徴として，初期世代における *schema* の進化の様子を見るため，世代別に *schema* の *order* ごとの分散の最大値を調べた．Fig. 8 に，$D = 8$ の $MR = 0, 500$ の場合についてそれぞれ示す．Fig. 8

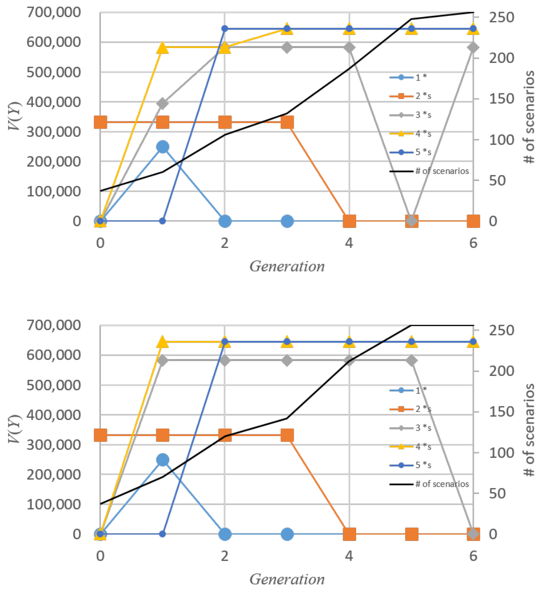


Fig. 8: $D = 8$ での初期から数世代の *order* 別の最大標本分散，$MR = 0$(上),$MR = 500$(下).

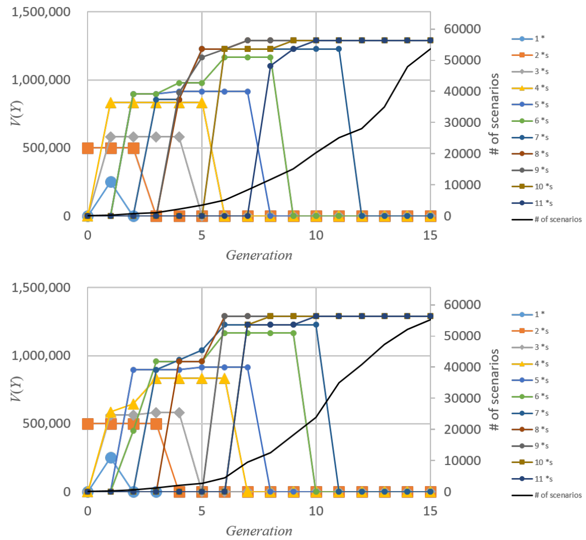


Fig. 9: [再実験データ]$D = 16$ での初期から数世代の *order* 別の最大標本分散，$MR = 0$(上),$MR = 500$(下).

より，$D = 8$ のとき，MR の差は見られなかった．Fig. 9 に，$D = 16$ の $MR = 0$(上),$MR = 500$(下) の場合についてそれぞれ示す．Fig. 9 より，$D = 16$ のとき，突然変異確率が高い $MR = 500$ 方が，より早い世代において *order* の少ない *schema* が淘汰されているのが確認された．さらに，突然変異確率が高い $MR = 500$ のとき，*order* の多い *schema* がより早い世代で誕生していることが確認された．Fig. 10 に，$D = 24$ の $MR = 0, 500$ の場合についてそれぞれ示す．Fig. 10 より，$D = 24$ のとき，$D = 16$ のときと同様の傾向が確認された．

3.2 重回帰分析によるモデルパラメータ推定

実験計画法の目的は，データを統計モデルに当てはめそのパラメータを推定することである．ここでは，進化的実験計画法より得られたモデルパラメータを示す．

ベンチマーク問題として利用した式 (1) は 2 変数の

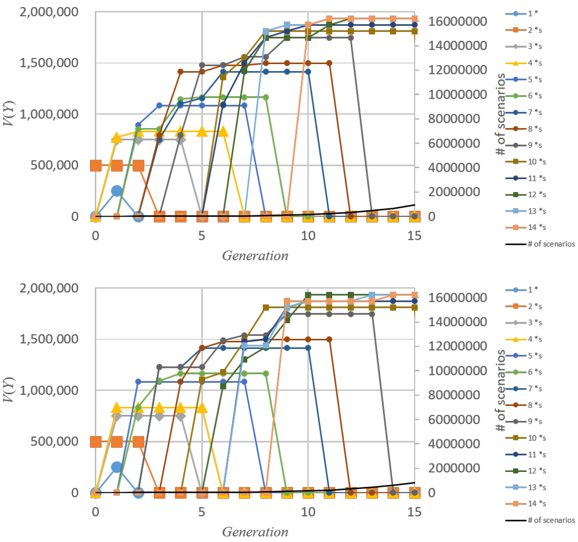


Fig. 10: $D = 24$ での初期から数世代の *order* 別の最大標本分散，$MR = 0$(上),$MR = 500$(下).

交互作用を考慮したモデルになっている．そして，求められるべき係数は，$k(i) = [i, \ldots, i + 8 * mod(D, 8)]$, $k(ij) = [ij, \ldots, (i + 8 * mod(D, 8))(j + 8 * mod(D, 8))]$ として $a_{k(0)}, a_{k(1)}, a_{k(2)}, b_{k(1)k(2)}, b_{k(3)k(4)}$ である．進化的実験計画法より得た実験全てを利用して，係数を推定する重回帰分析を実施した．推定された係数を，$\hat{a}_{k(1)}, \hat{a}_{k(2)}, \hat{a}_{k(3)}, \hat{b}_{k(1)k(2)}, \hat{b}_{k(4)k(5)}$ と表し，誤差 *residuals* は，

$$
\begin{aligned}
Residuals = & |a_{k(1)} - \hat{a}_{k(1)}| + |a_{k(2)} - \hat{a}_{k(2)}| + \\
& |a_{k(3)} - \hat{a}_{k(3)}| + |b_{k(12)} - \hat{b}_{k(12)}| + \\
& |b_{k(45)} - \hat{b}_{k(45)}|
\end{aligned} \tag{2}
$$

と定義する．

進化的実験計画法では，2 変数の交互作用を予め考慮した初期実験を与えるため，3 変数以上の交互作用を持たない式 (1) では，初期世代から係数推定が可能である．Fig. 11, Fig. 12, Fig. 13 に $D = 8, 16, 24$ の場合の *residuals* を，初期世代から数世代それぞれ示す．式 (1) から得られる y にはノイズが加えられていないため，推定誤差は世代や実験数の増加に関わらず，小さいままである．誤差が増加しているように見えるが，増加した誤差は僅かであり，多くの実数値を計算した計算誤差であると考えられる．

3.3 考察

事前に報告した実験 [5) より，突然変異確率が大きいほど *order* の少ない *schema* の探索を早い世代で打ち切り *order* の多い *schema* の探索へ移行した．また，得られた *schema* の集合に対して重回帰分析を実施し，誤差を評価した結果，初期世代から誤差小さく，また大きく変化していないことが分かった．このことから，進化的実験計画法の初期 *schema* 生成により，2 変数までの交互作用を持つモデルパラメータを実験数を増やすことなく推定できることが示された．一方で，進化的実験計画法の突然変異操作をほとんど行わないことが，式 (1) に対して，より実験数を抑える結果となった．シナリオ数の急激な増加を招く突然変異は不必要

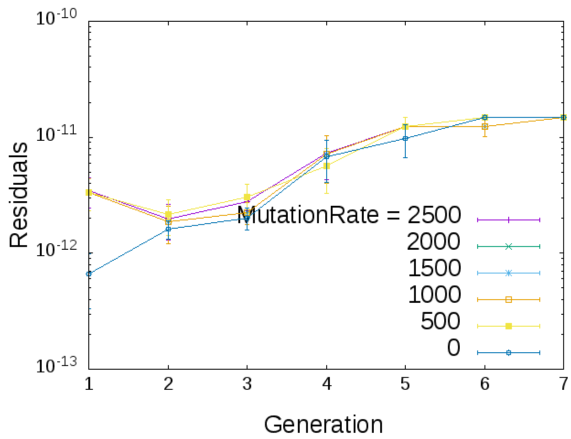


Fig. 11: 推定された係数と新の係数の誤差 ($D = 8$).

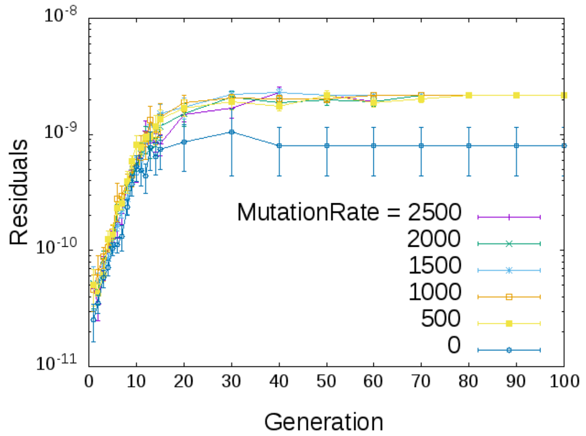


Fig. 12: 推定された係数と新の係数の誤差 ($D = 16$).

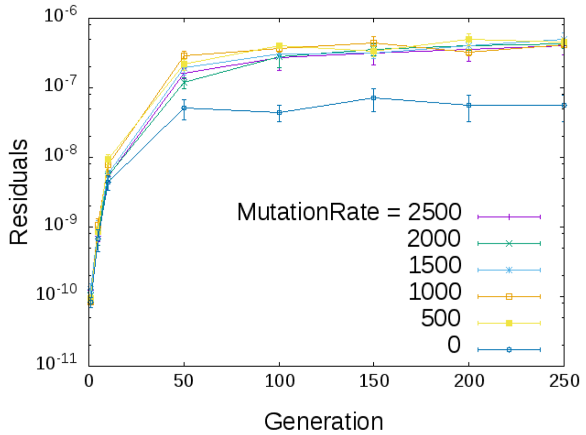


Fig. 13: 推定された係数と新の係数の誤差 ($D = 24$).

に見えるが，3 変数以上の交互作用が存在するようなモデルに対しても同様のことが言えるかは検証する必要がある．

式 (1) は 2 変数の交互作用を持つモデルであり，提案した実験計画法では，初期世代から係数を推定可能である．ただし，係数を推定するためには，モデルの出力にとって重要な変数を分類する作業が事前に必要となる．本稿では，すでに重要な変数や変数間の交互作用が正しく推定できた前提で誤差を評価している．今後は，進化的実験計画により得られた実験から重要な変数や交互作用を推定する枠組みを与える必要がある．

4　おわりに

本稿では，内種らが提案した進化的実験計画法において，実験計画より得られたデータからモデルパラメータを重回帰分析で推定し真の値との誤差を評価した．数値実験より，初期世代から良い精度のモデルパラメータが推定できていることが示され，また実験数が突然変異の操作をしない場合より多く削減できることが示されたことから，進化的実験計画法は有用であると言える．しかし，3 変数以上の交互作用を持つモデルで同様のことが言える保証はない．また，今回は突然変異についてのみ検討を行ったが，交叉や親個体の選択方法が実験数及ぼす影響についても検討を進める．

謝辞

本研究は JSPS 科研費 JPKT0105 の助成を受けたものである．

参考文献

1) 内種岳詞, 山下倫央, 辻順平, 松島裕康, 野田五十樹, 伊藤伸泰：避難シミュレーションへの進化計算適用結果の分析，計測自動制御学会 システム・情報部門 学術講演会 2015 講演論文集, SS3–18, 664/646, (2015)

2) 内種岳詞：進化的実験計画法による統計モデルの自由度とパラメータ値の推定，第 10 回進化計算学会研究会資料集, 173/177, (2015)

3) 内種岳詞：進化的手法による実験計画獲得法の考察，第 11 回進化計算学会研究会資料集, 173/177, (2016)

4) Takeshi Uchitane, Chenting Zhou, Toshiharu Hatanaka : Applying Evolutionary Design of Experiments to Sensitivity Analysis of Tsunami Evacuation Simulation, proceedings of NOLTA2016, 538/541, (2016)

5) 内種岳詞, 周晨, 畠中利治：進化的手法による実験計画獲得法の考察，第 12 回進化計算学会研究会資料集, 173/177, (2016)

6) 逃げ地図プロジェクト：http://www.nigechizuproject.com/

第11回コンピューテーショナル・インテリジェンス研究会
2017年6月26日-27日, 大津

CNNを用いた環境音スペクトログラムによる授業状況の推定に関する考察

○北橋未先　半田久志　（近畿大学）

Consideration on Estimation of Classroom Situations by Environmental Sound Spectrograms by using CNN

*M. Kitahashi and H. Handa (Kindai University)

Abstract– In this paper, the estimation method of the classroom situations from environmental sounds. We could formulate such problems as a sort of voice recognition problems. In this study, however, we convert the environmental sounds to imgaes, i.e., spectrograms. Such spectrograms are used to estimate the classroom situations by using Convolutional Neural Network (CNN). Moreover, we compare the accuracy of the CNN with the one of Support Vector Machine (SVM). The experiments indicate the effectiveness of the CNN.

Key Words: Convolutional Neural Network, Spectrogram, Momentum Stochastic Gradient Descent

1　はじめに

日本全国の教育現場へICT活用の波が広がるなかで，学生がPCやタブレット等の端末を操作する授業が増えている．また，文部科学省が教育の情報化加速化プランを策定するなど，今後も更にこの波は広がっていくことが予想される[1)]．それに伴い，端末操作に不慣れな先生であっても，授業をしながら学生の端末画面を見て授業態度を確認しなければならない場面が増え，先生の負担が増大しつつある．その問題を解決するため，先生の代わりに学生の授業態度を評価するシステムの開発を試みるに至った．

学生の授業態度を正しく評価するには，授業の状況に応じた評価基準を設けなければならない．しかし，先生が授業の状況を逐一入力する仕様では手間がかかる上，入力のし忘れやミスなどが起こりうる．これらの対策のために，収集が容易である環境音を用いて，システムに授業状況を推定させることを考える．

環境音は話者が大人数である場合も想定されるため，音声認識による文字列への変換や話者の判別が困難であることが予想できる．そこで，画像認識において事前学習を用いずに成果を出している畳み込みニューラルネットワーク (Convolutional Neural Network:CNN) に着目する．授業中の環境音をスペクトログラム画像に変換し，CNNを用いてそれらを分類することで授業状況を推定する．

本稿では，初めに本研究で用いるCNNと学習手法について述べる．そして，授業中の環境音に状況 (講義，演習，試験) をラベル付けし，スペクトログラム画像に変換したものをCNNで分類する実験を行う．並びに，その性能をサポートベクターマシン (Support Vector Machine:SVM) と比較し，考察する．最後に，結論と今後の課題について述べる．

2　CNN

CNNとは畳み込み層とプーリング層を交互に接続した構造を持つ多層ニューラルネットワークである．畳み込み層とプーリング層によって，領域単位で特徴が抽出でき，特徴の位置変化に対して柔軟になる．これにより画像認識の分野で注目されることとなった．ここでは，畳み込み層とプーリング層の構造について解説し，本研究で用いるCNNモデル全体の構造を示す．

2.1　畳み込み層

畳み込み層では，入力と複数のフィルタの畳み込み演算を行う．

1つのフィルタを畳み込む場合を考える．入力が縦横サイズ $S\times S$ 画素の D 枚の画像であるときの入力は $S\times S\times D$，その入力に対して畳み込むフィルタは $T\times T\times D$ の形をとる．そのときの入力値を $IN_{xyz}(x,y,z\in[0,S-1],[0,S-1],[1,D])$，フィルタの重みを $w_{xyz}(x,y,z\in[0,T-1],[0,T-1],[1,D])$ とする．入力画像の画素 (i,j) を頂点とするサイズ $T\times T\times D$ の領域にフィルタを畳み込み，バイアス b を足した結果

$$u_{ij}=\sum_{z=1}^{D}\left(\sum_{(p,q)\in P_{ij}} IN_{pqz}w_{p-i,q-j,z}\right)+b$$

となる．ただし，P_{ij} は画像中の画素 (i,j) を頂点とするサイズ $T\times T$ の正方領域

$$P_{ij}=\{(i+i',j+j')|i'=0,...,T-1,j'=0,...,T-1\}$$

である．ここで使われている w_{pqz} と b が最適化によって更新されることになる．縦や横方向に頂点 (i,j) をストライド s で移動させていき，別の領域にも同じフィルタを畳み込んでいく．

このように計算された u_{ij} は，その後活性化関数 a を経て，畳み込み層の出力

$$OUT_{ij}=a(u_{ij})$$

となる．D' 枚のフィルタを畳み込むと，出力OUTの形は $S'\times S'\times D'$ となる．

2.2　プーリング層

プーリング層では，ある領域内の入力を1つの値に集約する．これにより，画像内の特徴の微小な位置変化に対する応答の不変性を実現することができる．

プーリング層の入力の形を畳み込み層と同様に $S\times S\times N$ と表す．入力値を IN_{xyz} とし，その中の $T\times T$

サイズの正方領域 P_{ij} を対象にプーリングする．平均プーリングでは

$$OUT_{ijz} = \frac{1}{|P_{ij}|} \sum_{(p,q)\in P_{ij}} IN_{pqz}$$

最大プーリングでは

$$OUT_{ijz} = \max_{(p,q)\in P_{ij}} IN_{pqz}$$

が出力となる．畳み込みと同様に，縦や横方向に頂点 (i, j) を間隔 s で移動させ，別の領域もプーリングする．

2.3 モデル

モデル全体の構造を示す前に，畳み込み層とプーリング層以外での処理について説明する．

パディングは，畳み込みの前に入力の周囲を 0 で埋める処理である．これには，端の領域が他に比べて畳み込まれる回数が少なくなることを防ぐ目的がある．パディング幅を p，入力が縦横サイズ $S \times S$ とすると，パディング後のサイズは $(S + 2p) \times (S + 2p)$ となる．

Dropout は，順伝播時には対象となる層の出力をすべて α 倍にし，学習時には一定割合 $0 \le \alpha < 1$ のノードを消す処理である．学習時に消されたノードは存在しないかのように扱い，学習に影響を与えない．これには，各ノードに対して他のノードに頼らず活動させ，過学習を防ぐ目的がある．

では，本研究で用いる CNN 全体の構造を説明する．CNN 全体の構造を Fig. 1 に示す．Fig. 1 中の表記の意味は Table 1 に示す．CNN の入力画像のサイズを 256×256 画素とする．スペクトログラム画像はカラーであるため，1 画素の表現は RGB の 3 次元ベクトルとなる．したがって，入力層は $256 \times 256 \times 3$ の形をとる．また，講義，演習，試験の 3 クラスに分類するため，出力層には各クラスに対応する 3 個のノードを有する．これらは Softmax 関数により確率分布となっている．3 個のノードのうち，最も値が高いノードに対応するクラスが推定の結果となる．

CNN では通常出力層の直前に全結合層を含むが，このモデルは全結合層の代わりに特徴マップ全域での平均プーリングを採用している．この構造は Network In Network(NIN) と呼ばれ，2014 年に Min Lin らによって提唱された [2]．また，通常の畳み込み層の後にフィルタサイズ 1 の畳み込み層を組み込んでいるのも，NIN の特徴である．この NIN の構造が過学習を防ぐことがわかっている．

3 学習

CNN のパラメータ θ の更新方法について述べる．1 個のある入力に対する理想の出力 $d_z (z = 1, 2, 3)$ と実際の出力 p_z の交差エントロピーは

$$C = -\sum_{k=1}^{3} d_k \log p_k$$

となる．このとき d_z は，正解クラスのとき 1，正解クラスでないとき 0 とする．本研究ではミニバッチ法で学習するため，バッチサイズ n のミニバッチ N の交差エントロピーの平均 C_N をコストとする．そして誤差逆伝

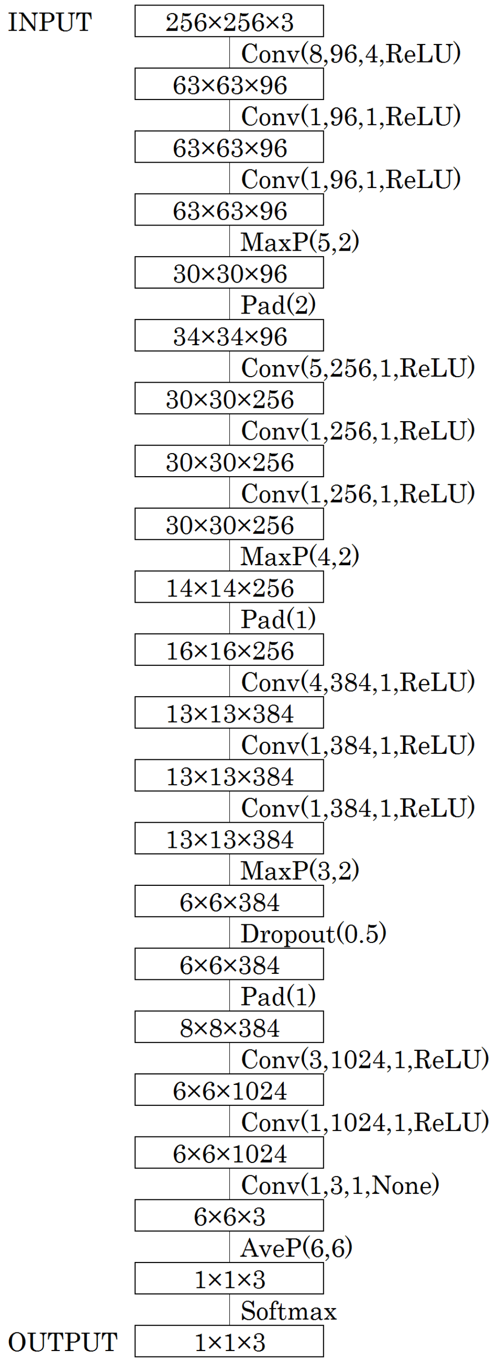


Fig. 1: 本研究で用いる CNN 全体の構造

Table 1: 表記の意味

表記	意味
Conv(L,D,s,a)	畳み込み層 (L：フィルタサイズ，D：フィルタ枚数，s：ストライド，a：活性化関数)
MaxP(L,s)	最大プーリング (L：領域サイズ，s：ストライド)
AveP(L,s)	平均プーリング (L：領域サイズ，s：ストライド)
Pad(p)	パディング (p：幅)
Dropout(α)	Dropout(α：割合)
Softmax	Softmax 関数

播によって勾配 $\partial C_N/\partial\theta$ を求めた後，MomentumSGD (Momentum Stochastic Gradient Descent) と呼ばれる最適化手法を用いて θ を更新する．

MomentumSGD は θ の更新に慣性を加え収束を早めた確率的勾配降下法である．この手法はハイパーパラメータとして学習率 η，慣性パラメータ α を持ち，速度 v を更新しつつ学習する．速度 v は初期値を 0 とする．速度を

$$v \leftarrow \alpha v - \eta \frac{\partial C_N}{\partial \theta}$$

のように更新．この αv が慣性を意味する．その後 θ を

$$\theta \leftarrow \theta + v$$

のように更新する．これらの作業をミニバッチ毎に行い，C_N を小さくする θ を学習する．以降，θ の 1 回の更新を 1 train，訓練データ 1 周分の学習を 1epoch と数える．

4 実験

これまでに述べた CNN モデルと学習方法の有用性を比較実験と適応実験により検証する．

4.1 比較実験

4.1.1 手法

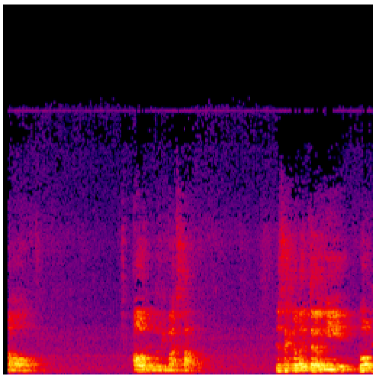


Fig. 2: スペクトログラム画像の例 (講義クラス)

比較実験では，CNN による講義状況の推定正解率をソフトマージン SVM と比較する．

訓練データ及びテストデータには，大学の講義中の環境音を用いる．講義中の環境音を 5 秒間隔で切り取り，256×256 のスペクトログラム画像を生成する．スペクトログラムは Fig. 2 のような画像になる．スペクトログラムの横軸は時刻 t (s) で $0 \leq t \leq 5$，縦軸は周波数 f (kHz) で $0 \leq f \leq 22$，色は音声信号の大きさの対数スケール db (dBFS) で $-100 \leq db \leq 0$ の範囲の値をとる．そして，画像の各画素を [0,1] の範囲で正規化する．それらをその時の状況から講義，演習，試験にラベル付けしたものを CNN への入力とする．

用意した学習データは各クラス 3881 個，合計 11643 個である．これらをランダムにソートした後に 10 等分する．この際，1 グループ内には各クラスのデータが同じ数あるようにした．その内の 9 グループを訓練データにして 20 epoch 分学習させる．それと同時に，残りの 1 グループをテストデータにして 1 train 毎にその時点での正解率を測る．この作業を全てのグループが 1 回ずつテストデータになるようにグループを交代して，計 10 回行う．最終的に 10 回分の正解率を 1 train 毎に平均する．つまり，10 分割交差検証によって実験する．

学習時のバッチサイズは 32 とし，MomentumSGD のハイパーパラメータは学習率 $\eta = 0.01 \cdot 0.97^{epoch-1}(epoch \in [1, 20])$，慣性パラメータ $\alpha = 0.9$ とする．

比較対象となるソフトマージン SVM も 10 分割交差検証する．SVM の検証は以下の 3 パターンの方法で行う．

- 線形カーネル
- 線形カーネル +BoF
- RBF カーネル +BoF

BoF(Bag-of-Features) とは，SIFT(Scale-Invariant Feature Transform) と k-means を用いて画像をヒストグラムで表現する手法のことで，本研究では画像を 100 次元で表現し SVM に入力する．SVM のパラメータは探索によって決定する．

4.1.2 結果

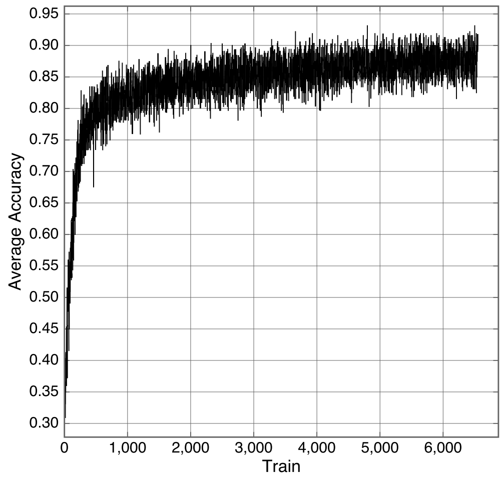


Fig. 3: 平均正解率の推移

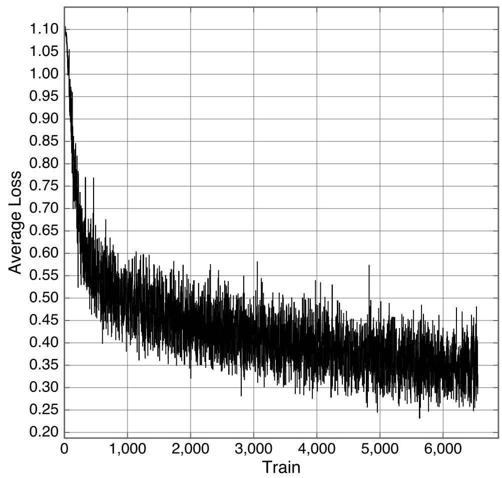


Fig. 4: 平均コストの推移

10 分割交差検証による平均正解率の推移を Fig. 3 に，平均コストの推移を Fig. 4 に示す．この実験では，最終的に平均正解率 89.06% で状況の推定に成功した．

Table 2: 比較実験の混同行列

		推定		
		講義	演習	試験
正解	講義	3385	153	343
	演習	144	3652	85
	試験	535	117	3229

Table 3: SVM の推定結果

方法	パラメータ	平均正解率
線形カーネル	$C = 2^{-1}$	82.41%
線形カーネル +BoF	$C = 2^{-5}$	76.17%
RBF カーネル +BoF	$C = 2^{0}$ $\gamma = 2^{-9}$	79.58%

比較実験の混同行列を Table 2 に示す．また，SVM の推定結果を Table 3 に示す．CNN の方が全パターンの SVM よりも正解率が高い結果となった．Fig. 3 から，CNN の平均正解率は 5000 train 以降，約 82.5% から 92.5% の間を振動し続け，停滞していることがわかる．最終正解率は 89.06% であるが，振動域の中央値はそれよりもやや劣る．しかし，それを考慮しても SVM の中で最も高い 82.41% を超えており，CNN の方が正解率が高かったと言える．

4.2 適応実験

4.2.1 手法

適応実験では，比較実験で学習した CNN モデルを用いて未知の講義の状況を推定し，比較実験と正解率を比較する．

比較実験と同様のデータ生成手順により，講義クラス 1445 個，演習クラス 136 個，試験クラス 435 個のデータが得られた．比較実験時に学習した CNN モデルは 10 分割交差検証により 10 個あるため，それぞれで推定を行う．

4.2.2 結果

Table 4: 適応実験の混同行列

		推定		
		講義	演習	試験
正解	講義	11393	634	2423
	演習	235	266	859
	試験	806	422	3122

適応実験の混同行列を Table 4 に示す．この実験での正解率は 73.32% となり，比較実験の結果である 89.06% に比べ 15.74 ポイント低い結果となった．

5 考察

はじめに，比較実験で比較した推定方法において，CNN の方が SVM より正解率が高かった原因を考察する．正解率に差が出た原因の一つに，入力の変化に対する不変性に差があったことが考えられる．入力のスペクトログラムの横軸は時刻であるため，推定に必要な特徴の位置が横軸方向に変化する．したがって，高い正解率で推定するためには，この入力の変化に対してモデルに不変性を持たせなければならない．CNN はプーリングにより，入力画像の特徴の位置変化や歪みに対して柔軟であるため，SVM よりも高い不変性を実現できた．SVM で用いた BoF でも特徴の位置変化に対応出来るが，BoF の場合は特徴の位置情報を完全に落としてしまい，スペクトログラム画像の縦軸で表される周波数までもが欠落し，正解率を下げたと考えられる．

次に，適応実験の結果が比較実験に比べ 15.74 ポイント低い結果となった原因を考察する．Table 4 から，未知の講義の推定では演習クラスの 63.16% を試験クラスと誤って推定したことが判る．この誤推定は Table 2 の比較実験の結果と比較してもかなり高い割合である．これは，適応実験の対象の講義では，学習データで用いた講義と演習の内容が異なっており，学生が静かで相談する声がほとんど無かったことが原因であると考えられる．

最後に，推定の正解率をより良くする手段を考察する．Fig. 4 から平均正解率が停滞した 5000 train 以降も平均コストは少しずつ減少していることが判る．この時点ですでに過学習している恐れがあるため，これ以上 epoch 数を増やしても未知の講義の状況推定正解率が大きく向上することは見込めない．それ以外の手段としては，ハイパーパラメータやモデル，最適化手法の変更，データ形式の変更が考えられる．データ形式の変更について述べる．Table 2 から判るように，演習クラスのデータは 94.10% という高い精度で推定に成功している．これは，演習中は学生同士が相談をしているため環境音に特徴が出やすいからであると考えられる．一方，試験クラスのデータの推定精度は 83.20% と低く，13.79% を誤って講義クラスと推定している．講義クラスは講師の声，試験クラスは静かであることが特徴であると想定できるが，試験中の講師からの指示や補足する声が講義の特徴と酷似している．それにより一部の試験クラスのデータが誤って推定されたのではないかと考えられ，これを解決することで正解率の向上が見込める．この対策として，データ生成時の環境音を区切る間隔を長くすることで，データの偏りを軽減するといった方法が考えられる．また，未知の講義の状況推定正解率を高めるためには，より様々な講義を学習データに用い，講義の内容や学生の変化に強いモデルを作る必要があると考えられる．

6 おわりに

本稿では，授業の状況 (講義，演習，試験) の推定方法として，環境音のスペクトログラム画像を用いた CNN による推定を提案し，性能実験を行った．実験の結果より，CNN は SVM よりも正解率が高いことが判った．

しかし，実験から試験クラスのデータの推定精度が比較的低いことも判った．試験クラスのデータの推定精度を更に上げることで，正解率の向上も見込める．今後の課題としては，ハイパーパラメータやモデル，最適化手法，データ形式を変更するなどの手段で，更に正解率を高められるかを検討することがあげられる．

参考文献

1) 文部科学省：教育の情報化加速化プラン，http://www.mext.go.jp/b_menu/houdou/28/07/__icsFiles/afieldfile/2016/07/29/1375100_02_1.pdf (2016)

2) Min Lin, Qiang Chen, Shuicheng Yan : Network In Network, https://arxiv.org/pdf/1312.4400v3.pdf (2014)

第11回コンピューテーショナル・インテリジェンス研究会
2017年6月26日-27日, 大津

機械学習を用いた高リスク学生の早期発見

〇近藤伸彦（首都大学東京）　大久保緑　畠中利治（大阪大学）

Early Detection of At-risk Students Using Machine Learning

* N. Kondo (Tokyo Metropolitan University), M. Okubo and T. Hatanaka (Osaka University)

Abstract－　In this paper, a detection method of academically at-risk students by using log data of learning management systems is considered. Some well-known machine learning methods are used to build a predictive model of student performance evaluated by GPA. The experimental results indicated that some characteristics of behavior about learning which affect the learning outcomes can be detected with only the LMS log data.
Key Words:　Learning analytics, Institutional Research, Detection of at-risk students, Machine learning, LMS

1. はじめに

この 20 年来の ICT 関連技術の進展や普及にともなって急速に量的拡大を続けるデジタルデータの活用は，マーケティング，医療，交通，ソーシャルメディアなど多方面にわたって推進されている．教育・学習の分野においても，大規模データを活用した教育・学習改善や意思決定支援，価値創出などを推進する動きが続いている．組織的な改善や教育の質保証の観点からは，データに基づく意思決定支援機能をあらわす IR（Institutional Research）が全世界的に発展しつつある[1]．一方で，大規模データをもとに学習者の特性や環境を分析し，効果的・効率的な学習の支援へつなげるための研究分野として，LA（Learning Analytics）がこの数年で大きく成長している[2]．

IR と LA はそれぞれ異なる文脈から発展してきた経緯はあるが，用いるデータの粒度や所属，分析目的などを適切に整理すれば，両者を連続性をもって統一的に取り扱うことは可能であると考えられる[3]．たとえば従来より，IR と LA いずれの領域においても，高リスク（at-risk）な学生を早期に発見し，適切な支援策にむすびつけることがしばしば研究されているが[4][5]，これらの知見を統合することは IR，LA の双方にとって有用であると考えられる．

本研究では，成績不振や留年・退学など，大学生活におけるリスクをもつ学生の早期発見という IR 的な要求に対し，従来 LA 研究において頻繁に用いられてきた LMS（Learning Management System; 学習管理システム）のログデータを活用するアプローチについて検討する．LMS のログデータから学生の行動をとらえる特徴量を抽出し，これを説明変数とするパターン分類問題として高リスク学生の発見を取り扱うことを考え，実際のデータを用いた数値実験によりこれを検証する．

2. LMS と高リスク学生の予測モデル

2.1. 本研究で対象とする LMS

本研究ではX大学のデータをもとに検討を進める．X大学では，全学的に独自LMSを運用している．当LMSは，授業ごとの学習管理に用いられるほか，大学全体の学習におけるポータルサイトとしての機能も果たしており，大学からのお知らせの閲覧，eポートフォリオ，自主学習eラーニングコンテンツなどを含む．大学からのお知らせを確認する必要性や，必修科目において授業の学習管理を当LMSで行っていることなどから，X大学において学習を進めるためには当LMSを一定程度以上利用することが求められるため，当LMSの利用状況は，その学生の大学生活へのコミットメントの程度をある程度反映していると考えられる．

当LMSのログデータは，学生がLMS上で何らかの操作をするごとに，学生ID，操作内容，操作日時のセットを1件のレコードとして記録される．操作内容には，ログイン，ログアウト，授業の受講開始，受講終了，eラーニング教材のプレイヤーの起動，ファイルの提出などがあり，学生が行った操作の詳細が記録される．

2.2. LMS からの特徴量抽出と本研究で用いた変数

本研究では，前節で述べたLMSの操作ログデータから抽出した6つの特徴量および必修科目の欠席率を説明変数とし，1年次前期末のGPAの高低を予測するパターン分類問題を取り扱う．

本研究で用いた変数をTable 1にまとめる．目的変数に用いる「(1) GPA」は，1年次前期末のGPAを2値化したものである．2値化においては，GPAの全学生の平均値をμ，標準偏差をσとしたとき，（$\mu-\sigma$）より大きい場合 「(1) GPA」をhigh，（$\mu-\sigma$）以下の場合lowとしている．「(1) GPA」がlowの学生を本研究では「高リスク」の学生とみなす．

説明変数には Table 1 に示す 7 種類を考えた．「(2) attendance」は，1 年次前期に履修する必修 3 科目の出席率であり，毎週（全 15 週）の授業回ごとに手動で記録される値である．(3)～(8)は自動的に蓄積される LMS のログデータから計算される特徴量であり, LMS 上の行動をさまざまな観点からとらえるために本研究において独自に考えたものである．「(3) player」は，配布資料など授業で用いられる教材や自主学習コンテンツへアクセスするためのプレイヤーを起動した回数である．「(4) night」は，夜間（午前 0 時から 5 時）の

Table 1: 本研究で用いた変数

変数の種類	変数	データソース
説明変数	(1) GPA	成績データ
目的変数	(2) attendance	出欠席データ
	(3) player	LMSログデータ
	(4) night	
	(5) login	
	(6) start	
	(7) submission	
	(8) time	

間の操作回数を表わす．「(5) login」はLMSへのログイン回数である．「(6) start」は，レポートやテストなど，学生がアウトプットする活動のための機能を起動した回数であり，「(7) submission」は，その活動を完了させた回数である．「(8) time」は，ログイン時間の合計（秒）である．

2.3. 機械学習による予測モデル構築

LAやIRにおいては, なんらかの予測を行うために予測モデル（Predictive model）を利用することがしばしば行われる[6]．データの規模が大きくなり，また対象がより複雑になるにつれ，予測モデルには機械学習やデータマイニングの手法を用いることが多くなっている．データにもとづき適切な学習／修学上の支援を行うために，学習の成果や継続に関してリスクがある学生（at-risk students）を早期発見することに関しても多くの研究が行われており[4][5]，この早期発見は予測モデルの運用により実現される．

本研究では，LMSログからの特徴量を中心とする変数をもとに，いくつかの機械学習の手法を用いて高リスク学生を早期発見することを検討する．前節で述べたように，1年次前期末のGPAの高低を予測する2値のパターン分類問題としてこの問題を扱い，機械学習のなかでも，ロジスティック回帰，サポートベクターマシン, およびランダムフォレストによる2値分類を行う．いずれもさまざまな分野で応用の進んでいる有用な手法として知られているものである[7].

3. 数値実験

3.1. 使用したデータと実験環境

本研究では，X 大学 Y 学部における 2015 年度入学生 202 名について，2015 年 4 月 1 日から 8 月 5 日までに記録された 200,979 件のレコードを用いて数値実験を行った．この期間は，入学時オリエンテーションの週から，前期の最終週（第 15 週）までに相当するものである．

予測モデルとしてはロジスティック回帰，サポートベクターマシン, およびランダムフォレストを使用し，Python 3.6.0 と scikit-learn パッケージを用いてこれを実装した．

3.2 高リスク学生の早期発見

変数「(1) GPA」が low である学生を高リスク学生としたとき，そうした学生をなるべく早期に発見することが本研究の前提となる目的である．そこで本実験では，ある授業週において，それまでに得られた全データを用いてモデルの学習を行う，ということを全授業週において行い，モデルの予測性能が週を追うごとにどのように変化するかを検討する．

以下の実験における予測性能の検証では，予測性能の指標として，precision（適合率），recall（再現率）および F 値を用いた．precision は，モデルが予測した分類ラベルが真の値と等しい割合，recall は，真の値のうち，予測モデルによって正しく分類された割合を示す．precision と recall は一般にトレードオフの関係にある．F 値は precision と recall の調和平均であり，両者を総合的に考慮した指標として知られる．本実験では，注目する週において利用可能なデータについて 10-fold cross validation を 10 回行い，指標それぞれについて平均値を示している．

Fig. 1 から Fig. 6 は, 各週における予測性能の違いを予測モデルごとに示したものである．Fig. 1 から Fig. 3 は，LMS ログからの特徴量である(3)〜(8)の変数に加えて，実際の授業への出席率である「(2) attendance」の変数を説明変数に用いた場合の結果であり，Fig. 4 から Fig. 6 は LMS ログからの特徴量のみを説明変数に用いた場合のものである．ここで，週のインデックス「0」は入学時のオリエンテーションの週を示す．

これらの結果から，週を追うごとにいずれの性能指標も上昇傾向にあることが確認できるが，手法によってふるまいが異なることがわかる．サポートベクターマシンは比較的高い precision を示し続ける一方で recall は他の手法より低い値を取り続けており, 実際に高リスクな学生の検出力は相対的に低いことがわかる．ロジスティック回帰は, 出席率を使用する場合（Fig. 1）は precision, recall ともに安定して高い値へと推移しているが，出席率を使用しない場合（Fig. 4）は，サポートベクターマシン同様に precision に比べて recall が低くなっている．これに対し，ランダムフォレストは出席率の使用の有無に関わらず precision と recall のバランスがとれた推移を示している．

実際に高リスクである学生を早期になるべく多く検出するという観点から recall に着目すると，ランダムフォレストを用いれば，春の大型連休前後の第 3 週〜第 4 週の時点で，出席率を使用すると 50%程度，出席率を使用しなければ 40%程度の高リスク学生を検出できている．また，出席率データがまだない第 0 週（オリエンテーション期間）の時点でも 30%程度の検出ができている．

3.3 変数の相対的重要度の時間変化

単に高リスク学生の 2 値分類を行うだけでなく，それぞれの説明変数が予測に寄与する度合いを検証できればさらに有用な情報を得ることができると考えられる．ランダムフォレストを用いる場合，ジニ係数の減少に各説明変数が寄与した割合をもとに，説明変数の相対的重要度を算出することができることから，本実験では相対的重要度の週ごとの変化を検討することとした．

Fig. 7 および Fig. 8 に各変数の相対的重要度の推移を示す．Fig. 7 は説明変数に出席率を用いた場合，Fig. 8 は用いない場合である．とくに出席率を用いない場合に着目し，同じくランダムフォレストで出席率を用いない場合の予測性能を示す Fig. 6 と対応づけて考察する．Fig. 6 において特徴的な変化をみると，第 0 週から第 3 週にかけて予測性能が大きく変化しているほか, 第 9 週から第 10 週へ移るタイミングで性能の大きな向上があることがわかる．Fig. 8 においてこれらの週をみると，こうした変化に呼応して相対的重要度も変化しているように見て取れる．たとえば第 0 週から第 3 週にかけては，ログイン時間やプレイヤー起動数の重要度が小さくなる一方で，ログイン回数や学習完了回数の重要度が大きくなっている．第 9 週から第 10 週においては，逆にログイン回数の重要度が小さくなり，ログイン時間の重要度が大きくなっている．こうした変化点においては，なんらかの学習活動の質的変化が学生に求められている可能性がある．このように,

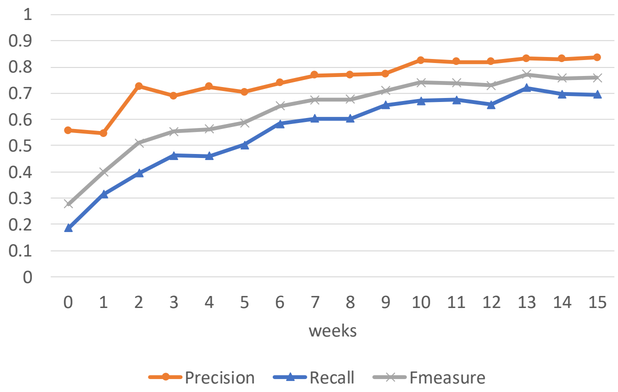


Fig. 1: Classification metrics for logistic regression with attendance data.

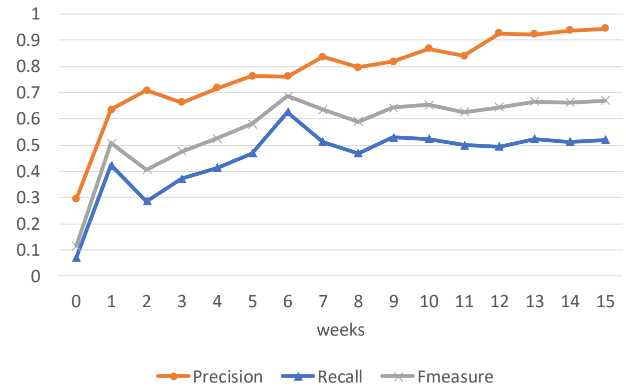


Fig. 2: Classification metrics for SVM with attendance data.

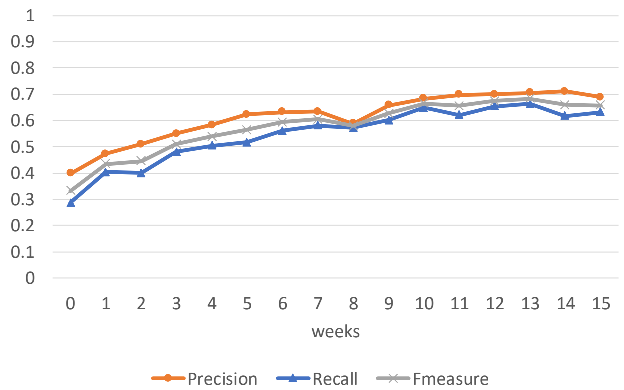


Fig. 3: Classification metrics for random forest with attendance data.

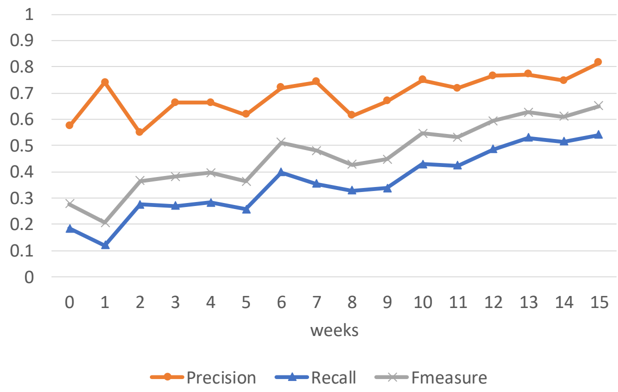


Fig. 4: Classification metrics for logistic regression without attendance data.

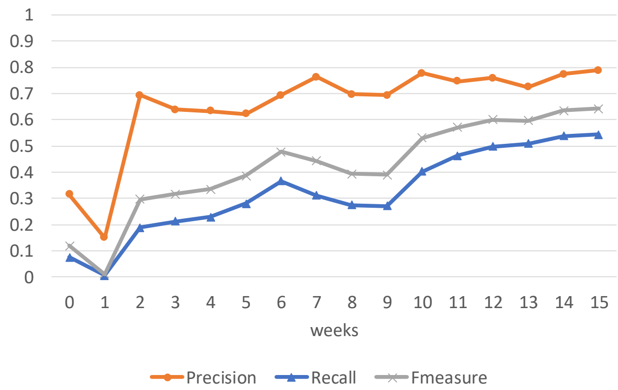


Fig. 5: Classification metrics for SVM without attendance data.

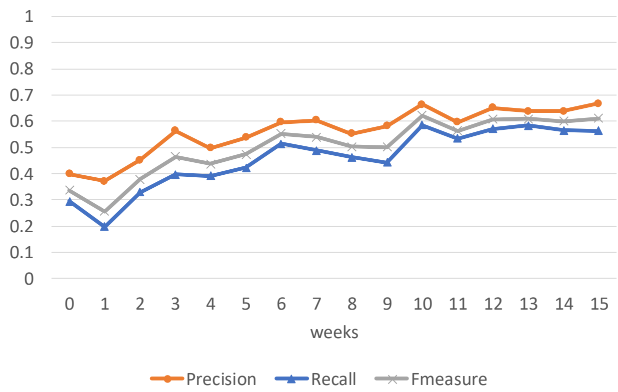


Fig. 6: Classification metrics for random forest without attendance data.

変数の相対的重要度の変化を詳細に検討していくことで，カリキュラムや授業方法などの改善に対する知見が得られるなど，IR としての分析も可能になると考えられる．

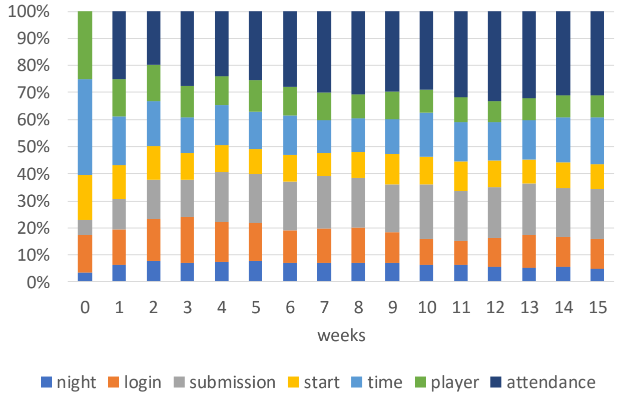


Fig. 7: Weekly change of the comparative importance of explanatory variables with attendance data.

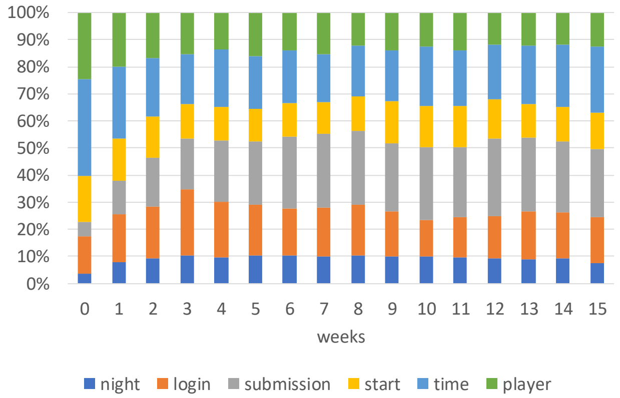


Fig. 8: Weekly change of the comparative importance of explanatory variables without attendance data.

4. おわりに

本研究では，機械学習によって高リスクな学生を早期発見する手法について実験的に検討した．とくに，LMS のログデータから特徴量を抽出して説明変数に使用し，ここへさらに出席率を使用する場合との比較を行った．

実験結果から，本実験で用いたデータセットに対しては，ランダムフォレストを用いると比較的安定した予測が可能であり，LMS のログデータのみでも第 3 週の時点で 40%の高リスク学生を発見できることが示された．この結果は，システムで自動的に蓄積されるデータのみでの高リスク学生の早期発見が一定程度可能であり，効率的な修学支援へ LMS ログが有効に活用できることを示唆している．

さらに，ランダムフォレストで算出できる説明変数の相対的重要度は，その時間変化が予測性能の変化と呼応する傾向がみられた．このような情報は，教育方法やカリキュラム等の改善を検討する際の重要な知見となり得る．組織的な施策検討支援としての IR にも本手法のような分析が貢献できる可能性がある．

謝辞

本研究の一部は JSPS 科研費 JP16K16331 および JP16H03082 の助成を受けた．

参考文献

1) 沖清豪，岡田聡志（編著）：データによる大学教育の自己改善：インスティテューショナル・リサーチの過去・現在・展望，学文社 (2011)
2) The 1st International Conference on Learning Analytics and Knowledge, Call For Papers, https://tekri.athabascau.ca/analytics/call-papers (2017 年 6 月 9 日アクセス)
3) 船守美穂: デジタル技術は高等教育のマス化問題を救えるか？—MOOCs，教育のビッグデータ，教学 IR の模索，情報知識学会誌，**24**-4，424/436 (2014)
4) S. M. Jayaprakash, E. W. Moody, E. J. M. Lauria, J. R. Regan, and J. D. Baron: Early Alert of Academically At-Risk Students: An Open Source Analytics Initiative, Journal of Learning Analytics, **1**-1, 6/47 (2014)
5) 藤原宏司: 学業を中断する学生の予測モデル構築について．大学評価と IR, **5**, 8/22 (2016)
6) C. M. Bishop, Pattern Recognition and Machine Learning, Springer (2006)
7) C. Brooks and C. Thompson: Predictive Modelling in Teaching and Learning, Handbook of Learning Analytics, 61/68 (2017)

第11回コンピューテーショナル・インテリジェンス研究会
2017年6月26日-27日, 大津

多目的機会制約問題に対する チェビシェフの不等式に基づく差分進化

○田川聖治　綿谷剛至　（近畿大学）

Chebyshev Inequality based Differential Evolution For Multi-Objective Chance Constrained Problems

*Kiyoharu Tagawa and Takeshi Watatani (Kindai University)

Abstract– A new approach to solve Multi-objective Chance Constrained Problems (MCCPs) without using the Monte Carlo simulation is proposed. Specifically, according to Chebyshev inequality, the prediction interval of a stochastic function value included in MCCP is estimated from a set of samples. By using the prediction interval, MCCP is transformed into Multi-objective Upper-bound Constrained Problem (MUCP). The feasible solution of MUCP is proved to be feasible for MCCP. For finding a set of solutions of MUCP, an Evolutionary Multi-objective Optimization Algorithm (EMOA) based on Differential Evolution (DE) is also proposed. Through the numerical experiments, the usefulness of the proposed approach is demonstrated.

Key Words: Multi-objective optimization, Chance constrained problem, Differential evolution

1　はじめに

現実の世界の最適化問題では，様々な不確実性を考慮する必要がある．さらに，それらの最適化問題は制約条件のほか，複数の目的関数を持つ場合もある．

不確実性を含む最適化問題は，ロバスト最適化問題，期待値最適化問題，機会制約問題に大別できる．ここで，後述するように，機会制約問題が最も実用的な定式化と考えられる．しかし，その解を求めるためにはモンテカルロ法による確率の計算が必要となる．

著者らは，単目的の機会制約問題に対して，計算コストの大きなモンテカルロ法を用いない求解法を提案している[1, 2]．提案手法では，チェビシェフの不等式[3]から導出した不確実な関数値の予測区間に基づき，機会制約問題を上限制約問題に変換する．関数値が正規分布に従うなら，予測区間の導出は容易である[4]．しかし，正規分布の仮定が有効な問題は限られる．チェビシェフの不等式によれば，関数値の確率分布が未知の場合でも，その予測区間を推定できる．次に，差分進化（DE：Differential Evolution）[5] に基づく最適化アルゴリズムで上限制約問題の最適解を探索する．

本稿では，新たに多目的の機会制約問題（MCCP：Multi-objective Chance Constrained Problem）に対して，モンテカルロ法を使用しない求解法を提案する．提案手法では，上記のような不確実な関数値の予測区間を用いて，MCCP を多目的上限制約問題（MUCP：Multi-objective Upper-bound Constrained Problem）に変換する．また，MUCP の実行可能解は MCCP の実行可能解でもあることを証明する．次に，DE に基づく進化型多目的最適化アルゴリズム（EMOA：Evolutionary Multi-objective Optimization Algorithm）を提案し，MUCP に対するパレート最適解集合を探索する．最後に，数値実験により提案手法の有効性を検証する．

2　不確実性を含む最適化問題

不確実性を含む最適化問題の代用的な 3 種類の定式化を紹介する．決定変数を $\boldsymbol{x} = (x_1, \cdots, x_D) \in \boldsymbol{X}$, $\boldsymbol{X} = [x_j^L, x_j^U]^D \subseteq \Re^D$ とする．不確実性は $\boldsymbol{\Xi}$ を台とする確率変数のベクトル $\boldsymbol{\xi} \in \boldsymbol{\Xi}$ とする．不確実性を含む目的関数を $f : \boldsymbol{X} \times \boldsymbol{\Xi} \to \Re$ とし，制約条件の左辺の関数を $g_k : \boldsymbol{X} \times \boldsymbol{\Xi} \to \Re$, $k = 1, \cdots, K$ とする．

2.1　ロバスト最適化問題

最適化問題の定式化における「ロバスト」の定義は多種多様である[6]．本稿では，最悪状況を想定した最適化問題[7] をロバスト最適化問題と呼ぶ．このため，ロバスト最適化問題を以下のように記述する．

$$\left[\begin{array}{ll} \min\limits_{\boldsymbol{x}\in\boldsymbol{X}} & \max\limits_{\boldsymbol{\xi}\in\boldsymbol{\Xi}} \{f(\boldsymbol{x}, \boldsymbol{\xi})\} \\ \text{sub. to} & \max\limits_{\boldsymbol{\xi}\in\boldsymbol{\Xi}} \{g_k(\boldsymbol{x}, \boldsymbol{\xi})\} \le 0, \ k \in \mathcal{I}_K \end{array}\right. \tag{1}$$

ただし，添字集合を $\mathcal{I}_K = \{1, 2, \cdots, K\}$ とする．

ロバスト最適化問題では最悪状況を考えるため，その解は過剰に保守的となる恐れがある．また，確率変数 $\boldsymbol{\xi} \in \boldsymbol{\Xi}$ のすべての値を考慮することは困難であるため，確率変数 $\boldsymbol{\xi}$ の幾つかの標本に対して式 (1) の問題を解くシナリオアプローチ[8, 9] が考案されている．

2.2　期待値最適化問題

期待値最適化問題は以下のように定式化される．

$$\left[\begin{array}{ll} \min\limits_{\boldsymbol{x}\in\boldsymbol{X}} & E[f(\boldsymbol{x}, \boldsymbol{\xi})] \\ \text{sub. to} & E[g_k(\boldsymbol{x}, \boldsymbol{\xi})] \le 0, \ k \in \mathcal{I}_K \end{array}\right. \tag{2}$$

ただし，$E[v]$ は確率変数 v の期待値である．

期待値は標本平均で近似できるため，期待値最適化問題の解の評価は容易である．このため，進化計算の分野では期待値最適化問題に対する最適化手法が数多く報告されている[10]．しかし，解の平均的な性能評価のみでは，不確実性の影響を十分に考慮できない．

2.3　機会制約問題

機会制約問題は以下のように定式化される[11]．

$$\left[\begin{array}{ll} \min\limits_{\boldsymbol{x}\in\boldsymbol{X}} & \gamma \\ \text{sub. to} & \Pr(f(\boldsymbol{x}, \boldsymbol{\xi}) \le \gamma) \ge 1 - \alpha \\ & \Pr(g_k(\boldsymbol{x}, \boldsymbol{\xi}) \le 0) \ge 1 - \alpha, \ k \in \mathcal{I}_K \end{array}\right. \tag{3}$$

ただし，$\Pr(e)$ は事象 e が起こる確率である．

機会制約問題では制約条件が満たされないリスクを任意の危険率 $\alpha \in (0, 1)$ で指定できる．このため，証券投資や電力供給など，不確実性を含む現実の最適化問題が機会制約問題として記述される [12]．また，多目的の機会制約問題を対象とした EMOA も報告されている [13]．しかし，機会制約問題の解の評価では，モンテカルロ法を用いた確率の計算が必要となる．

3 多目的機会制約問題（MCCP）

M 個の目的関数 $f_m : \boldsymbol{X} \times \boldsymbol{\Xi} \to \Re$, $m = 1, \cdots, M$ を持つ MCCP を以下のように定式化する．

$$\left[\begin{array}{lll} \min\limits_{\boldsymbol{x}\in\boldsymbol{X}} & (\gamma_1, \gamma_2, \cdots, \gamma_M) & \\ \text{sub. to} & \Pr(f_m(\boldsymbol{x}, \boldsymbol{\xi}) \le \gamma_m) \ge 1-\alpha, & \\ & \quad m \in \mathcal{I}_M = \{1, 2, \cdots, M\} & \\ & \Pr(g_k(\boldsymbol{x}, \boldsymbol{\xi}) \le 0) \ge 1-\alpha,\ k \in \mathcal{I}_K & \end{array}\right. \tag{4}$$

式 (4) の MCCP で $\boldsymbol{\gamma} = (\gamma_1, \cdots, \gamma_M) \in \Re^M$ を目的変数ベクトルと呼ぶ．また，MCCP のすべての機会制約条件を満たす解 $\boldsymbol{x} \in \boldsymbol{X}$ を実行可能解と呼ぶ．

MCCP の解 $\boldsymbol{x}^a \in \boldsymbol{X}$ に対する $\boldsymbol{\gamma}^a \in \Re^M$ と，別の解 $\boldsymbol{x}^b \in \boldsymbol{X}$ の $\boldsymbol{\gamma}^b \in \Re^M$ を比較し，以下の関係が成り立つとき，解 $\boldsymbol{x}^a \in \boldsymbol{X}$ は解 $\boldsymbol{x}^b \in \boldsymbol{X}$ を優越するという．

$$(\forall m \in \mathcal{I}_M : \gamma_m^a \le \gamma_m^b) \wedge (\exists \widehat{m} \in \mathcal{I}_M : \gamma_{\widehat{m}}^a < \gamma_{\widehat{m}}^b) \tag{5}$$

ほかの解に優越されない解 $\boldsymbol{x} \in \boldsymbol{X}$ をパレート最適解と呼ぶ．通常，パレート最適解は無数に存在する．

本稿の目的は，式 (4) の MCCP に対して実効可能で多様なパレート最適解の集合を求めることである．

4 多目的上限制約問題（MUCP）

式 (4) の MCCP を MUCP に変換する．

4.1 不確実な関数値の予測区間

ある解 $\boldsymbol{x} \in \boldsymbol{X}$ に対する関数値 $f_m(\boldsymbol{x}, \boldsymbol{\xi})$ と $g_k(\boldsymbol{x}, \boldsymbol{\xi})$ は，不確実性 $\boldsymbol{\xi} \in \boldsymbol{\Xi}$ の影響により観測するたびに値が異なる確率変数となる．ここで，確率変数 $f_m(\boldsymbol{x}, \boldsymbol{\xi})$ の標本 $f_m(\boldsymbol{x}, \boldsymbol{\xi}^n)$, $n = 1, \cdots, N$ から，標本平均 $\overline{f}_m(\boldsymbol{x})$ と標本不偏分散 $s_m^2(\boldsymbol{x})$ を以下のように求める．

$$\overline{f}_m(\boldsymbol{x}) = \frac{1}{N}\sum_{n=1}^{N} f_m(\boldsymbol{x}, \boldsymbol{\xi}^n) \tag{6}$$

$$s_m^2(\boldsymbol{x}) = \frac{1}{N-1}\sum_{n=1}^{N} (f_m(\boldsymbol{x}, \boldsymbol{\xi}^n) - \overline{f}_m(\boldsymbol{x}))^2 \tag{7}$$

標本平均 $\overline{f}_m(\boldsymbol{x})$ と標本不偏分散 $s_m^2(\boldsymbol{x})$ を用いると，チェビシェフの不等式は任意の定数 $\lambda > 0$ に対して

$$\Pr\left(|f_m(\boldsymbol{x}, \boldsymbol{\xi}) - \overline{f}_m(\boldsymbol{x})| \ge \lambda\sqrt{\frac{N+1}{N}}\, s_m(\boldsymbol{x})\right) \le \frac{1}{N+1}\left\lfloor \frac{(N+1)(N-1+\lambda^2)}{N\lambda^2} \right\rfloor \tag{8}$$

となる [3]．ただし，$\lfloor v \rfloor$ は $v \in \Re$ の床関数である．

式 (8) から，以下の定理 1 が証明できる [14]．

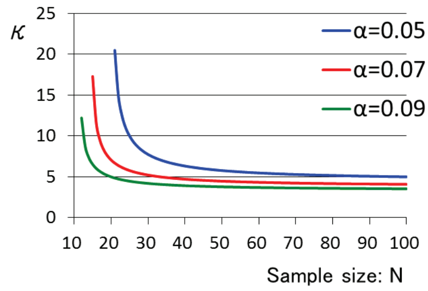


Fig. 1: Change of κ in (10) for N and α.

〔**定理 1**〕$f_m(\boldsymbol{x}, \boldsymbol{\xi})$ の予測区間 $[f_m^L(\boldsymbol{x}),\ f_m^U(\boldsymbol{x})]$ は，

$$\Pr([f_m^L(\boldsymbol{x}),\ f_m^U(\boldsymbol{x})] \ni f_m(\boldsymbol{x}, \boldsymbol{\xi})) \ge 1-\alpha \tag{9}$$

となる．ただし，下限値は $f_m^L(\boldsymbol{x}) = \overline{f}_m(\boldsymbol{x}) - \kappa\, s_m(\boldsymbol{x})$，上限値は $f_m^U(\boldsymbol{x}) = \overline{f}_m(\boldsymbol{x}) + \kappa\, s_m(\boldsymbol{x})$ である．

上記の係数 κ は危険率 α と標本数 N から

$$\kappa = \sqrt{\frac{N^2-1}{N(\alpha N - 1)}} \tag{10}$$

とする．また，標本平均と標本不偏分散の計算に必要な標本数 N の最小値 $N_{\min}$ は以下の通りである．

$$N_{\min} = \left\lfloor \frac{1}{\alpha} + 1 \right\rfloor \tag{11}$$

〔**証明**〕$\lfloor v \rfloor \le v$ より，式 (8) の右辺は

$$\Pr\left(|f_m(\boldsymbol{x}, \boldsymbol{\xi}) - \overline{f}_m(\boldsymbol{x})| \ge \lambda\sqrt{\frac{N+1}{N}}\, s_m(\boldsymbol{x})\right) \le \frac{N-1+\lambda^2}{N\lambda^2} \tag{12}$$

となる．式 (12) で $\kappa = \lambda\sqrt{(N+1)/N}$ と置くと，

$$\Pr(|f_m(\boldsymbol{x}, \boldsymbol{\xi}) - \overline{f}_m(\boldsymbol{x})| \ge \kappa\, s_m(\boldsymbol{x})) \le \frac{N^2-1+N\kappa^2}{N^2\kappa^2} \tag{13}$$

となる．ここで，危険率 α を以下のように与える．

$$\alpha = \frac{N^2-1+N\kappa^2}{N^2\kappa^2} \tag{14}$$

式 (13) と式 (14) から式 (9) の予測区間が得られる．また，式 (14) から式 (10) の κ が得られ，式 (10) と $\kappa > 0$ から，式 (11) の $N_{\min}$ が得られる． ■

Fig. 1 に標本数 N に対する式 (10) の係数 κ の変化を示す．式 (10) から，係数 κ の極限値は

$$\lim_{N\to\infty} \kappa = \lim_{N\to\infty}\sqrt{\frac{N^2-1}{N(\alpha N-1)}} = \sqrt{\frac{1}{\alpha}} \tag{15}$$

となる．また，危険率 α が小さいほど κ は大きい．

4.2　多目的上限制約問題の定式化

MCCP に含まれる関数値に対する式 (9) の予測区間の上限値から，以下のように MUCP を定式化する．

$$\left[\begin{array}{ll}\min\limits_{\boldsymbol{x}\in\boldsymbol{X}} & (f_1^U(\boldsymbol{x}),\, f_2^U(\boldsymbol{x}),\, \cdots,\, f_M^U(\boldsymbol{x})) \\ \text{sub. to} & g_k^U(\boldsymbol{x}) \leq 0,\ k\in\mathcal{I}_K\end{array}\right. \tag{16}$$

ただし，予測区間の上限値の計算に使用する標本数 N は式 (11) の $N_{\min}$ よりも十分に大きいものとする．

〔定理 2〕式 (16) の MUCP の実行可能解 $\boldsymbol{x}$ は，以下の条件下で式 (4) の MCCP の実行可能解となる．

$$\forall m\in\mathcal{I}_M : \gamma_m = f_m^U(\boldsymbol{x}) \tag{17}$$

〔証明〕式 (9) より，MUCP の実行可能解 $\boldsymbol{x}\in\boldsymbol{X}$ は

$$\begin{aligned}&\Pr(f_m(\boldsymbol{x},\boldsymbol{\xi})\leq\gamma_m) = \Pr((-\infty,\ \gamma_m]\ni f_m(\boldsymbol{x},\boldsymbol{\xi}))\\ &\quad\geq \Pr([f_m^L(\boldsymbol{x}),\ f_m^U(\boldsymbol{x})]\ni f_m(\boldsymbol{x},\boldsymbol{\xi}))\geq 1-\alpha\end{aligned}$$

となる．また，式 (9) と $g_k^U(\boldsymbol{x},\boldsymbol{\xi})\leq 0$ から

$$\begin{aligned}&\Pr(g_k(\boldsymbol{x},\boldsymbol{\xi})\leq 0) = \Pr((-\infty,\ 0]\ni g_k(\boldsymbol{x},\boldsymbol{\xi}))\\ &\quad\geq \Pr([g_k^L(\boldsymbol{x}),\ g_k^U(\boldsymbol{x})]\ni g_k(\boldsymbol{x},\boldsymbol{\xi}))\geq 1-\alpha\end{aligned}$$

となり，$\boldsymbol{x}\in\boldsymbol{X}$ は MCCP の制約条件を満たす．　■

5　多目的差分進化

5.1　差分進化の関連研究

単目的の最適化問題における成功に触発され，DE を拡張した EMOA は活発に研究されている [15)]．また，DE に基づく EMOA は現実の設計問題にも適用されている [16)]．DE は集団の更新方法の違いにより同期型と非同期型に大別される．例えば，同期型 DE を拡張した EMOA に GDE3[18)] があり，非同期型 DE を拡張した EMOA に DEMO[19)] がある．一般的に同期型 DE と比較して，非同期型 DE は解の収束性で勝る [17)]．

DE を含む進化計算（EA：Evolutionary Algorithm）は，基本的に制約条件のない最適化問題を対象とする．そこで，制約条件付き最適化問題に EA を適用するため，様々な制約条件の取り扱い法が考案されている．特に，近年では DE を基盤とした制約条件の取り扱い法の報告が増えている [22)]．制約条件の取り扱い法としては，生存選択で実行可能な個体を優先的に選ぶ実行性規則（Feasibility Rule）[23)] のほか，動的ペナルティ法，適応的ペナルティ法 [22)]，ε 制約法 [24)] などがある．

5.2　提案する多目的差分進化（DEMU）

式 (16) の MUCP に対して，DE に基づく EMOA を提案する．新たに提案する EMOA を DEMU と呼ぶ．DEMU では MUCP の解候補を個体と呼び，N_P 個の個体 $\boldsymbol{x}_i$，$i=1,\cdots,N_P$ を集団 $\boldsymbol{P}\subset\boldsymbol{X}$ とする．

DEMU は非同期型の DEMO を基盤とし，3 目的以上の MUCP も対象とするため，優越関係に基づくランク [20)] と参照点による NSGA-III[21)] の生存選択を採用する．また，制約条件の取り扱い法には実行性規則 [23)] を採用し，個体 $\boldsymbol{x}_i\in\boldsymbol{P}$ の制約違反量 $\phi(\boldsymbol{x}_i)$ を

$$\phi(\boldsymbol{x}_i) = \sum_{k\in\mathcal{I}_K}\max\{0,\ g_k^U(\boldsymbol{x}_i)\} \tag{18}$$

とする．$\phi(\boldsymbol{x}_i)>0$ ならば $\boldsymbol{x}_i$ は実行不可能である．

新たな個体の候補であるトライアル・ベクトルの生成には，DE の基本戦略である「DE/rand/1/bin」[5)] を使用する．ただし，その戦略の制御パラメータであるスケール係数 S_F と交叉率 C_R は，Brest ら [25)] が考案した技法を用いて適応的に調整する．

以下に DEMU のアルゴリズムを示す．

手順 1 個体 $\boldsymbol{x}_i\in\boldsymbol{X}$，$i=1,\cdots,N_P$ をランダムに生成して初期集団 $\boldsymbol{P}\subset\boldsymbol{X}$ とする．

手順 2 各個体 $\boldsymbol{x}_i\in\boldsymbol{P}$ を N 回評価し，標本から上限値 $f_m^U(\boldsymbol{x}_i)$，$m\in\mathcal{I}_M$ と $g_k^U(\boldsymbol{x}_i)$，$k\in\mathcal{I}_K$ を求める．

手順 3 終了条件を満たせば，ほかの個体に優越されていない実行可能な個体群 $\boldsymbol{S}\subseteq\boldsymbol{P}$ を出力する．

手順 4 各個体を順番にターゲット・ベクトル $\boldsymbol{x}_{\widehat{i}}\in\boldsymbol{P}$ に指定し，手順 4.1 から手順 4.4 を繰り返す．

手順 4.1 集団 $\boldsymbol{P}$ からランダムに異なる 3 つの個体 $\boldsymbol{x}_{i1}$，$\boldsymbol{x}_{i2}$，$\boldsymbol{x}_{i3}$（$\widehat{i}\neq i1\neq i2\neq i3$）を選び，DE の戦略を用いて，トライアル・ベクトル $\boldsymbol{u}$ の要素 $u_j\in\Re$，$j=1,\cdots,D$ を以下のように決める．

$$u_j = \begin{cases} x_{j,i1} + S_F\,(x_{j,i2}-x_{j,i3}), \\ \qquad \text{if}\quad (\text{rand}_j < C_R)\vee(j=j_r) \\ x_{j,\widehat{i}},\quad \text{otherwise}\end{cases} \tag{19}$$

ただし，$\text{rand}_j\in[0,\,1]$ は一様乱数である．また，添字 $j_r\in[1,\,D]$ はランダムに選択する．

式 (19) で $\boldsymbol{u}$ の要素 u_j が探索範囲 $[x_j^L,\,x_j^U]$ の外に作られた場合，以下のように u_j を修正する [5)]．

$$u_j = \begin{cases} x_{j,i1} + \text{rand}_j\,(x_j^L - x_{j,i1}), & \text{if}\quad u_j < x_j^L \\ x_{j,i1} + \text{rand}_j\,(x_j^U - x_{j,i1}), & \text{if}\quad u_j > x_j^U\end{cases}$$

手順 4.2 上記の $\boldsymbol{u}\in\boldsymbol{X}$ を N 回評価し，標本から上限値 $f_m^U(\boldsymbol{u})$，$m\in\mathcal{I}_M$ と $g_k^U(\boldsymbol{u})$，$k\in\mathcal{I}_K$ を求める．

手順 4.3 下記の条件が満たされれば，直ちにターゲット・ベクトル $\boldsymbol{x}_{\widehat{i}}\in\boldsymbol{P}$ を $\boldsymbol{u}\in\boldsymbol{X}$ で置き換える．

$$\begin{aligned}&(\forall m\in\mathcal{I}_M :\ f_m^U(\boldsymbol{u})\leq f_m^U(\boldsymbol{x}_{\widehat{i}}))\\ &\qquad\wedge(\phi(\boldsymbol{u})\leq\phi(\boldsymbol{x}_{\widehat{i}}))\end{aligned} \tag{20}$$

手順 4.4 下記の条件が満たされれば，トライアル・ベクトル $\boldsymbol{u}\in\boldsymbol{X}$ を集団 $\boldsymbol{P}$ に追加する．

$$\begin{aligned}&(\exists m\in\mathcal{I}_M :\ f_m^U(\boldsymbol{u}) > f_m^U(\boldsymbol{x}_{\widehat{i}}))\\ &\qquad\vee(\phi(\boldsymbol{u}) > \phi(\boldsymbol{x}_{\widehat{i}}))\end{aligned} \tag{21}$$

手順 5 集団の個体数が $|\boldsymbol{P}|=N_P$ なら，手順 6 に進む．個体数が $|\boldsymbol{P}|>N_P$ の場合，下記の手順 5.1 から手順 5.3 の生存選択で $|\boldsymbol{P}|=N_P$ とする．

手順 5.1 集団 $\boldsymbol{P}$ から実行可能な個体を集団 $\boldsymbol{Q}$，実行不可能な個体を集団 $\boldsymbol{Q}^c$ に移す．$\boldsymbol{P}=\emptyset$ とする．

手順 5.2 $|\boldsymbol{Q}| \le N_P$ なら，$\boldsymbol{Q}$ の全個体を $\boldsymbol{P}$ に移し，必要な個数の個体を $\boldsymbol{Q}^c$ から違反量 $\phi(\boldsymbol{x}_i)$ が小さい順に選択して $\boldsymbol{P}$ に追加する．手順 6 に進む．

手順 5.3 $|\boldsymbol{Q}| > N_P$ なら，$\boldsymbol{Q}$ の各個体に優越関係に基づくランク [20] を付け，$\boldsymbol{Q}$ からランクが小さい順に個体を選択して $\boldsymbol{P}$ に移す．同じランクの個体群から個体を選ぶ場合は，後述する参照点に基づく方法 [21] によって $\boldsymbol{Q}$ から $\boldsymbol{P}$ へ個体を移す．

手順 6 手順 3 に戻る．

5.3 参照点に基づく生存選択

提案した DEMU の手順 5.3 で採用した参照点に基づく生存選択 [21] について紹介する．まず，M 次元の目的関数空間に参照点 $\boldsymbol{w}_h \in \Re^M$，$h \in \mathcal{I}_H = \{1, \cdots, H\}$ を均等に配置する．参照点の個数 H は集団サイズ N_P とほぼ同数とする．次に，原点と各参照点を結ぶ参照線を考え，個体 $\boldsymbol{x}_i \in \boldsymbol{Q}$ から参照点 $\boldsymbol{w}_h$ の参照線への垂線距離を，以下のように定義する．

$$d(\boldsymbol{x}_i, \boldsymbol{w}_h) = \left|\left| \boldsymbol{f}^U(\boldsymbol{x}_i) - \frac{(\boldsymbol{w}_h^T \boldsymbol{f}^U(\boldsymbol{x}_i))\, \boldsymbol{w}_h}{||\boldsymbol{w}_h||^2} \right|\right| \tag{22}$$

ただし，$\boldsymbol{f}^U(\boldsymbol{x}_i) = (f_1^U(\boldsymbol{x}_i), \cdots, f_M^U(\boldsymbol{x}_i))$ とする．また，$||\boldsymbol{v}||$ はベクトル $\boldsymbol{v} \in \Re^M$ のノルムである．

Fig. 2 に目的数 $M = 3$ として垂線距離の概念図を示す．$\bullet$ 印は参照点 $\boldsymbol{w}_h \in \Re^M$，$\circ$ 印は目的関数空間における個体 $\boldsymbol{f}^U(\boldsymbol{x}_i) \in \Re^M$，破線は参照線である．

以下に，DEMU の手順 5.3 でランク 1 からランク $r-1$ の個体 $\boldsymbol{x}_i \in \boldsymbol{Q}$ が集団 $\boldsymbol{P}$ に移された後，ランク r の個体群から必要な個数の個体を選ぶ手順を示す．

手順 1 ランク 1 から $r-1$ の個数 $\boldsymbol{x}_i \in \boldsymbol{P}$ と各参照点 $\boldsymbol{w}_h$ との垂線距離を求め，参照点 $\boldsymbol{w}_h$ の混雑度 q_h を垂線距離が最短の個体数とする．ただし，ランクが $r = 1$ の場合，$\forall h \in \mathcal{I}_H : q_h = 0$ とする．

手順 2 混雑度 q_h が最小の参照点 $\boldsymbol{w}_{\widehat{h}}$ を 1 つ選ぶ．混雑度が最小の参照点が複数ある場合は，それらの中からランダムに 1 つの $\boldsymbol{w}_{\widehat{h}}$ を選択する．

手順 3 参照点 $\boldsymbol{w}_{\widehat{h}}$ との垂線距離 $d(\boldsymbol{x}_i, \boldsymbol{w}_{\widehat{h}})$ が最短となる個体 $\boldsymbol{x}_{\widehat{i}} \in \boldsymbol{Q}$ を選び，集団 $\boldsymbol{P}$ に移動する．

手順 4 参照点 $\boldsymbol{w}_{\widehat{h}}$ の混雑度を $q_{\widehat{h}} = q_{\widehat{h}} + 1$ と更新する．

手順 5 必要な個数の個体がランク r の個体群から選ばれるまで，手順 2 から手順 5 の処理を繰り返す．

6 テスト問題の設計

EMOA の標準的なテスト問題集 [26] から，目的数 M を任意に設定できる単峰性の DTLZ2 と多峰性の DTLZ3 を選び，式 (4) の MCCP に拡張する．

6.1 標準的テスト問題

DTLZ2 と DTLZ3[26] は，以下のように制約条件のない M（$M \ge 2$）目的の最小化問題である．

$$\min_{\boldsymbol{x} \in \boldsymbol{X}} \quad (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \cdots, f_M(\boldsymbol{x})) \tag{23}$$

ただし，$\boldsymbol{X} = [0, 1]^D \subseteq \Re^D$ である．

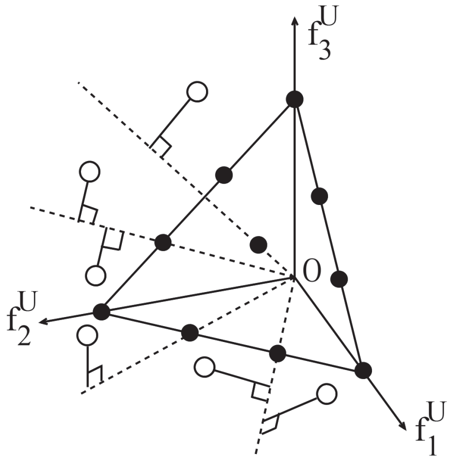


Fig. 2: Reference points in objective space.

決定変数の個数 D は，任意の定数 $Z > 0$ に対して $D = M + Z - 1$ である．ここで，D 個の決定変数 $x_j \in \Re$ のうち，Z 個の決定変数 x_j，$j \in \mathcal{I}_Z = \{M+1, \cdots, D\}$ から構成されるベクトルを $\boldsymbol{x}_Z \in \Re^Z$ とする．

DTLZ2 と DTLZ3 の M 個の目的関数は

$$\left[\begin{array}{l} f_1(\boldsymbol{x}) = (1 + \rho(\boldsymbol{x}_Z))\, c_1\, c_2 \cdots c_{M-2}\, c_{M-1} \\ f_2(\boldsymbol{x}) = (1 + \rho(\boldsymbol{x}_Z))\, c_1\, c_2 \cdots c_{M-2}\, s_{M-1} \\ f_3(\boldsymbol{x}) = (1 + \rho(\boldsymbol{x}_Z))\, c_1\, c_2 \cdots s_{M-2} \\ \qquad \vdots \\ f_{M-1}(\boldsymbol{x}) = (1 + \rho(\boldsymbol{x}_Z))\, c_1\, s_2 \\ f_M(\boldsymbol{x}) = (1 + \rho(\boldsymbol{x}_Z))\, s_1 \end{array}\right. \tag{24}$$

である．c_j と s_j の意味は以下の通りである．

$$c_j = \cos\left(\frac{x_j\, \pi}{2}\right), \qquad s_j = \sin\left(\frac{x_j\, \pi}{2}\right)$$

DTLZ2 では関数 $\rho(\boldsymbol{x}_Z)$ を以下のように定義する．

$$\rho(\boldsymbol{x}_Z) = \sum_{j \in \mathcal{I}_Z} (x_j - 0.5)^2 \tag{25}$$

DTLZ3 では関数 $\rho(\boldsymbol{x}_Z)$ を以下のように定義する．

$$\rho(\boldsymbol{x}_Z) = 100\, [Z + \sum_{j \in \mathcal{I}_Z} ((x_j - 0.5)^2 - \cos(20\, \pi (x_j - 0.5)))] \tag{26}$$

DTLZ2 と DTLZ3 の最適解は等しく

$$\boldsymbol{x}_Z^\star = (0.5, 0.5, \cdots, 0.5) \in \Re^Z \tag{27}$$

であり，$x_j \in [0, 1]$，$j \notin \mathcal{I}_Z$ は任意である．最適解では $\rho(\boldsymbol{x}_Z^\star) = 0$ となり，式 (24) の M 個の目的関数値は，半径 1 の超球面上に分布する．最適解の目的関数値が存在可能な超球面をパレート・フロントと呼ぶ．

6.2 制約条件付きテスト問題

以下のように DELZ2 と DTLZ3 に制約条件を追加したテスト問題を DTLZ2C，DTLZ3C とする．

$$\left[\begin{array}{ll} \displaystyle\min_{\boldsymbol{x} \in \boldsymbol{X}} & (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \cdots, f_M(\boldsymbol{x})) \\ \text{sub. to} & g(\boldsymbol{x}_Z) = \displaystyle\sum_{j \in \mathcal{I}_Z} x_j^2 - 0.5^2\, Z \le 0 \end{array}\right. \tag{28}$$

式 (28) の制約条件を DTLZ2 と DTLZ3 に加えても，各テスト問題の最適解 $\boldsymbol{x}_Z^\star$ は変わらず，目的関数値のパレート・フロントも変わらない．また，$g(\boldsymbol{x}_Z^\star) = 0$ が成り立つため，式 (28) の制約条件は活性である．

6.3 MCCP のテスト問題

上記の各テスト問題の決定変数 x_j，$j \in \mathcal{I}_Z$ に対し，以下のように平均 0，分散 σ^2 の正規分布に従う摂動 ξ_j を加えて式 (4) の MCCP のテスト問題とする．

$$x_j + \xi_j, \quad \xi_j \sim \mathcal{N}(0, \sigma^2), \quad j \in \mathcal{I}_Z \tag{29}$$

DTLZ2 と DTLZ3 では，式 (27) の最適解 $\boldsymbol{x}_Z^\star$ が実行可能領域 $[0, 1]^Z \subset \Re^Z$ の中心にあるため，摂動を加えても最適解の位置は変わらない．一方，DTLZ2C と DTLZ3C では，最適解 $\boldsymbol{x}_Z^\star$ が実行可能領域の境界上にあるため，決定変数に摂動を加えると，その最適解は実行可能領域の内部に移動すると予想される．

7 数値実験

7.1 実験方法

MCCP の 4 種類のテスト問題では，目的数を $M = 3$ または $M = 6$ とした．また，摂動を加える決定変数の個数は $Z = 10$ とした．前述の通り，決定変数の総数は $D = M + Z - 1$ である．さらに，危険率は $\alpha = 0.05$ とした．次に，MCCP の各テスト問題を式 (16) のような MUCP に変換した．MUCP では関数値の上限値を求める際の標本数を Fig. 1 から $N = 80$ とした．

DEMU のプログラムは Java 言語で実装した．DEMU の集団サイズは $N_P = 120$，終了条件は 400 世代とした．ここで，初期集団 $\boldsymbol{P}$ を変えて，MUCP の各テスト問題に対して DEMU を 30 回ずつ適用した．

7.2 実行可能な非劣解の評価指標

各テスト問題に対して DEMU で得られた実行可能な非劣解 $\boldsymbol{S} \subseteq \boldsymbol{X}$ には，多様性と最適性が期待される．本稿では，以下の 4 種類の評価指標を用いる．

7.2.1 実行可能な非劣解の個数

実行可能な非劣解の個数 $|\boldsymbol{S}|$ は，$\boldsymbol{S}$ の多様性を評価する．$|\boldsymbol{S}| = 0$ の場合，ほかの評価指標は求めない．

7.2.2 Maximum Spread

Maximum Spread（MS）は，以下のように目的関数値の範囲の総和であり，$\boldsymbol{S}$ の多様性を評価する．

$$MS(\boldsymbol{S}) = \sum_{m \in \mathcal{I}_M} \left(\max_{\boldsymbol{x}_i \in \boldsymbol{S}} \{f_m(\boldsymbol{x}_i)\} - \min_{\boldsymbol{x}_i \in \boldsymbol{S}} \{f_m(\boldsymbol{x}_i)\} \right) \tag{30}$$

7.2.3 Convergence Measure

Convergence Measure（CM）は，以下のように目的関数値 $(f_1(\boldsymbol{x}_i), \cdots, f_M(\boldsymbol{x}_i)) \in \Re^M$ のパレート・フロントからの平均距離であり，$\boldsymbol{S}$ の最適性を評価する．

$$CM(\boldsymbol{S}) = \frac{1}{|\boldsymbol{S}|} \sum_{\boldsymbol{x}_i \in \boldsymbol{S}} \left\| \sum_{m \in \mathcal{I}_M} f_m(\boldsymbol{x}_i) - 1 \right\| \tag{31}$$

Table 1: Experimental results on DTLZ2

σ^2	$\|\boldsymbol{S}\|$	MS	CM	PD
		$M = 3$		
—	120.0	2.000	2.4E-4	0.008
	—	(2.6E-4)	(1.2E-4)	(0.002)
0.01^2	120.0	2.007	0.007	0.016
	—	(9.1E-4)	(3.5E-4)	(0.001)
0.05^2	120.0	2.166	0.166	0.048
	—	(0.012)	(0.002)	(0.002)
0.1^2	120.0	2.663	0.702	0.074
	—	(0.031)	(0.007)	(0.004)
		$M = 6$		
—	120.0	5.365	0.117	0.225
	—	(0.148)	(0.025)	(0.023)
0.01^2	120.0	5.465	0.164	0.219
	—	(0.180)	(0.039)	(0.028)
0.05^2	120.0	6.062	0.387	0.189
	—	(0.215)	(0.047)	(0.023)
0.1^2	120.0	7.592	1.140	0.213
	—	(0.338)	(0.085)	(0.023)

Table 2: Experimental results on DTLZ2C

σ^2	$\|\boldsymbol{S}\|$	MS	CM	PD
		$M = 3$		
—	120.0	2.000	6.6E-7	2.8E-4
	—	(3.3E-7)	(1.6E-6)	(8.7E-5)
0.01^2	120.0	2.010	0.010	0.029
	—	(8.0E-4)	(2.8E-4)	(8.6E-4)
0.05^2	120.0	2.252	0.252	0.131
	—	(0.015)	(0.003)	(0.002)
0.1^2	120.0	3.051	1.167	0.261
	—	(0.059)	(0.017)	(0.004)
		$M = 6$		
—	120.0	5.211	0.070	0.174
	—	(0.099)	(0.017)	(0.023)
0.01^2	120.0	5.351	0.120	0.185
	—	(0.137)	(0.028)	(0.025)
0.05^2	120.0	6.302	0.529	0.256
	—	(0.220)	(0.048)	(0.018)
0.1^2	120.0	9.071	2.004	0.404
	—	(0.343)	(0.132)	(0.020)

7.2.4 Positional Deviation

Positional Deviation（PD）は，以下のように式 (27) の最適解 $\boldsymbol{x}_Z^\star = (0.5, \cdots, 0.5)$ と個体 $\boldsymbol{x}_i \in \boldsymbol{S}$ との距離の平均値であり，$\boldsymbol{S}$ の最適性を評価する．

$$PD(\boldsymbol{S}) = \frac{1}{|\boldsymbol{S}|} \sum_{\boldsymbol{x}_i \in \boldsymbol{S}} \sqrt{\sum_{j \in \mathcal{I}_Z} (x_{j,i} - 0.5)^2} \tag{32}$$

多様性の評価指標である $|\boldsymbol{S}|$ と MS は大きなほど良い．ただし，$|\boldsymbol{S}| \leq N_P$ である．一方，最適性の評価指標である CM と PD は小さなほど良い．集団 $\boldsymbol{S}$ の全個体がパレート・フロント上にあるとき $CM(\boldsymbol{S}) = 0$ となり，最適解であるとき $PD(\boldsymbol{S}) = 0$ となる．

7.3 実験結果と考察

Table 1 と Table 2 は単峰性の DTLZ2 と DTLZ2C における実験結果である．各表では摂動の分散 σ^2 を変えて求めた個体群 $\boldsymbol{S}$ に対する 4 種類の評価指標の平均

Table 3: Experimental results on DTLZ3

σ^2	$\lvert\boldsymbol{S}\rvert$	MS	CM	PD
		$M=3$		
—	120.0	2.778	1.570	0.027
	—	(1.601)	(4.094)	(0.046)
0.001^2	119.8	16.28	58.51	0.054
	—	(2.915)	(23.81)	(0.070)
0.005^2	118.9	371.1	23E+3	0.299
	—	(97.76)	(2731)	(0.047)
0.01^2	119.6	1268	29E+4	0.509
	—	(158.6)	(15E+3)	(0.076)
		$M=6$		
—	120.0	12.95	8.572	0.080
	—	(8.606)	(13.15)	(0.071)
0.001^2	120.0	67.37	141.8	0.115
	—	(40.97)	(100.9)	(0.088)
0.005^2	120.0	1839	48E+3	0.449
	—	(448.8)	(8183)	(0.052)
0.01^2	120.0	5379	60E+4	0.881
	—	(472.5)	(63E+3)	(0.127)

Table 4: Experimental results on DTLZ3C

σ^2	$\lvert\boldsymbol{S}\rvert$	MS	CM	PD
		$M=3$		
—	120.0	4.005	5.417	0.057
	—	(3.101)	(9.459)	(0.080)
0.001^2	120.0	21.67	115.9	0.161
	—	(7.339)	(111.0)	(0.076)
0.005^2	117.7	392.9	25E+3	0.359
	—	(118.3)	(3309)	(0.090)
0.01^2	119.1	1280	29E+4	0.580
	—	(170.8)	(14E+3)	(0.072)
		$M=6$		
—	120.0	20.61	23.24	0.137
	—	(14.33)	(41.00)	(0.075)
0.001^2	120.0	68.74	160.6	0.164
	—	(28.70)	(106.9)	(0.068)
0.005^2	120.0	1633	46E+3	0.443
	—	(349.0)	(9642)	(0.102)
0.01^2	120.0	5183	59E+4	0.887
	—	(499.1)	(76E+3)	(0.131)

値を示す．σ^2 の値が無記入の行は摂動を加えない場合の結果である．また，括弧内は標準偏差である．

Table 1 と Table 2 では，すべての場合で $|\boldsymbol{S}| = N_P$ であり，集団 $\boldsymbol{P}$ 内の全個体が実行可能な非劣解となる．また，ほかの評価指標は目的数が $M = 3$ の方が小さく，$M = 6$ では解の最適性が不十分である．

Table 1 と比較し，Table 2 では摂動の分散 σ^2 の増加に伴う PD の変化が顕著であり，DTLZ2C の最適解が実行可能領域内に移動することが確認できる．

Table 3 と Table 4 は多峰性の DTLZ3 と DTLZ3C における実験結果である．単峰性のテスト問題に比べて，多峰性のテスト問題の方が摂動の影響が大きいため，摂動の分散 σ^2 の値を 1 桁小さくしている．

Table 3 と Table 4 から，実験結果の傾向は単峰性のテスト問題とほぼ同じである．ただし，多峰性のテスト問題の方が集団の最適化は難しい．特に，摂動の分散が大きな場合，個体群 $\boldsymbol{S}$ は目的関数空間でパレート・フロントから遠く離れた場所に散らばっている．

8　おわりに

本稿では，多目的機会制約問題（MCCP）に対して，モンテカルロ法を使用しない求解法を提案した．提案手法では，チェビシェフの不等式による関数値の予測区間を用いて，MCCP を多目的上限制約問題（MUCP）に変換した後，差分進化に基づく最適化アルゴリズム（DEMU）を MUCP に適用した．また，4 種類のテスト問題において，DEMU の探索性能を検証した．

既存のモンテカルロ法と比較して，関数値の予測区間の推定に必要な標本数は遥かに少ない．しかし，大規模なシミュレーションによって解を評価する現実的な最適化問題では，標本数は可能な限り少ないことが望まれる．今後の課題としては，先行研究[1, 2] で提案している最適解の探索過程における標本数の幾つかの削減手法を，DEMU にも導入することである．

謝辞　本研究は，JSPS 科学研究費補助金（科研費）課題番号 17K06508 の助成を受けたものである．

参考文献

1) 藤田翔平，田川聖治：チェビシェフの不等式に基づく機会制約問題に対する差分進化，第 12 回進化計算学会研究会資料集，16/23 (2017)

2) K. Tagawa and S. Fujita: Chebyshev inequality based approach to chance constrained optimization problems using differential evolution, Proc. of The 8th Int. Conference on Swarm Intelligence (2017) 掲載予定

3) J. G. Saw, M. C. K. Yang, and T. C. Mo: Chebyshev inequality with estimated mean and variance, The American Statistician, **38**-2, 130/132 (1984)

4) 田川聖治，原田翔一：ノイズを含む多目的最適化問題に対する最悪状況の予測に基づく差分進化の適用，電気学会論文誌 C，**136**-2，189/198 (2016)

5) K. V. Price, R. M. Storn, and J. A. Lampinen: Differential Evolution - A Practical Approach to Global Optimization, Springer (2005)

6) A. Parkinson, C. Sorensen, and N. Pourhassan: A general approach for robust optimal design, Journal of Mechanical Design, 115(1), 74/80 (1993)

7) 武田朗子：ロバスト最適化法とその動向，電気学会論文誌 C，**134**-6，760/764 (2014)

8) 藤崎泰正，和田孝之：ロバスト凸最適化のためのランダマイズドアルゴリズム，計測と制御，**50**-11，950/955 (2011)

9) R. Tempo, G. Calafiore, and F. Dabbene: Randomized Algorithms for Analysis and Control of Uncertain Systems, Springer (2012)

10) Y. Jin and J. Branke: Evolutionary optimization in uncertain environments - a survey, IEEE Trans. on Evolutionary Computation, **9**-3, 303/317 (2005)

11) A. Prékopa: Stochastic Programming, Kluwer Academic Publishers (1995)

12) 椎名孝之：確率計画法，朝倉書店 (2015)

13) B. Liu, Q. Zhang, F. V. Fernández, and G. G. E. Gielen: An efficient evolutionary algorithm for chance-constrained bi-objective stochastic optimization, IEEE Trans. on Evolutionary Computation, **17**-6, 786/796 (2013)

14) K. Tagawa: Worst case optimization using chebyshev inequality, Proc . of The 7th Int. Conference on Bioinspired Optimization Methods and Applications, 173/185 (2016)

15) E. Mezura-Montes, M. Reyes-Sierra, and C. A. Coello Coello: Multi-objective optimization using differential evolution: a survey of the state-of-the-art, Advances in Differential Evolution, Springer, 173/196 (2008)

16) 田川聖治，佐々木幸紀，中村弘幸：Differential Evolution による SAW フィルタの多目的最適設計，電気学会論文誌 C，**130**-7，1238/1246 (2010)

17) K. Tagawa: A statistical study of the differential evolution based on continuous generation model, Proc. of CEC2009, 2614/2621 (2009)

18) S. Kukkonen and J. Lampinen: GDE3: The third evolution step of generalized generalized differential evolution, Proc. of IEEE CEC2005, 443/450 (2005)

19) T. Robič and B. Filipič: DEMO: differential evolution for multiobjective optimization, Proc. of EMO2005, LNCS, 3410, 520/533, Springer (2005)

20) K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan: A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. on Evolutionary Computation, **6**-6, 182/197 (2002)

21) K. Deb, H. Jain: A evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints, IEEE Trans. on Evolutionary Computation, **18**-4, 577/601 (2014)

22) E. M. Montes and C. A. Coello Coello: Constraint-handling in nature-inspired numerical optimization: past, present and future, Swarm and Evolutionary Computation, **1**, 173/194 (2011)

23) K. Deb: An efficient constraint handling method for genetic algorithms, Computer Methods in Applied Mechanics and Engineering, **186**, 311/338 (2000)

24) 高濱徹行，阪井節子：ε 制約遺伝的アルゴリズムによる制約付き最適化，情報処理学会論文誌，**47**-6, 1861/1871 (2006)

25) J. Brest, S. Greiner, B. Bošković, M. Merink, and V. Žumer: Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems, IEEE Trans. on Evolutionary Computation, **10**-6, 646/657 (2006)

26) K. Deb, L. Thiele, M. Laumanns, and E. Zitzler: Scalable test problems for evolutionary multiobjective optimization, TIK-Technical Report, **112**, 1/27 (2001)

第11回コンピューテーショナル・インテリジェンス研究会
2017年6月26日-27日，大津

パズルゲームに対する強化学習を用いた汎化性能の向上
キャラクターの経路を決定する方法

○大西鴻哉　飯間等　（京都工芸繊維大学）

Improving Generalization Ability in a Puzzle Game Using Reinforcement Learning
Method to Determine a Character's Path

∗Hiroya Oonishi and Hitoshi Iima (Kyoto Institute of Technology)

Abstract– Nowadays machine learning has attracted much attention. In order to apply it to various problems without relearning, its generalization ability is needed. Geometry Friends is a puzzle game where a character has to collect all targets in a two-dimensional world, and it is used in some artificial intelligence competitions. Sufficient generalization ability is needed to apply the machine learning to this game. We recently proposed the structure of a method based on reinforcement learning in which the generalization ability is improved for Geometry Friends. In this method, a character's path to collect all the targets is determined, and then character's actions to follow the path is learned. We already proposed a method to learn the character's actions. In this paper, we propose a method to determine the character's path.

1　緒言

近年，音声や文字の認識，検索エンジンやゲームプレイなどの分野で機械学習が注目を集めているが，実際の問題に応用するには学習時に用いる訓練データに過適合せず，未知のデータにも対応できる汎化性能を有していることが必要とされる．

種々の機械学習などの方法の性能などを競うビデオゲームのコンペティションがしばしば行われており，このコンペティションで用いられるゲームの1つにGeometry Friends[1] がある．Geometry Friends とは Fig. 1 のように2次元平面上に配置されている目的物を，キャラクターを操作してすべて獲得することを目的としたパズルゲームである．このゲームでは目的物などの位置が異なる様々なレベルがあり，それゆえ汎化性能の優れた手法が必要となる．しかしながら，汎化性能の点で優れた手法は未だ提案されていない．

これまで提案されてきた Geometry Friends に対する強化学習に基づく手法では，Geometry Friends においてどのようなレベルに対しても目的を達成するという問題を，すべての目的物を獲得する経路を決定する問題とその経路に従うためのキャラクターの行動選択の問題という2つの部分問題に分割し，行動選択問題については強化学習[2] を用いて解決しようとしている[3]．しかし学習の対象となる問題設定が大きすぎたために考慮すべき状態が増え，すべての状態の学習を十分に行えず，必ずしも汎化的であるとは言えなかった．

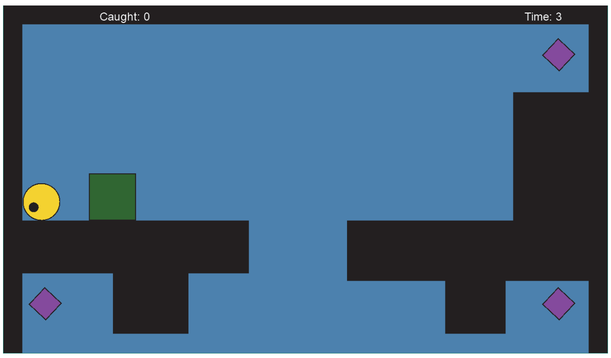


Fig. 1: Geometry Friends のゲーム画面

そこで著者らは，Geometry Friends に対する汎化性能の向上を目的とする手法を以前に提案した[4]．この方法では，従来法[3] と同様に2つの部分問題に分割し，行動選択問題を強化学習で解決する．しかしここでは，前述の学習問題が大きくなりすぎるという問題を解決するため，この学習問題を簡易なものとし，少ない学習回数でもより高い汎化性能が得られる方法を提案した．本論文では，もう1つの問題である経路決定問題を定義し，この問題を解く方法を提案する．そして数値実験を通して，提案手法によりこの部分問題を汎化的に解決できていること，また多くのレベルでゲームの目的を達成できることを示す．

本論文の構成は以下の通りである．まず2章では Geometry Friends とそのコンペティションについての説明を行う．3章では提案手法の構成を説明し，提案手法が2つの部分問題を解くものであることを説明する．そして4章では，本論文で対象とする部分問題を解決する手法を提案する．5章では提案手法を適用する実験を行った結果を示し，提案手法の性能を評価する．6章ではまとめと今後の課題について論ずる．

2　パズルゲーム Geometry Friends

ここでは本論文で対象とするパズルゲーム Geometry Friends を説明する．

2.1　ゲームの説明

Geometry Friends は2次元平面上に複数個配置されたダイヤモンドの形をした目的物のすべてを，黄色の円の形をしたキャラクターおよび緑色の長方形のキャラクターを操作して制限時間以内に獲得するパズルゲームである．

キャラクターは現実と同様に重力や摩擦の力の影響を受けて動き，目的物の獲得はキャラクターが目的物に触れることで達成される．また黒色，黄色，緑色の3種の障害物が配置されており，黒色の障害物は円と長方形のキャラクターが，黄色の障害物は長方形のキャラクターのみが，緑色の障害物は円のキャラクターのみ

が衝突する．Fig. 1 にゲームの画面の例を示す．Fig. 1 で，左寄りに配置されている黄色の円と緑色の長方形はキャラクターであり，黒色のものが障害物である．ゲーム画面上には表示されないが，ゲーム画面の最下端にも黒色の障害物が配置されている．また Fig. 1 のようにキャラクターは障害物の上に乗ることができる．

Geometry Friends には目的物と障害物の位置や，キャラクターの初期位置の異なる様々なレベルが用意されている．そのため，機械学習などの方法によってこのような種々のレベルでゲームの目的を達成させるには，汎化性能の優れた方法が必要となる．

キャラクターが取り得る行動は円と長方形で異なっており，円のキャラクターの場合は次の 4 つとなる．

- 左へ転がる
- 右へ転がる
- ジャンプする
- 円のサイズを変更する

長方形のキャラクターの場合取り得る行動は次の 3 つとなる．

- 左へスライドする
- 右へスライドする
- 長方形を変形する

ここで円のサイズの変更は，円の面積を変更する行動であるのに対し，長方形の変形は長方形の面積は変更せずに，例えば横長な長方形，正方形，縦長な長方形などに変形を行う．

人工知能プレイヤーが Geometry Friends から得られる情報としては次のとおりである．

- キャラクターの位置座標
- キャラクターの垂直速度，水平速度
- 障害物の位置座標，サイズ
- 目的物の位置座標

各々のレベルは円のキャラクターのみを操作するもの，長方形のキャラクターのみを操作するもの，両者を操作するものに分類される．本論文では円のキャラクターのみを操作するレベルのみを対象とする．

2.2 コンペティション

Geometry Friends を人工知能プレイヤーが行うコンペティションが IEEE Conference on Computational Intelligence and Games において 2013 年から開催されている．

コンペティションは円のキャラクターのみを操作するもの，長方形のキャラクターのみを操作するもの，両者のキャラクターを操作するものの 3 つの分野に分かれている．各分野においてレベルはそれぞれ 10 種類用意されている．このうちの 5 種はコンペティションが開催される以前から公開され，参加者はそのレベルを用いて人工知能プレイヤーの性能評価を事前に行うことができる．しかし残り 5 種は非公開であり，コンペティション時に初めて遭遇するレベルとなる．また競技の勝敗は 10 種のレベルの目的をいくつ達成できたか，どれくらい短い時間で達成できたかなどから計算される得点の大小で決定される．

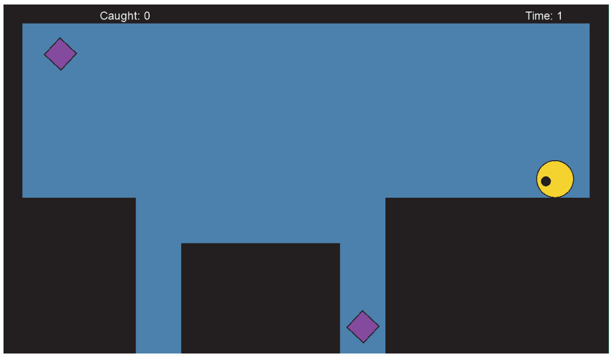


Fig. 2: Geometry Friends のレベルの例

これまでのコンペティションで，機械学習に基づくいくつかの人工知能プレイヤー設計法が提案されてきた [3)5)] が，あるレベルに対しては難なく目的が達成できても，別のレベルに対しては達成が困難といったように，汎化性能の点で優れたものは未だ開発されていない．

3 提案手法

著者らは以前に，Geometry Friends に対して汎化性能の優れた人工知能プレイヤー設計法を提案した [4)]．この提案手法は，Geometry Friends においてレベルの目的を達成するために解決すべき問題が 2 つあることから，これら 2 つの問題を解決するように構成されている．ここでは，その解決すべき 2 つの問題を示し，そしてそれらを解決する提案手法の構成を述べる．

3.1 Geometry Friends において解決すべき問題

Geometry Friends においてレベルの目的を達成するために解決すべき問題は次のように分解できる．

- すべての目的物を最短時間で獲得する経路の決定
- その経路に従うための行動の選択

これらの問題はどのようなレベルにおいてもそのレベルを達成するためには解決すべき共通の問題であり，従ってこれらの問題が解決できればどのレベルに対しても目的が達成できることが言える．

これらの問題をなぜ解決すべきかを，実際のレベルの例を用いて説明する．Fig. 2 に示すレベルの例では障害物と障害物の間に落ちてしまうとキャラクターはそこから抜け出せなくなってしまうため，それを避けるようにして目的物を獲得しなければならない．また，障害物間にある目的物は先ほどの理由により，左上の目的物を獲得した後に，獲得する必要がある．更にレベルには制限時間が設けられているため，できるだけ短い時間ですべての目的物を獲得する必要がある．従ってレベルの目的を達成するためには，以上を考慮した経路を決定する必要がある．

また経路が決定してもキャラクターをその経路に従わせるにはどの行動を選択させるべきかを学習できなければ，レベルの目的を達成できないため，こちらの解決も必要不可欠となる．

3.2　提案手法の構成

Geometry Friends に対してある 1 つのレベルの目的を達成するという問題を設定し，学習できたとしても他のレベルの目的を達成できるとは限らず，従って汎化性能を得ることはできない．そこで前節で示した解決すべき問題をそれぞれ部分問題として設定し，それら部分問題を解決することを通してどのようなレベルの目的も達成することができるよう設計を行うことで汎化性能を得る．

なお，従来法 [3] でも同様の部分問題を設定しているが，行動選択の問題をさらに 2 つの部分問題に分割し，これらを強化学習を用いて解決している．しかし，学習の対象となる問題設定が大きすぎるために，考慮すべき状態が増え，すべての状態の学習を十分に行えず必ずしも汎化的であるとは言えなかった．そこで提案手法では，経路決定問題を解決して経路に従うための情報を得る．次に行動選択問題を，先ほどの情報を用いることで目的の位置に目的の水平速度で達するという簡易な学習問題とし，目的の位置や目的の水平速度が任意であっても行動が学習できる汎化的な強化学習法でこの問題を解決する．

従って提案手法では，未知のレベルに適用する前に，任意の目的の位置および目的の水平速度に対する行動を強化学習で学習をさせておく．次に未知のレベルに適用する時は，経路決定問題を生成してその問題を解き，先ほどの学習結果を用いてその経路に従う行動を実行させる．ここで，未知のレベルの目的も達成できるような汎化性能に優れた手法とするには，経路を決定する手法およびその経路に従う行動を選択する手法が汎化的である必要がある．

文献 4) では行動選択問題を解決する方法を提案したが，経路決定問題は未解決であった．次章では経路決定問題を定義し，その解法を提案する．

4　すべての目的物を獲得する経路の決定法

本章では，キャラクターの初期位置からすべての目的物を最短時間で獲得する経路を決定する問題を設定し，それを解決する手法を提案する．

4.1　提案手法の考え方

まず Fig. 3 に示すように，平坦な障害物のうち，キャラクターが上に乗ることができるものを足場と呼ぶ．またジャンプや落下なしにその足場の左端から右端まで移動できない足場は分割を行って，それらを別の足場とする．従って異なる足場に移動する際には，キャラクターは必ずジャンプか落下のいずれかをしなければならない．また，キャラクターは足場の上で左右に移動したり，ジャンプや落下をすることで目的物を獲得することができる．

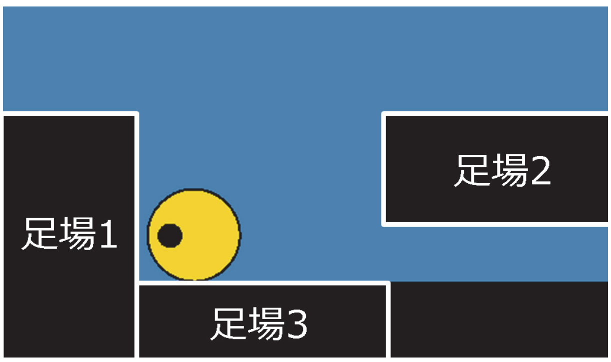


Fig. 3: 足場となる障害物

キャラクターは基本的には種々の足場の上を通りながら，各足場から獲得可能な目的物を獲得することとなる．そこでここでの経路とは，キャラクターが通る足場の順序とする．より具体的には足場を頂点とし，移動可能な足場を辺とした有向グラフを生成し，このグラフ上で最短経路を求める問題を経路決定問題とする．このような経路決定問題には従来から種々の解法が提案されているので，そのうちの 1 つを用いてこの問題を解く．

さて，グラフを生成するには次の 2 つの情報が必要となる．

- 各々の足場から移動可能な足場
- 各々の足場から獲得可能な目的物

これらの情報を取得するためには，キャラクターがジャンプした時および落下した時の軌道を把握する必要がある．これらの軌道は勿論，実際にキャラクターを動作させることで取得することはできるが，これには時間がかかってしまう．特に，コンペティションのレベルでは制限時間が設けられているため，これを行う余裕はない．そこで提案手法ではキャラクターの軌道を予測することとし，この予測した軌道を用いて移動可能な足場と獲得可能な目的物を取得する．

以上をまとめると提案手法では以下の手順を実行することとなる．

1. ジャンプ，落下時のキャラクターの軌道の予測
2. 各々の足場から獲得できる目的物，移動可能な足場の取得
3. 有向グラフの生成
4. 有向グラフに探索アルゴリズムの適用

ここで 1，2 の手順を汎化的に行うことで，どのようなレベルに対してもグラフを生成し，汎化的に最短経路を求めることができると考えられる．これらの手順については次節より具体的に説明する．

また，従来法 [3] でも同様の有向グラフが生成され経路が決定されるが，それぞれの手順が提案手法と異なっている．これらの違いについても，次節より具体的に説明する．

4.2　キャラクターの軌道の予測

Geometry Friends ではレベルにより障害物の位置が異なるため，キャラクターがジャンプした時や落下した時の軌道は障害物との衝突を考慮するとレベルにより異なると言える．そこでまず，キャラクターが障害物に衝突しない場合の軌道を求め，その軌道上に障害物があった場合はその障害物に衝突した後の軌道を求めることで，どのようなレベルでも軌道の予測が行えるようにする．

キャラクターがジャンプ，落下した時の軌道は，軌道上に障害物が存在しない場合は放物線であると考え

られる．これより，軌道は次式に従うこととなる．

$$x = v_x t \tag{1}$$

$$y = v_y t - \frac{1}{2} g t^2 \tag{2}$$

ここで t は時間で，キャラクターがジャンプまたは落下し始めた時間を 0 としており，x と y はそれぞれ $t=0$ でのキャラクターの位置座標を原点とする時の水平方向，垂直方向の座標である．v_x と v_y はそれぞれ $t=0$ でのキャラクターの水平速度，垂直速度で，g は重力に基づく垂直方向の加速度である．ここで，いずれの時間においてもキャラクターの水平速度は一定で，v_x に等しいと仮定している．

式 (1)，式 (2) より，もし v_x，v_y および g の値をゲームから取得することができれば，キャラクターの軌道を表す x と y を予測することができる．しかしながら，これらの値の取得はゲームの仕様上困難であり，従ってこれらの値を推定することが必要とされる．但し，落下時の v_y の値は経験的に 0 であると考えられるため 0 に設定する．また，v_x の値は仮定により任意の時間でのキャラクターの水平速度と等しいため，この値をゲームから取得して v_x に設定する．

ジャンプ時の v_y と g の推定は最適化問題を解くことで行う．v_y，g の推定値をそれぞれ $\hat{v}_y$，$\hat{g}$ とし，これらの推定値を式 (2) に代入して得られた y を $\hat{y}$ とする．最適化問題において，最小化する目的関数 f を次式とする．

$$f = \sum_t \bar{f}_t \tag{3}$$

$$\bar{f}_t = \frac{1}{2}(\hat{y}(t) - y(t))^2 \tag{4}$$

ここで決定変数は $\hat{v}_y$ と $\hat{g}$ である．この最適化問題を解くために，提案手法では機械学習で広く用いられている確率的勾配降下法[6] を用いる．この手法の 1 回の反復において，キャラクターをジャンプまたは落下させ，時間 t に対して $\hat{v}_y$ と $\hat{g}$ を次式により更新する．

$$\begin{bmatrix} \hat{v}_y \\ \hat{g} \end{bmatrix} = \begin{bmatrix} \hat{v}_y \\ \hat{g} \end{bmatrix} + \alpha \nabla \bar{f}_t(\hat{v}_y, \hat{g}) \tag{5}$$

ここで，α は学習率，$\nabla \bar{f}$ は $\bar{f}$ の勾配である．また $y(t)$ が推定値の更新に必要なため，この値を適当な時間でゲームから取得し，取得した時間を t とする．以上の手順を実行することで v_y，v_x，g の値を得ることができるため，障害物に衝突しない場合のキャラクターの軌道は予測することができる．

次に，キャラクターが障害物に衝突した後の軌道を考える．ここでは，単純にキャラクターが障害物に衝突した後，キャラクターは真下に落下すると仮定する．このような単純な動きでも，実際の軌道との誤差は十分に小さいと経験的に考えられる．

4.3　獲得できる目的物と移動可能な足場の取得

従来法[3] では 2 つの足場間が移動可能かどうかは，その 2 つの足場がどれぐらい垂直方向に離れているか，また 2 つの足場間に障害物が存在するかを見て，例えば障害物があればキャラクターはその障害物に衝突するので移動不可能と判定し，なければ移動可能と判定する．しかし，2 つの足場間に障害物が存在しても必ずしもその障害物に衝突するとは言えず，従って誤判定が多くなることが考えられる．また，ある足場からどの目的物が獲得できるかについては，その足場の真上にあり，かつその足場から垂直にある程度離れていない目的物のみ獲得できると判定している．しかしこちらも例えば足場間に存在する目的物に対して，どの足場からも獲得できないと誤判定されることが多くなってしまうことが考えられる．

そこで提案手法では，前節で予測された軌道を用いることでこれらをより正確に判定する．具体的には，まず足場上のすべての位置においてキャラクターをすべての水平速度でジャンプまたは落下させた場合の軌道を予測する．そしてこの軌道上で，もし障害物が存在していなければキャラクターは障害物に衝突しないと判定する．移動可能な足場と獲得可能な目的物についても同様に判定する．ここで，いずれの軌道も前節の手法により予測ができるため，これらの判定もどのようなレベルに対しても行うことができる．しかしながら，足場上のすべての位置，すべての水平速度に対して軌道を予測するには計算に時間を要してしまうため，限定されたいくつかの位置および水平速度に対してのみ予測する．

これらの判定の結果，次のような情報が複数個得られる．

- ジャンプまたは落下を行う足場
- キャラクターの行動 (ジャンプまたは落下)
- ジャンプまたは落下する位置
- ジャンプまたは落下する時の水平速度
- 獲得可能な目的物
- 着地する足場
- 着地する位置

これらの情報は有向グラフを生成する際に用いられる．

4.4　有向グラフの生成

それぞれの足場を頂点とし，前節で得られた情報に基づき辺を結ぶことで有向グラフを生成する．具体的には，それぞれの情報に対して「ジャンプまたは落下を行う足場」に相当する頂点から「着地する足場」に相当する頂点に向かって辺を結ぶ．また辺のコストはキャラクターの移動距離，具体的にはキャラクターの現在の位置から「ジャンプまたは落下する位置」までの距離と「ジャンプまたは落下する位置」から「着地する位置」までの距離の和とする．更に辺の属性として，その情報の「キャラクターの行動」,「ジャンプまたは落下する位置」,「ジャンプまたは落下する時の水平速度」,「獲得可能な目的物」を設定する．

生成した有向グラフにおいて，キャラクターのある頂点からある頂点への移動は，その頂点間に結ばれた辺を経由することで行うことができる．辺で結ばれている隣同士の頂点間の移動は，キャラクターをその辺の属性の「ジャンプまたは落下する位置」に「ジャンプ

または落下する時の水平速度」で到達させ,「キャラクターの行動」を実行することで実現できる．また，移動の際に「獲得可能な目的物」を獲得できる．

前章で述べたようにキャラクターをジャンプまたは落下させるための位置，水平速度に到達させる行動は強化学習で学習される [4) が，この学習法では足場の長さが十分にある環境で学習が行われている．従って，特に足場が短い場合に，足場から落下することなくキャラクターはその位置，水平速度に到達できない問題が生じることがある．この問題を具体的に説明すると，キャラクターがある位置，水平速度に到達することを考える時，キャラクターはその位置からある程度の助走距離が必要となることが経験的に分かっている．従って，もしキャラクターが乗っている足場に助走するだけの十分の長さがなかった場合，学習で獲得した行動を選択しても，キャラクターは助走の段階で足場から落下してしまい，目標の足場に着地したり，目標の目的物を獲得したりすることができない．そこで提案手法では，学習で獲得した行動選択を用いて，それぞれの水平速度に到達するためにどれ位の助走距離が必要となるかをあらかじめ求めておく．そして，有向グラフの辺を設定する際に，もし足場が短く助走距離を確保できない場合は，その辺は経由不可と判定し設定しないものとする．

助走距離の求め方をより具体的に説明する．まず Fig. 4 に示されるレベルに対して，キャラクターの初期位置に，任意の水平速度で到達することを目的として，キャラクターの行動を学習で獲得した通りに選択させる．この時キャラクターは初期位置からある程度離れてから，再度初期位置に向かって動き，初期位置にその水平速度で到達する．そこで，キャラクターが初期位置から最も離れた時の距離を求め，これをその水平速度で到達するための助走距離とする．

また，前節で得られたすべての情報に対して辺を生成すると，1 つの頂点に対して多数の辺が接続する場合があり，このようなグラフに対して最短経路を求めるため探索アルゴリズムを適用すると，計算量が大きくなってしまう可能性がある．そこで，不要と考えられる辺は消去することでこの問題を解決する．具体的には，まず接続している 2 つの頂点が同じで，かつ「獲得可能な目的物」がない辺は明らかに不要であるので消去する．また接続している頂点,「獲得可能な目的物」および「ジャンプまたは落下する時の水平速度」の正負がそれぞれ同じ辺については，それらの中で最も「ジャンプまたは落下する時の水平速度」の絶対値が小さい

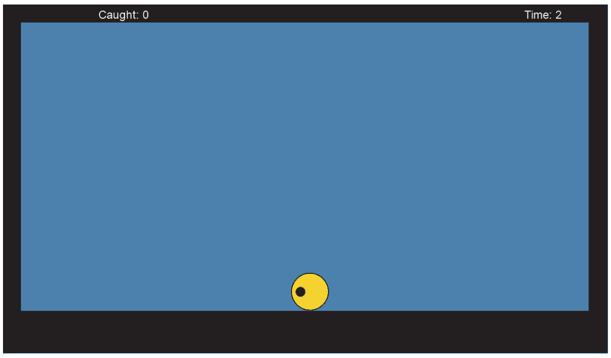


Fig. 4: 助走距離を求めるレベル

初期位置でキャラクターが乗っている足場に相当する頂点を初期頂点 n_0 とし $f(n_0) = 0$ とする．
オープンリストとクローズドリストを空に初期化し，オープンリストに n_0 を加える．
while オープンリストが空でない **do**
　オープンリストに含まれている頂点のうち，$f(n)$ が最も低い頂点 n をクローズドリストに移す．
　n が目標とする頂点であれば探索を終了する．
　for all n から接続している頂点 m **do**
　　$f'(m) = f(n) + c(n, m)$ とする．
　　if m がオープンリストに含まれていて，かつ $f(m) > f'(m)$ **then**
　　　$f(m) = f'(m)$ とする．
　　else if m がクローズドリストに含まれていて，かつ $f(m) > f'(m)$ **then**
　　　$f(m) = f'(m)$ とし，m をクローズドリストからオープンリストに移す．
　　else if m がオープンリストとクローズドリストいずれにも含まれていない **then**
　　　$f(m) = f'(m)$ とし，オープンリストに m を加える．
　　end if
　end for
end while

Fig. 5: Subgoal A*

もののみを残し，残りは消去する．これは水平速度の絶対値が小さければ，キャラクターがその辺を経由した際，着地する足場からさらに落下して経路通りに移動できなくなる危険性が低いと考えたためである．

4.5　有向グラフへの探索アルゴリズムの適用

有向グラフ上で最短経路を求めるために，探索アルゴリズムの 1 つである Subgoal A*[7) を適用する．Subgoal A*とは A*探索アルゴリズムを改良した手法であり，目標とする頂点が複数ある場合にも最短経路を求めることができる．Subgoal A*のアルゴリズムを Fig. 5 に示す．ここでオープンリストとクローズドリストは頂点のリストであり，$f(n)$ はそれまでに求められた初期頂点 n_0 から頂点 n に到達するまでに要する最小コストである．$c(n, m)$ は隣同士の頂点 n と頂点 m の間にある辺のコストである．

また，Subgoal A*では各頂点に，その頂点に至るまでにキャラクターがどの目的物を獲得したかという情報を属性として設定している．この情報は，その頂点に到達するまでに経由した辺の属性から得られる．そして，Fig. 5 における目標とする頂点はキャラクターがすべての目的物を獲得した頂点とする．従ってこの属性により，複数の目的物をすべて獲得したときにアルゴリズムを正しく終了させることができる．

なお，この属性を設定せずに Subgoal A*を適用した場合，グラフによってはすべての目的物を獲得できないという問題が生じる．この問題が生じる例を Fig. 6 に示す有向グラフを用いて説明する．ここでは A，B，C が頂点であり，矢印が有向辺，ダイヤモンドはその辺の属性に「獲得可能な目的物」が設定されていることを表している．また，A を初期頂点 n_0 とする．Subgoal A*では，まず $f(\mathrm{A}) = 0$ として A から出発し，B と C でより短いコストで移動できる頂点に移動する．例え

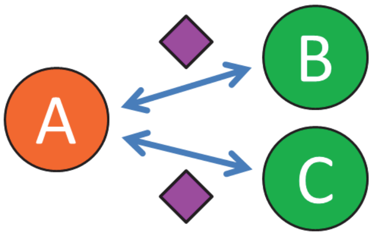


Fig. 6: 有向グラフの例

ば，A から B に移動し，経由した辺の目的物が獲得されたとする．この結果，B がオープンリストに加えられ，次は $n = \mathrm{B}$，$m = \mathrm{A}$ の場合を考えることとなる．しかしここで $f(\mathrm{A}) = 0$ であったので，$f(\mathrm{A}) > f'(\mathrm{A})$ を満たす $f'(\mathrm{A})$ は得られないことから，そこで探索が終了してすべての目的物を獲得できなくなってしまう．この問題を解決するために，先に追加した属性を利用し，辺を通ったときに新たな目的物を獲得していれば，直前の頂点を異なる頂点として区別する．従って Fig. 6 の例では，目的物を獲得した後の A は異なる頂点として区別される．これにより，目的物獲得後の A を A′ とすると，A′ はオープンリストにもクローズドリストにも含まれていないことから，A から B，B から A′，A′ から C の順で移動する経路を得ることができる．

5 数値実験

本章では提案手法を適用する実験を行い，その結果から提案手法により経路決定問題を汎化的に解決できること，また過去のコンペティションで用いられたレベルのほとんどでゲームの目的を達成できることを示す．

5.1 経路決定手法の性能評価

本節では 4 章で提案した経路決定手法を Geometry Friends のレベルに適用し，その性能を評価する．

まず，ジャンプ，落下時のキャラクターの軌道を予測する手法の性能を評価する．確率的勾配降下法では，$\hat{v}_y$，$\hat{g}$ の初期値を 0，学習率 α を 0.1，更新の反復回数を 500 とした．$\hat{v}_y$，$\hat{g}$，$\bar{f}_t$ の値の推移をそれぞれ Figs. 7, 8, 9 に示す．ここで y 軸は $\hat{v}_y$，$\hat{g}$，$\bar{f}_t$ それぞれの値を表し，x 軸は更新回数を表している．Fig. 9 から分かるように，$\bar{f}_t$ が 0 に収束していることから，式 (3) における目的関数 f は 0 となって誤差は存在しないと言える．また Fig. 7 や Fig. 8 から分かるように，$\hat{v}_y$，$\hat{g}$ の値も収束している．以上よりキャラクターの軌道は正しく予測できると言える．

次にキャラクターの移動可能な足場および獲得可能な目的物の判定を行う．この判定は 4.3 節で説明したように，キャラクターの軌道を予測することで行う．軌道の予測は，限定されたいくつかの位置および水平速度でジャンプまたは落下する場合にのみ行う．キャラクターの水平速度の絶対値が取り得る最大値はおよそ 180 であることが確認されているので，ここでは足場上でそれぞれの間隔の長さが 16 である位置および $-180, -160, \cdots, 0, \cdots, 160, 180$ の水平速度に対して軌道の予測を行った．

次に有向グラフを生成するが，ここでは 4.4 節で説明したようにそれぞれの水平速度に対する助走距離を求める必要がある．ただし，軌道の予測では前述のように $-180, -160, \cdots, 0, \cdots, 160, 180$ の水平速度に対してのみ行ったため，ここでも同様にこれらの水平速度に対してのみ助走距離を求めれば良い．それぞれの水平速度に対して求められた助走距離を Table 1 に示す．表には負の水平速度に対する助走距離が示されていないが，Geometry Friends におけるキャラクターの動きは左右対称であることが経験的に分かっているため，負の水平速度に対する助走距離は，絶対値が等しい正の水平速度に対する助走距離を用いることとする．

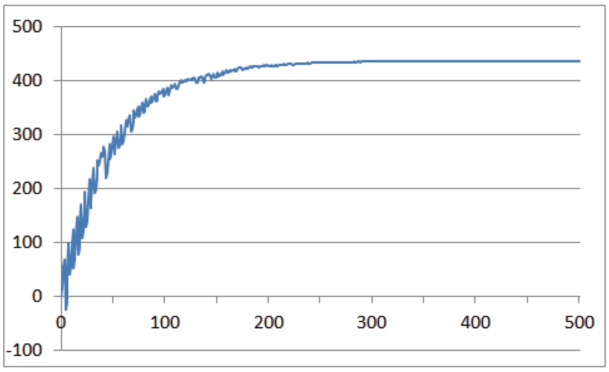


Fig. 7: $\hat{v}_y$ の推移

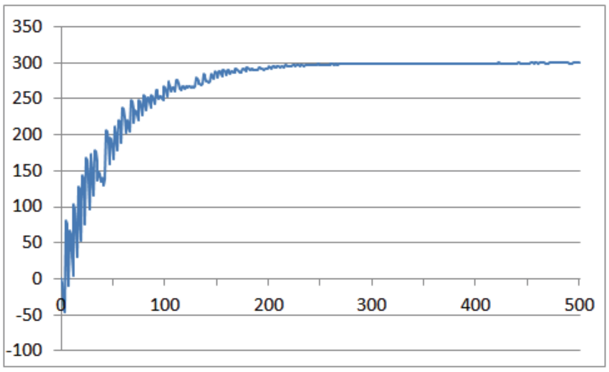


Fig. 8: $\hat{g}$ の推移

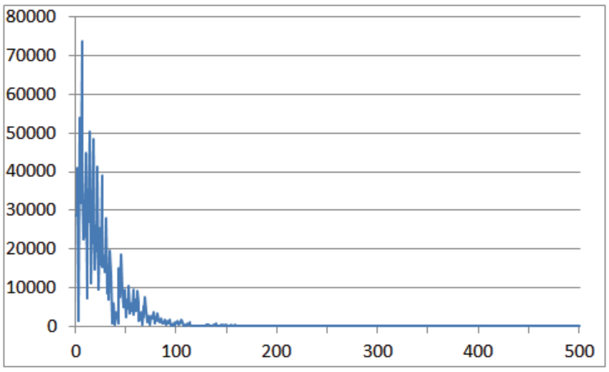


Fig. 9: $\bar{f}_t$ の推移

最後に，生成した有向グラフに Subgoal A*を適用することで経路を決定する．決定された経路の一例を Fig. 10 に示す．ここでは白線が決定された経路を表しており，このレベルにおいてはこの経路に従うことでキャラクターは最短時間ですべての目的物を獲得でき

Table 1: 水平速度に対する助走距離

水平速度	0	20	40	60	80
助走距離	0	14	30	48	68
水平速度	100	120	140	160	180
助走距離	90	112	137	166	197

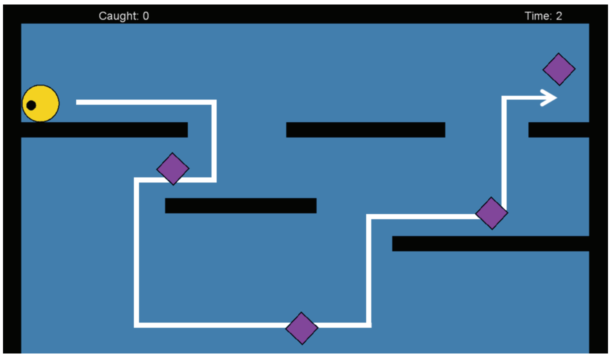


Fig. 10: 決定された経路

ると思われる．また他のレベルにおいても同様の経路が決定されることを確認している．しかしながら，ある特定のレベルにおいては有向グラフの生成の段階で，すべての目的物を獲得するために必要な辺が生成されないという問題が生じた．これは 4.2 節で説明したように，キャラクターは障害物に衝突後，真下に落下すると仮定したために，実際には到達可能である足場に到達できないと誤判定したためであると考えられる．この問題を除けば，提案手法により最短時間ですべての目的物を獲得する経路を汎化的に決定できると言える．

5.2　提案手法全体の性能評価

本節では，提案手法により設計した人工知能プレイヤー (AI プレイヤー) に Geometry Freinds の過去のコンペティションで使用された 10 種のレベルをプレイさせる．そして，その結果を従来の AI プレイヤーである Agent α[3]，CIBot[8] の結果と比較し，提案手法により汎化性能を向上できていることを示す．Agent α は強化学習に基づく手法で設計された AI プレイヤーで，CIBot は 2014 年に IEEE Conference on Computational Intelligence and Games で開催されたコンペティションで優勝した AI プレイヤーである．

Agent α，CIBot および提案手法の AI プレイヤーによる結果を Tables 2，3，4 に示す．ここで Table 2 と Table 3 は Geometry Friends の Web ページ[1] からの引用である．各々の AI プレイヤーは 10 種のレベルそれぞれを 10 回ずつプレイしており，Level はそれぞれのレベルを，Success は 10 回のうち何回すべての目的物を獲得できたかを表している．Target は各回で獲得した目的物の個数の平均を表しており，括弧内の数字はそのレベルに配置されている目的物の個数である．Time はすべての目的物を獲得するために要した時間の平均を秒数で表しており，括弧内の数字はそのレベルの制限時間である．Score は各 AI プレイヤーに優劣をつける際に用いられ，次式で計算される $ScoreRun$ の平均である．

$$ScoreRun = V_{Completed} \times \frac{(maxTime - agentTime)}{maxTime} + (V_{Collect} \times N_{Collect}) \qquad (6)$$

ここで $V_{Completed}$ はすべての目的物を獲得したときの得点で値として 1000 が用いられており，$V_{Collect}$ は 1 つの目的物を獲得したときの得点で値として 100 が用いられている．$maxTime$ は各レベルの制限時間，

Table 2: Agent α による結果

Level	Success	Target	Time	Score
1	8	1.5(2)	13.0(20)	495
2	0	1.6(3)	45.0(45)	160
3	0	1.0(3)	60.0(60)	100
4	0	2.0(4)	80.0(80)	200
5	2	1.2(2)	65.7(70)	182
6	6	1.4(2)	22.8(40)	574
7	1	0.8(3)	56.5(60)	138
8	3	2.3(3)	38.9(40)	257
9	2	2.2(3)	75.6(80)	275
10	0	0.6(3)	100.0(100)	60
Total Score				2441

Table 3: CIBot による結果

Level	Success	Target	Time	Score
1	10	2(2)	12.7(20)	567
2	10	3(3)	19.9(45)	858
3	10	3(3)	14.8(60)	1053
4	0	1.2(4)	80.0(80)	120
5	0	1(2)	70.0(70)	100
6	0	1(2)	40.0(40)	100
7	10	3(3)	26.2(60)	864
8	0	0(3)	40.0(40)	0
9	10	3(3)	50.0(80)	675
10	0	0(3)	100.0(100)	0
Total Score				4337

Table 4: 提案手法の AI プレイヤーによる結果

Level	Success	Target	Time	Score
1	10	2(2)	11.4(20)	630
2	10	3(3)	21.4(45)	824
3	10	3(3)	25.4(60)	877
4	10	4(4)	21.4(80)	1133
5	10	2(2)	24.8(70)	846
6	10	2(2)	13(40)	875
7	10	3(3)	19.4(60)	977
8	10	3(3)	28.6(40)	585
9	10	3(3)	31.4(80)	908
10	0	2(3)	100.0(100)	200
Total Score				7853

$agentTime$ は目的物をすべて獲得するために要した時間 (制限時間内にすべて獲得できなかった場合はそのレベルの $maxTime$ が $agentTime$ となる)，$N_{Collect}$ は獲得した目的物の数を表す．コンペティションでは各レベルの Score の合計である Total Score が最も高い AI プレイヤーが優勝となる．

Tables 2，3，4 より，提案手法の AI プレイヤーが他の AI プレイヤーよりも高い得点を得ていることから，提案手法により汎化性能を向上できていることが確認できる．しかしながら，Table 4 から分かるように，提案手法の AI プレイヤーがすべての目的物を獲得できないレベルが 1 つあった．これは，前節で説明したように必要な辺が生成されないという問題のためである．

6 結言

本論文では，Geometry Friends に対する汎化性能の向上を目的として強化学習を用いた AI プレイヤーの設計法を提案した．提案手法では，まずどのようなレベルに対しても目的を達成するという問題を経路決定の問題と決定された経路に従うための行動選択の問題という 2 つの部分問題に分割し，それらを解決することであらゆるレベルの目的を達成できるようにしている．特に本論文では，経路決定問題を汎化的に解決する手法を提案した．そして数値実験の結果より，提案手法により経路決定問題を汎化的に解決できていること，そしてこのゲームに対する汎化性能を向上できていることを示した．

参考文献

1) GAIPS INESC-ID laboratory, Geometry Friends — Cooperation Puzzle Game — The Cooperative Agent Competition, <http://gaips.inesc-id.pt/geometryfriends/> (2017/05/28).

2) 牧野，渋谷，白川 (編)，これからの強化学習，森北出版，2016.

3) João Quitério, Rui Prada, Francisco S. Melo, "A Reinforcement Learning Approach for the Circle Agent of Geometry Friends," IEEE Conference on Computational Intelligence and Games, pp. 423-430, Tainan, Taiwan. August 2015.

4) 大西 鴻哉，飯間 等，パズルゲームに対する強化学習を用いた汎化性能の向上，第 61 回システム制御情報学会研究発表講演会，2017 年 5 月.

5) Yen-Wen Lin, Li-Jia Wei and Tao-Hsing Chang, "KUAS-IS Lab Technical Report," Geometry Friends Game AI Competition, 2014.

6) 岡谷 貴之，深層学習，講談社，2015.

7) Daniel Fischer, "Development of Search-Based for the Physics-Based Simulation Game Geometry Friends," Geometry Friends Game AI Competition, 2015.

8) Du-Mim Yoon,Kyung-Joong Kim, "CI Bot Circle Technical Report," Geometry Friends Game AI Competition, 2014.

第11回コンピューテーショナル・インテリジェンス研究会
2017年6月26日-27日，大津

球面上のデイジーワールドモデルにおけるパターン形成

○陰山真矢　八木厚志　（大阪大学）

Pattern Formation in Daisyworld Model on a Sphere

∗M. Kageyama and A. Yagi (Osaka University)

Abstract– A planetary biota modifies its environment and its environment regulates a biota by natural selection. Daisyworld has been introduced by James Lovelock (1983) as a simple parable to verify a hypothesis that the feedback between a biota and its environment keeps the planetary surface environment stable and habitable for a biota. In the original Daisyworld model defined by Andrew Watson and James Lovelock (1983), the whole Earth is regarded as a single point. Its numerical results showed how the interaction between a biota and its environment affect themselves. Here the Daisyworld is extended to complete two-dimensional version on the sphere. The asymptotic behavior of solutions to the model and some numerical results of the model are discussed.

Key Words: Daisyworld, self-regulating homeostasis, reaction-diffusion system, sphere, finite-difference methods, pattern formation

1　はじめに

1972 年に J.E. Lovelock は地球における自己調節恒常性システムの存在を提唱した．これは，生物とそれを取り巻く環境が相互に作用することによって，地球システム全体を安定化に向けて自己調節するような機能である．このようなシステムを単純化したものが Lovelock [5] によるデイジーワールドである．

デイジーワールドは地球によく似た惑星で，恒星の周りを公転している．デイジーワールドには黒色と白色の２種類のデイジーしか存在せず，これらは地球の植物と同様に生育域を争っている．この仮想の惑星を照らす恒星は，太陽と同じく年月を経るごとに熱出力が上昇するという特徴をもつ．また，デイジーワールドは雲ひとつない惑星であり，二酸化炭素のようなガスは常に安定した状態にある．これはデイジーの生育には十分だが，気候に影響を及ぼすほどではない．したがって，惑星の大域温度は恒星からの放射熱と地表面の光反射率（アルベド）によってのみ決まる．つまり，地表面の色が濃いほど恒星からの光を吸収して温度が上昇しやすく，色が薄いほど光を反射して温度が低下しやすい．Watson-Lovelock [8] によって導入されたデイジーワールドの方程式は，惑星全体をひとつの点として考えた０次元のモデルだった．これは非常に単純なものであったにもかかわらず，２種類のデイジーが互いの生育域を争いながら，惑星の大域的な気温を自らの成長に最適な値へと自律的に調節していくような結果を示した．

現在，地球上の生物が環境に何らかの影響を与えていることについてはもはや議論の余地はない．しかし，デイジーワールドモデルは植生と環境の相互作用とその根底にあるメカニズムを再現する最も単純な数理モデルのひとつとして多くの分野で注目されており，様々な関連研究や拡張研究が行われている．例えば空間次元の拡張においては，von Bloh-Block-Schellnhuber [7] によるセル・オートマトン法を用いた平面拡散モデル，Adams-Carr [1]，Adams-Carr-Lenton-White [2] による円周拡散モデルなどがある．本研究では，デイジーワールドモデルを２次元球面上へ拡張し，モデルの定式化を行う．この球面上への拡張によって，パターン形成における新たな効果が生まれることを期待する．さらに，この２次元モデルに対して，抽象放物型発展方程式の理論 [10] を用いて，指数アトラクタを構成する．最後に，数値計算によって得られたパターン解について述べる．

2　球面上のデイジーワールドモデル

２次元球面 S 上で以下の初期値･境界値問題を考える．

$$
(1)\quad \begin{cases}
\dfrac{\partial u}{\partial t} = d\varDelta u + [(1-u-v)\Phi(u,v,w) - f]\,u, \\
\dfrac{\partial v}{\partial t} = d\varDelta v + [(1-u-v)\Psi(u,v,w) - f]\,v, \\
\dfrac{\partial w}{\partial t} = D\varDelta w + [1-g(u,v)]R(\omega) - \sigma w^4, \\
u(\omega,0) = u_0(\omega), \;\; v(\omega,0) = v_0(\omega), \\
w(\omega,0) = w_0(\omega).
\end{cases}
$$

ただし，領域 S は，

$$S = \mathbb{S}^2 \equiv \left\{\omega = (x,y,z) \in \mathbb{R}^3; x^2+y^2+z^2 = l^2\right\},$$

とする．未知変数 $u = u(\omega,t)$，$v = v(\omega,t)$ は，場所 $\omega \in S$，時刻 t での白色と黒色のデイジーの密度分布をそれぞれ示す．したがって，各 (ω,t) で $u \geq 0, v \geq 0, u+v \leq 1$ を満たしている．$w = w(\omega,t)$ は地表面温度を示している．各デイジーと地表面温度はそれぞれ d, D の割合で球面 S 上を拡散する．したがって，$\varDelta$ は球面上のラプラシアン（Laplace-Beltrami 作用素）である．このとき，デイジーと温度の関係から $0 < d < D$ と仮定する．$g(u,v)$ は地表面の平均アルベドを表し，u, v の関数として各 (ω,t) で次のように与えられる．

$$g(u,v) = A_w u + A_b v + A_g(1-u-v).$$

ここで，A_w, A_b, A_g はそれぞれ白色のデイジー，黒色のデイジー，裸地のアルベドであり，$A_w = 0.75, A_b = 0.25, A_g = 0.5$ の場合を考えている．さらに，$\Phi(u,v,w)$ と $\Psi(u,v,w)$ はそれぞれ白色のデイジーと黒色のデイ

ジーの成長率を表し，

$$\Phi(u,v,w) = \{1-\delta(\overline{w}-w-q[g(u,v)-A_w])^2\}_+,$$
$$\Psi(u,v,w) = \{1-\delta(\overline{w}-w-q[g(u,v)-A_b])^2\}_+,$$

によって決まる．$\overline{w}$ はデイジーの成長に最適な温度，q,δ は適当な正定数である．式 (1) の第 3 式はエネルギー平衡方程式から導き出され，σ はシュテファン・ボルツマン定数である．$R(\omega)$ は太陽から流入するエネルギー量を表し，

$$R(\omega) = R_0\sqrt{1-(z/\ell)^2},$$

で定義される．

3 ラプラシアンの定式化

S 上でナブラ演算子 ∇ を

$$\nabla u(\omega) = [\nabla_{\mathbb{R}^3}\tilde{u}(x,y,z)]_{|S}, \qquad \omega\in S,$$

と定める．$\mathbb{R}^3$ 内で ∇u を計算すると，

$$\nabla u = \Big(\sin\theta\cos\phi\frac{\partial u}{\partial r} + \frac{\cos\theta\cos\phi}{r}\frac{\partial u}{\partial\theta} - \frac{\sin\phi}{r\sin\theta}\frac{\partial u}{\partial\phi},$$
$$\sin\theta\sin\phi\frac{\partial u}{\partial r} + \frac{\cos\theta\sin\phi}{r}\frac{\partial u}{\partial\theta} + \frac{\cos\phi}{r\sin\theta}\frac{\partial u}{\partial\phi},$$
$$\cos\theta\frac{\partial u}{\partial r} - \frac{\sin\theta}{r}\frac{\partial u}{\partial\theta}\Big),$$

が得られる．r 方向に定数であることから，$\frac{\partial\tilde{u}}{\partial r}=0$ となるので，S 上では，

$$\nabla u = \frac{1}{\ell}\Bigg(\cos\theta\cos\phi\frac{\partial u}{\partial\theta} - \frac{\sin\phi}{\sin\theta}\frac{\partial u}{\partial\phi},$$
$$\cos\theta\sin\phi\frac{\partial u}{\partial\theta} + \frac{\cos\phi}{\sin\theta}\frac{\partial u}{\partial\phi}, -\sin\theta\frac{\partial u}{\partial\theta}\Bigg),$$

となる．

さらに，S 上のラプラシアン Δ を次のように定義する．

$$\Delta u(\omega) = [\nabla_{\mathbb{R}^3}\cdot\nabla_{\mathbb{R}^3}\tilde{u}(x,y,z)]_{|S}, \qquad \omega\in S.$$

$\mathbb{R}^3$ 内で Δu を計算すると，

$$\Delta u =$$
$$\frac{1}{r^2}\left[\frac{\partial}{\partial r}\left(r^2\frac{\partial u}{\partial r}\right) + \frac{1}{\sin\theta}\frac{\partial}{\partial\theta}\left(\sin\theta\frac{\partial u}{\partial\theta}\right) + \frac{1}{\sin^2\theta}\frac{\partial^2 u}{\partial\phi^2}\right],$$

を得る．$\frac{\partial\tilde{u}}{\partial r}=0$ より，S 上では，

$$\Delta u = \frac{1}{\ell^2}\left[\frac{1}{\sin\theta}\frac{\partial}{\partial\theta}\left(\sin\theta\frac{\partial u}{\partial\theta}\right) + \frac{1}{\sin^2\theta}\frac{\partial^2 u}{\partial\phi^2}\right],$$

となる．

ここで，半双線形形式

$$a(u,v) = \int_S \nabla u\cdot\nabla\overline{v}d\omega + \int_S u\overline{v}d\omega, \qquad u,v\in H^1(S),$$

を考える．すべての $u\in H^1(S)$ に対して，$a(u,v) = (f,v)_{L_2}$ であるような唯一つの $f\in L_2(S)$ が存在し，$\Lambda u = f$ と定める．すなわち，

$$\begin{cases}\mathcal{D}(\Lambda) = \big\{u\in H^1(S); \\ \qquad\qquad (u,v)_{H^1} = (f,v)_{L_2}, {}^\forall v\in H^1(S)\big\}, \\ \Lambda u = f,\end{cases}$$

とする．$\mathcal{D}(\Lambda)$ は線形空間，$\Lambda:\mathcal{D}(\Lambda)\to L_2(S)$ は線形作用素である．このとき，

$$-\Delta u = f,$$

となることから，$-\Delta u$ は $L_2(S)$ の非負自己共役作用素である．

4 主結果

問題 (1) に対して局所解，大域解を構成する．さらに適切な無限次元力学系を定義し，指数アトラクタの構成を行う．

基礎空間

$$X = \big\{U = {}^t(u,v,w);\, u,v,w\in L_2(S)\big\},$$

で問題 (1) は Cauchy 問題

$$\begin{cases}\dfrac{dU}{dt} + \mathcal{A}U = F(U), \quad 0<t<\infty, \\ U(0) = U_0,\end{cases}$$

に定式化される．初期関数の空間を，

$$K = \Big\{U_0 = {}^t(u_0,v_0,w_0)\in X;$$
$$u_0\ge 0, v_0\ge 0, u_0+v_0\le 1, 0\le w_0\le (R_0/\sigma)^{\frac14}\Big\},$$

とする．

定理 4.1 任意の初期関数 $U_0\in K$ に対して，関数空間

$$U\in\mathcal{C}((0,T_{U_0}];\mathcal{D}(\mathcal{A}))\cap\mathcal{C}([0,T_{U_0}];X)\cap\mathcal{C}^1((0,T_{U_0}];X),$$

で唯一の局所解が存在する．ここで時間 $T_{U_0}>0$ はノルム $\|U_0\|_X$ によってのみ決まる．

定理 4.2 任意の $U_0\in K$ に対して，関数空間

$$U\in\mathcal{C}((0,\infty);\mathcal{D}(\mathcal{A}))\cap\mathcal{C}([0,\infty);X)\cap\mathcal{C}^1((0,\infty);X),$$

で唯一の大域解が存在する．

さらに，非線形半群を

$$S(t)u_0 = u(t;u_0), \qquad u_0\in K,$$

と定めると，$(S(t),K,X)$ は非線形力学系となる．

定理 4.3 力学系 $(S(t),K,X)$ に対して指数アトラクタが構成できる．

5 球面上での数値計算手法

球面上の偏微分方程式に対する数値計算手法については，地球ダイナモシミュレーションの分野などで多く開発されている．例えば，Kageyama-Sato[4)] による Yin-Yang 格子，Ronch-Iacono-Paolucci[6)] による Cubed Sphere 格子，Half-Step-Shifted 格子（例えば，Fornberg-Merrill[3)] を参照）を用いた計算手法などがある（また，Williamson[9)] にいくつかの手法がまとめられている）．これらの数値計算手法のほとんどは計算コストと計算精度のトレードオフとなっている．

本研究では，陽的 Half-Step-Shifted 格子スキームを用いて計算を行う．

5.1 陽的 Half-Step-Shifted 格子スキーム

球面上での数値計算においては，空間を等間隔に分割する緯度経度格子（Fig. 1 参照）を用いるのが比較的簡単である．しかし，この緯度経度格子を用いる場合，極付近でのいくつかの問題に注意する必要がある．まず，極が座標特異点となっていることがある．この問題については，極上に格子点を置かない，L'Hôpital の定理を用いて解くべき方程式を座標特異点のない形に書き換えるなどの方法によって回避することができる．今回の場合は，極上から半グリッドずつずらした格子（Half-Step-Shifted grid）を考える（Fig. 2 参照）．もうひとつの問題は極付近に格子点が集中することにより，経度方向の格子点間隔が極めて狭くなることである．これについては緯度方向の格子間隔を極に近づくほど狭くとった不等間隔格子を用いることで対応する．

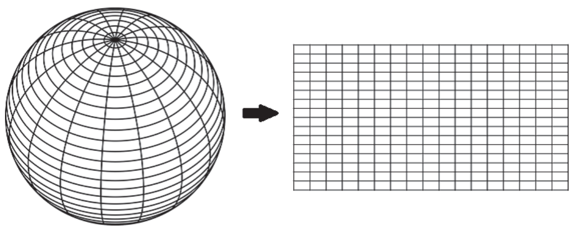


Fig. 1: Latitudinal-Longitudinal grids.

したがって，空間離散化における i 番目の緯度格子点 θ_i と j 番目の経度格子点 ϕ_j は以下のように定義される．

$$\begin{aligned} \theta_i &= \left(i - \frac{1}{2}\right)\Delta\theta_i, && (i = 1, 2, \cdots, N), \\ \phi_j &= j\Delta\phi, && (j = 0, 1, \cdots, M). \end{aligned}$$

ここで，N, M はそれぞれ緯度方向と経度方向の格子点総数を表し，$\Delta\theta_i$ は極に近いほど狭くなるような θ 方向の非一様格子幅，$\Delta\phi$ は ϕ 方向の一様な格子幅（$\Delta\phi = 2\pi/M$）である．また，n 番目時刻ステップを $t_n = n\Delta t$ とする．

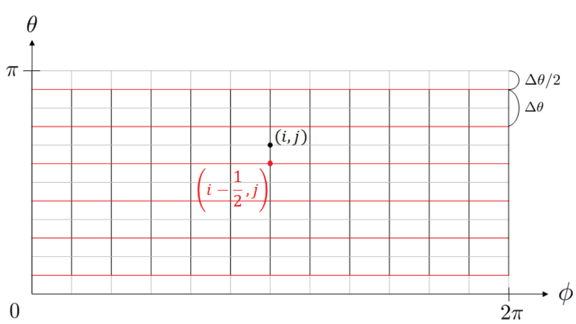


Fig. 2: Black point (i, j) is on the Latitudinal-Longitudinal grid. Red point $(i - 1/2, j)$ is on the Half-Step-Shifted grid.

問題 (1) の解 $u(\theta_i, \phi_j, t_n)$, $v(\theta_i, \phi_j, t_n)$, $w(\theta_i, \phi_j, t_n)$ に対応する近似値を，それぞれ $U^n_{i,j}, V^n_{i,j}, W^n_{i,j}$ とする．

このとき問題 (1) は以下のように離散化される．

$$\begin{aligned} &\frac{U^{n+1}_{i,j} - U^n_{i,j}}{\Delta t} \\ &= \left[\left(1 - U^n_{i,j} - V^n_{i,j}\right)\Phi(U^n_{i,j}, V^n_{i,j}, W^n_{i,j}) - f\right]U^n_{i,j} \\ &\quad + d\left[\frac{\sin\theta_{i+\frac{1}{2}}}{\sin\theta_i}\frac{U^n_{i+1,j} - U^n_{i,j}}{(\Delta\theta_i)^2} - \frac{\sin\theta_{i-\frac{1}{2}}}{\sin\theta_i}\frac{U^n_{i,j} - U^n_{i-1,j}}{(\Delta\theta_i)^2}\right. \\ &\qquad\qquad \left. + \frac{1}{\sin^2\theta_i}\frac{U^n_{i,j+1} - 2U^n_{i,j} + U^n_{i,j-1}}{(\Delta\phi)^2}\right], \end{aligned}$$

$$\begin{aligned} &\frac{V^{n+1}_{i,j} - V^n_{i,j}}{\Delta t} \\ &= \left[\left(1 - U^n_{i,j} - V^n_{i,j}\right)\Psi(U^n_{i,j}, V^n_{i,j}, W^n_{i,j}) - f\right]V^n_{i,j} \\ &\quad + d\left[\frac{\sin\theta_{i+\frac{1}{2}}}{\sin\theta_i}\frac{V^n_{i+1,j} - V^n_{i,j}}{(\Delta\theta_i)^2} - \frac{\sin\theta_{i-\frac{1}{2}}}{\sin\theta_i}\frac{V^n_{i,j} - V^n_{i-1,j}}{(\Delta\theta_i)^2}\right. \\ &\qquad\qquad \left. + \frac{1}{\sin^2\theta_i}\frac{V^n_{i,j+1} - 2V^n_{i,j} + V^n_{i,j-1}}{(\Delta\phi)^2}\right], \end{aligned}$$

$$\begin{aligned} &\frac{W^{n+1}_{i,j} - W^n_{i,j}}{\Delta t} \\ &= \left[1 - g(U^n_{i,j}, V^n_{i,j})\right] - \sigma(W^n_{i,j})^4 \\ &\quad + D\left[\frac{\sin\theta_{i+\frac{1}{2}}}{\sin\theta_i}\frac{W^n_{i+1,j} - W^n_{i,j}}{(\Delta\theta_i)^2} - \frac{\sin\theta_{i-\frac{1}{2}}}{\sin\theta_i}\frac{W^n_{i,j} - W^n_{i-1,j}}{(\Delta\theta_i)^2}\right. \\ &\qquad\qquad \left. + \frac{1}{\sin^2\theta_i}\frac{W^n_{i,j+1} - 2W^n_{i,j} + W^n_{i,j-1}}{(\Delta\phi)^2}\right]. \end{aligned}$$

数値計算上の境界条件については，経度方向（$j = 0, M$）は周期境界条件，緯度方向（$i = 1, N$）は，

$$\begin{aligned} U_{0,j} &= U_{1,\frac{M}{2}+j}, \\ U_{N+1,j} &= U_{N,\frac{M}{2}+j}, && (j = 0, 1, ..., M/2), \\ U_{0,j} &= U_{1,-\frac{M}{2}+j}, \\ U_{N+1,j} &= U_{N,-\frac{M}{2}+j}, && (j = M/2 + 1, ..., J), \end{aligned}$$

を満たす．

6 数値計算結果

$\ell = 1$ であるような球面 S を考え，$0 < \theta < \pi$, $0 \le \phi \le 2\pi$ とする．ここで各パラメータの値は $d = 10^{-6}$, $D = 1.0$, $\delta = 0.003265$, $f = 0.3$, $\overline{w} = 295.5$, $q = 40$, $\sigma = 5.67 \times 10^{-8}$ で与えられる．また惑星に流入するエネルギー $R(\theta)$ は，

$$R(\theta) = \frac{4 \cdot 917}{\pi} L \sin\theta,$$

によって決まる．ここでの L は太陽光度を表し，$L = 0.85$ の場合を考える．

以上の仮定で問題 (1) に対する数値計算結果を Fig. 3 に示す．Fig. 3 は時刻 $t = 600$ での (a) 白色デイジーの

密度分布，(b) 黒色デイジーの密度分布，(c) 温度分布をそれぞれ表す．これらの解は安定な状態となっており，グローバル・アトラクタへと引き寄せられていることを示唆している．

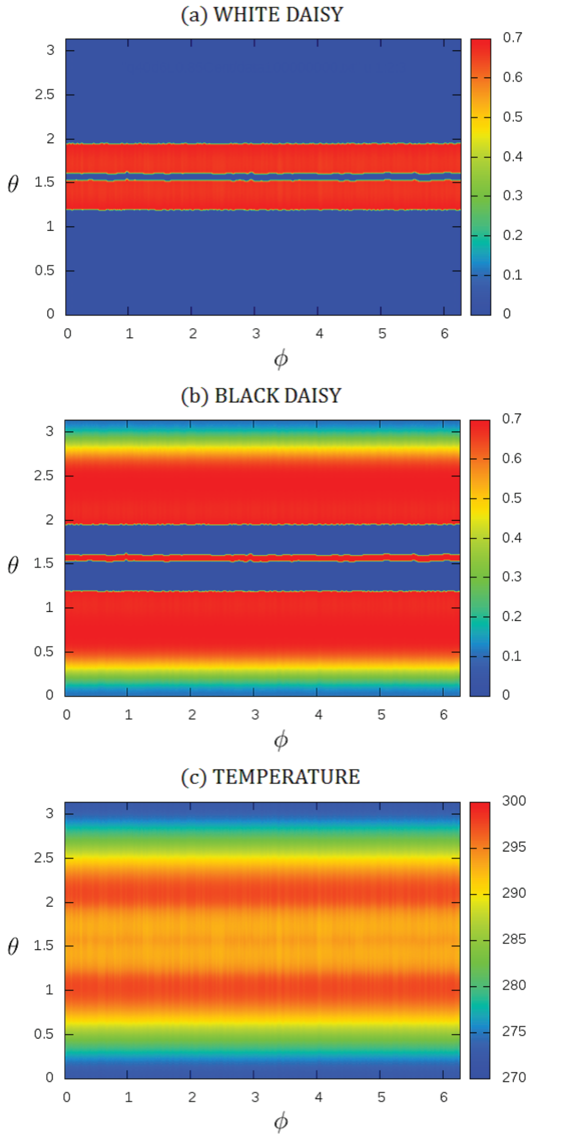


Fig. 3: (a) Graph of $u(\theta, \phi)$, (b) Graph of $v(\theta, \phi)$ and (c) Graph of $w(\theta, \phi)$ at time $t = 600$.

Fig. 3(a),(b) で見られるように，2 種類のデイジーはその境界面を除いて完全に分かれて生息している．白色デイジーは光を反射しやすいため，高温地帯である赤道付近で優位に育ち，その反対に黒色デイジーは光を吸収しやすいため，比較的温度の低い極付近で優位に育っている．また，白色デイジーの冷却効果によって，その生育域である赤道で温度の低下が見られる．この温度低下によって黒色デイジーが赤道上で生育可能となっている．

さらに興味深いことに，赤道付近における白色デイジーと黒色デイジーの境界線が赤道とほぼ平行なジグザグ曲線で与えられていることが見てとれる．このような 2 次元パターンが現れることについては，今後更なる検討と考察を重ねたい．またこれらの結果については，別の計算手法によっても同様のものが得られている．

7 まとめ

地球における自己調節恒常性システムについて簡単に述べ，そのシステムを再現する非常に単純な数理モデルであるデイジーワールドモデルを 2 次元球面上で定式化した．さらに，その方程式に対して局所解，大域解，無限次元力学系，指数アトラクタの構成を行い，数値計算結果の一部を紹介した．

今後は数値計算によって現れた 2 次元パターンについて検討を重ねたい．また，今回扱った球面デイジーワールドモデルに対しても従来モデルと同じくデイジーによる温度の調節が見られるかについて，さらなる数値計算によって確認したい．

参考文献

1) B. Adams and J. Carr: *Spatial pattern formation in a model of vegetation-climate feedback*, Nonlinearity **16**, 1339/1357, (2003).

2) B. Adams, J. Carr, T. M. Lenton, and A. White: *One-dimensional daisyworld: spatial interactions and pattern formation*, J Theor Biol **223**, 505/513, (2003).

3) B. Fornberg and D. Merrill: *Comparison of finite difference- and pseudospectral methods for convective flow over a sphere*, Geophys Res Lett **24**, 3245/3248, (1997).

4) A. Kageyama and T. Sato: *'Yin-Yang grid': An overset grid in spherical geometry*, Geochem Geophys Geosyst **5**, 1/15, (2004).

5) J. E. Lovelock: *Gaia as seen through the atmosphere*, in Biomineralization and Biological Metal Accumulation, edited by P. Westbroek and E. W. deJong, D. Reidel, Dordrecht, Netherlands, 15/25, (1983).

6) C. Ronchi, R. Iacono, and P. S. Paolucci: *The "Cubed sphere": A new method for the solution of partial differential equations in spherical geometry*, J Comput Phys **124**, 93/114, (1996).

7) W. von Bloh, A. Block, and H. J. Schellnhuber: *Self-stabilization of the biosphere under global change: a tutorial geophysiological approach*, Tellus **49(B)**, 249/262, (1997).

8) A. J. Watson and J. E. Lovelock: *Biological homeostasis of the global environment: the parable of Daisyworld*, Tellus **35(B)**, 284/289, (1983).

9) D. L. Williamson: *The evolution of dynamical cores for global atmospheric models*, J Meteor Soc Japan **85(B)**, 241/269, (2007).

10) 八木 厚志: 放物型発展方程式とその応用，岩波数学叢書，岩波書店，(2011).