

スキーマ処理を用いた文法進化の改良について

杉浦 秀幸¹ 水野 貴央¹ 丸田 峻也¹ 北 栄輔^{1,2,a)}

概要：文法進化は，設計目的を満たす関数やプログラムを生成することを目的とした進化的計算法の一つである．本研究では，文法進化の収束性能の改善のために，スキーマ処理の応用について述べる．関数同定問題において有効性を検討する．

キーワード：文法進化，スキーマ処理，関数同定問題

Improvement of Grammatical Evolution by Schemata Operation

Abstract: Grammatical Evolution (GE) is one of the evolutionary computations which is designed for finding function, program or program fragment satisfying the design objective. A simple improvement idea of Grammatical Evolution is presented in this study. Instead of the simple Genetic Algorithm concept, stochastic schemata exploiter concept is employed for the Grammatical Evolution. Effectiveness of the present algorithm is discussed in the symbolic regression problem.

Keywords: Grammatical Evolution, Schemata Exploiter, Symbolic Regression Problem.

1. 緒論

進化的計算法は動物の進化のプロセスなどに着想を得た計算アルゴリズムで，関数の解探索，機械学習等に広く用いられている．代表的な方法として，遺伝的アルゴリズム (Genetic Algorithm : GA)[1], [2] や遺伝的プログラミング (Genetic Programming : GP)[3] などがある．

GA とは 1975 年に Holland[1] によって提案された進化的計算法である．最初に解候補を遺伝子とした個体集団を用意する．個体の評価値である適合度に応じて親個体を選択し，交叉，突然変異と言った処理により親個体から子個体を生成する．最適化問題においては，この操作を繰り返すことで，最適解を探索しようとする．GP とは 1990 年に Koza[3] によって提案された進化的計算法である．GP の目的は GA とは異なり，与えられた数値データや目的を満たすような関数やプログラムを生成することを目的としている．GP では解探索過程で遺伝的操作により新たに生成された個体の遺伝子型 (2 進数など) から，文法的に正し

い表現型 (関数やプログラム) を必ず生成できる保証はないため，致死個体が多くなる．この問題を解決するために，文法進化 (Grammatical Evolution : GE)[4], [5] が提案されている．GE では，予め遺伝子型から表現型への変換方法をバックス・ナウア記法 (Backus Naur Form : BNF) 等を用いて定義しておく．この文法を用いることで，任意の遺伝子型から必ず文法的に正しい表現型を生成するようにして GP の問題を解決している．また，GA と同様な個体遺伝子表現を用いることができるので，GA で用いられる遺伝操作を利用できる．

本研究の目的は，GE においては GA と同様の遺伝操作を利用できる点に着目し，GE で用いられる単純 GA の探索プロセスの代わりに，確率的スキーマ貪欲法 (Stochastic Schemata Exploiter : SSE)[6] の処理を組み合わせることである．この手法を Grammatical Evolution with Stochastic Schemata Exploiter (GE-SSE) と名付けることにする．SSE は相澤らが提案した進化的計算手法のひとつである．SSE は，GA と同様に 2 進数で定義された個体集団を用いて探索を行うが，優良個体間に共通に存在する 2 進数の並びである共通スキーマを元に次世代の個体を生成する．その結果，SSE は解空間において適合度の高い個体の近傍を集中的に探索するので，単純 GA 等よりも収束速度が速

¹ 名古屋大学
Nagoya University, Nagoya 464-8601, Japan

² 神戸大学
Kobe University, Kobe 657-0013, Japan

^{a)} kita@is.nagoya-u.ac.jp

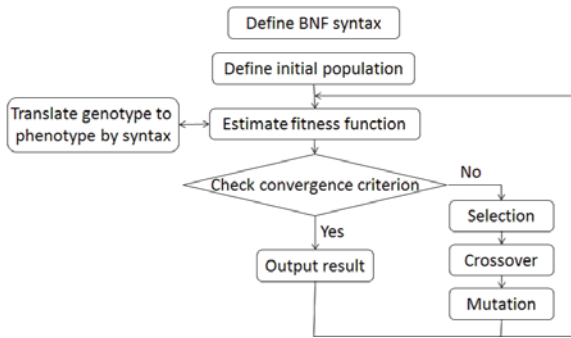


図 1 文法進化のフローチャート

Fig. 1 Flowchart of Grammatical Evolution

く、解探索に必要な制御パラメータが少ないという特徴がある [6].

本論文の構成は次のようになっている。第 2 節では、文法進化 (GE) と確率的スキーマ貪欲法 (SSE), それらに基づく GE-SSE について述べる。第 3 節では、GE-SSE を関数同定問題に適用する。最後に、第 4 節は本稿全体のまとめと今後の課題である。

2. アルゴリズム

2.1 文法進化

2.1.1 アルゴリズム

文法進化 (GE) のフローチャートを図 1 に示し、アルゴリズムを以下に述べる。

- (1) パッカス・ナウア記法 (BNF) を用いて遺伝子型を表現型に変換する文法を定義する。
- (2) 各個体をランダムに生成した 2 進数列で定義する。
- (3) 以下のようにして各個体の遺伝子型を表現型に変換する。
 - (a) 遺伝子を 2 進数列から n -bit 毎に 10 進数へ基数変換する。
 - (b) 生成の最初で選択される非終端記号を開始記号とする。
 - (c) 生成している構文中で最も左にある非終端記号を α , α に対応する遷移規則数を n_α とする。
 - (d) 10 進数列の値 β を n_α で割った剰余 γ を求める。ただし β として、10 進数列の左から順に値を利用していくものとする。
 - (e) α に対応する遷移規則の中で γ 番目の遷移規則によって α を置換する。
 - (f) すべての非終端記号が終端記号に変換されるまでこれを繰り返す。
- (4) 生成された構文を用いて適合度を計算する。
- (5) 設定した終了条件を満たせば終了。満たさなければ次へ進む。
- (6) 個体集団に選択、交叉及び突然変異等の遺伝操作を適用し、新たな個体集団を生成する。

表 1 BNF 文法の例

Table 1 Example of BNF Syntax

(A)	$\langle \text{expr} \rangle ::= \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$	(A0)
	$\quad \quad \quad \langle \text{var} \rangle$	(A1)
(B)	$\langle \text{op} \rangle ::= +$	(B0)
	$\quad \quad \quad -$	(B1)
	$\quad \quad \quad *$	(B2)
	$\quad \quad \quad /$	(B3)
(C)	$\langle \text{var} \rangle ::= X$	(C0)
	$\quad \quad \quad Y$	(C1)
	$\quad \quad \quad Z$	(C2)

(7) ステップ (3) へ戻る。

2.1.2 遺伝子型から表現型への変換

遺伝子型から表現型への変換プロセスを例を用いて説明する。ここでは例として表 1 のような BNF を定義し、定義した文法を用いて構文の生成を行う。開始記号は $\langle \text{expr} \rangle$ とする。表 1 において左辺に現れる各非終端記号に対して右辺に現れる $|$ で区切られた終端記号もしくは非終端記号が遷移規則として対応することを表す。例えば $\langle \text{var} \rangle$ に対応する遷移規則は $\langle \text{var} \rangle$ を X , Y 又は Z に置き換える (C0), (C1), (C2) の 3 通りである。

- (1) 遺伝子 010001111101101110 が与えられたとする。ここでは 3-bit 毎にとして基数変換を行う。変換の結果与えられた遺伝子は次のように変換される。

010 001 111 101 101 110 基数変換前
↓

2 1 7 5 5 6 基数変換後

- (2) 10 進数列の最初の値は 2 である。BNF の開始記号は $\langle \text{expr} \rangle$ である。表 1 より、 $\langle \text{expr} \rangle$ から遷移することのできる遷移規則数は 2 となる。よって遺伝子値を遷移規則数で割った剰余は 0 なので (A0) が選択される。この選択された遷移規則を用いて開始記号 $\langle \text{expr} \rangle$ を置き換える。

$\langle \text{expr} \rangle$ 置換前
↓

$\langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$ 置換後

- (3) 最も左にある非終端記号は $\langle \text{expr} \rangle$ である。2 番目の遺伝子値は 1, 遷移規則の遷移数が 2 通りなので、剰余は 1 となり (A1) が選択される。そこで $\langle \text{expr} \rangle$ をこの遷移規則で置き換える。

$\langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$ 置換前
↓

$\langle \text{var} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$ 置換後

- (4) 以下同様に遺伝子値と遷移規則数の剰余を用いて変換に利用する遷移規則を選択し、すべての非終端記号が終端記号に変換されるまでこの操作を繰り返す。この例では変換の結果 $Y-X$ という数式が生成される (表 2)。

表 2 生成される構文の例
Table 2 Generated symbol list

対象	遺伝子値	構文
<expr>	2	<expr><op><expr>
<expr>	1	<var> <op><expr>
<var>	7	Y<op><expr>
<op>	5	Y-<expr>
<expr>	5	Y-<var>
<var>	6	Y-X

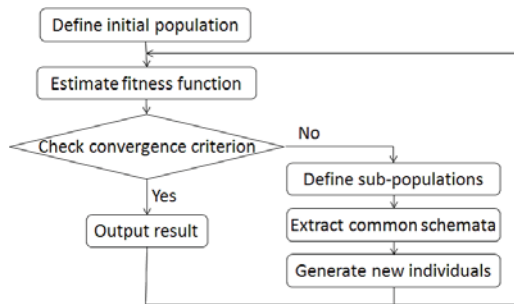


図 2 SSE のフローチャート
Fig. 2 Flowchart of SSE

2.2 確率的スキーマ貪欲法

確率的スキーマ貪欲法 (Stochastic Schemata Exploiter : SSE) は、相澤らにより提案された [6], [7]. スキーマとは 2 進数で定義された遺伝子において、特定の 2 進数の並びのことである。SSE では優良個体の共通スキーマを基にして次世代の解を生成するので、良いスキーマを母集団に急速に広めることで、局所的探索を改善する。

2.2.1 アルゴリズム

図 2 に示すフローチャートにしたがい、SSE のアルゴリズムを以下に述べる。

- (1) 初期個体 M 個をランダムに生成し、初期集団を定義する。
- (2) 各個体の適合度を評価する。
- (3) 収束条件が満足されれば結果を出力して終了する。そうでなければ次へ進む。
- (4) 個体を適合度の降順に並べて $c_1, c_2, \dots, c_M$ とし、最上位の個体から半順序関係に従って個体部分集合を生成する。
- (5) リストに格納されている上位 M 個の個体部分集合からそれぞれ共通スキーマを抽出する。
- (6) M 個のスキーマからランダムに個体を生成し、突然変異操作を適用する。
- (7) 生成された M 個の子個体により次世代の母集団を生成し、ステップ (2) へ進む。

2.2.2 スキーマと個体部分集合の評価値について

スキーマ (schema) とは解構造 (0/1 のバイナリ列) における特定の 0/1 の並びを指し、 $\{0, 1, *\}$ の 3 種類の記号の並びにより表現される。以下、任意のスキーマを H 、その

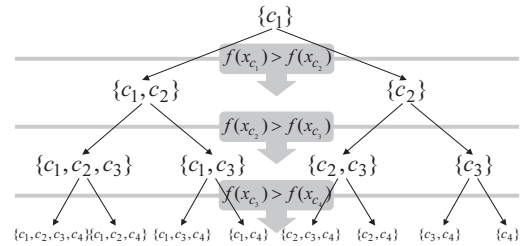


図 3 個体部分集合の派生
Fig. 3 Subpopulations

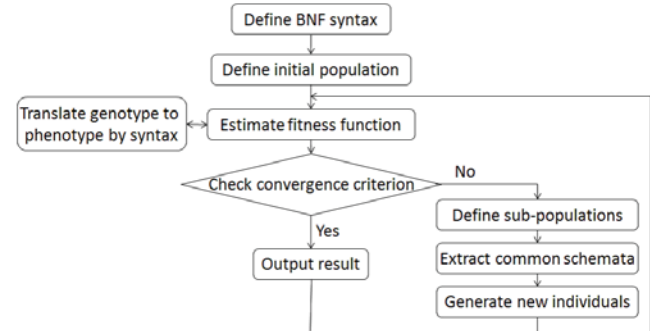


図 4 GE-SSE のフローチャート
Fig. 4 Flowchart of GE-SSE

スキーマを含む解の集合を $s(H)$ 、解の総数を $|s(H)|$ と表す。スキーマ H の評価値は、スキーマ H を含むすべての個体の評価値 (適合度値) の平均値として表され、以下の式で定義される。

$$f(H) = \frac{1}{|s(H)|} \sum_{x \in s(H)} f(x) \quad (1)$$

2.2.3 個体部分集合の生成の手順

母集団 P_t において、 M 個の個体をその適合度の降順に並べたインデックス (番号付け) を $c_1, c_2, \dots, c_M$ と置く。 P_t の任意の個体部分集合 $S (\neq \phi)$ について、その中に含まれる最大のインデックスを $L(S)$ で表す。 $(S - c_k)$ は、集合 S から要素 c_k を除いた集合とする。また、和集合を $\cup$ で表す。 $L(S) < M$ のとき、 P_t の個体部分集合の間に以下の半順序関係が存在する。

- S の平均評価値は $S \cup c_{(L(S)+1)}$ の平均評価値よりも良い。
- S の平均評価値は $(S - c_{L(S)}) \cup c_{(L(S)+1)}$ の平均評価値よりも良い。

SSE では、以上の半順序関係を用いることで、 $S = \{c_1\}$ から順に個体部分集合を生成していく (図 3)。

2.3 GE-SSE

Grammatical Evolution with Stochastic Schemata Exploiter (GE-SSE) のフローチャートを図 4 に示し、各個体の適合度評価における遺伝子型から表現型への変換アルゴリズムを以下に述べる。

- (1) バッカス・ナウア記法 (BNF) を用いて遺伝子型を表

表 3 BNF 文法 (関数同定問題)

Table 3 BNF syntax (Symbolic regression problem)

(A)	$\langle \text{expr} \rangle ::= \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$ $\langle \text{var} \rangle$	(A0) (A1)
(B)	$\langle \text{op} \rangle ::= +$ $-$ $*$ $/$	(B0) (B1) (B2) (B3)
(C)	$\langle \text{var} \rangle ::= x$ $\langle \text{num} \rangle$	(C0) (C1)
(D)	$\langle \text{num} \rangle ::= 1$ 2 3 4 5 6 7 8 9	(D0) (D1) (D2) (D3) (D4) (D5) (D6) (D7) (D8)

現型に変換する文法を定義する。

- (2) 各個体をランダムに生成した 2 進数列で定義する。
- (3) GE のアルゴリズムに従い各個体の遺伝子型を表現型に変換する。
- (4) 生成された構文を用いて適合度を計算する。
- (5) 設定した終了条件を満たせば終了。満たさなければ次へ進む。
- (6) SSE のアルゴリズムに従い集団を更新する。
- (7) ステップ (3) へ戻る。

GP では、ブロートを防ぐために枝狩りという操作を導入する。本研究では GE に同様な目的の操作を加えることにし、生成される構文の長さが最大構文長である $MaxN$ 以下となるようにする。

3. 解析例

3.1 関数同定問題

関数同定問題を扱う。関数同定問題とは、 n 個の入出力データ $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ が与えられたとき、真の関数 f の近似関数 $\bar{f}$ を求めることである。但し、 $y_i = f(x_i)$ である。

本実験では、真の関数 f として次式を用いる。

$$f(x) = x^4 + x^3 + x^2 + x \quad (2)$$

式 (2) において、 x の値を -10.0 から 10.0 まで 0.1 刻みで変化させた計 201 個のサンプル点を実験に用いる。

表 3 のような BNF 文法を定義する。開始記号は $\langle \text{expr} \rangle$ である。適合度は、サンプル点における真の関数と GE, GP によって生成した関数との平均二乗誤差を用いる。平均二乗誤差は式 (3) で与えられる。

表 4 関数同定問題のパラメータ設定

Table 4 Parameters for symbolic regression problem

Max. generation	1000
Number of trials	100
Population size	300
Chromosome length	800
Number of elite	3
Bit length for radix conversion	8bit
Max length of sentences	24

$$E = \sqrt{\frac{1}{201} \sum_{i=1}^{201} (f(x_i) - \bar{f}(x_i))^2} \quad (3)$$

したがって、適合度は 0 に近づくほど良いこととなる。

進化的処理において、選択方法は個体 x_i についての適合度を $f(x_i)$ 、個体数を N としたときに個体 x の選択確率 P_x を式 (4) としたルーレット選択を採用する。

$$P_x = 1 - \frac{f(x)}{\sum_{i=1}^N f(x_i)} \quad (4)$$

交叉方法には一点交叉を採用する。また、エリート保存戦略を用いる。GE については交叉率を 0.9, 0.8, 0.7, 0.6, 0.5 とし、突然変異率を 0.5, 0.4, 0.3, 0.2, 0.1, 0.075, 0.05, 0.025, 0.01 と設定してそれぞれ実験する。GE-SSE については突然変異率を 0.5, 0.4, 0.3, 0.2, 0.1, 0.075, 0.05, 0.025, 0.01 と設定してそれぞれ実験する。そのほかのパラメータについては表 4 に示す。

まず、GE についての交叉率、突然変異率の影響を比較する。図 5 は各突然変異率における最終世代における最良個体の平均適合度を各交叉率ごとに示したグラフである。縦軸は平均適合度を示し、横軸は各交叉率を並べており、各実線はそれぞれの突然変異率のものである。図 5 からどの交叉率においても最適な突然変異率が異なっていることがわかる。また、図 6 は各交叉率における最終世代における最良個体の平均適合度を各突然変異率ごとに示したグラフである。縦軸は平均適合度を示し、横軸は各突然変異率である。突然変異率が 0.01 のとき、どの交叉率においても悪い適合度となっている。これは突然変異率が極端に小さな値の場合、初期個体群の遺伝子配列に大きく依存してしまうためだと考えられる。最も良い適合度を示した交叉率と突然変異率の組合せは、交叉率が 0.6、突然変異率が 0.075 であった。

次に、GE-SSE の突然変異率の影響について考察する。図 7 は各突然変異率における最終世代における最良個体の平均適合度を示したグラフである。縦軸は平均適合度、横軸は突然変異率となっている。最も良い適合度を示した実験は突然変異率が 0.3 のものであった。

GE の交叉率が 0.6、突然変異率が 0.075 と設定した実験の最良個体の平均適合度の収束曲線と GE-SSE の突然変異率を 0.3 と設定した実験の最良個体の平均適合度の収束曲

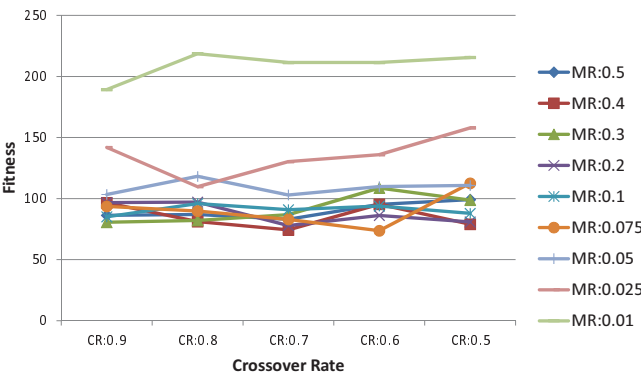


図 5 各交叉率の最終世代における最良個体の平均適合度

Fig. 5 Effect of crossover rate to average fitness of best individuals

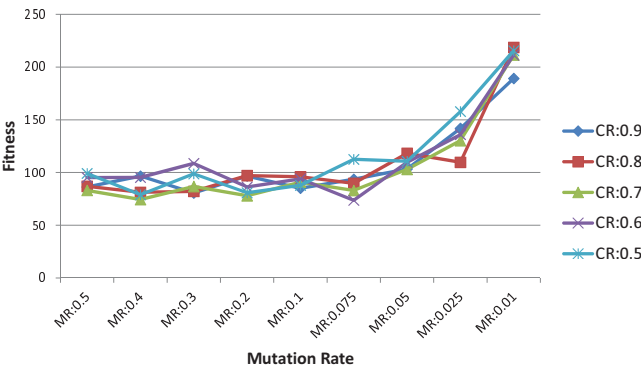


図 6 各突然変異率の最終世代における最良個体の平均適合度

Fig. 6 Effect of mutation rate for average fitness of best individuals

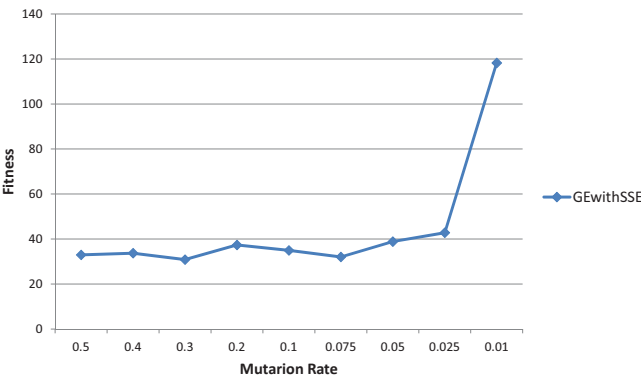


図 7 各突然変異率の最終世代における最良個体の平均適合度

Fig. 7 Effect of crossover rate for average fitness of best individuals

線を図 8 に示す。この図において、横軸は世代数を縦軸は最良個体の平均適応度を示す。点線が通常の GE による結果を示し、実線が GE-SSE による結果を示している。これより、GE-SSE は GE と比べ初期収束速度が速いことがわかる。

3.2 日経平均株価予測問題

日経平均株価（日経平均）は日本の株式市場を代表する株価指数であり、東京証券取引所第一部に上場する約 1700

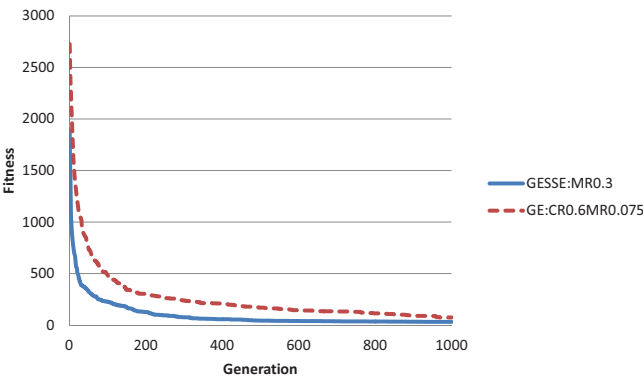


図 8 最良個体の平均適応度の収束

Fig. 8 Convergence of average fitness of best individuals

表 5 日経平均株価予測問題における BNF 文法

Table 5 BNF syntax for NIKKEI average forecast problem

(A)	$\langle \text{expr} \rangle ::= \langle \text{expr} \rangle \langle \text{expr} \rangle \langle \text{op} \rangle$	(A0)
	$\quad \mid \langle \text{var} \rangle$	(A1)
(B)	$\langle \text{var} \rangle ::= \langle \text{stock} \rangle$	(B0)
	$\quad \mid \langle \text{num} \rangle$	(B1)
(C)	$\langle \text{op} \rangle ::= +$	(C0)
	$\quad \mid -$	(C1)
	$\quad \mid *$	(C2)
	$\quad \mid /$	(C3)
(D)	$\langle \text{stock} \rangle ::= y_{t-1}$	(D0)
	$\quad \mid y_{t-2}$	(D1)
	$\quad \mid y_{t-3}$	(D2)
	$\quad \mid y_{t-4}$	(D3)
	$\quad \mid y_{t-5}$	(D4)
(E)	$\langle \text{num} \rangle ::= 1$	(E0)
	$\quad \mid 2$	(E1)
	$\quad \mid 3$	(E2)
	$\quad \mid 4$	(E3)
	$\quad \mid 5$	(E4)
	$\quad \mid 6$	(E5)
	$\quad \mid 7$	(E6)
	$\quad \mid 8$	(E7)
	$\quad \mid 9$	(E8)

銘柄のうち、代表的な 225 銘柄を対象として算出する。学習データとして 2010 年 10 月 1 日から 2011 年 9 月 30 日までの株価のデータ（訓練データ）をとり、訓練データ内での近似関数の当てはめの成績が良くなるように進化させる。そして、訓練データ内で最も適合度の良い近似関数を 2011 年 10 月 3 日から 2011 年 11 月 30 日までの株価のデータ（テストデータ）と比較して評価を行う。

日経平均株価の予測式を GE-SSE で生成するため、表 5 のような文法を定義する。開始記号は $\langle \text{expr} \rangle$ である。

適合度は、訓練データ内における実際の値と GE-SSE によって生成した近似関数から得られる予測値との平均二乗誤差を用いる。平均二乗誤差は式 (5) で与えられる。

表 6 日経平均株価予測問題のパラメータ設定

Table 6 Parameters for NIKKEI average forecast problem

Max. generation	1000
Number of trials	100
Population size	300
Chromosome length	800
Number of elite	3
Bit size for radix conversion	8bit
Maximum length of sentences	24

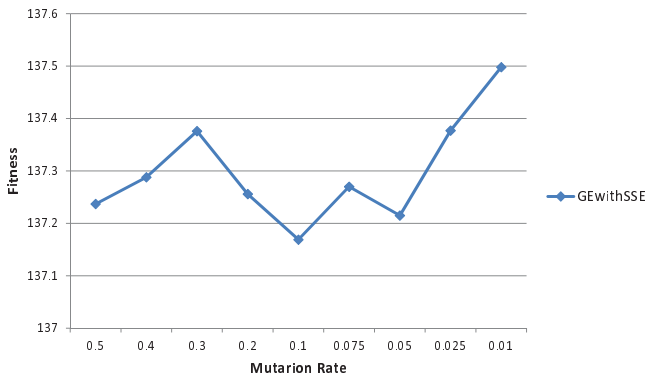


図 9 各突然変異率の最終世代における最良個体の平均適合度

Fig. 9 Effect of mutation rate for average fitness of best individuals

$$E = \sqrt{\frac{1}{N} \sum_{t=1}^N (y_t - \bar{y}_t)^2} \quad (5)$$

ここで、 N は株価を予測する日数、 y_t は t 日における真の株価、 $\bar{y}_t$ は GE が生成した近似関数から計算した予測株価である。この値をそのまま適合度として扱うので、適合度は 0 に近いほど良いことになる。

表 6 のようにパラメータを設定し、突然変異率を 0.5, 0.4, 0.3, 0.2, 0.1, 0.075, 0.05, 0.025, 0.01 と設定し比較する。

図 9 は各突然変異率における最終世代における最良個体の平均適合度を示したグラフである。縦軸は平均適合度、横軸は突然変異率となっている。最も良い適合度を示した実験は突然変異率を 0.1 と設定したものであった。各突然変異率における最良個体の平均適合度の収束曲線を図 10 に示す。ここで、横軸は世代数を縦軸は平均適応度を示す。

100 回の試行のうち最も良い適合度は 134.95 であり、その最良個体から得られた近似曲線を次式に示す。

$$\bar{y}_t = (y_{t-1} - 5) + \frac{(y_{t-3} - y_{t-5}) + 10}{y_{t-2}} + \frac{y_{t-5} - y_{t-4}}{5} \quad (6)$$

予測期間における日経平均と予測株価との平均二乗誤差は 112.284 となった。

4. 結論

本研究では、文法進化における初期収束速度の改善のた

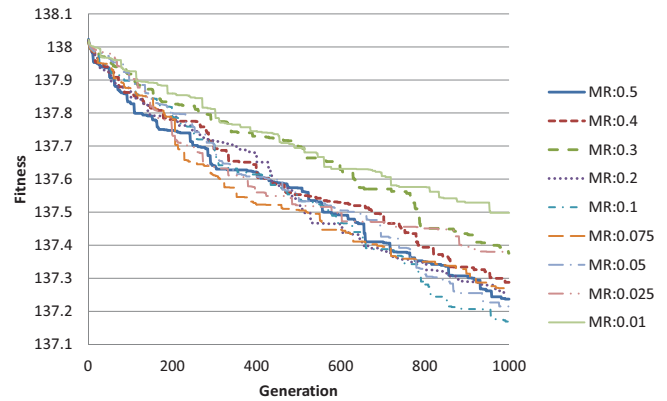


図 10 最良個体の平均適応度の収束

Fig. 10 Convergence of average fitness of best individuals

め GE の遺伝操作に SSE を適用した GE-SSE を提案した。GE と GE-SSE に対し、関数同定問題によって比較実験を行い、GE-SSE の初期収束速度が GE と比べ高いことを確認した。SSE は評価値の良いスキーマを基にして子個体を生成し、解探索を行なっていくため、各世代での優良個体に似た解構造を集中的に探索する傾向がある。そのため、GE と比べ初期収束速度が速くなったと考えられる。

今後の課題と展望としては以下が挙げられる。まず、GE と GE-SSE との比較のため多種のテスト問題を扱うことで遺伝操作の違いによる探索性能の特徴の確認が挙げられる。

参考文献

- [1] Holland, J. H.: *Adaptation in Natural and Artificial Systems*, The University of Michigan Press, 1 edition (1975).
- [2] Goldberg, D. E.: *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison Wesley, 1 edition (1989).
- [3] Koza, J. R.(ed.): *Genetic Programming II*, The MIT Press (1994).
- [4] Ryan, C., Collins, J. J. and O'Neill, M.: Grammatical Evolution: Evolving Programs for an Arbitrary Language, *Proceedings of 1st European Workshop on Genetic Programming*, Springer-Verlag, pp. 83–95 (1998).
- [5] C.Ryan and M.O'Neill: *Grammatical Evolution: Evolutionary Automatic Programming in an Arbitrary Language*, Springer-Verlag (2003).
- [6] Aizawa, N. A.: Evolving SSE: A Stochastic Schemata Exploiter, *Proc. 1st IEE Conf. Evol. Comp.*, IEEE, pp. 525–529 (1994).
- [7] 相澤彰子: スキーマ貪欲な遺伝的探索アルゴリズム, 遺伝的アルゴリズム 2 (北野宏明 編), 産業図書, chapter 1, pp. 3–32 (1995).