

文法進化の株価予測問題への適用について

黒田 卓也 † 岩澤 博人 † 北 栄 輔†

進化的計算手法の一つに文法進化 (GE) がある。Original GE の問題点を改善するために、3 つの異なるスキームを用いた改良型 GE が提案されている。本研究では、解析例において日経平均株価の予測問題に Original GE と改良型 GE を適用する。解析結果より、スキーム 1+2 またはスキーム 1+2+3 を用いることで、改良型 GE は Original GE よりも収束速度が改善することがわかった。

Application of Grammatical Evolution to Stock Price Prediction

TAKUYA KURODA ,† HIROTO IWASAWA † and EISUKE KITA†

Grammatical Evolution (GE) is one of evolutionary algorithms. Three schemes have been presented for improving the search performance of original GE. In this paper, the original GE and the advanced GE are compared on the prediction problem of NIKKEI stock average. The results show that the advanced GE with scheme 1+2 or 1+2+3 overtakes the original GE.

1. 緒 論

進化的計算手法とは、生物の進化のプロセスを工学的システムの最適化のために利用したアルゴリズムの総称であり、代表的なものとして遺伝的アルゴリズム (GA)¹⁾ や遺伝的プログラミング (GP)²⁾ などがある。

GA はダーウィンの進化論をモチーフに解を個体として扱い、個体が環境に適合する度合いを最適化問題の目的関数と考える。選択・交叉・突然変異などの操作を繰り返すことで探索を行う。GP は遺伝子として木構造を用いることで、GA では扱うことの難しい数式やプログラムなどを扱う。

GP の改良として、一次元配列の遺伝子を利用する GP が提案されている³⁾。この方法では、一次元配列の各遺伝子座の値を、あらかじめ設定した対応表に基づき木構造の各ノードに相当する記号に置き換えることで、木構造を生成する。この方法では、遺伝的操作により新たに生成された数式やプログラムが必ずしも文法的に正しい構造である保証はなく、しばしば評価不可能な構造が生じる。Whigham は文法型 GP という手法を提案している⁴⁾。個体の生成にあらかじめ定義しておいた文法を用いることで正しい構造を持つ個体の生成が可能となる手法である。しかし文法型 GP で用いる遺伝子は GP と同様の木構造を用いるため、

様々なパターンの遺伝的操作を必要とするという GP の問題は解決されていない。

Grammatical Evolution (GE)^{5),6)} は C.Ryan らによって提案された GP の一手法である。この手法では Banzhaf の手法に Whigham の手法を組み合わせることで、Banzhaf の手法の問題を解決している。GE の実問題への応用として、企業倒産の予測⁷⁾ などが行われている。また、探索性能の検討として、C.Ryan らによって交叉法についての研究⁸⁾ や、岩澤らによって構文生成の改良についての研究⁹⁾ が行われている。

本研究では日経平均株価の予測問題に対して C.Ryan らが提案した Original GE と岩澤らが提案した改良型 GE を適用し、両者の性能を比較することで改良型 GE のこの問題に対する有効性を検討する。

2. Grammatical Evolution

2.1 Original GE のアルゴリズム

GE のアルゴリズムを以下に示す。

- (1) バッカス・ナウア記法 (BNF) を用いて扱う問題に適した文法を定義する。
- (2) 初期個体をランダムに生成した 2 進数列で定義する。
- (3) 各個体の遺伝子型を表現型に変換する。(詳細は以下で述べる。)
- (4) 生成された構文を元に評価関数を用いて適合度を計算する。
- (5) 個体集団に選択、交叉及び突然変異等の遺伝的

† Nagoya University, Graduate School of Information Sciences

操作を適用し、新たな個体集団を生成する。

- (6) 設定した終了条件を満たせば終了。満たさなければ3へ戻る。

GPではプロットを防ぐために枝狩りという操作を導入する。GEにおける遺伝子長はGAと同様に固定長であるが、生成される構文は可変長であるため、遺伝子型から表現型への変換が終了しない場合がある。そこで、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。

遺伝子型から表現型への変換は以下のようにして行われる。

- (1) 遺伝子を n -bit 毎に2進数列から10進数列へ基数変換する。
- (2) 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。ここで、生成の最初で選択される非終端記号を開始記号とする。
- (3) 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。
- (4) α に対応する遷移規則の中で γ 番目の遷移規則を選択し、 α をこの遷移規則にしたがって置き換える。
- (5) 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。

2.2 Original GEにおける問題点

Original GEには2つの問題が存在する。1つは遺伝子値と遷移規則数の剰余を用いて行われる文法選択方法に関する問題である。もう1つは遷移規則を選択する際の各遷移規則の選択確率に関する問題である。

2.2.1 文法選択方法に関する問題点

Original GEの文法選択方法では遺伝子値と遷移規則数の剰余を用いる。遺伝子操作により、ある遺伝子値が1変化するとこの値から算出される剰余の値も同時に1変化する。このため各個体から生成される構文は遺伝子値の変化に対して過敏であり、適合度の高いスキーマが生成されてもすぐに破壊されてしまう可能性が高い。

2.2.2 遷移規則選択確率に関する問題点

Original GEでは遺伝子値を遷移規則数で割った剰余によって文法を選択するため、各遷移規則の選択確率は等しい。しかし、問題によってはどちらか一方のより有効と思われる方が多く選択されるよう、確率を変動させて探索を行う方が収束が早くなる可能性がある。

遷移規則を生成される構文に与える影響で分類すると、再帰規則と終端規則の2種類に分けることができる。再帰規則とは置き換え後の記号に置き換え前の記号を含んでいる遷移規則のことであり、再帰規則が繰り返して選択されることで、生成される構文の長さが長くなる。終端規則とは置き換え後の記号に非終端記号が含まれていない遷移規則のことであり、構文内容に大きく影響を与える。

再起規則と終端規則は役割が異なるので、異なる選択確率を与える必要がある。

2.3 改良型 GE

2.3.1 文法選択方法の改良

文法選択に関する問題を解決するため、遺伝子値の取り得る範囲を遷移規則数で分割し、分割した値の範囲毎に各遷移規則に対応させる手法が提案されている⁹⁾。この手法では遺伝子値が1変化しても必ずしも選択される遷移規則が変化するとは限らず、Original GEと比較して優良なスキーマが破壊されにくくなるといった利点がある。

2.3.2 再帰規則の選択確率の改良

再帰規則の選択確率の問題を解決するため、生成中の構文の長さに応じて再帰規則の選択確率を変化させる手法が提案されている⁹⁾。

この手法では、再帰規則の選択確率 RP は式(1)を用いて求める。

$$RP = 1 - \text{Now}N / \text{Max}N \quad (1)$$

ここで、 $\text{Now}N$ は現在生成中の構文の長さを、 $\text{Max}N$ は最大構文長を示す。

2.3.3 終端規則の選択確率の改良

終端規則の選択確率の問題を解決するため、各終端記号の出現頻度から選択確率を求める手法が提案されている⁹⁾。

まず各個体の遺伝子から定義した文法を用いて生成された全構文に使用されている各終端記号の個数を数え、それぞれ An 、 Bn 、 Cn とする。次に、終端記号 A 、 B 、 C の選択確率 TPa 、 TPb 、 TPc を次式により求める。

$$\left. \begin{aligned} TPa &= An / (An + Bn + Cn) \\ TPb &= Bn / (An + Bn + Cn) \\ TPc &= Cn / (An + Bn + Cn) \end{aligned} \right\} \quad (2)$$

3. 解析例

3.1 問題設定

解析例として日経平均株価の予測問題を扱う。学習データとして1984年1月4日から2007年12月28日までの株価のデータ(訓練データ)を用い、訓練デー

表 1 日経平均株価予測問題における文法

(A)	$\langle \text{expr} \rangle ::= \langle \text{expr} \rangle \langle \text{expr} \rangle \langle \text{op} \rangle$ $\langle \text{var} \rangle$	(A0) (A1)
(B)	$\langle \text{var} \rangle ::= \langle \text{stock} \rangle$ $\langle \text{num} \rangle$	(B0) (B1)
(C)	$\langle \text{op} \rangle ::= +$ $-$ $*$ $/$	(C0) (C1) (C2) (C3)
(D)	$\langle \text{stock} \rangle ::= y_{t-1}$ y_{t-2} y_{t-3} y_{t-4} y_{t-5}	(D0) (D1) (D2) (D3) (D4)
(E)	$\langle \text{num} \rangle ::= 0$ 1 2 3 4 5 6 7 8 9	(E0) (E1) (E2) (E3) (E4) (E5) (E6) (E7) (E8) (E9)

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 2008 年 1 月 4 日から 2008 年 10 月 6 日までの株価のデータ（テストデータ）と比較して評価する。

表 1 のように文法を定義する。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 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。つまり、

$$E = \sqrt{\frac{1}{N} \sum_{t=1}^N (y_t - \bar{y}_t)^2} \quad (3)$$

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。

実験パラメータは次のように与える。世代数は 500、集団の個体数は 100、個体長は 100bit である。遺伝的操作として、トーナメントサイズ 5 のトーナメント選択、交叉率 0.5 の一点交叉、エリート保存戦略を用い、突然変異率は 0.1 である。2 進数から 10 進数へ変換する際の bit 数を 4bit、最大構文長 $\text{MaxN}=100$ とする。実験は異なる乱数を用いて 50 回の試行を行い、適合度の平均値によって比較を行う。スキーム 3 における終端記号の選択確率を変化させる間隔は、50 世代ごととする。

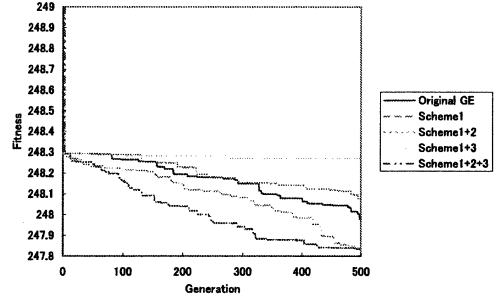


図 1 日経平均株価予測問題における各スキームの収束速度比較

3.2 結果と考察

3.2.1 スキーム 1 の実験結果と考察

Original GE の結果とスキーム 1 の結果を比較すると、スキーム 1 を適用することで収束速度が悪化していることが分かる。スキーム 1 は個体の遺伝子値の変化による構文の変化を抑制する効果があるが、本実験においては構文の変化を抑制したことで解の探索範囲が狭くなってしまい最適解に到達しにくくなったのではないかと考えられる。本実験は日経平均株価を過去の株価の多項式として予測するものであるが、このような問題においては精度の良い解を生成するためにより複雑な構文を作成する必要があると考えられる。よって、本問題において探索を有効に進めるには個体の多様性を維持しつつ世代を進める必要があると考えられる。そのため、遺伝的操作による構文の変化を抑制する働きを持つスキーム 1 は逆効果となったと考えられる。

3.2.2 スキーム 1+2 の実験結果と考察

スキーム 1 に加えてスキーム 2 を適用すると、スキーム 1 単独の結果と比較して収束速度が向上していることが分かる。本実験で用いた文法（表 1）における再帰規則は (A) の $\langle \text{expr} \rangle$ のみである。本実験において、世代初期の優良個体として多く生成される個体は y_{t-i} という構文である。これは過去の株価の値をそのまま返す、という予測である。過去の株価の値をそのまま返してもある程度の適合度は保証されるため、このような個体は多く生き残るが、過去の株価の値に何らかの数学的操作を加えたほうが適合度は良くなると考えられる。そのためには y_{t-i} という構文よりも長い構文を作成する必要があるため、生成中の構文が短い時には長くする働きを持つスキーム 2 が有効に働いたのではないかと考えられる。

3.2.3 スキーム 1+3、スキーム 1+2+3 の実験結果と考察

スキーム 1、スキーム 1+2 に加えてスキーム 3 を

適用すると、スキーム 1 に加えてスキーム 3 を適用した場合はスキーム 1 単独の結果と比較して収束速度が悪化しているのに対し、スキーム 1+2 に加えてスキーム 3 を適用した場合はスキーム 1+2 の結果と比較して収束速度が向上していることが分かる。本実験で用いた文法における終端規則は (C) の $\langle op \rangle$, (D) の $\langle stock \rangle$, (E) の $\langle num \rangle$ である。

スキーム 1 に加えてスキーム 3 を適用した場合、再帰規則とそうでない遷移規則は等確率で選択されることから、長い構文が生成されにくい環境下で終端記号の選択確率を変化させることになる。日経平均株価を過去の株価の値から予測する場合、過去の株価の値をそのまま将来の株価とするよりも、過去の株価の値に何らかの数学的操作を加えたほうが適合度は良くなると考えられるため、長い構文の個体の適合度は短い個体の適合度と比較して良好であると考えられる。そのため、短い構文に含まれる終端記号からその選択確率を変化させても、適合度の良い個体に含まれる終端記号の選択確率を高くすることにはならなかったと考えられるさらに、スキーム 1 によって生成された構文を壊れにくくした上で終端記号の選択確率を変化させているため、局所解に収束してしまい解の改善を行うことができなくなったと考えられる。

しかし、スキーム 1+2 に加えてスキーム 3 を適用した場合、スキーム 2 の作用によって長い構文を持つ個体生成を実現しているため、終端記号の選択確率を初めて変化させる 50 世代目において、各個体が生成する構文に含まれる終端記号はより良い解の要素である可能性が高くなっていると考えられる。そのためスキーム 1, 2 に加えてスキーム 3 を適用することで収束速度が向上したと考えられる。

4. 結 論

GE は BNF により定義された文法を用いることで一次元配列の遺伝子から木構造を生成する進化的計算手法である。本研究では、岩澤らが提案した改良型 Grammatical Evolution を日経平均株価の予測問題に適用した。

改良方法には以下の 3 つがある。文法選択方法を改善することで良いスキームを保存して探索性能を改善するスキーム 1、再帰規則の選択確率を現在生成中の構文長に基づいて変化させるスキーム 2、終端規則の選択確率を各個体における出現頻度に基づいて変化させるスキーム 3 である。

数値実験より、スキーム 1 またはスキーム 1+3 を用いた改良型 GE は、探索性能において Original GE

より劣るが、スキーム 1+2 またはスキーム 1+2+3 を用いた改良型 GE は、Original GE よりもかなり良い探索性能を示すことがわかった。

参 考 文 献

- 1) J.H.Holland. **Adaptation in Natural and Artificial Systems**. The University of Michigan Press, 1975.
- 2) J.R.Koza. **Genetic Programming: On the Programming of Computers by Means of Natural Selection**. Cambridge, MA: MIT Press, 1992.
- 3) W.Banzhaf. Genotype-Phenotype-Mapping and Neutral Variation - A case study in Genetic Programming. **Parallel Problem Solving from Nature III**, pp.322-332, Springer-Verlag, 1994.
- 4) P.Whigham. Search Bias, Language Bias and Genetic Programming. In **Genetic Programming 1996: Proceedings of the First Annual Conference**, pp.230-237, MIT Press, 1996.
- 5) C.Ryan, J.J.Collins, and M.O'Neill. Grammatical Evolution: Evolving Programs for an Arbitrary Language. **Proc. of 1st European Workshop on Genetic Programming**, pp.83-95, Springer-Verlag, 1998.
- 6) C.Ryan and M.O'Neill. **Grammatical Evolution: Evolutionary Automatic Programming in an Arbitrary Language**. Springer. 2003.
- 7) A.Brabazon, M.O'Neill, R.matthews, and C.Ryan. Grammatical Evolution and Corporate Failure Prediction. **GECCO 2002: Proceedings of the Genetic and Evolutionary Computation Conference**, pp.1011-1018, 2002.
- 8) C.Ryan and M.O'Neill. Crossover in Grammatical Evolution: A Smooth Operator? **Lecture Notes in Computer Science, Vol. 1802, Proceedings of the European Conference on Genetic Programming**, pp.149-162, Springer-Verlag, 2000.
- 9) 岩澤博人, 北栄輔. Grammatical Evolution の文法選択方法の変更による性能改善. **情報処理学会研究報告**, Vol.2007, No.128, pp.101-104, 2007.