

複数の染色体表現方法を用いた Grammatical Evolution

水原 拓哉[†] 上野 敦志[†] 田窪 朋仁[†] 辰巳 昭治[†]

大阪市立大学大学院 工学研究科[†]

1 はじめに

遺伝的プログラミング (Genetic Programming ; GP) は遺伝的アルゴリズム (Genetic Algorithm ; GA) と異なり, 文法的に正しくない表現型を持つ個体が発生しないように遺伝的操作を適切に設定しなければならないという問題がある. この問題を解決するために提案された手法が Grammatical Evolution (GE) [1]である. GE では GA と同様に遺伝子型は 1 次元のビット列で表現される. これに遺伝的操作を行うことで準最適解を探索する. また, 遺伝子型は対象問題の解を正しく表現できる文法を用いて表現型に変換されるため, GP の様に遺伝的操作に制約を設ける必要がない. しかし, GE では前端に発現した有用な形質を次世代に引き継がせることができるが, 後端の形質は次世代に引き継がれず失われてしまう可能性が非常に高いという問題がある.

この問題に対し, 前端と後端に発現した有用な形質を次世代に引き継ぐことができる Bidirectional Representation[2]という手法が提案されている. ただし, Bidirectional Representation では発現した形質の前端と後端は子個体に引き継がれる可能性が高いが, 中央部の形質に関しては依然として引き継がれる可能性が低いままであると考えられる.

そこで, 本論文では新たに前端, 後端, および中央部 2 か所の計 4 か所付近の有用な形質を子個体に引き継がせることができる手法として Four-way Representation を提案する. 提案手法と Bidirectional Representation との比較実験を行うことで, 提案手法の性能を示す.

2 Grammatical Evolution

Grammatical Evolution (GE) とは, GA を拡張し任意のプログラミング言語のプログラムや数式といった構造的表現を扱うことができる手法である. GE の遺伝子型は GA と同様に可変長のビット列で表現される. 遺伝子型から表現型への変換には, コドン (遺伝子型を 8 ビットごとに区切り, それを 10 進数に変換した値) を用いる. あらかじめ適切に定義された対象問題の解を生成することができる文法とコドンを用いて表現型へ変換する方法が GE の特徴となっている. 図 1 に文法の例を示す. ここで N, T, P はそれぞれ非終端記号, 終端記号, 生成規則の有限集合, S は開始記号である.

```

N = { <expr>, <op>, <pre-ope>, <var> }
T = { +, -, *, /, sin, cos, exp, log, (, ), x }
S = <expr>
P = { <expr> ::= (<expr><op><expr>) (0)
      | <pre-ope>(<expr>) (1)
      | <var> (2)
      <op> ::= + (0)
      | - (1)
      | * (2)
      | / (3)
      <pre-ope> ::= sin (0)
      | cos (1)
      | exp (2)
      | log (3)
      <var> ::= x }
    
```

図 1 文法の例

遺伝子型から表現型への変換は開始記号から始まり, P の生成規則を用いて最左導出で別の記号列へ展開する. 全ての記号が終端記号に展開できれば, その終端記号列を個体の表現型とする. 1 つの非終端記号に複数の生成規則が定義されている場合, 式 (1)を用いてその生成規則番号(rule)を決定する.

$$\text{rule} = (\text{コドン}) \bmod (\text{展開する非終端記号の生成規則数}) \quad (1)$$

図 2 に表現型への変換例を示す.

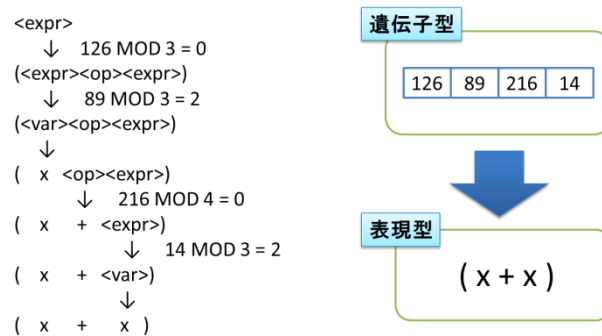


図 2 遺伝子型から表現型への変換

一般的な GE では一点交叉の際, 交叉点より後方の部分は交叉後に全く異なる表現型へと変換される可能性が高い. GE では, 個体の形質の残りやすさはその個体の遺伝子型の配置に依存し, 前方に配置された遺伝子ほど親個体と同じ形質を子個体に引き継ぐことができる.

この問題を解決するために, Bidirectional Representation を用いた GE が提案された. Bidirectional Representation を用いた GE では, 個体は 2 つの染色体を持つ. 2 つの染色体のうち, 一方

Grammatical Evolution Using Multiple Chromosome Representation Methods
[†]Takuya Mizuhara, Atsushi Ueno, Tomohito Takubo and Shoji Tatsumi
[†]Graduate School of Engineering, Osaka City University

は前端的の形質を優先的に保存し（最左導出染色体）、他方は表現型の後端的の形質を優先的に保存する（最右導出染色体）。同じ表現方法の染色体同士で交叉を行い、その後足りない染色体を片方の染色体から生成する。最左導出染色体と最右導出染色体は非終端記号の展開の順序が異なるだけなので、遺伝子型のコドン並び替えることで片方からもう片方の染色体を生成することができる。

3 Four-way Representation

Bidirectional Representation では、前端と後端の有用な形質を引き継ぐ可能性が高いが、中央部の形質に関しては依然として引き継がれる可能性が低いままである。交叉時に中央部の形質が失われる様子を図3に示す。

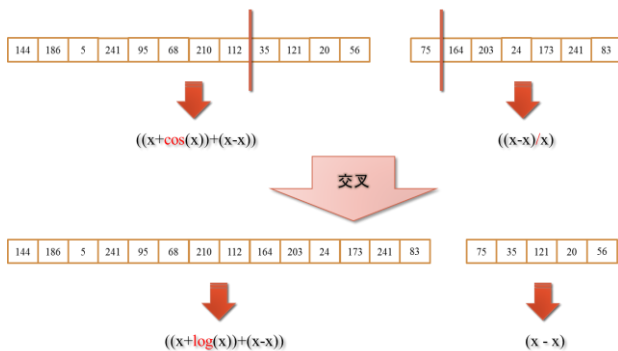


図3 中央部の形質の損失

染色体の長さが長くなるにつれて中央部の形質は子個体に引き継がれない可能性が高くなる。

そこで、個体に4つの染色体を持たせ、前端、後端だけではなく中央部の有用な形質も子個体に引き継ぐことができる Four-way Representation を提案する。Four-way Representation は Bidirectional Representation と同様に最左導出染色体と最右導出染色体を持ち、他に表現型の中央部分の形質を優先的に保存する染色体（中央左優先導出染色体と中央右優先導出染色体）を持つ。中央左優先染色体は、展開すべき非終端記号数が奇数の場合は中央の非終端記号を展開し、偶数の場合は中央から1つ左にある非終端記号を展開する。逆に中央右優先導出染色体は中央から1つ右の非終端記号を展開する。

4 実験及び考察

実験には Symbolic Regression を用いる。Symbolic Regression とは目的関数を近似する関数を求める問題である。本実験で用いた目的関数は文献[2]で用いられている $X^4 + X^3 + X^2 + X$ と $X^5 - 2X^3 + X$ の2つの関数である。それぞれ実験1、実験2とする。個体の評価値 e は、実験1では $-1 \leq x \leq 1$ の20個の点での目的関数との誤差の合計、実験2は $-1 \leq x \leq 1$ の50個の点での目的関数との誤差の合計である。適応度 f は以下の式(2)を用いる。

$$f = 1.0/e + 1.0 \quad (2)$$

実験に用いる文法は図1の文法を用いる。それぞれの実験で100試行を行い、各世代で得られた最良個体の平均適応度を図4、5に示す。

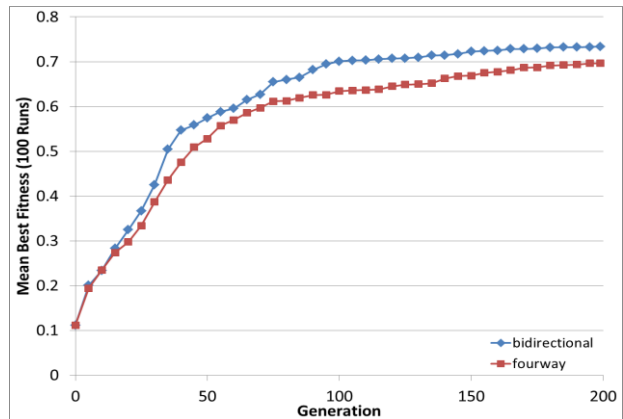


図4 実験1における最良個体の平均適応度

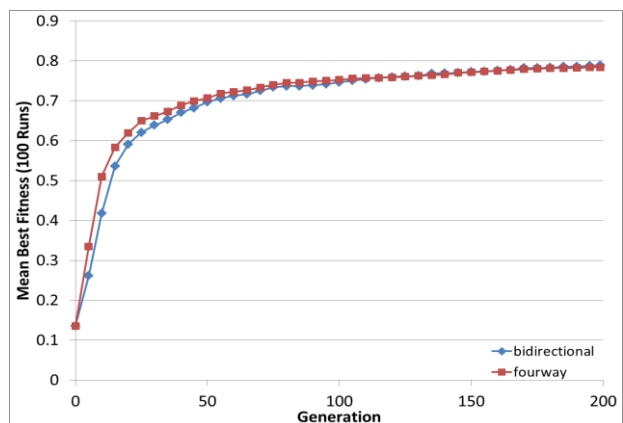


図5 実験2における最良個体の平均適応度

図4、5から Bidirectional と Four-way はほぼ同等の性能を示していることがわかる。これは、目的関数が比較的短い長さの染色体で生成することができるため、中央部の形質も Bidirectional で引き継がれる可能性が高いからと考えられる。

5 おわりに

本論文では新たに前端、後端、および中央部2か所の計4か所付近の有用な形質を子個体に引き継がせることができる手法として Four-way Representation を提案した。本論文では文献[2]で扱った目的関数でしか実験を行っていないので、異なる目的関数で実験を行うことが今後の課題である。

参考文献

[1] O'Neill, M., Ryan, C. "Grammatical Evolution", *IEEE Trans. Evolutionary Computation*. Vol. 5, No.4, 2001.
 [2] Jiri Kubalik, Jan Koutnik, and Leon J. M. Rothkrantz. "Grammatical Evolution with Bidirectional Representation", *EuroGP*, pp. 354-363, 2003.