2007－MPS－66 (2)
2007／9／3

Grammatical Evolutionの性能改善及びGPとの性能比較

岩澤博人　北栄輔
名古屋大学大学院　情報科学研究科

Grammatical Evolution(GE)は遺伝的アルゴリズム(GA)を拡張し，バッカス・ナウア記法(BNF)で定義された文法を用いて遺伝子型から表現型への変換を行うことにより，線形の遺伝子を用いながら遺伝的プログラミング(GP)と同様の構造的表現を扱うことができる進化的計算手法である．本研究ではこのGEの性能改善のために遺伝的操作に関するパラメータの変更，及び親2個体と子2個体から良い2個体を残す世代交代モデルであるElitist Recombination(ER)の導入を行った．その結果GEの性能改善を行うことができ，GPとの性能比較においても性能改善を行ったGEではGPよりも良い結果を得ることが出来た．

Performance improvement of Grammatical Evolution and performance comparison with GP

HIROTO IWASAWA　EISUKE KITA
Graduate School of Information Science, Nagoya University

Grammatical Evolution (GE) is an evolutionary computation that enhancing Genetic Algorithm (GA) and can treat a structural representation similar to Genetic Programming (GP) while using a linear gene by converting it into the phenotype from the genotype using the grammar defined by Backus-Naur Form (BNF). In this research, change concerning genetic operation parameter and Elitist Recombination (ER) who was the generation alternation model by whom two good individuals were select from two parent individuals and two child individuals was introduced for the performance improvement of GE. Therefore, the performance of GE improved, and the performance comparison with GP attained good results from GP.

1. はじめに

Grammatical Evolution(GE)はC.Ryanらによって提案された進化的計算手法で，遺伝的アルゴリズム(Genetic Algorithm：GA)をベースに遺伝的プログラミング(Genetic Programming：GP)と同様の構造的表現を扱うができるようにした手法である[1].

GEではGAと同様に各個体の遺伝子にビット列を用い，それに対して交叉や突然変異等の遺伝的操作を行うことで最適解の探索を行う．そして遺伝子の適合度評価の際に遺伝子型から表現型への変換を行う時，バッカス・ナウア記法(Backus Naur Form：

BNF)を用いることでGPと同様の構造的表現を扱うことができる.これにより従来GAでは扱うことの難しかった複雑な数式やプログラム等をGAの遺伝子と同様に扱うことができるようになり,GPでは必要であった構造の操作無しに構造的表現を扱うことが可能となっている.

本研究ではこのGEの基本性能の調査及び改善と,GPとの比較を行い,GEの有用性を調べる.

2. GEのアルゴリズム

GE全体のアルゴリズムはGAと同様に最初に初期個体の評価を行い,次に各遺伝子に交叉などの遺伝的操作を行った後,各個体の適合度の評価を行う.この操作を終了条件を満たすまで繰り返す.

GEのGAと異なる部分は,各個体の適合度を評価する際の遺伝子型から表現型への変換方法である.GAではビット列を数値に変換する事が多いが,GEではこの変換にBNFを用いた遷移規則を利用することによりビット列を数式やプログラム等の構造的表現へと変換することが可能となっている.この変換方法がGEの特徴であり,次にその方法を述べる.

2.1 BNFを用いた遺伝子型から表現型への変換

最初に遺伝子型から表現型へ変換を行うためのBNFを定義する.例として以下のようなBNFを定義する(開始記号:<expr>).

(A) <expr> ::= <expr><expr><op> (A0)
| <var> (A1)

(B) <op> ::= + (B0)
| - (B1)
| * (B2)
| / (B3)

(C) <var> ::= X (C0)
| Y (C1)
| Z (C2)

このBNFの定義に従って数式を作る場合,X,Y,Zの3変数を用いた四則演算を逆ポーランド記法で表記する数式を生成することができる.

次に各遺伝子をこの遷移規則に従い数式に変換する.例としてある個体の遺伝子が次のようになっているとする.

10000111000010110011111111000110

このような場合に以下の手順で遺伝子型から表現型への変換を行う.

1. 遺伝子型の2進数列を任意の決められた長さ毎に整数へ変換する.例として3bit毎に整数へ変換すると,前述の遺伝子は次のような整数列に変換される.
   4 1 6 0 5 4 7 7 0 6
2. 次にこの整数列を遷移規則に当てはめ,数式への変換を行う.最初に選択される遷移規則は開始記号が<expr>なので(A)であり,適用させる整数は先頭の**4**である.(A)の遷移規則の数は(A0)と(A1)の2つあるが,GEではこの選択に適用させる整数の遷移規則数に対する除算の剰余を用いる.この場合では適用させる整数**4**を遷移規則数2で割った余りは0であるので,余り0に対応する規則<expr><expr><op>(A0)が選択される.
3. 選択された規則<expr><expr><op>の左端から非終端記号が終端記号に変換されるまで順に変換を行う.ここでは左端の非終端記号が<expr>であるので選択される遷移規則は(A),整数は**1**になるので上記と同様に剰余(1%2=1)から選択される規則<var>(A1)が決まる.

以下同様にして適用される整数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的XYY+*という数式を生成することができる(図1).そしてこの生成された数式に対して評価関数を用いて各個体の適合度を決定する.

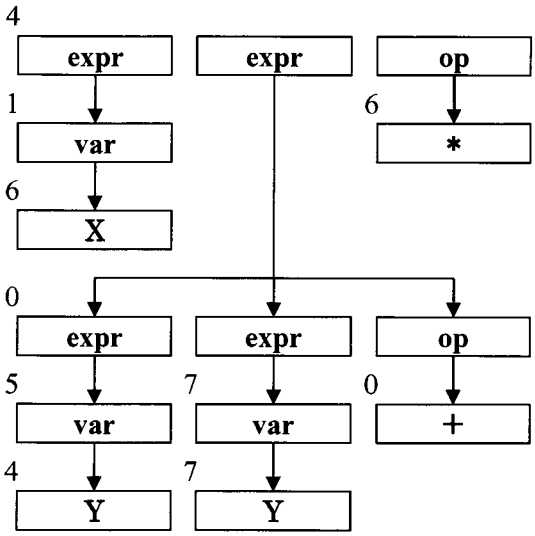

図 1：遺伝子型から表現型への変換例
Figure.1：Transform genotype to phenotype

3 実験

GE の性能を調べるために実験を行った．最初に単純 GA(SimpleGA：SGA)と同様のルーレット選択，一点交叉，突然変異にエリート保存選択を加えた遺伝的操作により実験を行った．使用したパラメータは以下のようである．

世代数：1000　個体数：100　個体長：100
選択方法：ルーレット選択
エリート保存選択で残す個体数：1 体
交叉方法：一点交叉　交叉率：0.9
突然変異率：0.01
また使用した文法は以下の通りである．

<expr>　::= <expr><expr><op> | <x>
<op>　::= + | - | * | /
<x>　::= x

この文法を用いて関数同定問題の実験を行う．同定させる関数は

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

を使用し，−1 から 1 までの計 21 個のデータを用いた．適合度は各データとの差の平均二乗誤差とし，値が小さいほど良いとする．試行回数は各 50 回，ビット列から整数列への変換は 4bit 毎とした．この実験の結果を図 2 に示す．

3.1 遺伝的操作パラメータの変更による性能改善

次に各遺伝的操作のパラメータの変更を行い良い結果が出るものを探した．その結果，以下のパラメータの時に最も良い結果が得られた．

選択方法：トーナメント選択
トーナメントサイズ：20
エリート保存で残す個体数：1 体
交叉方法：一点交叉　交叉率：0.5
突然変異率：0.1

なお表記していない世代数，個体数，個体長の 3 種のパラメータ及び使用した文法は本研究においてすべて共通である．この実験の結果を図 3 に示す．

3.2 Elitist Recombination を用いた性能改善

続いて更なる性能改善の為に Elitist Recombination(ER)を導入した．ER は次世代を選ぶ際に各家族，つまり親として選択された 2 個体と親個体から交叉により生成された子 2 個体の中から最良の 2 個体を次世代の個体として選択する世代交代モデルである．この実験の結果を図 4 に示す．

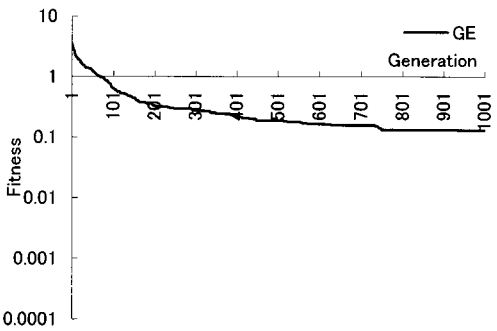

図 2：GE 実験結果
Figure.2：Experimental result of GE

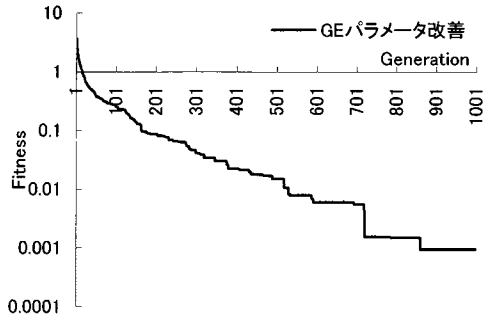

図 3：GE パラメータ改善実験結果
Figure.3：Experimental result of improvement parameter for GE

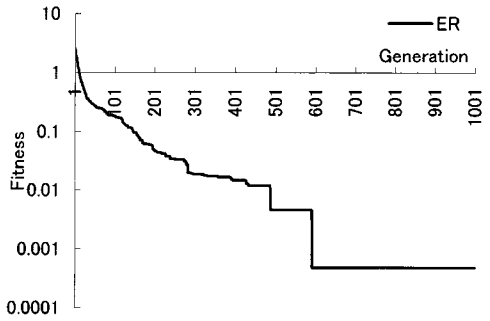

図 4：ER を用いた GE 実験結果
Figure4：Experimental result of using ER for GE

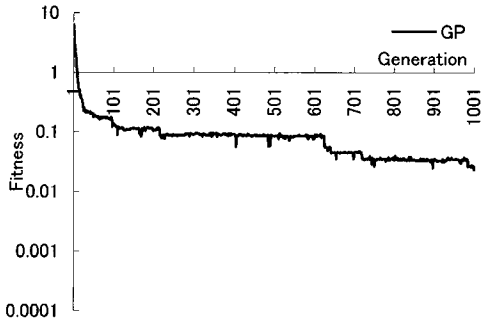

図 5：GP 実験結果
Figure.5：Experimental result of GP

4. GP との比較

3 章で行った実験と同じ関数同定問題を GP を用いて解いた実験の結果を図 5 に示す.

結果からパラメータの変更を行っていない GE は GP よりも性能が劣るが, パラメータの調整を行った GE 及び ER を用いた GE では GP よりも結果がよくなっていることがわかる. このことから GE は関数同定問題に関して GP と比較して有用であると考えられる.

5. まとめ

本研究では GE の遺伝的操作のパラメータの変更, 及び世代交代モデルの変更により性能の改善が行えることを示し, 同時に GP と比較しても良い結果を出せることがわかった.

今後は遺伝的操作部分の改善だけではなく、文法部分に関わる部分においても改良を行うことによるさらなる性能の改善及び関数同定以外の様々な問題に対して GE の適用を行っていきたいと考えている。

参考文献

[1] C.Ryan, J.J.Collins, and M..O'Neill. Grammatical Evolution: Evolving Programs for an Arbiturary Language. Proc. of 1st European Workshop on Genetic Programming, pages 83-95. Springer-Verlag.

[2] A.Brabazon and M.O'Neill. Biologically Inspired Algorithms for Financial Modelling. Springer. 2006.

[3] 本間淳也, 渡辺慎哉, 宮本衛市. Grammatical Evolution の拡張による構造化プログラムの自動生成. 電子情報通信学会技術研究報告. Vol.100, No.185(20000706) pp. 9-17

[4] 坂和正敏, 田中雅博. 遺伝的アルゴリズム. 朝倉書店. 1995.

[5] 伊庭斉志. 進化論的計算手法. オーム社. 2005.

[6] 中田育男.コンパイラ. オーム社. 1995.

[7] 佐藤 浩, 小野 功, 小林 重信. 遺伝的アルゴリズムにおける世代交代モデルの提案と評価. 人工知能学会誌 Vol.12, No.5(19970901) pp. 734-744.