システム制御情報学会論文誌, Vol. 20, No. 11, pp. 454–456, 2007

研 究 速 報

関数同定問題での遺伝的プログラミングにおける螺旋交叉法の実験的検討*

中山　茂†・前薗　正宜†・飯村伊智郎‡・小野　智司†

Experimental Consideration on Helical Crossover Method in Genetic Programming for Function Identification Problem*

Shigeru NAKAYAMA†, Masaki MAEZONO†, Ichiro IIMURA‡ and Satoshi ONO†

1. はじめに

量子系の干渉効果を模した干渉交叉（Interference Crossover: IX）は，遺伝的アルゴリズム（GA）における交叉オペレータの一つとして，1996年にNarayananらによって提案され[1]，巡回セールスマン問題（TSP）を対象とした実験でその効果が示された．筆者らは，TSPの都市数を増やして評価実験を行い，GAおよび免疫アルゴリズムにおける干渉交叉は最適解発見率の向上および探索世代数の短縮の観点で有効であることを確認した[2,3]．その後，この干渉交叉を，DNAの螺旋構造に由来する螺旋交叉（Helical Crossover: HX）として解釈し，遺伝的プログラミング（GP）[4]に適用する方法を提案した[5]．本研究では，GPにおける螺旋交叉の有無による探索性能への影響を，関数同定問題[6]に初めて適用し計算機実験を通して検討した．

2. GPにおける螺旋交叉法

GPにおける染色体は木構造を成しており，従来の交叉としては，部分木の交換による方法が一般的である．GAの染色体のようなリスト構造に対して用いられる螺旋交叉[2]の，木構造型染色体への適用は，木構造型染色体を一時的にリスト構造へ変換することで実現可能となる．木構造型染色体のリスト構造への変換，および具体的な螺旋交叉の処理手順については，文献[5]を参照されたい．

3. 螺旋交叉法を用いたGPの評価実験

3.1 関数同定問題への適用

GPによる関数同定では，終端ノードとして変数や定数など，非終端ノードとして演算子などを用いる．これらのノードを木構造状に組み合わせたものを1個体とし，関数木fを表す．関数木fは葉を入力とし，葉から根へと各ノードが示す演算を行い，最終的に根から関数木fの出力値が得られる．GPでは，個体集団に対して遺伝的操作を行い，各木が示す関数の出力値が目的とする未知関数の出力値に近づくように進化させる．本研究では，未知関数を表す木構造を構成するための終端・非終端ノードとして，以下のものを用意した．

終端ノード： 変数$\{x,y\}$，定数．
非終端ノード（引数：1個）： sin, cos, tan, ln, abs, floor, ceil, rint（引数に最も近い整数値を出力）．
非終端ノード（引数：2個）： +, −, ×, ÷, pow.

関数同定は，対象とする関数によってその難易度が異なるため，今回の実験においては，次に示す20種類（(1)式から(20)式まで）の関数を対象とした．

$$f_1(x)=\frac{1}{x}-\frac{1}{x^2} \quad (1) \qquad f_2(x)=\frac{1}{x^2+x+1} \quad (2)$$

$$f_3(x)=3^x \quad (3) \qquad f_4(x)=(x+1)^3 \quad (4)$$

$$f_5(x)=1.5^{\frac{x}{2}} \quad (5) \qquad f_6(x)=\ln(x^2)+x^2 \quad (6)$$

$$f_7(x)=\left(x+\frac{1}{x}\right)^2 \quad (7) \qquad f_8(x)=\left(x+\frac{1}{x^2}\right)^2 \quad (8)$$

$$f_9(x)=\frac{9}{x^2+x+3} \quad (9) \qquad f_{10}(x)=\frac{3}{x}-\frac{2}{x^2}+1 \quad (10)$$

$$f_{11}(x)=3^x+x^3 \quad (11) \qquad f_{12}(x,y)=x^3+y^3 \quad (12)$$

$$f_{13}(x)=x^3+x^2+x+1 \quad (13)$$

$$f_{14}(x,y)=\cos(x-y)+\cos(x+y) \quad (14)$$

$$f_{15}(x,y)=-x^2+xy+y^2 \quad (15)$$

$$f_{16}(x)=x^3+2x^2+2x+1 \quad (16)$$

$$f_{17}(x,y)=\cos\left(\frac{x-y}{3}\right)+\cos\left(\frac{x+y}{3}\right) \quad (17)$$

$$f_{18}(x,y)=x^2-xy+y^2+1 \quad (18)$$

$$f_{19}(x)=\frac{1}{x^2+1}+\cos(x) \quad (19)$$

$$f_{20}(x,y)=\sin\left(\frac{x}{2}\right)+\sin\left(\frac{y}{2}\right) \quad (20)$$

3.2 実験内容

GPにおける螺旋交叉の有無による探索性能への影響を調べるため，古典的交叉[1]と螺旋交叉とを併用するGP

* 原稿受付　2007年7月2日

† 鹿児島大学 工学部　Faculty of Engineering, Kagoshima Univ.; 1-21-40 Korimoto, Kagoshima 890-0065, JAPAN
‡ 熊本県立大学 総合管理学部　Faculty of Administration, Pref. Univ. of Kumamoto; 3-1-100 Tsukide, Kumamoto 862-8502, JAPAN

Key Words: genetic programming, helical crossover, interference crossover, function identification problem.

[1] 螺旋交叉との対比のため，ここでは従来の交叉を古典的交叉（Classical Crossover: CX）とよぶ．

(GP_{CX+HX} とよぶ）と，古典的交叉のみの従来の GP（GP_{CX} とよぶ）との比較を行った．本実験では，関数同定に適切な古典的交叉率 r_{CX} や螺旋交叉率 r_{HX} を確かめるため，それら交叉率を変化させながら行った．GP_{CX} では，r_{CX} を 100% から 0% まで −10% の間隔で変化させ，GP_{CX+HX} では，GP_{CX} の場合と同様に r_{CX} を変化させ，併せて r_{HX} を 0% から 100% まで +10% の間隔で変化させた．各条件ごとに 30 回試行し，同定成功回数が最も多い交叉率について，両者の同定成功率を比較評価した．

GP の個体数は 200 とし，初期集団は，各個体の最大ノード数が 20，最大葉数が 8，最大深さが 15 を超えない木をランダムに生成した．このとき，定数は 0 から 100 までの範囲で，小数第 2 位までの精度の正の小数値をランダムに生成した．適応度[1]の算出に用いる入力データは，−10 から +10 までの整数のうち，0 を除いた 20 個の値とした．また，2 変数の場合には，変数ごとに上記の 20 個の値を用意し，各値の全組合せである 400 個を入力データとした．選択にはエリート保存戦略を，古典的交叉には部分木の交換による 1 点交叉を採用し，突然変異では部分木をランダムに変更するものとした．古典的交叉では，まずエリート個体を除く個体集団から古典的交叉率 r_{CX} に応じた数の個体をランダムに非復元抽出し，親個体集団とする．親個体をそれぞれ 2 個 1 組として 1 点交叉を行い，2 個の子個体を生成する．交叉後，親個体は生成された子個体で淘汰される．その後，螺旋交叉，そして突然変異と処理される．螺旋交叉 [5] では，その実行後，親個体と子個体を合わせた集団からルーレット選択[2]により螺旋交叉率 r_{HX} に応じた数の個体を選び出し，それらを次世代へ残すものとした．なお，螺旋交叉の処理を決定する設定項目 [5] は，以下のようにした．

【親個体の選択】 エリート個体を除く個体集団のうち，古典的交叉の親として選ばれなかった個体の集団からランダムに螺旋交叉率 r_{HX} に応じた数の個体を非復元抽出し，親個体集団とする．

【親個体の配置】 親個体の配置は，ランダムとする．

【主幹ノードの決定】 主幹ノードは，最深葉ノードが主幹ノードの終端となるように設定する．

【ダミーノードの挿入位置】 ダミーノードは，最深葉ノードの直前に挿入する．

突然変異では，まず突然変異率に応じた数の個体をランダムに選び出し，その個体の部分木を，ランダムに生成した新たな部分木と置き換えたものを新たな個体とする．なお，突然変異率は 1%とした．また，すべての遺伝的操作（選択，古典的交叉，螺旋交叉，突然変異）の後，GP 特有のブロートを抑制するための冗長ノードの除去，および効率的な探索を目的とした終端ノードにおける定数に対する局所探索を行うものとした．終了条件としては，探索が成功したか，進化が 5,000 世代に達した時点で探索を終了するものとした．

3.3 実験結果と考察

各関数について，関数同定の成功率（同定成功率）が最も高かったときの交叉率である最適交叉率（Best crossover rate）と，そのときの同定成功率（Success rate）をまとめたものを，Table 1 に示す．同表では，最適交叉率は「$r_{CX}+r_{HX}$」の形式で示し，同定成功率は 30 回の試行において適応度が 1.0 に達した割合である．適応度が一度も 1.0 に達しなかった場合については，その関数の最適交叉率は “—” と表示している．

3.3.1 同定成功率の比較

同定成功率を比較すると，大きく 4 ケースに分類できる．まずケース A では，GP_{CX}，GP_{CX+HX} ともに同定成功率が 20%以下と極端に低い．これは，今回用いたアルゴリズムでは問題が難しすぎたためにほとんどの試行において関数を同定できず，両者の違いによる差が現れにくくなっている．またケース B では，GP_{CX}，GP_{CX+HX} ともに同定成功率が 85%以上と高い値を示している．これは，問題がやさしすぎたためにほとんどの試行において関数を同定できており，前者同様，差が現

Table 1　Experimental results on GP_{CX} and GP_{CX+HX}

Case	Func.	Best crossover rate [%]		Success rate [%]	
		GP_{CX}	GP_{CX+HX}	GP_{CX}	GP_{CX+HX}
A	f_5	—	50 + 50	0.0	10.0
	f_9	—	—	0.0	0.0
	f_{10}	100 + 0	50 + 50	13.3	10.0
	f_{17}	—	40 + 60	0.0	3.3
	f_{19}	90 + 0	90 + 10	13.3	10.0
B	f_1	90 + 0	90 + 10	90.0	100.0
	f_3	80 + 0	50 + 50	100.0	100.0
	f_{13}	100 + 0	70 + 30	86.7	93.3
C	f_2	90 + 0	80 + 20	46.7	30.0
	f_4	50 + 0	90 + 10	46.7	36.7
	f_{11}	100 + 0	90 + 10	43.3	46.7
	f_{12}	100 + 0	80 + 20	76.7	66.7
	f_{14}	100 + 0	70 + 30	63.3	70.0
	f_{15}	90 + 0	80 + 20	30.0	26.7
	f_{20}	100 + 0	90 + 10	20.0	20.0
D	f_6	60 + 0	40 + 60	20.0	70.0
	f_7	70 + 0	40 + 60	26.7	93.3
	f_8	50 + 0	60 + 40	46.7	86.7
	f_{16}	70 + 0	80 + 20	50.0	76.7
	f_{18}	100 + 0	40 + 60	53.3	93.3

[1] 個体 S の適応度 $fitness(S)$ は，次式によって定義される．

$$fitness(S)=\frac{1}{K}\sum_{k=1}^{K}\delta(x_k)$$

ここで，K は入力データ数，$\delta(x_k)$ は $O(x_k)$ と $T(x_k)$ との非類似性を表し，次式によって求める．

$$\delta(x_k)=\frac{1.0}{|O(x_k)-T(x_k)|+1.0}$$

ただし，x_k は k 番目の入力データであり，$O(x_k)$ は S が表す関数に x_k を入力して得られた出力値，$T(x_k)$ は目標関数の出力値である．

[2] 各個体が選択される確率を決めるルーレット盤は，予備実験をもとに，各個体の適応度を二乗した値を用いて作成した．

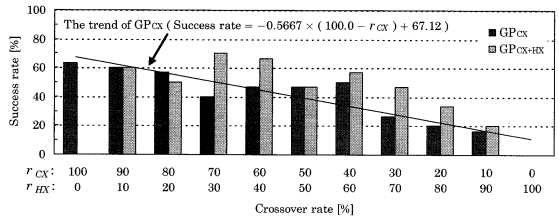

Fig. 1 Transition of success rate in the case of f_{14}

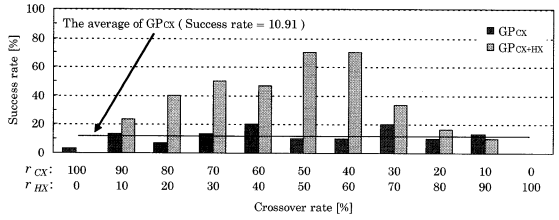

Fig. 2 Transition of success rate in the case of f_6

れにくくなっている．上記以外で同定成功率がほぼ等しいケースCでは，Fig. 1に示す関数f_{14}の例から読み取れるように，r_{CX}が減少するに従い，ほぼ一様に同定成功率が減少しており，古典的交叉が同定成功率に強く影響していることがわかる．Table 1の最適交叉率に注目すると，GP_{CX}では関数f_4を除くすべての関数でr_{CX}が100%あるいは90%である．GP_{CX+HX}においても，r_{CX}は90%〜70%であり，高いr_{CX}においてよい結果を示している．本実験では，r_{CX}とr_{HX}との和が100%となるように設定しているため，r_{CX}が高いときによい結果が得られるケースでは，結果的にr_{HX}は低いものとなる．このため，螺旋交叉が十分に適用されず，その効果が得にくい状況になっているものと考えられる．この点については，今後詳細に検討していく必要がある．一方でケースDでは，GP_{CX+HX}の同定成功率がGP_{CX}を大きく上回っており螺旋交叉の効果が確認できる．これは，Fig. 2に示す関数f_6の例からも読み取れ，ケースCとは異なり，r_{CX}を変化させても一様に同定成功率は低く，このことからr_{CX}が同定成功率に及ぼす影響は弱く，古典的交叉がほとんど効果がないことがわかる．螺旋交叉の効果が大きく現れている関数では，r_{HX}が40%〜60%であり，r_{CX}はケースCと比べて低くなっている．これらの結果は，先行研究[2,3]と同様，螺旋交叉の導入が個体の多様性を維持し，局所解からの脱出に貢献できた結果であると考えられる．一方，r_{HX}が高すぎると同定成功率が低下することも確認できた．このような現象が起きた理由としては，螺旋交叉による過剰な多様化が考えられ，螺旋交叉を効果的に組み込むには適切なr_{HX}の設定が必要であることを示唆している．

3.3.2 平均同定成功世代に関する考察

Table 1の最適交叉率において，適応度が1.0に達した世代（同定成功世代）の平均値について考察する．Fig. 3は，同定成功世代の平均値におけるGP_{CX+HX}に対するGP_{CX}の比率を示している．同図より，関数f_3，f_6，f_7，

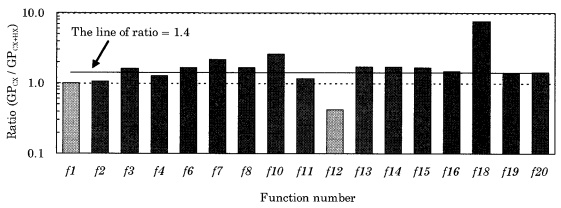

Fig. 3 Ratios (GP_{CX}/GP_{CX+HX}) on average generation for searching an optimal solution

f_8，f_{10}，f_{13}，f_{14}，f_{15}，f_{16}，f_{18}，f_{20}で，GP_{CX}の平均同定成功世代がGP_{CX+HX}の1.4倍以上であることがわかる．それ以外の関数についても，関数f_1，f_{12}を除いた関数で，比率が1.0を上回っており，GP_{CX+HX}は多くの関数でGP_{CX}よりも早い世代で解を探索できていることがわかる．しかしながら，螺旋交叉を併用した場合にはルーレット選択が加わることから，GP_{CX+HX}の適応度評価回数がGP_{CX}に比べ多くなる．この点をふまえた平均同定成功世代の評価については，今後，適応度評価回数を等しくしたうえで詳細に評価していく必要があると考えている．

4. おわりに

本研究では，GPにおける螺旋交叉の有無による探索性能への影響を調べるため，関数同定問題を用いて評価実験を行った．実験の結果，螺旋交叉を導入することにより，古典的交叉のみでは十分に探索できない問題の一部に対して，同定成功率を大幅に改善できており，螺旋交叉が有効であることがわかった．その際，古典的交叉と螺旋交叉を同程度の割合で用いることで効果を発揮することもわかった．

しかしながら，関数によっては螺旋交叉の導入効果が見られないものもあり，今後は，GPにおける螺旋交叉による多様性の維持について詳細に分析したい．

参考文献

[1] A. Narayanan and M. Moore: Quantum-inspired Genetic Algorithms; *Proc. IEEE Int. Conf. Evolutionary Camputation*, pp. 61–66 (1996)

[2] 中山，飯村，松尾，前薗：遺伝的アルゴリズムにおける干渉交叉法の検討；情処学論，Vol. 47, No. 8, pp. 2625–2635 (2006)

[3] 中山，飯村，伊藤：免疫アルゴリズムにおける量子干渉交叉法の検討；信学論 D-I，Vol. J88-D-I, No. 12, pp. 1795–1799 (2005)

[4] J. R. Koza: *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, MIT press (1992)

[5] 中山，前薗，小野：遺伝的プログラミングにおける螺旋交叉戦略；システム制御情報学会論文誌，Vol. 19, No. 6, pp. 262–268 (2006)

[6] 又吉：Tree-染色体構造をもったGAでの関数同定；信学論 D，Vol. J82-D-I, No. 11, pp. 1327–1335 (1999)