

科学研究費助成事業 研究成果報告書

平成 27 年 5 月 29 日現在

機関番号：25403

研究種目：若手研究(B)

研究期間：2013～2014

課題番号：25730149

研究課題名(和文)アリの群知能を利用したグラフ構造プログラムの自動生成に関する研究

研究課題名(英文)Automatic Generation of Graph-structural Programs by Using Swarm Intelligence of Ants

研究代表者

原章(HARA, Akira)

広島市立大学・情報科学研究科・准教授

研究者番号：70347615

交付決定額(研究期間全体)：(直接経費) 1,000,000円

研究成果の概要(和文)：本研究は、群知能を用いてグラフ構造プログラムを自動生成する手法を提案することを目的としている。先に筆者らが提案したアリコロニー最適化に基づくグラフ構造生成手法Cartesian Ant Programming (CAP)では、プログラムの構成要素となる関数記号をノードに固定的に割り当てるため、局所解に陥りやすいという問題があった。平成25年度の取り組みでは、探索状況に応じて各ノードへの関数記号の割り当てを修正する「ノードの置換を行うCAP」を提案した。また、平成26年度には、さらに探索性能の向上を図るため、「ノードへの関数記号の割り当て機能を備えたノード解放型CAP」を提案した。

研究成果の概要(英文)：The aim of this research is to develop an automatic generation method of graph-structural programs by using swarm intelligence. We have already proposed Cartesian Ant Programming (CAP) as an automatic programming method. But, CAP search tends to fall into local optima because assignments of functional symbols (program elements) to respective nodes are fixed and inflexible. In my 2013's research, we have proposed an improved CAP method, CAP with adaptive node replacements. The assignments of functional symbols changes adaptively based on the state of search in this method. In my 2014's research, for improving the search performance, we have introduced two new mechanisms into CAP, optimization of function assignments by pheromone communication, and limitation or release of the use of a part of nodes. These methods showed better performance than conventional CAP.

研究分野：進化計算

キーワード：群知能 アリコロニー最適化法 遺伝的プログラミング

1. 研究開始当初の背景

進化計算を用いたプログラムの自動生成手法として、遺伝的プログラミング (Genetic Programming; GP) がある。通常、GP では木構造プログラムを取り扱う。プログラムの構成要素は、木構造の葉ノードを構成する終端記号と、木構造の内部節点を構成する関数記号からなる。初期世代では記号群をランダムに組み合わせることによって問題の解候補となる木構造プログラムを生成する。この木構造で表現された個体の集団に対し、選択、交叉、突然変異といった遺伝操作を適用することにより、優れた解を獲得する。

GP における交叉や突然変異は、個体群の多様性を維持するために用いられる。この遺伝操作を繰り返すことにより、より適応度の高い新たな個体が生成される可能性がある。その一方で、イントロンと呼ばれる木の評価に影響を与えない不要な部分木が増大し、探索効率に悪影響を与えるブロートと呼ばれる現象が引き起こされる。また、部分木の選択や生成をランダムに行う従来の遺伝操作では、良い解の周辺を局所探索する効果を生み出すことは困難である。

本研究では、生物の進化のメカニズムに基づくのではなく、生物の環境への適応モデルである群知能に基づいたプログラムの自動生成手法を提案する。ここでは、群知能の1つであるアリコロニー最適化法 (Ant Colony Optimization; ACO) を利用する。ACO は、フェロモンを介したコミュニケーションを用いた蟻の採餌行動をモデル化したものである。ACO は、従来、巡回セールスマン問題に代表される組み合わせ最適化問題を対象としたアルゴリズムであるが、これを自動プログラミングに応用する。プログラムの構成要素の組み合わせを、交叉や突然変異ではなく、アリのフェロモンコミュニケーションにより最適化する。ここでは、GP における終端記号、非終端記号を図1のようにマップ上に配置した環境を想定し、アリの群れがその環境を動くことによりその結合方法を自動的に最適化することを考える。

従来研究においても、Dynamic Ant Programming (DAP)[1]など、既にACOを利用したアントプログラミング手法の試みがいくつかなされている。しかし、従来の方法では、アリは木構造プログラムを作成するため、部分構造を再利用した表現力の高いプログラムを作成することができていない。そこで本研究では、グラフ構造プログラムを自動生成できるアントプログラミング手法を提案する。グラフ構造プログラムは、ある部分グラフの出力を何度も活用することが可能となるため、コンパクトでありながら表現能力の高いプログラムを自動生成できる。既に筆者らのこれまでの研究において、関数記号を表すノードを環境内にグリッド状に固定して配置したアントプログラミング手法の検証実験[2]を行い、従来のGPに迫る探索性

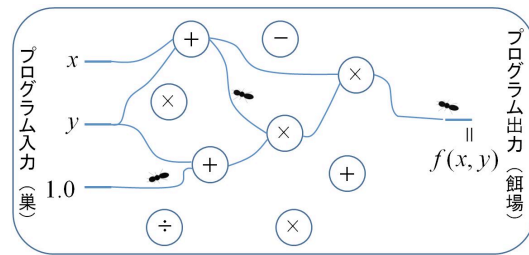


図1. アリによるプログラム (関数) 生成

能を示すことを確認している。関数ノードを探索の過程で環境内に適応的に配置できれば、従来のGPを上回る性能が得られる可能性があり、本研究課題でその方法を探求することとした。

2. 研究の目的

本研究は、群知能を用いてグラフ構造プログラムを自動生成する手法を提案することを目的とする。群知能による探索のポジティブフィードバック性を活かすことで、従来のGPよりも局所探索性能を向上させた手法の実現を目指す。本研究では群知能としてアリエージェントの群行動を考える。図1のように、プログラムの出力をアリエージェントにとっての餌場、プログラムの入力アリエージェントにとっての巣と考え、これらを2次元平面上のマップの両端に配置する。その間にプログラムの構成要素となる関数記号を配置し、アリの餌場から巣への移動の流れを最適化することでプログラムの自動生成を行う。この際、生成されるプログラムがグラフ構造プログラムとなるように環境およびアリの行動の設計を行う。従来手法が作成する木構造プログラムに比べて、グラフ構造プログラムは部分グラフ構造を再利用することが可能になる。このため、従来手法よりもコンパクトで表現力の高いプログラムの生成が可能となる。本研究では、グラフ構造プログラムの自動生成に適したアリの移動環境 (マップ) およびアリの行動規則を明らかにする。

本研究で提案する手法は、群中の個体同士および個体と環境とのインタラクションにより効果的な局所探索を実現する。群知能が持つ探索のポジティブフィードバック性により、局所探索性能の向上が達成できると考えられる。それに加えて、群知能アプローチは、群を構成する個体レベルの行動設計により様々な拡張が可能となることが利点である。

3. 研究の方法

(1) 平成25年度は、グラフ構造プログラムを自動生成するアントプログラミングの基本アルゴリズムの設計を行った。アルゴリズムの設計にあたっては、以下の3要素の観点からの検討を行った。

- ・アリエージェントの行動環境の設計

- ・アリエージェントの行動規則の設計
- ・アリエージェントの行動評価およびフェロモンによる環境へのフィードバック方法の設計

アリエージェントの行動環境に関数記号ノードを配置する場合に、ノードの種類や位置を固定して配置するには問題に対する事前知識が必要となる。また、探索開始時に割り当てられた配置では局所解に陥る可能性がある。この問題を解決するため、探索途中で解の表現に利用されていないノードについて、割り当てられた関数記号を別の記号に置換して新たな解の発見を促す手法について検討した。

提案手法の有効性は、関数同定問題および二重螺旋問題において、従来手法である Cartesian GP (CGP) および CAP と比較することにより検証した。

(2) 平成 26 年度は、探索開始時の初期化プロセスにおける関数記号のノードへの割り当てに対する依存を完全に解消した汎用性の高い手法を実現するため、ノードの機能(関数記号)を固定しない汎用的なノードを配置し、そのノードにどの機能を割り当てるかについてもアリエージェントの選択によって決定する方法を検討した。また、従来手法について、獲得されたプログラムがノード群をどのように利用しているか解析したところ、入力(巢)に近い位置にあるノードを集中して利用していることが明らかになった。これが探索性能の向上を妨げる要因となっていると考えられるため、ノードの利用の偏りを解消する方法についても検討した。

提案手法の有効性は、関数同定問題において、従来手法 CAP と性能比較することにより検証した。

4. 研究成果

平成 25 年度、26 年度の研究成果である「ノード置換を行う CAP」、および「ノードへの関数記号の割り当て機能を備えたノード解放型 CAP」について、その概要を述べる。

(1) ノード置換を行う CAP

従来 CAP では、非終端記号が周期的に割り当てられたノード列をアリエージェントが移動することによりプログラムを生成する。非終端記号の並びが固定であるため少ないノード数では探索がうまく行えないことが問題点として挙げられる。平成 25 年度の研究では、この CAP の問題を解消するため、ノードの置換を行う CAP を提案した。

提案手法では、利用されていないノードに割り当てられた関数記号を有用と思われる関数記号に置換することにより、CAP の問題を解決する。ノードの置換の手順を以下に示す。

1. 図 2 のように各ノードに接続する経路のフェロモン濃度の平均を求め、その平均値を

比較して最も低い平均値を持つノードを置換対象ノード n_{rep} として選択する。

2. 直近の t_{rep} 世代の間に各関数記号に付与されたフェロモン濃度の総計を求める。

3. n_{rep} の関数記号を 2. で求めたフェロモン濃度の割合に基づいて選択した新たな関数記号と置き換える。

4. n_{rep} の入力端子から伸びる全ての経路のフェロモン濃度を初期化する。

5. n_{rep} の選択確率を上げるため、全てのノードの入力に接続する経路のフェロモン濃度を次式で調整する。

$$P = P \times j \quad (1)$$

ここで P はノード j の入力に接続する各経路のフェロモン濃度である。フェロモン量には上限が設定されているため、結果的にフェロモンの偏りが減少し、大域的探索に移行する。

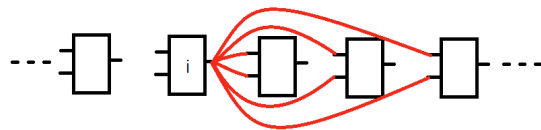


図 2 ノード i の出力に接続する経路の例

提案手法の有効性を検証するため、関数同定問題および二重螺旋分類問題において、CGP、従来の CAP と性能を比較した。表 1 は関数同定問題における各手法の結果である。なお、表中の n_c は遺伝子型のノード数を表している。提案手法は CGP、従来の CAP の両手法を上回るか同等の結果を示した。また、従来の CAP より接続ノード数が少ないことから、よりコンパクトなプログラムを生成できたことが分かる。

表 1 関数同定問題における性能比較

	CGP		従来のCAP		ノードの置換を行うCAP	
	適合度	接続ノード	適合度	接続ノード	適合度	接続ノード
$n_c=10$	16.5	9.8	28.1	8.9	25.5	8.68
$n_c=30$	4.7	14.2	3.99	20.4	2.45	20.1
$n_c=50$	3.89	14.3	2.22	26.6	1.68	22.9
$n_c=70$	3.36	14.9	2.36	29.4	0.767	22.1
$n_c=100$	4.52	21.9	2.03	32.6	0.485	23.7

(2) ノードへの関数記号の割り当て機能を備えたノード解放型 CAP

平成 26 年度に新たに提案した手法は、従来の CAP において周期的かつ固定的に割り当てていた関数記号を、フェロモンに応じてアリによって決定するものである。関数記号を動的に割り当てることで、ノード数に関わらず効率的な探索が可能になる。また、世代を分割し使用できるノードを徐々に増やすノード解放型 CAP を提案する。これは、入力側に近いノードに偏って使用するアリの習性を解消し、最終世代までより有効に進化させるためのものである。

① フェロモン濃度に基づき関数記号を割り当てる CAP

ノードへの関数記号の割り当ては各世代でアリごとに行う。CAP と同様に、全体の出力からスタートしたアリは入力に向かって経路用のフェロモンテーブルに従い移動を繰り返す。この際、訪問したノードに関数記号を割り当てる。割り当ては関数記号用のフェロモンテーブルを用いて確率的に行う。テーブルには各ノードにおける関数記号ごとのフェロモン濃度が保存されている。

フェロモン更新については、経路と同様に、ノードごとに使用された関数記号に対して適応度に応じて加算し、その他の使用されていないものについては一定の割合で蒸発させる。

② ノード解放型 CAP

この手法は、世代を分割し各世代で使用できるノードを徐々に解放していくことで、最終的なノードの使用を分散させ、ノード使用の偏りによる自由度の低下を改善するものである。

解放メカニズム「ATTACH」は、探索序盤の世代において使用できるノードを、入力ノードおよび定数番目より後のノードのみとし、探索中盤以降、使用不可ノードを解放するメカニズムである。解放メカニズム「MULTIPLE」は、探索序盤の世代において入力ノードおよびある倍数番目のノードのみを使用可とするメカニズムである。ノード解放手法「ATTACH」の例を図 3 に、「MULTIPLE」の例を図 4 に示す。本論文では、それらを両方とも導入したハイブリッドモデルについても実験を行った。

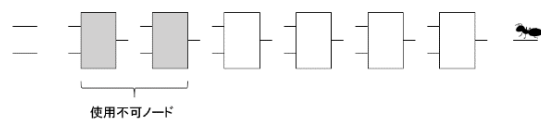


図 3 解放メカニズム ATTACH

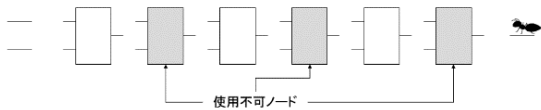


図 4 解放メカニズム MULTIPLE

提案手法の有効性を示すため、関数同定問題について従来の CAP と比較した。フェロモン濃度に基づき関数記号を割り当てる機能と 2 つの解放メカニズムをすべて導入した提案手法の CAP と従来の CAP との比較を図 5 に示す。探索序盤の約 1000 世代までは従来手法よりも悪い適応度を示しているが、ノードを解放した直後から従来手法を上回る適応度を示している。

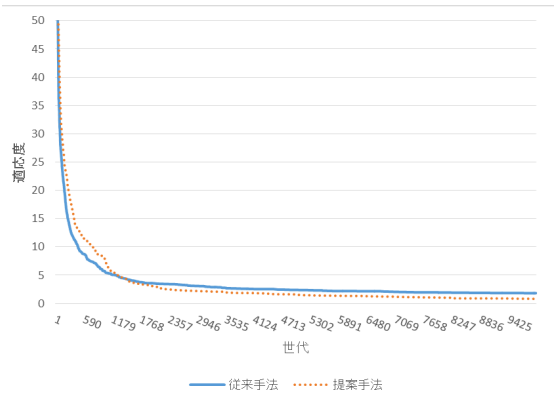


図 5 提案手法と従来手法 CAP の性能比較

実験の結果、解放メカニズム ATTACH、MULTIPLE のどちらのメカニズムも、導入する以前のものより良い適応度を示し、そのハイブリッドモデルではさらに良い適応度と標準偏差値を示した。

以上の結果化より、従来の CAP に比べて、より柔軟性があり、汎用性の高い手法の開発ができたと考える。

<引用文献>

[1] S. Shirakawa, S. Ogino, and T. Nagao: “Dynamic Ant Programming for Automatic Construction of Programs”, IEEJ Trans. on Electrical and Electronic Engineering, Vol.3, Issue 5, pp.540–548, 2008.
 [2] Akira Hara, Manabu Watanabe, and Tetsuyuki Takahama: “Cartesian Ant Programming”, Proc. of 2011 IEEE International Conference on Systems, Man, and Cybernetics (SMC2011), pp.3161–3166, 2011.

5. 主な発表論文等

(研究代表者、研究分担者及び連携研究者には下線)

[学会発表] (計 3 件)

① Jun-ichi Kushida, Natumi Mizobe, Akira Hara, and Tetsuyuki Takahama, Cartesian Ant Programming with Node Release Mechanism, 査読有, 2015 IEEE International Conference on Systems, Man and Cybernetics, 採録決定, 2015.
 ② 串田淳一, 溝部夏海, 原章, 高濱徹行, ノードへの関数記号の割り当て機能を備えたノード解放型 Cartesian Ant Programming, 査読無, 第 8 回進化計算学会研究会, pp.1–10, 2015.
 ③ Akira Hara, Jun-ichi Kushida, Keita Fukuhara, and Tetsuyuki Takahama, Cartesian Ant Programming with Adaptive Node Replacements, 査読有,

2014 IEEE 7th International Workshop
on Computational Intelligence and
Applications, pp.119-124, 2014.

6. 研究組織

(1) 研究代表者

原 章 (HARA, Akira)

広島市立大学・大学院情報科学研究科・
准教授

研究者番号：70347615