

Ant System の改善方法の提案 ~ グラフ点彩色問題への適用 ~

長原 正寿 (沼田 一道 助教授)

1. はじめに

近年, 組み合わせ最適化問題に適用できる, 蟻の振舞いを真似した進化的なアルゴリズムが提案された. そのアルゴリズムは Ant System[1]といい, メタヒューリスティックの一種に分類される. これまでに提案されたメタヒューリスティックの中にも, 自然界にアナロジーを持つものがいくつかある. 例をあげると, Simulated Annealing は焼き鈍し現象, Genetic Algorithm は遺伝学, Tabu Search は人間の記憶過程にアナロジーを持つ. Ant System についても, 近い将来, コンピュータプログラムの新しい枠組みを生成するはずである. しかし, 提案されたばかりで問題点もあるように思われる. それは解を求めるのに多くの時間を必要とすることである. 本研究では, Ant System を組み合わせ最適化問題の一つであるグラフ点彩色問題に適用し, 理解を深めると共にアルゴリズムの速度を少しでも向上させる改善案を提案することが目的である. そのために, Delphi によりソフトを作成し数値実験を行い改善案の効果を調べる.

2. グラフ点彩色問題

グラフ G とは, 頂点と呼ばれる要素の有限集合 $V = \{v_1, \dots, v_n\}$ と, それらをつなぐ枝と呼ばれる集合 E からなり, $G = (V, E)$ と記される. 枝 $e = (v_i, v_j)$ が存在するとき, 頂点 v_i, v_j は隣接するといいい, 頂点 v_i, v_j は枝 e に接続するという. 安定集合とは, 互いに隣接していない頂点の集合である.

グラフ点彩色問題における頂点集合 V と枝集合 E から成るグラフ $G = (V, E)$ の q 彩色とは, E が頂点 v_i と v_j を結ぶ枝 $e = (v_i, v_j)$ を含むときにはいつでも, $c(v_i) \neq c(v_j)$ (つまり, 隣接点同士には異なる色を割り当てる) となるよう, 各頂点に q 色のうちの 1 色を割り当てる事 ($c: V \rightarrow \{1, 2, \dots, q\}$) である. q 彩色が存在するような q の最小の値を G の 染色数とよび, $\chi(G)$ と書く. 最適彩色とは, 正確に $\chi(G)$ 色を使った物である.

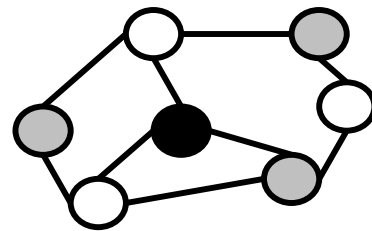


図1 グラフ点彩色問題

3. Ant System の概要

Ant System とは, 蟻のコロニーが近く of 食物リソースを探す振舞いを真似たアルゴリズムである. 蟻はフェロモンを巣から食物までの道につけることにより, 自然に巣から一番近い食物を見つけることができる.

この探索原理を一般的な組み合わせ最適化問題に適用したものが Ant System である. Ant System をグラフの点彩色問題に適用したものを例に説明する.

各々の蟻の役割は, 構成的な方法でグラフに色付けすることである. そこで用いられる構成的な方法は何でも構わない. ただし, 本研究で用いた構成的な方法は RFL (Recursive Largest First) であり, これはグラフの頂点を $\chi(G)$ 個の安定集合に分けるものである. 以前の解構築の間に蓄

積された経験は $n \times n$ 行列 M に記憶され、定期的に変更される。二つの隣接していない頂点 v_r, v_s が与えられたときの M の M_{rs} という値は、 v_r と v_s に同じ色を与えることによって得られる彩色の質(目的関数値)に比例している。それゆえに、 M_{rs} は頂点 v_r と v_s の間に存在するフェロモンの跡に対応する。フェロモンの跡が多ければ頂点 v_r と v_s とに同じ彩色がされる確率が高くなる。最初に M の中のすべての値が 1 にセットされる。 M に貯えられたフェロモンの跡は時間が経つにつれ次第に消えていく。蒸発の程度は蒸発係数 $(1 - \rho)$ によって示される。 s_1, \dots, s_{nants} を各サイクルの最後に得られた彩色とする。また、 S_{rs} を v_r と v_s が同じ色をもつ解の集合 $(\{s_1, \dots, s_{nants}\})$ の部分集合とする。最後に、 q_a を彩色 s_a ($1 \leq a \leq nants$) で用いられる色数とする。 M の値 M_{rs} は、以下のように更新される。

$$M_{rs} = \rho \cdot M_{rs} + \sum_{s_a \in S_{rs}} \frac{1}{q_a} \cdot \dots \quad (1)$$

そのような進化アルゴリズムは ANTCOL と呼ばれる。ANTCOL の簡略化したアルゴリズムを表 1 に示す。

表 1 The algorithm ANTCOL

$M_{rs} := 1 \quad \forall [v_s, v_r] \notin E$; (Initialization of the trail between pairs of non adjacent vertices)
 $f^\circ := \infty$; (number of colours in the best colouring reached so far)
For $ncycles := 1$ to $ncycles_{max}$ do
 $\Delta M_{rs} := 0; \forall [v_s, v_r] \notin E$;
 For $a := 1$ to $nants$ do
 colour the graph by means of a constructive method;
 let $s = (V_1, V_2, \dots, V_q)$ be the colouring obtained
 If $q < f^\circ$ then $s^\circ := (V_1, V_2, \dots, V_q); f^\circ := q$;
 $\Delta M_{rs} := \Delta M_{rs} + \frac{1}{q} \quad \forall [v_s, v_r] \notin E, \{v_s, v_r\} \subseteq V_j, j = 1, \dots, q$;
 end;
 $M_{rs} := \rho \cdot M_{rs} + \Delta M_{rs} \quad \forall [v_s, v_r] \notin E$;
end;

4. 数値実験

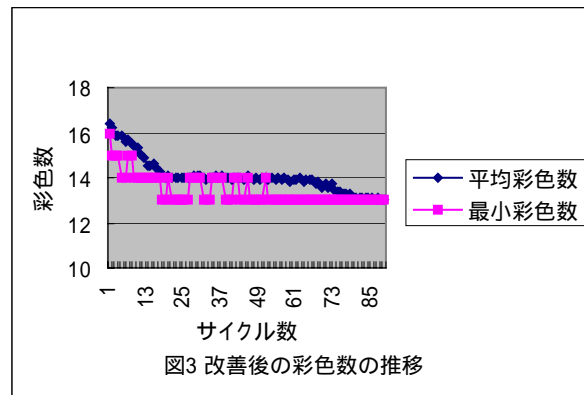
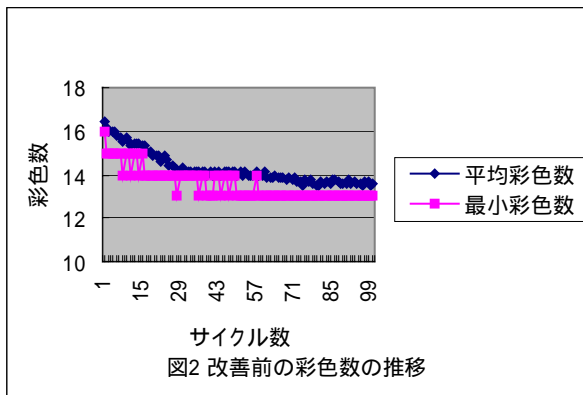
本研究では、DIMACS のベンチマーク問題の中から、染色数 k が知られている問題を用い、Ant System の精度を確かめる。そして、表 2 に、アルゴリズムの改善前と改善後の比較した表を示す。上段には、染色数が初めて得られた時のサイクル数を示した。下段には、解が収束した値、括弧内にはそのサイクル数を示す。実験は、最大 100 サイクル繰り返した。上段の括弧内の - は、そのサイクル内で染色数が求まらなかったことを示し、下段の - は収束せず振動したことを示す。

図 2、図 3 には、改善案の効果がもっとも現れた改善前と改善後の比較(頂点数 100、枝の存在確率 0.4、蟻の数 40)を示す。平均彩色数とは、各サイクルで求めた許容彩色の彩色数の平均値である。最小彩色数とは、各サイクルで求めた許容彩色の中で一番彩色数が少ないものであ

る .

表 2 改善前と改善後の変化の比較

頂点数	枝の存在確率	染色数	オリジナルの ANTCOL			改善案の ANTCOL		
			蟻の数			蟻の数		
			20	30	40	20	30	40
30	0.4	6	1	1	1	1	1	1
			6(62)	6(61)	6(66)	6(40)	6(43)	6(40)
30	0.5	7	1	1	1	1	1	1
			7(8)	7(17)	7(24)	7(8)	7(12)	7(14)
30	0.6	8	3	1	1	3	1	1
			8(9)	8(8)	8(12)	8(9)	8(9)	8(11)
50	0.4	9	1	1	1	1	1	1
			9(18)	9(17)	9(16)	9(20)	9(17)	9(27)
50	0.5	10	3	3	6	4	10	3
			10(13)	10(26)	10(25)	10(16)	10(28)	11(11)
50	0.6	11	5	5	3	5	4	3
			11(11)	11(18)	11(15)	11(15)	11(14)	11(20)
100	0.4	13	58	37	28	33	17	18
			14(55)	14(40)	13(-)	14(39)	14(49)	13(83)
100	0.5	16	13	19	17	27	18	16
			16(39)	16(52)	17(25)	16(41)	16(32)	16(50)
100	0.6	19		20	48		41	12
				20(28)	19(76)	20(34)	19(61)	19(41)



6. 考察

改善案は，Ant System の欠点である速度の向上を目的としたものである．しかし，あくまで準最適解の求まる速度が確率的に向上するだけであり，全ての問題について改善案の方が早く最適解が求まるとはいえない．これは，表 2 の結果からもわかる．それでは，図 2、図 3 のように改善案の効果が現れやすい問題とはどのような問題であろうか．それは局所最適解の中で最適解の占める割合が小さい問題であるように思われる．グラフ点彩色問題を例に説明すると、頂点数が多いグラフほど、頂点数が同じであれば、枝の存在確率が小さいものほど効果が現れやすい．一番効果が現れた図 2，図 3 の例はこれを一番満たしている．

7.まとめ

Ant System は多くのパラメータを用いるため、そのパラメータによって解の求まる速度、解の質は大きく変化する。本研究では(1)式の部分を変更したが、まだまだ改善する余地は残っている。Ant System は、その並列性からコンピュータを多数用いた並列的実行には非常に合っているように思われる。そして、様々な割り当て型の組み合わせ最適化問題を解くためのアルゴリズムを少ない努力で得ることができるのでとても有用である。今後の課題は、さらに実験を繰り返しアルゴリズムの速度を向上する案を見つけることと共に、解の質を更にアップすることである。

【参考文献】

[1]D.Costa and A. Hertz : “Ants can colour graphs” , Journal of the Operational Research Society Vol.48 , 295-305 (1997).

[2]室田 一雄 (編): 離散構造アルゴリズム , 近代科学社, 1995.

[3]恵羅 博 , 土屋 守正: グラフ理論, 産業図書, 1996.

[4]角田 充弘: “グラフ彩色問題のスケジューリングへの適用”

東京理科大学卒業論文 (1997年度).