

生態ピラミッドの概念を取り入れた遺伝的アルゴリズムの提案

広島市立大学情報科学部 上土井 陽子 (Yoko Kamidoi)
広島大学工学部 岸本 善久 (Yoshihisa Kishimoto)
広島大学工学部 若林 真一 (Shin'ichi Wakabayashi)

あらまし 最適解を求めることが困難な組合せ最適化問題に対し、メタヒューリスティック解法の一つとして遺伝的アルゴリズムが注目されている。遺伝的アルゴリズムでは対象とする問題の複雑さが増加するのに従い、人口数などの入力パラメータ値に対する解の質の依存性が大きくなるため、パラメータ値の選定が重要となってくる。また、従来の遺伝的アルゴリズムにおいては人口の大半が類似した解になった場合は人口における多様性が失われ、その結果、更に交配を繰り返しても良い解を得ることが困難になるという問題点もある。本稿では従来の遺伝的アルゴリズムにおける上述した問題点を解消すると共に、従来手法より優れた解を生成する遺伝的アルゴリズムの一手法を提案する。具体的には、生物界における生態ピラミッドの概念を遺伝的アルゴリズムに導入し、アルゴリズムの実行中に階層的に情報を伝達させる階層遺伝的アルゴリズムを提案すると共に、提案アルゴリズムの有効性を実験的に検証する。

1 まえがき

最適解を求めることが困難な組合せ最適化問題に対する近似解法として生物進化のプロセスの概念に基づいた遺伝的アルゴリズムが注目されている [2, 4]。遺伝的アルゴリズムは大局的に解空間を探索するため、局所的な探索による通常の発見的手法と比較して良質な解を求めることが知られている。しかし、対象とする問題の複雑さが増加するのに従い、遺伝的アルゴリズムの人口数や突然変異率等の入力パラメータ値に対する解の質の依存性が大きくなるため、遺伝的アルゴリズムにおいてはパラメータ値の選定が非常に重要であるが、それらの設定方法についての一般的な手法は知られていないという問題点がある。また、従来の遺伝的アルゴリズムにおいては人口の大半が類似した解になった場合は人口における解の多様性が失われ、その結果、更に交配を繰り返しても良い解を得ることが困難になるという問題点もある。

本稿では従来の遺伝的アルゴリズムにおける上述の問題点を解消するとともに、従来手法より優れた解を生成する遺伝的アルゴリズムを開発することを目的とする。具体的には、生物界における生態ピラミッドの概念を遺伝的アルゴリズムに導入し、アルゴリズムの実行中に階層的に情報を伝達させることによって対象とする問題に対するパラメータ設定を自動的に行う階層遺伝的アルゴリズムを提案する。生物進化において種の多様性を増大させる一つの要因として刈り込み者の概念の導入が知られている。すなわち、ある種が空間を独占している生態系に刈り込み者を導入すると一部の種が独占的に君臨する能力が制限され、他の種のために空間が解放されることから、適度に刈

り込まれた系は多種多様となる。すなわち、生態のピラミッドに一つの新しい段階を導入することはそのすぐ下位の段階を多様化する結果となることが多い。本稿ではこの概念を遺伝的アルゴリズムに導入する。また、提案アルゴリズムを複数のワークステーションからなる分散環境上に実現し、VLSI レイアウト設計において重要な応用を持つ組合せ最適化問題であるハイパーグラフ分割問題 [11] に適用することにより提案アルゴリズムを評価する。

2 一般的な遺伝的アルゴリズムの概要

遺伝的アルゴリズムは Holland により以下の 2 つの目的を持って提案された。(1) 自然界における適応プロセスを理解し、説明する。(2) 自然界におけるいくつかの重要な機構を備えた人工システムソフトウェアを設計する。すなわち、任意の環境に適応する能力を持つ一般的なプログラムや機構の生成に必要な理論や手続きを開発することを目的として遺伝的アルゴリズムは開発された。遺伝的アルゴリズムは自然システム科学と人工システム科学の両方においていくつかの重要な成果を挙げたことが知られている。その一つが組合せ最適化問題への適用の成功である [4]。

一般に遺伝的アルゴリズムは複数の許容解を保持しながら、それらの間の交配を繰り返すことにより最適解に近い許容解を得ることを目的とした手法である [2, 4]。遺伝的アルゴリズムは大局的に解空間を探索するため、局所的な探索による通常の発見的手法と比較して良質な解を求めることが知られている。遺伝的アルゴリズムは通常、4 つの段階から成っている。第 1 段階では複数の許容解により人口を構成する。第 2 段階では各許容解を評価する。第 3 段階では人口を構成している複数の解を選択する。第 4 段階では選択した複数解に対して交配や突然変異などの遺伝学的操作を行ない、新しい人口を構成する候補となる新たな許容解を生成する。以上の 4 つの段階を与えられた条件を満たすまで、または十分良質な解が得られるまで繰り返す。一般に許容解は遺伝学的操作に適した文字列で表現され、文字列に対し交配や突然変異に相当する操作が行われる。また、ある人口から新しい人口を構成するまでの一連の手続きを世代と呼ぶ。各世代により構成される人口の構成方法は前世代までに生成された解を含むものや新しく生成された解のみによるものなど様々である。遺伝的アルゴリズムは複数の許容解を扱っていることより本質的に並列探索を行っていることになる。また、数理計画法や他の解析的手法と異なりランダム探索に基づいているので解空間の広い組合せ最適化問題における近似最適解を求める場合に有効である。

3 生態ピラミッドの概念を取り入れた遺伝的アルゴリズム

3.1 遺伝的アルゴリズムへの新たな導入概念

本稿において遺伝的アルゴリズムへの新たな導入を提案する 2 つの概念を以下に示す。

[刈り込み者の概念]

少数の種が空間を独占している系において、刈り込み者が導入された場合、一部の種が独占的に君臨する能力が制限され、他の種のために空間が開放される結果となる。適度に刈り込まれた系は

最高度に多種多様であり，種数は多いが個々の個体数は少ない [5]．このような刈り込み者の概念を遺伝的アルゴリズムに取り入れ，解の多様性を保持することを目的とする．遺伝的アルゴリズムにおいては人口数が小さい試行において解が急速に互いに似かよってしまい十分な改良が得られていない解を出力する傾向が強い．これは早い世代で得られる解が同じ親または似かよった経歴を持つ親から生成される可能性が大きいためと考えられる．そこで解の多様性の保持を目的として，刈り込み者の概念を導入する．すなわち，保持している解のいくつかを削除し，かわりに新しい解を人口に加える．入れ替える解の数やどのような解を入れ替えるかは対象としている解集合の成熟度によって決定する．多様性の保持を目的として，遺伝的アルゴリズムにより改良が十分には行われていない解やランダムに生成した初期解などを入れ替える解として用いる．

[生態ピラミッドの概念]

生物界においては刈り込み者の概念の導入により，一層多様な生産者たちのための空間が作りだされ，この増大した多様性をもっと特殊化した刈り込み者たちの進化を可能にする．生態ピラミッドはより下位の段階ではたくさんの種を付け加え，その頂点では動物食種の新しい段階を付け加えるという具合にして，両方向に向かって爆発的に多様化したと考えられている [5]．刈り込み者の概念を階層的に構築して行くことにより，さらなる多様化を生じる結果となることが生態ピラミッドの概念の利点である．遺伝的アルゴリズムにおいては，生態ピラミッドの概念を2つの目的を達成するために導入する．1つは刈り込み者の概念と同時に導入することにより，刈り込み者自身の多様化，及び，それによる下位階層のさらなる多様化を促すことである．他の目的は各階層において実行される遺伝的アルゴリズムの解集合の成熟度を予測し，問題の複雑さに応じたパラメータ値の自動選定を行う機構を構築することである．具体的には，各階層での遺伝的アルゴリズムにより求められる解の目的関数の値の差により，各階層での遺伝的アルゴリズムにより生成された解集合の成熟度を判定し，十分な成熟が得られていない場合には最上位階層として新たな階層を付け加える．

3.2 提案アルゴリズムの概要

提案アルゴリズムにおいては，階層的に情報を伝達できる階層構造を導入することにより生態ピラミッドを模倣する．生物界における生態ピラミッドの生物の捕食関係における強弱を遺伝的アルゴリズムにおける人口数の大小と考える．1組のパラメータ値が具体的に与えられた遺伝的アルゴリズムをインスタンスと呼ぶ．提案アルゴリズムにおいてはパラメータ値の異なる複数のインスタンスを並列に実行させることにより，生物界における生態ピラミッドを模倣し，対象とする組合せ最適化問題に対する適切なパラメータ値の自動設定，及び，解の多様性を保持する．提案アルゴリズムにおいては生物界における生態ピラミッドでの刈り込み者をパラメータ値の異なるインスタンスの解とみなし，下位階層におけるインスタンスの解と刈り込み者となっている解を比較する．解の比較によって下位階層のインスタンスにおける解集合の成熟度を推定し，十分な解探索が行われたと判定した場合は最良解を出力して停止する．

提案アルゴリズムにおいては複数のプロセス間を木構造のネットワークで接続するものとする．各プロセスは一般的なプロセス，葉プロセス，根プロセスの3種類のプロセスの状態に類別できる

ものとする。一般的なプロセスは2つの子プロセスと1つの親プロセスに連結している。葉プロセスは1つの親プロセスのみと連結し、根プロセスは親プロセスを持たないものとする。親プロセスと子プロセスでは遺伝的アルゴリズムのパラメータ値、特に人口数において強弱の関係、つまり、親プロセスでの人口数が子プロセスの人口数よりも大きいなどの関係を満たすものとする。世代数がある一定数繰り返して得られた解を互いに通信し、比較して各プロセスの成熟度を推定し、成熟度が十分でないと判定した場合、根プロセスがもう1つ上位のプロセスを生成するものとする。与えられた階層数の上限以上の根プロセスの生成の命令が出た場合、及び、各プロセスの成熟度が十分であると判定された場合にはそれまでに得られた最良解を出力し、停止する。以下では各プロセスが実行するアルゴリズムの概要を示す(図1)。ここで対象とする問題はコスト最小化問題と仮定する。

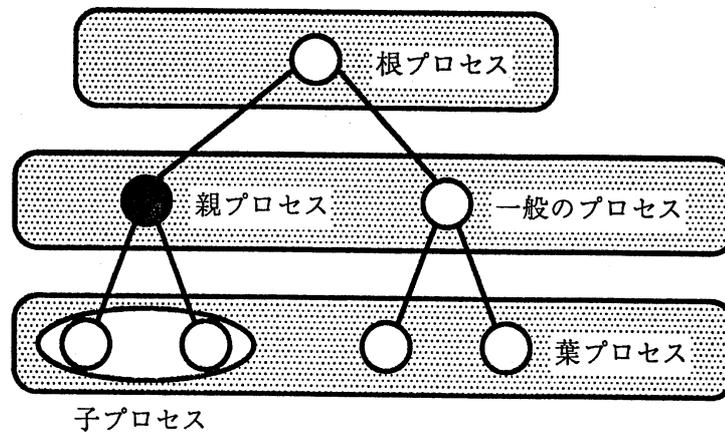


図 1: 階層遺伝的アルゴリズム

[アルゴリズムの概要]

1. 人口数比 p , 世代数 r , 許容差 s を入力する。
2. プロセス $\text{Process}(\text{根プロセス}, 1, p, r, s)$ を生成する。
3. 根プロセスが終了したなら、最良解を出力し停止する。

[プロセス $\text{Process}(\text{プロセスの状態}, rank, p, r, s)$]

1. $rank \geq 3$ であれば、2つまたは1つのプロセス $\text{Process}(\text{一般プロセス}, rank - 1, p, r, s)$ を生成する。 $rank = 2$ であればプロセス $\text{Process}(\text{葉プロセス}, 1, p, r, s)$ を生成する。
2. $rank * p$ の人口数, 世代数 r を持つ遺伝的アルゴリズムを試行する。
3. 葉プロセスでないなら、自分の子プロセスに1. で得られた平均解を伝達する。

4. 根プロセスでないなら、最良解を親プロセスに通信する。親プロセスより停止命令が出ているなら子プロセスに停止命令を出し、子プロセスが停止したなら自分も停止する。また子プロセスを持たない場合は停止する。停止命令が出ていないなら、人口を構成している解において親の平均解より悪い解があれば子プロセスにより生成された解と入れ替える。葉プロセスならランダムに生成された初期解などに入れ替え、2. に戻る。
5. 根プロセスであるなら、子プロセスの最良解のコスト $Ccost$ と自分の最良解のコスト $Pcost$ を比較し、 $|Ccost - Pcost| / \max(Ccost, Pcost) \leq s/100$ であるなら、子プロセスに停止命令を出し子プロセスが停止したなら最良解を出力し終了する。そうでなければ、プロセス $Process(\text{根プロセス}, rank + 1, p, r, s)$ を生成し、2. に戻る。

3.3 ハイパーグラフ分割問題への適用

ハイパーグラフの分割問題はサイズの制約条件を満たすカット数最小のハイパーグラフの分割を求める問題である。ハイパーグラフ分割問題は VLSI レイアウト設計等の応用分野で重要な問題である [11]。以下にハイパーグラフ分割問題 WHB (Weighted Hypergraph Bisection Problem) を定式化する。

[問題 WHB]

[入力] ハイパーグラフ $H = (V, E)$

[出力] カット数が最小な V の 2 分割 (V_L, V_R)

[制約条件] V_L に属する節点の重みの総和と V_R に属する節点の重みの総和の差が最大節点重み、または、 $\beta \times W$ 以下である。ここで β は 0.1 以下の定数、 W は節点の重みの総和とする。

ハイパーグラフ分割問題は一般に NP 困難なことが知られており [3]、現在までにいくつかのヒューリスティックアルゴリズムが提案されている [1, 6, 9, 10]。しかし、提案されている手法はいずれも目的関数が減少する場合のみ解の改良を行っているため、局所最適解に陥り易いという問題点がある。

著者らはハイパーグラフ分割問題に対する遺伝的アルゴリズムに基づくヒューリスティックアルゴリズム GWHB を既に提案している [8]。この手法は遺伝的アルゴリズムと高速ヒューリスティックアルゴリズム WHB [7] を組み合わせた手法である。アルゴリズム GWHB では解の初期化、交配においてヒューリスティックアルゴリズムを用いており、大規模な入力に対しても実用的な計算時間で良質な解を求めることが示されている。ハイパーグラフ分割問題 WHB に対する遺伝的アルゴリズム GWHB を以下に示す。

[アルゴリズム GWHB]

1. 人口数 p , 世代数 R , 悪い解の数 m を入力する。

2. アルゴリズム WHB を適用し, p 個の初期解を求め, 解集合 P を作成する.
3. 解集合 P の全ての非順序対に対しアルゴリズム WHB を適用し交配する.
4. 3. で求められた $p(p-1)/2$ の解より $p-m$ 個の目的関数の良い解と残った解からランダムに m 個選ぶ. 選ばれた p 個の解により集合 P を更新し, 次世代を構成する.
5. 終了条件を満たしていれば終了. そうでなければ 3. へ.

紙面の都合上, 解の交配手法などのアルゴリズムの詳細は文献 [8] に譲り, ここでは省略する. 本稿では 3.2 で概要を示した階層遺伝的アルゴリズムにアルゴリズム GWHB を組み込み, 問題 WHB に対して階層遺伝的アルゴリズムを適用する.

4 実験結果

提案手法を LUNA2010, SPARC station 1000, 10, 2, IPX の 5 台のワークステーションより構成される分散システム上に C 言語を用いて実現した. 通信方法としてはソケットを用いた. 実験データとして表 1 に示すベンチマークデータを使用した. 実験において遺伝的アルゴリズムの各パラメータは, 人口数比 p を 5, 世代数 r を 2 とした. また, プロセス数の上限は 7 とした. 比較手法として人口数 5, 10, 15 の場合のアルゴリズム GWHB を用いた. 実験においては各手法共 20 回の試行を行ない, 解の最小値と平均値を求めた. アルゴリズム GWHB の実験結果を表 2 に示す.

提案アルゴリズムは許容差を 40, 35, 30, 25, 20, 15 の場合で実験を行なった. ここで許容差とは根プロセスの最良解と子プロセスの最良解の差に対する相対的な値である (3.2 で説明した s の値). 提案アルゴリズムの実験結果を表 3,4,5 に示す. 提案アルゴリズムはパラメータの調整を行なうことなく GWHB と同等の解を求めていることが分かる.

表 1: 実験データ

データ	節点数	枝数
Prim1	833	904
Prim2	3014	3029
Industry2	12142	12924

表 2: 実験結果

データ	アルゴリズム GWHB					
	人口数 = 5		人口数 = 10		人口数 = 15	
	min.	ave.	min.	ave.	min.	ave.
Prim1	50	70	47	60	50	58
Prim2	174	272	174	238	164	195
Industry2	326	505	228	326	217	271

表 3: 実験結果 (許容差 = 40, 35)

データ	許容差 = 40		許容差 = 35	
	min.	ave.	min.	ave.
Prim1	69	88	58	78
Prim2	202	291	180	266
Industry2	504	830	526	806

表 4: 実験結果 (許容差 = 30, 25)

データ	許容差 = 30		許容差 = 25	
	min.	ave.	min.	ave.
Prim1	61	83	56	74
Prim2	212	240	178	234
Industry2	527	761	464	788

表 5: 実験結果 (許容差 = 20, 15)

データ	許容差 = 20		許容差 = 15	
	min.	ave.	min.	ave.
Prim1	57	81	58	72
Prim2	190	271	199	272
Industry2	515	739	363	647

5 まとめ

本稿ではパラメータ値の自動選定, 及び, 人口を構成する解の多様性の保持を目的として生態ピラミッドの概念を遺伝的アルゴリズムに導入し, アルゴリズムの実行中に階層的に情報を伝達させる階層遺伝的アルゴリズムを提案した. また, 提案アルゴリズムを複数のワークステーションからなる分散環境上を実現し, VLSI レイアウト設計において重要な応用をもつハイパーグラフ分割問題に対して提案アルゴリズムを適用し, 評価した.

今後の課題としては木構造以外のプロセス間の通信トポロジを用いたアルゴリズムの開発, 解の多様性の評価方法の考察などが挙げられる.

謝辞

日頃, 御指導を賜っております広島大学教授吉田典可先生に謹んで感謝の意を表します. また, 御議論をいただいた小出哲士先生, 並びに実験の手伝いをして下さった広島大学学生松田憲治氏に感謝いたします. 本研究の一部は文部省科学研究補助金試験研究(B)(2)(課題番号 06558042)による.

参考文献

- [1] C. M. Fiduccia *and* R. M. Mattheyses : "A linear-time heuristic for improving network partitions," Proc. 19th ACM/IEEE Design Automation Conf., pp.175-181 (1982).
- [2] J. L. R. Filho, P. C. Treleaven *and* C. Alippi : "Genetic-algorithm programming environments," IEEE Computer Magazine, Vol. 27, No. 6, pp. 28-43 (1994).
- [3] M. R. Garey *and* D. S. Johnson : "Computers and Intractability : A Guide to the Theory of NP-completeness," W. H. Freeman (1979).
- [4] D. E. Goldberg : "Genetic Algorithms in Search, Optimization, and Machine Learning," Addison-Wesley (1989).
- [5] S. J. Gould 著/浦本 昌紀訳 : "ダーウィン以来 (上)", 早川書房 (1984).
- [6] A. B. Kahng : "Fast hypergraph partition," Proc. 26th ACM/IEEE Design Automation Conf., pp. 762-766 (1989).
- [7] Y. Kamidoi, S. Wakabayashi *and* N. Yoshida : "An efficient hypergraph bisection algorithm for partitioning VLSI circuits," IEICE Trans. Fundamentals, Vol. E75-A, No. 10, pp. 1272-1279 (1992).
- [8] Y. Kamidoi, S. Wakabayashi *and* N. Yoshida : "An efficient GA hybrid for hypergraph bisection with application to VLSI placement," Proc. IEEE Asia-Pacific Conf. on Circuits and Systems 1992, pp. 369-401 (1992).
- [9] B. W. Kernighan *and* S. Lin : "An efficient heuristic procedure for partitioning graphs," Bell System Technical Journal, 49(2), pp. 291-307 (1970).
- [10] B. Krishnamurthy : "An improved min-cut algorithm for partitioning VLSI networks," IEEE Trans. on Computers, Vol. C-33, No. 5, pp. 438-446 (1984).
- [11] T. Lengauer : "Combinatorial Algorithms for Integrated Circuit Layout," Wiley (1990).