

ラマルク進化を導入した LLT-GA の構築

吉井 伸一郎[†] 鈴木 恵 二^{††} 嘉数 侑 昇^{††}

本論文では、ラマルク型の Lookup-Table 型遺伝的アルゴリズムを提案している。遺伝的アルゴリズムは、自然選択に主眼をおいたダーウィンの進化理論を模倣した進化的モデルとして幅広く用いられている。一方、ラマルクによる進化説は、個体の環境への自己適応を目的として獲得された形質の遺伝をその理論の中心的要素とした進化説である。適応的な戦略の獲得を目的とした系においては、個体の環境での適応によって得られた情報を利用することの重要性から、ラマルク進化のインプリメントが有効視されており、多くの進化、学習アルゴリズムの補強が期待される。本論文では、エコシステム的な環境において適応的振る舞いを実現する Lookup-Table 型遺伝的アルゴリズム (LT-GA) にラマルク進化の概念を導入した Lamarckian LT-GA (LLT-GA) の構築を試みている。さらに、この LLT-GA の適応能力を明示的に評価するために、最適化問題へと適用しており、優れた適応能力を示すことを確認している。

A Study on Lamarckian Lookup-Table Type Genetic Algorithms

SHINICHIRO YOSHII,[†] KEIJI SUZUKI ^{††} and YUKINORI KAKAZU ^{††}

In this paper, we propose Lamarckian Lookup-Table type Genetic Algorithms (LLT-GA). GA is the evolutionary algorithm derived from the scheme of Darwinism marked by natural selection. On the other hand, the evolutionary theory advocated by Lamarck focuses on the inheritance of characteristics acquired for the purpose of self-adaptation to the environment. In the domains with the object of acquisition of adaptive strategies, it is supposed to be effective to implement Lamarckism, because it is important to use the information of experiences through adaptation to the environment. Lamarckian evolution is expected to augment the power of many kinds of evolving or learning algorithms. In this paper, we construct LLT-GA by introducing Lamarckian evolution into Lookup-Table type Genetic Algorithms (LT-GA), which realize adaptive behaviors in environments with interactive components. We apply LLT-GA to optimization problems in order to inspect its ability objectively, and show its superior adaptation ability through the numerical experiments.

1. はじめに

近年、生物の有する高度な情報処理機構をもとにした人工システムに関する研究が盛んである。なかでも、生物の進化の過程に着想を得た遺伝的アルゴリズム^{1),2)}は、ダーウィンの進化理論³⁾を模倣したアルゴリズムであり、淘汰や交叉、突然変異といった遺伝的オペレータを用いる、多点情報を利用した確率的探索法の一つである。そして、最適化問題においては強力な探索手法として、また、人工生命の研究領域では進化的モデルの基本的な演算機構として幅広く用いられており、適用する問題に応じて様々なアプローチが提案

されている。

特に、人工的な生態系をシミュレートするエコシステム^{4),5)}的環境のような、動的に変動する環境での逐次決定問題を対象とした場合に、個体の戦略を表すオートマトンとして行動選択テーブル (Lookup-Table; LT)^{6),7)}がある。そして、この LT を用いて遺伝子を構成することで遺伝的アルゴリズムによる適応的な戦略の獲得も成されている^{8)~11)}。本論文では、このような遺伝的アルゴリズムを Lookup-Table 型遺伝的アルゴリズム (LT-GA) と呼ぶものとする。

しかし、このような環境において、より適応的な戦略の獲得を考えた場合、重要視すべきなのは環境に期待する個体の振る舞いに対して自己を適応させるような情報である。したがって、環境からのフィードバックとして得られる情報が個体の環境での評価値のみである遺伝的アルゴリズムは、動的な環境に対する能力が乏しく、やや力不足の感は否めない。この点に関し

[†] 北海道大学工学部 情報工学科

Division of Information Engineering, Faculty of Engineering, Hokkaido University

^{††} 北海道大学工学部 精密工学科

Department of Precision Engineering, Faculty of Engineering, Hokkaido University

て、個体の環境への自己適応を目的として獲得された情報を子孫に直接遺伝させるラマルキズム³⁾が有効であると考えられる^{12),13)}。

そこで、本論文では、エコシステムの環境において適応的振る舞いを実現する LT-GA にラマルク進化の概念を導入した Lamarckian LT-GA (LLT-GA) を提案する。一般的に、遺伝的アルゴリズムは、交叉を主要な原動力とした大域サンプリングを中心とした手法であり、局所的な探索は突然変異や集団の収束に委ねられる。種の適応を扱う遺伝的アルゴリズムに個体の環境への自己適応を目的としたラマルク進化を導入することで局所的な探索が可能となり、より優れた戦略が獲得される結果、個体の環境に対する適応能力が向上することが期待される。

ここでは特に、エコシステムの環境を前提として LT-GA を取り上げているが、このような環境は多数の個体間の複雑なインタラクションに起因する動的な環境であるがゆえに、ラマルク進化のもたらす効果を客観的に評価するのは一般的に困難である。また、上記のような人工生命的立場を離れ、具体性のある工学的問題を前提にした場合、現時点ではベンチマークとしてふさわしく、かつ動的な要素を持った問題が見当たらない。ゆえに、本論文では、客観的な評価を可能にする環境として次の二つの問題を取り上げる。すなわち、個体が、点在する巣を最短経路で巡回するというアナログを用いることで定義される巡回セールスマン問題 (TSP) と、個体の過去に決定した振る舞いにより将来の制約条件が変化する環境を考え、その具体例としてジョブショップスケジューリング問題である。

特に動的環境においてその効果が期待されるラマルク進化を評価するには、これらの問題は対極に位置するものである。しかし、本論文では、まず客観的な指標を与える環境としてこのような静的な問題を採用することで、提案手法の明示的な評価を行い、計算機実験によって LLT-GA の有効性について議論する。

2. ラマルク進化

ラマルクによる進化説は、進化に関する体系的な理論として、19 世紀初頭に発表されたものである。ラマルク進化はいくつかの説から成っているが、その中心的理論は獲得された形質の遺伝に関するものである。

図 1 および図 2 に、ダーウィン進化とラマルク進化における形質の遺伝という情報の処理プロセスを示す。遺伝的アルゴリズムのもとになるダーウィン進化を形質という情報の流れとして捉えると、図中の矢印の方向に示されるように、多様化を目的としてランダ

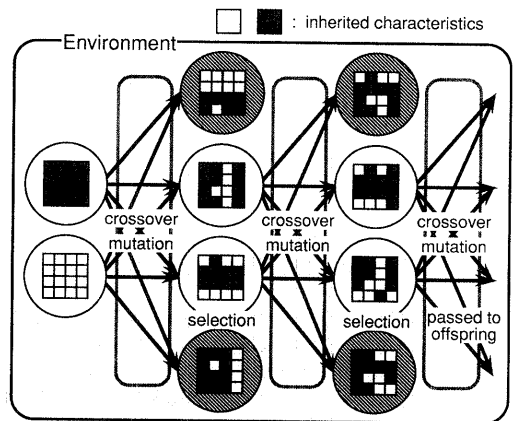


図 1 ダーウィン進化における情報処理機構
Fig. 1 The scheme of Darwinian evolution.

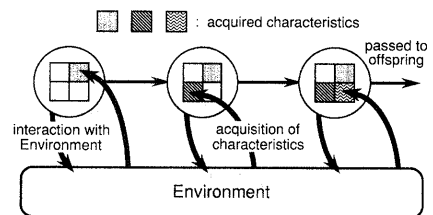


図 2 ラマルク進化における情報処理機構
Fig. 2 The scheme of Lamarckian evolution.

ムな方向へと情報が伝搬される。そして、自然淘汰により、さまざまな性質を持つ個体の中から環境に対して有効な形質を持つ個体が選択され、この過程を何世代も繰り返すことで進化が生じる。一方、ラマルク進化では、各個体は環境とのインタラクションを通して、図中の矢印の先に示されるような、環境にとって有効な何らかの形質を獲得する。この獲得された形質は、個体の環境での生存を優位にする情報であり、これを子孫に遺伝させることで進化が進行していくものと考えられる。このように、生物の進化を環境に対する最適化を目的とした単なる情報処理機構として眺めると、ダーウィン進化に比べてラマルク進化は個体の環境への自己適応を目的とした積極的な進化であるといえる。

しかし、獲得形質の遺伝に関する理論は、生物学における近代の研究が、この理論の論点を支持しなかったために、多くの信奉者を失った。このような理由のために、遺伝的過程をシミュレートするアルゴリズムにおいてラマルク進化のインプリメントが積極的に行われないものと思われるが、ラマルク進化を用いることにより、これまでの進化アルゴリズムが補強されることが期待できる^{12),13)}。また、機械学習の研究分野に

においても、ラマルク進化的なオペレータを採用することで学習が改善される可能性が示されている^{14),15)}。

ラマルク進化の導入には様々なインプリメントが考えられるが、その特徴は表現型での適応によって得られた情報を遺伝子型へとフィードバックし、獲得された情報を次の世代へと遺伝させることにある。Grefenstette らによる SAMUEL^{14),16)}では、各ルールが環境との相互作用により、自らの戦略を変更するようなオペレータを採用することによりラマルク進化を実現している。

本論文におけるラマルク進化のインプリメントについては、LT-GA の最適化問題への適用方法の後に詳述する。

3. 巡回セールスマン問題の場合における議論

3.1 LT-GA

多数の個体からなるエコシステム的な環境において、適応的な振る舞いを実現する手法である行動選択テーブル LT には、環境や他の個体とのインタラクションによる入力に対して、どのようなアクションを行うかを決定するルールが記述されている。刺激-反応系における LT の振る舞いには、その戦略の複雑性を評価する数学的な解析も行われている⁹⁾。本論文では、入力状態集合と出力状態集合のマッピングを確率的に保存する出力選択確率ベクトル集合からなるオートマトンとして LT を記述する。

LLT-GA の評価を行うにあたって、まずここでは、個体が点在する巢を最短経路で巡回することを目的とした環境として、巡回セールスマン問題(TSP)¹⁷⁾を考える。TSP とは、都市間の巡回コストが与えられた複数の都市の間を最短経路で巡回する経路を求める問題である。このとき、ある一つの都市を二度通過することはできない。TSP は最適解を求めようとして決定論的なアルゴリズムで計算を行うと、多数項オーダに展開することが不可能な問題、いわゆる NP-完全な問題として知られており、遺伝的アルゴリズムを含め^{18)~20)}、様々な手法によるアプローチが試みられている。本手法では、現在個体のいる都市の次にどの都市を訪れるかをそれぞれの LT により決定する。

以下、本手法における LT による TSP への適用方法について説明する。また、ここでは便宜上、多数の個体からなる集団があり、それぞれが LT を持っているものとする。

いま、全都市数を N としたとき、個体の LT である S を次のように表す。

$$S = \{S_1, S_2, \dots, S_N\}. \quad (1)$$

$$S_i = \{a_{i1}, a_{i2}, \dots, a_{iN}\}. \quad (2)$$

$$a_{ij} = [u, v] \in R. \quad (3)$$

ここで、 S_i が遺伝的操作を加える際のストリングに相当し、一点交叉を施す。 a_{ij} は現在個体がいる都市 i の次に都市 j を訪れる度合いを表す。突然変異は、ランダムに選択された LT の要素 a_{ij} に式 (3) の範囲で任意の実数値を割り当てる。この S_i をもとに、各個体は都市 i の次に都市 j を訪れる確率を次式によって求める。

$$p_{ij} = \begin{cases} \frac{e^{a_{ij}}}{\sum_{k=0}^N e^{a_{ik}}} & (j \in C). \\ 0 & (j \notin C). \end{cases} \quad (4)$$

ここで、 C は個体が既に訪れた都市の集合である。

各個体は、任意の都市から出発し、式 (4) をもとに次に訪れる都市を決定する。そして、 $\bar{C} = \phi$ となるまで、すなわちすべての都市を訪れるまで、この操作を繰り返すことで巡回経路を生成する。

このように、各個体の LT は TSP における巡回経路そのものを直接エンコードしたものではない。むしろ、環境からの入力に対してどのような振る舞いを行うかを逐次決定していくための戦略といえる。現時点においては、ラマルク進化の客観的な評価を目的として最適化問題を取り上げているため、ごく単純な方法により戦略を記述しているが、上記のように TSP に関しては都市数の増加に伴い、LT の長さが指数関数的に増加してしまう。この問題を避けるためには、LT の状態圧縮の技術が不可欠であり、今後、探索コストを軽減するための工夫を検討する必要がある。

3.2 ラマルク進化の導入

次に、この LT-GA にラマルク進化を導入する。本手法では、個体の環境への自己適応として、強化学習^{21),22)}的なオペレータによりそれぞれが自らの戦略を評価、変更し、次の世代へと遺伝させるものとする。個体独自の戦略の評価および変更は、目的指向型の適応であり、その結果は獲得された形質といえる。

本手法における強化則として、現在の LT にもとづいて生成した巡回経路が、前回に比較して評価が高ければ今回生成した巡回経路を生成する確率を上げ、逆に低ければ確率を下げるものとする。すなわち、巡回経路を生成するにあたって、ある時点で選択された LT の要素 $a_{\alpha\alpha+1}, a_{\alpha\alpha-1}$ ($\alpha, \alpha+1, \alpha-1$ は巡回経路上の都市 c_i, c_{i+1}, c_{i-1}) に対して、

$$a_{\alpha\alpha+1}(t+1) = a_{\alpha\alpha+1}(t) + \Delta. \quad (5)$$

$$a_{\alpha\alpha-1}(t+1) = a_{\alpha\alpha-1}(t) + \Delta. \quad (6)$$

ただし、

$$\Delta = \begin{cases} bs(v - a_{a+1}(t)) & (L_t < L_{t-1}). \\ bt(v - a_{a+1}(t)) & (l_{t,aa+1} < l_{t-1,aa_{prev}}). \\ wr(u - a_{a+1}(t)) & (l_{t,aa+1} > l_{t-1,aa_{prev}}). \end{cases} \quad (7)$$

ここで、

L_t : t 回目に生成した経路の総距離。

$l_{t,aa+1}$: t 回目に訪れた都市 c_i から都市 c_{i+1} までの距離。

a_{prev} : 前回、都市 c_i の次に訪れた都市

bs, bt, wr : ラマルク進化に関するパラメータ

である。

bs, bt, wr はリワードとペナルティに関するパラメータであるが、ここでは特に、次のような性質を持つ。

bs : 今回通った経路が今までに見つかっている最短経路よりも短い場合にリワードを与える、大局的な情報にもとづくパラメータ。

bt : 今回通った一区間が前回通った一区間よりも評価が高い場合にリワードを与える局所的な情報に関するパラメータ。

wr : bt とは反対の特性を持つパラメータ。

また、 bs, bt, wr の定義域は、

$$bs = [0.0, 1.0]. \quad (8)$$

$$bt = [0.0, 1.0]. \quad (9)$$

$$wr = [0.0, 1.0]. \quad (10)$$

であり、絶対値が大きいほど獲得される形質が大きい。

図3にLLT-GAにおける個体の構造を示す。各個体は行動モジュールLTとラマルク進化の評価モジュールから構成され、 bs, bt, wr に従ってそれぞれのLTを変更する。この意味で、 bs, bt, wr はラマルク進化の評価モジュールの性質を決定するものである。また、図4にラマルク進化を導入したLT-GAの概念図を示す。まず、各個体は、環境に対するアクションとして、それぞれのLTにもとづき巡回経路を生成する。次に、TSPという環境との評価をもとに各戦略を評価し、その結果を獲得形質としてLTを変更する。その後、これらのLTに遺伝的アルゴリズムによる遺伝的操作を

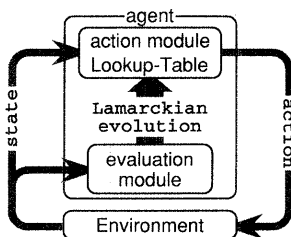


図3 ラマルク進化の評価モジュールを持つ個体の構造

Fig. 3 The structure of an agent with the evaluation module of Lamarckian evolution.

施す。

3.3 計算機実験および考察—その1

ここでは、都市配置の異なる問題を用いて計算機実験を行い、各世代での最も評価の高い巡回経路によりシステムの評価を行う。最適化問題において発見された解による性能評価法は、獲得形質の遺伝により個体の環境に対する適応能力が向上することが期待されるラマルク進化を評価する方法として妥当と思われる。

まず、図5のような格子状に配置された36都市に対して実験を行った。この都市配置においては、隣り合う都市間の距離はすべて等しく、それらを結ぶことで得られる最適解は複数存在する。図5は最適解の一例である。ここで、先に説明したラマルク進化の評価モジュールの性質を決定するパラメータを表1のように設定した。評価モジュールの性質の差異により、異なる

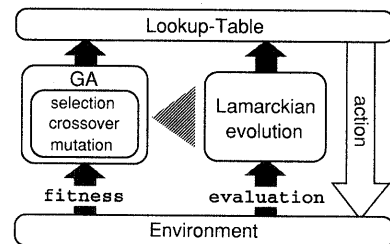


図4 ラマルク進化を導入したLLT-GA

Fig. 4 LLT-GA introduced Lamarckian evolution into.

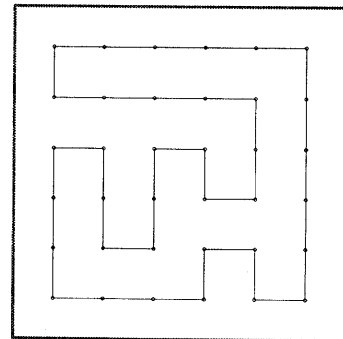


図5 格子状に配置された都市

Fig. 5 The map of lattice-placed cities.

表1 ラマルク進化のパラメータ
Table 1 Parameters of Lamarckian evolution.

	bs	bt	wr
case 1	0.8	0.1	0.2
case 2	0.2	0.7	0.2
case 3	0.4	0.2	0.1
case 4	0.8	0.6	0.5
case 5	0.2	0.1	0.9

る挙動が観測されることが予想される。また、遺伝的アルゴリズムに関しては、文献 11) を参考に表 2 のような経験的に適切と思われるパラメータを用いた。これらの条件のもとでの計算機実験の結果を図 6 に示す。図は 5 回の試行の平均である。

図 6 からわかるように、格子状に配置された都市においては、ラマルク進化を用いることにより評価モジュールの性質の違いに関係なく、いずれの場合でも最適解に達するまでの世代数が減少し、環境に対する LT-GA の収束度が大きく向上することが確認された。

次に、図 7 のようなランダムに配置された 30 都市において同様の比較実験を行った。この都市配置での実験条件は、格子状に配置された都市の場合と同じである。図 8 に 5 回の試行の平均を示す。

図 8 に示されるように、この都市配置においてもラマルク進化を用いることで LLT-GA の収束度が大きく向上する結果が得られたが、その結果は評価モジュールの性質を決定するパラメータセットにより若干のばらつきが見られる。この問題の場合には、case 1, 3 のような 3 つのパラメータの中で bs が最も大きな評価モジュールの収束度が大きく向上している。一方、 wr が最も大きい case 5 の場合には、case 1, 3 に比較して収束度は向上していない。また、 bt が大きい case 2, 4 は中間的な結果となっている。

以上、ここまでの実験結果をまとめると、LT-GA に

表 2 遺伝的アルゴリズムに関するパラメータ
Table 2 Parameters of Genetic Algorithms.

population size	50
crossover rate	0.666
mutation rate	0.333

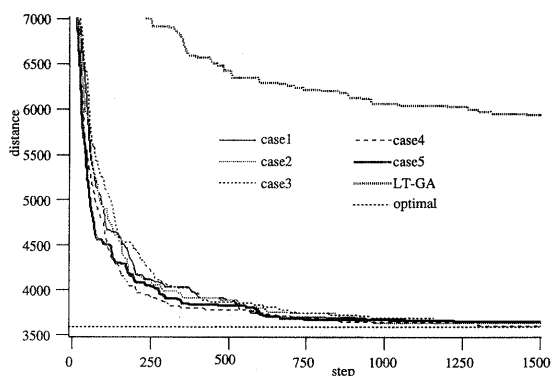


図 6 格子状配置問題の実験結果

Fig. 6 Experimental results of lattice-placed problem.

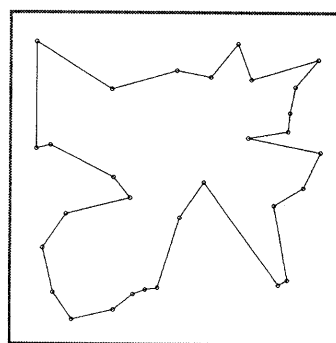


図 7 ランダムに配置された都市

Fig. 7 The map of random-placed cities.

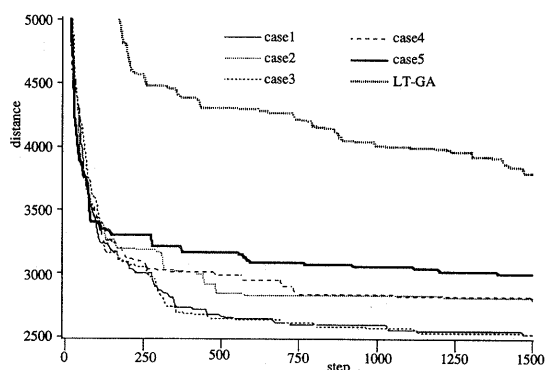


図 8 ランダム配置問題の実験結果

Fig. 8 Experimental results of random-placed problem.

比較して、本手法で提案する LLT-GA の方がシミュレーションの初期段階において優れた解が得られていることから、ラマルク進化を導入することにより適応能力を向上させることができると考えられる。これは、個体の獲得した情報を次の世代へと伝達するラマルキズムの有効性を示すものである。

しかし、ここでの実験からわかるように、どのような情報を重要視するかは適用する問題に依存するものと考えられる。つまり、格子状に配置された都市配列においては、隣り合う都市を結ぶことで最短経路が得られるため、 bt や wr による局所的な評価をもとに、ある一区間の戦略を変更することは全行程の距離を最小化することにつながる。一方、ランダムに配置された都市においては、ある一区間の距離を最小化することが全行程の距離を最小化するとは限らない。よって、case 2, 4, 5 のように bt や wr による局所的な情報をあまりに大きく扱うと、適応能力が向上する可能性はあるものの、局所解に陥る危険性がある。

また、次に、獲得された染色体を示し、染色体に表現された戦略の洗練さを比較する。図 9 に 30 都市配置

問題での LT の三次元グラフを示す。グラフは、個体が系列 p 上の都市にいるとき、次に系列 n 上の都市に遷移する確率を z 軸に表したものであり、各系列 p 上の要素の和は、1 に等しい。

図 9 a) は探索の初期段階において任意に選択した個体の LT であるが、この段階では系列 p 上の遷移確率にばらつきがあるため、個体の通る経路は安定せず、洗練された戦略とは言えない。一方、図 9 b) および c) はそれぞれ、case 3 のパラメータを用いた LLT-GA および LT-GA によって獲得された LT である。これらにおいては、ある特定の経路を通るように、LT に記述された確率が成長している。

これらをより厳密に分析するために、系列 p 上の遷移確率を変量としてそれぞれ分散を計算し、系列 p1 から系列 p30 までの分散の平均を表 3 に示す。分散の値が大きいほど、系列 p 上の都市にいるときに系列 n 上の特定の都市を訪れる傾向が強く、洗練された戦略であると考えられる。表 3 からわかるように、LLT-GA および LT-GA によって獲得された LT は、図 9 a) の LT に比較して分散の平均値が高く、特定の経路を通るような洗練された戦略となっている。しかし、LT-GA によって獲得された LT には p4 または p5 などの系列において分散の値が 0.00108, 0.00133 と非常に低く、随所に洗練さを欠く系列が見られる。以上、先の実験結果も含め、ラマルク進化を導入することで、より環境に適応するような洗練された戦略が獲得されることが確認された。

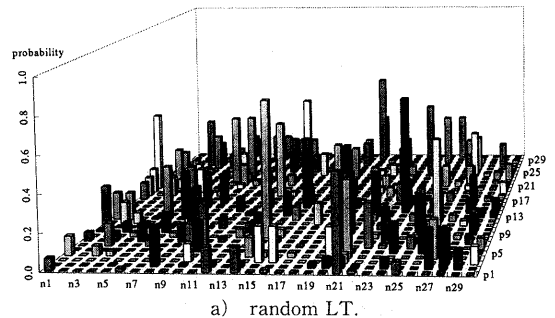
4. ジョブショップスケジューリング問題での議論

次に、個体が過去に決定した振る舞いにより将来の制約条件が変化するような環境として、ジョブショップスケジューリング問題を取り上げる。ジョブショップスケジューリング問題は、プランニングの段階において必然的に生じる複雑な最適化問題であり、多くの研究が成されている^{23)~29)}。問題の目的は、与えられた技術的順序を満足し、最大完了時間を最小になるように各機械上での作業の処理順序を決定することである。

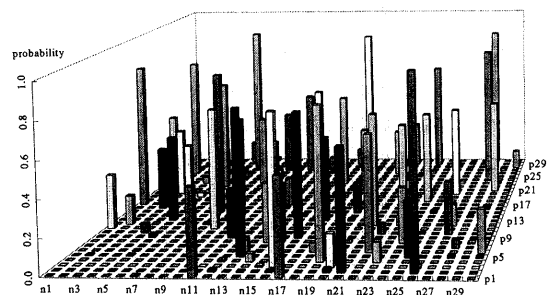
この種の順序づけ問題のスケジュールは、実行可能なスケジュールを意味するが、そのようなスケジュールは図 10 のようないくつかの種類に分類され、その最適解はアクティブスケジュールの中に存在することが知られている。アクティブスケジュールとは、技術的順序を保ちながら各機械上での作業の処理順序の変更を許容した上で、どの作業も、他の作業の開始時刻を

遅らせることなしに、その処理開始を早めることができないスケジュールである。

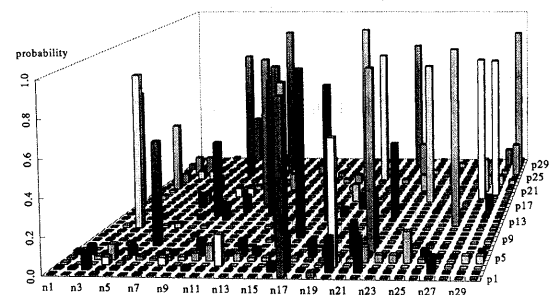
本論文では、この性質を考慮し、Giffier と Thomp-



a) random LT.



b) LT acquired with LLT-GA.



c) LT acquired with LT-GA.

図 9 LT の 3D グラフ化

Fig. 9 3D graphs of LTs.

表 3 系列 p 上の要素である遷移確率の分散の平均値
Table 3 Mean of variances on p-series.

	a) random LT	b) LLT-GA	c) LT-GA
mean of s^2	0.00677	0.01740	0.01730

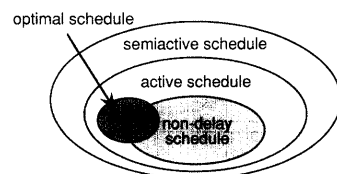


図 10 スケジュール集合

Fig. 10 Set of schedules.

son らによるアクティブスケジュール生成アルゴリズム (GT 法)³⁰⁾と LT とを対応させる。

4.1 アクティブスケジュール生成アルゴリズム (GT 法)

いま, n 個のジョブ $J_i (i=1, \dots, n)$ が m 台の機械 $M_r (r=1, \dots, m)$ で処理されるものとする。また, 機械 M_r 上でジョブ J_i が l 番目に処理されるとき, この作業を O_{ilr} と書くものとする。

まず, スケジューリングの各段階において, スケジュールされていない作業のうちで技術的順序によって特定される処理開始可能作業の集合 C (カット集合) を求める。また, カット集合 C に含まれる各作業 O_{il} の最早完了時刻を E_{il} とする。GT 法では, すべてのスケジュールが確定するまで次の手順を繰り返す。

1. $E_{i^*l^*r^*} = \min\{E_{il} | O_{il} \in C\}$ となるような作業 $O_{i^*l^*r^*}$ を求める。 $O_{i^*l^*r^*}$ と同じ機械 M_{r^*} で処理される作業 $O_{ilr^*} \in C$ のうちで, $O_{i^*l^*r^*}$ と処理時間区間の重なっている作業の集合 G (コンフリクト集合) を求める。
2. G の中から作業 O_{isls} を一つ選択し, O_{isls} を E_{isls} に従ってスケジュールする。
3. カット集合 C , およびスケジュールされていない作業の最早完了時刻 E_{il} を修正する。

4.2 LT-GA

先に述べたように, 本手法においては, 各個体の LT を GT アルゴリズムに対応させる。すなわち, GT アルゴリズムによって求められるコンフリクト集合において, どのジョブをスケジュールするかを各個体の LT により決定する。以下, 本手法における LT-GA によるスケジュールの生成法について説明する。

いま, n 個のジョブ $J_i (i=1, \dots, n)$ が m 台の機械 $M_r (r=1, \dots, m)$ で処理されるとき, 個体の LT である S を次のように表す。

$$S = \{S_1, S_2, \dots, S_m\}. \quad (11)$$

$$S_r = \{S_1^r, S_2^r, \dots, S_n^r\}. \quad (12)$$

$$S_i^r = \{a_{i1}^r, a_{i2}^r, \dots, a_{in}^r\}. \quad (13)$$

$$a_{ij}^r = [u, v] \in R. \quad (14)$$

ここで, S_r は機械 M_r に関する LT であり, 機械 M_r 上でジョブ J_i の次に作業するジョブを決定するための S_i^r からなっている。この S_i^r が遺伝的操作を加える際のストリングに相当する。 a_{ij}^r は機械 M_r 上でジョブ J_i の次にジョブ J_j をスケジュールする度合いを表し, この S_i^r をもとに, ジョブ J_i の次にジョブ J_j をスケジュールする確率を次式によって求める。

$$p_{ij}^r = \begin{cases} \frac{e^{a_{ij}^r}}{\sum_{k=0}^n e^{a_{ik}^r}} & (J_j \in G). \\ 0 & (J_j \notin G). \end{cases} \quad (15)$$

G は GT アルゴリズムにおけるコンフリクト集合である。

ここで, スケジュールを生成するまでの LT の動作を簡単に追跡してみよう。まず, カット集合という環境からのメッセージが入力される。そして, $O_{i^*l^*r^*}$ を特定し, $O_{i^*l^*r^*}$ と処理時間区間の重なっている作業の集合 G を求める。ここで各個体は, コンフリクト集合 G において機械 M_{r^*} 上でどのジョブをスケジュールするかを, それぞれの LT である $S_{i^*l^*r^*}^{(r^*, l^*-1)}$ により決定する。 $J(r^*, l^*-1)$ は, 機械 M_{r^*} 上で直前にスケジュールされたジョブである。

4.3 ラマルク進化の導入

次に, ラマルク進化を導入する。ここでも先の TSP での LLT-GA と同じ強化則を用いる。すなわち, 機械 M_r 上で k 番目に処理されるジョブ J_i を $J_{i,k,r}$ とすると, 今回選択された LUT の要素 a_{k+1}^r ($K, K+1$ はジョブ $J_{i,k,r}$ および $J_{i,k+1,r}$) について

$$a_{k+1}^r(t+1) = a_{k+1}^r(t) + \Delta. \quad (16)$$

ただし,

$$\Delta = \begin{cases} bs(v - a_{k+1}^r(t)) & (T_t < T_{\min}). \\ bt(v - a_{k+1}^r(t)) & (T_t < T_{t-1}). \\ wr(u - a_{k+1}^r(t)) & (T_t > T_{t-1}). \end{cases} \quad (17)$$

ここで,

T_t : 個体が t 回目に生成したスケジュールの所要時間

$T_{\min}$: 現在までの最小所要時間

bs, bt, wr : ラマルク進化に関するパラメータ

である。

TSP の場合と同様, bs と bt は, 今回生成したスケジュールの評価が高かった場合にそのスケジュールを生成する確率を高くするリワードに関するパラメータであり, 反対に, wr は評価が低い場合にそのスケジュールを生成する確率を低くするペナルティに関するパラメータである。

4.4 計算機実験および考察—その 2

以上の定義をもとに, 最適解の知られている問題を用いて計算機実験を行う。ジョブショップスケジュールリング問題は, TSP に比較して制約条件が多く, 過去に決定したスケジュールによりその後の解空間のランドスケープが変化するため, より複雑な挙動を示すことが予想される。

まず, 10 ジョブ 5 タスク問題 (10×5 問題) におい

て、比較実験を行った。この問題の最適スケジュールはいくつか存在し、その総所要時間は 666 である。また、ラマルク進化の評価モジュールの性質を決定するパラメータは、何回かの予備実験により表 4 のように設定した。おおまかに分類すると、case 1 は評価が高かった場合のリワードに関するパラメータが大きく、反対に case 2 は、case 1 に比較してペナルティに関するパラメータが大きなパラメータセットである。また、遺伝的アルゴリズムに関しては、TSP の場合と同じパラメータを用いた。これらの条件の下での実験結果を図 11 に示す。図は 10 回の試行の平均である。ここで、LLT とは、選択、交叉といった遺伝的アルゴリズムによるオペレータを一切用いず、ラマルク進化のみによる LT を意味する。

図からわかるように、 10×5 問題においては、case 1 のパラメータセットによるリワード型の評価モジュールを持つ LLT-GA の適応能力が向上しているが、その一方で、case 2 のペナルティ型の評価モジュールを持つ LLT-GA に関しては、ラマルク進化を用いることによる効果は見られなかった。

次に、10 ジョブ 10 タスク問題 (10×10 問題)³¹⁾ に関して同様の比較実験を行った。この問題は、先の 10×5 問題に比較して、ある機械上でコンフリクトとなるジョブが多く、スケジュールの結果に対してクリティカルな結果をもたらす作業が存在する、大規模な問題である。 10×10 問題での実験結果を図 12 に示す。図は

表 4 ラマルク進化のパラメータ

Table 4 Parameters of Lamarckian evolution.

	<i>bs</i>	<i>bt</i>	<i>w_r</i>
case 1	0.8	0.3	0.1
case 2	0.5	0.2	0.3

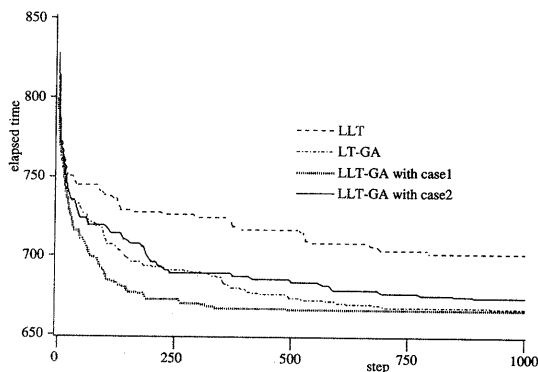


図 11 10×5 問題における実験結果

Fig. 11 Experimental results of 10×5 problem.

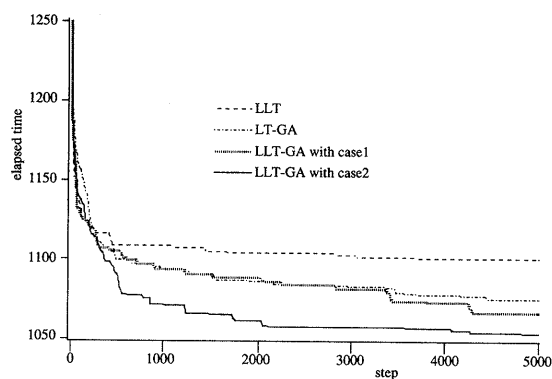


図 12 10×10 問題における実験結果

Fig. 12 Experimental results of 10×10 problem.

10 回の試行の平均である。また、各種パラメータの設定は、 10×5 問題の場合と同じである。

この 10×10 問題においては、先程の実験結果とは逆の結果が得られた。すなわち、case 2 を用いたペナルティ型の評価モジュールを持つ LLT-GA については、優れた適応能力を示す結果が得られたが、 10×5 問題において高い収束度を示した case 1 を用いた LLT-GA に関しては良い結果は得られず、単なる LT-GA と同じような結果に終わっている。

これらの実験から、まずここで確認しておきたいのは、個体の自己適応のみによる LLT に比較して、集団の適応を扱った LT-GA の収束度が高いことである。すなわち、個体独自の戦略評価、変更よりも遺伝的アルゴリズムにおける積木仮説²⁾が効果的に働いたものと考えられる。そして、本手法で提案するラマルク進化を導入した LLT-GA は、表現型での評価をもとに得られた情報を building block へとフィードバックすることで、さらに優れた適応能力を示すことが確認された。しかし、LLT-GA の能力は、評価モジュールの性質を決定するパラメータにより変化するものであり、適用する問題空間に応じた評価モジュールの設定が必要になるものと思われる。

5. おわりに

本論文では、個体の環境への自己適応を重視したラマルク型の LLT-GA を提案し、これを客観的に評価するために最適化問題へと適用した。そして、計算機実験により LLT-GA の優れた適応能力を示し、個体の環境から得た情報を利用することの重要性を確認した。

しかし、計算機実験に示されるように、LLT-GA の能力は評価モジュールの設定に影響される。これは、適用する問題空間に依存するものであり、その問題空

間に応じた形質の有効性を考慮しなくてはならない。したがって、ラマルク進化により適応的な進化を達成するには、何らかの制御機構が必要になると考えられるが、この点に関する議論は今後の課題である。さらに、LLT-GA の性質を評価するには、問題空間における評価値のランドスケープが動的に変化するような環境での検討が必要である。

参 考 文 献

- 1) Holland, J. H.: *Adaptation in Natural and Artificial Systems*, The MIT Press (1992).
- 2) Goldberg, D. E.: *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley (1989).
- 3) Cherfas, J. 編, 松永俊男, 野田春彦, 岸由二共訳: 生物の進化 最近の話題, 培風館 (1984).
- 4) Ackley, D. H. and Littman, M. L.: Interactions between Learning and Evolution, Artificial Life II, *Santa Fe Institute Studies in the Sciences of Complexity, Proc.*, Vol. X, pp. 487-509, Addison-Wesley (1991).
- 5) Floreano, D.: Emergence of Nest-Based Foraging Strategies in Ecosystems of Neural Networks, From Animals to Animats 2, *Proc. the Second Int. Conf. on Simulation of Adaptive Behavior*, pp. 410-416, The MIT Press (1993).
- 6) Horn, J.: Measuring the Evolving Complexity of Stimulus-Response Organisms, Toward a Practice of Autonomous Systems, *Proc. the First European Conf. Artificial Life*, pp. 365-374, The MIT Press (1991).
- 7) Bedau, M. A. and Packard, N. H.: Measurement of Evolutionary Activity, Teleology, and Life, Artificial Life II, *Santa Fe Institute Studies in the Sciences of Complexity, Proc.*, Vol. X, pp. 431-461, Addison-Wesley (1991).
- 8) Dewdney, A. K.: Computer Recreations, Simulated Evolution: Wherein Bugs Learn to Hunt Bacteria, *Scientific American*, No. 5, pp. 104-107 (1989).
- 9) Iba, H., Akiba, S., Higuchi, T. and Sato, T.: BUGS: A Bug-Based Search Strategy Using Genetic Algorithms, *Parallel Problem Solving from Nature 2*, pp. 165-174, Elsevier Science Publishers B. V. (1992).
- 10) 伊庭斉志: GA と探索, 計測と制御, Vol. 32, No. 1, pp. 39-45 (1993).
- 11) 伊庭斉志, 佐藤泰介: 遺伝アルゴリズムを拡張した虫型探索: BUGS, 人工知能学会誌, Vol. 8, No. 6, pp. 797-809 (1993).
- 12) Farmer, J. D. and Belin, A. d'A.: Artificial Life: The Coming Evolution, Artificial Life II, *Santa Fe Institute Studies in the Sciences of Complexity, Proc.*, Vol. X, pp. 815-840, Addison-Wesley (1991).
- 13) Ackley, D. H. and Littman, M. L.: A Case for Lamarckian Evolution, Artificial Life III, *Santa Fe Institute Studies in the Sciences of Complexity, Proc.*, Vol. XVII, pp. 3-10, Addison-Wesley (1993).
- 14) Grefenstette, J. J.: Lamarckian Learning in Multi-agent Environments, *Proc. the Forth Int. Conf. Genetic Algorithms*, pp. 303-310, Morgan Kaufmann (1991).
- 15) Davidor, Y.: Sub-Goal Reward and Lamarckism in a Genetic Algorithm, *Proc. the Ninth European Conf. Artificial Intelligence*, Stockholm, pp. 189-194 (1990).
- 16) Grefenstette, J. J.: A System for Learning Control Strategies with Genetic Algorithms, *Proc. the Third Int. Conf. Genetic Algorithms*, pp. 183-190, Morgan Kaufmann (1989).
- 17) Lawler, E. L., Lenstra, J. K., Rinnooy Kan, A. H. G. and Shmoys, D. B.: *The Traveling Salesman Problem; A Guided Tour of Combinatorial Optimization*, A Wiley-Interscience Publication (1985).
- 18) Grefenstette, J. J., Gopal, R., Rosmaita, B. and Van Gucht, D.: Genetic Algorithm for the Traveling Salesman Problem, *Proc. the First Int. Conf. Genetic Algorithms and their Applications*, pp. 160-165 (1985).
- 19) Goldberg, D. E. and Lingle, R., Jr.: Alleles, Loci, and the Traveling Salesman Problem, *Proc. the First Int. Conf. Genetic Algorithms and their Applications*, pp. 154-159 (1985).
- 20) 山村雅幸, 小野貴久, 小林重信: 形質の遺伝を重視した遺伝的アルゴリズムに基づく巡回セールスマン問題の解法, 人工知能学会誌, Vol. 7, No. 6, pp. 1049-1059 (1992).
- 21) Narendra, K. and Thathachar, M. A. L.: *Learning Automata An Introduction*, Prentice Hall, Englewood Cliffs (1989).
- 22) Sutton, R.: Reinforcement Learning Architectures for Animats, From Animals to Animats, *Proc. the First Int. Conf. Simulation of Adaptive Behavior*, pp. 288-296, The MIT Press (1991).
- 23) Conway, R. W., Maxwell, W. L. and Miller, L. W.: *Theory of Scheduling*, Addison-Wesley (1967).
- 24) 鍋島一郎: スケジューリング理論, 森北出版 (1974).
- 25) Balas, E.: Machine Sequencing via Disjunctive Graphs: An Implicit Enumeration Algorithm, *Operations Research*, Vol. 17, pp. 941-957

(1969).

- 26) Nakano, R.: Conventional Genetic Algorithm for Job Shop Problems, *Proc. the Fourth Int. Conf. Genetic Algorithms*, pp. 474-479, Morgan Kaufmann (1991).
- 27) Bagchi, S., Uckun, S., Miyabe, Y. and Kawamura, K.: Exploring Problem-Specific Recombination Operators for Job Shop Scheduling, *Proc. the Fourth Int. Conf. Genetic Algorithms*, pp. 10-17, Morgan Kaufmann (1991).
- 28) Yamada, T. and Nakano, R.: A Genetic Algorithm Applicable to Large-Scale Job-Shop Problems, *Parallel Problem Solving from Nature 2*, pp. 281-290, Elsevier Science Publishers B. V. (1992).
- 29) Tamaki, H. and Nishikawa, Y.: A Paralleled Genetic Algorithm based on a Neighborhood Model and Its Application to the Jobshop Scheduling, *Parallel Problem Solving from Nature 2*, pp. 573-582, Elsevier Science Publishers B. V. (1992).
- 30) Giffler, B. and Thompson, G. L.: Algorithms for Solving Production Scheduling Problems, *Operations Research*, Vol. 8, pp. 487-503 (1969).
- 31) Muth, J. F. and Thompson, G. L.: *Industrial Scheduling*, Prentice-Hall, Englewood Cliffs (1963).

(平成 6 年 11 月 21 日受付)

(平成 7 年 4 月 14 日採録)



吉井伸一郎 (正会員)

昭和 46 年生. 平成 6 年北海道大学工学部精密工学科卒業. 現在, 同大学大学院修士課程に在学中. 平成 6 年情報処理学会北海道支部奨励賞受賞. 機能の自己組織化, および自動プログラミングに興味を持ち, 遺伝的アルゴリズム, 人工生命, 複雑系等の研究に取り組んでいる.



鈴木 恵二 (正会員)

昭和 40 年生. 昭和 63 年北海道大学工学部精密工学科卒業. 平成 2 年同大学院精密工学専攻修士課程修了. 平成 5 年同大学院博士課程修了. 工学博士. 北海道大学工学部複雑系工学講座助手. 現在に至る. 人工生命, 遺伝的アルゴリズムなどに興味を持つ. 日本機械学会, 人工知能学会等の会員.



嘉数 侑昇 (正会員)

昭和 16 年生. 昭和 48 年北海道大学大学院工学研究科精密工学専攻博士課程修了. 工学博士. 北海道大学工学部複雑系工学講座教授. 創発的学習, 自律系工学, ロボティクス, 知識工学, ニューラルネットワーク等の研究に従事. 日本機械学会, 精密工学会, 日本ロボット学会, 計測自動制御学会等の会員.