蟻の理論を用いた遺伝的アルゴリズムの環境変化の考察

安孫子 和哉*　　神谷 昭基**

Worker Ants' Rule-Based Genetic Algorithms Dealing with Changing Environments

Kazuya ABIKO　　Akimoto KAMIYA

Abstract- Genetic Algorithms (GA) is applied to search an optimal or near-optimal solution by imitating the biological evolution. In GA, all individuals in a population tend to converge to an optimal point. Once it converges, the conventional GA is difficult to adapt itself in changing environments. In nature, it has been found that in each worker ant colony, about 20% are diligent worker ants, 60% are ordinary, and 20% are lazy. That is 20:60:20 rule. We have applied this rule to preserve not only the best individuals but also the poorest ones. Simulation results verified that the poorest individuals contribute to dealing with optimization problems in changing environments.

Key Words: Genetic Algorithms, Worker Ants' Rule

1　はじめに

産業の分野において，カーナビの経路選択や工場のジョブスケジューリングなどでは， 最適化という手法は非常に重要な技術である。また, 工学分野において最適化は多くの手法が研究されている [1]。

現実世界において，カーナビの経路選択のような最適化問題を例に挙げると，多くの経路を対象とするため解の探索空間が広がる。また，経路の形状によっては多峰性のある問題となり，複雑な問題となるため, 最適解を探索することは難しくなる。そこで注目されたのが，最適化手法の中の進化的戦略の一種である遺伝的アルゴリズム (Genetic Algorithms:GA，以下 GA) である。

GA は生物の進化を模倣したアルゴリズムであり，個体 (解) に選択，交叉，突然変異という操作を繰り返し行うことで最適解を見つけ出すことが期待できるアルゴリズムである。 GA は探索空間の情報を遺伝子という形で持ち，個体を集団として形成することで多点探索をし，解の探索を行う。そのため，大規模な最適化問題を解くことが期待できる。

* 釧路高専専攻科 電子情報システム工学専攻
** 釧路高専情報工学科

カーナビの経路選択問題のように，移動時間を最短にするような問題では，渋滞などによって経路の通行所要時間が変化する場合，最適解が変化する。このような動的な最適化問題が，ある時点で最適解が変化することを環境変化と呼ぶ。GA は解の探索が進むことで最良個体の割合が増えるため，集団の「多様性」が失われる. 環境変化が起きた時個体に多様性がない場合，類似した親の遺伝子を受け継ぐことで個体の進化が進まず，新たな環境に対する最適解を探索することが難しくなる。

従来の研究では，働き蟻の集団が「優秀な個体：中間な個体：劣等な個体」を「2:6:2」[2] という割合で多様性を維持しながら行動すること (蟻の理論) をヒントに，劣等個体を意図的に保存し，多様性を維持する GA が構築された [3][4]。これにより，限られた環境での環境変化への適応や環境変化の小さい環境での適応が確認された。

本研究では，劣等個体を保存するだけではなく，交叉に含めるアルゴリズムを新たに提案し，様々な環境で実験を行うことで新たなアルゴリズムの有効性を検証し，環境変化に対する考察を行う。

実験では，提案するアルゴリズムに加え，もっと

も一般的な GA である SGA(Simple GA)，固定環境で強力な GA である CHC(Cross generational elistist selection,Heterogeneous recombination,Cataclysmic mutation)[5]，環境変化を考慮して提案された hyper mutation step[6] を導入した GA，従来の蟻の理論を用いた GA である AT(Genetic Algorithms applying Ant Theory)[3]，CIAT[4] と比較する。CHC を用いることで，提案するアルゴリズムにおける環境変化前での性能，hyper mutation step を導入した GA を用いることで，環境変化に対する性能，そして AT, CIAT と比較することで，従来提案されている蟻の理論を用いたアルゴリズムとの性能の違いを考察する。

2 GA

GA では, 計算機上で生物進化を模倣するため, 遺伝子コードと呼ばれる細胞の染色体に相当するものを用いる. そのコードを用いて, 個体を表現する. その個体が集まったものを個体群と呼び, 個体群の個体数を個体群サイズと呼ぶ. また, 各遺伝子の置かれる位置を遺伝子座という.

個体は自らの遺伝子コードによって, その環境に対する適応度が決められる. 適応度の高い個体は高い確率で次世代へと残り, 逆に適応度の低い個体は高い確率で淘汰されていく. この操作を GA では, 選択と呼ぶ.

選択には, 適応度に比例して個体を確率的に選択するルーレット選択, 個体を適応度の大きい順に並べ, 順位に応じて個体を残すランク方式, ランダムに数個体を選び出し, その中で一番適応度の高い個体を個体群サイズ分選ぶトーナメント方式などがある [7].

適応度の高い個体は確率的に多く選ばれ, 個体群中の割合が増加する. しかし, 適応度の高い個体の割合が増加するだけでは個体が進化することは不可能である. そのため, 生物進化の過程で必要な生殖, 突然変異を模倣して, 遺伝子コードの複数部の組み換え, 一部の変化を行う. この操作を GA では, 交叉, 突然変異と呼ぶ.

交叉により 2 つの親個体から 2 つの子個体が生み出される. 交叉の手順は, まず, 遺伝子コード上に交叉点と呼ばれる仕切り位置を決める. 交叉点はランダムに決める. 次に, 交叉点の右側にある遺伝子をそのまま入れ替えることで,2 つの子個体を生み出す. これを一点交叉と呼ぶ. 交叉点を 2 つ用いるものを 2 点交叉, それ以上の交叉点を多点交叉, または複数点交叉と呼ぶ. また, 両親の各遺伝子を 50%の確率で入れ替える一様交叉と呼ばれる交叉法もある [7]. これらの交叉は設定した交叉率という確率によって行われる.

交叉によって新たな個体を生成することが可能になった. しかし, 交叉は親個体の遺伝子に依存するため, 親個体が持っていない遺伝子を得ることができない. これでは, 十分に進化しない恐れがあり, ある世代において進化が滞る可能性がある. そのため, GA は遺伝子コードの各遺伝子が低い確率で突然変異を起こす. この確率を突然変異率と呼ぶ.

突然変異を行うことで, 交叉のみでは生成することのできない個体を生成する. また, 個体に多様性を持たせる役割も行う.

図 1 に GA の流れ図を示す.

図 1: GA の流れ図

GA は図 1 の手順により, 選択で確率的に良い個体を選び, 交叉と突然変異で各個体を進化させることで解の探索を行い, 最適解を求めることが期待できる.

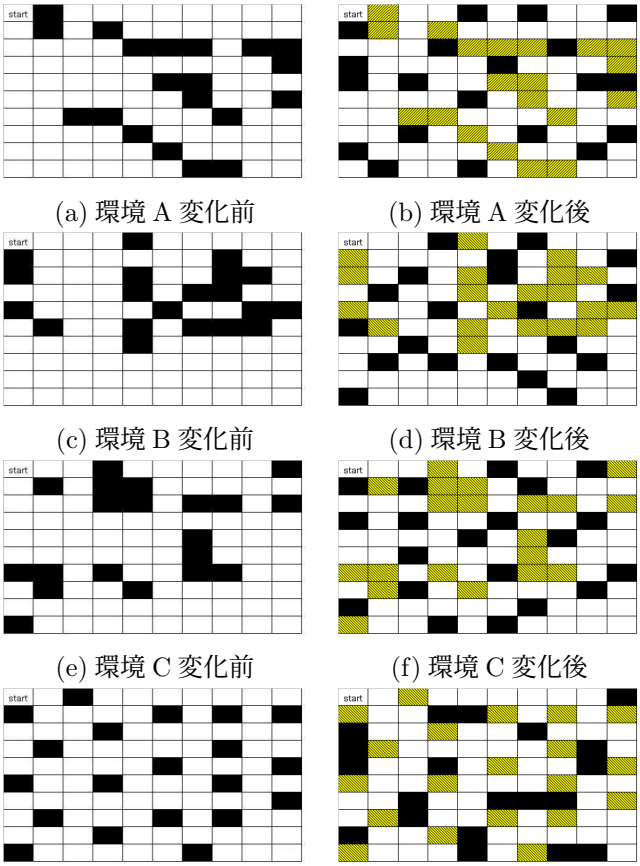
3 本研究における問題定義

本節では，本研究の実験で用いる問題の定義について述べる。

実験の問題として，ANT 問題という問題を用いた [8]。この ANT 問題とは, 10 × 10 マスの碁盤目状の探索フィールドの中に餌を配置し，人工蟻がその餌を探索するというものである。人工蟻は一歩前進するか方向転換するたびに，エネルギーを一つずつ失っていく。そして, エネルギーが 0 になるまで餌を探索し続け，できるだけ多くの餌を見つけるような遺伝子を探す。この問題において，探索できる餌の数が適応度となり，エネルギーは遺伝子コード長となる。図 2 に，遺伝子コードの例を示す。

1311213131121111311113311111221211231311111312113 1

図 2: 遺伝子コードの例



(a) 環境 A 変化前　(b) 環境 A 変化後

(c) 環境 B 変化前　(d) 環境 B 変化後

(e) 環境 C 変化前　(f) 環境 C 変化後

(g) 環境 D 変化前　(h) 環境 D 変化後

図 3: 実験で使用する環境

遺伝子の 1 は前進，2 は右方向への方向転換，3 は左方向への方向転換を表す。遺伝子コードを左から順に読み込み，個体の行動を決定する。なお，出発地点では右を向いているとする。この問題は，難易度を可視化できるため，難易度の設定が容易である。例を上げると，餌を連なりをもたせて配置すると次の餌が発見しやすく比較的易しい問題，連なりをなくしバラバラに配置すると次の餌が発見しにくいため，難しい問題となる。図 3 に本実験で使用する 4 つの環境を示す.

各環境で環境変化を実現する方法として，前環境に配置してある餌を 20 個総入れ換えすることで行う。図 3 の (b)(d)(f)(h) において，斜線部分が環境変化前に配置してあった餌の場所である。

環境 A，B は乱数を用いてランダムに餌を配置し, 環境変化後も環境変化前の場所には配置されないように乱数を用いてランダムに配置を行った。環境 C は乱数を用いてランダムに餌を配置した後，餌が連ならないような条件の下で配置し，環境変化後に難易度の高いものとした。環境 D は環境変化前に餌が連ならないような条件の下に餌を配置し，難易度の高い問題とした後，連なりを持たせた餌を何個配置するかを決め，ランダムに餌を配置し確率的に餌の連なりを作り，難易度の低い問題とした。餌の連なりが規定の個数を下回った場合，再度配置を行う。今回は連なりのある餌の個数を 7 個以上と設定した。

4　実験で用いるアルゴリズム

ここからは，本研究の実験で用いるアルゴリズムについて説明する。実験では，基本的な操作を実装し単純化した SGA(Simple GA)，選択や交叉が効率的であることから収束が早く，固定の環境での解探索が強力とされている CHC[5]，定期的に個体の一部を初期化する hyper mutation step を導入した GA[6]，蟻の理論を用いた GA である AT[3] ，AT の欠点である収束の遅さを補うために CHC を融合した CIAT[4] を用いる。また，本研究として，交叉法を変更した CIAT を二つ提案することで，これらのアルゴリズムと比較することで考察を行う。

4.1　SGA

SGA は GA の基本的な操作をもっとも単純にしたアルゴリズムである。ここで，SGA で行う選択，交叉，突然変異について説明する。

- (選択) ルーレット選択を用いる。
- (交叉) 一点交叉を用いる。
- (突然変異) 遺伝子座の値を確率的に変化させる。

選択ではルーレット選択を用いる。ルーレット選択とは，各個体 s_i の適応度 $f(s_i)(i$=1,⋯,N) とその総計を求め，各個体 s_i の選択確率 (ただし，個体群サイズを N とする) を

$$p_i = \frac{f(s_i)}{\sum_{n=1}^{N} f(s_i)}$$

として，各個体を選択していく。この選択方法の特徴は，適応度の高い個体ほど選択されやすく，低い個体ほど淘汰されやすい。しかし，低い個体が必ずしも淘汰されるわけではないので，個体の多様性は維持される [7]。

交叉では，一点交叉を用いる。交叉点はランダムに生成し，交叉を行う。突然変異は遺伝子座の値を確率的に変化させるものである。

4.2 CHC

CHC は操作の単純な SGA と違い，単純性を犠牲にする代わり各操作をより効率的に行い，得られた最良個体を確実に残すという特徴を強くもつアルゴリズムと言える。ここでは，CHC の選択，交叉，突然変異について説明する。

- (交叉) 類似した個体間の交叉を行わない. 交叉方法として，一様交叉を改良した交叉法を用いる。
- (選択) 個体群から，交叉によって新しい個体を生成した後，前の個体群と新しい個体群の中から，優秀な個体を確率を使わずに選択する。
- (突然変異) 個体群が収束したときのみ，個体群の最良個体を用いて一部の個体を初期化する。

選択では，新しい個体群と古い個体群の間で優秀な個体を確率を使わずに選んでいる。そのため，次世代に効率よく優秀な個体を残すことが可能である。

交叉では，近親間交叉を行わないようにしている。近親間交叉とは，遺伝子コードのハミング距離が小さい個体同士の交叉のことである。近親間交叉を行うと，親個体どちらも似た遺伝子コードを持っているため，子個体も親と類似した遺伝子コードを持ったものとなり，多様性の維持が困難になる。また，親と類似した子個体のため，解の探索が進まない。CHC はこれを防ぐため，閾値を設定し，閾値よりハミング距離が小さい 2 個体間の交叉を禁止している. 交叉方法では, 一様交叉を修正したもの (修正一様交叉) を使用している. 修正一様交叉では，2 個体間に異なる遺伝子座が m 個あるとき，その中で $m/2$ 個の位置をランダムに選んで，その遺伝子を交換するものである。一様交叉は確率的に遺伝子を交換したが，修正一様交叉は確率的ではなく，異なる遺伝子座の遺伝子を確実に交換する。そのため，個体の早い進化が見込まれる。

突然変異は，個体群が収束したときに行われるため，収束する度に個体の多様性が保持される。また，最良個体を用いた個体の初期化手順は，最良個体が保たれるため，初期化による最良解の消滅はない。初期化の手順は，まず最良個体の遺伝子コードを最良個体以外の全ての個体にコピーする。この時点では，最良個体のみの個体群となる。次に，最良個体を 1 個体残し，それ以外の個体全ての遺伝子座をランダムに rL 個選んで突然変異させる。ここで，L は遺伝子コード長，r は拡散率を示し，通常 0.35 と設定されている。CHC の交叉と選択が非常に効率的であることから，CHC は解の収束が早い。

4.3 hyper mutation step を導入した GA

hyper mutation step とは，GA が行う選択，交叉，突然変異に加え，一部の個体をある世代において初期化する操作のことである。初期化が定期的に行われることにより，個体の多様性が維持されることで，環境変化に対して提案されている操作である。従来では，環境変化が起こるとこの操作を行っていたが，本研究では一定の世代間隔で行っている。実際に起こる環境変化とは不規則なものであるため，本研究ではそのように行った。

hyper mutation step では，置換率と呼ばれる初期化を行う割合を示すパラメーターが設定されている。また，初期化のみでは最良個体が消滅してしまう可能性があるので，初期化前の最良個体と初期化後の最悪個体とを入れ換えるエリート戦略を行った。hyper mutation step を導入した GA の手順は次のとおりである。

- (選択) ルーレット選択を用いる。
- (交叉) 一様交叉を用いる。
- (突然変異) 遺伝子座の値を確率的に変化させる。
- (hyper mutation step) 決められた世代で, 置換率に沿って一部の個体を初期化する。

個体の多様性が維持されることで個体の進化が促進されるので，環境変化に適応できると考えられている。

4.4 AT

AT は，蟻の理論を組み込んだ GA のことである。蟻の理論を導入することで，劣等な個体を意図的に残し，個体の多様性が維持できる。そのため，環境変化が起こった場合でも，進化によって新しい個体が生まれ，解の探索を行うことができる。ここからは，AT の選択，交叉，突然変異の詳細を説明する。なお，図 4 は AT の世代交代の一例である。

- (選択) 個体群を適応度によって，優秀なグループ 2 割，中間なグループ 6 割，劣等なグループ 2 割に分類し，優秀なグループと劣等なグループは確率を使わずにそのまま次世代へと残す。
- (交叉) 片親を優秀なグループ，もう一つの片親を中間，劣等なグループの中からランダムに選び出し交叉させる。生成する個体数は 6 割分となる。また，交叉の方法は一様交叉を用いる。
- (突然変異) 遺伝子座の一箇所の値を変化させる。

選択では，優秀，中間，劣等の 3 つのグループに分類している。優秀なグループを意図的に残す理由は，交叉での最良の個体を消滅させないためである。劣等な

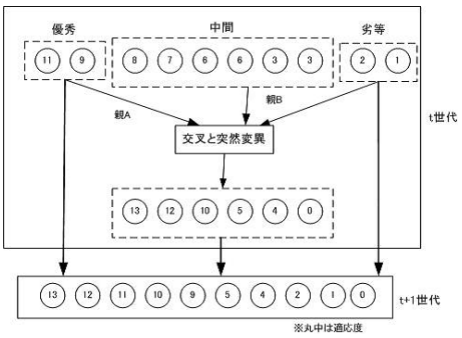


図 4: AT の世代交代の例

グループを意図的に残す理由は，蟻の理論に則り，個体群の多様性の維持を行うためである.

交叉では，必ず片親を優秀なグループから選んでいる。これは，片親を優秀な個体とすることで，新しく生成される個体に優秀な個体の遺伝子を継承させる確率が高くなり，次世代により優秀な個体を残すことが期待できるためである。また，交叉法では一様交叉を用いている。

AT は, 意図的に優秀な解と劣等な解を残しているため，次世代に生成する個体が少なく，また，一様交叉により確率的に進化が促進しない個体がでてくるため解の収束が遅いという欠点がある。そのため，解の収束が早い CHC を融合することで欠点を補う。

4.5 CIAT

CIAT とは，AT に CHC の要素を融合したアルゴリズムである。これは，劣等な個体を意図的に次世代へ残す AT の要素と CHC の効率のよい交叉，選択を組み合わせたものである。CIAT は AT と違い，優秀な個体と劣等な個体という 2 つのグループのみで分割する。ここからは，CIAT における交叉，選択, 突然変異について説明する。

- (交叉)CHC と同様の方法を用いる (4.2 節参考)。
- (選択) 個体群を適応度によって，優秀なグループと劣等なグループに分類する。次に，交叉を行い子個体を生成する。そして，優秀なグループと劣等なグループ, 新しい個体のグループとの間で，優秀な個体と劣等な個体を選択する。
- (突然変異)CHC と同様の方法を用いる (4.2 節参考)。ただし，優秀なグループの個体のみに適用する。

これにより，効率的に優秀な個体を残せるため，AT より収束が早くなることが期待できる。既存の CIAT[4] では，劣等個体を交叉に含めていないが，本研究では劣等個体を交叉に含める CIAT を提案する。劣等個体を交叉に含めることで，解の収束後に環境変化が発生した場合でも個体の進化が望めるので，環境変化に有効ではないだろうかと考えた。

5 実験

本論文では，4 つの環境を用いて環境変化を起こし，SGA，CHC，hyper mutation step を導入した GA，AT， 従来の CIAT(CIAT1) の他，本研究で提案する交叉方法を変更した CIAT(CIAT2，CIAT3) を 2 つ用意し，実験を行うことで，環境変化における考察を行う。

5.1 実験の設定

ここからは実験に用いる GA の設定について述べる。まず，従来の CIAT の交叉法と本研究で提案する CIAT の交叉法について説明する。

(1) 両方の親を優秀な個体のグループから選択し，交叉を行う (CIAT1)。

(2) 片親を優秀な個体のグループから，もう一つの片親を優秀な個体，劣等な個体の二つのグループからランダムに選択し，交叉を行う (CIAT2)。

(3) 両方の親を優秀な個体, 劣等な個体の二つのグループからランダムに選択し，交叉を行う (CIAT3)。

CIAT1 の方法は，従来の方法であり，CHC の方法に近いものである。優秀な個体のみで交叉を行うので, 収束が早く，良い個体を効率に生成することができる。しかし，世代が増えるにつれ，優秀な個体の多様性が失われていくので，解が収束した場合に環境変化が起こると個体の進化が困難であると考える。本研究で提案する CIAT2 の方法は， AT で用いた方法に近い方法である。CIAT2，CIAT3 は交叉に劣等な個体を用いる場合があるので， CIAT1 に比べ個体の多様性を確保することができ，解が収束した場合においても個体が進化できるので，CIAT1 に比べ解探索を行うことができる。

次に，実験で用いる各アルゴリズムの設定を説明する。

- (個体群サイズ)100 個体
- (遺伝子コード長)50
- (最大世代数)1000 世代
- (SGA の交叉率) 0.95
- (SGA の突然変異率) 0.01
- (hyper mutation step を導入した GA の交叉率) 0.95
- (hyper mutation step を導入した GA の突然変異率) 0.01

- (hyper mutation step を導入した GA の置換率) 0.7
- (hyper mutation step を導入した GA の初期化間隔) 25 世代
- (CIAT における分割比率) 優秀:劣等 7:3

交叉率は一般的に高いものが良いとされ，通常 0.6〜0.95 の間をとる [7]。今回は，CHC，CIAT の交叉率が 1 であるため，比較することを考慮し SGA，hyper mutation step を導入した GA の交叉率は 0.95 を選択した。突然変異率はあまり高すぎると解が収束しにくくなるため，SGA と hyper mutation step を導入した GA の突然変異率は，予備実験の結果 0.01 と設定した。また，hyper mutation step を導入した GA の置換率，初期化間隔は予備実験の結果より，今回構築したアルゴリズムでは 0.7，25 世代が環境変化後でもっとも良い結果を示し，環境変化前でも固定の環境で行った実験の結果とさほど変わらない結果を示したため，0.7，25 世代と設定した。CIAT における分割比率は，8:2 や 7:3 が良いとされ [4]，今回は劣等解の割合を多くすることで，環境変化後に個体の進化がより望まれる 7:3 を選択した。

5.2 実験の結果

表 1 に，実験の結果を示す。各アルゴリズムに対して，1000 世代で計算した結果である。環境変化前は 500 世代時の解であり，環境変化後は 1000 世代時の解を示す。値として 15 試行での環境変化前後の最良，最悪解，平均値を掲載する。GA は最適解を探索するアルゴリズムなので，ここでは最良解に注目する。環境変化前では，どの環境に対しても CHC， CIAT1 が最も良い結果を示している。SGA はどの環境に対しても最も悪い結果を示している。環境変化後を見ると，環境 A では CHC と CIAT2 が，環境 B では CIAT1 と CIAT2 が，環境 C では CIAT2 が，環境 D では CIAT1 が最も良い結果を示している。しかし，環境変化後でも SGA はどの環境に対しても最も悪い結果を示している。

表 2 に，実験の総合結果を示す。各アルゴリズムに対しての 4 環境における最良解の平均，各環境に対して最も良い結果を示した回数 (最良結果回数) である。

まず表 2 に注目し，総合結果から考察を行う。

環境変化前では，CHC，CIAT1 が平均 17，最良結果回数 4 と最も良い結果を示している。これは，CHC，CIAT1 共に効率の良い選択，交叉を行うことで個体の進化が進むことでこのような結果になったと言える。次に，平均 16.75，最良結果回数 3 で AT と CIAT2 である。親の両個体を優秀個体とし交叉を行う CHC， CIAT1 に比べると，片親を優秀，片親をランダムに選択し交叉を行うため，平均が少し低い結果となっている。CIAT3 は，交叉の際に親個体をランダムに選択するため，劣等個体同士の交叉が確率的に発生してしまい，個体の進化が進まず解の探索が停滞する可能性があるため，CHC，AT，CIAT1，CIAT2 に比べると平均が低い結果となってしまう。

表 1:各アルゴリズムによる実験結果

環境 A	環境変化前			環境変化後		
	最悪解	最良解	平均	最悪解	最良解	平均
SGA	10	14	11.33	8	12	9.87
CHC	15	18	16.93	11	15	13.13
HMS	11	15	12.8	9	13	11.13
AT	15	18	16.6	10	14	11.67
CIAT1	15	18	16.67	10	14	13
CIAT2	14	18	16.6	10	15	13.13
CIAT3	15	17	15.33	9	13	11.2

環境 B	環境変化前			環境変化後		
	最悪解	最良解	平均	最悪解	最良解	平均
SGA	12	17	14.2	6	9	7.8
CHC	18	19	18.2	10	13	11.6
HMS	14	18	15.4	9	12	9.73
AT	14	19	16.87	9	13	10.73
CIAT1	17	19	17.87	11	14	12
CIAT2	16	19	18	8	14	12
CIAT3	14	18	16.27	9	12	10.13

環境 C	環境変化前			環境変化後		
	最悪解	最良解	平均	最悪解	最良解	平均
SGA	8	11	9.53	6	10	8.47
CHC	14	17	15.2	11	14	13.2
HMS	12	14	13.27	10	12	11.07
AT	13	16	14.87	11	13	12.07
CIAT1	14	17	15.4	12	14	12.93
CIAT2	14	16	15.4	10	15	12.13
CIAT3	12	15	13.73	8	13	10.87

環境 D	環境変化前			環境変化後		
	最悪解	最良解	平均	最悪解	最良解	平均
SGA	8	11	9.47	8	12	9.8
CHC	12	14	12.87	13	16	14.6
HMS	9	12	10.4	11	13	11.87
AT	11	14	13.07	11	15	13.27
CIAT1	12	14	12.73	14	17	14.67
CIAT2	12	14	12.87	11	16	14.27
CIAT3	11	12	11.2	12	14	13.07

表 2:実験の総合結果

	環境変化前		環境変化後	
	平均	最良結果回数	平均	最良結果回数
SGA	13.25	0	10.75	0
CHC	17	4	14.5	1
HMS	14.75	0	12.5	0
AT	16.75	3	13.75	0
CIAT1	17	4	14.75	1
CIAT2	16.75	3	15	3
CIAT3	15.5	0	13	0

hyper mutation step は定期的に個体の初期化を行うことで多様性が維持できるため進化が促進される。しかし，交叉の際に親個体の選択がランダムであることやランダムに個体を選択して初期化するため，進化途中の個体を初期化してしまう可能性があるなどの理由から CHC，AT，CIAT1，CIAT2，CIAT3 に比べ平均が低い結果となっている。 SGA は平均 13.25 と最も低い結果となり，最も高い平均に比べ 3.75 小さい。最

も低い結果の要因として，ルーレット選択であるため確率的に様々な個体を残すこと，一点交叉であるため個体の進化が小さいことなどが考えられる。

環境変化後では，CIAT2 が平均 15，最良結果回数 3 と最も良い結果を示している。 CIAT2 は劣等個体を意図的に保存することで多様性の維持を行い，さらに効率の良い選択，交叉に加え，確率的に劣等個体を交叉に含めることで解が収束した場合でも個体が進化できる可能性があるため，このような結果になったと言える。次に平均 14.75 で CIAT1，平均 14.5 で CHC である。これらは共に最良結果回数 1 である。CHC は解が収束した場合に突然変異を行い，この突然変異の方法が個体に多様性を持たせる役割を果たす。しかし，CIAT1 は意図的に劣等個体を保存することで多様性を維持するため，収束が発生しないと多様性を持たせることができない CHC に比べ，安定して環境変化に適用できるため， CIAT1 の方が平均で良い結果になったと考える。 AT は，劣等個体を意図的に保存することで多様性を維持するが，効率の良い選択，交叉を持つ CHC，CIAT1， CIAT2 と比べると平均が少し低い結果となっている。hyper mutation　step，CIAT3 は多様性は維持されるが，確率的に親個体を選択し交叉を行うため，効率的に良い個体の遺伝子が継承されないため，平均 13， 平均 12.5 と低い結果になっている。SGA は，環境変化後に対してもルーレット選択によりわずかながら個体に多様性があるものの，世代が進むことで個体が収束してしまうため，最も悪い結果となっている。

各環境に対して各アルゴリズムが残す結果は, 全体から見た要因が大きく関わる. しかし, 環境によって結果が異なるため, 各環境に対して表 1 で考察を行う.

環境変化前では，各アルゴリズムが示す結果が最も良いのは環境 B である。各アルゴリズムが示す結果が悪いものは環境 D となっている。環境 B は餌の連なりが多く，また密集している餌が多いため餌を見つけやすく，各アルゴリズムの結果に大きな差がない。また，環境 D は餌の連なりをなくし，バラバラに配置したため，餌の探索が困難になり，各アルゴリズムの結果が悪いものとなっている。その中でも，良い個体の遺伝子を確実に継承する CHC，AT，CIAT1， CIAT2 は他のアルゴリズムに比べ良い結果を示すことができた。次に，餌の連なりがある環境 A，B， C に注目すると，環境 B が一番良い結果を示し，次に環境 A，そして環境 C である。環境 B は先ほど説明したように，餌に連なりがあり密集している餌が多いため各アルゴリズムが各環境に対して一番良い結果を示している。環境 A と環境 C では，環境 A は連なりのない単体の餌が連なりのある餌の近くにあるため発見しやすいのに対し，環境 C では端に配置されているなど連なりのない単体の餌が分散されて配置されていることから発見しにくい。そのため，連なりのない単体の餌が発見しやすい環境 A の方が，各アルゴリズムは良い結果を示していると考える。

環境変化後では，環境 A， B は数個の連なりのある餌があり，環境 C は分散され連なりがないように餌が配置されている環境である。これら 3 つの環境はほとんどの餌が分散され配置されるため，餌を探索しづらい環境である。環境 D は，餌に連なりをもたせ配置しているため餌を発見しやすく，環境変化前から見ると，餌探索の難易度の高い環境から難易度の低い問題へと変化させたものである。結果を見ても，環境 D のみ最良解の値が他の環境に比べ高くなったアルゴリズムがほとんどである。4 つの環境を通し， CHC は効率の良い選択，交叉により個体が進化し，突然変異の方法により多様性が確保され，CIAT1， CIAT2 は劣等個体を意図的に残したことで多様性が維持され，良い個体の遺伝子を確実に継承させることで個体が進化するため，安定してどの環境においても比較的良い結果を残すことができたと考える。

実験の結果より総合結果を見ると，環境変化前では，効率の良い選択と交叉を持つ CHC と CIAT1 が優秀な個体の遺伝子を確実に継承させ個体を進化させるため，平均的に良い結果になったと考える。環境変化後では，意図的に劣等個体を保存し多様性を維持し，確率的に劣等個体を交叉に含める CIAT2 が，効率的に個体を進化させることができ環境変化に適応できるため，平均的に良い結果になったと考える。

6　おわりに

本研究では，従来提案されていた CIAT の交叉法を変更したアルゴリズムを提案し，SGA, CHC, hyper mutation step を導入した GA, AT, 従来の CIAT と比較することで考察を行った。総合的な結果から環境変化前では，効率的に優秀な個体を残す CHC や従来の CIAT に及ばないものの，さほど差はなく，AT と同等の結果を示している。また，環境変化後ではもっとも良い結果を示していることから，環境変化後に対して劣等個体を交叉に含めるアルゴリズムは有効であると考える。しかし，提案したもう一つの方法であるランダムに親個体を選び出し交叉させる方法は進化が進まない恐れがあるため，このことから片親に優秀な個体を含めることは重要である。

今後の研究としては，自然界では強い個体ほど交叉を行うことができるため，今回のように片親を固定し，片親をランダムに選択するのではなく，ルーレット選択などを用いて適応度の高い個体ほど親として選ばれやすいなどとすると，より生物の成り立ちを模倣することができる。ルーレット選択であるため，確率的に劣等な個体も交叉できるため，環境変化に適応できる。また，AT に多様性を維持できる新たなアルゴリズムを組み込むことで，さらなる環境変化に対する適応が見込まれる。

参考文献

[1] 茨木 俊英, 福島 雅夫, ”最適化の手法”, 共立出版, 1993.

[2] 堀場 雅夫, ”仕事のできる人できない人”, 三笠書房, 2000.

[3] Fumiaki Makino, ”Worker Ant’s Rule-Based Genetic Algorithms Dealing with Changing Environments”, IEEE Mid-Summer Workshop on Soft Computing in Industrial Aplications, pp.117-121, 2005

[4] Keigo Hobara, ”Worker Ant’s Rule-Based CHC Dealing with Changing Environments”, IEEE Three-Rivers Workshop on Soft Computing in Industrial Aplications, pp.97-101, 2007

[5] 坂和 正敏, ”遺伝的アルゴリズム”, 朝倉書店, 1995.

[6] John J. Grefenstette, ”Genetic algorithms for changing environment”, Proceedings of the Second Conference on Parallel Problem Solving from Nature, Brussels, Belgium, pp.28-30, 1992

[7] 伊庭 斎志, ”遺伝的アルゴリズム”, 医学出版, 2002.

[8] 伊庭 斎志, ”遺伝的プログラミングと進化論的な学習, 人工知能学会誌 vol.9 No.4”, pp.36-41, 1993.