

# 環境変化の特性および多目的最適化との類似性を考慮した動的環境への進化的な適応法

山崎 和子<sup>†</sup> 北風 和久<sup>†</sup> 関口 益照<sup>†</sup>

この研究は、適応度が外生的に時間変化するときの適応の問題を GA により取り扱う。この問題を以下の 2 つの視点から論ずる。(1) 多目的最適化との類似性と (2) 環境変化の特性。(1) に基づき提案された方法では、それぞれの個体は、染色体のほかに、世代の適応度を記憶するための時系列を持ち、変化する適応度  $\{f_t, f_{2t}, f_{3t}, \dots\}$  に対してパレート最適な個体を選択し、次世代に残す。(2) に基づいた方法では、環境が連続的に変化するとき、その特性を積極的に用いて、近傍探索により多くのコスト(適応度評価)を費やす。ナップザック問題や Branke のテスト関数を用いた実験をし、良い結果を得た。

## Evolutionary Adaptation Method against Dynamic Environments Considering Characteristics of Changes and Similarities to Multi-objective Optimization

KAZUKO YAMASAKI,<sup>†</sup> KAZUHISA KITAKAZE<sup>†</sup>  
and MASUTERU SEKIGUCHI<sup>†</sup>

This paper studied the problem of the adaptation under the exogenous changes of the fitness landscape using GA. This problem was discussed from the following two aspects. (1) Characteristics of environmental changes and possibility of adaptation (2) Similarity to multiobjective optimization. In the proposed method based on (1), each individual has the time series to memorize the fitness at each generation besides the chromosome, and the best Pareto individuals in changing fitness landscape  $\{f_t, f_{2t}, f_{3t}, \dots\}$  etc are carried into the next generation. In the proposed method based on (2), when the environment changes continuously, it spends a lot of cost on local search. The experiments of knapsack problem and of using Branke's test functions were shown.

### 1. 現在までの仕事とその問題点

進化型計算の分野では、「自然淘汰」をモデルとし、目的あるいは価値規準の存在を前提として、有用な情報を記憶伝搬する過程の研究が一般的である。近年、その研究の広がりとともに、多目的最適化や雑音のある環境など、何をもちて有用とするか、価値規準の存在の仕方そのものに関わる問題にも関心が持たれるようになった。その中の 1 つに、動的な問題がある。一般に、動的といわれるとき、適応度関数が内生的に変化する場合と、外生的に変化する場合があるが、ここでは、外生的に変化する場合を扱う。

この分野での現在までの仕事は、通常、次のように分類される<sup>7),10)</sup>。

#### (1) 多様性を用いた研究

- (a) 多様性を環境変化時に増すもの  
一定割合の遺伝子をランダムに初期化するもの (Hypermutation<sup>9)</sup>) 下位ビットから突然変異率を上げていって表現形での局所探索の範囲を広げていくもの (Variable Local Search) など。
- (b) 多様性をつねに保持するもの  
選択圧を弱くするもの、クラウドニングやニッチングをするもの、選択や交差を制限するもの (島モデル, 発散モデル, タグビットを持つもの)<sup>9)</sup>, ThemodynamicGA<sup>18)</sup> など。

#### (2) 記憶を用いた研究

- (a) 暗黙に冗長な記憶として持つもの  
ディプロイド (Goldberg<sup>13)</sup>) やハプロイド, 多層構造

<sup>†</sup> 東京情報大学総合情報学部  
Faculty of Informatics, Tokyo University of Information Sciences

- (b) 明示的な記憶として持つもの  
記憶のための集団と探索のための集団を別に持つもの、集団の1部を定期的に保存しておくもの( Branke<sup>8)</sup>, Ursem<sup>20)</sup>)

しかし、1995年頃から、盛り上がりを見せたこの分野も、今は行き詰まりの感がある。その理由として考えられることを、次に示す。

- (1) 動的な環境への適応の不明確さ  
静的環境では、どのような問題でも、適応度地形の最大値へ、無限の時間を使えば原理的には到達可能で適応の意味は最適値への到達ということで明確である。ただし、実際問題では、計算時間と最適性のトレードオフが問題となる。それに対して、動的環境では、時間の制限は本質的であるから、何が適応した状態なのか明確ではない。さらに、適応の目的が変化するため、ある世代でよく適応することが次の世代の適応を困難にすることがある。つまり、しばしば、ある世代の最適性と次の世代の最適性の間には、トレードオフが存在する。このような状況の中で、どのような状態が動的な環境へ適応した状態か、明確に述べることができるのであろうか？
- (2) バイナリ空間で適応度関数の変化を扱う困難  
初期の動的環境下での研究はバイナリ空間で適応度関数を決めるパラメータを周期的に変化させて行われた<sup>13),17)</sup>。しかし、バイナリ空間の問題では、パラメータを少し変化させただけで、適応度関数が大きく変わる。したがって、周期的変化以外では適応することが望めない。それで、実数空間でパラメータの連続的な変化に対して適応度地形が連続的に変化するテスト関数が、Jong<sup>10)</sup>, Grefenstette<sup>14)</sup>, Branke<sup>8)</sup>らにより提案され、それをを用いて多くの研究がなされた。しかし、連続的に変化する環境のみが研究の対象とはいえない。
- (3) 実際の応用分野があまり知られていない。
- (4) この分野のキーとなる方法がない。

このうち、(1), (2)の問題について、2章で、それぞれ新たな視点から考察を行い、それに基づき、3章で、適応方法を提案し、4章で、それらを用いた実験の結果を示す。

2. 動的な適応法における2つの視点

2.1 第1の視点：多目的最適化と動的環境への適応の類似性

〈 保守的 ←→ 進歩的 〉

表 1 動的環境と多目的最適化の類似性  
Table 1 Similarity with multiobjective optimization.

多目的最適化 → t	動的環境 → t	動的環境+再現性 → t
$f_{1\tau}$	$f_{1\tau} f_{2\tau} f_{3\tau} f_{4\tau} f_{5\tau}$	$f_{1\tau} \text{ --- } f_{1\tau} \text{ --- } >$
$f_{2\tau}$		$\text{--- } f_{2\tau} \text{ --- } f_{2\tau} \text{ --- } >$
$f_{3\tau}$		$\text{--- } f_{3\tau} \text{ --- } >$
$f_{4\tau}$		$\text{--- } f_{4\tau} \text{ --- } >$
$f_{5\tau}$		$\text{--- } f_{5\tau} \text{ --- } > f_{5\tau}$

〈 安定 ←→ ハイリスクハイリターン 〉

〈 長期的洞察 ←→ 短期的洞察 〉

〈 流れに乗る → 流れに逆らう 〉

我々の日常生活の中には、このような環境が変化することに起因した様々なトレードオフが存在し。我々は日々意志決定を迫られている。これらは、ある世代の最適性と次の世代の最適性と間のトレードオフに起源する。動的環境では目的関数(適応度地形)  $F = \{f_{1\tau}, f_{2\tau}, f_{3\tau} \dots\}$  は  $\tau$  の間隔を置いて、相次いで現れる。一方、多目的最適化の問題では、複数の目的関数は同時に、現れる(表1)。しかし、以下の(A), (B)の条件が満たされるとき、動的環境の問題に多目的最適化と同じ手法を用いることができる。

- (A) 環境変化の間隔のスケール  $\tau$  が、適応の速度(テークオーバータイム)より小さくなる。
- (B) 現在の環境が再び将来再現される。

つまり(A)のとき、変化する  $f_{1\tau}, f_{2\tau}, f_{3\tau} \dots$  に1つ1つ適応する以外の方法をとる必要が生じる。このとき、(B)なら表1右のように、多目的最適化と同様の状況が生じ、多目的最適化で使われる手法を応用することができる。次に、(A)の条件が非常に一般的であることを述べる。一度テークオーバーが起きると、適応度関数が変化したとき、新しい最適値を探索するのに時間を要するため、今まで開発された様々な方法では、「多様性を保つ」(Mori<sup>18)</sup>, Sarma<sup>19)</sup>)や「多様性を生成する」(Cobb<sup>9)</sup>)などを行いテークオーバーが起きないようにしている。このような状態は、適応の速度より小さいタイムスケールにシステムをとどめているといえる。つまり、動的な環境への適応に多目的最適化と同様の状況が生ずるのは一般的である。

多目的最適化では、唯一の最適解は一般的に存在せず、パレート最適解の集合を求める。つまり、動的な環境への適応の不明確さは、多目的最適化と動的環境への適応の類似性から生じており、このとき、最適解は一般的に存在しないので、パレート最適解の集合を求めるのが合理的である。

## 2.2 第2の視点：環境変化の特性

先に述べた、バイナリ空間で適応度関数の変化を扱うときの困難は、次のようにも考えられる。たとえば、頻繁に大きく不規則に飛び回っている最適値への適応は、ランダムに初期化した状態が平均的に一番適応度が高いのは明らかで、このような環境への適応を考えるのは無意味である。しかし、だからといって、連続的に変化する環境のみが研究の対象という結論には至らない。静的環境と違って、無限の時間をかけたとしても、どんな動的環境にでも適応できるというのではない。これは、「連続的な変化という環境の特性があるときには、適応する方法を開発することが可能だ」ということだと考える。つまり、「適応度関数の変化に、持続的な何か変化の特性があって初めて適応する方法の開発が可能で、動的な環境への適応が意味を持つ。持続的な変化の特性に対して、各適応方法は開発されるべきである」と考える。適応することができる典型的な環境変化の特性とそれに対する適応方法を以下に示す。

### (1) 再現性

再現性とは、 $n (> 1)$  世代前に現れた適応度地形の形状のなんらかの特徴が再び現れる特性を意味する。これらの環境に有用な適応方法は明らかに記憶を含んだものである。

### (2) 連続性

連続性とは、適応度地形の形状の変化が、1世代前と比べて小さいことを意味する。これらの環境に対する有用な適応方法の1つは適応度地形上の近傍探索演算子である。さらにいえば、理論的には、環境変化に対応する探索演算子さえあれば、環境が連続的に変化しなくても、連続性に対して開発された方法がそのまま使える。

### (3) 希少性

希少性とは、適応度地形の形状の変化が、大きく不規則であっても、変化が稀にしか起こらないならば適応することができることを意味する。このようなカタストロフィーに対する有用な適応方法は、多様性を保つあるいは多様性を瞬間的に生成する方法である。

### (4) 予測性

予測性とは、適応度地形の次の変化をなんらかの方法で予測することが可能であることを意味する。これらの環境に対する有用な適応方法は個別の問題による（たとえば Hemert<sup>15)</sup>）。

従来から分類されている、記憶に基づいた方法と多様性に基づいた方法のほかに、近傍探索に基づいた方法や予測に基づいた方法などを考えることは有用である。つまり、バイナリ空間で適応度関数の変化を扱う困難は、何も動的な環境に対して、バイナリ空間が固有に持つ困難というわけではない。静的環境と違って、無限の時間をかけたとしても、一般的に、適応の速度より速いか同程度の環境変化には適応が困難で、なんらかの適応できうる環境変化の特性があるときのみ適応できるということである。

## 3. 2つの視点に基づいた適応法

### 3.1 多目的最適化との類似性に基づいた適応法

2.1節で述べた(A)、(B)を満たす問題に対しては多目的最適化と類似の方法が適用可能である。多目的最適化での複数の目的関数は、動的最適化での変化する目的関数  $f_{1T}, f_{2T}, f_{3T} \dots$  に相当する。多目的最適化と同じように、これらに対するパレート最適解の集団を、多目的ランキングにより求める。そのためには、過去の適応度の値を各個体が記憶しておかなければいけない。

記憶の仕方について、従来の記憶に基づいた方法との差異を以下に述べる。従来の方法では、2倍体の劣性の遺伝子の中に過去の記憶を残すもの(Goldberg<sup>13)</sup>)、適応度のほかに、遺伝子の年齢や探索空間での遺伝子の位置を用いて選択をし、遺伝子を保存しているもの(Branke<sup>8)</sup>)、集団をいくつかに分割し、それぞれが探索空間の1部を受け持つもの(Ursem<sup>20)</sup>)などがある。これらの研究では、選択に適応度や年齢や遺伝子の位置のみを用い、遺伝子そのものを保存しているためたとえば(Ursem<sup>20)</sup>)では遺伝子の数は1,000個にも達する。提案方法では、過去の適応度をそれぞれの個体の内部に時系列として保存し、より詳細な情報によって選択を行ため、遺伝子の数は増やす必要はない。したがって、遺伝子は長さ  $D$  の染色体のほかに最大長  $L$  の実数値列である過去の適応度の時系列を持つ。さらに、個体は複数世代にわたって生きてい

パレート最適性とは、どこかを悪くすることなしにこれ以上良い結果が得られない状態、つまり、 $\{f_1, f_2, f_3, \dots, f_n\}$  において  $f_s(x) \{s \neq t\}$  を下げることをなしに  $f_t(x) \{t = 1, n\}$  を上げることができないことをいう。以下の2つが満たされるとき

$$\forall t = 1, 2, \dots, n \quad f_t(x) \geq f_t(y)$$

$$\exists t = 1, 2, \dots, n \quad f_t(x) > f_t(y)$$

のとき、解  $y$  が解  $x$  に支配されるいう。どの解にも支配されない解の集合をパレート最適解の集合という。

バイナリ空間の問題は連続性という適応できうる環境の特性がないだけである。動的なスケジューリングのような問題でも、何かある特性のある変化に対しては、その特性を利用した適応方法が開発可能ではないかと推測する。

なければ、適応度の記録ができないため、世代ギャップは1以下とする。

しかし、この方法の従来の記憶に基づいた方法との最も大きな差異は、記憶の仕方ではない。「動的な環境への適応には不明確さがあり、どのように適応するかを選択する必要がある」ということを指摘し、その選択を可能にしていることである。

パレート最適集団の保持の利点をまとめると以下のようなになる。

- (1) 多様性を維持するために、パレート最適集団を保持するのは、最も合理的である。
- (2) 多様性をなんらかの形で維持する実験では、適応度の集団平均値が性能の比較において意味を持たないことから、通常は、オフライン性能  $\frac{1}{T} \sum_{t=1}^T \max_i f_{t,i}$  (ここで  $f_{t,i}$  は個体  $i$  の世代  $t$  における適応度)(最良個体の適応度の時間平均)で計測される。この提案方法では、最良個体に限らず、パレート最適集団を保持しながら、最も安定的な個体や最もハイリスクハイリターン個体の個体などを、必要に応じてモニタすることが可能である。
- (3) 集団自体をパレートフロントに均一に分布したのから、安定的な個体やハイリスクハイリターン個体の分布を厚くすることも可能である。

したがって、アルゴリズムは以下のようなになる。

- (1) 適応度評価
- (2) それぞれの個体は適応度を時系列に記録
- (3) 多目的ランキング<sup>16)</sup>
- (4) シェアリング<sup>12)</sup>
- (5) 世代ギャップの割合の個体を選択、交差、突然変異
- (6) 最良個体あるいは目的にあった個体をモニタ
- (7) (1)へ戻る

多目的ランキングでは、ある個体を支配する集団内の別の個体の個数をランクとしてランキングを行う。

$f_t$  は世代  $t$  の適応度地形、 $F = f_1, f_2, f_3, \dots$  を多目的最適化問題における多目的関数と見なす。個体の支配関係を次のようにして定める。次の条件が満たされるならば、個体  $j$  は個体  $i$  に支配される。

$$\forall t = 1, 2, \dots, L \quad f_{t,i} \geq f_{t,j}$$

$$\exists t = 1, 2, \dots, L \quad f_{t,i} > f_{t,j}$$

すべての個体の組合せで支配関係を定め、個体  $j$  を支配する個体の個数を個体  $j$  のランクとする。

多目的ランキングを疑似コードで書くと以下のようなになる。

```
for(i=0; i<numberOfAgents; i++){
  for(j=0; j<i; j++){
    int di=0,dj=0;
    for(t=0; t<L; t++){
      if(agent[i].memory[t] > agent[j].memory[t]) di++;
      if(agent[i].memory[t] < agent[j].memory[t]) dj++;
    }
    if( di==L && dj<L ) rank[j] ++;
    if( di<L && dj==L ) rank[i] ++;
  }
}
```

シェアリングは次のように行う。

```
for(i=0; i<numberOfAgents; i++){
  for(j=0; j<i; j++){
    d[i][j]=0;
    d[i][j]+=distance(agent[i],agent[j]);
    s[i][j]=1.0 - (d[i][j]/σ)**α;
  }
  for(i=0; i<numberOfAgents; i++){
    z=0;
    for(j=0; j<numberOfAgents; j++) z+=s[i][j];
    rank[i]/=z;
  }
}
```

ここで、distance は個体間の距離を求める関数で、問題ごとに定める。

### 3.2 環境変化の特性に基づいた適応法

再現性、希少性、予測性に基づいた適応法は、すでに述べた、記憶を用いた方法、多様性を用いた方法、予測を用いた方法が、それに相当する。それで、ここでは、今まで積極的に利用されてこなかった連続性を利用した適応法を提案する。すでに述べたように、これらの環境に対する有用な適応法の1つは適応度地形上の近傍探索演算子である。また、近傍探索としては確率的なGAの突然変異より決定論的な山登り法が効率的であり、また、連続変化に対しては、近傍探索に多くのコストを使った方が、良い結果が期待される。ここでコストは、主に適応度の評価にかかるとし、1個体1回の適応度の評価を単位として換算する。たとえば100個の個体が、近傍探索をせずに、それぞれ1世代に1回評価を行い、50世代ごとに環境が変化する

文献 2)~5), 22), 23) では、適応度関数として、文献 11) で提案された多目的最適化のテスト関数を、交互に周期的にとる環境の中で、平均値の高い個体または最高値の高い個体をパレートフロント上に厚く分布させ、安定的な個体またはハイリスクハイリターン個体の解が得られることなどを示した。

場合 (noLocal) と 100 個の個体がそれぞれ 1 世代に 5 回近傍探索を行い、したがって 5 回評価を行い、10 世代ごとに環境が変化する場合 (Local) とでは、同じ 5,000 回の評価ごとに環境が変化しているが、後者のほうが良い結果が期待できる。したがって、アルゴリズムは以下ようになる。

- (1) 近傍探索および適応度評価
- (2) ランキング
- (3) 選択, 交差, 突然変異
- (4) 最良個体をモニタ
- (5) (1) へ戻る

#### 4. 実験

##### 4.1 実験 1: 多目的最適化との類似性に基づいた適応法

ナップザック問題で、制限重量が定期的にランダムに変化する実験を行う。この問題は、ある世代で価値の高い組合せほど、制限重量が低くなったときに、それを超えやすいというトレードオフを含んだ問題である。個体  $i$  の適応度  $fitness[i]$  は以下のように決める。

$$\text{if } \left( \sum_{k=1}^{k=D} x[k] * weight[k] < limit \right) \\ fitness[i] = \sum_{k=1}^{k=D} value[k] * weight[k]; \\ \text{else } fitness[i] = 0;$$

ここで、 $x[k]$  は荷物を袋に入れるかどうかを示すビット列、 $weight[k]$ 、 $value[k]$  はそれぞれその荷物の重量と価値であり、 $[1,100]$  の乱数により生成した。制限重量  $limit$  は総重量の 30% から 70% の範囲で 10 世代ごとに乱数により決めた。個体数 300、遺伝子長 100、世代ギャップ 0.2、交差確率 1.00 突然変移率 0.001 記憶列の長さを 300 とした。ここでは、環境変化はランダムに生じるため、記憶列に十分な長さというのは存在しない。5,000 世代の実験を行ったので、その程度の世代で、適応度の急激な降下が生じないように、記憶列の長さを実験的に決めた。また、シェアリングの係数は  $\alpha = 0.2$ 、 $\sigma = 1000$ 、距離として総重量の差の絶対値を用いた。比較のため、表 2 の Pareto, Average, noMemory 集団で実験を行った。

図 1 にそれぞれの集団の最良個体の適応度の時間変化を示す。Pareto 以外の集団では、制限重量が低くなったときに適応度の急激な低下がしばしば見られる。また、乱数の異なる種で行った 50 試行で平均した適応度を表 3 に示す。多目的最適化との類似性に基づいた適応法をする Pareto 集団が良い結果を示している。

この理由を明示するため、このときの個体の分布を図 2 に示す。Pareto 集団では、パレート最適面に

表 2 比較のために、実験で用いた集団  
Table 2 Groups used in experiments.

Pareto	多目的ランキングによる集団
Average	適応度の記憶列の平均値によって ランキングした集団
noMemory	適応度の記憶列を用いず、 現在の適応度によってランキングした集団
Local	近傍探索を行う集団
noLocal	近傍探索を行わない集団

Average, noMemory はランキングの方法以外は Pareto と同じ方法を用いる (シェアリングも行う)。

表 3 ナップザック問題における平均適応度  
Table 3 The values of average fitness.

Pareto	3,768
Average	3,149
noMemory	2,208

沿って個体が様に分布し、どのような制限重量のときでも、最適解に近い解を保持しているのが分かる。この場合、一度でもいいから良い結果を示す個体は、他の個体から支配されず、したがって集団内に保持される。環境変化にともない交替で最良個体となり、良い結果が得られている。Average 集団では、総重量の高い個体は、制限重量が低くなったとき、適応度が 0 となるため平均の適応度が低くなり、淘汰される。また、総重量の低い個体も、制限重量の高い状態が続くと、他の個体と比較して平均の適応度が低くなり、淘汰される。したがって、Pareto 集団より、狭い範囲に分布するようになる。noMemory 集団はまったく適応することができない。Pareto 集団が最も多様性を保ち、しかも、無駄のない合理的な保ち方をしていくことが分かる。

##### 4.2 実験 2: 多目的最適化との類似性および環境変化の特性に基づいた適応法

###### Branke のテスト関数

Branke<sup>(6),(8)</sup> で提案された、テスト関数を用いる。これは、 $D$  次元実数空間にピークが複数存在し、ピークは幅と高さ而变化しながら、連続的に移動をする。このテスト関数のピークの高さと幅と位置は、最小値と最大値 ( $minheight \leftrightarrow maxheight$ ,  $minwidth \leftrightarrow maxwidth$ ,  $0 \leftrightarrow 100$ ) の間を決まった歩幅 ( $height\ severity$ ,  $width\ severity$ ,  $vlength$ ) でランダムウォークをする。ここでは、5次元空間  $[0:100]$  上に 50 個のピークを作り、ピークの形として、コーン型とコーシー分布型の 2 種類を用いる。 $(x_{p,1}(t), x_{p,1}(t), \dots, x_{p,5}(t))$ ,  $p = 1, 2, \dots, 50$  を 5次元空間上のピークの位置として、適応度関数を

Pareto 集団と同じようにシェアリングをしている。

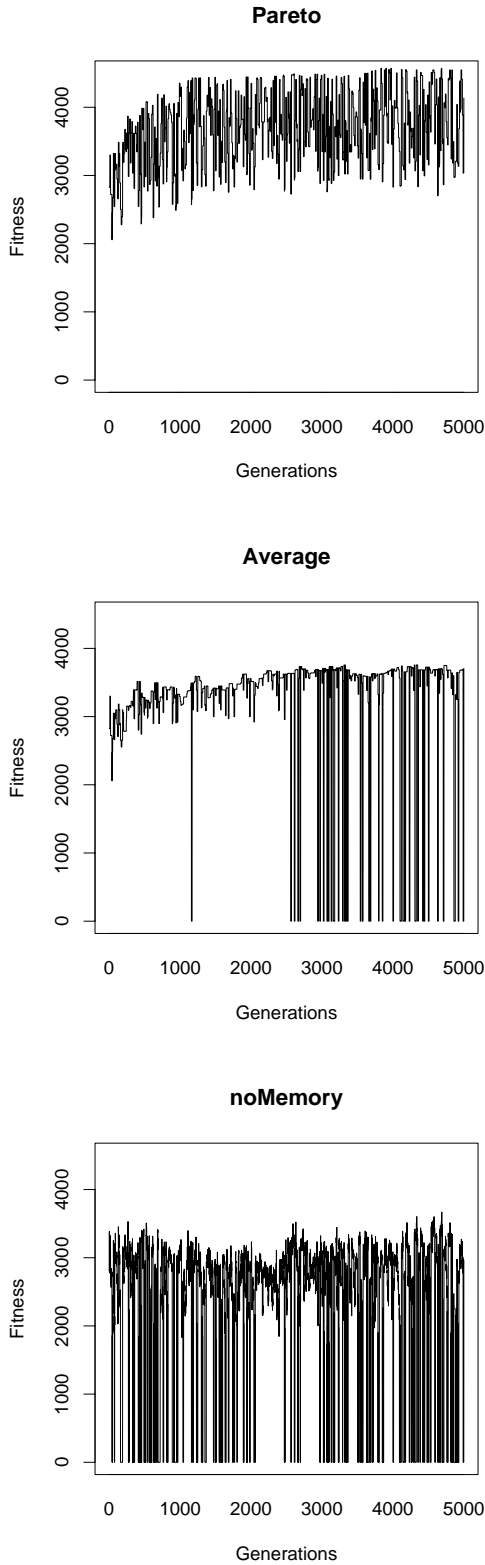
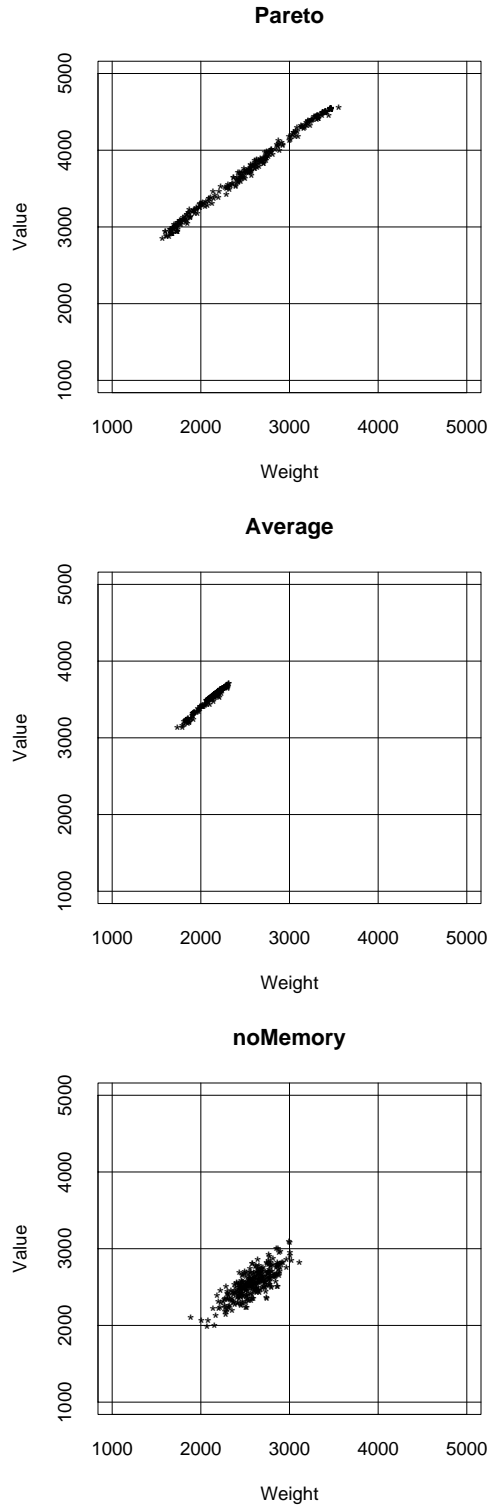


図 1 ナップザック問題における、最良個体の適応度の世代変化  
Fig. 1 Fitness vs. generation.



縦軸と横軸がそれぞれ、袋に入った荷物の総価値と総重量、各々の点が、5,000 世代における個体を表す。

図 2 ナップザック問題における個体の分布  
Fig. 2 The distribution of agents.

以下のように定める .

● コーン型

$$func(x_1, x_2, x_3, x_4, x_5, t) = \max_p [height_p(t) - width_p(t) \sqrt{\sum_{d=1}^5 (x_d - x_{p,d}(t))^2}]$$

● コーシー分布型

$$func(x_1, x_2, x_3, x_4, x_5, t) = \max_p [height_p(t) / (1.0 + width_p(t) \sum_{d=1}^5 (x_d - x_{p,d}(t))^2)]$$

図 3, 図 4 は, それぞれコーン型, コーシー分布型の関数で, 見やすくするため, ピークが 2 次元上に 5 個ある場合についての図を例示する. コーン型では, 性能を比較するため, テスト関数のパラメータを, 文献 6), 8) のシナリオ 2 とまったく同一に設定し, コーシー分布型では, シナリオ 1 とまったく同一に設定して実験を行った. それらのパラメータは, コーン型 (シナリオ 2), コーシ分布型 (シナリオ 1) それぞれについて minheight 30, 30 maxheight 70, 70 minwidth 1, 0.0001 maxwidth 12, 0.2 lambda 0, 0 vlength 1, 1 height severity 7, 7 width severity 1, 0.01 である .

これらの関数は, ピークの高さと幅において再現性を持つ. このような連続性と再現性を持った適応度地形の変化に対し, 3.1 多目的最適化との類似性に基づいた適応法と, 3.2 環境変化の特性に基づいた適応法で示した 2 つ方法を同時に適用する. 近傍探索としては, バイナリコーディングをし, 下位ビットのみを反転させるオペレータを用いて近傍探索をさせた研究はあるが<sup>21)</sup>, 実数コーディングをした方が近傍探索には自然であると考える. 我々の 2 つ提案方法を同時に適用した場合の方法を示す .

- (1) 近傍探索および適応度評価
- (2) それぞれの個体は適応度を時系列に記録
- (3) 多目的ランキング
- (4) シェアリング
- (5) 世代ギャップの割合の個体を選択, 交差, 突然変異
- (6) 最良個体あるいは目的にあった個体をモニタ
- (7) (1) へ戻る

比較のため, 表 2 の Pareto-noLocal, Average-noLocal, noMemory-noLocal, Pareto-Local, Average-Local, noMemory-Local の 6 つの集団で実験を行った. 実験のパラメータは, 個体数 100, 遺伝子長 5, 世代ギャップ 0.1, 交差確率 0.5, 突然変異率 1.0, 突然変異の標準偏差 10, 近傍探索の回数 5,

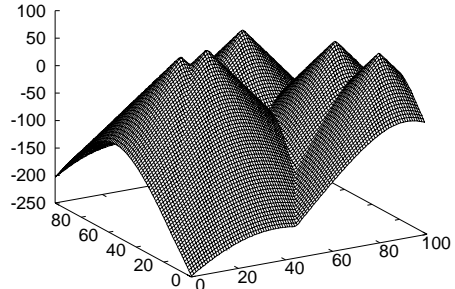


図 3 コーン型のテスト関数  
Fig.3 Corn type test function.

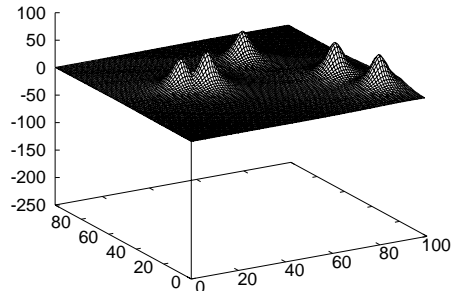


図 4 コーシー分布型のテスト関数  
Fig.4 Cauchy type test function.

近傍探索の標準偏差 1, 記憶列の長さ 250 である. ここでは, ピークの高さは上限と下限の間でランダムウォークをするので, ピークの高さに再現性があり, この再現の周期より長く記憶列の長さを定める必要がある. ピークの上限下限の差が 40, 標準偏差 7 の正規分布に従い 1 ステップごとに上下をするので記憶列の長さ  $L$  は,  $80 < 7\sqrt{L}$  の必要がある. これに基づいて  $L$  を定めた. すべての実験で, 1 試行 3,000 世代の実験を, 乱数の異なる種で 50 回試行を行った. その平均した結果を表 4 に示す. 表の中で, 実験の誤差は現在の最良個体の適応度から最適値を引いたものである. (1) Pareto と Average 集団の結果の差が多目的ランキングを使った効果である. (2) Local と noLocal 集団の結果の差が, 近傍探索をし, 連続性という環境変化の特性を積極的に利用した効果である. (3) また, Pareto, Average と noMemory 集団の結果の差は, 記憶列を用いることにより, この関数が, ピークの高さと幅において再現性を持つことを利用した効果である. 著しい差は (1) および (3) には見られず, コーン型, コーシー分布型とも高い性能の原因が主に近傍探索によることを示している. 多目的ランキングは, 実験 1 と異なり, 効果を上げていない. これは, どのピークも確率的に等価であるため, トレードオフが存在しないことによる. つまり, この

実験では、平均的に良い解 (Average) は、いずれかのピークの周辺にあって近傍探索によりそのピークを保持している解である。多目的ランキングによって保持する解 (Pareto) も同様の解であって、差が見られない。ピークを確率的に等価でなくし、高いピークはすぐ沈み、低いピークは長い間その高さを保つようにパラメータを設定した Branke のテスト関数で実験をした結果では、多目的ランキングの効果が見られた<sup>22)</sup>。

## 5. 結 論

適応度地形が外生的に時間変化する中での適応の問題を扱った。以下、この分野の研究では、現在、次のような問題点がある。

- (1) 動的な環境への適応の不明確さ
- (2) バイナリ空間で適応度関数の変化を扱うときの困難
- (3) 実際の応用分野があまり知られていない
- (4) この分野のキーとなる方法がない

このうち (1), (2) の問題点になんらかの形で解決、あるいは別の視点を与える適応方法を提案した。

### (1) 多目的最適化との類似性に基づいた適応方法

GA の適応の速度に近いかより速いタイムスケールで再現性のある問題では、動的な環境への適応の問題は、多目的最適化と類似性を持つ。

つまり、動的な環境への適応の不明確さは、多目的最適化と動的環境への適応の類似性から生じており、このとき、最適解は一般的に存在せず、パレート最適解の集合を求めるのが合理的である。

この適応法を用いて、ナップザック問題で実験を行った。この問題は、ある世代で価値の高い組合せほど、制限重量が低くなったときに、それを超えやすいというトレードオフを含んだ問題である。提案手法はパレート最適面に均等に分布した集団を保持し、この問題での効果的な適応を示した。

このように、多目的最適化との類似性に基づいた適応方法は、変化する環境にトレードオフがあるときに最もその威力を発する。そのような環境では、平均的に良い個体を維持するよりも、一度でもいいから良い結果を出す個体を保持し、環境変化にともない交替で担当させた方が良い結果が得られることを、示している。人間の組織に射影して考えると興味深い。

### (2) 環境変化の特性に基づいた適応法

動的な適応ができるためには、適応度地形の変化のしかたにある程度持続的な特性があって、それを利用して適応する必要がある。そのような特性とは、再現

表 4 実験の誤差

Table 4 Error.

noLocal		
	コーン型	コーシー分布型
Branke's	-4.605	
noMemory	-22.305	-19.971
Pareto	-18.665	-24.415
Average	-22.692	-26.455
Local		
	コーン型	コーシー分布型
noMemory	-3.967	-3.080
Pareto	-3.794	-2.743
Average	-3.732	-2.815

誤差 = 最良個体の適応度 - 最適値

性、連続性、希少性、予測可能性などが考えらる。それらの特性ごとに方法を開発することが、効率的な研究方法である。これらの特性のうち、今まで積極的に利用されていない連続性のある環境変化に対しては、近傍探索に多くのコストを使った方法を提案した。連続性と再現性のある環境変化の中での実験によると、我々の結果は、Branke<sup>7),8)</sup> の実験と比べて、高い性能を示し、その原因が、近傍探索によることを確かめた。

多目的最適化との類似性と環境変化の連続性に基づいた適応法では、過去一度でも最も高い適応度を示した個体はほかに支配されないから集団内に保持されるから、これにより、ピークはたとえ適応度が低くなくても追跡され、最近様々な人が研究しているマルチポピュレーション法<sup>20)</sup> と同様の効果がある。さらに、我々の提案方法では最適値をとるのが点でなく線や領域であってもその効果に変わりはない。

(3) の実際の応用分野があまり知られていないという問題点に関しては、Yamasaki<sup>1),24)</sup> で詳しく述べられている。ここでは、ファイナンスの分野には、動的な環境への適応として扱うのに適した豊富な問題が存在することを述べ、ここで提案した多目的最適化との類似性に基づいた適応方法を、日経 225 先物 5 分足の時系列解析に用いた。そして、日中足における 1 日周期の規則性をとらえ実際に収益を得る可能性のある結果を得た。

謝辞 この研究は東京情報大学の学術フロンティアと共同研究基金により資金の提供を受けた。東京情報大学の情報システム学科、環境情報学科の皆さんから貴重な助言をいただいたことに感謝します。

## 参 考 文 献

- 1) 山崎和子, 北風和久, 関口益照: 動的環境への適応と経済時系列への応用, 電子情報通信学会技術研究報告, pp.9-16 (2002).



- 2) 山崎和子: 動的環境下での適応における, 多目的最適化との類似性と, 個体内記憶の利用. *MPS シンポジウム予稿集*, pp.171–178, 情報処理学会 (2001).
- 3) 山崎和子: 動的環境での ga, 電子情報通信学会研究会報告, Vol.101, No.66, pp.55–62 (2001).
- 4) 山崎和子: 動的環境下での適応の持つ多目的最適化の側面, 第9回マルチ・エージェントと協調計算ワークショップオンライン予稿, ソフトウェア学会 (2000).
- 5) 山崎和子: 動的最適化問題が含むべき2つの設定, 第13回自律分散システム・シンポジウム予稿集, pp.383–388, 計測自動制御学会 (2001).
- 6) Branke, J.: <http://www.aifb.uni-karlsruhe.de/~jbr/movpeaks>.
- 7) Branke, J.: Evolutionary approaches to dynamic optimization problems: A survey, *1999 Genetic and Evolutionary Computation Conference Workshop Program GECCO*, pp.134–137 (1999).
- 8) Branke, J.: Memory enhanced evolutionary algorithms for changing optimization problem, *Proc. Congress on Evolutionary Computation CEC99*, pp.1875–1882 (1999).
- 9) Cobb, H.G.: An investigation into the use of hypermutation as an adaptive operator in genetic algorithms having continuous time-dependent nonstationary environments, Technical report, Naval research Laboratory, Washington, USA (1990).
- 10) De Jong, K.: *Evolving in a Changing World*, volume 1609 of *Lecture Note In Artificial Intelligence*.
- 11) Deb, K.: Multi-objective genetic algorithms: Problem difficulties and construction of test problems, *Evolutionary Computation*, Vol.7, No.3, pp.205–230 (1999).
- 12) Goldberg, D.E.: *Genetic Algorithms in Search, Optimization and Machine Learning*.
- 13) Goldberg, D.E. and Smith, R.E.: Nonstationary function optimization using genetic algorithms with dominance and diploid, *2nd International Conference on genetic Algorithms*, pp.59–68 (1987).
- 14) Grefenstette, J.J.: Evolvability in dynamic landscapes: A genetic algorithm approach, *Proc. Congress on Evolutionary Computation CEC99*, pp.2031–2037 (1999).
- 15) Hemert, J.V., Hoyweghen, C.V., Lukschandl, E. and Verbeeck, K.: A futurist approach to dynamic environments, *2001 Genetic and Evolutionary Computation Conference Workshop Program GECCO*, pp.35–38 (2001).
- 16) Horn, J., Nafpliotis, N. and Goldberg, D.E.: A niched pareto genetic algorithm for multiobjective optimization, *Proc. 1st IEEE Int. Conf. on Evolutionary Computation*, pp.82–87 (1994).
- 17) Lewis, J., Hart, E. and Ritchie, G.: A comparison of dominance mechanisms and simple mutation on non-stationary problems, *Parallel Problem Solving from Nature 1498*, pp.139–148, Springer (1998).
- 18) Mori, N., Kita, H. and Nishikawa, Y.: Adaptation to a changing environments by means of the thermodynamical genetic algorithm, *Parallel Problem Solving from Nature 1141*, pp.513–522, Springer (1996).
- 19) Sarma J. and De Jong, K.: The behavior of spatially distributed evolutionary algorithms in non-stationary environments, *1999 Genetic and Evolutionary Computation Conference GECCO*, pp.572–578 (1999).
- 20) Ursem, R.K.: Multimodal gas: Multimodal optimization techniques in dynamic environments, *2000 Genetic and Evolutionary Computation Conference GECCO*, pp.19–26 (2000).
- 21) Vavak, F., Fogarty, T.C. and Jukes, K.: A genetic algorithm with variable range of local search for tracking changing environments, *Parallel Problem Solving from Nature*, pp.376–385, Springer (1996).
- 22) Yamasaki, K.: Dynamic pareto optimum ga against the changing environments, *2001 Genetic and Evolutionary Computation Conference Workshop Program GECCO*, pp.47–55 (2001).
- 23) Yamasaki, K.: The problems under the dynamic environments need the consideration of the evaluation standard between the environments that are changing, *Proc. 6th International Symposium on Artificial life and Robotics*, pp.577–580 (2001).
- 24) Yamasaki, K., Kitakaze, K. and Sekiguchi, M.: Dynamic optimization by evolutionary algorithms applied to financial time series, *Proc. Congress on Evolutionary Computation CEC preprint*, pp.2017–2022 (2002).

(平成 14 年 7 月 10 日受付)

(平成 15 年 1 月 7 日採録)



山崎 和子(正会員)

1982年九州大学大学院理学研究科物理学専攻博士課程修了。理学博士。1988年ガウス株式会社代表取締役。1994年東京情報大学経営情報学部情報学科講師。現在同大学総合情報学部環境情報学科助教授。進化学習システム、経済物理等の研究に従事。人工知能学会、ソフトウェア科学会、日本ファイナンス学会、日本金融証券計量工学会、ISGEC各会員。



北風 和久(正会員)

1956年生。1979年大阪大学理学部化学科卒業。1985年神戸大学大学院自然科学研究科物質科学専攻博士課程修了。1988年東京情報大学電算センター助手。現在同大学総合情報学部情報システム学科助教授。進化型計算等の研究に従事。学術博士。電子情報通信学会、日本物理学会、IEEE各会員。



関口 益照

1939生。1964年東京大学経済学部経済学科卒業。同年富士通信機製造株式会社(現富士通)入社。データ通信技術部、システム本部を経て、1986年富士通システム総研(現富士通総研)取締役研究開発部門長。1995年東京情報大学経営情報学部教授。1998年同大学院経営情報学研究科教授。システム設計論、経営情報システム論の研究、教育に従事。経営情報学会、日本社会情報学会、日本ファイナンス学会各会員。