

数学的保証をもった遺伝的アルゴリズムの構築とその応用

G03MK015 澤谷 智

1. はじめに

遺伝的アルゴリズム[1](Genetic Algorithm 以下 GA)は、交叉、突然変異、自然淘汰といった生物の進化の過程を模倣したもので、複数の個体を使用し並列に探索を行う。特に交叉は GA 独特のもので、遺伝子のコーディング方法が良ければ探索においてかなりの効果を発揮する。このように GA は、最適化問題において大域的な準最適解を探索する手法として優れている。しかし、GA における一連の動作はあいまいな部分が多く、焼きなまし法[2]のような明確な最適解到達の保証はない。

本研究の目的は、数学的に最適解到達を保証した遺伝的アルゴリズム[3][4]を構築することである。

2. 遺伝的アルゴリズム(Genetic Algorithm)

遺伝的アルゴリズムは、生物の進化の過程を模倣したもので、環境に適応したものが栄え適応できないものは滅びていくという自然淘汰の原理を利用している。具体的な流れは図 1 のようになる。

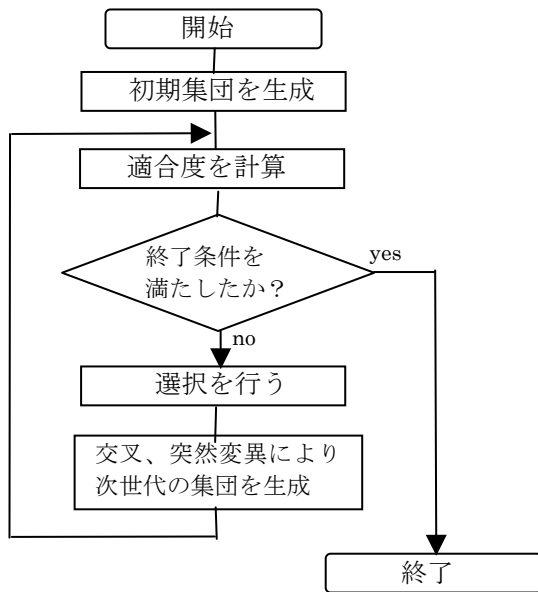


図 1. 遺伝的アルゴリズムの流れ

遺伝的アルゴリズムにおいては、探索の中心となる操作は交叉である。部分的に良い個体を組み合わせることによりさらに良い個体を生成する遺伝的アルゴリズム特有の概念である。しかし、交叉のみ行っていたのでは集団の多様性はすぐに失われてしまう。そこで、突然変異というランダムな変化を与えることで集団の多様性を保つようにしている。そして、選択を行うことで悪いものは淘汰され、良いものが増えていく。

図 1 のループの部分を実行を繰り返すにつれて多様性を保ちつつ集団全体が良いものになる。これが遺伝的アルゴリズムの仕組みだが、これら一連の流れは、あいま

いな側面が多い。つまり、数学的な裏付けがないため、最適解が必ず出現することは保証できないのである。

3. 数学的保証をもった遺伝的アルゴリズム

マルコフ連鎖[5]に基づいて動作させることにより、最適解到達を保証した GA を提案する。

3.1 マルコフ連鎖

マルコフ過程は、将来の出現確率が過去の履歴に左右されない確率過程である。離散的な時点しか考えない場合、マルコフ過程をとくにマルコフ連鎖という。さらに状態数が有限である場合、有限マルコフ連鎖という。今、

n : 全状態数, p_{ij} : $i \rightarrow j$ への推移確率
とすると推移確率行列 P は以下ようになる。

$$P = \begin{pmatrix} P_{11} & \cdots & P_{1n} \\ \vdots & \ddots & \vdots \\ P_{n1} & \cdots & P_{nn} \end{pmatrix} \quad p_{ij} \geq 0, \quad \sum_{j=1}^n p_{ij} = 1$$

P が時点によらず一定の場合、定常という。

$\vec{f}^{(a)}$: 全状態の存在確率分布 ${}^t \vec{f}^{(a)} = ((f^{(a)})_1, \dots, (f^{(a)})_n)$

$(f^{(a)})_i$: 時点 a での状態 i の存在確率

$$(f^{(a)})_i \geq 0 \quad \sum_{i=1}^n (f^{(a)})_i = 1$$

とすると、時点 a の次の時点 $a+1$ の確率分布は、

$${}^t \vec{f}^{(a+1)} = {}^t \vec{f}^{(a)} P$$

となる。よって初期分布から N 時点後の確率分布は、

$${}^t \vec{f}^{(N)} = {}^t \vec{f}^{(0)} P P \cdots P = {}^t \vec{f}^{(0)} P^N \quad (1)$$

となる。定常マルコフ連鎖において、すべての状態がエルゴード的(再帰的、非ゼロ、非周期的)である場合

$$\lim_{N \rightarrow \infty} {}^t \vec{f}^{(0)} P^N = \vec{u} \quad (2)$$

となる \vec{u} が存在する。これを定常分布という。

この、 P をかけ続けると定常分布に収束するということを利用することにする。

3.2 マルコフ連鎖に基づく GA (突然変異のみ)

ここでは、突然変異のみを用いた GA を考える。GA は、現在の集団が、その 1 世代まえの集団から生成され、かつ子供の生成に突然変異のみを用いる場合、1 つの親個体の遺伝子にランダムな変異を起こすことで 1 つの子供が生成されるという 1 対 1 の推移が行われるため、単純マルコフ連鎖とみなすことができる。つ

まり、

$f^{(i)}$: i 世代目の個体の存在確率分布

P : 親から子への推移確率行列

N : 世代数

とすると、マルコフ連鎖による GA のモデルは(1)式のように記述することができる。つまり、世代を十分に重ねることで、各状態の存在確率分布が定常分布に達する。このとき、最適解となる状態が最も高い存在確率となるようにしたい。そこで、親個体 i から子個体 j への状態推移の推移確率に(3)式のようなシグモイド関数を用いることにする。

$$P_{i \rightarrow j} = \frac{1}{1 + e^{(E_i - E_j)/T}} \quad (3)$$

E_i : 個体 i の適合度 T : 温度パラメータ

こうすることで、定常分布は式(4)のようなボルツマン分布となる。

$$f_i = C e^{E_i/T} \quad (C: \text{定数}) \quad (4)$$

つまり、1 番適合度の高い個体が 1 番高い存在確率を持つようになる。よって、世代を重ねれば、各個体の存在確率はボルツマン分布となるので最も適合度の高い個体が最も高い存在確率をもつことを数学的に保証できる。ただし、突然変異のみを用いるより、交叉も用いた方が探索は速くなるので交叉を導入することを考える。

3.3 交叉の導入

マルコフ連鎖における状態推移は、1 つの状態から 1 つの状態への推移への推移を前提としている。つまり、交叉のような 2 つの状態から 2 つの状態が生まれるというものはマルコフ連鎖では表現できない。交叉における状態推移を 1 対 1 とするための考えられる方法の 1 つに状態 2 つを連結して 1 つの状態とすることを考える。つまり、交叉、突然変異による状態推移は、図 2 のように行う。

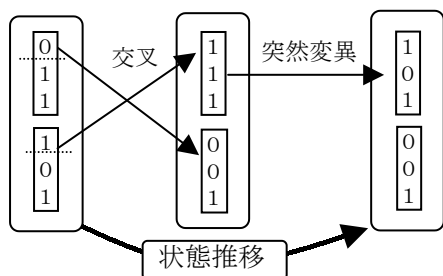


図 2. 交叉、突然変異による状態推移

交叉は、自己の状態内にある 2 つの個体で行う。こうすることで、交叉、突然変異による状態推移は、自己の変化のみによる状態推移となるため、定常マルコフ連鎖として実行でき、世代を重ねることで定常分布に達することが数学的に保証される。しかし、状態が大きくなったため、探索空間が莫大に大きくなり、さらに、1 つの状態の適合度は、その中にある 2 つの状態の適合度の和となるため、解の平均化がおき効率の良い探索を行うことができないという欠点がある。

そこで、まず図 3 のように交叉と突然変異を別々に

考えることにする。

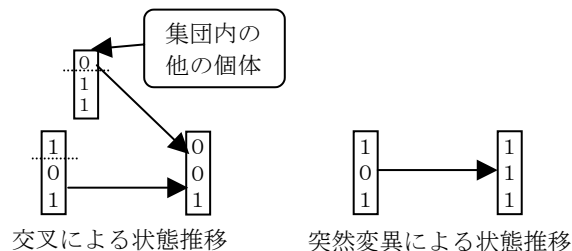


図 3. 交叉、突然変異それぞれによる状態推移

交叉による状態推移は、交叉相手を集団の中から、各個体の存在確率に従い選択し、交叉を行い状態を生成する。つまり、通常の GA に近い形となっている。

交叉による状態推移行列を P_c 、突然変異による状態推移を P_m とする。そして、(5)式のように基本的には突然変異のみを実行し、数世代おきに交叉を実行するようにする。

$$f^{(N)} = f^{(0)} P_c P_m \cdots P_m P_c P_m \cdots P_m \quad (5)$$

交叉による状態推移の推移確率は、相手の存在確率が世代ごとに異なるので定常ではなく、さらに交叉のみでは全状態に到達することはできないので既約でもない。つまり、交叉を行えば定常分布を崩してしまう。しかし、その後、突然変異を複数回行うことで定常分布に近づけるようにしている。よって、この方法を用いれば、数学的に定常分布に到達することは証明できないが、実際に実行した場合定常分布に近い分布を得ることができる。

ここで、状態推移に突然変異のみ用いた場合と、交叉と突然変異を用いた場合にどれくらいの効果が得られるか実験を行った。適用する問題は、Tight 問題を用いた。

Tight 問題とは、表 1 で定義されるだまし問題をいくつか連結したものである。

表 1. Goldberg's order-3 deceptive problem

x	000	001	010	011	100	101	110	111
f(x)	28	26	22	0	14	0	0	30

表 1 は、Goldberg's order-3 deceptive problem というだまし問題である。これは、0 の数が多くなるほど評価が良くなっていくが、最適解は"111"であるというだまし問題になっている。

実験には 6bit Tight 問題を用いた。交叉は 10 世代おきに実行、温度パラメータ T は 0.5 とし 3000 世代実行し、ボルツマン分布への収束の違いを見た。

結果は図 4 のようになり、交叉を取り入れた方がボルツマン分布への収束が速くなっていることがわかる。このことより、交叉が効果的に働いていることがわかる。よって、この手法を用いれば、突然変異のみ行うより効果的に探索を行うことができる。

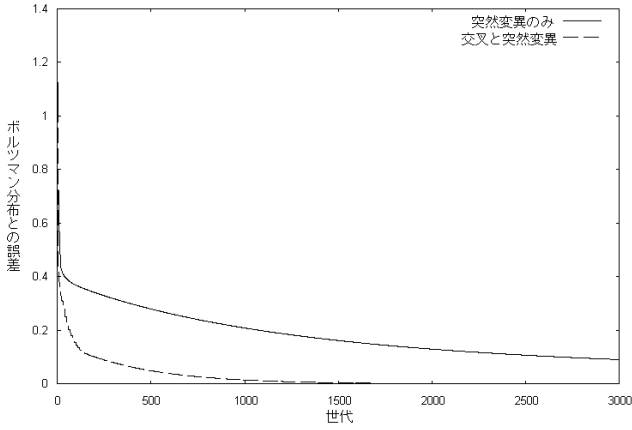


図4. 突然変異のみを用いた場合と、交叉と突然変異を用いた場合によるボルツマン分布への収束の違いを示している。横軸は世代で、縦軸は、横軸の世代における確率分布において、各状態ごとに理想的なボルツマン分布との差をとり、その絶対値の和をとったものである。

しかし、マルコフ連鎖の計算は、常に全状態を保持しながら定常分布を求める計算を行っていて、莫大なメモリと膨大な時間を要する。よってこのままではサイズの大きな問題には適用できない。そこで、不要な状態を排除する選択を行う必要がある。ここで重要なことは、確率分布がボルツマン分布となるという数学的保証を崩さないような選択を用いなければならないということである。

3.4 新選択法

大規模な問題にも適用できるように、数学的保証を崩さない選択法を導入する。そのために、以下に定義する確率変数 η を用いる。

• On-Off Probability Function[6]

下式のような確率関数を定義する。

$$P(\eta) = \frac{1}{a} \delta(\eta - a) + \left(1 - \frac{1}{a}\right) \delta(\eta) \quad (a : 1 \text{ 以上の定数})$$

変数 η は $a(\text{on})$ か $0(\text{off})$ の2つの値を、それぞれ確率 $1/a, 1-1/a$ でとる。

$$\langle\langle \eta \rangle\rangle = \int_0^\infty P(\eta) d\eta = 1 \quad \langle\langle \cdot \rangle\rangle : \text{統計平均}$$

確率変数 η の特徴は、統計平均が1となることである。

この確率変数 η を用いて以下のような行列 M を定義する。

$$M = \begin{bmatrix} \eta_1 & 0 & \cdots & 0 \\ 0 & \eta_2 & \cdots & 0 \\ \vdots & \cdots & \ddots & \vdots \\ 0 & 0 & \cdots & \eta_n \end{bmatrix} \quad \eta_i = \begin{cases} a_i & P_{a_i} = 1/a_i \\ 0 & P_0 = 1-1/a_i \end{cases} \quad \frac{1}{a_i} = \min\left\{1, \frac{f_i}{\varepsilon}\right\} \quad (\varepsilon : \text{定数})$$

この行列 M を用いて、次式のように不要な状態(個

体)を落とす選択を行う。

$${}^t \vec{f}^{(N)} = {}^t \vec{f}^{(0)} MPMP \cdots MP$$

(M は random matrix であるので世代ごとで異なる) ここで、 $f^{(N)}$ はボルツマン分布とはならないが、統計平均を取ることにより、

$$\langle\langle {}^t \vec{f}^{(N)} \rangle\rangle = {}^t \vec{f}^{(0)} \langle\langle M \rangle\rangle P \langle\langle M \rangle\rangle P \cdots \langle\langle M \rangle\rangle P$$

$\langle\langle M \rangle\rangle = E$ より、

$$\langle\langle {}^t \vec{f}^{(N)} \rangle\rangle = {}^t \vec{f}^{(0)} EPEP \cdots EP = {}^t \vec{f}^{(0)} PP \cdots P$$

となり、選択を行わない場合と同様にボルツマン分布となる。よって、この方法を用いれば、最適解到達の保証を得られる。さらに、良い個体は必ず出現するが、悪い個体もその存在確率に従い出現することが保証されるため、多様性を実現している。

3.5 子供の生成数削減

3.4章で提案した選択法を用いれば、保持する状態の数を減らすことができるが、マルコフ連鎖に基づいて実行するため、大規模な問題では1つの状態から生成される子供の数が多すぎて、未だ実行に多大な時間が必要となってしまう。そこで、生成する子供の数を減らすことを考える。今、状態 i から生成可能な状態の数 n 個であったとすると、状態 i から推移先の候補の中から状態 j が選択される確率は式(6)のようになる。

$$P(\text{select}(j) / X = i) = 1/n \quad (6)$$

ここで、子供をランダムに1つ生成するように変更を加える。つまり、式(6)が式(7)のように変更される。

$$P(\text{select}(j) / X = i) = \begin{cases} 1 & P(1) = 1/n \\ 0 & P(0) = 1-1/n \end{cases} \quad (7)$$

ここで注目すべき点は、式(7)の統計平均は式(6)となることである。つまり、子供をランダムに1つ生成するように変更した場合の推移確率行列を P_{rand} とした場合、

$$P = \langle\langle P_{rand} \rangle\rangle$$

が成立すると考えられる。つまり、生成する子供の数を減らしても、平均的にボルツマン分布が得られると考えられる。

ここで、このような考え方を用いた場合に得られる分布の統計平均がどのようになるか実験を行った。具体的には以下の式のように実行し、得られた分布の統計平均をとる。

$${}^t \vec{f}_{rand}^{(N)} = {}^t \vec{f}^{(0)} MP_{c,rand} MP_{m,rand} \cdots MP_{m,rand} MP_{c,rand} MP_{m,rand} \cdots MP_{m,rand}$$

問題には 6bit Tight 問題を用い、 $\varepsilon=0.05$ 、交叉間隔は 20、温度パラメータ T は 10 とし、300 回実行し、その平均をとった。

結果は図5のようになり、ほぼボルツマン分布のような分布を得ることができた。以上のことより、生成する子供の数を減らしても、数学的保証は保たれると考えられる。

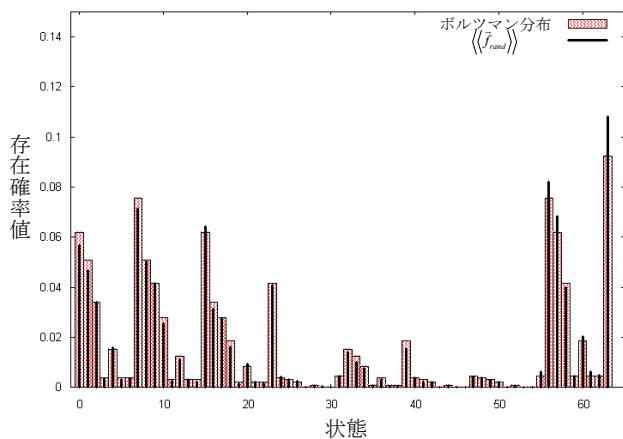


図 5. 実験により獲得した分布の平均と、理想的なボルツマン分布をプロットしたグラフである。縦軸は状態の存在確率値で、横軸の値は状態を表すビット列を 10 進数に直したものである。

4. 実験

ここで、今までに説明した数学的保証付き GA と、通常の GA を能力を比較する実験を行った。実験の条件は表 2 のとおりである。

表 2. 実験の条件

通常の GA	数学的保証付き GA
個体数…100	$T = 5$
交叉確率…0.8	$\epsilon = 0.01$
突然変異確率…0.01	交叉間隔は 10 世代
選択…トーナメント選択	
共通の条件	
48bit tight 問題	
試行回数…20 回	
1 回につき 60 秒実行	
Celeron1.7GHz のマシンを使用	

数学的保証付き GA については、突然変異のみの場合と、交叉と突然変異を用いた場合の 2 種類について実験を行った。

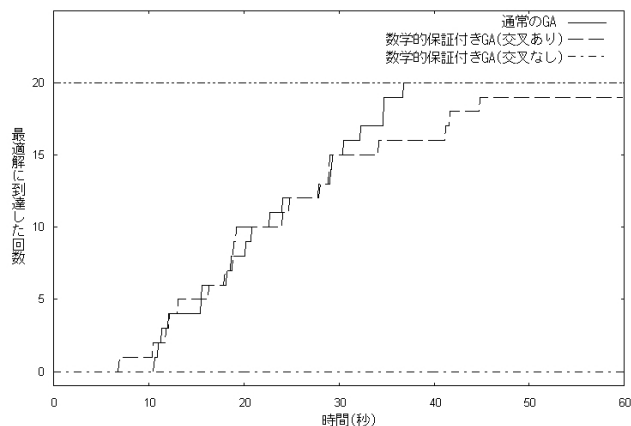


図 6. 表 2 の条件に基づき実験を行った結果である。横軸は世代で、縦軸は 20 回の試行中、横軸の世代までに最適解を獲得できた回数を示している。

結果は図 6 のようになり、通常の GA と数学的保証

付き GA (交叉あり) がほぼ同等の性能を持っていることがわかる。突然変異のみを用いた場合は、時間内に最適解を出すことができなかった。これは、交叉のような他の状態の情報を利用することができないため、全てのトラップを自分ひとりで乗り切らなければならないので、このような結果になったと考えられる。

5. 突然変異による状態推移方法の変更

これまでの方法は、突然変異による状態推移は、1 bit 突然変異を一回のみおこない状態を生成していたが、突然変異を複数回行って状態を生成することを考える。図 7 は、最適な状態へのパスを表したものである。

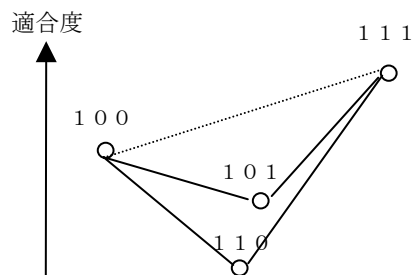


図 7. だまし問題の 1 部分

今までの方法では、最適な状態へたどり着くには必ず 1 度悪い状態を通過しなければならず、たどり着くのが困難であった。ここでもし、2bit 同時に突然変異が起これば容易に最適な状態へたどり着くことが可能である。そこで、突然変異を以下のように変更する。

$$P_{m,rand} \rightarrow \begin{cases} P_{m,k,rand} \\ P_{m,1,rand} \end{cases} \quad \text{交互に実行}$$

$P_{m,k,rand}$: k 回 1bit 突然変異を実行

なぜ交互に実行するかというと、マルコフ連鎖が周期的になるのを防ぐためである。

この方法を用いることで、どの程度能力が改善されるか、先ほどの 48bit Tight 問題を用いた実験を行った。ここでは、突然変異の回数 k を 2 とした。

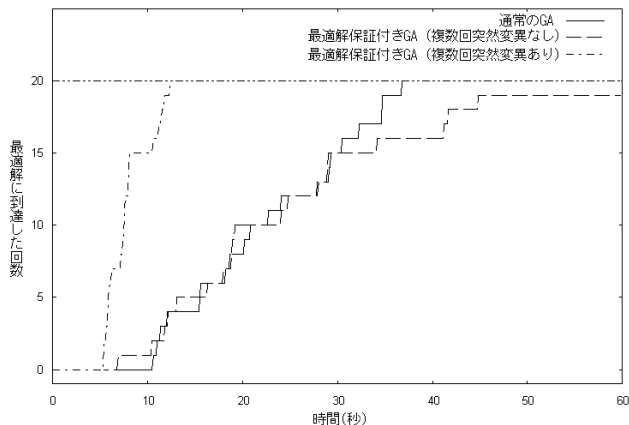


図 8. 実験の結果である。横軸は世代で、縦軸は 20 回の試行中、横軸の世代までに最適解を獲得できた回数を示している。

結果は図 8 のようになり、圧倒的に複数回突然変異

を用いた方が良い結果となった。

しかし、Tight 問題はあくまで人為的に設定した問題であるため、これでよい結果が出たからといって、一般的に良い結果が出るとはいえない。そこで、より一般的な問題に対し、適用することを考える。

6. 巡回セールスマン問題への応用

より一般的な問題として、代表的な組み合わせ最適化問題である巡回セールスマン問題(TSP)を用いることにする。TSP は、地図上に配置された全ての街を、ちょうど 1 度ずつ経由してもとに戻る閉路のうち、長さが最小のものを求める問題である。

TSP に GA を適用する場合、従来の交叉では制約を満たさない遺伝子(致死遺伝子)が生成されてしまう。これまでに制約を満たす子供を生成できるさまざまな交叉法が考案されている。ここでは、以下のような交叉法を取り上げる。

- Cycle Crossover
- Edge Recombination
- Edge Assembly Crossover

Cycle Crossover は、TSP の制約を満たすことのみを考えた交叉法である。しかし、TSP においては、親のエッジを継承させることが重要である。この交叉法はエッジを継承させることについては考慮させていないため、良いパフォーマンスは得られない。Edge Recombination[7]、Edge Assembly Crossover[8]は、親のエッジを継承させることを考慮した交叉法であるため、良いパフォーマンスを持っている。

6.1 既存の交叉法を用いた数学的保証付き GA

Edge Recombination, Edge Assembly Crossover では 1 つの親から複数個の子供を生成し、その中に親の個体 2 つを加えた中から優秀な個体 2 つを子供としているので、数学的保証付き GA で用いる場合も、交叉による状態推移は、交叉により生成される優秀な状態 2 つに状態推移を起こすようにする。Cycle crossover は親のペアから 2 つの個体しか生成できないため、生成した 2 つの状態に状態推移をおこす。

ここで、用いる交叉法によりボルツマン分布への収束がどのように違うのかをみる実験を行った。問題はテスト用の 6 都市問題を用い、選択は使用せず、各交叉法につき 100 世代実行した。そして温度パラメータ T は 0.3 とし、交叉は 10 世代ごとに実行した。

結果は図 9 のようになり、有効な交叉を用いた場合は突然変異のみを用いる場合よりボルツマン分布への収束が速くなっていることがわかる。そして、定常分布に近づいてから交叉を実行すると、分布を崩してしまう。しかし、その後突然変異を行うことでまた定常分布へ近づいていくということを繰り返している。

よって、有効な交叉法を取り入れればパフォーマンスが良くなることがわかる。

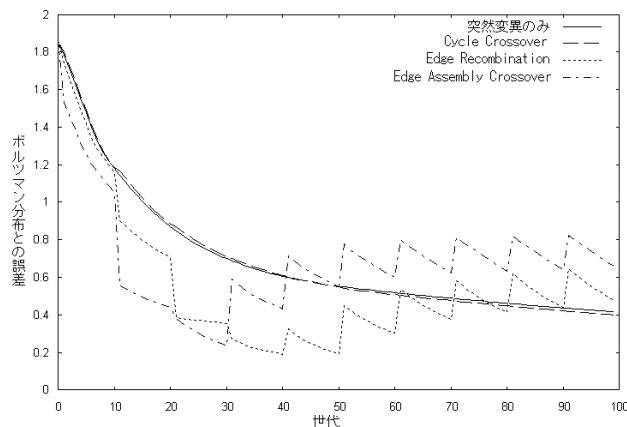


図 9. 用いる交叉法の違いによるボルツマン分布への収束の違いを示したものである。横軸は世代で、縦軸は、横軸の世代における確率分布において、各状態ごとに理想的なボルツマン分布との差をとり、その絶対値の和をとったものである。

6.2 実験

巡回セールスマン問題をもちいて従来の GA と数学的保証付き GA の比較を行った。もちいた交叉法は一番性能のよい Edge Assembly Crossover を用いた。問題は、TSPLIB[9]で公開されている以下のようなベンチマーク問題を用いた。

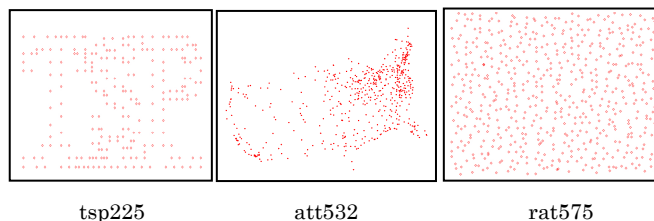


図 10. tsp225 は 225 都市問題であり、都市の配置が「TSP」という文字を形成している。att532 は 532 都市問題であり、アメリカの国土上に都市を疎密があるように配置されている問題である。rat575 は 575 都市問題であり、都市はランダムに配置されている問題である。

実験には、表 3,4,5 のようなパラメータを用いた。

表 3. 通常の GA

	tsp225	att532	rat575
個体数	200	300	300

表 4. 数学的保証付き GA

	tsp225	att532	rat575
ϵ	0.005	0.00333	0.00333
温度 T	30	20	20
突然変異回数	3	3	3
交叉間隔	20	30	30

表 5. 共通の条件

	tsp225	att532	rat575
EAX における子供の生成数	10	50	50
実行時間(s)	300	10800	10800
試行回数	20	10	10

EAX: Edge Assembly Crossover

実験には Pentium4 3GHz のマシンを用いた。結果は以下の表のようになった。

表 6. tsp225 に対する結果

	通常の GA	数学的保証付き GA
最良	3916(0)	3916(0)
平均	3917.45(0.037)	3917.75(0.045)
最適解獲得回数	12	18

括弧の中の値は、最適解 3916 に対する誤差(%)である

表 7. att532 に対する結果

	通常の GA	数学的保証付き GA
最良	27696(0.036)	27686(0)
平均	27708.4(0.081)	27705.1(0.069)
最適解獲得回数	0	1

括弧の中の値は、最適解 27686 に対する誤差(%)である

表 8. rat575 に対する結果

	通常の GA	数学的保証付き GA
最良	6775(0.03)	6773(0)
平均	6781.5(0.12)	6775.5(0.066)
最適解獲得回数	0	1

括弧の中の値は、最適解 6773 に対する誤差(%)である

表 6,7,8 の結果より、通常の GA より数学的保証付き GA の方が良い解を出すことができ、また、平均的に良い解を出すことができた。とくに 500 都市クラスの問題においては、通常の GA では最適解を出すことができなかったが、数学的保証付き GA では、一回だけではあるが、最適解をだすことができた。

図 11,12,13 より、通常の GA より数学的保証付き GA のほうが少し時間がかかっていることがわかる。

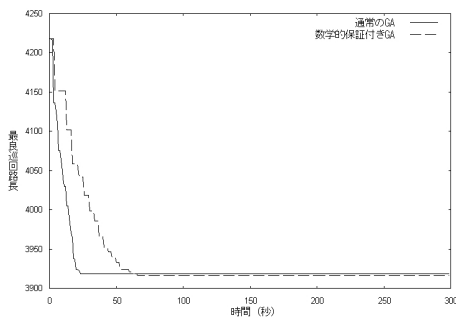


図 11. tsp225 に対する実行の 1 例

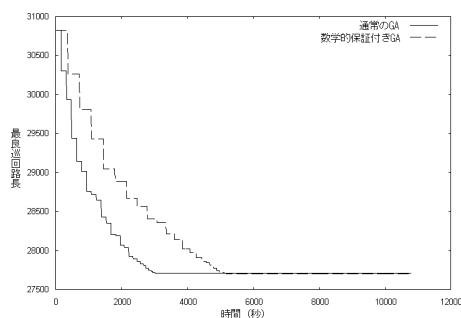


図 12. att532 に対する実行の 1 例

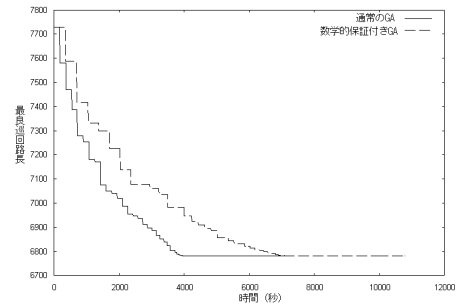


図 13. rat575 に対する実行の 1 例

7. 結論

今回提案した数学的保証付き遺伝的アルゴリズムは、マルコフ連鎖に基づき動作させ、推移確率をボルツマン分布とすることで最適解到達の保証をしている。大規模な問題にも適用できるように、数学的保証を崩さないような選択を導入した。この選択法を用いることにより、良い状態だけではなく、悪い状態もその存在確率に従い出現することが保証され、多様性を実現している。このように数学的保証付き GA を用いることで、TSP のような一般的な問題においても、通常の GA より良い結果を残すことができると考えられる。

交叉の役割は、良い状態を早期に生成することであるため、交叉以外に良い状態を生成できる方法があるなら、それを交叉の代わりに取り入れて実行することも可能であると考えられるため、今後さまざまな応用が可能であると考えられる。

参考文献

- [1] 伊庭 斉志, 遺伝的アルゴリズムの基礎, オーム社, 1994
- [2] Emile Aarts and Jan Korst, Simulated Annealing and Boltzmann Machines, pp 13-31, John Wiley & Sons Inc, 1988
- [3] 平早 哲明, 澤谷 智, 宗久 保子, 宗久 知男, 遺伝的アルゴリズムにおける新しい選択法 -理論-, pp45-50, 電子情報通信学会技術研究報告, June, 2004
- [4] 澤谷 智, 平早 哲明, 宗久 保子, 宗久 知男, 遺伝的アルゴリズムにおける新選択法(2) -DeJong の標準関数を用いた実験-, pp51-56, 電子情報通信学会技術研究報告, June, 2004
- [5] 西田 俊夫, 応用確率論, pp61-129, 培風館, 1973
- [6] Tomo MUNEHISA and Yasuko MUNEHISA, "A new Approach to Stochastic State Selection in Quantum Spin Systems", pp.2759-2760, Journal of the Physical Society of Japan, November, 2003
- [7] P.Whitley T.Starhweath D.Fugurary, Scheduling Problems and Traveling Salesman : The Genetic Recombination Operator, Proceeding of the 3th International Conference on Genetic Algorithms, pp.133-139, 1989
- [8] 永田 雄一 小林 重信, 巡回セールスマン問題に対する交叉: 枝組み立て交叉の提案と評価, 人工知能学会誌, vol14, no5, 1999
- [9] Gerhard Reinelt, TSPLIB, <http://www.iwr.uni-heidelberg.de/iwr/comopt/software/TSPLIB95>