

# 琉球大学学術リポジトリ

## 2集団の共生関係に基づく共進化アルゴリズムの性能解析

メタデータ	言語: 出版者: 琉球大学工学部 公開日: 2010-01-13 キーワード (Ja): キーワード (En): Co-evolutionary algorithm, Symbiosis, Genetic algorithms, N-Queens problem 作成者: 根路銘, もえ子, 遠藤, 聡志, 山田, 孝治, 宮城, 隼夫 メールアドレス: 所属:
URL	<a href="http://hdl.handle.net/20.500.12000/14707">http://hdl.handle.net/20.500.12000/14707</a>

## 2集団の共生関係に基づく共進化アルゴリズムの性能解析

根路 銘もえ子\* 遠藤 聡志\*\* 山田 孝治\*\* 宮城 隼夫\*\*

### Analysis for Performance of Symbiosis Co-evolutionary Algorithm

Moeko NEROME\* Satoshi ENDO\*\* Koji YAMADA\*\* Hayao MIYAGI\*\*

#### Abstract

In this paper, we analyze the behavior of symbiotic evolution algorithm for the  $N$ -Queens problem as benchmark problem for search methods in the field of artificial intelligence. It is shown that this algorithm improves the ability of evolutionary search method. When the problem is solved by Genetic Algorithms (GAs), an ordinal representation is often used as one of gene conversion methods which convert from phenotype to genotype and reconvert. The representation can hinder occurrence of lethal genes. Typically, the representation pattern is fixed to one pattern. However, we consider that the kinds of generated solution are increased by preparing the permutation pattern with the several and the better solutions may be generated by the permutation pattern evolving. This paper introduces the symbiotic evolution model in which two evolutionary populations are solutions and permutations for solving the  $N$ -Queens problem. To investigate the performance of symbiotic algorithm, it is compared with three kinds of evolutionary methods based on GA. From the results of computer simulations, we consider the characteristics of this algorithm.

**Key Words:** Co-evolutionary algorithm, Symbiosis, Genetic algorithms,  $N$ -Queens problem.

#### 1. まえがき

競合共進化アルゴリズムは、競合関係を形成する種間の相対評価に基づき適応的に解を獲得するため、遺伝的アルゴリズム (Genetic Algorithms: GAs) [1] のような一般的な進化的探索手法に比べ、良い解を獲得すると期待されている。特に、fitness landscape が明示的に決定できない問題において有効解を獲得でき、お互いに影響を及ぼし合う集団間の進化を促進する特徴を持つ。したがって、各個体に絶対評価を与えることが難しい問題に対して適用され、その有効性が示されている [2], [3], [4], [5]。我々は、これまでにゲーム戦略獲得問題における適応型探索手法としての競合共進化アルゴリズムに着目し、2人零和ゲームの一種である Tic-Tac-Toe ゲームにおいて、有効な戦略の探索に優れていることを示した [6]。

自然界において、生物種間の関係は競合だけではなく他の種間関係も存在する [7]。その一例が共生である。Paredis は、共生関係を用いた共生共進化アルゴリズムを提案した [8]。Paredis の研究により、共生共進化アルゴリズムは、1つの集団に GA を適用する手法よりも、epistasis の高い問題を扱える事が示されている [8], [9]。しかしながら、

共生共進化が解探索に有効な理由が明確に述べられておらず、集団を2つに分けて進化させる利点が明らかではない。

本稿ではこれらの不明確な点に着目し、共生共進化アルゴリズムによる解の進化過程を解析する。ここでは、人工知能の分野においてベンチマーク問題として知られている  $N$ -クイーン問題を扱う [10]。この問題を GA によって解く場合、個体の遺伝子型の設計方法として、しばしば、致死遺伝子を生成しない順序表現へと変換する手法が用いられる [11]。一般に、変換には、固定された順序表現パターンを用いる。しかしながら、いくつかの順序表現パターンを用意しておくことによって、生成される解の種類が増加する。また、順序表現パターンを進化させる事によって、より良い解が生成される可能性が高いと考えられる。本稿では、まず、 $N$ -クイーン問題を解くために、解と順序表現パターンの2集団を進化させる共生進化アルゴリズムを説明する。さらに、アルゴリズムの性能を検証するために、3種類の GA による進化方法との比較を行う。また、計算機シミュレーションの結果から、共生共進化アルゴリズムの特徴を考察する。

#### 2. 共生共進化

##### 2.1 共進化の概念

共進化とは、明らかに生態学的関係をもつ2つ以上の生物種の相互依存進化である [12]。共進化の進化過程における生物種間の関係には、競争・共生・寄生等がある。共進化する種 (集団) の進化過程を以下に示す (Fig. 1)。

受理: 2000年6月5日

\*大学院理工学研究科 総合知能工学専攻

(Graduate Student, Doctoral Course in Complex Intelligent Systems Engineering, Graduate School of Science and Engineering)

\*\*工学部情報工学科

(Department of Information Engineering, Faculty of Engineering)

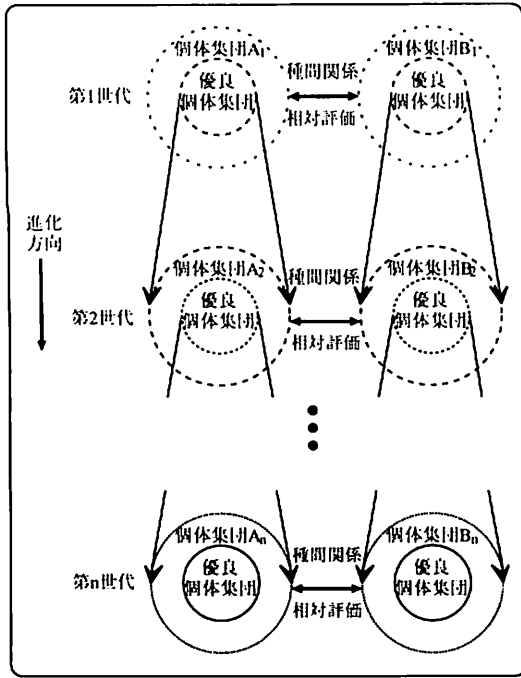


Fig. 1. 共進化の概念

Fig. 1 は、2 集団間の共進化を示している。ある集団内の 1 個体は、同種もしくは異種の他の集団との種間関係に基づき、相対的な評価が与えられる。集団内における優良個体集団が次世代へ存続する。

競争は、種間関係の 1 つであるだけでなく、進化メカニズムの 1 種でもある。競争に基づく共進化は、ある生物が他の生物に対して優位に立とうとする結果、相互作用に関連している性質が互いに進化する現象 [13] としてよく知られている。競合 (predator-prey) 共進化アルゴリズムは、このメカニズムを適応的探索手法としてモデル化されており、その解探索の有効性が示されている [2], [3], [4], [5], [6]。競合関係にある集団同士において、一方の集団の成功は、他方の集団の失敗を意味している。したがって、競合共進化アルゴリズムにおける適応度は、2 個体間の競合において優位であった個体の適応度が高くなるように設定される。

一方、共生は、異なる生物が密接な関わりを持ち一緒に生活する状態を指す [13]。一般的には、共進化の進化過程における種間関係の 1 種と定義されている。しかしながら、Paredis は、共生を進化メカニズムの 1 種と捉え、共生共進化アルゴリズムを提案し、解探索における有効性を示している [8]。共生関係にある集団同士において、一方の集団の成功は、他方の集団の存続の機会を高くすることを意味している。したがって、共生共進化アルゴリズムにおける適応度は、2 個体間の競合において一方の適応度が他方の適応度に反映されるように設定される。

### 2.2 共生共進化アルゴリズム

Paredis によって提案された共生アルゴリズムにおいて、一方は問題を解く解集団、もう一方は解を生成するときに使われる順列表現パターン (オーダー) として設定される [8], [9]。共生アルゴリズムを Fig. 2 に示す。

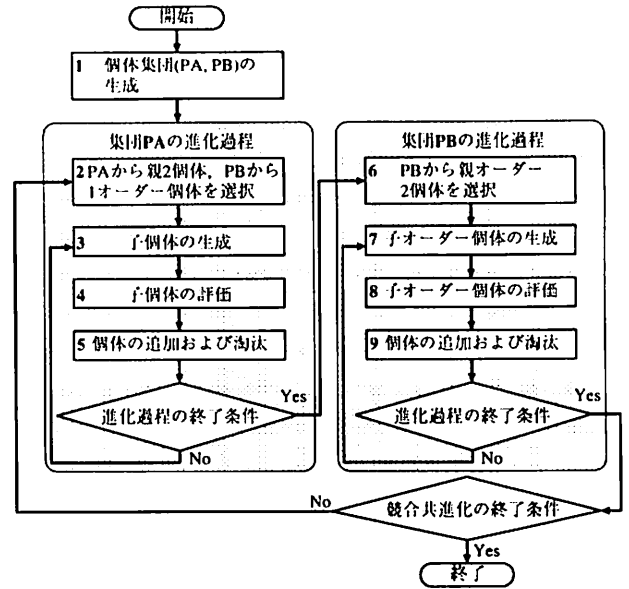


Fig. 2. 共生共進化アルゴリズム

アルゴリズムは、集団 PA の進化過程と集団 PB の進化過程を交互に実行する。

- step1: 初期個体集団 PA(解), PB(オーダー) の生成。
- step2: 親個体とオーダー個体の選択。  
集団 PA から親となる 2 個体を選択し、集団 PB から 1 個体選択する。
- step3: 子個体の生成。  
親 2 個体を選択されたオーダー個体と遺伝オペレータを適用し、子個体を生成する。
- step4: 子個体の評価。  
子個体の評価値と集団 PA の最小評価値を比較し、子個体が高ければ、子個体とその個体を入れ換える。そうでなければ、淘汰する。ここで、終了条件が満たされていれば、step6 の集団 PB の進化過程へと移行する。満たされていない場合は、step2 へ戻る。
- step6: 親オーダー 2 個体の選択。
- step7: 子オーダー個体の生成。  
親オーダー 2 個体に遺伝オペレータを適用し、子オーダー個体を生成する。
- step8: 子オーダー個体の評価。  
子オーダー個体の評価値と集団 PB の最小評価値を比較し、子オーダー個体が高ければ、子オーダー個体とその個体を入れ換える。そうでなければ、淘汰する。ここで、終了条件を満たしていれば、集団 PA の進化過程へと移行する。満たされていない場合は、step6 へ戻る。

以上の処理を共生共進化の終了条件を満たすまで繰り返す。

### 3. N-クイーン問題における共生共進化モデルの設計

本稿では、共生共進化アルゴリズムの特性を検証するために、解とオーダーの進化関係を実現できる問題として、N-クイーン問題を扱う。

3.1 N-クイーン問題

N-クイーン問題は、 $N \times N$  の正方形のチェス盤のマス目にチェスのクイーンのコマを  $N$  個並べ、どの2つのクイーンも同じ行・同じ列・同じ対角線上にないような局面を探す問題である。

問題サイズの設定が容易であるため、問題サイズを変更してのシミュレーションが行える。また、解空間の小さい問題は全探索ができるため、アルゴリズムの進化過程を解析することが可能である。

3.2 遺伝子コーディング

共生共進化アルゴリズムでは、N-クイーン問題における解とその解を表現型から順序表現の遺伝子型へ変換する時に使用する順序表現パターン(オーダー)を共進化させる。

3.2.1 解個体のコーディング

1個体における染色体は、チェスボードにおけるクイーンが置かれる場所を示す配列である。染色体  $S$  は、 $N$  個の変数(遺伝子)  $s_i$  で構成される。各遺伝子座はチェスボードの各列に対応しており、1遺伝子はそれらの列におけるクイーンの行を示している。つまり、Fig. 3 中の  $s_3 = 2$  はボードの2行3列にクイーンを置くことを指す。

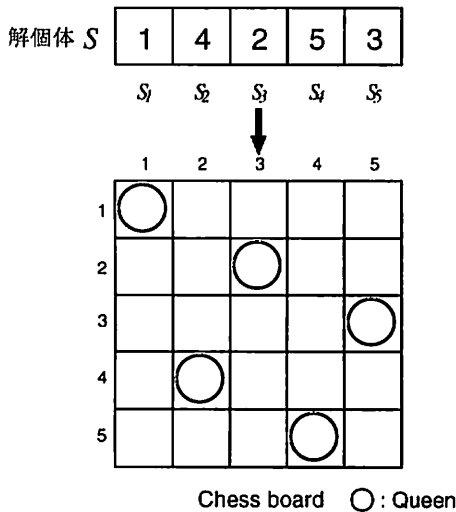


Fig. 3. 解のコーディング例

したがって、対立遺伝子は  $1 \sim N$  の値をとる。このように1個体をコーディングする場合、最適解は以下の条件を満たさなければならない。

条件1  $s_i \neq s_j$

条件2  $|s_i - s_j| \neq |i - j|$

条件1は、同じ行に2つ以上のクイーンが存在しないことを示しており、条件2は、対角線上に存在しないことを表している。また、個体の生成時に各遺伝子座が同値にならないように設計することにより、条件1を考慮せずに解探索を行うことができる。

各遺伝子座が同値にならない設計とは、個体が順序配列の状態を指す。順序配列の個体に対して1点交叉などの一般的な遺伝オペレータを適用する場合には、致死遺伝子

を生じる可能性が非常に高い。交叉などによって致死遺伝子を生じることのない遺伝子型の設計方法として、基準となるオーダーを用いて順序表現へと変換する方法がある[11]。本稿では、変換する際に使用されるオーダー個体を解と共進化させる。

3.2.2 オーダー個体のコーディング

解個体を順序表現から順序表現へと変換するためのオーダーは、解個体の各遺伝子座を順序付けたものである。つまり、オーダー個体の染色体は、解個体の各遺伝子座の順序系列であり、遺伝子はその順序を示す。したがって、対立遺伝子は、 $1 \sim N$  の値をとり、ある遺伝子の値が他の遺伝子の値と同値になることはない。

オーダーを用いた解の順序表現への変換方法を以下に示す(Fig. 4)。

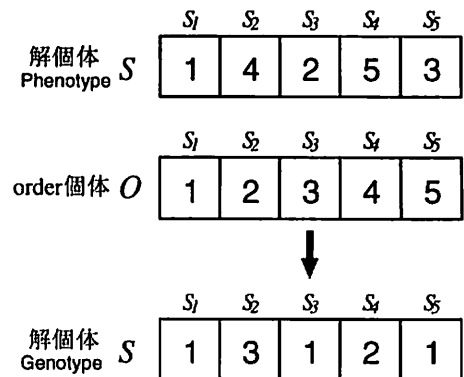


Fig. 4. 遺伝子型の変換例

解の表現型をオーダーを用いて順序表現に変換する場合、表現型の遺伝子の値がオーダーの何番目であることを示す数値に変換される。ただし、一度数えられた数値は除外して残りの数値間での順位の数値をとる。Fig. 4の例では、まず、 $s_1 = 1$  は、オーダーの1番目と一致するので、遺伝子型における値は1となる。 $s_2 = 4$  は、先程数えられた1の値をオーダーから除外すると、一致する値はオーダーの3番目であるので、遺伝子型における値は3となる。同様にして表現型を順序表現の遺伝子型に変換することができる。順序表現によって表される遺伝子は、交叉・突然変異等のオペレータによって致死遺伝子を生じない。また、遺伝子型は逆変換されることによって、表現型に戻される。

一般には、1つのオーダーを固定し、そのオーダーのみを用いて全ての解を変換する。しかしながら、我々は、いくつかのオーダーを用いることにより生成される解の種類が増え、また、オーダーを進化させることによってより良い解が獲得されると考える。

3.3 適応度

解個体の適応度関数を式(1)に示す。

$$F(S_i) = \frac{1}{1 + NQ(S_i)} \tag{1}$$

ここで、

$S_i$ :  $i$  番目の解個体

$F(S_i)$ :  $S_i$  の適応度

$NQ(S_i)$  :  $S_i$  の表す配置でクイーンが競合する数

オーダーは解のように問題から絶対的な適応度が決定される訳ではない。そこで、1 回の解の生成における適応度は、親個体と子個体の平均適応度とする。しかしながら、1 回の評価のみでオーダーの適応度を決定せず、使用される度に適応履歴を残しておき、その平均値をオーダー 1 個体の適応度とする [14]。その適応度を用いてオーダー 1 個体集団を進化させる。

### 3.4 遺伝オペレータ

解個体集団に対しては、1 点交叉および突然変異を適用する。また、オーダー個体集団に対しては、順列表現における交叉方法の 1 つである辺組換え交叉 [15] を適用する。

## 4. シミュレーション

本稿では、解とオーダーの種を分化させることによる有効性を検証するために、共生共進化アルゴリズムと GA に基づく 3 種類の探索手法との比較を行う。

### 4.1 比較手法

比較する 4 手法を以下に示す。

#### 1. Symbiotic algorithm

解とオーダーを 2.2 節で紹介したアルゴリズムに基づき共進化させる。

#### 2. Fixed GA

2.2 節のアルゴリズムにおける集団 PA(解) の進化過程のみを実行する。ここで、オーダーは固定されたオーダー  $[1\ 2\ \dots\ N]$  を用いる。

#### 3. Random GA

2.2 節のアルゴリズムにおける集団 PA(解) の進化過程のみを実行する。ここで、使用されるオーダーは、解を生成する度にランダムに生成される。

#### 4. Combinational GA

1 個体が解とオーダーから成り、GA により進化する。

step1: 初期個体集団 PA(解とオーダー) の生成。

step2: 親個体の選択。

step3: 子個体の生成。

親 2 個体を各々の個体が持つオーダーを用いて、解を順序表現に変換し、交叉点をランダムに決定する。

case1: 交叉点が解の遺伝子座間であれば、解の部分に遺伝オペレータを適用する。

case2: 交叉点が解とオーダーの間であれば、両方のオーダーを入れ換える。

case3: 交叉点がオーダーの遺伝子座間であれば、オーダーの部分に遺伝オペレータを適用する。

step4: 子個体の評価 (case1 のみ実行)。

step5: 子個体の追加および淘汰 (case1 のみ実行)。

子個体の評価値と集団 PA の最小評価値を比較し、子個体が高ければ、子個体をその個体と入れ換える。そうでなければ、淘汰する。ここで、終了条件を満たさなければ step2 へ戻る。

### 4.2 シミュレーション結果

各パラメータの値を Table 1 に示す。

シミュレーションの結果をもとに、以下の解析を行う。

#### 1. 解獲得の性能比較

Table 1. パラメータ

問題サイズ ( $N$ )	8	16	32
集団サイズ	10	100	200
突然変異率	0.0625	0.03125	0.015625
履歴	10	20	20
PA の進化過程実行回数	10	20	20
PB の進化過程実行回数	1	1	1
生成個体数 (終了条件)	1000	100000	200000

#### 2. オーダー集団の多様性

#### 3. 使用されているオーダーの有効性

4.2.1 解獲得の性能比較 問題サイズに応じた各手法の最適解獲得率を Table 2 に示す。

Table 2. 10 試行における最適解獲得率

問題サイズ	8	16	32
Symbiotic algorithm	80%	100%	80%
Fixed GA	50%	70%	10%
Random GA	80%	90%	80%
Combinational GA	80%	80%	20%

Table 2 より、Symbiotic algorithm および Random GA は、問題サイズに関わらず、高い獲得率を示していることがわかる。一方、Fixed GA および Combinational GA は、問題サイズが大きいと獲得が困難であることが示されている。 $N = 32$  におけるシミュレーション結果の 1 例を Fig. 5 に示す。

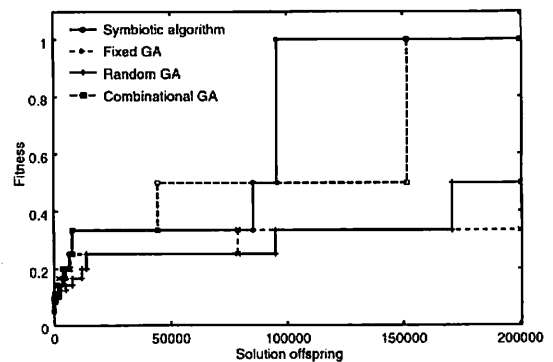


Fig. 5. 解の適応度推移

Fig. 5 は、解の最大適応度の推移を示している。同図において、横軸は生成される解の子個体の数を示しており、縦軸は、解の適応度を示している。1.0 の値は、最適解の獲得を意味する。Symbiotic algorithm は、徐々に解を改善し、最終的に最適解を獲得している。一方、Fixed GA は、獲得に失敗している。その理由として、1 つの固定されたオーダーを用いているため、解が局所解に陥った場合には、脱出が困難になると考えられる。それに対し、Random GA は、常に様々なオーダーを適用するため局所解からの脱出に成功し、結果として最適解を獲得している。さらに、Combinational GA は、複数のオーダーが存

在しているにも関わらず、最適解の獲得に失敗している。その理由として、解は解自身と結合しているオーダー以外利用できないため、最適解を生成する可能性が低くなると考えられる。

これらの結果から、オーダーの多様性および有効なオーダーの利用が最適解の獲得に貢献していると考えられる。そこで、それらを検証するためにさらなる解析を行う。

**4.2.2 オーダー集団の多様性** 本稿では、オーダー集団の多様性を確認するために、まず、ある基準オーダーを定め、そのオーダーに対する各オーダー個体の類似度を計算する。その類似度の総和を集団の類似度と設定する。基準オーダーは、 $[1\ 2\ \dots\ N]$ とした。個体の類似度は、基準オーダーに一致する bit を 1、一致しない bit を 0 で表し、それを 10 進数に変換した値とする。子個体が 1000 個体生成される毎に、その時のオーダー集団の類似度を計算し、リターンマップ [16] で示す。Random GA の場合は、使用されるオーダー 200 個体ずつを 1 集団として計算した。3 手法のオーダー集団の類似度推移を Fig. 6 に示す。

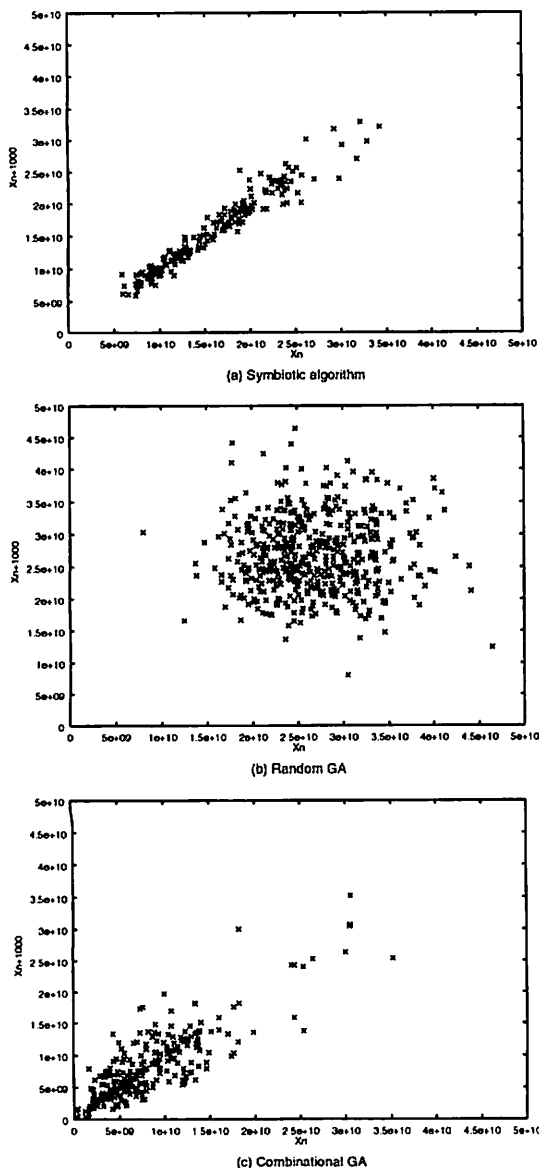


Fig. 6. オーダー集団類似度のリターンマップ

Fig. 6 において、横軸は  $n$  個体生成時の集団の類似度を示し、縦軸は、 $n+1000$  個体生成時の類似度を示している。右上への収束は基準オーダーに近いオーダーへの収束を、左下への収束は基準オーダーと類似しないオーダーへの収束を意味する。また、 $y = x$  上に点が存在するという事は、1000 個体生成の間、オーダー集団に変化がない、もしくは、変化したとしても入れ替わる前と同じスキーマを有するオーダーに入れ替わる状況を示している。3 手法を比較すると、Random GA が規則性なしに点在しているのに対し、Symbiotic algorithm および Combinational GA は、 $y = x$  の周囲に点在していることが確認できる。これは、アルゴリズムがオーダー集団に対して新しいオーダーを追加していく設計であるため、追加されるオーダーのみの変化が反映されるからである。2 手法を詳細に比較すると、Combinational GA が類似度の低い個体を集中して獲得しているのに対し、Symbiotic algorithm は、やや Random GA と近いオーダー集団を獲得している。また、 $y = x$  上に存在する点の数を調査したところ、Symbiotic algorithm は 1 点もなく、Combinational GA は、プロットしたポイント中、4 分の 1 が  $y = x$  上に存在した。したがって、Combinational GA は、あるスキーマを持つオーダーに収束している状況が多いことがわかる。これらの結果から、Symbiotic algorithm は、常にオーダーが変化しており、徐々に探索点を移動させ、広範囲にわたってオーダーを獲得していることがわかる。

集団におけるスキーマの数が、各手法毎にどの程度の差があるのかを確認するために、実際のスキーマの数の推移を Fig. 7 に示す。検証したスキーマの種類はそれぞれ、 $\{(1\ 2), (1\ 3), \dots, (1\ 32), \dots, (2\ 3), (2\ 4), \dots, (31\ 32)\}$  であり合計 496 種類である。

Fig. 7 において、縦軸はオーダー集団の各スキーマの数であり、横軸は生成された個体数である。これらの結果は、明らかに異なっていることがわかる。Random GA は、定常的に多様性を維持している。また、Combinational GA は、集団内に存在する割合の高いスキーマが多く、世代が進むにつれ、さらにいくつかのスキーマの数が増えていることから、徐々に多様性が失われていることがわかる。Combinational GA では、オーダーと解が結合しているため、解の子個体が生成される場合には、親個体に結合していたオーダーが子個体に結合する。つまり、子個体が集団に入る場合には、親個体が持っているオーダーと全く同じオーダーが集団内に追加されることになる。結果として、オーダーの多様性が徐々に失われる。一方、Symbiotic algorithm は、Combinational GA よりも各スキーマの数が少なく、ある程度の多様性を維持している。また、他のスキーマの数よりも若干多いスキーマが常に存在し、そのスキーマが入れ替わっていることから、多様性を保ちつつも、Random GA と異なりある方向性を持って探索していることがわかる。

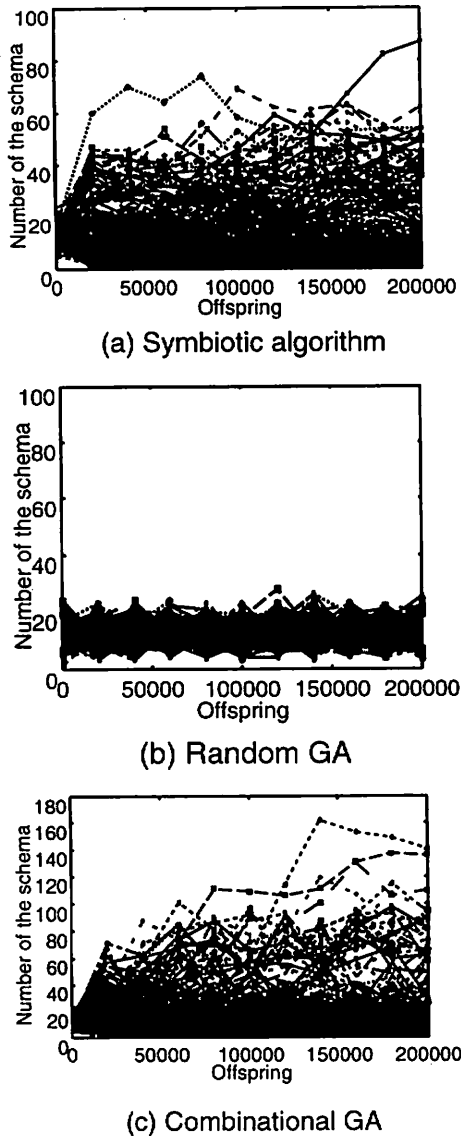


Fig. 7. 各手法におけるスキーマの推移

**4.2.3 使用されているオーダーの有効性** Symbiotic algorithm において、解の生成時に使用されているオーダーの有効性を検証するために、オーダーが最適解を作り得る可能性の割合(期待値)を計算する。あるオーダーの期待値を計算する場合、解集団からの2つの解の組合せ全ておよび全ての交叉点について、最適解と成り得るかを調べなければならない。そのためには、膨大な計算量を必要とするため、本稿では、3つの問題サイズのうち、一番小さいサイズの  $N=8$  の場合について、検証する。ここでは、集団からオーダーを選択する Symbiotic algorithm と Combinational GA の解析結果について考察する。解析結果を Fig. 8 に示す。解の進化との関係を確認するために、Fig. 9 にシミュレーションにおける解の適応度推移を示す。

Fig. 8 において、左図は各手法で使用されたオーダーの期待値の推移を示しており、右図は、集団内におけるその期待値の順位を示している。Symbiotic algorithm において、集団内のオーダーがある程度評価されるまでは、使

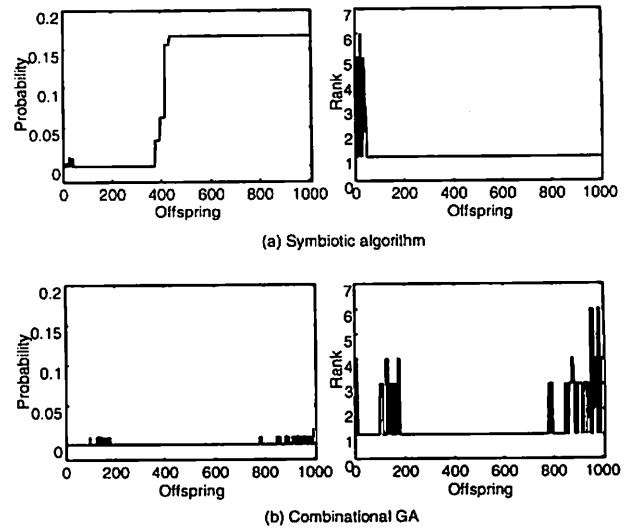


Fig. 8. 使用されたオーダーの期待値および順位

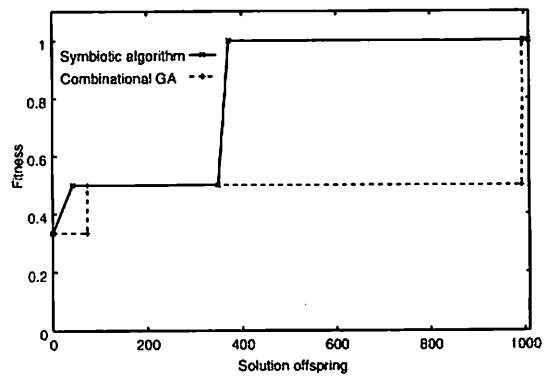


Fig. 9. 解の適応度推移

用オーダーの順位は振動しているものの、初期の段階で常に1位の期待値を持つオーダーを採用していることがわかる。したがって、最適解を作れる可能性が生じると、優先的にそのオーダーを用いて有効な解、もしくはオーダーを徐々に生成し、オーダーの期待値を上げるだけでなく、最終的に最適解を獲得している。一方、Combinational GA は、200 世代前に出現した最適解を生成できるオーダーによって解を向上させているが、その後、そのオーダーをいかして解を生成することができず、オーダーが淘汰されている。また、800 世代以降もオーダーが出現したにもかかわらず、それを連続して使用していないため、最適解の獲得までに時間がかかっている。Combinational GA では、解とオーダーが結合しているため、オーダーの選択は解の適応度に依存する。つまり、良いオーダーであっても適応度の低い解に結合されている場合、選択される可能性が低いという状況が生じる。したがって、有効なオーダーが常に使われるとは限らない。Symbiotic algorithm は、解とオーダーを分割することによって、この問題を解消できる。オーダーは、オーダー自身が貢献した解から適応度が与えられるため、1つの解だけに依存せず独立した適応度を持つことになる。さらに、その適応度を元に独立に進化するため、有効なオーダーを淘汰する可能性が低くなり、結果

として有効なオーダーを利用しての探索が可能である。そのため、大きな問題サイズであっても同じ原理で探索が進み、最適解を獲得する可能性が高いといえる。さらに、常に有効なオーダーを用いることから、Combinational GAよりも早く確実に獲得できるといえる。

#### 4.3 考察

以上のシミュレーション結果および解析結果より、各手法の探索の特徴をまとめると以下ようになる。

##### 1. Symbiotic algorithm

2つの集団に分けることによって、オーダーにある程度の多様性を持たせることができる。また、オーダーに、独立の評価を与え進化させることによって、有効オーダーを利用することを可能にしている。これらの作用により最適解を獲得する。

##### 2. Fixed GA

1つのオーダーに固定されているため、局所解に陥りやすい欠点はあるが、オーダーが有効なオーダーであった場合には、他の手法よりも最適解を早く獲得する可能性がある。

##### 3. Random GA

多くのオーダーを利用することにより、最適解を獲得を実現している。

##### 4. Combinational GA

解の収束に伴いオーダーの多様性も失われるため、世代が進むにつれ最適解の獲得が困難になる。

以上のことから、様々な種のオーダーの利用および、有効オーダーの利用は、どちらも最適解の獲得に継る要因となる。したがって、両方を備えているSymbiotic algorithmは、解探索に非常に有効な手法であるといえる。

#### 5. おわりに

本稿では、解探索における2つの要素を2集団に分割して進化させる共生共進化アルゴリズムの有効性および特徴を検証するために、N-クイーン問題へ適用し、獲得オーダーに関して解析を行った。シミュレーション結果および解析結果より、共生共進化アルゴリズムは、オーダーの多様性の維持と有効オーダーの利用によって、最適解を獲得するアルゴリズムであることを示した。また、これらの特徴は、1つの集団を2つの集団に分割することによって得られることを示した。これらの結果から、共生共進化アルゴリズムは、解とオーダーの進化を設定できる問題に対して有効に機能するアルゴリズムであるといえる。

#### 文献

- [1] D.E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, 1989.
- [2] W.D. Hillis, *Co-evolution parasites improve simulated evolution as an optimization procedure*, Artificial Life II, Addison-Wesley, pp.313-323, 1991.
- [3] Peter J. Angeline and Jordan B. Pollack, *Competitive Environments Evolve Better Solutions for Complex Tasks*, In Proceedings of the Fifth International Conference on Genetic Algorithms (ICGA'93), pp.264-270, Morgan Kaufmann, 1993.
- [4] C.D. Rosin and R.K. Belew, *Methods of Competitive Co-evolution (Finding Opponents Worth Beating)*, In Proceed-

- ings of the Sixth International Conference on Genetic Algorithms (ICGA'95), Morgan Kaufmann, 1995.
- [5] Dario Floreano and Stefano Nolfi, *God Save the Red Queen! Competition in Co-Evolutionary Robotics*, In Proceedings of the second International Conference on Genetic Programming (ICGP'97), pp.398-406, Morgan Kaufmann, 1997.
- [6] Moeko Nerome, Koji Yamada, Satoshi Endo, Hayao Miyagi, *Competitive Co-evolution Based Game-Strategy Acquisition with the Packaging*, In Proceedings of the Second International Conference on Knowledge-Based Intelligent Electronic Systems (KES'98), Adelaide, South Australia, pp.184-189, 1998.
- [7] 河田 雅圭, 進化論の見方, 紀伊國屋書店, 1989.
- [8] J. Paredis, *The Symbiotic Evolution of Solutions and their Representation*, Proceedings of the Sixth International Conference on Genetic Algorithms (ICGA'95), Eshelman, L. (editor), Morgan Kaufmann, 1995.
- [9] J. Paredis, *Symbiotic Coevolution for Epistatic Problems*, Proceedings of the 12th European Conference on Artificial Intelligence (ECAI'96), W. Wahlster(editor), John Wiley & Sons, Ltd., 1996.
- [10] 伊庭 齊志, 遺伝的アルゴリズムの基礎 -GAの謎を解く-, オーム社, 1994.
- [11] Grefenstette, J. J., Gopal, R., Rosmaita, B., Van Gucht, D., *Genetic Algorithm for the traveling salesman problem*, Proceedings of the First International Conference on Genetic Algorithms (ICGA'85), Morgan Kaufmann, 1985.
- [12] Lincorn, Boxshall, Clark (editor), *A dictionary of ecology, evolution and systematics*, CAMBRIDGE UNIVERSITY PRESS, 1982.
- [13] Michael Allaby (editor), *THE OXFORD DICTIONARY OF NATURAL HISTORY*, OXFORD UNIVERSITY PRESS, 1985.
- [14] J. Paredis, *Steps towards coevolutionary classification neural networks*, Proceedings of the Fourth International Workshop on Synthesis and Simulation of Living Systems (Artificial Life IV), R. Brooks, and P. Maes (editor), pp. 102-108, MIT Press/Brandford Books, 1994.
- [15] Starkweather, T., McDaniel, S., Mathias, K., Whitley, D., *A Comparison of Genetic Sequencing Operators*, Proceedings of the Fourth International Conference on Genetic Algorithms (ICGA'91), Morgan Kaufmann, 1991.
- [16] R. Odagiri, Md. Monirul Islam, M. Okura, T. Asai, K. Murase, *Deterministic Chaos in Sensory Information of Real Autonomous Mobile Robot Khepera*, Collection of Papers ER2000, pp. 71-78, Applied AI Systems Inc., 2000.