Vol.2009-MPS-75 No.5
2009/9/10

強化学習を用いた進化的アルゴリズムのパラメータ学習

櫻井義尚† 鶴田節夫†

遺伝的アルゴリズム（GA）など進化的アルゴリズムを利用した探索手法はパラメータが適切に設定されていれば，高い能力を発揮するが，そのパラメータ設定は難しく，問題パターン毎の最適化手法を個別に開発する必要があった．そのため高度な専門知識と大量の検証実験を必要としてきた．
この問題を解決するため，進化的アルゴリズムのパラメータを適応的に制御する適応型パラメータ制御と呼ばれる手法が新しく提案されている．しかし，これは主に良い個体を生成した探索オペレータの選択確率を上げていくといった方法で，即時的な探索結果だけをパラメータ制御に反映させるため，近視眼的な最適化になる可能性がある．一方，強化学習を用いて長期的に最適な GA のパラメータ制御を実現しようとする手法も提案されているが，探索オペレータの計算コストや GA の多点探索手法としての特性を考慮していないため，効率改善の余地がある．
本論文では，報酬決定則として探索オペレータの計算コストと GA の多点探索手法としての特性を考慮した報酬決定則を実装した強化学習を用いる事により，効率的に進化的アルゴリズムのパラメータ制御を行う手法を提案する．

A Parameter Control Method for Evolutionary Algorithms using Reinforcement Learning

YOSHITAKA SAKURAI† SETSUO TSURUTA†

The search technique using the evolutionary algorithm like genetic algorithm (GA) is very effective if the parameter is appropriately set. The optimum parameter setting is difficult, has needed difficult, advanced expertise and a large amount of verification experiment. Therefore, it has needed advanced expertise and a large amount of verification experiment. In order to solve this problem, the technique that is called an adaptive parameter control that controls the parameter of the evolutionary algorithm adaptive is new and it is proposed. In this paper, parameter controlling method using the reinforcement learning that mounts the reward decision rule that considers the characteristic of the multipoint search and the calculation cost of the search operator.

1. はじめに

遺伝的アルゴリズム（Genetic Algorithms, GA）[1] をはじめとする進化的アルゴリズムは生物の進化をモデル化した近似解の探索アルゴリズムである．これらの近似解法において最適な方策は，一般に問題規模，応答時間・精度の要求などの問題種別によって異なり，例えば巡回セールスマン問題（Traveling Salesman Problems, TSP）[2]では巡回地点の配置や到着時間の制約などの問題パターン，に依存する．このため，問題種別･パターン毎の最適化手法を現場や該当分野で個別に開発する必要があった[3)4)]．

進化型計算法のような近似解法では、乱数を使うためもあり，問題に適した方策やパラメータ設定の汎用的理論は現在のところ存在しない．一般に，近似解探索手法はパラメータが適切に設定されていれば，高い能力を発揮するが，そのパラメータ設定[5]は難しい．

これまで，実用の中大規模（数十から千数百拠点）巡回セールスマン問題の近似解を高精度かつリアルタイムに解く研究を行ってきた[3)4)]．その過程で，近似解法の各方策には得意/不得意な問題が有り、その解決が必要であった．

そこで，問題に適した方策を(半)自動で選べれば，より効率的・汎用的に解けると考えた．しかし，各方式とそれに付随する多数のパラメータやヒューリスティクスの組合せの中からの選択のため，膨大な回数の試行が必要であり，その効率が重要となる．

一方，強化学習の研究において，状態概念を持つこの枠組みは熟達者の技能を長期的視点からも効率良くパラメータ学習するのに有効であった[6]．また，個体の適応度を報酬と見なせば強化学習は進化・学習の構造において GA と類似し親和性が高い．以上 2 点から，強化学習なら個体の最適化を目指して、GA のパラメータの動的制御が期待できる．

進化的アルゴリズムのパラメータ設定についての先行研究としては，Eiben らによる文献 7)によると，前もってパラメータの設定を行うパラメータチューニング（parameter turning）と実行中にパラメータ設定を行うパラメータ制御（parameter control）に分けられる．パラメータチューニングとしては，文献 8)のように，設定されたスケジュールに従って様々なパラメータ設定での試行を行い，この結果からパラメータ設定を行う手法が提案されている．これはすべてのパラメータを設定できるが，後述の適応型パラメータ制御と違い探索過程での情報を利用しないため，効率が悪い．その上，パラメータは探索過程で動的･適応的に変えることが出来ず，固定値となる．

パラメータ制御としては，進化的アルゴリズムのパラメータを適応的に制御する適

† 東京電機大学 情報環境学部 情報環境学科
Department of Information Environment, School of Information Environment, Tokyo Denki University

応型パラメータ制御（adaptive parameter control）が提案されている．例えば文献 9)や 10)がこれにあたる。遺伝的アルゴリズムの探索オペレータである交叉や突然変異オペレータのそれぞれに重みを設定し，これらのオペレータにより生成された探索点（つまり、子個体）の評価に基づいてパラメータを設定するものである．しかし，これは，選択結果を直ちに評価して報酬を与える必要があるので交叉や突然変異などのオペレータの近視眼的なパラメータ設定にしか使えない．

長期的に最適な GA のパラメータ制御を実現する手法として，強化学習を用いたものが提案されている．初期の研究としては，Q-learning を用いて探索オペレータの選択方策を学習した RL-GA [11]がある。次に、リアルタイム性など実応用を考慮して、この学習アルゴリズムを事前トレーニングが必須ではない Sarsa に変更した SCGA [12]が提案されている．しかし，これらの手法では，探索オペレータの計算コストや GA の多点探索手法としての特性を考慮していないため，効率改善の余地がある．

本論文では，報酬決定則として探索オペレータの計算コストと GA の多点探索特性を考慮した強化学習を用いる事により，効率的に進化的アルゴリズムのパラメータ制御を行う手法を提案する。以下では、これを実装した GA を多点-時間考慮型 RL 制御 GA (Multi-point and Time cost regarded Reinforcement Learning Control Genetic Algorithm, MT-RLcGA) と呼ぶ.

次節では，適応型パラメータ制御などの新しい学習方式，あるいは，強化学習を GA に適応した方式の問題点と，その解決のためのアプローチを述べる．3 節では，そのアルゴリズム MT-RLcGA およびその振舞いや予想効果を述べ, 4 節で結果をまとめる．

2. 研究課題・アイデアと問題点

2.1 遺伝的アルゴリズム

遺伝的アルゴリズム（Genetic Algorithms, GA）[1] は生物の進化をモデル化した近似解の探索アルゴリズムである．GA は，はじめに初期解生成により解の候補を遺伝子で表現した個体群（初期集団）を生成する．そして，その個体群をもとに交叉，突然変異などの探索オペレータを用いて新しい個体（解候補）を生成し，適応度に基づいて選択・淘汰などを繰り返す事により解の探索を行う手法である．

以下では，GA を用いて組合せ最適化問題として有名な対称ユークリッド型巡回セールスマン(Traveling Salesman Problem, TSP) [13]を解く場合を考える．TSP は複数の拠点を巡回する最短経路を見つける問題であり, これを定式化すると以下のようになる．

巡回路として重み付き完全グラフ $G=(V,E,w)$を与える．ここで V はノード（頂点）集合であり，ノード（頂点）v_i （i=1,…,N）は巡回する拠点を表す．N はノード（頂点）数である．E は枝集合であり，枝 e_{ij} はノード v_i とノード v_j を結ぶルートを表す．w は枝の重み集合であり，枝の重み d_{ij} はノード v_i とノード v_j の 2 地点間距離であり $d_{ij}=d_{ji}$ とする．

グラフ $G=(V,E,w)$上で最も短い巡回路長をもつハミルトン閉路を求める問題を TSP と呼び，この解を最適解，ハミルトン閉路を実行可能解と呼ぶ．

解の候補となる個体は TSP におけるツアー(配送ルート)を表現する染色体を持つ．その遺伝子型は図 1 のように巡回順にノード番号(配送先の ID 番号)を並べた構造になっている．各遺伝子は配送先の ID 番号を示す．

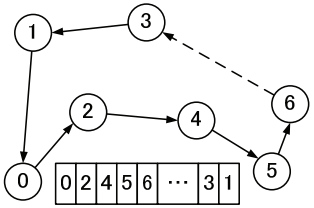

図 1. TSP における遺伝子型
Figure 1 Genetic Type of TSP.

2.2 パラメータ制御

このような TSP の解候補（最適解に近いハミルトン閉路）を生成する探索オペレータとしてはヒューリスティックスを用いる事により効率的に探索を行う様々なものが提案されている [14]．また，複数の探索オペレータを使う事により効率的に探索を行う手法も提案されている．対象とする問題のタイプや基となる個体，集団の状態によってどの探索オペレータが効率的に働くかは異なるため，そのパラメータの調整が不可欠である．

これまで，実用の中大規模（数十から千数百拠点）巡回セールスマン問題の近似解を高精度かつ実時間に解く研究を行ってきた [3] [4]．その過程で，近似解法の各方策には得意/不得意な問題が有ることが分ってきた．そこで，問題に適した方策を(半)自動で選べれば，より効率的・汎用的に解けると考えた．しかし，各方式とそれに付随する多数のパラメータやヒューリスティクスの組合せの中からの選択のため，膨大な回数の試行が必要であり，その効率が重要となる．

一方，強化学習の研究において，状態概念を持つこの枠組みは熟達者の技能を長期的視点からも効率良くパラメータ学習するのに有効であった [6]．また，個体の適応度を報酬と見なせば強化学習は進化・学習の構造において GA と類似し親和性が高い．以上の 2 点から，強化学習なら個体の最適化を目指して GA などのパラメータを動的かつ長期的にも効率良く制御できると考えた．

2.3 強化学習

強化学習(Reinforcement Learning, RL) [15]とは，ある環境内におけるエージェントが，現在の状態を観測し，取るべき行動を決定する問題を扱う機械学習の一種である．エージェントは行動を選択することで環境から報酬を得る．強化学習は一連の行動を通じて報酬が最も多く得られるような方策(policy)を学習する．

エージェントと環境は離散的な時間ステップ t=0,1,2,…の各々において相互作用を行う．各時間ステップ t においてエージェントは何らかの環境の状態（state）の表現 $s_t \in S$ （S は可能な状態の集合）を受け取り，これに基づいて行動 $a_t \in A(s_t)$を選択する（$A(s_t)$は状態 s_t において選択可能な行動の集合）．1 ステップ後にその行動の結果として報酬 $r_{t+1} \in R$ を受け取り，新しい状態 s_{t+1} に移る．各時間ステップにおいて，状態 s から可能な行動 a を選択する確率の写像は方策 $\pi_t(s, a)$と呼ばれる．強化学習は最終的に受け取る報酬の量を最大化するように方策を学習する．最終的に受け取る報酬の量は割引の概念を実装し，式(0.1)の様に定式化される．

$$R_t = \sum_{k=0}^{\infty} \gamma^k r_t + k + 1 \qquad (0.1)$$

γ は割引き率（discount rate）と呼ばれるパラメータで，0≤γ≤1 であり，将来の報酬が現在においてどれだけの価値があるかを決定する．

方策 π のもとで状態 s において行動 a を取ることの価値を $Q^{\pi}(s, a)$で表し，方策 π に従った時の期待報酬と定義する．

$$Q^{\pi}(s,a) = E_{\pi}\left\{R_t \mid s_t = s, a_t = a\right\} \qquad (0.2)$$

Q^{π} を方策 π に対する行動価値関数と呼ぶ．強化学習として有名な Q-learning では，以下のように更新することにより行動価値関数を近似する．

$$Q(s_t,a_t) \leftarrow (1-\alpha)Q(s_t,a_t) + \alpha\left[r_{t+1} + \gamma \max_a Q(s_{t+1},a)\right] \qquad (0.3)$$

α は学習率で，0≤α≤1 である．Q-learning では，使われている方策とは独立に最適行動価値関数 Q^*を直接近似する off-policy 型のアルゴリズムである．on-policy 型のアルゴリムとしては Sarsa が提案されており，以下のように行動価値関数を近似する．

$$Q(s_t,a_t) \leftarrow (1-\alpha)Q(s_t,a_t) + \alpha\left[r_{t+1} + \gamma Q(s_{t+1},a_{t+1})\right] \qquad (0.4)$$

2.4 従来方式の問題点

進化的アルゴリズムのパラメータを適応的に制御する手法，適応型パラメータ制御としては，良い個体を生成した探索オペレータの選択確率を上げていく [9) 10)]などの手法が提案されている．しかし，これらの手法は探索オペレータにより生成された新個体，つまり，即時的な探索結果だけを頼りにパラメータを制御するため，近視眼的な最適化になる可能性がある．また，直接評価するための指標が必要なため，GA のパラメータ制御としては，交叉や突然変異など探索点の生成に直接関係するパラメータにしか適用できない．

これに対して，強化学習を用いたパラメータ制御手法が提案されている．強化学習は長期的に得られる報酬を最大化することを目的として方策を学習するアルゴリズムであり，これにより長期的に最適なパラメータ制御の方策を獲得できる可能性がある．

強化学習を GA に適応した初期の研究として，Q-learning [16)] を用いて探索オペレータの選択方策を学習した RL-GA [11)] がある．これは，実行の前にトレーニング期間が必要な off-Policy 型の手法である Q-learning を使っているが，実際の応用では事前のトレーニングは好ましくない場合がある [12)]．

この問題を解決するため，文献 12)では，on-Policy 型の強化学習手法である Sarsa（0）を用いた SCGA を提案している．on-Policy 型の手法はトレーニングと実行を分ける必要が無いため，一般に毎回巡回先が変わる実応用の面で改善されている．

これらのアルゴリズムでは，親と子の評価差を報酬としているが，GA オペレータの計算コストはそれぞれ異なり，計算コストの高いオペレータの方が良い結果を生むのは当然である．このような報酬の与え方は計算時間を考えずに最終的に良い解を見つける場合には有効であるが，出来るだけ短い時間で最適な解を見つけ出す場合には適さない．そのため，報酬には単位時間あたりの改善度を用いる必要があると考えられる．

また，文献 11)や 12)では， 1 個体生成毎に報酬を計算し行動価値関数を更新している．しかし，GA は多点探索手法であり，1 個体（1 つの探索点）毎に報酬を与えるのではなく，集団（複数探索点）での探索結果から報酬を計算することにより，より効率的な方策を学習できる可能性がある．

3. 提案手法

本節では，GA の多点探索特性や計算コストを考慮した報酬を与えることにより，効率的なパラメータ制御を実現する強化学習によるパラメータ制御手法を提案し，これを実装した GA アルゴリズムを多点-時間考慮型 RL 制御 GA (Multi-point and Time cost regarded Reinforcement Learning Control Genetic Algorithm, MT-RLcGA) と呼ぶ．

本アルゴリズは RL-GA [11]をベースとして，下記で提案する報酬決定則と行動価値関数の更新ステップを導入したものである．行動価値関数の更新には SCGA[12]と同様に Sarsa を用いた．

3.1 多点-時間考慮型 RL 制御 GA (MT-RLcGA)

提案手法 MT-RLcGA は，GA の探索オペレータの選択を強化学習エージェントにより制御する．

GA は複数の探索オペレータ（交叉，突然変異オペレータ）を実装しており，状態に応じてエージェントにより選択された探索オペレータにより子個体を生成する．この子個体の評価から，エージェントは報酬を受け取り，探索オペレータの選択方策を学習する．生成された子個体集合と親個体集合から適応度に基づいて上位の個体群が選択され，次世代の親個体集合になる．これを繰り返すことにより解の探索を行う．

3.1.1 学習アルゴリズム

エージェントの学習タイミングの異なる 2 タイプのアルゴリズムを提案する．1 つは 1 つの個体生成毎にエージェントの行動価値関数の更新を行う個体毎 Q 更新．もう 1 つは, 1 世代の新個体をすべて生成してからエージェントの行動価値関数の更新を行う集団毎 Q 更新である．集団毎 Q 更新を用いた MT-RLcGA のアルゴリズムを図 2 に示す．

RL-GA，SCGA では，個体毎 Q 更新が用いられている．これらの研究では親と子の評価差を報酬として用いているが，この場合，エージェントは個体毎に解の改善度合いの合計が最大になるように学習する．しかし，GA は複数の個体において多点探索をするからこそ効率良い探索が可能となる．そのため，多点探索でのパラメータの最適化が重要になる．つまり，良い個体を生成したオペレータの選択確率を上げる事が全体的な性能の向上をもたらすとは限らない．そこで，一つの個体生成だけで評価せず, 1 世代全体で他のオペレータによる処理の影響などを含めて評価することにより，より効率的な GA の多点探索の方策が獲得できる可能性がある．

また，行動価値関数の更新ルールには，on-policy での方策を学習するため Sarsa を用いる．

```
Q(s, a)を初期化
初期個体群の生成
  s←現在の状態
  終了世代まで繰り返し：
  子個体が M 個作られるまで繰り返し：
    s で取る行動 a を選択　式(0.8)
    時間計測開始
    行動 a に対応する方法で親を選択
    行動 a に対応するオペレータで子個体生成
    親と子の適応値の差を計算（評価）
    時間計測終了
    個体毎の報酬 r_m を受け取り記憶　式(0.9)
  記憶報酬{r_m}_m=1…M から集団探索報酬 r_pop を算出
  適応値の低い個体を淘汰
  行動 a の結果，r と s'を観測　式(0.12)
  状態価値関数を更新　式(0.4)
  s←s'
```

図 2. 集団毎 Q 更新アルゴリズム

3.1.2 状態

GA の探索過程の環境を表現する状態 s_t としては，RL-GA と同じく，現在の計算時間，集団の平均適応度，集団の適応度のエントロピーを用いる．

計算時間：開始から終了予定時間までの時間を対数時間 $\log t$ に変換し，4 つに等分割する．

集団の平均適応度：下記のように最初の集団適応度により正規化された現在の集団の平均適応度（式(0.5)）を 4 つに分割する（[0,0.3),[0.3,0.4),[0.4,0.6),[0.6,1]）．M は集団個体数，$f(x)$は個体 x の適応値，f^t は t 世代での適応値を意味する．

$$\bar{f}^t = \frac{\sum_{m=1}^{M} f(x_m^t)}{\sum_{m=1}^{M} f(x_m^0)} \qquad (0.5)$$

集団の適応度のエントロピー：集団の適応度のシャノンエントロピーHは式(0.7)のように計算される．

$$p_m = f(x_m) / \sum_m^M f(x_m) \qquad (0.6)$$

$$H = \sum_{m=1}^{M} p_m \log_2 \frac{1}{p_m} \qquad (0.7)$$

エントロピーはすべての個体が同じ適応度の時に最小，適応度が均一に分散しているときに最大になる．

3.1.3 行動

エージェントは行動として，親の選択方法と探索オペレータの組合せを選択する．親の選択では，適応度の高い親(F)と低い親(U)に分類し，それぞれの組合せを選択する．交叉用に 4 種類{FF, FU, UF, UU}，突然変異用に 2 種類{F, U}になる．

探索オペレータの選択では, 交叉として実装された 3 手法(MPX,GER,PMX)と突然変異として実装された 4 手法(MOVE,SCRM,INVR,SWAP)の 7 種類から選択する [11]．

行動の選択方策 π は行動価値関数に基づいて決定されるが，常にグリーディな行動を選択すると，行動価値関数の学習が進まない．そのため，確率的に探査する行動選択則として，ε グリーディやソフトマックスなどがある．本手法では，推定価値に従って各行動の選択確率の重み付けをすることにより探査と知識利用のバランスを取るソフトマックス行動選択則を利用する（式(0.8)）．

$$\pi(s_t, a_t) = \frac{e^{\beta Q(s_t, a_t)}}{\sum_{a'_t=1}^{N} e^{\beta Q(s_t, a'_t)}} \qquad (0.8)$$

3.1.4 報酬

RL-GA，SCGA では，親と子の適応値の差を報酬としている．しかし，GA 探索オペレータの計算コストはそれぞれ異なり，計算コストの高いオペレータの方が良い結果を生むのは当然である．このような報酬の与え方は計算時間を考えずに最終的に良い解を見つける場合には有効であるが，出来るだけ短い時間で最適な解を見つけ出す場合には適さない．本アルゴリズムでは，GA 探索の高速化を図るため，報酬には単位時間あたりの適応値の改善量を用いる．個体毎 Q 更新の場合，交叉による報酬を式(0.9)のように定義した．

$$r = \frac{\max\{f(x) \mid x \in parents\} - \max\{f(x) \mid x \in offspring\}}{\max\{f(x) \mid x \in parents\} \times Time_{\pi(s,a)}} \qquad (0.9)$$

$Time_{\pi(s,a)}$は状態 s の時に行動 a を実行して新個体を生成するのにかかった時間である．突然変異の場合，親は 1 つなので適応度の大きい方の選択が無くなる．

集団毎 Q 更新の場合，報酬は個体毎の報酬 r_m だけでなく，1 世代に生成されたすべての個体の評価から求める．集団探索での報酬 r_{pop} を式(0.10)または(0.11)で定義すると，報酬 r は式(0.12)のように，個体毎の報酬 r_m と集団探索での報酬 r_{pop} との重み結合和により定義される．ω は集団評価寄与率と呼び，集団探索結果が報酬に与える影響度合いを表す．

$$r_{pop} = \left(\sum_{m=1}^{M} r_m \right) \Big/ M \qquad (0.10)$$

$$r_{pop} = \max\{r_m \mid m = 1 \ldots M\} \qquad (0.11)$$

$$r = (1-\omega) r_m + \omega r_{pop} \qquad (0.12)$$

3.2 提案アルゴリズムの動作シナリオと効果

提案する報酬の与え方による GA の動作とその効果を検証する．まず，計算時間による報酬の割引について考える．個体毎 Q 更新において，状態 s_t において行動 a_t を選択した結果，3CPU タイムで 0.2 適応値が改善し，状態 s_{t+1} において行動 a_{t+1} を選択した結果，1CPU タイムで 0.1 適応値が改善した場合を CASE1 とし，状態 s_t において行動 a'_t を選択した結果，1CPU タイムで 0.1 適応値が改善し，状態 s_{t+1} において行動 a_{t+1} を選択した結果，1CPU タイムで 0.1 適応値が改善した場合を CASE2 とする．報酬を改善した適応値し，簡単のため割引率を 1 に設定すると，受け取る報酬は CASE1 が 0.3，CASE2 が 0.2 となり，CASE2 の方が短時間で同じ適応値の改善が得られるにもかかわらず低くなる．これに対して報酬を 1CPU タイムあたりに改善した適応値とすると，CASE1 は 0.17，CASE2 は 0.2 となり CASE2 の行動価値の方が高くなる．

これにより，従来法では評価されなかった小さい計算コストで複数回探索を行った方が効果のある探索オペレータの評価が適切になされるため，より短時間での最適化方策が見つかる可能性がある．

次に集団での探索結果から報酬を計算し，1 世代毎にまとめて行動価値関数の更新を行う場合について考える．

最適化効率が高い反面，解の多様性が失われやすい探索オペレータ A と最適化効率は低いが，解の多様性の維持に効果的な探索オペレータ B などが実装された GA があ

ったとする. 個体生成毎に報酬を与えると, 探索オペレータ A の選択確率が高くなる．しかし，探索オペレータ A ばかりが用いられると集団の多様性が低下し将来的には報酬が低くなる可能性がある．強化学習では，将来的な報酬も見積もり，行動価値関数を更新する．そのため，集団多様性を表す状態が適切に細かく定義されていれば，多様性が大きく悪化する前に他の探索オペレータの選択確率を高く設定できるが，そうでない場合は多点探索の利点を生かせない可能性が高い．

これに対して，1 世代での集団による探索結果から，集団全体での適応値改善率を報酬とした場合，探索オペレータ B により多様性が維持された効果による報酬も探索オペレータ B に割り振られるため，探索オペレータ A だけでなくオペレータ B の選択確率も高くなる．これにより，多点探索の特性を生かしたより効率的なパラメータ制御方策が獲得出来る可能性がある．

問題点としては，実際は貢献度の低かった探索オペレータにも報酬が割り振られる事になる可能性がある．これは同条件で複数回の学習が行われるうちに収束していくが, 直接的に評価出来ないためにより多くのトレーニング時間を要する可能性がある．この影響を出来るだけ小さくするためには，集団探索での報酬だけでなく個体生成毎の直接的な報酬との重み加算和を用いることにより，トレードオフの調整が可能になると考えられる．

4. おわりに

遺伝的アルゴリズムなど進化的アルゴリズムを利用した探索手法の効率化のため, パラメータの設定を制御する学習型アルゴリズムを提案した．

on-Policy 型の強化学習手法を利用した方法 [12]に対し，オペレータの計算コストや GA の多点探索特性なども利用して更に効率改善を図ったものである．すなわち，各パラメータ設定での評価結果だけでなく，探索負荷つまりオペレータ 1 回適用毎の計算時間などや GA の多点探索特性も利用しながら，効率的にパラメータ設定を行う強化学習型の進化的アルゴリズム手法を提案した．今後は，進化的アルゴリズムのパラメータ制御のための学習方式、特に強化学習における報酬の与え方の改善方法について更に研究してゆく．

謝辞 本研究は柏森情報科学振興財団の助成を受けて遂行された．本研究は，東京電機大学総合研究所研究 02Q823 として行ったものである．

参考文献

1) L. Davis, editor,: "Handbook of Genetic Algorithms", Van Nostrand Reinhold, N.Y., (1991).
2) D. Applegate, R. Bixby, V. Chvatal, and W. Cook, The Traveling Salesman Problem: A Computational Study (Princeton Series in Applied Mathematics): Princeton University Press, 2007.
3) 小野山隆，前川拓也，久保田仙，鶴田節夫，薦田憲久：“マルチステージ GA による共同物流網における配送計画作成手法”，電気学会論文誌 C, Vol.127, No.9, pp 1460-1467, 2007.
4) 櫻井義尚，小野山隆，久保田仙，中村嘉宏，鶴田節夫 :"制約付き TSP を解くための局所利己的遺伝子許容動的制御 GA"，情報処理学会論文誌 数理モデル化と応用, Vol.48 , No.SIG19(TOM19), pp.127-138, 2007.
5) F. G. Lobo, C. F. Lima, and Z. Michalewicz, Parameter Setting in Evolutionary Algorithms: Springer Publishing Company, Incorporated, 2007.
6) Y. Sakurai, N. Honda, J. Nishino: "Acquisition of Knowledge for Gymnastic Bar Action by Active Learning Method", Journal of Advanced Computational Intelligence & Intelligent Informatics (JACIII), Vol.7, No.1, pp.10-18 (2003)
7) A. Eiben, Z. Michalewicz, M. Schoenauer, and J. Smith, "Parameter Control in Evolutionary Algorithms," in Parameter Setting in Evolutionary Algorithms, 2007, pp. 19-46.
8) V. Nannen and A. E. Eiben, "A method for parameter calibration and relevance estimation in evolutionary algorithms," in Proceedings of the 8th annual conference on Genetic and evolutionary computation Seattle, Washington, USA: ACM, 2006.
9) L. DaCosta, A. Fialho, M. Schoenauer, Mich, and l. Sebag, "Adaptive operator selection with dynamic multi-armed bandits," in Proceedings of the 10th annual conference on Genetic and evolutionary computation Atlanta, GA, USA: ACM, 2008.
10) D. Thierens, "Adaptive Strategies for Operator Allocation," in Parameter Setting in Evolutionary Algorithms, 2007, pp. 77-90.
11) J. E. Pettinger and R. M. Everson, "Controlling Genetic Algorithms With Reinforcement Learning," in Proceedings of the Genetic and Evolutionary Computation Conference: Morgan Kaufmann Publishers Inc., 2002.
12) Fei. Chen, Yang. Gao, Zhao-qian. Chen, and Shi-fu. Chen, "SCGA: Controlling Genetic Algorithms with Sarsa(0)," in Computational Intelligence for Modelling, Control and Automation, 2005 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, International Conference on, 2005, pp. 1177-1183.
13) 山本芳嗣, 久保 幹雄: 巡回セールスマン問題への招待, 朝倉書店, (1997).
14) 永田裕一, 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, 14, [5], pp.848-859, (1999).
15) R. Sutton and A. Barto, Reinforcement Learning: An Introduction (Adaptive Computation and Machine Learning): Mit Pr, 1998.
16) Watkins C J C H and Dayan P., “Technical note: Qlearning,” Machine Learning, Vol. 8(3-4), pp. 279-292, 1992.