

進化型移動ロボットのゴール到達問題における

逐次近似最適化手法の検証

(サンプル点の追加戦略による解探索性能への影響)

○大島 直人 片田 喜章 (摂南大学)

Application of the sequential approximate optimization method to an evolutionary robotics problem --

Investigation of the effect of sampling method on the performance of the best evolved robot

*Naoto Ohshima, Yoshiaki Katada (Setsunan University)

Abstract—Due to the parallel population search of evolutionary robotics, it takes a long time to evaluate all robot in a real environment. Thus, such techniques as to shorten the time are required. One of them would be the sequential approximate optimization (SAO) method in which the fitness function is estimated by an approximate model during an evolutionary optimization process and samples are taken sequentially after every optimization process. This study investigates the performance of sampling methods of the SAO by varying the number of samples on a robot control problem using simulated robots.

Key Words : Sequential Approximate Optimization, Genetic Algorithm, RBF Networks

1 はじめに

高い解探索能力が期待できる人工進化を用いて自律ロボットの制御器を構築する進化ロボティクスがある1)。進化ロボティクスでは制御器を構築する際、制御器の評価方法として実環境で評価する方法とシミュレーションを用いて評価する方法がある。これらの方法にはそれぞれ問題点が指摘されている。実環境で評価する場合には、ロボットを評価するために実時間がかかる。人工進化に要する進化時間は数百世代から数千世代が一般的であり、膨大な実験時間が必要となる。一方、シミュレーションによって評価する場合には、シミュレーションで獲得した制御器を実ロボットに搭載した場合、シミュレーションと実環境とのギャップによってロボットが必ずし

も実環境で適切に振舞えるとは限らない。つまり、シミュレーションで用いることのできる制御問題は、その環境およびロボットとの相互作用が適切にモデル化できる問題に限られる。これに対し、我々はこれまでに遺伝的アルゴリズム(Genetic Algorithm: GA)の遺伝長が短くなるように制御器となるニューラルネットワーク(Artificial Neural Networks: ANN)をコーディングすることで短い進化時間で自律移動ロボットの制御器を構築する手法を提案してきた2)。そして、計算機実験および実機実験によってその有効性を確認してきた。

一方、自律移動ロボットの制御器をGAとANNを用いて構築する際に、実環境またはシミュレーションで評価したロボットの制御器となるANNの設

計変数, およびその評価値から RBF ネットワークを用いて作成する近似評価関数を用いて GA により ANN の最適化を行う. これにより, ロボットの評価回数を削減する手法を提案している 3). 北山らはこの手法を統合的最適化システム 4)と呼んでおり, 一般的には逐次近似最適化手法 5)と呼ばれている. しかしながら, 進化ロボティクスにおいて逐次近似最適化手法が用いられた事例は少なく, この文献 3)で用いられている手法がより一般的な進化ロボティクスにおける制御問題に対してどのような性能を示すのかは知られていない. 特に, 文献 3)では北山らによって提案されている逐次近似最適化手法のサンプル点追加戦略 4)は考慮されていない.

以上より, 本稿では進化ロボティクスにおけるロボット制御問題の一つであるゴール到達問題に対し逐次近似最適化手法を適用し, サンプル点追加戦略が解の探索性能に与える影響を計算機実験により検証する.

以下, 第 2 節では応答曲面を作成するために用いる RBF ネットワークについて述べる. 第 3 節では逐次近似最適化手法によるロボットの制御器の構築について述べる. 第 4 節では計算機を用いた検証実験を行う. 第 5 節では本稿のまとめを述べる.

2 RBF ネットワーク

RBF ネットワークは 3 層の階層型ネットワークである 4). 入力変数ベクトル $\mathbf{x} = (x_1, x_2, \dots, x_p)^T$ に対するネットワークの出力 $O(\mathbf{x})$ は (1) 式で計算される.

$$O(\mathbf{x}) = \sum_{j=1}^m w_j h_j(\mathbf{x}) \quad (1)$$

ここで, m は隠れ層の素子数, $h_j(\mathbf{x})$ は隠れ層の j 番目の基底関数の出力, w_j は隠れ層の基底関数と出力層の素子間の結合荷重である. また, 隠れ層の基底関数は, (2) 式で表されるガウス関数である.

$$h_j(\mathbf{x}) = \exp\left(-\frac{(\mathbf{x} - \mathbf{c}_j)^T(\mathbf{x} - \mathbf{c}_j)}{r_j^2}\right) \quad (2)$$

ここで, \mathbf{c}_j は j 番目の基底関数の中心, r_j は j 番目の基底関数の半径である. また, 本研究における基底関

数の半径 r_j は文献 5) で提案されている以下の (3) 式を用いて計算する.

$$r_j = \frac{d_{j,\max}}{\sqrt{n}\sqrt{m-1}} \quad (3)$$

ここで, $d_{j,\max}$ は j 番目のサンプル点からの最大距離を表している. n は入力ベクトルの次元数であり, m は基底関数の数と同一とする 4).

3 逐次近似最適化手法による自律移動ロボットの制御器の構築

ここでは, ロボットの制御器である ANN の設計に関して本研究で適用する逐次近似最適化手法を用いた方法について述べる.

3.1 逐次近似最適化手法

文献 4) をもとに, RBF ネットワークにより評価関数空間を近似する応答曲面を作成する. 作成した応答曲面を用いてロボットの制御器である ANN の結合荷重値の最適化を行う. その手順を以下に示す.

- (1) ANN の結合荷重値を設計変数として初期サンプル点を M_{ini} 個生成する.
- (2) 計算機実験により初期サンプル点である ANN を搭載したロボットの評価値を求める.
- (3) RBF ネットワークを用いてサンプル点とその評価値から評価関数空間を近似する応答曲面を作成する.
- (4) 最適化手法により作成した応答曲面に対して最適化を行う.
- (5) サンプル点追加戦略に従い, サンプル点を M 個追加する.
- (6) 計算機実験により, 手順 (4) で追加したサンプル点である ANN を搭載したロボットの評価値を求める.
- (7) 追加サンプル点数が最大サンプル点数 M_{\max} になるまで, 手順 (3) から手順 (6) を繰り返す.

3.2 最適化手法

3.1 の手順 (4) における最適化手法には ANN の設計に対しその有効性が確認されている拡張型遺伝的アルゴリズムである Operon GA 6) (以下, OGA) を用い

る. Operon GAはバイナリコーディングを用いたGAであり, バイナリストリングを実数値に変換することでANNの設計に用いる.

ここで, 3.1で述べた評価関数空間を近似する応答曲面を近似適応度関数と呼び, 以下にその最適化の手順を示す.

- (1) 初期個体群を N_{ini} 個とする. この初期個体群は3.1における初期サンプル点と同一とする.
- (2) 近似適応度関数を用いて各個体の近似適応度を計算する.
- (3) 近似適応度に基づき OGA の遺伝的操作を実行し, 次世代の個体群を N 個生成する.
- (4) 規定世代数 G_{max} まで(2), (3)を繰り返す.

3.3 サンプル点の追加戦略

サンプル点追加手法としてつぎの三つが考えられる.

- (a) 追加サンプル点をランダムに追加する.
- (b) 応答曲面の最適解近傍とサンプル点の疎な領域にサンプル点を追加する.
- (c) 手法(b)において設計変数空間をOGAの遺伝子空間で近傍を定義し, サンプル点を追加する.

手法(a)は基底関数空間に一様乱数でランダムにサンプル点を追加することで最適化手法によって大域的最適解を得ることを期待している.

手法(b)は文献4)に従い, 3.2より最適化された応答曲面最適解 \hat{x} の近傍を作成し, その近傍に3.1で述べたサンプル点を $M/2$ 個ランダムに追加する. また, サンプル点の疎な領域に対してもサンプル点を $M/2$ 個ランダムに追加する. ここで, 近傍の大きさ l_i は次式により決定する(文献4)).

$$l_i = \frac{1}{count + 1} (x_i^U - x_i^L), (i = 1, 2, \dots, n) \quad (4)$$

(4)式中の x_i^U は設計変数の上限値, x_i^L は下限値をとし, n は設計変数の次元数を表す. また, $count$ はパラメータであり, 1以上の整数である.

手法(c)は手法(b)のサンプル点の追加方法と比較するためのものである. 追加手法(c)と同様に手法(b)の最適解 \hat{x} の近傍とサンプル点が疎な領域にそれぞれ

$M/2$ 個サンプル点を追加する. 但し, 手法(b)は設計変数空間で近傍を定義しているのに対し, 手法(c)ではOGAの遺伝子空間で近傍を定義する. ここでは最適解 \hat{x} の遺伝子型に対し, ハミング距離を1とする遺伝子を近傍とサンプル点が疎な領域にそれぞれ $M/2$ 個ランダムに生成する.

4 計算機実験

4.1 実験設定

4.1.1 制御器の設定

移動ロボットはセンサを7個搭載し, 移動機構は左右二輪独立駆動型と仮定する. この移動ロボットの制御器であるANNは入力層ニューロン数 N_s を7, 隠れ層ニューロン数 N_h を1, 出力層ニューロン数 N_o を2と設定する. 隠れ層と出力層のニューロンは完全相互結合とする. また, ANNの結合荷重値の上限値を $x_i^U = 1$ とし, 下限値を $x_i^L = -1$ とする.

4.1.2 ゴール到達問題

計算機実験に使用するゴール到達問題7)をFig.1に示す. 移動ロボットはより早くゴールに到達することを目的とする. 移動ロボットの初期位置をFig.1の環境における左下, ゴールの位置を右上とする. 試行はロボットがゴールに到達するか, あるいは到達せずに200ステップ経過したときに終了する. ロボットの初期姿勢を 0° から 45° ずつ変えた試行を3回行う. 以下の(5)式を用いて個体の適応度を計算する.

$$Fitness = \frac{1}{NumTrials} \sum_{k=1}^{NumTrials} \left(1 - \frac{Step_k}{MaxStep} \right) \quad (5)$$

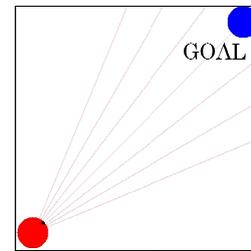


Fig.1 Experimental setup for a simulated environment

ここで, $NumTrials$ は試行回数(3)であり, $MaxStep$ は200となる.

4.1.3 OGA と逐次近似最適化手法の設定

本研究における OGA は $N_{ini} = 50$ とし, $G_{max} = 500$ とする. また, 3.1 で述べた逐次近似最適化手法の設定として, $M_{ini} = 50$ とする. 追加するサンプル点数 $M = \{10, 30, 50\}$ と変化させて実験を行う.

$M = 10, 50$ の場合では最大サンプル点数 $M_{max} = 300$, $M = 30$ の場合では $M_{max} = 290$ として, 3.2 で述

べた手法(a)(b)(c)に対して乱数の種を変えた試行を 50 回試行する.

ここで, 解探索性能を比較するために, 本タスクでゴールに到達できる制御器を持つロボットを単に解と呼ぶ. また, ANN の結合荷重値の上下限值の設定に従い, $x_i^U = 1, x_i^L = -1$ ((4)式)とする.

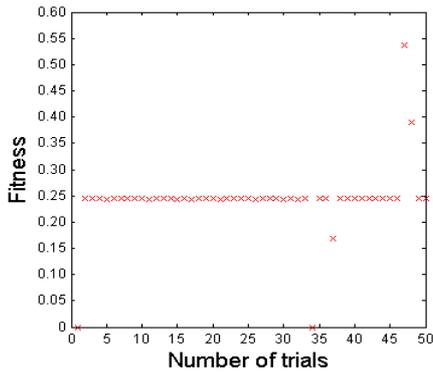


Fig.2 Sampling method (a)
(add 10 points each samples)

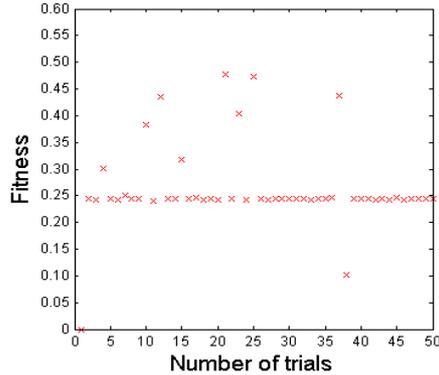


Fig.5 Sampling method (b)
(add 10 points each samples)

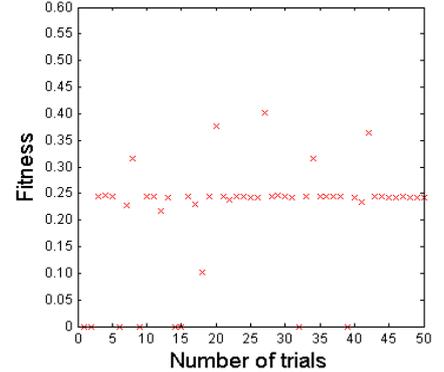


Fig.8 Sampling method (c)
(add 10 points each samples)

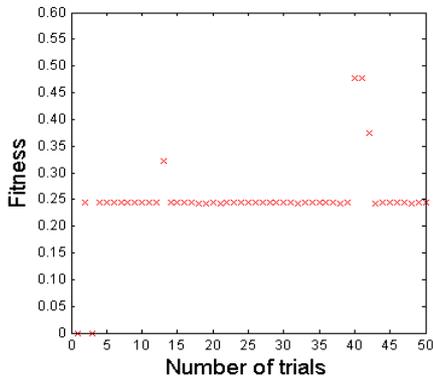


Fig.3 Sampling method (a)
(add 30 points each samples)

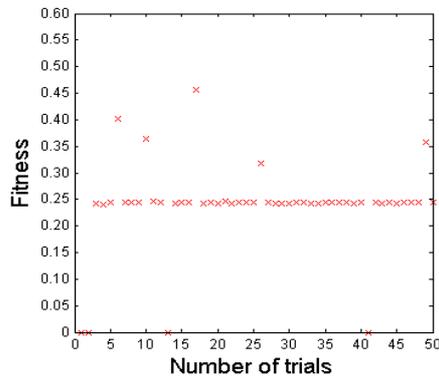


Fig.6 Sampling method (b)
(add 30 points each samples)

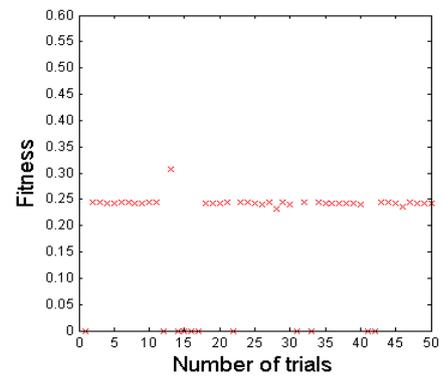


Fig.9 Sampling method (c)
(add 30 points each samples)

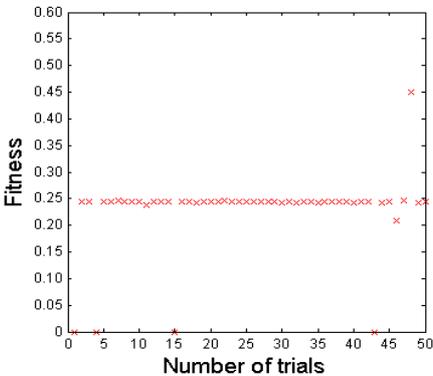


Fig.4 Sampling method (a)
(add 50 points each samples)

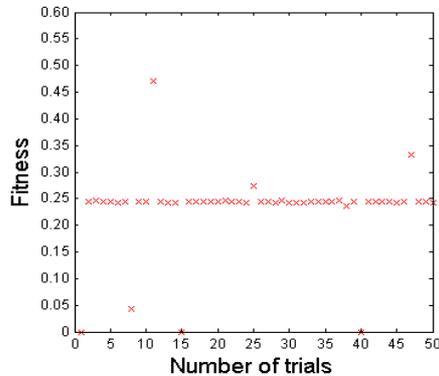


Fig.7 Sampling method (b)
(add 50 points each samples)

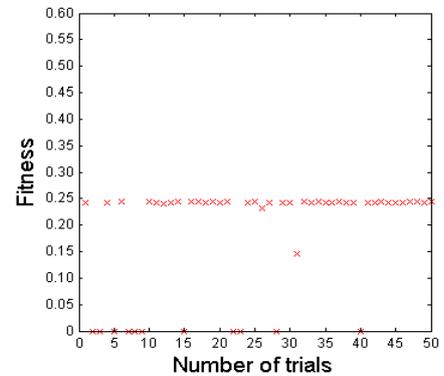


Fig.10 Sampling method (c)
(add 50 points each samples)

Table.1 Number of successful trials to reach the goal for each method

Method M	(a)	(b)	(c)
10	2	8	5
30	4	4	1
50	1	4	0

4.2 実験結果

Fig.2-10に各実験の各試行に対してOGAの最終世代において制御器が獲得した真の最大適応度を示す。まず、手法(a)に対して各 M の様子を比較する。Fig.2-4よりのサンプル点数が $M = 30$ の場合(Fig.3)に $M = 10, 50$ の実験より高い適応度を持つ解が少数ではあるが多く得られていることが確認できる。つぎに手法(b)に対して各 M の様子を比較する(Fig.5-7)。 $M = 10$ の場合(Fig.5)に $M = 30, 50$ のより高い適応度を持つ解が多く得られていることが確認できる。つぎに手法(c)に対して各 M の様子を比較する(Fig.8-10)。手法(b)と同様に $M = 10$ の場合(Fig.8)に $M = 30, 50$ のより高い適応度を持つ解が多く得られていることが確認できる。

つぎに Table.1 に手法(a),(b)および手法(c)における各実験に対する解のうち、4.1.2 で述べたゴールに到達する回数が2回以上の解の数を表す。まず、手法(a)と手法(b)の方が試行数に対してゴールに2回以上到達する解を多く獲得していることが確認できる。また、手法(c)が手法(b)よりも試行回数に対してゴールに2回以上到達する解の獲得数が減少していることが確認できる。

以上より、手法(a)よりも手法(b)を用いる場合に適応度の高い解を得られることが確認できる。また、手法(b)において追加するサンプル点数を少なく設定することで高い適応度を持つ解が多く得られることが確認できる(Fig.7)。これは、最大サンプル点数に上限があり追加サンプル点数を少なく設定すると逐次的

な更新回数が増す。そのため、応答曲面の最適解の精度が逐次的に向上することで高い適応度を持つ解が多く得られるものと考えられる。つぎに手法(c)では手法(b)の応答曲面の最適解近傍とサンプル点が疎な領域にサンプル点を追加する場合、3.3で述べた遺伝子空間で最適化の遺伝子が増えることでサンプル点を生成するため、設計変数空間では大幅にサンプル点の配置が変化すると考えられる。このため、遺伝子空間でサンプル点を追加することは設計変数空間でサンプル点をランダムに追加することに等しくなり、近適応度の高い解を得ることが困難になると考える。

また、Fig2-10より手法(a),(b)および(c)によって獲得される真の最大適応度は0.25程度であり、構築した制御器のほとんどが低い適応度を示している。これは、設計変数空間において2の(2)式で表される各基底関数間の距離が遠く配置されているため、応答曲面の形状が平坦になると推測される。そのため、応答曲面を用いて最適化される解が低い適応度を示すと考えられる。また、RBFネットワークを用いて応答曲面を作成する場合、最大サンプル点数が少ないことにより近似した応答曲面の最適解の精度が向上せず、解の探索性能が低くなると考える。

5 まとめ

本研究では、RBFネットワークによる逐次近似最適化手法を用いて自律移動ロボットの制御器を設計し、サンプル点追加戦略が解探索性能にどのような影響を与えるのかを計算機実験により検証した。その結果、応答曲面の最適解近傍とサンプル点が疎な領域にサンプル点を追加する場合に適応度の高い制御器を得られることを確認した。これは、文献4)におけるサンプル点追加戦略を用いた結果の傾向と同一である。また、追加するサンプル点数を少なく設定することで応答曲面の更新回数が増し、高い適応度を持つ制御器が多く得られることを確認した。

今後、今回の実験に用いたゴール到達問題の評価関数を再度検討し、検証実験を行っていきたい。また、

追加サンプル点数をさらに増やす場合や最大サンプル点数の上限を増やす場合でも同様に実験を行っていきたい。

参考文献

- 1) S. Nolfi and D. Floreano: *Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines*, MIT Press (2000)
- 2) 片田, 日高 : 簡易コーディングを用いた進化型人工神経回路網の移動ロボット制御問題における性能検証, システム制御情報学会論文誌, Vol.0, No.0, pp.149-154(2007)
- 3) 原, 元木, 山崎, 安田 : RBF ネットワークと遺伝アルゴリズムによる自律移動ロボットコントローラ構築, 電気学会論文誌 C(電子・情報・システム部門誌), Vol.131, No.5, pp.1000 - 1008(2011)
- 4) 北山, 安田, 山崎 : RBF ネットワークと Particle Swarm Optimization による統合的最適化, 電気学会論文誌 C , Vol.128, No.4, pp. 636-645 (2008)
- 5) 北山, 酒井, 荒川, 山崎 : RBF ネットワークによる逐次近似最適化(離散変数への適用), 日本機械学会論文集(C編), 78 巻, 787 号 pp.177-190(2012)
- 6) 片田, 大倉 : ニューラルネットワークを含む適応度景観における遺伝的アルゴリズムの進化ダイナミクス - 進化ロボティクスでの検証, システム制御情報学会論文誌, Vol.21, No.2, pp.31 - 39(2008)