

行為に基づく環境モデリングのための 移動ロボットの進化的行動獲得

Evolutionary Learning of Behaviors for Action-Based Environment Modeling

山田 誠二*
Seiji Yamada

* 東京工業大学大学院総合理工学研究科知能システム科学専攻
CISS, IGSSE, Tokyo Institute of Technology, Yokohama, 226-8502 Japan

19YY年MM月DD日 受理

Keywords: Action-based Environment Modeling, a mobile robot, behavior learning, GA

Summary

This paper describes an evolutionary way to design behaviors of a mobile robot for recognizing environments. We have proposed AEM approach for a mobile robot to recognize environments. In AEM, a mobile robot acts in the environments and the action sequences are obtained. The action sequences are transformed into environment vectors which characterize the environments, and the robot identifies the environments with similarity between environment vectors. Suitable behaviors for AEM have been described by a human designer. However the design is very difficult for them because of the huge search space. Thus we propose an evolutionary design method of such behaviors using genetic algorithm and make experiments in which a robot recognizes the environments with different structures. As results, we find out that our approach is promising to automatically acquire behaviors for AEM.

1. はじめに

本研究では、行為に基づく環境モデリング AEM (Action-based Environment Modeling) に適した移動ロボットの行動を、遺伝的アルゴリズム GA (Genetic Algorithm) を用いて、進化的に獲得することを目指す。AEM とは、従来のように移動ロボットのセンサで観測された情報を統合して環境の全体的な精度の高い幾何学的地図をつくるのではなく、その環境でロボットが一連の行動を実行して、その結果得られる行為系列 (action sequence) に基づいて環境をモデリングし、環境認識をすることである。ここで、行動 (behavior) とは、“IF 状態 THEN 行為” というルールであり、本研究において移動ロボットは、行動の集合を与えられ、センサ値を状態に変換し、その状態を IF 部としてもつ行動の THEN 部の行為を実行するという行動ベースに

より動作する。また、状態 (state) とは、環境のセンサ情報を離散化したものであり、行為 (action) とは、ロボットのアクチュエータによる環境への操作であり、具体的には移動ロボットのモータコマンドである。

例えば、5.1 節の図 6 ~ 図 10 は、形の違う環境でのロボットの移動軌跡を示している。この移動ロボットは与えられた行動に従って移動を行っているが、その結果それぞれの環境に依存して異なった行為の系列が得られる。AEM は、この行為系列を基に環境を識別する。ただし、AEM に適した行動は識別すべき環境に依存するので、最初から考えられるすべての環境に対して有効な行動を与えることは難しく、またそのような行動を人間が試行錯誤的に設計することも非常に困難である。よって、本研究では、AEM に適した行動を GA により自動獲得させることを目指す。

なお、探索手法として、GA を用いたのは、評価関数についての仮定、探索ヒューリスティックス (パイ

アス)などの知識がなくても良好な結果が期待できるという理由による。

遺伝的プログラミング GP (Genetic Programming) [Koza 92] による移動ロボットの行動学習は、いくつかの先行研究 [Reynolds 94, Koza 91, Ito 96, Nordin 96] がある。しかし、それらは、障害物回避等のように行動毎に独立に評価が行える単純な領域を扱っている。それに対して本研究では、それぞれの環境において得られた行為系列を用いて環境認識ができるような行動を GA で獲得する。よって、行動は単独では評価できず、各環境で実行された行為系列間の関係により評価されるため、より問題が複雑になっている。

2. 行為に基づく環境モデリング

本研究では、「ロボットが現在いる環境を、過去に訪れたことのある環境と同定すること」を環境認識と呼ぶ。赤外線近接センサしか持たず、精度の悪いデッドレコニング (dead reckoning) *1 しかできない低機能な移動ロボットに環境認識をさせることを考えよう。このような単純な移動ロボットを用いて環境認識を実現することは、実装のコスト面、あるいは、大域的な情報が得られにくい環境において価値がある。このとき、従来のように局所的なセンサ情報を統合して、絶対座標のどの位置に障害物があるかを示す幾何学地図をつくる方法 [Clowly 85] は実現困難である。なぜなら部分的情報の統合に必要な絶対座標が大きな累積誤差を含んでしまい、それからつくられた幾何学地図は、現実と著しく異なったものとなるからである。

よって、より頑健で環境認識に必要な情報を含む環境のモデリングが必要になる。そこで我々は、環境の構造を反映するように行動を設計し、環境でロボットが実際に行動することで得られる行為系列から環境の構造を抽出して環境をモデリングする方法を提案してきた [Yamada 98b]。行動を設計する際に、幅をもったセンサ情報に対して行為を割り当てておけば、ノイズに対して頑健である抽象化された環境の記述として行為系列を利用できる。また、同様の発想に基づいたいくつかの手法 [Nehmzow 91, Mataric 92, 加藤 96] がある。我々は、このような手法を行為に基づく環境モデリング AEM と呼ぶ。AEM には、構造化されている環境を扱うもの [Mataric 92, Yamada 98b] から構造を必要としないもの [加藤 96] まで、また視覚等の高度な情報も扱うもの [Mataric 92] から近接センサだけを

*1 移動距離計、モータの回転計などの内界センサだけを用いて、現在位置を決定すること。

用いるもの [Nehmzow 91, 加藤 96, Yamada 98b] までの様々な実現方法がある。ここでは、実機での実装の容易性と、得られた環境モデルから実際の環境認識までを扱っている点を考慮して、我々の手法 [Yamada 98b] を本研究で用いる AEM として採用する。

また、AEM のように行為系列だけから環境モデリングをするのではなく、行動により得られたセンサ情報の系列を基に環境モデリングを行う研究 [Mataric 92, Nakamura 96] もある。これらの研究は、センサ情報の系列に基づいてはいるが、センサ情報と行為が強く関連付けられている場合は、AEM と本質的に等価なものと考えられる。

2.1 AEM 手続きの概要

我々の研究 [Yamada 98b] に基づいて、本研究で対象とする AEM を説明する。ただし、本研究で対象となる環境は、多角形の閉領域である。AEM の処理の流れを図 1 に示す。学習方法として、1-Nearest Neighbor 法 [Dasarathy 91] を用いる。この方法は、どのクラスに属するかがわかっている既与の訓練例を事例として蓄えておき、与えられた試験例に対し事例との距離を計算して、それが最小である事例のクラスにその試験例も属すると判定する。

処理は、訓練フェーズ (図 1(a)) と試験フェーズ (図 1(b)) にわかれる。まず、訓練フェーズでは、ロボットが、与えられた行動をもとに実際に識別すべき複数の環境 (訓練環境) で行為を行う。そして、それぞれの訓練環境で得られた行為系列を環境ベクトルに変換した後に事例として蓄える。次の試験フェーズでは、判定すべき環境 (試験環境) においてロボットが行動し、そこで得られた環境ベクトルとのユークリッド距離最小の事例のクラスにより環境認識を行う。

2.2 AEM の問題点

AEM では、行為系列が環境の構造を反映するような適切な行動を用意する必要がある。しかし、そのような行動をいかに設計するかという問題がある。これに対し、従来は環境に応じて適切な行動を設計者が設計していた。壁沿い移動 (wall-following) [Mataric 92, Yamada 98b, Nehmzow 91] や、ランダムウォーク [加藤 96, Nakamura 96] がよく用いられるが、これらの行動は、環境の輪郭の形状や面積などの構造が、環境認識に利用できることを前提にしている。よって、そのような前提が成り立たない場合、さらには、環境の構造のどの部分に注目して認識すればよいのか設計者にもわからないような場合、AEM にとって適切な

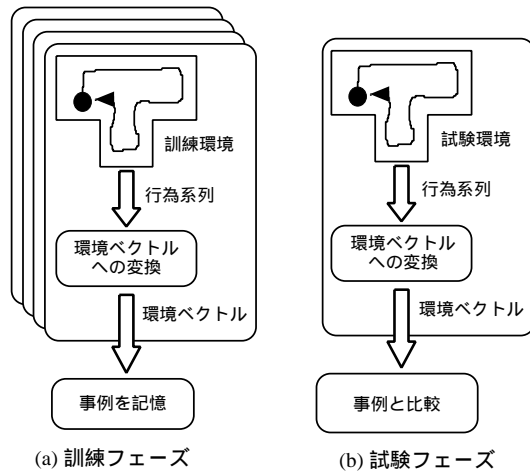


図1 AEMの概要

行動を設計することは非常に困難なものとなる。例えば、輪郭の異なる環境を識別するためには、壁沿い移動が適切と考えられるが、さらに輪郭は同じだが、窓の有無や障害物の有無等により環境を識別する場合は、はるかに複雑な行動が必要になる。また、一般に適切な行動の候補は膨大にあるため、人間の設計者が思考錯誤的に探索することは現実的でない。よって、認識すべき環境の構造に応じて、AEMにとって適切な行動を自動的に決定するメカニズムが必要となる。

3. 環境認識のための行動獲得

本研究では、計算機上のシミュレーションで行動を評価して、遺伝的アルゴリズムにより、既与の環境をAEMを用いて識別するために適切な行動を進化的に獲得することを目指す。なお、本論文は、[Yamada 98a]に更なる実験結果と考察を加えたものである。

以降では、状態と行為の記述を行っていくが、探索空間を狭める意味で、状態記述及び行為記述ともできるだけ少なくすることが望ましい。

3.1 小型移動ロボット Khepera のモデル

KheperaTM は、人工生命、人工知能研究で広く使われている小型移動ロボット（直径 55mm、高さ 30mm）であり、本研究でも、最終的に学習された行動を実装する実ロボットとして Khepera を用いることにする。Khepera の概観図とセンサ位置を図 2、図 3 に示す。Khepera は、マイクロプロセッサ 68331（16MHz）を搭載し、赤外線近接センサと光センサが一体になったセンサを 8 個と駆動用に 2 個の DC モータをもつ。

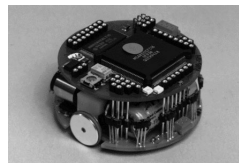


図2 Khepera 概観

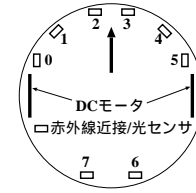


図3 センサ位置

3.2 状態の記述

Khepera のような単純な移動ロボットでは、累積誤差のためにデッドレコニングで自分の絶対座標を求めることは非常に困難である。よって、ロボットにとっての環境の状態を、現在のセンサ値の範囲で記述する。

〔1〕距離による状態

Khepera のもつ 8 つのセンサは、光の強度、障害部との距離に関連してそれぞれ 10bit (= 1024) の離散値を返す。しかし、距離では、センサから 50mm 付近でクリスプな反応を示すので、近接センサの値は、50mm 以内に物体が存在するか否かの 2 値とする。

さらに、状態数を少なくするために、前方の 2 つの近接センサ（図 3 の 2, 3）は、2 つのセンサ値の OR をとることにより一つにまとめる。また、左右斜め前方の 2 つのセンサ（図 3 の 1 と 4）は、左右と前方のセンサで捕捉できるので、これらは使わない。さらに、後述のように移動ロボットは後進しないとするため、後方の近接センサも使わない。よって、考慮すべき赤外線近接センサの数は、前方、左右方向の 3 つとなり、各センサが 2 値をとることから、近接センサの全状態数は、 $2^3 = 8$ となる。

〔2〕光による状態

次に、光センサの状態を考える。本研究では、光源に接近し、そして近付き過ぎると離れる、あるいは、最初から離れる等の単純な行為を考えているため、移動ロボットに対する光源の相対的な方向と、2 値程度の光の強度を用意する。ただし、ここでも出来る限り状態数を少なくするため、図 3 のセンサ 0, 2, 5, 6 だけを用いて、最も光の強いセンサ位置とそのセンサ値をしきい値 100 で 2 値にしたもので、光による状態を記述する。ただし、ほとんど光の強さに偏りのない状況で光の強さ最大方向を調べても意味がないので、センサ値の最大と最小の差が 100 以下の場合は、ひとつの状態とする。よって、光による状態は、光強度最大のセンサ位置が 4 通り、その値が 2 通り、そして、光の強度に偏りが無い状態が 1 通りとなり、考えられる状

状態数は、 $4 \times 2 + 1 = 9$ となる。

以上の距離による状態と光による状態を組み合わせた総状態数は、 $8 \times 9 = 72$ となる。

3・3 行為の記述

行為は、我々の AEM 研究 [Yamada 98b] を参考に、以下に示すような 4 つの行為を用意する。

- A1：直進、5mm 進む。
- A2：進行方向に対し、 30° 左回転する。
- A3：進行方向に対し、 30° 右回転する。
- A4：進行方向に対し、 180° 左回転する。

これら 4 つの行為が実行されて、移動ロボットは、環境内を徘徊する。その結果、実行された行為系列が得られる。

3・4 行為系列のベクトルへの変換

得られた行為系列は、次に示す手続きにより、環境の構造的特徴を表す環境ベクトルに変換される。今、行為系列を $[a_1, a_2, \dots, a_n]$ ($a_i \in \{A1, A2, A3, A4\}$)、環境ベクトルを $V = (v_1, v_2, \dots, v_m)$ ($n \leq m$) とすると、 V のベクトル値が v_1 から v_n の順に、以下に示す BI 変換 [Yamada 98b] により決定される。ただし、 $v_0 = 0$ とする。この v_0 は、 v_1 を計算するときだけに使われ、出力の環境ベクトル中には現れない。なお、この BI 変換は、チェインコーディング [Arkin 91] の一種である。

- (1) $a_i = A1$ なら、 $v_i = v_{i-1}$ 。
- (2) $a_i = A2$ なら、 $v_i = v_{i-1} + 1$ 。
- (3) $a_i = A3$ なら、 $v_i = v_{i-1} - 1$ 。
- (4) $a_i = A4$ なら、 $v_i = -v_{i-1}$ 。

4. 遺伝的アルゴリズムの適用

前節で設定された状態数は 72 であり、行為数は 4 である。ここで、行動とは、“IF 状態 THEN 行為” というルールなので、考えられる行動数は、 4^{72} ($\approx 2.2 \times 10^{43}$) となる。この膨大な組合せの中から、AEM に適した行動を探索しなければならない。本研究では、進化的探索手法である GA を用いて行動の探索を行う。

遺伝的アルゴリズム GA [Holland 75, 伊庭 94] は、問題の解候補を染色体としてコーディングし、それらを適合度関数により評価して、個体間の交叉、突然変異などのオペレーションを世代毎に適用することにより、進化的探索をおこなう。

4・1 GA の手続き

以下に、本研究で用いる GA の手続きを示す。

- Step1 初期集団の生成：個体数 N の個体をランダムに生成し、初期個体集団 I_1, \dots, I_N とする。
- Step2 適合度の計算：個体集団の各個体 I_1, \dots, I_N について、適合度 f_1, \dots, f_N を計算する。
- Step3 終了条件が満たされれば、終了（なお、具体的な終了条件は、5 章で示される。）
- Step4 選択： f_1, \dots, f_N に基づき個体集団から、子の候補集団 C を選択する。
- Step5 交叉： C から交叉率 P_{mut} の確率でペアをランダムに選択する。そして、それぞれのペアに対し交叉を適用し、その結果生成されたの 2 つの子を先のペアと置き換えて、集団 C を更新する。
- Step6 突然変異：突然変異率 P_{mut} を突然変異の生起確率として、集団 C の遺伝子をランダムに変更することにより、個体集団を更新する。そして、 C で、全体の集団を更新する。
- Step7 Step2 へ。
- 選択の方法、交叉オペレータ、交叉率、突然変異率、さらに染色体のコード化について、以降述べていく。

4・2 各種手続き、パラメータの設定

経験的に適切と考えられている各種手続き、各パラメータを以下のように設定した。

- 個体数：50 に設定。
- 選択方法：エリート戦略とトーナメント方式 [伊庭 94] を用いる。適合度が最大であるエリート個体一つをそのまま残す。また、エリート以外の個体から、トーナメントサイズ = 2 のトーナメント方式を用いて、集団 C を構成する。
- 交叉オペレータ：一様交叉 [伊庭 94] を用いる。
- 交叉率 P_{mut} ：0.8 に設定。
- 突然変異率 P_{mut} ：0.05 に設定。

4・3 行動のコード化

本研究では、同じ状態に対して複数の行為を割り当てるような確率的な行為選択は行わないため、問題の解候補、つまり行動は、一つの状態から一つの行為への 1 対 1 写像である。よって、図 4 のように、72 の全状態状態 $\{s_1, s_2, \dots, s_{72}\}$ に対し、1 つの行為 $A_n \in \{A1, \dots, A4\}$ を割り当てるコード化を考える。図 4 の 72 文字からなる文字列が GA での染色体（遺伝子型）になる。

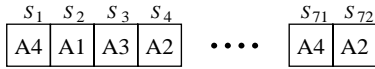


図4 行動のコード化

4・4 適合度関数の設定

適合度関数は、染色体を評価する重要な基準である。本研究では、“AEM に適した行動の獲得”を目標としているため、当然適合度は、その目的に沿ったものでなければならない。本節では、そのような適合度関数の設定を行う。

まず、AEM の行動に求められる条件として、以下のものが考えられる。

- 行動の停止性：環境においてロボットが停止しないと、行為系列を得ることはできないため、停止性は AEM にとって最も重要な条件である。また、一定の移動距離あるいは移動時間で行動を停止するようなことは避け、移動ロボット自身が環境構造に依存した条件で停止すべきである。例えば、行為系列の長さが、環境の大きさに相関があるような設定が望ましい。
- 環境の識別精度：AEM の行動にとって、次に重要な条件として、異なった環境ができるだけ高い精度で判別できることがある。
- 認識効率：AEM では、認識のために行動しなければならないため、行動に依存して認識効率が大きく変わる。よって、できるだけ効率のよい行動、つまりできるだけ少ない行為で環境認識できるような行動を獲得することが望ましい。

以下で、各条件について、具体的に適合度関数を設定していく。

〔1〕行動の停止性の評価

本研究では、移動ロボットの出発点は、大まかに同定可能であると仮定する。よって、環境の大きさを反映できる停止条件として、環境を徘徊し、一定ステップ内に、出発点の近傍に戻って来たときに停止させることにする。このような行動を帰巢 (homings) と呼び、帰巢が達成されたときに行動は停止する。さらに、一度は出発点近傍から出たことも評価することを考え、下の評価関数 g を用いる。この関数の値域は、 $[0, 1]$ であり、 $g=1$ のとき、すべての環境で帰巢が成功したことを意味する。なお、下式の“総環境数”とは、訓練環境と試験環境の和集合である。

$$g = \frac{(\text{脱出成功個体数}) + (\text{帰巢成功個体数})}{2 \times (\text{総環境数})}$$

上の評価値を得るには、ある環境で帰巢できなかったとしても、各個体をすべての環境で行動させなければならない。しかし、経験的には、ある一つの環境で帰巢できない個体は、他の環境でも停止しない場合が多いこと、さらに、環境での行動の回数が全体の試験時間に大きく影響することから、個体の評価は、ある環境で帰巢できなかった時点で、 $g=0$ として終了することにする。よって、環境は順に調べられ、帰巢できなかったときにその個体の行動は打ち切られる。

〔2〕環境の識別精度

本研究では、環境の識別方法として、1-Nearest Neighbor 法を用いているが、この方法により試験環境がどの程度正しく識別されるかが識別精度の評価指針となる。よって、正しく識別された試験環境数を全体の試験環境数で割って正規化した下式で記述される関数 h を、環境の識別精度の評価関数として導入する。関数 h の値域は、 $[0, 1]$ であり、 $h=1$ のとき正答率 100% となる。また、識別精度よりも停止性を優先するため、 g が 1 になるまで、 h の値は 0 とする。

$$h = \begin{cases} \frac{\text{正しく識別された試験環境数}}{\text{試験環境の総数}} & g = 1 \\ 0 & 0 \leq g < 1 \end{cases}$$

〔3〕認識効率

AEM では、環境を観測するだけではなく、実際に環境において行動することなしには、認識ができない。一般に実世界での行動は多くの時間コストを必要とするため、AEM は、観測による認識よりも環境認識の効率が悪いことが問題となる。よって、できるだけ短い行為系列で環境を認識できることが重要である。そこで、下式の評価関数 k を導入する。訓練環境と試験環境のすべての例において、その認識に要した行為系列の長さの総和をとり、行為系列の長さの上限の総和で割って正規化する。

$$k = \begin{cases} 1 - \frac{\sum_{i=1}^n S_i}{n * S_{max}} & h = 1 \\ 0 & 0 \leq h < 1 \end{cases}$$

上式において、 S_i は、環境 i で得られた行為系列の長さ (= 行為数)、 S_{max} は、許されている行為系列の長さの上限、 n は環境数である。関数 k の値域は、 $(0, 1]$ であり、値が大きいほど、環境認識の効率がよい。また、認識効率よりも識別精度を優先するため、識別精度関数 h が 1 になるまで、 k の値は 0 とする。

以上で述べた、評価関数 g, h, k を使って、適合度関数 $f = g + h + k$ を設定する。これにより、まず $0 \leq f \leq 1$ で停止性が成り立つようにして、次に $1 < f \leq 2$ で環境の判別能力を保障させ、最後に $2 < f < 3$ で認識効

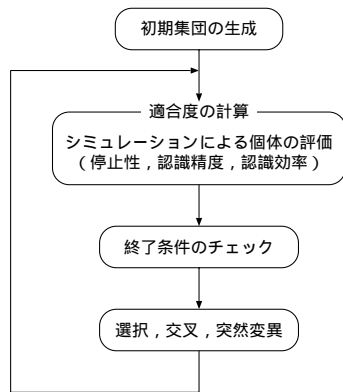


図5 全体の手続き

率向上を目指すということになる。

以上の手続きをまとめると、図5のようになる。シミュレーション上で移動ロボットを各個体の遺伝子に書かれた行動を使って走らせた結果から、適合度を計算している。

5. 実験

以上のような枠組で、AEMにとって適切な行動が実際に獲得できるのかどうかを調べるため、実験を行う。

障害物回避等のように行動毎に独立に評価が行える領域では、GAのそれぞれの個体の行動を時分割的に一定時間だけ適用してロボットを走らせ、適合度を評価できるので、実機のロボットを連続して走らせながらのオンライン学習の研究例がある [Nordin 96]。しかし、本研究では、行動は単独では評価できず、各環境で実行された行為系列がすべて得られたときに、始めて適合度が評価される。このことは、適合度評価のコストが、障害物回避の場合に比べて非常に高いことを意味する。さらに、試行毎に初期位置を設定しなければならないことが、実機を用いた実験では大きな負荷となる。よって、実環境で実機を走らせて、適合度の評価を行ってGAを走らせることは、現実的に無理がある。そこで、本研究では、Kheperaのシミュレータ [Michel 96] に手を加えたものを使って、シミュレーションによる実験を行った。実験環境のパラメータ設定を以下に示す。これらは、本研究の実験において共通した設定である。

- 環境サイズ: 100cm×100cmの正方形内の多角形。
- 出発点近傍のサイズ: 半径10cmの円。
- 移動ステップ数の上限: 2000。

なお、シミュレータには、モータ速度に±10%、左

表1 実験1の結果

環境	訓練環境	停止世代	最大適合値
(a)	{empty, L}	1.0 (0)	2.80 (0.106)
(b)	(a) + L2	2.6 (1.84)	2.43 (0.131)
(c)	(b) + invL	2.8 (1.69)	2.44 (0.025)
(d)	(c) + small-empty	2.8 (1.75)	2.44 (0.093)
(e)	10の環境	2.1 (0.738)	2.51 (0.024)
(f)	12の環境	5.2 (3.08)	2.48 (0.07)

右のモータの速度差による回転角に対し±5%のランダムな誤差がある。さらに、赤外線近接センサの距離の値に±10%、光センサの値に±5%のランダムノイズがある。また、本実験では、適合度が2を越えると停止させる。つまり、訓練環境と試験環境に対し、認識が確実に行われるとGAは停止する。

すべての実験において、訓練環境と同じ試験環境を与えた。また、ロボットの出発位置と方向も、全ての環境において中央下で左方向と固定されている、しかし、ノイズのために、同じ環境と同じ出発条件でも全く同じ行動をとるとは限らない。また、それぞれの実験では、センサとモータのノイズと初期世代での各個体の遺伝子の決定に用いる乱数の初期値を変えた10回の試行を行い、GAが停止した世代数とそのときの最大適合度の平均と標準偏差を調べた。

5・1 実験1: 輪郭の異なる環境の認識

まず、輪郭の異なる環境の識別が可能かどうかを実験的に調べる。empty, L, L2, invL, small-emptyの5つの環境から、2, 3, 4, 5種類を選択して実験した。さらに、10種類、12種類の輪郭の異なる環境での実験も行った。実験結果を表1に示す。表中の数値は初期値の異なる乱数を用いた場合の平均値であり、括弧中の数値はその標準偏差である。この表記は、以降の実験でも同様である。

5種類の実験で得られた行動を実行した軌跡を図6から図10に示す。大体時計周りに環境内を移動しているのがわかる。また、表1の標準偏差からわかるように、乱数の初期値の違いにより停止世代のばらつきが大きい。対照的に、最大適合度のばらつきは小さい。この傾向は、以降の実験でも共通してみられた。

5・2 実験2: 光源位置の異なる環境の認識

次に、環境の輪郭は正方形に固定したまま、光源の個数と位置を変えた環境 empty, 1-lamp, 2-lamp, 3-lamp, 4-lampを用意して実験を行った。光源は、環境全体を照らすことが可能で、移動ロボットは環境内のどの位置でも光源の方向を認識可能である。前述の5

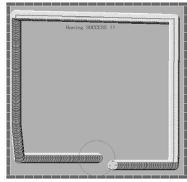


図6 empty 環境

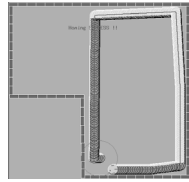


図7 L 環境

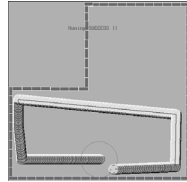


図8 L2 環境

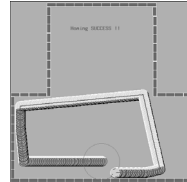


図9 invT 環境

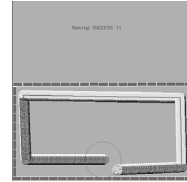


図10 small-empty 環境

表2 実験2の結果

環境	訓練環境	停止世代	最大適合値
(g)	empty, 1-lamp	1.6 (0.966)	2.68 (0.196)
(h)	(g) + 2-lamp	4.8 (2.78)	2.59 (0.131)
(i)	(h) + 3-lamp	9.3 (5.19)	2.59 (0.111)
(j)	(i) + 4-lamp	10.0 (5.94)	2.62 (0.075)

つの環境から、2, 3, 4, 5 種類を選択し、実験を行った。その結果を表2に示す。また、5種類の環境の実験で得られた行動を実行した軌跡を図11~図15に示す。なお、図中の黒丸は、光源を表す。

また、図16に、表2(i)の1試行における適合度の平均と最大値の推移を示す。平均値、最大値ともに漸近的に適合度が増加しており、この問題がGAに適していることがわかる[伊庭94]。また、エリートにより、停止性(適合度 ≥ 1)が第7世代で獲得されている。

図11~図15でのロボットの挙動は、人間が見て、容易に解釈できない。つまり、人間の設計者が直観的に設計することが難しい行動が獲得されている。

5・3 実験3：光源位置と輪郭の異なる環境の認識

次に、3つの輪郭の異なる環境、そしてその3つそれぞれにランプを一つ加えた3つの環境を混ぜた場合の実験を行った。ここでは、与えられた6つの環境 empty, 1-lamp, L, L-1-lamp, invT, invT-1-lamp の一つ一つを全て環境を別環境として識別する。実験結果として

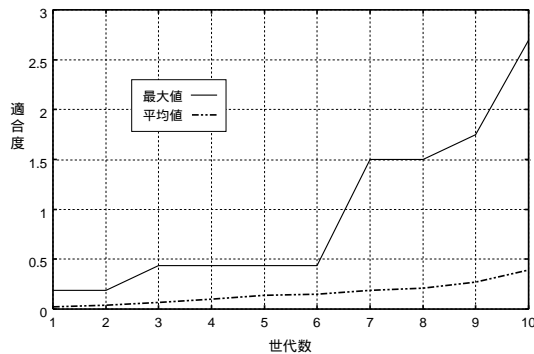


図16 適合度の推移

は、停止世代の平均が13.2(標準偏差 7.67)、最大適合値の平均2.61(標準偏差 0.091)がであった。また、6種類の環境の実験で得られた行動を実行した軌跡を図17~図22示す。

これらのロボットの挙動も、実験2と同じく人間が解釈することが難しい。よって、このような行動を、人間がGAなどの探索を用いることなしに獲得することは難しい。

5・4 実験4：複数環境からなる同一環境の認識

さらに、複数の環境を同一環境とすべき問題で実験を行った。実験3と同じ6つの環境を用いて、輪郭が同じならランプが灯っていようがいまいが同じ環境とする行動の獲得を調べた。つまり、{empty, 1-lamp}, {L, L-1-lamp}, {invT, invT-1-lamp} の3種類に分類する。実験結果としては、適切な行動が獲得され、そのときの停止世代の平均が8.8(標準偏差 4.24)、最大適合値の平均が2.61(標準偏差 0.091)であった。

6. 考 察

6・1 行動学習の難しさ

表1をみると、環境(a)では停止世代が1であり、初期世代の個体集団に既に解が含まれていることになる。よって、探索空間のサイズが膨大でありながら、環境(a)では解の数が非常に多いと考えられる。このことを調べるために、環境(a)の実験において前述と同じ10試行において停止条件を満たした最大適合度をもつ個体を実行した行動を調べた。その結果10個体の行動は、表3の5つに分類された。図中の括弧内の数字は、そのパタンに分類される個体の数であり、各行為は3・3節で説明したものである。また、表中の各状態の直感的な意味を以下に示す。表3のどの行動集合も、回りに何もなければ直進し、障害物があれば、回転により衝突を回避するという行動パターンである。

- 状態 S0：周囲に障害物も明かりもない。
- 状態 S9：左に障害物があり、明かりはない。
- 状態 S18：前に障害物があり、明かりはない。

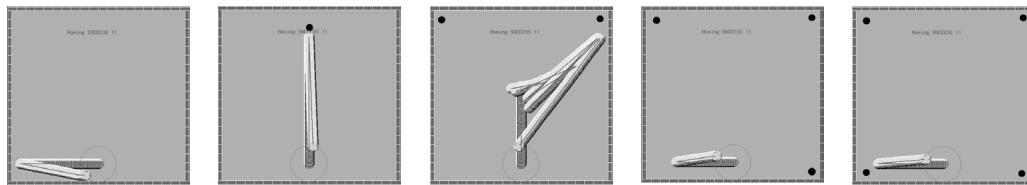


図 11 empty 環境 図 12 1-lamp 環境 図 13 2-lamp 環境 図 14 3-lamp 環境 図 15 4-lamp 環境

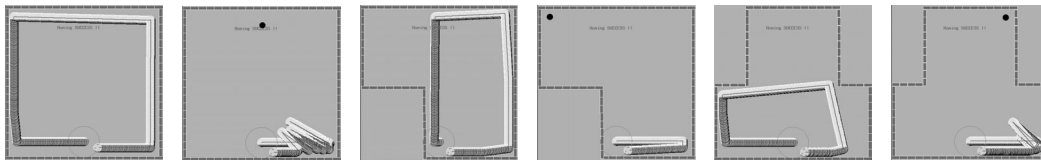


図 17 empty 図 18 1-lamp 図 19 L 図 20 L-1-lamp 図 21 invT 図 22 invT-1-lamp

●状態 S36：右に障害物があり，明かりはない．

表 3 から分かるように，これらの 10 個体は，4 つの状態 {S0, S9, S18, S36} の部分集合にしか遭遇していないことがわかる．つまり，実際の行動については， $4^4 = 256$ の組み合わせを考えればよい．また，その探索空間中に，少なくとも，BS-1 に属する解が 4 つ*2，BS-2 に属する解も 4 つ，そして BS-3, BS-4, BS-5 の解が一つずつあるので，解の数の合計は，11 となる．実験した GA の個体数は 50 なので，初期個体中に解が一つ以上含まれる確率は， $1 - \left(\frac{256-11}{256}\right)^{50} = 0.89$ となる．よって，初期個体中に解が含まれている場合が多いという表 1 の環境 (a) の結果が納得できる．ただし，停止世代の平均が 1 なのは，環境 (a) だけであり，他の環境では，識別すべき環境数や形状に依存して，より大きい値をとっている．

さらに，他の実験においても同様に実行された行動の種類の数調べたところ，10 試行での平均が 3.4 ~ 18.4 であった．つまり，移動ロボットは，全 72 状態のうちのごく一部にしか遭遇していないことがわかる．このことを利用して，遭遇した状態を条件部にもつ行動の突然変異率を高くするなどのヒューリスティックスが今後考えられる．

6・2 学習された行動の一般性と効率

本研究において学習された行動の一般性，つまり変化した環境でも正しく認識できる頑健性について考察する．5 章で述べたようにシミュレータには，センサ

*2 状態 S0, S18, S36 の行為がそれぞれ A1, A2, A2 であればよい．つまり，残りの状態 9 については，4 つの行為のどれでもよいので，BS-1 に属する解は 4 つとなる．次の BS-2 の場合も同様．

値とモータにランダムノイズが含まれている．よって，実験で得られた行動は，それらのノイズに対しては頑健であると言える．また，環境ベクトルが大きく変化しない範囲での環境の変化（例えば，壁の変形，移動の障害にならない障害物など）には頑健である [Yamada 98b]．

また，進化計算による移動ロボットの行動学習は，初期位置と初期方向の変化に対して頑健でないことが，先行研究 [Reynolds 94, 星野 97] により指摘されている．このことを調べるために，初期位置と方向に 1% のランダムノイズを含んだ試験例を用いて実験 1 を再度行ったところ，行動が学習されたのは，環境 (a) ~ (c) だけであり，(d) ~ (f) は 100 世代までには行動が学習されなかった．この結果から，本研究においても，初期位置と方向の変化に対して頑健な行動を学習することは困難であることがわかる．

次に，人間が設計可能な典型的行動である壁沿い移動との比較を行なう．実験 1 のそれぞれの環境に対して，壁沿い移動と実験 1 で学習された行動を基に移動ロボットが動作したときの移動距離を比較した結果が，表 4 である．ただし，本研究で用いた状態と行為では，

表 3 実行された行動の集合

行動集合	行動	行動集合	行動
BS-1 (5)	S0 A1	BS-4 (1)	S0 A1
	S18 A2		S9 A1
	S36 A2		S18 A2
	S36 A4		
BS-2 (1)	S0 A1	BS-5 (1)	S0 A1
	S9 A3		S9 A4
	S18 A3		S18 A3
	S36 A2		
BS-3 (2)	0 A1		
	S9 A3		
	S18 A2		
	S36 A4		

表4 壁沿い移動との移動距離 (cm) の比較

環境	(a)	(b)	(c)	(d)	(e)	(f)
壁沿い移動	680	1020	1360	1610	2996	3471
学習された行動	161	694	989	1280	2280	2822

壁沿い移動を設計することが難しかったため、実際に移動ロボットを走らせることはせずに、壁の5cm内側をなぞった軌跡の距離を測って表中の壁沿い移動の移動距離とした。表4から明らかなように、どの環境においても、壁沿い移動よりも移動距離の短い行動、つまり効率の良い行動が学習されている。ただし、学習された行動で識別できる環境は、訓練(試験)環境に依存しているため、より多くの環境を識別できるという意味での一般性は、壁沿い移動の方が高い。

7. ま と め

環境で実際に行動することにより得られる行為系列を基に、環境をモデリングするAEMにおいて、適切な行動を遺伝的アルゴリズムにより獲得する方法を提案し、簡単な環境において実験を行った。その結果、単純な環境においては、人間が直観的には設計するのが困難と考えられる行動が得られることがわかった。

本研究の各実験における学習は、1分～数時間(PentiumPro200, RAM128M)を必要とする。しかし、人間が行動を設計すると「壁沿い移動」という単純な行動でさえも、状態と行為の組み合わせを変えて移動ロボットを走らせてみるという試行錯誤を何回も繰り返す必要があり、この作業は相当の労力を要する。さらに、光源がある環境になると適切な行動が直感的に把握しにくくなるため、その設計はより難しくなる。よって、ある程度の時間をかけてでも、自動的にAEMに適切な行動を学習できることは、価値があると考えられる。

謝 辞

本研究は、文部省科学研究費重点領域研究「創発システム」(研究課題番号08233214)の支援を受けました。記して感謝いたします。

参 考 文 献

- [Arkin 91] Arkin, E. M., et al.: An efficiently computable metric for comparing polygonal shapes, *IEEE Trans. on PAMI*, Vol. 13, No. 3, pp. 209-216 (1991).
- [Clowly 85] Clowly, J. L.: Navigation of an intelligent mobile robot, *IEEE Trans. on R&A*, Vol. 1, No. 1, pp. 31-41 (1985).
- [Dasarathy 91] Dasarathy, B. V.: *Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques*, IEEE

- Computer Society Press (1991).
- [Goldberg 89] Goldberg, D. E.: *Genetic Algorithm in Search, Optimization and Machine Learning*, Addison Wesley (1989).
- [Holland 75] Holland, J.: *Adaptation in Natural and Artificial Systems*, University of Michigan Press (1975).
- [星野 97] 星野, 光本, 長野: ロボットの行動の進化とその頑健性, 計測自動制御学会, Vol. 33, No. 6, pp.533-540 (1997).
- [伊庭 94] 伊庭齊志, 遺伝的アルゴリズムの基礎, オーム社 (1994).
- [Ito 96] Ito, T., Iba, H., and Kimura, M.: Robustness of robot programs generated by genetic programming, In *Proc. of GP-96*, pp. 321-326 (1996).
- [加藤 96] 加藤, 石黒, 辻: 統計的解析による複雑な環境における環境モデルの獲得, 日本ロボット学会誌, Vol. 14, No. 5, pp. 660-667 (1996).
- [Koza 91] Koza, J. R.: Evolution of subsumption using genetic programming, In *Proc. of ECAL*, pp. 110-119 (1991).
- [Koza 92] Koza, J. R.: *Genetic Programming*, MIT Press (1992).
- [Mataric 92] Mataric, M. J.: Integration of representation into goal-driven behavior-based robot, *IEEE Trans. on R&A*, Vol. 8, No. 3, pp. 14-23 (1992).
- [Michel 96] Michel, O.: *Khepera Simulator v.2 User Manual*, University of Nice-Sophia Antipolis (1996). (<http://diwww.epfl.ch/lami/team/michel/khep-sim/>).
- [Nakamura 96] Nakamura, T., et al.: Behavior-based map representation for a sonar-based mobile robot by statistical methods, In *Proc. of IROS-96*, pp. 276-283 (1996).
- [Nehmzow 91] Nehmzow, U. and Smithers, T.: Map-building using self-organizing networks in really useful robots, In *Proc. of SAB-91*, pp. 152-159 (1991).
- [Nordin 96] Nordin, P. and Banzhaf, W.: An On-Line Method to Evolve Behavior and to Control a Miniature Robot in Real Time with Genetic Programming, *Adaptive Behavior*, Vol. 5, No. 2, pp. 107-140 (1996).
- [Reynolds 94] Reynolds, C. W.: Evolution of obstacle avoidance behavior: Using noise to promote robust solutions, In *Advances in Genetic Programming*, Vol. 1, chapter 10, pp. 221-241. MIT Press (1994).
- [Yamada 98a] Yamada, S.: Learning Behaviors for Environment Modeling by Genetic Algorithm, In *Proc. of The First European Workshop on Evolutionary Robotics*, pp. 179-191 (1998).
- [Yamada 98b] Yamada, S. and Murota, M.: Unsupervised learning to recognize environments from behavior sequences in a mobile robot, In *Proc. of ICRA-98*, pp.1871-1876 (1998).

{ 担当編集委員 : × × , 査読者 : × × }

著 者 紹 介

山田 誠二(正会員)

1984年大阪大学基礎工学部卒業。1989年同大学院博士課程修了。同年大阪大学基礎工学部助手。1991年同大学産業科学研究所講師。1996年4月より、東京工業大学大学院助教授、現在にいたる。工学博士。人工知能、Webでの情報検索、ロボット学習に興味をもつ。情報処理学会、日本ロボット学会、AAAI、IEEE各会員。

yamada@ymd.dis.titech.ac.jp