TC5-1

移動ロボットのための 複数のタスク環境における学習に適した形態の設計法

杉浦 孔明 *†‡, 川上 浩司 ‡
† ATR 音声言語コミュニケーション研究所, ‡ 京都大学 情報学研究科

A Method of Designing the Morphology of Mobile Robots for Learning Multiple Tasks

Komei Sugiura†‡, Hiroshi Kawakami‡
† ATR Spoken Language Communication Research Laboratories, ‡ Graduate School of Informatics, Kyoto University

Abstract: This paper proposes a method that automatically designs the sensory morphology of an autonomous robot. This method uses two kinds of adaptation, ontogenic adaptation and phylogenetic adaptation, to optimize the sensory morphology of the robot. In ontogenic adaptation, individuals with many different sensory morphologies use reinforcement learning to adapt to a task. In phylogenetic adaptation, a Genetic Algorithm is used to select morphologies with which the robot can learn the task faster. The method is applied to the design of the sensory morphology of a line-following robot. Some example of morphologies designed by the method are shown.

キーワード: センサ進化, 学習と進化, 形態による計算, 身体性

Keywords: sensor evolution, learning and evolution, morphological computation, embodiment

1. はじめに

ロボットの行動学習に関する研究には，ロボットの形態が固定されているという前提に立つものが多い．それらの研究では，形態がロボットの制御系と独立に設計されており，強化学習などを制御系に導入してタスクを学習させる．しかし，環境の変化に適応的なロボットを作り出すためには，形態と制御系の関係を利用することが重要である [13]．ロボットの振る舞いは形態・制御系・環境の三者によって決定されるためである [4]．

浅田らは，実環境で学習手法を適用する際の制約条件を 2 つ挙げている [7]．ただし，これらの制約条件は，教師なし学習，特に強化学習を念頭に置いていると考えられる．

1. 学習・進化アルゴリズムが適用可能な状態-行動空間を構成できるか？

   ロボットが自らの体験を通じて，どのように状態-行動空間を構成するかといった問題は，実は学習手法自体よりも重要で困難な問題である．

2. 実世界で学習可能な時間か？

   学習時間は状態-行動空間の指数オーダーで増大するので，少し複雑な問題では，事実上実行不可能なほど探索量が増える．実世界で演算可能な範囲にするためには，何らかのバイアスが必要となる．

これに対し，提案手法は，(a1) 強化学習の結果得た収益を評価値として形態の探索を行なう，(a2) ユーザが学習時間を設定する，という特徴を持つ．これらの特徴により，上記の制約条件を解決できる．

強化学習では，経験系列は状態遷移の確率構造を推定するための訓練データとして用いられるのと同時に，方策を最善することにより収益を最大化するように，それ自体が最適化される [2]．つまり，形態の探索 (a1) は，確率構造を推定しやすく，かつ経験系列を改善しやすいような入力を得るために用いられる．ただし，本研究が扱うタスクは部分観測マルコフ決定過程であるうえ，有限時間の制限 (a2) があるため，客観的な最適性は議論できない．

本研究の独自性は，学習に適した形態を探索すること，つまり学習に有効なバイアスとして働く形態を設計することである．一方，先行研究では，認識系 (制御系の一部) または環境をバイアスとして利用する方法が試みられてきた [7]．これらは，状態空間の構成法 [1,3,11] やスケジューリング [6] といった手法に代表される．

状態空間の構成を工夫する場合には，経験を通じてタスクに適した認識系を内部構造として獲得する．これに対し，提案手法は認識系を陽に用いず，形態ベー

スの環境解釈系を探索する．ただし，提案手法は，認識系や環境をバイアスとして用いる手法と組み合わせて用いることもできる．

以下では，まず提案手法について説明する．次に，提案手法を適用した実験の設定と結果を述べ，考察を加える．最後に 4 節で本稿のまとめを行なう．

2. 学習に適したセンサ形態の自動設計法

本節では，まず提案手法を適用する対象およびタスクについて述べる．次に，提案手法の概略について述べた後，タスク学習とセンサ形態の設計法のそれぞれに関して詳説する．

2.1 提案手法の概略

提案手法は，多数のタスクを学習しやすいセンサ形態を設計するものである．本手法では，ロボットのセンサ形態をパラメータ化した後，多点探索により形態を設計する．具体的には，学習の進み具合を評価値として実数値遺伝的アルゴリズム (Real-Coded Genetic Algorithms, RCGAs) によりセンサ形態を探索する．GA の世代毎に新しいタスクを自動生成し，それぞれのタスクをロボットに学習させる．

本手法は，「ユーザが設定した時間内に，より良い性能を示すように学習が進む」形態を設計する．これは，実機ではバッテリの消耗や部品の摩耗の影響があるので，タスク試行時間には限りがあるためである．ロボットがタスクを学習する場合，その方法にはニューラルネットや強化学習など様々なものが考えられる (例えば [7])．ここで本手法では，強化学習を用いてタスクを学習させる．なお，1 つの形態につきタスク学習の機会を 1 回与え，タスクごとに学習結果を初期化する．

Figure 1 に，提案手法の概略を NS チャートで表現したものを示す．以下では，強化学習によるタスク学習と実数値 GA による形態設計について述べる．

2.2 強化学習によるタスクの学習

タスク学習のために Q 学習 [5] を用いる．学習のアルゴリズムを以下に示す．Figure 1 の対応する部分を括弧付きアルファベットで示す．

1. Q 値，位置を初期化する (e)
2. 行動ステップ t が最大行動ステップ数 T に達するまで，以下の 3. から 6. を繰り返す (f)
3. 行動ステップ t における行動を選択する (g)

Figure 1: NS チャートによる提案手法の概略

ロボットが t において，状態 s_t にあるとする．s_t は，各センサの入力 (0:黒，1:白) のみに依存する．このとき，ロボットは行動価値関数を参照し，ε-greedy 戦略に基づいて行動 a_t を選択する．ただし，a_t は 9 種類の行動 (直進，左折，右折について各々 3 種類) のいずれかとする．

4. 次状態 s_{t+1} と報酬 r_{t+1} を観測する (h)

   報酬は決定論的に与えられるとする．具体的な報酬の設定については 3 節で述べる．

5. Q 値および状態を更新する (i)

   Q 学習の学習則として，以下に示す一般的な更新則を用いる．

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

   ここに，α は学習率，γ は割引率である．

   $s_{t+1} \leftarrow s_t$ とする．

2.3 実数値遺伝的アルゴリズムによる形態設計法

本手法では，2 次元平面上のセンサ位置を設計変数とし，実数値 GA を用いてセンサ形態を設計する．GA の遺伝子型を以下の実数値ベクトル $\mathbf{p} \in \mathbb{R}^{2M}$:

$$\mathbf{p} = [p_{x_1}, p_{y_1}, p_{x_2}, p_{y_2}, ..., p_{x_M}, p_{y_M}]^T \qquad (1)$$

ここに，$[p_{x_i}, p_{y_i}]^T$ は，x-y 平面上での i 番目のセンサ位置であり，M はセンサの数を表す．

GA を用いる理由として，(a) 勾配を用いないこと，(b) 多目的最適化での実績があること，の 2 点が挙げられる．本研究ではタスク環境が単一ではないため，センサ配置の評価に関する勾配情報を利用しにくい．加えて，多くのタスク環境に対して十分なレベルの性能が得られる設計解が求められる．実数値 GA を用いることで，上記 2 つの問題を解決できると考えられる．

実数値 GA の交叉方法としては，単峰性正規分布交叉 (UNDX) [8]，UNDX-m [9]，シンプレクス交叉 (SPX) [12] などが提案されている．本手法では，交叉操作のために SPX を用いる．この理由として，SPX は変数間依存性が強い問題に適用可能であることが挙げられる．ロボットのコントローラがセンサ値を用いて学習を行う場合，学習結果はセンサ配置に大きな影響を受ける．そのため，このような場合におけるセンサ位置の設計では，各センサを独立に考えるのではなく，他のセンサとの関係を考慮する必要がある．つまりこのような問題では，変数 (センサ位置) 間依存性が非常に強い．SPX のアルゴリズムの詳細は後述する．

個体の評価値 ϕ は，タスク学習で与えられた報酬の総和とする．

$$\phi = \sum_{t=1}^{T-1} r_t \tag{2}$$

ここに，r_t はステップ t における報酬を表し，T は最大行動ステップ数を表す．本手法では，コントローラの学習結果は次世代に引き継がれない．そのため，ランダムに生成されたコースのいずれも対して，より多くの報酬を得られる個体が，より高い評価値の期待値を得ると考えられる．

本手法では，GA の世代交代モデルとして，トーナメント選択とエリート保存戦略を併用して用いる (後述)．これは，予備実験において Minimal Generation Gap(MGG) [10] が望ましい性能を示さなかったためであるが，個々の GA 手法に関する評価は本論文の主旨ではないので説明を省略する．

以下では，まず SPX の詳細について述べた後，設計のアルゴリズムについて述べる．

シンプレクス交叉 (SPX)　SPX の子個体生成のアルゴリズムは以下の通りである．

1. 個体群から $n+1$ 個の親個体をランダムに選び，これらを $\{\mathbf{p}_k : k = 0, 1, ..., n\}$ とする．
2. 親個体群 $\{\mathbf{p}_k\}$ の重心 $\mathbf{g}$ を求める．

$$\mathbf{g} = \frac{1}{n+1} \sum_{k=0}^{n} \mathbf{p}_k \tag{3}$$

3. $\mathbf{g}$ と各親個体 $\mathbf{p}_k$ $(k = 0, 1, ..., n)$ を $1 : 1 - \frac{1}{\alpha_s}$ に外分し，ベクトル $\mathbf{p}'_k$ $(k = 0, 1, ..., n)$ を得る．ここで，α_s は拡張率と呼ばれる正のパラメータで，推奨値は $\alpha_s = \sqrt{n+2}$ である．

$$\mathbf{p}'_k = \alpha_s \mathbf{p}_k + (1 - \alpha_s)\mathbf{g} \tag{4}$$

4. $\mathbf{p}'_k$ を用いて以下の漸化式で定義されるベクトル $\mathbf{c}'_k$ を求める．

$$\mathbf{c}'_k = \begin{cases} r_{k-1}(\mathbf{p}'_{k-1} - \mathbf{p}'_k + \mathbf{c}'_{k-1}) & (k = 1, ..., n) \\ \mathbf{0} & (k = 0) \end{cases} \tag{5}$$

ここに，r_k は区間 $[0, 1]$ 内の一様乱数 $u(0, 1)$ を次式で変換して得られる乱数である．

$$r_k = \begin{cases} 0 & (k < 0) \\ \{u(0, 1)\}^{\frac{1}{k+1}} & (0 \le k < n) \\ 1 & (k \ge n) \end{cases} \tag{6}$$

これより $\mathbf{c}'_n$ は，

$$\mathbf{c}'_n = -\mathbf{p}'_n + \sum_{k=0}^{n} \left\{ \prod_{i=0}^{k-1} r_{n-i-1} \right\} (1 - r_{n-k-1})\mathbf{p}'_{n-k} \tag{7}$$

となる．ただし，$\prod_{i=0}^{-1} r_{n-i-1} = 1$ とする．

5. 子個体 $\mathbf{c}$ を次式で得る．

$$\mathbf{c} = \mathbf{p}'_n + \mathbf{c}'_n \tag{8}$$

$$= \sum_{k=0}^{n} \left\{ \prod_{i=0}^{k-1} r_{n-i-1} \right\} (1 - r_{n-k-1})\mathbf{p}'_{n-k} \tag{9}$$

設計の流れ　以下に設計アルゴリズムを示す．手順 6. と 7. が提案手法の世代交代操作に該当する．ただし，括弧付きアルファベットで示された記号は，Figure 1 の対応する部分を表す．

1. 初期世代をランダムに生成する (a)
2. 世代数 n が最大世代数 N_g に達するまで，3. から 7. を繰り返す (b)
3. n 世代に対するタスク環境を設定する (c)
4. n 世代の各個体がタスク学習を行なう (d)
5. 全個体の学習が終了した後，その結果に応じて評価値 ϕ を与える (j)
6. 子個体数が非エリート個体数 (全個体の 90%で固定) に達するまで 6.1 と 6.2 を繰り返す (k)

6.1 全個体からランダムに $2n_t$ 個体を非復元抽出する

6.2 上位 n_t 個体を親個体として，SPX を用いて n_t 個の子個体を生成する．

7. n 世代のエリート個体と，5. で得られた子個体を併せて $n+1$ 世代を生成する (l)

3. 実験

本実験の目的は，提案手法を定量的に評価することである．本節では，まず実験設定，つまりタスク環境と設計対象，パラメータ設定について述べる．次に，実験結果を述べ，それに対して検討を加える．

3.1 タスク設定

ライントレースロボットを提案手法の適用対象とする．ライントレース競技では，ロボットは床に引かれたラインに沿って移動し競技タイムを競う．本手法を用いて学習に適したセンサ形態を設計させるために，コースを多数生成して，ロボットにライントレースを学習させる．用いるコースの例を Figure 2 に示す．生成されたコース 6 m 四方の平面上に設置される．

コースは GA の世代毎に以下のようにして生成される．

1. Figure 3 のように点 O を中心とする円を 16 分割し，各分割線上に一様乱数を用いてポイントを生成する．ただし，周回可能なコースを生成させるために，ポイント生成は中央部分を除き網掛けで示される部分に限る (Figure 3 参照)．

2. ポイントをスプライン補間で結合して閉曲線にする．Figure 4 のようにコース中央線を白線 (40 mm) とし，その両側を黒線 (100 mm) にする．これは，マイコンカーラリー (ライントレースの代表的大会) で標準的に用いられているコースを参考にした．

本研究では，シミュレーション中でセンサ配置を変更することにより，学習に適したセンサ配置を探索する．シミュレータの実装には Cyberbotics 製の Webots[1] を利用する．Webots では，ロボット形態やタスク環境などは Virtual Reality Modeling Language(VRML) で記述されているが，それらの変更は前提とされていない．そこで，シミュレーション中に VRML ファイルを変更し，それを Webots に読み込ませるソフトウェアを構築する．これにより，Webots を利用した形態設計が初めて可能になる．なお，本研究ではセンサ形態に限定するものの，本ソフトウェアを用いてアクチュエータや身体形状を変更することも可能である．

[1] http://www.cyberbotics.com

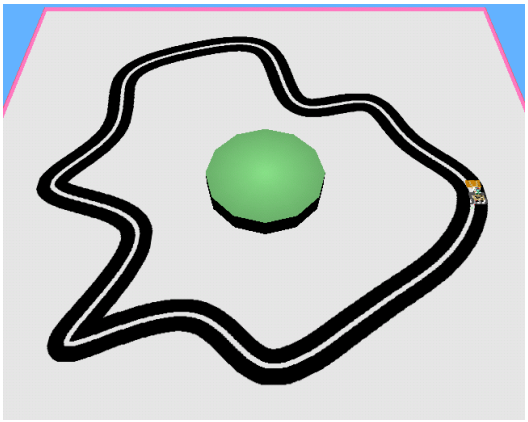

Figure 2: タスク環境の例

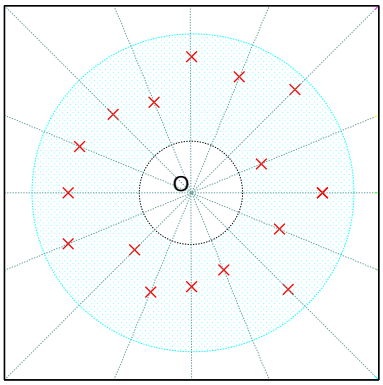

Figure 3: ランダムに生成されたポイントの例

報酬　ロボットには，ステップ t でのコース上の位置に応じて即時報酬が決定論的に与えられる．報酬の設計に際しては，「(a) コースを左回りに周回し，(b) 速い速度で，(c) コースの中央線 (白線) から近い位置を走行する」ほど高い報酬を与えるようにする．具体的には，ステップ t における即時報酬 r_t は以下の式で与えられるものとする．

$$r_t = \mathrm{sgn}(\omega_t) \cdot \|\dot{\mathbf{x}}_t\| \cdot B(\mathbf{x}_t) \tag{10}$$

$$\mathrm{sgn}(\omega_t) = \begin{cases} 1 & (\omega_t \geq 0) \\ -1 & (\omega_t < 0) \end{cases} \tag{11}$$

ただし，ω_t はコース中心 O から見たロボットの角速度，$\mathbf{x}_t$ は O を原点とする座標系における時刻 t でのロボットの位置，$\|\dot{\mathbf{x}}_t\|$ はロボットの速度の大きさである．また，$B(\mathbf{x}_t)$ は以下で与えられる．

$$B(\mathbf{x}_t) = 1 - \exp(-\beta \|\mathbf{x}_t - \mathbf{d}\|^2) \tag{12}$$

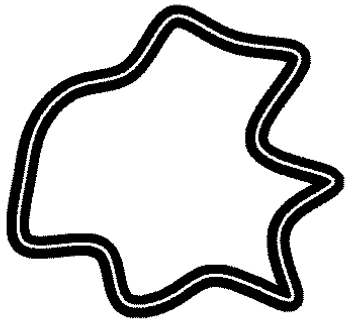

Figure 4: Figure 3 より得られるコース

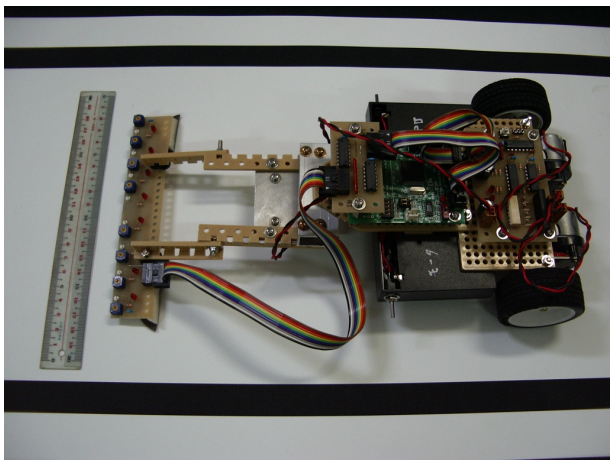

Figure 5: 通常のライントレースロボット

ここに，$\mathbf{d}$ はコース中央線上で $\mathbf{x}_t$ から最も近い点を表すベクトルであり，$\|\mathbf{x}_t - \mathbf{d}\|^2$ は $\mathbf{x}_t$ とコース中央線とのユークリッド距離の自乗を表す．また，減衰係数 β は 0.00125 で固定とした．

3.2 設計対象

センサ形態を設計する対象として，Figure 5 に示すライントレースロボットを用いる．このロボットは，マイコンカーラリーにおいて標準的に用いられている日立インターメディックス製のマイコンカーを元にしたものである．このロボットには，床面の色を 2 値で読み取る床センサを取り付けることができる．通常のセンサ形態では，センサは前部のボード上に規則的に配置される (Figure 5 参照)．このようなセンサ形態は，設計者がコントローラを設計する場合には都合がよい．

これに対し，ロボットが学習可能なコントローラを有する場合には，センサ配置がボード上に限定される必要はなく，規則的である必要もない．さらに，そのようなセンサ配置により性能を改善できる可能性がある．人工指先のセンサ形態設計において，センサデータの学習を前提とすることにより，センサ配置を簡略化できる試みについても報告されている [13]．そこで本研究では，Figure 6 のようにセンサを設置する範囲を拡張し，図の 160 mm × 350 mm の領域中で学習に適したセンサ形態を探索させる．センサ配置の範囲をボード上以外にも拡張したため，本手法による設計解は車輪より後部にもセンサが配置され得る．そのような設計解が得られた場合には，必要な部分にボードを取り付けて実機を製作すればよい．

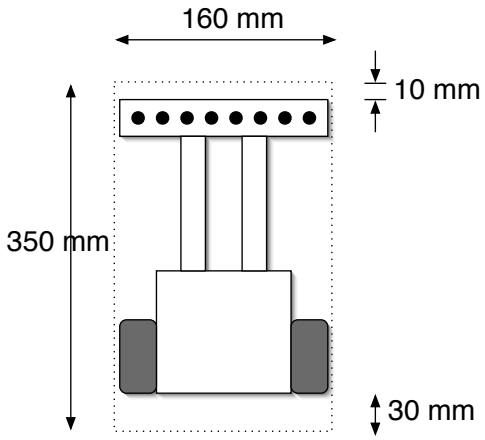

Figure 6: センサの設置範囲

3.3 パラメータ設定

実験で用いたパラメータの設定を Table 1 に示す．なお，実験を 10 回行ない，実験毎に世代数と等しい数のコースを新たに生成する．そのため，用いるコースの総数は 1000 種類である．

一般的には，学習時間が無限にあればセンサを分散させた方が有利であると考えられる．そこで，学習時間を長く，つまり T を 100,000 ステップ (6400 sec) として同様の実験を行なった．ただし，バッテリの消耗などの理由からこの設定は現実的ではない．

3.4 結果

まず定量的な結果を検討する．Figure 7 に世代数に対する評価値 ϕ の変化を示す．図において，× (実線で連結) はその世代における ϕ の平均を表し，+(点線で連結) はその世代の最良の ϕ を表す．また，値より上

Table 1: パラメータ設定

分類	パラメータ	値
タスク学習	最大行動ステップ数 T	30000, 100000
	学習率 α	0.05
	割引率 γ	0.99
	ε	0.01
形態設計	センサ数 M	8
	遺伝子長 $2M$	16
	最大世代数 N_g	100
	個体数 P	50
	親個体数 n_t	3

に伸びたエラーバーは，ϕ の標準偏差を示している．図の結果は，実験を 10 回行なってその平均値をプロットしたものである．

次に, Figure 8, 9 に得られた形態の例を示す．Figure 8，9 は，それぞれ $T = 30000$(通常)，$T = 100000$ における結果を示す．これらの形態はそれぞれの実験における最終世代の最良個体である．図には，ライントレーサの形状を重ねて描き，×によってセンサの位置を示した．なお，前にも述べた通り，それぞれの実験で用いた 100 種類の環境は，他の実験では一切用いられない．

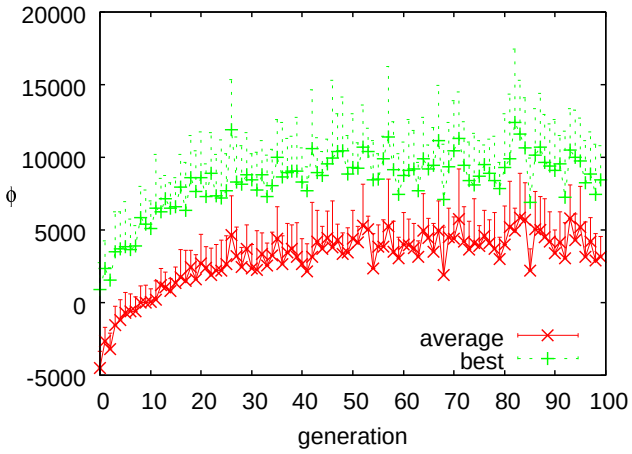

Figure 7: 世代数に対する評価値 ϕ の変化

3.5 考察

Figure 8 に対して検討を加える．設計解の定量的評価のためには，ϕ の最大値より ϕ の平均値の方が重要である．これは，世代間でタスク環境が異なるため，ある種の環境に特化した個体の ϕ が高くなる可能性があることによる．Figure 8 より，世代数に対して ϕ の平均値が増加していることから，提案手法により学習に

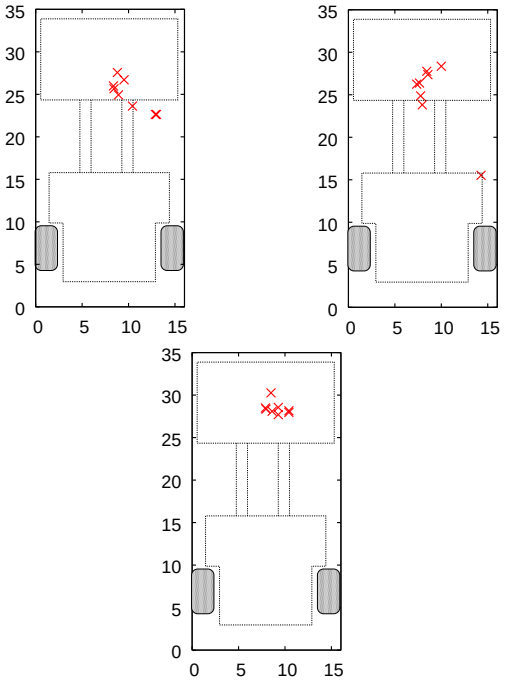

Figure 8: 設計された形態の例 (T=30,000)

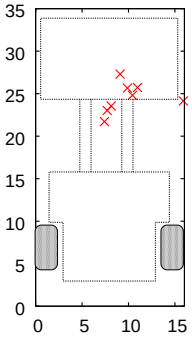

Figure 9: 設計された形態の例 (T=100,000)

適した形態が設計されたことがわかる．

次に，センサ配置の特徴について考察する．本実験では，センサの設置可能な領域をライントレーサ全体が覆われるように設定した．しかし Figure 8 より，設計解ではセンサがライントレーサの前部に集まって配置されていることがわかる．このような配置の利点は，複数のセンサを実質的に 1 つにまとめて，可能な状態遷移を少なくすることである．つまり，状態遷移を少なくして学習時間を短縮する利点があったためであると考えられる．

一般的には，学習時間が無限にあればセンサを分散させた方が有利であると考えられる．一方，Figure 8 と 9 を比較すると，センサの配置に関する T の影響は小さいことがわかる．つまりこのタスクにおいては，学習時間を現実的な範囲に留める場合，センサは分散

しない傾向があることが確認された．

4. おわりに

提案手法は，タスクに適した状態空間を設計するために，環境の解釈系を身体上に構築する手法であるといえる．Pfeifer は，形態による情報処理機能を morphological computation (形態による計算) と呼んでいる．しかし，morphological computation や類似の概念は，定量化されているとは言い難い．センサ形態設計に着目することは，「身体性の定量化」への第一歩となる可能性がある．

参考文献

[1] Inoue, K., Ota, J., Katayama, T. and Arai, T.: Acceleration of reinforcement learning by a mobile robot usinggeneralized rules, *Proceedings of 2000 IEEE/RSJ International Conference on INtelligent Robots and Systems (IROS 2000)*, pp. 885–890 (2000).

[2] Iwata, K.: *An Information Theoretic Analysis of Reinforcement Learning*, PhD Thesis, Kyoto University (2005).

[3] McCallum, A.: Instance-Based Utile Distinctions for Reinforcement Learning with Hidden State, *Proceedings of International Conference on Machine Learning*, pp. 387–395 (1995).

[4] Pfeifer, R. and Scheier, C.: *Understanding Intelligence*, MIT Press, Cambridge, MA. (1999). (石黒章夫, 小林宏, 細田耕監訳: 知の創成 – 身体性認知科学への招待, 共立出版, (2001)).

[5] Sutton, R. S. and Barto, A. G.: *Reinforcement Learning: An Introduction*, MIT Press (1998). (三上貞芳, 皆川雅章共訳: 強化学習, 森北出版, 2000).

[6] 浅田稔: 特集 ロボカップ 3 . ロボットプレーヤの感覚と学習, *bit*, Vol. 28, No. 5, pp. 37–43 (1996).

[7] 浅田稔, 國吉康夫: ロボットインテリジェンス, 岩波書店 (2006).

[8] 小野功, 佐藤浩, 小林重信: 単峰性正規分布交叉 UNDX を用いた実数値 GA による関数最適化, 人工知能学会誌, Vol. 14, No. 6, pp. 1446–1455 (1999).

[9] 喜多一, 小野功, 小林重信: 実数値 GA のための正規分布交叉の多数の親を用いた拡張法の提案, 計測自動制御学会論文集, Vol. 36, No. 10, pp. 875–883 (2000).

[10] 佐藤浩, 小野功, 小林重信: 遺伝的アルゴリズムにおける世代交代モデルの提案と評価, 人工知能学会誌, Vol. 12, No. 5, pp. 734–744 (1997).

[11] 高橋泰岳, 浅田稔: 実ロボットによる行動学習のための状態空間の漸次的構成, 日本ロボット学会誌, Vol. 17, No. 1, pp. 118–124 (1999).

[12] 樋口隆英, 筒井茂義, 山村雅幸: 実数値 GA におけるシンプレクス交叉の提案, 人工知能学会論文誌, Vol. 16, No. 3, pp. 147–155 (2001).

[13] 細田耕: 形態が学習にもたらすもの，学習が形態にもたらすもの, 日本ロボット学会誌, Vol. 22, No. 2, pp. 186–189 (2004).