目次

第 1 章　序論 **-1-**

1.1 はじめに 1

1.2 強化学習における学習空間の構成方法と問題点 1

1.3 従来研究 4

1.4 本研究の目的 4

1.5 アプローチ 4

1.6 本論文の構成 6

第 2 章　強化学習 **-7-**

2.1 強化学習の概要 7

2.2 行動学習手法 8

2.3 行動選択手法 8

2.4 強化学習におけるセンサの役割 8

第 3 章　提案手法 **-10-**

3.1 提案手法の概要 10

3.2 センサの重要度を算出する方法 11

3.3 センサの重要度からセンサの状態数を変更する方法 13

3.4 提案手法の処理手順 14

第 4 章　実験 **-16-**

4.1 仮想環境における検証実験 1 16

4.1.1 実験目的 16

4.1.2 実験設定 16

4.1.3 実験結果 18

4.1.4 考察 21

4.2 仮想環境における検証実験 2. 21

4.2.1 実験設定 21

4.2.2 実験結果 22

4.2.3 考察 25

4.3 実環境における検証実験 25
4.3.1 実験目的 25
4.3.2 使用するロボット 25
4.3.3 実験設定 28
4.3.4 実験結果 31
4.3.5 考察 38

第 5 章　結論 **-39-**
5.1 本論文のまとめ 39
5.2 これからの課題 39
5.2.1 遅延報酬環境への適用 39
5.2.2 複雑な実験設定での検証実験 39

謝辞 **-41-**

参考文献 **-42-**

第1章　序論

本章では，最初の第 1 .1 節にて，これまでのロボットと制御方法と現在ロボットの行動制御として期待されている強化学習について述べる．第 1.2 節では，強化学習の学習空間について説明し学習空間の構成方法による学習時間増加の問題を述べる．第 1.3 節では，従来研究について説明する．続く第 1.4 節では，本研究の目的を述べ，第 1.5 節では，目的達成に対するアプローチを説明する．最後に第 1.6 節では，本論文の構成を述べる．

1.1　はじめに

ロボットが運用され始めた当初，ロボットの用途は主に工場の生産ラインにおける単純な作業を行う事であった．現在では，技術の発達に伴い，ロボットが運用される環境や用途は変化している．これまで，運用されていた工場の生産ラインに加え，家庭環境や災害現場や深海等の人間が作業を行う事が困難である環境で運用されるロボット，ホビー目的や研究目的のロボット等，様々ロボットが開発・運用されている[1]-[5]．しかし，運用される環境や用途が変化する事による問題も発生している．従来ロボットが使用されていた工場の生産ラインではロボットの周囲の環境が変化する要因は少なく，ロボットが直面する環境を設計者が想定し環境に応じた行動を設計する事が可能であった．しかし，家庭環境や災害現場等では，ロボットが運用される環境が変化する要因が多く存在し，あらかじめロボットが直面する状況を想定する事が難しくロボットの行動を設計するのが困難になり始めている．

そこで，近年ではロボットの行動制御として，ロボットが自律的に環境に適した行動を獲得する強化学習が期待されている．機械学習の一種である強化学習は，学習者が周囲の環境に対して試行錯誤を繰り返す事によって，その環境に適した行動を学習する学習手法である．学習者は報酬と呼ばれるスカラー値を受け取り，報酬により行動の良し悪しを学習していく．強化学習については「第 2 章 強化学習」にて詳しく説明する．この強化学習をロボットに適用し，制御する方法が注目されている．実際に，強化学習をロボットに適用する研究は数多く行われている[6]-[9].

1.2 強化学習における学習空間の構成方法と問題点

強化学習における学習空間は，図 1.1 に示す様に行動軸と状態軸を持った空間で表現される．行動軸はロボットが選択できる行動を，状態軸は搭載されているセンサに対応している．よって，ロボットに搭載されるセンサが増加すると，それに伴い対応する状態軸が追加され状態数が増加する．そのため，学習空間が拡大してしまい学習時間が増加してしまう問題がある．

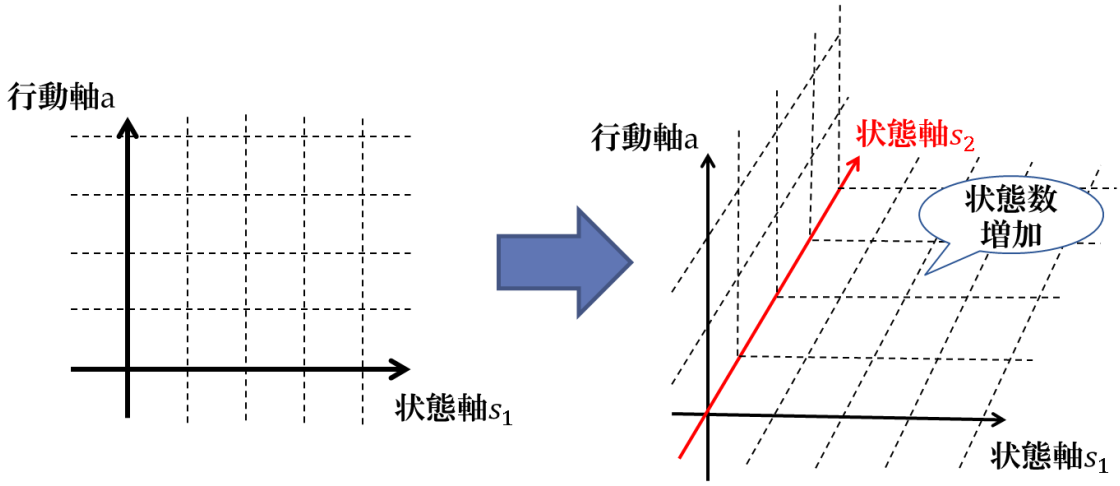


図 1.1 学習空間の拡大

学習時間増加の原因として学習空間の構成方法に問題があると考える．強化学習ではどのようなタスクに対しても搭載されている全てのセンサを十分考慮した学習空間を構成し学習を行う．例を図 1.2 に示す．図 1.2 は，金属検知センサ・距離センサ・音センサを搭載したロボットが空き缶ひろいタスク・パトロールタスクを行う場合の例である．

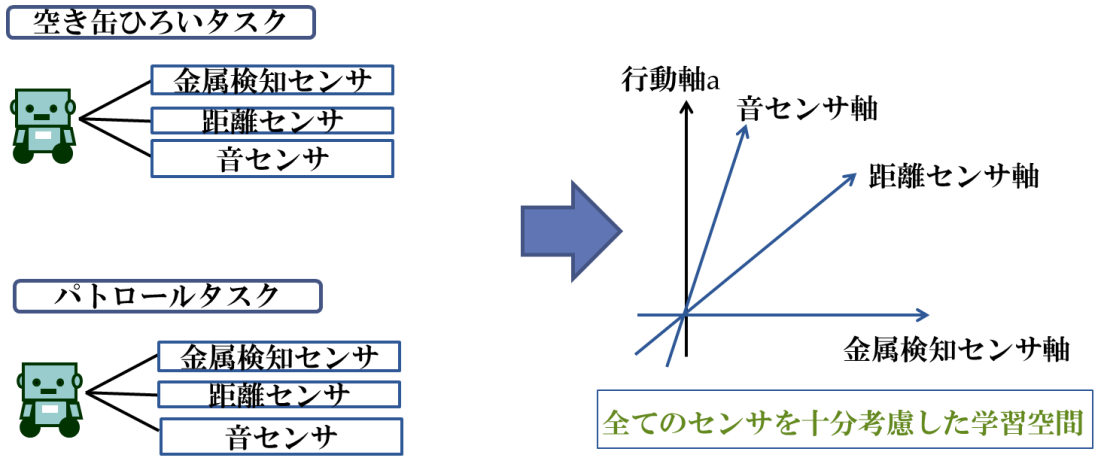


図 1.2 強化学習での学習空間

しかし，実際にはロボットに搭載されているセンサが全てタスク達成に重要とは限らないと考えられる．現に図 1.2 で述べた例では，空き缶ひろいタスクでは，空き缶を認識する距離センサや空き缶との距離を検出する距離センサは重要であるが，音センサに関しては重要ではない．一方，パトロールタスクでは音センサや距離センサといった周囲の状況を確認するセンサは重要であるが金属検知センサは重要ではない．この様に，搭載されている各センサのタスク達成に対しての重要度(以下，センサの重要度)は図 1.3 に示す様に異なると考えられる．

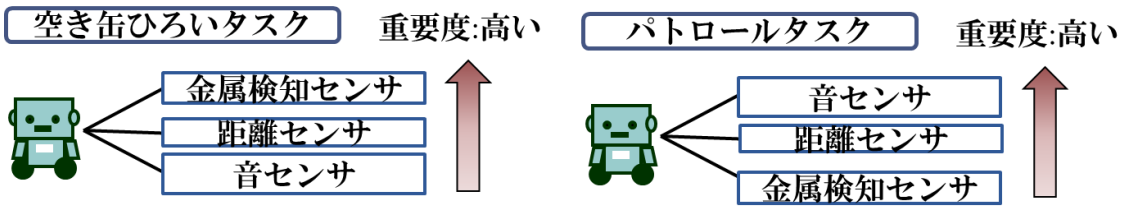


図 1.3 センサの重要度が異なる例

タスクによって搭載されているセンサの重要度が異なると考えられるため，本来であればタスク毎にセンサの重要度を考慮した学習空間を構成するのが望ましいと考えられる．今まで例に挙げてきたロボットでは，空き缶ひろいタスクに対しては距離センサと金属検知センサを考慮し音センサをあまり考慮しない学習空間を，パトロールタスクでは音センサと距離センサを考慮し金属検知センサをあまり考慮しない学習空間を構成するのが望ましいと考えられる(図 1.4).

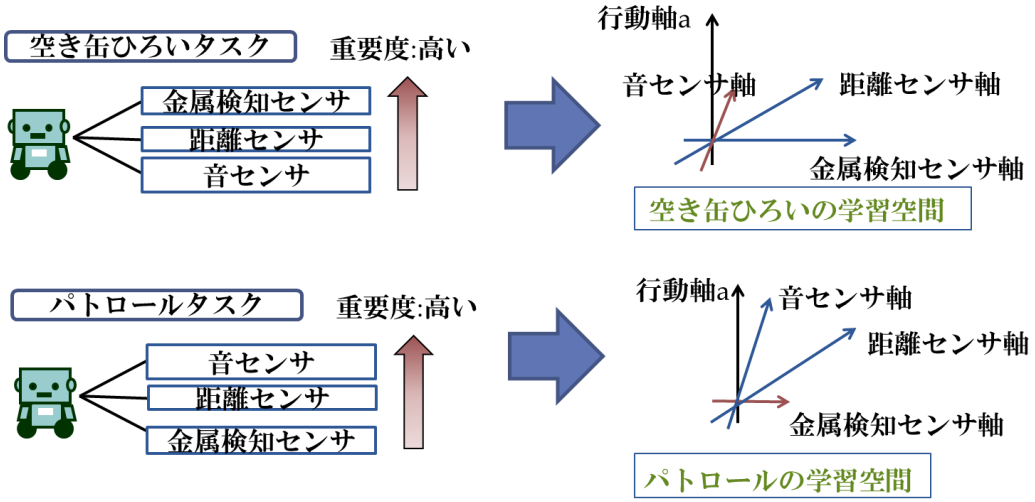


図 1.4 理想的な学習空間

以上で述べた様に強化学習では，センサの重要度を考慮した学習空間が構成できていない．タスク達成に重要では無いセンサに対しても考慮した学習空間を構成しているため，状態数が無駄に増え学習空間が拡大し学習時間が増加する．本研究は，今まで述べた強化学習における学習空間の構成方法に注目した研究であり，学習空間の構成方法が原因で起こる学習時間増加の問題を対象としている．

1.3 従来研究

学習空間の構成方法について扱っている研究として「強化学習によるロボットの行動獲得のための状間空間の自律的構成」，「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」，「センサ情報の自律的選択による効率的な行動選択の実現」等が挙げられる[10]-[12].

文献[10][11]では，ロボットに搭載されている全てのセンサがタスク達成に必要であるという前提で，学習空間の構成を行っている．そのため，タスク達成に必要が無いセンサがある場合でも，そのセンサを考慮した学習空間を構成している．一方，文献[12]では，搭載されているセンサがタスク達成に対して必要・不要の 2 種類に判別し，必要なセンサのみを使用して学習空間を低次元化している．しかし必要・不要と判別するため，必要ではあるが細かくセンシングする必要が無いセンサが存在する場合に，適切な学習空間が構成できない.

そこで，本研究ではセンサの重要度に応じて各センサを段階的に考慮した学習空間の構成方法を考える．センサがタスク達成に必要が無ければそのセンサに関しては考慮せず，必要であるならば，重要度に応じて重みを付けた学習空間を構成する手法を考える.

1.4 本研究の目的

本研究では，タスクに応じた学習空間の構成方法としてロボットに搭載されているセンサの重要度の違いに注目する．重要度の違いを考慮した学習空間をロボットが自律的に構成する手法を提案する．センサの重要度を考慮した学習空間により，強化学習の学習速度を向上させる事が目的である.

1.5 アプローチ

ロボットが搭載している各センサの重要度を算出する方法として，センサの出力値と報酬の関係に注目する．ロボットは状態をセンサを通して認識し，認識した状態がタスクに対して良いか悪いかにより報酬が決定される．よって，ロボットが状態を認識するために使用しているセンサの出力値と報酬の間には相関関係があると考えられる．例として，高度センサを搭載した登山タスクを行うロボットを図 1.5 に示す.

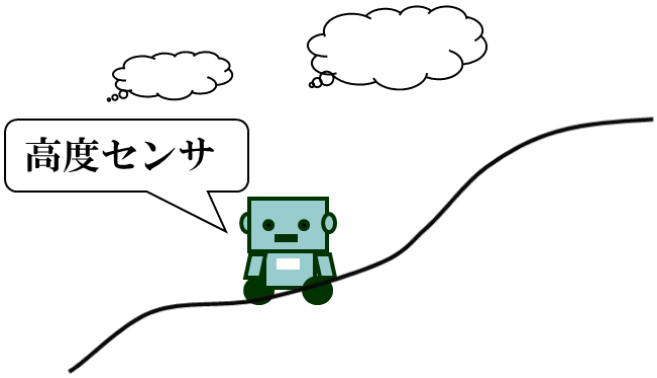


図 1.5　登山タスクを行うロボット

この時，高度センサの出力値が高ければ高い程，ロボットは登山タスクの進捗度が上がるためより高い報酬を受け取れる(図 1.6)．よって，本研究ではこの相関関係を利用する事でセンサの重要度を算出する．

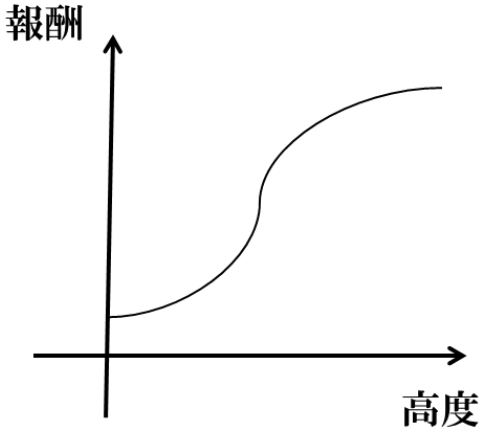


図 1.6　センサの出力値と報酬の関係

算出したセンサの重要度が高ければ，タスク達成に重要なセンサであると予測されるため，より細かくセンシングする必要がある．逆に，センサの重要度が低ければ，タスク達成に重要ではないセンサと予測されるため，細かくセンシングする必要はない．よって，図 1.7 に示す様に重要度の高いセンサの状態数を多く，重要度の低いセンサの状態数を少なくする．重要度の低いセンサの状態数を減らす事で学習空間が縮小し学習速度が向上すると考える．この様にセンサの重要度に応じて各センサの状態数を変更した学習空間をロボットが自律的に作成する手法を提案する．

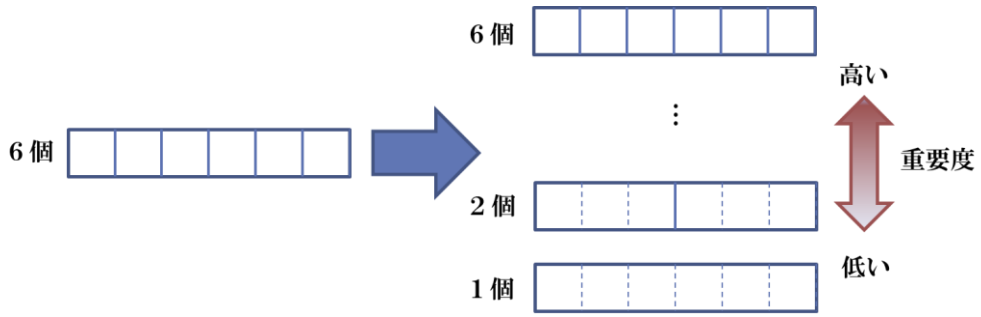


図 1.7 センサの重要度による状態数の変更

1.6 本論文の構成

第 1 章では、近年ロボットの制御に強化学習が注目されており様々な研究が行われている事を説明した．また，強化学習の問題点の一つである学習空間の構成方法による学習時間増加の問題点を述べた．この問題の解決を研究目的とすることを述べ，従来研究と問題解決へアプローチについて説明した．

第 2 章では，強化学習の概要を述べる。また、本研究で使用している行動学習手法と行動選択手法ついても説明し，強化学習におけるセンサの役割についても述べる．

第 3 章では、提案する手法について述べる．センサの重要度を算出する方法とセンサの重要度に応じてセンサの状態数を変更した学習空間の構成方法を説明し、提案手法の処理手順を示す．

第 4 章では、本研究が提案する手法の有効性を検証するために行った実験について述べる．仮想環境と実環境における 2 つの結果を示し，提案手法と一般的な強化学習の性能を比較する．

第 5 章では、本論文の全体としてのまとめを述べ，これからの課題についても説明する．

第 2 章　強化学習

本章では，強化学習について述べる．第 2.1 節では，強化学習の概要について述べる．続く第 2.2 節，第 2.3 節では本研究で使用した行動学習手法と行動選択手法について説明する．最後に第 2.4 節では，センサ情報による状態の定義方法について述べる．

2.1　強化学習の概要

強化学習[13]とは，エージェント(学習者)が周囲の環境に対して試行錯誤を繰り返す事により，その環境に適した行動を学習する機械学習の一種である．エージェントは報酬と呼ばれるスカラー値を基に学習を行う．報酬はエージェントの行動の良し悪しを数値化した物であり，行った行動の結果として環境から与えられる．エージェントはより高い報酬を獲得できる行動を探索し，報酬の総和を最大化する事が目的である．

また，エージェントは価値関数と呼ばれる関数を所持しており，価値関数により行動の価値を決定する．価値とはその状態を起点として将来に渡って獲得できる報酬の期待値であり，実際に獲得した報酬を基に算出される．価値が高い行動であれば将来的に高い報酬を獲得できるため，優先するべき行動と判断できる．価値関数の計算補法は利用する行動学習手法に基づき異なる．

エージェントは，行動価値を基に行動選択手法に応じて行動を選択する．行動選択手法とはエージェントが選択する行動の方針を表した物であり．手法により行動の選択方法が異なる．

強化学習の概要図を図 2.1 に示す．エージェントは時刻 t における状態s_tをセンサを通して認識し，行動選択手法に応じて行動a_tを選択する．行動a_tを行った結果として環境から報酬r_tを受け取り、行った行動の価値を行動学習手法に基づいて更新する．その後，行動a_tより変化した環境から時刻 $t+1$ の状態s_{t+1}をセンサを通して認識する．以後，同様の処理を繰り返す事により行動の学習を行っていく．

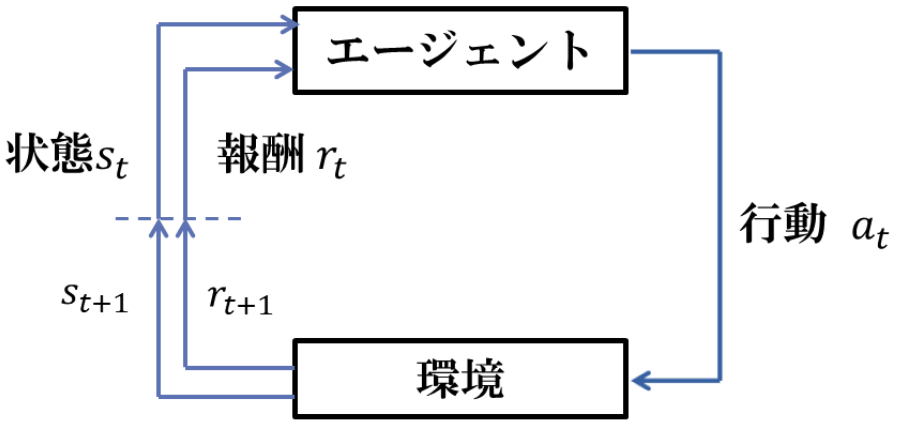


図 2.1　強化学習の概要

2.2　行動学習手法

行動学習手法とは，エージェントが行った行動を，どのように学習するかを表した手法であり，行動学習手法により価値関数の計算方法が異なる．行動学習手法として，Q 学習や加重平均手法等が挙げられる．本研究では加重平均手法を採用している．加重平均手法での価値関数の計算方法を式(2.1)に示す．時刻 t における状態をs_t ，エージェントが行った行動をa_t ，状態s_tにおける行動a_tの行動価値を$Q(s_t, a_t)$，時刻 $t+1$ における報酬をr_{t+1}としている．また，αは学習率と呼ばれる定数であり，$0 < \alpha \leq 1$の範囲の値を取る．行動価値 Q は，全ての状態と行動の組み合わせに対して存在する．

$$\mathbf{Q}(\boldsymbol{s_{t+1}, a_{t+1}}) \leftarrow \mathbf{Q}(\boldsymbol{s_t, a_t}) + \boldsymbol{\alpha}[\boldsymbol{r_{t+1}} - \mathbf{Q}(\boldsymbol{s_t, a_t})] \qquad \textbf{(2.1)}$$

2.3　行動選択手法

行動選択手法とは，エージェントが認識した状態に対して，どのような行動を選択するかを表した手法である．行動選択手法として，greedy法，$\epsilon-$greedy法，softmax 法等が挙げられる．本研究では，$\epsilon-$greedy法と呼ばれる行動選択手法を採用している．$\epsilon-$greedy法では，確率 1-ϵで最も行動価値が高い行動を，確率ϵで探査行動(ランダム行動)を選択する(図 2.2)．確率ϵで探査行動を選択する理由として，常に行動価値が高い行動を選択し続けた場合，十分な探索が行われず最適な行動が選択されない可能性が生じ，局所解に陥ってしまう場合がある． そのため，確率ϵで探査行動を選択する事により局所解に陥るのを防いでいる．確率ϵが高ければ高い程，探査的な行動(ランダム行動)を取る確率が高くなる．

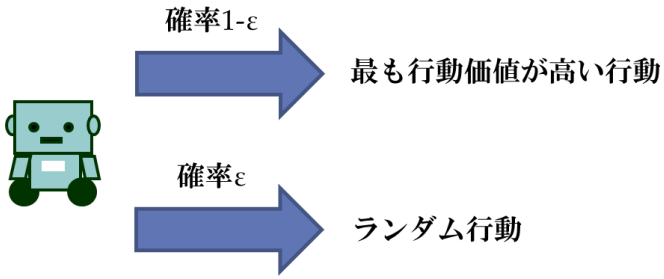


図 2.2　$\epsilon-$greedy法

2.4　強化学習におけるセンサの役割

強化学習におけるエージェントは状態をセンサを通して認識する．時刻 t における搭載しているn個のセンサの出力値を，それぞれ$e_{1,t}$，$e_{2,t}$，…，$e_{n,t}$とし，$e_{1,t}$，$e_{2,t}$，…，$e_{n,t}$がある時刻までに取得した値からなる集合をE_1，E_2，…，E_nとする．この時，エージェントの状態s_tは式(2.2)により定義される．これは，センサ情報の列ベクトルで表された物であり，最も簡単な定義方法である．

$$\boldsymbol{s_t} := \{(\boldsymbol{e_{1,t}},\ \boldsymbol{e_{2,t}},\ \cdots,\ \boldsymbol{e_{n,t}}) \mid \boldsymbol{e_{1,t}} \in \boldsymbol{E_1},\ \boldsymbol{e_{2,t}} \in \boldsymbol{E_2}, \cdots,\ \boldsymbol{e_{n,t}} \in \boldsymbol{E_n}\} \qquad (\boldsymbol{2.2})$$

しかし，各センサ値の集合E_1，E_2，…，E_nに含まれる要素の数が多い場合，このような定義方法はあまり適切ではない．例として，各センサ値の集合E_1，E_2，…，E_nに含まれる要素がいずれも 100 である場合, 状態数は100^nとなり, 学習空間が膨大になってしまうため, 現実的な時間内で学習する事が困難な場合がある．

そこで，センサ値を任意の長さ L の固定区間により分割し，分割した区間からセンサ値$e_{1,t}$，$e_{2,t}$，…，$e_{n,t}$をある変数$u_{1,t}$，$u_{2,t}$，…，$u_{n,t}$に写像する事を考える．写像は式(2.3)の通りに定義される．任意の長さ L の固定区間により分割されているため，$u_{1,t}$，$u_{2,t}$，…，$u_{n,t}$のある時刻までに取得した値からなる集合U_1，U_2，…，U_nに含まれる要素の数 N は各センサ値の集合E_1，E_2，…，E_nに含まれる要素の数よりも少なくなる．

$$\begin{cases} \boldsymbol{f}: \boldsymbol{E} \to \boldsymbol{U} \\ \boldsymbol{f}(\boldsymbol{e_{i,t}}) = \{\boldsymbol{u_{i,t}} = \ \boldsymbol{j} \mid (\boldsymbol{j-1}) \times \boldsymbol{L} \le \boldsymbol{e_{i,t}} \le \ \boldsymbol{j} \times \boldsymbol{L}\ ,\ \boldsymbol{j} \in \boldsymbol{N}\} \end{cases} \qquad \textbf{(2.3)}$$

式(2.3)により写像された状態$u_{1,t}$，$u_{2,t}$，…，$u_{n,t}$を基に状態s_tは式(2.4)により定義される．

$$\boldsymbol{s_t} := \{(\boldsymbol{u_{1,t}},\ \boldsymbol{u_{2,t}},\ \cdots,\ \boldsymbol{u_{n,t}}) \mid \boldsymbol{u_{1,t}} \in \boldsymbol{U_1}, \boldsymbol{u_{2,t}} \in \boldsymbol{U_2}, \cdots, \boldsymbol{u_{n,t}} \in \boldsymbol{U_n}\} \qquad (\boldsymbol{2.4})$$

以上で述べた様にエージェントは，時刻 t における状態s_tを搭載されているセンサを基に式(2.2),もしくは式(2.4)により認識する．

第 3 章　提案手法

本章では，初めに第 3. 1 節で本研究が提案する手法の概要を述べる．第 3.2 節では，各センサの重要度算出方法について述べる．第 3.3 節では，算出した重要度を基に，状態を統合し状態数を変更する方法，及び状態統合後の行動価値の算出方法を示す．最後に第 3.4 節では，提案手法の具体的な処理の流れについて述べる．

3.1　提案手法の概要

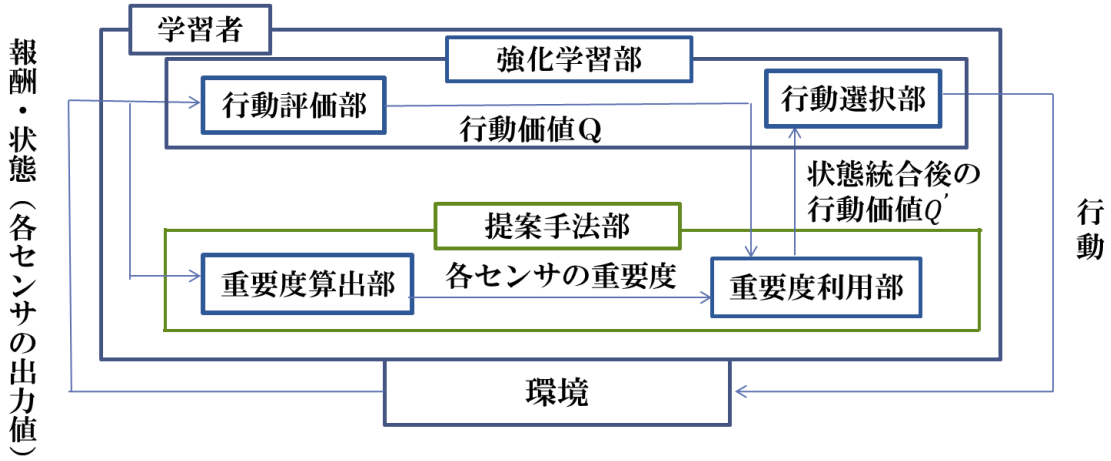


図 3.1 提案手法の概要図

本研究で提案する手法の概要図を図 3.1 に示す．提案手法は「重要度算出部」と，「重要度利用部」に分かれている．「重要度算出部」では，入力として各センサの出力値と報酬を毎時刻受け取る．その後，現時刻までに受け取ったデータを基に各センサの重要度を算出する．「重要度利用部」では，入力として各センサの重要度と各状態の行動価値 Q を受け取る．受け取った各センサの重要度を基に，各センサの状態を統合し状態数を変更した一時的な Q 空間を作成する ．例を図 3.2 に示す．図 3.2 では状態軸s_1に対応するセンサについては重要と判定され，状態軸s_2に対応するセンサが重要では無いと判定された場合である．状態軸s_2の状態が統合され状態数が少なくなっている．その後，受け取った各状態の行動価値 Q から作成した一時的な Q 空間の行動価値Q'を算出し「強化学習部」に受け渡す．「強化学習部」では算出された状態統合後の行動価値Q'を基に行動を選択する．以上が提案手法の概要であり，この手法により強化学習の学習速度の向上を目指す．

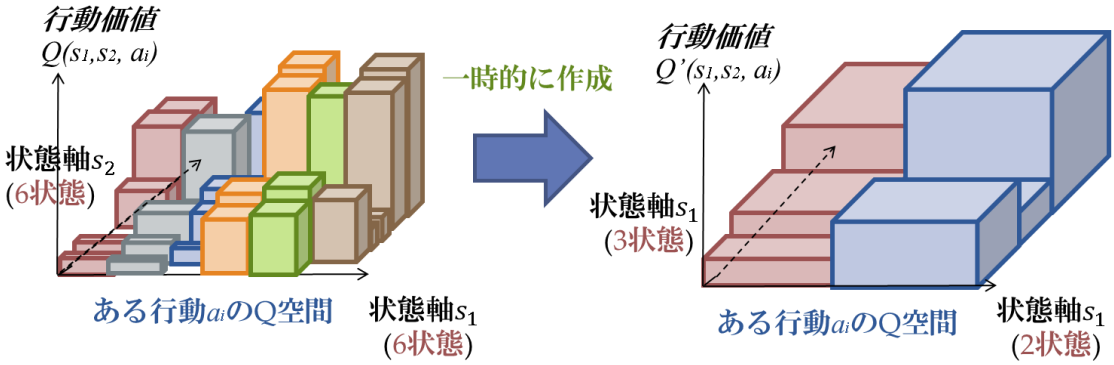


図 3.2 一時的に *Q* 空間を作成するイメージ図

3.2 センサの重要度を算出する方法

本節では，図 3.1「重要算出部」にてセンサの重要度を算出する方法を説明する．本研究では「1.4 アプローチ」で述べた通り，センサの出力値と報酬の間には相関関係があると考える．この相関関係を利用して，各センサの重要度を算出する．前提として，エージェントが搭載している n 個のセンサに対して，任意の時刻 t におけるセンサ情報を $e_{1,t}, e_{2,t}, \cdots, e_{n,t}$ とし，$e_{1,t}, e_{2,t}, \cdots, e_{n,t}$ が現時刻 T までに取得した値から成る集合を $E_{1,T}, E_{2,T}, \cdots, E_{n,T}$ とする．また，任意の時刻 t における即時報酬を r_t とし，現時刻 T までにエージェントが獲得した報酬の集合を R_T とする．エージェントは重要度を算出するために必要なデータとして現時刻 T までに得たセンサの出力値及び即時報酬を表 3.1 に示す様に記憶する．記憶したデータを基に各センサの重要度を算出する．

表 3.1 現時刻 T における重要度算出データ

時刻 T	センサ情報$E_{1,T}$	センサ情報$E_{2,T}$	…	センサ情報$E_{n,T}$	報酬R_t
0	$e_{1,0}$	$e_{2,0}$	…	$e_{n,0}$	r_0
1	$e_{1,1}$	$e_{2,1}$	…	$e_{n,1}$	r_1
⋮	⋮	⋮	⋮	⋮	⋮
T	$e_{1,T}$	$e_{2,T}$	…	$e_{n,T}$	r_T

センサの重要度を算出する方法として，報酬を目的変数，各センサの出力値を説明変数とした重回帰分析を行う．重回帰式として式(3.1)が成り立つ．$a_i(1 \leq i \leq n)$は回帰係数，a_0は定数項である．

$$r_t = a_1 e_{1,t} + a_2 e_{2,t} + \cdots + a_p e_{p,t} + a_0 \quad (3.1)$$

ここで，回帰係数a_iは各センサの出力値$e_{i,t}$の報酬r_tに対しての影響度を表している．つまり回帰係数a_1であれば，センサの出力値$e_{1,t}$がどの程度報酬に影響を与えているかを表している．よって，本研究では回帰係数a_iを各センサの重要度として利用する．回帰係数a_iは式(3.2)の連立方程式を解くことで求められる．

$$\begin{cases} s_1^2 a_1 + s_{12} a_2 + \cdots + s_{1n} a_n = s_{1r} \\ s_{12} a_1 + s_2^2 a_2 + \cdots + s_{2n} a_n = s_{2r} \\ \vdots \\ s_{1n} a_1 + s_{2n} a_2 + \cdots + s_n^2 a_n = s_{nr} \end{cases} \quad (3.2)$$

ここで，$s_i^2 (1 \leq i \leq n)$は$E_{i,T}$の分散，s_{ij}は$E_{i,T}$と$E_{j,T}$の共分散，s_{ir}は$E_{i,T}$とR_Tの共分散の表しており，s_i^2は式(3.3)，s_{ij}は式(3.4)，s_{ir}は式(3.5)によりそれぞれ求められる．

$$s_i^2 = \frac{1}{T} \sum_{t=1}^{T} \left(e_{i,t} - \bar{E}_{i,T} \right)^2 \quad (3.3)$$

$$\left(\bar{E}_{i,T} = \frac{1}{T} \sum_{t=1}^{T} e_{i,t} \right)$$

$$s_{ij} = \frac{1}{T} \sum_{t=1}^{T} \left(e_{i,t} - \bar{E}_{i,T} \right) \left(e_{j,t} - \bar{E}_{j,T} \right) \quad (3.4)$$

$$\left(\bar{E}_{i,T} = \frac{1}{T} \sum_{t=1}^{T} e_{i,t} \ , \quad \bar{E}_{j,T} = \frac{1}{T} \sum_{t=1}^{T} e_{j,t} \right)$$

$$s_{ir} = \frac{1}{T} \sum_{t=1}^{T} \left(e_{i,t} - \bar{E}_{i,T} \right) \left(r_t - \bar{R}_T \right) \quad (3.5)$$

$$\left(\bar{E}_{i,T} = \frac{1}{T} \sum_{t=1}^{T} e_{i,t} \ , \quad \bar{R}_T = \bar{R}_{t-1} + \alpha_r (r_t - \bar{R}_{t-1}) \right)$$

$\bar{E}_{i,T} (\bar{E}_{j,T})$は$E_{i,T} (E_{j,T})$の標本平均，$\bar{R}_T$は$R_T$の加重平均を表している．以上の計算を行う事により，算出された回帰係数a_iを各センサの重要度として利用し，これを基に各センサの状態数を変更する．

3.3　センサの重要度からセンサの状態数を変更する方法

本節では，図 3.1「重要利用部」にてセンサの状態数を変更する方法を述べる．前提として全体で n 個の状態を 1 刻みが r のセンサを考える(図 3.3)．強化学習では，1 刻みを 1 つの状態として認識する．本研究では，この状態を単位状態として定義する．この単位状態を統合する事により各センサの状態数を変更する．

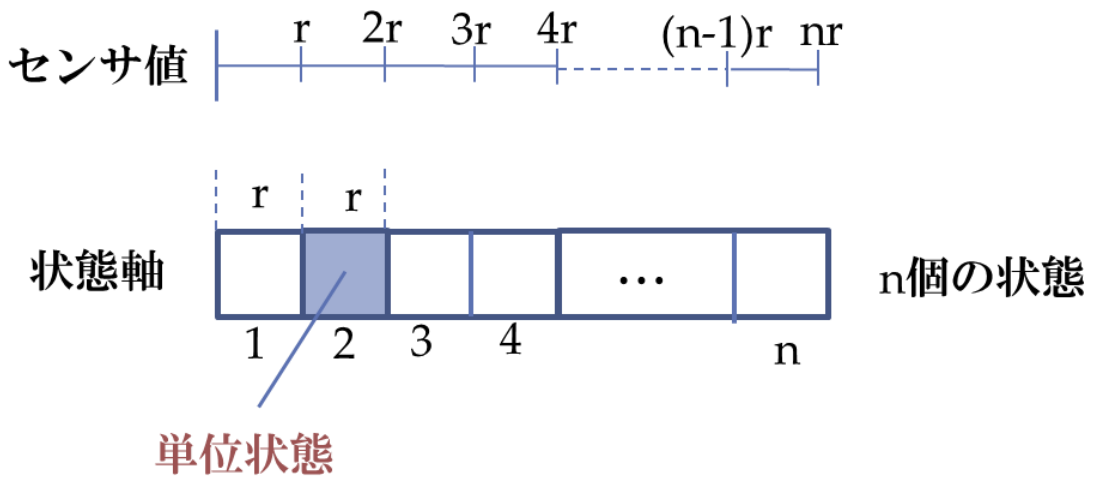


図 3.3　n 個の状態を持つセンサ

最初に，前節の「3.2 センサの重要度を算出する方法」にて述べた方法により算出した各センサの重要度a_iから，式(3.6)により各センサの状態数v_i(1 ≤ i ≤ n)を算出する．

$$v_i = \begin{cases} v_{i\_min} & (|a_i| < m_\alpha) \\ \left\lceil \dfrac{v_{i\_max} - v_{i\_min}}{m_\beta - m_\alpha} |a_i| + \dfrac{m_\beta v_{i\_min} - m_\alpha v_{i\_max}}{m_\beta - m_\alpha} \right\rceil & (m_\alpha \le |a_i| \le m_\beta) \\ v_{i\_max} & (m_\beta < |a_i|) \end{cases} \quad (3.6)$$

ここで，v_{i_min}はセンサ i が表現できる最少の状態数，v_{i_max}はセンサ i が表現できる最大の状態数である．また，m_α，m_βは任意の定数である．式(3.6)をグラフ化した物を図 3.4 に示す．センサの重要度の絶対値$|a_i|$が$|a_i| < m_\alpha$である場合，センサが表現できる最少の状態数 v_{i_min}とし，$m_\beta < |a_i|$である場合，センサが表現できる最大の状態数 v_{i_max}とする．また，$m_\alpha \le |a_i| \le m_\beta$である場合，センサの重要度の絶対値$|a_i|$に応じて比例的に状態数を算出する．

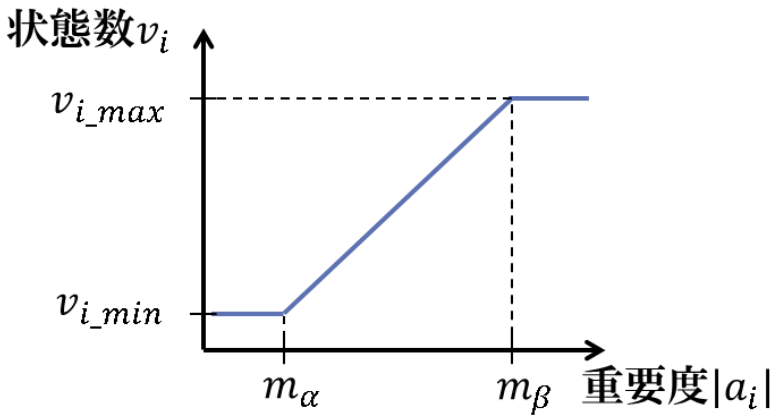


図 3.4　式(3.6)のグラフ化

続いて各センサの状態数v_iから，1 状態に対して統合する単位状態数v_i^* を式(3.7)により求める．算出された統合する単位状態数v_i^* を基に，各センサの状態を統合し状態数を変更した Q 空間を一時的に作成する(図 3.2).

$$v_i^* = \left\lceil \frac{v_{i\_max}}{v_i} \right\rceil \qquad (3.7)$$

作成した一時的な Q 空間に対して，現状態s'の行動a_iに対しての行動価値$Q'(s', a_i)$を式(3.8)から算出する．

$$Q'(s', a_i) = \frac{\sum c_k \sum c_l \cdots \sum c_m \, Q(s_w, a_i) \cdot N(s_w, a_i)}{\sum c_k \sum c_l \cdots \sum c_m \, N(s_w, a_i)} \qquad (3.8)$$

ここで，s_wは統合された現状態s'に含まれる単位状態であり，$Q(s_w, a_i)$は単位状態s_wの行動a_iに対しての行動価値，$N(s_w, a_i)$は$Q(s_w, a_i)$の評価回数を表している．式(3.8)は統合された現状態s'に含まれる単位状態s_wの行動価値に対して，評価回数を重みとした加重平均式である．行動価値$Q'(s', a_i)$の算出は選択できる行動すべてに対して行う．その後，式(3.8)で算出された行動価値$Q'(s', a_i)$を基に行動選択を行う．

3.4　提案手法の処理手順

本章の「3.2 センサの重要度を算出する方法」，「3.3 センサの重要度からセンサの状態数を変更する方法」で述べた事を踏まえ，提案手法の具体的な処理の流れについて述べる．前提として，エージェントが n 個のセンサを搭載している場合を考える．以下に処理手順を示す.

1. 時刻 T においてエージェントは現状態s_Tを認識する．

2. 時刻 T>1 の時，各センサの重要度を報酬を目的変数，各センサの出力値を説明変数とした重回帰分析により求める．時刻 T までに得たデータから式(3.2)の連立方程式を解くことで回帰係数a_iを算出し各センサの重要度として利用する．

3. 算出した各センサの重要度a_iに応じて，式(3.6)により各センサの状態数v_iを求める．

4. 各センサの状態数v_iを基に，式(3.7)から 1 状態に対して統合する単位状態数v_i^* を求める．求めたv_i^* から状態を統合し状態数を変更した Q 空間を一時的に作成する(図 3.2)．

5. 作成した一時的な Q 空間上での状態統合後の現状態s'に対して，行動価値$Q'(s', a_i)$を式(3.8)から求める．

6. エージェントは行動価値$Q'(s', a_i)$を基に行動選択を行い，次時刻 $T+1$ において状態s_{T+1}に遷移し即時報酬r_{t+1}を受け取る．また，このとき各センサの出力値$e_{1,T+1}, e_{2,T+1}, \cdots, e_{n,T+1}$と即時報酬$r_{t+1}$を重要度を算出するデータとして記憶する．

7. エージェントは行動学習手法に従い，状態s_Tにおける行動a_Tの行動価値$Q\ (s_T, a_T)$を算出する．また，行動価値$Q\ (s_T, a_T)$の評価回数$N(s_T, a_T)$の更新も行う．

8. 時刻 $T+1$ を時刻 T に改める．

9. タスクが終了するまで，手順 1～8 を繰り返す．

以上が提案手法の処理手順であり，エージェントはこの処理を繰り返すことで各センサの重要度の算出と行動の学習を並行して行う．

第 4 章　実験

本章では，第 3 章で説明した提案手法の有効性を検証する実験について述べる．検証実験は仮想環境，実環境で行いそれぞれ提案手法を適応したエージェントと一般的な強化学習を適応したエージェントの性能を比較する．第 4. 1 節，第 4.2 節では仮想環境におけるシミュレーション実験について述べ，続く第 4.3 節では実環境における実機実験について述べる．

4.1　仮想環境における検証実験 1

4.1.1　実験目的

本実験は提案手法の有効性を検証するための実験である．提案手法を適用したエージェントと一般的な強化学習を適用したエージェントに対して，シミュレーション上で比較実験を行い，実験結果を考察する事で提案手法の有効性を示す．

4.1.2　実験設定

シミュレーション実験を行う環境を図 4.1 に示す．四方を壁に囲まれた，10×10 の計 100 個の状態を持つ空間である．前方の壁を壁 A とし，左の壁を壁 B とする． この環境でエージェントは「前方の壁 A の近傍に到達する」というタスクを行う．「ス」がスタート地点，「ゴール」がゴール地点を表しており，スタートからゴールに到達するまでを 1 試行とする． エージェントはゴール到達後，スタート地点に戻る．

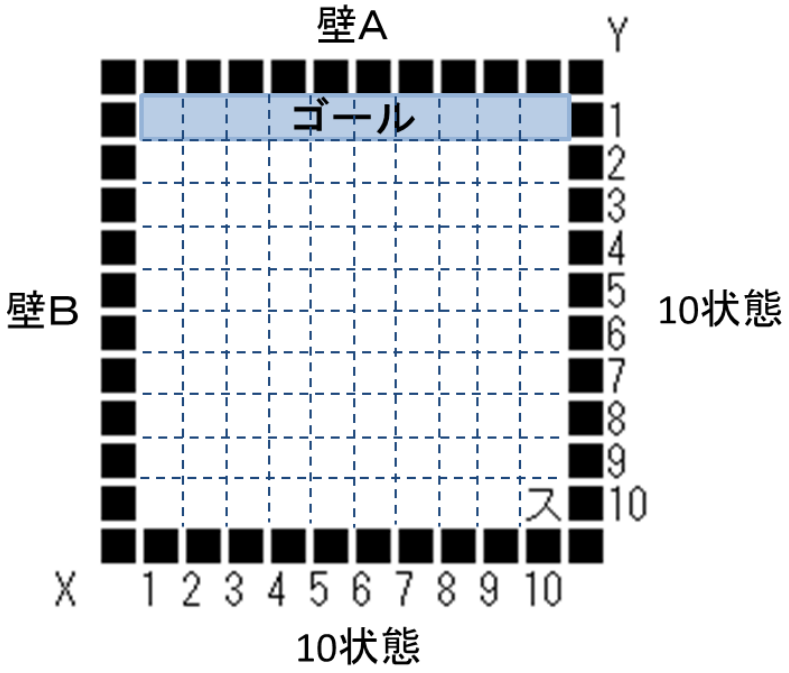


図 4.1　シミュレーション実験の環境 1

エージェントの設定を図 4.2 に示す. エージェントは前方の壁 A との距離を測るセンサ A と左の壁 B との距離を測るセンサ B を搭載している．各距離センサは対象となる壁までのマス数を距離として測定する．エージェントはセンサ A とセンサ B により現在の状態を認識する．各線センサの最大の状態数は「10」，最少の状態数は「1」である．エージェントがタスクを達成するためには，壁 A との距離がわかれば十分であり，壁 B との距離に関しては必要ではない．よって，壁 A との距離を測るセンサ A の重要度が高くなり，壁 B との距離を測るセンサ B の重要度が低くなると考えられる．また，エージェントは，前後左右斜めの 8 方向のいずれかへ移動と停止の計 9 種類の行動を選択できる．1 行動で 1 マス移動し，移動先が壁である場合はその場に留まる．スタートからゴールまでの最低行動回数は「9」回である．

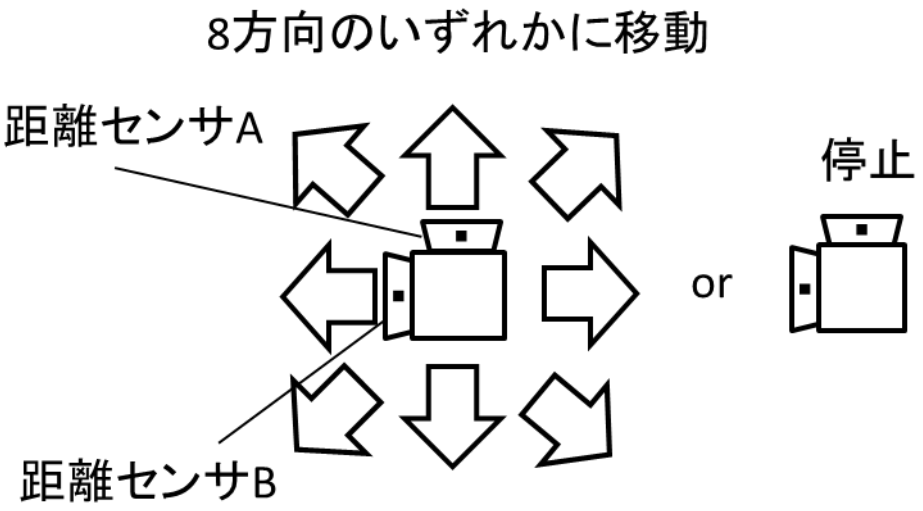


図 4.2　エージェントの設定

続いて報酬設定について説明する．エージェントは即時報酬として，式(4.1)を毎時刻受け取る．ここで，d_Aはエージェントと壁 A との距離を表している．エージェントは壁 A に近づけば近づく程高い報酬を受け取る事ができ，最低報酬は図 4.1 の Y=10 の地点で「1」，最高報酬はゴール地点で「10」となっている．

$$\mathbf{r} = \mathbf{10} - \boldsymbol{d_A} \qquad (\mathbf{4.1})$$

エージェントの行動学習手法は加重平均手法，行動選択手法は ε -greedy 法を使用する．それぞれの手法に関しては，「2.2 行動学習手法」及び「2.3 行動選択手法」にて説明している．

最後に，実験パラメータを表 4.1 に示す．以上の実験設定の元，提案手法を適用したエージェントと強化学習を適用したエージェントで実験を行い結果を考察する．

表 4.1 実験パラメータ

項目		内容
総試行回数		10000 回
試行終了条件		図 4.1 ゴールに到達
行動学習手法		加重平均手法
行動選択手法		ε -greedy 法
ε		0.05
α_{RL}		0.1
α_{r}		0.1
Q 値の初期値		0.0
v_{min}		1
v_{max}		10
閾値	m_{α}	0.2
	m_{β}	0.8
初期状態数	センサ A	10
	センサ B	10
初期重要度	センサ A	1.0
	センサ B	1.0

4.1.3 実験結果

各試行終了時における各センサの重要度の推移を図 4.3 に，各センサの状態数の推移を図 4.4 に示す．図 4.3 の横軸は試行回数，縦軸は試行終了時の各センサの重要度を，図 4.4 の横軸は試行回数，縦軸は試行終了時の各センサの状態数を表している．図 4.3 から，タスク達成に重要である壁 A との距離を測るセンサ A の重要度が「1」に，タスク達成に重要ではない壁 B との距離を測るセンサ B の重要度が「0」に収束しているのがわかる．また，図 4.4 から，重要度の高いセンサ A の状態数が多く，重要度の低いセンサ B の状態数が少なくなっているのがわかる．センサ A に関しては，重要度が$m_{\beta} = 0.8$を下回っていないため，状態数は最大である「10」になっている．一方，センサ B に関しては重要度が$m_{\alpha} = 0.2$を上回っていないため，状態数は最少である「1」になっている．

続いて，図 4.5 に提案手法と強化学習の行動回数の推移を示す．横軸が試行回数，縦軸が行動回数を表している．(a)は縦軸が全範囲のグラフ，(b)は(a)の縦軸を[0：250]の範囲に拡大したグラフである．図 4.5 を見ると，強化学習よりも提案手法の方がより少ない試行回数で行動回数が収束し，学習速度が速いことがわかる．

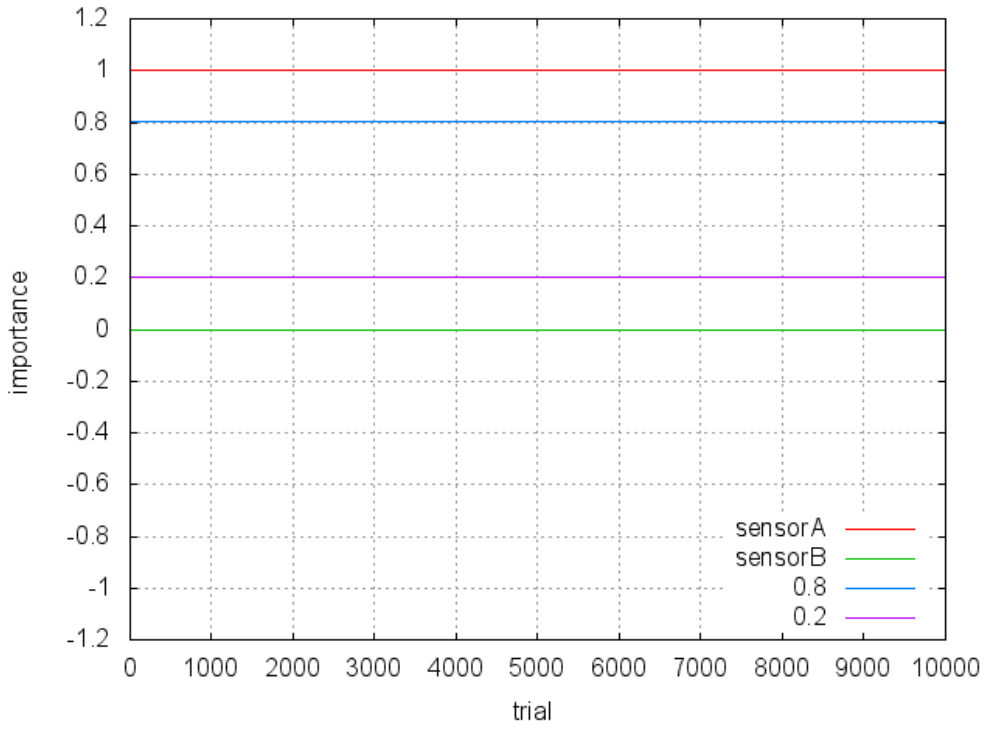


図 4.3　試行終了時における各センサの重要度の推移

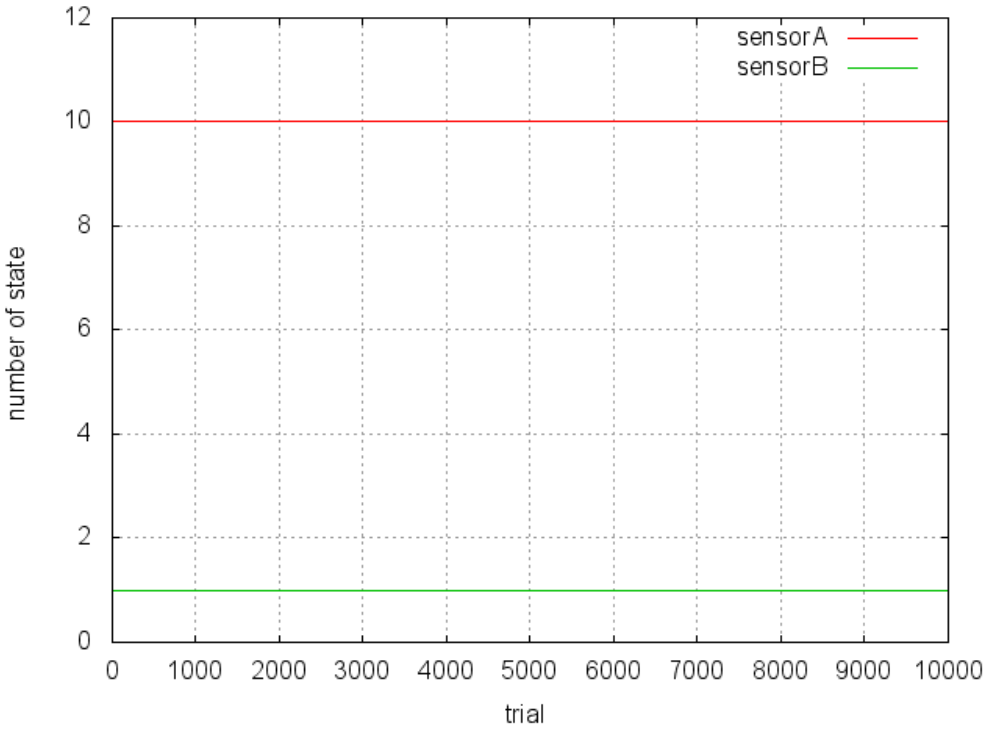
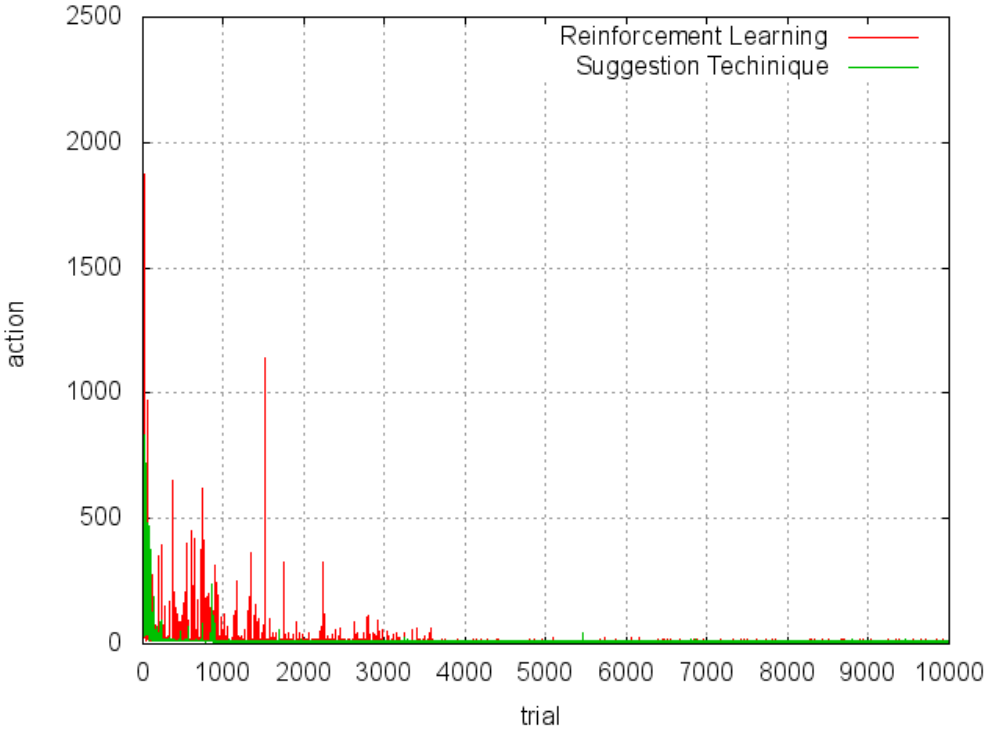
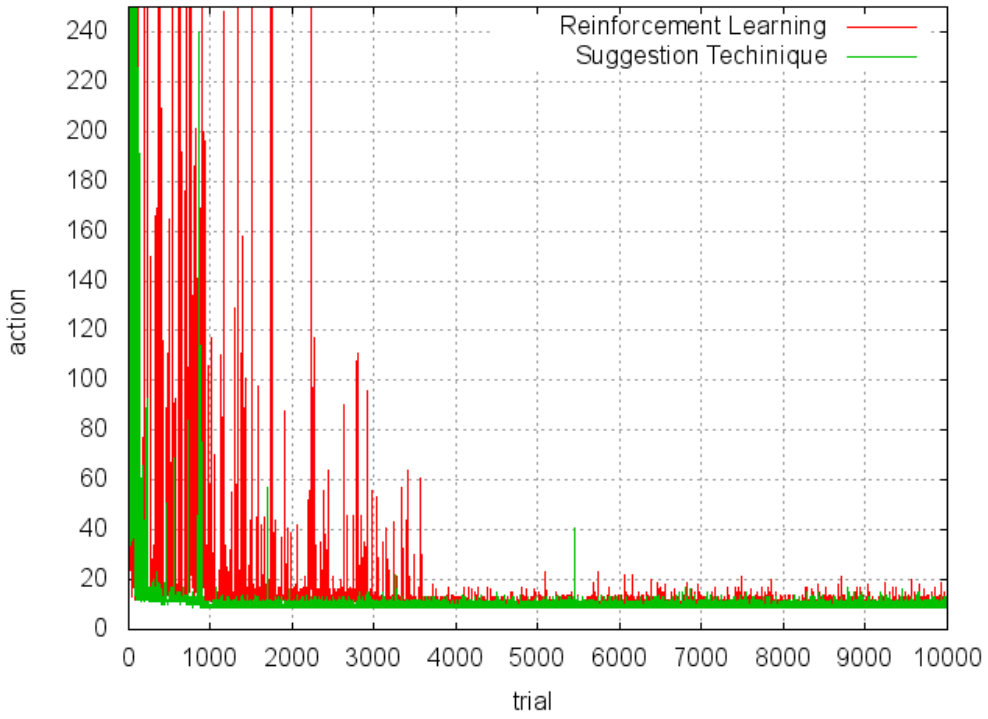


図 4.4　試行終了時における各センサの状態数の推移



(a)全範囲



(b)縦軸[0:250]の範囲拡大

図 4.5　行動回数の比較グラフ

4.1.4　考察

本実験の設定では搭載されているセンサの内，壁 A との距離を測るセンサAが重要であり，壁 B との距離を測るセンサ B は重要ではない．実際に結果として，センサ A の重要度は「1」に，センサ B の重要度は「0」に収束していた．よって，センサの重要度の算出は問題無く行えていると考える．

また，提案手法は強化学習よりも少ない試行回数で行動回数が収束し学習速度が速かった．これは，提案手法ではセンサの重要度を考慮した学習空間が構成できたためであると考える．一般的な強化学習ではセンサ A に関して「10」状態，センサ B に関して「10」の状態数が「10×10」の学習空間で行動を学習している．一方，提案手法ではタスク達成に重要であるセンサ A に関して「10」状態，重要ではないセンサ B に関して「1」状態の状態数が「10×1」の学習空間で行動を学習している．よって，提案手法では重要度の低いセンサの状態数が減り学習空間が縮小した事で学習時間が速くなったと考える．

4.2　仮想環境における検証実験 2

4.2.1　実験設定

本実験は「4.1 仮想環境における検証実験 1」にて行った実験に対し，タスク設定と報酬設定を変更した実験である．そのため，実験目的及びエージェントの設定等に関しては割愛する．実験パラメータに関しては表 4.1 の値を使用している．

本実験を行う環境を図 4.6 に示す．前方の壁を壁 A とし，左の壁を壁 B とする． この環境でエージェントは「壁 A と壁 B の両方の近傍に到達する」というタスクを行う．「ス」がスタート地点，「ゴール」がゴール地点を表しており，スタートからゴールに到達するまでを 1 試行とする． エージェントはゴール到達後，スタート地点に戻る．スタートからゴールまでの最低行動回数は「9」回である．エージェントがタスク達成するには，壁 A との距離を測るセンサ A と壁 B との距離を測るセンサの両方が必要である．

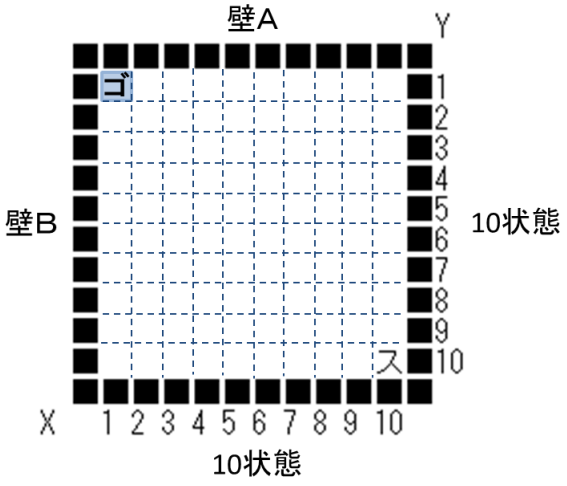


図 4.6　実験環境 2

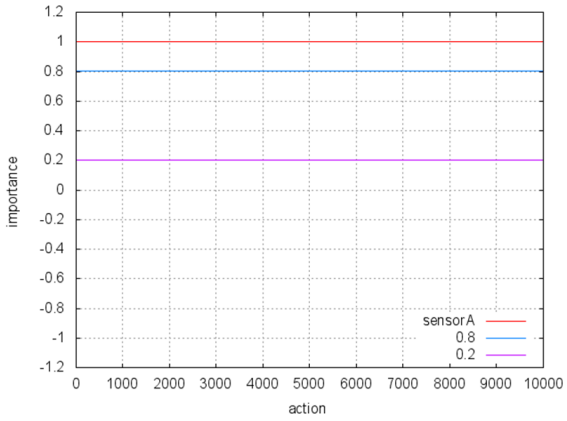
続いて報酬設定について説明する．エージェントは即時報酬として，式(4.2)を毎時刻受け取る．ここでd_Aはエージェントと壁 A との距離を，d_Bはエージェントと壁 B との距離を表している．最低報酬は図 4.6 のスタート地点で「2」，最高報酬はゴール地点で「20」となっている．

$$\mathbf{r} = (\mathbf{10} - \boldsymbol{d_A}) + (\mathbf{10} - \boldsymbol{d_B}) \qquad (\mathbf{4.2})$$

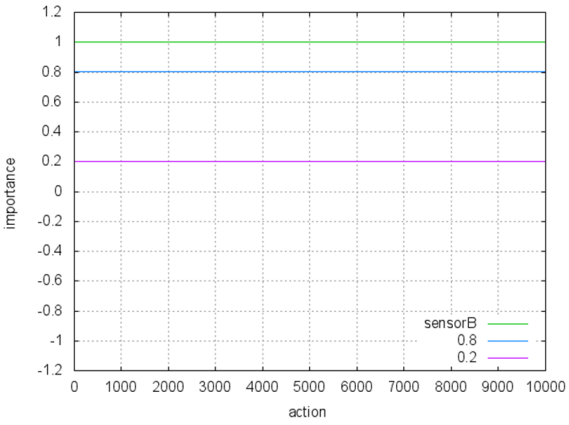
4.2.2 実験結果

各試行終了時における各センサの重要度の推移を図 4.7 に，各センサの状態数の推移を図 4.8 に示す．それぞれ，(a)がセンサＡ，(b)がセンサＢに関してである．図 4.7 の横軸は試行回数，縦軸は試行終了時の各センサの重要度を，図 4.8 の横軸は試行回数，縦軸は試行終了時の各センサの状態数を表している．図 4.7 から，タスク達成に重要である壁 A との距離を測るセンサ A と壁 B との距離を測るセンサ B の両方の重要度が「1」に収束しているのがわかる．また，図 4.8 から，センサ A とセンサ B の両方の状態数が，重要度が$m_\beta = 0.8$を下回っていないため，状態数は最大である「10」になっている．

続いて，図 4.9 に提案手法と強化学習の行動回数の推移を示す．横軸が試行回数，縦軸が行動回数を表している．(a)は縦軸が全範囲のグラフ，(b)は(a)の縦軸を[0：250]の範囲に拡大したグラフである．図 4.9 を見ると，一般的な強化学習と提案手法の行動回数は同程度になっているのがわかる．

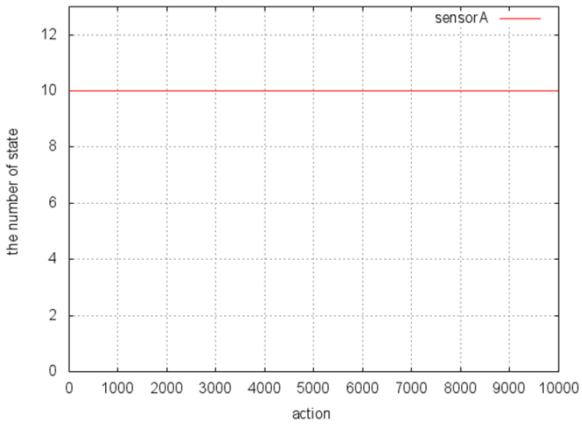


(a)センサＡの重要度の推移

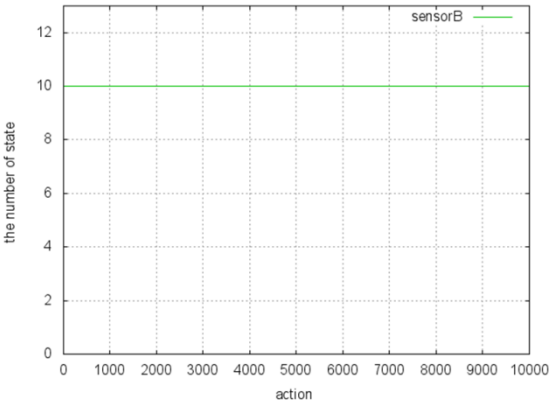


(a)センサＢの重要度の推移

図 4.7　各試行終了時における各センサの重要度の推移

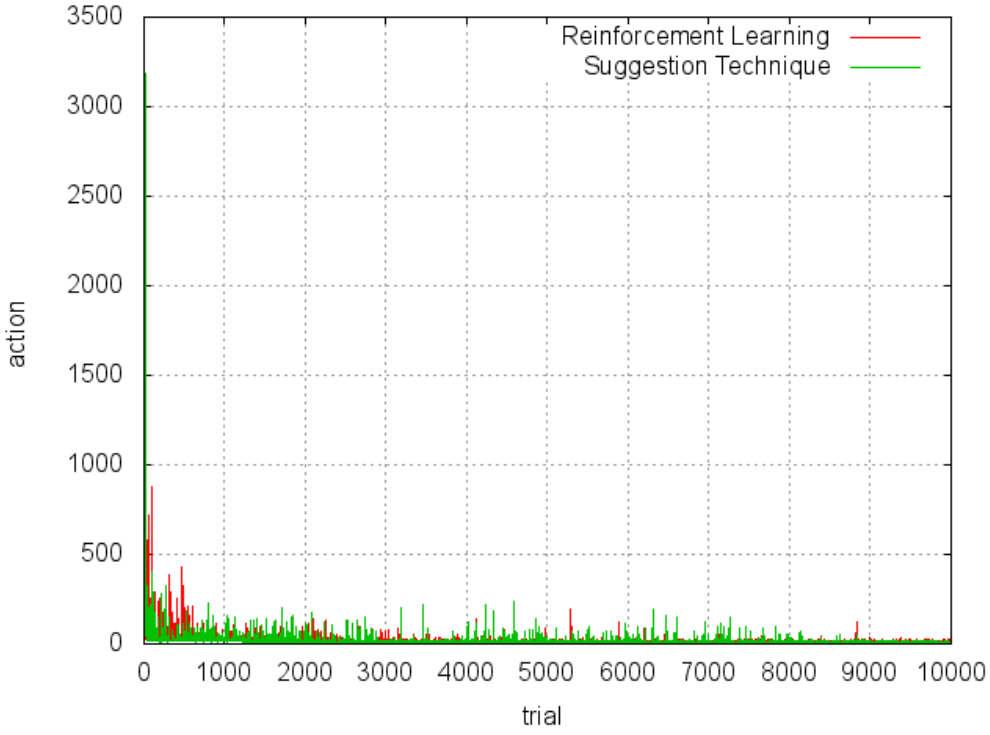


(a)センサＡの状態数の推移

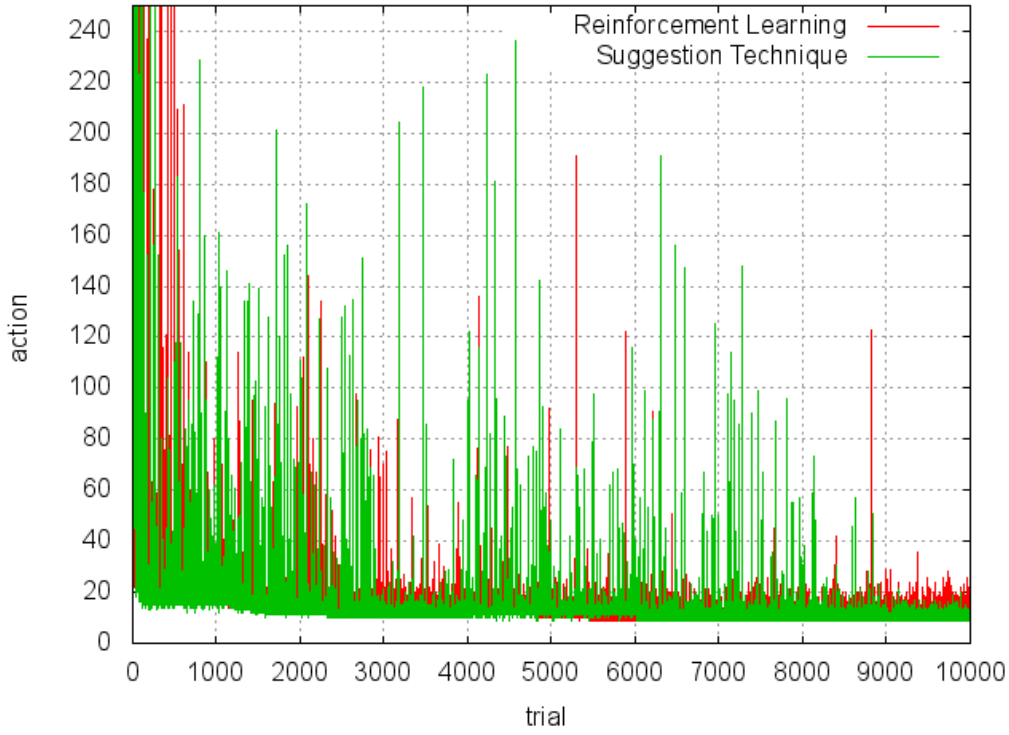


(a)センサＢの状態数の推移

図 4.8　各試行終了時における各センサの状態数の推移



(a) 全範囲



(b)縦軸[0:250]の範囲拡大

図 4.9　各試行における行動回数の比較グラフ

4.2.3 考察

本実験の設定では搭載されているセンサの内, 壁 A との距離を測るセンサAt と壁 B との距離を測るセンサ B の両方が重要である．実際に結果として，どちらのセンサも重要度は「1」に収束していた．よって，搭載されているが全て重要である場合もセンサの重要度の算出は問題無く行えていると考える．

また，提案手法と強化学習の行動回数は同程度の物となっていた．よって，提案手法は最低限，一般的な強化学習と同等の学習速度はあると考えられる．

4.3 実環境における検証実験

4.3.1 実験目的

本実験は，提案手法の有効性を検証する実験である．実際のロボットを使用した実験を行い，センサにノイズ等が発生し状態の誤認識やセンサの重要度の算出に影響が出る実環境においても有効的に機能するか検証する．シミュレーション実験と同様にロボットに提案手法と強化学習をそれぞれ適用し結果を考察する．

4.3.2 使用するロボット

本実験で使用するロボットを図 4.10 に示す．また，ロボットの寸法を図 4.11 に示す．(a)が正面からの寸法，(b)が上面からの寸法である．ロボットの寸法は正面から見て，高さ 262[mm]，幅 400[mm]，奥行き 400[mm]となっている．下部のオムニホイールにより，前後左右斜めの方向に移動可能である．また，赤外線センサ「GP2Y0A21YK0F」を前後左右の側面に搭載していて，周囲の物体に対しての距離を計測できる [14]．赤外線センサ「GP2Y0A21YK0F」の電圧と距離の関係を図 4.12(a)に示す．これは文献[14]に掲載されているグラフであるが，電圧と距離の関係式が明示されておらず，グラフによる概形が示されているだけである. そのため, 本研究では式(4.3)により, 電圧と距離の関係を近似する. 式(4.3)のグラフ化した物を図 4.12(b)に示す．図 4.12 の(a)と(b)のグラフでは，0～5[cm]の範囲の形状が異なっているが，センサの取り付け位置からロボットの端部までに 10[cm]の幅があり，0～5[cm]の範囲の計測は行われないため問題はないと考える．

$$\boldsymbol{f(x) = \frac{32}{x + 4}} \qquad (\boldsymbol{4.3})$$

またロボットは，プログラムを実行するプラットフォームとして Armadillo-300 を，モータ駆動やセンサデータの取得に Arduino UNO を搭載している[15][16]．Armdillo-300 と Arduino UNO は双方向にシリアル通信を行い，強化学習における行動選択や状態認識に伴いモータ駆動の指令やセンサデータ等を転送する．このロボットを使用し実環境における検証実験を行う．

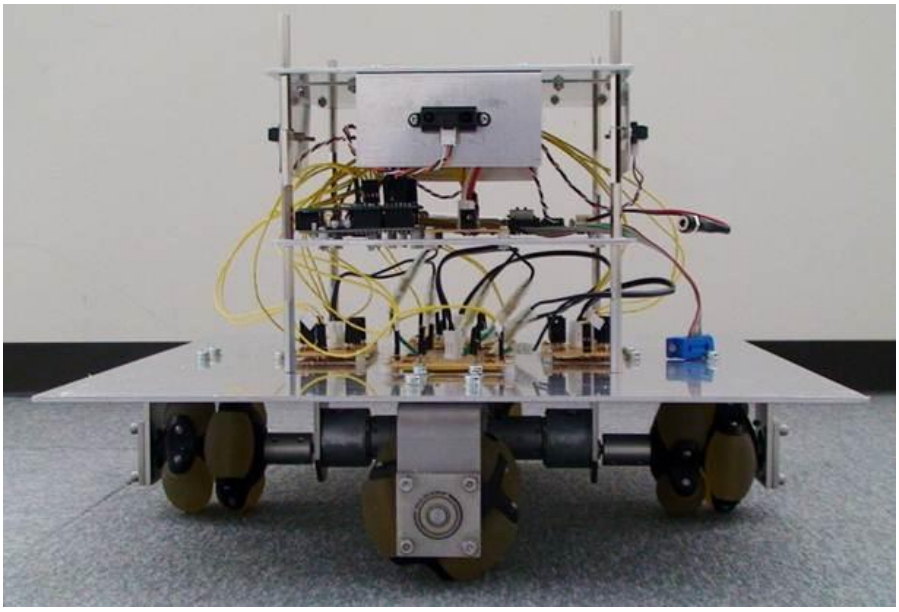
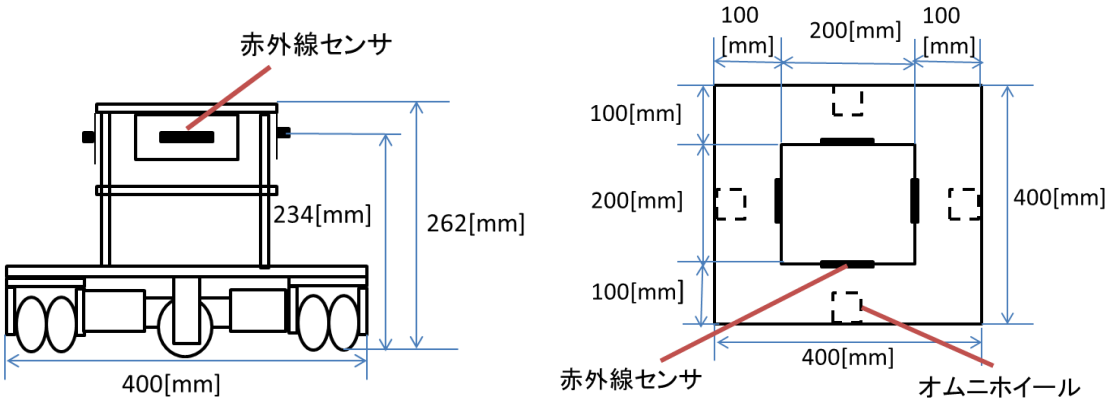
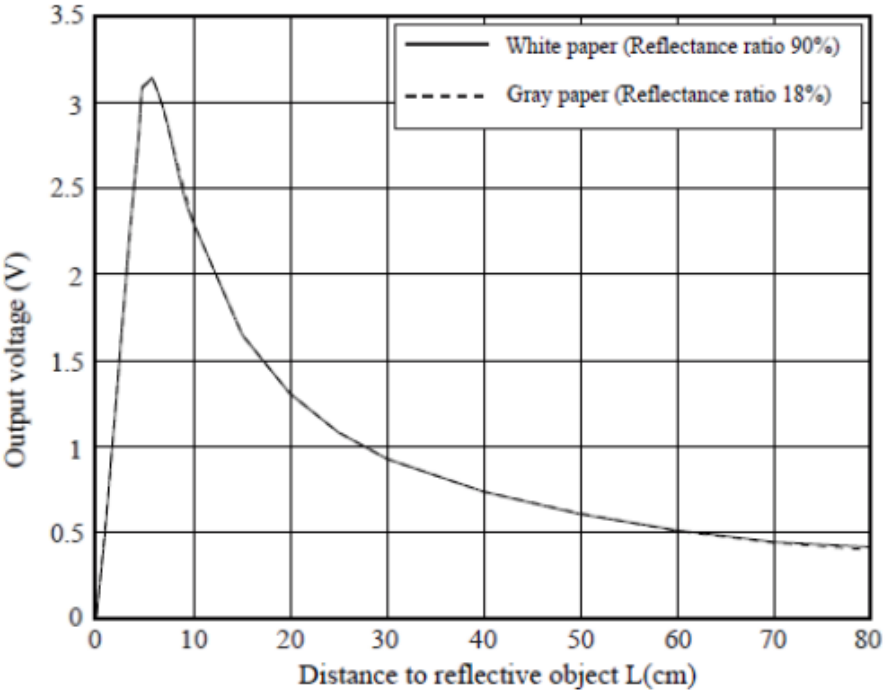


図 4.10　使用するロボット

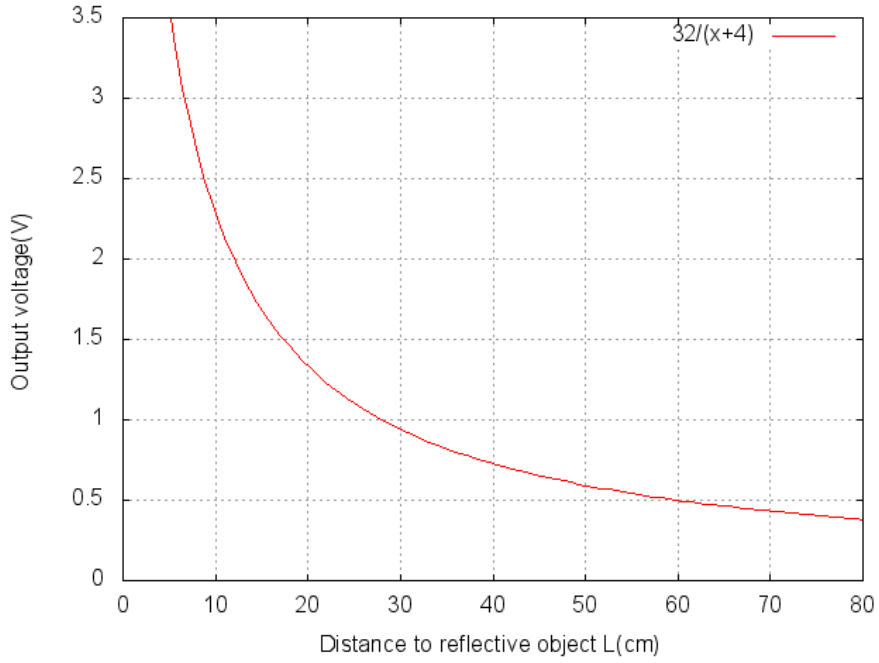


(a)正面図　　(b)上面図

図 4.11　ロボットの寸法



(a) 電圧と距離の関係



(b) 近似式$f(x) = \frac{32}{x+4}$

図 4.12　赤外線センサの電圧と距離の関係および近似式

4.3.3　実験設定

本実験は「4.1 仮想環境における検証実験」にて述べたシミュレーション実験と同様の実験を行い提案手法の性能を検証する．

最初に，実験を行う環境を図 4.13 に示す．四方を壁に囲まれた空間で前方の壁を壁A，左の壁を壁 B とする．壁には「スタイロフォーム IB」という製品を使用している[17]．実験環境の寸法を図 4.14 に示す．(a)が上面から見た寸法図，(b)が側面から見た実験環境とロボットの寸法図になっている． 1 辺が 1100[mm]であり，壁と壁は蝶番とネジで連結している．この実験環境に対してロボットは「前方の壁 A の近傍に到達する」というタスクを行う．「ス」がスタート地点，「ゴール」がゴール地点を表しており，スタートからゴールに到達するまでを 1 試行とする．

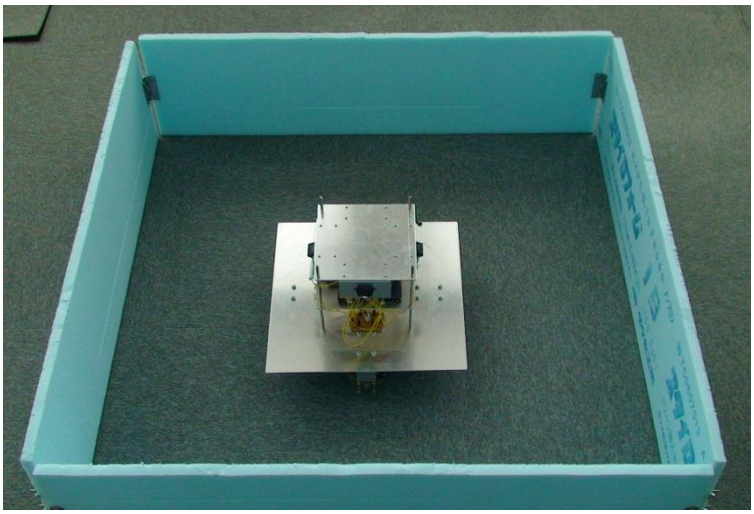
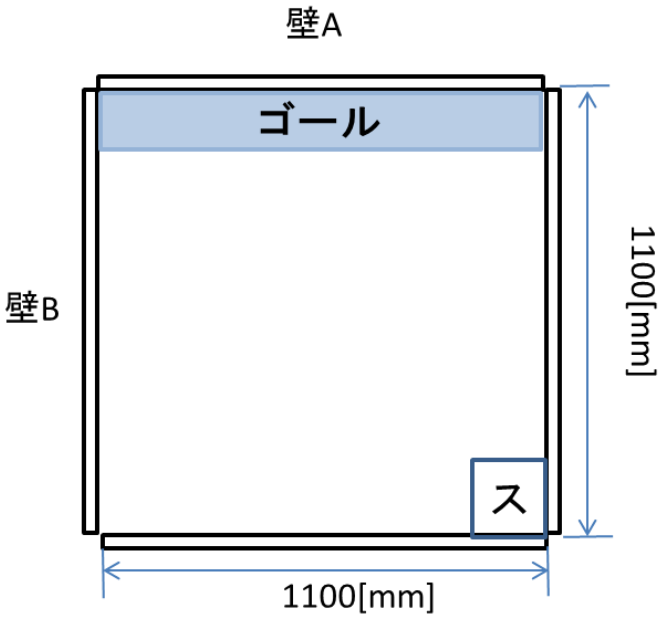
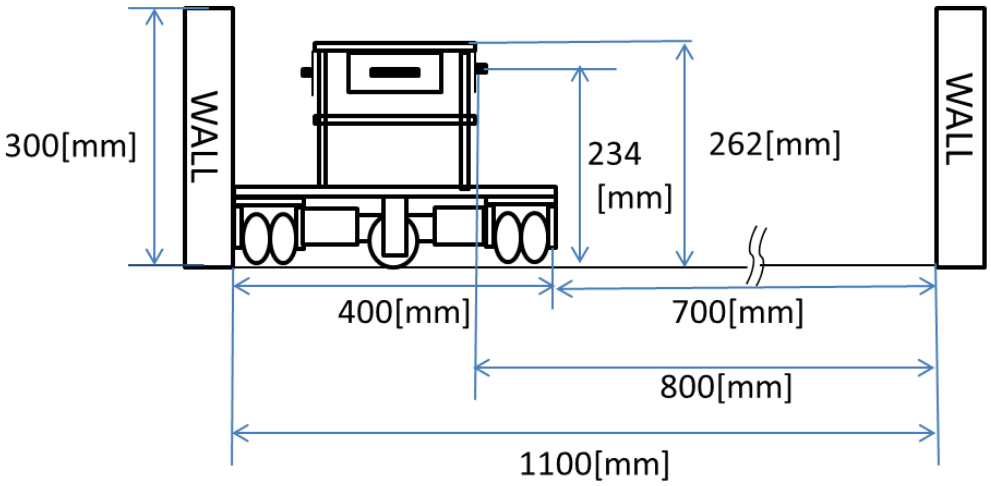


図 4.13　実験環境



(a) 上面から見た実験環境



(b) 側面から見たロボットと実験環境

図 4.14 実験環境の寸法

続いてロボットの設定について説明する．ロボットは搭載されている赤外線センサにより，前方の壁 A との距離と，左の壁 B との距離を測定し状態を認識する．壁 A との距離を測るセンサをセンサ A とし，壁 B との距離を測るセンサをセンサ B とする．状態は図 4.15 に示す通りに定義する．壁との距離を 70[mm]間隔に区切り，1 区間を 1 つの状態として認識する．このとき，壁 A であれば，壁から順に昇順に$0 \leq u_A \leq 10$ を割り振り，変数u_Aにより壁 A に対しての状態を認識する．壁 B に関しても同様の設定で状態を認識し，変数u_A, u_Bの組み合わせで状態を定義する．状態数は 11×11 の計 121 個となる．また，ロボットは前後

左右斜めのいずれかに移動と停止の計 9 種類の行動を選択できる．斜め方向への移動は前後左右の行動を組み合わせて行う．移動する場合，設定上では 1 回で 70[mm]移動するが，実際には車輪の空転等により 70[mm]移動するとは限らない．

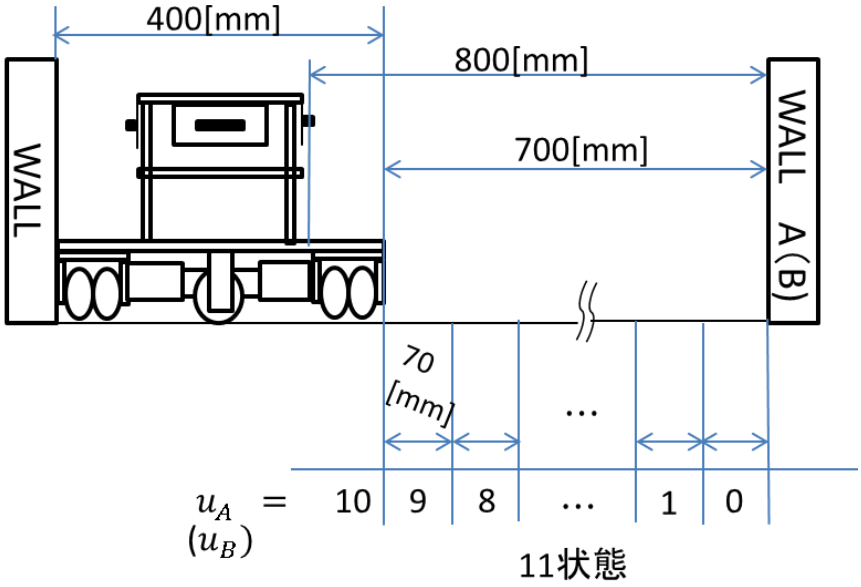


図 4.15 ロボットの状態認識

報酬設定と 1 試行の終了条件について説明する．ロボットは即時報酬として，式(4. 4)を行動毎に受け取る．ここで，d_Aは図 4.15 に示したセンサ A の状態認識と同様の設定で $0 \leq d_A \leq 10$の範囲の値をとり，外部より入力される．センサ A の出力値を基に報酬を決定しない利用として，ロボットが移動する際に車輪の空転等によりロボットが斜め方向を向いてしまい，壁 A との実際の距離を測定できない場合があるためである．また，$d_A = 0$の時，ロボットはゴールに到達したと判定しその試行を終了する．試行終了後ロボットをスタート地点に戻す．

$$\mathbf{r} = \mathbf{11} - \boldsymbol{d_A} \qquad (\mathbf{4.4})$$

最後に，実験パラメータを表 4.2 に示す．以上の設定でロボットに提案手法と強化学習を適用し実験結果を考察する．

表 4.2　実験パラメータ

項目		内容
総試行回数		100 回
試行終了条件		$d_A =$ 0
行動学習手法		加重平均手法
行動選択手法		ε -greedy 法
ε		0.1
α_{RL}		0.5
α_{r}		0.5
Q 値の初期値		0.0
v_{min}		1
v_{max}		11
閾値	m_α	0.2
	m_β	0.8
初期状態数	センサ A	11
	センサ B	11
初期重要度	センサ A	1.0
	センサ B	1.0

4.3.4　実験結果

試行終了時における各センサの重要度の推移を図 4.16 に，各センサの状態数の推移を図 4.17 に示す．図 4.16 の横軸は試行回数，縦軸はセンサの重要度を，図 4.17 の横軸は試行回数，縦軸は状態数を表している．図 4.16 から，タスク達成に重要である壁 A との距離を測るセンサ A の重要度が約「0.85～0.9」に，タスク達成に重要ではない壁 B との距離を測るセンサ B の重要度が「0.1～0.15」に収束しているのがわかる．また，図 4.17 から，重要度の高いセンサ A の状態数が多く，重要度の低いセンサ B の状態数が少なくなっているのがわかる．センサ A に関しては，重要度が$m_\beta = 0.8$を下回っていないため，状態数は最大である「11」になっている．一方，センサ B に関しては重要度が$m_\alpha = 0.2$を上回っていないため，状態数は最少である「1」になっている．

また，図 4.18 に提案手法と強化学習の行動回数の推移を示す．横軸が試行回数，縦軸が行動回数を表している．(a)は縦軸が全範囲のグラフ，(b)は(a)の縦軸を[0：250]の範囲に拡大したグラフである．図 4.18 から，提案手法は強化学習よりも少ない試行回数で行動回数が収束しているのがわかる．

続いて，1 試行目・50 試行目・100 試行目における行動毎の各センサの重要度と状態数の推移を図 4.19，図 4.20，図 4.21 に示す．それぞれ(a)がセンサの重要度の推移を，(b)がセンサの状態数の推移を表したグラフである．(a)の横軸は行動回数，縦軸はセンサの重要度を，

(b)の横軸は行動回数，縦軸はセンサの状態数を表している．図 4.19 から，1 試行目の 50 行動までは, 各センサの重要度は変動しているが, 50 行動経過後からセンサ A に関しては「0.85～0.9」に，センサ B に関しては「0.1～0.15」に収束している．状態数に関しても 50 行動までは重要度の変動に伴い変化しているのがわかる．開始直後，センサの重要度が「1」になっているのは，ロボットがその場で停止していて報酬が一定値を取り重要度が計算できず初期値が残っているためである．また図 4.20，図 4.21 から，50 試行・100 試行では各センサの重要度は収束していて，状態数も変動していないのがわかる．

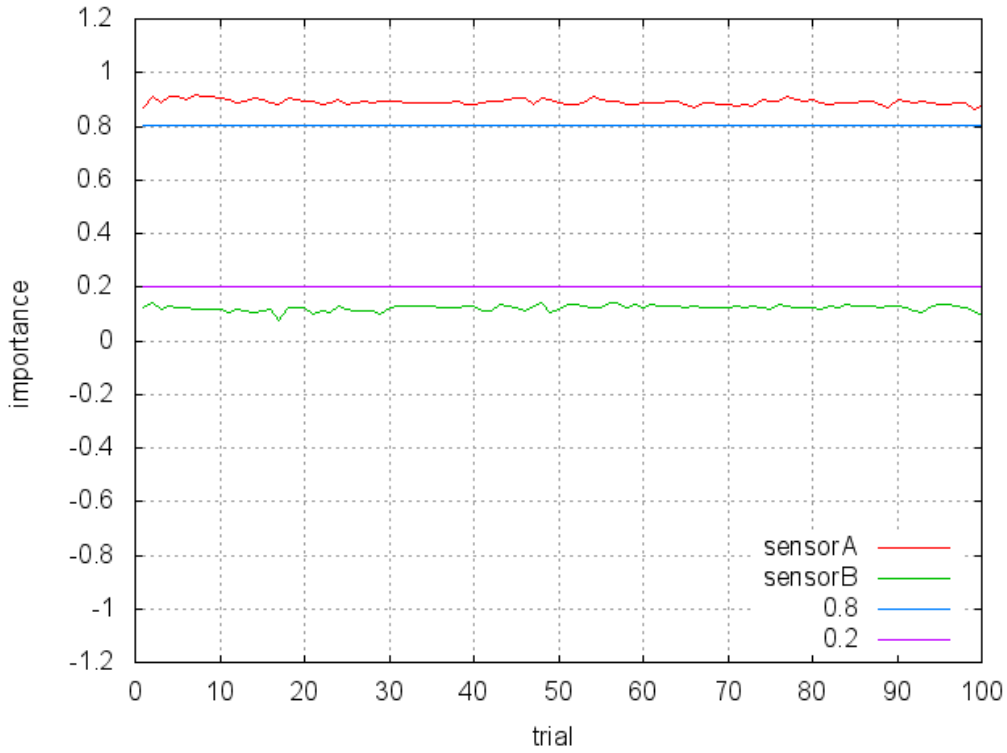


図 4.16　各試行終了時における各センサの重要度の推移

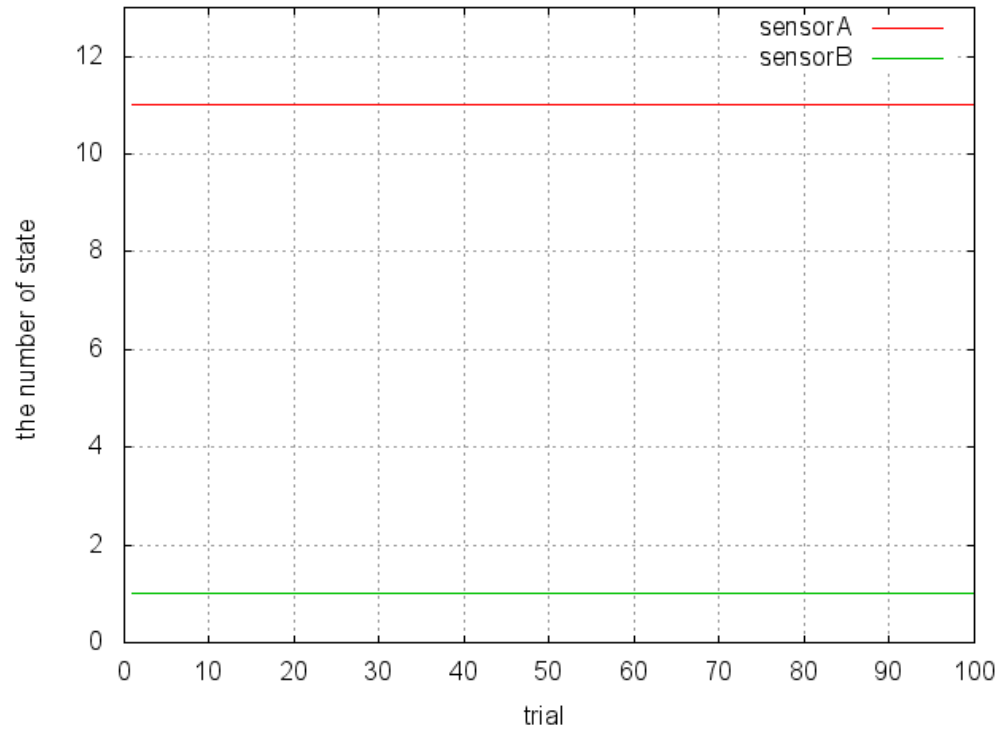
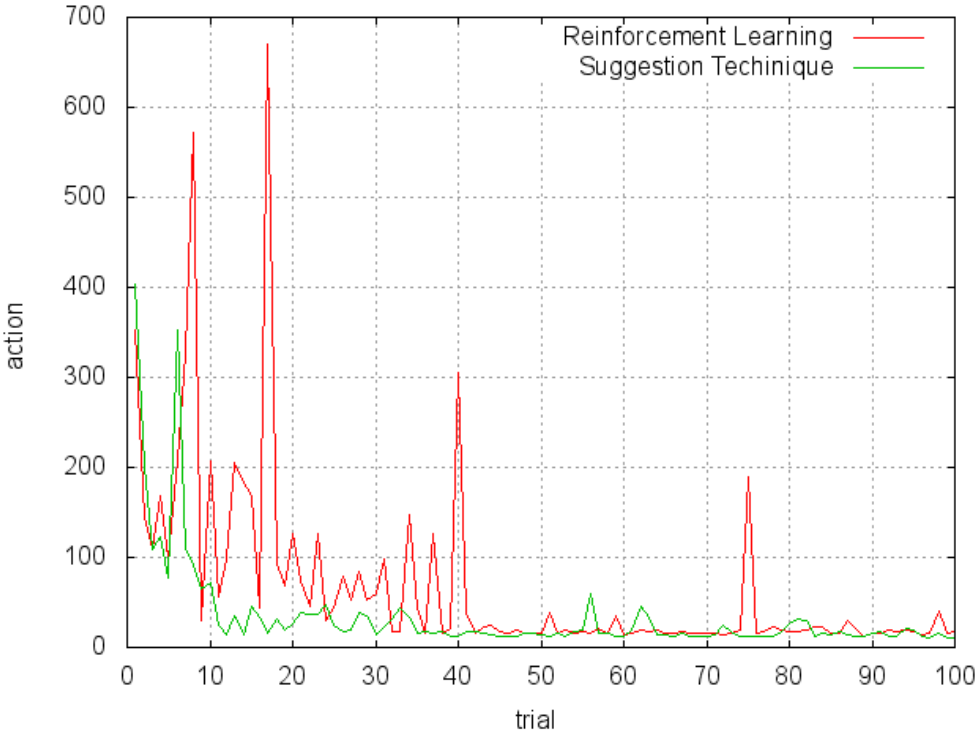
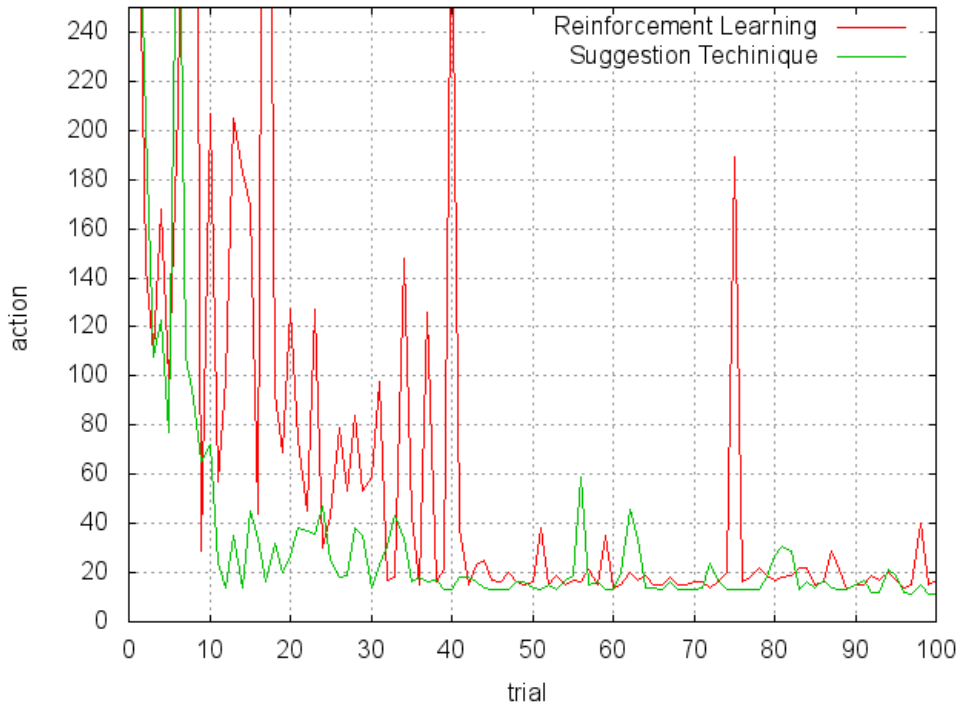


図 4.17　各試行終了時における各センサの状態数の推移

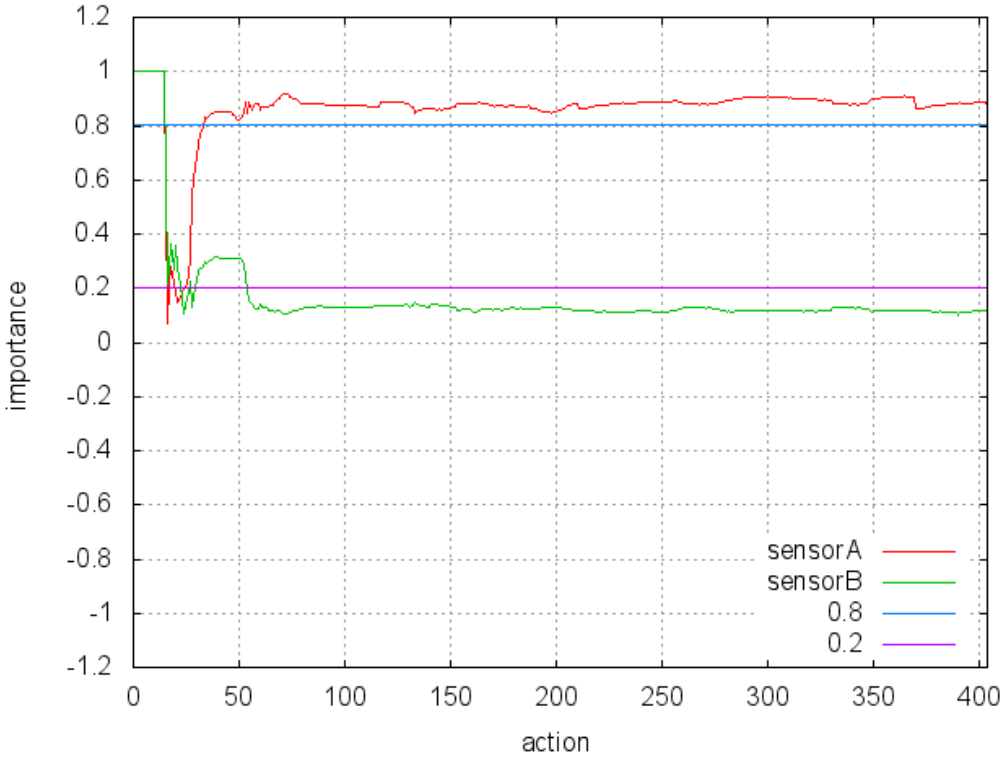


(a) 全範囲

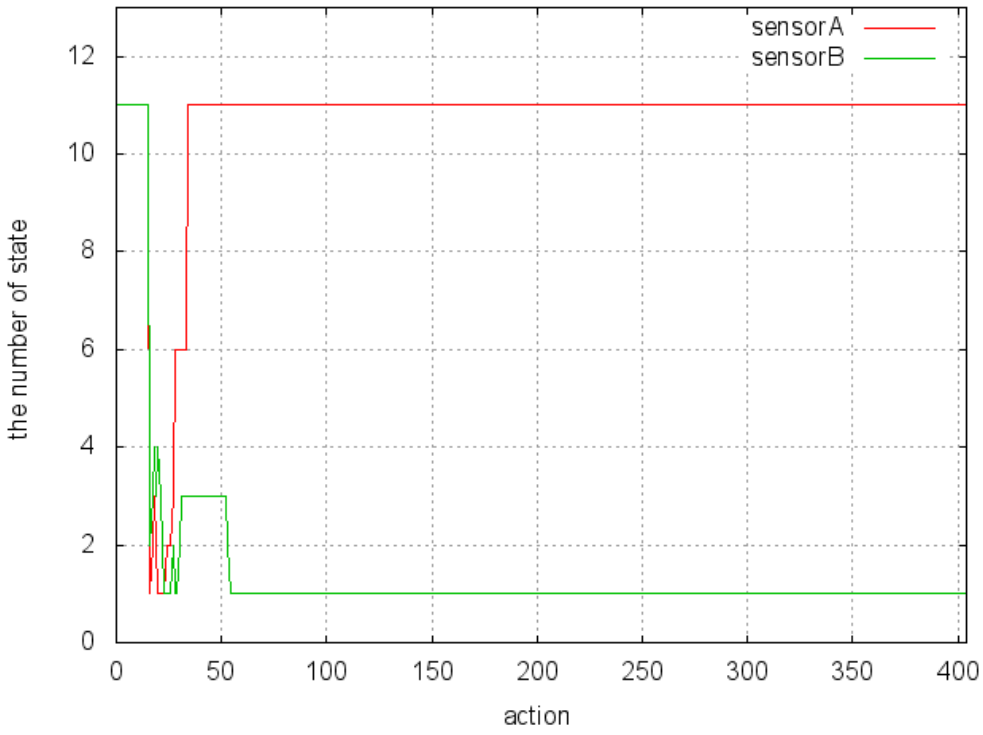


(b)縦軸[0:250]の範囲拡大

図 4.18　各試行における行動回数の比較グラフ

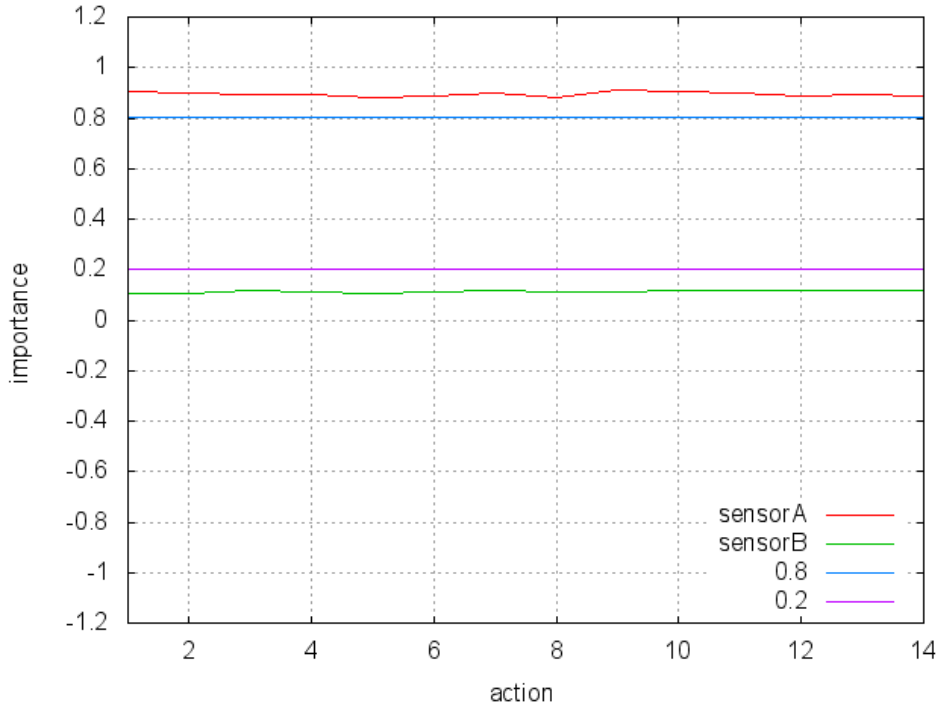


(a) 重要度の推移

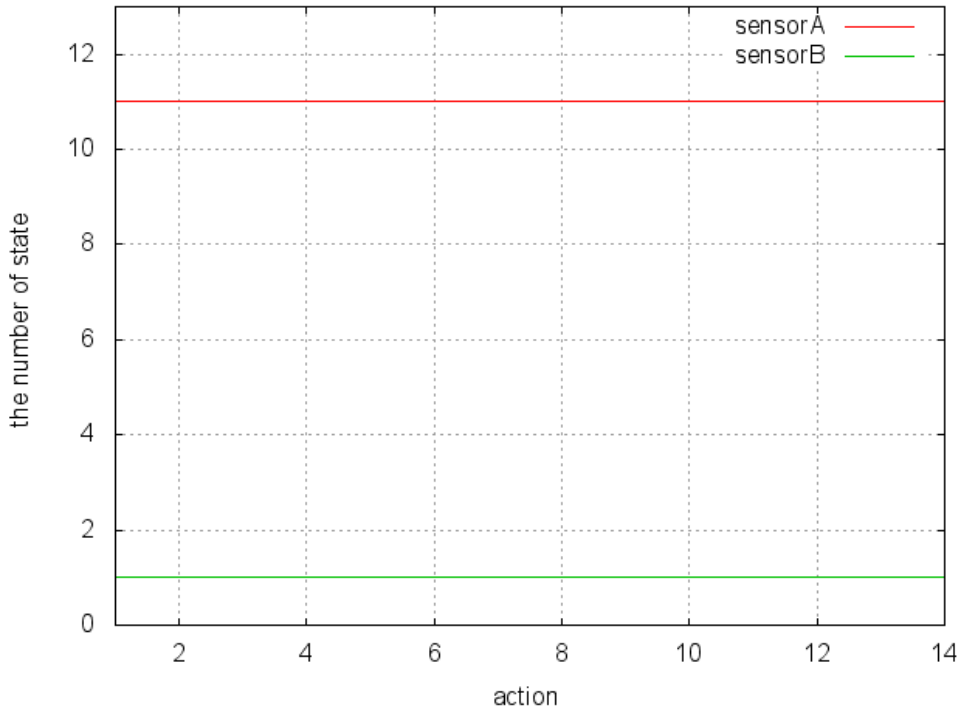


(b)状態数の推移

図 4.19　1 試行目におけるセンサの重要度と状態数の推移

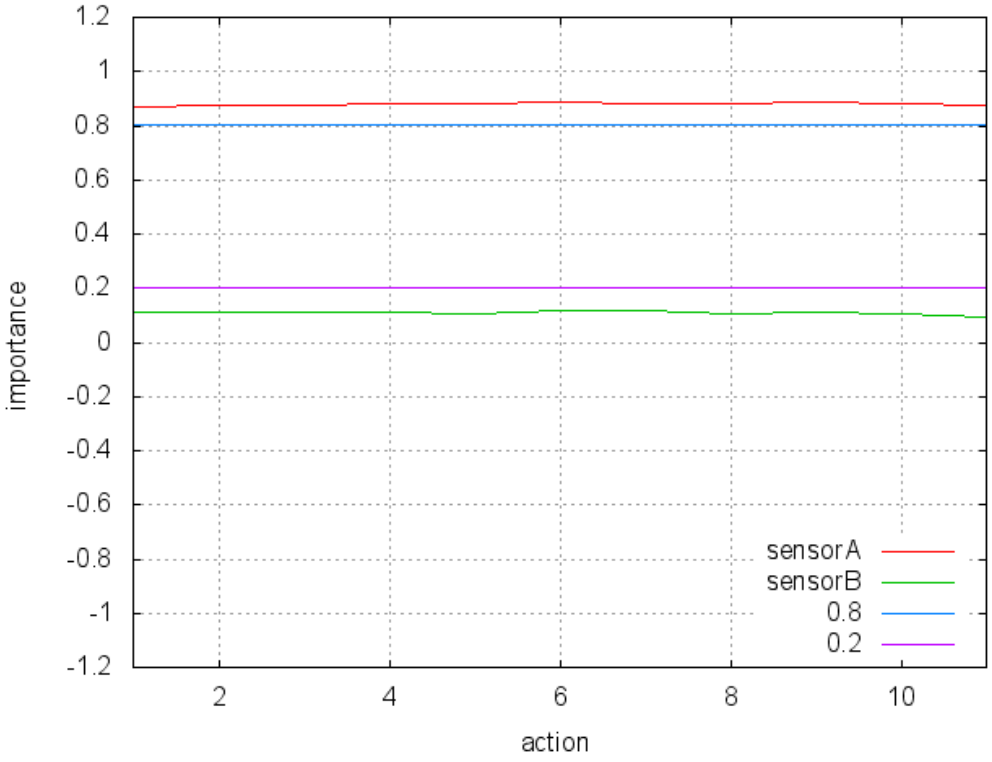


(a) 重要度の推移

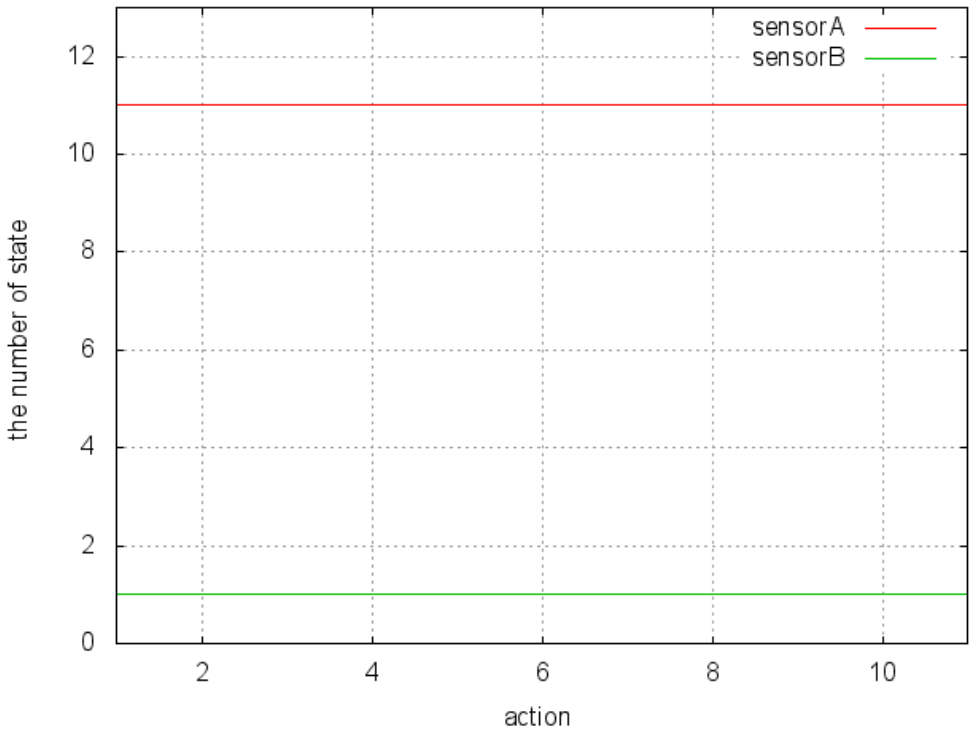


(b)状態数の推移

図 4.20　50 試行目におけるセンサの重要度と状態数の推移



(a)重要度の推移



(b)状態数の推移

図 4.21　100 試行目におけるセンサの重要度と状態数の推移

4.3.5 考察

本実験も先述した「4.1 仮想環境における検証実験 1」と同様に，壁 A との距離を測るセンサ A が重要であり，壁 B との距離を測るセンサ B は重要ではない．実際に結果として 1 試行目の初期からセンサ A の重要度は「0.85〜0.9」に，センサ B の重要度は「0.1〜0.15」に収束していた．よって，センサにノイズ等が発生する実環境においてもセンサの重要度の算出は問題無く行えていると考える．

また，センサの重要度が問題無く算出できているため，タスク達成に重要であるセンサ A に関して「11」状態，タスク達成に重要ではないセンサ B に関して「1」状態の状態数が「11×1」のセンサの重要度を考慮した学習空間を構成できている．よって，提案手法は強化学習よりも少ない試行回数で行動回数が収束したと考える．

第 5 章　結論

本章の第 5.1 節では，本論文全体としてのまとめを述べ，続く第 5.2 節では，これからの課題を述べる．

5.1 本論文のまとめ

本研究では，近年ロボットの制御方法として機械学習の一種である強化学習が期待されている事を述べた．しかし，強化学習には学習空間の構成方法による学習時間増加の問題が存在した．本研究ではこの問題を解決する事を目的とした．問題解決のために，ロボットに搭載されているセンサの重要度の注目した．センサの重要度の違いを考慮し，重要度が高いセンサの状態数を多く，重要度の低いセンサの状態数を少なくした学習空間をロボットが自律的に構成する手法を提案した．

また，提案手法の有効性を検証するために，仮想環境と実環境での検証実験を行った．それぞれ提案手法を適用したエージェントと一般的な強化学習を適用したエージェントの性能を比較した．結果として，仮想環境と実環境の両方の実験で提案手法が一般的な強化学習よりも良い結果を得られた．特に，センサにノイズ等の外乱が発生する実環境において，提案手法はセンサの重要度の違いを考慮した学習空間を構成でき有効的に機能した．よって，この結果により本研究で提案した手法の有効性を示せたと考える．

5.2 これからの課題

5.2.1 遅延報酬環境への適用

本研究で提案した手法は即時報酬環境を想定した物である．よって，遅延報酬環境でロボットを運用する場合，提案手法の適用は難しいと考えられる．そこで，センサの重要度を算出する際に，即時報酬とは別のデータを使用して算出する必要がある．代替データとして行動価値 Q が挙げられる．行動価値 Q は将来獲得できる報酬の期待値を表しているため代替データとして利用できると考えられる．しかし，行動価値 Q はある程度学習が進行しなければ算出する事が出来ない．そのため，センサの重要度の算出に時間がかかってしまうと考えられる．よって，行動価値 Q を利用する場合，Q 空間を作成するタイミング等考え直さなければならない部分が存在する．

5.2.2 複雑な実験設定での検証実験

本研究では，提案手法の有効性を検証するため実験環境内に障害物等が無いオープンスペースの環境で壁に近づくという単純タスクの実験を行った．実験結果として提案手法は強化学習よりも良い結果が得られた．しかし，今回得られた結果だけでは，単純なタスクに対しての有効性しか検証できていない．また，今回行った実験はセンサの数が「2」個の最低限の実験だった．よって，今後は複雑なタスクやロボットに搭載されるセンサが増加

した場合においても提案手法は有効的に機能するかより詳しく検証する必要があると考えられる．実験案として，今回実験を行った環境に対して，障害物等を設置する事や，タスクを途中で変更しロボットに複数のタスクを実行させる事，使用するセンサの数を増やし 3 つ以上のセンサで状態を認識するロボットでの実験等が考えられる．

謝辞

はじめに，本論文を作成するにあたり，日頃より懇切なる御指導を賜りました主指導教員の倉重健太郎先生に，深く感謝の意を表します．また，研究内容について大変貴重な御指導と御助言，御意見を下さいました佐賀聡先生，畑中雅彦先生，本田泰先生に厚く御礼申し上げます．そして，本研究に関して多大な御協力を頂きました木島康隆さんに心より感謝致します．最後に，研究活動において貴重な御助言と御意見を頂きました認知ロボティクス研究室の梅津祐介さん，北山直樹さん，澁谷和さん，杉本大志さん，高泉昇太郎さん，三浦丈典さん，狭間重直さん，平間経太さんに感謝致します．

参考文献

[1] 石川 和良, 青山 元, 関 淳也, 足立 佳儀, 石村 左緒里, 薩見 雄一, 向殿 政男：“清掃ロボットにおける安全技術とコンポーネント”， 日本ロボット学会誌, 2011, Vol.29, No.9, pp.17-20

[2] 成岡 健一, 細田 耕：”筋骨格ヒューマノイドのロコモーション研究”, 日本ロボット学会誌, 2011, Vol.30, No.1, pp.8-13

[3] 竹西 素子：”ビジネスとしての2足歩行プラットフォーム”, 日本ロボット学会誌, 2012, Vol.30, No.4, pp.22-28

[4] 大谷 健一：”原子力設備用ロボット”, 日本ロボット学会誌, 2009, Vol.27, No.3, pp.24-25

[5] 岡本 球夫：” 医療福祉ロボットの開発と安全技術”， 日本ロボット学会誌, 2011, Vol.29, No.9, pp.14-16

[6] 郷古 学, 小林 祐一, 金 天海：”移動ロボットを用いた物体識別のための探索行動の学習”, URL：https://kaigi.org/jsai/webprogram/2012/pdf/42.pdf

[7] 港 隆史, 浅田 稔：“環境変化に適応する移動ロボットの行動獲得”, 日本ロボット学会誌, 2000, Vol.18, No.5, pp.706-712

[8] 山口 明彦, 杉本 徳和, 川人 光男：“回避行動の再利用メカニズムを備えた強化学習手法と多関節ロボットの全身運動学習への応用”, 日本ロボット学会誌, 2009, Vol.27, No.1, pp.209-220

[9] 横山 智弘, 坂井 直樹, 豊田 希, 薮田 哲郎：“強化学習を用いた大車輪運動の報酬と運動の解明”,
URL：http://yabsv.jks.ynu.ac.jp/PaperPDF/ROBOMEC2010/yokoyama2010.pdf

[10] 野田 彰一, 浅田 稔, 細田 耕：“強化学習によるロボットの行動獲得のための状間空間の自律的構成”,
URL：http://www.er.ams.eng.osaka-u.ac.jp/Paper/1995/Noda95a.pdf

[11] 高橋 秦岳, 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, 1999, Vol.17, No.1, pp.118-124

[12] 沼田 利伸:”センサ情報の自律的選択による効率的な行動選択の実現”,
室蘭工業大学卒業研究, 2011

[13] Richard S. Sutton and Andrew G.Bart (三上貞芳・皆川雅章共訳): ”強化学習”,
森北出版株式会社, 2001 年8 月10 日, 第1 版第2 版発行

[14] Pololu Robotics ant Electronics “GP2Y0W21YK0F”,
URL: http://www.pololu.com/file/download/gp2y0a21yk0f.pdf?file_id=0J85

[15] 株式会社 アットマークテクノ “Armadillo-300”,
URL:http://armadillo.atmark-techno.com/products/a300

[16] Arduino Uno, URL:http://arduino.cc/en/Main/arduinoBoardUno

[17] ダウ化工 株式会社 “スタイロフォームIB”,
URL:http://www.dowkakoh.co.jp/product/styrofoam.html