

修士論文

二重学習器を用いる強化学習の
性質とその応用

平成22年度

三重大学大学院 工学研究科 電気電子工学専攻

柴田 信雄



平成 22 年度 修士論文

二重学習器を用いる強化学習の 性質とその応用

専攻 三重大学大学院 工学研究科 電気電子工学専攻
研究室 情報処理研究室

平成 21 年度入学 409M224

氏名 柴田 信雄

目次

1	はじめに	1
2	二重学習器を用いる強化学習法	3
2.1	Q 学習	4
2.2	アルゴリズム	5
2.3	Q-table の選択	6
2.4	実験	7
2.4.1	実機実験環境	8
2.4.2	学習空間の構成	9
2.4.3	シミュレーション実験	11
2.5	実験結果	12
3	提案手法	15
3.1	概要	15
3.2	アルゴリズム	17
3.2.1	ソースエージェントの学習	17
3.2.2	ターゲットエージェントの学習	19
4	シミュレーション実験	22

目次	ii
5 実験結果	24
6 まとめ	27
参考文献	28
謝辞	30

目次

2.1	Whole Q-table and Partial Q-table	4
2.2	NS chart of Learning with Dual Q-tables	5
2.3	Results of simulation	7
2.4	MieC and Poles	8
2.5	Experiment environment	8
2.6	Action set	9
2.7	States of Pole and Goal	10
2.8	State set	11
2.9	Environment of simulation	12
2.10	Result of simulation	14
3.1	AT-table and Env-table	16
3.2	Learning algorithm of Source agent	18
3.3	Learning algorithm of Target agent	18
4.1	Simulation environment	23
4.2	Actions of Source and Target agents	23
5.1	Result of simulation	25

第1章

はじめに

人は歩いているとき、どう動くとどのように景色が変化するかを知識として記憶している。そして、例えば車の運転を練習するとき、アクセル操作量やハンドル操作量に対して、知識を利用してどのように景色が変化するかで、車の動きを理解する。しかし、車の運転では、歩いているときには起こらない景色の変化が起きることもあり、その場合は新しい知識として学ぶ必要がある。本研究では、このような人の知識の再利用の機構をモデル化し、学習の試行回数を削減する手法である異形態間学習を考える。

ロボットの学習法の一つとして、環境とエージェントの相互作用を通して学習する強化学習 (Reinforcement Learning)[1, Kaebing96][2, Sutton98] がある。しかし、実環境において複雑な環境を学習する場合、学習器が複雑膨大になり、学習時間が増大する。そのため、強化学習における試行回数の削減は、実環境での学習において重要な問題となる。

強化学習の効率化に関する研究には、最初にゴールに近い簡単な状況から学習し、徐々に複雑な状況へと移行していく [3, Asada96] や、既に持っている行動政策の中で不都合な部分のみを学習しなおすことによって学習時間を短縮する [4, Minato00] などがある。

また、複雑なタスクを細かいサブタスクに分解した強化学習モジュールを階層的に並べて学習する階層型強化学習も研究されている [6, Takashi03][7, Uchibe04]。これは、下位モ

ジュールが単純なサブタスクを学習し、上位の学習器が下位の学習器を利用してより高いレベルのタスクを学習することで学習の早期収束を目指している。

その他にも、連続状態空間を離散化する際、タスクに応じた適切な状態数を維持することで、過剰な状態が分割されることによる学習の遅延を防ぐ [5, hamagami03] や、環境に対応する状態空間を複数の部分空間に分け、それら部分空間における比較的単純なセンサ-モータ写像をモジュールとして学習・記憶しておくことで、環境の変化に伴い異なる行動の生成が必要な場合でもモジュールを組み換えることで速やかな対応が可能な [9, Gouko08] がある。

これらの研究は、最適な状態空間の構成法や、タスクの分割法を議論することによって学習の効率化を図っているが、本論文では、以前に学習した知識を再利用することで学習の効率化を図る。

知識を再利用する学習法は転移学習 (Transfer Learning) [12, Taylor09] と呼ばれ、複数のタスクで知識として使える共通タスクを分離学習し、それを再利用する [10, Yamaguchi09] や、ニューラルネットワークを用いて異なる環境間の状態や行動の対応付けを学び、異なる環境での学習結果を再利用する [11, Taylor07] などがある。これらに対し、本研究では、二重学習器を用いる強化学習法 [8, Nishimura06] を応用した異形態間学習を提案する。

具体的には、同じ環境で同じタスクを形の異なるエージェントが学んだ学習結果を再利用し、学習の効率化を図る。本論文では、2 章で、仮想空間でしか有効で無かった二重学習器を用いる強化学習法 [8, Nishimura06] が実機でも有効であることをシミュレーション実験との比較検討によりその有効性を確認する。その後、3 章で異形態間学習を提案し、5 章でシミュレーション実験の結果を示し、実機でも適用できる可能性を示す。

第2章

二重学習器を用いる強化学習法

人は一度経験した環境では、過去の経験から行動を選択する。そして経験していない環境では、過去に経験した知識から適正と思われる行動を推論し、選択する。もし選択した行動が適正行動でなくとも、その行動が不適切であることを知識と経験として蓄え、次からその行動を選択しなくなる。二重学習器を用いる強化学習法 [8, Nishimura06] は、この人の知識と経験を利用する行動学習をモデル化し、ロボットが効率良く学習することを目的としている。

具体的には、Fig. 2.1 に示すように、環境に対する学習空間を2つ用意し、これを同時に学習させる。一つは、環境空間を完全に表現する全学習空間（以下、全空間と呼ぶ）とし、これを経験の蓄積として用いる。もう一つは、全学習空間の一部を圧縮した部分学習空間（以下部分空間と呼ぶ）とし、これを知識の蓄積として用いる。全空間は、空間が大きいので学習は遅いが、環境に対して細かく対応付けをする。部分空間は空間が小さいので学習は速いが、環境に対して荒く対応付けをする。行動選択をするたびに、この2つの学習空間のより学習できている方の行動を選択し、同時に更新することで、学習が速く環境にも細かく対応付けができる。

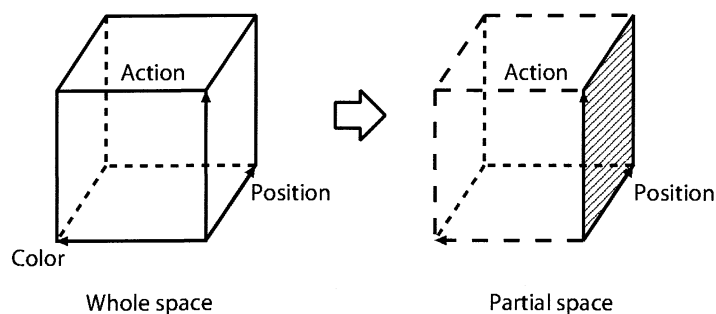


Fig. 2.1 Whole Q-table and Partial Q-table

2.1 Q 学習

本論文では、強化学習に Q 学習を用いる。本節では、Q 学習について説明する。

Q 学習は、環境と行動の組（以下、ルールと呼ぶ。）ごとに評価値 Q をもち、目標達成に至るまで各ステップごとに以下の式 (2.1) 式および式 (2.2) 式を繰り返し用いて、 Q 値を更新することで学習する。

$$Q(s_k, a) \leftarrow (1 - \alpha)Q(s_k, a) + \alpha(r + \gamma V(s_{k+1})) \quad (2.1)$$

$$V(s_{k+1}) = \max_{a \in A} Q(s_k, a) \quad (2.2)$$

ここで、 s_k は現在の状態、 A は行動集合、 a は選択行動、 s_{k+1} は遷移後の状態、 r は報酬値、 $\gamma(0 \leq \gamma < 1)$ は割引率、 $\alpha(0 < \alpha < 1)$ は学習定数である。

ルールごとの Q 値を表にしたものを Q-table と呼ぶ。

2.2 アルゴリズム

Fig. 2.2 に、二重学習器を用いる強化学習法のアルゴリズムを示す。ここでは NS チャーットの説明をする。

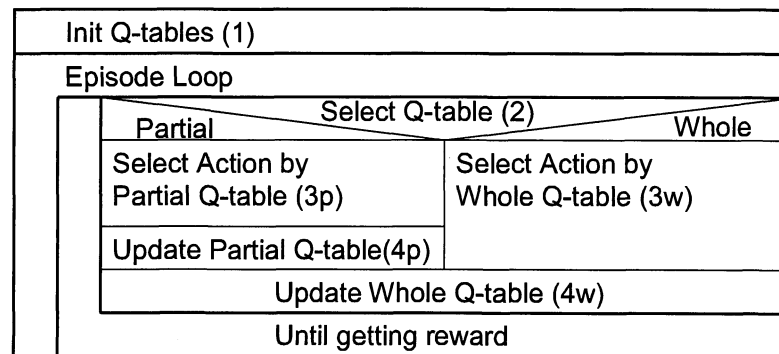


Fig. 2.2 NS chart of Learning with Dual Q-tables

1. 環境の設定

シミュレーションで使う学習環境や学習パラメータ、タスクなどを設定する。

2. Q-table の初期化

学習エージェントが使う全空間 Q-table および部分空間 Q-table の Q 値を初期化する。

3. Q-table の選択

学習エージェントが全空間 Q-table と部分空間 Q-table のどちらを用いて行動選択するかは、どちらの Q-table が有用な情報を持っているかを平均情報量を用いて判断し、決定する。詳しくは 2.3 節で説明する。

4. 行動選択

学習エージェントは、平均情報量により選択された Q-table に対し Boltzmann 選択を使って行動を選択する。Boltzmann 選択は、式 (2.3) から行動選択確率を求めるものである。

$$p(a \mid s_k) = \frac{\exp(\frac{Q(s_k, a)}{T})}{\sum_{a' \in A} \exp(\frac{Q(s_k, a')}{T})} \quad (2.3)$$

ここで、 $p(a \mid s_k)$ は、ある時刻 k の状態 s_k で行動 a を選択する確率、 $Q(s_k, a)$ は、ある時刻 k の状態 s_k における行動 a の Q 値、 T は温度を示す。

5. Q-table の更新

学習エージェントは 2.1 節で述べた Q 学習の Q 値の更新関数式 (2.1) および式 (2.2) を用いて学習サイクルごとに Q 値を更新し、行動を評価する。

以降、(3)-(5) のサイクルを報酬を得るまで繰り返し Q 値を更新する。

2.3 Q-table の選択

全空間 Q-table と部分空間 Q-table のどちらを使うかは、平均情報量を用いて判断する。平均情報量とは“情報の不確かさ”を評価するものである。これを行動選択確率に当てはめると、平均情報量が低ければ低いほど行動が確定的であり、学習空間が有効であることを示す。具体的には、全空間 Q-table と部分空間 Q-table の平均情報量を計算し、平均情報量が低い学習空間を使って行動を選択することにより、環境に最適な行動が選ばれることが期待できる。平均情報量 $H(s)$ は、式 (2.4) で求められる。

$$H(s) = \sum_{a \in A} p(a \mid s) \log_2 \frac{1}{p(a \mid s)} \quad (2.4)$$

$p(a \mid s)$ は式 (2.3) で定義される、状態 s で行動 a の選択される確率である。

2.4 実験

本手法は、西村ら [8, Nishimura06] によって、シンプルなシミュレーション実験において **Fig. 2.3** に示すように有効性が確認されている。全空間または部分空間だけを用いた通常の学習に比べ、2つの学習空間を同時に学習することにより、高速かつ正確に学習できていることが確認できる。すなわち、部分空間が全空間の全てを表現できる場合は **Fig. 2.3(a)** のように、部分空間とほぼ同じ速度で速く学習できている。部分空間が全空間の半分を表現できる場合は、**Fig. 2.3(b)** のように学習初期は部分空間を用いて速く学習が進み、学習後半では全空間を用いて正確に学習が来ている。そして、部分空間が全空間をまったく表現できない場合に関しても、**Fig. 2.3(c)** のように全空間だけを用いる場合とほぼ同じ速度で学習できている。

本手法は、実機での学習における有効性が示されていないため、ここでは、部分空間が全空間の一部を表現できる場合に関して実機実験をして、実機におけるこの手法の有効性を確認する。

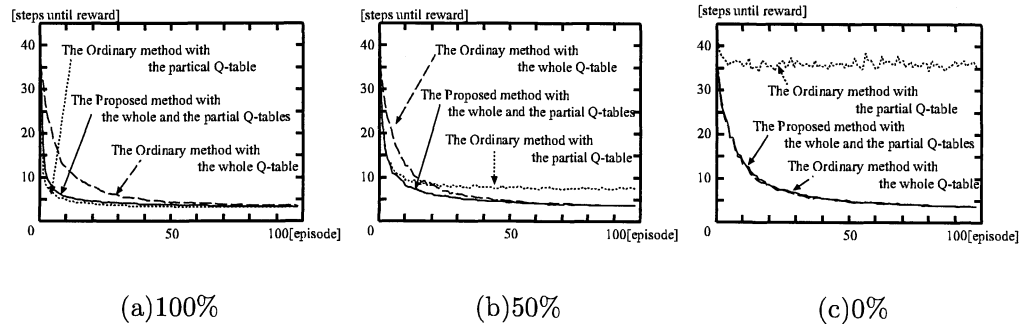


Fig. 2.3 Results of simulation

2.4.1 実機実験環境

本論文の実験では、Fig. 2.4 に示す自律移動ロボット MieC を用いて、色の異なる4色のボールを Fig. 2.5 の環境において黒のマーカで示されるゴールまで運ぶタスクを学習する。実験環境の大きさは、 $0.84[m] \times 0.54[m]$ となっている。

MieC は三重大学機械工学科メカトロニクス研究室で開発された自律移動ロボットで、移動機構として2本の無限軌道を用いる。2つの無限軌道は、2つのモータにより、それぞれ独立に駆動される。外部センサとしては、CCD カメラ (Logicool 製の QV-4000) を搭載している。外部通信には無線 LAN を用いる。また、CPU カードと FPGA カードを搭載しており、画像処理などは CPU カードが担当し、モータ制御などの処理は FPGA カードが担当する。今回使用する MieC には永久磁石を内蔵したブレードが前方に取り付けられており、内部に鉄を埋め込んだボールを一度捕まえると離さないようになっている。ボールおよびゴールの認識には搭載している CCD カメラを用いる。

報酬はボールがゴールに入って初めてエージェントに与えられる。各ボールは色によってゴールの右側に入れるか左側に入れるかが決められており、赤いボールはゴールの左 (A)、青はゴールの右 (B)、緑と黄色は A、B どちらでも良い。

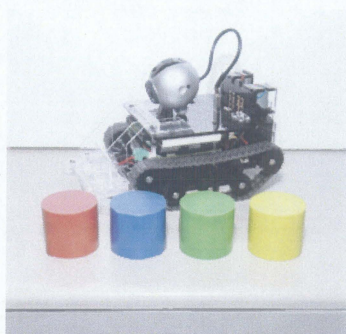


Fig. 2.4 MieC and Poles

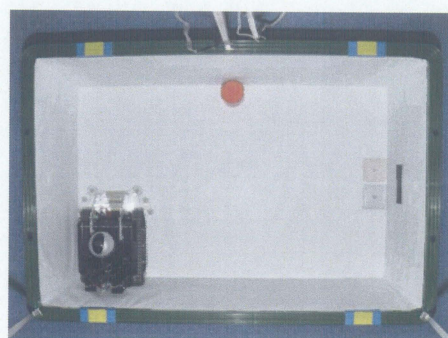


Fig. 2.5 Experiment environment

2.4.2 学習空間の構成

行動集合と状態空間の構成方法を説明する。

行動集合は、**Fig. 2.6** に示すように、{Forward, Backward, Pivot turn right, Pivot turn left} の 4 つの行動で構成される。今回の実験では速度は一定とする。

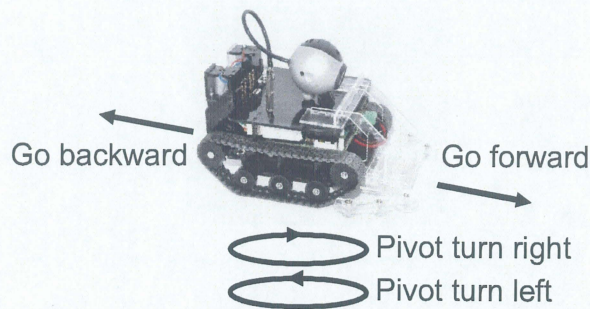


Fig. 2.6 Action set

状態集合は、ボールとゴールの見え方の状態と、ボールの色の状態で構成される。ボールとゴールの見え方の状態は、MieC の CCD カメラから取得した画像中のボールとゴールの重心位置によって構成する。

1. ゴールの見え方の状態空間

Fig. 2.7(a) に示すように、重心の垂直方向の位置からエージェントとの距離 $distance\{far, near\}$ 、重心の水平方向の位置 $position\{left, center, right\}$ 、マーカーの傾き角から $direction\{left\ direction, center, right\ direction\}$ のそれぞれの組み合わせ $18(2 \times 3 \times 3)$ 通りに加え、右に見えなくなったか左に見えなくなったかの 2 通りの全 20 通りで構成する。

2. ボールの見え方の状態空間

Fig. 2.7(b) に示すように、重心の垂直方向の位置からエージェントとの距離 $distance\{far, near\}$ 、重心の水平方向の位置 $position\{left, center, right\}$ の組み合わせ

6(2×3) 通りに加え, 右に見えなくなったか左に見えなくなったかの 2 通りの全 8 通りで構成する.

3. ポールの色の状態集合

Fig. 2.4 に示すように, ポールの 4 色 color{red, blue, green, yellow} に加え, 色が不明の状態の全 5 通りで構成する.

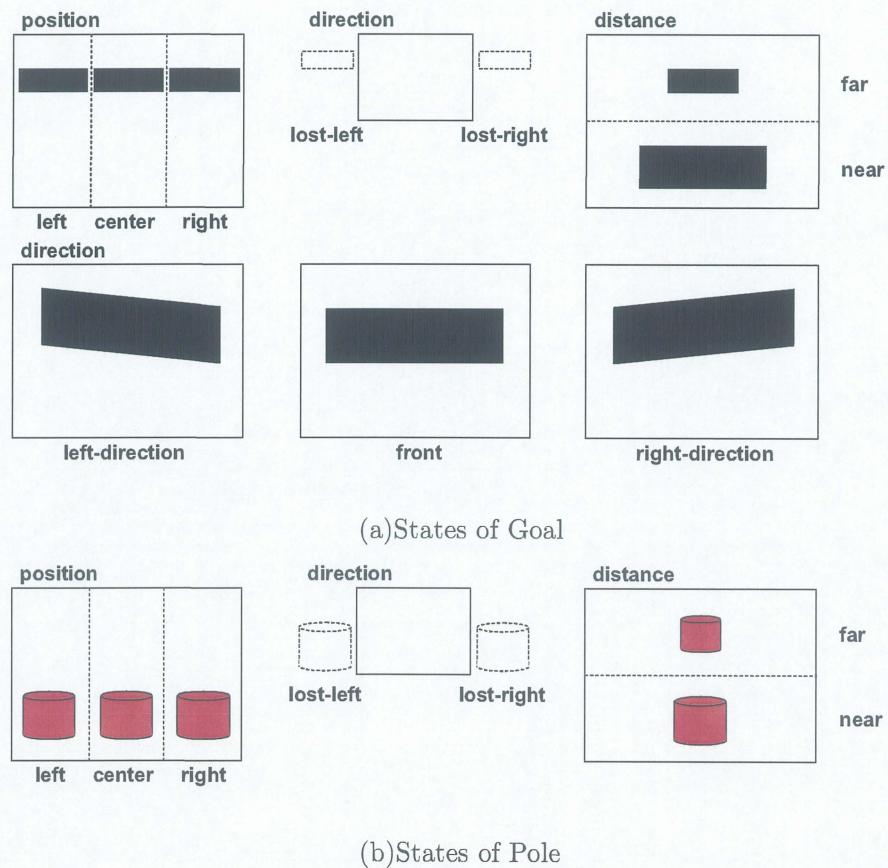
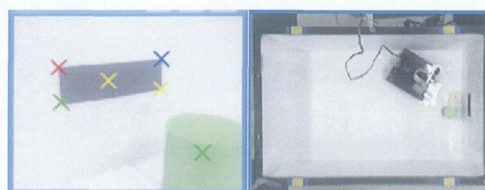


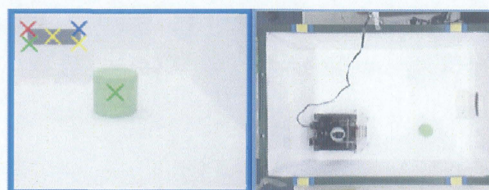
Fig. 2.7 States of Pole and Goal

ポールとゴールの状態の具体例を Fig. 2.8 に示す. (a) の例では, ゴールの状態は {position, distance, direction} = {center, near, right direction} となり, ポールの状態は {position, dis-

tance}={right, near}となる。同様に, (b)の例ではゴールの状態は{left, far, center}, ポールの状態は{center, far}となる。



(a)Example1



(b)Example2

Fig. 2.8 State set

これらの状態を二重学習器を用いる強化学習に適用するため, 全空間をゴールとポールの見え方の状態の組み合わせとポールの色の状態の組み合わせからなる $800(20 \times 8 \times 5)$ 状態で構成し, 部分空間をポールの色の状態を除いたゴールとポールの見え方の状態の組み合わせのみの $160(20 \times 8)$ 状態で構成する。

2.4.3 シミュレーション実験

今回の実験では, まず2.4.1項で説明した実験環境のシミュレータ実験をし, その結果と実機実験の結果を比較し, 有効性を確認する。本項では, シミュレータについて説明する。

シミュレーション実験の環境をFig. 2.9に示す。今回の実験では, 初期状態としてFig. 2.9のように学習エージェントとポールとゴールが直線上に配置される。このとき, ゴールとポールの状態はそれぞれ{center, far, center}, {center, far}となっている。

学習エージェントは、前後進は $0.5[\text{pixel}/\text{step}]$ ，超信地旋回は $0.1[\text{deg}/\text{step}]$ の速さで移動する。また、状態変化が起きるまでは同じ行動をとり続け、状態変化が起きて初めて Q-table を更新し、次の行動を選択する。状態変化が起こらない状況になった場合（例えば壁に向かってまっすぐ走り続けるなど）は、負の報酬を与えて Q-table を更新し、次の行動を選択する。

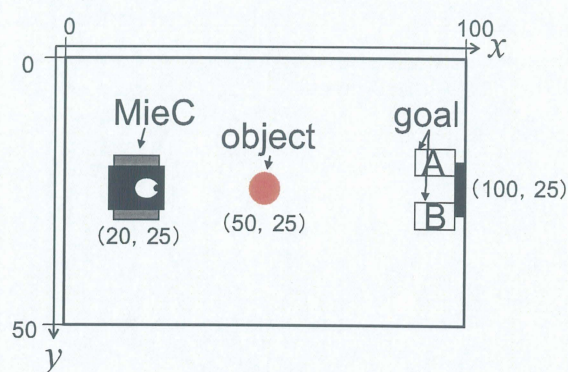


Fig. 2.9 Environment of simulation

2.5 実験結果

Fig. 2.10 にシミュレーション実験の結果を示す。各結果は 1000 試行の平均値である。各パラメータの値は、各 Q 値の初期値は 0.0，報酬 r は正の報酬が 1.0，負の報酬が -1.0，学習率 α は 0.3，減衰率 γ は 0.85，ボルツマン選択の温度 T は 0.07 となっている。

Fig. 2.10 の結果より、二重学習器を用いる強化学習法は、

1. 学習初期においては全空間 Q-table だけを用いた場合よりも速く学習できており、
2. 学習後半においては部分空間 Q-table だけを用いた場合よりも正確に学習できている。

本項では、**Fig. 2.5** に示す環境で、パラメータの値や Q-table の構成をシミュレータと同じ条件で実機実験し、次の2つのポイント (1) 学習初期および (2) 学習後半においてシミュレーション実験と同じ傾向が見られるかどうかを確認して実機における有効性を検討する。

Table.2.1, Table.2.2 にポイント (1) およびポイント (2) の実機実験の結果を示す。

Table.2.1 の結果は1~16 エピソードまでの全ステップ数の合計値、Table.2.2 の結果は501~516 エピソードまでの各エピソードのステップ数の平均値である。実機実験の結果は4 試行の平均値であり、シミュレーションの結果は1000 試行の平均値である。ポイント (2) の結果は、実機で全て学習するには多くの時間が必要となるため、シミュレーションで500 エピソードまで学習した Q-table を用いて501 エピソード目から学習している。

これらの結果から、ポイント (1) およびポイント (2) についてそれぞれ次のことが確認できる。

1. シミュレーション結果と同様に全空間 Q-table だけを用いた場合と比較すると、ステップ数の減少、および実時間での学習時間の減少が認められ、速く学習できていることがわかる。
2. シミュレーションの学習結果を用いて実機で学習すると、シミュレーションと同様の傾向が確認できる。すなわち、部分空間 Q-table だけを用いた場合にはゴールまで多くのステップ数が必要となっており正確に学習できていない。それに対し、二重学習器を用いる強化学習では少ないステップ数でゴールまで到達できており正確に学習できている。

以上の結果から二重学習器を用いる強化学習法は実機においても有効であることが確認できた。なお、実機の結果とシミュレーションの結果の数値に差があるのは、実機実験の試行回数がシミュレーション実験に対して非常に少ないためと考えられる。

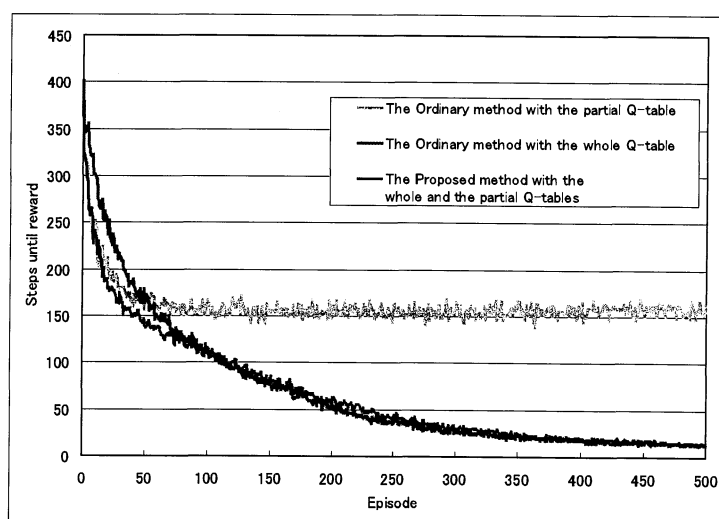


Fig. 2.10 Result of simulation

Table 2.1 Result of actual experiment at the point(1)

	Whole Q-table		Dual Q-tables	
	Actual	Simulation	Actual	Simulation
Steps	2435	3352.8	2199	2760.7
Time[sec]	2843		2317	

Table 2.2 Result of actual experiment at the point(2)

	Partial Q-table		Dual Q-tables	
	Actual	Simulation	Actual	Simulation
Steps	81	151.6	8	12.5

第3章

提案手法

本章では、2章で有効性を確認した二重学習器を用いる強化学習法を応用した、異形態間学習を提案する。

3.1 概要

学習エージェント（以後ターゲットエージェントと呼ぶ）があるタスクを学習する際、同じ環境で同じタスクを形の異なる別の学習エージェント（以後ソースエージェントと呼ぶ）が以前学んだ結果を知識として再利用することで効率的に学習する。

提案手法では、ターゲットエージェントは学習に次の4つのテーブルを用いる。

1. ターゲットエージェントの Q-table (Target Q-table)

各状態とターゲットエージェントの行動で構成される Q-table.

2. ソースエージェントの Q-table (Source Q-table)

各状態とソースエージェントの行動で構成される Q-table. この Q-table はすでにソースエージェントによって学習されている.

3. 行動変換テーブル Action translation table (AT-table)

Fig. 3.1(a) に示すように, ソースエージェントの行動とターゲットエージェントの行動で構成されるテーブル. ソースエージェントの学習結果を再利用する際に, ソースエージェントの行動をターゲットエージェントの行動と対応付けるために用いられる. これは, ターゲットエージェントの学習時に Target Q-table と共に学習される.

4. 環境テーブル Environment table (Env-table)

Fig. 3.1(b) に示すように, ソースエージェントの学習時に, ある状態においてある行動をとった時の状態遷移確率を記録しておくためのテーブル. このテーブルは AT-table の学習時に用いられる.

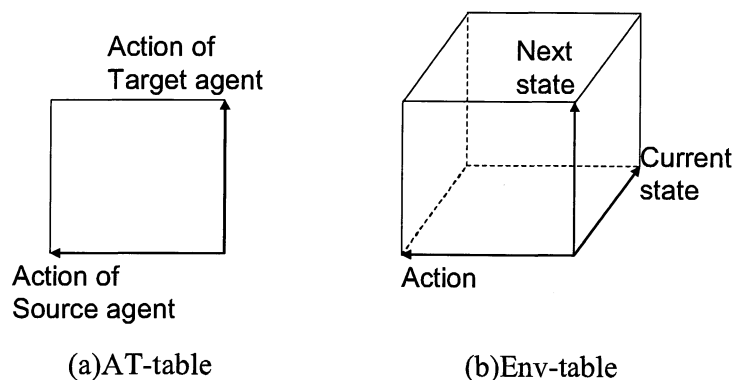


Fig. 3.1 AT-table and Env-table

提案手法では、ターゲットエージェントは 2 通りの方法で学習し、行動を選択する。まず一つ目の方法 (Way1) は、Target Q-table を用いて学習する。これは、環境とターゲットエージェントの行動全てを表現したテーブルを用いて学習するため、正確に行動を学習できるが、状態数が多くなるため学習に時間がかかる。二つ目の方法 (Way2) は、Source Q-table と AT-table を用いて学習する。具体的には、学習済みの Source Q-table が各状態に対して出力するソースエージェントの最適行動を、AT-table を用いてターゲットエージェントの行動に変換する。この方法では学習するのは状態数の少ない AT-table だけでよいので学習は非常に速く進むが、Source Q-table と AT-table の組み合わせではターゲットエージェントの行動全てを正しく表現できないため、正確に学習できない。

提案手法では二重学習器を用いる強化学習法を応用し、これら二つの方法を同時に学習し、行動選択毎により学習できている方法 (Way1 または Way2) の行動を選択する。これにより、学習初期においては速く学習の進む Way2 の行動が選択され、学習後半は正確に学習できる Way1 の行動が選択されることで、高速かつ正確に学習することが期待できる。

3.2 アルゴリズム

提案手法のアルゴリズムのブロック図と NS チャートを Fig. 3.2 および Fig. 3.3 に示す。

3.2.1 ソースエージェントの学習

ソースエージェントは、Fig. 3.2 にしたがって、ターゲットエージェントの学習前にあらかじめ Source Q-table を学習し、Env-table を記録する。

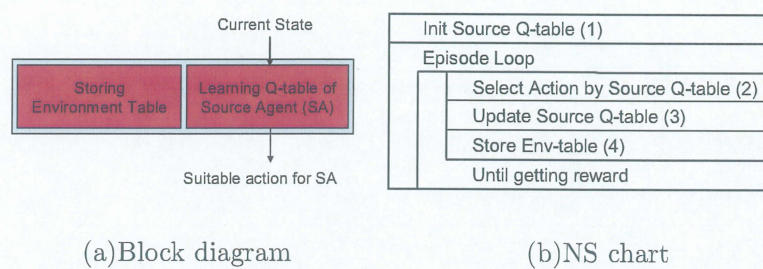


Fig. 3.2 Learning algorithm of Source agent

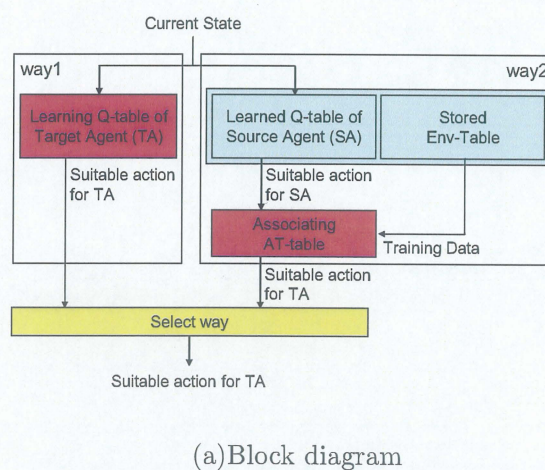


Fig. 3.3 Learning algorithm of Target agent

3.2.2 ターゲットエージェントの学習

1. Target Q-table および AT-table の初期化と Source Q-table および Env-table の読み込み

ターゲットエージェントは Target Q-table を初期化し、学習済みの Source Q-table と Env-table を読み込む。Source Q-table と Env-table の各値は、ターゲットエージェントの学習中に変化することはない。

2. Way(Way1, Way2) の選択

Way1 と Way2 のどちらかを、二重学習器を用いる強化学習法と同様、各テーブルの平均情報量を用いて選択する。各 Way の平均情報量は式 (3.1)、式 (3.2)、式 (3.3)、式 (3.4)、式 (3.5) によって計算される。

$$H_{Q_{target}}(s) = \sum_{a_t \in A_t} p(a_t|s) \log_2 \frac{1}{p(a_t|s)} \quad (3.1)$$

$$H_{Q_{source}}(s) = \sum_{a_s \in A_s} p(a_s|s) \log_2 \frac{1}{p(a_s|s)} \quad (3.2)$$

$$H_{AT-table}(a_s) = \sum_{a_t \in A_t} p(a_t|a_s) \log_2 \frac{1}{p(a_t|a_s)} \quad (3.3)$$

$$H_{way1}(s) = H_{Q_{target}}(s) \quad (3.4)$$

$$H_{way2}(s) = H_{Q_{source}}(s) + H_{AT-table}(a_s(s)) \quad (3.5)$$

ここで、 a_t はターゲットエージェントの行動、 a_s はソースエージェントの行動、 $p(a|s)$ は式 (2.3) で定義される状態 s において行動 a が選択される確率、 $p(a_t|a_s)$ は式 (3.8) で定義され、AT-table で行動 a_s のときに行動 a_t が選択される確率である。式 (3.5) の関数 $a_s(s)$ は、状態 s の時に Boltzmann 選択を用いて Source Q-table で選択された行動 a_s を出力する。

3. 行動選択

ターゲットエージェントは Boltzmann 選択を用いて式 (3.6) および式 (3.7) から得られる選択確率で Source Q-table および Target Q-table から行動 a_s , a_t を選択する.

$$p(a_s | s_k) = \frac{\exp(\frac{Q_{source}(s_k, a_s)}{T})}{\sum_{a'_s \in A_s} \exp(\frac{Q_{source}(s_k, a'_s)}{T})} \quad (3.6)$$

$$p(a_t | s_k) = \frac{\exp(\frac{Q_{target}(s_k, a_t)}{T})}{\sum_{a'_t \in A_t} \exp(\frac{Q_{target}(s_k, a'_t)}{T})} \quad (3.7)$$

ここで, $p(a_s | s_k)$ は, ある時刻 k の状態 s_k で行動 a_s を選択する確率, $p(a_t | s_k)$ は, 状態 s_k で行動 a_t を選択する確率, $Q_{source}(s_k, a_s)$ は, 状態 s_k における行動 a_s の Q 値, $Q_{target}(s_k, a_t)$ は, 状態 s_k における行動 a_t の Q 値, T は温度を示す.

4. AT-table による行動変換

Source Q-table で選択されたソースエージェントの行動 a_s を AT-table を用いてソースエージェントの行動 a_t に変換する. a_s の時, a_t は Boltzmann 選択を用いて式 (3.8) で得られる確率で選択される.

$$p(a_t | a_{s_k}) = \frac{\exp(\frac{AT(a_{s_k}, a_t)}{T_{AT}})}{\sum_{a'_t \in A_t} \exp(\frac{AT(a_{s_k}, a'_t)}{T_{AT}})} \quad (3.8)$$

ここで, $p(a_t | a_s)$ はある時刻 k のソースエージェントの行動 a_{s_k} でターゲットエージェントの行動 a_t を選択する確率, $AT(a_{s_k}, a_t)$ は, a_{s_k} における a_t の AT-table の値, T_{AT} は温度を示す.

5. AT-table の更新

AT-table はソースエージェントの行動とターゲットエージェントの行動の対応付けを学ぶためのテーブルである. ターゲットエージェントはソースエージェントが学習し

たときに記録した Env-table の状態遷移確率の値と更新関数式 (3.9) を用いて行動選択毎に AT-table の値を更新する.

$$AT(a_{s_k}, a_{t_k}) \leftarrow (\alpha_{AT} AT(a_{s_k}, a_{t_k}) + \gamma_{AT} p(s_{k+1} | s_k, a_{s_k})) \quad (3.9)$$

ここで, $p(s_{k+1} | s_k, a_{s_k}) = Env(s_k, s_{k+1}, a_{s_k})$ (状態 s_k においてソースエージェントの行動 a_{s_k} を取ったとき, 次の状態 s_{k+1} に遷移する確率, すなわち, s_k, s_{k+1}, a_{s_k} における Env-table の値), α_{AT} および γ_{AT} は, $0 < \alpha_{AT} < 1, 0 \leq \gamma_{AT} < 1$ の範囲の値である.

6. Target Q-table の Q 値の更新

ターゲットエージェントは Q 学習の Q 値の更新関数式 (2.1) および式 (2.2) を用いて学習サイクルごとに Target Q-table の Q 値を更新する.

第4章

シミュレーション実験

提案手法の有効性をシミュレーション実験で確認する。本章ではシミュレータの詳細を説明する。

Fig. 4.1 にシミュレーション環境を示す。状態集合は、2.4.2 項で説明した構成法と同様、ゴールの見え方の状態 20 通りと、オブジェクトの見え方の状態 8 通りの組み合わせの全 160 状態で構成する。ただし、今回のシミュレーションではオブジェクトの色は変化しないため、色の状態は存在しない。

エージェントの初期配置に関しては、**Fig. 4.1** に示す 2 つの初期位置をエピソード毎にランダムに選択する。

行動集合に関しては、**Fig. 4.2** に示すように、ソースエージェントは {Forward, Backward, Pivot turn Left, Pivot turn Right} の 4 つの行動を、ターゲットエージェントは {Forward, Backward, Forward Left, Forward Right, Backward Left, Backward Right} の 6 つの行動を持っている。

学習エージェントは、前後進は $0.5[\text{pixel}/\text{step}]$ 、超信地旋回は $0.1[\text{deg}/\text{step}]$ の速さで移動する。また、前後方への旋回は直進方向へ $0.5[\text{pixel}/\text{step}]$ 、回転方向へ $0.1[\text{deg}/\text{step}]$ 移動する。

学習エージェントに与えられる報酬は2.4.2項の実験とは異なり，ゴールの左右ではなくゴールの正面にオブジェクトを運ぶと報酬が与えられる．また，状態変化しない状況に陥った場合には，2.4.2項の実験同様，負の報酬を与えて各学習テーブルを更新し，次の行動を選択する．

各パラメータの値は，各 Q 値および AT-table, Env-table の初期値は 0.0, r は正の報酬が 1.0, 負の報酬が -1.0, $\alpha = 0.4$, $\gamma = 0.9$, $T = 0.05$, $\alpha_{AT} = 0.9$, $\gamma_{AT} = 0.35$, $T_{AT} = 0.5$ となっている．

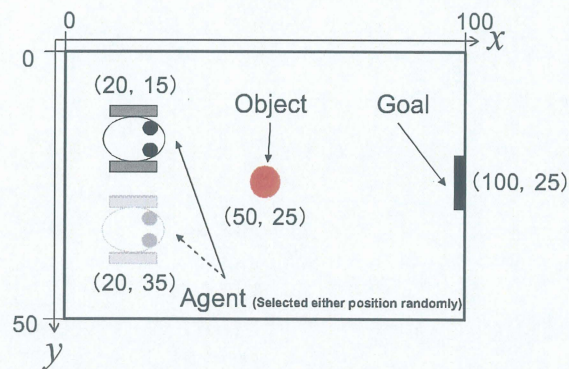


Fig. 4.1 Simulation environment

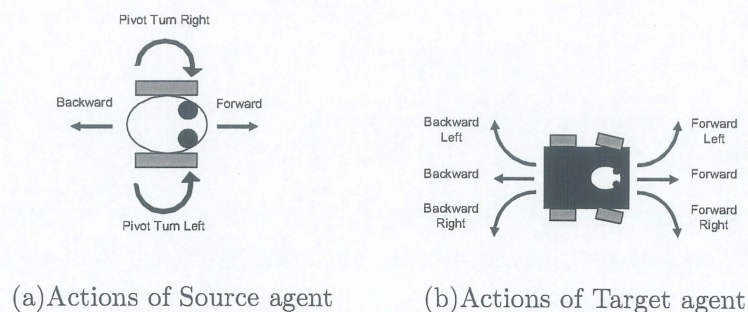


Fig. 4.2 Actions of Source and Target agents

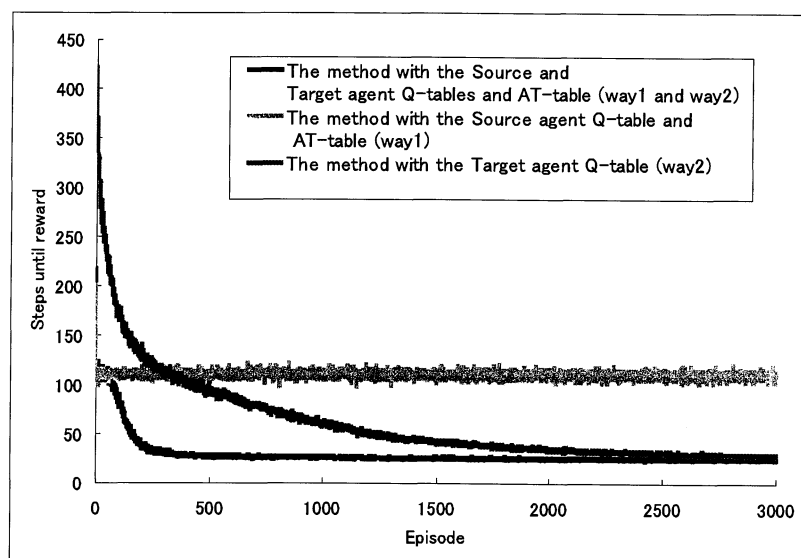
第5章

実験結果

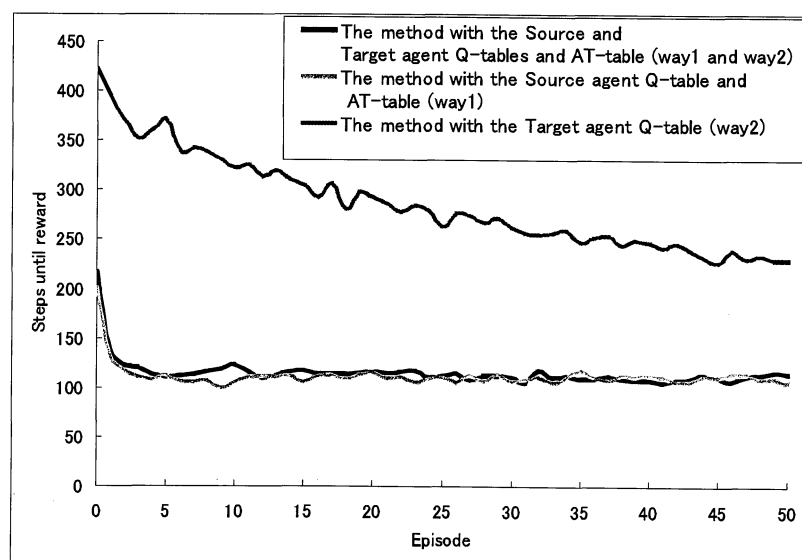
提案手法のシミュレーション実験の結果を **Fig. 5.1** および Table.5.1 に示す.

Fig. 5.1 から, Target Q-table だけを用いて学習 (Way1) すると正確に学習できるが時間がかかっており, Source Q-table と AT-table だけを用いて学習 (Way2) すると速く学習できているが正確に学習できていないことがわかる. それに対し, 提案手法では Way1 と Way2 を同時に学習することにより, 学習初期では Way2 同様速く学習が進み, 学習後半では Way1 同様正確に学習できていることから, 本手法の有効性がシミュレーション実験において確認できた.

また, Table.5.1 をみると, ソースエージェントとターゲットエージェントの行動で似た行動 (例えば Forward(source) \rightarrow Forward(target), Pivot turn Right \rightarrow {Forward Right, Backward Left} など) がそれぞれ高い値で対応付けられていることが確認できた.



(a) 1 to 3000 episodes



(b) 1 to 50 episodes

Fig. 5.1 Result of simulation

Table 5.1 Result of AT-table

	Forward	Backward	PT Right	PT Left
Forward	2.573	0.172	0.517	0.150
Backward	0.175	1.761	0.469	0.405
Forward Right	0.875	0.405	1.758	0.002
Forward Left	0.652	0.339	0.028	1.292
Backward Right	0.451	0.645	0.002	1.189
Backward Left	0.687	0.711	1.848	0.003

第6章

まとめ

本論文では、二重学習器を用いる強化学習法を実機へ適用し、さらにその応用である異形態間学習を提案した。二重学習器を用いる強化学習法は、実機を用いた実験においてもシミュレーション実験同様に学習時間が削減できており、その有効性を示せた。その応用の提案手法である異形態間学習は、シミュレーション実験により学習時間の削減が確認でき、その有効性を示せた。

参考文献

- [1] Leslie Pack Kaelbling, Michael L. Littman and Andrew W. Moore, Reinforcement Learning A Survey, Journal of artificial Intelligence Research, vol.4, pp.237-285, 1996.
- [2] Richard S. Sutton and Andrew G. Barto, Reinforcement Learning, MIT Press, Cambridge, MA, 1998.
- [3] MINORU ASADA, SHOUICHI NODA, SUKOYA TAWARATSUMIDA, KOH HOSODA, Purposive Behavior Acquisition for a Real Robot by Vision-Based Reinforcement Learning, Machine Learning, 23, pp279-303, 1996.
- [4] 港隆史, 浅田稔, 環境の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, Vol.18, No.5, pp.706-712, 2000.
- [5] 濱上知樹, 小坪成一, 平田廣則, 適応的な状態分割を行う Q-Learning における状態数の調整方法, 電子情報通信学会論文誌 D, Vol.J86-D-I, No.7, pp.490-499, 2003.
- [6] 高橋泰岳, 浅田稔, 階層型学習機構における状態行動空間の構成, 日本ロボット学会誌, Vol.21, No.2, pp164-171, 2003.
- [7] 内部英治, 銅谷賢治, 複数報酬のもとでの階層強化学習, 日本ロボット学会誌, Vol.22, No.1, pp.120~129, 2004.

- [8] Osamu NISHIMURA, Hirokazu MATSUI, Chieko HIOKI, Yoshihiko NOMURA, Reinforcement Learning with Self-Instruction by using dual Q-tables, AROB 11th, 2006.
- [9] 郷古学, 伊藤宏司, 環境変化の予測情報を利用するモジュール切換型行動生成モデル, 電子情報通信学会論文誌 D, Vol.91-D, No.3, pp.813-822, 2006.
- [10] 山口明彦, 杉本徳和, 川入光男, 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, Vol.27, No.2, pp.209~220, 2009.
- [11] Matthew E. Taylor, Gregory Kuhlmann, and Peter Stone, Autonomous Transfer for Reinforcement Learning, In The Autonomous Agent and Multi-Agent Systems Conference (AAMAS-07), 2008.
- [12] Matthew E. Taylor, Peter Stone, Transfer Learning for Reinforcement Learning Domains: A Survey, Journal of Machine Learning Research, 10, pp1633-1685, 2009

謝辞

本研究の遂行および修士論文の作成にあたり、丁寧なご指導とご助言を頂きました本学工学部機械工学科の松井博和助教、電気電子工学科の、篠木剛元教授に深く感謝いたします。また、本研究に関してご助言を頂きました情報処理研究室の鶴岡信治教授、高瀬治彦准教授、川中普晴助教に感謝いたします。

本研究を進めるにあたり、懇切なるご指導を頂いた荒川先輩、国分先輩、鹿間先輩、谷岡先輩に感謝いたします。また、同グループの同期として助け合い相談し合った加藤氏、森氏、佐野氏、および情報処理研究室、メカトロニクス研究室諸氏にも重ねて感謝いたします。