報酬に基づいた環境情報の取捨選択による行動学習の効率化に関する研究

木島康隆

2013年6月28日

目 次

第1章　序論

1.1　はじめに

近年，ロボットは工場や研究室のみならず，エンターテインメントロボット・家庭用ロボット・ホビーロボットといったように一般の人々にも広く認知されるようになった．例えば，エンターテインメント用途に用いられることが多いロボットして，本田技研工業の ASIMO[1] や NEC の Papero[2] などがある．家庭用ロボットとしては，掃除ロボット「ルンバ」，ホビーロボットでは近藤科学の KHR-3HV が有名であり，研究・開発が進んでいる [3]-[5]．また，宇宙空間 [6]-[13]・海中 [14]-[16]・原発作業用 [17]-[19] や災害現場でのレスキュー用 [20]-[25] といった極限環境作業用ロボットや医療用 [26]-[28] といった人々の目に触れにくい領域で活動するロボットも研究・開発されている．このように，現在多数のロボットが存在し，使用されている．こうした中で，人々のロボットに対する期待はさらに高まっている．

現在，実用化されているロボットの多くは予めプログラムされた通りの動作しか行わないタイプのロボットと人間が操作するタイプに分かれる．予めプログラムされた通りの動作しか行わないタイプのロボットは，工場のラインで用いられる産業用ロボットが主である．このようなロボットは，特定の作業を特定の環境下で特定の手順で行うため，プログラムによりロボットの行動を自動化することが可能である．例えば，自動車を組み立てるラインでは，各々のロボットがエンジンの取り付けやドアの取り付けといった単純な作業を与えられる．各ロボットは自身では移動せず特定位置に固定され，ラインから流れてくる組立途中の自動車に対しそれぞれ担当したタスクを遂行する．そのため，ロボットが直面するであろう状況（タスクの失敗含む）は容易に予測ができ，それらに対応する動作をプログラムすることが可能である．

一方，極限環境下で活動するロボットは人間が操作するタイプが多い．このようなロボットは，産業用ロボットなどとは異なり，多種多様な作業を周囲の状況変化が大きい環境下で行う．例えば，惑星探査ローバーは，ある惑星の地質や資源を探査する．探査する惑星の地形や気候を予め詳細に予想することは難しい．そのため，ロボットが惑星に適応するような挙動をプログラムすることは困難である．従って，人間が遠隔操作によって動かすことで惑星の調査を行う．

このように，現状ではロボットが活動する環境・タスクの複雑さによってロボットの利用方法は異なる．工場や研究室といった環境は，ロボットのために構築された環境といえる．現在ロボットは，工場や研究室といった環境のみならず家庭環境や極限環境での活動といったように多種多様な環境下で用いられることが求められている．そのような中で，ロボットがより便利に人間の社会に役立つ存在となるためには，複雑な環境下での自律行動能力が必要である．こうした自律的行動能力はロボット自身がタスクを達成するために直面している状況において最適な行動は何であるかを考える能力，すなわち知能を持つことが必要である．

1.2　機械知能

ロボットのような機械の知能に関する研究は数多く行われてきた．機械の知能で目指すところは，人間を始めとした生物の知能を実現することである．生物での知能は理解・認識・

適応・学習・推論といった情報処理能力のことである．こうした機械知能は人工知能という学術領域で研究が行われている．

1.2.1 人工知能の基礎領域

人工知能の基礎領域としては，問題解決・論理・知識表現・知識ベース・記号処理・学習・認知科学といった分野が存在する [29][30]．それぞれの分野に関して簡単に述べる．

問題解決

問題解決とは，対象問題を，状態空間や AND OR 木などのコンピュータ処理可能なデータ形式で表現し，解を探索する技術分野である．状態空間の内部に存在する解を高速に効率よく探索するための探索アルゴリズムや人間の経験的知識を利用する手法が提案されてきた．

論理

論理は数学的論理学などを基礎として発展してきた分野であり，人工知能における推論のための最も基礎的メカニズムである．対象問題を命題論理や述語論理で表現し，演繹的推論，導出原理などに基づく推論方法を用いて解の探索を行う．論理は現在でも人工知能の主要テーマであり，またさまざまな応用が行われている．様相論理，時相論理などのさらに高度な推論を可能にする論理体系の研究が進められている [31][32]．

知識表現

機械において知能を実現するにあたり，機械が蓄えるべき知識のデータ形式は重要である．人間の知識を表現するためには，論理のみでは不十分であり，特に知識工学では経験的知識をコンピュータ処理可能なデータ形式で表現することは重要である．これらは知識表現モデルとよばれ，プロダクションルール [33]・意味ネットワーク [34]・フレームシステム [35] などが提案されている．これらの知識表現に基づく知識記述言語が開発されており，これらを使って様々なエキスパートシステム [29] が開発されている．

知識ベース

知識表現モデルで記述された大量の知識の集合を蓄積・管理し利用するためのシステムである．単に知識を蓄積するだけでなく，既存の知識を基に新たな解を推論し，生成することができる．

記号処理

人工知能では，科学技術計算や事務処理のような数値処理や定期のデータ処理以外にも扱う情報が存在する．リストのような不定形なデータや言語のような概念を意味する記号の処理を行う．このような，記号処理をコンピュータで実行できるように LISP・Ruby・Perl といった多種多様なプログラム言語が開発されている．

学習

人間は過去の経験に基づいて，類似の問題を解いたり，経験と教示によってよりよい解法を見つけることができる．また，推論のような多数の事例から一般的な法則を導き出すことができる．こうした能力を学習能力という．学習能力に関する研究も古くから行われているが，人工知能の中では最も困難とされている研究分野の一つである．近年では，エキスパートシステムなどの知識獲得問題や，データベースやウェブなどから自動的に知識を獲得する技術が盛んに研究されている [36]-[38].

認知科学

認知科学では，人が事物を理解する過程やメカニズムを解明する．おもに，ニューラルネットワークやファジー理論といったものが注目されている [39]-[44]．ニューラルネットワークは脳の情報処理機能をモデル化した情報処理システムである．一方，ファジー理論は，人間のもつ「あいまいさ」を表現することができる．これらは，知識獲得のために訓練による学習を行う.

1.2.2　ロボットと人工知能

機械に人工知能を組み込むことで人間にとってより便利に機械を扱うことができると期待される．ロボットは人工知能を搭載する機械の 1 つとして考えることができる．ロボットに人工知能を組み込むことで，人間の命令の理解や自分自身による学習を可能とし，ロボット自身の判断で行動することができる．こうした人工知能技術が組み込まれたロボットを知能ロボットとよぶ.

知能ロボットと他の人工知能技術との違いは，知能の組み込み先が実体を伴う身体であるということである．知能が実環境に対して何らかのアクションを起こすためには身体は必要不可欠である．歩くという行動は，ロボットの身体に足が存在し，大地を踏みしめることができて初めて実現する行動である．また，環境そのものの存在もロボットには必要不可欠な要素である．歩くという行動には大地あり，そこに重力・摩擦力といった要素を含んだ環境があって可能である．すなわち，ロボットにとっては，身体と環境の両方の要素が必要になる．知能は身体を通して環境と相互作用することにより発達していく．ロボットは環境を認識し，行動する．行動の結果，変化した環境をロボットは再び認識する．このとき，行動が環境へ与える影響を学習する．認識と行動を繰り返すたびにロボットは，環境をどのように表現し行動を獲得するかといったことを学習する．こうしたロボットの認知発達過程に注目した研究分野を認知発達ロボティクスとよび，現在注目されている [45]-[47].

1.3　ロボットにおける知能研究

ロボットが環境に対する適切な行動を獲得するために必要なことは，直面している環境やタスクに関する知識の獲得とその利用である．すなわち，学習能力が必要となる．ロボットにおける学習能力に関する研究は数多く行われている．学習手法として，ニューラルネットワーク・遺伝的アルゴリズム・強化学習など種々のアルゴリズムが提案されている．各学習アルゴリズムに共通する事柄として，何らかの指標を基に行動を学習する．行動の指標として，教師信号・評価関数・報酬を用いて学習を行う．例えば，ニューラルネットワークでは教師信号を指標として，誤差逆伝播法によって教師信号との誤差を基に各層の結合荷重の変更を行う．遺伝的アルゴリズムでは，適合度関数が指標にあたる．強化学習では，報酬が指標となる．学習の完了後は，学習を行った意思決定部を用いて行動を行う．加えて，モデルの獲得による未来予測も行われ，行動生成に用いる [48].

こうした学習手法は運動学習 [49]-[52]，行動プランニング [53]-[55] や経路 [56]-[58] に関する学習手法の提案，人やロボット同士のコミュニケーションを用いた学習 [59]-[65][69]-[71] など多様な分野に適用されている．

ロボットの運動学習では，歩行動作の学習 [51]，大車輪動作の学習 [50] といったように単一の動作の学習を行う．また，行動プランニングに関する学習の研究では，定義された行動を組み合わせて一つのタスクを達成する手順を学習する．例えば，コーヒーを淹れるというタスクの場合，カップを食器棚から取る，お湯を沸かす，カップにコーヒーの粉を入れお湯を注ぐといった一連の行動の結果タスクが達成される．経路学習は，目的地への最短経路の学習や障害物を避けるように進む経路の学習など多岐にわたる．人やロボットとのコミュニケーションによる学習では，見まね学習 [59]-[62] や指示学習 [63]-[65] といったように相手からの情報を学習する．その他，捕食者・被捕食者の関係を基に，他のロボットとの競合により共進化的な学習も存在する [66]-[68]．

1.4 本研究で対象とする学習手法とその問題点

1.4.1 強化学習の概要

前述したとおり，様々な学習手法が提案されロボットに適用されている．本論文では，学習手法の中でも強化学習に注目する．強化学習は，実ロボットに適用されることの多い学習手法である．強化学習の特徴として，設計者が目標とする状態に対して報酬を設定することで，強化学習が目標状態に至るまでの行動の系列を自動的に学習するということが挙げられる．そのため，ニューラルネットワークなどに代表される教師信号のような正解データを用意する必要がなく，扱いやすいため近年注目されている．

強化学習の概要および具体的な手法については第 2 章で改めて述べるが，ここではまず先に強化学習の概要についてのみ説明する．強化学習の概要図を図 1.1 に示す．ロボット環境状態 s を読み取る．環境状態は主にセンサを通して読み取る．そして，読み取った状態に対して適切な行動 a を行う．行動の結果，環境はそれに見合った報酬 r を与える．報酬は，行った行動がどの程度望ましいものだったかを表す指標で，設計者によって定義され，与えられる．報酬が高いほど行った行動は望ましいものであるといえる．行動の結果，環境の状態は変化する．ロボットは，変化した状態をセンサを介して読み取り，それに適した行動を推測し，実行する．このプロセスを繰り返すことで，入力される環境状態に対して高い報酬を得ることができる行動を学習していく．

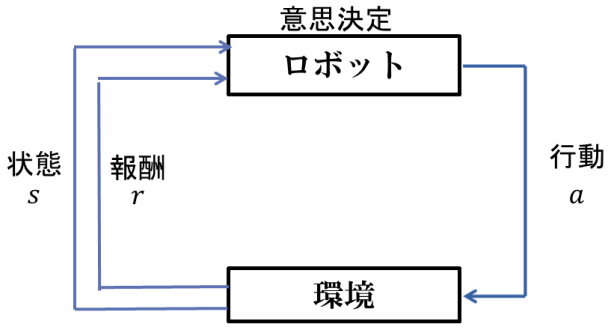


図 1.1: 環境との相互作用

強化学習において，学習された情報は Q 空間に蓄えられる．Q 空間は図 1.2 に示す通り，状態軸，行動軸，Q 値軸で構成される．状態軸とは，センサから得た環境状態の集合から構成され，行動軸はロボットが取りうる行動の集合から構成される．Q 値軸は，行動価値 Q の

集合から構成され，行動価値 Q（Q 値）は各状態と行動の対（状態行動対）における期待獲得報酬量を示す．すなわち，ある状態のときある行動がどの程度タスクの達成に適しているかを示す値である．強化学習は報酬を基に各状態行動を経験し，Q 空間上の Q 値を更新していくことで学習が進行する．

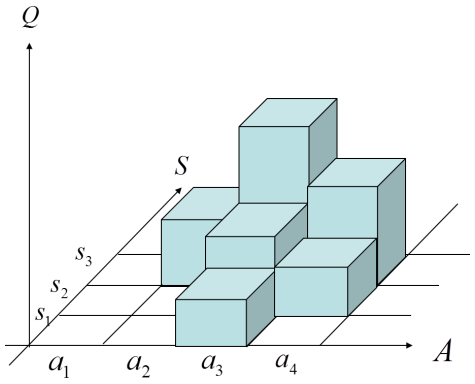


図 1.2: Q 空間

1.4.2 強化学習の問題点

強化学習の問題点として，学習に要する時間が挙げられる．ロボットが入力として扱うセンサの種類数の増加や分解能といったセンサの性能の向上に伴い，学習に要する時間が指数関数的に増大する．カメラやマイクといったように異なる環境要素を知覚するセンサが増えることで，センサの種類数が増加する．また，センサの有効範囲や分解能の向上により，性能が向上する．

学習空間（Q 空間）拡大のイメージ図を図 1.3 に示す．図 1.3 は，Q 空間内の状態行動対の増加の仕方に注目したものである．行動軸を A とし，図 1.2 における状態軸 S をセンサ軸 E に置き換えている．状態 E は各センサからのセンサ情報によって構築される．図 1.3 には 3 つの図があり，中央の上の図を基準として，左下の図（B）がセンサの種類数が増えた場合，右下の図（C）がセンサの分解能が向上した場合を表している．センサの種類数が増える場合（図 1.3B）は，センサ軸 E が増加しロボットが取りうる状態が増加する．センサの有効範囲や分解能が向上する場合（図 1.3C）は，センサ軸 E がとりうる値が増加し，ロボット取りうる状態が増加する．ロボットが取りうる状態が増加することは，Q 空間の増大を意味する．Q 空間が増大すると，その分各状態行動対の Q 値の更新回数が増加するため，学習に要する時間が増大する．近年のロボットは，ハードウェア技術の発達により高性能なセンサを多数搭載することが可能になり，学習時間の増大は現実的な問題となる．現実世界では，学習のための時間を確保することが難しく，可能な限り迅速に学習を行いタスクを遂行することが求められる．そのため，実際にロボットを運用するにあたり解決する必要がある問題である．そのためには，より効率的に学習を行う方法を考察することが求められる．

1.4.3 強化学習の学習効率化に関する従来研究

強化学習の効率化に関する従来研究として大きく分けて以下の 2 つのアプローチがある．

- コミュニケーションによる情報共有
- タスクに応じた状態空間の構成

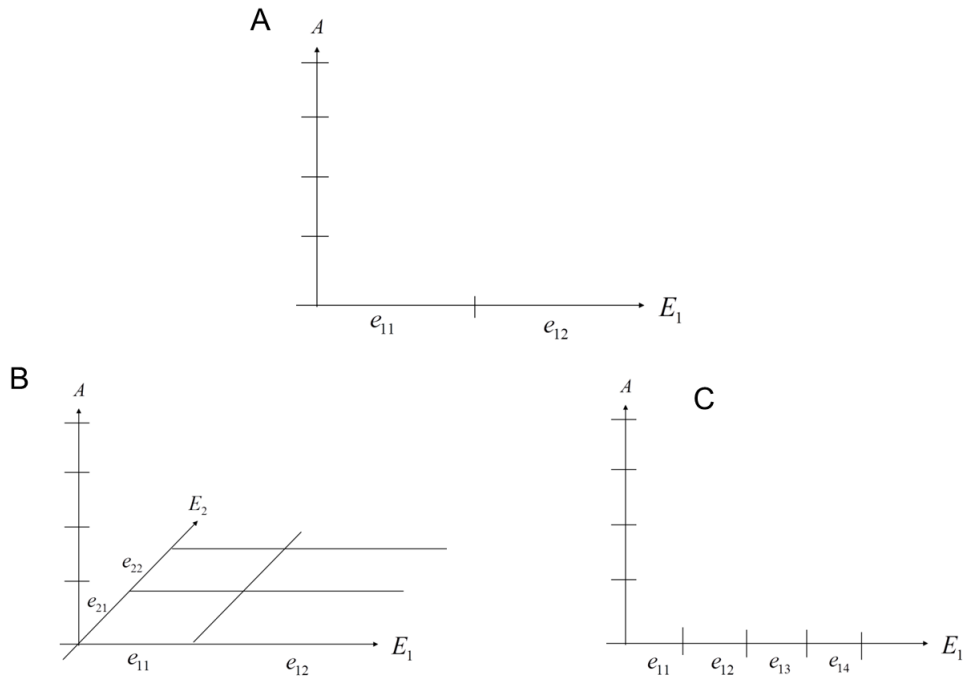


図 1.3: 学習空間拡大の例：センサの場合

コミュニケーションによる情報共有 [84]-[88] では，ロボットは群を構成し互いにコミュニケーションを行う．それにより学習に必要な経験量を互いに補完しあう．その結果，個体単体で学習するよりも効率的な学習が可能であると考えられる．各研究でコミュニケーションしている情報やその利用方法は異なるが，学習効率は向上している．

一方，状態空間の構成 [89]-[93] では，環境・タスクに応じて動的に状態空間を構成することで学習に応じた状態空間を構築する．従来の強化学習では，状態空間はグリッド状に構成するが，これらの研究ではタスクに合わせて大きさの異なる状態が構成される．報酬に近い場所は細かく状態が構成され，報酬から遠い状態に関しては大雑把に状態が構成される．その結果，従来よりも学習すべき状態数が削減され学習が効率化される．

1.5 本研究で考える効率化

強化学習では Q 空間内の各状態行動対の Q 値を報酬によって更新することで学習が進行する．従来研究では，コミュニケーションよる学習はロボット外部からの情報を学習に利用している．状態空間の構成はロボットの内部状態である Q 空間を改変することで行なっている．このことから，本研究では以下の 2 つのアプローチによって学習の効率化について考える．

- ロボット外部の情報の取捨選択：コミュニケーションにおける有益情報の取捨選択
- ロボット内部の情報の取捨選択：タスクに応じた環境情報の取捨選択による Q 空間の構成

1.5.1 ロボット外部の情報の取捨選択：コミュニケーションにおける有益情報の取捨選択

ロボット外部の情報の取捨選択とは，ロボットが入手する情報の取捨選択を意味する．従来研究では，コミュニケーションによる学習の効率化に関しては行われているが，コミュニ

ケーションされる情報が自身にとって有益であるかどうかはあまり考えられていない．コミュニケーションによって入手する情報は様々であり，それら全てが自身にとって有益であるとは限らない．そのため，コミュニケーションされた情報を取捨選択し利用することが望ましい．Ahmadabadi ら [87][88] はコミュニケーション相手の選定する手法を提案しているが，この手法は熟練者かそうでないかを判断するものであり，目的が異なる個体が混在している状況でのコミュニケーションに関しては特に考えられていない．目的が異なるというのは，例えば，経路計画問題においてそれぞれ目的地が異なる個体が混在する場合が挙げられる．このとき，目的地が異なる者同士のコミュニケーションはたとえ相手が熟練者であれコミュニケーションした情報が有効であるとは言えない．こうした場合，学習者が自身に有益な情報をもたらす他者を取捨選択しコミュニケーション行うことが望ましい．すなわち，外部からの情報を取捨選択することで，自身にとって有益な情報のみよって Q 空間を構成することを考える．それにより，コミュニケーションによる学習が効果的になると期待される．

1.5.2 ロボット内部の情報の取捨選択：タスクに応じた環境情報の取捨選択による Q 空間の構成

強化学習を適用したロボットは内部情報として Q 空間を保持している．従来研究では，すべてのセンサを使用することを前提として，タスクに応じた Q 空間を構成するというアプローチをとっている．本研究では，もっとハードウェアよりの思想から，センサそのものの必要性の有無の判断を行うことがまず有効であると考えた．ロボットはセンサから受け取った周囲の環境情報を入力情報として意思決定を行う．従来の強化学習では，すべてのセンサを基に Q 空間を構築している．しかし，全てのセンサからの情報がタスクの達成に必要であるとは限らない．ロボットの動作するタスクの内容に依存して必要となるセンサと不要であるセンサが存在する．タスクによるセンサの必要・不要の例として，山登りタスクとゴミ拾いタスクにおける高度センサを考える．山登りタスクの場合は，ロボットの現在の高度を知ることがタスクの達成状況を知ることに繋がるため，必要なセンサである．一方ゴミ拾いタスクの場合は，高度センサを用いてもゴミの位置を知ることはできず，タスクの達成に貢献しない．そのため，不要なセンサである考えられる．このように環境・タスクに必要なセンサが存在する．ロボットが必要なセンサを自律的に判別することで，不要なセンサ情報を削減した Q 空間を基に学習を行い，効率的な学習を実現することができる．

1.6 本研究における目的と本論文の概要

1.6.1 本研究の目的

本研究では，上記に挙げたロボットの外部情報と内部情報の取捨選択による学習の効率化を実現する手法を提案する．外部情報の取捨選択では，コミュニケーション相手の取捨選択による個体学習の効率化を実現する手法を提案する．内部情報の取捨選択では，ロボットが必要なセンサを自律的に判別することで，不要なセンサ情報を削減した Q 空間を基に学習を行い，効率的な学習を実現する手法を提案する．

1.6.2 本論文の概要

まず，第 2 章において，各アプローチで共通となる強化学習の概念や各手法について述べる．第 3 章，第 4 章において他のロボットからの情報の学習への利用について述べる．ロボット間で経験情報を共有することにより個体単体が入手可能な情報量を増やすことでより早く Q 値空間を更新でき学習速度が向上する．第 3 章ではこの概念が有効であるかを確認す

るために個体学習を促進するシステムの提案を行う．コミュニケーションされる情報および相手といったコミュニケーションの規定を全て人間が設定した場合において，コミュニケーションをせずロボット単体の学習との学習速度を比較する．その上で第４章では，コミュニケーション相手の取捨選択を行う手法を提案する．

第５章，第６章では，タスクに必要な環境情報の取捨選択による意思決定のための状態空間の構成について述べる．第５章では，タスクに応じて必要・不要センサを決定し，必要センサのみで状態空間を構成する手法について提案する．第６章では，第５章で提案した手法を改良し，必要・不要の２値ではなくセンサの重要度に応じて状態空間内の状態数を動的に変更する手法を提案する．提案する手法はいずれもタスクに応じて必要な分だけ状態を構成し，それにより不要な Q 値の更新が削減され学習効率が向上する．

1.6.3　本研究における適用タスク

本研究では，経路計画問題を適用タスクとして考える．経路計画問題は，強化学習を適用するタスクとして最もオーソドックスな問題であり，学習成果の確認を行いやすい．経路計画問題は，スタート地点とゴール地点間の最適経路を学習するものである．第３章においては，コミュニケーション学習の有効性の予備実験というスタンスであるため，状態の概念が無くより効果が分かりやすい N 本腕バンディット問題を対象タスクとして適用する．第４章～第６章までは，経路計画問題を対象問題としている．

1.6.4　本研究における効率化の評価項目

本研究では，一定期間内での獲得報酬と行動回数を効率化の評価項目とする．強化学習では報酬を基に学習を行う．報酬はある状態でとった行動がどれだけ適当かを表現するパラメータである．常に最適な行動をとり続けることができれば，ある期間内で考えた場合，獲得できる報酬量も増加する．学習が不完全な場合は，最適行動をとり続けることは難しいため，獲得報酬は少ない．一方，学習が終了している場合は，最適行動をとり続けることができるため，獲得報酬量は多くなる．ある手法を用いて学習を行い，従来手法に比べて学習が早く終われば，一定期間内でより多くの報酬を獲得することができるため，その手法は効率的な学習ができているといえる．そのため，本研究では，一定期間内に獲得できる報酬量を通常の強化学習と比較することで効率化ができているかどうかを考える．

また，経路計画問題においては，スタートからゴールに至るまでの行動回数が少ないほど学習が行われていると考える．学習が終了した時点で行動回数は収束する．そのため，行動回数の収束時点が通常の強化学習と比べて早ければ効率的に学習ができているといえる．

1.7　本論文の構成

図 1.4 に本論文の構成を示す．本論文は全７章で構成される．

第１章では，本論文の背景を述べた．ロボットの開発および学習を含めた制御手法の発達とセンサの関係について述べた．また本論文の対象とする問題領域および本論文の目的を述べた．

第２章では，本論文で用いる機械学習手法の１種である強化学習について概要を述べる．また，本論文で使用する学習手法について紹介する．

第３章では，コミュニケーションを基に行う学習が効果的なものであるかを確認する．そこで，他のロボットからの情報を自身の学習の効率化に利用するシステムを構築し，シミュレーションによって有効性を確認する．

第 4 章では，自身に有用な情報を持つ個体を取捨選択することで，コミュニケーションによる学習をより効果的にするシステムを考える．外部情報として入ってくる他者からの情報を基に，自身にとって有益な情報をもたらす他者を学習する．自身にとって有益な情報をもたらす他者からの情報を基に学習を行うことで，より効率的な学習を実現するシステムを提案する．

第 5 章では，群から個のロボットに立ち返り，このロボットが入手するセンサ情報を学習に最適化することを考える．ここでは，タスク遂行に重要な役割を持つセンサとそうでないセンサが存在することに注目する．タスク遂行に必要なセンサ，不要なセンサを判断し，学習に必要なセンサのみを用いることで，不要な入力情報を削減し学習を効率的にするシステムを提案する．

第 6 章では，第 5 章で提案したシステムの問題点を指摘し，解決を図る．ここでは，センサを必要不要の 2 値で判断するのではなくより柔軟なセンサの扱い方を考える．そのために，タスクに対するセンサの重要度に応じて状態空間を自律的に構築するシステムを提案する．

第 7 章において，本論文の結論および今後の課題を述べる．

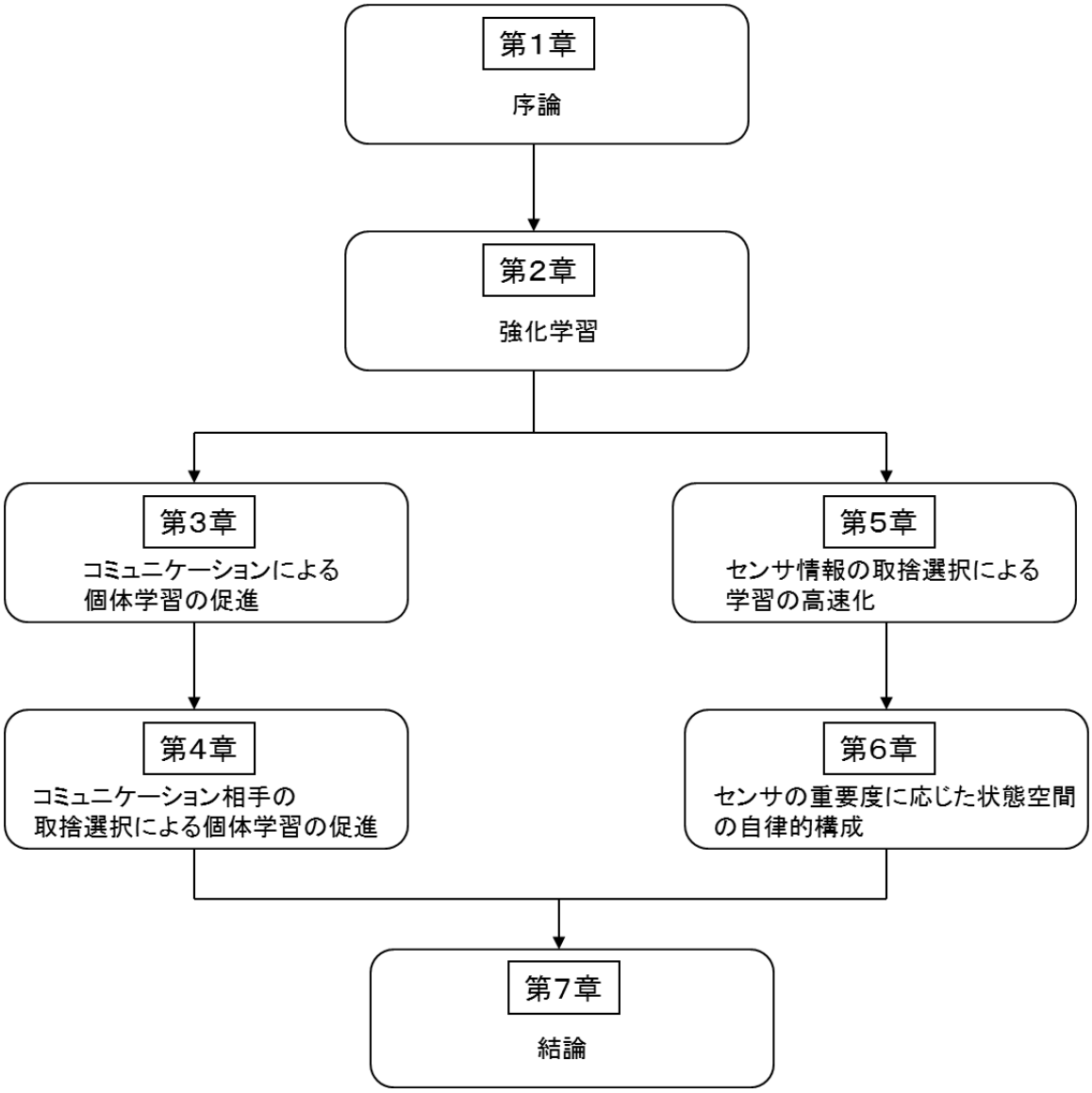


図 1.4: 本論文の構成

第2章　強化学習

本章では，本論文で用いる機械学習手法である強化学習 [94][95] について概要と各章での実験において用いる行動選択手法，行動評価手法について述べる．

2.1　強化学習の概要

2.1.1　環境とエージェントとの相互作用とエージェントの目的

強化学習では，エージェント（学習者）は周囲の環境状態（以降，状態）を認識し，その状態で目的を達成するためには何をすべきかを学習する．学習は，行った行動の良し悪しを数値化したものにあたる報酬を基にして行われる．報酬は環境によって与えられる．エージェントはどの行動がより高い報酬に結びつくかを探索し，得られる報酬の総和を最大化することを目的とする．学習の流れを以下の箇条書きに示す．

1. エージェントは時刻 t でセンサを通して知覚される環境の状態 s_t に基づいて意思決定を行い，行動 a_t をとる
2. エージェントの行動 a_t の結果として環境から報酬 r_t を受取る
3. エージェントの行動 a_t により，環境は状態 s_{t+1} へ遷移する

エージェントは環境に対してこのような状態の観測，行動，状態の変化，報酬獲得の獲得という一連の流れを繰り返す（環境との相互作用）ことで学習を行う（図 2.1）．

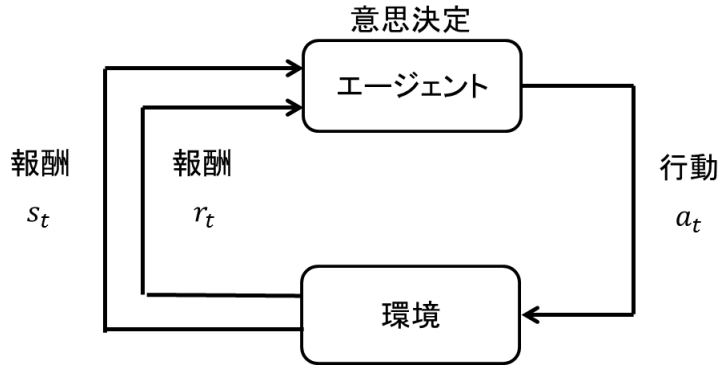


図 2.1: 環境との相互作用

先に述べたようにエージェントの目的は得られる報酬の総和を最大化することである．報酬はエージェントの設計者が設定する．したがって，目的を達成するような行動に対して高い報酬を設定しておくことで，目的に至るまでの行動のプロセスはエージェントが自動的に獲得する．

2.1.2 強化学習の特徴

強化学習の特徴としては以下のようなことが挙げられる．

- 報酬を基にした探索
- 遅延報酬に対応

報酬を基にした探索　強化学習では，学習のための正解情報を直接与えられることはない．その代わりに，エージェントは行った行動に対してその行動の良し悪しを報酬として与えられる．与えられた報酬が最良か最悪かはエージェントは知らされない．したがって，エージェントはどの行動がより高い報酬に結びつくかを試行錯誤により探索する．

遅延報酬に対応　強化学習では，試行錯誤により最終的に目的を達成した際に報酬が与えられることが多く，ある時点でエージェントが選択した行動の良し悪しの判定には時間的な遅れが存在する．このことを遅延報酬と呼ぶ．

2.1.3 応用上されていること

文献 [96] によると以下のようなことが応用上期待できる．

制御プログラミングの自動化・省力化

実環境では，環境の不確実性や計測不能な未知のパラメータが存在する場合，ロボットの設計者にとってタスクの達成方法を完全に把握するのは難しい．したがって，ロボットに対してタスク遂行のための行動を規定することはロボットの設計者にとって大きな負担となる．一方，達成すべき目標を報酬によってロボットに明示することは前記に比べれば遥かに簡単である．タスク達成のための行動を強化学習によりロボットが自動的に学習することで，設計者の負担軽減が期待できる．また，十分に優れた性能を持つ強化学習エージェントをコントローラとして１つだけ開発しておけば，あとはロボットの目的に応じて報酬の与え方だけをロボットの設計者が設定するだけで，あらゆる種類のロボット制御方法を同一のコントローラによって自動的に獲得できる．

ハンドコーディングよりも優れた解

試行錯誤により学習するため，人間のエキスパート（専門家）が得た解よりも優れた解を発見する可能性がある．特に不確実性（摩擦やガタ，振動，誤差など）や計測が困難な未知パラメータが多い場合，人間の予測のみでは対処しきれないことが予想される．このような場合に対して，強化学習は有効であると考えられる．この新しい解の発見には 2 つのアプローチが存在する．1 つは，エキスパートの制御規則を学習初期状態に設定して，それを改善する方法である．そしてもう一つは，全くのゼロから学習を開始し，設計者にとっては意外な新しい解を発見する方法である．

自律性と想定外の環境変化への対応

機械故障などの急激な変化やプラントの経年変化のような緩慢な変化など，予め事態を想定してプログラミングしておくことが困難な環境の変化に対しても追従する．特に宇宙や海底など，通信が物理的に困難な極限環境の場合や，通信ネットワークの制御のように現象のダイナミクスが人間にとって速すぎる場合において，強化学習の自律的な適応能力が有用である．

2.1.4　強化学習の構成要素

ここでは強化学習の構成要素である，環境・行動学習手法（方策）・報酬関数・評価関数・行動学習手法について解説する．

・環境

環境はエージェントがセンサを通して知覚することができる全ての状態を内包している．例として，家庭環境を考える．家庭環境には机や椅子のような家具，リモコンや食器といった小物類といったものが存在する．これらの家具や小物は置いてある場所が変わったり，経年劣化によって外見がくすんでくるなど絶えず状態が変化する．強化学習における環境では，家庭環境はこのような絶えず変化する状態を全て内包していると捉える．エージェントはセンサを介して自身が直面している環境の一部を認識する（2.2）．エージェントが持つセンサには認識できる環境の範囲が存在する．エージェントはセンサの認識できる範囲の環境しか認識することができない．したがって，エージェントが認識する環境は環境の一部分になる．このセンサを介して認識できる一部の範囲のことを状態と呼ぶ．また，環境はエージェントの行動の結果，どのように状態が遷移するかといったルール情報も保持している．したがって，エージェントがある状態である行動をとった時に次に遷移する状態が予測できる．

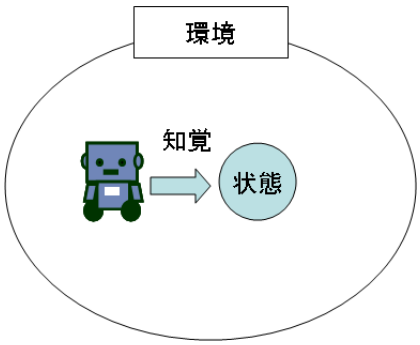


図 2.2: 環境と状態

・行動選択手法（方策）

行動を決定する手法である．例えば，ある状態で行動を決定する際にその行動をランダムに決定する手法，今までで最も高い報酬を得られた行動に決定する手法，過去の報酬の累積から確率的に決定する手法，というように行動の決定のルールが行動選択手法である．

・報酬関数

報酬関数は強化学習問題において目標を定義する．目標は設計者がエージェントに学習させたい状態や行動である（例えば，平均台でバランスを取る，歩くなど）．この関数は状態行動対に対して報酬という数値情報を出力する．報酬は現在の状態における，その状態にあることの望ましさを表している．ゆえに，報酬関数は即時的な意味合いでエージェントにとってなにが良いのか示している．一般に報酬関数は設計者が設計するものでエージェントが変更することはない．

・価値関数

報酬関数が即時的な意味合いで何が良いのか示しているのに対して，価値関数は，最終的な状態または行動の価値を決定する．価値とは，エージェントがその状態を基点として将来にわたって入手できる報酬の期待値である．報酬はその環境が即時的で固有の望ましさを決定するのに対して，長期的な望ましさを示すものである．例えば，ある状態では常に低い報酬しか得られないかもしれないが，高い報酬が得られるような状態が続くのならば，高い価値を持つ．

・行動学習手法

価値関数は報酬関数を基に更新されてゆく．行動学習手法は報酬関数をもとに価値関数を更新する手法である．例えば，過去に得られた全ての報酬の平均といったように，どのように価値関数を更新するかを規定したものが行動学習手法である．

2.1.5 強化学習の流れ

強化学習の流れを図 2.3 に示す．環境から知覚した状態 s によって，エージェントは自身が行うことのできる行動の中から，その状態 s における行動価値 Q に基づき行動選択手法を用いて行動 a を選択（意思決定）し，実行する．その結果，環境より得られた報酬 r を基に，エージェントは状態 s において選択した行動 a の価値 $Q(s,a)$ の更新を行動評価手法によって行い（学習），次回同様の状態においての行動選択に生かす．エージェントは，行動価値 Q を基にして行動選択手法を用い行動を決定する．行動の結果受け取る報酬を基に行動価値 Q を行動学習手法により更新する．行動の決め方を規定する行動選択手法と行動価値の更新の仕方を規定する行動学習手法によって学習する．したがって，強化学習の学習法は行動選択手法と行動学習手法の 2 つの組み合わせによって決定する．

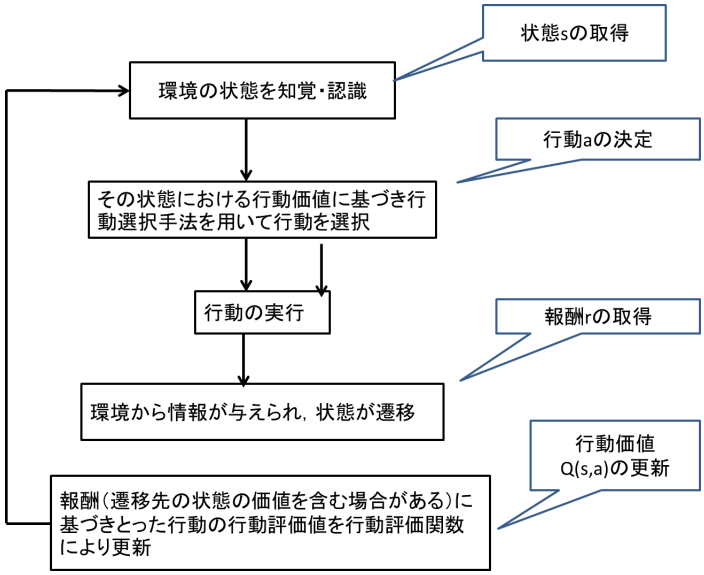


図 2.3: 強化学習の流れ

2.2 行動選択手法

強化学習における行動選択手法とは，エージェントが認識した状態 s においてとる行動 a を選択する際に用いられる手法である．ここでは，本論文で用いる主な行動選択手法について述べる．本論文では，行動選択手法として greedy 法， ε -greedy 法，softmax 法，追跡手法を扱う．

2.2.1 greedy 法

最も高いと推定された行動価値を持つ行動（あるいは行動群から 1 つ）を選択する（図 2.4）．この方法は常に即時の報酬を最大にするために，現在の行動価値を利用するものである．すなわち，価値が低いと判断される行動に対しては，それが本当はさらに良いかもしれないという可能性を確かめる目的での試行を一切行わないという性質がある．

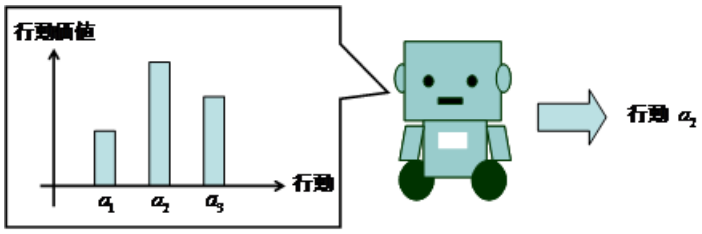


図 2.4: greedy 法

2.2.2 ϵ-greedy 法

ϵ-greedy 法は，基本的には推定される行動価値が最も高い行動（グリーディな行動）を選択するが，たまに小さい確率 ϵ で行動価値の高さとは無関係にランダムで行動を選択する手法である（図 2.5）．常に行動評価値の最も高い行動しか行わない greedy 法とは異なり，確率 ϵ で探索行動を行う．しかし，確率 ϵ における行動選択の際にほとんど最悪と思われる行動とほとんど最適に近い行動を選択する可能性が同程度であるという欠点がある．

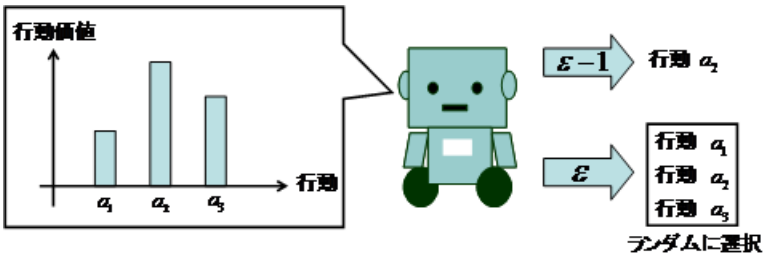


図 2.5: ϵ-greedy 法

2.2.3 softmax 法

softmax 法は，行動価値を等級付けした関数によって行動確率を変化するやり方である．すなわち，行動価値の最も高い行動には最も高い選択確率が与えられ，他のすべての行動は，その推定価値に従って重みをかけられ，ランク付けされる（図 2.6）．Softmax 法では一般に，Gibbs 分布 [102]，あるいは Boltzmann 分布 [103] が使われる．具体的には，t 回目のプ

レイにおける行動 を選択する確率 は式 (2.1) で与えられる．ここで，$\pi_t(s,a)$ は時間 t，状態 s で行動 a を選択する確率，$Q_t(s,a)$ は時間 t，状態 s で行動 a を選択したときの行動価値である．τ は温度と呼ばれるパラメータでこの値の大小で重みのつけ方が変わる．τ を小さくすると各行動推定価値の差が少しでも行動選択確率は大きく異なる．逆に，τ が大きいと各行動の推定価値の差が大きくても行動確率の差は小さくなる．つまり，τ が小さいと確定的になり，τ が大きいと確率的になる．τ の大小の基準は報酬の大きさとに依存するため，τ はタスクごとに設計者が綿密に定めることが望ましい．

$$\pi_t(s,a) = \frac{e^{Q_t(s,a)/\tau}}{\sum_{b=1}^{n} e^{Q_t(s,b)/\tau}} \tag{2.1}$$

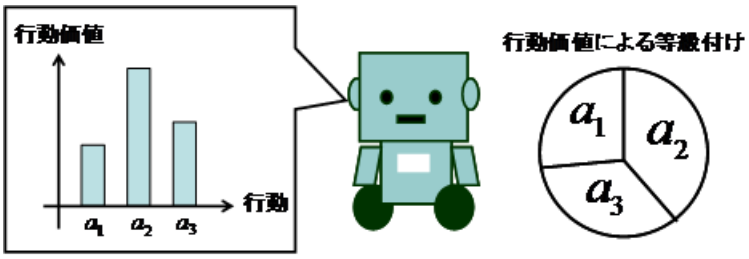


図 2.6: softmax 法

2.2.4 追跡手法

追跡手法は，現状態において最も高い行動価値を持つ行動の選択確率を増加させ，その他の行動については選択確率を減じる (図 2.7)．よって，静的な環境の場合は，最も高い行動価値を持つ行動選択確率は 1 に向かって増加し，その他の行動選択確率は 0 に向かって減少する．つまり，静的な環境の場合には時間が経つにつれて学習者の行動は greedy なものになっていく．時間 t，状態 s における各行動の選択確率の算出方法について説明する．最大の行動価値をもつ行動 $a^\star$ を選択する確率 $\pi_t(s,a^\star)$ は式 (2.2) で計算される．β_p は追跡手法における各行動確率の増減割合を決定するパラメータであり $0 \leq \beta_p \leq 1$ の範囲をとる．β_p が 0 に近ければ各試行での行動選択確率の変化量が少ないため，各行動があまり偏りなく選択される．そのため，将来の獲得報酬を視野にいれた長期的な視点で学習を行う際には有効になる．一方，β_p が 1 に近い場合には，各行動確率の変化量が大きくなるため，各行動の選択確率に大きな偏りが生じやすくなる．そのため，少しでも行動価値の高い行動は数試行で選択確率が大きくなる．これは，短期的な報酬を重視した行動選択を行う際に有用である．

$$\pi_t(s,a^\star) = \pi_{t-1}(s,a^\star) + \beta_p[1 - \pi_{t-1}(s,a^\star)] \tag{2.2}$$

$$\pi_t(s,a) = \pi_{t-1}(s,a) + \beta_p[0 - \pi_{t-1}(s,a)] \tag{2.3}$$

2.3 行動評価手法

強化学習において，エージェントは行動の真の価値そのものを知ることはできないため，毎回の行動によって得られる報酬からその行動の真の価値を推定する．そして，その推定値

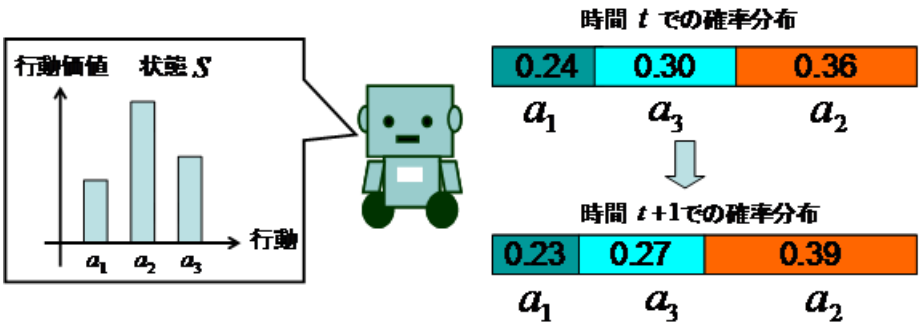


図 2.7: 追跡手法

を使って行動選択手法を通して行動を選択する．この行動の真の価値を推定するための方法が行動評価手法である．ここでは，行動評価手法として，標本平均手法，加重平均手法，Q 学習法の 3 手法について述べる．

2.3.1　標本平均手法

標本平均化手法では，その行動が選ばれたとき実際に得られた報酬を平均化してゆく．時間 t，状態 s，行動 a について標本平均手法を用いた時の行動の価値 $Q_t(s,a)$ は式 (2.4) で更新する．分子は過去に状態 s で行動 a をとったときに得た報酬の総和である．また分母は状態 s での行動 a の累積選択回数である．$k_{s,a}=0$ の場合には，$Q_t(s,a)$ を $Q_0(s,a)=0$ のような初期値にする．大数の法則より，$k_a \to \infty$ の極限において $Q_t(s,a)$ は真の価値 $Q^*(s,a)$ に収束する．

標本平均化手法は定常環境での動作に適したものである．しかし，非定常環境ではあまり有効とはいえない．これは，標本平均化手法では，より多くの試行を行なうほど結果が反映されにくくなるためである．試行回数が増えることでその行動の選択回数が増え，分母の値が大きくなる．その結果，最新の報酬が評価値に影響を与えにくくなる．このため，非定常環境下での評価関数はより最新の報酬に対し重みをおいて評価するといった工夫が必要になってくる．そのような手法が次節で紹介する加重平均手法である．

$$Q_t(s,a) \leftarrow \frac{r_{s1}+r_{s2}+\cdots+r_{sk}}{k_{s,a}} \tag{2.4}$$

2.3.2　加重平均手法

加重平均手法は，遠い過去の報酬よりも最近に受け取った報酬の方により重みを与えるような方法である．重みを与えるために，定数値のステップサイズ・パラメータを使用する．時間 t で状態 s において行動 a をとり報酬 r_t を受け取ったときの行動価値 $Q_t(s,a)$ の更新式は式 (2.5) のようになる．ここで，α はステップサイズパラメータ $(0 \le \alpha \le 1)$ である．ステップサイズ・パラメータが大きいほどより最近報酬情報を重要視するようになる．

$$Q_t(s,a) \leftarrow Q_t(s,a) + \alpha[r_t - Q_{t-1}(s,a)] \tag{2.5}$$

2.3.3　Q 学習法

標本平均手法，加重平均手法ともに行動の都度入手される報酬を基に評価を行なう．これは行動ごとに報酬が得られる環境に対してしか適用できないということである．したがって，

目標状態に対してのみ割り振られるタスク（遅延報酬）の場合，目標状態までに行った各行動に対して行動価値が割り振られることはない．例えば，ゴールのみで報酬が与えられる迷路問題を考える．このよな迷路問題ではゴール以外の経路に対して報酬が割り当てられていないため，標本平均手法や加重平均手法を用いた場合，学習者はゴール手前の状態に対しての行動を評価することはできるが，それ以外の状態に対する各行動の評価を行うことができない．そのため，遅延報酬タスクに対して標本平均手法，加重平均手法では学習が進まない．

Q 学習法は，このような遅延報酬環境下でも利用できる手法である．Q 学習では，現在の状態で選択した行動の価値とその行動の結果，推移した先の状態の行動価値によって現在の行動価値を更新する (図 2.8)．迷路問題の場合，ゴール時にもらえる報酬価値をスタートまでのルートに対し伝播させることが可能になる．これによりスタートからゴールまでのそれぞれの状態に対して，その状態の価値が算出されるため学習が可能となる．

時間 t で状態 s において行動 a をとり報酬 r_t を受け取ったときの行動価値 $Q_t(s,a)$ の更新式は式 (2.6) のようになる．ここで α はステップサイズパラメータ $(0 \leq \alpha \leq 1)$，γ は割引率 $(0 \leq \alpha \leq 1)$ である．

$$Q_t(s,a) \leftarrow Q_t(s,a) + \alpha[r_t - \gamma Q_{t+1}(s,a) - Q_t(s,a)] \tag{2.6}$$

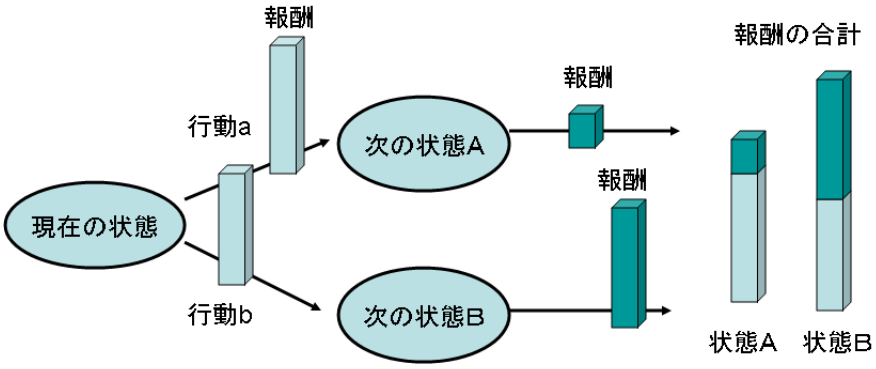


図 2.8: Q 学習法

2.4 強化比較法

強化比較法では，行動価値を用いない．その代わりに，リファレンス報酬という値を用いる．リファレンス報酬は，獲得した報酬が高いのか低いのかを判断するために用いる基準となる報酬である．基本的には過去すべての状態行動対において獲得した報酬の平均である．このリファレンス報酬によって，現在獲得した報酬が高いのか低いのかを判定する．現在獲得した報酬がリファレンス報酬よりも高い場合，その報酬を獲得した行動に対する選択確率を上げる．逆に現在獲得した報酬がリファレンス報酬よりも低い場合，その報酬を獲得した行動に対する選択確率を下げる．

強化比較法では，各行動に対して行動選択確率の他に行動の優先度を設けている．各行動の行動選択確率は行動の優先度を基に決定する．行動の優先度は獲得報酬とリファレンス報酬との差によって算出される．行動優先度 $p_t(s,a)$ の更新式を式 (2.7) に示す．

$$p_t(s,a) = p_{t-1}(s,a) + \beta_r[r_{t-1} - \overline{r}_{t-1}] \tag{2.7}$$

ここで，$\overline{r}_{t-1}$ はリファレンス報酬である．β_r はステップサイズ・パラメータで，$0 < \beta_r < 1$ である．β_r が高い場合，リファレンス報酬との差が大きい報酬ほど優先度の変化量が大き

くなる．これにより，学習者は探索的な行動よりも知識利用を重要視するような行動選択を行うようになりやすくなる．

行動の選択は，通常 softmax 手法 (2.2.3) が用いられる．行動選択確率決定式を式 (2.8) で決定する．

$$\pi_t(s,a) = \frac{e^{p_t(s,a)}}{\sum_{b=1}^{n} e^{p_t(s,b)}} \tag{2.8}$$

リファレンス報酬の更新は加重平均手法を用いて式 (2.9) を用いて行う．α_r はステップサイズ・パラメータで $0 \leq \alpha_r \leq 1$ の範囲を取る．

$$\overline{r}_t(s,a) = \overline{r}_{t-1} + \alpha_r[r_{t-1} - \overline{r}_t] \tag{2.9}$$

2.5 まとめ

本章では，本論文で扱う学習法である強化学習について概要を説明した．また本論文で使用する行動選択手法，行動評価手法について基本的な手法を説明した．

第3章　コミュニケーションによる個体学習の促進

本章では，コミュニケーションによって得た他者からの情報を学習に利用する．他者からの情報によって自身の Q 空間を改変し，学習を効率化する．これを N 本腕バンディットシミュレーション実験を通して確認する．

ロボットには，他のロボットからのメッセージをカメラ，マイクや感圧センサといった，人間も持ちうるセンサから得る以外にも，他のロボットとの通信を行うものがある．例えば，赤外線ポートや無線 LAN のポートなど，他のロボットに対して情報の送受信を行うものがある．本章では，コミュニケーションが学習効率に与える影響の調査が主であるため，こうしたセンサの違いによるコミュニケーションの手法の違いについては論じない．

3.1　センサとしてのロボットの通信装置

ロボットはセンサを通して外界から情報を取り入れる．センサは人間で言うところの感覚器官に当たる．例えば，カメラは視覚情報，マイクは聴覚，感圧センサは触覚にあたる．ロボットは赤外線や X 線といった人間では感知できない情報も感知することができる．さらに，ロボットは人間とは違い通信のための装置を持つ．この通信のための装置は，他のロボットから送られる情報を知覚するためのセンサであるといえる．このセンサにより他のロボットとコミュニケーションができる．他のロボットとコミュニケーションを行うことで，自身が未だ知らない情報を入手することが可能となる．

3.1.1　学習時間問題の解決策としての他のロボットからの情報の利用

ロボット外部の情報として，他のロボットのもつ情報に注目する．自身の経験情報に加えて他のロボットが持つ情報を学習に利用することで，個体単体が保持する情報を増やす．これは，自身が学習すべき空間の探索しきれていない部分を他者からの情報で補完するということである．その結果，個体単体の時よりも高速にかつよりよい解にたどり着くことが可能になる．

3.1.2　従来の群ロボット問題との違い

従来の群ロボット研究では，各個体が協調して 1 つの目的を達成する．例えば，分散センシング [72]-[75] や物体搬送 [76]-[78], 群ロボットが集まり 1 体のロボットして行動制御を行う研究 [79][80] などである．分散センシングの場合，各個体の得た観測情報を交換し，統合することで環境全体のモデルを獲得する．物体搬送の場合，個体単体では運搬困難な物体を複数台のロボットと協調することで，運搬を可能とする．従来の群ロボットにおける個体の学習対象は，目的達成のための他者と協調ための一連の行動である．分散センシングであれば，他のロボットの探索情報から未探索の領域を探索する行動を行う．物体搬送では，他のロボットの行動を基に目的地へ向かうように行動を行う．群ロボットが集まり 1 体のロ

ボットとして行動を制御する研究では，1 体のロボットしての行動をロボットを構成する各モジュールロボット間での協調により生み出す．

しかし，本章で対象としているのは，個体の学習である．すなわち，自身の目的達成のために他者と協調を行う．従来の群ロボットとは違い，群としての目的はもたず，各個体がそれぞれに達成すべき目的を持つ．そして，他のロボットとの協調し情報を共有することで，自身の目的を達成する方法を効率的に獲得する．つまり，自身の発達のために他のロボットと協調するという立場である．

3.2 コミュニケーションによる個体学習促進システム

3.2.1 単体の学習システムと提案するシステムの違い

本章では，他のロボットとの情報共有手段としてのコミュニケーションを考える．従来の学習を図 3.1 に，今回提案する他のロボットとの情報共有による学習を図 3.2 に示す．図に示す通り，意思決定に用いる情報量が異なる．従来の個体単体での学習では，自己が獲得した経験情報を基に各種機械学習手法によって意思決定が行われ，行動として出力される．これに対し，コミュニケーションを用いた学習では，自己が獲得した情報に加えて他者からコミュニケーションによって得られた情報を加えて意思決定を行う．個体単体での学習に比べて意思決定に用いる情報量が多くなるため，センサからの情報に対してより適切な行動を行うことが可能となる．

図 3.1: 個体単体での学習

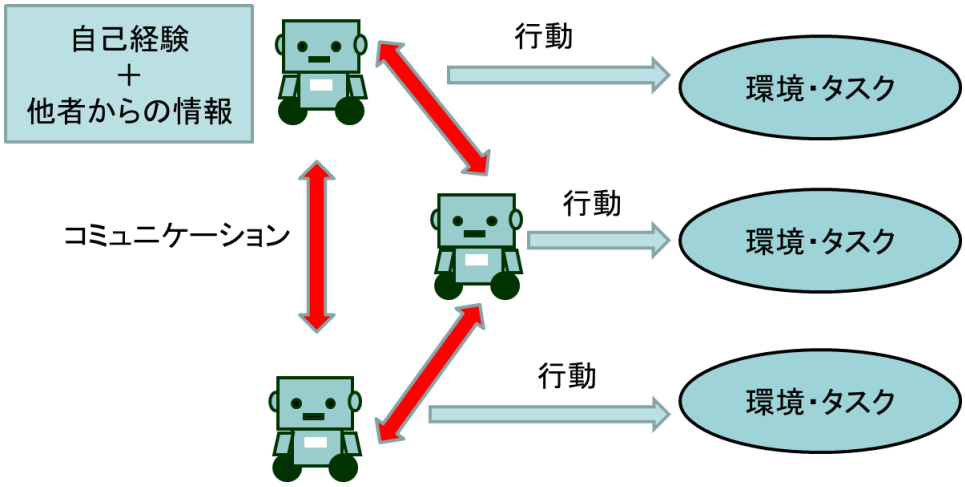


図 3.2: コミュニケーションを活用した学習

また，提案するシステムは，個体がより効率よく学習するために，他のロボットとのコミュニケーションによる情報を利用するという立ち位置のものである．そのため，他のロボットの存在が前提となる，すなわちコミュニケーションする相手が存在しない場合に学習が進ま

なくなるようなことがないようなシステムが望ましい．よって，自身の周囲にコミュニケーションすべき他のロボットが存在しない場合は，自身の経験情報のみを基に学習を行い，他のロボットが存在する場合はコミュニケーションを行い情報共有することで，効率的な学習を実現するシステムの構築を目指す．(図 3.3)．

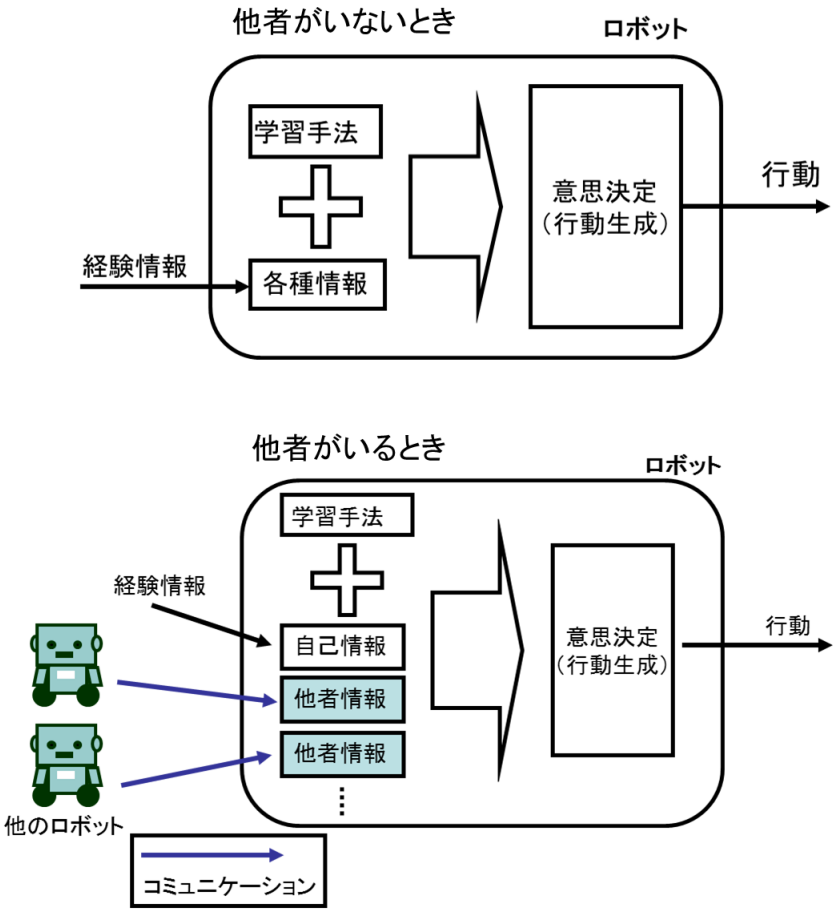


図 3.3: 目指すシステム

3.3 コミュニケーション情報に関する考察

本システムで重要なのは，コミュニケーションした結果得た他のロボットからの情報（他者情報）の扱い方である．そこで，コミュニケーションに際し個体間で交換される情報について考察する．

3.3.1 コミュニケーションで交換される情報において考えるべきこと

・ 扱うことのできる情報について

ロボットが扱うことのできる情報は，予め開発者によって決められたフォーマットに従って記述された情報だけである．そのため，フォーマットが異なるロボット間のコミュニケーションは成立しない．例えば，カメラ情報ならば，RGB や輝度といったデータの記述の順番がロボット間で異なっている場合は，全く異なるフォーマットとなる．コミュニケーションを成立させるためには，コミュニケーションを行うロボット間で共通のフォーマットを用いるか，自身に扱えるように他のロボットから送られてきた情報を自身のフォーマットの形式に変換する必要がある．

・ 利用出来る情報について

コミュニケーション相手の扱う情報のフォーマットが自身と同一で，扱うことができたとしても有効に活用できるとは限らない．例えば，温度情報の場合を考える．自身の利用可能な温度情報がセ氏で，他者からの情報もセ氏で記述されていれば，その他者からの情報は，自身が利用可能な情報であるといえる．他者情報は，形式が自身と同一でも，情報が利用可能な形で記述されていない場合がある．そのような時は，情報を自身が利用可能な形に変換する．例えば，他者情報は絶対温度であり，自身が扱う情報がセ氏であった場合，温度の単位が異なるため，情報をそのまま利用すると悪影響が出る可能性がある．そのため，温度情報を絶対温度からセ氏に変換して利用するといったようにロボット間で利用可能情報にする必要がある．

・利用の仕方について

他者からの情報は，自身の試行錯誤によって得た情報と合わせて意思決定に使われる．自身の得た情報と他者情報を合わせるには，情報を処理する必要がある．情報の処理の仕方は，タスクの種類と目的によって異なる．情報の処理の仕方が個体間で異なった場合，コミュニケーションする情報によって評価が異なる場合がある．目的が個体間で異なっている例として，図 3.4 のような迷路タスクで考える．一方は，迷路中に落ちている回収物を出来るだけたくさん獲得してゴールすることが目的である．もう一方は最短ルートでゴールすることが目的である．コミュニケーションする情報を自身にとって最適のルートの情報とし，この両者でコミュニケーションした場合，各ロボットの最適なルート情報は，互いに悪いルートである．そのため，価値の低い情報となる．このようなことを回避して他者情報を利用するためには，相手のタスク・目的に合わせて，自身の他者情報の処理方法を考えなくてはならない．先の迷路タスクを例にすると，相手の最適なルート情報は，自身にとって通るべきではないルートとして利用する．

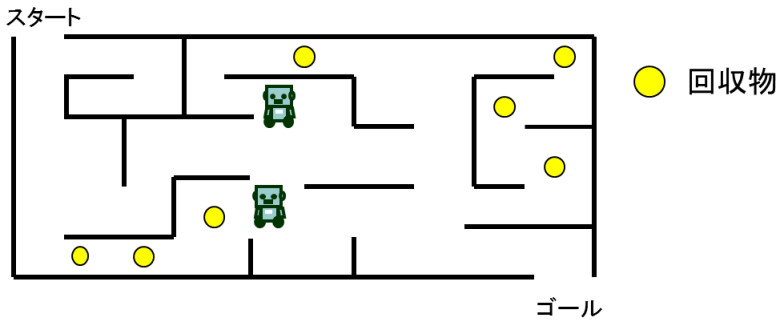


図 3.4: 迷路問題の例

3.3.2 提案システムで用いるコミュニケーション情報

ロボット間でのコミュニケーションの条件についての考察

情報のフォーマットについては，コミュニケーションをする個体間で同一なのかそうでないのかによって受け取った他のロボットからの情報を扱うことが出来るかどうかが決まる．他のロボットからの情報を扱えるようにするためには，情報の形式がロボット間で同一である必要がある．そのためには，ロボットの身体構造が同じであることがあげられる．身体構造が同じであれば，情報のフォーマットも同一となる．

しかし，身体構造が同じであっても，個体固有の要素が存在する．個体固有の要素とは，身体の大小のようなものである．これがコミュニケーションに影響を及ぼす可能性がある．例えば，身体大きさが異なるヒューマノイドロボット同士のコミュニケーションを考える．身体の小さい方が通れる道があり，その情報を身体の大きいロボットに伝えるとする．このとき，身体大きいロボットがその道を通ろうとしても通れない．このように，個体固有の要素がコミュニケーションを利用した学習に悪影響を及ぼす場合がある．そのため，今回はこのような個体固有の要素をなるべく排除した情報をコミュニケーションに利用することを考える．

利用可能な情報について考えると，他のロボットから得られた情報を自身が利用可能でなければ，自身が利用可能な形に変換必要があるため，その分難しくなる．そのため，提案システムでやり取りされる情報は利用可能な形に変更不要なものを扱う．

情報の処理の仕方については，タスクの種類・目的によって異なってくる．コミュニケーション相手とタスクの種類または目的が異なると，相手ごとに情報の処理方法を考えなければならず，コミュニケーションが難しくなる．逆にタスクの種類・目的が同じであれば，コミュニケーションする相手ごとに情報の処理方法を考える必要がないため，コミュニケーションが簡単になる．そのため，今回はタスクの種類・目的が同じロボット間でのコミュニケーションを考える．

整理すると，今回のコミュニケーションのためのロボットの条件としては以下を挙げる．

- 情報のフォーマットが個体間で同一，つまり身体構造が同一であること
- 個体固有の情報を含まないものであること
- 情報は個体間で特に変換の必要なく利用可能であること
- タスクの種類・目的が個体間で共通であること

コミュニケーションに用いる情報

先に提示したコミュニケーションのためのロボットの条件をもとに本論文においてコミュニケーションする情報を考える．本章ではコミュニケーション情報に「学習法」を採用する．学習法は文字通り学習の進め方である．学習法は大きくみれば，各種機械学習手法であり，小さくみれば，例えば強化学習における標本平均手法や加重平均手法といった各種行動評価手法である．学習法には得手不得手がある．例えば，NN は連続的な運動を学習することに優れているが，迷路問題のようなゴールまでのルートといったような行動計画の学習には不向きである．強化学習は逆に行動計画の学習に向いている．このように学習法は学習するタスクに依存して，得手不得手が存在する．また，学習法は一度試してみないとその学習法が現在のタスクに適切であることを判断することはできず，タスクに対して最適な学習法の特定には時間を要する．よって，他のロボットと学習法に関する情報を共有することは，個体単体よりも効率的に最適な学習法を発見が見込まれるため有用である．

コミュニケーション情報は与えられているタスクに適した学習法について自身の最もよいと判定した手法とする．この情報は，情報フォーマットを個体間で同一のものを用いても問題が出ない．方法に関しての情報なので，個体固有の情報を含まず，特に変換の必要がない．以上のことから学習法に関する情報をコミュニケーションする．

3.4 学習法をコミュニケーション情報とした個体学習促進システム

学習法をコミュニケーション情報とした個体知能の発達方法の概念図を図 3.5 に示す．この概念図より，個体の学習は，行動学習部と学習法学習部の 2 つの部分で構成する．学習法

学習部と行動学習部の関係を図 3.6 に示す．学習法学習部では，現在のタスク・環境に対して適切な学習法を学習する．また行動学習部はセンサからの入力に対してどのような行動が適切かを学習する．行動学習部で用いる学習法は学習法学習部で決定した学習法となる．また，学習空間は行動学習部と学習法学習部とでそれぞれ独立したものを保持する．つまり，ロボットは学習法と行動の 2 つの学習を同時に行う．

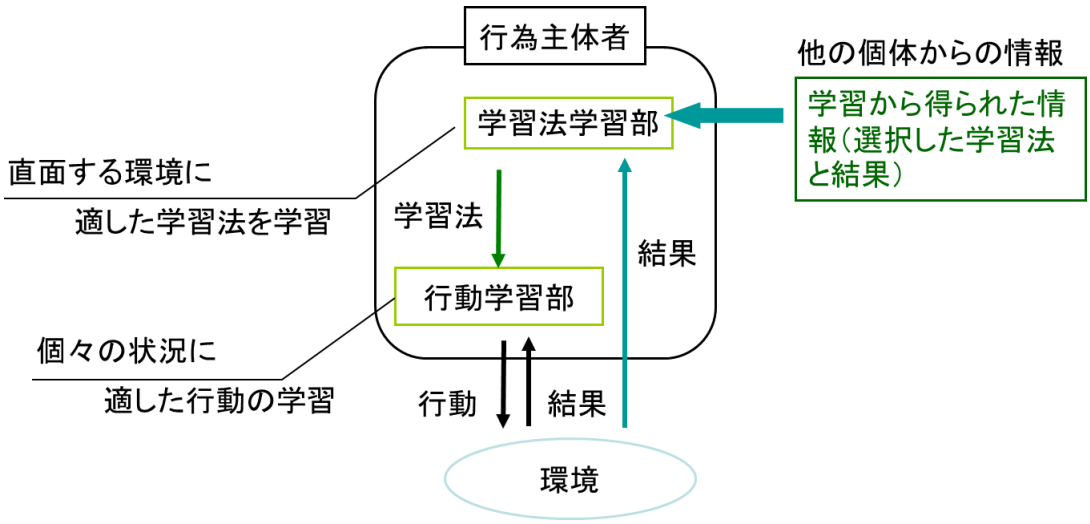


図 3.5: 学習法をコミュニケーション情報とした個体学習促進システムの概念図

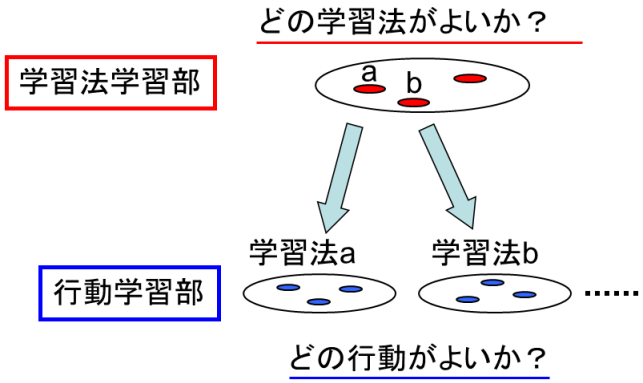


図 3.6: 学習法学習部と行動学習部の関係

行動学習部

行動学習部では，学習法学習部で決定した学習法を用いて，センサからの入力に適した行動を学習する．ロボットは各学習法固有の学習空間から行動の選択および行動の結果のフィードバックを行う．行動学習は以下の流れで行う．

1. 学習法学習部で決定した学習法に従い，学習空間から次に行う行動を選択する．
2. 選択した行動を実行し，結果を得る．
3. 学習法学習部で決定した学習法に従い，結果を学習空間にフィードバックする．
4. 1 に戻る．

これらの手順を繰り返すことで，ロボットは直面する環境に適した行動を学習する．

学習法学習部

学習法学習部では，自身の直面する環境に適した学習法を学習する．学習は以下の流れで行われる．ここでは，自身の経験に加えて他のロボットからの情報も学習に用いる．コミュニケーションによる他のロボットからの情報は，他者が行った学習法とその結果である．

1. 他のロボットとコミュニケーションを行い情報を取得する．
2. 他のロボットからの情報と自身の学習空間から，学習法を決定する．
3. 決定した学習法を用い，行動学習部で行動を学習する．
4. 選択した学習法を適用し，結果を得る．
5. その結果から決定した学習法に対してフィードバックを行い，学習空間を更新する．なお，このフィードバックに用いる情報は行動学習部で得られた結果と同じものを用いる．

学習全体の流れを以下に示す．

1. 他のロボットとコミュニケーションを行い情報を取得する．
2. 学習法学習部で自身の知識に基づいて学習法を決定する．
3. 決定した学習法を適用し，行動学習部で行動の選択が行われる．
4. 選択した行動を実行し，結果を得る．
5. 結果から，学習法決定部では決定した学習法を評価し，行動学習部では，選択した行動を評価する．学習法決定部，行動学習部での評価が自身の知識となる．

3.5 強化学習を適用した提案システムの概要

本章では，システム全体の学習手法として強化学習を適用する．強化学習を適用した提案システムの概念図を図 3.7 に示す．強化学習を適用したシステムの個々の部分について解説する．

学習法学習部

強化学習における学習法は，行動選択手法と行動評価手法の対となる．第 2 章で取り扱った通り，行動選択手法と行動評価手法はそれぞれ独立であり，これらを組み合わせることで強化学習は成り立っている．この組み合わせによって，得手不得手の環境がある．例えば，ε-greedy ＋加重平均手法ならば，ある程度環境が変化するような場合でも素早く適応することができる． ε-greedy 法の ε によって探索行動がとれ，また加重平均手法により評価が最近の結果に対して評価されるためである．また，ε-greedy 法＋標本平均手法であれば，静的な環境で有効である． ε-greedy 法で最適解の探索が可能で，標本平均手法によって 1 回の状態行動対の経験で報酬期待値を正確に算出することができる．一方，静的な環境の場合，加重平均手法では報酬期待値を正確に算出まで，何回か同じ状態行動対を経験する必要がある．この差が，学習収束に影響する．このように，学習法の組み合わせには適切不適切が存在する．よって，今回学習法学習部では，行動選択手法と行動評価手法から幾つかの組み合わせを作成し，ロボットが環境に適した学習法を学習する．なお，例外的に強化比較法は行動評価手法を必要としないため，それ単体で一つの学習法とする．また，コミュニケーションでやり取りされる情報に関しても強化学習に固有の情報となる．コミュニケーション情報を以下に示す．

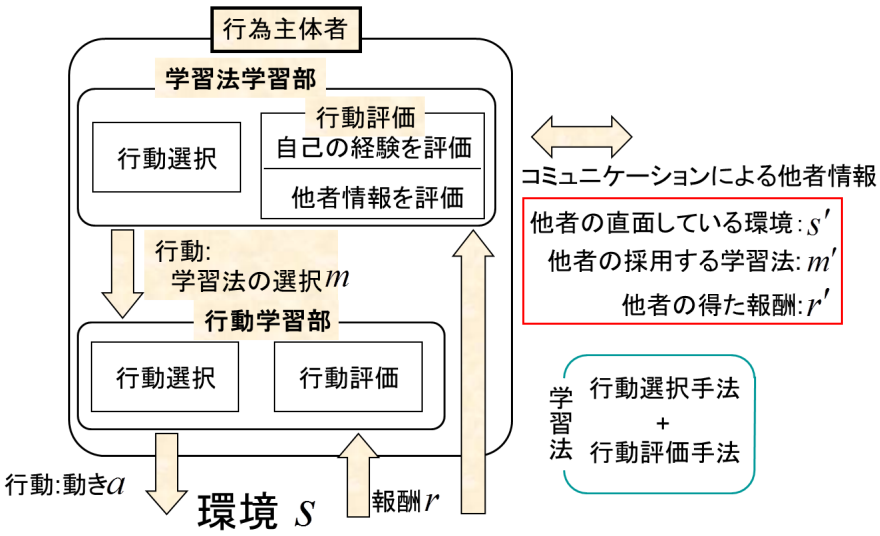


図 3.7: 強化学習を適用した提案システム

- 他のロボットの現在の状態
- 他のロボットが採用している学習法
- その学習法によって得た報酬

コミュニケーションで入手した情報は，相手の状態とその時の学習法によって得た報酬である．ロボットこの情報を基に自身の学習空間を更新する．つまり，学習法学習部では，自身の報酬による学習法の評価に加え，他のロボットからの情報も自身の報酬と同様に評価をしている．この時の評価手法は強化学習の任意の評価手法を採用する．例を図 3.8 に示す．これは加重平均手法の例である．この図では，学習空間を構成するセンサ軸 M と行動軸 S から構成される．この中の四角柱は各状態の Q 値を表している．他者からの報酬情報と自身の Q 値との差分を算出し，それに割引率を掛けたものを自身の Q 値に加えることで，他者からの情報を自身に反映させる．

行動学習部

行動学習部では，学習法学習部で決定した行動選択手法と行動評価手法の組み合わせを使って行動学習を行う．行動学習部での学習空間は，学習法ごとに用意せず 1 つの空間を共有する．Q 空間の更新手法と行動選択方法が学習法であり，Q 空間そのものとは完全に独立したものである．そのため，単一の Q 空間に対して様々な学習法を試していると考えられる．

3.6 N 本腕バンディット問題における提案システムの有効性の確認

3.6.1 N 本腕バンディットとは

N 本腕バンディット問題について説明する．学習者は N 本の腕（レバー）のあるスロットマシン（N 本腕バンディットマシン）をプレイする．スロットマシンの各腕にはそれぞれ当選確率が設定されており，当選すると報酬が支払われる（図 3.9）．N 本腕バンディット問題における学習者の目的は，直面する N本腕のバンディットマシンに対して，期待値として最も高い報酬を得ることができる腕を学習し，可能な限り多くの報酬を得ることである．

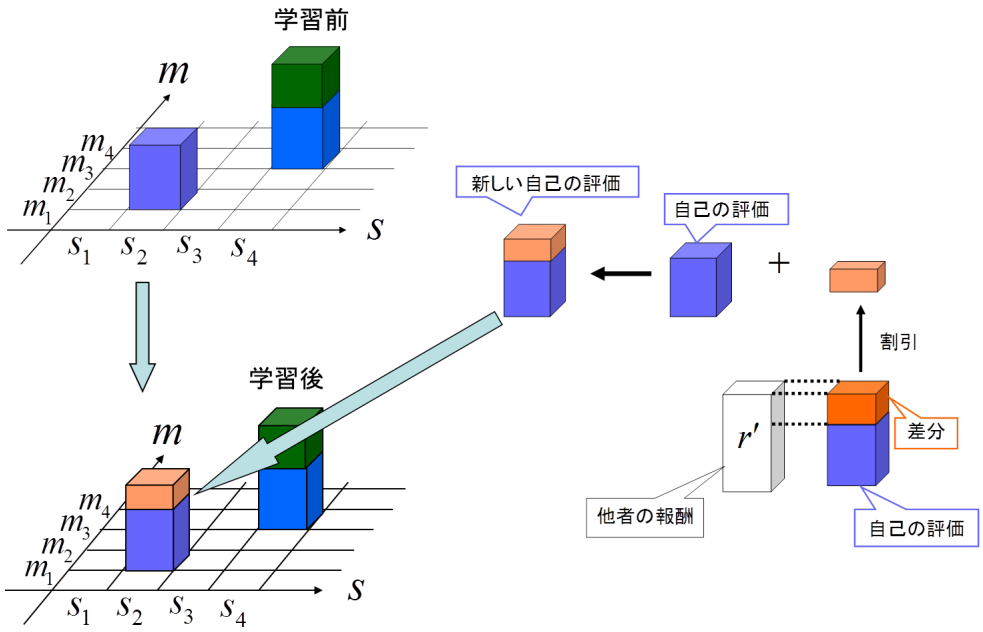


図 3.8: 学習法学習部の評価

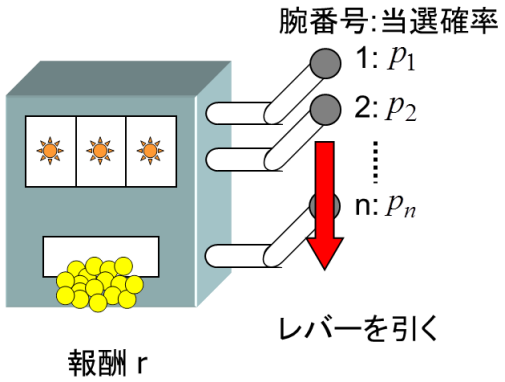


図 3.9: N 本腕バンディットマシン

一般的にいえば，N 種類の異なる行動選択肢があり，その中から学習者は 1 つの行動を選択するという動作を繰り返す．各行動選択の後に選んだ行動に依存する定常確率分布から報酬値が決定され，学習者に渡される．学習者の目的はある期間内の総獲得報酬値を最大化することである．このような問題を N 本腕バンディット問題と呼ぶ．

3.6.2 N 本腕バンディットを対象とした実験概要

本実験では，N本腕バンディット問題に対して，強化学習を適用したコミュニケーションを用いた学習システム（以降，提案システム）の有効性を検証する．本実験はコンピュータシミュレーションで行う．そのため，シミュレーション上で行動する仮想的なロボットを定義し，そのロボットをエージェントと呼称する．実験環境として，各バンディットの腕の当たり確率が試行の度に変化するような非定常環境を考える．そのような環境に対してバンディットマシンは，自身の試行錯誤による学習とコミュニケーションによる他者情報を使い，直面する環境に合った学習法と直面する状況に合った腕の選択を学習する．

実験目的

それぞれのエージェントが直面している環境について適した学習法を選択していることを確認する．

実験環境

実環境では自身が他のロボットと同じ環境にあるということはないため，各エージェントはそれぞれ異なる環境に直面することを考える．エージェントはそれぞれの直面する環境に合った学習法と状況に合った行動（最も当たり確率の高い腕）を学習する．そのために，本実験ではバンディットマシンの腕の当選確率を試行毎に変動するような実験環境を考える．変動の仕方（変動のしやすさ，変動の大きさ）はバンディットマシンによってそれぞれ異なるようにする．こうすることによってそれぞれ異なった環境を構築する．

バンディットマシンの腕の当たり確率

腕の当選確率の変動の仕方は，変動頻度（Th）と変動振幅（Amp）の 2 つの指標によって決定する．変動頻度は試行毎に当選確率の変動が起きるかどうかを確率的に決定するための変数で，$0 \leq Th \leq 1$ の範囲の値をとる．変動頻度が高いと試行毎に確率変動が起きる場合が多くなり，低いと少なくなる．変動振幅は確率変動が起きた時の確率の変化量である．変動振幅が大きいと確率の変化量も大きくなり，確率変化の安定度は下がる．

変動頻度および変動振幅の値の大小による環境の特徴について説明する．説明する環境は以下の 4 種類である．

- Th：小，Amp：小（図 3.10）

  この環境では，変動頻度が少ないため当選確率はあまり変わらない．また，変動振幅が小さいため，試行開始時に設定した当選確率から大きく変化することがない．基本的に変化の少ない静的な環境といえる．

- Th：小，Amp：大（図 3.11）

  この環境では，変動頻度が少ないため当選確率はあまり変わらない．しかし，変動振幅が大きいため，当選確率の変動幅が広く，当選確率が大きく変動することがある．

- Th：大，Amp：小（図 3.12）

  この環境では，変動頻度が多いため当選確率が頻繁に変化する．また，変動振幅が小さいことから，変化の度合いは小さく，試行開始時に設定した当選確率から大きく変化することがない．

- Th：大，Amp：大（図 3.13）

  この環境では，変動頻度の値が多いため当選確率が頻繁に変化する．また，変動振幅も大きいことから当選確率の変動幅も広い．したがって，試行するたびに最も高い当選確率を持つ腕が変わるという予測不可能な環境になる．

また，当選確率の変動値は変動振幅に従って式 3.1 で決定される．

$$p_i(n) = p_i(n-1) + RAND \times Amp \times 2 - Amp \tag{3.1}$$

ここで，$p_i(n)$ は n 回試行目でのバンディットの腕 i の当たり確率，RAND はランダムで決定した実数で $0 \leq Th \leq 1$ の範囲となる．また，Amp は変動振幅で $0 \leq Amp \leq 1$ の範囲をとる．例えば， $\mathrm{Amp} = 0.5$ であれば，-0.5〜0.5 の範囲で腕の当選確率が変動する．変動振幅の範囲内でランダムに決定された値を前回の試行での腕の当たり確率に加えることで，新たな腕の当たり確率とする．

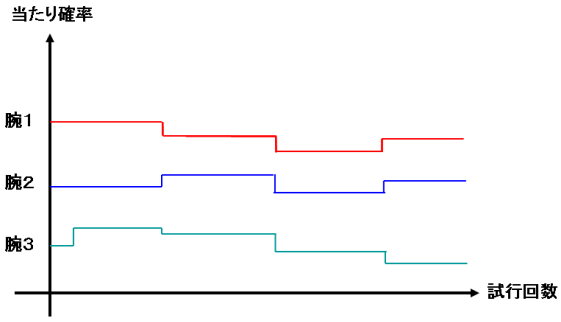


図 3.10: 変動頻度：小　変動振幅：小

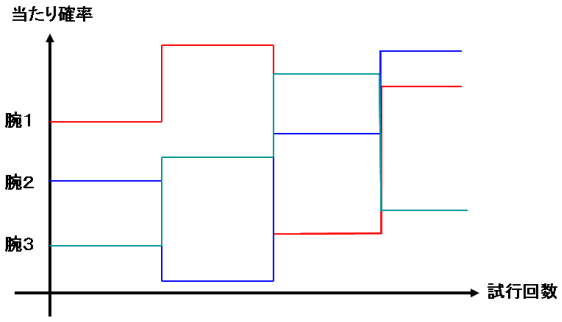


図 3.11: 変動頻度：小　変動振幅：大

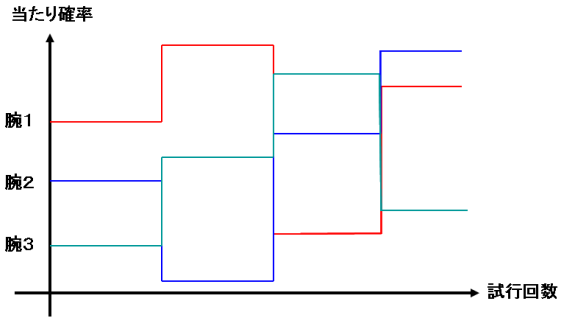


図 3.12: 変動頻度：大　変動振幅：小

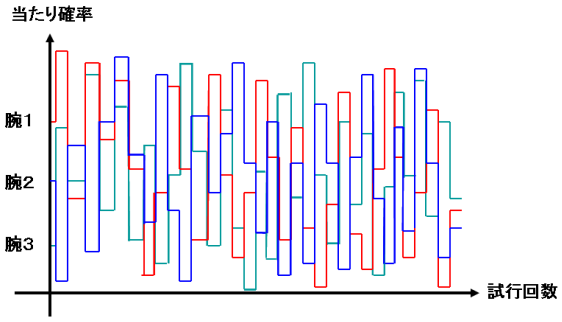


図 3.13: 変動頻度：大　変動振幅：大

エージェントが直面する環境

今回，エージェントが直面する環境は，エージェント毎にわずかに異なる環境を構築する．それは，実環境では他エージェントと全く同じ環境となるのは稀なことであるためである．実環境を考えると，各エージェントはわずかに異なる環境下でタスクを行うのが適当である．

そのような条件を満たすような実験環境として，変動振幅・変動頻度を用いたマップを構築する（図 3.14）．このマップのそれぞれのマスの変動頻度・変動振幅を設定したバンディットマシンを配置する．配置された各バンディットマシンに 1 体のエージェントを配置し，バンディットマシンをプレイする（図 3.15）．

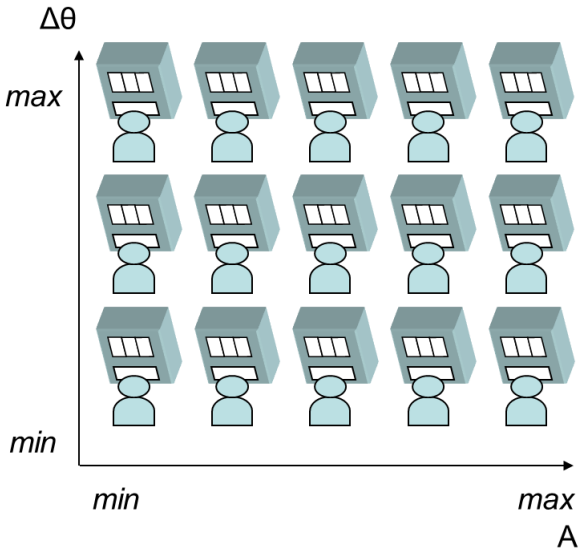
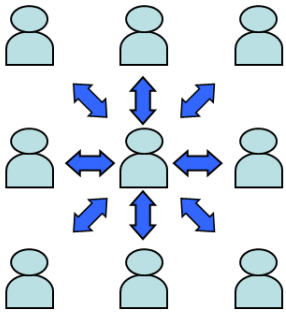


図 3.14: バンディットマシン環境マップ



コミュニケーション相手:8近傍

図 3.15: バンディットマシンとエージェントの対応

3.6.3 実験設定

タスク環境に関する設定

腕の確率変動を決定する変動頻度（Th）と変動振幅（Amp）を設定する環境マップのイメージを図 3.16 に示す．変動頻度と変動振幅を軸とし，Th，Amp は 0 から 1 までマス毎に 0.01 刻みで設定する．つまり，Th について 100 パターン，Amp について 100 パターンあり，合計 10000 パターンの異なるバンディットマシンが存在する．

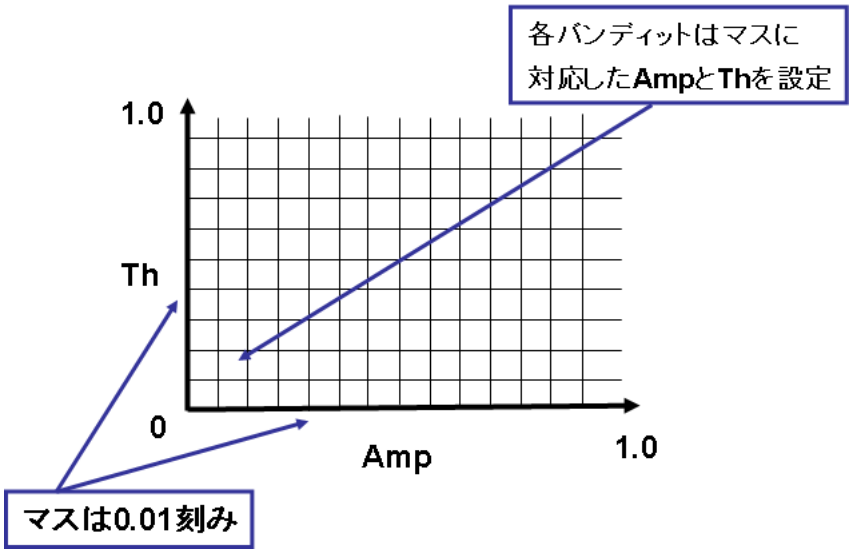


図 3.16: マスの刻み幅

バンディットマシンに関する設定

実験に使用するバンディットマシンの台数，バンディットマシンの腕本数，また支払われる報酬については表 3.1 に示す．本バンディット問題は，すべてのバンディットの腕の本数および，個々の腕に割り当てる報酬は共通とする．バンディット毎に異なるのは，確率の変動頻度と変動振幅のみである．また，各腕の初期確率について表 3.2 に示す．これは，全てのバンディットマシン共通の設定である．

表 3.1: バンディットマシンの設定

バンディットの台数	10000 台
腕の本数	6 本
報酬	1

表 3.2: バンディットマシンの各腕の初期確率

腕番号	1	2	3	4	5	6
当たり確率	0.8	0.61	0.33	0.1	0.01	0.56

エージェントに関する設定

エージェントの数，総試行回数，コミュニケーションに関する設定を表 3.3 に示す．エージェント数は本実験で用いるエージェントの数である．バンディットマシンと 1 対 1 対応させる．総試行回数は，エージェント 1 体当たり何回試行するかという設定である．コミュニケーション頻度は何回試行毎に行うかの設定である．コミュニケーションする情報は相手に送る・相手から受け取る情報の内容である．本実験では，コミュニケーション対象は周囲 8 近傍に存在するエージェントとする．また，環境の端にいるエージェントに関しては，8 近傍のエージェントとコミュニケーションすることはない．例えば環境マップの角のエージェントは 4 体のエージェントとしかコミュニケーションを行わない．

また，エージェントの 1 試行はバンディットのレバーを 1 回引くことに相当する．今回の総試行回数は 30000 回なので，エージェントは 30000 回バンディットのレバーを引くことになる．

表 3.3: エージェントに関する設定

エージェント数	10000 体
総試行回数	30000 回
コミュニケーション頻度	自己の周囲 8 マスに存在するエージェント
コミュニケーション頻度	1 行動毎
コミュニケーション情報	コミュニケーション時に自己が適用していた学習法と得られた報酬

また，学習法学習部で学習し，行動学習部で用いる学習法について，表 3.4 に示す．行動学習手法として softmax 法， ε -greedy 法，追跡手法の 3 種類を採用した．行動評価手法として，標本平均手法，加重平均手法，Q 学習の 3 種類を採用した．学習法は基本的に行動選択手法と行動評価手法を対として用いる．そのため，今回は行動選択手法・行動評価手法の組み合わせは $3 \times 3 = 9$ 種類の学習法が存在する．これらに強化比較法を加えた全 10 種類の学習法を学習法学習部で用いる．エージェントはこれら 10 種類の学習法から自身が直面する環境に合った学習法を学習する．

各学習法で用いるパラメータを表 3.5 に示す．

学習法学習部にて最適な学習法を学習するための手法としては， ε -greedy 法と加重平均手法を用いる．加重平均手法は他のエージェントからの情報を自身の学習空間に反映する際にも用いる．ただし，自己の選択した学習法に対する評価式と他のエージェントからの情報に対する評価式が異なる（式 (3.2)，式 (3.3)）．自身の選択した学習法に対する評価式を式 (3.2) に示す．m は自身が選択した学習法，r_{n+1} が n+1 試行目で得た報酬，α がステップサイズ・パラメータである．また，他のエージェントからの情報を自身の学習空間へ反映させ

表 3.4: 学習法学習部で用いる手法

手法番号	行動選択手法	行動評価手法
0	softmax 法	標本平均手法
1	softmax 法	加重平均手法
2	softmax 法	Q 学習
3	ε-greedy 法	標本平均手法
4	ε-greedy 法	加重平均手法
5	ε-greedy 法	Q 学習
6	追跡手法	標本平均手法
7	追跡手法	加重平均手法
8	追跡手法	Q 学習
9	強化比較法	強化比較法

表 3.5: 各学習法の学習パラメータ

パラメータ名	数位値
softmax 法 τ	0.1
ε -greedy 法 ε	0.1
追跡手法 β	0.1
強化比較法 α	0.1
強化比較法 β	0.1
強化比較法 リファレンス報酬の初期値	1
加重平均手法 α	0.08
Q 学習 α	0.05
Q 学習 β	0.01

る式は式 (3.3) で定義する．m' は他のエージェントが選択した学習法，r'_{n+1} は n+1 試行目で他のエージェントが得た報酬，γ はステップサイズ・パラメータである．

式 (3.2)，式 (3.3) で異なっているのは，ステップサイズ・パラメータ α，γ である．α の大きさで，自身の学習を重視するかどうか，γ の大きさで他者の情報を重視するかどうかが決まる．両者とも値が大きいほど，影響が大きくなる．

$$Q^{mth}_{n+1}(m) \leftarrow Q^{mth}_{n}(m) + \alpha[r_{n+1} - Q^{mth}_{n}(m)] \tag{3.2}$$

$$Q^{mth}_{n+1}(m') \leftarrow Q^{mth}_{n}(m') + \gamma[r'_{n+1} - Q^{mth}_{n}(m')] \tag{3.3}$$

学習法学習部の学習パラメータに関する設定を表 3.6 に示す．

3.6.4 実験結果・考察

手法の有効性の評価を，環境マップ上での選択学習法の推移および，獲得報酬においてコミュニケーションを用いないエージェントとの比較することで行う．環境マップ上で選択学習法の推移を観察することで，エージェントが環境に適した学習法を選択していることを確認する．また，コミュニケーションなしのエージェントと獲得報酬の比較を行うことで，提案手法がより効率的に学習を行うことが可能であることを確認する．

表 3.6: 学習法学習部の学習パラメータ

パラメータ名	数位値
ε -greedy 法 ε	0.1
加重平均手法：自己の選択した学習法に対する評価 α	0.08
加重平均手法：他のエージェントからの情報に対する評価 γ	0.01

学習法選択の推移について

実験結果を図 3.17〜図 3.25 に示す．これらの図は，それぞれ 10 回・500 回・1000 回・2000 回・5000 回・10000 回・15000 回・20000 回・30000 回試行目で提案手法を適用したエージェントが選択のした学習法を環境マップ上にカラーグラフでプロットしたものである．なお，図中のカラーバーの番号は，表 3.4 の手法番号に対応している．

- 試行 0 回目の学習手法の分布

  試行 0 回目の学習手法の分布を図 3.17 に示す．この段階では，エージェントは未学習の状態であるため，すべてのエージェントはランダムに行動を選択する．全体的に色番号 7 番，8 番が多く見えるのは，greedy な行動をとった際にプログラム上で一番上のインデックスである 9 番をまず最初に選んでいることがまず挙げられる．これにより，環境マップ全体の傾向が黄色寄りの色傾向になる．また，このグラフでは，マスの刻み幅が 0.01 なので，グラフ軸の値 0.1 の間に 100 体のエージェントがいることになる．このグラフでは，これらのエージェントの選択した行動を色で細かく表示しきれていない．そのため，手法 9 番以外の手法を選択したエージェントがいた場合，周囲のエージェントと色情報が混ざってプロットされてしまう．よって，全体的に 7 番 8 番の手法が選ばれているように見える．実際は，今回 ε -greedy 法の ε の値は 0.1 なので，約 9 割のエージェントは手法 9 を選んでいる．

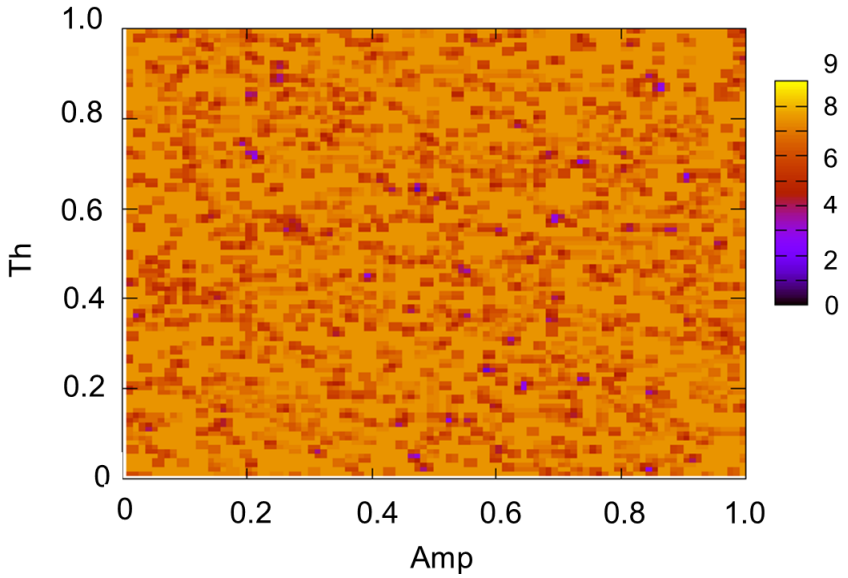


図 3.17: 試行 0 回目の学習手法の分布

- 試行 500 回目・1000 回目・2000 回目の学習手法の分布

試行 500 回目・1000 回目・2000 回目の学習手法の分布を図 3.18・図 3.19・図 3.20 に示す．この段階では，エージェントは様々な学習手法を試し自身の環境に適した学習を行なっている最中である．そのため，エージェントが選択する学習手法は常に変化している．

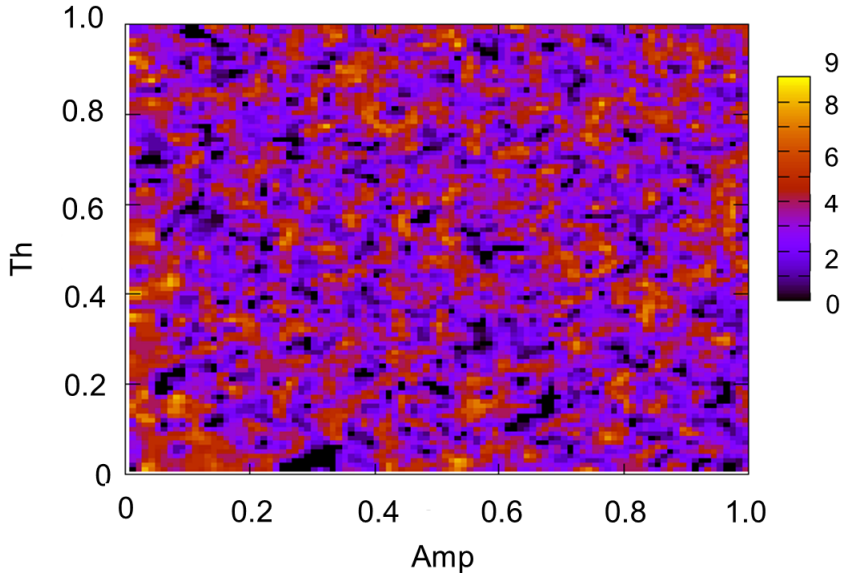


図 3.18: 試行 500 回目の学習手法の分布

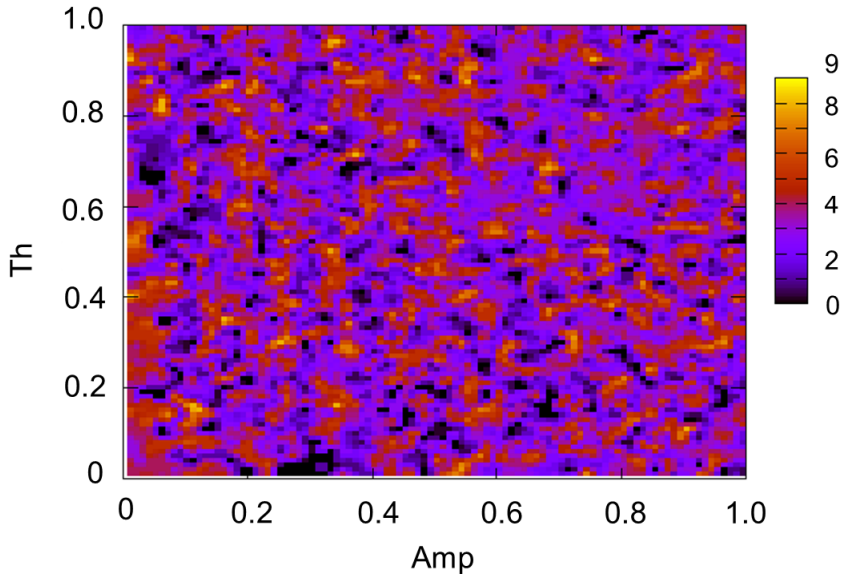


図 3.19: 試行 1000 回目の学習手法の分布

- 試行 5000 回目・試行 10000 回目の学習手法の分布

  試行 5000 回目・試行 10000 回目の学習手法の分布を図 3.21・図 3.22 に示す．この段階から，一部のエージェントは学習が完了し始める．図の左下を中心に L 字型のエリアは手法番号 1〜3 が選択されている．また $0.2 \leq Th \leq 1$ かつ $0.2 \leq Amp \leq 0.3$ のエリアと $0 \leq Th \leq 0.4$ かつ $0.3 \leq Amp$ 1 のエリアでは，手法 6〜8 が選択されている．残りのエリアは手法番号 2・3 と 4〜6 が混在している．

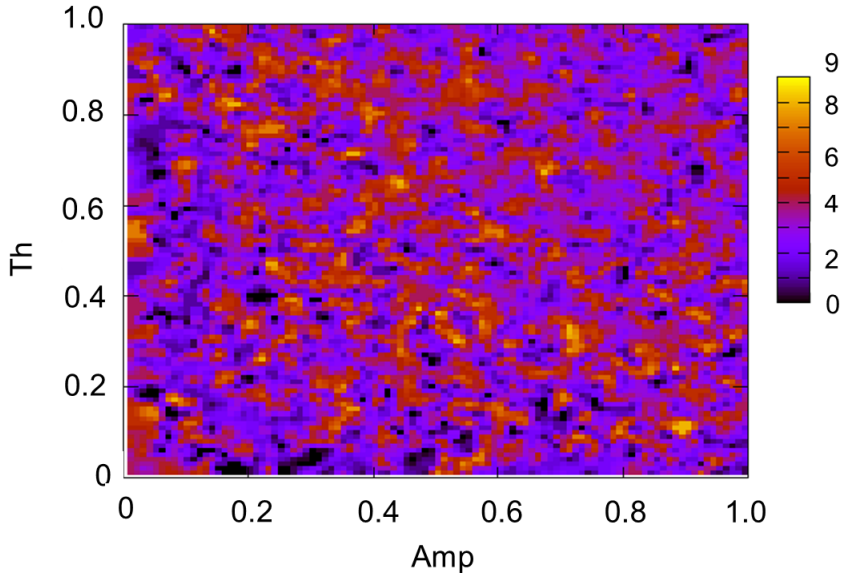


図 3.20: 試行 2000 回目の学習手法の分布

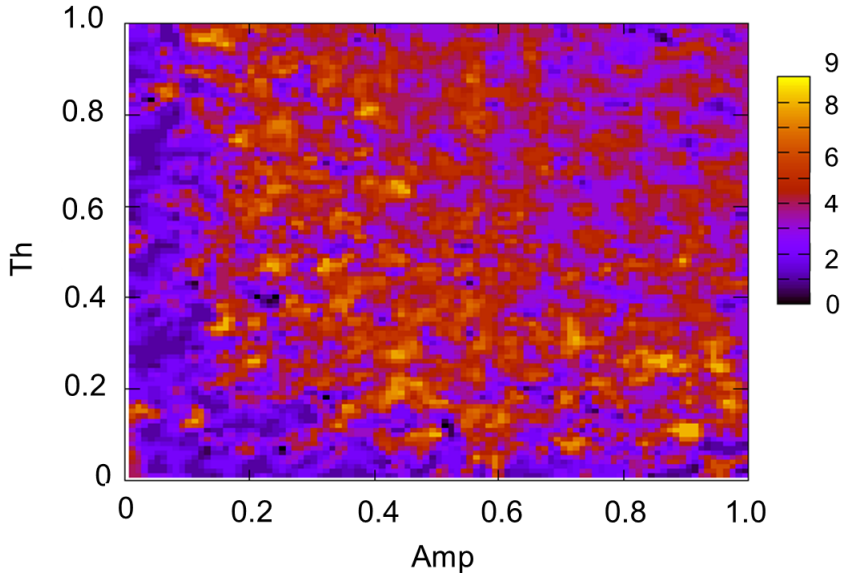


図 3.21: 試行 5000 回目の学習手法の分布

- 試行 15000 回目・試行 20000 回目・試行 30000 回目の学習手法の分布

  試行 15000 回目・試行 20000 回目・試行 30000 回目の学習手法の分布を図 3.23〜図 3.25 に示す．この段階では，試行 5000 回時点の状態から選択されている学習手法が大きく変化することはない．しかしながら，学習が進むにつれて学習手法が選択されているエリアのエッジがはっきりとしてくる．

- 試行 30000 回目での選択手法の領域

  図 3.26 に試行 30000 回時点での各エージェントが学習法の分布の領域分析を示す．最終的には 3 つのエリアに分かれる．それぞれの領域について分析していく．

・図中の黄緑色の線で囲まれた領域（左の領域）について
図中の緑色に囲まれた領域では，softmax 法＋加重平均手法，softmax 法＋ Q 学習法が選

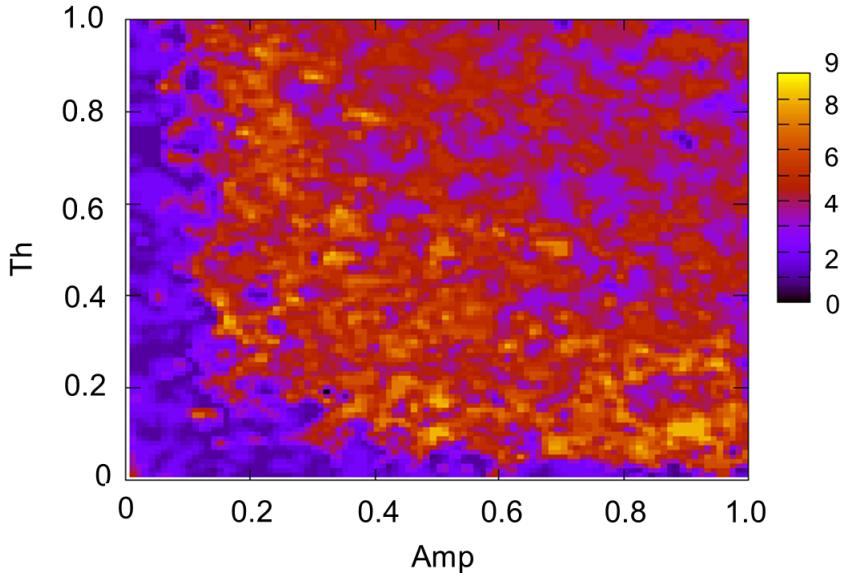


図 3.22: 試行 10000 回目の学習手法の分布

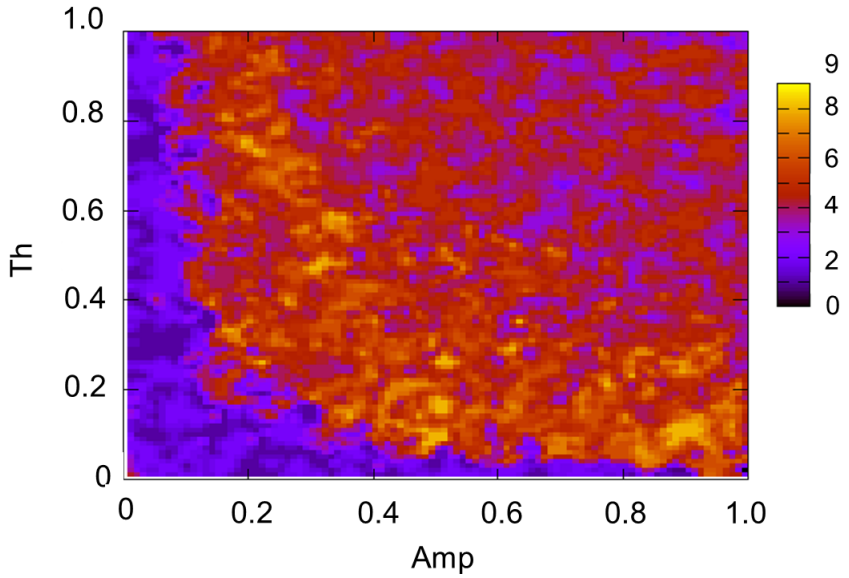


図 3.23: 試行 15000 回目の学習手法の分布

択されている．この領域では，主に 3 つの環境変化のパターンが存在する．1 つ目の領域は，$Th \leq 0.1$ でかつ $0.2 \leq Amp \leq 0.9$ の変化パターンである．この領域では，滅多に環境が変化することはない．そのため，変化量の大小に関わらず，環境に追従することができる加重平均手法や Q 学習法が有効となる．2 つ目の領域は，$0.2 \leq Th \leq 1$ かつ $Amp \leq 0.1$ の変化パターンの領域である．この領域では，高確率で腕の当たり確率が変化する．しかし，Amp が低いためそう大きく変化はせず，変化した腕の Q 値の変化量は大きくない．そのため，加重平均手法や Q 学習法で数回 Q 値の更新を行うだけで環境変化に追従することができる．3 つ目の領域は，$Th \leq 0.1$ かつ $Amp \leq 0.1$ の変化パターンである．この領域は，ほとんど腕の当たり確率が変化せず，変化しても微量である．しかしながら，静的な環境ではなく変化が起きればエージェントはそれに追従する必要がある．そのため，環境の変化に追従しやすい加重平均手法や Q 学習が選択されている．

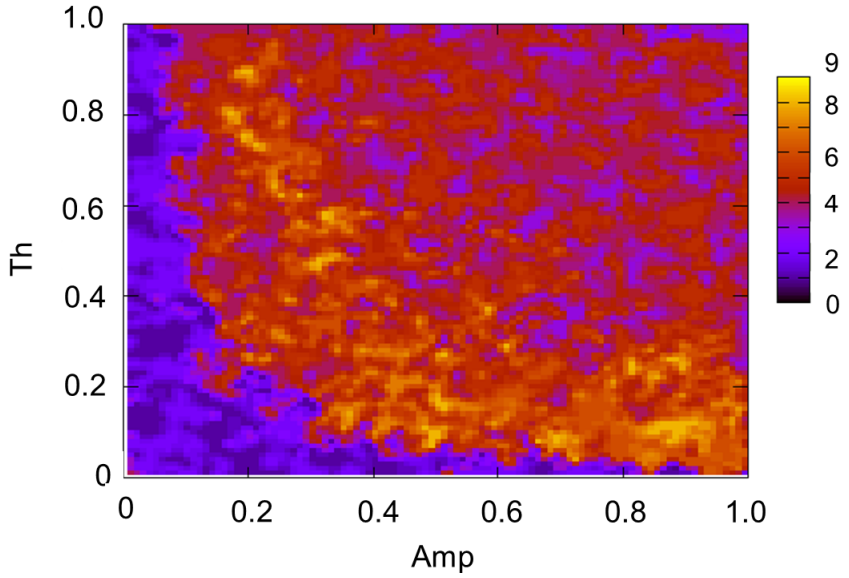


図 3.24: 試行 20000 回目の学習手法の分布

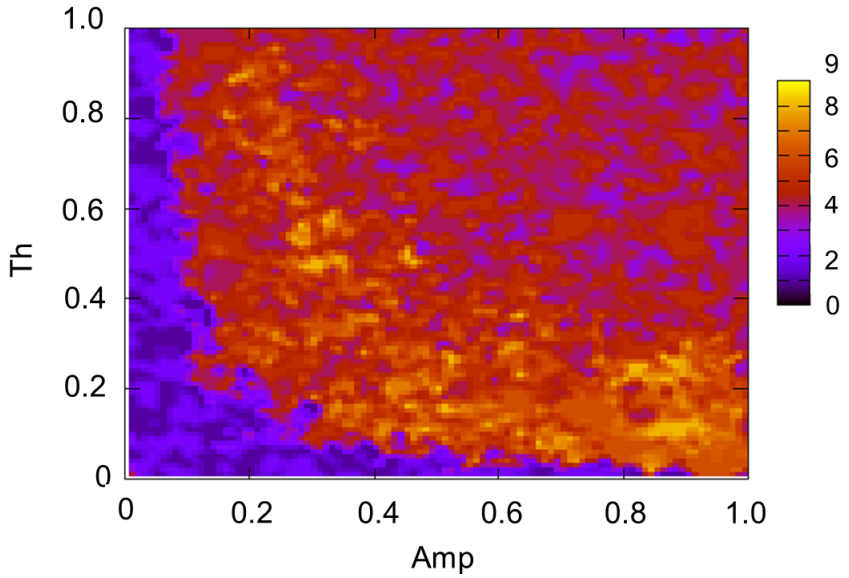


図 3.25: 試行 30000 回目の学習手法の分布

また，全体的に softmax 法が選択されている．この理由としては，大きな変化が起こりにくいことにある．よって，探索を重視せず greedy な行動選択手法が有効に作用する．今回の softmax 法の τ の値では，僅かな Q 値の変化で腕の選択確率が大きく変化する．これは，今回採用した行動選択手法の中で最も greedy な挙動をとることを意味する．そのため，このエリアでは softmax 法が選択されている．

・図中の水色の線で囲まれた領域（中央の領域）についてこの領域では， ε -greedy 法＋加重平均法， ε -greedy ＋ Q 学習，追跡手法＋加重平均手法，追跡手法＋ Q 学習で構成されている．腕の当選確率の変動が多く，変動量があまり多くない環境（$0.4 \leq Th \leq 1$ かつ $0.2 \leq Amp \leq 0.4$）と変動があまり多くなく，変動量が大きい領域（$0 \leq Th \leq 0.4$ かつ $0.4 \leq Amp \leq 1$）の環境でこの領域は構成されている．このような領域では，腕の当選確率

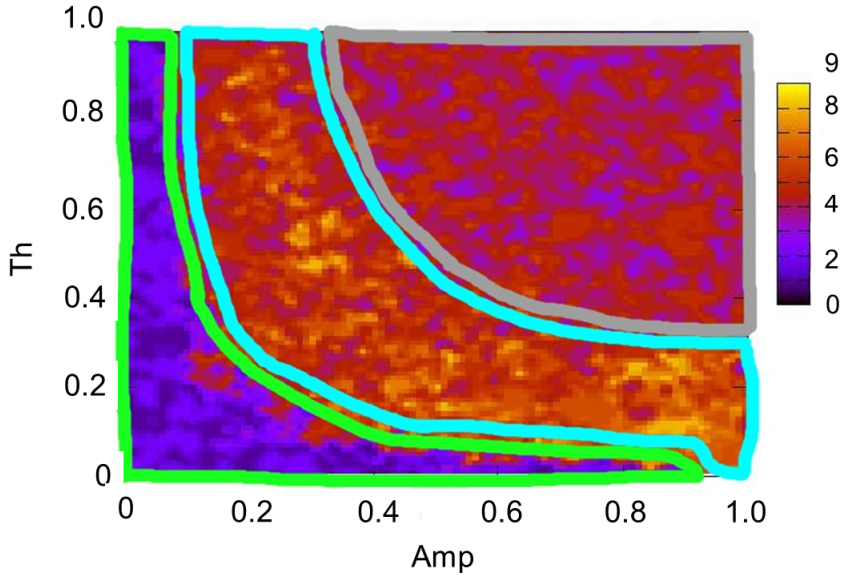


図 3.26: 試行 30000 回目の手法の分布に関しての領域分析

の変化の仕方が若干緩やかである．そのため，ある程度確率的な手法が適している．この領域では，追跡手法と ε ‐ greedy 法が選ばれている．追跡手法はある程度確率的に動くが，環境変化への追従性はあまり高くない．そのため，変動頻度の大きい領域（$0.4 \leq Th \leq 1$ かつ $0.2 \leq Amp \leq 0.4$）ではあまり選択されていないが，変動頻度の小さく変動量の大きい領域（$0 \leq Th \leq 0.4$ かつ $0.4 \leq Amp \leq 1$）では選択されていることが多い．また，本実験では行動学習部で用いられる ε -greedy 法の ε の値は 0.1 であるため，10 試行に 1 回はランダムで腕を選択する．そのため，追跡手法に比べて多く探索的な行動をとることができた．変動頻度の大きい領域では最適な腕を探索し直すことが多く発生するため，探索行動を高い確率で行う手法が適当である．今回このような領域では ε -greedy 法が選ばれており，このことからエージェントが環境に適した手法を選択できているといえる．また行動評価手法であるが， ε -greedy 法，追跡手法を用いている領域では，腕の確率が変化してもそれにあわせて Q 値を適切にアップデートすることが可能な加重平均手法や Q 学習が用いられている．

・図中の緑色の線で囲まれた領域（右の領域）この領域では， ε -greedy 法を用いたすべての手法が選択されている．この領域では，腕の当選確率が変動しやすく，変動量も多い．よって，当選確率の高い腕は頻繁に変わる．このような領域では，greedy に行動しつつもある程度ランダムな挙動をする手法が適していると考える．これは，頻繁に当選確率の高い腕が変わるため，現在選択している腕の当選確率が下がっても，またすぐに腕の当選確率が高くなる場合多いためである．また，頻繁に当選確率の高い腕が変わることから，ランダムな挙動をした方がよい場合がある．本実験では，行動学習部の ε ‐ greedy 法の ε の値は 0.1 なので，10 回試行に 1 回はランダムに腕を選択する．これは，ある程度ランダムな挙動をする手法といえる．したがって，この領域では ε ‐ greedy 法が選択されていると考える．変動頻度と変動振幅の大きい領域では，行動評価手法はどの手法を用いてもあまり違いがない．これは，手法の追従性の問題である．あまりに変化の激しい環境の場合，手法が追従しようとしてもしきれないということが起こる．そのため，どのような手法を用いても環境に追従しきれないため，行動選択手法のみに影響がでている．

獲得報酬の比較

次に各手法固定の場合およびコミュニケーションを用いない場合と提案手法との比較を行う．比較は手法の学習が収束した時点の平均獲得報酬の比較を行う．図 3.25 より，試行数が 15000 回以降選択手法の分布にほとんど変化が見られなくなったため，試行 30000 回目では学習が安定したといえる．そのため，試行 29000〜30000 回の 1000 回の獲得平均報酬の比較は妥当だと考える．よってここからは，この区間での獲得平均報酬から提案システムの有効性を考察する．比較は提案手法エージェントの平均獲得と各手法固定エージェントおよびコミュニケーションなしエージェントの平均獲得の差をグラフ化する．

- コミュニケーションなしとの比較

  提案手法とコミュニケーションなしの場合を図 3.27，図 3.28 に示す．図 3.28 は図 3.27 の z 軸のスケールを変え拡大したものである．コミュニケーションなしの場合とは，コミュニケーションをしないで，自身の経験のみで学習を行った場合である．そのため，自律的に学習法の学習を行なっている．これらの図より，提案手法がコミュニケーションなしの場合に比べて 0.03 程度獲得報酬量が多くなった．従って，個体単体よりもコミュニケーションを行った方が学習が効率的に行うことが可能であることが確認された．

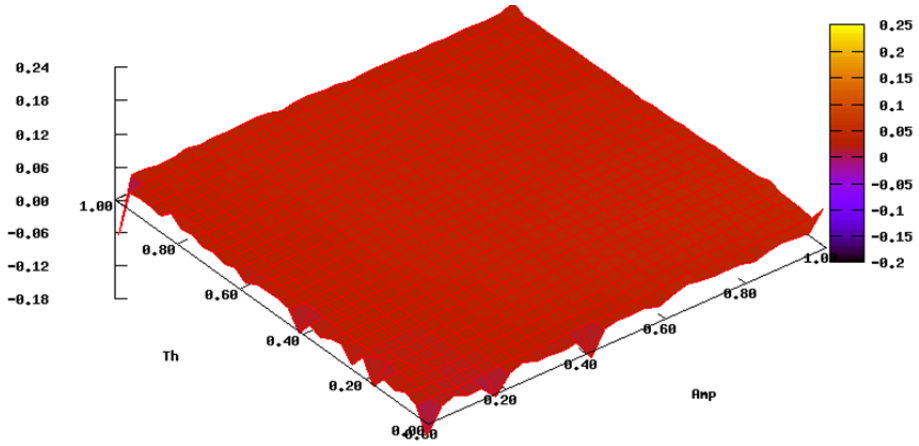


図 3.27: 提案システムとコミュニケーションなしの場合との平均獲得報酬の差

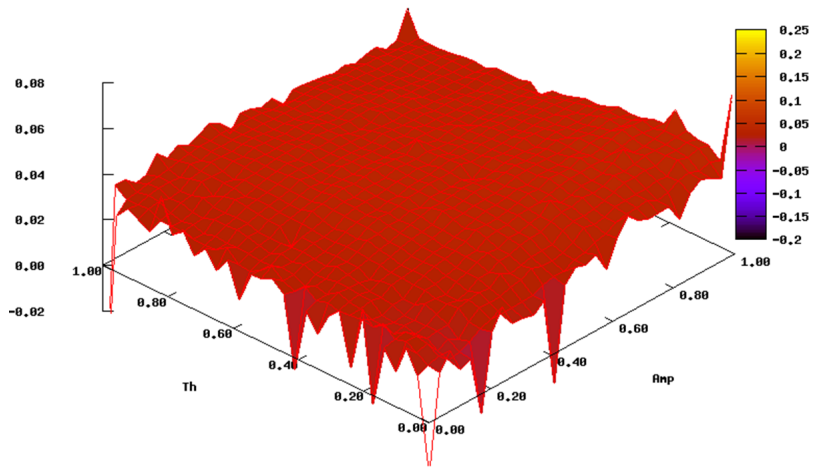


図 3.28: 提案システムとコミュニケーションなしの場合との平均獲得報酬の差（拡大）

- softmax 法を用いた手法での比較

  提案手法と softmax 法＋標本平均手法との比較を図 3.29 に示す．全体的に 0.15〜0.20 ほど獲得報酬報酬は高くなっている．

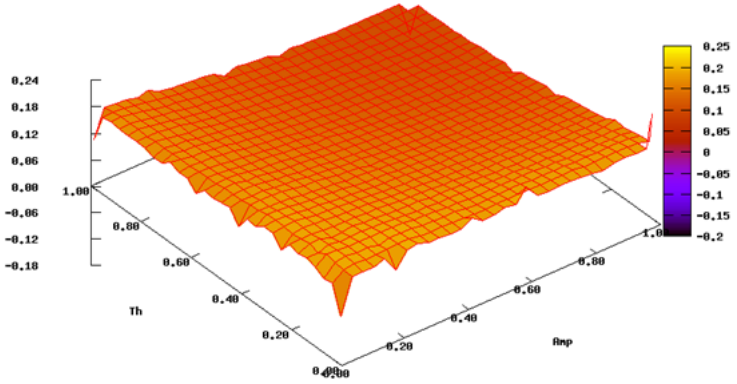


図 3.29: 提案システムと softmax 法＋標本平均手法との平均獲得報酬の差

提案手法と softmax 法＋加重平均手法との比較を図 3.30・図 3.31 に示す．また，提案手法と softmax 法＋ Q 学習との比較を図 3.32・図 3.33 に示す．低 Th 低 Amp のエリアを覗いて全体的に 0.01 ほど softmax 法＋加重平均手法，softmax 法＋ Q 学習に対して提案手法が優位な結果となっている．しかし，低 Th 低 Amp のエリアでは，提案手法は低い報酬値となっている．このエリアでは，図 3.26 に示す通り，softmax 法＋加重平均手法，及び softmax 法＋ Q 学習が最適な学習手法である．対して，提案手法では，一定の割合でランダムに学習法を選択する．それにより，不適切な学習法を選択する可能性が高い．そのため，始めから適した手法である softmax 法＋加重平均手法及び softmax 法＋ Q 学習を行うエージェントの方が高い報酬を獲得することができたと考えられる．

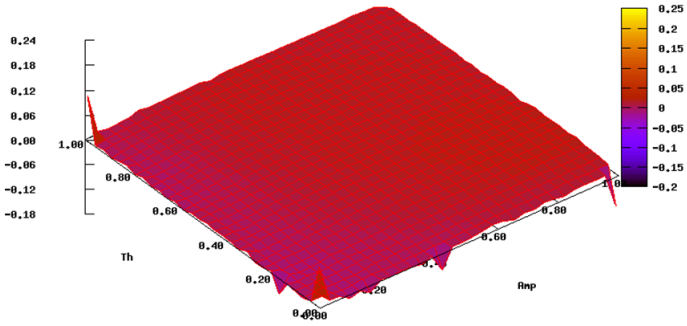


図 3.30: 提案システムと softmax 法＋加重平均手法手法との平均獲得報酬の差

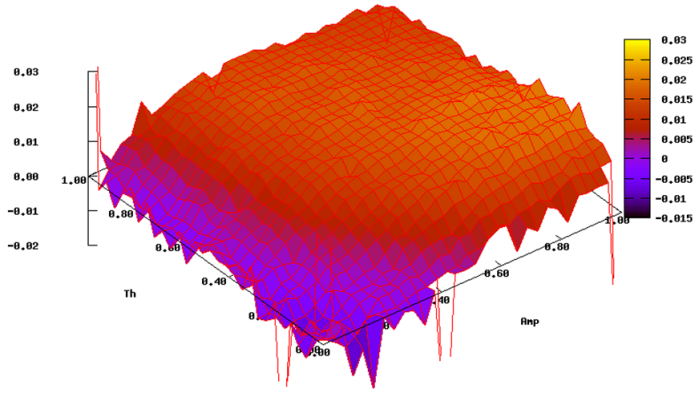


図 3.31: 提案システムと softmax 法＋加重平均手法の平均獲得報酬の差（拡大）

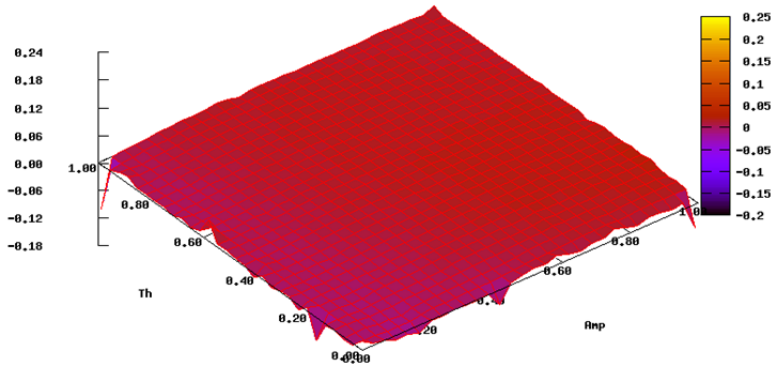


図 3.32: 提案システムと softmax 法＋ Q 学習との平均獲得報酬の差

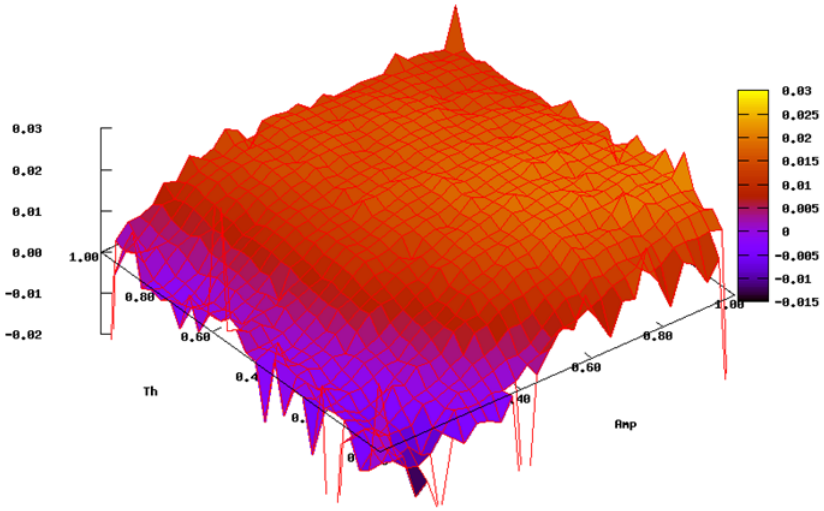


図 3.33: 提案システムと softmax 法＋ Q 学習の平均獲得報酬の差（拡大）

- ε-greedy 法を用いた手法との比較

  提案手法と ε-greedy 法＋標本平均手法との比較を図 3.34 に示す．全体的に 0.15〜0.20 ほど獲得報酬報酬は高くなっている．

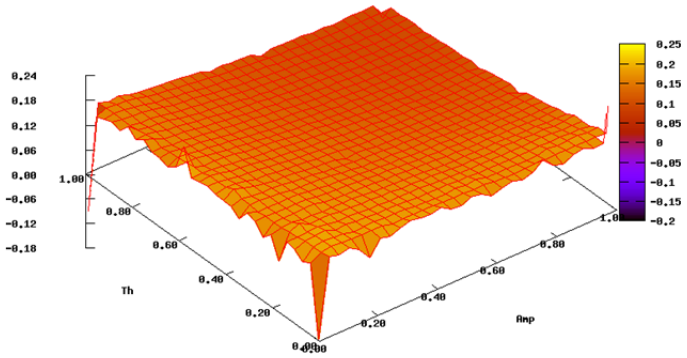


図 3.34: 提案システムと ε-greedy 法＋標本平均手法との平均獲得報酬の差

提案手法と ε-greedy 法＋加重平均手法との比較を図 3.35・図 3.36 に示す．また，提案手法と ε-greedy 法＋ Q 学習との比較を図 3.37・図 3.38 に示す．低 Th 低 Amp のエリアを覗いて全体的に 0.02 ほど ε-greedy 法＋加重平均手法，ε-greedy 法＋ Q 学習に対して提案手法が優位な結果となっている．しかし，低 Th 低 Amp のエリアでは，提案手法は低い報酬値となっている．このエリアでは，図 3.26 に示す通り，softmax 法＋加重平均手法，及び softmax 法＋ Q 学習が最適な学習手法である．しかし，ε-greedy 法＋加重平均手法および ε-greedy 法＋ Q 学習と提案手法を比較した場合，提案手法が低い獲得報酬量であった．このエリアでは，環境の変化が少ないため，greedy な行動選択手法が望ましい．今回の実験パラメータでは，softmax 手法が最適であるが，次点で ε-greedy 法が greedy な戦略を取りやすい．そして，始めから ε-greedy 法を用いた手法の方が適切な学習法を探索するよりも迅速に最適なバンディットの腕を学習することができる．そのため，このエリアでは，ε-greedy 法を用いた手法の獲得報酬量に比べて提案手法の方が低いという結果になったと考えられる．

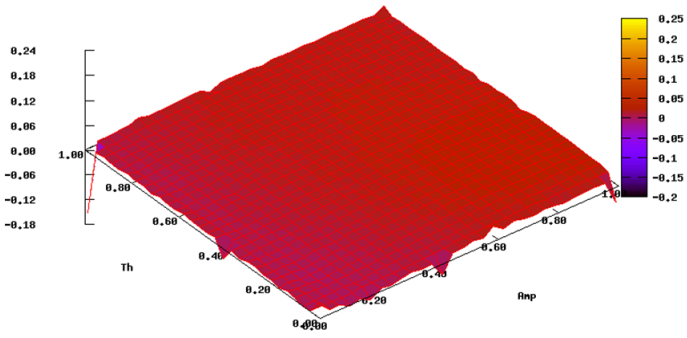


図 3.35: 提案システムと ε-greedy 法＋加重平均手法との平均獲得報酬の差

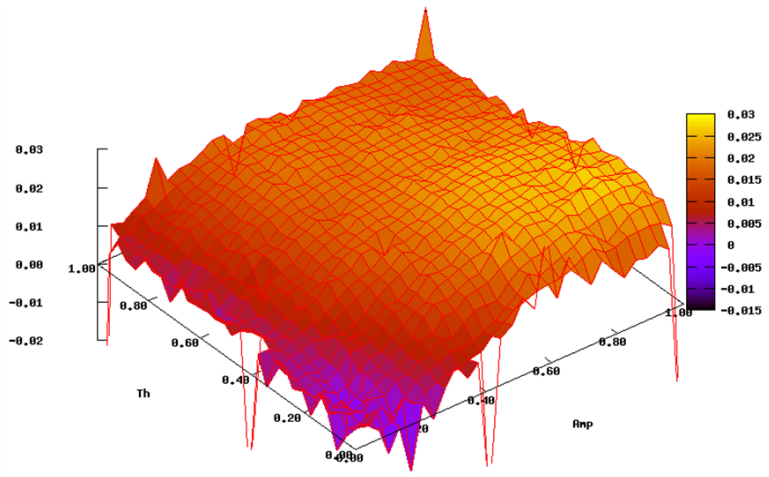


図 3.36: 提案システムと ε -greedy 法＋加重平均手法の平均獲得報酬の差（拡大）

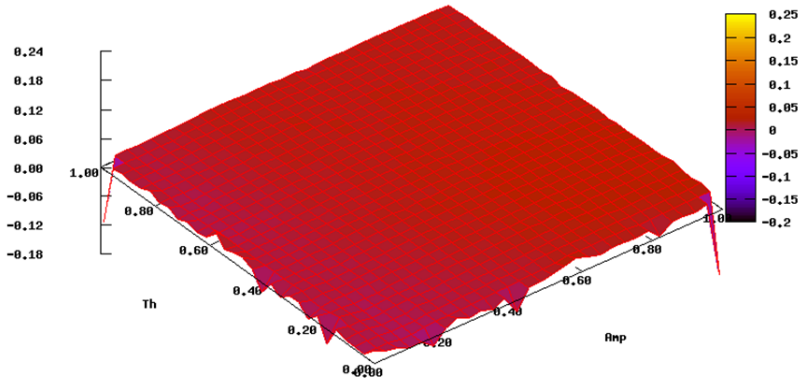


図 3.37: 提案システムと ε -greedy 法＋ Q 学習との平均獲得報酬の差

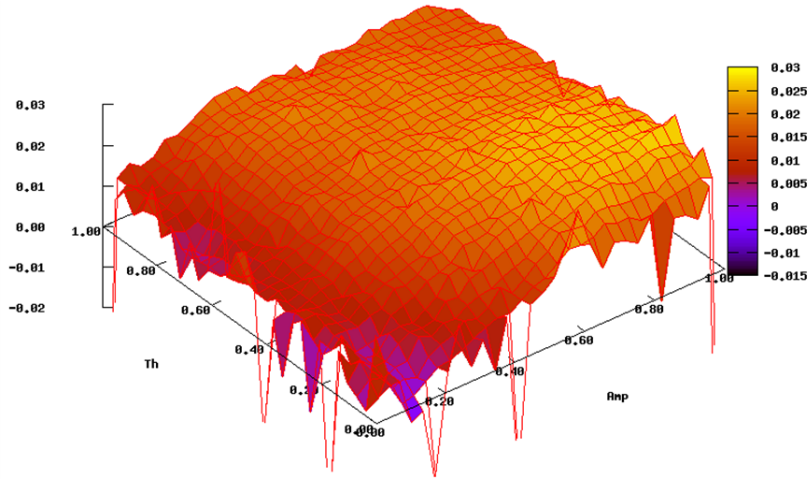
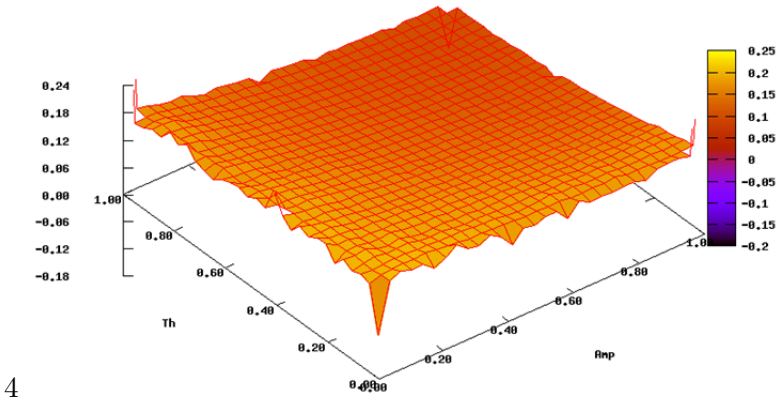


図 3.38: 提案システムと ε -greedy 法＋ Q 学習の平均獲得報酬の差（拡大）

- 追跡手法を用いた手法との比較

　提案手法と追跡手法＋標本平均手法との比較を図 3.39 に示す．全体的に 0.15〜0.20 ほど獲得報酬報酬は高くなっている．提案手法と追跡手法＋加重平均手法との比較を図 3.40・図 3.41 に示す．こちらも全体的に 0.10〜0.15 ほど高い報酬を得ている．



4

図 3.39: 提案システムと追跡手法＋標本平均手法との平均獲得報酬の差

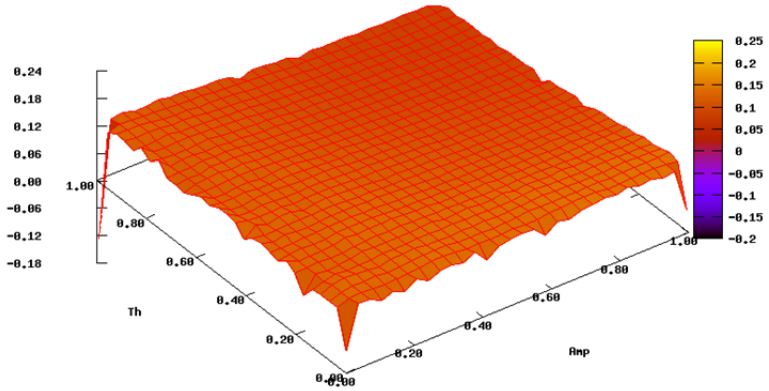


図 3.40: 提案システムと追跡手法＋加重平均手法との平均獲得報酬の差

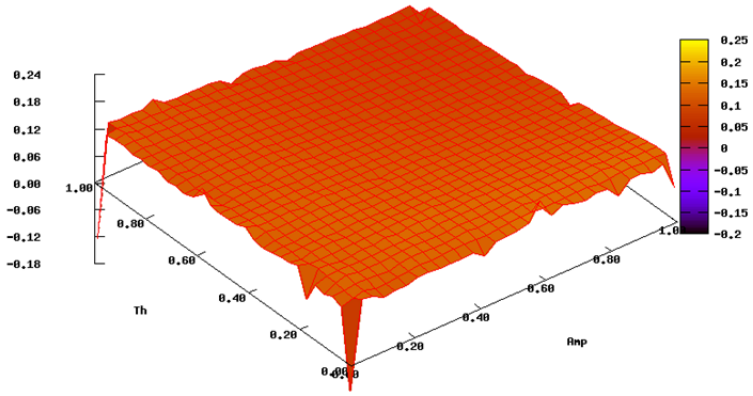


図 3.41: 提案システムと追跡手法＋ Q 学習との平均獲得報酬の差

- 強化比較法との比較

　提案手法と強化比較法の比較を図 3.42 に示す．図より提案手法の方が 0.10〜0.12 ほど高い報酬を得ている．強化比較法では，リファレンス報酬を基に腕の選択確率を決

定している．そして，リファレンス報酬は過去に得た報酬値の平均値としている．そのため，試行数が増えるほど，1 回の試行で得た報酬がリファレンス報酬に与える影響が小さくなる．その結果，環境変化に追従するのが難しくなっていくと考えられる．全体的に獲得報酬が提案手法に及ばないのは，これが原因だと考えられる．

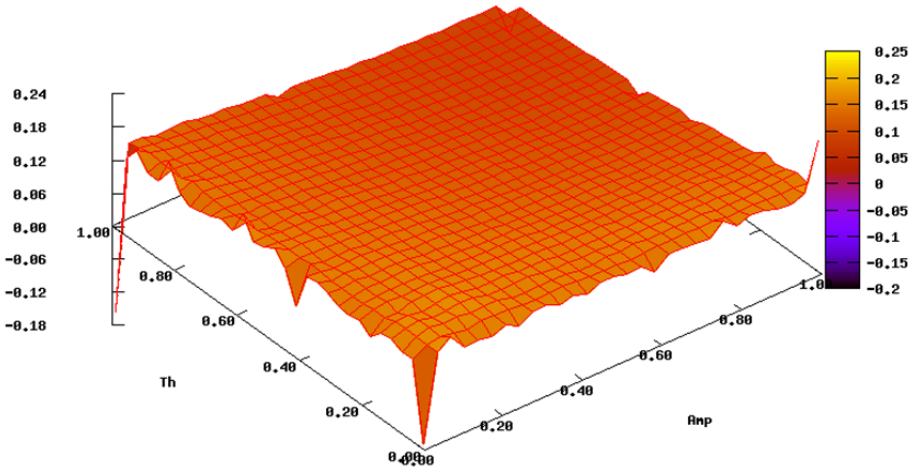


図 3.42: 提案システムと強化比較法の平均獲得報酬の差

実験結果のまとめ

本実験では，提案手法の有効性を確認した．その方法として，提案手法の選択手法の分布を確認した．学習が完了した段階で，主に 3 つの手法領域に分かれた．それぞれ，環境変化の少ない領域，環境変化が中程度の領域，環境変化が大きい領域に分かれた．選択されている学習手法もそれぞれの領域に適している学習法が選択されていることを確認した．

また，コミュニケーションなしの場合および今回採用した各学習法を固定して実験を行った場合について，提案手法と平均獲得報酬を比較した場合について確認した．結果は，コミュニケーションなしの場合よりも提案手法のほうが多くの報酬を獲得していること確認した．また，各手法との比較では，全体的に提案手法が高い報酬を獲得していることを確認した．また，低い Th および低い Amp のように環境の変化が少ない領域では，一部の学習法に劣ることが確認された．しかし，実環境では，人間が環境の変化傾向を予測して適した学習法を決定するのは難しい．そのため，提案手法のように自律的に環境に適した手法を学習することができることは有効である．

3.7 まとめ

本章は，外部情報の取捨選択を実現する前に，他のロボットとのコミュニケーションが個体の学習の促進に有効であるという仮説の検証という位置づけである．そのために，センサ情報としての他のロボットからの情報を挙げ，他のロボットとのコミュニケーションについて考察した．そして，コミュニケーションを利用した個体学習促進システムの概念ついて述べた．実システムとして，強化学習を基礎とした学習法・行動学習システムを提案した．提案したシステムの有効性の確認のために，非定常環境 N 本腕バンディット問題に適用し，提案システムの有効性および仮説が正しいことを確認した．次章からは，コミュニケーションを利用した個体学習促進システムを拡張し，コミュニケーション対象を自律的に選択するシステムについて述べる．

第4章　コミュニケーション相手の取捨選択による個体学習の促進

4.1　コミュニケーション学習の問題点

前章では，コミュニケーションによる個体学習の効率化に注目した．そのため，コミュニケーションの規則を下記のようにすべて定義した上で，個体学習の効率化について議論してきた．

- 情報のフォーマットが個体間で同一，つまり身体構造が同一であること
- 個体固有の情報を含まないものであること
- 情報は個体間で特に変換の必要なく利用可能であること
- タスクの種類・目的が個体間で共通であること

ロボットの学習に有用なコミュニケーションを実現するためには，個体間で身体構造や情報形式，タスクや目的の一致といったことを考える必要がある．しかし，人間がコミュニケーションすべき相手を決定するのは人間にとって大きな負担となる．汎用ロボットであれば，様々なタスク・目的が与えられ，動作する環境も多種多様である．また，一緒にタスクを行うロボットも自身と同じ身体構造とは限らない．そのため，ロボットが直面する状況を人間が予想し，コミュニケーションすべき相手を考えるのは難しい．また，ロボットの故障や追加によるコミュニケーション設定の再設定も人間にとっては大きな負担となる．

4.2　コミュニケーション相手の取捨選択

ロボットが直面する状況を人間が予測することが難しいため，ロボット自身が自分にとって有益な情報を持つ相手とコミュニケーションを行うことが望ましい．そこで，本章ではコミュニケーション相手の自律的選択について考える．それにより，人間によるコミュニケーション相手の設定の負担を軽減するとともに，ロボット自身がコミュニケーション相手を選択することで人間の設定よりも適したコミュニケーション相手が選択される可能性がある．

コミュニケーション相手は，前述の通り身体構造や情報形式，タスクや目的の一致といった要素を考えることが必要である．しかし，これらすべての要素について自律化を行うのは難しい．そこで，本章ではロボットの目的に着目する．本章では，各個体の目的が異なり，その他の身体構造や扱うことのできる情報の形式・タスクは各個体で共通の状況を考える．

タスクが共通で目的が異なるという状況について例を用いて説明する．例えば，荷物配送タスクを考える．各ロボットはそれぞれ与えられた目的地に荷物を配送することがタスクの達成にあたる．ここで目的が異なるというのは，目的地が異なることにあたる．

目的の違うロボット同士では，コミュニケーション情報が有効に作用しない可能性が高い．荷物配送タスクにおいて配送先へのルート情報を交換した場合，互いに全く異なる目的地へのルートに関する情報を基に行動するため，各ロボットの学習に悪影響を及ぼす可能性がある．しかし，自身の目的地と同じ目的地または近い目的地を持つロボットであればコミュニケーションは有用に働く．自身と同じ目的地を持つロボットは，コミュニケーション情報の

すべてを有効に使うことができる．また，自身と近い目的地を持つロボットとも，大部分の情報は有用であるためそのようなロボットともコミュニケーションが有効である．

4.3 従来研究との違い

このようなコミュニケーションによる個体知能の発達に関する研究は幾つか存在する．しかし，その多くは群と個体の発達そのものに着目したもの [84]-[86] が多く，コミュニケーション個体の取捨選択に注目したものは少ない．コミュニケーション個体の取捨選択の類似研究では，文献 [87][88] のようなものがある．これらの研究では，コミュニケーション相手の取捨選択の基準を熟練度とし，コミュニケーションの相手を熟練者のみにすることで，より効率的なコミュニケーション学習を行うシステムを提案している．しかし，このシステムでは，本章で注目している目的の違いによる取捨選択はできないため，様々な目的のロボットが含まれる環境には対応できない．本章で提案するシステムは目的変化に注目しているという点で従来の研究とは異なっている．

4.4 本章の目的

本章では，自身にとって有益な情報をもたらすロボットを学習する手法を提案する．それにより，コミュニケーションを用いた行動学習のパフォーマンスを向上させるシステムの構築を行う（図 4.1）．

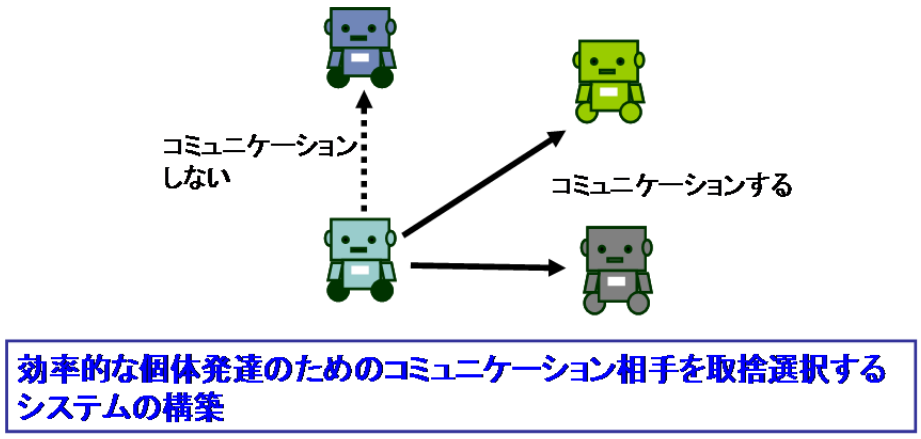


図 4.1: 本章の目的

4.5 コミュニケーション相手の取捨選択による個体学習促進システムの概念

本章で提案するシステムの概念図を図 4.2 に示す．本システムは，コミュニケーション個体の選択を学習する部分と直面する状況に対して適切な行動を学習する部分の 2 つの学習を行う．エージェントが保持する知識は，他者に関する知識（自身から他者に対しての評価 (V)）と行動に関する知識 (Q) となる．他者に関する知識 V はコミュニケーション相手を取捨選択する際の指標なり，行動に関する知識 (Q) は直面する状況に対して行動を選択する際の指標となる．それぞれの知識は，行動の結果得られる報酬を基にして更新，蓄積される．他者からの情報は，自身の知識とともに行動選択に利用することで自身の保持する知識量を

増やし，行動選択に反映する．エージェントは自身にとって適切なコミュニケーション相手を学習すると同時に直面する状況に適した行動を学習する．

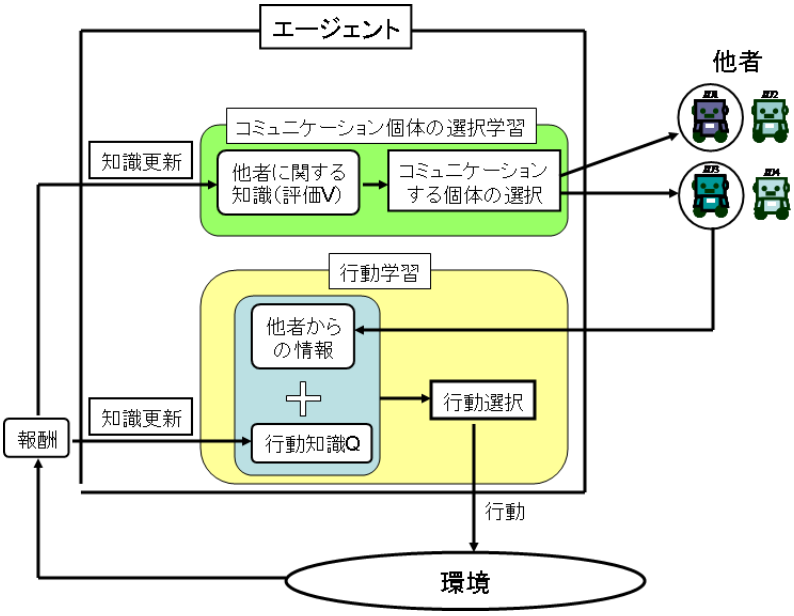


図 4.2: システムの概念図

4.6 強化学習を用いた提案システム

4.6.1 システムの概要

本章では，2 章で紹介した強化学習の枠組みを用いてシステムを構築し，実験により有効性を検証する．強化学習は実ロボットに使われることが多い手法である [97]-[101]．本章で構築するシステムは，行動した結果すぐに報酬が入手できるような即時報酬型（行動に対する報酬が即時与えられる）の環境に対しての手法である．強化学習の適用部は図 4.3 の赤い枠で囲ってある部分である．コミュニケーション個体の選択学習と行動学習の 2 つの部分でそれぞれ別個に強化学習を使用する．これは，他者に関する知識と行動知識という 2 つの異なった知識を更新するためである．

作成するシステムの流れを図 4.4 に示す．まず，他者に対する知識を基にコミュニケーションする個体を決定し，コミュニケーションを行う．次にコミュニケーションにより得た情報と自身の知識から行動を選択する．そして，行動の結果得られた報酬から他者に対する知識と自身の行動知識を更新する．この流れを繰り返すことで学習を進める．

4.6.2 コミュニケーションに用いる情報

本研究では強化学習の枠組みを用いて問題を考える．強化学習では，学習空間は，知覚能力 s と意思決定能力 a と知識 $Q(s,a)$ が構成する空間となる (図 4.5)．エージェントはこの学習空間を探索し，知識 $Q(s,a)$ を更新していく．コミュニケーションに用いる情報としては，自身から送信する情報として，現在状態 s_k を送る．そして，他者からは状態 s_k において，選択することの出来るすべての行動に対する Q 値を得る（図 4.6）．

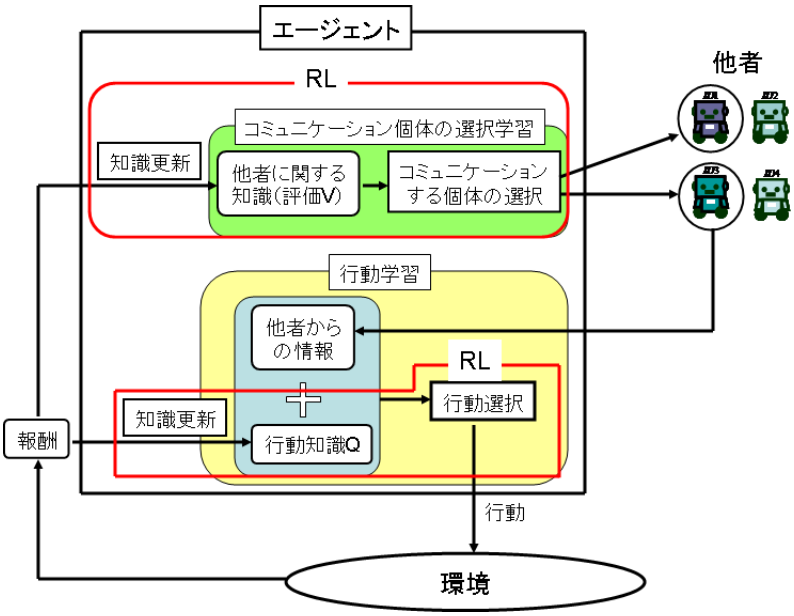


図 4.3: システムの概念図 RL 採用部

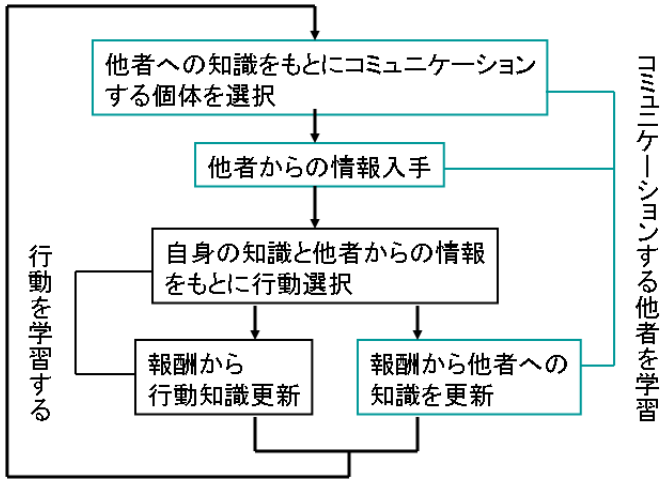


図 4.4: システムの流れ

4.6.3 コミュニケーション相手の選択方法

コミュニケーション相手の選択は，自身の他者に対する評価を基にして行う．ある他者への評価が高い場合，その他者から提供される情報は自身にとって有益であると考えられる．コミュニケーションはそのような他者と行なうことが望ましい．そこで，他者への評価が高いほど高確率でコミュニケーションを行なうようにする．個体 i の保持する評価値は図 4.7 のようにエージェントごとに保持する．この評価値はエージェントが直面する状態ごとに保持するものではなく，環境に対して 1 つだけ保持する．つまり，状態に依存しない知識となる．

$V_i(j)$ を個体 i の個体 j に対する評価値とすると，個体 i が個体 j をコミュニケーション相手として選択する確率 $P_i(j)$ は式 (4.2) で算出される．式 (4.1) で自分自身の評価も含め，すべての個体の評価値を比較し最大のものを V_{max} として算出する．そして，式 (4.2) では，自分以外の他者に対してそれぞれ V_{max} を基準として選択確率を算出する．自分自身に対する評価も含め最大評価値 V_{max} を算出することで，他者を信じきることが無いようにする．特に学習初期では，コミュニケーションによって得た情報よりも，自分自身の経験を頼りに行

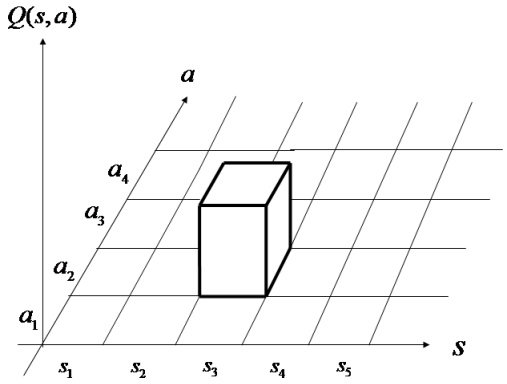


図 4.5: エージェントの学習空間

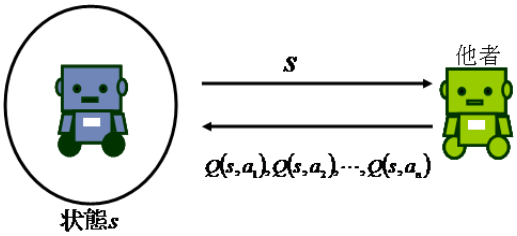


図 4.6: コミュニケーション情報

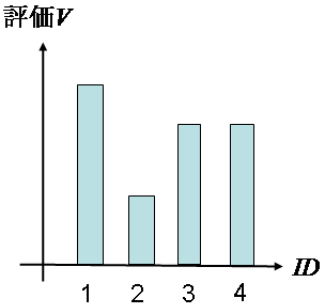


図 4.7: 自身の他者に対する知識

動したほうがよい場合が多いと考えられる．このようなときは，コミュニケーションを控える必要がある．そのため，自分自身も評価基準とするように V_{max} を算出する．

$$V_{max} = max_j V_i(j) \tag{4.1}$$

$$P_i(j) = \frac{V_i(j)}{V_{max}} \quad (i \neq j) \tag{4.2}$$

4.6.4 他者の評価方法

ここでは，強化学習の報酬環境の違いにより即時報酬環境の場合と遅延報酬環境の場合に分けられる．

即時報酬環境

他者に対する評価の更新方法は，入手した情報と実際に入手した報酬との差を基にして決定する．他者から提供される情報とその情報を基に自身が実際に入手した報酬の差が大きければ評価を下げ，小さければ評価を上げる．これは，情報と実際の結果が近ければ近いほど自身にとってその他者の情報は正しく，それに伴いその他者自体も信用できるという考え方を基にしている．今回，他者から提供される情報は，自身が直面している状態に対して，その状態でとることのできるすべての行動の Q 値である．しかし，自身が選択し結果を得ることが出来るのは，1 行動のみなので，評価はその行動のみに絞り評価する．

状態行動対 (s_k, a_l) に対する個体 i の入手した報酬 r_i と個体 j の提供する情報の差 $D_i(j)$ は式 (4.3) で算出される．

$$D_i(j) = |r_i - Q_j(s_k, a_l)| \tag{4.3}$$

個体 i から個体 j への評価を $V_i(j)$ とすると，評価の更新は $D_{i,j}$ を基に式 (4.4) で行なわれる．τ_v は学習率 $(0 \leq \tau_v \leq 1)$ である．なお，この信頼度の更新は情報交換を行なった全ての他者に対して行なわれる．

$$V_i(j) \leftarrow V_i(j) + \left(e^{-\frac{Diff}{\tau_v}} - V_i(j)\right) \tag{4.4}$$

遅延報酬環境

他者に対する評価の更新方法について説明する．即時報酬環境では，エージェントが行動する度に受け取った報酬から他者評価を更新していた．しかし，遅延報酬環境では目的を達成したときにのみ報酬が与えられるという性質上，エージェントが行動する度に他者評価をすることができない．そこで，報酬を受け取った時点で過去に遡って他者評価を行う．評価はエージェントが報酬を受け取るまでの各ステップごとに受け取った報酬を基にして，評価対象となる他者に対して行う．評価対象となる他者は，エージェントがコミュニケーションを行い情報を採用された他者となる．本手法においてコミュニケーション情報は，自身の現在状態 s における最良行動 a_{best} とその評価値 $Q(s, a_{best})$ である．したがって，エージェントの選択した行動と同じ行動を情報として提供した他者が，情報を採用された他者となる．他者情報の採用の概念図を図 4.8 にしめす．図 4.8 のように各時間ごとに情報を採用した個体を評価対象として記録しておく． この評価対象を記録を個体毎に定式化したものが式 (4.5) である．$CommLog_i(j)$ は個体 i の個体 j に対する情報採用の記録である．報酬を受け取った時のステップ数を t とする．$CommLog_i(j)$ の Log_t には t 時点までの各時点で情報を採用したかどうかが格納される．格納される値は式 (4.6) に従う．情報を採用した場合には 1 が，情報が採用されなかったまたはコミュニケーションを行っていない場合には 0 が格納される．記録の格納イメージを図 4.9 に示す．図 4.9 のようにエージェントはすべての他者に対して採用したかどうかの情報を保持する．

$$CommLog_i(j) = [\ Log_1 \quad Log_2 \quad \cdots \quad Log_t\ ] \tag{4.5}$$

$$Log_t = \begin{cases} 1 & (\text{コミュニケーションを行いかつ情報を採用した個体}) \\ 0 & (\text{コミュニケーションしていない，または情報を採用していない場合}) \end{cases} \tag{4.6}$$

他者の評価は報酬を入手した段階で行なわれる（迷路タスクならばゴールした時点）．評価の基となる情報は報酬と $CommLog$ である．また，評価はより最近にコミュニケーションした他者に対して大きい重みをつける．これは，より最近の情報ほど報酬獲得に貢献する情報を提供した個体である可能性が高いためである．更新式を式 (4.7) に示す．$V_i(j)$ は個体 i

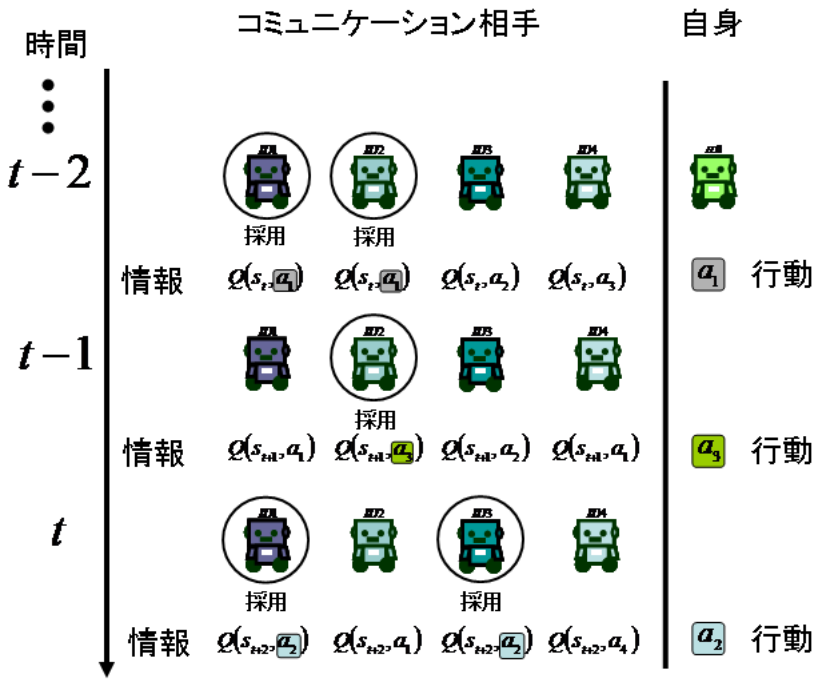


図 4.8: 情報を採用された他者の例

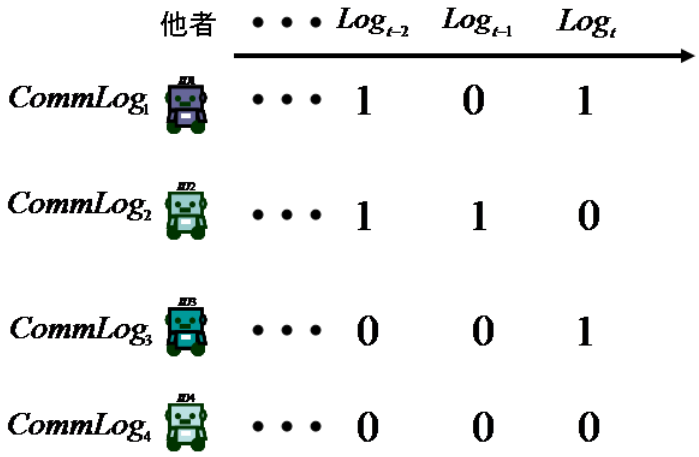


図 4.9: CommLog の格納イメージ

の個体 j に対する評価値を表す．$CommLog_i(j)$ は個体 i の個体 j に対する情報採用記録を表す．また，r は報酬，行列 Γ は割引率 γ の累乗が行列の要素として格納されており，最後の列要素に近くなるにつれ重みが増すようになっている（式 (4.8)）．$A_i(j)$ は個体 i が個体 j の情報を採用した回数である．式 (4.7) は 1 回のコミュニケーション当たりに得られる報酬の期待値を計算している．なお，γ_v は $0 \leq \gamma_v \leq 1$，α_v は $0 \leq \alpha_v \leq 1$ の範囲の値をとる．

$$V_i(j) \leftarrow V_i(j) + \alpha_v(fracr \times CommLog_i(j) \times \Gamma A_i(j) - V_i(j)) \tag{4.7}$$

$$\Gamma = [\ \gamma_v^{t-}1 \quad \gamma_v^{t-2} \quad \cdots \quad 1\ ] \tag{4.8}$$

4.6.5 コミュニケーション情報の利用方法

行動の選択は自身の Q 空間と他者の情報を合成して一時的な Q 空間を作成し，その空間を用いて，行動選択を行なう（図 4.10）．したがって，自身の Q 空間が他者の Q 空間によって改変されることはない．このため，他者情報が自身の知識を改変することによる学習効率の低下が起きない．

状態行動対 (s_k, a_l) において，個体 i の Q 空間を $Q_i(s_k, a_l)$，他者 j の Q 空間を $Q_j(s_k, a_l)$ とすると一時的に作成する個体 i の Q 空間 Q_{tmp_i} は式 (4.10) によって定義する．なお，γ_{tmpQ} は割引率 $(0 \leq \gamma_{tmpQ} \leq 1)$ である．式 (4.10) は，状態 s_k でとることの出来る全ての行動に対して実行される．

$$Q_{tmp_i}(s_k, a_l) = Q_i(s_k, a_l) + \gamma_{tmpQ} \sum_{j \in M} Q_j(s_k, a_l) \tag{4.9}$$

(M はコミュニケーションを行なった個体の集合)

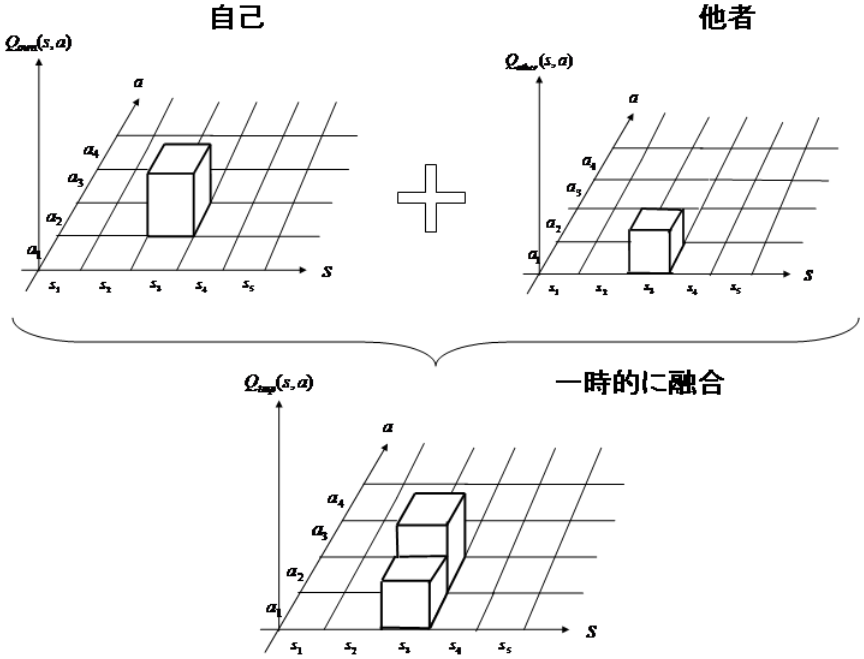


図 4.10: 一時的な Q 空間の作成（この例では他者は 1 体だが，実際には複数いる可能性がある）

4.6.6 行動学習

行動学習部では，強化学習の報酬設定により異なる手法を用いる．よって行動学習部の設定は個々の実験で解説する．

4.7 多ゴール迷路環境における提案システムの有効性の確認

4.7.1 実験概要

提案システムの有効性の検証のために迷路タスクを適用する．検証はシミュレーションによって行う．本実験で用いる迷路タスクでは，各エージェントに対し数個のゴールからそれ

ぞれランダムに割り当てる．迷路の構造としては，ゴールまでのルートは 1 つではなく複数あるものを考える．また，コミュニケーションは知識量が多いものと知識量が少ないものとが行なう場合が最も効果的に作用すると考えられる．そこで，本実験ではエージェントが一定ステップ数ごとに上限数に達するまで個体の追加を行い，上限数に達した場合は古いエージェントから順に新しいエージェントと交換を行なう．これにより新しい個体（知識量が少ない）と古い個体（知識量が多い）が共存する環境になる．実験はエージェントのステップ数が上限に達するまで行なう．

結果は提案システムとランダムにコミュニケーションを行なったもの，全ての個体とコミュニケーションを行なったもの，コミュニケーションを行なわないものの 4 つの手法で比較を行なう．

なお，実験は即時報酬環境と遅延報酬環境の 2 種類について行う．以降は，それぞれの実験環境と実験設定を示す．

4.7.2 実験環境

本実験で作成する迷路環境の意義

本実験で作成する迷路は，ゴールに至る道のりが複数ある迷路である．このような複数の解が存在する環境では，単体学習では局所解に陥りやすい．コミュニケーションによる他者と情報交換をすることで局所解を脱出し，よりよい解へたどり着くことが可能である．また，各エージェントに複数のスタート・ゴールを割り当てることで，同一環境上でも異なった状況（ゴール）におかれるようにした．これにより様々な状況（スタート・ゴール）におかれる個体が存在する．そのため，自身と似た状況に置かれる個体とコミュニケーションしないと有益な情報が得られない．コミュニケーション相手を適切に選択することが高い報酬につながるような迷路環境を作成することで，提案手法の優位性をより顕著に確認することが出来ると考えられる．

迷路環境の作成方法

本実験に用いる迷路環境は $n \times n$ マスのグリッド空間を用いる．迷路環境の作成は棒倒し法を用いて行う．棒倒し法は，図 4.11 のように初期状態を生成する．次に図 4.12 のように，最初の 1 行は上下左右にランダムに壁を作る．そして 2 行目以降は上方向以外の下左右に対しランダムで壁を作っていく．すでに壁があった場合はそのまま次のマスに移る．このようにすることでゴールまでのルートが複数存在する迷路を作成することができる．棒倒し法が終了した時点の例を図 4.13 に示す．この後図 4.13 に複数のゴールを設定することで迷路が完成する．本実験では，スタートは 1 箇所，ゴールは一定数設置する．設置方法は，スタートの場合は 1 行目の通路上にランダムに配置する．ゴールの場合は最終行の通路上にランダムに配置する．スタートを 1 箇所，ゴールを 3 箇所配置した場合の例を図 4.14 に示す．エージェントはそれぞれゴールをランダムに指定される．

エージェントに関して

実験に使用するエージェントは迷路環境内を上下左右に動くことができる．エージェントは，1 行動あたり 1 マス移動することができる．また，エージェントは現在位置を座標で把握することができる．この座標は迷路環境の左上を原点としている．1 座標は迷路環境の 1 マスに相当する．

また，エージェントは学習開始前にゴールをそれぞれランダムに 1 箇所指定される．ここで指定されたゴールは，学習終了まで固定である．

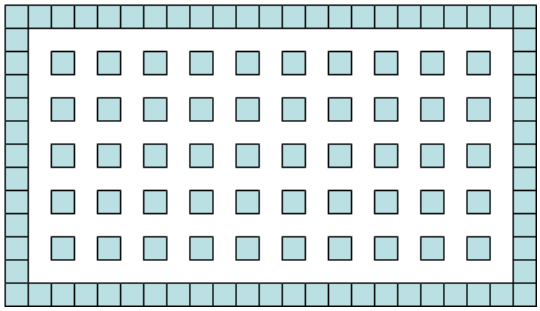


図 4.11: 棒倒し法初期状態

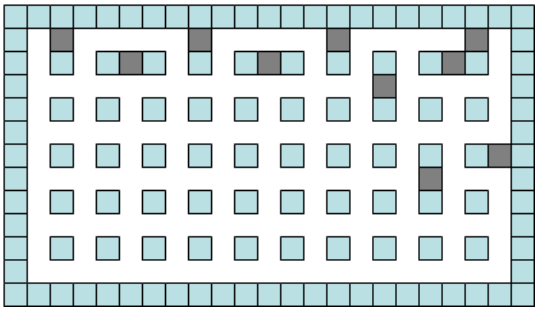


図 4.12: 棒倒し法イメージ

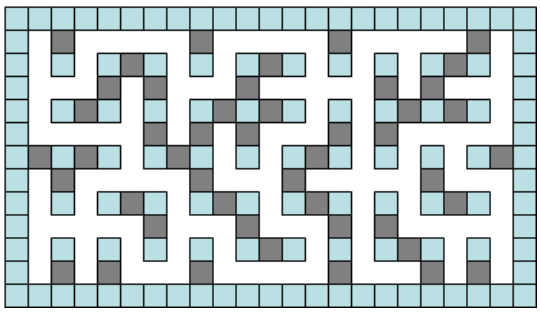


図 4.13: 棒倒し法終了時の迷路

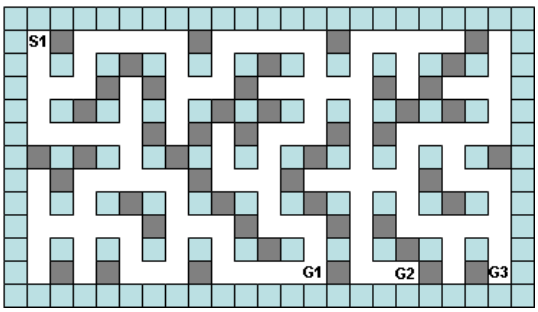


図 4.14: スタート・ゴールの設定の例

4.7.3 即時報酬環境での実験

報酬設定

即時報酬環境での報酬設定は，すべてのマスに報酬が設定されている．各マスの報酬は，エージェントの現在位置とゴール位置の差を基に決定する．エージェントの現在位置がゴールに近いほど，高い報酬が得られる．エージェントの現在位置とゴール位置の差 d_r は式 (4.10) で決定する．

$$d_r = |goalX - currentX| + |goalY - currentY| \tag{4.10}$$

d_r を基に報酬を決定する．報酬は式 (4.11) で決定する．

$$r = r_{goal} \times \gamma_r^{d_r} \tag{4.11}$$

r_{goal} はゴール地点で得られる報酬，γ_r は係数である．γ_r の大きさによって，現在位置とゴール位置の距離に応じた報酬の増加幅が変わる．

行動学習部の手法

行動学習手法として加重平均手法を用いる．また行動選択手法としては softmax 法を使用する．

実験パラメータ

実験パラメータを表 4.1 に示す．スタートとゴールはともに 6 種類である．エージェントは試行循環ステップ数は 500 なので，1 エージェントあたりの学習時間は 循環ステップ数 × エージェント数 = 500 × 80 = 40000 ステップ となる．

表 4.1: 即時報酬環境実験パラメータ

迷路のサイズ n	50×50
ゴールの数	4
ゴール報酬	20
エージェントの数	80
総ステップ数	100000
循環ステップ数	500
τ_v	0.3
α_v	0.1
行動学習部加重平均手法 α_{act}	0.1
行動学習部 softmax 法 tau_{act}	4
γ_{tmpQ}	0.05
γ_r	0.95

実験結果・考察

実験結果を図 4.15，図 4.16 に示す．図 4.15 は各ステップの 1 個体あたりの平均獲得報酬量である．ステップ数が 40000 回まではエージェント数は上限数（80 体）存在せず，循環ステップ数（500 ステップ）ごとに 1 体ずつ追加されている．40500 ステップ以降は循環ステップ数ごとに最も古い個体を取り除き新しい個体を加えている．よって，個体の循環が行なわれるようになるのは 40500 ステップ以降である．図 4.15 において，25000 ステップあたりから提案手法の平均獲得報酬量が他の手法を上回っている．25000 ステップ以降からはエージェント数は 25 体程度となり，共通するゴールを持つ個体も徐々に増加する．そのため，コミュニケーション相手の取捨選択が有効に作用し始め，獲得報酬の増加につながっていると考えられる．以降エージェント数が増えていくと共に獲得報酬量も増加していく．これはエージェントが増えることで自身と共通のゴールを持つ個体が増え，よりコミュニケーション相手の取捨選択が有効に働くようになったためである．

提案手法以外の手法では，平均獲得報酬に波がありどの手法も似たような平均獲得報酬量である．全個体とコミュニケーションした場合は，有益な情報と不利益な情報の両方が入ってくる．そのため不利益な情報に自身の意思決定が影響され，提案手法ほど高い獲得報酬は得られなかったと考えられる．また，ランダムにコミュニケーションを行なった場合では，コミュニケーションする相手の数および，相手そのものもランダムに決定する．そのため，コミュニケーションを通じて入手する他者からの情報の質が安定しない．これが，学習に大きな影響を与え，提案システムに比べて平均獲得報酬量を下げている要因であると考えられる．コミュニケーションなしの場合は，自身が得る情報のみから学習を行うしかない．そのため，学習そのものが提案システムよりも遅い．そのため，平均獲得報酬に違いが出ていると考えられる．

これは，生涯獲得報酬の平均を示す図 4.16 が分かりやすい．このグラフは各エージェントがエージェント入れ替えまでに獲得した報酬量の平均である．提案システムの生涯獲得報酬は他の手法に比べて 20 %程度高くなっている．その他の手法は同程度の獲得報酬量となっている．コミュニケーションなしの場合が若干ではあるがすべての個体とコミュニケーションする場合とランダムにコミュニケーションを行う場合に比べて生涯獲得報酬が高い．これは，不適切なコミュニケーション相手とコミュニケーションを行うことは学習に悪影響を与えることを示している．このことから，コミュニケーション相手の取捨選択は学習に良い影響を与えることが確認できた．以上から，本手法は有効であるといえる．

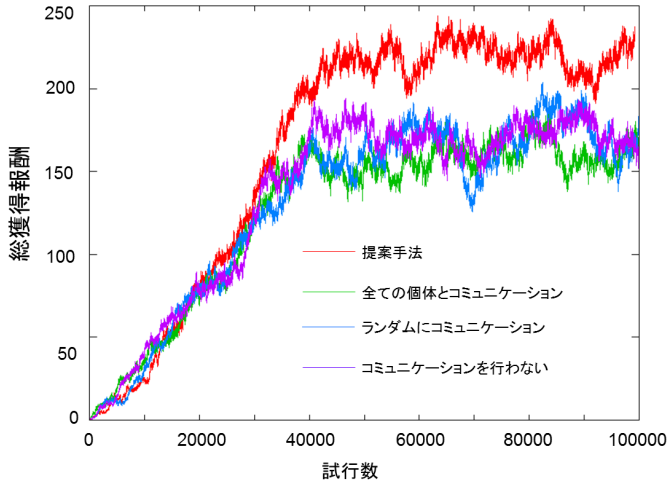


図 4.15: 各ステップの 1 個体あたりの平均獲得報酬量

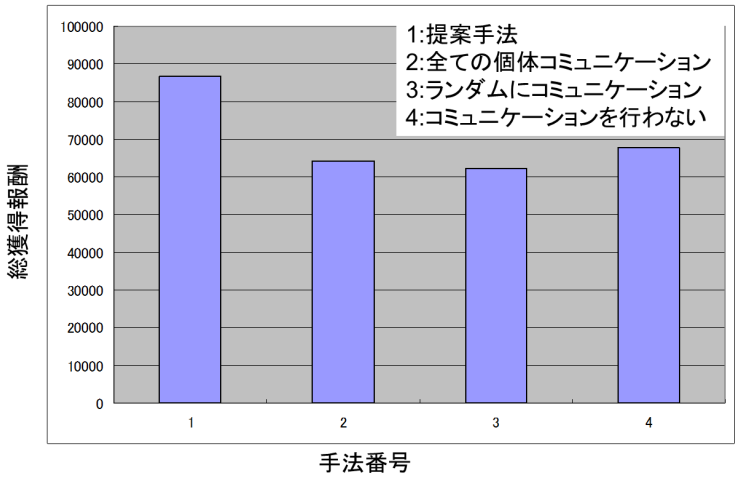


図 4.16: 1 個体あたりの生涯獲得報酬量

4.7.4 遅延報酬環境での実験

遅延報酬環境での実験について，以降で即時報酬環境とは異なる点を述べる．

報酬設定

即時報酬環境ではすべてのマスに報酬が与えられていた．しかし，遅延報酬環境では，報酬はゴールにたどり着いたときのみ与えられる．

行動学習部の学習手法

行動学習部では，行動評価手法に Q 学習，行動選択手法に ε -greedy 法を用いる．Q 学習は遅延報酬のある強化学習問題に向いている手法である．

4.7.5 実験パラメータ

パラメータ設定を表 4.2 に示す．迷路のサイズは，即時報酬環境よりも小さく，また総ステップ数と循環ステップ数は多くとっている．これは，遅延報酬環境では，即時報酬環境に比べて Q 値の伝搬が遅く学習に時間を要するためである．遅延報酬環境でのエージェント 1 体当たりの活動時間は $50 \times 3000 = 150000$ ステップである．

表 4.2: 遅延報酬環境パラメータ設定

迷路のサイズ	21×21
スタートの数	1
ゴールの数	3
ゴール報酬	1
エージェントの数	50
総ステップ数	600000
循環ステップ数	3000
α_v	0.5
γ_v	0.9
α_{act}	0.5
γ_{act}	0.8
γ_{tmpQ}	0.01
ϵ	0.05

4.7.6 実験結果・考察

実験結果を図 4.17 に示す．図 4.17 から，コミュニケーション相手の取捨選択を行った場合の 1 個体あたりの生涯獲得報酬量は他の手法に比べ多いことがわかる．また図 4.18 から，群全体で見た場合の生涯獲得報酬量の総和についてもコミュニケーション相手の取捨選択を行った方がより多くの報酬を獲得していることがわかる．これは，コミュニケーション相手の取捨選択により，行為主体者は自身の発達に有益な情報を多く入手することができる．そのため，コミュニケーション相手の取捨選択が行動学習の効率を向上したことがいえる．

ランダムにコミュニケーションを行った場合と全個体とコミュニケーションを行った場合では，世代数が 30〜50 世代のあたりでコミュニケーションなしの手法に生涯獲得報酬量において低い結果となっている．これは，ランダムにコミュニケーションを行う場合と全ての個体とコミュニケーションを行う場合では，悪影響を与える情報を取り入れやすいためである．ランダムにコミュニケーションする場合では，コミュニケーション個体数と相手をランダムで決定するため自身に悪影響を与える情報を持つ個体ともコミュニケーションを行ってしまう．

以上のことから，自身にとって有益な他者を学習するまでは一時的に学習効率は落ちるが，他者の学習が完了すると行動学習の効率が上昇するといえる．したがって，提案手法はごく短時間の学習時間では効率的な学習は行えないが，長期的な学習では十分に効率的な学習が可能になるといえる．

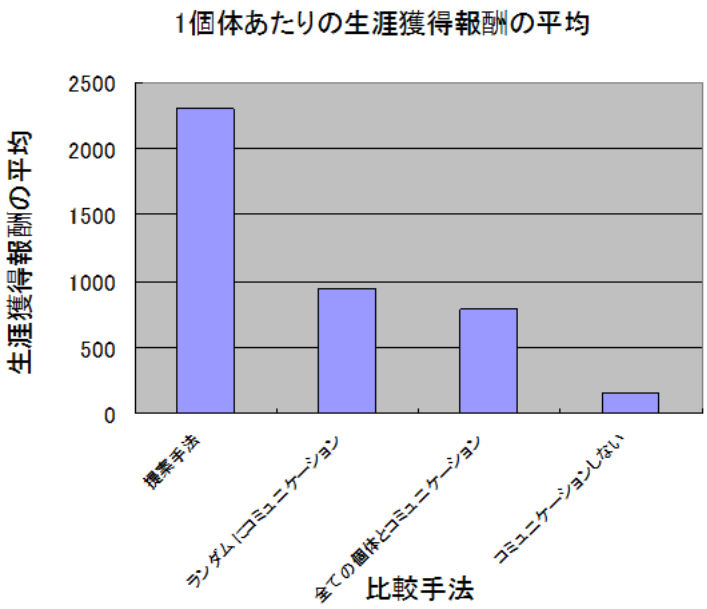


図 4.17: 1 個体あたりの生涯獲得報酬平均

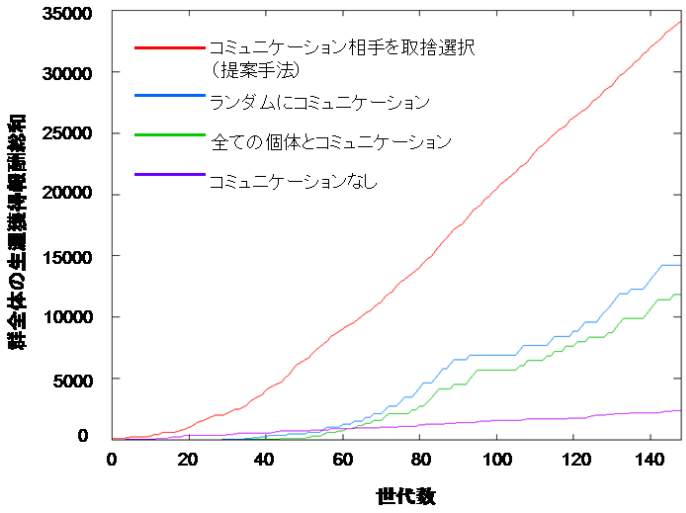


図 4.18: 群全体の獲得報酬総和

4.8 まとめ

本章では，従来のコミュニケーションによる学習の問題点を述べた．問題点の中のコミュニケーション情報の内容に注目し，自身と同じ目的の他者とコミュニケーションすることが望ましいことを述べた．そのために，コミュニケーション相手の取捨選択による個体学習促進システムを提案した．提案手法を即時報酬環境及び遅延報酬環境に適用し，その有用性を確認した．

第5章　センサ情報の取捨選択による学習の高速化

第3章および第4章では，センサ情報としての他のロボットからの情報に注目した．そして，他のロボットからの情報を有効利用し，自己がより効率的な発達が可能なシステムを提案した．

ここからは，個体ロボットにおける内部情報の取捨s選択として，センサ情報の扱い方に注目する．扱うセンサ情報として，自身のセンサを通した環境情報を考える．本章では，その環境情報を効果的に用い自己を効率的に発達させるシステムについて議論する．

5.1　ロボットの学習とセンサ情報の問題

ロボットの学習は，センサから入力される情報をもとに周囲の状況に対して適切な行動を学習する．ロボットに搭載されるセンサは，従来ロボット設計者によって決定される．ロボット設計者は，ロボットが適用されるタスクや環境を想定しそれに適したセンサを搭載する．例えば，工場用のアーム型ロボットであれば，対象が把持されているかを知るために感圧センサが搭載される．

近年，ロボットは様々なタスクに対応できるように，多様なセンサを多数搭載されることが多くなっている．このようなロボットは様々なタスクに対応することができる汎用性をもつ反面，それらに応じて使用するセンサの設定が難しくなる．すなわち，ロボットの設計者にとってロボットが使われるタスクを想定し，それらに適応するために重要となるセンサの決定が難しい．例として，図5.1のような場合を考える．ロボットには金属検知センサ・赤外線センサ・音センサが搭載されている．このとき，ロボットが空き缶拾いタスクを行う場合，空き缶との距離を判断する赤外線センサや金属検知センサが重要なセンサであり，音センサは重要なセンサではない．一方で，警備タスクを行う場合では，不審な音を感知するために音センサおよび不信者を感知する赤外線センサは重要であるが，金属検知センサは重要ではない．このように，タスクに応じて重要となるセンサは異なる．

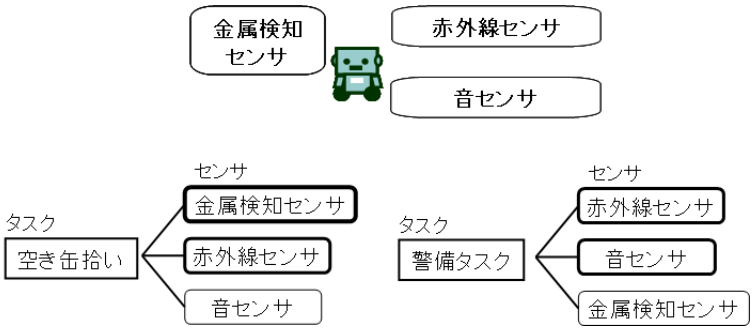


図 5.1: タスクによる重要センサの違い

この問題に対する最も簡単な解決策として，学習にすべてのセンサを用いることが挙げられる．すべてのセンサを用いることで，設計者がタスクに対して重要となるセンサが何であるかを考える必要はなくなる．またロボットにとっては入力としてタスク遂行に重要となる

センサからの情報を含んでいる．しかし，すべてのセンサを使う分，設計者が使用するセンサを決定する場合に比べて入力情報が多くなり，それに対して適切な行動を発見するための探索量も多くなる．その結果，学習に要する時間が多くなるという問題がある．実環境では，学習のために使うことができる時間は有限である．少なくとも，使用者である人間が許容出来る時間の範囲内に学習時間を収めることが望ましい．従って，学習時間が増える要因は可能な限り取り除くことが学習時間の短縮に重要な要素である．

5.2 重要センサを判別し学習に利用する手法

すべてのセンサ情報を入力する場合，あらゆる状況に対応することができる反面，学習に要する時間が増大する．この問題に対して，学習に必要なセンサのみを用いて学習を行うことで学習時間を短縮することを考える．本章では，タスクに対して有用なセンサを特定し，それらを学習に利用する手法の提案を行う．また，トータルシステムとして提案手法を用いた学習システムを構築する．

なお，提案手法および構築する学習システムは各学習手法依存である．そこで，本システムは強化学習を用いて構築する．

5.3 提案手法の構成

提案システムは大きく分けて以下の 2 つのモジュールで構成される．

- 重要センサの特定
- 重要センサの強化学習への利用

提案手法は，重要センサの判別モジュールおよび重要センサを学習に利用するモジュールの 2 つのサブモジュールにより構成されている．以降にそれぞれのモジュールについて説明していく．

5.3.1 重要センサの特定

センサ情報の取捨選択を行う基準

学習に使用するセンサの取捨選択には基準となる情報が必要となる．提案手法では，センサの値と強化学習における報酬の相関関係に注目した．強化学習における報酬はタスクの達成度を表す．タスク達成に重要なセンサを用いた場合とそうでないセンサを用いた場合では，ロボットが得る報酬に違いがある．タスクに必要なセンサを用いた場合は，ロボットが得る報酬は多くなる．この時，必要なセンサの値の増減と報酬には相関がある．例として，図 5.1 のロボットが空き缶拾いタスクを考える．ロボットが空き缶に近づき，自身のマニピュレータで空き缶を拾うことができる距離まで移動し，空き缶を拾うことでタスクが達成される．このときロボットと空き缶までの距離に応じた報酬を得る．ロボットが空き缶に近づくほど，つまり空き缶までの距離が短いほどタスクの達成度は大きくなるため報酬は多くなる．空き缶までの距離は赤外線センサによって計測することができる．従って，このとき赤外線センサの値とロボットが得る報酬には相関関係がある．一方で，タスク達成に重要ではないセンサと獲得報酬には相関関係が存在しない．そのようなセンサの値がどうであれ，タスクの達成に直結しない．空き缶拾いロボットにおける，音センサはタスク達成に重要ではない．これは，空き缶が常に音を発する訳ではなく，仮に空き缶が倒れるなどしてロボットの近くで音が出たとしても，それがターゲットの空き缶であるかをロボットは判断できない．そのた

め，音センサの値と報酬には相関がない．以上のようにセンサ値とタスクの達成度である報酬には相関関係が存在する．

このような相関関係を基にタスクに重要なセンサを決定する．重要センサ決定の概念図を図 5.2 に示す．ロボットは各状態でのセンサ値の情報と報酬をデータとして蓄える．そのデータを基に各センサと報酬の相関を割り出す．あるセンサのセンサ値と報酬に高い相関があれば，そのセンサはタスク遂行に必要なセンサであると判断することができる．一方で，相関関係がない場合は不要なセンサであると判断する．つまり，相関の強さがセンサの重要度となる．相関の強さは，相関係数で表される．従って，各状態におけるセンサの値と報酬値の相関係数をセンサの重要度とする．各センサに対してセンサの重要度を算出し，重要度が高いものをタスクに対して必要なセンサとする．

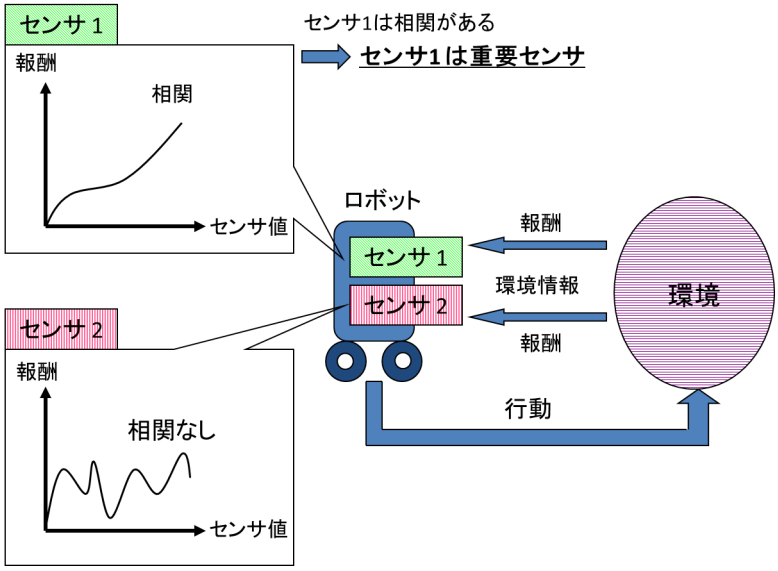


図 5.2: センサ値と報酬の相関をによる重要センサの決定

5.3.2 重要センサの特定プロセス

重要センサ決定のプロセスの概念図を図 5.3 に示す．また，重要センサ決定プロセスを以下に示す．各センサのセンサ値と平均獲得報酬のデータは図 5.4 のような表の形式で保存される．相関係数はこのリストのデータを用いて算出される．重要センサを決める閾値は人間によって決定される．閾値決定の基準として，一般に非常に強い相関があると判断される場合，相関係数の絶対値は 0.8 程度，強い相関で 0.6 程度とされている．

1. 各時間における，その時のセンサ値と平均獲得報酬をデータとして記録
2. それらのデータを基に各センサの相関係数を計算
3. センサの相関係数が設定された閾値を超えた場合，そのセンサは重要センサと判断する
4. 手順 1 にもどる

相関係数算出用知識リストの平均獲得報酬の更新式について述べる．状態 s_i を式 (5.1) で定義する．i は図 5.4 における状態番号である．$e_{i,j}$ センサ j のセンサ値である．E_j はセンサ j が表現可能なセンサ値の集合である．

$$s_i := \{e_{i,1}, e_{i,2}, \cdots, e_{i,j}, e_{i,n} | e_{i,1} \in E_1, e_{i,2} \in E_2, \cdots, e_{i,j} \in E_j, e_{i,n} \in E_n\} \tag{5.1}$$

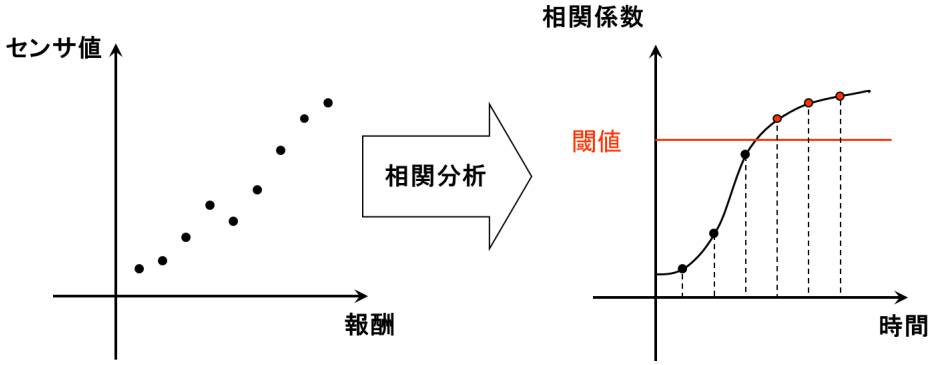


図 5.3: 重要度センサ決定プロセスの概念図

状態番号	センサ1のセンサ値	センサ2のセンサ値	・・・	センサ n のセンサ値	平均獲得報酬 $\bar{r}'_s$
1	S	56	・・・	111	45
2	23	45	・・・	123	23
⋮					
m	90	10	・・・	15	132

図 5.4: 相関係数算出用知識リストの例

報酬の平均化手法に関しては，標本平均や加重平均など任意の手法を用いる．

ロボットは相関係数算出用知識リストを用いて相関係数を計算する．相関係数の算出にはピアソンの積率相関係数 [81] の算出式を用いる．時間 t におけるセンサ j の相関係数を $c_{t,j}$ とすると相関係数は式 (5.2) で計算される．ここで，$\bar{e_j}$ はセンサ j のセンサ値の平均であり式 (5.3) で定義される．$\bar{r'}_t(s_i)$ は相関係数算出用リストの平均獲得報酬の平均であり式 (5.4) で定義される．

$$c_{t,j} = \frac{\sum_{i=1}^{m}(e_{i,j} - \bar{e_j})(r'_t(s_i) - \bar{r'}_t(s_i))}{\sqrt{\sum_{i=1}^{m}(e_{i,j} - \bar{e_j})^2}\sqrt{\sum_{i=1}^{m}(r'_t(s_i) - \bar{r'}_t(s_i))^2}} \tag{5.2}$$

$$\bar{e_j} = \frac{1}{n}\sum_{i=1}^{n} e_{i,j} \tag{5.3}$$

$$\bar{r'}_t(s_i) = \frac{1}{n}\sum_{i=1}^{n} r'_t(s_i) \tag{5.4}$$

相関係数の絶対値 $|c_{t,j}|$ が閾値 Th を超えたときにそのセンサが重要センサであると判断する．この閾値 Th は人間によって決定される．

5.4 重要センサの学習への利用

重要センサが決定した後にそれらを強化学習に利用する方法について説明する．強化学習では，学習データは Q 空間に保存される．Q 空間は，各センサ軸と行動軸，Q 値軸によって構成されている．従来の強化学習では，現在状態における各行動 Q 値を参照して行動を決定する．提案手法では，意思決定の際に参照する Q 空間を，新たに構築する．構築する Q 空間は，すべてのセンサによって構成されている Q 空間から重要センサのみを用いた Q 空間である．この Q 空間は，行動決定のための一時的なもので，構成元であるすべてのセ

ンサを用いた Q 空間は保持される．このようにすることで，タスクが変わっても Q 空間を初期化することなく使用可能になる．そのため，タスクが変化後でも使用可能な知識をそのまま利用することができる．

一時的な Q 空間の構成例を図 5.5 に示す．この例では，ある行動 a に対する一時的な Q 空間の構成を行なっている．センサ軸は S_1，S_2 の 2 軸であり，センサ S_1 が重要センサの場合を考える．ロボットは重要では無いセンサ S_2 軸を Q 空間から取り除いた Q 空間を新たに構成する．このとき重要ではないセンサ軸 S_2 の Q 値を S_1 の各要素に射影することで S_2 軸を Q 空間から取り除く．この空間は意思決定にのみ使用される一時的なものである．ロボットは，この一時的な Q 空間を基に行動を選択する．重要センサのみの Q 空間を構成することで，重要ではないセンサ軸に存在する Q 値を必要センサ軸に集約させることができる．そのため，従来の Q 空間よりも多くの情報を用いた行動選択が可能となる．

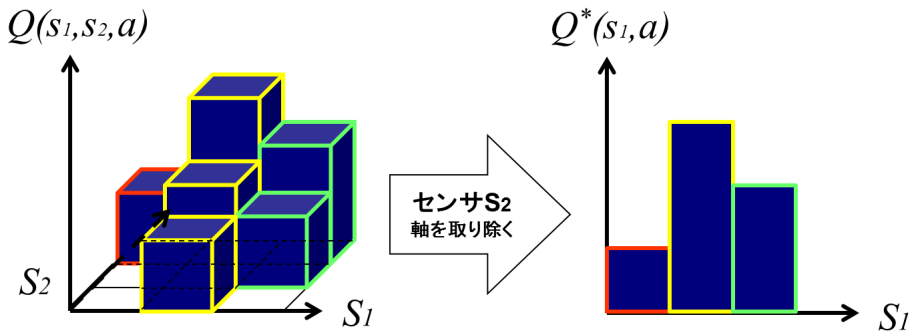


図 5.5: Q 値の射影概念図

一時的な Q 空間の構成は Q 空間全体に対して行う必要はなく現状態のみに対して行えば良い．ここからは，現状態に対する一時的な Q 空間の構成方法について述べる．提案手法では，Q 値そのものを直接取り扱うことはせず，一旦推定獲得報酬に変換する．推定獲得報酬への変換は各状態行動対の Q 値とその状態行動対の経験数の積を計算することで行う．このようにすることで，状態経験数を考慮にいれた Q 空間を構成する．数多く経験された状態であるほど，平均獲得報酬の信頼度が高くなる．そのため，経験数に比例した重みをかけて集約した Q 値を求める．

現状態を s とすると，$s := (e_1, e_2, \cdots, e_n \mid e_1 \in E_1, e_2 \in E_2, \cdots, e_n \in E_n)$ で表される．また，重要センサのみの状態を $s^\star := (e_1, e_2, \cdots, e_p \mid e_1 \in E_1, e_2 \in E_2, \cdots, e_n \in E_p)$，重要ではないセンサのみの状態を $u = (e_{p+1}, e_{p+2} \cdots e_m \mid e_{p+1} \in E_{p+1}, e_{p+2} \in E_{p+2}, \cdots, e_n \in E_n)$ で表す．この時，状態 s で行動 a をとるときの推定総獲得報酬 $R(s,a)$ は式 5.5 で定義される．

$$R(s,a) = R(s^\star, u, a) = Q(s^\star, u, a) \times E(s^\star, u, a) \tag{5.5}$$

$E(s^\star, u, a)$ は状態行動対 $(s^\star, u, a)$ の経験数である．状態行動対 $s^\star, a$ における一時的な Q 値 $Q(s^\star, a)$ は式 (5.6) で定義される．

$$Q(s^\star, a) = \frac{\sum_{\forall e_{p+1} \in E_{p+1}} \sum_{\forall e_{p+2} \in E_{p+2}} \cdots \sum_{\forall e_n \in E_n} R(s^\star, u, a)}{\sum_{\forall e_{p+1} \in E_{p+1}} \sum_{\forall e_{p+2} \in E_{p+2}} \cdots \sum_{\forall e_n \in E_n} E(S^\star, U, a)} \tag{5.6}$$

式 (5.6) は一つの状態行動対のみに注目した式である．ロボットの行動は，その状態における全ての行動に対する Q 値を基に選択される．そのため，全ての行動に対して Q 値の集約を行い，これらを統合して一時的な Q 空間を構築する．ロボットは，構築された Q 空間を基に行動を選択する．この時の行動選択手法は強化学習の各種手法に依存する．

5.5　トータルシステムとしての概念図

図 5.6 に提案手法を用いたトータルシステムを示す．システムは大き分けて以下の 2 つの部分で構成される．

- 提案手法部
- 強化学習部

提案手法部は，先程まで説明してきた重要センサの判別部分および重要センサを学習に利用する部分の 2 つの機能により構成されている．重要センサ決定部でタスクに対して重要なセンサを特定する．重要度利用部において，重要センサのみを用いた一時的な Q 空間を構築する．この一時的な Q 空間は，強化学習部の Q 空間を基につくられる．強化学習部の Q 空間そのものは保持される．ロボットはこの一時的な Q 空間を基に行動の選択を行う．

一方，強化学習部では，タスクの学習を行う．行動選択部において，一時的な Q 空間を基に行動の選択を行う．行動評価部において，Q 空間の更新を行う．この Q 空間の更新は一時的な Q 空間ではなく，強化学習部が保持するすべてのセンサによって構成される Q 空間である．

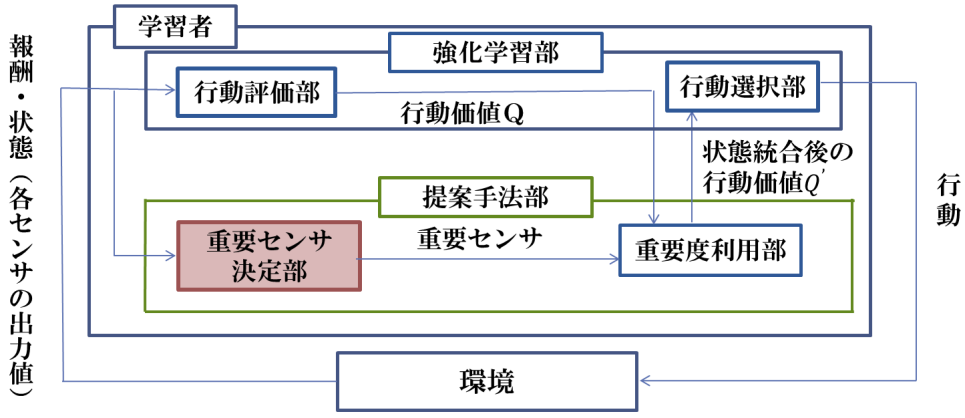


図 5.6: トータルシステムの概念図

次に，システム全体の流れを図 5.7 に示す．ロボットは現在の状態を認識した後，保持している知識を基に重要センサを判定する．ロボットは，判定した重要センサを基に一時的な Q 空間を構成し，それに基づき行動選択を行う．行動の結果得た報酬を基に相関係数算出のための知識リストと Q 空間の更新を行う．この行程を繰り返すことで学習を行う．

5.6　コンピュータシミュレーションによる提案手法の有効性の確認

5.6.1　実験目的

提案手法の有効性の確認のためにコンピュータシミュレーションによる実験を行う．実験の目的は以下の通りである．

- 提案手法がタスクに応じて重要センサを適切に判別していることを確認する
- 提案手法と通常の強化学習を比較し，強化学習に比べて学習の収束が早いことを確認する

実験結果として，相関係数の変化をエピソードの最後の値をプロットしたグラフとエピソード毎の総行動回数をプロットしたグラフの 2 種類のグラフを用意する．これらを基に提案手法の有効性を確認する．

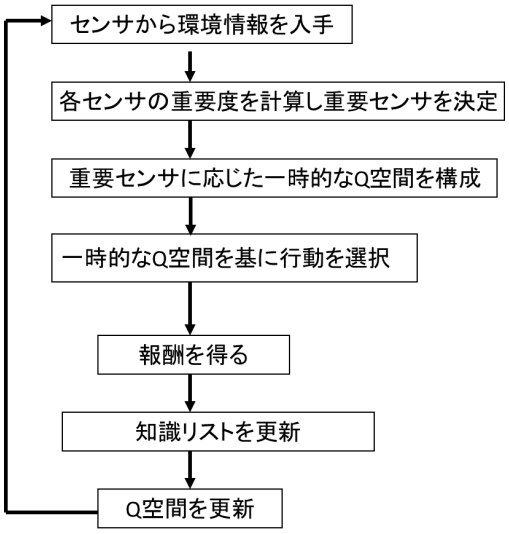


図 5.7: トータルシステムの流れ

5.6.2 実験環境

図 5.8 に示すようなグリッド空間を実験環境とする．この空間は，四方を壁に囲まれており，空間内には壁は存在しない．空間の広さは u マス× u マスである．この空間内にエージェントを配置しタスクを行う．エージェントは，8 方向に移動することが可能であり，1 回の行動で 1 マス移動する．エージェントには 2 つの距離センサが搭載されており，それぞれエージェントと壁 A と壁 B との距離をマスの数で認識する．

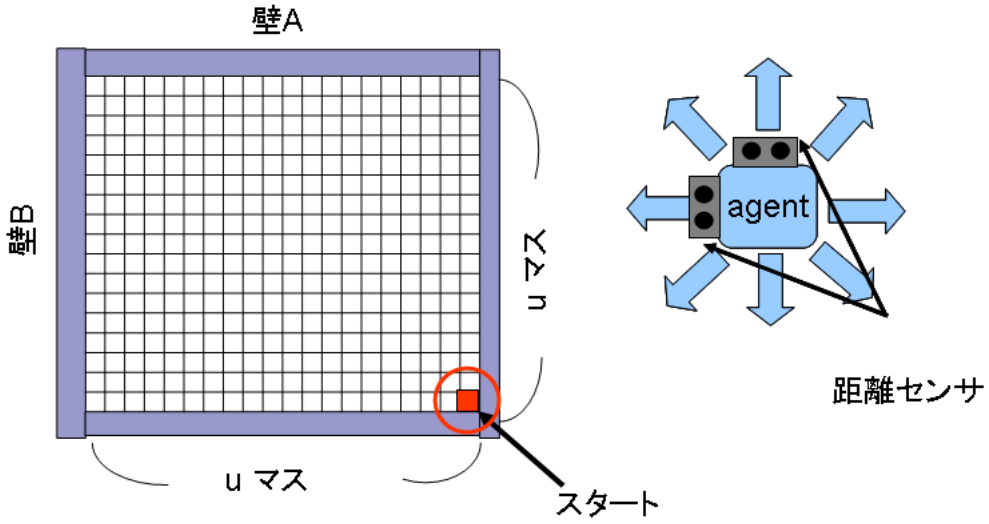


図 5.8: 実験環境とエージェントの設定

5.6.3 実験設定

実験はエピソード型逐次報酬のタスクを採用する．報酬は各状態で与えられ，エージェントが目標状態にたどり着いた段階，または n_{act} 行動した段階で 1 エピソードが終了する．1 エピソード終了後はスタート地点に戻り次のエピソードを行う．エージェントが n_{ep} エピソード終わった時点で，実験終了とする．

本実験では，タスク変更への対応を観察する．そのため，実験途中でエージェントが行うタスクが変わる．タスクは n_{ch} エピソードでタスク A からタスク B に変化する．タスク A はスタート地点から壁 A に到達するタスクである．スタート地点は環境の右下である．この時，エージェントが各状態で得る報酬値は，壁 A までの距離 d_A に基づき式 (5.7) で決定する．このタスクでは，エージェントが壁 A に近づくほど高い報酬を得ることができる．

$$r = u - d_A \tag{5.7}$$

タスク B は壁 B に到達するタスクである．この時，エージェントが各状態で得る報酬値は，壁 B までの距離 d_B に基づき式 (5.8) で決定する．このタスクでは，エージェントが壁 B に近づくほど高い報酬を得ることができる．

$$r = u - d_B \tag{5.8}$$

本実験では，平均獲得報酬の計算に加重平均手法を用いる．加重平均手法はより最近に得た報酬ほど大きい重みをかけて平均値を算出する．加重平均手法を用いることで，タスクの変化に対して柔軟に対応することができる．タスクの変化はすなわち，報酬構造が変化することを意味する．報酬構造が変化すると，それに合わせて相関係数算出用知識リストを更新する必要がでくる．この更新に加重平均手法を用いることで，タスク変化後に得た報酬の重みが大きくなり，より高速に相関係数算出用知識リストを更新することが可能になる．

状態 s_i における平均獲得報酬を $r'_t(s_i)$ とし，その更新式は式 (5.9) で定義される．ここで，$r_t(s_i)$ は時間 t で獲得した報酬である．なお，$\alpha_{ave}(0 \leq \alpha_{ave} \leq 1)$ は重みパラメータである．この値が大きいほど新しい報酬に大きく重みをかける．大きい値ほどタスク変化への追従速度は早くなる．

$$r'_t(s_i) \leftarrow r_t(s_i) + \alpha_{ave}(r_t(s_i) - r'_{t-1}(s_i)) \tag{5.9}$$

エージェントは行動学習手法として加重平均手法を用いる．加重平均手法の学習率パラメータを α_{RL} とする．行動選択手法として，ϵ-greedy 法を用いる．

5.6.4 実験パラメータ

実験パラメータを表 5.1 に示す．本実験では，総エピソード数を 60000 として，30000 エピソード目でタスクがタスク A からタスク B に切り替わる．

表 5.1: 実験パラメータ

u	20
n_{ep}	60000
n_{ch}	30000
n_{act}	1000
α_{ave}	0.1
ϵ	0.1
α_{RL}	0.1
初期 Q 値	0.0
Th	0.7

5.6.5　シミュレーションによる実験結果

実験結果を図 5.9 図 5.10 に示す．図 5.9 は，各エピソードの最終行動時点での相関係数をプロットしたものである．29999 エピソードまではタスク A，30000 エピソード目からはタスク B に切り替わる．タスク変化前は実験開始から数エピソードで壁 A センサの相関が 1 付近に，壁 B センサが 0.1 に推移している．タスク A においては，報酬は壁 A センサのセンサ値のみに依存するため重要センサは壁 A センサとなる．壁 A センサの相関のみが閾値を超えていることから，エージェントは重要センサを壁 A センサと判断している．

タスク変化後は壁 B センサのみが重要センサとなる．タスク変化後数エピソードで壁 A センサの相関係数が急激に下がり，壁 B センサの相関係数が急激に上がっている．最終的に壁 B センサが閾値を超え，エージェントは壁 B センサを重要センサと判断した．この結果から重要センサの判別は適切に行われていることを確認した．

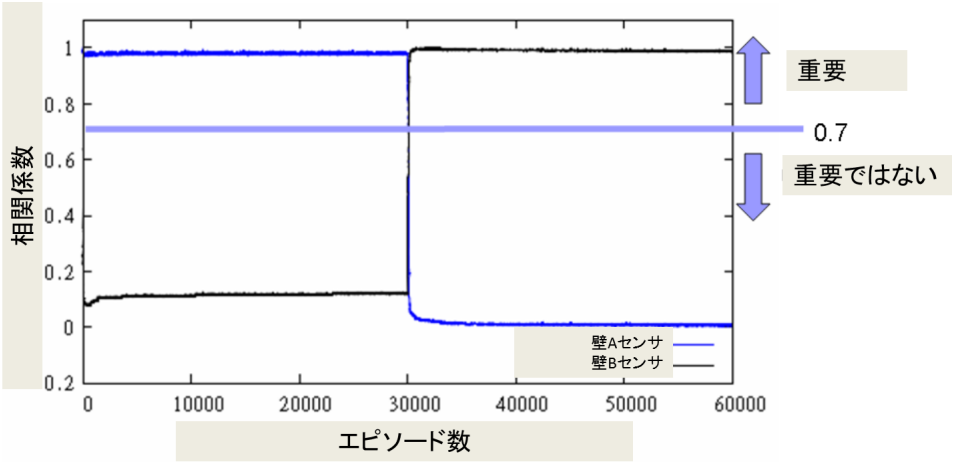


図 5.9: 各エピソードでの相関係数の推移

図 5.10 は，各エピソードで目標状態に到達するまでの総行動回数をプロットしたものである．29999 エピソードまではタスク A，30000 エピソード目からはタスク B に切り替わる．

タスク A では，提案手法が強化学習に比べて少ないエピソード数で総行動数が収束している．提案手法により，タスク達成に不要なセンサ B に関する軸を排除することで，センサ B の Q 値を集約して利用することが可能である．それにより，最適行動に必要な Q 値がより通常の強化学習に比べて少ない経験数で集めることができる．その結果，より少ないエピソード数で学習が収束していると考えられる．

タスクがタスク B に変化した後は，提案手法，従来の強化学習ともに行動数が増加する．このとき，Q 値が変化前のタスクに適したものになっているため，変化後のタスクに適応的な動きが取れなくなっている．再学習を行い Q 値が変化後のタスクに適応するために数多くの経験数が必要となり，一時的に行動数が増加している．しかし，収束速度は提案手法の方が速い．

タスク変化前，変化後の両方に言えることとして，提案手法は行動数収束後の行動数のばらつきを抑える効果がある．タスク変化前・変化後にかぎらず，タスク収束後の行動数のばらつきが通常の強化学習に比べ少ない．この原因は，提案手法の未経験状態への迷い込みの少なさにある．

迷い込みの例を図 5.11 に示す．エージェントは通常の強化学習で taskA を行なっている．この例ではタスク変化前であるとする．このとき，ハッチされている領域を経験済の領域，その他の領域を未経験領域と考える．エージェントが一旦未経験の領域に迷い込んだ場合，その領域の Q 値は全て初期値のままである．そのため，その領域での行動選択はランダムになる．ランダム行動を続けた結果，どんどん未経験領域に迷い込むことが考えられる．こ

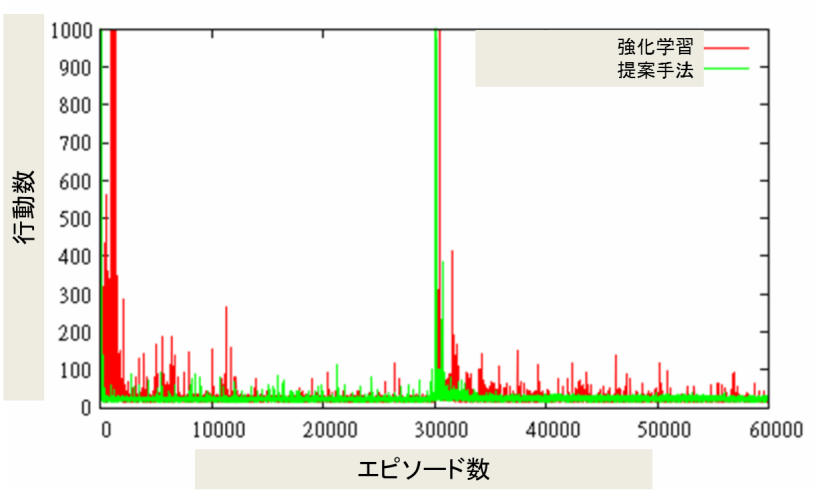


図 5.10: 各エピソードでの総行動回数

のような状況は，タスク変化後でも起こりうる．この場合の迷い込みは新しいタスクに対して再学習が行われていない領域に迷い込むということである．この領域では，再学習するまでは過去の Q 値の影響を受けて過去のタスクでの最適行動をとってしまう．その結果，エージェント目標状態へ到達するのが難しくなる．このような迷い込み現象が通常の強化学習では起こりやすい．特に今回の実験環境のような壁のないオープンスペースに特有の問題であるといえる．

提案手法の場合は，重要ではないセンサの Q 値を重要センサに集約することができる．このため，ロボットは未経験状態および再学習が行われていない状態でも他の学習済みの Q 値を参照して行動を行うことができる．その結果，迷い込み状態を脱出しやすい．この作用が，行動数のばらつきが少ないことにつながっていると考えられる．

以上より，提案手法がタスクに応じて重要センサを選択できていることと，従来の強化学習よりも収束速度およびその後の安定性において優れていることが確認された．これらの結果から提案手法の有効性が確認された．

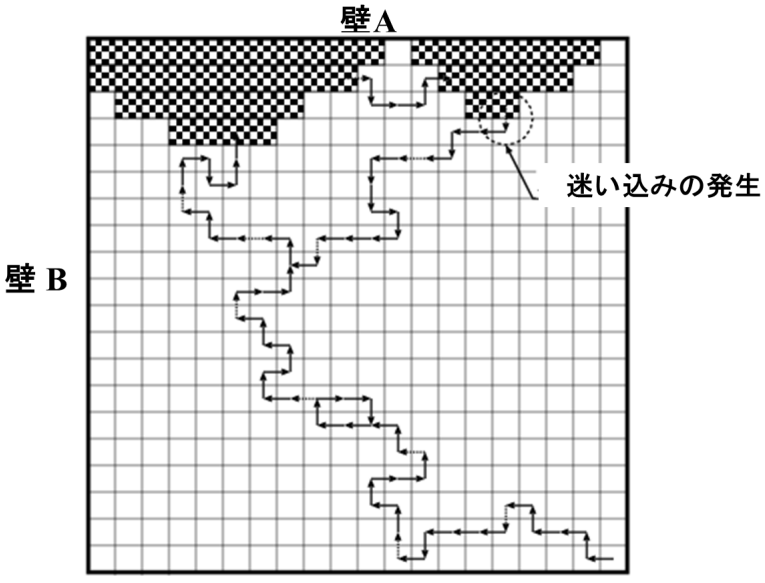
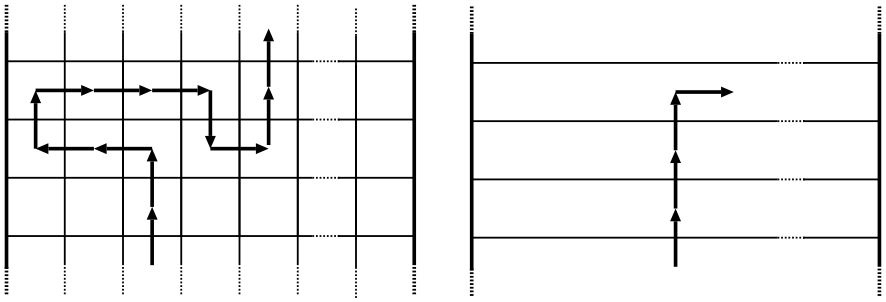


図 5.11: 未経験状態への迷い込みの例



(a) $\rho_{1,t} < \rho$ の時の状態認識と行動　(b) $\rho_{1,t} \geq \rho$ の時の状態認識と行動

図 5.12: 提案手法を適用したエージェントの学習の流れ

5.7　実ロボットを用いた提案手法の有効性の確認

5.7.1　実験目的

次に実ロボットにおける提案システムの有効性の検証を行う．実環境では，センサ情報にノイズが乗る．そのため，相関係数が適切に算出されず，適切なセンサの取捨選択が行われない可能性がある．そこで，本実験において実環境下でも提案システムが正常に動作することを確認する．

5.8　実験に使用するロボットおよび実験環境

5.8.1　本実験で使用するロボット

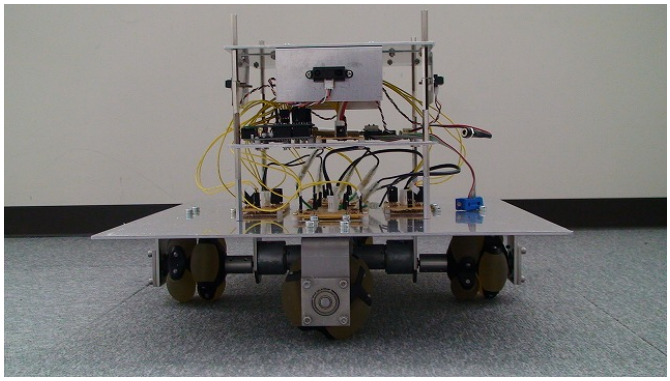
本実験で使用するロボットについて説明する．図 5.13 に今回使用するロボットを示す．寸法は，高さ 262(mm)，幅 400(mm)，奥行き 400(mm) で，正方形の形状をしたロボットである．本ロボットは，4 つのオムニホイールが搭載されており，様々な方向に行動することができる．また，各辺に赤外線センサが搭載されており，前後左右の障害物との距離を測定することができる．本ロボットの技術的な解説は付録に記載している．本実験で使用する赤外線センサは壁 A との距離を計測するセンサと壁 B との距離を計測するセンサの 2 つのセンサのみである．

赤外線センサは障害物との距離を電圧の形で出力する．赤外線センサの電圧-距離特性を図 5.14 に示す．この電圧-距離特性はメーカーのデータシートから抜粋したものである．本特性図より，距離 0cm から 5cm 強未満の範囲とそれ以降の 80cm までの範囲において，取りうる電圧が重複する．そのため，赤外線センサが示す電圧信号から現在の距離を一意に特定することができない．そこで，本論文で行う実ロボットを用いた実験においては 5cm 強未満の領域はロボットが測定不可能な領域とする．加えて，センサの設置場所をロボットの 3 段目に取り付ける．それにより，1 階層目より 10cm 奥にセンサを取り付けることができる．実際に距離測定が必要なのは 10cm 以降の距離なので，このようにすることで 5cm 強未満の領域がハードウェア的に測定不要な領域になる．従って，距離を出力電圧から一意に特定することができる．

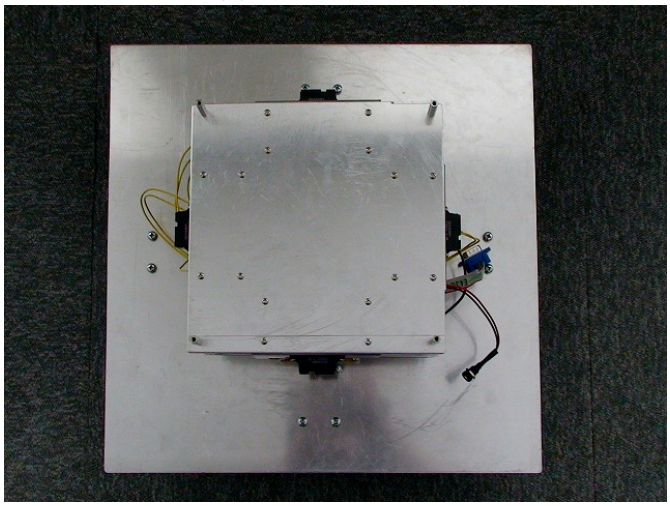
本研究では図 5.14 を表す近似式を求めた．近似式を式 (5.10) に示す．

$$V(d) = \frac{32}{d+4} \tag{5.10}$$

V はセンサの出力電圧であり，d が距離である．式 (5.10) を図示したものが図 5.15 である．本研究で用いるロボットはこの近似式に基づいて距離を決定する．



(a) ロボットの正面



(b) ロボットの上面

図 5.13: 実験に使用するロボット

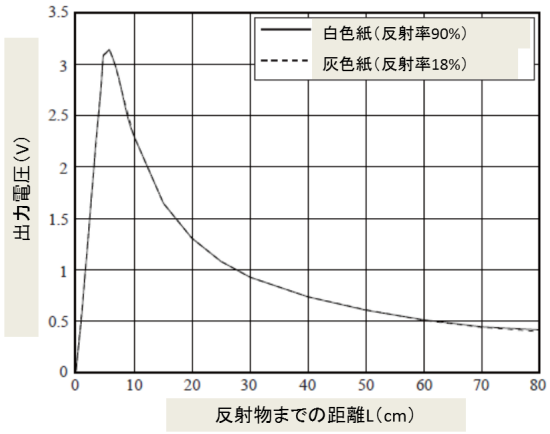


図 5.14: 赤外線センサの電圧-距離の特性関係

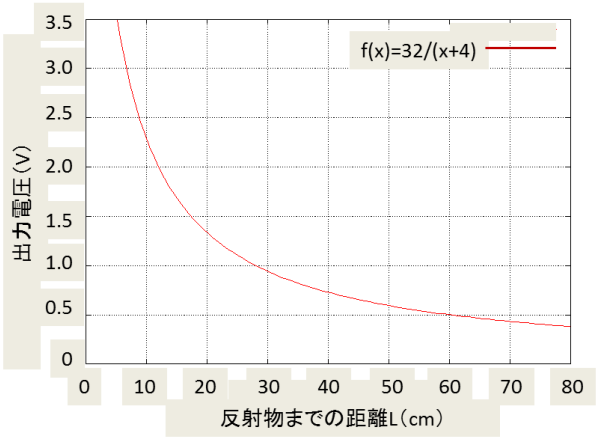


図 5.15: 電圧-距離特性の近似式

5.8.2 実験環境

実機実験環境はシミュレーションと同じ四方を壁に囲まれたオープンスペースを用いる．実機実験環境を図 5.16 に示す．この図の上側の壁を壁 A，左側の壁を壁 B とする．本実験環境では，四方の壁をダウ化工株式会社製「スタイロフォーム IB」を用いて作成した．厚さ 20mm のスタイロフォーム IB を高さ 300mm，幅 1100mm に加工し，これを壁の 1 辺分

とする．この壁を 4 枚作り，それらを正方形状に配置することで実験環境を構築する．各壁の固定には，蝶番とネジを使用した（図 5.17）．

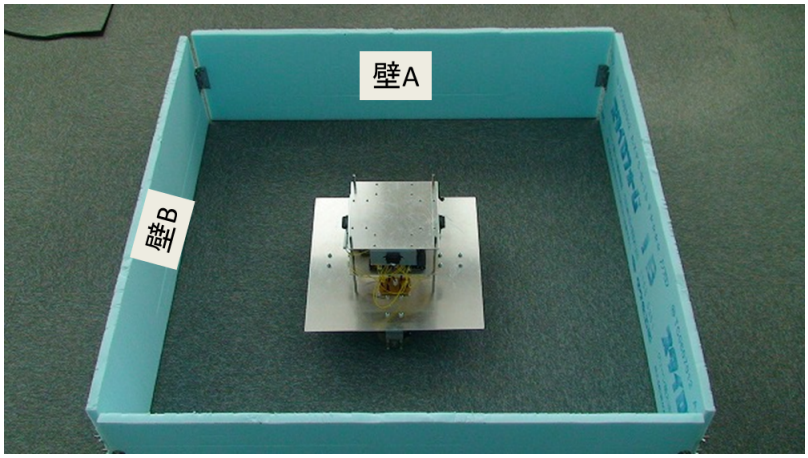


図 5.16: ロボットがタスクを実行する環境

(a) 外側の連結　　(b) 内側の連結

図 5.17: スタイロフォーム IB を使用した壁とその連結

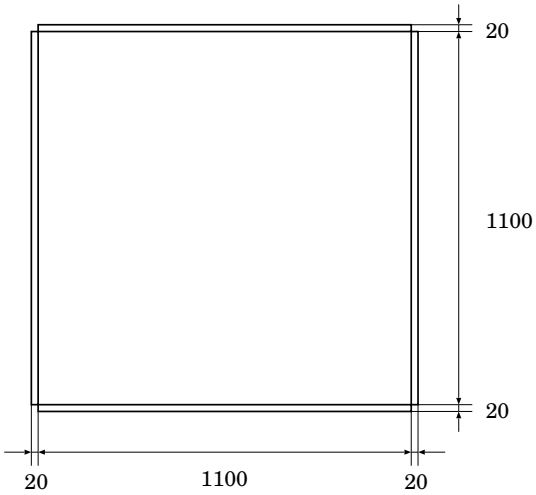
次に，この環境とロボットの大きさの関係を図 5.18 に示す．ロボットが移動可能な範囲は 1100mm×1100mm の空間内である．赤外線センサの搭載位置が高さ 234mm の位置であるのに対して，壁の高さは 300mm である．そのため，赤外線センサは壁を十分に認識することができる．このような環境でロボットはタスクを行う．

5.8.3　実験設定

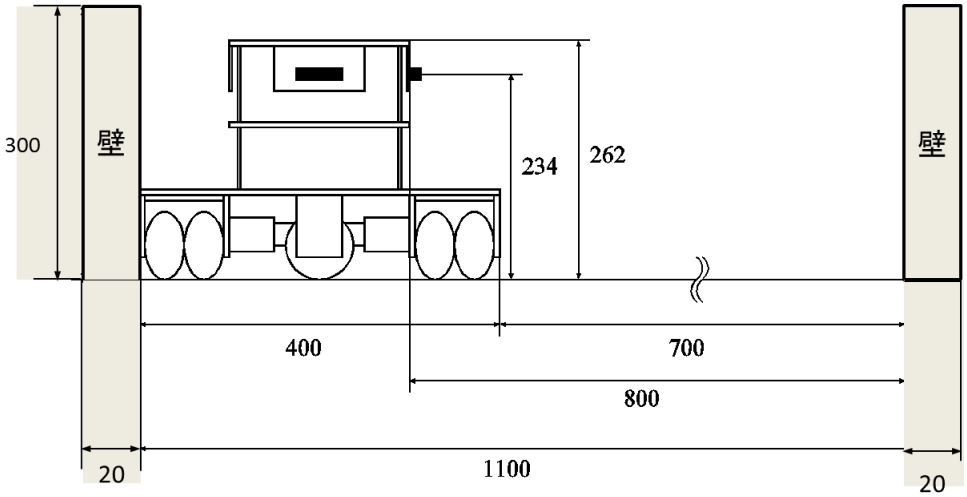
ロボットの状態認識

提案システムおよび比較対象である通常の強化学習におけるロボットの状態認識に関して述べる．本実験環境におけるロボットの 1 センサあたりの状態は図 5.19 のように設定する．ロボットは 70mm 間隔で等しく状態を認識する．この時，離散化した距離の区間には，壁から順に自然数を昇順に割り振る．この状態は 1 つのセンサあたり 10 状態である．この自然数を状態変数 u，状態変数の集合を U とする．また，ある距離を d とその集合を D とする．この時，状態変数 u は，ある距離 d から写像 g' によってマッピングされる．

$$\begin{cases} g' : D \mapsto U \\ g'(d) = \{u = i | (i-1) \times 70 \leq d < i \times 70, i \in N\} \end{cases} \tag{5.11}$$



(a) 上面から見た環境の寸法



(b) 側面から見た環境とロボットの寸法

図 5.18: 環境とロボットの大きさの関係

図 5.19 では横向きの例であるが，紙面鉛直向きの方向の状態認識も同様に行われる．しかし，ノイズにより壁との距離が 700mm 以上と測定され，設定した領域外の状態が出てしまう可能性がある．そのような状態になると，その状態に対応する学習空間が存在しないため，行動選択ができない．そこで，実際の状態数を各センサごと距離が 700mm となる場合を考慮し 1 状態増やし，1 センサ 11 状態として扱う．これにより領域外の状態が出ないようにする．

ロボットの行動

ロボットの行動は，前後左右への移動および停止が存在する．実際には行動はモーターに回転方向（順・逆）とモーターの駆動時間を与えることで行なっている．この時，1 回あたりのモーターの駆動時間は 70mm 移動できる程度である．この移動量は，現状態から次状態に遷移する分に相当する．ただし壁に隣接する位置に相当する状態にロボットが存在し，ロボットが壁方向へ移動するときは行動を実行しない．すなわち，モーターを回転させない．しかし，停止とは異なり，強化学習上では移動したと判断する．このとき，ロボットは壁方向に移動したが，壁に阻まれその状態に留まっているという状態と判断する．よって，価値関数の更新は停止行動ではなく移動行動に対して行う．

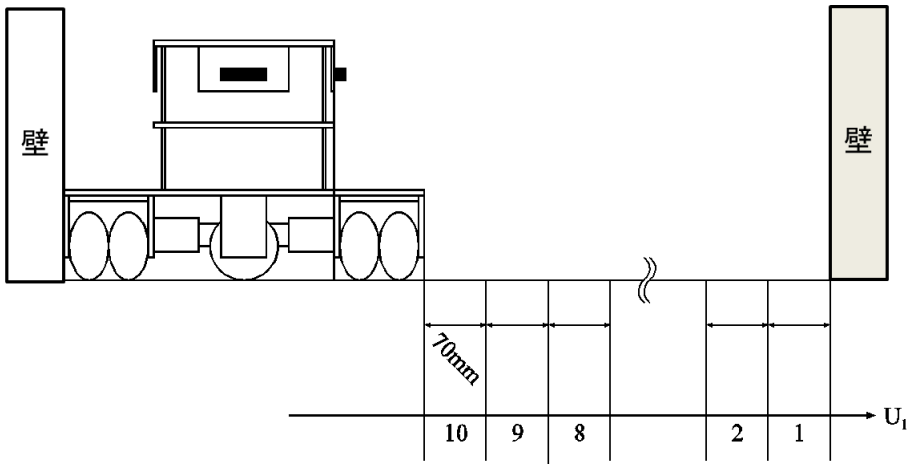


図 5.19: ロボットの状態認識の設定

相関係数算出のための平均報酬算出手法および学習手法

本実験では，相関係数算出のための平均報酬の算出方法に標本平均化手法を用いる．また，強化学習モジュールでは，行動選択手法に ϵ-greedy 法，行動評価手法に Q 学習を用いる．

5.8.4 ロボットのタスク

ロボットのタスクは図 5.2 のように，壁 A および壁 B から最も遠い位置である環境右下の位置から，ロボットの正面の壁である壁 A に到達することである．この時，ロボットの報酬は，ロボットの状態値を基に決定する．報酬 r は式 (5.12) で定義する．

$$r = 20 - u \tag{5.12}$$

u はロボットの状態値である．本実験の場合は壁 A 側から 1〜11 の値をとる．つまり，壁 A の直近の状態値では，ロボットは最大報酬である 19 を受け取り，壁 A から最も遠い状態では報酬 9 を受け取る．

タスク 1 試行の終了条件は 1000 回の行動選択を行った時点で終了とする．実験はこのタスクを 30 試行した時点で終了となる．

5.8.5 実験パラメータ

実験パラメータを表 5.2 に示す．

5.8.6 実験結果

はじめに，提案手法の相関係数の推移から，タスクに対して重要なセンサを選択していることを確認する．本実験では壁 A との距離を計測するセンサ (sensor01)，壁 B との距離を計測するセンサ (sensor02) としている．このとき，各試行の最終行動終了時点での各センサの相関係数の推移を示すグラフを図 5.20 に示す．このグラフより，壁 A との距離を計測するセンサと報酬の相関は全試行において，0.9 付近の値を示している．対して，壁 B との距離を計測するセンサと報酬の相関は，4 試行目までは負の相関が高くなる方に推移していたが，それ以降は相関係数 0 に向かって推移している．本実験において重要センサを判断する閾値は 0.8 である．そのため，このグラフより壁 A との距離を計測するセンサが重要である

表 5.2: 実験設定

項目	内容
1 試行の終了条件	1000 回の行動選択
総試行回数	30
選択可能な行動群 A	{ 前方移動，右方移動，後方移動，左方移動, 静止 }
遷移可能な状態数 $\|S\|$	11×11
行動選択	$\epsilon-$greedy
探査的な行動を選択する確率 ϵ	0.20
行動評価	Q 学習
ステップサイズ・パラメータ α	0.5
割引率 γ	0.5
行動価値 Q の初期値	0.0
相関係数の閾値 ρ	0.8

と判断され，壁 B との距離を計測センサは重要ではないと判断されている．これは，タスクの報酬が壁 A との距離に依存することからも妥当であると考えられる．このことから，提案手法がタスクに重要なセンサを適切に判断しているといえる．

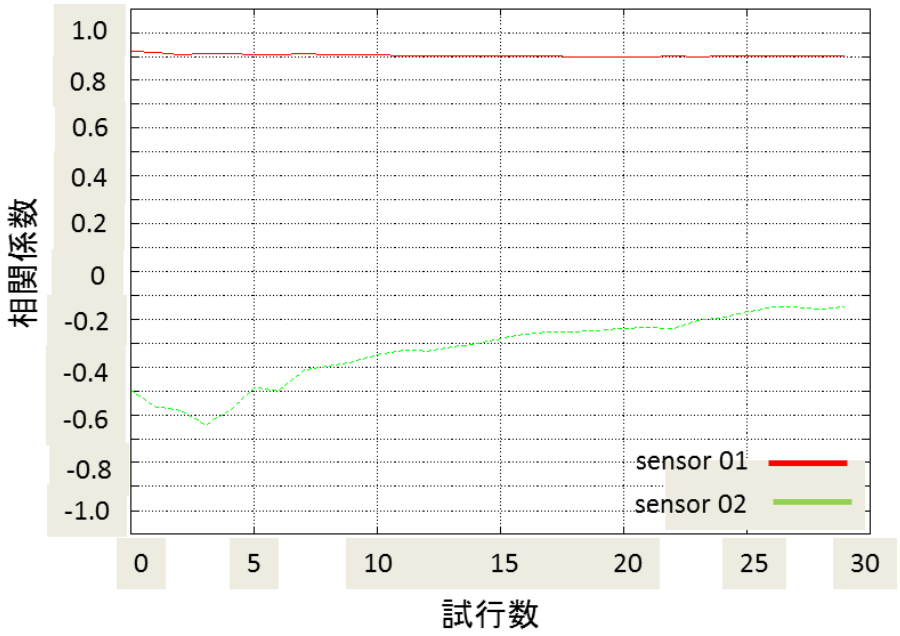


図 5.20: 試行回数における最終行動終了時点での各センサの相関係数の推移

また，学習序盤，中盤，終盤の各行動における相関係数の推移を観察するために，1 試行目，15 試行目，30 試行目の推移を示すグラフを，図 5.21-図 5.23 に示す．1 試行目は相関関係に関する情報がないので，相関係数のバラつきがあるが，15，30 試行目では，相関係数のバラ付きがなく安定している．

次に，提案手法と通常の強化学習の比較を行うことで，提案手法の有効性を確認する．タスクの終了条件はある一定の行動回数であるため，タスク終了時まで獲得した報酬を比較する．1 試行で獲得した報酬が多いほど学習がスムーズに行われたことを意味する．そこで，1 試行の間にそれぞれ獲得した報酬値の累計を算出した．

試行回数に対する両エージェントの累計報酬の推移を示すグラフを図 5.24 に示す．また，

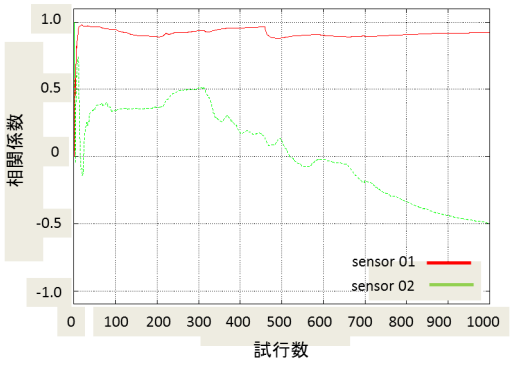


図 5.21: 1 試行目の相関係数の推移

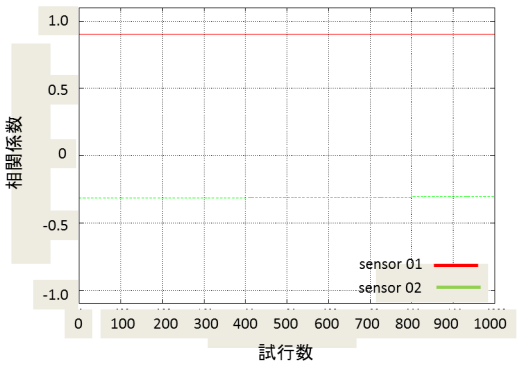


図 5.22: 15 試行目の相関係数の推移

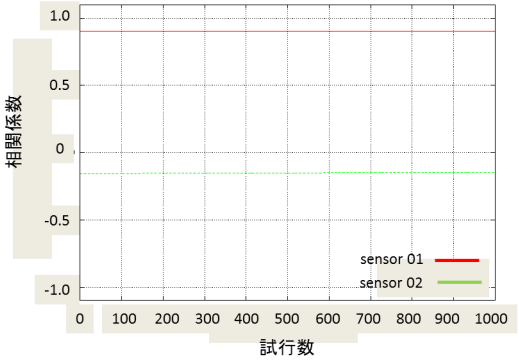


図 5.23: 30 試行目の相関係数の推移

30 試行目までの両エージェントの累計報酬の推移を示すグラフを図 5.25 に示す．

図 5.24 より，提案手法を適用したロボットが獲得した累計報酬は，一般的な強化学習を適用したロボットのものよりも，多くの試行において上回っていることが分かる．さらに，通常の強化学習を適用したロボットが獲得した総獲得報酬は各試行において安定しないのに対して，提案手法を適用したロボットの場合は安定している．その結果，図 5.25 のように，累計報酬に差が出ている．したがって，提案手法を適用したエージェントの方が効果的に学習を行なっていると言える．

5.8.7 考察

通常の強化学習に比べ，提案手法は各試行における獲得報酬のばらつきが少ない．これは，シミュレーション実験においても述べたが，提案手法によってロボットの迷い込みを軽減することができたことを意味する．通常の強化学習を適用したロボットは，目標状態に一度到達してその近傍の状態で学習を行うようになる．しかし，行動選択手法である ϵ-greedy 法の確率 ϵ によりランダム行動を行った結果，未探索な状態群へ迷い込むことがある．

通常の強化学習を適用したロボットの場合，目標状態に一度到達してその近傍の状態で行動を行う．このときに，未探索な状態群へ迷い込んだと考えられる．このことは，特に 15 試行目と 24 試行目における累計報酬の大きな下がり方として現れている．

また，実機特有の問題としてセンサのノイズによる状態の誤認識がある．センサのノイズにより，本来来の状態 s_t ではなく，それとは別の何らかの状態 s'_t を誤って認識をしてしまう．それにより，不適切な意思決定を行い，学習が適切に行われないことがある．提案手法では，通常の強化学習とは異なり，不要状態の価値関数を集約することができる．そのため，

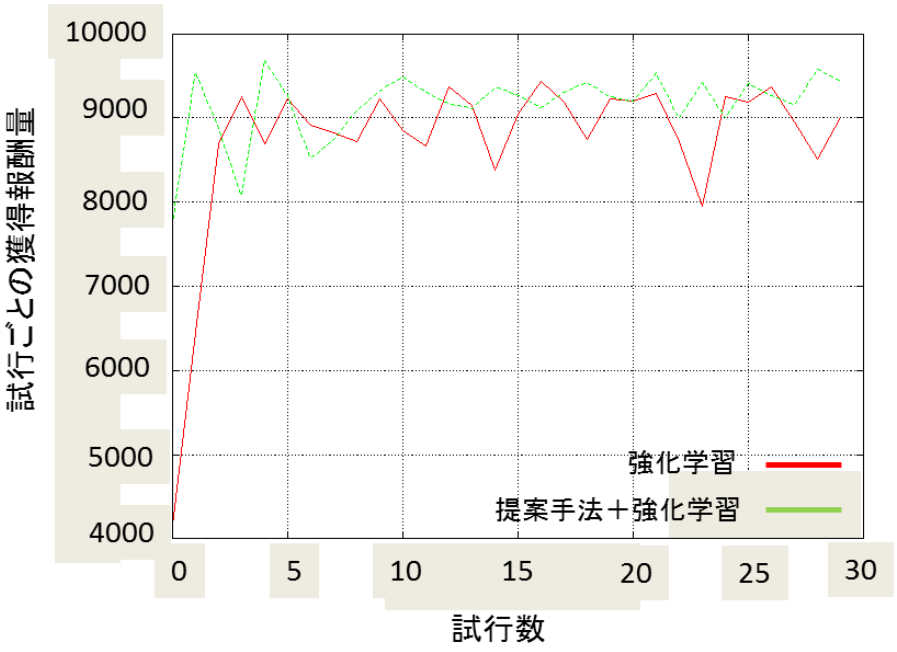


図 5.24: 各試行試における総獲得報酬の推移

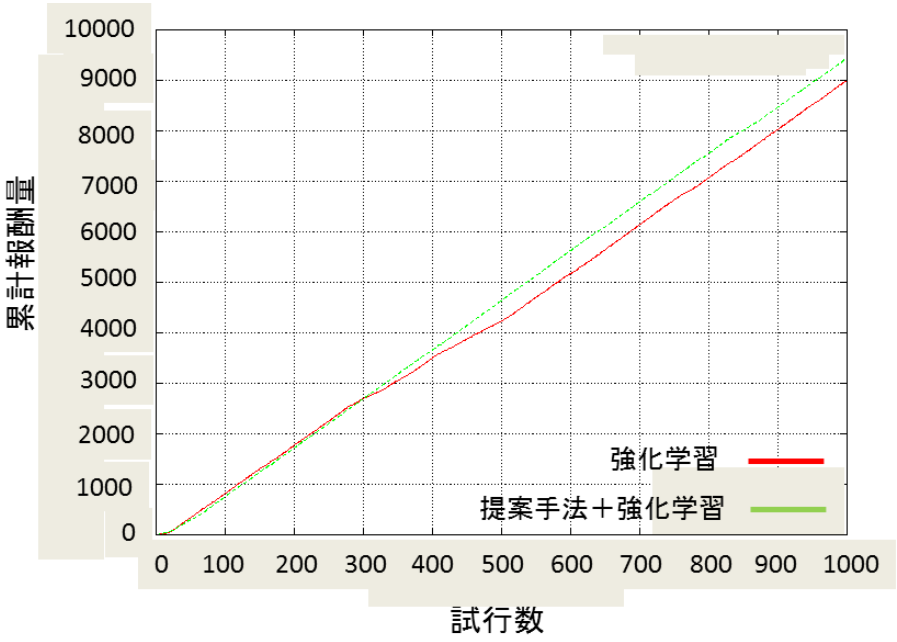


図 5.25: 30 試行目までの両エージェントの累計報酬の推移 $\sum_{t=1}^{t=1000} r_t$

誤認識の影響に関して不要状態の部分に関しては考える必要がない．そのため，通常の強化学習に比べ状態の誤認識の影響量は少ないと考えられる．

以上のことから，提案手法は学習を効率化することができ，有用であるといえる．

5.9 まとめ

本章では，タスクに応じて使用するセンサを取捨選択し，選択したセンサのみで学習空間を構築する手法を提案した．ロボットは構築された学習空間を基に行動選択を行うことでより効率的に学習を行うことができる．センサ情報と強化学習における報酬との相関関係に注目し，相関係数をセンサの重要度と考えた．センサの重要度が人間の設定する閾値を超えるとそのセンサがタスク遂行に重要なセンサであると判断する．強化学習における Q 空間を基に，重要センサのみを用いた一時的な Q 空間を構成，ロボットはこの一時的な Q 空間を基に行動選択を行う．これにより，従来の強化学習に比べて多くの情報を用いて行動選択を行うことができるため，効率的な学習が可能となる．この手法を，シミュレーションおよび実ロボットに適用し，その有効性を確認した．

第6章　センサの重要度に応じた状態空間の自律的構成

本章では，タスクに応じたセンサの重要度に注目し，センサの重要度に応じた状態空間の構成を行う．前章までは，タスクの重要度として相関係数を用いて，相関係数がある閾値が超えた場合を重要センサと判断していた．しかし，この手法では重要センサは必要と不要の2値での判断になる．そのため，わずかでも相関係数が閾値を下回っていた場合でも重要センサではないと判断され，利用されなくなる．しかし，閾値の設定が人間に依存していることもあり，本来は閾値を下回っているが考慮すべきセンサである場合がある．このような場合，学習速度に悪影響が出る可能性がある．

そこで，本章では前章のようにセンサを利用する・しないの2値で判別するのではなく，センサの重要度に応じて各センサの状態数を変更する手法を提案する．この手法により，人間が重要センサの閾値を設定する必要はなくなり，ロボットの経験から自律的にセンサの重要度に応じた状態を構成することが可能になる．以降で，センサの重要度に応じた状態空間の構成の概念について説明する．

6.1　重要度に応じた状態構成システム

システムの概念図およびシステムの流れを図6.1，図6.2に示す．本システムは前章のシステム概念図と大筋では同じである．センサ値と報酬値を知識として収集し，相関係数を算出する．算出した相関係数をセンサの重要度とするところまでは前章のシステムと同じである．

前章のシステムと異なるのは，重要度の利用部である．前章では，強化学習部のQ値空間を基に重要センサのみで構成された一時的なQ値空間を構成する．本章で提案するシステムでも，強化学習部のQ値空間を基に一時的なQ値空間を構成する．しかし，その構成の方法が異なる．重要度を基に各センサ軸の状態数を決定し，その状態数に応じた一時的なQ値空間を構成するという方法を用いる．

ロボットはこの一時的なQ値空間を基に行動を選択し報酬を得る．得た報酬を基に強化学習部にて元のQ値空間を更新する．次節にて，センサの重要度に応じた状態空間構成方法の詳細について述べる．

6.2　重要度に応じた状態空間構成

タスクに対するセンサの重要度の概念は，前章と同様である．すなわち，センサ値とタスクの達成度である報酬との相関関係を基にする．相関関係の算出の方法は，報酬を目的変量，各センサの値を説明変量とし，報酬と個々のセンサ毎の回帰分析による相関係数を基にした方法や重回帰分析による各センサの回帰係数を相関係数として表す．タスクと報酬の相関係数の絶対値が1に近いほど，強い相関関係がある．このような場合，そのセンサがタスク遂行に重要であることを意味している．つまり，重要センサから認識される状態は報酬に直結する情報である．そのため，重要なセンサほど詳細に環境を分析することが求められる．よって，重要度が高いほど状態の分割数を多くする．重要度の低いセンサは獲得報酬にあま

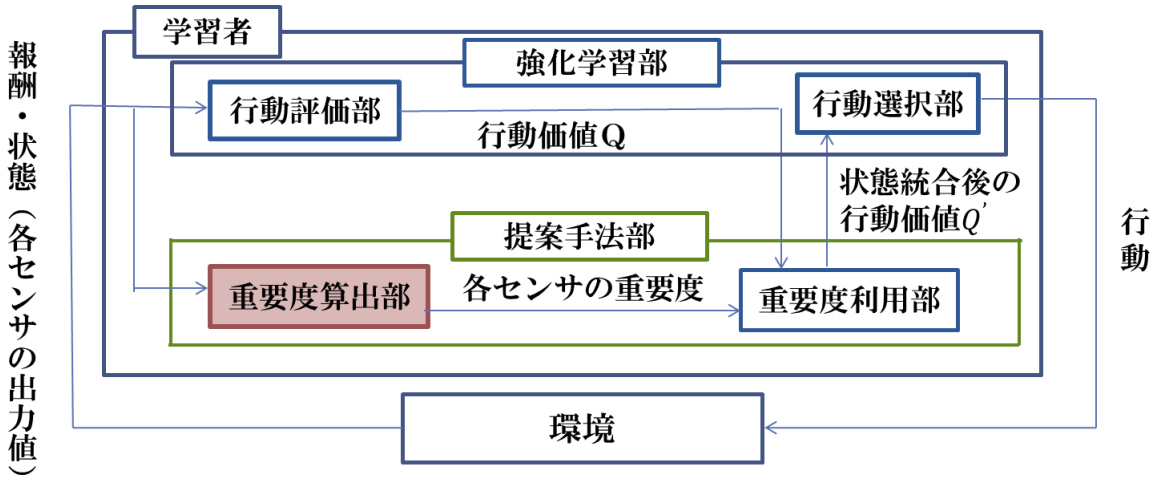


図 6.1: システムの概念図

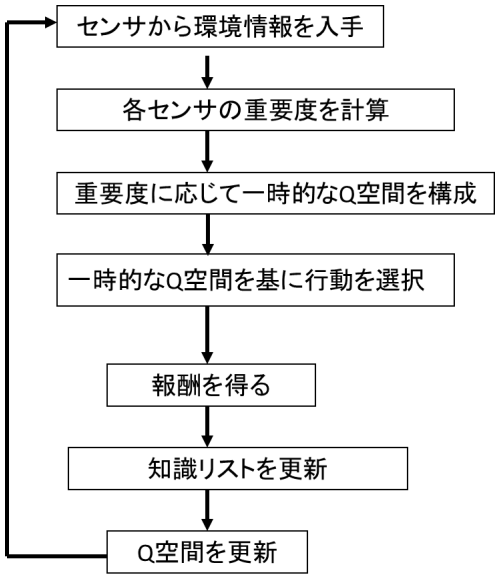


図 6.2: システムの流れ

り影響しないため，詳細に環境を分析する必要はない．よって，状態数を少なくする．このようにすることで，タスクに応じた状態空間を構成し学習することができる．

各センサの状態数の求め方について述べる．センサの重要度は，センサ値と報酬の相関関係を表すことのできる任意の方法を用いる．例えば，前章の式 (5.9)，(5.2) に示したものを用いる．この他にも目的変量を報酬，説明変量を各センサの値とした重回帰式における回帰係数がセンサの重要度を表すことができる．センサの重要度を基に各センサの状態数を決定する．

状態数の決定には単位状態という概念を用いる．単位状態は，センサの分解能に依存した，センサが表現可能な最小の状態を指す．例えば，分解能 1cm の距離センサの場合は 1 つの状態を 1 センチ間隔で構築することが可能である．このように分解能を基準として構築された状態を単位状態とする．単位状態の概念図を図 6.3 に示す．図のようにセンサの各状態は分解能 r 毎に分割されるときの各状態を単位状態とする．この単位状態を重要度に応じて統合することで Q 値空間を構築する．

単位状態の総数はセンサの性能によって異なり，センサの分解能と測定可能レンジによっ

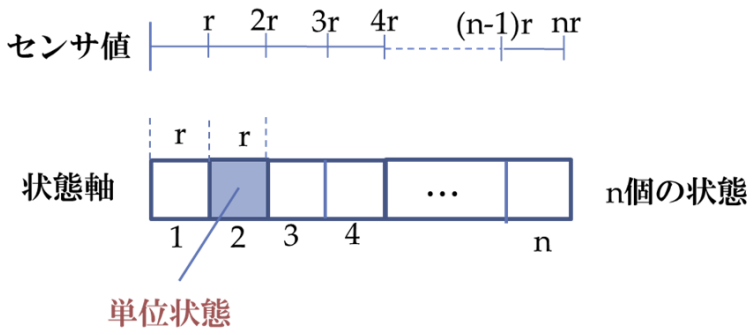


図 6.3: 単位状態の概念図

て決定する．測定可能レンジは最大測定可能距離と最小測定可能距離の差によって求められる．測定可能レンジと分解能の商がそのセンサの単位状態の総数となる．センサ i の分解能を r_i，最大測定可能距離を $g_{max,i}$，最小測定可能距離を $g_{min,i}$ とする．このときセンサ i が表現可能な最大状態数 $v_{max,i}$ を式 (6.1) に示す．

$$v_{max,i} = \frac{g_{max,i} - g_{min,i}}{r_i} \tag{6.1}$$

式 (6.1) によって算出された状態数が，そのセンサが表現可能な最大状態数である．この最大状態数を基にセンサの重要度を用いて実際の状態数を決定する．今回は，状態数を求めるセンサをセンサ i とすると，状態数は図 6.4 に示す特性に従って決定される．重要度が m_α までは最小状態数 $v_{min,i}$，m_α から m_β までは重要度に応じて比例して状態数が増加する．重要度が m_β を超えると状態数は最大の $v_{max,i}$ となる．なお，m_α，m_β および $v_{min,i}$ は設計者が任意で決定する．この特性を式 6.2 に示す．この特性式に従って，重要度から状態数を決定する．

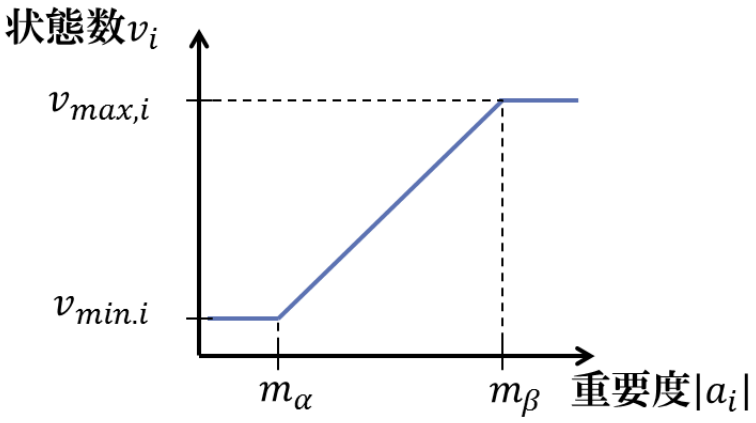


図 6.4: 重要度に応じた状態数の推移

$$v_i = \begin{cases} 1 & (|a_i| < m_\alpha) \\ \lceil \frac{v_i^\star - 1}{m_\beta - m_\alpha} |a_i| + \frac{m_\beta - m_\alpha v_i^\star}{m_\beta - m_\alpha} \rceil & (m_\alpha \leq |a_i| \leq m_\beta) \\ v_i^\star & (m_\beta < |a_i|) \end{cases} \tag{6.2}$$

求めた状態数と強化学習モジュールの Q 値空間を基に一時的な Q 値空間を構成する．Q 値空間の集約例を図 6.5 に示す．この例では，センサ S_1 とセンサ S_2 から状態空間が構成さ

れている．センサ S_1 とセンサ S_2 の最大状態数が 6，センサ S_1 の重要度に応じた状態数が 3，センサ S_2 の重要度に応じた状態数が 2 である．この時，センサ S_1 は状態数を 2 単位状態ずつ統合を行い 3 つの状態として扱う．センサ S_2 は状態数を 3 単位状態ずつ統合し 2 つの状態として扱う．統合される状態の Q 値は平均化される．その結果，図 6.5 のように全 6 つの状態から構成される Q 値空間ができる．

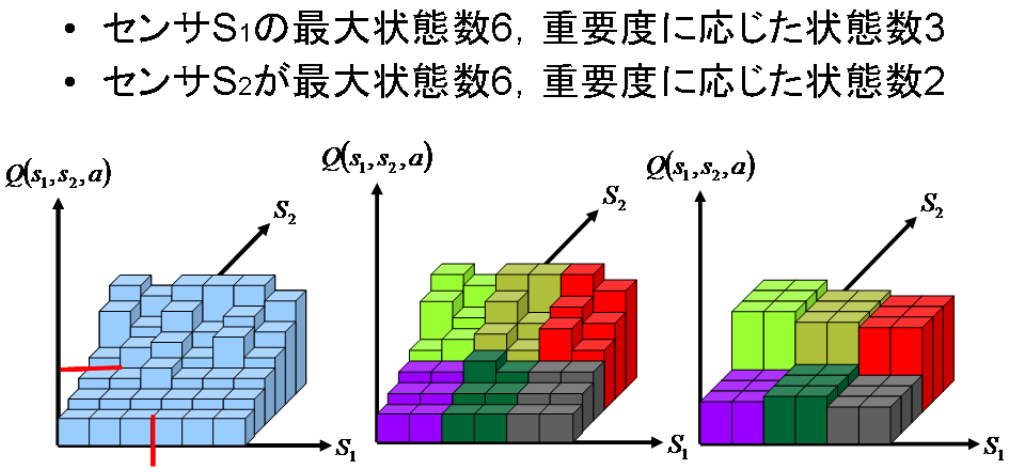


図 6.5: センサの重要度に応じた状態の構成例

この例では，各センサ軸の状態数は割り切れているが，綺麗に分割することができない場合がある．そのときは，分割しきれなかった余りの状態を 1 つの状態として考える．例を図 6.6 に示す．この例では，単位状態数が 9 の状態を 5 つの状態に分ける場合を示している．このとき，1 つの状態は 2 単位状態で構成される．しかし，1 単位状態余りが出てしまう．本手法では，この余りの状態を 1 状態として扱うことにする．

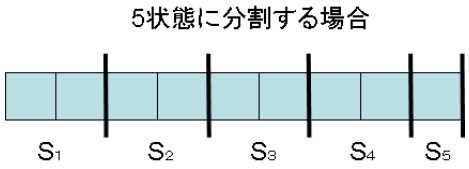


図 6.6: 余り状態の取り扱い例

行動選択に用いる一時的な Q 値空間は，ロボットの現状態に対してのみ作成する．そのためには，式 (6.2) で算出した総状態数を基に 1 つの状態あたりの単位状態数を求める．センサ i の 1 状態あたりの状態数を $R_{m,i}$ とすると，$R_{m,i}$ は式 (6.3) で定義する．この時，状態数は自然数である必要があるため，分解能 $R_{m,i}$ の小数点以下は切り捨てる．ただし，例外として $R_{m,i}$ が 1 未満の場合は 1 状態あたりの単位状態数は 1 として扱う．

$$R_{m,i} = \frac{v_{max,i} - v_{min,i}}{v_i} \tag{6.3}$$

$R_{m,i}$ に従い単位状態を併合し 1 つの状態として扱う．本手法では，ロボットが持つ Q 空間を直接改変することはせず，一時的な Q 空間を作る．それに対して意思決定を行う．評価はロボットが基から持つ Q 空間に対して行うようにする．このようにすることで，Q 空間の改変による学習への悪影響を軽減することができる．

一時的な Q 値空間の Q 値は併合の対象となる状態の Q 値に対して状態行動対の経験数に応じた加重平均によって構成する．つまり，状態行動対の経験数が多いほどその Q 値の重みも大きくなる．

センサ i における，併合対象となるセンサ状態の集合は $C_i = \{v_{i,o}, v_{i,p}, \ldots, v_{i,q}\}$ となる．この時，集合内の v は各センサの単位状態を示す．同様に，センサ t の対象集合を $C_t = \{v_{t,o}, v_{t,p}, \ldots, v_{t,q}\}$，センサ u の対象集合を $C_u = \{v_{u,o}, v_{u,p}, \ldots, v_{u,q}\}$ とする．この時，併合後の一時的な Q 空間の対象状態の Q 値 $Q(s_m, i)$ は式 (6.4) で定義する．ただし，$s_w = \{v_{i,w}, v_{t,w}, \ldots, v_{u,w}\}$ である．$E(s_w, a)$ は状態 s_w で行動 a を選択した回数である．式 (6.5) は併合対象状態の推定獲得報酬の総和である．式 (6.6) は対象の単位状態での総獲得報酬である．同様に式 (6.7) は併合状態における行動 a の総経験回数である．

$$Q(s_m, a) = \frac{R(s_m, a)}{E(s_m, a)} \tag{6.4}$$

$$R(s_m, a) = \sum_{v_{i,w} \in C_i} \sum_{v_{t,w} \in C_t} \cdots \sum_{v_{u,w} \in C_i} R(s_w, a) \tag{6.5}$$

$$R(s_w, a) = Q(s_w, a) \cdot E(s_w, a) \tag{6.6}$$

$$E(s_m, a) = \sum_{v_{i,w} \in C_i} \sum_{v_{t,w} \in C_t} \cdots \sum_{v_{u,w} \in C_i} E(s_w, a) \tag{6.7}$$

6.3 シミュレーションによる検証

6.3.1 実験目的

シミュレーション実験により，提案手法の有効性の確認を行う．実験目的を以下に示す．

- タスクに応じて適切に状態数を調節することができていることを確認する
- 通常の強化学習に比べて学習の収束性が高いことを確認する

有効性の確認は通常の強化学習との比較で行う．比較項目は各エピソード終了時点での行動数とする．

6.3.2 実験環境

実験環境および，エージェントの設定を図 6.7 に示す．四方を壁に壁に囲まれたオープングリッド空間を用意する．空間の広さは $u \times u$ マスである．この環境内でロボットはタスクを行う．

6.3.3 エージェント設定

エージェントは上下左右斜めの 8 方向に加え停止の計 9 動作が可能である．エージェントは 1 回の行動で 1 マス進むことができる．また，エージェントは 2 つの距離センサを搭載しており，それぞれ壁 A と壁 B との距離をマス目で計測することができる (図 6.8)．今回エージェントの相関係数の算出方法は，ピアソンの積率相関係数を用いる．ピアソンの積率相関係数の算出式は式**??**ch5-correlation)-式 (5.4) を参照されたい．

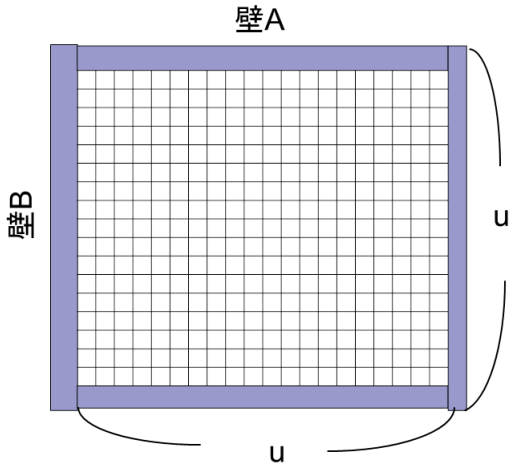


図 6.7: 実験環境

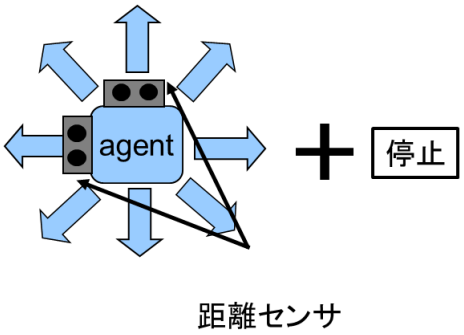


図 6.8: エージェントの設定

タスク設定

今回行うタスクはエピソード型タスクである．エージェントのタスクは主に 2 種類行う．1 つは前章 5.6 で行ったシミュレーション実験で用いたタスクである．5.6 では，スタートから壁 A に近づくタスクと壁 B に近づくタスクを行った．これらのタスクは実験開始から n_ch エピソードまでは，壁 A に近づくタスク，それ以降は壁 B に近づくタスクを行う．報酬設定は，前章の実験の設定を用いる．すなわち，壁 A に近づくタスクの報酬は式 5.7，壁 B に近づくタスクの報酬は式 5.8 で決定する．

１エピソードの終了条件は，エージェントがタスクを達成するか，n_{act} 回行動を行うことである．エピソードが終了するとエージェントはスタートに戻され，次のエピソードが始まる．実験の終了条件は n_e エピソード終了時点である．

パラメータ設定

実験パラメータの設定を表 6.1 に示す．本実験では 15000 エピソードでタスクが変化する．また，最小状態数 1 の閾値は 0.2，最大状態数 20 の閾値は 0.8 となっている．本実験で用いる各センサは同じものなので，センサの最大レンジ $g_{max,i}$・最小レンジ $g_{min,i}$・最大状態数 $v_{max,i}$・最小状態数 $v_{mim,i}$ および解像度 r_i はすべてのセンサで共通とする．

表 6.1: 実験パラメータ

u	20
n_{ep}	40000
n_{ch}	15000
n_{act}	1000
α_{ave}	0.1
m_α	0.2
m_β	0.8
$g_{max,i}$	20
$g_{min,i}$	0
分解能 r_i	1
$v_{max,i}$	20
v_{min_i}	1
ϵ	0.1
α_{RL}	0.1
初期 Q 値	0.0

6.3.4 実験結果・考察

実験結果を図 6.9-図 6.11 に示す．はじめに，各エピソード終了時点での相関係数の推移とそれに伴う各センサの状態数の推移について見ていく．図 6.9 は各エピソード終了時の各センサの相関係数の推移である．また図 6.10 は各エピソード終了時の各センサの状態数の推移である．15000 エピソードでタスクが変化する．タスク変化前では，壁 A センサの相関係数が高く，壁 B センサの相関係数は低い．それに応じて，状態数も壁 A センサの状態数は最大の 20，壁 B センサの状態数は最小の 1 となっている．このタスクの報酬設定は壁 A との距離のみに依存するため，壁 A センサの状態数が最大となり，壁 B センサの状態数が最小となるのは理想的な結果であるといえる．15000 エピソード付近でのタスク変化の直後のエピソードでは，壁 A センサの相関係数は大きく下がり，反対に壁 B センサの相関係数が大きく上昇する．壁 A センサの相関係数は，15000 エピソードから 1000 エピソード程度は相関係数の変化が少なくなっている．これは，相関係数の算出にタスク変化前の知識が影響していることが原因である．タスク変化後に適応した相関係数を算出するためには，相関係数算出の為の知識としての各状態での平均報酬を更新する必要がある．しかしタスク変化後 1000 エピソードに渡り，エージェントが再経験し知識更新を行なっている状態数が，適切な相関係数を算出するのに十分ではないため，相関係数が 0.5 付近の値を示している．知識の更新が十分に行われると相関係数は適切な値になっていき，最終的には，両センサの相関係数はタスク変化前と逆の関係になる．すなわち，壁 B センサの相関係数が 1 付近になり，壁 A センサの相関係数が 0 付近になる．それに応じて状態数の推移も壁 B センサが最大数の 20，壁 A センサの状態数が最小の 1 となる．タスク変化後の，報酬は壁 B との距離のみに依存するため，このような状態数となるのは理想的である．以上から，タスクに応じた状態空間の構成は適切に行われている．

次に各エピソードでの提案手法と通常の強化学習の行動数の推移における比較を図 6.11 に示す．タスクの変化に関わらず提案手法の収束速度が高い．提案手法では，複数の単位状態を統合するため，エージェントは通常の強化学習に比べて多くの経験を基に構築された Q 値を参照して行動することができる．そのため，適切な行動を高速に探索することが可能である．したがって，エピソード毎の行動数が収束しやすい．

タスク変化に対しては，相関係数算出のための知識の更新を行う必要があるため，タスク変化前よりも収束速度は低い．しかし，相関係数がタスクに対して適切に算出されるようになると，状態数がタスクに適したものになるため，学習の収束性が高くなる．通常の強化学習がゴールまでの経路を学習する前に，提案手法は相関係数を算出し状態を構成し，学習を行う．その結果，提案手法は通常の強化学習に比べて学習収束が速いと考えられる．以上のことから，提案手法は有効である．

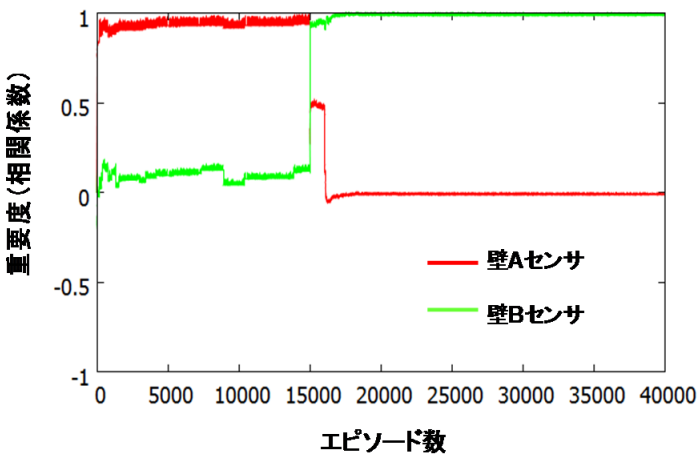


図 6.9: エピソード毎の相関係数の推移

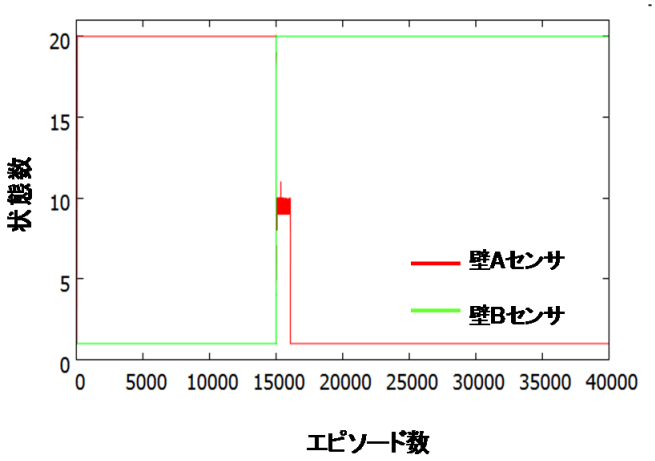


図 6.10: エピソード毎の状態数の推移

6.4 実機実験による提案手法の有効性の確認

6.4.1 実験目的

実ロボットを用いた実験により，提案手法の有効性の確認を行う．実ロボットでは，シミュレーションとは異なり，センサのノイズやセンサ値の取得ミスが起こる．その結果，ロボットが現状態を正しく把握できない場合がある．このような状況下においても有用な結果となることを確認する．

実験目的を以下に示す．

- タスクに応じて適切に状態数を調節することができていることを確認する

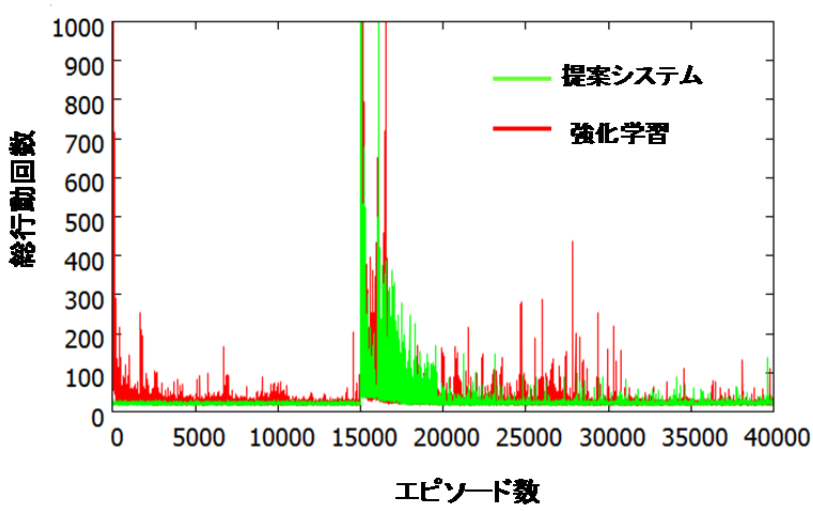


図 6.11: エピソード毎の行動の推移

- 通常の強化学習に比べて学習の収束性が高いことを確認する

有効性の確認は通常の強化学習（すべてのセンサの状態数が常に最大の場合）との比較で行う．

6.4.2 実験環境

実験環境および，エージェントの設定は前章の図 5.16 に示したものを用いる．すなわち，四方を壁で囲まれた 1100mm ×1100mm のオープンスペースを用いる．壁はスタイロフォームにより構成している．ロボットはこの中でタスクを行う．

6.4.3 ロボットの設定

今回用いるロボットは 5.7 の実機実験にて用いたロボットを用いる．ロボットが行う行動は 5.7 とは異なり，上下左右，停止に加えて斜め移動を含む．移動距離は 70mm である．斜め移動をする際には，斜めに直接移動せず一旦横方向に移動した後に縦方向へ移動する．これは，斜め方向へ直接移動すると，車輪の空転によりロボットの車体が斜め方向を向くことが多い．その結果，各センサが正しく壁との距離を認識することが不可能になるためである．

また，ロボットの状態認識は図 6.12 に示すように，スタート地点から 70mm 毎に状態を分割し，計 11 状態認識する．分割した状態ごとに 0〜10 まで状態値が割り振られており，ロボットはこの状態値を基に学習を行う．

本手法では，相関係数の算出方法として，重回帰式を用いた手法を選択した．目的変量として各状態における報酬，説明変量として各センサの値をとる．このとき重回帰式は式 (6.8) で表される．$a_{t,1}, a_{t,2}, \ldots a_{t,j}$ は回帰係数，$r'_t(t,s)$ は状態 s における平均報酬である．a_0 は定数である．

$$r'_t(s_i) = a_1 e_{i,1} + a_2 e_{i,2} + \cdots + a_i e_{i,n} + a_0 \tag{6.8}$$

式 6.8 における回帰係数が各センサ値が報酬に与える影響度を表す．すなわち，回帰係数が各センサの重要度となる．回帰係数 $a_{t,1}, a_{t,2}, \ldots a_{t,j}$ および定数 a_0 は連立方程式 (6.9) を解くことで求められる．ここで，b_{ij} はセンサ i とセンサ j におけるセンサ値の共分散であり式 (6.10) で求められる．式 (6.10) 中の $e_{i,k}$ はセンサ i における知識リスト k 番目の値を示す．b_i^2 はセンサ i のセンサ値分散，$\bar{e_i}$ はセンサ i のセンサ値の平均でありそれぞれ式 (6.11),

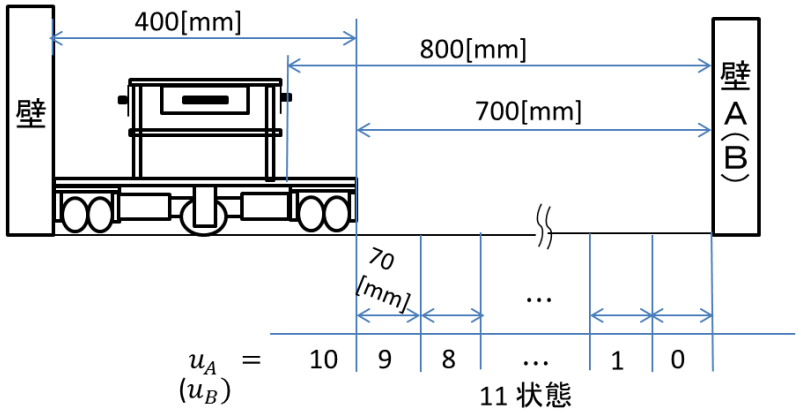


図 6.12: ロボットの状態認識

式 (6.12) で求められる．b_{ir} はセンサ i のセンサ値と報酬 r の共分散であり，式 (6.13) で求められる．また，$\bar{r}$ は報酬の平均の平均であり，式 6.14 で求められる．

$$\begin{cases} a_1 b_1^2 + a_2 b_{12} + \cdots + a_n b_{1n} = b_{1r} \\ a_1 b_{12} + a_2 b_2^2 + \cdots + a_n b_{2n} = b_{2r} \\ \vdots \\ a_1 b_{1n} + a_2 b_{2n} + \cdots + a_n b_n^2 = b_{nr} \end{cases} \tag{6.9}$$

$$b_{ij} = \frac{1}{m} \sum^{m} k = 1(e_{i,k} - \bar{e_i})(e_{j,k} - \bar{e_j}) \tag{6.10}$$

$$b_i^2 = \frac{1}{m} \sum^{m} k = 1(e_i, k - \bar{e_i})^2 \tag{6.11}$$

$$\bar{e_j} = \frac{1}{m} \sum_{k=1}^{m} e_{i,k} \tag{6.12}$$

$$b_{ir} = \frac{1}{m} \sum^{m} k = 1(e_{i,k} - \bar{e_i})(r_k - \bar{r'}) \tag{6.13}$$

$$\bar{r'} = \frac{1}{m} \sum_{k=1}^{m} r'(s_k) \tag{6.14}$$

タスク設定

タスクを図 6.13 に示す．ロボットは右下の角からスタートし，壁 A の方に近づく．報酬は壁 A に近づくほどロボットは高い報酬を得ることができ，ロボットの行動後に式 6.13 で算出する．この計算式ではロボットの現在位置と壁 A との実測値を基にして，11 の状態値に射影した値 d_A を用いる．射影する状態値は，ロボットの状態認識と同様に 70mm ごとに状態が分割されており，0〜10 の値が各状態に割り当てられている．タスクはエピソード型で，1 エピソードはロボットが壁 A に到達した時点で終了する．次のエピソード開始前にロボットはスタート地点に戻される．n_e エピソード行った時点で実験終了とする．

$$r = 11 - d_A \tag{6.15}$$

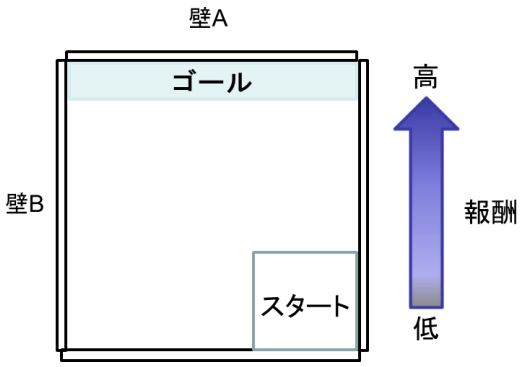


図 6.13: ロボットのタスク

6.4.4 実験パラメータ

実験パラメータを表 6.2 に示す．本実験で用いる各センサは同じものなので，センサの最大レンジ $g_{max,i}$・最小レンジ $g_{min,i}$・最大状態数 $v_{max,i}$・最小状態数 $v_{mim,i}$ および解像度 r_i はすべてのセンサで共通とする．

表 6.2: 実験設定

n_e	100
α_{ave}	0.1
ϵ	0.1
α	0.1
初期 Q 値	0
センサの最大レンジ $g_{max.i}$	1100mm
センサの最小レンジ $g_{min,i}$	0
センサの分解能 b	100mm
$v_{max,i}$	11
v_{min_i}	1
m_α	0.2
m_β	0.8
初期状態数	11
初期重要度	1.0

6.4.5 実験結果・考察

実験結果を図 6.14-6.19 に示す．まず，図 6.14 は各エピソードの終了時点での重要度の変化を示している．第 1 エピソード終了時点からすぐに重要度が収束している．センサ A に関する重要度は 0.8 の閾値を超えており，反対にセンサ B に関する重要度は閾値 0.2 を下回っている．本タスクでは，センサ A のセンサ値のみが報酬に影響を与えている．そのため，センサ A の重要度が高くセンサ B の重要度が低くなるのは妥当である．図 6.15 は各エピソード終了時点での各センサの状態数を示している．重要度の推移と同様に各センサの状態数に関しても第 1 エピソード終了時点から収束している．重要度が 0.8 を超えているセンサ A の状態数に関しては，最大状態数 11 をとっている．重要度が 0.2 を下回っているセンサ B の状態数に関しては，最小状態数 1 をとっている．このことから，エージェントは適切に状態を構築することができているといえる．

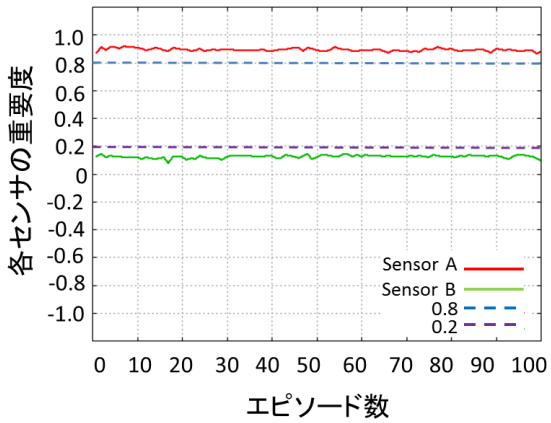


図 6.14: 各エピソード終了時点での各センサの重要度

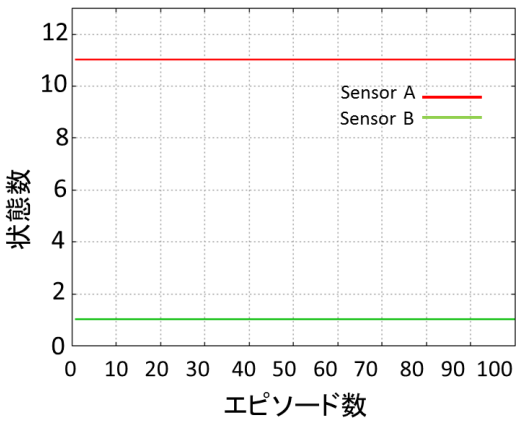


図 6.15: 各エピソード終了時点での各センサの状態数

図 6.16 および図 6.17 はそれぞれ 1 エピソード目における行動ごとの各センサの重要度の推移と状態数の推移である．30 回目の行動まではセンサ A の重要度は安定していない．30 回目の行動以降はセンサ A の重要度が 0.8 を超えるようになる．それに応じて，重要度の推移に応じて状態数も変化している．センサ A ははじめ 1 状態であったが徐々に状態数を増やしていき最終的には最大数 20 となっている．一方，センサ B では 50 回行動まで，センサの重要度が安定していない．行動 30 回目から 50 回目にかけて，重要度 0.3 で安定しているが，行動 60 回目以降は重要度 0.1 程度となる．これに伴い状態数も 2 状態，4 状態，1 状態と推移した後，3 状態になり最終的に 1 状態に収束する．

このようにセンサ A は 30 回目行動までセンサ B は 50 回目行動まで重要度および状態数が安定しない．これは，ロボットは環境内の全ての状態を経験していないため，センサ値と報酬に関する情報が不足していることが原因である．その結果，各センサの重要度計算が適切に行うことができない．ある程度行動を行うと環境内の状態を十分に経験することができるため，適切な重要度および状態数を構築することができる．

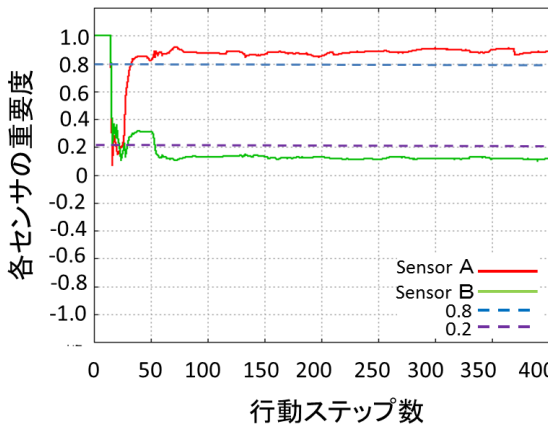


図 6.16: 1 エピソード目の各行動ステップにおける各センサの重要度の推移

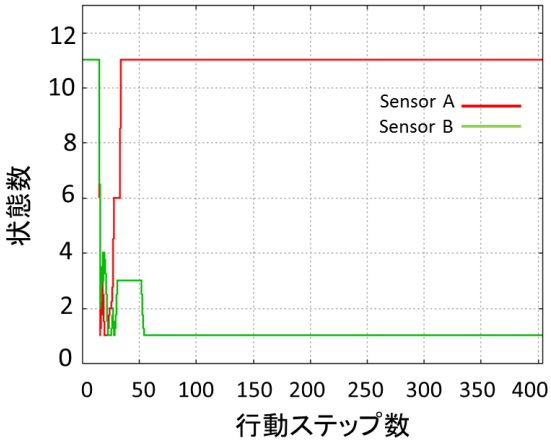


図 6.17: 1 エピソード目の各行動ステップにおける各センサの状態数の推移

図 6.18 は提案手法と通常の強化学習（すべてのセンサを最大の状態数で学習を行った場

合）との比較である．また，図 6.19 は行動数の軸のスケールを [0:240] の範囲にしたものである．提案手法では，通常の強化学習に比べて収束速度が早い．これは，重要度の低いセンサの Q 値を統合することにより，未経験の状態群への迷い込みが軽減されているためである．通常の強化学習では，全てのセンサの状態数が常に最大数となっている．そのため，行動選択の際に現在状態のみの Q 値を基に行動選択を行う．現在状態が未経験の状態である場合，その状態の Q 値は初期値であるため次の行動はランダムに選択される．そして，その行動の結果遷移した状態が再び未経験の状態であった場合，さらに次の行動はランダムに決定される．このように連鎖的にランダム行動を行うことで未経験の状態群に迷い込むことがある．提案手法では，低い重要度のセンサは状態数の削減の際に単位状態を併合する．そのとき，各単位状態の Q 値も平均化される．そのため，現在の状態が未経験であっても，Q 値を併合した結果他の状態が経験済みの場合は Q 値が獲得できる．それにより未経験状態からの脱出が容易である．その結果，目的状態への遷移が容易になり学習の収束速度が上昇している．

以上の結果から，提案手法はタスクに対して適切な Q 値空間を構築し，それにより高速に学習できることを確認した．従って，提案手法は有効であるといえる．

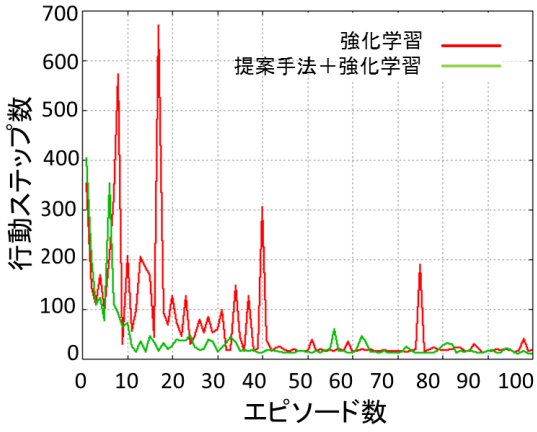


図 6.18: 各エピソードにおける総行動数

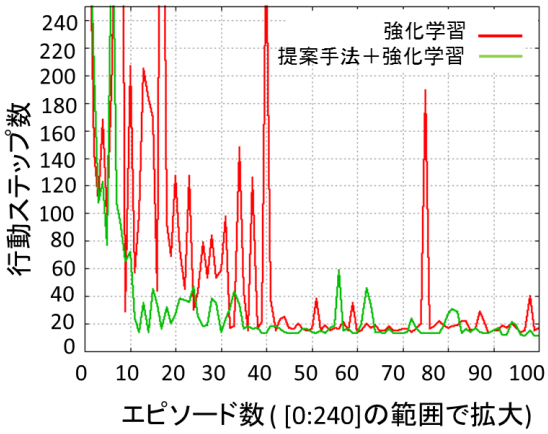


図 6.19: 各エピソードにおける総行動数（拡大）

6.5 まとめ

本章では，センサの重要度に応じて適切な状態空間を構築し，それによりより効率的な意思決定を行う手法を提案した．本提案手法は，センサ値と報酬の相関により重要度を算出する．タスク遂行に重要なセンサはセンシングを細かく行い，そうでないセンサはセンシングを大まかに行う．すなわち，重要度が高いセンサほど状態数を多くし重要度の低いセンサは状態数を少なくする．この概念に基づいて強化学習における Q 値空間から，重要度に応じて一時的な Q 値空間を構成，ロボットはこの一時的な Q 値空間を基に行動選択を行う．これにより，従来の強化学習に比べて多くの情報を用いて行動選択を行うことができるため，効率的な学習が可能となる．この手法を，シミュレーションおよび実ロボットに適用し，その有効性を確認した．

第7章　本論文のまとめ・今後の課題

7.1　本論文のまとめ

本論文では，ロボットに利用されることの多い強化学習に注目し，問題点として学習時間が掛かることを述べた．そのため，学習の効率化について考察した．強化学習は，Q 空間を更新することで学習が進行する．Q 空間はロボット内部の情報であるが，他者とのコミュニケーションによって他者の Q 値を受け取り自身の Q 空間を更新することができる．しかし，無差別にコミュニケーションを行えば，却って学習に悪影響を与える可能性がある．一方で，Q 空間そのものは，ロボットのセンサの数や性能によってその大きさを変える．そのため，高性能な環境認識能力を持つロボットほど Q 空間が大きくなり学習に時間を要する．タスクを遂行するために，全てのセンサが必要とは限らない．タスクを遂行するために必要なセンサ情報を用いて Q 空間を構築することで，Q 空間の大きさを抑え学習を効率化できる．そこで，以下の 2 つのアプローチから学習の効率化について考察し，それぞれに対して情報を取捨選択することでより効率化な学習を実現する手法を考察した．

- ロボット外部の情報の取捨選択：コミュニケーションにおける有益情報の取捨選択
- ロボット内部の情報の取捨選択：タスクに応じた環境情報の取捨選択による Q 空間の構成

第 3 章，第 4 章では，ロボット外部の情報の取捨選択：コミュニケーションにおける有益情報の取捨選択として，コミュニケーション相手の取捨選択を行う手法を考える．まず，第 3 章では，コミュニケーションが個体の学習にどう影響するのかを調査した．それを受け第 4 章では，より高度なコミュニケーションとして，コミュニケーションする相手を取捨選択し，学習を効率化する手法を提案した．コミュニケーションの相手は誰でも良いというわけではない．自身の学習に有益な結果をもたらす情報を持つ他者とコミュニケーションすることが望ましい．提案手法は他者からの情報である Q 値とそれを基に行動した結果である報酬を比較することで，他者を評価する．比較の結果が近いほどその他者がもたらす情報は自身にとって有益であるし，その情報をもたらした他者の評価を上げる．コミュニケーションと他者の評価を繰り返すことで，自身に有益な情報を持つ他者を学習する．最終的に，自身の学習に不要な情報をもたらす他者とコミュニケーションしなくなるため，コミュニケーションによる学習がより高速になる．本手法を迷路問題に適用し，手法の有効性を確認した．

第 5 章，第 6 章では，ロボット内部の情報の取捨選択：タスクに応じた環境情報の取捨選択による Q 空間の構成として，ロボット周囲の環境情報を考え，環境情報を自身の目的に合わせて適応的に変化させることによる学習の高速化に着目した．第 5 章では，タスク遂行に重要なセンサを取捨選択し学習に使用することで，強化学習における状態空間をタスクに応じて構成することが可能となる手法の提案を行った．タスクの遂行に重要なセンサを決定する基準として，センサの計測値（センサ情報）と報酬の相関関係に注目した．ロボットはタスクを遂行しつつ，センサ情報と報酬情報を集め，相関係数を算出する．算出した相関係数が閾値よりも高い場合は，そのセンサがタスクの遂行に重要なものであると判断する．そして，重要なセンサが判断された後に，行動選択にのみ用いる一時的な Q 値空間を構成する．この一時的な Q 値空間は全てのセンサを用いた Q 値空間を基に構成される．一時的な Q 値空間はタスク遂行に不要なセンサ軸が排除され，タスク遂行に重要なセンサのみで構成

される．そのため，不要なセンサ軸の Q 値要素をタスクの遂行に重要な Q 値に射影することで，ロボットはより多くの情報量を持った Q 値を基に行動の選択を行うことができる．その結果，ロボットは効率的に学習を行うことができるため，学習収束が高速化する．このことを，シミュレーションおよび実機実験によって示した．

第 6 章では，第 5 章での状態空間の構成方法を発展させた状態空間構成手法を提案した．第 5 章での手法では，閾値によってセンサを学習に利用する・利用しないの 2 値で決定していた．そのため，相関係数が閾値を少しでも下回るとそのセンサは不要センサと判断される．しかし，閾値をわずかに下回る程度であれば，使用した方が良い場合もある．よって閾値のみで，完全に分けるのは問題がある．より柔軟にタスクに対して適切な Q 値空間を構成するためには，重要度に応じて動的に状態数を決定することが望ましい．相関係数が高いほどそのセンサのタスク遂行における重要度は高い．そこで，本手法では相関係数に応じて学習に用いるセンサの状態数を動的に決定する．センサの相関係数の絶対値が高いほど，より細かいセンシングがタスク遂行に重要であるため，状態数を多くする．一方，センサの相関係数の絶対値が低いほど状態数を少なくする．その結果，タスクに適した Q 値空間が構築される．構築する Q 値空間は最大の状態数を持つ Q 値空間を基に，各状態の Q 値を統合することで一時的な Q 値空間を構成する．ロボットはこの一時的な Q 値空間を構築し，これを基に意思決定を行う．本手法の有効性をシミュレーションと実機実験により示した．

以上から，ロボット外部と内部のから情報の取捨選択を行うことで強化学習をより効率的に行う手法を提案し実験を通してその有効性を確認した．このことから，本研究のアプローチにより強化学習の効率化が実現した．

7.2 今後の課題

本論文では，ロボットのセンサ情報の学習への利用による効率的な学習について群ロボット間のコミュニケーションおよび個体の学習に関して述べてきた．センサ情報を大きく 2 種類に分けて，それぞれの利用方法について述べた．

第 3 章，第 4 章では，群ロボットにおける個体間の学習では，コミュニケーションを用いた個体の相互発達に関して述べた．しかし，コミュニケーションで扱う情報やコミュニケーションの利用方法などは全て人間が与えている．コミュニケーションによる情報交換自体は，TCP/IP などの通信規格がロボット間で合えばやり取りすることができる．しかし，やり取りされた情報が何に関する情報であるかは，ロボットにとってはわからない．従来はロボットの設計者がその情報の利用方法を規定しているため，ロボットはアルゴリズムに従いコミュニケーションされた情報を利用することができる．人間が行うようなコミュニケーションを考えると，コミュニケーション情報は誰かに規定されるものではなく，自身の現在の状況やコミュニケーションする他者の状況に合わせて動的に変化するものである．ロボットにおいても，タスク・環境に応じて適切にコミュニケーション情報を変化させていくことが重要である．同時に，個々の個体がコミュニケーションされた情報を自身の知能の発達に有効に利用する能力が必要となる．従って，他者からの情報の利用の適切な利用の仕方を自律的に獲得する枠組みを考えることが今後の課題である．これにより，設計者がコミュニケーションの利用方法を決定する必要は無くなるため，タスク・環境によらずコミュニケーション情報を有効に利用するロボットが実現できる．

第 5 章，第 6 章では，個体のセンサから得られる環境情報を学習に利用する手法について提案した．タスクに応じて不要なセンサ情報を特定し切り捨てるというアプローチであった．今後の課題として，まず現行の手法を遅延報酬に対応させる．本論文で提案したものは，即時報酬環境にのみ対応するものである．しかし，実環境では遅延報酬環境は数多く存在するため，遅延報酬間環境に対応する手法の考案を行う．

次に，時系列要因を含めたタスクに適応することが挙げられる．強化学習はマルコフ決定

過程を原則とした手法である．そのため，時系列的な要素がタスクの達成につながる場合は学習が適切に行われない場合がある（例えば，特定の状態を特定の手順で経験する）．このような場合に対して，時系列要因を加味したセンサと報酬の関係を考えることで，このような場合に対して適した状態空間の構成を行うことで時系列要因を含めたタスクに適応することができる．

最終的な目標としては，他者からの情報や周囲の環境情報といった情報に関わらずロボットがそれらの情報を自身に有効に利用できる枠組みの実現である．すなわち，ロボットが入手した情報を自身が最も効果的発達することができるような，情報の利用方法を創発することを実現したい．それは，人間における高度な情報処理能力，すなわち情報の解釈能力の実現である．情報の意味を自分自身で考え，その意味に基づいて最適な行動を行う能力をロボットが獲得することが望ましい．

謝辞

本論文を結ぶにあたり，日頃より懇切なる御指導と御鞭撻を賜りました主指導教官の倉重健太郎先生に深く感謝の意を表します．また，本論文に対し，貴重なご助言を与えて下さった佐賀聡人先生，畑中雅彦先生，本田泰先生，須藤秀紹先生，高氏秀則先生に深く感謝いたします．そして，本研究に関して多大なご協力をいただきました，OB の沼田利伸さん，木村敏久さんに深く感謝いたします．研究報告の場で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諸兄に感謝いたします．最後に，私の大学院進学を了解し，励まし支えてくださった両親に深く感謝いたします．

付録

付録：実験機ロボットの図面

付録として第 5 章の実験で使用したロボットの投影図，組立図および回路図を記載する．投影図，組立図の縮尺は，可能な限り拡大して見易くするため，それぞれ異なっている．

投影図

本研究の実験で使用したロボットを構成する各部品の投影図を記載する．ここで，以下に載せる投影図は，第三角法に従って製図したものである．また，正面図の上下もしくは左右が対称である場合は，慣例に従って，右側面図や上面図の記載は割愛する．

はじめに，ロボットの第 1 階層の投影図を図 1 に示す．第 1 階層は，ロボットにセンサを搭載するための階層である．この階層には，距離を計測する赤外線センサの GP2Y0A21YK0F が搭載されている．図中の想像線（二点鎖線）で描かれた物体が，その赤外線センサ GP2Y0A21YK0F である．上図では，GP2Y0A21YK0F の概形とその寸法のみしか記載していない．

次に，ロボットの第 2 階層の投影図を図 2 に示す．ただし，上面図および右側面図は，回路や電子素子などの点数が多いために非常に複雑となるので，記載は割愛する．第 2 階層は，ロボットに電子回路を搭載するための階層である．この階層には，Armadillo-300，Arduino UNO，AGB65-RSC2 および AGB65-232C が搭載されている．図中の Armadillo-300 は，回路を配置した方向が分かるように，UBS のコネクタ，LAN ケーブルのコネクタ，コンパクト・ディスクのコネクタの概形を右から順に描いた．第 2 階層に配置した回路は，それぞれ電気的に繋がっている．これらの回路の構成については，後節の回路図を参考にされたい．

最後に，ロボットの第 3 階層の投影図を以下に示す．第 3 階層は，ロボットの移動機構のための階層である．図 3 の通り，第 3 階層には H ブリッジ回路を搭載している．H ブリッジ回路は，DC モータの回転方向を制御するものである．これについては，後節の回路図を参考にされたい．また，第 3 階層の裏面には，オムニホイール・モータを搭載している．オムニホイール・モータは，能動回転方向に対して垂直な方向に受動回転できる．したがって，図の通りに十字型に搭載しても，ロボットは平行する 2 輪を同じ方向に回転させることで，問題なく移動することができる．この搭載位置の寸法は，後節の組立図を参考にされたい．

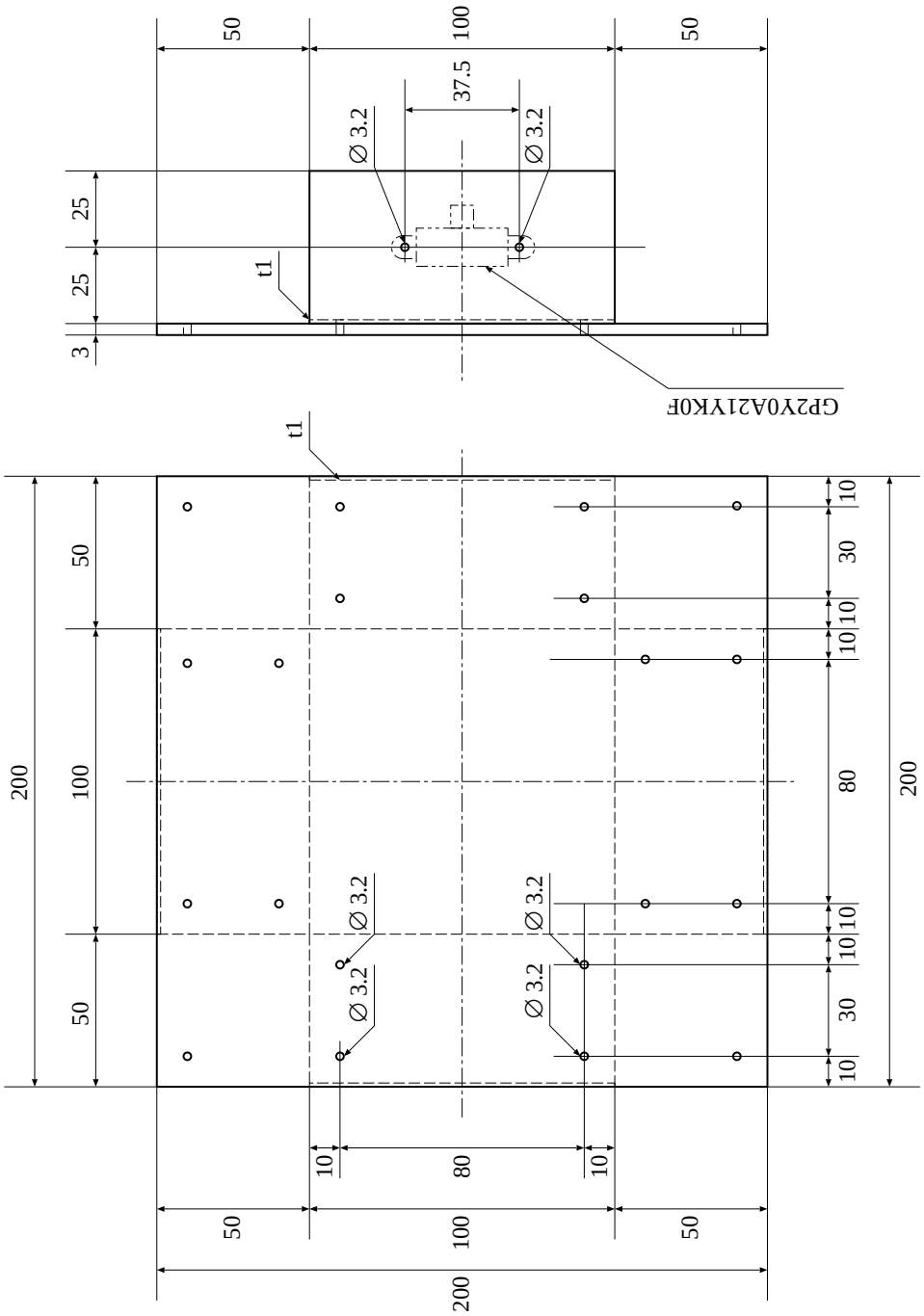


図 1: 第 1 階層（最上階層）の投影図（反時計回りに 90 度回転）

組立図

本研究の実験で使用したロボットの組立図を記載する．ここで，以下に載せる組立図は，第三角法に従って製図したものである．

はじめに，ロボットの上面図を図 4 に示す．図の通り，ロボットは幅 400 (mm)，奥行き 400 (mm) の広さをもつ．第 1 階層，第 2 階層，第 3 階層は，ロボットの機体の中央で階層的に組み立てられる．かくれ線（破線）で描かれた物体は，オムニホイール Urethane-Omni TYPE2581 と DC モータ RP380-ST である[1]．これは，第 3 階層の裏側で十字方向に取り

[1] これは，株式会社の土佐電子が販売している商品である．オムニホイールは，同社が販売している「オムニホイール TD-80」，モータは，株式会社タミヤが販売している 380 シリーズのギヤード・モータである．図 4, 図 5 では，それぞれの概形とそれらの特徴的な部分の寸法のみしか記載していない．

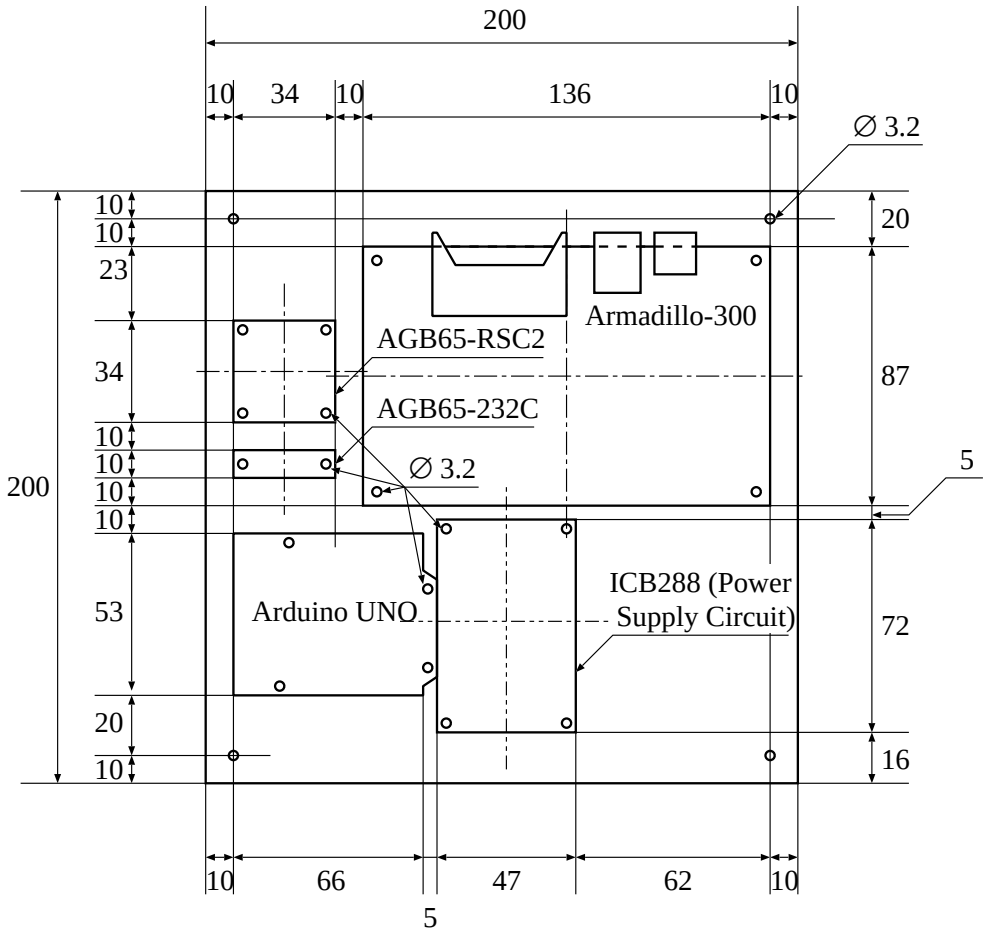


図 2: 第 2 階層（中間層）の投影図

付けられている．中央に存在する 4 つの穴は，それぞれのモータの電力線を通すためのものである．

次に，ロボットの正面図を図 5 に示す．図の通り，ロボット 262 (mm) の高さを持つ．また，赤外線センサ GP2Y0A21YK0F は 234 (mm) の高さでロボットに搭載されている．さらに，赤外線センサ GP2Y0A21YK0F は第 3 階層の端から中心側へ 100 mm のところに搭載されている．

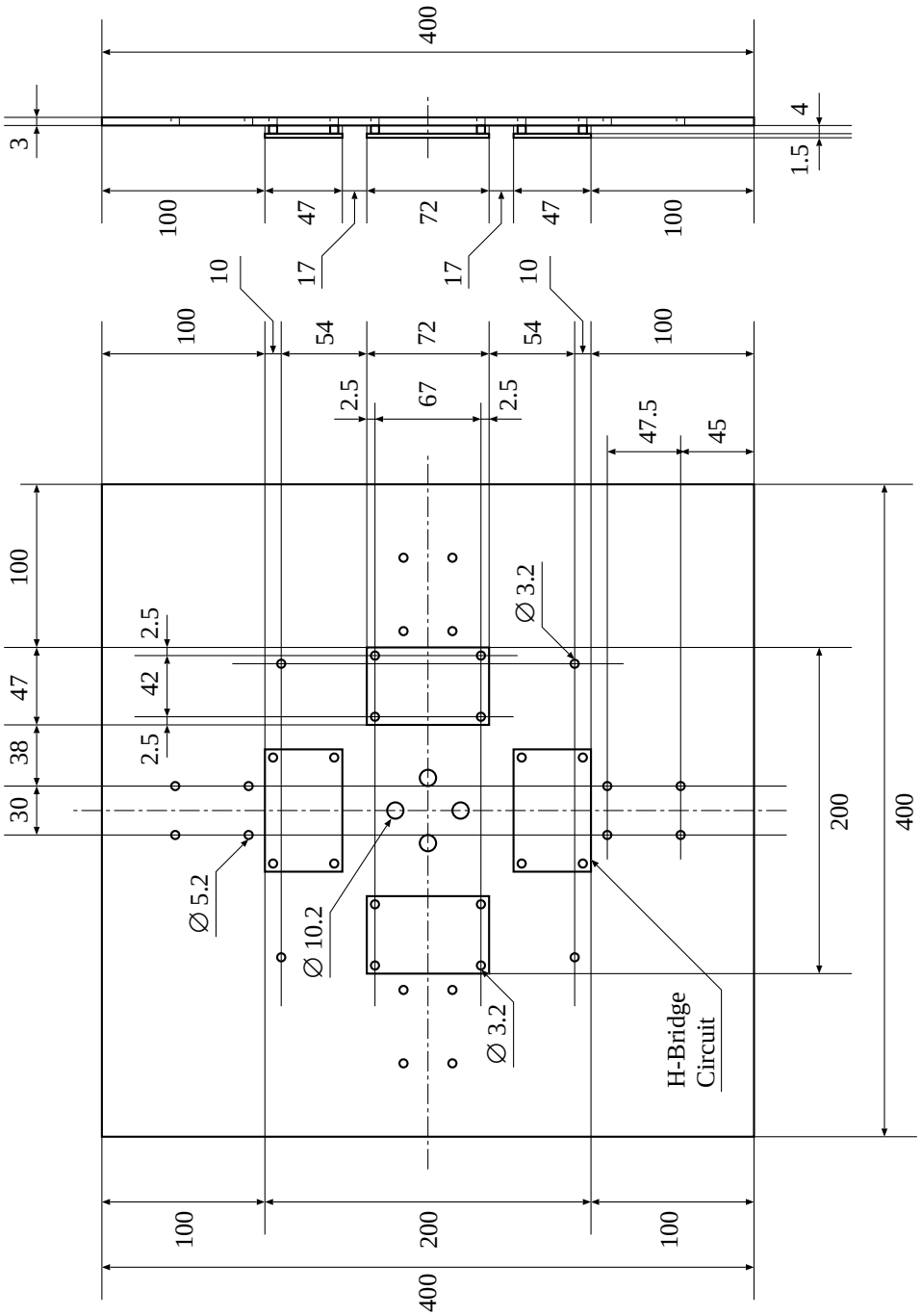


図 3: 第 3 階層（最下階層）の投影図（反時計回りに 90 度回転）

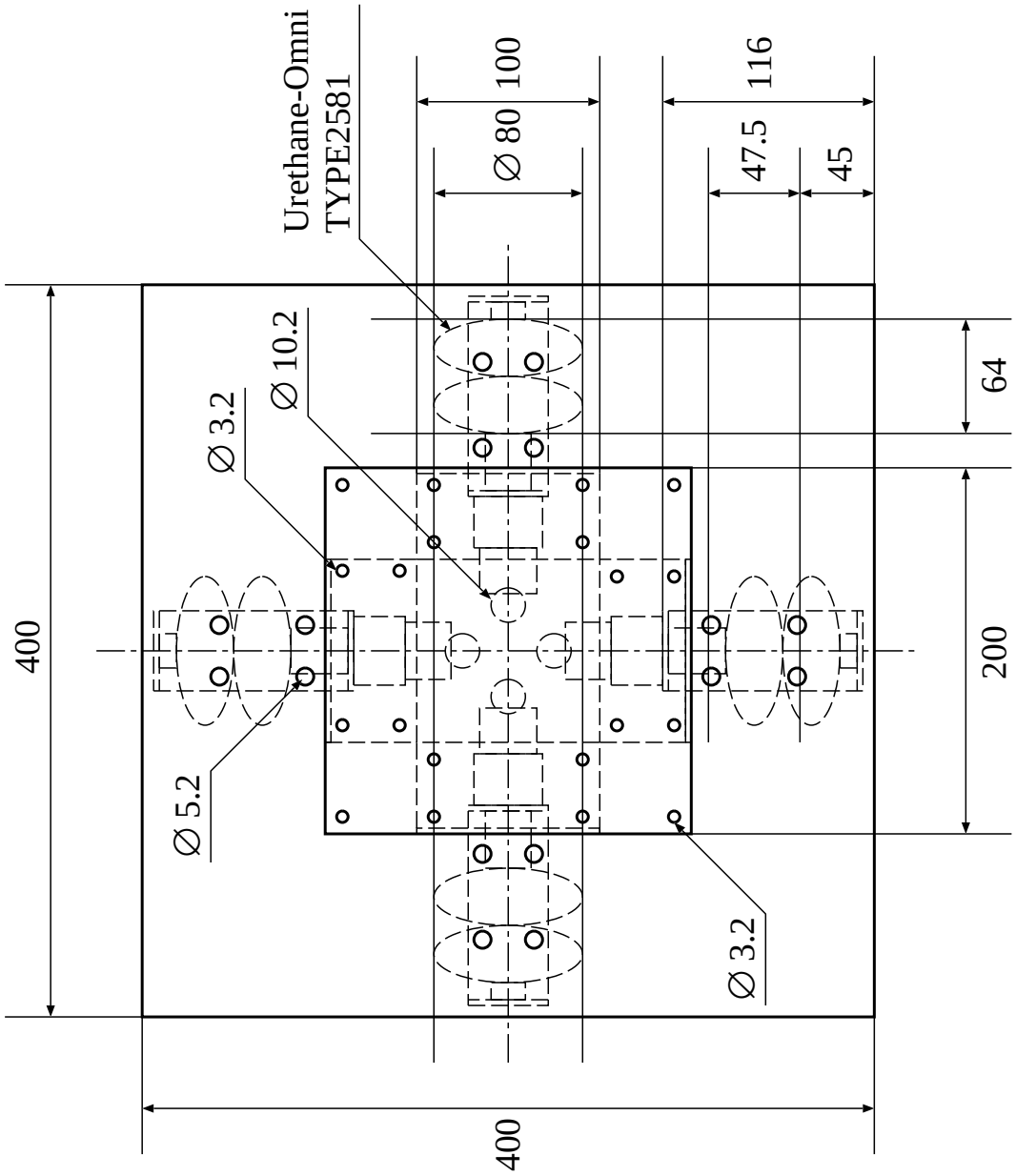


図 4: ロボットの上面図（反時計回りに 90 度回転）

回路図

本研究の実験で使用したロボットを構成する電子回路の回路図を記載する．ここでは，電子回路の構成を表現するブロック図と，H ブリッジ回路の実体配線図を記載する．

はじめに，ロボットの電子回路の構成を表現するブロック図を図 6 に示す．Armadillo-300 は，学習のプログラムを実行するプラットフォームとして動作し，Arduino UNO にモータ駆動の指令やセンサ情報の要求をする．AGB65-232C は，シリアル通信の信号レベルを変換するインターフェイス回路である．Armadillo-300 と Arduino UNO は，AGB65-232C を介してシリアル通信によって双方向にデータ転送する．このとき，Armadillo-300 は RS232C（PC）レベルの信号を出力し，Arduino UNO は TTL（5.0V マイコン）レベルの信号を出力する．Arduino UNO は，Armadillo-300 の命令に従って，モータを駆動させたりセンサ情報を取得する．Arduino UNO がモータを駆動させるときは，H ブリッジ回路に制御信号を送る．H ブリッジ回路は，2 値 × 2 本の制御信号の組み合わせにより，DC モータを任意の方向に回転させることができる．AGB-RSC2 は，RC サーボモータを使用するために搭

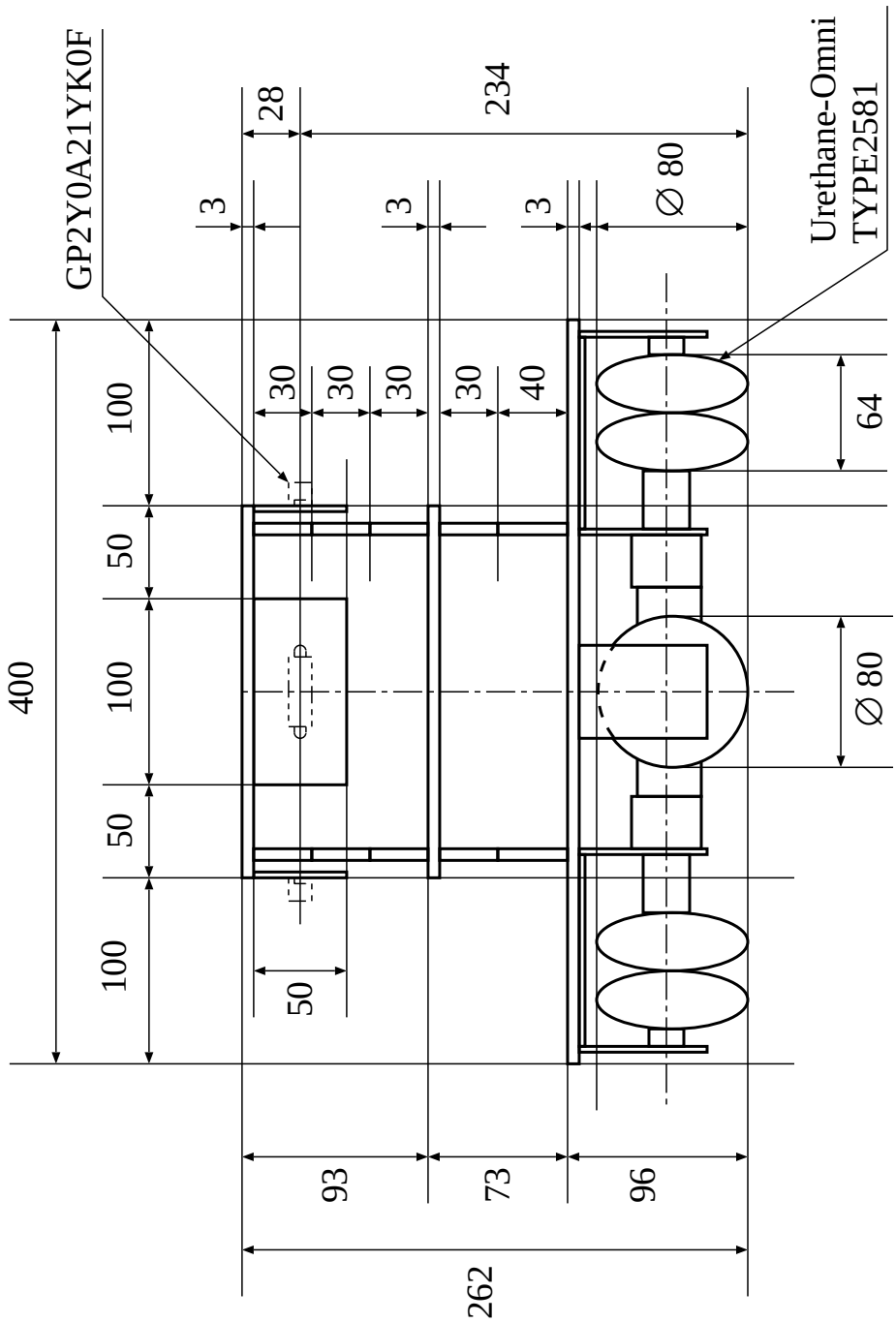


図 5: ロボットの正面図（反時計回りに 90 度回転）

載しているのではなく，単に AGB65-232C へ電力を仲介して供給するように使用している．ICB288(Power Supply Circuit) は，オムニホイール・モータおよび AGB65-RSC2 へ電力を供給している．

次に，H ブリッジの実体配線図を図 7 に示す．ここで，図 7 は，プリント基板エディタ Paas (Parts Arrange Support System) を使用して作成したものである．緑色の線は裏面被膜配線，青色の線は裏面配線である．左側のピンソケットは制御信号用，真ん中のピンソケットはモータ駆動用，右側のピンソケットは電力供給用である．長方形の物体は，電界効果トランジスタ（FET）である．左側は p 型 FET であり，TOSHIBA 製の 2SJ334 である．右側は n 型 FET であり，同社製の 2SK2232 である．この H ブリッジ回路は，Arduino UNO から送信される制御信号に従って機能するものである．ここで，左側のピンソケット

の上側が信号線 1，下側が信号線 2 とすると，これらの信号線の電圧の組み合わせとモータの回転は，表 1 の通りに対応する．これにより，Armadillo-300 は，Arduino UNO を介してモータを任意の方向に回転，もしくは静止やブレーキさせることができる．

表 1: H ブリッジ回路の機能

信号線 1	信号線 2	モータの回転
OFF	OFF	静止
OFF	ON	逆回転
ON	OFF	正回転
ON	ON	ブレーキ

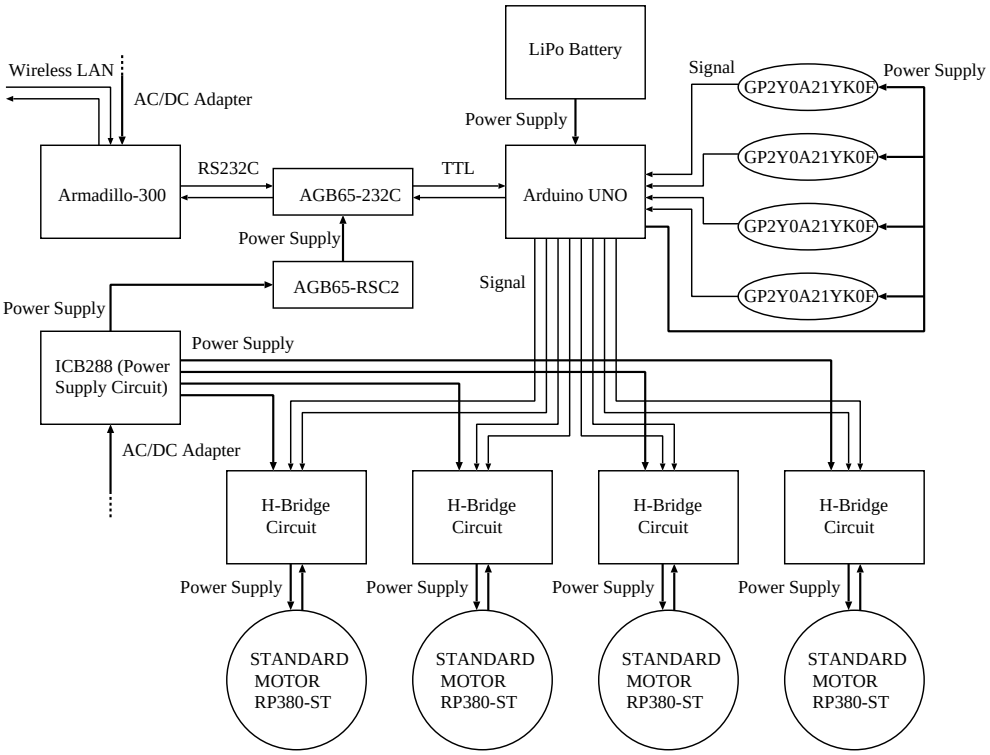
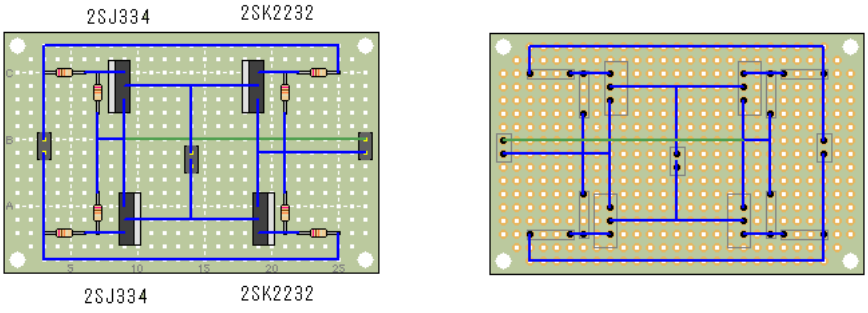


図 6: 電子回路の構成（ブロック図）



(a) 表面　　(b) 裏面

図 7: H ブリッジ回路の実体配線図

参考文献

[1] Masato Hirose, Kenich Ogawa,“Honda humanoid robots developmen”, Philosophical Transactions of the Royal Society A, vol.365, No.1850, 11-9, 2007.

[2] Junichi Osada, Shinichi Ohnaka, Miki Sato, “The scenario and design process of child-care robot, PaPeR”, Proceedings of the 2006 ACM SIGCHI international conference on Advances in computer entertainment technology Article No. 80, 2006.

[3] 神田崇行, 石黒浩, 小野哲雄, 今井倫太, 前田武志, 中津良平, “研究用プラットホームとしての日常活動型ロボット “Robovie” の開発”, 電子情報通信学会論文誌, D-I, Vol.J85-D-I, No.4, pp.380-389, 2002.

[4] 柴田 崇徳, “人の心を豊かにするメンタルコミットロボット”, 日本機械学會誌, Vol.109, No.1051, 2006.

[5] 川内直人, 古結義浩, 長島是, 大西献, 日浦亮太, “ホームユースロボット “wakamaru””, 三菱重工技報, Vol.40, No.5, 2003.

[6] K.Yoshida, “Achievements in space robotics”, IEEE Robotics & Automation Magazine, Vol.16, No4, pp.20-28, 2009.

[7] Richard Volpe, Richard Doyle, “Recent Robotics Developments at NASA/JPL”, 日本ロボット学会誌, Vol.27, No.5, pp.490-493, 2009.

[8] 中須賀真一, 田中秀幸, 矢入健久, “小型衛星とロボット・知能化研”, 日本ロボット学会誌, Vol.27, No.5, pp.502-505, 2009.

[9] 稲葉典康, “宇宙機運用への「AI」技術応用の期待”, 人工知能学会誌, Vol.21, No.1, pp.14-19, 2006.

[10] 久保田考, “惑星別探査ローバ” 日本ロボット学会誌, Vol.21, No.5, pp.468-471, 2003.

[11] 茂原正道, 西田信一郎, “宇宙探査ローバの作り方”, 日本ロボット学会誌, Vol.21, No.5, pp.472-476, 2003.

[12] 金森洋史, “月・惑星探査のテラメカニクス”, 日本ロボット学会誌, Vol.21, No.5, pp.480-483, 2003.

[13] 玉圭樹, 中谷一郎, “深宇宙探査機の自律化とその検証”, 日本ロボット学会誌, Vol.21, No.5, pp.488-493, 2003.

[14] 浦環, “水中に求められるロボット”, 日本ロボット学会誌, Vol.22, No.6, pp.692-696, 2004.

[15] 近藤逸人, “知的観測を行う水中ロボット”, 日本ロボット学会誌, Vol.22, No.6, pp.714-717, 2004.

[16] Xichuan Lin, Shuxiang Guo, “Development of a Spherical Underwater Robot Equipped with Multiple Vectored Water-Jet-Based Thrusters”, Journal of Intelligent & Robotic Systems, Volume 67, Issue 3-4, pp 307-321, 2012.

[17] 鈴木正憲, “原子力発電プラント水中検査用 ROV の開発”, 日本ロボット学会誌, Vol.22, No.6, pp.697-701, 2004.

[18] 伊藤智之, 木村元比古, “小型水中点検ロボットの開発”, 日本ロボット学会誌, Vol.22, No.6, pp.702-705, 2004.

[19] K.Ohono, S.Kawatsuma, T.Okada, E.Takeuchi, K.Higashi, S.Tadokoro, “Robotic control vehicle for measuring radiation in Fukushima Daiichi Nuclear Power Plant”, IEEE International Symposium on Safety, Security and Rescue Robotics 2011, pp.38-48, 2011.

[20] 小柳栄次, “サブクローラを持つレスキューロボット”, 日本ロボット学会誌, Vol.28, No.2, pp.147-150, 2010.

[21] 広瀬茂雄, “ヘビ型ロボットの移動機構”, 日本ロボット学会誌, Vol.28, No.2, pp.151-155, 2010.

[22] Koji Ueda, Michele Guarnieri, Takao Inoh, Paulo Debenest, Ryuichi Hodoshima, Edwardo.F.Fukushima, and Shigeo Hirose, “Development of HELIOS IX: An Arm-Equipped Tracked Vehicle”, Journal of Robotics and Mechatronics, Vol.23, No.6, pp.1031-1040, 2011.

[23] 濱田彰一, 間野隆久, “欧米における原子力防災ロボットの調査報告”, 日本ロボット学会誌, Vol.19, No.6, pp.678-684, 2001.

[24] 田所諭, 大須賀公一, 天野久徳, “レスキューロボット”, 日本ロボット学会誌, Vol.19, No.6, pp.685-688, 2001.

[25] 亀川哲志，松野文俊, “遠隔操作性を考慮した双頭ヘビ型レスキューロボット KOHGA の開発”, 日本ロボット学会誌, Vol.25, No.7, pp.1074-1081，2007.

[26] Park Chang-Woo, Kim Bong-Seok, Song Jae-Bok, Hwang Jung-Hoon, “Design of robotic surgical instrument for minimally invasive surgical robot system”, 12th International Conference on Control, Automation and Systems (ICCAS), 2012.

[27] Yuki Horise, Atsushi Nishikawa, Mitsugu Sekimoto, Yu Kitanaka, Norikatsu Miyoshi, Shuji Takiguchi, Yuichiro Doki, Masaki Mori, Fumio Miyazaki, “Development and evaluation of a master-slave robot system for single-incision laparoscopic surgery”, International Journal of Computer Assisted Radiology and Surgery, Vol.7, No.2, pp 289-296, 2012.

[28] Carl A. Nelson, Xiaoli Zhang, Bhavin C. Shah, Matthew R. Goede, Dmitry Oleynikov, “Multipurpose surgical robot as a laparoscope assistant”, Surgical Endoscopy, Volume 24, Issue 7, pp.1528-1532, 2012.

[29] 菅原研次, “情報工学入門シリーズ 20 人工知能 第二版”, 森北出版,1997.

[30] 白井良明, “コンピュータ数学シリーズ 16 人工知能の理論”, コロナ社, 1992.

[31] 堂下修司, 西田豊明, 三浦欽也, “様相論理をその情報処理への応用 (I) 様相論理”, 情報処理 Vol.29, No.1, pp.2-10, 1988.

[32] 松本一教, 内平直志, 本井田真一, “時相論理とその応用 (¡特集¿非標準論理とその応用)”, 情報処理 Vol.30, No.6, pp.651-657, 1989.

[33] Klahr D, Langley P and Neches R, “Production System Models of Learning and Development”, Cambridge, Mass.: The MIT Press, 1987.

[34] G.A.Ringland and D.A.Duce, “Knowledge Representation-An Introduction”, Research Studies Press. 1998.

[35] Marvin Minsky, “A FRAMEWORK FOR REPRESENTING KNOWLEDGE”, Artificial Intelligence Memo, No.306, 1974.

[36] 中原和洋, 山田茂雄, “日本でのコモンセンス知識獲得を目的とした Web ゲームの開発と評価”, UNISYS TECHNOLOGY REVIEW, No.107, pp.13-23, 2011.

[37] 柳吉沫, 志村正道, “故障診断用エキスパートシステムにおける知識獲得”, 人工知能学会誌, 人工知能学会誌, Vol.1, No.1, pp.93-100, 1986.

[38] 飯田龍, 小林のぞみ, 乾健太郎, 松本裕治, 立石健二, 福島俊一, “意見抽出を目的とした機械学習による属性-評価値対同定”, 情報処理学会研究報告 自然言語処理研究会報告 No.1, pp.21-28, 2005.

[39] 安居院猛, 長橋宏, 高橋裕樹, “ニューラルプログラム”, 昭晃堂, 1993.

[40] 石川眞澄, “ニューラルネットワークと高次情報処理”, シミュレーション, Vol.12, No.3, pp.177-186, 1993.

[41] 中西完太, 筧一彦, “ニューラル・ネットワークによる概念学習のモデル化”, 電子情報通信学会技術研究報告 TL 思考と言語, Vol.99, No.76, 1-7, 1999.

[42] Rumelhart, D.E., Hinton, G.E., and Williams, R.J., “Learning internal representations by error propagation.”, Parallel Distributed Processing: Explorations in the Microstructures of Cognition, 1, MIT press, pp.318-362, 1986.

[43] 木本武一郎, 萩原政文, “ファジィ認知マップの自動生成手法の提案”, 日本ファジィ学会誌, Vol.10, No.1, pp.81-88, 1998.

[44] Stylios C.D, “Modeling complex systems using fuzzy cognitive maps”, IEEE Transactions on Systems Man and Cybernetics Part A:Systems and Humans, Vol.34, No.1, pp.155-162, 2004.

[45] 土井利忠, 藤田雅博, 下村秀樹, “インテリジェンス・ダイナミクス 1 脳身体性・ロボット 知能の創発を目指して”, シュプリンガー・ジャパン ,2005.

[46] R.Pfeifer, J.Bongard, “How the Body Shapes the Way We think : A New View of Intelligence”, MIT Press, 2010.

[47] 浅田稔, 石黒浩, 國吉康夫, “認知ロボティクスの目指すもの”, 日本ロボット学会誌, Vol.17 No.1, pp.1-5, 1999.

[48] Michael I. Jordan, “Forward models:Supervised learning with a distal teacher”, Cognitive Science, Vol.16, pp.307-354, 1992.

[49] 魚田紫織, 横井博一, “階層型運動スキーマによるロボットハンドの運動多様性の実現”, 電子情報通信学会技術研究報告, ニューロコンピューティング Vol.103, pp.25-28, 2003.

[50] 原正之, 川辺直人, 久嶋肇, “強化学習を用いた人型ロボットよる大車輪運動の獲得”, 第 25 回日本ロボット学会学術講演会予稿集, 3N34, 2007.

[51] 川村貞夫, 川村竜也, 藤野大助, 宮崎文夫, 有本卓, “運動パターン学習による 2 足歩行ロボットの歩行実現”, 日本ロボット学会誌, Vol.3, No.3, pp.177-187, 1985.

[52] Andrew G. Barto, Richard S. Sutton, Charles W. Anderson, “Neuronlike Adaptive Elements That Can Solve Difficult Learning Control Problems”, IEEE Trans. on Sys., Man, Cybern., SMC-13, pp.834-846, 1983.

[53] Richard Fikes, Nils J. Nilsson, “STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving”, pp.189-208, 1971.

[54] 宮下和雄, “プランニングとスケジューリング”, 人工知能学会誌, Vol.16, No.5, pp.611-616, 2001.

[55] 三浦純, “ロボットにおけるプランニング”, 人工知能学会誌, Vol.16, No.5, pp.617-622, 2001.

[56] 石田享, 新保仁, “実時間探索による経路学習”, 日本ロボット学会誌, Vol.11, No.3, pp.411-419, 1996.

[57] 小倉崇, 岡田慧, 稲葉雅幸, 井上博允, “ヒューマノイドのオンサイト誘導プランナの実現と行動学習の研究”, 日本機械学会ロボティクス・メカトロニクス講演会’04 講演論文集, pp. 2P1-H-76, 2004.

[58] 小平実, 大友照彦, 田中敦, 岩月正見, 大内隆夫, “ニューラルネットを用いた移動ロボット車の障害物回避走行制御”, 電子情報通信学会論文誌 D-II, 情報・システム, II-情報処理, pp.91-100, 1996.

[59] Stephane Ross, J. Andrew Bagnell, “Efficient Reductions for Imitation Learning”, Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, 2010.

[60] 天岡侑己, 下平順, 平井宏明, 宮崎文夫, “主成分分析を用いたヒトのスキルの再現とロボットへの移植”, 日本ロボット学会誌, Vol.28, No.8, pp.989-995, 2010.

[61] 宮本弘行, “ヒト運動の最適化原理と見まねに基づくタスク学習”, 日本ロボット学会誌, Vol.19, No.5, pp.547-550, 2001.

[62] 鮫島和幸，銅谷賢治, 川人光男, “強化学習 MOSAIC:予測性によるシンボル化と見まね学習”, 日本ロボット学会誌, Vol.19, No.5, pp.551-556, 2001.

[63] 片上大輔, 山田誠二, “ノード使用頻度に依存した交叉による進化ロボティクスの高速化”, 人工知能学会誌, Vol.16, No.4, pp.329-399, 2001.

[64] 岡田将吾，伊豆蔵拓也，名渕博人，高橋徹，西田豊明, “言語・非言語情報を統合した指示パターンに対応するロボットの行動則獲得”, 第 32 回人工知能学会 AI チャレンジ研究会資料, 2010.

[65] 田中健太, 木原康之, 横小路泰義, “人間の直接教示動作の統計的性質に基づいた折り紙ロボットの目標軌道とセンサフィードバック則生成法”, 日本ロボット学会誌, Vol.27, No.6, 2009.

[66] Dario Floreano, Stefano Nolfi, “Adaptive Behavior in Competing Co-evolving Species”, In Fourth European Conference on Artificial Life, pp.378-387, 1997.

[67] Rafal Drezewski, “A Model of Co-evolution in Multi-agent System”, 3rd International Central and Eastern European Conference on Multi-Agent Systems, CEEMAS 2003 Prague Proceedings, pp.314-323, 2003.

[68] Jordan B. Pollack, Alan D. Blair, “Co-Evolution in the Successful Learning of Backgammon Strategy”, Machine Learning, Vol.32, No.3, pp.225-240, 1998.

[69] LIVIU PANAIT, SEAN LUKE, “Cooperative Multi-Agent Learning: The State of Art”, Autonomous Agents and Multi-Agent Systems, Vol.11, No.3, pp.387-434, 2005.

[70] 有田隆也, “コミュニケーションの創発”, 計測と制御, Vol.48, No.1, 2009.

[71] 日下航, 尾形哲也, 小嶋秀樹, 高橋徹, 奥野博, “RNN を備えた 2 体のロボット間における身体性に基づいた動的コミュニケーションの創発”, 第 27 回日本ロボット学会学術講演会, 2009.

[72] Marc Szymanski, Tobias Breitling, Jorg Seyfried, Heinz Worn, “Distributed Shortest-Path Finding by a Micro-robot Swarm”, Ant Colony Optimization and Swarm Intelligence Lecture Notes in Computer Science Volume 4150, pp.404-411, 2006.

[73] 倉爪亮,　広瀬茂男, 岩崎倫三, 長田茂美, “協調ポジショニングシステムの研究－ CPS アクティブタッチ融合型地図生成法－”, 日本ロボット学会誌, Vol.17, No.1, pp.84-90, 1999.

[74] 石岡宏治，開一夫，安西祐一郎, “MARSHA：複数の自律移動ロボットの個体差を考慮した地図獲得システムの設計と制御”，日本ロボット学会誌，Vol.12, No.6, pp.846-856, 1994.

[75] 横矢剛, 長谷川勉, 倉爪亮, “群ロボットによる未知環境三次元地図の自動作成のための動作計画手法”, 電子情報通信学会論文誌. D, 情報・システム Vol.93, No.6, pp.1024-1035, 2010.

[76] 吉村裕司，太田順，井上康介，平野智一，倉林大輔, 新井民夫,“群ロボットによる多数物体の繰返し搬送計画”, 日本ロボット学会誌, Vol.16, No.4, pp.499-507, 1998.

[77] 風間俊哉, 菅原研, 水口毅, 渡辺俊典, “場との相互作用による群ロボットの協調搬送行動”, 電子情報通信学会技術研究報告. NLP, 非線形問題 Vo.104, No.50, pp.41-46, 2004.

[78] J.I.U. Rubrico et al, “Scheduling Multiple Agents for Picking Products in a Warehouse”, Proc. 2006 IEEE Int. Conf. Robotics and Automation, pp.1438-1443, 2006.

[79] 黒河治久, 吉田英一, 神村明哉, 富田康浩, 村田智, 小鍛治繁, “変形し移動する自立モジュール型ロボット,” 日本ロボット学会誌, Vol.21, No.8, pp.855-859, 2003.

[80] 黒河治久, 神村明哉, 富田康浩, 村田智, 小鍛治繁, “モジュール型ロボット M-TRAN の分散型制御システムと変形実験”, 自律分散システム・シンポジウム資料, Vol.18, pp.49-52, 2006.

[81] 有馬哲, 石村貞夫, “多変量解析のはなし”, 東京図書, 1987.

[82] 銅谷賢治, “計算神経科学への招待〜脳の学習機構の理解を目指して〜”, サイエンス社, 2007.

[83] T. Kohonen, “Self-Organizing Maps”, Spriner-Vcrlag, New York, 2001 third edition.

[84] Ian D. Kelly, David A. Keating, “Increased Learning Rates Through the Sharing of Experiences”, Proceedings of the Seventh IEEE International Conference on Fuzzy Systems, 1998

[85] Ming Tan, “Multi-Agent Reinforcement Learning: Independent vs. Cooperative Agents”, In Proceedings of the Tenth International Conference on Machine Learning , 1993

[86] Ian D. Kelly, David A. Keating, Kevin Warwick, “Mutual Learning By Autonomous Mobile”, Proceedings of the First Workshop on Teleoperation and Robotics, Applications in Science and Arts, 1997

[87] M. N. Ahmadabadi, M. Asadpur, S. H. Khodanbakhsh, E. Nakano, “Expertness Measuring in Cooperative Learning”, Proceedings of International Conference on Intelligent Robots and Systems 2000 (IROS2000), Vol. 3, pp. 2261-2267 (2000).

[88] M. N. Ahmadabadi, M. Asadpur, “Expertness Based Cooperative Q-learning”, IEEE Transactions on Systems, Man, and Cybernetics, Vol. 32, No. 1 , pp. 66-76 (2002).

[89] M. Asada, S. Noda, K. Hosoda, “Action-Based Sensor Space Categorization for Robot Learning”, Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems 1996, pp. 1518-1524 (1996).

[90] H. Ishiguro, R. Sato, T. Ishida, “Robot Oriented State Space Construction”, Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems 1996, pp. 1496-1501 (1996).

[91] S. Samejima, T. Omori, “Adaptive Internal State Space Construction Method for Reinforcement Learning of a Real-World Agent”, Neural Networks, Vol.12, pp.1143-1155 (1999).

[92] A. J. Smith, “Applications of the Self-Organising Map to Reinforcement Learning”, Neural Networks, Vol. 15, pp. 1107-1124 (2002).

[93] Kyathy Thi Aung, Takayasu Fuchda, “A Proposition of Adaptive State Space Partition in Reinforcement Learning with Voronoi Tessellation”, Proceedings of the 17th International Symposium on Artificial Life and Robotics 2012, pp. 638-641 (2012).

[94] Richard S. Sutton, Andrew G. Barto, “Reinforcement Learning”, The MIT Press, 1998.

[95] Leslie Park Kaelbling, Michael L. Littman, Andrew W. Moore, “Reinforcement Learning A Survey”, Journal of Artificial Intelligence Research 4, pp.237-285, 1996.

[96] 木村元, 宮崎和光, 小林重信, “強化学習システムの設計指針”, 計測と情報, Vol.38, No.10, pp.618-623, 1996.

[97] J. Morimoto, K. Doya, “Acquisition of stand-up behavior by a real robot using hierarchical reinforcement learning ”, Robotics and Autonomous Systems, Volume 36, pp. 37-51, 2001

[98] H. Kimura, T. Yamashita and S. Kobayashi, “Reinforcement Learning of Walking Behavior for a Four-Legged Robot”, 40th IEEE Conf. on Decision and Control, pp.411-416, 2001

[99] M. Asada, E. Uchibe, and K. Hosoda, “Cooperative behavior acquisition for mobile robots in dynamically changing real worlds via vision-based reinforcement learning and development”, Artificial Intelligence, Vol.110, pp.275-292, 1999

[100] Maja J. Matari, “Reinforcement Learning in the Multi-Robot Domain”, Autonomous Robots, Vol.4, Number 1, 1997

[101] William D. Smart, Leslie Pack Kaelbling, “Effective reinforcement learning for mobile robots”, Robotics and Automation, 2002. Proceedings. ICRA ’02. IEEE International Conference, 3404-3410 vol.4, 2002

[102] 伊庭幸人, “統計学者・数理工学者のための統計物理入門　-格子スピン模型とマルコフ連鎖モンテカルロ法を中心として-”, 統計数理研究所, 1997.

[103] 伊庭幸人, 種村正美, 大森裕浩, 和合肇, 佐藤整尚, 高橋昭彦, “計算統計 II マルコフ連鎖モンテカルロ法とその周辺（統計科学のフロンティア 12）”, 岩波書店, 2005.

[104] 浅田稔, 北野宏明, “ロボカップ戦略: 研究プロジェクトとしての意義と価値”, 日本ロボット学会誌, Vol.18, No.8, pp.1081-1084, 2000.

[105] 田所諭, “ロボカップレスキューリーグ”, 日本ロボット学会誌, Vol.27, No.9, pp.983-986, 2009.

研究業績

学会誌等

- Yasutaka Kishima, Kentarou Kurashige, “Reduction of state space in reinforcement learning by sensor selection”, Artificial Life and Robotics, Springer, 2013, 10.1007/s10015-013-0092-2.
- Yasutaka Kishima, Kentarou Kurashige, “Decision making in reinforcement learning using a modified learning space based on the importance of sensors”, Journal of Sensors, Article ID 141353, 9 pages, 2013, doi:10.1155/2013/141353.

国際会議

- Yasutaka Kishima, Kentarou Kurashige, “Growth of individual intelligence using communication”, SCIS & ISIS 2008(CD), pp.287-292, Nagoya, Japan, Sept. 17-21, 2008.
- Yasutaka Kishima, Kentarou Kurashige, “THE GROWTH OF INDIVIDUAL INTELLIGENCE IN GROUPS OF AGENTS BYAUTONOMOUS SELECTION OF OTHERS TO COMMUNICATE TO”, WAC 2010, IFMIP-545(CD-ROM), Kobe, Japan, Sept. 19th - 23, 2010.
- Yasutaka Kishima, Kentarou Kurashige, Toshinobu Numata, “Reduction of learning space by making a choice of sensor information”, Proceedings of the seventeenth International Symposium on Artificial Life and Robotics (AROB 17th ’12), pp.971-974, Jan. 19-21, 2012, B-Con Plaza, Beppu, Oita, JAPAN.
- Yasutaka Kishima, Kentarou Kurashige, “Reduction of state space on reinforcement learning by sensor selection”, Proceedings of 2012 International Symposium on Micro-NanoMechatronics and Human Science(MSH2012 & Micro-Nano Global COE), pp.138-143, Nov. 4-7, 2012, Nagoya, Japan.

国内学会

- 木島康隆, 倉重健太郎, “強化学習におけるセンサの重要度に応じた状態空間の構成”, 日本ロボット学会第 30 回記念学術講演会, RSJ2012AC4F1-7, 札幌, 北海道, 2012.9.17-20.

その他

- 倉重健太郎, 木島康隆, “群の中の個体知能の発展-コミュニケーションを用いた方策の学習-”, 第四回「認知エージェント技術 (CAT)」研究会, 登別，北海道, 2007.12.