強化学習における状態フィルタの適応的獲得手法に関する研究

永吉，雅人

(Degree)
博士（工学）

(Date of Degree)
2007-03-25

(Date of Publication)
2009-07-15

(Resource Type)
doctoral thesis

(Report Number)
甲4020

(URL)
https://hdl.handle.net/20.500.14094/D1004020

博 士 論 文

強化学習における
状態フィルタの適応的獲得手法に関する研究

平成 19 年 2 月

神戸大学大学院自然科学研究科

永 吉　雅 人

目 次

図 目 次

表 目 次

第 1 章

序論

1.1 研究の背景と目的

近年，人間が関与するシステムの複雑化・大規模化が急速に進みつつある．こうした複雑・大規模なシステムに対して，人間が予め制御規則を設定する従来の方式では，システムの僅かな変化にも対応できないこと等に起因して，適切な制御を行うことが困難となってきている．これに対する解決策の一つとして，自律的・適応的に制御規則を獲得する方式の開発が求められている．これまでにも，制御規則を適応的に獲得する汎用性の高い手法として強化学習 (Reinforcement Learning: RL)[1] が注目され，研究が活発に行われている．強化学習は，対象システムのモデルを必要とせず，明示的な教師なしに報酬および罰という強化信号を拠り所とした試行錯誤により，自律的に適切な制御規則を獲得していく汎用性の高い学習の枠組みである．強化学習手法の応用には，制御プログラミングの自動化・省力化や，人手による場合よりも質の高い制御が可能となるといった期待がもたれている [2]．しかしながら，強化学習手法の実用化のためには，探索 (exploration) と搾取 (exploitation) のトレードオフ，学習が遅いこと，不完全知覚問題など，未だ多くの問題が残されている．その一つに，状態空間の設計問題がある．これは，エージェントがシステムの状態を観測する情報と，その情報をどのくらいの粒度で観測するかで決定されるエージェントの状態空間について，

- 短時間で制御規則を獲得させるためには，エージェントの状態空間を低次元かつ粗く設計する必要がある，
- 獲得する制御規則をより良いものにするためには，エージェントの状態空間を (必要最低限) 高次元かつ細かく設計する必要がある，

といったトレードオフの関係にある要件が課されることに起因するものであり，状態空間を予め適切に設計することが難しいといった問題である．これまで状態空間の設計問題への接近法が多く提案されているものの，ほとんどの手法において強化学習手法が特有のものであり，すべての方法において計算時間が考慮されていないなど，未だ問題が解決されているとは言えない．

また，強化学習の実用化に向けて，マルコフ決定過程 (Markov Decision Processes: MDPs) を前提とした強化学習手法を部分観測系へ拡張した部分観測マルコフ決定過程 (Partially Observable MDPs: POMDPs) へ拡張し適用する試みが近年多くなされている．POMDPs とは，MDPs によりモデル化できるものとして扱われた対象システムとエージェントの部分観測系とを併せて学習器からみて形式化されたモデルである．具体的には，エージェントはシステムの状態を一意に決定できず，システムの状態が異なっていたとしてもそれらを区別することができない不完全知覚 (incomplete perception) 問題 [3, 4] を含めて形式化されたモデルである．これまで POMDPs への接近法が多く提案されているものの，ほとんどの接近法が，膨大なメモリ容量や計算時間，設計者の多くの事前知識を必要としており，また，リカレントニューラルネットワークを用いた接近法 [5, 6] を除くと，ほとんどが連続状態空間を有する POMDPs に対して有効ではないなどの問題が残されており，これらの問題を解決する手法が望まれている．

以上のことを踏まえて本論文では，強化学習の実用化に向けて，強化学習における状態空間の設計問題と不完全知覚問題の解決を目的とする．そして，エージェントの観測空間と行動学習器の間に状態フィルタを導入した計算モデルを提案し，MDPs の場合および POMDPs の場合において状態フィルタの一実現手法を提案する．さらに，より実際的な問題への適用例として，電動車いす使用者の操作負担軽減を目的とした，電動車いすの直進走行システムを提案する．

1.2　論文の構成

以下，本論文の構成について述べる．

第 2 章では，強化学習方式の概要の説明を行い，現在強化学習方式が抱える多くの問題の中から盛んに研究がなされている，探索と搾取のトレードオフ，学習の高速化，不完全知覚問題，状態空間の設計問題といった問題を取り上げ，その問題に対してそれぞれの代表的アプローチを示す．

第 3 章では，エージェントの観測空間と行動学習器の間に状態フィルタを導入した計算モデルを提案し，理想的状態フィルタについても記述する．つぎに，MDPs の場合と POMDPs の場合について，状態フィルタを獲得する方法の違いという観点から従来手法の分類を行い，従来手法の構成面での特徴を明確にする．そして，内部状態を分割・統合する手法の状態フィルタの基本的機能として，内部状態の分割・統合方法について 4 つの調整法を提案する．

第 4 章では，状態空間の設計問題に焦点をあて，MDPs の場合への従来手法の問題点を踏まえて，MDPs を対象としたエントロピーを用いた状態フィルタの一実現法を提案する．さらに，離散状態空間を有する迷路問題および連続状態空間を有するロボットナビゲーション問題を例題として取り上げ，計算機実験を通して，提案手法を従来手法と比べることで，

提案手法の有効性・可能性について検討する．

第 5 章では，POMDPs への対応・状態空間のコンパクト化に焦点を当て，POMDPs の場合への従来手法の問題点を踏まえて，第 4 章で提案した手法を POMDPs に拡張する形で，適応的に履歴情報の記録・参照を行い，繰り返し内部状態を分割・統合することで POMDPs を対象とした状態フィルタの一実現手法を提案する．さらに，強化学習問題の例題として離散状態空間を有する迷路問題および連続状態空間を有するロボットナビゲーション問題を取り上げ，計算機実験を通して，提案手法を従来手法と比べることで，提案手法の有効性・実問題への適用可能性について検討する．

第 6 章では，第 5 章で提案する手法のより実際的な問題への適用例として，電動車いすの直進走行システムを取り上げる．ここではまず，まちにあるバリアとして傾斜路面に注目し，左右 DC モータから駆動力を得る後輪駆動型電動車いすを対象として，三次元上における電動車いすの基本ダイナミクスのモデル化を行う．つぎに，基本的なダイナミクスを考慮した上，電動車いすの直進走行システムとして，第 5 章で提案する手法を再構築し，適用することによって，DC モータへの入力電圧を調節するシステムを提案する．そして，電動車いすの基本ダイナミクスに基づいたシミュレーションによって，(1) 傾斜角が一定な路面を走行する場合，(2) 傾斜角が一定でない路面を走行する場合，さらに，補足として，(3) 重心位置が変化する場合，(4)2 つの路面を走行する場合，について，提案システムの有効性，実際の電動車いすへの適用可能性，をそれぞれ検討する．

最後に，第 7 章は本論文のまとめであり，本研究で得られた結論を整理するとともに，今後の課題について述べる．

第 2 章

強化学習方式

2.1 諸言

本章では，強化学習方式の概要の説明を行い，現在強化学習方式が抱える多くの問題の中から盛んに研究がなされている，探索と搾取のトレードオフ，学習の高速化，不完全知覚問題，状態空間の設計問題といった問題を取り上げ，その問題に対してそれぞれの代表的アプローチを示す.

2.2 強化学習方式の概要

強化学習とは，試行錯誤を通じて制御対象のシステムに適応する学習制御の枠組みである．教師付き学習 (supervised learning) と異なり，対象システムの状態に対する正しい制御信号を明示的に示す教師が存在しない．その代わりスカラー値である報酬と呼ばれる特別な評価信号 (強化信号) を手がかりにして自律的に適切な制御ルールを獲得する．ここで，通常強化学習の分野ではエージェントの側からの用語を用いて，エージェントのもつ状態空間を「内部状態空間」，エージェントが対象システムに対して出力する制御信号を「行動」，制御ルールを逐次的に適切なものに構築していくことを「学習する」と呼んでおり，本論文においてもこれに従うこととする.

強化学習は,「エージェントの行動系列は，現在のシステムの状態においてどのような行動を実行すべきかを規定する制御ルールによって表現できる」という仮定に基づいている．制御ルールとは内部状態空間 $\boldsymbol{S}$ から行動空間 $\boldsymbol{A}$ へ写像 ($f : \boldsymbol{S} \rightarrow \boldsymbol{A}$) を意味している．このときシステムは，不確実性を含む実世界の多くの制御問題を精度よく近似可能な MDP としてモデル化され，システムの状態とエージェントが出力した行動により確率的な状態遷移を行う．時刻 t におけるシステムの状態を $x(t)$，その時エージェントが出力した行動を $a(t)$，次のシステムの状態を $x(t+1)$ とすると，これらのセット $(x(t), a(t), x(t+1))$ に対して，その時のシステムの状態遷移確率 $P(x(t), a(t), x(t+1))$ が定義される．また，同様に時刻 $t+1$ に獲得する報酬 $r(t+1) = r'(x(t), a(t), x(t+1))$ が定義される．以上のように，システムとエージェントは離散的な時間ステップ $t = 0, 1, 2, \cdots$ の各々において相互作用を行う (図 2.1

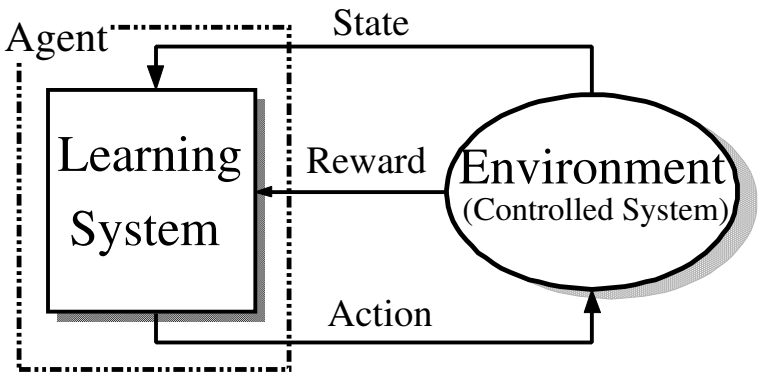

図 2.1: 強化学習におけるエージェントと環境間の相互作用.

参照). ただし，強化学習方式は連続時間の場合に拡張可能 [7, 8] であるが，本論文では単純化のため，離散的な時間に対象を限定する.

一般的な強化学習のタスクは通常，式 (2.1) で表される無限時間に対する報酬の減衰総和である積算報酬 (R)(報酬が確率的であるならば積算報酬の期待値) を最大化する政策を発見することである.

$$R(t) = r(t+1) + \gamma r(t+2) + \gamma^2 r(t+3) + \cdots = r(t+1) + \sum_{n=1}^{\infty} \gamma^n r(t+n+1) \quad (2.1)$$

ここで，$\gamma \in [0, 1]$ は割引率 (discount rate) を表し，γ^n の値は n の値の増加とともに減少する．これは直観的に，遠い将来 (n が大きい) の報酬は現在の報酬や近い将来に比べて一般に信頼度が低いと理解することができ，もし $\gamma = 0$ ならば，エージェントは即時報酬 $r(t)$ のみを最大化するようになる．一方，$\gamma = 1$ ならば，エージェントは将来の報酬も考慮し，長期的な展望を持つようになる.

遷移確率と報酬の分布が既知であれば通常の動的計画法の枠組みで最適な制御ルールを求めることができる．しかし，基本的に強化学習は，エージェントの持つシステムに対する知識は全くないことを前提としており，エージェントがシステムを試行錯誤的に探索しながら，逐次的にシステムの状態遷移と最適な制御ルールを推定していく必要がある.

2.3　強化学習方式の問題点

探索と搾取のトレードオフ

搾取とは現在最も高い報酬が得られると予測される行動をとることである．一方探索とは，搾取より，よい行動が存在するかを調べるために，現在最も高い報酬が得られると予測される行動以外の行動をとることである．エージェントの目的は，最終的に得られる積算報酬を最大化することである．そのためには，探索と搾取をバランスよく行わなければならな

い．従来の強化学習アルゴリズムは探索と搾取のいずれかに重点を置き，このトレードオフを考慮していなかった．

これに対し，MarcoPolo[9] は，このトレードオフを考慮した手法であるが，システムがMDPs であることを仮定しているため，システムの状態空間のサイズが巨大になると対応できなくなる点や，探索と搾取の割合は設計者がタスクによって予め調節する必要があるため，設計者のタスクに対する予備知識を前提としており，まだ問題点が残る．また，従来提案されている強化学習手法に対して探索と搾取のトレードオフの問題を改善する手法が提案されている [10, 11] が，適用範囲が限られているため本質的な解決には至っていない．

学習の高速化

強化学習は基本的に，タスクに対する予備知識が全くない状況から相互作用のみによって学習する．そのため，学習の収束に膨大な相互作用が必要になり，多くの時間を必要とする．強化学習は，実時間で学習を行うシステムへの応用が期待されていることを考えると，学習の高速化は重要な課題である．

これに対して，適正度の履歴 (eligibility trace) を用いる手法 [12, 13, 14] は，訪問したシステムの状態に対応づけされたエージェントの内部状態に適正度を設定し，適正度を持つ内部状態の価値関数を一斉に更新することによって，学習を高速化する．しかしここで学習の高速化とは，収束までの相互作用回数の短縮のことであり，時間計算量のことではない．この手法は，時間計算量は逆に増えてしまう．

一方，プランニング (planning) を行う手法 [15] は，得られた経験 (「経験」とはシステムとの相互作用から得られた結果，つまり行動前のシステムの状態，その時の行動，その結果遷移した次のシステムの状態と報 酬のセットのことを示し,「経験する」とは上記のセットを獲得することを示す) を使ってシステムのモデルを作り，そこから得られるシミュレーション上の経験から価値関数を更新することによって，学習を高速化する．しかし，モデルに誤りがある場合や，システムが動的な場合には適切なモデルを獲得できないといった問題がある．

状態空間の設計問題

システムの状態空間が巨大または連続になると，システムの状態空間を単純にエージェントの内部状態空間に 1 : 1 で写像を行なった場合，エージェントの内部状態すべてを経験することが困難または不可能になる．そのため，学習に多大な時間を費す，あるいは多大な時間を費しても学習することができないという結果を生じる．この問題に対応するためには，内部状態空間を予め適切に設計する必要がある．しかし，

- 短時間で制御規則を獲得させるためには，エージェントの状態空間を低次元かつ粗く

設計する必要がある，

- 獲得する制御規則をより良いものにするためには，エージェントの状態空間を (必要最低限) 高次元かつ細かく設計する必要がある，

といったトレードオフの関係にあるため，状態空間を予め適切に設計することが難しいといった問題である．

これに対して，エージェントが自律的に，システムのいくつかの状態をまとめて一つの内部状態に写像する状態集約 (state aggregation) を行う手法 [16, 17, 18, 19, 20, 10, 21, 22, 23] が提案されているが，適用範囲に多くの制限があるなどの問題がある．

一方，状態価値関数をパラメータ表現することによって関数近似を行う線形最急降下法 (linear gradient-descent method) を用いて，内部状態をコンパクトにする手法 [24, 25, 26] が提案されている．しかし関数近似を用いると，収束性の保証が成り立たなくなること，多くの計算時間を必要とすることなどの欠点がある．

不完全知覚問題

エージェントの観測能力がタスクに対して十分でない場合に発生する問題として，不完全知覚問題 [3, 4] がある．また，エージェントはシステムの状態を観測するが，センサのノイズ等のため，いつも正しく観測できるとは限らない．観測が不完全であると異なる状態を同じ状態と認識する場合，またその逆が起きる場合は，学習の効率を著しく損なう可能性がある．このようなシステムにおける行動決定問題は POMDPs としてモデル化される．POMDPs とは，MDPs によりモデル化できるものとして扱われた対象システムとエージェントの部分観測系とを併せて学習器からみて形式化されたモデルである．具体的には，エージェントはシステムの状態を一意に決定できず，システムの状態が異なっていたとしてもそれらを区別することができない不完全知覚問題を含めて形式化されたモデルである．

これに対して，メモリベース法 (memory-based method)[27] は，エージェントの内部にシステムのモデルを作ることによって，システムがどの状態であるかを表す確率分布を求め，不完全知覚に接近する．しかしメモリの使用量が大きいことに起因して，対象システムの状態空間が大きくなると対応できないこと問題となる．

一方，確率的に行動を決定する確率的政策 (stochastic policy)[28, 29] は，限られたメモリや計算資源のもとで，オンラインかつリアルタイムにシステムに適応することを指向している．この特徴は，強化学習に適しており，メモリベース法の欠点を克服する方法として期待されているが，確率的に行動を決定するために結果的に無駄な行動を多くとってしまうという欠点がある．

今後，急速に複雑・大規模化している人工システムに対して強化学習が実用化されるためには，状態空間の設計問題および不完全知覚問題を解決することが重要課題であると考え，

またこれらの問題に取り組むことによって探索と搾取のトレードオフの解決，学習の高速化にも繋がると考え，本研究では複雑・大規模なシステムに対する実用化のため状態空間の設計問題および不完全知覚問題に取り組むことにする．

2.4 結言

本章では，強化学習方式の概要の説明を行い，現在強化学習方式が抱える多くの問題の中から盛んに研究がなされているものを取り上げ，その問題に対してそれぞれの代表的アプローチを示した．そして，本研究において状態空間の設計問題および不完全知覚問題に取り組むことを述べた．

第 3 章

状態フィルタリング

3.1 諸言

本章では，状態空間の設計問題，不完全知覚問題の解決に向けて，エージェントの観測空間と行動学習器の間に状態フィルタを導入した計算モデルを提案し，理想的状態フィルタを記述する．つぎに，MDPs の場合と POMDPs の場合について，状態フィルタを獲得する方法の違いという観点から従来手法の分類を行い，従来手法の構成面での特徴を明確にする．そして，内部状態を分割・統合する手法に焦点をあて，状態フィルタの基本的機能として，内部状態の分割・統合方法について 4 つの調整法を提案する．

3.2 状態フィルタリングの枠組み

ここでは，状態フィルタを導入した計算モデルを提案する．ここでは，システムの状態空間 $\boldsymbol{\mathcal{X}}$ から観測空間 $\boldsymbol{\mathcal{O}}$ へ多：1 の関係で写像 g が定義される．つまり，システムの状態空間 $x \in \boldsymbol{\mathcal{X}}$ のとき $o = g(x) \in \boldsymbol{\mathcal{O}}$ が観測される．ここで，MDPs の場合つまりエージェントがシステムの状態を完全に観測できる場合は，g は，$\boldsymbol{\mathcal{X}}$ から $\boldsymbol{\mathcal{O}}$ へ 1：1 または 1：多の関係で写像を行う．また，POMDPs の場合，エージェントの観測能力の限界などによって，g は $\boldsymbol{\mathcal{X}}$ から $\boldsymbol{\mathcal{O}}$ へ多：1：の関係で写像を行う．

従来の強化学習方式では，学習器に観測空間 $\boldsymbol{\mathcal{O}}$ から行動空間 $\boldsymbol{\mathcal{A}}$ への適切な写像を学習させる形で行動決定器を調整するといった枠組みが典型的である．これに対して，状態フィルタを導入したモデルでは，$\boldsymbol{\mathcal{O}}$ からエージェントの状態空間 $\boldsymbol{\mathcal{S}}$ (以下，内部状態空間) への写像 f と $\boldsymbol{\mathcal{S}}$ から $\boldsymbol{\mathcal{A}}$ への写像 y に分解され，写像 y の調整を学習器に担わせる．ここで,，$\boldsymbol{\mathcal{O}}$ から有為な観測状態のみを選択的に取捨するという意味で写像 f を「状態フィルタ」と呼ぶ．そして，POMDPs の場合は，f が履歴情報を記録・参照することで，POMDPs について局所的に MDPs への近似が行われる．つまり，あるステップ $t \geq 0$ における観測状態，内部状態をそれぞれ $o(t), s(t)$ とすると，

$$s(t) = \begin{cases} f(o(t), z(1), \cdots, z(t)) & (t > 0) \\ f(o(t)) & (t = 0) \end{cases} \tag{3.1}$$

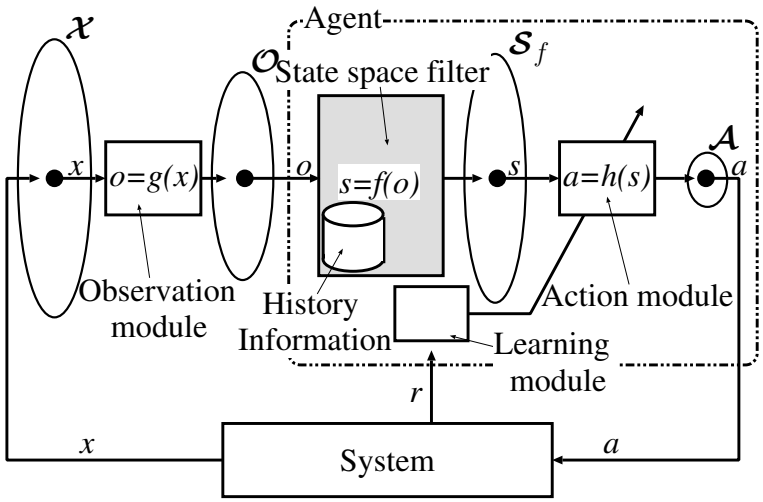

図 3.1: 状態フィルタを考慮した学習システムの枠組み

と表される．ただし，1 ステップは離散時間上の 1 つの時刻を，$z(t')$ は履歴情報に記録されている $t'(t \geq t' > 0)$ ステップ過去についての情報を表す．具体的には，従来手法において用いられる情報として，過去の観測状態，内部状態，行動などが挙げられ，例えば，

$$z(t') =< o(t-t'), s(t-t'), a(t-t') > \tag{3.2}$$

と表すことができる．ただし，$a(t)$ はステップ t における行動を表す．なお，図 3.1 に学習システムの枠組みを示す.

ここで，上記の枠組みを用いることで，不完全知覚問題を問題発生の原因によって，2 つのクラスに分けることができる．

(1) g による不完全知覚問題：$\boldsymbol{\mathcal{X}} \mapsto \boldsymbol{\mathcal{O}}$ において過剰に多：1 の関係で写像が行われることによって発生する．

(2) f による不完全知覚問題：$\boldsymbol{\mathcal{O}} \mapsto \boldsymbol{\mathcal{S}}$ において過剰に多：1 の関係で写像を行うことによって発生する．

ただし，不完全知覚問題発生時において，上記のどちらのクラスか特定することはできない．そのため，本論文では，はじめに f による不完全知覚問題と仮定して接近する．それでもなお問題解決ができない場合，g による不完全知覚問題と仮定して接近する．ここで，MDPs の場合，不完全知覚問題は上記 (2) の場合に発生することになる．

3.3　理想的状態フィルタ

3.2 節で述べたように，状態フィルタをシステムの状態空間 $\boldsymbol{\mathcal{X}}$ から内部状態空間 $\boldsymbol{\mathcal{S}}$ への写像 $f : \boldsymbol{\mathcal{X}} \rightarrow \boldsymbol{\mathcal{S}}$ を担うものと定義する．理想的な状態フィルタ f^* を，次の 3 つの要件を満

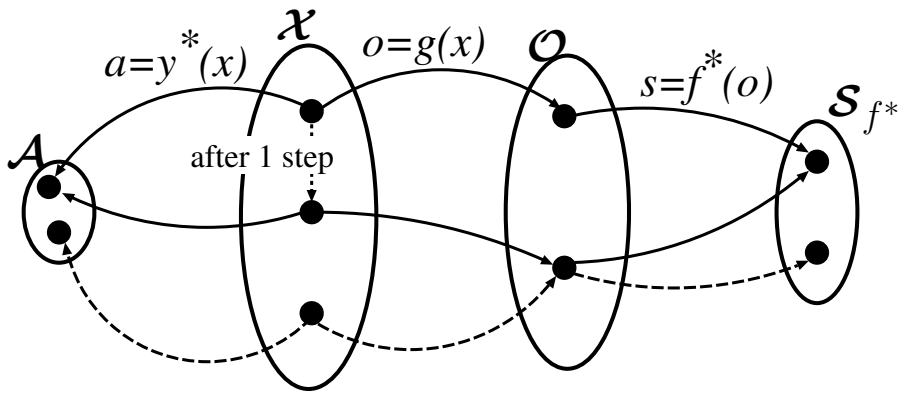

図 3.2: 理想的状態フィルタによる $\boldsymbol{\mathcal{O}}$ から $\boldsymbol{\mathcal{S}}$ への写像

たすものであると定義する．ただし，あるシステムの状態 x に対して最適な行動 a^* をかえす写像 $y^* : \boldsymbol{\mathcal{X}} \mapsto \boldsymbol{\mathcal{A}}$ が与えられていると仮定する．

(1) $x \in \mathcal{X}^i$ から写像される観測状態 $o = g(x)$ が，1 つの内部状態 $s \in \boldsymbol{\mathcal{S}}$ に写像される．ただし $\mathcal{X}^i = \{x | a_i = y^*(x)\}$ を表す．つまり，$y^*(x_i) = y^*(x_j) : (x_i, x_j \in \boldsymbol{\mathcal{X}}, x_i \neq x_j)$ のとき，$f^*(g(x_i)) = f^*(g(x_j))$ となる．

(2) 観測状態空間での状態遷移において直接 (次のステップにおいて) 遷移可能なある観測状態 $o_i, o_j \in \mathcal{O}^i$ が，1 つの内部状態 s に写像される．ただし $\mathcal{O}^i \subseteq \boldsymbol{\mathcal{O}}$ は，o_i から直接遷移可能な観測状態の集合 $\mathcal{O}^i = \{o(t+1) | o_i = o(t)\}$ を表す．つまり，$g(x_i), g(x_j) \in \mathcal{O}^i : (x_i, x_j \in \boldsymbol{\mathcal{X}}, x_i \neq x_j)$ のとき，$f^*(g(x_i)) = f^*(g(x_j))$ となる．

(3) $y^*(x)$ の異なるシステムの状態 $x_j, x_k \in \mathcal{X}_i$ からある単一の観測状態 o_i に写像される場合，履歴情報が用いられることで，異なる内部状態 s_j, s_k に写像される．ただし $\mathcal{X}_i = \{x | o_i = g(x)\}$ を表す．つまり，$y^*(x_i) \neq y^*(x_j) : (x_i, x_j \in \boldsymbol{\mathcal{X}}_k, x_i \neq x_j)$ のとき，$f(g(x_i), \cdots) \neq f(g(x_j), \cdots)$ とされる．

図 3.2 に，$f*$ による $\boldsymbol{\mathcal{O}}$ から $\boldsymbol{\mathcal{S}}$ への写像関係の様子を示す．ここで，要件 (1) によって，f による不完全知覚問題を解決し，要件 (2) によって，状態空間のコンパクト化を実現する．また，要件 (3) によって，g による不完全知覚問題を解決する．ただし，MDPs の場合，理想的状態フィルタ f^* は要件 (1),(2) によって定義される．

一般に $\boldsymbol{\mathcal{X}}$ から $\boldsymbol{\mathcal{A}}$ への最適な写像 y^* は既知なものではない．そのため，理想的なフィルタ f^* を獲得するためには，状態フィルタの調整と行動学習器についての学習を逐次的または同時並行的に行われる必要がある．

ここで，エージェントがシステムとの相互作用を通じて適切な状態フィルタを構成されていくことを状態フィルタの学習，行動決定器についての学習を行動学習と呼ぶことにする．また，状態空間の設計における要件，つまり，エージェントがシステムの状態を観測する情報と，その情報をどのくらいの粒度で観測するかで決定されるエージェントの状態空間につ

いて，

- 短時間で制御規則を獲得させるためには，エージェントの状態空間を低次元かつ粗く設計する必要がある，
- 獲得する制御規則をより良いものにするためには，エージェントの状態空間を (必要最低限) 高次元かつ細かく設計する必要がある，

といったトレードオフの関係にある要件をうまくバランスさせるという意味で，適切な状態フィルタを構成させる問題を，状態フィルタ獲得問題と呼ぶことにする．そして，状態フィルタの観点から状態空間の設計問題を捉え直すことで，従来研究されてきた適応的に内部状態空間が構成される手法を，適切な状態フィルタを獲得させる手法としてまとめて捉えることが可能となる．

ここで，状態フィルタの観点から，従来研究されてきた MDPs への接近法，POMDPs への接近法を，適切な状態フィルタを獲得する手法としてまとめて捉えることが可能となる．

3.4　従来の接近法の分類と問題点

3.4.1　MDPs の場合

MDPs における状態空間の (適応的) 構成に関する従来研究を状態フィルタの枠組みで捉えると，大まかに以下のような二つのグループに分類できる．

(1) 関数近似の手法を用いて状態フィルタを学習させる方法： 代表的なものに CMAC (Cerebellar Model Arithmetic Computer)[24]，RBF (Radial Basis Function) ネットワーク (RBF ユニットは固定)[8] といった方法が挙げられる．関数近似の手法を用いた方法は内部空間を連続として扱うため，観測状態空間も行動空間も連続である．そのため学習器において適用できる強化学習手法が限定される．観測状態空間と行動空間の対応が滑らかな (例えば，観測状態が少し変化する時に，行動も少しだけ変化するような) 場合，次のグループ (2) に分類される方法よりも良くなると期待できる．しかし，行動価値関数の収束性が保証されておらず [30]，事前知識などにより予め適切な状態フィルタを設計する必要がある．

(2) 内部状態を分割・統合することで状態フィルタを学習させる方法： 内部状態は基本的に離散である．そのため，行動空間が離散的なものである場合は，グループ (1) に分類される方法より有効であると考えられる．代表的なものに次の方法が挙げられる．

   a. 過去の行動結果のデータを保存し，行動結果の同一性／類似性に基づく方法 [17, 22, 18, 10, 31, 11]： オフラインで状態フィルタの学習を行わせる必要があるもの [17] や，状態空間の粒度を予め設定する必要のあるもの [22]，学習器において適用できる強化学習手法が特別なもの [31] や特定されているもの [18, 10, 11] である．

b. ART(Adaptive Resonance Theory) ニューラルネットワークや GNG(Growing Neural Gas Network) ネットワークを用いる方法[32, 33, 34]：多くのパラメータ設定が必要とされるもの[33, 34]や，多大な計算時間が必要とされるものである．

c. RBF ニューラルネットにおいて RBF ユニットを追加・分割・削除する方法[25, 26]：ある関数近似の手法を用いて内部状態を分割・統合する方法は内部空間を連続として扱うことができるが，多大な計算時間が必要とされる．また目標とする状態フィルタは価値関数を完全に近似できるフィルタであり，本論文の理想的状態フィルタとは異なる．

d. 複数の学習器に対する方法[35, 36, 37]：基本的に対象がマルチエージェント問題とされている．

以上より，グループ (1) および (2) に分類される方法ともに，学習器において適用できる強化学習手法が特別なものや特定されているものがほとんどで，またすべての方法において計算時間が考慮されていない．そこで，本論文では MDPs において，上記の課題を踏まえて，より計算時間がかからない内部状態を分割・統合する手法に焦点を当てる．また，行動空間については，従来手法と同様に離散的なものを考えることにする．

3.4.2 POMDPs の場合

POMDPs への従来研究を状態フィルタの枠組みで捉えると，大まかに以下のような 4 つのグループに分類できる．

(1) 履歴情報を用いて局所的に MDPs への近似が行われる状態フィルタ[3, 38, 27, 39, 40, 41]：定性的な考察によって得られた判断基準が用いられて識別が行われるため，多くのメモリを必要としていない手法[41]もある．しかし，一般的に履歴情報の収集に膨大な行動とメモリを必要とする．

(2) ニューラルネットワークが用いられる状態フィルタ[5, 6, 42, 43]： リカレントニューラルネットワークが用いられることによって，対象システムが連続状態空間を有する場合において有効である手法[5, 6]もある．しかし，ニューラルネットワークを用いるため，多大な計算時間を必要とし，また，学習結果がブラックボックス的である．

(3) 階層型構造が用いられる状態フィルタ[44, 45, 46, 47, 48]： 学習器に用いられる強化学習手法が限定されない手法である．しかし，設計者が事前知識などを用いて階層構造を決定しておく必要がある．

(4) $\boldsymbol{\mathcal{O}}$ から $\boldsymbol{\mathcal{S}}$ へ 1 : 1 に写像が行われる状態フィルタ[49, 29, 50]： 確率的な制御ルールが学習器において獲得される．履歴情報が用いられないため，多くのメモリを必要としない手法である．しかし，確率的な制御ルールが獲得されるため，決定的制御ルールが獲得されることはなく，必要以上に多くの行動を要してしまう場合がある．

以上より，リカレントニューラルネットワークが用いられる手法 [5, 6] を除くと，対象システムが連続状態空間を有する場合に有効である方法はほとんどなく，また，すべての方法において，状態空間のコンパクト化や計算時間が考慮されていない．そこで，本論文では POMDPs において MDPs と同様，上記の課題を踏まえて，ニューラルネットワークを用いるよりも計算時間がかからない内部状態を分割・統合する手法に焦点を当てる．また，行動空間については，従来手法と同様に離散的なものを考えることにする．

3.5　状態フィルタの調整法

本論文では，理想的状態フィルタの 3 つの要件を満たすために，計算時間のかからない内部状態を分割および統合する手法を用いることによって，状態フィルタの調整を図ることを考える．内部状態の分割・統合方法として，履歴情報を用いない場合と用いる場合で，以下のような 4 つの調節法を考える．また，MDPs の場合，履歴情報を用いる必要はなく，通常分割と統合のみ行うことにする．

通常分割

ある状態フィルタ f が与えられているとして，

$$\mathcal{O}_i = \mathcal{O}_j \cup \mathcal{O}_k, \quad \mathcal{O}_j \cap \mathcal{O}_k = \phi \tag{3.3}$$

ただし，

$$\mathcal{O}_i = \{o | s_i = f(o), o \in \boldsymbol{\mathcal{O}}\} \tag{3.4}$$

を満たす $\mathcal{O}_j$, $\mathcal{O}_k$ を異なる内部状態 s_j，s_k へ写像が行われるように状態フィルタを調整することを，内部状態 s_i の通常分割とする．図 3.3 に内部状態に対する分割・統合の概要を示す．ここで，結果的に内部状態空間のサイズは大きくなり，s_i が内部状態空間から削除される．

通常分割は，理想的状態フィルタの要件 (1), (2) を満たすために，適宜行われる．

統合

状態フィルタ f が与えられているとして，式 (3.3)，ただし，

$$\mathcal{O}_j = \{o | s_j = f(o), o \in \boldsymbol{\mathcal{O}}\} \tag{3.5}$$

$$\mathcal{O}_k = \{o | s_k = f(o), o \in \boldsymbol{\mathcal{O}}\} \tag{3.6}$$

を満たす $\mathcal{O}_i$ を同一の内部状態 s_i に写像が行われるように状態フィルタを調整することを，内部状態 s_j, s_k $(s_j \neq s_k)$ の統合とする (図 3.3 参照)．ここで，s_j, s_k が内部状態空間から削除されることにより，結果的に，内部状態空間のサイズが小さくなる．

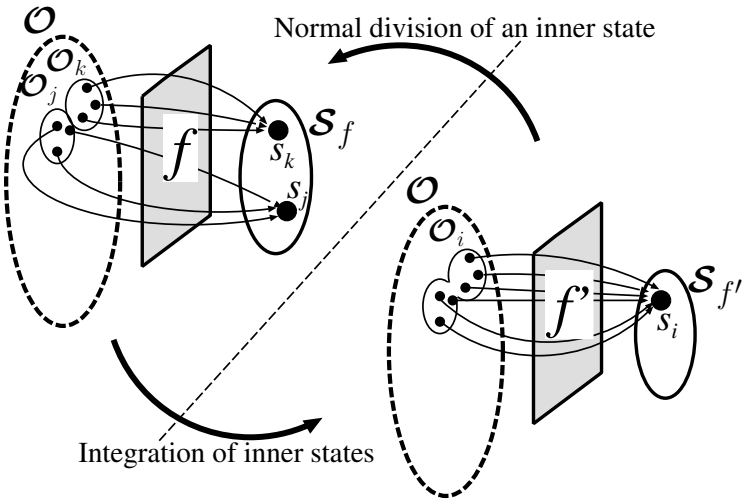

図 3.3: 内部状態に対する分割・統合の概要

統合も通常分割と同様，理想的状態フィルタの要件 (1), (2) を満たすために，適宜行われる．

履歴参照分割

ある状態フィルタ f が与えられているとして，あるステップ t において，$d \geq 0$ ステップ過去までの履歴情報を用いて写像されるある内部状態 s_i

$$s_i = f(o(t), z(1), \cdots, z(d)) \tag{3.7}$$

に対して，さらに 1 ステップ過去の履歴情報を用いて，異なる内部状態 s_j

$$s_j = f(o(t), z(1), \cdots, z(d), z(d+1)) \tag{3.8}$$

に写像が行われるように状態フィルタを調整することを， 内部状態 s_i の履歴参照分割とする．ここで，s_i と s_j では，s_j を優先して写像することにする．そして，結果的に内部状態空間のサイズは大きくなる．

履歴参照分割は，理想的状態フィルタの要件 (3) を満たすために，適宜行われる．

削除

あるステップ t において，$d > 0$ ステップ過去までの履歴情報をもちいて写像されるある内部状態 s_i (式 (3.7)) を内部状態空間から削除されるように状態フィルタを調整することを，内部状態 s_i の削除とする．ここで，s_i が内部状態空間から削除されることにより，結果的に，内部状態空間のサイズが小さくなる．

削除は，必要以上に履歴参照分割を行っている場合に，適宜行われる．

3.6　結言

本章では，状態空間の設計問題，不完全知覚問題の解決に向けて，エージェントの観測空間と行動学習器の間に状態フィルタを導入した計算モデルを提案し，理想的状態フィルタについても示した．つぎに，MDPs の場合と POMDPs の場合について，状態フィルタを獲得する方法の違いという観点から従来手法の分類を行い，従来手法の構成面での特徴を明確にした．そして，内部状態を分割・統合する手法の状態フィルタの基本的機能として，内部状態の分割・統合方法について通常分割・統合・履歴参照分割・削除の 4 つの調整法を提案した．

第 4 章

MDPsでの状態フィルタの適応的獲得手法

4.1 諸言

本章では，強化学習問題の中でタスクを遂行するために十分な情報が与えられているMDPsの問題を対象とする．

3.4.1 節にて，MDPs の場合への従来手法の問題点を次のように示した．

(1) ほとんどの手法において，学習器に適用できる強化学習手法が特別なものや特定されているものである，

(2) すべての方法において，計算時間が考慮されていない．

そこで，本章では MDPs において，上記の課題を踏まえて，より計算時間がかからない内部状態を分割・統合する手法に焦点を当てた状態フィルタを考慮することで，学習器に適用できる強化学習手法が限定されない手法を考える．そして，与えらえた情報に対して適切な状態フィルタの調整を試みており，これを実現する手法として，エントロピーを用いた状態フィルタの一実現法を提案する．さらに，離散状態空間を有する迷路問題および連続状態空間を有するロボットナビゲーション問題を例題として取り上げ，計算機実験を通して，提案手法を従来手法と比べることで，提案手法の有効性・可能性について検討する．

4.2 状態フィルタの適応的獲得手法

4.2.1 基本的考え方

内部状態におけるボルツマン選択法を用いた行動選択確率についてのエントロピーの和：

$$\phi(f) = \sum_{s \in \boldsymbol{\mathcal{S}}_f} H(s) \tag{4.1}$$

および内部状態空間のサイズ $|\boldsymbol{\mathcal{S}}_f|$ を最小化することを目標として状態フィルタが適応的に獲得される．ここで $\boldsymbol{\mathcal{S}}_f$ は状態フィルタ f が構成する内部状態空間を表す．また，$H(s)$ はある内部状態 s でのエントロピー：

$$H(s) = -\frac{1}{\log |\boldsymbol{\mathcal{A}}(s)|} \sum_{a \in \boldsymbol{\mathcal{A}}(s)} p(a|s) \log p(a|s) \tag{4.2}$$

ただし $p(a|s)$ は内部状態 s で行動 a を選択する確率，$\boldsymbol{\mathcal{A}}$ は行動空間，$|\boldsymbol{\mathcal{A}}(s)|$ は s に写像された際に取り得る行動数を表す．このような状態フィルタ実現方法として状態フィルタの学習と行動学習を異なる時定数で実行される手法も考えられるが，ここでは状態フィルタの学習と行動学習を同じ時定数で行われるものとする．

式 (4.2) で示される $H(s)$ は，エントロピーと呼ばれる乱雑さの度合を示す指標を正規化したものである．これを伊藤ら [51] は学習残エントロピーと呼び，粗く量子化された内部状態空間から，細かく量子化された内部状態空間への切替えに学習残エントロピーの平均を用いていた．

ここではこのエントロピー $H(s)$ が，一つの内部状態についての状態集約の正しさを評価する指標として捉えられることで，状態フィルタの調整が行われる．具体的には，観測状態空間が粗く離散化され，内部状態空間へ写像されると不完全知覚問題が発生し，その場合にはエージェントが出力すべき行動が一意に決まらない．よって分割すべき内部状態では上記のエントロピーは小さくならない．これを利用して，ある内部状態において十分な回数行動学習をしているにも関わらずエントロピーが小さくならなければ，その内部状態において不完全知覚問題が発生しているとし，その内部状態が分割されるように状態フィルタが調整される．

また，ある内部状態においてエントロピーが小さく，かつ遷移先の観測状態から写像される異なる内部状態においてもエントロピーが小さく，さらに互いの内部状態における代表行動が同じであれば，必要以上に内部状態の分割が行われているとし，これらの内部状態が統合されるように状態フィルタが調整される．ただし，システムが動的である場合に，状態フィルタの調整が繰り返されることも考慮して，統合後の内部状態に写像される観測状態の範囲が超直方体領域である場合にのみ内部状態が統合されるように状態フィルタが調整される．ここで，内部状態における代表行動とは，その内部状態において最も選択確率の高い行動をさす．

以上，行動空間について離散なものを前提としていたが，行動空間が連続である場合は，

(1) 式 (4.2) において和ではなく積分をとる形に変更する，

(2) 行動空間を細かな刻み幅で分割するなどの対応によって代表行動を定義する，

といった対応によって，適用が可能であると考えられる．

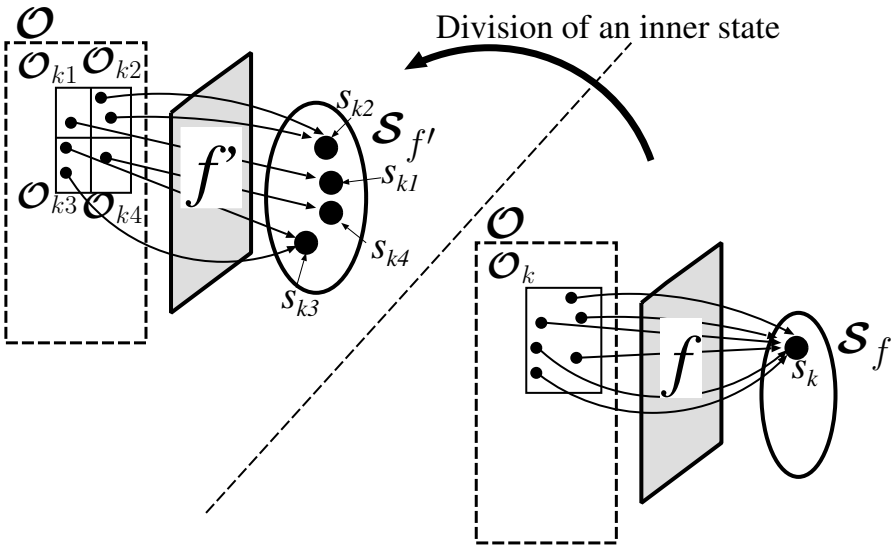

図 4.1: 提案手法による内部状態の分割

4.2.2 状態フィルタの調整法

内部状態の通常分割法

ある内部状態 s における行動学習回数を $L(s)$ とし，$L(s) > \theta_{\mathrm{L}}$ かつ $H(s) > \theta_{\mathrm{H}}$ であれば，内部状態 s に写像される観測状態の範囲を各次元ごと中心で 2 分割され，それぞれ異なる内部状態に写像されるように状態フィルタが調整される．図 4.1 に内部状態に対する通常分割の様子を示す．ただし，θ_{L} は内部状態に関する行動学習回数についての閾値，θ_{H} はエントロピーについての閾値を表す．この操作により，分割後の内部状態空間のサイズは，観測状態の次元数を N とすれば，$(2^N - 1)$ だけ増加する．

なお，新たに生成された 2^N 個の内部状態の評価値は，通常分割前の内部状態の評価値とする．

内部状態の統合法

ある観測状態から写像される内部状態 s，次に遷移した観測状態から写像される内部状態 s' とすると，$s \neq s'$ のとき，$H(s) \leq \theta_{\mathrm{H}}$ かつ $H(s') \leq \theta_{\mathrm{H}}$ で，$a(s)^+ = a(s')^+$ であれば，s と s' を一つの内部状態に統合されるように状態フィルタが調整される．また，ある期間 θ_{t} 一度も写像されない内部状態 s が存在するとき，s とその隣合う (具体的には，写像される観測状態の範囲が隣接する) 内部状態を一つの内部状態に統合されるように状態フィルタが調整される．ここで，$a(s)^+$ は内部状態 s における代表行動を表し，θ_{H} は分割法に用いられたものと同じ値が用いられる．統合後の内部状態空間のサイズは，1 だけ減少する．ただし，

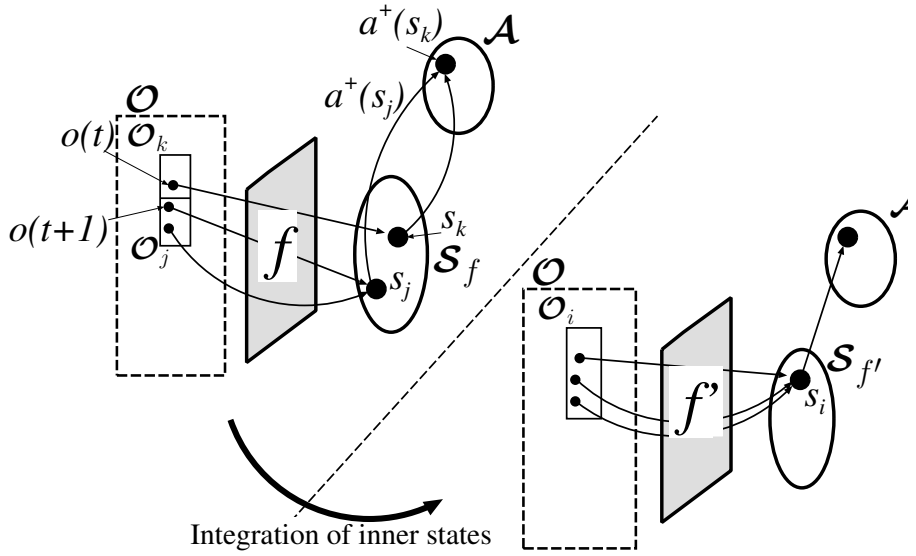

図 4.2: 提案手法による内部状態の統合

統合が行われるのは，統合後の内部状態に写像される観測状態の範囲が超直方体領域となる場合に限る．図 4.2 に内部状態に対する統合の概要を示す.

なお，統合に伴い新たに生成された内部状態の評価値は，統合前の 2 つの内部状態における評価値の平均とする.

状態フィルタを調整するための手順を以下にまとめておく.

(1) サイズ 1 の内部状態空間から，一回分割を行った後の内部状態空間のサイズ 2^N から学習を始める．ここで N は観測状態の次元数を表す.

(2) 観測状態 $o(t)$ を観測する.

(3) 観測状態 $o(t)$ から状態フィルタによって内部状態 $s(t)$ を得る．そして $s(t-1)$ における行動学習を実行し，$s(t-1)$ における行動学習回数について $L(s(t-1)) \leftarrow L(s(t-1))+1$ と更新する.

(4) 内部状態 $s(t-1)$ における行動学習によって変化したエントロピー $H(s(t-1))$ を更新し，$s(t)$ におけるエントロピー $H(s(t))$ を計算する.

(5) 以下の条件に従って，通常分割・統合を行う.

   a. もし $L(s(t-1)) > \theta_{\mathrm{L}}$ かつ $H(s(t-1)) > \theta_{\mathrm{H}}$ であれば，内部状態 $s(t-1)$ において通常分割を実行する.

   b. もし $s(t-1) \neq s(t)$ の場合，$H(s(t-1)) \leq \theta_{\mathrm{H}}$ かつ $H(s(t)) \leq \theta_{\mathrm{H}}$ かつ $a(s(t-1))^+ = a(s(t))^+$ であれば，$s(t-1), s(t)$ において統合を実行する．ただし，統合を行うのは，統合後の内部状態に写像される観測状態の範囲が超直方体となる場合に限る.

(6) もし (5) によって状態フィルタを変更していれば，再度 $o(t)$ から状態フィルタによって内部状態 $s(t)$ を得る．

(7) 内部状態 $s(t)$ から行動空間への写像を行い，行動を実行する．

(8) もし前回一度も写像されていない内部状態を統合してから，ある期間 θ_{t} 経過していれば，再度一度も写像されていない内部状態を統合する．ただし，統合を行うのは，統合後の内部状態に写像される観測状態の範囲が超直方体となる場合に限る．

(9) $t \leftarrow t+1$ と更新して，(2) に戻る．

4.3 計算例および考察

4.3.1 離散状態空間

例題の設定

図 4.3 に示す 20×20 格子空間においてエージェントにスタート地点からゴール地点までのパスを学習させる迷路問題を設定する．ここでは，エージェントがゴール地点に到着した時点でのみ環境から報酬 10 を得る．その他の場合には，一切報酬を得ることはない．エージェントの行動は上下左右の 4 種類で 1 マス移動する．黒いマス目で示された壁に移動することは許されず，壁へ向かう行動をとっても，行動後の環境の状態は，行動前の環境の状態と変化しない．1 ステップは，エージェントが 1 回行動を取り終えるまでとする．1 エピソードは，エージェントがゴール地点に到着し報酬が与えられるまでとする．エピソードが終るごとに，エージェントは初期状態に再配置される．

学習エージェント

提案手法 (以下” 手法 A” とする)，増尾ら [19] の手法 (以下” 手法 B” とする)，深尾ら [10] の手法 (以下” 手法 C” とする)，そして単純に観測状態から内部状態空間へ 1：1 に写像する状態フィルタを用いた手法 (以下” 手法 D” とする) との比較実験を行った．ここで，手法 B は，SOM(Self Organization Map) を用いた観測状態の訪問頻度を利用する．そして，状態フィルタの学習と行動学習が同時並行的に行われ，かつ学習器に適用する強化学習手法が限定されないという特徴をもった手法である．手法 C は 2 つの観測状態要素についてそれぞれ異なる内部状態に写像され独立に学習を行う．そして，状態フィルタの学習と行動学習が同時並行的に行われ，学習器に適用する強化学習手法は Q-learning に限定された手法である．

行動学習はすべての手法について同じパラメータ (学習率 $\alpha = 0.1$，割引率 $\gamma = 0.9$) の Q-learning(行動選択確率は温度係数 $\tau = 0.1$ のボルツマン選択法) を用いた．また，手法 A において，行動学習回数の閾値 $\theta_{\mathrm{L}} = 100$，エントロピーの閾値 $\theta_{\mathrm{H}} = 0.3$，内部状態の統合

を行うある期間 $\theta_{\mathrm{t}} = 100$[episode] を，手法 B において，SOM に使われる学習率の初期値 $\alpha_{\mathrm{SI}} = 0.3$，$\alpha_{\mathrm{SI}}$ の割引率 $\mu_{\mathrm{S}} = 0.999$，分散値 $\sigma_{\mathrm{S}} = 1.0$， SOM の学習率の閾値 $\theta_{\alpha \mathrm{S}} = 0.05$，統合の閾値 $\epsilon_{\mathrm{T}} = 5.0$ を，手法 C において，判断の基準値 $\beta = 0.09$ を用いた．ここでエントロピーの閾値は，1 つの行動選択確率が 0.9 の時のエントロピーの最大値が約 0.312 であることを参考に設定した．

つぎに，状態フィルタリングの観点から学習速度と解の良さ，内部状態空間のサイズ，ならびに計算時間の項目に関して比較検討を行い，最後に手法 A により獲得された状態フィルタについて検討する．

性能の比較および評価

実験結果を図 4.4，図 4.5，表 4.1 に示す．なお，

- 図 4.4 に各エピソード完了に要したステップ数の変化,
- 図 4.5 に内部状態空間のサイズの変化,
- 表 4.1 に 1 回の実験に要した計算時間 (Intel Pentium4 3.06GHz, メモリ 1GB, Vine Linux 3.2, java 1.5.0 使用時),

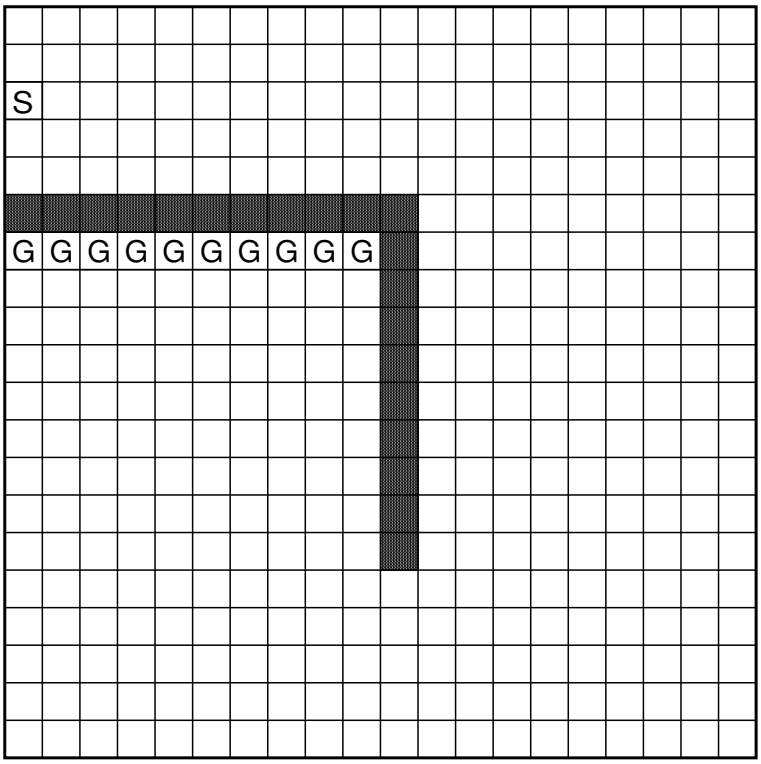

図 4.3: 迷路設定

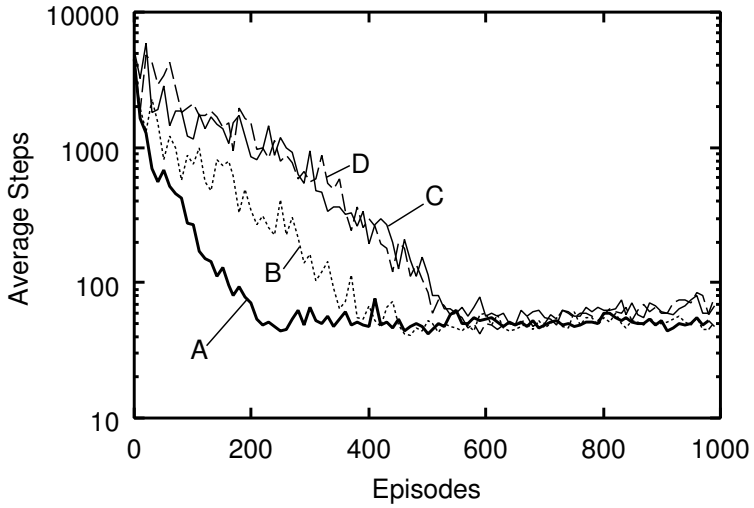

図 4.4: 所要ステップ数の変化

表 4.1: 計算時間

Method	Computational time[s]
A	1.06
B	34.64
C	3493.71
D	1.74

を示す．ここで，実験結果はすべて 20 回の実験の平均である．

各検討項目に対して，次のようなことが確認できる．

- 検討項目 (1)：学習速度と制御ルールの良さについて図 4.4 より，
  - 学習速度において，手法 A が他の手法より良い性能を示していること，
  - 獲得された制御ルールにおいて，すべての手法が同じくらいの性能であること，
- 検討項目 (2)：内部状態空間のサイズについて図 4.5 より，
  - 手法 A が他の手法より小さいこと，
- 検討項目 (3)：計算時間にいて表 4.1 より，
  - 手法 A が他の手法より短いこと．

ここで，手法 A により獲得された状態フィルタの一例を図 4.6 に示す．図 4.6 より手法 A により獲得された状態フィルタは，一部細かく分割されているものの，大部分において理想的な状態フィルタが獲得されていることが確認できる．

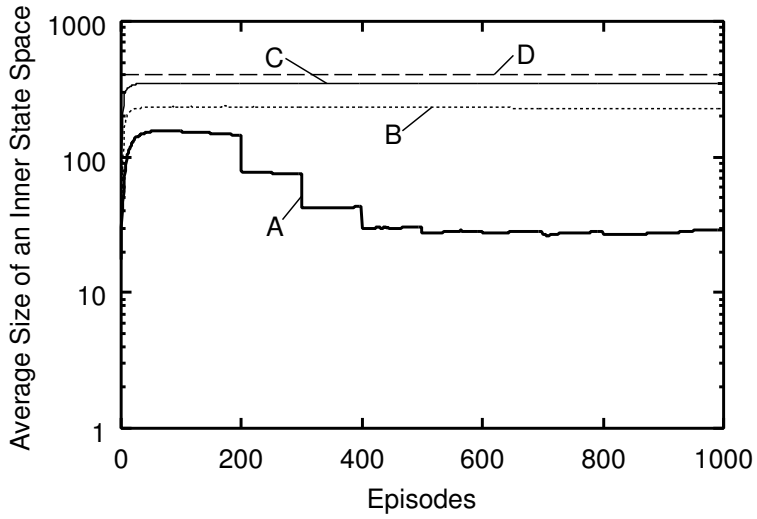

図 4.5: 内部状態空間のサイズの変化

4.3.2　連続状態空間

例題の設定

提案手法の有効性を検討するため，計算機シミュレーションによる実験を行う．壁に囲まれた 1000[mm] ×1000[mm] の連続空間において，空間の中心 $(x, y) = (500, 500)$ に半径 15[mm] の目標物を設置する．そこで，半径 25[mm] の円筒型のエージェントにスタート地点からエージェントの前面が目標物に到達するまでの行動系列を獲得させる例題に適用する．エージェントは，目標物に到着した時点でのみ環境から報酬 10 が与えられる．エージェントは，出力として 2 輪の車輪を持ち，両輪の回転速度を制御することによって，前進 [forward]，左 (右) 前進 [left(right) forward]，左 (右) 回転 [left(right) rotate] の 5 種類の行動が可能であるとする．図 4.7 に，各行動における両輪の回転速度比を示す．具体的には，前進行動で 15[mm] 進み，左 (右) 前進行動時には，回転しながら進み，結果的に ±0.1[rad] 回転する．左 (右) 回転行動時には移動せず，±1.0[rad] 回転する．エージェントが 1 回の行動を取り終えるまでを 1 ステップとし，エージェントが目標物に到着し報酬が与えられるまでを 1 エピソードとする．

エージェントの設定

例題 (1)：2 次元状態

- 観測状態：＜エージェントの中心から目標物の中心までの距離 d_1[mm]$(0 \leq d_1 \leq 750)$，エージェントの進行方向から目物の中心までの角度 θ_1[rad]$(-\pi \leq \theta_1 \leq \pi)$ ＞ の 2 次元とする．

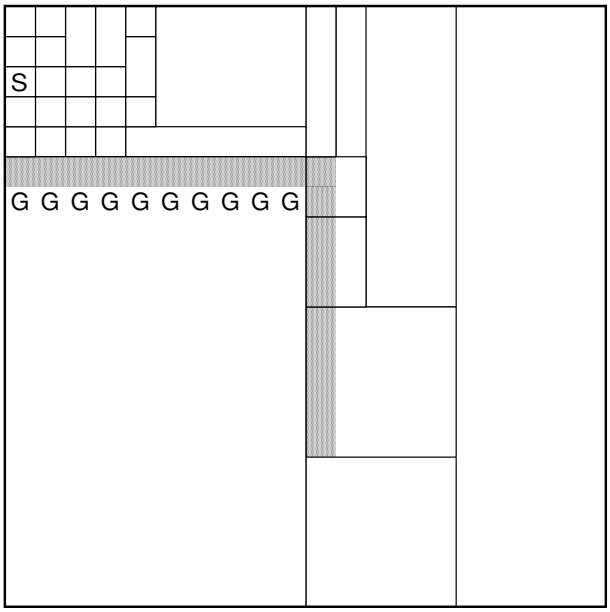

図 4.6: 手法 A により獲得された状態フィルタの一例

- エピソード開始時：スタート地点・進行方向がランダムに決定され，エージェントが再配置される．

例題 (2)：4 次元状態

中心を $(x, y) = (650, 650)$ とする一辺の長さ 100[mm] の正方形となる障害物を配置する．

- 観測状態：＜エージェントの中心から目標物の中心までの距離 d_1[mm]$(0 \leq d_1 \leq 750)$，エージェントの進行方向から目標物の中心までの角度 θ_1[rad]$(-\pi \leq \theta_1 \leq \pi)$，障害物の中心までの距離 d_2[mm]$(0 \leq d_2 \leq 750)$，障害物の中心までの角度 θ_2[rad]$(-\pi \leq \theta_2 \leq \pi)$ ＞ の 4 次元とする．図 4.8 に入力情報の概要を示す．
- エピソード開始時：中心 $(x, y) = (850, 850)$ とする一辺の長さ 250[mm] の正方形となるスタートエリア内からスタート地点・進行方向がランダムに決定され，エージェントが再配置される．

性能の比較および評価

提案手法 (以下” 手法 A” とする)，半田ら [32] に基づいた手法 (以下” 手法 B” とする)，そして内部状態空間をそれぞれの次元均等に $2, 4, 8$ 分割したグリッド分割法 (以下，それぞれ” 手法 2”,” 手法 4”,” 手法 8” とする) との比較実験を行なった．手法 B は，状態フィルタの学習と行動学習が同時並行的に行われ，かつ学習器に適用する強化学習手法が限定されないと

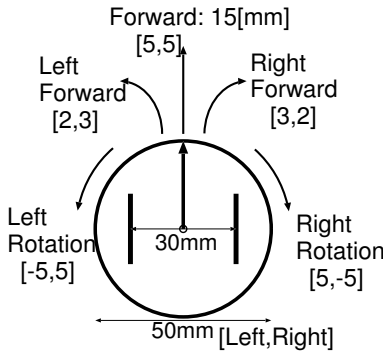

図 4.7: エージェントの行動

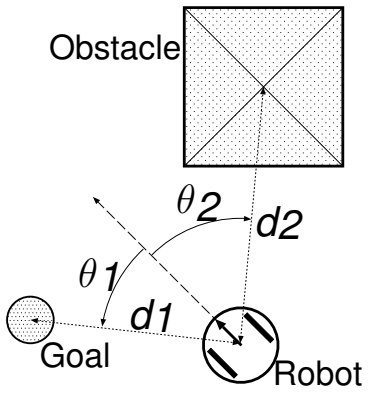

図 4.8: エージェントへの入力情報

いう特徴をもった手法である．ここで，半田ら [32] の手法をそのまま用いた場合，入力状態を決定する各状態要素の取り得る値の範囲が異なる場合，取り得る値の範囲が大きい状態要素ほど，内部状態空間の分割に関与することになる．これを防ぐために各状態要素を $[0, 1]$ に正規化した値を手法 B の観測状態とした．

行動学習はすべての手法について同じパラメータ (学習率 $\alpha = 0.1$，割引率 $\gamma = 0.9$) の Q-learning(行動選択確率は温度係数 $\tau = 0.1$ のボルツマン選択法) を用いた．また，手法 A において，行動学習回数の閾値 $\theta_{\mathrm{L}} = 1000$，エントロピーの閾値 $\theta_{\mathrm{H}} = 0.3$，内部状態の統合を行うある期間 $\theta_{\mathrm{t}} = 100$[episode] を，手法 B において，選択強度を求める正のパラメータ $\alpha = 1.0$，$\gamma = 0.1$，警戒変数 $\rho = 0.95$ を用いた．ここでエントロピーの閾値は，1 つの行動選択確率が 0.9 の時のエントロピーの最大値が約 0.288 であることを参考に設定した．

状態フィルタリングの観点から学習速度と解の良さ，内部状態空間のサイズ，ならびに計算時間の項目に関して比較検討を行い，最後に例題 (1) において手法 A により獲得された状態フィルタについて検討する．

実験結果を図 4.9〜図 4.12，表 4.2，表 4.3 に示す．なお，

- 図 4.9，図 4.11 にはそれぞれ例題 (1)，例題 (2) における各エピソード完了に要したステップ数の変化，
- 図 4.10，図 4.12 にはそれぞれ例題 (1)，例題 (2) における内部状態空間のサイズの変化，
- 表 4.2，表 4.3 にはそれぞれ例題 (1)，例題 (2) における 1 回の実験に要した計算時間 (Intel Pentium4 3.06GHz, メモリ 1GB, Fedra Core release 3, java 1.5.0 使用時)，

を示す．ここで，実験結果はすべて 20 回の実験の平均である．

各検討項目に対して，次のようなことが確認できる．

- 検討項目 (1)：学習速度と制御ルールの良さについて図 4.9 より例題 (1) では，
  - 学習速度において，手法 A，B および手法 4 は，他の手法より良い性能を示して

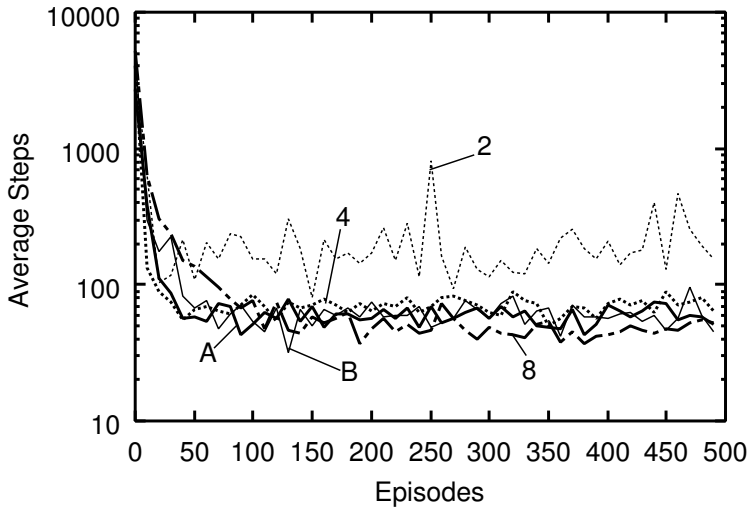

図 4.9: 例題 (1) における所要ステップ数の変化

表 4.2: 例題 (1) における計算時間

Method	Computational time[s]
A	0.530
B	0.743
2	0.708
4	0.300
8	0.354

いること，

– 獲得された制御ルールにおいて，手法 8 が最も良く，手法 A，B，4 は，お互い同じくらいの性能であること，

また，図 4.11 より例題 (2) では，

– 学習速度において，手法 A，B，および手法 4 は，他の手法より良い性能を示していること，

– 獲得された制御ルールにおいて，手法 A が他の手法より良い性能を示していること，

- 検討項目 (2)：内部状態空間のサイズについて図 4.10 より例題 (1) では，

– 手法 A が手法 B より大きいこと，

図 4.12 より例題 (2) では，

– 手法 B が手法 A より大きいこと，

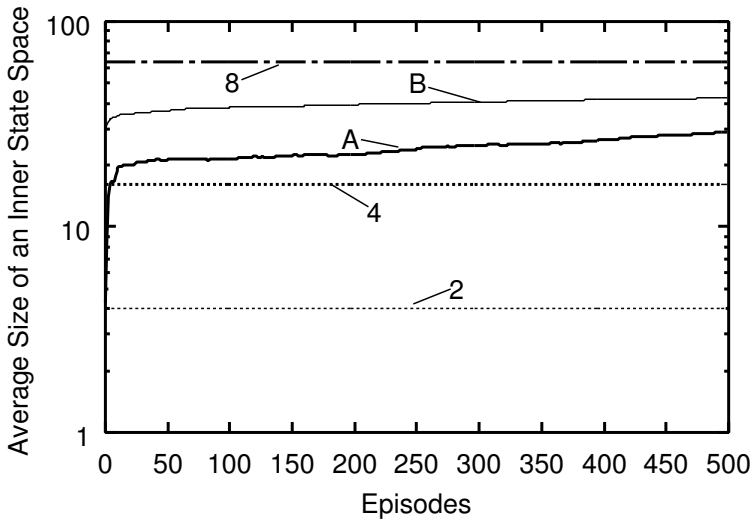

図 4.10: 例題 (1) における内部状態空間のサイズの変化

表 4.3: 例題 (2) における計算時間

Method	Computational time[s]
A	77.42
B	125.77
2	320.76
4	29.06
8	48.14

- 検討項目 (3)：計算時間について表 4.2，表 4.3 より，
  - 手法 A が手法 B より計算時間をより少なく抑えられること．

ここで，例題 (1) において手法 A により獲得された状態フィルタの一例を図 4.13 に示す．図 4.13 より手法 A により獲得された状態フィルタは，一部細かく分割された部分も確認できるが，大方各次元均等に 4 分割された状態フィルタが獲得されていることが確認できる．

以上より，

(1) 例題 (2) において手法 B は，手法 A より内部状態空間のサイズを小さく抑えれているものの，獲得された制御ルールの質が手法 A より悪いことから，状態フィルタがさらに細かく構成される必要があることが分かる．したがって，手法 A が手法 B より少ない計算時間で，良い状態フィルタが獲得されていると判断できる．

(2) グリッド分割法に関し，例題 (1) において手法 8 により最も良い制御ルールが獲得されているものの，学習速度，内部状態空間のサイズ，計算時間を考慮すると，手法 4 に

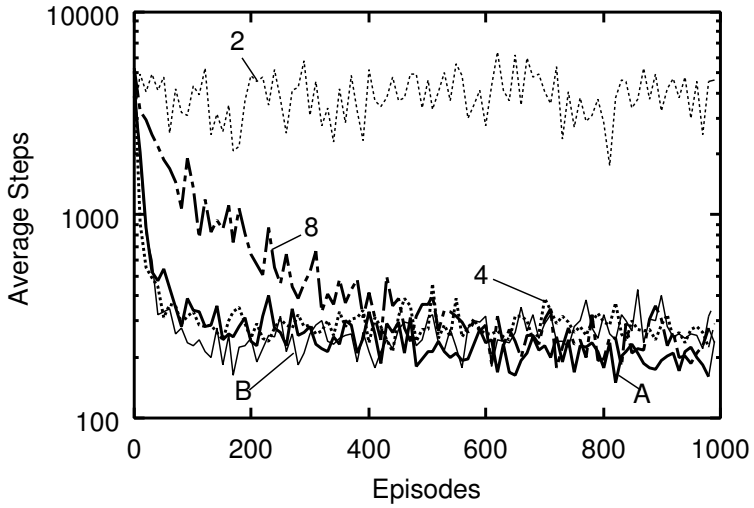

図 4.11: 例題 (2) における所要ステップ数の変化

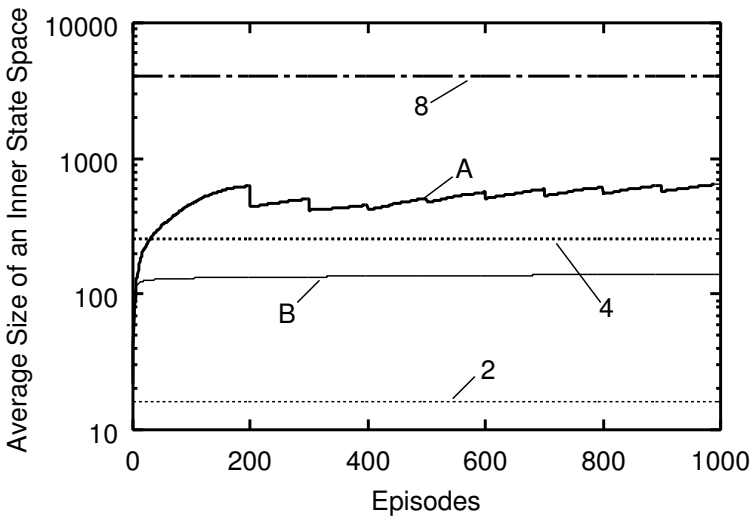

図 4.12: 例題 (2) における内部状態空間のサイズの変化

より最も良い状態フィルタが獲得されていると判断することができる．また，グリッド分割法で最も良い状態フィルタが獲得されている手法 4 よりも学習速度が速いことより，学習段階に応じて良い状態フィルタが獲得されていると考えられる．これらより，手法 A がより複雑かつ大規模な対象システムに対しても有効であると考えられる．

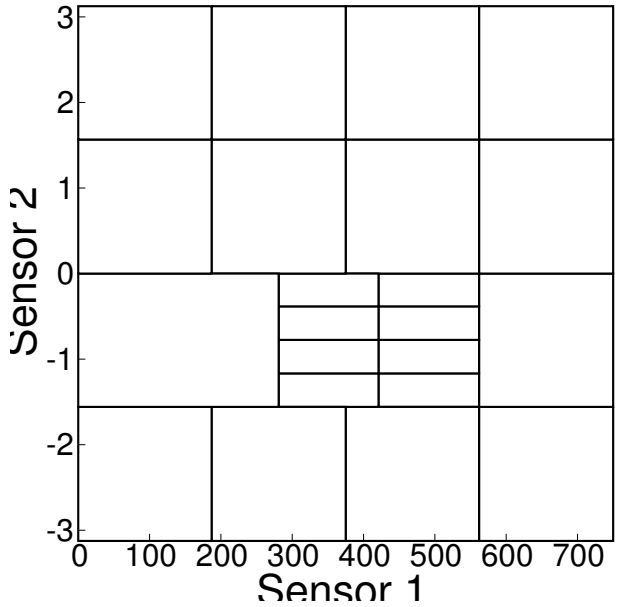

図 4.13: 例題 (1) において手法 A により獲得された状態フィルタの一例

4.4　結言

本章では，状態空間の設計問題に焦点をあて，タスクを遂行するために十分な情報が与えられている MDPs の問題を対象として，エントロピーを用いた状態フィルタの一実現法を提案した．この際，ある内部状態における行動選択確率のエントロピーを，その内部状態についての状態集約の正しさを評価する指標として用いた．また，この手法は，状態フィルタの学習と行動学習を同時並行して行わせることができ，学習器において適用できる強化学習手法が限定されないという特徴をもっている．

さらに，強化学習問題の例題として離散状態空間を有する迷路問題および連続状態空間を有するロボットナビゲーション問題を取り上げ，計算機実験を通して提案手法を従来手法・グリッド分割法と比べることで，提案手法の有効性・可能性を，それぞれ確認した．

今後の課題として，提案手法について (1) 観測状態の中から不要な状態要素を見つけ出すことで，状態空間の低次元化を実現できる手法への拡張，(2) パラメータ (行動学習回数に関する閾値 θ_{L}) の適応的な調節法，などが挙げられる．

第 5 章

POMDPsでの状態フィルタの適応的獲得手法

5.1 諸言

本章では，強化学習問題の中でタスクを遂行するために十分ではない部分的な情報が与えられている POMDPs の問題を対象とする．

3.4.2 節にて，POMDPs の場合への従来手法の問題点を次のように示した．

(1) リカレントニューラルネットワークが用いられる手法[5, 6]を除くと，対象システムが連続状態空間を有する場合に有効である方法はほとんどない．

(2) すべての方法において，状態空間のコンパクト化や計算時間が考慮されていない．

そこで，本章では POMDPs において MDPs と同様，上記の課題を踏まえて，ニューラルネットワークを用いるよりも計算時間がかからない内部状態を分割・統合する手法に焦点を当てる．そして，状態空間のコンパクト化を可能とする適切な状態フィルタの調整を試みる手法として，第 4 章で提案している MDPs を対象とした行動選択確率のエントロピーを用いた状態フィルタの一実現手法[52]を，POMDPs を対象とするように拡張を行う．その際，対象システムが連続状態空間および離散状態空間のどちらを有する場合においても，適応的に履歴情報の記録・参照を行い，エージェントの状態空間を繰り返し分割・統合することで，状態空間のコンパクト化を実現しつつ POMDPs への対応が可能な状態フィルタの一実現手法を提案する．さらに， 例題として離散状態空間を有する迷路問題および連続状態空間を有するロボットナビゲーション問題を取り上げ，計算機実験を通して従来手法と比べることで，提案手法の有効性と実問題への適用可能性を確認する．

5.2 状態フィルタの適応的獲得手法

5.2.1 基本的考え方

以下の考え方に基づいて，式 (4.1) で示される内部状態におけるボルツマン選択法を用いた行動選択確率についてのエントロピーの和および内部状態空間のサイズ $|\boldsymbol{\mathcal{S}}_f|$ をできるだけ小さくすることを目標として状態フィルタを適応的に獲得する．ここで $\boldsymbol{\mathcal{S}}_f$ は状態フィル

タ f が構成する内部状態空間を表す．また，式 (4.2) で示される $H(s)$ はある内部状態 s でのエントロピーを示す．このような状態フィルタの獲得方法として状態フィルタの学習と行動学習を異なる時定数で実行する手法も考えられるが，ここでは状態フィルタの学習と行動学習を同じ時定数で行う．

ここではこのエントロピー $H(s)$ を，1 つの内部状態 s についての分割の十分性を評価する指標として捉えられることで，状態フィルタを調整する．具体的には，不完全知覚問題が発生している場合，エージェントが出力すべき行動が一意に決まらず，$H(s)$ は小さくならない．逆に言えば，$H(s)$ が小さい場合，g による不完全知覚問題が発生せず，そして，f による不完全知覚問題が発生しない程度に，内部状態 s が十分に分割されていると考えることができる．しかしながら，ある内部状態 s におけるエージェントの最適な行動が複数ある場合には，不完全知覚問題が発生していない場合においても，$H(s)$ が小さくならない．すべての内部状態において $H(s)$ を小さくできるシステムの条件として，必要十分条件を正確に導くことはできないが，要件：

- すべてのシステムの状態において，エージェントの最適な行動が 1 つに定まること，

が満たされた場合に，すべての内部状態において $H(s)$ を小さくできる，すなわち提案手法が有効に働くことが期待される．ここで,「最適な行動が 1 つに定まる」ことについて，本章では次のように解釈しておくものとする．エージェントの最適な行動が複数あるシステムの状態 x に対し，条件：

(1) x' において最適な行動が 1 つに定まる，

(2) x' における最適な行動が x における最適な行動群の 1 つである，

が満たされる $x' \in N(x)$ が存在するとき，x から状態フィルタを通して写像された内部状態において，エージェントの最適な行動が 1 つに定まると期待できる．そのため，このような x において最適な行動が 1 つに定まるものとする．なお，$N(x)$ は状態空間内において x に連続した (システムの状態空間が離散である場合は，状態空間内において隣り合う) システムの状態の集合である．

以上を利用して，ある s において十分な回数行動学習をしているにも関わらず，$H(s)$ が小さくならない場合，f による不完全知覚問題が発生しているとして，s を通常分割する．また，必要以上に内部状態を通常分割している場合には，内部状態を統合する．具体的には，あるステップ t のある s_i について $H(s_i)$ が小さく，かつ次のステップ $t+1$ の異なる $s_j (i \neq j)$ についても $H(s_j)$ が小さく，さらに互いの内部状態における代表行動が同じであれば，必要以上に内部状態を通常分割していると判断し，s_i, s_j を統合する．ここで，内部状態における代表行動とは，その内部状態において最も選択確率の高い行動を指す．さらに，ある s において，十分に分割されており，十分な回数行動学習をしているにも関わらず，$H(s)$ が小さくならないのであれば，g による不完全知覚問題が発生しているとして，s を履歴参照

分割する．また，履歴情報を用いて写像されるある s について，十分な期間写像されない場合，必要以上に履歴参照分割していると判断し，s を削除する．

ここでは，離散状態空間を有する POMDPs と連続状態空間を有する POMDPs を同様に扱うことを可能とするために，まず状態フィルタに，集約状態空間 $\boldsymbol{\mathcal{Q}}$ を保持させ，観測状態 $o \in \boldsymbol{\mathcal{O}}$ のみから集約状態 $q \in \boldsymbol{\mathcal{Q}}$ への写像 f^{Q} を以下のように定義する．

$$q_i = f^{\mathrm{Q}}(o) : (\forall n, o^{\mathrm{S}}_{in} \leq o_n < o^{\mathrm{E}}_{in}) \tag{5.1}$$

ただし，観測状態を $o = \{o^1, o^2, \cdots, o^M\}$ で表される複数の観測状態要素 $o^j (1 \leq j \leq M)$ から構成されるものとし，$o^{\mathrm{S}}_{in}, o^{\mathrm{E}}_{in}$ はそれぞれ，q_i に写像される観測状態についての次元 n の観測状態要素の開始値と終端値を表し，q_i に写像される観測状態の次元 n の区間幅 $R(q_i, n)$ を $R(q_i, n) = o^{\mathrm{E}}_{in} - o^{\mathrm{S}}_{in}$ とする．すなわち，単一の集約状態に写像される観測状態の集合は，観測状態空間内で超直方体領域で表されるものとする．観測状態内において超直方体領域内の複数の観測状態を単一の状態に集約するものとして，写像 f^{Q} を「状態集約モジュール」と呼び，理想的状態フィルタの要件 (1),(2) を満たすように，f^{Q} を調整する．

また，状態フィルタに，ある集約状態 q_i において用いる履歴情報の深さ d_i を保持する「履歴参照モジュール」と呼ばれるモジュールを用意する．ここで，用いる履歴情報の深さ d_i は q_i において d_i ステップ過去までの履歴情報を用いることを表し，$q_i = f^{\mathrm{Q}}(o(t))$ のとき，

$$s(t) = f(o(t), z(1), \cdots, z(d_i)) \tag{5.2}$$

となる．そして，状態フィルタは履歴情報としてステップ $d = \max_i d_i + 1$ までの履歴情報を記録することで，必要以上に履歴情報を記録することなく，メモリの問題を解決できると考えられる．

ここでは，履歴情報として，過去の観測状態と行動を記録することにする．履歴情報を用いる際に，過去の観測状態から写像される過去の集約状態と過去の行動の対を用いる．つまり，

$$z(t') = \{f^{\mathrm{Q}}(o(t - t')), a(t - t')\} \tag{5.3}$$

とする．履歴情報として観測状態ではなく，集約状態を用いることで，より状態空間のコンパクト化を実現できると考えられる．

5.2.2 状態フィルタの構成

基本的な考え方に添って状態フィルタを調整するため，状態フィルタを，以下のように構成する．

(1) 状態識別モジュール：理想的状態フィルタの 3 つの要件が満たすように識別状態空間 $\boldsymbol{\mathcal{P}}$ を調整する．特に，理想的状態フィルタの要件 (3) を満たすために，履歴情報を用いて $\boldsymbol{\mathcal{P}}$ を調整する．詳細は後述する．

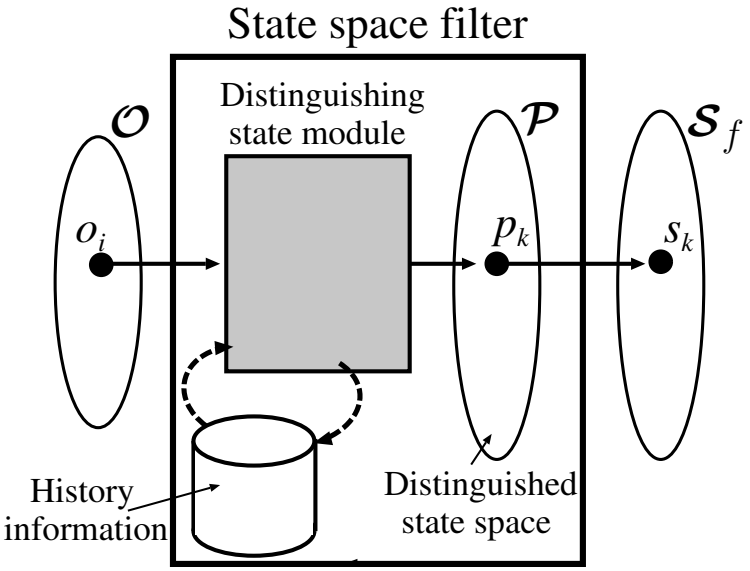

図 5.1: 状態フィルタの構成の概要

(2) 履歴情報：$d_{\mathrm{T}} = \max_i d_i + 1$ ステップ過去までの観測状態と行動が記録され，各ステップが進むごとに情報が更新される．

(3) 識別状態空間 $\mathcal{P}^*$：状態要素を，現在の集約状態と 0 個以上の過去の集約状態と 0 個以上の過去の行動とする状態空間とする．状態識別モジュールから識別状態空間上の一点が指定され，内部状態空間へ 1 : 1 で写像される．

図 5.1 に，状態フィルタの構成の概要を示す．つぎに，状態識別モジュールの構成について述べる．

状態識別モジュールの構成

状態識別モジュールを，以下のように構成する．

(1) 状態集約モジュール f^{Q}：観測状態空間 $\mathcal{O}$ から集約状態空間 $\mathcal{Q}$ への写像を行う．その際，理想的状態フィルタの要件 (1), (2) を満たすように，調整する．

(2) 集約状態空間 Q：状態集約モジュールによってある観測状態から集約状態空間上の一点に写像される．

(3) 履歴参照モジュール：各集約状態 q_i に対して用いる履歴情報の深さ d_i を保持する．また，過去 d_i ステップ分の履歴情報を参照し，過去の観測状態を集約モジュールに，過去の行動を状態指定モジュールへ送る．

(4) 状態指定モジュール f^{P}：状態集約モジュールから現在と過去の観測状態に対応する集

*次元の異なる要素で構成されるため，厳密には「リスト」であるが，ここでは便宜上「空間」と呼ぶ．

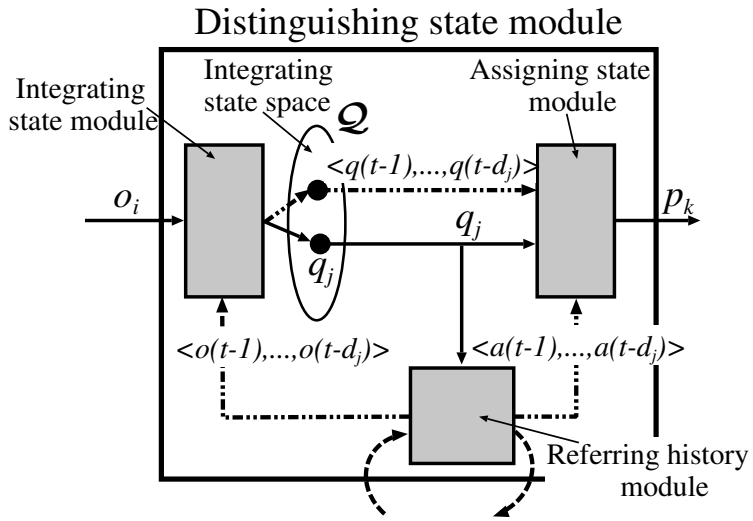

図 5.2: 状態認識モジュールの構成の概要および認識状態 p_k への写像

約状態が，履歴参照モジュールから過去の行動がそれぞれ送られ，それらによって構成される識別状態を指定する．つまり，$q_i = f^{\mathrm{Q}}(o(t))$ のとき識別状態 $p(t)$ を，

$$p(t) = \begin{cases} f^{\mathrm{P}}(q_i, z(1), \cdots, z(d_i))) & (d_i > 0) \\ f^{\mathrm{P}}(q_i) & (d_i = 0) \end{cases} \tag{5.4}$$

とする．

図 5.2 に，状態認識モジュールの構成の概要と $q_i = f^{\mathrm{Q}}(o_i)$ かつ $d_i > 0$ の場合において，認識状態 p_k が指定される様子を示す．

5.2.3 状態フィルタの調整法

ここでは，はじめに f による不完全知覚問題を解消するため，状態集約モジュールを適宜調整し．内部状態を「通常分割」する．通常分割が十分に行われた後に，g による不完全知覚問題を解消するため，履歴参照モジュールと状態指定モジュールを適宜調整し，内部状態を「履歴参照分割」する．

必要以上に通常分割が行われた際は，状態集約モジュールを調整し，内部状態を「統合」する．また，必要以上に履歴参照分割が行われた際は，状態指定モジュールを調整し，内部状態を「削除」する．

以下に，上記の 4 つの調整法を詳細に記述する．

通常分割

q_j を通して写像される s_i において，$L(s_i) > \theta_{\mathrm{L}}$ かつ $H(s_i) > \theta_{\mathrm{H}}$ かつ $d_j = 0$ かつ $\exists n(R(q_j, n) \geq 2\theta_r)$ のとき，$R(q_j, n) \geq 2\theta_{\mathrm{r}}$ となる 1 つ以上の次元 n において，q_i に写像される観測状態の区間を中心で 2 分割する．つまり，$R(q_j, n) \geq 2\theta_{\mathrm{r}}$ となる n が 1 つの場合，$o^{\mathrm{S}}_{j'n} = o^{\mathrm{S}}_{jn}, o^{\mathrm{E}}_{j'n} = o^{\mathrm{S}}_{jn} + R(q_j, n)/2$ と $o^{\mathrm{S}}_{j''n} = o^{\mathrm{S}}_{jn} + R(q_j, n)/2, o^{\mathrm{E}}_{j''n} = o^{\mathrm{E}}_{jn}$ とする $q_{j'}, q_{j''}$ に分割する．そして，それぞれ異なる集約状態を指定し，それぞれの集約状態に対して用いる履歴情報の深さをすべて 0 とし，それぞれ異なる識別状態を指定するように，状態集約モジュールと履歴参照モジュール，そして状態指定モジュールをそれぞれ調整する．ただし，$L(s_i)$ は s_i における行動学習回数，θ_{L} は $L(s)$ についての閾値，θ_{H} は $H(s)$ についての閾値，θ_{r} は分割される区間幅についての閾値を表す．つまり，θ_r を調整することで，集約状態がどのくらいまで細かく分割するのかを調整する．

この操作により，分割後の $|\boldsymbol{\mathcal{S}}_f|$ は，$R(q_j, n) \geq 2\theta_r$ となる n の次元数を N とすれば，$(2^N - 1)$ だけ増加する．なお，新たに生成された 2^N 個の内部状態の評価値は，分割前の内部状態の評価値とする．

統合

$q_j = f^{\mathrm{Q}}(o(t-1))$ を通して写像される s_i と，$q_l = f^{\mathrm{Q}}(o(t)), j \neq l$ を通して写像される s_k において，$H(s_i) \leq \theta_{\mathrm{H}}$ かつ $H(s_k) \leq \theta_{\mathrm{H}}$ かつ $d_j = 0$ かつ $d_l = 0$ かつ，$a^+(s_i) = a^+(s_k)$ のとき，q_j と q_l を 1 つの集約状態に統合されるように，そして s_i と s_k を 1 つの内部状態に統合されるように状態集約モジュールと履歴参照モジュール，そして状態指定モジュールを調整する．なお，統合に伴い新たに生成された内部状態の評価値は，統合前の 2 つの内部状態における評価値の平均とする．

また，ある期間 θ_{t} に一度も写像されない集約状態 $q_i(d_i = 0)$ が存在するとき，q_i とその隣合う (具体的には，写像される観測状態の範囲が隣接する) 集約状態 q_j を 1 つの集約状態 $q_k(d_k = d_j)$ に，そしてそれぞれ q_i, q_j から識別状態空間を通して写像される 2 つの内部状態を 1 つの内部状態に統合するように，状態集約モジュールと履歴参照モジュール，そして状態指定モジュールを調整する．ここで，$a^+(s)$ は内部状態 s における代表行動を表し，θ_{H} は分割法に用いられたものと同じ値が用いられる．なお，統合に伴い新たに生成された内部状態の評価値は，s_j における評価値の平均とする．

この操作により，統合後の $|\boldsymbol{\mathcal{S}}_f|$ は，1 だけ減少する．ただし，統合が行われるのは，統合後の写像される観測状態の範囲が超直方体領域となる場合に限る．

履歴参照分割

$q_j = f^{\mathrm{Q}}(o(t))$ を通して写像される s_i において，$L(s_i) > \theta_{\mathrm{L}}$ かつ $H(s_i) > \theta_{\mathrm{H}}$ であり，$d_j > 0$ または $d_j = 0$ かつ $\max_n R(q_j, n) < 2\theta_{\mathrm{r}}$ のとき，$d_j \leftarrow d_j + 1$ と更新した後，履歴情報から $o(t-d_j), a(t-d_j)$ が参照され，そして，$q(t-d_j), a(t-d_j)$ を用いて，s_i とは異なる内部状態に写像されるように，履歴参照モジュールと状態指定モジュールをそれぞれ調整する．

ここでは，1 ステップ過去の履歴情報に，限られた行動が記録されており，また限られた集約状態が参照されると想定し，1 ステップ過去の履歴情報を用いて一括して分割するのではなく，1 ステップ過去の履歴情報に新たな行動や新たな集約状態が参照される度に 1 つずつ識別状態を生成し，新たな内部状態に写像する．

この操作により，分割後の $|\boldsymbol{\mathcal{S}}_f|$ は，1 だけ増加する．なお，新たに生成された内部状態の評価値は，分割前の内部状態の評価値とする．

以上より，履歴参照分割するために，履歴情報の深さ $d_{\mathrm{T}} = 1$ から始められ，適応的に $d_{\mathrm{T}} = \max_i d_i + 1$ とされる．

削除

ある期間 θ_{t} 一度も写像されない内部状態 s_i について，識別状態空間を通して s_i に写像される q_j について $d_j > 0$ のとき，s_i を削除するように履歴参照モジュールと状態指定モジュールを調整する．

この操作により，削除後の $|\boldsymbol{\mathcal{S}}_f|$ は，1 だけ減少する．

あるステップ t のときの状態フィルタを調整する手順を，内部状態空間の初期設定と履歴情報の適応的な記録方法を加えて，以下にまとめる．

(1) サイズ 1 の内部状態空間から，1 回通常分割を行った後の内部状態空間のサイズ 2^M から，また，履歴情報の深さ $d_{\mathrm{T}} = 1$ から始める．ここで M は観測状態の次元数を表す．

(2) $o(t)$ を観測する．

(3) $o(t)$ から状態フィルタによって $s(t)$ が得られる．ただし，複数の過去の観測状態から同じ集約状態に繰り返し写像することを避けるため，$q(t) = q(t-1)$ の時，$s(t) \leftarrow s(t-1)$ とする．ただし，$q(t) = f^{\mathrm{Q}}(o(t))$ とする．そして $s(t-1)$ における行動学習を実行し，$s(t-1)$ における行動学習回数について $L(s(t-1)) \leftarrow L(s(t-1)) + 1$ と更新する．

(4) $s(t-1)$ における行動学習によって変化したエントロピー $H(s(t-1))$ を更新し，$s(t)$ におけるエントロピー $H(s(t))$ を計算する．

(5) 以下の条件に従って，状態フィルタを調整する．ただし $q_i = q(t-1), q_j = q(t)$ とする．

   a. もし $L(s(t-1)) > \theta_{\mathrm{L}}$ かつ $H(s(t-1)) > \theta_{\mathrm{H}}$ かつ $d_i = 0$ かつ $\exists n(R(q_i, n) \geq 2\theta_r)$

の場合，$s(t-1)$ を通常分割する．またこの際，$d_i > 0$ 及び $\max_n R(q_i, n) < 2\theta_r$ である場合，履歴参照分割する．特に，$d_i \leftarrow d_i + 1$ とした後，履歴情報の深さ $d_{\mathrm{T}} = d_i$ の場合，$d_{\mathrm{T}} \leftarrow d_{\mathrm{T}} + 1$ とする．

b. もし $s(t-1) \neq s(t)$ の場合，$d_i = 0$ かつ $d_j = 0$ であり，$H(s(t-1)) \leq \theta_{\mathrm{H}}$ かつ $H(s(t)) \leq \theta_{\mathrm{H}}$ かつ $a(s(t-1))^+ = a(s(t))^+$ であれば，$s(t-1), s(t)$ を統合する．なお，統合が行われるのは，統合後の内部状態に写像される観測状態の範囲が超直方体となる場合に限る．

(6) もし前回一度も写像されていない集約状態・内部状態がそれぞれ統合・削除されてから，ある期間 θ_{t} 経過していれば，再度一度も写像されていない集約状態・内部状態がそれぞれ統合・削除される．ただし，統合する場合は，統合後の内部状態に写像される観測状態の範囲が超直方体となる場合に限る．

(7) もし (5) によって状態フィルタを変更している場合，再度 $o(t)$ から状態フィルタによって $s(t)$ を得る．

(8) ステップ $t-1$ の情報を用いて履歴情報を更新する．そして $s(t)$ から行動空間への写像を行い，行動を実行する．

(9) $t \leftarrow t+1$ と更新され，(2) に戻る．

5.3　計算例および考察

5.3.1　離散状態空間

例題の設定

釜谷ら [46] が用いた図 5.3 に示す格子空間において，エージェントにスタート地点 S からゴール地点 G に到達するまでの行動系列を獲得させる迷路問題と呼ばれる例題に適用する．この問題において，エージェントは黒色のセルで示された壁への移動はできない．エージェントがゴール地点に到着した時点のみ報酬 10 が与えられる．そして，1 ステップに，エージェントが 1 回の観測・行動を取り終えるものとし，エージェントがスタート地点を出発してゴール地点に到着し報酬が与えられるまで，もしくは 100 ステップ経過するまでを 1 エピソードとする．

学習エージェント

入力として，現在のセルに隣接する 4 方向<北，東，南，西>のセルについて，壁の有無を観測できるものとする．つまり，エージェントが観測する値は 4 次元，最大で 16 種類とする．図 5.3 中の数字が同じセルにおいて，同じ状態が観測されることになり，ここに不完

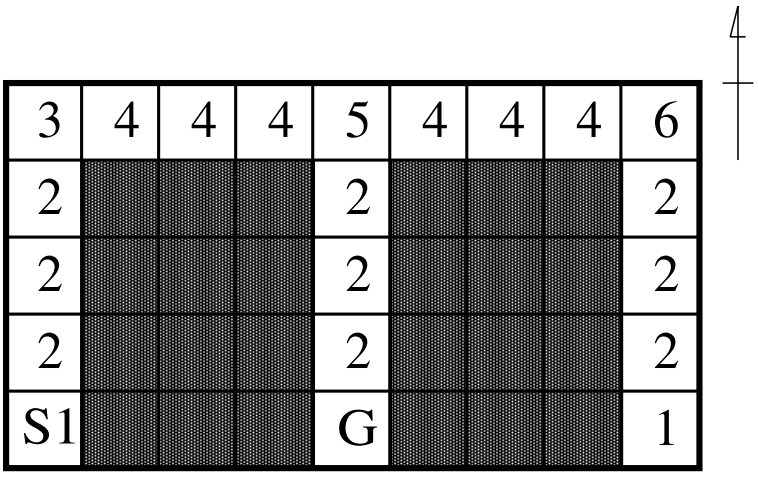

図 5.3: 迷路設定

全知覚問題が発生する．

出力として，4 方向に 1 セルの移動，すなわち 4 種類の行動を可能とする．

性能の比較および評価

提案手法 (以下，手法 A)，釜谷ら [46] に基づいた手法 (以下，手法 B)，そして単純に観測状態と内部状態を 1 : 1 に写像を行う状態フィルタを用いた手法 (以下，手法 C) との比較実験を行なった．ただし，手法 A の集約状態空間は，1 次元 [0 : 2) の区間で構成される 4 次元空間とした．また，手法 B は，連続状態空間を有するシステムには有効でないものの，手法 A と同様に，状態フィルタの学習と行動学習が同時並行的に行われ，かつ学習器に適用する RL 手法が特定されないという特徴をもった手法であるため，比較対象として用いた．

手法 B は，内部状態空間を複数用意し (状態フィルタリングの枠組みでは，複数の内部状態群によって構成される 1 つの内部状態空間として捉える)，それを階層構造学習オートマトン (Hierarchical structure Learning Automaton: HLA) によって生成されるサブゴールと呼ばれる観測状態のときに，内部状態空間を順次切り替えることで，POMDPs を MDPs へ局所的に近似する．また，同時に HLA によって生成されたサブゴール系列に対して，タスクの成功または失敗という 2 値の報酬を与えることで，良好なサブゴール系列を学習させる手法である．ここで，釜谷ら [46] の手法をそのまま用いた場合，HLA に与えられる 2 値の報酬を，ゴールに到達した場合を成功，到達できなかった場合を失敗として設定することになる．しかし，この設定の場合，一旦ゴールに到達することができるようになると (例えば，迂回経路であってもゴールに到達可能であるので)，その後は成功の報酬が与えられることが多くなり，それ以降 HLA の学習は進みにくくなる．これを防ぐために，θ_{B} ステップ以下でゴール到達した場合を成功，それ以外を失敗として報酬を与えることにした．

表 5.1: 例題 (1) における計算時間

Method	Computational time[s]
A	0.108
B	0.202
C	0.244

行動学習器に，すべての手法に同じパラメータの POMDPs にて有効である [1)]Sarsa(λ)(学習率 $\alpha = 0.1$，割引率 $\gamma = 0.9$，eligibility trace メカニズムに replacing trace[53)]，eligibility 係数 $\lambda = 0.9$，行動選択確率に温度係数 $\tau = 0.1$ のボルツマン選択法) を用いた．ただし，手法 A の行動学習器に用いられる eligibility 値は，状態フィルタが変化する際に，リセットされるものを用いた．

また，手法 A において，行動学習回数の閾値 $\theta_{\mathrm{L}} = 500$，エントロピーの閾値 $\theta_{\mathrm{H}} = 0.3$，分割される区間幅についての閾値 $\theta_{\mathrm{r}} = 1$，内部状態の統合・削除を行う期間 $\theta_{\mathrm{t}} = 100$[episode] を，手法 B において，サブゴール系列長 $N = 1$，HLA のステップサイズ $\alpha_B = 0.05$，HLA への報酬パラメータ $\theta_B = 15$ を用いた．ここで θ_{H} は，行動数 4 における，1 つの行動選択確率が 0.9 の時のエントロピーの最大値が約 0.312 であること，θ_{r} は対象システムが離散状態空間であること，N, α_B は釜谷ら [46)] の設定を，θ_B はこの例題の最適解が 12 ステップであることを参考に設定した．

状態フィルタリングの観点から学習速度と解の良さ，内部状態空間のサイズ，ならびに計算時間の項目に関して比較検討を行う．また，提案手法の状態フィルタの調整に用いられる履歴情報の深さについても検討を行い，離散状態空間における提案手法の有効性を検討する．

実験結果を図 5.4，図 5.5，表 5.1，図 5.6 に示す．なお，

- 図 5.4 に各エピソード完了に要したステップ数の変化，
- 図 5.5 に内部状態空間のサイズの変化，
- 表 5.1 に 1 回の実験に要した計算時間 (Intel Pentium4 3.06GHz, メモリ 1GB, Vine Linux 3.2, java1.5.0 使用時)，
- 図 5.6 に手法 A の履歴情報の深さの変化，

を示す．ここで，実験結果はすべて 20 回の実験の平均である．

各検討項目に対して，次のようなことが確認できる．

- 検討項目 (1)：学習速度と制御ルールの良さについて図 5.4 より，
  - 学習速度について，手法 A が他の手法より良い性能を示していること，
  - 獲得される制御ルールにおいて，手法 A，B ともに最適な制御ルールを獲得していること，

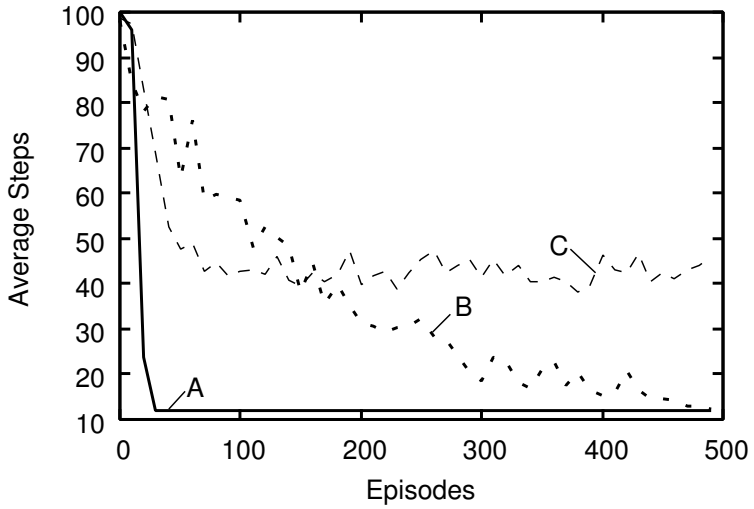

図 5.4: 例題 (1) における所要ステップ数の変化

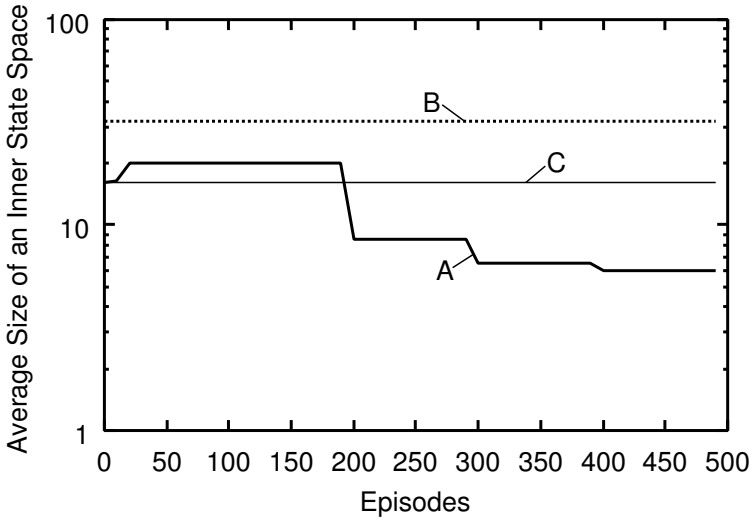

図 5.5: 例題 (1) における内部状態空間のサイズの変化

- 検討項目 (2)：内部状態空間のサイズについて図 5.5 より，
  - 手法 A が他の手法より小さく抑えられていること，
  - 手法 A の内部状態空間のサイズが 6 で収束しており，システムの状態空間のサイズである迷路の格子数 21 よりも小さいこと，
- 検討項目 (3)：計算時間について表 5.1 より，
  - 手法 A が他の手法より計算時間を短く抑えられること，
- 検討項目 (4)：履歴情報の深さについて図 5.6 より，手法 A は，
  - 2 ステップ過去までの履歴情報を記録していること，

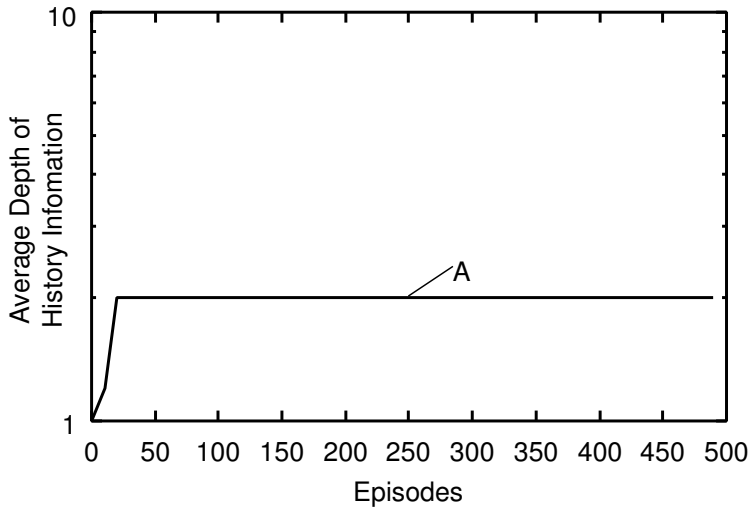

図 5.6: 例題 (1) における手法 A の履歴情報の深さの変化

– つまり 1 ステップ過去までの履歴情報を参照していること，

以上より，POMDPs における離散状態空間を有する迷路問題において，手法 A が他の手法より良い性能を示していることから，手法 A の有効性が確認できた．そして，検討項目 (1) から POMDPs への対応を，検討項目 (2) から状態空間のコンパクト化を，それぞれ実現できていること，検討項目 (2) より，理想的な状態フィルタが獲得できていること，検討項目 (2), (3), (4) より，メモリ量・計算時間がそれぞれ絶対的に膨大でないことから実問題への適用可能性，がそれぞれ確認できた．

5.3.2　連続状態空間

例題の設定

壁に囲まれた 100[mm] ×250[mm] の連続空間において，半径 25[mm] の円筒型のエージェントにスタート状態からゴール状態に到達するまでの行動系列を獲得させるロボットナビゲーション問題と呼ばれる例題に適用する．ここで，エージェントの中心座標や進行方向を表すため，xy 座標を，壁の一角を原点，対角を $(x, y) = (100, 250)$ とするように設定する．

システムの状態は＜エージェントの中心座標 (x_A, y_A)，エージェントの進行方向 θ_A ＞と表すことができる．そして，スタート状態を＜ $(x_A, y_A) = (25, 25), \theta_A$: $(x, y) = (0, 0)$ の向き＞，ゴール状態を＜ $(x_A, y_A) = (75, 225), \theta_A$: $(100, 250)$ の向き＞とする．ここで，エージェントがゴール状態に到着した時点のみ報酬 10 が与えられる．

1 ステップに，エージェントが 1 回の観測・行動を取り終えるものとし， エージェントが終端状態に到着し報酬が与えられるまで，もしくは 5000 ステップ経過するまでを 1 エピ

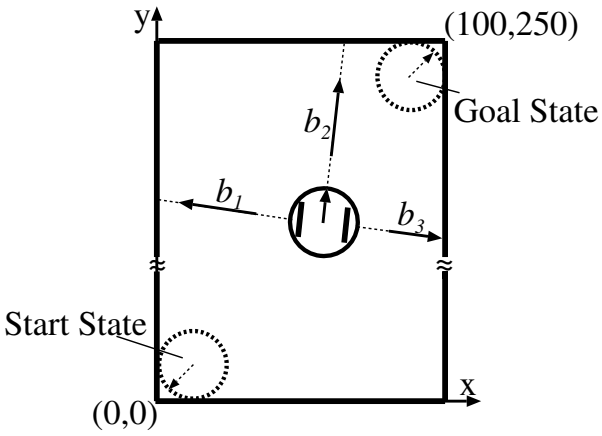

図 5.7: エージェントへの入力情報

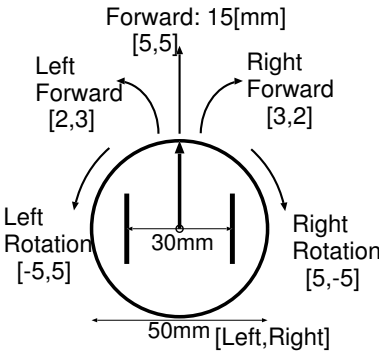

図 5.8: エージェントの行動

ソードとする．

学習エージェント

入力として，3 つの同性能の距離センサを持ち，進行方向を中心にそれぞれ $-1/2\pi$[rad], 0[rad]，$1/2\pi$[rad] だけ異なる方向の壁までの距離を測定できるものとする．ただし，それぞれの測定可能距離は 20[mm] から 75[mm] とする．つまり，入力状態は，＜エージェントの左側の距離センサの値 b_1[mm]$(20 \leq b_1 \leq 75)$，エージェントの正面の距離センサの値 b_2[mm]$(20 \leq b_2 \leq 75)$，エージェントの右側の距離センサの値 b_3[mm]$(20 \leq b_3 \leq 75)$ ＞の 3 次元とする．図 5.7 にエージェントへの入力情報の概要を示す．ここで，各センサの測定可能距離の制限によって，特にスタート状態付近においてゴール状態付近と同じ状態が観測され，ここに不完全知覚問題が発生する．

出力として，2 輪の車輪を持ち，両輪の回転速度を制御することによって，前進，左 (右) 前進，左 (右) 回転の 5 種類の行動が可能であるとする．図 5.8 に，各行動における両輪の回転速度比を示す．具体的には，前進行動で 15[mm] 進み，左 (右) 前進行動時には，回転しながら進み，結果的に ± 0.1[rad] 回転する．左 (右) 回転行動時には移動せず，± 1.0[rad] 回転する．

性能の比較および評価

提案手法 (以下，手法 A) と観測状態空間をそれぞれの次元均等に 10 分割する状態フィルタを用いたグリッド分割法 (以下，手法 C) との比較実験を行なった．ただし，手法 A の集約状態空間と手法 C の観測状態空間は，1 次元を $[0 : 100]$ の区間で構成される 3 次元空間とした．行動学習器，パラメータについては，例題 (1) と同じものを用いた．ただし，分割される

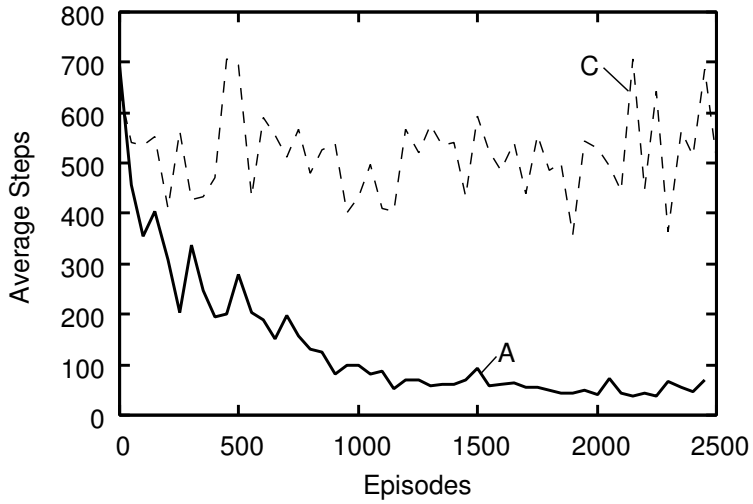

図 5.9: 例題 (2) における所要ステップ数の変化

表 5.2: 例題 (2) における計算時間

Method	Computational time[sec]
A	20.41
C	176.22

区間幅についての閾値 $\theta_r = 10.0$[mm]，内部状態の統合・削除を行う期間 $\theta_{\rm t} = 500$[episode] を用いた．ここで，$\theta_{\rm H}$ は，行動数 5 において，1 つの行動選択確率が 0.9 の時のエントロピーの最大値が約 0.288 であることを参考に設定した．

例題 (1) と同様に状態フィルタリングの観点から学習速度と解の良さ，内部状態空間のサイズ，ならびに計算時間の項目に関して比較検討を行う．また，提案手法の状態フィルタの調整に用いられる履歴情報の深さについても検討を行い，連続状態空間における提案手法の有効性を検討する．

実験結果を図 5.9，図 5.10，表 5.2，図 5.11 に示す．なお，

- 図 5.9 に各エピソード完了に要したステップ数,
- 図 5.10 に内部状態空間のサイズの変化,
- 表 5.2 に 1 回の実験に要した計算時間 (例題 (1) と同じ PC 使用時),
- 図 5.11 に手法 A の履歴情報の深さの変化,

を示す．ここで，実験結果はすべて 20 回の実験の平均である．

各検討項目に対して，次のようなことが確認できる．

- 検討項目 (1)：学習速度と制御ルールの良さについて図 5.9 より,

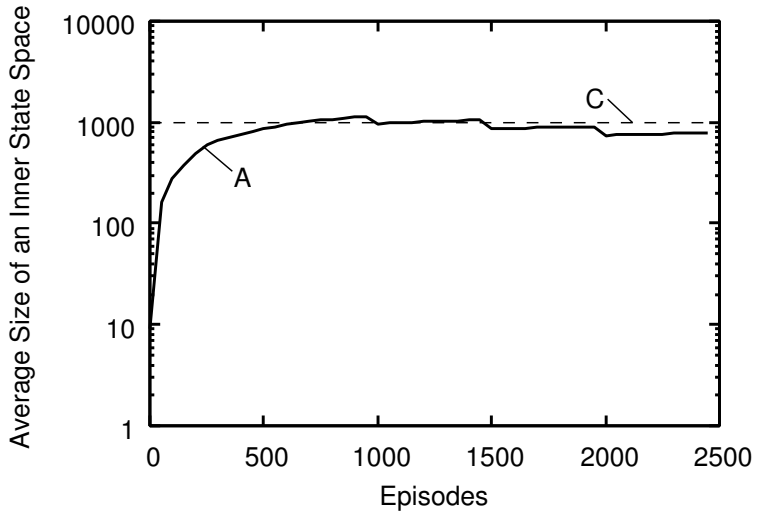

図 5.10: 例題 (2) における内部状態空間のサイズの変化

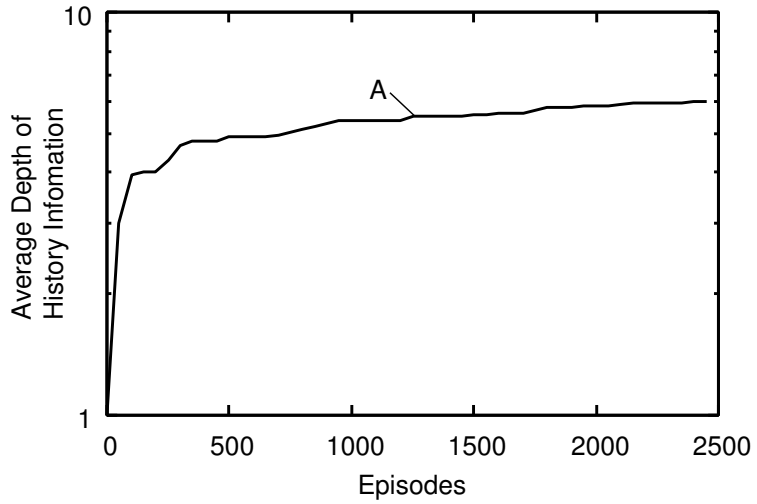

図 5.11: 例題 (2) における手法 A の履歴情報の深さの変化

  - 手法 A が良い性能を示していること,
  - 手法 C が適切な制御ルールを獲得することができなかったことにより，この例題において不完全知覚問題が発生していること,

- 検討項目 (2)：内部状態空間のサイズについて図 5.10 より,
  - 手法 A，C が共に約 1000 程度で同程度の内部状態空間のサイズとなっていること,

- 検討項目 (3)：計算時間について表 5.2 より,
  - 手法 A が手法 C より計算時間を短く抑えられること,

- 検討項目 (4)：履歴情報の深さについて図 5.11 より，手法 A は,

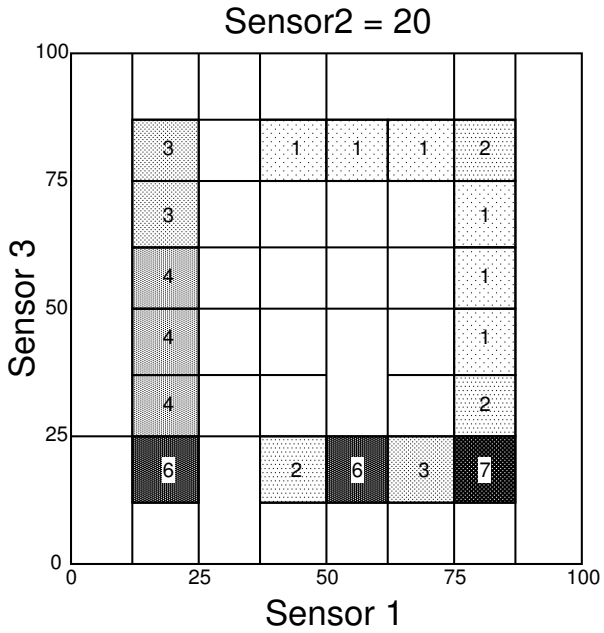

図 5.12: 例題 (2) において手法 A により獲得された状態フィルタの一例 (b_2=20)

- 最終的に約 6 ステップ過去までの履歴情報を記録していること,
- つまり約 5 ステップ過去までの履歴情報を参照していること.

ここで，例題 (2) において手法 A により獲得された状態フィルタの一例を

- $b_2 = 20$ における状態フィルタを図 5.12,
- $b_2 = 47.5$ における状態フィルタを図 5.13,
- $b_2 = 75$ における状態フィルタを図 5.14,

にそれぞれ示す．ただし，示した状態フィルタについて，1 つの格子が 1 つの集約状態を表し，格子内の数字はその格子で表される集約状態に対して用いる履歴情報の深さを表す．なお，集約状態における履歴情報の深さが 0 の場合は数字を省略した．図 5.12〜図 5.14 より，手法 A によって獲得された状態フィルタの一例は，壁に近付くほど細かく分割されていることが確認できる．

以上より，対象システムが連続状態空間を有するロボットナビゲーション問題において，手法 A が良い性能を示していることから，手法 A の有効性が確認できた．そして，検討項目 (1) より POMDPs への対応を，検討項目 (2) より状態空間のコンパクト化を，それぞれ実現できていること，検討項目 (4) より，適応的に履歴情報を記録・参照できていること，検討項目 (2), (3), (4) よりメモリ量・計算時間がそれぞれ絶対的に膨大ではないことから実問題への適用可能性，がそれぞれ確認できた．

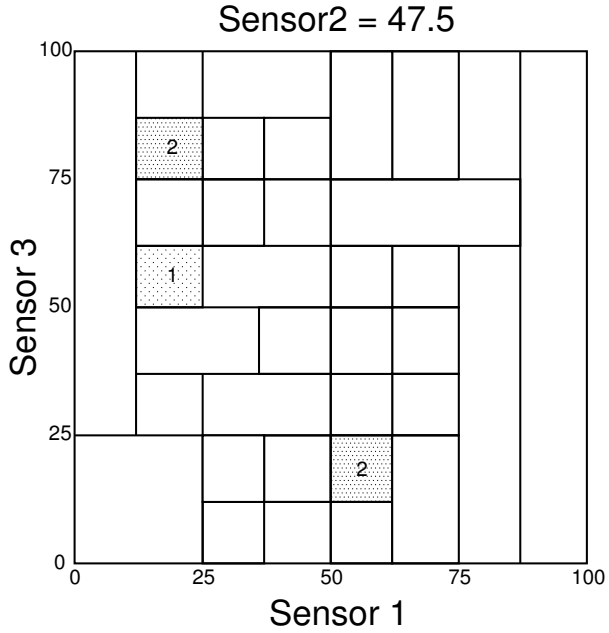

図 5.13: 例題 (2) において手法 A により獲得された状態フィルタの一例 (b_2=47.5)

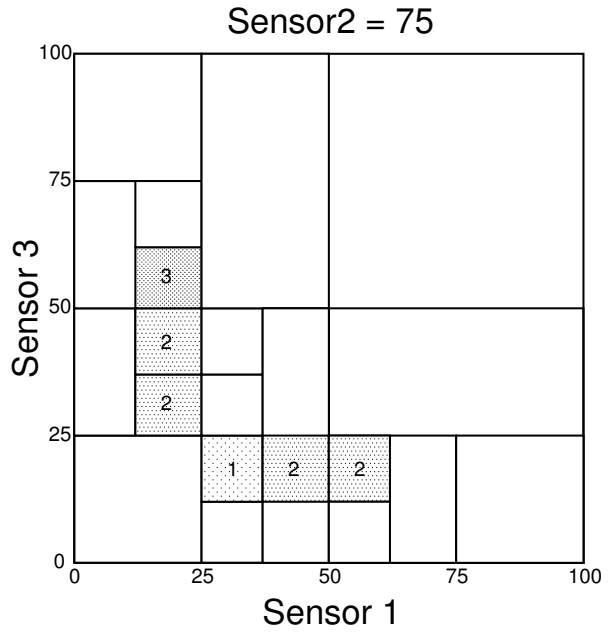

図 5.14: 例題 (2) において手法 A により獲得された状態フィルタの一例 (b_2=75)

5.4 結言

本章では，POMDPs への対応・状態空間のコンパクト化に焦点を当て，第 4 章で提案した手法を POMDPs に拡張する形で，適応的に履歴情報の記録・参照を行い，繰り返し内部

状態を分割・統合することで POMDPs を対象とした状態フィルタの一実現手法を提案した．この際，ある内部状態における行動選択確率のエントロピーを，その内部状態の細かさの十分性を評価する指標として用いた．また，この手法は，状態フィルタの学習と行動学習を同時並行して行わせることができ，学習器において適用できる強化学習手法が特定されないという特徴をもっている．

さらに，強化学習問題の例題として離散状態空間を有する迷路問題および連続状態空間を有するロボットナビゲーション問題を取り上げ，計算機実験を通して，提案手法を従来手法と比べることで，POMDPs への対応・状態空間のコンパクト化を実現できることを，そして提案手法の有効性・実問題への適用可能性を，それぞれ確認した．

今後の課題として，観測できる情報について優先度を評価することにより，より合理的に状態を分割することができる状態フィルタの獲得方法の検討，などが挙げられる．

第 6 章

電動車いすの適応的直進走行システム

6.1 諸言

本章では，第 5 章で提案する手法のより実際的な問題への適用例として，電動車いすの直進走行システムを取り上げる．

一般に，電動車いすの使用者は肢体麻痺などのかなり重度の障害を持つ．そして，電動車いすの標準的な入力装置であるジョイスティックの操作が筋力低下などのため困難な場合は，操作スイッチと呼ばれる入力装置を用いて電動車いすを操作する．操作スイッチは，僅かな指先の運動，まばたき，呼吸，音声など，随意に動かせる身体機能で操作できるものである [54, 55]．このような状況においてもなお，ユニバーサル社会実現のためには，障害が重度である電車いす使用者も安全・快適に自立移動できるまちの環境の整備が必要である．その一方，まちにあるすべてのバリアを除去することは現実的には不可能であることから，電動車いすに，バリアを走破しやすい機構を付与するなどの対応も同時に必要となっている．そこで近年，電動車いす側からユニバーサル社会実現に向けた研究がなされている [56, 57]．

ここでは，まちにあるバリアとして傾斜路面 [58, 59, 60] に注目する．傾斜路面において電動車いすを直進走行させることを考えた場合，使用者が幾度となく方向を修正することが必要となる．その際，入力装置として操作スイッチを用いる場合は，使用者にとって特に大きな負担となっている．さらに，使用者の重心が電動車いす本体に対して左右方向に対して偏りのある場合には，より一層使用者の操作負担が大きくなると考えられる [61]．しかしながら現在，傾斜角が一定でない通常の傾斜路面に対して有効な対策はほとんどなされていない．加えて，従来手動の車いすや電動車いすの基本ダイナミクスについてモデル化 [61, 59] が行われているものの，平坦路面上であることを前提とした二次元上でのモデル化 [61] であることが多く，車いすの左右方向の傾きのみが考慮されたモデル化 [59] はなされているものの，車いすの前後方向の傾きと左右方向の傾きを考慮にいれた三次元上のモデル化がほとんどなされていない．

そこで本章では，まちにあるバリアとして傾斜路面に注目し，まず従来のモデル [61] をもとに，左右 DC モータから駆動力を得る後輪駆動型 EWC を対象として，三次元上における電動車いすの基本ダイナミクスのモデル化を行う．つぎに，使用者の操作負担を軽減するた

めに，傾斜路面においても 1 つの入力で電動車いすに直進走行をさせるシステム (以下，直進走行システム) を考える．このようなシステムには次の 3 つの要件が課せられる．

(1) 使用者一人一人に合わせた適切な制御ルールが実現されること，

(2) 未経験の路面を走行する場合や使用者の重心位置，車輪の空気圧が変化した場合など，使用者を含む EWC の特性が変化した場合においても，適切な制御ルールが維持されること，

(3) 使用者を含む電動車いすに関して，センシングする情報が部分的であっても適切な制御ルールが構築されること．

上記の要件を満たすために，第 5 章で提案した手法 [62] を再構築し適用することを考える．そこでまず，電動車いすの基本的なダイナミクスを考慮して，強化学習手法の設定を行う．そして，

(1) 傾斜角が一定な路面を走行する場合，

(2) 傾斜角が一定でない路面を走行する場合，

さらに補足として，

(3) 重心位置が変化する場合，

(4) 2 つの路面を走行する場合，

についてシミュレーションを行い，提案システムの有効性，実際の電動車いすへの適用可能性を検討する．

6.2　電動車いすの基本ダイナミクスのモデル化

6.2.1　電動車いすに関する主な記号

ここでは，EWC として標準的な左右 DC モータによって駆動力を得る後輪駆動型 EWC を対象とする．まず，三次元空間における基本ダイナミクスを求める．EWC の基本ダイナミクスを求める際に利用する主な記号を以下に記し，図 6.1 に電動車いすのフレームモデルを示す．ただし，本体は使用者を含む電動車いす全体を示す．また，F_{L*} と F_{R*} が同様に記述できる際，$F_{L*/R*}$ と略記する．ここで，O'-xyz は x 軸を電動車いすの左右方向右向き，y 軸を電動車いすの前後方向前向き，z 軸を電動車いすの上下方向上向き，電動車いすの後輪軸の中心を原点とする運動座標系とし，O-XYZ は Z 軸を鉛直方向上向きにとった静止座標系とする．

O-XYZ　：静止座標系
O'-xyz　：左右後輪の中心を原点とする運動座標系
$V_{\mathrm{L/R}}$　：左右モータへの入力電圧 [V]

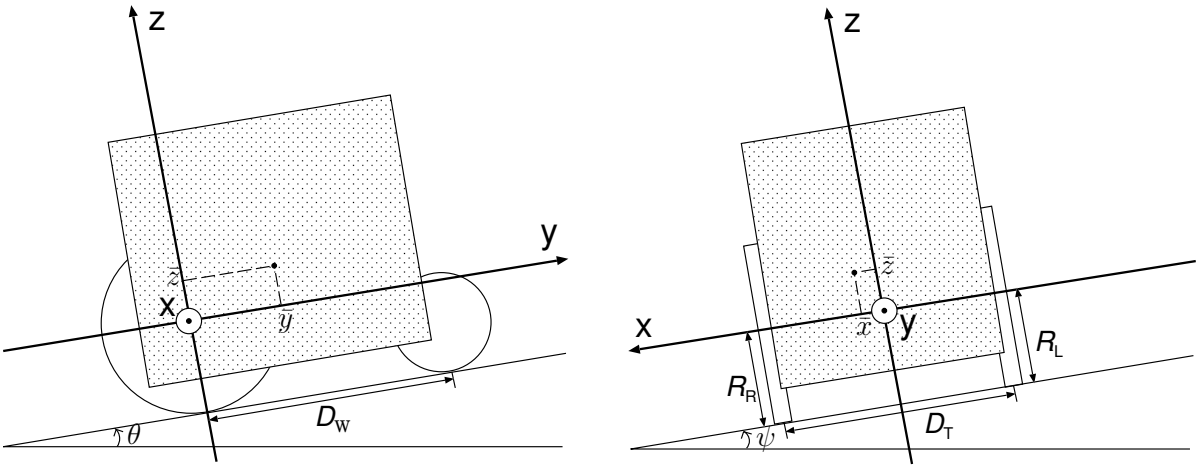

図 6.1: 電動車いすのフレームモデル

ω_B	:	本体 (重心) の角速度 [rad/s]
$\dot{\omega}_B$	:	本体 (重心) の角加速度 [rad/s^2]
ω_{BZ}	:	本体の Z 軸まわりの回転速度 [rad]
v_y	:	本体 (重心) の並進方向の速度 [m/s]
a_y	:	本体 (重心) の並進方向の加速度 [m/s^2]
v_X, v_Y	:	本体の X 軸，Y 軸方向の速度 [m/s]
$\omega_{L/R}$	:	左右後輪の角速度 [rad/s]
$\dot{\omega}_{L/R}$	:	左右後輪の角加速度 [rad/s^2]
δ	:	本体のヨー角 (方位角)[rad]
θ	:	本体のピッチ角 (前後方向に傾いている角度)[rad]
ψ	:	本体のロール角 (左右方向に傾いている角度)[rad]
$F_{L/R}$	:	左右後輪の接地点に働く力 [N]
F_{Gx}, F_{Gy}, F_{Gz}	:	本体 (重心) に働く重力の x,y,z 軸成分 [N]
F_C	:	本体 (重心) に働く遠心力 [N]
$F_{LG/RG}$	:	左右後輪の接地点に働く勾配抵抗 [N]
$F_{LFN/RFN}$	:	左右前輪の接地点に働く垂直抗力 [N]
$F_{LRN/RRN}$	:	左右後輪の接地点に働く垂直抗力 [N]
$F_{LI/RI}$	:	左右モータから左右後輪の接地点に働く駆動力 [N]
$\tau_{L/R}$	:	左右モータから左右後輪に働くトルク [N · m]
N_B	:	本体 (重心) まわりに働くトルク [N · m]
M_B	:	本体の質量 [kg]
M_H	:	使用者の質量 [kg]
$R_{L/R}$	:	左右後輪の半径 [m]

D_{T}　：後輪の車軸幅 [m]
D_{W}　：前後車輪軸間の水平距離 [m]
μ_{F}　：左右前輪の転がり抵抗係数
μ_{R}　：左右後輪の転がり抵抗係数
V_{M}　：左右モータへ入力される中心電圧 [V]
$A_{\mathrm{M}}, B_{\mathrm{M}}$　：左右モータの性能を表す係数
I　：O' の原点における慣性モーメント $[\mathrm{kg \cdot m^2}]$
$(\bar{x}, \bar{y}, \bar{z})$　：O' における重心の座標 [m]
g　：重力加速度 $[\mathrm{m/s^2}]$

6.2.2　電動車いすの基本ダイナミクス

次の仮定をもとに EWC の基本ダイナミクスを求める.

(1) 本体は剛体とし，車輪にスプリングなどの緩衝装置はない,

(2) 車輪の横滑りはない,

(3) 前輪軸の回転に伴う走行抵抗は生じない,

(4) 空気抵抗は生じない,

(5) 転がり抵抗は各車輪の接地点に働く垂直抗力に比例する．ただし，車輪の変形はない,

(6) DC モータは瞬時に定常状態に達する.

はじめに，運動座標系 O'-xyz において考える．$\omega_{\mathrm{B}}, \dot{\omega}_{\mathrm{B}}, v_{\mathrm{y}}, a_{\mathrm{y}}$ はそれぞれ,

$$\omega_{\mathrm{B}} = (-R_{\mathrm{L}}\omega_{\mathrm{L}} + R_{\mathrm{R}}\omega_{\mathrm{R}})/D_{\mathrm{T}} \tag{6.1}$$

$$\dot{\omega}_{\mathrm{B}} = (-R_{\mathrm{L}}\dot{\omega}_{\mathrm{L}} + R_{\mathrm{R}}\dot{\omega}_{\mathrm{R}})/D_{\mathrm{T}} \tag{6.2}$$

$$v_{\mathrm{y}} = \frac{(D_{\mathrm{T}} - 2\bar{x})R_{\mathrm{L}}\omega_{\mathrm{L}} + (D_{\mathrm{T}} + 2\bar{x})R_{\mathrm{R}}\omega_{\mathrm{R}}}{2D_{\mathrm{T}}} \tag{6.3}$$

$$a_{\mathrm{y}} = \frac{(D_{\mathrm{T}} - 2\bar{x})R_{\mathrm{L}}\dot{\omega}_{\mathrm{L}} + (D_{\mathrm{T}} + 2\bar{x})R_{\mathrm{R}}\dot{\omega}_{\mathrm{R}}}{2D_{\mathrm{T}}} \tag{6.4}$$

と表される．このとき，重心における並進方向の運動方程式は次式となる.

$$(M_{\mathrm{B}} + M_{\mathrm{H}})a_{\mathrm{y}} = F_{\mathrm{L}} + F_{\mathrm{R}} \tag{6.5}$$

本体がロール角 ψ，ピッチ角 θ の状態であることを考えると，$F_{\mathrm{Gx}}, F_{\mathrm{Gy}}, F_{\mathrm{Gz}}$ はそれぞれ,

$$\begin{pmatrix} F_{\mathrm{Gx}} \\ F_{\mathrm{Gy}} \\ F_{\mathrm{Gz}} \end{pmatrix} = \begin{pmatrix} \sin\psi\cos\theta(M_{\mathrm{B}} + M_{\mathrm{H}})g \\ -\sin\theta(M_{\mathrm{B}} + M_{\mathrm{H}})g \\ -\cos\psi\cos\theta(M_{\mathrm{B}} + M_{\mathrm{H}})g \end{pmatrix} \tag{6.6}$$

となり，また O'-xyz は ω_{B} で回転しているため，F_{C} は,

$$\begin{aligned} F_{\mathrm{C}} &= (M_{\mathrm{B}} + M_{\mathrm{H}})\omega_{\mathrm{B}} v_{\mathrm{y}} \\ &= \frac{(M_{\mathrm{B}} + M_{\mathrm{H}})(-R_{\mathrm{L}}\omega_{\mathrm{L}} + R_{\mathrm{R}}\omega_{\mathrm{R}})}{2D_{\mathrm{T}}^2}\{(D_{\mathrm{T}} - 2\bar{x})R_{\mathrm{L}}\omega_{\mathrm{L}} + (D_{\mathrm{T}} + 2\bar{x})R_{\mathrm{R}}\omega_{\mathrm{R}}\} \end{aligned} \tag{6.7}$$

となる．式 (6.6) および (6.7) を用いて，$F_{\mathrm{LG}}, F_{\mathrm{RG}}, F_{\mathrm{LFN}}, F_{\mathrm{RFN}}, F_{\mathrm{LRN}}, F_{\mathrm{RRN}}$ はそれぞれ，

$$F_{\mathrm{LG}} = \frac{\bar{y}}{D_{\mathrm{T}}}(F_{\mathrm{Gx}} + F_{\mathrm{C}}) + \frac{D_{\mathrm{T}} - 2\bar{x}}{2D_{\mathrm{T}}}F_{\mathrm{Gy}} \tag{6.8}$$

$$F_{\mathrm{RG}} = -\frac{\bar{y}}{D_{\mathrm{T}}}(F_{\mathrm{Gx}} + F_{\mathrm{C}}) + \frac{D_{\mathrm{T}} + 2\bar{x}}{2D_{\mathrm{T}}}F_{\mathrm{Gy}} \tag{6.9}$$

$$F_{\mathrm{LFN}} = -(\frac{\bar{y}}{D_{\mathrm{W}}})(\frac{D_{\mathrm{T}} - 2\bar{x}}{2D_{\mathrm{T}}})F_{\mathrm{Gz}} - (\frac{\bar{y}}{D_{\mathrm{W}}})(\frac{2\bar{z} + R_{\mathrm{L}} + R_{\mathrm{R}}}{2D_{\mathrm{T}}})(F_{\mathrm{Gx}} + F_{\mathrm{C}}) \tag{6.10}$$

$$F_{\mathrm{RFN}} = -(\frac{\bar{y}}{D_{\mathrm{W}}})(\frac{D_{\mathrm{T}} + 2\bar{x}}{2D_{\mathrm{T}}})F_{\mathrm{Gz}} + (\frac{\bar{y}}{D_{\mathrm{W}}})(\frac{2\bar{z} + R_{\mathrm{L}} + R_{\mathrm{R}}}{2D_{\mathrm{T}}})(F_{\mathrm{Gx}} + F_{\mathrm{C}}) \tag{6.11}$$

$$F_{\mathrm{LRN}} = -(\frac{D_{\mathrm{W}} - \bar{y}}{D_{\mathrm{W}}})(\frac{D_{\mathrm{T}} - 2\bar{x}}{2D_{\mathrm{T}}})F_{\mathrm{Gz}} - (\frac{D_{\mathrm{W}} - \bar{y}}{D_{\mathrm{W}}})(\frac{2\bar{z} + R_{\mathrm{L}} + R_{\mathrm{R}}}{2D_{\mathrm{T}}})(F_{\mathrm{Gx}} + F_{\mathrm{C}}) \tag{6.12}$$

$$F_{\mathrm{RRN}} = -(\frac{D_{\mathrm{W}} - \bar{y}}{D_{\mathrm{W}}})(\frac{D_{\mathrm{T}} + 2\bar{x}}{2D_{\mathrm{T}}})F_{\mathrm{Gz}} + (\frac{D_{\mathrm{W}} - \bar{y}}{D_{\mathrm{W}}})(\frac{2\bar{z} + R_{\mathrm{L}} + R_{\mathrm{R}}}{2D_{\mathrm{T}}})(F_{\mathrm{Gx}} + F_{\mathrm{C}}) \tag{6.13}$$

となる．また， 左右の独立した同性能の DC モータによって駆動力を得ることから，$F_{\mathrm{LI/RI}}$ は，

$$F_{\mathrm{LI/RI}} = \frac{\tau_{\mathrm{L/R}}}{R_{\mathrm{L/R}}} \tag{6.14}$$

$$\tau_{\mathrm{L/R}} = -A_{\mathrm{M}}\omega_{\mathrm{L/R}} + B_{\mathrm{M}}(V_{\mathrm{M}} - V_{\mathrm{L/R}}) \tag{6.15}$$

と表される．ここで，

$$F_{\mathrm{LT/RT}} = F_{\mathrm{LI/RI}} + F_{\mathrm{LG/RG}} \tag{6.16}$$

$$F_{\mathrm{LM/RM}} = \mu_{\mathrm{F}}F_{\mathrm{LFN/RFN}} + \mu_{\mathrm{R}}F_{\mathrm{LRN/RRN}} \tag{6.17}$$

とおくと，$F_{\mathrm{L/R}}$ は，

$$F_{\mathrm{L/R}} = \begin{cases} F_{\mathrm{LT/RT}} - F_{\mathrm{LM/RM}} & (\omega_{\mathrm{L/R}} > 0 \vee (\omega_{\mathrm{L/R}} = 0 \wedge F_{\mathrm{LT/RT}} > F_{\mathrm{LM/RM}})) \\ F_{\mathrm{LT/RT}} + F_{\mathrm{LM/RM}} & (\omega_{\mathrm{L/R}} < 0 \vee (\omega_{\mathrm{L/R}} = 0 \wedge F_{\mathrm{LT/RT}} < -F_{\mathrm{LM/RM}})) \\ 0 & (\omega_{\mathrm{L/R}} = 0 \wedge |F_{\mathrm{LT/RT}}| \leq |F_{\mathrm{LM/RM}}|) \end{cases} \tag{6.18}$$

となる．ここで，重心まわりの回転の運動方程式は，

$$\{I + (\bar{x}^2 + \bar{y}^2)(M_{\mathrm{B}} + M_{\mathrm{H}})\}\dot{\omega}_{\mathrm{B}} = N_{\mathrm{B}} \tag{6.19}$$

$$N_{\mathrm{B}} = \{-(D_{\mathrm{T}} + 2\bar{x})F_{\mathrm{L}} + (D_{\mathrm{T}} - 2\bar{x})F_{\mathrm{R}})\}/2 \tag{6.20}$$

と表され，式 (6.2) より，

$$\dot{\omega}_{\mathrm{L}} = A_1\dot{\omega}_{\mathrm{R}} + B_1 \tag{6.21}$$

となる．ただし，

$$A_1 = \frac{R_{\mathrm{R}}}{R_{\mathrm{L}}} \tag{6.22}$$

$$B_1 = \frac{D_{\mathrm{T}}(D_{\mathrm{T}} + 2\bar{x})}{2R_{\mathrm{L}}\{I + (\bar{x}^2 + \bar{y}^2)(M_{\mathrm{B}} + M_{\mathrm{H}})\}}F_{\mathrm{L}} - \frac{D_{\mathrm{T}}(D_{\mathrm{T}} - 2\bar{x})}{2R_{\mathrm{L}}\{I + (\bar{x}^2 + \bar{y}^2)(M_{\mathrm{B}} + M_{\mathrm{H}})\}}F_{\mathrm{R}} \tag{6.23}$$

である．また，式 (6.4) および (6.5) より，

$$\dot{\omega}_{\mathrm{R}} = A_2\dot{\omega}_{\mathrm{L}} + B_2 \tag{6.24}$$

となる．ただし，

$$A_2 = -\frac{(D_{\mathrm{T}} - 2\bar{x})R_{\mathrm{L}}}{(D_{\mathrm{T}} + 2\bar{x})R_{\mathrm{R}}} \tag{6.25}$$

$$B_2 = \frac{2D_{\mathrm{T}}}{(M_{\mathrm{B}} + M_{\mathrm{H}})R_{\mathrm{R}}(D_{\mathrm{T}} + 2\bar{x})}(F_{\mathrm{L}} + F_{\mathrm{R}}) \tag{6.26}$$

である．よって，式 (6.21) および (6.24) より，

$$\dot{\omega}_{\mathrm{L}} = \frac{A_1B_2 + B_1}{1 - A_1A_2} \tag{6.27}$$

$$\dot{\omega}_{\mathrm{R}} = \frac{A_2B_1 + B_2}{1 - A_1A_2} \tag{6.28}$$

となる．

つぎに，静止座標系 O-XYZ において考える．$v_{\mathrm{y}}, \omega_{\mathrm{B}}$ を O-XYZ へ座標変換すると，$v_{\mathrm{X}}, v_{\mathrm{Y}}$ は，

$$\begin{pmatrix} v_{\mathrm{X}} \\ v_{\mathrm{Y}} \end{pmatrix} = \begin{pmatrix} -\sin\delta\cos\theta v_{\mathrm{y}} \\ \cos\delta\cos\theta v_{\mathrm{y}} \end{pmatrix} \tag{6.29}$$

となり，ω_{BZ} は，

$$\omega_{\mathrm{BZ}} = \cos\theta\cos\psi\omega_B \tag{6.30}$$

となる．

そして次節では，電動車いすのダイナミクスを考慮して，使用者の操作負担を軽減するために，直進走行システムを提案する．

6.3　直進走行システム

6.3.1　システムの概要

ここでは，状態フィルタを用いた強化学習手法によって，傾斜路面においても適応的に電動車いすの直進走行を可能とするシステムを提案する．ただし，使用者の安全性を考慮し，

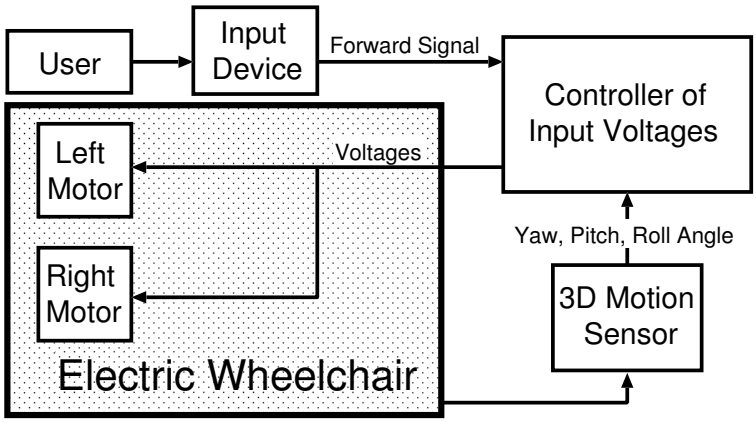

図 6.2: 直進走行システムの構成.

使用者が操作スイッチやジョイスティックなどの入力装置によって前進信号を入力している場合においてのみ作動するシステムとする.

システムは，次の 5 つのモジュールから構成される.

(1) 使用者,

(2) 入力装置,

(3) 左右の DC モータによって駆動力を得る後輪駆動型電動車いす (EMC-230，(株) 今仙技術研究所),

(4) 電動車いす本体のヨー角やピッチ角，ロール角を検知できる 3D モーションセンサ ((Vector Cube(VC-03)，(株) 電通企工)(以下，3D センサ),

(5) 入力電圧調整器 (以下，調整器).

図 6.2 にシステムの構成を示す．なお，3D センサは，ヨー角，ピッチ角，ロール角をそれぞれ $[-180°, 179°]$，$[-90°, 90°]$，$[-90°, 90°]$ の整数値を分解能 1° にて検知できる．EMC-230 は，左右モータの入力電圧 $V_{\mathrm{L/R}} = 2.5[\mathrm{V}]$ にて静止状態，$V_{\mathrm{L/R}} = [0.5, 4.5][\mathrm{V}]$ として，$V_{\mathrm{L/R}} < 2.5[\mathrm{V}]$ で前進する．また，平坦路面走行時において $V_{\mathrm{L/R}} = 0.5[\mathrm{V}]$ のとき，定常状態における最高速度が 6.0[km/h] となるように設定されている.

ここで，入力装置から前進信号が出力されている際に，3D センサから電動車いすとしてヨー角，ピッチ角，ロール角が調整器に入力され，調整器から左右のモータに電圧が出力される．そして，その電圧に基づいて電動車いすが走行し，結果として，3D センサからヨー角が調整器に入力され，調整器のもつ制御ルールが更新される.

調整器は，3D センサから入力される電動車いすの情報に基づいて，適切な電圧をモータに出力する．調整器には，次の 3 つの要件が課せられる.

(1) 使用者一人一人に合わせた適切な制御ルールが実現されること,

(2) 未経験の路面を走行する場合や使用者の重心位置，車輪の空気圧が変化した場合など，使用者を含む EWC の特性が変化した場合においても，適切な制御ルールが維持されること，

(3) 使用者を含む電動車いすに関して，センシングする情報が部分的であっても適切な制御ルールが構築されること．

上記の要件を満たすため，5 章で記述した状態フィルタを用いた強化学習手法を調整器に適用することにする．しかしながら，一般的に強化学習手法を用いる際には，次のことに注意する必要がある．すなわち，適切な制御ルール獲得のためには，

(1) 多くの試行錯誤が必要になる，

(2) 適切でない電圧も多く出力される．

しかし，ここでは，次のようにシステムを限定することで，上記の問題を回避している．

- 直進走行時に使用するといった多くの回数使用されると考えられるシステムとする，
- モータへ出力する電圧を制限する，

つぎに，状態フィルタを用いた強化学習手法の概要と設定について述べる．

6.3.2　強化学習手法の設定

設定

行動空間の設定として，事前に，直進走行時の速度を考慮にいれて，ここでは，片方のモータへの入力電圧を 1.5[V] として，横断傾斜 10° において直進走行を可能とするもう一方のモータへの入力電圧 V_{P} を調べておくことにする．なお，直進走行時の最高速度を 3.0[km/h](最高速度の半分) とした．そして，行動空間をできるだけ小さくするように，行動空間を以下の 7 つの入力電圧によって構成する．

(1) $(V_{\mathrm{L}}, V_{\mathrm{R}}) = (1.5, 1.5)$

(2) $(V_{\mathrm{L}}, V_{\mathrm{R}}) = (1.5, (V_{\mathrm{P}} + 1.5)/4)$

(3) $(V_{\mathrm{L}}, V_{\mathrm{R}}) = (1.5, (V_{\mathrm{P}} + 1.5)/2)$

(4) $(V_{\mathrm{L}}, V_{\mathrm{R}}) = (1.5, V_{\mathrm{P}})$

(5) $(V_{\mathrm{L}}, V_{\mathrm{R}}) = ((V_{\mathrm{P}} + 1.5)/4, 1.5)$

(6) $(V_{\mathrm{L}}, V_{\mathrm{R}}) = ((V_{\mathrm{P}} + 1.5)/2, 1.5)$

(7) $(V_{\mathrm{L}}, V_{\mathrm{R}}) = (V_{\mathrm{P}}, 1.5)$

ここで，調整器によって，電動車いすができるだけ直進経路を進むような制御ルールを獲得させることも考えられるが，そのためにはセンシングする情報として直進経路との距離

(ズレ幅) が必要である．しかし，直進経路との距離をセンシングすることは現実的には容易でないため，ここでは，調整器によって，電動車いすのヨー角が目標ヨー角をできるだけ維持するような制御ルールを獲得させることを考える．よって報酬は，3D センサから入力されるヨー角の値をもとに，式 (6.31) に示される即時報酬を用いることにする．

$$r(t) = \frac{5.0}{\exp\left(\delta(t) - \delta_{\mathrm{T}}\right)^2} - 2.5 \tag{6.31}$$

ここでは，目標ヨー角とヨー角との差の絶対値が小さいほど報酬が大きくなるように設定した．ただし，$\delta(t)$ をステップ t におけるヨー角を示す．また，δ_{T} を使用者がはじめに前進信号を入力した際のヨー角とし，これを目標ヨー角とする．ここで，使用者が入力装置によって本体のヨー角を修正した場合，目標ヨー角 δ_{T} を修正されたヨー角に更新する．なお，使用者が本体のヨー角を修正した場合として，入力装置が操作スイッチであれば，前進信号以外が選択された場合とし，入力装置がジョイスティックであれば，ジョイスティックが明らかに前方向に倒されていない (例えば，左/右回転信号が入力された) 場合とする．

学習器には学習に要する計算時間が短い Sarsa[1)](学習率 $\alpha = 0.5$，割引率 $\gamma = 0.9$，行動選択確率に温度係数 $\tau = 0.2$ のボルツマン選択法) を，そして効率的に Q 値を更新するため，新たに生成される内部状態の Q 値を含めたすべての Q 値の初期値にオプティミスティック初期値 [1)] として 25.0 を用いる．

状態フィルタを通した内部状態空間は，ヨー角，ピッチ角，ロール角の次元をそれぞれ $[-180, 180)$，$[-90, 90.1)$，$[-90, 90.1)$ の区間で構成される空間とした．そして，状態フィルタのパラメータとして，行動学習回数の閾値 $\theta_{\mathrm{L}} = 100$，エントロピーの閾値 $\theta_{\mathrm{H}} = 0.26$，分割される区間幅についての閾値は用いるセンサの分解能より $\theta_{\mathrm{r}} = 1.0$，そして内部状態の統合・削除を行う期間 $\theta_{\mathrm{t}} = 100$[episode] を用いる．ただし，エントロピーの閾値は，行動数 7 における 1 つの行動選択確率が 0.9 の時のエントロピーの最大値が約 0.259 であることを参考に設定した．

初期の集約状態空間は，ヨー角，ピッチ角，ロール角の次元をそれぞれ $-5, 0, 5$ で通常分割したものを用いることとする．つまり，初期の内部状態空間のサイズは 4^3 とする．

つぎに，2 節で求めたダイナミクスに基づいたシミュレーションによって，

(1) 傾斜角が一定な路面を走行する場合，

(2) 傾斜角が一定でない路面を走行する場合，

さらに補足として，

(3) 重心位置が変化する場合，

(4) 2 つの路面を走行する場合，

について，提案システムの有効性，実際的な適用可能性を確認する．

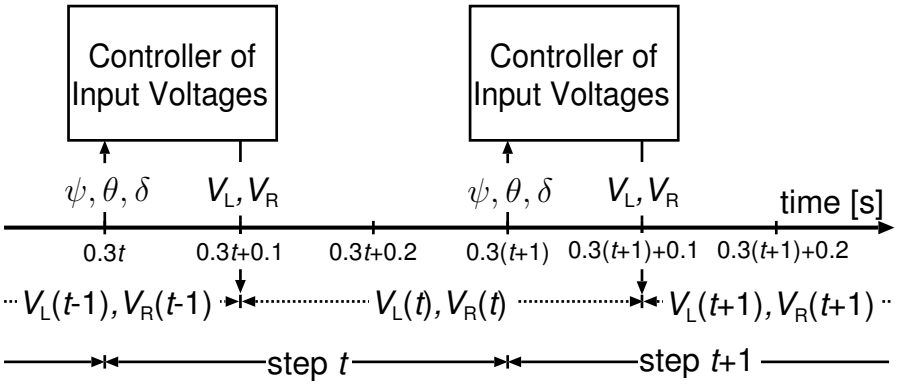

図 6.3: 入力電圧の時間変化

6.4　シミュレーション

6.4.1　シミュレーションの設定

電動車いすのスタート状態を $(X, Y) = (0.0, 0.0)$ にて停止状態，目標ヨー角 $\delta_{\mathrm{T}} = 0.0$ とし，ゴール状態を $Y > 20.0$ とする．また，電動車いすの位置が $X > 5.0$ または $X < -5.0$ のとき，使用者が EWC を停止状態にさせ，そして $(X, Y) = (0.0, 20.0)$ の方向に本体のヨー角を修正することとする．

調整器が状態フィルタや学習器の調整を行う時間を考慮して，3D センサから調整器に電動車いすの情報が入力された後，0.1[s] 後に，調整器が左右モータに電圧を出力するものとし，電圧出力後 0.2[s] 後に再び 3D センサから調整器に電動車いすの情報が入力される．以上の 0.3[s] を 1 ステップとする．つまり，3D センサから調整器に電動車いすの情報が入力された後の 0.1[s] 間は前のステップで出力した電圧に基づいて電動車いすは走行し続ける．そして，調整器は 0.1[s] 前の電動車いすの情報に基づいて左右モータに電圧を出力することとなる．また，電動車いすがスタート状態からゴール状態に到達するまでを 1 エピソードとする．図 6.3 に入力電圧の時間変化を示す．

電動車いすモデルのパラメータは，EMC-230 を想定し，表 6.1 に示すように設定した．ただし，モータの性能を表す係数 $A_{\mathrm{M}}, B_{\mathrm{M}}$ は，JIS 規格 (JIS T 9203:2006) に合うように設定した．

そして予備実験として，重心位置 $(\bar{x}, \bar{y}, \bar{z}) = (0.0, 0.265, 0.178)$ とした電動車いすにて，横断傾斜 10° において直進走行を可能とするモータへの入力電圧 V_{P} を求め，$V_{\mathrm{P}} = 2.02$ とした．

表 6.1: 電動車いすのパラメータ

Parameters	Value
$V_{\mathrm{L/R}}$	[0.5,4.5][V]
M_{B}	80.0[kg]
M_{H}	75.0[kg]
$R_{\mathrm{L/R}}$	0.165[m]
D_{T}	0.532[m]
D_{W}	0.475[m]
$\mu_{\mathrm{F/R}}$	0.02
V_{M}	2.5[V]
A_{M}	16.7
B_{M}	85.9
I	$10.0[\mathrm{kg}\cdot\mathrm{m}^2]$

6.4.2 実験 (1)：傾斜角が一定な路面

横断傾斜角 1° とする路面を走行する場合，$(\bar{x},\bar{y},\bar{z}) = (0.0, 0.265, 0.178)$ のときについて，それぞれ乱数の種を変えて 20 回実験を行った．学習速度と制御ルールの良さ，内部状態空間のサイズ，学習に要した計算時間について，提案システム (以下，手法 A) と観測状態空間をそれぞれの次元均等に $4, 8, 16$ 分割する状態フィルタ (以下，それぞれ手法 4，8，16) を用いた場合との比較検討を行う．つぎに，状態フィルタの調整に用いられる履歴情報の深さ，最終的に獲得した制御ルールについて検討を行い，提案システムの実機への適用可能性を確認する．

実験結果を図 6.4〜図 6.7，表 6.2，に示す．なお，

- 図 6.4 に目標ヨー角との差の最大値の変化，
- 図 6.5 に内部状態空間のサイズの変化，
- 図 6.6 に履歴情報の深さの変化，
- 図 6.7 に最終的に獲得した制御ルールにおける，目標ヨー角との差の最大値が最小の場合 (以下，ケース best) と最大の場合 (以下, ケース worst) の走行軌跡，
- 表 6.2 に 1 ステップの学習に要した平均計算時間 (Intel Pentium4 3.20GHz, メモリ 1GB, Vine Linux 3.2, java1.5.0 使用時),

を示す．ここで，実験結果は図 6.7 を除いて 20 回の実験の平均である．なお，図 6.7 には，比較のため，調整器を用いない場合として，

- $V_{\mathrm{L/R}} = 1.5$(前進信号) に固定，$(\bar{x},\bar{y},\bar{z}) = (0.0, 0.265, 0.178)$ のとき (以下，手法 B),

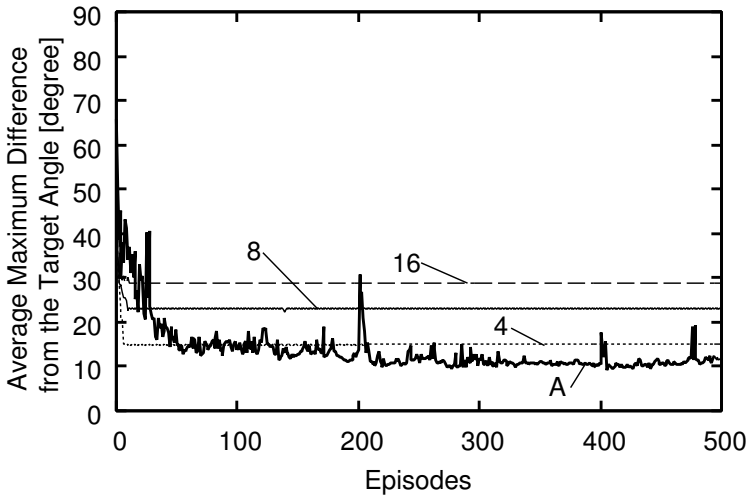

図 6.4: 実験 (1) における目標ヨー角との差の最大値の変化

表 6.2: 実験 (1) において 1 ステップの学習に要した計算時間

Method	Computational time [ms]
A	0.026

についての走行軌跡を合わせて示す.

各検討項目に対して，次のようなことが確認できる.

- 検討項目 (1)：学習速度と制御ルールの良さについて，図 6.4 より，手法 A は,
  - 学習速度について,
    - 手法 4，8，16 に比べると遅いが，約 50[episode] 目には目標ヨー角との差の最大値が約 15° になっており，学習初期の段階で良い性能を示していること,
    - その後時間をかけて徐々に制御ルールを改善していること,
  - 制御ルールの良さについて,
    - 手法 4，8，16 に比べると，良い性能を示していること,
    - 目標ヨー角との差の最大値が 30° 以下にはならなかったこと,
- 検討項目 (2)：内部状態空間のサイズについて図 6.5 より，手法 A は,
  - 最終的には手法 4 よりも小さいこと,
- 検討項目 (3)：履歴情報の深さについて図 6.6 より，手法 A は,
  - 最終的に約 1.2 ステップ過去までの履歴情報を記録していること,
  - つまり多くとも 1 ステップ過去までの履歴情報を参照していること,
- 検討項目 (4)：最終的に獲得した制御ルールについて図 6.7 より，手法 A は,

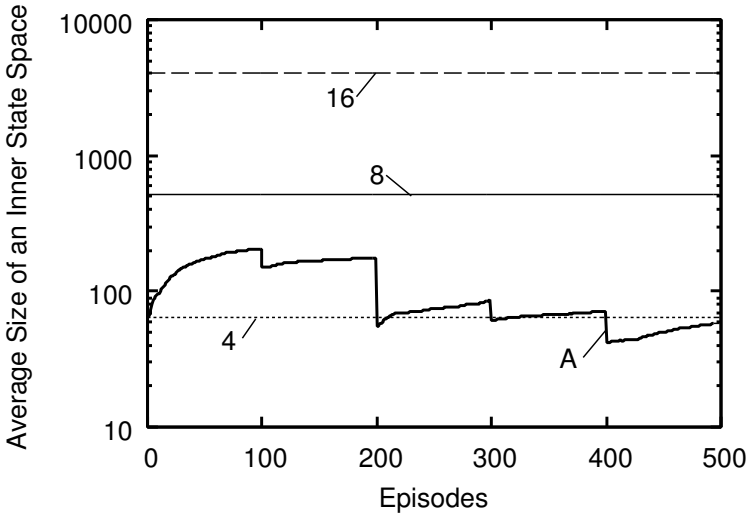

図 6.5: 実験 (1) における内部状態空間のサイズの変化

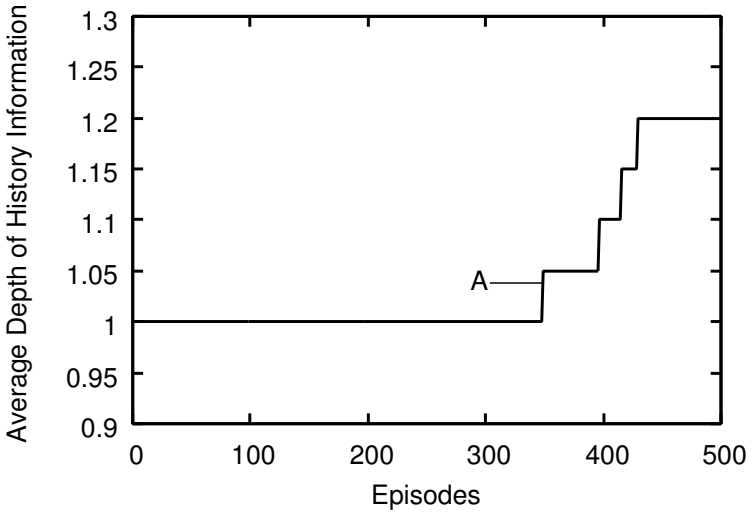

図 6.6: 実験 (1) における履歴情報の深さの変化

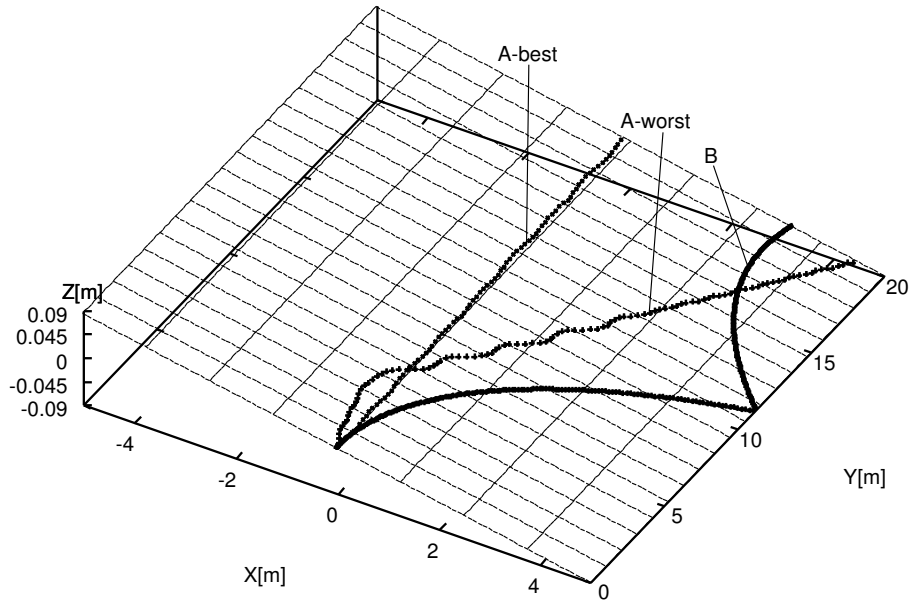

図 6.7: 実験 (1) において提案システムが獲得した目標ヨー角との差の最大値が最小の場合と最大の場合の制御ルールによる走行軌跡と直進信号 $V_{\mathrm{L/R}} = 1.5$ のみによる走行軌跡

  - 手法 B と比較すると，ケース worst においても，良い性能を示していること，
  - ケース best において，直進走行を，完全ではないものの，ほぼ実現できていること，

- 検討項目 (5)：計算時間について，表 6.2 より，
  - 手法 A は約 0.025[ms] であり，シミュレーションで用いた計算時間 0.1[s] の 1/4000 倍に抑えられていることから，計算時間は十分に小さいこと．

ここで，手法 A について目標ヨー角との差の最大値が 10° 以下にはならなかったことについて，行動空間が予め決められた離散的な値によって構成されているためであると考えられる．

また，実験 (1) において手法 A により獲得された状態フィルタの一例について図 6.8〜図 6.10 に示す．なお，

- 図 6.8 に目標ヨー角との差が-10° における状態フィルタ,
- 図 6.9 に目標ヨー角との差が 0° における状態フィルタ,
- 図 6.10 に目標ヨー角との差が 10° における状態フィルタ,

を示す．ただし，示した状態フィルタについて，１つの格子が 1 つの集約状態を表し，格子内の数字はその格子で表される集約状態に対して用いる履歴情報の深さを表す．なお，集約状態における履歴情報の深さが 0 の場合は数字を省略した．図 6.8〜図 6.10 より，手法 A によって獲得された状態フィルタの一例は，ピッチ角，ロール角が 0 度に近付くほど細かく分割されていることが確認できる．

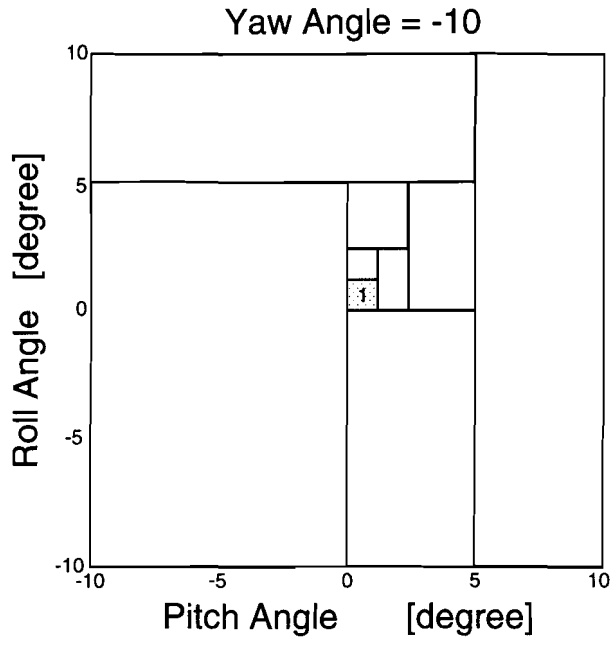

図 6.8: 実験 (1) において手法 A により獲得された状態フィルタの一例 (目標ヨー角との差：-10)

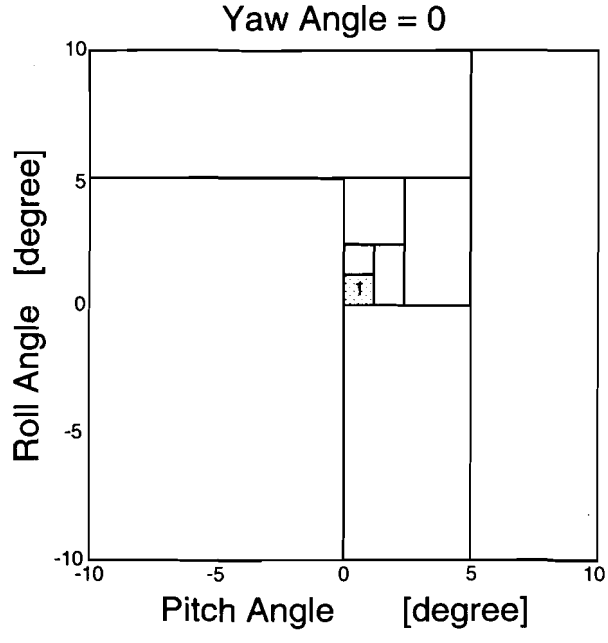

図 6.9: 実験 (1) において手法 A により獲得された状態フィルタの一例 (目標ヨー角との差：0)

以上より，提案システムの有効性と実機への適用可能性をそれぞれ確認できた．特に，提案システムを用いることによって使用者の操作負担を軽減できることが確認できた．

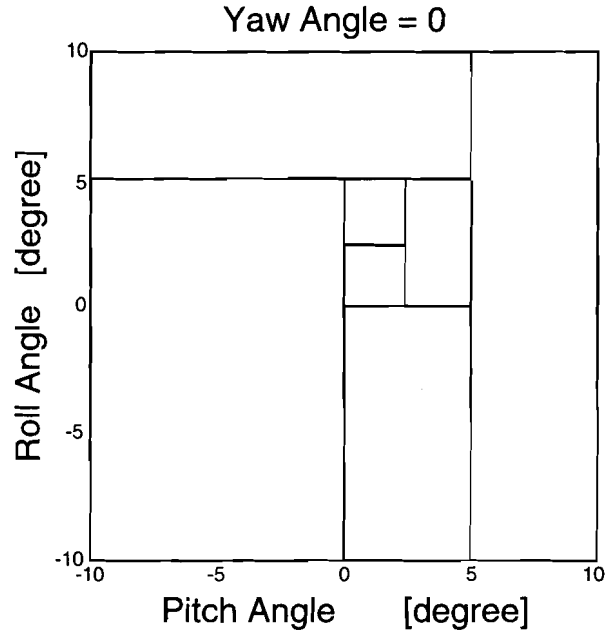

図 6.10: 実験 (1) において手法 A により獲得された状態フィルタの一例 (目標ヨー角との差：10)

6.4.3　実験 (2)：傾斜角が一定でない路面

式 (6.32) に示される傾斜角が一定でない路面を考える．ただし，電動車いすの最大勾配が 10 ° (電動車いすが走行可能でなければならない最大勾配 (JIS 規格 (JIS T 9203:2006))) となるように $\sigma = 0.82$ と設定し，直進走行を獲得させるには考え得る限り困難な路面とした．

$$f(x,y) = \frac{1}{2\pi\sigma^2}\left\{\exp\left(-\frac{(x+0.5)^2+(y-5.0)^2}{2\sigma^2}\right)\right.$$
$$\left.+\exp\left(-\frac{(x-0.5)^2+(y-15.0)^2}{2\sigma^2}\right)\right\} \tag{6.32}$$

上記の傾斜角が一定でない路面を走行する場合，$(\bar{x},\bar{y},\bar{z}) = (0.0, 0.265, 0.178)$ のときについて，それぞれ乱数の種を変えて 20 回実験を行った．学習速度と制御ルールの良さ，内部状態空間のサイズについて，提案システム (以下，手法 A) と観測状態空間をそれぞれの次元均等に 16, 32, 64 分割する状態フィルタ (以下，それぞれ手法 16，32，64) を用いた場合との比較検討を行う．つぎに，状態フィルタの調整に用いられる履歴情報の深さ，最終的に獲得した制御ルール，学習に要した計算時間について検討を行い，提案システムの実機への適用可能性を確認する．

実験結果を図 6.11 ～図 6.14，表 6.3，に示す．なお，

- 図 6.11 に目標ヨー角との差の最大値の変化，
- 図 6.12 に内部状態空間のサイズの変化，

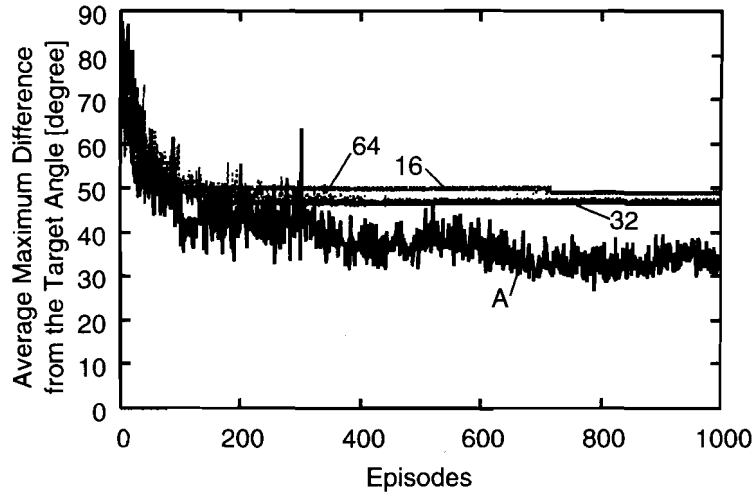

図 6.11: 実験 (2) における目標ヨー角との差の最大値の変化

表 6.3: 実験 (2) において 1 ステップの学習に要した計算時間

Method	Computational time [ms]
A	0.023

- 図 6.13 に履歴情報の深さの変化,
- 図 6.14 に最終的に獲得した制御ルールにおいて，目標ヨー角との差の最大値が最小の場合 (以下，ケース best) と最大の場合 (以下, ケース worst) の走行軌跡,
- 表 6.3 に 1 ステップの学習に要した平均計算時間 (実験 (1) と同じ PC 使用時),

を示す．ここで，実験結果は図 6.14 を除いて 20 回の実験の平均である．なお，図 6.14 には，実験 (1) と同様に比較のため，調整器を用いない場合として,

- $V_{\mathrm{L/R}} = 1.5$(前進信号) に固定，$(\bar{x}, \bar{y}, \bar{z}) = (0.0, 0.265, 0.178)$ のとき (以下，手法 B),

についての走行軌跡を加えて示す.

各検討項目に対して，次のようなことが確認できる.

- 検討項目 (1)：学習速度と制御ルールの良さについて図 6.11 より，手法 A は,
  - 学習速度について,
    - 約 100[episode] 目には目標ヨー角との差の最大値が約 40°, になっており，学習初期の段階で良い性能を示していること,
    - その後時間をかけて徐々に制御ルールを改善していること,
  - 制御ルールの良さについて,
    - 手法 16，32，64 に比べると，良い性能を示していること,

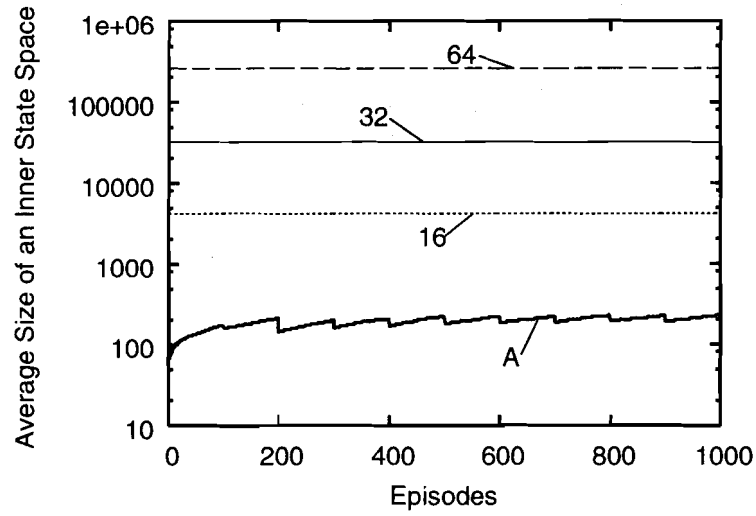

図 6.12: 実験 (2) における内部状態空間のサイズの変化

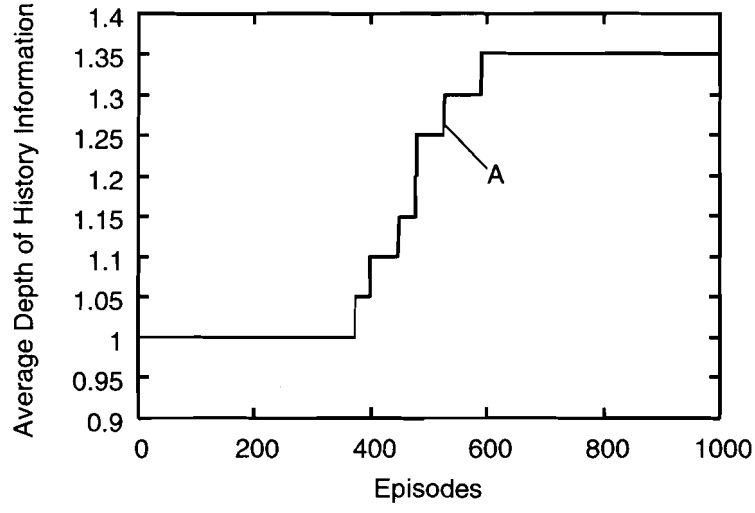

図 6.13: 実験 (2) における履歴情報の深さの変化

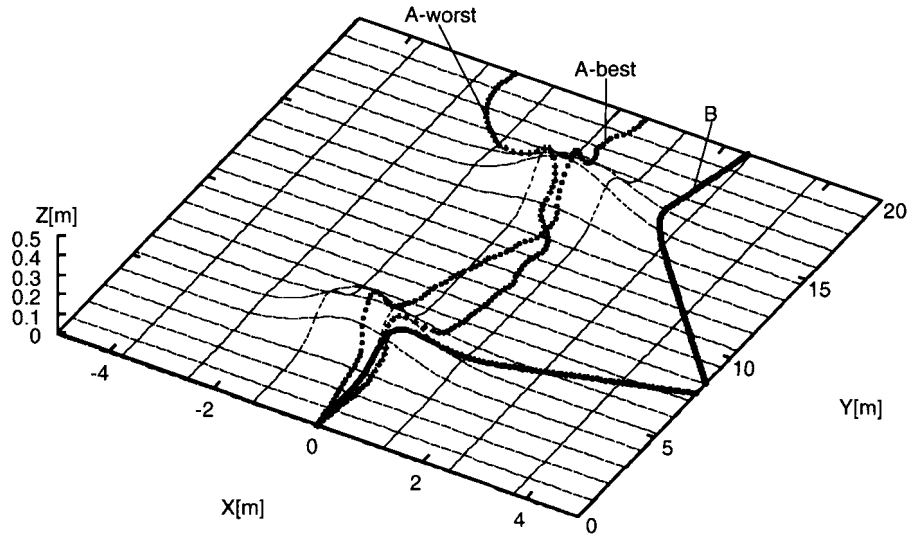

図 6.14: 実験 (2) において提案システムが獲得した目標ヨー角との差の最大値が最小の場合と最大の場合の制御ルールによる走行軌跡と直進信号 $V_{\mathrm{L/R}} = 1.5$ のみによる走行軌跡

　　* 目標ヨー角との差の最大値が 10° 以下にはならなかったこと，

- 検討項目 (2)：内部状態空間のサイズについて図 6.12 より，手法 A は，
  - 手法 16 よりも小さいこと，
- 検討項目 (3)：履歴情報の深さについて図 6.13 より，手法 A は，
  - 最終的に約 1.35 ステップ過去までの履歴情報を記録していること，
  - つまり多くとも 1 ステップ過去までの履歴情報を参照していること，
- 検討項目 (4)：最終的に獲得した制御ルールについて図 6.14 より，実験 (1) と同様に手法 A は，
  - 手法 B と比較すると，ケース worst においても，良い性能を示していること，
  - ケース best において，直進走行を，完全ではないものの，ほぼ実現できていること，
- 検討項目 (5)：計算時間表 6.2 より，実験 (1) と同様に手法 A は，
  - 約 0.025[ms] であり，シミュレーションで用いた計算時間 0.1[s] の 1/4000 倍に抑えられていることから，計算時間は十分に小さいこと．

また，実験 (2) において手法 A により獲得された状態フィルタの一例について図 6.15〜図 6.17 に示す．なお，

- 図 6.15 に目標ヨー角との差が-10° における状態フィルタ，
- 図 6.16 に目標ヨー角との差が 0° における状態フィルタ，

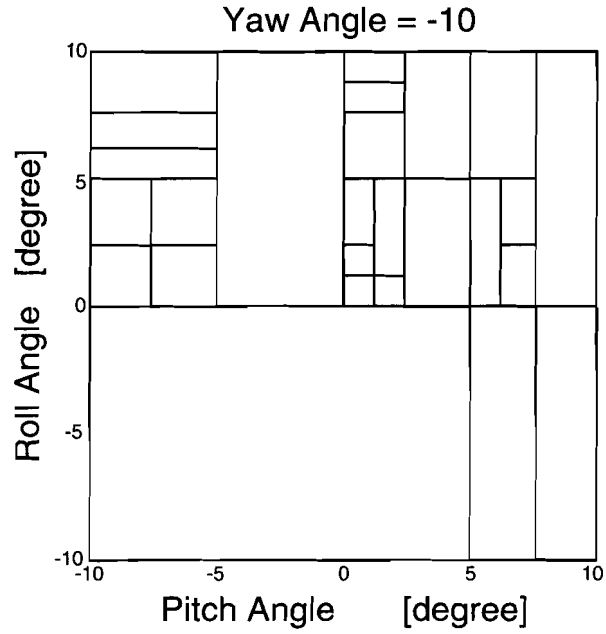

図 6.15: 実験 (2) においてケース A により獲得された状態フィルタの一例 (目標ヨー角との差：-10)

- 図 6.17 に目標ヨー角との差が 10° における状態フィルタ,

を示す．ただし，示した状態フィルタについて実験 (1) と同様に，1 つの格子が 1 つの集約状態を表し，格子内の数字はその格子で表される集約状態に対して用いる履歴情報の深さを表す．なお，集約状態における履歴情報の深さが 0 の場合は数字を省略した．図 6.15～図 6.17 より，手法 A によって獲得された状態フィルタの一例は，次のことが確認できる．

- 実験 (1) において獲得された状態フィルタの一例と同様に，ピッチ角，ロール角が 0 度に近付くほど細かく分割されていること,
- 図 6.8～図 6.10 と比較すると，実験 (1) において獲得された状態フィルタの一例よりも，細かく分割されていること．

以上より，傾斜角が一定でない路面において，提案システムの有効性，実機への適用可能性をそれぞれ確認できた．特に，傾斜角が一定でない路面において，単純なフィードバック制御を基本としたシステムでは，直進走行が困難と考えられる．しかし，提案システムでは，傾斜角が一定である路面と同様に，提案システムを用いることによって使用者の操作負担を軽減できることが確認できた．

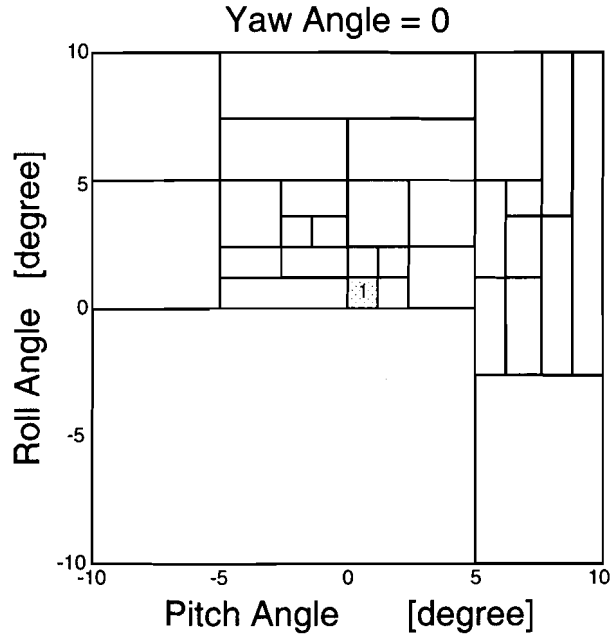

図 6.16: 実験 (2) においてケース A により獲得された状態フィルタの一例 (目標ヨー角との差：0)

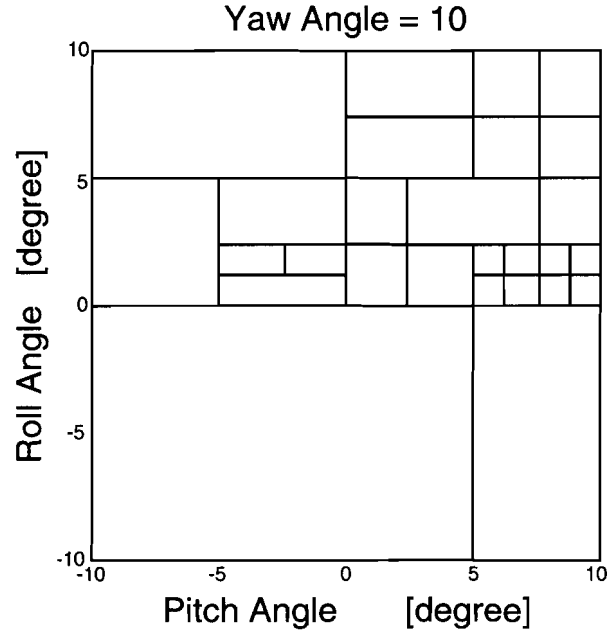

図 6.17: 実験 (2) においてケース A により獲得された状態フィルタの一例 (目標ヨー角との差：10)

6.4.4 補足実験 (1)：重心位置が変化

まず補足実験 (1) として，提案システムを用いて，横断傾斜角 1° とする路面において，重心位置が変化する場合，

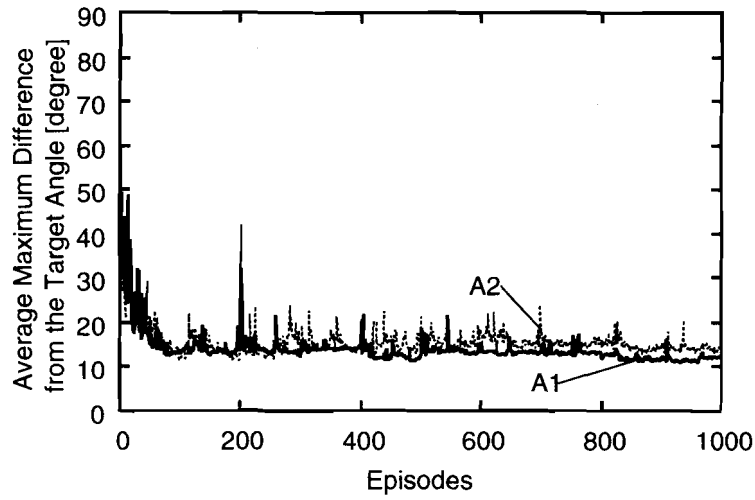

図 6.18: 補足実験 (1) における目標ヨー角との差の最大値の変化

- 500[episode] の開始時において，$(\bar{x}, \bar{y}, \bar{z}) = (0.0, 0.265, 0.178)$ から $(\bar{x}, \bar{y}, \bar{z}) = (0.05, 0.265, 0.178)$ に変化するとき (以下，ケース A1)，
- 250[episode] の開始時から 750[episode] の開始時において，段階的に $(\bar{x}, \bar{y}, \bar{z}) = (0.0, 0.265, 0.178)$ から $(\bar{x}, \bar{y}, \bar{z}) = (0.05, 0.265, 0.178)$ に移動するとき (以下，ケース A2)，

について，それぞれ乱数の種を変えて 20 回実験を行った．重心位置変化時の制御ルールの良さ，内部状態空間のサイズ，履歴情報の深さについて検討を行う．

実験結果を図 6.18，図 6.19，図 6.20 に示す．なお，

- 図 6.18 に目標ヨー角との差の最大値の変化，
- 図 6.19 に内部状態空間のサイズの変化，
- 図 6.20 に履歴情報の深さの変化，

を示す．ここで，実験結果はすべて 20 回の実験の平均である．

各検討項目に対して，次のようなことが確認できる．

- 検討項目 (1)：路面変化時の制御ルールの良さについて図 6.18 より，
  - ケース A1，A2 共に重心位置の変化に対して良い性能を示していること，
- 検討項目 (2)：内部状態空間のサイズについて図 6.19 より，
  - ケース A1，A2 も共に 220 未満であること．
- 検討項目 (3)：履歴情報の深さについて図 6.20 より，ケース A1，A2 は，
  - それぞれ最終的に約 1.5，約 5 ステップ過去までの履歴情報を記録していること，
  - つまりそれぞれ多くとも 1，4 ステップ過去までの履歴情報を参照していること．

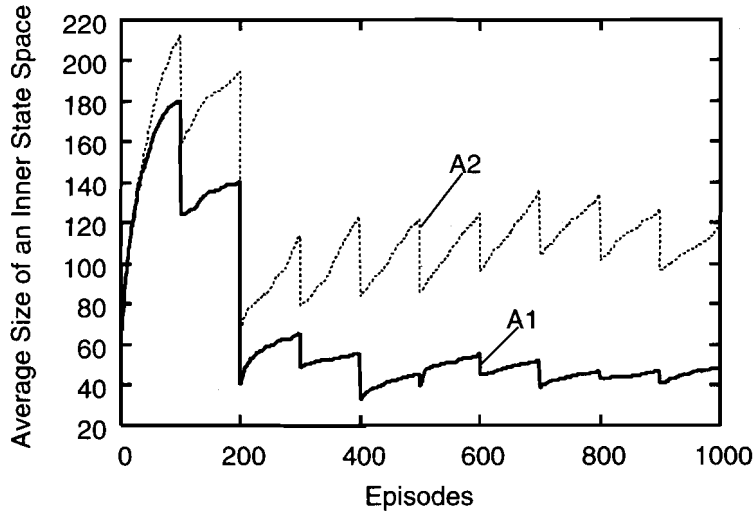

図 6.19: 補足実験 (1) における内部状態空間のサイズの変化

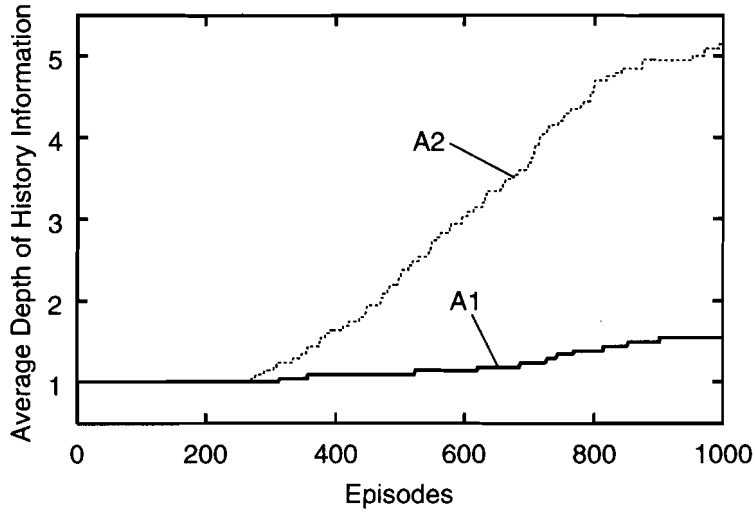

図 6.20: 補足実験 (1) における履歴情報の深さの変化

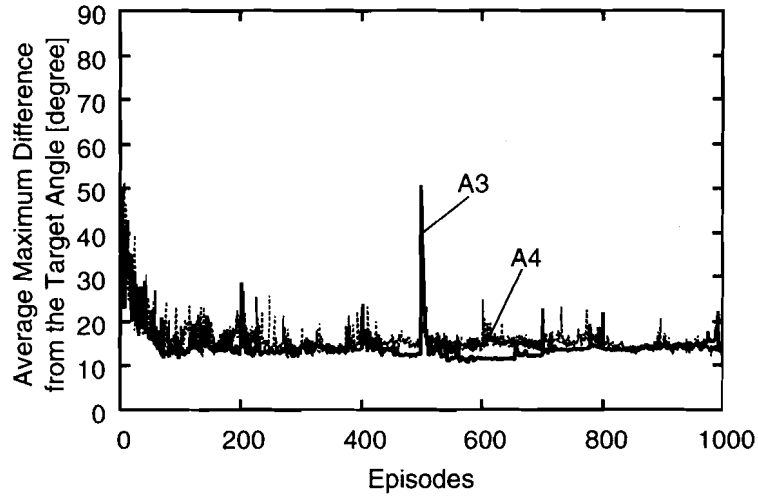

図 6.21: 補足実験 (2) における目標ヨー角との差の最大値の変化

以上より，重心位置が変化する場合においても，提案システムの有効性，実機への適用可能性をそれぞれ確認できた．特に，重心位置の変化に対して，すぐに良い制御ルールを獲得することが確認できた．

6.4.5　補足実験 (2)：2 つの路面

さらに，補足実験 (2) として，提案システムを用いて，2 つの路面を走行する場合，

- 500[episode] の開始時において，横断傾斜角 1° とする路面から横断傾斜角 −1° とする路面に移るとき (以下，ケース A3)，
- 同一路面の往復を想定し，横断傾斜角 1° とする路面と横断傾斜角 −1° とする路面が 1[episode] ずつ入れ替わるとき (以下，ケース A4)，

について，それぞれ乱数の種を変えて 20 回実験を行った．ただし，重心位置は一定で $(\bar{x}, \bar{y}, \bar{z}) = (0.0, 0.265, 0.178)$ とした．路面変化時の制御ルールの良さ，内部状態空間のサイズ，履歴情報の深さについて検討を行う．

実験結果を図 6.21，図 6.22，図 6.23 に示す．なお，

- 図 6.21 に目標ヨー角との差の最大値の変化，
- 図 6.22 に内部状態空間のサイズの変化，
- 図 6.23 に履歴情報の深さの変化，

を示す．ここで，実験結果はすべて 20 回の実験の平均である．

各検討項目に対して，次のようなことが確認できる．

- 検討項目 (1)：路面変化時の制御ルールの良さについて図 6.21 より，

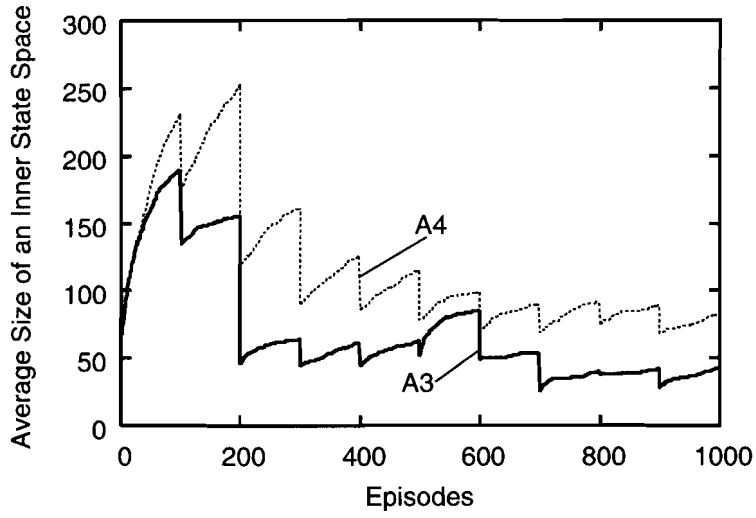

図 6.22: 補足実験 (2) における内部状態空間のサイズの変化

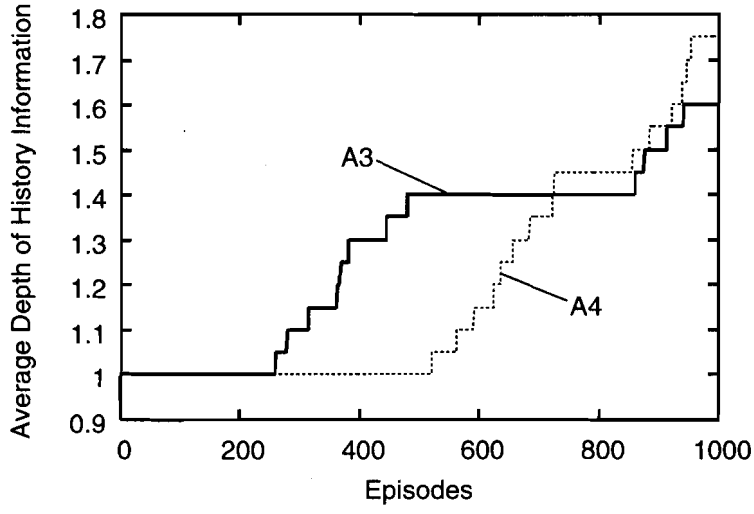

図 6.23: 補足実験 (2) における履歴情報の深さの変化

- ケース A3 は路面が変化した際に，一旦制御ルールの性能が落ちるものの，すぐに良い制御ルールを獲得していること，
- ケース A4 は実験 (1) と同様の性能を示していること，

- 検討項目 (2)：内部状態空間のサイズについて図 6.22 より，
  - ケース A3，A4 も共に約 250 未満であること，
  - ケース A3 に比べ，ケース A4 はエピソード毎に入れ替わって異なる路面を走行しているため，内部状態のサイズも約 2 倍であること．
- 検討項目 (3)：履歴情報の深さについて，図 6.23 より，ケース A3，A4 は，
  - それぞれ最終的に約 1.6，約 1.75 ステップ過去までの履歴情報を記録していること，
  - つまり，多くとも 1 ステップ過去までの履歴情報を参照していること．

以上より，2 つの路面を走行する場合においても，提案システムの有効性，実機への適用可能性をそれぞれ確認できた．特に，異なる路面を走行した際に，一旦性能が落ちるものの，すぐに良い制御ルールを獲得することが確認できた．

6.5　結言

本章では，第 5 章で提案した手法のより実際的な問題への適用例として，電動車いすの直進走行システムを取り上げた．ここではまず，左右 DC モータから駆動力を得る後輪駆動型電動車いすを対象として，三次元上における電動車いすの基本ダイナミクスのモデル化を行った．つぎに，基本的なダイナミクスを考慮した上，電動車いすの直進走行システムとして，第 5 章で提案した手法を再構築し，適用することによって，DC モータへの入力電圧を調整するシステムを提案した．この提案システムは，次のことが期待できる．

(1) 使用者一人一人に合わせた適切な制御ルールが実現されること，

(2) 未経験の路面を走行する場合や使用者の重心位置，車輪の空気圧が変化した場合など，使用者を含む電動車いすの特性が変化した場合においても，適切な制御ルールが維持されること，

(3) 使用者を含む電動車いすに関して，センシングする情報が部分的であっても適切な制御ルールが構築されること．

そして，電動車いすの基本ダイナミクスに基づいたシミュレーションによって，(1) 傾斜角が一定な路面を走行する場合，(2) 傾斜角が一定でない路面を走行する場合，さらに補足として，(3) 重心位置が変化する場合，(4) 2 つの路面を走行する場合，について，提案システムの有効性ならびに，実際の電動車いすへの適用可能性を確認した．また特に，提案システムを用いることによって使用者の操作負担を軽減できることも合わせて確認した．

今後の課題として，1) 提案システムの連続行動空間への拡張，2) 実際の電動車いすへ適用を通して，システムの有効性を検証すること，などが挙げられる．

第 7 章

結論

本論文では，強化学習の実用化に向け，強化学習における状態空間の設計問題と不完全知覚問題の解決を目的として，状態フィルタの枠組みおよび実現手法を提案した．さらに，実際的な問題への適用例として，電動車いすの直進走行システムを取り上げ，電動車いす使用者の操作負担軽減に対する有効性について検討した．以下，本論文で得られた結論を整理する．

第 2 章では，強化学習方式の概要の説明を行い，現在強化学習方式が抱える多くの問題の中から盛んに研究がなされているものを取り上げ，その問題に対してそれぞれの代表的アプローチを示した．そして，本研究において状態空間の設計問題および不完全知覚問題に取り組むことを述べた．

第 3 章では，エージェントの観測空間と行動学習器の間に状態フィルタを導入した計算モデルを提案し，理想的状態フィルタについても記述した．つぎに，MDPs の場合と POMDPs の場合について，状態フィルタを獲得する方法の違いという観点から従来手法の分類を行い，従来手法の構成面での特徴を明確にした．そして，内部状態を分割・統合する手法の状態フィルタの基本的機能として，内部状態の分割・統合方法について通常分割・統合・履歴参照分割・削除の 4 つの調整法を提案した．

第 4 章では，状態空間の設計問題に焦点をあて，タスクを遂行するために十分な情報が与えられている MDPs の問題を対象として，エントロピーを用いた状態フィルタの一実現法を提案した．この際，ある内部状態における行動選択確率のエントロピーを，その内部状態についての状態集約の正しさを評価する指標として用いた．また，この手法は，状態フィルタの学習と行動学習を同時並行して行わせることができ，学習器において適用できる強化学習手法が限定されないという特徴をもっている．さらに，強化学習問題の例題として離散状態空間を有する迷路問題および連続状態空間を有するロボットナビゲーション問題を取り上げ，計算機実験を通して提案手法を従来手法・グリッド分割法と比べることで，提案手法の有効性・可能性を，それぞれ確認した．

第 5 章では，POMDPs への対応・状態空間のコンパクト化に焦点を当て，タスクを遂行するためには十分でない部分的な情報が与えられている POMDPs の問題を対象として，第 4 章で提案した手法を POMDPs に拡張する形で，適応的に履歴情報の記録・参照を行い，

繰り返し内部状態を分割・統合することで POMDPs へ接近する状態フィルタの一実現手法を提案した．この際，ある内部状態における行動選択確率のエントロピーを，その内部状態の細かさの十分性を評価する指標として用いた．また，この手法は，第 4 章において提案した手法と同様，状態フィルタの学習と行動学習を同時並行して行わせることができ，学習器において適用できる強化学習手法が特定されないという特徴をもっている．さらに，強化学習問題の例題として離散状態空間を有する迷路問題および連続状態空間を有するロボットナビゲーション問題を取り上げ，計算機実験を通して，提案手法を従来手法と比べることで，POMDPs への対応・状態空間のコンパクト化を実現できることを，そして提案手法の有効性・実問題への適用可能性を，それぞれ確認した．

第 6 章では，第 5 章で提案する手法のより実際的な問題への適用例として，電動車いすの直進走行システムを取り上げた．ここではまず，まちにあるバリアとして傾斜路面に注目し，左右 DC モータから駆動力を得る後輪駆動型電動車いすを対象として，三次元上における電動車いすの基本ダイナミクスのモデル化を行った．つぎに，基本的なダイナミクスを考慮した上，電動車いすの直進走行システムとして，5 章で提案した手法を再構築し，適用することによって，DC モータへの入力電圧を調整するシステムを提案した．この提案システムは，次のようなことが期待できる．

(1) 使用者一人一人に合わせた適切な制御ルールが実現されること，

(2) 未経験の路面を走行する場合や使用者の重心位置，車輪の空気圧が変化した場合など，使用者を含む電動車いすの特性が変化した場合においても，適切な制御ルールが維持されること，

(3) 使用者を含む電動車いすに関して，センシングする情報が部分的であっても適切な制御ルールが構築されること．

そして，電動車いすの基本ダイナミクスに基づいたシミュレーションによって，(1) 傾斜角が一定な路面を走行する場合，(2) 傾斜角が一定でない路面を走行する場合，さらに補足として，(3) 重心位置が変化する場合，(4) 2 つの路面を走行する場合，について，提案システムの有効性ならびに実際の電動車いすへの適用可能性を確認した．また特に，提案システムを用いることによって使用者の操作負担を軽減できることを確認した．

以上より，本論文において，状態空間の設計問題と不完全知覚問題に焦点をあて，これらの問題が解消される有効な状態フィルタの実現手法を提案した．さらに，より実際的な問題への適用例として，電動車いすの直進走行システムを取り上げ，提案システムの有効性ならびに実際的な適用可能性を確認した．

しかしながら，本論文で提案した手法の有効性は，全て計算機実験を通して経験的に確認されたものであり，その理論的な解析は全く行われていない．さらには，獲得された制御ルールの理論的な解析も必要である．これらに関する検討が今後の課題である．一方，連続行動空間への拡張，冗長かつ部分的な入力情報の場合に対する有効な状態フィルタの一実現

手法についても今後，研究を進めていきたい．

謝 辞

本研究を進めるにあたり，終始適切なご指導およびご助言を賜わりました神戸大学工学部 玉置久教授に厚く御礼を申し上げます．また，本論文をまとめるにあたって懇切なる御指導および御討論を戴きました神戸大学大学院自然科学研究科 上原邦昭教授，ならびに神戸大学大学院自然科学研究科 小島史男教授に謹んで感謝の意を表します．

研究面で多大な御指導およびご鞭撻を賜りました神戸大学理事 北村新三名誉教授に心より感謝申し上げます．そして，本研究を進めるきっかけを与えて下さると共に，研究全般にわたり，ご指導およびご討論を戴きました神戸大学国際文化学部 村尾元助教授に深く感謝の意を表します．

研究を遂行するに際して，多くのご助言を戴きました摂南大学工学部 諏訪晴彦助教授に深く感謝致します．

研究を温かく見守り気遣って下さいました兵庫県立福祉のまちづくり工学研究所の皆様に心より深く感謝致します．特に, 米田郁夫主任研究員兼第三課課長 (現　東洋大学ライフデザイン学部教授) には，丁寧なご指導とご助言を賜わりました．また，原良昭第四課研究員，松原裕幸第四課義肢装具士には，熱心なご助言とご討論を戴きました．

研究のみならず，様々な方面でお世話になりました玉置研究室の学生諸氏，先輩諸氏には心より感謝致します．特に，大森清博 博士，榊原一紀 博士，稲元勉氏，松本卓也氏，杉川智氏，伊藤義人氏には研究環境の整備など，研究途上において公私にわたり助けて戴きました．

最後になりましたが，長い学生生活をあらゆる面でサポートしていただきました両親をはじめ姉 理恵，美穂，妹 佳代，そして生活面ならびに心理面からしっかりとサポートしてくれた妻 裕絵，息子 晟人に深く感謝致します．

参考文献

1) Sutton, R. S. and Barto, A. G.,“ Reinforcement Learning,” A Bradford Book, MIT Press, 1998. (邦訳 三上, 皆川,“強化学習,” 森北出版, 2000.)

2) 木村 元, 宮崎 和光, 小林 重信,“強化学習システムの設計指針,” 計測と制御, Vol. 38, No. 10, pp. 618-623, 1999.

3) Chrisman, L.,“Reinforcement Learning with Perceptual Aliasing: The Perceptual Distinctions Approach,” Proc. of the 10th International Conference on Artificial Intelligence, pp. 183-188, AAAI Press, 1992.

4) Kaelbling, L. P., Littman, M. L. and Moore, A. W.,“Reinforcement Learning: A Survey,” Jornal of Artificial Intelligence Research, Vol. 4, 1996.

5) Lin, L. J. and Mitchell, T. M.,“Reinforcement Learning with Hidden State,” Proc. of the 2nd International Conference on Simulation of Adaptive Behavior, pp. 271-280, 1992.

6) Whitehead, S. D. and Lin, L. J.,“Reinforcement Learning of Non-Markov Decision Processes,” Artificial Intelligence 73, pp. 271-306, 1995.

7) Bradtke, S. J. and Duff, M. O.,“Reinforcement Learning Methods for Continuous-Time Markov Decision Problems,” In Tesauro, G., Touretzky, D. and Leen, T. (eds.), Advances in Neural Information Processiong Systems: Proc. of the 1994 Conference, pp. 393-400. MIT Press, Cambridge, MA., 1995.

8) Doya, K.,“Temporal Difference Learning in Continuous Time and Space,” In Touretzky, D. S., Mozer, M. C. and Hasselmo, M. E. (eds.),Advances in Neural Information Processing Systems: Proceeding of the 1995 Conference, pp. 1073-1079. MIT Press, 1996.

9) 宮崎 和光, 山村 雅幸, 小林 重信,“MarcoPolo：報酬獲得と環境同定のトレードオフを考慮した強化学習システム,” 人工知能学会誌, Vol. 12, No. 1, pp. 78-89, 1997.

10) 深尾 隆則, 大村 亮祐, 足立 紀彦,“Q-learning における状態空間の適応的分割法,” 計測自動制御学会論文集, Vol. 37, No. 3, pp. 242-249, 2001.

11) 濱上 知樹, 小圷 成一, 平田 廣則,“適応的な状態分割を行なう Q-Learning における状態数の調整方法,” 電子情報通信学会論文誌, Vol. J86-D-I, No. 7, pp. 490-499, 2003.

12) Sutton, R. S.,“Learning to Predict by the Methods of Temporal Difference,” Machine Learning, Vol. 3, pp. 9-44, 1988.

13) Watkins, C. J. C. H.,“Learning from Delayed Rewards,” PhD Thesis, University of Cambridge, 1989.

14) Peng, J. and Williams, R. J.,“Incremental Multi-Step Q-Learning, Machine Learning,” Vol. 22, pp. 283-290, 1996.

15) Sutton, R. S.,“Integrated Architectures for Learning, Planning, and Reacting Based on Approximating Dynamic Programming,” Proc. of the 7th International Conference of Machine Learning, pp. 216-224, 1990.

16) Takahashi, Y., Asada, M. and Hosoda, K.,“Reasonable Performance in Less Learning Time by Real Robot Based on Incremental State Space Segmentation,” Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1518-1524, 1996.

17) 浅田 稔, 野田 彰一, 細田 耕,“ロボットの行動獲得のための状態空間の自律的構成,” 日本ロボット学会誌, Vol. 15, No. 6, pp. 886-892, 1997.

18) Murao, H. and Kitamura, S.,“QLASS: An Enhancement of Q-learning to Generate State Space Adaptively,” Fourth European Conference on Artificial Life(ECAL97), 1997.

19) 増尾 篤史, 木村 元, 小林 重信,“連続値空間の強化学習における動的な状態分割法の提案,” 第 25 回知能システムシンポジウム, pp. 143-148, 1998.

20) 上野 敦志, 中須賀 真一, 堀 浩一,“自律エージェントのための状況認識と行動規則の同時学習,” 人工知能学会誌, Vol. 15, No. 2, pp. 297-308, 2000.

21) 柴 武将, 石川 孝,“強化学習における状態空間の自律的構成方法,” 情報処理学会研究報告 知能と複雑系, 2001-ICS-123, pp. 141-146, 2001.

22) 矢入 健久, 堀 浩一, 中須賀 真一,“複数行動結果を考慮した最尤推定に基づく状態一般化法,” 人工知能学会誌, Vol. 16, No. 1, pp. 128-138, 2001.

23) 井上 康介, 太田 順, 新井 民夫,“部分観測環境下での自律的状態空間構成を伴う実移動ロボットのナビゲーション行動獲得,” 第 20 回日本ロボット学会学術講演会予稿集, 3H31, 2002.

24) Sutton, R. S.,“Generalization in Reinforcement Learning: Successful Examples Using Sparse Coarse Coding,” Advances in Neural Information Processing Systems: Proceeding of the 1995 Conference, pp. 1038-1044, 1996.

25) 森本 淳, 銅谷 賢治,“強化学習を用いた高次元連続状態空間における系列運動学習：起き上がり運動の獲得,” 電子情報通信学会論文誌, Vol. J82-D-II, N0.11, pp. 2118-2131,

1999.

26) 鮫島 和行, 大森 隆司,“強化学習における適応的状態空間構成法,” 日本神経回路学会誌, Vol. 6, No. 3, pp. 144-154, 1999.

27) McCallum, R. A.,“Instance-Based Utile Distinctions for Reinforcement Learning with Hidden State,” Proc. of the 12th International Conference on Machine Learning, pp. 387-395, 1995.

28) Kimura, H., Yamamura, M. and Kobayashi, S.,“ Reinforcement Learning by Stochastic Hill Climbing on Discounted Reward,” Proc. of the 12th International Conference on Machine Learning, pp. 295-303, 1995.

29) 木村 元, 山村 雅幸, 小林 重信,“部分観測マルコフ決定過程下での強化学習：確率的傾斜法による接近,” 人工知能学会誌, Vol. 11, No. 5, pp. 761-768, 1996.

30) Baird, L.,“Residual Algorithms: Reinforcement Learning with Function Approximation,” Proc. of the 12th International Conference on Machine Learning, pp30-37, 1995.

31) 後藤 亮, 松尾 啓志,“強化学習における Support Vector Machine を用いた状態一般化法,” 電子情報通信学会論文誌, J86-D-I, 12, pp. 897-905, 2003.

32) 半田 久志, 二宮 明, 堀内 匡, 小西 忠孝, 馬場 充,“強化学習における矛盾の概念に沿った漸増的な状態空間の構成法,” 計測自動制御学会論文集, 38, 5, pp. 469-476, 2002.

33) 神尾 武司, 曽我 咲十美, 三堀 邦彦,“ファジィART ニューラルネットワークによる強化学習のための状態空間の構成法,” 信学技報, NC2002-213, pp. 43-48, 2004.

34) 半田 久志,“知覚変化予測に基づいた状態分割法を併用した強化学習,” 信学技報, NC2004-81, pp. 75-79, 2004.

35) Ono, N. and Fukumoto, K.,“Multi-agent Reinforcement Learning: A Modular Approach,” Proc. of the 2nd International Conference on Multi-agent Systems(ICMAS-96),AAAI Press, 1996.

36) 藤田 和幸, 松尾 啓志,“状態空間の部分的高次元化法によるマルチエージェント強化学習,” 電子情報通信学会論文誌, J88-D-I, 4, pp. 864-872, 2005.

37) 永吉 雅人, 玉置 久, 村尾 元, 北村 新三,“モジュール型強化学習における適応的状態空間構成法,” 神戸大学大学院自然科学研究科紀要, 23-B, pp. 13-20, 2005.

38) Littman, M. L., Cassandra, A. R. and Kaelbling, L. P.,“Learning Plicies for Patially Observable Environment: Scaling up,” Proc. of Conf. on Machine Learning-1994, pp. 362-370, 1995.

39) 末松 伸朗, 林 朗, 李 仕剛,“部分観測環境での強化学習へのモデルベースアプローチ：可変長記憶モデルのベイズ学習,” 人工知能学会誌, Vol. 13, No. 3, pp. 68-78, 1998.

40) 宮崎 和光, 小林 重信,“Profit Sharing の不完全知覚環境下への拡張： PS-r*の提案と評価” 人工知能学会誌, Vol. 18, No. 5, pp. 286-296, 2003.

41) 斎藤 健, 増田 士朗,“不完全知覚判定法を導入した Profit Sharing,” 人工知能学会誌, Vol. 19, No. 5, pp. 379-388, 2004.

42) Glickman, M. R. and Sycara, K.,“Evolutionary Search, Stochastic Policies with Memory, and Reinforcement Learning with Hidden State,” Proc. of the 18th International Conference on Machine Learning, pp. 194-201, 2001.

43) 瀧田 航一朗, 萩原 将文,“部分観測マルコフ決定過程下の強化学習のためのパルスニューラルネットワーク学習則,” 電子情報通信学会論文誌, Vol. J86-D-II, No. 7, pp. 1067-1077, 2003.

44) Wiering, M. and Schmidhuber, J.,“HQ-Learning,” Adaptive Behavior, Vol. 6, No. 2, pp. 219-246, 1997.

45) Sun, R. and Sessions, C.,“Self-Segmentation of Sequences: Automatic Formation of Hierarchies of Sequential Behaviors,” IEEE Trans., SMC-B-30, pp. 403-418, 2000.

46) 釜谷 博行, 阿部 健一,“部分観測マルコフ環境における階層型強化学習 -スイッチングQ-学習の提案-,” 電気学会論文誌 C, Vol. 122, No. 7, pp. 1186-1193, 2002.

47) Lee, H. and Kamaya, H.,“Labeling Q-Learning in POMDP Environments,” IEICE TRANS.INF.&SYST., Vol. E85-D, No,9, pp. 1425-1432, 2002.

48) Kamaya, H. and Abe, K.,“Hierarchical Q-learning in POMDP Environments,” Proc. of the Eleventh International Symposium on Artificial Life and Robotics 2006 (AROB 11th’06), OS7-4(CD-ROM), 2006.

49) Singh, S. P., Jaakkola, T. and Jordan, M. I.,“Learning Without State-Estimation in Partially Observable Markovian Decision Processes,” Proc.of the 11th International Conference on Machine Learning, pp. 284-292, 1994.

50) Sutton, R. S., McAllester, D., Singh, S. and Mansour, Y.,“Policy Gradient Methods for Reinforcement Learning with Function Approximation,” Advances in Neural Information Processing Systems 12(NIPS12), pp. 1057-1063, 2000.

51) 伊藤 昭, 金渕 満,“知覚情報の粗視化によるマルチエージェント強化学習の高速化- ハンターゲームを例に-,” 電子情報通信学会論文誌, J84-D-I, 3, pp. 285-293, 2001.

52) 永吉 雅人, 村尾 元, 玉置 久, “強化学習における状態フィルタの提案と一実現手法,” 電気学会論文誌 C, Vol. 126, No. 7, pp. 832-839, 2006.

53) Singh, S. P. and Sutton, R. S, “Reinforcement Learning with Replacing Eligibility Traces,” Machine Learning, Vol. 22, pp. 123-158, 1996.

54) 齋藤 隆之, 林 豊彦, 中村 康雄, 遁所 直樹,“利用可能な身体機能に応じた障碍者用操作スイッチの選択支援システム,” 信学技報, TL2003-9, pp. 49-54, 2003.

55) 小宮 加容子, 中島 康博, 橋場 参三, 景川 耕宇, 黒須 顕二,“狭い空間における音声指令による電動車いす走行テスト,” 日本機会学会論文集 (C 編), Vol. 69, No. 688, pp. 210-217, 2003.

56) 濱上 知樹, 平田 廣則,“知的車椅子における環境適応のための状態空間の構成法,” 電気学会論文誌 D(産業応用部門誌), Vol. 123, No. 10, pp. 1118-1124, 2003.

57) 黒住 亮太, 山本 透,“強化学習による電動車椅子の障害物回避補助システムの構築,” システム制御情報学会論文誌, Vol. 19, No. 1, pp. 7-14, 2006.

58) 米田 郁夫, 橋詰 努, 藤記 拓也, 木原 寿紀, 平川 雅子, 鎌田 実,“ 片流れ路面が車いす利用者に強いる負担増の定量的評価,” 第 14 回リハ工学カンファレンス, pp. 81-84, 1999.

59) 水口 文洋, 大鍋 寿一,“片流れ横断歩道の手動車いすによる横断シミュレーション,” 第 16 回リハ工学カンファレンス, pp. 53-56, 2006.

60) 長瀬 浩明, 北野 哲彦, 吉田 健ニ, 浜 淳, 相澤 淳平,“動作特性にもとづく車椅子等の傾斜路面適合化技術に関する研究,” 長野県情報技術試験場研究報告, No. 18, pp. 6-10, 2002.

61) 大垣 斉, 池田 義弘, 竹田 晴見,“電動車いすのモデルについて,” システム制御情報学会論文誌, Vol. 7, No. 6, pp. 207-212, 1994.

62) 永吉 雅人, 村尾 元, 玉置 久, “POMDPs での強化学習における状態フィルタ,” 計測自動制御学会論文集, (投稿中).

本研究に関する発表

発表論文

(1) 永吉雅人, 村尾 元, 玉置 久, “強化学習における状態フィルタの提案と一実現手法,” 電気学会論文誌 C, Vol. 126, No. 7, pp. 832-839, 2006.

(2) 永吉雅人, 村尾 元, 玉置 久, “POMDPs での強化学習における状態フィルタ,” 計測自動制御学会論文集, 投稿中.

(3) 永吉雅人, 村尾 元, 玉置 久, 米田 郁夫, “強化学習を用いた電動車いすによる適応的直進走行,” 計測自動制御学会論文集, 投稿中.

国際会議

(1) Masato Nagayoshi, Hajime Murao and Hisashi Tamaki, “A State Space Filter for Reinforcement Learning,” Proc. of the Eleventh International Symposium on Artificial Life and Robotics 2006 (AROB 11th’06), pp. 615-618 (GS1-3 (CD-ROM)), 2006.

(2) Masato Nagayoshi, Hajime Murao and Hisashi Tamaki, “A State Space Filter for Reinforcement Learning in POMDPs - Application to a Continuous State Space -,” Proc. of the SICE-ICSE International Joint Conference 2006 (SICE-ICCAS 2006), pp. 6037-6042 (SE18-4 (DVD-ROM)), 2006.

紀要

(1) 永吉雅人, 玉置 久, 村尾 元, 北村新三, “モジュール型強化学習における適応的状態空間構成法,” 神戸大学大学院自然科学研究科紀要第 23 号-B, pp. 13-20, 2005.

口頭発表

(1) 永吉雅人, 村尾 元, 玉置 久, 北村新三, “適応的に状態のモジュール間統合を行なうモジュラー型強化学習の提案, 第 46 回システム制御情報学会研究発表講演会,” pp.

433-434, 2002.

(2) 永吉雅人, 村尾 元, 玉置 久, 北村新三, “状態のフィルタリングを考慮した強化学習方式,” 第 48 回システム制御情報学会研究発表講演会, pp. 607-608, 2004.

(3) 永吉雅人, 村尾 元, 玉置 久, 北村新三, “高次元連続状態空間での強化学習における状態フィルタの適応的獲得,” 平成 16 年電気学会電子・情報・システム部門大会, pp. 429-434, 2004.

(4) 永吉雅人, 村尾 元, 玉置 久, 北村新三, “強化学習における状態フィルタの適応的獲得-マルチエージェント問題への適用,” 第 17 回自律分散システム・シンポジウム, pp. 195-200, 2005.

(5) 永吉雅人, 村尾 元, 玉置 久, 北村新三, “強化学習における状態フィルタの評価指標,” 第 49 回システム制御情報学会研究発表講演会, pp. 293-294, 2005.

(6) 永吉雅人, 村尾 元, 玉置 久, “強化学習における状態フィルタ：冗長な入力情報を含む連続状態空間への適用,” 平成 17 年電気学会電子・情報・システム部門大会, pp. 474-479, 2005.

(7) 小山陽平, 永吉雅人, 村尾 元, 玉置 久, “連続状態空間・連続行動空間での強化学習における状態フィルタの適応的獲得,” 第 50 回システム制御情報学会研究発表講演会, pp. 355-356, 2006.

(8) 米田郁夫, 北川博巳, 橋詰 努, 室崎千重, 永吉雅人, 神吉優美, 糟谷佐紀, 谷内久美子, “プラットホーム-列車間の段差・隙間が手動車いすに与える影響と対策に関する研究,” 第 21 回リハ工学カンファレンス, pp. 13-14, 2006.

(9) 米田郁夫, 橋詰 努, 室崎千重, 神吉優美, 永吉雅人, 北川博巳, 糟谷佐紀, “環境バリアにおける負担軽減のための車いす側の工夫の可能性－横断勾配路面への対応方法について－,” 日本福祉のまちづくり学会第 9 回全国大会，pp. 123-126, 2006.

(10) 永吉雅人, 村尾 元, 玉置 久, “POMDPs での強化学習における状態フィルタ：離散状態空間と連続状態空間への適用,” 平成 18 年電気学会電子・情報・システム部門大会, pp. 487-492, 2006.

ポスター発表

(1) 永吉雅人, 村尾 元, 玉置 久, 北村新三, “強化学習における状態フィルタの適応的獲得,” 第 10 回創発システム・シンポジウム, pp. 129-132, 2004.

(2) 永吉雅人, 村尾 元, 玉置 久, 北村新三, “強化学習における状態フィルタの適応過程,” 第 11 回創発システム・シンポジウム, pp. 123-126, 2005.

(3) 永吉雅人, 村尾 元, 玉置 久, “強化学習における POMDPs での状態フィルタの適応的

獲得,” 第 12 回創発システム・シンポジウム, pp. 68-71, 2006.