平成 13 年度　博士論文

部分観測環境における
エージェントの自律的行動獲得

Autonomous Behavior Acquisition by an Agent in Partially Observable Environments

指導教官　太田 順　助教授

東京大学 大学院 工学系研究科 精密機械工学専攻

学生証番号　　87070

井上 康介

論文概要

本論文では，身体性を持つエージェントが部分観測環境において連続的観測空間の自律的分節化を伴う状態空間の自律的構成を行いながら，即時報酬に基づいて状況認識・行動決定機構を獲得し，これにより複数タスクを実現する手法の提案を行う．

自らの身体を通じて環境と相互作用するエージェントにとって，エージェントの視点に即した状況認識の方法は，エージェントの身体・環境・タスク間の相互作用の様態に立脚したものとなるため，これを事前設計により付与することは困難であり，エージェントにより自律的に獲得される必要がある．

ところが，エージェントの身体性からの直接的な帰結として，エージェントの世界観測の能力は限定されており，エージェントは俯瞰的な視点に基づく大域的な状況認識を行うことが不可能である．即ち，エージェントは自らの観測能力の範囲内における局所的な観測入力に基づいて状況認識を行う必要があり，このために観測入力の部分性に起因する部分観測 Markov 決定過程を扱わなければならない．

更に，エージェントの観測入力は一般に連続的空間であり，特定の観測入力をタスク遂行に対する特定の意義を持つ状況に関連づけるための観測の解釈方法が事前に与えられないことから，エージェントは観測入力の解釈方法を自ら規定しなければならない．

以上 2 点の問題が存在する場合，エージェントの状況認識方法は身体・環境・タスク間の相互作用に依存するものとなる．この結果として，エージェントが複数のタスクを扱わねばならない場合，エージェントは現在扱われているタスクの認識方法と現在置かれている状況の認識方法を，ともに身体・環境・タスクに立脚して獲得しなければならない．

本研究では，エージェントが動作する各時刻において外界から適切な評価信号としての即時報酬を獲得することが可能であるという前提のもとで，上記の問題を解決しうる状態認識機構・行動決定機構の獲得を実現する手法を提案する．各時点において即時報酬を獲得するという点において，提案する手法は一種の教示システムと見ることが可能であるが，提案手法においてはエージェントの状況認識を予め設計せず，評価値という少ない教示情報に基づいて状況認識・タスク実現行動を実現するという点で，設計コスト・教示コストの小さい教示システムと言える．

個別の単一のタスクに対しては，提案手法ではエージェントが実際に行った観測・動作の短期記憶を状況認識機構の入力として用いることで環境の部分観測性に対処する．具体的には，エージェントの内部状態表現として，短期記憶に基づく状況の識別を表現する決定木構造の状態表現を採用する．この決定木内において，観測情報に基づく識別を示す分岐を適切に追加することで，観測入力の自律的解釈を実現する．初期の状態表現は単一の状態からなり，エージェントはこれに対して，即時報酬に基づく状態表現の適切性の判断に基づいて適切に分岐を加える．特定の時点において状態表現が不適切であると判断された場合，エージェントは状態表現の決定木に新たな分岐を加えるが，ここで状態表現の不適切性には，観測入力のより詳細な識別が必要である場合と，より過去の経験に基づく識別が必要である場合があり，提案手法ではこれを過去の経験データ基づいて統計的に判断する．

また，複数タスクを実現するために，個別のタスクに対して上記手法により行動獲得を行った上で，現在エージェントが行っているのがそれらのうちのいずれかを実経験データに基づいて判別し，適切な行動を実行する機構を提案する．ここで，タスク識別のために必要な経験のデータが，個別のタスクの学習の過程で得られていないという問題，および環境の非 Markov 性に起因して，タスク識別のための行動が個別タスクの実現に対して悪影響をもたらす可能性があるという問題があり，これに対しては個別のタスクに関して適切な追加学習を行う学習のスケジューリング手法によって対処する．

これらの手法を実ロボットによるナビゲーション問題に適用し，スタート地点が複数存在するナビゲーションタスクにおいて，特定の試行におけるスタート地点を実際にロボットが得た観測・行った動作の経験データに基づいて適切に判別し，スタート点に対応する状況認識・行動決定機構を利用することでゴール地点へ到達するシミュレーション・実験を行い，提案手法が実ロボットによる実環境でのタスク実行に対して有効であることを示す．

提案手法における，計算量・記憶量の消費や即時報酬の付与方法，誤差により受ける影響などについて，考察・評価を行う．

目 次

1 序 論 **1**

1.1 本研究の背景 · · · · · 2

1.2 従来研究 · · · · · 9

1.2.1 Markov性の回復 · · · · · 9

1.2.2 知覚入力の自律的解釈 · · · · · 10

1.2.3 複数タスクの学習 · · · · · 11

1.2.4 教 示 · · · · · 13

1.3 研究の目的 · · · · · 16

1.4 論文の構成 · · · · · 17

2 問題の構造化 **19**

2.1 はじめに · · · · · 20

2.2 問題設定 · · · · · 21

2.3 離散的時間の想定 · · · · · 24

2.4 提案手法の教示システムとしての枠組みと意義 · · · · · 26

2.5 おわりに · · · · · 28

3 単一タスクに対する学習手法 **29**

3.1 はじめに · · · · · 30

3.2 概 要 · · · · · 31

3.3 Utile Suffix Memory · · · · · 33

3.4 状況認識機構の構成 · · · · · 37

3.4.1 状態表現 · · · · · 37

3.4.2 学習過程全体の流れ · · · · · 41

3.4.3 状態分割 · · · · · 42

3.5 シミュレーション · · · · · 49

3.5.1 シミュレーション条件 · · · · · 49
3.5.2 比較対象とする学習手法 · · · · · 50
3.5.3 結 果 · · · · · 51
3.5.4 考 察 · · · · · 52
3.6 おわりに · · · · · 57

4 複数タスクへの応用 **59**
4.1 はじめに · · · · · 60
4.2 手法の概要 · · · · · 61
4.3 タスクの推定 · · · · · 66
4.4 追加学習 · · · · · 67
4.5 タスクの追加 · · · · · 68
4.6 計算機シミュレーション · · · · · 69
4.6.1 シミュレーション設定 · · · · · 69
4.6.2 追加訓練の必要性の確認 · · · · · 70
4.6.3 提案手法の検証 · · · · · 73
4.7 おわりに · · · · · 77

5 実環境への適用 **79**
5.1 はじめに · · · · · 81
5.2 シミュレーション及び実験の目的 · · · · · 82
5.3 シミュレーションおよび実験の概要 · · · · · 84
5.3.1 実機ロボット Khepera · · · · · 84
5.3.2 行動規範型動作プリミティブ · · · · · 87
5.3.3 誤差への対応 · · · · · 91
5.4 実環境における複数タスク実現に関するシミュレーション · · · · · 92
5.4.1 シミュレーション・実験の環境 · · · · · 92
5.4.2 シミュレーションの設定 · · · · · 92
5.4.3 シミュレーションの結果 · · · · · 93
5.4.4 考 察 · · · · · 93
5.5 実機複数タスク実現に関する実験 · · · · · 99
5.5.1 実験結果 · · · · · 99
5.5.2 考 察 · · · · · 100

5.6 異なる視点に基づく報酬付与への対応に関するシミュレーション ····· 105
5.6.1 報酬付与方法 ····· 105
5.6.2 離脱動作の導入 ····· 107
5.6.3 作業環境 ····· 110
5.6.4 パラメータ設定 ····· 111
5.6.5 シミュレーション結果 ····· 112
5.6.6 考察 ····· 113
5.7 おわりに ····· 117

6 考察と評価 119
6.1 はじめに ····· 120
6.2 単一タスクに対する学習に関して ····· 121
6.2.1 計算量および記憶量 ····· 121
6.2.2 環境の性質に対する学習性能の依存性 ····· 124
6.2.3 学習パラメータ設定 ····· 126
6.2.4 観測ベース分割の意義 ····· 129
6.2.5 即時報酬の満たすべき条件 ····· 130
6.3 実環境への適用に関して ····· 135
6.3.1 誤差の影響 ····· 135
6.4 提案手法の適用可能範囲とその拡張への展望 ····· 142
6.4.1 適用可能範囲について ····· 142
6.4.2 適用範囲の拡張への展望 ····· 144
6.5 おわりに ····· 146

7 結論及び今後の展望 147
7.1 結 論 ····· 148
7.2 展 望 ····· 152

第 1 章

序　論

1.1　本研究の背景 · · · · · 2
1.2　従来研究 · · · · · 9
1.2.1　Markov性の回復 · · · · · 9
1.2.2　知覚入力の自律的解釈 · · · · · 10
1.2.3　複数タスクの学習 · · · · · 11
1.2.4　教　示 · · · · · 13
1.3　研究の目的 · · · · · 16
1.4　論文の構成 · · · · · 17

1.1 本研究の背景

近年，ロボットをはじめとする身体性 (embodiment) を有するエージェントにおいて，より知的で自律的な行動を実現しようとする動機付けに基づいて盛んに研究が行われている [27]．身体性とは，エージェントと環境との相互作用に対して，エージェント固有の身体（センサ・アクチュエータ）が与える拘束条件のことをいい，エージェントが身体性を持つとき，エージェントの認識・行動はエージェントの身体と環境との相互作用を通して行われるため，その結果はその身体性を通じたエージェントと環境との相互作用の様態に依存するものとなる (Fig.1.1)．従って，このようなエージェントの知能を設計しようとする際には，エージェントの認識および行動の様態について予め正確なモデルを設計することは困難であり，特定の状況における認識・行動の結果は，エージェントが実際にそれを行う時点で初めて明らかとなる．

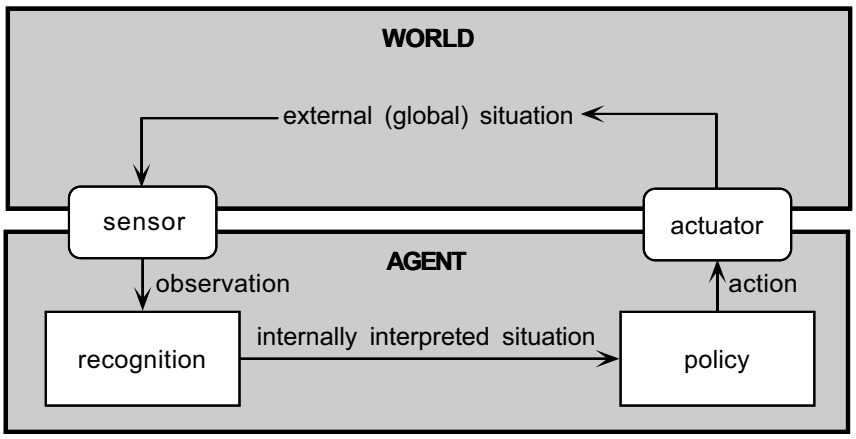


Fig. 1.1: Interaction betwee agent and world

上記の理由から，身体性を持つエージェントにおける知的な行動の実現においては，エージェントの認識様式・行動様式は，タスク実現における環境との相互作用に基づいて，エージェント自身により獲得される必要がある [14]．言い換えれば，身体性を有するエージェントに対しては，その状況認識及び行動を律するメカニズムを設計段階で適切に付与することは困難であり，これらを環境との相互作用を通じてエージェント自らが獲得するという形でエージェントの自律性に委任するという方法が必要であると言える．

強化学習 [17] やニューラルネットワークなどの学習手法を用いたエージェントの行動獲得を扱った研究は従来多く行われている．しかし，これらの方法に基づく従来法のほとんどにおいては，特定の状況における特定の行動がタスクの実現に対して持つ意義に基づいて，個別の状況を識別する能力がエージェントにとって所与であることが前提されている．前述の通り，これをアプリオリなものとして前提するのは不適切である．なぜなら設計者

の想定した状況識別の様式が実際の身体・環境・タスクに対して妥当であるためには，多くの設計努力と設計者の想定範囲内への問題の限定が必要となるからである．

以上の議論から，エージェントは現在置かれた状況の認識方法を，与えられた身体性・環境・タスクに立脚して自ら獲得しなければならない．

ここで，本論文を通じて,「状況」とは，環境内においてエージェントが置かれている個別の状態を意味するものとする．エージェントのおかれた状況は，エージェント及び環境の大域的な様態を外的に俯瞰することのできる外部観測者の視点に基づく場合と，エージェント自身の視点からエージェント自身の認識機構に基づいて規定される場合とでは異なり，以下，前者を「外的状態 (external state)」，後者を「内的状態 (internal state)」と表記する．Fig.1.2には，例として移動ロボットが環境中で置かれる状況を示す．円形がロボットであるとし，ロボットに搭載された 2 つのセンサの読み取り値が付記した数字であるとするとき，図に示すロボットの 3 つの位置・姿勢 (A, B, C) という状況の違いは外部観測者には 3 つの外的状態として識別される．これに対して，ロボット自身の視点では必ずしもこれらの状況の差異が認識可能であるわけではない．例えば，ロボットが現在得ているセンサ値のみに基づいて状況の識別を行っているとした場合には，同一のセンサ値を与える 2 つの外的状況 (A, B) は同一の内的状況として識別される．

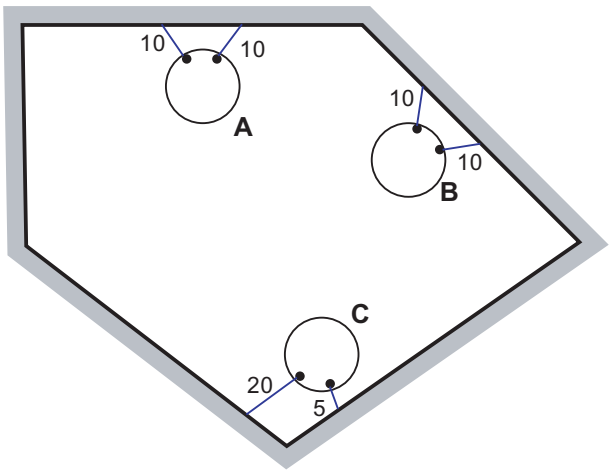


Fig. 1.2: External states

また，以下「状況を認識する」とは，エージェントが環境との相互作用に基づいて現在置かれた状況を特定の内的状態に対応づけることを言うものとする．

本研究では，上に示す身体性の存在がエージェントに与える属性のうち，特にエージェントのセンシング能力が与える限定による観測入力の局所性の問題と，特定の状況に対応するセンサ読み取り値を事前に正しく与えることが不可能であるという問題の 2 点に着目

し，この想定のもとでエージェントが多様なタスクを実行する能力を獲得する上で解決されなければならない問題として，以下の 3 点に着目する：

問題 1： 観測の非 Markov 性
問題 2： 観測値の解釈方法の身体・環境・タスクへの立脚性
問題 3： 複数のタスクに対する対応の困難性

以下，それぞれの問題について説明する．

問題 1：観測の非 Markov 性

まず第 1 点として，身体性を有するエージェントの状況認識は，自らのセンサから直接得られた観測値に基づいて行われなければならないが，一般にロボットをはじめとする身体性を有するエージェントにとって，そのセンサが与える観測値は，エージェントの置かれた状況を一意に特定するために十分な情報を備えていない．

例えば移動ロボットを例に挙げて考えると，一般的な移動ロボットの持つセンサ系，およびセンサ情報処理系の性能は限定されたものであることから，観測値は誤差を含み，観測情報の及ぶ範囲に一定の範囲があることがあり，またオクルージョンなどにより環境の特徴が隠される問題も考えられる．更には，ロボットがビジョンなどの高解像度のセンサを有していたとしても，その画像を処理して解釈する段階においては，現状では極めて限定された能力しか発揮できない．

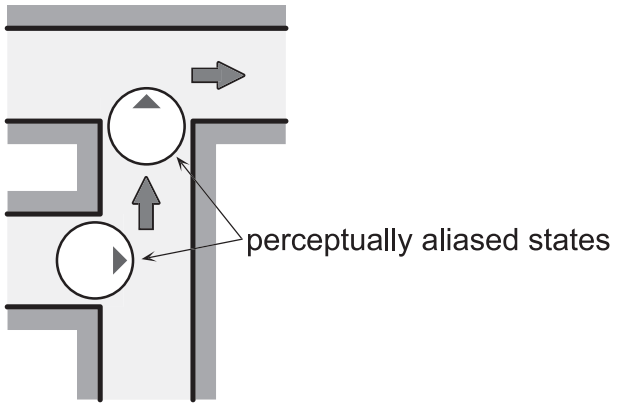


Fig. 1.3: Perceptually aliased states

Fig.1.3は，移動ロボットのセンサ系の限定により起こる具体的な問題を示している．図中，円形を移動ロボットとし，その方向を三角形が示している．ロボットは通路上の T 字路を，最初は左，次は右へ曲がることで目的地を目指すことができるものとする．ところが，センサのレンジが短い，あるいはセンサの分解能が低い場合，即時的な観測の結果だけからは，これら 2 つの T 字路を識別することはできない．

このようにして，身体性を持つエージェントにとっての世界観測の結果求められる現在の状況の特徴は部分性を持っており，この結果，環境内にはエージェントにとって同様の観測結果を与える複数の状況や，異なる観測結果を与える単一の状況が存在しうることになり，エージェントの観測結果はMarkov性を失う．このように異なる状況が同様の観測を与えることにより，観測に基づく状況の識別が不可能となる問題は**知覚騙し問題 (perceptual aliasing problem)** と呼ばれ，このような問題を含む環境においては，即時的な観測入力のみに基づいて十分な行動決定を行うことは不可能となる．

一般的なロボット工学においては，これらの問題に対しては，予めモータ駆動量と状態変位量との関係（オドメトリ）を与える，あるいは環境内に人工ランドマークを設置するなどの方法で設計者による状況認識の補助が行われている．しかし，このような方法では，設計者が十分想定可能な身体・環境・タスクへ適用範囲が限定され，この想定を成立させるための環境整備やセンサ・アクチュエータのモデル化・キャリブレーションといった設計努力が必要となる．

このようにエージェント外部の世界における状態空間において実際にはMarkov 性が成り立つが，エージェントの観測の不完全性に起因してエージェントの観測においてMarkov性が失われるという状況における行動決定問題は，**部分観測 Markov 決定過程 (POMDP : partially observable Markov decision process)** [3] としてモデル化することができ，近年ではPOMDP 上での行動獲得を扱った学習手法が提案されている [26].

以上の議論から，エージェントの有する身体性からの帰結として重要な属性である観測の不完全性を伴う状況認識問題においては，このPOMDP に対する対応が必要となる．

POMDP 上では，エージェントが現在の状態を特定するためには現在の状態における観測値だけでは情報が不十分であり，状況を特定するためには動作を行うことで，得られる観測情報を蓄積し，これによって状況識別の根拠として利用可能な情報を増加させ，このようにして集められた情報を用いて状況を識別する状況認識機構が必要となる．

問題2：観測値の解釈方法の身体・環境・タスクへの立脚性

身体性を有するエージェントの状況認識において解決されなければならない第2点の問題として，エージェントがセンサから得た観測データを特定の内的状態に対応づける変換方法が，それ自体としてエージェントの身体性・環境・タスクに依存したものとなるという問題がある．

例えば，赤外線距離センサにより前方の障害物との間隔を知ることのできる移動ロボットが障害物へ一定距離まで近づくことを考えた場合，センサの機差 (ロボットの身体の特性による条件)，障害物の表面の反射特性や周辺光 (環境条件) および要求されるロボット・障害物間の間隔 (タスクによる条件) に応じて，停止条件となるセンサ読み取り値の境界を変更せねばならない (Fig.1.4)．更に，タスクに応じて，見分けるべき特徴と無視して良

い特徴があり，これらはタスク上の意義 (例えばタスク達成への関連) に依存して決められる必要がある．

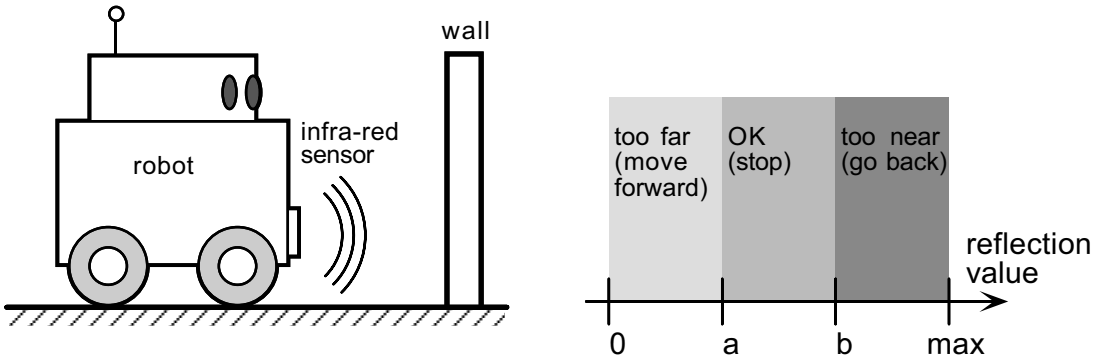


Fig. 1.4: Categorization of significance of observated value

従って，身体性を有するエージェントの状況認識における，観測値から，タスク上の意義に基づいて定められる特定の状況への写像関係は，タスクを実行する上で初めて明らかになるものであり，事前に設計者によって正しく与えることは困難である．

この議論から，特定の観測値からタスク上での意味合いに基づく特定の状況への写像関係はアプリオリではなく，タスク実行における環境との相互作用に基づいてエージェント自らによってこれを定めなければならない．

以上示した問題1および2からの帰結として，以下のことが言える：すなわち，身体性を有するエージェントの状況認識問題においては，外部からエージェントの状況を眺める人間の視点からの状況の規定方法に基づいた方法論では，エージェントの身体性に即した実際のエージェントと環境との相互作用に立脚した適切な状況認識機構を構築することが困難であり，エージェント自身の視点に即してこれを獲得する方法が適切である．

問題3：複数のタスクに対する対応の困難性

従来，多くの研究が単一のタスク実現行動を獲得する手法を提案してきた．例えば強化学習により単一のスタート状態から報酬を得ることのできるゴール状態への単一の動作系列を獲得する方法は，厳密な数学的枠組みに基づいて多く提案されている．

しかしながら，エージェントがより複雑な作業を実現しようとするとき，一般的にはエージェントのタスクの全体は状況に応じて遂行されるべき複数のタスクからなり，エージェントは状況に応じて現在遂行すべきタスクを適切に判断し，それに応じた振る舞いを実現する能力を持たなければならない．

身体性を有し，状況の認識能力に限定をもつエージェントにとっては，これは極めて困

難なものとなる．なぜなら，先に述べたとおりエージェントの状況認識の様式は身体性に依存したエージェント・環境間の相互作用に立脚して規定される必要があるため，現在行うべきタスクの認識と現在行っているタスク上での状況の認識とを同時に実現するための認識機構を実現することが極めて困難となるからである．

従って，身体性を持つエージェントによる認識機構を，全てのタスクの識別および個別のタスク内部における状況の識別を可能とする程度に構造化されたものとするか，あるいはタスクに依存した認識機構をタスクに応じて適切に利用するためのメタ・レベルの認識機構を加える必要がある．

以上挙げた3点の問題は，いずれも身体性を有するエージェントの行動設計において不可避の問題であり，これらは同時に解決されなければならない．

従って，これらの条件をふまえた上でのエージェントの望ましいあり方は，環境の部分観測性に適切に対処しながら自らの状況認識機構・行動決定機構を環境との相互作用に基づいて構築し，個別のタスクに対して獲得されたこれらの機構を，現在行っているタスクに対応して適切に利用する自律エージェントとなる．

しかしながら，上記の問題1，2が未解決である場合，環境上に存在するそれぞれの状況を識別するための認識様式をいかに規定するかについての判断材料が必要となるため，例えば一般的な強化学習問題で扱われている遅延を伴う報酬に基づく行動獲得などの問題クラスにおいては，問題の解決が極めて困難となる．即ち，認識の様式をいかに規定するかを決定する際によりどころとなるべき情報としての環境からの評価情報が極めて乏しい場合，エージェントは状況識別の判断基準を失う．

そこで，この問題を解決する第一歩としては，エージェントが動作を行う都度，その動作のタスク実行に対する寄与度に基づく妥当な即時報酬を，環境からの評価として得ることができるという設定の下でこの問題の解決を図るという方向が考えられる．つまり，エージェントが与えられる情報の中で，報酬だけは信頼できるものとすることで，これに基づいて環境の部分観測性や観測空間の自律的な構築を行うシステムを目指すという問題である．

自律学習エージェントの構築においては，エージェントが外部から与えられる弱い評価信号に基づいて，それを自らの内的構造の各部分に適切に反映することで問題を解決しようとするものと考えるならば，上述のアプローチで扱う問題設定は，この内的な評価の解釈の部分をエージェント内部で行わず，正確な自己評価が可能であると前提した場合に，それに基づいてどのように認識様式と行動様式を構築するかの問題を扱っていることになる．従って，このアプローチは，行動獲得の観点から見て，学習ではなく教示と呼ぶべき問題を扱うものと言える．

このように外部から正確な評価が常に得られるという問題設定は，問題領域を外部から

俯瞰する観察者の視点から見れば困難なものではないといえる．なぜなら，Markov 性を満たし，タスクに対して適切に意義づけられた状態量を直接観測することが可能な視点から見たとき，与えられる即時報酬と現在の状況との対応付けが所与であり，行動獲得に必要なのは，このように適切に認識されたそれぞれの状況に置ける各動作に対して，即時報酬を割り付けることのみとなるからである．エージェント自身の視点に即してこれを考えるとき，エージェントから眺められた世界の様態と，外部から与えられる評価信号との連関を自ら判断せねばならない．即ち，外部観察者の視点から見た状況の評価様式を，限られた状況認識能力しか有しないエージェントの視点の中に包含する必要があり，これはいわば外部観測者の視点からエージェントの視点への**視点の移動**の問題といえ，これは困難な問題であるといえる．

1.2 従来研究

本節では，序論において扱った以下の4つの問題について扱った従来の研究状況について説明する：

- Markov性の回復
- 知覚入力の自律的解釈
- 複数タスクの学習
- 教示

1.2.1 Markov性の回復

身体性を持つエージェントにとって，エージェントに与えられる知覚入力が，直接大域的状態量を特定するのに十分な情報をもっているという想定は，多くの場合非現実的である．現状のロボットの持つ限定されたセンシング能力のもとでは，知覚は局所的であり，誤差を含み，オクルージョンなどの問題を生じるため，知覚のマルコフ性が失われるからである．これに対して，従来の方法論では，この想定を成り立たせるために人為的な補助が導入されてきた．例えば，移動ロボットによるナビゲーションに対しては，ロボットにはあらかじめ，モータ駆動量と状態量の変位との関係（オドメトリ）が与えられていたり，人工ランドマークなどによって外的に状態量を特定する手がかりが与えられていたりする．これらの補助手法の導入には高い設計・実装コストがかかる上に，このような補助による方法では，その補助を導入しうる環境・あるいは導入された環境に，エージェントの適用範囲が限定されてしまう．状況認識に対して与えられるこれらの補助によらず，エージェント自身の視点から状況認識の非マルコフ性を排除するためには，非マルコフ的な知覚入力を自ら解釈して，十分なマルコフ性を備えた状況認識を導く認識機構を環境との相互作用に基づいて自律的に構築せねばならない．

これまで，環境を部分観測Markov決定過程（POMDP）としてモデル化し，そのモデルの上で行動学習を行うという方法が研究されてきた[26]．POMDPモデルでは，真の世界に対してマルコフ性を想定する一方で，エージェントはその状態を直接特定することはできず，世界から得られる観測値を手がかりにして状態を推定せねばならない．

POMDPに関する従来研究を概観すると，大きく2つの方法がある：(1) メモリレスな方法と，(2) メモリに基づく方法である．

このうち，メモリレスな方法[15, 7]では，不完全な知覚情報をそのまま状態量として用いるが，このような方法が有効に動作する問題は非常に限られており，得られる解の品質

も不十分である．なぜなら，エージェントが即時的な観測データだけに基づいて状況を識別できない場合に，識別に用いることのできる追加的なデータは短期記憶だけであり，これを利用しない場合，非マルコフ的な状態を避けるなどの近似的な方法しか利用できないからである．

メモリに基づく方法は，信念状態（belief state）を用いる方法と，短期記憶を明示的に扱う方法に大きく分けられる．

信念状態とは，短期記憶に含まれる情報を，真の状態空間上の確率分布という形で表現したものであり，短期記憶の十分統計量となっている．一般的には，離散的な状態空間を扱うが [5]，Thrun らの方法では連続的状態空間を扱っている [19]．

これらの方法において問題となるのは，信念状態の計算のために，真の状態の空間的な構造が与えられている必要があり，さらに観測確率分布や状態遷移確率といった POMDP 上での先験的モデルを利用しなければならないという点である．

これに対して，得られた観測値および行った動作の短期記憶を直接的に扱う方法群がある [10, 16]．これらの方法では，POMDP モデルが本来もっている汎化能力を状態分割の詳細度という形で明示的に表現することができ，タスクにとって必要十分な詳細度をもった状態表現を作ることができる．

[10] では，過去から現在にかけて得られた観測・行った動作の生経験データに基づく決定木構造の状態表現を自律的に構成することで環境の部分観測性に対応する状態構成を伴う強化学習アルゴリズムである Utile Suffix Memory (USM) が提案されている．ここでは統計学的手法を用いて状態分割の実施を決定するというアルゴリズムにより，状態数が必要最小限の効率のよい状態構成が行われている．また，McCallum は USM を更に発展させ，離散的な多次元の観測入力に対して注視機構を導入した U-Tree を提案している [11]．

しかしながらこれらの方法では，存在しうる観測入力が離散的な集合として与えられているものとして，身体性を通して世界から得られる生の観測値をこのような離散的なシンボルのいずれに対応づけるかの問題はシステムの外部に置かれている．この点で，これらの手法は身体性を持つエージェントによる認識機構としての不十分性を持っていると言える．

1.2.2 知覚入力の自律的解釈

連続的で多次元の知覚入力を，あらかじめ分かっている各知覚入力のタスクに対する「意味」に基づいて，設計者があらかじめカテゴリーに切り分けるという方法は，知覚の解釈方法を人間の手にゆだねてしまうという意味で，エージェントの自律性・適応性を大

きく損なうものである．連続的な状態空間をエージェントが自律的に切り分ける手法は多く提案されている [12, 13, 31, 22].

高橋らは，視覚センサを持つ実ロボットにおけるサッカータスクにおいて，逐次的にセンサ空間を分割することによって状態空間を構成する手法を提案した [31]．ここでは，Fig.1.5(a) の実験環境において，カメラから見えるボールの直径とゴールの大きさに基づくセンサ空間を図 (b) のように分節化し，それぞれの状態において異なる行動を対応づけることにより，ボールをゴールにシュートする行動を実現している．

また，村尾らは，それぞれの領域における適応度ランドスケープが傾斜した平面となるように状態空間を分節化するアルゴリズムである QLASS を提案した．このようにして作られた離散的状態空間に基づいて，[12] では 2 次元ないし 6 次元のセンサ空間を持つ移動ロボットの行動学習を実現している．1.6は QLASS によって分節化された状態空間をあらわしている．

しかしながら，これらの研究においては，即時的なセンサ入力だけからエージェントが実行すべき動作が決定可能であるという想定，すなわち観測の Markov 性が前提されている．前に述べたとおり，この想定は身体性を持つエージェントにとって一般には成立しない．

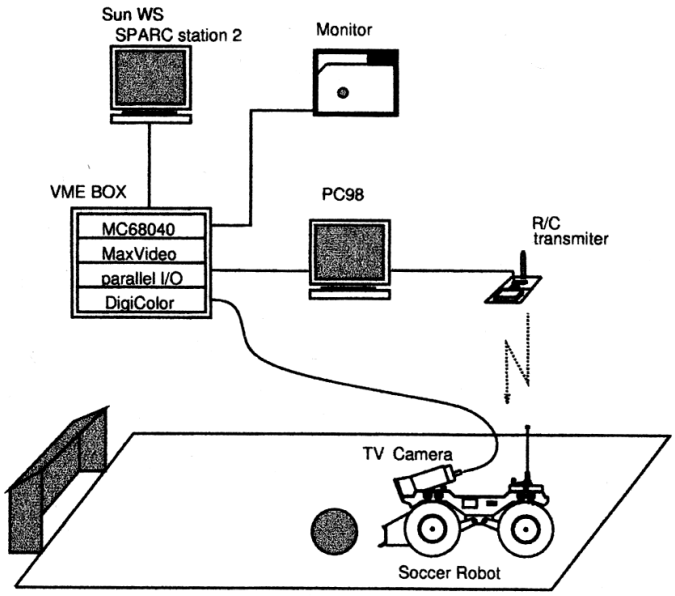
従って，POMDP に対応可能な行動獲得機構において，これらの研究における方法をいかに利用可能であるかが重要である．

1.2.3　複数タスクの学習

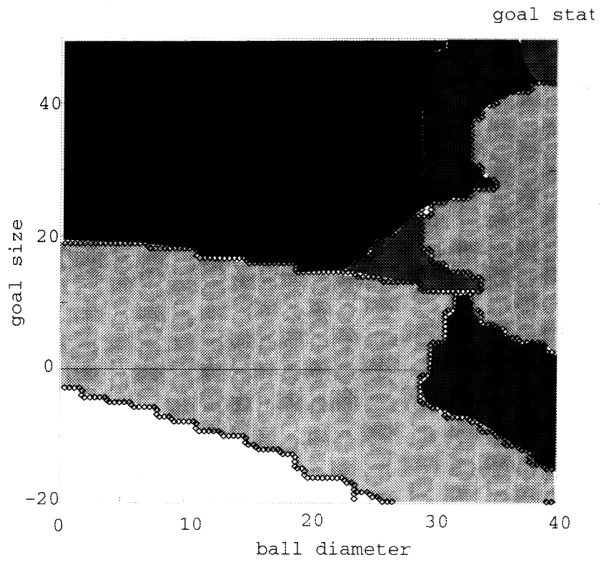
従来，複数タスクに対応する行動獲得のための学習手法はいくつか提案されている．これらのうち，特定の環境条件に対して獲得された行動方策を環境の変動に応じて適用させる方法 [18, 34] では，複数の問題に共通して利用可能な普遍知識を獲得し，タスク依存の知識を修正する，あるいは環境の変動に応じて破壊される行動方策の範囲をできるだけ小さくするなどの方法が採られるが，既に獲得された個別のタスクに関する知識が利用可能のまま保持される保証がない．従って，獲得される行動の一般性を高めるためには，個別の環境・タスクに対応して得られた知識を適切に統合し，これらを適切に使い分けることのできる行動決定機構が獲得されることが望ましい [32]．しかし，上に挙げた研究はいずれも，POMDP および認識機構の自律的構成という問題を扱っていない．

これに対して，POMDP 学習における複数タスクへの対応に対しては，上に述べた信念状態を用いた方法がある [5]．信念状態とは，エージェントが状態空間上の各状態に存在する確率の確率分布ベクトルを，行動方策の記述において用いる内的状態空間とするという方法論である．信念状態はエージェントの状態推定の不確実性の情報を含んでいることから，複数の初期状態が確率的に生起するという問題設定において，初期状態の推定を行っ

た上で初期状態に応じた行動を行うことが可能となる．この枠組みによって，異なるタスクが複数存在する問題設定に対して，現在扱っているタスクの推定に基づく適切な行動方策の適用が可能となり，これを複数タスクへ応用するという可能性が考えられる．Thrun らの方法では，信念状態に対して Monte Carlo 法を適用することで連続的な状態空間を扱



(a) Experimental environment



(b) Constructed state-space

Fig. 1.5: State-space construction by a real vison-based robot [31]

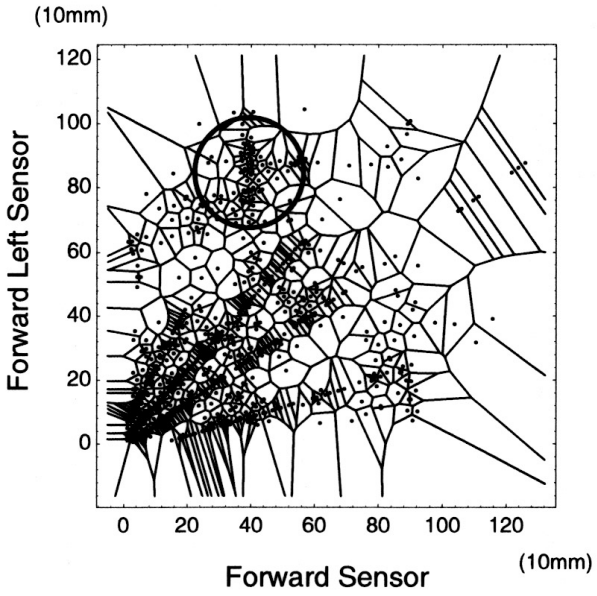


Fig. 1.6: State-space constructed by QLASS [12]

い，初期位置・姿勢に不確実性のある実ロボットタスクを実現している [19].

しかし，前述の通り信念状態の枠組みを利用するためには，信念状態を定義する上での定義域となる状態空間の構造，すなわちその次元と範囲が予め与えられている必要があるほか，観測確率分布，状態遷移確率分布といった事前知識が必要となる．この点で，前述の（2）の問題に対する解決が実現されていないと言うことができる.

これに対して，身体性に起因する前述の 2 点の問題の解決を図りながら行動獲得を行うためには，状態認識に対する事前知識を適用するのではなく，直接身体性に依存する観測入力・動作出力に即した表現に基づく状態認識機構を，タスクに依存する環境との相互作用を通じて獲得する必要があり，このようにして身体性に即して獲得された認識機構・行動方策に基づいて複数タスクへの対応を図らねばならない.

1.2.4 教 示

人工エージェントの行動獲得を補助する目的で，教示手法は多く提案されている [29].

最も単純な方法としては直接教示 (direct teaching) がある [28]．これは，教示者が直接ロボットのアームやツールを手に持って，行わせたい動作に沿ってロボットを動かすことにより動作をそのまま伝える方法である．また，これはその直接性において，ロボットの軌道を完全にプログラミングする従来の産業用ロボットへの作業の埋め込みと等しいと言

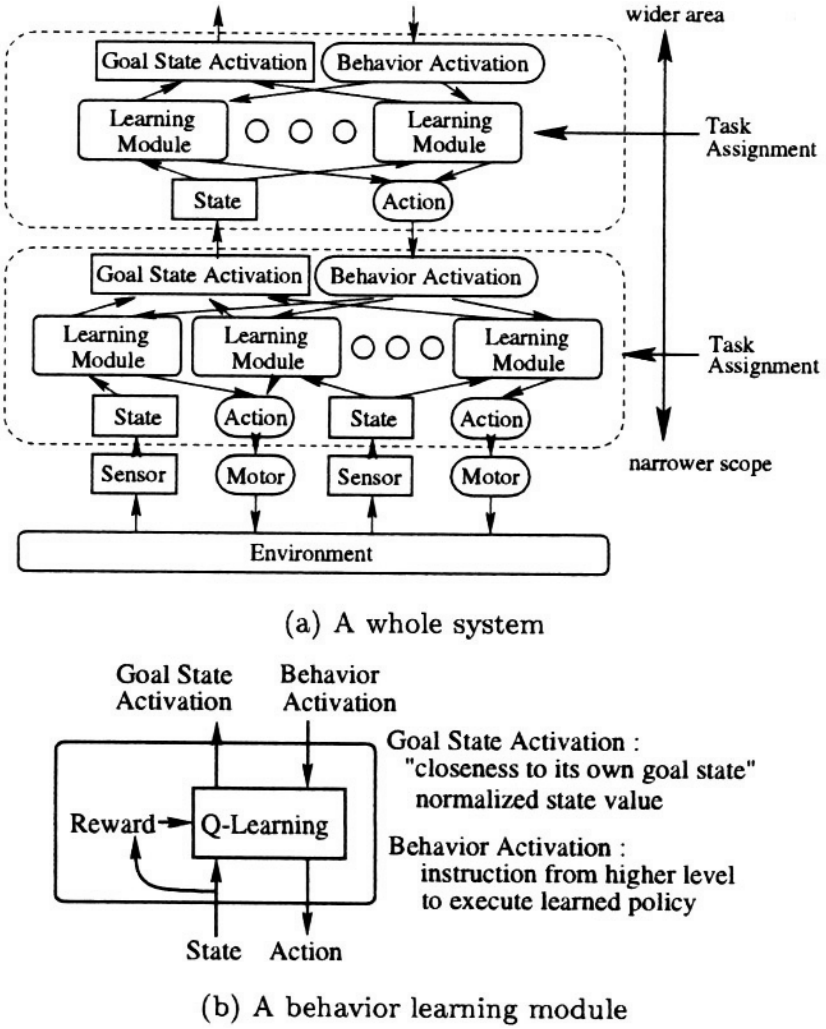


Fig. 1.7: Hierarchichy learning architecture [31]

える．このような方法では，エージェントの行うべき動作の系列を直接的に教示情報として与える必要があり，与えなければならない情報の量が大きいものとなってしまう．

このような直接教示による方法に対して，ロボットなどのエージェントの高機能化に伴ってエージェントの学習機能を利用することで少ない教示情報により行動獲得を目指す研究が近年では目指されている [25]．また，上述のような直接的な教示では，要求される一つの動作のみが教示されるのに対して，帰納学習や演繹学習といった方法で，より一般的な知識をより少ない教示の労力によって実現するという方向性が目指されている．

ところが，より一般的な知識を表現する上で行われる知識の抽象化においては，エージェントの状況認識の方法，及びそれに基づく行動方策の表現方法が必要となるが，前節で議論したとおり，これらは身体性を有するエージェントにとっては，本来は事前に明ら

かではなく，これを事前設計する場合，設計努力が大きいこと，および扱いうる対象に限定を加えてしまうということが問題となる．

即ち，外部からエージェントを眺める教示者の視点からは，エージェントの視点から眺められた世界がいかにエージェントによって解釈されるべきかがアプリオリではない．なぜなら，エージェントが身体性を通して世界と相互作用しているとき，前述の通りこの解釈の様態は，エージェントの身体性・環境・タスクに依存しているからである．従って，このような問題の下では，外的な観測者の視点に基づく知識を直接エージェントに教示することは困難である．

そこで，エージェントが実際に環境と相互作用を行いながらタスクを実行する過程で，エージェント自身により状況認識機構・行動決定機構を獲得するという方法をとり，教示者はその実現の上で必要となる評価信号，すなわち各時点でのエージェントの動作がタスク実現にとって良いか悪いかの評価である即時報酬をエージェントに与えるというアプローチを取ることが考えられる．

このような方法では，教示者は自らの外的な視点とエージェントの視点との差異について意識することなく，エージェントの行動に対して外部の視点からの評価を与えるのみによって，エージェントに作業を教示することが可能である．即ち，ここではエージェントが行う認識機構・行動決定機構の獲得機構が，教示者の評価の情報を自らの視点に基づく状況認識の様態に変換することで，教示者・エージェント間の視点のギャップを吸収する働きをしていると言える．

このようなアプローチによって，教示情報量を小さくすると同時に，エージェントの身体性に即した状態認識・行動決定機構を利用することが可能となり，予めエージェントの身体性に特化した設計を行うことが困難である場合でも労力の少ない教示を行いうるシステムを実現することができる．

1.3 研究の目的

以上の議論から，本研究では，以下の目的を置く．

部分観測的環境において，観測入力をタスク実現に対する意義に基づく状況識別へ変換する機構を全く有しないエージェントが，特定のタスクの実現のための行動を表現する状況認識機構および行動決定機構を環境との相互作用を通じて獲得し，あり得る複数のタスクの中から現在自らが扱っているタスクを識別して適切な状態認識機構及び行動決定機構を適用してタスクを実現するための行動獲得手法を提案する．ただし，ここでエージェントが各時点において行う動作に対して，外部から適切な評価信号が即時報酬として与えられるという仮定をおく．

1.4 論文の構成

本論文の構成は以下の通りである.

- 第1章では，研究の背景，従来研究及び研究の目的について説明した.
- 第2章では，本研究で扱う問題を定式化する．まず本研究で扱う問題を数学的に定式化する．更に，本研究においてエージェントにとっての時間を離散的なものとして扱うという想定に関して議論し，提案手法の教示システムとしての枠組みと意義について議論を行う.
- 第3章では，まず単一のタスクに対して身体性・環境・タスクに依存した状況認識・行動決定機構を即時報酬に基づいて構成する手法を提案する．まず扱う問題設定を定式化した上で，提案手法の概要を述べ，手法の各部分について詳細を説明した後，通路状のグリッド環境におけるナビゲーションを扱った計算機シミュレーションにより，提案手法を検証する.
- 第4章では，第2章において提案された手法により個別のタスクに対して獲得された状況認識・行動決定機構を適切に利用しながら複数タスク実現行動を獲得する手法を提案する．提案手法では，個別のタスクに対するタスク上の状況を識別する識別機構に対して，メタレベルのタスク識別機構を導入して識別過程を階層化することにより問題の簡単化を図る．タスク識別には，個別のタスクに対する学習の過程で得られた経験データを用いるが，タスク識別に対して経験が不足するという問題が起こり得るほか，環境の非Markov性に起因してタスク識別行動が個別のタスクに対するタスク実現行動に対して悪影響をもたらす可能性があり，これに対処するための追加学習のスケジューリングを行う．再び通路状グリッド環境におけるシミュレーションにより複数タスク実現行動の獲得を示す.
- 第5章では，以上までで提案した手法の実世界の問題への適用を図る．これにより，実世界における誤差の問題に対する扱い，および実ロボットの制御系と提案手法との結合の妥当性を議論する．具体的には小型移動ロボットを想定したシミュレーションにより複数タスクに対する行動獲得を行い，シミュレーション上で獲得された行動を実機のロボットに移植することで，獲得された行動の実世界における妥当性を検証する.
- 第6章では，提案手法に関する考察・評価を行う.

第 2 章

問題の構造化

2.1 はじめに 20
2.2 問題設定 21
2.3 離散的時間の想定 24
2.4 提案手法の教示システムとしての枠組みと意義 26
2.5 おわりに 28

2.1 はじめに

本章では，本論文で想定する問題の定式化し，提案する手法の教示システムとしての枠組みと意義について説明する．

第2.2節では，本論文を通じての問題設定を規定する．

第2.3節では，本論文における離散的な時間の想定について議論する．

第2.4節では，本論文で提案する手法の教示手法としての枠組みと意義について説明する．

2.2 問題設定

本節では，本論文で扱う問題の定式化を行う．

エージェントが世界に置かれた大域的な状況を外的状態 (external state) （以下，単に「状態」と記した場合は外的状態を示すものとする）とし，エージェントは世界から外的状態の手がかりとして観測 (observation) を受け取る．エージェントは世界に対して動作 (action) を出力することにより，外的状態を変更する．エージェントが動作を行う都度，エージェントはその動作の評価に相当する報酬 (reward) を世界から受け取る．

時間については，一単位を「ステップ」とする離散的な時間を考える．エージェントは1ステップあたり1回の動作を行う．なお，離散的時間の想定に関しては，第2.3節において議論する．

エージェントおよびエージェントの外的世界に関するモデルをFig.2.1に示す．

エージェントは，時刻tにおいて世界から観測 o_t および報酬 r_t を受け取り，世界に対して動作 a_t を出力する (図中 (a))．

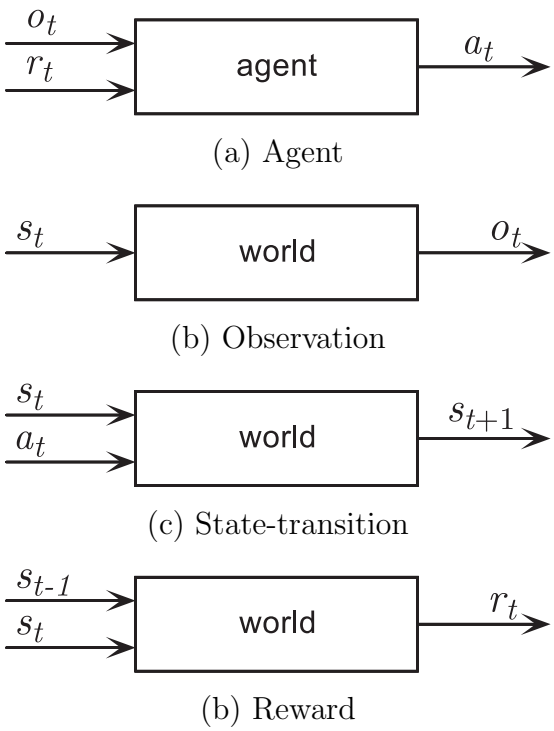
一方，外的世界は，エージェントの置かれた外的状態s_tのみに依存して，時刻tにエージェントが受け取る観測 o_t を与える (図中 (b))．また，エージェントの1ステップ前の外的状態 s_{t-1} と現在の外的状態 s_t とに依存して，状態遷移の評価として報酬 r_t を与える (図中 (c))．さらに，エージェントの現在の外的状態 s_t とエージェントが現在時刻に行った動作 a_t のみに依存して，1ステップ後の外的状態 s_{t+1} を生成する (図中 (d))．

この枠組みに基づいて，問題を以下のように定式化する：

- **外的状態**　$s \in \mathcal{S}$
  外的状態のなす空間（外的状態空間）の次元や取りうる値の範囲については，制限を設けない．外的状態空間上には，スタート状態s_{start}およびゴール状態の集合 S_{goal} を想定する．エージェントは外的状態空間上においてスタート状態から出発し，ゴール状態を目指すものとする．

- **観　測**　$o \in \Omega \subset \Re^{\dim(\Omega)}$
  観測については，あり得る観測のなす空間Ω (観測空間) は有限次元の実数空間を考える．

- **動　作**　$a \in \mathcal{A} = \{A_1, \cdots, A_{|\mathcal{A}|}\}$
  動作としては，動作空間 (action space) として，離散化された動作の集合$\mathcal{A}$ が与えられているものとする．



(a) Agent

(b) Observation

(c) State-transition

(b) Reward

Fig. 2.1: Model of agent and world

- **報　酬**　$r \in \Re$
  報酬は，実数スカラ値からなる．

- **観測関数**　$O : \mathcal{S} \to \Pr(\Omega)$
  外的状態のみに依存して観測値を確率的に規定する観測関数を想定する．

- **状態遷移関数**　$T : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$
  特定の時刻における外的状態と，その時刻においてエージェントが行う動作のみに依存して1ステップ後の外的状態を規定する状態遷移関数を考える．ここでは，状態遷移は決定論的であるとし，確率的状態遷移は想定しない．

- **報酬関数**　$R : \mathcal{S} \times \mathcal{S} \to \Re$
  遷移前の外的状態と遷移後の外的状態との関数として，各ステップの動作の良し悪しを適切に評価する即時報酬を仮定する．

以上のモデルのうち，エージェントに事前に与えられている情報は以下のとおりである：

- 観測空間の次元および各次元の値の範囲および観測空間上での距離の測度（ここではユークリッド距離を用いる）．

- 可能な動作の有限集合.

これらは，エージェントの低次のセンサ・モータ系の設計に直接に依存する知識である.

以上の問題設定の，従来のPOMDP研究との相違点について述べる．信念状態に基づく方法では，状態空間の空間的構造，遷移状態確率T，観測確率Oに関する事前知識を前提していた．一方で，短期記憶を明示的に用いる方法では，観測空間Ωを離散的有限集合として表現していた．これに対し，本研究では状態空間に関して事前知識を仮定せず，観測空間を事前には切り分けない.

エージェントの一連の行動は動作・観測を交互に繰り返す系列として考える．以下，添え字tは時間ステップを表し，s_tは時刻tにおける外的状態，a_tは時刻tに行った動作，o_tは時刻tに得た観測，r_tは時刻tに受けた即時報酬を表す.

2.3 離散的時間の想定

本節では，本研究全体を通じて想定している離散的な時間について議論する．

実在の身体性を有するエージェントである実ロボットでは，サンプリング時間の大小に関わらず，何らかのサンプリング時間を周期として観測の入力，動作の出力を行っている．従って，身体性を有するエージェントにその離散的な時間の周期を想定することは一般的な問題設定であるといえる．

ただし，ここで述べたサンプリング時間とは，エージェントの動作を決定する最小の時間単位であり，一般的にはこのサンプリング時間は事実上連続と考えてよい程度に短い．このサンプリング時間を周期としてエージェントが意志決定を行うことは，以下の問題を意味する：(1) 計画に基づく意志決定において：計画は一般には一定以上の計算コストを伴うため，その周期が極めて短い場合，エージェントの行動速度に対して計算時間がボトルネックとなる，(2) 学習に基づく意志決定において：サンプリング時間が短くなればなるほど，必要となる意志決定回数が増大し，この結果として学習において扱うべき探索空間が増大するため，学習コストが増大する．以上から，極めて短いサンプリング時間に対して意志決定を行う方法によりエージェントを動作させようとする場合，エージェントの知能は反射的なものでない限り現実的なパフォーマンスを実現できないと言える．従って，計画・学習を行うエージェントに対しては，動作・観測の周期としての動作サンプリングを，ハードウェアの与える最小サンプリング時間よりも長いものとして与えるという方法，即ち何らかの方法で時間方向の分節化を行うという方法が現実的である．

一方で，観測値および動作出力に関しても，一般には最小単位をセンサ入力あるいはモータ出力の分解能として細分することが可能であり，この分解能が十分小さい場合は事実上連続量と考えることができる．これらについても最小単位に関して細分化した一つ一つのセンサ入力・モータ出力を計画・学習の対象とすることは理論的には可能であるが，実際上計算コストの観点から，これらをそのまま扱うという方法は現実的ではなく，一般的には閾値を設けるなどによりこれらを十分小さな数のまとまりとして分節化した上で扱うという方法が採られる．

本研究での序論で述べたとおり，本研究では時間・センサ入力・動作出力のそれぞれの空間を分節化する上で，その分節化の境界条件はエージェントと環境・タスクの間の相互作用に基づいてエージェント自身により規定されることが望ましいという立場をとっている．

従って，理想的には時間・観測・動作のそれぞれの（事実上）連続的な空間についてそれぞれ適応的な分節化を加えていくという方法によって問題を扱うことが望ましい．

しかし，この場合それぞれの空間をいかに分割するかに関しての判断基準をどのよう

に与えるかという問題が極めて困難なものとなる．なぜなら，本研究での想定であるエージェントの観測における部分観測性が存在する場合，エージェントが現在置かれた外的状態を識別するためには，過去から現在時刻に至る動作・観測の短期記憶をその判断材料として利用する必要があり，これを分節化する場合，時間・観測・動作の全ての次元を同時に扱う必要があるからである．

これに対して，特定の空間（例えば観測）が連続的であり，それ以外の空間（時間・動作）が既に分節化されていた場合，既に分節化された空間の構造により，これから分節化しようとする空間に対する分節化の判断基準をより詳細に与えることが可能である．本研究では，問題を扱う上での最初の段階として，ここに示すように特定の空間のみを自律的に分節化するというアプローチをとる．

具体的には，エージェントの基本的動作要素として動作プリミティブを予め設計し，動作空間を有限の離散的な集合として規定するとともに，それぞれの動作プリミティブの開始から終了へ至る時間を1ステップと考えることで，時間方向の分節化を行い，この上で，連続的な観測空間を部分観測性を考慮しながら適応的に分節化するという問題を扱う．部分観測性に起因して，エージェントの状態空間は現在から過去へ至る観測・動作の系列に基づいて規定される必要があり，エージェントの動作様式として，動作を行った後に観測を行い，動作選択を行った上で再び動作を行うというモデルを採用する場合，状態空間は，特定の時点での観測・動作を識別における属性値とする決定木の構造として表現することが可能である．即ちこのとき，決定木の縦方向の階層は離散的な時間ステップにより区切られ，横方向に関しては，動作に対応する階層では所与の動作プリミティブにより分類がなされる．これに対して，提案手法では観測に対応する階層における横方向の分節化を適応的に行う．

2.4 提案手法の教示システムとしての枠組みと意義

本節では，本研究における提案手法の教示システムとしての枠組みを説明する．

2.2に，基本的な枠組みを示す．システムは（1）外部世界,（2）教示者,（3）エージェントの3つの要素から構成される．

各時刻においてエージェントが置かれた状態s_t（ただし，これは外的な視点に基づく，大域的に規定される状態）に即して，エージェントは観測値o_tを得る．ただし，エージェントの身体性に起因する観測能力の限定から，このo_tは外的状態s_tを一意に特定するのに十分な情報量を一般には含まない．

これに対して，教示者は外部の視点から，外的状態量s_tを観測する．教示者はこの観測に基づき，前回の時刻$t-1$でエージェントが行った動作a_{t-1}による状態遷移（$s_{t-1} \rightarrow s_t$）がタスク実現に対して寄与しているか否かに即した評価信号として，即時報酬r_tをエージェントに与える．

エージェントは，観測値，自らが行った動作の短期記憶に基づいて，自らの現在の状況に対応する内的状態x_tを求め，これに対応する行動方策に基づいて現時点で行う動作a_tを選択する．

動作a_tは外的世界へ出力され，時刻tにおける外的状態s_tと動作a_t の関数として次の時刻s_{t+1}における外的状態s_{t+1}が決まる．

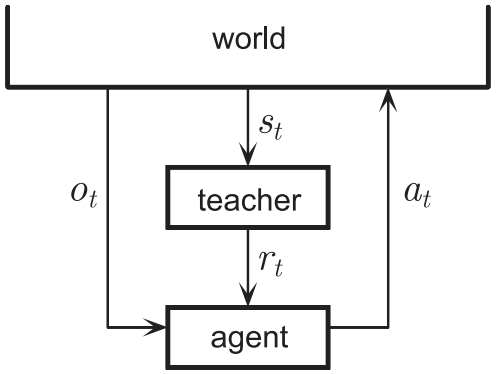


Fig. 2.2: Framework of the proposed method

このシステムが実現するのは，以下の特徴を備えた教示システムである：エージェントの大域的な状態を外部から観測する教示者が，各時点のエージェントの動作が帰結するエージェントの状態遷移の評価をエージェントに与えることにより，各時点での状態に対応する評価の高い動作を選択することが可能な状態認識機構の構築，およびそれに基づく行動決定機構の獲得をエージェント自身が実現する．

このようなシステムにより，教示者はエージェントのセンサ・アクチュエータの特性（いわばエージェント側の「都合」と言える部分）や，エージェントの視点から眺められた環境の様態について考慮することなく，各動作の良し悪しの評価のみを与えるという小さい教示労力によってエージェントに行動を教示することができる．

2.5 おわりに

本章では，本論文で想定する問題を定式化し，提案する手法の教示システムとしての枠組みと意義について説明した．

第2.2節では，本論文を通じての問題設定を規定した．

第2.3節では，本論文を通じての想定である離散的時間について議論した．

第2.4節では，本論文で提案する手法の教示手法としての枠組みと意義について説明した．ここでは，提案手法は，外的視点からエージェントを眺める教示者が，各時点におけるエージェントの動作を評価する即時報酬という少ない教示情報のみに基づいて，エージェントの状態認識機構および行動決定機構を教示する教示システムとして捉えることができることを示した．

第 3 章

単一タスクに対する学習手法

3.1 はじめに · · · 30
3.2 概 要 · · · 31
3.3 Utile Suffix Memory · · · 33
3.4 状況認識機構の構成 · · · 37
3.4.1 状態表現 · · · 37
3.4.2 学習過程全体の流れ · · · 41
3.4.3 状態分割 · · · 42
3.5 シミュレーション · · · 49
3.5.1 シミュレーション条件 · · · 49
3.5.2 比較対象とする学習手法 · · · 50
3.5.3 結 果 · · · 51
3.5.4 考 察 · · · 52
3.6 おわりに · · · 57

3.1 はじめに

本章では，連続的・多次元の未設計の観測空間を持つエージェントが，部分観測環境において，即時報酬に基づいて状態認識機構を構築し，行動獲得を行う手法の説明を行う．

まず第2.2節では，提案する手法において対象とする問題の設定を行う．

第3.2節では，単一タスクに対する認識機構およびそれに基づく行動決定機構の獲得手法についての概要を説明する．

第3.4節では，手法の詳細について説明する．

第3.5節では，通路状グリッド環境に関して行ったシミュレーションについて説明する．

3.2 概 要

本節では単一タスクに対して状況認識機構と行動決定機構を獲得する手法について，その概要を説明する．

エージェントのモデルをFig.3.1に示す．エージェントは各時刻において得られる不完全な観測 o_t と即時報酬 r_t に基づいて，現在置かれている状況を判断して対応する内的な状態 x_t を求め，その状態に応じた動作を出力する

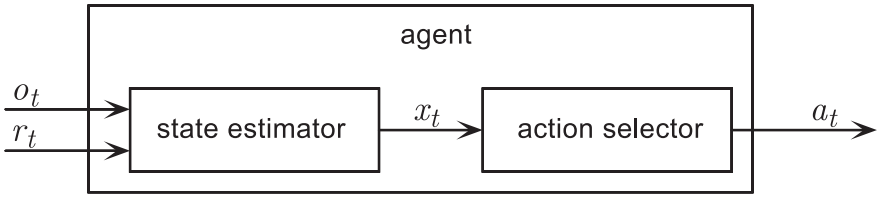


Fig. 3.1: Internal state estimation

このとき，観測は部分観測性を持ち，また観測空間の解釈方法が事前に与えられていないことから，エージェントは内的状態空間を環境との相互作用に基づいて構築する必要があり，更にこの際に観測空間の構築を行わなければならない．部分観測性に対処するためには内的状態の規定において短期記憶の情報が用いられねばならないことから，状況認識機構は過去の観測・行動の履歴を入力として内的状態を出力するものとなる．

これを実現する一つの方法として，[33, 6] におけるように，リカレントニューラルネットワークを用いて状態認識を実現する方法が考えられるが (Fig.3.2)，このような方法では，短期記憶の情報を内的状態に変換する写像過程がブラックボックス化し，エージェントの内的モデル妥当性や具体的な認識方法，拡張性などについての評価が難しく，またこれらは単純な環境においては動作しうるが複雑な環境においては局所解へトラップされることが多い．

従って，本研究では，内的状態を離散的に分節化する形での状態表現として扱い，内的状態表現の上では，状況識別のために用いられる短期記憶のデータを明示的に扱う．

このようにエージェントの過去の経験に基づいてPOMDPにおいて状態識別を行う方法論として，信念状態を用いる方法 [5] があるが，これを利用するためには，予め外的な状態空間の構造が分かっている必要があるほか，観測確率分布，状態遷移確率分布についての知識をエージェントが持っている必要がある．

従って，ここでは短期記憶を明示的に状態識別に利用して，離散的な内的状態表現を生成する方法を採用する．具体的には，[10, 16] と同様の，エージェントの短期記憶に基づ

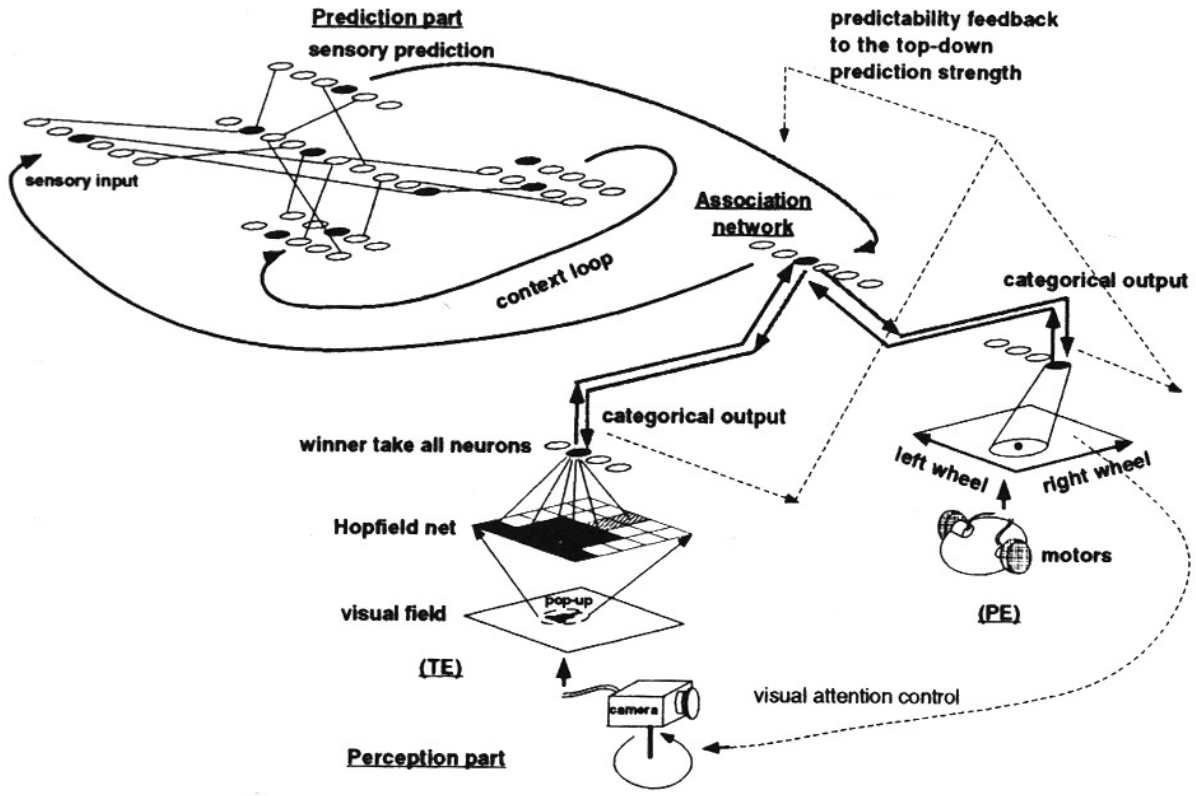


Fig. 3.2: RNN-based recognition mechanism [33]

く決定木構造の状態表現を採用する．これにより，エージェント内部で，短期記憶を用いた状態識別の手順が明示的に表現される．

提案手法では，この状態表現を環境との相互作用に基づいて逐次的に構成する．具体的には，単一の状態からなる最も単純な状態表現から開始し，必要に応じてこれに分割を加えてゆくことで，最終的にタスク実行に適した状態表現を得る．

状態表現構成の目的は，単一の内的状態において同一の動作を実行した場合に得られる即時報酬が一定となることとする．そこで，過去に報酬にばらつきがある場合には状態分割の必要が生じるが，ここで扱う問題では観測に基づく識別の様相が所与でないことを仮定しているため，報酬のばらつきには以下の 2 つの原因があることになる：(1) 知覚騙し問題，(2) 観測の識別が粗すぎる．従ってこれらのうちどちらの原因がより大きいかを判別しながら，それぞれの原因に対応する対処法として状態分割を実施する．

3.3 Utile Suffix Memory

本節では，単一タスクに対する状況識別機構および動作決定機構に関する提案手法が採用している状態表現について，基本的にその構造を踏襲しているUtile Suffix Memory (USM)[10]について，概略を説明する．

McCallumの提案したPOMDP学習手法であるUSMは，履歴情報を明示的に利用して環境の部分観測性に対処するために，生の経験データに基づく決定木構造の状態表現を自律的に構築する．

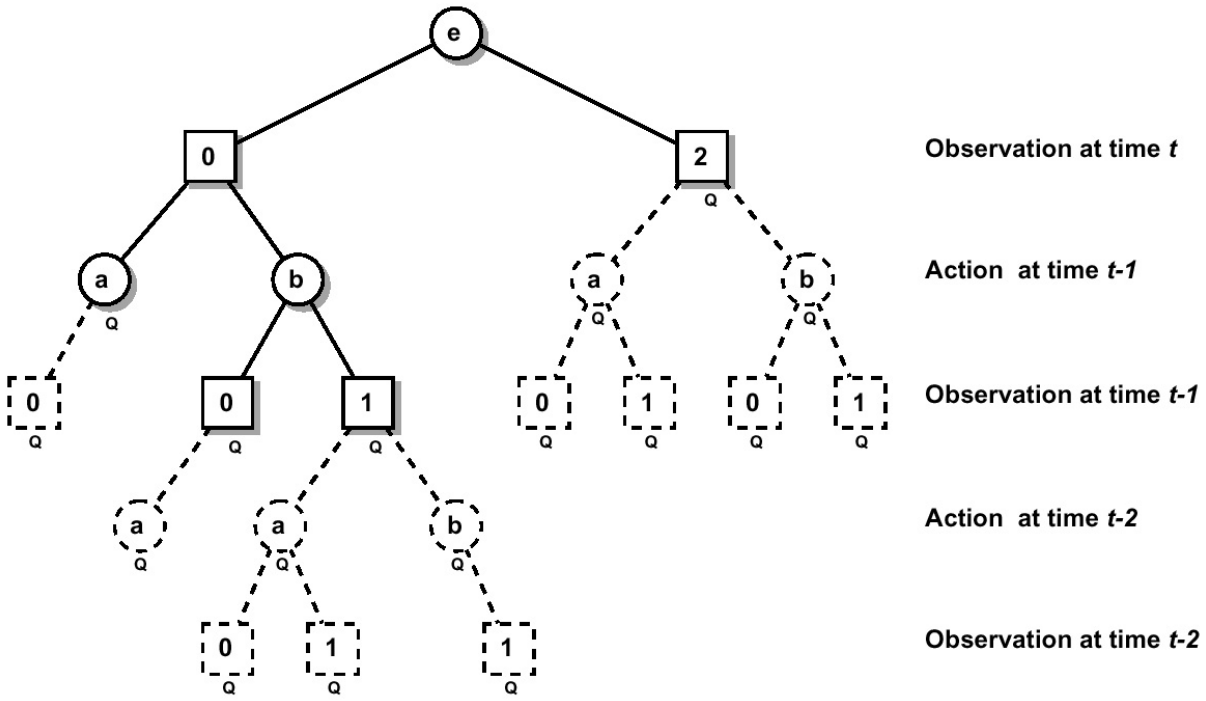


Fig. 3.3: State-representation in USM [10]

Fig.3.3は，USMにおける木構造の内部状態表現である．木構造の各レイヤはそれぞれ，特定の時刻における観測と動作に基づく識別を表しており，根ノード (図中“e”と表記されたノード) から下方向に，現在の観測，1ステップ前の動作，1ステップ前の観測，2ステップ前の動作· · ·と，過去に遡る形で識別が加えられてゆく．図中，四角形のノードは観測に基づく識別を表すノード，円形のノードは動作に基づく識別を表すノードである．この木構造のうち，状態に相当するのは葉ノードであり，ここには対応する状態における各行動の評価の見積もりであるQ値が格納されている．

また，葉ノードの更に下のレイヤには房 (fringe) ノード群が追加される．これは，更に

1 段階の分割を加えたとした時に根ノードになるものであり，実際にその部分を分割するか否かは，これらの房ノード間における報酬の統計的分布の差異を Kormogorov-Smirnov 検定により検出することで判断される．これにより，タスクに対して必要十分なサイズの状態表現の構成を実現する．

USM における状態表現の具体的な説明のために，Fig.3.4 に，(a) エージェントがたどった経路，(b) 対応する短期記憶，(c) USM における決定木構造状態表現における対応する状態の識別を示す．

(a) 図に示したのは，移動ロボット (円形) が時刻 $t = 1$ にスタート点を出発し，状況 (B) を経て状況 (A) に至るまでの経路である．

この経路に沿った観測・行動の経験を (b) に示した．ただし，四角形は観測，円形は動作を表しており，その内容は以下の通りである：

観測	N	何も観測していない
	L	左方向に障害物を観測
	FL	前方と左方向に障害物を観測
	LB	左方向と後方に障害物を観測
動作	F	前進
	L	左に 90 度回転
	R	右に 90 度回転

ただし，ここではロボットは自分から見て前後左右方向に障害物が存在するか否かのみを観測するものとする．

(c) には，この観測・動作シーケンスに基づく状態の識別を最大限詳細に行った場合の，状態表現を示した．ただし，木構造の深さの最大値を「3 ステップ前の観測」に対応するレイヤまでとした．

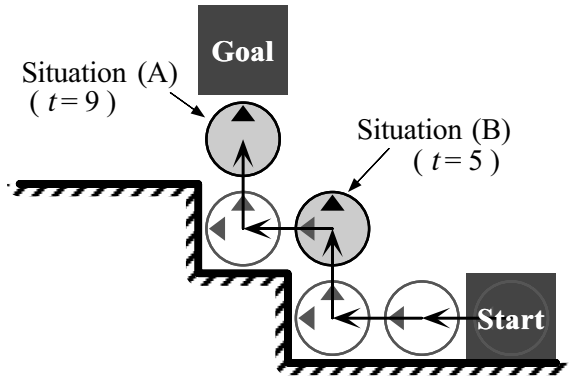
まず，スタート時点 $t = 1$ では，ロボットは左側にのみ障害物を検出する (観測 “L”)．木構造では，この状況は，現在の観測に関する識別のレイヤ (o_t) において観測 “L” に対応するノードが選択される．それ以前のインスタンスは存在しないため，木構造上で現在の状況に対応するノードは “1” と表記されたノードとなる．

$t = 2$ のときは，現在の観測が “L”，1 ステップ前の動作が “F”，1 ステップ前の観測が “L” であることから，レイヤ o_t において “L” が選択された後，a_{t-1} のレイヤにおいて “F”，o_{t-1} において “L” が選ばれ，結局 “2” と表記されたノードが現在の状況に相当するノードとして選ばれる．

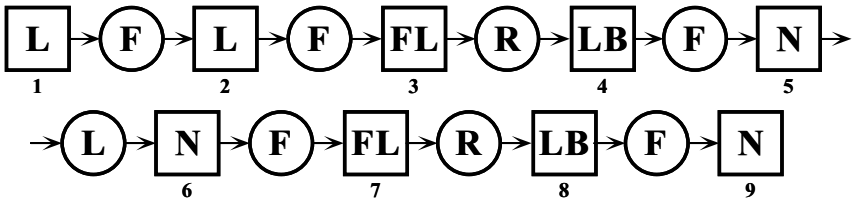
また，知覚騙し問題の例として 2 つの状況 (A) と (B) を示した．これらの状況では，現在の観測値はともに “N” であるため，過去の経験を参照しなければ状況が識別不可能であ

る．2つの状況の識別は，木構造上では以下の通りに行われる．

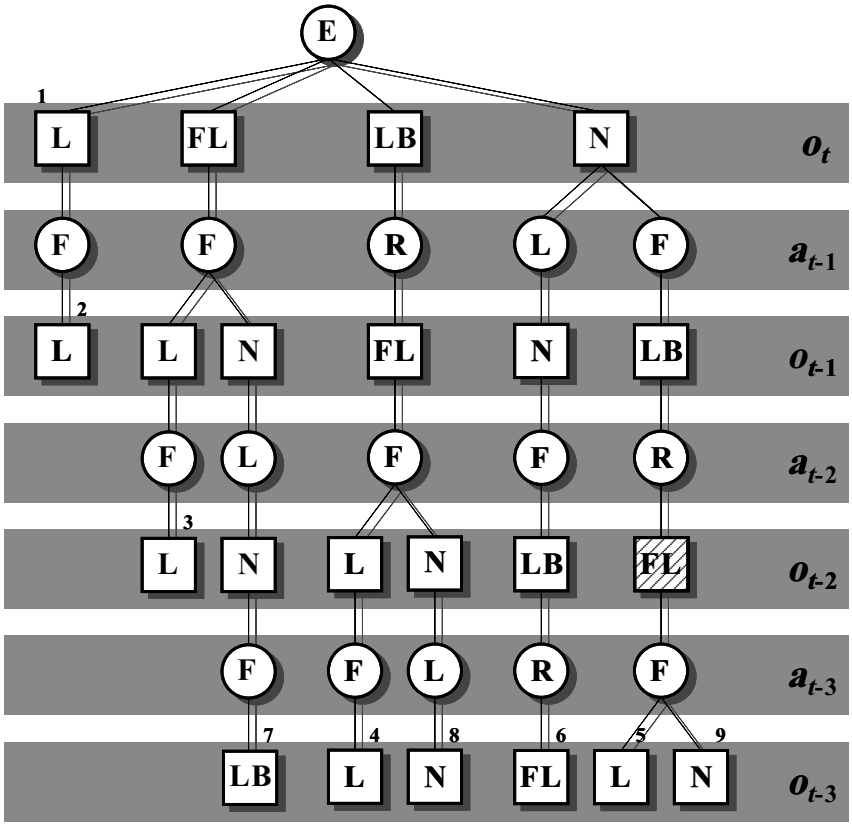
1. 現在の観測はともに “N” であるため，レイヤ o_t ではともに右端のノードが選ばれる．
2. 1ステップ前の動作，1ステップ前の観測，2ステップ前の動作，2ステップ前の観測はともに “F-LB-R-FL” であるため，同様に各レイヤの右端のノードがたどられる．ここまでては，2つの状況は o_{t-2} にともに対応するため，これ以上下に分割されたノードが存在しなければ，状態表現上で2つの状態が同一のノードに混同されている (斜線で示したノード)．
3. 更に1ステップ前まで遡るとき，3ステップ前の動作は同じく “F” だが，3ステップ前の観測において，状況 (A) においては “N” の観測が得られていたのに対して，(B) においては “L” であり，この時点の観測に基づく識別のレイヤを加えることにより，2つの状況に対して異なるノードが対応する．



(a) path of robot



(b) sequence of actions and observations



(c) state-representation-tree

Fig. 3.4: Schematic view of USM

3.4 状況認識機構の構成

本節では，本論文において提案する，単一タスク実現に対する状況認識・行動決定機構の構成手法について詳細を説明する．

本研究で提案する状態構成方法を説明する．提案手法では，与えられた身体性・環境・タスクに対して適切な状態構成を目指す．即ち，状態構成はタスクにおける合目的性を目指して行われるため，行動学習と同時並行的に，各時点における行動学習の結果に即した形で行われる．

3.4.1 状態表現

提案手法では，短期記憶を明示的に表現するために，Utile Suffix Memory （USM）[10] やBLHT[16] と同様の，観測・動作のシーケンスに基づく決定木構造の状態表現を用いる．状態表現の概略をFig.3.5に示す．

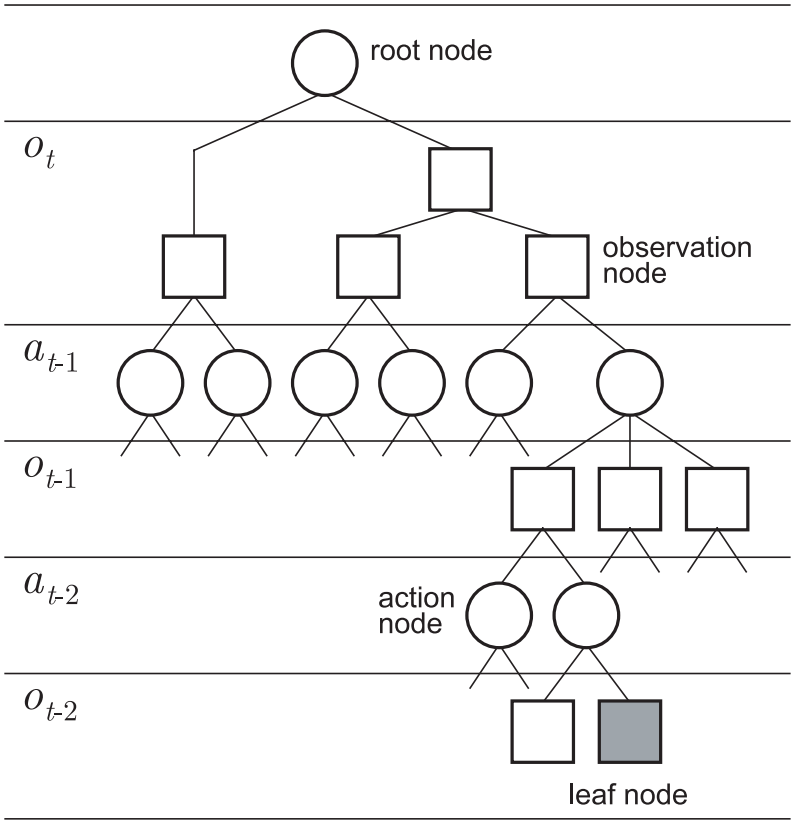


Fig. 3.5: State-representing tree

状態表現は，短期記憶を表現する決定木の構造をもつ．この木構造中で現在の状況に最も近いノードが現在の状態に対応したノードであり，エージェントは各時点においてこれを探索する．

各レイヤは，現在時刻からさかのぼった時間における観測・動作に基づく識別を示している．すなわち，o_{t-i} と示したレイヤは i ステップ前の観測，a_{t-i} と示したレイヤは i ステップ前の動作に基づく識別である．このように，この決定木は現在から短期記憶を順次さかのぼる形で識別を加えていく構造になっている．観測に対応するレイヤを**観測レイヤ (observation layer)**，動作に対応するレイヤを**動作レイヤ (action layer)** と呼ぶ．

観測レイヤに含まれるノード（四角形で示した）を**観測ノード (observation node)** と呼び，ここには識別のための**参照ベクトル (representative vectors)** が一つあるいは複数記載されている．

動作レイヤにあるノード（円で示した）を**動作ノード (action node)** と呼び，エージェントに可能なそれぞれの動作に対応する．

エージェントは，過去の経験を固定長のインスタンスの配列という形で保持している（**インスタンスベース (instance-base)** と呼ぶ）．時刻 t に対応するインスタンスは，

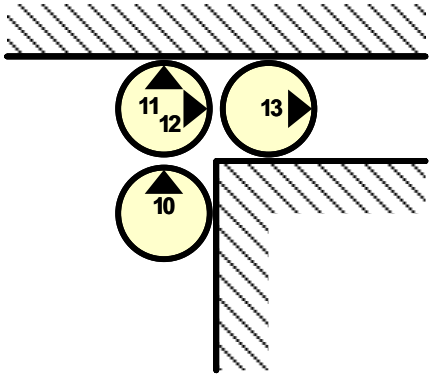
$$\langle o_t, r_t, a_{t-1}, st_t \rangle$$

として表現される．ただし，st_t は時刻 t における該当試行でのステップ数 (後述) である．インスタンスベースの配列は固定長であり，この長さを超えるほど過去のデータは忘却される．インスタンスベースは，短期記憶における過去の観測・動作を求める際，および特定の状態を過去に訪れた経験を求める際に参照される．

Fig.3.6に，インスタンスベースの例を示した．図中 (a) は円形の移動ロボット (姿勢を三角形で示す) が，時刻 10 から 13 までの間に移動した様子を示しており，(b) はこの一連の動作シーケンスに対応するインスタンスベースを示している．ただし，(b) において，観測の “F”, “L”, “B”, “R” はそれぞれ，ロボットの前方，左，後方，右方向の障害物の有無を示し，動作 “F” は前進，“R” は右回転を示すとする．

状態表現の中で，内的状態に相当するのは葉ノードであり，インスタンスベースを参照することで得られる観測・動作の短期記憶に基づいて，エージェントは各時点における状況に対応する内的状態を探索する．探索は，根ノードから下方向へ，実際の短期記憶と最も近いノードをたどることにより行う：

(1) 観測レイヤでは，該当時点に得られた観測ベクトルと最もユークリッド距離の近い参照ベクトルを持つ観測ノードをたどる (従ってこのレイヤに対応する観測空間は，参照ベクトルに基づくボロノイ超平面で分割される)．



(a) Action sequence

time	observation				action	reward	step
	F	L	B	R			
10	0	0	0	1	-	0	0
11	1	0	0	0	F	1	1
12	0	1	0	0	R	-1	2
13	0	1	0	1	F	-1	3

(b) Corresponding instance base

Fig. 3.6: An example of instance-base

Fig.3.7に，状態表現の観測レイヤにおける観測空間の識別の模式図を示す.

図では，簡単のためにs_1，s_2の2次元からなる観測空間を想定した．この観測レイヤにおける最上位の分岐 (行動ノード直下の観測識別) における，左右の観測ノードの持つ参照ベクトルを，それぞれ灰色，白色の小さな円で示した．このとき，観測空間は太線により図のように分割され，現在エージェントが持っている該当時刻に対応する観測値の記憶が，このどちらの領域に入っているかに応じて，ノード探索における分岐のそれぞれの方向がたどられる．また，状態構成手順によっては，一度分割された観測空間が更に分割されることもある (図中左下の分岐).

(2) 動作レイヤでは該当時点に実際に行った動作のノードをたどる.

このように過去の記憶にさかのぼりながらノードをたどり，葉ノードにたどり着いたか，あるいは試行開始時刻に対応する観測レイヤにたどり着いたとき，そのノードが現在の状態に対応する（状態に対応するノードを**状態ノード (state-node)** と呼ぶ）．なお，時刻tの状況に対応する状態ノードをx_tと表記する.

状態ノードとなりうる全てのノードxには，内的状態xにおいて動作aを行うことに対

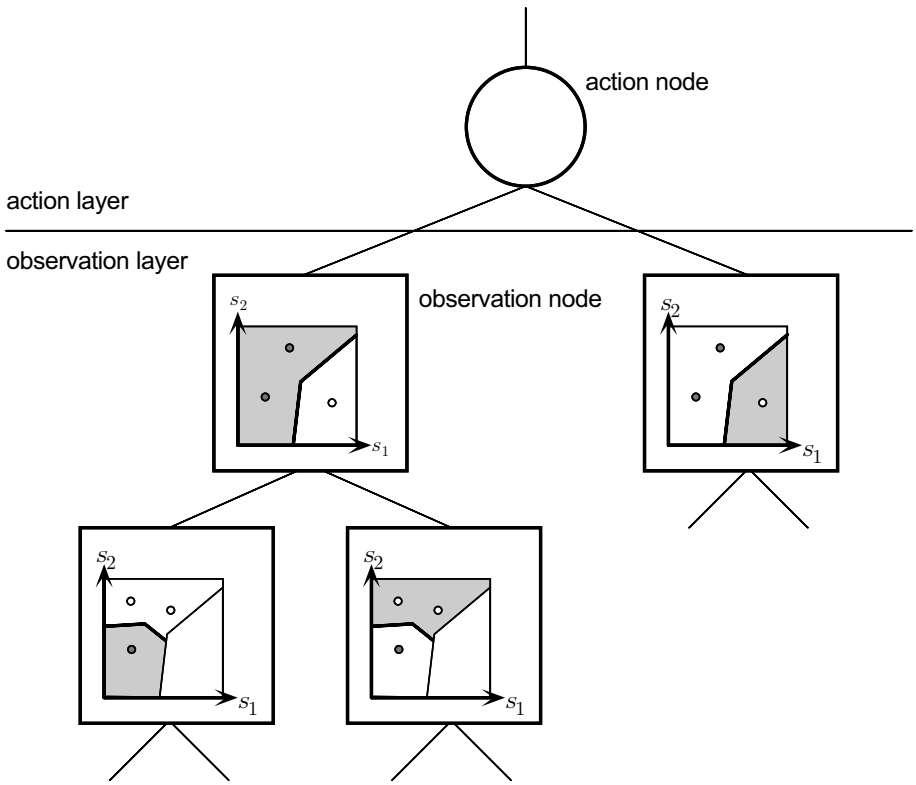


Fig. 3.7: Distinction based on observation

する評価の見積もりとして価値関数 (value function) $Q(x, a)$ が記録されており，実際に得られる即時報酬に基づいてこれを更新することによって行動学習を行う．

また，状態ノードには，過去にそのノードを訪れた時刻がインスタンスベースに記載されている範囲内で全て記録されている．これを**インスタンスリンク**と呼び，$\mathcal{I}$ と表記する．インスタンスリンクは状態分割において判断材料として利用される．

状態構成は，単一の状態からなる状態表現から開始する．すなわち，根ノードと，レイヤ o_t にある一つの観測ノードである．このノードを適宜分割することにより，適切な状態表現の構成を目指す．状態表現構築の最終的なゴールは，各状態ノードにおいて，同一の動作を行った場合には同一の即時報酬が得られるようにすることである．

ここで示した状態表現の，USM や BHLT との主要な違いは，観測レイヤにおける観測に基づく識別において，予め与えられた離散的な観測ベクトル集合を用いるのではなく，これを Voronoi 超平面による空間の分割という形で，実際の経験に基づいて学習の過程で規定することである．

3.4.2 学習過程全体の流れ

エージェントは各試行においてスタート状態 s_{start} から開始し，ゴール状態集合に含まれる状態に到達したとき（$s_t \in \mathcal{S}_{\mathrm{goal}}$），あるいは所与の最大消費ステップ数に達したとき，1回の試行を終了する．

各時刻 t においてエージェントが行う手順は以下の通りである (Fig.3.8)：

1. 観測 o_t，即時報酬 r_t を受け取る．これらを前回行った動作 a_{t-1} とともにインスタンスベースに記録する．
2. 状態表現木のうち，現在の状態に対応するノード x_t を求める．
3. 前回訪れた状態ノード x_{t-1} を分割すべきかどうかを判断し，必要に応じて分割を実施する．
4. 前回の状態ノード，前回の動作に対応する価値関数 $Q(x_{t-1}, a_{t-1})$ を以下のように更新する：

$$Q(x_{t-1}, a_{t-1}) \leftarrow (1-\alpha)Q(x_{t-1}, a_{t-1}) + \alpha r_t \tag{3.1}$$

   ただし，α はステップサイズパラメータである．
5. $s_t \in \mathcal{S}_{\mathrm{goal}}$ のとき試行を終了する．
6. 価値関数に基づいて，Q 値の大きい動作 a_t を Boltzmann 分布に基づいて確率的に決定する．即ち，動作 A_i が選択される確率は以下の通りである：

$$\Pr(A_i) = \frac{\exp\left(Q(x_t, A_i)/T_b\right)}{\sum_{A_j \in \mathcal{A}} \exp\left(Q(x_t, A_j)/T_b\right)} \tag{3.2}$$

   ただし，T_b は Boltzmann 温度パラメータである．

式 (3.1) の意味するところは，得られた即時報酬に対して Q 値を近づける方向での Q 値更新であり，ステップサイズパラメータ α は，1回の更新における Q 値の更新幅を与える．

また，式 (3.2) は，より大きな Q 値に対応する行動がより選択されやすくなるような確率的動作選択を意味する．温度パラメータ T_b は，大きくなるほど低い Q 値を持つ行動も選択されやすくなり，行動にランダムネスを与える．このランダムネスにより学習過程において探索的行動が促進され，局所解へのトラップを避けるが，T_b が大きすぎることは得られた優良な解からの逸脱の頻度が高いことを意味し，このとき行動のパフォーマンスは低下する．

3.4.3 状態分割

第3.4.2項で述べた手順における 3. において，状態ノード x_{t-1} に対する分割の必要性の判断，および分割の実施方法について説明する．

3.4.3.1 状態構成の目的

状態構成の目的は，状態表現中に含まれるそれぞれの状態ノードにおいて，その状態を訪れた際に，同一の行動が同一の意義，すなわち報酬をもたらすということである．

この基準を採用することで，状態構成はタスク実現に対して意義を持つ分割だけが実施されることになり，無駄な状態分割を省く傾向をもたらす．

このことが重要なのは，一般に学習エージェントの状態空間が大きくなればなるほど，その状態空間を表現するために必要なメモリ量が増加するのみならず，そのように詳細に分節化された状態それぞれに対して行動の学習を行う必要が生じるため，学習のコストが高価なものとなるからである．

3.4.3.2 分割の実施の決定

前目で述べた状態構成の目的から，特定の状態を分割する必要性の有無は，過去にその状態を訪れ，同じ動作を行ったときに得られた報酬の値が一定であるか否かで判定可能である．判定手順は以下の通りである（現在時刻を t とする）．

状態ノード x_t のインスタンスリンク集合 $\mathcal{I}$ を，それぞれのインスタンスリンクが示す時刻の 1 ステップ後に行った動作 a ごとの集合 $\mathcal{I}_a$ に分け，各 $\mathcal{I}_a$ ごとに，行った動作に対して得られた報酬の標準偏差を求める．$\mathcal{I}_a$ の要素数が閾値 ν を超え，かつ標準偏差が閾値 δ を超える $\mathcal{I}_a$ が存在するとき，分割の実施を決定する．以下，分割方法の判断基準として，最大の標準偏差を与える $\mathcal{I}_a$ を用いる．

ここで，要素数に閾値を設けたのは，状態ノードが生成された直後であり，まだそのノードにおける動作の学習が不十分であるときは，動作に対する即時報酬のばらつきが大きくなってしまうためである．

3.4.3.3 分割方法の決定，分割の実施

USM においては，状態構成が不十分であるときの対処方法は一意に決定可能であった．なぜなら，動作空間および観測空間が，予め有限の離散的集合として与えられており，より詳細な分割を加えた場合の分割後の状態表現が一意に決定可能であったからである．

ところが，本研究で扱う問題設定では，エージェントは観測空間をタスクに対する意義に基づいて自ら分節化する必要がある．このことが意味するのは，状態分割の必要性の原因，すなわち報酬のばらつきの原因は2通りあるという事実である：(1) 観測空間の切り分けが不十分，(2) 騙しが生じている．(1) の場合には，観測空間上での報酬の分布は観測空間上での位置に依存しており，該当観測レイヤ上で観測空間を分割することで，それぞれの領域における報酬のばらつきを抑えることができる．これを**観測ベース分割 (observation-based segmentation)** と呼ぶ（Fig. 3.9，左側）．一方，騙しが生じているときは，同様の観測値に対して異なる報酬が得られており，観測空間を分割しても問題は解決しない．この場合は，さらに過去のデータを参照するために，木構造を下方向に延長する．これを**履歴ベース分割 (history-based segmentation)** と呼ぶ （同図，右側）．

これらの分割方法の選択の手順を以下に示す．

Fig.3.10には，それぞれの原因によって報酬のばらつきが生じている場合の報酬の観測空間上における分布を模式的に示した．もしも現在の状態表現において，異なる観測値を混同していることが報酬のばらつきである場合には，報酬の分布は図 (a) のように，観測空間上の位置に対する依存性を持っている．これに対して，原因が知覚騙し問題であり，どれだけ詳細に観測値を識別したとしても問題が解決されない場合は，図 (b) に示すように，ほとんど同様の観測値に対しても大きく異なる報酬が得られている．

従って，分割方法の決定においては，これら2つの報酬の分布の仕方を判別するために，以下の手順を踏む：

$\mathcal{I}_a$ に含まれる各インスタンス ι_i に対して，知覚騙し測度 $\mathcal{D}_i$ を以下のように求める：

$$\mathcal{D}_i = \max_{j \in \mathcal{I}_a} d_{ij} \tag{3.3}$$

ここで，d_{ij} は ι_i と ι_j の間の騙しの測度である：

$$d_{ij} = |r_j - r_i| \exp\left(-k \frac{|\boldsymbol{v}_j - \boldsymbol{v}_i|}{\sqrt{D+1}}\right) \tag{3.4}$$

ただし，r_l は ι_l に対応する即時報酬，D は現在の状態に対応するノードの深さである（x_t がレイヤ o_{t-D} にある）．$\boldsymbol{v}_l$ は ι_l に対応する観測ベクトルであり，根ノードのレイヤから x_t のあるレイヤに至る複数の時間ステップにわたる観測ベクトルを用いる（観測空間の次元が d_0 とすると，$d_0 \cdot (D+1)$ 次元ベクトル）．この式が意図しているのは，互いに近い観測値を持ち，報酬の偏差が大きな2つのインスタンスの組に対して大きな値を返す関数である．

こうして得られた $\mathcal{D}_i$ について，$\mathcal{I}_a$ の要素数のうち，$\mathcal{D}_i > \mathcal{D}_\mathrm{T}$（$\mathcal{D}_\mathrm{T}$ は閾値）となるものの占める割合が閾値 ρ を超えるとき，強い騙しが見られると判断して履歴ベース分割を選択し，そうでないとき観測ベース分割を選択する．

以下，それぞれの分割方法を説明する．

観測ベース分割

観測ベース分割は，o_t レイヤから下方向に，順次再帰的に行われる．

まず，現在扱っているレイヤに対応する観測空間上における $\mathcal{I}_a$ の要素インスタンス群に対応する観測値群をユークリッド距離に基づいてクラスタリングする (Fig.3.11 (a))．この際，互いに距離が近く，報酬において大きく異なる，騙し測度の高い事例は省く．

このようにして作られた，互いに近い観測に対応するクラスタの中で，同様の報酬分布をもつもの同士を結合する (同図 (b))．そしてできたクラスタそれぞれの含むインスタンスに対応する観測ベクトルを参照ベクトルとする観測ノードを作る．

次に，インスタンス群 $\mathcal{I}_a$ を新しい観測分割に対応して分割し，それぞれのインスタンス群を用いて，新しい観測ノードの更に下のレイヤの分割を行う．葉ノードに達したら手順を終了する．

この手順をレイヤ o_t から x_t の存在するレイヤまで下方向に再帰的に行う．

履歴ベース分割

履歴ベース分割では，現在のリーフノードの下にさらに動作・観測の 2 レイヤを追加する．

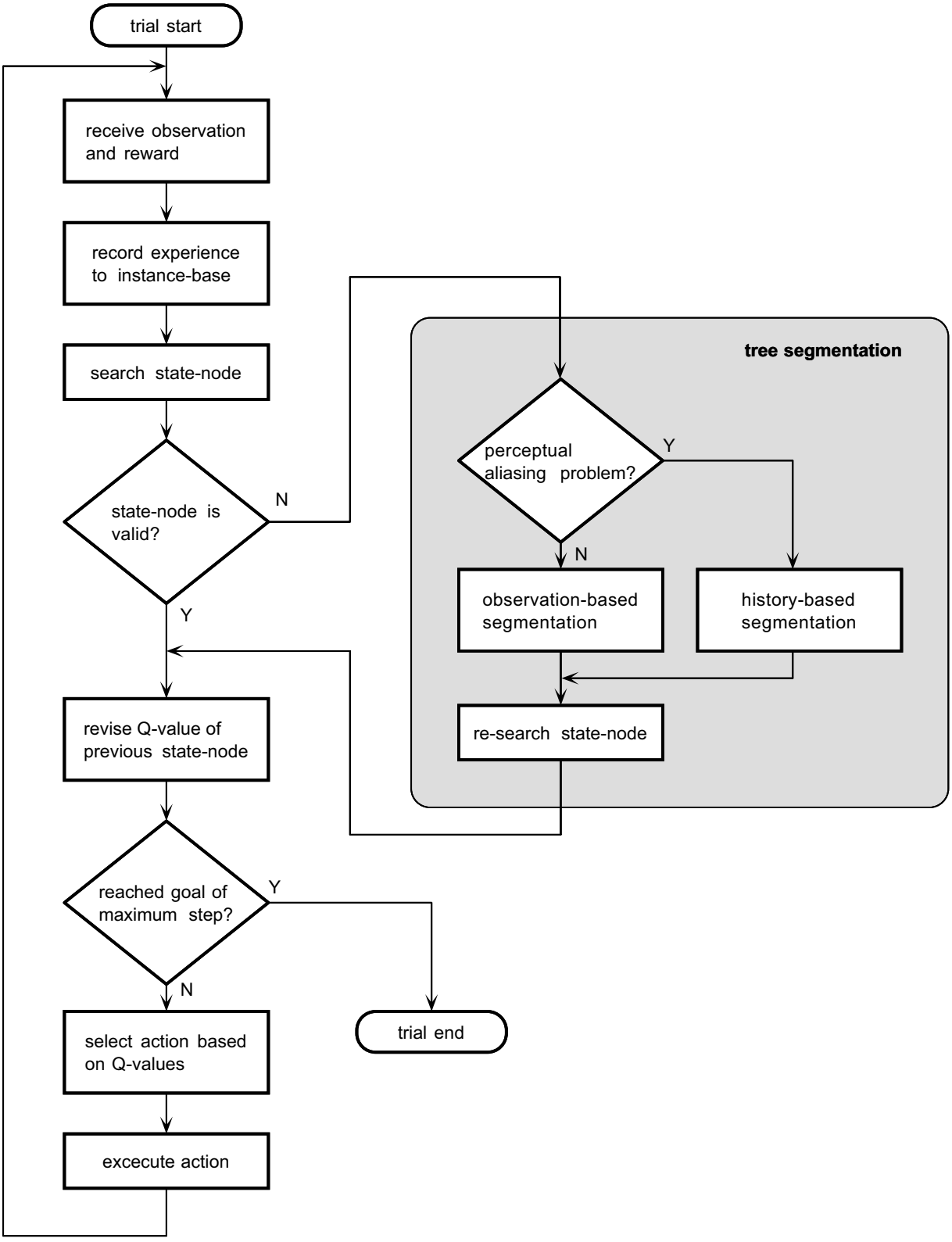


Fig. 3.8: Flowchart of learning procedure

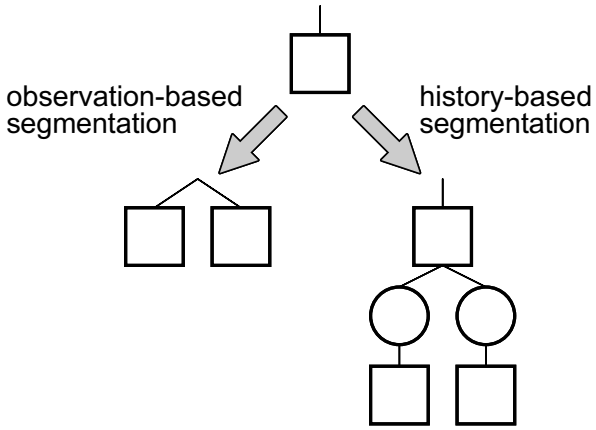
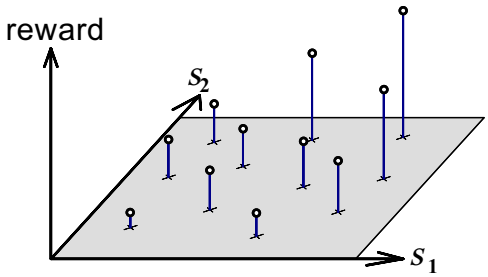
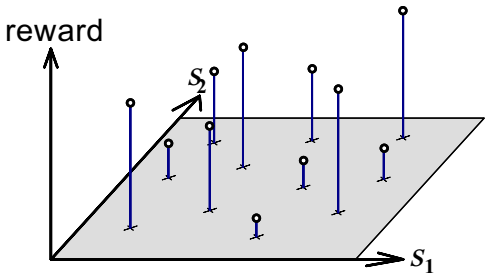


Fig. 3.9: Two alternative methods of segmentation

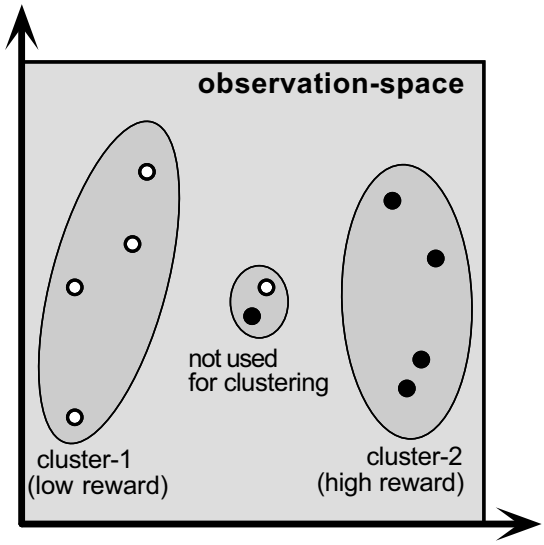


(a) Insufficiency of observation-based segmentation

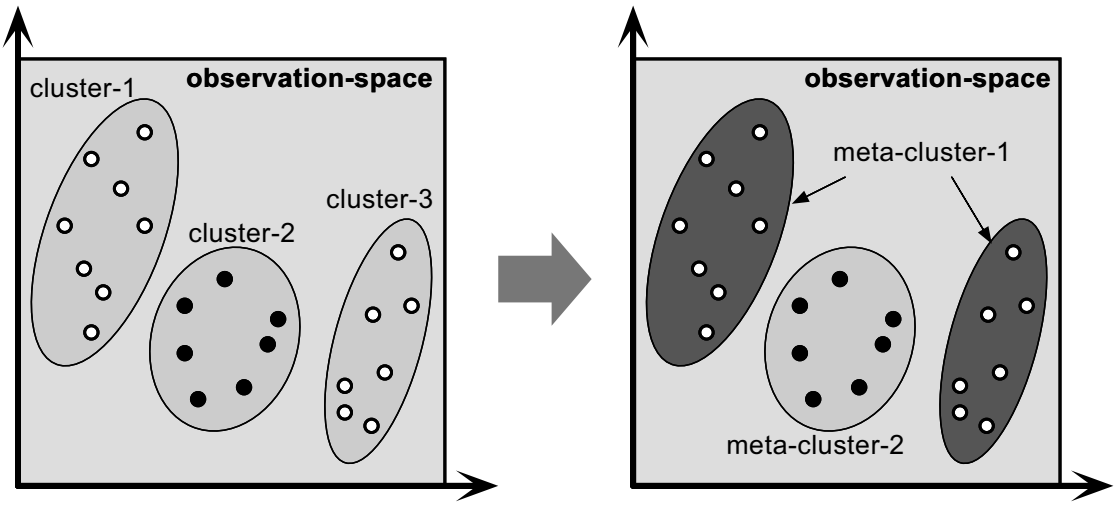


(b) Perceptual aliasing problem

Fig. 3.10: Two alternative methods of segmentation



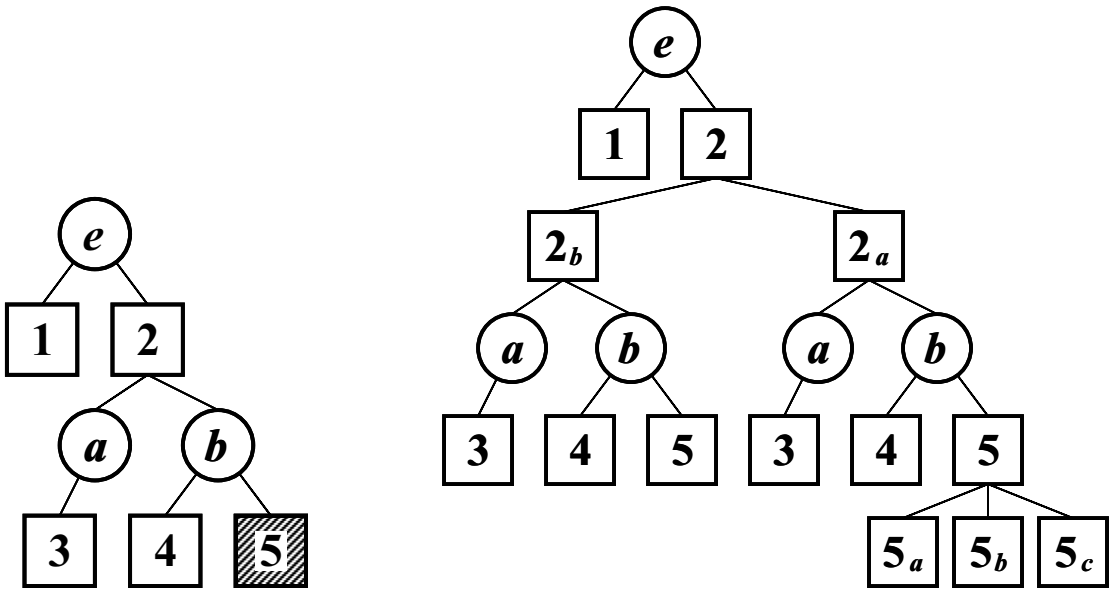
(a) Generation of clusters



(1) Generating clusters (2) Generating meta-clusters

(b) Generation of meta-clusters

Fig. 3.11: Clustering of instances based on observation



(a) Before segmentation (b) After segmentation

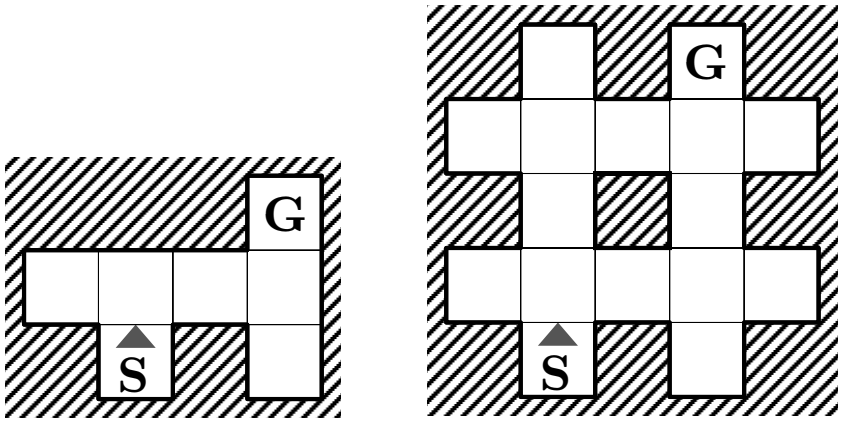
Fig. 3.12: Recursive segmentation

3.5 シミュレーション

単一タスクに対する状況認識・行動決定機構の構成手法の有効性を検証するために，計算機シミュレーションを行った．ただし，ここでのシミュレーションにおいては観測値は離散的に与えられるが，状態識別・状態分割においてはこの観測空間は連続値空間として扱われる．

3.5.1 シミュレーション条件

シミュレーションにおけるエージェントのタスクとして，Fig. 3.13に示す通路状の2次元グリッド環境におけるナビゲーションを扱った．それぞれの環境の特徴をTable3.1に示す．双方の環境とも，あり得る状態の数に対して観測値の数が少なく，知覚騙しが頻繁に発生する環境となっている．



(a) Environment 1　　(b) Environment 2

Fig. 3.13: Simulation environments

Table 3.1: Characteristics of environments

	environment 1	environment 2
possible observations	10	7
possible states	24	60
length of shortest path	6	8

各試行において，エージェントは“S”と表記されたスタート位置・三角形の示す姿勢か

らスタートし，“G” と表記されたゴール位置に到達するか，あるいはその試行で消費したステップ数が所与の制限（ここでは 500 とした）に到達したときその試行を終了する．

エージェントは，前後左右および左右斜め前方に計 8 個の近接覚センサを持つ．前後左右のセンサはそれぞれの方向に障害物が存在するとき 1，存在しないとき 0 を返し，斜め方向のセンサは前方あるいは側方のどちらかに障害物が存在するとき 1 を返すものとする（具体的には，ローザンヌ連邦工科大学（EPFL）で開発された小型移動ロボット Khepera を想定している）．エージェントに可能な動作は，グリッド 1 個分の前進（動作 F とする），左右方向 90 度の回転（動作 L，R）の 3 つとする．ただし，エージェントの前方に障害物が存在するとき動作 F が実行された場合には，エージェントの位置・姿勢は変化しないものとする．即時報酬は，1 ステップの動作の前後におけるそれぞれの状態からのゴール到達に必要な最小ステップ数の増減に -1 を乗じた値を与えた．

シミュレーションにおいて用いた学習パラメータは以下の通りに試行錯誤的に決定した．

Table 3.2: Parameters in simulation

behavior learning related	α	0.1
	$_b$	0.01
state-space construction related	ν	50
	δ	0.5
	ρ	0.5
	$\mathcal{D}_{\mathrm{T}}$	0.5

3.5.2 比較対象とする学習手法

シミュレーションにおける提案手法との比較対象として，以下に示す，USM と類似した学習手法を採用した．

この方法では，あり得る観測ベクトルの集合を予め与えておき，観測に基づく識別においては，観測空間は所与の観測ベクトル群を参照ベクトルとする分割に従って分類される．本シミュレーションではグリッド環境を扱っているため，所与の観測ベクトル群には，前後左右それぞれにおける障害物の有無に基づく $2^4 = 16$ 個の観測ベクトルを用いた．

ただし，ここで用いた対照手法は，離散的で有限の観測空間が所与であるという点を除いては，提案手法と同様である．観測空間の分割の必要性が生じないことから，観測ベース分割と履歴ベース分割の区別はなく，分割の必要性が生じた場合には，所与の観測ベクトル群を用いた定型的な履歴ベース分割が実行される．

3.5.3 結 果

環境1に関するシミュレーション結果をFig.3.14に示す．図中,（a）は各試行において消費したステップの数,（b）は学習の開始から各試行終了時までに消費された総ステップ数,（c）は各試行終了時における状態表現木内のノード数,（d）は各試行終了時における状態表現木の最大深さである．なお，これらのデータは全て，10通りの乱数の種を用いて行った結果を平均したものであり，“proposed” が提案手法，“USM-like” が比較対象の手法を示している．

図中（a）および（b）に示すとおり，双方の手法ともに，試行100程度で最短ステップ数6に収束した．

Fig.3.15は，学習の結果収束した動作シーケンスである．最適行動が実現されている．

Fig.3.16に，提案手法において獲得された状態表現木のうち，比較的ノード数の少なかったものを例として示す．

図中，各ノードの左上の数字はノード番号，観測ノード内のアルファベットはそのノードにおいて収束した動作である．また，観測空間の分割が行われている観測レイヤにおいては，観測ノード内に参照ベクトルを示した．ノード内の小さな四角形はそれぞれ，三角形の方向をエージェントの姿勢としたときの前後左右の障害物の有無を示している．初期状態 s_{start} はノード1に対応し，ここでの収束した動作がFであるため，エージェントはまず前進を行う．ステップ1では，前方に障害物が観測されているため，レイヤ o_t ではノード2がたどられ，前回の動作が “F” であるため，状態ノードは8番となる．ここでは収束した行動が “R” なので，右回転動作が行われる．ステップ2では，現在の観測は左にのみ障害物が存在するというものであり，レイヤ o_t では6 番のノードが選ばれる．1ステップ前の動作が “R” なので次に17番，18番がたどられ，2ステップ前の行動 “F” に対応するノード33を経て状態ノードは34番となり，前進動作が選ばれる．同様にして，各ステップにおける行動方策が対応する状態ノードに記録され，これがゴール達成行動を表現する行動ルールとなっていることが分かる．

Fig.3.17には，異なる乱数の種を用いて同様の条件下でシミュレーションを行った結果得られた状態表現を示す．

双方の状態表現は最適行動を表現するものであるが，乱数の種に応じて状態分割の起こり方に違いが見られる．

次に，環境2に関するシミュレーション結果をFig.3.18に示す．この場合も，消費ステップ数において提案方が優れているという結果になったが，生成されるノード数においては顕著な違いが現れた．

また，シミュレーションにおける，行動の収束までに要した総ステップ数，最終的な木構造内ノード数，および最終的な木構造の最大深さの結果を Table3.3に示す.

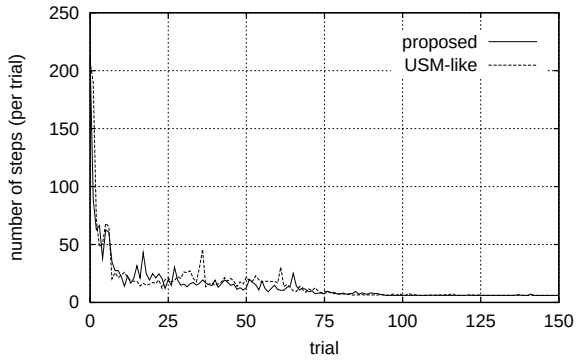
Table 3.3: Comparison of simulation results

environment		proposed	USM-like
environment 1	number of steps	1643.5	2125.5
	number of nodes	81.1	98.9
	maximum depth	2.5	2.0
environment 2	number of steps	12712.8	14511.2
	number of nodes	610.1	2371.1
	maximum depth	6.6	7.3

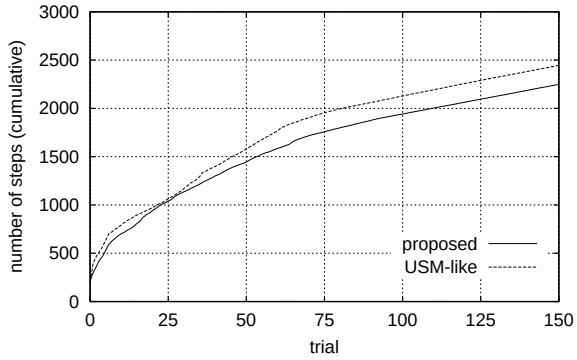
3.5.4 考 察

提案手法により，通路状グリッド環境において，局所的観測値と所与の動作集合のみに基づいてナビゲーション行動が獲得された.

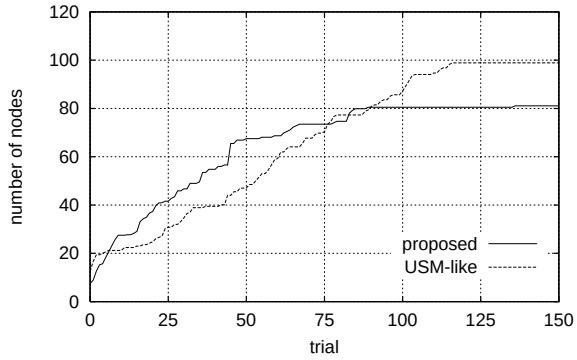
また，可能な観測値の集合を予め与え，これに基づいて観測空間を分割する方法に比較して，提案手法は効率の良い観測空間分割法を与え，これにより状態数が削減されることで，消費メモリ量および学習における行動のコストが低減されていることが示された.



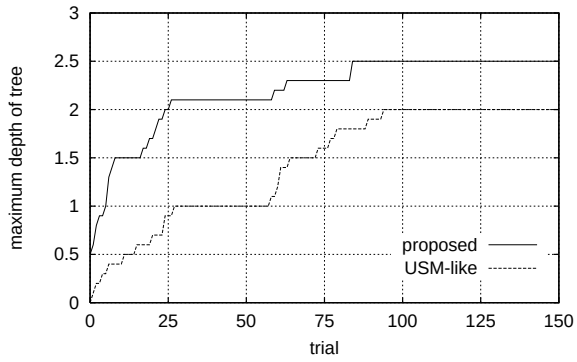
(a) Number of steps (per trial)



(b) Number of steps (cumulative)



(c) Number of nodes



(d) Maximum depth of tree

Fig. 3.14: Results with environment 1

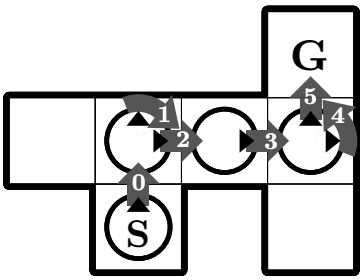


Fig. 3.15: Acquired behavior for environment 1

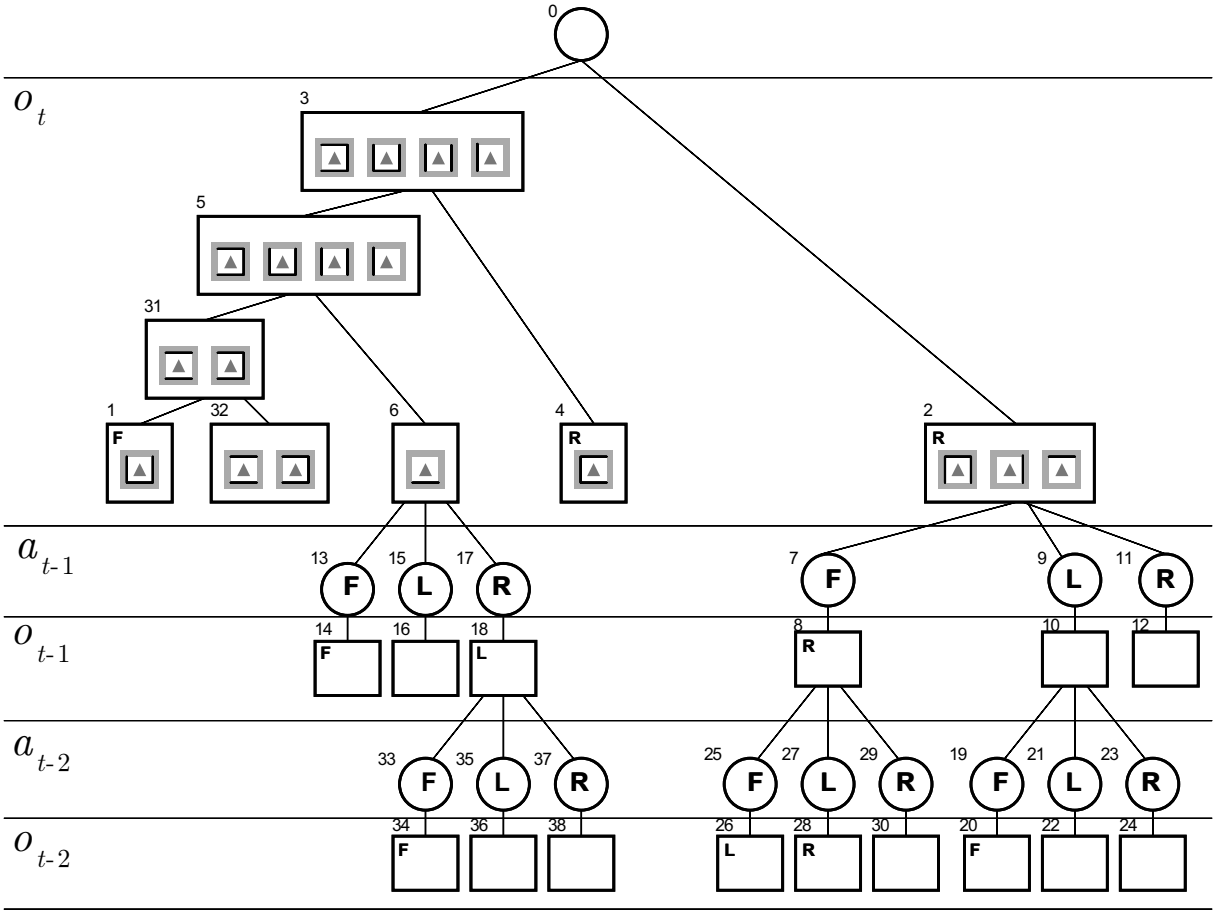


Fig. 3.16: Example 1 of acquired state-representation

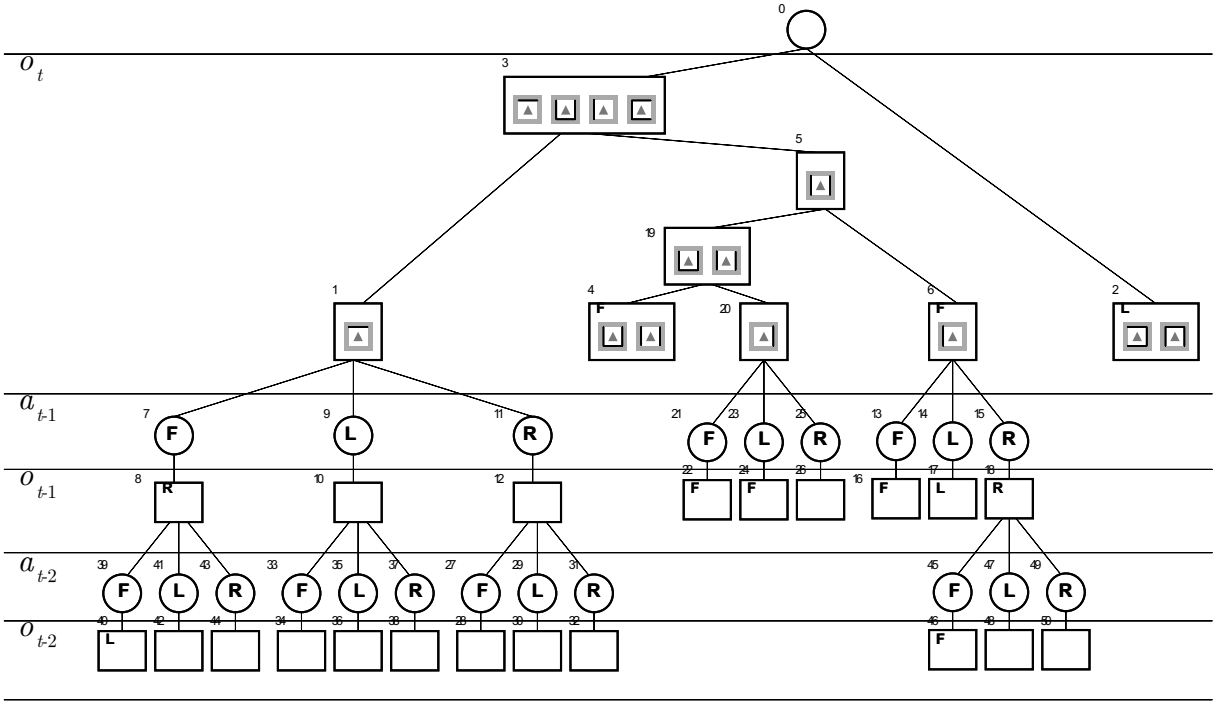
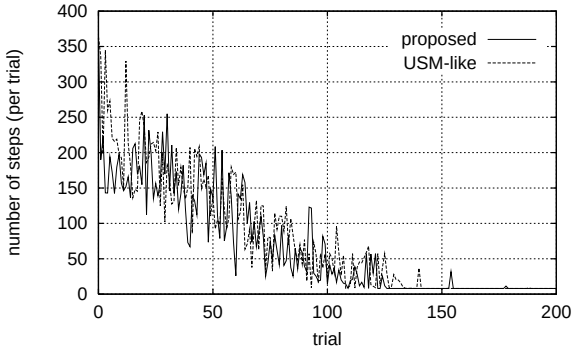
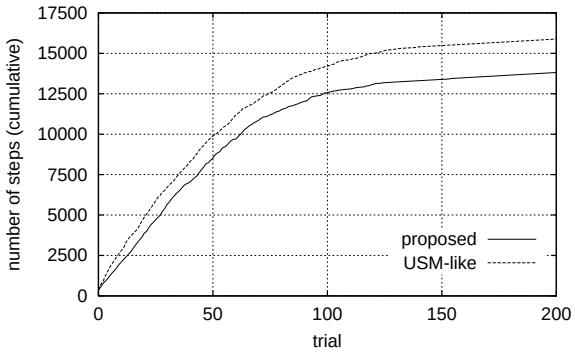


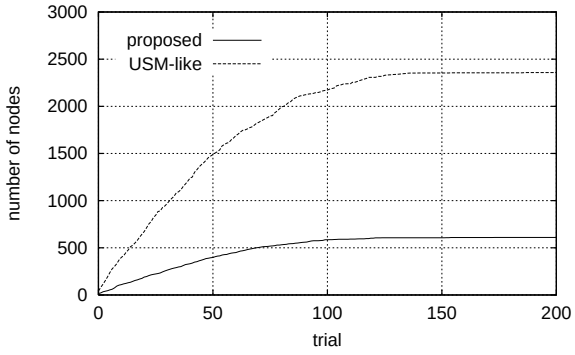
Fig. 3.17: Example 2 of acquired state-representation



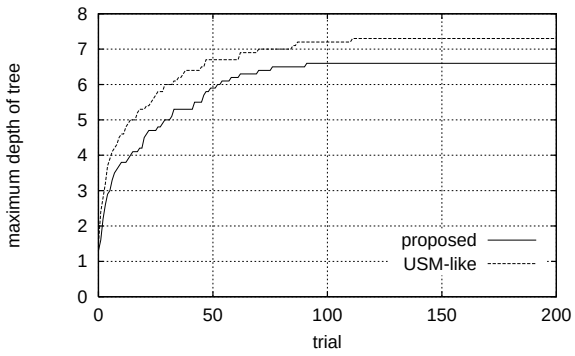
(a) Number of steps (per trial)



(b) Number of steps (cumulative)



(c) Number of nodes



(d) Maximum depth of tree

Fig. 3.18: Results with environment 2

3.6 おわりに

本章では，部分観測環境における，予め分節化されていない連続的観測空間を持つエージェントによる，即時報酬に基づく状況認識・行動決定機構の獲得手法を提案した．

提案手法では，部分観測性に対応するために短期記憶に基づく決定木構造の状態表現を用い，この木構造に適宜分割を加えてゆくことでタスク実現行動を実現するために必要な状態認識・動作選択を実現しうる状態表現を生成する．

ここで，観測空間の分節化が所与でないため，状態分割においては，観測空間の分割を加えるべきか，あるいは知覚騙しへの対処のために，より過去の記憶に基づく識別を加えるべきかの判断が必要となり，提案手法ではこれを観測空間上の報酬の記録の分布についての分析に基づいて判断する．

計算機シミュレーションにより，騙し状態を多く含む通路状の2次元グリッド環境におけるナビゲーション行動の獲得が確認された．

第 4 章

複数タスクへの応用

4.1 はじめに · · · 60
4.2 手法の概要 · · · 61
4.3 タスクの推定 · · · 66
4.4 追加学習 · · · 67
4.5 タスクの追加 · · · 68
4.6 計算機シミュレーション · · · 69
4.6.1 シミュレーション設定 · · · 69
4.6.2 追加訓練の必要性の確認 · · · 70
4.6.3 提案手法の検証 · · · 73
4.7 おわりに · · · 77

4.1 はじめに

本章では，前章で説明した単一タスクに対する状況認識・動作決定機構の獲得手法に基づく，複数タスクに対する行動獲得手法を提案する．

第4.2節では，複数タスクに対する行動獲得手法の全体的概略を説明する．提案手法では，行動決定機構を2段階の階層構造とし，個別のタスクに対する認識・行動決定機構に対してメタレベルに置かれる機構として，タスク識別機構を用いる．また，学習過程も2段階に分け，まず個別タスクに対する知識を獲得した上で，複数タスクの識別と適切な知識適用を行う機構の獲得を行う．

第4.3節では，現在行っているタスクの推定方法について説明する．

第4.4節では，タスク識別に伴って，個別のタスクに関して行われる追加学習について説明する．

第4.5節では，新たにタスクが追加された場合について説明する．

第4.6節では，複数タスクに対する行動獲得を計算機シミュレーションにより検証する．

4.2 手法の概要

本節では，複数タスクへの対応手法について，その概要を説明する．

序論で説明したとおり，本論文では身体性を有するエージェントにとって解決すべき問題として，部分観測環境および観測空間の自律的構成の問題を扱うことにより，タスクにおける個別の状況の識別を，身体性・環境・タスクに依存した形で獲得する方法を対象としている．

複数タスクへの対応においても，この身体性・環境・タスクに依存した認識機構の構築を目的とする．

複数タスクへの対応においては，現在エージェントが扱っているタスクをまず判別し，それぞれのタスクに応じたタスク実現行動を実行する必要がある．このとき，最も望ましい方法は，単一の認識機構によってタスクを識別し，更にそれぞれの識別されたタスク上におけるエージェントの状況をも識別する常用識別機構を獲得することである．

しかし，これを獲得するためには認識機構の学習の過程において，認識機構の不適切の原因として (1) 観測空間の分節化の不十分性, (2) 知覚騙し問題, (3) タスクの誤認の3つから正しいものを選択して対処する必要があり，非現実的な問題となる．

これに対して，この問題に対する現実的な対処としては，個別のタスクに対する認識・行動決定機構の獲得と，こうして獲得された知識を現在行っているタスクに応じて適切に適用する機構の獲得とを異なるタイムスパンにおいて行うという方法が現実的である．

従ってここでは，個別のタスクに対して状況識別・行動決定機構をそれぞれ獲得する学習過程をまず最初に実行し，この学習過程でタスクごとに得られた経験に基づくタスク識別を行うメタレベルの機構を用いるという階層的構造を採用する (Fig.4.1)．

これを実現するために，エージェントは2段階のプロセスによって行動を獲得する．まず第1の段階（**知識獲得段階**）としてそれぞれのタスク設定に対して知識を獲得する．第2の段階（**知識適用段階**）においては，それぞれの個別のタスクに対して**確信度 (credit)** を割り当て，これを現在得られている観測値・行った動作に基づいて更新することで各タスク設定が現在扱われている蓋然性を表現し，最も確信度の高い知識を利用することで，適切な行動を実現する．

確信度は，知識適用段階での試行において，初期状態から現在の状態に至るまでにエージェントが行ったのと同一の動作シーケンスを実行した事例を，各タスクに対して知識獲得段階で得られているインスタンスベースの中から抽出し，知識獲得段階において経験された観測と現在得ている観測との偏差を評価することによって更新する．同一の動作シーケンスに対して知識獲得段階において得られた観測値が現在得ているものと近いときは対応する設定に対する確信度を増加させ，そうでないときは減少させる．

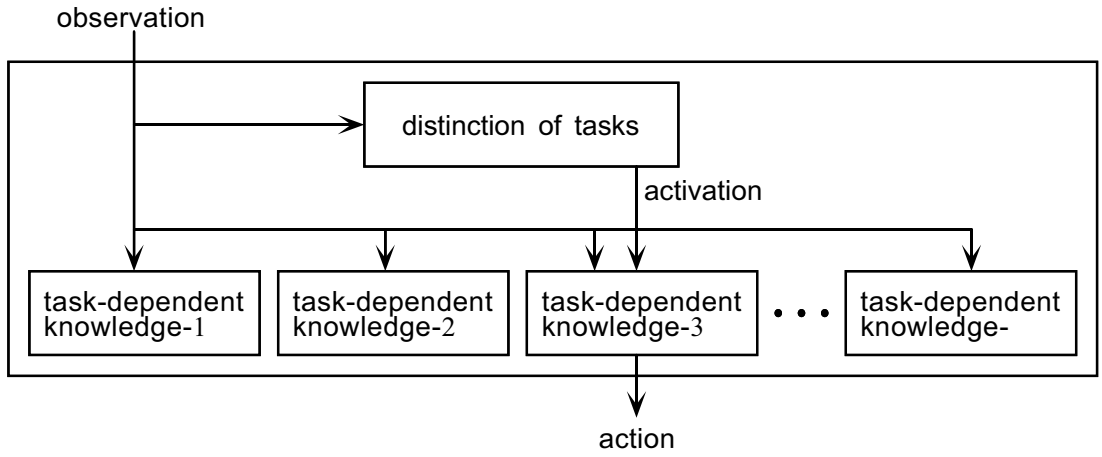


Fig. 4.1: Schematic view of proposed system

この更新の結果，確信度が最も高い設定に対応する状態表現・行動方策に基づいて行動することでその設定におけるタスク実行に対して適切な行動が実現される．

ところが，ここで以下の 2 つの問題により，適切な行動の実現が不可能となる場合がある：

(a) **手がかりの喪失**

知識獲得段階において獲得される経験の多様性は，対応するタスクにおけるタスク達成行動の獲得に至るまでに探索的行動をどれだけ行ったかに依存しており，知識適用段階における動作シーケンスと同様のシーケンスを経験しているという保証はない．このため，知識適用段階において，ある時点までに行った動作シーケンスと同様のシーケンスが一つ以上のインスタンスベースにおいて一つも存在しない場合があり，このとき類似度の比較が不可能となるため，確信度の更新が不可能となってしまう．

この問題に対する最も単純な対処法として，知識獲得段階において獲得する経験を多様化するために，学習における探索的行動を促進するようなパラメータを採用する方法があるが，動作シーケンス長の増加に対して，要求される動作シーケンス数は幾何級数的に増大するために，この方法は現実的ではない．この点については 5.2 節でシミュレーションを行った結果により具体的に示す．

(b) **状態表現・行動方策の不適切性**

知識適用段階において，初期状態から終端状態に至る動作系列上で得られる観測が，数ステップにわたって同様であるとき，観測に偏差が見られるまでの間，確信度に差が現れない．ところが，このとき知識獲得段階で得られた行動方策が，それぞれのタスク設定に対して異なっている場合，エージェントは一つの動作系列を選択せねばならないため，そ

の動作系列が最適行動と異なるタスク設定に関しては，獲得されたものではない行動を行われることになる．ところが，ここで選択されていたタスク設定が実際には間違っており，ある時点で別のタスク設定に対応する行動方策が改めて利用されたとき，その時点で置かれている状態からタスクを遂行するための行動が，対応する行動方策に記述されているという保証がなく，適切な行動が不可能となる．

上記の2点の問題に対応するため，提案手法では，上記のいずれかの問題が起こった，あるいは起こりうることを検出した場合，それぞれに対応する追加学習を行うという方法を採る．

追加学習においては，初期状態から上記の問題が発生するまでの動作シーケンスを行った時点からのタスク達成行動を獲得する．

システム全体の概略をFig.4.2に示す．

システムは，知識統合部と個別の学習データとからなる．学習データとは，知識獲得段階において各タスク設定に対して得られた状態表現・行動方策およびインスタンスベースである．知識統合部は知識適用段階での短期記憶を格納するインスタンスベース，類似度評価器および各タスク設定に対応する確信度からなる．知識適用段階における各ステップにおいて得られた観測データは，知識統合部のインスタンスベースに短期記憶として獲得される．

類似度評価器においては，この短期記憶における動作シーケンスと同一のシーケンスが各学習データ内のインスタンスベースから抽出され，それらの観測値と現在得られている観測値との距離に基づいてタスク設定iに対応する確信度c_iが更新される．行動を行うために用いられる状態表現・行動方策はこの確信度に基づいて選択され，選択された方策に従って動作が環境に出力される．

複数タスクに対する行動獲得過程における流れをFig.4.3に示す．

以下，類似度評価器において行われる現在のタスク設定の推定手順および追加学習について詳細を説明する．

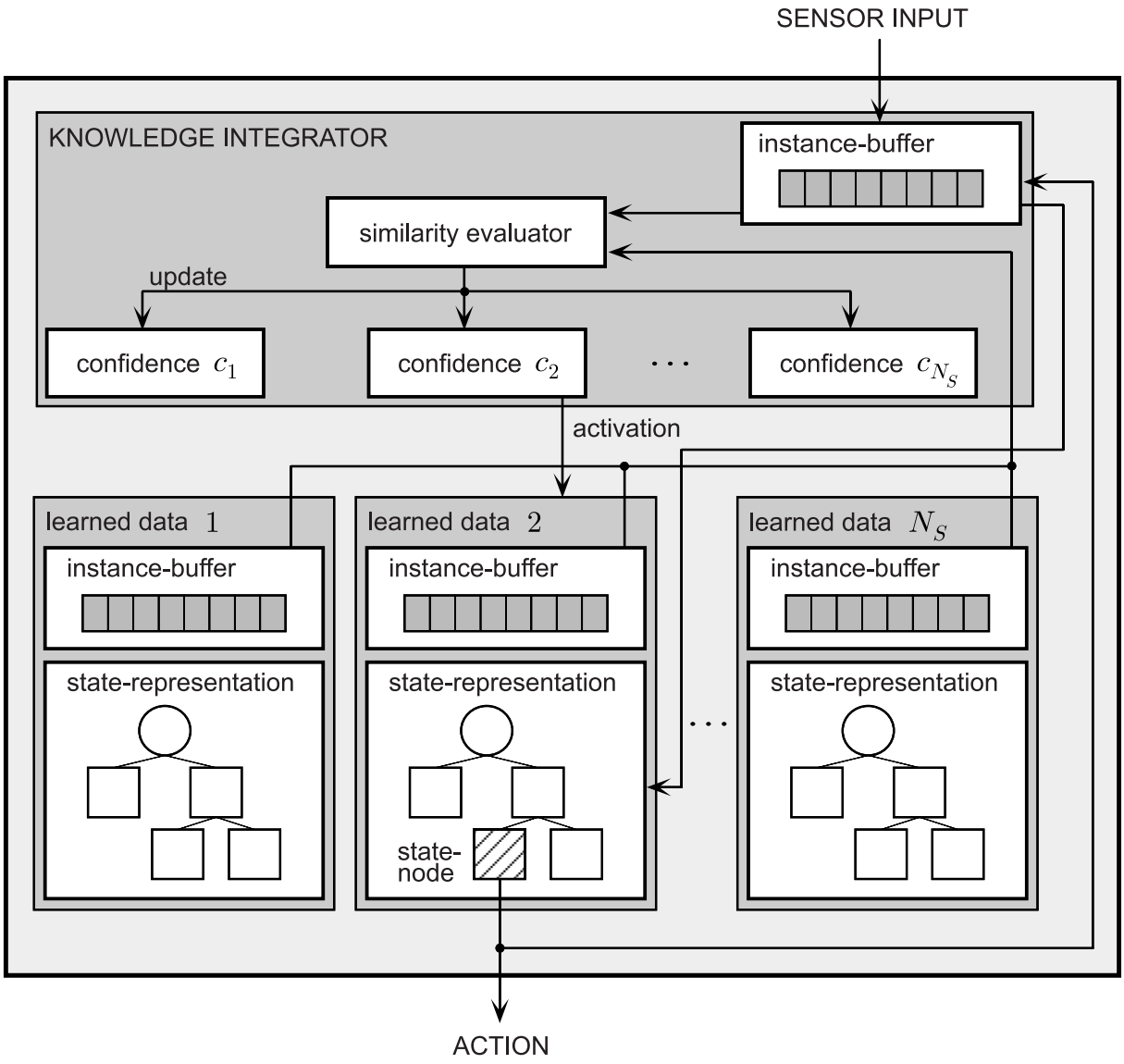


Fig. 4.2: System Architecture

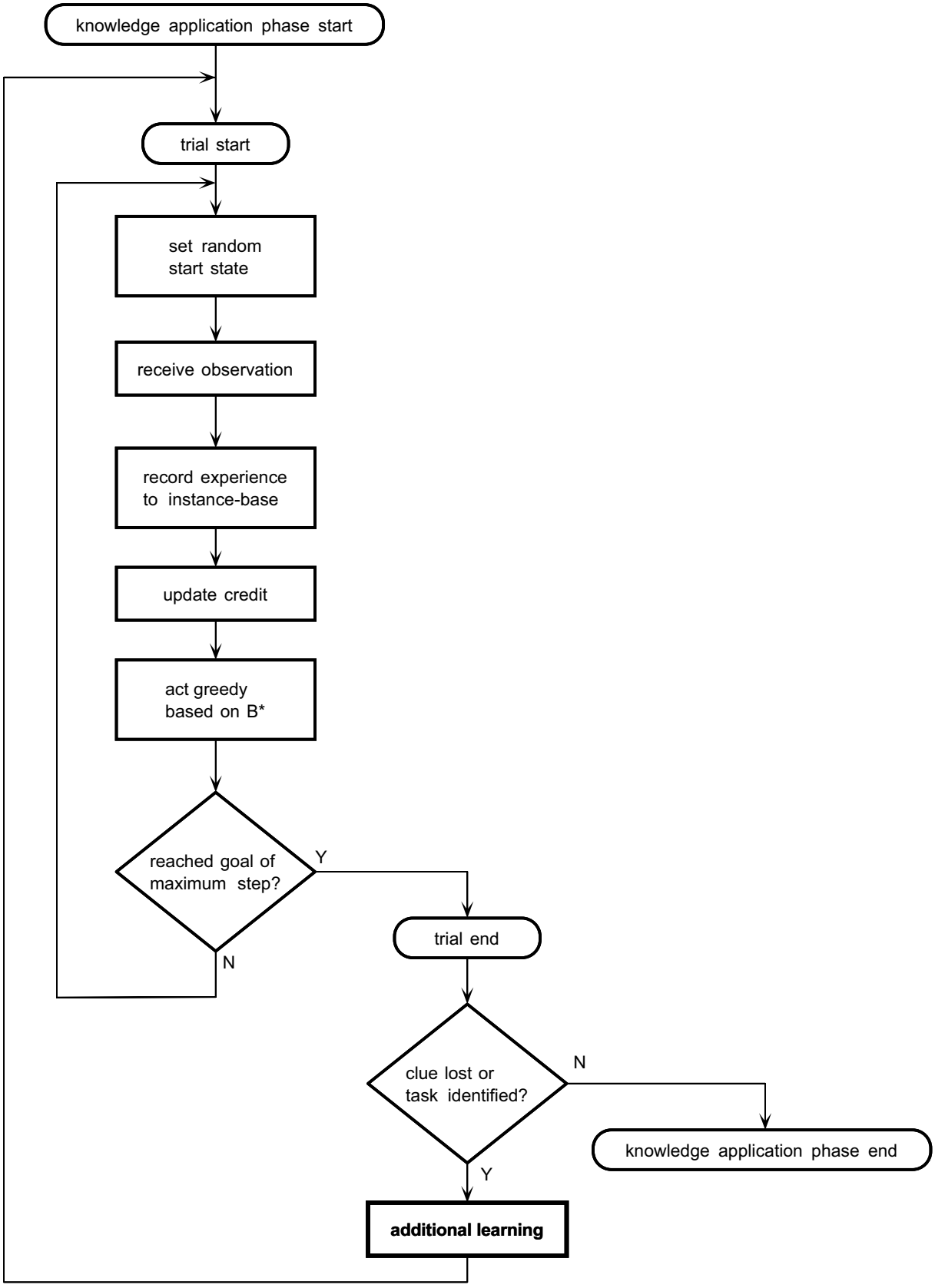


Fig. 4.3: Flowchart of knowledge application phase

4.3 タスクの推定

エージェントは各時点において，今回の試行がそれぞれのタスク設定であることの蓋然性を示す確信度の更新を行う．タスク設定 i に対する確信度を c_i とする．

確信度の更新手順は以下の通りである：

1. **事例の抽出**：知識獲得過程においてそれぞれのタスク設定に対して得られているインスタンスベースから，今回の試行においてこれまで行った動作シーケンスと同一のシーケンスを行った時の事例を抽出する（現在，ステップ 0 である場合，即ち今回の試行でまだ動作を行っていない場合は全ての事例が抽出される）．

2. **観測の類似度の評価**：抽出された各事例について，今回の試行におけるステップに対応するステップで得られた観測値と，現在得られている観測値との距離を Euclid 距離として計算し，さらに各タスク設定に対応する全ての事例における距離の値の平均値をそれぞれのタスク設定に対して求める．

3. **類似度に基づく確信度の更新**：観測値間の距離が大きいタスク設定ほど，今回の試行と異なっている可能性が高いことから，距離の大きいタスク設定に対応する確信度が低下するよう，以下の式に基づいてタスク i に対応する確信度 c_i の更新を行う．

$$c_i \leftarrow c_i + \frac{\bar{d} - d_i}{\bar{d}} \tag{4.1}$$

   ただし，d_i はタスク設定 i に対応する観測値距離の平均値，$\bar{d}$ は全てのタスク設定に対する観測値距離の平均値である．

更新された確信度は，行動出力のために利用すべき状態表現・行動方策を選択するために用いられる．すなわち，最も高い確信度に対応する状態表現に即して行動が決定され，実行される．

ただし，最大の確信度を与えるタスク設定が複数存在する場合，それらのうち，どれが現在の試行において対応しているかが未だ明らかになっていないことを意味している．この場合，それらの設定のうち，どれに対応する方策を適用するかについては，本研究では最も単純な方法として，最も番号の若いものを選択するという方法を採る．

4.4 追加学習

本節では，第4.2節において述べた2つの問題に対する対応策である，追加学習について述べる．

前述の通り，確信度の更新における事例の抽出において，現在の動作シーケンスと同一のシーケンスが特定のスタート設定に関して一つも得られない場合，確信度の更新が不可能となる．また，スタート設定が一つに特定された時点では，既に初期状態とは異なる状態にいる可能性があり，この状態からタスクを達成する行動方策は得られている保証がない．

従って，以下のタイミングで追加訓練を行い，経験の補充あるいは適切な行動の獲得を行う：

(a) **手がかりの喪失に対して**：ある時点で，最大の確信度を与えるスタート設定が複数存在し，それらのうちの少なくとも一つに対して，抽出できる事例が一つも存在しなかった場合．

(b) **状態表現・行動方策の不適切性に対して**：特定のタスク設定に対して，初めてそのタスク設定が一意に特定された場合．

(a)のケースについては，初めて手がかりを失うまでの動作シーケンスからタスク達成に至るまでの行動獲得を，手がかりを失ったタスク設定に対して行わせることで，経験の補充を行う．(b)のケースについては，初めて特定されたタスク設定に対して，特定が達成されるまでの動作シーケンスから開始してタスク達成に至る行動の獲得を行う．

追加学習の手順は，対象となる動作シーケンスを用いて，対象となるタスク設定に対して以下の通り行う：知識獲得段階において用いた3章で説明した手法を対象タスク設定に対して実行する．ただし，各試行における，開始から対象動作シーケンス長分のステップにおいては，エージェントの行動選択は獲得された行動方策ではなく，対象動作シーケンスと同様の動作を強制的に選択させる．行動獲得の過程で得られた経験は，動作獲得段階で記録されたインスタンスバッファに続けて記録し，追加学習過程全体にわたって，状態表現および行動方策（Q値）の更新を実行する．

4.5 タスクの追加

ある時点で，それまで扱っていなかったタスクの追加の必要が生じた場合には，前節（b）に対応する，各タスクに対して特定が完了したか否かの情報を初期化した上で，同様の手順を踏むことで，それまで獲得された行動決定機構を拡張することが可能である．

4.6 計算機シミュレーション

以下の2種類の計算機シミュレーションを行った：

(1) **追加訓練の必要性の確認**：知識獲得段階における探索的行動を，学習パラメータにより促進するという方法で経験の多様化を目指す方法の不適切性を具体的に検証し，追加訓練が必要であることを示す.

(2) **提案手法の検証**：提案手法により，実際に複数のタスク設定に対応できる行動決定機構が構築できることを示す.

4.6.1 シミュレーション設定

シミュレーションでは，Fig.4.4に示す通路上2次元グリッド環境におけるナビゲーションタスクを扱った．図中，“G” はゴール状態を示し，“S_i” および小さな三角形はタスク設定 i に対応するスタート位置・姿勢を示している．各初期状態近傍で得られる観測値は各スタートに関して同様であることから，エージェントが現在の試行でどのスタートから出発したのかを識別するためには，一定の長さの動作シーケンスを実行しなければならない．エージェントは各試行において，特定の S_i に置かれ，そこからゴール状態に到達するか，所与のステップ数制限（ここでは250）に達したとき1回の試行を終了する．この環境は，ゴール状態を省く状態数が20であるのに対して，あり得る観測値が7通りであり，多くの騙し状態を含むものである.

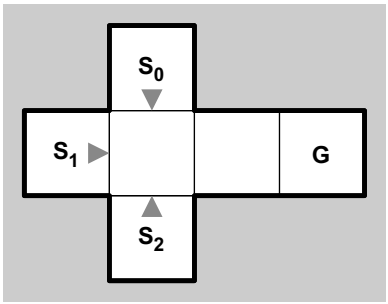


Fig. 4.4: Simulation Environment

エージェントとしてはローザンヌ連邦工科大学（EPFL）で開発された小型移動ロボットKheperaを想定しており，前後左右及び左右斜め前方に合計8つの近接覚センサを持つ．前後左右のセンサはそれぞれの方向に障害物が存在するとき1，存在しないとき0を返し，斜め方向のセンサは前方あるいは側方のどちらかに障害物が存在するとき1を返すものとする．従って観測空間は8次元空間となるが，エージェントはこの空間をタスクに応じて

超平面により分割する．エージェントに可能な動作は，グリッド 1 個分の前進（動作 F とする），左右方向 90 度の回転（動作 L，R）の 3 つとする．ただし，エージェントの前方に壁面が存在するとき動作 F が実行された場合，エージェントの位置・姿勢は変化しないものとする．即時報酬は，1 ステップの動作の前後におけるそれぞれの状態からのゴール到達に必要な最小ステップ数の増減に -1 を乗じた値を与えた．

シミュレーションにおいて用いた学習パラメータは以下の通りに試行錯誤的に設定した．

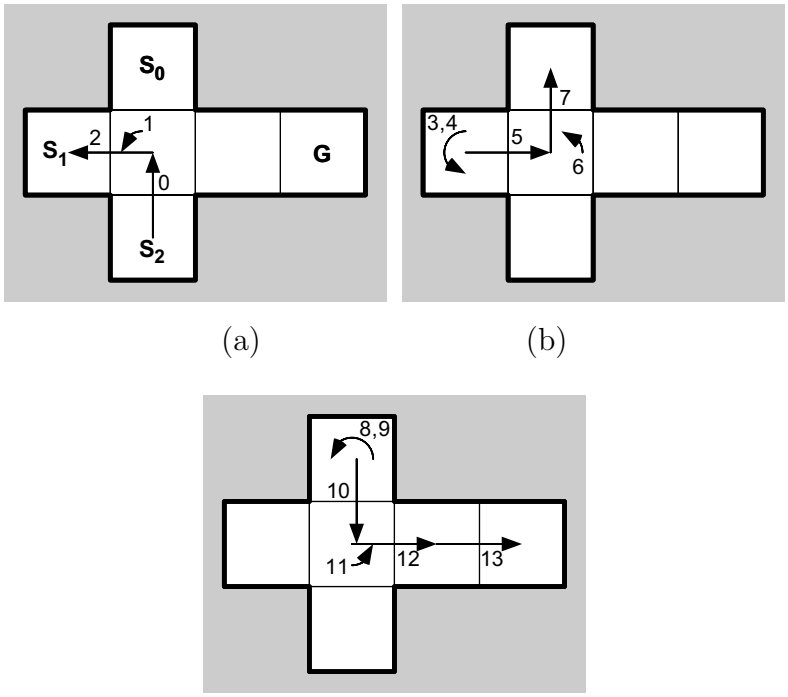
行動学習関連：$\alpha = 0.1$，$T_b = 0.1$（ただし，次節のシミュレーションでは T_b を幾通りかに変化させた）

状態構成関連：$\nu = 50$，$\delta = 0.2$，$\rho = 0.2$，$\mathcal{D}_T = 0.5$

4.2 節で示したとおり，未だ確信度に差が現れていない段階では，最も番号の若いタスク設定に対応する状態表現・行動方策が利用される．従って，知識適用段階における，タスク 2 に対応するスタート位置・姿勢から出発した試行においては，以下の動作シーケンスが望ましいものとなる：最初 0 番のタスク設定に対応する行動が適用され，エージェントは “F,L, F” の動作シーケンスを行う（Fig.4.5(a)）．この時点で，前方に壁面が現れるためタスク設定 0 に対応する確信度が減少し，次に 1 番のタスク設定に対応する行動方策が適用される．ここからは動作 L（または R）を 2 回行って 180 度の回転を行った後，動作 F,L,F で十字路を再び左折して前進するという動作が行われる（同図 (b)）．この時点で再び前方に壁面を見るため，現在のタスク設定が 2 番であることが分かり，再び振り返って左折してゴールに向かう（同図 (c)）．

4.6.2 追加訓練の必要性の確認

知識適用段階における現在扱っているタスク設定の推定において，推定に利用可能な事例が知識獲得段階において得られていないとき，推定が不可能となる．従って，学習の過程で多くの事例を蓄積しておくための方法が必要となる．この際採りうる方策として最も単純なものは，知識獲得段階での学習において，学習パラメータによって行動のランダムネスを増加させ，行動の収束までに行われる探索的行動を増加させるというものである．提案手法では，知識獲得段階において，Q 値に基づく行動決定における行動のランダムネスを与えるパラメータは Boltzmann 温度（T_b）であり，これを増大させることによって得られる経験を多様化することができる．これにより，獲得される認識機構の含む情報量をも増加させることができ，タスク識別のための行動により，タスクごとの最適動作シーケンスから逸脱した場合にも，ここから復帰する行動を状態表現の中に内包することができれば，タスク識別の結果この認識・行動決定機構が起動された場合，そのままタスク実現行動を実現することができる (Fig.4.6)．



(a) (b)

(c)

Fig. 4.5: Desired path for task 2

従って，ここではいくつかのT_bに対して知識獲得段階で経験を獲得したのちに，知識適用段階においてタスク設定2に対する上記の所望の動作シーケンスを行う上で，タスク設定の識別に必要な事例が喪失されるまでのステップ数がどのように変化するかを評価する．

また，行動のランダムネスを増加させることは，状態空間上の探索を促進することにより準最適解へのトラップを防止する効果を持つ反面，行動の収束までに要する時間を増大させ，学習コストを増大させるという負の効果を持っている．本シミュレーションでは，上記ステップ数と同時に学習コストを評価する．

Fig.4.7には，タスク2に対する知識適用段階の実行における結果を示す．ここでは，10通りの乱数の種に対して得られた結果の平均値を示した．

図中，“performance of distinction” と示したのは，タスク2に対応するインスタンスベースから事例が喪失されるまでのステップ数であり，T_bの増加に伴って増加している．ところが，上記の動作シーケンスにおいて，タスク2に対応する初期状態から出発してタスク設定1と2とを識別可能となるのはステップ8の時点であることから，T_bが2.0までの間では事例が不十分であり，ステップ8以前に識別の手がかりが失われてしまうことが分かる．

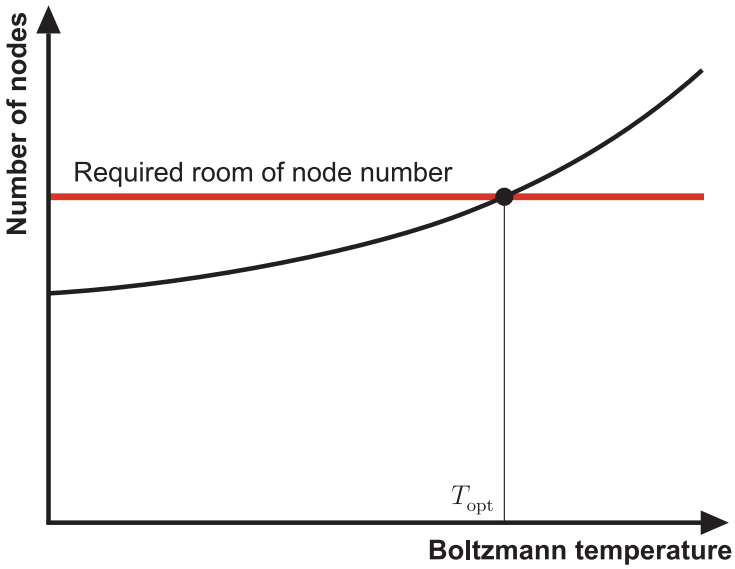


Fig. 4.6: Required room of nodes

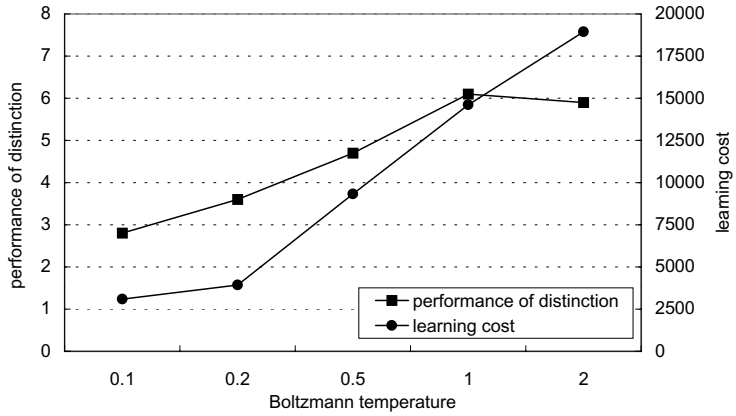


Fig. 4.7: Result for different Boltzmann temperatures

これに対して，図中 “learning cost” と示したのは，知識獲得段階での学習において，所定の試行回数（250 試行）を行う間に消費された総動作ステップ数である．部分観測環境下で観測空間の構成を自律的行うという問題の性質から，行動のランダムネスにより不適切な行動を選択することによる学習のパフォーマンス低下が大きいことが示されている．

以上の結果は，行動のランダムネスを増大させることによって，タスク設定識別において手がかりとなる事例を増加させるというアプローチが不適切であり，タスクの識別に要求される動作系列に応じた経験を恣意的に獲得する必要があることを示している．

従って提案手法では，必要な事例数が不十分であることを検出するつど，必要な経験を追加学習という形で適切に補充する．

4.6.3 提案手法の検証

提案手法により実際に複数のタスク設定に対して対応可能な行動決定機構が獲得されることを示す．

シミュレーションにおける行動獲得過程の一つの例を順を追って示す．なお，知識適用過程における，各試行で適用するタスク設定はランダムに与えた：

1. 知識獲得過程において，各タスクを実現する行動を表現する状態表現・行動方策，および学習過程で得た経験のインスタンスベースを獲得する．

2. 知識獲得過程の第1試行として，タスク0から出発する．この時点では他のタスクと同一の観測が得られているため，確信度は同一であり，従ってタスク0に対応する行動方策が採用され，これに従ってF,Lの動作シーケンスが実行される．この時点で設定1に関して，同一の動作シーケンスを行った経験が存在しないため，これ以降の確信度の更新が不可能となるが，採用されている状況認識機構・行動方策が現在のスタート，0番のものであるため，そのまま動作F,Fを実行してゴールへ到達する．この後，動作シーケンスF,Lに関して設定0,1,2全てに対して追加学習が行われる．タスク識別の手がかりを失う時点までのエージェントの経路をFig.4.8(a)に示す．

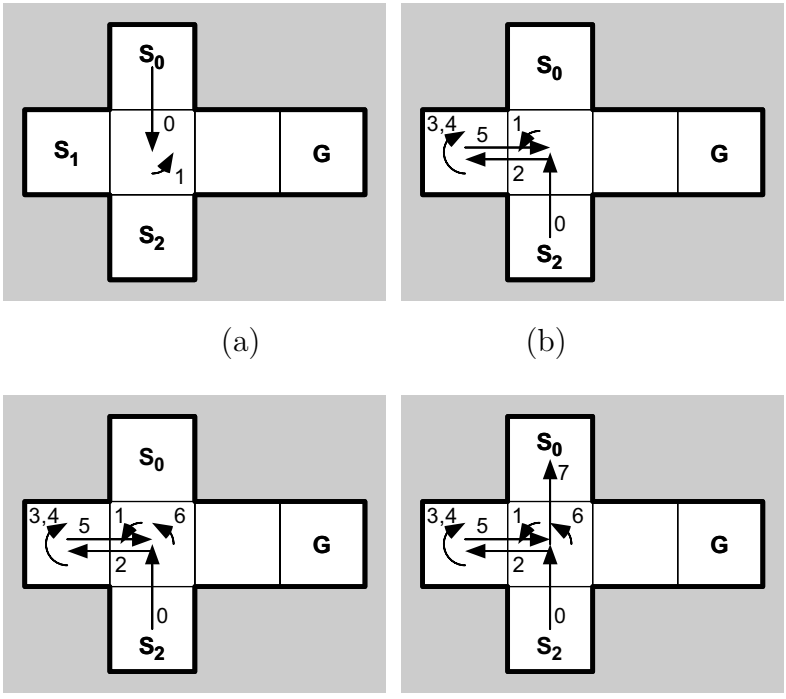
3. 第2試行ではタスク設定2が適用され，同様に0番の行動原理に従って動作するが，動作F,L,Fが行われた時点で前方に壁面が存在するために，設定0に対応する確信度が低下し，これ以降設定1に対応する行動方策が適用される．この方策に従って，動作R,R,Fを行って振り返って十字路へ至るが，この時点でタスク1に対応する事例がなくなり，タスク1と2との識別が不可能となる..従って，今回の試行ではシーケンスF,L,F,R,R,Fを用いた追加学習がスタート1，2に対して行われる．同様に，手がかりを失うまでの経路を同図(b)に示す．

4. 第3試行においては再びタスク2が適用され，前回と同様F,L,Fにより利用される行動規則が1番のタスクに対応するものにシフトし，続いてR,R,F,Lが方策1に従って実行されるが，この時点で設定2に対応する事例がなくなり，シーケンスF,L,F,R,R,F,Lに対して設定1，2について追加学習を行う．同様に，手がかりを失うまでの経路を同図(c)にしめす．

5. 第 4 試行でも再び設定 2 が適用されるが，今回は動作シーケンス F,L,F,R,R,F,L,F が行われた時点で再び前方に壁面が存在したため，この時点で設定 1 ではないことが分かり，現在の設定が 2 番であることが特定される．従って以降方策 2 が適用されるが，設定が特定されるまでの動作シーケンスを実行後にゴールに到達する行動が獲得されていないため，不適切な行動が行われる．また，同一の動作シーケンスにより設定 1 の場合も特定可能であることが分かったため，この試行に対してはシーケンス F,L,F,R,R,F,L,F を用いた追加学習が行われ，適切な行動を獲得する．ただし，この追加学習において設定 2 に関しては準最適解に収束した．タスク 1 と 2 が識別される時点までの経路を同図 (d) に示す．

6. 第 5 試行ではスタート 1 が選択され，既に獲得された動作シーケンスに従って最適経路ゴールに到達する．

7. 第 6 試行でスタート 0 が選択され，動作シーケンス F,L,F によって設定 0 が初めて一意に特定される．この結果スタート 0 に対して追加学習が行われ，全ての行動獲得が完了する．



(a) (b)

(c) (d)

Fig. 4.8: Paths of the agent

以上と同様の過程により最終的に，各タスク設定を識別することにより適切に利用する

方策を切り替えながら準最適経路でゴール到達を行う行動が，他の乱数系列を用いた例においても獲得され，提案手法によって複数タスクへ対応可能な動作決定機構が獲得されていることが示された．

Table4.1には，いくつかのT_bによる知識獲得段階の適用により追加学習を行わずに知識適用段階を実行した場合と，提案手法とにより，10通りの乱数系列に対してタスク設定2において得られたパフォーマンスの比較を示す．“success ratio” には所定のステップ数（250ステップとした）以内に終端状態に到達できた率，“number of steps” には1試行あたりの平均消費ステップ数を示した．表に示すとおり，追加学習を行わない場合についてもある確率でタスクを達成することができるが，エージェントが現在タスク設定2を扱っているということを特定できた例は1例もなく，全てのケースでタスク設定1に対して獲得された認識機構・行動方策を用いた準最適行動が用いられている．これに対して提案手法では適切な追加学習により，全ての場合においてタスク設定2を特定した上でタスクを達成した．

Table 4.1: Comparison of task achievement performance

	$T_b = 0.1$	0.2	0.5	1.0	2.0	proposed
success ratio	0.7	0.2	0.4	0.6	0.4	1.0
number of steps	96.9	202.6	158.4	109.2	156.3	16

Fig.4.9には，前節で行ったシミュレーションとの比較として，追加学習を行った場合と行わなかった場合の，事例数の比較を示す．“step” と示したのは，タスク設定2に対する知識適用段階における試行でのステップであり，“ratio of matching instance” と示したのは，各時点までに行った動作シーケンスに適合する事例数を，それまでに行った学習の試行数で割った比率である．これにより，追加学習によって必要となる事例が適切に獲得されていることが示されている．

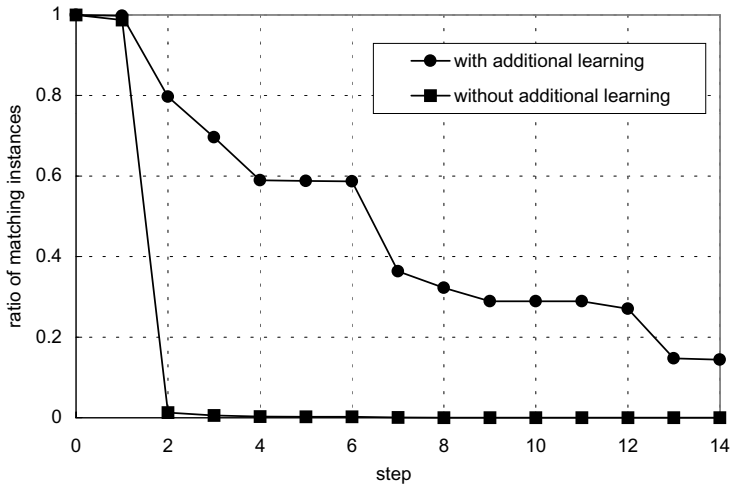


Fig. 4.9: Reduction of matching instance

4.7 おわりに

本章では，事前設計のなされていない連続的観測空間を持つエージェントが，部分観測環境において，複数のタスク設定に対して，状態識別を可能とする状態表現および行動方策を獲得し，これをタスクに応じて適切に適用するための知識の統合手法を提案した．提案手法では，個別の知識の獲得過程において得られた，エージェント自身の視点に基づく経験データを利用して，エージェントが現在置かれているタスク設定を推定し，適切な知識を適用する．また，タスク設定の判別に必要な経験が不足している場合，およびタスクの不確定性に起因して特定のタスクに対して既に得られた知識が不十分となった場合に対応するために，適切な追加学習を行う．

計算機シミュレーションにより，単純な通路上グリッド環境において，異なる初期状態から終端状態へ到達する複数のナビゲーションタスクに対して，各試行におけるスタート点を適切に特定し，適切な行動方策を選択してタスクを達成する行動決定機構が実現されていることを示した．

第 5 章

実環境への適用

5.1　はじめに · 81

5.2　シミュレーション及び実験の目的 · · · · · · · · · · · · · · · · · · · 82

5.3　シミュレーションおよび実験の概要 · · · · · · · · · · · · · · · · · · 84

5.3.1　実機ロボット Khepera · 84

5.3.2　行動規範型動作プリミティブ · · · · · · · · · · · · · · · · · 87

5.3.3　誤差への対応 · 91

5.4　実環境における複数タスク実現に関するシミュレーション · · · · 92

5.4.1　シミュレーション・実験の環境 · · · · · · · · · · · · · · · · 92

5.4.2　シミュレーションの設定 · 92

5.4.3　シミュレーションの結果 · 93

5.4.4　考　察 · 93

5.5　実機複数タスク実現に関する実験 · · · · · · · · · · · · · · · · · · 99

5.5.1　実験結果 · 99

5.5.2　考　察 · 100

5.6　異なる視点に基づく報酬付与への対応に関するシミュレーション · 105

5.6.1　報酬付与方法 · 105

5.6.2　離脱動作の導入 · 107

5.6.3　作業環境 · 110

5.6.4　パラメータ設定 · 111

5.6.5　シミュレーション結果 · 112
5.6.6　考察 · 113
5.7　おわりに · 117

5.1 はじめに

本章では，第3章，および第4章において提案した行動獲得手法の実環境への応用を行う．ここでは，2つの方法で提案手法を適用する．最初に，ロボットの動作原理に対して親和性の高い報酬付与方法を用いて計算機シミュレーション上において複数タスクに対する行動獲得を行い，獲得された状況認識機構および行動決定機構を実ロボットに適用する．これにより，提案手法による実機ロボットでの複数タスクに対する行動獲得の実現を確認する．第二に，ロボットの動作原理とは独立の，外的視点に基づく報酬付与方法により単一タスクに対する行動獲得をシミュレーション上で行い，教示者とエージェントとの視点の違いを提案手法が吸収し，教示者がエージェントの動作原理について考慮していない場合でも行動の獲得が可能であることを示す．

以下，第5.2節では，ここで行うシミュレーションおよび実験の目的を説明する．

第5.3節では，ここで行う2つのシミュレーションに共通するシミュレーション条件および実験条件を説明する．

第5.4節では，上記の前者のシミュレーションについて，シミュレーション条件及び結果，考察を示す．

第5.5節では，第5.4節で示したシミュレーションにおいて得られた行動を実ロボットに移植し，実ロボットにおけるタスク実現を評価する．

第5.6節では，後者のシミュレーションについて，シミュレーション条件，結果，考察を示す．

5.2 シミュレーション及び実験の目的

本章で行うシミュレーション及び実験では，実機のロボットと同様の身体性を持つシミュレーションにより，実世界と同様の環境において複数タスクに対する行動獲得を行い，ここで獲得された知識を実ロボットに対して適用することで，獲得された知識が実ロボットに対して妥当性を持つことを示すことを目的とする．

前章までで示したシミュレーションでは，グリッド環境におけるナビゲーション問題を扱った．グリッド環境への適用と本章で扱う連続的空間上での実際的な問題との違いは以下に挙げられる：

1. グリッド環境においては，あり得る観測値は0か1の要素からなるベクトルであり，これにノイズを加えたとしても，経験する観測値のクラスタは特定の離散的な点の周辺に集中する．このため，クラスタリングが容易であり，また騙し測度が正確に実際の騙しの有無を反映するため，状態構成が容易である．これに対して，本章で示すシミュレーションでは，観測ベクトルは連続的なベクトルとして現れるため，形状の異なる障害物の近傍で観測を行った場合，それらの観測値間の距離も連続値として求まる．

2. グリッド環境における報酬の与え方としてゴール地点へのManhattan距離を用いた場合，あり得る報酬値が $-1, 0, 1$ という有限かつ離散的な集合となり，経験する報酬の分布が明確に分離されるため，これを報酬ごとのクラスタに分割することが容易である．これに対して，本章ではゴールへの到達距離に基づく連続的な報酬値を採用している．

3. 本章で示すシミュレーション及び実験では，アクチュエータの誤差を特に仮定していないが，ロボットの動作原理が誤差を含むセンサ値からのフィードバックに基づいているため，動作にも誤差が含まれる．この結果として，動作が完了した時点でのロボットの位置・姿勢に誤差が含まれ，従って，観測値にはセンサ自体の誤差の他に位置・姿勢の誤差に起因する誤差も含まれる．エージェントの状況認識において用いる観測値は，この動作が完了した時点で得られる観測値であるため，この誤差は直接学習器において用いる観測空間上の誤差として表れることになる．

本章では，実世界の誤差の問題に対する提案手法のロバスト性，および提案手法を実際に実ロボットの制御機構と組み合わせて適用することの妥当性の検証を行う．

第5.4節で示すシミュレーション，および第5.5節で示す実験は，実ロボットによる複数タスクに対する行動獲得の実現の確認および評価を目的とする．ここでは，第3章および

第4章において提案した手法が，実機によるナビゲーションタスクという誤差を含んだ連続的空間における問題に対して適用可能であることを確認する．

第5.6節で示すシミュレーションでは，教示者の報酬付与方法が，ロボットの動作原理に即したものではない場合について，教示者・エージェント間の視点の違いに対して提案手法が対処しうることを確認することを目的とする．

5.3 シミュレーションおよび実験の概要

本節では，本章で扱うシミュレーション及び実験の概要を説明する．

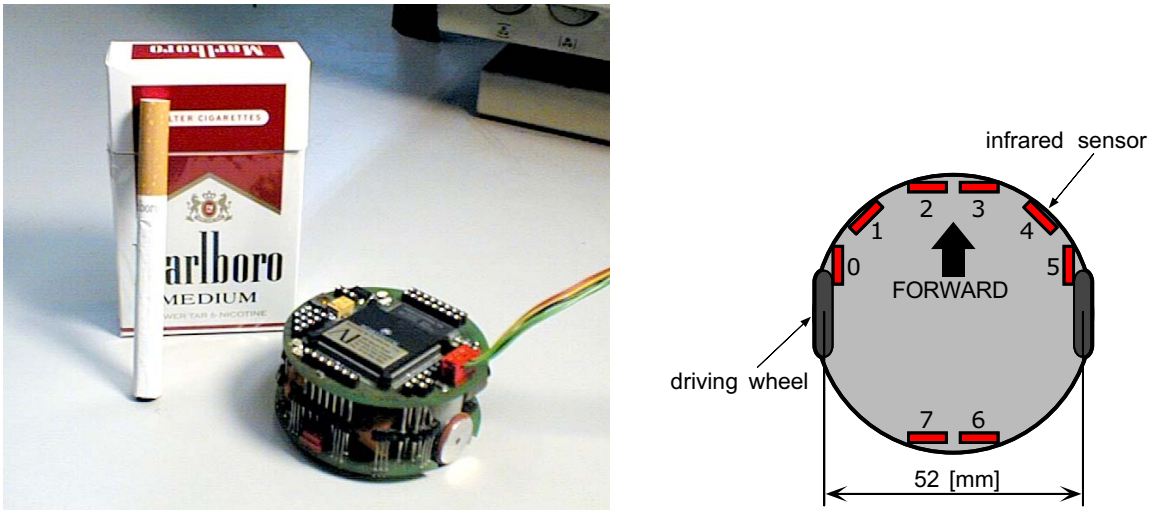
本章では，小型移動ロボット Khepera を想定したシミュレーション，および Khepera を用いた実験を行う．

第 5.3.2項では，Khepera を動作させる際に用いる基本的動作原理として用いた，行動規範型動作プリミティブについて説明する．

第 5.3.3項では，実環境に対応するために手法に加える変更について説明する．

5.3.1 実機ロボット Khepera

シミュレーションで想定し，実機実験において用いる実機ロボット Khepera (Fig.5.1) は，ローザンヌ連邦工科大学 (EPFL) で開発された実験用小型移動ロボットである．



(a) Khepera　　(b) arrangement of sensors

Fig. 5.1: Khepera

5.3.1.1 移動機構

移動機構は 2 駆動輪型車輪機構であり，左右の車輪にそれぞれ速度指令あるいは位置指令を与えることで動作する．速度指令の分解能は 8 [mm/s] である．シミュレーションで想定し，実験で用いるのは速度指令による動作である．

5.3.1.2 センサ

センサとして，センサを図中 (b) の配置で8つ搭載している．このセンサは赤外線を放射して反射光の強さを測定する近接覚センサとして，および周辺光の強さを測定する周辺光センサとして利用可能である．シミュレーションで想定し，実験において用いるのはこのうち，近接覚赤外線センサである．

センサの返す値は0から1023までで，距離が近いほど大きな値を返す．Fig.5.2には，8つのそれぞれのセンサの正面方向に白色プラスチック平面の障害物を置き，Kheperaの中心位置と障害物との距離に対するセンサ読みとり値を計測したものである．図が示すとおり，センサごとの機差が大きい．

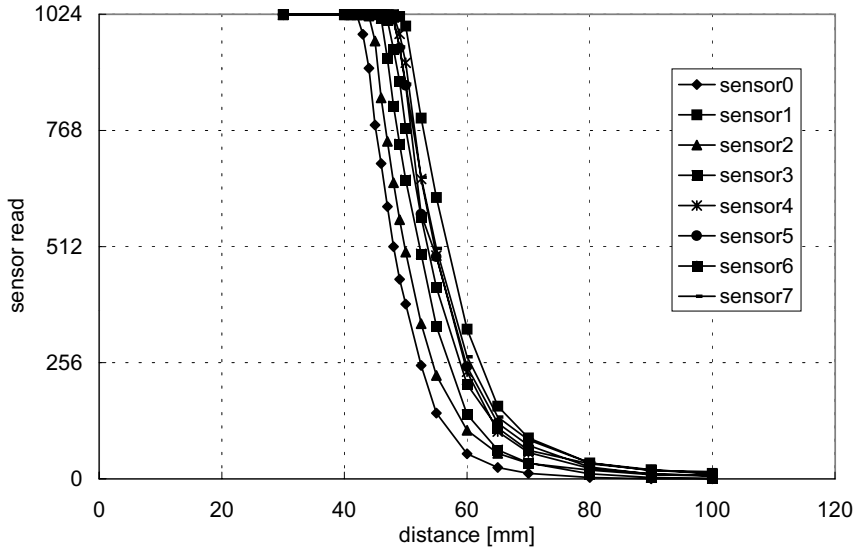


Fig. 5.2: Sensor profiles

また，Fig.5.3は同一の条件に対して複数回センサを読みとったときのセンサが返す値の誤差を，読みとり値の大きさに対してプロットしたグラフである．

5.3.1.3 CPU, 動力

Kheperaは内蔵するCPUおよび電池による自律動作および，外部処理装置・外部電源とのケーブル接続による動作が可能である．今回の実験においては外部のワークステーションとのシリアルポート接続により動作させた．

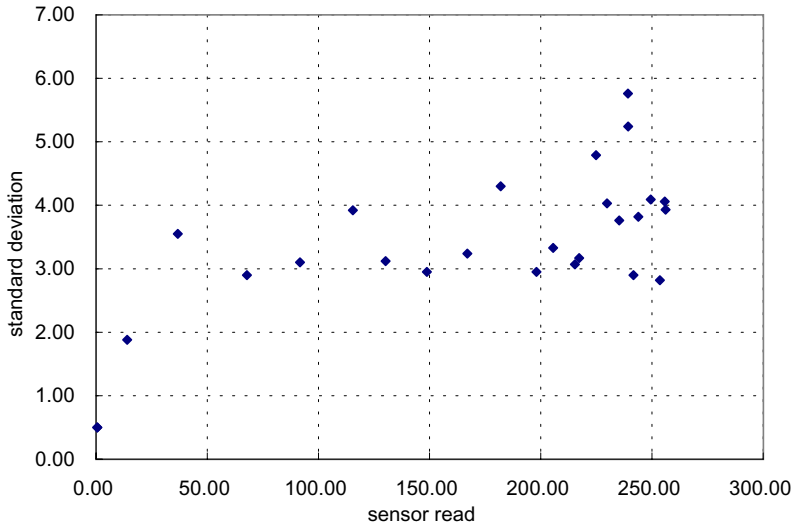


Fig. 5.3: Sensor errors

5.3.1.4 センサ・モータのモデル化

シミュレーションにおける Khepera のセンサのモデル化は以下に示す通りに行った．Fig.5.4に示すように各センサから各方向θにn本の視線ベクトルを想定し，それぞれの視線ベクトルと最も近い位置で交わる障害物について，交点までの距離d，視線ベクトルと障害物表面の法線ベクトルがなす角度ϕを求める．

また，事前に測定した実測データに基づき，各θ方向の単位角度あたりのセンサ読みとり値，交点までの距離dに応じた読み取り値の減衰率，視線と障害物のなす角度ϕによる減衰率を用いて各視線方向の読み取り値を求め，これを合計する．

誤差としては，Fig.5.3のグラフを線形近似して求めた標準偏差による正規分布の誤差を印可した．ただし，誤差を加える以前の読み取り値がほぼ 0 の場合に限り，標準偏差を 2.0 とすることで，実機のノイズに近い分布を与えた．

モータについては，速度指令から求まる両車輪の速度に基づく単位時間あたりロボット変位を機構学的に求めた．実機のノイズが比較的小さいこと，および後述する行動規範型動作プリミティブにおいてはセンサフィードバックが用いられているためにセンサの誤差に比較してモータ側の誤差が大きな影響を持たないと考えられることから，モータ系については誤差を与えなかった．

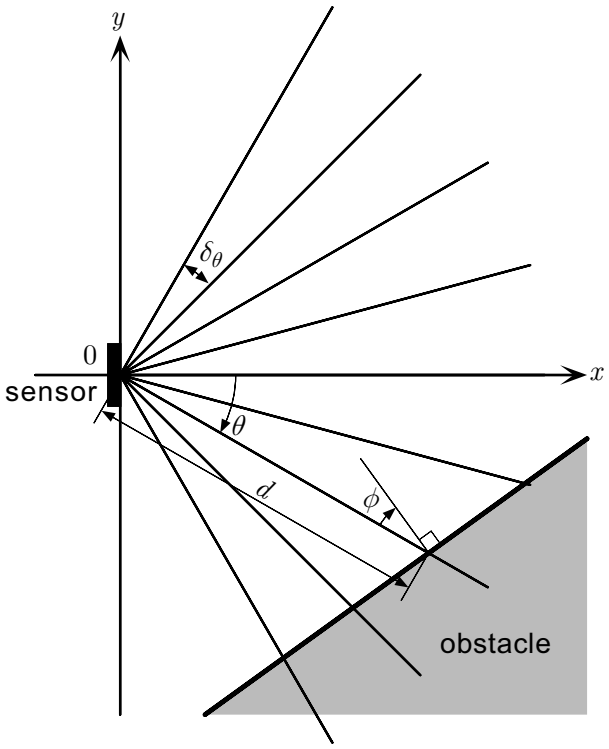


Fig. 5.4: Sensor model

5.3.2 行動規範型動作プリミティブ

シミュレーションおよび実験におけるロボットの行動は，行動規範型動作プリミティブ (behavior-based motion primitives) を用いた．行動規範型動作プリミティブとは，Brooks によって提唱された行動設計のパラダイムである行動規範型 (Behavior-based) の知能設計の枠組みに基づいて設計された動作プリミティブである [1, 8].

行動規範型ロボティクスの枠組みでは，ロボット内部における知識構造 (内部モデル) に基づく複雑な内的処理を排除し，外部からの各時点のセンサ入力に対する反射行動を実現するモジュール群の相互作用から行動を創発することにより，未知・動的な外部世界との間をセンサ・アクチュエータによって直結された行動決定機構を構築する．Brooks はこの行動規範型アーキテクチャに基づく行動決定機構である包摂アーキテクチャ (subsumption architecture) [2] を用いて動的な環境においてロバストに動作しうる移動ロボットを設計した．[4] では，行動規範型を用いた複数ロボットによる協調行動の計画手法を提案している．

用いた動作プリミティブは以下の 3 種 (壁沿い走行については，左右どちら側に障害物を見て動くかに応じた 2 通りがある) である．

5.3.2.1 自由行程の前進 (Freeway)

障害物に近づくまで，自由空間を前進する．

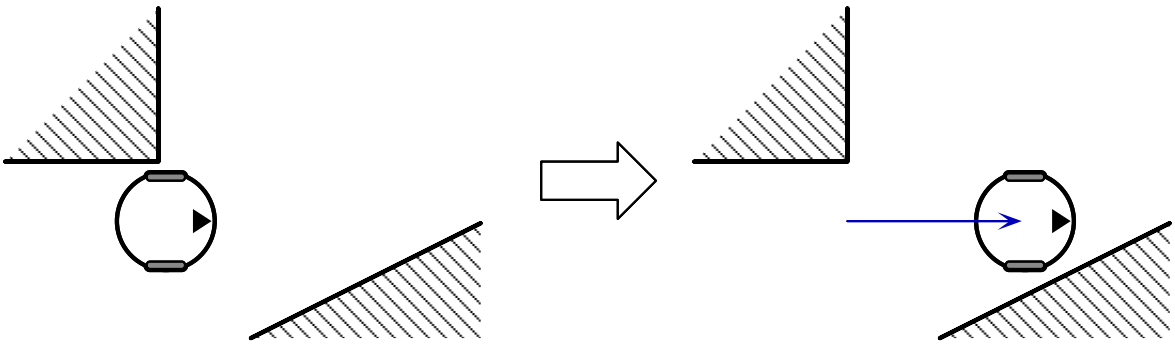


Fig. 5.5: Freeway behavior

起動条件

前方および側方に障害物が存在しない．具体的には後方以外の 6 つのセンサのいずれかが閾値 (300 を与えた) を超えていない．

動作指令

左右両輪に対して 2，すなわち 16[mm/s] の前進動作．

停止条件

前方及び側方に障害物が存在する．具体的には後方以外の 6 つのセンサのいずれかが閾値 (300 を与えた) を超えている．

5.3.2.2 壁沿い走行 (Wall-following)

障害物の表面に倣って前進する．

起動条件

いずれかの方向の近傍に障害物が存在する．具体的には全てのセンサの読み取り値を加えた値が 200 を超えている．

この動作は 2 段階の手順を踏む．

(a) **障害物表面に対して平行な方向を向く**

左に壁を見る壁沿い走行を行う場合は，右方向に，逆の場合は左方向に，壁に沿って走行できる状態になるまでその場で回転運動を行う．

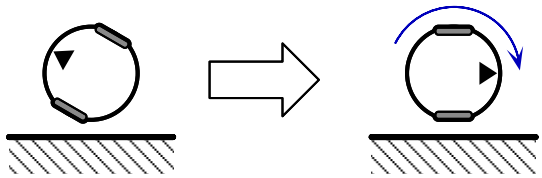


Fig. 5.6: Aligning to obstacle

動作指令

右回転の場合は，左輪に 2，右輪に-2．左回転の場合は，左輪に-1，右輪に 2．

停止条件

右回転の場合は，(1) 前方の 2 つのセンサの読み取り値のいずれもが 10 を下回り，(2) 左斜め前方のセンサの読み取り値が 100 以下，かつ (3) 左方向のセンサの読み取り値が 100 以上．

左回転の場合は，(1) 前方の 2 つのセンサの読み取り値のいずれもが 10 を下回り，(2) 右斜め前方のセンサの読み取り値が 100 以下，かつ (3) 右方向のセンサの読み取り値が 100 以上．

(b) **障害物に沿って走行する**

障害物表面との間に一定の距離を保ちながら走行する．

動作指令

基準の速度指令を左右輪ともに 3 とし，センサ値に応じて以下の補正をかける．

左側 (右側) に壁を見る壁沿い走行において，

左 (右) 側のセンサ値が 300 未満の時は左 (右) 輪の速度指令から 1 を，100 以下の時は 2 を引く．(壁から離れすぎたら近づく)

左 (右) 側のセンサ値が 300 以上の時は右 (左) 輪の速度指令から 1 を，600 以上の時は 2 を引く．(壁に近づきすぎたら離れる)

左 (右) 斜め前のセンサ値が 150 以上の時は右 (左) 輪の速度指令から 1 を，600 以上の時は 2 を引く．(斜め前方に角があるときは避ける方向を向く)

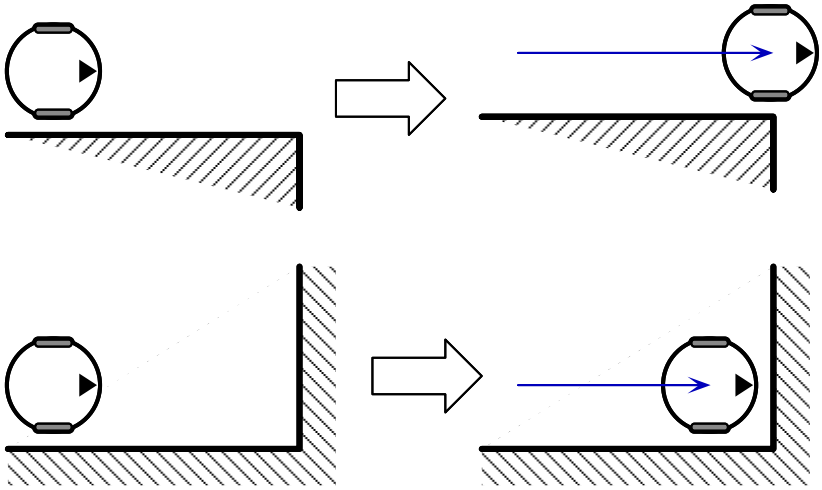


Fig. 5.7: Freeway behavior

これらのセンサフィードバックにより，ロボットは真横のセンサが 300 程度の値を保つ程度の距離を障害物との間に保ちながら走行することができる．

停止条件

沿っていた障害物がとぎれた場合，あるいは前方に障害物が現れた場合．具体的には，ロボット側方の障害物を向いたセンサの読み取り値がほぼ 0 となった場合，あるいは前方のセンサのいずれかが 300 を超える値を返した場合．

5.3.2.3 行動規範型動作プリミティブの導入の意義

行動規範型動作プリミティブを導入することで，エージェントと環境との身体性に基づく拘束関係の実現が図られる．即ち，エージェントの身体性に即した表現に基づいて与えられた環境との相互作用の様式としての動作プリミティブにより，外部の設計者の視点から構築したエージェントのモデルにおいてしばしば生じる，設計者によるモデルとエージェント自身の身体性とのずれの問題を解決し，制御器レベルにおいてロバストな環境との相互作用が実現される．この方法は，行動規範型動作プリミティブに基づく，定性的地図による移動ロボットの定性ナビゲーションなどにより，環境の未知性，動的性に適応可能なロバストな制御モデルとして認められている．

5.3.3 誤差への対応

本章で行うシミュレーションでは，観測に誤差が含まれるため，ここまでで提案した手法に誤差に対応した若干の修正を行う．

具体的には，第4章において提案した複数タスクへの対応手法において，確信度の更新式4.1を修正し，以下のようにする：

$$c_i \leftarrow c_i + \bar{d} - d_i \tag{5.1}$$

また，複数タスクでの知識適用段階においては，最高の確信度に対応するタスクに関する知識が適用されるが，確信度には観測のノイズがそのまま含まれることから，これを多純な数的比較での最大値とするのではなく，最大値から幅ωの範囲内の確信度は，全て「最大確信度」と指定し，確信度に大きな幅が出なければタスクの識別は行われないものとした．

5.4 実環境における複数タスク実現に関するシミュレーション

本節では，実環境における複数タスク実現を検証するための計算機シミュレーションについて示す.

5.4.1 シミュレーション・実験の環境

Fig.5.8にシミュレーション及び実験の環境を示す．図中，障害物の各頂点には，左下隅を原点とした座標値を mm 単位で表記した.

3つのスタート点 S_1，S_2 および S_3 の位置・姿勢はそれぞれ,

S_0 : (40.0,40.0) 0.0[rad]
S_1 : (544.0,40.0) 1.570796[rad]
S_2 : (189.0,387.0) -1.82

とした．また，ゴール点の位置は (356.5,226.5) であり，ゴール点に到達したと見なす領域はゴール座標 (356,225) から半径 26[mm] (ロボット半径) 以内の領域とした.

Fig.5.9には，この環境においてあり得る状態を示した．小さな円形は，エージェントがいずれかの行動規範型動作プリミティブを実施した際に，停止条件が満たされて停止する位置であり，これが外部世界における状態空間における状態となる.

図中に示した矢印は，動作による状態遷移を表し，アルファベットはその遷移に対応する動作である．また，状態に記載された数字は，対応する状態からゴール状態に至るまでの最小動作ステップ数を示しており，これに基づいて即時報酬を与える.

実機システムにおいては，図と同様の環境を，白色プラスチック板を用いた障害物によって構築した.

5.4.2 シミュレーションの設定

シミュレーションにおける，状態認識・行動決定機構の獲得過程におけるパラメータ設定を 5.4.2に示す.

また，知識適用段階において最大確信度と見なす確信度の幅 ω としては 0.2 を与えた.

エージェントに与える即時報酬は，以下の通り与えた：1 ステップ前の状態 s_{t-1} におけるゴール状態までの最短所用動作ステップ数に比較して，現在のそれが減少したとき $+1$，増加したとき -1，増減なしの時 0.

Table 5.1: Parameters used in the simulation

State- space construction	α	0.1
	T_b	0.1
behavior learning	ν	50
	δ	0.25
	k	2.0
	$\mathcal{D}_T$	0.5
	ρ	0.4

5.4.3 シミュレーションの結果

Fig.5.10にシミュレーションにおいて得られたエージェントの軌跡を示す．ただし，このプロットは実機Kheperaの制御サンプリング時間 (およそ100[ms]) ごとのエージェントの位置を示している．また，経路脇に示したアルファベットは，その部分においてエージェントが行っていた動作プリミティブを示している．

図に示すとおり，全てのタスクに対して最短ステップ数でのゴール到達が得られた．

Fig.5.11には，タスク2において各時点における，(a) 知識獲得段階において経験された観測値と実際の観測値との距離，および各時点における確信度 (b) を示した．

図に示すとおり，左に壁を見る壁沿い走行を1度行った時点で，正面に現れる壁面の角度の違いによって今回のタスクを識別している．

Table5.2には，各タスクに対して得られた木構造のノード数，最大深さを示す．

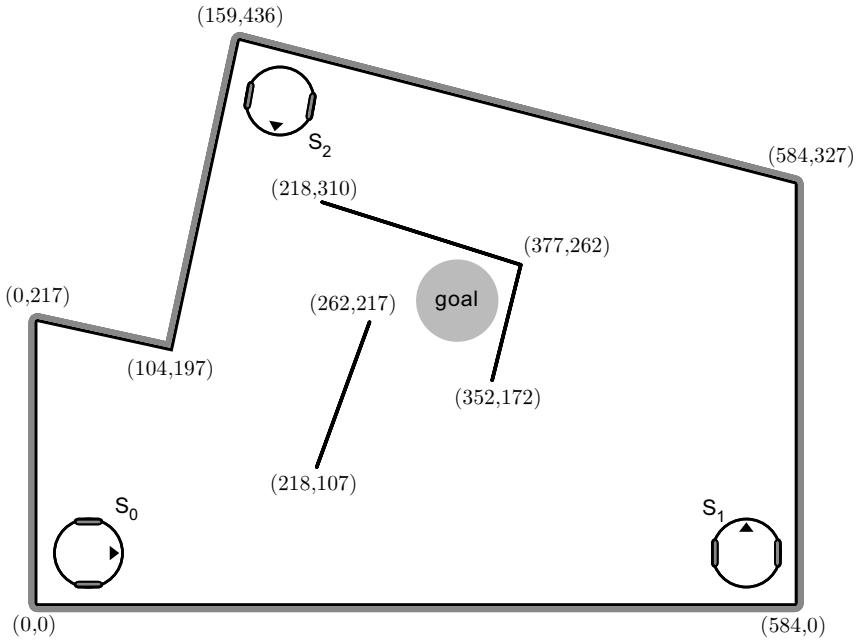
Table 5.2: Acquired tree

	task 0	task 1	task2
number of nodes	70.6	58.4	74.2
maximum depth	3	3.2	3.2

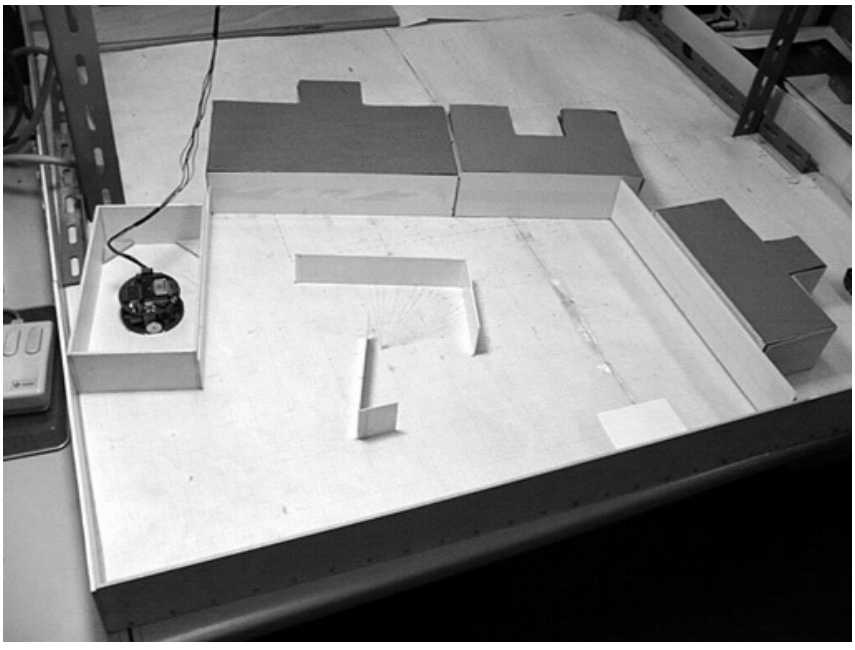
5.4.4 考 察

結果に示したとおり，シミュレーション上では各スタート点に対してステップ1でスタート点の識別が行われ，タスクが識別された後の行動も正しく行われた．

このシミュレーション結果から，ノイズを含んだ連続的観測において，実際のロボットと同様の制御則に結合された提案手法が正しく動作することが確認された．



(a) Arrangement of environment



(b) Real experimental environment

Fig. 5.8: Experimental environment

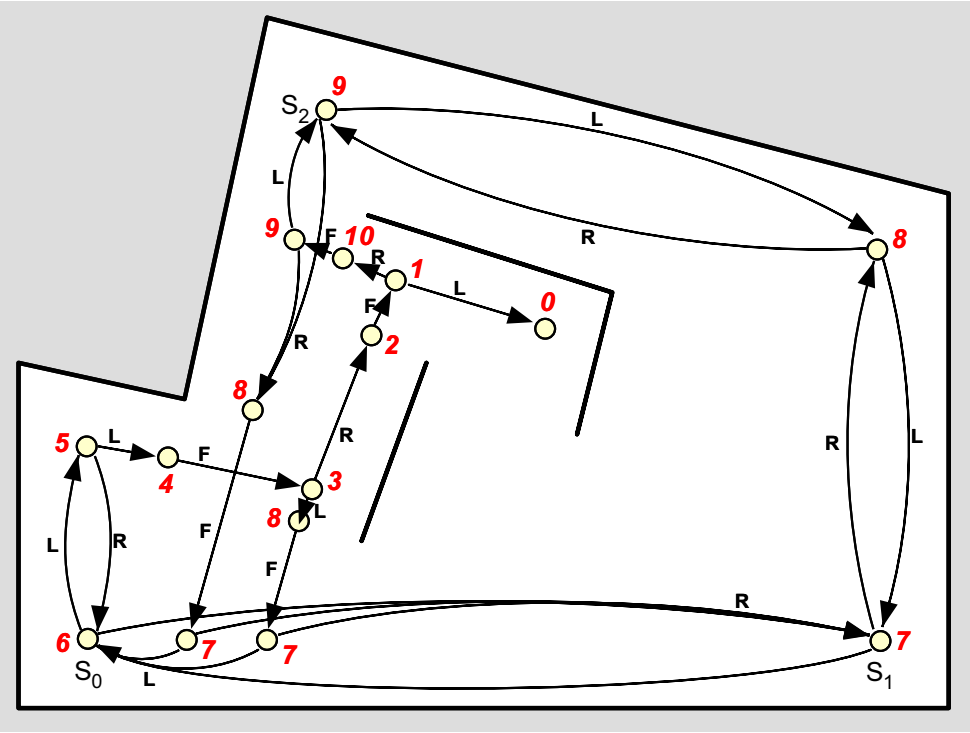
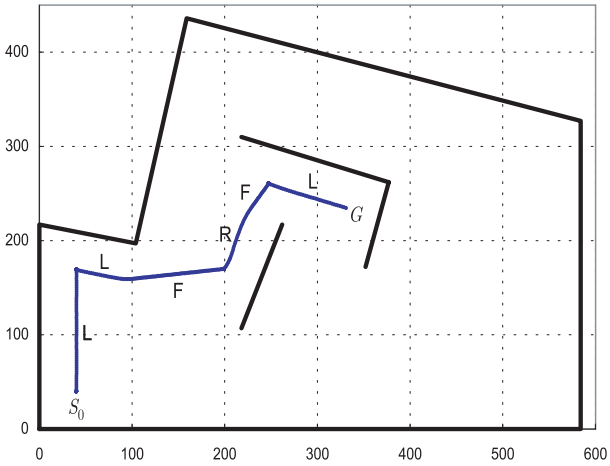
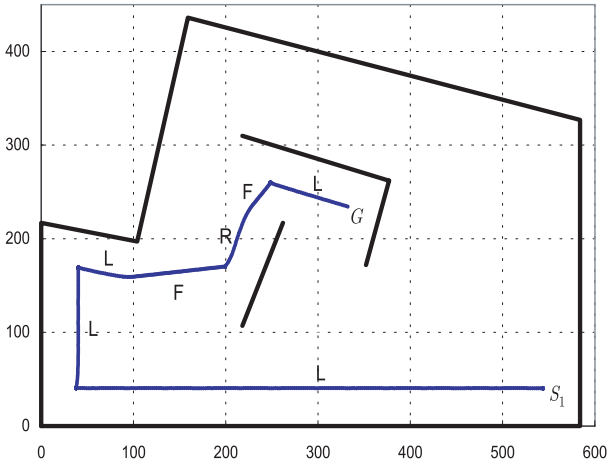


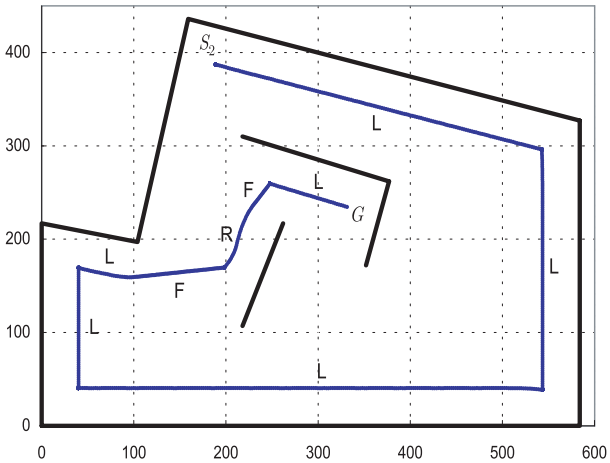
Fig. 5.9: Experimental environment



(a) S_0

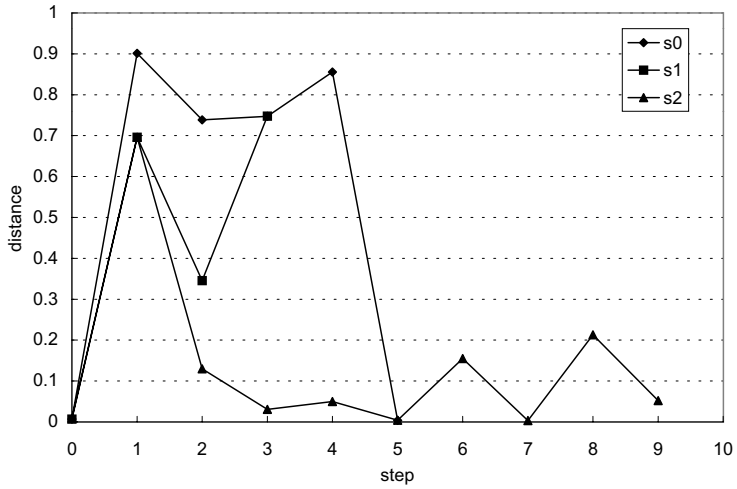


(b) S_1

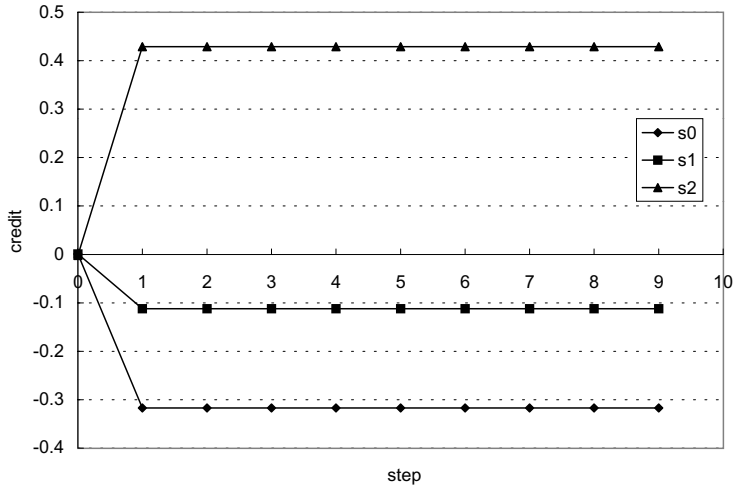


(c) S_2

Fig. 5.10: Trajectory of the simulated agent



(a) Distance



(b) Credit

Fig. 5.11: Development of credit for task2

5.5 実機複数タスク実現に関する実験

本節では，前節のシミュレーションの結果として得られた知識を用いた実機実験の結果について説明する．

5.5.1 実験結果

Fig.5.12には，実験におけるロボットの動作の例として，タスク2に対応するスタート点にロボットをおいたときのロボットの軌跡を示す．図に示すとおり，ロボットはシミュレーションと同様の最適行動を実行した．

Fig.5.13には，実験においてそれぞれのタスクに対応するスタート点に置かれたロボットの軌跡を示した．ただし，図はそれぞれの動作プリミティブが停止した時点でのロボット位置を示しており，途中の経路については数量的な測定は行っていない．

Fig.5.14は，タスク2に対応するスタート点にロボットをおいたときの，知識獲得段階における観測値の経験と実測観測値との距離，およびそれに基づいて更新された確信度を示している．シミュレーション時と同様，ステップ1において既にタスクが識別されている．

しかし，シミュレーションにおける距離・確信度の値とは若干のずれが見られた．これは，シミュレーションシステムにおける実機ロボット・環境のモデル化の誤差であると考えられる．

Fig.5.15は，タスク1に対する同様のグラフである．この場合は，ステップ0において実行された左壁沿い動作の停止位置・姿勢に誤差があったことから，1ステップ目ではタスクの識別が完了しなかったが，2ステップ目で識別ができた．

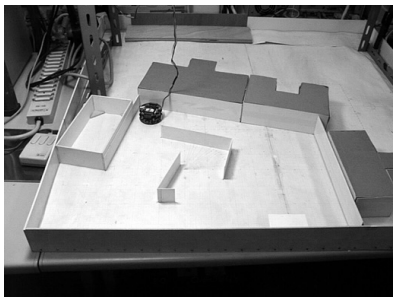
実験においては，シミュレーションとは異なりタスクの識別に2ステップを要するケースも見られたが，最終的には正しいタスクを識別した．

ただし，実験においては，上記の通りシミュレータのモデル化誤差に起因して，失敗が起こることがあった．具体的には，スタート0から行動 “L” を2回行って停止する位置の状態において，停止した時点での姿勢がシミュレーションに比較して大きなばらつきを持っており，このことに起因して，この後前進動作を行って障害物に突き当たって停止した瞬間に得られる観測において実験・シミュレーション間に比較的大きな違いが見られ，これにより状況識別機構における観測に基づく識別において不都合が起こる．

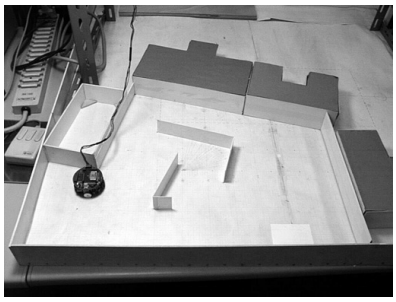
5.5.2 考 察

シミュレーションで得られた行動が実機システムにおいて動作したことで，シミュレーション結果が実機での問題を解決しうる認識機構・行動決定機構を獲得することに成功したことが示された．

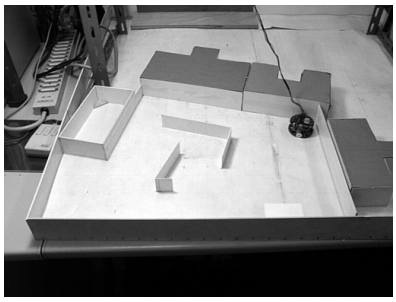
実世界のセンサ値のシミュレータへのエミュレーションの不適切性から，若干シミュレーションとは異なる動作が得られ，これにより行動が失敗することが起こった．ただし，シミュレーション世界の中においては学習が正しく行われていることから，もしも全プロセスを実機により行ったとすればこの問題は解決されると思われ，これは本質的な問題ではないと言える．



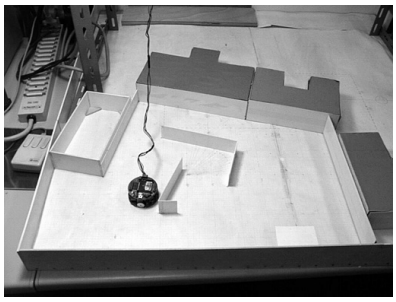
(a) Step 0



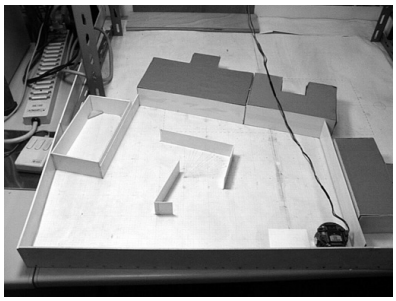
(b) Step 5



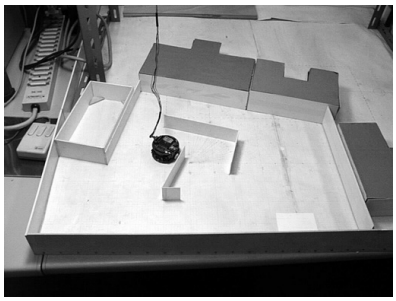
(a) Step 1



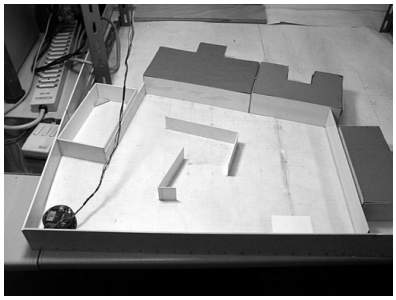
(b) Step 6



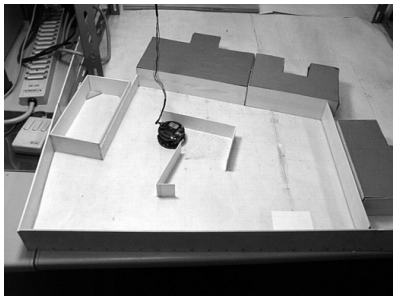
(a) Step 2



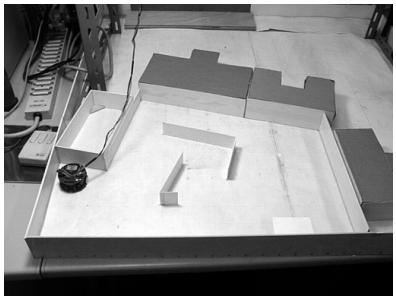
(b) Step 7



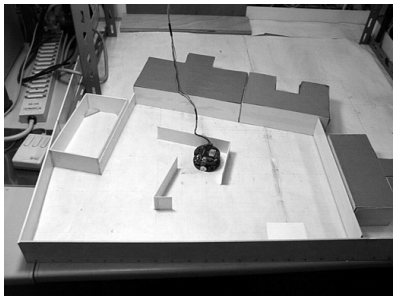
(a) Step 3



(b) Step 8

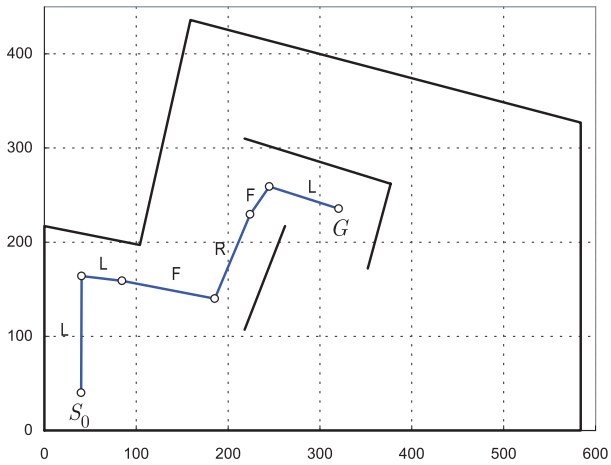


(a) Step 4

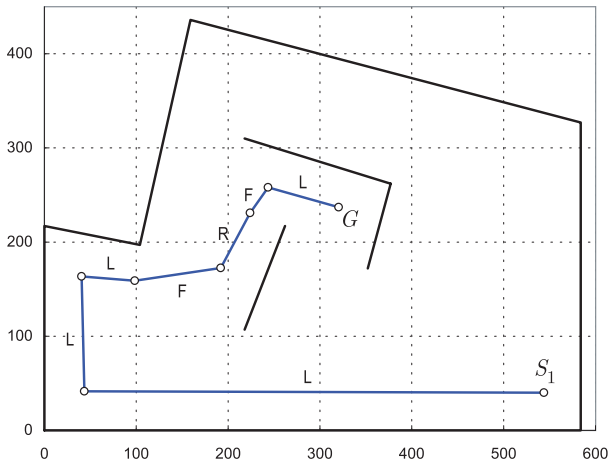


(b) Step 9

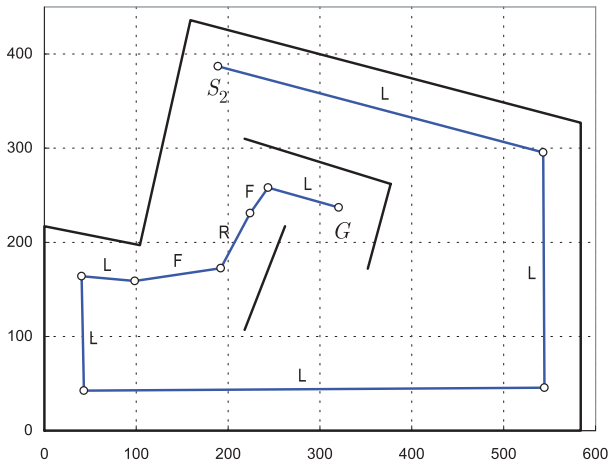
Fig. 5.12: Path of robot starting from S_2



(1) S_0 からの軌跡

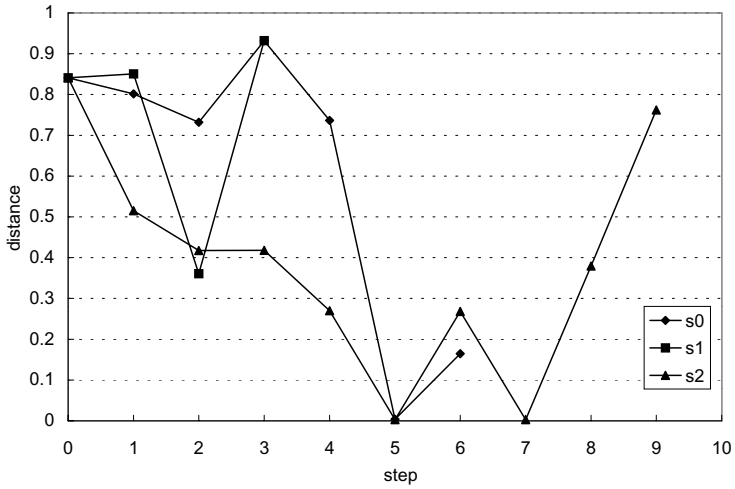


(2) S_1 からの軌跡

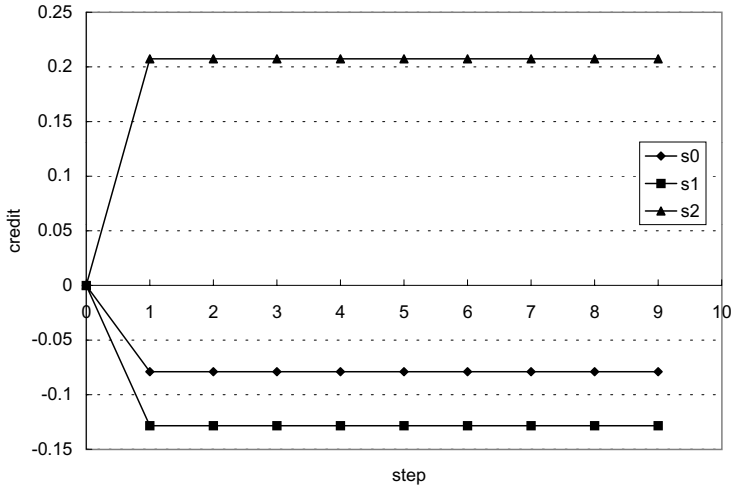


(3) S_2 からの軌跡

Fig. 5.13: Trajectory of robot

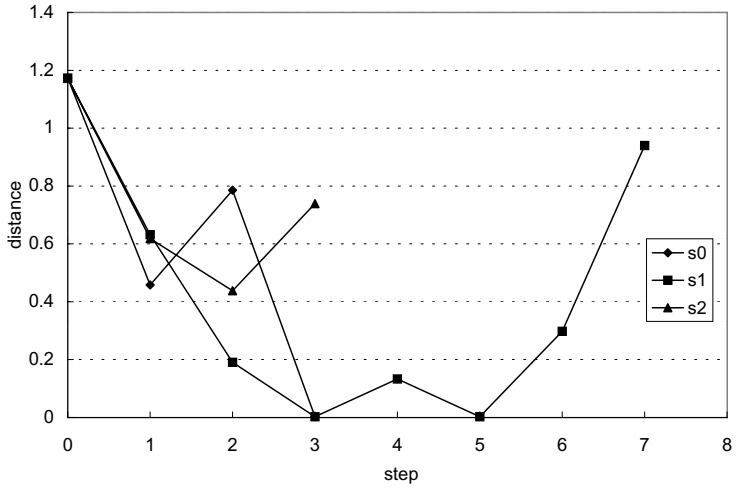


(a) Distance

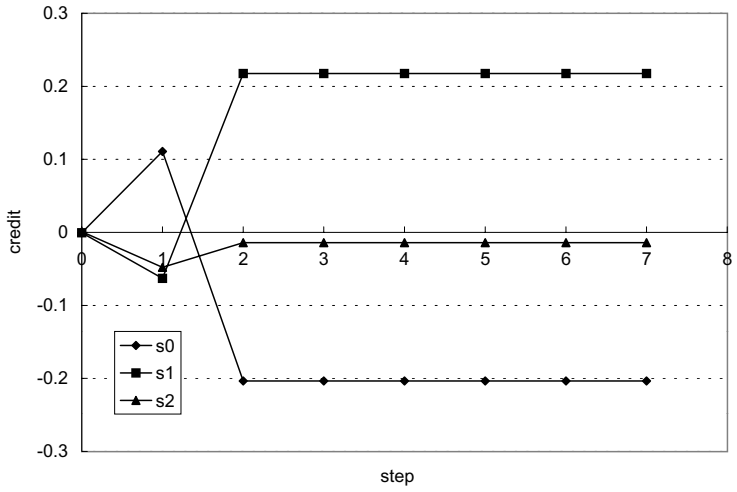


(b) Credit

Fig. 5.14: Development of credit for task2



(a) Distance



(b) Credit

Fig. 5.15: Development of credit for task2

5.6 異なる視点に基づく報酬付与への対応に関するシミュレーション

本節では，外部から即時報酬を与える教示者の報酬付与方法がエージェントとは異なる視点に基づいている場合についての行動獲得に関するシミュレーションを行う．

前節までのシミュレーションおよび実験においては，エージェントに与えられる即時報酬はエージェントの動作原理，すなわち行動規範型動作プリミティブによってゴールへ到達するまでの，エージェントの動作原理に従った場合の経路に沿った距離に基づいて算出されていた．

しかし，本来教示者はエージェントの外部の視点からエージェントの行動を評価しており，その評価は一般にエージェントの動作原理に依存しない．そこで，ここではエージェントの動作原理とは独立した報酬算出方法に基づいて与えた即時報酬によって，提案手法によるエージェントが状況認識機構・行動決定機構の獲得を確認する．

5.6.1 報酬付与方法

本節で行うシミュレーションにおいては，エージェントの動作原理とは独立した報酬付与方法として，Wavefront アルゴリズム [9]（pp.431-478）に即した報酬計算方法を用いる．

ここで，Wavefront アルゴリズムの概略を 5.16に示す．図中，黒点はゴール位置であり，太線で障害物を示した．また，灰色で示したのはコンフィギュレーション障害物（C-obstacle）であり，この領域の中にロボットが入るとロボットと障害物が干渉する領域を表す（本節でのシミュレーションでは，簡単のためコンフィギュレーション障害物の形状は図に示す通りとした）．従って，ロボットが移動可能な自由空間はコンフィギュレーション障害物の外側の領域となる．

Wavefront アルゴリズムは，移動ロボットのナビゲーション問題において，環境上の全ての自由空間上の位置からゴール位置までの最短経路上の距離を算出する手法である．ゴール点に距離 0 を割り当て，既に距離の割り付けられた全ての点から，距離を増大させながら近傍の点へ距離の割付を拡張していく．例えば図上では，ゴールから距離 1 から距離 6 までにに対応する点の集合はゴールを中心とする円となる．このようにして距離を増加させながら等方的に自由空間上に距離の割付られた領域を伝播させていき，全自由空間が被覆されたとき，自由空間上の全ての位置に，その点からゴールまでの最短経路上の距離が割り付けられる．

Wavefront アルゴリズムによって作られたこのような勾配に従って，最急降下方向に向かうことで，ロボットはゴールにたどり着くことができる．

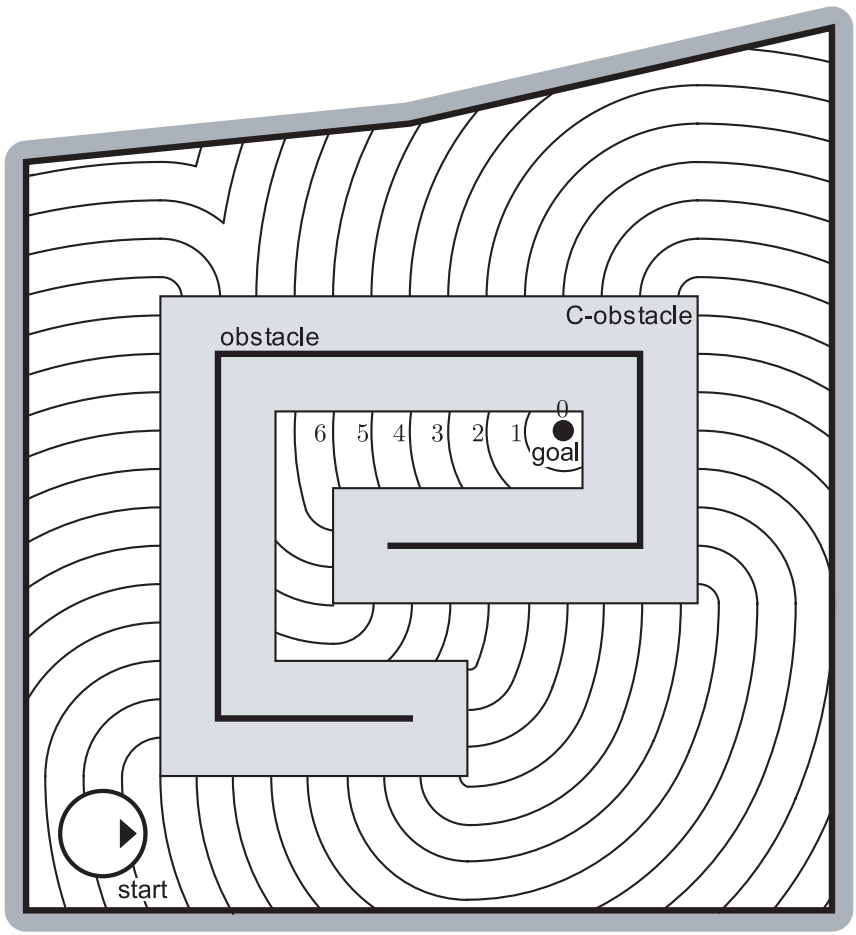


Fig. 5.16: Potential field based on wave-front method

Wavefront アルゴリズムによって張られた勾配場における，各点のゴールまでの最短距離は，実際には以下の方法で計算することが可能である：コンフィギュレーション障害物の頂点とゴール点および対象とする点（ロボット位置）を頂点として可視グラフ [24] を張る（Fig.5.17）．このグラフの各片に対して辺の長さの重みを与え，Dijkstra のアルゴリズム等を用いた最短路探索により対象とする点からゴール点への最短経路長を求める．

Wavefront アルゴリズムによる勾配に即した即時報酬には，ある動作を開始する時点におけるロボット位置でのゴールへの経路長と，動作を終了した時点での経路長とを比較し，その減少量を定数倍した値を用いる．

ここで，ロボットに評価信号を与える外部教示者の視点からは，特定の時点におけるロボット位置姿勢からゴールまでの最短距離の経路長が常に計算可能であり，ロボットは動作プリミティブ実行途上においても常にこの値を外部から受けることが可能であるとする．

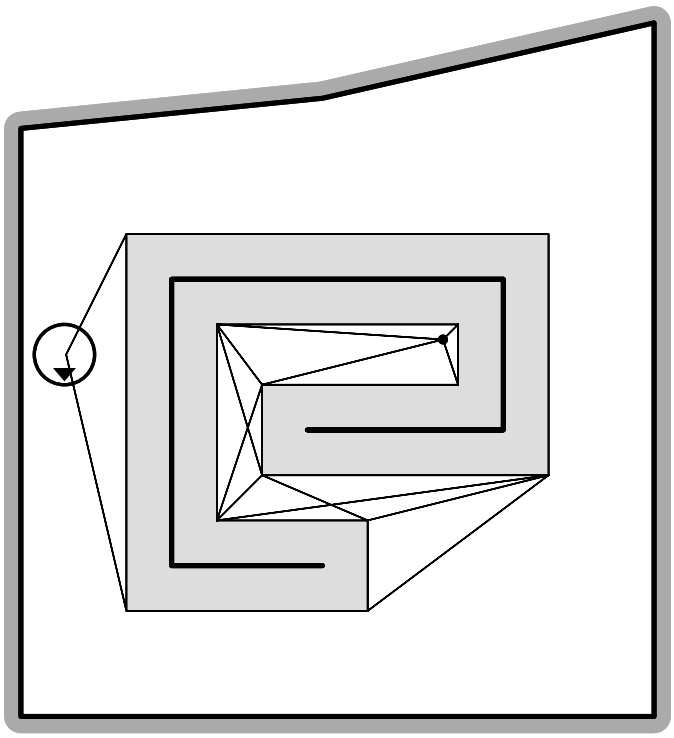


Fig. 5.17: Visibility graph

5.6.2 離脱動作の導入

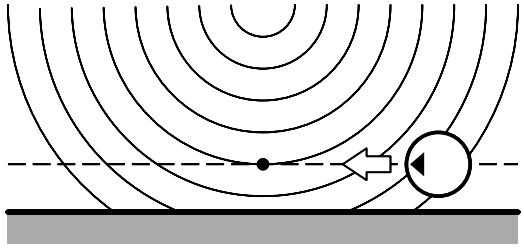
5.6.2.1 離脱動作の概要

前述のWavefrontアルゴリズムに基づく勾配によるナビゲーション手法では，勾配の最急降下方向に直接移動可能なロボットであればゴールに到達することが可能である．しかし，本章で想定している行動規範型動作プリミティブでは，ロボットの動きは環境からの拘束を直接に受ける．具体的には，自由行程の前進動作は動作開始時点の方向以外には進むことができず，壁沿いビヘービアは障害物表面に平行な方向にしか移動ができない．

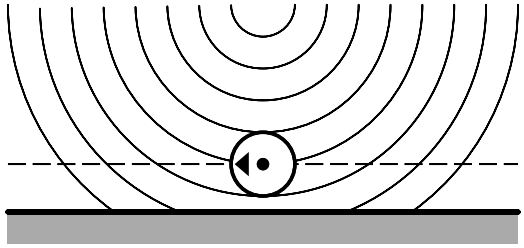
本シミュレーションでは，勾配の減少方向に積極的に接近することのできる動作プリミティブとして，離脱動作を新たに加え，同時に壁沿い動作に変更を加える．

離脱動作の概要をFig.5.18に示す．

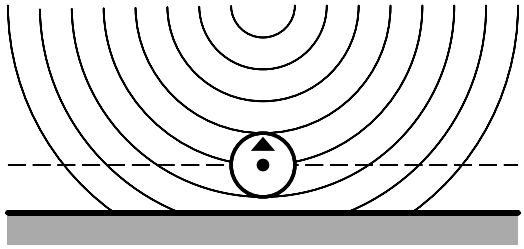
離脱動作は，壁沿い走行においてWavefront勾配の谷間に到着した際，その位置から垂直方向に障害物を離脱する動作である．図中(a)は，ロボットが左に壁を見る壁沿い動作を実行している様子を示している．同心円はWavefront勾配で，円の中心ほどゴールへの距離が小さいものとする．このとき，壁沿い動作の経路上に，ゴールへの距離の極小点を



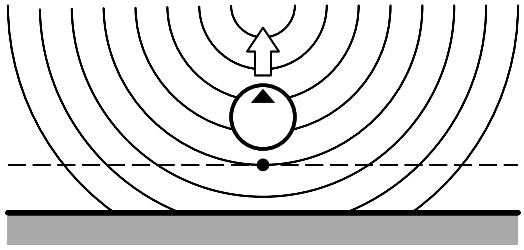
(a) Wall-Following behavior



(b) Stopping at minimal point



(c) Turning to normal vector direction



(d) Breaking away from wall

Fig. 5.18: Breakaway behavior

与える位置が存在する（図中黒丸）.

後述するとおり，壁沿い動作の停止条件としてこの極小点の検知を追加することで，ロボットは極小点で動作を完了する.

離脱動作はこの停止条件に基づいて壁沿い動作が停止した直後においてのみ起動可能な動作であり，この状態で離脱動作が起動されると，ロボットは障害物から離れる方向に転

回し，そこから自由行程の前進動作と同様の動作原理に基づいて直進運動を開始する．

離脱動作の詳細は以下の通りである．

5.6.2.2 離脱動作（Breakaway）

障害物から離れる方向に直線運動を行う．

起動条件

(1) 前回行った動作が壁沿い動作 (L または R) である．

(2) 前方に障害物が存在しない：前方の 2 つのセンサ 2，3 の読み取り値がともに 300 を下回る．

(3) 障害物がとぎれたことによる停止ではない：障害物側のセンサ（0 または 5）の読み取り値が 5 以上．

動作指令

(1) 障害物と反対方向に展開する

前回の動作が左に壁を見る壁沿い動作の場合は右に 90 度，逆の場合は左に 90 度回転する．移動ロボット Khepera を想定する場合，Khepera には左右両輪の目標回転角を与える位置制御指令を与えることができ，位置制御指令による動作は正確であるため，ここではセンサフィードバックを用いずにその場で 90 度回転が正確にできることを想定する．

(2) 自由空間の前進動作

回転が完了した後，自由空間の前進動作と同様の動作指令，停止条件で動作する．

停止条件

自由空間の前進動作と同様に，前方，左右あるいは斜め前方のセンサが 300 を超える値を返したとき停止．

5.6.2.3 壁沿い動作の変更

障害物から離脱する行動を実現するため，壁沿い動作に対して新たに停止条件を一つ追加する．

停止条件

前々回の動作サンプリング時の位置におけるゴールまでの最短距離が前回の動作サンプリングにおけるそれよりも減少しており，かつ今回の動作サンプリングにおける距離が前回よりも増加している．

5.6.3 作業環境

本シミュレーションではFig.5.19に示す環境を用いる．

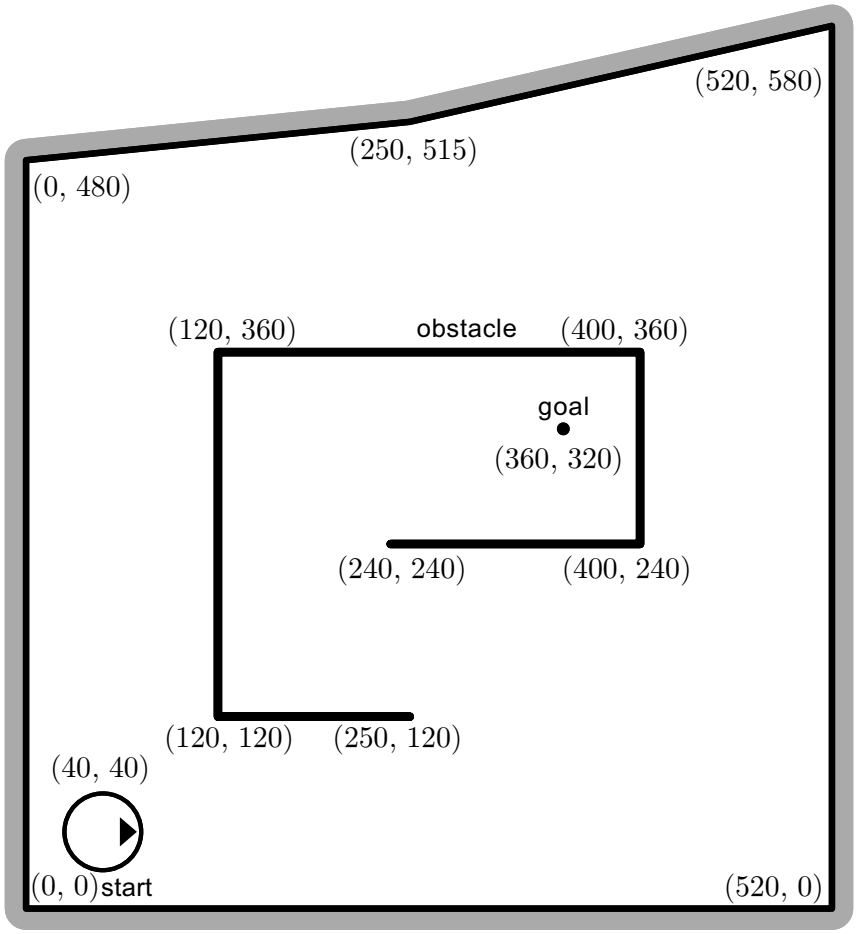


Fig. 5.19: Working environment

ここでは単一タスクを扱い，スタート点は座標 $(40, 40)$，スタート姿勢は0[rad]（図上では右方向）とする．

この環境において，ゴール点からWavefront法による勾配を生成したとき，壁沿い動作の経路上における距離の極小点は5点存在し，従って起動可能な離脱動作はFig.5.20に示す5通りが可能である．

なお，図に示すとおり，離脱動作を実行した場合の自由行程前進動作の経路はコンフィギュレーション障害物に極めて近くなってしまうため，ゴールへの最短経路長を求めるグラフ探索においては，実際のコンフィギュレーション障害物から一定距離のオフセットをとった点をグラフの頂点として利用した．具体的には，障害物表面からコンフィギュレー

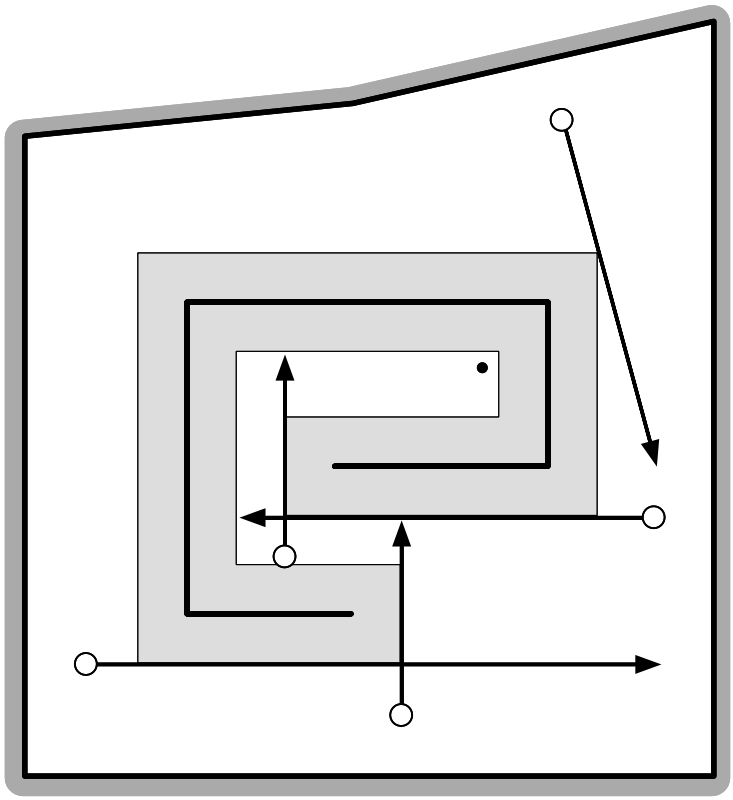


Fig. 5.20: Breakaway points

ション障害物表面までのオフセットが35[mm]であるのに対して，障害物から可視グラフ頂点までのオフセットを50[mm]とした．

Wavefrontアルゴリズムで与える報酬は，可視グラフに沿った大域的最短経路上を通過した場合の，各点からのゴールへのコストに基づいており，ここで想定されている最適経路はFig.5.21に示すものである．

これに対して，ここで与えた動作プリミティブを用いた場合のこの環境における最適経路をFig.5.22に示す．ただし，図中，“F”は自由行程の前進，“L”および“R”は左および右に壁を見る壁沿い動作，“B”は離脱動作を示す．図に示すように，この経路は動作プリミティブが与える拘束に基づいて，Fig.5.21が与える，即時報酬付与方法において想定されている最適経路とは異なるものとなっている．

5.6.4 パラメータ設定

本節でのシミュレーションで用いた学習パラメータを以下に示す．

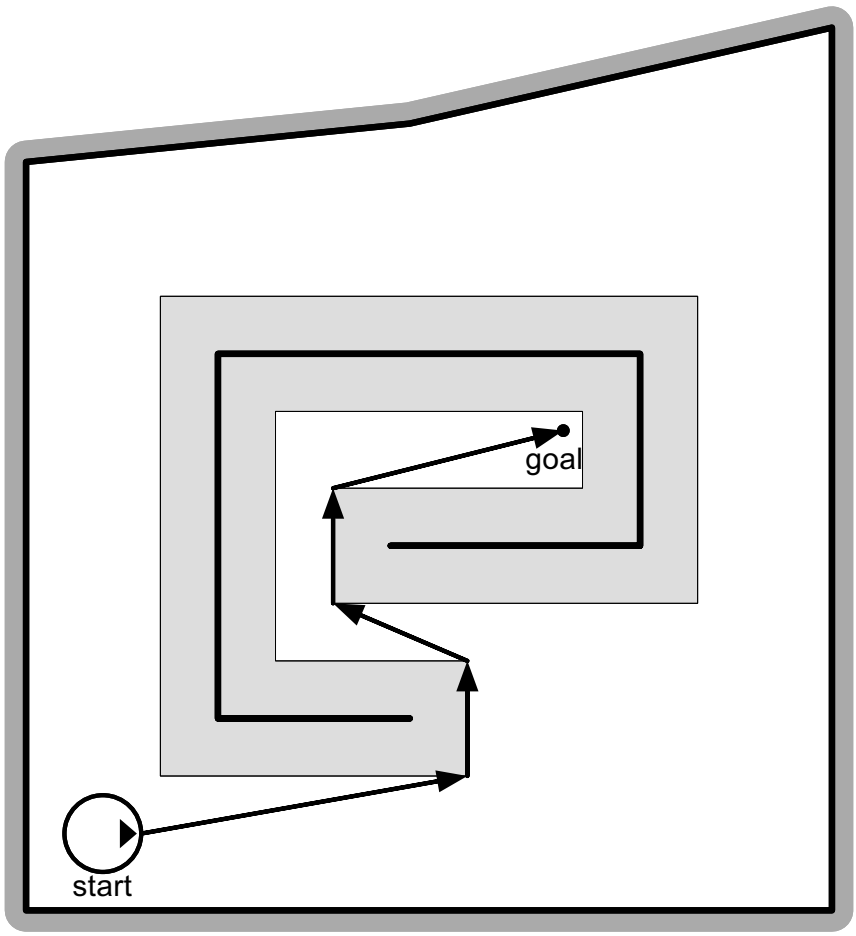


Fig. 5.21: Optimal path based on wavefront algorithm

5.6.5 シミュレーション結果

シミュレーションの結果を Fig.5.23〜5.26に示す．ただし，これらの結果は 10 通りの乱数の種を用いた結果を平均したものである．

Fig.5.23には各試行においてゴール到達までに消費されたステップ数を示す．図に示すとおり，試行 50 程度でおおむね行動は最短ステップ数 6 に収束した．ただし，この段階では状態表現上にインスタンスに対応する報酬のばらつきを持つ状態ノードが存在しており，状態構成が完全に完了するのは 280 試行付近であるため，状態分割に伴って最適動作シーケンスから逸脱する場合がしばしば見られる．

Fig.5.24には，各試行終了時点での状態表現の木構造上のノード数を，Fig.5.25には，各試行終了時点での状態表現木構造の最大深さを示す．行動獲得後も状態分割が行われていることが分かる．

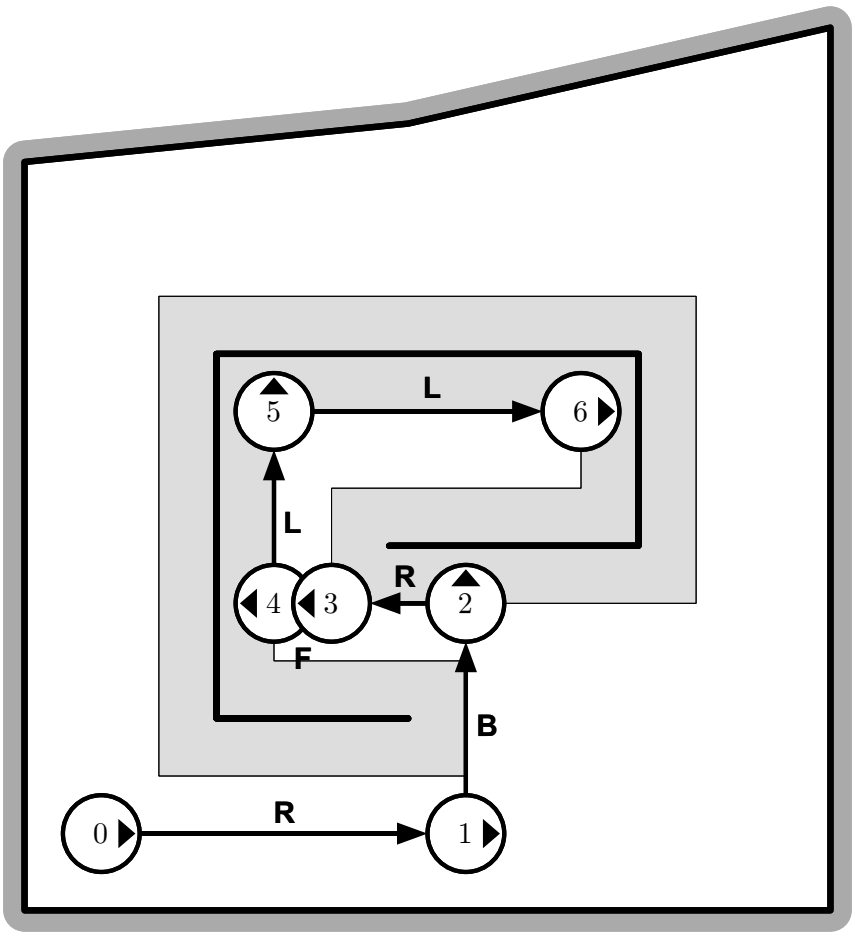


Fig. 5.22: Optimal path based on the motion primitives

Fig.5.26には，各試行完了時点までに行われた観測ベース分割および履歴ベース分割の総回数を示す.

5.6.6 考察

以上に示した結果の通り，Wavefrontアルゴリズムに即した勾配場に基づく即時報酬付与方法によって，提案手法によるナビゲーション行動の獲得が確認された.

このシミュレーション設定では，第5.4節に示したシミュレーションとは異なり，ロボットに与える即時報酬はロボットの動作原理に直接的に立脚したものではない．即ち，ここで与える即時報酬は，ゴールに近づく方向に直線的に進んだ場合の最短経路長に基づいており（Fig.5.21, 5.22），それに対してロボットに可能な動作プリミティブはこのような進路を必ずしも許さない.

Table 5.3: Parameters in simulation

behavior learning related	α	0.1
	$_b$	0.1
state-space construction related	ν	50
	δ	0.25
	ρ	0.75
	$\mathcal{D}_{\mathrm{T}}$	0.6

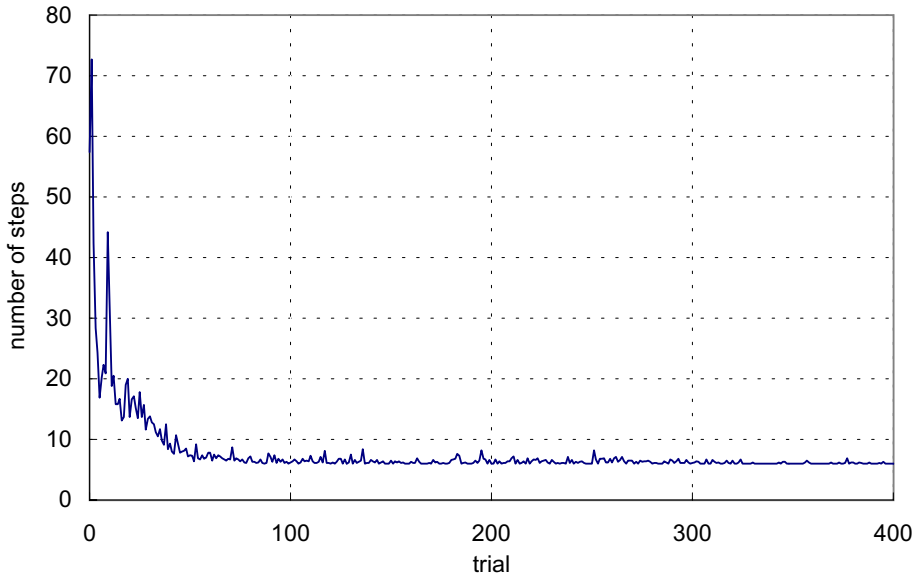


Fig. 5.23: Number of steps per trial

このような前提のもとでのシミュレーションの上で行動獲得が実現されたという結果は, 即時報酬付与方法がエージェントの視点に即したものでない場合においても提案手法が適用可能であることを示している.

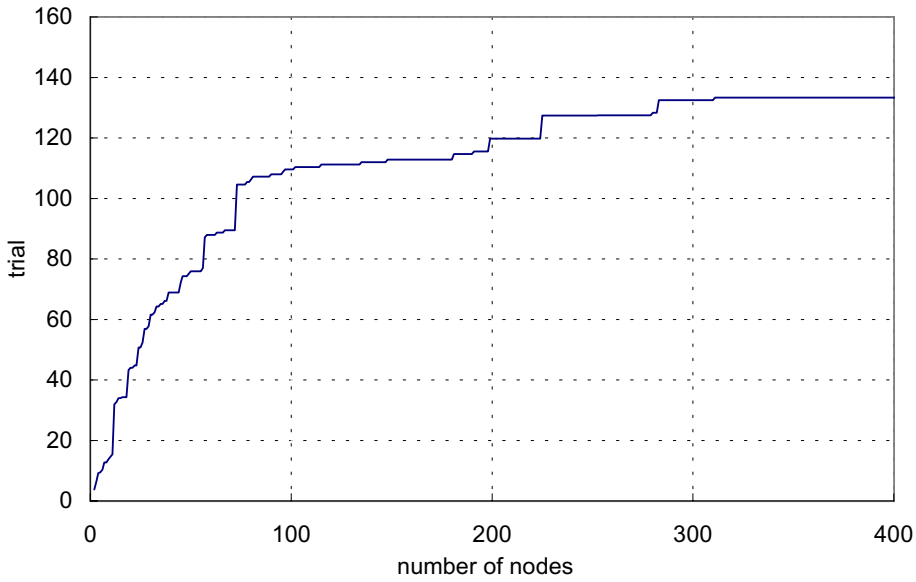


Fig. 5.24: Number of nodes

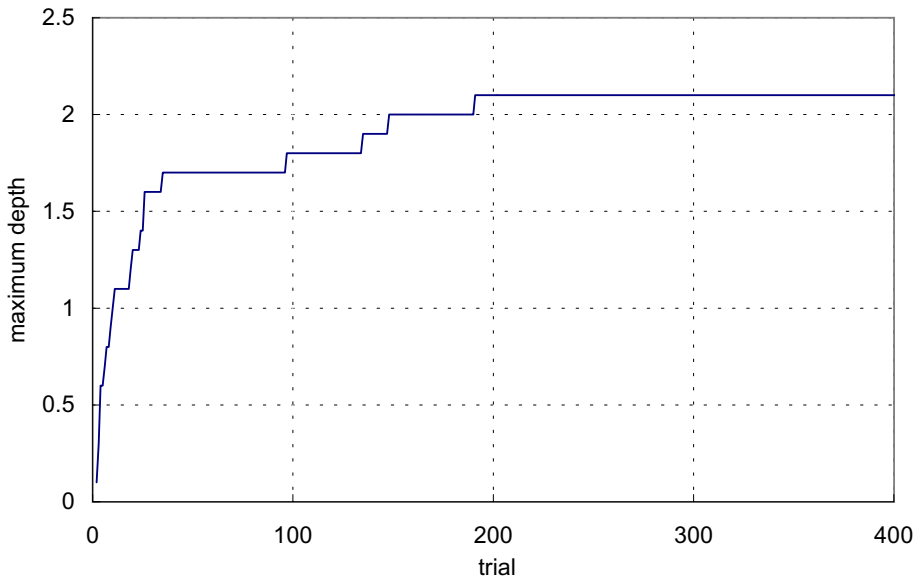


Fig. 5.25: Maximum depth of state-representing tree

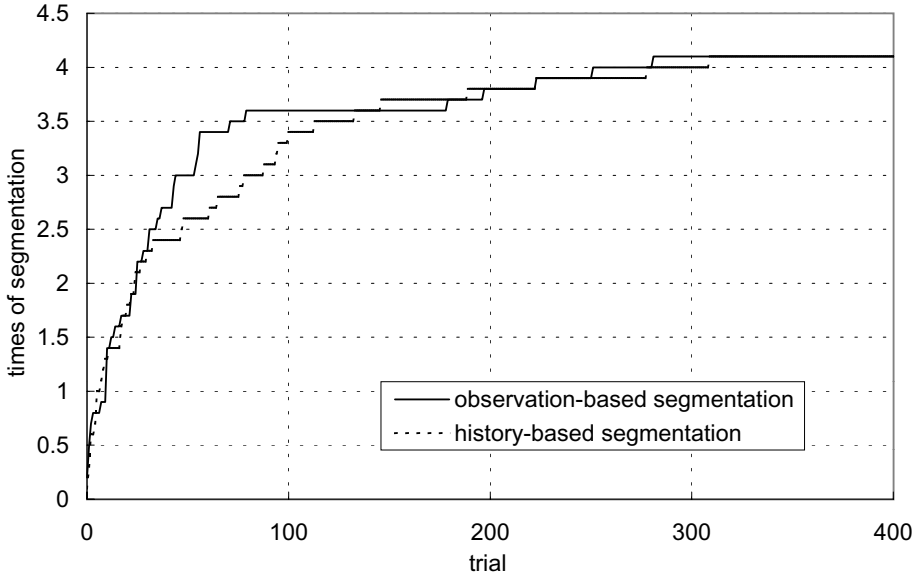


Fig. 5.26: Times of segmentation

5.7 おわりに

本章では，実機ロボットを想定した現実的なシミュレーションによる行動獲得，およびそれにより獲得された行動の実機ロボットへの適用実験について説明した．

実機ロボットにおけるセンサ特性を再現したシミュレーションにおいて，複数タスクに対して正しく行動が獲得され，獲得された行動は実機ロボットにおいても有効に動作したことが確認された．

また，即時報酬を付与する外的な教示者が，ロボットの動作原理とは異なる原理に基づいて報酬を付与した場合についても，行動の獲得が確認された．

第 6 章

考察と評価

6.1 はじめに 120
6.2 単一タスクに対する学習に関して 121
6.2.1 計算量および記憶量 121
6.2.2 環境の性質に対する学習性能の依存性 124
6.2.3 学習パラメータ設定 126
6.2.4 観測ベース分割の意義 129
6.2.5 即時報酬の満たすべき条件 130
6.3 実環境への適用に関して 135
6.3.1 誤差の影響 135
6.4 提案手法の適用可能範囲とその拡張への展望 142
6.4.1 適用可能範囲について 142
6.4.2 適用範囲の拡張への展望 144
6.5 おわりに 146

6.1 はじめに

本章では，本論文において提案した個別の手法および全体に関して，より詳細な考察および議論を行う．

第6.2節では，単一タスクに対して第3 章で提案した状況認識・行動獲得機構の獲得手法に関して，考察・評価を行う．

第6.3節では，提案手法の実世界への適用に関して議論する．

第6.4 節では，本研究で提案した手法の適用が可能な範囲及びその拡大への展望に関する議論を行う．

6.2 単一タスクに対する学習に関して

本節では，第3章において提案した単一タスクに対する状況認識・行動決定機構の獲得手法について，全体的な考察・評価を行う．

6.2.1 計算量および記憶量

本項では，計算量および消費記憶容量について議論する．

以下，記号の定義を以下の通りに定義する：

木構造深さ	d_T
観測レイヤにおける観測ベース分岐の枝の数	b
観測レイヤにおける観測ベース分岐の段数	l
1ノードあたりに格納されているインスタンス数	i
1ノードあたりに格納されている参照ベクトルの数	r

簡単のため，これらの値は木構造の各部分において一定である時を考える．

6.2.1.1 計算量

学習過程の各段階における計算量のオーダを評価する．

状態ノードの探索

観測レイヤ1階層あたりの分岐がl段で，木構造深さがd_Tであるため，根ノードから葉ノードにかけて，該当時刻の観測ベクトルと参照ベクトルの比較を行わなければならない観測ベース分岐の数はld_Tである．

1つの観測ベース分岐に対しては，分岐先のノードそれぞれの参照ベクトルとの距離計算を行うため，br回の計算が必要となる．

従って，探索の計算量オーダは$O(brld_T)$となり，多項式オーダである．

状態ノードの妥当性の検証

状態ノードの分割を行うか否かの判断において，過去にその状態ノードを訪れた際の事例に関して報酬の標準偏差を求める．これに必要な計算量は$O(i)$であり，多項式オーダである．

騙しの検出

騙し測度を計算するに当たっては，全ての事例について，互いの騙し測度 d_{ij} を式3.4に従って求める計算が必要であり，このオーダは $O(i^2)$ であり，多項式オーダである．

観測ベース分割

1つのノードあたりの計算量は，インスタンスの観測値に基づくクラスタリングにおいて，互いの観測値間の距離を計算する $O(i^2)$ の計算量が必要となる．

観測ベース分割は根ノードから葉ノードに渡る各観測レイヤにおける最下位の観測ノードについて再帰的に行われる (Fig.6.1．従って，分割手順が全てのレイヤにおいて行われるとすれば，分割手順が行われる回数は

$$1 + s + s^2 + \cdots + s^{d_T} = \frac{s^{d_T+1} - 1}{s - 1} \tag{6.1}$$

となる．

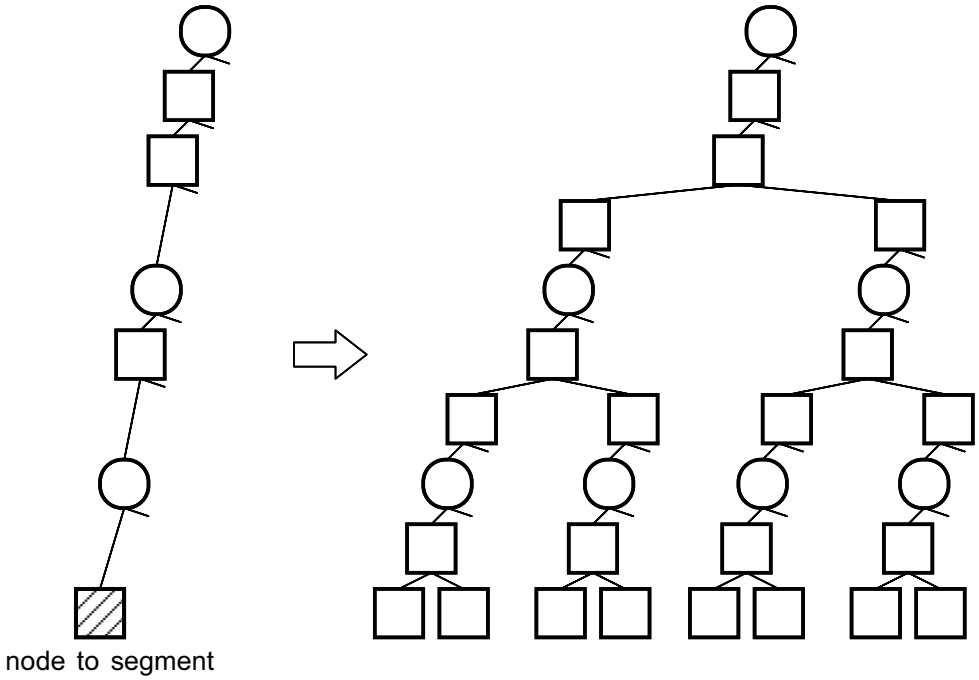


Fig. 6.1: Schematic view of observation-based recursive segmentation

従って，総合した観測ベース分割の計算量は，$O(s^{d_T} i^2)$ となり，指数オーダとなる．

即ち，必要とされる木構造深さに対しては計算量の爆発が起こりうる．

ただし，上に示した計算は最も最悪のケースを示しており，実際的には第3章で示したシミュレーションにおいて，観測ベース分割は多くの場合，1回の分割において1つのノー

ドに対してのみ行われており，計算時間はこの程度の単純さの環境に対しては問題とならなかった．

使用した計算機はIBM互換機で，CPUはPentiumIII，クロック周波数は700MHzであり，オペレーティングシステムはLASER5 Linux 6.0であったが，この計算機環境において，第3章の環境1に対して消費された計算時間は20秒以内であった．

6.2.1.2 消費メモリ

第3章で示した手法において主なメモリ消費はインスタンスベースと木構造である．

インスタンスベースについては，インスタンスベース長に比例した容量が消費される．

木構造については，ノード数，および記録されているインスタンスリンク数に依存するが，インスタンスリンク数は最大で合計インスタンスベース長の数のインスタンスリンクが木構造の状態ノードに割り振られる．

一方，ノード数については，特定の行動ノードの直下の観測レイヤ内の総ノード数は，$\frac{b(b^l-1)}{b-1}$ である．

このレイヤにおける最下位の観測ノード数はb^l，このそれぞれから$|\mathcal{A}|$個の行動ノードが派生しているため，このレイヤから派生する行動ノード数は$B = b^l|\mathcal{A}|$個．従って総ノード数は

$$\frac{B(B^{d_T+1}-1)}{B-1}\frac{b(b^l-1)}{b-1}$$

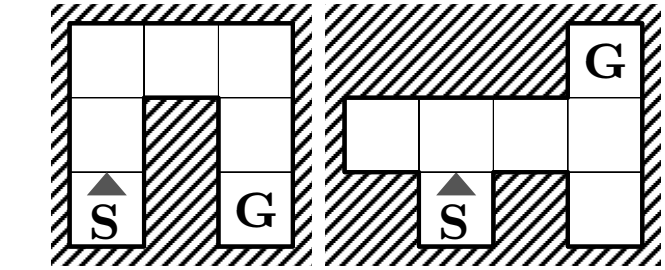
となる．

従って，木構造深さや観測レイヤ1段あたりの観測ベース分割の段数に対してそれぞれ指数オーダの増加を示しており，木構造の複雑性が要求される複雑な環境への適用においては，何らかの対策なしには計算機的に非現実的となる．

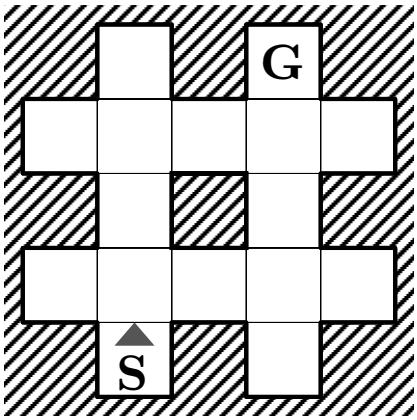
ただし，実際に行ったシミュレーションにおいては，第3章の環境1に対するシミュレーションの結果の例を示すと，収束後の木構造ノード数96に対して，利用したメモリはインスタンスベース容量を含めて10372KB であり，このような単純な環境においては問題が生じないことが分かった．

6.2.2 環境の性質に対する学習性能の依存性

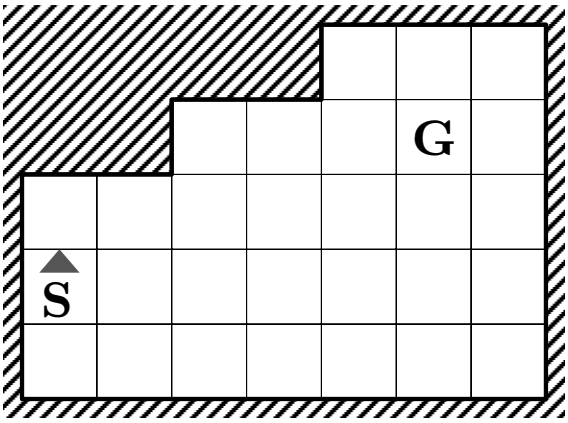
学習性能の環境の性質に対する依存性を調べるため，Fig.6.2に示す 4 つの環境についてのシミュレーション結果について議論する．



Environment A　　Environment B



Environment C



Environment D

Fig. 6.2: Simulation environments

これらの環境の特徴を Table6.1に示す．

Table 6.1: Simulated environments

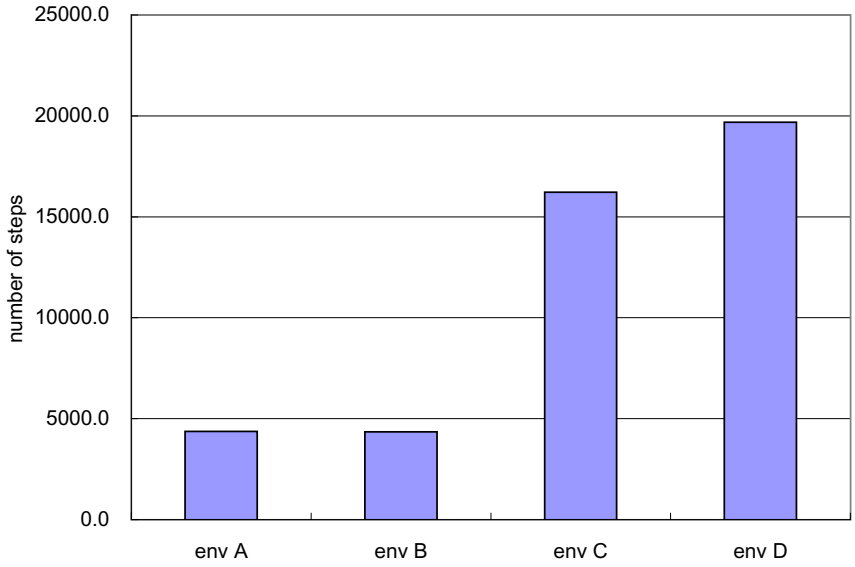
	Env.A	Env.B	Env.C	Env.D
Number of states	24	24	60	112
Number of possible observations	10	10	7	9
Degree of aliasing	2.4	2.4	8.57	12.44
Length of optimal path	8	6	8	9

“Number of states” は環境上のゴール状態を除く状態数であり，“Number of possible observations” はあり得る観測の種類である．前者を後者で割った値として，“Degree of aliasing” には環境の騙し問題の程度を示す測度を求めた．また，“Length of optimal path” は最適経路の長さであり，問題の複雑さの一つのパラメータとなる．

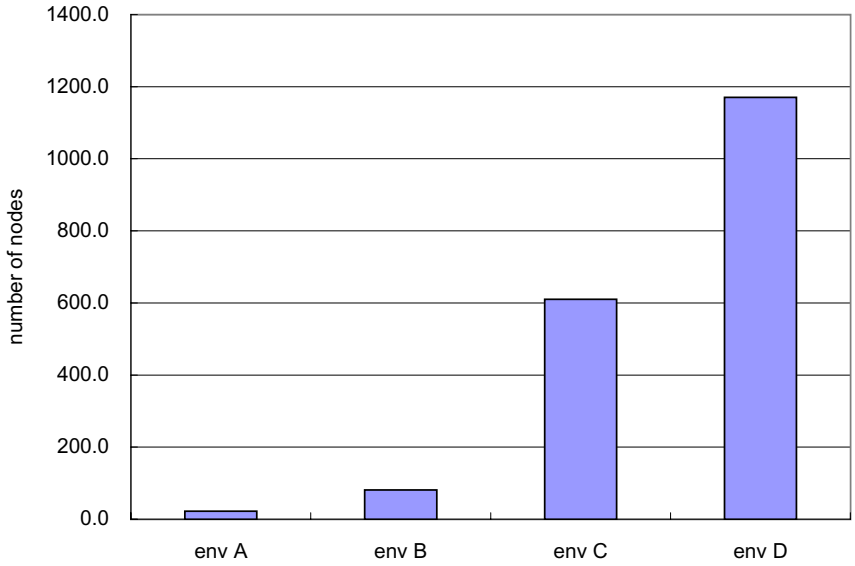
これらの環境に対してシミュレーションを行った結果をFig.6.3，Fig.6.4に示す．

Fig.6.3 (a) は，500 試行が終了する時点までに消費された動作ステップ数である．状態数に応じて増加していることが分かる．(b) は生成された状態表現におけるノード数である．ここでは，環境の複雑化に伴って爆発的にノードが増加することが分かる．

Fig.6.4(c) は木構造の最大深さである．環境の特徴量の少ない環境Dでは，環境Cに比較してより深い木構造を用いて騙し状態の識別が行われている．図 (d) は，学習の過程で行われた状態分割の回数にしめる観測ベース分割の割合であり，環境が大きくなるに従って減少する．これは，環境CやDではより深い履歴を用いた騙し状態の識別が必要となるという事実と，観測ベース分割が再帰的な処理をすることによって一気に多くのノードを分割することができるという事実によると思われる．



(a) Number of steps



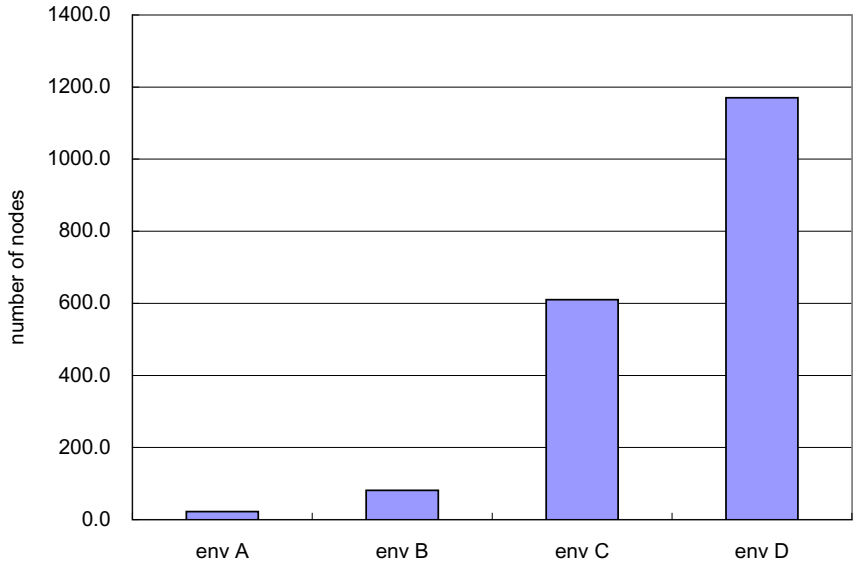
(b) Number of nodes

Fig. 6.3: Comparison of simulation results for different environments

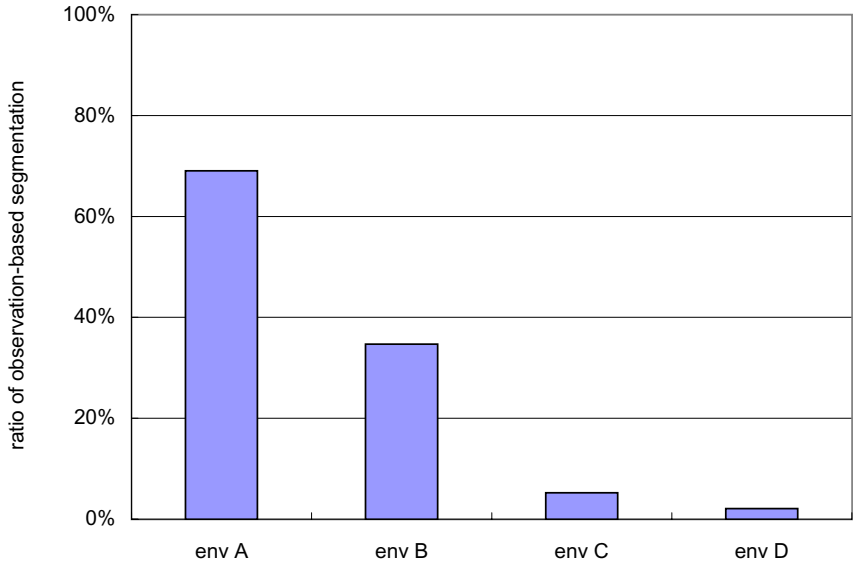
6.2.3 学習パラメータ設定

提案手法では多くの学習パラメータが用いられている．ここではδおよびρについて，これらのパラメータが学習性能に及ぼす影響の大きさについて考察する．

δとは，状態の分割の実施を決定するパラメータであり，状態ノードにおいて過去に得られた報酬の標準偏差がこれを越えているか否かにより分割の実施の有無が決定される．



(c) Maximum depth of tree



(d) Ratio of observation-based segmentation

Fig. 6.4: Comparison of simulation results for different environments

つまり，δが小さいとき，分割は多く行われることになる．

また，ρは騙しに関する閾値であり，状態ノード中のインスタンスの中で，そのインスタンスに対応する騙し測度が閾値を超えるものの割合がδを越えるとき，騙しが存在すると判断して履歴ベース分割を行う．即ち，δが大きくなるほど観測ベース分割を行う割合が大きくなる．

なお，ここで用いた環境は前節の環境ＢおよびＤである．

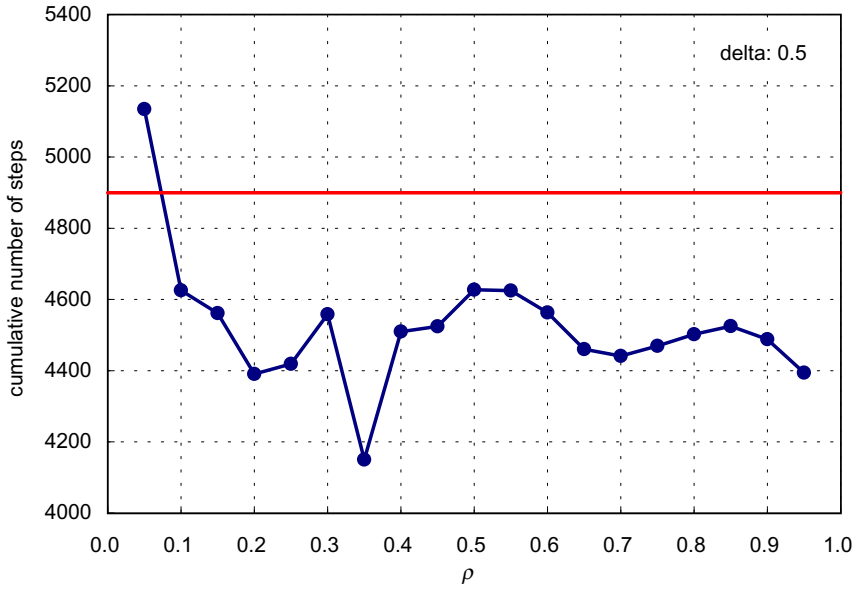


Fig. 6.5: Simulation result of different ρ in environment B

Fig.6.5には，環境 B において異なる ρ に対する学習結果を示す．この図は，各 ρ に対して，学習が収束するまでに消費したステップ数を示している．なお，横線は第3章で行った USM に類似した比較対照手法における結果である．

図から分かるとおり，ρ に対するパフォーマンスの依存性が極めて大きく，適切なパラメータ設定を行わなければ USM 類似法に劣るパフォーマンスさえ見られる．

Fig.6.6には，環境 D に対して ρ および δ を変化させた結果を示す．

この図は，$d = 0.3, 0.5$ の 2 つの条件に対して様々な ρ に対してのシミュレーションを行った際の，1000 試行終了までに消費した総ステップ数を表している．

図から分かるとおり，δ に対しても依存性が大きいことが分かる．

以上に示すとおり，提案手法は，適切なパラメータ設定を行わないと最適なパフォーマンスが得られず，また問題の構造からこれを求めることが困難であるという問題点を含んでいる．

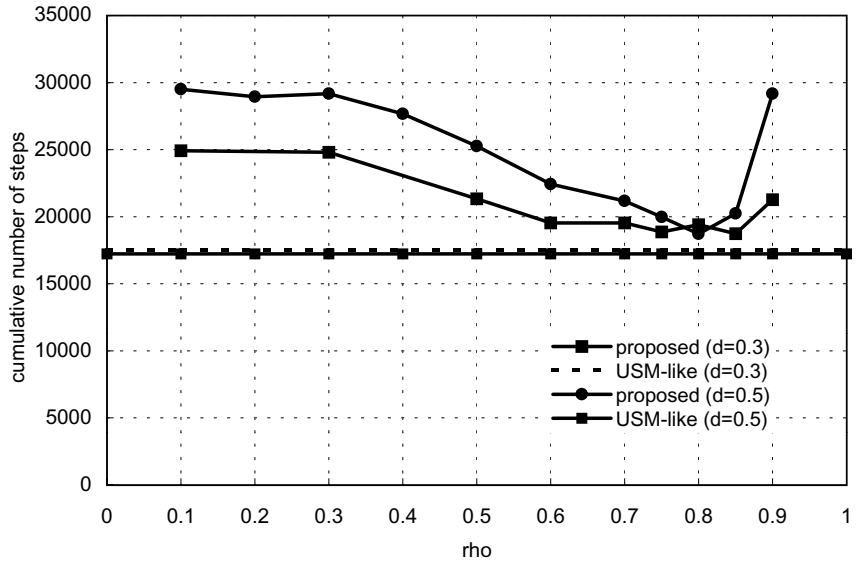


Fig. 6.6: Simulation result of different δ and ρ in environment D

6.2.4 観測ベース分割の意義

状態分割において，観測ベース分割を行うことの意義について考察する．

観測ベース分割を行わず，履歴ベース分割のみで状態を構成した場合，エージェントの状態の規定は過去の動作ステップの履歴のみに基づいて行われる．

この際の結果を，観測ベース分割を行った場合と比較した結果をFig.6.7に示す．

図中，(a) は各試行が終了するまでに消費した総ステップ数，(b) は生成されたノードの数，(c) は木構造深さを示している．

ただし，図中 “d” はδ，“r” はρを表し，$\rho = 1.0$の場合が履歴ベース分割のみによる学習の結果を表している．

図が示すとおり，観測ベース分割を禁止した場合，(a) 学習速度は遅くなり，(b) 状態空間の大きさは大きくなり，そして (c) 木構造深さは大きくなる．

このことから，提案手法における観測ベース分割の意義について考察する．観測ベース分割は状態識別過程において環境からの入力を判断材料として利用することで，環境との相互作用に基づくIF-THENルールを実現する．これにより，必ずしも同様の動作履歴によって到達した状態でなくても，タスク実現に対して同様の意義を持つ状態は汎化され，状態表現での識別プロセスを効率の良いものにすることができる．

6.2.5 即時報酬の満たすべき条件

ここでは，提案手法による認識機構の獲得およびタスク達成行動の獲得のために満たすべき即時報酬の条件について議論する．

6.2.5.1 認識機構獲得のための条件

状態認識機構の構成の目的は，特定の状態ノードにおける特定の動作に対応する即時報酬が一定となることであるから，特定の外的状況における特定の動作に対して与える報酬が常に一定であれば，同様に観測値においても一貫性が成り立つ限りにおいて，観測ベース分割を利用した効率の良い認識機構の獲得が可能である．ただし，ここで言う効率とは，汎化により状態数が抑制され，それにより行動獲得における探索領域が狭くなることで行動学習に必要な試行の回数を少なくできることを示す．

6.2.5.2 行動獲得のための条件

提案手法では，即時報酬に基づく行動獲得を行うため，即時報酬がタスクを達成する動作系列を反映したものである必要がある．即ち，スタート状態から開始してそれぞれの状態において可能な動作のうち最大の報酬を与える動作を選択し続けることでゴール状態に到達することが可能でなければならない．

Fig.6.8に，例を挙げて説明する．図中，“A” から “D” まではそれぞれあり得る外的状態をしめし，それらをつなぐ矢印は可能な動作，矢印に付記した数字はその動作に対応する即時報酬を表す．また，“A” をスタート状態，“D” をゴール状態とする．

図中，(a) では初期状態 A において可能な 2 つの動作のうち，最大の報酬 2 を与える動作を選択することでゴール状態に到達可能であり，この即時報酬によりタスク達成が可能である．

同様に，(b) では，状態 A において最大報酬 1 を与える動作により状態 B に遷移し，同様に状態 C を経由してゴール状態 D に到達することができる．

これに対して (c) では，初期状態 A において最大報酬を与える動作により B に遷移するが，状態 B における最大報酬 0 を与える動作はスタート状態 A への遷移を帰結するため，このような即時報酬のもとでは状態 A と B の間を振動する行動が獲得され，ゴール達成行動を獲得することはできない．

実際に，第 5.4章において示した環境において，Fig.6.8 (c) と同様の状況をシミュレーションにより再現した．このシミュレーションでは，即時報酬を，各状態からゴールまでの直線距離に基づいて与えた．すなわち，動作を実行する前と後におけるそれぞれの状態

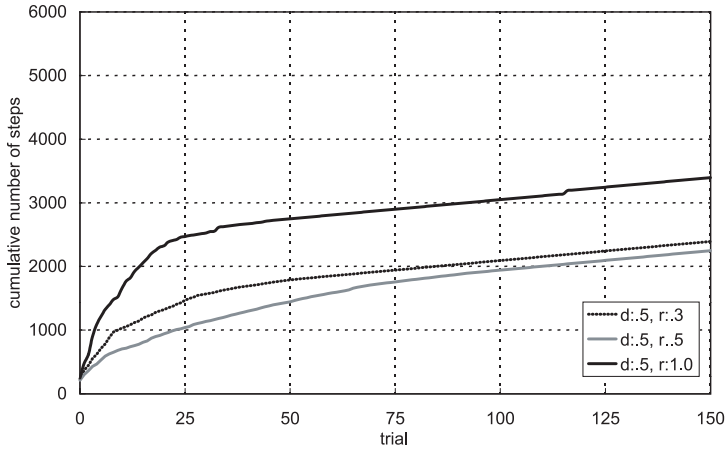
におけるゴール位置までの直線距離を求め，この減少量を定数倍した値を報酬として与えた．Fig.6.9に，シミュレーション環境を示す．図中円は動作プリミティブの停止条件が成立する位置，すなわち状態であり，矢印は可能な動作を示している．また，主な状態からゴールまでの直線距離を付記する数字で示した．このシミュレーションにおけるスタート状態・ゴール状態はS_0，Gで示した．

この設定においては，ロボットはまずスタート状態において最も大きな報酬（従ってゴールまでの距離の減少量）を与える動作として，右下隅の状態へ遷移する壁沿い動作を選択する．同様にして，右上隅の状態に遷移し，更に距離232の状態へ至る．ところがこの状態においては最大報酬を与える動作は右上隅の状態への遷移を表しており，この2点間を往復する動作が最大の報酬を与えるものとなる．実際のシミュレーションでもロボットの行動はこの往復行動に収束し，ゴール到達行動は獲得されなかった．

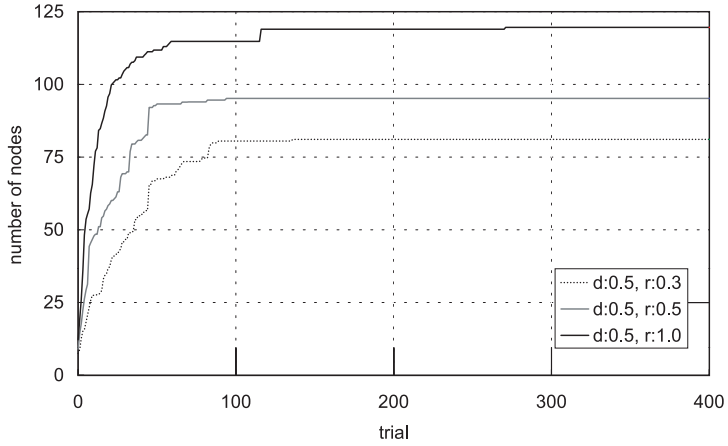
以上の議論における行動獲得の成立のための条件を言い換えれば，ゴールまでに経由される全ての状態において，局所最適解が存在しないことであるといえる．しかしながら，この条件はエージェントの動作原理およびこの動作原理に従った場合のエージェントの行動が既知である場合に初めて利用可能な表現に基づいている．一般的にはこれらの情報はエージェントが実際の行動を行う以前の段階においては未知であり，上の条件を成立させる即時報酬付与方法を予め適切に与えることを保証することはできない．

従って，設計者の立場から考えた場合，この問題に対しては実際にタスクを実行した上でエージェントが獲得する行動が不適切であった場合に，報酬付与方法を変更するか，あるいはエージェントの動作プリミティブを設計し直すという方法が採られなければならない．

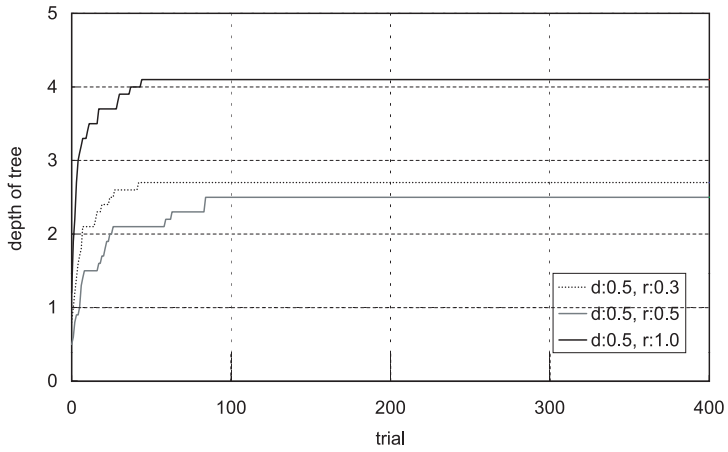
ナビゲーションタスクのように，問題の（エージェントにとって外的な）状態空間に対して局所最適解の存在しない勾配を与えることが容易な問題に関しては，ここでの改善方法としては動作プリミティブの再設計という方法が適切である．即ち，報酬の勾配が局所解を持たないにも関わらず，エージェントの動作プリミティブに従う場合，その動作の結節点間のグラフ構造上で局所解が生じてしまう場合，結節点が粗でありすぎるために，勾配を下る方向の動作を実現するために必要な結節点が現れないと言うことが本質的な問題である．従って，ここでは勾配を下る方向の動作のために必要な結節点を形成するのに適した動作プリミティブに改善するというアプローチが有効である．第5.6.2項で導入した離脱動作は，壁沿い動作の過程で勾配の極小点を結節点する動作の導入によって問題の解決を図った例と言える．



(a) Cumulative number of steps

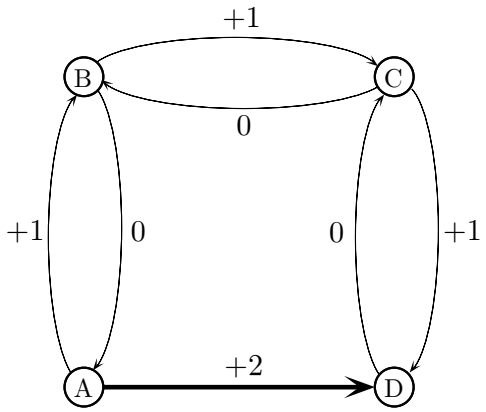


(b) Number of nodes

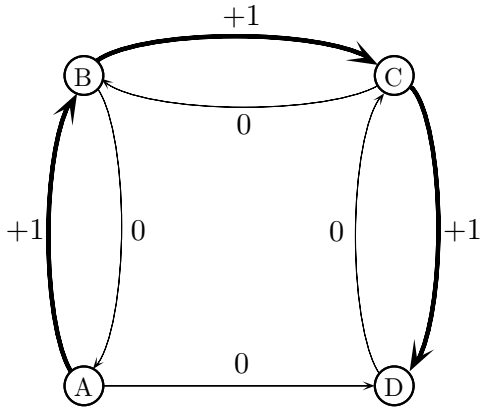


(c) Maximum depth of state-representing tree

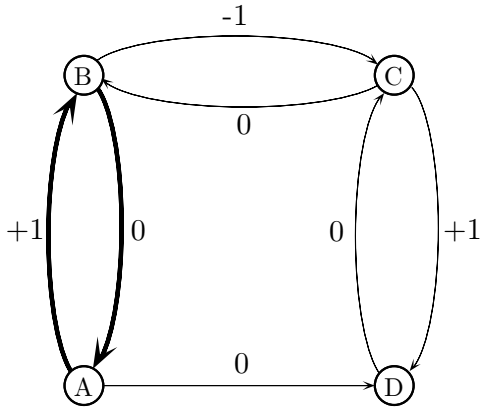
Fig. 6.7: Result of simulation without observation-based segmentation



(a) Case-1



(a) Case-2



(a) Case-3

Fig. 6.8: Condition for acquisition of task-realizing behavior

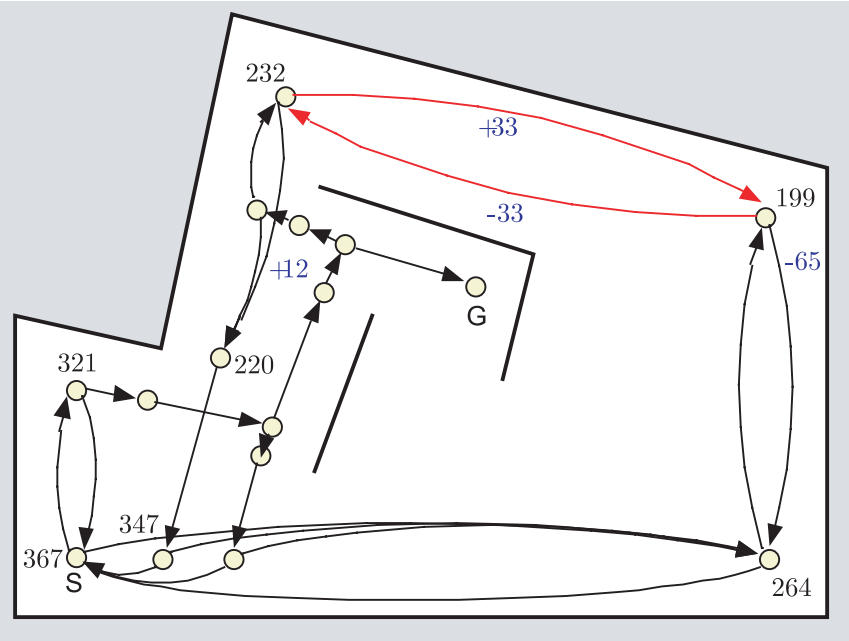


Fig. 6.9: Direct distance from subgoals to goal

6.3 実環境への適用に関して

6.3.1 誤差の影響

本項では，提案手法におけるセンサ，報酬及び動作の誤差に対する影響を，第5.6節で示したシミュレーション環境を用いて評価する．

6.3.1.1 センサ誤差に関して

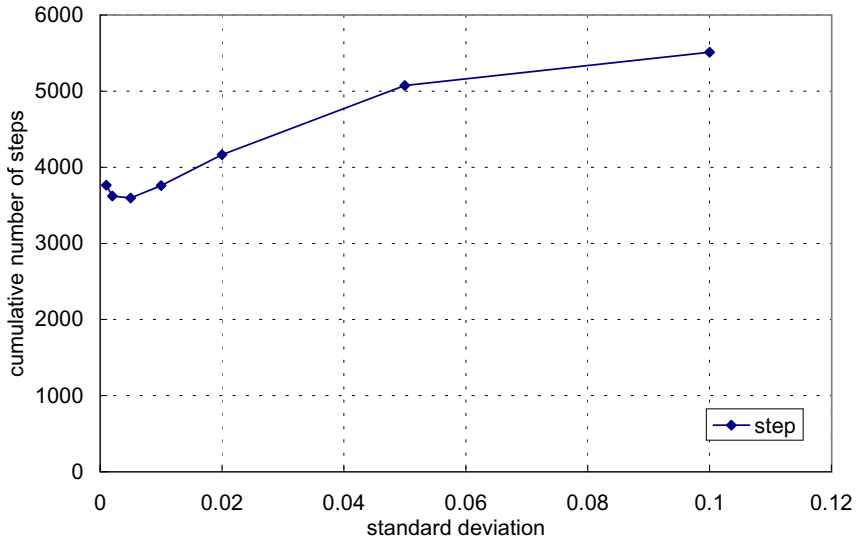
Fig.6.10およびFig.6.11 に，センサ値に正規分布のノイズを印可し，正規分布の標準偏差を変化させた場合の，500 試行の時点での，(a) 累積消費ステップ数，(b) 木構造内ノード数，(c) 木構造最大深さ，(d) 状態分割回数を示す．

ただしセンサ値への誤差は，センサ読みとり値 s に対して，$s\Delta$ の標準偏差を持つ正規分布のノイズを与えた．図中 “standard deviation” と示したのは，Δ の値である．

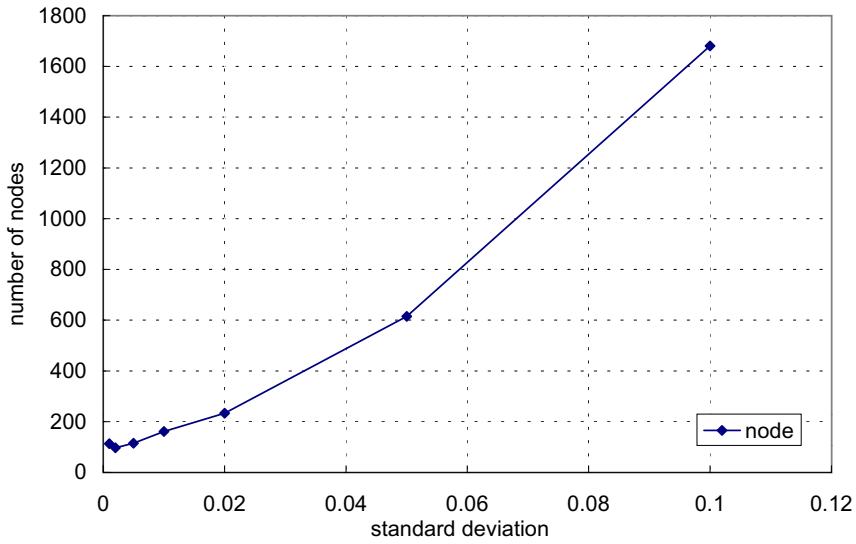
この図に示すように，観測値の誤差が大きくなるとき状態分割回数が増加し，それに伴う状態空間の大規模化に起因して，行動学習に消費する時間が増加して学習効率が低下する．

この原因としては，状態構成において，特定の内的状態に対する報酬の分布を分析する際，特定の外的状態に対応するインスタンスのクラスタが誤差による広がりを持つため，クラスタに重なり合いができてしまい，これに起因して知覚騙しの検出において騙しの測度が増加し，履歴ベース分割が促進されることが考えられる (Fig.6.12)．必要以上の履歴ベース分割が行われた場合，その分割によって生成されたより下層のレイヤに対して，それぞれ観測ベース分割を行う必要が生じ得るため，適切な状態を構成するまでに必要な経験データがより多く必要となる．また，無駄な分割が行われた場合，その分割の結果生じるそれぞれの状態ノードに対して個別に行動学習を行う必要が生じるため，探索的行動がより多く実行され，この過程でさらに分割回数が増加する．

シミュレーションでは，Δ の値は0.1 まで増加させたが，これをさらに増加させた場合，本シミュレーションで用いている動作プリミティブが適切に動作することが不可能となった．これは，動作プリミティブにおけるロボットのモータ指令がセンサフィードバックにより決定されており，センサ値に大きなノイズが加わった場合，動作が不安定となり，本来動作の停止条件が成立しない場所において動作が停止するなどの不具合が生じてしまうためである．すなわち，Kheperaが行動規範型動作プリミティブを用いて動作しうる限度の誤差の割合に対して，提案手法により状況認識機構・行動決定機構を獲得することが可能であること，ただし誤差が大きければ大きいほど学習の効率は低下することが，本シミュレーションにより示されたと言える．



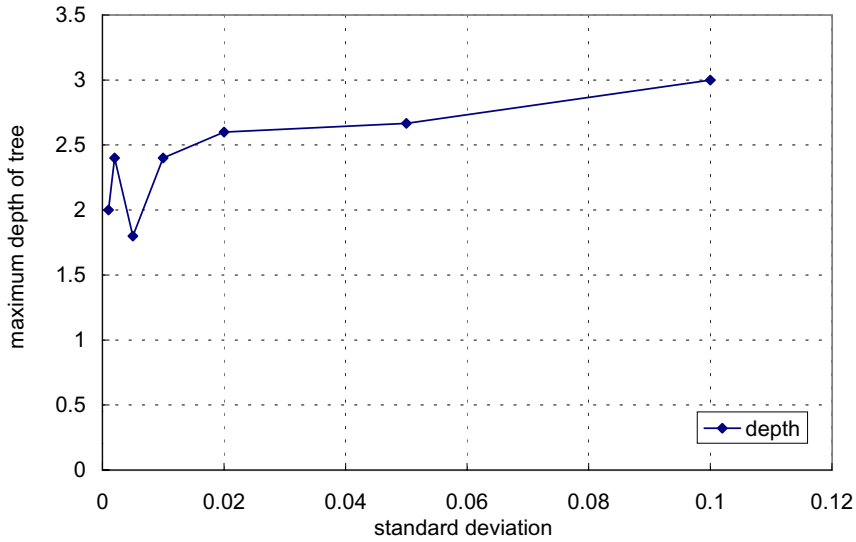
(a) Cumulative number of steps



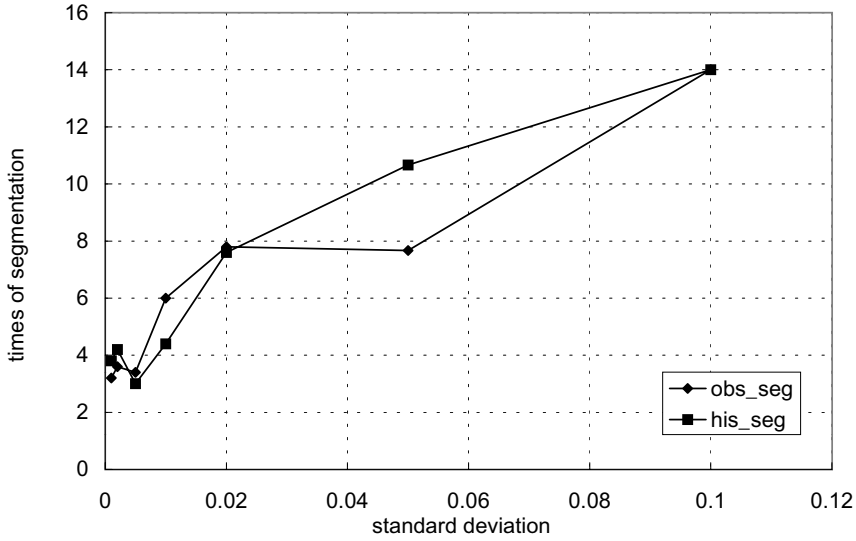
(b) Number of nodes

Fig. 6.10: Effect of sensor noise (1)

なお，実機の Khepera ロボットにおける 1 回のセンシングにおける Δ の実測値は最大でも 2% 程度であるが，本シミュレーションおよび第5章で示した実機実験においては，ロボットのセンシングでは10 回のセンシングを行った結果を平均化して観測値としており，この想定を加味すると実機での Δ は 0.002 程度となる．このことから，実機移動ロボット Khepera におけるセンサ誤差の大きさに対しては，提案手法は十分に対応することができると言える．



(c) Maximum depth of state-representing tree



(d) Times of segmentation

Fig. 6.11: Effect of sensor noise (2)

6.3.1.2 報酬誤差に関して

次に，報酬の誤差の影響を評価する．

ここでは，センサの誤差を 0.0025 に固定し，報酬に対して誤差を与えた．具体的には，報酬値が r であるとき，$r\Delta$ の標準偏差を持つ正規分布のノイズを加えた．

Fig.6.13および Fig.6.14 に，試行 500 の時点での (a) 累積消費ステップ数，(b) 木構造ノー

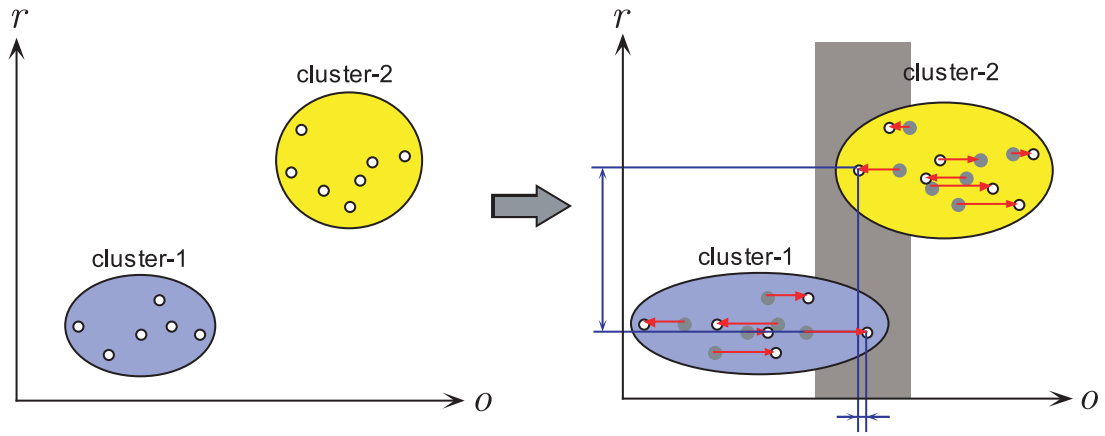


Fig. 6.12: Distribution of instance with sensor noise

ド数，(c) 木構造最大深さ，(d) 状態分割回数を示した．ただし，図中 “standard deviation” は Δ を示す．

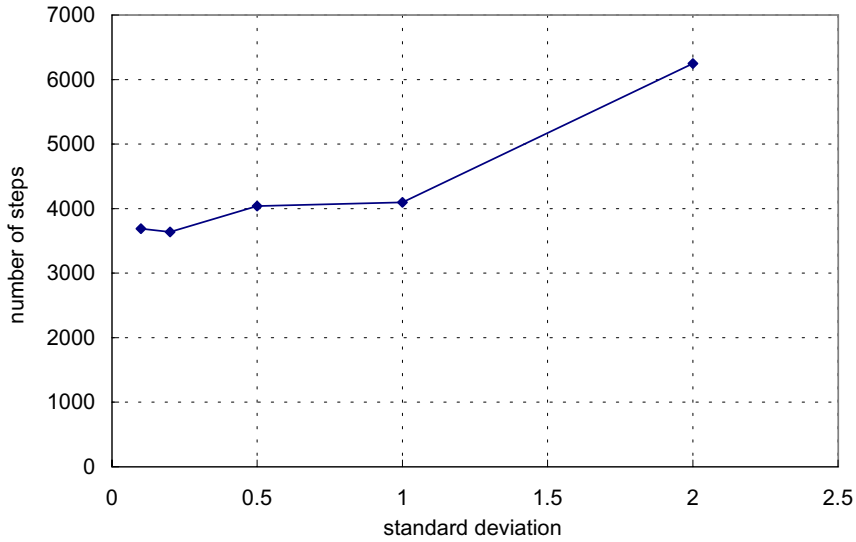
シミュレーション結果から，報酬の誤差が大きくなるほどノード数，木構造深さが増加し，これに伴って消費ステップ数も増加していることが分かる．分割回数については，誤差が増大するほど観測ベース分割が行われる回数が減少し，逆に履歴ベース分割の回数が増大することが分かる．

これは，状態構成の過程での状態分割のためのインスタンスの分析において，特定の状態ノードに対応するインスタンス群のうち，特定の外的状態に対応するクラスタ内のインスタンスに対応する報酬が，誤差に起因して大きくばらついているため，互いに近い観測値に対応するインスタンス間での報酬偏差が大きくなり，結果として履歴ベース分割が実行されることによる (Fig.6.15)．

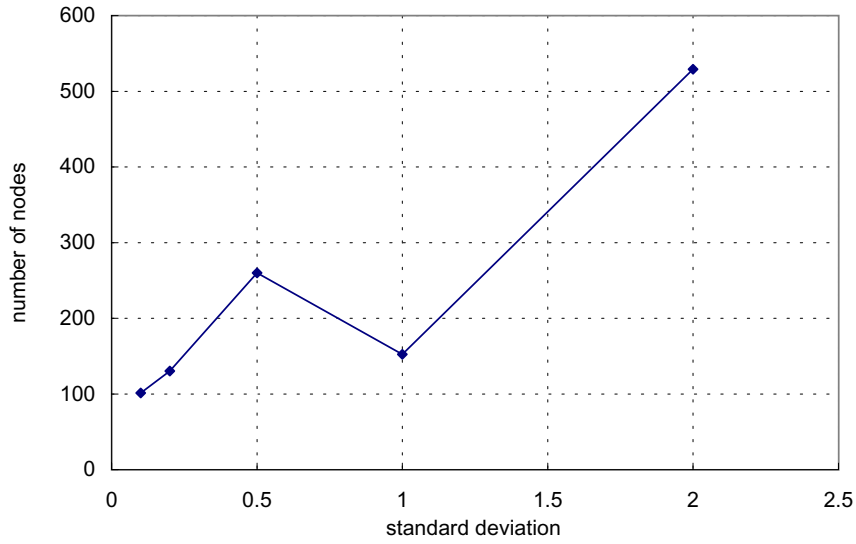
この結果，観測ベース分割が実行されなくなるため，構成される状態表現は観測値をほとんどあるいは全く参照せず，過去の動作系列のみによって状態を識別するものとなる．この結果として状態表現の汎化能力が低下し，エージェントの行動獲得の効率が低下する．

なお，$\Delta = 5.0$ の設定におけるシミュレーションも行ったが，この場合は行動獲得のための Q 値の更新が適切に行われなくなり，全く有意味な行動が現れなかった．

以上の結果から，提案手法は行動獲得が可能な範囲の報酬の誤差に対しては行動の獲得は可能だが，観測値と報酬との関係の分析に基づく状態識別が可能な範囲を超える報酬誤差に対しては，動作系列の全探索という方法で行動獲得が行われるため，学習の効率が低下するということが言えるが，行動獲得の成立は報酬の誤差によって不可能となることはないと言える．



(a) Cumulative number of steps



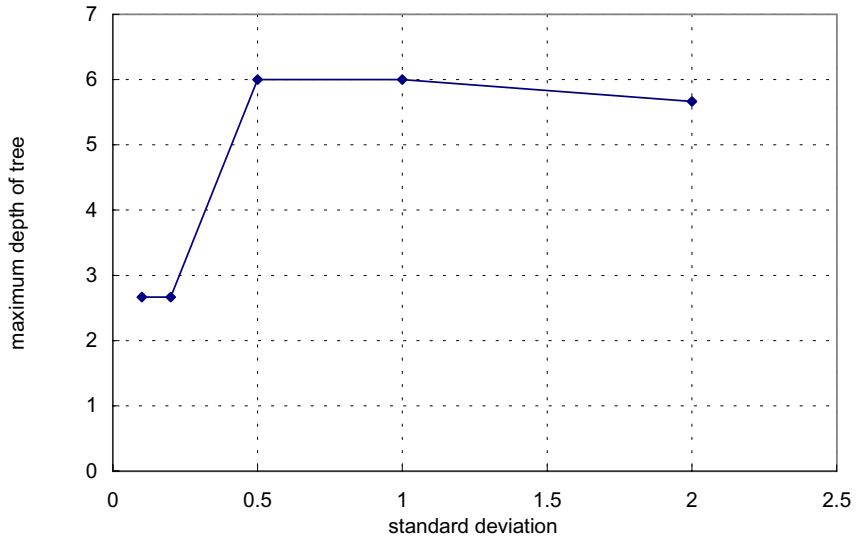
(b) Number of nodes

Fig. 6.13: Effect of reward noise (1)

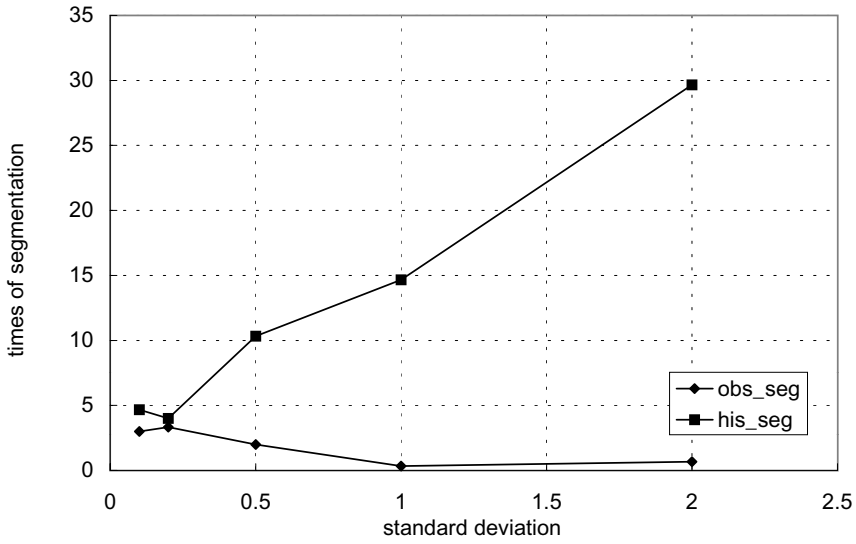
6.3.1.3 動作誤差に関して

第5章では，実ロボットの行動獲得に対して提案手法を適用し，提案法が実世界での動作に耐えうる程度のロバスト性を持つことを示したが，ここでは実験の成功の理由として実機の移動ロボットの動作の確実性が高いことがあった．

即ち，実験において，特定の状態において特定の動作を実行したとき，ロボットは決定論的に同一の次状態へ遷移した．このことは，2.2においても示した，本研究での前提条件



(c) Maximum depth of state-representing tree



(d) Times of segmentation

Fig. 6.14: Effect of reward noise (2)

であるが，一般に実ロボットにおいてはこの前提が成り立つとは限らない．ロボットが同一の動作を試みたとしても，車輪が地面のゴミを踏んでしまったり，人間が通りかかったりなどによってロボットはしばしば動作を失敗したり，異なる結果をもたらしたりする．

このような状況が起こったとき，提案手法ではこれに対応できない．なぜなら，提案手法では状態の差異を同一の動作を行った際に得られる報酬の偏差に基づいて検出しており，実際に同一の状態で同一の動作を行った際に得られる報酬が異なった場合，提案手法

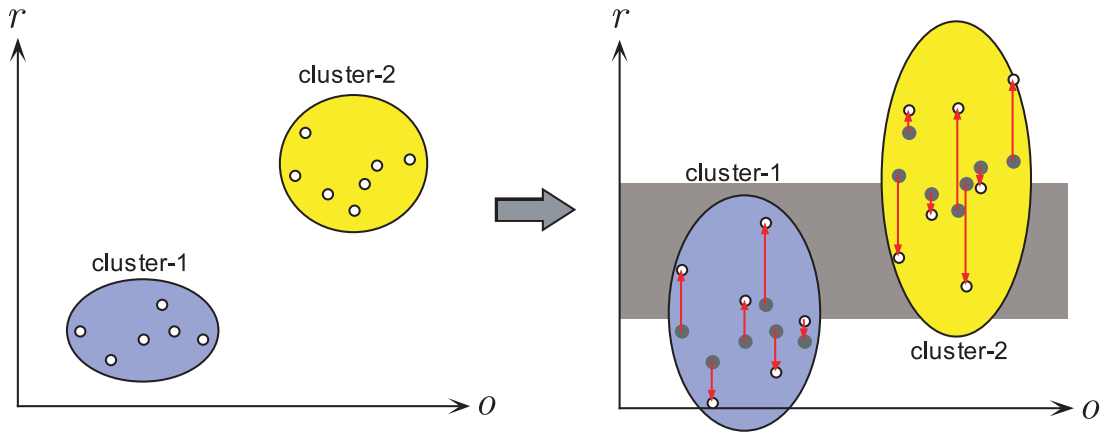


Fig. 6.15: Distribution of instance with reward noise

は遷移元の内的状態が，識別されるべき複数の外的状態を混同しているものと見なしてしまうからである．

従って，行動の不確実性を持つエージェントに対しては，提案手法は現在の実装では適用できず，この問題は提案手法の適用範囲を狭めるものである．

6.3.1.4 まとめ

以上の考察をまとめる．

(1) 観測・報酬における誤差に関しては，実ロボットにおいてあり得る程度の観測の誤差および報酬の誤差によって，学習の効率は低下するが，学習の成立が不可能となることはなく，学習により得られる内的状態表現が効率の悪いものとなることによる学習コストの増加という害がある．ただし，獲得される行動にはこれらの誤差は影響を及ぼさない．

(2) 動作の誤差が動作の結果に対して定性的な偏差をもたらす場合，すなわち外的状態における状態遷移の関係が変化する程度に大きな動作誤差がある場合，提案手法では行動獲得が不可能となる．

6.4 提案手法の適用可能範囲とその拡張への展望

本節では，提案手法の適用が可能な問題領域について議論し，これに基づいて提案手法の様々な問題への拡張に関しての展望を述べる．

6.4.1 適用可能範囲について

提案手法によるエージェントの行動獲得は，以下の 2 つの段階に基づいて行われると考えることができる：

(a) 内的状態空間の構成： 環境との相互作用に基づいて，あり得る外的状態を適切に分類するものとして離散的内的状態の集合を構築する．

(b) 状態・行動対の評価の学習： 獲得された各々の内的状態において行われる各々の動作に対する評価を，タスク実現に対して十分詳細に獲得することにより，タスク達成状態への動作シーケンスを獲得する．

以下，それぞれについて扱いうる問題クラスと扱い得ない問題クラスの区別を議論する．

(a) **内的状態空間の構成について**

内的状態空間の構成は，(1) 環境と動作プリミティブとの関係によってあり得る離散的外的状態を与えられ，(2) この離散的外的状態群をエージェントが適切に分類することで内的状態の集合に変換する，という段階を経て行われる．

このとき，(1) において，エージェントの動作プリミティブが，特定の動作プリミティブの集合によってタスクのスタート状態からゴール状態へ到達可能である必要がある．この条件は，実際にはエージェントがタスク環境と相互作用を実際に行った上で初めて明らかになるものであり，事前にこれを保証することは不可能である．ただし，タスク達成可能な動作プリミティブ系列が存在しないという問題は動作プリミティブ・環境・タスクの設定の段階での問題であり，この条件が満たされない場合にはいかなる行動決定機構を用いても問題は解決できないため，この問題はエージェントの知能の扱いうる領域の外部に存在するものとして，この条件が満たされていることは前提とする．

従って，提案手法によってエージェントが問題を解決しうるか否かは，(2) の段階において，あり得る離散的外的状態群の中で，スタート状態からゴール状態へ至る状態のシーケンスをたどる行動を実現するために十分な外的状態の識別をエージェントが獲得しうる報酬付与方法が採られているか否かに対応する．

例えば，外的状態がFig.6.16のような離散的集合として与えられ，それぞれの外的状態間の遷移が2種類の動作 A_1，A_2 によって図に示した矢印のように与えられるとする．s_1 をスタート状態とし，s_6 をゴール状態とするとき，エージェントがゴール状態に到達するためには，外的状態 s_2 において動作 A_1，外的状態 s_3，s_5 において動作 A_2 を行う必要があり，このためには外的状態 s_2 と外的状態 s_3，s_5 とがエージェントによって異なる内的状態として認識されなければ，それぞれの状態で異なる動作を獲得することができない．即ち，エージェントによってこれら2つの内的状態を識別する状態分割が行われるための条件を報酬付与方法が満たしている必要がある．

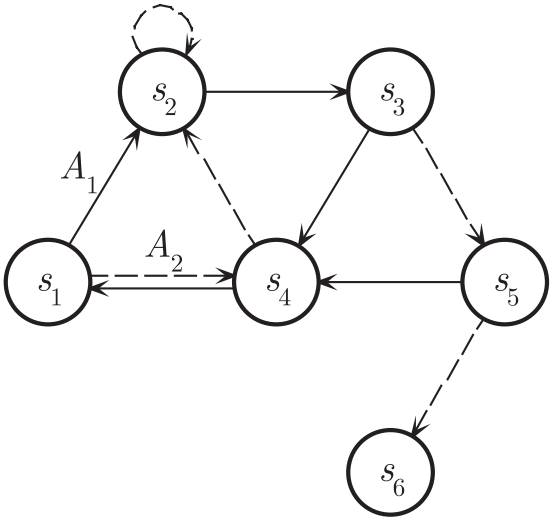


Fig. 6.16: States that have to be dinsinguished

このための条件は，区別されなければならない外的状態群において（図の例では外的状態 s_2 において，および外的状態 s_3 あるいは s_5 において）とりうる動作（図では動作 A_1 あるいは A_2）に対して与えられる報酬が偏差を持ち，それぞれの外的状態において経験された報酬値群を総合したデータについて，その標準偏差が内部状態表現の不適切性として検知される程度に大きいものである必要がある．

(b) **状態・行動対の評価の学習**

内的状態空間の構成が上述の条件に従って構成された上で，エージェントがゴール到達動作シーケンスを獲得するためには，特定のゴール到達可能な動作シーケンスを行った場合の経由する外的状態のそれぞれについて，次に遷移すべき外的状態への状態遷移を与える動作が，その状態においてあり得る動作の中で最大の報酬を与えるものである必要がある．

このような報酬を与えるためには，俯瞰的視点から外的状態空間を眺める報酬付与者

（教示者）が，ゴールへ到達可能なそれぞれの動作が正の価値を持つことを認識している必要がある．このためには，外的状態空間においてゴールへ到達可能な外的状態のシーケンスが少なくとも一つ見いだされている必要がある．このことは，報酬を付与する教示者にとって問題の解決方法が予め分かっていない問題については適切な報酬を与えることができないことを意味しており，教示者の予想の範囲を超える解を提案手法によって導出することができないということを意味している．

この限定は本質的には提案手法が適切な即時報酬に基づいてのみ問題を解決可能であるという事実に起因しており，外部観測者の想定する範囲を超えた解を自律的に見いだすためには，学習手法は遅れを伴う評価（遅延報酬）に対応する必要がある．即ち，遅延報酬に基づく学習においては，評価者は最終的なエージェントのゴール到達のみ認識可能であればよく，ゴール状態へ至る過程をエージェントが自律的に導出することができるのである．

以上の議論から，提案手法の適用可能領域に加えられる限定は本質的に以下の 2 点の問題に起因していると言える：

(1) エージェントの各時点における動作の評価を，ゴール到達のための情報として十分正確な即時報酬として外部から与えられなければならない．

(2) 動作プリミティブと環境との相互作用の結果として与えられる離散的な外的状態のネットワーク構造において，ゴール到達行動の学習に対して適切な報酬付与方法が可能である必要がある．

6.4.2 適用範囲の拡張への展望

本項では，提案手法の適用範囲の拡張への方向性を，(1) 即時報酬に基づく方法として，および (2) 遅延報酬への展望として考察する．

6.4.2.1 即時報酬に基づく方法

第 2.4節において議論した通り，提案手法は即時報酬を与える教示者を想定した教示システムとして考えることができ，俯瞰的視点に基づいて与えられる教示信号によってエージェントの視点に基づく状況認識を実現することができるとき，教示システムとしての優位点を持つといえるが，動作プリミティブの与える拘束条件に起因して第 6.2.5項で示したように，外的な評価を適切に利用しうる状態遷移が不可能となる場合がある．

第 5.6節で示したシミュレーションにおいては，Wavefront アルゴリズムに基づくポテンシャル場に適切にロボットを追従させるために離脱動作を導入した．このように，一つの

解決の方向は，エージェントの動作プリミティブをより粒度の小さいものとして分節化することで，教示者の与える報酬の場に追随しうる動作系列の存在する問題に改良するというものである．ただし，動作の粒度が小さくなれば，その分ゴール状態に到達するために必要な動作数が増加するため，学習で扱うべき問題のサイズが大きくなり学習コストが増加する．更に，動作プリミティブとは本来エージェントの身体性に直接に依存する部分と見なすべきものであり，この方向の解決は，エージェントの知能それ自体に対する改善とは異なるものであると言える．

動作プリミティブに変更を加えずに適切な報酬付与方法を実現するための方法としては，学習開始以前の段階において，教示者（あるいはシステム）に対して適切な即時報酬を与えるのに必要な知識を与える方法が必要となる．即ち，事前にエージェントに可能な動作プリミティブを外的に起動することで，ゴール到達行動として有望な動作シーケンスを発見し，この動作シーケンスを実現するための個別の動作を評価する評価信号を与えることで，エージェントは自らの観測・動作情報に基づいてその動作シーケンスを実現するための状況認識・行動決定機構を獲得するという方法である．

6.4.2.2 遅延報酬への展望

提案手法による遅延報酬への対応を困難としているのは，内的状態表現の構成における状態分割の判断基準として報酬の情報を利用しているという事実である．遅延報酬においては，内的状態群の間の状態遷移に基づいて個別の状態の報酬が算出されなければならないため，遅延報酬を適切に算出するためには，内部状態空間が適切に構成されている必要があるという点で，解決不可能な問題となってしまうのである．

この問題に対する一つの解決の方向性として，状態構成において用いる判断基準として報酬以外の情報を用いるというアプローチが考えられる．報酬を除くと，エージェントが得ることの可能な情報は，自らのセンサから得た観測と自らのアクチュエータを通じて出力した動作の履歴情報となるため，具体的には報酬に代わって観測データを用いるという方法が必要となる．

このようにして，報酬情報を利用せずに状態を構成することができれば，問題を (a) 外部状態空間をエージェントの身体性に立脚して表現される内部状態空間への変換，(b) 得られた内部状態空間における遅延報酬に基づく行動学習，の2段階に分割することが可能である．

6.5 おわりに

本章では，本論文において提案した個別の手法および全体に関して，考察および議論を行った．

第6.2節では，単一タスクに対して第3 章で提案した状況認識・行動獲得機構の獲得手法に関して，考察・評価を行った．具体的には，計算量・記憶量，環境の性質に対する学習性能の依存性，学習パラメータの学習性能への影響，観測ベース分割の意義，および報酬の妥当性に関する議論を行った．

第6.3節では，第5章で行った実世界への適用における，誤差の影響および状態遷移の非決定論性に関して議論した．

第 7 章

結論及び今後の展望

7.1 結 論 148
7.2 展 望 152

7.1 結 論

本論文では，連続的かつ多次元の観測入力を持つエージェントによる部分観測環境下での複数タスク実現のための行動獲得を，外部から与えられる即時報酬に基づいて行うための手法を提案した．また，手法の妥当性をシミュレーションおよび実ロボットを用いた実験により検証した．

人工エージェントが身体性を持ち，身体性を通して環境と相互作用する中でタスク実現を行う上では，エージェントが得た観測データをタスク上での意義に即した状況の規定への写像関係はアプリオリに事前設計する事が困難であり，この関係をエージェント自身が環境との相互作用を通じて定め，こうして得られた状況認識機構に基づいて行動が獲得される必要がある．

ところがここで，エージェントが持つ身体性に起因して，以下の 2 点の問題が生じる．すなわち，(1) エージェントの観測能力が限定されていることから，観測入力はそれ自体ではエージェントの置かれた状況を一意に特定するために必要な Markov 性を失っていて，エージェントにとって環境は部分観測 Markov 決定過程 (POMDP) となっている，(2) エージェントの観測入力は，エージェントの身体性・置かれた環境・扱っているタスクに依存しているため，観測から状況への写像関係は，実際に環境と相互作用を行って初めて明らかになる．

従って，身体性を有するエージェントの行動獲得は，POMDP でありかつ観測入力の解釈方向が所与でない状態から，タスク遂行に必要なだけの Markov 性を伴う状況識別を，観測入力の解釈方法を自律的に規定しながら行いうる状況識別機構を環境との相互作用に基づいて構築するという方法で行われなければならない．

この問題を解決するため，本論文では，各時点における観測入力および動作出力の短期記憶を表現する決定木構造の状態表現を，状態表現上の観測の識別を与える境界を必要に応じて定めながら，身体性・環境・タスクに応じて適切に構築する状況認識機構獲得手法を提案した．

ここでは，自らの状況の識別の不十分性を環境から与えられる即時報酬に基づいて検出し，この不十分性が Markov 性の破れに起因しているのか，あるいは観測入力の識別が粗すぎることに起因しているのかを，過去の経験に対する統計的処理に基づいて適宜判断しながら，状態表現を構築する．

ここで，エージェントは外部から各時点における動作の適切な評価を獲得し，それに基づいて状況認識の機構及び行動決定機構を獲得するという点で，提案手法は一種の教示手法と見なすこともできる．ただし，提案手法の特徴は，教示者がエージェントの内的構造，即ちいかにしてエージェントが自らの状況を認識しているかに関しての知識を用いる必要はなく，外的な視点から眺めたエージェントの大域的状況に対して，外的な視点に基づく

評価を与えるだけで，エージェント側が自律的に自らの状況認識機構を獲得するという点で，教示情報および設計労力の少ない教示システムとしての意義を持つと言える．

極めて知覚騙し問題のおこる頻度の高い通路上グリッド環境におけるナビゲーション問題のシミュレーションにより，ここで提案した状況認識機構獲得手法の妥当性を示した．

また，以上の手法は単一のタスクにおける単一の行動を表現する状態認識機構および行動決定機構を獲得するものであり，適用可能範囲が狭いという問題があったため，これを複数のタスクに適用可能なものとした．

エージェントが身体性を有し，環境の部分観測性と観測入力の解釈方法の構築の必要性が存在するとき，状態の認識機構そのものがタスクに依存したものとなるため，エージェントが現在行うべきタスクを識別するためには，タスクに依存した状態認識機構の上位に位置するメタレベルの識別機構が必要となり，この識別機構により識別されたタスクに対応する適切な状況認識機構・行動決定機構が適用される必要がある．

ところがこのとき，環境にMarkov性が存在するために，タスクを特定するためには一連の行動を伴う認識が必要となる．即ち，一定の行動シーケンスを実行した際に得られる経験データが，過去にそれぞれのタスクを行った際に得られた経験データとどの程度逸脱しているかを検証することで，現在どのタスクを行っているのかを特定する必要がある．

ここで2つの問題が生じる．まず，タスクごとの状況認識機構は，特定のタスク実現行動を表現するものであり，これを獲得する過程で，該当するタスクを実現するための行動シーケンスとは異なる，タスク特定のための行動シーケンスが経験されているとは限らない．更に，状況認識は非Markov的に行われるため，状況認識機構に対応していない行動シーケンスを行った時点からの行動実現がもはや獲得されていないという事態が生じうる．

そこで本論文では，複数のタスク実現行動を獲得するための行動獲得において，上記の問題が生じたことを検出し，それぞれの問題に対応する追加的行動獲得過程を設けるという行動獲得スケジューリング方法を提案した．

知覚騙し問題によって互いに判別しにくい複数のスタート点に置かれたエージェントが，通路上グリッド環境で共通のゴール点を目指すナビゲーションを行うというタスクをシミュレーションし，提案した複数タスク実現行動の獲得方法の妥当性を示した．

以上示した方法の実世界に対する適用可能性を検証するため，実移動ロボットKheperaをモデル化したシミュレーションによる複数タスク実現行動の獲得，および獲得された行動の実機ロボットによる検証を行った．

シミュレーションおよび実験では，8次元の連続的センサ空間を持つ移動ロボットが，行動規範型動作プリミティブに基づくナビゲーションにより複数のスタート点からゴール点を目指すナビゲーションタスクを扱った．シミュレーション結果は，このような状況に置

いてもタスクの識別が正しく行われ，獲得された状態認識機構に基づく最適行動が確認された．

実機実験においては，シミュレーションと実世界との差異に起因する失敗が見られたものの，シミュレーションにおいて獲得された最適行動が実行され，提案手法が実世界でのロボットによる行動獲得に対しても適用可能であることが示された．

また，ここではエージェントに与える即時報酬の算出原理とエージェントの動作原理とが異なる場合についてシミュレーションによる検証を行い，即時報酬を与える教示者の視点とエージェントの視点との差異を提案手法が吸収しうるという結果を示した．

最後に，提案手法の全体に渡って，考察および評価を行った．

単一タスクに対する状況認識・行動決定機構の獲得手法に関しては，計算量及び使用記憶量，学習パラメータ設定の困難性において改善の余地があり，また報酬の妥当性において，限定がある．

また，実世界への応用においては，エージェントの動作が不確定な結果をもたらすことから，状態遷移が非決定論的である場合への対応が必要となるが，現在の提案手法はこの問題に対応することができず，この点でも改善の余地があることを議論した．また，エージェントの動作が定性的な変動を帰結しない場合について，つまり状態遷移に変動がない場合については，センサや報酬における誤差に対してのロバスト性を有することを議論した．

最後に，提案手法の適用可能範囲およびその拡大への展望に関して議論した．提案手法では，即時報酬に基づく行動獲得を扱っているという点に起因して，いくつかの限定がその適用範囲に対して加わっている．特に，動作プリミティブが与える外的状態空間上の構造に対して適切な報酬を与える報酬付与方法を事前に見いだすことは一般には不可能である．これに対しては，動作プリミティブの改変や，動作プリミティブの与える拘束条件に関して教示者（システム）が事前に情報を獲得できるシステムの追加などの対策が考えられる．また，提案手法を内的状態空間の構成の部分と行動獲得の部分に分割し，状態空間構成を報酬に基づかずに行うことが可能となれば，こうして構築された内部状態空間を利用した遅延報酬に基づく行動獲得が可能となる．

以上により，単一タスク実行に対する行動獲得に関して，予め分節化されていない連続的観測空間を持つエージェントが部分観測環境において，身体性を通じた環境との相互作用に基づいてタスク実行に即した状況認識・行動決定機構を，外部から与えられる即時報酬に基づいて構築することによる行動獲得が実現された．

また，複数タスク実行において個別のタスクに対して獲得された状態認識・行動決定機構をタスク識別機構によって適切に使い分けることによって，複数タスクの実行が可能な

行動の獲得が実現された.

更に，これらの手法が実世界で動作する実ロボットによるナビゲーションタスクにおいてもロバストに動作しうることが示された.

7.2 展 望

(1) **状態遷移の不確実性への対応**

提案手法では，身体性を持つエージェントにおける一つの重要な問題である，行動の不確実性を扱っていない．すなわち，特定の状態から他の特定の状態への遷移確率が 0 または 1 の場合のみを扱っている．実際のロボットにおいては，一定の確率で動作を失敗するなどの理由により，この前提は必ずしも成立しない．これに対する対応が必要となる．

(2) **遅延報酬への対応**

提案手法では，エージェントの動作に対する評価が即時報酬という形で外的世界から与えられるという過程をおいており，その点で提案行動獲得手法は学習ではなく教示の方法であるといえる．エージェントの自律性をより高めるためには，エージェントは各時点の行動が将来のある時点に対して及ぼす影響を自律的に計算し，最終的なタスク実現においてのみ得られる報酬に基づいて，それに至る全行動シーケンスを自ら評価するための内的評価機構を備えている必要がある．

このことは，遅延を含む報酬に基づく状態空間構成の困難性を示しており，一つの解決の方向性は，報酬以外の情報に基づいて状態空間を構成した上で，獲得された状態空間の上で遅延報酬に基づく行動獲得を行うという方法である．即ち，状態空間構成をそれぞれの状態・動作の評価に基づいて行うという方法を棄却するという方法論によって，状態空間構成と行動獲得の問題を分割するという方法が考えられる．

謝　辞

本論文は，著者が東京大学大学院 工学系研究科 精密機械工学専攻 新井・湯浅・太田研究室において博士課程の期間に行った研究をまとめたものです．その間，常に熱心にご指導頂きました指導教官の東京大学 大学院 工学系研究科精密機械工学専攻 助教授 太田順先生に心から感謝いたします．扱っている問題における本質を的確に見極めた先生の助言を頂くことが，本研究をまとめる上で非常に助けになりました．

また，本論文をまとめるにあたり，副査をお引き受けくださりました

東京大学 大学院 工学系研究科 精密機械工学専攻	小林郁太郎 先生
東京大学 大学院 工学系研究科 精密機械工学専攻	高増潔 先生
東京大学 大学院 情報理工学研究科 システム情報学専攻	新誠一 先生
東京大学 生産技術研究所	藤井輝夫 先生

には，謹んで感謝の意を表します．

東京大学 大学院 工学系研究科 精密機械工学専攻 教授 新井民夫先生には，研究内容から研究の方法論，研究生活に至るまで，貴重な助言を頂きました．先生に頂いた幅広い視点からのコメントやアドバイスによって，本研究を幅広い視点から評価し直すことができました．

東京大学 大学院 工学系研究科 精密機械工学専攻 助教授 湯浅秀男先生には，主に理論的な側面において貴重なコメントを頂きました．先生の数学的に厳格な理論の構築への姿勢は大変参考になりました．

東京大学 大学院 工学系研究科 精密機械工学専攻 助手 前田雄介氏には，研究生活全般において大変お世話になりました．研究者としてのあり方などを含めて，たくさんの事を教えていただきました．

本研究の理論的枠組みを考案する上で同じ研究グループに所属して協力してくれた現博士課程1年の千葉龍介君には，研究や方法論に関する意味論的議論において，数々の鋭い指摘を頂きました．

その他，研究の合間の気晴らしにつきあってくれた杉正夫君（現博士課程2年），金子慎一郎君（現修士課程2年）をはじめとして，研究室の学生の皆さんにはとてもよくして頂きました．

研究室秘書の井口幸葉様，米岡道江様には，研究を行う上での経理的・事務的手続きに加え，英語の原稿のチェックをして頂いたり，お菓子の差し入れを頂いたりなど，大変お世話になりました．

また，新井先生の奥様の新井雅世様には，常々励ましのお言葉を頂きました．深く感謝いたします．

2000年 2月

参考文献

[1] R. Brooks, “Intelligence without representation,” Artificial Intelligence, **47**, 139/159, (1991)

[2] R. Brooks, “A Robust Layered Control System For A Mobile Robot,” IEEE J. of Robotics and Automation, **RA-2**-1, 14/23, (1986)

[3] A. R. Cassandra, L. P. Kaelbling, M. L. Littman, “Acting Optimally in Partially Observable Stochastic Domains,” Proceedings of 12th National Conference on AI, **2**, 1023/1028, (1994)

[4] D. Jung, A. Zelinsky, “An architecture for distributed cooperative planning in a behavior-based multi-robot system,” Robotics and Autonomous Systems, **26**, 149/174 (1999)

[5] L. P. Kaelbling, M. L. Littman, A. R. Cassandra, “Planning and Acting in Partially Observable Stochastic Domains,” Artificial Intelligence, **101**, 99/134 (1998)

[6] L. Lin, T. M. Mitchell, “ Memory approaches to reinforcement learning in non-Markovian domains,” Technical Report CMU-CS-92-138, Carnegie Mellon University, (1992)

[7] M. Littman, “Memoryless policies: Theoretical limitations and practical results,” Proc. International Conf. on Simulation of Adaptive Behavior: From Animals to Animats 3, MIT Press, 297/305 (1994)

[8] P. Maes, “Behavior-Based Artificial Intelligence,” Proc. of the 2nd Conf. on Simulated and Adaptive Behavior. MIT Press, 2/10, (1993)

[9] P. J. McKerrow, “Introduction to Robotics,” Addison-Wesley, (1991)

[10] R. A. McCallum: “Instance-Based Utile Distinction for Reindorcement Learning with Hidden State,” Proc. 12th International Conf. on Machine Learning, 387/395 (1995)

[11] A. K. McCallum: "Learning to use selective attention and short-term memory in sequential tasks," From Animals to Animats 4: Proc. of 4th International Conf. on Simulation of Adaptive Behavior', The MIT Press, 315/324, (1996)

[12] H. Murao, S. Kitamura, "QLASS: an enhancement of Q-learning to generate state space adaptively," Proc. European Conf. on Artificial Life, (1997)

[13] H. Murao, S. Kitamura, "Q-learning with adaptive state space construction," Learning Robots, vol.1545 of Lecture Notes in Artificial Intelligence, 13/28. Springer-Verlag, (1999)

[14] R. Pfeifer, C. Scheier, "Understanding Intelligence," MIT Press (1999), 邦訳：石黒, 小林, 細田 監訳, "知の創成 —身体性認知科学への招待—," 共立出版, (2001)

[15] S. Singh, T. Jaakkola, M. Jordan, "Learning Without State-Estimation in Partially Observable Markovian Decision Processes," Proc. 11th International Conf. on Machine Learning, 284/292 (1994)

[16] N. Suematsu, A. Hayashi, "A Reinforcement Learning Algorithm in Partially Observable Environments Using Short-Term Memroy," Neural Information Processing Systems, **11**, MIT Press, 1059/1065 (1999)

[17] R. S. Sutton, A. G. Barto, "Reinforcement Learning : An Introduction," MIT Press, (1998), 邦訳：三上, 皆川, "強化学習," 森北出版, (2000)

[18] F. Tanaka, M. Yamamura, "An approach to lifelong reinforcement learning through multiple environments," Proc. of 6th European Workshop on Learning Robots, 92/99 (1997)

[19] S. Thrun, "Monte Carlo POMDPs," Neural Information Processing Systems, **12**, MIT Press, 1064/1070 (2000)

[20] C. J. C. H. Watkins and P. Dayan, "Technical Note : Q-Learning," Machine Learning, **8**, 279/292, (1992)

[21] S. D. Whitehead, L. J. Lin, "Reinforcement learning of non-Markov decision processes," Artificial Intelligence **73**, 271/306, (1995)

[22] K. Yamada, K. Ohkura, M. M. Svinin, K. Ueda, "Adaptive Segmentation of the State Space based on Bayesian Discrimination in Reinforcement Learning," Proc. 6th Int. Symp. on Artificial Life and Robotics, 168/171, (2001)

[23] 臼井, 岩田, 久間, 淺川, "基礎と実践 ニューラルネットワーク," コロナ社, (1995)

[24] 太田，倉林，新井，“知能ロボット入門,” コロナ社，(2001)

[25] 川村, 深尾, 櫟, “ロボットの教示と学習,” 日本ロボット学会誌, **17**-2, 162/165, (1999)

[26] 木村, L. P. Kaelbling, “部分観測マルコフ決定過程下での強化学習,” 人工知能学会誌, **12**-6, 822/830 (1997)

[27] 國吉, ベルトゥーズ, “身体性に基づく相互作用の創発に向けて,” 日本ロボット学会誌，**17**-1, 29/33, (1999)

[28] 酒井, “ロボットの直接教示,” 日本ロボット学会誌, **13**-5, 627/628, (1995)

[29] 椹木, “知識獲得と教示,” 日本ロボット学会誌, **13**-5, 588/591, (1995)

[30] 末松, 林, 李, “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, **13**-3, 404/414, (1998)

[31] 高橋, 浅田, “実ロボットによる行動学習のための状態空間の漸次的構成,” 日本ロボット学会誌, **17**-1, 118/124 (1999)

[32] 高橋, 浅田, “複数の学習器の階層的構築による行動獲得,” 日本ロボット学会誌, **18**-7, 1040/1046 (2000)

[33] 谷, “自己および自己意識の問題への構成論的アプローチ,” 日本ロボット学会誌, **17**-1, 25/28, (1999)

[34] 港, 浅田, “環境の変化に適応する移動ロボットの行動獲得,” 日本ロボット学会誌, **18**-5, 706/712 (2000)

研究業績

著 書

1) Yuji YOSHIMURA, Jun OTA, Kousuke INOUE, Daisuke KURABAYASHI, Tamio ARAI : Iterative Transportation Planning of Multiple Objects by Cooperative Mobile Robots, Distributed Autonomous Robotic Systems 2, Eds. Asama, H., Fukuda, T., Arai, T., Endo, I., Springer, 171/182, (1996.10)

2) Kousuke INOUE, Jun OTA, Tomokazu HIRANO, Daisuke KURABAYASHI, Tamio ARAI : Iterative Transportation by Cooperative Mobile Robots in Unknown Environment, Distributed Autonomous Robotic Systems 3, Eds. Leuth, T., Dillmann, R., Dario, P., Worn, H., Springer, 3/12, (1998.5)

3) Kosuuke INOUE, Jun OTA, Tomokazu HIRANO, Daisuke KURABAYASHI, Tamio ARAI : Iterative Transportation by Cooperative Mobile Robots in Unknown Environment, Intelligent Autonomous Systems, Eds. Kakazu, Y., Wada, M., Sato, T., IOS, 30/37, (1998.6)

査読付き投稿論文

1) 吉村裕司, 太田順, 井上康介, 平野智一, 倉林大輔, 新井民夫: 群ロボットによる多数物体の繰り返し搬送計画, 日本ロボット学会誌, **16**, 4, 499/507, (1998.5)

2) 平野智一, 太田順, 井上康介, 倉林大輔, 新井民夫 : 未知環境における移動ロボット群の経路学習, 日本機械学会論文集C編, **66**, 642, 522/529, (2000.2)

3) 小林祐一, 太田 順, 井上 康介, 新井民夫 : 視覚情報を用いた状態・行動空間の自律的生成, 計測自動制御学会論文集, **36**, 11, 1029/1036 (2000.11)

4) 太田順, 新井 民夫, 井上 康介, 千葉 龍介, 平野 智一, 前田雄介 : コンベア搭載型AGVの協調による物体搬送システム, 日本機械学会論文集C編, **67**, 658, 1905/1911 (2001.6)

5) 井上康介, 太田順, 新井民夫 : 連続的観測空間を持つエージェントによる部分観測環境における自律的状態空間構成, 計測自動制御学会論文集, (投稿中)

6) 井上康介, 太田順, 新井民夫：部分観測環境における複数タスクに対する行動獲得, 計測自動制御学会論文集, (投稿中)

査読付き講演論文

1) Yuichi Kobayashi, Jun Ota, Kousuke Inoue, Tamio Arai : State and Action Space Construction Using Vision Information, Proc. 1999 IEEE Int. Conf. Systems, Man, and Cybernetics, 447/452, (1999.10)

2) Kousuke INOUE, Jun OTA, Tomohiko KATAYAMA, Tamio ARAI : Acceleration of Reinforcement Learning by a Mobile Robot using Generalized Rules, Proc. 2000 IEEE/RSJ Int. Conf. Intelligent Robots and Systems, 885/890, (2000.4)

3) Jun Ota, Tamio Arai, Kousuke Inoue, Ryousuke Chiba, Tomokazu Hirano : Flexible Transport System by Cooperation of Conveyer-Loaded AGVs, Proc. 2000 IEEE Int. Conf. Robotics and Automation, 1144/1150 (2000.10)

4) Kousuke Inoue, Jun Ota, Tamio Arai : Autonomous State-Space Construction in POMDP with Continuous Observation Space, Proc. 4th IFAC Symp. Intelligent Autonomous Vehicles, 255/260 (2001.9)

口頭発表

1) 太田順, 吉村裕司, 吉田英一, 倉林大輔, 井上康介, 新井民夫: 群ロボットによる多数物体の搬送計画に関する研究 (第2報：階層型協調搬送アルゴリズムの提案), 1995年度精密工学会秋季大会学術講演論文集, 479/480, (1995.9)

2) 太田順, 吉村裕司, 吉田英一, 倉林大輔, 井上康介, 新井民夫：群ロボットによる多数物体の搬送計画 (第1報：階層型搬送アルゴリズムの提案), 第13回日本ロボット学会学術講演会予稿集, 861/862, (1995.11)

3) 吉村裕司, 太田順, 吉田英一, 倉林大輔, 井上康介, 新井民夫：群ロボットによる多数物体の搬送計画 (第2報：群ロボットの経路生成法), 第13回日本ロボット学会学術講演会予稿集, 865/866, (1995.11)

4) 太田順, 吉村裕司, 井上康介, 新井民夫：移動ロボット群による多数物体の繰り返し搬送, 第8回自律分散システム・シンポジウム資料, 343/348, (1996.1)

5) 太田順, 吉村裕司, 井上康介, 新井民夫：群ロボットによる多数物体の搬送計画, 日本機械学会ロボティクス・メカトロニクス講演会'96 講演論文集 (ROBOMEC'96), 497/500 (1996.6)

6) 井上康介, 太田順, 吉村裕司, 平野智一, 新井民夫：群ロボットによる多数物体の搬送

計画 (第3報：近接覚センサを用いたロボット群による経路学習), 第14回日本ロボット学会学術講演会予稿集, 663/664, (1996.11)

7) 井上康介, 太田順, 吉村裕司, 平野智一, 新井民夫：未知環境における群ロボットによる繰返し搬送作業における動作計画, 第9回自律分散システム・シンポジウム資料, 25/28, (1997.1)

8) 平野 智一, 太田 順, 井上 康介, 吉村 裕司, 倉林 大輔, 新井 民夫：未知環境における複数台移動ロボットの経路学習, 1997年度精密工学会春季大会学術講演会講演論文集,569/570, (1997.3)

9) 平野智一, 井上康介, 太田順, 倉林大輔, 新井民夫：群移動ロボットによる多数物体の搬送計画 (第4報：未知環境における複数台移動ロボットの経路計画), 第15回日本ロボット学会学術講演会予稿集, 885/886, (1997.9)

10) 井上康介, 太田順, 平野智一, 倉林大輔, 新井民夫：群ロボットによる多数物体の搬送計画 (第5報：未知環境における繰り返し搬送計画), 第15回日本ロボット学会学術講演会予稿集, 903/904, (1997.9)

11) 小林 祐一, 井上 康介, 相山 康道, 太田順, 新井 民夫：自己組織型ハンドアイシステム, 日本機械学会ロボティクス・メカトロニクス講演会'98講演論文集, 1CII3-3, (1998.6)

12) 小林 祐一, 井上康介, 相山 康道, 太田 順, 新井 民夫：押し操作における視覚情報からの状態空間の自律的生成, 第16回日本ロボット学会学術講演会予稿集, 415/416, (1998.9)

13) 井上康介, 太田順, 千葉龍介, 新井民夫：部分観測環境下における強化学習による移動ロボットの行動獲得, 第11回自律分散システム・シンポジウム, 271/274, (1999.1)

14) 小林 祐一, 井上 康介, 太田 順, 新井 民夫：視覚・接触情報を用いた状態・行動空間の自律的生成, 第11回自律分散システム・シンポジウム資料, 275/280, (1999.1)

15) 平野 智一, 太田 順, 井上 康介, 新井 民夫：ベルトコンベア搭載型AGVを用いた柔軟な搬送システム, 1999年度精密工学会春季大会学術講演会講演論文集, 31, (1999.3)

16) 千葉龍介, 太田順, 井上康介, 小林祐一, 新井民夫：部分観測Markov決定過程における自律的状態分割による移動ロボットの強化学習, 日本機械学会ロボティクス・メカトロニクス講演会'99, 2A1-27-041, (1999.6)

17) 太田 順, 平野 智一, 井上 康介, 新井民夫, 千葉 龍介：ベルトコンベア搭載型AGVを用いた搬送システム, 第17回日本ロボット学会学術講演会予稿集, 265/266, (1999.9)

18) 小林 祐一, 太田 順, 湯浅 秀男, 井上 康介, 新井 民夫：脚型ロボットにおける視覚情報と蹴球動作の関係の獲得, 第17回日本ロボット学会学術講演会予稿集, 1003/1004, (1999.9)

19) 井上 康介, 太田 順, 小林 祐一, 新井 民夫：部分観測環境下における自律的状態分割

による強化学習, 第12回自律分散システム・シンポジウム資料, 223/226, (2000.1)

20) 井上康介, 片山朋彦, 太田順, 新井民夫：汎化ルールによる強化学習の加速, 日本機械学会ロボティクス・メカトロニクス講演会'00, 2P1-33-039, (2000.5)

21) 千葉 龍介, 太田 順, 井上 康介, 新井 民夫： AGV群の走行経路と行動計画の統合的設計, 日本機械学会ロボティクス・メカトロニクス講演会'00講演論文集, 2P1-33-039, (2000.5)

22) 井上 康介, 太田 順, 湯浅 秀男, 新井 民夫： 局所情報の利用に基づく強化学習による移動ロボットのナビゲーションの学 習, 第18回日本ロボット学会学術講演会予稿集, 1275/1276, (2000.9)

23) 井上 康介, 太田 順, 小林 祐一, 湯浅 秀男, 新井 民夫：局所センサ入力に基づく移動ロボットのナビゲーション行動の学習, 第13回自律分散システム・シンポジウム資料, 379/382, (2001.1)

24) 井上 康介, 太田 順, 新井 民夫：連続的かつ多次元の観測空間を持つエージェントによる部分観測環境における自律的状態空間構成, 第19回日本ロボット学会学術講演会予稿集, 87/88 (2001.9)

訳 書

R. Pfeifer, C. Scheier 著, 石黒章夫, 小林宏, 細田耕 監訳, 知の創成 —身体性認知科学への招待—, (原著：R. Pfeifer, C. Scheier: Understanding Intelligence, MIT Press, (1999)), 第4章, 共立出版, (2001.11)

受 賞

平成12年度 IMS (Intelligent Manufacturing Systems) 論文賞 (論文名：Flexible Transport System by Cooperation of Conveyer-Loaded AGVs) (2000.11)