平成 13 年度

学士学位論文

# 階層型強化学習の RoboCup エージェントへの適用

## Application of Hierarchical Reinforcement Learning for RoboCup Agents

1020319　日野 慎一

指導教員　Ruck Thawonmas

2002 年 2 月 8 日

高知工科大学 情報システム工学科

# 要 旨

## 階層型強化学習の
## RoboCup エージェントへの適用

日野 慎一

本研究は，階層型強化学習を RoboCup エージェントへ適用し，その有効性を検証する．本稿では，階層型強化学習と従来の強化学習の収束性の違いを迷路問題にて確認し，その後，階層化強化学習を強化学習の適用が難しいとされている RoboCup エージェント・ゴールキーパへの実装を試みる．

**キーワード**　強化学習，階層型強化学習，RoboCup

# Abstract

## Application of Hierarchical Reinforcement Learning for RoboCup Agents

Shinichi Hino

This study are applying Hierarchical Reinforcement Learning to RoboCup agents, and verifying the agents validity. In this paper, the convergent difference is checked between Hierarchical Reinforcement Learning and Convertional Reinforcement Learning in maze problem. And then, the Hierarchical Reinforcement Learning is tried to mount on RoboCup goalkeeper agent. It is said that mounting Convertional Reinforcement Learning on RoboCup goalkeeper agent is difficult.

***key words*** Reinforcement Learning, Hierarchical Reinforcement Learning, RoboCup

# 目次

# 図目次

# 表目次

# 第 1 章

# はじめに

強化学習は目標までの行動を試行錯誤して見つけて出してゆく試行錯誤型の機械学習である．しかし，RoboCup のような状態空間が非常に大きな問題では，試行錯誤回数が多くなるため，学習が収束しにくく，学習までに莫大な時間を要してしまうという欠点がある．

そこで，そのような大きな問題に対し，強化学習を階層化し問題を分割することにより状態空間を小さくすることで学習の高速化を計ることが試みられている．

本研究では，階層化した強化学習を RoboCup エージェントへ適用し，その有効性を検証する．

# 第 2 章

# RoboCup について

本章では，はじめに RoboCup 全体について述べる．その後，本研究で使用されるシミュレーションリーグについて詳しく説明する．

## 2.1 RoboCup とは

RoboCup は, 人工知能と知的ロボットに関する研究を促進させるためのものである．自律移動ロボットによるサッカーを題材として，日本の研究者たちによって 1995 年に提唱された．現在では, サッカーだけではなく，大規模災害へのロボットの応用としての取り組みとして RoboCup レスキュー，次世代の技術の技術の担い手を育てるジュニアなどが組織されている．

RoboCup サッカーには, 現在 4 つのリーグがある．

- シミュレーションリーグ

  コンピュータ上の仮想フィールド上で, 1 チーム 11 体のソフトウェアエージェント同士によって行なわれるリーグ．RoboCup サッカーの中では一番古くから存在するリーグで，一番洗練されたチームプレイをする．
- 小型リーグ

  卓球台とほぼ同じ大きさのフィールドで, 直径 18cm 以内に入る小さなロボットが 5 台以 内でチームを組み，オレンジ色のゴルフボールを使って対戦するリーグ.
- 中型リーグ

  卓球台 9 枚分の大きさのフィールドで，直径 50cm 以内に入るロボット 4 台でチームを

組み，オレンジ色のフットサルのボールを使って対戦するリーグ.

- Sony 四脚ロボットリーグ

  SONY のエンターテインメントロボットによる 3 対 3 で行うリーグ．このリーグは，選抜されたチームに貸与されたロボットで競技を行うため，一般参加は出来ない.

さらに，今年 (2002 年) にはヒューマノイドリーグが加わる予定になっている.

毎年 RoboCup では, 国内大会と世界大会が行なわれる．ここで, 各チームの研究・開発されたロボット同士が勝負をしている.

RoboCup サッカーの最終目標は，「2050 年までに，ヒューマノイド型ロボットで，人間のサッカーの世界チャンピオンに公式ルールで勝利する」ことである [1].

## 2.2 RoboCup シミュレーションリーグについて

次に本研究で使用する，シミュレーションリーグについて述べる.

### 2.2.1 RoboCup シミュレーションリーグ概要

RoboCup サッカーシミュレーションリーグは，RoboCup の中では 1 番古くから存在するリーグである．そのため，オフサイドトラップなどの洗練されたチームプレイをするに至っている.

それだけではなく，現在では複数の大学・高等専門学校でプログラム演習の教材として，使用されている．本学でも 1999 年から RoboCup を使用した実験科目を行っている．(第 3 章参照)

そのシミュレーションの仕組みは，サーバ・クライアント方式を採用している．この方式を簡単にまとめた図を図 2.1 に示す.

サッカークライアントとサッカーサーバの間は, UDP/IP 通信によって通信が行なわれる．よって, サッカークライアントを作成する場合に使用するプログラム言語は, UDP/IP 通信をサポートする言語であれば何でも良い.

また, 1 つのクライアントは原則として, 1 つのエージェントのみを制御しなければならない. 1 つのクライアントが複数のエージェントの情報を得て, 行動を集中制御することは許されていない.

そして, サッカーサーバによってシミュレートされた結果は, サッカーモニタと呼ばれる表示プログラムによって表示される.

現在 (2001 年 1 月), サッカーサーバの最新版は SoccerServer8.02 である [2].

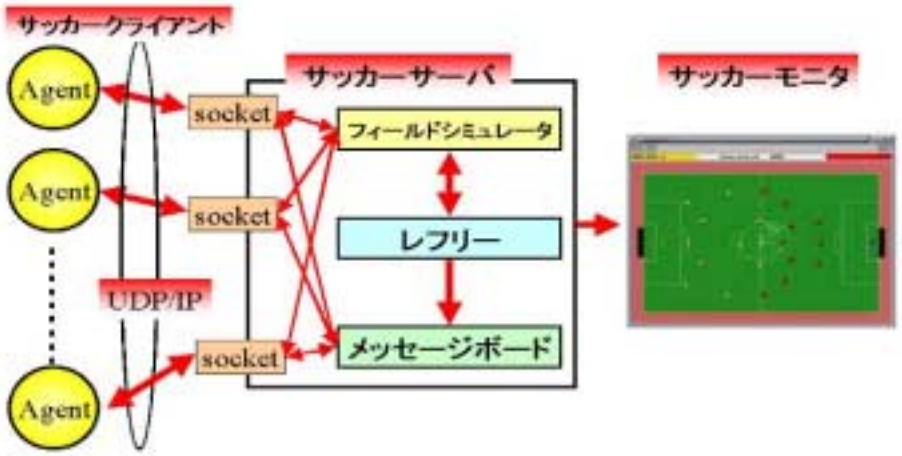



図 2.1 シミュレータの構成図

### 2.2.2 サッカーサーバの仕組み

サッカーサーバは, サッカーフィールド, ボール, 審判, プレイヤの位置などをシミュレートする. サッカーフィールドとフィールド上の物体は, 平面で扱われ, 高さの情報は持っていない. また, プレイヤとボールは円として扱われる. 動きの情報は, 1 シミュレーションサイクル毎 (100msec) に離散的に更新される. サイクルの終わりに, サッカーサーバは受信したすべての行動コマンドをフィールド上のすべての物体に適用して, 次のサイクルの位置と速度を計算する.

シミュレーションには，現実世界に近づけるためにボールやプレイヤに対し，ノイズや風の影響が加えられる．

### 2.2.3 プレイヤの行動

プレイヤは次の行動をすることが出来る．

- turn　プレイヤの向いている方向を体ごと変える．
- turn_neck　プレイヤの向いている方向を頭だけ変える．体の方向は変わらない．
- dash　プレイヤの体が向いている方向に加速をつける．
- kick　ボールが蹴れる範囲内 (kickable_area) にある場合に，方向と強さを決めてボールを蹴る．
- move　得点が決まったとき等の初期フォーメーションに戻るためと，ゴールキーパが catch した時，ゴールキック位置に移動する時のみ使えるの行動
- catch　ゴールキーパーがボールをキャッチする行動．ゴールキーパ以外やペナルティーエリア外では無効．
- say　メッセージをすべてのプレイヤに伝える．
- change_view　視野角度と視覚情報の品質を変える．これを変えると視覚情報の送られてくる頻度が変わる．

サッカークライアントは，これらの行動を駆使して設計しなければならない．原則として 1 サイクルの 1 つの行動しかできない．ただし，turn_neck,say,change_view は，ほかのコマンドと同時に 1 サイクルに行うことができる．

また，プレイヤにはスタミナがある．スタミナは，行動を起こすと減少する．減少する割合は，行動の度合い（全力で走る等）で変化する．スタミナが少ないと全力で走れないなど等の制限が加わる．行動をしなければ，スタミナは少しずつ回復する．さらに，SoccerServer7 からは，プレイヤによってスタミナ等に個人パラメータがつけられ個性が持たせられた．

### 2.2.4 センサ情報

サーバからセンサ情報としてクライアントに次のような情報が送られてくる.

- 視覚情報　クライアントの視野内に入ったオブジェクトの情報.
- 聴覚情報　審判からの判定メッセージと，クライアントプログラムが say コマンドで送ったメッセージ情報.
- 感覚情報　自プレイヤのスタミナ，視野モード，速度，頭の向き，kick, dash, turn, say, turn_neck のコマンドを送った回数の情報.

センサ情報がサーバから送られてくる頻度は，それぞれ違う.

視覚情報は，視野の大きさ・クオリティによって送られてくる時間は変化するが，初期状態では 150msec 毎にクライアントに送られる．ただし，シミュレーションサイクルとは非同期である.

聴覚情報は，メッセージが発生したときにシミュレーションサイクルとは非同期に送られる.

感覚情報は，シミュレーションサイクル毎に必ず送られてくる.

# 第 3 章

# KUT RoboCup Project について

この章では，本学の RoboCup への取り組みについて述べる．

## 3.1 KUT RoboCup Project とは

高知工科大学情報システム工学科では,「RoboCup シミュレーションを通じてのソフトウェア工学及び人工知能の教育」と題して，RoboCup を使った授業・研究を 1999 年度より行っている [3]．KUT RoboCup Project の概要図を図 3.1 に示す.

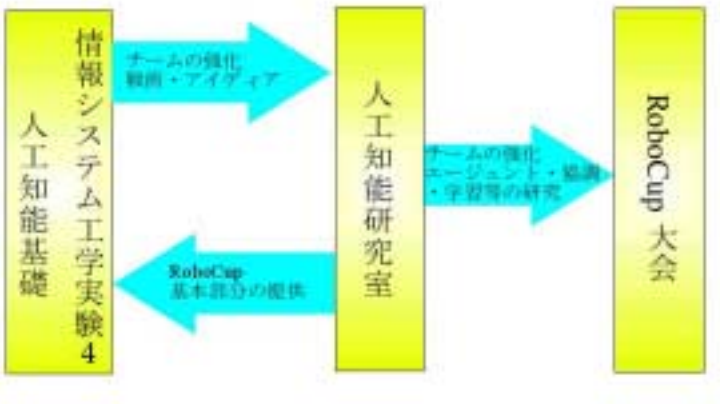



図 3.1　KUT RoboCup Project 概略図

図 3.1 で示したように，人工知能研究室と実験科目で相互協力し，年 1 回の RoboCup 大

会に代表チームを送り込むことが目的である．実験 4 では，実際にクライアントを作ってもらう．最終的には大会を行い，戦術や機能のアイディアを出し実装してもらう．また，人工知能研究室の役割としては，次の 3 つがある．1 つは，実験 4 で使用するクライアントプログラムの基本部分を開発・提供を行う．2 つ目は，実験 4 で出されたアイディアや，戦術，機能，などを統合し，代表チームの強化に利用する．3 つ目は，人工知能に関する研究の成果を代表チームに追加する．

## 3.2　RAIK-NTG4 における本研究の位置付け

本研究の位置付けを述べる前に RAIK-NTG4 について説明する．

### 3.2.1　RAIK-NTG4

RAIK-NTG4(Researching A.I. of KUT - Nohohon Technology Generation IV) は，2002 年の高知工科大学の代表チームである．

RAIK-NTG4 の構成を図 3.2 に示す．

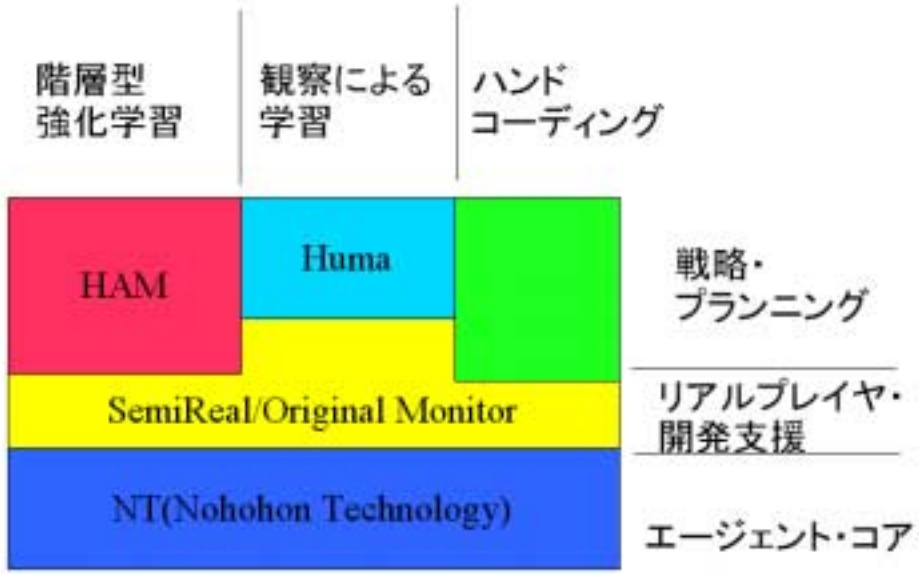



図 3.2　RAIK-NTG4 概略図

RAIK-NTG4 は大きく分けて，3 つの部分で成り立っている．

- エージェントコア部

  この部分は，NT(Nohohon Technology) と呼ばれ，クライアントを動かす上で最低限必要な部分である．

  サーバへの接続，サーバからのデータの送受信，さらに，World Model[4] などサーバからの情報をクライアントが使いやすいよう加工する機能もこの部分に含まれる．

- リアルプレイヤ [5]・開発支援部

  この部分は，戦略・プランニングを作成する上でエージェントが持っている情報・正確さを視覚的に表し，エージェント作成の開発を支援する．

  - Original Monitor

    エージェントが持っている情報を視覚的に表すだけのものである．表示はサッカーモニタのクラシックモニタに似せてあり，エージェントが実際どのような情報を持っているのかを表示する．

  - SemiReal[6]

    Original Monitor と同じく，エージェントか持っている情報を視覚的に表すが，SemiReal ではマウス等の入力ディバイスを用いてエージェントを動かす事もできる．ここでの行動等に関する情報は，ログとしてファイルに吐き出される．

- 戦略・プランニング部

  エージェントがさまざまな情報を元に行動を決定する部分．人間の脳にあたる部分である．RAIK-NTG4 では戦略・プランニングをさまざまな手法で取得する．

  - ハンドコーディング

    従来の作成法で，人間が IF-THEN ルールによって記述してゆく．ここでの戦略・プランニングは基本的に実験 4 でのアイデアを強化したものである．

  - Huma[7]

    SemiReal によって抽出されたログを解析し，人間が行動を決定する要因を見つけ

だす．その後，それに基づいてエージェントは行動する．人の意志決定を行動に反映する．

– HAM

行動を強化学習によって，取得・最適化してゆく．

### 3.2.2 RAIK-NTG4 における位置付け

本研究は，RAIK-NTG4 の戦略・プランニング部に位置する HAM の部分に位置する．

強化学習を取り入れたエージェントは，様々な相手に対し，効果的な対応を見つけだすと考えられる．更に，階層化した強化学習を使用することによって，収束性を速める．

これにより，様々な相手に素早く適応することのできるエージェントを作成する．

# 第 4 章

# 階層型強化学習

本章ではまず，従来の強化学習について説明する．強化学習の特徴，利点，欠点をあげる．その後，階層型強化学習することについての利点を上げ，階層化の方法を示す．そして，本研究の目的である，RoboCup エージェントに適用することでの利点を述べる．

## 4.1 強化学習とは

迷路のゴール地点に鼠の餌になるものを置いておき，その迷路の中に鼠を放し，放って置くと，鼠は試行錯誤を繰り返しゴール地点にたどりつく．ゴール地点にたどりついたところで餌を与え再び鼠を迷路のスタート地点に戻す.

このことを繰り返すと，鼠はゴールまでの経路を学習してゆく．

強化学習とは，上記の現象を機械に行わせる (すなわち，行動とその行動に対する報酬を与えることによって，その行動を強化する) 機械学習の 1 つである．

学習・行動決定を行う者をエージェントと呼ぶ．エージェントは「環境」から与えられる，状態によって行動を決定する．その結果,「環境」は変化する．この強化学習の枠組みを図 4.1 に示す.

1. エージェントは環境の状態 $S_t$ において行動を選択し、行動 $a_t$ を行う.
2. 行動の結果，環境に影響をあたえ，その結果，状態が $S_{t+1}$ になる．
3. 状態 $S_{t+1}$ に応じて，報酬 $r_t$ を受け取る．
4. 状態 $S_{t+1}$ で 1 に戻る．

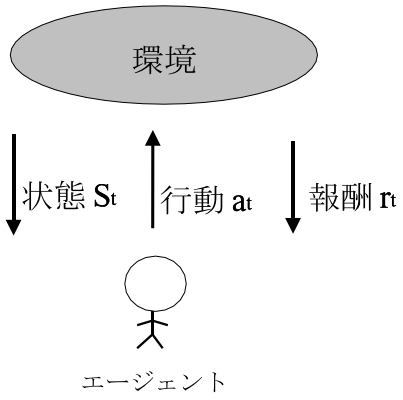



図 4.1　強化学習の枠組み

強化学習は，上記を繰り返すことにより学習をする．

### 4.1.1　マルコフ決定過程

強化学習は通常，マルコフ決定過程と呼ばれる数学的モデルで表すことのできる空間で行われる．

マルコフ決定過程は図 4.2 のような，状態遷移図によって表現することができる．

マルコフ決定過程は

- 状態
- 行為
- 状態遷移確率
- 報酬

で記述される．

マルコフ決定仮定では，状態遷移確率は状態のみに依存し，それ以前の状態には依存しないこと，また，状態遷移確率は時間的に変動しないこと等が仮定されている [8].

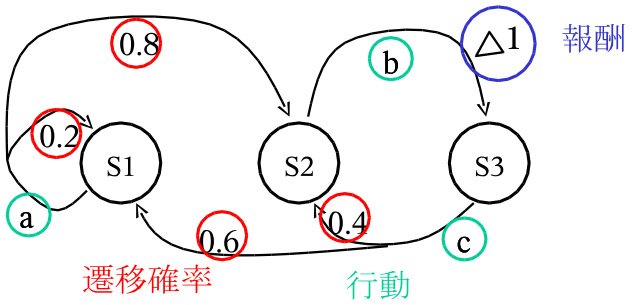



図 4.2　マルコフ決定過程

図 4.2 では，状態 *S3* で $c$ の行動を行った時，0.6 の確率で状態 *S1* になり，0.4 の確率で状態 *S2* となる．

しかし多くの場合，状態を固定できても状態遷移確率は未知であることが多い．Q 学習は，そのような場合にも学習することのできる，強化学習の代表的な学習手法である．

### 4.1.2　Q 学習

Q 学習は強化学習の代表的な学習手法の 1 つである [8][9]．

本研究でも，第 5 章の実験では学習法に Q 学習を用いている．

Q 学習では，Q 値と呼ばれる評価値行動決定に用いる．Q 値は，状態と行動をペアにした個々のルールの評価値である．Q 学習では，様々な状態でいろいろな行動をすることにより，その時得た報酬を基に，評価値である Q 値を更新してゆき，強化が行われる．Q 値は行動を行ったとき，前の状態・行動の Q 値に対して行われる．

下に Q 値の更新式を示す．

$$Q_{(s,a)} \leftarrow (1-\alpha)Q_{(s,a)} + \alpha(r(s,a) + \gamma \max_{a'} Q_{(s',a')}) \tag{4.1}$$

上の式では，状態 $s'$ の時に，前回の状態 $s$・行動 $a$ のペアである Q 値に対して，Q 値の変更を行っている．

$\alpha$ は学習率・$\gamma$ は割引率で，それぞれ $0 \leq \alpha = 1$，$0 < \gamma < 1$ の値をとる．

## 4.2　強化学習の利点

強化学習は，エージェントに目標のみを与えておけば，エージェント自身が試行錯誤を繰り返し，そこに至るまでの行動を学習してゆく．

そのため，以下のような利点を持つ．

- 目標までの行動を人間が知らなくて良い

  エージェントは目標までの行動を試行錯誤によって見つけ出すため，人間が目標までの行動を知っておく必要はない．
- 人間以上の行動を見つけ出す可能性がある

  未知空間では試行錯誤を繰り返し，目標までの行動を見つけ出す強化学習は，人間以上の最適な行動を見つけ出す可能性がある．
- 状態空間の変化に学習により追従できる

  予期しない状態空間の変化によって，目標の位置が変わってしまっても学習によって追従することができる．

## 4.3　強化学習の問題点

先に述べたよう，強化学習はエージェント自身が試行錯誤を繰り返し目標までたどり着く．そして，そのことを繰り返すことにより，目標までの行動を学習してゆく．

そのため，状態変数・行動の数が多ければ多い程，試行錯誤回数が指数関数的に増大するため，学習収束までにかかる時間も増大してしまう．

## 4.4　階層型強化学習の利点

4.3 での問題を解決するため，強化学習を階層化し問題を分割することが試みられている．これにより，学習収束までの時間が高速化できると考えられる．

また，類似空間で，上位階層または下位階層の学習済みデータを利用するなどといった，学習データの再利用も容易になる．

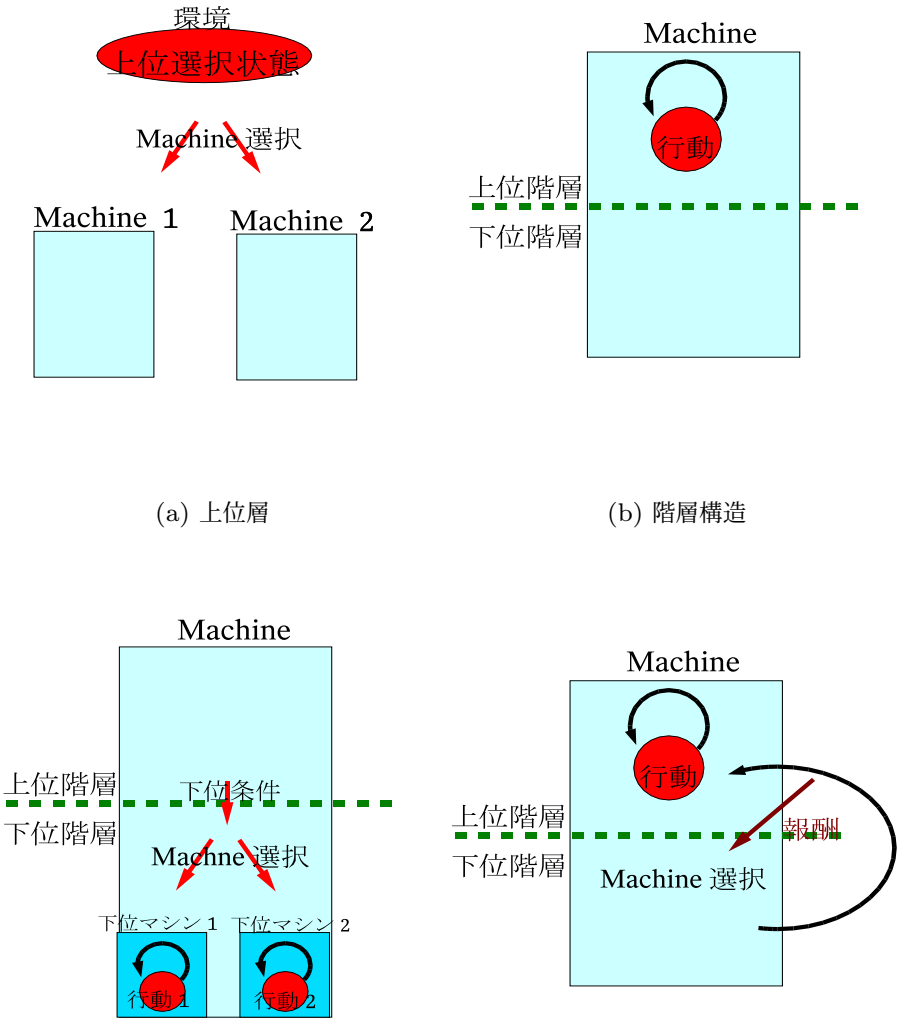
## 4.5　階層化の方法

本研究では、階層化を 1998 年に Ron Parr 氏らによって提案された手法 [10] を基に行ってゆく．

その手法を以下に示す．

- 階層型強化学習では，エージェントは行動を選択するのではなく Machine を選択する．(図:4.3(a))
- この Machine の選択は特定の状態の時のみに行われる (このエージェントの状態を以後，選択状態と呼ぶ)．
- この Machine は，階層構造になっており，行動と下位層をもつ．(図:4.3(b))
- Machine は，基本的に Machine 内で定義されている行動をするが，エージェントがある特定の状態 (以後，下位状態) になると，下位階層を呼び出す．
- 上位階層より呼び出された，下位階層は Machine 選択を行い，(図:4.3(c))，選択された Machine 内の行動を行う．
- 下位階層での行動によって，下位状態ではなくなった時，行動を上位階層の行動に返す．そして，下位階層の選択に対して報酬を与える．(図:4.3(d))
- 上位階層への報酬は，目的を達した時に与えられる．

このように，強化学習を階層化することにより，複数の問題を含む大きな問題を分割化し，問題の状態空間を小さくすることができる．




(a) 上位層　　(b) 階層構造

(c) 下位層　　(d) 下位層への報酬

図 4.3　階層化方法

階層型強化学習での Q 学習は，(4.2) の更新式を使用する．

$$Q_{(c,m)} \leftarrow (1-\alpha)Q_{(c,m)} + \alpha(r(c,m) + \gamma \max_{c'} Q_{(c',m')}) \tag{4.2}$$

階層型強化学習では，状態と行動のペアを Q 値にするのではなく，Machine とその Machine が選択された” 選択位置” をペアとして Q 値にする．

式 (4.2) の，c は選択位置，m は選択された Machine である．

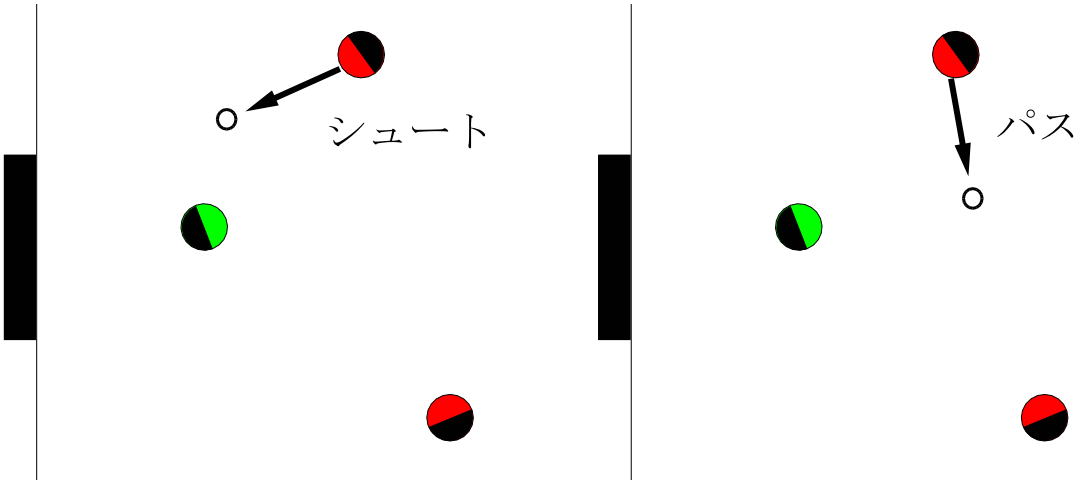
これにより，階層型強化学習は，選択位置でどの Machine を選べばよいのかを学習してゆく．

## 4.6　RoboCup への応用による利点

次に，階層型強化学習を RoboCup エージェントへ応用することによる利点を述べる．

従来の IF-THEN ルールでのハンドコーディングは，人間が経験・勘などに頼り，行動する部分の実装は非常に困難である．

更に，RoboCup では様々なチームと対戦するため，図 4.4 のように同じ状態でも，対戦相手，または，エージェントによって行う行動が異なるということが多々起こる．そのような時，IF-THEN ルールでのハンドコーディングにより作成されたエージェントでは，対応することは難しい．もし，どちらかに強いエージェントを作成したとしても，もう片方には逆に弱くなってしまうということが起こり得てしまう．




(a) A チーム　　(b) B チーム

図 4.4　チームによって異なる行動

そこで，強化学習を RoboCup エージェントに適用することにより，人間の経験・勘などに頼っている部分の行動の解を見つける．さらに，様々な相手への対応を学習により追従す

ることが可能となる.

しかし，RoboCup では状態空間が大きく，従来の強化学習では学習・学習による追従するまでに莫大な時間を要してしまう.

そこで，階層化した強化学習を使用することにより，RoboCup エージェントでの高速な学習を期待する.

# 第 5 章

# 実験

本章では，階層型強化学習と従来型の強化学習の収束性を確かめるため，ベンチマークテストを行い，その結果を示す．その後，実際に RoboCup エージェントへの実装を試み，その有効性を検証する．

## 5.1 迷路問題における従来型と階層型の比較

迷路問題は強化学習の収束を確かめる上でしばしば使われる．迷路問題では，エージェントにゴールまでの経路を学習させる．

### 5.1.1 実験内容

今回，ベンチマークとしての問題には，図 5.1 のマップを使用した．

このマップについては，

- マップの S の部分 (図 5.1 では左上) が，この迷路のスタート地点に当たる．
- マップの黒い場所は壁で，エージェントはその位置に侵入することはできない．
- Goal 地点 (図 5.1 では右下) にエージェントがたどり着いたとき，エージェントに対して報酬が与えられ，スタート位置に戻される．

というようにする．

次に，この問題で使用するエージェントの仕様について述べる．

- エージェントは，1 サイクルに 1 回，上下左右いずれかの方向に移動することができる．

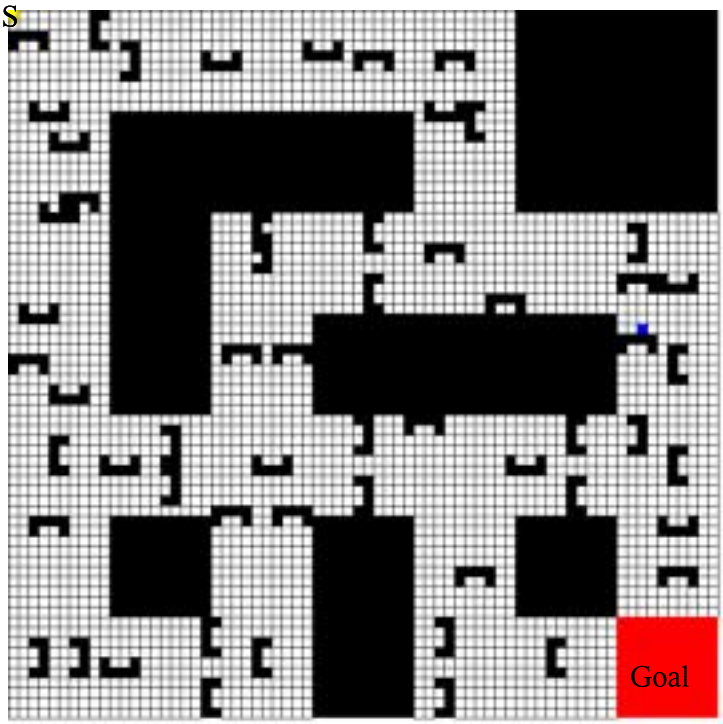



図 5.1　迷路問題

ただし，進行方向が壁だった場合，そちらの方向へは進むことができない．

- エージェントは自分の位置情報を正確に把握することができる．
- エージェントは最初，ゴールの位置，壁の位置を知らない．
- エージェントは，ある程度のソナーを持っており自分の付近ならば壁を感知することができる．

以上のことを，階層型・従来型エージェントの共通の仕様とし，迷路問題の経路を学習させた．

### 5.1.2　実験方法

階層型強化学習のエージェントと従来型強化学習のエージェントを図 5.1 で学習させ，時間の経過と収束性を見る．

実験の際，学習率・割引率などといった学習に必要なパラメータは同じにし，双方とも学

習法に Q 学習法，行動の選択にはソフトマックス行動選択 [11] を使用する．

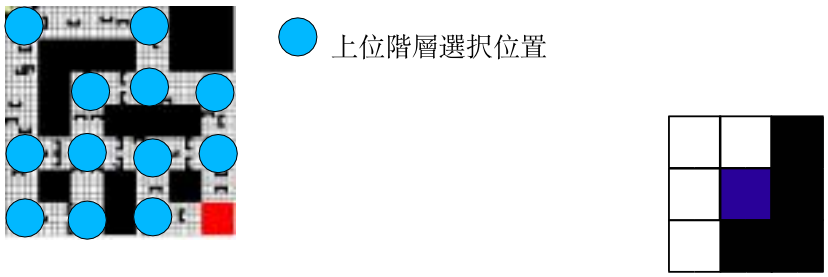
### 5.1.3　迷路問題での強化学習の階層化

次にこの迷路問題での階層化方法を述べる．

この迷路問題では，2 つの副問題が含まれる．1 つはゴールまでの大まかな経路，そして，各所に点在する壁の回避法である．

そこで，階層型強化学習では階層を 2 つに分け，上位階層でゴールまでの大まかな経路，下位階層で壁の回避法を学習させる．

迷路問題における階層型強化学習の，上位階層での行動選択位置，下位条件を図 5.2 に示す．




(a) 上位階層　　(b) 下位階層

図 5.2　迷路問題における上位選択位置・下位階層条件

- 上位階層

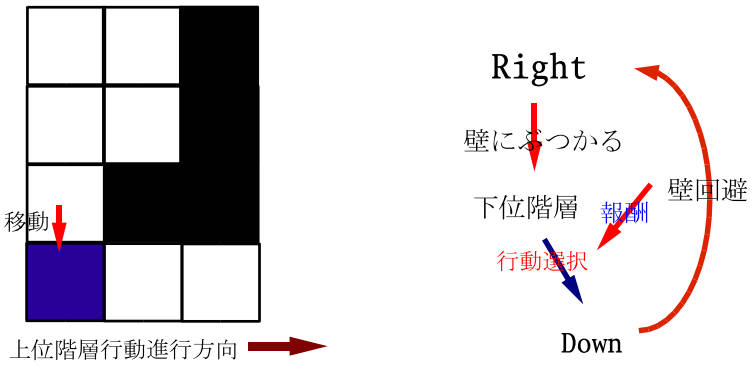
  上位階層では，エージェントの方向性を決定させるため，図 5.2(a) の様に，マップの角・三差路・十字路でのみ行動を選択する．その他の状態のとき，エージェントは下位階層に入らない限り，その時に選択された行動を繰り返す．

  上位階層への報酬は，目的であるゴール地点にたどりついたときに与えられる．

- 下位階層

下位階層へはエージェントが壁にぶつかった時に推移する．その際，行動も下位階層に移り，下位階層で決定された行動を起こす．そして，壁を回避した際，報酬を与え，上位階層の行動に再び移る．

図 5.3 に，下位から上位階層へ移るときの報酬を示す．




(a) 壁回避　　(b) 下位への報酬

図 5.3　壁回避による下位階層への報酬例

### 5.1.4　結果

この迷路問題での結果を図 5.4 に示す．

図 5.4 のグラフの縦軸は，ゴールまでにかかったサイクル数である．ゴールまでにかかったサイクル数が多ければ多い程，多く試行錯誤を繰り返しゴールしている事になる．

グラフの横軸はゴールの回数である．強化学習は，1 度ゴールに入っただけではほとんど学習しない．何度もゴールすることにより，そこに至るまでの行動を学習する．

強化学習では初め，ランダムな行動を繰り返すためゴールするのに莫大な時間を要してしまう．

実際，従来型強化学習の場合，毎サイクル，エージェントの行動を選択・決定しているため，ゴールにたどりつくまでに莫大な時間を要してしまっているのが分かる．さらに，状態

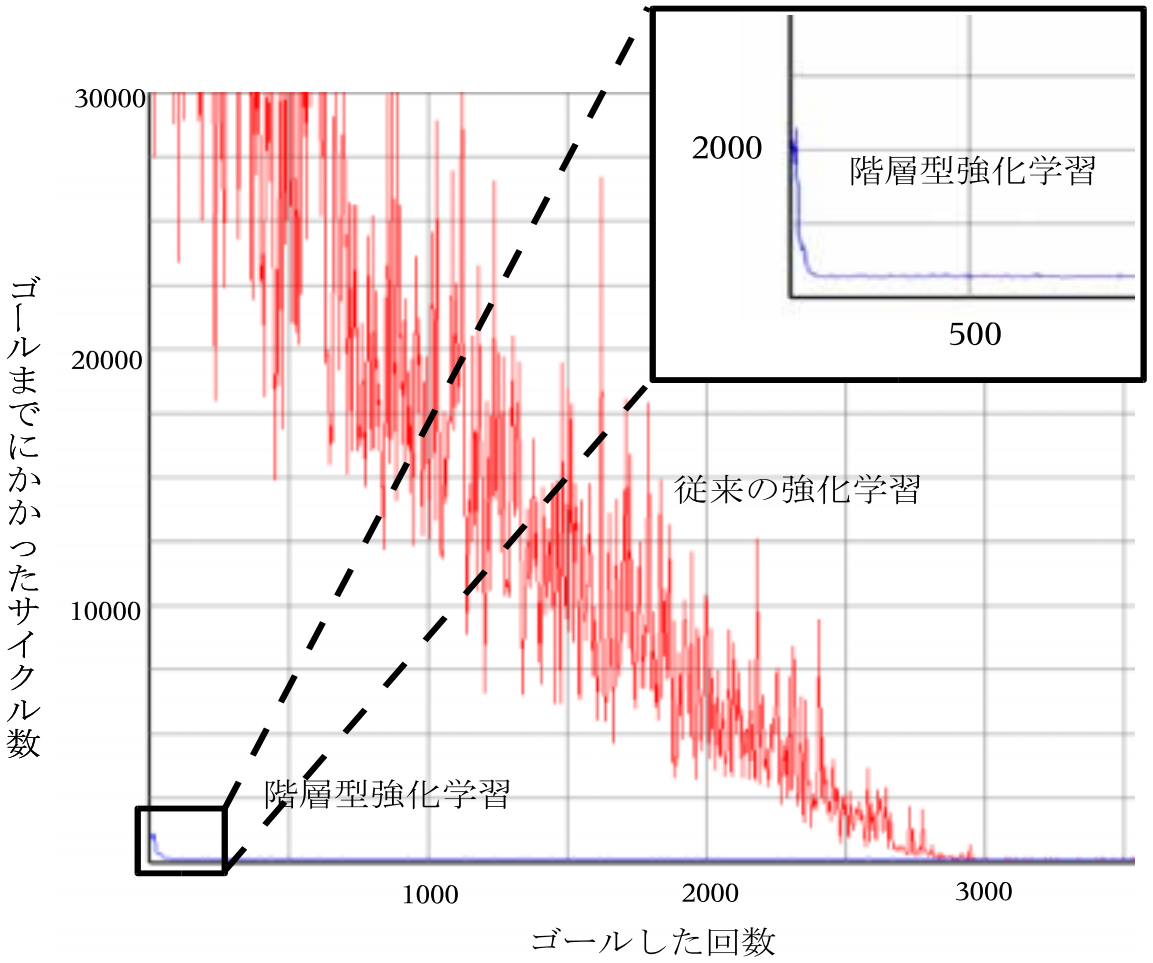



図 5.4　迷路問題における階層型と従来型の比較

数が多いため，学習が収束し，ゴールまでの行動が一定になるまでに約 3000 回のゴールを必要としてしまう．

それに対し階層型強化学習ではどうだろう．ゴールまでの方向性を学習する上位階層では，行動の選択位置を角・三差路・十字路のみにしているため，状態空間が小さくなる．

そのため，試行錯誤の回数も少なく，初めにゴールするまでのサイクル数も毎サイクル行動を選択し，試行錯誤を繰り返す従来型の強化学習ほど莫大にはならなし，実際なっていない．

さらに，状態空間が小さいため学習の収束も速くなっているのが分かる．

### 5.1.5　考察

階層型強化学習が従来の強化学習に比べ，飛躍的に学習速度が高速化したことを確認することができた．

これは，強化学習を階層化することにより，ゴールまでの方向性と壁の回避法といったように，問題を分割化することができたからだと思われる．これにより，問題の状態空間が小さくなり学習速度が上がったと考えられる．

更に，下位層では，下位層から上位層への行動の推移時に報酬が与えられるため，小さな状態空間で頻繁に報酬をもらうことができる．そのことも，高速化の要因と考えられる．

## 5.2 RoboCup サッカーエージェントへの適用

次に，本研究の目的である RoboCup エージェントへ適用する．本研究では，強化学習の実装が難しいとされ，これまで実装例のない，ゴールキーパに階層型強化学習を実装する．

RoboCup では，マルチエージェントである．その影響の大きいゴールキーパ・エージェントの状態空間は，他のエージェントの状態空間よりも大きい．そのため，試行錯誤を繰り返し，行動を取得する強化学習では莫大な時間がかかってしまう，そのため，ゴールキーパ・エージェントは通常の強化学習の適用に向いてないとされる．

### 5.2.1 RoboCup サッカーエージェント・ゴールキーパへの階層型強化学習の実装

ゴールキーパでの問題の分割は図 5.5 のようにした．

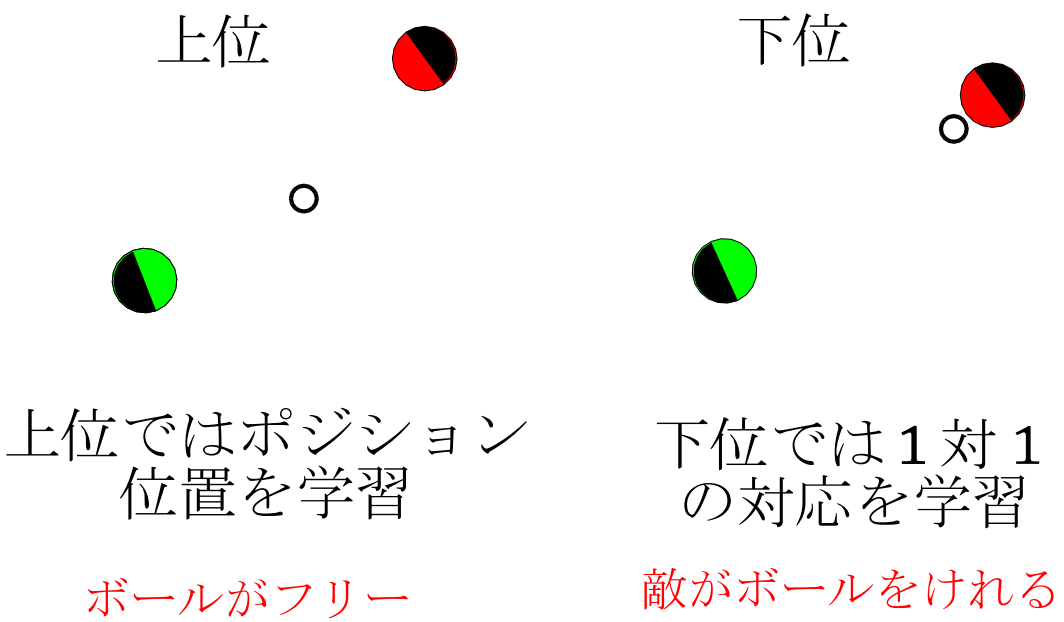



図 5.5 ゴールキーパでの階層化

- 上位階層

  上位階層では，敵エージェントがボールを蹴ることのできないときの守備位置を学習させる．

  上位階層での報酬は，自分がボールをキャッチすることができたときに与えられる．

- 下位階層

  下位階層では，敵エージェントがボールが蹴れる状態の時に 1 対 1 の対応を学習させる．

  下位階層への報酬は，上位階層へ行動を返したとき，すなわち，敵エージェントがボールを蹴り，ボールが蹴れない状態になったときに与えられる．

### 5.2.2　結果・考察

現在 (2001 年 1 月) の時点では，学習が収束することを確認することができなかった．これは，階層化による問題の分割がうまく行かず，階層化することによる効果がなかったために，学習が進まなかったものと考えられる．

RoboCup エージェントへの実装では，下位層から上位層への遷移は図 5.6 のようなものだと考えていた．

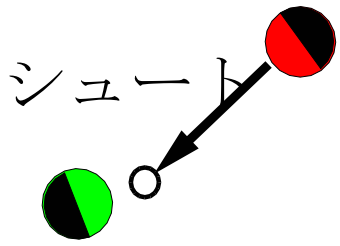
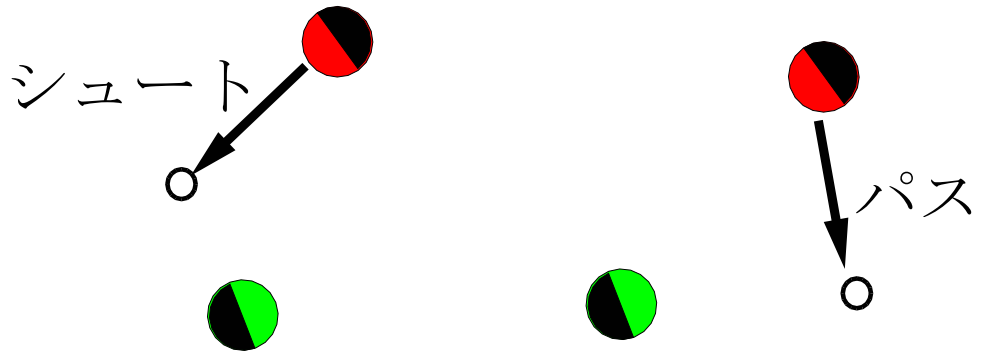



図 5.6　下位層への報酬

しかし，敵エージェントがボールを蹴れなくなった状態で下位状態をぬけるため，図 5.7 の時にも下位階層に報酬が与えられてしまう．




(a) シュート　　(b) パス

図 5.7　下位層への報酬 2

図 5.7(a) では，下位階層での行動が誤っているため，敵エージェントにシュートを打たれ，且つ，抜かれてしまう．しかし，ボールは敵エージェントから離れ，敵エージェントがボールを蹴ることができなくなるため，下位条件を満たさず，上位階層へ行動を返し、下位階層に対して報酬が与えられてしまう．

上記のように、不適当なときにも下位層に対して報酬を与えることになってしまったため、学習が収束しなかったものと考える．

# 第 6 章

# まとめ

強化学習を階層化することにより，学習速度が飛躍的に速くなることをベンチマーク問題でによって示した．

しかし今回，RoboCup サッカーエージェントへ適用しその有効性を示すことはできなかった．

今後は，RoboCup エージェント・ゴールキーパでの，問題分割の再検討が必要である．更に，ゴールキーパ以外のエージェントに階層型強化学習を適用し，その有効性を示すことが必要である．

# 謝辞

本研究を進めるにあたり，とても丁寧に御指導くださった Ruck Thawonmas 助教授に心より御礼申し上げます.

RoboCup に関しましては，大学院生の平山純一郎さん，4 年の田村和也君，そして，情報システム工学実験 4 に参加してくださった多くの学生の皆様に感謝いたします.

また，論文を書くにあたり，人工知能研究室の皆様に大変お世話になりました.

ありがとうございました.

# 参考文献


[1] http://www.robocup.or.jp/.

[2] gttp://sourceforge.net/progects/sserver/.

[3] Ruck THAWONMAS，大森洋一，平山純一郎，
“RoboCup ソフトウェアエージェントロボットづくりによる問題設計解決型学習，
第 9 回情報教育方法研究発表予稿集，pp．50-51，2001 年 7 月．

[4] 平山純一郎，
“RoboCup における World Model の構築に関する研究”，
平成 12 年度学士学位論文，2001．

[5] 秋田淳一，西野順次，久保長徳，下羅弘樹，藤墳倒，
“RoboCup シミュレーションリーグ人間参戦システム OZ-RP の提案”，
人工知能学会第 12 回 Sig-Challenge 研究会資料，pp.23-28,April.2001．

[6] 平山純一郎，田村和也，日野慎一，川口宏，Ruck THAWONMAS，竹田 史章，
” 人間挙動観察によるエージェント学習システム”，
情報処理学会四国支部研究シンポジューム，平成 14 年 3 月 15 日 (口頭発表予定)．

[7] 平山純一郎，Ruck Thawonmas，
“RoboCup ソフトウェアエージェントへの人間の意思決定挙動の適用”，
電子情報通信学会「人工知能と知識処理」11 月研究会，信学技報 AI2001-54，pp.
57-61，2001 年 11 月．

[8] 馬場口登，山田誠司 共著，
“人工知能基礎”
昭昇堂，pp．145-149，1999．

[9] http://www.ncfreak.com/asato/doc/cs/cs-memo.html.

[10] Ron Parr，Stuart Russell，

"Reinforcement Learning with Hierarchies of Machines",
Advances in Neural Information Processing Systems 10, pp. 1043-1049, 1998.

[11] Richard S. Sutton, Andrew G. Barto, (三上貞房, 皆川雅章 共訳),
"強化学習",
森北出版株式会社, pp. 32-33,2000.