学術・技術論文

複数の学習するロボットの存在する環境における協調行動獲得のための状態空間の構成

内部英治[*1]　浅田稔[*2]　細田耕[*2]

State Space Construction for Cooperative Behavior Acquisition in the Environments Including Multiple Learning Robots

Eiji Uchibe[*1], Minoru Asada[*2] *and* Koh Hosoda[*2]

This paper proposes a method that acquires cooperative behaviors based on the estimation of the state vectors. In order to acquire the cooperative behaviors in multi robots environments, each learning robot estimates the local predictive model between the learner and the other objects separately. Based on the local predictive models, robots learn the desired behaviors using reinforcement learning. The proposed method is applied to a soccer playing situation, where a rolling ball and other moving robots are well modeled and the learner's behaviors are successfully acquired by the method. Computer simulations and real experiments are shown and a discussion is given.

Key Words: reinforcement learning, multiple rewards, hierarchical architecture, policy invariance

1. はじめに

実世界で与えられたタスクを遂行することを自律的に獲得できるロボットを実現することは，ロボティクスと AI の中心課題の一つである．近年，複数の自律的エージェントが相互作用を通して問題を解決するための研究が盛んであり，視覚情報と行動の統合に基づく分散協調視覚のプロジェクトもある [10]．分散人工知能 [4] の分野では，各学習者の行動戦略または契約ネット [14] などの通信のプロトコルを事前知識として各エージェントに与え，エージェント間の通信量を抑えるための研究がなされてきた (例えば [12])．しかし，エージェント間の関係は協調だけでなく，競合や無視といった様々な関係があり，常に適切な通信が仮定できるわけではなく，他のエージェントとの相互作用を通して関係を構築すべきである [15].

しかし，通常の学習理論は単一エージェントの場合を想定しており，環境中で能動的に状態を遷移できるのは学習者だけである．この場合，学習者のセンサ情報と行動を 1 対 1 に対応づけることで目的の行動を獲得することが可能である．しかし，マルチエージェント環境では，学習者以外のエージェントが状態遷移を引き起こす可能性があり，他者の行動政策は学習者にとって一般的には未知であることから，従来の学習方法を適用するための前提条件が満足されないことになる．つまり，センサから得られる瞬間の情報だけでは，次の状況を予測することは困難であり，観測をそのまま状態として用いた場合，マルコフ性の条件が満足されないため，目的の行動を獲得できないことになる．これは知覚的見せかけ問題 [22] と呼ばれ，学習理論を実際の問題に適用する際に解決しなければならない重要な問題である．

ロボットに自律的に目的行動を獲得させる手法として強化学習法 [18] があり，複数の学習者に協調，競合行動を獲得させるために様々な手法が提案されている．Littman [8] は，格子環境下においてマルコフゲームの枠組みを応用した強化学習法を提案している．しかし，学習者の評価関数は 2 人ゼロ和の関係にあり，学習者は常に他者の最悪の行動を想定することで学習するため，協調の問題には適用できない．Sandholm and Crites [13] は 繰り返しの囚人のジレンマ問題に強化学習を適用し，学習が成功するためには，十分な過去の観測量と行動が必要であることを示した．しかし，その履歴の長さを決定するのは，一般に困難な問題である．荒井ら [2] は格子状環境での追跡問題において，Q 学習と Profit Sharing の比較を行ない，Profit Sharing の方がマルチエージェント環境での学習に適していると結論している．

また，実ロボットに適用した例として Stone and Veloso [16] は階層構造の学習法によりサッカーゲームを実現した．しかし，彼らの手法は瞬間のセンサ情報 (観測量) を状態として利用して

原稿受付　1998 年 10 月 28 日
[*1]ATR 人間情報科学研究所 第三研究室
[*2]大阪大学大学院工学研究科
[*1]ATR Human Information Science Laboratories, Department 3
[*2]Graduate School of Engineering, Osaka University

おり，センサの変化量と行動が 1 対 1 に対応しない複数ロボットの学習問題に適用することは困難である．Sugita and Tani [17] は単一のリカレントニューラルネット (RNN) に環境のモデルの学習と行動の学習を実装し，幾つかの分岐点を含む迷路環境において 2 台のロボットの簡単な協調ゲームを実現している．しかし，彼らの手法では,「壁伝い行動」,「目標物体への到達行動」などの基本行動が埋め込まれており，RNN は分岐点においてどの基本行動を選択するかを学習すれば良いように問題が抽象化されている．また，物理的に意味の異なる学習を単一の RNN に適用するといった問題点がある．

本論文では，学習者の観測と行動を通して，学習者と他者の行動の関係を局所予測モデルとして推定し，その結果をもとに強化学習をおこなう手法を提案する．局所予測モデルは線形の状態空間表現を持ち，学習者と他者の関係の複雑さは推定される状態ベクトルの次元で表現され，情報量基準 [1] をもとに決定される．また，学習を安定にするための学習のスケジューリング法を提案する．

提案する手法を簡単な 1 対 1 のサッカーゲームに適用する．環境には 2 台のロボットが存在し，それぞれに対して異なるタスクを与える．環境は静的エージェント (ゴール)，受動エージェント (ボール)，能動エージェント (移動ロボット) から構成され，学習者はそれぞれに対して局所予測モデルを構築する．各学習者は局所予測モデルを構築した後に，強化学習によって目的行動の学習を開始する．実ロボットによる実験結果を示し，本手法の有効性を検証する．

2. 観測と行動に基づく内部モデルの構築

2.1 各ロボットに与えられる構造

学習を安定に収束させるためには，適切な状態ベクトルが必要であることは 1 節で述べた．しかし，学習者はセンサ能力の限界や，他の学習者の戦略の不確実性のため，完全な状態ベクトルを事前に知ることができない．学習者にとって可能なことは，観測結果と学習者の行動のシーケンスを解析して，パラメータ数と予測能力のトレードオフを考慮した表現を獲得することである．

Fig.1 は各ロボットに与えられる行動獲得のための学習システムである．環境内に学習者以外に N 個の認識できる対象物が存在するとき，学習者は N 個の局所予測モデルによって学習者自身と対象物間の相互作用を推定する．各局所予測モデル LPM_i $(i = 1, \cdots N)$ は観測と行動のシーケンス $\{\boldsymbol{y}_i, \boldsymbol{u}\}$ から内部表現として状態ベクトル $\boldsymbol{x}_i$ を出力し，相互作用の複雑さは状態ベクトルの次数で表現される．また，学習者は一つの強化学習部を持ち，推定された N 個の状態ベクトル $\boldsymbol{x}_i$ と環境から与えられる報酬 r をもとに，目的行動を学習する．強化学習部は次の行動を選択し，それが局所予測モデルに貯えられ，モデルが更新される．

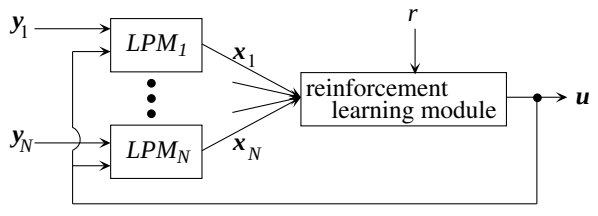


Fig. 1 Whole system of our proposed method

厳密には全ての物体が互いに相互作用しているため，全体の相互作用を考慮する必要があるが，

(1) 学習者が観測できるのは部分情報だけであり，全ての対象物を同時に観測できるとは限らない，

(2) 対象が異なれば学習者との関係の複雑さも異なり，対象ごとに複雑さを推定すべきである，

といった理由から局所予測モデルが学習者とそれ以外の対象物との局所的な相互作用を別々に推定する．

2.2 局所予測モデルにおける状態表現

局所予測モデルは，多入力 (行動) 多出力 (観測) の関係を記述する必要がある．局所予測モデルの状態表現の方法として，システム同定の一つである正準変量解析 (Canonical Variate Analysis，以下 CVA) [6] を用いる．

CVA は離散時間で線形時不変の状態空間モデル

$$\begin{aligned} \boldsymbol{x}(t+1) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t), \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t), \end{aligned} \tag{1}$$

を用いる．ここで $\boldsymbol{u}(t) \in \Re^m$ と $\boldsymbol{y}(t) \in \Re^q$ はそれぞれロボットの行動ベクトルと観測ベクトルであり，$\boldsymbol{x}(t) \in \Re^n$ は状態ベクトルである．また，$\boldsymbol{A} \in \Re^{n\times n}$, $\boldsymbol{B} \in \Re^{n\times m}$, $\boldsymbol{C} \in \Re^{q\times n}$, $\boldsymbol{D} \in \Re^{q\times m}$ はパラメータ行列である．学習者は観測と行動のシーケンス $\{\boldsymbol{y}, \boldsymbol{u}\}$ から状態ベクトルを次数を含めて推定しなければならない．

今，新たな状態ベクトル

$$\boldsymbol{p}(t) = \begin{bmatrix} \boldsymbol{u}(t-1) \\ \vdots \\ \boldsymbol{u}(t-l) \\ \boldsymbol{y}(t-1) \\ \vdots \\ \boldsymbol{y}(t-l) \end{bmatrix}, \quad \boldsymbol{f}(t) = \begin{bmatrix} \boldsymbol{y}(t) \\ \boldsymbol{y}(t+1) \\ \vdots \\ \boldsymbol{y}(t+l-1) \end{bmatrix}, \tag{2}$$

を考える．ここで，l は考慮する履歴長さである．部分空間同定法は，この二つのベクトル $\boldsymbol{p}$, $\boldsymbol{f}$ がそれぞれ生成する部分空間の関係から直接状態ベクトルを推定することができる [21].

具体的には，状態ベクトル $\boldsymbol{x}$ は過去の観測と行動のシーケンスの線形和

$$\boldsymbol{x}(t) = [\boldsymbol{I}_n \ \boldsymbol{0}]\boldsymbol{M}\boldsymbol{p}(t), \tag{3}$$

によって表現される．ここで $\boldsymbol{M} \in \Re^{l(m+q)\times l(m+q)}$ は CVA によって計算される行列であり，n は状態ベクトルの次数，$\boldsymbol{I}_n$ は n 次の単位行列である．$\boldsymbol{M}$ の計算方法は付録を参照されたい．

2.3 局所予測モデルのパラメータの決定

推定された状態ベクトル $\boldsymbol{x}$ の次元 n と考慮する履歴長さ l を決定することは重要な問題である．履歴長さ l が長くなれば推定精度は改善されるが，推定に必要なデータ数は増加し，記憶容量の問題からも l はできるだけ小さいほうが望ましい．また，n は局所的な相互作用の複雑さに影響を及ぼし，n に応じ

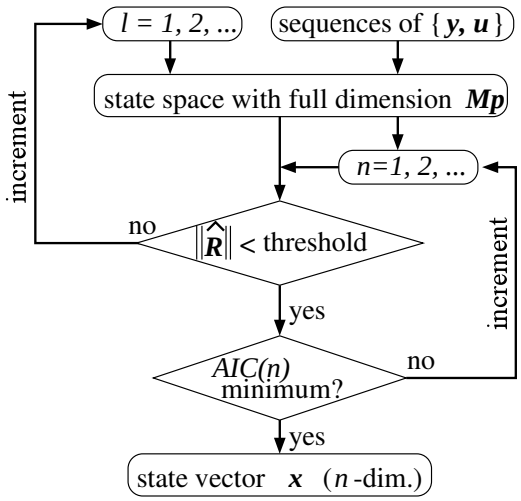


Fig. 2 Flowchart of local predictive model

て学習に要する時間が指数関数的に増大する．

学習者にとってはマルコフ性を満足する状態ベクトルが必要であり，メモリを犠牲にしてでも予測精度を保証する必要がある．よって，誤差の共分散行列 $\boldsymbol{R}$ のノルムが閾値 $threshold$ 以上であった場合，l を増加することで対処する．どの程度の誤差を許容するかは与えられたタスクに依存するが，ここでは経験的に設定する．また，状態ベクトルの次元 n を決定するために，本論文では赤池の情報量 (Akaike's Information Criterion, AIC) [1] を用いて，推定量とパラメータ数のトレードオフをとる．観測ベクトル $\boldsymbol{y}$ の予測誤差を $\boldsymbol{\varepsilon}$ とすると，誤差の共分散行列 $\boldsymbol{R}$ の推定値は

$$\hat{\boldsymbol{R}} = \frac{1}{N_{all} - 2l + 1} \sum_{t=l+1}^{N_{all}-l+1} \boldsymbol{\varepsilon}(t)\boldsymbol{\varepsilon}^T(t). \qquad (4)$$

と計算される．ここで，N_{all} は推定に用いたデータの個数である．したがって，赤池の情報量 $AIC(n)$ は

$$AIC(n) = (N_{all} - 2l + 1)\log|\hat{\boldsymbol{R}}| + 2\lambda(n), \qquad (5)$$

となる．ここで，$\lambda(n)$ は n 次の状態ベクトルを用いた場合のパラメータ数である．この $AIC(n)$ を最小にする n を局所予測モデルの状態ベクトルの推定次数とする．

Fig.2 に局所予測モデルの処理の流れを示す．観測と行動のシーケンス $\{\boldsymbol{y},\ \boldsymbol{u}\}$ に対して，

(1) 初期化：$l = 1, n = 1$

(2) ある l に対して以下の計算を行なう，

(a) 部分空間を構成し，その関係を式 (A.2) で行列 $\boldsymbol{M}$ を計算する，

(b) $n = 1, \cdots ql$ に対して，予測誤差行列 $\boldsymbol{R}$ を式 (4) で計算する．

(c) $||\boldsymbol{R}|| < threshold$ を満足する l, n がなければ，$l \leftarrow l+1$, として，(2) に戻る．そうでなければ，(d) へ進む，

(d) $n = 1, \cdots ql$ の間で $AIC(n)$ を最小にする n を求めて計算終了，

といった流れによって状態ベクトルは決定される．各局所予測モデルは最新の N_{all} 個の観測と行動の組が蓄積されると，局所予測モデルは最新のデータによって更新される．

3. 局所予測モデルと強化学習の統合

3.1 状態空間の構成

2 節の方法で次の観測ができる状態ベクトルを推定した後，学習者は推定された状態ベクトルをもとに，与えられたタスクを達成する行動を強化学習 [18] によって獲得する．Q 学習などの強化学習では，状態 $\boldsymbol{x} \in \boldsymbol{X}$ と行動 $\boldsymbol{u} \in \boldsymbol{U}$ の組に対して行動価値関数 $Q(\boldsymbol{x}, \boldsymbol{u})$ が定義され，環境からの報酬 r をもとに適切な $Q(\boldsymbol{x}, \boldsymbol{u})$ をオンラインで推定する．本研究では，強化学習部は局所予測モデル LPM_i $(i = 1, \cdots n)$ が推定した状態ベクトル $\boldsymbol{x}_i$ を統合するため，行動価値関数は $Q(\boldsymbol{x}_1, \cdots \boldsymbol{x}_n, \boldsymbol{u})$ と表記する．

一般に強化学習では，状態の扱い方として

(1) 連続である状態を適当に離散化し，行動価値関数をルックアップテーブルで表現する．

(2) 状態量はそのまま使用するが，行動価値関数をニューラルネットなどで近似する [7].

という二つの考え方がある．ここでは解析の容易さから前者の連続な状態量を離散化する方法を採用する．以下，離散化の方法について述べる．式 (3) で推定される状態ベクトルの共分散行列は，単位行列に規格化されているため，状態ベクトル $\boldsymbol{x}_i \in \Re^n$ の各要素 x_j $(j = 1, \cdots, n)$ をそれぞれ

$$x_j < -1, \quad -1 \le x_j < 1, \quad 1 \le x_j. \qquad (6)$$

の 3 つの要素に分割する．よって，LPM_i によって推定された状態ベクトル $\boldsymbol{x}_i \in \Re^n$ は観測されない場合を追加した $3n+1$ 個の離散化された状態 ${}^d x_i$ に変換される．以下，

$${}^d\boldsymbol{x} = [{}^d x_1, \cdots, {}^d x_N]^T$$

を離散化された混合状態ベクトルとする．${}^d\boldsymbol{x}$ を用いて行動獲得するために，本論文では，学習時間とパフォーマンスのトレードオフを考慮した複数のビヘービアを統合できるモジュール型強化学習法 [20] を用いる．ここでビヘービア i は

$\boldsymbol{X}^i$: 状態空間
$\boldsymbol{A}$: 行動空間
Q^i : 行動価値関数

の組で定義され，例えば「障害物回避」や「目標地点への到達行動」などの一連の行動のシーケンスを指す．適用した強化学習法は n 種類のビヘービア $(\boldsymbol{X}^i, \boldsymbol{A}, Q^i, i = 1, \cdots n)$ が与えられた場合，それまでの学習結果 Q^i を用いて，全状態空間を n 個の**非学習領域**と 1 個の**再学習領域**に分類する．そして再学習領域だけを効率よく学習することで，学習時間を短縮することが可能である．

3.2 モデル更新による行動政策の変更

一定期間ロボットが行動すると，観測と行動のシーケンスから状態ベクトルを決定する行列 $\boldsymbol{M}$ は正準変量解析によって再計算され，状態ベクトルの次元も更新される．そのため，行動価値関数もそれに応じて変更しなければならないが，一般に状態

表現の異なる行動価値関数の関係を求めるのは困難である．しかし，これまでの学習結果を全て破棄して再学習するのは効率が悪い．そこで，これまでに獲得した行動政策を初期政策として利用することを考える．

いま，${}^d\boldsymbol{x}_k$, $\boldsymbol{f}_k$, Q_k を第 k 段階での局所予測モデルに基づく状態ベクトル，最適行動政策ベクトル，行動価値関数とする．このとき第 k_t 段階での最適政策は，一段階前の学習結果と現在学習中の行動価値関数のトレードオフを考慮した

$$\boldsymbol{f}'_k({}^d\boldsymbol{x}_k) = \boldsymbol{\Phi}\{(1-\beta)\boldsymbol{f}_{k-1}({}^d\boldsymbol{x}_{k-1}) + \beta\boldsymbol{f}_k({}^d\boldsymbol{x}_k)\}, \quad (7)$$

として用いる．ここで β $(0 \leq \beta \leq 1)$ は行動政策のバランスを取るパラメータ，$\boldsymbol{\Phi}$ は行動を離散化するためのパラメータであり，$\boldsymbol{\Phi}(a)$ はベクトル $\boldsymbol{a}$ の各要素を $\{-1, 0, 1\}$ に写像する．また，

$$\boldsymbol{f}_k({}^d\boldsymbol{x}_k) = \arg\max_{\boldsymbol{u}' \in \boldsymbol{U}} Q_k({}^d\boldsymbol{x}_k, \boldsymbol{u}'), \quad (8)$$

であり，状態 ${}^d\boldsymbol{x}_k$ で最大の行動価値を取る行動 $\boldsymbol{u}$ を返す関数であり，ベクトル表記であること以外は通常の Q 学習における最適政策の定義である．β は 0 から次第に 1 へと変化させれば良く，

$$\beta = \frac{\mathrm{trial} - 1}{T_{\max} - 1}, \quad \mathrm{trial} = 1, \cdots, T_{\max},$$

と線形に変更する．ここで $T_{\max}$ は全試行回数である．実際には，学習中は探索行動が必要であるため，確率 α_{opt} で式 (7) によって行動を選択し，それ以外はランダムに行動を選択する行動選択を用いる．

4. 複数ロボットのための学習のスケジュール

一般に，実ロボットで学習により実現する場合，

(1) 実環境だけで学習 [9]:
単純な環境で単純なタスクである場合を除いて，シミュレーションと同じ学習法は現実的ではない．教示などの手法によって探索領域を削減する必要がある．

(2) 計算機上の学習結果を実ロボットに適用 [3]:
計算機上でのシミュレーションと実環境にはギャップがあり，いくらかの修正を必要とする．

(3) 計算機上で獲得された結果を実環境で修正：
シミュレーション結果をもとにして，実環境での学習をスケジューリングする．これは，教示されるデータを人間が生成するのではなく，シミュレーション結果が教示データに対応する．

と分類することができる．ここでは，(3) の方法を採用する．

しかし，マルチエージェント環境では他者の行動戦略が未知であるために生じる状態遷移の不確実性のため，特に初期段階における学習が不安定になる．そのため，初期段階における学習を安定化させるための方法が必要になる．**Fig.3** に提案する効率的な学習のスケジュールを示す．設計者は複数存在する学習者の中から，適当に学習者を一つだけ指定する．指定されなかった学習者は行動価値を更新せず，それまでに獲得された行動政策にしたがって行動する (Fig.3 では学習者 i が指定されている)．よって環境内で自律的に行動を選択できるのは指定された学習者だけである．学習が終了した後で，設計者は次の学習者を指定する．これを繰り返すことにより全てのロボットに局所予測モデルを構築させ，目的行動を学習させる．

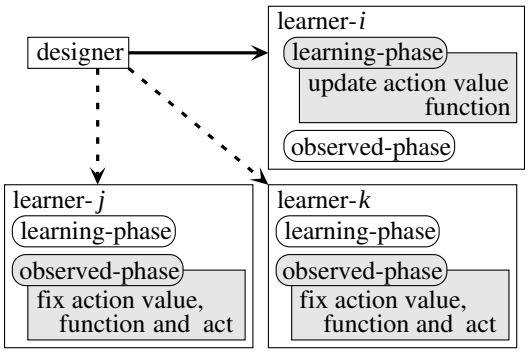


Fig. 3 Schedule for efficient learning in multi robots environments

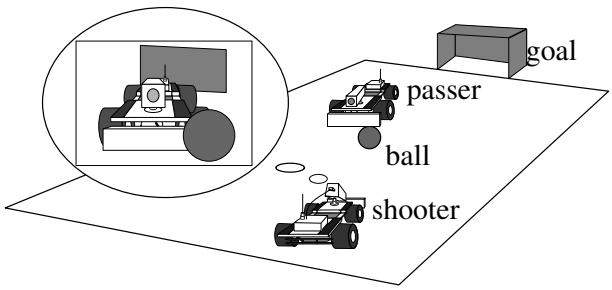


Fig. 4 Two robots and the environment

5. タスクと想定

5.1 対象となるシステム

マルチエージェントを研究する例題として近年ロボカップ [5] が提案されている．サッカーのようなゲームでは，ロボット間の関係には協調だけでなく競合も含まれている．そこで，提案手法を 2 台のロボットが存在する環境下での，簡単なサッカーゲーム (**Fig.4** 参照) に適用し，有効性を示す．環境内にはボール，ゴールと 2 台のロボット (シューターとパッサー) が存在し，ボール，ゴールの大きさはロボカップの中型リーグのルール [5] に従った．各ロボットは TV カメラを一つ搭載し，そこから得られる画像情報から環境の状況を観測する．シューターに与えられるタスクは「ボールをゴールにシュートする」ことであり，パッサーの場合は「ボールをシューターにパスする」ことである．また，両方のロボットは互いにできるだけ衝突を回避しなければならない．

モータコマンドとして，各ロボットは 2 自由度を持ち，行動ベクトル $\boldsymbol{u}$ は 2 次元ベクトル

$$\boldsymbol{u} = \begin{bmatrix} v \\ \phi \end{bmatrix}, \quad v, \phi \in \{-1, 0, 1\}, \quad (9)$$

として表現できる．ここで，v は台車の移動速度であり，ϕ はステアリングの角度である．実際に選択できる行動の合計は物理的に意味を持たない $(v, \phi) = (0, \pm 1)$ を除外した計 7 通りである．

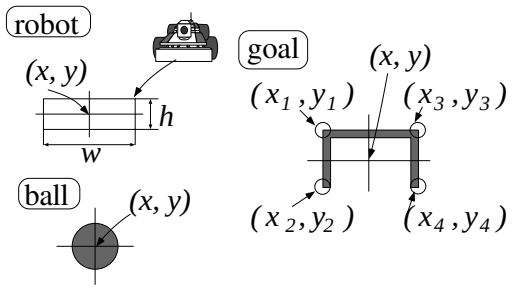


Fig. 5 Image features of the ball, goal, and other robot

各ロボットが観測できる観測ベクトル (画像特徴量) を説明する．画像特徴量の選択によっては，適切な状態ベクトルを構成できなくなるが，ここでは基本と考えられる画像特徴をすべて観測ベクトルとして利用した．具体的には

- ボール：画像上での重心座標 (x, y)，半径，面積，
- 相手ロボット：前面プレートの重心の座標 (x, y)，幅 w，高さ h，面積，
- ゴール：画像上での重心座標 (x, y)，四隅の座標 (x_i, y_i) $(i = 1, \cdots, 4)$，

のように設定した (**Fig.5** 参照)．結果として，ボール，ロボット，ゴールに関する観測ベクトルの次数はそれぞれ 4, 5, 11 となる．画像のサイズは利用した画像プロセッサの能力の限界から 64×60 画素であり，シミュレーションもこれに合わせた．

Fig.6 に構築した実システムを示す．各ロボットに搭載された TV カメラからの画像は，ビデオ送信器でホスト側の UHF 受信器に送られ，一つの画像に統合された後，パイプライン型画像処理装置 (MaxVideo200) で処理される．処理の簡単化と高速化のため，ボール，ゴール，ロボットの前面プレートはそれぞれ赤，青，黄色に塗装されている．画像処理や行動選択などは，ホスト CPU(MC68040) 上の OS(VxWorks) によって制御される．ホスト CPU はイーサネットを介して Sun ワークステーションに接続されている．

5.2 実験方法

計算機でのシミュレーションで，各ロボットは局所予測モデルを構築し，目的行動を学習する．1 試行は (1) ロボットがフィールドから出る，(2) ロボット間で衝突が発生する，(3) ボールがゴールに入る，のどれかの条件を満足した時点で終了し，パス行動が達成された†場合には試行を終了しない．シミュレーション環境でのスケジューリングは

- **期間 A**：シミュレーション環境で第 250×10^2 試行までパッサーが学習する．シューターは静止した状態，
- **期間 B**：第 500×10^2 試行までシューターが学習する．パッサーは期間 A で獲得された結果をもとに行動，
- **期間 C**：第 750×10^2 試行までパッサーが学習する．シューターは期間 B で獲得された結果をもとに行動，
- **期間 D**：第 1000×10^2 試行までシューターが学習する．パッサーは期間 C で獲得された結果をもとに行動，

と設定した．期間 A ～ D までを 1 セットとし，合計 10 セット行った．その中で最善の結果を実環境での追加学習のための初期値として用いた．実環境でのスケジューリングは

- **期間 E**：実環境に移行する．シューター，パッサーともにシミュレーションで獲得した局所予測モデルを用いる．第 100 試行まで，局所予測モデルの構築のためのデータ収集と目的行動の学習を同時に行なう††．
- **期間 F**：期間 E で獲得したデータから局所予測モデルを再構築し，第 150 試行まで，行動学習を行なう．

と設定した．結果として，パッサーは期間 A, C, E で，シューターは期間 B, D, E でそれぞれ局所予測モデルを構築することになる．各学習期間 $T_{\max}$ は，強化学習が終了した時点で切り替えることも可能である．しかし今回の実験では，10 セットのパフォーマンスの統計量を計算するために強化学習が収束するのに十分な長さとして，$T_{\max} = 250 \times 10^2$ とに設定した．

パッサーは，ボールをシューターにパスしたときに報酬 1 を受け取り，シューターはボールをゴールにシュートしたときに，報酬 1 を受け取る．さらに，ロボット間で衝突が発生した場合，-0.3 の報酬が与えられる．各報酬の絶対値の比率は，各タスクの優先順位に相当する．つまり，衝突回避行動よりもパス行動やシュート行動のほうが優先されていることになる．今回の実験では，これらの報酬値は経験的に決定した．計算機上での学習が終了した時点で，獲得された結果を実ロボットに適用する．行動選択のパラメータは $\alpha_{opt} = 80\%$ とした．

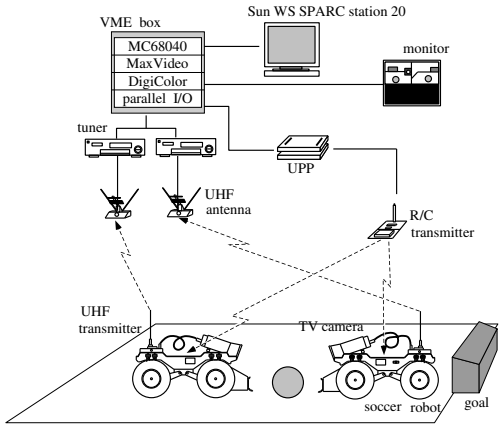


Fig. 6 A configuration of the real system.

6. 実 験 結 果

6.1 局所予測モデルの推定次数

Table 1 は計算機シミュレーションおよび実環境での推定された各対象物の状態ベクトルの次元 n と考慮する履歴長さ l を示している．ここで，シミュレーション → シミュレーション → 実環境での実験と表記している．

計算機と実環境で推定された状態ベクトルの次数 n や履歴長さ l が異なる理由として，

(1) 観測ノイズや行動の不確実性のために，実環境での予測誤

†本論文では，一方のロボットが蹴ったボールを他方が一定時間内に蹴った場合，パスが成立したとみなす．

††実環境での行動学習は，実験システムの都合上，1 試行ごとに各ロボットの観測と行動データをもとに，バッチ処理的に行動価値を更新した．

Table 1 Differences of the estimated dimension (simulation → simulation → real experiments)

observer	target	estimated dimension n	historical length l
shooter	goal	$2 \to 2 \to 3$	$1 \to 1 \to 1$
	ball	$4 \to 4 \to 4$	$2 \to 2 \to 4$
	passer	$6 \to 6 \to 4$	$3 \to 3 \to 5$
passer	ball	$4 \to 4 \to 4$	$2 \to 2 \to 4$
	shooter	$5 \to 5 \to 4$	$3 \to 3 \to 5$

差は，計算機上での予測誤差よりも大きく，状態の次数を増加しても推定精度が向上されない，

(2) 各ロボットはその時点での学習結果に基づいて行動するため，局所予測モデル構築のためのデータには，その時点での学習結果を反映したタスク依存の偏りがある，

ことが挙げられる．例えば，実環境での履歴長さ l は，シミュレーションでの履歴長さよりも長くなる傾向にある．一方で，ゴールの場合は次元が増えることで誤差を低下させることができるのに対し，他ロボットの場合はシミュレーションよりも次元が低くなっている．

また，**Fig.7** (a), (b), (c) に，期間 F におけるシューターから見たゴール，ボール，パッサーの特徴量 (画像上での重心の y 座標) の予測誤差を示す．図からもゴールに関する誤差が最も小さく，パッサーに対する誤差が最も大きい．各対象物に対する誤差に関しては

- ゴール：ゴールは静止しているため，学習者 (この場合はシューター) との相互作用が簡単に推定できている．15 sec で予測誤差が増加しているのはボールによりゴールの大半が隠されたためである，
- ボール：ボールは能動的に行動することはないが，局所的に推定するだけでもうまく運動を推定できている，
- パッサー：パッサーは能動的に行動できるために，その相互作用は一番複雑である．予測誤差も大きい理由として，シューターがボールとパッサーの関係を無視して局所的に状態ベクトルを推定しているためである，

といったことがわかる．

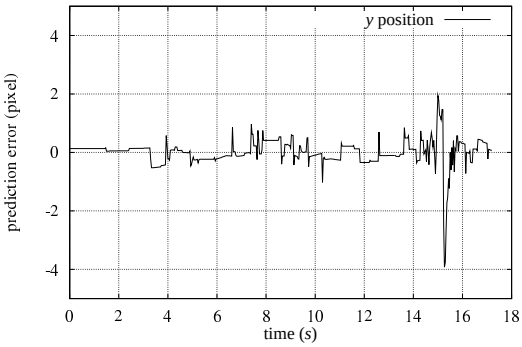
6.2 パフォーマンスに関する考察

次に，獲得されたシュート行動とパス行動のタスク達成率について考察する．提案手法と，従来法 (センサ情報だけから状態空間を構成した) 場合のタスク達成率の平均値と標準偏差を **Fig.8** (a), (b) に示す．期間 A ではパッサーは静止したシューターに向かってボールを蹴るだけであり，パフォーマンスの本質的な違いはボールに関する内部表現だけに集約され，パスの成功確率はそれほど差が無い．

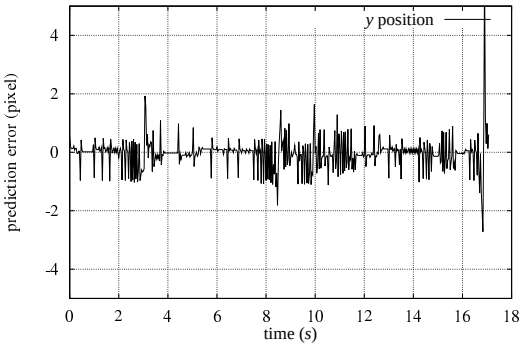
しかし期間 B でシューターが学習を開始した場合，提案手法ではパス行動の成功確率が一度下降するが，学習を続けることでシュート，パスの成功確率は次第に上昇しているのに対し，従来法では，期間が変わると成功確率が大きく変動する．このことから，局所予測モデルは提案した学習のスケジューリングのもとでは，適切な協調行動を獲得することが実験的に示された．

Table 2 に，実環境での学習中のパフォーマンスの比較結果を示す．これからシミュレーション結果をそのまま適用するよりも，パフォーマンスが改善されていることがわかる．実環境では複数の学習者が同時に学習してもうまく行動を獲得できた理由としてシミュレーションで大体の行動が獲得されており，実環境ではその微調整しているだけであり，最適行動を $\alpha_{opt} = 80\%$ と高い割合で選択したためである．

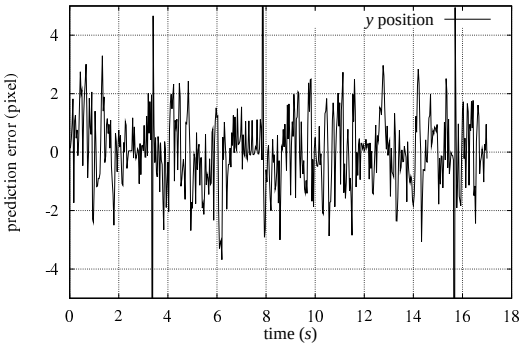
また，実環境で獲得されたシューターとパッサーのボールに関する局所予測モデルを交換した場合のビヘービアを観察したところ，両方のロボットにおいて局所予測モデルの予測誤差が大きくなり，適切な行動を生成できなかった．これは学習者の身体の個体差だけでなく，経験の違いのために生じる現象であり，このことは，実ロボットにおいては推定された局所予測モデルは交換不可能なことを示している．



(a) y position of the goal image

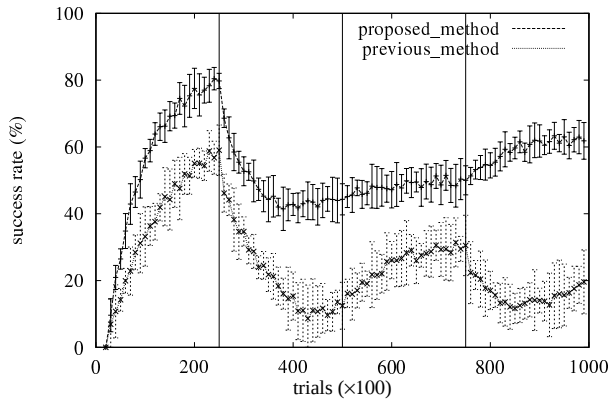


(b) y position of the ball image

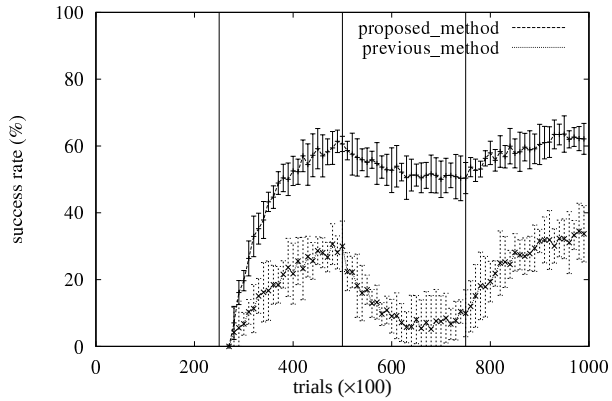


(c) y position of the other robot image

Fig. 7 Estimation based on the local predictive models in the real environment



(a) passing behavior



(b) shooting behavior

Fig. 8 Success rates in computer simulation

Table 2 Performance results in real experiments

	period E	period F
success rate of shooting	57/100	32/50
success rate of passing	30/100	22/50
number of collisions	25/100	6/50

最後に，提案手法により獲得されたシミュレーションおよび実環境での行動例をそれぞれ **Fig.9, 10** に示す．まず，パッサーがボールをシューターに向かってボールを蹴り，シューターはボールをゴールにシュートする．パッサーはボールを蹴った後は，シューターとの衝突を回避するための行動をしていることがわかる．

7. 結論と今後の課題

本論文では，強化学習を複数のロボットが存在する環境下に適用するための手法について提案した．局所予測モデルは観測と行動の相互作用を通して，学習者と他の対象物との間の局所的な関係を獲得し，強化学習部は大局的な局所予測モデル間の関係を学習することで，他者の行動の曖昧性を解消した．2 台のロボットのサッカーゲームに適用し，ボールのパス行動，シュート行動，衝突回避行動などの行動を実現し，本手法によって協

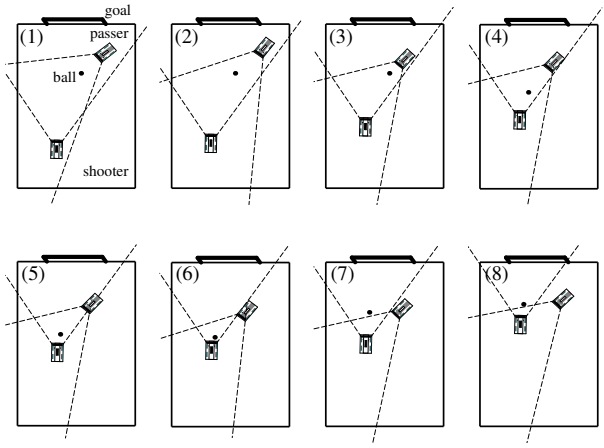
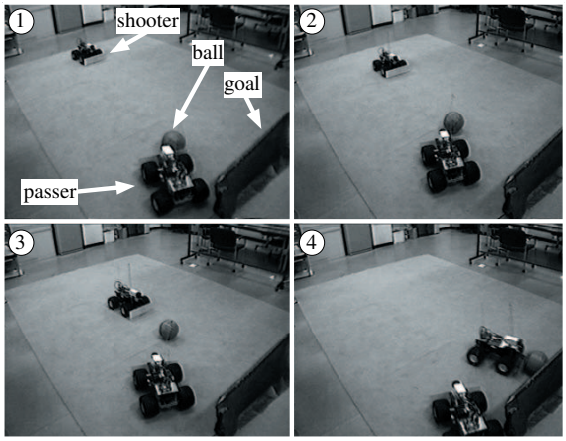
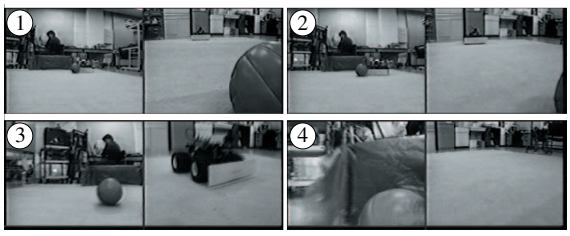


Fig. 9 Acquired cooperative behavior in computer simulation



(a) top view



(b) obtained images (left: shooter, right: passer)

Fig. 10 Acquired cooperative behavior in the real environment

調行動が獲得できることを示した．

本手法は状態空間のセグメンテーションの問題は扱っていないが，これは Parti-game アルゴリズム [11] などの方法を適用することも考えられる．今後の課題として，3 台以上のロボットが存在する環境下で，協調，競合行動を学習させるタスクに適用することが考えられる．我々は提案手法により獲得される行動を基本のビヘービアとして，共進化的手法を適用 [19] しており，その有効性を確認している．また，今回は各学習者に与

えられるタスクは明示的に区別されていたが，自律分散の立場からすると，相互作用を通して各学習者の役割を推定すべきである．そのため，他者の行動を観測と行動を通して理解する枠組みが必要であり，局所予測モデルを下位の構造として構築することが今後の課題である．

謝　　辞

本研究は，日本学術振興会 未来開拓学術研究推進事業「分散協調視覚による動的 3 次元状況理解」プロジェクト (課題番号 JSPS-RFTF96P00501) の補助を受けた．

参 考 文 献

[1] H. Akaike. A New Look on the Statistical Model Identification. *IEEE Trans. AC-19*, pp. 716–723, 1974.
[2] 荒井, 宮崎, 小林. マルチエージェント強化学習の方法論 — Q-learning と Profit sharing による接近—. 人工知能学会誌, 13(4):609–618, 1998.
[3] M. Asada, S. Noda, S. Tawaratumida, and K. Hosoda. Purposive Behavior Acquisition for a Real Robot by Vision-Based Reinforcement Learning. *Machine Learning*, 23:279–303, 1996.
[4] 石田, 片桐, 桑原. 並列処理シリーズ 11. 分散人工知能. コロナ社, 1996.
[5] H. Kitano ed. *RoboCup-97 : Robot Soccer World Cup I*. Springer Verlag, 1997.
[6] W. E. Larimore. Canonical Variate Analysis in Identification, Filtering, and Adaptive Control. In *Proc. 29th IEEE Conference on Decision and Control*, pp. 596–604, Honolulu, Hawaii, December 1990.
[7] L.-J. Lin and T. M. Mitchell. Reinforcement Learning with Hidden States. In *Proc. of the 2nd International Conference on Simulation of Adaptive Behavior : From Animals to Animats 2.*, pp. 271–280, 1992.
[8] M. L. Littman. Markov Games as a Framework for Multi-agent Reinforcement Learning. In *Proc. of the 11th International Conference on Machine Learning*, pp. 157–163, 1994.
[9] P. Maes and R. A. Brooks. Learning to Coordinate Behaviors. In *Proc. of AAAI-90*, pp. 796–802, 1990.
[10] T. Matsuyama. Cooperative Distributed Vision. In *First International Workshop on Cooperative Distributed Vision*, pp. 1–28, 1997.
[11] A. W. Moore and C. G. Atkeson. The Parti-Game Algorithm for Variable Resolution Reinforcement Learning in Multidimensional State-Spaces. *Machine Learning*, 21:199–233, 1995.
[12] T. Ohko, K. Hiraki, and Y. Anzai. Reducing Communication Load on Contract Net by Case-based Reasoning — Extension with Directed Contract and Forgetting. In *Proc. of the 2nd International Conference on Multi-Agent Systems*, pp. 244–251, 1996.
[13] T. W. Sandholm and R. H. Crites. On Multiagent Q-learning in a Semi-Competitive Domain. In *Workshop Notes of Adaptation and Learning in Multiagent Systems Workshop, IJCAI-95*, 1995.
[14] R. G. Smith and R. Davis. Frameworks for Cooperation in Distributed Problem Solving. *IEEE Transaction on Systems, Man and Cybernetics*, 11(1):61–70, 1981.
[15] L. Steels and P. Vogt. Grounding Adaptive Language Games in Robotic Agents. In *Fourth European Conference on Artificial Life*, pp. 474–482, 1997.
[16] P. Stone and M. Veloso. Using Machine Learning in the Soccer Server. In *Proc. of IROS-96 Workshop on Robocup*, 1996.
[17] Y. Sugita and J. Tani. Emergence of Cooperative / Competitive Behavior in Two Robots' Games : Plans or Skills? In *SAB-98 Workshop 1 : Adaptive Behavior using Dynamic Reccurent Neural Nets*, 1998.
[18] R. S. Sutton and A. G. Barto. *Reinforcement Learning : An Introduction*. MIT Press, 1998.
[19] 内部英治, 浅田稔, 中村理輝. 共進化によるマルチ移動ロボット環境における協調行動の獲得. 北野宏明編. 遺伝的アルゴリズム 4, 第 7 章, pp. 193–220, 産業図書, 2000.
[20] E. Uchibe, M. Asada, and K. Hosoda. Behavior Coordination for a Mobile Robot Using Modular Reinforcement Learning. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1329–1336, 1996.
[21] P. van Overschee and B. de Moor. *Subspace Identification for Linear Systems*. Kluwer Academic Publishers, 1996.
[22] S. D. Whitehead and D. H. Ballard. Active Perception and Reinforcement Learning. In *Proc. of Workshop on Machine Learning*, pp. 179–188, 1990.

付録 A. CVA に基づく局所予測モデルの構築

CVA [6] は多入力多出力のシステムを入力と出力のシーケンスから直接同定する部分空間同定法の一手法であり，状態ベクトルの計算法を以下に示す．

1.観測と行動のシーケンス $\{\boldsymbol{u}(t), \boldsymbol{y}(t)\}, t = 1, \cdots N$ に対して，新しい状態ベクトル $\boldsymbol{p}(t)$，$\boldsymbol{f}(t)$ を式 (2) によって作成する．

2.共分散行列の推定値 $\hat{\boldsymbol{\Sigma}}_{pp}$，$\hat{\boldsymbol{\Sigma}}_{pf}$，$\hat{\boldsymbol{\Sigma}}_{ff}$ を計算する．

3.次の特異値分解

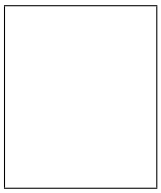
$$\hat{\boldsymbol{\Sigma}}_{pp}^{-1/2}\hat{\boldsymbol{\Sigma}}_{pf}\hat{\boldsymbol{\Sigma}}_{ff}^{-1/2} = \boldsymbol{U}_{aux}\boldsymbol{S}_{aux}\boldsymbol{V}_{aux}^T, \quad \text{(A.1)}$$
$$\boldsymbol{U}_{aux}\boldsymbol{U}_{aux}^T = \boldsymbol{I}_{l(m+q)},\ \boldsymbol{V}_{aux}\boldsymbol{V}_{aux}^T = \boldsymbol{I}_{kq},$$

を計算する．また，$\boldsymbol{M}$ を

$$\boldsymbol{M} := \boldsymbol{U}_{aux}^T\hat{\boldsymbol{\Sigma}}_{pp}^{-1/2}, \quad \text{(A.2)}$$

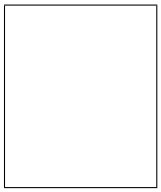
と定義する．

4.式 (3) を用いて n 次元の状態ベクトル $\boldsymbol{x}(t)$ を計算する．



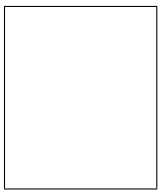
内部 英治（Eiji Uchibe）

1972 年 2 月 17 日生．1999 年大阪大学大学院電子制御機械工学専攻博士後期課程終了．同年，日本学術振興会，未来開拓学術研究推進事業「分散協調視覚による動的 3 次元状況理解」プロジェクトの研究員．2001 年 4 月科学技術振興事業団 ERATO 川人学習動態脳プロジェクト計算学習グループ研究員，2001 年 10 月 ATR 人間情報科学研究所第三研究室の研究員となり現在に至る．1998 年度人工知能学会研究奨励賞受賞．強化学習，進化的手法を用いたロボット学習の研究に従事．人工知能学会，IEEE, AAAI の会員．（日本ロボット学会正会員）



浅田 稔（Eiji Uchibe）

1953 年 10 月 1 日生．1982 年大阪大学大学院基礎工学研究科博士後期課程修了．同年同大学基礎工学部助手．1989 年同大学工学部助教授，1995 年同教授となり現在に至る．この間，1986 年から 1 年間米国メリーランド大学客員研究員．1989 年情報処理学会研究賞，1992 年 IEEE/RSJ IROS'92 Best Paper Award，1996 年日本ロボット学会第 10 回論文賞受賞．知能ロボットの研究に従事．工学博士．人工知能学会，電子情報通信学会，情報処理学会，日本機械学会，計測自動制御学会，システム制御情報学会，IEEE などの会員．（日本ロボット学会正会員）



細田 耕

1965 年 11 月 9 日生．1988 年京都大学工学部精密工学科卒業．1993 年同大学大学院工学研究科機械工学専攻博士後期課程修了．同年大阪大学工学部助手，1997 年同大学助教授となり現在に至る．この間，1998 年チューリッヒ大学客員教授．1995 年日本ロボット学会研究奨励賞，1996 年日本ロボット学会第 10 回論文賞受賞．視覚サーボ系，知能ロボットの研究に従事．博士 (工学)．IEEE，計測自動制御学会の会員．

（日本ロボット学会正会員）