Vol.2017-GI-37 No.3
2017/3/7

RoboCup Soccer における Q 学習を用いた対人守備の強化

水島諒[†1]　穴田一[†1]

概要：近年,「ゲーム AI」の開発が盛んである．そして，チェスや将棋，囲碁といったゲームにおいて AI が人間のチャンピオンに勝利するといった事も起きている．サッカーゲームではロボカップと呼ばれる世界大会が行われており，移動可能範囲が広いことや，計算時間が制限されていること，複数人同士の対戦であることから，前述のゲームより難しいと考えられている．そして，ロボカップで使用されているチームモデルの多くは 1 対 1 の場面における守備などの局所的な場面に対応していない．しかし，実際のサッカーでは 1 対 1 の状況が起きやすく，対人守備が重要視されている．そこで，本研究では Q 学習を用いて 1 対 1 における守備を学習する方法を学習し，その有効性を確認した．

キーワード：RoboCup，Q 学習，人工知能

Person to Person Defense using Q learning for RoboCup Soccer

RYO MIZUSHIMA[†1]　HAJIME ANADA[†1]

1. はじめに

近年,「ゲーム AI」の開発が盛んに行われている．例えば，チェスや将棋，囲碁といったゲームが挙げられる．そして，これらのゲームにおいては AI が人間のチャンピオンに勝利するといった事も起きている．また，RoboCup と呼ばれる，自律型ロボットによるサッカーの世界大会が毎年行われている[1]．RoboCup とは，西暦 2050 年迄にサッカーの世界チャンピオンチームに勝てる，自律型ロボットのチームを作ることを目標とした大会である．この RoboCup には 5 つのリーグがあり，リーグごとに異なる特徴がある．本研究では 5 つのリーグのうち，各選手がそれぞれ思考し，人間のような戦術的なサッカーが行われている 2D リーグを扱う．2D リーグは，移動可能範囲が広いこと，リアルタイムに計算し判断を下す必要があること，11 人同士の対戦であることなどから，前述のチェスや将棋，囲碁といったゲームより難しいと考えられている．

秋山は RoboCup の 2D リーグで使用可能な agent2d (Ver 3.1.1) というチームモデルを公開している[2]．このチームモデルでは，全てのエージェントがボールの位置のみを使用し，移動先を決定するという，秋山が開発したフォーメーションシステムを用いている．しかし，このシステムではボールの位置が同じであれば，どのような戦況においても同じポジショニングを目指すという問題がある．また，守備の際に間合いを取り，相手に抜かれないような行動を取ることが多く，守備の基本的な動きであるボールを奪う動きが少ない．そこで，本研究では人間のサッカーの対人守備の練習メニューを取り入れ，Q 学習を用いてエージェントにボールを奪う守備を学習させ，その有効性を確認した．

2. 既存研究のチームモデル

秋山は RoboCup の 2D リーグで使用可能な agent2d (Ver 3.1.1) というチームモデルを公開している[2]．agent2d は，エージェントの移動先の決定を，秋山が提案したフォーメーションシステムを用いて行う．

2.1 フォーメーションシステム

秋山は，エージェントの移動先の決定を行うためにフォーメーションシステムを提案した．フォーメーションシステムでは，全てのエージェントがボールの位置のみを用いて移動先を決定している．このシステムでは，事前にフィールド上の複数の位置に，その位置にボールがあった場合の 11 人のエージェントの最適位置を，それぞれ設定している．そして，エージェントの移動先の決定方法は，フォーメーションシステムで設定した位置にボールがある場合と，ない場合で異なる．以下にそれぞれの移動先の決定方法を記す．

- フォーメーションシステムで設定した位置にボールがある場合

全エージェントは，今いる場所からフォーメーションシステムで事前に設定したフォーメーションに従って，指定された目標位置を目指して進む．図 2.1 はある設定した位置にボールがあった時の移動先の決定した例である．

†1 東京都市大学
Tokyo City University

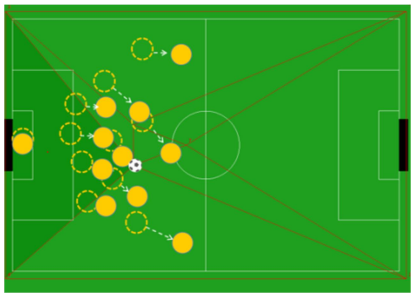

図 2.1. フォーメーションシステムを用いた
移動先の例

自チームのゴールが左で相手のゴールは右である．そして，フィールド上にある ◯ はエージェントの今いる場所を表し，● はエージェントの移動先の場所を表し，⚽ はボールを表す．各エージェントは今いる ◯ の位置から指定された移動する先の ● の位置を目指して移動する．

- フォーメーションシステムで設定した位置にボールがない場合

全エージェントは，フォーメーションシステムで事前に設定したボールの位置の集合から，ドロネー図を用いてフィールドを三角形に分割し，ボールを含む三角形の頂点 3 点を選ぶ．選んだ 3 点には，それらの位置にボールがある場合のフォーメーションを事前に設定してある．これら 3 つのフォーメーションを用いてフォーメーションの補間を行い，今いる場所から補間を行ったフォーメーションに従って，目標位置を目指して進む[5]．図 2.2 はドロネー図を用いてボールを含む三角形の頂点 3 点を選び，その 3 点のフォーメーションとそこから補間して求まったフォーメーションの例である．

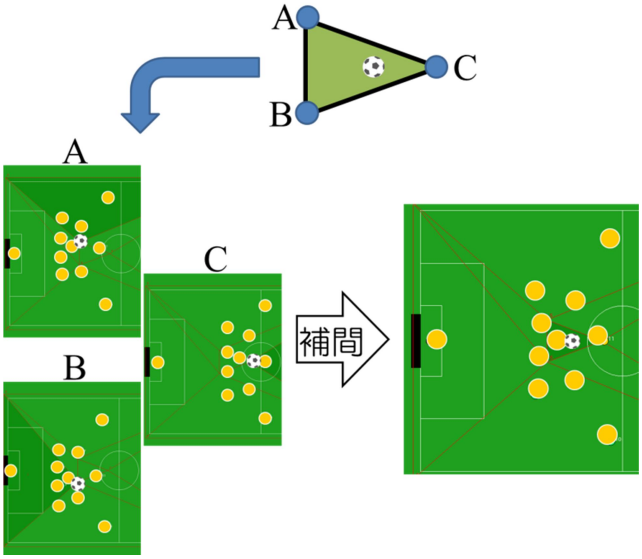

図 2.2. フォーメーションシステムでの補間の例

⚽のボールを含む三角形 ABC からフォーメーションを補間する例である．これらの図は全て自分のゴールが左であり，左側のフィールド半面を切り取った図となっている．そして，フィールド上にある●は味方エージェントを表す．ボールを含む三角形の頂点 A, B, C で設定してあるフォーメーションを用いて補間を行い，補間したフォーメーションを用いて移動先を決定する．

2.2 既存研究のチームモデルの問題点

ボールの位置のみを用いて移動先を決定しているため，ボールの位置が同じであれば，どのような戦況においても各エージェントが同じポジショニングを目指してしまうという問題がある．また，守備の際に間合いを取り，相手に抜かれないような行動を取ることが多く，守備の基本的な動きであるボールを奪う動きが少ない．

3. 提案モデル

ボールを奪う動きが少ないという問題を解決するため，Q 学習を用いて 1 対 1 の状況におけるボールを奪う守備を学習する．守備の学習メニューとしては，実際のサッカーで使用される練習メニューを用いる．

チームモデルの基本となる動きは既存のフォーメーションシステムを用いて移動先を決定する．しかし，1 対 1 の場面においては学習結果を用いて移動先を決定する．

3.1 練習メニュー

実際のサッカーでは，1 対 1 の状況の練習メニューとして，図 3.1 のようなものがある．攻撃側のエージェントは左の赤色のラインの外にドリブルでボールを出すよう攻める．守備側のエージェントは左のライン以外の青色のライン外に出すように守るという練習方法である．

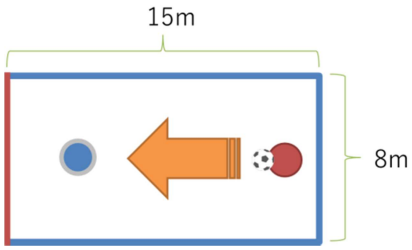

図 3.1. 1 対 1 の練習メニュー

横幅が 15m で縦幅が 7m である．●は攻撃側のエージェントであり，左の赤色のラインの外にドリブルでボールを出すよう攻める．●は守備側のエージェントであり，左のライン以外の青色のライン外にボールを出すように守る．この練習メニューは実際の人間の練習メニューと同じである．

図 3.1 の練習メニューを Q 学習で使用するため，練習メニューで用いるフィールド座標系は図 3.2 のようにした．赤色ライン中央を原点とし，長辺方向を x 軸，短辺方向を y 軸とした直交座標系で表す．

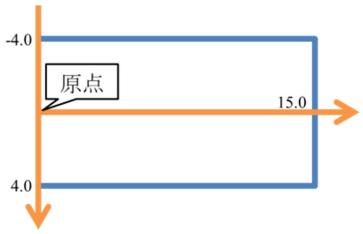

図 3.2. 1 対 1 の練習メニューで用いるフィールドの座標系

図の左中央を原点とし，長辺方向を x 軸，短辺方向を y 軸とした直交座標系となっている．

3.2 Q 学習の適用方法

3.2.1 Q 学習の適用方法

本研究で用いる Q 学習の基本的な枠組みを述べる．Q 学習では，エージェントは現在の環境の状態 S_t を観測し，実行すべき行動 a を選択する．行動 a による環境の変化に応じた報酬 r を受取り，環境の状態は S_{t+1} に変化する．その時，次式を用いて状態 S_t における行動 a の行動価値 $Q(S_t, a)$の更新を行う．

$$Q(S_t, a) \leftarrow Q(S_t, a) + \alpha\left[r + \gamma \max_p Q(S_{t+1}, p) - Q(S_t, a)\right] \quad (1)$$

ここで， α は学習率， r は報酬の量， γ は割引率を表し， $\max_p Q(S_{t+1}, p)$ は状態 S_{t+1} における可能な行動の中の最大の行動価値を表す．エージェントは，観測と行動選択を繰り返すことにより，守備に有効な行動に対する行動価値$Q(S_t, a)$を更新していく．

Q 学習を用いるために，エージェントの報酬 r ，エージェントの観測できる状態 S_t と選択できる行動 a を定義しなければならない．それぞれについて後述で詳しく述べる．

3.2.2 報酬の設計

良い行動をした時には目的の達成度合に応じた報酬を与える．本実験は，図 3.1 の青色のライン外にボールを出し，赤色のライン外にボールを出されないことが目的である．そこで，この目的を達成するために次の 5 つの小目的を設定した．

I. ボールを奪う
II. コート外に出ない
III. ボールに対して正面を向く
IV. ボールを後方に戻させる
V. ボールに 1m まで近づく

この 5 つの小目的それぞれの達成具合を表す報酬 r を次式のように定義した．

$$r = \sum_{i=1}^{5} reward_i \quad (2)$$

$$\begin{cases} rew\quad_1 = \begin{cases} 10000 & if\ Take\ the\ ball \\ 0 & otherwise \end{cases} \\ reward_2 = \begin{cases} -100 & if\ Out\ of\ the\ field \\ 0 & otherwise \end{cases} \\ reward_3 = \max\left(\dfrac{90.0 - |\theta_{sb}|}{6.0}, 0.0\right) \\ reward_4 = \max(x_b - prebx_b, 0.0) \times 5.0 \\ rewar\ _5 = \min(15.0 - dist_{sb}, 15.0 - 1.0) \end{cases}$$

ここで，$reward_i$ 小目的 i に応じた報酬を表す．そして，$rewar\ _i$ の上限値が 15.0 前後となるように調整した． x_b はボールの x 座標， $prebx_b$ は 0.1 秒前のボールの x 座標，$dist_{sb}$ は自分とボールとの距離を表す．そして，θ_{sb} は自分とボールが x 軸に対して成す角を表す．図 3.3 に報酬のための変数設定を示す．

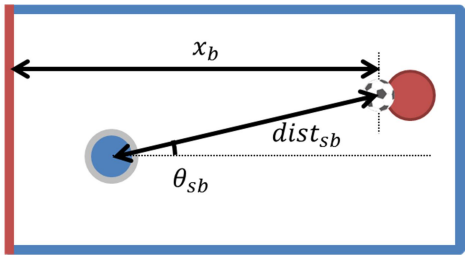

図 3.3. 報酬で用いる変数設定

は攻撃側のエージェントであり，は守備側のエージェントである．x_b はボールの x 座標，$dist_{sb}$ は自分とボールとの距離， θ_{sb} は自分とボールが x 軸に対して成す角を表す．

3.2.3 状態の定義

守備を行う際，ボールの位置や自分の位置，ラインからの距離などを考慮して行動を選択しなければならない．そこで，守備行動を学習するための状態を次のように定義する．

- ボールが自分から見て 8 方向の内，どの方向にあるか
- ボールの y 座標が正の値か負の値か
- 自分の y 座標が正の値か負の値か
- ボールが自分から 1.5m 以内にあるか
- ボールが自分から 5.0m 以内にあるか
- 図 3.1 の青色のラインからボールまでの距離が 1.5m 以内か
- 敵とボールの距離が 1.5m 以内か

以上のように状態を定義すると，エージェントが識別する状態は 1024 種類となる．

3.2.4 行動の選択

エージェントが可能な行動は 8 方向に歩く，走る，又はタックル，動かない，の計 18 種類とする．学習中の行動の選択は，よりよい行動を探すために， ε の確率でランダムに行動を選択し， $(1-\varepsilon)$ の確率でそれまでに獲得した最適な行動を選択するように設定した．

4. モデルの評価

4.1 実験設定

本研究では，Q 学習で一般的に用いられるパラメータを使用し，学習率 α は 0.1，割引率 γ は 0.9，行動の選択確率 ε は 0.3 とした．守備側のエージェントの初期配置は，(0.5, 0.0)の位置とし，攻撃のエージェントの初期配置は，(14.5, 0.0)とした．攻撃側のエージェントには，既存研究のチームモデル agent2d (Ver 3.1.1)のエージェントを使用した．

学習方法は，3 秒間練習メニューを用いた練習を行い，それ迄にコート外に出すことが出来なかった際は，初期配置に戻し，練習を行う．また，30 秒以内にコート外にボールが出た場合も同様に初期配置に戻し，練習を行う．これを行動価値 $Q(S_t, a)$ が収束するまで繰り返す．図 4.1 に練習の繰り返し回数における各状態の最大の行動価値を合計した値を示す．

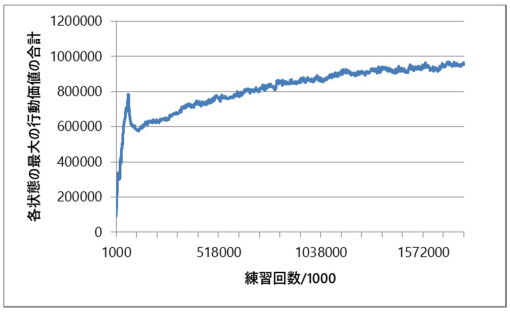

図 4.1. 練習回数と各状態の最大行動価値の合計値の関係

横軸は練習の繰り返し回数を 1000 で割った値，縦軸は各状態の最大の行動価値を合計した値を表す．練習を繰り返すと値が高くなり，収束していくのがわかる．

この図から，練習を繰り返すことで学習が進み，時間が経つと収束していくことが分かる．次に，守備側のエージェントが agent2d のエージェントの場合，学習後のエージェントの場合で，それぞれ先ほどと同じ練習メニューを 20 回行った．どちらも攻撃側のエージェントは agent2d のエージェントを用いた．その結果を表 4.1 に示す．

表 4.1. 練習メニューにおける成功確率

	守備側 攻撃側	agent2d agent2d	守備側 攻撃側	学習後 agent2d
成功確率	6%		94%	

この表から，学習が上手くいったことが分かる．そこで，実際の対戦の場合に学習結果がうまく適用できるかを確かめる．本研究では対戦時に学習結果を適用する条件として，1 対 1 の状況でかつ前方 15m 以内に敵がいる状況とし，適用条件を満たした際には 10 秒間学習結果を適用した．agent2d と学習後のエージェントで構成したチームがそれぞれ agent2d と 200 回対戦し，平均失点数を調べた結果を表 4.2 に示す．

表 4.2. agent2d と 200 回対戦した結果

	agend2d vs agent2d	学習後 vs agent2d
平均失点数	2.2	7.8

この表から，学習後のエージェントで構成したチームの方が既存のチームモデルである agent2d より平均失点数が高くなり，弱くなってしまったことが分かる．これは，複数人同士の練習メニューを取り入れていないため，学習結果を適用している選手の後ろにいる相手にパスをされてしまい，相手をフリーの状況にしてしまうことが多くなってしまったことが原因であった．そのため，学習結果の適用条件や，練習メニューについて検討しなければならない．

5. 今後の方針

本研究では，対戦時に学習結果を適用する条件の設定が上手くいかなかったと考えられるため，より良い適用条件を検討していかなければならない．そして，実際のサッカーでは周りと協調して守備を行う．そのため，練習メニューを拡張し，ボールを奪いに行く選手のフォローを行う選手の守備方法を学習するなど，複数人同士の学習方法も考えていかなければならない．

参考文献

[1] Hiroaki Kitano, Minoru Asada, Yasuo Kuniyoshi, Itsuki Noda, Eiichi Osawa and Hitoshi Matsubara, “RoboCup: A Challenge Problem for AI”, AI Magazine , Vol.18, No.1, 1997, pp.73-85.
[2] 秋山 英久，“アクション連鎖探索によるオンライン戦術プランニング”, 人工知能学会研究会資料, SIG-Challenge-B101-6, pp.23-28 (2011).
[3] “ロボカップ日本委員会”, http://www.robocup.or.jp/original/about.html, (参照 2016-10-31).
[4] 大島真樹 : ” Java でつくる RoboCup サッカーエージェントプログラム”, 森北出版株式会社（2005）.
Hidehisa Akiyama, Hiroki Shimora, and Itsuki Noda: HELIOS2009 Team Description. In Robocup 2009 (2009).