サッカーエージェントによる協調パスプレイの生成

Creating Cooperative Pass Play with Soccer Simulation Agent

秋山 直之　鈴木 恵二　山本 雅人　大内 東

北海道大学大学院工学研究科システム情報工学専攻複雑系工学講座調和系工学分野

Abstract

The purpose of this reseach is to create the cooperative pass play with soccer simulation agent.

The cooperative pass play, like "wall pass" or "through pass", needs to synchronize passer's intending space to pass with receiver's intending space to receive the pass. In this recearch, we propose the method for realizing the play.

In addition, we adopt the hybrid learning system - combined neural network, learning classifier system, and Q-learning - proposed by Robert E. Smith in order that soccer agents may learn autonomously to decide appropriate space in accordance with the environmental states to complete pass play.

By using the agent implemented the method, we try to realize the cooperative pass play in the two-on-one situation on the RoboCup Soccer Server.

1　はじめに

近年マルチエージェント研究の例題として，サッカーがさかんに取り上げられている．これは，サッカーが世界的に見て最もポピュラーなスポーツであるため例題として判りやすく，またAI並びにマルチエージェントの例題として，興味深い課題を含んでいるからである．

マルチエージェント研究としてのサッカーに関する共通の土台を提供し，サッカーを共通問題として取り上げていく環境を設定するために，RoboCup [4] が提案された．RoboCupでは，実機部門，シミュレーション部門，エキスパート部門の三つの部門からなっている．この内のシミュレーション部門では，計算機上に仮想的なフィールドを構成して，その上のプレイヤーをプログラムで制御して試合をすることで，マルチエージェント環境における分散制御や協調動作の技術を競わせることを主眼としている．

シミュレーションにおいて，学習や進化を用いてサッカーエージェントに様々な協調行動をとらせようとする研究が盛んに行なわれている．例として，以下のようなものが挙げられる．

- ニューラルネットワークを組み込まれたエージェントが環境に応じてポジションを変更する [Andou, 1997][6]
- ニューラルネットワークを用いて動いているボールを捉える技能を高め，そのエージェントがID3によって，状態に応じてパスをするべき味方を選択する [Stone, 1997][7, 8]
- GPを意思決定関数として組み込んだエージェントによる協調行動を生み出す [Luke, 1998][9]

サッカーにおける最も代表的な協調行動である「パスプレイ」を考えると，現状ではボール保持者がプレイを決定しているが，実世界では必ずしもそうではなく，周囲にいるレシーバーがプレイを決定する事が多い．例えば，スペースへのスルーパスなどでは，レシーバーが動き出したり，手や声で合図を出したりすることによってパスの方向を決定し，ボール保持者はそれに合わせてパスをだすことになる．このようなプレイは，RoboCupではまだ見られてはいない [11]．

そこで，本研究ではこのようなパスプレイに注目し，RoboCupの公式シミュレーションサーバーであるSoccer Server上において，「スルーパス」，「壁パス」に代表されるような，レシーバーの意図も反映された高度な協調パスプレイを，サッカーシミュレーションエージェントに実現させるための手法について示す．また，本手法では，Robert E. Smithによって提案されたニューラルネットワーク，クラシファイアーシステム,Q-learningを組み合わせた学習システム [10] をエージェントに組み込み，フィールド状況に応じたパスプレイを成立させるために取るべき行動を，自律的に獲得させる事を試みる．

2　協調パスプレイ

パスはサッカーにおける最も基本的な協調行動である．複数のプレイヤーの間でボールをうまくパスするためには，パスを出すプレイヤー (パサー) とパスを受けるプレイヤー (レシーバー) との間で，互いに意図した行動が同調していなければならない．例えば，最も単純な状況であろう攻撃側2人対守備側1人の状況であっても，敵守備者をパスプレイによって突破することを考えた場合，パ

スプレイとしては少なくとも壁パス (Figure 1), スルーパス (Figure 2) の 2 種類が考えられる.

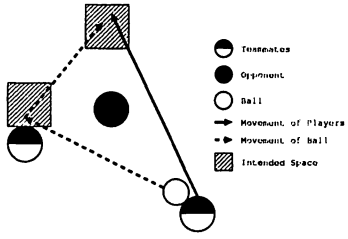

Figure 1: 壁パス

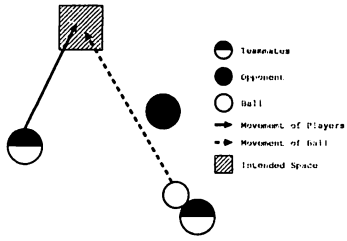

Figure 2: スルーパス

敵守備者を突破するためには, まずこれらのパスプレイのどちらを行なうかについての, 2プレイヤーの意思決定が同調していなければならない. また, 2 プレイヤーが意図するパスプレイに関して同調しても, パサーがパスを出すと意図した場所と, レシーバーがパスを受けるために走り込むと意図した場所が同調していなければ, パスがつながることはない.

本研究において目標とする協調パスプレイとは, この 2 つの例のような敵守備を突破するのに有効である, 「プレイヤーが走り込んだ先へのパス」を指す.

3 問題設定

協調パスプレイの生成を検証するための例題として, SoccerServer 上に以下に述べるような単純な 2 対 1 の状況を設定した.

この問題においては, まず, 攻撃側のエージェント「アタッカー」が二人存在する. アタッカーの目的は, 協調パスプレイによって得点をあげる事である.

アタッカーに対抗するエージェントとして「ディフェンダー」が一人存在する. その目的は, アタッカーが得点するのを彼らのパスをインターセプトすることによって妨げる事にある. 本問題におけるディフェンダーの行動戦略は単純で, ボールを追いかけるのみである.

本問題における行動範囲は, フィールドの敵陣内に限定される. また, パスを介さない得点 (開始地点からのロングシュート等) を防ぐ為に, ゴール近辺の領域を「シュートエリア」(Figure 3) とし, アタッカーはこのエリア内でのみシュートができるよう, 制限を加えている. このため, アタッカーはシュートエリア内までパスをつながなければ得点できない.

問題の初期配置の時点では, ボールは必ずアタッカー 1 が蹴る事ができる位置に配置される. また三人のエージェントは, 初期配置の時点では近距離に配置される事はない.

アタッカー 1 がボールを蹴った時点をもって, 問題の一試行が開始したと見なされる. アタッカーが得点をあげた場合, その試行は「成功」したとみなされる. 逆に, ディフェンダーがパスをインターセプトした時点で, その試行は「失敗」したと見なされる.

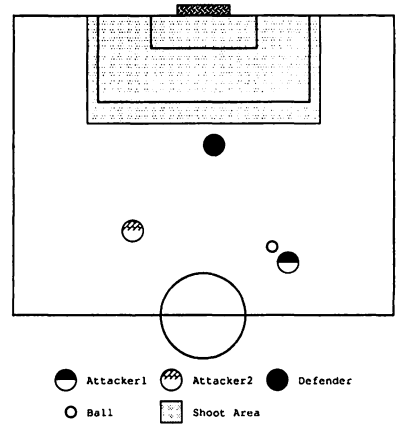

Figure 3: 問題設定

一試行の中でのアタッカー 1 の行動の流れは以下の通りである.

1. アタッカー 2 が受ける事ができ, かつディフェンダーにインターセプトされないような場所へパスをする (行動 1).
2. シュートエリア内へ向けて移動する (行動 2).
3. シュートエリア内へ移動した後, 可能であれば, シュートする.

一方, アタッカー 2 の行動の流れは以下の通りになる.

1. アタッカー 1 からパスが来そうな場所へ移動する (行動 1)
2. アタッカー 1 からのパスを受ける事ができたなら, シュートエリア内の, アタッカー 1 が走り込みそうな場所へパスをする (行動 2). ただし, 最初にパスを

受けた地点がシュートエリア内であれば，シュートする．

この状況下においてアタッカーがパスを繋いでゴールを決めるには，両アタッカーのパスを出した場所と，パスを受けるために走り込む場所が同調していなければならない．

また，その場所が同調していたとしても，場合によっては，パスが届かなかったり，走り込みとのタイミングが合わなかったり，アタッカーよりも先にディフェンダーがボールに到達したりして失敗に終わる事もある．

以上を考えると，協調パスプレイを実現する為には，

- パス出し，走り込みの場所の指標を備えたフィールド認識能力
- 把握したフィールド状況より，最適なパス出し，走り込みの場所を選択する能力

がエージェントには必要とされる．

4 サッカーエージェントの基本行動戦略

本章では，協調パスプレイを実現する為の，サッカーエージェントの基本行動戦略について述べる．

本研究におけるサッカーエージェントは，Pre-RoboCup'96 の優勝チームである「Ogalets」[5] をベースとして設計されている．Ogalets では，各エージェントには固有のホームポジションとゾーンが設定されている．エージェントはまず，ボールが見えているかどうかの判断を行う．見失っている場合は，ボールを探す．ボールが見えており，かつ自らのゾーンにある場合は，十分に近い位置にあるならばキックをする．離れているならば，ボールの動きに合わせてボールに接近する．ボールが自らのゾーン外にある場合は，ボールの位置を確認する．以上が Ogalets の行動戦略である．

本研究で用いるエージェントは，フィールドを m × n 個の四角形のゾーンに分割して認識し，フィールドの状況に応じて，パサーはパスを出すゾーンを，レシーバーはパスを受けるゾーンを選択する．すなわち，本手法のエージェントの行動選択は，ゾーンの選択と置き換えられる．

この「ゾーン」が，パス出し，走り込みの同調のための指標となる．両エージェントの選択したゾーンが一致している事が，パスが繋がる前提条件である．また，ゾーン内での行動は Ogalets の行動戦略に準拠する．

エージェントの行動決定の流れは Figure 5のようになる．

本エージェントの設計において最も重要視すべきは，「どこへ移動するか」，「どこへパスするか」という意思決定を如何にして行なうかという点である．両エージェントの選択したゾーンが一致していても，例えばパスが

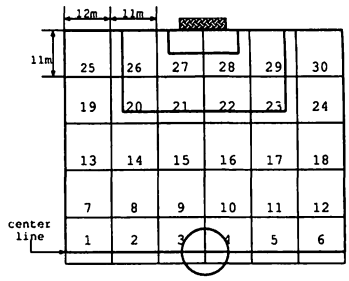

Figure 4: 分割例 (5×6)

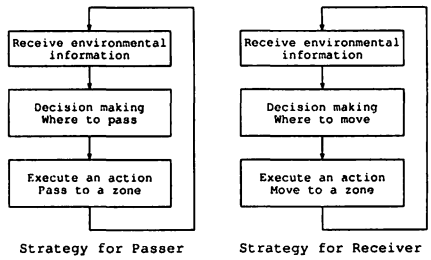

Figure 5: エージェントの行動決定の流れ

届かなかったり，走り込みとのタイミングが合わなかったり，アタッカーよりも先にディフェンダーがボールに到達したりして失敗に終わる事もある．

このような動的に環境が変化していく状況に置いては，全状況についてそれに対応した行動を予めプログラムしておくことは不可能に近い．よって，本研究では，この状況に適応した意思決定をエージェントに自律的に獲得させる為に，複合学習システム [Smith, 1997][10] を意思決定関数としてエージェントに組み込む．

5 複合学習システム

複合学習システムは Robert E. Smith によって提案されたシステム [10] であり，ニューラルネットワーク，クラシファイアーシステム，Q-learning を統合したシステムである．本システムは 3 層構造 (入力層，隠れ層，出力層) のニューラルネットワークをベースとしている．各々の隠れ層のノードはクラシファイアーで表現される．ネットワークは，環境状態を Q 値にマッピングするのに用いられる．

5.1 システムの概要

本システムのネットワークの隠れ層の各ノードは，クラシファイアーによって表現される．クラシファイアーは，各ノードが発火するための閾値としての働きを持つ．つ

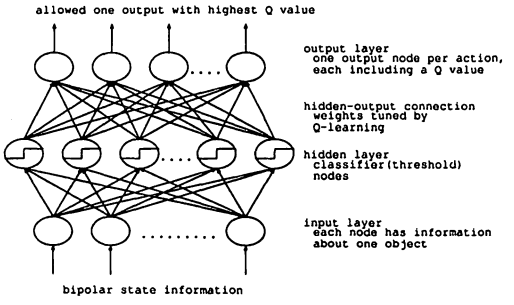

Figure 6: 複合学習システム

まり, 入力層からの入力と, 中間層の各クラシファイアーとのマッチングが成立したノードが, 発火する権利を有することになる.

また, 各クラシファイアーノードは全ての行動に対応する Q 値を持っており, 発火したノードは, これらの Q 値の全てを出力層に向けて出力することができる.

出力層では, 各ノードは取り得る行動と関連付けられる. つまり, 出力層ノードの総数は, 取り得る行動の総数に等しい. 全ての出力層ノードでは, 発火した隠れ層ノードから送られた Q 値が合計される. そして, 最も高い合計値を持つノードが出力することを許され, その出力したノードに対応した行動を行う事となる.

出力によって選択された行動が行なわれた後に, 出力に関係したクラシファイアーノードの Q 値は, Q-learning の手法によって更新される. この Q 値そのものが, 隠れ層と出力層との結合の重みを表す. 状態 i において行動 u をとった結果, 状態 jに遷移したとすると一般の Q-learning 同様に式 (1) に則って Q 値を更新する.

$$Q_{t+1}(i,u_t)=Q_t(i,u_t)(1-\alpha) \\ +\alpha[r_i(u_t)+\gamma \max Q_t(j,u_{t+1})] \quad (1)$$

$0 \leq \alpha \leq 1$は学習係数, $0 \leq \gamma \leq 1$は割引率, $Q_{t+1}(i,u_t)$ は状態 iにおける行動 u に割り当てられた Q 値, $r_i(u_t)$ は環境報酬, $\max Q_t(j,u_{t+1})$ は状態 jにおける最大の Q 値である.

Q 値の更新を一定期間行なった後, 遺伝的操作がクラシファイアーノードに対して行なわれる.

6 サッカーエージェントへの適用

複合学習システムをエージェントに適用するに当たって決定すべき事は, 環境情報の内の何を入力情報とするか, また, 出力に対応する行動をどのようなものにするか, そして, 選択した行動に対する評価戦略の三点である. 本章では, これらの入力情報, 出力, 評価戦略の定義について述べる.

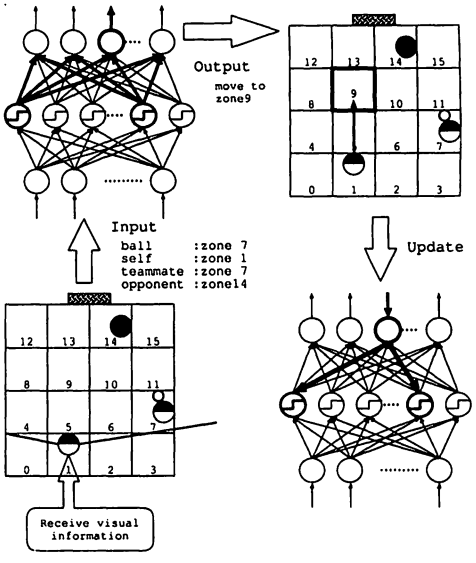

Figure 7: システムの作動例

6.1 入力と出力

入力情報 入力情報はエージェントの視覚を通じて得られるフィールド状況である. サッカーサーバにおいては, オブジェクト (エージェント自身, 他のエージェント, ボール, ゴール等) のフィールド上の位置が視覚情報としてエージェントに渡される. 本手法ではフィールド状況を「どのオブジェクトがどのゾーンに存在しているか」と定義する. このような定義を行うことによって, エージェントが他オブジェクトやスペース (どのオブジェクトも存在していないゾーン) との位置関係を判断基準として用いることが期待できる. よってシステムへの入力は, 「各オブジェクトが存在しているゾーン番号」の集合になる. また, 隠れ層のクラシファイアーとのマッチング作業のため, 情報は 2 進数のビット列にコーディングされる.

Figure 6を例に取ると, フィールドを 4×4 の 16 のゾーンに分割しているため, 一物体についての情報を表現するには 4 ビットが必要となる. 自身はゾーン 1, 味方がゾーン 7, ボールがゾーン 7, 敵がゾーン 14 に存在するため, これらを 2 進数表現した「0001」「0111」「0111」「1110」が入力となり, この集合であるビット列「0001011101111110」が中間層ノードのマッチング対象となる.

出力 エージェントはこのシステムを通して「入力に応じた各ゾーンの評価値」を決定し, 最も評価の高い領域を選択し, そのゾーンに対して行動を行う. ボールを保持していれば, そのゾーンへ向けてパスを出

す. 保持していなければ, そのゾーンへ移動する. この選択されたゾーンが, システムの出力となる.

6.2 評価戦略の決定

このシステムにおいて最も重要な点は, 選択された行動に対する評価戦略である. 本研究で例題として取り上げている 2 対 1 の状況でも,

- アタッカー 1(A1) からアタッカー 2(A2) へのパスがインターセプトされる
- A1 から A2 へのパスは成功したが, A2 から A1 へのパスがインターセプトされる
- A1 から A2 へのパス, A2 から A1 へのパスのいずれもが成功し, A1 がシュートする
- A1 から A2 へのパスが成功したのち, A2 がシュートする

といった場合が考えられる. これに対応するように, Q 値の更新式は以下のように設定した. R_X1 は, アタッカー X の行動 1 に対する環境報酬, R_X2 は, アタッカー X の行動 2 に対する環境報酬, Q_1, Q_2はそれぞれ各アタッカーの行動 1, 行動 2 に対応した Q 値である.

パスが成功した場合, すなわち, パサーがパスを出したゾーンと, レシーバーがパスをもらったゾーンが一致した場合は, そのゾーンの評価値を増加させ, 逆にパスが失敗した場合 (ゾーンが一致しなかった場合) は, そのゾーンの評価値を減少させるというのが基本的な評価戦略である.

各行動に対する環境報酬は, アタッカーが行動した後に, 試行の成功もしくは失敗が成立した瞬間に渡される. 試行の成功は得点のみをもって成立するが, 失敗には二種類の場合がある. すなわち, A1 から A2 へのパスが失敗した場合と, A2 から A1 へのパスが失敗した場合である. 本問題における環境報酬を以下に示す.

A1 から A2 へのパス失敗 A1, A2 双方の行動 1 が同調せず, 行動 2 は実行されていない. よって失敗の要因は双方の行動 1 にあるため, 行動 1 の Q 値が減少するような環境報酬を与える.

$$R_1 1 = Q1 - Q1/10 \cdot (300/e\_time) \quad (2)$$

$$R_2 1 = Q1 - Q1/10 \cdot (m\_dist/87) + 87/g\_dist \quad (3)$$

$$R_1 2 = R_2 2 = Q2 \quad (4)$$

A2 から A1 へのパス失敗 A1 から A2 へのパスは成功している. つまり, A1, A2 双方の行動 1 は同調しているが, 行動 2 が同調していない. 失敗の要因は双方の行動 2 にある. 行動 1 の Q 値は増加, 行動 2 の Q 値は減少するような環境報酬を与える.

$$R_1 1 = Q1 + INIT\_Q \cdot (e\_time/300) \quad (5)$$

$$R_2 1 = Q1 + INIT\_Q \cdot (e\_time/300) \quad (6)$$

$$R_1 2 = Q2 - INIT\_Q/10 \cdot (300/e\_time) \quad (7)$$

$$R_2 2 = Q2 - INIT\_Q/10 \cdot (300/e\_time) \quad (8)$$

A1 の得点 A1, A2 双方の行動 1, 行動 2 のいずれもが同調した結果である. よって, 全ての行動の Q 値が増加するような環境報酬を与える.

$$R_1 1 = R_2 1 = Q1 + INIT\_Q \cdot (300/e\_time) \quad (9)$$

$$R_1 2 = R_2 2 = Q2 + INIT\_Q \cdot (300/e\_time) \quad (10)$$

A2 の得点 A1, A2 双方の行動 1 が同調しており, A2 はパスをもらうとそのままシュートする. この場合, 評価の対象となるのは行動 1 のみであり, 成功している為, その Q 値は増加する.

$$R_1 1 = R_2 1 = Q1 + INIT\_Q \cdot (300/e\_time) \quad (11)$$

$$R_1 2 = R_2 2 = Q2 \quad (12)$$

e_time は一試行にかかった時間 (シミュレーションステップ), m_dist はエージェントの移動距離, g_dist はエージェントとゴールとの距離を表している.

パス失敗時における試行時間の短かさは, パスがすぐインターセプトされたことを示す. 関連している行動は悪い行動といえるため, 式 (2), (7), (8) では試行時間が短ければ短い程 Q 値が減少するよう設定した. 逆にパス成功時においては, 試行時間の短さはパスが早くつながったことを示す. 関連した行動は良い行動といえるため, 式で (5), (6), (9), (10), (11), は試行時間が短い程 Q 値が増加するように設定した. 300 は, 一試行に費される最大時間である.

最初のパスがつながるかどうかは A2 の移動に負う部分が大きいと考えたため, 式 (3) はエージェントの移動距離によって報酬を調節する設定にした. 移動距離が長すぎるとパスのつながる可能性が低くなると考えられる. よって移動距離が長い程 Q 値は減少するよう設定した. ただし, ゴールに近付くような動きには良い評価が与えられるようにした. (3) で用いられる 87 とは, 本問題におけるエージェントの最大移動距離 (敵陣を四角形とした時の対角線の長さ) である.

7 実験

3章で述べた問題設定にのっとり, 2 対 1 の状況下で, 4 章で述べた行動戦略と, 複合学習システムを備えたエージェントが, 協調パスプレイを生成できるかどうかを実験した.

7.1 実験設定

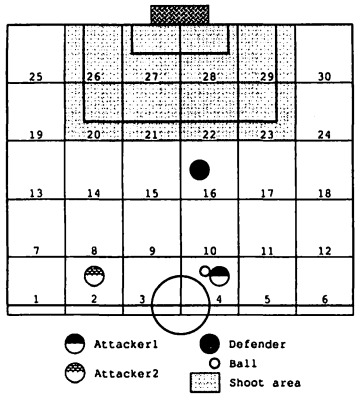

Figure 8: 初期配置

実験の初期配置を Figure 8に示す. 敵陣を 6×5 の 30 のゾーンに分割している. アタッカー 1 とボールはゾーン 4, アタッカー 2 はゾーン 2, ディフェンダーはゾーン 16 に配置される.

各種のパラメータは以下の通りである.

Q 値の初期値 ($INIT\_Q$)	100
学習係数 (α)	0.3
割引率 (γ)	0
交叉率	0.6
突然変異率	0.15

「アタッカーが 2 ゴールをあげるのに費した時間」を, 学習システムのクラシファイアーノードの Q 値を更新する期間と定義した. この期間が終了する時に, アタッカーは遺伝的操作をクラシファイアーノードに対して実行する. この, Q 値が更新され, 遺伝的操作が行われるまで試行が繰り返される期間を「1 世代」とみなす.

各アタッカーの行動 1 と行動 2 が同調しているのならば, より短い時間で 2 ゴールを奪えるはずである. この時間が, アタッカーが効率良い行動を生成できているかどうかの指標となる.

7.2 結果と考察

先に述べた設定の実験を, 学習システムの中間層ノードが 200 世代に達するまでを 1 回の実験の期間として 5 回行なった. Figure 9は各世代における 2 ゴールをあげるまでの時間の推移状況の, 5 回の実験の平均を表す. 初期の段階では, かかった時間は非常に大きい. だが, 世代の進行とともに時間は劇的に減少していき, 120 世代近辺からは, 時間は 3000〜5000 程度に収束している. Figure 10は各世代における試行数の推移状況の, 5 回の実験の平均を表す. これも Figure 9と同様の変化を見せており, 初

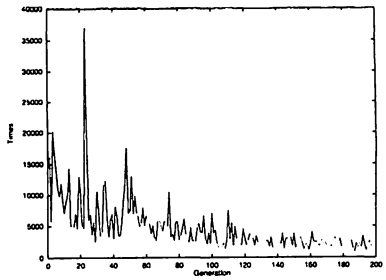

Figure 9: 時間の推移

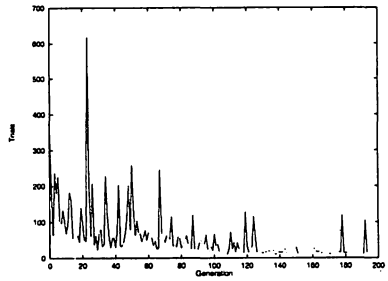

Figure 10: 試行数の推移

期段階の膨大な試行数が世代進行とともに減少し, 120 世代近辺からは 50 試行程度に収束している.

Figure 11, 12, 13は, 5 回の実験の中で得られた, 協調パスプレイの一つの例である.

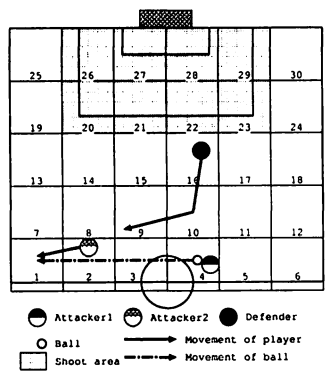

Figure 11: 生成されたパスプレイ (段階 1)

まず最初に, アタッカー 1 はゾーン 1 へパスをする. それと同時に, アタッカー 2 はゾーン 1 へ移動をする (Figure 11).

次に, パスをしたアタッカー 1 はゾーン 22 へ移動する. ゾーン 1 でパスを受けたアタッカー 2 は, ゾーン 22 へパスをする (Figure 12).

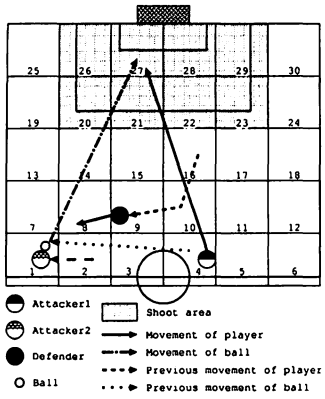

Figure 12: 生成されたパスプレイ (段階 2)

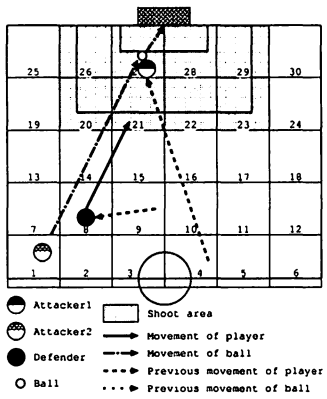

Figure 13: 生成されたパスプレイ (段階 3)

最後に, ゾーン 22 でパスを受けたアタッカー 1 がシュートをする (Figure 13).

これは, 実際のサッカーにおいて, 「壁パス」ないしは「ワンツーパス」と呼ばれるプレイと同じプレイでであり, 2 対 1 の状況で選択されるプレイとしては, 妥当なものである.

8 今後の展開

本研究では 2 対 1 という単純な状況の下で, 攻撃側のエージェントが学習によって協調パスプレイを生成することが確認できた.

同じ 2 対 1 の状況でも, 各エージェントの初期配置が変わって行くような状況下において, パスプレイを生成し, またアジャストできるかどうかを検証する事が, 今後の課題となる.

9 おわりに

本研究では SoccerServer 上において「スルーパス」, 「壁パス」に代表されるような, パサー, レシーバー両者の同調によって生成される協調パスプレイをサッカーシミュレーションエージェントに実現させるための手法について示した.

また, 本手法では, 複合学習システムをエージェントに組み込み, フィールド状況に応じたパスプレイを成立させるために取るべき行動を, 自律的に獲得させる事を試みた.

協調パスプレイの生成を検証するために例題として設定した 2 対 1 という単純な状況の下で, 攻撃側のエージェントが学習によって協調パスプレイを生成することが確認できた.

参考文献

[1] 北野 宏明, 大沢 英一, 松原 仁 : なぜ今, RoboCup なのか?, bit, vol.28, No.5, pp.22-27(May 1996)

[2] 野田 五十樹, 國吉 康夫 : シミュレーション部門と Soccer Server, bit, vol.28, No.5, pp.28-25(May 1996)

[3] 野田 五十樹, 松原 仁 : サッカーエージェントの研究, 人工知能学会誌, vol.11, No.5, pp.18-25(September 1996)

[4] RoboCup webpage, http://www.robocup.org/02.html

[5] Koichi Ogawara's Program "Ogalets", http://ci.etl.go.jp/ noda/soccer/client/ index.html#PreRoboCup96

[6] Tomohito Andou : Refinement of Soccer Agents' Positions Using Reinforcement Learning, Proceedings of RoboCup-97, pp.373-388 (1998)

[7] Peter Stone and Manuela Veloso : Using Decision Tree Confidence Factors for Multiagent Control, The Proceedings of The First International Workshop on RoboCup, pp.31-36(1997)

[8] Peter Stone and Manuela Veloso : A Layered Approach to Learning Client Behaviors in the RoboCup Soccer Server, Applied Artificial Intelligence, 12 (1998)

[9] Sean Luke : Genetic Programming Produced Competitive Soccer Softbot Teams for RoboCup97, Genetic Programming, pp.214-222(1998)

[10] Robert E. Smith, H.B. CribbsIII : Combined biological paradigms: A neural, genetic-based autonomous system strategy, Robotics and Autonomous Systems, Vol.22, No.1, pp. 65-74(1997)

[11] 田中 久美子, Ian Frank, 野田 五十樹, 松原 仁 : RoboCup シミュレーションリーグの統計的分析, 人口知能学会誌, Vol.14, No.2, pp.200-207(March 1999)

[12] 北野 宏明 : 遺伝的アルゴリズム, 産業図書

[13] 坂和 正敏, 田中 雅博 : 遺伝的アルゴリズム, 朝倉書店

[14] Marco Doringo and Marco Colombetti : ROBOT SHAPING : A Bradford Book The MIT Press, pp.22-33