修士論文　　平成 27 年度 (2015)

サッカーゲームにおける
大局的戦術指示による即応機械学習に関する研究

東京工科大学大学院

バイオ・情報メディア研究科

メディアサイエンス専攻

菅原 俊生

修士論文　　平成 27 年度 (2015)

サッカーゲームにおける
大局的戦術指示による即応機械学習に関する研究

指導教員　渡辺 大地

東京工科大学大学院

バイオ・情報メディア研究科

メディアサイエンス専攻

菅原 俊生

論 文 の 要 旨

論文題目	サッカーゲームにおける 大局的戦術指示による即応機械学習に関する研究
執筆者氏名	菅原 俊生
指導教員	渡辺 大地
キーワード	ゲーム AI、サッカーゲーム、機械学習、人工知能 、自動指示

[**要旨**]

賛テレビゲームの人気ジャンルの一つに，サッカーや野球などの実在するスポーツと同じルールで対戦するスポーツゲームがある．

サッカーのように複数の人間が同時に行動するようなスポーツをゲームとして再現する場合，11 人いる味方を同時に動かすのは難しいため基本的にチームの中の一人の選手しか自由に操作することはできない．そこで一般的なサッカーゲームでは，プレイヤーが操作していない味方の選手は AI により自動で操作を行うのが主流である．一般的なサッカーゲームでは，11 人いる味方のうち 10 人は AI により自動で動くため味方の AI がどのような動きをするかは非常に重要である．しかし，戦略やフォーメーションが多様に存在するサッカーのようなスポーツでは，AI が常にプレイヤーの想定した通りの動きをし続けることは難しい． そこで本研究では，味方 AI をプレイヤーの意図に合わせた行動を取らせる方法として，プレイヤーが味方 AI に対し大局的な指示を出し，その指示情報を学習しプレイヤーの意図する通りに AI を動かす手法を提案する．

まずプレイヤーが味方選手に対して大局的な指示を送ることが可能なサッカーゲームを開発した．このゲームではプレイヤーが味方 AI に対し指示を行う際，どのようなタイミングで指示が出されたのかを随時記録していく．次に記録した情報の中からどのようなタイミングでどのような指示が出されやすいのか，その傾向を見つけ出す．そしてゲームを進行中に，現在の状況がプレイヤーが指示を送りやすいタイミングであると AI が判断すると，その時点で自動で指示行動を行う．これによりプレイヤーが指示を出し続けることで，次第に指示を出さなくても AI がプレイヤーの意図した通りに動くようになる．

実験では 15 人を対象にユーザーテストを行い，実際に作ったサッカーゲームを遊んでもらった．そして手動指示と自動指示がどれだけ一致しているかを調べた．

実験の結果，人によっては手動の指示とほぼ同じ内容の自動指示を出すことができたパターンもあった．しかし，まだ自動指示の精度にまだ問題があることが分かった．

A b s t r a c t

Title	Research of readiness machine learning by global tactical instruction in soccer game
Author	Toshiki Sugawara
Advisor	Taichi Watanabe
Key Words	Game AI, Soccer game, Machine learning, AI, Automatic indication

[**summary**]

In one of the most popular genres of video games, there is a sports game to play with the same rules as in the sport a real, such as soccer and baseball.

If you want to reproduce the sport, such as to act more than one person at the same time as the football as a game, and move at the same time the allies have 11 people flame Kashii for Can not basically only one of the players in the team free to operate that. Therefore, in a typical football game, the ally of the player the player does not operate is the mainstream to carry out the operation automatically by the AI. In a typical football game, 10 out of the allies are 11 people or it is very important to what motion AI of allies order to move automatically by the AI. However, in sports such as football strategies and formations exist diverse, it is difficult to continue the movement of the street AI has always assumed the player. The reason for AI is a movement that is not expected, the ideal movement sought to players during the game, be mentioned that there are various by player. Some games that can send some instructions to AI, but carrying out instructions simultaneously and accurately with respect to the operation and AI character the player itself moves is difficult.

Therefore, in this study, as a way assume a combined action of the ally AI to the intention of the player, the player is issued a global instructions to the ally AI, We propose a method to move the AI as it is intended for players to learn the instruction information.

First player has developed a big-picture that can send instructions soccer game against teammate player. This is a game when a player issues an instruction to ally AI, what timing to go to record whether an instruction is issued from time to time. What what indication is easy to be issued at the timing out of the next recorded information, to find the trends. And during the course of the game, when the AI determines the player the current situation is a feed easy timing instructions, it instructs action automatically at that point. Thus the fact that players keep instructs, the AI without gradually out an instruction will move as intended by the player.

目 次

第 1 章　はじめに 1

第 2 章　実験に使用するサッカーゲームについて 7
2.1　ゲームの概要 8
2.2　選手の行動パターン 10
2.3　危険度 13
2.4　守備範囲の指定 20
2.5　プレイヤーが可能な操作 21

第 3 章　提案手法 23
3.1　選手のステータスについて 24
3.2　自動指示の流れ 27
3.3　指示実行ステータスの作成 28
3.4　自動指示の実行 32

第 4 章　実験 35
4.1　評価実験の内容 36
4.2　検証の方法 36
4.3　一致率の計算法 37
4.4　実験結果 38

第 5 章　まとめと今後の展望 42

謝辞 44

参考文献 46

図 目 次

1.1 コナミ ウィニングイレブン 2016 3
1.2 RoboCup サッカーシミュレーション 4

2.1 実験用のゲーム 9
2.2 AI の動きを分かりやすく表示した画面 9
2.3 指示可能な味方選手 10
2.4 敵の選手 10
2.5 サッカーボール 10
2.6 選手の状態表示画面 12
2.7 行動パターンのフローチャート 13
2.8 相手選手の周りに展開される危険度 15
2.9 危険度の加算 15
2.10 危険度のイメージ 16
2.11 理想的と思われる危険度の設定 17
2.12 ボールが右にある場合の危険度 18
2.13 ボールが下にある場合の危険度 18
2.14 ボールが右下にある場合の危険度 19
2.15 ドリブルのイメージ 20
2.16 パスのイメージ 20
2.17 選手の守備範囲 21

3.1 ステータスのイメージ 25
3.2 ステータス保存のイメージ 26
3.3 グループ化の流れを示したフローチャート 28

4.1 検証のイメージ 37
4.2 方法 1:上がれの一致率 40

4.3　方法 1:下がれの一致率 40
4.4　方法 2:上がれの一致率 41
4.5　方法 2:下がれの一致率 41

表 目 次

3.1 ステータスを表にしたもの 26
3.2 グループ内における保管ステータスの各項目の確率 31
3.3 指示『上がれ』における指示実行ステータス 32

4.1 実験結果 39

第 1 章

はじめに

テレビゲームの人気ジャンルの一つに，サッカーや野球などの実在するスポーツと同じルールで対戦するスポーツゲームがある．

サッカーのように複数の人間が同時に行動するようなスポーツをゲームとして再現する場合，11 人いる味方を同時に動かすのは難かしいため基本的にチームの中の一人の選手しか自由に操作することはできない．そこで一般的なサッカーゲームでは，プレイヤーが操作していない味方の選手は AI により自動で操作を行うのが主流である．一般的なサッカーゲームの例としてウィニングイレブンシリーズなどがある．図 1.1 はウィニングイレブン 2016[1] のゲームプレイ中の画面である．ウィニングイレブンのような一般的なサッカーゲームでは，11 人いる味方のうち 10 人は AI により自動で動くため味方の AI がどのような動きをするかは非常に重要である．しかし，戦略やフォーメーションが多様に存在するサッカーのようなスポーツでは，AI が常にプレイヤーの想定した通りの動きをし続けることは難しい．AI が想定しない動きをする理由として，ゲーム中の選手に求められる理想的な動きが，プレイヤーによって様々であることが上げられる．AI に対してある程度の指示を送ることのできるゲームもあるが，プレイヤー自身が動かすキャラクターの操作と AI に対する指示を同時にかつ的確に行うことは困難である．



図 1.1　コナミ ウィニングイレブン 2016

http://www.konami.jp/we/2016/index.php5

ゲーム AI を対象とした研究 [2][3][4][5] は多数存在する．その中でもゲーム AI における機械学習の研究 [6][7][8][9][10][11][12] はその多さから注目されていることが分かる．ゲームにおける機械学習の研究として星野ら [13] は格闘ゲームのような対戦アクションゲームにおいて，プレイヤーの操作情報から AI により動く対戦相手の行動パターンの拡張を行った．

また，サッカーゲームのゲーム AI に関しては特に注目されており，研究の題材によく扱われるものとして RoboCup サッカー [14] がある．RoboCup サッカーとは，人間同士のサッカーの試合と同じように自分で考えて動く自律移動型ロボットを使った競技会である．RoboCup サッカーには，ロボット同士を対戦させるロボットリーグと，実機を使うことなくコンピュータ上の仮想フィールドで異なった人工知能プログラミング同士を対戦させるシミュレーションリーグが存在する．ロボットリーグには小型，中型，人型などのロボットの機能や大きさに応じた分野に分かれており，シミュレーションリーグでは左右前後移動のみの 2D リーグとボールの上下移動

を考慮した 3D リーグが存在する．

シミュレーションリーグを対象とした研究は多数存在するが比較的 2D シミュレーションリーグを対象とした研究 [15][16][17][18][19][20] が多い．3D シミュレーションリーグを対象とした研究 [21][22][23] には内種ら [24] の研究のようにボールやロボットの動力を考慮したものもある．そしてシミュレーションリーグを対象とした研究成果は基本的に他のサッカーゲームにも流用することができると考えられる．図 1.2 はロボカップサッカーシミュレーションリーグのシミュレータの画面 [25] である．

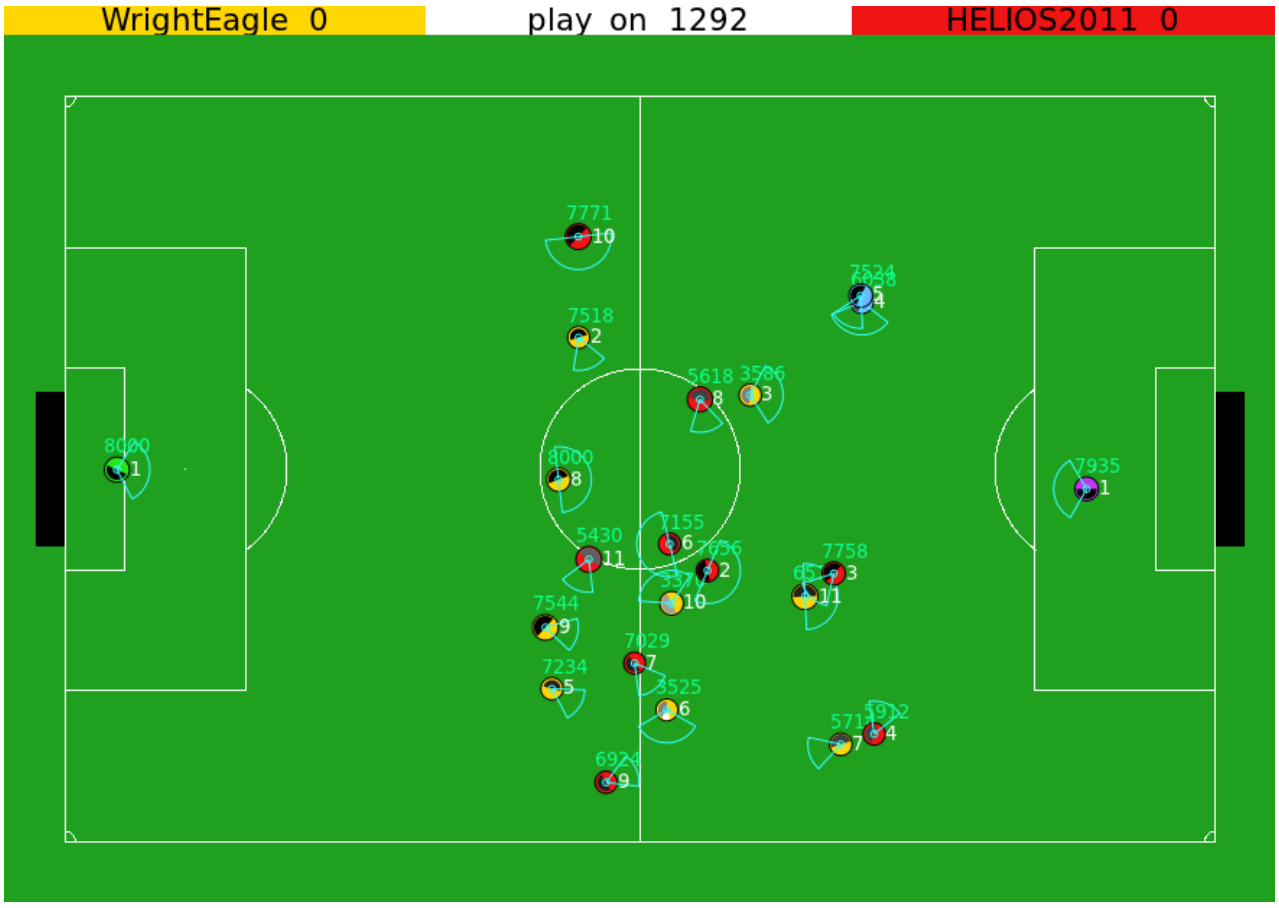

図 1.2　RoboCup サッカーシミュレーション

http://rc-oz.osdn.jp/pukiwiki/index.php?SimJP

RoboCup サッカーシミュレーションにおける機械学習の研究としては，金重ら [26] による

RoboCup サッカーシミュレーションにおいて機会学習の有用性を証明したものなどがある．このように RoboCup サッカーシミュレーションを題材にした研究は数多く存在するが，RoboCup サッカーシミュレーションは自律型のエージェント同士を対戦させることを目的としており，一般的なサッカーゲームのようにプレイヤーが操作することを想定していない．そのためプレイヤーの操作を想定したサッカーゲームにおける味方 AI の研究というものがほとんど存在しない．

また，RoboCup サッカー以外のサッカーゲームを対象とした研究として，井上 [27] はサッカーゲームにおける強化学習において報酬値の設定に指標がないという問題に対して，遺伝的アルゴリズムによる報酬値の探索を提案した．しかし，この研究は数百回の試行を重ねた上で成立する研究であり，プレイヤーが操作を通して学習していくということを想定していない．

また，プレイヤーが操作可能なサッカーゲームとしてはプレイヤーの想定通りに味方選手を動かす研究として，川上ら [28] はベイジアンネットワークを用いてプレイヤーの考えを推測し，隠れマルコフモデルによって各選手の行動を決定する手法を提案している．この研究ではプレイヤーの操作情報からプレイヤーの行動の予測を試みているが，予測部分では高い評価は得られていない．その理由として考えられるのが，アクションゲームのようにキャラクターを精密な操作で常に動かし続けるようなゲームの場合，プレイヤーの操作情報のみからプレイヤーがなぜその行動を取ったのかという意図を読み取ることは難しいからだと思われる．

そこで本研究では，味方 AI をプレイヤーの意図に合わせた行動を取らせる方法として，プレイヤーが味方 AI に対し大局的な指示を出し，その指示情報を学習しプレイヤーの意図する通りに AI を動かす手法を提案する．ここで言う大局的な指示とは，『上がれ』や『下がれ』と言った AI が取るべき行動を味方全体に対して出すことである．また本研究における機械学習とは，人間が学習をするのと同様のシステムをコンピュータ上で実現することとしている．この研究ではプレイヤーが出す指示からその傾向を学習し，最善と思われる動きをすることがそれにあたる．

研究の流れとして，まずプレイヤーが味方選手に対して大局的な指示を送ることが可能なサッ

カーゲームを開発した．このゲームではプレイヤーが味方 AI に対し指示を行う際，どのようなタイミングで指示が出されたのかを随時記録していく．次に記録した情報の中からどのようなタイミングでどのような指示が出されやすいのか，その傾向を見つけ出す．そしてゲームを進行中に，現在の状況がプレイヤーが指示を送りやすいタイミングであると AI が判断すると，その時点で自動で指示行動を行う．これによりプレイヤーが指示を出し続けることで，次第に指示を出さなくても AI がプレイヤーの意図した通りに動くようになる．

本研究の提案手法により，プレイヤーが指示を行いやすいタイミングを検出し，自動で指示を行うシステムの開発を行った．

第 2 章では開発したゲームの概要について述べる．第 3 章では研究の提案手法について述べる．第 4 章では提案手法に基づく実験について述べる．第 5 章ではまとめと今後の展望について述べる．

第 2 章

実験に使用するサッカーゲームについて

本研究では実験に使用するゲームを自作で開発を行った．既存の RoboCup サッカーのシミュレータを使わない理由として，プレイヤーによる指示機能などを搭載することが難しいという理由が上げられる．

実験で使用するゲームでは，選手に対する指示機能や学習機能を重視している．基本的にはプレイヤーがゲームに関与しなくても，AI 同士が自動でサッカーの試合を行うことができるが，指示を行うことで味方の選手を大局的に動かし，その指示の傾向を学習するものとしている．

2.1 節ではゲームの概要について述べる．2.2 節では選手の行動パターンについて述べる．2.3 節では選手の進行方向などに関わる値である危険度について述べる．2.4 節では守備範囲の指定について述べる．2.5 節ではプレイヤーが可能な操作について述べる．

2.1 ゲームの概要

サッカーゲームの開発には RoboCup サッカーのシミュレーションリーグのシステムを参考にして実装した．サッカーフィールドを真上から見下ろした画面に，選手を配置し，それぞれの選手に独立した AI が備わっている．図 2.1 は今回の研究で作成したサッカーゲームである．また，図 2.2 は AI 部分の動きを分かりやすく表示したものである．

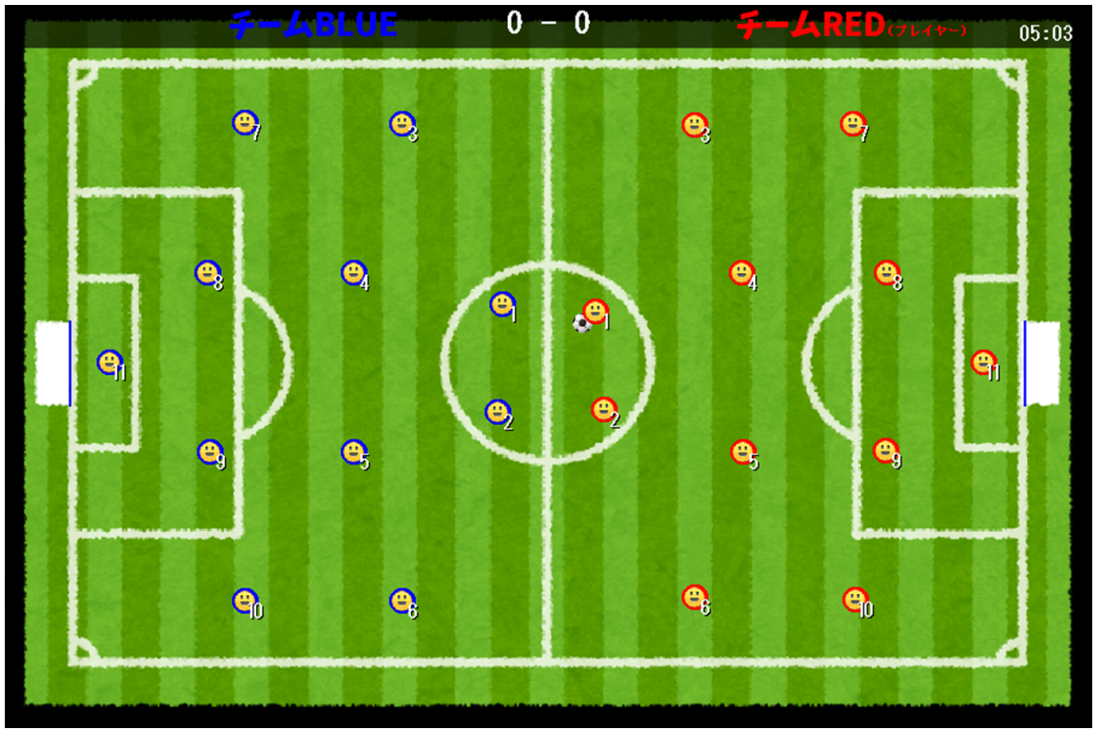

図 2.1　実験用のゲーム



図 2.2　AI の動きを分かりやすく表示した画面

図 2.3 の赤いキャラクターは指示による操作が可能な味方選手である．図 2.4 の青いキャラクターは対戦相手となる選手であり，すべて自動で動く．キャラクターごとに能力の違いはない．図 2.5 はゲーム中に扱われるサッカーボールである．



図 2.3　指示可能な味方選手



図 2.4　敵の選手



図 2.5　サッカーボール

2.2　選手の行動パターン

このゲームにおける選手の AI はそれぞれ独立しており，それぞれの選手がボールの位置や選手の位置などの情報を元に，各選手が一番取るべきだと考えられる行動をとる．選手には 5 つの状態があり，その状態によって行動パターンが変化する．状態は試合中の状況に応じて状態が変化していく．選手が 2 つ以上の状態に同時になることはない．

選手の状態とそれに対応する行動パターンは以下の通りである．

- 『ball chase』状態

近くにボールがあるときはそれを追いかける動作を行う．

- 『ball hold』 状態

  ボールを持っている状態．ドリブルや味方へのパスを行う．

- 『pos support』状態

  味方がボールを所持しているとき，かつボールを持っている味方の一番近くに自分がいるとき，パスを受け取りやすい場所へ移動する．

- 『back pos』 状態

  上の 3 つの状態に当てはまらない，かつ自分が守備範囲の外にいる状態．

  この状態のときは自分の守備範囲に戻る動作を行う．守備範囲とはそれぞれのキャラクターに別々に用意された，選手が居るべき理想的なポジションのことである．図 2.2 における赤線，青線で描かれた円形の部分が守備範囲となる．守備範囲については 2.5 節で詳しく述べる．

- 『wait』状態

  上の状態すべてに当てはまらない場合 wait 状態となり，何も行動は行わない．

図 2.6 は本手法によるサッカーゲームを実行している様子の拡大図であり，選手の上に表示された文字列が選手の状態を示す．



図 2.6　選手の状態表示画面

これらの状態の変化をフローチャートにしたものが図 2.7 である。

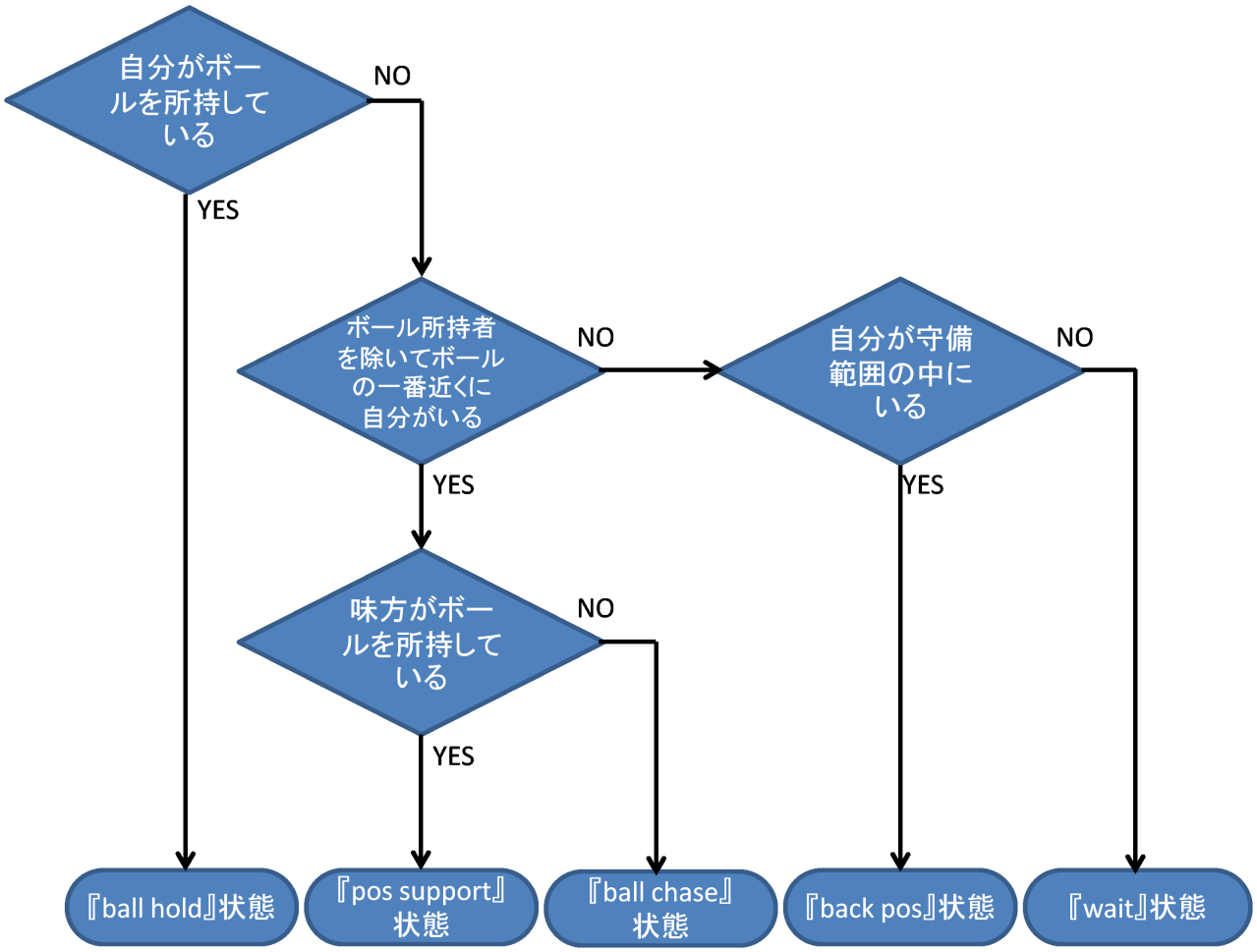

図 2.7　行動パターンのフローチャート

この手法は「実例で学ぶゲーム AI プログラミング」[29] を参考にした.

2.3　危険度

ボールを所持した選手はパスやシュートといったボールを遠くへ飛ばす行動と，ボールを持ちながら移動するドリブルの行動を取ることができる．そして選手がドリブルして移動する際にはできるだけ，相手の選手を避けて移動する必要がある．そのドリブルの進行ルートを決めるために，サッカーフィールド上に設定した危険度を使用した.

危険度とは選手がボール所持している際に，そのエリアがどれだけ相手選手にボールを奪われてしまう可能性が高いかを表した度合いである．危険度は，サッカーフィールドを 16:24 のエリ

アに分割し，そのそれぞれのエリアに設けてある数値のことである．

0 から 100 の数値で表示され，0 に近いほど安全で 100 に近いほど敵にボールを取られる確率が高い場所となる．選手が ball hold 状態の時はこの数値を参考にしながら進行方向を決める．そしてこれ以上進行することが難しいと判断したら，パスの届く範囲の味方の中で危険度の低い位置にいる味方にパスを出す．この危険度は，相手の選手がどこにいるかによって決まり，ゲーム進行中常に変化し続ける．図 2.2 でフィールド上に表示している数値が危険度であり，危険度が高いエリアほど，エリアの色が赤く表示されている．

危険度は基本的にボールを所持していないチームの選手の周辺が高くなるようになっており，基本的には相手選手に近ければ近い程危険度は上がる．図 2.8 は危険度を数値的に現したものである．

また，二人以上の選手が展開する危険度が重なり合う場合，危険度は加算され，図 2.9 のようになる．



図 2.8　相手選手の周りに展開される危険度

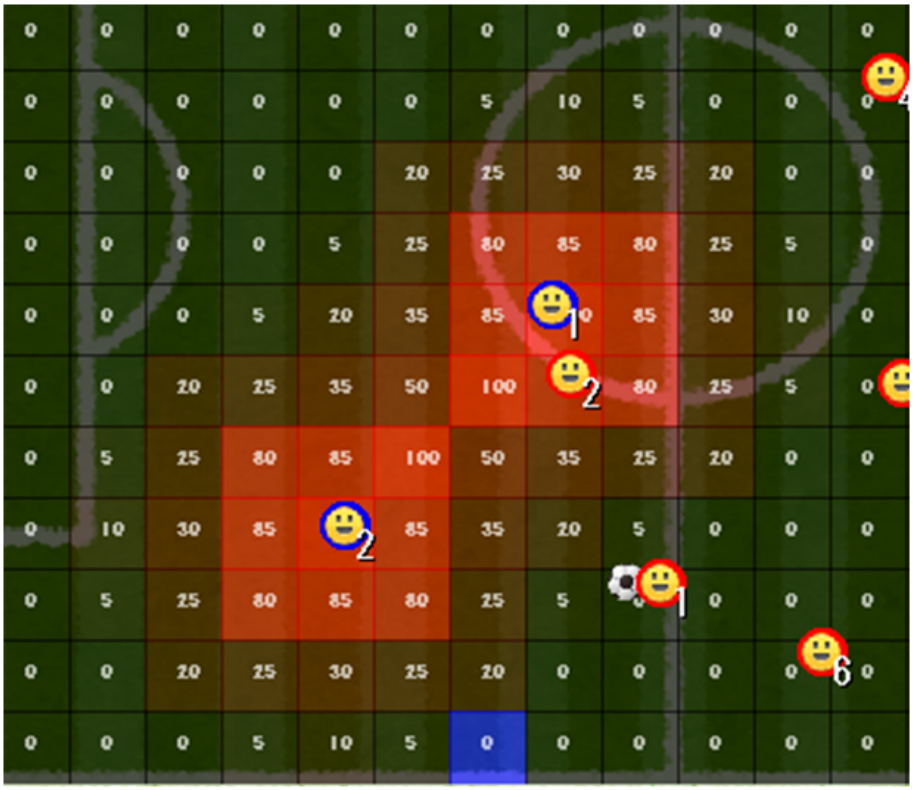

図 2.9　危険度の加算

ただし，図 2.8 のように選手に対し全方位に同じだけ危険度を増やす方法は適切ではない．このゲームでは，高さの概念の無い 2D のサッカーゲームであるため，実際のサッカーのような頭の上を越えるロングパスなどはできない．そのためボールを持っている選手から見て，相手選手の手前にいる味方選手と後ろにいる味方選手では手前にいる味方選手のほうがパスを成功させやすい．しかし，図 2.8 のように危険度を設定すると，相手選手の手前にいる味方選手と後ろにいる味方選手の立ち位置で危険度の差は無く，場合によっては相手選手の後ろにいる味方選手にパスを出してしまう場合がある．図 2.10 は前述した状況をイメージした図であり，味方 A と味方 B のどちらも同じ危険度の場所にいるが，どちらかにパスを出す場合，味方 A の方がパスを成功させやすい．

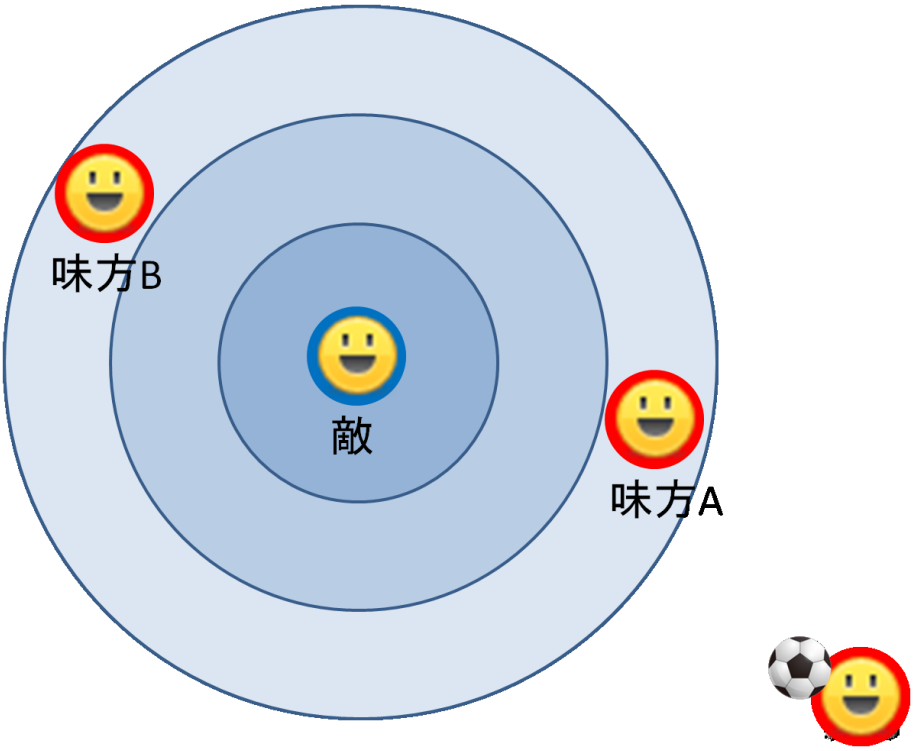

図 2.10　危険度のイメージ

そこで危険度の値をボールの位置に応じて変化するように設定した．基本的に図 2.8 のように相手選手の前後左右均等に危険度が上がり，相手選手に近ければ近い程危険度の値が高くなると

いう形式を基礎としている．そしてボールに近いエリアは数値が減り，反対側のエリアの数値は増えるようにしてある．図 2.11 は理想的と思われる危険度の設定である．図 2.12 は相手選手に対してボールが右側にある場合であり，図 2.13 はボールが下側にある場合，図 2.14 は斜め右下にある場合の危険度を示している．

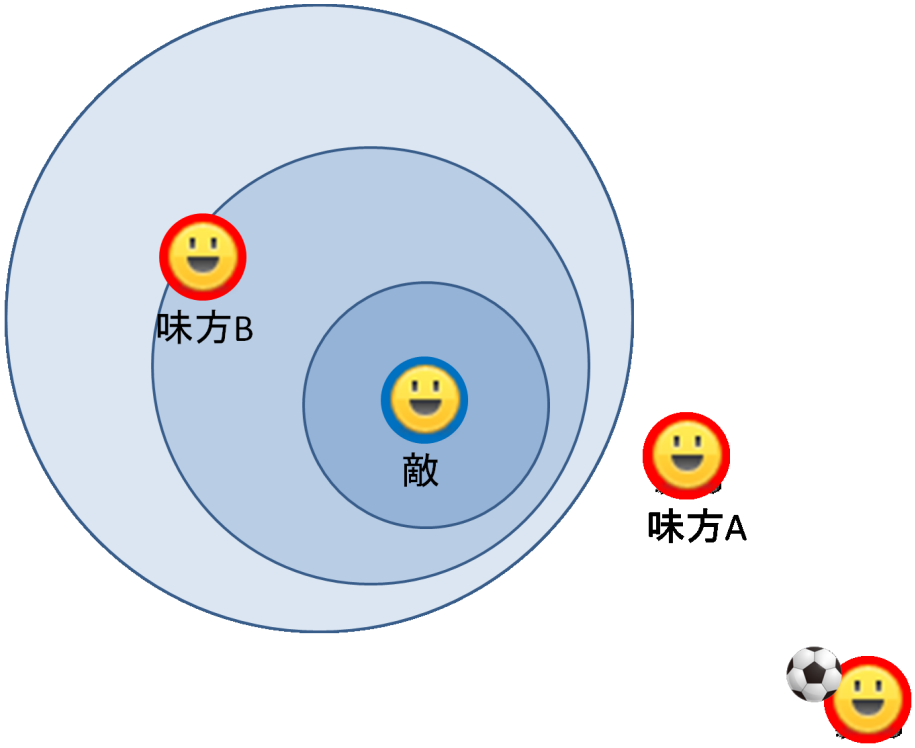

図 2.11　理想的と思われる危険度の設定



図 2.12　ボールが右にある場合の危険度



図 2.13　ボールが下にある場合の危険度



図 2.14　ボールが右下にある場合の危険度

上記の手法で得られた危険度から，各選手 AI にドリブルの進行方向や，パスを行うかどうかを決める．基本的にボールを持っている選手は相手のゴールがある方向へ移動するものとした．ただ単純にゴールへ向かって進むのではなく，ボールを持っている選手は次に進むエリアを常にチェックしながら進行させるものとした．進行方向に対し前方の縦 3 マスの中から最も危険度の低いエリアを探し出し，その方向へと進む．図 2.15 はドリブルのイメージである．そして次に進むエリアのすべてが危険度 50 を超えていた場合，それ以上先に進むのは危険と判断し，図 2.16 のようにパスが届く範囲の味方の中で最も危険度の低いエリアにいる味方へパスをするものとした．

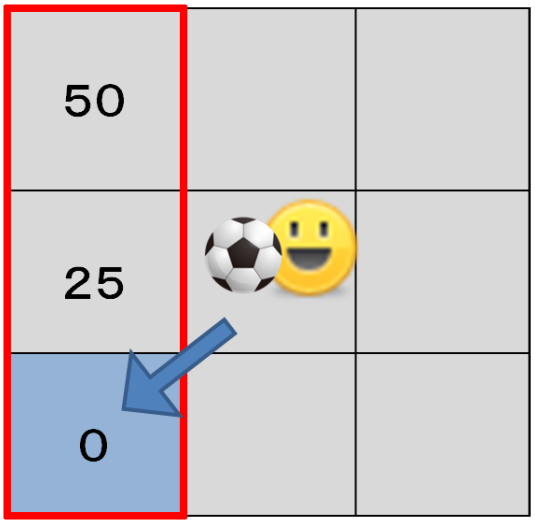

図 2.15　ドリブルのイメージ

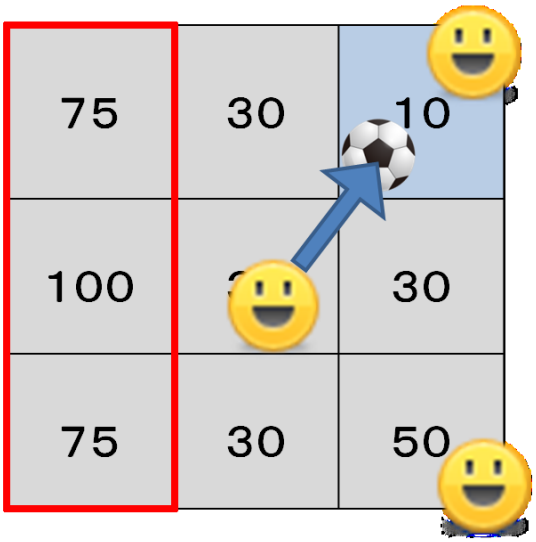

図 2.16　パスのイメージ

この手法は「ゲーム開発者のための AI 入門」[30] を参考にした.

2.4　守備範囲の指定

選手は back pos 状態のときは自分の守備範囲に戻る仕様とした．この守備範囲の場所を変化さすることでボールを持っていない選手をどこに移動するかを指定することができる．図 2.17 の赤と青の円形で示されている箇所が守備範囲となっている．赤い円が味方の，青い円が敵の守備

範囲であり，すべての選手に別々に設けられている．守備範囲の位置はボールの位置と連動して，前後左右に移動する．また，『上がれ』や『下がれ』といったプレイヤーの指示により，守備範囲の位置が変化する．

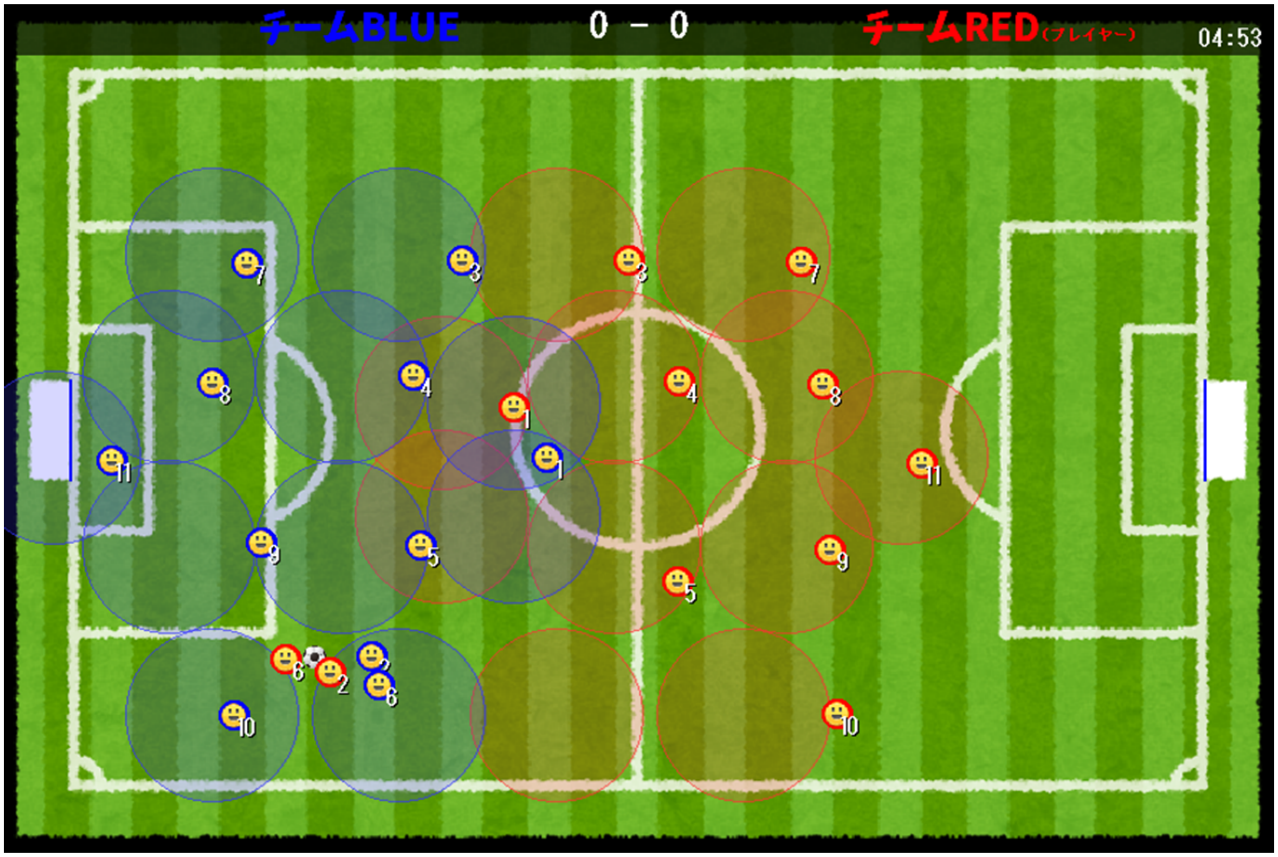

図 2.17　選手の守備範囲

2.5　プレイヤーが可能な操作

この研究で使用するゲームでは基本的にそれぞれの選手が AI により自動で行動を行うため，プレイヤーが何も操作をしなくてもゲームは展開されていく．それに加えてプレイヤーがキー操作により味方選手に指示を出すことにより，味方選手は受け取った指示行動を実行する．

プレイヤーがキー操作により出すことのできる指示には以下のようなものがある．

- 『上がれ』

  味方を前線に出す.

- 『下がれ』

  味方を後ろに下げる.

- 『シュート』

  ボールを味方が所持している時にシュートを行う.

- 『パス』

  ボールを味方が所持している時に近くの味方へパスを行う.

また，一般的なサッカーゲームのように一人の選手を精密に動かすことも可能である．キー操作により移動することが可能で，操作している選手がボールを所持している時は『シュート』『パス』の行動を行うことができる．この操作中に味方選手に対して指示を行うこともできる.

第 3 章

提案手法

本研究では第 2 章にて説明したサッカーゲームを用いて提案手法を実装を行った．

3.1 節では選手のステータスについて述べる．3.2 節では自動指示の流れについて述べる．3.3 節では自動指示を行う際に必要となる，指示実行ステータスについて述べる．3.4 節では自動指示の実行について述べる．

3.1 選手のステータスについて

本研究の提案手法では，プレイヤーが選手に対して指示を送った際に，どのような意図でどのような指示を行ったのかをシステム側が理解する必要がある．そこで本研究で使用するゲームでは、それぞれの選手がどういう状況下にあるのかを認識するメタ AI[31] を作った．

メタ AI とはゲームシステムに宿る AI であり，ゲーム内に身体をもたない AI である．一般的なゲームではメタ AI が追加されることで，ユーザのスキルやゲーム進行を監視しながら，コンテンツを柔軟に生成・変化させていくことが可能となる．

本研究で使うゲーム内のメタ AI が認識できる選手の状態は，すべて YES か NO かで答えられる内容であり，その状態はゲーム中常に更新し続ける．選手が認識できる状態は以下の 10 通りである．

1. チームで一番ボールに近い
2. チームで二番目にボールに近い
3. 自分がボールを所持している
4. ボールを味方が所持している
5. 自分が敵陣にいるかどうか
6. 自分が右サイドにいるかどうか
7. 自分が左サイドにいるかどうか

8. 自分がセンターにいるかどうか

9. 自分が『上がれ』の指示を受けているかどうか

10. 自分が『下がれ』の指示を受けているかどうか

この 10 個の状態は 22 人の選手それぞれを対象に認識することができ，図 3.1 はある特定の選手の状況を分かりやすく表示したものである．



図 3.1　ステータスのイメージ

この研究では，各々の選手がどのような状態にあるかを表したものをステータスと呼ぶ．ステータスは表 3.1 のように表す．選手 ID とはそれぞれの選手を表す番号で，選手 ID が 1 から 11 の選手は指示が可能な味方チーム．選手 ID が 12 から 22 の選手は敵対するチームの選手となる．メタ AI が認識できる状態に対して，YES の場合は Y を，NO の場合は N を当てはめる．選手が 22 人とメタ AI が認識できる状態が 10 個あるため，この項目は全部で 220 項目あることになる．

このゲームではゲーム進行中，選手の状況を示したステータスを常に更新し続ける．このゲー

ム進行中のステータスを現行ステータスと呼ぶ．

表 3.1　ステータスを表にしたもの

	選手 ID								
	1	2	3	4	5	6	7	...	22
1. チームで一番ボールに近い	N	N	N	N	N	Y	N	...	N
2. チームで二番目にボールに近い	N	N	N	N	Y	N	N	...	N
3. 自分がボールを所持している	N	N	N	N	Y	N	N	...	N
4. ボールを味方が所持している	Y	Y	Y	Y	Y	Y	Y	...	N
5. 自分が敵陣にいるかどうか	Y	Y	Y	N	Y	Y	N	...	N
6. 自分が右サイドにいるかどうか	N	N	N	N	N	N	N	...	N
7. 自分が左サイドにいるかどうか	N	N	N	N	Y	Y	N	...	N
8. 自分がセンターにいるかどうか	Y	Y	Y	Y	N	N	Y	...	Y
9. 自分が『上がれ』の指示を受けているかどうか	Y	Y	Y	Y	Y	Y	Y	...	N
10. 自分が『下がれ』の指示を受けているかどうか	N	N	N	N	N	N	N	...	N

そしてプレイヤーが選手に対して指示を行う度に，その瞬間の現行ステータスを保存していく．ここで保存されたステータスを保管ステータスと呼ぶ．保管ステータスはプレイヤーが指示を行った回数分存在し，ステータスとは別にプレイヤーがそのとき何の指示を出したのかという情報も同時に保存される．図 3.2 はステータスを保存していく過程のイメージである．この保管ステータスを参考にし，自動指示を行うタイミングを決定する．

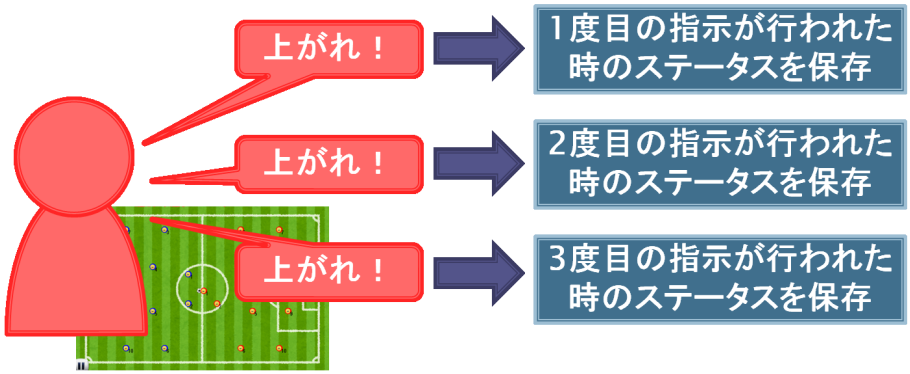

図 3.2　ステータス保存のイメージ

3.2 自動指示の流れ

本研究では現行ステータス，保管ステータスとは別に，指示を行ってほしいタイミングのステータスを用意する必要がある．それを指示実行ステータスと呼ぶ．

ゲーム進行中，現行ステータスと指示実行ステータスの比較を常に行い続け，現行ステータスと指示実行ステータスが類似したときに自動指示を行うという方法を取っている．

指示実行ステータスを作るには，指示時のステータスを保存した保管ステータスの中から，どのようなタイミングでどのような指示が出されやすいのかを読み解く必要がある．本研究では，プレイヤーが指示を行った際に，最後に保存した最新の保管ステータスと，それ以前に保存された『上がれ』『下がれ』といった最新の保管ステータスと同じ指示内容の保管ステータスを対象に比較を行う．そして類似したステータスが 5 つ以上見つかればそれらの保管ステータスを一つのグループとして扱う．グループ内の保管ステータスは全て同じ指示内容のものとなる．

保管ステータスの中に類似したものがいくつも見つかるということは，同じような状況で何度も特定の指示が行われていると考えられる．そしてこのグループ内の保管ステータスの平均値を指示実行ステータスとして扱う．

一つ以上のグループが作成された後は，最新の保管ステータスを生成した際に，最新の保管ステータスとグループの保管ステータスの平均値との比較を行う．そして最新の保管ステータスとグループの保管ステータスの平均値が類似したときは，最新の保管ステータスはその類似したグループに含まれる．そしてすべてのグループと比較し，類似しなかった場合は，グループに含まれていない過去の保管ステータスと比較を行う．

この工程を分かりやすくフローチャートにしたものが図 3.3 である．

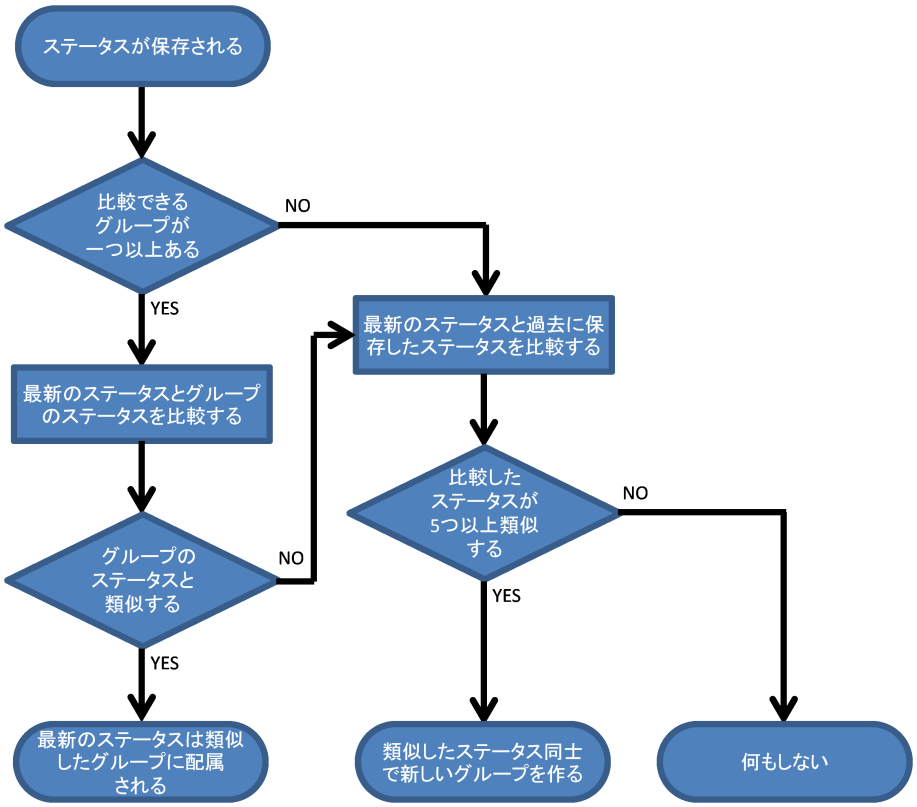

図 3.3　グループ化の流れを示したフローチャート

3.3　指示実行ステータスの作成

現行ステータスを A としたとき A の式は

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \tag{3.1}$$

となる．

i は状態番号である．状態番号とはメタ AI が認識できる状態を，番号で表したものである．j は選手 ID である，n は選手 ID の最大数であり，m は状態番号の最大数を示す．a_{ij} はステータス中の一つの項目を示していおり，a_{ij} の値は，選手 IDi の選手の状態番号 j の項目が YES のときは 1，NO のときは 0 が入る．

次に保管テータスを B_{xy} としたとき数式は

$$B_{xy} = \begin{bmatrix} b_{xy11} & b_{xy12} & \cdots & b_{xy1n} \\ b_{xy21} & b_{xy22} & \cdots & b_{xy2n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{xym1} & b_{xym2} & \cdots & b_{xymn} \end{bmatrix} \tag{3.2}$$

となる．

x は指示番号である．指示番号とは『上がれ』『下がれ』などを区別するための番号であり，ステータスを保存した際に何の指示を出したのかを認識することができる．y は指示番号 x において保存した保管ステータスの番号を示す．b_{xyij} の値は，B_{xy} において選手 IDi の選手の認識できる状態 j が YES のときは 1，NO のときは 0 が入る．そして指示番号 x において，最後に保存した保管ステータスを l とし，y 番目の保管ステータスと最後に保存した保管ステータス l を比較したものを，比較ステータスと呼ぶ．保管ステータス同士の比較ステータスを C_{xyl} としたとき，数式は

$$C_{xyl} = \begin{bmatrix} c_{xyl11} & c_{xyl12} & \cdots & c_{xyl1n} \\ c_{xyl21} & c_{xyl22} & \cdots & c_{xyl2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{xylm1} & c_{xylm2} & \cdots & c_{xylmn} \end{bmatrix} \tag{3.3}$$

となる．C_{xyl} は B_{xy} と B_{xl} を比較し，b_{xyij} と b_{xlij} の値が一致している時，c_{xylij} には 1 が入り，そうでない場合は 0 が入る．数式で表すと

$$c_{xylij} = \begin{cases} 1 & (b_{xyij} = b_{xlij}) \\ 0 & (\text{上記以外}) \end{cases} \tag{3.4}$$

となる．

そして，保管ステータス同士が類似しているかどうかを調べる類似判定を行う必要がある．保管ステータスの類似判定の値を D_{xyl} としたとき，その式は

$$D_{xyl} = \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} c_{xylij}}{nm} \tag{3.5}$$

となる．

D_{xyl} の値が 90% 以上ならば類似していると判断する．D_{xyl} の計算を比較する保管ステータス分行い，類似したものが 5 つ以上ある場合はグループを作成する．

次に作成したグループの中から平均的なステータスを求める．まず，作成されたグループ内の保管ステータスすべてを対象にそれぞれの項目が YES だったときの確率を求める．選手 IDj の選手において，メタ AI が認識できる状態 i の項目が YES だったときの確率を項目の平均値と呼び，E_{xij} としたときその数式は

$$E_{xij} = \frac{\sum_{n=1}^{m} e_n}{m} \tag{3.6}$$

となる．

E_{xij} は YES だったときの確率であり，m がグループ内の保管ステータスの総数である．n はグループ内の保管ステータスの番号を示し，e_n は n 番目の保管ステータスが YES の時は 1,NO のときは 0 が当てはめられる．この計算を 220 あるステータスの項目分行う．それを表にしたものが，表 3.2 である．

表 3.2　グループ内における保管ステータスの各項目の確率

	選手 ID								
	1	2	3	4	5	6	7	...	22
1. チームで一番ボールに近い	62	28	0	0	4	0	0	...	0
2. チームで二番目にボールに近い	12	34	9	0	0	0	0	...	0
3. 自分がボールを所持している	58	17	0	0	4	0	0	...	0
4. ボールを味方が所持している	74	74	74	74	74	74	74	...	12
5. 自分が敵陣にいるかどうか	89	78	61	62	23	26	18	...	8
6. 自分が右サイドにいるかどうか	54	0	79	32	0	0	0	...	0
7. 自分が左サイドにいるかどうか	10	42	0	0	22	88	89	...	60
8. 自分がセンターにいるかどうか	36	58	21	68	78	12	11	...	40
9. 自分が『上がれ』の指示を受けているかどうか	80	80	80	80	80	80	80	...	0
10. 自分が『下がれ』の指示を受けているかどうか	20	20	20	20	20	20	20	...	0

次に表 3.3 にて取得した各項目の YES だったときの確率を，50 %以上のときは YES，それ以下を NO として扱う．それにより，表 3.2 に示したデータを論理値に変換する．変換した様子を表にしたものを表 3.3 に示す．これが指示実行ステータスとなる．

表 3.3　指示『上がれ』における指示実行ステータス

	選手 ID								
	1	2	3	4	5	6	7	...	22
1. チームで一番ボールに近い	Y	N	N	N	N	N	N	...	N
2. チームで二番目にボールに近い	N	N	N	N	N	N	N	...	N
3. 自分がボールを所持している	Y	N	N	N	N	N	N	...	N
4. ボールを味方が所持している	Y	Y	Y	Y	Y	Y	Y	...	N
5. 自分が敵陣にいるかどうか	Y	Y	Y	Y	N	N	N	...	N
6. 自分が右サイドにいるかどうか	Y	N	Y	N	N	N	N	...	N
7. 自分が左サイドにいるかどうか	N	N	N	N	N	Y	Y	...	Y
8. 自分がセンターにいるかどうか	N	Y	N	Y	Y	N	N	...	N
9. 自分が『上がれ』の指示を受けているかどうか	Y	Y	Y	Y	Y	Y	Y	...	N
10. 自分が『下がれ』の指示を受けているかどうか	N	N	N	N	N	N	N	...	N

この指示用実行テータスを F_{xz} としたとき数式は

$$F_{xz} = \begin{bmatrix} f_{xz11} & f_{xz12} & \cdots & f_{xz1n} \\ f_{xz21} & f_{xz22} & \cdots & f_{xz2n} \\ \vdots & \vdots & \ddots & \vdots \\ f_{xzm1} & f_{xzm2} & \cdots & f_{xzmn} \end{bmatrix} \tag{3.7}$$

となる．

z は指示番号 x において z 番目に作成されたグループの番号を示す．f_{xzij} の値は，F_{xyij} において選手 IDi の選手の認識できる状態 j が YES のときは 1，NO のときは 0 が入る．

3.4　自動指示の実行

現行ステータスと指示実行ステータスとの比較を行う．比較方法は C_{xyl} の保管ステータス同士の比較ステータスと同様に行う．現行ステータス A と指示用のステータス F_{xz} との比較ステータスを G_{xz} としたとき数式は

$$G_{xz} = \begin{bmatrix} g_{xz11} & g_{xz12} & \cdots & g_{xz1n} \\ g_{xz21} & g_{xz22} & \cdots & g_{xz2n} \\ \vdots & \vdots & \ddots & \vdots \\ g_{xzm1} & g_{xzm2} & \cdots & g_{xzmn} \end{bmatrix} \tag{3.8}$$

となる．

a_{ij} と f_{xzij} の値が一致している時，g_{xyij} には 1 が入り，そうでない場合は 0 が入る．g_{xyij} を数式で表すと

$$g_{xylij} = \begin{cases} 1 & (a_{ij} = f_{xzij}) \\ 0 & (\text{上記以外}) \end{cases} \tag{3.9}$$

となる．

この比較結果の数式 G_{xz} を使って現行ステータスと，指示実行ステータスが類似しているかどうかを判断する．これも D_{xyl} の類似判定とほぼ同様である．現行ステータスと指示実行ステータスの類似判断を行う数式を H としたとき，その数式は

$$H = \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} g_{xzij}}{nm} \tag{3.10}$$

となる．

H は A と G_{xz} がどれだけ一致しているかを百分率で示した値である．本研究では D の値が 90% を越えた時点で類似しているものとして扱い，90% を越えている状態の時にプレイヤーの操作と同じように自動で指示行動を行う．

ゲーム中は常にこの比較を行い続け，グループが複数作成されている場合は，その数だけ現行ステータスとの比較を行う．ただし，別々の指示が一致率が同時に 90% を越えてしまう場合がある．例えば指示『上がれ』のグループとの一致率が 92%，指示『下がれ』のグループとの一致率が 94% というような状況になった場合は一致率の高い指示を優先して自動指示を行う．また，自

動指示は基本的に一致率が 90% を越えた瞬間に，プレイヤーがキー入力により指示を送るのと同様の処理が行われる．ただし，プレイヤーのキー入力と違い自動指示の場合は一致率が数秒間継続的に 90% を越え続ける場合がある．その場合 1 秒置きに再度同じ指示を出すものとした．

第 4 章

実験

4.1 評価実験の内容

本研究の提案手法により，プレイヤーが指示を行いやすいタイミングを検出し，自動で指示を行うシステムを開発した．そこで開発したシステムの自動指示機能が，実際にプレイヤーの意図を汲み取った動きをしているかどうかの検証を行った．

検証方法は，ユーザーテストによる評価実験である．評価実験の方法は以下の通りである．

1. 本研究で制作したゲームを被験者に 10 分間自由に遊んでもらう．
2. ゲーム中被験者が可能な操作は「上がれ」と「下がれ」の指示のみとする．
3. 終了 2 分前に手動の指示ができなくなり、自動指示に切り替える．

   このことは被験者には伝えない．ただし被験者には途中で動きがおかしくなっても指示を続けてほしいと伝えておく．

4.2 検証の方法

実験により取得した情報から，プレイヤーによる手動指示と AI による自動指示がどれだけ一致しているのかを検証した．

キー操作による手動指示は，キーを押した瞬間のみ入力情報を伝えるのに対し，自動指示は現在のステータスと指示用のステータスが 90% 以上一致し続ければ，継続的に指示が出続ける．これを現したのが図 4.1 である．今回の検証では手動指示のタイミングと自動指示のタイミングが一致していれば，自動指示が上手く作動しているものとする．図 4.1 では手動指示を 3 回行い，2 回は自動指示と同じタイミングで指示を行っている．このような場合，手動指示と自動指示の一致率は 2/3（66.6%）として考える．

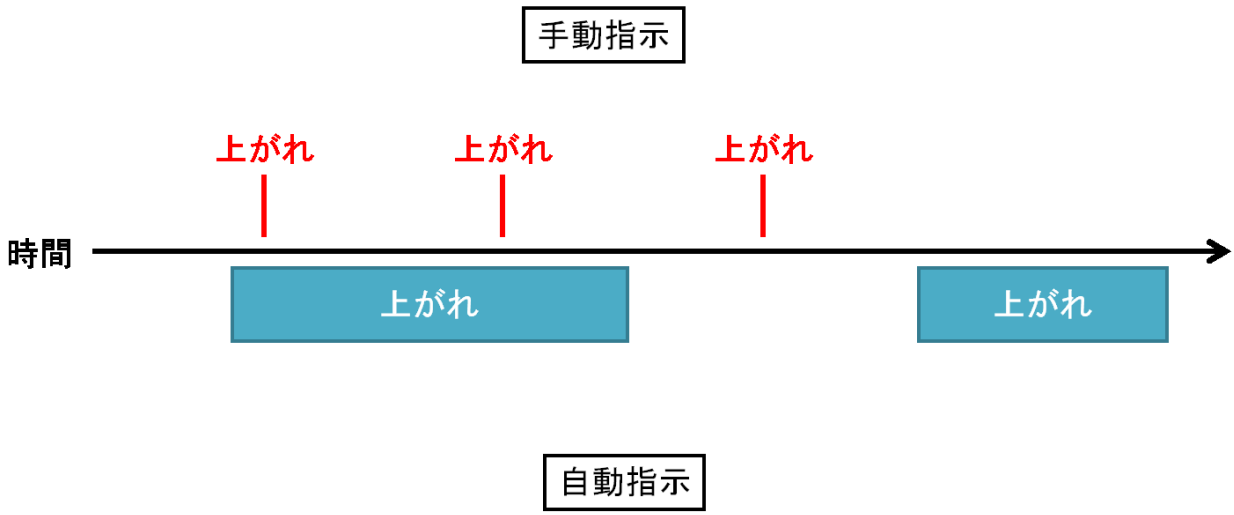

図 4.1　検証のイメージ

今回の実験では被験者に 10 分間ゲームを遊ばせる過程の中で，最初の 8 分間でプレイヤーの指示情報を学習させる．次に終了 2 分前に手動指示を停止し，その期間中の手動指示と自動指示がどれだけ一致していたかを求める．

4.3　一致率の計算法

この実験では基本的に手動指示と自動指示の一致率が高いほどプレイヤーにとって理想的な動きをしていると想定した．しかし，予備実験として何度かユーザーテストを行った際に，どれだけ理想的な動きをしていたかという 5 段階評価で答える質問を被験者に行ったところ，同じ評価でありながら手動指示と自動指示の一致率が大きく離れるケースがあった．その理由として考えられるのは被験者のプレイスタイルの違いである．

また実験中に被験者によってはあまりキー入力を行わないケースや，逆に何度もキーを連打するケースが存在した．本研究ではユーザーテストの終了 2 分前で手動指示を行った回数に対しどれだけ自動指示と一致しているかを方法を取っているため，上記のようなケースでは実際の評価と手動指示と自動指示の一致率が一致しない可能性がある．

そこで予備実験の方法に加え，実際の評価とプレイヤーの入力情報から得られる評価に差が出

にくくなると考えられる方法を途中から追加し，2 つの方法で計測を行った．計測方法は以下の通りである．

1. 方法 1

   すべてのキー入力時の情報を対照に，どれだけ自動指示と一致していたかを計測する．予備実験と同じ方法.

2. 方法 2

   キー入力後 1 秒間は入力を受け付けない．キー入力を受け付けていない間の入力情報は無視し，どれだけ自動指示と一致していたかを計測する．

4.4 実験結果

ユーザーテストを行い，また実験後以下のようなアンケートをとった.

1. 質問 1

   途中から指示ができなくなっていることに気がついたか

2. 質問 2

   途中から指示が自動で行われていることに気がついたか

実験の結果は表 4.1 のようになった.

表 4.1　実験結果

	方法 1:上がれ	方法 1:下がれ	方法 2:上がれ	方法 2:下がれ	質問 1	質問 2
被験者 1	33%	48%	-	-	YES	YES
被験者 2	10%	81%	-	-	YES	NO
被験者 3	30%	33%	-	-	YES	YES
被験者 4	92%	95%	-	-	NO	NO
被験者 5	65%	78%	-	-	YES	NO
被験者 6	69%	88%	-	-	YES	YES
被験者 7	89%	90%	-	-	NO	YES
被験者 8	44%	38%	30%	41%	YES	NO
被験者 9	60%	70%	54%	92%	NO	NO
被験者 10	55%	36%	50%	38%	YES	NO
被験者 11	60%	46%	63%	46%	NO	YES
被験者 12	53%	61%	50%	65%	YES	YES
被験者 13	64%	76%	56%	78%	YES	YES
被験者 14	84%	92%	78%	94%	NO	NO
被験者 15	28%	42%	29%	42%	YES	YES
平均	56%	64%	45%	62%	-	-

この結果からそれぞれの方法，指示に対する一致率の割合を図 4.2，4.3，4.4，4.3 に示す．

比較的一致率の高い被験者 4,9,14 は二つの質問に対しどちらとも NO と答えた．また，比較的一致率の低い被験者 1,3,12 は二つの質問に対しどちらとも YES と答えた．このことから一致率が高い程違和感のない自動指示ができていると考えられる．

実験の途中から方法 2 を追加したが方法 1 と方法 2 で大幅に結果が変わることは無かった．

他の手法と比較した場合，ベイジアンネットワークを用いてプレイヤーの考えを推測し，隠れマルコフモデルによって各選手の行動を決定する手法を提案している川上ら [28] の研究では，25 人の学生を対象に本手法と同様の実験を行った．実験の結果，実験後のアンケートでは味方選手は想定通りに動いたか，という質問に対し 10 段階評価で平均は 4.8 という結果であった．

自動指示と手動指示の一致率は，方法 2 の上がれ以外は全て 50% を超えており質問 2 に対しても 15 人中 8 人が YES と答えていることから，既存手法よりも高い結果が出せていると思われる．

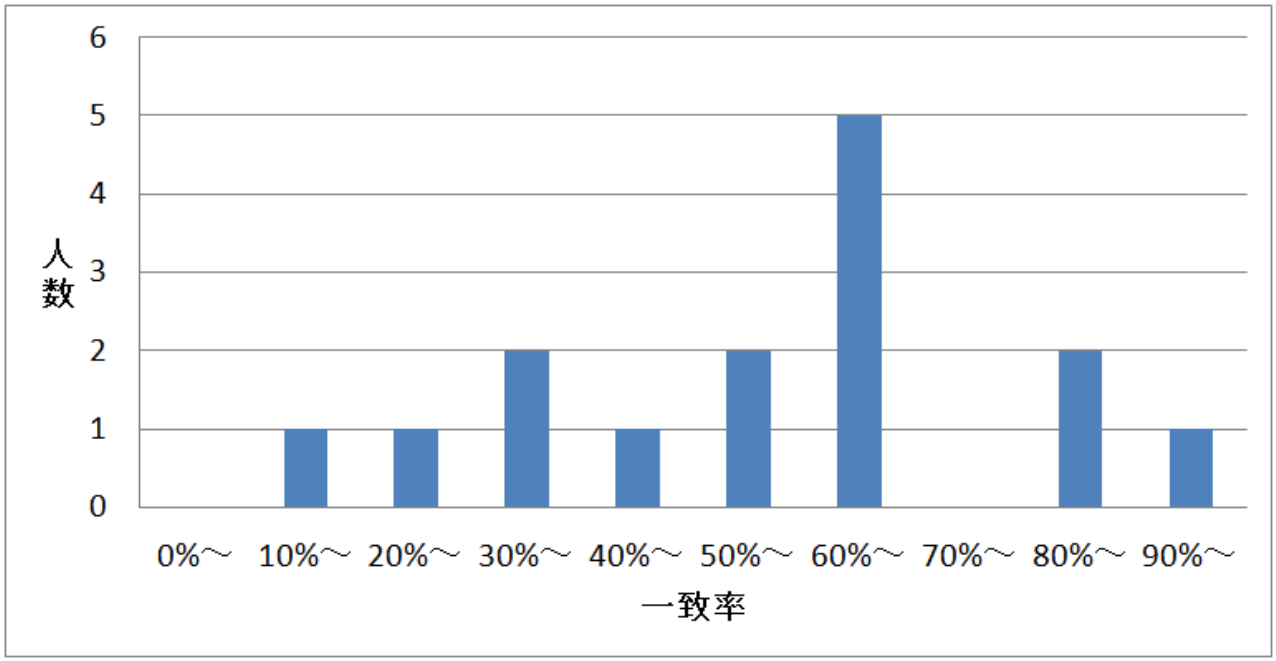

図 4.2　方法 1:上がれの一致率

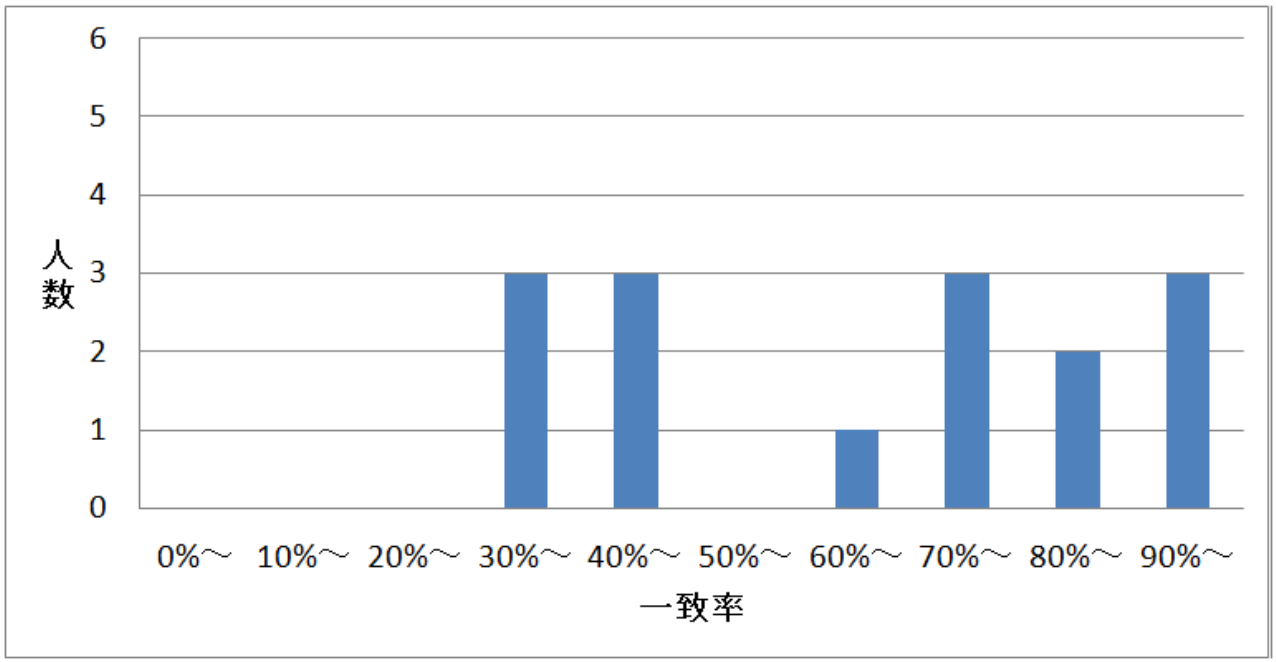

図 4.3　方法 1:下がれの一致率

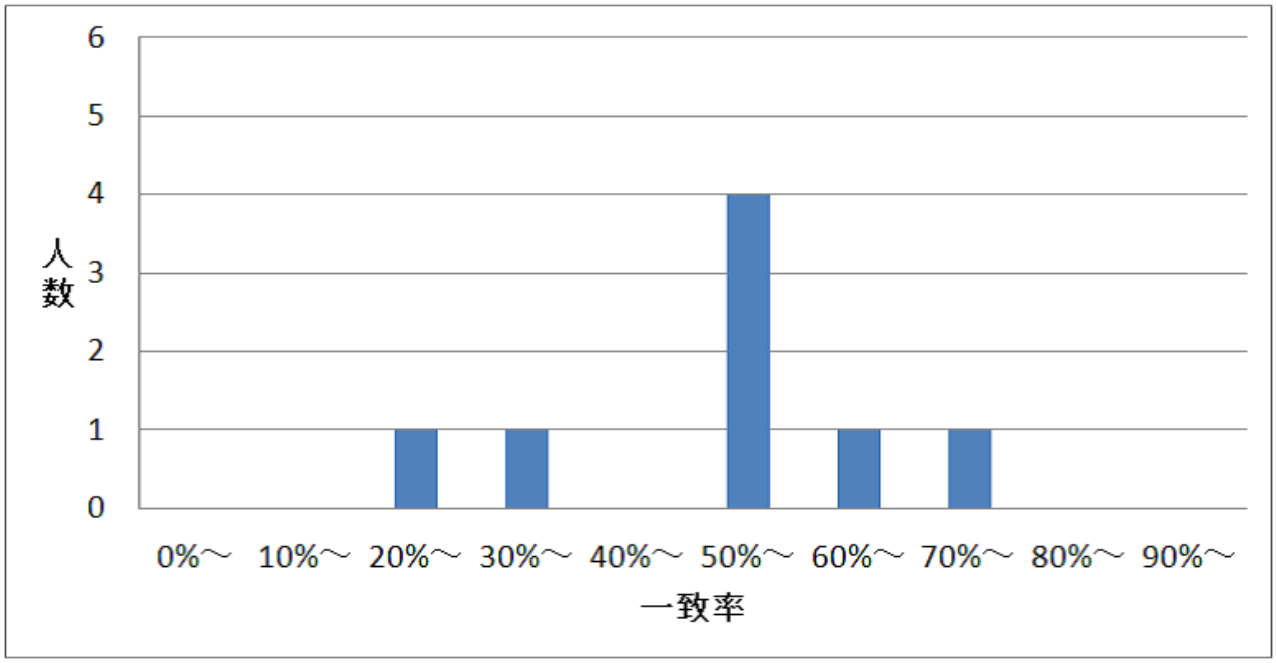

図 4.4　方法 2:上がれの一致率

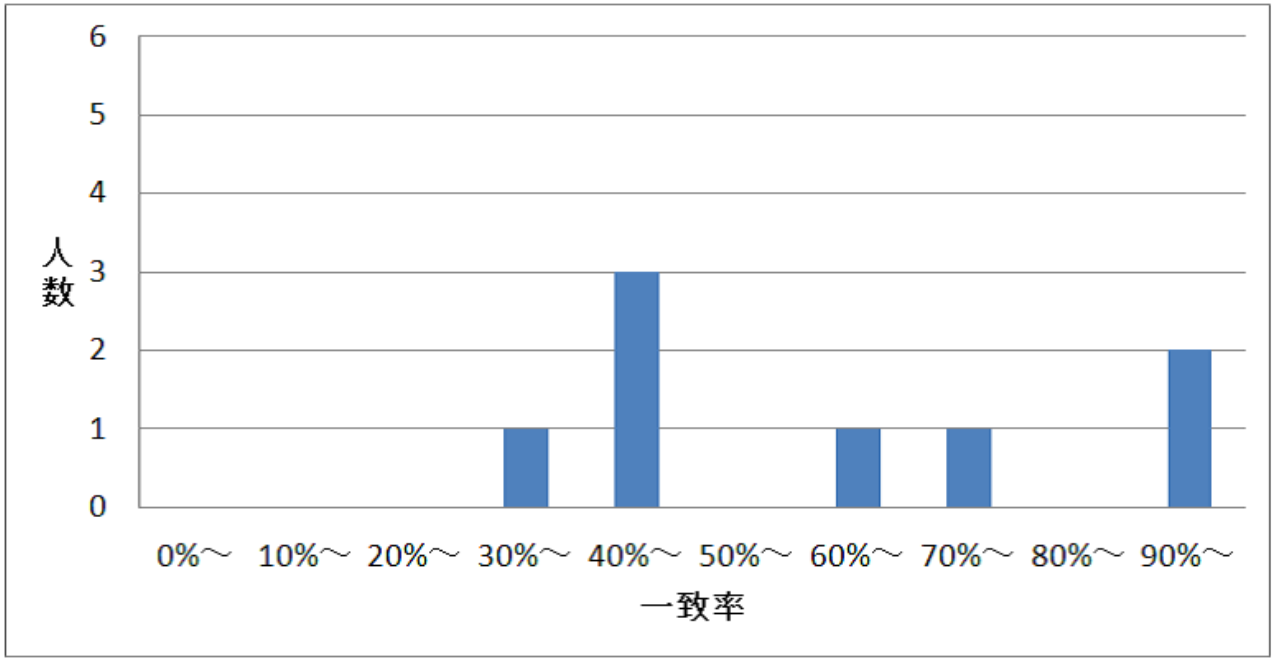

図 4.5　方法 2:下がれの一致率

第 5 章

まとめと今後の展望

本研究では，プレイヤーが行った過去の指示情報からプレイヤーの意図に合わせた行動を可能とする AI の実装行った．そして，プレイヤーが指示を行ったタイミングと類似したタイミングで自動指示が行うことができた．しかし，ユーザーテストの結果，まだ自動指示の精度にまだ問題があることが分かった．

より精度を上げる方法として，ゲームをするときプレイヤーがどこに注目をしているのかについてより理解する必要がある．本研究の手法ではプレイヤーが指示を送る際に参考にすると思われる項目を 10 個用意したが，それぞれの項目に対してどこに重みを置くかについては考慮しなかった．また，自動指示を実行する際に現行ステータスと指示実行ステータスが 90% 以上一致していることを条件としたが，明確な基準がある訳ではない．これらの問題を解決するには，プレイヤーがどこに注目しているのかをより研究する必要がある．そのためにもユーザーテストを繰り返し，プレイヤーが指示を行いやすい傾向やその基準を見つけていけば，本研究による自動指示の精度をさらに高めることができると考えられる．

本研究では研究対象のゲームをサッカーゲームに絞った．これは，指示を送った際の動きが分かりやすく，またゲームのルールの認知率が高いスポーツゲームとしてサッカーゲームを選んだという理由がある．ただし，本研究の手法は基本的に自動で動く味方の AI に対し指示を出すことのできるゲームであれば，どのようなゲームでも対象になるものと考えられる．

今後の展望として様々な展開，シチュエーションに対応できるのかどうかについても検討したい．今回の研究では 10 分間という短い試合時間の中で評価実験を行ったが，これを数時間遊ぶことを想定し，さらに敵のチームに関しては選手の能力や AI に多様性を持たせることで，また違う結果が出ると考えられる．多様性のある敵チームとの対戦を重ねることで，それぞれの敵の弱点を突くような動きができればなお良い．また，違うプレイヤー同士でそれぞれが学習させた選手同士を対戦させることで，面白い結果が出るかもしれない．

謝辞

本研究を締めくくるにあたり、ご指導ならびに適切なご助言を下さいました先生方に感謝の意を表します。また、研究するに当たってユーザーテストに付き合ってくれた皆様、並びに研究室のメンバーに深く感謝致します。

参考文献

[1] Konami Digital Entertainment. ウイニングイレブン 2016 オフィシャル WEB サイト. http://www.konami.jp/we/2016/index.php5. 参照:2015.12.21.

[2] 千裕加藤, 誠三輪, 慶雅鶴岡, 隆近山. ターン制ストラテジーゲームにおける戦術決定のための uct 探索とその効率化. ゲームプログラミングワークショップ 2013 論文集, pp. 138–145, nov 2013.

[3] 明紀中川, 翔太/ 逢坂, 智哉柴崎. ニューラルネットワークによる格闘ゲーム ai の難易度調整及び行動多様性向上手法. 全国大会講演論文集, 第 70 回, pp. 801–802, mar 2008.

[4] 安武諒, 山口崇志, マッキンケネスジェームス, 永井保夫, ヤスタケリョウ, ヤマグチタカシ, マッキンケネスジェームス, ナガイヤスオ, Yasutake Ryo, Yamaguchi Takashi, Mackin Kenneth James, Nagai Yasuo. チューリングテストによるゲーム ai の客観的評価. 東京情報大学研究論集, Vol. 16, No. 1, pp. 33–46, sep 2012.

[5] 木下茂雄. 効率的ゲーム開発のためのゲーム ai 構築支援環境の提案. 第 73 回全国大会講演論文集, Vol. 2011, No. 1, pp. 191–192, mar 2011.

[6] 藤田肇, 石井信. 部分観測カードゲームのためのモデル同定型強化学習 (バイオサイバネティックス, ニューロコンピューティング). 電子情報通信学会論文誌. D-II, 情報・システム, II-パターン処理, Vol. 88, No. 11, pp. 2277–2287, nov 2005.

[7] 佐々木宣介. 機械学習と自動プレイを用いた将棋類の類似度比較について. 情報処理学会研究報告ゲーム情報学（GI）, Vol. 2006, No. 23, pp. 41–48, mar 2006.

[8] 星野孝総, 亀井且有. ファジィ環境評価型強化学習の lightsout ゲームへの応用と探索における迂回行動の回避. システム制御情報学会論文誌, Vol. 14, No. 8, pp. 395–401, aug 2001.

[9] 掘之内剛史, 河口信夫, 稲垣康善. 遺伝的プログラミングを用いたゲームの局面評価関数の学習. 電子情報通信学会技術研究報告. AI, 人工知能と知識処理, Vol. 96, No. 594, pp. 17–24, mar 1997.

[10] 稲垣裕伸, 魚住超, 小野功一. 報酬変動型繰返しゲームにおけるエージェントの協調行動の性質. 情報処理学会論文誌, Vol. 40, No. 3, pp. 1056–1064, mar 1999.

[11] 梶原健吾, 鳥海不二夫, 大橋弘忠, 大澤博隆, 片上大輔, 稲葉通将, 篠田孝祐, 西野順二. 強化学習を用いた人狼における最適戦略の抽出. 第 76 回全国大会講演論文集, Vol. 2014, No. 1, pp. 597–598, mar 2014.

[12] 伊藤昭, 大橋資紀, 寺田和恵. 非零和ゲームの強化学習一相手の行動を読むプログラム. 情報処理学会研究報告知能と複雑系（ICS）, Vol. 2005, No. 109, pp. 53–60, nov 2005.

[13] 星野准一, 田中彰人, 濱名克季. 模倣学習により成長する格闘ゲームキャラクタ. 情報処理学会論文誌, Vol. 49, No. 7, pp. 2539–2548, jul 2008.

[14] ロボカップ日本委員会事務局. RoboCup Japanese National Committee Official Homepage. http://www.robocup.or.jp/. 参照:2015.12.21.

[15] 森下卓哉, 久保長徳, 青柳博紀, 河原林友美, 下羅弘樹, 廣嶋恭一, 西野順二, 小高友宏, 小倉久和. Robocup におけるサッカーエージェントの設計. 福井大学工学部研究報告, Vol. 47, No. 2, pp. 277–290, sep 1999.

[16] 坂本, 聖也, 尾関, 基行, 岡, 夏樹. Robocup サッカーの keepaway サブタスクにおけるパスの受け手の強化学習. 岡山理科大学紀要. A, 自然科学, Vol. 29, , 2009.

[17] 笹岡久行, 村木俊介. E-023 robocup サッカーエージェントに対する強化学習を用いた行動選択手法の提案 (e. 自然言語・文書・ゲーム). 情報科学技術フォーラム一般講演論文集, Vol. 3, No. 2, pp. 163–164, aug 2004.

[18] 熊田陽一郎, 植田一博. 予測能力を持つサッカーエージェントによる協調戦術の獲得. 人工知能学会誌, Vol. 16, No. 1, p. 153, jan 2001.

[19] 並川直樹, 小野玄基, 横山智史, 高谷将裕, 中島智晴, 石渕久生. ロボカップサッカーにおけるニューラルネットワークを用いた模倣学習. 日本知能情報ファジィ学会 ファジィ システム

シンポジウム 講演論文集, Vol. 22, pp. 159–159, 2006.

[20] 浅田稔, 北野宏明. ロボカップ戦略: 研究プロジェクトとしての意義と価値. 日本ロボット学会誌, Vol. 18, No. 8, pp. 1081–1084, 2000.

[21] 杉原一臣, 芝田稔, 釜田友希, 大熊一正, 山西輝也, 魚崎勝司. Robocup サッカー 3d シミュレーションリーグにおける試合状況の類似性を考慮した行動選択確率の更新. 日本知能情報ファジィ学会 ファジィ システム シンポジウム 講演論文集, Vol. 27, pp. 242–242, 2011.

[22] 宮嶋健太, 屋敷仁人, 杉原一臣, 大熊一正, 山西輝也, 魚崎勝司. Robocup サッカー 3d シミュレーションリーグにおけるエージェントへの確率的行動選択の実装とその改良. 第 73 回全国大会講演論文集, Vol. 2011, No. 1, pp. 499–500, mar 2011.

[23] 杉原一臣, 芝田稔, 釜田友希, 大熊一正, 山西輝也, 魚崎勝司. Robocup サッカー 3d シミュレーションリーグにおける行動選択確率の更新と試合結果への影響. 第 74 回全国大会講演論文集, Vol. 2012, No. 1, pp. 49–50, mar 2012.

[24] 内種岳詞, 山本将司, 畠中利治. Robocup 3d soccer simulation における歩行動作獲得. 日本知能情報ファジィ学会 ファジィ システム シンポジウム 講演論文集, Vol. 27, pp. 241–241, 2011.

[25] ロボカップ日本委員会事務局. SimJP. http://rc-oz.osdn.jp/pukiwiki/index.php?SimJP. 参照:2015.12.21.

[26] 金重徹, 片山謙吾, 南原英生, 成久洋之. Robocup サッカーシミュレーションリーグにおける強化学習の有効性. 人工知能学会 AI チャレンジ研究会, Vol. 41, pp. 105–112, 2005.

[27] 井上勇気. ゲーム内エージェントの強化学習に用いる報酬値探索へのＧＡの適用. 法政大学大学院情報科学研究科情報科学専攻修士課程修了, 2009.

[28] 武尊川上, 規雄富井. サッカーゲームにおける味方選手の行動決定アルゴリズム. 【全国大会】平成 23 年電気学会全国大会論文集, pp. 52–53, mar 2011.

[29] Mat Buckland , 松田晃一. 実例で学ぶゲーム AI プログラミング. 株式会社オライリー・ジャパン, 2005.

[30] David M.Bourg , Glen Seemann , 株式会社クイープ. ゲーム開発者のための AI 入門. 株式会社オライリー・ジャパン, 2005.

[31] 陽一郎三宅. ディジタルゲームにおける人工知能技術の応用の現在. 人工知能, Vol. 30, No. 1, pp. 45–64, jan 2015.