

## サッカーシミュレーションにおける声を使った協調行動

吉田 真士\* 白井 治彦\*\* 高橋 勇\*\*  
黒岩 丈介\*\* 小高 知宏\*\* 小倉 久和\*\*

### Cooperation action using the voice in a soccer simulation

Masashi YOSHIDA\*, Haruhiko SHIRAI\*\*, Isamu TAKAHASHI\*\*,  
Jousuke KUROIWA\*\*, Tomohiro ODAKA\*\* and Hisakazu OGURA\*\*

(Received January 31, 2007)

In multi-agent system, a cooperation behavior is an attractive problem. In the RoboCupSoccer simulation the environment of each agent dynamically and suddenly changes. Therefore, it is difficult to solve the problem in the cooperation behavior. The difficulty would come from the uncertainty of the position of the other agents and the ball, getting from the visual information, and from the fact that each agent can not know the internal state of the other agents. In order to overcome the problem, we focus on the voice as one of communication tools and design the strategy of the soccer team.

**Key Words:** RoboCup, Soccer Simulation, Multi Agent, University of Fukui

### 1. はじめに

ロボカップは、複数のサッカーエージェントを用いて対戦させるものでマルチエージェントの問題を提供するためやロボット工学と人工知能の融合・発展のために発足された。今では全国で多数の研究者や学生が参加する一大国際プロジェクトになっている。1) ロボカップは5つの部門から構成されており、小型ロボットリーグ・中型ロボットリーグ・四足ロボットリーグ・シミュレーションリーグ・ヒューマノイドリーグに分けられる。ロボカップ最終目標を「2050年までに、自律型ヒューマノイドロボットで人間している。そして、その過程で生み出された技術が社のサッカーの世界チャンピオンに勝つこと」と設定会基盤や次世代産業など、いろいろな分野に波及

及することを目指している。今回の研究ではロボカップの部門の一つであるシミュレーションリーグを用いている。シミュレーションリーグとは、それぞれの参加者がサッカーエージェントの選手のプログラムを用意しそれを架空2次元フィールド上で対戦させるリーグであり、その制御はサッカーサーバが行っている。サッカーサーバとクライアント間の通信方法はUDP/IPを用いて通信している。サッカーサーバはクライアントから送られてくる命令コマンドを実行させ、それを2次元フィールドに反映させてクライアント側にデータを送るというシステムで構成されている。また現実の世界のサッカーに近づけるためにサッカーサーバではノイズや量子化誤差を取り入れている。ノイズは物体が必ずしも思ったところに飛んでくれないということを実現している。量子化誤差は視覚情報に用いられ、遠くにいるプレイヤーやボールに関する情報が不確実なものとなってきている。また、エージェントの視野範囲は限られている。そのような状況のなかでマルチエージェントの環境を生かしエージェント同士が協力しあい、より大きな効果を得ることはロボカップにとってとても重要な課題の一つである。そこで本研究では、重要

\* 工学研究科知能システム工学専攻

\*\* 知能システム工学科

\* Human and Artificial Intelligence Systems Course,  
Graduate School of Engineering

\*\* Dept. of Human and Artificial Intelligence Systems

な課題だと考えられる効果的な協調行動の方法として、聴覚情報を利用する手法を提案している。ロボカップでは実際のサッカーと同じように、他のエージェントにメッセージを伝えることができる。このメッセージをうまく使うことにより視覚情報だけでは難しかった有効な協調行動を行い、マルチエージェントの環境を有効に生かした行動選択を行うエージェントの作成を行った。

## 2. 声を使った協調行動

### 2.1 ロボカップにおける協調行動

ロボカップはマルチエージェントの環境を提供するために考案されたものであるが、協調行動をどのように実現させるかがロボカップにおいて重要なテーマの一つとなっている。ロボカップにおける協調行動とは、複数のエージェントが協力して優位な状況を作りだしたり自チームの危機を回避したりすることである。例えば単純な協調行動の例としてパスが上げられる。パスをすることによってボールを持っているエージェントに迫っている危機を回避することができたり、少ない時間とスタミナの消費で前線にボールを運べたりすることができる。

またロボカップの協調行動の実現手法は今までさまざまな手法<sup>2),3),4)</sup>が提案されてきた。今回この論文で提案する手法はエージェントによる声を使った手法である。エージェントの聴覚の情報を使うことによって、有効な協調行動を行う手法を実現している。

### 2.2 協調行動における声の役割

ロボカップにて声はさまざまな使い方ができると考えられる。まず一つに、声を協調行動の発動のトリガーとして使うという使用方法がある。サッカーの作戦において、複数のエージェントがあらかじめ決められた動きをしてパターン攻撃や連携守備をする作戦が一般的にある。この協調行動の作戦に参加する各エージェントらが、パターン行動を発動するタイミングを他のエージェントとうまく合わせなければならない。もし一人のエージェントでもパターン行動を発動しなかったり大幅に遅れてしまったりすると作戦は失敗となってしまう。エージェントには視野範囲が限られているので、視覚情報だけでは作戦発動のタイミングを逃してしまうことも十分に考えられる。そこで、そのパターン行動を発動する合図を音声によって行うことによってより確実に作戦を成功させることができる。

次に考えられる使用方法として、他のエージェントに自分もっている情報を送っての協調行動が挙げられる。ロボカップのサッカーシミュレーションリーグでは

実現モデルに近づけるために各エージェントは視野角度が制限されている。そのためエージェントは自分より後方の情報は取得することができない。そこでそれぞれのエージェントが持っている情報を他のエージェントが声を使って教えることによって広い範囲の情報をエージェントは取得できるようになる。多くの情報を取得できるようになるので、より適切な判断をすることができると考えられる。

3つ目に考えられる使用方法として、他のエージェントに自分の意思を伝えることができることである。視覚情報だけだと自分の考えていることを他のエージェントに伝えることが困難であるが声を使うことによって自分の意思をほぼ確実に伝えることができる。自分が何をしたいか、他のエージェントに何をしてほしいかを伝えることによって複数のエージェントによる協調的な行動を行うことができる。

上記に挙げたとおり協調行動における声の役割はいろいろあることが分かる。視覚情報だけでは精度の高い協調行動や作戦はなかなか難しいが上記のような聴覚情報も取り入れることによって協調行動がより容易に、完成度がより高くなるのではないかと考えられる。今回の論文では先ほど述べた一つ目と二つ目の声の使用法を用いて協調行動を実装している。一つ目に述べた声を利用して協調行動の具体的な実現例としてオフサイドトラップを取り上げ実装し、二つ目は効果的なパス相手を選択する方法として声を取り入れ実装している。

### 2.3 Robocupにおける音声について

サッカーサーバーコマンドの say コマンドを使用することでほかのエージェントにメッセージを送ることができる。say コマンドで送ることができるメッセージは、そのプレイヤーから 50 メートル以内までという制限があるので、遠く離れたエージェント間では声を使わなかった協調行動ができないことになる。また、say コマンドで伝達されるメッセージの長さは 10 バイト以内の文字列という決まりがあり、送る情報量はある程度限られている。また、発声されたメッセージは敵のプレイヤーも聞くことができる。

## 3. エージェントの設計

### 3.1 エージェントの設計について

エージェントの行動ルールは優先順位付きルールベース形式によって実装されている。優先順位付きルールベース形式とは、各行動ごとに優先順位と行動が発生するための条件を設けておき、与えられた情報を基に優先度の高いものから行動決定の判定を順に行っていく、条

件と一致したものについて実行するという方法である。

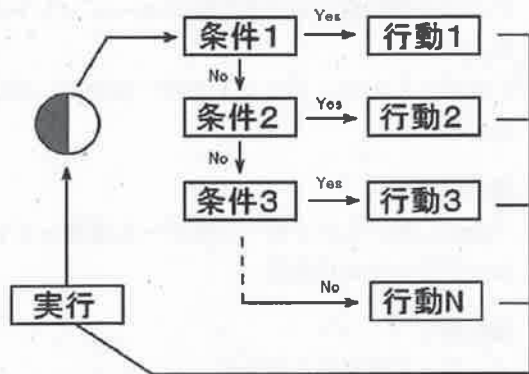


図1: 優先順位付きルールベース形式

### 3.2 声を使ったオフサイドトラップについて

ディフェンダーは敵に攻められている時に、オフサイドトラップの仕掛ける状況になっているかなっていないのかを調べるように設計してある。オフサイドトラップを仕掛ける状況は、簡単に述べるとディフェンスライン近くにいる敵に縦のロングパスができる状況になったときに設計した。ロングパスに限定した理由はオフサイドトラップに失敗した場合でもリスクが少ないからである。(パスが通るまでの時間が長いためオフサイドがとれなくてもパスカットができるため。)

オフサイドトラップの仕掛ける状況に気づくとディフェンダーは、他のディフェンダーはオフサイドトラップを仕掛ける状況になったことを知らせつつ前方へ移動する。メッセージを受け取ったディフェンダーもディフェンスラインを引き上げる動作をする。

#### 3.2.1 オフサイドトラッププログラムの構成

声を使用したオフサイドトラップは、今回作成したライブラリの「say-offside.act」と「offside.act」をプログラム本体から呼び出すことでディフェンスラインを押し上げる行動を行うことができる。これらのプログラム詳細は下記の通りである。

##### 「say-offside.act」について

- 発動条件
  - 敵が中央付近でボールを持っている。
  - 敵チームのフォワードの最前線にいるエージェントのx座標の値が味方チームのオフサイドラインの付近にある。

- ボールと最前線の敵のパスコースが開いている。

以上の3つの条件をすべて満たしたとき、

- 実行内容
  - ”アガレ”と発声しつつ、敵チームの一番前の選手(フォワードライン)のx座標の値を求めてそのx座標の値に5を引いた位置まで移動する。この移動するときのyの値は変化しないように移動するようにする。
- 終了条件
  - 目標地点までの移動が終わる

##### 「offside.act」について

- 発動条件
  - ”アガレ”というメッセージが聞こえたら発動
- 実行内容
  - 敵チームの一番前の選手(フォワードライン)のx座標の値を求めてそのx座標の値に5を引いた位置まで移動する。この移動するときのyの値は変化しないように移動するようにする。
- 条件終了
  - 目標地点までの移動が終わる

### 3.3 声を使ったパス相手選択について

次に声を用いてのパス相手の選択方法の方法について説明する。通常、パスは見えている範囲内にいる味方選手の中で最もパスに相応しい選手を選びその選手にパスをする。最もパスに相応しい選手を選ぶ方法は、味方選手の一人一人に評価値を与え最も高い点数の選手がパスに相応しい選手としている。しかしパスの候補は見えている範囲の選手に限られてくるのでマルチエージェントの環境を効率よく生かしているとはいえない。ボールを持っているエージェントが後方を確認することでパスの候補の選手は増えるが、もし後方にパスに相応しい選手がいなかった場合は完全に無駄な行動となりボールを取られるリスクが増すだけなので積極的にやるべきではない。

そこで今回は声を使用してボールを持っていない選手が、ボールを持っている選手に情報をあたえることでより有効なパスの選択を行う。ボールをもっていない選手は、自分が最もパスに相応しい選手だと考えたときはボールを持っている選手に「pass」というメッセージを送る。こうすることによりボールをもっている選手は、



どの方向に最もパスに相応しい選手がいるのかを理解しながらプレーすることができ、試合をより有利な展開で行うことができると考えた。

声を出す側のエージェントは、下記の手順を行いパスの要求を行う行動条件を満たしているのかを判定している。

- (1) 自分の評価値を求める
- (2) 5ステップ以内に見えたボールを持っていないエージェントについて、それぞれの評価値を求める
- (3) 「自分の評価値が最も高い」と「自分の評価値が閾値以上」の2つの行動条件を満たすとパスの要求を行う

また、(3)で明記の条件判定に用いている閾値の値は常に一定にはしていない。これは連続的な時間帯に声が出るのを防ぐためである。ボールを持ったエージェントは敵のプレッシャーがない時、声を聞いた方向を確認してパスを出すようにしてあるがパスを出す前にまた新しい情報が声によるメッセージにて来た場合は再び確認作業に入ってしまう。このように連続した時間帯にメッセージが続くと行動ができず、ボールを持ったエージェントはもたもたしてしまうことが考えられる。このことについての解決するために、情報を提供する側のエージェントは、声を出す際に過去の数ステップ間に誰か声を使っていた場合は、デフォルトの閾値をあげてよほどのことがない限り声を出さないようにした。閾値を上げることによりどうしてもパスがほしい時（パス評価関数の返す値がとてつ大きい時）に限り、連続した時間帯でも声を出すように設計している。

### 3.3.1 パス相手の評価方法

声を使用したパス相手の選択は、今回作成したライブラリの「say-free.act」、「keep-ball-find-player」、「pass.act」をプログラム本体から呼び出すことで実現している。これらのプログラム詳細は下記の通りである。なお下記のパスの評価関数とは5つの項目を引数としてフィールドから受け取り評価値を返す自作した関数である。「パス受け手のシュート角度」「パス受け手のポジション」「パスを出す側と受ける側の距離」「パスの角度」「パス受け手の周辺の敵の数」の5つの引数により評価値を求めている。

#### 「say-free.act」について

- 発動条件
  - パス評価関数の返す値が閾値以上
  - パス評価関数の返す値が他のエージェントよりも高い
  - 自チームのエージェントがボールをキープしている
- 実行内容
  - 「pass」というメッセージをボールを持っているエージェントに伝える
- 条件終了
  - メッセージを発声し終わる

#### 「keep-ball-find-player」について

- 発動条件
  - 自分がボールを持っている
  - 「pass」というメッセージを受け取る
- 実行内容
  - フィールドの状況次第で実行内容が異なる。(下図)
- 条件終了
  - 各行動が終わる

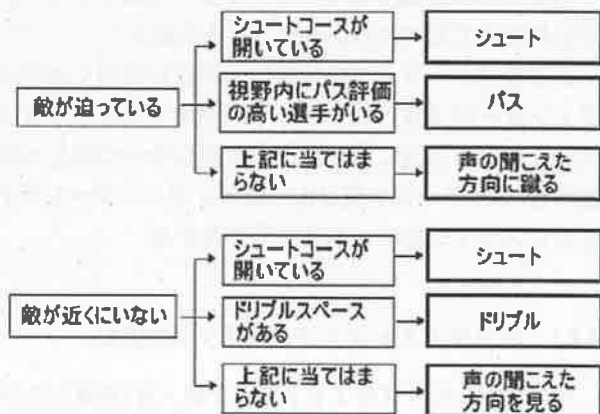


図 2: パスの要求をされた時の行動

#### 「pass.act」について

- 発動条件
  - (ボールを持っている) & (シュートチャンスではない) & (ドリブルスペースがない)
- 実行内容
  - 5ステップ以内に見えた味方エージェントに対しパス評価関数を使い評価値を与え、評価値の一番高かったエージェントに向かってボールを蹴る。

- 条件終了  
パスを出し終わる

#### 4. 実験

##### 4.1 オフサイドトラップの実験

ここでは、第3章での述べたオフサイドトラップが正しく機能しているかを確認した。また、声を掛けるエージェントの人数によって効果がどのように変わってくるのか、声を使わないで視覚情報のみでオフサイドトラップを仕掛けた場合とでは成功率がどのくらい変わってくるのかを調べた。

##### 4.1.1 ローカル環境による動作確認の実験

###### Case1:声をかけるエージェントが一人の場合

Case 1では、オフサイドトラップが本当に機能しているのか、確かめるためにオフサイドトラップがかけやすい環境を作成した。環境は以下のようなローカル環境を用意し実験を50回行った。オフサイドトラップの掛け声を掛けるのは特定の1人のみとしている。

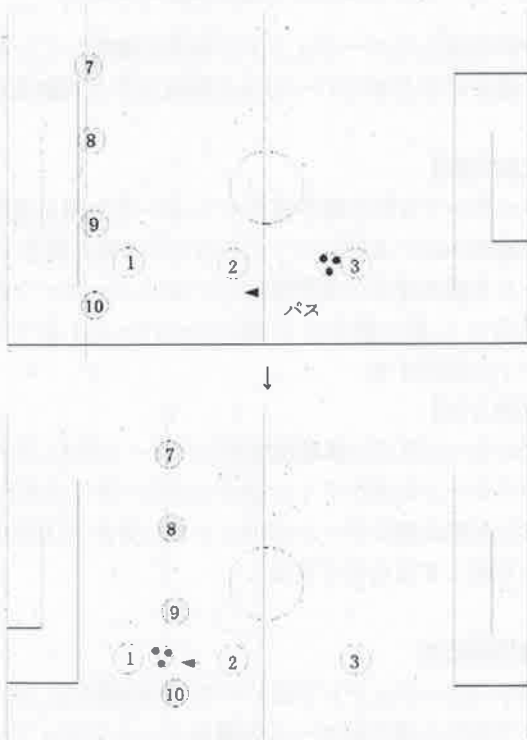


図3: ローカル環境による実験

###### 【味方側】

キーパー1人を配置し、味方ディフェンダーをy座標

の-35の線上に4人並べた。ディフェンダーの行動はオフサイドトラップの行動以外は行動しない(ボールを追ったりはしない)ものとする。オフサイドの声を発声させるのは図3上の背番号9の選手で、それを聞いた選手及び9番の選手は前方へ移動し、オフサイドラインを押し上げ敵の背番号1をオフサイドポジションに陥れる。

###### 【敵側】

敵のエージェントの背番号1, 2, 3を座標(33,-13), (8,-13), (-5,-13)上にならべて3の選手は2にパスをして、パスをもらった2の選手はボールをいったん止めて1番にパスをするように作成している。

###### ■ Case1: 結果

実験1を50回行った結果、オフサイドトラップの成功数は10回となり成功率は20%だった。どのディフェンダーも前へ移動するのが、遅れることなくフラットなラインを保っていたがトラップを仕掛けるタイミングがほんの少し(1~2ステップ)遅れてオフサイドトラップが失敗することが多かった。

###### Case 2: 声をかける人数が4人の場合

Case 2では、ディフェンダー全員(4人)の内誰でも命令してもよいとした場合での実験してみた。敵側の環境はさきほどのローカル環境と同じである。

予想される結果としてはCase 1の結果よりも、Case 2の声を出すエージェントの数が4人の方がよい結果が得られると考えられる。声を出すエージェントを特定の一人に限定してしまうと、その一人が判断が遅れてしまったりしてしまった場合はオフサイドトラップが失敗してしまうからである。

一方ディフェンダー1人が状況判断して声を出してオフサイドトラップをした場合は4人の内、誰かがオフサイドトラップの命令を出せば良いということになるので条件を満たしていた場合に特定の1人が声を出した時の場合より、より早く命令を出せる確率が高いことになる。

###### ■ Case2: 結果

実験2を50回行った結果、オフサイドトラップの成功数は27回で成功率は54%となった。特定の一人の選手のみが発声するときより高くなっていた

###### Case 3: 声を出さない場合でのオフサイドトラップ

Case 3では声を使わないで視覚情報のみでオフサイド

トラップを実装し実験することにする。ディフェンダー4人が条件判定をして条件が真の場合に前方へ移動する。敵の環境は Case 1, 2と同じローカル環境とする。予想される結果としては声を出していた時よりもオフサイドトラップの成功率は低いと思われる。4人全員がオフサイドトラップの条件が真と満たさなければ成功しないからである。それに最終ディフェンスラインも一番遅く判断をしたエージェントのx軸になってしまうことも、オフサイドトラップを成功させるには不都合なこととなってしまふことになる。

#### ■ Case3：結果

Case3を50回行った結果は下記のようになった。失敗する時のパターンとして多かったのはエージェントの内、一人だけが大きく出遅れてディフェンスラインの引き上げが遅くなってしまふ、というパターンだった。その例を図4を用いて説明すると、ディフェンダー(赤)の背番号7番が背番号8,9,10番に比べて大きく出遅れオフサイドトラップに失敗している。

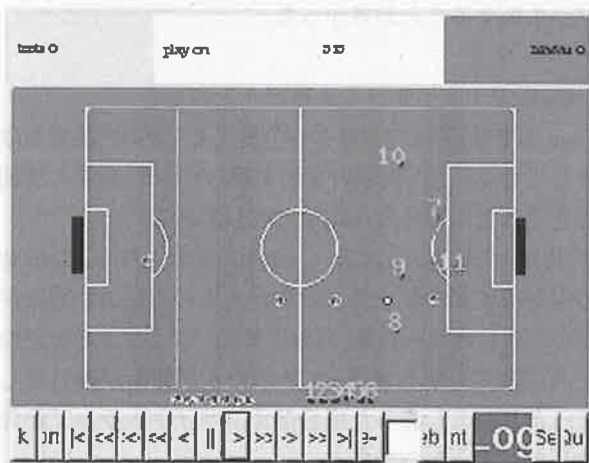


図4: 大きく遅れるディフェンダー

表1: 各 Case での結果

方法	成功回数	成功率
Case1:1人のみ声を使う	10回	20%
Case2:4人が声を使う	27回	54%
Case3:声なし	16回	32%

#### 4.1.2 実践による実験

このセッションでは実践の試合において、今回作成した声を用いてのオフサイドトラップが本当に有効なの

かを確かめる実験を行っている。声を使わない場合と声を使う場合の試合で結果を比べることによって声の有効性を確かめた。実験では Case2 と Case3 においてシミュレーションの試合を10試合ずつ行う実験を行った。対戦相手は Case1~3のようなオフサイドトラップを実装していないチームと行った。

#### 実験結果

実験結果は下記のようになった。声を使った時の方がオフサイドトラップの成功数が多く平均失点も少なく抑えることができた。このことにより、声がオフサイドトラップ発動のトリガーとしての役割が有効に作用し、試合結果にも好影響を与えているということが出来る。

表2: 実践による実験での結果

方法	平均成功回数	平均失点
Case2:4人が声を使う	7.1回	1.8失点
Case3:声なし	2.6回	2.4失点

## 4.2 声を使ったパス選択の実験

### 4.2.1 ローカル環境による動作確認の実験

今回作成したエージェントが本当に機能しているのかを確かめるためにローカルな環境を作り実験を行った。

#### 【実験環境】

フリーキックを行う味方選手×1, ボールを蹴る選手の視野範囲内にいる敵にマークされている味方選手×1, ボールを蹴る選手の視野範囲にいないノーマークの味方選手×1, 敵の選手×4, 敵のキーパー×1をフィールド上に配置する。

#### 【実験方法】

上記にそった任意の実験環境を5パターン作り、それぞれのパターンにおいて10回ずつフリーキックを行う。1回の実験は敵にボールがカットされるかパスが成功したら終了するものとする。

#### ■実験結果

声ありのエージェントではノーマークの選手は、フリーキックを行う選手がボールを蹴る4~5ステップ前に「pass」というメッセージを送っていることがプレイヤーログを見ることにより確認することができた。この声によるメッセージによってフリーキックを蹴る選手は、ノーマークの選手の存在を知ることができより有効なパスを出すことができた。ただし、稀に声を使った場合でもマークされた選手にパスをすることがあった。これはパ



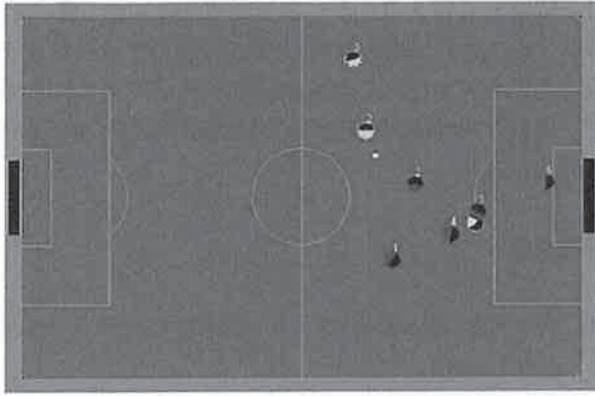


図 5: ローカル環境での実験例

スを要求するタイミングが遅れてしまいボールを蹴る瞬間やボールを蹴った直後にパスの要求を行ってしまったからである。タイミングの遅れは声を出すエージェントの視覚情報の誤差によるものである。

表 3: ローカル環境での動作実験結果

フリーキックでのパス相手		
方法	ノーマークの選手	マークされた選手
声あり	43 回	7 回
声なし	0 回	50 回

#### 4.2.2 実践による閾値を変えた実験

選手がパスを要求する条件は自チームがボールをキープしている時に、(1) パス評価関数の返す評価値が閾値以上である、(2) パス評価関数の返す評価値が他のエージェントよりも高い、この2つの条件を満たした時であるが(1)の判定時の閾値の値の設定によって、1試合におけるパスを要求する回数が大きくことなることが考えられる。ここでの実験は閾値の設定における試合内容(ボール支配率)と1試合における味方選手のパスを要求した回数を調べる実験を行った。

パスの評価関数は自作した関数であり、パスの相応しさによって0~200(評価値が高い方が高得点)の評価値を返している。今回の実験ではデフォルトの閾値を100~200で20刻みの設定したエージェントと声を使わないのエージェントそれぞれのボール支配率を調べた。対戦相手には研究室内のチームであるFU-Iと対戦させそれぞれの閾値で10試合ずつ行った。

##### ■実験結果

実験結果は下記の表のようになった。閾値が低いと、さ

ほどパスに相応しくなくてもパスを要求するため1試合におけるパスの要求回数も増えてしまい、また連続な時間帯にパスを要求するケースも多かった。連続した時間帯にパスを要求されるとボールを持った選手がもたもたしてしまいこのことが試合内容に悪影響を及ぼしていた。閾値が200の場合だと値が高すぎるためなかなか声を出す条件にならずパスの要求をほとんどしなかった。ボール支配率も声なしのエージェントとほとんど変わらなかった。

表 4: 閾値を変えた実験での結果(10試合の平均)

閾値	ボール支配率	パスを要求した回数
100	54.2 %	483 回
120	54.9 %	392 回
140	56.8 %	183 回
160	58.9 %	97 回
180	62.2 %	34 回
200	58.6 %	5 回
声なし	58.4 %	0 回

#### 4.3 実験の考察

##### 4.4 オフサイドトラップの考察

ローカル環境における実験においてCase1とCase2の比較(声を出すエージェントの数がディフェンダーが特定の一人の場合とディフェンダー全員の内誰でも命令してもよいことにした場合の比較)を見てみると明らかにディフェンダー全員の内誰でも命令してもよいことにした場合の方が良い結果が確認できた。これはディフェンダー4人の内一番速く判断したエージェントのx軸がほぼオフサイドトラップのラインとなるため、オフサイドトラップの条件を満たしてから、より早くラインを上げることができるためだと考えられる。Case1とCase3を比較すると声なしの方が結果がよかった。この原因として考えられるのが声を聞いてから行動を起こすまでの僅かな時間である。この時間がオフサイドトラップを失敗させる原因となっていると考えられる。Case2とCase3の比較で声ありの方が成功率が高い原因として考えられるのは、Case2がディフェンダーの4人が判断をしているのでオフサイドトラップの条件が発生してから声をだすまでの時間がCase1と比べて短かいからだと考えられる。

図は各Caseの時間の流れを表している。図の説明を見ると、Case3は声なしの場合なので条件に一番気づくのが遅い選手の時間帯がライン押上げの時間帯となっている。yCase1は特定の一人の選手が声を出すオフサイドトラップであるが、その特定の一人の選手がオフサイド

トラップの条件に気づく時間は、図の一番早く気づく時間帯のときもあれば、一番遅く気づく時間帯のこともあるので図ではその平均の時間ということで、赤線と青線の中央の時間帯で声をだすと表わしている。声を出してからライン押し上げまでに若干時間がかかるので、Case1のライン押し上げの時間帯がCase3よりも遅れることとなった。Case2は、4人で条件判定をしているので一番条件に早く気づいた選手が声をだすことができるので、一番早くライン押し上げができています。

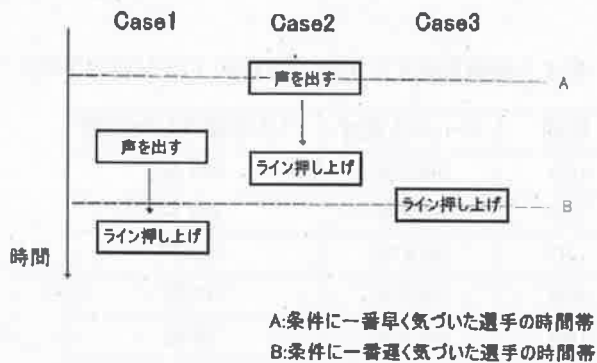


図 6: 各実験の行動時間帯

#### 4.5 声を使ったパス選択の考察

閾値を変えた実験において、閾値を適当な値に設定することによって声の使用が試合に好影響を及ぼすが、閾値が小さすぎると悪影響を及ぼしていることが分かった。たいしたパスチャンスでもない時にパス要求をしてもボールを持っているエージェントにとっては悪影響の時がある。情報を送る側のエージェントは、よい情報を厳選して情報を送るべきであると考えられる。これは実際のサッカーでもいえることである。

#### 5. おわりに

今回ロボカップサッカーのシミュレーションにおいて、声を「協調行動の発動タイミングのトリガーとしての使用」と「自分もっている情報を他のエージェントに伝える」という2通りの活用方法による協調行動を提案をした。y 声をトリガーとして使用した場合は、複数のエージェントがほぼ同時期に作戦の発動が行われていることが実験の結果とエージェントのログから確認できた。このことにより協調行動の発動のトリガーとして、声によるメッセージが有効に活用できると言える。二つの声によって自分の持っている情報と伝える、という協調行動の手法はエージェントのパスの選択においては情報を送る側のエージェントが、良い情報だ

けに限定して伝えることにより効果的な協調行動を行うことができた。今後の課題として、2.2章で述べたように他にも声を使っての協調行動手法が考えられるので、これらの具体的な方法を行い実装することが今後の課題である。

#### 参考文献

- [1] 大島真樹: RoboCup サッカー選手プログラム, 森北出版 (2005).
- [2] 高橋友一: RoboCup ではじめるエージェントプログラミング, 共立出版 (2001).
- [3] Peter Stone, Manuela Veloso: Layered Learning and Flexible Teamwork in RoboCup Simulation Agents, Lecture Notes in Computer Science (2000).
- [4] Ali Ajdari Rad, Navid Qaragozlou, Maryam Zaheri: Scenario-Based Teamworking, How to Learn, Create, and Teach Complex Plans, Springer-verlag Berlin Heidelberg (2004).