

5L-7

RoboCup Soccer Simulation における開発者支援ツールの開発

瀧澤 崇[†] 久保 長徳^{††} 高橋 勇[†] 黒岩 丈介[†]
 小高 知宏[‡] 小倉 久和[†]
[†]福井大学 工学部 ^{††}仁愛大学 人間学部 [‡]福井大学 大学院

1 はじめに

RoboCupSimulation では、プレイヤーエージェントを開発する際、単体のエージェントにキックや移動などの基礎的な動作を組み込み、その動作の各種パラメータや発動条件などを制御することでパスなどの複数のエージェントの協調行動を実現する。そのためには1つのエージェントに着目しつつ他のエージェントの動作を確認できるような開発環境が必要となる。しかし2003年に発表された3Dサーバーではこのような環境が十分には整備されていない。本研究ではその基礎的行動をコーディングする際に開発を支援、特に動作確認を支援するツールを開発し、効率よくエージェントの開発を進められることを目指す。

2 RoboCup Simulation League と ログプレイヤー

RoboCup はマルチエージェント系の研究分野の一つとして世界でも多くの研究者が研究を進めている[1]。研究テーマは様々であるが協調問題や、エージェント自身の行動決定の方法などが含まれている。SimulationLeague では、11体対11体の、1体ずつ独立したプログラムによるエージェントによるサッカーを行い、チーム戦術などによって互いに協調をとりながら試合を進めていく。現在、2次元フィールドで試合が行われる2Dリーグと3次元の立体空間のフィールドで試合が行われる3Dリーグが開催されている。大きな違いは、エージェントの視界である。2Dのエージェントの視界は平面であるが3Dでは立体となり、上下方向も考慮する必要がある。本研究では3Dリーグについて扱う。

エージェントをコーディングした後の動作確認の際、通常はSoccerServer(以下SS)に付属のSoccerMonitorを使用し、動いている様子を見て動作確認して新たな修正点を検討、修正していく。しかし、SoccerServerがシミュレーションのために受け取るパラメータが多く、計算に非常に時間がかかり、SoccerMonitorへの情報送信にも時間がかかってしまう。また、SoccerMonitorは3Dで描画されボールの位置などによって視点を変えたりするため、エージェント間の位置関係が把握しにくく、またモニター自身の描画速度によ

ては連続的な動きとはならず、動作と確認を同時に行うとサッカーの連続的な動きを確認しづらくなってしまふ。また、開発を支援する機能も十分には実装されていない。

本研究では1試合毎にSoccerServerにより生成される試合のログに注目する。RoboCupの試合のログにはシミュレーションステップ毎のエージェントの位置・ボールの位置等の情報が含まれる。その情報を用いることで、試合を低負荷で再生することができる。この機能にエージェント開発を支援する機能を追加したログプレイヤーの開発を目指す。

3 ログプレイヤーの設計

効率よく動作確認をするために、おおまかに次のような2つの機能を搭載する。

- ムービープレイヤーとしての基礎機能

エージェント開発時には試合進行のなかで発生したゴールシーンなどの特徴的な出来事に着目する必要がある。そのため、図1に示すようなタイムラインバーを実装する。このバーは試合開始から終了までの時間を表し開発者が着目したい位置をマーキングしたりログに記録されたゴール等のイベントの発生時刻を自動的にマークできるようになっている。また、着目したい部分を容易に繰り返し確認できるようにするために指定した区間を連続再生するA-Bリピート機能を実装する。その他にも利用者が見たい部分を容易に指定して再生できるようにするために一般的なムービープレイヤーに実装されている巻き戻し、早送り、一時停止のような機能も実装する。

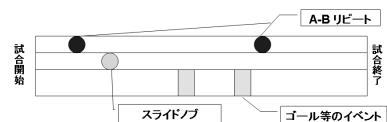


図1: タイムライン

- 試合の見方に関する機能

フィールド全体の様子やエージェント間の位置関係を把握しやすくするために図2のような俯瞰表示機能を実装する。また開発支援機能として、

オブジェクト強調表示と移動軌跡表示を実装する。強調機能により開発時に着目しているエージェントと他のエージェントを区別し個々の動作を確認することを支援する。また、動作の軌跡を見せることにより、各エージェントが当時どのような判断を下したかを考えるヒントを与える。開発者は動作している様子を目で見てそれについて評価する必要があり、開発するログプレイヤーは試合そのものを見やすくするだけではなく動いているエージェント・ボールなどのオブジェクトについてのいくつかの情報が得られるものにする。

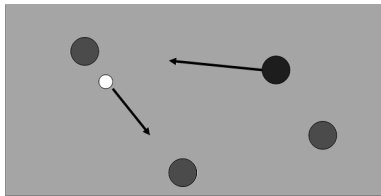


図 2: フィールドの俯瞰表示

4 ログプレイヤーの開発

本研究でのログプレイヤーは、始めにログを読み込み記述されているデータを整理し、それをを用いて1試合分のムービー、及びログプレイヤーの機能を含んだFlash ファイルを生成する形式とする。 ログプレイ

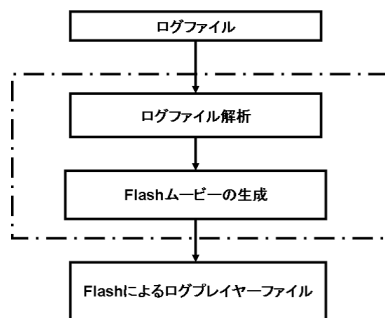


図 3: ログプレイヤーの構造

ヤーの開発にはFlash[2]を用いる。Flashとは動画を扱うための規格の一つで、映像や音声を少ないリソースで表現することが可能である。また再生環境への依存度が低いために様々な環境で再生することができる。さらにActionScriptと呼ばれるスクリプトで制御することによって例えばマウスの動きによってアニメーションさせたりすることができる。一般的に、FlashはインタラクティブなWebサイトを作る場合によく用いられるが、本研究では試合のムービーを生成し、それをスクリプトによって制御するFlashのファイルを生成する。Flashファイルはプレイヤーがあれば再生

できるため、開発者は自分の環境を気にすることなくログを再生し、そこに含まれる様々な機能を用いて動作を確認することができる。さらに、少ないリソースで再生できるので快適な動作で動作の確認ができる。

• ログファイル解析部

今回の使用するログは1行目に試合に関する基本情報が記述され、2行目以降には1行につきシミュレーションステップ1回の各オブジェクトの座標情報が含まれる。解析の際はまず1行目を読み込みそれにより試合のフィールドなどの情報を得る。またその際、ボールの大きさ・エージェントのシミュレーションでの大きさ等も読み込む事によりログプレイヤーでの描画サイズも決定する。二行目以降は1行づつ読み込み、ログファイルのテキストデータを座標などの数値データに変換し、ムービー生成部に渡す。

• ムービー生成部

ムービー生成部ではMingライブラリを用いてFlashファイルを生成する[3]。ログファイル解析部から受け取った座標データによりシミュレーションステップ毎の各座標位置にオブジェクトを配置していく。それを連続したシミュレーションステップで繰り返す事により1つのムービーを完成させる。またActionScriptを用いて、ムービーの動きを制御してさらに前述の機能を付加していく事になる。

5 まとめ

エージェント開発の場合、正しく動作しているかどうかの評価基準は開発者自身の感覚になる場合が多いため、評価が正しく行えているかどうかの判断が難しくなってしまう。以上のことを考慮すると、このようなエージェントの動作評価を支援するようなツールの開発は、RoboCupエージェントの開発において重要である。

参考文献

- [1] ロボカップ日本委員会 監修 高橋友一, 伊藤暢宏 『RoboCupではじめるエージェントプログラミング』 共立出版 2001
- [2] <http://www.macromedia.com/jp>
- [3] <http://ming.sourceforge.net>