

USARSim を使った Mixed Reality Simulator の提案

Proposal of Mixed Reality simulator using USARSim

橋本哲也[†] 清水優[‡] 高橋友一[†]

Tetsuya Hashimoto[†] and Masaru Shimizu[‡] and Tomoichi Takahashi[†]

名城大学[†], 中京大学[‡]

Meijo University[†] Chukyo University[‡]

103430027@ccalumni.meijo-u.ac.jp, shimizu@sist.chukyo-u.ac.jp, ttaka@meijo-u.ac.jp

Abstract

Simulation is one of key technologies to develop robots. In RoboCup Real Robot leagues, various simulation tools have presented to check their algorithms. In the Mixed reality League, a small robot, CITIZEN Micro RobotTM, has been used as a soccer player. The positions of robots are sensed by a global camera and sent to a server program, Soccer Server. A soccer ball is not a physical one but a ball projected and the position of the ball is controlled by the Soccer Server. Mixed reality league has a feature that movements of the robots and the interactions among robots are done in a physical world and the ball is in a computer world.

We propose a simulator that the physical movements of robots are simulated in USARSim that adopts a mathengine KarmaTM. Games of 5 vs 5 are presented and it is shown that in our system a game of CITIZEN Micro RobotTMs with a camera or range sensors is simulated.

1 はじめに

シミュレーションはロボット開発において重要なツールであり、Mixed Reality でも、Fagner らが提案したシミュレータが利用されている。このシミュレータは、ロボットが衝突した際の挙動が詳細に考慮されていない。ロボットの動きはモータのトルクやロボットと地面間の摩擦等によって決まる。更に衝突をする際には、ロ

ボット間の摩擦等も考慮する必要があり、ロボットは複雑な挙動をする。

本論文では、USARSim を利用した Mixed Reality Simulator を提案する。我々が提案する Urban Search And Rescue Simulation(USARSim) を利用した Mixed Reality Simulator は、ロボットやフィールドの質量や材質、摩擦係数等のパラメータを変更する事ができるため、実際の競技環境に合わせる事ができる。更に、競技環境とは異なる環境下でのアルゴリズム検証も行う事ができる。また、センサやカメラ等のデバイスをシミュレーション上で搭載する事ができるため、新規デバイスの実装に関する議論にも利用する事ができる。

2 Mixed Reality System

図 1 は Mixed Reality のシステム構成図である。Mixed Reality では、ロボットに実機を使い、シミュレーションで計算されたボールを用いる。

システムを構成するモジュールは以下の 6 つである。

Soccer Server: ボールの物理シミュレーションとモジュールの中継

Graphics : ボールやロボット、コート of 描画

Operator : 試合開始, 停止管理

Robot Control : 赤外線送信機を制御しロボットへの命令送信

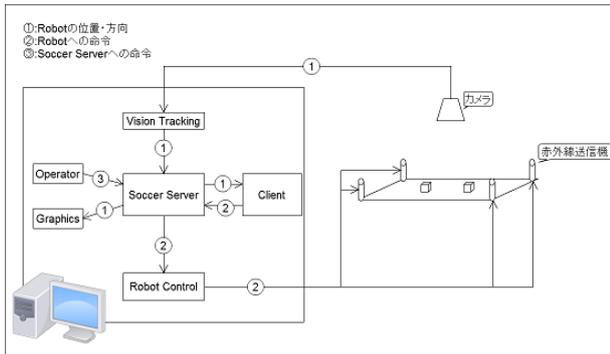


図 1: Mixed Reality システム構成図

Vision Tracking : カメラから送られた画像を処理し、ロボットの位置情報取得

Clients : 各ロボットの動作プログラム

2.1 ロボットの衝突モデル

シミュレータはロボット間の衝突モデルを考慮する必要がある。ロボットは衝突したり、様々な角度で押し合いをする事があり、衝突する際には、ロボット同士の摩擦やタイヤと地面の摩擦等を考慮した複雑な動きになる。

Fagner らはロボットの動きのシミュレーションに独自の Trajectory, Collision モデルを提案した [2]。そのモデルは、簡単な衝突しか考慮されていない。例えば、3 体以上での衝突、ロボット間の摩擦、押し合いをする際のモータの回転等を考慮していない。

我々はロボットの動きのシミュレーションに USARSim を利用する。USARSim は探索・救助ロボットのシミュレータであり、RoboCup Rescue Virtual Robot League で使用されている [1]。図 2 は、USARSim の動作例である。USARSim は物体の動きの計算に 3D 物理エンジンである Karma engine を用いている [4]。Karma engine は、市販のゲーム等の物理エンジンにも使用されており、物体同士の摩擦や物体に加わる力を詳細に計算する。専用のエディタも付属しており、ロボットやマップの作成が容易に行える。

縦線で示したモジュールは従来の Mixed Reality モジュールである。横線で示したモジュールは USARSim のモジュールである。そして線で囲ったモジュールが今回作成したモジュールである。Clients は Soccer Server



図 2: USARSim 動作例

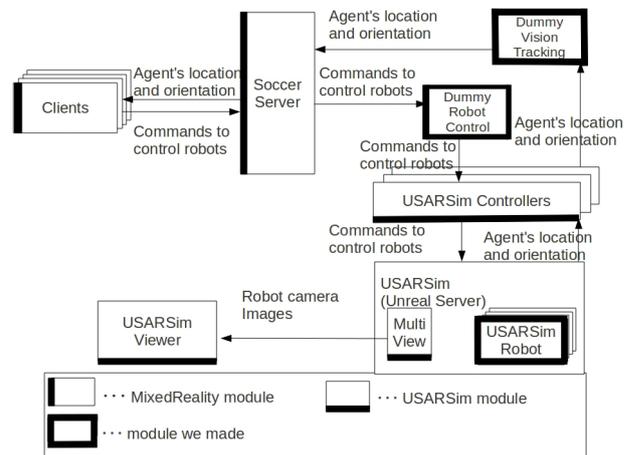


図 3: 実装システム構成図

に対してロボットの制御命令を出したり、Soccer Server からロボットの座標を取得する。Soccer Server は USARSim と接続しロボットの座標を取得したり、Clients からのロボット制御命令を USARSim に送信する。

2.2 モデルの作成

図 4 は Mixed Reality リーグで実際に使用している CITIZEN Micro Robot と、それを元に USARSim 上に作成したモデルを示す。モデルは、本体、ヘッド基板、ホイール 2 個、補助用ホイール 2 個の 6 つのパーツから構成されている。



図 4: CITIZEN Micro Robot 実機 (左) と USARSim 上に作成したモデル (右)

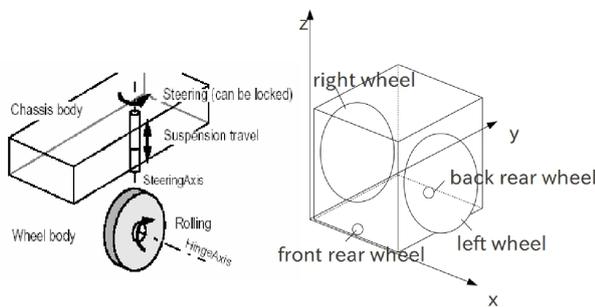


図 5: USARSim での車体と車輪の接続 (左)[3] と実機のパーツ構成図 (右)

USARSim 上で動作するロボットにタイヤ等のパーツを連結するには、次の 2 通りの接続方法がある。

- A car wheel joint : タイヤの接続に使用.2 つの軸に対してパーツを接続.
 - spin axis(HingeAxis) : パーツをスピンさせる
 - steering and suspension axis(SteeringAxis) : 舵をとったり, 軸に対して平行に移動する.
- A hinge joint : パーツの接続に使用.1 つの軸に対してパーツを接続.

今回の実装では、設定ファイル内で SteeringAxis にトルク値を与えて、シミュレーション時にロボットに対し Drive コマンドを送信する事でロボットを制御する事ができる。以下にロボットの定義リスト usarbot.ini の一部を示す。

```
[USARBot.EcoBe]
bDebug=False
Weight=14
Payload=40
ChassisMass=1.000000
```

```
MotorTorque=90.0
bMountByUU=False

JointParts=(
PartName="RightFWheel",
PartClass=class'USARModels.EcoBetire1',
DrawScale3D=(X=1.0,Y=1.0,Z=1.0),
bSteeringLocked=True,
bSuspensionLocked=true,Parent="",
JointClass=class'KCarWheelJoint',
ParentPos=(Y=0.0120,X=-0.0001,Z=-0.01),
ParentAxis=(Z=1.0),
ParentAxis2=(Y=1.0),
SelfPos=(Z=-0.0),
SelfAxis=(Z=1.0),
SelfAxis2=(Y=1.0))
```

```
JointParts=(
PartName="LeftFWheel",
PartClass=class'USARModels.EcoBetire2',
ParentPos=(Y=-0.0120,X=-0.0001,Z=-0.01),
```

2.3 Mixed Reality からの USARSim モデルの制御

この節では、USARSim 上のモデルの制御方法について説明する。Mixed Reality クライアントはロボットの左右のホイールに与えるパラメータを Soccer Server に送信する。Soccer Server は送られた命令を Dummy-RobotControl に送信する。DummyRobotControl は、送られた信号を USARSim コントローラに USARSim でのホイール制御コマンドに変換し、Unreal Server に送信する (図 6)。

ロボットを制御する為に、ロボットに書き込まれたファームウェアにクライアントが Velocities(pr, pl) コマンドを送信する。pr, pl はそれぞれ右, 左のホイールのモータに与えられるパラメータである。pr, pl は 0 から 30 の値をとる。図 7 に実機とシミュレーションそれぞれにおけるパラメータと速度との関係を示す。

提案シミュレータでは Drive() 関数を通して左右のシミュレーションのモータにパラメータを送信する。pr, pl をそのまま送信すると、シミュレーションのロボット走行速度は図 7 に示す様な比例関係となり、実機と一致しない。

そこで、Dummy Robot Control 部で実機のデータを元にシミュレーションのモータに送るパラメータを変換した。pr, pl と変換後の速度との関係を図 8 に示す。pr, pl を f(pr)f(pl) と変換する事で、実機と同等の動きを実装する事ができた。

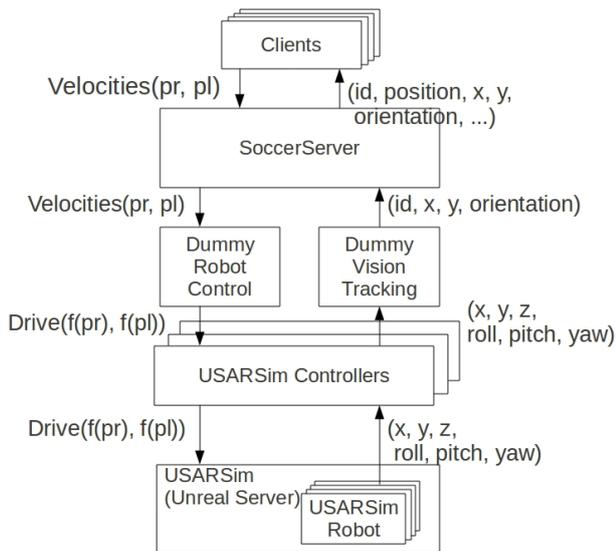


図 6: MR クライアントから USARSim モデルへのコマンド変換

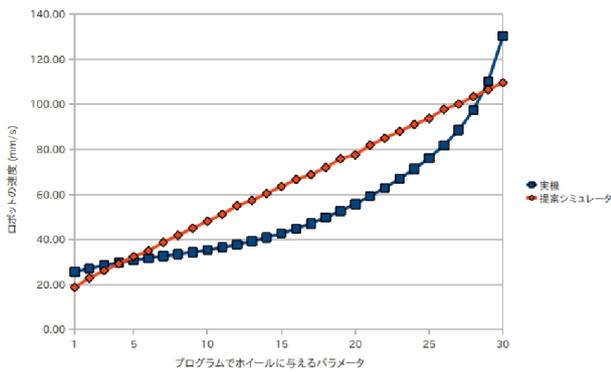


図 7: 関数パラメータと出力速度の対応表 [5]

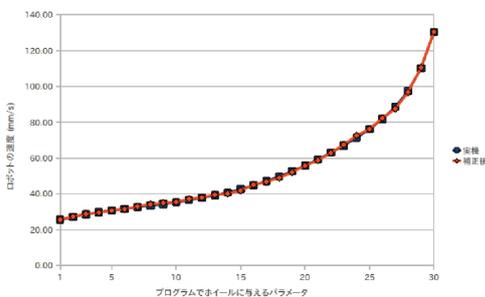


図 8: 補正後の USASim ロボットの出力速度対応表



図 9: USARSim 上で動作した 5 v.s. 5(左サイド blue, 右サイド yellow)

表 1: 試合結果

sample	v.s.	sample	0 - 1
Tatsunootoshigo	v.s.	sample	11 - 0
Tatsunootoshigo	v.s.	Tatsunootoshigo	0 - 0

3 動作実験例

3.1 5 v.s. 5 の動作

これまでの実装を終えた上で、Mixed Reality 5 v.s. 5 ゲームを動作させた。図 9 はフィールド上に 10 台のロボットが並んだ状態を示している。USARSim 上で以下の 3 試合を行った。

- sample v.s. sample
- Tatsunootoshigo v.s. sample
- Tatsunootoshigo v.s. Tatsunootoshigo

sample プログラムは Mixed Reality パッケージに同梱されているものである [6]。Tatsunootoshigo プログラムは RoboCup 2010 Singapore Mixed Reality Demo Event で使用したものである。それぞれの結果を表 1 に示す。

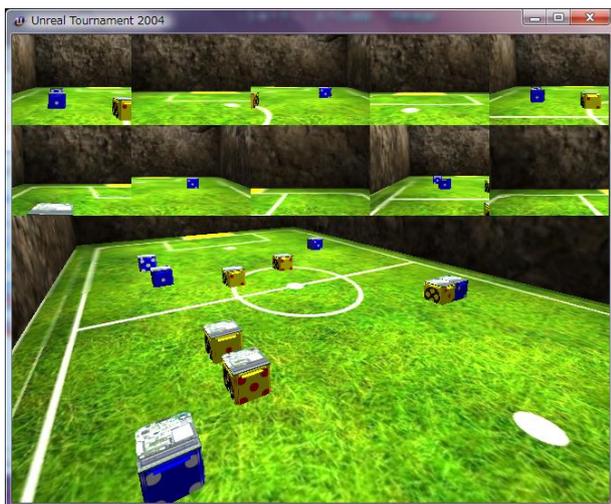


図 10: 各ロボットの搭載カメラからの図 (上部), センサを搭載したロボットの 5 v.s. 5(下部)

3.2 センサ装着例

USARSim にはデフォルトで多くのセンサが定義されている。前述したパーツの接続定義の中でセンサをロボットに接続をし, CITIZEN Micro Robot 各台に, カメラとレンジセンサを搭載した。図 10 は, センサを搭載したロボットを使って 5 v.s. 5 を動作させた時のスクリーンショットである。

カメラ 画像上の 10 個の小さな画像はそれぞれのロボットに搭載したカメラが捉えている映像が個別に表示されている。ロボットは blue チームと yellow チームに分かれている。所属するチームによって, ロボットの車体は青色か黄色になる。サイコロの目状に描かれた丸印の数がロボットの ID を示す。ID は各チーム 1 から 5 まで存在する。フィールドを真上から見た図を図 11 左に示す。カメラの解像度は 208×160 (pixels) である。

レンジセンサ 図 11 右は blue チーム ID2 番のレンジセンサの距離データをプロットした図を示している。また, 図 11 右のセンサプロット図からもまわりにロボットが存在する事が確認できる。表 2 にレンジセンサの仕様を示す。

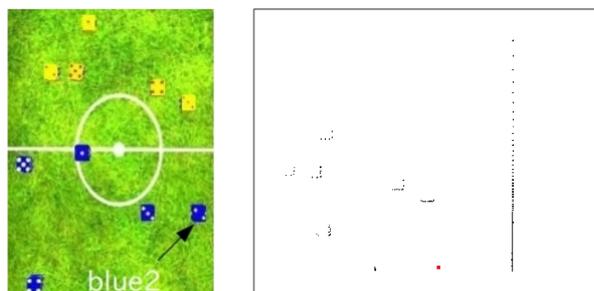


図 11: フィールドを真上から見たロボット配置図 (左), blue2 からのレンジセンサのデータプロット図

表 2: レンジセンサ仕様

max range	49.0[m]
scan interval	0.2[sec]
scan resolution	1[deg]
FOV	180[deg]

4 まとめ

本研究で, USARSim を使ったシミュレータを提案した。提案したシミュレータは, 以下の 2 点において従来のシミュレータより優れている。

- Karma engine の衝突モデルは Fagner らが提案した衝突モデルより詳細な物理計算を行える。
- センサデバイスを CITIZEN Micro Robot に装着する事により, 仮想的に機能を拡張できる。

参考文献

- [1] <http://www.robocuprescue.org/wiki/index.php?title=Virtualrobots>
- [2] Fagner de A.M.Pimentel, et al MR-Simulator:A Simulator for the Mixed Reality competition of RoboCup, RoboCup2010:Robot Soccer World Cup XIV p.82-p.96, RoboCup International Symposium 2010,
- [3] USARSim Manual 3.1.3 p.136, <http://jaist.dl.sourceforge.net/project>

/usarsim/Documentation/3.1.3/USARSim-
manual3.1.3.pdf

- [4] MathEngine Karma User Guide p.54,
<http://udn.epicgames.com/Two/rsrc/Two/KarmaReference/KarmaUserGuide.pdf>
- [5] [http://odin.pcag.fh-wolfenbuettel.de:8080
/mixedreality/index.php/topic,73.msg301.html](http://odin.pcag.fh-wolfenbuettel.de:8080/mixedreality/index.php/topic,73.msg301.html)
- [6] <http://sourceforge.net/projects/pv-league/>