

マリオAI エージェントプログラミング の基礎

馬場 雪乃

2016年10月4日

計算機科学実験及演習2ソフトウェア

ゲームAIのベンチマークソフトウェア

- ゲームAI技術を競う目的で開発された、ベンチマークソフトウェア
- マリオAIの競技会が2009年から2012年まで開催され、ゲームAIの発展に寄与

- デモ：

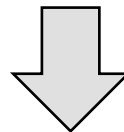
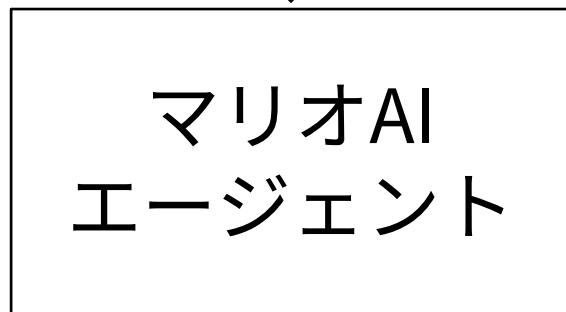
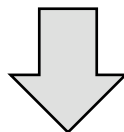
<https://www.youtube.com/watch?v=DlkMs4ZHHr8>



マリオAIエージェントのプログラミング

入力情報を用い、マリオの次の行動を決定し出力する

入力：マリオの状態情報、環境情報等



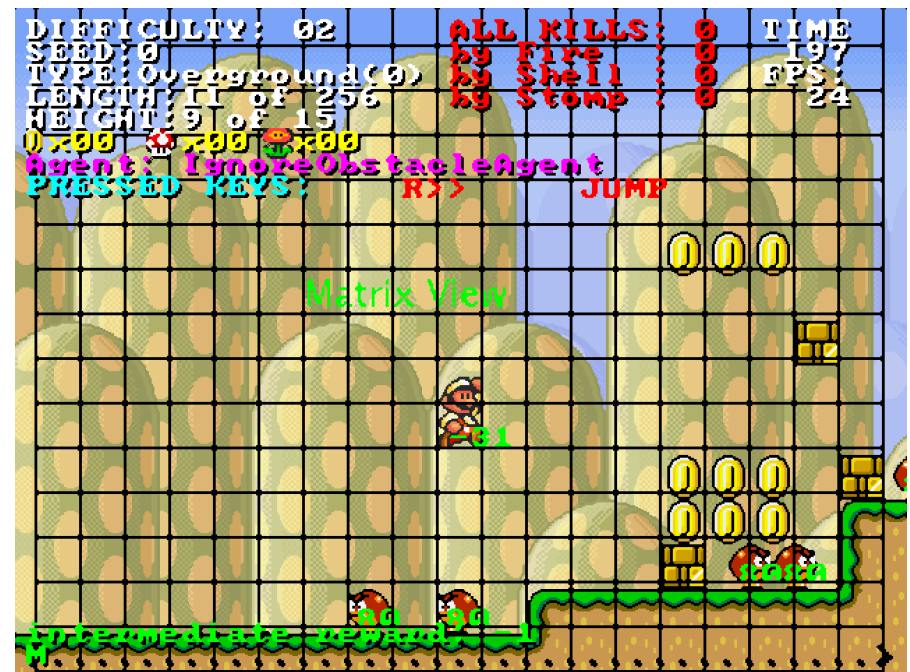
出力：マリオの次の行動



ジャンプ？ しゃがむ？ 走る？

マリオの状態情報や環境情報が与えられる

- マリオの状態情報：
「ジャンプ可能か」「地面にいるか」
「ファイアーを打てるか」等
- 環境情報：
マリオの上下左右9マスの
マップ情報・敵情報



マリオの行動を6次元の配列で出力する

- 行動は、6次元のboolean配列で出力する
- 配列の各要素が各ボタンに対応する
- ボタンを押している=true、ボタンを押していない=false

例：

左移動	右移動	しゃがむ	ジャンプ	ダッシュ& ファイア	上移動
False	True	False	False	True	False

この場合、マリオは右方向にダッシュする

エージェントの例：ForwardJumpingAgent

ひたすら右ダッシュジャンプするエージェント

- reset()メソッド：ゲーム開始時の設定を記述

```
public void reset() {  
    action = new boolean[Environment.numberOfKeys];  
    action[Mario.KEY_RIGHT] = true; ← 「右移動」 「ダッシュ&ファイア」  
    action[Mario.KEY_SPEED] = true;   ボタンを押す  
}
```

- getAction()メソッド：毎ターンの行動を出力

```
public void getAction() {  
    action[Mario.KEY_SPEED] = isMarioAbleToJump || !isMarioOnGround;  
    action[Mario.KEY_JUMP] = isMarioAbleToJump || !isMarioOnGround;  
    return action;  
}
```

↑
「ジャンプ可能」または「空中にいる（地上にいない）」
ときは「ジャンプ」ボタンと「ダッシュ&ファイア」ボタンを押す

ステージパラメータの設定

ステージの設定を自由に変更可能

- 難易度
- 敵の有無
- 障害物（土管、落とし穴、キラー砲台等）の有無



敵なし・障害物なし・難易度0



敵あり・障害物あり・難易度100

自分でいくつかのエージェントを実装する

報告書1 (10月25日締切)	課題1	ステージパラメータの変更と動作を見る、サンプルソースの解説
	課題2	敵のないシーンでのエージェントのプログラム
報告書2 (12月6日締切)	課題3	敵のあるシーンでのエージェントのプログラム
	課題4	高度なシーンでのエージェントのプログラム