

強化学習を用いた初級者と上級者の特性を見つける シューティングゲームAIの作成

Finding Beginner and Advanced Characteristics Using Reinforcement Learning
Creating Shooting Game AI

田中泰輔

指導教員 大島真樹

サレジオ工業高等専門学校 情報工学科 制御情報研究室

キーワード:AI, 弾幕シューティング, 強化学習

1. 研究背景

現在のAIの研究では、より単純化された世界で新しい手法を試すためにゲームを用いて人工知能の研究を数多く行っている。

ゲームAIにおいて、ただ強いだけではなくプレイヤーを楽しませるための研究が行われている。今まで勝率を一定に調整する手法が提案されてきたが、それによって不自然な挙動を起こしてしまうことがある。[1]

できる限りヒューマンエラーが起こりうるものでかつ単純なシューティングゲームを活用することで、より人間らしい行動をとるAIを作成できると考えた。

2. 研究目的

強化学習と影響マップを用いることで、弾幕を避けるためのシューティングゲームのAIプレイヤーを作成する。また、その時の特性を動作から得る。

3. 問題点

人間らしい動きをする弾幕ゲームにおけるAIの研究は、すでにいくつか存在する。[2]

しかし、強化学習において今回使用するQ学習という手法は状態ごとにおいてデータを格納するものであるが、座標と時間、被弾の有無などをすべて考慮すると組み合わせが多くなりすぎてしまい、十分な結果が得られない可能性が高いことが挙げられる。

4. 過去の研究

・勝率の拮抗と手加減の不自然さとの2点の実験について、オセロで実施した先行研究[1]や、人間らしい弾避けのシューティングゲームAIについて[2]の先行研究が過去に行われている。

5. 研究環境

本実験で使用する環境は以下の通りである。

・統合開発環境 Visual Studio Code

→Visual Studioと違い、様々な拡張機能があり、コードを扱うにあたって便利なため。

Unity2021.3.24f1

→ゲームの環境を作成する上で、最も扱いやすいゲーム制作エンジンであり、倍速ができるなどシミュレーションとしての機能も十分であるため。

・使用言語 C#

・OS Windows10(Pro 64bit)

6. 理論

今回の実験では学習方法として、Q学習を扱う。また、3章にて挙げられた組み合わせの問題の対策として、影響マップを使用する。

影響マップは自身を中心 3×3 のブロックを用意し、それぞれのブロックに弾幕が存在するか、存在しないかという2つの状態を記録する。そのため、組み合わせの総数は $2^9 = 512$ 通りである。

Q学習の式は、以下の通りである。

$$Q(s_t, a_t) = (1 - \alpha)Q(s_t, a_t) + \alpha(r + \gamma \max_p Q(s_{t+1}, p))$$

また、それぞれの変数の意味は以下の通りである。

・ s : 現在の状態

・ a : 状態 s で選択する行動

・ t : 現在の時刻

・ $Q(s_t, a_t)$: 状態 s の時に行動 a を選択した場合のQ値

・ $\text{Max}Q(s_{t+1}, a)$: 状態 s_{t+1} の時の最も高いQ値

・ r : 報酬(今回は、選択した行動で被弾した場合に負、避けた場合に正となる)

・ α : 学習率、Q値を更新するときの割合

・ γ : 割引率、将来得られる報酬をどれだけ現在の価値として考慮するかという値

7. 研究方法

シューティングゲームを作成し、それをAIにプレイさせて学習させる。自機と敵を一つずつ用意し、敵からは一定間隔で決まったパターンの弾が発射される。これに自機が当たった時にミスをカウントし、ゲームを初期状態に戻す。この時、初期状態から経過したフレーム数を保存する。作成したシューティングゲームの環境を図1に示す。

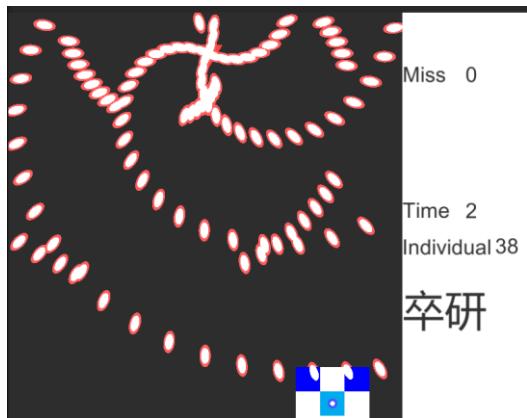


図1. シューティングゲームの実行画像

それぞれのオブジェクトとその役割について、以下に示す。

・自機(図2参照)

AIが操作する媒体。青い丸の内側が当たり判定であり、そこに弾が当たると被弾判定になる。

また、周りの白い四角は影響マップを表すものであり、弾幕が存在するときに青色に変化する。

・敵弾(図3参照)

敵が放つ弾幕。これに自機の当たり判定部分が当たると被弾扱いになる。

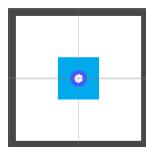


図2. 自機



図3. 敵弾

9. 実験内容

作成した環境を使用して、3000回学習を行ったときのAIの生存フレームを記録する。また、500回ごとに平均をとり、生存フレーム数が増加するかの確認を行う。

8. 実験結果

作成したAIの世代数ごとの生存フレーム数を図4に示す。

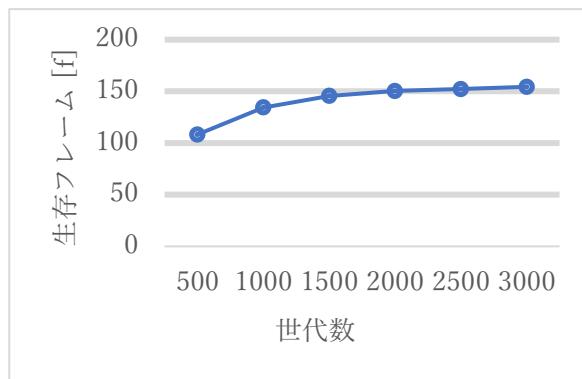


図2. AIエージェントの世代別生存フレーム数

9. 考察

図4の結果より、世代を重ねるごとに生存時間が伸びていることから、AIが学習をし、弾幕を避けることができていると考えられる。また、人にとっては避けにくいと思われる画面端にて弾幕を避けるような動作があったが、これは弾幕の生成点から最も遠い場所であるからであると考えられる。また、画面端にて弾幕を避けるような動きに変わったタイミングは、500から1000の間であった。図4より、この時の生存時間の伸びがかなり大きいため、単純に避けるAIにおいて、弾幕の密度が一番重要な値であると考えられる。

12. 今後の展望

弾幕をAIが避けるようにはなったが、まだ人間らしさを出す工夫ができていないので、これから生物学的制約を導入する必要がある。また、画面中央で弾幕を避けるような動きをとるための工夫を施してエージェントの動作を試す必要がある。

・参考文献

- [1]著者 高木 謙也、藤井 叙人、片寄 晴弘
強さの異なる複数のAIエージェントによるオセロのための自然な棋力調整手法の提案
<https://cir.nii.ac.jp/crid/1390857136564510976>

- [2]著者 佐藤 直之、Sila Temsiririrkkul、Luong Huu Phuc、池田 心

Influence Mapを用いた経路探索による人間らしい弾避けのシューティングゲームAIプレイヤー
https://ipsj.ixsq.nii.ac.jp/ej/?action=pages_view_main&active_action=repository_view_main_item_detail&item_id=175338&item_no=1&page_id=13&block_id=8