5602

縦型シューティングゲームにおける 回避経路計画の自動生成とその評価

Automatic generation and evaluation of avoidance path planning in vertical shooting game

CS09　小澤直弥
指導教員　大島真樹

1. 緒　言

本研究は，縦型シューティングゲームを作成して，操作を自動化したときの生存時間を測定する．自機移動の自動化を行う際に様々な条件を指定して，長い時間生存できるかを比較する．そして，どの条件の生存時間が長かったのかを考察する．

2. 研究のアプローチ

まず元となる縦型シューティングゲーム（以下STG）を作成する．次に配列に各操作のデータが入った入力列を入れることで操作を自動化させる．

例 1：data=6 は右方向に動くプログラムを入れる，と言った感じで複数の data 列を用意する．

例 2：例 1 の配列が 666・・・666 と読み込むと常に右に移動した状態になる．今回は八角形，四角形，三角形の 3 つの動きと読み込む数を変更した複数のパターンを用意して行う．

次に自機が被弾した時に STG を初期化させ，被弾した個所から指定した分だけ戻して，新しく経路を作る（図 1）．これを”ずれ”と呼ぶ．結果が収束するまで（進めなくなるまで）再測定させる．

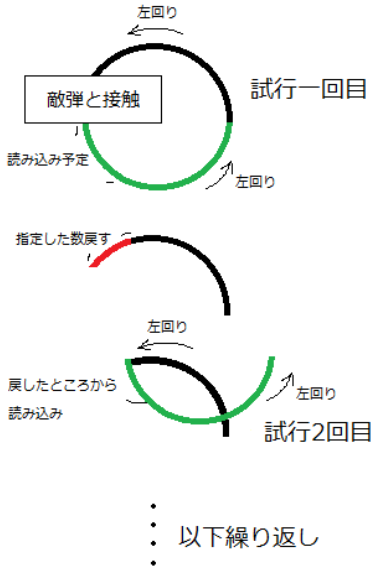

図1　配列を戻して新しく経路を作る方法の例

3. 結　果

図 2 に八角形の動きの時と三角形の動きの時と四角形の動きの時の生存時間を，移動半径大，ずれを 20 ステップで測定した結果を示す．

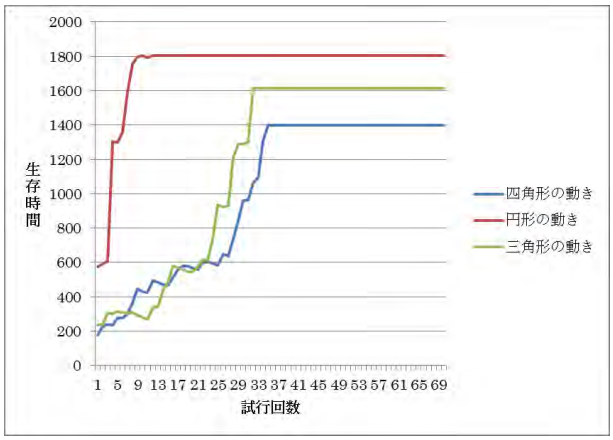

図 2　試行回数に対する 3 つの動きの時の生存時間

4. 結　論

図 2 を見ると 3 つの動きの中では八角形の時の動きが最も生存時間が長く，収束するまでの試行回数が最も少ないことがわかる．逆に四角形の動きが最も生存時間が短く，収束するまでの試行回数が最も多かったので円形に近い動きが有利である．

5. 今後の発展

本実験ではズレを行うときに予めステップ数を指定してあるためにかなり正確な回避経路を生成できるとは限らない．だが STG を AI がプレイしても最終的に人と変わらないような成績を出せることがわかった．

本実験では実装できなかったが，読み込む配列やズレを行う際に予め値を設定しているので，同じ結果が何度も出た場合，ズレの値を変えたり読み込む配列の数を変える・・・といったように可変にすることによって生存時間を伸ばし，クリアを可能にすることも出来るのではないかと考えた．

文　献

[1] Dixq(著)，龍神録プログラミングの館 http://dixq.net/rp/ （閲覧　2013/1/20）

[2] Dixq(著)，新・C 言語～ゲームプログラミングの館～［DX ライブラリ］ http://dixq.net/g/ （閲覧　2013/1/20）