

アメーバからヒントを得た数理モデルを用いた 格闘ゲームAIの提案

吉田 修武*, 石原 誠†, 原田 智弘*, ターウォンマツ ラック* 知能エンターテインメント研究室
* 立命館大学情報理工学部, † 立命館大学院情報理工学研究科人間情報科学コース

I. はじめに

対戦格闘ゲームの AI 国際大会が, CIG¹ で開催されている. この大会は我々の研究室が開発した対戦格闘ゲームのプラットフォーム FightingICE[1] を用い, 出場した AI 同士を総当りで戦わせ最も相手を打ち倒すことができる AI を決めるものである. 近年, 優勝または上位の AI は学習を必要とせず, 予め決められたルールで行動するルールベースの AI が多くを占めた. ルールベースの AI にはアクションのパラメータ変更や, 新しいアクションの追加, 異なるキャラクターを用いた対戦などで性能が変化してしまう問題がある. また, 人間との対戦 (本稿では考慮しない) や相手の行動を予測する機能を搭載した AI に対して次の行動が容易に予測されてしまう.

これらの問題に対して, 2015 年度大会終了後, ルールベースとは違い予め決められた行動ではなく, 敵の行動をランダムに選択したシミュレーション結果から行動を決定するモンテカルロ木探索の AI (MCTSAI) を公式ホームページで公開した [2]. この MCTSAI は 2015 年度大会出場 AI に対して優れた性能を示した. また, 2016 年度大会の優勝または上位の AI は MCTSAI を改良したものが占めた. モンテカルロ木探索は優れた探索方法であるが, AI 研究として他の探索手法との比較が必要となる. よって, MCTS が優れた性能を示した多腕バンディット問題において, 同様に優れた性能を示した綱引きモデル (Tug-of-war) [3] に著者らは着目する.

このモデルは, 青野氏らが真正粘菌アメーバが光刺激を避ける行動からヒントを得て提案した. 生物の振る舞いを正確に再現するものではなく, 特定の問題を効果的に解くための計算手法である. しかし, これは多腕バンディット問題のための数理モデルであるため, 格闘ゲーム用に改変する必要がある. 本稿の目的は対戦格闘ゲームの AI に綱引きモデルを適用し, その性能を評価することである. まず, 綱引きモデルを適用させた AI (TOWAI) を開発する. 次に, MCTSAI と対戦させ, その結果から性能を評価する.

II. 提案手法

A. 対戦格闘ゲームにおける綱引きモデル

ここでは, 対戦格闘ゲーム用に改変した綱引きモデルを説明する. アメーバの中心をハブノードとし, そのハブノードから N 個の分岐が接続されてネットワークを構築する. アメーバの特徴として, 体積が一定のため, 分岐の増減がハブノードにも影響を与える. 時刻 t に, ある分岐 $i \in I$ において x_i^t はハブノードと分岐の境界からの体積の変位を表している. $I = \{1, 2, \dots, N\}$ とし, それぞれにアクションが

設定されていると考える. また, x_0^t はハブノードの変位とする. x_i^t が正の値である場合, その分岐が選択されたと考える. この時, ステップ関数 $\theta(x)$ を用いる.

$$\theta(x) = \begin{cases} 1 & (x > 0) \\ 0 & (\text{otherwise}) \end{cases} \quad (1)$$

つまり, x_i^t が正の時, i 番目の腕のアクションをシミュレートする. シミュレーションにおいて, 相手 AI の行動はランダムに選択し, シミュレーション時間は T_{sim} 行う. また, 本稿で扱う対戦格闘ゲームはリアルタイムゲームであり, 更に対戦相手の行動によって状態が変わるため, 最初のシミュレーションの結果によって最良の手を時間内に見つけることができない可能性がある. よって, 本稿で用いる TOWAI は分岐のシミュレーションの回数がある定数 S_{min} 以下の場合においてもシミュレートする. これは, 有望な手が 1 回目のシミュレーションで運悪く評価値が低かった場合, 他の手が悪い評価をされない限り, 再びシミュレーションされるまでに時間がかかると. その結果, 十分なシミュレーションが行えず, 正しい評価が得られない状態を防ぐためである.

次に, 分岐 i がシミュレーションされた時, 以下の条件で光刺激を与える.

- 自身が与えたダメージ量 ($myScore$) が 0 だった場合
- 相手に与えられたダメージ量 ($oppScore$) が $myScore$ より大きかった場合

これは罰として扱われる. 報酬と違い, 好ましくない選択を抑制する. 以下の式で更新される.

$$l_i^t = \begin{cases} 1 & (\theta(x_i^t) = 1 \text{ かつ 前述のいずれかの条件}) \\ -1 & (\text{otherwise}) \end{cases} \quad (2)$$

また, 体積の変位 x_i^t は [3] の通りに以下の式で更新される.

$$x_i^{t+1} = \begin{cases} x_i^t + v_i^t & (i \in I) \\ x_0^t - \sum_{j=1}^N v_j^t & (i = 0) \end{cases} \quad (3)$$

$$v_i^t = v_i^{t-1} + a_i^t \quad (4)$$

ここで, v_i^t は x_i^t の分岐 i における体積の変位速度を表し, a_i^t は加速度を表す. 式 (3) において, アメーバの体積が保存されるのと同様に $\sum_{i=0}^N x_i^t = \sum_{i=0}^N x_i^0 = C$ (C は定数) が保持される. 即ち, 総体積は初期状態から変化しないということである. また, 加速度 a_i^t は格闘ゲーム AI のために Table. I と以下の式を用いて決定される.

$$Score = \begin{cases} 0.1 & (V_i^t = 0) \\ \min(1, myScore/a) & (l_i^t = -1) \\ \max(-1, oppScore/a) & (l_i^t = 1) \end{cases} \quad (5)$$

¹ <http://www.ieee-cig.org/>

TABLE I
加速度 a_i^t

	$s_i^t > 0$	$s_i^t = 0$	$s_i^t < 0$
$l = -1$	Score	Score	0
$l = 1$	0	-Score	-Score

ただし, $-1 \leq \text{Score} \leq 1$, a は定数とし, V_i^t は時刻 t までの分岐 i の訪問回数とする. s_i^t は分岐 i に割り当てられた体積の偏差を示し, [3] の通りに以下の式で更新される.

$$s_i^t = x_0^t + q_i^{t-1} - \text{mean}_{j \in I \setminus \{i\}} \{q_j^{t-1}\} \quad (6)$$

$$q_i^t = q_0^{t-1} + \mu \cdot (\rho_i^t + \omega \cdot \sum_{j \in I \setminus \{i\}} \pi_j^t) \quad (7)$$

$$\rho_i^t = \theta(x_i^t) - \theta(l_i^t) \quad (8)$$

$$\pi_i^t = \theta(l_i^t) \quad (9)$$

s_i^t は腕を延ばす基準として用いられており, ハブノードの体積 x_0^t と i 番目の腕が他の腕よりも有望であるかを判断する項で構成されている. q_i^t は t 時刻までの光刺激などの情報を基に算出された腕の評価値である. パラメーター μ は式 (6) の x_0^t とのバランスを調整するものであり, パラメーター ω は他の腕の影響度を調整するものである. 式 (8) はシミュレーションが行われ, 光刺激が与えられたかどうかを判断する式であり, 式 (9) は光刺激が与えられたかを判断する式である. TOWAI では, 以上の式の更新とシミュレーションを制限時間内に繰り返し行い, 最終的に最も x_i^t が大きい腕を選択する.

III. 実験

A. FightingICE

FightingICE は 2D の対戦格闘ゲームであり, 対戦格闘ゲーム AI の性能を競う国際大会² のプラットフォームに使用されている [1]. 1 試合が 1 ラウンド 60 秒の 3 ラウンドで構成されている. 各キャラクターの HP の初期値は 0 に設定されており, 攻撃を受けると際限なく減少していく. そして 60 秒が経過した時点で次のラウンドに移行し, HP が 0 にリセットされる. 各ラウンド終了時のお互いの HP から, 式 (10) によってスコアがラウンドごとに算出される.

$$\text{Score}_{my} = \frac{HP_{opp}}{HP_{my} + HP_{opp}} \times 1000 \quad (10)$$

HP_{my} , HP_{opp} はそれぞれ自分, 相手 AI の HP である. 同式より, スコアの上限は 1000 である. 自分よりも相手の HP の方が小さい場合, 自分の獲得スコアは 500 より大きくなる. 反対に, 相手の HP の方が大きい場合, 500 より小さくなる.

各 AI は, 1P (ステージの左側) と 2P (ステージの右側) の 2 つの初期サイドについてそれぞれ対戦を行う. 本実験では, FightingICE version 2.10 を使用して行う.

²<http://www.ice.ci.ritsumeai.ac.jp/ftgaic/>

TABLE II
THE PARAMETERS USED IN THE EXPERIMENTS

Notations	Meanings	Values
μ	バランスの調整	8
ω	影響度の調整	5
a	スコアの調整	20
S_{min}	シミュレーション回数の閾値	3
T_{sim}	シミュレーション時間	60 frames

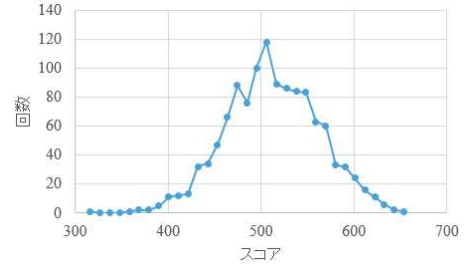


Fig. 1. MCTS AI に対する TOWAI の 1200 ラウンドの獲得スコアの分布

B. 実験手法

モンテカルロ木探索を用いる AI (MCTS AI) と, 綱引きモデルを用いる AI (TOWAI) で対戦を行う. MCTS AI と TOWAI で 1P 側で 200 試合, 2P 側で 200 試合の計 400 試合 1200 ラウンド行う. その後, 獲得スコアによって提案 AI の性能を評価する. AI に用いたパラメータを Table. II に示す. これらのパラメータは, 予備実験を行い最適なものを使用している.

IV. 結果と考察

MCTS AI と TOWAI との 1200 ラウンドにおける TOWAI の獲得スコアを Fig. 1 に示す. 同図において, 横軸はスコアを表し, 縦軸はその回数を表す. スコア平均 507, ラウンド勝利数 654, 勝率 55% となった. この結果から TOWAI を適用した AI は, リアルタイムゲームである格闘ゲーム FightingICE において優れた性能を示すことができた.

TOWAI は現在, モンテカルロ木探索のような木探索などにある数アクション先まで考慮する構造がないため, シミュレーションにおいて 1 アクションしか考慮することができない. この構造をうまく追加することができれば, より深い探索が求められる AI と対戦する場合に優れた性能を示すことができると考えられる. これを今後の課題として研究する予定である.

REFERENCES

- [1] F. Lu et al. *Fighting Game Artificial Intelligence Competition Platform*, Proc of. IEEE 2nd Global Conference on Consumer Electronics (GCCE 2013), pp. 320-323, 2013.
- [2] S. Yoshida et al. *Application of Monte-Carlo tree search in a fight-ing game AI*, Proc of. IEEE 5th Global Conference on Consumer Electronics (GCCE 2015), pp. 623-624, 2016.
- [3] M. Aono et al. *Amoeba-inspired Tug-of-War algorithms for exploration-exploitation dilemma in extended Bandit Problem*, BioSystems, 117, pp. 1-9, 2014.