

GA を用いた NPC の多様な振舞いの生成

福嶋 良平^{†1,a)} 藤井 叙人^{†1,b)} 片寄 晴弘^{1,†1,c)}

概要: ビデオゲームに登場する NPC の振舞いはプレイヤーの印象に大きく影響する。振舞いの設計はゲーム開発者にとって重要な課題であるが、ゲームの規模が大きくなると、手作業による設計は困難となる。本論文ではその一つの解決手段として、GA を用いた NPC の多様な振舞いの生成方法を提案する。ここでは、人間プレイヤーのプレイスタイルに関わる特徴量を導入しその重みの組み合わせを多様化する。この提案に基づいた 2 つの実験の結果、生成される NPC の振舞いに多様性が表出することがわかった。

Generating NPC's various behaviors using GA

FUKUSHIMA RYOHEI^{†1,a)} FUJII NOBUTO^{†1,b)} KATAYOSE HARUHIRO^{1,†1,c)}

Abstract: The behaviors of non-player characters (NPCs) in video games greatly influence the player's impressions of those games. Designing behaviors is one of the most crucial tasks for game developers. The programming required for NPC behavior has become time-consuming because of the increasing demand for massive video games. In this study, we propose generating NPC's various behaviors by using the genetic algorithm (GA). Attention is paid to identifying a human player's play-style, and feature quantities related to play-style are introduced to solve the problem. The results of two experiments based on this proposal shown that diversity of NPC's behavior is exposed by the introduction of the feature parameters related to play-style and reproduction of the variation in the weight of the feature parameters.

1. 初めに

ビデオゲームでは、敵キャラクタやプレイヤーを含む仲間のキャラクタの役割など、NPC に数多くの役割が与えられている。NPC の行動や人間プレイヤーに対する反応はゲーム AI と呼ばれる思考ルーチンによって決定され、ゲームのプレイの印象に大きく影響を与えるため、ゲーム AI の調整はゲーム開発における重要な課題である。近年のビデオゲームはその規模の拡大傾向にあり、登場する NPC の種類が増加している。そのため、ゲーム AI の設定を全て人の手によって行うには時間的コストを費やしてしまう。この問題を解決するため、ゲーム AI の自動生成に関する

技術開発が進められている。

ゲーム AI に関する技術開発として、強いゲーム AI を自動生成する研究が盛んにおこなわれている。特に、将棋やオセロといった二人零和完全情報ゲームにおける研究は人間プレイヤーよりも強くなることを実現している [1], [2]。人間プレイヤーよりもゲーム AI が十分に強くなった今、強さに関する研究はゲーム AI 同士の強さを競うまで発展している [3], [4]。ゲームに関する研究では、より多くのジャンルのゲームへと研究分野を広げている [5], [6], [7]。近年では、人間らしいゲーム AI の生成に関する研究に焦点が当てられている。この分野の研究では、人間の強さに適応したゲーム AI の生成 [8] や、ゲーム AI により決定される行動に生物学的な制約を導入する手法 [9]、人間プレイヤーの振舞いを模倣させる手法 [10] など、人間らしさを実現するために様々なアプローチが行われている。これらの研究は、人間プレイヤーに適応するようにゲーム AI の振舞いを生成する事を目的としている。しかし、ここでは提案されるゲーム AI の行動の多様性については言及されていない。

¹ 情報処理学会

IPSJ, Chiyoda, Tokyo 101-0062, Japan

^{†1} 現在、関西学院大学理工学研究科人間システム工学専攻

Presently with Department of Human System Interaction, Graduate School of Science and Technology, Kwansai Gakuin University, Sanda, Hyogo 669-1337, Japan

a) erv95460@kwansai.ac.jp

b) nobuto@kwansai.ac.jp

c) katayose@kwansai.ac.jp

ビデオゲームに登場する NPC は、概ね敵キャラクタとパートナーとなる NPC（以下 PNPC）に分類される。近年ではオンラインゲームが普及したことにより、NPC が人間プレイヤーを代行する機会が増加し、PNPC の振舞いの人間らしさが要求されている。また、人間プレイヤーは個々が異なるプレイスタイルを有するため、個々の PNPC の振舞いにも個性があることが望まれる。個々の振舞いが異なる PNPC を実現するため、本稿では遺伝的アルゴリズム (GA) を用いた手法を提案する。GA は進化的アルゴリズムの一つであり、探索的に最適解を得るためには有用であるが導出される解が一意に定まる特性があり、多様な解を得ることができない。本研究では、PNPC が様々な振舞いを持って自動生成されることに着目しており、我々はそのための GA の活用方法について提案していく。提案手法の有用性を検証するためには、ターン制のゲームであり、登場する NPC が取りうる行動が多数存在するローグライクゲームを用いる。生成されるゲーム AI のプレイログの類似度の計測と、被験者による主観評価により PNPC の振舞いの多様性を検証する。

2. 関連研究

池田らの研究 [8] では、人間プレイヤーの強さに適したゲーム AI の生成がオセロに対して行われている。この研究は GA の適応度関数に特徴量の重みのばらつきに高い評価を与える項を導入している。このことにより、人間プレイヤーに適応しつつ振舞いが一意に定まらないゲーム AI の生成に成功している。

藤井らの研究 [9] では、ゲーム AI が人間らしい行動を獲得するために、機械学習へ生物学的制約を導入する効果を調べている。この検証の結果、躊躇いや間違いなどの人間らしい行動が先験的な知識なしにゲーム AI に見られたことが報告されている。

NPC の様々な振舞いを獲得することに関しては、GA を用いた手法が提案され、FPS ゲームにおいて検証されている [11]。ここでは、事前に NPC の行動に関する特徴を二つのグループに分類し、それらのグループに基づいて学習を進めることで NPC の振舞いに違いを生み出すことを試みている。この手法を用いてより多くの分類のゲーム AI を生成するためには、行動に関する特徴をより多くの分類に分ける必要がある。

我々の研究の目的は、ゲーム AI の振舞いが自動的に多数生成される事であり、人間プレイヤーがそれらの個性を見出すことができることが重要である。この目的のために、典型的なターン性のゲームの一種であるローグライクゲームにおいて、[8] で提案されている、生成されるゲーム AI が有する特徴量の重みのばらつきに高い評価を与える項を GA に導入することによる PNPC の振舞いの多様性の導出可能性を調査する。

表 1 プレイヤキャラクタに設定される代表的なパラメータ
Table 1 Shows parameter configuration of player character.

パラメータ	備考
レベル	キャラクタの強さ
HP	0 になるとゲーム終了となる
攻撃力	ダメージ計算に関与する
防御力	ダメージ計算に関与する
経験値	レベルの変動に関与

3. ローグライクゲーム

ローグライクゲームは探索型 RPG として長く存在しているゲームジャンルであり、1980 年に発売されたコンピュータゲームの ROGUE を基に発展したゲームである。このゲームジャンルはターン制 RPG に分類される。プレイヤーは一人のキャラクタを操作し、ダンジョンと呼ばれるマップを探索する。このジャンルのゲームの最終的な目的はダンジョンの最も深い階層まで探索をすることである。

ゲーム内のターンの初めには、プレイキャラクタは 8 方向への移動か攻撃のうち、いずれかの行動をとることができる。続いて、すべての敵はプレイキャラクタと同様の行動を一度ずつ行うことができる。ターンはプレイヤーのターンと敵のターンから構成され、このプロセスはプレイヤーがダンジョンの探索を終了するまで継続される。プレイヤーが探索するダンジョンは深さの概念を持っており、その形状をランダムに生成される。プレイヤーがダンジョン内の階層を通過することでより深い階層へと移動することができるが、敵はダンジョンの深さに応じて徐々に強くなる。

表 1 はプレイヤキャラクタに設定される代表的なパラメータを示す。レベルはプレイキャラクタの強さを示し、攻撃力と守備力に関係する。攻撃力と防御力はダメージを計算する際に用いられる。敵を倒すことで経験値を獲得することができる。プレイキャラクタのレベルを増加することができる。パラメータが高いことはプレイキャラクタが強いことを意味し、キャラクタのレベルを上げることはローグライクゲームにとっては重要である。

4. アプローチ

MinMax 法は木探索アルゴリズムの一つであり、将棋やオセロなど二人零和完全情報ゲーム等のゲーム AI として利用されるアルゴリズムである。このアルゴリズムは木探索を行うため、ゲームの進行を木構造として表現できるものには適応が可能である。ローグライクゲームはターン制のゲームであり、その進行を木構造で表現できるため、ゲーム AI の基本木構造として MinMax 法を利用する。この手法は子ノードの値を含む、遷移可能な各ノードの値を評価することにより最適なノードを選択する。本稿ではノードを評価するための関数を局面評価関数とする。局面

評価関数は PNPC の HP などのゲーム状態を表す特徴量と重みを掛け合わせたものの線型和として定義する。特徴量の種類が多い場合、網羅的な方法で各特徴量に対する重みの組み合わせを検証することは現実的ではない。進化的アルゴリズムの一つである GA は、そのような計算コストを削減するために採用されており、協力的なゲーム AI を作成する事に関しては、Chin Kim らの研究 [6] によって有用であることが示されている。本研究では、特徴量の重みがばらついていることに高い評価を与える項 [8] を GA の適応度関数に採用することで、GA を用いて PNPC の多様な振舞いを獲得する可能性を検討する。

一般的に GA を使用した場合、最も個体が集中しているクラスターから一つの個体が最適解として選択される。対照的に、本研究で導入する適応度関数は導出される解が一意に定まることを防止する本質的な性質を有している。従って、このアプローチによる PNPC の振舞いは、確率的ではなく決定的な方法でランダムに選択された個々の特徴量の重みに基づいて決定される。

今回実装する GA では佐藤らによって提案された MGG モデル [12] を世代交代モデルとして利用している。PNPC の振舞いに多様性を実現するために、提案手法ではプレイスタイルに着目し、適応度関数を用いてプレイスタイルに関する特徴量の重みがばらつくように動的に調整する。特徴量の重みは実数値であるため、ESHELMAN によって提唱された BLX- α 法を交叉法として採用する [13]。

提案手法を検証するためにはゲーム内の内部変数へアクセスすることができる必要があるため、エニックスにより発売されたトルネコの大冒険を基とした独自のローグライクゲームを開発する。独自に開発するローグライクゲームに対し GA を実装し、生成されるゲーム AI のゲームの到達率、すなわちゲーム AI の強さによって適応度関数の影響を調査する。続いて、PNPC の振舞いの多様性を、生成されるゲーム AI のプレイログのコサイン類似度の計測による数学的なアプローチとアンケートを用いた被験者実験による主観的なアプローチによって検証する。

5. 本研究で用いるローグライクゲームのデザイン

提案手法の検証に用いるゲームではダンジョンの深さは 10 階層とする。ダンジョンの形状は矩形の部屋と部屋をつなぐ直線の通路で構成し、ダンジョン内に登場する敵キャラクターのパラメータはトルネコの大冒険の NPC に持ちづいて調整する。敵キャラクターのゲーム AI は一種類に統一し、どの敵キャラクターも同じ環境下では同じ行動をとるようにする。敵キャラクターはプレイキャラクターとの距離が一定範囲内である場合、プレイキャラクターまでの最短距離を移動し、プレイキャラクターに隣接しているときはプレイキャラクターを攻撃する。それ以外の場合、敵キャラク

タは 8 方向にランダムに移動する。

一般的なローグライクゲームでは、ダンジョンの形状やキャラクターの初期配置はランダムに決定されるが、生成されるゲーム AI 間の振舞いの類似度を測定するために、同じ状況から行動ログを取得する必要がある。開発したローグライクゲームでは、乱数のシードを指定することで、完全に同じダンジョンを生成する機能を持つ。

5.1 本研究で用いるローグライクゲームの特徴量

ローグライクゲームでは敵キャラクターを倒しながらダンジョンを探索する必要がある。このゲームの敗北条件は操作キャラクターの体力が 0 となることであるため、人間プレイヤーには操作キャラクターの体力を維持しながら敵キャラクターを攻撃するような効率的な行動が要求される。効率的な行動を要求される一方、人間プレイヤーはそれぞれのプレイスタイルを有しており、同じ局面であっても個々の行動選択に差異が生じる。このプレイスタイルをゲーム局面の特徴として定義することで提案手法で生成する PNPC の振舞いに個々の差異を再現する。今回定義する特徴は以下とし、ゲームを効率的に進行するためのものとプレイスタイルに関するものに分類する。

- PNPC の立ち位置。
- PNPC の残り HP の割合。
- 敵キャラクターの残り HP の割合。
- PNPC に隣接する敵の数。
- ダンジョンの各階層への PNPC の推奨レベル。

プレイキャラクターの残り HP の割合はゲームの勝敗に関わり、敵キャラクターの HP の割合は敵にどれだけ攻撃したかを表す。また、ダンジョンの各階層への PNPC の推奨レベルは階層を移動するタイミングを決定するために必要となる。これらは PNPC の行動を効率的にするために必要な特徴である。PNPC の立ち位置とは、PNPC の周囲のダンジョンの形状を周囲 8 マスから判断する特徴であり、立ち位置は、壁側・通路内・それ以外の 3 種類に分類する。PNPC に隣接する敵の数はどれだけ PNPC が敵に囲まれているかを示す。これらの特徴は各ターンでの PNPC の周囲の環境を示す。このような特徴は、プレイヤーごとに好み異なると考えられるため、PNPC のプレイスタイルに関する特徴とする。

ローグライクゲームにおける特徴量は、局面の状況に応じて実数値を取るものとし、ゲーム中の各局面での評価値を算定する。以下で定義した特徴量の詳細を示す。

- $v_{m1} = \begin{cases} 0: \text{The PNPC が通路にいない時} \\ 1: \text{The PNPC が通路にいる時} \end{cases}$
- $v_{m2} = \begin{cases} 0: \text{The PNPC が壁際 to いない時} \\ 1: \text{The PNPC が壁際 to いるとき} \end{cases}$

- $v_{m_3} = \begin{cases} 0: v_{m_1}, v_{m_2} \text{ のどちらかが } 1 \text{ の時} \\ 1: v_{m_1}, v_{m_2} \text{ のどちらも } 0 \text{ の時} \end{cases}$
- $v_{ch} \in [0.0, 1.0]$: NPC の残り HP の割合
- $v_{eh_n} \in [0.0, 1.0]$: 敵キャラクターの残り HP の割合
- $v_{en} \in [0, 8]$: NPC に隣接する敵の数
- $v_{lv_{floor}}$: ダンジョンの各階層への NPC の推奨レベル
 $v_{m_1}, v_{m_2}, v_{m_3}$ は NPC の立ち位置を表し、いずれかが必ず 1 となり同時に 2 つ以上が 1 となることはない。
 $v_{lv_{floor}}$ 以外の特微量には重み w を設定した。 w は 1 から 1000 の範囲の値をとる。これらの特微量と各特微量に対する重みを用いて局面評価関数を定義する。重みの組み合わせを多様化することで、同じ局面でも NPC が導出する行動選択に違いが生じる。

5.2 ローグライクゲームにおける局面評価関数

ローグライクゲームにおける局面評価関数は各特微量とその特微量に対する重みによって算定される 5 つの項で構成する (式 1)。

$$V = V_m + V_{ch} + V_{eh_n} + V_{en} \quad (1)$$

NPC の立ち位置を評価する項を式 2 で表す。 $v_{m_1}, v_{m_2}, v_{m_3}$ の 3 つの特微量のうちいずれかが 1 となるため、NPC の立ち位置によって重みに応じた評価を得ることができる。

$$V_m = \sum_{k=1}^3 w_{m_k} v_{m_k} \quad (2)$$

NPC の残り HP を評価する項を式 3 で表し、残り HP が多いほど高い評価を与える。

$$V_{ch} = w_{ch} v_{ch} \quad (3)$$

敵キャラクターの残り HP を評価する項を式 4 で表す。ダンジョン上にいるすべての敵キャラクターの残り HP の割合の合計が評価として用いられ、敵キャラクターの残り HP の割合が少ないほど評価が高くなる。

$$V_{en} = w_{eh} \frac{1}{1 + \sum_j v_{eh_j}} \quad (4)$$

NPC に隣接する敵キャラクターの数を評価する項を式 5 で表す。隣接する敵キャラクターの数が少ないほど高い評価を意味する。

$$V_{eh_n} = w_{en} \frac{1}{v_{en} + 1} \quad (5)$$

NPC の HP が 0 となった場合はゲームに敗北するため局面の評価値は 0 とした。NPC が階層の移動を行う場

表 2 GA の構成

Table 2 The construction of GA.

世代交代モデル	改良した MGG モデル
交叉法	BLX- α
α	0.5
遺伝子情報の数	5 つ
各要素の初期値	450-550
各要素の値の範囲	1-1000
個体数	30
子個体数	20
最大シミュレーション回数	500 回
世代交代の回数	30 回

合、それまで配置されていたダンジョンは破棄され、新しいダンジョンが生成される。加えて、敵キャラクターも再生成され、各キャラクターの配置も再決定されるため、式 1 を用いた局面の評価が行えない。NPC が階層の移動を行った場合は式 6 を用いて階層の移動が妥当であったかどうかを評価する。

$$V = (Plv - v_{lv_{Pf}} + 1) * 10000 \quad (6)$$

Plv はプレイキャラクターが階層の移動を行った時点のレベルを表し、 Pf はプレイキャラクターが階層の移動を行った時点でのダンジョンの深さを表す。階層ごとに設定された推奨レベルより NPC のレベルが高い場合は高い評価を与える。NPC のレベルが推奨レベルに達していない場合は、評価は負の値になる。

6. GA について

提案システムでは特微量の重みを調整するために GA を用いる。世代交代の際には NPC の立ち位置に関する特微量の重みがばらつくように動的に調整される。この調整は人間プレイヤーに見られるプレイの好みを再現するために適用する。GA における個体の取捨選択は以下の 2 つの項目に着目する。

- 親個体・子個体すべての個体間の立ち位置に関する特微量の重みのばらつき
- ゲームの達成率

表 2 に今回用いた遺伝的アルゴリズムの詳細を示す。GA の世代交代モデルには MGG モデルを改良したものをを用いる。交叉法には BLX- α 法を用いる。GA では個体に $w_{m_1}, w_{m_2}, w_{m_3}, w_{ch}, w_{eh}, w_{en}$ の 6 種類の特微量の重みを遺伝子情報として設定する。これらが、 $\{10, 10, 10, 10, 10, 10\}$ と $\{100, 100, 100, 100, 100, 100\}$ のように実質的に同じ意味を持たないように w_{ch} を 500 に固定した。

6.1 世代交代モデルの詳細

今回世代交代モデルの基礎とした MGG モデルは、親個体の中からランダムに非復元抽出された個体を用いて交叉

を行い、生成された個体から最良の個体1つとランダムに1つを選択する手法である。MGGモデルでは親個体に含まれている個体と子個体数の関係性が考慮されていない。しかし、提案手法ではすべての個体間の特徴量のばらつきが個体選択の評価となるため、子個体から次世代の個体を選択する際には親個体を考慮する必要がある。そのため、次世代の個体を選択する際に、1つ目の最良の個体を選択した後に個体群の構成を修正し、再び適応度を計算したうえで2つ目の個体を選択した。以下に改良した世代交代モデルの手順を示す。

- (1) n 個の個体群をランダムに生成する
- (2) 個体群から親となる個体をランダムに2つ選択する
- (3) 親個体として選択された個体を個体群から削除する
- (4) 選ばれた親個体間で交叉を行い、次世代の個体の候補となる子個体を c 個生成する
- (5) 親個体と子個体 ($2+c$ 個) すべての適応度を求める
- (6) 親個体と子個体 ($2+c$ 個) の中で最良の適応度の個体を1つ選択する
- (7) 選択された個体を個体群へ戻す
- (8) 残った親個体と子個体 ($(2+c)-1$ 個) すべての適応度を求める
- (9) 残った親個体と子個体 ($(2+c)-1$ 個) の中で最良の適応度の個体を1つ選択す
- (10) 選択された個体を個体群へ入れる
- (11) 一定回数行うまで2に戻る

6.2 シミュレーション方法

GAによって生成される個体の強さを計測するために以下の手順でシミュレーションを行う。このシミュレーションでは到達することができた階層の深さとどれだけの回数勝ち続けることができるかを計測する。

- (1) PNPC に設定されるパラメータを初期状態にする
- (2) シミュレーション用の簡素なマップを生成する
- (3) PNPC, 敵キャラクタ, 階段をランダムに配置する
- (4) 終了条件となるまで Min-Max 法によりゲームを進行する
- (5) 現在の局面に勝利した場合, 手順2に戻る
- (6) 現在の局面に敗北した場合, シミュレーションを終了する
- (7) PNPC が到達することのできなかったダンジョンの各階層に対する推奨レベル $v_{lv_{floor}}$ を引き上げる

PNPCの初期状態とは、ゲーム開始時の状態を示す。手順4までが1回の試行であり、シミュレーションを継続するかどうかは終了条件のいずれに該当するかで決定される。以下の3項目で終了条件を示す。

- PNPC が全ての敵を倒す
- PNPC が階層の移動を行う

- 敵からの攻撃により PNPC の HP が 0 になる

第一・第二終了条件は PNPC が現在のダンジョンを攻略したことを示す。この場合、シミュレーションの手順5を実行する。一方、第三終了条件は PNPC が現在のダンジョンで敗北したことを示す。この場合はシミュレーションを終了する。シミュレーションが終了する時にはその PNPC が到達できなかったダンジョンの階層の推奨レベルを引き上げることで各階層の推奨レベルの調整を行う。シミュレーションの1試行はいずれかの終了条件に該当するまでとし、PNPC が勝利し続けた場合には、最大500試行が経過したらシミュレーションを終了する。なお、このシミュレーションでは矩形の部屋1つとその4辺から直線の通路が伸びているシンプルなダンジョンを用いる。

6.3 適応度関数の定義

GAで用いる適応度関数を式7で定義する。 f_{reach} はゲームの到達率に関する適応度であらわし、 $f_{variance}$ は PNPC の立ち位置に関する特徴量の重みのばらつきを表す。 β は線形和パラメータであり、2つの適応度の比重を調整するために用いる。

$$f = \beta f_{reach} + f_{variance} \quad (7)$$

ゲームの到達率に関する適応度を式8で求める。 n_{floor} はシミュレーションで到達することのできたダンジョンの階層の深さを表し、PNPC がダンジョンのより深い階層へ到達することができれば高い評価を与える。 n_{simnum} は PNPC がシミュレーションを継続することのできた回数を意味し、継続回数が多いほど良い個体であることを意味する。 n_{simmax} は最大シミュレーション回数を意味し、今回は500とする。

$$f_{reach} = n_{floor} + \frac{n_{simnum}}{n_{simmax}} \quad (8)$$

PNPCの立ち位置に関する特徴量の重みのばらつきに関する適応度は式9を用いて求める。ここでは重み w_{m1}, w_{m2}, w_{m3} についての評価を行う。適応度を求める子個体と個体群の共通のパラメータを昇順に並べた集合をそれぞれ N_{m1}, N_{m2}, N_{m3} とし、各集合の n 番目と $(n+1)$ 番目の値の差の集合を $Dif_{m1}, Dif_{m2}, Dif_{m3}$ とする。それらの要素の平均を ave_{mk} とし、分散を var_{mk} とする。式9は ave_{mk} を最大化しつつ var_{mk} を最小化する数式である。この式は集合 Dif_{mk} の範囲を最大化しつつ等差数列に近似する式である。

$$f_{variance} = \frac{\sum_{k=1}^3 ave_{mk} - \sum_{k=1}^3 \log var_{mk}}{3} \quad (9)$$

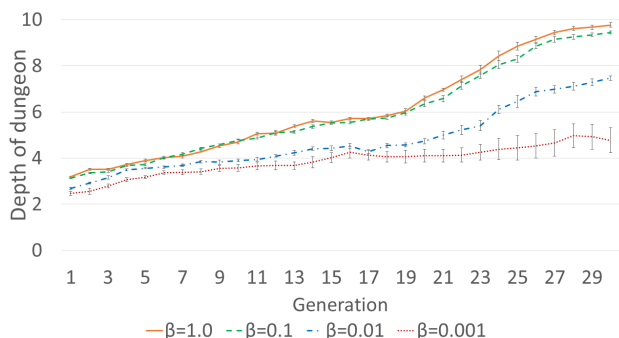


図 1 到達率 n_{floor} の推移
Fig. 1 Transition of the arrival rate of n_{floor} .

6.4 提案手法の基本性能の実験

まず初めに、提案手法の基本性能を確認するために適応度関数の線型パラメータである β の値を複数用意し、生成された PNPC によるゲームの到達率の結果から検証する。さらに、PNPC の位置に関する特徴量の重みの分布について、プロットした画像を用いて検証する。

6.4.1 学習プロセスの検証

図 1 は各世代におけるすべての PNPC の到達率の平均をグラフ化したものである。この図においてそれぞれの線は β が 1.0, 0.1, 0.01, 0.001 の時の到達率の推移を表している。 β が 1.0, 0.1 の時はほぼすべての PNPC が第 26 世代あたりでダンジョンの最深部である 10 階層まで到達していることがわかる。一方、 β が 0.01, 0.001 の時は真に強い AI は生成されていないことがわかる。

6.4.2 生成された個体の遺伝子情報の分布による検証

この研究の目的は PNPC の行動の多様性を確保することであり、強い AI を生成する必要は必ずしもない。そのため、十分に強い AI が生成された $\beta = 1.0$ の結果と、到達率が平均して 8 割程度まで強化することができた $\beta = 0.01$ の結果を比較し、生成された個体に現れる特徴量の重みのばらつきを確認する。

図 2 は $\beta = 1.0$ で生成された個体が持つ遺伝子情報のうち w_{m_1} と w_{m_3} を二次元平面上にプロットした図である。同じように、図 3 は $\beta = 0.01$ で生成された個体が持つ遺伝子情報のうち w_{m_1} と w_{m_3} を二次元平面上にプロットした図である。各図の 3 種類の点は PNPC が到達することのできたダンジョンの深さを示す。円形の点はダンジョンの 9~10 階層まで到達したことを示し、十分に強い AI である。三角の点はダンジョンの 6~8 階層まで到達したことを示し、ほどほどの強さを持っている。四角の点はダンジョンの 5 階層以内で敗北したことを示し、AI の強さは低い。

図 2 からは、各個体の立ち位置に関する特徴量の重みが

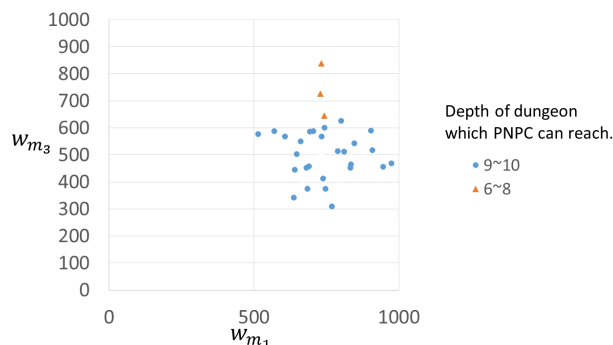


図 2 $\beta = 1.0$ で生成された個体における、 w_{m_1} 、 w_{m_3} の分布図
Fig. 2 The plot of w_{m_1} and w_{m_3} of all generated individuals with $\beta = 1.0$.

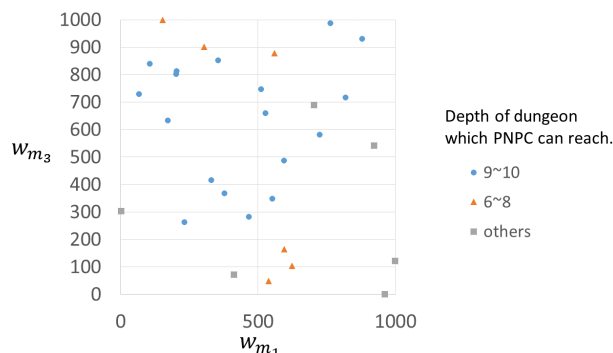


図 3 $\beta = 0.01$ で生成された個体における、 w_{m_1} 、 w_{m_3} の分布図
Fig. 3 The plot of w_{m_1} and w_{m_3} of all generated individuals with $\beta = 0.01$.

近傍に配置されている。 β の値を大きくすることでゲームを攻略することが重視され、最適な個体に収束しつつあることが理解できる。対照的に図 3 からは生成された個体の立ち位置に関する特徴量の重みは、取りうる値の全域に散らばっていることが見て取れ、二次元平面上の周辺に三角や四角がプロットされていることがわかる。このことから、 β の値を小さくすることで、特徴量の重みのばらつきが重要視され、到達率が多少下がる代わりに生成される特徴量の重みにばらつきが生じていることがわかる。この結果から、 $\beta = 0.01$ として生成された個体はほどほどの実力を保持しながら多様な振舞いを有している可能性があることが示される。

7. PNPC の振舞いの多様性の検証

このセクションでは、各個体の特徴量の重みによって決定される PNPC の振舞いの多様性について、数学的手法

表 3 抽出した 4 つの個体の立ち位置に関する特徴量の重み

Table 3 Weights of the feature parameters related PNPC's position of the using this experiment.

個体	w_{m1}	w_{m2}	w_{m3}
A	902	64	304
B	72	997	413
C	104	153	623
D	368	311	379

およびアンケートを用いた被験者実験により検証する。数学的手法については行動ログのコサイン類似度を用いて PNPC の振舞いの類似度を測定する。

7.1 コサイン類似度を用いた PNPC 間の振舞いの計測による多様な振舞いの検証

この実験では、ゲーム AI の多様性を PNPC が選択する行動ログに対しコサイン類似度の測定によって検証する。類似度については、8 方向への移動および攻撃、行動を何もしない、ダンジョンの階層の移動の計 18 行動について計測する。類似度を測定するためには完全に同じ状況からゲームを開始する必要があるため、行動ログを取得するために同じ乱数シード値を用いた同一のダンジョンを生成する。行動ログは 150 種類の異なるマップから取得し、各マップの開始時の PNPC のレベルはマップごとに統一する。ダンジョンの形状は学習時にシミュレーションで用いたものと同様とする。

この実験の初めに、GA により生成された AI が持つ特徴量の重みの妥当性を検討する。 $\beta = 1.0$ および 0.01 として生成されたすべての個体に対して行動ログを取得し、各個体間の類似度を計測する。 $\beta = 1.0$ として生成された個体間の類似度と $\beta = 0.01$ として生成された個体間の類似度について t 検定を行ったところ、 $\beta = 0.01$ として生成された個体間の類似度が有意に小さいことがわかった。このことから、PNPC の立ち位置に関する特徴量の重みにばらつきを実現することで、実際の振舞いに差が表出することがわかる。

続いて、 $\beta = 0.01$ として生成された AI の中から特徴的な 4 つの AI の行動選択の類似度について比較を行う。特徴的な 4 つの AI はゲームの到達率が 6 割以上のものうち、 w_{m1} , w_{m2} , w_{m3} のうち、いずれかが突出している個体と 3 つの特徴量の重みが平均的な個体を用いる。表 3 でこの実験で用いた個体が持つ PNPC の立ち位置に関する特徴量の重みを示す。

表 4 は 4 つの個体間の類似度の平均を示している。個体 A・B・C に関しては互いの類似度が 0.6 前後と低いことがわかる。この結果はそれぞれの特徴的な個体が異なる振舞いを行っていることを示す。また、個体 C・D 間の類似度は他の組み合わせに比べて高いことがわかる。これは個

表 4 4 個体間の行動ログの類似度の平均

Table 4 The similarity of the behavior log between the individuals.

個体	類似度
A,B	0.5914
A,C	0.6530
A,D	0.6678
B,C	0.6813
B,D	0.6899
C,D	0.9555

体 C および D は似た振舞いを行っていることを示す。個体 C は w_{m3} に特化した個体であり、ダンジョン内の部屋の中央部で行動することを好む AI といえる。一方個体 D は平均的な個体であるため、立ち位置には特別こだわることなく、現在の配置付近で適した行動を行っていると考えられる。各マップの開始時にはランダムな位置にキャラクターが配置されるが、配置される場所は部屋内に限り、通路内に初期配置されることはない。また、生成される部屋の大きさにもよるが壁側に該当する配置よりもフロアの中央に位置する配置の方が総数が多く、初期配置で部屋の中央から始まる確率が高い。そのため、個体 C・D どちらも部屋の中央での行動を行ったものと考えられる。

7.2 主観評価実験による PNPC の区別の検証

この実験ではアンケートを用いた被験者実験を行い、PNPC の行動の多様性を検証する。この実験では表 3 と同じ個体を用いる。以下に実験の手順を示す。

- (1) 4 つの個体から 2 つの個体を選択し、10 グループを作る。(同じ個体を含むグループ 4 つ、異なる個体を含むグループ 6 つ)
- (2) 被験者へグループを構成する個体によるプレイ動画をそれぞれ 1 つずつ提示する。
- (3) プレイ動画の視聴後、PNPC の振舞いに関するアンケートに答えてもらう。
- (4) 10 グループ分、手順 2・3 を繰り返す。

この実験は 12 人の大学生に対して実施した。被験者へ提示する動画はあらかじめ撮影されたものを用い、その長さは 1 つあたり 1 分 30 秒程度である。PNPC が探索するダンジョンは学習時のシミュレーションで用いたものと同様とし、部屋の大きさ・通路の配置・キャラクターの配置は完全にランダムなものとする。被験者へのアンケートは 2 つの質問で構成しており、1 つ目の質問は提示した 2 つのプレイ動画が違うプレイヤーが操作しているかどうかを答えてもらうものである。この質問では各グループの動画が異なる操作者によるものに見えるかを 5 段階で評価してもらう。評価が高いと異なるプレイヤーが操作していると被験者が知覚したことを意味する。2 つ目の質問は自由筆記のアンケートであり、提示した 2 つのプレイ動画におけるプレ

表 5 自由記述アンケートの結果

Table 5 Excerpt of answers to question 2.

グループ	コメント
A,A	どちらも通路に逃げ込む どちらも上級者に見える
A,B	Aは通路に逃げている Bは通路までは逃げない
A,C	Aは何度か通路に逃げているがCは逃げない Cは敵に近づいて倒す
A,D	Aは敵をひきつけて倒すがDは積極的である Dはとりあえず近くの敵に向かっている
B,B	どちらも同じ行動をしているように見える どちらも中級者に見える
B,C	Bの方が壁側にいることが多く見える Cは勇敢なイメージが強い
B,D	They look bellicose.
C,C	どちらも敵に向かって動いているように見える 同じ倒し方に見える
C,D	Cはどんな状況でも逃げていない Dは敵に囲まれそうなら通路に逃げている
D,D	1つ目のDはとりあえず攻撃している 2つ目のDは壁際で攻撃している

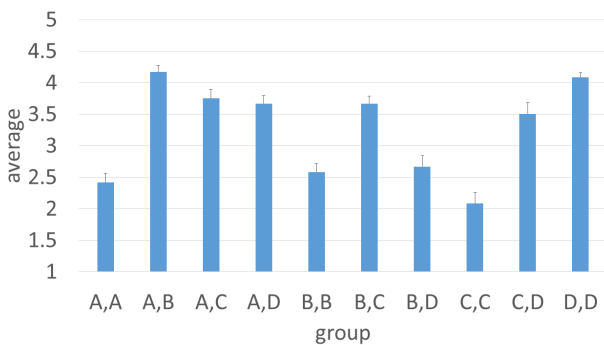


図 4 設問 1 の解答の平均

Fig. 4 Average of the answer in the first question.

イの特徴について答えてもらう。

図 4 でアンケートの設問 1 の結果を示しており、横軸は各グループを、縦軸は評価の平均値を表す。グループ (A,A) の平均値はダネット法に基づいて有意水準 5% でグループ (A,B)・(A,C)・(A,D) よりも小さい事がわかった。同様に、グループ (B,B) はグループ (A,B) よりも、グループ (C,C) はグループ (A,C)・(B,C)・(C,D) よりも小さいことがわかった。この結果から、個体 A・B・C のように特徴的な個体は、人の目から見て他の個体と異なる振舞いが表出していることがわかる。

表 5 は自由記述に寄せられたコメントの一部を抜粋したものである。多くの被験者は、グループの構成が同じ AI である時には同じプレイヤーによってプレイされていると回答していることがわかる。立ち位置に関する特徴量の重みのうち、いずれかの重みに特化している個体は、他の個体に比べて振舞いに違いが表出していることもこのアンケートから見て取れる。個体 D については固有の特徴はなく、好戦的である、逃げることを好むなど様々な特徴が見られた。これらのコメントは個体 D がその場の状況に合わせて、3 種類の立ち位置のうち最善な領域を選択し、行動していることが示唆される。

7.3 考察

$\beta 1.0$ として生成された十分に強いが特徴量の重みが集中している NPC 群と 0.01 として生成されたそこそこ強く、特徴量の重みがばらついている NPC 群の個体間の行動の類似度を測定したところ、生成された個体間の行動の類似度は $\beta = 0.01$ として生成された個体間の行動の類似度が有意に小さかった。この結果から、立ち位置に関する特徴量の重み、すなわちプレイスタイルに関わる特徴量の重みを調整する事が NPC の振舞いの多様性に有用であると考えられる。 $\beta = 0.01$ として生成された個体のうち、特徴的な 4 つ個体については特に振舞いの類似度が低いこと

が行動ログから計測された。これらの結果は、GA の初期設定として、意図的に多様化する特徴量を指定することによって NPC の動作を制御できる可能性を示唆している。

被験者実験では、プレイスタイルに関する特徴量の重みのいずれかを特化した NPC の振舞いは被験者にとって、それぞれが別のプレイヤーであると認識できることが示された。また、自由記述では NPC の立ち位置に関するコメント意外に憶病や積極的など性格に関する解答が含まれていた。このことから、提案手法は NPC の個性を形成するために利用できる可能性が見られた。

立ち位置に関する特徴量の重み 3 つが平均的な個体 D については、初めの実験では V_{m3} に特化した個体と類似した振舞いが見られたが、2 つ目の実験では複数の特徴がみられる回答が得られた。個体 D に対するこれらの結果は立ち位置に関する特徴量の重みが平均的な AI が機知に富んだ振舞いを実現していること示唆している。

今回実施した 2 つの実験より、GA における適応度関数に遺伝子情報のばらつきに評価を与える項を適応する事で、導出される個体が一意に収束することを防ぐことができ、それらの振舞いに違いが表出することがわかった。しかし、特徴的な個体の類似度は検証したが、類似度の低さはどのような行動選択からもたらされているのか、また、生成されたすべての個体の振舞いについてそれぞれがどのような特性を持っているかは検証できていない。加えて、本稿では NPC の立ち位置に関する特徴量の重みにのみ着目しており、そのほかの特徴量の重みに着目して学習した場合に同様に振舞いに違いが表出するかはわからない。

8. 終わりに

本研究では、GA を用いて PNPC の振舞いの多様性を実現するためのシステムを提案し、 로그樂ゲームを用いて検証した。今回の検証対象となる PNPC のゲーム AI の基本構造としては MinMax 法を用いた。局面評価関数に使用される特徴量の重みを多様化する事を実現するために、GA における適応度関数に遺伝子情報がばらつくことに評価を与える項を適用した。

提案手法によって生成された PNPC の行動については 2 つの実験によりその多様性を検証した。1 つ目の実験は個体間の行動ログの類似度の測定によるもので、GA における適応度関数に用いた 2 つの項 f_{reach} , $f_{variance}$ の比重を調整する事でゲームをそれなりに攻略できるだけの実力を保持しつつ、各個体間の行動の類似度が低い、すなわち個体ごとの行動に違いが生じる個体群が生成されたことが検証された。さらに、特徴的な 4 個体について注目すると、PNPC の立ち位置に関する特徴量の重みを多様化することで個々の個体は大きく異なる振舞いを取っていることがわかった。2 つ目の実験では、提案手法によって生成された PNPC が人間から見て異なるプレイヤーに見えるかを被験者実験により検証した。その結果、個体間の類似度の測定でも用いた特徴的な 4 つの PNPC については、人の目から見ても十分に異なったプレイヤーによって操作されていると知覚されていることがわかった。さらに、特徴量の重みが平均的な個体については特定の特徴は見られず、臨機応変な行動選択がされていることがわかり、提案手法により生成される個体は特徴的なもののみではなく、機知に富んだ行動ができる AI の生成が可能になる可能性が示唆された。

2 つの実験の結果から、プレイスタイルに関する特徴量の重みにばらつきを実現することで PNPC の振舞いに多様性が生じることが示唆された。しかし、今回の実験では特徴的な 4 つの個体についてのみ深い検証を行っており、生成されたそのほかの PNPC についての検証は今後精緻に行う必要がある。また、今回の検証では PNPC の立ち位置意外の特徴量についてのみの検証を行った。しかし、PNPC の立ち位置がプレイスタイルに最も影響しているかは明確ではないため、そのほかの特徴量の重みについても検証を行い、どのような特徴量がプレイスタイルに大きく関わっているかを明確にする必要がある。提案手法によって生成される PNPC が持つ振舞いについてはその類似度の測定は行ったが、どのような特性を有しているかは検証できていない。提案手法を実用的にするためには、単に多様な振舞いを生成するのみでなく、生成される PNPC の振舞いのある程度制御できる必要があるため、提案手法を改良し、生成される PNPC の振舞いを制御できるシステムを目指す。

参考文献

- [1] Fusita, H. and Ishii, S: *Model-based reinforcement learning for partially observable games with sampling-based state estimation*, Neural Computation(2007), pp3051-3087.
- [2] Hoki, K. and Kaneko, T: *The Global Landscape of Objective Functions for the Optimization of Shogi Piece Values with a Game-Tree Search*, Advances in Compute Games(2012), pp184-195.
- [3] Julian Togelius, Sergey Karakovskiy and Robin Baumgarten: *The 2009 Mario AI Competition*, IEEE(2010).
- [4] om Pepels, Mark H.M. Winands, and Marc Lanctot: *Real-time Monte-Carlo Tree Search in Ms Pac-Man*, IEEE(2013).
- [5] Amit Benbassat and Moshe Sipper: *Evolving Lose-Checkers Players using Genetic Programming*, IEEE(2010).
- [6] Ng Chee Hou, Niew Soon Hong, Chin Kim On, and Jason Teo: *Infinite Mario Bros AI using Genetic Algorithm*, IEEE(2011).
- [7] 北清 勇摩, 伊藤 毅志: カーリングの戦略を支援するシステムの提案と構築, GPW(2013).
- [8] 上田 陽平, 池田 心: 遺伝的アルゴリズムによる人間のレベルに適應する多様なオセロ AI の生成情報処理学会研究報告 (2012).
- [9] 藤井 叙人, 佐藤 祐一, 若間 弘典, 風井 浩志, 片寄 晴弘: 生物学的制約の導入によるビデオゲームエージェントの「人間らしい」振舞いの自動獲得, 情報処理学会論文誌,(2014).
- [10] 星野 准一, 田中 彰人, 濱名 克季: 模倣学習により成長する格闘ゲームキャラクタ, IPJS(2008), pp 2539-2548.
- [11] Anna I. Esparcia-Alcázar, Anaís Martínez-García, Antonio Mora, JJ Merelo and Pablo García-Sánchez: *Controlling bots in a First Person Shooter Game using Genetic Algorithms*, IEEE(2010).
- [12] 佐藤 浩, 小野 功, 小林 重信: 遺伝的アルゴリズムにおける世代交代モデルの提案と評価, 人工知能学会 (1996).
- [13] Larry J.Eshelman and J.David Schaffer: *Real-Coded Genetic Algorithms and Interval-Schemata*, 1993, pp187-202.