

確率モデルを用いた協調共進化と確率増殖プログラム進化による 対戦ロールプレゲームの設計方法

陳 浩洋[†] 森 康久仁[†] 松葉 育雄[†]

千葉大学融合科学研究科[†]

1. はじめに

対戦 RPG(Role-playing Game)ゲームの設計において、ゲームのバランスが適正か否かはゲームの品質を評価する時の最も重要な要因になっている^[1]。なぜならば、バランスの良し悪しがゲームの面白さに大きな影響を与えるからである。

典型的な対戦 RPG では、プレイヤーはゲーム中に操作するキャラクターの特質を決定する「種族」を選び、ゲームをプレイすることになる。各種族は特有の能力を持ち、ゲームバランスはその能力の値の影響が非常に大きい。

従来の能力値決定方法では、種族の能力値を経験的に設定し、適切な値は試行錯誤によって決められる。従って、最適な値が求まるまで、多くの時間がかかることになり、効率が悪い。

本研究では、ゲームの確率モデルを導入し、「確率増殖プログラム進化」と「協調共進化」の二つの進化的メカニズムの結合により種族の能力関数の設計方法を提案した。また、実験により、提案方法の有効性を検証した。

2. 確率増殖プログラム進化と協調共進化

確率増殖プログラム進化(PIPE)は Schmidhuber et al. ^[2]によって提唱されたプログラムの自動生成を行うメカニズムである。この手法では、確率原型木(PPT)に保存されている可変の確率分布に基づき、プログラム要素が生成される。また、確率分布は PBIL (Population-Based Incremental Learning)などを用いた学習により修正される。

協調共進化は、最適化問題を複数の部分問題に分解し、各部分問題に Species と呼ばれる部分集団ごとに進化させるアルゴリズムである^[3]。

協調共進化は分解可能な最適化問題に対して、通常の進化的アルゴリズムにより非常に有効である^[4]。

3. 提案方法

Design Method for Fighting Role-playing game with Probabilistic Incremental Program Evolution and Cooperative Coevolution using Stochastic Model

[†]Chen Haoyang, Yasukuni Mori, Ikuo Matsuba; Graduate School of Advanced Integration Science, Chiba University

対戦型 RPG ゲームには大きく分けて「Turn-based ゲーム」と「Action-based ゲーム」の 2 種類が存在する。本報告では主に「Turn-based ゲーム」を扱う。また、ここでは簡単のため、「1 対 1」の戦闘を考える。これは「1 対 1」の戦闘が RPG ゲームにおける最も基本的な過程であるからである。

3.1 評価関数の分析

どちらの種類ของเกมでも、ゲームバランスを保つためには、異なる能力値を持つ異なる種族のキャラクター間の戦闘であっても、同じレベルのキャラクターであればどちらかの種族に勝つ確率が極端に偏ってはならない。

従って、進化的アルゴリズムによって、ゲームバランスを適切に保つように各種族の能力値を決定するには、戦闘の勝つ確率に基づいた評価関数を用いなければならない。

3.2 CC と PIPE との結合

CC と PIPE は互いに独立なメカニズムなので、それらを結合するために、以下の五つの修正を行った。また、図 1 に提案方法の枠組を示す。

- 1 ゲームにおける能力関数の数に応じて Species を用意し、Species ごとに、PPT を一つ生成する。
- 1 各 Species の集団個体はその Species の対応する PPT によって生成される。
- 1 CC のメカニズムで、評価関数により順番に各 Species を評価する。ここでは、強化学習により代表者を選択する「T_CCEA」と当世代の最良個体を代表者として選択する「CCEA」の二つのアルゴリズムを用いた。
- 1 PPT が最適なプログラムに収束するため、各 Species ではエリート保存戦略を行う。
- 1 各 Species の集団個体数が小さいため、閾値 FIT_S を PPT の学習関数に導入した。

4. 実験

今回使った「Turn-based ゲーム」モデルでは、各種族は生命値と攻撃力と回避率の三つの要素

を持つとする。種族数を 3 とし、各種族の回避率をそれぞれ 0.1, 0.6, 0.2 にした。E_{ij}, D_{ij} は種族 i が種族 j に勝つ確率の期待値、分散と定義する。∀i, j ∈ {1, 2, 3} (但し, i ≠ j) に対して、E_{ij} を 0.5, D_{ij} を 0 にすることがゲーム設計の目標である。

提案法によって求めた最良解の適応度に対応する E₁₂, E₂₃, E₃₁ は 0.5041, 0.4942, 0.5074 になり、D₁₂, D₂₃, D₃₁ は 0.0255, 0.0079, 0.0179 になった。また、図 2, 図 3 に各種族の最良な生命値関数、攻撃力関数を示し。図 4 に CCEA と T_CCEA と通常の進化的アルゴリズム(EA)との性能比較の結果を示す。

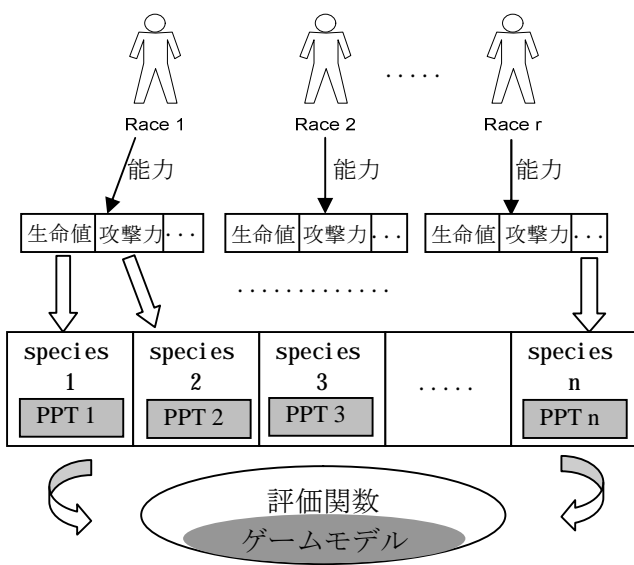


図 1 システムの枠組

5. まとめと今後の課題

実験の結果から、各種族の回避率が異なっていたとしても、E_{ij}, D_{ij} の値を目的とした値に近づかせることができた。つまり、設計の目標を達成したと言える。従来の方法と比べると、提案法では、最適値を探索する時間を短縮できるだけでなく、各種族の能力値の正確性も改善することができた。

今後の課題として、四つの点が挙げられる。

- 1 「Turn-based ゲーム」に対して、実際のゲームに近づくモデルを目指す。
- 1 「Action-based ゲーム」の場合は、対戦の運動モデルを導入し、確率モデルを分析する。
- 1 新たなモデルに応じ、評価関数を修正する。
- 1 今回の実験で出た進化しにくい現象を無くすため、今後はもっとヒューリスティック要素を導入し、Bayesian Network によって依存関係の推定を行うモデルで考え、アルゴリズムを見直しをする。

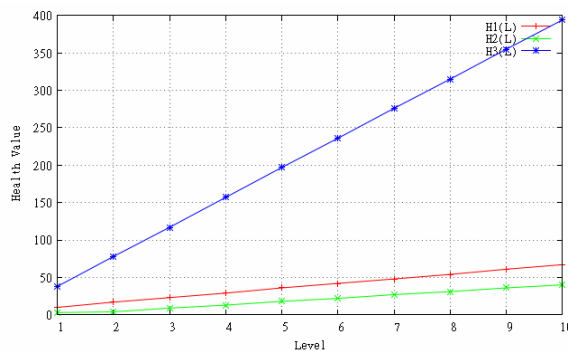


図 2 生命値関数

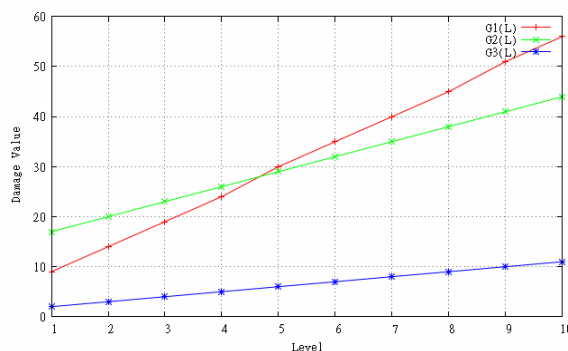


図 3 攻撃力関数

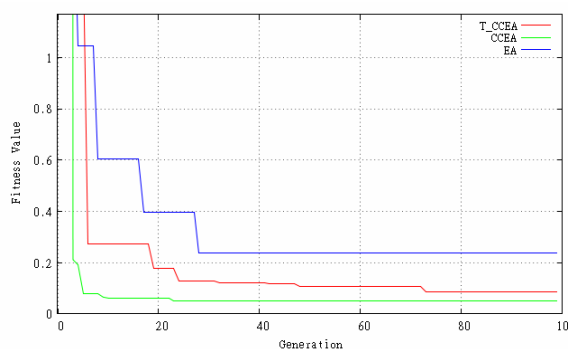


図 4 CCEA, T_CCEA, EA 性能の比較

参考文献

- [1] Ernest, Adams. : Fundamentals of Game Design 2nd Edition, New Riders, pp.324-355, 2009.
- [2] Salustowicz, R. and Schmidhuber, J. : Probabilistic Incremental Program Evolution, Evolutionary Computation, vol.5, no.2, 1997.
- [3] M, Potter. and K, De Jong. : A Cooperative coevolutionary approach to function optimization, In Proc. of the Third Conference on Parallel Problem Solving from Nature, p.p.249-257, Jerusalem, Israel, 1994.
- [4] M, Potter. and K, De Jong. : Cooperative Coevolution: An architecture for evolving coadapted subcomponents, Evolutionary Computation, 8(1):1-29, 2000.