

## 複数木構造を持つ GP のための木の数の自動決定

伊藤 崇<sup>†</sup> 高橋 健一<sup>†</sup> 稲葉 通将<sup>†</sup>

広島市立大学大学院情報科学研究科情報科学専攻<sup>†</sup>

### 1. はじめに

近年、計算機上で人間の知能のモデル化を目指す研究分野である人工知能に注目が集まっている。その中にエージェント学習がある。エージェント学習に用いられる手法に、進化的計算の一つである遺伝的プログラミング (GP) [1],[2]がある。GP では、1つの個体を非終端ノードと終端ノードから成る1つの木で表現する。交叉や突然変異といった遺伝操作によって子個体を生成し、進化を繰り返すことで問題に適した解を獲得する。

我々は1個体が複数の木を持つように拡張した調整ノード付き GP ( $GP_{CN}$ ) を提案している。 $GP_{CN}$  では1個体が複数の木を持つために、木の数  $M$  と木ごとの参照回数を表すプロセス数  $P$  をあらかじめ設定する必要がある。先行研究において、進化の過程で木に適したプロセス数  $P$  を獲得するために、プロセス数  $P$  の継承手法を提案した[3]。

本研究では、進化の過程で木の数  $M$  を自動決定するための方法を提案する。先行研究では、評価関数に終端ノードのみの木を考慮した適合度を用いていた [3]。しかしながら、この適合度では木の数が評価関数に反映されない。そこで、先行研究で用いた適合度に、木の数  $M$  の少ない個体が優良であるという評価を加えた新たな適合度を提案する。

また、新たな適合度は木の数  $M$  を減らす働きしかないために、木の数  $M$  が1度最小の数に収束してしまうと他の木の数を探索することができない。そこで、木の数  $M$  に多様性を持たせるために、木の数  $M$  を増やす個体の突然変異を提案する。これらの提案手法をごみ拾い問題に適用し、評価比較を行う。

### 2. 調整ノード付き GP

調整ノード付き GP ( $GP_{CN}$ ) の個体は、図1に示すように複数の木で構成されている。個体の木の数  $M$  は、個体ごとに[2, 20]からランダムに設定される。 $GP_{CN}$  の個体は、それぞれが規則に対応する複数の木から構成されている。各木は、参照順を表す識別番号と参照回数を表すプロセス数  $P$  を持っている。また、プロセス数  $P$  は木ごとに独立しており、初期値として[1, (TotalSteps) $M$ ]からランダム

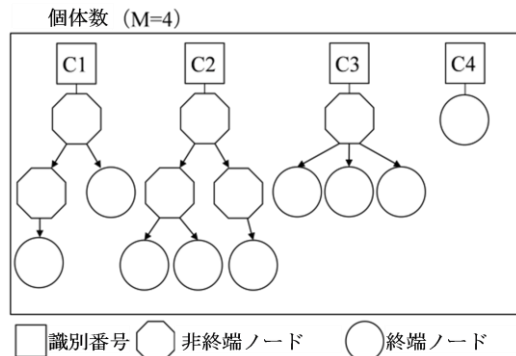


図1  $GP_{CN}$  の個体例

ムに選択された値を設定する。ここで、*TotalSteps* は問題ごとに設定されているエージェントの最大参照回数である。木の参照は、識別番号の最も小さい木から行い、エージェントがその木を参照した回数とその木のプロセス数  $P$  を超えたとき、次の識別番号を持つ木に遷移する。また、識別番号の最も大きい木においてエージェントの参照した回数がプロセス数  $P$  を超えた場合、次に遷移するのは、識別番号の最も小さい木である。 $GP_{CN}$  の進化のアルゴリズムは GP と同様であり、遺伝操作には交叉、突然変異、逆位を用いる。

### 3. プロセス数 $P$ の継承手法

$GP_{CN}$  は1個体が複数の木を持つため、それぞれの木の参照回数を表すプロセス数  $P$  をあらかじめ設定する必要がある。これを進化の過程で獲得するために、プロセス数  $P$  の継承手法を提案している。継承手法は、2つの親個体を交配して2つの子個体を生成する交叉において、2つのうち優良な親個体のプロセス数  $P$  の値を2つの子個体に引き継ぐ手法である。

### 4. 提案手法

#### 1) 木の数 $M$ を考慮した適合度

先行研究では、評価関数に終端ノードのみの木を淘汰するための適合度を用いていた。本研究では、この適合度に木の数  $M$  による個体の評価を加える。本研究で用いる適合度の計算式は実験の中で説明する。これにより、木の数  $M$  が小さい個体ほど評価が良くなる。

#### 2) 個体の突然変異

木の数  $M$  を考慮した適合度を評価関数に加えることで、木の数  $M$  が少ない個体ほど適合度が高くなり、できるだけ少ない木の数  $M$  で問題に適した

Automatic determination of the number of tree for genetic programming with multiple trees

<sup>†</sup>Graduate School of Information Sciences, Hiroshima City University

行動規則を獲得することが可能になった。

しかしながら、初期生成時にはほとんどの個体において問題ごとに設定された適合度が 0 であるため、木の数  $M$  を評価関数に加えることで早い世代で木の数  $M$  が最小の数に収束し、それ以外の木の数が全く探索されない可能性がある。そこで、木の数  $M$  に多様性を持たせるために、個体の突然変異を提案する。

個体の突然変異は、木を増やす 2 つの操作から成る。1 つ目の操作は木の複製である。この操作では、個体内の 1 つの木を複製し追加することで木の数を増やす。2 つ目はランダム木の追加である。この操作では、ランダムに生成した 2 つの木を個体に追加する。どちらの手法の木も参照順の最後に追加するのではなく、ランダムな位置に追加される。また、個体の突然変異は毎世代行われず、個体集団中の個体の木の数  $M$  が全て同じ時、設定した確率で行われる。

## 5. 実験

### 1) ごみ拾い問題

ごみ拾い問題は、エージェントが限られた行動回数の中でより多くのごみを廃棄するという問題である。縦 11×横 11 のフィールドに、ごみを 2 つまで保持できるエージェントが 1 体、ごみが 10 個、ごみ収集所が 1 箇所設置されている。通常、エージェントとごみの配置を変えた 10 種類の環境を用意し、それぞれの環境で行動回数 250 以内に廃棄できたごみの総数を適合度とする。個体  $j$  の適合度は、環境の種類を  $i$ 、その環境で廃棄されたごみの数を  $N_i(j)$  とすると式(1)により求めることができる。

$$T(j) = \sum_{i=1}^{10} N_i(j) \quad (1)$$

本研究では、先行研究で用いた終端ノードのみの木を淘汰するための適合度に、木の数による評価を加えた適合度を用いる。個体  $j$  の木の数を  $M(j)$ 、その個体に含まれる終端ノードのみの木の数を  $M_{single\ node}(j)$ 、終端ノードのみの木に対する重みを  $\alpha$  とすると、次の式によって適合度を求めることができる。

$$f_{new}(j) = T(j) + (TotalSteps - M(j)) - \alpha \times \frac{M_{single\ node}(j)}{M(j)} \quad (2)$$

ここで、第 2 項の個体の木の数  $M$  による評価において  $TotalSteps$  から  $M(j)$  を引いているのは、適合度が負の値になることを防ぐためである。

### 2) 実験結果

性能評価には、各試行でエージェントが廃棄したごみの数の合計  $T$  を用いる。各試行で最も高い  $T$  の 30 試行の平均を図 2 に示す。図 2 には、従来手法である GP、 $GP_{CN}$  と  $GP_{CN,I}$ 、提案手法である

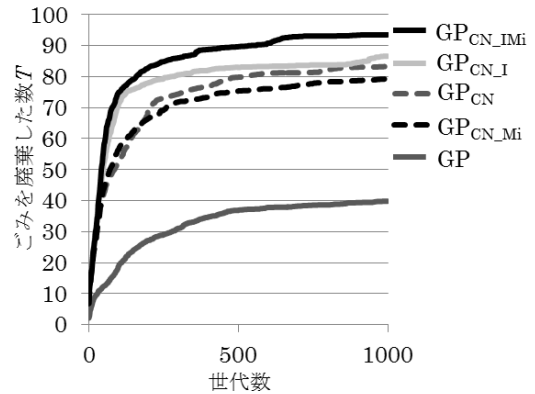


図 2 ごみ拾い問題における 1000 世代で廃棄したごみの数  $T$  の変化

$GP_{CN,Mi}$ 、 $GP_{CN,IMi}$  の結果を示している。

プロセス数  $P$  の継承手法と個体の突然変異を組み合わせた  $GP_{CN,IMi}$  が、従来手法である GP や  $GP_{CN}$ 、 $GP_{CN,I}$  よりも高い  $T$  の値を示した。これは、進化の過程で問題に適したプロセス数  $P$  と木の数  $M$  をうまく獲得できたためだと考えられる。個体の突然変異を単体で導入した  $GP_{CN,Mi}$  の  $T$  の値が  $GP_{CN,IMi}$  よりも低くなったのは、ランダムな木の追加によってランダムなプロセス数  $P$  が与えられてしまい、適したプロセス数  $P$  が獲得できなかったからだと考えられる。

### 6. おわりに

本研究では、GP の拡張手法である  $GP_{CN}$  において、進化の過程で木の数  $M$  を自動決定するために適合度の変更し、個体の突然変異を提案した。ごみ拾い問題において、木ごとの参照回数を自動決定するプロセス数  $P$  の継承手法と個体の突然変異を組み合わせることで最も良い性能を示した。

今回提案した個体の突然変異では、木の数  $M$  を追加する操作しか行っていない。そのため、今後の課題として削除する操作の追加が考えられる。

### 謝辞

本研究は広島市立大学特定研究（一般研究費）による。

### 参考文献

- [1] John Reed Koza, “Genetic programming: on the programming of computers by means of natural selection”, MIT Press, Cambridge, 1992.
- [2] 松村康平, 花田良子, 小野景子, “木構造最適化におけるノードの依存関係を考慮した近傍探索に基づく多段階探索交叉”, 第 26 回インテリジェント・システム・シンポジウム, pp.178-180, 大阪大学吹田キャンパス, 2016.
- [3] Takashi Ito, Kenichi Takahashi, and Michimasa Inaba, “Obtaining repetitive actions for genetic programming with multiple trees”, Procedia Computer Science, vol.96, pp.120-128, 2016.