ISS-SP-050

# 線形遺伝子型を導入したグラフ構造 GP

阪野　優太†　　長尾　智晴††
† 横浜国立大学理工学部　　†† 横浜国立大学　大学院環境情報研究院

## 1. はじめに

自動プログラミングの分野では遺伝的プログラミング(GP)に関する研究が盛んに行われている. 個体を木構造で表現する GP に対して, 個体を有向グラフによって表現するグラフ構造 GP などの拡張手法が提案されている. 本稿では, グラフ構造 GP の一種である Graph Structured Program Evolution(GRAPE)[1]に対して, 線形構造を組み込んだ新たな自動プログラミング手法を提案する.

## 2. GRAPE

GRAPE ではプログラムを有向グラフとデータセットによって表現する. グラフ中の各ノードの処理はデータセットに対して行われる. 遺伝子型は, 各ノードの情報が並んだ固定長の整数列である. 各ノードには接続先の情報が含まれており, ノードの並びに処理の流れは反映されていない. このことによって, 遺伝操作が個体の有用な形質を破壊しやすく, 探索が非効率となる.

## 3. 提案手法

図 1 に提案手法の表現型と遺伝子型を示す. 有向グラフとデータセットで表現したプログラムを 2 次元の遺伝子型へ符号化する. プログラム中の 1 本の処理の流れを最初の行に割り当て, そこから分岐する処理を順番に各行へ割り当てることによって符号化を行う.

提案手法では, 処理ノード, 分岐ノード, ジャンプノード, 終了ノードの 4 種類のノードによってプログラムを表現する. 遺伝子型における各ノードの接続はノードの種類ごとに異なる. 処理ノードは隣接する次のノードに接続する. 分岐ノードは条件式が true の場合に i 行目の行頭に, false の場合に隣接する次のノードに接続する. ジャンプノードは k 行目の行頭に接続する. k はジャンプノードがもつ接続先の情報であり, ジャンプノードが i 行目に存在するとき, k は 0 から i-1 のいずれかの値をとる. なお, 0 行目にジャンプノードは出現しないものとする. 終了ノードは接続しない.

以上の接続規則に従って, 先頭のノードからの処理の流れが決定する. 提案手法の遺伝子型では, 各行がプログラム中の 1 本の処理の流れに対応するため, 遺伝操作によって有用な形質が破壊されにくい. また, ジャンプノードのみが接続先の情報を使用するため GRAPE よりも探索空間が小さい. さらに, 分岐方法, ジャンプ先の制約によって染色体の中で発現する領域が左上に固まるため, 発現領域同士の交叉が起こりやすい. このことによって効率的に探索が進むことが考えられる.

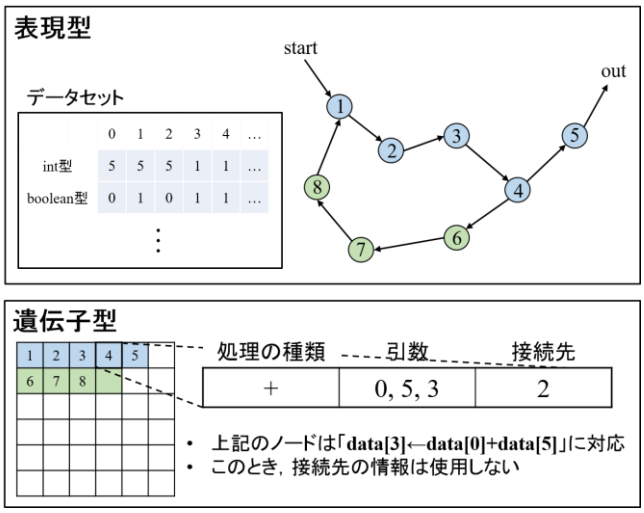


図1. 提案手法の表現型と遺伝子型

表1. 100 試行中の成功回数

| | factorial | | fibonacci | | sort | |
|---|---|---|---|---|---|---|
| | training | test | training | test | training | test |
| GRAPE | 69 | 59 | 8 | 6 | 75 | 28 |
| Proposal | **94** | **93** | **19** | **19** | **85** | **55** |

## 4. 実験結果

本稿では, 階乗計算, フィボナッチ計算, ソートプログラムの自動生成実験を行った. 階乗およびフィボナッチ計算では最大行数を 10 行, ソートでは最大行数を 20 行として実験を行った. 表 1 に, 100 試行中の成功回数を示す. 表 1より, 適用した 3 つの問題に対して, 提案手法は GRAPE を上回った. このことから, 提案手法の遺伝子型の構造によって, 遺伝操作が効率的にはたらいたことがわかった. また, 階乗およびフィボナッチ計算では 2 行, ソートでは 4 行のプログラムが生成されたことから, 最小行数での最適化が行われていることが確認できた.

## 5. まとめ

本稿では, GRAPE の遺伝子型に線形構造を導入した. 階乗計算, フィボナッチ計算, ソートプログラムの自動生成を行い, 成功回数において有効性を示した. 今後の課題として, 様々なプログラムの自動生成を行い, 有効性を示すことによって提案手法の汎用性を検証する.

## 参考文献

[1] Shinichi Shirakawa, Shintaro ogino, and Tomoharu Nagao. Graph Structured Program Evolution. *Proceedings of the genetic and Evolutionary Computation Conference (GECCO 2007)*, Vol. 2, pp. 1686-1693, 2007.