

複雑な構造体に適したコーディン手法の検討

6Q-3

山本 公洋 内藤 昭三 伊藤 正樹

NTT ソフトウェア研究所

1 はじめに

逆導出原理 [1] と遺伝的アルゴリズム [2] を用いて事例から Prolog プログラムを帰納推論するシステム: GA-CIGOL を提案した [3]. しかし, 従来試作したシステムでは交叉による致死遺伝子の多量発生が観測された. 本報告ではこの問題を解決するために, コーディング手法について考察, 提案を行なう. また, その性能評価のための計算機実験結果を示す.

2 GA-CIGOL のコーディング問題

逆導出は, 三段論法の過程を逆向きに辿る記号操作オペレータ (Truncation, Absorption, Intra-Construction) で, Horn 節を入力として受け取り, Horn 節を出力する (図 1 参照). あるオペレータにより生成された Horn 節は, さらに他のオペレータの入力となることができる. 逆導出オペレータの繰り返し適用の結果, Prolog プログラムが生成される. この過程は有向グラフ (図 1 参照) で表現することができる.

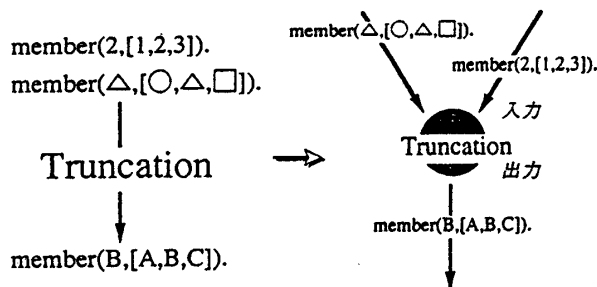


図 1: Prolog プログラム生成過程

逆導出によって生成される多量の Horn 節候補群の中から, 遺伝的アルゴリズムを用いて適切な Horn 節集合を選択する. 具体的には, 先に示した Prolog プログラムを生成する過程 (グラフ) を組み変えることで適切な Horn 節集合の探索をおこなう.

グラフは複雑な構造体であり, 交叉を単純に適用すると致死遺伝子を生じる.

3 致死遺伝子の定義

遺伝的アルゴリズムの特徴は, 領域対象の解表現形から別の解表現形を生成する規則にある. 交叉や突然変異の遺伝的的操作がこの規則に対応する.

¹Codings suitable for complicated structures in genetic search
Kimihiko YAMAMOTO, Shozo NAITO, Masaki ITOH
NTT Software Laboratories

複雑な構造体の解表現形に交叉や突然変異を行なうと, 解としての条件を満たさない表現形まで生成してしまう. この様な解表現形を致死遺伝子と定義する.

致死遺伝子を生じないためには, 遺伝的操作あるいは領域対象の解表現形どちらかを変更する必要がある.

以下では, 致死遺伝子を生じさせない 2 種類のコーディング手法を提案する.

4 コーディング手法の提案

4.1 コーディング手法 1

逆導出グラフを染色体として用いる. 染色体長 (ノード数) は最大値の範囲内で可変とする.

交叉は, グラフを 2 分割することにより行なう (図 2 参照). 分割により入力を失ったノードは分割グラフから削除する. 分割グラフの片方同士を互いに交換することで新しいグラフを作成する.

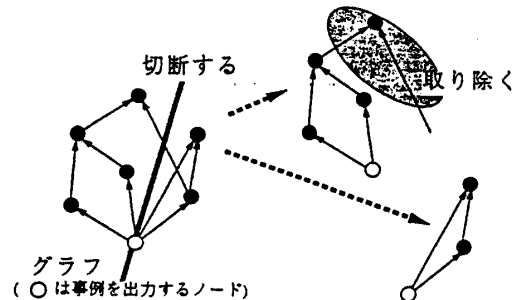


図 2: コーディング手法 1

突然変異では, 入出力条件を満たす範囲で, 任意にノードの変更 (追加あるいは削除) を行なう. 変更に伴い入力を失ったノードはグラフから削除する.

コーディング手法 1 ではノード数が多く, かつ複雑な接続関係を持つグラフの生成は困難である (交叉機能欠損).

4.2 コーディング手法 2

固定長のビット列を染色体とする. ビット列は L-system [4] の書き換え規則として解釈する (図 3 参照).

L-system とは, 細胞の分裂増殖する過程の計算モデルであり, 初期記号に対して書き換え規則が継続的に適用される. L-system により生成される記号列を decoder で解釈することによりグラフを生成する.

第 1 段階の記号で, グラフ各ノードのオペレータ種別を決定する. 第 2 段階の記号列で, 各ノードの入出力条件 (Horn 節の骨格) を決定する. 入出力条件が決定したら, 接続可能なノード候補を表にまとめる. そして, 第 3 段階の記号列に基づき, 候補中から接続

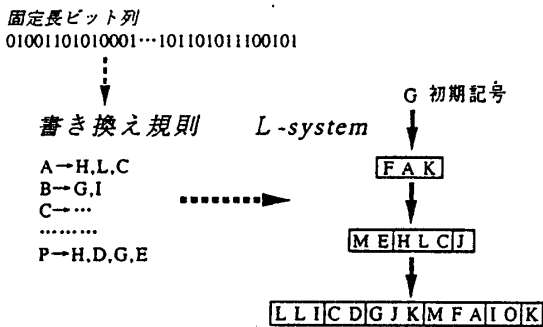


図 3: L-system

するノードを決定する。入力のでられないノードはグラフから削除する。極力、入力を得られるようにノードは接続する。交叉には、1点交叉を用いる。

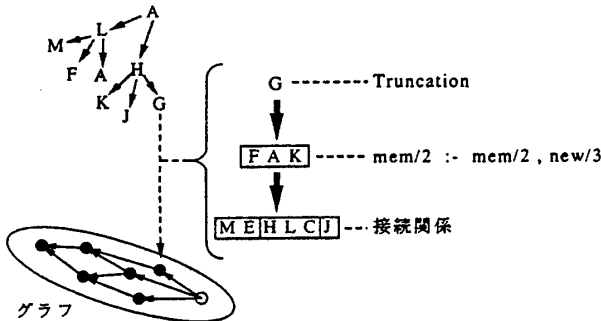


図 4: コーディング手法 2

コーディング手法 2 では、染色体間で共通する部分表現が、対応する Prolog プログラム間で共通する部分全部を再現できないので、有効な遺伝形質の継承が困難である（形質遺伝機能欠損）。

5 実験

2種類のコーディング手法は各々交叉機能欠損、形質遺伝機能欠損という欠点を持つ。以下では、計算機実験を通して、複雑な構造体に適したコーディング手法を検証する。

member 関数を対象として、2つのコーディング手法の学習能力を比較する。実験では、以下の式を Horn 節の選択基準（適応値）とした。

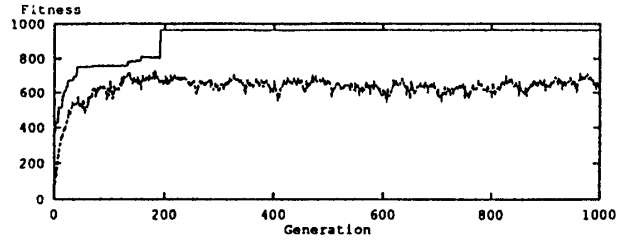
$$\text{適応値} = \text{定数} - \text{記述長} - \text{事例演繹時間}$$

個体数は 100、交叉率は共に 0.6、突然変異率は染色体長に応じ 0.05 と 0.005 の 2 種の値を用いた。世代交代数は 1000、淘汰圧は 50 世代周期で変化させた。

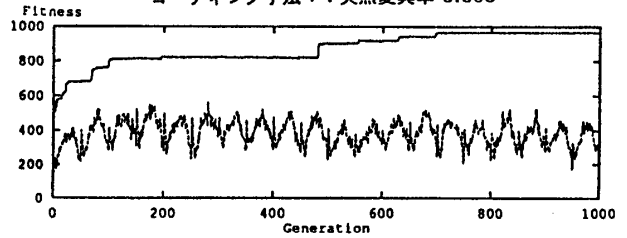
最優良解の適応値（実線）および適応値平均（破線）の世代交代に伴う変化を図 5 に示す。

4つの実験中、3つで目的とする解（関数）を得た。突然変異率が 0.05 である 2 種類の実験では、変化する適応値平均の上限がほぼ同じ値を示した。このことより、コーディング手法 2 にもある程度の形質遺伝機能があると考えられる。また、コーディング手法 1 の交叉で生成可能な解の範囲は限定され、member 関数の学習に適している。しかし実験では、2種類のコーディング手法とも同程度の速度で最適解に収束した。コー

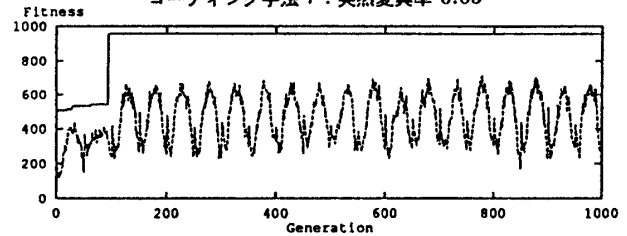
ディング手法 2 では、コーディング手法 1 と同等の探索空間特性が保存されていると考えられる。



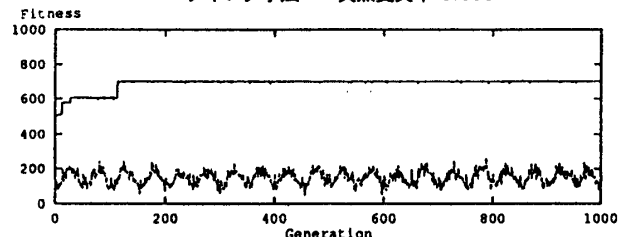
コーディング手法 1: 突然変異率 0.005



コーディング手法 1: 突然変異率 0.05



コーディング手法 2: 突然変異率 0.005



コーディング手法 2: 突然変異率 0.05

図 5: 実験結果

6 おわりに

複雑な構造体をコーディングする際の問題である致死遺伝子の定義を与え、解決法について考察した。また、計算機実験を通して、遺伝的アルゴリズムにおける個々の機能が果たす役割の重要性を比較した。

今後は適用事例を増やし、詳細な学習能力の比較を行なう予定である。

参考文献

- [1] Muggleton, S. and Buntine, W.: Machine Invention of First-order Predicates by inverting Resolution, *Proc of the 5th Int. Workshop on Machine Learning*, pp.339-352(1988).
- [2] 北野宏明: 遺伝的アルゴリズム, 産業図書 (1992).
- [3] 山本公洋 他: 逆導出原理と遺伝的アルゴリズムを用いた規則集合獲得手法 GA-CIGOL, 情報処理学会人工知能研究報告, 94-AI-94, pp1-8(1992).
- [4] 土居洋文: 生物のかたちづくり, サイエンス社 (1988).