589

GPを用いた原始帰納関数を実現するTRSの学習

桑山 高史　　河口 信夫　　稲垣 康善　(名古屋大学)

1 はじめに

Genetic Programming(GP)[1]は，木構造(Lispの S 式など)を遺伝子コードとして直接扱えるように，遺伝的アルゴリズム(GA)を拡張したものである．本稿では，原始帰納関数の定義に従って，再帰的関数を実現するTRS[3]を，GPによって学習する手法を提案する．文献[2]で再帰述語のGAによる学習が試みられているが，述語の構造に制約があり広い応用性に欠ける．この研究を含めて，従来のGPによる再帰的関数の学習には無限ループの処理の問題があったが，本手法では関数を実現するTRSの合流性・停止性が保証され，その問題は回避されている．また本手法では，シグネチャを考慮して関数を学習することができる．

2 原始帰納関数を実現するTRS

自然数上の原始帰納関数 F を実現するTRSは次のように表現できる．

$$\begin{cases} F(\mathrm{x}, Z) \to h(\mathrm{x}) \\ F(\mathrm{x}, S(y)) \to g(F(\mathrm{x}, y), \mathrm{x}, y) \end{cases}$$

Z は零関数，S は後者関数である．以下では，この定義を一般の項集合上の原始帰納関数に自然に拡張したものとして議論を進める．そこでは，便宜上，F を目的関数，h，g を構成する関数を基本関数と呼ぶ．また各関数はシグネチャを持つ．この時，このTRSの合流性・停止性が保証される．また，項は木構造で表現され，GPの母集団を構成する個体として扱える．

3 GPによるTRS学習システム

本システムの基本的なアルゴリズムはGPに従う．左辺の形式は基本的に変わらないので既知とし，基本関数の合成により $h(\mathrm{x})$ 及び $g(r, \mathrm{x}, y)$ を獲得することによって，目的関数を実現するTRSを学習する(ここで r は再帰部 $F(\mathrm{x}, y)$ を表す変数)．h，g の解候補の組を母集団中の個体として扱う．もちろんGPの交叉等の各操作は，各関数のシグネチャの整合がとれたときのみ許されるものとする．

入力： 基本関数記号の集合とそのシグネチャ，事例(目的関数の入力項と意図する答の項の組)の集合，GPのパラメータ

終了条件： 全ての事例を満たす個体の獲得または世代が最大世代数を超えた場合

アルゴリズム：

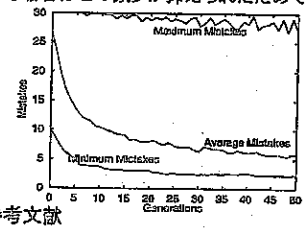
1. ランダムな個体の構成と母集団の形成
   "個体はそれぞれ基本関数記号の集合及び h, g で用いる変数の集合からランダムに構成する．"
2. 適応度の計算，終了条件のチェック
   "事例に対し，個体が定義するTRSを実行させ，適応度を求める．この時，個体 i の入力に対する誤答数を $m(i)$ とすると，適応度 $f(i)$ は以下の式で定義される．"
   $$f(i) = \frac{1}{1 + m(i)}$$
3. 交叉・突然変異などの遺伝的操作
   "適応度に比例した確率で個体のペアを抜き出し(ルーレット戦略)，h または g 同士で交叉を行なう．突然変異は個体をランダムに選んで施す．"
4. 2にもどる

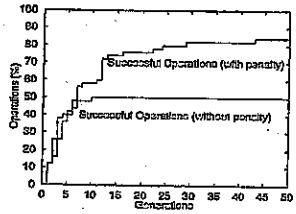
4 実験及び考察

本システムを用いて階乗関数，リストの要素であるか否かを判定するmember関数等の学習に成功した．学習実験により獲得したmember関数を以下に示し，実験の各種データを下図に示す．

$$\begin{cases} F(x, nil) \to false \\ F(x, cons(y, z)) \to if(eq(x, y), true, F(x, z)) \end{cases}$$

この実験では，基本関数として if，eq，$true$，$false$，$cons$，nil を与えている．下の左図は，事例数30，母集団サイズ50，最大世代数50でGPを50回実行した際の，各世代の最小誤答数，最大誤答数，平均誤答数の全体の実行における平均値のグラフである．なおアルゴリズムの2において，再帰部 r が g 上に出現しない個体に対してペナルティを設け適応度を下げた．この図から母集団全体が学習していく様子が確認できる．また右図は，再帰部に対するペナルティがある場合とない場合で実験を行なった際の，各世代での目的関数獲得率の累積(%)を示している．初期に目的関数を獲得しやすいのは，シグネチャの明示によって意味のある個体が出現しやすいためであると考えられる．ペナルティがある場合は全実行の84%で学習に成功したが，ない場合は50%に留った．これは，ペナルティがない場合は世代が進むにつれて再帰部を持つ個体の数が減少しやすい傾向にあったが，ある場合はこの減少が抑えられたためである．





参考文献

[1] Koza, J.: Genetic programming : A paradigm for genetically breeding population of computer programs to solve problems, Report No. STAN-CS-90-1314, Dept. of Computer Science, Stanford Univ., 1990
[2] 岸, 小井土, 柴田：GAを用いた多地型変数配置による述語の帰納的学習について, 人工知能学会誌, Vol.11, No.4, pp. 112-122, 1996
[3] 二木, 外山：項書換え型計算モデルとその応用, 情報処理, Vol. 24, No. 2, pp. 133-146, 1982