

文章编号:1001-9081(2005)09-1978-04

基于基因表达式编程的知识发现的三项新技术 ——转基因,重叠基因表达和回溯进化

唐常杰,彭京,张欢,钟义啸
(四川大学计算机学院,四川成都 610065)
(tangchangjie@cs.scu.edu.cn)

摘要:介绍了在研发基于基因表达式编程(GEP)的知识发现的三项新技术,包括:(a)转基因技术,通过注入转基因,引导进化方向,控制知识发现过程;(b)重叠基因表达,借鉴生物基因片段重叠表达,引入重叠基因概念,节约了表达空间;(c)回溯进化,借鉴生物“返祖现象”,引入回溯检查点概念和可回溯 GEP 算法,设计了等比递增检查点序列和加速递增检查点序列,约束回溯过程。实验表明,三项技术在一定的场合下分别提高了知识发现的性能 1 至 2 个数量级。

关键词:知识发现;基因表达式编程;转基因;重叠基因表达;回溯进化

中图分类号: TP311 **文献标识码:** A

Three new techniques for knowledge discover by gene expression programming — transgene, overlapped gene expression and backtracking evolution

TANG Chang-jie, PENG Jing, ZHANG Huan, ZHONG Yi-xiao

(School of Computer Science and Engineering, Sichuan University, Chengdu Sichuan 610065, China)

Abstract: Three new technologies were introduced by the authors in the past year, i. e.: (a) TranGene technique. By injection gene segment, it guides the evolution direction, controls knowledge discover process. (b) Overlapped gene expression. It borrows the idea of overlap gene expression from biological study, introduces overlapped gene expression, and saves space for gene expression. (c) Backtracking evolution. It comes from atavism in biology and proposes the concept of backtracking GEP algorithms, designing geometric proportion increased checkpoint sequence and accelerated increased checkpoint sequence to restrict the backtracking process. Experiments show that all three techniques respectively boost the performance of GEP by one or two magnitudes.

Key words: knowledge discover; gene expression programming (GEP); transgene; overlapped gene expression; backtracking evolution

0 引言

基因表达式编程(Gene Expression Programming, GEP)是一种知识发现的仿生计算新技术。GEP 模拟生物进化过程对问题进行优化求解。以适者生存原则消解不良结构,以遗传实现继承,以变异求得发展。GEP 融合了遗传算法(GA)和遗传编程(GP)的优点,通过简单紧凑的编码解决复杂应用问题。定长线性编码使进化过程简单有效,解码为表达式树使进化结果清晰明了,有很强的解决实际问题的能力。Candida Ferreira 于 2001 年 12 月发表 GEP 的首批研究成果^[1],受到学术界的高度关注。一批研究成果和软件相继问世。GEP 目前广泛应用于函数发现,时间序列预测,分类问题等领域。国内新成果也不断出现。文献[2]将 GEP 应用于数据挖掘的典型任务——谓词关联规则挖掘,并从理论上证明了 GEP 中基因编码的有效性;文献[3]对函数发现的特点进行分析,提出任意维定义域上的一致表达式挖掘(UEM)和分域表达式挖掘

(MEM)算法,从理论上给出 MEM 成功率的证明;文献[4]把函数挖掘的目标引申到函数集上,提出描述能力更强的函数挖掘对象——频繁函数集、可配置频繁函数集挖掘算法以及用户制导可配置频繁函数集挖掘算法。文献[5]提出了 GEP 的弱适应模型和带内集、带外集概念,设计了在弱适应模型下基于相对误差计算适应度的算法。

本文介绍作者在最近一年中开发的基于基因表达式编程的知识发现的三项新技术,即转基因技术、重叠基因表达技术和可回溯 GEP 技术。为了简明表述思想和成果,文中将避开复杂形式化定义和深奥的公式。

1 GEP 核心概念

染色体(Chromosome)和表达式树(Expression Tree, ET)是 GEP 技术中最重要的概念。染色体作为承载遗传信息的基因型实体,参与遗传操作;表达式树作为信息的表现型,表达遗传实体中的信息编码。染色体和表达式树结构简单清

收稿日期:2005-06-26;修订日期:2005-07-12 基金项目:国家自然科学基金资助项目(60473071);高等学校博士学科点专项科研基金 SRFDP 资助项目(20020610007);四川省青年软件创新工程资助项目(350)

作者简介:唐常杰,男,教授,博士生导师,主要研究方向:数据库与知识工程;彭京(1973-),男,博士研究生,主要研究方向:数据库与进化计算;张欢(1977-),男,硕士研究生,主要研究方向:数据库与知识工程;钟义啸(1980-),女,硕士研究生,主要研究方向:数据库与知识工程。

晰,通过简单的编码和解码规则可无歧义地互化。GEP 将这两者分别作为独立个体,对 GA 和 GP 的优点分别加以继承,使遗传操作易于实施,结果方便表达。

染色体由若干个基因(Gene)通过连接运算符连接组成。Gene 由头部(head)和尾部(tail)组成,head 包含了函数(Functions)和终结符(Terminals),tail 只含有 Terminals。设头部长度为 h ,尾部长度为 t ,函数中的最大目数为 n ,例如:如果基因中有 FunctionSet $\{+, -, \times, /\}$,那么 $n = 2$,因为在 FunctionSet 中函数最大的目数为 n ,则良好定义的表达式满足关系 $t = h(n - 1) + 1$,我们证明了良好定义的表达式集合并在遗传操作下封闭^[2]。

GEP 先将个体编码为固定长度的线性串,待进行优化求解时再对操作对象进行编码形成基因组。其编码规则可以简单描述为:将表达式根据其语义表示为表达式树(ET),然后从上到下,由左至右按层次遍历 ET 中的节点,得到的符号序列即为基因编码的有效部分。一些尾部结点可能不出现在表达式树中,这些冗余结点容纳将来的遗传操作可能产生的结构变化留下了空间。

每一代进化结果经适应度函数评价,高适应度个体被保留下来,并有更高机会繁殖后代,如此循环往复,直到出现满意解或达到预定进化代数,算法停止。文献[4,5]中描述了 GEP 基本算法的基本框架。

2 基于转基因的 GEP 技术

2.1 GEP 中转基因技术的基本思想

Candida F 原创的 GEP 进化过程是自由放任的。一旦程序开始运行,整个种群就处于失控状态,人们只能被动接受计算机程序在连续进化若干代之后给出的结果,多代进化积累的优良基因有可能在一次不良变异中毁于一旦,如果能够将其“好基因”定性、分离并保存,在进化中适当注入,则有希望加快收敛速度。例如衰减因子 $(1 - x + x^2/2 - x^3/6)$ (模拟 e^{-x}) 在变异中被拆散为 20 个基本符号参加变异,使得通过多代进化选择固定下来的优良性质在变异阶段被打乱,如果能够将其作为一个“大基因”封装起来,以基因为单位进行遗传和变异,则有希望提高基因表达式编程发现知识的速度。

受现代生物工程转基因技术的启发,我们引入了基因注入(Gene Injection)和转基因技术来干预进化过程,在一定程度上控制种群发展方向,通过自然选择与人工选择的共同作用,在较短的时间内(相比自然选择所花费的上百万年时间而言)得到性能优良的品种。结合特定问题,在启发性知识引导下向规则强制注入基因组成新规则或公式,具体方法可表现为事先人为对函数表达式的部分内容进行估计。

2.2 生物工程中的转基因技术

生物工程转基因技术旨在把人工分离和修饰过的基因导入到目的生物体的基因组中,从而达到“拔苗助长”、改造生物的目的。当转入的基因整合到染色体上或基因组中以后,这些基因就会与寄主生物的遗传物质一起向子代传递,并产生应有的生物学功能。生物学家将所需要的基因进行定位,分离克隆,然后再将这个目的基因,通过载体转移到目标生物品种中去(细节参见文献[7])。要点有:(1)分离目标基因或 DNA 片段;(2)体外 DNA 重组,将外源 DNA 片段连接到能自我复制又带有选择标记的载体上重组的 DNA 转移到受

体细胞;(3)筛选目的 DNA 的受体细胞并克隆;(4)目的基因克隆到表达载体上,转入受体细胞进行表达,产生人们需要的基因产物。

2.3 转基因 GEP 进化的基本方法

GEP 转基因技术的核心是注入外源基因片段,促进种群向目的基因的进化。GEP 不能预知目标函数的特性,外源基因依赖于进化结果。实践表明,进化一定时间后,优秀个体中某些基因中可能进化出比较好的结构,这些基因可以暂时保留下来进行培养。我们提出基因分离和基因肢解的方法来保留这种“优秀”结构。文献[6]描述了一系列的概念和算法。限于篇幅和复杂的背景知识,这里从目的、思想方面进行非形式化的介绍。主要算法如下:

- 基因分离算法 M_divide_gene, 将一条染色体中的各个单一基因(G_single)分离出来。
- 基因肢解算法 M_dismember_gene。G_single 在抛弃部分函数和终结符后肢解为若干 G_sengment。
- 单一基因进化(Single Gene Evolution)算法。获取前阶段成果。被选出的染色体中含优良基因,但不能确定编号,将这些基因分散到各个种群中独立进化,希望可以从中找到优良基因的痕迹。
- 基因片段进化算法(Gene Segment Evolution)。从基因中分离出基因片段,以基因片段为单位分别进化。
- 筛选基因算法(FGSE: Filter Gene for GSE)。筛选出有价值的染色体为以后的进化做准备。
- GEP 转基因进化算法 TGEP 思想:之前先对目标表达式的部分内容进行估计;对进化过程中出现的基因或基因片段筛选,将选出的部分分别进化。

2.4 转基因实验

实验平台为: Intel Celeron III 1.7G, 256M 内存, Windows XP Professional, VC++ 6.0。实验采用不同的参数分别对 GEP 和 TGEP 作了两组对比实验,每组实验中,两种方法各自进行 10 次实验。详细数据可参见[6],我们选择文献[7]中的 Schaffer 函数 F6 作为实验对象。实验中的 x_1, x_2 采用随机化方法得到, y 由公式计算得到,总共生成 1000 组数据:

$$f_6(x_1, x_2) = 0.5 = \frac{\sin^2 \sqrt{x_1^2 + x_2^2} - 0.5}{[1.0 + 0.001(x_1^2 + x_2^2)]^2} - 100 \leq x_i \leq 100 \quad (i = 1, 2)$$

图 1 给出了其中一个关于 GEP 与 TGEP 对 F6 的进化效果比较。为模拟真实的数据挖掘环境,未对参数人工干预,实验结果表明:与传统 GEP 相比, TGEP 有一定优势。在 10 次实验中, TGEP 有 4 次的结果都优于 GEP 的最佳结果,平均适应度 TGEP 也比 GEP 高了 0.08107。

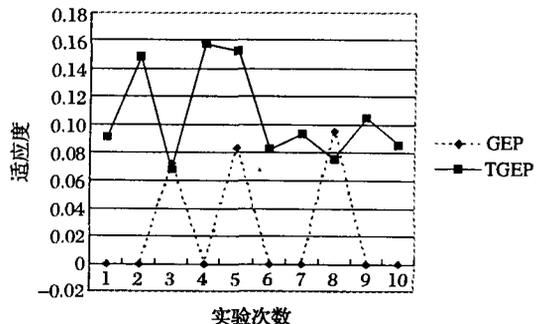


图 1 GEP 与 TGEP 对 F6 的进化效果比较

3 重叠基因技术

3.1 生物重叠基因的启示

生物学家发现,一定条件下,组成基因的核苷酸序列有些可重复,即存在重叠基因(overlapping gene)。生物界认为重叠基因能经济和有效地利用 DNA 的遗传信息量,“节约”碱基,更重要的是能对基因表达起调控作用。受此启发,我们提出了新的算法——基于重叠表达的多基因进化算法 MEOE (multi-gene evolutionary algorithm based on overlapped expression)。它有如下特点:

- 1) 基因的片段在一定条件下可重复表达。
- 2) MEOE 的表达空间效率优于其他算法。
- 3) 无需对基因或染色体内容约束。而 GP 与 GEP 必须对基因格式作限制,如 GEP 中尾部不允许出现任何的操作算子。减少限制使得 MEOE 算法效率高,实验表明,MEOE 的速度为 GEP 的 2.5~9.4 倍。
- 4) 同 GEP 相比较,MEOE 在函数发现的成功率方面有较大幅度提高。

3.2 定义和编码方法

GEP 为保证表达式合法性,分基因为头部和尾部,头部含函数 F 和终结符 T ,尾部只含 T ,头尾长度必须满足公式 $t = h(n - 1) + 1$,文献[2]证明了满足 $t = h(n - 1) + 1$ 的基因(良性基因)集合在遗传操作下封闭。新的 MEOE 做了改革,取消头尾概念, F 和 T 中的元素可出现在基因的任何位置,因此 MEOE 的编码具有 GA 的简单性。另一方面,同 GEP 算法一样,MEOE 可以根据基因编码翻译为唯一对应的表达式树,具体翻译规则如下:

- 1) 依次读取基因中的每个元素。
- 2) 如果读取的元素属于 T ,则将此作为表达式树的叶子节点。
- 3) 如果读取的元素属于 F ,则将此元素作为表达式树的非叶节点,它的子树数目为函数的参数个数,每个子树的根节点依次使用此元素的直接连续后继元素。如果到达基因的末尾,则自动填充 T 中的元素。

新的编码方式使得 MEOE 具有 GA 的简单性,无需对基因元素做出任何限制;另一方面,MEOE 同样可以形成表达式树,完成从基因型到表现型的映射,从而为解决复杂函数发现问题打下基础。

3.3 重叠基因技术的成果

MEOE 算法模拟自然界的生物进化,按照“物竞天择,适者生存”的原则对由若干个体构成的种群 P 实施选择,重组和变异等遗传操作,使种群一代代地进化,从中寻出最优的个体,从而得到问题的最终解。

在文献[8]中给出了一系列算法和定理。限于篇幅,这里只能介绍其中最重要的结果。这些结果包括:

(a) 多基因表达空间定理。指出单基因个体 I 的最大表达空间为:

$$MAX_m(D_I) = k \times \frac{2}{\sqrt{5}} \left[\left(\frac{1 + \sqrt{5}}{2} \right)^{\frac{m}{k} + 2} - \left(\frac{1 - \sqrt{5}}{2} \right)^{\frac{m}{k} + 2} \right] - 1$$

其中 m 为染色体长度, k 为基因个数。

(b) MEOE 算法的等价基因型定性定理。即设

$$f(x_1, x_2, \dots, x_n) = \sum_{i=1}^k x_1^{p_{i1}} x_2^{p_{i2}} \dots x_n^{p_{in}}$$

其中 x_i 为变量, k 为非 0 正整数, p_{i1}, \dots, p_{in} 为非 0 整数。则必定存在一个 MEOE 算法的基因型 E ,使得 E 对应的表达式树与上述公式等价。

3.4 三重叠基因技术的实验结果

文献[8]作了 4 组实验。限于篇幅,这里介绍其中实验 1 的结果。考虑二元函数发现问题,根据以下公式:

$$Z = X^5 + 3 * X * Y$$

随机产生 20 个实数,数据范围为 $[-3, 3]$, $M = 10000$ (即规范因子),分别对 MEOE、GEP 取基因长度从 9~23,分别运行 100 次。MEOE 算法在不同基因长度的平均进化辈数。最大进化辈数和最小进化辈数如图 2 所示。二者耗时间如图 3 所示。详尽的实验证明,MEOE 算法在速度上是 GEP 算法的 2.5~9.4 倍。在高次函数发现问题上 MEOE 算法的成功率比 GEP 提高至少一个数量级。另外,通过实验证明了基于密度的概率选择函数在高次函数发现问题上具有一定优势,具体细节可参见文献[8]。

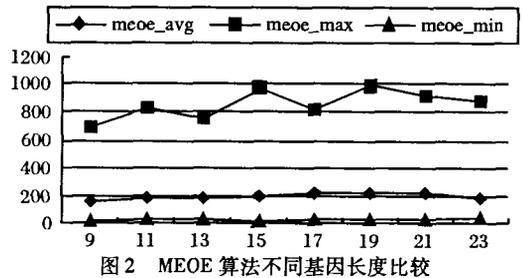


图 2 MEOE 算法不同基因长度比较

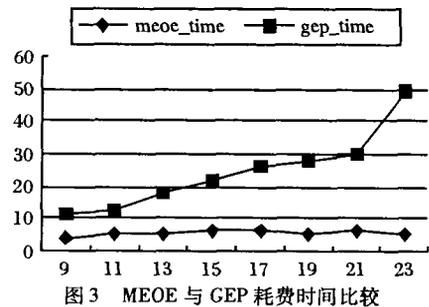


图 3 MEOE 与 GEP 耗费时间比较

4 可回溯 GEP

4.1 返祖现象的启示

GEP 进化过程达到一定的代数以后,平均适应度增大,群体多样性变差,可能陷入局部最优分支,失去获得全局最优解的机会^[7],类似于生物界的早熟现象。“返祖现象”给出了破解的灵感。现代遗传学认为返祖现象的原因是:(a) 已分开的,决定某种祖先性状的基因通过杂交等原因重组,使祖先性状重现;(b) 祖先性状基因在进化过程中早被阻遏蛋白封闭,某种原因使阻遏蛋白脱落,被封闭的基因恢复活性,重现祖先性状。这说明进化过程并非不可逆。为改善未成熟收敛难题,我们引入可回溯 GEP,使传统 GEP 能以退为进,修正方向,重新开始。

4.2 可回溯 GEP 的技术要点

(a) 进化以进化代数为坐标,设立回溯检查点序列 (Backtrace Checkpoint Sequence, BCS),设置堆栈。

(b) 当进化到达某一个检查点时,检查当前最大适应度,与序列中上一回溯检查点的最大适应度值比较。

(c) 若两回溯检查点之间最大适应度有提高,则认为进

化过程有效前进,将当前种群入栈保存。

(d) 否则,进化可能陷入停滞,放弃该进化方向,从栈中弹出上一个回溯检查点保存的种群,作进化出发点。

GEP 基于随机搜索,重新进化极可能进入另一进化分支,而不完全重复上一次进化路径,从而跳出了早熟。

4.3 可回溯 GEP 中的重要新概念

(1) 回溯检查点。预设的检查是否应该回溯的进化代数号,回溯检查点对应的适应度和栈结点。

实践中,观察到收敛速度的非线性特征。进化初期初始种群的多样性好,收敛较快;急剧收敛后,形成“近亲繁殖”,收敛速度渐缓。为此引入了:

(2) 等比递增检查点序列和加速递增检查点序列。分别模拟等比级数或加速运动距离级数的回溯检查点序列。

(3) 退化因子 α 。它是约束回溯机制的正数,在需从检查点 g_{i+1} 回溯到检查点 g_i 时,回溯后 α 应使 g_i 点的种群组成为 $\alpha * P_{g_i} + (1 - \alpha) * P_{g_{i+1}}$ 。

(4) 按比例回溯策略。在回溯中引入退化因子 α ,用以约束回溯过程。当 $\alpha = 1$ 时,进化过程为可回溯 GEP;当 $\alpha = 0.5$ 时,称该方法为半回溯 GEP;当 $\alpha = 0$ 时,该方法退化为传统 GEP。

4.4 可回溯 GEP 的实验

在文献[9]中给出了两组实验,实验1的数据来自文献[10]。Candida 曾用该函数验证 GEP 在五维参数空间中拟合数据的能力。用函数随机生成了 50 个数据作为进化环境。由于函数较复杂,维数比较高,回溯检查点序列采用了等比递增检查点序列(GPICS)。规定适应度极值的 80% 以上作为实验成功的标志,则传统 GEP 方法和可回溯 GEP 的成功率对比如表 1 所示:可回溯 GEP 将成功率提高了一倍左右。平均适应度分别为 2 888.43 和 4 309.36,新算法将平均适应度提高了 49.2%。

表 1 实验 1 的成功率比较

	传统 GEP	PGEPBS
记录条数	10	10
成功记录编号	2,5,6,8	2~10
成功率	40%	90%

实验 2 的数据来自文献[11]。该组实验数据较上一组简单,采用加速递增检查点序列(AICS)。规定该组实验中以达到适应度极值的 84% 以上作为实验成功的标志,则传统 GEP 方法和可回溯 GEP 的成功率对比如表 2 所示:可回溯 GEP 将实验的成功率提高了 5 倍。平均适应度分别为 3 123.33和3 526.33,新算法将平均适应度提高了 12.5%。

表 2 实验 2 的成功率对比表

	传统 GEP	PGEPBS
记录条数	10	10
成功记录编号	3,4	1~10
成功率	20%	100%

实验结果还表明在相同的进化代数之内可回溯 GEP 可有效地避免早熟,较传统算法更易获得全局最优解。图 4 显

示了实验 2 中两种算法对数据的拟合程度,选择两组实验中适应度最大的函数,使用数据编号作为横坐标,纵坐标为数据的值。从图中可以看出,新算法得到值更接近真实值,曲线的分布也更加合理。

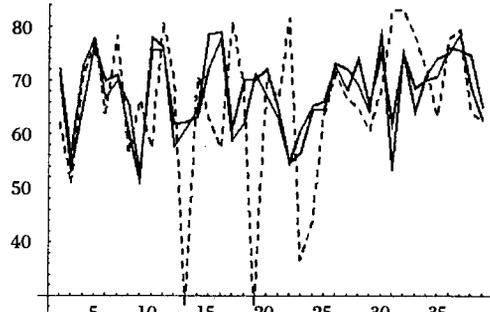


图 4 两种算法对数据的拟合对比

5 结语

GEP 以其编码简单,通用性强的优势在符号回归、分类和时间序列预测中广泛应用,成为了一个非常有力的数据挖掘工具。在过去的一年中,我们针对在应用实践中发现的目前 GEP 的缺陷,研发了关于基于基因表达式编程的知识发现的三项技术,即:(a) 转基因技术,通过注入转基因,引导进化方向,控制知识发现过程;(b) 重叠基因表达,借鉴生物基因片段重叠表达,引入重叠基因概念,节约了表达空间;(c) 回溯进化。借鉴生物“返祖现象”,引入 GEP 回溯算法、回溯检查点、设计等比递增检查点序列和加速递增检查点序列,约束回溯过程。实验表明,三项技术在一定的场合下分别提高了知识发现的性能 1 至 2 个数量级。

参考文献:

[1] FERREIRA C. Gene Expression Programming: A New Adaptive Algorithm for Solving Problems[J]. Complex Systems, 2001, 13(2): 87 - 129.

[2] ZUO J, TANG CJ, LI C, et al. Time Series Prediction based on Gene Expression Programming[M]. LNCS (Lecture Notes In Computer science). Springer Verlag Berling Heidelberg 2004. 55 - 64.

[3] 黄晓冬, 唐常杰, 李智, 等. 基于基因表达式编程挖掘函数关系[J]. 软件学报, 2004, 15(suppl.): 96 - 105.

[4] 贾晓斌, 唐常杰, 钟义啸, 等. 频繁函数集的可配置挖掘算法[J]. 计算机研究与发展, 2004, 41(Suppl.): 240 - 245.

[5] 段磊, 唐常杰, 左劭, 等. 基于基因表达式编程的抗噪声数据的函数挖掘方法[J]. 计算机研究与发展, 2004, 41(10): 1684 - 1689.

[6] 张欢, 唐常杰, 余弦, 等. 基于转基因技术的基因表达式编程[EB/OL]. 中国科技论文在线(教育部). <http://www.paper.edu.cn>.

[7] 周明, 孙树栋. 遗传算法原理及应用[M]. 第 1 版. 北京: 国防工业出版社, 1999. 123 - 166.

[8] 彭京, 唐常杰, 元昌安, 等. 基于重叠表达的多基因进化算法[EB/OL]. 中国科技论文在线(教育部). <http://www.paper.edu.cn>.

[9] 钟义啸, 唐常杰, 陈宇, 等. 提高基因表达式编程发现知识效率的回溯策略[EB/OL]. 中国科技论文在线(教育部). <http://www.paper.edu.cn>.

[10] <http://www.gene-expression-programming.com/gep/GepBook/Chapter4/Section1/SS2.htm> [EB/OL].

[11] http://www.amstat.org/publications/jse/jse_data_archive.htm [EB/OL].