应用禁忌基因表达式编程提高模型精度

张雪东[1],饶 元[2,3],元昌安[3],赵传信[2,4]
ZHANG Xue-dong[1],RAO Yuan[2,3],YUAN Chang-an[3],ZHAO Chuan-xin[2,4]

1.安徽财经大学 信息工程学院,安徽 蚌埠 233041
2.南京邮电大学 计算机学院,南京 210003
3.广西师范学院 软件研究所,南宁 530001
4.安徽师范大学 计算机系,安徽 芜湖 241000
1.School of Information Engineering,Anhui University of Finance & Economy,Bengbu,Anhui 233041,China
2.College of Computer,Nanjing University of Posts and Telecommunications,Nanjing 210003,China
3.Institute of Software,Guangxi Teachers Education University,Nanning 530001,China
4.Institute of Computer,Anhui Normal University,Wuhu,Anhui 241000,China
E-mail:zxd_01@163.com

ZHANG Xue-dong,RAO Yuan,YUAN Chang-an,et al.Improving model accuracy using Gene Expression Programming and Tabu Search.Computer Engineering and Applications,2009,45(28):35-38.

Abstract: To improve model accuracy,PTS-GEP (Gene Expression Programming Based on Parallel Tabu Search) is proposed.In PTS-GEP,tabu search is introduced to improve GEP's local search ability.The research conducts two experiments over the data from previously reported research and compares the results to two other algorithms namely simple GEP,UC-GEP.The results demonstrate the optimal performance of PTS-GEP in model accuracy.

Key words: Gene Expression Programming(GEP);tabu search;model accuracy

摘 要:为提高建模精度,将禁忌搜索引入到基因表达式编程的遗传操作中,改善基因表达式编程的局部搜索能力,提出了并行禁忌基因表达式编程算法 PTS-GEP(Gene Expression Programming Based on Parallel Tabu Search)。通过两组实验比较算法的性能,实验结果表明,PTS-GEP 挖掘出的模型精度优于 GEP、UC-GEP 算法。

关键词:基因表达式编程;禁忌搜索;模型精度
DOI:10.3778/j.issn.1002-8331.2009.28.010 **文章编号**:1002-8331(2009)28-0035-04 **文献标识码**:A **中图分类号**:TP301

1 引言

基因表达式编程 GEP[1](Gene Expression Programming)的灵感来源于生物的进化,具有强大的全局搜索能力。以编码简单、收敛速度快、对未知数据建模能力强等优点,GEP 已成为国际上的研究热点。为避免 GEP 早熟,Ferreira.C 开创了插串等新的遗传操作算子[2-3],在一定程度上克服了 GEP 易陷入局部最优的不足,但实际问题中仍表现为算法的局部搜索能力差、存在未成熟收敛和随机漫游等现象[4-7],算法的收敛性能差。如何改善 GEP 的局部搜索能力,提高模型发现的精度,使其更好地应用于实际问题,是各国学者一直探索的一个主要课题,如 VPS-GEP[5]、RG-GEPSA[6]、UC-GEP[7]等相继被提出。

禁忌搜索 TS(Tabu Search)具有很强的局部搜索能力,但它是串行结构且依赖于初始解,而 GEP 是并行结构,且具有较强的全局搜索能力。若将 TS 嵌到 GEP 里,GEP 就会弥补 TS 对初始解的依赖的缺陷,TS 则可以弥补 GEP 局部搜索能力不强的缺陷。该文采用扬长避短的策略,将 TS 融合到 GEP 的遗传操作中,提出了并行禁忌基因表达式编程算法 PTS-GEP(Gene Expression Programming Based on Parallel Tabu Search)。实验结果表明,PTS-GEP 比传统 GEP 更稳定,寻优能力更强,挖掘出的模型精度更高。

2 相关工作

2.1 GEP 简介

GEP 是遗传计算家族的革命性的新成员,是借鉴生物遗传的基因表达规律提出的知识发现新技术。GEP 作为遗传算法 GA(Genetic Algorithm)和遗传编程 GP(Genetic Programming)

基金项目:国家自然科学基金(the National Natural Science Foundation of China under Grant No.60763012);安徽省高等学校自然科学研究项目(No.KJ2009B125Z);安徽财经大学青年科研项目(No.ACKYQ0921)。

作者简介:张雪东(1980-),男,讲师,主要研究领域为智能算法;饶元(1982-),男,博士研究生,主要研究领域为数据挖掘、移动 Agent 等;元昌安(1964-),男,博士,教授,主要研究领域为数据库与知识工程;赵传信(1977-),男,博士研究生,讲师,主要研究领域为无线网络,最优化算法等。

收稿日期:2009-03-24 **修回日期**:2009-05-18

的继承和发展,综合了二者的优点,具有更强的解决问题的能力。它们最本质的区别在于:在 GA 中个体由固定长度的线性串(染色体)来表示,在 GP 中个体由不同大小和形状的非线性实体(解析树)所表示;而 GEP 将个体先编码为固定长度的线性串再表示成大小、形状都不同的非线性实体。这样,GEP 就克服了 GA 损失功能复杂性的可能和 GP 难以再产生新的变化的可能。

个体在 GEP 中又称为染色体,染色体是由基因通过连接运算符连接组成的。基因由头部和尾部组成,头部包含了变量集中的变量和函数集中的函数,而尾部只包含了变量集中的变量。基因头尾长度满足下式:

$t=h(m-1)+1$

其中,h 为头部长度,m 为运算符最大目数,t 为尾部长度。

GEP 步骤如下:(1)初始化种群;(2)计算个体的适应度函数,若不符合结束条件,继续下一步,否则结束;(3)保留最好个体;(4)选择操作;(5)变异;(6)插串操作(IS 插串、RIS 插串、Gene 插串);(7)重组(1-点重组、2-点重组、基因重组);(8)转到(2)。

Ferreira.C 在提出了 GEP 的概念后,对 GEP 做了大量的研究[2-3]。此外,新的 GEP 研究成果不断被推出,例如:将 GEP 应用于时间序列分析[4],基于 GEP 的智能模型库系统的实现[8],基于网格的 GEP 函数挖掘算法[9],基于 GEP 和神经网络的属性约简分类算法[10],基于 IP 和 GEP 算法的股票预测[11]等等。

2.2 TS 简介

TS 是对人类思维过程本身的一种模拟,它通过对一些局部最优解的禁忌达到接纳一部分较差解,从而跳出局部搜索的目的。

TS 步骤如下:

步骤 1 设置算法参数,随机产生初始可行解 $x\in X$,初始化禁忌表 $T=\phi$ 等参数;

步骤 2 若满足停止规则,停止计算;否则,根据选择策略从 x 的邻域中选出满足禁忌要求的候选集;在候选集中选一个评价值最佳的解 x',令 $x=x'$;更新禁忌表 T,重复步骤 2。

TS 具有很强的局部搜索能力,但它是串行的并且得到的解依赖于初始解的质量。该文对其进行了改进,提出了基于多初始解的并行禁忌搜索算法 PTS(Multiple Population-based Parallel Tabu Search),再将它应用到 GEP 的遗传操作中,将传统 TS 的串行搜索变为并行搜索,提高了 PTS 的鲁棒性。

3 算法描述

3.1 相关定义

在介绍整个算法之前,首先给出几个定义。

假设目标适应度值为 $fmax$,第 k 代不重复的个体数(与 $k-1$ 代相比)为 m,最优适应度为 $fmax(k)$,N 为种群大小。

定义 1(第 k 代种群目标距离度)令 d 为种群目标距离度,则 $d=(fmax-fmax(k))/fmax$。d 表示当前最优个体与目标个体相差程度,其值越小,表示离目标值越近。

定义 2(第 k 代种群多样性指数)令 v 表示当前群体中个体多样性程度,则 $v=m/N$。v 表示当前群体中个体多样性程度,其值越小,表示当前种群中个体多样性程度越低。

定义 3(第 k 代种群早熟度)第 k 代群体的早熟度为:$prem(k)=d*v$,该参数能有效反应第 k 代种群的当前状态。

关于 $prem(k)$ 的物理意义将在 3.4 节的算法分析部分深入分析。

定义 4(早熟种群)给定早熟度阈值 α,若 $prem(k)<\alpha$,则称第 k 代种群为早熟种群。

PTS-GEP 算法需要多次调用 PTS 算法,如果每次调用 PTS 算法都新构造一个禁忌表,有违算法初衷,失去意义。PTS-GEP 算法在一代(该代满足调用条件时)进化中采用同一个禁忌表,将这个禁忌表定义为 GEP 禁忌表。

定义 5(GEP 禁忌表)GEP 禁忌表 T 由禁忌表项组成,禁忌表项由二元组$<Lim_{lower},Lim_{upper}>$定义,即 $T=\{<Lim_{lower},Lim_{upper}>\}$,其中 Lim_{lower}、Lim_{upper} 分别代表 GEP 禁忌表项的下限和上限。

适应度值通常是浮点数,为了实现有效禁忌,将被禁忌适应度函数值确定为目标禁忌值的一个上下浮动区间,即 $Lim_{lower}=TabuValue-fmax/1\,000$,$Lim_{upper}=TabuValue+fmax/1\,000$。

定义 6(破禁指数)PTS 一次运行中,破禁被执行的次数 u。

定义 7(破禁阈值)给定阈值 β,若 $u>\beta$,PTS 算法终止,则称阈值 β 为破禁阈值。

3.2 基于多初始解的并行禁忌搜索算法

算法 1 Multiple Population-based Parallel Tabu Search (PTS)

PTS 算法采用两种邻域搜索函数,即插串操作和重组操作。插串操作有助于 PTS 算法搜索新的邻域;重组操作有助于染色体之间不断交换搜索到的信息,加强了对有效模式的搜索。两种邻域函数的互补,使算法同时具备高效的全空间探索能力和局部优化能力。整个种群采用一个基于目标值的禁忌表,保证了种群的多样性,有效地提高了算法的效率。

PTS 算法步骤如下:

输入:n 个老个体,插串率 Pis,重组率 Prc,GEP 禁忌表 T,破禁阈值 β。

输出:n 个新个体。

Begin

Step1:$u=0$。

Step2:最优适值函数 $F(x^*)=F(x)$,渴望水平函数 $A(s,x)=F(x^*)$。

Step3:若 $u>\beta$,返回 n 个新个体,否则按插串操作和重组操作两种方式产生个体的邻域解,在老个体和新个体共 $2n$ 个个体范围内确定候选解。

Step4:选择最佳的非禁忌解或满足特赦准则的解,并按定义 6 计算破禁指数 u。

Step4.1:选择适应度值高于渴望水平的新个体,若其在禁忌表中则破禁。

Step4.2:若不足 n 个个体,从剩下的候选解中选择不在禁忌表中的个体(新个体优先)。

Step4.3:若不足 n 个个体,从剩下的候选解中依次按适应度值从大到小的顺序选取个体,直到满足个体数达到 n 为止。

Step5:更新禁忌表,转 Step2。

End

说明:

(1)禁忌对象。禁忌的目的是为了多探索一些有效的邻域。PTS 算法选用个体的适应度函数值为禁忌对象,这有利于 PTS 算法在不同的适配值处搜索。

(2)特赦准则。在算法中可能会出现候选解全部被禁忌的情况,这时给满足解禁条件的候选解解禁,以实现更好的优化性能。

(3)算法终止准则。为了兼顾算法的搜索性能和时间性能，采用阈值法设计算法终止准则，在 PTS 算法的一次调用中，从破禁表中破禁的个体数达到一定次数时，说明进行邻域搜索不能再增加个体的多样性，则算法终止。

(4)GEP 禁忌表的使用保证了种群的多样性，有效地克服了可能出现的不同次调用 PTS 算法多次在同一目标值处循环搜索的情况。

(5)PTS 算法也能够接受劣化解。

3.3　基于并行禁忌搜索的基因表达式编程算法

算法 2 Gene Expression Programming Based on Parallel Tabu Search Algorithm(PTS-GEP)

输入：种群规模 N，变异率 Pm，早熟识别参数 α，破禁阈值 β。

输出：挖掘模型(函数表达式)。

Begin

Step1：令运行代数 gen=0。

Step2：初始化种群。

Step3：计算初始种群每个个体的适应度值。

Step4：进行服从保优原则的选择操作。

Step5：变异操作。

Step6：计算遗传新种群个体的适应度值。

Step7：由定义 3 计算 $prem(gen)$，若 $prem(gen)<\alpha$，继续下一步，否则转 Step9。

Step8：按定义 5 初始化 GEP 禁忌表 $T=\Phi$，分 N/n 次分别调用 PTS 算法，进行邻域搜索。

Step9：gen++。

Step10：满足算法停止条件，算法结束，输出结果，否则，转 Step4。

End

PTS-GEP 算法的流程如图 1。

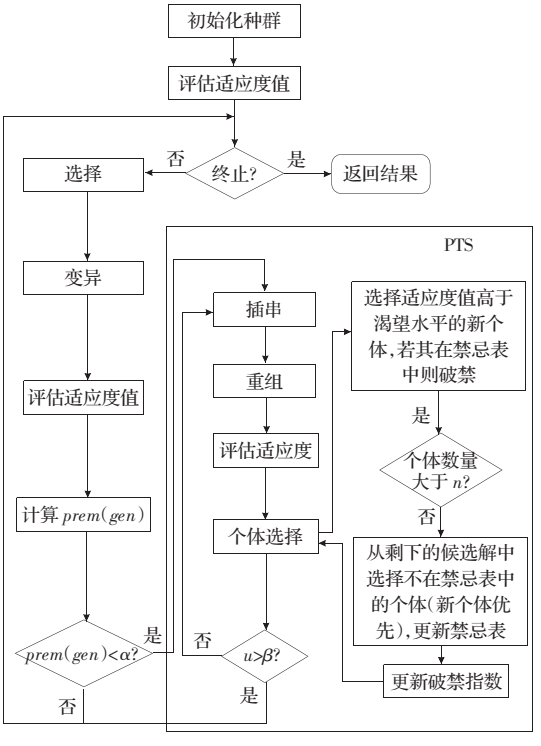

图 1　PTS-GEP **算法流程图**

PTS-GEP 融合了 PTS 和传统 GEP。算法运行过程中先用 GEP 进行全局搜索，当早熟现象发生，种群早熟度满足 $prem(gen)<\alpha$，算法陷入局部最优。此时调用 PTS 进行局部搜索，改善种群多样性。PTS-GEP 有效结合了传统 GEP 并行的大范围搜索能力和 PTS 的局部搜索能力。

3.4　PTS-GEP 算法分析

分三种情况分析 $prem(k)=d*v$：

(1)PTS-GEP 刚开始运行时，主要进行全局搜索。因为这时种群目标距离度 d、种群多样性指数 v 均偏大，$prem(k)=d*v$ 偏大，很少调用 *PTS*。

(2)随着 PTS-GEP 的运行，d 越来越小，种群早熟度 $prem(k)$对多样性指数 v 非常敏感。一旦种群多样性偏低(即 v 偏小)，$prem(k)=d*v$ 迅速偏小，所以，当种群多样性低时，PTS-GEP 能及时调用 PTS 进行邻域搜索，改进种群质量。

(3)PTS-GEP 临近结束时(通常也是算法最容易陷入局部最优)，d 偏小。此时，种群早熟度参数 $prem(k)=d*v$ 对多样性指数 v 更敏感，一旦种群多样性偏低，$prem(k)$将更小，PTS-GEP 频繁调用 PTS 改进群体质量，挖掘出的模型精度得到提高。

总之，PTS-GEP 运行初期 PTS 调用较少，种群趋于早熟时则 PTS 的调用次数较多。实验证明早熟度阈值 α 设为 0.005 时能达到良好的效果。

4　实验结果与分析

通过两组实验，对比 GEP、UC-GEP 和 PTS-GEP 的结果，检验 PTS-GEP 的性能。实验的全部程序用 C# 实现，实验平台为 Microsoft VS2005，.NET Framework2.0，Windows XP SP2 操作系统；P4 2.80 GHz，256 M 内存。实验参数见表 1。

表 1　实验参数

	杉木生产力	太阳黑子
最大代数	5 000	500
函数集	+-*/~SL	+-*/
变量集	T	a-j
种群大小	100	100
基因头长	6	7
变异率	0.044	0.044
插串率	0.1	0.1
重组率	0.3	0.3
早熟度阈值 α	0.005	0.005
破禁阈值 β	15	15

“L”代表 ln 运算；“~”代表 10(x)运算；“S”代表 sin 运算

4.1　杉木生产力模型

实验的数据来自于文献[8]，适应度函数和模型精度评估函数均与文献[8]相同。每个算法独立运行 20 次，通过比较适应度值和模型精度来检验算法的性能。实验结果如表 2 所示。

表 2　杉木生产力模型对照表

	GEP	UC-GEP	PTS-GEP
最优适应度	0.903 5	0.920 1	0.941 7
平均适应度	0.753 8	0.852 3	0.932 6
最优精度/(%)	81.34	87.45	93.27
平均精度/(%)	70.21	84.68	90.82

从表 2 可以看出，PTS-GEP 挖掘出的最优模型无论在适应度函数和精度上都明显优于其他两种算法，该模型精度比 GEP 和 UC-GEP 分别提高了 11.93%和 5.73%。而且，PTS-GEP 挖掘的模型在平均适应度和精度上也明显优于其他两种算法，

说明了 PTS-GEP 稳定性比其他两种算法高。这主要是因为 PTS 改善了 PTS-GEP 的局部搜索能力。

GEP、UC-GEP、PTS-GEP 挖掘出的模型对应的杉木生产力与年平均气温的关系如图 2 所示。从图 2 中可以看出,PTS-GEP 挖掘出的模型更接近于真实模型。

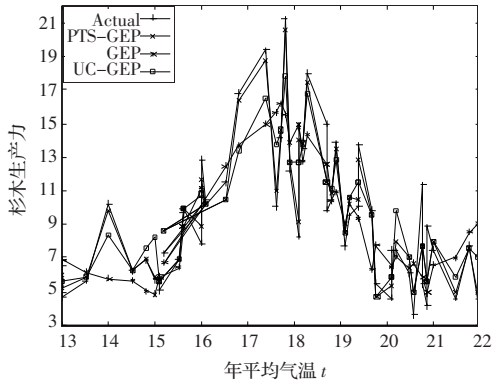

图 2 杉木生产力模型对比

PTS-GEP 挖掘出的最优模型为:

$$NPP=10.000\,7+\sin(0.784\,258*t)-\sin(0.131\,5-t)*(-3.819\,5+\sin(t))-\sin((1.458\,8-t)*t)-\sin(1.458\,8(-2.360\,5+t)*t)+\sin(23.606*t*t)+\sin(2.904\,44*t*\ln(t*t))$$

4.2 太阳黑子模型

为检验 PTS-GEP 多维数据的建模能力,用 Wolfer 太阳黑子序列的观测值(见文献[2]表 1)作为实验目标,其隐含的维数是 10,延时为 1,这样可以得到 90 组测试数据。实验采用的适应度函数和模型精度评估函数分别为基于绝对误差的适应度函数和 R-Square[2]。每个算法独立运行 20 次,通过比较适应度值和模型精度来检验算法的性能。实验结果如表 3 所示。

表 3 太阳黑子模型对照表

	GEP	UC-GEP	PTS-GEP
最优适应度	89 176.61	89 351.30	89 399.40
平均适应度	89 033.29	89 331.35	89 363.41
最优精度/(%)	88.28	91.16	95.97
平均精度/(%)	81.19	85.99	90.99

从表 3 中可以看出,就最优解来说,PTS-GEP 挖掘出的模型精度最高、UC-GEP 次之、GEP 模型精度最差。PTS-GEP 得到的模型精度比 GEP 和 UC-GEP 分别提高了 7.69% 和 4.81%。PTS-GEP 挖掘出的模型的平均适应度和精度也明显优于其他两种算法,再次验证了 PTS-GEP 稳定性优于其他两种算法。

PTS-GEP 挖掘出的最优模型为:

$$y=0.901\,5+\frac{e}{b-f}+\frac{c}{c-g}+0.212\,8*\left(\frac{10.007+2*a+1.109\,26*f}{i-e}-i\right)+j+\frac{0.131\,5*c*j}{2.360\,5+i}+\frac{j}{i-f}$$

4.3 算法时间复杂度

以上两组实验中,三种算法均独立运行 20 次,得到的最优模型平均耗时和平均代数如表 4 所示。

表 4 算法时间复杂度对照表

	杉木生产力		太阳黑子	
	平均时间/ms	平均代数	平均时间/ms	平均代数
GEP	193	413	2 300	4 325
UC-GEP	413	437	2 631	3 008
PTS-GEP	521	211	2 850	2 221

从表4 中可以看出,两组实验中,GEP 耗时最少,平均代数较高。PTS-GEP 平均代数明显低于其他两种算法,但耗时最高,这主要是因为 PTS-GEP 调用 PTS 进行的局部搜索耗费时间较多。这说明了参数 α 和 β 的设置对 PTS-GEP 的时间复杂度非常关键,通过优化这些参数可以降低 PTS-GEP 的时间复杂度。

5 结论

将禁忌搜索引入到 GEP 的遗传操作选择中,提出了并行禁忌基因表达式编程算法 PTS-GEP。通过两组实验验证了 PTS-GEP 比传统 GEP 在模型挖掘方面稳定性高、寻优能力强,挖掘出的模型更接近于真实模型。同时,实验表明 PTS-GEP 的时间复杂度明显高于传统 GEP。未来的工作中,将致力于降低 PTS-GEP 的时间复杂度。

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] Ferreira C.Function finding and the creation of numerical constants in gene expression programming[EB/OL].(2002).http://www.gene-expression-programming.com/webpapers/Ferreira-WSC7.pdf.

[3] Ferreira C.Mutation,transposition,and recombination:An analysis of the evolutionary dynamics[EB/OL].(2002).http://www.gene-expression-programming.com/webpapers/ferreira-fea02.pdf.

[4] Zuo Jie,Tang Chang-jie,Li Chuan,et al.Time series predication based on gene expression programming[C]//LNCS 3129(Lecture Notes in Computer Science):WAIM04,International Conference for Web Information Age 2004.Berlin Heidelberg:Springer Verlag,2004,3129:55-64.

[5] 胡建军,唐常杰,彭京,等.快速跳出局部最优的 VPS-GEP 算法[J].四川大学学报:工程科学版,2007,30(1):128-133.

[6] 饶元,元昌安.基于模拟退火的基因改进型 GEP 算法[J].四川大学学报:自然科学版,2008,45(4):767-772.

[7] Xu Kai-kuo,Liu Yin-tian,Tang Rong,et al.A novel method for real parameter optimization based on gene expression programming[J].Applied Soft Computing Journal,2008(9).

[8] Yuan Chang-an,Tang Chang-jie.Intelligent function model discovery system based upon gene expression programming [J].Journal of Computational Information Systems,2006,2(4):1299-1307.

[9] 邓松,王汝传.基于网格的 GEP 函数挖掘算法研究[J].通信学报,2008,29(6):69-74.

[10] 邓松,元昌安,赵波,等.基于 GEP 和神经网络的属性约简分类算法[J].计算机工程与应用,2006,42(23):154-157.

[11] 陈锋,陈月辉,张建中.基于 IP 和 GEP 算法的股票预测[J].计算机工程与应用,2007,43(26):227-229.

[12] 张颖,刘燕秋.软计算方法[M].北京:科学出版社,2002.