

带共轭梯度算子的爆炸搜索算法

曹 炬 李艳姣 陈 钢

(华中科技大学数学与统计学院 武汉 430074)

摘 要 爆炸搜索算法作为一种全局优化算法,在迭代后期会出现收敛速度慢、精度低的问题,而传统的优化算法恰好能克服这些缺点。因此,引入一种传统算法——近似共轭梯度法,即用差商代替导数的共轭梯度法。在此基础上,提出了带共轭梯度算子的爆炸搜索算法,先引入了新的变异算子来提高算法的全局搜索能力,再运用共轭梯度法添加一个新的算子——共轭梯度算子,实现对最优炸点的局部搜索,从而提高算法的收敛速度与精度。6 个常用的 benchmark 函数的测试结果说明,改进算法的优化结果明显优于原算法。

关键词 爆炸搜索算法,变异算子,共轭梯度法

中图分类号 TP301 **文献标识码** A

Explosion Search Algorithm with Conjugate Gradient Operator

CAO Ju LI Yan-jiao CHEN Gang

(School of Mathematics and Statistics, Huazhong University of Science and Technology, Wuhan 430074, China)

Abstract As a global optimization algorithm, Explosion Search Algorithm (ESA) has some problem of low convergence speed and low optimization precision in the later period of the optimization. Fortunately, some deterministic optimization algorithms can overcome these shortcomings. Therefore, a deterministic algorithm without derivate information, which is called approximate conjugate gradient algorithm using difference quotient, was added in ESA. Based on the above, an improved Explosion Search Algorithm with Conjugate Gradient Operator (CGESA) was proposed. In CGESA, a new mutation operator is introduced to enhance the global search ability. Meanwhile, a new operator is introduced namely conjugate gradient operator to improve the local search ability of the optimal burst point, so that the convergence speed and optimization precision of CGESA are improved. Experimental results of the six well-known benchmark functions indicate that CGESA achieves better performance than ESA.

Keywords Explosion search algorithm, Mutation operator, Conjugate gradient method

1 引言

通过观察和模拟自然界的一些现象,人们提出了很多智能优化算法,如遗传算法^[1]、粒子群算法^[2]、果蝇算法^[3]、磷虾群算法^[4]、萤火虫算法^[5]、水循环算法^[6]等,但是任何一种算法从提出到成功应用于实际问题都需要一个不断探索改进的过程。文献[7]提出的爆炸搜索算法(Explosion Search Algorithm, ESA)作为一种新型智能优化算法,具有全局搜索能力较强、稳定性好等优点。但是对 ESA 中的变异算子的研究发现,变异后的炸点很有可能会完全抛弃原炸点携带的信息,这在一定程度上降低了算法的收敛速度。另外,在很多智能优化算法中,每次迭代过程中最优个体的位置起着至关重要的指导作用。然而在 ESA 中,显然忽略了对最优炸点的单独操作。

文献[8]通过在最优炸点处引入差商最速下降法来找到新的最优位置,但对于一些测试函数,如 Rosenbrock 函数,该方法的改进效果并不明显。因此考虑引入另一种传统的优化

算法——共轭梯度法,该算法是通过利用负梯度来构造共轭梯度方向,具有较强的局部搜索能力,收敛速度快,且计算过程中不需要存储矩阵相关的信息。

结合这两个算法的特点,提出了带共轭梯度算子的爆炸搜索算法(Explosion Search Algorithm with Conjugate Gradient Operator, CGESA),先设计出新的变异算子来提高变异后炸点的性能,又通过加入共轭梯度法来实现对最优炸点的局部搜索,提高算法的局部搜索能力。实验结果表明,CGESA 能取得较好的寻优结果。

2 爆炸搜索算法^[7]

炸弹的爆炸是设计爆炸搜索算法的思想来源,单个炸点的爆炸相当于对搜索空间的一次局部性的搜索,如果在整个搜索空间中随机安放 M 个炸点,那么炸点的爆炸就相当于对整个搜索空间的一次并行局部的搜索。在 ESA 中,通过爆炸搜索算子来模拟此过程,再通过加入迁移算子、变异算子来提高算法的性能。各个算子的具体实施步骤如下。

到稿日期:2013-07-03 返修日期:2013-10-20 本文受国家自然科学基金项目(11171122)资助。

曹 炬(1955—),男,博士,教授,主要研究方向为运筹学与智能计算及其应用;李艳姣(1989—),女,硕士,主要研究方向为智能优化算法及其应用, E-mail: liyanjiao039@163.com(通信作者);陈 钢(1989—),男,硕士,主要研究方向为智能优化算法及其应用。

2.1 爆炸搜索算子

假设所求解的优化问题为最小值问题,对空间维数为 D 、爆炸半径为 $R_t=(R_{t_1}, R_{t_2}, \dots, R_{t_D})$ 的炸点 $X=(x_1, x_2, \dots, x_D)$ 而言,以该炸点位置为起点,以 $rand * e \otimes R_t$ (其中 e 为搜索空间中各个坐标轴的单位向量, \otimes 表示两个向量的对应分量相乘) 为步长进行 $2 * D$ 次搜索,将搜索到的 $2 * D$ 个点进行排序,从中挑选出适应值最小的 k 个点,以这 k 个点所在的方向 $e_{i_1}, e_{i_2}, \dots, e_{i_k}$ 为搜索的下降方向。接着从点集

$$\{X + 2 * rand * m * \frac{R_t}{Num} \otimes e_{i_j} \mid m=1, 2, \dots, Num, j=1, 2, \dots, k\}$$

(其中, Num 表示某个方向上的取点数目) 中取点。如果新找到炸点的适应值优于原炸点,则对原炸点的位置进行更新。

关于炸点爆炸半径 R_t 的变化有下面两种情况:

当炸点位置有更新时, R_t 更新公式为

$$R_t = R_{t_{max}} - t / MaxDT * (R_{t_{max}} - R_{t_{min}}) \quad (1)$$

其中, $R_{t_{max}}, R_{t_{min}}$ 为爆炸半径的最大值与最小值, t 为当前迭代次数, $MaxDT$ 为最大迭代次数。

当炸点位置没有更新时, R_t 更新公式为

$$R_t = \vartheta * R_t \quad (2)$$

其中, ϑ 为自适应因子。

2.2 迁移算子

执行完爆炸搜索算子后,按各炸点适应值的大小进行排序,对于适应值较差的 $[\mu M]$ (μ 为迁移比例) 个炸点进行如下的更新:

$$X' = X + 2 * rand * (gbest - X) \quad (3)$$

其中, $gbest$ 为历史最优值。

2.3 变异算子

对炸点 $X=(x_1, x_2, \dots, x_D)$ 来说,变异公式为

$$x_i' = x_i (1 + N(0, \sigma_i)) \quad (4)$$

其中, $\sigma=(\sigma_1, \sigma_2, \dots, \sigma_D)=0.1 * (X_{max} - X_{min})$, X_{min}, X_{max} 分别对应整个搜索区间的最大与最小边界值。

3 带共轭梯度算子的爆炸搜索算法

本文提出了一种带共轭梯度算子的爆炸搜索算法,在保持原算法中爆炸搜索算子和迁移算子不变的前提下,对原有的变异算子进行了重新设计,在此基础上又加入共轭梯度算子。下面是对这两个算子的详细介绍。

3.1 新变异算子

变异算子的作用是避免炸点多样性的减小,使炸点能够搜索到的范围更广,从而提高算法找到全局极值点的可能性。但是,从文献[9]中不难发现,对于一个具体的算法而言,不同变异算子的效果是不一样的,因此,有必要对变异算子进行重新设计,以提高改进算法的性能。

原算法中的变异公式涉及到正态分布 $N(0, \sigma_i)$,当搜索空间的取值范围较大时,由 σ_i 的含义可知,该正态分布 $N(0, \sigma_i)$ 产生的随机数有可能偏大。在此情况下对炸点执行变异算子,新炸点与原炸点相差甚远的概率偏大,虽然整个炸点的多样性增加了,但是新产生的炸点可能是没有意义的,即对提高整个算法的性能作用不大。

下面对该正态分布的均值和方差进行重新设计,以取得更满意的结果。考虑炸点 $X=(x_1, x_2, \dots, x_D)$,新变异公式

如下:

$$x_i' = x_i (1 + N(1, 1))$$

改进后的变异公式有两个方面的优势:第一,算法涉及的参数减少一个,使算法相对简化;第二,当搜索范围相对较大时,对炸点执行新的变异公式相当于在原炸点周围进行一次变动,这不但可以防止炸点多样性的缺失,还可以充分利用到原炸点的信息,提高算法的收敛速度。

3.2 共轭梯度算子

共轭梯度法作为无约束优化问题中的常用算法之一,在求解一般函数的优化问题中,需要解决两个问题。首先,对于一般性的函数而言,FR 共轭梯度法很难像求解正定二次函数那样具有二次终止性^[10],因此在实际应用过程中,一般采用的是修改后的以 n 步为一轮的 FR 共轭梯度法。即每执行完 n 步后,本轮结束,用本轮中的最后一个点作为下一轮迭代的起点,同时将搜索方向重新设定为负梯度方向进行下一轮搜索,直至满足条件。其次,一般函数不一定存在导数,这时参照文献[11]的思想用差商代替导数,其目的是使算法的适用范围更广。这种用差商代替导数的共轭梯度法称为近似的共轭梯度法。

在智能优化算法中,很多都有非最优个体向最优个体学习的机制,如粒子群算法^[2]等。因此,每次迭代后找到的最优个体位置的优劣会影响整个算法的收敛速度和精度。在改进算法中,通过添加共轭梯度算子来实现对最优炸点的操作。具体而言,就是在执行新变异算子以后,先对全局最优炸点 $gbest$ 进行更新,接着用近似共轭梯度法对更新后的炸点进行局部性的搜索。如果能找到更优的炸点位置 $gbest1$,则令 $gbest = gbest1$ 。共轭梯度算子的引入,使得每次迭代后找到的最优炸点位置更为精确,进而有利于算法找到问题的最优解。

CGESA 算法的具体实现步骤如下:

Step 1 令 $t=0$,初始化 M 个炸点的位置,找出全局最优炸点 $gbest$ 。

Step 2 对 M 个炸点依次执行爆炸搜索算子。

Step 3 对所有炸点按适应值的大小进行排序,对适应值较差的炸点执行式(3)。

Step 4 在 $[0, 1]$ 之间产生一个随机数,如果该数小于预先设定的变异概率,则对该炸点执行式(4),否则,对下一个炸点执行该操作。

Step 5 更新全局最优炸点 $gbest$ 。

Step 6 对最优炸点 $gbest$ 执行 n 步为一轮的 FR 近似共轭梯度法,如果能找到更好的炸点位置,则更新 $gbest$ 。令 $t=t+1$,如果 t 小于最大迭代次数,转 Step 2,否则迭代结束。

4 实验结果及分析

4.1 函数介绍^[12]

表 1 中第 1、2、5 个函数均为单峰函数。其中, Sphere 是较为简单的函数,全局极值点在 $(0, 0, \dots, 0)$ 处,主要用于测试算法收敛精度的好坏。Rosenbrock 为非凸函数,全局极值点在 $(1, 1, \dots, 1)$ 处,对大多数算法都很难找到正确的搜索方向,因为其全局最优与局部最优之间有条很窄的山谷,常用来检验算法的执行性能。Schwefel12 函数的全局极值点在 $(0, 0, \dots, 0)$ 处,也用于测试算法的执行能力。

表1中第3、4、6个函数是多峰函数,全局极值点均在(0, 0, ..., 0)处。其中,Rastrigin 函数通过在 Sphere 函数上加入余弦函数来产生大量的局部极值点,从而使寻找其全局极值点变得很难。而 Griewank 函数也存在很多个局部极值,很难优化。Ackley 函数通过增加余弦波来调整指数函数,有很多局部最优值。这3个函数都用于测试算法是否有跳出局部极值的能力。

表1 6个测试函数的介绍

函数名	函数表达式	取值范围
Sphere	$f_1 = \sum_{i=1}^{30} x_i^2$	$[-100, 100]$
Rosenbrock	$f_2 = \sum_{i=1}^{29} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	$[-30, 30]$
Rastrigin	$f_3 = \sum_{i=1}^{30} [x_i^2 - 10 \cos(2\pi x_i) + 10]$	$[-5.12, 5.12]$
Griewank	$f_4 = \frac{1}{4000} \sum_{i=1}^{30} x_i^2 - \prod_{i=1}^{30} \cos(\frac{x_i}{\sqrt{i}}) + 1$	$[-600, 600]$
Schwefel12	$f_5 = \sum_{i=1}^n (\sum_{j=1}^i x_j)^2$	$[-100, 100]$
Ackley	$f_6 = -20 \exp(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}) - \exp(\frac{1}{n} \sum_{i=1}^n \cos 2\pi x_i) + 20 + e$	$[-32, 32]$

4.2 参数设置

实验中,CGESA的参数设置为:炸点规模 $M=20$,方向数 $k=3$,各个方向上选取炸点数 $Num=16$,搜索空间的上下界分别为 X_{max}, X_{min} ,爆炸最大半径 $Rt_{max} = 0.05 * (X_{max} - X_{min})$,最小半径 $Rt_{min} = (1e-6) * (X_{max} - X_{min})$,自适应因子 $\theta = 1.5$,迁移比例 $\mu = 0.4$,变异概率 $P = 0.05$ 。其中,上述参数的设置采用单因素分析法,即检验单个参数变动对实验结果的影响程度,并从中选取最优参数值。另外,ESA和PSO^[13]算法的参数设置详见文献[7, 14]。实验重复运行次数 $T=20$ 。

4.3 测试结果及分析

测试结果与分析分为3个部分:第一部分选取前4个测试函数,给出了它们的优化曲线和 PSO、ESA、CGESA 3种算法下的实验数据,并对它们进行了分析;第二部分选取后面两个测试函数,给出了维数为30、50这两种情况下的优化曲线和3种算法下的实验数据,并对它们进行了分析;第三部分讨论了这6个测试函数在维数为30的情形下,达到预先设定条件时的收敛率。

4.3.1 前4个测试函数的优化曲线、实验数据和分析

由图1知,在200代以后,ESA的优化曲线趋于平缓,CGESA的优化曲线基本上一直保持较好的下降趋势。由图2和图3可以看出,CGESA在较少的迭代次数内就可以取得较为满意的结果。由图4知,在迭代次数为700多之前,CGESA与ESA的结果相差不大,但是随着迭代的继续进行,CGESA几乎呈直线型下降,而ESA的下降仍较平缓。综合图1—图4以及上面的分析可知,CGESA优于ESA和PSO算法。

表2中迭代终止条件为达到最大迭代次数 $MaxDT=1000$ 。由表2可知,对于最小值、平均值还有标准差,CGESA的优化结果均好于ESA和PSO算法。平均值较好说明CGESA无论在收敛速度还是收敛精度方面都能取得较好的结果,且对于多峰函数 f_3 和 f_4 ,该算法不易陷入局部最优;标准差较小说明CGESA的稳定性较好。综合这3个指标可以看出,改进算法的整体优化效果比ESA、PSO算法好。

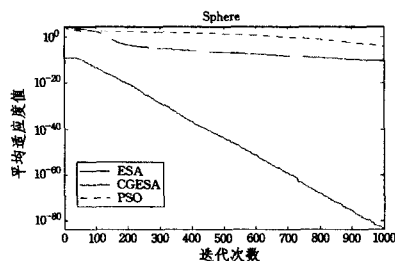


图1 函数 f_1 的优化曲线

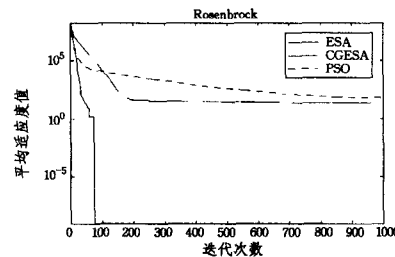


图2 函数 f_2 的优化曲线

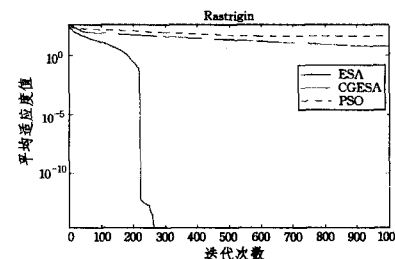


图3 函数 f_3 的优化曲线

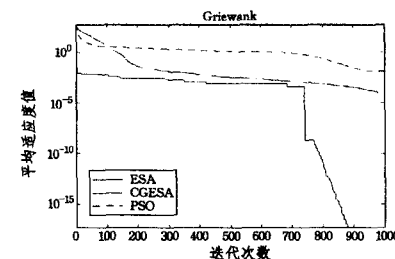


图4 函数 f_4 的优化曲线

表2 PSO、ESA与CGESA的实验结果

函数		1	2	3	4
PSO	min	2.302e-06	23.04	20.89	4.428e-05
	mean	7.077e-05	62.65	41.93	0.0116
	std	7.176e-05	1.122e+02	12.95	0.0118
ESA	min	2.865e-12	13.77	0	4.577e-06
	mean	2.879e-11	21.70	5.574	1.458e-05
	std	3.400e-11	18.62	6.610	1.007e-05
CGESA	min	9.740e-92	3.574e-12	0	0
	mean	3.422e-84	9.114e-10	0	0
	std	1.222e-83	1.188e-09	0	0

4.3.2 最后两个测试函数的优化曲线、实验数据和分析

由图5、图6可知,维数的增大对CGESA的影响不大,其优化曲线均明显优于其它两种算法。由图7、图8可知,在30维情况下,CGESA在400代左右就趋于平缓,且其优化曲线一直优于ESA、PSO算法;当维数增大到50维时,在300代以内,CGESA的优化效果并不好,但是在300代以后,CGESA的优化曲线迅速下降,而其它两种算法优化曲线下降速度一直较慢。综合图5—图8,CGESA的优化效果较好。

表3的迭代终止条件为达到最大迭代次数 $MaxDT=1000$ 。与前4个维数确定的测试函数不同,最后两个测试函数5、6的维数分别取30、50。由表3可知,无论维数取30还是50,CGESA的优化结果均优于ESA和PSO算法。对于这3种智能优化算法,增加测试函数的维数,会增加算法的搜索难度。但是CGESA在两种维数下,计算结果均较好,说明CGESA整体优化效果比ESA、PSO算法好。

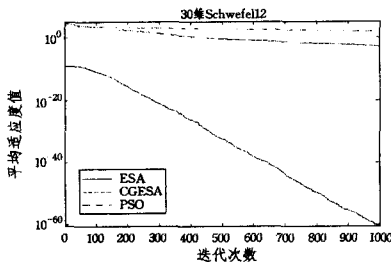


图5 函数 f_5 的优化曲线(30维)

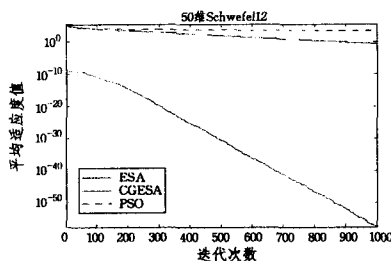


图6 函数 f_5 的优化曲线(50维)

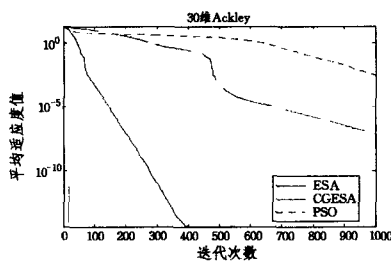


图7 函数 f_6 的优化曲线(30维)

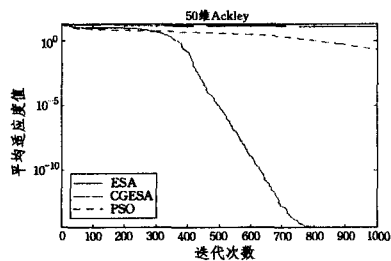


图8 函数 f_6 的优化曲线(50维)

表3 PSO、ESA与CGESA的实验结果

函数	5		6		
	30	50	30	50	
PSO	min	1.205e+03	7.371e-04	0.0566	
	mean	2.353e+03	0.0028	0.2230	
	std	1.449e+03	0.0018	0.2099	
ESA	min	3.723e-04	0.0477	4.025e-09	6.735e-07
	mean	0.0012	0.238	6.489e-08	13.09
	std	8.050e-04	0.236	1.636e-07	9.563
CG-ESA	min	1.584e-68	6.784e-65	8.881e-16	8.881e-16
	mean	3.380e-61	2.046e-58	4.263e-15	3.7303e-15
	std	9.390e-61	4.592e-58	7.944e-16	1.458e-15

4.3.3 6个测试函数的收敛率和分析

表4的迭代终止条件为达到预先设定的收敛精度 ϵ (其中 ϵ 的取值为PSO算法在维数为30、迭代次数为1000的情况下取得的平均值)或达到最大迭代次数 $MaxDT1=200$ 。由表4可知,CGESA达到预先设定的收敛精度 ϵ 所需的平均迭代次数远远小于ESA所需的平均迭代次数;在最大迭代次数为200的前提下,除了第3个测试函数,CGESA达到收敛精度的成功率均为100%,明显优于ESA。

表4 ESA与CGESA的实验结果

函数	收敛精度 ϵ	算法	最小迭代数	平均迭代数	方差	成功率
1	$7e-5$	ESA	190	198	3.187	20%
		CGESA	1	1	0	100%
2	62	ESA	116	182	26.23	45%
		CGESA	15	23	7.893	100%
3	41	ESA	68	175	51.89	20%
		CGESA	18	83.3	50.16	95%
4	$1e-2$	ESA	179	198	4.693	10%
		CGESA	2	39.6	53.89	100%
5	90	ESA	200	200	0	0%
		CGESA	1	1	0	100%
6	$3e-3$	ESA	200	200	0	0%
		CGESA	60	75	10	100%

结束语 针对爆炸搜索算法的不足,提出了一种改进的算法——带共轭梯度算子的爆炸搜索算法。该算法在最优点处进行基于差商信息的共轭梯度法计算,提高了算法的收敛精度与速度;新的变异算子增强了算法的全局搜索能力。不同于原算法,新算法将爆炸搜索算法的全局搜索能力和共轭梯度法的局部搜索能力有机结合起来,具有较好的性能。该改进算法对于Rosenbrock这一很难优化的函数能取得较为满意的结果。但是改进算法也存在一些不足,即算法的时间复杂度高于原算法,如何降低改进算法的时间复杂度和将其用于解决实际问题今后工作的重点。

参考文献

- [1] Holland J. Adaptation in Natural and Artificial Systems [M]. Ann Arbor; University of Michigan Press, 1975
- [2] James K, Eberhart R C. Particle Swarm Optimization [C]// IEEE International Conference on Neural Networks. IEEE, 1995; 1942-1947
- [3] Pan W T. A new Fruit Fly Optimization Algorithm; Taking the financial distress model as an example [J]. Knowledge-Based Systems, 2012, 26: 69-74
- [4] Gandomi A H, Alavi A H. Krill herd; A new bio-inspired optimization algorithm [J]. Commun Nonlinear Sci Numer Simulat, 2012, 17(12): 4831-4845
- [5] 吴斌, 崔志勇, 倪卫红. 具有混合群智能行为的萤火虫群优化算法研究 [J]. 计算机科学, 2012, 39(5): 198-199
- [6] Eskandar H, Sadollah A, Bahreininejad A, et al. Water cycle algorithm-A novel metaheuristic optimization method for solving constrained engineering optimization problems [J]. Computers & Structures, 2012, 110-111: 151-166
- [7] 曹炬, 侯学脚. 具有邻域搜索机制的爆炸搜索算法 [J]. 计算机工程, 2011, 37(18): 183-187
- [8] 曹炬, 侯学脚. 爆炸搜索算法及其收敛性证明 [J]. 计算机科学,

- [9] Paul S. Andrews. An Investigation into Mutation Operators for Particle Swarm Optimization[C]//IEEE Congress on Evolutionary Computation. 2006:1044-1050
- [10] 吴祁宗. 运筹学中最优化方法[M]. 北京:机械工业出版社, 2003:114-117
- [11] 赵小平. 差商最速下降法及其收敛性[J]. 华东化工学院学报, 1992,18(6):807-809
- [12] Yao Xin, Liu Yong, Lin Guang-ming. Evolutionary Programming Made Faster[J]. IEEE Transactions on Evolutionary Computation, 1999,3(2):82-102
- [13] Shi Yu-hui, Eberhart R C. A Modified Particle Swarm Optimizer [C]//Proceeding of IEEE International Conference on Evolutionary Computation. 1998,69-73
- [14] Poli R, Kennedy J, Blackwell T. Particle swarm optimization[J]. Swarm Intelligence Journal, 2007,1:33-57

(上接第 226 页)

- [14] Nakano K, Nishimura S. Deriving Event-Based Document Transformers from Tree-Based Specifications [J]. Electronic Notes in Theoretical Computer Science, 2001,44:181-205
- [15] Becker O. Transforming XML on the Fly[C]//XML Europe. 2003
- [16] Jiang H, Ho H, Popa L, et al. Mapping-driven XML transformation[C]//WWW, Banff, Alberta, Canada, 2007:1063-1072
- [17] Frisch A, Nakano K. Streaming XML transformations using term rewriting[C]//PLAN-X. Nice, France, 2007:2-13
- [18] Dvoráková J, Zavoral F. A Low-Memory SSXT Algorithm for XSLT Transformations [J]. Journal of Information Assurance and Security, 2008,3:230-239
- [19] Tanaka Y, Ito K, Fujima J. Meme media for clipping and combining Web resources [C]//International Conference on World Wide Web. Hong Kong, 2001:201-210
- [20] Fagin R, Haas L M, Hernández M, et al. Clio: Schema Mapping Creation and Data Exchange [C]//Lecture Notes in Computer Science. vol. 5600. Heidelberg Berlin: Springer, 2009:198-236
- [21] 宋玲, 吕强, 邓薇, 等. 基于语义和结构的 XML 文档相似度的计算方法[J]. 中文信息学报, 2012,26(5):59-64
- [22] 陈谊, 侯望, 新吉乐, 等. 基于 XML 和关系数据库的可视化工作流系统[J]. 系统仿真学报, 2012,24(1):110-116

(上接第 229 页)

测试。实验结果表明,本文提出的误差敏感的分类器算法在系统没有噪音的情况下分类预测的准确性要优于相关的算法;此外,误差敏感的分类器算法对噪音不敏感,当流数据中含有噪音时仍然具有很好的预测准确性。因此,该算法可以应用于大数据环境下数据流的在线分类预测。

参 考 文 献

- [1] Domingos P, Hulten G. Mining high-speed data streams [C]//Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery And Data Mining. ACM, 2000:71-80
- [2] Yang H, Fong S. Moderated VFDT in stream mining using adaptive tie threshold and incremental pruning[M]//Data Warehousing and Knowledge Discovery. Springer, 2011:471-483
- [3] Hulten G, Spencer L, Domingos P. Mining time-changing data streams[C]//Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery And Data Mining. 2001:97-106
- [4] Li W, Han J, Pei J. CMAR: Accurate and efficient classification based on multiple class-association rules[C]//IEEE International Conference on Data Mining. ACM, 2001:369-376
- [5] Han J. CPAR: Classification based on predictive association rules [OL]. <http://sci2s.ugr.es/keel/pdf/algorithm/congreso/2003-Yin-CPAR.pdf>, 2003
- [6] Thabtah F, Cowling P, Peng Y. MCAR: multi-class classification based on association rule[C]//The 3rd ACS/IEEE International Conference on Computer Systems and Applications. IEEE, 2005
- [7] 詹英, 吴春明, 王宝军. 一种与缓冲区紧耦合的环形循环滑动窗口的数据流抽取算法[J]. 电子学报, 2011,39(4):2262-2267
- [8] 崔贯勋, 李梁, 王柯柯, 等. 关联规则挖掘中 Apriori 算法的研究与改进[J]. 计算机应用, 2010,30(11):2952-2955
- [9] 詹英, 吴春明, 王宝军. 基于 RCSW 的数据流速度异常检测算法研究[J]. 电子学报, 2012,40(4):674-680
- [10] 吴枫, 仲妍, 吴泉源. 基于增量核主成分分析的数据流在线分类框架[J]. 自动化学报, 2010,36(4):534-542
- [11] Tang L, Tian L F, Steward B L. Classification of broadleaf and grass weeds using gabor wavelets and an artificial neural network[J]. Transactions of the Asae, 2003,46(4):1247
- [12] Pfahringer B, Holmes G, Kirkby R. New options for hoeffding trees[M]//AI 2007: Advances in Artificial Intelligence. Springer, 2007:90-99
- [13] Gama J A O, Rocha R, Medas P. Accurate decision trees for mining high-speed data streams[C]//Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2003:523-528
- [14] Hashemi S, Yang Y. Flexible decision tree for data stream classification in the presence of concept change, noise and missing values[J]. Data Mining and Knowledge Discovery, 2009,19:95-131
- [15] Bifet A, Holmes G, Kirkby R, et al. Moa: Massive online analysis [J]. The Journal of Machine Learning Research, 2010,99:1601-1604
- [16] Oza N C. Online bagging and boosting[C]//2005 IEEE International Conference on Systems, Man And Cybernetics. IEEE, 2005:2340-2345
- [17] Bifet A, Holmes G, Pfahringer B, et al. New ensemble methods for evolving data streams[C]//Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery And Data Mining. 2009:139-148
- [18] Bifet A, Gavaldà R. Learning from time-changing data with adaptive windowing [OL]. <http://www.lsi.upc.edu/~abifet/TimevaryingE.pdf>
- [19] 王柯柯, 崔贯勋, 倪伟, 等. 基于单元的快速的大数据集离群数据挖掘算法[J]. 重庆邮电大学学报: 自然科学版, 2010,22(5):673-677