

基于混沌万有引力搜索算法的 SVM 参数优化及应用

龚 安^{1,2} 吕 倩² 胡长军¹ 康忠健² 李华昱²

(北京科技大学计算机与通信工程学院 北京 100083)¹ (中国石油大学(华东) 青岛 266580)²

摘 要 针对万有引力搜索算法存在局部优化能力差的问题,引入混沌序列和遗传算法的交叉思想对其改善,并将其应用于 SVM 的参数优化,通过仿真实验验证了该 SVM 模型具有更高的精度。最后将该模型应用于火电厂一次风机的状态监测,实验结果表明该模型是有效的。

关键词 万有引力搜索算法,混沌序列,交叉,SVM,状态监测

中图法分类号 TP391 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2015.4.049

Parameter Optimization and Application of SVM Based on Chaos Gravitational Search Algorithm

GONG An^{1,2} LV Qian² HU Chang-jun¹ KANG Zhong-jian² LI Hua-yu²

(School of Computer & Communication Engineering, University of Science and Technology Beijing, Beijing 100083, China)¹

(China University of Petroleum, Qingdao 266580, China)²

Abstract Chaotic series and the crossover of genetic algorithm were introduced into gravitational search algorithm to overcome its poor local optimization problem. The improved algorithm was used for the SVM parameters optimization. And the simulated experiments show that the SVM model has a higher accuracy. At last the model was applied to the condition monitoring of primary air fan in power plant. And the experimental results indicate that the model is valid.

Keywords Gravitational search algorithm, Chaotic series, Crossover, SVM, Condition monitoring

1 引言

SVM 的性能与其相关参数的选取有很大关系。目前常用的 SVM 参数优化方法有蚁群优化算法^[1]、遗传算法^[2]、粒子群优化算法^[3]等。上述算法在对应的文献中证明了各自的有效性,但在寻优的过程中均会不同程度地陷入局部最优,因此探索新的群体智能算法来优化 SVM 参数是有意义的。

2009 年,伊朗克曼大学的教授 Esmat Rashedi 等^[4,5]提出一种新的启发式优化算法——万有引力搜索算法(Gravitational Search Algorithm, GSA)。研究发现,在对标准测试函数进行优化时, GSA 的寻优精度和收敛速度都明显优于粒子群优化算法和遗传算法等优化算法^[4]。但是与其它智能优化算法一样, GSA 也存在局部优化能力差的问题。因此本文提出将混沌序列^[6,7]和遗传算法^[8,9]的交叉思想引入 GSA 中,并将改进后的算法用于 SVM 参数优化,通过实验仿真验证了该算法能够获得更高的精度。将其应用到火电厂的一次风机状态监测中,实验结果表明该方法是有效的。

2 万有引力搜索算法

万有引力搜索算法将优化问题的解视为一组在空间运行的粒子,粒子因为万有引力的作用相互吸引,使得朝着质量最大的粒子移动,从而质量最大的粒子处在最优的位置,其即为

所求问题的最优解。算法步骤如下所示。

1. 粒子位置的初始化。系统由 N 个粒子组成,每个粒子定义在 D 维空间中,第 i 个粒子的位置定义如下:

$$x_i = (x_i^1, \dots, x_i^d, \dots, x_i^D), i = 1, 2, \dots, N \quad (1)$$

式中, x_i^d 表示第 i 个粒子在第 d 维空间的位置。

2. 粒子质量的计算。在 t 时刻,粒子的质量计算如下:

$$m_i(t) = \frac{fit_i(t) - worst(t)}{best(t) - worst(t)} \quad (2)$$

$$M_i(t) = \frac{m_i(t)}{\sum_{j=1}^N m_j(t)} \quad (3)$$

式中, $fit_i(t)$ 和 $m_i(t)$ 是 t 时刻粒子 i 的适应度值和质量, $best(t)$ 和 $worst(t)$ 分别表示在 t 时刻所有粒子的最优适应度值和最差适应度值。

对求最大值问题, $best(t)$ 和 $worst(t)$ 的定义如下:

$$best(t) = \max_{i \in \{1, \dots, N\}} fit_i(t) \quad (4)$$

$$worst(t) = \min_{i \in \{1, \dots, N\}} fit_i(t) \quad (5)$$

对求最小值问题, $best(t)$ 和 $worst(t)$ 的定义如下:

$$best(t) = \min_{i \in \{1, \dots, N\}} fit_i(t) \quad (6)$$

$$worst(t) = \max_{i \in \{1, \dots, N\}} fit_i(t) \quad (7)$$

3. 计算万有引力。粒子 i 与 j 的万有引力 $F_{ij}^d(t)$ 推导为:

$$F_{ij}^d(t) = G(t) \frac{M_i(t) M_j(t)}{R_{ij}(t) + \epsilon} (x_j^d(t) - x_i^d(t))$$

到稿日期:2014-04-28 返修日期:2014-08-11 本文受国家自然科学基金:基于阻抗模型故障特征匹配法的含 DG 配电网故障测距研究(61271001),中央高校基本科研业务费专项资金(14CX02030A)资助。

龚 安(1971—),男,博士生,副教授,硕士生导师,主要研究方向为数据挖掘与知识发现, E-mail: gongan0328@sohu.com; 吕 倩(1988—),女,硕士,主要研究方向为数据挖掘; 胡长军(1963—),男,博士,教授,博士生导师,主要研究方向为数据工程与知识工程、高性能计算; 康忠健(1971—),男,博士,教授,博士生导师,主要研究方向为电力系统故障监测与诊断; 李华昱(1977—),男,博士,副教授,主要研究方向为数据挖掘。

$$(d=1,2,\dots,D) \quad (8)$$

其中, $G(t)=G_0 e^{-\frac{t}{T}}$ 表示 t 时刻的万有引力常数, $G_0=100$, $\alpha=20$, T 是系统迭代的次数。 $R_{ij}(t)=\|x_i(t), x_j(t)\|_2$ 表示粒子 i 与 j 间的欧氏距离。

4. 计算合力。粒子 i 所受的合力为:

$$F_i^d(t)=\sum_{j=1, j \neq i}^N rand_j F_{ij}^d(t) \quad (9)$$

其中, $rand_j$ 是 $[0,1]$ 之间的一个随机数。

5. 更新加速度、速度和位置的值。

$$a_i^d(t)=\frac{F_i^d(t)}{M_i(t)} \quad (10)$$

$$v_i^d(t+1)=rand_i \times v_i^d(t)+a_i^d(t) \quad (11)$$

$$x_i^d(t+1)=x_i^d(t)+v_i^d(t+1) \quad (12)$$

其中, $rand_i$ 是 $[0,1]$ 之间的一个随机数。

6. 返回到步骤 2 循环迭代, 直到达到循环次数或要求的精度为止。

3 混沌万有引力搜索算法

分析 GSA 发现, 初始种群的产生是随机的, 容易产生重复和分布不均匀, 同时, GSA 也容易陷入局部最优。因此本文将混沌序列和遗传算法的交叉思想引入 GSA 中, 利用混沌序列的遍历性来产生初始种群和进行扩展操作, 采用交叉思想实现较差个体获取最好个体的信息。改进后的算法用 IGSA 表示。

3.1 初始种群的产生

利用混沌序列产生初始种群的步骤如下。

1. 产生一个 D 维的随机向量:

$$\eta=[\eta_1^d, \eta_2^d, \dots, \eta_d^d, \dots, \eta_D^d] \quad (13)$$

式中, $\eta_i^d \in [0,1]$ 且各分量之间有微小差别。

2. Logistic 映射。将 η 作为迭代初始值, 根据 Logistic 映射得到方程为:

$$\eta_{i+1}^d=\mu \eta_i^d(1-\eta_i^d) \quad (14)$$

式中, $\mu=4$; $i=1,2,\dots,N$, $d=1,2,\dots,D$ 。

3. 将混沌运动的遍历范围映射到优化变量的搜索区间:

$$x_i^d=\min^d+\eta_i^d(\max^d-\min^d) \quad (15)$$

式中, $\max^d$ 为取值上限, $\min^d$ 为取值下限。

3.2 对新种群的处理

将新种群按照适应度大小分成 3 个等级: 好(A)、中(B)、差(C), 对其分别给予不同的操作。

1. 群体 A 中每个粒子在其小邻域内^[10], 利用混沌序列产生 p 个新粒子, 最后保留包括自身在内的 $p+1$ 个粒子中适应度最好的粒子。该操作相当于在优秀粒子的小邻域内搜索更优秀的粒子。利用混沌序列产生小邻域内的粒子的步骤如下:

(1) 利用式(13)、式(14)得到 p 个混沌变量 η_i 。

(2) 将混沌变量按式(16)“放大”到一个以粒子当前位置 x_i^d 为中心、以 r 为半径的区域上。这里取 $r=0.005(\max^d-\min^d)$, $\max^d$ 为 d 维最大值, $\min^d$ 为 d 维最小值。

$$y_i^d=x_i^d+r(2\eta_i^d-1) \quad (16)$$

2. 群体 B 中每个粒子与群体 A 中最好的粒子进行交叉操作, 实现 B 中粒子获取 A 中最好粒子的信息, 使其向最好的粒子靠近。其中交叉概率从 0.9 到 0.6, 随着迭代次数线性变小。

3. 群体 C 中每个粒子在其大邻域内^[10], 利用混沌序列产

生 q 个新粒子, 最后保留包括自身在内的 $q+1$ 个粒子中适应度最好的粒子。该操作相当于使较差粒子尽快跳出当前位置。利用混沌序列产生较大邻域内的粒子的步骤如下:

(1) 利用式(13)、式(14)得到 q 个混沌变量 η_i 。

(2) 将混沌变量按式(17)“放大”到一个以粒子当前位置 x_i^d 为中心、以 R 为半径的区域上, 这里取 $R=0.5(\max^d-\min^d)$ 。

$$y_i^d=x_i^d+R(2\eta_i^d-1) \quad (17)$$

3.3 IGSA 在函数优化中的性能分析

为了全面分析 IGSA 的性能, 选取文献[4]中单峰高维函数、高维多峰函数、多峰低维函数每类中的两个函数做测试, 分别为 $F_3 F_7$ 、 $F_9 F_{12}$ 、 $F_{14} F_{15}$, 如表 1 所列, 此处的函数下标、实验的种群个数和迭代次数与文献[4]中的一致。实验的收敛速度和精度结果如图 1—图 6 所示。文中实验仿真平台均为 Windows 7 下的 Matlab R2010b。

表 1 标准测试函数

测试函数	S
$F_3(x)=\sum_{i=1}^n(\sum_{j=1}^i x_j)^2$	$[-100,100]^n$
$F_7(x)=\sum_{i=1}^n ix_i^4+\text{random}[0,1]$	$[-1.28,1.28]^n$
$F_9(x)=\sum_{i=1}^n [x_i^2-10\cos(2\pi x_i)+10]$	$[-5.12,5.12]^n$
$F_{12}(x)=-\frac{\pi}{n}\{10\sin(\pi y_1)+\sum_{i=1}^{n-1}(y_i-1)^2[1+10\sin^2(\pi y_{i+1})+(y_n-1)^2]+\sum_{i=1}^n u(x_i,10,100,4)$	
$y_i=1+\frac{x_i+1}{4}$	$[-50,50]^n$
$u(x_i,a,k,m)=\begin{cases} k(x_i-a)^m, & x_i>a \\ 0, & -a<x_i<a \\ k(-x_i-a)^m, & x_i<-a \end{cases}$	
$F_{14}(x)=(\frac{1}{500}+\sum_{j=1}^{25}\frac{1}{j+\sum_{i=1}^j(x_i-a_{ij})^6})^{-1}$	$[-65.53,65.53]^2$
$F_{15}(x)=\sum_{i=1}^{11}[a_i-\frac{x_1(b_i^2+b_2x_2)}{b_i^2+b_3x_3+x_4}]^2$	$[-5,5]^4$

图 1—图 6 表明, IGSA 在收敛速度和精度上优于 GSA, 为其用于优化 SVM 参数、提高精度提供了依据。

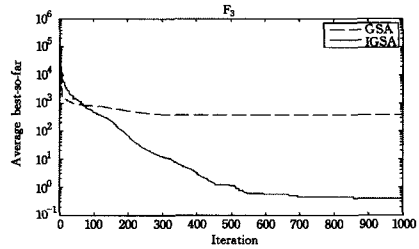


图 1 IGSA 和 GSA 在 F_3 上的性能比较

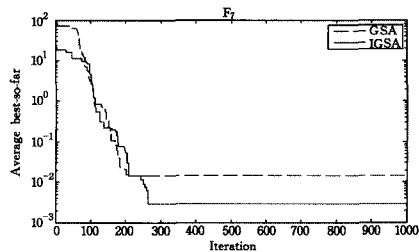


图 2 IGSA 和 GSA 在 F_7 上的性能比较

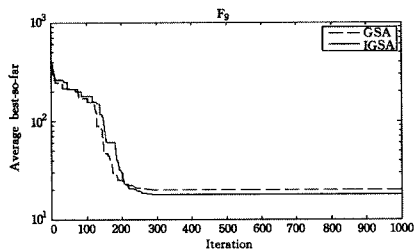


图3 IGSA 和 GSA 在 F_9 上的性能比较

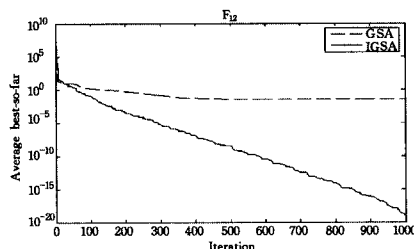


图4 IGSA 和 GSA 在 F_{12} 上的性能比较

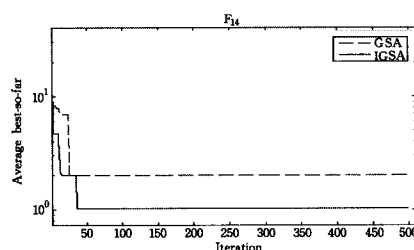


图5 IGSA 和 GSA 在 F_{14} 上的性能比较

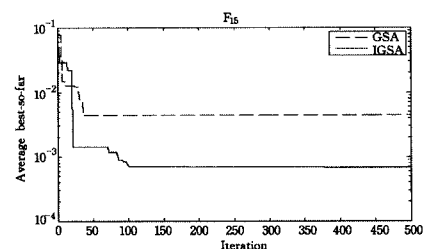


图6 IGSA 和 GSA 在 F_{15} 上的性能比较

4 基于 IGSA 的 SVM 参数优化

SVM 模型建立过程中采用均方误差最小值作为适应度函数,公式如下:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (18)$$

其中, n 为样本数, y_i 为实际输出值, $\hat{y}_i$ 为期望输出值。

IGSA 对 SVM 进行参数优化的步骤如下:

1. 利用混沌序列初始化 SVM 参数(惩罚系数 C 、核参数 σ),形成初始种群,并随机初始化每个粒子的速度;
2. 计算每个粒子的适应度值 fit_i ,将粒子分为 3 个等级 A、B、C,对其分别进行处理,得到新种群;
3. 计算新种群中的每个粒子的质量;
4. 计算万有引力;
5. 计算合力;
6. 更新加速度、速度和位置的值;
7. 返回到步骤 2 循环迭代,直到达到循环次数或要求的精度为止,输出结果。

5 仿真及应用

5.1 函数仿真

为了验证 IGSA 优化 SVM 参数的有效性,对加有噪声的 $\sin c$ 函数($y = \frac{\sin(x)}{x} + \sigma_0 \cdot N(0, 1), \sigma_0 = 0.01$)进行测试,在区间 $[-3, +3]$ 上以 0.06 为等间隔产生 100 个点作为训练样本,以 0.1 为等间隔产生 60 个点作为测试样本,以 MSE 为目标,利用 IGSA 进行参数优化。

设定群体规模 N 为 30,最大迭代次数 T 为 100,初始化模型参数(C, σ)的范围: $C \in [1, 1000], \sigma \in [0, 1]$ 。根据 IGSA 优选所得的参数和训练样本训练 SVM 模型,用该模型对测试样本进行评估,其结果如图 7 和表 2 所示。

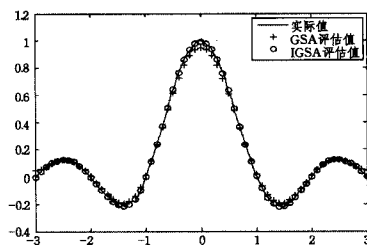


图7 IGSA 和 GSA 在 $\sin c$ 上的评估结果比较

表2 IGSA 和 GSA 优化 SVM 的 MSE 比较

SVM 参数优化方法	训练样本 MSE	测试样本 MSE
GSA	1.1932×10^{-4}	1.7981×10^{-4}
IGSA	7.5243×10^{-7}	7.9603×10^{-7}

从图 7 和表 2 可见,基于 IGSA 的 SVM 模型比基于 GSA 的 SVM 模型的拟合效果更好,精度也更高,因此将基于 IGSA 的 SVM 模型用于火电厂一次风机状态监测。

5.2 基于 IGSA 的 SVM 模型用于火电厂一次风机状态监测

对某火电厂 1# 机组的一次风机进行状态监测,状态监测是用正常工作状态下的数据建模,当设备工作异常时,其动态特性偏离正常工作空间,从而导致残差偏大,当残差超出预先设定的阈值时,发出报警信息,以提示运行人员检查设备状态^[11]。该案例中收集的系統设备测点信息以及历史数据来源于该电厂的厂级监测系统。以一次风机的轴承 X 向振动为例,选取 2012 年 8 月 1 日到 2013 年 2 月 1 日的真实数据(共 20 多万条数据,归一化后如图 8 所示),从图中可知发生了因为故障造成跳机的后果(振动为 0 即为跳机),而且在跳机之前已经出现异常,本文的目的就是能够及时发现异常。

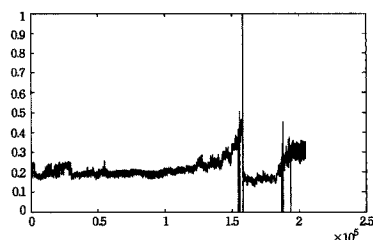


图8 电厂真实数据

选取 $1 \sim 1 \times 10^5$ 中的正常工作状态下的设备所有测点数据作为建模的训练数据(其中 X 向振动为模型的因变量,其他测点数据为自变量),训练数据的 MSE 结果如表 3 所列;

$1 \times 10^5 \sim 1.6 \times 10^5$ 的数据作为测试数据,实际值和估计值曲线如图 9 所示,两值的残差曲线如图 10 所示。

表 3 两种模型在训练数据上的 MSE 比较

建模方法	训练样本 MSE
GSA+SVM	5.0531×10^{-4}
IGSA+SVM	3.4994×10^{-6}

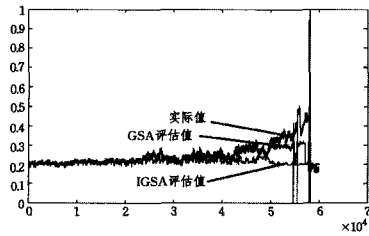


图 9 X 向振动实际值和估计值曲线

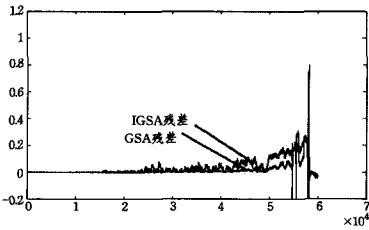


图 10 实际值和估计值的残差曲线

分析表 3 并结合 5.1 节的仿真实验可得,利用正常数据建模时,基于 IGSA 的 SVM 模型比基于 GSA 的 SVM 模型要好。而且从图 9 与图 10 中可以发现:当设备工作在正常状态时,两模型的残差都几乎为 0;但是当设备出现异常时,基于 IGSA 的 SVM 模型的残差比基于 GSA 的 SVM 模型大,当阈值确定后(阈值设定参考文献[11]中第 3 节的残差统计分析),基于 IGSA 的 SVM 模型更容易及时发现异常并报警。因此,基于 IGSA 的 SVM 模型实现了对正常数据拟合更好而对异常数据更敏感的目标。

结束语 本文针对 GSA 容易陷入局部最优,引入了混沌序列和遗传算法的交叉思想,实验结果表明,IGSA 在收敛速度和精度上优于 GSA;然后建立基于 IGSA 的 SVM 模型,通

过实验仿真验证了该模型具有更高的精度。将该模型用于火电厂的一次风机状态监测中,实时地对设备状态做出判断,能够使运维人员及早发现故障隐患,对电厂故障预警具有重大意义。

参考文献

[1] 刘春波,王鲜芳,潘丰. 基于蚁群优化算法的支持向量机参数选择及仿真[J]. 中南大学学报:自然科学版,2008,39(6):1309-1313

[2] 张艳秋,王蔚. 利用遗传算法优化的支持向量机垃圾邮件分类[J]. 计算机应用,2009,29(10):2755-2757

[3] 邵信光,杨慧中,陈刚. 基于粒子群优化算法的支持向量机参数选择及其应用[J]. 控制理论与应用,2006,23(5):740-743,748

[4] Rashedi E, Nezamabadi-pour H, Saryazdi S. GSA: a gravitational Search algorithm[J]. Information Sciences, 2009, 179(13): 2232-2248

[5] Rashedi E, Nezamabadi-pour H, Saryazdi S. BGSA: Binary gravitational search algorithm [J]. Natural Computing, 2010, 9(3): 727-745

[6] Liao Gwo-ching, Tsao Ta-peng. Application Embedded Chaos Search Immune Genetic Algorithm for Shortterm Unit Commitment [J]. Electric Power Systems Research, 2004, 71(2): 135-144

[7] Alatas B, Akin E, Ozer A B. Chaos embedded particle swarm optimization algorithm[J]. Chaos, Solutions and Fractals, 2009, 40(4): 1715-1734

[8] Tang K S, Man K F, Kwong S, et al. Genetic algorithms and their applications[J]. IEEE Signal Processing Magazine, 1996, 13(6): 22-37

[9] 李敏强,寇纪淞,等. 遗传算法的基本理论与应用[M]. 北京:科学出版社,2002

[10] 左兴权,李士勇. 一类自适应免疫进化算法[J]. 控制与决策, 2004, 19(3): 252-256

[11] 郭鹏, Infield D, 杨锡运. 风电机组齿轮箱温度趋势状态监测及分析方法[J]. 中国电机工程学报, 2011, 31(32): 129-136

(上接第 208 页)

[6] 刘万军,张孟华,郭文越. 基于 MPSO 算法的云计算资源调度策略[J]. 计算机工程, 2011, 37(11): 43-44, 48

[7] Yuan H, Li C, Du M. Optimal Virtual Machine Resources Scheduling Based on Improved Particle Swarm Optimization in Cloud Computing[J]. Journal of Software, 2014, 9(3): 705-708

[8] 刘卫宁,靳洪兵,刘波. 基于改进量子遗传算法的云计算资源调度[J]. 计算机应用, 2013, 33(8): 2151-2153

[9] 葛宇,许波,梁静. 基于改进蛙跳策略的 Map-Reduce 作业调度算法[J]. 计算机应用研究, 2013, 30(7): 1999-2002

[10] 师雪霖,徐格. 云虚拟机资源分配的效用最大化模型[J]. 计算机学报, 2013, 36(2): 252-262

[11] 沈时军,刘欣然,张鸿,等. 云计算中的服务可用性保障机制[J].

通信学报, 2014, 35(2): 202-206

[12] 阎高伟,李闯勤,石兵,等. 基于社会力模型的群体优化算法[J]. 控制工程, 2012, 19(6): 1238-1243

[13] Zahafia M, Konwinski A, Joseph A. Improving MapReduce performance in heterogeneous environments [C] // 8th USENIX Symp Osium on Operating Systems Design and Implementation. San Diego, California, USA, 2008

[14] 孟凡超,张海洲,初佃辉. 基于蚁群优化算法的云计算资源负载均衡研究[J]. 华中科技大学学报:自然科学版, 2013, 41(12): 57-62

[15] Cal heiros R N, Ranjan R, Beloglazov A, et al. CloudSim: a tool kit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms [J]. Software: Practice and Experience, 2011, 41(1): 23-50