

多感官群集智能算法及其在前向神经网络训练方面的应用

刘帆^{1,2)} 解仑¹⁾ 李秉杰³⁾ 王志良¹⁾ 郑雪峰¹⁾

1) 北京科技大学信息工程学院, 北京 100083 2) 西安卫星测控中心, 西安 710043

3) 民政部一零一研究所, 北京 101601

摘要 针对连续域函数优化问题, 提出了一种新的全局极大值搜索方法——多感官群集智能算法(multi-sense swarm intelligence algorithm, MSA). 受鱼群算法(artificial fish-swarm algorithm, AFA)和FS算法(free search algorithm, FSA)的启发, MSA的搜索机制将大范围勘察和小范围精确搜索相结合, 个体在使用视觉信息快速逼近局部较优解的同时, 利用嗅觉信息避免群体过于集中并引导个体向全局较优解方向移动. 仿真结果证明: MSA鲁棒性较强, 全局收敛性好, 收敛速度较快, 收敛精度较高. 最后, 将该方法应用于前向神经网络训练, 结果表明满足应用要求.

关键词 多感官群集智能算法; 人工鱼群算法; FS算法; 连续函数优化; 神经网络训练

分类号 TP 301.6

Multi-sense swarm intelligence algorithm and its application in feed-forward neural networks training

LIU Fan^{1,2)}, XIE Lun¹⁾, LI Bingjie³⁾, WANG Zhiliang¹⁾, ZHENG Xuefeng¹⁾

1) School of Information Engineering, University of Science and Technology Beijing, Beijing 100083, China

2) Xian Satellite Control Center, Xian 710043, China

3) No.1 Institute of Domestic Affairs, Beijing 101601, China

ABSTRACT A novel method for global optimization, multi-sense swarm intelligence algorithm (MSA), was presented to solve continuous function optimization problems. Inspired by the artificial fish-swarm algorithm (AFA) and the FS algorithm (free search algorithm, FSA), the search mechanism of MSA combined large scale exploration and local precise search; even more, in this algorithm, the unit employed both visual information for quick approaching to local optimization solution and pheromone information to avoid overcrowding and to guide itself to global solution. Simulation shows that MSA has strong robustness, good global convergence, quick convergence speed and high convergence accuracy. At last, MSA was applied to feed-forward neural network training. The result shows that this algorithm is fit for the application.

KEY WORDS multi-sense swarm intelligence algorithm; artificial fish-swarm algorithm; free search algorithm; continuous function optimization; neural network training

自然界中动物、昆虫常以集体的力量进行觅食生存, 单个个体所表现的行为通常是缺乏智能的, 但由个体组成的群体则表现出了一种有效的复杂的智能行为——群集智能(swarm intelligence). 群集智能可以在适当的进化机制引导下通过个体交互以某种突现形式发挥作用, 这是个体以及可能的个体智

能难以做到的^[1]. 在群集智能的研究基础上, 人们先后提出了多种群集智能算法(swarm intelligence algorithm), 如遗传算法(GA)、蚁群算法(ACO)、粒子群算法(PSO)以及人工鱼群算法(AFA). 群集智能算法将搜索和优化过程模拟成个体的进化或觅食过程, 用搜索空间中的点模拟自然界中的个体; 将求

收稿日期: 2007-09-20 修回日期: 2008-07-30

基金项目: 国家自然科学基金资助项目(No. 60573059); 国家高技术研究计划资助项目(No. 2007AA04Z218)

作者简介: 刘帆(1978-), 男, 硕士研究生; 解仑(1968-), 男, 副教授, 博士, E-mail: ygao@mail.tsinghua.edu.cn

解问题的目标函数度量成个体对环境的适应能力;将个体的优胜劣汰过程或觅食过程类比为搜索和优化过程中用好的可行解取代较差可行解的迭代过程。从而,形成了一种以“生成+检验”特征的迭代搜索算法,它是一种求解极值问题的自适应人工智能技术。自组织性、突现性成为群集智能优化算法的两大基本特征^[2]。本文针对连续域函数优化问题,借鉴 AFA 和 FSA 中的个体行为模式,提出了一种使用视觉和信息素的多感官群集智能算法(multi-sense swarm intelligence algorithm, MSA)。

1 人工鱼群算法和 FS 算法

1.1 人工鱼群算法

人工鱼群算法是由李晓磊、邵之江等提出的一种群集智能优化算法。它采用了自上而下的寻优模式模仿自然界鱼群觅食行为,主要利用鱼的觅食、聚群和追尾行为,构造了个体的底层行为^[3];通过鱼群中各个体的局部寻优,达到全局最优值在群体中突现出来的目的^[4]。

聚群行为模仿了鱼出于生存和躲避危害的需要而自发聚集成群的生活习性。人工鱼个体搜索其可见域范围内的伙伴数目及伙伴中心,如果伙伴中心有较多的食物并且不太拥挤,则朝伙伴的中心位置方向前进一步,否则执行觅食行为。

鱼的追尾行为模仿在鱼群的游动过程中,当其中一条或几条发现食物时,其临近的伙伴会尾随其快速到达食物点。人工鱼个体搜索其可见域内的适应度最大的伙伴,如果该伙伴的食物浓度高且周围不太拥挤,则人工鱼向伙伴的位置方向前进一步,否则执行觅食行为。

觅食行为模仿了鱼在水中随机地自由游动,当发现食物时,则会向食物逐渐增多的方向游去,一般作为人工鱼个体的缺省行为。人工鱼在其可见域随机选择一个状态,如果该状态的适应度大于人工鱼当前适应度,则向该状态方向前进一步。反之,再重新随机选择状态并判断是否满足前进条件。重复几次后如果仍不满足前进条件,则随机移动一步。

人工鱼在选择追尾行为和聚群行为时,一般采用试探的方法,最终以行动结果大的行为为准。

1.2 FS 算法

FS 算法^[5]是由 Penev 和 Littlefair 提出的一种新的进化计算方法。这种算法在求解机制上采用的是一种具有个体意识的、自由的、不确定的个体行为方式,适用于解决实数空间的优化问题,具备鲁棒性强、收敛速度较快等特点。它仿照一些高等群居动

物在进行生物行为时,动物个体的行为特性依靠自身本能——直觉(sense)和运动(mobile),而种群整体仍处于一种有序的进程之中,其中的个体存在着与整体进程相协调的个体随意行为,不考虑群体首领的作用,个体与群体之间保持着协调一致性。

FS 算法(free search algorithm, FSA)中的一次行走包括两类行为:大范围内的勘察和小范围内的搜索。个体首先进行小范围搜索,即在小范围内进行有限步随机搜索,个体位置的修正策略决定于勘察边界和个体在第 i 维解空间中的可变领域值,并且允许以大于 0 的概率在解空间中间进行搜索,从而避免陷入局部最优陷阱。完成有限步随机搜索后,记录最大搜索解,并据此更新信息素表。个体转为进行大范围内的勘察,首先随机生成敏感系数,然后根据敏感系数并参照信息素表选择勘察位置,由此完成一次行走^[6]。

2 多感官群集智能算法(MSA)

2.1 MSA 简介

本文提出的 MSA 借鉴并改进了人工鱼群算法中的个体行为模式和 FS 算法中的搜索策略,是一种基于视觉和嗅觉的全局优化搜索策略,该算法适用于连续域函数优化问题。相对于其他群集算法,MSA 拥有如下特点:

(1) 搜索策略将大范围勘察和小范围精确搜索相结合。通过小范围精确搜索提高个体对极值解的逼近精度,同时通过大范围勘察实现了个体对极值的快速逼近。

(2) 个体采用了基于视觉和嗅觉的勘察模式。个体通过视觉信息分析群体当前搜索结果的同时,也通过嗅觉信息充分了解群体的过往搜索结果,由此提高了群体对全局极值解的逼近速度。

(3) 依据个体与信息素中心间的距离判断群体疏密,促使个体向新的领域探索,防止其陷于局部极大值。

(4) 采用信息素表记录群体搜索过程。在信息素表未满时依次记录最新搜索解,而在信息表已满时依据当前搜索解更新表中最小记录。

(5) 引入了健康因子,使个体的感知范围和移动步长包含随机因素,扩大了个体的随机搜索范围。

2.2 算法流程图

MSA 流程图如图 1,个体搜索流程图如图 2。

2.3 算法步骤

(1) 程序初始化并开始迭代。对程序基本运行参数进行初始化,包括个体数目、个体状态、迭代上

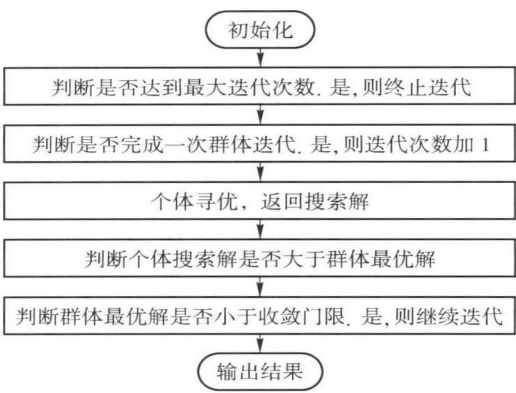


图 1 MSA 流程图
Fig. 1 MSA flow chart

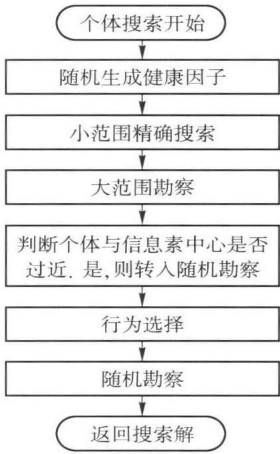


图 2 个体搜索流程图
Fig. 2 Unit search flow chart

限、收敛门限、信息素表记录个数、个体的标准感知范围和标准步长。在实际使用中,须注意以下问题:个体状态是寻优问题的优化解,可采用两种初始化方法:随机赋初值或指定固定值;个体的标准感知范围、标准步长应设为固定比例。对于连续域函数优化问题,参考比例一般可取为 5:1;信息素表记录个数不宜过多,一般应小于等于个体数目。

(2) 判断是否达到迭代上限。是,则退出并返回当前群体最优解;否则继续进行执行。

(3) 判断是否所有个体已轮流搜索过一遍。是,则迭代次数加 1;否则继续执行。

(4) 各个体轮流从当前状态开始搜索,具体流程如下。

① 随机生成健康因子,并据此计算出个体的感知范围和移动步长。各个体在进行每次搜索前,均需先随机生成健康因子。健康因子与个体的嗅觉范围、视觉范围和移动步长直接相关:

$$\text{Health} = \alpha + \beta \text{random}(0, 1) \quad (1)$$

式中,Health 为健康因子, α 、 β 为比例因子且满足 $\alpha + \beta = 1 (0 \leq \alpha \leq 1, 0 \leq \beta \leq 1)$, $\text{random}(0, 1)$ 为区间 $[0, 1]$ 上的均匀分布随机数。

$$\text{Sense} = \text{Health} \times \text{Sense-std} \quad (2)$$

式中, Sense-std 为标准感知范围, Sense 表示个体感知范围。

$$\text{Step} = \text{Health} \times \text{Step-std} \quad (3)$$

式中, Step-std 为个体的标准步长, Step 为移动步长。

② 进行小范围精确搜索。在小范围精确搜索过程中,个体进行有限次随机搜索。在每次搜索中,个体都依据

$$X'_i = X_i + \text{Step} \times (2 \times \text{random}(0, 1) - 1), \quad (4)$$

生成随机的新状态,并通过比较新旧两状态的适应度决定是否进行状态更新。这里提到的适应度代表一个个体所对应解的优劣,通常由适应度函数表示。对于求极大值问题,适应度函数可直接采用待优化函数,也可采用设定的指标函数。式中, $X_i = (x_1, x_2, \dots, x_s)$ 为个体 i 当前状态, s 为个体状态的分量总数, X_j 表示个体状态的第 j 个分量, j 取值范围为 1 到 s 间的整数。 X'_i 表示个体 i 新状态。 X_i 、 X'_i 满足 $d_i = \|X_i - X'_i\| < \text{Step}$ 。

状态更新公式如下:

$$X_i = X'_i \quad (5)$$

③ 进行大范围勘察。小范围精确搜索完成后,个体计算出信息素中心状态和感知范围内的群体中心状态,并搜索出感知范围内拥有最大适应度的相邻个体状态。

信息素中心状态计算公式如下:

$$P_{\text{center}} = \frac{1}{m} \sum_{k=1}^m P_k \quad (6)$$

式中, $P_{\text{center}} = (p_1, p_2, \dots, p_s)$ 为信息素中心状态, $P_j (j = 1, 2, \dots, s)$ 表示信息素的第 j 个分量。 P_k 表示信息素表中第 k 条信息素记录, m 为信息素表中记录总数。

群体中心状态计算公式如下:

$$X_{\text{center}} = \frac{1}{n} \sum_{l=1}^n X_l$$
$$d'_i = \|X_i - X_l\| \leq \text{Sense} \quad (7)$$

式中, X_{center} 为群体中心状态, X_l 为其感知范围内的相邻个体状态, n 为可感知个体总数。

④ 判别是否拥挤。若信息素中心与个体间的距离当前小于某固定值 ϵ (ϵ 远小于标准感知范围),说明个体周围过于拥挤或处于某一极值附近,此时个体直接转入第⑥步,进行随机勘察。

⑤ 行为选择。依次将信息素中心状态、群体中

心状态和最大相邻个体状态代入

$$X_j=X_i+\text{Sense}\times(X'_i-X_i)/\parallel X'_i-X_i\parallel \quad (8)$$

选取其中适应度最大者与个体当前适应度进行比较. 若大于, 则据此更新个体当前状态并返回勘察结果; 否则执行⑥步.

⑥ 随机勘察. 个体在其感知范围内随机选择一个新状态, 如

$$X_j=X_i+\text{Sense}\times(2\text{random}(0,1)-1) \quad (9)$$

若新状态的适应度大于当前值, 则以此更新个体状态并返回勘察结果; 否则继续勘察. 重复几次后如果仍不满足条件, 则以最后一次选择的状态进行更新并返回勘察结果.

(5) 若该返回解大于当前群体最优解, 则将该返回解赋给当前群体最优解.

(6) 更新信息素表. 若信息素表未满, 将返回解直接添加至表中; 若已满, 则仅在返回解的适应度值大于表中最小解时以返回解覆盖表中最小解.

(7) 判断是否达到收敛门限. 是, 则退出并返回当前群体最优解; 否则返回步骤(2)继续执行.

2.4 算法仿真测试

Scharffer's F6 函数^[7]优化问题数学描述如下:

$$\max f(x_1,x_2)=0.5+\frac{(\sin\sqrt{x_1^2+x_2^2})^2-0.5}{[1+0.001\times(x_1^2+x_2^2)]^2} \quad (10)$$

Scharffer's F6 函数有无数个极大指点, 全局极大值点为(0, 0), 最大值为 1. 全局极大值点由一圈圈的脊峰围绕, 其中最内一圈脊峰取值均为 0.990 283, 故极易陷入在此的局部极大点, 是一个很难求解的多峰值优化函数. 该函数的局部三维如图 3 所示.

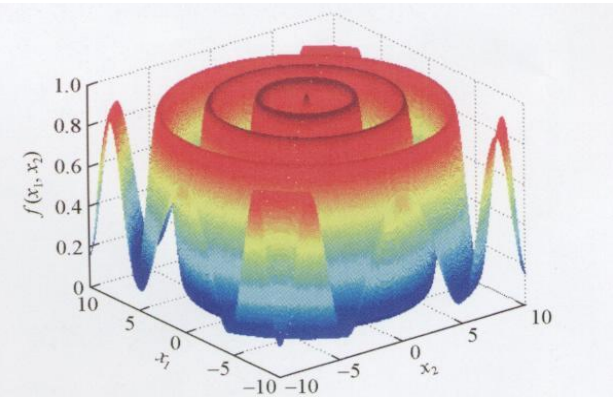


图 3 Scharffer's F6 函数局部三维视图
Fig.3 Scharffer's F6 three-dimensional chart

以 Scharffer's F6 函数优化问题对多感官群集智能算法(MSA)进行测试, 设定个体数为 10 个, 初

始个体状态均设为(-10, -10), 个体标准感知范围为 2.5, 个体标准步长为 0.5, 迭代上限为 500 次, 收敛门限为 0.995, 适应度函数直接取为 Scharffer's F6 函数.

为作比较, 这里同时给出了鱼群算法和 FS 算法测试结果. 计算过程中, 这两种算法所采用的个体数、初始个体状态、迭代上限和迭代门限均与 MSA 计算时相同.

测试计算各进行 10 次, 三种算法的计算结果表 1~3. 三种算法的综合比较结果如图 4 所示.

表 1 MSA 测试结果
Table 1 MSA simulation results

序号	迭代次数	搜索解	序号	迭代次数	搜索解
1	88	0.995 2	6	104	0.995 7
2	137	0.996 3	7	48	0.997 1
3	60	0.995 6	8	125	0.997 7
4	49	0.998 9	9	42	0.999 5
5	92	0.996 2	10	24	0.995 7

表 2 基本 AFA 测试结果
Table 2 AFA simulation results

序号	迭代次数	搜索解	序号	迭代次数	搜索解
1	65	0.996 4	6	48	0.997 2
2	91	0.995 5	7	43	0.995 6
3	128	0.999 9	8	121	0.999 2
4	25	0.999 3	9	47	0.999 0
5	71	0.997 5	10	160	0.997 7

表 3 FSA 测试结果
Table 3 FSA simulation results

序号	迭代次数	搜索解	序号	迭代次数	搜索解
1	202	0.998 8	6	159	0.998 9
2	119	0.996 5	7	88	0.998 1
3	160	0.996 4	8	67	0.999 4
4	209	0.999 5	9	34	0.998 7
5	146	0.998 4	10	203	0.996 1

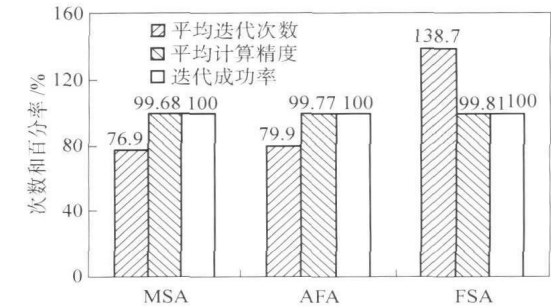


图 4 三种算法计算结果的比较
Fig.4 Comparison of results by three algorithms

数据显示,上述三种算法在测试过程中均未出现陷于局部极值的情况.当参数设定接近时,MSA 与基本 AFA 在计算速度和计算精度方面性能是比较接近的.而相对于 FS,MSA 的计算速度要稍快一些.可见,多感官群集智能算法具备鲁棒性强、全局收敛性好、收敛速度较快、收敛精度较高等优点.

3 使用 MSA 训练前向神经网络

前向神经网络是整个神经网络体系中最常见的一种网络结构,其网络中各个神经元接受前一级的输入,并输出到下一级,网络中没有反馈,可以用有向无环路图表示(如图 5).这种网络实现信号从输入空间到输出空间的变换,它的信息处理能力来自于简单非线性函数的多次复合.

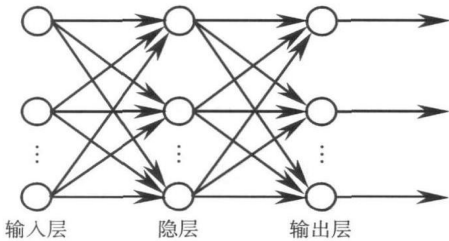


图 5 前向神经网络
Fig.5 Feed-forward neural network

前向神经网络训练的训练问题实际上是一个无约束连续域函数优化问题,目前对其训练算法的研究主要集中在以下三方面^[8]:

- (1) 改进基于梯度信息的一阶算法(BP 算法),如增加动量项或采用变学习率,这种算法一般都比较有效,但是收敛速度仍较慢,且属于局部搜索算法,从本质上仍然摆脱不了陷入局部极小的可能;
- (2) 改进并引入新的二阶算法,如拟 Newton 法、L-M 算法等,这些算法因为充分利用了二阶导数信息,具有收敛速度快、逼近精度高的优点,但仍属局部搜索算法;
- (3) 引入和改进具备全局优化特性的群集算法,如遗传算法、蚁群算法、粒子群算法以及人工鱼群算法^[9],本文将多感官群集智能算法(MSA)应用于神经网络训练就属于这个研究方向.

3.1 前向神经网络的 MSA 训练方法

以三层神经网络为例,训练方法如下.

(1) 根据设定网络结构,构造个体状态 $X_i = \{w_{21}, b_2, w_{32}, b_3\}^{[10]}$. 其中, w_{21} 为输入层到隐含层权值矩阵、 w_{32} 为隐含层到输出层权值矩阵、 b_2 为隐含层阈值向量、 b_3 为输出层阈值向量,每个个体代表一个神经网络.

(2) 生成训练样本集 $K = \{r^1 = (x^1, y^1), r^2 = (x^2, y^2), \dots, r^n = (x^n, y^n)\}$, 适应度函数可取为 $E = 1 \sqrt{\frac{1}{2} \sum_{j=1}^n (y^j - \hat{y}^j)^2}$ 或 $E = -\frac{1}{2} \sum_{j=1}^n (y^j - \hat{y}^j)^2$.

(3) 程序初始化. 设置个体数目、个体状态、最大迭代次数、收敛门限、个体标准感知范围、个体标准步长以及信息素表记录个数.

(4) 调用多感官群集算法进行群体寻优,当达到收敛门限或迭代上限时,停止迭代并输出结果.

3.2 网络训练及结果分析

这里选取训练前向神经网络逼近 sine 函数来评价 MSA 的训练性能. 设定神经网络结构为 1-4-1, 输入层有 1 个神经元, 隐含层有 4 个神经元, 激励函数为 sigmoid, 输出层有 1 个神经元, 激励函数为 pureline. 网络结构如图 6 所示.

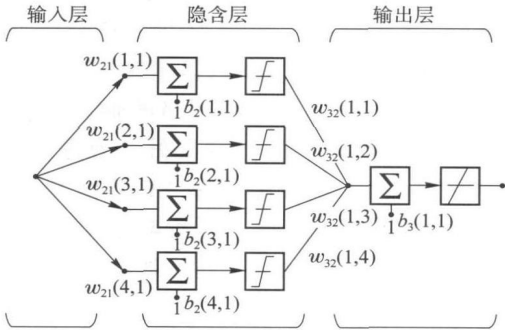


图 6 前向神经网络结构图
Fig.6 Feed-forward neural network

应用 MSA, 设置个体数为 30, 个体标准感知范围为 2.5, 个体标准步长为 0.5, 个体状态初始化为 $[-5, 5]$ 内均匀分布的随机数, 每个状态都包含 13 个状态变量. 样本点选为 $[0, 2\pi]$ 上等间距分布的 500 个点. 算法迭代及训练结果如图 7 和图 8.

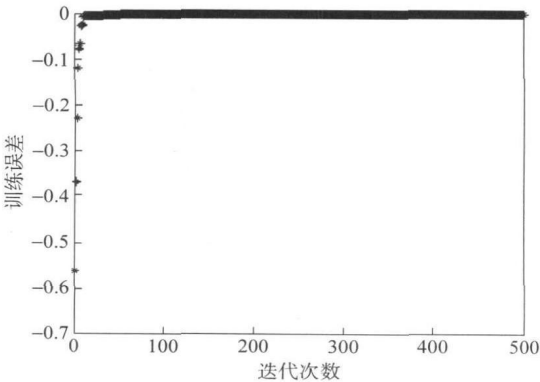


图 7 MSA 迭代情况
Fig.7 MSA iteration

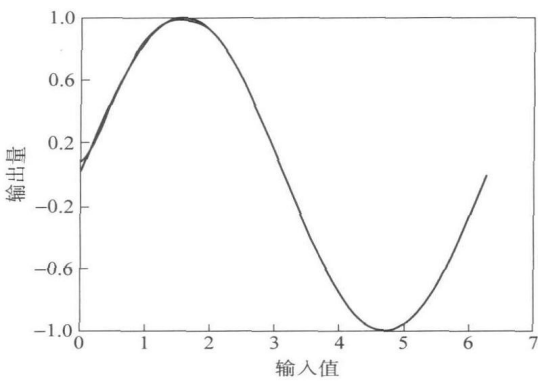


图 8 神经网络训练结果
Fig·8 Neural network training

由图 7 可以看出,MSA 的训练速度在起始阶段是比较快的,实际上当迭代次数达到 50 次时训练误差已降至 0.000 6,当达到 100 次时训练误差降至 0.000 4. 但在此后的迭代过程中,训练误差并未进一步减小. 这主要是受 MSA 的搜索机制所限制,它只能达到满意解域;当需要获得精确解时,则必须借助其他算法. 图 8 为所训练网络的输出特性,可见网络输出与样本输出符合得很好. 为进一步检验 MSA 的训练性能,将变学习率 BP 算法和 MSA 对 sine 函数的实验结果对比,如表 4. 结果表明,相对于变学习率 BP 算法,使用 MSA 训练前向神经网络速度更快,精度更高.

表 4 前向神经网络训练误差

Table 4 Feed-forward neural network training errors

迭代次数	变学习率 BP 算法	MSA
10	0.221 4	0.012 4
20	0.016 7	0.001 0
50	0.008 8	0.000 6
100	0.001 6	0.000 4

4 结 论

多感官群集智能算法是一种新的针对连续域函数优化问题的全局极大值搜索方法,它鲁棒性强,全局收敛性好,收敛速度较快,收敛精度较高. 将其应用于前向神经网络训练,结果显示其满足应用要求.

在实验过程中,多感官群集智能算法也暴露出一些问题. 首先是该算法只能达到满意解域,当迭代精度要求较高时计算效率较低;其次对于复杂

网络结构——高维函数优化问题,该算法存在收敛速度慢、收敛精度低的缺点. 这两点是多感官群集智能算法需要改进的方面.

参 考 文 献

[1] Duan H B, Wang D B, Yu X F. Research on some novel bionic optimization algorithms. *Comput Simul*, 2007, 24(3): 169
(段海滨,王道波,于秀芬.几种新型仿生优化算法的比较研究.计算机仿真,2007,24(3):169)

[2] Zhang T H, Lu X Y. Prospect and research for swarm intelligence optimum algorithm. *Shanxi Archit*, 2007, 33(1): 14
(张统华,鹿晓阳.群体智能优化算法的研究进展与展望.山西建筑,2007,33(1):14)

[3] Li X L, Shao Z J, Qian J X. An optimizing method based on autonomous animates: fish-swarm algorithm. *Syst Eng Theory Pract*, 2002, 22(11): 32
(李晓磊,邵之江,钱积新.一种基于动物自治体的寻优模式:人工鱼群算法.系统工程理论与实践,2002,22(11):32)

[4] Li X L, Qian J X. Studies on artificial fish swarm optimization algorithm based on decomposition and coordination techniques. *J Circuits Syst*, 2003, 8(1): 1
(李晓磊,钱积新.基于分解协调的人工鱼群优化算法研究.电路与系统学报,2003,8(1):1)

[5] Gao B, Tian D, Hu L, et al. Free Search: a novel evolutionary algorithm. *Chin J Sci Instrum*, 2006, 27(Suppl 1): 819
(高波,田东,胡亮,等.Free Search:一种新的进化算法.仪器仪表学报,2006,27(增刊1):819)

[6] Penev K, Littlefair G. Free search: a comparative analysis. *Inf Sci*, 2005, 172(122): 173

[7] Sachffer J D, Caruana R, Eshelman L J, et al. A study of control parameters affection online performance of genetic algorithms for function optimization// *Proceedings of the 3rd International Conference on Genetic Algorithm*. San Diego, 1991: 51

[8] Jia L H, Zhang X R. Analysis and improvements of BP algorithm. *Comput Technol Dev*, 2006, 16(10): 101
(贾丽会,张修如.BP算法分析与改进.计算机技术与发展,2006,16(10):101)

[9] Guo J. *Studies on BP Neural Network Algorithm* [Dissertation]. Wuhan: Huazhong University of Science and Technology, 2005: 8
(郭军.BP神经网络算法研究[学位论文].武汉:华中科技大学,2005:8)

[10] Ma J W, Zhang G L, Xie H, et al. Optimization of feed-forward neural networks based on artificial fish-swarm algorithm. *J Comput Appl*, 2004, 24(10): 21
(马建伟,张国立,谢宏,等.利用人工鱼群算法优化前向神经网络.计算机应用,2004,24(10):21)