

一种新的粒子群算法与人工鱼群算法的混合算法

袁光辉¹, 樊重俊¹, 张惠珍¹, 王斌², 覃太贵²

(1. 上海理工大学 管理学院, 上海 200093; 2. 三峡大学 理学院, 宜昌 443002)

摘要: 通过分析粒子群算法和人工鱼群算法的优缺点, 利用粒子群算法收敛速度快及人工鱼群算法能较好地收敛到全局最优解的特点, 提出了一种新的混合算法. 算法以粒子群为基础进行设计, 根据人工鱼群的公告板、群聚和随行策略的模式对粒子群进行速度与位置变更, 使原有的粒子群变成具有一定智能的粒子, 从而达到提高搜索精度及效率的目的. 通过 Generalize-Schwefel 等 3 个经典函数进行优化仿真后发现, 该混合算法具有搜索精度更高及收敛速度更快的特点, 同时该算法在求解高维问题时具有明显优势.

关键词: 粒子群算法; 人工鱼群算法; 混合算法

中图分类号: TP 301 **文献标志码:** A

Hybrid Algorithm Integrating New Particle Swarm Optimization and Artificial Fish School Algorithm

YUAN Guang-hui¹, FAN Chong-jun¹, ZHANG Hui-zhen¹, WANG Bing², QIN Tai-gui²

(1. Business School, University of Shanghai for Science and Technology, Shanghai 200093, China;

2. College of Science, China Three Gorges University, Yichang 443002, China)

Abstract: A new hybrid algorithm with fast convergence speed and capability of searching optimal solution within defined space was proposed by uniting the advantages of particle swarm optimization and artificial fish swam algorithm. In the new algorithm, the velocity and position of the particle swarm were modified in optimization according to the bulletin boards, cluster of artificial fish and accompanying strategy model. Then the original particle swarm was turned to intelligent particles and the search precision and efficiency were improved. The simulations prove that the new hybrid algorithm possesses the characteristic of higher accuracy search and faster convergence by using three classic functions in optimization, like Generalize-Schwefel function etc.

Key words: particle swarm optimization (PSO); artificial fish school algorithm (AFSA); hybrid algorithm

收稿日期: 2013-06-08

基金项目: 上海市教委科研创新(重点)资助项目(14ZZ131); 上海市研究生创新基金资助项目(JWCXSL1302)

第一作者: 袁光辉(1987-), 男, 硕士研究生. 研究方向: 信息系统工程. E-mail: flame_yuan@163.com

通讯作者: 樊重俊(1963-), 男, 教授. 研究方向: 信息系统工程. E-mail: cjfan@sh163.net

粒子群算法(particle swarm optimization, PSO)是一种基于群体行为的算法,是模拟鸟类的觅食行为设计提出的^[1].该算法在使用时具有实现过程简单、设计参数较少、粒子群规模较小,且收敛速度快等特点,因此受到了广泛的研究与应用.同时在研究过程中发现,正是由于其收敛速度快这一特点导致其算法容易陷入局部最优,引起搜索精度下降等问题.针对这些问题目前有一些研究者致力于该算法的改进研究方面,对其研究可分为两类^[2-4]:一是在原始算法的基础上增加一些改进策略进行改进;二是将 PSO 算法与其它的优化算法混合而成一种混合算法.

人工鱼群算法(artificial fish school algorithm, AFSA)模拟鱼群的觅食行为,由我国学者李晓磊等^[5]于 2002 年提出.该算法具有较好的全局最优解的求解能力、对初始值和参数要求较低、鲁棒性强、易操作等特点,在优化计算的多个领域得到了广泛应用.目前对 AFSA 的较多研究发现,虽然其全局搜索能力强,但其收敛速度往往较慢.对其研究可归为两类^[6]:一是在基本算法基础上对初始设置或参数进行改进以达到改进其性能的目的;二是将 AFSA 算法和其它算法进行混合构造新的混合算法.

目前,PSO 和 AFSA 两种算法已进行了多种改进,对这两种算法的混合应用也有一些研究.其中罗德相等^[7]于 2009 年提出的 PSO 和 AFSA 混合算法将开始随机产生的种群分成两个子种群.其中一组按照粒子群算法进行计算,另一组按照人工鱼群算法进行计算.在计算结束后比较最优值,以最优值作为公告板的值,用于下一次比较.其所引出的变异因子是将公告板的值加上一个小于变异因子的随机数赋值给 5 个适应度最差的粒子.这一混合改进将两算法原有的不足进行了一定的调和,但有可能加强局部最优解的聚集性,进而导致算法失效.按照两个单独种群使用不同方法进行搜索,只是将两种算法进行了值的比较,没有从根本上解决这两个算法的缺点,还有可能将算法过早地引入局部最优,导致算法失效.

基于 PSO 和 AFSA 算法特点,以及这两者的混合算法研究,本文提出一种新的粒子群-人工鱼群混合算法(PSO-AFSA).该算法保留了 PSO 的快速收敛性、易实现性,继承了 AFSA 的全局收敛性,进行协同搜索,同时克服了文献^[7]的不足.仿真实验结果表明,PSO-AFSA 算法是一种比较有效的优化算法.

1 粒子群算法

PSO 算法假设每一个潜在解是搜索空间内的一只鸟,称为粒子.每个粒子具有由被优化函数确定的适应度值,同时每个粒子具有自身的速度用来决定其飞行方向及距离.空间内的粒子跟随着适应度最高的粒子对自身的速度及位置进行调整^[8].

PSO 求解过程为初始化一群粒子(生成随机解),然后按照一定既定规则搜索最优解.在每次迭代过程中,各个粒子通过跟踪自身最优解和群体最优解来更新自身位置及速度.在选取全局最优解时,可选用粒子自身周围邻居作为小的群体,使用局部最优解代替全局最优解.

本文使用的粒子群算法为 Shi 等^[9]于 1998 年提出的一种改进的粒子群算法.该改进通过基本 PSO 算法增加了一个惯性因子,用来调和目前速度与下一速度的跳跃,同时也对全局搜索能力和局部搜索能力起到了一个平衡作用.其更新公式为

$$\left. \begin{aligned} v_{i+1,d} &= wv_{i,d} + c_1 r_1 (p_{i,d} - x_{i,d}) + \\ &\quad c_2 r_2 (g_{g,d} - x_{i,d}) \\ x_{i+1,d} &= x_{i,d} + v_{i,d} \end{aligned} \right\} \quad (1)$$

式中, v 代表速度; x 代表粒子所处位置; c_1 和 c_2 表示学习因子,亦称为加速常数; r_1 和 r_2 是 $[0,1]$ 区间内的随机数; w 表示惯性因子; $p_{i,d}$ 表示粒子当前自身最优位置; $g_{g,d}$ 表示粒子当前全局最优位置.根据经验,通常 $c_1 = c_2 = 2$. $v_{i,d}$ 表示第 i 代粒子中第 d 个粒子的当前速度($i = 1, 2, \dots, n; d = 1, 2, \dots, D$), $v_{i,d} \in [-v_{\max}, v_{\max}]$, $v_{\max}$ 是常数,由用户设定用来限制粒子的速度.

2 人工鱼群算法

AFSA 模拟鱼群觅食过程,由觅食现象、聚群现象和追尾现象来构造最佳寻优策略,通过各个鱼寻优的过程,达到找到全局最优值的目的.这一算法的相关应用表明,AFSA 具有较强的收敛性.

2.1 定义

在算法中定义每条鱼的个体状态 $\mathbf{X} = (x_1, x_2, \dots, x_n)$,表示鱼所处位置.每条鱼的适应度用当前食物浓度表示,记为 $Y = f(\mathbf{X})$;任意 2 个个体间的距离使用 $d_{ij} = \|\mathbf{X}_i - \mathbf{X}_j\|$ 表示; Va 表示感知距离; Tr 表示每次移动最大试探次数; S 表示鱼移动的最大步长; δ 表示拥挤度因子.

2.2 人工鱼群行为

a. 觅食行为

设每条人工鱼的当前状态为 $\mathbf{X}_i$, 在其可感知距离内随机选择另一个状态 $\mathbf{X}_j$ 的人工鱼, 并求其最小值. 当 $Y_i > Y_{i+1}$ 时, 向随机选的鱼方向前进一步; 否则重新选择一个 $\mathbf{X}_{i+1}$. 当重新进行 Tr 次尝试后都无法找到, 则随机前进一步. 定义觅食行为 AF_pre, 其表达式为

$$\left. \begin{aligned} \mathbf{X}_{i,i+1} &= \mathbf{X}_i + \frac{\text{Rand}() S(\mathbf{X}_{i+1} - \mathbf{X}_i)}{d_{i,i+1}}, & Y_i > Y_{i+1} \\ \mathbf{X}_{i,i+1} &= \mathbf{X}_i + \text{Rand}() S, & \text{否则} \end{aligned} \right\} \quad (2)$$

b. 聚群行为

设每条人工鱼的当前状态为 $\mathbf{X}_i$, 搜索其当前所在邻域(即 $d_{i,i+1} < Va$)的人工鱼数目 n_i , 以及该邻域的中心 $\mathbf{X}_c$, 对应的中心值为 Y_c . 若 $Y_c n_i < \delta Y$, 说明相邻中心拥有食物且相对不拥挤, 则该中心位置前进一步; 否则执行觅食行为 AF_pre₁. 其表达式为

$$\left. \begin{aligned} \mathbf{X}_{i,i+1} &= \mathbf{X}_i + \frac{\text{Rand}() S(\mathbf{X}_{i+1} - \mathbf{X}_i)}{|\mathbf{X}_c - \mathbf{X}_i|}, & Y_c n_i < \delta Y \\ \text{AF_pre}_1, & & \text{否则} \end{aligned} \right\} \quad (3)$$

c. 追尾行为

设每条人工鱼的当前状态为 $\mathbf{X}_i$, 搜索其当前所在邻域(即 $d_{i,i+1} < Va$)的人工鱼中最小值 $\mathbf{X}_{\min}$ 为最小的 $Y_{\min}$, 若 $Y_{\min} n_i < \delta Y$, 说明 $\mathbf{X}_{i+1}$ 拥有食物且相对不拥挤, 则该 $\mathbf{X}_{\min}$ 处前进一步; 否则执行觅食行为 AF_pre₂. 其表达式为

$$\left. \begin{aligned} \mathbf{X}_{i,i+1} &= \mathbf{X}_i + \frac{\text{Rand}() S(\mathbf{X}_{i+1} - \mathbf{X}_i)}{|\mathbf{X}_{\min} - \mathbf{X}_i|}, & Y_{\min} n_i < \delta Y \\ \text{AF_pre}_2, & & \text{否则} \end{aligned} \right\} \quad (4)$$

d. 随机行为

在可感知距离内随机选择一个鱼, 然后向该鱼位置移动.

e. 公告板

设立公告板, 表示并记录最优个体状态及其所在位置的食物浓度. 每条鱼在更新自身位置后将把自身状态与公告板进行对比, 若自身的状态较公告板状态优秀, 则更新公告板状态. 这样可以使公告板策略一直为最优.

3 粒子群-人工鱼群算法

3.1 算法描述

人工鱼群算法与粒子群算法均是借助于研究生物种群的群体行为进行设计的, 源于人工生命和演

化计算理论. 前者是根据鱼的觅食行为构造算法, 通过鱼群变换行为来寻找全局最优解, 能较快地找到最优解. 但该算法具有较高的时间复杂性, 不利于高维数问题的求解. 后者模拟鸟群的觅食行为, 通过鸟的协作行为找到最优解. 该算法运行效率高且容易实现, 但是仅通过更新位置和速度来寻找全局最优解, 因可调参数少、无需梯度信息等不足, 常造成结果早熟, 无法收敛到全局最优. 因此本文通过将两种算法的搜索策略进行混合使用, 可以在一定程度上提高求解精度和算法收敛速度.

3.2 算法原理

定义本算法中所使用粒子为智能粒子. 在本算法中, 智能粒子在搜索食物时的速度转换策略为, 以智能粒子的“飞行”速度、个体极值(粒子迄今为止搜索到的最优位置)、全局极值(整个粒子群迄今为止搜索到的最优位置)3个子速度进行加权求和计算得到下一步的飞行速度. 在该更新策略下, 引入人工鱼群算法的觅食行为策略、聚群行为策略、追尾行为策略、随机行为策略、公告板策略等觅食行为. 为避免智能粒子群结果早熟和局部最优的出现, 同时依靠粒子的更新位置和速度来不断进化加速智能粒子群的收敛, 使得混合后的算法不仅具有快速的搜索速度, 同时也保证了算法的全局收敛性能.

假设在一个 D 维的目标求解空间中, 初始种群大小为 n , 每个智能粒子用一个 D 维的向量表示:

$$\mathbf{X}_i = (x_{i1}, x_{i2}, \dots, x_{id}), i = 1, 2, \dots, n$$

第 j 个智能粒子的“飞行”速度也是一个 D 维向量, 记为

$$\mathbf{V}_i = (v_{i1}, v_{i2}, \dots, v_{id})$$

个体极值: 第 i 个智能粒子截至目前搜索到的最优位置, 记为

$$\mathbf{P}_{\text{best}} = (p_{i1}, p_{i2}, \dots, p_{id})$$

全局极值: 整个种群截至目前搜索到的最优位置, 记为

$$\mathbf{g}_{\text{best}} = (g_{g1}, g_{g2}, \dots, g_{gd})$$

式中, g 为迭代次数, $g = 1, 2, \dots, m$.

在以上变换完成之后, 根据 AF_pre₁ 或 AF_pre₂ 变换粒子所在位置. 变换后速度和位置为

$$\left. \begin{aligned} v_{i+1,d} &= wv_{i,d} + c_1 r_1 (p_{i,d} - x_{i,d}) + \\ &\quad c_2 r_2 (g_{g,d} - x_{i,d}) \\ x_{i+1,d} &= \alpha x_{i,d} + \beta(\text{AF_pre}_1) + \gamma(\text{AF_pre}_2) \end{aligned} \right\} \quad (5)$$

式中, α 和 β (其中 β 与 γ 有且仅有一个非零) 为位置转换权重 ($\alpha + \beta = 1$), 使用该权重主要是避免粒子群的早熟或局部最优. 该权重在变化时, α 的值为增

函数.

3.3 粒子群-人工鱼群算法流程

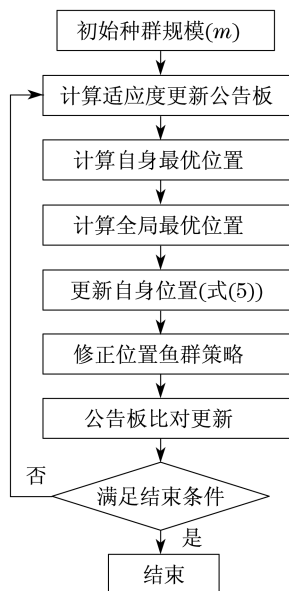


图1 算法流程图

Fig.1 Algorithm flowchart

4 算法测试及结果分析

为测试本文算法的有效性,应用3个经典函数优化问题进行仿真求解:Rosenbrock函数最小值求解并与文献[10]中结果进行对比;大海捞针问题求解并与文献[11]中结果进行对比;Generalize-Schwefel函数求解并与文献[2]中结果进行对比.为提高可比性,在计算时均采用与参考文献相同的维度.

4.1 Rosenbrock函数求解最优化问题

$$\min f(x) = \sum_{j=2}^D 100(x_{j-1} - x_j^2) + \sum_{j=1}^D (1 - x_j)^2, \\ -30 \leq x_j \leq 100$$

此函数为单峰值函数,较为复杂,向量 $\mathbf{X}$ 的维度同样取10,20,30进行对比,其中种群大小选100进行求解.惯性系数取0.8,转换权重 α 取0.6, β 取0.4,运算次数上限取5000次.运算结果对比见表1,维度为10时仿真计算收敛图如图2所示.从图中对比可以看出本算法的收敛速度较文献[10]的收敛速度快,结果更优.

4.2 大海捞针问题求解

$$\max Z = \left[\frac{3}{0.05 + (x_1^2 + x_2^2)} \right]^2 + (x_1^2 + x_2^2)^2$$

设定搜索范围为 $[-8, 8]$,种群规模为50,惯性

表1 Rosenbrock函数在不同维数下的对比

Tab.1 Contrast of Rosenbrock function under different dimensions

维数	本算法	文献[10]算法	文献[7]算法
10	0.793 141	8.248 203	7.357 601
20	11.258 611	18.419 688	16.336 218
30	20.832 151	28.586 522	29.831 218

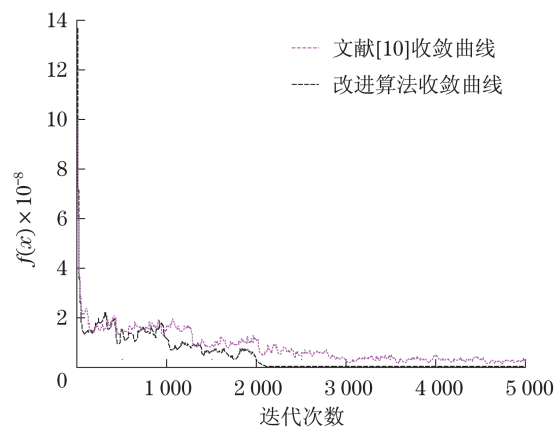


图2 Rosenbrock函数在维数规模为10时对比图

Fig.2 Contrast figure of Rosenbrock function in the dimension of size 10

系数取0.8,转换权重 α 取0.6, β 取0.4,最大搜索次数为2000.优化结果为2748.6953,与文献[11]的结果相近的情况下,计算速度高于其10%左右.

4.3 Generalize-Schwefel问题求解

$$f(x) = \sum_{j=1}^D (x_j \sin \sqrt{|x_j|})$$

依照文献[2]选择计算维数为2,搜索范围为 $[-500, 500]$.种群规模为50,惯性系数取0.8,转换权重 α 取0.6, β 取0.4,最大搜索次数为2000.

优化结果为-837.9736,与文献[2]基本相同,略高于参考文献0.01,计算效率高于其5%.

通过算例对比,可以说明算法是有效的.然而本算法的优势在于高维运算,当Generalize-Schwefel问题达到1000维时,搜索计算的优势将更加明显.

5 结束语

分析粒子群算法及人工鱼群算法优缺点基础上提出了两者的混合算法.本算法在作仿真计算时,将全部优化问题转化为求解最大值问题.通过Generalize-Schwefel问题、Rosenbrock问题进行不同维数多次的仿真计算,最大计算维数达到

(下转第238页)