

带收敛因子的变邻域粒子群算法

范成礼, 邢清华, 范海雄, 李响

(空军工程大学 防空反导学院, 西安 710051)

摘要: 针对基本粒子群算法在求解高维空间中的复杂多峰函数时容易发生早熟收敛而陷入局部最优的问题, 汲取变邻域搜索算法全局搜索的优势, 提出了带收敛因子的变邻域粒子群算法. 首先由基本粒子群的快速搜索能力得到较优的群体; 然后通过收敛因子判断发生早熟收敛的粒子, 并利用变邻域搜索算法的全局搜索能力对陷入早熟收敛的粒子进行优化, 从而得到全局最优. 相关实验表明, 带收敛因子的粒子群算法的性能较常规粒子群算法更加优越.

关键词: 粒子群优化; 变邻域搜索; 收敛因子; 全局搜索

中图分类号: TP182; TP391

文献标志码: A

Particle swarm optimization and variable neighborhood search algorithm with convergence criterions

FAN Cheng-li, XING Qing-hua, FAN Hai-xiong, LI Xiang

(School of Air and Missile Defense, Air Force Engineering University, Xi'an 710051, China. Correspondent: FAN Cheng-li, E-mail: aw7983@163.com)

Abstract: For the complex multi-peaks function with high dimension, the particle swarm optimization and variable neighborhood search algorithm with convergence criterions(VNS-PSO-CC) is proposed on the basis of analyzing the problem of premature. This method combines the particle swarm optimization(PSO) with the global search ability of variable neighborhood search(VNS) algorithm, and adds the convergence criterions. Firstly, the preferable swarm is obtained by using the fast searching ability of PSO algorithm. Furthermore, the premature swarm, which is estimated by convergence criterions, is optimized by using VNS algorithm. Finally, experimental results show that the performance of VNS-PSO-CC algorithm is superior to the traditional PSO algorithm.

Key words: particle swarm optimization; variable neighborhood search; convergence criterions; global search

0 引言

粒子群优化(PSO)算法是一种基于社会群体行为的智能优化算法, 由 Kennedy 等^[1-2]于 1995 年首次提出. 由于 PSO 算法在易用性、时效性和创新性等方面的突出表现, 使其在优化问题中取得了巨大的成功, 现已成为国内外研究的热点^[3-10].

与其他进化算法类似, PSO 算法在解决高维空间中的复杂多峰函数时, 往往存在容易发生早熟收敛而陷入局部最优的问题. 针对这一问题, 近几年来人们提出了许多改进算法^[4-6]. 但这些改进算法大多着眼于 PSO 的参数选择或某个参数的动态修改策略, 难以克服 PSO 算法易陷入局部极小的固有弱点. 文献 [7] 提出了将混沌优化与 PSO 相结合的 CPSO 算法, 在低维函数优化中效果良好, 但对于高维函数优

化则效果不佳. 文献 [8] 提出了一种基于动态邻居和变异因子的粒子群算法, 该算法扩大了粒子群的探索空间, 加强了群的多样性, 但其本身受到邻居个数的限制. 变邻域搜索(VNS)算法是一种求解优化问题的启发式算法, 由 Mladenovic' 等^[11]于 1997 年首次提出, 其基本思想是系统地改变邻域结构集来拓展搜索范围, 包括局部搜索和动态改变邻域两部分.

对于地形复杂且含有较多局部最优的函数优化问题, 现有的 PSO 算法之所以容易发生早熟收敛, 是因为大量粒子聚集在局部最优附近而难以搜索其邻域空间. 对于这类函数, 全局最优往往分布在局部最优的附近, 因此规避早熟收敛、提高优化精度的主要办法是加强粒子的邻域搜索能力. 鉴于此, 本文将 PSO 算法的快速搜索能力与 VNS 算法^[12-16]的全局搜

收稿日期: 2012-12-25; 修回日期: 2013-04-13.

基金项目: 国家自然科学基金项目(61272011); 全军军事学研究生课题项目(2012JY003-577).

作者简介: 范成礼(1988-), 女, 博士生, 从事智能信息处理、信息融合的研究; 邢清华(1966-), 女, 教授, 博士生导师, 从事防空作战仿真建模、防空作战智能决策等研究.

索能力合理结合,通过聚类操作以及聚合度判断,提出了带收敛因子的变邻域粒子群算法(VNS-PSO-CC),以有效避免常规PSO算法在求解过程中容易发生早熟收敛而陷入局部最优的问题.典型测试函数的仿真实验和对比分析结果表明,VNS-PSO-CC算法的性能优于常规粒子群算法.

1 基本 PSO 算法

考虑优化一个含 m 个变量的连续函数,其搜索空间为 d 维连续空间,对其使用 PSO 算法进行优化.设函数优化问题为

$$\begin{aligned} \min f(X), X = [x_1, x_2, \dots, x_m]; \\ \text{s.t. } x_i \in [a_i, b_i], i = 1, 2, \dots, m. \end{aligned}$$

其中: $f(X)$ 为目标函数, d 为自变量 x_i 的维数, $[a_i, b_i]$ 为 x_i 的搜索范围.设种群由 m 个粒子组成,群中每个粒子 X_i 代表目标函数 $f(X)$ 的一个候选解.

粒子的速度和位置更新公式如下:

$$\begin{cases} V_i(t+1) = \omega V_i(t) + c_1 \times \text{rand}() \times (P_i - X_i(t)) + \\ \quad c_2 \times \text{rand}() \times (P_g - X_i(t)), \\ X_i(t+1) = X_i(t) + V_i(t+1). \end{cases}$$

其中: $X_i = (x_{i1}, x_{i2}, \dots, x_{id})$ 为第 i 个粒子在 d 维解空间的位置; $V_i = (v_{i1}, v_{i2}, \dots, v_{id})$ 为第 i 个粒子的速度; $P_i = (p_{i1}, p_{i2}, \dots, p_{id})$ 为第 i 个粒子从初始到当前搜索到的最优位置; $P_g = (p_{g1}, p_{g2}, \dots, p_{gd})$ 为整个粒子种群搜索到的最优位置; c_1 和 c_2 为加速因子,通常取值为 2; $\text{rand}()$ 为 $[0, 1]$ 区间的随机数; ω 为惯性权重系数,是粒子保持原先速度的能力.对于 ω , 本文采用线性递减权重(LDW)策略,其具体形式为

$$\omega_{\text{LDW}} = \omega_{\text{ini}} - \frac{(\omega_{\text{ini}} - \omega_{\text{end}})}{t_{\text{max}}} \times t. \quad (1)$$

其中: ω_{ini} 为初始权重, ω_{end} 为最终权重, t_{max} 为最大迭代次数, t 为当前迭代次数.典型取值为 $\omega_{\text{ini}} = 0.9$, $\omega_{\text{end}} = 0.4$.

基本粒子群算法存在很多缺陷,如对环境的变化不敏感,常常会受到 P_i 和 P_g 的影响而很难收敛到全局最优.实际上,许多高维空间中的复杂多峰函数在全局最优附近往往分布着一些局部最优点,这些局部最优将很容易吸引住粒子群,因此算法容易发生早熟收敛而陷入局部最优,算法优化精度难以提高.

2 VNS 算法

2.1 邻域结构集的构造

邻域结构集 $N_k(x)$ 的构造是 VNS 算法最核心的部分. VNS 算法在求解过程中通过动态地改变邻域结构,从而使搜索过程跳出局部最优向全局最优靠近. $N_k(x)$ 的构造包括如下问题:邻域结构集的形式,邻域结构集的个数以及邻域结构间的顺序.

对于函数优化问题 $\text{global} \min_{x \in S} f(x)$, $S \subset R^n$, 邻

域结构集 $N_k(x) (k = 1, 2, \dots, k_{\text{max}})$ 可定义为

$$N_k(x) = \{y \in S | \rho(x, y) \leq \rho_k\}, \quad (2)$$

或者

$$N_k(x) = \{y \in S | \rho_{k-1} \leq \rho(x, y) \leq \rho_k\}. \quad (3)$$

其中: k_{max} 为邻域结构集的个数,后文将对 k_{max} 的选取进行测试及分析; ρ_k 为邻域结构集的搜索半径, ρ_k 随着 k 的增加单调递增.

Drazic' 等^[13]将距离 $\rho(x, y)$ 定义为

$$\begin{cases} \rho(x, y) = \left(\sum_{i=1}^n |x_i - y_i|^p \right)^{1/p}, 1 \leq p < \infty; \\ \rho(x, y) = \max_{1 \leq i \leq n} |x_i - y_i|, p = \infty. \end{cases} \quad (4)$$

其中通常选取 $p = 1, 2, \infty$, p 的选取将决定邻域结构的几何结构.

邻域结构间的顺序可以通过邻域结构间次序的改变来实现, ρ_k 随着 k 的增加单调递增,即 $\rho_1 \leq \rho_2 \leq \dots \leq \rho_{k_{\text{max}}}$, 故邻域结构按由小到大排序,即 $|N_1(x)| \leq |N_2(x)| \leq \dots \leq |N_{k_{\text{max}}}(x)|$.

2.2 VNS 算法描述

VNS 算法基本流程如图 1 所示.

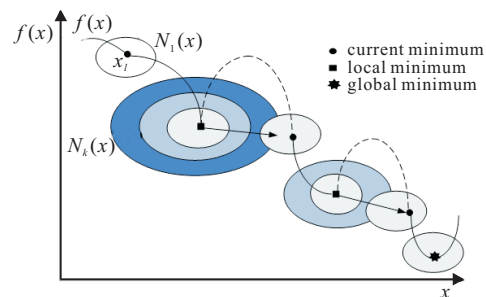


图 1 VNS 基本流程

下面给出 VNS 算法的详细步骤.

Step 1: 初始化. 给出初始解 x , 根据式 (3) 和 (4) 构造邻域结构集 $N_k(x) (k = 1, 2, \dots, k_{\text{max}})$ 和停止准则(遍历所有邻域结构).

Step 2: 令 $k = 1$, 直到 $k = k_{\text{max}}$.

Step 2.1: 随机搜索. 在 x 的第 k 个邻域中随机搜索产生 $x_l (x_l \in N_k(x))$.

Step 2.2: 局部搜索. 将随机搜索产生的解 x_l 作为初始解, 通过局部搜索获得局部最优解 x'_l .

Step 2.3: 更新. 若局部最优解 x'_l 优于当前最优解 x_l , 则令 $x_l = x'_l$, 并继续在当前邻域结构内搜索, 否则, 令 $k = k + 1$.

Step 3: 重构. 变换邻域结构集 $N_k(x)$ 的几何结构, 以得到更好的全局寻优效果.

3 VNS-PSO-CC 算法

为克服基本 PSO 算法在运算后期由于种群多样性的快速丧失而造成的算法“早熟收敛”问题, 对 t 代

种群中的粒子加入收敛因子. 收敛因子包括如下两个方面: 1) 收敛距离判断. 对第 t 代粒子群执行 c 均值聚类算法, 产生 c 个聚簇, 并计算各聚簇内每个粒子与聚类中心粒子的距离. 2) 聚合度判断. 对第 t 代粒子群进行聚合度判断.

通过收敛因子判断后, 对已陷入“早熟收敛”的粒子进行变邻域搜索, 从而得到全局最优.

3.1 收敛因子

定义 1 收敛中心. 设种群中某个粒子的位置为 $X_i = (x_{i1}, x_{i2}, \dots, x_{id}) (i = 1, 2, \dots, m)$, 聚簇 $c_j (j = 1, 2, \dots)$ 的收敛中心为

$$X_{c_j} = (x_{c_j,1}, x_{c_j,2}, \dots, x_{c_j,d}) = \left(\sum_{i \in c_j} \frac{x_{i,1}}{m}, \sum_{i \in c_j} \frac{x_{i,2}}{m}, \dots, \sum_{i \in c_j} \frac{x_{i,d}}{m} \right).$$

聚簇 c_j 内第 i 个粒子与收敛中心 X_{c_j} 的距离可表示为

$$d_{i,c_j} = \sqrt{(x_{i1} - x_{c_j,1})^2 + \dots + (x_{id} - x_{c_j,d})^2}. \quad (5)$$

聚簇 c_j 内每个粒子与收敛中心的平均距离为

$$d_{c_j,ave} = \frac{\sum_{i \in c_j} \sqrt{(x_{i1} - x_{c_j,1})^2 + \dots + (x_{id} - x_{c_j,d})^2}}{n_j}, \quad (6)$$

其中 n_j 为聚簇 c_j 内粒子的个数.

定义 2 聚合度. 设 t 代中所有粒子的历史最优平均值 $avgpi_k = \left(\sum_{i=1}^N F(plbest_i)_k \right) / N$, t 代中所有

粒子的平均值 $avg_k = \left(\sum_{i=1}^N F(x_i)_k \right) / N$, 则第 k 代粒子群的聚合度为

$$s = \min(avgpi_k, avg_k) / \max(avgpi_k, avg_k). \quad (7)$$

其中: F 为目标函数, N 为种群规模. 显然, 聚合度 $s \in (0, 1)$ 是反映粒子多样性的测度. s 越小, 说明种群活性越强; s 越大, 说明粒子的聚集程度越高; 当 $s = 1$ 时, 所有粒子具备同一性.

基于收敛距离和聚合度对每个粒子进行判断, 若粒子与其所在聚簇收敛中心的距离 d_{i,c_j} 小于平均距离 $d_{c_j,ave}$ 且聚合度 $s \rightarrow 1$, 则认为该粒子已经陷入“早熟收敛”, 并对该粒子进行变邻域搜索. 将需要进行变邻域搜索的粒子记作 x_l , 作为变邻域搜索的初始解.

3.2 VNS-PSO-CC 算法描述

VNS-PSO-CC 算法在基本 PSO 算法的框架内增加了收敛因子和变邻域搜索模块, VNS-PSO-CC 算法的流程如图 2 所示.

下面给出 VNS-PSO-CC 算法的详细步骤.

输入: 对粒子群进行随机初始化, 设置种群规模 $N = 200$, 粒子维数 $d = 30$, 最大迭代次数 $T = 500$, 聚

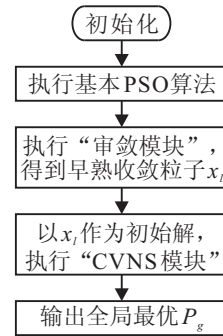


图 2 VNS-PSO-CC 算法流程

簇个数 $c = 4$, 邻域结构集个数 k_{max} , 迭代计数器 $b = 0$, 惯性权重 $\omega_{ini} = 0.9$, $\omega_{end} = 0.4$, 学习因子 $c_1 = c_2 = 2$;

输出: 全局最优 P_g .

Step 1: 初始化. 随机初始化粒子的位置和速度.

Step 2: 计算每个粒子的适应度值.

Step 3: 更新每个粒子的个体最优 P_i 和整个群体的全局最优 P_g .

Step 4: 聚类操作. 对第 t 代种群中的粒子执行 c 均值聚类算法, 产生 c 个聚簇, 计算每个聚簇的收敛中心 X_{c_j} .

Step 5: 根据式 (5) 和 (6) 分别计算每个聚簇内的粒子与其收敛中心的距离 d_{i,c_j} 和聚簇内粒子与收敛中心的平均距离 $d_{c_j,ave}$.

Step 6: 若 $d_{i,c_j} < d_{c_j,ave}$, 则根据式 (7) 计算第 t 代每个粒子的聚合度 s .

Step 7: 若粒子 x_l 的聚合度 $s \rightarrow 1$, 则将 x_l 作为变邻域搜索的初始解, 并对每个粒子 x_l 执行 VNS 模块.

Step 8: 判断是否满足终止条件. 如果满足, 则转 Step 9, 否则, 转 Step 2, 继续迭代.

Step 9: 输出全局最优 P_g , 算法运行结束.

4 实验及结果分析

4.1 参数设置

本文选取 4 个高维空间中的多峰函数作为测试函数^[8]. 其中: Rosenbrock 是很难极小化的典型病态二次函数, 常用该函数来评价优化算法的执行性能; Rastrigrin 为复杂多峰函数, 通常很难找到全局最优, 在求解中容易陷入接近 $(1, 1, \dots, 1)_d$ 的局部最小点, 最优值为 0, 如图 3 所示; Griewank 和 Ackley 为旋转、不可分离的、可变维数的多峰函数, 最优值为 0.

设置种群规模 $N = 200$, 维数 $d = 30$, 最大迭代次数 $T_{max} = 500$, 聚簇个数 $c = 4$, 邻域结构集个数为 k_{max} , 迭代计数器 $b = 0$, 惯性权重 $\omega_{ini} = 0.9$, $\omega_{end} = 0.4$, 学习因子 $c_1 = c_2 = 2$, 设定 3 个测试函数的可接受误差均为 10. 对每个测试函数独立运行 300 次. 定义判定标准: 1) 成功率 (SuccR). 规定最终解达到可接受误差为成功, 成功率为有效求解次数在总运行次

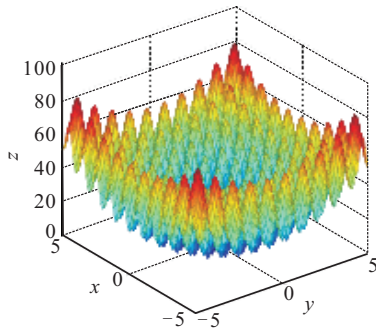


图3 Rastrigrin测试函数

数中的比例. 2) 平均最优值 (MeanBst). 300次运行完成后所得到的算法平均最优值, 可衡量粒子寻优的平均质量. 3) 最终适应值 (FinalBst). 表示最终收敛时函数的最优值. 4) 标准差. 表示每次迭代得到的适应值与平均最优值的差值, 可衡量粒子寻优的波动性.

4.2 测试及结果分析

1) 邻域结构集个数 k_{max} 的确定及相关结果分析. 参数设置如上所述, 分别测试 $k_{max} = \{1, 5, 10\}$ 时对 VNS-PSO-CC 算法性能的影响, 如表1所示.

表1 k_{max} 对算法性能的影响

Function	$k_{max} = 1$			$k_{max} = 5$			$k_{max} = 10$		
	SuccR	MeanBst	FinalBst	SuccR	MeanBst	FinalBst	SuccR	MeanBst	FinalBst
Rosenbrock	0.3975	6.8153	5.1134	0.7941	3.8727	3.3841	0.9937	2.5418	2.1985
Rastrigrin	0.5016	19.17	13.59	0.8391	8.5714	4.1813	1	7.3498	0
Griewank	0.4237	7.4891e-2	5.7553e-2	0.8486	6.1387e-3	4.8591e-3	1	2.8693e-3	1.3429e-3
Ackley	0.3391	6.1596e-2	3.8549e-2	0.8135	9.5841e-4	7.3225e-4	0.9641	3.3497e-5	1.1863e-5

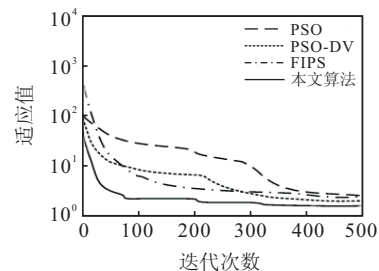
从表1可以看出, k_{max} 的取值对 VNS-PSO-CC 算法性能的影响很大, k_{max} 越大, 算法性能越好. 对于 VNS 模块, 由算法的全局性可知, 当停止准则为 $k = k_{max}$ 时, 算法全局收敛性仅与邻域结构集 N_k 的构造有关; 当 $k \rightarrow +\infty$ 且邻域结构集 N_k 内可获得局部最优解时, 可找到全局最优解. 然而, 实际上算法的搜索效率会随着 k_{max} 的增大而降低, 故本文在 VNS-PSO-CC 算法性能测试中选取 $k_{max} = 10$.

2) VNS-PSO-CC 算法性能测试. 将 VNS-PSO-CC 算法的测试结果与基本 PSO 算法、文献[3]提出的 FIPS 算法、文献[6]提出的 PSO-DV 算法进行比较, 4种函数的测试指标对比如表2所示, 图4给出了4种算法的收敛特性曲线.

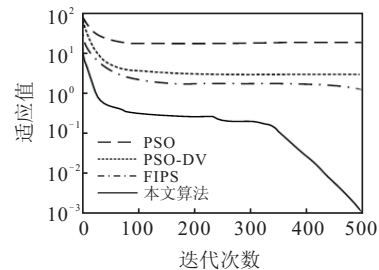
表2 不同算法测试指标对比

Function	Algorithm	SuccR	MeanBst	FinalBst
Rosenbrock	PSO	0.9270	3.9960	3.9866
	PSO-DV	0.8930	3.2965	3.1743
	FIPS	0.9481	3.7625	3.0596
	本文算法	0.9937	2.5418	2.1985
Rastrigrin	PSO	0.7990	24.5284	13.9294
	PSO-DV	0.9915	8.9213	6.9647
	FIPS	0.9841	8.1875	3.5208
	本文算法	1	7.3498	0
Griewank	PSO	0.9885	2.0215e-1	2.0203e-1
	PSO-DV	0.9735	8.2603e-2	8.1291e-2
	FIPS	0.9934	7.4597e-2	5.3076e-2
	本文算法	1	2.8693e-3	1.3429e-3
Ackley	PSO	0.8761	4.2783e-1	2.1375e-1
	PSO-DV	0.8933	6.7159e-3	3.6698e-3
	FIPS	0.9158	1.5311e-4	1.0973e-4
	本文算法	0.9641	3.3497e-5	1.1863e-5

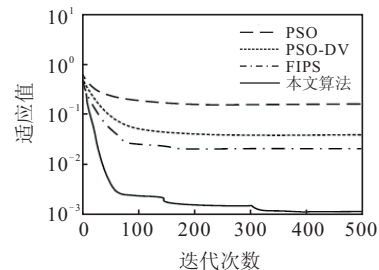
实验结果表明, 4种算法对 Griewank 函数都取得了较好的效果, 其主要原因是当维空间超过15维后, Griewank 函数特性趋向于单峰, Rosenbrock 函数的全局最优与可达到的局部最优之间有一道狭窄的山谷,



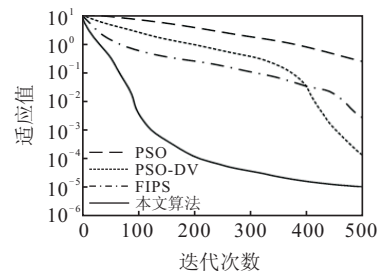
(a) Rosenbrock



(b) Rastrigrin



(c) Griewank



(d) Ackley

图4 测试函数收敛特性曲线

Rastrigrin 函数和 Ackley 函数具有大量的局部最优优点。由表 2 和图 4 可以看出: 本文算法成功地跳出了局部最优优点, 具有较好的全局收敛特性; 算法成功率明显提高; 平均最优值和最终适应值均小于其他 3 种算法, 充分表明本文算法在性能上优于其他算法。

VNS-PSO-CC 算法对 Rastrigrin 函数的测试曲线如图 5 所示。由 Rastrigrin 函数标准差的波动曲线及收敛曲线可以看出, 本文算法通过增加收敛因子和变邻域搜索模块, 在求解过程中动态地改变邻域结构, 使得粒子群摆脱早熟收敛和局部最优效果明显, 搜索过程中粒子不断跳出局部最优向全局最优靠近。

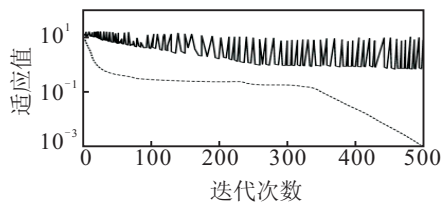


图 5 Rastrigrin 函数测试曲线

5 结 论

本文为解决高维空间中复杂多峰函数的优化问题, 结合 PSO 算法快速搜索以及 VNS 算法全局搜索的能力, 提出了带收敛因子的变邻域粒子群算法, 有效地避免了常规 PSO 算法在复杂多峰函数的求解过程中容易陷入早熟收敛的问题, 提高了全局搜索能力。由算法性能测试及对比分析可以看出, 与常规 PSO 算法相比, PSO-VNS-CC 算法在整体性能上有了很大改进, 寻优能力、收敛速度以及求解精度都有了显著提高, 适合于处理工程应用中的复杂函数优化问题。

参考文献(References)

- [1] Kennedy J, Eberhart R. Particle swarm optimization[C]. Proc of IEEE Int Conf on Neural Networks. Piscataway, 1995: 1942-1948.
- [2] Eberhart R, Kennedy J. A new optimizer using particle swarm theory[C]. Proc of Int' Symposium on Micro Machine and Human Science. Piscataway: IEEE Service Center: 1995: 39-43.
- [3] Mendes R, Kennedy J, Neves J. The fully informed particle swarm: Simpler, maybe better[J]. IEEE Trans on Evolutionary Computation, 2004(8): 204-210.
- [4] 张项学, 关治洪, 刘新芝. 一种动态改变惯性权重的自适应粒子群算法[J]. 控制与决策, 2008, 23(11): 1254-1257. (Zhang D X, Guan Z H, Liu X Z. Adaptive particle swarm optimization algorithm with dynamically changing inertia weight[J]. Control and Decision, 2008, 23(11): 1254-1257.)
- [5] 卢峰, 高立群. 基于改进粒子群算法的优化策略[J]. 东北大学学报: 自然科学版, 2011, 32(9): 1221-1224. (Lu F, Gao L Q. Novel optimization mechanism based on improved particle swarm optimization[J]. J of Northeastern University: Natural Science, 2011, 32(9): 1221-1224.)
- [6] 姚灿中, 杨建梅. 基于变惯性权重及动态邻域的改进 PSO 算法[J]. 计算机工程, 2011, 37(21): 20-22. (Yao C Z, Yang J M. Improved PSO algorithm based on variety inertia weight and dynamic neighborhood[J]. Computer Engineering, 2011, 37(21): 20-22.)
- [7] Liu B, Wang L, Jin Y H, et al. Improved particle swarm optimization combined with chaos[J]. Chaos, Solutions and Fractals, 2005, 25(5):1261-1271.
- [8] 刘衍民, 赵庆祯, 隋常玲. 一种基于动态邻居和变异因子的粒子群算法[J]. 控制与决策, 2010, 25(7): 968-974. (Liu Y M, Zhao Q Z, Sui C L. Particle swarm optimizer based on dynamic neighborhood topology and mutation operator[J]. Control and Decision, 2010, 25(7): 968-974.)
- [9] 刘衍民, 隋常玲, 赵庆祯. 基于 K-均值聚类的动态多种群粒子群算法及其应用[J]. 控制与决策, 2011, 26(7): 1019-1025. (Liu Y M, Sui C L, Zhao Q Z. Dynamic multi-swarm particle swarm optimizer based on K-means clustering and its application[J]. Control and Decision, 2011, 26(7): 1019-1025.)
- [10] Yannis Marinakis, Georgia-Roumbini Iordanidou, Magdalene Marinaki. Particle swarm optimization for the vehicle routing problem with stochastic demands[J]. Applied Soft Computing, 2013(13): 1693-1704.
- [11] Mladenovic' N, Hansen P. Variable neighborhood search[J]. Computers and Operations Research, 1997, 24(11): 1097-1100.
- [12] Hansen P, Mladenovic' N. Variable neighborhood search: Methods and applications[J]. Annals of Operations Research, 2010(175): 367-407.
- [13] Milan Drazic', Carlile Lavor, Nelson Maculan. A continuous variable neighborhood search heuristic for finding the three-dimensional structure of a molecule[J]. European J of Operational Research, 2008(185): 1265-1273.
- [14] Nima Hamta S M T, Fatemi Ghomi F Jolai. A hybrid PSO algorithm for a multi-objective assembly line balancing problem with flexible operation times, sequence-dependent setup times and learning effect[J]. Int J of Production Economics, 2013(141): 99-111.
- [15] Bassem Jarboui, Houda Derbel. Variable neighborhood search for location routing[J]. Computers and Operations Research, 2013(40): 47-57.
- [16] Yin-Yann Chen, Chen-Yang Cheng, Li-Chih Wang. A hybrid approach based on the variable neighborhood search and particle swarm optimization for parallel machine scheduling problems — A case study for solar cell industry[J]. Int J of Production Economics, 2013(141): 66-78. (责任编辑: 曹洪武)