

基于模式搜索法的云模型粒子群算法

吴建辉^{1†}, 王博华¹, 张小刚², 陈 华¹

(1. 湖南大学 信息科学与工程学院, 长沙 410082; 2. 湖南大学 电气与信息工程学院, 长沙 410082)

摘 要: 针对粒子群算法在多峰寻优中易陷入局部最优、精度低、难以搜寻到较多的极值点等问题, 提出一种基于模式搜索法的云模型粒子群算法. 首先通过云模型粒子群算法在可行域内进行全局搜索, 然后使用模式搜索法对搜索到的较优解进行局部寻优以提高解的精度. 仿真测试表明, 在保证收敛速度的同时, 所提算法的收敛精度和搜索到的极值点数目均得到显著提高.

关键词: 粒子群算法; 模式搜索法; 云模型; 多峰函数

中图分类号: TP18 文献标志码: A

Cloud model particle swarm optimization algorithm based on pattern search method

WU Jian-hui^{1†}, WANG Bo-hua¹, ZHANG Xiao-gang², CHEN Hua¹

(1. College of Information Science and Engineering, Hu'nan University, Changsha 410082, China; 2. College of Electrical and Information Engineering, Hu'nan University, Changsha 410082, China)

Abstract: In order to overcome the shortcomings of particle swarm optimization(PSO) in solving multimodal function optimization problems which includes easily falling into local minimum, low accuracy and difficultly searching extreme points as many as possible, a novel cloud model particle swarm optimization algorithm based on the pattern search method(PCPSO) is proposed. The cloud model particle swarm optimization(CPSO) algorithm is used to do global searching in the feasible zone, and then the pattern search method(PSM) is used to improve the accuracy of the sub-optimal solution. The simulation tests demonstrate that the proposed method can ensure the convergence speed, meanwhile the convergence accuracy and the number of extreme points are strikingly improved.

Keywords: particle swarm optimization; pattern search method; cloud model; multimodal function

0 引 言

在数学、建筑、机械等领域的工程实践中, 经常会遇到存在多个全局解或局部解的问题, 这类问题统称为多峰函数优化问题. 如何构造出一种有效解决该问题的优化算法, 使之能够求出全部全局最优解和尽可能多的局部最优解, 一直是进化计算领域的热门研究方向. 许多专家学者对此进行了广泛和深入的研究, 如带动态小生境半径策略的差分进化算法^[1]、带队列选择的混合差分进化算法^[2]、小生境混沌优化算法^[3]、带代理模型的大爆炸算法^[4]等. 这些算法在一定程度上取得了良好的效果, 但也存在一定的缺陷, 如文献[1-2]的方法存在应用场合受限, 文献[3-4]的方法在低维度多峰寻优中稳定性欠佳等问题.

粒子群优化算法(PSO)^[5]是模拟鸟群觅食行为

提出的一种全局优化算法. 作为一种重要的智能进化计算技术, 它具有易实现、前期收敛性能好、鲁棒性强等特点, 已被广泛应用于科学与工程实践中^[4,6-8]. 然而, PSO也存在易陷入局部最优、进化后期收敛速度慢、精度较差等缺陷^[9].

模式搜索法(PSM)^[10]是一种求解多维最优化问题的局部搜索算法, 具有结构简单、求解精度高、不依赖函数性态、局部搜索能力强等特点, 然而该算法对起始点位置的选择有较高要求, 否则容易陷入局部最优, 影响寻优效率.

为提高多峰函数寻优的性能, 本文融合了具有较强全局搜索能力的云模型粒子群算法(CPSO)和具有较强局部搜索能力的模式搜索法, 提出一种基于模式搜索法的云模型粒子群算法(PCPSO). 仿真实验表

收稿日期: 2016-08-31; 修回日期: 2016-11-28.

基金项目: 国家自然科学基金项目(61673162, 61672216).

作者简介: 吴建辉(1970—), 男, 讲师, 博士, 从事自然计算的研究; 张小刚(1972—), 男, 教授, 博士, 从事智能控制与数据挖掘的研究.

[†]通讯作者. E-mail: jianhuiw@hnu.edu.cn

明,该算法具有较好的全局及局部搜索能力,在确保搜索速度的同时,能够搜寻到更多的局部峰值,提高了计算精度.

1 云模型粒子群算法

1.1 多峰粒子群算法

多峰粒子群算法表达式如下:

$$V_{id}(t+1) = \omega(t)V_{id} + c_1(P_{id} - X_{id}(t)), \quad (1)$$

$$X_{id}(t+1) = X_{id}(t) + V_{id}(t+1). \quad (2)$$

在多峰粒子群算法中,没有标准粒子群算法中的参数 c_2 (即 $c_2 = 0$),这样设置是为了让算法可以找出所有的局部解,而不仅仅是一个或多个全局解.在式(1)中,惯性权重 ω 采用线性递减策略,其优点是在搜索前期,较大的惯性权重有利于进行快速的全局搜索,便于跳出局部最优;在搜索后期,较小的惯性权重有助于进行精确的局部搜索,提升寻优精度.为使各粒子尽快完成自我认知,提高收敛速度,将参数 r_1 设置为1.

1.2 基于云模型的变异策略

云模型^[11-12]是对自然语言描述的某个定性概念及其定量数值之间相互转换的数学模型,其中正态云模型具有模糊性及稳定趋向性等特点^[12],可以平衡粒子群算法在不同时期对全局和局部搜索能力的需求差异.因此,在多峰粒子群算法中引入基于云模型的云变异策略,提出一种云模型粒子群算法.

定义1(变异条件) 变异条件的数学表达式为

$$t > kt_{\max} \ \& \ t < (1 - k)t_{\max} \ \& \ \text{mod}(t, i) == 0. \quad (3)$$

其中: t 和 t_{\max} 分别为CPSO当前代数和最大迭代代数, mod 为求余运算符.事先给定比例系数 k 及间隔系数 i ,当进化代数达到总代数的一定比例时,即认为各粒子经过充分迭代已逐渐逼近自身所能找到的最好位置,可进行变异进化操作.设置间隔系数有助于保持粒子运动的连贯性,使粒子在进行变异操作后有充足的时间在新的位置上展开搜索,而不是“跳跃式”的随机寻优.

定义2(变异操作) 当满足条件(3)时,使用正向云发生器^[12]对粒子的位置进行变异操作,具体方法见文献[12].

2 模式搜索法

模式搜索法^[13]是一种直接搜索方法,其核心思想是通过计算和比较函数值,找出函数最优的下降方向,从而解决目标优化问题.PSM主要由探测移动和模式移动两部分构成,详细步骤见文献[13].

由于不要求解目标函数的导数,在面对不可或难以求导的问题时,PSM比求导算法具有更强的适用性,但这也导致了PSM难以快速确定目标函数

的下降方向,使其搜索效率受初始点位置的影响较大.对于不同的初始点,PSM寻优精度和寻优速度都会有较大幅度的波动,所以确保较佳的初始位置是保证其高效寻优的关键之一.

3 基于模式搜索法的云模型粒子群算法

基于模式搜索法的云模型粒子群算法通过设置混合算法的最大重复搜索代数 M ,避免出现在一次迭代中无法搜寻到全部极值点的情形.在每一次重复迭代中,首先使用CPSO在可行域内作全局搜索,然后使用PSM对CPSO搜索到的较优解进行局部搜索.PCPSO之所以采用这种串联形式的搭配,是因为PSO在运行前期收敛速度快,全局搜索能力强,通过引入基于云模型的云变异策略提高了PSO的性能;PSM计算精度高、局部搜索能力强,但对初始点的选择有较高要求,采用CPSO搜索到的较优解作为起始点,有助于提高PSM的寻优性能,进一步提高解的精度.

PCPSO的算法步骤描述如下.

Step 1: 随机初始化CPSO中各粒子的速度和位置,并对种群数 N 、惯性权重 ω 、最大迭代代数 t_{\max} 、PCPSO的最大重复搜索代数 M 及PSM的搜索精度 ϵ 等参数设置初值.

Step 2: 评估每个粒子的适应度,得到各粒子自身找到的最佳位置 P_i 以及全局最优粒子的位置 P_g .

Step 3: 若满足算法中止条件(即当前搜索代数 $m > M$),则停止搜索,输出所有搜寻到的极值点,否则转入Step 4.

Step 4: CPSO进行全局搜索:

If 当前迭代代数 $t \leq t_{\max}$

If 满足变异条件

对粒子的位置进行变异操作;

Else

按式(1)和(2)对粒子的速度、位置进行迭代

更替;

End If

更新 P_i 及 P_g ;

$t = t + 1$;

End If

Step 5: PSM进行局部搜索,对CPSO搜寻到的极值点进行局部寻优.

Step 6: 将新搜索到的极值点与极值点库进行对比更新(即极值点的同峰判断算子,详细步骤见文献[14]).

Step 7: $m = m + 1$,重新初始化各粒子的位置及速度.

Step 8: 将极值点库中全局极值点的位置赋给新

的粒子,并转入Step 2.

4 仿真实验及性能分析

为了验证PCPSO的有效性,评价其在多峰寻优中的搜索能力、收敛速度及求解精度等性能,引入如表1所示的多个结构复杂的多维度多峰函数(Function)进行测试.表1中:Range为自变量的取值

范围,Dim为函数的维度,Globe和All分别为全局极值点和全部极值点的个数; f_7 和 f_8 为旋转函数,分别由函数 f_4 和 f_6 通过线性变换矩阵 M 转换而来, M 的生成方法见文献[15].仿真环境为Intel(R) I5-5 200 U处理器,4G内存,Windows8.1操作系统,Matlab R2013b仿真软件.

表1 测试函数

Function	Range	Dim	Globe/All
$f_1 = 900 - (x_1 - 5)^2 + 10 \cos[(2\pi(x_1 - 5))] + (x_2 - 5)^2 + 10 \cos[2\pi(x_2 - 5)]$	$1.8 \leq x_1, x_2 \leq 8.2$	2	2/49
$f_2 = 20 + x_1^2 + x_2^2 - 10(\cos 2\pi x_1 + \cos 2\pi x_2)$	$-2.5 \leq x_1, x_2 \leq 2.5$	2	1/25
$f_3 = \exp\left(-2 \log 2 \left(\frac{x - 0.08}{0.854}\right)^2\right) \sin^6(5\pi(x^{3/4} - 0.05))$	$0 \leq x \leq 1$	1	1/5
$f_4 = 200 - (x_1^2 + x_2 - 11)^2 - (x_1 + x_2^2 - 7)^2$	$-6 \leq x_1, x_2 \leq 6$	2	4/4
$f_5 = -\prod_{i=1}^d \sum_{j=1}^5 j \cos[(j+1)x_i + j]$	$-10 \leq x_i \leq 10$	3	$d \times 3^d$ /many
$f_6 = 0.5 - \frac{(\sin \sqrt{x_1^2 + x_2^2})^2 - 0.5}{1 + 0.001(x_1^2 + x_2^2)}$	$-10 \leq x_1, x_2 \leq 10$	2	1/many
$f_7 = 200 - (y_1^2 + y_2 - 11)^2 - (y_1 + y_2^2 - 7)^2, y = M \times x$	$-6 \leq x_1, x_2 \leq 6$	2	4/4
$f_8 = 0.5 - \frac{(\sin \sqrt{y_1^2 + y_2^2})^2 - 0.5}{1 + 0.001(y_1^2 + y_2^2)}, y = M \times x$	$-10 \leq x_1, x_2 \leq 10$	2	1/many

算法参数设置如下:因为表1中每个函数所需搜寻的极值点数不同,所以对PCPSO的参数设置也有所差异,粒子数 N 为30~100,最大重复搜索代数 M 为2~10, $c_1 = 1.49445, \omega_{start} = 0.9, \omega_{end} = 0.4, t_{max} = 450, k = 0.05, i = 8, \epsilon = 10^{-7}$.

4.1 基本多峰函数仿真结果及分析

表2为针对函数 f_1 和 f_2 ,PCPSO与其他3种算法测试50次的实验结果.其中:Niching Genetic Algorithm(NGA)、autounning NGA(ANGA)、Big Bang-Big Crunch(BB-BC)的实验数据均出自文献[4];SP(number of searched peaks)为可搜索到的峰值数量;FC(number of function calls)为评价函数调用次数;CR(convergence rate)为收敛率,定义为搜寻到的平均峰值数与峰值总数的比值.

表2 PCPSO与BB-BC等3种算法的比较(50次测试)

Function	Alg	SP	CR/%	FC
f_1	NGA	49	92.7	9 848
	ANGA	49	98.3	6 792
	BB-BC	49	99.2	487
	PCPSO($N = 60, M = 9$)	49	99.9	27 500
f_2	NGA	25	92.5	8 112
	ANGA	25	98.4	5 011
	BB-BC	25	99.4	329
	PCPSO($N = 30, M = 4$)	25	100	14 000

由表2可知,PCPSO每次都能搜索到所有的极值点,收敛率均能达到或接近1,稳定性与精确性均高于其他3种算法,虽然在评价函数调用方面高于BB-BC

等算法,但因PSO迭代更替快,所以PCPSO在速度方面并无明显劣势.

表3为针对函数 $f_3 \sim f_5$,PCPSO与其他3种算法运行50次的实验结果.其中:crowding differential evolution(CDE),ring topology PSO(r3PSO),composite DE with queueing selection(CoDE-QS)的实验数据均来自文献[2];success rate(SR)为成功率,定义为成功找到所有全局极值点的运行次数与总运行次数的比值;peak ratio(PR)为峰比率,定义为找到的平均全局极值点数与所有全局极值点个数的比值;PCPSO的参数 N 和 M 分别设置为30、30、100和2、2、10.

表3 PCPSO与CoDE-QS等3种算法的比较(50次测试)

Function	r3PSO	CDE	CoDE-QS	PCPSO
	PR/SR	PR/SR	PR/SR	PR/SR
f_3	1.000/1.000	1.000/1.000	1.000/1.000	1.000/1.000
f_4	0.995/0.980	0.965/0.860	0.745/0.140	1.000/1.000
f_5	0.118/0.000	0.541/0.000	1.000/1.000	0.975/0.260

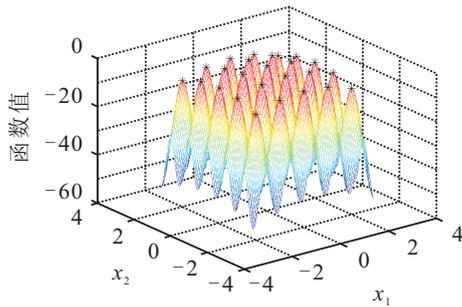
由表3可知:对于除 f_5 外,PCPSO的函数峰比率和成功率均优于或等于其他算法;对于 f_5 ,PCPSO平均能找到79个全局极值点,略低于理论值81个,因为找到所有全局极值点的次数较少,所以成功率较低.

4.2 性能比较及分析

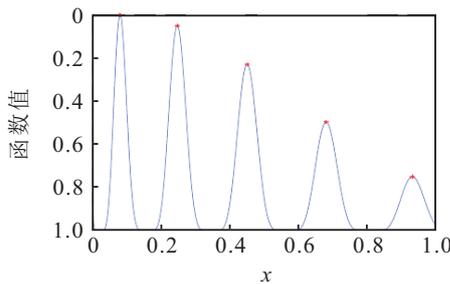
表4为PCPSO与PSO和CPSO算法性能的比较,其中PSO为依据式(1)和(2)改进而来的多峰PSO,Mean_g、Std_g、Mean_l、NSGO和NSLO分别为运行50次搜索到的全局解均值、全局解标准差、局部解均值、全局解个数和局部解个数.

表4 PSO、CPSO和PCPSO性能比较(50次测试)

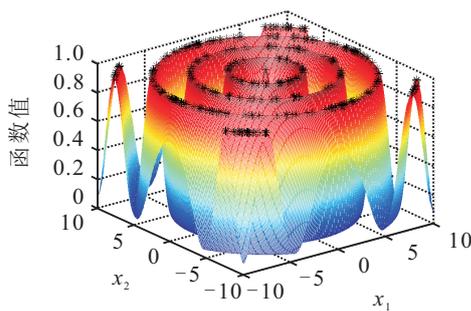
Function	Alg	Mean _g	Std _g	Mean ₁	NSGO/NSLO	Time/s
f_2 ($N = 30, M = 4$)	PSO	0.393 569 314 341	$4.021e^{-1}$	5.839 874	0.1/24	17.2
	CPSO	0.000 000 000 966	$2.333e^{-9}$	4.145 658	1/24	18.4
	PCPSO	0	0	4.145 658	1/24	19.5
f_3 ($N = 30, M = 2$)	PSO	0.707 360 552 258	$1.735e^{-1}$	0.268 322	0/4	2.8
	CPSO	0.806 208 993 234	$1.232e^{-1}$	0.379 208	0/4	3.6
	PCPSO	0.999 999 828 454	$4.252e^{-15}$	0.623 691	1/4	5.9
f_4 ($N = 30, M = 2$)	PSO	199.376 457 652 231	1.171	/	3.4/0	1.6
	CPSO	199.999 999 996 457	$9.832e^{-9}$	/	4/0	2.5
	PCPSO	200	0	/	4/0	3.4
f_6 ($N = 80, M = 9$)	PSO	0.996 724 061 348	$1.956e^{-3}$	0.964 924	0.2/352.7	34.4
	CPSO	0.999 348 774 376	$1.660e^{-3}$	0.962 787	0.9/509.5	58.6
	PCPSO	0.999 837 194 031	$8.764e^{-4}$	0.962 984	0.9/509.4	62.4



(a) f_2 解的分布



(b) f_3 解的分布



(c) f_6 解的分布

图1 f_2 、 f_3 与 f_6 的分布图

由表4可知: CPSO的求解精度、找到的峰值个数均高于改进后的多峰PSO,说明了云模型变异策略的有效性;与CPSO相比,虽然PCPSO花费时间有所增多,但其求解精度有了近一步的提高,验证了在优秀个体附近往往存在更优秀的个体,说明了PCPSO串联搭配的合理性,可见PCPSO算法性能优于CPSO.

图1为PCPSO针对函数 f_2 、 f_3 和 f_6 所搜寻到的解的空间分布图.从图1可以看出,PCPSO能够精确地搜寻到函数 f_2 、 f_3 的全部极值点和 f_6 的大量极值点,说明其具有良好的多峰寻优能力.

4.3 旋转多峰函数仿真结果及分析

旋转函数^[15]定义为 $f_{rot} = f(M \times x)$, M 为一个线性转换矩阵.当函数未发生旋转时,自变量 x 在各维度上的分量是相互独立的,经旋转变换后, x 各维度上的分量将受其他维度分量的影响,某一维度上的变化将起到牵一发而动全身的作用,所以使用旋转的多峰函数进行测试能更好地验证算法的寻优性能.

表5为针对旋转函数 f_7 和 f_8 ,PCPSO与PSO和CPSO的性能对比.由表4和表5的对比可知,3种算法面对旋转函数时,时间开销均有所增加,评价指标均有不同程度的下降,其中PSO性能下降严重,全局解均值和标准差较旋转前大幅下降,CPSO的全局解均值和标准差略有下降,PCPSO在计算精度和稳定性方面依然保持明显优势,其全局解均值和标准差均达到或接近理论值,优于其他2种算法.

表5 PSO、CPSO和PCPSO性能比较(50次测试)

Function	Alg	Mean _g	Std _g	Mean ₁	NSGO/NSLO	Time/s
f_7 ($N = 30, M = 2$)	PSO	196.291 542 630 768	8.984	/	1.7/0	2.0
	CPSO	199.999 999 971 081	$9.433e^{-8}$	/	4/0	2.9
	PCPSO	200	0	/	4/0	3.6
f_8 ($N = 80, M = 9$)	PSO	0.996 661 237 822	$2.050e^{-3}$	0.930 270	0.2/399.7	41.7
	CPSO	0.999 185 969 955	$1.820e^{-3}$	0.921 943	0.9/469.4	62.5
	PCPSO	0.999 348 764 221	$9.660e^{-4}$	0.925 711	0.9/469.0	68.6

图2为PCPSO针对旋转函数 f_8 搜寻到的解空间分布图,可见PCPSO可以找到 f_8 的一个全局极值点和大量的局部极值点,面对扰动表现出良好的鲁棒性。

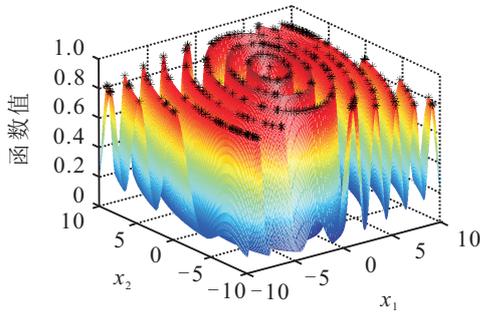


图2 f_8 解的分布图

5 结论

针对多峰函数优化问题,本文提出了一种基于模式搜索法的云模型粒子群算法.该算法不但克服了PSO寻优精度低、易陷入局部最优等缺陷,还有效地发挥了PSM局部搜索能力强和计算精度高的优势.通过引入基于云模型的云变异策略,使PSO保持了良好的种群多样性,提高了其搜索性能;通过使用CPSO搜索到的较优解作为初始点,有效地提高了PSM的寻优效率.因为该算法不依赖于函数的导数,所以是一种求解多峰问题的通用算法.

由实验结果可知,尽管PCPSO拥有较高的求解精度和较强的稳定性,但面对极值点较多的函数时,需要增加重复搜索的次数以防止峰值点漏判,从而导致时间开销增大.如何能在较少的重复搜索次数里搜寻到较多的极值点,从而获得更好的搜索效率,将是下一步研究和解决的问题.

参考文献(References)

- [1] Zhang G, Li D, Zhou X, et al. Differential evolution with dynamic niche radius strategy for multimodal optimization[C]. Proc of the 27th Chinese Control and Decision Conf. Qingdao: IEEE, 2015: 3059-3064.
- [2] Zhang Y H, Gong Y J, Chen W N, et al. Composite differential evolution with queueing selection for multimodal optimization[C]. Proc of IEEE Congress on Evolutionary Computation. Sendai: IEEE, 2015: 425-432.
- [3] Rim C, Piao S, Li G, et al. A niching chaos optimization algorithm for multimodal optimization[J]. Soft Computing, 2016, 20: 1-13.
- [4] Yoo C H, Lim D K, Jung H K. A novel multimodal optimization algorithm for the design of electromagnetic machines[J]. IEEE Trans on Magnetics, 2016, 52(3): 1-4.
- [5] Bonyadi M R, Michalewicz Z. Stability analysis of the particle swarm optimization without stagnation assumption[J]. IEEE Trans on Evolutionary Computation, 2016, 20(5): 814-819.
- [6] Guo L, Zhang H, Galea M, et al. Multiobjective optimization of a magnetically levitated planar motor with multilayer windings[J]. IEEE Trans on Industrial Electronics, 2016, 63(6): 3522-3532.
- [7] Yu J, Wu Z, Wang M, et al. CPG network optimization for a biomimetic robotic fish via PSO[J]. IEEE Trans on Neural Networks and Learning Systems, 2016, 27(9): 1962-1968.
- [8] Fong S, Wong R, Vasilakos A V. Accelerated PSO swarm search feature selection for data stream mining big data[J]. IEEE Trans on Services Computing, 2016, 9(1): 33-45.
- [9] Qin Q, Cheng S, Zhang Q, et al. Particle swarm optimization with interswarm interactive learning strategy[J]. IEEE Trans on Cybernetics, 2016, 46(10): 2238-2251.
- [10] Zhang X, Jiang H, Yu Z, et al. A generator tripping control method based on trajectory sensitivity analysis and pattern search[C]. Proc of the 34th Chinese Control Conf. Hangzhou: IEEE, 2015: 9043-9047.
- [11] 李国栋, 胡建平, 夏克文. 基于云PSO的RVM入侵检测[J]. 控制与决策, 2015, 30(4): 698-702. (Li G D, Hu J P, Xia K W. Intrusion detection using relevance vector machine based on cloud particle swarm optimization[J]. Control and Decision, 2015, 30(4): 698-702.)
- [12] 吴建辉, 章兢, 陈华, 等. 多峰函数优化的免疫云粒子群优化算法[J]. 仪器仪表学报, 2013, 34(8): 1756-1765. (Wu J H, Zhang J, Chen H, et al. Immune cloud particle swarm optimization algorithm for multimodal function optimization[J]. Chinese J of Scientific Instrument, 2013, 34(8): 1756-1765.)
- [13] 龚纯, 王正林. 精通MATLAB最优化计算[M]. 北京: 电子工业出版社, 2014: 130-134. (Gong C, Wang Z L. Proficient in Matlab optimization calculation[M]. Beijing: Publishing House of Electronics Industry, 2014: 130-134.)
- [14] 吴建辉, 章兢, 陈红安. 融合Powell搜索法的粒子群优化算法[J]. 控制与决策, 2012, 27(3): 343-348. (Wu J H, Zhang J, Chen H A. Particle swarm optimization algorithm combination with Powell search method[J]. Control and Decision, 2012, 27(3): 343-348.)
- [15] Suganthan P N, Hansen N, Liang J J, et al. Problem definitions and evaluation criteria for the CEC2005 special session on real-parameter optimization[R]. Singapore: Nanyang Technological University, 2005.