

doi:10.3969/j.issn.1673-4785.2010.04.009

均匀粒子群算法

刘宏达, 马忠丽

(哈尔滨工程大学 自动化学院, 黑龙江 哈尔滨 150001)

摘要:由于粒子群算法本质上的随机性,其搜索质量和速度也呈随机性.这使得普通的粒子群算法难以满足某些需要快速优化的工程需要.利用均匀设计方法产生 PSO 算法的初始种群(或关键代次种群),可以使种群中的粒子在搜索空间分布更均匀,更好地保持分散性.算法中给出了4种种群的生成方案,通过测试和对比分析表明:基于值域分割的均匀设计种群生成法能使算法的搜索效果最好;算法可以在不丧失搜索精度和效率的前提下,提高搜索效率和搜索精度的稳定性,有效减少粒子聚集和早熟的发生.

关键词:粒子群;均匀设计;实时优化;关键代次种群

中图分类号: 文献标识码:A 文章编号:1673-4785(2010)04-0336-06

A particle swarm optimization algorithm based on uniform design

LIU Hong-da, MA Zhong-li

(College of Automation, Harbin Engineering University, Harbin 150001, China)

Abstract: In a normal PSO algorithm, particle populations are usually produced randomly, leading to variations in search quality and speed. Such algorithms cannot be used to solve engineering problems that must be optimized quickly. In order to solve such problems, a PSO algorithm based on uniform design was used to generate initial PSO populations. This made the distribution of particles more uniform in the search space. Four methods for generation of particle swarms were studied. Test results showed that search precision was considerably improved if the generation of a particle swarm by the uniform design method was based on value range division. That method maintained search efficiency and precision, stabilized search efficiency and precision, and reduced the possibility of swarm aggregation and premature convergence.

Keywords: PSO; uniform design; real-time optimization; key population

目前优化算法的研究主要集中在搜索算法上,搜索算法分为确定性搜索法和随机性搜索法2种.在确定性搜索算法寻优过程中,一个搜索点到另一个搜索点的转移有确定的方法和规则,计算受初始点的影响较大,典型代表是禁忌搜索.随机性搜索算法在算法中加入随机性因素,并以一定概率接受比当前解更差的解,所以计算不受初始点限制且提高了全局寻优性能,典型代表是模拟退火、免疫算法、蚁群算法和粒子群算法等.

Kennedy 和 Eberhart (1995) 受鸟群和鱼群运动启发,并借用人类的决策过程中所采用的个体学习与文化传递的观念,将其用于优化问题,提出了粒子

群优化算法^[1]. PSO 算法的简洁性和快速性使它一经提出便引起世界上许多研究者的极大关注.

由于粒子群算法种群产生的随机性,算法的搜索质量和速度也呈随机性,一次测量不能反映粒子群算法的性能,所以一般都是通过多次测量以均值和标准差的形式给出.但是,在某些需要实时优化的工程中,工作条件不允许进行多次优化.对这类问题,能否在算法搜索质量不会降低的情况下,能显著提高算法解的稳定性和集中性,是研究的出发点.

搜索算法初始种群的确定问题其实是一个如何利用有限个体来全面科学地表征所求问题解空间特征的优化设计问题.算法初始种群的确定问题实质上是一个如何利用有限的个体来全面科学地表征问题解空间特征的优化设计问题^[2].显然,对于初始

种群既不能粗略地随机产生,也不能遍历所有的状况,尤其是解决多变量问题时,只有将解空间中最有代表性的个体作为初始种群,才能更好地反映解空间的内在特征,从而较好地表征解空间^[2,3].文献[3]利用数论中的佳点集的理论和方法,对GA算法中的交叉操作进行了重新设计,并应用于求解优化问题、SAT问题、TSP问题和背包问题.通过模拟比较表明新的算法提高了计算的速度和精度,而且避免了其他方法常有的早期收敛的现象.文献[4]在层次色谱研究中,利用均与设计选择实验,再利用遗传算法探寻函数曲线,试验收到良好的效果.文献[5]利用正交设计的方法产生初始化种群;用正交交叉算子代替传统的算术交叉算子;利用Agent间的竞争作用与每个Agent所具有的知识 and 自学能力进行启发式搜索.仿真试验和性能分析表明,算法不但具有很强的全局优化能力和较快的收敛速度,而且具有很强的鲁棒性.文献[6]将遗传算法的参数设定描述为一个多因素多水平优化设计问题,同时考虑到参数设定方法的可行性,提出应用解决多因素多水平优化设计问题的均匀设计方法设定遗传算法的操作参数,实例应用仿真结果验证了这种方法的可行性、有效性.通过比较,其中均匀设计法性价比最高,但也存在着当因素的个数和等级增多时,对应的均匀设计表很难得到,所产生的初始种群的多样性保持不好的不足.

该文将均匀试验设计和粒子群算法相结合,提出了均匀粒子群算法(PSO based on uniform design, UD-PSO).该算法不但充分考虑粒子的记忆特性,而且考虑粒子群中粒子分布的均匀性,使粒子能更好地适应周围环境,从而引导粒子群不断进化,最终达到全局优化的目的.为了当因素的个数和等级增多时均匀设计法的不足,引入了4种种群生成方案,测试效果表明,可以很好地解决了此问题.

1 算法提出背景

由于传统PSO的初始种群是随机选取的,初始种群的覆盖空间具有很大的不确定性,如果初始种群空间不包含全局最优解的信息,而算法算子又不能在有限的进化代数内将覆盖空间扩延到全局最优解所在的区域,那么过早收敛就不可避免.到目前为止,还无法证明在有限迭代次数内,PSO算法一定能达到全局最优解所在的区域.所以,确保初始种群的多样性与个体分布的相对合理性就成了改善PSO

全局收敛性首先要解决的问题^[2].

当然,初始种群中若最优解的信息较少也可以通过群体的进化来得到,从而达到最优解,但由于PSO算子本身的随机性,目前仅靠PSO算子是不能确保算法搜索到最优解区域的.保持多样性的确实可以对算法的全局收敛性能进行改善,但这一多样性必须是具有一定代表性并将群体引导到有利于全局收敛的多样性.

另外,无论是从PSO算法的基础理论还是从算子的运算方式上讲,粒子从自身和社会得来的经验对初始种群的依赖性都较强,而算法的搜索本身就是靠粒子的位置运动实现的,所以,初始种群的优劣会对算法的快速性产生较大的影响.在无法预知最优解所在区域的情况下,只有科学充分地代表解空间特征的初始种群才能使算法以较优的方式快速逼近最优解.可见,初始种群必须充分代表解空间的个体,在有限数量内最大限度地表征所有个体的信息,达到初始种群在收敛性与快速性的协调统一.但初始种群的合理性是较难判别的,尤其是多变量优化问题就更加困难.随机产生的初始种群无法确保种群的合理性,当然也就无法确定算法的高效性.实际上,确定初始种群的核心就是希望初始种群拥有合理的多样性并在尽可能少的代数内将寻优搜索引导到最优解区域.

如上所述,粒子群算法初始种群的确定问题其实是一个如何利用有限个体来全面科学地表征所求问题解空间特征的优化设计问题.显然,对于初始种群既不能粗略随机产生,也不能遍历空间中所有点的情况,只有将解空间中最有代表性的个体作为初始种群,才能更好地反映解空间的内在特征,从而较好地表征解空间.在试验范围内均匀散布的特点,使均匀设计成为解决这个问题的有益思路.

2 标准粒子群算法和均匀设计

2.1 标准粒子群算法

目前标准粒子群算法被公认的有2种,这里使用LDW-PSO.它是这样规定的:粒子群运动在一个 n 维搜索空间中,种群 (x_1, \dots, x_m) 由 m 个粒子组成. $x_i = (x_{i1}, x_{i2}, \dots, x_{in})$ 为第 i 个粒子的位置,每个粒子的速度为 $v_i = (v_{i1}, v_{i2}, \dots, v_{in})$.其个体极值为 p_i ,种群的最优值为 p_g ,粒子 x_i 按式(1)和(2)更新自己每一维的速度和位置:

$$v_{id}^{t+1} = wv_{id}^t + c_1r_1(p_{id}^t - x_{id}^t) + c_2r_2(p_{gd}^t - x_{id}^t), \quad (1)$$

$$x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1}. \tag{2}$$

式中: $d = 1, 2, \dots, n, i = 1, 2, \dots, m$. m 是种群规模, t 是当前进化代数, c_1, c_2 为加速因子. 惯性权重采用 Shi 建议的线性递减权重 (liner degression weight, LDW) 策略来控制惯性权重, 即

$$w^t = w_{ini} - (w_{ini} - w_{end})t/T_{max}. \tag{3}$$

式中: T_{max} 为最大进化代数; w_{ini} 为初始惯性权重; w_{end} 为进化至最大代数时的惯性权重, 并推荐取值为 $w_{ini} = 0.9, w_{end} = 0.4$.

2.2 均匀设计简介和均匀设计表的构建

2.2.1 均匀设计简介

均匀设计^[7-8]是由我国数学家方开泰和王元于 1981 年提出的. 它是基于数论中的一致分布理论, 将数论和多元统计相结合, 从均匀性角度出发, 基于试验点在整个试验范围内均匀散布的一种试验设计方法. 对于 n 个因素、 q 个水平的试验, 它从 q^n 个试验组合中选出 q 个在解空间分布均匀的组合进行试验, 与正交设计至少需要 q^2 次试验相比较, 试验次数大大减少, 在许许多多水平试验的问题中非常实用. 与正交设计相比, 均匀设计为试验者提供更多的选择, 可以用较少的试验获得期望的结果, 这使得均匀设计特别适合于多因素、多水平的试验和系统模型完全未知的情况. 它已成为统计试验设计的重要方法之一.

2.2.2 均匀设计表的构建

要进行均匀试验设计, 首先需要选择均匀设计表. 均匀设计表 (简称均匀表) 是均匀设计的基础. 每一个均匀设计表都有一个代号, 等水平的均匀设计表可用 $U_n(q^s)$, 其中 U 表示均匀设计, n 为均匀表的行数 (也就是需要做的试验数), q 为每个因素的水平数, s 为均匀表的列数. 当水平数 q 为偶数时, $U_n(q^s)$ 的均匀性较差, 所以方开泰等人建议用 $U_n^*(q^s)$ 代替. 通常, 同因素、同水平的均匀表 $U_n^*(q^s)$ 比 $U_n(q^s)$ 有更好的均匀性, 应优先选用. 如果各个因素的水平数目不同, 则选用均匀表 $U_n(q_1^{s_1} \times q_2^{s_2} \times \dots \times q_j^{s_j})$, 其中, $s_j (1 \leq j \leq j)$ 表示具有不同水平的因素数目, $q_j (1 \leq j \leq j)$ 表示与 s_j 相应的水平数目.

均匀设计表的构造方法有很多, 如好格子点法 (good loattice point, GLP)、拉丁方法 (latin hyper-cube, LH)、正交表扩充法 (expending orthogonal design, EOD)、优化搜索法以及 RBIBO 法等^[3-4]. 由于易于编程实现, 好格子法和方幂生成向量法被使用的最为普遍. 当然也可以使用已算好的均匀设计表.

对因素数为 s 和水平数为 n 的试验, 可产生 C_s^m

种均匀设计方案; 而方案的选择不同, 试验的效果也大不一样. 评价各设计方案好坏一般用均匀性度量指标来实现. 均匀度越小表示方案的均匀性越好, 通常利用偏差 (discrepancy) 最小为标准, 从均匀设计表中选出均匀性最好的 s 列, 这一过程也被称为使用表的选择. 均匀性度量常用的准则有: 偏差准则 D 、 L_2 -偏差、修正的 L_2 -偏差 MD_2 、中心化 L_2 -偏差 CD_2 、对称 L_2 -偏差 SD_2 、可卷 L_2 -偏差 WD_2 、散度准则、海明距等^[3]. 中心化 L_2 -偏差 CD_2 取消了原点的特殊地位, 对数据比较灵敏, 测试结果比较合理, 现在使用的许多均匀设计表都是采用它来设计的. 下面给出对于均匀设计 $U_n(q^s)$, 中心化 L_2 -偏差的计算公式.

$$CD_2(P) = \left[\left(\frac{13}{12} \right)^s - \frac{2^{s-1}}{n} \sum_{i=1}^n \prod_{j=1}^s \left(2 + \left| x_{ij} - \frac{1}{2} \right| - \left| x_{ij} - \frac{1}{2} \right|^2 \right) + \frac{1}{n^2} \sum_{i=1}^n \prod_{j=1}^s \left(1 + \frac{1}{2} \left| x_{kj} - \frac{1}{2} \right| + \frac{1}{2} \left| x_{kj} - \frac{1}{2} \right|^2 - \frac{1}{2} |x_{kj} - x_{ij}| \right) \right]^{\frac{1}{2}}. \tag{4}$$

3 算法的实现

将粒子群算法的问题域划分为适当多的水平, 在这些水平中均匀地取值, 就可以使生成的粒子均匀分布在问题域中. 设 u_{ij} 是均匀设计表 $U_n(n^s)$ 中的元素, 令 $a_{ij} = (2u_{ij} - 1)/2n, j = 1, 2, \dots, n$, 则集合 $p_m = \{a_k = (a_{k1}, a_{k2}, \dots, a_{kn}), k = 1, 2, \dots, m\}$ 是 $[0, 1]^m$ 中的均匀分布的 m 个点. 图 1 表示的是在 2 维空间中, 利用均匀设计的方法和随机方法产生的点集分布情况.

从图 1 可以看出, 均匀设计的方法产生的点集所构成的初始群体比随机的初始群体更能从统计意义上反映出目标函数的特性. 随机分布的初始群体在分布的分散性上, 不如均匀设计的方法产生的点集.

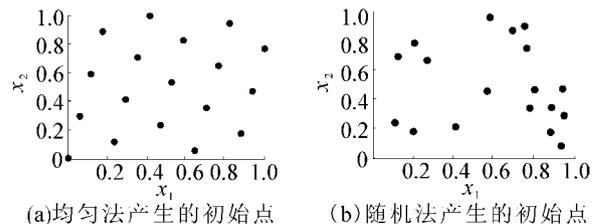


图 1 均匀法和随机法产生的初始点集

Fig. 1 Initial dots brought by uniform design and random method

3.1 关键代次种群生成的基本步骤

算法实现的关键在于:根据问题选择合适的种群粒子数 n , 并确定待优化问题维数 D , 按照表 1 的对应关系, 设计一个 $U_n(n^D)$ 均匀设计表.

表 1 均匀设计表和 PSO 算法种群的关系

Table 1 Relationship of uniform design table and parameters of PSO

	均匀设计	均匀设计表	PSO 算法
n	水平	行数	群中的粒子数
D	因素	列数	待优化变量维数

以构建初始种群为例, 给出关键代次种群生成的基本步骤:

- 1) 根据给定的 n , 生成等水平均匀表 $U_n(n^m)$, 得到对应的均匀设计矩阵 $U_{n \times m}$;
- 2) 根据待优化问题的维数 D , 求得 CD^2 偏差最小的矩阵 $U_{n \times D}$, 其中 $U_{n \times D} = (v_{ij}), i \in I_n, j \in I_D$.
- 3) 按照 $u_{ij} = (2v_{ij} - 1)/2n$, 将 v_{ij} 转化为 $[0, 1]$ 中的点 u_{ij} ;
- 4) 转换为 $X = \{x_{ij} | l_j \leq x_{ij} \leq h_j\}$ 中的初始点;
- 5) 得出初始种群 $x_{ij} = l_j + \frac{u_{ij}}{n}(h_j - l_j), i \in I_n, j \in I_D$.

在具体应用时, 可以利用提前算好的 $U_{n \times D}$, 例如利用文献[13]可以直接获得推荐的 CD^2 偏差最小的矩阵 $U_{n \times D}$. 这样直接可以进入基本步骤的 4), 所以算法复杂度并未增加多少.

3.2 不同种群规模的种群生成方法

文献[13]只提供了中心化 L_2 -偏差最小的均匀使用表的最大因素数 29. 也就是说, 如直接利用文献[13]进行均匀粒子群设计, 粒子数最大只能选到 29.

3.2.1 小种群规模的算法实现

一般情况下, 粒子群算法的粒子数取 20 ~ 30^[9], 这里将其称为小种群规模. 这时可以通过均匀设计表直接产生初始种群(或关键代次种群). 由于一般问题只需取粒子数为 30 即可, 所以只要利用等水平均匀表 $U_{30}(30^D)$ 设计即可. 但是目前直接可以使用的均匀表最多只能进行 30 个因素的试验, 而具有中心化 L_2 -偏差最小的均匀使用表的最大因素数 29, 所以尝试用 2 种方案来解决. 1) 随机生成一个粒子补充进入. 2) 利用均匀等价准则^[10](均匀设计方案阵的任 2 列交换所得新方案与原方案在均匀性相同意义下是等价的), 将均匀阵 $U_{n \times D}$ 任意一些列互调位置, 然后随机抽取一列, 作为第 30 列.

3.2.2 大种群规模的算法实现

当被优化问题需要的种群规模较大时, 均匀设计的表中的水平也要增大, 此时为了选择均匀使用表的计算量将很大. 例如种群规模为 60, 需设计 $U_{60}(60^5)$, 即使采用方幂生成向量法也需要比较几十组可能列来选出生成向量. 为了便于应用, 降低计算复杂性, 希望应用目前现有的生成向量表. 但由于目前常用的修正好格子点法生成向量表最多只有 31 水平, 即只能得到 31 个个体, 这显然不能满足大种群规模的要求.

那么, 是否可以利用低水平均匀设计表近似代替 30 以上的高水平均匀设计表呢? 通过分析和查找资料, 提出以下 2 种方案.

3) 是利用中心化 L_2^2 偏差对坐标系旋转有不变性^[11-12], 将 $n/2$ 水平的均匀设计表中的任 2 列交换便得到另一均匀性等价的 $n/2$ 水平均匀设计表, 利用它们共同组成 n 个个体的均匀表. 当种群规模大于 30 时, 如种群规模为 40, 则种群可由 $u_{20}(20^5)$ 的 20 个个体和其均匀性等价的 $u_{20}(20^5)$ 的 20 个个体共同构成. 由于每一个体都是其所处水平上的均匀设计个体即都具有均匀性. 同时, 这 2 组个体又是均匀性等价的, 不存在各区域的相互作用组合问题, 所以, 较大限度地在简化计算的同时, 尽可能地保持了种群的均匀性.

4) 是将取值区域 $l_j \leq x_{ij} \leq h_j$ 分割成 s 份, 在每份中采用一个均匀表. 例如, 如种群规模为 40, 则种群可由 $U_{20}(20^5)$ 的 20 个个体分别在分割区域 1 和分割区域 2 产生一个 $U_{20 \times 5}$ 的矩阵.

4 实验的设置与测试

4.1 测试环境和对象

对 F_1 : Sphere 函数, F_2 : Schaffer's 函数, F_3 : Ackley 函数, F_4 : Griewank 函数, F_5 : Rastrigin 函数, F_6 : Rosenbrock 函数分别就进行了测试, 变量域统一取为, 问题维数统一取 30 (Schaffer 为 2 维).

在对方案 1 和 2 的测试中, 测试分别基于 LDW-PSO 和 GLDW-PSO^[2] 进行. 以此对比均匀设计法和随机法产生初始种群时, 算法表现出来的搜索性能. PSO 种群规模为 $m = 30$, 学习因子 $c_1 = c_2 = 2$, 惯性权重 w 随在迭代过程中线性递减, 变化范围是 0.9 ~ 0.4, $T_{end} = 0.5$. 均匀设计表选择 $U_{30}(30^{29})$.

在对方案 3 和 4 的测试中, 基本表(均匀设计表和使用表)使用方案 1 所用的表. 测试都是基于 LDW-PSO 进行, 其中学习因子 $c_1 = c_2 = 2$, 惯性权重

随在迭代过程中线性递减,变化范围是 0.9 ~ 0.4.

4.2 测试结果与分析

由于篇幅所限,测试结果图只给出 Sphere 函数和 Rosenbrock 函数的.

图 2 和图 3 是方案 1 方案 2 的测试曲线. 图中 UD 表示初始种群由均匀设计产生, rand 表示初始种群随机产生;GLDW 表示基于 GLDW-PSO 算法, LDW 表示基于 LDW-PSO 算法. 通过测试可以发现:

1) 方案 1 和方案 2 效果差别不大,在测试中各有胜出,但方案 1 更易于编程实现.

2) 对于导向性好的优化问题,UD-PSO 效果提升并不明显好;但对 Rosenbrock 这种导向性复杂的函数,算法后期效果较好. 测试中发现,若以函数值是否小于 1 000 做为收敛与否的判据,对 Rosenbrock 函数采用普通 LDW-PSO 在到达最大迭代次数时,收敛概率只有 10% 左右;而采用 UD-PSO 算法时,收敛概率提升为 67% .

3) 在测试初期,UD-PSO 收敛速度较 LDW-PSO 慢,但在 400 代以后,适应值要好于 LDW-PSO.

4) 采用 UD-PSO 收敛率和最佳适值比较稳定,偏差小.

5) UD-PSO 对于 LDW-PSO 和 GLDW-PSO 都适用. 一般来说,使用 GLDW-PSO 算法时效果更好,这主要是因为 GLDW 算法自身优势有关. 但在 Rosenbrock 等某些函数或优化问题中,采用 UD 设计种群可能会缩小 LDW-PSO 和 GLDW-PSO 算法之间的差距,这可以通过图 3 中的 UD-LDW 和 UD-GLDW 性能比较得出.

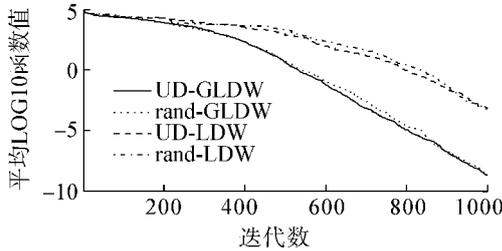


图 2 Sphere 函数 Log10 测试图

Fig. 2 Testing for Sphere Log10 function

图 4 ~ 6 为方案 3 和 4 的测试曲线. 图中,UD30 表示按方案 3 通过坐标旋转设定初始种群,UD60 表示按方案 4 通过区域分割来得到初始种群;LDW30 和 LDW60 分别表示随机产生种群规模为 30 和 60 的初始种群. 通过测试可以发现:

1) 在搜索效果方面,方案 3 的效果要差于方案 4,方案 3 的效果基本和方案 1 基本相同.

2) 方案 3 要优于随机种群生成法,说明改进的 UD-PSO 完全可以适用于大种群特点.

3) 与小种群一样:对于导向性好的优化问题,UD-PSO 效果提升并不明显好;但对 Rosenbrock 这种导向性复杂的函数后期效果较好,可以显著提高算法的收敛率.

4) 在测试初期,UD-PSO 收敛速度仍较 LDW-PSO 慢,但在中后期以后,适应值要好于 LDW-PSO.

5) 采用 UD-PSO 收敛率和最佳适值比较稳定,偏差小.

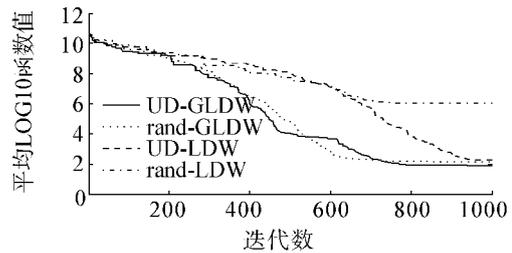


图 3 Rosenbrock 函数 Log10 测试图

Fig. 3 Testing for Rosenbrock Log10 function

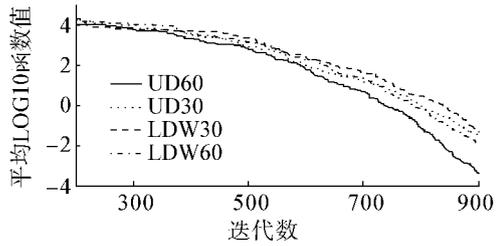


图 4 Sphere 函数 Log10 测试图

Fig. 4 Testing for sphere log10 function

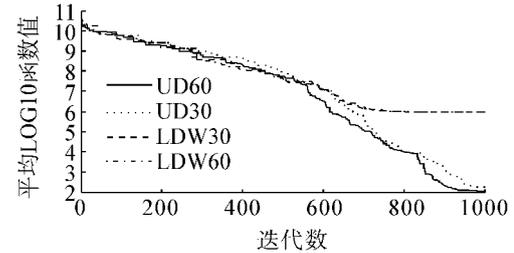


图 5 Rosenbrock 函数 Log10 测试图

Fig. 5 Testing for rosenbrock log10 function

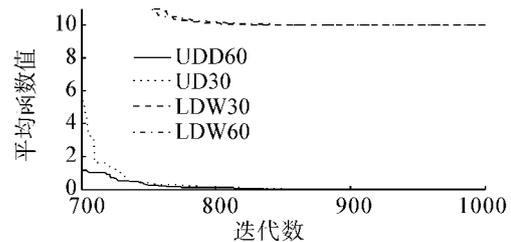


图 6 Rosenbrock 函数测试图后期

Fig. 6 Data changing in the end of testing for Rosenbrock function

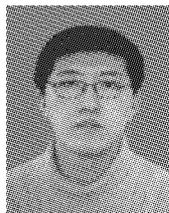
5 结束语

针对粒子群算法搜索效率随机,普通的粒子群算法难以满足某些实时优化的工程需要的现状,提出了基于均匀设计法确定关键代次种群的 PSO 算法. 利用均匀设计方法产生 PSO 算法的初始种群,使种群中的粒子在搜索空间分布更好地保持均匀分散性. 提出和对比了 4 种均匀设计方案,且都没有增加算法计算复杂性. 通过测试和对比分析表明,新算法能明显影响粒子的收敛性、快速性,能使算法具有更稳定的搜索效率和搜索精度,减少粒子聚集和搜索早熟的发生.

参考文献:

- [1] KENNEDY J E R. Particle swarm optimization[C]//Proc of IEEE International Conference on Neural Networks. Perth, WA, USA, 1995: 1942-1948.
- [2] 刘宏达. 粒子群算法的研究及其在船舶工程中的应用[D]. 哈尔滨: 哈尔滨工程大学, 2007.
LIU Hongda. Research of particle swarm optimization algorithm and its application in ship engineering[D]. Harbin: Harbin Engineering University, 2007.
- [3] 张铃, 张钺. 佳点集遗传算法[J]. 计算机学报, 2001, 24(09): 917-922.
ZHANG Ling, ZHANG Bo. Good point set based genetic algorithm[J]. Chinese Journal of Computers, 2001, 24(9): 917-922.
- [4] XUE Guochen, XIN Li. Application of uniform design and genetic algorithm in optimization of reversed-phase chromatographic separation[J]. Chemometrics and Intelligent Laboratory Systems, 2003, 68(2): 157-166.
- [5] 薛明志, 钟伟才, 刘静, 焦李成. 正交 Multi-Agent 遗传算法及其性能分析[J]. 控制与决策, 2004, 19(3): 290-294.
XUE Mingzhi, ZHONG Weicai, LIU Jing, JIAO Licheng. Orthogonal multi-Agent genetic algorithm and its performance analysis[J]. Control and Decision, 2004, 19(3): 290-294.
- [6] 何大阔, 王福利, 张春梅. 基于均匀设计的遗传算法参数设定[J]. 东北大学学报: 自然科学版, 2003, 24(05): 409-411.
HE Dakuo, WANG Fuli, ZHANG Chunmei. Establishment of parameters of genetic algorithm based on uniform design[J]. Journal of Northeastern University: Natural Science, 2003, 24(05): 409-411.
- [7] 方开泰, 马长兴. 正交与均匀试验设计[M]. 北京: 科学出版社, 2001: 89-105.
- [8] QIN Hong, ZHANG Shangli, FANG Kaitai. Constructing uniform designs with two-or three-level. Acta Mathematica Scientia, 2006, 26(3): 451-459.
- [9] ZHENG Yongling, MA Longhua, ZHANG liyan, QIAN Jixin. On the convergence analysis and parameter selection in particle swarm optimization[C]//Proceedings of the and International Conference on Machine Learning and Cybernetics. Xi'an, China, 2003: 1802-1807.
- [10] 孙先仿, 范跃祖, 宁文如. U^* 均匀设计的均匀性研究[J]. 应用概率统计, 2001, 17(4): 341-345.
SUN Xianfang, FAN Yaozu, NING Wenru. On the uniformity of U^* uniform designs[J]. Chinese Journal of applied probability and statistics, 2001, 17(4): 341-345.
- [11] HICKERNELL F J. A generalized discrepancy and quadrature error bound[J]. Math Computation, 1998, 67: 299-322.
- [12] FANG K T, QIN H. A note on construction of nearly uniform De-signs with large number of runs[J]. Statistics and Probability Letters, 2003, 61(2): 215-224.
- [13] Fang Kaitai. [http://www.math.hkbu.edu.hk/Uniform Design/](http://www.math.hkbu.edu.hk/UniformDesign/).

作者简介:



刘宏达,男,1976年生,副教授,主要研究方向为群智能理论及其应用,在群智能领域发表论文 12 篇.



马忠丽,女,1974年生,副教授,主要研究方向为机器视觉检测技术.