

基于动态聚集距离的多目标粒子群 优化算法及其应用*

刘丽琴^{1,2} 张学良² 谢黎明¹ 李明磊² 温淑花² 卢青波²

(1. 兰州理工大学机电工程学院, 兰州 730050; 2. 太原科技大学机电工程学院, 太原 030024)

【摘要】 为了增加 Pareto 集的多样性,提高多目标优化的全局寻优能力,提出了一种基于动态聚集距离的多目标粒子群算法(DCD-MOPSO)。该算法利用改进的快速排序方法来减少计算量,采用动态变化的惯性权重和加速因子以增强算法的全局寻优能力,并基于动态聚集距离对外部集进行维护以增加 Pareto 集的多样性。通过典型测试函数的仿真实验和应用实例对 DCD-MOPSO 算法性能进行了分析,并与多目标优化算法 MOPSO 和 NSGA-Ⅱ进行了比较。结果表明,DCD-MOPSO 算法收敛速度较快,且得到的 Pareto 集分布均匀。

关键词: 粒子群算法 多目标优化 改进的快速排序法 动态聚集距离 Pareto 集

中图分类号: TP391.9; TP301.6 **文献标识码:** A **文章编号:** 1000-1298(2010)03-0189-06

Multi-objective Particle Swarm Optimization Algorithm Based on Dynamic Crowding Distance and Its Application

Liu Liqin^{1,2} Zhang Xueliang² Xie Liming¹ Li Minglei² Wen Shuhua² Lu Qingbo²

(1. College of Mechanical Electronic Engineering, Lanzhou University of Technology, Lanzhou 730050, China

2. College of Mechanical Electronic Engineering, Taiyuan University of Science and Technology, Taiyuan 030024, China)

Abstract

A multi-objective particle swarm optimization algorithm based on dynamic crowding distance (DCD-MOPSO) was proposed. Applying the improved quick sorting to reduce the time for computation, both the dynamic inertia weight and acceleration coefficients were used in the algorithm to explore the search space more efficiently. A new diversity strategy called dynamic crowding distance was used to ensure sufficient diversity amongst the solutions of the non-dominated fronts. Some benchmark functions and the optimization of four-bar plane truss were tested to compare with the performance of DCD-MOPSO and NSGA-Ⅱ. The results show that DCD-MOPSO has better convergence with even distributing of Pareto set.

Key words Particle swarm algorithm, Multi-objective optimization, Improved quick sorting, Dynamic crowding distance, Pareto set

引言

对多数多目标优化问题(multi-objective optimization problem,简称MOP)而言,各目标函数间往往存在冲突,难以优化,因为MOP中的多个目标

不存在唯一的同时达到最优的全局最优解。然而,MOP有一个非劣解集,称之为Pareto最优解集。这个解集的最优性是指:当所有目标都予以考虑时,搜索空间中Pareto最优解集之外的解不优于Pareto最优解集的解^[1]。因此,求解MOP的最大难点在于寻

收稿日期:2009-06-16 修回日期:2009-06-26

* 国家自然科学基金资助项目(50775153)、山西省自然科学基金资助项目(2008011027-1)、山西高校科技研究开发项目(200811025)和山西省研究生教育改革研究项目(20092016)

作者简介: 刘丽琴,博士生,太原科技大学讲师,主要从事多目标优化、现代优化设计理论与方法研究,E-mail: liuliqin56@sohu.com

通讯作者: 张学良,教授,博士生导师,主要从事进化算法、现代优化设计理论与方法研究,E-mail: zhang_xue_l@sina.com

找非劣最优解集。传统的求解 MOP 的方法是通过加权法将多目标问题转化为单目标优化问题,但该方法要求对问题背景有很强的先验知识,因此难以有效处理 MOP 问题^[2]。基于种群操作的进化计算可隐并行地搜索解空间中的多个解,并能利用不同解之间的相似性来提高其并发求解的效率,故适用于 MOP 的求解^[3]。

粒子群算法 (particle swarm optimization, 简称 PSO)^[4] 是一种群体智能优化算法,具有快速收敛和易于实现的特点,在单目标优化问题中得到了成功的应用。研究发现,PSO 算法在求解多目标优化问题时也展现出极强的竞争力^[5-11]。国内外文献在多目标优化算法中对外部集进行维护多采用拥挤距离或拥挤度排序,至今未见将动态聚集距离应用到 MOPSO 对外部集进行维护以增加 Pareto 集的多样性的报道。为了增加 Pareto 集的多样性,提高多目标粒子群算法的全局寻优能力,本文提出一种基于动态聚集距离的多目标粒子群优化算法 (DCD-MOPSO)。

1 多目标优化问题的数学描述

一般地,多目标优化问题就是要求所有目标函数在满足约束条件下越小越好,一个含有 n 个决策变量, m 个目标变量的多目标优化问题可表述为^[12]

$$\min \mathbf{y} = f(\mathbf{x}) = (f_1(x), f_2(x), \cdots, f_m(x)) \quad (1)$$

s. t.
$$\begin{cases} g_i(x) \leq 0 & (i = 1, 2, \cdots, p) \\ h_j(x) = 0 & (j = 1, 2, \cdots, q) \end{cases}$$

其中, $\mathbf{x} = (x_1, x_2, \cdots, x_n) \in X \subset \mathbf{R}^n$ 为 n 维的决策矢量, X 为 n 维的决策空间, $\mathbf{y} = (y_1, y_2, \cdots, y_m) \in Y \subset \mathbf{R}^m$ 为 m 维的目标矢量, Y 为 m 维的目标空间。 $f_k(x)$ ($k = 1, 2, \cdots, m$) 为 m 个目标函数,满足 p 个不等式约束和 q 个等式约束。在此基础上,给出以下定义。

定义 (Pareto 占优) 假设 $x_a, x_b \in X_f$ 是式 (1) 所示多目标优化问题的两个可行解,则称与 x_b 相比, x_a 是 Pareto 占优的,当且仅当任取 $i = 1, 2, \cdots, m$, $f_i(x_a) \leq f_i(x_b)$ 且有 $j = 1, 2, \cdots, m, f_j(x_a) < f_j(x_b)$, 记作 $x_a > x_b$, 也称为 x_a 支配 x_b 。

此外,对于一个解 x^* 是 Pareto 最优解 (或非支配解),则当且仅当在问题的可行域中不存在另外一个解 $x' > x^*$ 。所有 Pareto 最优解的集合就构成 Pareto 最优集, Pareto 最优集中的解向量构成了 Pareto 最优前沿。

2 基于动态聚集距离的多目标粒子群优化算法

多目标进化算法需要保证 Pareto 最优边界的收

敛性和多样性特征,DCD-MOPSO 算法采用动态变化的惯性权重和加速因子以及变异操作增强了算法的全局寻优能力;采用外部集存储种群产生的非支配解,通过非支配解之间动态聚集距离对外部集进行维护,可以得到分布更为均匀的 Pareto 前沿。

2.1 改进的快速排序法

算法每次循环都从种群中选取 2 个比较个体。其中,第 1 个依次与种群中的其他个体进行比较,第 2 个则是每一轮比较中与第 1 个比较个体无关的首个或者是支配第 1 个比较个体的个体。若在排序过程中找到了第 2 个比较个体,则这 2 个比较个体均加入与其他个体的比较,任何被这 2 个比较个体一所支配的个体都被清除,只有同时不被此二者支配的个体才能保留下来并进入下一轮比较^[13]。

2.2 动态变化的惯性权重与加速因子

2.2.1 惯性权重的动态设置

粒子群算法中,惯性权重用于控制粒子的历史速度对当前速度的影响,对群体的全局探测和局部开发起着平衡作用,对收敛性能具有重要的影响。一般地,较大的惯性权重可加大粒子群的搜索空间,较小的惯性权重则有利于改善粒子的局部搜索性能。在本文算法中,惯性权重随着迭代的进行线性减小,在每次迭代中,惯性权重为

$$\omega(t) = (\omega_i - \omega_e) \frac{t_{\max} - t}{t_{\max}} + \omega_e \quad (2)$$

式中 ω_i ——惯性权重初始值
 ω_e ——惯性权重终值 t ——迭代次数
 $t_{\max}$ ——最大迭代次数

由式 (2) 可看出,惯性权重值在算法前期趋向于取较大的初始值,而后期趋向于取较小的终值。这样可保证算法开始时各粒子能够以较大的速度在全局范围内搜索到较好的种子,而在搜索后期较小的 ω 值可保证粒子能够进行精细的局部搜索。通过惯性权重线性减小使得算法能够达到全局探测与局部开发间的有效平衡,增强算法的全局寻优能力。

2.2.2 加速因子的动态设置

粒子群算法中,加速因子用来平衡个体和群体认知能力,通常取常数。在 DCD-MOPSO 中,为了更好地平衡粒子在搜索空间的探测和开发能力,加速因子随迭代的进行动态变化,在每次迭代中,加速因子 c_1 和 c_2 为^[14]

$$c_1(t) = (c_{1e} - c_{1i}) \frac{t}{t_{\max}} + c_{1i} \quad (3)$$

$$c_2(t) = (c_{2e} - c_{2i}) \frac{t}{t_{\max}} + c_{2i} \quad (4)$$

式中 c_{1i}, c_{2i} —— c_1 和 c_2 的初始值

$c_{1e}、c_{2e}$ —— c_1 和 c_2 的终值

c_1 线性减小, c_2 线性增加, 保证了粒子的全局寻优能力。

2.3 动态聚集距离(DCD)计算

在 DCD-MOPSO 算法中保留外部集, 以存储进化过程中产生的非支配个体, 基于文献[11]提出的动态聚集距离排序进行外部集的削减, 可以保证外部集中个体分布广度和分布均匀性。

个体 i 的动态聚集距离为

$$I_{id} = I_i / \lg(1/V_i) \tag{5}$$

其中
$$I_i = 1/m \sum_{k=1}^m |I_{i+1,k} - I_{i-1,k}| \tag{6}$$

$$V_i = 1/m \sum_{i=1}^m (|I_{i+1,k} - I_{i-1,k}| - I_i)^2 \tag{7}$$

式中 I_i ——个体 i 的聚集距离

m ——目标维数

$I_{i,k}$ ——第 i 个个体 I_i 在第 k 维目标上排序后的第 k 维函数值

在外部集维护过程中, 若外部集中的非支配个体数 N 未达到规定规模 M 时, 将产生的非支配个体直接存储到外部集; 否则, 根据 DCD 从外部集中去除 $N - M$ 个个体。其更新策略如下: ① 根据式(5)计算外部集中每个个体的动态聚集距离 I_{id} 。② 对外部集中的个体按动态聚集距离 I_{id} 进行排序。③ 将外部集中的动态聚集距离最小的个体从外部集中去除。④ 若外部集中非支配个体数 $N \leq M$, 则结束外部集维护; 否则返回①, 继续执行。

基于 DCD 的分布性保持策略在进行外部集维护时表现出两个重要特点: ① 每次只去除外部集中 DCD 最小的一个个体。然后, 重新计算外部集中个体的 DCD。这样就避免了一次性去除多个个体造成某一区域个体缺失的现象, 故而可以得到分布更为均匀的 Pareto 前沿。② 由于使用式(5)来计算个体的 DCD, 而式(5)不仅考虑了个体间的拥挤情况, 而且还考虑到了个体在不同维目标上聚集距离的差异情况。这样, 就有利于维护在不同维目标上聚集距离差异较大的 Pareto 前沿的分布性。

2.4 全局最优值的更新

多目标粒子群算法需要获得分布均匀的 Pareto 前沿, 因此全局最优值 G_{best} 选择不同于单目标 PSO 中仅选取目标值最优值个体, 而要选择处于 Pareto 前沿中分散区域的个体, 引导粒子群向分散区域进化。用以下策略对全局最优个体进行更新:

(1) 若 Pareto 中所有个体的拥挤距离都为无穷大, 即仅包括数量较少的边界个体, 则随机选择一个作为 G_{best} 。

(2) 若 Pareto 中含有拥挤距离不为无穷大的个体, 则使用轮盘法选择, 即以较大概率选择拥挤距离较大的个体为 G_{best} 。计算公式为

$$P(x_i) = \frac{I_i}{N} \tag{8}$$
$$\sum_{i=1}^N I_i$$

其中 $P(x_i)$ 为个体 i 被选中的概率, I_i 为个体 i 的聚集距离, N 为当前 Pareto 解集的大小。需要注意的是, 个体聚集距离含有无穷大会造成轮盘法选择失效, 因此定义个体聚集距离无穷大(边界个体)的个体的聚集距离为除去这些个体后其余个体的拥挤距离的中位数。

实验表明, 使用这种方法选择 G_{best} 所得到的 Pareto 前沿有更好的分布性。

2.5 DCD-MOPSO 算法描述

(1) 初始化种群。令迭代代数, 最大迭代次数为 t_{max} , 给定种群规模 M , 在决策变量的可行域中随机产生每个粒子的位置 x_i , 所有粒子的初始速度 v_i 均设为零, 并设个体极值 $P_{i,best}$ 等于 x_i ; 计算每个粒子的目标函数值。

(2) 利用改进的快速排序法, 根据目标函数值对粒子进行支配关系排序, 将非支配解存储到外部集 A 中, 并对 A 中的个体按动态聚集距离进行降序排列。

(3) 根据全局最优值更新策略设置 G_{best} 。

(4) 更新种群的速度和位置, 对种群进行变异操作, 计算目标函数值, 依据支配关系, 更新个体极值。

(5) 更新外部集 A 。若种群新产生的非支配解不受外部集 A 中个体的支配, 则将其插入到 A 中, 删除 A 中所有受新非支配解支配的解。若外部集 A 已满, 则基于个体动态聚集距离进行 A 的削减。

(6) $t = t + 1$, 若 $t < t_{max}$, 则转步骤(3); 否则, 输出外部集, 获得 Pareto 最优解集。

3 仿真结果与分析

采用收敛性能(GD)和分散性能(SZVP)^[7]的度量方法, 对 DCD-MOPSO 算法与文献[7]中提出的多目标粒子群算法 MOPSO 及 NSGA-II 算法进行对比研究。3 种算法种群规模均设为 100, 外部集规模为 100, 迭代 500 次, DCD-MOPSO 惯性权重与加速因子设置为: $\omega_i = 0.4$, $\omega_e = 0.9$, $c_{1i} = 2.5$, $c_{1e} = 0.5$, $c_{2i} = 0.5$, $c_{2e} = 2.5$ 。对于 MOPSO 和 NSGA-II 算法所涉的其他参数与文献[7]一致。选取文献[7]中测试函数 2——Kursawe 函数和测试函数 4——Deb 函数, 各个算法分别进行 30 次独立实验, 图 1~2 为

3种算法对测试函数所生成的 Pareto 前沿与理论 Pareto 前沿的对比图,表 1 和表 2 分别为 3 种算法对 Kursawe 函数和 Deb 函数的比较评价指标 GD 值和 SP 值。

对于 Deb 函数,由图 2 可以看出,3 种算法中

DCD-MOPSO 与 MOPSO 的非劣解都能完全覆盖 Pareto 前沿,NSGA-Ⅱ 的非劣解却不能完全覆盖 Pareto 前沿。由表 1 和表 2 可以看出,与 MOPSO 及 NSGA-Ⅱ 相比 DCD-MOPSO 具有较好的收敛性和分布性。

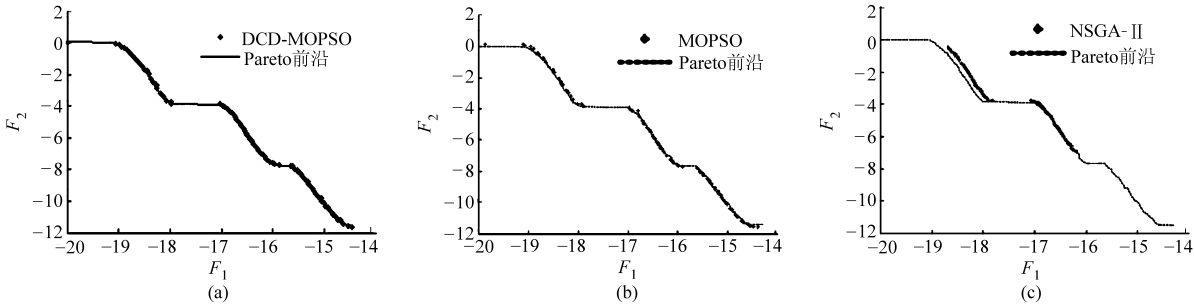


图 1 3 种算法对 Kursawe 函数的实验结果

Fig.1 Results from function Kursawe experiments with three algorithms

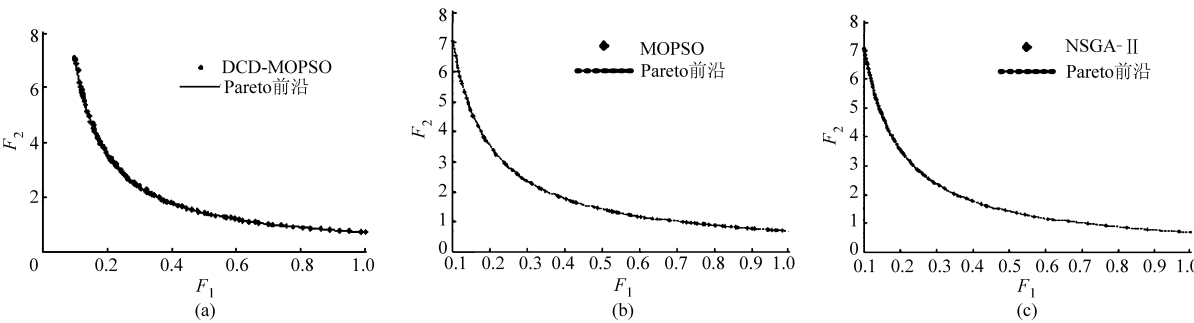


图 2 3 种算法对 Deb 函数的实验结果

Fig.2 Results from function Deb experiments with three algorithms

由表 1、表 2 及图 1 可以看出,对于 Kursawe 函数, MOPSO 的非劣解能完全覆盖 Pareto 前沿,但分

布性较差;NSGA-Ⅱ 的 SP 值较小,分布性较好,但 GD 值较大,说明收敛性差。DCD-MOPSO 的收敛性和分布性相比最好。

表 1 测试函数的 GD 值

Tab.1 GD of benchmark functions				
测试函数及实例	GD	DCD-MOPSO	MOPSO	NSGA-Ⅱ
Kursawe	平均值	0.005 33	0.008 45	0.029 255
	中值	0.005 18	0.008 45	0.017 357
	标准差	0.000 46	0.000 51	0.027 17
Deb	平均值	0.031 18	0.032 73	0.044 236
	中值	0.000 48	0.000 51	0.000 856
	标准差	0.059 82	0.060 62	0.073 68

表 2 测试函数的 SP 值

Tab.2 SP of benchmark functions				
测试函数及实例	SP	DCD-MOPSO	MOPSO	NSGA-Ⅱ
Kursawe	平均值	0.032 596	0.097 47	0.036 136
	中值	0.029 538	0.103 96	0.036 085
	标准差	0.009 008	0.016 75	0.010 977
Deb	平均值	0.033 854	0.083 58	0.037 447
	中值	0.028 942	0.054 94	0.035 529
	标准差	0.008 176	0.118 21	0.009 238

4 DCD-MOPSO 算法的应用

以曳引效率 η 、曳引功率 P 、制动力矩 M_d 作为优化目标的电梯有齿曳引系统的数学模型为

max $\eta = \frac{100 - k_1 \sqrt{i_{12}}}{100}$ (9)

min $P = \frac{\pi(1 - \varphi) QnD(100 - 3.6\sqrt{i_{12}})}{6 \times 10^7 i_1 C i_{12} (100 - k_1 \sqrt{i_{12}})}$ (10)

min $M_d = \frac{39\pi(1 - \varphi) QD(100 - 3.6\sqrt{i_{12}})}{2.4 \times 10^8 i_1 C i_{12}}$ (11)

s. t. $v = \frac{\pi D n}{6 \times 10^4 i_1 i_{12}} = v_e$ (12)

$42 \leq D/d \leq 50$ (13)

$D \geq \frac{2.650 8 \cos \gamma (1 + v)}{i_1 n_s d (\pi - 2\gamma - \sin(2\gamma)) (12.5 + 4v)}$ (14)

式中 i_1 ——电梯曳引比 i_{12} ——减速器减速比
 k_1 ——曳引系统传动机构的特征系数,对于普通圆柱蜗杆副 (ZK₁, ZI), $k_1 = 3.2$,

对于圆弧圆柱蜗杆副 (ZC), $k_1 = 2.5$

n ——电动机额定转速, r/min

D ——曳引轮直径, mm

Q ——电梯额定载重量, N

C ——效率常数, 取 0.50 ~ 0.55

φ ——平衡系数, 取 0.4 ~ 0.6

n_s ——曳引钢丝绳根数

2γ ——曳引轮切口角, rad

v ——电梯的运行速度, m/s

d ——曳引钢丝绳的直径^[15], mm

以普通有齿货梯曳引机为例, 选取曳引效率损失 $1 - \eta$ 、曳引功率 P 和制动力矩 M_d 为优化目标的优化模型

$$\min F = (1 - \eta, P, M_d)$$

满足式 (12) ~ (14) 的约束条件, 数学模型中的所有参数选取与文献 [15] 一致。

DCD-MOPSO 算法应用于该实例所使用的种群规模为 500, 外部集规模为 20, 迭代 500 次, 惯性权重与加速因子设置为: $\omega_i = 0.4$, $\omega_e = 0.9$, $c_{1i} = 2.5$, $c_{1e} = 0.5$, $c_{2i} = 0.5$, $c_{2e} = 2.5$ 。图 3 为 DCD-MOPSO 算法求解该三目标模型所得 Pareto 前沿, 所得 20 个 Pareto 最优解中至少有 5 个优于文献 [15] 表 2 中根据决策指数最大得到的综合性能最优的解。例如, $1 - \eta = 0.180\ 3$, $P = 4.722\ \text{kW}$, $M_d = 4.194\ \text{kN} \cdot \text{m}$; $1 - \eta = 0.182$, $P = 4.627$, $M_d = 4.101$; $1 - \eta = 0.185$, $P = 4.448\ \text{kW}$, $M_d = 3.925\ \text{kN} \cdot \text{m}$ 等。由此可以看出:

该算法应用于机械产品曳引系统性能的多目标优化中能够获得比文献 [15] 和 SPEA2 分布性和边界性更好的 Pareto 前沿。

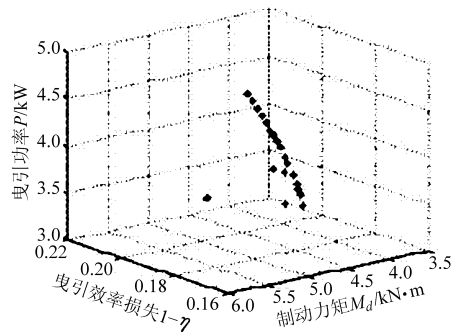


图 3 三目标优化设计的 Pareto 前沿

Fig. 3 Pareto front of three objectives optimization design

5 结束语

提出了利用改进的快速排序方法来减少优化计算量; 采用非支配个体的动态聚集距离进行外部集的缩减, 并使用轮盘赌的全局最优值更新策略可以得到分布更为均匀的 Pareto 前沿; 同时 DCD-MOPSO 算法还采用动态变化的惯性权重和加速因子增强了算法的全局寻优能力。通过典型测试函数验证, 结果表明, DCD-MOPSO 算法是一种有效的多目标进化算法, 特别是在求解三目标优化问题时的突出表现, 表明这种算法在多目标优化领域具有一定的适用性。

参 考 文 献

- 1 曾三友, 魏巍, 康立山, 等. 基于正交设计的多目标演化算法[J]. 计算机学报, 2005, 28(7): 1 153 ~ 1 162.
Zeng Sanyou, Wei Wei, Kang Lishan, et al. A multi-objective evolutionary algorithm based on orthogonal design[J]. Chinese Journal of Computers, 2005, 28(7): 1 153 ~ 1 162. (in Chinese)
- 2 王辉, 钱锋. 基于拥挤度与变异的动态微粒群多目标优化算法[J]. 控制与决策, 2008, 23(11): 1 238 ~ 1 242, 1 248.
Wang Hui, Qian Feng. Improved PSO-based multi-objective optimization by crowding with mutation and particle swarm optimization dynamic changing[J]. Control and Decision, 2008, 23(11): 1 238 ~ 1 242, 1 248. (in Chinese)
- 3 张利彪, 周春光, 刘小华, 等. 求解多目标优化问题的一种多子群体进化算法[J]. 控制与决策, 2007, 22(11): 1 313 ~ 1 316, 1 320.
Zhang Libiao, Zhou Chunguang, Liu Xiaohua, et al. A multiple subswarms evolutionary algorithm for multi-objective optimization problems[J]. Control and Decision, 2007, 22(11): 1 313 ~ 1 316, 1 320. (in Chinese)
- 4 Kennedy J, Eberhart R C. Particle swarm optimization[C] // Proceedings of 1995 IEEE International Conference on Neural Networks, Piscataway, USA, 1995: 1 942 ~ 1 948.
- 5 Parsopoulos K E, Vrahatis M N. Particle swarm optimization method in multi-objective problems[C] // Proceedings of the 2002 ACM Symposium on Applied Computing (SAC 2002), 2002: 603 ~ 607.
- 6 Coello C A C, Lechuga M S. MOPSO: a proposal for multiple objective particle swarm optimization[C] // Proc. IEEE Int. Conf. on Evolutionary Computation, Piscataway: IEEE Service Center, 2002: 1 051 ~ 1 056.
- 7 Coello C A C, Pulido G T, Lechuga M S. Handling multiple objectives with particle swarm optimization[J]. IEEE Transactions on Evolutionary Computation, 2004, 8(3): 256 ~ 279.
- 8 Hu X, Eberhart R. Multiobjective optimization using dynamic neighborhood particle swarm optimization[C] // Proc. IEEE Int. Conf. on Evolutionary Computation, Hawaii, 2002: 1 677 ~ 1 681.

- 9 Fieldsend J E, Singh S. A multi-objective algorithm based upon particle swarm optimization, an efficient data structure and turbulence[C]// Proc. UK Workshop on Computational Intelligence, Birmingham, 2002: 37 ~ 44.
 - 10 Liu D S, Tan C K, Goh C K, et al. A multi-objective mimetic algorithm based on particle swarm optimization[J]. IEEE Transactions on Systems, Man and Cybernetics, 2007, 37(1): 42 ~ 50.
 - 11 罗彪, 郑金华. 多目标进化算法中基于动态聚集距离的分布性保持策略[J]. 计算机应用研究, 2008, 25(10): 2 934 ~ 2 938.
Luo Biao, Zheng Jinhua. Dynamic crowding distance based diversity maintenance strategy in MOEAs[J]. Application Research of Computers, 2008, 25(10): 2 934 ~ 2 938. (in Chinese)
 - 12 张学良, 温淑花, 李海楠, 等. PSO 算法在多目标优化问题中的仿真应用[J]. 农业机械学报, 2007, 38(7): 112 ~ 115.
Zhang Xueliang, Wen Shuhua, Li Hainan, et al. Applications of particle swarm optimization in multi-objective problems[J]. Transactions of the Chinese Society for Agricultural Machinery, 2007, 38(7): 112 ~ 115. (in Chinese)
 - 13 郑金华. 多目标进化算法及其应用[M]. 北京: 科学出版社, 2005: 101 ~ 112.
 - 14 Praveen K T, Sanghamitra B, Sankar K P. Multi-objective particle swarm optimization with time variant inertia and acceleration coefficients[J]. Information Sciences, 2007, 177(22): 5 033 ~ 5 049.
 - 15 李中凯, 谭建荣, 冯毅雄, 等. 基于拥挤距离排序的多目标粒子群优化算法及其应用[J]. 计算机集成制造系统, 2008, 14(7): 1 329 ~ 1 336.
Li Zhongkai, Tan Jianrong, Feng Yixiong, et al. Multi-objective particle swarm optimization algorithm based on crowding distance sorting and its application[J]. Computer Integrated Manufacturing Systems, 2008, 14(7): 1 329 ~ 1 336. (in Chinese)
-

(上接第 142 页)

参 考 文 献

- 1 刘斌. 微型冷库系统优化研究[D]. 天津: 天津大学, 2003.
Liu Bin. Optimum study on the mini-cold storage[D]. Tianjin: Tianjin University, 2003. (in Chinese)
- 2 陈天及, 余克志. 装配式冷库温度场特性的实验研究[J]. 制冷, 2001(3): 21 ~ 23.
Chen Tianji, Yu Kezhi. The experimental study on the characteristics of temperature field inside a fabricated cold store[J]. Refrigeration Precooling, 2001(3): 21 ~ 23. (in Chinese)
- 3 刘斌, 杨昭, 李喜宏, 等. 微型冷库内气流组织及贮藏效果实验[J]. 天津大学学报: 自然科学版, 2005, 38(10): 897 ~ 900.
Liu Bin, Yang Zhao, Li Xihong, et al. Experiment on air distribution and storage effects in mini-cold storage[J]. Journal of Tianjin University: Science and Technology, 2005, 38(10): 897 ~ 900. (in Chinese)
- 4 Robert E, Paull. Effect of temperature and relative humidity on fresh commodity quality[J]. Postharvest Biology and Technology, 1999, 15(3): 263 ~ 277.
- 5 Bourhan Tashtoush. Nature loss from vegetable and fruit products in cold storage[J]. Food Control, 2000, 11(6): 465 ~ 470.
- 6 Hasse H, Becker M, Grossmann K, et al. Top-down model for dynamic simulation of cold-storage plants[J]. International Journal of Refrigeration, 2006, 19(1): 10 ~ 18.
- 7 郁文红, 杨昭. 小型实验用恒温恒湿保鲜库[J]. 制冷与空调, 2006(5): 89 ~ 91.
Yu Wenhong, Yang Zhao. Small constant temperature and humidity cold storage for experiment[J]. Refrigeration and Air-conditioning, 2006(5): 89 ~ 91. (in Chinese)
- 8 谭晶莹. 果蔬冷藏工艺及品质衰变理论与实验研究[D]. 天津: 天津大学, 2007.
Tan Jingying. Theoretical and experimental study on storage techniques and quality disintegration of the fruits and vegetable [D]. Tianjin: Tianjin University, 2007. (in Chinese)