

元胞多目标粒子群优化算法与其应用*

朱大林 詹腾 张屹 田红亮

(三峡大学水电机械设备设计与维护湖北省重点实验室, 宜昌 443002)

摘要: 针对现有多目标粒子群算法多样性不佳,难以平衡多目标优化的全局搜索和局部寻优的能力,提出了一种元胞多目标粒子群算法。在分析多目标粒子算法理论基础,该算法将元胞自动机思想融入粒子群算法,研究粒子之间相互关系和信息传递机制,并提出一种粒子飞行速度控制策略。实验证明,新算法相对于4种比较算法,在求解含有无约束和有约束的多目标优化问题时有更好的收敛性和多样性,将其应用于盘式制动器优化设计,得到的解精度更高。

关键词: 元胞自动机 粒子群算法 速度控制策略 多目标优化

中图分类号: TP301.6 **文献标识码:** A **文章编号:** 1000-1298(2013)12-0280-08

Algorithm and Application of Cellular Multi-objective Particle Swarm Optimization

Zhu Dalin Zhan Teng Zhang Yi Tian Hongliang

(Hubei Key Laboratory of Hydroelectric Machinery Design & Maintenance, China Three Gorges University, Yichang 443002, China)

Abstract: For improving the diversity of existing multi-objective particle swarm optimization algorithm and keeping the balance between exploration and exploitation well, a multi-objective cellular PSO was proposed. The algorithm combined the concept of cellular automata with the multi-objective PSO theory. In addition, the relationship between the particles and the information transmission mechanism was studied, and a particle flight speed control strategy was presented. The results indicate that the improved algorithm outperforms the four compared algorithms concerning the convergence and diversity in solving multi-objective optimization problems with unconstraint and constraint. And also, the new algorithm can get more accurate solutions when applied in disc brake design problem.

Key words: Cellular automata Particle swarm optimization Speed control strategy Multi-objective optimization

引言

对于多目标优化问题通常不存在各目标均为全局最优解,而是存在一组非劣解(非支配解),称为Pareto最优解集。相对于传统的多目标优化方法(加权法等),多目标进化算法(Multi-objective evolutionary algorithms, MOEA)已被证明是解决多目标优化问题的一类有效方法。粒子群优化算法(Particle swarm optimization, PSO)具有编程简单、算法直观、实现容易的特点,在许多优化问题中得到应

用^[1-5],并且很多情况下比遗传算法更有效率,所以PSO算法在求解多目标优化问题方面,许多学者对多目标粒子群算法的理论与应用进行了研究^[6-10]。

上述MOPSO算法存在以下问题:①Pareto最优集的多样性保持能力差,易于陷入局部收敛。②平衡全局搜索和局部寻优能力不佳,且更新策略复杂,降低了算法效率。本文将元胞自动机思想和粒子群算法相结合,得到一种元胞多目标粒子群算法(Cellular multi-objective particle swarm optimization, CMOPSO)。该算法将粒子分布于一种二维空间拓

扑结构中,研究粒子之间的相互关系和信息传递。为了更好地控制粒子的飞行过程,提出一种限制粒子速度的策略。通过 3 个无约束的多目标测试函数和 2 个含有约束的测试函数对盘式制动器进行优化设计。

1 多目标粒子群算法

1.1 粒子群算法基本理论

粒子群优化算法 PSO 是由 Kennedy 和 Eberhart 于 1995 年提出的一种优化算法^[11]。PSO 算法的运行机理不是依靠个体的自然进化规律,而是对生物群体的社会行为进行模拟。在生物群体中存在着个体与个体、个体与群体间的相互作用、相互影响的行为,这种行为体现的是一种存在于生物群体中的信息共享的机制。PSO 算法是对这种社会行为的模拟,即利用信息共享机制,个体间可以相互借鉴经验,从而促进整个群体的发展进化。因其具有易理解、易实现、全局搜索能力强等特点,大量应用于实际工程领域。其基本数学表达式如下:

设群体规模为 N_p , 则第 i ($i = 1, 2, \dots, N_p$) 个粒子的位置可以表示为 x_i , 速度表示为 v_i , 它的个体历史最优解记为 p_{ibest} , 群体的历史最优解用 p_{gbest} 表示。因此,任一粒子 i 将根据式(1)、(2)来更新其速度和位置。

$$v_i(t+1) = wv_i(t) + c_1r_1(t)(p_{ibest}(t) - x_i(t)) + c_2r_2(t)(p_{gbest}(t) - x_i(t)) \quad (1)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (2)$$

式中 w ——权重因子 c_1, c_2 ——学习因子

r_1, r_2 ——(0,1)的随机数

式(1)由 3 部分组成,第一部分是粒子先前的速度,说明了粒子目前的状态;第二部分是认知部分,表示粒子本身的思考;第三部分为社会部分,表示粒子受到的影响。这 3 个部分共同决定了粒子的空间搜索能力。

1.2 多目标粒子群算法基本框架

多目标粒子群算法 (Multi-objective particle swarm optimization, MOPSO) 的基本框架和多目标进化算法类似,但是存在一些差异。MOPSO 一般不需要根据 Pareto 支配和密度等信息进行适应度赋值,简化了算法设计^[12]。含有精英策略的 MOPSO 基本框架如下:

(1) 令 $t=0$, 在初始化范围内,随机生成粒子种群 NP_t , 包括随机位置和速度,计算各粒子对应的目标向量,将其中非支配解存入外部文档 M_t 。

(2) 定义初始自身历史最优值 p_{ibest} (一般为初始化粒子本身) 和种群历史最优值 p_{gbest} 。

(3) 在保证粒子的搜索空间内,对粒子群体进行进化操作,不断地更新粒子的速度和位置。

(4) 根据新的非支配解,维护外部文档 M_{t+1} , 形成新种群 NP_{t+1} , 并依据支配关系更新 p_{ibest} 和 p_{gbest} 。

(5) $t=t+1$, 如果满足条件则终止,否则转到步骤(2)。

2 元胞多目标粒子群算法

Alba Enrique 和 Bernabé Dorronsoro^[13] 将元胞自动机模型和遗传算法基本理论相互结合,得到一种元胞遗传算法。此类算法通常将种群中的每个个体分配于一个网格拓扑结构中,而每个个体严格规定只能与其邻居结构内个体进行遗传操作。这一特性决定了元胞遗传算法(图 1a)相对于随机交配的遗传算法(图 1b)而言,能够令最优解在种群内更平缓的扩散,不同范围内的个体收敛到搜索空间的不同区域,从而形成一个小小生境,保持了种群的多样性,增强了算法的局部搜索能力。每个个体组成的一个小小生境群体,也赋予了该类算法存在隐并行机制。后续的研究发现将元胞自动机和遗传算法相结合能够降低算法的选择压力,能够很好地平衡全局搜索和局部寻优的能力,尤其在算法多样性方面有一定的提高。

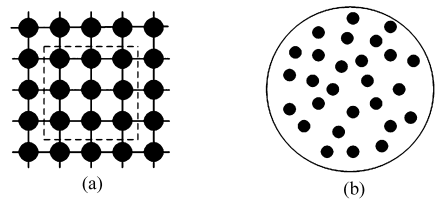


图 1 种群分布图

Fig. 1 Distribution of population

(a) 元胞种群机构 (b) 随机交配种群机构

基于此,本文在研究多目标粒子群算法的基础上,引入元胞自动机思想,提出一种元胞多目标粒子群算法。借鉴元胞遗传算法的优势,结合粒子群算法本身的特点,将粒子群算法收敛速度快,全局搜索性能好的优势融入元胞自动机模型中,以提高算法的多样性与局部搜索能力,从而提高算法的性能。

2.1 元胞粒子群算法

基于粒子群的特点,首先随机产生粒子,然后将每一个粒子被按序无重复地放置于一个二维环形网格拓扑结构(图 2a)中,每个粒子看成一个元胞,每个元胞赋予相同的邻居数,且元胞的个数等于种群大小。如图 2b 所示,采用 Moore 邻居结构,每个元胞拥有 8 个邻居个体,其中灰色圆圈为元胞个体,含 # 字型的灰黑色圆圈为中心元胞个体,虚线框示的为其邻居个体,黑色个体为两个邻居之间相互重叠的

邻居。实际上,这种邻居个体相互重叠为算法提供了一种隐性迁移机制。此机制能够促使最优个体平缓地在整个种群中进行扩散,从而降低算法的选择压力,提高算法的多样性。

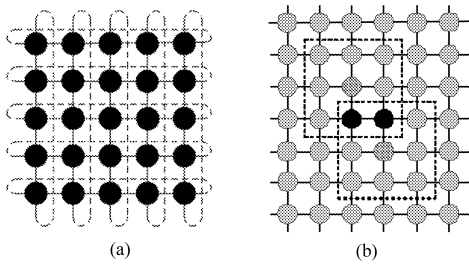


图2 2种群、邻居结构示意图

Fig.2 Two diagram of population and neighborhood structure

(a) 二维环形拓扑结构 (b) Moore型邻居结构

粒子在每次更新位置时,需知自身历史最优位置 p_{lbest} 和种群的全局最优位置 p_{gbest} 。每个粒子的自身历史最优位置更新方式为

$$p_{lbest}(t+1) = \begin{cases} x(t+1) & (x(t+1) \geq p_{lbest}(t)) \\ p_{lbest}(t) & (x(t+1) < p_{lbest}(t)) \end{cases} \quad (3)$$

由于本文将所有粒子放置于二维环形网格拓扑结构中,每个粒子都有其邻居粒子,因此为了更好地指导粒子飞行,增加算法多样性,将粒子自身历史最优位置 p_{lbest} 更替为邻居最优位置 p_{nbest} 。计算邻居最优位置 p_{nbest} 方法如下:

假设粒子的个体最优位置 p_{lbest} 和邻居最优位置 p_{nbest} 被定义为元胞状态,这两种信息分别由 $S_i^t(p_i)$ 和 $S_i^t(p_n)$ 表示,则邻居最优位置 p_{nbest} 的选取规则为

$$S_i^{t+1}(p_n) = f(S_i^t(p_i), S_{i+1}^t(p_{i+1}), \dots, S_{i+N}^t(p_{i+N})) = \text{Ranking}(\text{fitness}(S_i^t(p_i)), \text{fitness}(S_{i+1}^t(p_{i+1})), \dots, \text{fitness}(S_{i+N}^t(p_{i+N}))) = \min(\text{Ranking}(S_i^t(p_i), S_{i+1}^t(p_{i+1}), \dots, S_{i+N}^t(p_{i+N}))) \quad (4)$$

其中

$f(S_i^t(p_i)) = [f_1(S_i^t(p_i)), f_2(S_i^t(p_i)), \dots, f_M(S_i^t(p_i))]$
 $\text{Ranking}()$ 函数是对不同个体 $f(S^t(p))$ 函数之间的支配关系进行排列,其数值是由该个体被其他个体支配多少决定,如被一个个体支配,则 $\text{Ranking}() = 1$,依次累加类推; N 表示邻居结构大小; M 表示目标函数个体。

对于种群到达的最优位置 p_{gbest} ,即为存储于外部文档中的非支配解。种群最优个体的选取采用随机的方式在外部文档中选取一个个体作为 p_{gbest} 。

2.2 粒子速度控制策略

粒子群算法具有很好的全局搜索性能,但其局

部搜索性能相对较弱,导致这一原因很大程度上是因为粒子飞行速度过快所致,因此针对该问题,采用如下策略控制粒子的飞行速度,提高算法的局部搜索能力。

(1) 采用 Clerc 提出的 PSO 收敛因子模型^[14],实验证明该收敛因子模型能够确保算法的收敛。收敛因子模型为

$$v_i(t+1) = k[uv_i(t) + c_1r_1(p_l - x_i(t)) + c_2r_2(p_g - x_i(t))] \quad (5)$$

其中 $k = \frac{2}{|2 - \varphi - \sqrt{\varphi^2 - 4\varphi}|}$ (6)

$$\varphi = \begin{cases} c_1 + c_2 & (c_1 + c_2 \geq 4) \\ 0 & (c_1 + c_2 < 4) \end{cases}$$

式中 k ——收敛因子

(2) 粒子在飞行过程中如果飞行速度过快,将导致算法陷入局部收敛,而有效地控制粒子的飞行速度有利于提高算法的性能^[15]。基于此,对粒子采用如下限速操作

$$v_{i,j}(t) = \begin{cases} \frac{2}{3}\delta & (v_{i,j}(t) \geq \delta) \\ -\frac{1}{3}\delta & (v_{i,j}(t) < \delta) \\ v_{i,j}(t) & (\text{其他}) \end{cases} \quad (7)$$

其中 $\delta = j_{\max} - j_{\min}$ (8)

式中 j_{\max}, j_{\min} ——决策变量的上、下约束值

(3) 为了更好地平衡全局搜索和局部搜索,对影响速度的权重因子 c_1 和 c_2 分别进行线性递增/减。在算法初期引导算法向 Pareto 前端逼近,加大社会认知部分的影响因子,减少邻居认知部分的影响因子;随着进化的不断进行,种群中非支配个体增加,为了获得更好的分布性,算法将加大邻居认知部分的影响因子,减少社会认知部分的影响因子。因此,影响因子 c_1, c_2 为

$$c_1 = (c_{1\max} - c_{1\min}) \frac{g_{en}}{g_{en\max}} + c_{1\min} \quad (9)$$

$$c_2 = (c_{2\max} - c_{2\min}) \frac{g_{en\max} - g_{en}}{g_{en\max}} + c_{2\min} \quad (10)$$

式中 $c_{1\max}, c_{1\min}$ ——影响因子 c_1 的最大、最小值
 $c_{2\max}, c_{2\min}$ ——影响因子 c_2 的最大、最小值
 g_{en} ——当前迭代数
 $g_{en\max}$ ——最大迭代次数

c_2 线性减小, c_1 线性增加,保证了粒子的局部寻优能力。

2.3 算法流程

元胞多目标粒子群算法的示意图如 3 所示,基本流程为

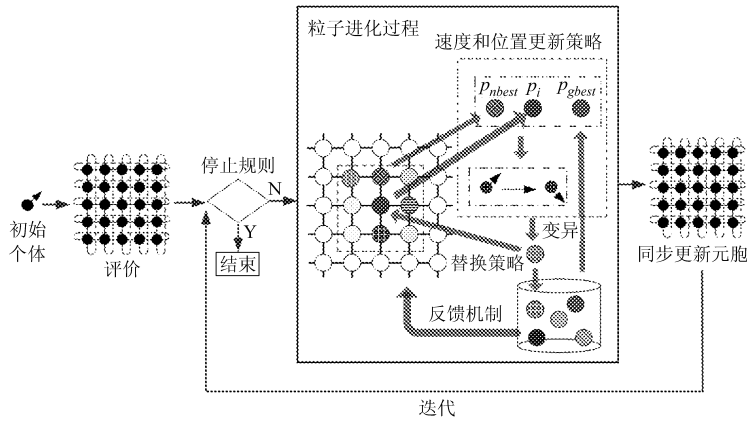


图 3 元胞多目标粒子群算法示意图

Fig. 3 Diagram of cellular multi-objective particle swarm optimization

(1) 对粒子群体进行随机初始化, 将粒子分布于二维网格结构中, 并计算各粒子对应的目标向量, 将其中的非劣解存入外部文档中。

(2) 评估每个粒子的邻居粒子, 确定粒子的初始邻居结构最优值 p_{nbest} 和种群最优值 p_{gbest} 。

(3) 根据新粒子飞行策略, 改变粒子的速度和位置。

(4) 生成的粒子群进行一定概率的扰动, 采用多项式变异算子和非均质性变异算子。每个算子分别对 1/2 个种群进行变异操作。

(5) 对新生历代粒子群进行评估, 替换操作, 将其中的非劣解存入外部文档中, 并更新 p_{nbest} 和 p_{gbest} 。

(6) $t = t + 1$, 如果满足条件则终止, 否则转到步骤(2)。

3 仿真实验及结果

为了验证算法 CMOPSO 的性能, 选取 3 个无约束的多目标测试函数 ZDT1、ZDT2 和 ZDT3 和 2 个带有约束的多目标测试函数 Srinivas 和 Tanaka, 分别将 CMOPSO、NSGAII、SPEA2、PAES 及 OMOPSO 对上述测试函数进行计算, 并分析在相同的评价体系下 5 种算法的性能。

3.1 算法性能评价指标

这里采用 Epsilon^[16]、分布指标^[17]和超体积^[18] 3 个性能指标进行度量。

(1) Epsilon (ε)

假设 Pareto 前端集合为 A , 那么 Epsilon 指标是移除 A 中每个个体所需要的最小距离的一个尺度。其具体形式是: 假设 $Z^1 = (Z_1^1, Z_2^1, \dots, Z_n^1)$, $Z^2 = (Z_1^2, Z_2^2, \dots, Z_n^2)$, 其中 n 是问题的目标维数, 即

$$\varepsilon(A) = \inf \{ \varepsilon \in R \mid \forall z^2 \in PF, \exists z^1 \in A : z^1 <_{\varepsilon} z^2 \} \quad (11)$$

其中, 仅当 $\forall 1 \leq i \leq n : z_i^1 < \varepsilon + z_i^2$, 则 $z^1 <_{\varepsilon} z^2$ 成立。

(2) 分布指标 (Spread)

Deb 提出的分布指标是衡量所得的 Pareto 前端解集分布情况。其计算公式为

$$\Delta = \frac{d_f + d_l + \sum_{i=1}^{n-1} |d_i - \bar{d}|}{d_f + d_l + (n-1)\bar{d}} \quad (12)$$

式中 d_i ——所得 Pareto 前端上每两个连续解点的欧氏距离

\bar{d} ——欧氏距离的平均距离

d_f, d_l ——所得 Pareto 前端的边界点与 Pareto 最优边界点的欧氏距离

n ——所得 Pareto 前端个体的数目

对于分布均匀的解来说, 该指标取零。因此, 该指标的值越小, 表明分布程度越均匀。

(3) 超体积 (Hypervolume)

超体积是用来计算获得的 Pareto 解集个体在目标域所覆盖的体积。其计算公式为

$$H_v = \text{volume} \left(\bigcup_{i=1}^{|Q|} v_i \right) \quad (13)$$

式中 Q ——获得的 Pareto 前端的个数

对于这个 Pareto 前端中的每一个体 i, v_i 是由参考点 $w = (0, 0, \dots, 0)$ 和成员 i 所形成的超体积, 此指标越大表明所得的 Pareto 解越能广泛地覆盖在其前端上。

3.2 测试函数计算结果及分析

NSGA-II、PESA2 及 PAES 均采用实数制进行编码, 模拟二进制^[19]交叉算子和多项式变异。OMOPSO 和 CMOPSO 中 w, r_1, r_2 参数为 $(0, 1)$ 之间的随机数, 其中 OMOPSO 中 c_1 和 c_2 的取值范围为 $[1.5, 2.0]$, 其变异算子采用一致性变异和非一致性变异, 两种变异算子作用于 2/3 个种群; CMOPSO 中 $c_{1\min}$ 和 $c_{2\min}$ 为 1.5, $c_{1\max}$ 和 $c_{2\max}$ 为 3.0, 变异算子采用多项式变异和非一致性变异, 分别作用于 1/2 个

种群。种群规模均为 100,最大进化代数为 250,交叉概率为 0.9,变异概率为 $1/l_{en}$, l_{en} 为变量维数。这 5 种算法分别对每个测试函数进行 30 次独立运行计算。

为了比较直观地展示 CMOPSO 算法的优势,随机选取 ZDT2 和 Tanaka 基准测试问题,作上述 5 种算法计算 ZDT2 和 Tanaka 时获得的 Pareto 前端的分布图,如图 4 和 5 所示。由图 4 可知,计算 ZDT2 时

获得的 Pareto 前端与真实前端的对比图,4 种算法均能找到 Pareto 前端,但 CMOPSO 得到的 Pareto 前端较其他 4 种算法的分布更加均匀。由图 5 可知,CMOPSO 所求的 Pareto 前端解集,在分布性和收敛性方面均比 NSGA-II、SPEA2、PAES 和 OMOPSO 好。同时从所得解的覆盖范围来看 CMOPSO 比其他 4 种算法要好,NSGA-II 和 SPEA2 次之,紧接着是 OMOPSO,最差的是 PAES。

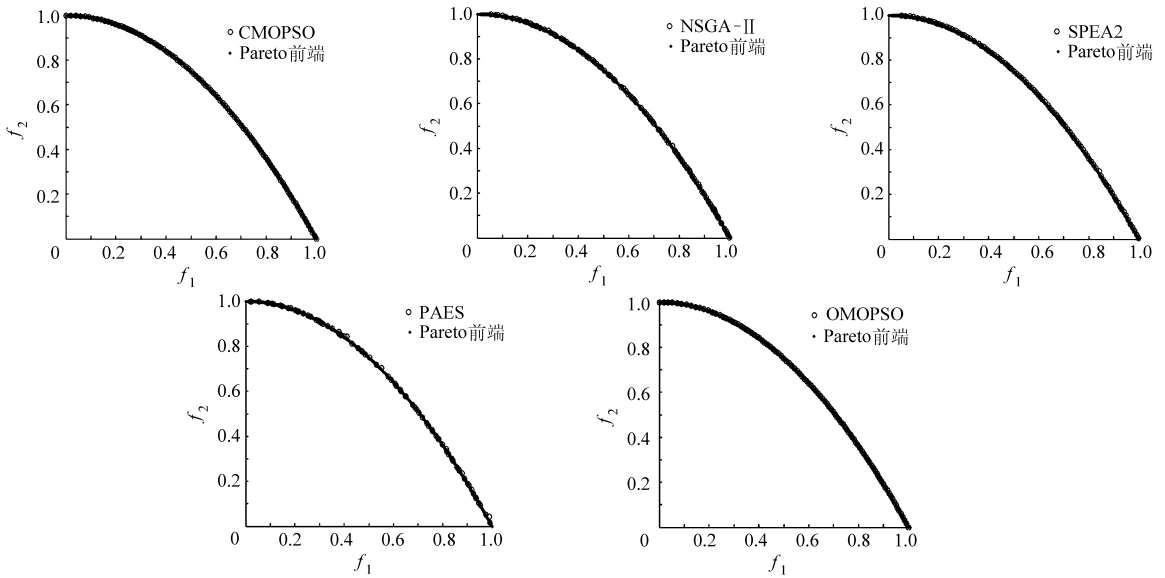


图 4 算法在 ZDT2 上仿真对比

Fig. 4 Comparison of algorithms' simulation in ZDT2

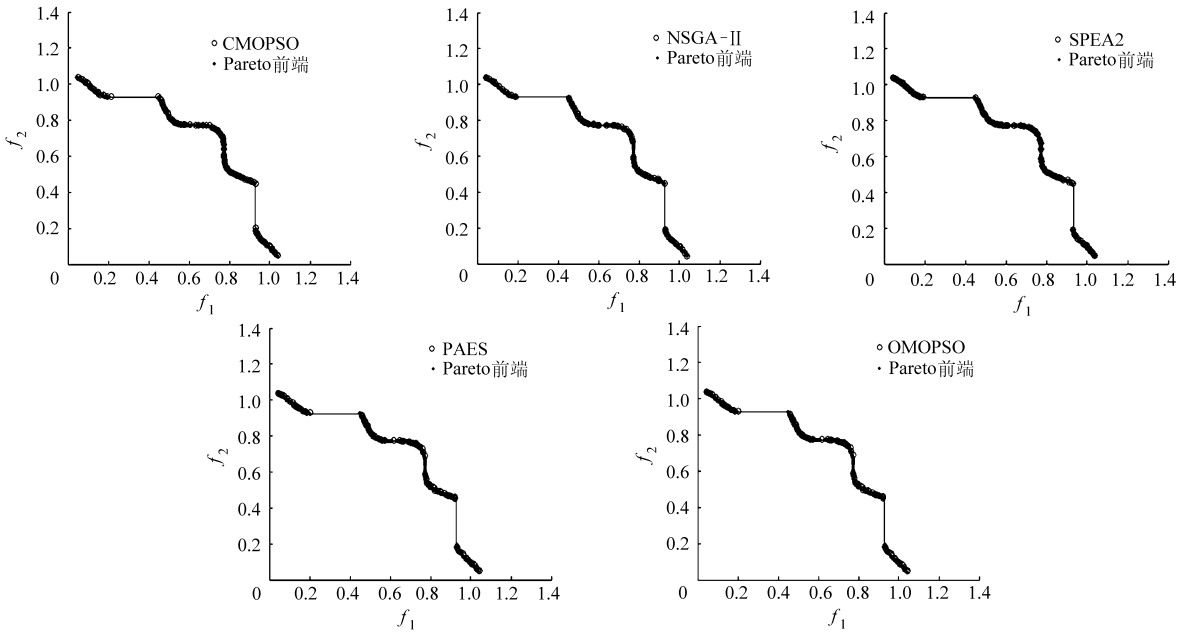


图 5 算法在 Tanaka 上仿真对比

Fig. 5 Comparison of algorithms' simulation in Tanaka

图 6 和 7 为 5 种算法在求解 ZDT2 和 Tanaka 时 30 次独立运行实验盒装图,其中矩形条代表指标的四分位差,数学意义是指标的分布区域,矩形条越短就意味着数据越集中,也就越好。矩形条中的横线

代表数据的中位数,对于指标 HV,中位数越大越好,图 6 中 CMOPSO 中位数最大,其 HV 指标最好。对于指标 Epsilon,中位数越小越好,其中 CMOPSO 中位数最小,说明 Epsilon 指标最好。对于指标

Spread, 中位数越小越好, 其中 CMOPSO 中位数最小, 说明 Spread 指标最好。由图 7 所示, 在解决 Tanaka 问题

时, CMOPSO 算法的 HV 指标相对于 SPEA2 差一点, 对于 Epsilon 和 Spread 指标都是最好的。

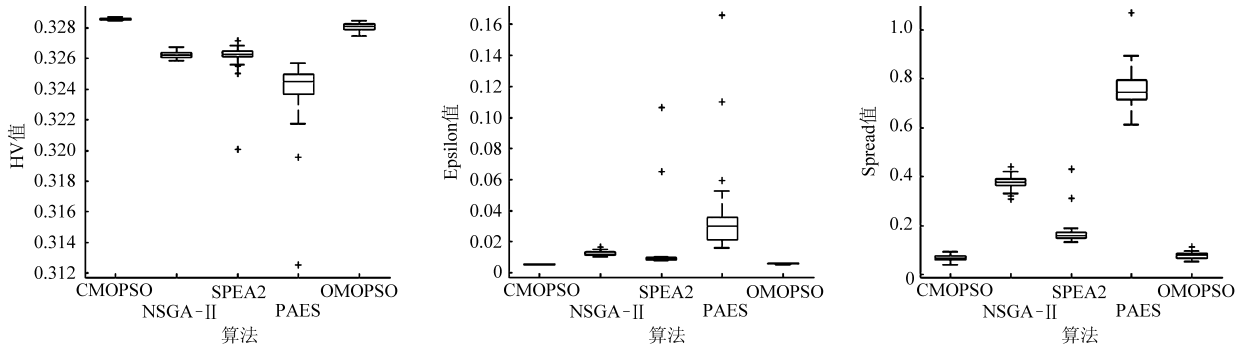


图 6 算法对 ZDT2 测试函数的统计盒分布图

Fig. 6 Comparison of algorithms' simulation in ZDT2 by box diagram

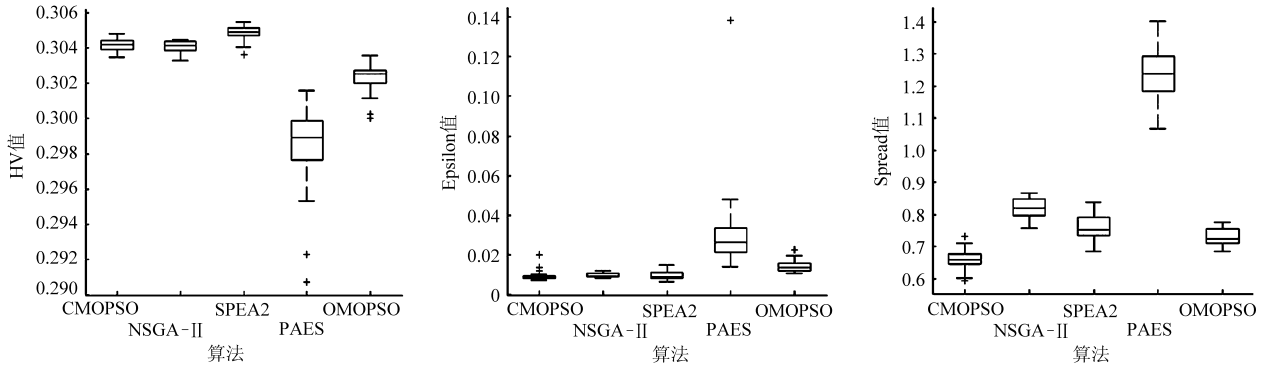


图 7 算法对 Tanaka 测试函数的统计盒分布

Fig. 7 Comparison of algorithms' simulation in Tanaka by box diagram

5 种算法从定量的角度分析它们的性能如表 1 ~ 3 所示, 其中括号外的数据为标准差, 括号内的数据为方差。

分析表 1 HV 指标值, CMOPSO 得到的结果比其他算法明显较好。针对这一指标, CMOPSO 在解 5 个函数时有 4 个函数解为最佳, 而 SPEA2 仅有 1 个最佳值。NSGA - II、PAES 和 OMOPSO 未获得。

说明 CMOPSO 算法的 Pareto 前端覆盖性较好。

Epsilon 指标, 由表 2 可看出, 对于所选用的基准测试函数, CMOPSO 在 5 个多目标问题中有 4 个达到了最优值, SPEA2 有 1 个最佳值, 其他 3 种算法均未获得。根据这些结果不难发现 CMOPSO 在解决高维多目标函数时的收敛效果整体上要优于其他 4 种算法。

表 1 HV 标准平均值及方差

Tab.1 Standard mean and variance of HV

	CMOPSO	NSGA - II	SPEA2	PAES	OMOPSO
ZDT1	$6.62 \times 10^{-1} (1.2 \times 10^{-4})$	$6.59 \times 10^{-1} (3.6 \times 10^{-4})$	$6.60 \times 10^{-1} (2.4 \times 10^{-4})$	$6.53 \times 10^{-1} (1.6 \times 10^{-2})$	$6.61 \times 10^{-1} (3.8 \times 10^{-4})$
ZDT2	$3.29 \times 10^{-1} (7.2 \times 10^{-5})$	$3.26 \times 10^{-1} (2.3 \times 10^{-4})$	$3.26 \times 10^{-1} (1.2 \times 10^{-3})$	$3.24 \times 10^{-1} (2.5 \times 10^{-3})$	$3.28 \times 10^{-1} (2.2 \times 10^{-4})$
ZDT3	$5.15 \times 10^{-1} (5.0 \times 10^{-4})$	$5.15 \times 10^{-1} (1.6 \times 10^{-4})$	$5.14 \times 10^{-1} (6.5 \times 10^{-4})$	$4.90 \times 10^{-1} (3.2 \times 10^{-2})$	$5.14 \times 10^{-1} (7.6 \times 10^{-4})$
Tanaka	$3.04 \times 10^{-1} (3.3 \times 10^{-4})$	$3.04 \times 10^{-1} (3.4 \times 10^{-4})$	$3.05 \times 10^{-1} (4.2 \times 10^{-4})$	$2.98 \times 10^{-1} (2.3 \times 10^{-3})$	$3.02 \times 10^{-1} (7.8 \times 10^{-4})$
Srinivas	$5.35 \times 10^{-1} (4.6 \times 10^{-4})$	$5.32 \times 10^{-1} (3.6 \times 10^{-4})$	$5.34 \times 10^{-1} (1.3 \times 10^{-4})$	$5.30 \times 10^{-1} (1.1 \times 10^{-3})$	$5.35 \times 10^{-1} (6.8 \times 10^{-5})$

表 2 Epsilon 标准平均值及方差

Tab.2 Standard mean and variance of Epsilon

	CMOPSO	NSGA - II	SPEA2	PAES	OMOPSO
ZDT1	$7.61 \times 10^{-2} (1.3 \times 10^{-2})$	$3.67 \times 10^{-1} (3.6 \times 10^{-2})$	$1.51 \times 10^{-1} (1.6 \times 10^{-2})$	$7.32 \times 10^{-1} (6.0 \times 10^{-2})$	$8.22 \times 10^{-2} (1.2 \times 10^{-2})$
ZDT2	$6.85 \times 10^{-2} (1.2 \times 10^{-2})$	$3.75 \times 10^{-1} (2.9 \times 10^{-2})$	$1.73 \times 10^{-1} (5.7 \times 10^{-2})$	$7.60 \times 10^{-1} (8.6 \times 10^{-2})$	$7.74 \times 10^{-2} (1.3 \times 10^{-2})$
ZDT3	$7.13 \times 10^{-1} (1.8 \times 10^{-2})$	$7.51 \times 10^{-1} (1.9 \times 10^{-2})$	$7.10 \times 10^{-1} (6.3 \times 10^{-3})$	$1.08 (8.6 \times 10^{-2})$	$7.14 \times 10^{-1} (1.1 \times 10^{-2})$
Tanaka	$6.59 \times 10^{-1} (2.8 \times 10^{-2})$	$8.18 \times 10^{-1} (2.8 \times 10^{-2})$	$7.61 \times 10^{-1} (3.8 \times 10^{-2})$	$1.24 (8.0 \times 10^{-2})$	$7.31 \times 10^{-1} (2.7 \times 10^{-2})$
Srinivas	$6.74 \times 10^{-2} (1.0 \times 10^{-2})$	$4.04 \times 10^{-1} (2.8 \times 10^{-2})$	$1.70 \times 10^{-1} (1.0 \times 10^{-2})$	$6.51 \times 10^{-1} (4.9 \times 10^{-2})$	$7.83 \times 10^{-2} (7.8 \times 10^{-3})$

表3 Spread 标准平均值及方差
Tab.3 Standard mean and variance of Spread

	CMOPSO	NSGA-II	SPEA2	PAES	OMOPSO
ZDT1	$5.77 \times 10^{-3} (2.5 \times 10^{-4})$	$1.39 \times 10^{-2} (2.7 \times 10^{-3})$	$8.99 \times 10^{-3} (9.2 \times 10^{-4})$	$4.01 \times 10^{-2} (6.3 \times 10^{-2})$	$6.41 \times 10^{-3} (3.8 \times 10^{-4})$
ZDT2	$5.59 \times 10^{-3} (1.5 \times 10^{-4})$	$1.24 \times 10^{-2} (1.4 \times 10^{-3})$	$1.42 \times 10^{-2} (2.0 \times 10^{-2})$	$3.64 \times 10^{-2} (3.0 \times 10^{-2})$	$6.00 \times 10^{-3} (2.5 \times 10^{-4})$
ZDT3	$5.96 \times 10^{-3} (7.3 \times 10^{-4})$	$8.78 \times 10^{-3} (1.4 \times 10^{-3})$	$2.01 \times 10^{-2} (5.5 \times 10^{-2})$	$2.95 \times 10^{-1} (2.9 \times 10^{-1})$	$7.38 \times 10^{-3} (1.8 \times 10^{-3})$
Tanaka	$9.33 \times 10^{-3} (2.4 \times 10^{-3})$	$9.72 \times 10^{-3} (1.0 \times 10^{-3})$	$9.59 \times 10^{-3} (1.8 \times 10^{-3})$	$3.07 \times 10^{-2} (2.1 \times 10^{-2})$	$1.45 \times 10^{-2} (3.1 \times 10^{-3})$
Srinivas	$1.42 (1.1 \times 10^{-1})$	$3.47 (6.2 \times 10^{-1})$	$1.84 (1.1 \times 10^{-1})$	$4.88 (1.7)$	$1.53 (1.8 \times 10^{-1})$

就分布性 Spread 指标而言,表3中的结果显示CMOPSO所得的非支配解,在沿 Pareto 前端分布的多样性方面占有绝对优势,其中5个函数都获得最佳。因此说明CMOPSO算法,在进化过程中多样性较好,Pareto 前端分布更加均匀。

CMOPSO 算法之所以表现出良好的收敛性和多样性,其原因在于它将元胞自动机和 PSO 的优势结合,并很好地控制了粒子的飞行速度。从实验数据可知,CMOPSO 的确提高了解集的收敛性能,有效地保持了种群的多样性,增大了解集的覆盖范围。因此,通过以上结果分析可知,CMOPSO 在解决多约束、多变量、高维的多目标问题时,相比其他4种算法,具有更大优势。

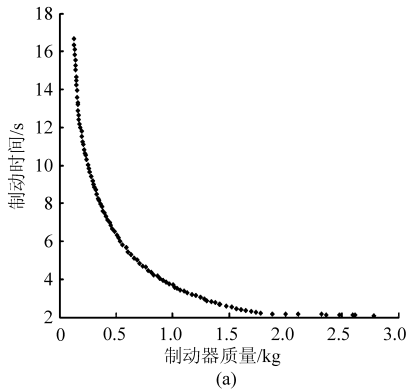
4 盘式制动器的优化设计

盘式制动器的优化设计是实际工程中非常经典的一种结构设计模型^[20-21],该问题是一个混合约束多目标问题。此问题优化目标为制动器的质量最小 $f_1(x)$ 和制动时间最短 $f_2(x)$ 。该问题的设计变量分别为:制动鼓的内半径 x_1 , 制动鼓的外半径 x_2 , 制动力为 x_3 和摩擦曲面个数 x_4 。该问题的数学模型如下:

目标函数为

$$\min f_1(x) = 4.9 \times 10^{-5} (x_2^2 - x_1^2) (x_4 - 1)$$

$$\min f_2(x) = \frac{9.82 \times 10^6 (x_2^2 - x_1^2)}{x_3 x_4 (x_2^3 - x_1^3)}$$



s. t.

$$g_1(x) = (x_2 - x_1) - 20 \geq 0$$

$$g_2(x) = 30 - 2.5(x_4 + 1) \geq 0$$

$$g_3(x) = 0.4 - x_3 / [3.14(x_2^2 - x_1^2)] \geq 0$$

$$g_4(x) = 1 - \frac{2.22 \times 10^{-3} x_3 x_4 (x_2^3 - x_1^3)}{(x_2^2 - x_1^2)^2} \geq 0$$

$$g_5(x) = \frac{2.66 \times 10^{-2} x_3 x_4 (x_2^3 - x_1^3)}{x_2^2 - x_1^2} - 900 \geq 0$$

设计变量的取值范围: $55 \leq x_1 \leq 80, 75 \leq x_2 \leq 110, 1000 \leq x_3 \leq 3000, 2 \leq x_4 \leq 20$ 。

为了比较算法的性能,除了CMOPSO之外,同时也采用当前最流行的多目标遗传算法NSGA-II对盘式制动器模型进行优化计算。算法的参数设置如下:种群规模 $M = 100$, CMOPSO 中文档容量 $N = 100$, 反馈数目 $C = 25$, 变异概率 $p_m = 1/n_{var}$, 其中 n_{var} 为决策变量的个数。 w, r_1, r_2 参数为 $(0, 1)$ 之间的随机数, c_{1min} 和 c_{2min} 为 $1.5, c_{1max}$ 和 c_{2max} 为 3.0 。设置最大进化代数 $g_{enmax} = 250$, 将两种算法在相同的环境下运行,对盘式制动器优化设计模型进行求解,独立运行10次,取其中优化结果最优的一次进行对比。

如图8所示,CMOPSO 算法获得 Pareto 解集更均匀,且得到的端点值更广。由表4可得,CMOPSO 算法获得解集的范围为 $(0.1274, 16.6500) \sim (2.7930, 2.0710)$; NSGA-II 解集的范围为 $(0.1366, 15.9300) \sim (2.7930, 2.0730)$; 文献[18]中提

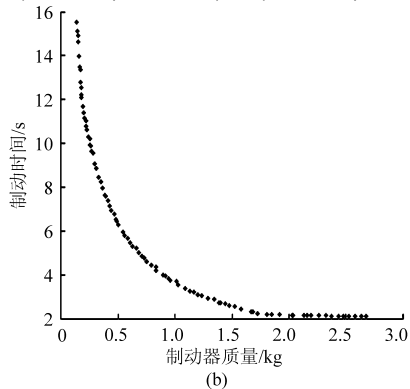


图8 盘式制动器的优化设计结果比较

Fig.8 Comparison of optimization results of disc brake

(a) CMOPSO (b) NSGA-II

表 4 盘式制动器优化设计问题的结果对比

Tab. 4 Comparison of optimization results of disc brake problem

算法	$f_1(x)$	$f_2(x)$
NSGA-II	0.133 6	15.930 0
	2.793 0	2.073 0
$pa \in -ODEMO^{[18]}$	0.127 4	16.654 9
	2.868 4	2.090 6
CMOPSO	0.127 4	16.650 0
	2.793 0	2.071 0

出的 $pa \in -ODEMO$ 算法获得解集的范围为 $(0.127 4, 16.654 9) \sim (2.868 4, 2.090 6)$ 。由此可见 CMOPSO 算法相对于 NSGA-II 和 $pa \in -ODEMO$ 算法,得到的结果比另外两种算法求到的最小结果要小,其端点数据都可以支配另外两种算法。说明 CMOPSO 算法全局搜索性能优异,且算法稳定性较好,收敛精度较高。在求解盘式制动器优化设计问题时,能够为

决策者提供更为广泛的方法选择。

5 结束语

针对现有的多目标粒子群算法在求解复杂的多目标问题时,算法的收敛性和分布性差等问题,本文提出一种元胞多目标粒子群算法(CMOPSO)。该算法首先分析多目标粒子群算法的基本理论,将元胞自动机融入粒子群算法中,研究粒子之间的相互关系和信息交流,尤其将粒子自身最优位置更替为邻居结构中最优位置。为了避免粒子群易陷入局部最优解,提出一种控制粒子飞行速度策略。通过 2 个含有约束的基准测试函数实验说明,与其他 4 种算法相比,CMOPSO 算法在覆盖性指标方面有优势,尤其在收敛性指标、分布性指标上表现更优异,是求解高维约束多目标算法的有效方法。将其求解盘式制动器优化设计问题,相对于其他算法获得 Pareto 解集分布更为均匀,且解集的分布更为广泛。

参 考 文 献

- 吕振肃,侯志荣. 自适应变异的粒子群优化算法[J]. 电子学报, 2004, 32(3): 416 ~ 420.
Lü Zhensu, Hou Zhirong. Particle swarm optimization with adaptive mutation [J]. Chinese Journal of Electronics, 2004, 32(3): 416 ~ 420. (in Chinese)
- 张选平,杜玉平,秦国强,等. 一种动态改变惯性权的自适应粒子群算法[J]. 西安交通大学学报, 2005, 39(10): 1 039 ~ 1 042.
Zhang Xuanping, Du Yuping, Qin Guoqiang, et al. Adaptive particle swarm algorithm with dynamically changing inertia weight [J]. Journal of Xi'an Jiaotong University, 2005, 39(10): 1 039 ~ 1 042. (in Chinese)
- 刘道华,原思聪,张锦华,等. 粒子群参数自适应调整的优化设计[J]. 农业机械学报, 2008, 39(9): 134 ~ 138.
Liu Daohua, Yuan Sicong, Zhang Jinhua, et al. Optimization design of particle swarm with self-adaptive parameter adjusting [J]. Transactions of the Chinese Society for Agricultural Machinery, 2008, 39(9): 134 ~ 138. (in Chinese)
- 杨雪榕,梁加红,陈凌,等. 多邻域改进粒子群算法[J]. 系统工程与电子技术, 2010, 32(11): 2 453 ~ 2 458.
Yang Xuerong, Liang Jiahong, Chen Ling, et al. Multi-neighborhood improved particle swarm optimization algorithm [J]. Systems Engineering and Electronics, 2010, 32(11): 2 453 ~ 2 458. (in Chinese)
- Zhan Zhihui, Zhang Jun, Li Yun, et al. Orthogonal learning particle swarm optimization [J]. IEEE Transactions on Evolutionary Computation, 2011, 15(6): 832 ~ 847.
- 王丽,刘玉树,徐远清. 基于在线归档技术的多目标粒子群算法[J]. 北京理工大学学报, 2006, 26(10): 883 ~ 886.
Wang Li, Liu Yushu, Xu Yuanqing. Online elite archiving in multi-objective particle swarm optimization [J]. Journal of Beijing Institute of Technology, 2006, 26(10): 883 ~ 886. (in Chinese)
- Li X. A non-dominated sorting particle swarm optimizer for multi-objective optimization [C] // Genetic and Evolutionary Computation—GECCO 2003. Springer Berlin Heidelberg, 2003: 37 ~ 48.
- Alvarez Benitez J, Everson R, Fieldsend J. A MOPSO algorithm based exclusively on Pareto dominance concepts [J]. Lecture Notes in Computer Science, 2005, 3 410: 459 ~ 473.
- 刘衍民,赵庆祯,牛奔,等. 基于 ε 占优的自适应多目标粒子群算法[J]. 控制与决策, 2011, 21(1): 89 ~ 95.
Liu Yanmin, Zhao Qingzhen, Niu Ben, et al. Adaptive multi-objective particle swarm optimizer based on ε dominance [J]. Control and Decision, 2011, 21(1): 89 ~ 95. (in Chinese)
- 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)
- Eberhart R, Kennedy J. A new optimizer using particle swarm theory [C] // Proceedings of the 6th Inter Symposium on Micro Machine and Human Science. Piscataway, NJ: IEEE Service Center, 1995: 39 ~ 43.
- 雷德明,严新平. 多目标智能优化算法及其应用[M]. 北京: 科学出版社, 2009.
- Alba Enrique, Bernabé Dorronsoro. Cellular genetic algorithms [M]. New York: Springer-Verlag New York Inc., 2008.

- [J]. IEEE Transactions on Industrial Electronics, 2003, 50(5): 944 ~ 952.
- 12 Canudas de Wit C, Olsson H, Astrom K J, et al. A new model for control of systems with friction [J]. IEEE Transactions on Automatic Control, 1995, 40(3): 419 ~ 425.
- 13 Amthor A, Zschaeck S, Ament C. High precision position control using an adaptive friction compensation approach [J]. IEEE Transactions on Automatic Control, 2010, 55(1): 274 ~ 278.
- 14 San P P, Ren B, Ge S S, et al. Adaptive neural network control of hard disk drives with hysteresis friction nonlinearity [J]. IEEE Transactions on Control Systems Technology, 2011, 19(2): 351 ~ 358.
- 15 Davis T A, Shin Y C, Yao B. Observer-based adaptive robust control of friction stir welding axial force [J]. IEEE/ASME Transactions on Mechatronics, 2011, 16(6): 1 032 ~ 1 039.
- 16 刘丽兰, 刘宏昭, 原大宁, 等. 考虑粘滑摩擦的时变振动系统模态参数辨识[J]. 农业机械学报, 2010, 41(1): 177 ~ 181.
- Liu Lilan, Liu Hongzhao, Yuan Daning, et al. Modal parameter identification of time varying vibration systems subjected to stick slip friction[J]. Transactions of the Chinese Society for Agricultural Machinery, 2010, 41(1): 177 ~ 181. (in Chinese)

(上接第 287 页)

- 14 Clerc M. The swarm and the queen: towards a deterministic and adaptive particle swarm optimization [C] // Evolutionary Computation, CEC 99, 1999.
- 15 Nebro A J, Durillo J J, García-Nieto J, et al. SMPSO: A new PSO-based metaheuristic for multiobjective optimization [C] // IEEE Symposium on Computational Intelligence in Multi-criteria Decision-making, Nashville, TN, 2009: 66 ~ 73.
- 16 Durillo Juan J, Nebro Antonio J. A java framework for multi-objective optimization [J]. Advances in Engineering Software, 2011, 42(10): 760 ~ 771.
- 17 Deb K, Pratap A, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA - II [J]. IEEE Transactions on, Evolutionary Computation, 2002, 6(2): 182 ~ 197.
- 18 Zitzler E, Thiele L. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach [J]. IEEE Transactions on Evolutionary Computation, 1999, 3(4): 257 ~ 271.
- 19 Deb K, Agrawal R B. Simulated binary crossover for continuous search space [J]. Complex Systems, 1995, 9(2): 115 ~ 148.
- 20 Gong Wenyin, Cai Zhihua, Zhu Li. An efficient multiobjective differential evolution algorithm for engineering design [J]. Structural and Multidisciplinary Optimization, 2009, 38(2): 137 ~ 157.
- 21 Tapabrata Ray, Liew K M. A swarm metaphor for multiobjective design optimization [J]. Engineering Optimization, 2002, 34(2): 141 ~ 153.

(上接第 293 页)

- 14 崔国华, 王国强, 赵春江, 等. 空间转动 3 自由度并联微调机构设计与运动学分析[J]. 农业机械学报, 2008, 39(9): 144 ~ 148.
- Cui Guohua, Wang Guoqiang, Zhao Chunjiang, et al. Mechanism design and kinematic analysis on a three degree-of-freedom spatial manipulator [J]. Transactions of the Chinese Society for Agricultural Machinery, 2008, 39(9): 144 ~ 148. (in Chinese)
- 15 崔国华, 李权才, 张艳伟, 等. 空间转动 4 - SPS - 1 - S 型并联机构动力学性能指标分析[J]. 农业机械学报, 2010, 41(7): 214 ~ 218.
- Cui Guohua, Li Quancai, Zhang Yanwei, et al. Dynamic performance indices analysis of 4 - SPS - 1 - S spatial rotation parallel manipulator [J]. Transactions of the Chinese Society for Agricultural Machinery, 2010, 41(7): 214 ~ 218. (in Chinese)
- 16 张艳伟, 韦斌, 王南, 等. 空间转动 3 - SPS - S 并联机构运动学性能分析[J]. 农业机械学报, 2012, 43(4): 212 ~ 215.
- Zhang Yanwei, Wei Bin, Wang Nan, et al. Kinematic performance analysis of 3 - SPS - S spatial rotation parallel mechanism [J]. Transactions of the Chinese Society for Agricultural Machinery, 2012, 43(4): 212 ~ 215. (in Chinese)
- 17 王德伦, 戴建生. 变胞机构及其综合的理论基础[J]. 机械工程学报, 2007, 43(8): 32 ~ 42.
- Wang Delun, Dai Jiansheng. Theoretical foundation of metamorphic mechanism and its synthesis [J]. Chinese Journal of Mechanical Engineering, 2007, 43(8): 32 ~ 42. (in Chinese)
- 18 刘辛军. 并联机器人机构尺寸与性能关系分析及其设计理论[D]. 秦皇岛: 燕山大学, 1999.
- Liu Xinjun. The relationships between the performance criteria and link lengths of the parallel manipulators and their design theory [D]. Qinghuangdao: Yanshan University, 1999. (in Chinese)