可行性规则动态调整的多目标粒子群算法*

赵乃刚[1]，李 勇[1]，王振荣[2]
(1. 山西大同大学 数学与计算机科学学院，山西 大同 037009；2. 山西省大同市人民政府信息化中心，山西 大同 037009)

摘 要：针对多目标粒子群算法存在的问题，提出了一种可行性规则动态调整的多目标粒子群算法。在算法中，根据粒子之间的相似度值动态非线性地更新算法的惯性权重，使得算法可以高效地平衡全局和局部搜索之间的矛盾；采用动态加权法解决随机性抽取群体最优粒子的缺陷，保证了种群的多样性；并且动态改变可行性规则的阈值，使得算法可以有效地利用某些不可行解包含的有效信息，提高了算法收敛到 Pareto 前沿的能力。最后，与其他四种多目标算法的实验比较验证了新算法的性能更好。

关键词：粒子群算法；多目标优化；惯性权重；动态加权法；可行性规则
中图分类号：TP301.6 **文献标志码**：A **文章编号**：1001-3695(2018)05-1304-03
doi:10.3969/j.issn.1001-3695.2018.05.005

Multi-objective particle swarm algorithm based on dynamic adjustment of feasibility rule

Zhao Naigang[1], Li Yong[1], Wang Zhenrong[2]
(1. *College of Mathematics & Computer Science, Shanxi Datong University, Datong Shanxi* 037009, *China*; 2. *Information Center of Datong People's Government of Shanxi Province, Datong Shanxi* 037009, *China*)

Abstract: Aiming at the problems of multi-objective particle swarm algorithm, this paper developed a multi-objective particle swarm algorithm based on dynamic adjustment of feasibility rule. This algorithm dynamically non-linear updated the inertia weight according to the similarity value between particle and best particle. In order to solve the defects of random selection of the best particle, this paper proposed dynamic update method of inertia weight and it could ensure the diversity of the population. It changed the threshold of feasibility rules dynamically so that new algorithm could take advantage of some valid information of some infeasible solutions and it could improve the ability of converging to the Pareto frontier. Experiment comparison with other four algorithms shows better performance of the new algorithm.

Key words: particle swarm algorithm; multi-objective optimization; inertia weight; dynamic weighting method; feasibility rule

0 引言

现实生活中大量的实际问题可以转换为多目标优化[1,2]问题的求解。粒子群算法由于其结构简单、收敛速度快已经被广泛地用于求解此类问题。在求解复杂的多目标优化问题的过程中，多目标粒子群算法存在易陷入局部最优等问题。针对此类问题，许多学者对其进行了深入研究并对算法提出了多种改进方法，然而目前大量的改进方法往往都只是针对某个或某一些问题的缺陷对算法进行优化。当面对新的优化问题时，必须通过反复的实验测试并对原算法进行适当修改才能够应用到新的待优化问题中。虽然可以取得一些理想的效果，但是搜索到的结果却并不一定是最合适的。如此，研究者希望获得一个完整可靠的理论体系支撑多目标粒子群算法的设计。在设计多目标粒子群算法时，确定种群中的全局最优位置成为了设计算法的一个难点。在求解多目标优化问题中，只能得到一组非劣解，即 Pareto 解[3]。如果待优化问题的目标函数增加，算法找出的非劣解也会迅速增加，这样将会使得粒子群算法的寻优性能急速下降。并且，粒子群算法快速收敛的性能也会导致群体多样性丧失，使得算法很容易陷入局部最优位置。因此求解多目标优化问题时，保证群体的多样性是十分必要的。近年来，很多研究者提出了不同的基于 Pareto 解的多目标粒子群算法并针对不同的优化问题对算法进行了讨论。Lei Deming 在文献[4]中提出了一种基于 Pareto 存档的多目标粒子群优化算法并将它用于车间作业多目标调度问题的求解。文献[5]提出了一种基于 Pareto 最优的新的多目标粒子群算法，并与其他改进算法进行了比较，实验结果展示它较其他算法性能提高明显，尤其是针对许多复杂的多目标优化问题更有效。文献[6]提出了一种混杂多目标粒子群算法，在算法中将群体的启发式局部搜索和全局搜索进行有机结合，并结合模糊全局最优的概念防止算法早熟收敛，保持种群的多样性。

为了改善多目标粒子群算法的收敛精度，并同时提高非劣解的均匀性和多样性，本文基于对惯性权重的分析，引入了一种基于相似度值非线性调整的惯性权重更新方式，给出了阈值动态变化的可行性规则，并在算法中用了一种新的群体最优粒子选择方法，提出了一种可行性规则动态调整的多目标粒子群

收稿日期：2017-01-02；**修回日期**：2017-03-06 **基金项目**：国家自然科学基金资助项目(61672331)；山西省高等学校教学改革项目(2015090)；山西大同大学科学研究项目(2016K1)

作者简介：赵乃刚(1975-)，男，山西应县人，讲师，硕士研究生，主要研究方向为数据挖掘(2893966820@qq.com)；李勇(1974-)，男，辽宁大连人，博士，主要研究方向为智能算法及无线网络；王振荣(1988-)，男，山西应县人，硕士研究生，主要研究方向为计算机科学与技术.

算法。通过与其他四种算法在五个测试函数上的实验结果验证了算法的性能。

1　多目标优化相关概念及粒子群算法简介

1.1　多目标优化的数学描述和相关概念

多目标优化问题一般可以描述为

$$\min f(x)=(f_1(x),f_2(x),\cdots,f_m(x))$$
$$\text{s.t.}\begin{cases} g_i(x)\leqslant 0 & i=1,2,\cdots,p \\ h_j(x)=0 & j=1,2,\cdots,q \\ x\in \mathrm{R}^n \end{cases} \tag{1}$$

其中:$f(x)=(f_1(x),f_2(x),\cdots,f_m(x))$是待优化的目标函数,其包含了 m 个目标函数;$x\in \mathrm{R}^n$ 为 n 维决策变量;$g_i(x):\mathrm{R}^n\to \mathrm{R},(k=1,2,\cdots,p)$表示多目标优化问题中的第 i 个不等式约束条件;$h_j(x):\mathrm{R}^n\to \mathrm{R},(k=1,2,\cdots,q)$表示第 j 个等式约束条件。它们共同组成了决策变量的可行域。

1.2　粒子群优化算法

在粒子群优化算法中,首先随机地在搜索空间初始一组解,然后粒子在寻优过程中根据两个极值更新自身,一个是它的个体极值,另一个是全局极值。目前广泛使用的 PSO 算法的数学模型可简述为:设在 PSO 算法中粒子的搜索空间是 m 维的,算法的最大寻优次数为 T,第 i 个粒子 t 时刻时在空间中的飞行速度为 $v_i(t)=[v_{i1},v_{i2},\cdots,v_{im}]^{\mathrm{T}}$,其所处的空间位置为 $x_i(t)=[x_{i1},x_{i2},\cdots,x_{im}]^{\mathrm{T}}$,粒子相应的个体极值为 $p_i(t)=[p_{i1},p_{i2},\cdots,p_{im}]^{\mathrm{T}}$,它的全局极值为 $g_b=[g_1,g_2,\cdots,g_m]^{\mathrm{T}}$。PSO 的寻优过程主要由粒子的速度更新和位置更新两部分组成,更新方式为

$$v_i(t+1)=\omega v_i(t)+c_1r_1(p_i(t)-x_i(t))+c_2r_2(g_b-x_i(t)) \tag{2}$$
$$x_i(t+1)=x_i(t)+v_i(t+1) \tag{3}$$

其中:ω 为惯性权重;c_1、c_2 为学习因子,一般取 2;r_1、r_2 是均匀分布在[0,1]上的随机数。

2　改进的多目标粒子群算法

2.1　基于相似度值动态调整的惯性权重

在多目标粒子群算法中,惯性权重 ω 的取值对算法的局部精细寻优和全局搜索能力有很大影响,它的取值方法控制着算法两个搜索能力(全局和局部搜索)的平衡。研究表明,若惯性权重取值较大,则越有利于对搜索空间进行全局搜索;若惯性权重取值较小,则越有利于在当前局部区域进行精细寻优。为了改善算法的搜索能力,这里给出了粒子与粒子之间相似度的定义,并根据粒子与群体最优粒子之间的相似度值动态非线性的更新算法的惯性权重值。

定义　微粒群中粒子 i 和 j 的相似度 $s(i,j)$可以表述为

$$s(i,j)=\begin{cases} 1 & d(i,j)<d_{\min} \\ 1-\left[\dfrac{d(i,j)}{2}\right]^2 & d_{\min}\leqslant d(i,j)<d_{\max} \\ 0 & d(i,j)\geqslant d_{\max} \end{cases} \tag{4}$$

其中:$d(i,j)$是粒子 i 和 j 的欧氏距离。从式(4)中可以看出,当 $d(i,j)\to 0$ 时,$s(i,j)=1$;当 $d(i,j)\to 1$ 时,$s(i,j)=0$;对微粒群中任意的粒子 i 和 j,$s(i,j)\in[0,1]$。参数 $d_{\min}$ 和 $d_{\max}$ 取正常数,它们的大小需要本文根据可行域的大小进行适当的调节。从粒子相似度值的定义可以看出,如果粒子越相似,它们的相似度值也就越大;相反,如果粒子之间差别越大,则它们的相似度值也就越小。在算法的执行过程中,粒子会逐步收敛到全局最优位置,即所有粒子与群体最优位置的粒子的相似度值会越来越小,最后趋向于 0。

在粒子群算法的执行过程中,希望获得的全局最优解有很大概率出现在群体最优粒子的邻近空间内。如果想要获得的全局最优解恰好就在群体最优粒子的附近区域,那么对当前的群体最优粒子附近的所有粒子来说,速度更新式的后两部分已经趋向于 0,此时的速度更新式可近似简化为 $v(t+1)=\omega v(t)$。显然,当前邻域粒子的飞行速度仅仅由此时的惯性权重和前一时刻的速度来决定,对惯性权重的取值方式有很强的依赖性。如果此时 ω 的取值较大,则当前粒子极有可能会飞出群体最优粒子的邻域,而如果此时 ω 取合适的值,则可以使得算法在当前的小邻域内进行局部精细搜索,很可能获得真正的全局最优解。因此,此时那些与群体最优粒子性质越接近的粒子的惯性权重取值越小。根据上述分析,本文提出了一种基于粒子与群体最优粒子的相似度值的动态非线性调整的惯性权重更新方式。在本文改进的多目标粒子群算法中,惯性权重值不仅会在算法的执行过程中逐渐递减,而且它会根据粒子与当前群体最优粒子的相似度值的大小进行有效调节。惯性权重的更新方式为:设第 i 个粒子与群体最优粒子 g_b 之间的相似度值为 $s(i,g_b)$,当 $s(i,g_b)=0$ 时,说明粒子与群体最优粒子的差异最明显,粒子必须跳出当前的区域,故粒子下次迭代时的惯性权重应取最大值 $\omega_{\max}$;当 $s(i,g_b)=1$ 时,说明当前粒子与群体最优粒子的差异最不明显,该粒子需要在当前的小邻域内进行局部精细搜索,粒子下次迭代时的惯性权重应取最小值 $\omega_{\min}$;当 $s(i,g_b)\in(0,1)$时,t 时刻的惯性权重的取值需根据相似度值的计算大小进行如下更新:

$$\omega(t)=\omega_{\min}+[\omega_{\max}-\omega_{\min}-s(i,g_b)\times(\omega_{\max}-\omega_{\min})]\times\sqrt{\frac{T-t}{T}} \tag{5}$$

很显然,ω 的取值范围其实仍然为$[\omega_{\min},\omega_{\max}]$。因此,$\omega$ 的设置方式不仅可以确保算法收敛性,而且能够使得算法在保持较好全局寻优能力的基础上获得更好的局部寻优能力。

2.2　群体和个体最优粒子的选择

在设计多目标算法解决多目标优化问题时,如何在种群中选择合适的群体最优粒子成为了算法成功与否的一个重点。与单目标粒子群算法不同,多目标粒子群算法中的群体最优解往往是不存在的,只能得到一组非劣解,即 Pareto 解;然后采用随机抽取的方式在非劣解集中随机选择一个粒子作为下一次迭代的群体最优粒子,这种随机抽取方法使得处于最前沿的非支配粒子和处于最低沿的非支配粒子均有相同的概率被选择,而这种方式很可能会使得种群的多样性丧失,不利于保证种群多样性。在本文中,采用文献[7]中的动态加权法解决随机性抽取群体最优粒子的缺陷,即根据式(6)动态地更新非劣解集中每个粒子的适应度

$$\text{fitness}(i)=1/\sum_{j=1}^{m}a_jf_j(x_i) \tag{6}$$

其中:$a_j(j=1,2,\cdots,m)$是分布在区间(0,1)的随机数字,且满足$\sum_{j=1}^{m}a_j=1$。当种群中每个粒子的适应度计算完成后,选择适应度值最大的粒子作为群体最优粒子。

对于个体最优粒子,本文采用一般的选择方式,即根据粒子间的支配与非支配的关系选择。如果当前粒子支配历史个体最优粒子,则更新个体最优粒子,将当前粒子设置为个体最

优粒子;如果历史个体最优粒子支配当前粒子,则个体最优粒子不更新;如果当前粒子和历史个体最优粒子不构成支配关系,粒子的历史个体最优位置和当前位置不变。

2.3 阈值动态变化的可行解规则

在处理多目标问题时,当解的分布用 Pareto 表示时,本文需要考虑粒子的可行性及粒子相互之间的支配关系。一般的可行性规则可简述为:当可行解优于不可行解时,一部分不可行解被删除,这种处理方式不利于不可行解的保存及后续使用。若处理不可行解时,将初始不可行阈值扩大,使得不可行阈值动态地下降,逐渐减少不可行解的个数,则可以使得算法能够充分利用一些重要的不可行解的有效信息。设不可行阈值为 eps,它的更新方式为

$$\mathrm{eps}(t)=\begin{cases}\mathrm{eps}_0\times(1-5t/3T) & t\leqslant 0.6T\\ 0 & t>0.6T\end{cases} \tag{7}$$

其中:t 是当前适应度的评价次数;T 为迭代总次数;eps_0 为初始不可行阈值,取 1。如此,算法可以利用一些重要的不可行解的优良信息,从而提高算法收敛到 Pareto 前沿的能力。

3 实验及结果分析

3.1 算法的性能度量

对于多目标优化算法而言,它的性能度量主要包含以下两个方面:a)计算算法在整个执行过程中所花费的时间或适应度函数的最大评价次数;b)比较算法获得最优解的优劣,可以通过数量化的方式对获得的最优解进行度量。现今,主要的性能度量方法[8]有:世代距离性能度量方法(GD),它是用于评价算法获得的 Pareto 前端和最优 Pareto 前端两者之间间隔的世代距离;非劣解间距度量方法(SP),用来评估算法获得的非劣解在最优 Pareto 前沿分布是否均匀;错误率度量法(ER),用来检验算法获得的非劣解不在 Pareto 前沿上的百分比。

3.2 算法实验结果

对于多目标优化问题,很难进行理论分析来评价一种算法的优劣,目前通常采用函数测试法来验证算法的优化性能。因此,为了验证本文改进算法的合理性和有效性,将本文算法(MOPSO)与带有精英策略的非支配排序遗传算法[9](NSGA-Ⅱ)、带两类正态变异的多目标粒子群算法[10](NMPSO)、基于 Pareto 熵的多目标粒子群算法[11](PEPSO)、基于多策略的多目标粒子群优化算法[12](MSPSO)在五个测试函数(ZDT1、ZDT2、ZDT3、ZDT4、ZDT6)上分别进行了测试,本文的实验数据均是通过 MATLAB 编程在 MATLAB 2012a 上获得的。最后,将五种不同算法的三个性能指标进行了比较。其中:ZDT1、ZDT2、ZDT6 函数有一个非凸的 Pareto 前沿;ZDT3 函数有一个非连续的 Pareto 前沿;ZDT4 函数具有局部 Pareto 最优。设置种群规模为 100,测试函数的维数分别为 30、30、30、10、10。五种算法分别运行 100 次,获得三种性能的平均值,对测试函数性能度量的三种均值度量结果如表 1~3 所示。

表 1 测试函数 GD 性能的均值

测试函数	NSGA-Ⅱ	NMPSO	PEPSO	MSPSO	MOPSO
ZDT1	5.3211e-002	2.0015e-005	4.2598e-005	3.1241e-004	3.1258e-007
ZDT2	3.5878e-002	4.2589e-003	4.6598e-003	7.9329e-003	6.3254e-006
ZDT3	2.3548e-002	7.2598e-005	6.6989e-003	5.5301e-005	1.2589e-006
ZDT4	1.2569e-003	1.1052e+001	6.3326e-001	3.9343e+001	5.3612e-003
ZDT6	0	0	0	0	0

表 2 测试函数 SP 性能的均值

测试函数	NSGA-Ⅱ	NMPSO	PEPSO	MSPSO	MOPSO
ZDT1	1.2589e-001	1.000e-000	1.000e-000	1.000e-000	1.000e-000
ZDT2	3.5285e-001	1.000e-000	1.000e-000	1.000e-000	1.000e-000
ZDT3	7.2215e-001	8.1159e-001	5.7498e-001	8.6509e-001	3.2598e-001
ZDT4	4.2589e-001	4.1498e+001	5.7089e+001	4.5495e+001	1.0025e-001
ZDT6	9.2001e-001	9.2366e-001	8.4736e-001	8.6598e-001	3.2589e-001

表 3 测试函数 ER 性能的均值

测试函数	NSGA-Ⅱ	NMPSO	PEPSO	MSPSO	MOPSO
ZDT1	1.000e-000	6.3265e-002	6.6258e-002	1.8312e-001	3.2598e-004
ZDT2	1.000e-000	2.4769e-002	3.1485e-003	4.4723e-002	5.3265e-004
ZDT3	1.000e-000	5.3665e-002	5.1988e-002	5.2394e-003	1.0025e-003
ZDT4	1.000e-000	1.000e-000	1.000e-000	1.000e-000	1.000e-000
ZDT6	0.000e-000	6.9759e-002	5.4532e-002	6.0136e-002	2.6981e-004

从表 1~3 中可以看出,本文提出的 MOPSO 算法在五个 ZDT 测试函数上的性能度量结果很好。对 GD 而言,它的值接近于 0,说明通过 MOPSO 算法求得的最优解与真实最优解之间的差异极小,收敛性较好;对 SP 而言,它的值较小,说明通过 MOPSO 算法得到的所有非劣解均匀地分布在 Pareto 前沿;本文提出的 MOPSO 算法的 ER 值也最小,表明执行 MOPSO 算法得到的大部分非劣解均分布在 Pareto 前沿上,算法获得的解较好。主要是因为 MOPSO 算法根据粒子的相似度值动态调整算法的惯性权重,高效地领导了整个粒子群体的进化,使得最优解更接近于真实的 Pareto 前沿。采用了动态加权法和动态改变阈值的可行性规则,使得 Pareto 最优解能够在 Pareto 前沿分布得更加均匀。因此,MOPSO 较其他四种算法而言,具有更好的性能优势。为更加直观地体现出 MOPSO 算法的寻优性能,本文将 MOPSO 算法与 NSGA-Ⅱ算法分别运行了 50 次,并将它们的 Pareto 前沿与真实的 Pareto 前沿画图进行了比较,结果如图 1 所示。从图 1 中可以很直观地看出,MOPSO 算法在收敛性上明显优于 NSGA-Ⅱ。

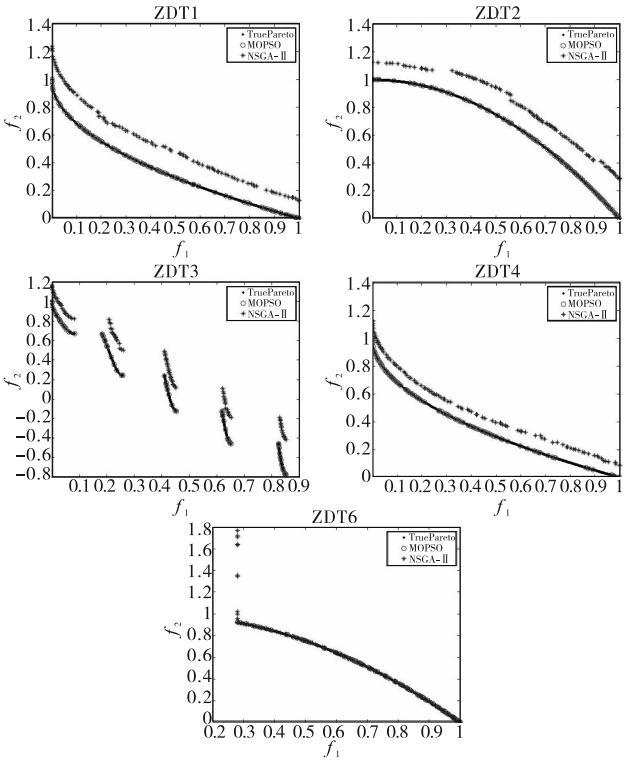

图1 MOPSO算法与NSGA-Ⅱ的比较结果

综上所述,MOPSO 算法在五个测试函数中(下转第 1336 页)

ger characteristics[J]. Journal of Public Transportation, 2006, 9(5): 23-40.
[2] Tao Chichung, Wu Chungjung. Behavioral responses to dynamic ride-sharing services-the case of taxi-sharing project in Taipei[C]//Proc of International Conference on Service Operations and Logistics, and Informatics. Piscataway, NJ: IEEE Press, 2008: 1576-1581.
[3] Deakin E, Frick K T, Shively K M. Markets for dynamic ridesharing, case of Berkeley, California[EB/OL]. https://dio. org/10. 3141/2187-17.
[4] 汤黎明,刘其华. 邻里合乘—社区拼车常态化的探索[J]. 城市交通,2010,8(6):29-33.
[5] 王茂福. 拼车的发展及其效应[J]. 中国软科学,2010(11):54-61.
[6] Minett P, Pearce J. Estimating the energy consumption impact of casual carpooling[J]. Energies, 2011, 4(1): 126-139.
[7] Calvo R W, De Luigi F, Haastrup P, *et al*. A distributed geographic information system for the daily carpooling problem [J]. Computers and Operations Research, 2004, 31(13): 2263-2278.
[8] Tao C C, Chen Chunying. Dynamic rideshare matching algorithms for the taxipooling service based on intelligent transportation system tech-nologies [C]//Proc of the 14th International Conference on Management Science and Engineering. Piscataway, NJ: IEEE Press, 2007: 20-22.
[9] 马华伟,左春荣,杨善林. 多时间窗车辆调度问题的建模与求解[J]. 系统工程学报,2009,24(5):607-613.
[10] 丁晓安,徐伯夏. 基于智能搜索功能的 Android 平台手机拼车系统设计[C]//全国第五届信号和智能信息处理与应用学术会议专刊(第一册). 2011: 89-92.
[11] 蔡延光,钱积新,孙优贤. 带时间窗的多重运输调度问题的自适应 Tabu Search 算法[J]. 系统工程理论与实践,2000,20(12):42-50.
[12] 隽志才,孙剑. 基于保守时间窗的交通分布式仿真同步算法研究与实现[J]. 公路交通科技,2004,21(6):84-87.
[13] Lin S W, Ying K C, Lee Z J, *et al*. Vehicle routing problems with time windows using simulated annealing[C]//Proc of IEEE International Conference on Systems, Man and Cybernetics. Piscataway, NJ: IEEE Press, 2006.
[14] 蔡延光,师凯. 带软时间窗的联盟运输调度问题研究[J]. 计算机集成制造系统,2006,12(11):1903-1908.
[15] 俞雪雷,周俊红,杨俊琴. 有交通条件约束与软时间窗约束的配送配载模型算法研究[J]. 交通与运输,2007(7):26-29.
[16] 袁二明,蔡小强,涂奉生. 基于时间窗的随机时变交通网络信号相位协调[J]. 计算机工程,2008,34(5):17-19.
[17] Chu Weibo, Guan Xiaohong, Cai Zhongmin, *et al*. A time window based analytical approach for splitting high-speed network traffic[C]//Proc of IEEE International Conference on Information Theory and Information Securing. Piscataway, NJ: IEEE Press, 2010: 294-300.
[18] Zulvia F E, Kuo R J, Hu Tunglai. Solving CVRP with time window, fuzzy travel time and demand via a hybrid ant colony optimization and genetic algorithm[C]//Proc of IEEE Congress on Evolutionary Computation. Piscataway, NJ: IEEE Press, 2012: 1-8.
[19] 陈飞飞,刘斌. 有时间窗的危险品道路运输路径选择研究[J]. 兰州交通大学学报,2013,32(3):109-111,117.
[20] Shao Kaili, Li Fengting. An optimization model and its algorithm of bulk grain transportation with time windows[C]//Proc of International Conference on Cyberspace Technology. Piscataway, NJ: IEEE Press, 2013: 496-499.
[21] 赖志柱. 和声搜索算法优化多时间窗多式联运运输方案[J]. 计算机应用,2013,33(9):2640-2642, 2693.
[22] 朱晓锋,林灼强. 基于时变交通流带时间窗的运输调度问题研究[J]. 中国储运,2014(3):112-115.
[23] Wang Jiankai, Indra-Payoong N, Sumalee A, *et al*. Vehicle re-identification with self-adaptive time windows for real-time travel time estimation[J]. IEEE Trans on Inteligent Transportation Systems, 2014, 15(2): 540-552.
[24] Correia G, Viegas J M. Carpooling clubs: solution for the affiliation problem in traditional & dynamic ridesharing systems[C]//Proc of the 10th Euro Working Group Transportation Meeting. 2005: 493-498.
[25] 汪漩,仲伟俊,梅姝娥. 合作型企业间电子商务中信任机制设计研究[J]. 系统工程学报,2006,21(1):34-39.
[26] Liu Nianbo, Liu Ming, Cao Jiannong, *et al*. When transportation meets communication: V2P over VANETs[C]//Proc of the 30th IEEE International Conference on Distributed Computing Systems. Washington DC: IEEE Computer Society, 2010: 567-576.
[27] Chaube V, Kavanaugh A L, Perez-Quinones M A. Leveraging social networks to embed trust in rideshare programs [C]//Proc of the 43rd Hawaii International Conference on Systems Science. Washington DC: IEEE Computer Society, 2010: 1-8.
[28] Habib K M N, Morency C, Islam M T, *et al*. Modeling users' behavior of a carsharing program: application of a joint hazard and zero inflated dynamic ordered probability model[J]. Transportation Research: Part A, 2001, 46(2): 241-254.

(上接第 1306 页)均取得了最优收敛性和多样性的相互平衡,表明它是一种高效的多目标算法。未来将在一些工程问题中检验 MOPSO 算法的性能,寻找更高效的优化模式,提高算法在求解多目标优化问题的综合性能。

4 结束语

在多目标粒子群算法的基础上,提出了可行性规则动态调整的多目标粒子群算法。在算法中,根据粒子与群体最优粒子之间的相似度值动态非线性地更新算法的惯性权重,使得算法可以有效地平衡全局搜索和局部搜索两者的矛盾;采用动态加权法解决随机性抽取群体最优粒子的缺陷,保证了种群的多样性;动态改变可行性规则的阈值,使得算法可以充分利用某些不可行解包含的有效信息,提高算法收敛到 Pareto 前沿的能力。与其他四种算法的数值实验结果表明,在解决多目标优化问题时,本文算法可以获得更好的收敛性及 Pareto 解的均匀分布性。

参考文献:

[1] Konak A, Coit D W, Smith A E. Multi-objective optimization using genetic algorithms: a tutorial[J]. Reliability Engineering & System Safety, 2006, 91(9): 992-1007.
[2] Meo S, Zohoori A, Vahedi A. Optimal design of permanent magnet flux switching generator for wind applications via artificial neural network & multi-objective particle swarm optimization hybrid approach[J]. Energy Conversion and Management, 2016, 110(2): 230-239.
[3] Li Xiaodong. A non-dominated sorting particle swarm optimizer for multi-objective optimization[C]//Genetic and Evolutionary Computation. Berlin: Springer, 2003: 37- 48.
[4] Lei Deming. Pareto archive particle swarm optimization for multi-objective fuzzy job scheduling problem[J]. Computers & Industrial Engineering, 2008, 54(4): 960-971.
[5] Tavakkoli-Moghaddam R, Azarkish M, Sadeghnejad A. A new hybrid multi-objective Pareto archive PSO algorithm for a bio-objective job shop scheduling problem[J]. Expert Systems with Application, 2010, 38(9): 10812-10821.
[6] Liu Dasheng, Tan K C, Goh C K, *et al*. A multiobjective memetic algorithm based on particle swarm optimization[J]. IEEE Trans on Systems, Man and Cybernctics, 2007, 37(1): 42-50.
[7] 陈民铀,张聪誉,罗辞勇. 自适应进化多目标粒子群优化算法[J]. 控制与决策,2009,24(12):1851-1855.
[8] 焦李成. 多目标优化免疫算法、理论和应用[M]. 北京:科学出版社,2009: 57-60.
[9] Deb K, Pratap A. A fast and elitist multi-objective genetic algorithm: NSGA-Ⅱ[J]. IEEE Trans on Evolutionary Computation, 2002, 6(2): 182-197.
[10] 高圣国,吴忠,李旭芳,等. 带两类正态变异的多目标粒子群算法[J]. 控制与决策,2015,30(5):939-942.
[11] 胡旺, Yen G G, 张鑫. 基于 Pareto 熵的多目标粒子群优化算法[J]. 软件学报,2014,25(5):1025-1050.
[12] 雷瑞龙,侯立刚,曹江涛. 基于多策略的多目标粒子群优化算法[J]. 计算机工程与应用,2016,52(8):19-24.