

基于快速二层解修补策略的区间离散遗传算法

李二超, 马玉泉

(兰州理工大学电气工程与信息工程学院, 甘肃 兰州 730050)

[摘要] 以等式约束下的区间离散多目标优化问题为研究对象, 提出了快速二层解修补策略, 其主要思想是: 首先, 用初始解生成器生成一个满足等式约束条件的种群, 然后, 将此种群中可修补个体以其违反约束度最小为目标函数, 将落在未定义区间的个体修补至定义区间内, 最后, 在定义子区间内微调得到满足约束条件的个体, 其调整方法如下: 按照当前个体中的每个变量在其所处区间内的可调节上下限在此个体总的可调节上下限值内所占的比例进行调节, 使得不满足等式约束的变量得到有效修补. 最后, 通过实验验证了本文算法的有效性.

[关键词] 多目标优化, 区间离散变量, 遗传算法, 快速二层解修补, 约束

[中图分类号] TP273 [文献标志码] A [文章编号] 1001-4616(2019)03-0073-07

Interval Discrete Genetic Algorithms Based on Fast Two-Level Solution Repair Strategy

Li Erchao, Ma Yuquan

(College of Electrical and Information Engineering, Lanzhou University of Technology, Lanzhou 730050, China)

Abstract: A fast two-level solution repair strategy is proposed for interval discrete multi-objective optimization problem with equality constraints. The main idea is: first, a population satisfying equality constraints is generated by using the initial solution generator, and then, the repairable individuals in this population are repaired to definitions by taking the minimum degree of violation of constraints as the objective function. In the interval, finally, the individual satisfying the constraints can be fine-tuned in the definition sub-interval. The adjusting method is as follows: according to the proportion of the upper and lower limits of each variable in the current individual in its interval, the variable that does not satisfy the equality constraints can be effectively repaired by adjusting the proportion of the upper and lower limits of the individual in the total adjustable upper and lower limits. Finally, the effectiveness of the proposed algorithm is verified by experiments.

Key words: multi-objective optimization, interval discrete variables, genetic algorithm, fast two-level repair, constraints

多目标优化是指两个或者两个以上相互冲突的待优化目标在一定约束条件下达到相互妥协最优的状态的一门学科, 多目标优化问题广泛存在于人们的生产实践中, 事实证明, 利用多目标优化技术得到的优化方案往往更优. 工程师和科学家们也相继提出了许多多目标优化算法, 传统多目标优化算法的基本原理是通过加权值的方法将多个待优化目标转化为一个待优化目标, 再按照单目标优化方法处理, 这种方法虽然操作简单, 但加权值难以确定. 近年来, 进化算法因其通用性和有效性而受到研究者和工程师们的青睐^[1-6], 并已成功应用在了各行各业中. 尽管多目标进化算法在连续优化问题中的应用已较为成熟, 但对离散多目标优化问题的研究相对较少, 在实际工程问题中却常常会碰到离散多目标优化问题. 在这种情况下, 已有许多科学家利用进化算法解决离散多目标优化问题, 如: Balram 等于 2004 年将多目标模拟退火算法 (multi-objective simulated annealing, MOSA) 成功应用在组合优化问题中^[7], Taysi 等于 2008 年将遗传算法应用在组合优化问题中^[8], 刘松兵等于 2009 年将蚁群算法成功应用在组合问题中^[9], 贾婷芳等于 2011 年将离散粒子群算法成功应用到资源分配问题 (resource allocation problem, RAP) 中并取得了良好的优化效果^[10], 苏凯等于 2012 年将离散分布约束算法与遗传算法结合, 应用在火力发电问题中, 效果明显^[11]. 其中, 文献[10-13]中涉及到的等式约束下的离散区间多目标优化问题研究相对较少, 且研究者没

收稿日期: 2019-07-05.

基金项目: 国家自然科学基金 (61763026, 61403175).

通讯联系人: 李二超, 博士, 副教授, 研究方向: 多目标优化. Email: lecstarr@163.com

有将其放在进化算法框架中进行研究,因此,利用进化算法解决此离散优化问题具有一定的研究意义.

1 问题描述

在某些涉及到离散多目标优化的工程问题中,有时为了满足某些特定要求,待优化问题决策变量的离散形式表现为离散区间集合的形式. 例如:厂级负荷经济调度问题需要考虑磨煤机启动与停止时相邻磨煤机出力交替区间,这些交替区间属于不可操作区域(prohibited operating zone, POZ),调度中心下发的负荷指令不能落入该区域^[12];梯级水电站群调度问题则需要考虑库群之间地理距离导致的水头不连续的影响^[13]. 类似以上两种情况下的优化问题,其决策变量一般表现为区间离散的形式,且当决策空间为非凸的时候,现有方法很难解决此类问题^[14-15].

离散区间多目标优化算法的难点在于决策变量空间是一系列离散区域的集合,其数学模型可表示如下:

$$x_i \in [x_{i,\min}, x_{i,1}^U] \cup [x_{i,1}^L, x_{i,2}^U] \cup [x_{i,2}^L, x_{i,3}^U] \cup \cdots [x_{i,j}^L, x_{i,j}^U] \cdots \cup [x_{i,m_i}^L, x_{i,\max}], \quad (1)$$

式中, x_i 表示第 i 个决策变量, $x_{i,\min}$ 表示第 i 个决策变量对应的最小值, $x_{i,\max}$ 表示第 i 个决策变量对应的最大值, $x_{i,j}^L$ 表示第 i 个决策变量对应的第 j 个可行区间的下限值, $x_{i,j}^U$ 表示第 i 个决策变量对应的第 j 个可行区间的上限值, m_i 表示第 i 个决策变量对应的可行区间数, x_{i,m_i}^L 表示第 i 个决策变量对应的第 m_i 个可行区间的下限值. 在空间中的表现形式为:在一维空间中,决策变量是一系列线段的集合(图 1);在二维空间中是一系列矩形的集合(图 2);在三维空间中是一系列长方体的集合(图 3).

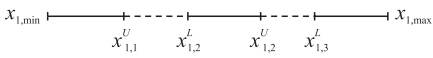


图 1 决策变量一维情况下的区间离散变量
Fig. 1 Interval discrete variables in one-dimensional case of decision variables

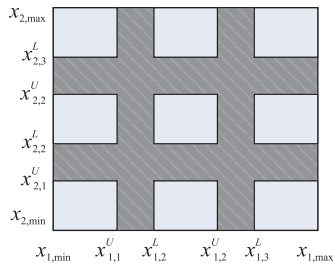


图 2 决策变量二维情况下的区间离散变量
Fig. 2 Interval discrete variables in two-dimensional case of decision variables

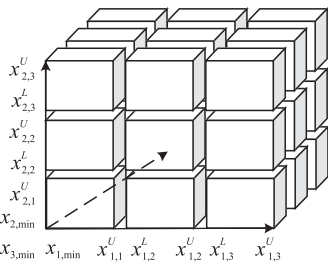


图 3 决策变量三维情况下的区间离散变量分布
Fig. 3 Distribution of interval discrete variables in three-dimensional case of decision variables

通常,可以将离散区间形式的多目标优化问题用以下数学模型来描述:

$$\begin{cases} \min F(x) = \min(f_1(x), f_2(x), \cdots, f_m(x)), \\ \text{Subject to:} \\ g_r(x) < 0, \quad r = 1, 2, \cdots, p, \\ g_j(x) = 0, \quad j = p+1, p+2, \cdots, q, \\ x_{i,\min} \leq x_i \leq x_{i,1}^U, \\ x_{i,2}^L \leq x_i \leq x_{i,3}^U, \\ \cdots \\ x_{i,k-1}^L \leq x_i \leq x_{i,\max}, \quad i = 1, 2, \cdots, n, \end{cases} \quad (2)$$

式中, m 表示多目标优化问题的目标个数, p 表示不等式约束的个数, q 表示等式约束和不等式约束的总个数, $i = 1, 2, \cdots, n$ 表示决策变量的维数, $x_{i,k}$ 表示第 i 个决策变量对应的第 k 个区间, $x_{i,\min}$ 和 $x_{i,\max}$ 分别表示第 i 个决策变量对应的下限值和上限值. 从其数学模型可以看出,区间离散形式的优化问题,其最大的特点在于决策变量是呈区间离散的形式.

2 算法描述

2.1 算法思想

本文提出了一种解决“等式约束下的非线性离散区间问题”优化算法,其提出的快速二层解修补策略

主要思想是:

(1)在生成初始化种群的时候,按照等式约束修补方法,生成规定数目满足等式约束的初始化种群.相当于在算法中加入一个初始解生成器.相比解约束法,用等式约束修补法生成的初始化种群更优质,能够保证种群稳定地进化和快速地收敛(生成的初始化种群都满足等式约束);

(2)因为对落在未定义区间的个体进行修补的对象是已经满足等式约束的个体,所以为了保证修补后的变量对等式约束的约束违反度尽可能小,在进行区间离散约束修补的时候,将落在非定义区间的决策变量以个体违反约束度最小为目标对其进行妥善调整到相应可行子区间;

(3)在经过以上两步后,就能够实现通过在定义子区间内微调得到满足约束条件的个体.其调整方法如下:按照当前个体中的每个变量在其所处区间内的可调节上下限在此个体总的可调节上下限值内所占的比例进行调节,使得不满足等式约束的变量得到有效修补.

2.2 算法核心

为了使得快速二层解修补策略更加清晰明了,在这里,以4个变量等式约束下区间离散优化问题为例说明改进型快速二层解修补策略,图4、图5形象地展示了其修补过程.

首先,按照等式约束条件对违反等式约束的不可行解进行修补,修补范围是整个区间;当初始化个体各变量值之和大于其对应各变量的上限值之和或者小于各变量的下限值之和时,会被判定为不可修复个体;对于不可修复个体将执行删除操作,并重新生成新的个体并对其进行修补,直到得到规定数目的初始化种群;在利用就近调整法(在离散区间内微调时,能够尽可能多地生成可行解)将落在未定义区间内的变量调整至定义区间后,再在此时各变量所在的离散区间之内进行微调直至满足所有约束条件.

快速二层解修补策略计算步骤:

步骤1 等式约束解修补:

步骤1.1 在各变量对应的区间范围内随机生成初始化解 $P_i (i=1,2,\dots,n)$, n 为决策变量数目;

步骤1.2 计算当前种群中每个个体对应决策变量之和与 total 的差值,如果为0,判断是否在定义区间内;如果既满足等式约束条件也满足区间约束,直接存放在预先设定的档案集中,否则,进入下面步骤1.3;

步骤1.3 计算每个个体各变量的可变动上下限 $P_{i,k}^u$ 和 $P_{i,k}^d$, $P_{i,k}$ 表示第 i 个个体对应的第 k 个可行区间,并计算每个个体的可调整上下限总和 $p_i^{au} = \sum_{k=1}^n p_{i,k}^u$ 和 $p_i^{ad} = \sum_{k=1}^n p_{i,k}^d$;

步骤1.4 计算当前个体与等式约束的误差值 $p_i^{\text{error}} = \sum_{k=1}^n p_{i,k} - \text{total}$, total 为等式约束的值;

步骤1.5 按照当前个体中每个变量的上下限调整值在整个个体上下限中所占的比例对每个个体进行调整;当 $p_i^{\text{error}} < 0$ 时,说明此时的个体值偏小,更新 p_i 为 $p_{i,\text{new}} = p_i + p_i^{\text{error}} \times \frac{p_{i,k}^u}{p_i^{au}}$;当 $p_i^{\text{error}} > 0$ 时,说明此时的个体

值偏大,更新 p_i 为 $p_{i,\text{new}} = p_i - p_i^{\text{error}} \times \frac{p_{i,k}^d}{p_i^{ad}}$;

步骤1.6 对调整后的 $p_{i,\text{new}}$ 做一个边界值限定,更新后的值大于对应变量的上限值的时候,设置成上限值,相反,则设置成下限值.

经过上述操作,种群中的个体只符合等式约束的要求,并不满足区间离散约束,某些决策变量可能会出现在未定义区间内,所以下面的工作是,按照一定的策略调整落入未定义区间内的决策变量的值,使其落入定义区间之内;但是将未定义区间的决策变量调整到定义区间时,势必会破坏等式约束条件,所以,将决策变量调整至定义区间后,还需要在调整后所在的定义区间内对决策变量值进行微调,直至其满足等式约束条件为止.等式约束修补和区间离散约束修补的修补顺序本来就是一对矛盾,先进行等式约束的修补,再进行区间离散的修补,调整后者会破坏前者约束;先进行区间离散约束修补,再进行等式约束修补,前者也会破坏后者的;所以,究竟是该先进行等式约束修补还是先进行区间离散约束修补有待研究.在本文中,笔者选择先进行等式约束修补,再进行区间离散约束修补.先进行等式约束修补可以将各决策变量大体调整至一个合适的范围,再通过决策变量区间离散的修补,将各决策变量进行较大调整,最后,通过等式约束的条件对各决策变量进行微调,直至满足等式约束.

步骤 2 区间离散约束修补:

步骤 2.1 统计当前个体中落入未定义区间的个体数目 m , 如果 m 为 0, 则直接将此个体保存到符合要求的档案集中, 否则, 转到步骤 2.2;

步骤 2.2 将此 m 个变量落入未定义区间的决策变量 p_i 按照欧氏距离离区间最近的原则, 将落入未定义区间的决策变量调整至离其最近的那个定义区间 $I_{i, \text{neighb}}$.

步骤 3 区间内微调弥补在修补区间离散约束时对等式约束条件的违背:

步骤 3.1 经过上面操作后, 所有的个体都落入了定义区间之内. 计算当前个体和等式约束值的差值

$$p'_{\text{error}} = \sum_{k=1}^n p_{i,k} - \text{total};$$

步骤 3.2 当 $p'_{\text{error}} = 0$ 时, 直接将此个体存到符合要求的存档集中; 当 $p'_{\text{error}} \neq 0$ 时, 计算当前个体中决策变量 p_i 在其所在的定义区间中的可调整上下限值 $p_{i,k}^u$ 和 $p_{i,k}^d$ 以及在区间内的可调整上下限总和 $p_{i,k}^{au}$ 和 $p_{i,k}^{ad}$;

步骤 3.3 当 $p'_{\text{error}} > 0$ 时, 且 $p'_{\text{error}} > p_{i,k}^{au}$ 时, 解无法修补, 所以放弃修补, 否则, 更新个体中的决策变量 $p_{i,k}^{\text{new}} = p_{i,k} - p'_{\text{error}} \times \frac{p_{i,k}^d}{p_{i,k}^{ad}}$; 当 $p'_{\text{error}} < 0$ 时, 且 $p'_{\text{error}} < p_{i,k}^{ad}$ 时, 解无法修补, 所以放弃修补, 否则, 更新个体中的决策变量 $p_{i,k}^{\text{new}} = p_{i,k} + p'_{\text{error}} \times \frac{p_{i,k}^u}{p_{i,k}^{au}}$;

步骤 3.4 对完成修补后的个体上下限做一个限制, 修补后的值不得超过对应变量的上下限;

步骤 3.5 完成解修补.

上述各步骤中对应的变量计算公式如下:

$$p_i^u = \sum_{i=1}^k p_{i,k}^u \quad (3)$$

$$p_i^d = \sum_{i=1}^k p_{i,k}^d \quad (4)$$

$$p_i^{\text{error}} = \sum_{i=1}^k p_{i,k}^u - \text{total} \quad (5)$$

$$p_{i,k}^{\text{new}} = p_i + p_i^{\text{error}} \times p_{i,k}^u / p_i^{au} \quad (6)$$

$$p_{i,k}^{\text{new}} = p_i - p_i^{\text{error}} \times p_{i,k}^d / p_i^{ad} \quad (7)$$

$$p'_{\text{error}} = \sum_{i=1}^k p_{i,k} - \text{total} \quad (8)$$

$$p_{i,k}^{\text{new}} = p_{i,k} - p'_{\text{error}} \times p_{i,k}^d / p_{i,k}^{ad} \quad (9)$$

$$p_{i,k}^{\text{new}} = p_{i,k} + p'_{\text{error}} \times p_{i,k}^u / p_{i,k}^{au} \quad (10)$$

上式中, total 表示等式约束值, 具体由等式约束条件决定.

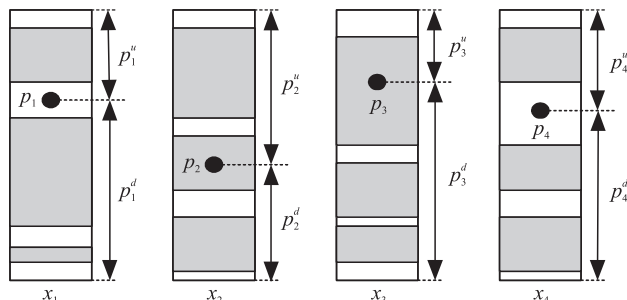


图 4 等式约束修补策略示意图

Fig. 4 Schematic diagram of constraint repair strategy for equation

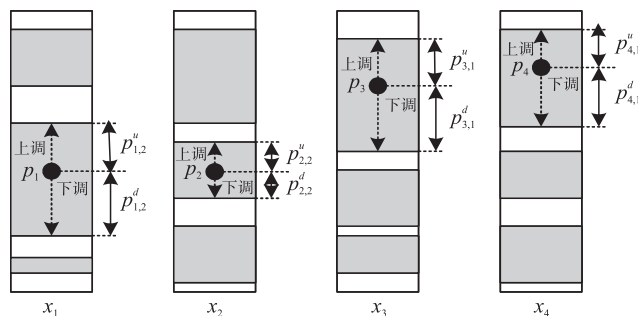


图5 区间离散约束二层解修补示意图

Fig. 5 Interval discrete constraints layer 2 solution patching diagram

2.3 算法流程

算法流程如图6所示。

3 算法测试和分析

本文主要测试算法在不同定义区间宽度下的性能,本次实验中,利用如表1~3所示3种情况下不同未定义区间宽度的情况,衡量两种修补策略的性能指标选用了当前普遍应用的收敛性指标 CS 和分布性指标 SP 以及在进化过程中可行解数目的变化情况。需要指出的是:对于收敛性指标 CS 需要同时考虑 $CS(P', P'')$ 和 $CS(P'', P')$ 两个指标。其中 $CS(P', P'')$ 表示解集 P'' 中的被解集 P' 中个体支配的所有个体在解集 P'' 中所占的比例,这个比例相对来说 P'' 越小越好(因为这说明 P'' 解要好于 P')。假设 P' 是对比算法的解集, P'' 是待测试算法的解集。当上述两个指标 $CS(P', P'') < CS(P'', P')$ 的时候,说明待测试算法收敛性要好于对比算法。对于分布性指标 SP , 越大说明解集的分布性越广,分布性越好。

测试函数模型:

$$\begin{aligned}
 F: \min f_1(x) &= (x_1 - 2)^2 + (x_2 - 1)^2, \\
 \min f_2(x) &= 9x_1 - (x_2 - 1)^2, \\
 \text{s.t. } x_1 + x_2 &= 1, \\
 x_1, x_2 &\in [-20, 20], \\
 x_1, x_2 &\in [-20, 15] \cup [-13, -10] \cup \\
 &\quad [-8, -4] \cup [-2, 2] \cup [4, 10] \cup [12, 20].
 \end{aligned}
 \tag{11}$$

表1 未定义区间宽度为2的定义区间

Table 1 Definition intervals of undefined interval width 2

变量	定义区间
x_1	$[-20, -15] \cup [-13, -10] \cup [-8, -4] \cup [-2, 2] \cup [4, 10] \cup [12, 20]$
x_2	$[-20, -15] \cup [-13, -10] \cup [-8, -4] \cup [-2, 2] \cup [4, 10] \cup [12, 20]$

表2 未定义区间宽度为3的定义区间

Table 2 Definition intervals of undefined interval width 3

变量	定义区间
x_1	$[-20, -15] \cup [-12, -10] \cup [-7, -4] \cup [-1, 2] \cup [5, 10] \cup [13, 20]$
x_2	$[-20, -15] \cup [-12, -10] \cup [-7, -4] \cup [-1, 2] \cup [5, 10] \cup [13, 20]$

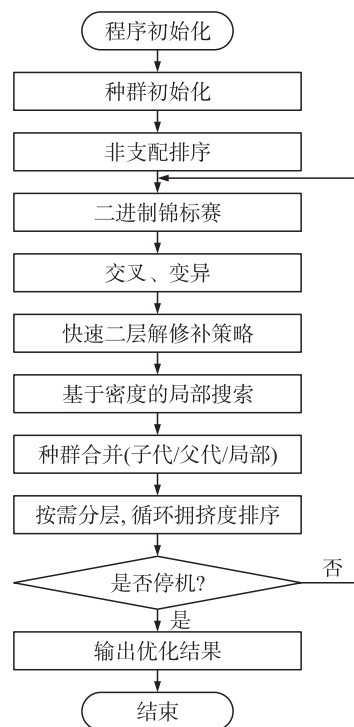


图6 算法流程图

Fig. 6 Flowchart of algorithm

表 3 未定义区间宽度为 4 的定义区间

Table 3 Definition intervals of undefined interval width 4

变量	定义区间
x_1	$[-20,-15] \cup [-11,-10] \cup [-6,-4] \cup [0,2] \cup [6,10] \cup [14,20]$
x_2	$[-20,-15] \cup [-11,-10] \cup [-6,-4] \cup [0,2] \cup [6,10] \cup [14,20]$

等式条件的构造是将整个区间进行离散区间划分后,按照所有可行离散区间进行组合后得出的结论平均值. 这样,就可以避免出现在等式约束下该离散区间无解的情况.

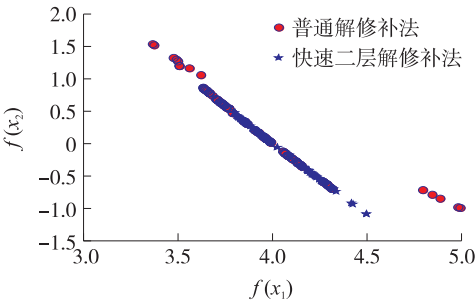


图 7 普通解修补法和快速二层解修补法优化结果比较
(未定义区间宽度为 2)

Fig. 7 The comparison of optimal between ordinary repairing and fast 2 layers repairing
(undefined interval width 2)

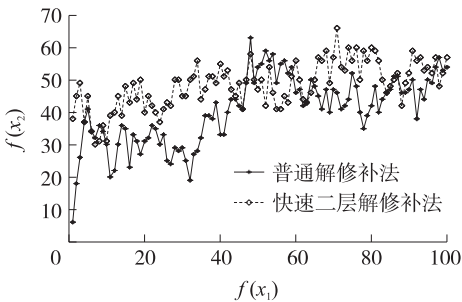


图 8 普通解修补法和快速二层解修补法可行解比较
(未定义区间宽度为 2)

Fig. 8 The comparison of amount of feasible solution between ordinary repairing and fast 2 layers repairing (undefined interval width 2)

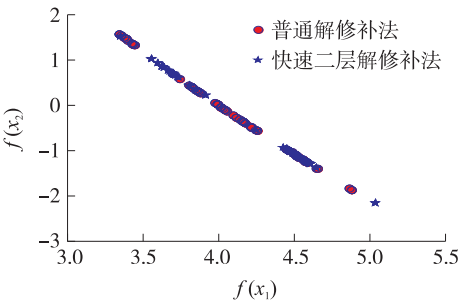


图 9 普通解修补法和快速二层解修补法优化结果比较
(未定义区间宽度为 3)

Fig. 9 The comparison of optimal between ordinary repairing and fast 2 layers repairing
(undefined interval width 3)

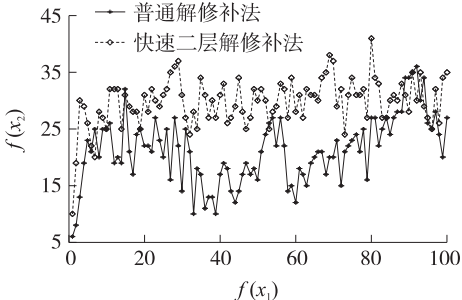


图 10 普通解修补法和快速二层解修补法可行解比较
(未定义区间宽度为 3)

Fig. 10 The comparison of amount of feasible solution between ordinary repairing and fast 2 layers repairing (undefined interval width 3)

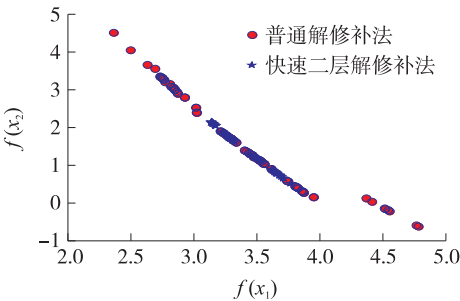


图 11 普通解修补法和快速二层解修补法优化结果比较
(未定义区间宽度为 4)

Fig. 11 The comparison of optimal between ordinary repairing and fast 2 layers repairing
(undefined interval width 4)

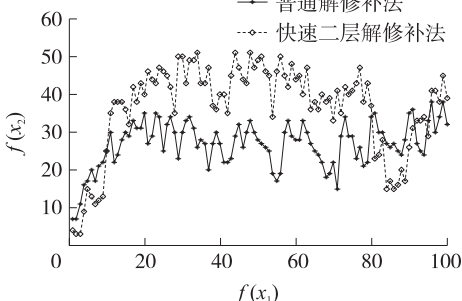


图 12 普通解修补法和快速二层解修补法可行解比较
(未定义区间宽度为 4)

Fig. 12 The comparison of amount of feasible solution between ordinary repairing and fast 2 layers repairing (undefined interval width 4)

得到的各性能指标如表 4、表 5.

表 4 不同宽度区间下收敛性指标比较

Table 4 Comparison of convergence indicators for different width intervals

未定义区间宽度	$CS(P', P'')$	$CS(P'', P')$
2	0.096 0	0.126 6
3	0.080 0	0.099 6
4	0.063 0	0.089 0

表 5 不同宽度区间下分布性指标比较

Table 5 Comparison of distribution indicators for different width intervals

未定义区间宽度	SP	
	对比算法	本文算法
2	1.296 7e-016	6.787 1e-016
3	2.029 6e-016	4.653 4e-016
4	3.387 0e-016	8.101 2e-016

从上面的实验结果图(图 7~图 12)可以看出,本文提出的快速二层解修补策略在算法进程中可行解的数量要多于普通解修补方法,这说明本文提出的快速二层解修补策略对不可行解的修补质量要高于普通解修补方法;同时,从表 4、5 可以看出,针对形如式(11)等式约束下的区间离散约束优化问题,利用快速二层解修补法得到的 Pareto 前沿在收敛性和分布性方面都要好于普通解修补方法. 以上实验证明了本文方法的有效性.

4 结语

本文主要是以第二代非支配遗传算法为基本框架对决策变量为离散区间形式的离散优化问题进行了研究. 针对等式约束下决策变量为离散区间的离散优化问题,提出了基于遗传算法的快速二层解修补算法,其特点主要有以下几点:(1)在生成初始化种群的时候,按照等式约束修补方法,生成规定数目满足等式约束的初始化种群,保证种群能够稳定地进化和快速地收敛(生成的初始化种群都满足等式约束);(2)因为对落在未定义区间的不可行解进行修补的对象是已经满足等式约束的个体,所以,为了保证修补后的变量对等式约束的约束违反度尽可能小,在进行区间离散约束修补的时候,将落在非定义区间的决策变量按照就近原则调整到离其最近(就近取值)的那个定义区间内;(3)在进行区间内微调的时候,按照当前个体中的每个变量在其所处区间内的可调节上下限在当前个体总的可调节上下限中所占的比例进行调节,使得不满足等式约束的变量得到有效修补. 最后,利用相关算例证明了本文算法的有效性.

[参考文献]

[1] 侯伟. 基于多目标优化算法的动压滑动轴承设计[J]. 控制工程,2018,25(6):1044-1049.

[2] 朱光宇,徐文婕. 考虑能耗与质量的机床构件生产线多目标柔性作业车间调度方法[J]. 控制与决策,2019,34(2):252-260.

[3] 王嵘冰,徐红艳,郭军. 自适应的非支配排序遗传算法[J]. 控制与决策,2018,33(12):2191-2196.

[4] 栗三一,李文静,乔俊飞. 一种基于密度的局部搜索 NSGA2 算法[J]. 控制与决策,2018,33(1):60-66.

[5] 侯莹,韩红桂,乔俊飞. 基于参数动态调整的多目标差分进化算法[J]. 控制与决策,2017,32(11):1985-1990.

[6] 郑金华. 多目标进化算法及其应用[M]. 北京:科学出版社,2007.

[7] BALRAM S. Study of simulated annealing based algorithms for multi-objective optimization of constrained problem[J]. Computers and chemical engineering,2004,28:1849-1872.

[8] TAYSI N,GOGUS M T,OZAKEA M. Optimization of arches using genetic algorithm[J]. Computer Optim Appl,2008,41:378-383.

[9] 刘松兵. 面向多目标优化的群智能算法研究[D]. 长沙:湖南大学,2009.

[10] 贾婷芳,张学良. 离散多目标优化算法的研究[J]. 机械工程与自动化,2011,5:206-208.

[11] 苏凯,刘吉臻,牛玉广,等. 一类解修补方法在厂级负荷优化分配中的应用[J]. 华北电力大学学报,2012,39(3):71-77.

[12] SUSHIL K R N. Nonconvex economic load dispatch using an efficient real-coded genetic algorithm[J]. Applied soft computing,2009,9(1):169-176.

[13] 贾江涛,管晓宏,翟桥柱. 考虑水头影响的梯级水电站群短期优化调度[J]. 电力系统自动化,2009,33(13):7-13.

[14] BAUMRUCKER B T. Mathematical programs with equilibrium constraint(MPECs)in process systems engineering[D]. Pittsburgh:Carnegie Mellon University,2009:1-26.

[15] ZHI Q,LUO J S P,Daniel R. Mathematical programs with equilibrium constraints[M]. Cambridge:The Press Syndicate of the University of Cambridge,1996:5-16.

[责任编辑:丁 蓉]