[Journal of System Simulation](https://dc-china-simulation.researchcommons.org/journal)

Volume 30 | Issue 1 Article 12

1-2-2019

Dynamic Relaxation Cooperative Optimization Method with Fast Convergence

Chen Jing
1.College of Mechanical and Control Engineering, Guilin 541004, China; ;

Yuchao Lü
2.College of Information Science and Engineering, Guilin University of Technology, Guilin 541004, China;

Limin Wang
2.College of Information Science and Engineering, Guilin University of Technology, Guilin 541004, China;

Dynamic Relaxation Cooperative Optimization Method with Fast Convergence

Abstract

Abstract: To solve problem of high computational cost and low convergence speed of initial points away from the optimal solution in collaborative optimization, *a new dynamic relaxation cooperative optimization method with fast convergence is presented. Two-phase optimization is adopted in this method. In the accelerating convergence phase, the calculation method of relaxation factor is improved, and the inconsistent information between the optimal value of disciplines and its mean value is used to construct the relaxation factor. The optimization solution of the first phase is adopted as the initial points in the optimization solution phase. The relaxation factor satisfying the consistent precision requirement is selected for cooperative optimization, and the global optimal solution is obtained.* A typical numerical example and the reducer MDO problem are adopted to test this optimization method. Experimental results show that the proposed method can greatly reduce the computational cost and accelerate the convergence speed of the initial points away from the optimal solution.

Keywords

multidisciplinary optimization, collaborative optimization, relaxation factor

Recommended Citation

Chen Jing, Lü Yuchao, Wang Limin. Dynamic Relaxation Cooperative Optimization Method with Fast Convergence[J]. Journal of System Simulation, 2018, 30(1): 96-104.

第 30 卷第 1 期 系统仿真学报© Vol. 30 No. 1
2018 年 1 月 Journal of System Simulation Jan., 2018

一种快速收敛的动态松弛协同优化方法

陈静 [1]，吕玉超 [2]，王利敏 [2]
（1.桂林理工大学机械与控制工程学院，广西 桂林 541000；2.桂林理工大学信息科学与工程学院，广西 桂林 541000）

摘要：针对 CO 优化过程中计算成本较大，远离最优解的初始点收敛速度较慢的问题，*提出一种快速收敛的动态松弛协同优化方法。该方法的计算过程分为两个阶段：加速收敛阶段对松弛因子的计算方法进行改进，采用各学科优化解与优化解均值之间的不一致信息构造松弛因子；优化求解阶段以加速收敛阶段的最优解作为初始点，选取符合一致性精度要求的松弛因子进行协同优化，求得全局最优解。*通过典型数值算例和减速器多学科设计优化问题对该方法进行验证，结果表明，该方法能够有效降低计算成本，加快远离最优解初始点的收敛速度。

关键词：多学科优化；协同优化；松弛因子
中图分类号：TP301.6 文献标识码：A 文章编号：1004-731X (2018) 01-0096-09
DOI: 10.16182/j.issn1004731x.joss.201801012

Dynamic Relaxation Cooperative Optimization Method with Fast Convergence

Chen Jing[1], *Lü YuChao*[2], *Wang Limin*[2]
(1. College of Mechanical and Control Engineering, Guilin 541004, China;
2. College of Information Science and Engineering, Guilin University of Technology, Guilin 541004, China)

Abstract: To solve problem of high computational cost and low convergence speed of initial points away from the optimal solution in collaborative optimization, *a new dynamic relaxation cooperative optimization method with fast convergence is presented. Two-phase optimization is adopted in this method. In the accelerating convergence phase, the calculation method of relaxation factor is improved, and the inconsistent information between the optimal value of disciplines and its mean value is used to construct the relaxation factor. The optimization solution of the first phase is adopted as the initial points in the optimization solution phase. The relaxation factor satisfying the consistent precision requirement is selected for cooperative optimization, and the global optimal solution is obtained.* A typical numerical example and the reducer MDO problem are adopted to test this optimization method. Experimental results show that the proposed method can greatly reduce the computational cost and accelerate the convergence speed of the initial points away from the optimal solution.

Keywords: multidisciplinary optimization; collaborative optimization; relaxation factor

引言

多学科设计优化(Multidisciplinary Design Optimization，MDO)通过探索和利用系统中相互作用的协同机制，利用多目标策略和计算机辅助技术来设计复杂系统及子系统，可以有效缩短设计周期，并获取系统整体最优性能。协同优化(Collaborative Optimization，CO)方法[1]是一种较高效且使用较多的多学科设计优化方法。该方法可以将复杂系统问题分解为两级优化问题，在系统级的

收稿日期：2015-11-24 修回日期：2016-02-13；
基金项目：国家自然科学基金(51365010)；
作者简介：陈静(1963-)，女，重庆，学士，副教授，研究方向为多学科设计优化；吕玉超(1990-)，男，济南，硕士，研究方向为多学科设计优化；王利敏(1992-)女，开封，硕士，研究方向为多学科设计优化。

协调下，学科级可实现分布式并行优化。CO 方法的分解形式符合实际工程的设计分工组织形式，因此，CO 方法得到了广泛应用。文献[2]使用 CO 方法解决飞机族设计问题。文献[3]将 CO 方法与多目标优化相结合解决多学科多目标优化问题。文献[4]将 CO 方法应用于电气领域复杂装置的优化设计。

CO 方法也存在一些不完善之处，CO 方法系统级优化中的一致性等式约束是一种理想状态[5]。在一致性等式约束条件下，系统优化难以进行，从而导致系统级优化问题的可行域可能不存在[6]。另外，许多研究结果表明，CO 方法的优化结果对初始点选取敏感，且容易收敛到局部最优解[7]。针对这些问题，文献[8]提出松弛因子法，将等式约束变为不等式约束。文献[5]从几何角度出发，利用学科间不一致信息动态调整松弛因子，在一定程度上解决了系统级优化解不存在的问题。文献[9]对初始点在可行域内外两种情况分别进行了分析研究，当初始点选在系统可行域内时，动态松弛因子法失效。文献[1]从全局稳定性出发，使用全局优化和局部优化两阶段优化策略，在两个阶段分别使用不同的松弛因子计算方法，该方法有效避免了协同优化方法初始点选取敏感的问题。文献[7]定义耦合一致度来衡量各学科变量间的一致性要求，并用耦合一致度确定系统级松弛因子，有效避免了 CO 方法收敛到局部最优解的问题。文献[10]依据各学科约束在设计变量空间的位置将设计空间划分为三类区域，并给出对应区域松弛因子的计算方法，提高了优化精度。文献[11]省掉各学科优化解不一致信息比较计算的步骤，直接使用学科级目标函数值来计算松弛因子，加快了系统收敛到全局最优解附近区域的速度。

目前，针对 CO 方法的研究，基本上解决了 CO 方法系统级优化可行域可能不存在以及对初始点选取敏感的问题，但是，CO 方法计算成本较大，并且对于远离最优解的初始点收敛速度较慢。因此，本文针对 CO 优化过程中计算成本较大，远离最优解的初始点收敛速度较慢的问题，对松弛因子的计算方法进行改进，提出一种快速收敛的动态松弛协同优化(Fast Dynamic Relaxation Collaborative Optimization，FDRCO)方法，以加快远离最优解初始点的收敛速度，降低计算成本，并通过算例进行验证。

1 CO 方法与松弛因子法

1.1 CO 方法

CO 方法将复杂工程系统设计问题分解为两级优化问题。在系统级一致性等式约束的协调下，学科级并行优化，经过系统级的多次协调，最终得到最优设计方案，假设系统设计问题涉及 n 个学科，系统级的数学模型为：

$$\left.\begin{aligned}&\min F(\mathbf{z})\\&\text{s.t.}\quad J_i^*(\mathbf{z})=\sum_{j=1}^{s_i}(z_j-x_{ij}^*)^2=0\\&i=1,2,\ldots,n\end{aligned}\right\}\tag{1}$$

式中：$\mathbf{z}$ 为系统级设计向量；s_i 为第 i 个学科的设计变量数；z_j 为第 j 个系统级设计变量；x_{ij}^*为第 i 个学科的第 j 个设计变量的优化结果。

学科级 i 的数学模型为：

$$\left.\begin{aligned}&\min J_i(\mathbf{x_i})=\sum_{j=1}^{s_i}(x_{ij}-z_j^*)^2\\&\text{s.t.}\quad C_i(\mathbf{x_i})\leqslant 0\end{aligned}\right\}\tag{2}$$

式中：$\mathbf{x_i}$为第 i 个学科的设计向量；x_{ij} 为第 i 个学科的第 j 个设计变量；z_j^*为系统级分配给学科级的第 j 个设计变量期望值；$C_i(x_i)$为第 i 个学科的约束函数。

1.2 松弛因子法

松弛因子法的主要思想是通过一个适当大小的松弛因子将一致性等式约束变为不等式约束，使得优化搜索空间变大，更容易得到全局最优解。采用松弛因子对一致性等式约束进行松弛，此时系统级的数学模型为：

$$\left.\begin{aligned}&\min F(\mathbf{z})\\&\text{s.t.}\quad J_i^*(\mathbf{z})=\sum_{j=1}^{s_i}(z_j-x_{ij}^*)^2<s\\&i=1,2,\ldots,n\end{aligned}\right\}\tag{3}$$

式中：s 为松弛因子。

https://dc-china-simulation.researchcommons.org/journal/vol30/iss1/12
DOI: 10.16182/j.issn1004731x.joss.201801012

第 30 卷第 1 期 系统仿真学报 Vol. 30 No. 1
2018 年 1 月 Journal of System Simulation Jan., 2018

松弛因子法主要包括静态松弛因子法和动态松弛因子法。静态松弛因子法使用一个固定的极小值作为松弛因子，但松弛因子的取值难以确定，取值过小会导致系统级优化可行域可能不存在，优化难以继续进行；取值过大又会失去一致性约束的意义[12]。

动态松弛因子法利用学科间的不一致信息动态调整松弛因子，以两学科为例，学科间的不一致信息为[5]：

$$\Delta = \|x_1 - x_2\| \tag{4}$$

式中：x_1 和 x_2 为两个学科的优化解；$\|\bullet\|$ 表示范数。

松弛因子的计算公式为[5]：

$$s = (\lambda \times \Delta)^2 \tag{5}$$

式中：λ 为常系数，$0.5 \leqslant \lambda \leqslant 1$。

动态松弛因子法的主要思想是利用 s 构造设计空间里的一个超球，s 表示超球空间的半径，通过动态改变 s 的值来逐步缩小设计空间，从而有效增强学科间的一致性。若 CO 问题涉及 n 个学科，则学科间的不一致信息 Δ 共有 $n(n-1)/2$ 个，为了保证各学科都在松弛范围内，需要比较后取最大的 Δ 值进行松弛因子的计算，计算成本较大[7]。并且，当选取的系统初始点距离最优解越远，学科间的不一致信息越大，这势必会造成系统收敛速度变慢，优化效率降低。因此，松弛因子法存在计算成本较大，初始点选取敏感，远离最优解的初始点收敛速度较慢等问题。

2 快速收敛的动态松弛协同优化方法

为解决松弛因子法存在的问题，本文提出了 FDRCO 方法，定义各学科优化解与其均值之间的不一致信息来衡量学科间的不一致度；为避免 FDRCO 方法对初始点选取敏感，借鉴文献[7]提出的渐进松弛的思想，将 FDRCO 方法分成加速收敛和优化求解两个阶段，并以加速收敛阶段的优化解作为优化求解阶段的初始点。

2.1 加速收敛阶段

随着协同优化的进行，各学科优化解与其均值之间的距离逐渐缩短，而各学科优化解与其均值之间的距离越短，则各学科优化解之间的不一致程度就越低，学科级返回给系统级的优化解就越能满足系统级一致性约束条件。因此，各学科优化解与其均值之间的不一致信息也可以用来衡量学科间的不一致程度。本文对松弛因子的计算方法进行改进，采用各学科优化解与其均值之间的不一致信息构造松弛因子，其具体的计算方法如下：

定义各学科优化解与优化解均值之间不一致信息为：

$$h = \|x_i^* - z\| \tag{6}$$

式中：x_i^* 为学科 i 的优化解；z 为各学科优化解的均值。

借鉴文献[5]中松弛因子的计算方法，松弛因子的计算公式为：

$$s = (\lambda \times h)^2 \tag{7}$$

式中：λ 为常系数，$0.5 \leqslant \lambda \leqslant 1$，该值可以保证系统级优化向学科间不一致信息向减小的方向进行。

为保证各学科都在松弛范围内，式(7)中的 h 应取各学科优化解与优化解均值之间不一致信息的最大值。由于各学科优化解与优化解均值之间的不一致信息只有 n 个，而学科间的不一致信息 Δ[5] 有 $n(n-1)/2$ 个，n 个数求最大值的计算复杂度为 $O(n)$，$n(n-1)/2$ 个数求最大值的计算复杂度为 $O(n^2)$，通过对比不一致信息的数量以及求最大值的计算复杂度可知，改进后的松弛因子计算方法，可以有效降低计算成本，减少计算时间，加快远离最优解初始点的收敛速度。

在加速收敛阶段，系统级第 k+1 次优化搜索空间，是以第 k 次各学科优化解 $X_i^*(x_{i1}^*, x_{i2}^*, x_{i3}^*, \ldots, x_{im}^*)$ 为球心，s 为半径的 m 维相交超球面，而第 k 次各学科优化解的均值 $Z(z_1, z_2, z_3, \ldots, z_m)$ 便在第 k+1 次系统级优化的搜索空间中[13]。因此，各学科优化解均值对系统级优化起指向性作用，每一次迭代过程中，各学科优化解都会向优化解均值方向移动，降低学科间的不一致度。由于采用各学科优化解均值作为参照点，加大了每一次迭代过程中各学科优化

解的移动距离，减少了优化解的收敛时间。这从另一方面说明，改进后的松弛因子计算方法，可以加快远离最优解初始点的收敛速度。

2.2 优化求解阶段

优化求解阶段以加速收敛阶段的最优解作为初始点，采用静态松弛因子法进行协同优化，并得到全局最优解。由于初始点已在全局最优解附近区域，可以避免优化陷入局部最优解，且迭代次数大大减少[7]。

静态松弛因子法的关键是选取合理的松弛因子。文献[14]研究表明，松弛因子取 10^{-4} 时，既能保证准确的优化结果，又能保证合理的收敛时间，并且能够减少系统级优化困难，满足各学科之间的一致性，因此，优化求解阶段的松弛因子选取 10^{-4}。同样，依据不同的实际问题，可以选择不同数量级的松弛因子。FDRCO 方法的计算流程如图 1 所示。

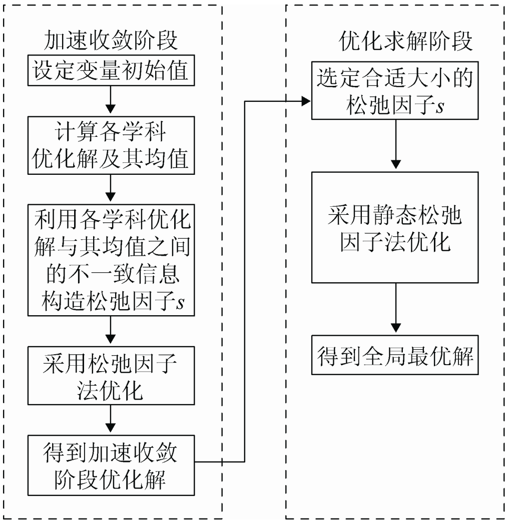

图 1 FDRCO 方法计算流程
Fig.1 Calculation process of FDRCO method

3 验证算例

3.1 数值算例

数值算例对一个约束非线性优化问题进行研究[15]：

$$\left.\begin{aligned} &\min f(\boldsymbol{x}) = x_1^2 + x_2^2 \\ &\text{s.t. } g_1 = x_1 + 0.1x_2 \leqslant 4 \\ &\qquad g_2 = 0.1x_1 + x_2 \geqslant 2 \end{aligned}\right\} \tag{8}$$

加入松弛因子后，转换为 CO 模型，包括一个系统级优化模型与两个学科级优化模型。

系统级优化模型：

$$\left.\begin{aligned} &\min f(\boldsymbol{x}) = \mathrm{x}_1^2 + x_2^2 \\ &\text{s.t. } J_1(\boldsymbol{x}) = (x_1 - z_{11}^*)^2 + (x_2 - z_{12}^*)^2 \leqslant s \\ &\qquad J_2(\boldsymbol{x}) = (x_1 - z_{21}^*)^2 + (x_2 - z_{22}^*)^2 \leqslant s \end{aligned}\right\} \tag{9}$$

学科 1 优化模型：

$$\left.\begin{aligned} &\min J_1(z_1) = (z_{11} - x_1^*)^2 + (z_{12} - x_2^*)^2 \\ &\text{s.t. } z_{11} + 0.1z_{12} \leqslant 4 \end{aligned}\right\} \tag{10}$$

学科 2 优化模型：

$$\left.\begin{aligned} &\min J_2(z_2) = (z_{21} - x_1^*)^2 + (z_{22} - x_2^*)^2 \\ &\text{s.t. } 0.1z_{21} + z_{22} \geqslant 2 \end{aligned}\right\} \tag{11}$$

文献[15]给出了优化结果：x_1=0.198，x_2=1.980，$f(x)$=3.9596。为了验证 FDRCO 方法在任何情况下都是高效可行的，本文选取 4 个具有代表性的初始点。在可行域内的有初始点 1 和初始点 2，在可行域外的有初始点 3 和初始点 4。针对该数值算例，在 FDRCO 方法中选取 $\lambda = 0.6$，分别使用标准 CO 方法、动态松弛 CO 方法、FDRCO 方法对该数值算例进行优化。4 个初始点的优化结果如表 1-4 所示，约束 1 和约束 2 的满足程度可以用来评价方法的可行程度。

表 1 初始点 1 的优化结果
Tab.1 Optimal solution of initial points 1

优化方法	初始点	最优解	目标函数值	约束 1	约束 2	迭代次数
标准 CO 方法	1, 2	1.000 0, 2.000 0	5.000 0	–2.800 0	0.100 0	1
动态松弛 CO 方法	1, 2	1.000 0, 2.000 0	5.000 0	–2.800 0	0.100 0	1
FDRCO 方法	1, 2	0.218 0, 1.968 1	3.921 1	–3.585 1	0.000 0	36

https://dc-china-simulation.researchcommons.org/journal/vol30/iss1/12
DOI: 10.16182/j.issn1004731x.joss.201801012

表 2 初始点 2 的优化结果
Tab.2 Optimal solution of initial points 2

优化方法	初始点	最优解	目标函数值	约束 1	约束 2	迭代次数
标准 CO 方法	2, 3	2.000 0, 3.000 0	13.000 0	–1.700 0	1.200 0	1
动态松弛 CO 方法	2, 3	2.002 0, 3.000 0	13.000 0	–1.700 0	1.200 0	1
FDRCO 方法	2, 3	0.198 7, 1.971 3	3.925 8	–3.604 2	0.000 0	48

表 3 初始点 3 的优化结果
Tab.3 Optimal solution of initial points 3

优化方法	初始点	最优解	目标函数值	约束 1	约束 2	迭代次数
标准 CO 方法	3, 1.5	3.019 9, 1.701 5	12.014 8	–0.809 9	0.003 5	11
动态松弛 CO 方法	3, 1.5	0.276 7, 2.024 1	4.173 5	–3.520 9	0.051 7	50
FDRCO 方法	3, 1.5	0.204 1, 1.971 4	3.928 1	–3.604 2	0.000 0	35

表 4 初始点 4 的优化结果
Tab.4 Optimal solution of initial points 4

优化方法	初始点	最优解	目标函数值	约束 1	约束 2	迭代次数
标准 CO 方法	4, 5	3.791 1, 2.081 5	18.705 1	0.000 0	0.460 6	54
动态松弛 CO 方法	4, 5	0.556 9, 1.943 5	4.087 7	–3.248 7	0.000 0	73
FDRCO 方法	4, 5	0.194 7, 1.971 8	3.926 0	–3.608 1	0.000 0	44

由表 1 和表 2 的优化结果可知，标准 CO 方法和动态松弛 CO 方法对可行域内的初始点只进行了一次迭代，优化结果为初始点本身，FDRCO 方法将 2 个可行域内的初始点均优化到了全局最优解附近。由表 3 和表 4 的优化结果可知，标准 CO 方法将可行域外的初始点优化到了靠近初始点的极值点。动态松弛 CO 方法将可行域外的初始点优化到了靠近全局最优解的局部极值点。FDRCO 方法将两个可行域外的初始点都优化到了全局最优解附近。

由此可知，标准 CO 方法对初始点选取敏感，只对可行域外的初始点有优化作用，且优化结果为靠近初始点的极值点，优化精度较差。动态松弛 CO 方法同样对初始点选取敏感，只对可行域外的初始点起优化作用，且优化结果为靠近全局最优解的局部极值点，优化精度不高。FDRCO 方法将 4 个初始点全都优化到了全局最优解附近，有较好的稳定性和较高的优化精度，同时迭代次数较少，有较高的优化效率。3 种方法的约束都满足条件，表明 3 种方法都有较好的可行性。

以远离最优解的初始点(3,1.5)为例，图 2 和图 3 所示为在 FDRCO 方法下系统变量 x_1 在加速收敛阶段和优化求解阶段的曲线图。图 4 和图 5 所示为在 FDRCO 方法下系统变量 x_2 在加速收敛阶段和优化求解阶段的曲线图。由图 2 和图 4 可知，系统变量 x_1 和 x_2 在加速收敛阶段经过 17 次迭代收敛至全局最优解附近。由图 3 和图 5 可知，系统变量 x_1 和 x_2 在优化求解阶段固定松弛因子的作用下，经过 18 次迭代收敛至全局最优解。由此可以看出，改进的松弛因子计算方法可以有效的加快初始点的收敛速度，从而提高整体的优化效率。因此，FDRCO 方法是有效可行的，并且是快速高效的。

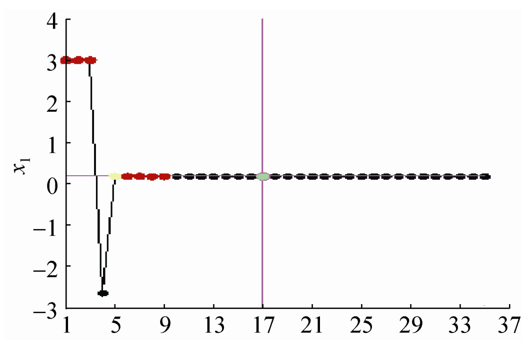

图 2 系统变量 x_1 加速收敛阶段优化曲线
Fig.2 Optimized curve of system variable x_1 at the accelerating convergence stage

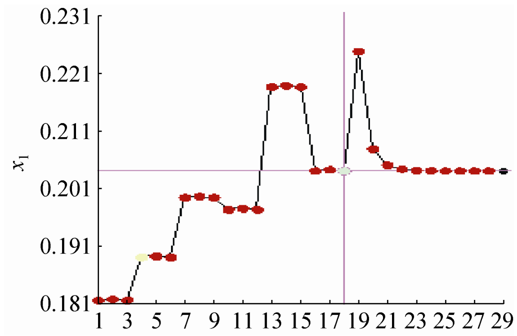

图 3 系统变量 x_1 优化求解阶段优化曲线
Fig.3 Optimized curve of system variable x_1 at the optimal solution stage

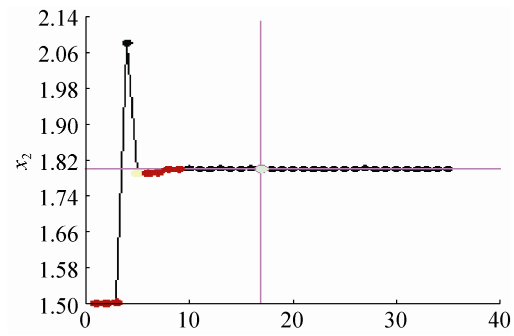

图 4 系统变量 x_2 加速收敛阶段优化曲线
Fig.4 Optimized curve of system variable x_2 at the accelerating convergence stage

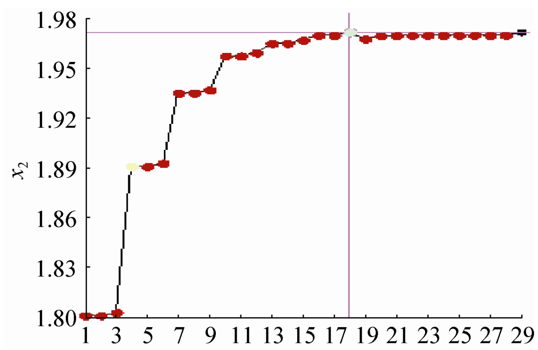

图 5 系统变量 x_2 优化求解阶段优化曲线
Fig.5 Optimized curve of system variable x_2 at the optimal solution stage

3.2 减速器算例

减速器设计问题是美国国家航空与航天局评估多学科设计优化方法性能的 10 个标准算例之一。减速器设计问题的目标是在满足转轴和齿轮约束条件的同时，使减速器体积或重量最小。该设计问题有 7 个设计变量：x_1 为齿面宽度，x_2 为齿轮模数，x_3 为小齿轮齿数，x_4 和 x_5 为轴承间距，x_6 和 x_7 分别为小齿轮和大齿轮的直径。减速器设计问题的数学模型为：

$$\left.\begin{aligned}
&\min f(X)=0.7854x_1x_2^2(3.3333x_3^2+14.933x_3-\\
&\qquad 43.0934)-1.5079x_1(x_6^2+x_7^2)+\\
&\qquad 7.477(x_6^3+x_7^3)+0.7854(x_4x_6^2+x_5x_7^2)\\
&\text{s.t.}\quad g_1=27/(x_1x_2^2x_3)-1.0\leqslant 0\\
&\qquad g_2=397.5/(x_1x_2^2x_3^2)-1.0\leqslant 0\\
&\qquad g_3=1.93x_4^3/(x_2x_3x_6^4)-1.0\leqslant 0\\
&\qquad g_4=1.93x_5^3/(x_2x_3x_7^4)-1.0\leqslant 0\\
&\qquad g_5=\sqrt{(745x_4/(x_2x_3))^2+16.9\times 10^6}\,/\\
&\qquad\quad (110x_6^3)-1.0\leqslant 0\\
&\qquad g_6=\sqrt{745x_5/(x_2x_3)^2+157.5\times 10^6}\,/\\
&\qquad\quad (85x_7^3)-1.0\leqslant 0\\
&\qquad g_7=x_2x_3-40.0\leqslant 0\\
&\qquad g_8=5.0-x_1/x_2\leqslant 0\\
&\qquad g_9=x_1/x_2-12.0\leqslant 0\\
&\qquad g_{10}=(1.5x_6+1.9)/x_4-1.0\leqslant 0\\
&\qquad g_{11}=(1.1x_7+1.9)/x_5-1.0\leqslant 0\\
&\qquad 2.6\leqslant x_1\leqslant 3.6,\quad 0.7\leqslant x_2\leqslant 0.8\\
&\qquad 17\leqslant x_3\leqslant 28,\quad 7.3\leqslant x_4\leqslant 8.3\\
&\qquad 7.3\leqslant x_5\leqslant 8.3,\quad 2.9\leqslant x_6\leqslant 3.9\\
&\qquad 5.0\leqslant x_7\leqslant 5.5
\end{aligned}\right\}\quad(12)$$

式中：g_1 为齿轮的最大弯曲应力；g_2 为接触应力约束；g_3 和 g_4 均为轴的横向最大挠度约束；g_5 和 g_6 均为轴内最大应力约束；g_7，g_8 和 g_9 为减速器的尺寸和空间限制条件；g_{10} 和 g_{11} 为计算轴尺寸的经验公式。根据标准 CO 的优化模型，可以将减速器设计优化问题分解为一个系统级和 3 个子系统级[16]。学科 1 由约束 g_1,g_2，g_7，g_8 和 g_9 组成，学科 2 由约束 g_1, g_2, g_4, g_6, g_7, g_8, g_9 和 g_{11} 组成，学科 3 由约束 g_1, g_2, g_3, g_5, g_7, g_8, g_9 和 g_{10} 组成。

文献[17]给出了减速器设计问题的优化结果：x_1=3.5，x_2=0.7，x_3=17，$f(x)$=2 996.170。本文借鉴文献[11]的划分方法，将 x_1，x_2，x_3 作为系统级变量，x_4，x_5，x_6，x_7 则划分为学科级变量。为了验证 FDRCO 方法在任何情况下都是可行的，本文选取减速器设计问题的 5 个具有代表性的初始点，在

https://dc-china-simulation.researchcommons.org/journal/vol30/iss1/12
DOI: 10.16182/j.issn1004731x.joss.201801012

第 30 卷第 1 期 系统仿真学报 Vol. 30 No. 1
2018 年 1 月 Journal of System Simulation Jan., 2018

可行域外的有初始点 1，初始点 2 和初始点 3，在可行域内的有初始点 4 和初始点 5。依次使用标准 CO 方法、动态松弛 CO 方法和 FERCO 方法对减速器优化设计问题进行优化。编程计算时，系统级与子系统级都采用序列二次规划法，各参数均取默认值。5 个初始点的优化结果如表 5~9 所示。

表 5　初始点 1 的优化结果
Tab.5　Optimal solution of initial points 1

优化方法	初始点	最优解	目标函数值	约束最大值	迭代次数
标准 CO	0, 0, 0	3.276 2, 0.700 0, 17.000 0	2 907.40	0.060 1	40
动态松弛 CO	0, 0, 0	3.495 6, 0.712 4, 17.020 8	2 997.47	0.050 1	69
FDRCO	0, 0, 0	3.490 1, 0.700 0, 17.000 0	2 992.10	0.000 1	42

表 6　初始点 2 的优化结果
Tab.6　Optimal solution of initial points 2

优化方法	初始点	最优解	目标函数值	约束最大值	迭代次数
标准 CO	3, 1, 30	3.500 0, 0.700 0, 27.897 5	5 217.8	0.001 2	187
动态松弛 CO	3, 1, 30	3.438 0, 0.700 0, 17.000 8	2 971.4	0.002 3	76
FDRCO	3, 1, 30	3.490 1, 0.700 0, 17.000 0	2 992.1	0.000 1	44

表 7　初始点 3 的优化结果
Tab.7　Optimal solution of initial points 3

优化方法	初始点	最优解	目标函数值	约束最大值	迭代次数
标准 CO	8, 2, 22	3.587 7, 0.712 4, 22.004 8	4 179.9	0.000 0	124
动态松弛 CO	8, 2, 22	3.601 2, 0.700 1, 17.003 1	2 956.2	0.010 2	58
FDRCO	8, 2, 22	3.490 1, 0.700 0, 17.000 0	2 992.1	0.000 1	37

表 8　初始点 4 的优化结果
Tab.8　Optimal solution of initial points 4

优化方法	初始点	最优解	目标函数值	约束最大值	迭代次数
标准 CO	3.5, 0.7, 17	3.501 0, 0.700 0, 17.002 0	2 996.0	0.000 0	9
动态松弛 CO	3.5, 0.7, 17	3.500 0, 0.700 0, 17.000 0	2 876.0	0.000 0	1
FDRCO	3.5, 0.7, 17	3.495 6, 0.700 0, 17.000 0	2 997.4	0.000 0	25

表 9　初始点 5 的优化结果
Tab.9　Optimal solution of initial points 5

优化方法	初始点	最优解	目标函数值	约束最大值	迭代次数
标准 CO	3.0, 0.75, 17	3.500 0, 0.700 0, 17.000 8	2 993.7	0.000 2	8
动态松弛 CO	3.0, 0.75, 17	3.000 0, 0.750 0, 17.000 0	2 967.5	0.000 1	1
FDRCO	3.5, 0.70, 17	3.495 6, 0.700 0, 17.000 0	2 997.4	0.000 0	25

由表 5~7 的优化结果可知，标准 CO 方法将可行域外的 3 个初始点优化到了靠近初始点的局部极值点。动态松弛 CO 方法将可行域外的 3 个初始点优化到了靠近全局最优解的局部极值点。FDRCO 方法将可行域外的 3 个初始点优化到了全局最优解附近。由表 8 和表 9 的优化结果可知，标准 CO 方法将可行域内的 2 个初始点优化到了全局最优解附近。动态松弛 CO 方法对于可行域内的 2 个初始点都只进行了一次迭代优化，优化结果是初始点本身。FDRCO 方法将可行域内的 2 个初始点优化到了全局最优解附近。由此可知。标准 CO 方法对初始点选取敏感，对可行域内的初始点优化效果较好，对可行域外的初始点优化精度较差，且计算代价较大。动态松弛 CO 方法同样对初始点选取

敏感，对可行域内的初始点无法进行优化，对可行域外的初始点优化优化精度不高。FDRCO 方法将 5 个初始点都优化到了全局最优解附近，有较好的稳定性和较高的优化精度，同时，5 个初始点的优化迭代次数都较少，有较快的收敛速度。

以远离最优解的初始点(0, 0, 0)为例，使用 FDRCO 方法进行优化。目标函数 f 在加速收敛阶段和优化求解阶段的优化曲线分别如图 6 和图 7 所示。由图 6 和图 7 可知，在加速收敛阶段，目标函数经过 29 次迭代已收敛至全局目标最优解附近；在优化求解阶段，目标函数经过 13 次迭代收敛到全局目标最优解。由此可以看出 FDRCO 方法可以有效加快远离最优解的初始点的收敛速度，提高系统优化效率。

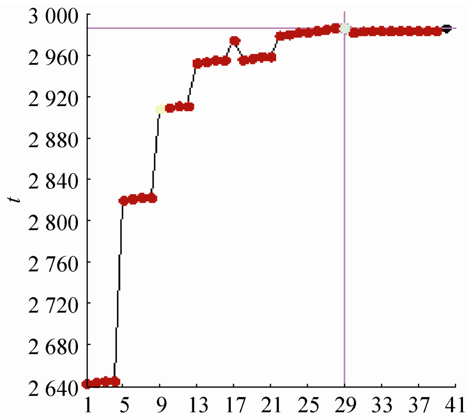

图 6 目标函数值 f 加速收敛阶段优化曲线
Fig.6 Optimized curve of objective function f at the accelerating convergence stage

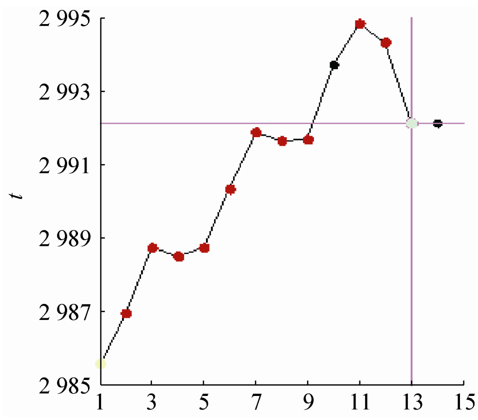

图 7 目标函数值 f 优化求解阶段优化曲线
Fig.7 Optimized curve of objective function f at the optimal solution stage

4 结论

本文针对当前 CO 优化过程中计算成本较大，远离最优解的初始点收敛速度较慢的问题，在前人研究松弛因子法的基础上对松弛因子的计算方法进行了改进，提出一种 FDRCO 优化方法。在加速收敛阶段，使用求均值的思想构造松弛因子，加快了远离最优解的初始点收敛到全局最优解附近区域的速度，降低了计算成本。在优化求解阶段，采用静态松弛因子法，以加速收敛阶段的最优解作为初始点进行优化，进一步加快了系统的收敛速度，增强了学科间的一致性，保证了该方法的可行性。最后，通过典型数值算例和减速器算例进一步验证了 FDRCO 方法的稳定性与有效性。

参考文献:

[1] 李海燕，马明旭，井元伟，等. 一种具有全局稳定性的多学科协同优化方法[J]. 计算机集成制造系统，2009, 15(12): 2363-2369.
Li Haiyan, Ma ingxu, Jing Yuanwei, et al. Improved multidisciplinary collaborative optimization with glob al stability[J]. Computer Intergrated Manufacturing Sy stems, 2009, 15(12): 2363-2369.

[2] Brian Roth, Ilan Kroon. Enhanced collaborative optimization: Applica- tion to an analytic test problem and aircraft design [C]//Proc of the 12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, USA: AIAA, 2008.

[3] Li Haiyan, Ma Mingxu, Zhang Wenlei. Improving co llaborative optimization for MDO problems with mult i-objective subsystems[J]. Structural and Multidisciplin ary Optimization(S1615-147X), 2014, 49(1): 609-620.

[4] Alexandru C Berbecea, Frederic Gillon. Multi-level design of an isolation transformer using collaborative optimization[J]. Computation and Mathematics in Electrical and Electronic Engineering(S0332-1649), 2014, 33(3): 1038-1050.

[5] 李响，李为吉. 基于超球近似子空间的协同优化方法及应用研究[J]. 西北工业大学学报，2003, 21(4): 461-464.
Li Xiang, Li Weiji. Collaborative Optimization Based on Inter-Disciplinary Inconsistency Information and its Application to Mechanical System Design[J]. Journal of Northwestern Polytechnical University, 2003, 21(4):

第 30 卷第 1 期 系统仿真学报 Vol. 30 No. 1
2018 年 1 月 Journal of System Simulation Jan., 2018

461-464.

[6] 李冬琴, 杨永祥, 陈智同. 一种改进的协同优化算法及其应用[J]. 计算机工程与科学, 2013, 35(1): 137-141.
Li Dongqin, Yang Yongxiang, Chen Zhitong. An improved collaborative optimization algorithm and its application[J]. Computer Engineering & Science, 2013, 35(1): 137-141.

[7] 方鹏亚, 常新龙, 胡宽, 等. 基于渐近松弛的协同优化方法[J]. 上海交通大学学报, 2013, 47(12): 1896-1899.
Fang Yapeng, Chang Xinlong, Hu Kuan, et al. Asymptotic Relaxation Based Collaborative Optimization [J]. Journal of Shang hai Jiao Tong University, 2013, 47(12): 1896-1899.

[8] Alexandrov N M, Lewis R M. Analytical and computational aspects of collaborative optimization for multidisciplinary design [J]. AIAA Journal (S0001-1452), 2002, 40(2): 301-309.

[9] 李海燕, 李永缙, 田英. 协同优化计算方法的分析与改进[J]. 沈阳大学学报(自然科学版), 2013, 25(6): 462-465.
Li Haiyan，Li Yongjin，Tian Ying. Analysis and Improvement on Algorithm of Collaborative Optimization [J]. Journal of Shenyang University (Natural Science), 2013, 25(6): 462-465.

[10] 郭健彬, 曾声奎. 自适应协同优化方法研究[J]. 系统工程与电子技术, 2009, 31(5): 1108-1112.
Guo Jianbin, Zeng Shengkui. Research on adaptive collaborative optimization method[J]. System Engineering and Electronics, 2009, 31(5): 1108-1112.

[11] 纪爱敏, 殷旭. 基于自适应松弛因子的协同优化方法[J]. 计算机集成制造系统, 2014, 20(7): 1530-1536.
Ji Aimin, Yin Xu. Collaborative optimization based on adaptive relaxation[J]. Computer Integrated Manufacturing Systems, 2014, 20(7): 1530-1536.

[12] 郭健彬, 曾声奎, 陈云霞. 稳健协同优化方法的改进和应用[J]. 火力与指挥控制, 2010, 35(4): 32-35.
Guo Jianbin, Zeng Shengkui, Chen Yunxia. Improvement and Application of Multidisciplinary Robust Design Optimization Method [J]. Fire Control& Command Control, 2010, 35(4): 32-35.

[13] 凌昊, 程远胜, 刘均, 等. 一种新的多学科协同优化方法及其工程应用[J]. 中国造船, 2011, 52(2): 87-99.
Ling Hao, Cheng Yuansheng, Liu Jun, et al. An Improved Multidiscipline Collaborative Optimization Algorithm and Its Application in Submarine Subdivision[J]. SHIPBUILDING OF CHINA, 2011, 52(2): 87-99.

[14] 温庆国, 宋保维, 王鹏. 基于 iSIGHT 软件的协同优化算法若干问题研究[J]. 西北工业大学学报, 2013, 31(1): 145-149.
Wen Qingguo, Song Baowei, Wang Peng. Some Problems of Collaborative Optimization Based on iSIGHT[J]. Journal of Northwestern Polytechnical University, 2013, 31(1): 145-149.

[15] Braun R D, Gage P, Kroo I. Implementation and performance issues in collaborative optimization[R]. AIAA-96-4017, USA: AIAA, 1996.

[16] 韩明红, 邓家禔. 协同优化算法的改进[J]. 机械工程学报, 2006, 42(11): 34-38.
Han Minghong, Deng Jiazhi. Improvement of Collaborative Optimization[J]. Chinese Journal of Mechanical Engineering, 2006, 42(11): 34-38.

[17] Azarm S, LI W C. Optimality and constrained derivatives in tow-level design optimization[J]. ASME Journal of Mechanical Design (S1050-0472), 1990, 112(12): 563-568.