

基于 Kriging 模型的改进协同优化算法

张 静^{1,2}, 李柏林³, 张卫华¹, 刘永均³

(1. 西南交通大学牵引动力国家重点实验室, 四川 成都 610031; 2. 成都理工大学核技术与自动化工程学院, 四川 成都 610059; 3. 西南交通大学机械工程学院, 四川 成都 610031)

摘要: 为了提高协同优化算法的求解效率, 利用 Kriging 模型, 构造系统级近似优化模型, 提出了基于 Kriging 模型的改进协同优化算法. 该算法采用置信域与均匀设计相结合的方法, 完成近似模型的更新; 采用序列二次规划算法, 完成优化问题的求解. 以经典函数和减速器设计为例, 验证了改进协同优化算法. 结果表明: 该算法能提高计算效率, 在减速器设计中, 迭代次数减少 50% 左右.

关键词: 多学科设计优化; 协同优化; Kriging 模型

中图分类号: TH123.1 **文献标识码:** A

Improved Collaborative Optimization Algorithm

ZHANG Jing^{1,2}, LI Bailin³, ZHANG Weihua¹, LIU Yongjun³

(1. Traction Power State Key Laboratory, Southwest Jiaotong University, Chengdu 610031, China; 2. School of Nuclear Technology and Automation, Chengdu University of Technology, Chengdu 610059, China; 3. School of Mechanical Engineering, Southwest Jiaotong University, Chengdu 610031, China)

Abstract: In order to improve the computational efficiency of the conventional collaborative optimization (CO), an improved collaborative optimization algorithm based on the Kriging model (Kriging-CO for short) was proposed. In this algorithm, the approximate optimization model at the system level is constructed with the Kriging model, and is updated by uniform design combined with confidence regions. The Kriging-CO was verified through the optimization of a classical function and the design of a speed reducer, and the optimization models were solved using the sequential quadratic programming algorithm. Numerical results show that the Kriging-CO can improve the computational efficiency. For the complex speed reducer design, the Kriging-CO, compared with the CO, reduced the number of iterations in the optimization computation by about 50%.

Key words: multidisciplinary design optimization; collaborative optimization; Kriging model

近似技术是多学科设计优化 (multidisciplinary design optimization, MDO) 中必不可少的环节, 国内外众多学者对此进行了深入研究, 李响等^[1] 提出基于超球近似子空间的协同优化方法, 并应用于减速箱的设计; 白小涛等^[2] 采用基于均匀设计的二次响应面技术得到近似的协同优化模型, 完成了机翼的设计优化; 韩永志等^[3] 基于 Kriging 模型的近似技术, 建立了三维涡轮叶片的 MDO 方法; 宋保

维等^[4] 采用子空间近似算法, 实现鱼雷的总体多学科设计. 在 MDO 中, 采用近似技术, 可以减少求解 MDO 问题的计算量, 降低计算成本, 提高优化效率^[5-6]. 因此, 本文采用 Kriging 模型的近似技术, 对协同优化算法 (collaborative optimization, CO) 中计算量大的问题, 进行 CO 算法改进研究, 以便提高计算效率.

为了提高协同优化算法的求解效率, 分析了协

同优化算法中影响求解效率的因素,采用均匀设计的实验方法,选取近似模型样本点,利用 Kriging 模型,构造 CO 算法中系统级约束的近似模型,将置信域与均匀设计相结合,完成近似模型的更新,提出了基于 Kriging 模型的改进 CO 算法.

1 影响 CO 算法求解效率的因素

CO 算法将复杂的系统设计分解为 1 个系统级和多个并行的学科级,通过系统级约束条件,协调各学科之间的共享设计变量和耦合状态变量. 在 CO 算法中,系统级优化模型为^[7]

$$\left. \begin{aligned} \text{find: } & \mathbf{Z} = (z_1, z_2, \dots, z_n), \\ \text{min: } & F(\mathbf{Z}), \\ \text{s. t.: } & J_i^*(\mathbf{Z}) = 0, \end{aligned} \right\} \quad (1)$$

式(1)中: \mathbf{Z} 为系统级设计向量; n 为系统级设计变量数; i 为学科数, $i=1, 2, \dots, n$; J_i^* 为学科 i 的优化目标函数,即学科间的一致性约束.

在 CO 算法中,通过调整 \mathbf{Z} 值,系统层优化器提高了整体数据处理效率. 在系统变量取值时,需满足所有学科一致的约束条件.

系统层优化模型存在如下问题:(1) 计算系统层优化模型需要多次运行学科层优化器,这是由于在运行系统层优化器时,每次优化迭代所需的 J_i^* 值,均需要运行 1 次学科层优化器得到,即系统层优化迭代多少次,学科层优化器就运行多少次,增加了计算量,影响系统层优化器的计算效率;(2) 在系统变量取值时,满足所有学科的约束条件 $J_i^* = 0$,系统层和学科层之间需要多次数据交换,增加了协调难度.

2 Kriging 模型的构造

在 CO 算法中,系统级传给各学科级 n 组变量值,各学科级优化后,得到 n 个最优目标函数值. 根据 n 组变量值和各学科的 n 个最优目标函数值构造 Kriging 模型,得到各学科级目标函数的近似模型.

Kriging 模型的构造过程如下.

假设系统有 1 个系统级和 2 个学科(子系统级)优化模型,系统级传给 2 个学科级目标变量值为 (Z_1, Z_2, \dots, Z_n) , 2 个学科级优化后,得到学科 1 和学科 2 的目标函数值分别为 $J_1^* = (j_{11}^*, j_{12}^*, \dots, j_{1n}^*)$ 和 $J_2^* = (j_{21}^*, j_{22}^*, \dots, j_{2n}^*)$, 那么,2 个学科最优目标函数值的 Kriging 模型为

$$J_1^* = \beta_1 + \mathbf{r}_1^T(\mathbf{Z})\mathbf{R}_1^{-1}(\mathbf{y}_1 - \mathbf{I}\beta_1), \quad (2)$$

$$J_2^* = \beta_2 + \mathbf{r}_2^T(\mathbf{Z})\mathbf{R}_2^{-1}(\mathbf{y}_2 - \mathbf{I}\beta_2), \quad (3)$$

式中: β_1 和 β_2 为常数; $\mathbf{r}_1^T(\mathbf{Z})$ 、 $\mathbf{r}_2^T(\mathbf{Z})$ 为样本点和预测点所组成的相关变量; \mathbf{R}_1^{-1} 、 \mathbf{R}_2^{-1} 为样本点相关矩阵; \mathbf{y}_1 、 \mathbf{y}_2 为长度列向量; \mathbf{I} 为单位列向量.

系统级向 2 个学科传递每组目标变量 (Z_1, Z_2, \dots, Z_n) 的值作为 1 个样本点集合,迭代过程中进行样本点的选取及近似模型更新. 为了提高计算的准确度,采用置信域与均匀设计相结合的方法,完成样本点的选取与近似模型的更新.

3 改进 CO 算法的步骤

在构造系统级约束 Kriging 模型的基础上,提出了基于 Kriging 的改进 CO 算法,即 Kriging-CO 算法. Kriging-CO 算法的计算流程如图 1 所示.

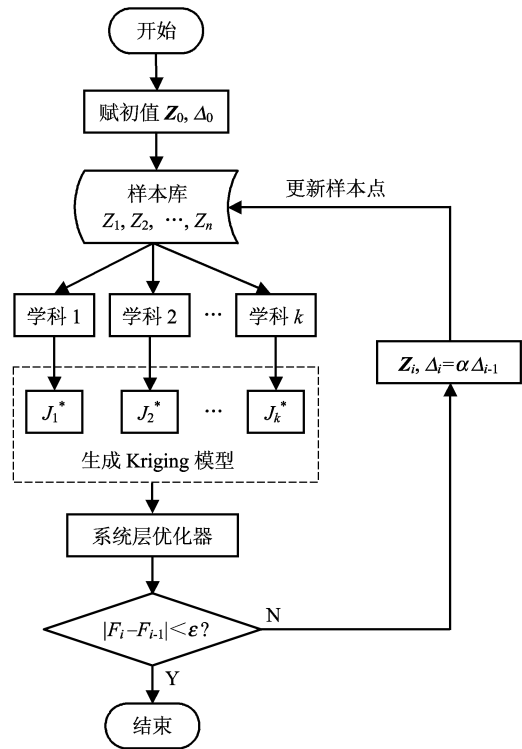


图 1 Kriging-CO 算法的计算流程
Fig. 1 Flow chart of Kriging-CO

具体步骤如下:

- (1) 系统层所有共享设计变量赋初值为 \mathbf{Z}_0 , 给定初始置信域为 Δ_0 ;
- (2) 传递共享变量值给学科层, 构造 Kriging 模型,
 - ① 根据初值 \mathbf{Z}_0 , 初始置信域 Δ_0 , 查均匀设计表, 获取一定数目样本点 (Z_1, Z_2, \dots, Z_n) ,
 - ② 运行学科层优化器得到学科 k 对应的目标

函数值($j_{k1}, j_{k2}, \dots, j_{kn}$),

③ 根据①、②数据,生成 Kriging 模型;

(3) 运行系统层优化器,求得此次迭代的目标值 F_i 和系统变量值 Z_i ,在迭代过程中,系统级一致性约束 J_k^* 由 Kriging 模型给出;

(4) 若 F_i 在相邻两次的差值绝对值小于规定正数 ε ,即 $|F_i - F_{i-1}| < \varepsilon$,则整个系统收敛,转到(7),否则执行(5);

(5) 缩小置信域, $\Delta_i = \alpha\Delta_{i-1}$ (α 为试验点范围的变化系数), $0 < \alpha < 1$;

(6) 取出此次迭代变量值 Z_i ,更新样本点,存入样本库,丰富样本,转到(2);

(7) 结束,输出系统的全局优化解.

在 Kriging-CO 算法中,每步迭代过程完成 1 次系统级优化,得到 1 个新的系统级设计变量最优解 Z^* ,以 Z^* 为基准确定 1 个样本点,选取范围为 Δ ,按照均匀设计原则,选取(Z_1, Z_2, \dots, Z_n)作为样本点下次迭代值的集合.迭代过程中可以改变 Δ ,以加快收敛速度.

4 实例

4.1 经典函数优化

采用文献[8]中的优化模型,应用 CO 算法和 Kriging-CO 算法求解.

表 1 标准 CO 算法(约束松弛)和 Kriging-CO 算法结果

Tab.1 Optimization results by CO and Kriging-CO

起始点	CO 算法			Kriging-CO 算法		
	最优解	目标值	迭代次数	最优解	目标值	迭代次数 n
(1, 1)	(0.197 5, 1.977 1)	3.947 8	15	(0.197 8, 1.994 2)	4.016 1	8
(4, -1)	(0.199 3, 1.976 9)	3.947 8	14	(0.198 7, 1.994 1)	4.016 1	29
(0.8, 1.5)	(0.197 7, 1.977 1)	3.947 8	10	(0.198 7, 1.994 3)	4.016 8	10
(2, 3)	(0.197 9, 1.977 0)	3.947 8	20	(0.197 6, 1.992 5)	4.009 3	11

由表 1 可以看出,对于不同初始点,CO 算法和 Kriging-CO 算法均收敛到最优解附近.当起始点为(4, -1)时, Kriging-CO 算法进行迭代的次数多于 CO 算法.这是因为经典函数的子学科模型不复杂,学科级优化能够在短时间内完成,系统级和子系统级间的信息传递量没有造成系统级的计算负担.

4.2 工程算例

以文献[11]中的减速器为例,设计目标为在满足减速器中转轴和齿轮约束的同时,使减速器体积最小.用 CO 算法和 Kriging-CO 算法求解时,将该优化问题分解为 1 个系统级优化模型和 3 个学

系统级优化模型为

$$\left. \begin{aligned} \min: & f(z) = z_1^2 + z_2^2, \\ \text{s. t.} & J_1^* = (z_1 - x_1)^2 + (z_2 - x_2)^2 = 0, \\ & J_2^* = (z_1 - x_1)^2 + (z_2 - x_2)^2 = 0. \end{aligned} \right\} \quad (4)$$

学科 1 优化模型为

$$\left. \begin{aligned} \min: & J_1 = (z_1 - x_1)^2 + (z_2 - x_2)^2, \\ \text{s. t.} & x_1 + \beta x_2 < 4. \end{aligned} \right\} \quad (5)$$

学科 2 优化模型为

$$\left. \begin{aligned} \min: & J_2 = (z_1 - x_1)^2 + (z_2 - x_2)^2, \\ \text{s. t.} & \beta x_1 + x_2 > 2. \end{aligned} \right\} \quad (6)$$

上述优化模型中:

z_1 和 z_2 ——系统级设计变量;

J_1^* 和 J_2^* ——系统级一致性等式约束;

x_1 和 x_2 ——学科设计变量;

β ——可变参数.

取 $\beta = 0.1$,分别用标准 CO 算法和 Kriging-CO 算法对上述优化模型求解,其中,子系统和系统级的优化方法均采用序列二次规划(sequential quadratic programming method, SQP)算法. Kriging 模型中参数优化由遗传算法^[9-10]完成.选取 4 个起始点进行计算,2 种方法得到的优化结果如表 1 所示.

科级优化模型.

系统级优化模型为

$$\left. \begin{aligned} \text{find:} & z_1, z_2, z_3, \\ \min: & f = f_1 + f_2 + f_3, \\ \text{s. t.} & J_1^* = (x_1 - z_1)^2 + (x_2 - z_2)^2 + \\ & (x_3 - z_3)^2 = 0, \\ & J_2^* = (x_1 - z_1)^2 + (x_2 - z_2)^2 + \\ & (x_3 - z_3)^2 = 0, \\ & J_3^* = (x_1 - z_1)^2 + (x_2 - z_2)^2 + \\ & (x_3 - z_3)^2 = 0, \end{aligned} \right\} \quad (7)$$

式中： f_1, f_2 和 f_3 ——学科级目标函数；
 z_1, z_2 和 z_3 ——系统级设计变量；
 J_1^*, J_2^* 和 J_3^* ——系统级一致性等式约束。
 学科 1 优化模型为
 find: $x_1, x_2, x_3,$
 min: $J_1 = (x_1 - z_1)^2 + (x_2 - z_2)^2 + (x_3 - z_3)^2,$
 s. t. $f_1 = 0.785 4x_1x_2^2 (3.333 3x_3^2 + 14.933 4x_3 - 43.093 4),$
 $g_j - 1.0 \leq 0, j = 1, 2, 7, 8, 9,$

(8)

式中： x_2, x_2 和 x_3 为学科 1 设计变量。
 学科 2 优化模型为
 find: $x_1, x_2, x_3, x_4, x_6,$
 min: $J_2 = (x_1 - z_1)^2 + (x_2 - z_2)^2 + (x_3 - z_3)^2,$
 s. t. $f_2 = 7.477 0x_7^3 + 0.785 4x_5x_7^2 + 1.508 0x_7^2x_1,$
 $g_j - 1.0 \leq 0, j = 1, 2, 4, 6, 7, 8, 9, 11,$

(9)

式中： x_1, x_2, x_3, x_4 和 x_6 为学科 2 设计变量。
 学科 3 优化模型为
 find: $x_1, x_2, x_3, x_5, x_7,$
 min: $J_3 = (x_1 - z_1)^2 + (x_2 - z_2)^2 + (x_3 - z_3)^2,$
 s. t. $f_3 = 7.477 0x_6^3 + 0.785 4x_4x_6^2 - 1.508 0x_6^2x_1,$
 $g_j - 1.0 \leq 0,$
 $j = 1, 2, 3, 5, 7, 8, 9, 10,$

(10)

式中： x_1, x_2, x_3, x_5 和 x_7 为学科 3 设计变量。
 文献[11]中给出 4 个不同起始设计点,如表 2 所示,其中： X_1, X_2 和 X_3 为可行域内的设计点； X_4 为可行域外的设计点. 分别用标准 CO 算法和 Kriging-CO 算法优化求解, 2 种方法得到的优化结果如表 3 和表 4 所示。

表 2 起始设计点^[11]
 Tab.2 Initial design points^[11]

起始点	齿面宽度 x_1/cm	齿轮模数 x_2/cm	小齿轮模数 x_3/cm	轴承间距 x_4/cm	轴承间距 x_5/cm	小齿轮轴直径 x_6/cm	大齿轮轴直径 x_7/cm
X_1	2.80	0.71	25.0	7.90	7.599	3.00	5.099
X_2	2.65	0.63	18.0	6.80	6.400	3.00	5.099
X_3	3.50	0.70	17.0	7.30	7.715	3.35	5.287
X_4	2.00	0.40	9.00	4.00	4.500	1.50	3.000

表 3 标准 CO 算法(引入罚函数)优化结果
 Tab.3 Optimization results by CO(with penalty function introduced)

起始点	最优解/cm	目标值/ cm^3	迭代次数
X_1	(3.505 1, 0.683 3, 16.999 9, 7.300 1, 7.713 9, 3.300 1, 5.285 4)	2 991.37	33
X_2	(3.440 8, 0.699 7, 17.000 0, 7.300 1, 7.713 9, 3.300 1, 5.285 4)	2 991.77	32
X_3	(3.486 0, 0.687 5, 16.999 1, 7.300 1, 7.713 9, 3.300 1, 5.285 4)	2 991.76	17
X_4	(3.487 2, 0.678 8, 16.998 9, 7.300 1, 7.713 9, 3.300 1, 5.285 4)	2 991.80	30

表 4 Kriging-CO 算法优化结果
 Tab.4 Optimization results by Kriging-CO

起始点	最优解/cm	目标值/ cm^3	迭代次数
X_1	(3.486 2, 0.703 0, 17, 7.300 1, 7.713 9, 3.300 1, 5.285 4)	2 991.8	15
X_2	(3.486 0, 0.600 3, 17, 7.300 1, 7.713 9, 3.300 1, 5.285 4)	2 991.7	16
X_3	(3.480 8, 0.700 0, 17, 7.300 1, 7.713 9, 3.300 1, 5.285 4)	2 991.7	6
X_4	(3.487 1, 0.700 0, 17, 7.300 1, 7.713 9, 3.300 1, 5.285 4)	2 991.4	16

由表 3 和表 4 可以看出,标准 CO 算法和 Kriging-CO 算法均能收敛到最优解附近,但 Kriging-CO 算法的优化计算迭代次数明显减少,起始点为 X_3 时,迭代次数减少明显,从原来的 17 次减少到 6 次,迭代计算量减少了 70%。

5 结论

(1) 采用 Kriging 模型近似计算 J_i^* ,可避免在系统层优化过程中频繁调用学科层优化器来获取 J_i^* 值,只有当 Kriging 模型重新创建时才运行学科

层优化器,减少了学科级优化的次数;

(2) 采用置信域与均匀设计相结合的方法选取更新近似模型的样本点,使得设计过程中获得的设计变量更接近最优解的样本点,节省了运算时间;

(3) 算例的计算结果,验证了 Kriging-CO 算法的有效性.在经典函数算例中,当子学科模型不复杂时,Kriging-CO 算法与标准 CO 算法的迭代次数基本相同;在模型复杂的减速器算例中,当采用 Kriging 模型近似系统级一致性约束时,Kriging-CO 算法的计算迭代次数大都减少了 50% 左右,最优情况下能减少 70% 的计算量.因此,Kriging-CO 算法适用于解决共享变量多、子模型复杂的问题.

参考文献:

- [1] 李响,李为吉. 基于超球近似子空间的协同优化方法及其应用研究[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): 461-464.
- [2] 白小涛,李为吉. 基于近似技术的协同优化方法在机翼设计优化中的应用[J]. 航空学报,2006,27(5): 847-850.
BAI Xiaotao, LI Weiji. Application of collaborative optimization based on approximate methods in wing design optimization[J]. Acta Aeronautica et Astronautica Sinica, 2006, 27(5): 847-850.
- [3] 韩永志,高行山,李立州,等. 基于 Kriging 模型的涡轮叶片多学科设计优化[J]. 航空动力学报,2007,22(7): 1055-1059.
HAN Yongzhi, GAO Hangshan, LI Lizhou, et al. Kriging model-based multidisciplinary design optimization for turbine blade[J]. Journal of Aerospace Power, 2007, 22(7): 1055-1059.
- [4] 宋保维,黎华,毛昭勇. 基于子空间近似算法的鱼雷总体多学科设计[J]. 火力与指挥控制,2008,33(6): 146-148.
SONG Baowei, LI Hua, MAO Zhaoyong. Torpedo conceptual multidisciplinary design based on SAO[J]. Fire Control and Command Control, 2008, 33(6): 146-148.
- [5] 王振国,陈小前,罗文彩,等. 飞行器多学科设计优化理论与应用研究[M]. 北京:国防工业出版社,2006: 110-111.
- [6] 颜力. 飞行器多学科设计优化若干关键技术的研究与应用[D]. 长沙:国防科技大学,2006.
- [7] BRAUN R D, KROO I. Use of the collaborative optimization architecture for launch vehicle design [C] // AIAA Paper 96-4018, 6th AIAA/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization. Washington: [s. n.], 1996: 306-318.
- [8] BRAUN R D, GAGE P, KROO I, et al. Implementation and performance issues in collaborative optimization[C]//AIAA 96-4017. Washington: AIAA, 1996: 1-11.
- [9] 吴广宁,付龙海,王颢,等. 基于改进遗传算法的接地网优化设计[J]. 西南交通大学学报,2007,42(2): 169-174.
WU Guangning, FU Longhai, WANG Hao, et al. Optimal design of grounding grid based on improved genetic algorithm [J]. Journal of Southwest Jiaotong University, 2007, 42(2): 169-174.
- [10] 张洪海,胡明华,陈世林. 机场终端区容量利用和流量分配协同优化策略[J]. 西南交通大学学报,2009,44(1): 128-134.
ZHANG Honghai, HU Minghua, CHEN Shilin. Collaborative optimization of capacity utilization and flow assignment in airport terminal area[J]. Journal of Southwest Jiaotong University, 2009, 44(1): 128-134.
- [11] KODIYALAM S. Evaluation of methods for multidisciplinary design optimization (MDO), phase I, NASA CR-208716[R]. Hampton: NASA, 1998.

(中文编辑:秦瑜 英文编辑:兰俊思)