

基于SQP的双层单目标优化算法

许裕楷

广东工业大学数学与统计学院, 广东 广州

收稿日期: 2024年2月8日; 录用日期: 2024年2月28日; 发布日期: 2024年4月22日

摘要

近年来, 进化算法已被证明是一种能有效处理双层优化问题的算法。在双层优化问题的嵌套结构中, 下层优化的效率以及优化所需的大量计算资源, 已经成为处理双层优化问题要面临的巨大挑战。因此, 本文提出了一种基于SQP方法的双层优化算法(SQPEA)来处理双层单目标优化问题。在SQPEA算法中, 我们引入了非线性优化算法SQP来对下层优化进行预优化处理, 将预优化的最优解作为下层优化的初始解, 并利用最优梯度向量来缩小下层搜索空间, 提高下层优化效率, 进而提高上层优化效率。最后, 在12个标准测试问题上与成熟的双层优化算法进行对比实验, 实验结果证明了所提算法在处理双层单目标问题上突出的性能表现。

关键词

双层优化, 进化算法, SQP

Bilevel Single-Objective Optimization Algorithm Based on SQP

Yukai Xu

School of mathematics and statistics, Guangdong University of Technology, Guangzhou Guangdong

Received: Feb. 8th, 2024; accepted: Feb. 28th, 2024; published: Apr. 22nd, 2024

Abstract

In recent years, evolutionary algorithm has been proved to be an efficient algorithm for solving bilevel optimization problems. In the nested structure of the bilevel optimization problem, the efficiency of the lower lower optimization and the large amount of computational resources required for the optimization have become a great challenge to deal with the bilevel optimization problem. Therefore, this paper proposes a bilevel optimization algorithm based on the SQP method (SQPEA) to deal with bilevel single-objective optimization problems. In the SQPEA algorithm,

文章引用: 许裕楷. 基于 SQP 的双层单目标优化算法[J]. 运筹与模糊学, 2024, 14(2): 634-641.

DOI: 10.12677/orf.2024.142166

we introduce the nonlinear optimization algorithm SQP to pre-optimize the lower optimization, and take the optimal solution of the pre-optimization as the initial solution of the lower optimization, and use the optimal gradient vector to reduce the search space of the lower optimization and improve the efficiency of the lower level optimization, then improve the optimization efficiency of the upper level. Finally, comparing with a mature bilevel optimization algorithm on 12 standard test problems, the experimental results show that the proposed algorithm has outstanding performance in dealing with bilevel single-objective problems.

Keywords

Bilevel Optimization, Evolutionary Algorithm, SQP

Copyright © 2024 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

人类社会本质在就是由多个层级共同构成的结构，最高级的决策者依次向下管理着其他层级，同时各层级内又自主决策管理。因此，各层级都有着自己的发展目标，都在追求自身利益最大化，但往往这些目标是相互矛盾的，故产生了对多层次规划问题的研究。其中，考虑到现实问题的复杂性和离散性，对双层优化问题的应用和研究更为普遍和广泛。双层优化问题存在于各行各业，例如在化学工程，路线规划等行业。对双层优化的研究也成为行业发展的重要因素。

双层优化问题涉及到两个层级的优化问题，但两个层级内实现各自利益最大化的决策往往是相互矛盾的，因此，求解双层优化问题的过程实际上是在两个层级的相互博弈下达到的共同最优。从结构上看，上层与下层之间属于嵌套关系，执行上层优化前需要先完成其下层优化，这需要消耗大量下层计算资源，这使得处理双层优化问题成为一个 NP-Hard 的任务。一旦下层问题涉及到多模态、欺骗性或病态等特征时，会给求解双层优化问题带来巨大的挑战。而双层单目标问题则是其中一类典型的问题。

在过去十多年里，越来越多的学者提出了各种有效的双层优化算法，尝试提高双层优化的精度表现，或者减少下层计算资源的消耗。随着计算技术的发展，人们面临的双层优化问题越来越复杂，而计算机性能的高速发展为处理这些问题提供了可能性，所以对双层优化的研究是具有巨大价值的。

2. 国内外研究现状

自提出双层优化问题的定义至今，各类双层优化算法层出不穷，根据作用原理大致可分为三类[1]。

第一类是基于简化的方法。这类方法是通过一些方法将双层优化问题简化为单层优化问题，并利用经典数学方法求解简化后的优化问题。例如，王云用罚函数法将双层问题简化为单层问题并求解[2]。还有单纯形法和分支定界法等[3][4]。这类方法虽然可以有效降低双层优化问题的求解难度，但十分依赖问题的连续性、可微等性质，不适合处理离散、复杂的现实问题，更多应用于理论研究。

第二类是基于代理模型的方法。这类方法通过拟合上层变量与下层变量间的映射关系，来提高下层优化的效率。如 Sinna 等人[5]利用克里金模型来拟合上下层变量间的函数关系，并引入了局部搜索策略，提高算法性能。另外，模型的选择也是多样的[6][7]。但是模型的设计十分考验研究者的经验，且与优化问题密切相关，无法推广到其他双层优化问题。其次，由于代理模型的近似性质，近似解与真实解间不

可避免存在误差。随着迭代次数增加, 误差不断积累, 最后必然会导致近似最优解与真实最优解之间存在较大误差, 精度表现不佳。

第三类方法是基于进化算法的方法(EA)。主要思想是利用进化算法来处理上层和下层优化问题。近年来, 进化算法已被证明是一种有效处理双层优化问题的方法, 主要有两种作用方式。一种是用 EA 处理上层问题, 用确定性算法处理下层问题。如 Yin 等人使用梯度下降法处理下层问题[8]。这类方法虽然效率高, 但得到局部最优的概率较大。另一种方式是用 EA 同时处理上下层问题。如在[9]中, 使用粒子群算法处理上下层优化问题。得益于 EA 的全局性优势, 进化算法虽然可以有效处理双层问题, 但需要消耗大量计算资源, 尤其在嵌套结构中, 所需的计算资源会成倍增加。后来, He 等[10]通过总结分析过往 EA 的特点, 指出充分利用上下层变量间的相关信息对提高算法性能的重要性, 并提出了一种基于自适应协方差矩阵的双层优化算法(BLCMAES), 利用协方差实现上下层间的信息共享, 提高下层优化效率。

因此, 如今双层优化的主流研究方向是将双层进化算法与其他的方法结合起来, 取长补短, 既能发挥经典算法计算速度快和消耗资源少的优势, 又能发挥双层进化算法全局性和效率优势。

3. 基于 SQP 的双层优化算法

3.1. 双层优化问题

首先, 双层单目标优化问题的数学模型为:

$$\begin{aligned} & \min_{x_u \in X_u, x_l \in X_l} F(x_u, x_l) \\ x_l \in \arg \min & \{f_j(x_u, x_l) : g_j(x_u, x_l) \leq 0, j = 1, 2, \dots, J\}, \\ & G_k(x_u, x_l) \leq 0, k = 1, 2, \dots, K. \end{aligned} \quad (1)$$

其中, $G_k : X_u \times X_l \rightarrow R$ 表示上层第 k 个约束条件, $g_j : X_u \times X_l \rightarrow R$ 表示下层第 j 个约束条件。上式清晰地展现了双层优化问题的嵌套结构。

在双层优化中, 对任意上层变量都要求解其下层最优解, 而随着迭代代数增加, 计算资源成倍的增长, 给双层优化带来了巨大的挑战。因此, 当下层搜索的效率得到提高时, 整个双层优化算法的效率也会显著提高。针对这一点, 我们提出使用序列二次规划方法(SQP)对下层问题作预优化处理, 得到一个更有利于下层搜索的下层初始解, 提高下层搜索的效率。并与 BLCMAES 算法框架结合, 提出基于 SQP 方法的双层优化算法(SQPEA)。

3.2. 下层预优化

序列二次规划方法(SQP)是在二次规划算法 QP 的基础上改良所得。是目前求解非线性优化问题最有效的算法之一[11]。SQP 算法是在每一代的迭代点附近, 通过泰勒展开将非线性优化问题转化为二次规划问题来求解下一代的迭代点, 并逐步搜索最优解。具有收敛性好, 计算效率高, 边界搜索能力强的优点。因此, 我们将其作为下层预优化的算法, 得到下层优化的初始解和当前负梯度方向, 并以此来缩小搜索范围, 加快下层搜索速度。

SQP 算法的核心思想与牛顿迭代法相似, 作用原理图如下。当迭代点为 x_n 时, SQP 算法在迭代点附近将原问题作二次泰勒展开, 对其优化得到步长向量 d_n , 并得到下一代的迭代点 x_{n+1} 。如此循环往复, 直到找到非线性优化问题的最优解。整个优化过程如图 1 所示, 给定一个序列和初始迭代点, 按照迭代点的负梯度方向, 如图中虚线所示, 一步步地沿着目标值的下降方向趋近最优解。在这期间, 步长向量会自适应地变化, 使其更好地适应优化状况。

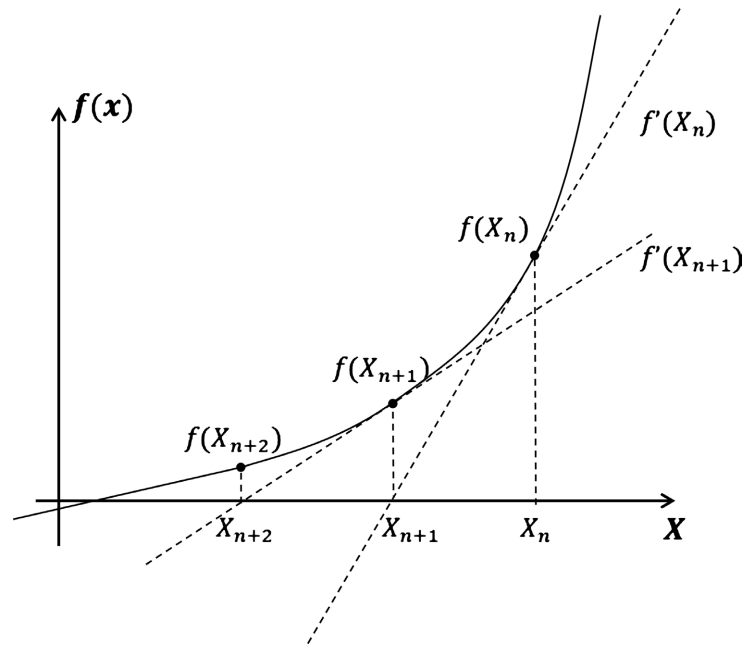


Figure 1. SQP method action schematic diagram
图 1. SQP 方法作用原理图

因此，下层预优化的具体操作如下。首先，在每一代下层优化前，将给定的上下层变量设为 X_u 和 X_l ， X_u 将作为参数输入到下层问题中，令 X_l 为 SQP 算法的初始迭代点，代入公式(2)中，得到优化后的 SQP 最优解 X_{l_SQP} 和最优梯度向量 $\nabla f(X_{l_SQP})$ ，其中 $\nabla f_i(X_{l_SQP})$ 为梯度向量分量， n 为下层变量维数。将 SQP 最优解 X_{l_SQP} 作为下层优化的初始解，使其更趋近下层最优解，更有利于快速搜索到下层最优。

$$\begin{aligned} [X_{l_SQP}, \nabla f(X_{l_SQP})] &= SQP(\text{Problem}, X_u, X_l) \\ \nabla f(X_{l_SQP}) &= (\nabla f_1(X_{l_SQP}), \nabla f_2(X_{l_SQP}), \dots, \nabla f_n(X_{l_SQP})) \end{aligned} \tag{2}$$

接着，根据负梯度优化的原理，结合最优梯度向量，我们进一步调整搜索范围，通过缩小搜索范围来帮助下层种群收敛。具体操作如式(3)所示。对任意 $\nabla f_i(X_{l_SQP})$ ，当它小于 0 时，表示负梯度方向指向上限，因此我们将该下层变量分量的下限调整为 SQP 最优解 X_{l_SQP} 。同理，当它大于 0 时，表示负梯度方向指向下限，将该下层变量分量的上限调整为 SQP 最优解 X_{l_SQP} 。

$$\begin{cases} \text{if } \nabla f_i(X_{l_SQP}) < 0, lb_i = X_{l_SQP}; \\ \text{else, } ub_i = X_{l_SQP}; \end{cases} \tag{3}$$

$$i = 1, 2, \dots, n$$

结合公式(2)、(3)后，我们通过下层预优化策略，使下层初始解更加靠近下层最优解，同时将下层搜索空间缩小，帮助下层种群快速收敛至下层最优解，从而达到减少下层计算资源，提高下层搜索效率的目的。另外，为了充分利用双层单目标问题中邻近上层变量具有相似下层最优解的性质，我们将每一代的下层最优解作为下一代 SQP 算法的初始迭代点，来减少 SQP 算法所需的计算资源，且提高整体下层搜索的效率。同时，考虑到 SQP 算法的评估次数仍然要计入总的下层评估次数中，我们限制它的最大评估次数为 $10n$ ， n 为下层变量维数，防止本末倒置。

4. SQPEA 算法流程框架

综合上文提到的两个优化策略，本文所提出的 SQPEA 算法的基本流程如图 2 所示。SQPEA 算法是以 BLCMAES 算法为框架，结合前文提出的三个策略所得。在 BLCMAES 算法框架中，上下两个层级都采用 CMAES 算法作为基本优化器。首先，初始化上层优化器的各项参数，并利用上层优化器中的分布矩阵参数生成一个维的上层种群，分别为上下层变量维数。之后利用协方差矩阵的边缘分布为下层优化提供先验信息。根据这些信息，执行下层预优化操作，将预优化得到的最优解作为下层初始解，用当前最优梯度向量来缩小下层搜索空间，如式(2)、(3)所示。再使用 CMAES 算法来进行下层优化，并将下层最优解返回到上层优化器中，与上层变量结合后进行评估。接着根据精英解的分布信息来更新上层优化器的各项参数。如此循环，直至满足双层优化的终止条件，输出双层最优解和其上下层目标函数值。

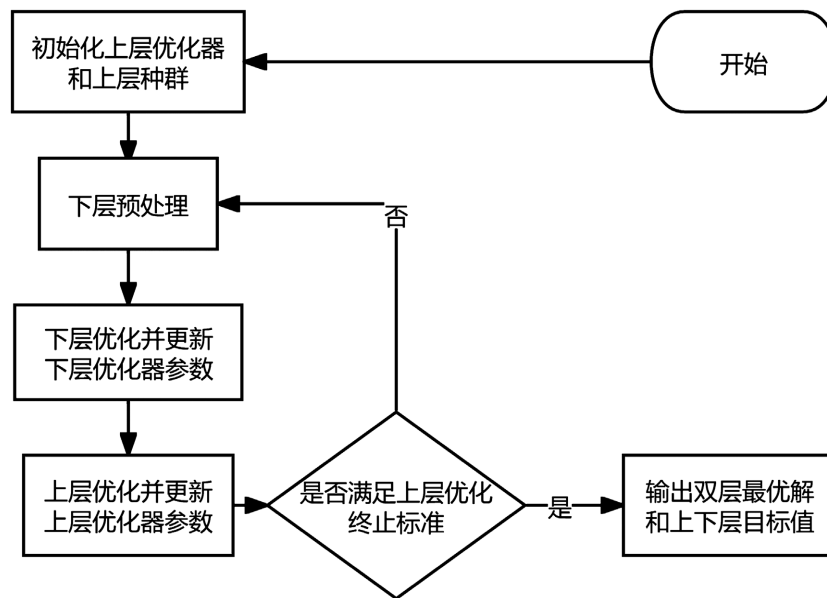


Figure 2. SQPEA algorithm flow chart
图 2. SQPEA 算法流程图

5. 仿真实验结果与分析

5.1. 标准测试问题与对比算法

为了验证 SQPEA 算法有效性和效率，本文选择文献[12]中提出的 SMD 测试问题集作为标准测试问题。SMD 测试集中包含了 12 个双层单目标问题，第 1~9 个测试问题不涉及约束条件，上下层的真实最优值都是 0；第 10~12 个测试问题涉及到约束条件，上下层真实最优值分别为(4, 3)，(-1, 1)和(3, 4)。SMD 测试集在双层单目标优化的研究被广泛使用，能够有效地测试算法的性能。

对比算法我们则选择 BLCMAES 算法，以此来检验本文提出的两个策略的作用和对提高算法性能的帮助，以此来证明 SQPEA 算法的性能和优势。

5.2. 参数设置与评价指标

本文选取的评价指标为上下层函数精度和上下层评估次数。上层函数精度为 $|F - F^*|$ ，下层函数精度为 $|f - f^*|$ ， F^* 和 f^* 分别为上下层的真实最优目标值。而上下层函数最小精度都为 1.00E-06。上层最大

评估次数为 3500，下层最大评估次数为 350。为了避免局部死循环，当上层函数精度保持连续 50 代不变时停止优化，下层则是连续 35 代。在测试问题集的维数上，我们选择了中等维数，使其能兼顾到低维和高维的特点。上层变量和下层变量的维数分别为 $m = 5, n = 5$ 。上层种群规模 $N_u = 9$ ，下层种群规模 $N_l = 6$ 。

5.3. 实验结果分析

为了保证实验结构的真实有效，我们设置了 19 次独立运行，并取 19 次实验结果的中位数来作为实验最终的精度结果和评估次数结果。表 1 展示了在 $m = 5, n = 5$ 的 SMD 测试集上 SQPEA 与 BLCMAES 算法的上下层函数精度结果。表 2 展示了在 $m = 5, n = 5$ 的 SMD 测试集上 SQPEA 与 BLCMAES 算法的上下层评估结果。其中，我们用粗体突出性能更好的算法，用斜体表示性能相似的算法。

Table 1. Results of precision comparison of simulation experiment functions

表 1. 仿真实验函数精度对比结果

评估指标 问题	上层函数精度		下层函数精度	
	SQPEA	BLCMAES	SQPEA	BLCMAES
SMD1	1.00E-06	1.00E-06	1.00E-06	1.00E-06
SDM2	1.00E-06	1.00E-06	1.00E-06	1.56E-06
SMD3	1.00E-06	1.00E-06	1.00E-06	1.02E-06
SMD4	1.00E-06	1.00E-06	1.00E-06	3.57E-06
SMD5	1.00E-06	1.00E-06	1.00E-06	2.90E-06
SMD6	1.00E-06	1.00E-06	1.00E-06	1.00E-06
SMD7	7.34E-02	9.75E-02	1.16E+02	1.16E+02
SMD8	1.00E-06	9.80E-04	1.39E-06	3.24E-05
SMD9	1.00E-06	1.00E-06	1.00E-06	1.01E-06
SMD10	1.54E+01	1.68E+01	1.66E+01	1.69E+01
SMD11	7.64E-01	5.08E+01	1.19E-02	4.97E+01
SMD12	1.53E+01	1.62E+01	1.83E+01	1.83E+01

Table 2. Results of precision comparison of simulation experiment functions

表 2. 仿真实验函数精度对比结果

评估指标 问题	上层评估次数		下层评估次数	
	SQPEA	BLCMAES	SQPEA	BLCMAES
SMD1	6.32E+02	7.09E+02	5.86E+04	7.53E+04
SDM2	6.14E+02	6.81E+02	6.39E+04	7.44E+04
SMD3	6.81E+02	7.06E+02	6.05E+04	6.86E+04
SMD4	6.21E+02	7.07E+02	6.00E+04	7.56E+04
SMD5	7.94E+02	8.88E+02	6.68E+04	8.22E+04
SMD6	8.11E+02	8.57E+02	5.67E+04	8.25E+04
SMD7	1.23E+03	1.25E+03	5.46E+04	7.62E+04
SMD8	1.42E+03	3.50E+03	1.87E+05	2.53E+05

续表

SMD9	7.14E+02	8.14E+02	6.33E+04	7.81E+04
SMD10	1.77E+03	3.51E+03	1.06E+05	2.12E+05
SMD11	2.67E+03	3.51E+03	1.68E+05	1.95E+05
SMD12	2.87E+03	3.51E+03	1.66E+05	2.14E+05

在表 1 中, 第 2~3 列展示了两种算法的上层函数精度, 第 4~5 列展示了它们的下层函数精度。我们可以发现, 在任意测试问题上 SQPEA 算法的函数精度都要比 BLCMAES 算法更好。这充分体现了下层预优化策略的作用, 通过让下层优化的初始解更靠近下层真实最优, 可以让算法把更多计算资源投入到下层最优的局部搜索中, 提高搜索到更精确个体的概率。同时, 利用最优梯度向量来缩小下层搜索空间, 促进下层种群收敛, 提高下层函数精度的表现。另外, 下层函数精度的提高也会促进上层函数精度的提高, 因为当能准确真实地评估上层变量时, 上层种群的进化方向也会更加正确和高效。

在表 2 中, 第 2~3 列展示了两种算法的上层评估次数, 第 4~5 列展示了它们的下层评估次数。结合表 1 的结果不难看出, 在相似甚至更好的精度表现下, SQPEA 算法消耗的评估资源远少于 BLCMAES 算法。在 SQPEA 算法中, 下层预优化策略可以有效提高下层优化效率, 从而减少下层评估资源的消耗。同时, 下层优化效率的提升也会促进上层优化效率提高, 两者相互促进, 形成良性循环。

6. 结论

本文为双层单目标优化问题设计了一种基于 SQP 方法的双层单目标优化算法。该算法将非线性优化算法 SQP 与进化算法相结合, 将两者的优势结合到一起。在 SQPEA 算法中, 在每一代下层优化前, 使用 SQP 方法对下层问题进行预优化处理, 将预优化得到的最优解作为下层优化的初始解, 并利用优化结束时的最优梯度向量来调整下层变量的上下线, 缩小下层搜索空间的范围。以此来加快下层优化的收敛速度, 提高下层优化效率。而下层优化效率的提高也会提高上层优化的效率, 相互促进, 形成一个良性循环, 提高算法的优化性能。最后在一套标准测试集上与一种成熟的双层优化算法进行仿真对比实验。仿真对比实验的结果证明了本文提出的 SQPEA 算法具有突出的性能表现, 且有效可靠。

参考文献

- [1] Sinha, A., Malo, P. and Deb, K. (2018) A Review on Bilevel Optimization: From Classical to Evolutionary Approaches and Applications. *IEEE Transactions on Evolutionary Computation*, **22**, 276-295. <https://doi.org/10.1109/TEVC.2017.2712906>
- [2] 王云. 罚函数方法及双层规划问题的研究[D]: [硕士学位论文]. 重庆: 重庆大学, 2012. <https://doi.org/10.7666/d.y2154688>
- [3] 赵茂先, 高自友. 基于单纯形方法的双层线性规划全局优化算法[J]. *应用数学*, 2006, 19(3): 642-647.
- [4] 赵莉, 袁振洲, 李之红, 许旺土. 综合客运通道设计的双层规划模型及算法[J]. *北京交通大学学报*, 2009, 33(6): 36-41.
- [5] Sinha, A. and Shaikh, V. (2022) Solving Bilevel Optimization Problems Using Kriging Approximations. *IEEE Transactions on Cybernetics*, **52**, 10639-10654. <https://doi.org/10.1109/TCYB.2021.3061551>
- [6] Sinha, A., Lu, Z., Deb, K., et al. (2020) Bilevel Optimization Based on Iterative Approximation of Multiple Mappings. *Journal of Heuristics*, **26**, 151-185. <https://doi.org/10.1007/s10732-019-09426-9>
- [7] Sinha, A., Malo, P. and Deb, K. (2017) Evolutionary Algorithm for Bilevel Optimization Using Approximations of the Lower Level Optimal Solution Mapping. *European Journal of Operational Research*, **257**, 395-411. <https://doi.org/10.1016/j.ejor.2016.08.027>
- [8] Yin, Y.F. (2000) Genetic-Algorithms-Based Approach for Bilevel Programming Models. *Journal of Transportation*

-
- Engineering*, **126**, 115-120. [https://doi.org/10.1061/\(ASCE\)0733-947X\(2000\)126:2\(115\)](https://doi.org/10.1061/(ASCE)0733-947X(2000)126:2(115))
- [9] Zhao, L. and Wei, J.X. (2019) A Nested Particle Swarm Algorithm Based on Sphere Mutation to Solve Bi-Level Optimization. *Soft Computing*, **23**, 11331-11341. <https://doi.org/10.1007/s00500-019-03888-6>
- [10] He, X.Y., Zhou, Y.R. and Chen, Z.F. (2018) Evolutionary Bilevel Optimization based on Covariance Matrix Adaptation. *IEEE Transactions on Evolutionary Computation*, **23**, 258-272. <https://doi.org/10.1109/TEVC.2018.2849000>
- [11] 陈宇. 求解不等式约束非线性优化问题的改进的 SQP 算法研究[D]: [硕士学位论文]. 长沙: 湖南大学, 2009.
- [12] Sinha, A., Malo, P. and Deb, K. (2014) Test Problem Construction for Single-Objective Bilevel Optimization. *Evolutionary Computation*, **22**, 439-477. https://doi.org/10.1162/EVCO_a_00116