中国科学院大学
University of Chinese Academy of Sciences

博士学位论文

无导数优化的算法与理论

作者姓名：谢鹏程

指导教师：袁亚湘 研究员

中国科学院数学与系统科学研究院

学位类别：理学博士

学科专业：计算数学

培养单位：中国科学院数学与系统科学研究院

2024 年 6 月

谢鹏程 XiePengCheng 2025-07-31 09:40:27 中国科学院数学与系统科学研究院

中国科学院数学与系统科学研究院 XiePengCheng 2025-07-31 09:40:21

Algorithms and Theory of Derivative-Free Optimization

A dissertation submitted to

University of Chinese Academy of Sciences

in partial fulfillment of the requirement

for the degree of

Doctor of Philosophy

in Computational Mathematics

By

Pengcheng Xie

Supervisor: Professor Ya-xiang Yuan

Academy of Mathematics and Systems Science, Chinese Academy of Sciences

June, 2024

中国科学院数学与系统科学研究院 XiePengCheng 2025-07-31

中国科学院数学与系统科学研究院 XiePengCheng 2025-07-31 09:40:21

中国科学院大学
学位论文原创性声明

本人郑重声明：所呈交的学位论文是本人在导师的指导下独立进行研究工作所取得的成果。承诺除文中已经注明引用的内容外，本论文不包含任何其他个人或集体享有著作权的研究成果，未在以往任何学位申请中全部或部分提交。对本论文所涉及的研究工作做出贡献的其他个人或集体，均已在文中以明确方式标明或致谢。本人完全意识到本声明的法律结果由本人承担。

作者签名:

日　　期:

中国科学院大学
学位论文授权使用声明

本人完全了解并同意遵守中国科学院大学有关收集、保存和使用学位论文的规定，即中国科学院大学有权按照学术研究公开原则和保护知识产权的原则，保留并向国家指定或中国科学院指定机构送交学位论文的电子版和印刷版文件，且电子版与印刷版内容应完全相同，允许该论文被检索、查阅和借阅，公布本学位论文的全部或部分内容，可以采用扫描、影印、缩印等复制手段以及其他法律许可的方式保存、汇编本学位论文。

涉密及延迟公开的学位论文在解密或延迟期后适用本声明。

作者签名:　　导师签名:

日　　期:　　日　　期:

中国科学院数学与系统科学研究院 XiePengCheng 2025-07-31 09:40:21

摘 要

大多数优化算法依赖于问题的导数信息. 然而, 在现实世界中, 许多工程计算、设计优化、数据科学、人工智能等领域的实际问题中的目标函数导数信息是不可用或非常昂贵的. 在这些应用场景中, 我们难以获得和利用问题的精确导数信息. 这促使我们研究无导数优化 (derivative-free optimization, DFO) 方法. 无导数优化是科学计算和工程领域最重要和最具挑战性的领域之一, 具有巨大的研究需求和潜力.

基于欠定二次插值模型的信赖域算法是一类高效的无导数优化方法. 使用不同技术更新二次模型将给出不同的插值模型. 本论文提出了一种更新二次模型的新方法, 该方法通过最小化相邻二次模型之间变化量的 H^2 范数来实现. 我们给出了使用 H^2 范数的动机, 也展示了所提出的新的更新方法的理论性质. 这样的二次模型通过使用 KKT 条件计算其系数来被确定. 数值结果展示了我们的新模型在求解所考虑的测试集时的数值优势. 我们还提出了最小加权 H^2 范数更新二次模型, 并讨论了最佳权重系数. 本论文给出了一个新的基于信赖域迭代性质的角度来分析著名的最小范数类欠定二次插值模型. 我们发现了在某些情况下构造考虑了信赖域迭代的二次模型时最优性条件中一个系数的不确定性. 随之导致的二次模型的不唯一性引导我们提出一种新的模型来对模型进行改进, 具体来说, 我们将前一步的欠定二次模型选择性地视为二次模型或线性模型. 我们给出了改进后的欠定二次插值模型, 该模型基于信赖域迭代考虑了模型的最优性, 我们进而给出了一个新的无导数方法. 本论文给出了理论动机、分析和计算细节. 我们的二次模型的公式是易于实现的. 数值结果展示了在无导数优化方法中使用我们的二次模型时的优势. 据我们所知, 我们给出了第一个在构建无导数方法的欠定二次模型时考虑信赖域迭代的性质和模型最优性的工作. 此外, 我们给出了信赖域中非凸二次函数极小点之间距离减小的条件, 并给出了相应的数值例子.

本论文提出了带变换目标函数的无导数优化 (derivative-free optimization with transformed objective functions, DFOTO), 并给出了一种基于最小 Frobenius 范数更新二次模型的信赖域方法. 该模型的更新公式基于 Powell 的公式给出, 是易于实现的. 我们的方法与求解目标函数无变换的问题的方法具有相同的框架, 相应的探测方案也已给出. 我们提出了与保最优性变换相关的定义, 以在极小化带变换目标函数时理解我们方法中的插值模型. 我们证明了, 除了平移变换之外, 仍存在保模型最优性变换. 我们给出了相应的充分必要条件. 我们还分析了目标函数经仿射变换后相应的模型及其插值误差. 变换后目标函数的相应可证算法框架的收敛性质也在本论文中给出. 数值结果表明, 我们的方法能够成功求解大多数带有保目标函数最优性变换的测试问题. 据我们所知, 这是首个为函数探测发生了变换的问题提供基于模型的无导数算法及分析的工作.

此外, 我们提出了一个名为 2D-MoSub 的新方法. 它是一个 2 维模型子空间

无导数方法. 特别地, 2D-MoSub 旨在求解大规模无导数问题. 2D-MoSub 结合了 2 维二次插值模型和信赖域技术来迭代地更新点和探索 2 维子空间. 我们介绍了其框架和计算细节, 包括初始化、插值集、二次插值模型、信赖域试探步以及信赖域半径和子空间的更新. 我们讨论了相应子空间中插值集的适定性和质量, 还分析了方法的一些理论性质, 包括模型的逼近误差、投影性质和 2D-MoSub 的收敛性. 数值结果显示了 2D-MoSub 的优势. 另外, 本论文提出了无导数优化算法 SUSD-TR. 加速减速 (speeding-up and slowing-down, SUSD) 方向被证明在某些情形下趋于梯度下降方向. 我们的 SUSD-TR 算法结合了基于插值点的协方差矩阵的 SUSD 方向和基于这些点的插值模型函数的信赖域子问题的解. 我们分析了 SUSD-TR 算法的优化过程的动力系统和搜索方向的性质. 我们讨论了试探步和结构步. 数值结果显示了 SUSD-TR 的优势.

关键词：无导数优化, 信赖域方法, 二次插值, 大规模问题, 子空间方法

中国科学院数学与系统科学研究院 XiePengCheng 2025-07-31 09:40:21

Abstract

Most optimization algorithms depend on the derivative information of the problem. However, in the real world, the derivative information of objective functions in many practical problems in engineering computations, design optimization, data science, artificial intelligence, and other fields is either unavailable or prohibitively expensive. In these application scenarios, it is difficult for us to obtain and utilize precise derivative information of the problem. This motivates us to study derivative-free optimization (DFO) methods. Derivative-free optimization is one of the most important and challenging areas in scientific computing and engineering, with significant research demands and potential.

The trust-region methods based on the under-determined interpolation quadratic models is an efficient class of derivative-free optimization methods. Updating the quadratic model using different techniques will derive different models. This thesis proposes a new method to update the quadratic model, which is achieved by minimizing the H^2 norm of the change between neighboring quadratic models. We give the motivation for applying the H^2 norm and the theoretical properties of the proposed new updating method. Such a model is determined by calculating the coefficients using the KKT conditions. Numerical results show our new model's numerical advantages in solving the considered test set. We also propose the least weighted H^2 norm updating quadratic model and discuss the best weight coefficients. This thesis gives a new perspective based on the property of trust-region iteration to analyze the famous least norm type under-determined quadratic interpolation model. We find the non-determinacy of a coefficient in the optimality condition when constructing a quadratic model considering the trust-region iteration in some cases. The consequent non-uniqueness of the quadratic model leads us to propose a new model to improve the model. In detail, we selectively treat the previous under-determined quadratic model as a quadratic model or a linear model. We give an improved under-determined quadratic interpolation model, and it considers the optimality of the model based on the trust-region iteration. We consequently give a new derivative-free method. This thesis gives the theoretical motivation, analysis, and computational details. Our quadratic model's formula is implementation-friendly. The numerical results show the advantages of using our quadratic model in the derivative-free optimization methods. To the best of our knowledge, we provide the first work considering the property of trust-region iteration and the model's optimality when constructing the under-determined quadratic model for derivative-free methods. In addition, we give the conditions of distance reduction between the minimizers of non-convex quadratic functions in the trust region and the corresponding numerical ex-

amples.

This thesis proposes derivative-free optimization with transformed objective functions (DFOTO) and gives a model-based trust-region method with the least Frobenius norm updating quadratic model. The model updating formula is based on Powell's formula, and it can be easily implemented. Our method has the same framework as the methods for solving problems without transformations, and its query scheme is also given. We propose the definitions related to optimality-preserving transformations to understand the interpolation model in our method when minimizing transformed objective functions. We prove the existence of model optimality-preserving transformations beyond translation transformations. We give the corresponding necessary and sufficient condition. We also analyze the corresponding model and its interpolation error when the objective function is affinely transformed. The convergence property of a provable algorithmic framework related to the transformed objective functions is given in this thesis. Numerical results show that our method can successfully solve most test problems with objective optimality-preserving transformations. To the best of our knowledge, this is the first work providing the model-based derivative-free algorithm and analysis for transformed problems with the function evaluation oracle.

In addition, we propose a novel method named 2D-MoSub. It is a 2-dimensional model-based subspace derivative-free method. 2D-MoSub especially aims to solve large-scale derivative-free problems. 2D-MoSub combines 2-dimensional quadratic interpolation models and trust-region techniques to iteratively update the points and explore the 2-dimensional subspace. We introduce its framework and computational details, including initialization, the interpolation set, the quadratic interpolation model, trust-region trial steps, and the updating of trust-region radius and subspace. We discuss the poisedness and quality of the interpolation set in the corresponding subspace and analyze some properties of our method, which include the model's approximation error, projection property and 2D-MoSub's convergence. Numerical results show the advantage of 2D-MoSub. Besides, this thesis proposes the derivative-free optimization algorithm SUSD-TR. The speeding-up and slowing-down (SUSD) direction is proved to converge to the gradient descent direction in some cases. Our SUSD-TR combines the SUSD direction based on the covariance matrix of interpolation points and the solution of the trust-region subproblem of the interpolation model function based on such points. We analyze the dynamics of the optimization process and the direction's properties of the algorithm SUSD-TR. We discuss the trial step and structure step. Numerical results show the advantage of SUSD-TR.

Key Words: derivative-free optimization, trust-region method, quadratic interpolation, large-scale problem, subspace method

目 录

中国科学院数学与系统科学研究院 XiePengCheng 2025-07-31 09:40:21

中国科学院数学与系统科学研究院 XiePengCheng 2025-07-31 09:40:21

图目录

表目录

中国科学院数学与系统科学研究院 XiePengCheng 2025-07-31 09:40:21

符号列表

字符

Symbol	Description
$\mathfrak{R}$	实数集
$\mathfrak{R}^+$	正实数集
$\mathfrak{R}^n$	n 维实向量集
$\mathfrak{R}^{m\times n}$	$m\times n$ 维实矩阵集
$\mathbb{Z}$	整数集
$\mathbb{N}$	自然数集
$\mathbb{N}^+$	正整数集
$\mathbf{X}^{-1}$	矩阵 $\mathbf{X}$ 的逆矩阵
$\mathbf{X}^\top$	矩阵 $\mathbf{X}$ 的转置
$\|\mathbf{X}\|_{\mathrm{F}}$	矩阵 $\mathbf{X}$ 的 Frobenius 范数
$\mathcal{B}_r(\boldsymbol{x}_0)$	$\left\{\boldsymbol{x}\in\mathfrak{R}^n:\left\\|\boldsymbol{x}-\boldsymbol{x}_0\right\\|_2\leq r\right\}$
$\mathcal{V}_n$	n 维 ℓ_2 范数单位球 $\mathcal{B}_1(\boldsymbol{x}_0)$ 的体积
$\boldsymbol{I}$	单位矩阵
$\mathbf{0}_{nm}$	$n\times m$ 维零矩阵
$\boldsymbol{e}_t$	单位矩阵 $\boldsymbol{I}$ 的第 t 列
$\langle\cdot,\cdot\rangle$	内积

算子

Symbol	Description
min	极小化
max	极小化
∂	偏导
∇	梯度

缩写

DFO	derivative-free optimization
SUSD	speeding-up and slowing-down
KKT	Karush-Kuhn-Tucker
s. t.	subject to

中国科学院数学与系统科学研究院 XiePengCheng 2025-07-31 09:40:21

第 1 章　绪论

在科学和工程领域, 优化问题一直是一个备受关注且至关重要的研究领域. 无论是在工业生产、物流运输、金融投资还是人工智能等领域, 优化问题的解决都能够直接影响到系统的效率和性能 [1–8]. 通过求解优化问题, 我们可以实现资源的最优利用, 降低成本, 提高产品质量, 甚至优化决策制定流程等.

实际应用中的优化问题往往具有复杂的特征 [9–19], 其中包括非线性、非凸以及多变量等. 这些特性使得优化问题的求解变得更加困难和复杂. 例如, 在工程设计中, 我们经常需要考虑到各种物理限制条件以及系统的非线性行为; 在金融领域, 投资优化问题往往涉及到多个资产的非线性收益率和风险关系; 在医疗领域, 药物配方优化问题可能涉及到多个药物成分的相互作用以及对患者生理状态的影响等. 这些复杂性促使我们寻找更加灵活和有效的数值算法来求解这些非线性优化问题.

本论文重点介绍无约束问题的求解, 考虑到这类问题不仅本身有广泛的应用, 其对应方法和求解思路亦可推广至求解有约束问题, 同时, 有些有约束问题可转化为无约束问题. 故本论文所讨论的无约束方法是优化方法的基本. 大多数无约束优化方法需要使用优化问题中目标函数的导数. 然而, 在部分实际情况中, 目标函数的计算成本很高, 且其导数无法获得. 一类典型的例子是, 问题中的目标函数不是通过解析函数表达的, 而是通过一个 “黑箱” 获得, 如化学过程或计算机模拟. 这类问题对应的优化不使用目标函数的导数, 也被称为无导数优化. 也就是说, 无导数优化方法是一类不需要目标函数一阶或更高阶导数的数值方法. 更多关于无导数优化的介绍可以参考 Conn、Scheinberg 和 Vicente 的著作 [20] 以及 Audet 和 Hare 的著作 [21].

本章作为绪论, 主要介绍了本论文的研究背景、目的和研究内容的组织结构. 首先, 在第 1.1 节中, 我们探讨了无导数优化的重要性和应用领域, 在第 1.1.1 节讨论了无导数优化的具体应用, 在第 1.1.2 节对无导数优化方法进行了分类和概述. 随后, 第 1.2 节详细介绍了基于模型的无导数优化方法, 这是本论文的研究重点, 我们分别在第 1.2.1 节和第 1.2.2 节中详细介绍了相关算法和插值模型的基础概念. 第 1.3 节主要介绍无导数优化算法的评价方法, 我们给出了评估无导数优化方法性能的标准和方案. 最后, 在第 1.4 节中, 我们概述了文章的主要内容和组织结构.

1.1　无导数优化

本论文考虑和研究无约束的无导数优化问题

$$\min_{\boldsymbol{x}\in\Re^n} \ f(\boldsymbol{x}), \tag{1-1}$$

其中 f 是一个实值函数, 没有可用的一阶或更高阶导数信息. 我们知道, 无导数优化方法是一类在优化过程中不使用目标函数的真实或精确导数信息的数值方法. 由于只需要使用函数值, 无导数优化方法有广泛的应用. 与此同时, 由于缺乏导数信息, 无导数方法存在一些缺点, 比如其很难获取对优化问题中原目标函数的近似, 我们难以较好地掌握黑箱函数的形态. 在本论文的讨论中, 我们假设很难获得目标函数或其导数的完美精确近似. 注意, 正因如此, 无导数优化方法的理论性质通常难以分析.

1.1.1 无导数优化的应用

无导数优化问题在实际应用中非常常见, 其应用在工程领域十分广泛. 在优化算法发展的早期阶段, 尽管在有些背景下非线性优化问题的导数信息已经可知, 但相应的理论框架还不够成熟, 缺乏有效的基于导数的方法, 在这种情况下, 许多无导数优化方法因其简单性和易用性而在当时受到用户的偏爱. 随着科学计算技术的快速发展, 面对日益增长的问题规模和复杂度, 原有的无导数方法在求解时开始显得力不从心, 这促进了基于导数的更为复杂的方法的持续发展与完善. 这些方法通常要求用户提供目标函数的导数信息. 然而, 这并不总是可行的, 这是因为所需要的函数值在很多实际工程场景中可能产生于某些物理、生物或计算机等实验的测量结果. 这些获取函数值的过程只能被视为一种黑箱系统.

无导数优化的一些应用例子包括数值算法的调参 [22]、神经网络的优化 [23]、自动误差分析 [24]、动态定价 [25] 和工程设计中的最优设计 [26] 等.

值得注意的是, 无导数优化方法在工业和工程领域得到了广泛应用 [27–29], 特别是在求解涉及复杂模型或实验的问题时发挥了重要作用. 这些问题来自于各种领域, 如机翼设计 [30]、气动声学外形设计 [31, 32]、流体动力学设计 [33]、电路设计 [34, 35] 等. 需要对复杂模型、仿真或实验进行优化的问题都可能涉及到无导数优化问题. 值得注意的是, 在工业界, 多学科设计优化 [36] 是无导数优化的一个著名的具体应用. 由于多学科设计优化问题通常涉及工业生产中的仿真或实验, 因此我们常常需要使用无导数优化方法来求解对应的黑箱问题 [37].

许多与数据科学和机器学习相关的无导数优化、黑箱优化问题也已经出现 [38, 39]. 例如, 针对神经网络的黑箱攻击 [40] 就可以被看作是要求解一个无导数优化问题. 在大多数情况下, 超参数的调整也是一个黑箱 [22]. 比如, 使用无导数优化方法改善气候建模中的参数选择 [41].

下面详细介绍一些具体的无导数优化的应用案例.

例 1.1 (参数优化)**.** 对数值算法中的参数进行优化是一个非常有意义的研究方向 [22]. 我们知道, 大多数数值算法需要依赖于一组提前设定的参数, 与此同时, 这些参数可能是依赖于算法开发者的经验提供的推荐值, 或者需要用户自行摸索尝试而设定. 这些参数的选择对算法的性能有着显著的影响. 一个有效的参数选取方法是通过求解一个黑箱优化问题来获得好的参数. 我们考虑将参数作为变量, 以算法在测试集上的性能 (例如, 通过计算机 CPU 运行时间或迭代次数来评估) 作

为目标函数. 通常, 这些参数是受到上下界约束的. 因此, 该优化问题可以简单地表述为

$$\min_{\boldsymbol{p}\in\mathfrak{R}^n} f(\boldsymbol{p}) = \text{性能}(\boldsymbol{p})$$

$$\text{s. t. } l_i \le \boldsymbol{p}_i \le u_i,\ \forall\, i = 1, \cdots, n,$$

这里的 $\boldsymbol{p}$ 表示需要调整的参数, $\boldsymbol{p}_i$ 为其元素. 注意, 这类问题的目标函数通常是无法给出解析表达式或计算导数的 [42].

例 1.2 (气候建模与预测)**.** 全球气候数值模型一般是一个极其复杂的计算机程序, 由复杂代码构建, 是一个黑箱, 其目的是通过模拟海洋、大气和陆地间复杂的相互作用来建模和预测地球的气候变化. 这些模型结合了众多的输入变量, 如大气中的温室气体浓度、太阳辐射、海洋流动和地表覆盖类型等, 来模拟、仿真和预测地球气候系统的行为. 由于全球气候模型需要处理的数据量巨大且变量众多, 它们需要在高性能计算机上运行, 这进一步增加了研究成本和复杂性, 同时这些模型覆盖的时间范围长且复杂, 综上, 我们可以将其中的参数选择视为一个昂贵的黑箱优化问题 [41].

例 1.3 (黑箱攻击)**.** 在人工智能的研究和应用中, 黑箱攻击是一种旨在破坏神经网络识别系统的行为, 其核心策略是向神经网络的输入数据中添加噪声, 以误导神经网络, 使其给出错误的输出结果. 举例来说, 攻击者可能会在图像数据中引入细微的扰动, 这种扰动虽然肉眼几乎察觉不到, 却能够导致神经网络做出错误的识别决策. 例如, 通过对一张熊猫图片添加精心设计的噪声, 攻击者可以使神经网络将其错误地识别为长臂猿. 这类攻击具有特别挑战性的原因在于, 攻击者通常没有访问对应神经网络内部结构和工作机制的权限, 即处于所谓的黑箱环境. 也就是说, 攻击者需要在完全不了解神经网络内部参数和架构的情况下, 找到能够有效干扰网络输出的噪声, 这本质上可以转换为一个黑箱优化问题 [40].

例 1.4 (分子几何)**.** 在化学和物理学的科学研究中, 一个引人注目的应用领域是分子几何学的优化. 当我们考虑一个包含多个原子的分子或原子团时, 其几何构造可以由一定的自由度或自变量来描述. 具体来说, 我们的目标是找到一个好的几何构造, 使得整个分子或原子团的势能最低, 即最小化其能量. 虽然有时这类优化问题的梯度可以获得, 但计算这些梯度可能成本较高且结果带有噪声. 在这种情况下, 无导数优化方法, 特别是直接搜索策略, 已经被证明是一种有效的工具 [43, 44].

除此之外, 还有一些最优策略问题也是无导数优化问题. 例如, Wild 和 Shoemaker 描述了减少地下水中有害物污染的最佳抽水策略问题. 简单地说, 问题中有一组井在一定的抽水速率下运行, 它们可以注入纯净水或去除和处理受污染的水. 如果我们想要探究: 对于运行 30 年以上的在 15 口井, 为了让含水层中的有害物质浓度符合环保标准, 最佳的、成本最低的抽水策略是什么. 这里, 进行一次评估 (在当时的条件下) 需要耗时 45 分钟以上的地下水流模拟, 同时, 该模拟代码太复杂, 无法得到对应的自动微分, 这就是一类典型的黑箱无导数优化问题 [45].

1.1.2 无导数优化方法的分类和概述

研究历史悠久的无导数方法具有不同类型: 例如, 直接搜索方法、线搜索方法、基于模型的方法、启发式算法等等. 我们在本节给出简单介绍, 丁晓东 [42] 和张在坤 [46] 的文章中也有过详细介绍, 这里也参考了他们的总结和介绍.

直接搜索方法

直接搜索方法是一大类无导数方法 [47–49], 其主要通过构建某种几何结构, 在变量可行域中进行搜索. 一般来说, 直接搜索方法既不使用原目标函数的导数值, 也不使用使用差分等方式所得到的近似导数信息. 这类方法主要依赖低维几何直观, 缺少丰富、严格的数学理论. 从历史上看, "直接搜索" 一词最初由 Hooke 和 Jeeves 于 1961 年提出. 在他们的研究中, 直接搜索的一个核心特征是仅需比较探测点处函数值的大小关系而不依赖于具体的目标函数值. 换言之, 任何目标函数值的减小都将被接受. 可以看到, 这种方法相对来说简洁直观.

具体而言, 直接搜索方法包括模式搜索方法、单纯形方法、方向直接搜索方法、网格自适应直接搜索方法等. 一些例子包括 Hooke-Jeeves 方法 [50]、Nelder-Mead 方法 [51]、改进的单纯形方法 [52]、生成集搜索方法 [49] 等. 这里提及的模式搜索方法是其中一类直接搜索方法 [46], 主要包括 Fermi-Metropolis 方法 [53]、Evolutionary Operation 方法 [54]、Hooke-Jeeves 方法 [50]、多方向搜索方法 [55–57]、广义模式搜索方法 [58–62]、异步并行模式搜索方法 [63, 64]、网格自适应直接搜索 MADS 方法 [65, 66] 等. 事实上, 现代意义上的直接搜索法至少可以追溯到 Fermi 和 Metropolis 的研究报告 [53].

用一种形象的表达, 极小化一个二元函数, 可类比为从一座山上的某点出发寻找海拔最低点, 可以想象, 一个基本的做法是找到一条山谷并沿其前进搜索. 模式搜索法的主要思想就是用探测步为寻找山谷获取信息, 而后在模式步实现沿山谷的前进 [67].

除此之外, 单纯形方法 [20] 也是一类直接搜索方法. Nelder-Mead 单纯形方法 [51] 是这类方法的代表 [46]. 该方法在实际应用得到了广泛的使用. 简单地说, 对于 n 维问题, Nelder-Mead 方法从 $n+1$ 个初始点构成的单纯形开始迭代, 根据单纯形顶点处的函数值对单纯形进行反射、扩张、收缩等基本操作, 该算法的基本原理是希望单纯形会迭代地实现对目标函数形态的局部逼近 [51], 最终顺利收敛. 遗憾的是, Nelder-Mead 方法没有很好的收敛性理论 (即便是极小化严格凸的函数 [68]). 此外, 有学者对 Nelder-Mead 方法进行了修正和改进 [69–71].

这里, 我们再举一类具体的直接搜索算法: 方向直接搜索方法. 简单地说, 方向直接搜索方法的每次迭代在当前点 $\boldsymbol{x}_k$ 附近产生有限的点集, 这些候选点从 $\boldsymbol{x}_k$ 点出发移动 $\alpha_k \boldsymbol{d}$ 来生成, 这里的 α_k 是一个正步长, 方向 $\boldsymbol{d}$ 则选自当前步对应的有限方向集合. 然后该方法在所有或一些候选点处获取对应的目标函数值, 并将 $\boldsymbol{x}_{k+1}$ 设置为可能会使目标函数值减小且步长可能增加的点, 注意, 如果算法发现该步没有给出足够下降的候选点, 则将 $\boldsymbol{x}_{k+1}$ 设置为 $\boldsymbol{x}_k$ 并减小步长. Kolda 等 [49] 提出术语 "生成集搜索方法" 来定义这类方法.

值得注意的是, 学者们在直接搜索算法的研究过程中, 逐步建立健全了对正向基这一数学概念的相关基础理论的系统研究 [20, 21].

线搜索方法

另一类无导数方法是不使用原目标函数导数信息的线搜索方法. 学者们发现直接搜索方法依赖于比较目标函数在网格点或单纯形顶点的值, 而没有考虑问题潜在的连续性和光滑性等特性, 这导致其收敛速度较慢. 当引入 1 维搜索技术后, 这种方法的效率得到了显著提升 [42]. 具体来说, 求解无约束优化问题 (1-1) 的经典、基本线搜索框架如算法1所示 [46]. 根据搜索方向的不同选择, 总体上有三种线搜索方法：交替方向法 (例如, Rosenbrock 方法 [72] 及其改进版本 [73])、共轭方向法 [74, 75], 以及基于近似梯度的方法 (例如, 有限差分拟牛顿法 [76–78]). 此外, 还有一些基于随机梯度近似的方法 [79–83]. 一般来说, 我们可以在给定搜索方向之后, 使用区间分割法等方法选取合适的步长 [2, 84].

算法 1 线搜索方法框架

步 1. (初始化) 获取初始点 $\boldsymbol{x}_1$, 令 $k = 1$.

步 2. (选择搜索方向) 选取搜索方向 $\boldsymbol{d}_k$.

步 3. (选择步长) 求解

$$\min_{\alpha \geq 0} \ f(\boldsymbol{x}_k + \alpha \boldsymbol{d}_k),$$

获取步长 α_k.

步 4. (更新) 令 $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k \boldsymbol{d}_k$, 令 $k = k + 1$. 转步 2.

作为一类无导数线搜索方法, 坐标轮换法的基本原理是轮流使用所有坐标方向作为搜索方向, 它的一个缺点是, 可能会出现锯齿现象. 克服这一困难的一个办法是引入 Hooke 和 Jeeves [50] 的模式搜索. 注意, 坐标轮换的一个简单推广是把坐标方向扩展为任意一组正交基, Rosenbrock 方法 [72] 使用了类似的思路, 也称为转轴法. 袁亚湘的著作 [2] 对交替方向法的收敛性作了介绍.

此外, 共轭方向法是指在求解过程中迭代地产生共轭方向作为搜索方向的方法. 最早的共轭方向法由 Smith [74] 提出. Powell [75] 和 Zagwill [85] 也研究了共轭方向法.

基于近似梯度的线搜索方法的思想是用某种方式得到数值近似梯度, 在算法框架上类比经典的基于导数的方法. 一种最简单直接的近似梯度的方式就是有限差分. 差分拟牛顿方法 [76] 是基于有限差分近似导数的拟牛顿方法. Gill 和 Murray [77] 研究了基于差分的 Broyden 族方法. 隐式滤子法 [86–88] 可以被视为一种基于单纯形梯度的拟牛顿方法. 此外, 这类方法中还有无导数拟牛顿法 [89, 90], 其给出了基于函数值的拟牛顿条件, 其中梯度和 Hessian 矩阵满足相应的变分极小性质 [46]. 此外, 近些年来, 关于随机梯度近似的研究和方法 [79–83] 也在日益成熟, 其中包括单点随机梯度近似、两点梯度随机近似以及多点梯度随机近似.

基于模型的方法

基于模型的方法是无导数方法中的重要类型. 具体来说, 二次插值模型方法、欠定二次插值模型方法和回归模型方法都使用了多项式模型 [91–99]. 另一种基于模型的无导数方法是基于径向基函数插值模型的方法 [100, 101]. 事实上, 还有一些基于模型的组合方法, 例如结合信赖域方法和线搜索方法的方法 [102]. 目前也已经有学者对基于概率模型的信赖域方法进行了讨论 [103, 104]. 基于模型的无导数优化方法是本论文的研究重点, 我们将在第1.2节重点给出进一步的展开介绍.

启发式算法

事实上, 大多数的现代启发式算法也不使用导数信息, 模拟退火算法 [105] 和遗传算法 [21] 就属于这类方法 [42].

其中, 遗传算法源于自然选择和遗传学的原理, 这一算法最初由 Holland 在 1975 年提出 [106–108], 它是一种模拟生物进化中 “适者生存” 原则和群体内部基因随机交换机制的搜索技术. 该方法仿照和参考生物进化和遗传的规律, 编码后, 算法从某一初始群体 (初始可行解) 开始迭代, 复制、杂交、变异等, 逐代重复操作、优胜劣汰, 找到最优解. 其核心思想是使用 “适者生存” 法则来淘汰差解并为优化问题培育新解的算法. 为了与该方法的生物学术语保持一致, 搜索点通常被称为个体, 点的集合形成群体. 一个点内的信息被编码为染色体, 并且使用类生物过程来获取新搜索点 [21].

模拟退火算法是基于金属热加工发展起来的一种搜索算法, 1983 年 Kivkpatrick 等 [109, 110] 将这种算法的思想用于求解优化问题. 它的基本思想是将一个优化问题类比成一个金属物体, 将优化问题的目标函数、问题的解、最优解分别类比为物体的能量、状态、能量最低的状态, 然后模拟金属物体的退火过程, 即从一个足够高的温度开始, 逐渐降低温度, 使物体分子达到能量最小的理想状态, 进而找到带求的优化问题的解.

其他方法和一些数值软件

无导数方法还包括混合方法, 其中有隐式滤子法 [87, 88]、自适应正则化法 [111] 等. 下面我们简单介绍一个混合方法的例子.

隐式滤子法是一类混合方法, 它可以被理解为是网格搜索算法和拟牛顿局部优化方法的混合, 其中相应的差分参数会适时调整.

此外, 演化算法 [112] 也是一大类算法, 有一大部分这种方法不需要用到目标函数的导数信息. 协方差矩阵自适应演化算法 CMA-ES [113–115] 是其中一种演化策略算法. 简单地说, 其使用高斯分布在优化问题的解空间中进行采样, 并且根据某种样本选择机制对高斯分布进行更新. 该方法利用采样和更新过程持续迭代, 最终搜索到满意的解.

受到鱼群在环境中寻找较暗区域这一行为的启发, 目前新兴了一种分布式源搜索策略. 这一策略产生了一个与鱼群中观察到的行为非常相似的加速减速 (SUSD) 行为 [116, 117]. 该策略可以被看作是一种粒子群 (仿生) 优化算法, 允许

每个搜索点实时测量场值 (与优化场景中的目标函数值对应), 并共同向一个近似负梯度方向移动. 其中, 每个搜索点移动速度被设计为与其场测量值成正比. 后文也会对此方法进行介绍和改进.

针对特殊问题的无导数方法也存在, 例如最小二乘法的相关方法 [118, 119]、复合优化的无导数方法 [120, 121] 等, 还有一个例子是带有特殊约束的优化, 如椭球约束无导数优化 [122], 分布式无导数优化 [123] 等. 另外还有一些其他类型的无导数优化方法, 如贝叶斯优化方法 [124]、使用了随机技巧的方法 [103, 125, 126] 和全局优化 [127] 等. 除了 Conn、Scheinberg 和 Vicente 的著作 [20] 以及 Audet 和 Hare 的著作 [21] 外, 一些综述文章, 如 Larson、Menickelly 和 Wild 的文章 [128]、张在坤的综述 [129] 以及 Rios 和 Sahinidis 的工作 [130], 也详细介绍了各种类型的无导数优化方法.

此外, 目前已经有学者和机构研究开发了无导数优化方法的软件和求解器. 其中的例子包括 CMA-ES [131]、DFO [132]、IMFIL [133]、SOLNP+ [134] 等. Powell 开发了 TOLMIN [135]、COBYLA [136]、UOBYQA [137]、NEWUOA [94]、BOBYQA [138] 和 LINCOA [139] 这一系列算法. 另外, 还有 DFO-LS [140] 和 DFBGN [141] 等算法. 最近, Ragonneau 和张在坤开发了 PDFO, PDFO 为 Powell 的无导数优化求解器提供了跨平台接口 [142], 他们还开发了 COBYQA [143] 算法. 我们开发了 NEWUOA 和 BOBYQA 算法的 MATLAB 版本 [144, 145] 和 Python 版本. 除此之外, 张在坤的 PRIMA [146] 为 Powell 的方法提供了现代化和改良的参考实现.

1.2　基于模型的无导数优化方法

1.2.1　算法

这里我们来具体介绍基于模型的无导数优化方法. 基于模型的方法是一种经典高效的无导数方法. 其所使用的获取多项式模型的方法包括线性插值 [147]、二次插值 [91, 148]、欠定二次插值 [92]、回归 [20]. 径向基函数插值 [100] 是获得所使用模型的另一选择. 同时, 随机模型也可以用于信赖域方法 [103, 104]. 除此之外, 还有为噪声问题设计的基于模型的方法 [149]、极小化带变换目标函数的方法 [150] 以及使用三次模型的方法 [151].

基于模型的无导数优化方法的主要思想是构造一个模型函数来在每一步迭代局部逼近原始黑箱目标函数. 然后算法利用模型函数的信息 (包括梯度) 逐步获得迭代点. 大多数基于模型的方法都采用信赖域框架 [152], 通过在当前或最优迭代点附近 (通常以之为中心) 的区域内极小化二次模型来生成新的迭代点, 如 (1-2) 所示. 对于无导数情况, 相应的模型通常通过在插值点处的函数值 (多项式) 插值构造, 这样的方法被称为基于 (插值) 模型的无导数信赖域方法.

具体而言, 二次插值模型方法 [91]、欠定二次插值模型方法 [92, 93, 97, 102, 150, 153, 154] 和回归模型方法 [95, 96] 都使用了逼近模型. 算法 UOBYQA [137] 和 CONDOR [155] 使用了二次模型. 算法 NEWUOA [94]、DFO [156] 和 MNH

[157] 是使用欠定二次插值模型的方法的例子. 另一种基于模型的无导数方法是径向基函数插值模型方法 [100], BOOSTERS [158] 和 ORBIT [101] 是此类算法的例子. 还有楔形信赖域方法 [159]、相关的最小二乘问题的方法 (包括算法 DFBGN [141] 和 DFO-LS [140]), 以及使用稀疏模型的算法 [153]. 本论文会详细考虑基于模型的无导数信赖域方法中的模型插值步骤, 基于模型的无导数信赖域算法的框架如算法 2 所示. 无导数信赖域算法的基本收敛性 [96, 160] 以及灵活性和鲁棒性改进 [140] 已得到研究. 关于求解二次模型函数信赖域子问题 (1-2) 的讨论可参考 Conn、Gould 和 Toint 著作 [152] 的第 7 章. 求解模型函数信赖域子问题的其中一种常用方法是截断共轭梯度法 [161–163].

算法 2 基于模型的无导数信赖域算法的框架

输入: 黑箱目标函数 f 和初始点 $\boldsymbol{x}_{\text{int}}$.

输出: 极小点 $\boldsymbol{x}^*$ 和函数极小值.

获取/设置初始信赖域半径 Δ_0 和其他初始参数.

步 1. (管理插值点集)

选择插值点集 $\mathcal{X}_k \subset \mathfrak{R}^n$. $\mathcal{X}_k$ 中的大多数点在之前的迭代中已经被探测了函数值. 这里, 算法会对所有尚未获取函数值的 $\boldsymbol{y} \in \mathcal{X}_k$ 获取函数值 $f(\boldsymbol{y})$.

步 2. (构造插值模型)

使用线性或二次插值模型函数 (或径向基函数插值模型) Q_k 来逼近 f.

步 3. (信赖域迭代)

通过求解

$$\begin{aligned} &\min_{\boldsymbol{x}} Q_k(\boldsymbol{x}) \\ &\text{s. t. } \left\|\boldsymbol{x}-\boldsymbol{x}_k\right\|_2 \leq \Delta_k \end{aligned} \tag{1-2}$$

来获取 $\boldsymbol{x}_k^+$ 和函数值 $f(\boldsymbol{x}_k^+)$. 相应地更新 $\boldsymbol{x}_{k+1}$ 和 Δ_{k+1}. 具体来说, 通过在新点处获取函数值并将目标值的真实减小量与模型预测的减小量进行比较来确定迭代是否成功, 并更新迭代点和信赖域.

若迭代未达到终止条件, 令 $k = k+1$, 转**步 1**.

事实上, 为使用或不使用目标函数导数信息的信赖域方法构造一个良好的局部逼近模型是非常重要的. 在无导数优化中, 获得模型的最常见方法是基于探测函数值的插值来确定模型 Q, 其形式为

$$Q(\boldsymbol{x}) = \frac{1}{2}\left(\boldsymbol{x}-\boldsymbol{x}_0\right)^{\top} \boldsymbol{H}\left(\boldsymbol{x}-\boldsymbol{x}_0\right) + \boldsymbol{g}^{\top}\left(\boldsymbol{x}-\boldsymbol{x}_0\right) + c,$$

其中 $\boldsymbol{x}_0$ 是一个给定向量, 对称矩阵 $\boldsymbol{H} \in \mathfrak{R}^{n\times n}$, $\boldsymbol{g} \in \mathfrak{R}^n$ 和 $c \in \mathfrak{R}$ 共有 $\frac{1}{2}(n+1)(n+2)$ 个未知系数需要确定, 从二次函数集合中选择一个满足函数值约束

$$Q(\boldsymbol{y}_i) = f(\boldsymbol{y}_i),\ \forall\ \boldsymbol{y}_i \in \mathcal{X}_k \tag{1-3}$$

的函数记为 Q, 其中 $\mathcal{X}_k$ 表示第 k 步的插值点集和 (简称为插值集或插值点集), 我们假设 $\mathcal{X}_k$ 中的 m 个插值点为 $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m$.

在本论文中, 确定的二次模型 [20] 指的是所有 $\frac{1}{2}(n+1)(n+2)$ 个独立参数可以通过插值条件 (1-3) 唯一确定的情况. 欠定二次模型 (Conn、Scheinberg 和 Vicente 著作 [20] 的第 5 章) 指的是在满足插值条件 (1-3) 后二次函数仍有剩余自由度的情况, 这是因为 $m < \frac{1}{2}(n+1)(n+2)$. 换句话说, 一个欠定二次模型不能仅通过函数值约束 (1-3) 唯一确定. 在无导数优化中, 欠定二次模型是信赖域算法中使用的最重要的二次模型类型之一.

需要指出的是, 出于节约使用算法求解时的函数值探测成本的目的, 在基于插值的方法中, 第 k 次迭代完成后, 插值点集 $\mathcal{X}_k$ 中的点通常并不会全部被丢弃, 其中的大部分点会被 $\mathcal{X}_{k+1}$ 继承; 同时, 在绝大多数情况下, 新得到的迭代点也会进入 $\mathcal{X}_{k+1}$, 除非这样做会严重影响 $\mathcal{X}_{k+1}$ 的适定性[1]. 考虑到插值的准确性通常在局部良好, 我们想要探索的欠定二次模型通常更适合用于表征目标函数的局部性质, 而不是表征目标函数的全局性质, 因此它经常被用于信赖域框架中.

考虑在信赖域框架中使用欠定二次插值函数主要有三个原因. 首先, 二次模型函数可以在局部捕捉和刻画目标函数的曲率信息. 其次, 使用较少的插值点可以减少函数值探测的次数. 最后但同样重要的是, 样本/插值点需要合理地靠近当前迭代点. 然而, 二次模型需要 $\mathcal{O}(n^2)$ 个插值点, 这样一来, 在很多情况下, 有用的插值点数量会低于或远低于多项式基中元素的数量.

注意, 如前文所述, 二次模型函数 Q_k 的系数是对称的 Hessian 矩阵 $\nabla^2 Q_k$、梯度向量 ∇Q_k 和常数项, 它们的自由度总和是 $\frac{1}{2}(n+1)(n+2)$, 即 $\mathcal{O}(n^2)$. 当问题的维数 n 较大时, 如果我们通过解插值方程 (1-3) 来直接确定模型函数 Q_k 的系数, 函数值探测的次数就很多. 为了减少函数值探测的次数, 我们可以使用较少的插值点来获取二次模型函数. 为了唯一确定 Q_k 的系数, Powell [94] 建议把子问题

$$\begin{aligned} &\min_{Q\in\mathcal{Q}} \left\|\nabla^2 Q - \nabla^2 Q_{k-1}\right\|_F^2 \\ &\text{s. t. } Q(\boldsymbol{y}) = f(\boldsymbol{y}),\ \forall\, \boldsymbol{y}\in\mathcal{X}_k \end{aligned} \tag{1-4}$$

的解作为我们要的二次模型 Q_k, 其中符号 $\|\cdot\|_F$ 表示 Frobenius 范数, 即对于给定矩阵 $\boldsymbol{C}\in\mathfrak{R}^{n\times n}$, 其 Frobenius 范数为 $\|\boldsymbol{C}\|_F = (\sum_{i,j} c_{ij}^2)^{\frac{1}{2}}$, 其中 $c_{ij}, 1\le i,j\le n$, 为矩阵 $\boldsymbol{C}$ 的元素. 注意, 这里 $\mathcal{X}_k$ 表示第 k 次迭代的插值集, $\mathcal{Q}$ 表示二次函数的集合[2]. 换句话说, 这样得到的二次模型函数 Q_k 满足: 在符合插值条件 $Q(\boldsymbol{y}_i) = f(\boldsymbol{y}_i),\ i = 1,\cdots,m$ 的所有二次函数中其对应的 $\nabla^2 Q_k - \nabla^2 Q_{k-1}$ 具有最小的 Frobenius 范数, 其中 $\boldsymbol{y}_1,\cdots,\boldsymbol{y}_m$ 表示当前的插值点, 并且 $m < \frac{1}{2}(n+1)(n+2)$.

根据 Frobenius 范数的凸性, 能够证明在第 k 次迭代中可以唯一确定二次模型 Q_k. 这种模型的一个优点是它在目标函数是二次函数时具有投影性质 [93], 即

$$\left\|\nabla^2 Q_k - \nabla^2 f\right\|_F \le \left\|\nabla^2 Q_{k-1} - \nabla^2 f\right\|_F$$

对于这样的 Q_k 和任意二次函数 f 都成立.

[1] 后文给出介绍.

[2] 我们使用 “二次函数” 来指代次数不大于 2 的非负整数阶的多项式.

如果 (1-4) 中的 $\nabla^2 Q_{k-1}$ 被换为零矩阵, 即对应最小 Frobenius 范数二次模型 [156, 157].

1.2.2 插值模型的相关概念

如前文所述, 插值模型对于基于模型的信赖域无导数优化算法非常重要, 而提供插值条件的插值集合对于构造插值模型也至关重要. 现在我们介绍插值模型和插值集的相关概念. 首先需要介绍的是我们应该如何定义一个插值点集合的适定性衡量标准. 事实上, 给定一个插值集 $\mathcal{X}$, 一个好的适定性衡量标准应该反映这个集合如何 "覆盖" 了插值所感兴趣、重点关注的区域. 比如说, 在线性情况下, 一个 "好的覆盖" 通常意味着 $\mathcal{X}$ 中的点是仿射独立的.

显然, 这样的度量将依赖于 $\mathcal{X}$ 本身和所考虑的区域. 例如, 在线性插值的情况下, 集合 $\mathcal{X} = \{(0,0)^\top, (0,1)^\top, (1,0)^\top\} \subset \mathfrak{R}^2$ 在 $\mathcal{B}_1(\mathbf{0})$ 球内是一个良好的适定集, 但在 $\mathcal{B}_{10^6}(\mathbf{0})$ 球内不是一个良好的适定集[3], 此外, 集合 $\mathcal{X}$ 的适定性也依赖于插值的相应多项式空间.

我们将使用下面所给出的适定集合的定义, 该定义参照 Conn、Scheinberg 和 Vicente 著作 [20] 的定义 3.6 给出.

定义 1.1. *设 $\Lambda > 0$, 集合 $\mathcal{B} \in \mathfrak{R}^n$. 设 $\phi = \{\phi_1(\boldsymbol{x}), \phi_2(\boldsymbol{x}), \cdots, \phi_m(\boldsymbol{x})\}$ 是在次数不高于 d 的 n 维多项式空间 $\mathcal{P}_n^d$ 中的一个基. 集合 $\mathcal{X} = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_m\}$ 被称为在 $\mathcal{B}$ 中 (在插值意义上) 是 Λ-适定的, 当且仅当*

1. 对于与 $\mathcal{X}$ 对应的 Lagrange 多项式的基[4] $l_i(\boldsymbol{x})$, 有

$$\Lambda \geq \max_{1\leq i\leq m} \max_{\boldsymbol{x}\in\mathcal{B}} \left|l_i(\boldsymbol{x})\right|,$$

或者, 等价地,

2. 对于任意 $\boldsymbol{x} \in \mathcal{B}$, 存在 $\boldsymbol{\lambda}(\boldsymbol{x}) \in \mathfrak{R}^m$ 使得

$$\sum_{i=1}^{m} \lambda_i(\boldsymbol{x})\phi(\boldsymbol{y}_i) = \phi(\boldsymbol{x}) \text{ 且 } \|\lambda(\boldsymbol{x})\|_\infty \leq \Lambda,$$

其中 λ_i 表示此处的 $\boldsymbol{\lambda}$ 的第 i 个元素.

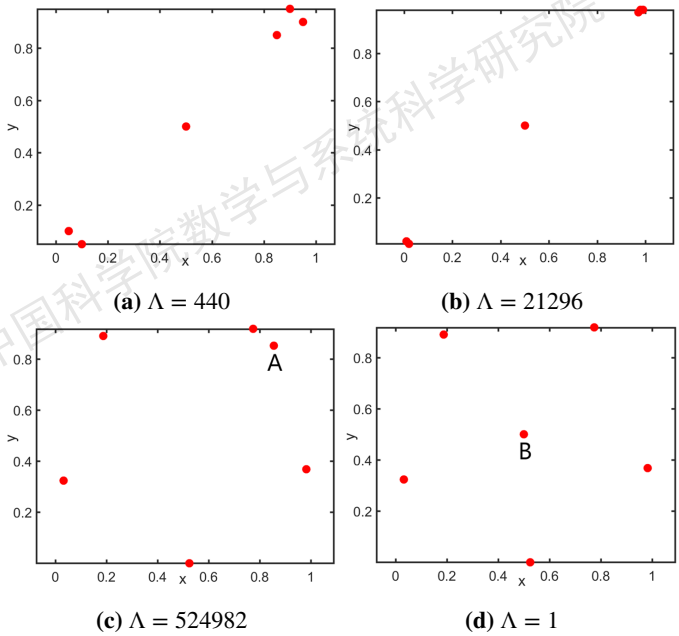
或者, 等价地,

3. 用 $\mathcal{B}$ 中的任意一点 $\boldsymbol{x}$ 替换 $\mathcal{X}$ 中的任意点, 最多可以将集合 $\{\phi(\boldsymbol{y}_1), \phi(\boldsymbol{y}_2), \cdots, \phi(\boldsymbol{y}_m)\}$ 的对应体积扩大 Λ 倍.

为了更形象地展示通过生成插值点来实现对插值集的改进的具体策略, 我们用图 1-1 给出示例 [20], 其中 6 个插值点分布在区域 $[0,1]\times[0,1]$ 上. 我们可以观察到, 图 1-1 中的子图 1-1d 中的适定性常数 Λ 是所有的子图中最小的. 此外, 有时, 当 6 个插值点中的一个点被另一个点替换后, 相应的适定性常数 Λ 可以得到大幅改善 (例如, 图 1-1 的子图 1-1c 中的点 A 被子图 1-1d 中的点 B 替换时).

[3] 这里省略具体原因.

[4] 详见 Conn、Scheinberg 和 Vicente 的著作 [20].



(a) $\Lambda = 440$　　**(b)** $\Lambda = 21296$

(c) $\Lambda = 524982$　　**(d)** $\Lambda = 1$

图 1-1 不同的分布在 $[0,1]\times[0,1]$ 上的适定性常数 Λ

Figure 1-1 The poisedness constants Λ with different distributions on $[0,1]\times[0,1]$

下面我们给出一个在理论上对目标函数 f 的假设, 注意, 该假设用于理论分析, 并不意味着无导数优化方法只能求解满足该假设条件的问题.

假设 1.2. 假设给定一个集合 S 和一个半径 $\Delta_{\max}$. 假设 f 在包含集合 S 的邻域 $\bigcup_{\boldsymbol{x}\in S}\mathcal{B}_{\Delta_{\max}}(\boldsymbol{x})$ 的适当开邻域内是连续可微的, 并且其梯度 Lipschitz 连续.

下面我们给出本论文需要提及的完全线性模型的定义.

定义 1.3 (完全线性模型, 详见 Conn、Scheinberg 和 Vicente 著作 [20] 的定义 6.1)**.** *给定一个满足假设 1.2 的函数 $f:\mathfrak{R}^n\to\mathfrak{R}$. 我们称一类模型函数 $\{Q:\mathfrak{R}^n\to\mathfrak{R}, Q\in C^1\}$ 为完全线性模型类, 如果它满足以下条件:*

1. 存在正常数 κ_{ef},κ_{eg} 和 ν_Q, 使得对于任意 $\boldsymbol{x}\in S$ 和 $\Delta\in(0,\Delta_{\max}]$, 存在一个在该类中的模型函数 $Q(\boldsymbol{y})$, 其梯度连续且对应的 Lipschitz 常数有上界 ν_Q, 并且满足: 模型梯度与函数 f 的梯度之间的误差满足

$$\|\nabla f(\boldsymbol{y})-\nabla Q(\boldsymbol{y})\|_2\le\kappa_{eg}\Delta,\ \forall\,\boldsymbol{y}\in\mathcal{B}_\Delta(\boldsymbol{x}),$$

同时, 模型与函数 f 之间的误差满足

$$|f(\boldsymbol{y})-Q(\boldsymbol{y})|\le\kappa_{ef}\Delta^2,\ \forall\,\boldsymbol{y}\in\mathcal{B}_\Delta(\boldsymbol{x}).$$

这样的模型 Q 被称为在 $\mathcal{B}_\Delta(\boldsymbol{x})$ 上完全线性.

2. 对于这个类, 存在一个算法, 我们将其称为“模型改进”算法 (见 Conn、Scheinberg 和 Vicente 著作 [20] 的第 6 章), 它可以在有限的、相对于 $\boldsymbol{x}$ 和 Δ 一致有界的步数内满足如下任意一条:

- 确定一个给定的该类中的模型 Q 在 $\mathcal{B}_\Delta(\boldsymbol{x})$ 上是完全线性的;

- 找到一个在 $\mathcal{B}_\Delta(\boldsymbol{x})$ 上完全线性的该类中的模型 $\tilde{Q}$.

1.3 无导数优化算法的评价方法

本节介绍无导数优化算法的评价方法. 事实上, 评价和比较不同算法的优劣是优化算法研究中的重要内容, 这也关系着我们如何判定算法设计的好坏. 为了指导面对实际应用时的算法选择和在研究领域的算法具体改进, 我们需要建立和使用可信赖的评价体系. 这里我们将介绍除了传统的观察算法的迭代函数值曲线之外的评价无导数优化方法的办法, 这里所介绍的办法将在后续章节中用于算法的比较. 需要重点说明的是, 无导数优化中, 我们通常关注函数值探测次数或探测点个数, 而非算法的迭代轮数.

无导数优化算法最常见的评价标准是 Performance Profile [21, 164, 165] 和 Data Profile [21, 165]. 它们是通过求解测试问题集来比较无导数算法的两种最常见方式. 两种 Profile 帮助我们以紧凑的图形来展示无导数优化算法的收敛速度和成功求解问题的比例等信息.

假设 $\boldsymbol{x}_N$ 是算法在 N 次函数值探测后找到的最优点, $\boldsymbol{x}_{\text{int}}$ 是初始点, 而 $\boldsymbol{x}^*$ 是已知的最佳解. 给定精度 $\tau \in [0,1]$, 我们定义

$$T_{a,p} = \begin{cases} 1, \text{如果对某些 } N \text{ 有 } f(\boldsymbol{x}_N) \le f(\boldsymbol{x}^*) + \tau \left(f(\boldsymbol{x}_{\text{int}}) - f(\boldsymbol{x}^*) \right), \\ 0, \text{否则}, \end{cases}$$

其中 a 表示对应的算法, p 表示对应问题. 注意, 在本论文的比较中, $\boldsymbol{x}^*$ 通过数值实验获得.

我们先给出 Performance Profile 的定义. 在 Performance Profile 中, 函数 $\rho_a : [1,\infty) \mapsto [0,1]$, 对应的是算法 $a \in \mathcal{A}$ 在测试集 $\mathcal{P}$ 上成功求解问题的比例, 其定义为

$$\rho_a(\alpha) = \frac{1}{|\mathcal{P}|} \left| \left\{ p \in \mathcal{P} : r_{a,p} \le \alpha \right\} \right|,$$

其中

$$r_{a,p} = \begin{cases} \dfrac{N_{a,p}}{\min \left\{ N_{\tilde{a},p} : \tilde{a} \in \mathcal{A}, T_{\tilde{a},p} = 1 \right\}}, & \text{如果 } T_{a,p} = 1, \\ \infty, & \text{如果 } T_{a,p} = 0, \end{cases}$$

$$N_{a,p} = \min \left\{ N \in \mathbb{N}^+, f(\boldsymbol{x}_N) \le f(\boldsymbol{x}^*) + \tau \left(f(\boldsymbol{x}_{\text{int}}) - f(\boldsymbol{x}^*) \right) \right\}.$$

注意, 这里的符号 $|\cdot|$ 表示对应集合中的元素数量.

以上 Profile 是通过为 $\mathcal{A}$ 中所有算法绘制函数 ρ_a 创建的. 它试图考察和展现算法的效率和鲁棒性. Performance Profile 的特点是: 较高的曲线对应该算法有较好的求解表现, 这里我们用 NF 表示求解时所用的函数值探测次数.

另外, 我们使用 Data Profile [165] 提供一些原始信息 (Performance Profile 侧重于比较不同的算法, 而 Data Profile 体现了求解测试问题集所需的函数值探测次数), 这对于具有特定计算预算并需要选择一个可能达到给定函数值下降量的算法的用户有参考价值. 在 Data Profile 中,

$$\delta_a(\beta) = \frac{1}{|\mathcal{P}|} \left| \left\{ p \in \mathcal{P} : N_{a,p} \le \beta (n+1) T_{a,p} \right\} \right|,$$

$\delta_a(\beta)$ 的值越高代表成功求解的问题越多.

本论文将会用到这两种评价方法来对包括新设计的算法在内的不同算法进行比较. 另外需要说明的是, 本论文的一些 Profile 中会对横坐标进行对数尺度的变换, 以此展示我们更关注的对比内容和区域.

1.4　论文主要内容

本论文将试图围绕无导数优化针对以下问题进行分析和回答.

(1) 如何为基于模型的无导数优化方法设计更好的逼近模型?

(2) 模型和基于模型的方法受带变换、噪声的输出函数值的影响如何?

(3) 如何更有效地求解大规模无导数优化问题?

(4) 二次插值模型是否可以改进带有或使用近似一阶导数的线搜索方法?

关于问题 (3), 我们知道基于模型的信赖域方法是求解非线性规划问题的一类成熟算法. 其中大部分算法使用二次模型, 这是因为二次模型能够有效地拟合目标函数的曲率信息. 然而, 对于现有的基于模型的无导数优化方法, 求解大规模问题仍然是一个瓶颈, 这是因为在问题维数较高时, 构建局部 (多项式) 模型的计算成本和插值误差可能很高, 导致实际求解效果不好. 这可以被看作是无导数优化中的 “维数灾难”. 传统的无导数优化方法使用二次插值模型来逼近目标函数, 但黑箱目标函数有限的可用信息会导致这些模型的可信度较低. 目前, 已经有学者提出和开发了一些方法来处理大规模问题, 其中一个重要的方法是使用子空间方法, 它的主要思想是在每次迭代中在一个低维子空间中极小化目标函数, 以获得下一个迭代点 [35, 141, 166–169]. 本论文将介绍围绕利用子空间方法求解大规模无导数优化问题所进行的研究.

具体而言, 本论文剩下内容的主要脉络如下: 第 2 章主要围绕无导数信赖域算法中二次插值模型的改进展开. 其中, 我们介绍了最小 H^2 范数更新二次插值模型, 包括使用 H^2 范数构造最小范数二次模型的动机及模型性质、最小 H^2 范数更新二次模型公式等, 并给出了相应的数值结果和小结. 同时, 探讨了最小加权 H^2 范数更新二次插值模型, 包括最小加权 H^2 范数更新二次模型和 KKT 矩阵、KKT 矩阵误差、最小加权 H^2 范数更新二次模型的权重系数区域的重心等, 并给出了相应的数值结果和小结. 此外, 我们介绍了使用新的欠定二次插值模型的无导数方法, 包括其背景和动机、考虑了前一信赖域迭代性质的模型细节、获取对应模型的子问题的凸性和模型的计算公式以及相应的数值结果. 我们还介绍了信赖域中非凸二次函数极小点之间距离减小的条件, 并给出了相应的数值结果和小结.

第 3 章介绍了带变换目标函数的无导数优化及基于最小 Frobenius 范数更新二次模型的算法. 具体包括了相应的问题、基于模型的信赖域优化算法和探测方案、变换后目标函数的最小 Frobenius 范数更新二次模型、信赖域子问题、保最优性变换等内容. 此外, 还探讨了正单调变换与仿射变换对应的模型性质, 以及相

应的完全线性模型和收敛性分析, 同时给出了相应的测试问题和实际问题的数值结果和小结.

第 4 章介绍了子空间方法和并行方法. 我们针对大规模问题, 提出了一个新的无导数子空间信赖域方法: 2D-MoSub 算法, 并具体给出了 2D-MoSub 算法、插值集的适定性和质量、2D-MoSub 的一些性质和数值结果等内容. 同时, 我们提出了一个新的结合线搜索和信赖域技术的 (可并行) 无导数优化算法, 内容包括其背景和动机、所用的 SUSD 方向与信赖域插值结合的具体过程、SUSD-TR 算法的迭代方向的理论分析、试探步和结构步以及数值结果等.

最后一章对全文进行了总结, 并展望了未来的研究方向.

中国科学院数学与系统科学研究院 XiePengCheng 2025-07-31 09:40:21

第 2 章　无导数信赖域算法中二次插值模型的改进

2.1　无导数信赖域算法中二次插值模型的最小 H^2 范数更新

本节提出的方法和模型是对 Powell 的无导数优化算法和模型 [92, 94, 170] 的改进. Powell 方法的主要思想是在每次迭代中通过欠定插值获得二次模型函数, 并对前一个模型进行更新, 具体来说, 是在第 k 次迭代中通过求解优化问题

$$\begin{aligned} &\min_{Q\in\mathcal{Q}} \left\|\nabla^2 Q-\nabla^2 Q_{k-1}\right\|_F^2 \\ &\text{s. t. } Q(\boldsymbol{y}_i)=f(\boldsymbol{y}_i),\ \boldsymbol{y}_i\in\mathcal{X}_k \end{aligned} \tag{2-1}$$

来获得唯一的二次模型 Q_k. 我们记 $\mathcal{X}_k$ 中的插值点个数为 m, 该方法期望通过在信赖域内极小化二次模型 Q_k 来获得新的迭代点. Conn 和 Toint [171]、Conn、Scheinberg 和 Toint [160, 172]、Wild [157] 提出可以在 (2-1) 中选择极小化目标函数 $\left\|\nabla^2 Q\right\|_F^2+\|\nabla Q\|_2^2$ 或 $\left\|\nabla^2 Q\right\|_F^2$. Bandeira、Scheinberg 和 Vicente [153] 讨论了在具有稀疏结构的问题中通过最小化 $\left\|\operatorname{vec}\left(\nabla^2 Q\right)\right\|_1$ 来获取欠定二次插值模型. 这里符号 vec 表示矩阵的向量化, 将矩阵转换为向量. 具体来说, 对于矩阵 $\boldsymbol{C}\in\mathfrak{R}^{m\times n}$, $\operatorname{vec}(\boldsymbol{C})$ 表示通过将矩阵 $\boldsymbol{C}$ 的列堆叠在一起获得的向量.

如 Conn、Scheinberg 和 Vicente [20] 所展示的, 为了确保获得具有完全线性性质的插值模型 (参考 Conn、Scheinberg 和 Vicente 著作 [20] 中的定义 6.1), 至少需要 $n+1$ 个插值点, 这在无导数优化应用中是昂贵的. 实际上, 我们发现 $n+1$ 个插值条件, 即 $n+1$ 个等式约束条件, 可以放宽为 “使平均插值误差在某一区域内整体较小”. 我们使用 H^2 范数来衡量和控制插值误差. 我们提出的最小 H^2 范数更新二次模型函数可以减小构造模型所需插值点数量的下界, 同时可以在局部控制插值误差. 这是第一次在无导数方法或信赖域方法中使用 H^2 范数来构造欠定二次模型. 在此之前, 张在坤 [46, 154] 讨论了 H^1 半范数的相关使用, 第 2.1.2 节开头将展示更多细节.

本部分的其余内容按照如下进行组织. 第 2.1.1 节介绍了关于二次函数的 H^2 范数的一些基本结果. 第 2.1.2 节讨论了使用最小 H^2 范数更新二次模型函数的动机、H^2 范数意义下的投影理论以及插值误差界. 第 2.1.3 节给出了最小 H^2 范数更新二次模型函数, 详细介绍了获取二次模型所需要求解的优化问题的 KKT 条件. 此外, 我们提供了相关的实现细节, 包括 KKT 逆矩阵的更新公式, 以及极大化更新公式分母的模型改进步. 第 2.1.4 节展示了数值结果. 最后, 给出了小结和一些可能的未来工作.

2.1.1　H^2 范数与无导数信赖域算法

我们首先给出相关符号, 然后再进行更多讨论. 这里我们定义 $\Omega=\mathcal{B}_r(\boldsymbol{x}_0)=\left\{\boldsymbol{x}\in\mathfrak{R}^n:\left\|\boldsymbol{x}-\boldsymbol{x}_0\right\|_2\leq r\right\}$, $\boldsymbol{H}=\nabla^2 Q\in\mathfrak{R}^{n\times n}$, $\boldsymbol{g}=\nabla Q(\boldsymbol{x}_0)\in\mathfrak{R}^n$, $c\in\mathfrak{R}$. 注意, 若不加说明, $\|\cdot\|$ 表示对应向量的 ℓ_2 范数. 下面给出了不同种类的 (半) 范数的定义.

定义 2.1. 假设 u 是在 $\Omega \subseteq \Re^n$ 上的函数, 且 $1 \leq p < \infty$. 若 u 在 Ω 上二次可微且对于任意满足 $a \leq 2$ 的自然数 a, 有 $\frac{\partial^a u}{\partial \boldsymbol{x}^a} \in L^2(\Omega)$, 则我们有

$$\|u\|_{H^0(\Omega)} = \left(\int_\Omega |u(\boldsymbol{x})|^2 \, d\boldsymbol{x} \right)^{\frac{1}{2}},$$
$$|u|_{H^1(\Omega)} = \left(\int_\Omega \|\nabla u(\boldsymbol{x})\|_2^2 \, d\boldsymbol{x} \right)^{\frac{1}{2}},$$
$$|u|_{H^2(\Omega)} = \left(\int_\Omega \left\|\nabla^2 u(\boldsymbol{x})\right\|_F^2 \, d\boldsymbol{x} \right)^{\frac{1}{2}}.$$

此外, 函数 u 的 H^2 范数的定义是

$$\|u\|_{H^2(\Omega)} = \left(\|u\|_{H^0(\Omega)}^2 + |u|_{H^1(\Omega)}^2 + |u|_{H^2(\Omega)}^2 \right)^{\frac{1}{2}}.$$

注意, 我们用 $|\cdot|$ 表示半范数, 用 $\|\cdot\|$ 表示范数 (搭配对应的脚标). 基于定义 2.1, 简单计算后可得出以下关于二次函数的 H^2 范数的定理.

注 2.1. 点 $\boldsymbol{x}_0$ 表示计算 H^2 范数的区域的中心点. $\boldsymbol{x}_0$ 有时被称为基点 [94].

定理 2.2. 给定二次函数

$$Q(\boldsymbol{x}) = \frac{1}{2} \left(\boldsymbol{x} - \boldsymbol{x}_0\right)^\top \boldsymbol{H} \left(\boldsymbol{x} - \boldsymbol{x}_0\right) + \left(\boldsymbol{x} - \boldsymbol{x}_0\right)^\top \boldsymbol{g} + c,$$

我们有

$$\begin{aligned} \|Q\|_{H^2(\mathcal{B}_r(\boldsymbol{x}_0))}^2 =& \mathcal{V}_n r^n \left[\left(\frac{r^4}{2(n+4)(n+2)} + \frac{r^2}{n+2} + 1 \right) \|\boldsymbol{H}\|_F^2 + \left(\frac{r^2}{n+2} + 1 \right) \|\boldsymbol{g}\|_2^2 \right. \\ & \left. + \frac{r^4}{4(n+4)(n+2)} (\operatorname{Tr}(\boldsymbol{H}))^2 + \frac{r^2}{n+2} c \operatorname{Tr}(\boldsymbol{H}) + c^2 \right], \end{aligned} \tag{2-2}$$

其中 $\mathcal{V}_n$ 表示 n 维 ℓ_2 范数单位球 $\mathcal{B}_1(\boldsymbol{x}_0)$ 的体积, $\operatorname{Tr}(\cdot)$ 表示对应矩阵的迹.

证明. 经过直接计算 (计算细节见张在坤的论文 [46]) 可以得到

$$\begin{aligned} \|Q(\boldsymbol{x})\|_{H^0(\mathcal{B}_r(\boldsymbol{x}_0))}^2 =& \mathcal{V}_n r^n \left(\frac{2r^4}{4(n+2)(n+4)} \|\boldsymbol{H}\|_F^2 + \frac{r^2}{n+2} \|\boldsymbol{g}\|_2^2 \right. \\ & \left. + \frac{r^4}{4(n+2)(n+4)} (\operatorname{Tr}(\boldsymbol{H}))^2 + \frac{r^2}{n+2} c \operatorname{Tr}(\boldsymbol{H}) + c^2 \right), \end{aligned}$$

以及

$$|Q|_{H^1(\mathcal{B}_r(\boldsymbol{x}_0))}^2 = \mathcal{V}_n r^n \left(\frac{r^2}{n+2} \|\boldsymbol{H}\|_F^2 + \|\boldsymbol{g}\|_2^2 \right).$$

此外, 我们有 $|Q|_{H^2(\mathcal{B}_r(\boldsymbol{x}_0))}^2 = \mathcal{V}_n r^n \|\boldsymbol{H}\|_F^2$. 因此, (2-2) 得证. □

考虑到本节所给出的插值模型函数适用于基于插值模型的一般无导数信赖域算法, 我们在给出更多细节之前, 先介绍基于模型的无导数信赖域算法的一般框架[1], 即算法 3. 注意, 本节没有给出新的框架, 而是专注于新的二次模型 (可用于一般的信赖域框架). 这里, 我们仍然保持 “接受模型” 的说法, 而不是 “接受插值集”, 这样是为了保持一致性. 更多关于无导数信赖域方法的介绍可以参见 Larson、Menickelly 和 Wild 的综述 [128]、Conn、Scheinberg 和 Vicente [20] 以及 Audet 和 Hare [21] 的著作等.

算法 3 基于模型的无导数信赖域算法框架

输入: 黑箱目标函数 f 和初始点 $\boldsymbol{x}_{\text{int}}$.

输出: 极小点 $\boldsymbol{x}^*$ 和函数极小值.

初始化并获取插值集 $\mathcal{X}_0$、初始二次模型函数 $Q_0(\boldsymbol{x})$ (在当前点的梯度记作 $\boldsymbol{g}_0$), 以及参数 $\Delta_0, \gamma, \varepsilon_c, \mu, \hat{\eta}_1, \hat{\eta}_2$. 令 $k = 0$.

步 1 (判别步): 若 $\|\boldsymbol{g}_k\| > \varepsilon_c$, 则接受 $\boldsymbol{g}_k$, Δ_k 和模型 Q_k. 若 $\|\boldsymbol{g}_k\| \leq \varepsilon_c$, 则调用模型改进步并判别当前信赖域上的模型是否被接受. 若模型 Q_k 无法被接受或 $\Delta_k > \mu\|\boldsymbol{g}_k\|$, 则使用模型改进步构造一个可接受的模型, 然后相应调整半径 Δ_k.

步 2 (试探步): 求解

$$\min_{\boldsymbol{d}} Q_k(\boldsymbol{x}_{\text{opt}} + \boldsymbol{d})$$
$$\text{s. t. } \|\boldsymbol{d}\|_2 \leq \Delta_k,$$

其中 $\boldsymbol{x}_{\text{opt}}$ 是当前迭代所有插值点中函数值最小的点, 并获得解 $\boldsymbol{d}_k$.

步 3 (试探点的接受): 计算 $f(\boldsymbol{x}_{\text{opt}} + \boldsymbol{d}_k)$ 并定义

$$\rho_k = \frac{f(\boldsymbol{x}_{\text{opt}}) - f(\boldsymbol{x}_{\text{opt}} + \boldsymbol{d}_k)}{Q_k(\boldsymbol{x}_{\text{opt}}) - Q_k(\boldsymbol{x}_{\text{opt}} + \boldsymbol{d}_k)},$$

若 $\rho_k \geq \hat{\eta}_1$, 或 $\rho_k > \hat{\eta}_2$ 且模型被接受, 则令 $\boldsymbol{x}_{k+1} = \boldsymbol{x}_{\text{opt}} + \boldsymbol{d}_k$, 更新模型和样本/插值集, 并获得 Q_{k+1} (当前点的梯度 $\boldsymbol{g}_{k+1}$) 和 $\mathcal{X}_{k+1} = \mathcal{X}_k \cup \{\boldsymbol{y}_{\text{new}}\} \backslash \{\boldsymbol{y}_t\}$, 其中 $\boldsymbol{y}_{\text{new}} = \boldsymbol{x}_{k+1}$ 是新的迭代/插值点, 最远点 $\boldsymbol{y}_t$ 在此步中被插值集舍弃. 否则, 模型和迭代保持不变, 令 $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k$.

步 4 (模型改进步): 若 $\rho_k < \hat{\eta}_1$ 且模型未被接受, 则改进模型. (测试版本在检查基于插值适定性的接受情况后迭代更新 KKT 逆矩阵, 其遵循第 2.1.3 节末尾的描述. 更多细节由 Conn、Scheinberg 和 Vicente 著作 [20] 的第 6 章给出.) 定义 Q_{k+1} 和 $\mathcal{X}_{k+1}$ 为 (可能改进了的) 新模型和样本/插值集.

步 5 (信赖域半径更新): 根据 ρ_k 和 Δ_k 更新 Δ_{k+1}, 例如, 若 $\rho_k < \hat{\eta}_1$ 且模型被接受, 则令 $\Delta_{k+1} = \frac{1}{\gamma}\Delta_k$; 若 $\rho_k \geq \hat{\eta}_1$, 则令 $\Delta_{k+1} = \gamma\Delta_k$; 在其他情况下, 令 $\Delta_{k+1} = \Delta_k$.

令 $k = k + 1$, 然后回到**步 1**, 直到 $\Delta_k < \varepsilon_c$ 且 $\|\boldsymbol{g}_k\| < \varepsilon_c$.

[1]本节的测试代码框架基于 Conn、Scheinberg 和 Vicente 著作 [20] 中的算法 10.1. 本节专注于我们的新二次模型及其计算.

2.1.2 使用 H^2 范数构造最小范数二次模型的动机及模型性质

本节介绍使用 H^2 范数获取目标函数逼近模型的动机. 张在坤 [46, 154] 的发现让我们开始关注在点处使用范数度量与在区域内使用平均意义或整体意义下的范数度量之间的关系. 此外, Powell 等人提出了通过求解 (2-1) 获得最小 Frobenius 范数更新二次模型函数的方法, 展示了最小范数更新二次模型的优势. 注意, 对于 Powell 的模型, 插值点或方程的数量有一个下界 [94]. 此外, 大多数现有模型仅通过在多点处的插值获得, 并没有考虑在 (信赖) 区域内目标函数或导数的平均值. 在介绍如何获得我们的新模型之前, 我们先给出目标函数 (2-3) 的凸性.

定理 2.3. *给定 $C_1, C_2, C_3 > 0$, 函数*

$$C_1 \|Q(\boldsymbol{x})\|^2_{H^0(\Omega)} + C_2 |Q(\boldsymbol{x})|^2_{H^1(\Omega)} + C_3 |Q(\boldsymbol{x})|^2_{H^2(\Omega)} \tag{2-3}$$

作为 Q 的函数是严格凸的.

证明. 我们有

$$\begin{aligned} &C_1 \|Q(\boldsymbol{x})\|^2_{H^0(\Omega)} + C_2 |Q(\boldsymbol{x})|^2_{H^1(\Omega)} + C_3 |Q(\boldsymbol{x})|^2_{H^2(\Omega)} \\ =& C_1 \int_\Omega |Q(\boldsymbol{x})|^2 \, d\boldsymbol{x} + C_2 \int_\Omega \|\nabla Q(\boldsymbol{x})\|_2^2 \, d\boldsymbol{x} + C_3 \int_\Omega \left\|\nabla^2 Q(\boldsymbol{x})\right\|_F^2 d\boldsymbol{x}, \end{aligned}$$

我们需要证明不等式

$$\begin{aligned} &\left[\mu \left(C_1 \left\|Q_a(\boldsymbol{x})\right\|^2_{H^0(\Omega)} + C_2 \left|Q_a(\boldsymbol{x})\right|^2_{H^1(\Omega)} + C_3 \left|Q_a(\boldsymbol{x})\right|^2_{H^2(\Omega)}\right)\right. \\ &\left. + (1-\mu) \left(C_1 \left\|Q_b(\boldsymbol{x})\right\|^2_{H^0(\Omega)} + C_2 \left|Q_b(\boldsymbol{x})\right|^2_{H^1(\Omega)} + C_3 \left|Q_b(\boldsymbol{x})\right|^2_{H^2(\Omega)}\right)\right] \\ &- \left[C_1 \left\|\mu Q_a(\boldsymbol{x}) + (1-\mu) Q_b(\boldsymbol{x})\right\|^2_{H^0(\Omega)} + C_2 \left|\mu Q_a(\boldsymbol{x}) + (1-\mu) Q_b(\boldsymbol{x})\right|^2_{H^1(\Omega)}\right. \\ &\left. + C_3 \left|\mu Q_a(\boldsymbol{x}) + (1-\mu) Q_b(\boldsymbol{x})\right|^2_{H^2(\Omega)}\right] > 0 \end{aligned} \tag{2-4}$$

在 $Q_a, Q_b \in \mathcal{Q}$, 且 $0 < \mu < 1$, $C_1, C_2, C_3 > 0$ 的情况下成立.

事实上, 不等式 (2-4) 的左侧可以转化为

$$\begin{aligned} &(\mu - \mu^2) \left(C_1 \int_\Omega \left|Q_a(\boldsymbol{x}) - Q_b(\boldsymbol{x})\right|^2 d\boldsymbol{x} + C_2 \int_\Omega \left\|\nabla Q_a(\boldsymbol{x}) - \nabla Q_b(\boldsymbol{x})\right\|_2^2 d\boldsymbol{x}\right. \\ &\left. + C_3 \int_\Omega \left\|\nabla^2 Q_a(\boldsymbol{x}) - \nabla^2 Q_b(\boldsymbol{x})\right\|_F^2 d\boldsymbol{x}\right) > 0. \end{aligned}$$

综上, 我们证明了 (2-3) 作为 Q 的函数是严格凸的. □

根据定理 2.3, 优化问题

$$\begin{aligned} &\min_{Q \in \mathcal{Q}} \ \left\|Q - Q_{k-1}\right\|^2_{H^2(\mathcal{B}_r(\boldsymbol{x}_0))} \\ &\text{s. t. } Q(\boldsymbol{y}) = f(\boldsymbol{y}), \ \boldsymbol{y} \in \mathcal{X}_k \end{aligned} \tag{2-5}$$

的解二次模型函数 $Q(\boldsymbol{x})$ 是唯一的.

此外, 我们还可以得到以下结果.

注 2.2. 获取最小加权 H^2 范数更新模型的优化问题

$$\begin{aligned}&\min_{Q\in\mathcal{Q}} C_1\left\|Q-Q_{k-1}\right\|^2_{H^0(\mathcal{B}_r(\boldsymbol{x}_0))}+C_2\left|Q-Q_{k-1}\right|^2_{H^1(\mathcal{B}_r(\boldsymbol{x}_0))}+C_3\left|Q-Q_{k-1}\right|^2_{H^2(\mathcal{B}_r(\boldsymbol{x}_0))}\\&\text{s. t. } Q(\boldsymbol{y})=f(\boldsymbol{y}),\ \boldsymbol{y}\in\mathcal{X}_k\end{aligned}\tag{2-6}$$

的解是唯一的, 其中 $C_1, C_2, C_3>0$.

容易知道, 二次模型函数在 $\mathcal{B}_r(\boldsymbol{x}_0)$ 上的 H^1 半范数与二次模型函数在区域 $\mathcal{B}_r(\boldsymbol{x}_0)$ 上一阶导数相应范数的平均值对应, 而在 $\mathcal{B}_r(\boldsymbol{x}_0)$ 上的 H^2 半范数与二次模型函数在区域 $\mathcal{B}_r(\boldsymbol{x}_0)$ 上二阶导数相应范数的平均值对应. 事实上, 最小化二次模型函数值的插值误差也很重要. 最小化 L^2 范数可以利用减小函数值的插值误差来达到插值条件 $Q(\boldsymbol{y}_i)=f(\boldsymbol{y}_i)$ 的一部分作用, 进而放宽插值点数量的下界. 根据接下来介绍的投影性质, 我们的确有理由在 (2-6) 的目标函数中引入 H^0 范数, 或者说 L^2 范数. H^0 范数、H^1 半范数和 H^2 半范数的 (加权) 总和蕴含着我们同时最小化模型函数的平均函数误差、一阶导数的平均误差和二阶导数的平均误差的目标. 这样可以减小插值点数量的下界, 即 m 可以小于 $n+1$. 第 2.1.4 节中的数值结果也支持了我们选择使用 H^2 范数来获取模型函数.

下面给出最小 H^2 范数更新二次模型的投影性质, 同时我们将证明其在局部具有插值误差上界.

定理 2.4. *设 Q_k 是问题 (2-5) 的解. 若 f 是二次函数, 则*

$$\left\|Q_k-f\right\|^2_{H^2(\mathcal{B}_r(\boldsymbol{x}_0))}=\left\|Q_{k-1}-f\right\|^2_{H^2(\mathcal{B}_r(\boldsymbol{x}_0))}-\left\|Q_k-Q_{k-1}\right\|^2_{H^2(\mathcal{B}_r(\boldsymbol{x}_0))}.\tag{2-7}$$

证明. 在第 k 次迭代时, 对于任意的 $\xi\in\mathfrak{R}$, 令 Q_ξ 为 $Q_k+\xi\left(Q_k-f\right)$. 则 Q_ξ 是一个满足 (2-5) 中插值条件的插值函数. 因此, 根据 Q_k 的最优性, $\varphi(\xi)=\left\|Q_\xi-Q_{k-1}\right\|^2_{H^2(\mathcal{B}_r(\boldsymbol{x}_0))}$ 在 $\xi=0$ 时具有最小值. 另外, 我们有

$$\begin{aligned}\varphi(\xi)&=\left\|Q_k+\xi\left(Q_k-f\right)-Q_{k-1}\right\|^2_{H^2(\mathcal{B}_r(\boldsymbol{x}_0))}\\&=\xi^2\left\|Q_k-f\right\|^2_{H^2(\mathcal{B}_r(\boldsymbol{x}_0))}+\left\|Q_k-Q_{k-1}\right\|^2_{H^2(\mathcal{B}_r(\boldsymbol{x}_0))}\\&+2\xi\Big\{\int_{\mathcal{B}_r(\boldsymbol{x}_0)}\left(Q_k(\boldsymbol{x})-Q_{k-1}(\boldsymbol{x})\right)\cdot\left(Q_k(\boldsymbol{x})-f(\boldsymbol{x})\right)d\boldsymbol{x}\\&+\int_{\mathcal{B}_r(\boldsymbol{x}_0)}\left(\nabla Q_k(\boldsymbol{x})-\nabla Q_{k-1}(\boldsymbol{x})\right)^{\top}\left(\nabla Q_k(\boldsymbol{x})-\nabla f(\boldsymbol{x})\right)d\boldsymbol{x}\\&+\int_{\mathcal{B}_r(\boldsymbol{x}_0)}(1,\cdots,1)\left(\nabla^2 Q_k(\boldsymbol{x})-\nabla^2 Q_{k-1}(\boldsymbol{x})\right)\circ\left(\nabla^2 Q_k(\boldsymbol{x})-\nabla^2 f(\boldsymbol{x})\right)(1,\cdots,1)^{\top}d\boldsymbol{x}\Big\},\end{aligned}\tag{2-8}$$

其中符号 $\circ$ 表示 Hardamard 积. 因此 (2-8) 中最后一括号内的项为 0. 考虑 $\varphi(-1)$, 定理得证.

□

根据 (2-7), 我们可以得到 H^2 范数意义下的关系

$$\left\|Q_k(\boldsymbol{x})-f(\boldsymbol{x})\right\|_{H^2(\mathcal{B}_r(\boldsymbol{x}_0))} \leq \left\|Q_{k-1}(\boldsymbol{x})-f(\boldsymbol{x})\right\|_{H^2(\mathcal{B}_r(\boldsymbol{x}_0))}.$$

这在一定程度上意味着模型函数 Q_k 关于所逼近的目标函数 f 具有比 Q_{k-1} 更准确的函数值和梯度 ($Q_k = Q_{k-1}$ 的情形除外). 事实上, 我们可以直接从定理 2.4 得到以下推论.

推论 2.5. *设 Q_k 是问题 (2-6) 的解. 若 f 是二次函数, 则*

$$\begin{aligned}
&C_1\left\|Q_k-f\right\|^2_{H^0(\mathcal{B}_r(\boldsymbol{x}_0))}+C_2\left|Q_k-f\right|^2_{H^1(\mathcal{B}_r(\boldsymbol{x}_0))}+C_3\left|Q_k-f\right|^2_{H^2(\mathcal{B}_r(\boldsymbol{x}_0))}\\
&=C_1\left\|Q_{k-1}-f\right\|^2_{H^0(\mathcal{B}_r(\boldsymbol{x}_0))}+C_2\left|Q_{k-1}-f\right|^2_{H^1(\mathcal{B}_r(\boldsymbol{x}_0))}+C_3\left|Q_{k-1}-f\right|^2_{H^2(\mathcal{B}_r(\boldsymbol{x}_0))}\\
&-C_1\left\|Q_k-Q_{k-1}\right\|^2_{H^0(\mathcal{B}_r(\boldsymbol{x}_0))}-C_2\left|Q_k-Q_{k-1}\right|^2_{H^1(\mathcal{B}_r(\boldsymbol{x}_0))}-C_3\left|Q_k-Q_{k-1}\right|^2_{H^2(\mathcal{B}_r(\boldsymbol{x}_0))},
\end{aligned}$$

其中 $C_1, C_2, C_3 > 0$.

证明. 证明与定理 2.4 的证明类似. □

模型函数对目标函数的逼近对我们基于模型的优化算法至关重要. 接下来我们给出对插值模型的误差分析. 首先, 我们给出以下两个引理.

引理 2.6 (插值不等式)**.** *假设 $1 \leq c_1 \leq c_2 \leq c_3 \leq \infty$, 且 $\frac{1}{c_2} = \frac{\theta}{c_1} + \frac{(1-\theta)}{c_3}$. 假设 $u \in L^{c_1}(\Omega) \cap L^{c_3}(\Omega)$. 则 $u \in L^{c_2}(\Omega)$, 且 $\|u\|_{L^{c_2}(\Omega)} \leq \|u\|^{\theta}_{L^{c_1}(\Omega)}\|u\|^{1-\theta}_{L^{c_3}(\Omega)}$.*

证明. 证明细节见 Evans 的著作 [173]. □

定义 2.7 (Sobolev 空间)**.** *Sobolev 空间 $\mathcal{W}^{k,p}(\Omega)$ 包含所有对于满足 $|\alpha| \leq k$ 的 α, $D^{\alpha}u$ 在弱意义[2]上存在并属于 $L^p(\Omega)$ 的在 Ω 上局部可和的函数 $u : \Omega \to \mathfrak{R}$. 在 $p = 2$ 时, 我们通常记 $\mathcal{H}^k(\Omega) = \mathcal{W}^{k,2}(\Omega)$.*

引理 2.8. *假设 $u \in \mathcal{H}^1(\Omega)$, 且 $\left|\partial_i u\right| \leq M_1$, $i = 1, \cdots, n$. 对于 $\forall\, \boldsymbol{y} \in \Omega := \mathcal{B}_r(\boldsymbol{x}_0)$, 我们有*

$$|u(\boldsymbol{y})| \leq \mathcal{V}_n^{-\frac{1}{2}} r^{-\frac{n}{2}}\|u\|_{L^2(\Omega)} + \mathcal{V}_n^{-\frac{1}{p}} r^{\frac{n}{q}+1-n} n^{\frac{1}{q}-\frac{1+\theta}{2}} (n+q-nq)^{-\frac{1}{q}} M_1^{1-\theta}|u|^{\theta}_{H^1(\Omega)}, \tag{2-9}$$

其中 $\frac{1}{p}+\frac{1}{q}=1$, $\theta = \frac{2}{p} \leq 1$, $p > n$, 且 $\mathcal{V}_n$ 表示 n 维 ℓ_2 范数单位球 $\mathcal{B}_1(\boldsymbol{x}_0)$ 的体积.

证明. 我们将函数 u 延拓至区域 Ω 之外, 使得对于 $\forall\, \boldsymbol{y} \notin \mathcal{B}_r(\boldsymbol{x}_0)$, 有 $u(\boldsymbol{y}) = 0$. 对于 $\forall\, \boldsymbol{y} \in \mathcal{B}_r\left(\boldsymbol{x}_0\right)$, 我们有

$$\begin{aligned}
|u(\boldsymbol{y})| &\leq \left|u(\boldsymbol{y}) - \frac{1}{\left|\mathcal{B}_r(\boldsymbol{y})\right|}\int_{\mathcal{B}_r(\boldsymbol{y})} u(\boldsymbol{x})d\boldsymbol{x}\right| + \frac{1}{\left|\mathcal{B}_r(\boldsymbol{y})\right|}\left|\int_{\mathcal{B}_r(\boldsymbol{y})} u(\boldsymbol{x})d\boldsymbol{x}\right|\\
&\leq \frac{1}{\left|\mathcal{B}_r(\boldsymbol{y})\right|}\left(\int_{\mathcal{B}_r(\boldsymbol{y})}|u(\boldsymbol{y})-u(\boldsymbol{x})|\,d\boldsymbol{x} + \left|\int_{\mathcal{B}_r(\boldsymbol{y})} u(\boldsymbol{x})d\boldsymbol{x}\right|\right),
\end{aligned} \tag{2-10}$$

[2]更多细节在 Evans 的著作 [173] 中给出.

其中 $\left|\mathcal{B}_r(\boldsymbol{y})\right|$ 表示球 $\mathcal{B}_r(\boldsymbol{y})$ 的体积. 此外, 在 (2-10) 右侧出现的两项都有上界. 基于 Hölder 不等式, 我们可以得到

$$\begin{aligned}\left|\int_{\mathcal{B}_r(\boldsymbol{y})} u(\boldsymbol{x})d\boldsymbol{x}\right| &\leq \left(\int_{\mathcal{B}_r(\boldsymbol{y})} (u(\boldsymbol{x}))^2 d\boldsymbol{x}\right)^{\frac{1}{2}} \left(\int_{\mathcal{B}_r(\boldsymbol{y})} 1^2 d\boldsymbol{x}\right)^{\frac{1}{2}} \\ &\leq \left|\mathcal{B}_r(\boldsymbol{y})\right|^{\frac{1}{2}} \|u\|_{L^2(\Omega)} = \mathcal{V}_n^{\frac{1}{2}} r^{\frac{n}{2}} \|u\|_{L^2(\Omega)}.\end{aligned}$$

根据 Evans 的著作 [173] 中 Morrey 不等式的证明, 我们有

$$\begin{aligned}\int_{\mathcal{B}_r(\boldsymbol{y})} |u(\boldsymbol{y}) - u(\boldsymbol{x})|\, d\boldsymbol{x} &\leq \frac{r^n}{n} \int_{\mathcal{B}_r(\boldsymbol{y})} \frac{\|\nabla u(\boldsymbol{x})\|_2}{\|\boldsymbol{y}-\boldsymbol{x}\|_2^{n-1}} d\boldsymbol{x} \\ &\leq \frac{r^n}{n} \left(\int_{\mathcal{B}_r(\boldsymbol{y})} \|\nabla u(\boldsymbol{x})\|_2^p \, d\boldsymbol{x}\right)^{\frac{1}{p}} \left(\int_{\mathcal{B}_r(\boldsymbol{y})} \frac{1}{\|\boldsymbol{y}-\boldsymbol{x}\|_2^{(n-1)q}} d\boldsymbol{x}\right)^{\frac{1}{q}},\end{aligned}$$

其中 $\frac{1}{p} + \frac{1}{q} = 1$ 且 $(n-1)(q-1) \in (0,1)$. 我们可以得到

$$\begin{aligned}\left(\int_{\mathcal{B}_r(\boldsymbol{y})} \frac{1}{\|\boldsymbol{y}-\boldsymbol{x}\|_2^{(n-1)q}} d\boldsymbol{x}\right)^{\frac{1}{q}} &= \left(\int_{\mathcal{B}_r(\boldsymbol{0})} \frac{1}{\boldsymbol{z}^{(n-1)q}} d\boldsymbol{z}\right)^{\frac{1}{q}} \\ &= \mathcal{V}_n^{\frac{1}{q}} n^{\frac{1}{q}} (n+q-nq)^{-\frac{1}{q}} r^{\frac{n}{q}+1-n}.\end{aligned}$$

此外, 引理 2.6 帮助我们得到

$$\begin{aligned}\left(\int_{\mathcal{B}_r(\boldsymbol{y})} \|\nabla u(\boldsymbol{x})\|_2^p \, d\boldsymbol{x}\right)^{\frac{1}{p}} &= \left\| \left(\sum_{i=1}^n \left|\partial_i u(\boldsymbol{x})\right|^2\right)^{\frac{1}{2}} \right\|_{L^p(\mathcal{B}_r(\boldsymbol{y}))} \\ &\leq \left\| \left(\sum_{i=1}^n \left|\partial_i u(\boldsymbol{x})\right|^2\right)^{\frac{1}{2}} \right\|_{L^p(\Omega)} \\ &\leq \left\| \left(\sum_{i=1}^n \left|\partial_i u(\boldsymbol{x})\right|^2\right)^{\frac{1}{2}} \right\|_{L^2(\Omega)}^{\theta} \left\| \left(\sum_{i=1}^n \left|\partial_i u(\boldsymbol{x})\right|^2\right)^{\frac{1}{2}} \right\|_{L^\infty(\Omega)}^{1-\theta} \\ &\leq n^{\frac{1-\theta}{2}} |u|_{H^1(\Omega)}^{\theta} M_1^{1-\theta},\end{aligned}$$

其中 $\theta = \frac{2}{p} \leq 1$, 且 $p > n$. 进而我们有

$$\begin{aligned}\int_{\mathcal{B}_r(\boldsymbol{y})} |u(\boldsymbol{y}) - u(\boldsymbol{x})|\, d\boldsymbol{x} &\leq \frac{r^n}{n} n^{\frac{1-\theta}{2}} |u|_{H^1(\Omega)}^{\theta} M_1^{1-\theta} \mathcal{V}_n^{\frac{1}{q}} n^{\frac{1}{q}} (n+q-nq)^{-\frac{1}{q}} r^{\frac{n}{q}+1-n} \\ &= \mathcal{V}_n^{\frac{1}{q}} r^{\frac{n}{q}+1} n^{\frac{1}{q}-\frac{1+\theta}{2}} (n+q-nq)^{-\frac{1}{q}} M_1^{1-\theta} |u|_{H^1(\Omega)}^{\theta}.\end{aligned}$$

因此我们有

$$|u(\boldsymbol{y})| \leq \frac{1}{\mathcal{V}_n r^n}\left[r^{\frac{n}{q}+1} n^{\frac{1}{q}-\frac{1+\theta}{2}} (n+q-nq)^{-\frac{1}{q}} |u|^{\theta}_{H^1(\Omega)} M_1^{1-\theta} \mathcal{V}_n^{\frac{1}{q}} + \mathcal{V}_n^{\frac{1}{2}} r^{\frac{n}{2}} \|u\|_{L^2(\Omega)}\right]$$
$$= \mathcal{V}_n^{-\frac{1}{2}} r^{-\frac{n}{2}} \|u\|_{L^2(\Omega)} + \mathcal{V}_n^{\frac{1}{q}-1} r^{\frac{n}{q}+1-n} n^{\frac{1}{q}-\frac{1+\theta}{2}} (n+q-nq)^{-\frac{1}{q}} M_1^{1-\theta} |u|^{\theta}_{H^1(\Omega)}.$$

故 (2-9) 成立. □

以下定理说明了给定区域内函数的 H^2 范数与函数值的绝对值及其梯度在某点的范数的关系.

定理 2.9. 设$u \in \mathcal{H}^2(\Omega)$, 其中 $\Omega = \mathcal{B}_r(\boldsymbol{x}_0)$. 假设存在 M_1, M_2 使得 $\left|\partial_i u\right| \leq M_1$, $\left|\partial^2_{ij} u\right| \leq M_2$, $i, j = 1, \cdots, n$, 则对于 $\forall\, \boldsymbol{x} \in \Omega$, 我们有

$$|u(\boldsymbol{x})| \leq \mathcal{V}_n^{-\frac{1}{2}} r^{-\frac{n}{2}} \|u\|_{L^2(\Omega)} + \mathcal{V}_n^{-\frac{1}{p}} r^{\frac{n}{q}+1-n} n^{\frac{1}{q}-\frac{1+\theta}{2}} (n+q-nq)^{-\frac{1}{q}} M_1^{1-\theta} |u|^{\theta}_{H^1(\Omega)}, \quad (2\text{-}11)$$

$$\|\nabla u(\boldsymbol{x})\|_2 \leq n^{\frac{1}{2}} \mathcal{V}_n^{-\frac{1}{2}} r^{-\frac{n}{2}} |u|_{H^1(\Omega)} + \mathcal{V}_n^{-\frac{1}{p}} r^{\frac{n}{q}+1-n} n^{\frac{1}{q}-\frac{\theta}{2}} (n+q-nq)^{-\frac{1}{q}} M_2^{1-\theta} |u|^{\theta}_{H^2(\Omega)}, \quad (2\text{-}12)$$

其中 $\frac{1}{p}+\frac{1}{q}=1$, $\theta = \frac{2}{p} \leq 1$, $p > n$, $\mathcal{B}_r(\boldsymbol{x}_0) = \left\{\boldsymbol{x} \in \mathfrak{R}^n : \left\|\boldsymbol{x} - \boldsymbol{x}_0\right\|_2 \leq r\right\}$, $\mathcal{V}_n$ 表示 n 维 ℓ_2 范数单位球 $\mathcal{B}_1(\boldsymbol{x}_0)$ 的体积.

证明. 我们可以由引理 2.8 直接得到 (2-11). 对于 $\forall\, \boldsymbol{x} \in \Omega := \mathcal{B}_r(\boldsymbol{x}_0)$, 我们有

$$|\partial_i u(\boldsymbol{x})| \leq \mathcal{V}_n^{-\frac{1}{2}} r^{-\frac{n}{2}} |u|_{H^1(\Omega)} + \mathcal{V}_n^{-\frac{1}{p}} r^{\frac{n}{q}+1-n} n^{\frac{1}{q}-\frac{1+\theta}{2}} (n+q-nq)^{-\frac{1}{q}} M_2^{1-\theta} |u|^{\theta}_{H^2(\Omega)},$$

其中 $\frac{1}{p}+\frac{1}{q}=1$, $\theta = \frac{2}{p} \leq 1$, 且 $p > n$. 因此, 我们可以得到

$$\|\nabla u\|_2 \leq n^{\frac{1}{2}} \max_{i=1,\cdots,n} \left\|\partial_i u\right\|_{L^\infty(\Omega)}$$
$$\leq n^{\frac{1}{2}}\left[\mathcal{V}_n^{-\frac{1}{2}} r^{-\frac{n}{2}} |u|_{H^1(\Omega)} + \mathcal{V}_n^{-\frac{1}{p}} r^{\frac{n}{q}+1-n} n^{\frac{1}{q}-\frac{1+\theta}{2}} (n+q-nq)^{-\frac{1}{q}} M_2^{1-\theta} |u|^{\theta}_{H^2(\Omega)}\right],$$

从而得到 (2-12). □

根据定理 2.9, 可以自然地得到下面的推论. 我们可以观察到, 减小函数 u 的 H^2 范数也会让 u 的函数值和梯度向量的范数减小.

推论 2.10. 给定目标函数 $f \in \mathcal{H}^2$(Ω 及其二次模型函数 Q, 假设 $\left|\partial_i (Q-f)\right| \leq M_1$, $\left|\partial^2_{ij}(Q-f)\right| \leq M_2$, $i, j = 1, \cdots, n$. 则对于 $\forall\, \boldsymbol{x} \in \Omega$, 我们有

$$|Q(\boldsymbol{x}) - f(\boldsymbol{x})| \leq \mathcal{V}_n^{-\frac{1}{2}} r^{-\frac{n}{2}} \|Q - f\|_{L^2(\Omega)}$$

$$+\mathcal{V}_n^{-\frac{1}{p}} r^{\frac{n}{q}+1-n} n^{\frac{1}{q}-\frac{1+\theta}{2}} (n+q-nq)^{-\frac{1}{q}} M_1^{1-\theta} |Q-f|^{\theta}_{H^1(\Omega)},$$

$$\|\nabla Q(\boldsymbol{x}) - \nabla f(\boldsymbol{x})\|_2 \leq n^{\frac{1}{2}} \mathcal{V}_n^{-\frac{1}{2}} r^{-\frac{n}{2}} |Q-f|_{H^1(\Omega)}$$
$$+\mathcal{V}_n^{-\frac{1}{p}} r^{\frac{n}{q}+1-n} n^{\frac{1}{q}-\frac{\theta}{2}} (n+q-nq)^{-\frac{1}{q}} M_2^{1-\theta} |Q-f|^{\theta}_{H^2(\Omega)},$$

其中 $\frac{1}{p}+\frac{1}{q}=1$, $\theta=\frac{2}{p}\leq 1$, $p>n$, 且 $\mathcal{V}_n$ 表示 n 维 ℓ_2 范数单位球 $\mathcal{B}_1(\boldsymbol{x}_0)$ 的体积.

证明. 在定理 2.9 中将 u 替换为 $Q-f$ 即可证明该推论. □

推论 2.10 中的误差分析从理论上分析了最小 H^2 范数更新二次模型的逼近效果. 因此, 我们认为通过求解子问题 (2-5) 或子问题 (2-6) 获取模型函数可以放松插值条件的最小要求, 也即允许我们使用更少的插值点来保持良好的逼近性质.

2.1.3　最小 H^2 范数更新二次模型

在本节中, 我们根据 KKT 条件给出获得最小 H^2 范数更新二次模型的计算方法. 定理 2.2 及其证明帮助我们通过在第 k 步求解问题

$$\begin{aligned}&\min_{c,\boldsymbol{g},\boldsymbol{H}} \eta_1\|\boldsymbol{H}\|_F^2+\eta_2\|\boldsymbol{g}\|_2^2+\eta_3(\mathrm{Tr}(\boldsymbol{H}))^2+\eta_4\,\mathrm{Tr}(\boldsymbol{H})c+\eta_5 c^2\\ &\text{s. t. } c+\boldsymbol{g}^{\top}\left(\boldsymbol{y}_i-\boldsymbol{x}_0\right)+\frac{1}{2}\left(\boldsymbol{y}_i-\boldsymbol{x}_0\right)^{\top}\boldsymbol{H}\left(\boldsymbol{y}_i-\boldsymbol{x}_0\right)=f(\boldsymbol{y}_i)-Q_{k-1}(\boldsymbol{y}_i), i=1,\cdots,m\end{aligned} \tag{2-13}$$

来获得二次模型函数的系数, 其中 $\boldsymbol{H}^{\top}=\boldsymbol{H}$, 并且 (2-13) 的解是模型函数之差 Q_k-Q_{k-1} 的系数. 实验中所用的计算 H^2 范数的半径 r 的选择将在第 2.1.4 节中给出. 简明起见, 我们用点 $\boldsymbol{y}_1,\cdots,\boldsymbol{y}_m$ 表示第 k 次迭代的插值点. 我们直接考虑带权重系数 C_1、C_2 和 C_3 的加权目标函数. 注意, 系数 $\eta_1,\eta_2,\eta_3,\eta_4,\eta_5$ 满足

$$\begin{cases}\eta_1=C_1\dfrac{r^4}{2(n+4)(n+2)}+C_2\dfrac{r^2}{n+2}+C_3,\\ \eta_2=C_1\dfrac{r^2}{n+2}+C_2,\\ \eta_3=C_1\dfrac{r^4}{4(n+4)(n+2)},\\ \eta_4=C_1\dfrac{r^2}{n+2},\\ \eta_5=C_1.\end{cases} \tag{2-14}$$

我们知道, 问题 (2-13) 对应的 Lagrange 函数是

$$\begin{aligned}\mathcal{L}(c,\boldsymbol{g},\boldsymbol{H})&=\eta_1\|\boldsymbol{H}\|_F^2+\eta_2\|\boldsymbol{g}\|_2^2+\eta_3(\mathrm{Tr}(\boldsymbol{H}))^2+\eta_4\,\mathrm{Tr}(\boldsymbol{H})c+\eta_5 c^2\\ &-\sum_{i=1}^{m}\lambda_i\left[c+\boldsymbol{g}^{\top}\left(\boldsymbol{y}_i-\boldsymbol{x}_0\right)+\frac{1}{2}\left(\boldsymbol{y}_i-\boldsymbol{x}_0\right)^{\top}\boldsymbol{H}\left(\boldsymbol{y}_i-\boldsymbol{x}_0\right)-f(\boldsymbol{y}_i)+Q_{k-1}(\boldsymbol{y}_i)\right].\end{aligned} \tag{2-15}$$

我们用 T 表示 $\mathrm{Tr}(\boldsymbol{H})$. 问题 (2-13) 的 KKT 条件包括

$$0=\frac{\partial \mathcal{L}(c,\boldsymbol{g},\boldsymbol{H})}{\partial c}=2\eta_5 c+\eta_4 T-\sum_{i=1}^{m}\lambda_i,$$

$$\mathbf{0}_n=\frac{\partial \mathcal{L}(c,\boldsymbol{g},\boldsymbol{H})}{\partial \boldsymbol{g}}=2\eta_2\boldsymbol{g}-\sum_{i=1}^{m}\lambda_i\left(\boldsymbol{y}_i-\boldsymbol{x}_0\right),$$

其中 $\mathbf{0}_n=(0,\cdots,0)^{\top}\in\mathfrak{R}^n$, KKT 条件中的其他方程在下面给出. 令 $\mathcal{L}(c,\boldsymbol{g},\boldsymbol{H})$ 对 $\boldsymbol{H}$ 的元素求导, 我们可以得到

$$2\eta_1\boldsymbol{H}-\frac{1}{2}\sum_{l=1}^{m}\lambda_l\left(\boldsymbol{y}_l-\boldsymbol{x}_0\right)\left(\boldsymbol{y}_l-\boldsymbol{x}_0\right)^{\top}+2\eta_3\,\mathrm{Diag}\left\{T,\cdots,T\right\}+\eta_4 c\boldsymbol{I}=\mathbf{0}_{nn}.$$

因此

$$2\eta_1\boldsymbol{H}=\frac{1}{2}\sum_{i=1}^{m}\lambda_i\left(\boldsymbol{y}_i-\boldsymbol{x}_0\right)\left(\boldsymbol{y}_i-\boldsymbol{x}_0\right)^{\top}-\left(2\eta_3 T+\eta_4 c\right)\boldsymbol{I}. \tag{2-16}$$

通过在 (2-16) 左侧和右侧分别乘以 $\left(\boldsymbol{y}_j-\boldsymbol{x}_0\right)^{\top}$ 和 $\left(\boldsymbol{y}_j-\boldsymbol{x}_0\right)$, 我们得到

$$\begin{aligned}&2\eta_1\left(\boldsymbol{y}_j-\boldsymbol{x}_0\right)^{\top}\boldsymbol{H}\left(\boldsymbol{y}_j-\boldsymbol{x}_0\right)\\&=\frac{1}{2}\sum_{i=1}^{m}\lambda_i\left[\left(\boldsymbol{y}_i-\boldsymbol{x}_0\right)^{\top}\left(\boldsymbol{y}_j-\boldsymbol{x}_0\right)\right]^2-\left(2\eta_3 T+\eta_4 c\right)\left\|\boldsymbol{y}_j-\boldsymbol{x}_0\right\|_2^2,\ 1\le j\le m.\end{aligned}$$

此外, 我们在 $\boldsymbol{H}$ 左右分别乘以 $\boldsymbol{e}_j^{\top}$ 和 $\boldsymbol{e}_j$. 这里, 向量$\boldsymbol{e}_j$ 被定义为单位矩阵 $\boldsymbol{I}$ 的第 j 列. 然后我们得到

$$2\eta_1\boldsymbol{e}_j^{\top}\boldsymbol{H}\boldsymbol{e}_j=\frac{1}{2}\sum_{i=1}^{m}\lambda_i\left[\boldsymbol{e}_j^{\top}\left(\boldsymbol{y}_i-\boldsymbol{x}_0\right)\right]^2-\left(2\eta_3 T+\eta_4 c\right)\boldsymbol{e}_j^{\top}\boldsymbol{e}_j,\ 1\le j\le n. \tag{2-17}$$

通过把 j 从 1 加到 n 计算 (2-17) 的求和, 我们可以得到

$$2\eta_1 T=\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n}\lambda_l\left[\boldsymbol{e}_j^{\top}\left(\boldsymbol{y}_i-\boldsymbol{x}_0\right)\right]^2-n\left(2\eta_3 T+\eta_4 c\right),$$

进而

$$0=\frac{1}{2}\sum_{i=1}^{m}\lambda_i\left\|\boldsymbol{y}_i-\boldsymbol{x}_0\right\|_2^2-(2n\eta_3+2\eta_1)T-n\eta_4 c.$$

此时我们得到 T 的表达式, 即

$$T=\frac{1}{2\left(2n\eta_3+2\eta_1\right)}\sum_{i=1}^{m}\lambda_i\left\|\boldsymbol{y}_i-\boldsymbol{x}_0\right\|_2^2-\frac{n\eta_4}{2n\eta_3+2\eta_1}c. \tag{2-18}$$

结合 (2-13) 中的约束, 我们得到关于$\boldsymbol{\lambda}=(\lambda_1,\cdots,\lambda_m)^{\top}, c, \boldsymbol{g}$ 的方程:

$$
\begin{aligned}
0 &= 2\eta_5 c + \frac{\eta_4}{4n\eta_3 + 4\eta_1} \sum_{i=1}^{m} \lambda_i \left\| \boldsymbol{y}_i - \boldsymbol{x}_0 \right\|_2^2 - \frac{n\eta_4^2}{2n\eta_3 + 2\eta_1} c - \sum_{i=1}^{m} \lambda_i, \\
\boldsymbol{0}_n &= 2\eta_2 \boldsymbol{g} - \sum_{i=1}^{m} \lambda_i \left(\boldsymbol{y}_i - \boldsymbol{x}_0 \right), \\
f(\boldsymbol{y}_j) - Q_{k-1}(\boldsymbol{y}_j) &= \frac{1}{8\eta_1} \sum_{i=1}^{m} \lambda_i \left[\left(\boldsymbol{y}_i - \boldsymbol{x}_0 \right)^{\top} \left(\boldsymbol{y}_j - \boldsymbol{x}_0 \right) \right]^2 \\
&\quad - \frac{\eta_3}{8\eta_1 \left(n\eta_3 + \eta_1 \right)} \sum_{i=1}^{m} \lambda_i \left\| \boldsymbol{y}_i - \boldsymbol{x}_0 \right\|_2^2 \left\| \boldsymbol{y}_j - \boldsymbol{x}_0 \right\|_2^2 \\
&\quad - \frac{\eta_4}{4n\eta_3 + 4\eta_1} c \left\| \boldsymbol{y}_j - \boldsymbol{x}_0 \right\|_2^2 + c + \left(\boldsymbol{y}_j - \boldsymbol{x}_0 \right)^{\top} \boldsymbol{g}, \ j = 1, \cdots, m.
\end{aligned}
$$

由于在第 k 次迭代中, $\boldsymbol{y}_t$ 被 $\boldsymbol{y}_{\text{new}}$ 替换, 且 $Q_k(\boldsymbol{y}_i) - Q_{k-1}(\boldsymbol{y}_i) = f(\boldsymbol{y}_i) - Q_{k-1}(\boldsymbol{y}_i)$, 给定当前插值集中的所有 $\boldsymbol{y}_i$, 我们可以得到方程组

$$
\overbrace{\begin{pmatrix} \boldsymbol{A} & \boldsymbol{J} & \boldsymbol{X} \\ \boldsymbol{J}^{\top} & \frac{n\eta_4^2}{2n\eta_3 + 2\eta_1} - 2\eta_5 & \boldsymbol{0}_n^{\top} \\ \boldsymbol{X}^{\top} & \boldsymbol{0}_n & -2\eta_2 \boldsymbol{I} \end{pmatrix}}^{m+1+n} \begin{pmatrix} \boldsymbol{\lambda} \\ c \\ \boldsymbol{g} \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ f(\boldsymbol{y}_{\text{new}}) - Q_{k-1}(\boldsymbol{y}_{\text{new}}) \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \tag{2-19}
$$

其中 $\boldsymbol{y}_{\text{new}}$ 表示新的插值点, 且 $\boldsymbol{\lambda} = (\lambda_1, \cdots, \lambda_m)^{\top}$. 此外, $\boldsymbol{A}$ 的元素为

$$
\boldsymbol{A}_{ij} = \frac{1}{8\eta_1} \left[\left(\boldsymbol{y}_i - \boldsymbol{x}_0 \right)^{\top} \left(\boldsymbol{y}_j - \boldsymbol{x}_0 \right) \right]^2 - \frac{\eta_3}{8\eta_1 \left(n\eta_3 + \eta_1 \right)} \left\| \boldsymbol{y}_i - \boldsymbol{x}_0 \right\|_2^2 \left\| \boldsymbol{y}_j - \boldsymbol{x}_0 \right\|_2^2,
$$

其中 $1 \le i, j \le m$. 此外, 有 $\boldsymbol{X} = \left(\boldsymbol{y}_1 - \boldsymbol{x}_0, \boldsymbol{y}_2 - \boldsymbol{x}_0, \cdots, \boldsymbol{y}_m - \boldsymbol{x}_0 \right)^{\top}$, 以及

$$
\boldsymbol{J} = \left(1 - \frac{\eta_4}{4n\eta_3 + 4\eta_1} \left\| \boldsymbol{y}_1 - \boldsymbol{x}_0 \right\|_2^2, \cdots, 1 - \frac{\eta_4}{4n\eta_3 + 4\eta_1} \left\| \boldsymbol{y}_m - \boldsymbol{x}_0 \right\|_2^2 \right)^{\top}.
$$

我们称 (2-19) 左侧的矩阵为 KKT 矩阵 $\boldsymbol{W}$.

基于 (2-19) 给出的 $\boldsymbol{\lambda}, c, \boldsymbol{g}$, 我们可以得到二次模型函数 $Q(\boldsymbol{x})$. 事实上, 最小 Frobenius 范数更新二次模型是最小 H^2 范数更新二次模型的一个特例, 如下所示.

注 2.3. 若 $C_1 = C_2 = 0, C_3 = 1$, 则 $\eta_1 = 1, \eta_2 = \eta_3 = \eta_4 = \eta_5 = 0$, 此时 KKT 矩阵为

$$
\boldsymbol{W} = \begin{pmatrix} \bar{\boldsymbol{A}} & \boldsymbol{E} & \boldsymbol{X} \\ \boldsymbol{E}^{\top} & 0 & \boldsymbol{0}_n^{\top} \\ \boldsymbol{X}^{\top} & \boldsymbol{0}_n & \boldsymbol{0}_{nn} \end{pmatrix}, \tag{2-20}
$$

其中 $\boldsymbol{E} \in \Re^m$ 是 $(1,\cdots,1)^\top$, 且 $\bar{\boldsymbol{A}}_{ij} = \frac{1}{8}\left[\left(\boldsymbol{y}_i - \boldsymbol{x}_0\right)^\top \left(\boldsymbol{y}_j - \boldsymbol{x}_0\right)\right]^2$, $1 \leq i, j \leq m$.

在这种情况下, 对应于插值点 $\boldsymbol{y}_1, \cdots, \boldsymbol{y}_m$ 的 Hessian 矩阵为

$$\boldsymbol{H} = \frac{1}{4}\sum_{i=1}^{m} \lambda_i \left(\boldsymbol{y}_i - \boldsymbol{x}_0\right)\left(\boldsymbol{y}_i - \boldsymbol{x}_0\right)^\top . \tag{2-21}$$

(2-20) 中的 $(m+n+1)\times(m+n+1)$ 维矩阵正是最小 Frobenius 范数更新二次模型对应的 KKT 矩阵 [94]. 注意, (2-21) 中的系数 $\frac{1}{4}$ 取决于 Lagrange 函数 (2-15) 中的系数, 这并不影响结果.

为了降低计算复杂度, 我们将在下文中讨论和使用 KKT 逆矩阵的更新公式. 在讨论 KKT 矩阵的逆矩阵之前, 我们首先介绍以下定理, 该定理给出了 KKT 矩阵可逆的条件.

定理 2.11. $(m+n+1)\times(m+n+1)$ 维矩阵 $\boldsymbol{W}$ 是可逆矩阵当且仅当 $(m+1)\times(m+1)$ 维矩阵

$$\begin{pmatrix} \boldsymbol{A} + \frac{1}{2\eta_2}\boldsymbol{X}\boldsymbol{X}^\top & \boldsymbol{J} \\ \boldsymbol{J}^\top & \frac{n\eta_4^2}{2n\eta_3+2\eta_1} - 2\eta_5 \end{pmatrix}$$

是可逆的.

证明. 我们有

$$\begin{pmatrix} \boldsymbol{A} & \boldsymbol{J} & \boldsymbol{X} \\ \boldsymbol{J}^\top & \frac{n\eta_4^2}{2n\eta_3+2\eta_1} - 2\eta_5 & \boldsymbol{0}_n^\top \\ \boldsymbol{X}^\top & \boldsymbol{0}_n & -2\eta_2 \boldsymbol{I} \end{pmatrix} \to \begin{pmatrix} \boldsymbol{A} + \frac{1}{2\eta_2}\boldsymbol{X}\boldsymbol{X}^\top & \boldsymbol{J} & \boldsymbol{X} \\ \boldsymbol{J}^\top & \frac{n\eta_4^2}{2n\eta_3+2\eta_1} - 2\eta_5 & \boldsymbol{0}_n^\top \\ \boldsymbol{0}_{nm} & \boldsymbol{0}_n & -2\eta_2 \boldsymbol{I} \end{pmatrix},$$

其中箭头表示初等变换. 进而结论得证. □

定理 2.11 给出了 KKT 矩阵可逆的充分必要条件. 在下文我们把 KKT 矩阵的逆矩阵简称为 KKT 逆矩阵.

注意, 通过在每次迭代中直接求解 KKT 方程 (2-19) 来获取二次模型函数的参数 $\boldsymbol{\lambda}, c, \boldsymbol{g}$ 在数值计算上其实是不高效的, 其计算复杂度为 $\mathcal{O}((m+n)^3)$. 我们尝试使用计算复杂度低的 KKT 逆矩阵更新公式. 与 Powell [94, 174] 给出的讨论类似, 一个自然的问题是在迭代增加时 KKT 矩阵会发生什么变化. 事实上, 当插值集更新时, 我们发现 $\boldsymbol{W}$ 只有第 t 列和第 t 行发生变化, 这是因为这里只有 $\boldsymbol{y}_t$ 被 $\boldsymbol{y}_{\text{new}}$ 替换. 我们借鉴了 Powell [174] 给出的 KKT 逆矩阵的更新公式.

我们定义向量 $\boldsymbol{\omega} \in \Re^{m+n+1}$, 其分量 ω_i 分别等于

$$\begin{cases} \frac{1}{8\eta_1}\left[\left(\boldsymbol{y}_i - \boldsymbol{x}_0\right)^\top\left(\boldsymbol{y}_{\text{new}} - \boldsymbol{x}_0\right)\right]^2 - \frac{\eta_3}{8\eta_1\left(n\eta_3+\eta_1\right)}\left\|\boldsymbol{y}_i - \boldsymbol{x}_0\right\|_2^2\left\|\boldsymbol{y}_{\text{new}} - \boldsymbol{x}_0\right\|_2^2, & \text{如果 } 1 \leq i \leq m, \\ 1 - \frac{\eta_4}{4n\eta_3+4\eta_1}\left\|\boldsymbol{y}_{\text{new}} - \boldsymbol{x}_0\right\|_2^2, \text{如果 } i = m+1, \\ \left(\boldsymbol{y}_{\text{new}} - \boldsymbol{x}_0\right)_{i-m-1}, \text{如果 } m+2 \leq i \leq m+n+1. \end{cases}$$

如果一个可逆的 KKT 矩阵 $\boldsymbol{W}$ 的第 t 列和第 t 行分别被向量 $\boldsymbol{\omega}$ 和 $\boldsymbol{\omega}^{\top}$ 替换, 新矩阵记为 $\boldsymbol{W}_{\text{new}}$, $\boldsymbol{V}_{\text{new}} := \boldsymbol{W}_{\text{new}}^{-1}$, $\boldsymbol{V} := \boldsymbol{W}^{-1}$, 那么新的 KKT 逆矩阵

$$\begin{aligned}\boldsymbol{V}_{\text{new}} = \boldsymbol{V} + \sigma^{-1}\Big\{&\alpha\left(\boldsymbol{e}_t - \boldsymbol{V}\boldsymbol{\omega}\right)\left(\boldsymbol{e}_t - \boldsymbol{V}\boldsymbol{\omega}\right)^{\top} - \beta\boldsymbol{V}\boldsymbol{e}_t\boldsymbol{e}_t^{\top}\boldsymbol{V} \\ &+\tau\left[\boldsymbol{V}\boldsymbol{e}_t\left(\boldsymbol{e}_t - \boldsymbol{V}\boldsymbol{\omega}\right)^{\top} + \left(\boldsymbol{e}_t - \boldsymbol{V}\boldsymbol{\omega}\right)\boldsymbol{e}_t^{\top}\boldsymbol{V}\right]\Big\},\end{aligned} \tag{2-22}$$

其中

$$\begin{cases}\alpha = \boldsymbol{e}_t^{\top}\boldsymbol{V}\boldsymbol{e}_t, \\ \beta = \dfrac{1}{8\eta_1}\left\|\boldsymbol{y}_{\text{new}} - \boldsymbol{x}_0\right\|_2^4 - \boldsymbol{\omega}^{\top}\boldsymbol{V}\boldsymbol{\omega}, \\ \tau = \boldsymbol{e}_t^{\top}\boldsymbol{V}\boldsymbol{\omega}, \\ \sigma = \alpha\beta + \tau^2.\end{cases} \tag{2-23}$$

我们根据更新公式 (2-22) 获得新的 KKT 逆矩阵 $\boldsymbol{V}_{\text{new}}$, 然后通过

$$\begin{pmatrix}\boldsymbol{\lambda} \\ c \\ \boldsymbol{g}\end{pmatrix} = \boldsymbol{V}_{\text{new}}\begin{pmatrix}0 \\ \vdots \\ 0 \\ f(\boldsymbol{y}_{\text{new}}) - Q_{k-1}(\boldsymbol{y}_{\text{new}}) \\ 0 \\ \vdots \\ 0\end{pmatrix}$$

得到 $(\boldsymbol{\lambda}, c, \boldsymbol{g})^{\top}$. 在这种情况下, 相应的 $\boldsymbol{V}$ 的更新可以在 $\mathcal{O}((m+n)^2)$ 的操作内完成计算.

在给出更新公式后, 我们进一步考虑更多其他细节, 我们试图通过在使用基于最小 $\boldsymbol{H}^2$ 范数更新二次模型函数 (通过 (2-22) 更新 KKT 逆矩阵) 的算法中利用其他方式选择新的迭代点来提高更新公式的稳健性.

注意到, KKT 矩阵的逆矩阵更新公式 (2-22) 有一个分母 $\sigma = \alpha\beta + \tau^2$, 其中 α, β 和 τ 的表达式在 (2-23) 中. 为了避免因为分母 σ 的绝对值太小而导致数值更新不稳定, 我们使用模型改进步作为算法 3 的第 4 步, 即获取迭代点 $\boldsymbol{y}_{\text{new}} = \boldsymbol{x}_{\text{opt}} + \boldsymbol{d}$, 其中 $\boldsymbol{d} \in \mathfrak{R}^n$ 通过近似求解问题

$$\begin{aligned}&\max_{\boldsymbol{d}} \ \left|\alpha\beta + \tau^2\right| \\ &\text{s. t. } \ \|\boldsymbol{d}\|_2 \le \Delta_k\end{aligned} \tag{2-24}$$

得到. (2-24) 中的目标函数是 $\boldsymbol{d}$ 的函数, 这是因为 $\boldsymbol{y}_{\text{new}} = \boldsymbol{x}_{\text{opt}} + \boldsymbol{d}$. 事实上, 子问题 (2-24) 是关于 $\boldsymbol{d}$ 的四次问题. 当更新公式 (2-22) 可能因插值集不够适定而不稳健时, 在基于最小 $\boldsymbol{H}^2$ 范数更新二次模型的算法实现中, 模型改进步被选为在信赖域中的上述四次问题的解. 注意, 当前的实现版本参考了 NEWUOA 的 BIGDEN 子程序的主要思想 (在 Powell [94] 的第 6 节中) 以在 (2-24) 中获取目标函数的相

对较大值 (我们不需要非常精确地求解它). 当前实现试图极大化 (2-24) 中目标函数的二次近似 (二阶展开), 然后迭代地获得探测点, 但只要它在探测点处找到目标函数 (2-24) 的值是 $\boldsymbol{d} = \boldsymbol{0}$ 对应值的 1.1 倍的情况时就停止子程序. 值得注意的是, 由于子问题本身不是一个无导数问题, 因此可以尝试其他可能的实用方法来求解此类子问题, 故这里没有给出更多其他细节.

在当前的实现版本中, 若 $\rho_k < \hat{\eta}_1$ 且 (距离 $\boldsymbol{x}_{\text{opt}}$ 的) 最远插值点 $\boldsymbol{y}_{\text{far}}$ 与 $\boldsymbol{x}_{\text{opt}}$ 的距离满足 $\|\boldsymbol{y}_{\text{far}} - \boldsymbol{x}_{\text{opt}}\|_2 > 2\Delta_k$, 算法将不接受该模型, 继而将调用模型改进步, 这与 Powell 的算法 NEWUOA 的做法类似 (与插值的适定性相关). 此外, 当 $\|\boldsymbol{x}_k - \boldsymbol{x}_0\|_2 > 10\Delta_k$ 时, 它将基点 $\boldsymbol{x}_0$ 更改为当前的 $\boldsymbol{x}_{\text{opt}}$, 即下一个信赖域的中心. 关于这一点的更多细节, Powell [94] 给出了详细讨论. 更新公式 (2-22) 降低了迭代过程中的整体计算复杂性. 关于 $\boldsymbol{x}_0$ 选择的更多细节在张在坤的工作 [154] 中有所介绍. 此外, 关于插值集几何性质和适定性的更多细节在 Conn、Scheinberg 和 Vicente 的工作 [175] 中有详细讨论.

2.1.4 数值结果

为了展示我们的最小 H^2 范数更新二次模型的优势, 我们给出了求解无约束无导数优化问题 (1-1) 的数值结果. 数值实验包含三个部分. 它们分别是插值误差和更新的观察与比较、简单的仿真, 以及通过求解测试问题集获取的 Performance Profile 和 Data Profile 进行比较. 我们根据算法 3 提供的框架使用 Python 实现了一个无导数信赖域算法进行数值测试. 测试中使用的最小 H^2 范数二次模型是通过更新公式 (2-19) 获得的, 同时我们使用了公式 (2-22) 来更新 KKT 逆矩阵. 算法中的模型改进步是通过近似求解子问题 (2-24) 获得的. 为了在本节中直接和公平地比较不同的模型函数, 我们保持算法框架相同, 将相应的公式逐个用其他模型的公式替换. 在这里的数值实验中, 权重系数 C_1, C_2, C_3 均被设置为 $\frac{1}{3}$. 此外, 在我们的数值实现中, 第 k 步的半径 r 被设定为 $\max\{10\Delta_k, \max\limits_{\boldsymbol{y}\in\mathcal{X}_k} \|\boldsymbol{y} - \boldsymbol{x}_{\text{opt}}\|_2\}$ (这与张在坤 [154] 的设置相同[3]). 数值结果说明了我们通过最小 H^2 范数更新而不是最小 Frobenius 范数更新来获得二次模型函数的选择是有优势的.

我们首先对基于最小化两个模型之间的 H^2 范数来更新二次模型时的插值误差和稳定性进行了数值观察. 为了展示使用 H^2 范数获取模型函数的优势, 我们使用以下例子来在数值上观察最小 H^2 范数更新插值与最小 Frobenius 范数更新插值之间的区别.

例 2.1. 我们知道更新公式对应的子问题 (2-5) 可以转化为通过求解

$$\begin{aligned} &\min_{D\in\mathcal{Q}} \ \|D\|^2_{H^2(\mathcal{B}_r(\boldsymbol{x}_0))} \\ &\text{s. t. } D(\boldsymbol{y}_{\text{new}}) = f(\boldsymbol{y}_{\text{new}}) - Q_{k-1}(\boldsymbol{y}_{\text{new}}),\ \boldsymbol{y}_{\text{new}} \in \mathcal{X}_k, \\ &\qquad D(\boldsymbol{y}_i) = 0,\ \boldsymbol{y}_i \in \mathcal{X}_k\backslash\{\boldsymbol{y}_{\text{new}}\} \end{aligned} \tag{2-25}$$

[3]有其他给 r 赋值的方式. 这里的赋值方式简单且对数值实验已经足够.

来获取 $\boldsymbol{D}_k$, 其中 $\boldsymbol{D}_k = \boldsymbol{Q}_k - \boldsymbol{Q}_{k-1}$. 因此, 在这个简单的 2 维例子中, 我们假定, 在第 k 次迭代, 问题 (2-25) 中的函数 $f - \boldsymbol{Q}_{k-1}$ 满足

$$f(\boldsymbol{x}) - \boldsymbol{Q}_{k-1}(\boldsymbol{x}) = \begin{cases} 1, \text{ 如果 } \boldsymbol{x} = \boldsymbol{y}_{\text{new}}, \\ 0, \text{ 否则.} \end{cases} \tag{2-26}$$

注 2.4. 这个例子与 Lagrange 基函数密切相关. 注意, 在相应的第 k 次迭代中, 旧点 $\boldsymbol{y}_t$ 被 $\boldsymbol{y}_{\text{new}}$ 替换. 函数 $\boldsymbol{Q}_k = \boldsymbol{Q}_{k-1} + \boldsymbol{D}_k$ 正是第 k 个模型函数, 初始模型是 $\boldsymbol{Q}_0(\boldsymbol{x}) = 0$. 在进入迭代之前, $f((0,0)^\top) = 1$, 且对于 $\forall\, \boldsymbol{x} \neq (0,0)^\top$ 有 $f(\boldsymbol{x}) = 0$, 然后 f 在后续步骤中满足 (2-26). 我们使用这个例子来观察模型的基本表现. Powell [176] 讨论了使用 Lagrange 基来获取模型的优势.

我们在每一步使用 3 个插值点, 这个简单例子的初始插值点是

$$\boldsymbol{y}_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \ \boldsymbol{y}_2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \ \boldsymbol{y}_3 = \begin{pmatrix} 0 \\ 1 \end{pmatrix},$$

这是一个常见的选择. 迭代次数总数为 3, 这里的信赖域设置为 $\mathcal{B}_1(\boldsymbol{x}_{\text{opt}})$, 其中 $\boldsymbol{x}_{\text{opt}}$ 是当前迭代插值点中函数值最小的点. 在这里, 我们关注和比较最小 Frobenius 范数更新二次模型和最小 $\boldsymbol{H}^2$ 范数更新二次模型在初始阶段的插值表现对比. 我们固定信赖域半径, 这样可以将前 3 次迭代中的插值误差视为一个足够公平直观的比较, 注意, 在对比时, 我们也考虑简单地选择借助区域网格点来计算插值误差作为参考.

图 2-1 展示了数值结果. 在图 2-1 的每个子图中, 我们分别绘制了两条线来表示 Powell 的模型和我们的模型, 它们分别基于最小 Frobenius 范数更新和最小 $\boldsymbol{H}^2$ 范数更新. 图 2-1 中的子图 2-1a 和子图 2-1b 展示了所有迭代点处的最大插值误差和插值误差平均值与迭代次数的关系. 图 2-1 中的子图 2-1c 和子图 2-1d 展示了格点处的最大插值误差和插值误差平均值与迭代次数的关系. 这里, 迭代点处的插值误差和网格点处的插值误差分别定义为

$$\text{Err}_{\text{itr}}(\boldsymbol{z}) = \left| f(\boldsymbol{z}) - \boldsymbol{Q}_k(\boldsymbol{z}) \right|$$

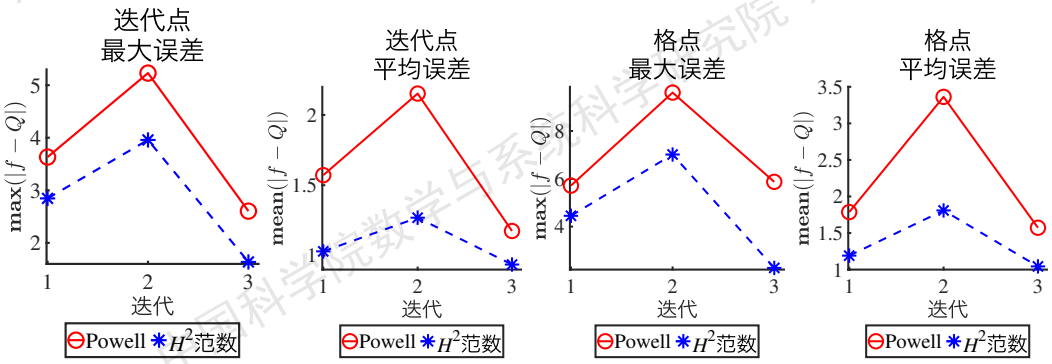
和

$$\text{Err}_{\text{grid}}(\boldsymbol{z}_{pq}) = \left| f(\boldsymbol{z}_{pq}) - \boldsymbol{Q}_k(\boldsymbol{z}_{pq}) \right|,$$

其中 $\boldsymbol{z}$ 是历史迭代点, 且 $\boldsymbol{z}_{pq} = (\frac{p}{100}, \frac{q}{100})^\top$, $p, q \in [-100, 100] \cap \mathbb{Z}$.

图 2-1 中插值误差的区别展示了我们的最小 $\boldsymbol{H}^2$ 范数二次模型的优势. 在迭代过程中, 我们的模型在旧的被舍弃的插值点和区域$\{\boldsymbol{x} = (x_1, x_2)^\top : x_1, x_2 \in [-1, 1]\}$ 中的网格点上的插值误差比最小 Frobenius 范数更新二次模型要小. 换句话说, 在本例中, 我们可以观察到最小 $\boldsymbol{H}^2$ 范数更新在数值上是更稳定的.

值得注意的是, 我们在这里设计的目标函数将 $\boldsymbol{y}_{\text{new}}$ 处插值约束中的函数值缩放为 1, 目标函数本身是不连续的. 考虑到模型函数是连续的, 并且上述设置可以帮助我们在相对公平和简单的条件下进行清晰的观察, 所以我们不必要求这里的插值误差总是非常小. 当然, 在数值上, 更小的误差是更受欢迎的.



(a) 历史迭代点最大误差 (b) 历史迭代点平均误差 (c) 格点最大误差 (d) 格点平均误差

图 2-1 不同插值二次模型的插值误差对比

Figure 2-1 Interpolation error comparison of different interpolation quadratic models

下面的例子进一步展示了在迭代地求解一个简单经典的测试问题时, 最小 H^2 范数更新二次模型的优势.

例 2.2. 我们测试的目标函数是 2 维 Rosenbrock 函数 [72]

$$f(\boldsymbol{x}) = f(x_1, x_2) = \left(1 - x_1\right)^2 + 100\left(x_2 - x_1^2\right)^2,$$

其中 x_1 和 x_2 表示变量 $\boldsymbol{x}$ 的第 1 和第 2 个分量. 实验中使用的初始插值点是原点和单位圆上 3 个均匀分布的点, 即

$$\boldsymbol{y}_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \ \boldsymbol{y}_2 = \begin{pmatrix} \frac{\sqrt{3}}{2} \\ \frac{1}{2} \end{pmatrix}, \ \boldsymbol{y}_3 = \begin{pmatrix} -\frac{\sqrt{3}}{2} \\ \frac{1}{2} \end{pmatrix}, \ \boldsymbol{y}_4 = \begin{pmatrix} 0 \\ -1 \end{pmatrix},$$

这是在 $\mathfrak{R}^2$ 中使用 4 个插值点时的一个简单且通用的设置. 这个例子的目的是从极小化一个经典函数的角度比较这两个模型. 根据 Powell [94] 的建议, 我们选择的插值点数量为 $n + 2$, 这样可以在极小化一个 n 维目标函数时, 至少对最小 Frobenius 范数更新二次模型的 Hessian 矩阵提供一个约束, 这里 $n = 2$.

我们分别使用基于最小 Frobenius 范数更新二次模型和最小 H^2 范数更新二次模型的无导数信赖域方法来迭代地极小化 2 维 Rosenbrock 函数. 初始插值点如上所述, 初始信赖域半径为 1. 此外, 信赖域半径和模型梯度范数的容差设置为 10^{-8}, 另外, $\mu = 0.1$. 更新信赖域半径的参数为 $\gamma = 2, \hat{\eta}_1 = \frac{1}{2}, \hat{\eta}_2 = \frac{1}{4}$. 图 2-2 显示了迭代结果, 关于函数值探测次数、最终函数值、模型梯度范数和数值最优点的细节在表 2-1 中给出. 符号 NF 表示函数值探测次数. 可以看到, 求解本例时, 使用我们模型的算法具有比使用最小 Frobenius 范数更新二次模型的算法更快的数值收敛性, 这在一定程度上依赖于我们模型的高逼近精度. 这个实验显示了在极小化 2 维 Rosenbrock 函数时, 最小 H^2 范数更新二次模型与最小 Frobenius 范数更新二次模型相比具有优势.

此外, 我们对基于我们的最小 H^2 范数更新二次模型的算法进行了进一步数值实验, 用以探究 “使用不同数量的插值点” 的影响, 也就是说我们分别把 m 赋值

表 2-1 例 2.2 中的函数值探测次数、最终函数值、模型梯度范数和解

Table 2-1 Numbers of function evaluations, final function values, model gradient norms, and the best points for Example 2.2

模型	NF	最终 f 值	模型梯度范数	数值最优点
Powell	67	3.8630×10^{-9}	0.0015	$(1.00005607, 1.00011483)^{\mathsf{T}}$
最小 H^2 范数更新	49	7.8825×10^{-12}	7.8587×10^{-6}	$(1.00000271, 1.00000535)^{\mathsf{T}}$

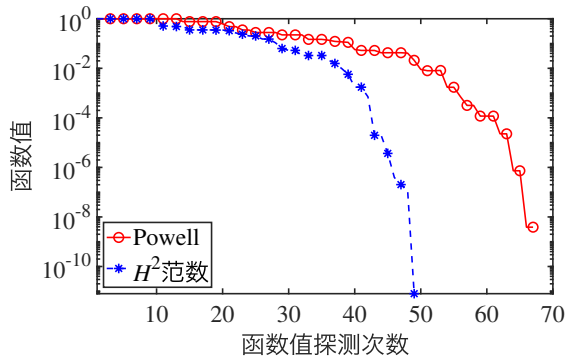


图 2-2 基于 Powell 的最小 Frobenius 范数更新二次模型和我们的最小 H^2 范数更新二次模型极小化 2 维 Rosenbrock 函数的收敛图

Figure 2-2 Convergence plot of minimizing 2-dimensional Rosenbrock function based on Powell's least Frobenius norm updating model and our least H^2 norm updating model

为 1 到 $\frac{1}{2}(n+1)(n+2)$. 表 2-2 显示了用基于使用不同数量的插值点的最小 H^2 范数更新二次模型函数的算法极小化 2 维 Rosenbrock 函数时的函数值探测次数. 其他设置与之前的设置相同.

关于在每步迭代中插值点最佳数量的主要考虑是: 较少的插值点具有较低的计算成本, 并且可能在定理 2.4 中的投影性质的保证下提供更好的更新. 这是因为当为了得到 Q_k 而求解问题 (2-5) 时, 如果有较少的插值约束, $\|Q_k - Q_{k-1}\|_{H^2(\Omega)}$ 可以更小. 注意, 这里的比较是在不同数量的插值条件的意义上进行的. 此外, 插值点的数量也可以根据优化过程中所需的精确程度来动态选择.

我们的方法尚有改进空间, 例如, 改变算法中的相关参数 (例如, 系数 C_1, C_2, C_3 和插值点的数量) 可能会导致不同的结果. 下面给出一个例子, 其中最小 Frobenius 范数更新二次模型的表现优于最小 H^2 范数更新二次模型, 这也表明在某些情况下最小 Frobenius 范数更新二次模型在数值上可以是更好的选择.

例 2.3. 在本例中, 我们尝试极小化测试函数 "DQRTIC" [177], 其表达式为

$$f(\boldsymbol{x}) = f(x_1, x_2) = (x_1 - 1)^4 + (x_2 - 2)^4.$$

该例子的全局最小值为 0. 我们把信赖域半径固定为 1. 初始插值点 $\boldsymbol{y}_1, \boldsymbol{y}_2, \boldsymbol{y}_3, \boldsymbol{y}_4, \boldsymbol{y}_5$

表 2-2 使用不同数量的插值点极小化 Rosenbrock 函数

Table 2-2 Minimizing Rosenbrock function with different number of interpolation points

初始插值点						NF
$(0,0)^\top$	-	-	-	-	-	56
$(0,0)^\top$	$(1,0)^\top$	-	-	-	-	58
$(0,0)^\top$	$(1,0)^\top$	$(0,1)^\top$	-	-	-	60
$(0,0)^\top$	$(\frac{\sqrt{3}}{2},\frac{1}{2})^\top$	$(-\frac{\sqrt{3}}{2},\frac{1}{2})^\top$	$(0,-1)^\top$	-	-	49
$(0,0)^\top$	$(1,0)^\top$	$(0,1)^\top$	$(-1,0)^\top$	$(0,-1)^\top$	-	61
$(0,0)^\top$	$(1,0)^\top$	$(0,1)^\top$	$(-1,0)^\top$	$(0,-1)^\top$	$(\frac{\sqrt{2}}{2},-\frac{\sqrt{2}}{2})^\top$	63

(我们在每次迭代中使用 5 个插值点) 分别为

$$\boldsymbol{y}_1=\begin{pmatrix}-20\\1\end{pmatrix},\ \boldsymbol{y}_2=\begin{pmatrix}-19\\1\end{pmatrix},\ \boldsymbol{y}_3=\begin{pmatrix}-20\\2\end{pmatrix},\ \boldsymbol{y}_4=\begin{pmatrix}-21\\1\end{pmatrix},\ \boldsymbol{y}_5=\begin{pmatrix}-20\\0\end{pmatrix}.$$

前 16 次迭代的最优函数值如图 2-3 所示, 在本例中 Powell 的模型表现优于我们的, 特别是在第 10 次函数值探测和第 16 次函数值探测之间, 它们在第 16 次探测分别达到了函数值 2.08×10^4 和 8.38×10^4.

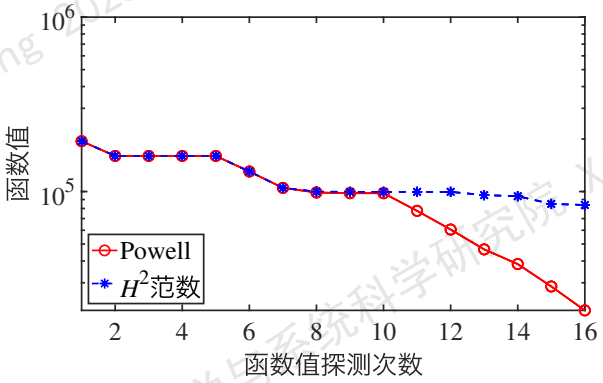


图 2-3 基于 Powell 的最小 Frobenius 范数更新二次模型和我们的最小 H^2 范数更新二次模型极小化 2 维 DQRTIC 函数

Figure 2-3 Minimizing 2-dimensional DQRTIC function based on Powell's least Frobenius norm updating model and our least H^2 norm updating model

为了进一步观察我们基于最小 H^2 范数更新二次模型函数的算法的数值表现, 我们尝试求解一些经典测试问题, 并使用 Performance Profile 和 Data Profile 来展示数值结果. 图 2-4 中的 Performance Profile 和图 2-5 中的 Data Profile 对应的测试问题在表 2-3 中给出, 它们都选自经典常见的无约束优化测试函数集, 测试优化问题中的目标函数是光滑的. 对于 Performance 和 Data Profile, 所有测试算法关于探测次数的停机条件都设置为函数值探测次数不超过 $100n$ 次, 其中 n 表

示相应问题的维数. 这里的 $\boldsymbol{x}_{\text{int}}$ 表示起始点, $\boldsymbol{x}^*$ 表示已知 (数值比较中获得) 的最好点.

表 2-3 图 2-4 和图 2-5 对应的 50 个测试问题

Table 2-3 50 test problems for Figure 2-4 and Figure 2-5

问题	维数	$f(\boldsymbol{x}_{\text{int}})$	$f(\boldsymbol{x}^*)$
ARGLINA [177, 178]	8	4.00×10^{1}	8.00
ARGLINB [177, 178]	10	8.66×10^{6}	4.63
ARGTRIG [177]	8	8.45×10^{-3}	5.01×10^{-14}
BDQRTIC [177]	100	2.17×10^{4}	3.79×10^{2}
BDVALUE [177, 178]	100	1.23×10^{-6}	9.33×10^{-7}
BRYBND [177, 178]	180	6.48×10^{3}	1.44×10^{-9}
CHAINWOO [177, 179]	140	5.16×10^{5}	2.98×10^{2}
CHEBQUAD [177, 178]	120	1.75×10^{-2}	6.56×10^{-3}
CHNROSNB [177, 180]	80	1.61×10^{3}	3.52×10^{-11}
CHPOWELLS [178, 181]	20	1.10×10^{3}	5.52×10^{-10}
COSINE [177]	90	7.81×10^{1}	-8.90×10^{1}
CUBE [177]	50	3.02×10^{4}	3.51×10^{-3}
CURLY10 [177]	10	-5.06×10^{-5}	-1.00×10^{3}
CURLY20 [177]	20	-1.01×10^{-4}	-2.01×10^{3}
CURLY30 [177]	30	-1.52×10^{-4}	-3.01×10^{3}
DIXMAANE [177]	60	4.45×10^{2}	1.00
DIXMAANF [177]	65	4.70×10^{2}	1.00
DIXMAANG [177]	70	8.81×10^{2}	1.00
DIXMAANH [177]	75	1.82×10^{3}	1.00
DIXMAANI [177]	80	5.26×10^{2}	1.00
DIXMAANJ [177]	85	5.65×10^{2}	1.00
DIXMAANK [177]	90	1.08×10^{3}	1.00
DIXMAANL [177]	95	2.19×10^{3}	1.00
DIXMAANM [177]	100	3.14×10^{2}	1.00
DIXMAANN [177]	105	3.33×10^{2}	1.00
DIXMAANO [177]	120	5.96×10^{2}	1.00
DIXMAANP [177]	130	1.14×10^{3}	1.00
DQRTIC [177]	110	3.01×10^{9}	4.76×10^{-6}
ERRINROS [177]	170	5.39×10^{5}	1.34×10^{2}
EXPSUM [182]	175	1.80×10^{6}	8.03×10^{3}
EXTROSNB [177, 180]	180	7.16×10^{4}	8.33×10^{-4}
FLETCHCR [177]	165	1.64×10^{4}	4.05×10^{-2}

表 2-3 (续表)

FREUROTH [177, 178]	100	9.96×10^{4}	1.08×10^{4}
GENROSE [177]	130	5.14×10^{2}	1.18×10^{2}
INTEGREQ [177, 178]	110	6.30×10^{-1}	3.84×10^{-12}
MOREBV [177, 178]	8	1.37×10^{-3}	6.50×10^{-14}
NCB20 [177]	175	5.47×10^{3}	2.81×10^{2}
NONDQUAR [177]	160	1.66×10^{2}	2.88×10^{-4}
POWELLSG [177, 178]	180	9.68×10^{3}	1.89×10^{-3}
POWER [177]	135	8.29×10^{5}	3.78×10^{-20}
ROSENBROCK [177, 178]	10	3.64×10^{3}	4.69×10^{-7}
SBRYBND [177, 178]	50	7.68×10^{2}	1.98×10^{1}
SCOSINE [177]	180	1.03×10^{1}	-5.45×10^{1}
SPARSINE [177]	160	5.33×10^{4}	1.47×10^{-5}
SPMSRTLS [177]	180	1.30×10^{2}	9.84×10^{-11}
SROSENBR [177, 178]	8	9.68×10^{1}	4.58×10^{-2}
TOINTGSS [177]	100	8.92×10^{2}	9.71
TQUARTIC [177]	20	8.10×10^{-1}	1.12×10^{-12}
WOODS [177, 178]	24	1.15×10^{5}	2.45×10^{1}
VARDIM [177, 178]	180	1.41×10^{16}	4.15

基于上述插值误差和简单例子的数值结果, 最小 H^2 范数更新二次模型函数对于例 2.1 和例 2.2 来说求解效果更好. 下面的数值结果将显示, 对于所测试的问题集, 使用我们模型的算法比使用最小 Frobenius 范数更新二次模型的算法在数值上收敛得更快、效果更好.

对于这个实验中的每个问题, 所有算法都从一致的输入点 $\boldsymbol{x}_{\text{int}}$ 开始, 精度 τ 分别设置为10^{-1}、10^{-3} 和 10^{-5}. 算法框架如算法 3 所示, “Powell” 表示使用 Powell 的最小 Frobenius 范数更新二次模型. 这里, 对于使用 Powell 最小 Frobenious 范数更新二次模型的算法, 每次迭代的插值点数为 $m = 2n + 1$. “H^2 $(m = 2n + 1)$” 和 “H^2 $(m = \lceil \frac{n}{2} \rceil + 1)$” 都使用最小 H^2 范数更新二次模型, 并与 “Powell” 共用相同的框架. 对于图 2-4 中的三种算法, 信赖域半径和梯度范数的容差设置为 10^{-8}. 它们共用相同的初始信赖域半径. 算法 3 中的参数为 $\gamma = 2, \hat{\eta}_1 = \frac{1}{2}, \hat{\eta}_2 = \frac{1}{4}, \mu = 0.1$. 为了实现与其他二次模型的公平比较, 方法 “Powell” 和 “H^2 $(m = 2n + 1)$” 在每次迭代中使用 $2n+1$ 个插值点, 并共用相同的初始插值点 $\boldsymbol{x}_{\text{int}}, \boldsymbol{x}_{\text{int}} \pm \boldsymbol{e}_i, i = 1, \cdots, n$. 此外, 方法 “$H^2$ $(m = \lceil \frac{n}{2} \rceil + 1)$” 在每次迭代中使用 $\lceil \frac{n}{2} \rceil + 1$ 个插值点, 其初始插值点为 $\boldsymbol{x}_{\text{int}}, \boldsymbol{x}_{\text{int}} + \boldsymbol{e}_j, j = 1, \cdots, \lceil \frac{n}{2} \rceil + 1$.

事实上, 求解不同问题时选择不同的 m 会有不同的数值表现, 考虑到最小 H^2 范数更新二次模型已经减小了每步插值点个数的下界, 这一点值得我们在未来进一步研究. “H^2 $(m = \lceil \frac{n}{2} \rceil + 1)$” 的性能可以显示出我们的方法和模型在每次迭代使用更少插值点的数值优势.

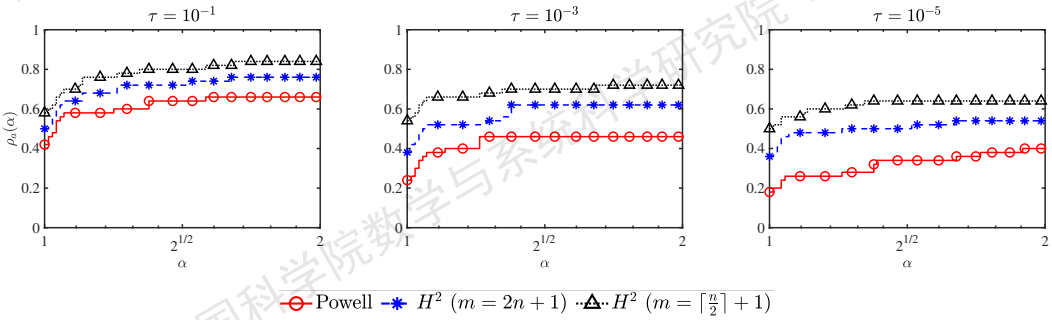


图 2-4 基于不同二次模型的无导数信赖域算法求解测试问题的 Performance Profile

Figure 2-4 Performance Profile of solving test problems with derivative-free trust-region algorithms based on different quadratic models

在图 2-4 中, $\rho_a(1)$ 的值的相关结果表明: 基于最小 H^2 范数更新二次模型且 $m = \lceil \frac{n}{2} \rceil + 1$ 的算法可以在所展示的三种情况中用最少的函数值探测次数求解最多的问题. 使用最小 H^2 范数更新二次模型的两个算法比使用最小 Frobenius 范数更新二次模型的算法在这个测试集上表现的更好.

为了进一步在数值上观察我们基于最小 H^2 范数更新二次模型的算法的整体性能, 我们在下面使用 Data Profile 给出了基于我们模型的算法与其他算法之间的数值比较结果. 用于测试的使用 Powell 模型 (最小 Frobenius 范数更新二次模型) 和最小 Frobenius 范数二次模型 [160] 的无导数信赖域算法分别是 PDFO [142] 中的 NEWUOA 的 Python 接口 [94] 和 DFO 算法的 Python 实现[4]. 此外, Nelder-Mead 单纯形算法和使用一阶差分估计导数的 BFGS 方法是从 scipy.optimize 库获得的[5].

对于实验中的每个问题, 所有算法都从每个问题对应的输入点 $\boldsymbol{x}_{\text{int}}$ 开始, 精度 τ 分别设置为 10^{-1}、10^{-3} 和 10^{-5}. 我们保持方法 "H^2 $(m = 2n + 1)$" 和 "H^2 $(m = \lceil \frac{n}{2} \rceil + 1)$" 的设置与第 2.1.4 节中的设置相同. "$H^1$ $(m = 2n + 1)$" 与我们的模型共用相同的算法框架和设置, 但其使用最小 H^1 半范数二次模型, 即经典的 $\left\|\nabla^2 Q\right\|_F^2$ 和 $\|\nabla Q\|_2^2$ 组合. 对于图 2-5 中的信赖域算法 "NEWUOA" 和 "DFO-py", 信赖域半径和模型梯度范数的容差分别设置为 10^{-8}. 它们共用相同的初始信赖域半径. 此外, 在我们的数值实验中, 信赖域方法 "NEWUOA" 和 "DFO-py" 的每次迭代使用 $2n + 1$ 个插值点, 并与 "H^2 $(m = 2n + 1)$" 共用相同的初始插值点. 对于 "Nelder-Mead-py", 初始单纯形是一个 $n + 1$ 维单纯形, 顶点为 $\boldsymbol{x}_{\text{int}}, \boldsymbol{x}_{\text{int}} + \boldsymbol{e}_i, i = 1, \cdots, n$, 并且迭代间函数值的绝对误差界限设置为 10^{-8}. 对于 "BFGS", 用于数值逼近梯度的相对步长通过设置为 "None" 自动选择, 相应的梯度范数在成功终止前必须小于 10^{-8}.

我们可以从图 2-5 中观察到, 基于最小 H^2 范数更新二次模型函数的信赖域算法比其他算法表现得更好. 更具体地说, 当 β 大约超过 40 时, 在 $\tau = 10^{-1}$ 的情况下, 它们都可以求解超过 60% 的问题. 使用每次迭代满足 $m = \lceil \frac{n}{2} \rceil + 1$ 个插值

[4] https://coral.ise.lehigh.edu/katyas/software

[5] https://docs.scipy.org/doc/scipy

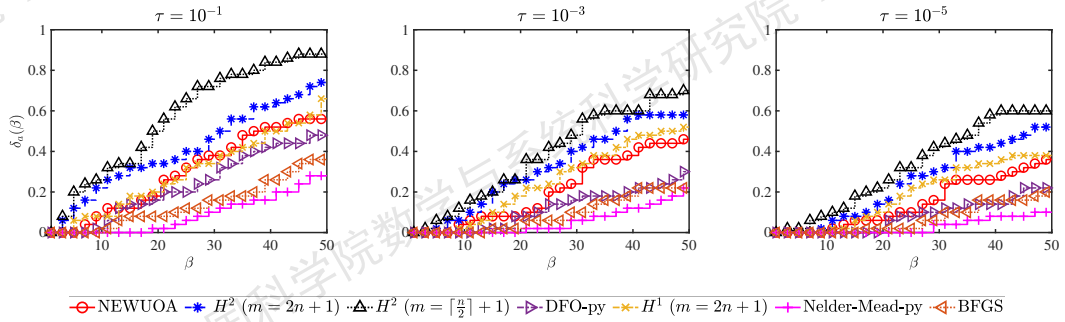


图 2-5 使用不同算法求解测试问题的 Data Profile

Figure 2-5 Data Profile of solving test problems with different algorithms

条件方程的模型的方法比其他方法更好. 上述结果显示了使用我们模型的算法的优势.

2.1.5 小结

对于无导数信赖域优化方法, 使用欠定二次插值模型可以在一定程度上减少插值点的数量以及函数值探测的次数. 为了获得唯一的二次模型函数, 一种常规方法是通过求解带插值条件约束的优化问题来确定每次迭代中二次函数的系数. 本节尝试通过最小化迭代中新旧二次模型函数变化量的 H^2 范数来获得二次模型函数, 我们发现这样可以合理地减小插值点或方程数量的下界. 我们在给出了投影性质和误差界, 还基于相应优化问题的 KKT 条件推导出了计算模型函数系数的相应更新公式. 基于求解 KKT 系统 (2-19) 和公式 (2-22) 得到了 KKT 逆矩阵的更新公式, 为欠定最小范数更新类型的二次模型提供了更多选择. 数值结果从不同角度表明了使用我们模型的算法的良好性能.

关于未来的工作, 我们还可以研究和发展基于最小 H^2 范数更新二次模型的无导数信赖域算法的更多收敛性质. 我们也可以为具有不同结构的问题设计自适应权重系数, 并通过最小化

$$C_1^{(k)} \left\| Q - Q_{k-1} \right\|_{H^0(\Omega)}^2 + C_2^{(k)} \left| Q - Q_{k-1} \right|_{H^1(\Omega)}^2 + C_3^{(k)} \left| Q - Q_{k-1} \right|_{H^2(\Omega)}^2$$

来获得第 k 次迭代的二次模型函数, 其中 k 与第 k 次迭代相对应. 另一个潜在的未来工作是寻找每次迭代中构造模型时插值点数量的更好选择, 这是因为我们现在已经实现了该数量下界的减小. 如第 2.1.4 节 (特别是例 2.3) 所示, 我们的模型仍有局限性, 因此寻找性能与系数或插值点数量之间更深层次的关系将是有价值的. 其他不同类型的无导数优化插值模型也可以作为进一步研究的对象, 包括适用于大规模稀疏无导数优化问题的欠定模型函数, 以及适合在求解具有非线性或线性约束的优化问题时使用的模型.

2.2 最小加权 H^2 范数更新二次插值模型

本节主要在前一节的基础上更进一步地讨论最小加权 H^2 范数更新二次插值模型, 进而应用于基于模型的无导数信赖域方法中. 这里的最小加权 H^2 范数

更新二次插值模型指的是求解子问题 (2-6) 所获得的解. 值得注意的是, 在基于模型的无导数信赖域方法中, 随着迭代次数的增加, 信赖域半径将收敛到 0, 本节将关注这种情况. 同时, 这种情况也适用于具有高精度环境的数值计算情形.

具体来说, 本节围绕最小加权 H^2 范数更新二次模型及其对应的 KKT 矩阵, 定义了新的距离和误差, 分别称为 KKT 矩阵距离和 KKT 矩阵误差; 给出了系数区域重心的定义, 并提供了最小加权 H^2 范数更新二次模型的权重系数区域的解析重心; 最后为权重系数区域的最佳选择提供了数值支持.

本节后面部分的内容组织如下. 第 2.2.1 节给出了最小加权 H^2 范数更新二次模型和 KKT 矩阵的更多细节. 展示了在计算对应的 H^2 范数所使用的半径 r 收敛到 0 的情况下相应的结果. 此外, 第 2.2.2 节提出了新的距离和误差, 分别称为 KKT 矩阵距离和 KKT 矩阵误差, 用以说明不同权重系数对 KKT 矩阵造成的差异. 该节还给出了系数区域重心的定义. 第 2.2.3 节给出了信赖域半径很小时最小加权 H^2 范数更新二次模型的权重系数区域的重心. 第 2.2.4 节展示了不同权重系数的数值性能比较结果.

2.2.1　最小加权 H^2 范数更新二次模型和 KKT 矩阵

根据前文, 系数 r 通常与信赖域半径和此次迭代中插值点与当前信赖域中心的最远距离成比例关系, 其中后者与信赖域半径成比例. 因此, $r \to 0$ 对应信赖域半径很小的情况.

注意, 我们把 (2-14) 中的 $\eta_1, \eta_2, \eta_3, \eta_4, \eta_5$ 分别写成函数形式:

$$\eta_1(C_1, C_2, C_3, n, r),\ \eta_2(C_1, C_2, n, r),\ \eta_3(C_1, n, r),\ \eta_4(C_1, n, r),\ \eta_5(C_1).$$

KKT 矩阵 $\boldsymbol{W}$ 的参数在信赖域半径很小的极限情况下的一些结果如以下命题所示.

命题 2.12. *$\eta_1, \eta_2, \eta_3, \eta_4, \eta_5$ 在 $r \to 0$ 时有*

$$\begin{cases} \lim\limits_{r\to 0} \eta_1(C_1, C_2, C_3, n, r) = C_3, \\ \lim\limits_{r\to 0} \eta_2(C_1, C_2, n, r) = C_2, \\ \lim\limits_{r\to 0} \eta_3(C_1, n, r) = 0, \\ \lim\limits_{r\to 0} \eta_4(C_1, n, r) = 0, \\ \lim\limits_{r\to 0} \eta_5(C_1) = C_1, \end{cases} \tag{2-27}$$

进而

$$\begin{cases} \lim\limits_{r\to 0} \dfrac{1}{8\eta_1} = \dfrac{1}{8C_3}, \\ \lim\limits_{r\to 0} -\dfrac{\eta_3}{8\eta_1\left(n\eta_3+\eta_1\right)} = 0, \\ \lim\limits_{r\to 0} -\dfrac{\eta_4}{4n\eta_3+4\eta_1} = 0, \\ \lim\limits_{r\to 0} -2\eta_2 = -2C_2, \\ \lim\limits_{r\to 0} \dfrac{n\eta_4^2}{2n\eta_3+2\eta_1} - 2\eta_5 = -2C_1. \end{cases}$$

证明. 直接计算可得结果. □

2.2.2 KKT 矩阵误差与系数区域的重心

我们知道，基于给定的 C_1 和 C_2 来计算相应的参数 $\boldsymbol{\lambda}, c, \boldsymbol{g}$ 可以得到最小加权 H^2 范数更新二次模型. 考虑到 $\boldsymbol{\lambda}, c, \boldsymbol{g}$ 直接依赖于 KKT 方程 (2-19) 中的 KKT 矩阵 $\boldsymbol{W}$, 我们试图使用 KKT 矩阵距离来刻画两个模型之间的距离, 具体定义见定义 2.16. 具体来说, 如 KKT 系统 (2-19) 所示, 向量 $(0,\cdots,0,f(\boldsymbol{y}_{\text{new}})-\boldsymbol{Q}_{k-1}(\boldsymbol{y}_{\text{new}}),0,\cdots,0)^{\mathsf{T}}$ 对于不同的权重系数保持不变, 唯一的区别是 (2-19) 左侧的 KKT 矩阵 $\boldsymbol{W}$. 这里的 KKT 矩阵 $\boldsymbol{W}$ 直接决定了我们二次模型的参数, 构造了与之对应的最小加权 H^2 范数更新二次模型. 因此, 一个事实是 KKT 矩阵可以充分表示通过最小化具有不同权重系数的加权 H^2 范数给出的不同模型之间的差异, 换言之, KKT 矩阵在一定意义上是衡量对应二次模型好坏的一个关键.

本节试图找到上述权重系数的一个 “平衡点”. 本节旨在通过给出权重系数区域的重心来找到一种中心 KKT 矩阵, 以寻找这种意义下的最优系数.

注 2.5. 不失一般性, 我们假设 $C_1+C_2+C_3=1$, 这并不影响我们对权重系数的讨论.

下面的分析在下述假设下进行.

假设 2.13. $\boldsymbol{W}$ 和 $\boldsymbol{W}^*$ 分别是根据 (2-19) 中的相应权重系数 C_1, C_2 和 C_1^*, C_2^* 确定的模型函数对应的 KKT 矩阵.

我们有以下定理.

定理 2.14. *假设 $\boldsymbol{W}$ 和 $\boldsymbol{W}^*$ 满足假设 2.13, 则 $\left\|\boldsymbol{W}-\boldsymbol{W}^*\right\|_F^2$ 为*

$$\begin{aligned} \left\|\boldsymbol{W}-\boldsymbol{W}^*\right\|_F^2 = & \left\{\sum_{i=1}^{m}\sum_{j=1}^{m}\left[\left(\boldsymbol{y}_i-\boldsymbol{x}_0\right)^{\mathsf{T}}\left(\boldsymbol{y}_j-\boldsymbol{x}_0\right)\right]^4\right\}\left(\frac{1}{8\eta_1}-\frac{1}{8\eta_1^*}\right)^2 \\ & +\left(\sum_{i=1}^{m}\sum_{j=1}^{m}\left\|\boldsymbol{y}_i-\boldsymbol{x}_0\right\|_2^4\left\|\boldsymbol{y}_j-\boldsymbol{x}_0\right\|_2^4\right)\left(\frac{\eta_3}{8\eta_1\left(n\eta_3+\eta_1\right)}-\frac{\eta_3^*}{8\eta_1^*\left(n\eta_3^*+\eta_1^*\right)}\right)^2 \end{aligned}$$

$$+\left(\sum_{i=1}^{m}\left\|\boldsymbol{y}_i-\boldsymbol{x}_0\right\|_2^4\right)\left[-\frac{\eta_4}{4n\eta_3+4\eta_1}-\left(-\frac{\eta_4^*}{4n\eta_3^*+4\eta_1^*}\right)\right]^2$$
$$+n\left[2\left(\eta_2^*-\eta_2\right)\right]^2+\left[\frac{n\eta_4^2}{2n\eta_3+2\eta_1}-2\eta_5-\left(\frac{n(\eta_4^*)^2}{2n\eta_3^*+2\eta_1^*}-2\eta_5^*\right)\right]^2,$$

其中 $\eta_1,\eta_2,\eta_3,\eta_4,\eta_5$ 按 *(2-14)* 定义, $\eta_1^*,\eta_2^*,\eta_3^*,\eta_4^*,\eta_5^*$ 为

$$\begin{cases}\eta_1^*=C_1^*\dfrac{r^4}{2(n+4)(n+2)}+C_2^*\dfrac{r^2}{n+2}+C_3^*,\\ \eta_2^*=C_1^*\dfrac{r^2}{n+2}+C_2^*,\\ \eta_3^*=C_1^*\dfrac{r^4}{4(n+4)(n+2)},\\ \eta_4^*=C_1^*\dfrac{r^2}{n+2},\\ \eta_5^*=C_1^*.\end{cases}$$

证明. 计算 $\boldsymbol{A}-\boldsymbol{A}^*,\boldsymbol{J}-\boldsymbol{J}^*$ 以及 $\boldsymbol{X}-\boldsymbol{X}^*$, 再计算平方和后可得到该结论. □

此外, 我们可以得到关于 $\|\boldsymbol{W}-\boldsymbol{W}^*\|_F^2$ 的以下推论.

推论 2.15. 假定 $\boldsymbol{W}$ 和 $\boldsymbol{W}^*$ 满足假设 *2.13*, 则 $\left\|\boldsymbol{W}-\boldsymbol{W}^*\right\|_F^2$ 是 C_1,C_2,C_1^*,C_2^*,n,r 的函数, 我们有

$$\begin{aligned}&\left\|\boldsymbol{W}-\boldsymbol{W}^*\right\|_F^2:=\mathcal{D}(C_1,C_2,C_1^*,C_2^*,n,r)\\&=\mathcal{R}_1(\boldsymbol{y}_1,\cdots,\boldsymbol{y}_m)\mathcal{P}_1(C_1,C_2,C_1^*,C_2^*,n,r)+\mathcal{R}_2(\boldsymbol{y}_1,\cdots,\boldsymbol{y}_m)\mathcal{P}_2(C_1,C_2,C_1^*,C_2^*,n,r)\\&+\mathcal{R}_3(\boldsymbol{y}_1,\cdots,\boldsymbol{y}_m)\mathcal{P}_3(C_1,C_2,C_1^*,C_2^*,n,r)+\mathcal{P}_4(C_1,C_2,C_1^*,C_2^*,n,r),\end{aligned}\tag{2-28}$$

其中

$$\begin{cases}\mathcal{R}_1(\boldsymbol{y}_1,\cdots,\boldsymbol{y}_m)=\displaystyle\sum_{i=1}^{m}\sum_{j=1}^{m}\left[\left(\boldsymbol{y}_i-\boldsymbol{x}_0\right)^{\top}\left(\boldsymbol{y}_j-\boldsymbol{x}_0\right)\right]^4,\\ \mathcal{R}_2(\boldsymbol{y}_1,\cdots,\boldsymbol{y}_m)=\displaystyle\sum_{i=1}^{m}\sum_{j=1}^{m}\left\|\boldsymbol{y}_i-\boldsymbol{x}_0\right\|_2^4\left\|\boldsymbol{y}_j-\boldsymbol{x}_0\right\|_2^4,\\ \mathcal{R}_3(\boldsymbol{y}_1,\cdots,\boldsymbol{y}_m)=\displaystyle\sum_{i=1}^{m}\left\|\boldsymbol{y}_i-\boldsymbol{x}_0\right\|_2^4\end{cases}$$

在给定当前迭代的基点 $\boldsymbol{x}_0$ 后仅依赖于插值点 $\boldsymbol{y}_1,\cdots,\boldsymbol{y}_m$, 另外, $\mathcal{P}_1,\mathcal{P}_2,\mathcal{P}_3,\mathcal{P}_4$ 是 C_1,C_2,C_1^*,C_2^*,n,r 的函数, 具体表达式将在后面给出.

证明. 将 $\eta_1,\eta_2,\eta_3,\eta_4,\eta_5$ 和 $\eta_1^*,\eta_2^*,\eta_3^*,\eta_4^*,\eta_5^*$ 的表达式带入即可以得到 (2-28). □

为了进一步讨论中心 KKT 矩阵, 我们首先给出 KKT 矩阵距离和 KKT 矩阵误差的定义.

定义 2.16 (KKT 矩阵距离)**.** *我们定义两个 KKT 矩阵 $\boldsymbol{W}$ 和 $\boldsymbol{W}^*$ 之间的 KKT 矩阵距离为 $\|\boldsymbol{W}-\boldsymbol{W}^*\|_F$.*

定理 2.17. *定义 2.16 中的 KKT 矩阵距离是 KKT 矩阵集上一个良定义的距离.*

证明. 我们有以下事实.

- KKT 矩阵距离是非负的, 并且 $\left\|\boldsymbol{W}-\boldsymbol{W}^*\right\|_F=0$ 当且仅当 $\boldsymbol{W}=\boldsymbol{W}^*$.
- 对称性 $\left\|\boldsymbol{W}-\boldsymbol{W}^*\right\|_F=\|\boldsymbol{W}^*-\boldsymbol{W}\|_F$ 成立.
- 三角不等式

$$\left\|\boldsymbol{W}-\bar{\boldsymbol{W}}^*\right\|_F=\left\|\left(\boldsymbol{W}-\boldsymbol{W}^*\right)+\left(\boldsymbol{W}^*-\bar{\boldsymbol{W}}^*\right)\right\|_F\leq\left\|\boldsymbol{W}-\boldsymbol{W}^*\right\|_F+\left\|\boldsymbol{W}^*-\bar{\boldsymbol{W}}^*\right\|_F.$$

因此, KKT 矩阵距离是一个良定义的距离. □

定义 2.18 (KKT 矩阵误差)**.** *假定 $\boldsymbol{W}$ 和 $\boldsymbol{W}^*$ 满足假设 2.13, 我们定义两组权重系数 (C_1,C_2) 和 (C_1^*,C_2^*) 之间的 KKT 矩阵误差为 $\sqrt{D(C_1,C_2,C_1^*,C_2^*,n,r)}$, 其中 $D(C_1,C_2,C_1^*,C_2^*,n,r)$ 在 (2-28) 中定义.*

我们将使用 KKT 矩阵误差来帮助寻找适当的权重系数组 C_1,C_2 和 $C_3=1-C_1-C_2$, 其中 $(C_1,C_2)^\mathsf{T}$ 位于区域 $\mathcal{C}$ 内. 图 2-6 展示了系数区域 $\mathcal{C}$. 为了避免在 $r\to 0$ 时 (2-19) 中 KKT 矩阵的分母 η_1 过小, 我们在分析时假设 C_3 有一个下界 ε, 且 $0<\varepsilon<1$.

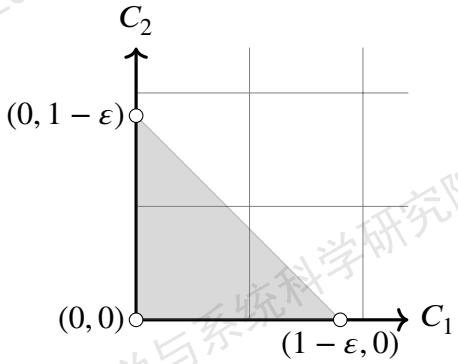


图 2-6 系数区域 $\mathcal{C}$

Figure 2-6 Coefficient region $\mathcal{C}$

为了进一步介绍平均 KKT 矩阵距离, 我们给出以下定义.

定义 2.19 (平均平方 KKT 矩阵误差)**.** *考虑 (2-28), 给定一对系数 C_1 和 C_2, 我们定义平均平方 KKT 矩阵误差为*

$$\begin{aligned}\mathrm{Error}_{\mathrm{ave}}(C_1,C_2,n,r,\varepsilon)&:=\frac{\int_0^{1-\varepsilon-C_1^*}\int_0^{1-\varepsilon}D(C_1,C_2,C_1^*,C_2^*,n,r)dC_1^*dC_2^*}{\int_0^{1-\varepsilon-C_1}\int_0^{1-\varepsilon}1dC_1^*dC_2^*}\\&=\frac{2\int_0^{1-\varepsilon-C_1^*}\int_0^{1-\varepsilon}D(C_1,C_2,C_1^*,C_2^*,n,r)dC_1^*dC_2^*}{(1-\varepsilon)^2}.\end{aligned}$$

定义 2.20 (系数区域的重心)**.** 最小加权 H^2 范数更新二次模型的权重系数区域 C 的重心是

$$\min_{(C_1,C_2)^\top\in C} \mathrm{Error}_{\mathrm{ave}}(C_1,C_2,n,r,\varepsilon) \tag{2-29}$$

的解.

权重系数区域 C 的重心具有最小的平均平方 KKT 矩阵误差. 注意, 这里是通过 KKT 矩阵误差 (而不是欧氏距离) 来衡量.

2.2.3　最小加权 H^2 范数更新二次模型的权重系数区域的重心

本节给出了信赖域半径很小时最小加权 H^2 范数更新二次模型的权重系数区域的重心的解析结果.

定理 2.21. 给定 C_3 的下界 ε, 若 $r\to 0$, 我们得到

$$\lim_{r\to 0}\mathrm{Error}_{\mathrm{ave}}(C_1,C_2,n,r,\varepsilon)=\mathcal{R}_1(\boldsymbol{y}_1,\cdots,\boldsymbol{y}_m)\mathrm{Error}^{(1)}_{\mathrm{ave}}(C_1,C_2,\varepsilon)+\mathrm{Error}^{(2)}_{\mathrm{ave}}(C_1,C_2,n,\varepsilon),$$

其中 $\mathrm{Error}^{(1)}_{\mathrm{ave}}(C_1,C_2,\varepsilon)$ 为

$$\frac{\frac{(\varepsilon-1)\left(\varepsilon(4C_1+4C_2-5)-2(C_1+C_2-1)^2+\varepsilon^2\right)}{2\varepsilon}+(C_1+C_2-3)(C_1+C_2-1)\ln(\varepsilon)}{32(1-\varepsilon)^2(C_1+C_2-1)^2},$$

$\mathrm{Error}^{(2)}_{\mathrm{ave}}(C_1,C_2,n,\varepsilon)$ 为

$$\frac{1}{6}\left(24{C_1}^2+16C_1(\varepsilon-1)+6{C_2}^2n+\varepsilon(4C_2n-2n-8)-4C_2n+\varepsilon^2(n+4)+n+4\right).$$

证明. 直接计算积分和极限即可得到如上结果. □

接下来, 我们可以得到下面的结果.

定理 2.22. 若 $\mathcal{R}_1(\boldsymbol{y}_1,\cdots,\boldsymbol{y}_m)\to 0$, 则 $C_1=\frac{1-\varepsilon}{3}, C_2=\frac{1-\varepsilon}{3}$ 对应提供中心 *KKT* 矩阵的权重系数组.

证明. 我们有

$$\lim_{\mathcal{R}_1(\boldsymbol{y}_1,\cdots,\boldsymbol{y}_m)\to 0}\mathrm{Error}_{\mathrm{ave}}(C_1,C_2,n,r,\varepsilon)=\mathrm{Error}^{(2)}_{\mathrm{ave}}(C_1,C_2,n,r,\varepsilon) \tag{2-30}$$

以及

$$\left(\frac{1-\varepsilon}{3},\frac{1-\varepsilon}{3}\right)^\top=\arg\min_{(C_1,C_2)^\top\in C}\mathrm{Error}^{(2)}_{\mathrm{ave}}(C_1,C_2,n,\varepsilon). \tag{2-31}$$

因此, 结论成立. □

现在, 我们给出一个数值例子.

例 2.4. 在 $\varepsilon=0.01, n=100$ 的情形下, 我们列出了信赖域半径很小时所采样的权重系数对应的 $\mathrm{Error}^{(1)}_{\mathrm{ave}}$ 和 $\mathrm{Error}^{(2)}_{\mathrm{ave}}$ 的值.

表 2-4 在数值上表明 $(\frac{1-\varepsilon}{3},\frac{1-\varepsilon}{3})^\top$ 在 6 对权重系数中具有最小的 $\mathrm{Error}_{\mathrm{ave}}$.

注意, 在理想情况下 C_3 的下界 ε 收敛到 0. 作为定理 2.22 的极限结果, 最佳的权重系数组是 $C_1=\frac{1}{3}, C_2=\frac{1}{3}, C_3=\frac{1}{3}$.

表 2-4 所采样的权重系数的 $\mathrm{Error}_{\mathrm{ave}}^{(1)}$ 和 $\mathrm{Error}_{\mathrm{ave}}^{(2)}$ 的值, $\varepsilon = 0.01$, $n = 100$

Table 2-4 Values of $\mathrm{Error}_{\mathrm{ave}}^{(1)}$ and $\mathrm{Error}_{\mathrm{ave}}^{(2)}$ for sampled weight coefficients, $\varepsilon = 0.01$, $n = 100$

$(C_1, C_2)^\top$	$(\frac{1-\varepsilon}{3}, \frac{1-\varepsilon}{3})^\top$	$(\frac{1}{2}-\varepsilon, \frac{1}{2})^\top$	$(0, \frac{1}{2})^\top$	$(1-\varepsilon, 0)^\top$	$(0, 1-\varepsilon)^\top$	$(0, 0)^\top$
$\mathrm{Error}_{\mathrm{ave}}^{(1)}$	5.663	8.655	8.988	18.3	49.66	16.99
$\mathrm{Error}_{\mathrm{ave}}^{(2)}$	2.467	136.2	2.611	136.2	136.2	2.795

2.2.4 数值结果

我们通过在算法 3 中使用上述的不同模型来给出如下数值例子.

例 2.5. 在这个数值例子中, 我们使用基于最小加权 H^2 范数更新二次模型 (使用不同的权重系数来构造) 的无导数算法来极小化 2 维 Rosenbrock 函数

$$f(\boldsymbol{x}) = \left(1 - x_1\right)^2 + 100\left(x_2 - x_1^2\right)^2, \tag{2-32}$$

这是一个光滑的非凸函数 (如前所述), 且 $\boldsymbol{x} = (x_1, x_2)^\top$. 测试问题的全局极小值为 0, 对应的极小点是 $(1, 1)^\top$.

在实验中, 初始输入点设置为 $\boldsymbol{x}_{\mathrm{int}} = (1.04, 1.1)^\top$. 在每次迭代中, 我们使用 5 个点进行插值. 函数值探测的最大次数被统一设定为 16, 初始信赖域半径为 $\Delta_0 = 10^{-4}$. 此外, 信赖域半径、函数值和梯度范数的容差都设置为 10^{-8}. 算法 3 中的参数为 $\gamma = 2, \hat{\eta}_1 = \frac{1}{4}, \hat{\eta}_2 = \frac{3}{4}, \mu = 0.1$. 初始插值点分别为 $\boldsymbol{x}_{\mathrm{int}}$, $\boldsymbol{x}_{\mathrm{int}} \pm (\Delta_0, 0)^\top$, $\boldsymbol{x}_{\mathrm{int}} \pm (0, \Delta_0)^\top$. 表 2-5 列出了 6 种不同的经典 (半) 范数, 这个数值例子将包括使用它们来构造最小加权 H^2 范数二次模型函数的相应结果.

表 2-5 不同的 (半) 范数及其在系数集中对应的系数

Table 2-5 Different (semi-)norms with corresponding coefficients in the coefficient set

权重系数			对应的 (半) 范数	编号
$C_1 = \frac{1}{3}$	$C_2 = \frac{1}{3}$	$C_3 = \frac{1}{3}$	H^2 范数	(a)
$C_1 = \frac{1}{2}$	$C_2 = \frac{1}{2}$	$C_3 = 0$	H^1 范数	(b)
$C_1 = 0$	$C_2 = \frac{1}{2}$	$C_3 = \frac{1}{2}$	H^1 半范数 + H^2 半范数	(c)
$C_1 = 1$	$C_2 = 0$	$C_3 = 0$	H^0 范数 (L^2 范数)	(d)
$C_1 = 0$	$C_2 = 1$	$C_3 = 0$	H^1 半范数	(e)
$C_1 = 0$	$C_2 = 0$	$C_3 = 1$	H^2 半范数	(f)

表 2-6 展示了这个数值实验的结果, 其中包括每种算法对应的函数值探测次数、获得的极小点和最佳函数值. 此外, 图 2-7 是求 Rosenbrock 函数极小值的迭代图 (随迭代进行所获取的当前最小函数值), 我们展示了前 16 步探测的结果. 可以看出, 在本例中, 在 (初始) 信赖域半径小的情况下, 使用最小 H^2 范数更新二次模型的算法具有优势.

表 2-6 例 2.5 的数值实验结果

Table 2-6 Results of the numerical experiment of Example 2.5

编号	函数值	解
(a)	0.0031	$(1.0495, 1.1040)^{\mathrm{T}}$
(b)	0.0169	$(1.0427, 1.0996)^{\mathrm{T}}$
(c)	0.0203	$(1.0418, 1.0990)^{\mathrm{T}}$
(d)	0.0078	$(1.0455, 1.1008)^{\mathrm{T}}$
(e)	0.0169	$(1.0427, 1.0996)^{\mathrm{T}}$
(f)	0.0147	$(1.0426, 1.0984)^{\mathrm{T}}$

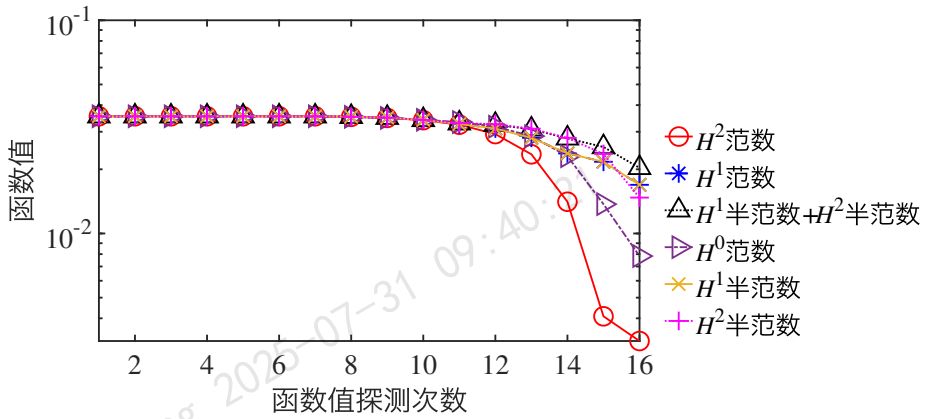


图 2-7 极小化 Rosenbrock 函数

Figure 2-7 Minimizing Rosenbrock function

表 2-7 图 2-8 对应的测试问题

Table 2-7 Test problems for Figure 2-8

ARGLINA	ARGLINA4	ARGLINB	ARGLINC	ARGTRIG
ARWHEAD	BDQRTIC	BDQRTICP	BDVALUE	BROWNAL
BROYDN3D	BROYDN7D	BRYBND	CHAINWOO	CHNROSNB
CHPOWELLB	CHROSEN	CRAGGLVY	CUBE	DQRTIC
EDENSCH	ENGVAL1	ERRINROS	EXPSUM	EXTROSNB
EXTTET	FIROSE	FLETCBV2	FLETCBV3	FLETCHCR
FREUROTH	GENBROWN	GENROSE	INDEF	INTEGREQ
LIARWHD	LILIFUN3	LILIFUN4	MOREBV	MOREBVL
NONDIA	PENALTY1	PENALTY2	PENALTY3	PENALTY3P
ROSENBROCK	SBRYBND	SBRYBNDL	SEROSE	SINQUAD
SROSENBR	STMOD	TOINTTRIG	TQUARTIC	TRIGSABS
TRIGSSQS	TRIROSE1	TRIROSE2	VARDIM	WOODS

中国科学院数学与系统科学研究院 XiePengCheng 2025-07-31 09:40:32

因此, 这里可以体现出在最小加权 H^2 范数更新二次模型函数中使用中心 KKT 矩阵对应的权重系数的优势.

为了展示基于最小加权 H^2 范数更新二次模型函数的算法的更多数值表现, 我们尝试求解一些经典的测试问题, 并使用 Performance Profile 来展示数值结果. 表 2-7 展示了 Performance Profile 对应的测试问题，这些问题的维数在 2 到 200 之间, 它们选自经典和常见的无约束优化测试函数集 [92, 174, 177, 180, 181, 183–186].

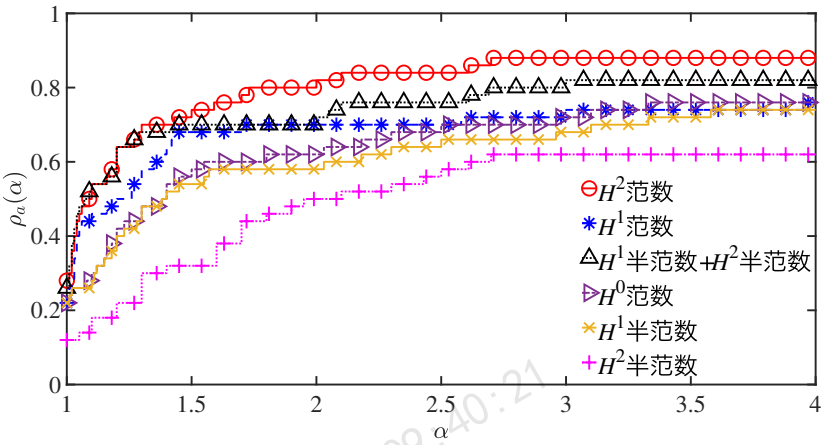


图 2-8 使用不同算法求解测试问题: Performance Profile

Figure 2-8 Solving test problems with different algorithms: Performance Profile

每次迭代中使用 $2n+1$ 个插值点. 此外, 每个算法都从问题集给定的对应初始点 $\boldsymbol{x}_{\text{int}}$ 开始, Performance Profile 中的精度 τ 被设置为 1%. 初始插值集是 $\{\boldsymbol{x}_{\text{int}}, \boldsymbol{x}_{\text{int}} \pm \boldsymbol{e}_j, i = 1, \cdots, n\}$. 信赖域半径和模型梯度范数的容差分别设置为 10^{-8}. 它们共同的初始信赖域半径是 1. 算法 3 中的参数是 $\gamma = 2, \hat{\eta}_1 = \frac{1}{4}, \hat{\eta}_2 = \frac{3}{4}, \mu = 0.1$.

我们可以从图 2-8 中观察到: 在列出的基于最小加权 H^2 范数更新二次模型的无导数算法中, 当 α 大约大于 1.5 时, 基于最小 H^2 范数更新二次模型函数的信赖域算法, 其权重系数为 $C_1 = C_2 = C_3 = \frac{1}{3}$, 可以求解超过 70% 的问题, 表现优于其他算法. 这展示了这组权重系数的优势.

2.2.5 小结

本节主要讨论了如何评估一组加权 H^2 范数的权重系数以及如何找到其最佳选择. 我们讨论了出现在获取插值模型子问题的目标函数中的权重系数, 最小化这些目标函数可以得到相应的二次模型函数. 我们定义了 KKT 矩阵距离、KKT 矩阵误差和权重系数区域的重心. 之后计算了在信赖域半径趋于 0 时最小加权 H^2 范数更新二次模型的权重系数区域 C 的重心. 我们给出了极小化 Rosenbrock 函数的相关数值实验. 还使用 Performance Profile 给出了不同模型对应的算法的数值性能的相关对比. 在将来的工作中, 我们可以考虑从其他角度比较最小加权 H^2 范数更新二次模型的权重系数, 进而探究无导数优化中欠定插值模型的更多性质.

表 2-8 所提出的欠定二次模型 Q_k 的子问题

Table 2-8 The subproblem for the proposed under-determined quadratic model Q_k

子问题	$\min\limits_{Q\in\mathcal{Q}} \|\nabla^2 Q-\nabla^2 Q_{k-1}\|_F^2+\alpha_k\|\nabla Q(\boldsymbol{x}_k)\|_2^2+\beta_k\|(\boldsymbol{I}-\boldsymbol{P}_k)\nabla Q(\boldsymbol{x}_k)\|_2^2$ s. t. $Q(\boldsymbol{y}_i)=f(\boldsymbol{y}_i),\ \forall\ \boldsymbol{y}_i\in\mathcal{X}_k$
参数	$\rho_{k-1}=\frac{f(\boldsymbol{x}_k)-f(\boldsymbol{x}_{k-1})}{Q_{k-1}(\boldsymbol{x}_k)-Q_{k-1}(\boldsymbol{x}_{k-1})},\ (\boldsymbol{x}_k\neq\boldsymbol{x}_{k-1})$ $\mathbb{1}$: 特征函数 ($\mathbb{1}([\text{真}])=1, \mathbb{1}([\text{假}])=0$) $\eta_0\geq 0$: 预给的算法超参数 $\alpha_k=\mathbb{1}\{0<\|\boldsymbol{x}_k-\boldsymbol{x}_{k-1}\|_2<\Delta_{k-1}\}\mathbb{1}\{\rho_{k-1}>\eta_0\}$ $\beta_k=\mathbb{1}\{\|\boldsymbol{x}_k-\boldsymbol{x}_{k-1}\|_2=\Delta_{k-1}\}\mathbb{1}\{\rho_{k-1}>\eta_0\}$ $\boldsymbol{I}$: 单位矩阵; Δ_{k-1}: 第 $k-1$ 次信赖域半径 $\boldsymbol{P}_k=\frac{(\boldsymbol{x}_k-\boldsymbol{x}_{k-1})(\boldsymbol{x}_k-\boldsymbol{x}_{k-1})^\top}{\|\boldsymbol{x}_k-\boldsymbol{x}_{k-1}\|_2^2}$ 将向量从 $\mathfrak{R}^n$ 投影到 $\text{span}\{\boldsymbol{x}_k-\boldsymbol{x}_{k-1}\}$ 上
模型	$Q(\boldsymbol{x}):=\frac{1}{2}(\boldsymbol{x}-\boldsymbol{x}_k)^\top\nabla^2 Q(\boldsymbol{x}-\boldsymbol{x}_k)+\nabla Q(\boldsymbol{x}_k)^\top(\boldsymbol{x}-\boldsymbol{x}_k)+c$ c: 二次函数 Q 的常数项; $\mathcal{Q}$: 二次函数集
插值	$\mathcal{X}_k=\{\boldsymbol{y}_1,\cdots,\boldsymbol{y}_m\}\subset\mathfrak{R}^n$: 第 k 步的插值点集 $\mathcal{X}_k=\mathcal{X}_{k-1}\cup\{\boldsymbol{x}_k\}\backslash\{\boldsymbol{x}_t^{(k)}\}$ (在成功步) ; $\boldsymbol{x}_t^{(k)}$: 被舍弃的点
$\boldsymbol{x}_k$	$\boldsymbol{x}_k\in\{\arg\min\limits_{\boldsymbol{x}} Q_{k-1}(\boldsymbol{x}),\ \text{s. t. }\boldsymbol{x}\in\mathcal{B}_{\Delta_{k-1}}(\boldsymbol{x}_{k-1})\}$ (在成功步) $\mathcal{B}_{\Delta_{k-1}}(\boldsymbol{x}_{k-1})=\{\boldsymbol{x},\|\boldsymbol{x}-\boldsymbol{x}_{k-1}\|_2\leq\Delta_{k-1}\}$

2.3　使用新的欠定二次插值模型的无导数方法

2.3.1　背景和动机

Conn 和 Toint [171] 为基于模型的无导数信赖域算法提出了最小范数类型的欠定二次插值模型, 它是关于二次函数 Q 的子问题

$$\begin{aligned}&\min_{Q\in\mathcal{Q}} \|\nabla^2 Q\|_F^2+\left\|\nabla Q(\boldsymbol{x}_k)\right\|_2^2\\&\text{s. t. } Q(\boldsymbol{y}_i)=f(\boldsymbol{y}_i),\ \forall\ \boldsymbol{y}_i\in\mathcal{X}_k\end{aligned}\tag{2-33}$$

的解. Conn 和 Toint 在数值上展现了该模型的优势. 继他们的工作之后, 学者们提出了更多的欠定二次插值模型 [93, 97, 153, 154]. 目前, 迭代获取欠定二次模型 Q_k 的常见方法是求解一个相应的最小范数 (变化/更新) 约束优化问题, 例如 Powell [93] 提出的模型是通过求解子问题

$$\begin{aligned}&\min_{Q\in\mathcal{Q}} \left\|\nabla^2 Q-\nabla^2 Q_{k-1}\right\|_F^2\\&\text{s. t. } Q(\boldsymbol{y}_i)=f(\boldsymbol{y}_i),\ \forall\ \boldsymbol{y}_i\in\mathcal{X}_k\end{aligned}\tag{2-34}$$

来获得 (如前文所述). 如果 (2-34) 中 $\nabla^2 Q_{k-1}$ 被零矩阵替换, 那么它就是 Conn、Scheinberg 和 Toint [156] 提出的最小范数二次模型, 该模型也有良好的数值性能, 故而在实际中被广泛使用.

如上所述, 构造模型函数并通过求解模型函数的信赖域子问题获得下一个迭代点是基于模型的无导数信赖域方法的两个主要步骤. 关于为无导数方法构造模型的大多数工作没有考虑前一个信赖域子问题及其数值解 (在大多数情况下是当前迭代点). 本节考虑根据前一个模型在前一个信赖域中给出的迭代点的位置和迭代的成功情况来构建二次模型. 提出该想法的主要动机是尽可能让前一个模型和其对应的信赖域子问题的数值解为当前模型提供更多有用的信息. 鉴于构造无导数优化模型的最终目标是使通过求解每个相应的信赖域子问题给出的下一个迭代点接近同一信赖域中的真实极小点, 我们可以利用当前迭代点的位置和迭代的成功信息来指导构造模型函数这一过程.

本节试图给出理解和分析 Conn 和 Toint 模型的新视角, 并试图提出一个新模型 (在表 2-8 中给出了一个粗略的展示, 图示见图 2-9, 其中 $\boldsymbol{x} \in \mathfrak{R}^n$, $\boldsymbol{P}_k \in \mathfrak{R}^{n\times n}$, 我们把向量写为列向量) , 模型的公式将基于对应子问题的 KKT 条件给出.

注 2.6. 若对于预给定的算法超参数 $\eta_0 \geq 0$ 有 $\rho_{k-1} > \eta_0$, 则称算法获得成功步; 否则 (包括 $\boldsymbol{x}_k = \boldsymbol{x}_{k-1}$ 的情况), 我们称其获得了不成功步.

据作者所知, 本节的工作是第一个考虑通过利用信赖域迭代性质来构造欠定插值模型的工作, 简言之, 我们的创新点是通过考虑前一步中的二次模型在前一步所生成的迭代点的位置和迭代成功情况来构造当前步所需要的二次模型.

本节剩余部分的内容组织如下. 第 2.3.2 节中讨论了如何利用信赖域迭代的性质对 Conn 和 Toint 模型进行理论分析并给出改进. 第 2.3.3 节分析了我们模型的相应子问题的严格凸性, 并基于 KKT 条件给出了获取我们新模型的计算公式. 第 2.3.4 节给出了数值结果. 最后, 我们给出了小结.

2.3.2 考虑前一信赖域迭代性质的模型

对于基于二次插值模型的无导数信赖域优化方法, 确保信赖域内二次模型的极小点足够接近同一信赖域内目标函数 f 的极小点是非常重要的. 这是因为基于模型的无导数信赖域优化方法中二次插值模型函数的作用是提供一个具有更小函数值的新迭代点, 该点通过在当前信赖域内极小化当前模型函数获得. 事实上, 基于模型的信赖域方法试图迭代地使用信赖域内一个好的模型函数的极小点来替代同一信赖域内黑箱目标函数 f 的极小点.

因此, 我们通过信赖域迭代和新获取的迭代点的最优性去分析 Conn 和 Toint 的模型, 这帮助我们更好地理解了 Conn 和 Toint 的模型并推导出了一个改进后的模型. 让我们通过以下引理来进一步观察, 该引理考虑了在 1 维情况下传统函数值插值 (1-3) 的缺点和风险, 由 Robinson [187] 在 1979 年给出.

引理 2.23 (二次插值的极小点风险)**.** 设 $x^* \in \mathfrak{R}$ 且 $\varepsilon > 0$. 假设 f 是从区间 $\left[x^* - \hat{\varepsilon}, x^* + \hat{\varepsilon}\right]$ 到 $\mathfrak{R}$ 的连续单峰函数, 具有极小点 x^*, 则除非 f 在 $[x^* - \hat{\varepsilon}, x^* + \hat{\varepsilon}]$ 上与某个二次函数一致, 否则存在区间 $\left[x^* - \hat{\varepsilon}, x^* + \hat{\varepsilon}\right]$ 内的点 $x_0 < x_1 < x_2$, 且 $x_1 \neq x^*$, 使得在这三个点处插值 f 的二次函数 Q 的唯一极小点是 x_1.

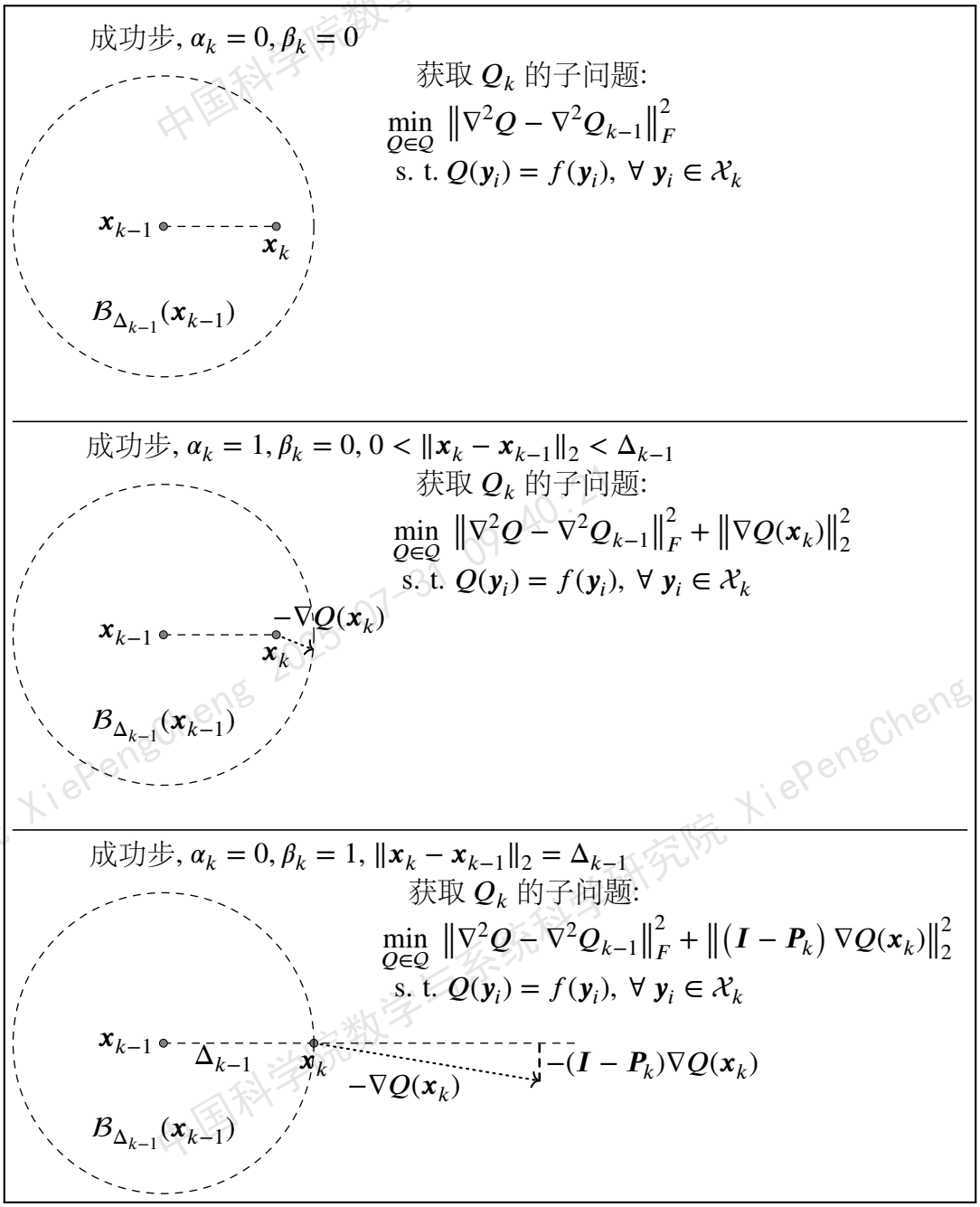


图 2-9 子问题刻画

Figure 2-9 Illustration of the subproblem

上述引理说明了一个二次插值模型函数 (更多细节可以参考 Conn、Scheinberg 和 Vicente 著作 [20] 中的定义 6.2) 可能无法提供准确的极小点. 此外, 我们甚至可以通过选择插值点来使二次插值模型的极小点落在完全错误的位置上. 这揭示了函数值约束 (1-3) 可能无法直接表征被逼近函数的最优性或极小点.

张在坤 [154] 建立了 Conn 和 Toint 模型与最小 H^1 半范数二次模型之间的联系. 在那之后, 罕有其他直接分析 Conn 和 Toint 最早所提出的逼近模型的工作. 据我们目前所知, 对基于模型的无导数优化的欠定插值模型的分析很少涉及到信赖域迭代的性质. 这里将试图通过信赖域迭代的性质来分析 Conn 和 Toint 的模型. 我们将通过在算法的迭代成功步中选择性地将欠定二次模型视为二次模型或线性模型来提出一个新模型, 如前文所述, 这可以被视为是模型和迭代性质的一种结合.

注 2.7 (无导数拟牛顿方法中模型的最优性). Greenstadt [89] 提出的无导数拟牛顿方法考虑了迭代点处模型的最优性. 然而, 在无导数信赖域优化算法中, 迭代点处模型的最优性并没有被直接用于构造模型, 这促使了我们在试图获得欠定二次插值模型函数时考虑已知迭代点处模型的最优性和梯度下降性质.

实际上, 在使用算法求解时的第 k 步, 如果这一步是成功的, 那么迭代点和插值点 $\boldsymbol{x}_k$ 其实是特殊的, 这是因为该点是通过数值求解信赖域优化子问题得到的点, 而不是单纯用于插值的普通样本点. 我们知道, 每个新添加的迭代/插值点 $\boldsymbol{x}_k$ 是第 $k-1$ 个模型 Q_{k-1} 在第 $k-1$ 个信赖域内的极小点. 传统的信赖域方法设计并使用一个子程序来求解信赖域子问题

$$\begin{aligned} &\min_{\boldsymbol{d}} Q_{k-1}(\boldsymbol{x}_{k-1}+\boldsymbol{d}) \\ &\text{s. t. } \|\boldsymbol{d}\|_2 \leq \Delta_{k-1} \end{aligned} \tag{2-35}$$

以获得新的迭代点 $\boldsymbol{x}_k = \boldsymbol{x}_{k-1} + \boldsymbol{d}_{k-1}$, 其中 $\boldsymbol{d}_{k-1}$ 是问题 (2-35) 的解. 实际上, 我们有理由信任并使用上一步获得的模型函数提供的信息. Powell 的最小范数更新也有类似的思路. 换句话说, 改进前一个模型的前提是相信前一步的模型 (尤其是当它提供了成功步时) 是好的.

我们知道, 目标函数中的 $\|\nabla^2 Q\|_F^2$ 一项出现在 (2-33) 的目标函数中, 是为了填补欠定二次模型剩余系数的自由度, 这些自由度没有被函数值约束 (1-3) 填补. 然而, 在 Conn 和 Toint 模型对应的子问题中, $\left\|\nabla Q(\boldsymbol{x}_k)\right\|_2^2$ 这一项看起来似乎对构造模型来说不是必需的. 事实上, 从信赖域迭代和模型最优性的角度来看, 它是有助于模型的, 我们在下面给出原因.

关于信赖域子问题的解, 我们回顾以下经典结果[6].

命题 2.24 (信赖域子问题的解). *对于一个二次函数 Q, 向量 $\boldsymbol{z} \in \mathfrak{R}^n$ 满足*

$$\boldsymbol{z} \in \left\{ \arg\min_{\boldsymbol{x}} \ Q(\boldsymbol{x}),\ \text{s. t. } \boldsymbol{x} \in \mathcal{B}_{\Delta_{k-1}}(\boldsymbol{x}_{k-1}) \right\}, \tag{2-36}$$

[6]更多细节可以在经典数值优化教科书中看到, 例如 Nocedal 和 Wright 著作 [3] 中的定理 4.1.

当且仅当 $\|\boldsymbol{z}-\boldsymbol{x}_{k-1}\|_2 \leq \Delta_{k-1}$ 并且存在 $\omega \geq 0$ 使得 $\boldsymbol{z}$ 满足条件

$$\left(\nabla^2 Q+\omega \boldsymbol{I}\right)\left(\boldsymbol{z}-\boldsymbol{x}_{k-1}\right)+\nabla Q\left(\boldsymbol{x}_{k-1}\right)=\boldsymbol{0}_n, \tag{2-37}$$

并且

$$\begin{aligned} \omega\left(\Delta_{k-1}-\|\boldsymbol{z}-\boldsymbol{x}_{k-1}\|_2\right)&=0, \\ \nabla^2 Q+\omega \boldsymbol{I} &\succeq \boldsymbol{0}_{nn}, \end{aligned} \tag{2-38}$$

其中矩阵 $\boldsymbol{A} \succeq \boldsymbol{0}$ 表示矩阵 $\boldsymbol{A}$ 是半正定的.

此外, 我们有

$$\nabla Q\left(\boldsymbol{x}_k\right)=\nabla^2 Q\left(\boldsymbol{x}_k-\boldsymbol{x}_{k-1}\right)+\nabla Q\left(\boldsymbol{x}_{k-1}\right).$$

因此, Conn 和 Toint 模型的对应子问题 (2-33) 可以被重写为子问题

$$\begin{aligned} &\min_{Q \in \mathcal{Q}} \left\|\nabla^2 Q\right\|_F^2+\left\|\nabla^2 Q\left(\boldsymbol{x}_k-\boldsymbol{x}_{k-1}\right)+\nabla Q\left(\boldsymbol{x}_{k-1}\right)\right\|_2^2 \\ &\text{s. t. } Q\left(\boldsymbol{y}_i\right)=f\left(\boldsymbol{y}_i\right), \forall\, \boldsymbol{y}_i \in \mathcal{X}_k, \end{aligned} \tag{2-39}$$

其中目标函数中的第二项在 $\boldsymbol{x}_k$ 位于 $\mathcal{B}_{\Delta_{k-1}}\left(\boldsymbol{x}_{k-1}\right)$ 的内部 (即 $\left\|\boldsymbol{x}_k-\boldsymbol{x}_{k-1}\right\|_2<\Delta_{k-1}$) 时迫使 $\boldsymbol{x}_k$ 靠近二次模型函数 Q_k 的稳定点, 这是因为在这种情况下$\omega=0$. 在这种情况下我们提前将 $\nabla^2 Q_k$ 视为半正定矩阵, 事实上, 求解子问题 (2-39) 有实现上述目标的趋势, 这是因为 $\boldsymbol{x}_k$ 在 $\mathcal{X}_k$ 中具有较小或最小的函数值, 并且 $\|\boldsymbol{x}_k-\boldsymbol{x}_{k-1}\|_2<\Delta_{k-1}$.

我们将子问题 (2-33) 写为子问题 (2-39) 以使其与 (2-41) 一致. 分析 $\|\boldsymbol{x}_k-\boldsymbol{x}_{k-1}\|_2<\Delta_{k-1}$ 的情况下成功的 $\boldsymbol{x}_k$ 帮助我们建立了对 Conn 和 Toint 模型的新见解. 在下文中, 我们考虑 $\|\boldsymbol{x}_k-\boldsymbol{x}_{k-1}\|_2=\Delta_{k-1}$ 的情况, 这帮助我们推导出了我们的模型.

基于上述讨论, 我们试图给出一个新模型. 根据上述讨论, 若 $0<\|\boldsymbol{x}_k-\boldsymbol{x}_{k-1}\|_2<\Delta_{k-1}$, 且 $\boldsymbol{x}_k$ 是算法找到的一个成功步, 即

$$\rho_{k-1}=\frac{f\left(\boldsymbol{x}_k\right)-f\left(\boldsymbol{x}_{k-1}\right)}{Q_{k-1}\left(\boldsymbol{x}_k\right)-Q_{k-1}\left(\boldsymbol{x}_{k-1}\right)}>\eta_0 \geq 0,$$

则认为欠定二次模型 Q_{k-1} 的二阶和一阶导数信息为获取 $\boldsymbol{x}_k$ 提供了指导信息是合理的. 根据前面的讨论, 若 $\nabla^2 Q_k$ 是正定的, 则正则项 $\left\|\nabla Q\left(\boldsymbol{x}_k\right)\right\|_2^2$ 会使得 Q_k 在信赖域 $\mathcal{B}_{\Delta_{k-1}}\left(\boldsymbol{x}_{k-1}\right)$ 内的极小点靠近 $\boldsymbol{x}_k$. 在这种情况下, Q_k 应该继承二次模型 Q_{k-1} 的二阶性质.

然而, 在 $\boldsymbol{x}_k$ 是一个成功步且 $\left\|\boldsymbol{x}_k-\boldsymbol{x}_{k-1}\right\|_2=\Delta_{k-1}$ 的情况下, 根据 KKT 条件, 如果算法仍然使用 Conn 和 Toint 的模型, 如我们前面讨论的, $\|\nabla Q\left(\boldsymbol{x}_k\right)\|_2^2$ 这一项意味着将 $\boldsymbol{x}_k$ 视为目标函数 f 的一个良好的近似稳定点, 这其实在某种意义下误用了 Q_{k-1} 和 $\boldsymbol{x}_k$ 提供的信息, 因为在这种情况下 $\boldsymbol{x}_k$ 可能不接近 f 的稳定点.

在这种情况下, 根据命题 2.24, 若算法仍然希望第 k 个模型遵循第 $k-1$ 个模型的二次性质, 并试图让 Q_k 在信赖域 $\mathcal{B}_{\Delta_{k-1}}(\boldsymbol{x}_{k-1})$ 内的极小点靠近 $\boldsymbol{x}_k$, 则正则项可以是 (2-37) 左侧的 ℓ_2 范数的平方, 即

$$\left\|\left(\nabla^2 Q+\omega \boldsymbol{I}\right)\left(\boldsymbol{x}_k-\boldsymbol{x}_{k-1}\right)+\nabla Q(\boldsymbol{x}_{k-1})\right\|_2^2=\left\|\omega(\boldsymbol{x}_k-\boldsymbol{x}_{k-1})+\nabla Q(\boldsymbol{x}_k)\right\|_2^2, \quad (2\text{-}40)$$

其中 $\omega \geq 0$ 且我们希望 ω 满足 $\nabla^2 Q_k+\omega \boldsymbol{I} \succeq \boldsymbol{0}$, 其中 Q_k 通过求解子问题

$$\begin{aligned} \min_{Q \in \mathcal{Q}} \ & \left\|\nabla^2 Q\right\|_F^2+\left\|\omega(\boldsymbol{x}_k-\boldsymbol{x}_{k-1})+\nabla Q(\boldsymbol{x}_k)\right\|_2^2 \\ \text{s. t. } & Q(\boldsymbol{y}_i)=f(\boldsymbol{y}_i), \ \forall\, \boldsymbol{y}_i \in \mathcal{X}_k \end{aligned} \quad (2\text{-}41)$$

获得. 然而, 参数 ω 的不确定性使得上述想法难以被直接实现, 详细原因如下. 这里需要注意的是我们的目标是在给定 ω 的情况下获得模型 Q_k, 这与获取已知二次函数在信赖域内的极小点的经典场景不同.

一方面, 如果我们在子问题 (2-41) 的目标函数中让 $\omega=0$ 来提供二次插值模型函数, 那么它正好是 Conn 和 Toint 的子问题 (2-33), 也因此对应给出了相应的模型函数, 但是我们已经讨论了这对于 $\boldsymbol{x}_k$ 是一个成功步但 $\|\boldsymbol{x}_k-\boldsymbol{x}_{k-1}\|_2=\Delta_{k-1}$ 的情况不合适.

另一方面, 如果我们考虑一个不一定为零的 ω, 我们很难选择一个合适的 ω 来保证 $\nabla^2 Q_k+\omega \boldsymbol{I}$ 是半正定的, 同时不同的 ω 会在求解子问题 (2-41) 后给出不同的模型函数 Q_k.

此外, 对于我们的欠定二次插值模型, 构造每个模型的插值点数少于 $\frac{1}{2}(n+1)(n+2)$. 因此, 我们不能仅靠插值来获得唯一确定的二次模型.

根据上述分析, ω 的不确定性导致二次模型函数不唯一, 这使得在子问题 (2-41) 的目标函数中添加第二项来捕捉 Q_{k-1} 提供的二次信息, 即便在迭代点 $\boldsymbol{x}_k$ 是成功迭代的情况下也不一定是准确的. 上述情况启发我们提出一个新模型, 即根据 $\|\boldsymbol{x}_k-\boldsymbol{x}_{k-1}\|_2$ 与 Δ_{k-1} 的关系, 选择性地将上一个模型视为线性模型或二次模型[7].

对于 $\|\boldsymbol{x}_k-\boldsymbol{x}_{k-1}\|_2=\Delta_{k-1}$ 且当前步是成功步的情形, 我们认为假设 "$\boldsymbol{x}_k-\boldsymbol{x}_{k-1}$ 的方向仍然提供了最新迭代点 $\boldsymbol{x}_k$ 的近似梯度下降方向" 是合理的. 具体来说, 算法在成功步获得了一个能充分减小函数值的新迭代点. 在这种情况下, 获得的第 $k-1$ 个二次模型被认为提供了相对高精度的梯度下降方向. 此外, 算法 (在这样的成功步中) 应该只获得了一个好的一阶近似.

因此, 我们认为使新模型 Q_k 与 Q_{k-1} 的梯度下降性质或信息尽可能一致是合理且符合实际的. 换句话说, 除了使 Q_k 在满足函数值约束 (1-3) 时最小化 $\|\nabla Q_k(\boldsymbol{x}_k)\|_2$ 外, $\boldsymbol{x}_k-\boldsymbol{x}_{k-1}$ 的方向应该与 $-\nabla Q_k(\boldsymbol{x}_k)$ 的方向接近. 此外, 如果 $Q_k(\boldsymbol{x})$ 在信赖域 $\mathcal{B}_{\Delta_{k-1}}(\boldsymbol{x}_{k-1})$ 内也在 $\boldsymbol{x}_k$ 达到最小值, 那么 $\left(\boldsymbol{I}-\boldsymbol{P}_k\right)\nabla Q(\boldsymbol{x}_k)=\boldsymbol{0}_n$, 其中

[7]我们根据 $\|\boldsymbol{x}_k-\boldsymbol{x}_{k-1}\|_2=\Delta_{k-1}$ 是否成立来选择性地将 Q_{k-1} 提供的 $\boldsymbol{x}_k$ 处的信息和最优性视为由可靠的线性模型或二次模型提供的信息.

$\boldsymbol{P}_k = \frac{(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})^\top}{\|\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\|_2^2}$ 将向量从 $\mathfrak{R}^n$ 投影到 $\operatorname{span}\{\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\}$ 上, $\mathbf{0}_n \in \mathfrak{R}^n$ 是零向量. 基于此, 我们提出可以通过求解子问题

$$\begin{aligned} &\min_{Q \in \mathcal{Q}} \left\|\nabla^2 Q - \nabla^2 Q_{k-1}\right\|_F^2 + \alpha_k \left\|\nabla Q(\boldsymbol{x}_k)\right\|_2^2 + \beta_k \left\|\left(\boldsymbol{I} - \boldsymbol{P}_k\right) \nabla Q(\boldsymbol{x}_k)\right\|_2^2 \\ &\text{s. t. } Q(\boldsymbol{y}_i) = f(\boldsymbol{y}_i),\ \forall\ \boldsymbol{y}_i \in \mathcal{X}_k \end{aligned} \tag{2-42}$$

来得到模型 Q_k, 其中系数 α_k 和 β_k 在表 2-8 中定义, 即

$$\alpha_k = \begin{cases} 1, \text{ 如果 } 0 < \|\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\|_2 < \Delta_{k-1} \text{ 且 } \rho_{k-1} > \eta_0, \\ 0, \text{ 否则}, \end{cases}$$

$$\beta_k = \begin{cases} 1, \text{ 如果 } \|\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\|_2 = \Delta_{k-1} \text{ 且 } \rho_{k-1} > \eta_0, \\ 0, \text{ 否则}, \end{cases}$$

$|\mathcal{X}_k| < \frac{1}{2}(n+1)(n+2)$ 且

$$\rho_{k-1} = \frac{f(\boldsymbol{x}_k) - f(\boldsymbol{x}_{k-1})}{Q_{k-1}(\boldsymbol{x}_k) - Q_{k-1}(\boldsymbol{x}_{k-1})}.$$

请注意, 我们在子问题 (2-42) 的目标函数中保留了 $\left\|\nabla^2 Q - \nabla^2 Q_{k-1}\right\|_F^2$ 这一项, 以此来继承 Powell 最小范数更新方法的一些优点, 这样处理是考虑到插值模型 Q_{k-1} 的历史信息仍然是有用的. 当 $\boldsymbol{x}_k$ 对应不成功步时, 该子问题将导出 Powell 的 Frobenius 范数更新欠定模型, 这是因为在这种情况下 $\alpha_k = 0$ 且 $\beta_k = 0$.

注 2.8. 在 $\|\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\|_2 = \Delta_{k-1}$ 且 $\omega = \frac{\|\boldsymbol{P}_k \nabla Q(\boldsymbol{x}_k)\|_2}{\|\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\|_2}$ 时, 极小化子问题 (2-42) 中的目标函数将减小等式 (2-37) 左侧的 ℓ_2 范数值, 原因是若 $\omega = \frac{\|\boldsymbol{P}_k \nabla Q(\boldsymbol{x}_k)\|_2}{\|\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\|_2}$, 则有

$$\left\|\left(\nabla^2 Q + \omega \boldsymbol{I}\right)\left(\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\right) + \nabla Q(\boldsymbol{x}_{k-1})\right\|_2^2 = \left\|(\boldsymbol{I} - \boldsymbol{P}_k)\nabla Q(\boldsymbol{x}_k)\right\|_2^2.$$

上述分析表明, 在上述情况下, 极小化 (2-41) 等价于极小化 (2-42) 来获得我们的模型 $\boldsymbol{Q}_k$.

我们应该注意, 仅通过基于 (1-3) 的函数值插值或传统的最小范数方案获得的二次模型插值, 可能也已经在某种意义上体现了二次函数的曲率和形状. 然而, 在不增加插值点的数量时, 考虑 $\boldsymbol{x}_k$ 在第 k 步的最优性在一定意义上会获得比不考虑最优性的情况更好的逼近, 第 2.3.4 节的数值结果展示了我们模型的优势.

2.3.3　子问题的凸性和模型的计算公式

接下来, 我们分析子问题 (2-42) 中的严格凸性来说明子问题 (2-42) 的解具有唯一性.

定理 2.25 (子问题目标函数的严格凸性). *给定 $\alpha_k \geq 0, \beta_k \geq 0, \boldsymbol{x}_k \in \mathfrak{R}^n, \boldsymbol{I} \in \mathfrak{R}^{n \times n}, \boldsymbol{P}_k \in \mathfrak{R}^{n \times n}$, 假定集合 $\mathcal{X}_k$ 不包含在维数小于 n 的子空间中. 对于所有满足 (2-42) 中插值条件的二次函数, 目标函数*

$$\left\|\nabla^2 Q - \nabla^2 Q_{k-1}\right\|_F^2 + \alpha_k \left\|\nabla Q(\boldsymbol{x}_k)\right\|_2^2 + \beta_k \left\|\left(\boldsymbol{I} - \boldsymbol{P}_k\right) \nabla Q(\boldsymbol{x}_k)\right\|_2^2$$

作为二次函数 Q 的函数是严格凸的.

证明. 用 $\mathcal{F}(Q)$ 表示目标函数. 我们需要证明对于 $0<\varphi<1$ 和满足 (2-42) 中约束的二次函数 $Q_a \neq Q_b$, 有

$$\mathcal{F}(\varphi Q_a + (1-\varphi)Q_b) < \varphi\mathcal{F}(Q_a) + (1-\varphi)\mathcal{F}(Q_b). \tag{2-43}$$

定理的假设意味着 $\nabla^2 Q_a \neq \nabla^2 Q_b$. 因此可得

$$\begin{aligned}
&\mathcal{F}(\varphi Q_a + (1-\varphi)Q_b) - \left(\varphi\mathcal{F}(Q_a) + (1-\varphi)\mathcal{F}(Q_b)\right) \\
&= \left(\varphi^2-\varphi\right)\Big[\left\|\nabla^2 Q_a - \nabla^2 Q_b\right\|_F^2 + \alpha_k\left\|\left(\nabla Q_a(\boldsymbol{x}_k) - \nabla Q_b(\boldsymbol{x}_k)\right)\right\|_2^2 \\
&\quad + \beta_k\left\|\left(\boldsymbol{I}-\boldsymbol{P}_k\right)\left(\nabla Q_a(\boldsymbol{x}_k) - \nabla Q_b(\boldsymbol{x}_k)\right)\right\|_2^2\Big] < 0.
\end{aligned}$$

故 (2-43) 成立, 严格凸性得证. □

我们随后得到子问题 (2-42) 的严格凸性和子问题 (2-42) 解的唯一性, 这保证了我们的欠定插值模型在算法的每一步都能被唯一确定.

为了得到获取模型的计算公式, 我们基于 KKT 条件给出以下定理.

定理 2.26 (二次模型的计算公式)**.** *二次函数*

$$Q(\boldsymbol{x}) = \frac{1}{2}\left(\boldsymbol{x}-\boldsymbol{x}_k\right)^{\top}\boldsymbol{H}\left(\boldsymbol{x}-\boldsymbol{x}_k\right) + \boldsymbol{g}^{\top}\left(\boldsymbol{x}-\boldsymbol{x}_k\right) + c,$$

其中

$$\boldsymbol{H} = \nabla^2 Q_{k-1} + \frac{1}{4}\sum_{j=1}^{m}\lambda_j\left(\boldsymbol{y}_j-\boldsymbol{x}_k\right)\left(\boldsymbol{y}_j-\boldsymbol{x}_k\right)^{\top},$$

$\boldsymbol{\lambda} = (\lambda_1, \cdots, \lambda_n)^{\top} \in \mathfrak{R}^n$, *且* $(\boldsymbol{\lambda}, c, \boldsymbol{g})^{\top} \in \mathfrak{R}^{m+1+n}$ *是 KKT 方程*

$$\begin{pmatrix} \boldsymbol{A} & \boldsymbol{E} & \boldsymbol{X} \\ \boldsymbol{E}^{\top} & 0 & \boldsymbol{0}_n^{\top} \\ \boldsymbol{X}^{\top} & \boldsymbol{0}_n & \boldsymbol{B} \end{pmatrix}\begin{pmatrix} \boldsymbol{\lambda} \\ c \\ \boldsymbol{g} \end{pmatrix} = \begin{pmatrix} \boldsymbol{r} \\ 0 \\ \boldsymbol{0}_n \end{pmatrix} \tag{2-44}$$

的解, 是子问题 (2-42) 的解, 其中

$$\begin{aligned}
\boldsymbol{r} &= \begin{pmatrix} f(\boldsymbol{y}_1) - \frac{1}{2}\left(\boldsymbol{y}_1-\boldsymbol{x}_k\right)^{\top}\nabla^2 Q_{k-1}\left(\boldsymbol{y}_1-\boldsymbol{x}_k\right) \\ \vdots \\ f(\boldsymbol{y}_m) - \frac{1}{2}\left(\boldsymbol{y}_m-\boldsymbol{x}_k\right)^{\top}\nabla^2 Q_{k-1}\left(\boldsymbol{y}_m-\boldsymbol{x}_k\right) \end{pmatrix}, \\
\boldsymbol{X} &= \left(\boldsymbol{y}_1-\boldsymbol{x}_k, \cdots, \boldsymbol{y}_m-\boldsymbol{x}_k\right)^{\top}, \\
\boldsymbol{B} &= -2\alpha_k\boldsymbol{I} - 2\beta_k(\boldsymbol{I}-\boldsymbol{P}_k)^{\top}(\boldsymbol{I}-\boldsymbol{P}_k),
\end{aligned}$$

矩阵 $\boldsymbol{A}$ 的元素是

$$\boldsymbol{A}_{ij} = \frac{1}{8}\left[\left(\boldsymbol{y}_i-\boldsymbol{x}_k\right)^{\top}\left(\boldsymbol{y}_j-\boldsymbol{x}_k\right)\right]^2,\ 1 \le i, j \le m,$$

此外, $\boldsymbol{E}$ 是元素均为 1 的向量.

证明. 子问题 (2-42) 对应的 Lagrange 函数是

$$\mathcal{L}(c, \boldsymbol{g}, \boldsymbol{H}) = \left\|\boldsymbol{H} - \nabla^2 Q_{k-1}\right\|_F^2 + \alpha_k \left\|\boldsymbol{g}\right\|_2^2 + \beta_k \left\|\left(\boldsymbol{I} - \boldsymbol{P}_k\right) \boldsymbol{g}\right\|_2^2 - \sum_{j=1}^{m} \lambda_j \left[\frac{1}{2}\left(\boldsymbol{y}_j - \boldsymbol{x}_k\right)^{\top} \boldsymbol{G}\left(\boldsymbol{y}_j - \boldsymbol{x}_k\right) + \boldsymbol{g}^{\top}\left(\boldsymbol{y}_j - \boldsymbol{x}_k\right) + c\right].$$

根据 KKT 条件, 我们有

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial c} &= \sum_{j=1}^{m} \lambda_j = 0, \\
\frac{\partial \mathcal{L}}{\partial \boldsymbol{g}} &= 2\alpha_k \boldsymbol{g} + 2\beta_k \left(\boldsymbol{I} - \boldsymbol{P}_k\right)^{\top}\left(\boldsymbol{I} - \boldsymbol{P}_k\right) \boldsymbol{g} - \sum_{j=1}^{m} \lambda_j \left(\boldsymbol{y}_j - \boldsymbol{x}_k\right) = \boldsymbol{0}_n, \\
\frac{\partial \mathcal{L}}{\partial \boldsymbol{H}} &= 2\boldsymbol{H} - 2\nabla^2 Q_{k-1} - \frac{1}{2} \sum_{j=1}^{m} \lambda_j \left(\boldsymbol{y}_j - \boldsymbol{x}_k\right)\left(\boldsymbol{y}_j - \boldsymbol{x}_k\right)^{\top} = \boldsymbol{0}_{nn},
\end{aligned}$$

和

$$f(\boldsymbol{y}_i) = c + \boldsymbol{g}^{\top}\left(\boldsymbol{y}_i - \boldsymbol{x}_k\right) + \frac{1}{2}\left(\boldsymbol{y}_i - \boldsymbol{x}_k\right)^{\top} \boldsymbol{H}\left(\boldsymbol{y}_i - \boldsymbol{x}_k\right), \ i = 1, \cdots, m.$$

从上述关系中, 我们可以得到 KKT 方程 (2-44), 定理得证. □

对于我们上述讨论的模型子问题, 我们只需更改 KKT 方程即可获取相应二次模型函数的公式, 这是易于实现的.

2.3.4　数值结果

本节给出数值结果, 包括求解数值例子的结果和求解一组测试问题的 Performance Profile 和 Data Profile.

以下例子将说明我们本节提出的模型和方法具有优势.

例 2.6 (迭代初始表现)**.** 我们给出一个无约束优化问题作为例子. 目标函数是 2 维 Rosenbrock 函数

$$f(\boldsymbol{x}) = \left(1 - x_1\right)^2 + 100\left(x_2 - x_1^2\right)^2,$$

最小值为 0, 初始信赖域半径为 $\Delta_0 = 1$.

步 1. 初始插值点 $\boldsymbol{y}_1, \boldsymbol{y}_2, \boldsymbol{y}_3$ 分别为

$$\boldsymbol{y}_1 = \begin{pmatrix} 0 \\ 7 \end{pmatrix}, \ \boldsymbol{y}_2 = \begin{pmatrix} 1 \\ 7 \end{pmatrix}, \ \boldsymbol{y}_3 = \begin{pmatrix} 0 \\ 8 \end{pmatrix}.$$

步 2. 通过求解

$$\begin{aligned}
&\min_{Q \in \mathcal{Q}} \ \left\|\nabla^2 Q\right\|_F^2 \\
&\text{s. t. } Q(\boldsymbol{y}_i) = f(\boldsymbol{y}_i), \ \forall\, i = 1, 2, 3
\end{aligned}$$

获取 Q_0.

步 3. 通过求解

$$\min_{\boldsymbol{d}} Q_0(\boldsymbol{y}_2+\boldsymbol{d})$$
$$\text{s. t. } \|\boldsymbol{d}\|_2 \le \Delta_0$$

获取 $\boldsymbol{d}^*$, 然后令 $\boldsymbol{y}_4=\boldsymbol{y}_2+\boldsymbol{d}^*$, 若这一步成功, 则用 $\boldsymbol{y}_4$ 替换具有最大函数值的插值点. 注意, 我们选择 $\boldsymbol{y}_2$ 作为信赖域中心, 因为它在 $\boldsymbol{y}_1,\boldsymbol{y}_2,\boldsymbol{y}_3$ 中具有最小的函数值.

步 4. 使用表 2-9 中列出的不同方法获取不同的 Q_1 进行比较, 如果新点的函数值小于插值集中的已有点, 则将新点添加到插值集中, 并舍弃距离具有最小函数值的点 (在欧氏距离的意义下) 最远的点.

步 5. 通过求解

$$\min_{\boldsymbol{d}} Q_1(\boldsymbol{x}_{\text{small}}+\boldsymbol{d})$$
$$\text{s. t. } \|\boldsymbol{d}\|_2 \le \Delta_0,$$

再次获取 $\boldsymbol{d}^*$, 其中 $\boldsymbol{x}_{\text{small}}$ 是当前探测点中具有最小函数值的点, 令 $\boldsymbol{y}_5=\boldsymbol{x}_{\text{small}}+\boldsymbol{d}^*$, 并记 $\boldsymbol{x}_{\min}$ 为使得

$$f(\boldsymbol{x}_{\min})=\min\left\{f(\boldsymbol{y}_1),f(\boldsymbol{y}_2),f(\boldsymbol{y}_3),f(\boldsymbol{y}_4),f(\boldsymbol{y}_5)\right\}.$$

成立的点.

注意, 所有算法在第一步中使用相同的模型 Q_0, 它们共用相同的 $\boldsymbol{y}_4=(1.6552,6.2446)^{\mathsf{T}}$, 对应的$f(\boldsymbol{y}_4)=1.23\times10^3$, 其由在信赖域内极小化最小 Frobenius 范数二次模型函数得到. 此外, 它们根据表 2-9 在第二步中构造不同的模型函数.

表 2-9 例2.6的结果: 使用不同模型的结果

Table 2-9 Results of Example 2.6: using different models

模型	子问题的目标函数	$f(\boldsymbol{x}_{\min})$
我们本节提出的模型 (考虑了最优性)	(2-42) 的目标函数	2.09
最小 Frobenius 范数二次模型	$\|\nabla^2 Q\|_F^2$ [156, 160, 188]	34.1
Powell 的二次模型	$\|\nabla^2 Q-\nabla^2 Q_{k-1}\|_F^2$ [93]	34.1
最小 H^2 范数更新二次模型	$\|Q-Q_{k-1}\|_{H^2}^2$ [97]	5.33
Conn 和 Toint 的二次模型	$\|\nabla^2 Q\|_F^2+\|\nabla Q(\boldsymbol{x}_k)\|_2^2$ [171]	74.9
共用相同的函数值约束条件: $Q(\boldsymbol{y}_i)=f(\boldsymbol{y}_i),\ \forall\ \boldsymbol{y}_i\in\mathcal{X}_k$		

可以看到, 使用我们在本节提出的二次模型的算法得到的迭代点的函数值比那些使用不考虑信赖域迭代的模型的算法得到的对应的函数值更小.

考虑到我们在本节提出的插值模型函数主要适用于基于插值模型的无导数信赖域算法, 我们在这里给出本节测试用的一个基于模型的无导数信赖域算法的

框架 (这里实验所使用的框架和之前的框架在细节上不完全一样), 如算法 4 所示. 关于无导数信赖域方法的更多介绍可以参见 Larson、Menickelly 和 Wild 的综述 [128] 以及 Conn、Scheinberg 和 Vicente 的著作 [20] 等.

算法 4 用于测试的基于模型的无导数信赖域算法

(初始化)

设置参数 $\varepsilon, \varepsilon_{\text{stop}} > 0$, η_0, η_1: $0 \leq \eta_0 \leq \eta_1 < 1$ 和 γ_0, γ_1: $0 < \gamma_0 < 1 \leq \gamma_1$, $\bar{\gamma}$.

选择一个初始点 $\boldsymbol{x}_{\text{int}}$ 和值 $f(\boldsymbol{x}_{\text{int}})$. 选择一个初始信赖域半径 $\Delta_0 > 0$ 和半径上界 $\Delta_{\text{up}} > \Delta_0$. 选择一个初始的适定插值集 $\mathcal{X}_0$ [20]. 确定 $\boldsymbol{x}_0 \in \mathcal{X}_0$, 使其在当前点中具有最小目标函数值, 即 $f(\boldsymbol{x}_0) = \min\limits_{\boldsymbol{y}_i \in \mathcal{X}_0} f(\boldsymbol{y}_i)$.

步 1. (构造模型)

使用插值集 $\mathcal{X}_k$ 构造一个插值模型 $Q_k(\boldsymbol{x})$.

while $\|\nabla Q_k(\boldsymbol{x}_k)\|_2 < \varepsilon$ **do**

　if Q_k 在 $\mathcal{B}_{\Delta_k}(\boldsymbol{x}_k)$ 上准确 **then**

　　令 $\Delta_k = \bar{\gamma}\Delta_k$.

　else

　　通过更新 $\mathcal{X}_k$ 使 Q_k 在 $\mathcal{B}_{\Delta_k}(\boldsymbol{x}_k)$ 上准确.

　end if

end while

步 2. (停机准则)

if $\Delta_k < \varepsilon_{\text{stop}}$ 和 $\|\nabla Q_k(\boldsymbol{x}_k)\|_2 < \varepsilon_{\text{stop}}$ **then**

　终止算法.

end if

步 3. (在信赖域内极小化模型)

计算 $\boldsymbol{d}_k$ 使得

$$Q_k(\boldsymbol{x}_k + \boldsymbol{d}_k) = \min_{\|\boldsymbol{d}\|_2 \leq \Delta_k} Q_k(\boldsymbol{x} + \boldsymbol{d}).$$

if $\boldsymbol{d}_k = \boldsymbol{0}$ **then**

　转到**步 4** 并按 $\rho_k \leq \eta_0$ 的情况处理;

else

　获取函数值 $f(\boldsymbol{x}_k + \boldsymbol{d}_k)$, 并令

$$\rho_k = \frac{f(\boldsymbol{x}_k) - f(\boldsymbol{x}_k + \boldsymbol{d}_k)}{Q_k(\boldsymbol{x}_k) - Q_k(\boldsymbol{x}_k + \boldsymbol{d}_k)}.$$

end if

步 4. (更新插值集和信赖域半径)

if $\rho_k < \eta_1$ 并且 Q_k 在 $\mathcal{B}_{\Delta_k}(\boldsymbol{x}_k)$ 上不准确 **then**

　在 $\mathcal{B}_{\Delta_k}(\boldsymbol{x}_k)$ 中生成新的插值点并将其添加到 $\mathcal{X}_k$ 中以改进 $\mathcal{X}_{k+1}$ 的适定性, 之后舍弃一个插值点.

end if

if $\rho_k \geq \eta_1$ **then**

扩大信赖域半径: 令 $\Delta_{k+1} = \min\{\Delta_{\text{up}}, \gamma_1 \Delta_k\}$;

更新插值集: 令 $\mathcal{X}_{k+1} = \mathcal{X}_k \cup \{\boldsymbol{x}_{k+1}\} \backslash \{\arg\max_{\boldsymbol{x}} \|\boldsymbol{x} - \boldsymbol{x}_{k+1}\|_2\}$.

else if Q_k 在 $\mathcal{B}_{\Delta_k}(\boldsymbol{x}_k)$ 上准确 **then**

缩小信赖域半径: 令 $\Delta_{k+1} = \gamma_0 \Delta_k$;

else

令 $\Delta_{k+1} = \Delta_k$.

end if

步 5. (更新当前迭代点)

if $\rho_k > \eta_0$ **then**

选取 $\boldsymbol{x}_{k+1}$ 为满足 $f(\boldsymbol{x}_{k+1}) \leq f(\boldsymbol{x}_k + \boldsymbol{d}_k)$ 的点;

else

令 $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k$.

end if

令 $k = k + 1$, 转到**步 1**.

注 2.9. 算法中提及的准确模型通常指的是完全线性模型 (更多细节见 Conn、Scheinberg 和 Vicente 著作 [20] 的第 6 章, 特别是定义 6.1), 此处对应的是 KKT 方程组 (2-44) 的系数矩阵是可逆的情况. 在当前实现中的第 1 步, 若系数矩阵不可逆, 则通过扰动插值点来改进插值集和模型 (测试中设定 $\bar{\gamma} = 1$), 以获得精确模型. 在第 4 步中, 改进 $\mathcal{X}_k$ 的适定性的步骤也被称为模型改进步 (更多细节见 Conn、Scheinberg 和 Vicente 著作 [20] 的第 6 章, 特别是算法 6.3). 为了简化并专注于模型的比较, 在当前实现中, 这一步的具体做法是: 在 $\rho_k < \eta_1$ 的情况下将 $\boldsymbol{x}_{\text{far}} = \arg\max_{\boldsymbol{x} \in \mathcal{X}_k} \|\boldsymbol{x} - \boldsymbol{x}_k\|_2$ 从 $\mathcal{X}_k$ 中去除, 并将 $\boldsymbol{x}_k + \boldsymbol{d}_k$ 包含进去, 即使算法没有接受 $\boldsymbol{x}_k + \boldsymbol{d}_k$ 作为下一个迭代点. 实际上, 我们已经获取了函数值 $f(\boldsymbol{x}_k + \boldsymbol{d}_k)$, 并且应该利用上这样的已知探测值 [160]. 在第 1 步, 它将使用 Frobenius 范数二次模型作为初始迭代的初始二次模型 $Q_0(\boldsymbol{x})$ (基于 $\mathcal{X}_0$ 的最小 Frobenius 二次模型), 然后在后续迭代中构造相应的欠定模型 (例如, 通过求解 (2-44) 获得我们在本节提出的模型).

如果 $m \geq n+1$, 那么至少可以得到与通过求解最小范数类型子问题而获得的模型的相应信赖域算法相同的收敛结果. 对于这样的模型, 可以证明, 在相应假设下, 信赖域无导数优化算法的任意极限点 $\boldsymbol{x}^*$ 都是一个稳定点, 即 $\nabla f(\boldsymbol{x}^*) = \boldsymbol{0}$ [20].

我们给出相应的 Performance Profile 和 Data Profile, 以观察使用考虑了其在信赖域内的最优性的新模型函数的信赖域算法求解无约束测试问题 (1-1) 的数值表现.

使用我们在本节所提出的模型函数的方法 (算法 4, 对应脚标 $a = 1$) 与使用相同框架但使用最小 Frobenius 范数二次模型 [156, 160, 188]、最小 Frobenius 范数更新二次模型 (Powell 的模型) [93]、最小 H^2 范数更新二次模型 [97]、Conn 和 Toint

的模型 [171], 以及最先进的无导数方法 NEWUOA [94]、Fminsearch (MATLAB 中的优化工具箱) [189, 190]、Fminunc (MATLAB 中的优化工具箱) [189]、CMA-ES [131] 和 NMSMAX [191–193] 进行了比较, 分别编号为脚标 $a = 2$ 到 $a = 10$.

算法 4 中用于缩小或扩大半径以及判断是否接受所获取的点的参数分别为 $\eta_1 = 0.75$ 和 $\eta_0 = 0$. 此外, 信赖域半径的缩小和扩大因子分别为 $\gamma_0 = 0.8$ 和 $\gamma_1 = 1.5$. 信赖域半径和模型函数梯度范数的容差分别为 10^{-6} 和 10^{-5}. 模型精度参数 $\varepsilon = 10^{-8}$. 它们在每次迭代时使用 $m = 2n + 1$ 个插值点, 初始插值点为原点和点 $\pm\frac{1}{2}\Delta_0 \boldsymbol{e}_i, i = 1, \cdots, n$, 其中 $\boldsymbol{e}_i \in \mathfrak{R}^n$ 表示仅第 i 个元素为 1, 其余为 0 的向量.

在数值测试中, NEWUOA 使用 $2n+1$ 个插值点 (初始点与本节所提出的方法相同) 来构造相应的二次模型函数, 且有 $\rho_{\text{end}} = 10^{-6}$ 和与前五种不同模型的方法的初始半径相同的 ρ_{beg}. 算法 Fminsearch 的函数值和迭代点的容差为 10^{-6}. 方法 Fminunc 被设置为使用带有步长为 0.1 的有限差分梯度的拟牛顿方法, 其他相应的容差选择和 Fminsearch 的选择具有相同规模. 算法 CMA-ES 的相关参数设置使用默认值 [131], 其函数值的相关停机准则为 10^{-5}. 对于算法 NMSMAX, 当单纯形的相应尺度小于或等于 10^{-6} 时终止迭代, 初始单纯形为正则单纯形, 其边等长. 所有比较的算法都有相同的函数值探测次数停机准则, 即总探测次数不超过 $100n$, 其中 n 是相应问题的维数.

我们选择表 2-10 中所列出的测试问题集 $\mathcal{P}$ (包含 110 个问题, 其中有 51 种不同类型的问题, 维数从 2 到 800 不等) 来测试我们的算法在求解无约束无导数问题时的效果. 注意到, 我们的测试问题的问题维数的平均值约为 74, 标准差约为 129. 它们来自经典常见的无约束优化测试函数集合, 大多数函数 f 描述的测试优化问题是光滑的 (BROYDN7D 和 TRIGSABS 是分段光滑的).

表 2-10 图 2-10 和图 2-11 对应的 110 个测试问题

Table 2-10 110 test problems for Figure 2-10 and Figure 2-11

问题	维数	$f(\boldsymbol{x}_{\text{int}})$	$f(\boldsymbol{x}^*)$
ARGLINA [177, 178]	2	1.00×10^{1}	2.00
ARGLINB [177, 178]	2	2.14×10^{2}	6.67×10^{1}
BDVALUE [177, 178]	2	2.43×10^{-2}	2.18×10^{-15}
BROYDN3D [177, 178]	2	1.30×10^{1}	6.03×10^{-17}
BROYDN7D [177, 184]	2	7.81	6.59×10^{1}
CHEBQUAD [177, 178]	2	1.98×10^{-1}	9.50×10^{-18}
CHROSEN [94]	2	2.00×10^{1}	2.09×10^{-18}
CURLY10 [177]	2	-1.01×10^{-5}	-2.01×10^{2}
CURLY20 [177]	2	-1.01×10^{-5}	-2.01×10^{2}
CURLY30 [177]	2	-1.01×10^{-5}	-2.01×10^{2}
DIXMAANE [177]	2	7.00	1.00
DIXMAANF [177]	2	7.00	1.00

表 2-10 (续表)

DIXMAANG [177]	2	7.00	1.00
DIXMAANH [177]	2	7.00	1.00
DIXMAANI [177]	2	6.00	1.00
DIXMAANJ [177]	2	6.00	1.00
DIXMAANK [177]	2	6.00	1.00
DIXMAANL [177]	2	6.00	1.00
DIXMAANM [177]	2	6.00	1.00
DIXMAANN [177]	2	6.00	1.00
DIXMAANO [177]	2	6.00	1.00
DIXMAANP [177]	2	6.00	1.00
ENGVAL1 [177]	2	5.90×10^{1}	0.00
EXPSUM [182]	2	5.00	0.00
INTEGREQ [177, 178]	2	2.06×10^{-2}	2.96×10^{-18}
MOREBV [177, 178]	2	2.43×10^{-2}	2.18×10^{-15}
MOREBVL [180]	2	7.13	4.10×10^{-17}
NONCVXU2 [177]	2	3.95×10^{1}	4.63
NONCVXUN [177]	2	5.32×10^{1}	4.63
NONDIA [177]	2	1.02×10^{4}	1.00
POWER [177]	2	5.00	0.00
SBRYBND [177, 178]	2	2.67×10^{14}	4.99×10^{-12}
SPARSINE [177]	2	1.24×10^{1}	9.47×10^{-15}
TOINTTRIG [184]	2	1.03×10^{1}	-2.00×10^{1}
TRIGSSQS [94]	2	3.22×10^{3}	1.25×10^{-17}
TRIROSE1 [186]	2	1.55×10^{3}	2.27×10^{-15}
ARGLINA [177, 178]	3	1.50×10^{1}	3.00
ARGLINB [177, 178]	3	3.03×10^{3}	1.15
ARGLINC [177]	3	8.60×10^{1}	2.67
BROYDN3D [177, 178]	3	1.40×10^{1}	1.64×10^{-15}
EXTTET [185]	3	2.91	2.56
NONCVXU2 [177]	3	1.18×10^{2}	6.95
NONCVXUN [177]	3	1.57×10^{2}	6.95
POWER [177]	3	1.40×10^{1}	0.00
SPARSINE [177]	3	2.48×10^{1}	1.23×10^{-13}
TOINTGSS [177]	3	1.10×10^{1}	2.00
FLETCHCR [177]	5	4.00×10^{2}	1.21×10^{-12}
TOINTTRIG [184]	5	-1.37×10^{1}	-2.50×10^{2}
BROYDN3D [177, 178]	4	1.50×10^{1}	1.79×10^{-13}

表 2-10 (续表)

NONCVXUN [177]	4	2.90×10^{2}	9.27
POWER [177]	4	3.00×10^{1}	0.00
FLETCHCR [177]	6	5.00×10^{2}	7.91×10^{-13}
POWELLSG [177, 178]	6	2.15×10^{2}	1.39×10^{-15}
SBRYBND [177, 178]	6	2.68×10^{14}	3.60×10^{3}
BROYDN3D [177, 178]	7	1.80×10^{1}	6.83×10^{-13}
EXTTET [185]	7	8.73	7.68
SBRYBND [177, 178]	9	2.73×10^{14}	2.83×10^{3}
GENROSE [177]	15	9.59×10^{1}	1.00
SCOSINEL [180]	16	4.56	-1.19×10^{1}
TRIGSSQS [94]	17	2.47×10^{6}	5.78×10^{1}
GENROSE [177]	18	1.06×10^{2}	1.00
TOINTTRIG [184]	20	-9.09×10^{1}	-4.75×10^{3}
SROSENBR [177, 178]	25	2.90×10^{2}	2.36×10^{1}
TRIGSSQS [94]	26	1.38×10^{7}	1.36×10^{3}
EXTTET [185]	37	5.24×10^{1}	4.61×10^{1}
COSINE [177]	39	3.33×10^{1}	-3.80×10^{1}
PENALTY3 [177]	40	2.72×10^{6}	1.00
TRIGSABS [94]	42	5.56×10^{3}	4.36×10^{1}
TOINTTRIG [184]	43	-9.89×10^{2}	-2.26×10^{4}
SCOSINEL [180]	46	3.72	-2.33×10^{1}
TRIGSSQS [94]	46	2.45×10^{7}	1.13×10^{1}
BROYDN7D [177, 184]	48	1.33×10^{2}	6.01
TRIGSABS [94]	50	8.48×10^{3}	3.32×10^{1}
COSINE [177]	55	4.74×10^{1}	-5.40×10^{1}
TRIGSSQS [94]	61	3.23×10^{7}	3.43×10^{2}
COSINE [177]	63	5.44×10^{1}	-6.19×10^{1}
PENALTY3 [177]	65	1.82×10^{7}	1.55×10^{4}
TRIGSABS [94]	66	2.38×10^{4}	1.11×10^{2}
TRIGSABS [94]	68	1.93×10^{4}	9.33×10^{1}
TOINTGSS [177]	84	7.48×10^{2}	9.65
TOINTGSS [177]	89	7.93×10^{2}	9.67
PENALTY3 [177]	90	6.62×10^{7}	2.90×10^{4}
PENALTY2 [177, 178]	100	9.73×10^{9}	8.93×10^{9}
TOINTGSS [177]	100	8.92×10^{2}	9.71
TOINTGSS [177]	108	9.64×10^{2}	9.73
SROSENBR [177, 178]	115	1.38×10^{3}	3.53

表 2-10 (续表)

PENALTY2 [177, 178]	118	3.55×10^{11}	2.63×10^{11}
SPARSINE [177]	119	2.95×10^{4}	1.46×10^{6}
SROSENBR [177, 178]	120	1.45×10^{3}	5.79
SROSENBR [177, 178]	130	1.57×10^{3}	4.29
PENALTY3 [177]	140	3.85×10^{8}	1.47
PENALTY2 [177, 178]	161	1.93×10^{15}	1.04×10^{15}
TQUARTIC [177]	165	8.10×10^{-1}	1.81×10^{3}
SROSENBR [177, 178]	175	2.11×10^{3}	2.31×10^{1}
PENALTY2 [177, 178]	192	9.51×10^{17}	5.52×10^{17}
TQUARTIC [177]	193	8.10×10^{-1}	2.69×10^{3}
ARGLINA [177, 178]	250	1.25×10^{2}	2.50×10^{2}
ARGLINB [177, 178]	250	4.11×10^{16}	1.25×10^{2}
CHNROSNB [177, 180]	250	5.46×10^{3}	1.04×10^{-9}
CHROSEN [94]	250	4.98×10^{3}	4.90×10^{-3}
COSINE [177]	250	2.19×10^{2}	-2.49×10^{2}
CURLY30 [177]	250	-2.91×10^{-4}	-2.50×10^{4}
PENALTY2 [177, 178]	300	2.29×10^{27}	1.78×10^{27}
TOINTTRIG [184]	350	1.04×10^{1}	-1.53×10^{6}
COSINE [177]	220	1.92×10^{2}	-2.19×10^{2}
ROSENBROCK [177, 178]	250	1.01×10^{5}	2.16×10^{2}
GENROSE [177]	320	1.21×10^{3}	2.03×10^{2}
ARGLINA [177, 178]	500	2.50×10^{3}	5.00×10^{2}
ARGLINA [177, 178]	600	3.00×10^{3}	6.00×10^{2}
PENALTY3 [177]	800	4.09×10^{11}	1.11

图 2-10 和图 2-11 给出了所测试的无导数优化方法的 Performance Profile 和 Data Profile, 其中 τ 的取值分别为 10^{-1}、10^{-3} 和 10^{-5}. 我们在本节提出的方法 (简单起见, 在图中标记为 “模型 (最优性)”) 在此类问题和精度上具有最好的数值表现.

我们可以看到所比较方法之间的性能差异. 例如, 在图 2-10 显示的 Performance Profile 中, 当 $\alpha = 1$ 时, 我们的方法在 τ 取值为 10^{-1}、10^{-3} 和 10^{-5} 的三种情况下有最高的值 (50%、52.73% 和 57.27%), 这意味着在所有测试和对比的方法中, 它成功求解问题的数量是最多的. 图 2-11 中的 Data Profile 也显示: 对于所有列出的精度 τ, 我们本节提出的模型方法成功求解了最高比例的问题.

此外, 表 2-11 显示了相应的问题求解比率, 其中脚标 $1, 2, \cdots, 10$ 表示图 2-10 和图 2-11 所比较的 10 种方法: “模型 (最优性)” (即本节所提出的)、“最小 Frob. 范数”、“Powell”、“最小 H^2 范数更新”、“Conn & Toint”、“NEWUOA”、“Fminsearch”、“Fminunc”、“CMA-ES” 和 “NMSMAX”. 例如, 在 Performance Profile 中, 针对 $\tau =$

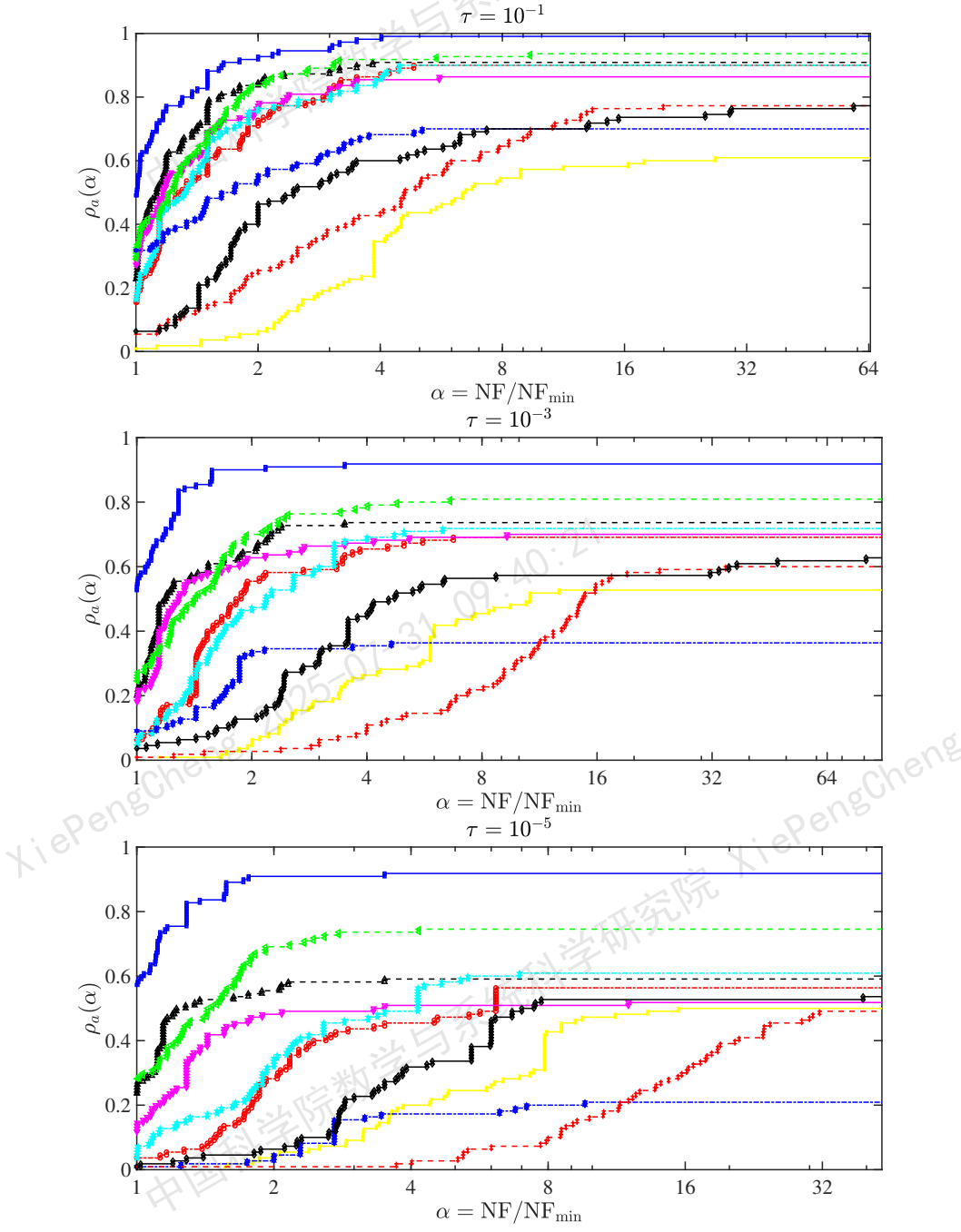


图 2-10 求解测试问题的 Performance Profile

Figure 2-10 Performance Profile of minimizing test problems

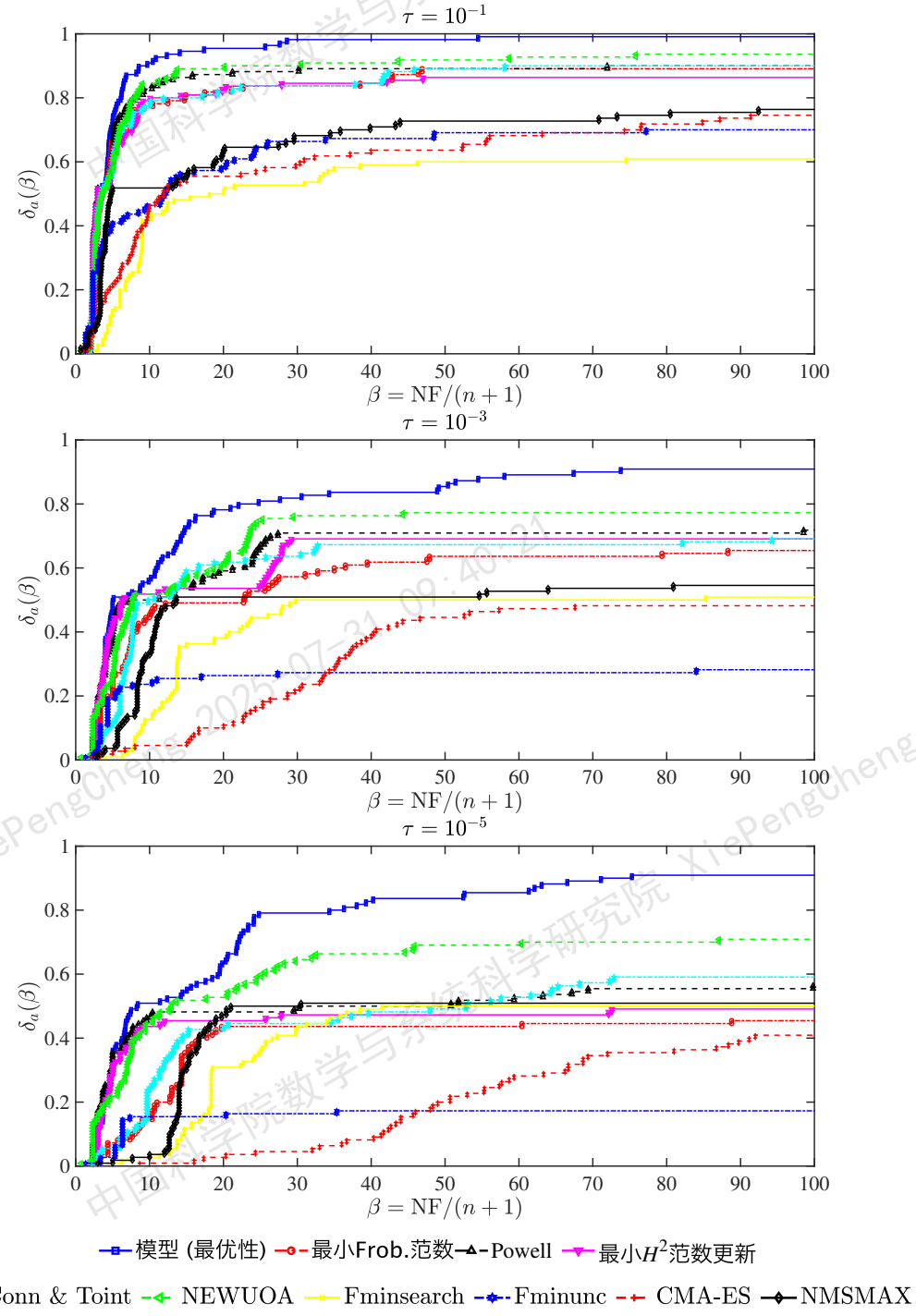


图 2-11 求解测试问题的 Data Profile

Figure 2-11 Data Profile of minimizing test problems

中国科学院数学与系统科学研究院 XiePengCheng 2025-07-31 09:40:21

表 2-11 成功求解问题的比例

Table 2-11 The ratio of the solved problems

$\alpha = 2$ 时 Performance Profile 中成功求解问题的比例					
τ	$\rho_1(2)$	$\rho_2(2)$	$\rho_3(2)$	$\rho_4(2)$	$\rho_5(2)$
10^{-1}	92.73%	72.73%	84.55%	77.27%	76.36%
10^{-3}	90.00%	55.45%	64.55%	62.73%	47.27%
10^{-5}	90.91%	28.18%	55.45%	48.18%	34.55%
τ	$\rho_6(2)$	$\rho_7(2)$	$\rho_8(2)$	$\rho_9(2)$	$\rho_{10}(2)$
10^{-1}	83.64%	6.36%	55.45%	25.45%	46.36%
10^{-3}	70.00%	6.36%	33.64%	2.73%	12.73%
10^{-5}	69.09%	4.55%	4.55%	0.91%	6.36%
$\beta = 30$ 时 Data Profile 中成功求解问题的比例					
τ	$\delta_1(30)$	$\delta_2(30)$	$\delta_3(30)$	$\delta_4(30)$	$\delta_5(30)$
10^{-1}	98.18%	83.64%	88.18%	84.55%	83.64%
10^{-3}	81.82%	57.27%	70.91%	69.09%	63.64%
10^{-5}	79.09%	43.64%	49.09%	47.27%	44.55%
τ	$\delta_6(30)$	$\delta_7(30)$	$\delta_8(30)$	$\delta_9(30)$	$\delta_{10}(30)$
10^{-1}	90.00%	52.73%	66.36%	59.09%	68.18%
10^{-3}	76.36%	50.00%	27.27%	21.82%	50.91%
10^{-5}	64.55%	43.64%	16.36%	4.55%	50.00%

10^{-5} 的情况, 我们在本节给出的方法在 $\alpha = 2$ 时可以成功求解 90.91% 的问题, 这比其他算法更多. 此外, 在最多 $30(n+1)$ 次函数值探测及 $\tau = 10^{-5}$ 的精度下, Data Profile 表明我们在本节提出的方法可以成功求解 79.09% 的问题, 这大约比第二高的算法 (NEWUOA) 多出 15%. 上述 Profile 充分展示了我们在本节所提出的模型和方法的优势.

2.3.5 小结

在本节中, 我们借助信赖域迭代分析和改进了 Conn 和 Toint 的模型, 并通过考虑迭代点的最优性构造了一个新的无导数优化逼近模型. 本节利用信赖域迭代来帮助构造插值模型. 我们找到并利用了最优性和插值之间更多的关系, 给出了我们本节所提出方法的动机、相应的凸性分析, 以及如何以便于实现的方式获得所提出的二次模型的系数. 数值结果显示了我们在本节所提出的模型和方法的优势.

除了基于使用一般模型的算法的传统收敛性结果之外, 使用我们在本节新提出模型的方法的新收敛性结果还有待进一步研究. 关于在构造我们本节所提出的模型时插值点数量最佳选择的研究也是有价值的. 另一项可能的未来工作是对当前迭代点不成功时的情况的相应处理.

2.4 信赖域中非凸二次函数极小点之间距离减小的充分条件

本节分析了非凸二次函数在信赖域内的极小点之间的距离在两次迭代后减小的充分条件. 本节还给出了一些与理论结果相对应的例子.

我们知道, 信赖域方法通过求解子问题

$$\begin{aligned} &\min_{\boldsymbol{x}\in\mathfrak{R}^n} \ \mathrm{Model}(\boldsymbol{x}) \\ &\text{s. t. } \ \|\boldsymbol{x}-\boldsymbol{x}_c\|_2 \le \Delta_k \end{aligned}$$

来获得下一个迭代点, 其中 $\boldsymbol{x}_c \in \mathfrak{R}^n$ 是信赖域 $\mathcal{B}_{\Delta_k}(\boldsymbol{x}_c) = \{\boldsymbol{z} \in \mathfrak{R}^n, \|\boldsymbol{z}-\boldsymbol{x}_c\|_2 \le \Delta_k\}$ 的中心, $\Delta_k > 0$ 是第 k 步的信赖域半径, 这里的函数 Model 是需要极小化的逼近目标函数的二次模型函数. 这里用函数 Model 来表示更一般情形下的模型函数, 同时与下文的 f 和 Q 作以区分. 因此, 二次模型函数对于给出下一个迭代点很重要. 本节考虑了两个非凸二次函数的极小点在相应信赖域中的距离, 这是因为我们观察和注意到, 对于使用二次模型函数的信赖域方法来说, 其模型的本质作用就是提供一个近似原真实目标函数极小点的模型函数极小点.

本节将试图探究两个二次模型函数 f 和 Q 之间具有何种关系时可以让这两个二次模型函数在信赖域中的数值极小点之间的距离在迭代后减小, 这将通过定理 2.29 和定理 2.31 给出. 这些结果有助于我们迭代地修正二次模型、处理基于导数的或不基于导数的信赖域方法 [20, 92, 97, 102, 150] 中二次模型的选择. 此外, 我们使用具体例子来说明我们的结果是适用的. 譬如, 我们可以直接使用这些条件来判断两个不同模型是否可以在迭代后让两个相应极小点之间的距离减小.

请注意, 二次函数 f 和 Q 指的是信赖域算法中的二次模型函数. 需要特别说明的是, 在本节, f 不是原始目标函数, 虽然在我们想要极小化二次函数的情况下 f 可以作为目标函数. 换句话说, 一般情况下, 我们认为这里的二次函数 f 和 Q 都是信赖域子问题中出现的二次模型.

总而言之, 两个非凸二次函数在相应信赖域中的极小点之间的距离在某些情况下会减小. 本节推导了这种情况的充分条件.

在下面的内容中, 我们假设 $\boldsymbol{x}_1 \in \mathfrak{R}^n$ 和 $\tilde{\boldsymbol{x}}_1 \in \mathfrak{R}^n$ 分别是非凸二次函数 f 和 Q 在信赖域 $\mathcal{B}_{\Delta_1}(\boldsymbol{x}_0)$ 和 $\mathcal{B}_{\tilde{\Delta}_1}(\boldsymbol{x}_0)$ 中的极小点, $\boldsymbol{x}_2$ 和 $\tilde{\boldsymbol{x}}_2$ 分别是 f 和 Q 在信赖域 $\mathcal{B}_{\Delta_2}(\boldsymbol{x}_1)$ 和 $\mathcal{B}_{\tilde{\Delta}_2}(\tilde{\boldsymbol{x}}_1)$ 中的极小点, 其中 $\boldsymbol{x}_0 \in \mathfrak{R}^n$ 是初始点 (或第一个信赖域的中心), $\Delta_1 \in \mathfrak{R}^+$, $\tilde{\Delta}_1 \in \mathfrak{R}^+$, $\Delta_2 \in \mathfrak{R}^+$, 和 $\tilde{\Delta}_2 \in \mathfrak{R}^+$ 是信赖域半径. 换句话说, 存在实参数 $\omega_1, \tilde{\omega}_1, \omega_2, \tilde{\omega}_2 > 0$ 使得

$$\begin{cases} \boldsymbol{x}_1 - \boldsymbol{x}_0 = -\left(\nabla^2 f + \omega_1 \boldsymbol{I}\right)^{-1} \nabla f(\boldsymbol{x}_0), \\ \tilde{\boldsymbol{x}}_1 - \boldsymbol{x}_0 = -\left(\nabla^2 Q + \tilde{\omega}_1 \boldsymbol{I}\right)^{-1} \nabla Q(\boldsymbol{x}_0) \end{cases} \tag{2-45}$$

且

$$\begin{cases} \boldsymbol{x}_2 - \boldsymbol{x}_1 = -\left(\nabla^2 f + \omega_2 \boldsymbol{I}\right)^{-1} \nabla f(\boldsymbol{x}_1), \\ \tilde{\boldsymbol{x}}_2 - \tilde{\boldsymbol{x}}_1 = -\left(\nabla^2 Q + \tilde{\omega}_2 \boldsymbol{I}\right)^{-1} \nabla Q(\tilde{\boldsymbol{x}}_1), \end{cases} \tag{2-46}$$

其中 $\Delta_1 = \|\boldsymbol{x}_1 - \boldsymbol{x}_0\|_2$, $\tilde{\Delta}_1 = \|\tilde{\boldsymbol{x}}_1 - \boldsymbol{x}_0\|_2$, $\Delta_2 = \|\boldsymbol{x}_2 - \boldsymbol{x}_1\|_2$, $\tilde{\Delta}_2 = \|\tilde{\boldsymbol{x}}_2 - \boldsymbol{x}_1\|_2$, 且 $\nabla^2 f + \omega_1 \boldsymbol{I} \succeq \boldsymbol{0}, \nabla^2 Q + \tilde{\omega}_1 \boldsymbol{I} \succeq \boldsymbol{0}, \nabla^2 f + \omega_2 \boldsymbol{I} \succeq \boldsymbol{0}, \nabla^2 Q + \tilde{\omega}_2 \boldsymbol{I} \succeq \boldsymbol{0}$.

假设 2.27. 假设 f 和 Q 是非凸二次函数, $\nabla^2 f + \omega_2 \boldsymbol{I} \succ \boldsymbol{0}, \nabla^2 Q + \tilde{\omega}_2 \boldsymbol{I} \succ \boldsymbol{0}$, 且 $\tilde{\boldsymbol{x}}_1 \neq \boldsymbol{x}_1$.

注 2.10. 简明起见, 本节在不同结果中使用的相同符号可能具有不同的维数. 此外, 矩阵 $\boldsymbol{A} \succ \boldsymbol{0}$ 表示矩阵 $\boldsymbol{A}$ 是正定的.

我们所讨论的问题如下.

问题. 在假设 2.27 下, 对于 $0 \leq \rho \leq 1$, 二次函数 f 和 Q 在信赖域中的极小点在什么样的充分条件下可以满足

$$\left\|\tilde{\boldsymbol{x}}_2 - \boldsymbol{x}_2\right\|_2 \leq \rho \left\|\tilde{\boldsymbol{x}}_1 - \boldsymbol{x}_1\right\|_2 \ ? \tag{2-47}$$

2.4.1　二次函数在信赖域中的极小点的距离分析

下面我们给出关于二次函数的极小点之间的距离的结果.

命题 2.28. *极小点之间的差满足*

$$\tilde{\boldsymbol{x}}_2 - \boldsymbol{x}_2 = \tilde{\omega}_1 \left(\nabla^2 Q + \tilde{\omega}_2 \boldsymbol{I}\right)^{-1} \left(\tilde{\boldsymbol{x}}_1 - \boldsymbol{x}_0\right) - \omega_1 \left(\nabla^2 f + \omega_2 \boldsymbol{I}\right)^{-1} \left(\boldsymbol{x}_1 - \boldsymbol{x}_0\right) + \left(\tilde{\boldsymbol{x}}_1 - \boldsymbol{x}_1\right).$$

证明. 根据 (2-45) 和 (2-46), 我们有

$$\begin{aligned}\boldsymbol{x}_2-\boldsymbol{x}_1 &= -\left(\nabla^2 f+\omega_2\boldsymbol{I}\right)^{-1}\nabla f(\boldsymbol{x}_1)\\ &= -\left(\nabla^2 f+\omega_2\boldsymbol{I}\right)^{-1}\left(\nabla f(\boldsymbol{x}_0)+\nabla^2 f\cdot\left(\boldsymbol{x}_1-\boldsymbol{x}_0\right)\right)\\ &= -\left(\nabla^2 f+\omega_2\boldsymbol{I}\right)^{-1}\left(-\left(\nabla^2 f+\omega_1\boldsymbol{I}\right)\left(\boldsymbol{x}_1-\boldsymbol{x}_0\right)+\nabla^2 f\cdot\left(\boldsymbol{x}_1-\boldsymbol{x}_0\right)\right)\\ &= \omega_1\left(\nabla^2 f+\omega_2\boldsymbol{I}\right)^{-1}\left(\boldsymbol{x}_1-\boldsymbol{x}_0\right)\end{aligned}$$

和

$$\tilde{\boldsymbol{x}}_2-\tilde{\boldsymbol{x}}_1=\tilde{\omega}_1\left(\nabla^2 Q+\tilde{\omega}_2\boldsymbol{I}\right)^{-1}\left(\tilde{\boldsymbol{x}}_1-\boldsymbol{x}_0\right),$$

进而可通过直接计算得证. □

定理 2.29 (1 维情况下的充分必要条件)**.** *假定假设 2.27 成立, 维数 $n=1$ 且 $\kappa:=\frac{\boldsymbol{x}_1-\boldsymbol{x}_0}{\tilde{\boldsymbol{x}}_1-\boldsymbol{x}_1}\in\mathfrak{R}$, 则 (2-47) 对于 $0\le\rho\le 1$ 成立当且仅当*

$$\begin{cases}\left(\nabla^2 Q+\tilde{\omega}_2\right)\omega_1>\left(\nabla^2 f+\omega_2\right)\tilde{\omega}_1,\\ \kappa_1\le\kappa\le\kappa_2,\end{cases}\tag{2-48}$$

或者

$$\begin{cases}\left(\nabla^2 Q+\tilde{\omega}_2\right)\omega_1<\left(\nabla^2 f+\omega_2\right)\tilde{\omega}_1,\\ \kappa_2\le\kappa\le\kappa_1,\end{cases}\tag{2-49}$$

其中

$$\begin{cases}\kappa_1=\dfrac{\left(\nabla^2 f+\omega_2\right)\left[(-\rho+1)\left(\nabla^2 Q+\tilde{\omega}_2\right)+\tilde{\omega}_1\right]}{\left(\nabla^2 Q+\tilde{\omega}_2\right)\omega_1-\left(\nabla^2 f+\omega_2\right)\tilde{\omega}_1},\\ \kappa_2=\dfrac{\left(\nabla^2 f+\omega_2\right)\left[(\rho+1)\left(\nabla^2 Q+\tilde{\omega}_2\right)+\tilde{\omega}_1\right]}{\left(\nabla^2 Q+\tilde{\omega}_2\right)\omega_1-\left(\nabla^2 f+\omega_2\right)\tilde{\omega}_1}.\end{cases}$$

证明. 条件 (2-48) 或 (2-49) 有一者成立的充要条件是

$$\left|1+\frac{\tilde{\omega}_1(1+\kappa)}{\nabla^2 Q+\tilde{\omega}_2}-\frac{\omega_1\kappa}{\nabla^2 f+\omega_2}\right|\left\|\tilde{\boldsymbol{x}}_1-\boldsymbol{x}_1\right\|_2\le\rho\left\|\tilde{\boldsymbol{x}}_1-\boldsymbol{x}_1\right\|_2,$$

结论得证. □

推论 2.30. *假定假设 2.27 成立, 问题的维数 $n=1$, 且 $\kappa:=\frac{\boldsymbol{x}_1-\boldsymbol{x}_0}{\tilde{\boldsymbol{x}}_1-\boldsymbol{x}_1}\in\mathfrak{R}$. 若 $-1<\kappa<0$, 即 $\tilde{\boldsymbol{x}}_1\le\boldsymbol{x}_0<\boldsymbol{x}_1$ 或 $\boldsymbol{x}_1<\boldsymbol{x}_0\le\tilde{\boldsymbol{x}}_1$, 则不存在 $0<\rho<1$ 使得 (2-47) 成立.*

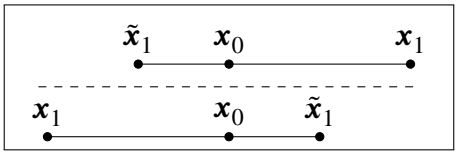


图 2-12 推论 2.30 对应的 $\boldsymbol{x}_0,\boldsymbol{x}_1,\tilde{\boldsymbol{x}}_1$ 的分布

Figure 2-12 Distribution of $\boldsymbol{x}_{k-1},\boldsymbol{x}_k,\tilde{\boldsymbol{x}}_k$ corresponding to Corollary 2.30

证明. 给定 $\omega_1 > 0, \tilde{\omega}_1 > 0, G > 0, H > 0, \rho > 0, -1 \leq \kappa \leq 0$, 有

$$-\rho \leq 1 + \frac{\tilde{\omega}_1(1+\kappa)}{G} - \frac{\omega_1 \kappa}{H} \leq \rho$$

等价于

$$\rho \geq \frac{GH - G\kappa\omega_1 + H\kappa\tilde{\omega}_1 + H\tilde{\omega}_1}{GH} \geq 1. \tag{2-50}$$

因此, 根据 (2-50), 结论得证. □

注 2.11. 图 2-12 展示了推论 2.30 的情况.

定理 2.31 (一般 n 维对角 Hessian 矩阵情况下的充分条件)**.** 假定假设 *2.27* 成立, 且 $\boldsymbol{\kappa} := \mathrm{Diag}\left\{\kappa^{[1]}, \kappa^{[2]}, \cdots, \kappa^{[n]}\right\} \in \mathfrak{R}^{n\times n}$ 满足 $\boldsymbol{\kappa}\left(\tilde{\boldsymbol{x}}_1 - \boldsymbol{x}_1\right) = \boldsymbol{x}_1 - \boldsymbol{x}_0$, 同时 $\nabla^2 f$ 和 $\nabla^2 Q$ 为对角矩阵. 若对任意给定的 $i \in \{1, 2, \cdots, n\}$, 有

$$\begin{cases} \left(\nabla^2 Q^{[i]} + \tilde{\omega}_2\right)\omega_1 > \left(\nabla^2 f^{[i]} + \omega_2\right)\tilde{\omega}_1, \\ \kappa_1^{[i]} \leq \kappa^{[i]} \leq \kappa_2^{[i]} \end{cases}$$

或

$$\begin{cases} \left(\nabla^2 Q^{[i]} + \tilde{\omega}_2\right)\omega_1 < \left(\nabla^2 f^{[i]} + \omega_2\right)\tilde{\omega}_1, \\ \kappa_2^{[i]} \leq \kappa^{[i]} \leq \kappa_1^{[i]} \end{cases}$$

成立, 其中

$$\begin{cases} \kappa_1^{[i]} = \dfrac{\left(\nabla^2 f^{[i]} + \omega_2\right)\left[(-\rho+1)\left(\nabla^2 Q^{[i]} + \tilde{\omega}_2\right) + \tilde{\omega}_1\right]}{\left(\nabla^2 Q^{[i]} + \tilde{\omega}_2\right)\omega_1 - \left(\nabla^2 f^{[i]} + \omega_2\right)\tilde{\omega}_1}, \\ \kappa_2^{[i]} = \dfrac{\left(\nabla^2 f^{[i]} + \omega_2\right)\left[(\rho+1)\left(\nabla^2 Q^{[i]} + \tilde{\omega}_2\right) + \tilde{\omega}_1\right]}{\left(\nabla^2 Q^{[i]} + \tilde{\omega}_2\right)\omega_1 - \left(\nabla^2 f^{[i]} + \omega_2\right)\tilde{\omega}_1}, \end{cases}$$

则 *(2-47)* 成立, 这里的上标 $[i]$ 表示相应矩阵 $\nabla^2 f$ 或 $\nabla^2 Q$ 的第 i 个对角元素, 或相应向量 κ_1 和 κ_2 的第 i 个元素.

证明. 我们有

$$\begin{aligned} &\left\|\tilde{\boldsymbol{x}}_2 - \boldsymbol{x}_2\right\|_2 \\ =&\left\|\left(\boldsymbol{I} + \tilde{\omega}_1\left(\nabla^2 Q + \tilde{\omega}_2 \boldsymbol{I}\right)^{-1}(\boldsymbol{I} + \boldsymbol{\kappa}) - \omega_1\left(\nabla^2 f + \omega_2 \boldsymbol{I}\right)^{-1}\boldsymbol{\kappa}\right)\left(\tilde{\boldsymbol{x}}_1 - \boldsymbol{x}_1\right)\right\|_2 \\ \leq&\left\|\rho\left(\tilde{\boldsymbol{x}}_1^{[1]} - \boldsymbol{x}_1^{[1]}, \cdots, \tilde{\boldsymbol{x}}_1^{[n]} - \boldsymbol{x}_1^{[n]}\right)^{\top}\right\|_2 = \rho\|\tilde{\boldsymbol{x}}_1 - \boldsymbol{x}_1\|_2, \end{aligned}$$

其中上标 $[i]$ 表示相应向量的第 i 个元素, 这是因为

$$\left|1 + \tilde{\omega}_1 \frac{1 + \kappa^{[i]}}{\nabla^2 Q^{[i]} + \tilde{\omega}_2} - \omega_1 \frac{\kappa^{[i]}}{\nabla^2 f^{[i]} + \omega_2}\right| \leq \rho, \ \forall\, i = 1, \cdots, n.$$

基于上述, 结论得证. □

推论 2.32. 假定假设 *2.27* 成立, 且 $\boldsymbol{\kappa} := \mathrm{Diag}\left\{\kappa^{[1]}, \kappa^{[2]}, \cdots, \kappa^{[n]}\right\} \in \mathfrak{R}^{n\times n}$ 满足 $\boldsymbol{\kappa}\left(\tilde{\boldsymbol{x}}_1 - \boldsymbol{x}_1\right) = \boldsymbol{x}_1 - \boldsymbol{x}_0$. 若对于 $\forall\, i$, 有 $-1 < \kappa^{[i]} < 0$, 即 $\tilde{\boldsymbol{x}}_1^{[i]} \leq \boldsymbol{x}_0^{[i]} < \boldsymbol{x}_1^{[i]}$ 或 $\boldsymbol{x}_1^{[i]} < \boldsymbol{x}_0^{[i]} \leq \tilde{\boldsymbol{x}}_1^{[i]}$, 则不存在 $0 < \rho < 1$ 使得 *(2-47)* 成立.

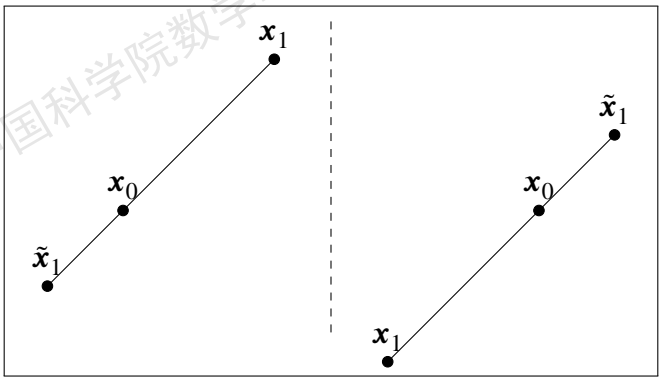


图 2-13 推论 2.32 对应的 $\boldsymbol{x}_0, \boldsymbol{x}_1, \tilde{\boldsymbol{x}}_1$ 的分布

Figure 2-13 Distribution of $\boldsymbol{x}_{k-1}, \boldsymbol{x}_k, \tilde{\boldsymbol{x}}_k$ corresponding to Corollary 2.32

证明. 可以直接根据推论 2.30 关于每个元素的结论得出结论. □

注 2.12. 图 2-13 展示了推论 2.32 中的情况.

2.4.2 例子

我们给出以下例子来说明上述结论.

例 2.7. 在这个例子中, 我们展示当维数 $n = 2$, 二次模型具有对角 Hessian 矩阵且 $\boldsymbol{\kappa}$ 具有不同的非零分量时的情况.

这里, 有

$$\begin{cases} f(\boldsymbol{x}) = -\dfrac{1}{2}\boldsymbol{x}^\top \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix} \boldsymbol{x} + \left(\dfrac{1}{7}, \dfrac{5}{3}\right)^\top \boldsymbol{x}, \\ Q(\boldsymbol{x}) = -\dfrac{1}{2}\boldsymbol{x}^\top \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \boldsymbol{x}. \end{cases}$$

此外, $\boldsymbol{x}_0 = (1,1)^\top$, $\omega_1 = 3$, $\tilde{\omega}_1 = 3$, $\omega_2 = 4$, $\tilde{\omega}_2 = 5$, $\rho = \frac{1}{2}$. 我们有

$$\boldsymbol{x}_1 = \boldsymbol{x}_0 - \left(\nabla^2 f(\boldsymbol{x}_0) + \omega_1 \boldsymbol{I}\right)^{-1} \nabla f = \begin{pmatrix} \frac{10}{7} \\ \frac{4}{3} \end{pmatrix},$$

$$\tilde{\boldsymbol{x}}_1 = \boldsymbol{x}_0 - \left(\nabla^2 Q(\boldsymbol{x}_0) + \tilde{\omega}_1 \boldsymbol{I}\right)^{-1} \nabla Q = \begin{pmatrix} \frac{3}{2} \\ \frac{3}{2} \end{pmatrix}$$

以及

$$\boldsymbol{\kappa} = \begin{pmatrix} \frac{\frac{10}{7}-1}{\frac{3}{2}-\frac{10}{7}} & 0 \\ 0 & \frac{\frac{4}{3}-1}{\frac{3}{2}-\frac{4}{3}} \end{pmatrix} = \begin{pmatrix} 6 & 0 \\ 0 & 2 \end{pmatrix}.$$

进而

$$\begin{cases} \left(\nabla^2 Q^{[1]} + \tilde{\omega}_2\right)\omega_1 = 12 > 9 = \left(\nabla^2 f^{[1]} + \omega_2\right)\tilde{\omega}_1, \\ \kappa_1^{[1]} \leq \kappa^{[1]} \leq \kappa_2^{[1]} \end{cases}$$

以及

$$\begin{cases} \left(\nabla^2 Q^{[2]} + \tilde{\omega}_2\right)\omega_1 = 12 > 6 = \left(\nabla^2 f^{[2]} + \omega_2\right)\tilde{\omega}_1, \\ \kappa_1^{[2]} \leq \kappa^{[2]} \leq \kappa_2^{[2]}, \end{cases}$$

其中

$$\begin{cases} \kappa_1^{[1]} = \dfrac{\left(\nabla^2 f^{[1]} + \omega_2\right)\left[(-\rho+1)\left(\nabla^2 Q^{[1]} + \tilde{\omega}_2\right) + \tilde{\omega}_1\right]}{\left(\nabla^2 Q^{[1]} + \tilde{\omega}_2\right)\omega_1 - \left(\nabla^2 f^{[1]} + \omega_2\right)\tilde{\omega}_1} = 5, \\ \kappa_2^{[1]} = \dfrac{\left(\nabla^2 f^{[1]} + \omega_2\right)\left[(\rho+1)\left(\nabla^2 Q^{[1]} + \tilde{\omega}_2\right) + \tilde{\omega}_1\right]}{\left(\nabla^2 Q^{[1]} + \tilde{\omega}_2\right)\omega_1 - \left(\nabla^2 f^{[1]} + \omega_2\right)\tilde{\omega}_1} = 9, \end{cases}$$

$$\begin{cases} \kappa_1^{[2]} = \dfrac{\left(\nabla^2 f^{[2]} + \omega_2\right)\left[(-\rho+1)\left(\nabla^2 Q^{[2]} + \tilde{\omega}_2\right) + \tilde{\omega}_1\right]}{\left(\nabla^2 Q^{[2]} + \tilde{\omega}_2\right)\omega_1 - \left(\nabla^2 f^{[2]} + \omega_2\right)\tilde{\omega}_1} = \dfrac{5}{3}, \\ \kappa_2^{[2]} = \dfrac{\left(\nabla^2 f^{[2]} + \omega_2\right)\left[(\rho+1)\left(\nabla^2 Q^{[2]} + \tilde{\omega}_2\right) + \tilde{\omega}_1\right]}{\left(\nabla^2 Q^{[2]} + \tilde{\omega}_2\right)\omega_1 - \left(\nabla^2 f^{[2]} + \omega_2\right)\tilde{\omega}_1} = 3. \end{cases}$$

因此其满足充分条件. 此外, 我们有

$$\begin{aligned} & \tilde{\boldsymbol{x}}_2 - \boldsymbol{x}_2 \\ = \ & \tilde{\omega}_1 \left(\nabla^2 Q + \tilde{\omega}_2 \boldsymbol{I}\right)^{-1} \left(\tilde{\boldsymbol{x}}_1 - \boldsymbol{x}_0\right) - \omega_1 \left(\nabla^2 f + \omega_2 \boldsymbol{I}\right)^{-1} \left(\boldsymbol{x}_1 - \boldsymbol{x}_0\right) + \left(\tilde{\boldsymbol{x}}_1 - \boldsymbol{x}_1\right) \\ = \ & \begin{pmatrix} \frac{1}{56} \\ \frac{1}{24} \end{pmatrix}, \end{aligned}$$

因此

$$\left\|\tilde{\boldsymbol{x}}_2 - \boldsymbol{x}_2\right\|_2 = \frac{\sqrt{\frac{29}{2}}}{84} < \frac{1}{2}\frac{\sqrt{\frac{29}{2}}}{21} = \rho \left\|\tilde{\boldsymbol{x}}_1 - \boldsymbol{x}_1\right\|_2.$$

下面的例子展示了在维数 $n = 1$ 时的数值观察.

例 2.8. 我们尝试在数值上观察系数满足定理 2.29 中条件的概率, 并在维数 $n = 1$ 时进行说明. 我们使用软件 Mathematica 进行数值实验. 具体来说, 我们分别在 $[0, q\omega_1]$ 和 $[0, q\tilde{\omega}_1]$ 范围内对 ω_2 和 $\tilde{\omega}_2$ 进行积分, 其中 q 是非负实系数. 然后, 我们将结果除以 $q^2\omega_1\tilde{\omega}_1$ 来表示概率, 即

$$\begin{aligned} \mathrm{Prob}(\rho) = & \frac{1}{q^2\omega_1\tilde{\omega}_1} \int_0^{q\omega_1} \int_0^{q\tilde{\omega}_1} \mathrm{Boole}\left[\nabla^2 Q + \tilde{\omega}_2 \geq -\frac{(\kappa+1)\tilde{\omega}_1\left(\nabla^2 f + \omega_2\right)}{(\rho+1)\left(\omega_2 + \nabla^2 f\right) - \kappa\omega_1}\right] \\ & \mathrm{Boole}\left[\nabla^2 Q + \tilde{\omega}_2 \leq \frac{(\kappa+1)\tilde{\omega}_1\left(\nabla^2 f + \omega_2\right)}{(\rho-1)\left(\omega_2 + \nabla^2 f\right) + \kappa\omega_1}\right] d\tilde{\omega}_2 d\omega_2, \end{aligned}$$

其中 Boole(·) 表示 0/1 输出的布尔函数.

注意, 在这个例子中, 我们定义常数 $\nabla^2 Q = -1$, $\nabla^2 f = -2$, $\omega_1 = 3$, $\tilde{\omega}_1 = 3$, $\kappa = -2$, 以及 $q = 10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^2, 10^3$.

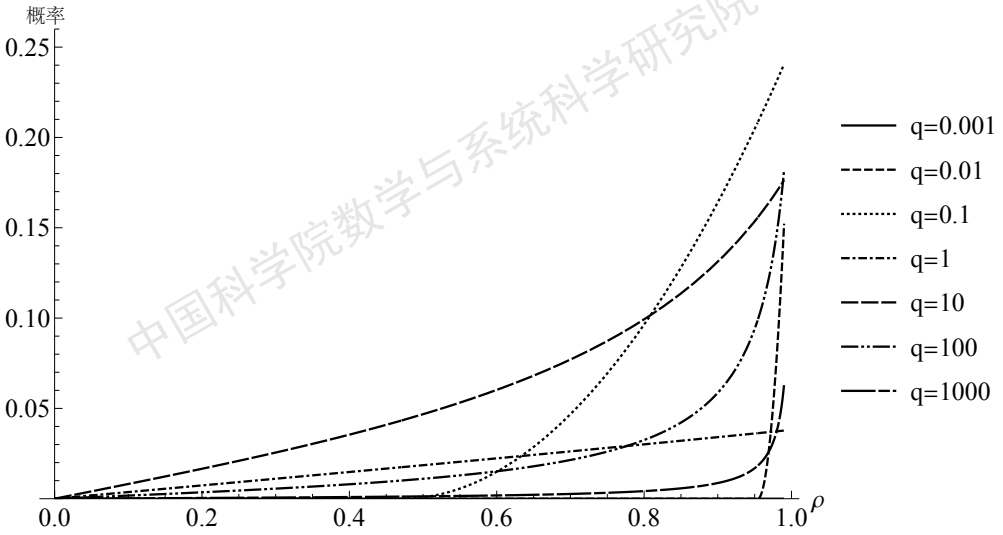


图 2-14 例 2.8 的数值结果

Figure 2-14 Numerical results for Example 2.8

图 2-14 展示了函数 Prob(ρ) 作为参数 ρ 的函数的数值结果. 不同的线对应参数 q 的不同取值. 从图 2-14 可以看出, 在这个 1 维例子中, 获得可以实现距离减小的系数 ω_2 和 $\tilde{\omega}_2$ 的概率最多约为 25%.

2.4.3 小结

本节分析了经过迭代后, 两个非凸二次函数在信赖域内的极小点之间的距离减小的充分条件. 注意, 二次函数通常用于在数值优化算法中对目标函数进行局部近似, 而在大多数非线性情况下我们无法获得准确的模型. 假如我们有不同的二次逼近模型函数, 本节的结果就能分析和提供一种减小不同模型函数的极小点之间距离的方法. 此外, 本节的例子表明, 在某些情况下, 上述的两个二次模型在信赖域内的极小点之间的距离在一次迭代后有较高的概率会增加, 考虑到这一点, 信赖域方法使用的二次模型在这种意义下的确应该在一次迭代后就更新, 即使模型是非凸的并且迭代步落在信赖域的边界.

第 3 章　带变换目标函数的无导数优化及基于最小 Frobenius 范数更新二次模型的算法

3.1　带变换目标函数的无导数优化

本章将重点讨论如何求解带有变换目标函数的无导数优化问题. 本章提出和研究的无约束无导数优化问题具有下面的一般形式

$$\min_{\boldsymbol{x}\in\mathfrak{R}^n} \ f(\boldsymbol{x}), \tag{3-1}$$

其中, 黑箱函数 f 可以在一次迭代/探测步中提供 m 个点的函数值, 探测方案在假设 3.1 中给出. 此外, 在相同的迭代/探测步中, m 个探测点共用目标函数的相同变换. 它将 f 变换为 $f_k := T_k \circ f$, 变换 T_k 仅依赖于当前 (第 k) 步, 定义如定义 3.2 所示. 注意, 我们最终想要极小化的仍然是原始目标函数 f.

假设 3.1 (带有变换目标函数的无导数优化的探测方案). 一次探测可以获取 m 个点对应的函数输出值, 这批探测点可以由优化算法选择. 图 3-1 展示了探测方案.

输入: $\boldsymbol{y}_1, \cdots, \boldsymbol{y}_m$ ⟶ 对黑箱的第 k 次探测 ⟶ 输出: $f_k(\boldsymbol{y}_1), \cdots, f_k(\boldsymbol{y}_m)$

图 3-1 带有变换目标函数的无导数优化的探测方案: 第 k 次探测, 探测点为 $\boldsymbol{y}_1, \cdots, \boldsymbol{y}_m$

Figure 3-1 Query oracle of derivative-free optimization with transformed objective functions: the k-th query, for the queried points $\boldsymbol{y}_1, \cdots, \boldsymbol{y}_m$

可以发现, 假设 3.1 中的探测方案有两个基本特征. 一个特征是同时探测一组点, 另一个特征是同时探测的点共用相同的变换. 注意, 在无导数优化中, 探测通常不能在很短的时间内或使用低成本完成, 因此我们也在多数场景中称其为昂贵优化. 这种探测方案通常与批量交互式探测或仿真机制对应. 此外, 我们将在第 3.1 节结尾给出更多的应用例子. 需要指出的是, 据我们所知, 尽管带有变换目标函数的无导数优化有广泛的应用, 但相关的概念和研究尚未得到深入和具体的探索. 为这类问题设计的算法, 特别是基于模型函数的算法, 是有限的. 本章旨在提出这类带有变换的问题, 并在使用基于欠定二次模型的算法求解它时给出一些初步结果. 我们还将回答最小 Frobenius 范数更新二次模型及相应算法如何受到变换的影响.

定义 3.2. *设 T 为从 $\mathfrak{R}$ 到 $\mathfrak{R}$ 的变换, 我们用函数 $T \circ f$ 表示变换后的函数, 它满足: 对于给定的函数 f 和任意的 $\boldsymbol{x} \in \mathfrak{R}^n$, $(T \circ f)(\boldsymbol{x}) = T(f(\boldsymbol{x}))$.*

下文讨论的每个变换都是从 $\mathfrak{R}$ 到 $\mathfrak{R}$ 的变换. 变换后的目标函数在随机、噪声或加密无导数/黑箱优化中起着重要作用[1]. 例如, 在加密黑箱优化中, 不同的变

[1]无约束加密黑箱优化问题的一般形式可以表述为 $\min_{\boldsymbol{x}\in\mathfrak{R}^n} f(\boldsymbol{x})$, 其中 f 是一个加密黑箱函数. f 的探测成本是昂贵的, 其输出值通过添加噪声被加密为 f_k.

换可以根据差分隐私理论被视为通过添加不同噪声形成的不同加密 [194–199]. 第 3.5.4 节将详细介绍求解一类特殊的加密工程设计优化问题的细节. 另一个例子是基于云的分布式优化问题, 这类问题旨在极小化本地和基于云的复合目标函数, 同时保护相应目标函数的隐私信息 [200]. 此外, 在个人健康领域也有相应的加密黑箱优化问题 [201]. 另外, Kusner 等人针对差分隐私贝叶斯优化进行了讨论 [202].

事实上, 带变换目标函数的无导数优化有多种应用, 加密黑箱优化只是其中一个实例. 例如, 带有系数随迭代而变化的正则化函数的问题属于带有变换目标函数的无导数优化问题. Grapiglia、Yuan 和 Yuan 提出了相关的复合非光滑优化的无导数信赖域算法 [120]. 对于极小化变换目标函数的讨论, 可以回顾 Deng 和 Ferris 的工作 [203], 他们的工作提供了关于使用算法 UOBYQA [137] 极小化随机目标函数的讨论.

除了刻画每次迭代时目标函数中存在变换或扰动时插值模型的变化, 本章还关注二次模型在信赖域内的极小点是否会改变.

注 3.1. 事实上, 一些无导数方法是不依赖函数值 (的绝对大小) 的, 例如, 协方差矩阵自适应进化策略 (CMA-ES) [131]、Nelder-Mead 方法 [51] (仅使用比较探测) 和其他仅使用布尔值函数比较的算法 [204]. 本章关注的是发展和改进基于模型的算法, 这些算法依赖于函数值来求解具有变换目标函数的问题. 我们将提供一个新的角度, 来刻画和理解在使用依赖函数值的无导数优化方法时变换对目标函数和模型及算法所产生的影响.

考虑使用基于最小 Frobenius 范数更新二次模型的无导数信赖域算法来求解提出的 DFOTO 问题, 本章将作出以下贡献. 我们将配合所提出的探测方案修改 Powell 的最小 Frobenius 范数更新二次模型的更新公式 [93] 以用于在求解带变换问题时使用基于模型的信赖域方法. 我们将分析极小化变换目标函数时的最小 Frobenius 范数更新插值模型. 我们将提出保最优性变换, 给出并证明这类变换的充分必要条件. 我们将讨论正单调变换. 给出仿射变换目标函数的最小 Frobenius 范数更新二次模型的解析表达式, 分析其插值误差, 并对此进行进一步的讨论. 本章将针对一阶临界点给出初步的收敛分析. 本章的数值结果将表明使用修改后的模型更新公式的必要性, 同时也说明我们的方法可以高效稳健地求解大多数测试问题和一个所展示的实际问题, 即便其中的变换会改变模型函数的最优性. 据我们所知, 这是首个使用需要利用函数值信息 (而非函数值的比较) 的基于模型的算法来极小化变换后的目标函数的工作. 值得注意的是, 当求解保序或保最优性变换问题时, 相应方法的表现令人满意.

本章的结构如下. 我们在第 3.2 节给出了我们的算法和探测方案, 其中包含了最小 Frobenius 范数更新二次模型、信赖域子问题以及与保最优性变换的相关定义. 证明了除平移变换外保模型最优性变换的存在性. 这类变换的充分必要条件也在该节给出. 在第 3.3 节, 我们介绍了正单调变换的性质, 特别是仿射变换. 我们发现具有 (非平凡的) 正乘法系数的仿射变换不是保模型最优性变换. 当目

标函数被仿射变换时, 我们给出了最小 Frobenius 范数更新二次模型函数的相应变换. 第 3.4 节展示了求解存在仿射变换的问题时完全线性模型插值误差的系数. 第 3.4 节还分析了在可证明的基于模型的无导数框架保证下, 算法极小化相应变换后目标函数时的收敛性. 第 3.5 节展示了极小化测试例子的数值结果, 以及求解一组随机仿射变换目标函数的测试问题的 Performance Profile 和 Sensitivity Profile. 这样的数值结果支持了我们的理论分析, 求解空间行波管加密工程最优设计问题的结果也显示了我们方法的实际优势. 在本章的最后, 我们提出了 “移动目标” 无导数优化问题.

3.2　算法、探测方案和保最优性变换

这里将介绍我们的算法和探测方案, 并详细介绍用于无导数优化的最小 Frobenius 范数更新二次模型. 此外, 我们将介绍基于模型的无导数算法中的信赖域子问题. 我们也将给出包括保最优性变换在内的一些基本概念.

3.2.1　基于模型的信赖域算法和探测方案

我们给出基于模型的信赖域算法和探测方案的一些细节, 这些将用于求解问题 (3-1). 我们还将解释我们选择这样的框架、插值模型函数和探测方案的原因.

基于模型的无导数信赖域算法 (针对变换后的目标函数) 的基本框架如算法 5 所示, 简明起见, 算法框架中省略了一些细节. 可以在 Conn、Scheinberg 和 Vicente 的著作中找到极小化原始目标函数的基于模型的无导数信赖域算法的算法框架 [20]. 如假设 3.1 和表 3-1 所示, 算法 5 中的探测是对一批共用一致变换的点进行的. 函数 $f_1, \cdots, f_k, \cdots$ 分别表示在第 $1, \cdots, k, \cdots$ 步中与变换 $\boldsymbol{T}_1, \cdots, \boldsymbol{T}_k, \cdots$ 对应的变换后目标函数. 在求解信赖域子问题时, 信赖域的中心通常设为 $\boldsymbol{x}_{\text{opt}}$, 这是当前迭代插值点中的最小函数值点. 为了给出更简明的算法框架, 我们使用 $\boldsymbol{x}_k$, 并且在 (3-10) 中将其写为 $\boldsymbol{x}_{\text{opt}}$. 当更新第 k 次插值集时, 我们通常会在这一步舍弃最差插值点 $\boldsymbol{y}_t$ 并将其替换为 $\boldsymbol{y}_{\text{new}}(= \boldsymbol{x}_{k-1} + \boldsymbol{d}_{k-1})$, 这是新加入的插值点. 算法 5 使用了基于模型的信赖域无导数方法中传统的 Λ-适定性, 相关的验证参考了基于模型的信赖域无导数方法的传统算法框架中的相同步骤 [20].

算法 5 用于极小化变换后目标函数的基于模型的无导数信赖域算法框架

给定一个初始点 $\boldsymbol{x}_{\text{int}}$ 和一个满足 $\boldsymbol{x}_{\text{int}} \in \mathcal{X}_1$ 的初始插值点集 $\mathcal{X}_1$, 以及

$$f_1(\boldsymbol{x}_{\text{int}}) = \min_{\boldsymbol{y} \in \mathcal{X}_1} f_1(\boldsymbol{y}).$$

选择初始信赖域半径 Δ_1. 令 $k = 1$.

步 1. (构造插值模型)

构造一个满足插值条件 $Q_k(\boldsymbol{y}) = f_k(\boldsymbol{y})$, $\boldsymbol{y} \in \mathcal{X}_k$ 的二次模型 Q_k.

步 2. (信赖域迭代)

求解信赖域子问题

$$\min_{\boldsymbol{d}\in\Re^n} Q_k(\boldsymbol{x}_k+\boldsymbol{d})$$

$$\text{s. t. } \|\boldsymbol{d}\|_2 \le \Delta_k$$

并获得其解 $\boldsymbol{d}_k$.

如果 $\boldsymbol{x}_k+\boldsymbol{d}_k$ 被接受, 例如 $f_{k+1}(\boldsymbol{x}_k+\boldsymbol{d}_k) < f_{k+1}(\boldsymbol{x}_k)$, 则令 $\boldsymbol{x}_{k+1}=\boldsymbol{x}_k+\boldsymbol{d}_k$; 否则, 令 $\boldsymbol{x}_{k+1}=\boldsymbol{x}_k$.

步 3. (管理插值点集)

检查插值点集是否适定. 如果必要, 执行模型改进步来提高插值点集的适定性. 更新插值点集为 $\mathcal{X}_{k+1}$ 以便使其包含 $\boldsymbol{x}_{k+1}$.

步 4. (更新)

根据 $\boldsymbol{d}_k$ 的表现和插值点集的适定程度更新信赖域半径以得到 Δ_{k+1}. 令 $k=k+1$. 转到**步 1**.

我们在这里介绍用于求解变换后目标函数问题的探测过程, 这将在后续讨论中使用. 我们同步探测前 m 个插值点. 一旦获得新的迭代点, 根据表 3-1 所示的过程更新插值点集[2], 并获取新插值点集中点的函数值探测. 注意每个探测集包含 m 个点. 探测集可以以不同方式更新, 例如, $\mathcal{X}_k := \mathcal{X}_{k-1}\backslash\{\arg\max_{\boldsymbol{y}\in\mathcal{X}_{k-1}} \|\boldsymbol{y}-\boldsymbol{x}_k\|_2\}\cup\{\boldsymbol{x}_k\}$ 或 $\mathcal{X}_k := \mathcal{X}_{k-1}\backslash\{\arg\max_{\boldsymbol{y}\in\mathcal{X}_{k-1}} f_{k-1}(\boldsymbol{y})\}\cup\{\boldsymbol{x}_k\}$.

表 3-1 求解带变换目标函数问题的算法中的函数值探测情况

Table 3-1 Query and evaluation in algorithms for solving problems with transformed objective functions

步	探测点集合	函数值探测集合
1	$\mathcal{X}_1$	$\{f_1(\boldsymbol{y}),\ \forall\, \boldsymbol{y}\in\mathcal{X}_1\}$
2	$\mathcal{X}_2$	$\{f_2(\boldsymbol{y}),\ \forall\, \boldsymbol{y}\in\mathcal{X}_2\}$
⋮	⋮	⋮
k	$\mathcal{X}_k$	$\{f_k(\boldsymbol{y}),\ \forall\, \boldsymbol{y}\in\mathcal{X}_k\}$
⋮	⋮	⋮

注意, 算法 5 中的插值模型函数是最小 Frobenius 范数更新二次模型. 接下来, 我们将给出在使用算法 5 时选择表 3-1 所示探测方案的原因. 首先, 可以在第 3.2.2 节中发现, 当我们基于最小 Frobenius 范数更新二次模型使用算法 5 求解带有变换目标函数的无导数优化问题时, 我们只需要改变插值方程右侧的向量 (细节见 (3-6)), 在这种情况下, 处理变换非常简单. 其次, 尽管不同迭代步的函数值输出的变换会发生变化, 但我们仍有理由信任在前一次迭代中已探测的点并依赖它们找到下一个迭代点. 否则, 如果算法完全重新搜索所有探测点, 我们的方法将出现迭代的不连续现象. 实际上, 我们的理论分析和数值结果也与上述论述吻合.

[2]我们称插值点集为 “插值集”.

3.2.2　变换后目标函数的最小 Frobenius 范数更新二次模型

事实上, 我们所讨论的这类变换问题的一个重要特征是相应的目标函数会随迭代而变换. 这里的 “随迭代而变换” 意味着目标函数的变换依赖于迭代/探测步骤. 因此, 对于处理带变换目标函数的无导数优化, 我们将给出相应的最小 Frobenius 范数更新二次模型函数. 新的更新公式应该在变换后的目标函数 f_k 随迭代次数 k 变化时保持其有效性. 注意, 本章分析的模型是最小 Frobenius 范数更新二次模型, 而不是最小 Frobenius 范数二次模型 [156, 160, 188], 后者在第 3.3 节中会有不同且更简单的结果. 高效稳健的数值表现激励着我们在存在变换的情况下探索我们所关注的这类模型的更多细节.

简明起见, 我们假设在第 k 次迭代中的适定的插值点集为 $\mathcal{X}_k = \{\boldsymbol{y}_1, \cdots, \boldsymbol{y}_m\}$. 变换函数 f_k 的二次模型 Q_k 通过求解子问题

$$\begin{aligned} &\min_{Q \in \mathcal{Q}} \ \left\|\nabla^2 Q - \nabla^2 Q_{k-1}\right\|_F^2 \\ &\text{s. t. } Q(\boldsymbol{y}) = f_k(\boldsymbol{y}), \ \forall\, \boldsymbol{y} \in \mathcal{X}_k \end{aligned} \tag{3-2}$$

得到, 我们定义 $D_k(\boldsymbol{x}) = Q_k(\boldsymbol{x}) - Q_{k-1}(\boldsymbol{x})$. 那么, 根据 (3-2), 我们可以通过求解子问题

$$\begin{aligned} &\min_{D \in \mathcal{Q}} \ \left\|\nabla^2 D\right\|_F^2 \\ &\text{s. t. } \begin{cases} D(\boldsymbol{y}_i) = f_k(\boldsymbol{y}_i) - f_{k-1}(\boldsymbol{y}_i), \ i = 1, \cdots, t-1, t+1, \cdots, m, \\ D(\boldsymbol{y}_{\text{new}}) = f_k(\boldsymbol{y}_{\text{new}}) - Q_{k-1}(\boldsymbol{y}_{\text{new}}), \end{cases} \end{aligned} \tag{3-3}$$

得到 $D_k(\boldsymbol{x})$, 这是因为根据基于模型的无导数信赖域算法的框架, 旧的 $\boldsymbol{y}_t$ 在当前 (第 k 次) 迭代中被舍弃并被 $\boldsymbol{y}_{\text{new}}$ 替换 [20]. 从 (3-2) 到 (3-3) 的推导是直接的, 事实上, 将函数 $Q(\boldsymbol{x}) - Q_{k-1}(\boldsymbol{x})$ 替换为 $D(\boldsymbol{x})$ 后即可.

设 $\lambda_j, j = 1, 2, \cdots, m$, 为优化问题 (3-3) 的 KKT 条件的 Lagrange 乘子, 正如 Powell [93] 指出的, 它们满足

$$\begin{cases} \displaystyle\sum_{j=1}^{m} \lambda_j = 0, \\ \displaystyle\sum_{j=1}^{m} \lambda_j \left(\boldsymbol{y}_j - \boldsymbol{x}_0\right) = \boldsymbol{0}_n, \\ \nabla^2 D_k = \displaystyle\sum_{j=1}^{m} \lambda_j \left(\boldsymbol{y}_j - \boldsymbol{x}_0\right)\left(\boldsymbol{y}_j - \boldsymbol{x}_0\right)^{\top}, \end{cases} \tag{3-4}$$

其中 $\boldsymbol{0}_n \in \mathfrak{R}^n$, $\boldsymbol{x}_0$ 是用以减小计算误差的基点, 其在一开始被设置为初始输入点. 二次函数 $D_k(\boldsymbol{x})$ 可以表示为

$$D_k(\boldsymbol{x}) = c + (\boldsymbol{x} - \boldsymbol{x}_0)^{\top} \boldsymbol{g} + \frac{1}{2} \sum_{j=1}^{m} \lambda_j \left((\boldsymbol{x} - \boldsymbol{x}_0)^{\top} (\boldsymbol{y}_j - \boldsymbol{x}_0)\right)^2, \ \boldsymbol{x} \in \mathfrak{R}^n. \tag{3-5}$$

在确定了参数 $\boldsymbol{\lambda}^{\top} = (\lambda_1, \cdots, \lambda_m)^{\top} \in \mathfrak{R}^m$、$c \in \mathfrak{R}$ 和 $\boldsymbol{g} \in \mathfrak{R}^n$ 后, 我们可以确定唯一的函数 $D_k(\boldsymbol{x})$, 从而得到新的二次模型函数 $Q_k(\boldsymbol{x})$. 很容易看出, $D(\boldsymbol{x})$ 的系数是线性方程组

$$\begin{pmatrix} \boldsymbol{A} & \boldsymbol{X} \\ \boldsymbol{X}^{\top} & \boldsymbol{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{\lambda} \\ c \\ \boldsymbol{g} \end{pmatrix} = \begin{pmatrix} \boldsymbol{r} \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

的解, 其中矩阵 $\boldsymbol{0} \in \mathfrak{R}^{(n+1)\times(n+1)}$ 是零矩阵. 矩阵 $\boldsymbol{A} \in \mathfrak{R}^{m\times m}$ 和 $\boldsymbol{X} \in \mathfrak{R}^{m\times(n+1)}$ 的元素分别为

$$\boldsymbol{A}_{ij} = \frac{1}{2}\left(\left(\boldsymbol{y}_i - \boldsymbol{x}_0\right)^{\top} \left(\boldsymbol{y}_j - \boldsymbol{x}_0\right) \right)^2$$

和

$$\boldsymbol{X} = \begin{pmatrix} 1 & \cdots & 1 \\ \boldsymbol{y}_1 - \boldsymbol{x}_0 & \cdots & \boldsymbol{y}_m - \boldsymbol{x}_0 \end{pmatrix}^{\top},$$

其中 $1 \le i, j \le m$. 向量 $\boldsymbol{r} \in \mathfrak{R}^m$ 的形式为

$$\boldsymbol{r} = \begin{pmatrix} f_k(\boldsymbol{y}_1) - f_{k-1}(\boldsymbol{y}_1) \\ \vdots \\ f_k(\boldsymbol{y}_{t-1}) - f_{k-1}(\boldsymbol{y}_{t-1}) \\ f_k(\boldsymbol{y}_{\text{new}}) - Q_{k-1}(\boldsymbol{y}_{\text{new}}) \\ f_k(\boldsymbol{y}_{t+1}) - f_{k-1}(\boldsymbol{y}_{t+1}) \\ \vdots \\ f_k(\boldsymbol{y}_m) - f_{k-1}(\boldsymbol{y}_m) \end{pmatrix}. \tag{3-6}$$

注 3.2. 向量 $\boldsymbol{r}$ 是极小化带变换目标函数和极小化不带变换目标函数的更新公式之间的主要区别. 这种形式非常自然, 同时对于获得带变换目标函数的最小 Frobenius 范数更新二次模型至关重要.

我们继续用 $\boldsymbol{W}$ 表示 KKT 矩阵, 记

$$\boldsymbol{W} = \begin{pmatrix} \boldsymbol{A} & \boldsymbol{X} \\ \boldsymbol{X}^{\top} & \boldsymbol{0} \end{pmatrix}.$$

如果 $\boldsymbol{W}$ 是可逆的, 可以通过

$$\begin{pmatrix} \boldsymbol{\lambda} \\ c \\ \boldsymbol{g} \end{pmatrix} = \boldsymbol{V} \begin{pmatrix} \boldsymbol{r} \\ 0 \\ \vdots \\ 0 \end{pmatrix} \tag{3-7}$$

来获得 $\boldsymbol{\lambda}, c, \boldsymbol{g}$, 其中 $\boldsymbol{V} = \boldsymbol{W}^{-1}$. $\boldsymbol{W}$ 的可逆性取决于插值点的位置, 这与集合的适定性有关, 这一点已被 Powell [174] 深入讨论. 初始插值点保证了初始 $\boldsymbol{W}$ 的可逆性, 而迭代 $\boldsymbol{W}$ 的可逆性以及公式 (3-7) 的数值精确性将通过在模型改进步中选择合适的插值点来迭代地保证. 这部分与 Powell 的工作 [94, 174] 中的讨论和方法相同. 在下面的讨论中, 我们假设矩阵 $\boldsymbol{W}$ 是可逆的.

注 3.3. 如果我们通过求解问题

$$\begin{aligned} &\min_{Q\in\mathcal{Q}} \left\|\nabla^2 Q - \nabla^2 Q_{k-1}\right\|_F^2 + \sigma \left\|\nabla Q - \nabla Q_{k-1}\right\|_2^2 \\ &\text{s. t. } Q(\boldsymbol{y}) = f_k(\boldsymbol{y}),\ \forall\, \boldsymbol{y}\in\mathcal{X}_k \end{aligned} \tag{3-8}$$

获得第 k 个模型函数, 其中权重系数 $\sigma \geq 0$, 那么上述结果和分析仍然成立, 唯一的区别是 $\boldsymbol{W}$ 变为矩阵

$$\boldsymbol{W} = \begin{pmatrix} \boldsymbol{A} & & \boldsymbol{X} \\ \boldsymbol{X}^\top & 0 & \boldsymbol{0}_n^\top \\ & \boldsymbol{0}_n & -\frac{\sigma}{2}\boldsymbol{I} \end{pmatrix}, \tag{3-9}$$

其中 $\boldsymbol{I}\in\mathfrak{R}^{n\times n}$ 是单位矩阵. 事实上, 本章的结论也适用于类似的其他范数意义下的最小范数更新二次模型, 包括第 2 章所给出的最小 $\boldsymbol{H}^2$ 范数更新二次模型和考虑了模型函数最优性及前一信赖域迭代性质的模型.

我们可以使用 Powell [174] 给出的矩阵 $\boldsymbol{V}$ 的更新公式. 最终, 我们可以获得形如 (3-5) 的 $\boldsymbol{D}_k(\boldsymbol{x})$, 其参数 $\boldsymbol{\lambda}$、c 和 $\boldsymbol{g}$ 由 (3-7) 给出, $\boldsymbol{r}$ 由 (3-6) 给定. 进而我们可以获得第 k 个模型 $Q_k = Q_{k-1} + D_k$.

3.2.3　信赖域子问题

基于模型的无导数信赖域算法通过求解当前二次模型函数的信赖域子问题来计算试探步, 即求解

$$\begin{aligned} &\min_{\boldsymbol{d}\in\mathfrak{R}^n} Q_k(\boldsymbol{x}_{\text{opt}} + \boldsymbol{d}) \\ &\text{s. t. } \|\boldsymbol{d}\|_2 \leq \Delta_k. \end{aligned} \tag{3-10}$$

在 (3-10) 中, $\boldsymbol{x}_{\text{opt}}$ 表示第 k 次插值集 $\mathcal{X}_k$ 中具有最优函数输出值的插值点.

这类算法的框架细节可以在 Conn、Scheinberg 和 Vicente 的著作 [20] 中看到. 求解相应基于模型的无导数信赖域算法中的信赖域二次模型子问题的子程序的一个终止条件是 $\|\boldsymbol{d}_k\|_2 < \hat{\rho}_k$, 其中 $\boldsymbol{d}_k$ 是信赖域子问题 (3-10) 的解, 而这里的参数 $\hat{\rho}_k$ 是信赖域半径的下界, 它被用来保持插值点之间有足够的距离, 这里不再赘述.

需要注意的是, 对于不带变换的无导数优化问题, 目标函数本身不会改变, 在迭代增加时只有信赖域发生变化. 然而, 在带变换的无导数优化问题中, 当迭代增加时, f_k 和信赖域都会改变, 二次模型 Q_k 不断更新以逼近 f_k, 进而子问题的解 $\boldsymbol{d}_k$ 可能根据其定义而改变. 在这种情况下, 条件 $\|\boldsymbol{d}_k\|_2 < \hat{\rho}_k$ 可能根本无法满足, 这样就会影响算法的终止. 如果终止条件不满足, 算法迭代就不能离开求解信赖域子问题的循环. 迭代次数可能会增长, 这意味着函数值探测的成本很高. 此外, 方法的模型改进步几乎无法被调用, 进而算法不能有效地减小模型的插值误差.

3.2.4　保最优性变换

在这部分, 我们将给出保最优性变换的分析和讨论.

给定一个黑箱函数 $f:\mathfrak{R}^n\to\mathfrak{R}$, 若一个二次函数 Q 满足 $Q(\boldsymbol{x})=f(\boldsymbol{x})$, $\forall\,\boldsymbol{x}\in\mathcal{X}$, 则称其为 f 在插值集 $\mathcal{X}\subset\mathfrak{R}^n$ 上的二次插值模型. 我们应注意, 下文讨论的每个变换都是从 $\mathfrak{R}$ 到 $\mathfrak{R}$ 的变换. 为了引出更多细节, 我们先给出以下定义.

定义 3.3 (在 $\mathcal{X}$ 上基于 Q_α 的函数 h 的最小 Frobenius 范数更新二次模型)**.** 给定函数 $h:\mathfrak{R}^n\to\mathfrak{R}$, 二次函数 Q_α 和适定集合 $\mathcal{X}\subset\mathfrak{R}^n$, 其中 $n+1\le|\mathcal{X}|<\frac{1}{2}(n+1)(n+2)$, 我们称一个二次模型函数是在 $\mathcal{X}$ 上基于 Q_α 的函数 h 的最小 *Frobenius* 范数更新二次模型, 如果它是

$$\begin{aligned}&\min_{Q\in\mathcal{Q}}\ \left\|\nabla^2Q-\nabla^2Q_\alpha\right\|_F^2\\&\text{s. t. }Q(\boldsymbol{y})=h(\boldsymbol{y}),\ \forall\,\boldsymbol{y}\in\mathcal{X}\end{aligned}\tag{3-11}$$

的解. 我们利用映射 $\mathcal{M}_{Q_\alpha}^{\mathcal{X}}$ 表示上述模型, 也即把 *(3-11)* 的解表示为 $\mathcal{M}_{Q_\alpha}^{\mathcal{X}}(h)$.

定义 3.4 (具有信赖域半径 Δ 的 Q 的子问题)**.** 给定点[3] $\boldsymbol{x}_{\text{opt}}\in\mathfrak{R}^n$、二次函数 Q 以及 $\Delta\in\mathfrak{R}$, 我们称问题

$$\begin{aligned}&\min_{\boldsymbol{d}\in\mathfrak{R}^n}\ Q(\boldsymbol{x}_{\text{opt}}+\boldsymbol{d})\\&\text{s. t. }\|\boldsymbol{d}\|_2\le\Delta\end{aligned}$$

为具有信赖域半径 Δ 并以 $\boldsymbol{x}_{\text{opt}}$ 为中心的 Q 的子问题. 注意, 如果没有必要强调中心, 我们将在相应叙述中省略“中心”一词.

我们给出保模型最优性变换的定义.

定义 3.5 (保模型最优性变换)**.** 假设适定集合 $\mathcal{X}=\{\boldsymbol{y}_1,\cdots,\boldsymbol{y}_m\}\subset\mathfrak{R}^n$, 且 Q_α 是一个二次函数. 如果一个变换 T 使得在 $\mathcal{X}$ 上基于 Q_α 的函数 f 的最小 *Frobenius* 范数更新二次模型的子问题的解与在 $\mathcal{X}$ 上基于 Q_α 的函数 $T\circ f$ 的最小 *Frobenius* 范数更新二次模型的子问题的解相同, 则称这样的 T 为具有信赖域半径 Δ 的保模型最优性变换. 也即, 给定点 $\boldsymbol{x}_{\text{opt}}\in\mathfrak{R}^n$, 若

$$\arg\min_{\|\boldsymbol{d}\|_2\le\Delta}\ Q_{\text{orig}}(\boldsymbol{x}_{\text{opt}}+\boldsymbol{d})=\arg\min_{\|\boldsymbol{d}\|_2\le\Delta}\ Q_{\text{trans}}(\boldsymbol{x}_{\text{opt}}+\boldsymbol{d}),$$

其中

$$\begin{aligned}Q_{\text{orig}}&:=\mathcal{M}_{Q_\alpha}^{\mathcal{X}}(f),\\Q_{\text{trans}}&:=\mathcal{M}_{Q_\alpha}^{\mathcal{X}}(T\circ f),\end{aligned}$$

则称变换 T 为具有信赖域半径 Δ 的保模型最优性变换.

假设 3.6. 给定函数 $f:\mathfrak{R}^n\to\mathfrak{R}$, $\boldsymbol{x}_{\text{opt}}\in\mathfrak{R}^n$, 一个二次函数 Q_α 和适定的插值集 $\mathcal{X}=\{\boldsymbol{y}_1,\cdots,\boldsymbol{y}_m\}\subset\mathfrak{R}^n$, 其中 $n+1\le|\mathcal{X}|=m<\frac{1}{2}(n+1)(n+2)$, 我们假设 $\boldsymbol{d}^*\in\mathfrak{R}^n$ 是在 $\mathcal{X}$ 上基于二次函数 Q_α 的函数 f 的最小 Frobenius 范数更新二次模型的具有信赖域半径 Δ 的子问题的解, 其中 $\|\boldsymbol{d}^*\|_2<\Delta$, 同时假定该模型是严格凸的.

[3] 点 $\boldsymbol{x}_{\text{opt}}$ 通常设置为插值集中具有最优函数值的插值点.

我们尝试给出包括保模型最优性变换的充分必要条件在内的一些理论结果.

定理 3.7. *假定假设 3.6 成立. 那么变换 T 是一个保模型最优性变换当且仅当 $(T(f(\boldsymbol{y}_1)),\cdots,T(f(\boldsymbol{y}_m)))^{\top}$ 是线性方程组*

$$\begin{aligned}&\sum_{j=1}^{m}\left(\left(\boldsymbol{y}_j-\boldsymbol{x}_{\text{opt}}\right)\left(\boldsymbol{y}_j-\boldsymbol{x}_{\text{opt}}\right)^{\top}\boldsymbol{d}^*\right)\boldsymbol{V}_j\begin{pmatrix}T(f(\boldsymbol{y}_1))-Q_\alpha(\boldsymbol{y}_1)\\ \vdots\\ T(f(\boldsymbol{y}_m))-Q_\alpha(\boldsymbol{y}_m)\\ 0\\ \vdots\\ 0\end{pmatrix}+\nabla^2 Q_\alpha\boldsymbol{d}^*\\ =&-\begin{pmatrix}\boldsymbol{V}_{m+2}\\ \vdots\\ \boldsymbol{V}_{m+n+1}\end{pmatrix}\begin{pmatrix}T(f(\boldsymbol{y}_1))-Q_\alpha(\boldsymbol{y}_1)\\ \vdots\\ T(f(\boldsymbol{y}_m))-Q_\alpha(\boldsymbol{y}_m)\\ 0\\ \vdots\\ 0\end{pmatrix}-\nabla Q_\alpha(\boldsymbol{x}_{\text{opt}})\end{aligned}\tag{3-12}$$

的解, 其中 $\boldsymbol{V}$ 是 KKT 矩阵的逆矩阵, 这里的 $\boldsymbol{V}_j$ 表示 $\boldsymbol{V}$ 的第 j 行.

证明. 假设函数 $T\circ f$ 在 $\mathcal{X}$ 上基于 Q_α 的最小 Frobenius 范数更新二次模型是

$$Q_u(\boldsymbol{x})=Q_\alpha(\boldsymbol{x})+c_u+\left(\boldsymbol{x}-\boldsymbol{x}_{\text{opt}}\right)^{\top}\boldsymbol{g}_u+\frac{1}{2}\sum_{j=1}^{m}(\boldsymbol{\lambda}_u)_j\left(\left(\boldsymbol{x}-\boldsymbol{x}_{\text{opt}}\right)^{\top}\left(\boldsymbol{y}_j-\boldsymbol{x}_{\text{opt}}\right)\right)^2.$$

我们可以得到

$$\begin{pmatrix}\boldsymbol{\lambda}_u\\ c_u\\ \boldsymbol{g}_u\end{pmatrix}=\boldsymbol{V}\begin{pmatrix}T(f(\boldsymbol{y}_1))-Q_\alpha(\boldsymbol{y}_1)\\ \vdots\\ T(f(\boldsymbol{y}_m))-Q_\alpha(\boldsymbol{y}_m)\\ 0\\ \vdots\\ 0\end{pmatrix},$$

简明起见, KKT 矩阵中出现的基点 $\boldsymbol{x}_0$ 被设置为 $\boldsymbol{x}_{\text{opt}}$. 我们知道 $\boldsymbol{d}^*$ 是可行的, 并且满足 $\nabla^2 Q_u\boldsymbol{d}^*=-\boldsymbol{g}_u$, 这是因为 $\|\boldsymbol{d}^*\|_2<\Delta$. 结合 (3-4) 和 $\boldsymbol{V}$ 的定义, 我们即可得到保模型最优性变换的上述充分必要条件. □

注 3.4. 假定假设 3.6 成立. 对于任意的 $c_2\in\mathfrak{R}$, $(f(\boldsymbol{y}_1)+c_2,\cdots,f(\boldsymbol{y}_m)+c_2)^{\top}$ 是线性方程组 (3-12) 的解, 因此, 满足 $T\circ f=f+c_2$ 的平移变换 T 是一个保模型最优性变换. 这个结论与推论 3.14 是一致的. 此外, 如果 $|\mathcal{X}|\geq n+2$, 根据定理 3.7, 存在更多其他保模型最优性变换.

(3-12) 的解空间包含一个平移的线性空间, 其维数至少为 $m-n$. Powell [94] 在 NEWUOA 中应用的最小 Frobenius 范数更新二次模型要求 $m \geq n+2$. 对于注 3.4, 如果 $n+2 \leq m < \frac{1}{2}(n+1)(n+2)$, 保模型最优性变换可以是除了满足

$$\begin{pmatrix} T(f(\boldsymbol{y}_1)) \\ \vdots \\ T(f(\boldsymbol{y}_m)) \end{pmatrix} = \begin{pmatrix} f(\boldsymbol{y}_1) + c_2 \\ \vdots \\ f(\boldsymbol{y}_m) + c_2 \end{pmatrix} \tag{3-13}$$

的变换以外的其他变换, 对于任意 $c_2 \in \mathfrak{R}$. 实际上, 为了获得完全线性性质, Conn、Scheinberg 和 Vicente 的著作 [20] 指出至少需要 $n+1$ 个插值点. 在下文中, 我们在进一步讨论之前给出一个自然的假设.

假设 3.8. 假设关于 $(T(f(\boldsymbol{y}_1)), \cdots, T(f(\boldsymbol{y}_m)))^{\mathsf{T}}$ 的 (3-12) 的齐次线性方程是线性独立的.

然后我们得到以下推论.

推论 3.9. *假定假设 3.6 和假设 3.8 成立. 若 $m = n+1$, 则变换 T 是一个保模型最优性变换当且仅当它满足 (3-13).*

证明. 当假设 3.8 成立时, 如果 $m = n+1$, 则 (3-12) 的解空间的维数是 1. 因此结论得证. □

注 3.5. 如果用于获取二次模型函数的子问题被选择为 (3-8), 则定理 3.7、推论 3.9 以及上述分析仍然适用, 相应的矩阵 $\boldsymbol{V}$ 为 (3-9) 中 KKT 矩阵的逆矩阵. 此外, 当 $m \leq n$ 且假设 3.6 和假设 3.8 的其余部分成立时, 变换 T 是一个保模型最优性变换当且仅当它满足

$$\begin{pmatrix} T(f(\boldsymbol{y}_1)) \\ \vdots \\ T(f(\boldsymbol{y}_m)) \end{pmatrix} = \begin{pmatrix} f(\boldsymbol{y}_1) \\ \vdots \\ f(\boldsymbol{y}_m) \end{pmatrix}.$$

考虑到如果 $m \leq n$, (3-12) 的解是唯一的, 以上结论直接成立.

一个保模型最优性变换的例子如下:

例 3.1. 假设原始黑箱目标函数为

$$f(x, y) = \frac{1}{2}\left((x-y)^2 + (x-1)^2 + (y-1)^2\right),$$

注意这里的 x 和 y 表示 2 维变量的分量. 此外, 基点 $\boldsymbol{x}_0$ 和初始插值点 $\boldsymbol{y}_1$, $\boldsymbol{y}_2$, $\boldsymbol{y}_3$, $\boldsymbol{y}_4$, $\boldsymbol{y}_5$ 是

$$\boldsymbol{x}_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix},\ \boldsymbol{y}_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix},\ \boldsymbol{y}_2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix},\ \boldsymbol{y}_3 = \begin{pmatrix} 0 \\ 1 \end{pmatrix},\ \boldsymbol{y}_4 = \begin{pmatrix} -1 \\ 0 \end{pmatrix},\ \boldsymbol{y}_5 = \begin{pmatrix} 0 \\ -1 \end{pmatrix}.$$

在这个例子中, 我们将信赖域半径设置为 10. 有

$$f(\boldsymbol{y}_1) = 1,\ f(\boldsymbol{y}_2) = 1,\ f(\boldsymbol{y}_3) = 1,\ f(\boldsymbol{y}_4) = 3,\ f(\boldsymbol{y}_5) = 3,\ \boldsymbol{x}_{\text{opt}}^{(1)} = \boldsymbol{y}_1,$$

在计算了 KKT 矩阵的逆矩阵 $\boldsymbol{V}$ 之后, 我们可以得到

$$\boldsymbol{\lambda} = \begin{pmatrix} -4 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \ c = 1, \ \boldsymbol{g} = \begin{pmatrix} -1 \\ -1 \end{pmatrix},$$

以及

$$Q_1(x, y) = 1 - x - y + x^2 + y^2.$$

然后我们将模型在信赖域中的极小点, 即 $\boldsymbol{y}_{\text{new}} = (\frac{1}{2}, \frac{1}{2})^{\mathsf{T}}$, 加入插值集, 并舍弃点 $\boldsymbol{y}_5$. 此外, 我们知道 $\boldsymbol{x}_{\text{opt}}^{(2)} = \boldsymbol{y}_{\text{new}}$, 简明起见, 我们将基点 $\boldsymbol{x}_0$ 设置为 $\boldsymbol{x}_{\text{opt}}^{(2)}$. 可以获取新的 KKT 矩阵的逆矩阵 $\boldsymbol{V}_{\text{new}}$, 我们有 $f(\boldsymbol{y}_{\text{new}}) = \frac{1}{4}, Q_1(\boldsymbol{y}_{\text{new}}) = \frac{1}{2}$, 继而我们可以得到

$$\boldsymbol{\lambda}^+ = \begin{pmatrix} \frac{2}{3} \\ 1 \\ \frac{4}{3} \\ -\frac{1}{3} \\ -\frac{8}{3} \end{pmatrix}, \ c^+ = -\frac{1}{4}, \ \boldsymbol{g}^+ = \begin{pmatrix} -\frac{1}{3} \\ -\frac{1}{3} \end{pmatrix},$$

$$D(x, y) = -\frac{2}{3}xy + \frac{1}{3}y^2 - \frac{1}{3}y,$$

以及

$$Q_2(x, y) = Q_1(x, y) + D(x, y) = x^2 - \frac{2}{3}xy - x + \frac{4}{3}y^2 - \frac{4}{3}y + 1.$$

进而我们得到模型函数在信赖域 $\{\boldsymbol{x} : \|\boldsymbol{x} - \boldsymbol{x}_{\text{opt}}^{(2)}\|_2 \leq 10\}$ 中的极小点, 即 $\boldsymbol{d}^* = (\frac{5}{22}, \frac{2}{11})^{\mathsf{T}}$, 接下来, 计算得到下一个迭代 (插值) 点是 $(\frac{8}{11}, \frac{15}{22})^{\mathsf{T}}$. 我们将 $\boldsymbol{d}^*$ 代入方程 (3-12), 得到充分必要条件

$$\begin{aligned} T(f(\boldsymbol{y}_4)) &= 2 + \frac{9}{10}T(f(\boldsymbol{y}_1)) - \frac{27}{5}T(f(\boldsymbol{y}_2)) + \frac{11}{2}T(f(\boldsymbol{y}_3)), \\ T(f(\boldsymbol{y}_{\text{new}})) &= -\frac{3}{4} + \frac{33}{40}T(f(\boldsymbol{y}_1)) + \frac{21}{20}T(f(\boldsymbol{y}_2)) - \frac{7}{8}T(f(\boldsymbol{y}_3)), \end{aligned} \tag{3-14}$$

其解空间是

$$\begin{pmatrix} T(f(\boldsymbol{y}_1)) \\ T(f(\boldsymbol{y}_2)) \\ T(f(\boldsymbol{y}_3)) \\ T(f(\boldsymbol{y}_4)) \\ T(f(\boldsymbol{y}_{\text{new}})) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 2 \\ -\frac{3}{4} \end{pmatrix} + k_1 \begin{pmatrix} 40 \\ 0 \\ 0 \\ 36 \\ 33 \end{pmatrix} + k_2 \begin{pmatrix} 0 \\ 20 \\ 0 \\ -108 \\ 21 \end{pmatrix} + k_3 \begin{pmatrix} 0 \\ 0 \\ 8 \\ 44 \\ -7 \end{pmatrix},$$

其中 $k_1, k_2, k_3 \in \mathfrak{R}$. 我们可以看到, 其包含了平移变换 (对应满足 $k_2 = 2k_1, k_3 = 5k_1$ 的常数). 注意, 原始函数值 $f(\boldsymbol{y}_1), f(\boldsymbol{y}_2), f(\boldsymbol{y}_3), f(\boldsymbol{y}_4), f(\boldsymbol{y}_{\text{new}})$ 也满足 (3-14).

在图 3-2 中, 上方包含了第一次迭代时的迭代/插值点和原始目标函数值. 下方包含了第二次迭代时的迭代/插值点以及通过保模型最优性变换变换后的目标函数值.

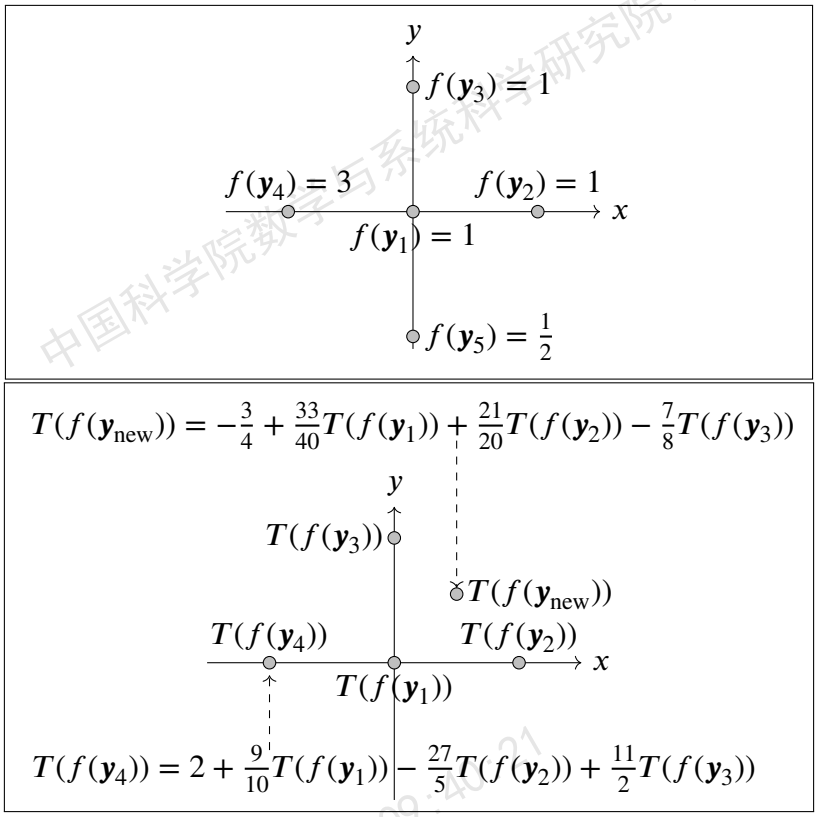


图 3-2 例 3.1 中的保模型最优性变换

Figure 3-2 Model optimality-preserving transformations in Example 3.1

3.3 正单调变换和仿射变换

请注意, $\boldsymbol{Q}_k$ 对应的信赖域子问题的解 $\boldsymbol{d}_k$ 是 f_k 对应的子问题解的近似, 这是因为 $\boldsymbol{Q}_k$ 是 f_k 的局部二次插值模型函数. 因此, 我们在下面给出保目标函数最优性变换的定义.

定义 3.10 (保目标函数最优性变换)**.** *若目标函数 f 的子问题解在信赖域半径 Δ 下与 $T\circ f$ 的子问题解相同, 则称变换 T 是在信赖域半径 Δ 下的保目标函数最优性变换. 即, 给定点 $\boldsymbol{x}_{\text{opt}}\in\mathfrak{R}^n$, 若我们有*

$$\arg\min_{\|\boldsymbol{d}\|_2\le\Delta} f(\boldsymbol{x}_{\text{opt}}+\boldsymbol{d})=\arg\min_{\|\boldsymbol{d}\|_2\le\Delta}(T\circ f)(\boldsymbol{x}_{\text{opt}}+\boldsymbol{d}),$$

则称变换 T 是信赖域半径 Δ 对应的保目标函数最优性变换.

本节将展示一些基本变换下的目标函数和相应的最小 Frobenius 范数更新二次模型函数. 我们首先给出正单调变换的定义.

定义 3.11. *若变换 $T:\mathfrak{R}\to\mathfrak{R}$ 保持数值大小的顺序, 即: 对于 $\theta_1>\theta_2$ 有 $T(\theta_1)>T(\theta_2)$, 对于 $\theta_1=\theta_2$ 有 $T(\theta_1)=T(\theta_2)$, 则我们称 T 是一个正单调变换.*

我们可以直接得到以下命题.

命题 3.12. *若变换 T 是一个正单调变换, 则 T 在任意信赖域半径下都是一个保目标函数最优性变换.*

证明. 结论可以直接由定义 3.11 得到. □

正单调变换可以是任意严格单调递增的函数, 例如具有正系数 (乘法系数) 的线性函数、指数函数和具有正奇次幂的幂函数. 这里, 我们给出一个最简单的例子: 仿射变换.

例 3.2. 一个满足 $T \circ f = c_1 f + c_2$ 的仿射变换 T, 其中常数 $c_1, c_2 \in \mathfrak{R}$, $c_1 > 0$, 是一个正单调变换.

我们可以看到, 即使目标函数 f 在某一步被仿射变换为 $c_1 f + c_2$, 其中 $c_1 > 0$, 其最小 Frobenius 范数更新二次模型也不一定能通过与原始目标函数相同的变换获得. 换言之, 在前一节中, 我们可以发现仿射变换通常不是保模型最优性变换. 然而, 目标函数在输出前被仿射变换的情况是基本且具有实际意义的. 因此, 我们将进一步讨论仿射变换后的目标函数、其模型函数的解析表达式以及插值模型的完全线性常数. 此外, 我们将给出相应的数值实验, 并尝试将其作为保目标函数最优性变换的典型例子来测试和展示我们方法的数值效果.

我们给出以下定理来获得仿射变换后的目标函数对应模型的表达式. 简明起见, 本章的其余部分用 $\boldsymbol{y}_t$ 表示 $\boldsymbol{y}_{\text{new}}$, 这是因为 $\boldsymbol{y}_{\text{new}}$ 在获得第 k 个模型之前已被放置在插值集的第 t 个位置.

定理 3.13. *假设 Q_α 是一个二次函数, 并且 $\mathcal{X} = \{\boldsymbol{y}_1, \cdots, \boldsymbol{y}_m\} \subset \mathfrak{R}^n$ 是一个适定集合, 其中 $n+1 \le m < \frac{1}{2}(n+1)(n+2)$. 则对于 $c_1, c_2 \in \mathfrak{R}$, 我们有*

$$\mathcal{M}^{\mathcal{X}}_{Q_\alpha}(c_1 f + c_2) = \left(c_1 \mathcal{M}^{\mathcal{X}}_{Q_\alpha}(f) + c_2\right) + \left(c_1 - 1\right)\left(\mathcal{M}^{\mathcal{X}}_0(Q_\alpha) - Q_\alpha\right), \tag{3-15}$$

其中 $\mathcal{M}^{\mathcal{X}}_0(Q_\alpha)$ 表示在 $\mathcal{X}$ 上基于零函数的 Q_α 的最小 Frobenius 范数更新二次模型, 即 Q_α 的最小 Frobenius 范数二次模型.

证明. 设 $Q_\beta := \mathcal{M}^{\mathcal{X}}_{Q_\alpha}(f)$, $\hat{Q}_\beta := \mathcal{M}^{\mathcal{X}}_{Q_\alpha}(c_1 f + c_2)$ 以及 $\tilde{Q} := \mathcal{M}^{\mathcal{X}}_0(Q_\alpha)$. 设 $D_\beta := Q_\beta - Q_\alpha$, $\hat{D}_\beta := \hat{Q}_\beta - Q_\alpha$. 则二次函数 D_β 是问题

$$\begin{aligned} &\min_{D \in Q} \ \left\|\nabla^2 D\right\|_F^2 \\ &\text{s. t. } D(\boldsymbol{y}) = f(\boldsymbol{y}) - Q_\alpha(\boldsymbol{y}),\ \forall\, \boldsymbol{y} \in \mathcal{X} \end{aligned}$$

的解, 二次函数 $\hat{D}_\beta$ 是问题

$$\begin{aligned} &\min_{D \in Q} \ \left\|\nabla^2 D\right\|_F^2 \\ &\text{s. t. } D(\boldsymbol{y}) = c_1 f(\boldsymbol{y}) + c_2 - Q_\alpha(\boldsymbol{y}),\ \forall\, \boldsymbol{y} \in \mathcal{X} \end{aligned}$$

的解.

我们分别用 $\boldsymbol{\lambda}_D \in \mathfrak{R}^m, c_D \in \mathfrak{R}, \boldsymbol{g}_D \in \mathfrak{R}^n$ 和 $\boldsymbol{\lambda}_{\hat{D}} \in \mathfrak{R}^m, c_{\hat{D}} \in \mathfrak{R}, \boldsymbol{g}_{\hat{D}} \in \mathfrak{R}^n$ 表示二次函数 $\boldsymbol{D}_\beta$ 和 $\hat{\boldsymbol{D}}_\beta$ 对应的参数. 此外, $(\boldsymbol{\lambda}_D^\top, c_D, \boldsymbol{g}_D^\top)^\top$ 和 $(\boldsymbol{\lambda}_{\hat{D}}^\top, c_{\hat{D}}, \boldsymbol{g}_{\hat{D}}^\top)^\top$ 共用同一个 KKT 矩阵的逆矩阵 $\boldsymbol{V}$, 即有

$$\begin{pmatrix} \boldsymbol{\lambda}_D \\ c_D \\ \boldsymbol{g}_D \end{pmatrix} = \boldsymbol{V} \begin{pmatrix} f(\boldsymbol{y}_1) - Q_\alpha(\boldsymbol{y}_1) \\ \vdots \\ f(\boldsymbol{y}_m) - Q_\alpha(\boldsymbol{y}_m) \\ 0 \\ \vdots \\ 0 \end{pmatrix},$$

$$\begin{pmatrix} \boldsymbol{\lambda}_{\hat{D}} \\ c_{\hat{D}} \\ \boldsymbol{g}_{\hat{D}} \end{pmatrix} = \boldsymbol{V} \begin{pmatrix} c_1 f(\boldsymbol{y}_1) + c_2 - Q_\alpha(\boldsymbol{y}_1) \\ \vdots \\ c_1 f(\boldsymbol{y}_m) + c_2 - Q_\alpha(\boldsymbol{y}_m) \\ 0 \\ \vdots \\ 0 \end{pmatrix},$$

我们可以直接得到

$$\begin{pmatrix} \boldsymbol{\lambda}_{\hat{D}} \\ c_{\hat{D}} \\ \boldsymbol{g}_{\hat{D}} \end{pmatrix} = c_1 \begin{pmatrix} \boldsymbol{\lambda}_D \\ c_D \\ \boldsymbol{g}_D \end{pmatrix} + \boldsymbol{V} \begin{pmatrix} c_2 \\ \vdots \\ c_2 \\ 0 \\ \vdots \\ 0 \end{pmatrix} + c_1 \boldsymbol{V} \begin{pmatrix} Q_\alpha(\boldsymbol{y}_1) \\ \vdots \\ Q_\alpha(\boldsymbol{y}_m) \\ 0 \\ \vdots \\ 0 \end{pmatrix} - \boldsymbol{V} \begin{pmatrix} Q_\alpha(\boldsymbol{y}_1) \\ \vdots \\ Q_\alpha(\boldsymbol{y}_m) \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

故有

$$\hat{\boldsymbol{D}}_\beta = \left(c_1 \boldsymbol{D}_\beta + c_2\right) + \left(c_1 - 1\right) \tilde{\boldsymbol{Q}},$$

其中 $\tilde{\boldsymbol{Q}}$ 是问题

$$\begin{aligned} &\min_{Q \in \mathcal{Q}} \ \left\| \nabla^2 Q \right\|_F^2 \\ &\text{s. t. } Q(\boldsymbol{y}) = Q_\alpha(\boldsymbol{y}), \ \forall\, \boldsymbol{y} \in \mathcal{X} \end{aligned}$$

的解. 进而

$$\begin{aligned} \hat{Q}_\beta &= Q_\alpha + \hat{D}_\beta \\ &= Q_\alpha + \left(c_1 \left(Q_\beta - Q_\alpha\right) + c_2\right) + \left(c_1 - 1\right) \tilde{Q} \\ &= c_1 Q_\beta + c_2 + \left(c_1 - 1\right) \left(\tilde{Q} - Q_\alpha\right). \end{aligned}$$

因此, (3-15) 成立, 定理得证. □

我们可以得到以下推论.

推论 3.14. 假设 $\mathcal{X} = \{\boldsymbol{y}_1, \cdots, \boldsymbol{y}_m\} \subset \mathfrak{R}^n$ 是一个适定集合, 其中 $n+1 \leq m < \frac{1}{2}(n+1)(n+2)$. 若 L_α 是一个线性函数, 则

$$\mathcal{M}^{\mathcal{X}}_{L_\alpha}(c_1 f + c_2) = c_1 \mathcal{M}^{\mathcal{X}}_{L_\alpha}(f) + c_2 \tag{3-16}$$

对于 $c_1, c_2 \in \mathfrak{R}$ 成立.

证明. 由于 $|\mathcal{X}| \geq n+1$ 且 L_α 是一个线性函数, 根据插值, $\mathcal{M}^{\mathcal{X}}_0(L_\alpha) = L_\alpha$. 根据 (3-15), (3-16) 成立, 推论得证. □

该推论对应的是构造最小 Frobenius 范数二次模型, 这是因为 $\nabla^2 L_\alpha$ 是零矩阵.

通常情况下 $\mathcal{M}^{\mathcal{X}}_0(Q_\alpha) \neq Q_\alpha$. 因此, 基于 Q_α 在 $\mathcal{X}$ 上函数 $c_1 f + c_2$ 的最小 Frobenius 范数更新二次模型可能无法通过相同的仿射变换获得, 除非 $c_1 = 1$.

上述分析还表明, 对于 $c_2 \in \mathfrak{R}$, 满足 $T \circ f = f + c_2$ 的平移变换是一个保模型最优性变换.

为了进一步分析仿射变换与模型函数之间的关系, 我们给出以下定理.

定理 3.15. 假设 Q_α 是一个二次函数, 并且 $\mathcal{X} = \{\boldsymbol{y}_1, \cdots, \boldsymbol{y}_m\} \subset \mathfrak{R}^n$ 是一个适定集合, 其中 $n+1 \leq m < \frac{1}{2}(n+1)(n+2)$. 给定常数 $\nu_1, \nu_2 \in \mathfrak{R}$, 我们有

$$\mathcal{M}^{\mathcal{X}}_{\nu_1 Q_\alpha + \nu_2}(f) = \nu_1 \mathcal{M}^{\mathcal{X}}_{Q_\alpha}(f) + (1-\nu_1)\mathcal{M}^{\mathcal{X}}_0(f). \tag{3-17}$$

证明. 设 $Q_\gamma := \mathcal{M}^{\mathcal{X}}_{\nu_1 Q_\alpha + \nu_2}(f)$, $Q_\beta := \mathcal{M}^{\mathcal{X}}_{Q_\alpha}(f)$, $Q_\phi := \mathcal{M}^{\mathcal{X}}_0(f)$. 记

$$Q_\gamma(\boldsymbol{x}) - \nu_1 Q_\alpha(\boldsymbol{x}) - \nu_2 = c_\gamma + (\boldsymbol{x} - \boldsymbol{x}_0)^\top \boldsymbol{g}_\gamma + \frac{1}{2}\sum_{j=1}^m (\boldsymbol{\lambda}_\gamma)_j \left((\boldsymbol{x} - \boldsymbol{x}_0)^\top (\boldsymbol{y}_j - \boldsymbol{x}_0)\right)^2,$$

$$Q_\beta(\boldsymbol{x}) - Q_\alpha(\boldsymbol{x}) = c_\beta + (\boldsymbol{x} - \boldsymbol{x}_0)^\top \boldsymbol{g}_\beta + \frac{1}{2}\sum_{j=1}^m (\boldsymbol{\lambda}_\beta)_j \left((\boldsymbol{x} - \boldsymbol{x}_0)^\top (\boldsymbol{y}_j - \boldsymbol{x}_0)\right)^2,$$

$$Q_\phi(\boldsymbol{x}) = c_\phi + (\boldsymbol{x} - \boldsymbol{x}_0)^\top \boldsymbol{g}_\phi + \frac{1}{2}\sum_{j=1}^m (\boldsymbol{\lambda}_\phi)_j \left((\boldsymbol{x} - \boldsymbol{x}_0)^\top (\boldsymbol{y}_j - \boldsymbol{x}_0)\right)^2.$$

我们定义 $\boldsymbol{q}_1 \in \mathfrak{R}^{m+n+1}$ 和 $\boldsymbol{q}_2 \in \mathfrak{R}^{m+n+1}$ 为

$$\boldsymbol{q}_1 = \begin{pmatrix} f(\boldsymbol{y}_1) \\ \vdots \\ f(\boldsymbol{y}_m) \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad \boldsymbol{q}_2 = \begin{pmatrix} Q_\alpha(\boldsymbol{y}_1) \\ \vdots \\ Q_\alpha(\boldsymbol{y}_m) \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

我们有

$$\begin{pmatrix} \boldsymbol{\lambda}_\gamma \\ c_\gamma + \nu_2 \\ \boldsymbol{g}_\gamma \end{pmatrix} = \boldsymbol{V}\left(\boldsymbol{q}_1 - \nu_1 \boldsymbol{q}_2\right), \begin{pmatrix} \boldsymbol{\lambda}_\beta \\ c_\beta \\ \boldsymbol{g}_\beta \end{pmatrix} = \boldsymbol{V}\left(\boldsymbol{q}_1 - \boldsymbol{q}_2\right), \begin{pmatrix} \boldsymbol{\lambda}_\phi \\ c_\phi \\ \boldsymbol{g}_\phi \end{pmatrix} = \boldsymbol{V}\boldsymbol{q}_1.$$

因此我们有

$$\begin{pmatrix} \boldsymbol{\lambda}_\gamma \\ c_\gamma + \nu_2 \\ \boldsymbol{g}_\gamma \end{pmatrix} = \nu_1 \begin{pmatrix} \boldsymbol{\lambda}_\beta \\ c_\beta \\ \boldsymbol{g}_\beta \end{pmatrix} + \left(1 - \nu_1\right) \begin{pmatrix} \boldsymbol{\lambda}_\phi \\ c_\phi \\ \boldsymbol{g}_\phi \end{pmatrix},$$

故

$$Q_\gamma - \nu_1 Q_\alpha = \nu_1 \left(Q_\beta - Q_\alpha\right) + \left(1 - \nu_1\right) Q_\phi,$$

进而 (3-17) 成立. 结论得证. □

为了分析对应于仿射变换后的目标函数的模型函数, 我们基于定理 3.15 推导出以下推论.

推论 3.16. 假设 $\mathcal{X} = \{\boldsymbol{y}_1, \cdots, \boldsymbol{y}_m\} \subset \mathfrak{R}^n$ 是一个适定集合, $\hat{Q}_\alpha$ 是函数 f 在 $\mathcal{X}\backslash\{\boldsymbol{y}_t\}$ 上的二次插值模型, 其中 $n+1 \leq m < \frac{1}{2}(n+1)(n+2)$. 则对于任意 $c_1, c_2 \in \mathfrak{R}$, $c_1\hat{Q}_\alpha + c_2$ 是 $c_1 f + c_2$ 在 $\mathcal{X}\backslash\{\boldsymbol{y}_t\}$ 上的二次插值模型. 更进一步地, 我们对于任意 $c_1, c_2 \in \mathfrak{R}$ 有

$$\mathcal{M}^{\mathcal{X}}_{c_1\hat{Q}_\alpha + c_2}(c_1 f + c_2) - \mathcal{M}^{\mathcal{X}}_0(c_1 f + c_2) = c_1\Big(\mathcal{M}^{\mathcal{X}}_{\hat{Q}_\alpha}(c_1 f + c_2) - \mathcal{M}^{\mathcal{X}}_0(c_1 f + c_2)\Big),$$

其中 $\mathcal{M}^{\mathcal{X}}_0(c_1 f + c_2)$ 正是 $c_1 f + c_2$ 的最小 *Frobenius* 范数二次模型.

证明. 这是定理 3.15 在取 $\nu_1 = c_1, \nu_2 = c_2$ 后的直接结果. □

注 3.6. 推论 3.16 讨论了基于原始目标函数 f 的模型获取最小 Frobenius 范数更新二次模型与基于变换后的目标函数 $c_1 f + c_2$ 的模型获取该更新模型之间的关系.

3.4 完全线性模型与收敛性分析

本节中的收敛性分析针对的是标准可证明的算法框架, 即 Conn、Scheinberg 和 Vicente 著作 [20] 中的算法 10.1, 区别是使用我们的最小 Frobenius 范数更新二次模型来极小化变换后的目标函数. 可证明算法框架的唯一改变之处在于变换后的输出函数值和对我们模型的使用. 换言之, 算法使用的函数值是新增点所在迭代的变换值. 考虑到我们的模型可以提供完全线性模型, 我们详细研究了一阶临界点的全局收敛性. 为了探究存在给定仿射变换时对应的插值模型的表现, 我们首先给出目标函数被仿射变换时对应的最小 Frobenius 范数更新二次模型的完全线性误差常数.

3.4.1　完全线性误差常数

我们给出以下关于带有仿射变换的目标函数与欠定二次插值模型之间的插值误差的假设和定理.

假设 3.17. 假设 $\mathcal{X} = \{\boldsymbol{y}_1, \cdots, \boldsymbol{y}_m\} \subset \mathfrak{R}^n$ 是一个位于 $\mathcal{B}_\Delta(\boldsymbol{y}_c)$ 内的样本/插值点集, 并且其在线性插值或回归意义上适定性良好, 其中 $\boldsymbol{y}_c \in \mathcal{X}$ 且 $n+1 \leq |\mathcal{X}| = m < \frac{1}{2}(n+1)(n+2)$.

此外, 我们定义 $\hat{\boldsymbol{L}} = \frac{1}{\Delta}\boldsymbol{L} = \frac{1}{\Delta}(\boldsymbol{y}_1 - \boldsymbol{y}_c, \cdots, \boldsymbol{y}_{c-1} - \boldsymbol{y}_c, \boldsymbol{y}_{c+1} - \boldsymbol{y}_c, \cdots, \boldsymbol{y}_m - \boldsymbol{y}_c)^\top \in \mathfrak{R}^{(m-1)\times n}$ 以及 $\hat{\boldsymbol{L}}^\dagger = (\hat{\boldsymbol{L}}^\top \hat{\boldsymbol{L}})^{-1}\hat{\boldsymbol{L}}^\top$. 我们给出下面的假设和定理.

假设 3.18. 假设 Q_α 是一个二次函数, 并且二次模型函数 $Q_\beta := \mathcal{M}^{\mathcal{X}}_{Q_\alpha}(f)$ 是函数 f 的一个具有误差常数 κ_g 和 κ_f 的完全线性模型 [20, 21], 即

$$\begin{aligned} \left\|\nabla Q_\beta(\boldsymbol{x}) - \nabla f(\boldsymbol{x})\right\|_2 &\leq \kappa_g \Delta, \ \forall\, \boldsymbol{x} \in \mathcal{B}_\Delta(\boldsymbol{y}_c), \\ \left|Q_\beta(\boldsymbol{x}) - f(\boldsymbol{x})\right| &\leq \kappa_f \Delta^2, \ \forall\, \boldsymbol{x} \in \mathcal{B}_\Delta(\boldsymbol{y}_c). \end{aligned}$$

定理 3.19. *假定假设 3.17 和 3.18 成立. 则二次模型函数 $\hat{Q}_\beta := \mathcal{M}^{\mathcal{X}}_{Q_\alpha}(c_1 f + c_2)$ 是 $c_1 f + c_2$ 的一个完全线性模型, 且其作为完全线性模型对于任意 $c_1, c_2 \in \mathfrak{R}$ 具有误差常数*

$$\begin{aligned} \hat{\kappa}_g &= |c_1|\kappa_g + |c_1 - 1|\left(\frac{5\sqrt{m-1}}{2}\left\|\hat{\boldsymbol{L}}^\dagger\right\|_2 \left(\mu_\alpha + \left\|\nabla^2 \tilde{Q}\right\|_2\right)\right), \\ \hat{\kappa}_f &= |c_1|\kappa_f + |c_1 - 1|\left(\frac{5\sqrt{m-1}}{2}\left\|\hat{\boldsymbol{L}}^\dagger\right\|_2 + \frac{1}{2}\right)\left(\mu_\alpha + \left\|\nabla^2 \tilde{Q}\right\|_2\right), \end{aligned}$$

其中 $\tilde{Q} := \mathcal{M}^{\mathcal{X}}_0(Q_\alpha)$, μ_α 是线性函数 ∇Q_α 的 Lipschitz 常数. 换言之, 我们有

$$\begin{aligned} \left\|\nabla \hat{Q}_\beta(\boldsymbol{x}) - \nabla(c_1 f(\boldsymbol{x}) + c_2)\right\|_2 &\leq \hat{\kappa}_g \Delta, \ \forall\, \boldsymbol{x} \in \mathcal{B}_\Delta(\boldsymbol{y}_c), \\ \left|\hat{Q}_\beta(\boldsymbol{x}) - (c_1 f(\boldsymbol{x}) + c_2)\right| &\leq \hat{\kappa}_f \Delta^2, \ \forall\, \boldsymbol{x} \in \mathcal{B}_\Delta(\boldsymbol{y}_c). \end{aligned}$$

证明. 根据 Conn、Scheinberg 和 Vicente 的著作 [20] 中的定理 5.4, 我们有

$$\begin{aligned} \left\|\nabla Q_\alpha(\boldsymbol{x}) - \nabla \tilde{Q}(\boldsymbol{x})\right\|_2 &\leq \frac{5\sqrt{m-1}}{2}\left\|\hat{\boldsymbol{L}}^\dagger\right\|_2 \left(\mu_\alpha + \left\|\nabla^2 \tilde{Q}\right\|_2\right)\Delta, \ \forall\, \boldsymbol{x} \in \mathcal{B}_\Delta(\boldsymbol{y}_c), \\ \left|Q_\alpha(\boldsymbol{x}) - \tilde{Q}(\boldsymbol{x})\right| &\leq \left(\frac{5\sqrt{m-1}}{2}\left\|\hat{\boldsymbol{L}}^\dagger\right\|_2 + \frac{1}{2}\right)\left(\mu_\alpha + \left\|\nabla^2 \tilde{Q}\right\|_2\right)\Delta^2, \ \forall\, \boldsymbol{x} \in \mathcal{B}_\Delta(\boldsymbol{y}_c). \end{aligned}$$

因此, 结合定理 3.13 和假设 3.18, 定理成立. □

3.4.2　一阶临界点的全局收敛性

现在转而讨论我们方法的收敛性. 我们假设相应模型的完全线性误差常数具有一致上界. 为了避免混淆, 这里需要指出的是我们的收敛性分析是针对一般的正单调变换的, 而不仅仅是第 3.4.1 节中讨论的仿射变换. 我们假设变换后的函数 f_k 及其梯度在相应的域内是 Lipschitz 连续的.

假设 3.20. 假设给定初始点 $\boldsymbol{x}_{\text{int}} \in \mathfrak{R}^n$ 和信赖域半径的上界, 即 $\Delta_{\max}$. 假设 f 以及所有 f_k 在包含集合 $\mathcal{L}_{\text{enl}}(\boldsymbol{x}_{\text{int}})$ 的区域内连续可微, 并且梯度是 Lipschitz 连续的, 其中

$$\mathcal{L}_{\text{enl}}(\boldsymbol{x}_0) = \bigcup_{\boldsymbol{x}\in\mathcal{L}(\boldsymbol{x}_{\text{int}})} \mathcal{B}_{\Delta_{\max}}(\boldsymbol{x}),$$

且$\mathcal{L}(\boldsymbol{x}_{\text{int}}) = \{\boldsymbol{x} \in \mathfrak{R}^n : f(\boldsymbol{x}) \le f(\boldsymbol{x}_{\text{int}})\}$.

我们假设每个变换后的函数 f_k 都有下界, 具体如下.

假设 3.21. 假设 f 和所有的 f_k 在 $\mathcal{L}(\boldsymbol{x}_{\text{int}})$ 上有下界, 即存在常数 κ_* 使得对于所有的 $\boldsymbol{x} \in \mathcal{L}(\boldsymbol{x}_{\text{int}})$, 有 $f(\boldsymbol{x}) \ge \kappa_*$ 和 $f_k(\boldsymbol{x}) \ge \kappa_*, \forall\, k \in \mathbb{N}^+$.

简明起见, 我们设模型函数的 Hessian 矩阵 (即 $\nabla^2 Q_k$) 是一致有界的, 细节如下.

假设 3.22. 存在常数 $\kappa_{\text{bhm}} > 0$ 使得对于算法生成的所有迭代, 我们有 $\|\nabla^2 Q_k\|_2 \le \kappa_{\text{bhm}}$.

参考 Conn、Scheinberg 和 Vicente 著作 [20] 中第 10 章算法 10.1 的收敛性分析的 (相同) 证明过程 (但是用变换后的函数 f_k 替代原始函数 f), 我们可以直接得到关于极小化变换后目标函数的算法的以下收敛性定理.

定理 3.23. *假定假设 3.20、假设 3.21 和假设 3.22 成立. 假设对于每个 $k \in \mathbb{N}^+$, 变换 T_k 是正单调变换, 并且算法给出的模型的完全线性误差常数和模型梯度的 Lipschitz 常数有一致的上界. 则*

$$\lim_{k\to\infty} \nabla f_k(\boldsymbol{x}_k) = \boldsymbol{0} \tag{3-18}$$

成立, 其中 $f_k(\boldsymbol{x}) = T_k(f(\boldsymbol{x}))$. 此外, 我们有

$$\lim_{k\to\infty} \nabla f(\boldsymbol{x}_k) = \boldsymbol{0}. \tag{3-19}$$

证明. (3-18) 的证明过程与 Conn、Scheinberg 和 Vicente 著作 [20] 中第 10.4 节关于基于模型的无导数信赖域方法的收敛性分析相同. 注意, 关于模型的完全线性误差常数和模型梯度的 Lipschitz 常数有一致上界的假设保证了与书中的引理 10.5 和引理 10.6 相对应的结果. 此外, 考虑到正单调变换, 我们知道存在 $\varepsilon > 0$, 使得

$$\liminf_{k\to\infty} \frac{df_k}{df} > \varepsilon,$$

进而

$$\nabla f_k(\boldsymbol{x}_k) = \frac{df_k}{df}\nabla f(\boldsymbol{x}_k).$$

因此, (3-19) 成立. 定理得证. □

定理 3.23 中的变换是正单调的, 它们包括具有正乘法系数的随机仿射变换, 如与 (3-20) 对应的随机仿射变换.

考虑到 Powell 的 NEWUOA 算法是使用最小 Frobenius 范数更新二次模型的经典高效的基于模型的算法, 第 3.5 节将展示我们基于 NEWUOA 所改进的方法的一个实现版本 NEWUOA-Trans 的结果. 需要特别说明的是, 上述的收敛性分析是基于可证明框架的, 而不是针对 NEWUOA 或 NEWUOA-Trans 的, 这是考虑到 NEWUOA 代码的复杂结构使其收敛性相当难以分析, 即便是极小化无变换的目标函数的情形也仍是一个公开难题.

NEWUOA-Trans 与 NEWUOA 共用相同的框架, 但其通过 (3-7) 更新相应的模型, 这可以被理解成是 Powell 最小 Frobenius 范数更新的直接扩展. 在 NEWUOA 和 NEWUOA-Trans 中, 模型改进步首先尝试替换掉距离当前 $\boldsymbol{x}_{\text{opt}}$ 和其他插值点太远的插值点 (例如, 替换超出以 $\boldsymbol{x}_{\text{opt}}$ 为中心、半径为 $2\Delta_k$ 的信赖域外的插值点). 当插值集中的所有点彼此足够接近时, NEWUOA-Trans 将检查插值集的适定性. 模型改进步将通过极大化相应的 Lagrange 多项式相关的绝对值或更新 KKT 矩阵逆矩阵公式的分母来找到新的插值点. 上述过程不会受到变换的影响. 如果插值集是适定的, 那么 NEWUOA 和 NEWUOA-Trans 就不需要再改进模型. 在插值集不适定的情况下, 每一步将替换集合中的一个点. 参考 Conn、Scheinberg 和 Vicente 著作 [20] 中的定理 6.3, 其在插值区域内将获得适定的插值集, 进而获得完全线性模型. 事实上, 模型改进步可以保证在有限次迭代内产生完全线性模型. 因此, NEWUOA 和 NEWUOA-Trans 的插值更新可以保证在有限且一致有界的步数内构造出完全线性模型.

3.5　数值结果

前文分析表明, 如果 $c_1 \neq 1$, 则满足 $T \circ f = c_1 f + c_2$ 且 $c_1 > 0$ 的仿射变换 T 在一般情况下并不是保模型最优性变换. 然而, 仿射变换极其基本和重要, 并且具有实际应用价值. 例如, 仿射变换对应在加密黑箱优化中具有不同隐私保护机制的加性和乘性噪声的添加机制. 实际上, 我们在前一节中已经理论分析了仿射变换目标函数的最小 Frobenius 范数更新二次模型的解析表达式和插值误差, 我们将在本节通过数值实验进一步观察我们方法的表现.

如前文所述, 为求解带有变换目标函数的无导数优化问题, 我们基于 Powell 的 NEWUOA 算法 [94] 实现了一种无导数算法, 并命名为 NEWUOA-Trans[4]. NEWUOA-Trans 中使用的欠定模型通过 (3-7) 更新. 本部分展示了使用 NEWUOA-Trans 求解某些带变换目标函数的无导数优化问题的数值结果. 数值结果刻画了 NEWUOA-Trans 的主要特性和优势. 总体上, NEWUOA-Trans 是一个稳健高效的算法, 它可以用于极小化带变换的目标函数. NEWUOA-Trans 的代码变动主要发生在更新模型的 Hessian 和梯度的部分 (处理这种复杂代码并非易事). 注意, NEWUOA-Trans 中的其他部分参考了 NEWUOA 中的对应部分.

[4]"-Trans" 表示它是为求解带有变换目标函数的问题而设计的.

3.5.1 算法对比和相关变换

在数值实验中, 我们用表 3-2 中的算法来求解带有变换目标函数的无导数优化问题. 所有问题的目标函数都经过了如 (3-20) 所示的变换. 此外, 我们还测试了 NEWUOA-N. 注意, NEWUOA-N 求解的问题没有噪声, 即目标函数没有变换, 这里的 "-N" 表示无噪声, NEWUOA-N 在某种意义上可以被视为基准. 另外, NEWUOA-Trans 和 NEWUOA-N 之间的对比可以说明 NEWUOA-Trans 是否可以减小或克服变换对目标函数的影响. 细节见表 3-2. 在 NEWUOA-trans、NEWUOA-N 和 NEWUOA 中, 我们设置 $\hat{\rho}_{\text{beg}} = 10^{-1}$, $\hat{\rho}_{\text{end}} = 10^{-8}$, $m = 2n + 1$ (Powell 的原文中使用的记号是 ρ_{beg} 和 ρ_{end}), NEWUOA 框架的更多细节见 Powell 文章 [94] 中的图 1.

表 3-2 比较的算法

Table 3-2 Compared algorithms

算法	模型	问题
NEWUOA-Trans	我们的模型	变换后的目标函数
NEWUOA	Powell 的模型 [94]	变换后的目标函数
NEWUOA-N	Powell 的模型	原始目标函数 (无变换)

在第 3.5.2 节和第 3.5.3 节的数值实验中, 第 k 步时, 第 k 批探测点中的任意 $\boldsymbol{x}$ 上的目标函数值 $f(\boldsymbol{x})$ 将被变换为

$$f_k(\boldsymbol{x}) = (\gamma_k + 1) f(\boldsymbol{x}) + C\eta_k, \tag{3-20}$$

其中 $\eta_k \sim \text{Lap}(b_k)$, $b_k > 0$, 且 $\gamma_k \sim \text{U}(-u_k, u_k)$, $0 < u_k < 1$. $\text{Lap}(b_k)$ 的概率密度函数是 $p(x) = \frac{1}{2b_k} e^{-\frac{|x|}{b_k}}$. 此外, U 表示均匀分布, 相应的概率密度函数是

$$p(x) = \begin{cases} \dfrac{1}{2u_k}, & \text{如果 } x \in [-u_k, u_k], \\ 0, & \text{否则.} \end{cases}$$

3.5.2 变换对 NEWUOA 算法的攻击: 一个简单的例子

下面的简单例子展示了目标函数中的变换 (甚至是仿射变换) 将导致未修改的 NEWUOA 在求解过程中失败. 即变换对 NEWUOA 算法形成了攻击或干扰.

例 3.3. 在与表 3-3 对应的数值实验中, 目标函数是

$$f(\boldsymbol{y}) = \sum_{i=1}^{10} y_i^4 + \sum_{i=1}^{10} y_i^2,$$

这里 $\boldsymbol{y} = (y_1, \cdots, y_n)^\top$. 在此例子中, 问题维数 n 是 10. 此外, 初始点是 $(10, \cdots, 10)^\top$, 在 (3-20) 中常数 $C = 1$. 数值实验的解析解是 $(0, \cdots, 0)^\top$, 对应的最小函数值是 0.

表 3-3 中的符号 ✓ 和 × 表示算法是否成功求解问题. 符号 ✓ 表示由算法获得的数值最优函数值 f_{opt} 小于 10^{-3}, 符号 × 则表示未达到该精度. 符号 NF 表示迭代终止时被获取函数值的点的个数. 此外, NEWUOA-N 使用 990 次函数值探测即可获得函数值小于 10^{-16} 的点.

表 3-3 例 3.3 的数值结果

Table 3-3 Numerical results for Example 3.3

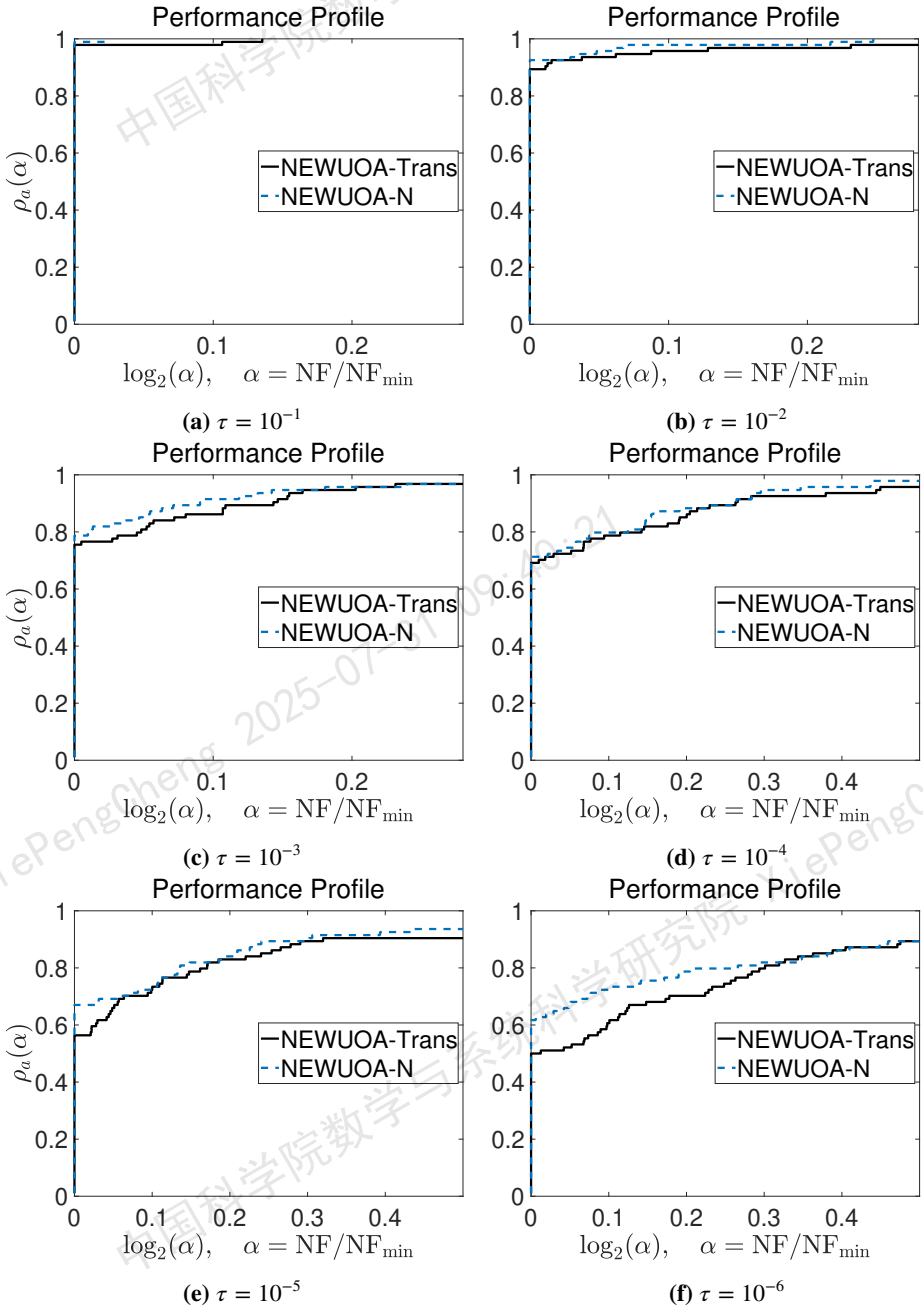
变换参数	$\eta_k \sim \text{Lap}(\frac{1}{k}), \gamma_k = 0$			$\eta_k \sim \text{Lap}(\frac{100}{k}), \gamma_k = 0$		
算法	NF	f_{opt}		NF	f_{opt}	
NEWUOA-Trans	1033	1.5626×10^{-13}	✓	1046	7.7485×10^{-13}	✓
NEWUOA	613	0.1375	×	348	7.2318	×
变换参数	$\eta_k \sim \text{Lap}(\frac{10}{k}), \gamma_k = 0$			$\eta_k = 0, \gamma_k \sim \text{U}(-\frac{1}{k}, \frac{1}{k})$		
算法	NF	f_{opt}		NF	f_{opt}	
NEWUOA-Trans	847	2.6014×10^{-13}	✓	1055	3.1489×10^{-13}	✓
NEWUOA	542	1.5818	×	408	0.7345	×
变换参数	$\eta_k \sim \text{Lap}(\frac{100}{k}), \gamma_k \sim \text{U}(-\frac{1}{k}, \frac{1}{k})$			$\eta_k \sim \text{Lap}(\frac{100}{k}), \gamma_k \sim \text{U}(-\frac{k}{10^4}, \frac{k}{10^4})$		
算法	NF	f_{opt}		NF	f_{opt}	
NEWUOA-Trans	1056	4.1928×10^{-13}	✓	948	1.1924×10^{-13}	✓
NEWUOA	432	6.5330	×	409	4.0762	×

从表 3-3 中可以看到 NEWUOA 在求解带有简单变换的目标函数的问题时几乎无法成功. 换言之, 它在求解带有变换目标函数的无导数优化问题时表现不佳, 这正是由变换/噪声的影响所导致的. 此外, NEWUOA-N 和 NEWUOA-Trans 的结果接近, 考虑到 NEWUOA-N 是一个基准, 这说明 NEWUOA-Trans 在求解带有变换目标函数的优化问题时的表现令人满意.

3.5.3　算法表现

我们使用 Performance Profile 来比较不同算法. 图 3-3 和图 3-4 中展示的测试问题和数值结果列在表 4-4 中. 它们的维数范围是 2 到 100, 来自经典常见的无约束优化测试函数集 [92, 177, 178, 180, 181, 183–186]. 对于每个算法, 函数值探测次数的上限被设置为 10000.

如例 3.3 所示, 未经过修改的 NEWUOA 不适合求解 DFOTO 问题. 我们在这里比较 NEWUOA-Trans 和 NEWUOA-N. 这里变换中的参数为 $C = 100$, $\eta_k \sim \text{Lap}(\frac{100}{k})$, $\gamma_k \sim \text{U}(-\frac{1}{k}, \frac{1}{k})$, 两个算法共用相同的初始点. 在图 3-3a 到图 3-3f 中, 可以



(a) $\tau = 10^{-1}$ **(b)** $\tau = 10^{-2}$

(c) $\tau = 10^{-3}$ **(d)** $\tau = 10^{-4}$

(e) $\tau = 10^{-5}$ **(f)** $\tau = 10^{-6}$

图 3-3 使用不同算法求解测试问题的 Performance Profile

Figure 3-3 The comparison of algorithms solving the test problems: Performance Profile

表 3-4 图 3-3 对应的测试问题

Table 3-4 Test problems for Figure 3-3

ARGLINA	ARGLINA4	ARGLINB	ARGLINC	ARGTRIG
ARWHEAD	BDQRTIC	BDQRTICP	BDALUE	BROWNAL
BROYDN3D	BROYDN7D	BRYBND	CHAINWOO	CHEBQUAD
CHNROSNBZ	CHPOWELLB	CHPOWELLS	CHROSEN	COSINECUBE
CURLY10	CURLY20	CURLY30	DIXMAANE	DIXMAANF
DIXMAANG	DIXMAANH	DIXMAANI	DIXMAANJ	DIXMAANK
DIXMAANL	DIXMAANM	DIXMAANN	DIXMAANO	DIXMAANP
DQRTIC	EDENSCH	ENGVAL1	ERRINROS	EXPSUM
EXTROSNB	EXTTET	FIROSE	FLETCBV2	FLETCBV3
FLETCHCR	FMINSRF2	FREUROTH	GENBROWN	GENHUMPS
GENROSE	INDEF	INTEGREQ	LIARWHD	LILIFUN3
LILIFUN4	MOREBV	MOREBVL	NCB20	NCB20B
NONCVXU2	NONCVXUN	NONDIA	NONDQUAR	PENALTY1
PENALTY2	PENALTY3	PENALTY3P	POWELLSG	POWER
ROSENBROCK	SBRYBND	SBRYBNDL	SCHMVETT	SCOSINE
SCOSINEL	SEROSE	SINQUAD	SPARSINE	SPARSQUR
SPHRPTS	SPMSRTLS	SROSENBR	STMOD	TOINTGSS
TOINTTRIG	TQUARTIC	TRIGSABS	TRIGSSQS	TRIROSE1
TRIROSE2	VARDIM	WOODS	-	-

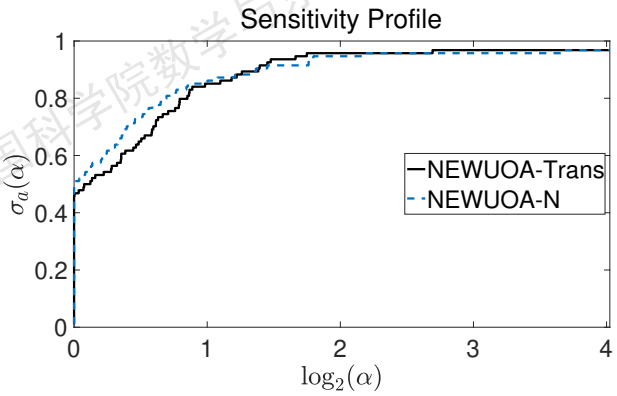


图 3-4 使用不同算法求解测试问题的 Sensitivity Profile

Figure 3-4 The comparison of algorithms solving the test problems: Sensitivity Profile

观察到, 当精度 $\tau = 10^{-1}, \cdots, 10^{-6}$ 时, NEWUOA-Trans 和 NEWUOA-N 表现接近. NEWUOA-Trans 在带有变换目标函数的问题上表现很好. 图 3-3a 到图 3-3f 中的比较显示 NEWUOA-Trans 可以成功求解大多数带有变换目标函数的无导数黑箱优化问题. NEWUOA-Trans 与 NEWUOA-N (求解无噪声问题的基准) 之间的轻微差异来自于随机噪声对模型的影响.

在图 3-4 报告的数值结果中, 目标函数选自表 4-4 中的测试问题. 此外, $C = 100$ 且 $\tau = 10^{-4}$. $\sigma_a(\alpha)$ 的值更高意味着在 Sensitivity Profile [46] 中算法的稳定性更强. 图 3-4 显示, NEWUOA-Trans 的表现与 NEWUOA-N 接近, 这意味着 NEWUOA-Trans 的舍入误差与我们比较的基准 NEWUOA-N 接近. 实际上, Sensitivity Profile 是另一个重要的标准, 我们用其评估算法的稳定性. 我们用 $\boldsymbol{P}_i \in \mathfrak{R}^{n\times n}$, $i = 1, 2, \cdots, M$, 表示随机排列矩阵. 在实验中, $M = 100$, 这里要用到的随机排列矩阵的一个例子是 $\boldsymbol{P}_1 = \left(\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_4, \boldsymbol{e}_3, \boldsymbol{e}_8, \boldsymbol{e}_5, \boldsymbol{e}_9, \boldsymbol{e}_{10}, \boldsymbol{e}_6, \boldsymbol{e}_7\right)^{\mathsf{T}}$. 此外, 我们定义 $\mathrm{NF} = \left(\mathrm{NF}_1, \cdots, \mathrm{NF}_M\right)$, 其中 NF_i 表示求解相应问题 $\min\limits_{\boldsymbol{x}\in\mathfrak{R}^n} f(\boldsymbol{P}_i\boldsymbol{x})$. 时的函数值探测次数. 我们定义 mean(NF) 为 $\frac{1}{M}\sum_i^M \mathrm{NF}_i$, 标准差 std(NF) 为 $\sqrt{\frac{1}{M}\sum_i^M (\mathrm{NF}_i - \mathrm{mean(NF)})^2}$. 在 Performance Profile 中, 我们使用 std(NF) 来替代 $N_{a,p}$, 对应于用算法 a 求解问题 p, 以最终获得 $\sigma_a(\alpha)$, 我们进而获得 Sensitivity Profile.

我们可以看到, NEWUOA-Trans 和 NEWUOA-N 的曲线彼此接近, 这说明变换的干扰几乎不影响 NEWUOA-Trans.

我们的数值实验表明, NEWUOA-Trans 能够高效稳定地求解大部分具有这类变换后目标函数的无导数优化问题. NEWUOA-Trans 和 NEWUOA-N 的表现接近, 这表明 NEWUOA-Trans 很好地求解了带有变换目标函数的优化问题, 克服了来自变换的干扰.

3.5.4 实际问题实验

我们将我们的方法用于针对带有隐私保护的空间行波管的工程最优设计 [205]. 行波管是一种关键的真空电子设备 [206], 影响信号质量和强度, 主要用于通信、交通、导航、气象测量、预测等领域.

考虑到空间行波管的特殊工作环境, 空间行波管的效率非常关键. 在空间行波管的设计中, 设计参数对效率非常重要, 这也正是我们想要极大化的, 而其表达式是未知的, 参数与目标函数之间的关系难以分析. 此外, 空间行波管的数据难以获得. 关于其效率性能的大部分数据需要实验或仿真, 这涉及到非常高的费用或时间成本, 除此之外, 实际中一些数据还是加密的. 因此, 求解这样一个昂贵优化问题正是典型的无导数优化问题. 作为与安全、版权、利益等相关的重要工业产品, 在优化过程中 (特别是对公众和优化算法的设计方, 即第三方), 特殊种类的空间行波管的效率真值会被加密. 因此, 优化特殊种类的行波管的效率是一个带有隐私保护的黑箱优化问题, 属于带变换的无导数优化.

我们进行了以下数值实验, 空间行波管的测试数据由北京真空电子技术研究

所提供. 该数值实验的目的是找到带有隐私保护的空间行波管的最佳设计参数, 使其具有最高效率, 这被形式化为带有隐私保护的无约束无导数优化问题[5]

$$\max_{\mathcal{P}_{\text{input}}} \ \text{效率}\left(\mathcal{P}_{\text{input}}\right),$$

其中 $\mathcal{P}_{\text{input}}$ 是一个 10 维向量, 表示设计空间行波管的参数. 为了保护效率 (目标函数和效率直接相关) 的真实值, 空间行波管的设计者在每个探测步应用随机仿射变换来加密探测过程中的真实函数值. 基本探测过程遵循假设 3.1 和表 3-1. 我们应用 NEWUOA-Trans 求解这个问题, 我们选择初始输入 $10^{-1} \times (2, \cdots, 2)^{\mathrm{T}}$ 作为初始点, $\hat{\rho}_{\text{beg}} = 10^{-1}$, $\hat{\rho}_{\text{end}} = 10^{-4}$. NEWUOA-Trans 在 226 次迭代后终止. 迭代过程可以在表 3-5 中看到, 该表展示了第 k 次迭代时最佳迭代点 $\boldsymbol{x}_k$ 与最终解 $\boldsymbol{x}^*$ 之间的欧氏距离.

表 3-5 第 k 步最佳迭代点与最终解之间的距离: $\|\boldsymbol{x}_k - \boldsymbol{x}^*\|_2$

Table 3-5 The distance between the best iteration point at the k-th step and the final solution: $\|\boldsymbol{x}_k - \boldsymbol{x}^*\|_2$

迭代	10	20	30	40	50	60	70	80
距离	84.4577	42.6845	19.0530	13.7870	7.7990	4.7825	0.9851	0.8116
迭代	90	100	110	120	130	140	150	160
距离	0.7110	0.5525	0.5106	0.4705	0.4034	0.3035	0.1318	0.1102
迭代	170	180	190	200	210	220		
距离	0.0800	0.0560	0.0370	0.0102	0.0025	0		

为了验证我们的结果, 我们可以使用大型仿真计算软件 CST 来进行仿真设计, 我们发现, 在工作频率带上, 我们所最终得到的参数对应的效率与基于专家知识和经验的最佳设置相比有明显提高, 如表 3-6 所示.

表 3-6 效率增量

Table 3-6 Efficiency increment

频率点 (GHz)	94	97	100
效率增量 (‰)	53	62	66

根据相应的业界评估机制, 使用 NEWUOA-Trans 求解相应带变换无导数优化问题获得的参数实现了这种特殊空间行波管设计的最大效率, 且所得最大效率在行业内是令人满意的. 这表明我们的方法具有强大的实用性. 业界也潜在地偏好我们方法的隐私保护功能, 这主要是因为在一些情况下数据提供者只能 (且只需要) 输出变换后的函数值. 上述的初步应用也启发我们在更广泛的领域应用我们的方法.

[5]为了简化, 一些约束事先已被调整删除.

3.6 小结

在本章结束之前, 我们提出了一个延伸的带变换优化问题, 这是一个新的、有挑战性的数学规划问题.

公开问题 3.24 (无导数方法极小化 “移动靶” 类型目标函数). 尝试为无约束问题 (3-1) 设计实用的数值优化算法, 其中 $f(\boldsymbol{x}, t)$ 是黑箱函数 f 在 $\boldsymbol{x} \in \mathfrak{R}^n$ 和给定的 $t \in \mathfrak{R}$ 处的实际输出值, 其中 t 严格依赖于当前点 $\boldsymbol{x}$ 的探测顺序, 或者说 t 可以被视为时间. 换言之, 探测的函数值集合将呈现 $\{f(\boldsymbol{x}_1, t_1), f(\boldsymbol{x}_2, t_2), \cdots, f(\boldsymbol{x}_k, t_k), \cdots\}$ 的形式, 其中 t_k 可以对应离散探测时间.

本章讨论了带有变换目标函数的无导数优化. 我们提出了一种相应的探测方案. 对于信赖域中具有唯一极小点的严格凸模型, 我们证明了: 除了平移变换之外, 存在其他保模型最优性变换. 本章提出了关于保模型最优性变换的变换函数值的充分必要条件. 我们获得了仿射变换目标函数的相应二次模型, 并证明了一些正单调变换 (即使是具有正乘法系数的仿射变换) 不是保模型最优性变换. 我们还给出了给定仿射变换目标函数的相应模型函数的插值误差分析. 给出了一阶临界点的收敛性分析. 测试问题和实际应用问题的结果也在数值上显示了我们方法的优势.

本章是对这个方向的初步尝试, 关于带变换的无导数优化仍有很多内容值得研究. 将来我们将调查和研究更多的应用, 包括在更多工程应用中构造带有变换目标函数的无导数优化的最小 Frobenius 范数更新二次模型的细节 (例如, 带有噪声添加机制的加密黑箱优化). 正如第 3.4 节所讨论的, 完全线性误差常数在每次迭代中会有变化, 而没有一致的界, 在这种情况下如果乘法系数 c_1 无界, 它们可能会在迭代中无限增长. 因此, 在比第 3.4 节中使用的假设更弱的情况下分析求解带有变换的问题的收敛性仍然是一个开放且具有挑战性的问题. 另外, 关于极小化 “移动目标” 类型目标函数的公开问题 3.24 也是有趣且有价值的.

第 4 章　子空间方法和并行方法

本章围绕无约束无导数优化, 提出了一个新的可以用于求解大规模问题的子空间无导数优化算法. 此外, 我们还将在本章提出一个新的将信赖域方法和线搜索方法相结合的并行方法.

4.1　无导数子空间信赖域方法 2D-MoSub

为了求解大规模无约束无导数优化问题, 本节将介绍一种新的无导数优化方法, 该方法利用子空间技术和低维二次插值模型有效地、迭代地搜索最优解.

张在坤 [46] 介绍了子空间技巧在无导数优化中的应用, 提出了一类无导数子空间算法的框架. 本部分将介绍我们新提出的一个具体的无导数优化方法 (2D-MoSub) 的框架和计算细节以及与子空间相关的坐标变换等, 还将讨论插值集的适定性和插值质量, 分析 2D-MoSub 的一些性质, 包括具有投影性质的近似误差和收敛性, 并将展示求解大规模问题的数值结果.

4.1.1　2D-MoSub 算法

在本节中, 我们将介绍我们所提出的 2D-MoSub 算法. 其框架如算法 6 所示.

算法 6 2D-MoSub 算法

输入: $\boldsymbol{x}_{\text{int}} \in \mathfrak{R}^n, \Delta_1, \Delta_{\text{low}}, \gamma_1, \gamma_2, \eta, \eta_0, \boldsymbol{d}^{(1)} \in \mathfrak{R}^n$, 令 $k = 1$.

步 0. (初始化)

获取 $\boldsymbol{y}_a, \boldsymbol{y}_b, \boldsymbol{y}_c$ 和 $\boldsymbol{d}_1^{(1)}$. 在 1 维空间 $\boldsymbol{x}_1 + \text{span}\{\boldsymbol{d}_1^{(1)}\}$ 上构造初始 1 维二次模型 Q_1^{sub}.

步 1. (构造插值集)

获取满足 $\langle \boldsymbol{d}_1^{(k)}, \boldsymbol{d}_2^{(k)} \rangle = 0$ 的单位方向 $\boldsymbol{d}_2^{(k)} \in \mathfrak{R}^n$. 获取 $\boldsymbol{y}_1^{(k)}, \boldsymbol{y}_2^{(k)}, \boldsymbol{y}_3^{(k)}$.

步 2. (构造二次插值模型)

在 2 维空间 $\boldsymbol{x}_k + \text{span}\{\boldsymbol{d}_1^{(k)}, \boldsymbol{d}_2^{(k)}\}$ 上构造 2 维二次模型 Q_k.

步 3. (信赖域试探步)

求解 Q_k 的信赖域子问题并选择性地求解修正后的模型 Q_k^{mod} 的信赖域子问题, 然后得到 $\boldsymbol{x}_k^+$. 计算

$$\rho_k = \frac{f(\boldsymbol{x}_k^+) - f(\boldsymbol{x}_k)}{Q_k(\boldsymbol{x}_k^+) - Q_k(\boldsymbol{x}_k)}. \tag{4-1}$$

更新并获取 $\boldsymbol{x}_{k+1}$ 和 $\boldsymbol{d}_1^{(k+1)}$. 转到**步 4**.

步 4. (更新)

若 $\Delta_k < \Delta_{\text{low}}$, 则终止. 否则, 更新 Δ_{k+1}, 令

$$\boldsymbol{d}_1^{(k+1)} = \frac{\boldsymbol{x}_{k+1} - \boldsymbol{x}_k}{\|\boldsymbol{x}_{k+1} - \boldsymbol{x}_k\|_2},$$

并在 1 维空间 $\boldsymbol{x}_{k+1}+\mathrm{span}\{\boldsymbol{d}_1^{(k+1)}\}$ 上构造 Q_{k+1}^{sub} 作为函数 Q_k. 令 $k=k+1$ 并转到**步 1**.

我们将在下面分别介绍算法 6 的每个部分的细节. 在进一步讨论之前, 我们给出以下注解和定义.

注 4.1. 在我们的方法中, 当我们讨论相应的 1 维插值模型和 2 维插值模型时, $\mathfrak{R}^n$ 中的点将被对应地视为 1 维子空间中的点或 2 维子空间中的点.

为满足计算需要, 给定 $\boldsymbol{a}\in\mathfrak{R}^n$ 和 $\boldsymbol{b}\in\mathfrak{R}^n$, 我们定义以下变换来分别表示从 2 维子空间 $S_{\boldsymbol{a},\boldsymbol{b}}^{(k)}=\boldsymbol{x}_k+\mathrm{span}\{\boldsymbol{a},\boldsymbol{b}\}$ 和 1 维子空间 $\hat{S}_{\boldsymbol{a}}^{(k)}=\boldsymbol{x}_k+\mathrm{span}\{\boldsymbol{a}\}$ 到 $\mathfrak{R}^2$ 或 $\mathfrak{R}$ 的坐标变换.

定义 4.1. 设 $\mathcal{T}_{\boldsymbol{a},\boldsymbol{b}}^{(k)}$ 为从 $S_{\boldsymbol{a},\boldsymbol{b}}^{(k)}$ 到 $\mathfrak{R}^2$ 的变换, 其定义为

$$\mathcal{T}_{\boldsymbol{a},\boldsymbol{b}}^{(k)}:\ \boldsymbol{y}\mapsto\left(\langle\boldsymbol{y}-\boldsymbol{x}_k,\boldsymbol{a}\rangle,\langle\boldsymbol{y}-\boldsymbol{x}_k,\boldsymbol{b}\rangle\right)^{\top}.$$

设 $\hat{\mathcal{T}}_{\boldsymbol{a}}^{(k)}$ 为从 $\hat{S}_{\boldsymbol{a}}^{(k)}$ 到 $\mathfrak{R}$ 的变换, 其定义为

$$\hat{\mathcal{T}}_{\boldsymbol{a}}^{(k)}:\ \boldsymbol{y}\mapsto\langle\boldsymbol{y}-\boldsymbol{x}_k,\boldsymbol{a}\rangle.$$

算法 2D-MoSub 首先通过初始化输入参数和向量开始. 它首先构造初始的 1 维二次插值模型 Q_1^{sub}. 为了启动算法, 我们首先使用初始点 $\boldsymbol{x}_0$ 进行初始化, 并设置各种参数: 信赖域参数 Δ_1 和 Δ_{low}, γ_1 和 γ_2, 以及成功步阈值 η 和 η_0. $\boldsymbol{d}^{(1)}$ 可以是 $\mathfrak{R}^n$ 中的任意方向. 例如, 我们可以让 $\boldsymbol{d}^{(1)}=(1,0,\cdots,0)^{\top}$. 根据大量的数值实验结果, 我们认为 $\boldsymbol{d}^{(1)}$ 的选择对算法的整体性能没有本质的影响.

算法 7 步 0. 初始化

1: **输入:** 获取初始点 $\boldsymbol{x}_{\mathrm{int}}$, 信赖域参数 Δ_1, Δ_{low}, γ_1, γ_2, 以及 η 和 η_0. 选择一个 $\boldsymbol{d}^{(1)}\in\mathfrak{R}^n$.

2: 获得三个点: $\boldsymbol{y}_a=\boldsymbol{x}_{\mathrm{int}}$, $\boldsymbol{y}_b=\boldsymbol{x}_{\mathrm{int}}+\Delta_1\boldsymbol{d}^{(1)}$, 以及基于 $f(\boldsymbol{y}_a)$ 和 $f(\boldsymbol{y}_b)$ 的相对值的 $\boldsymbol{y}_c$, 即

$$\boldsymbol{y}_c=\begin{cases}\boldsymbol{y}_a+2\Delta_1\boldsymbol{d}^{(1)}, & \text{如果 } f(\boldsymbol{y}_b)\le f(\boldsymbol{y}_a),\\ \boldsymbol{y}_a-\Delta_1\boldsymbol{d}^{(1)}, & \text{否则.}\end{cases}\tag{4-2}$$

3: 令 $\boldsymbol{x}_1$ 为 $\boldsymbol{y}_a$, $\boldsymbol{y}_b$ 和 $\boldsymbol{y}_c$ 中的函数值最小者, 即令

$$\boldsymbol{x}_1=\arg\min_{\boldsymbol{y}\in\{\boldsymbol{y}_a,\boldsymbol{y}_b,\boldsymbol{y}_c\}}f(\boldsymbol{y}).$$

4: 令 $\boldsymbol{y}_{\mathrm{max},1}^{(1)}$ 为 $\boldsymbol{y}_a$, $\boldsymbol{y}_b$ 和 $\boldsymbol{y}_c$ 中的函数值最大者, 即令

$$\boldsymbol{y}_{\mathrm{max},1}^{(1)}=\arg\max_{\boldsymbol{y}\in\{\boldsymbol{y}_a,\boldsymbol{y}_b,\boldsymbol{y}_c\}}f(\boldsymbol{y}).$$

5: 令 $\boldsymbol{d}_1^{(1)}$ 为从 $\boldsymbol{y}_{\max,1}^{(1)}$ 到 $\boldsymbol{x}_1$ 的归一化向量, 即令

$$\boldsymbol{d}_1^{(1)} = \frac{\boldsymbol{x}_1 - \boldsymbol{y}_{\max,1}^{(1)}}{\left\| \boldsymbol{x}_1 - \boldsymbol{y}_{\max,1}^{(1)} \right\|_2}.$$

6: 在 1 维空间 $\boldsymbol{x}_1 + \mathrm{span}\{\boldsymbol{d}_1^{(1)}\}$ 上构造初始 1 维二次插值模型 Q_1^{sub}, 即令

$$Q_1^{\mathrm{sub}}(\alpha) = f(\boldsymbol{x}_1) + a^{(1)}\alpha + b^{(1)}\alpha^2, \tag{4-3}$$

其中 $a^{(1)}, b^{(1)} \in \mathfrak{R}$ 由插值条件

$$Q_1^{\mathrm{sub}}(\hat{\mathcal{T}}_{\boldsymbol{d}_1^{(1)}}^{(1)}(\boldsymbol{y})) = f(\boldsymbol{y}),\ \forall\, \boldsymbol{y} \in \left\{\boldsymbol{y}_a, \boldsymbol{y}_b, \boldsymbol{y}_c\right\} \tag{4-4}$$

确定.

一旦上述参数的初始化完成, 2D-MoSub 将获得三个点: $\boldsymbol{y}_a = \boldsymbol{x}_{\mathrm{int}}$, $\boldsymbol{y}_b = \boldsymbol{x}_{\mathrm{int}} + \Delta_1 \boldsymbol{d}^{(1)}$, 以及根据 (4-2) 基于 $f(\boldsymbol{y}_a)$ 和 $f(\boldsymbol{y}_b)$ 的相对大小确定的 $\boldsymbol{y}_c$. 有了上述点后, 2D-MoSub 将 $\boldsymbol{x}_1$ 确定为 $\boldsymbol{y}_a$、$\boldsymbol{y}_b$ 和 $\boldsymbol{y}_c$ 中的函数值最小者. 同时, 2D-MoSub 令 $\boldsymbol{y}_{\max,1}^{(1)}$ 为 f 在同一组点 $\boldsymbol{y}_a$、$\boldsymbol{y}_b$ 和 $\boldsymbol{y}_c$ 中的函数值最大者. 注意, 如果它们具有相同的函数值, 我们将使用一定方式确保 $\boldsymbol{x}_1 \neq \boldsymbol{y}_{\max,1}^{(1)}$, 这里不再赘述, 我们假设本节中出现的选点操作都可以正常进行, 这不影响算法的整体思路和效果. 利用 $\boldsymbol{x}_1$ 和 $\boldsymbol{y}_{\max,1}^{(1)}$, 我们令 $\boldsymbol{d}_1^{(1)}$ 为从 $\boldsymbol{x}_1$ 指向 $\boldsymbol{y}_{\max,1}^{(1)}$ 的归一化向量. 最后, 2D-MoSub 根据插值条件 (4-4) 在 1 维空间 $\boldsymbol{x}_1 + \mathrm{span}\{\boldsymbol{d}_1^{(1)}\}$ 上依照 (4-3) 构造初始的 1 维二次插值模型 Q_1^{sub}. 细节在初始化步的算法伪代码中给出, 其编号为 2D-MoSub 的步 0.

基于 2 维子空间 $\boldsymbol{x}_k + \mathrm{span}\{\boldsymbol{d}_1^{(k)}, \boldsymbol{d}_2^{(k)}\}$ 中的插值集和 1 维二次插值模型 Q_k^{sub}, 2D-MoSub 将构造一个 2 维二次插值模型 Q_k. 模型 Q_k 将在由所选定方向张成的 2 维子空间中逼近目标函数. 此外, 如果通过在信赖域中极小化模型 Q_k 得到的迭代点没有达到充分小的函数值, 我们的方法将基于已探测的点构造修正后的模型 Q_k^{mod}. 我们将分别给出构造模型的细节.

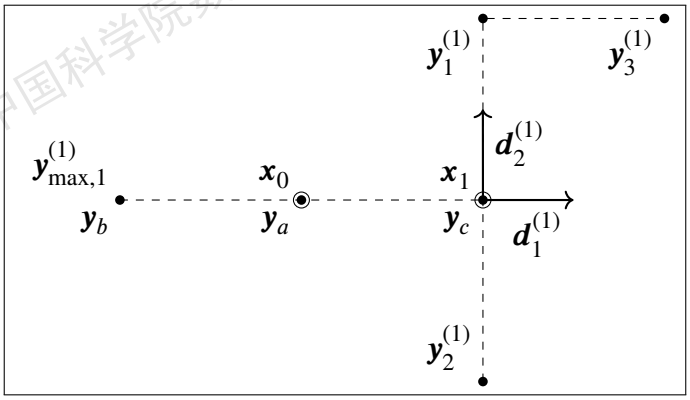


图 4-1 初始情况和子空间 $\boldsymbol{x}_1 + \mathrm{span}\{\boldsymbol{d}_1^{(1)}, \boldsymbol{d}_2^{(1)}\}$

Figure 4-1 The initial case and the subspace $\boldsymbol{x}_1 + \mathrm{span}\{\boldsymbol{d}_1^{(1)}, \boldsymbol{d}_2^{(1)}\}$

算法 8 步 1. 构造插值集

1: **输入:** $\boldsymbol{x}_k, \boldsymbol{d}_1^{(k)}, \Delta_k$

2: 选择一个满足 $\langle \boldsymbol{d}_1^{(k)}, \boldsymbol{d}_2^{(k)} \rangle = 0$ 的单位向量 $\boldsymbol{d}_2^{(k)}$.

3: 令 $\boldsymbol{y}_1^{(k)}$ 为

$$\boldsymbol{y}_1^{(k)} = \boldsymbol{x}_k + \Delta_k \boldsymbol{d}_2^{(k)}. \tag{4-5}$$

4: 根据 $f(\boldsymbol{y}_1^{(k)})$ 和 $f(\boldsymbol{x}_k)$ 的相对大小确定 $\boldsymbol{y}_2^{(k)}$:

$$\boldsymbol{y}_2^{(k)} = \begin{cases} \boldsymbol{x}_k + 2\Delta_k \boldsymbol{d}_2^{(k)}, \text{ 如果 } f(\boldsymbol{y}_1^{(k)}) \le f(\boldsymbol{x}_k), \\ \boldsymbol{x}_k - \Delta_k \boldsymbol{d}_2^{(k)}, \text{ 否则.} \end{cases} \tag{4-6}$$

5: 令 $\boldsymbol{y}_{\min,2}^{(k)}$ 为 $\boldsymbol{y}_1^{(k)}$ 和 $\boldsymbol{y}_2^{(k)}$ 中的函数值最小者, 即令

$$\boldsymbol{y}_{\min,2}^{(k)} = \arg \min_{\boldsymbol{y} \in \{\boldsymbol{y}_1^{(k)}, \boldsymbol{y}_2^{(k)}\}} f(\boldsymbol{y}).$$

6: 令 $\boldsymbol{y}_3^{(k)}$ 为

$$\boldsymbol{y}_3^{(k)} = \boldsymbol{y}_{\min,2}^{(k)} + \Delta_k \boldsymbol{d}_1^{(k)}. \tag{4-7}$$

在根据插值条件 (4-4) 讨论完初始 1 维模型 Q_1^{sub} 之后, 下面介绍我们如何基于 Q_k^{sub} 获得第 k 个模型 Q_k.

我们在 2 维子空间 $\boldsymbol{x}_k + \text{span}\{\boldsymbol{d}_1^{(k)}, \boldsymbol{d}_2^{(k)}\}$ 上构造 2 维二次插值模型 Q_k, 即

$$Q_k(\alpha, \beta) = f(\boldsymbol{x}_k) + a^{(k)}\alpha + b^{(k)}\alpha^2 + c^{(k)}\beta + d^{(k)}\beta^2 + e^{(k)}\alpha\beta, \tag{4-8}$$

其中系数 $a^{(k)}$ 和 $b^{(k)}$ 由 Q_k^{sub} 给出 (直接继承下来), 系数 $c^{(k)}, d^{(k)}$ 和 $e^{(k)}$ 由插值条件

$$Q_k(\mathcal{T}^{(k)}_{\boldsymbol{d}_1^{(k)}, \boldsymbol{d}_2^{(k)}}(\boldsymbol{z})) = f(\boldsymbol{z}), \ \forall\, \boldsymbol{z} \in \left\{ \boldsymbol{y}_1^{(k)}, \boldsymbol{y}_2^{(k)}, \boldsymbol{y}_3^{(k)} \right\} \tag{4-9}$$

确定. 细节在二次插值模型构造步的算法伪代码中给出, 编号为 2D-MoSub 的第 2 步.

算法 9 步 2. 构造二次插值模型

1: 在 2 维空间 $\boldsymbol{x}_k + \text{span}\{\boldsymbol{d}_1^{(k)}, \boldsymbol{d}_2^{(k)}\}$ 上构造 2 维二次插值模型 Q_k (如 (4-8) 所示), 其中 $c^{(k)}, d^{(k)}$, 和 $e^{(k)}$ 由插值条件 (4-9) 确定.

我们认为上述获取 Q_k 的方式是合理可靠的, 这是因为当 $\boldsymbol{x}_k$ 是一个成功的信赖域试探步时, $\boldsymbol{x}_k - \boldsymbol{x}_{k-1}$ 在数值上提供了一个足够好的方向和相应的 1 维子空间, 且在之前的 2 维子空间中的模型 Q_{k-1}^+ 沿着这样的 1 维子空间也是一个充分好的逼近. 这个 1 维子空间在大多数情况下是第 $k-1$ 个 2 维子空间和第 k 个 2 维子空间的交集. 换句话说, Q_k 在相应的 1 维子空间中遵循了与前一个较好模型的最优性一致性.

我们现在介绍我们的算法如何基于第 k 个模型函数 Q_k 获得 1 维子空间上的第 $k+1$ 个模型函数 Q_{k+1}^{sub}. 在第 k 步, 我们已经有了确定系数的模型函数 Q_k. 然而, 在我们获得迭代点 $\boldsymbol{x}_{k+1}$ 和函数值 $f(\boldsymbol{x}_{k+1})$ 之后, 通常有

$$Q_k(\mathcal{T}^{(k)}_{\boldsymbol{d}_1^{(k)},\boldsymbol{d}_2^{(k)}}(\boldsymbol{x}_{k+1})) \neq f(\boldsymbol{x}_{k+1}),$$

这表明 Q_k 在获取 $\boldsymbol{x}_{k+1}$ 处的信息后不是函数 f 的一个足够好的插值. 因此, 在获得模型 Q_{k+1}^{sub} 之前, 我们倾向于将 Q_k 更新为以下 Q_k^+, 即

$$Q_k^+(\alpha,\beta) = f(\boldsymbol{x}_{k+1}) + \bar{a}^{(k)}\alpha + \bar{b}^{(k)}\alpha^2 + \bar{c}^{(k)}\beta + \bar{d}^{(k)}\beta^2 + \bar{e}^{(k)}\alpha\beta, \tag{4-10}$$

它满足插值条件

$$Q_k^+(\mathcal{T}^{(k+1)}_{\boldsymbol{d}_1^{(k+1)},\boldsymbol{d}_*^{(k+1)}}(\boldsymbol{z})) = f(\boldsymbol{z}),\ \forall\, \boldsymbol{z} \in \left\{\boldsymbol{x}_{k-1}, \boldsymbol{x}_k, \boldsymbol{x}_{k+1}, \boldsymbol{y}_1^{(k)}, \boldsymbol{y}_2^{(k)}, \boldsymbol{y}_3^{(k)}\right\},$$

其中 $\boldsymbol{d}_*^{(k+1)} \in \text{span}\{\boldsymbol{d}_1^{(k)}, \boldsymbol{d}_2^{(k)}\}$ 满足 $\langle \boldsymbol{d}_*^{(k+1)}, \boldsymbol{d}_1^{(k+1)}\rangle = 0$. 我们在以下内容中给出构造二次插值模型 Q_k^+ 的更多细节.

注意, 在某些情况下, (4-10) 中的插值点集合并不是适定的. 因此, 2D-MoSub 将检查每个插值方程组的相应系数矩阵是否可逆. 如果当前的插值集在上述意义上不是适定的, 2D-MoSub 准备了不同的插值点作为备选, 2D-MoSub 将测试集合所对应插值方程的相应矩阵的可逆性, 并使用一个适定的集合来通过求解插值方程获得二次模型 Q_k^+. 备选的插值点集由从集合

$$\mathcal{Y}_k^+ = \left\{\boldsymbol{x}_{k-1}, \boldsymbol{x}_k, \boldsymbol{x}_{k+1}, \boldsymbol{y}_1^{(k)}, \boldsymbol{y}_2^{(k)}, \boldsymbol{y}_3^{(k)}, \boldsymbol{y}_4^{(k)}, \boldsymbol{y}_5^{(k)}\right\}$$

中挑选的 6 个点组成, 其中 $\boldsymbol{y}_4^{(k)} = \boldsymbol{x}_k + \frac{\sqrt{2}}{2}\Delta_k \boldsymbol{d}_1^{(k)} + \frac{\sqrt{2}}{2}\Delta_k \boldsymbol{d}_2^{(k)}$, 和 $\boldsymbol{y}_5^{(k)} = \boldsymbol{x}_k + \Delta_k \boldsymbol{d}_1^{(k)}$. 注意, 点 $\boldsymbol{x}_k, \boldsymbol{y}_1^{(k)}, \boldsymbol{y}_2^{(k)}, \boldsymbol{y}_3^{(k)}, \boldsymbol{y}_4^{(k)}, \boldsymbol{y}_5^{(k)}$ 有固定的分布, 因此这样的备选是可行的. 因此, 我们根据插值条件

$$Q_k^+(\mathcal{T}^{(k+1)}_{\boldsymbol{d}_1^{(k+1)},\boldsymbol{d}_*^{(k+1)}}(\boldsymbol{z})) = f(\boldsymbol{z}),\ \forall\, \boldsymbol{z} \in \mathcal{Y} \tag{4-11}$$

来获得 Q_k^+, 其中 $\mathcal{Y} \subset \mathcal{Y}_k^+$ 且 $\mathcal{Y}$ 是一个包含 6 个插值点的适定集合.

上述修正是为了使构造二次模型 Q_k^+ 时使用的插值集适定. 继而, 我们可以获得可靠的 2 维二次模型 Q_k^+. 然后, 1 维模型 Q_{k+1}^{sub} 被设置为

$$Q_{k+1}^{\text{sub}}(\alpha) = Q_k^+(\alpha, 0),\ \forall\, \alpha \in \mathfrak{R}. \tag{4-12}$$

事实上, 2D-MoSub 在一定意义上节省了计算成本, 这是因为它不需要执行一个复杂的子程序来实现传统算法中的模型改进.

注 4.2. 在成功步中还有另一种获得修正模型 Q_k^+ 的方法, 它基于更新的插值集 $\mathcal{X}_k^+ = \mathcal{X}_k \cup \{\boldsymbol{x}_{k+1}\}\backslash\{\boldsymbol{x}_{k-1}\}$ 使用最小范数更新方法, 在这种情况下, 我们只需要求解最小范数更新二次模型子问题对应的 KKT 方程, 用户可以自主选择如何进行修正.

如果通过求解模型函数 Q_k 的信赖域子问题得到的迭代点 $\boldsymbol{x}_k^+$ 在 $\boldsymbol{x}_k^+ \notin \{\boldsymbol{x}_{k-1}, \boldsymbol{x}_k, \boldsymbol{y}_1^{(k)}, \boldsymbol{y}_2^{(k)}, \boldsymbol{y}_3^{(k)}\}$ 的情况下无法足够好地实现函数值的减小, 我们的方法将重新求解修正后的二次模型 Q_k^{mod} 的信赖域子问题. 这时, 我们通过插值条件

$$Q_k^{\mathrm{mod}}(\mathcal{T}^{(k)}_{\boldsymbol{d}_1^{(k)}, \boldsymbol{d}_2^{(k)}}(\boldsymbol{z})) = f(\boldsymbol{z}),\ \forall\, \boldsymbol{z} \in \mathcal{Y}_k^{\mathrm{mod}} \tag{4-13}$$

来构造修正后的模型 Q_k^{mod}, 其中

$$\mathcal{Y}_k^{\mathrm{mod}} = \begin{cases} \left\{\boldsymbol{x}_{k-1}, \boldsymbol{x}_k, \boldsymbol{x}_k^+, \boldsymbol{y}_1^{(k)}, \boldsymbol{y}_2^{(k)}, \boldsymbol{y}_3^{(k)}\right\}, & \text{如果 } \boldsymbol{x}_k \neq \boldsymbol{x}_{k-1}, \\ \left\{\boldsymbol{x}_k, \boldsymbol{x}_k^+, \boldsymbol{y}_1^{(k)}, \boldsymbol{y}_2^{(k)}, \boldsymbol{y}_3^{(k)}, \boldsymbol{y}_4^{(k)}\right\}, & \text{如果 } \boldsymbol{x}_k = \boldsymbol{x}_{k-1} \text{ 且 } \boldsymbol{x}_k^+ \neq \boldsymbol{y}_4^{(k)}, \\ \left\{\boldsymbol{x}_k, \boldsymbol{x}_k^+, \boldsymbol{y}_1^{(k)}, \boldsymbol{y}_2^{(k)}, \boldsymbol{y}_3^{(k)}, \boldsymbol{y}_5^{(k)}\right\}, & \text{否则.} \end{cases}$$

注意, 以上所有插值点的函数值已经被探测过, 因此这样的插值不需要更多的函数值探测.

二次模型对于通过求解 2 维信赖域子问题获得的迭代的性质非常重要. 表 4-1 给出了我们的方法所用模型的插值条件.

表 4-1 2D-MoSub 中使用的模型的插值条件

Table 4-1 Interpolation conditions for models used in 2D-MoSub

模型	维数	插值条件
Q_k^{sub}	1	$Q_k^{\mathrm{sub}}(\alpha) = Q_{k-1}^+(\alpha, 0)$
Q_k	2	$Q_k(\mathcal{T}^{(k)}_{\boldsymbol{d}_1^{(k)}, \boldsymbol{d}_2^{(k)}}(\boldsymbol{z})) = f(\boldsymbol{z}),\ \forall\, \boldsymbol{z} \in \{\boldsymbol{y}_1^{(k)}, \boldsymbol{y}_2^{(k)}, \boldsymbol{y}_3^{(k)}\}\ \&\ Q_k(\alpha, 0) = Q_k^{\mathrm{sub}}(\alpha)$
Q_k^+	2	$Q_k^+(\mathcal{T}^{(k+1)}_{\boldsymbol{d}_1^{(k+1)}, \boldsymbol{d}_*^{(k+1)}}(\boldsymbol{z})) = f(\boldsymbol{z}),\ \forall\, \boldsymbol{z} \in \mathcal{Y},$ 其中 $\mathcal{Y} \subset \mathcal{Y}_k^+$ 且 $\lvert\mathcal{Y}\rvert = 6$
Q_k^{mod}	2	$Q_k^{\mathrm{mod}}(\mathcal{T}^{(k)}_{\boldsymbol{d}_1^{(k)}, \boldsymbol{d}_2^{(k)}}(\boldsymbol{z})) = f(\boldsymbol{z}),\ \forall\, \boldsymbol{z} \in \mathcal{Y}_k^{\mathrm{mod}}$

考虑到在通常情况下我们有 $\boldsymbol{d}_1^{(k)} = \frac{\boldsymbol{x}_k - \boldsymbol{x}_{k-1}}{\|\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\|_2}$, 并且它是一个近似梯度下降方向, 可以认为新模型 Q_k 沿着近似梯度下降方向继承了模型 Q_{k-1} 和 Q_{k-1}^+ 的良好性质.

和传统信赖域方法类似, 2D-MoSub 通过求解一个 2 维信赖域子问题来在对应的信赖域内找到最优的试探步. 然后, 它使用函数值改进量与模型值改进量的比率来评估试探步的质量. 根据比率和预定义的阈值, 算法更新子空间、插值集、信赖域参数和迭代点.

算法 10 步 3. 信赖域试探步

1: 求解信赖域子问题

$$\min_{\alpha,\beta}\ Q_k(\alpha, \beta)$$

$$\text{s. t. } \alpha^2 + \beta^2 \leq \Delta_k^2$$

并获得 $\alpha^{(k)}$ 和 $\beta^{(k)}$. 然后令

$$\begin{aligned} \boldsymbol{x}_k^{\mathrm{pre}} &= \boldsymbol{x}_k + \alpha^{(k)}\boldsymbol{d}_1^{(k)} + \beta^{(k)}\boldsymbol{d}_2^{(k)}, \\ \boldsymbol{x}_k^{+} &= \min_{\boldsymbol{x}\in\{\boldsymbol{x}_k, \boldsymbol{x}_k^{\mathrm{pre}}, \boldsymbol{y}_1^{(k)}, \boldsymbol{y}_2^{(k)}, \boldsymbol{y}_3^{(k)}\}} f(\boldsymbol{x}). \end{aligned}$$

2: 若 $\boldsymbol{x}_k^{+} \in \{\boldsymbol{x}_k, \boldsymbol{x}_{k-1}\}$, 则令 $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k$, $\Delta_{k+1} = \Delta_k$ 而不依照 (4-14), 令 $\boldsymbol{d}_1^{(k+1)} = \boldsymbol{d}_1^{(k)}$ 而不依照 (4-15), 然后转到 2D-MoSub 框架的**步 4**.

3: 否则, 计算

$$\rho_k = \frac{f(\boldsymbol{x}_k^{+}) - f(\boldsymbol{x}_k)}{Q_k(\mathcal{T}^{(k)}_{\boldsymbol{d}_1^{(k)}, \boldsymbol{d}_2^{(k)}}(\boldsymbol{x}_k^{+})) - Q_k(0,0)}.$$

4: 若 $\rho_k \geq \eta$ 或 $\boldsymbol{x}_k^{+} \in \{\boldsymbol{y}_1^{(k)}, \boldsymbol{y}_2^{(k)}, \boldsymbol{y}_3^{(k)}\}$, 则令 $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k^{+}$ 并转到 2D-MoSub 框架的**步 4**.

5: 否则, 通过 (4-13) 获得修正后的模型 Q_k^{mod} 并求解信赖域子问题

$$\begin{aligned} &\min_{\alpha,\beta} Q_k^{\mathrm{mod}}(\alpha, \beta) \\ &\text{s. t. } \alpha^2 + \beta^2 \leq \Delta_k^2, \end{aligned}$$

进而获得 $\alpha^{(k,\mathrm{mod})}$ 和 $\beta^{(k,\mathrm{mod})}$. 然后令

$$\boldsymbol{x}_k^{\mathrm{mod}} = \boldsymbol{x}_k + \alpha^{(k,\mathrm{mod})}\boldsymbol{d}_1^{(k)} + \beta^{(k,\mathrm{mod})}\boldsymbol{d}_2^{(k)}.$$

6: 若 $\boldsymbol{x}_k^{\mathrm{mod}} \in \{\boldsymbol{x}_k, \boldsymbol{x}_{k-1}\}$, 则令 $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k$, $\Delta_{k+1} = \Delta_k$ 而不依照 (4-14), 并且令 $\boldsymbol{d}_1^{(k+1)} = \boldsymbol{d}_1^{(k)}$ 而不依照 (4-15), 然后转到 2D-MoSub 框架的**步 4**.

7: 否则, 设置

$$\boldsymbol{x}_k^{+} = \arg\min_{\boldsymbol{x}\in\{\boldsymbol{x}_k^{+}, \boldsymbol{x}_k^{\mathrm{mod}}\}} f(\boldsymbol{x}).$$

计算

$$\rho_k = \frac{f(\boldsymbol{x}_k^{+}) - f(\boldsymbol{x}_k)}{Q_k(\mathcal{T}^{(k)}_{\boldsymbol{d}_1^{(k)}, \boldsymbol{d}_2^{(k)}}(\boldsymbol{x}_k^{+})) - Q_k(\boldsymbol{x}_k)}.$$

8: 若 $\rho_k \geq \eta_0$, 则令 $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k^{+}$ 并转到 2D-MoSub 框架的**步 4**.

9: 否则, 令 $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k$, $\boldsymbol{d}_1^{(k+1)} = \boldsymbol{d}_1^{(k)}$ 而不依照 (4-15), 然后转到 2D-MoSub 框架的**步 4**.

在信赖域试探步中, 我们执行以下操作. 首先, 我们求解信赖域子问题以找到二次模型 $Q_k(\alpha, \beta)$ 在信赖域内的极小点. 接下来, 我们基于 Q_k 和目标函数 f 计算相应的函数值减小量. 然后, 我们计算目标函数减小量与模型函数值减小量之间的比率, 并与预定义的阈值进行比较. 然后我们相应地更新 Δ_k 和 $\boldsymbol{d}_1^{(k)}$. 上述说明仅给出信赖域试探步的基本思想, 其他细节在算法伪代码中给出.

在信赖域试探步中, 我们按照算法框架第 3 步的伪代码执行. 在目前的测试实现中, 算法通过截断共轭梯度法 [161, 162] 求解信赖域子问题.

与传统的信赖域方法类似, 2D-MoSub 根据试探步的质量来更新信赖域半径. 如果半径小于下界, 2D-MoSub 终止. 否则, 算法通过基于更新后的解计算新方向来更新子空间. 2D-MoSub 在更新的子空间中构造一个新的 1 维二次插值模型. 细节在更新步的算法伪代码中给出, 这些步骤被列为 2D-MoSub 的第 4 步.

算法 11 步 4. 更新

1: 若 $\Delta_k < \Delta_{\text{low}}$, 则终止.

2: 否则, 更新 Δ_{k+1} 为

$$\Delta_{k+1} = \begin{cases} \gamma_1 \Delta_k, \text{如果 } \rho_k \geq \eta, \\ \gamma_2 \Delta_k, \text{否则}. \end{cases} \tag{4-14}$$

3: 令

$$\boldsymbol{d}_1^{(k+1)} = \frac{\boldsymbol{x}_{k+1} - \boldsymbol{x}_k}{\|\boldsymbol{x}_{k+1} - \boldsymbol{x}_k\|_2}. \tag{4-15}$$

4: 根据插值条件 (4-11) 更新 Q_k, 得到 Q_k^+.

5: 获取满足 (4-12) 的 1 维模型 Q_{k+1}^{sub}.

6: 令 $k = k + 1$, 转到 2D-MoSub 框架的**步 1**.

图 4-2 展示了 2D-MoSub 的第 k 步迭代情况.

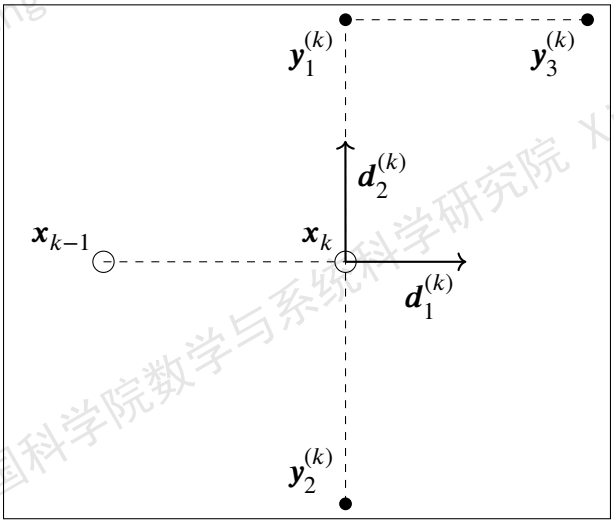


图 4-2 第 k 步的迭代情况和子空间 $\boldsymbol{x}_k + \text{span}\{\boldsymbol{d}_1^{(k)}, \boldsymbol{d}_2^{(k)}\}$

Figure 4-2 The iterative case at the k-th step and the subspace $\boldsymbol{x}_k + \text{span}\{\boldsymbol{d}_1^{(k)}, \boldsymbol{d}_2^{(k)}\}$

4.1.2 插值集的适定性和质量

在本节中, 我们讨论和分析了在每一步构造二次插值模型 Q_k 时插值集的适定性和质量. 正如我们所注意到的, 插值模型在相应区域的质量由插值点的位置

决定. 我们知道, Λ-适定性是衡量一组点分布良好程度的概念, 最终决定了插值模型对目标函数的逼近效果.

如前文所述, 衡量点在迭代当前步所感兴趣的区域中位置分布适定性的最常用的度量标准是基于 Lagrange 多项式给出的. 给定一组包含 p 个点的集合 $\mathcal{Y}=\{\boldsymbol{y}_1,\cdots,\boldsymbol{y}_p\}$, Lagrange 多项式的基满足

$$l_j(\boldsymbol{y}_i)=\begin{cases}1, \text{如果 } i=j,\\ 0, \text{否则.}\end{cases}$$

接下来, 我们简单地回顾 Λ-适定性的最基本经典定义 [20].

定义 4.2 (Λ-适定性). *如果一组点 $\mathcal{Y}$ 在集合 $\mathcal{B}$ 上线性独立, 并且与 $\mathcal{Y}$ 对应的 Lagrange 多项式 $\{l_1,\cdots,l_p\}$ 满足*

$$\Lambda\geq\max_{1\leq i\leq p}\max_{\boldsymbol{x}\in\mathcal{B}}\left|l_i(\boldsymbol{x})\right|.$$

对于我们的情况, 考虑到我们的算法在构造新的 2 维模型 Q_k 时有 3 个固定系数, 有 3 个待定系数要通过插值条件 (4-9) 来确定, 我们给出以下定义和讨论.

定义 4.3 (基函数). *给定 $\boldsymbol{y}_1^{(k)},\boldsymbol{y}_2^{(k)},\boldsymbol{y}_3^{(k)},\boldsymbol{x}_k$ 和 $\boldsymbol{d}_1^{(k)},\boldsymbol{d}_2^{(k)}\in\mathfrak{R}^n$, 令*

$$(\alpha_i^{(k)},\beta_i^{(k)})^{\top}=\mathcal{T}^{(k)}_{\boldsymbol{d}_1^{(k)},\boldsymbol{d}_2^{(k)}}(\boldsymbol{y}_i^{(k)}),\ i=1,2,3,$$

基矩阵[1]

$$\boldsymbol{\Psi}=\begin{pmatrix}\beta_1^{(k)} & (\beta_1^{(k)})^2 & \alpha_1^{(k)}\beta_1^{(k)}\\ \beta_2^{(k)} & (\beta_2^{(k)})^2 & \alpha_2^{(k)}\beta_2^{(k)}\\ \beta_3^{(k)} & (\beta_3^{(k)})^2 & \alpha_3^{(k)}\beta_3^{(k)}\end{pmatrix},$$

$c_0^{(i)}=f(\boldsymbol{x}_k)+a^{(k)}\alpha_i^{(k)}+b^{(k)}(\alpha_i^{(k)})^2, i=1,2,3,$ *以及*

$$\begin{pmatrix}h_1\\ d_1\\ e_1\end{pmatrix}=\boldsymbol{\Psi}^{-1}\begin{pmatrix}1-c_0^{(1)}\\ -c_0^{(1)}\\ -c_0^{(1)}\end{pmatrix},\ \begin{pmatrix}h_2\\ d_2\\ e_2\end{pmatrix}=\boldsymbol{\Psi}^{-1}\begin{pmatrix}-c_0^{(2)}\\ 1-c_0^{(2)}\\ -c_0^{(2)}\end{pmatrix},\ \begin{pmatrix}h_3\\ d_3\\ e_3\end{pmatrix}=\boldsymbol{\Psi}^{-1}\begin{pmatrix}-c_0^{(3)}\\ -c_0^{(3)}\\ 1-c_0^{(3)}\end{pmatrix}.$$

基函数是

$$l_i(\alpha,\beta)=c_0^{(i)}+h_i\beta+d_i\beta^2+e_i\alpha\beta,\ i=1,2,3. \tag{4-16}$$

现在我们定义我们情况下的 Λ-适定性.

定义 4.4 (2 维情况下有 3 个已知系数时的 Λ-适定性). *如果包含 3 个点的集合 $\mathcal{Y}\subset\mathfrak{R}^2$ 是线性独立的, 并且与 $\mathcal{Y}$ 相关的基多项式 $\{l_1,l_2,l_3\}$ 在 (4-16) 中满足*

$$\Lambda\geq\max_{1\leq i\leq 3}\max_{\|(\alpha,\beta)^{\top}\|_\infty\leq\Delta_k}|l_i(\alpha,\beta)|, \tag{4-17}$$

则我们称 $\mathcal{Y}$ 在集合 $\{(\alpha,\beta)^{\top},\|(\alpha,\beta)^{\top}\|_\infty\leq\Delta_k\}$ 上是 Λ-适定的,

[1]我们讨论可逆情况.

注 4.3. 在这种衡量标准下, 最适定的插值集是 1-适定的. 注意, 定义 4.4 中的区域是一个 ℓ_∞ 范数球, 这是为了在不失一般性的情况下获得解析解.

定理 4.5. *在每次迭代的二次插值模型构造步中, 2D-MoSub 具有以下用于计算的 Lagrange 基函数.*

在 $f(\boldsymbol{x}_k) \leq f(\boldsymbol{y}_1^{(k)})$ 的情况下, 我们有

$$\begin{cases} l_1(\alpha,\beta) = c_0^{(1)} + \dfrac{1}{2\Delta_k}\beta + \dfrac{1-2c_0^{(1)}}{2\Delta_k^2}\beta^2 + \left(-\dfrac{1}{\Delta_k^2}\right)\alpha\beta, \\ l_2(\alpha,\beta) = c_0^{(2)} + \left(-\dfrac{1}{2\Delta_k}\right)\beta + \dfrac{1-2c_0^{(2)}}{2\Delta_k^2}\beta^2, \\ l_3(\alpha,\beta) = c_0^{(3)} + \left(-\dfrac{c_0^{(3)}}{\Delta_k^2}\right)\beta^2 + \dfrac{1}{\Delta_k^2}\alpha\beta, \end{cases} \tag{4-18}$$

在 $f(\boldsymbol{x}_k) > f(\boldsymbol{y}_1^{(k)})$ 的情况下, 我们有

$$\begin{cases} l_1(\alpha,\beta) = c_0^{(1)} + \dfrac{4-3c_0^{(1)}}{2\Delta_k}\beta + \dfrac{-2+c_0^{(1)}}{2\Delta_k^2}\beta^2 + \left(-\dfrac{1}{\Delta_k^2}\right)\alpha\beta, \\ l_2(\alpha,\beta) = c_0^{(2)} + \left(-\dfrac{1+3c_0^{(2)}}{2\Delta_k}\right)\beta + \dfrac{1+c_0^{(2)}}{2\Delta_k^2}\beta^2, \\ l_3(\alpha,\beta) = c_0^{(3)} + \left(-\dfrac{3c_0^{(3)}}{2\Delta_k}\right)\beta + \dfrac{c_0^{(3)}}{2\Delta_k^2}\beta^2 + \dfrac{1}{\Delta_k^2}\alpha\beta, \end{cases} \tag{4-19}$$

其中 $c_0^{(1)} = f(\boldsymbol{x}_k)$, $c_0^{(2)} = f(\boldsymbol{x}_k)$, $c_0^{(3)} = f(\boldsymbol{x}_k) + a^{(k)}\Delta_k + b^{(k)}\Delta_k^2$.

证明. 在 $f(\boldsymbol{x}_k) \leq f(\boldsymbol{y}_1^{(k)})$ 的情况下, 我们有

$$\boldsymbol{\Psi}_1 = \begin{pmatrix} \Delta_k & \Delta_k^2 & 0 \\ -\Delta_k & \Delta_k^2 & 0 \\ \Delta_k & \Delta_k^2 & \Delta_k^2 \end{pmatrix}, \tag{4-20}$$

以及

$$\begin{pmatrix} h_1 \\ d_1 \\ e_1 \end{pmatrix} = \boldsymbol{\Psi}_1^{-1}\begin{pmatrix} 1-c_0^{(1)} \\ -c_0^{(1)} \\ -c_0^{(1)} \end{pmatrix}, \begin{pmatrix} h_2 \\ d_2 \\ e_2 \end{pmatrix} = \boldsymbol{\Psi}_1^{-1}\begin{pmatrix} -c_0^{(2)} \\ 1-c_0^{(2)} \\ -c_0^{(2)} \end{pmatrix}, \begin{pmatrix} h_3 \\ d_3 \\ e_3 \end{pmatrix} = \boldsymbol{\Psi}_1^{-1}\begin{pmatrix} -c_0^{(3)} \\ -c_0^{(3)} \\ 1-c_0^{(3)} \end{pmatrix}.$$

在 $f(\boldsymbol{x}_k) > f(\boldsymbol{y}_1^{(k)})$ 的情况下, 我们有

$$\boldsymbol{\Psi}_2 = \begin{pmatrix} \Delta_k & \Delta_k^2 & 0 \\ 2\Delta_k & 4\Delta_k^2 & 0 \\ \Delta_k & \Delta_k^2 & \Delta_k^2 \end{pmatrix}, \tag{4-21}$$

以及

$$\begin{pmatrix} h_1 \\ d_1 \\ e_1 \end{pmatrix} = \boldsymbol{\Psi}_2^{-1} \begin{pmatrix} 1 - c_0^{(1)} \\ -c_0^{(1)} \\ -c_0^{(1)} \end{pmatrix}, \begin{pmatrix} h_2 \\ d_2 \\ e_2 \end{pmatrix} = \boldsymbol{\Psi}_2^{-1} \begin{pmatrix} -c_0^{(2)} \\ 1 - c_0^{(2)} \\ -c_0^{(2)} \end{pmatrix}, \begin{pmatrix} h_3 \\ d_3 \\ e_3 \end{pmatrix} = \boldsymbol{\Psi}_2^{-1} \begin{pmatrix} -c_0^{(3)} \\ -c_0^{(3)} \\ 1 - c_0^{(3)} \end{pmatrix}.$$

因此, 我们可以得出结论 (4-18) 和 (4-19). □

图 4-3 展示了 $\boldsymbol{y}_1^{(k)}, \boldsymbol{y}_2^{(k)}, \boldsymbol{y}_3^{(k)}$ 的不同情况.

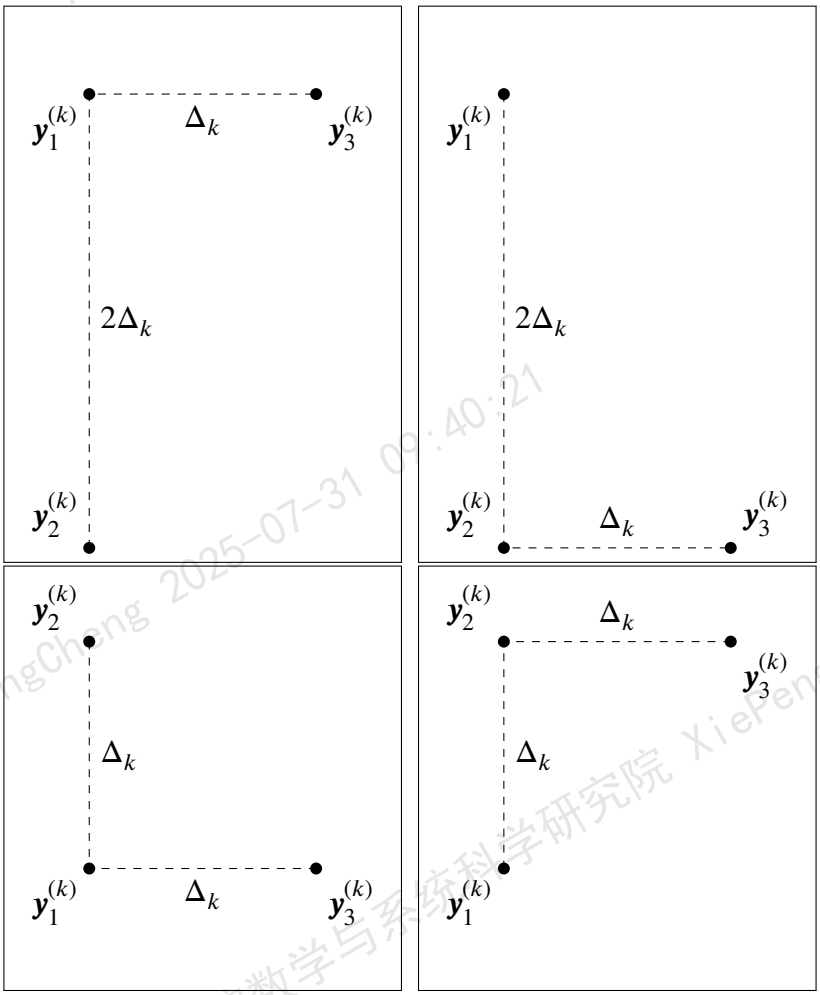


图 4-3 $\boldsymbol{y}_1^{(k)}, \boldsymbol{y}_2^{(k)}, \boldsymbol{y}_3^{(k)}$ 的不同情况

Figure 4-3 Different cases for $\boldsymbol{y}_1^{(k)}, \boldsymbol{y}_2^{(k)}, \boldsymbol{y}_3^{(k)}$

命题 4.6. *在 $f(\boldsymbol{x}_k) \leq f(\boldsymbol{y}_1^{(k)})$ 的情况下, 插值集 $\{\boldsymbol{y}_1^{(k)}, \boldsymbol{y}_2^{(k)}, \boldsymbol{y}_3^{(k)}\}$ 是 Λ_1-适定的, 其中 $\Lambda_1 = 2$. 在 $f(\boldsymbol{x}_k) > f(\boldsymbol{y}_1^{(k)})$ 的情况下, 插值集 $\{\boldsymbol{y}_1^{(k)}, \boldsymbol{y}_2^{(k)}, \boldsymbol{y}_3^{(k)}\}$ 是 Λ_2-适定的, 其中 $\Lambda_2 \leq \max\{4, 1 + 3\Delta_k(|a^{(k)}| + |b^{(k)}|\Delta_k)\}$.*

证明. 根据定理 4.5 和本命题中两种情况的 2 维问题

$$\max_{1 \leq i \leq 3} \max_{\|(\alpha,\beta)^\top\|_\infty \leq \Delta_k} \left| l_i(\alpha, \beta) \right|$$

的解析解, 结论经过计算后可直接得到. □

考虑到注 4.3, 上述由 2D-MoSub 算法使用的插值集在 2 维子空间上足够适定.

4.1.3 2D-MoSub 的一些性质

我们新提出的子空间无导数优化方法 2D-MoSub 的主要思想是每一步在一个 2 维子空间的信赖域内通过极小化二次模型函数来获得迭代点. 其中当前 2 维子空间中的二次模型和在 2 维子空间的一个维度中定义的模型继承了先前子空间、模型和迭代点的良好性质.

为了在每一步构造一个确定的二次插值模型函数, 2D-MoSub 获取了 3 个新的插值点来得到 2 维二次模型的其他 3 个未确定的系数.

接下来讨论我们方法的相关性质. 注意, 在理论上, 对于 $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha^{(k)}\boldsymbol{d}_1^{(k)} + \beta^{(k)}\boldsymbol{d}_2^{(k)}$, $\alpha^{(k)}$ 和 $\beta^{(k)}$ 满足

$$(\alpha^{(k)}, \beta^{(k)})^\top \in \left\{ \arg\min_{\alpha,\beta} \ Q_k(\alpha, \beta), \ \text{s. t. } \alpha^2 + \beta^2 \le \Delta_k^2 \right\}.$$

我们有以下命题.

命题 4.7. 我们有

$$\begin{cases} \min\limits_{\alpha^2+\beta^2\le\Delta_k^2} \ Q_k^+(\mathcal{T}^{(k+1)}_{\boldsymbol{d}_1^{(k+1)},\boldsymbol{d}_*^{(k+1)}}(\boldsymbol{x}_k + \alpha\boldsymbol{d}_1^{(k)} + \beta\boldsymbol{d}_2^{(k)})) \le f(\boldsymbol{x}_{k+1}), \\ \min\limits_{-\Delta_k\le\alpha\le\Delta_k} \ Q_{k+1}^{\text{sub}}(\hat{\mathcal{T}}^{(k+1)}_{\boldsymbol{d}_1^{(k+1)}}(\boldsymbol{x}_k + \alpha\boldsymbol{d}_1^{(k+1)})) \le f(\boldsymbol{x}_{k+1}), \\ \min\limits_{\alpha^2+\beta^2\le\Delta_{k+1}^2} \ Q_{k+1}(\mathcal{T}^{(k+1)}_{\boldsymbol{d}_1^{(k+1)},\boldsymbol{d}_2^{(k+1)}}(\boldsymbol{x}_{k+1} + \alpha\boldsymbol{d}_1^{(k+1)} + \beta\boldsymbol{d}_2^{(k+1)})) \le f(\boldsymbol{x}_{k+1}). \end{cases} \tag{4-22}$$

证明. 根据定义, 命题直接得证. □

按照前文所述的方式更新我们的新模型有两个优势. 一个优势是模型 Q_{k+1} 充分考虑了 Q_k 沿 1 维子空间 $\boldsymbol{x}_{k+1} + \text{span}\{\boldsymbol{d}_1^{(k+1)}\}$ 的性质, 这是因为 $\boldsymbol{x}_{k+1}$ 本身已经是由 Q_k 给出的成功步. 另一个优势是在信赖域上极小化 Q_{k+1} 可以根据 (4-22) 得到具有非递增模型函数值的迭代点.

此外, 由 2D-MoSub 获得的二次模型 Q_k 正是子问题

$$\begin{aligned} &\min_{Q\in\mathcal{Q}} \int_{-\infty}^{\infty} \left(Q(\alpha, 0) - Q_{k-1}^+(\alpha, 0)\right)^2 d\alpha \\ &\text{s. t. } Q(\mathcal{T}^{(k)}_{\boldsymbol{d}_1^{(k)},\boldsymbol{d}_2^{(k)}}(\boldsymbol{z})) = f(\boldsymbol{z}), \ \forall\, \boldsymbol{z} \in \{\boldsymbol{x}_k, \boldsymbol{y}_1^{(k)}, \boldsymbol{y}_2^{(k)}, \boldsymbol{y}_3^{(k)}\} \end{aligned} \tag{4-23}$$

的解, 事实上, Q_k 沿方向 $\boldsymbol{d}_1^{(k)}$ 与 Q_{k-1}^+ 相同, 这个方向其实是一个数值上的近似梯度下降方向.

接下来我们给出子问题 (4-23) 的凸性.

定理 4.8. *对于满足子问题 (4-23) 中插值条件的二次函数 Q, 子问题 (4-23) 是严格凸的.*

证明. 对于 $0 < c < 1$ 和满足 (4-23) 中插值条件的不同的 2 维二次函数 Q_a 和 Q_b, 我们有

$$\begin{aligned}&\int_{-\infty}^{\infty}\left(cQ_a(\alpha,0)+(1-c)Q_b(\alpha,0)-Q_{k-1}^{+}(\alpha,0)\right)^2 d\alpha\\ <\,&c\int_{-\infty}^{\infty}\left(Q_a(\alpha,0)-Q_{k-1}^{+}(\alpha,0)\right)^2 d\alpha+(1-c)\int_{-\infty}^{\infty}\left(Q_b(\alpha,0)-Q_{k-1}^{+}(\alpha,0)\right)^2 d\alpha.\end{aligned} \tag{4-24}$$

事实上, (4-24) 右侧与左侧的差为

$$\begin{aligned}&-2c(1-c)\int_{-\infty}^{\infty}Q_a(\alpha,0)Q_b(\alpha,0)d\alpha+(c-c^2)\int_{-\infty}^{\infty}(Q_a(\alpha,0))^2 d\alpha\\ &+(1-c-(1-c)^2)\int_{-\infty}^{\infty}(Q_b(\alpha,0))^2 d\alpha\\ =\,&(c-c^2)\int_{-\infty}^{\infty}(Q_a(\alpha,0)-Q_b(\alpha,0))^2 d\alpha<0,\end{aligned}$$

这是因为 $0 < c < 1$ 并且 $Q_a(\alpha,0) \neq Q_b(\alpha,0)$ (这可由 $Q_a \neq Q_b$ 且它们满足对应插值条件推得). 综上, 我们得到子问题目标函数的严格凸性. □

上述定理表明, 通过 2D-MoSub 获得的模型 Q_k 正是子问题 (4-23) 的唯一解.

以上内容介绍了当 $\boldsymbol{d}_1^{(k)}$ 是近似梯度下降方向时, 2D-MoSub 算法的优势. 我们还可以给出如下结论.

定理 4.9. *若 Q_k 是子问题 (4-23) 的解, 则对于二次函数 f, 有*

$$\begin{aligned}\int_{-\infty}^{\infty}\left(Q_k(\alpha,0)-\tilde{f}(\alpha,0)\right)^2 d\alpha=&\int_{-\infty}^{\infty}\left(Q_{k-1}^{+}(\alpha,0)-\tilde{f}(\alpha,0)\right)^2 d\alpha\\ &-\int_{-\infty}^{\infty}\left(Q_k(\alpha,0)-Q_{k-1}^{+}(\alpha,0)\right)^2 d\alpha,\end{aligned} \tag{4-25}$$

其中 $\tilde{f}=f\circ(\mathcal{T}^{(k)}_{\boldsymbol{d}_1^{(k)},\boldsymbol{d}_2^{(k)}})^{-1}$.

证明. 设 $Q_t = Q_k + t\left(Q_k - \tilde{f}\right)$, 其中 $t \in \mathfrak{R}$. 则函数 Q_t 是一个二次函数, 并且满足子问题 (4-23) 的插值条件. 根据 Q_k 的最优性, 我们可以知道二次函数

$$\varphi(t) := \int_{-\infty}^{\infty}\left(Q_t(\alpha,0)-Q_{k-1}^{+}(\alpha,0)\right)^2 d\alpha$$

在 $t = 0$ 时达到其极小值. 我们展开 $\varphi(t)$, 进而得到

$$\begin{aligned}\varphi(t)=&t^2\int_{-\infty}^{\infty}\left(Q_k(\alpha,0)-\tilde{f}(\alpha,0)\right)^2 d\alpha\\ &+2t\int_{-\infty}^{\infty}\left(Q_k(\alpha,0)-\tilde{f}(\alpha,0)\right)\left(Q_k(\alpha,0)-Q_{k-1}^{+}(\alpha,0)\right)d\alpha\\ &+\int_{-\infty}^{\infty}\left(Q_k(\alpha,0)-Q_{k-1}^{+}(\alpha,0)\right)^2 d\alpha,\end{aligned}$$

因此

$$\int_{-\infty}^{\infty} \left(Q_k(\alpha,0) - \tilde{f}(\alpha,0)\right)\left(Q_k(\alpha,0) - Q_{k-1}^{+}(\alpha,0)\right) d\alpha = 0.$$

考虑 $\varphi(-1)$, 结论即得证. □

此外, 以下推论也成立.

推论 4.10. *若 Q_k 是子问题 (4-23) 的解, 则对于二次函数 f, 有*

$$\int_{-\infty}^{\infty} \left(Q_k(\alpha,0) - \tilde{f}(\alpha,0)\right)^2 d\alpha \le \int_{-\infty}^{\infty} \left(Q_{k-1}^{+}(\alpha,0) - \tilde{f}(\alpha,0)\right)^2 d\alpha, \tag{4-26}$$

其中 $\tilde{f} = f \circ (\mathcal{T}^{(k)}_{\boldsymbol{d}_1^{(k)},\boldsymbol{d}_2^{(k)}})^{-1}$.

证明. 根据等式 (4-25) 和不等式

$$\int_{-\infty}^{\infty} \left(Q_k(\alpha,0) - Q_{k-1}^{+}(\alpha,0)\right)^2 d\alpha \ge 0,$$

不等式 (4-26) 成立. □

上述内容表明, 我们的 2D-MoSub 算法给出的模型 Q_k 沿方向 $\boldsymbol{d}_1^{(k)}$ 在一定意义上有着比上一步修正后的模型 Q_{k-1}^{+} 对目标函数的更好近似.

接下来, 我们介绍 2D-MoSub 的函数值下降情况. 我们有如下命题.

命题 4.11. *2D-MoSub 给出的迭代点的函数值递减, 即*

$$f(\boldsymbol{x}_{k+1}) \le f(\boldsymbol{x}_k),$$

并且在成功步中, 有

$$f(\boldsymbol{x}_{k+1}) \le f(\boldsymbol{x}_k) - \eta\left(Q_k(0,0) - Q_k(\mathcal{T}^{(k)}_{\boldsymbol{d}_1^{(k)},\boldsymbol{d}_2^{(k)}}(\boldsymbol{x}_{k+1}))\right)$$

成立.

证明. 根据算法框架和与 (4-1) 相关的准则, 上述结论可直接得到. □

下面的结果给出了 2D-MoSub 的收敛性. 在给出结论之前, 我们先给出如下基本假设.

假设 4.12. 目标函数 f 有下界、二次连续可微, 并且其二阶导数是有界的. 设 $\boldsymbol{C}$ 表示 $\left\|\nabla^2 f\right\|_2$ 的上界. 存在一个无穷集合 $\mathcal{I} \subseteq \mathbb{N}^+$, 使得

(1) 存在 $\varepsilon_1 > 0$ 使得 $\left\|\boldsymbol{P}_k \nabla f(\boldsymbol{x}_k)\right\|_2 \ge \varepsilon_1 \left\|\nabla f(\boldsymbol{x}_k)\right\|_2$ 对于 $k \in \mathcal{I}$ 成立, 其中 $\boldsymbol{P}_k$ 是从 $\mathfrak{R}^n$ 到 2 维子空间 $S^{(k)}_{\boldsymbol{d}_1^{(k)},\boldsymbol{d}_2^{(k)}}$ 的正交投影;

(2) $f(\boldsymbol{x}_{k+1}) - \inf_{\boldsymbol{d} \in S^{(k)}_{\boldsymbol{d}_1^{(k)},\boldsymbol{d}_2^{(k)}}} f(\boldsymbol{x}_k + \boldsymbol{d}) \to 0$ 在 $k \in \mathcal{K}$ 并且 $k \to \infty$ 时成立.

以下关于我们方法收敛性的定理参考并扩展了张在坤的论文 [46] 中的定理 5.7.

定理 4.13. *若目标函数 f 和 2D-MoSub 满足假设 4.12, 则*

$$\liminf_{k\to\infty} \left\|\nabla f(\boldsymbol{x}_k)\right\|_2 = 0,$$

其中每个 $\boldsymbol{x}_k$ 是由 2D-MoSub 生成的迭代点.

证明. 这里的证明参考并扩展了张在坤的论文 [46] 中定理 5.7 的证明. 事实上, 我们想证明: 当 $k \in \mathcal{I}$ 并且 $k \to \infty$ 时, 有 $\left\|\nabla f(\boldsymbol{x}_k)\right\|_2 \to 0$. 我们通过反证法来证明. 假设该结论不成立, 则存在 $\varepsilon_2 > 0$ 和 $\mathcal{I}$ 的一个无穷子集 $\mathcal{I}_{\text{sub}}$ 使得: 对于 $k \in \mathcal{I}_{\text{sub}}$, $\left\|\nabla f(\boldsymbol{x}_k)\right\|_2 \geq \varepsilon_2$. 根据假设 4.12, 不失一般性, 我们可以假设对于 $k \in \mathcal{I}_{\text{sub}}$, 有

$$\left\|\boldsymbol{P}_k \nabla f(\boldsymbol{x}_k)\right\|_2 \geq \varepsilon_1 \left\|\nabla f(\boldsymbol{x}_k)\right\|_2,$$

并且对于任意 $k \in \mathcal{I}_{\text{sub}}$, 有

$$f(\boldsymbol{x}_{k+1}) - \inf_{x \in S^{(k)}_{\boldsymbol{d}_1^{(k)}, \boldsymbol{d}_2^{(k)}}} f(\boldsymbol{x}_k + \boldsymbol{d}) \leq \frac{\varepsilon_1^2 \varepsilon_2^2}{4C}. \tag{4-27}$$

基于张在坤论文 [46] 中的引理 5.5, 我们有

$$f(\boldsymbol{x}_k) - \inf_{\boldsymbol{d} \in S^{(k)}_{\boldsymbol{d}_1^{(k)}, \boldsymbol{d}_2^{(k)}}} f(\boldsymbol{x}_k + \boldsymbol{d}) \geq \frac{1}{2C} \left\|\boldsymbol{P}_k \nabla f(\boldsymbol{x}_k)\right\|_2^2. \tag{4-28}$$

将 $f(\boldsymbol{x}_k)$ 和 $f(\boldsymbol{x}_{k+1})$ 这两项分别减去 $\inf_{\boldsymbol{d} \in S^{(k)}_{\boldsymbol{d}_1^{(k)}, \boldsymbol{d}_2^{(k)}}} f(\boldsymbol{x}_k + \boldsymbol{d})$ 后作差, 结合 (4-27) 和 (4-28) 可以得到

$$f(\boldsymbol{x}_k) - f(\boldsymbol{x}_{k+1}) \geq \frac{1}{2C} \left\|\boldsymbol{P}_k \nabla f(\boldsymbol{x}_k)\right\|_2^2 - \frac{\varepsilon_1^2 \varepsilon_2^2}{4C} \geq \frac{\varepsilon_1^2 \varepsilon_2^2}{2C} - \frac{\varepsilon_1^2 \varepsilon_2^2}{4C} = \frac{\varepsilon_1^2 \varepsilon_2^2}{4C}.$$

而这与 $\mathcal{I}_{\text{sub}}$ 是一个无穷子集并且 f 有下界这一事实矛盾. □

注意, 假设 4.12 是我们的 2D-MoSub 方法收敛的一个充分条件. 我们在下面不加证明地给出张在坤的论文 [46] 中的引理 5.5 的结果.

命题 4.14. *假设目标函数 f 有下界、二次连续可微, 且其二阶导数有界. 设 C 表示 $\left\|\nabla^2 f\right\|$ 的上界, 且 S 是 $\mathfrak{R}^n$ 的一个子空间. 则有*

$$f(\boldsymbol{x}) - \inf_{\boldsymbol{d} \in S} f(\boldsymbol{x} + \boldsymbol{d}) \geq \frac{1}{2C} \left\|\boldsymbol{P} \nabla f(\boldsymbol{x})\right\|_2^2,$$

其中 $\boldsymbol{P}$ 是从 $\mathfrak{R}^n$ 到 S 的正交投影.

表 4-2 2D-MoSub 参数设置

Table 4-2 Parameter settings of 2D-MoSub

参数	值	描述
Δ_1	1	初始信赖域半径
Δ_{low}	1×10^{-4}	信赖域半径下界
Δ_{upper}	1×10^{4}	信赖域半径上界
γ_1	10	信赖域半径扩大因子
γ_2	0.1	信赖域半径缩小因子
η	0.2	成功步阈值
η_0	0.1	修正后的成功步阈值
$\boldsymbol{d}^{(1)}$	$(1, 0, \cdots, 0)$	初始方向

4.1.4 数值结果

我们现在给出一些实验结果来展示 2D-MoSub 算法在求解所测试优化问题上的表现. 表 4-2 给出了测试我们算法时的参数设置.

为了展示我们子空间方法的一般数值表现, 我们尝试求解一些经典测试问题, 并使用 Performance Profile 和 Data Profile 来呈现数值结果. 表 4-3 中列出了测试问题, 这些问题选自经典常见的无约束优化测试函数集合, 包括 CUTEr 和 CUTEst [177, 207] 等.

这里所测试的问题维数范围是 10 到 20000. 此外, 算法从相应的相同初始点 $\boldsymbol{x}_{\text{int}}$ 开始, Profile 图中的精度 τ 被分别设置为 10^{-1}、10^{-3} 和 10^{-5}. 我们将 2D-MoSub 与方法 Nelder-Mead [190]、NEWUOA [94]、DFBGN [141] 和 CMA-ES [131] 进行了比较. 注意, 这样选择所对比的算法的原因是, 除了 Nelder-Mead 和 NEWUOA 这两个经典无导数优化方法外, 算法 DFBGN 和 CMA-ES 都含有相应的处理大规模无导数优化问题的改进技术.

我们可以从图 4-4 和图 4-5 中观察到, 2D-MoSub 能够比所对比的其他算法更好地求解大部分问题. 注意, 为综合展现, 所测试的问题集中的问题有难有易 (与维数、起始点、函数结构有关). 可以发现, 对于简单问题, 这些方法能在较少的探测次数内进行求解, 差别不大 (Data Profile 在初始阶段即有对应专门设置的简单问题的上升现象), 而对较难问题的求解则有较大区别.

4.1.5 小结

本节提出了一种新的基于信赖域方法和子空间技术的大规模无导数优化方法 2D-MoSub. 我们的方法通过求解 2 维信赖域子问题来实现迭代. 此外, 我们在 2 维子空间中定义了在第 k 步具有 3 个已知系数时的插值集的 2 维子空间 Λ-适定性. 我们给出了算法的主要步骤并分析了其理论性质. 数值结果显示了使用它来求解无导数优化问题的数值优势. 未来的工作包括设计选择子空间的新策略和

对大规模有约束无导数优化问题的研究.

表 4-3 图 4-4 和图 4-5 对应的测试问题

Table 4-3 Test problems for Figure 4-4 and Figure 4-5

ARGLINA	ARGLINA4	ARGLINB	ARGLINC	ARGTRIG
ARWHEAD	BDQRTIC	BDQRTICP	BDVALUE	BROWNAL
BROYDN3D	BROYDN7D	BRYBND	CHAINWOO	CHEBQUAD
CHNROSNB	CHPOWELLB	CHPOWELLS	CHROSEN	COSINE
CRAGGLVY	CUBE	CURLY10	CURLY20	CURLY30
DIXMAANE	DIXMAANF	DIXMAANG	DIXMAANH	DIXMAANI
DIXMAANJ	DIXMAANK	DIXMAANL	DIXMAANM	DIXMAANN
DIXMAANO	DIXMAANP	DQRTIC	EDENSCH	ENGVAL1
ERRINROS	EXPSUM	EXTROSNB	EXTTET	FIROSE
FLETCBV2	FLETCBV3	FLETCHCR	FREUROTH	GENBROWN
GENHUMPS	GENROSE	INDEF	INTEGREQ	LIARWHD
LILIFUN3	LILIFUN4	MOREBV	MOREBVL	NCB20
NCB20B	NONCVXU2	NONCVXUN	NONDIA	NONDQUAR
PENALTY1	PENALTY2	PENALTY3	PENALTY3P	POWELLSG
POWER	ROSENBROCK	SBRYBND	SBRYBNDL	SCHMVETT
SCOSINE	SCOSINEL	SEROSE	SINQUAD	SPARSINE
SPARSQUR	SPMSRTLS	SROSENBR	STMOD	TOINTGSS
TOINTTRIG	TQUARTIC	TRIGSABS	TRIGSSQS	TRIROSE1
TRIROSE2	VARDIM	WOODS	-	-

中国科学院数学与系统科学研究院 XiePengCheng 2025-07-31 07:40:21

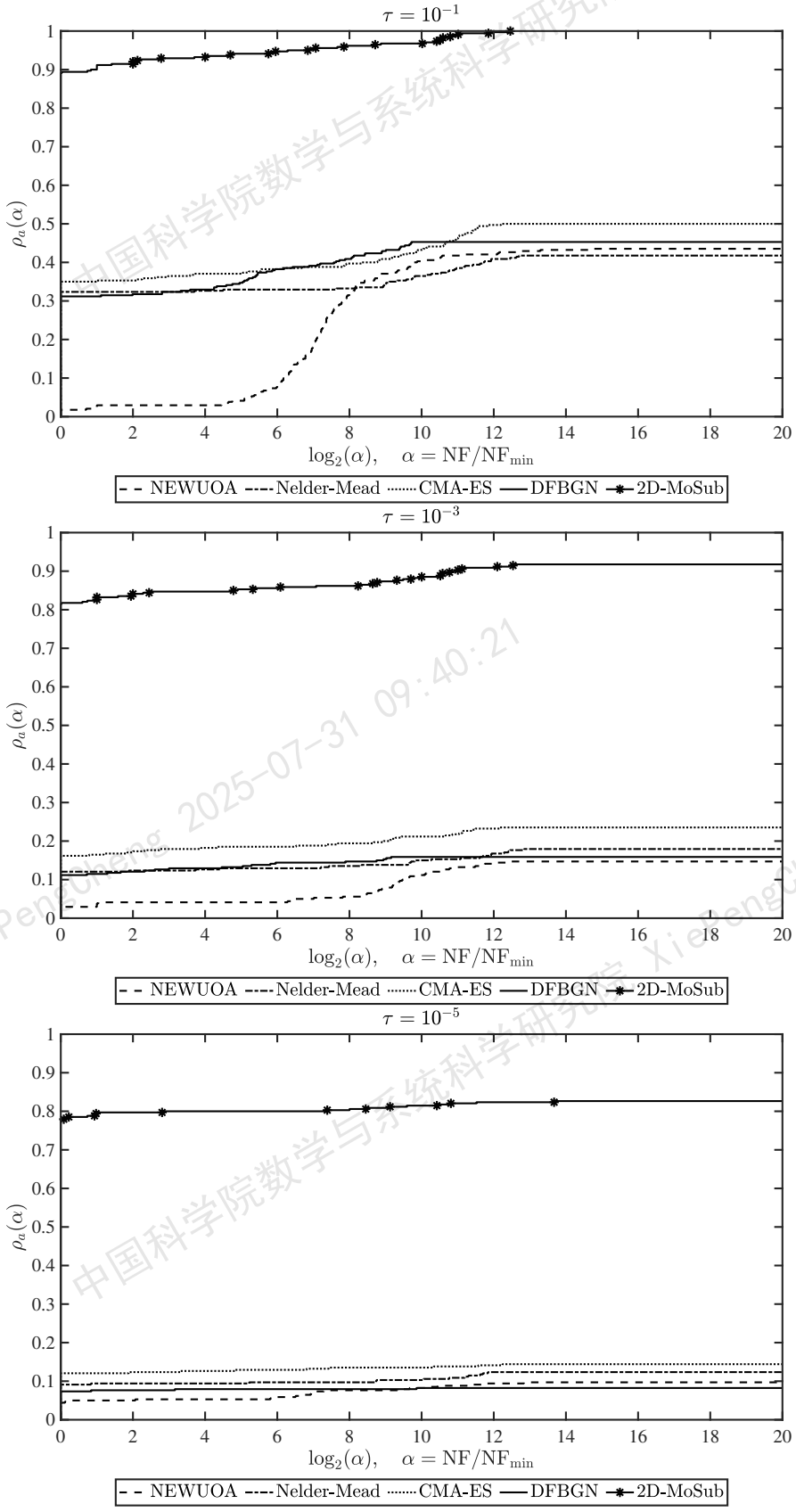


图 4-4 求解测试大规模问题的 Performance Profile

Figure 4-4 Performance Profile of solving test large-scale problems

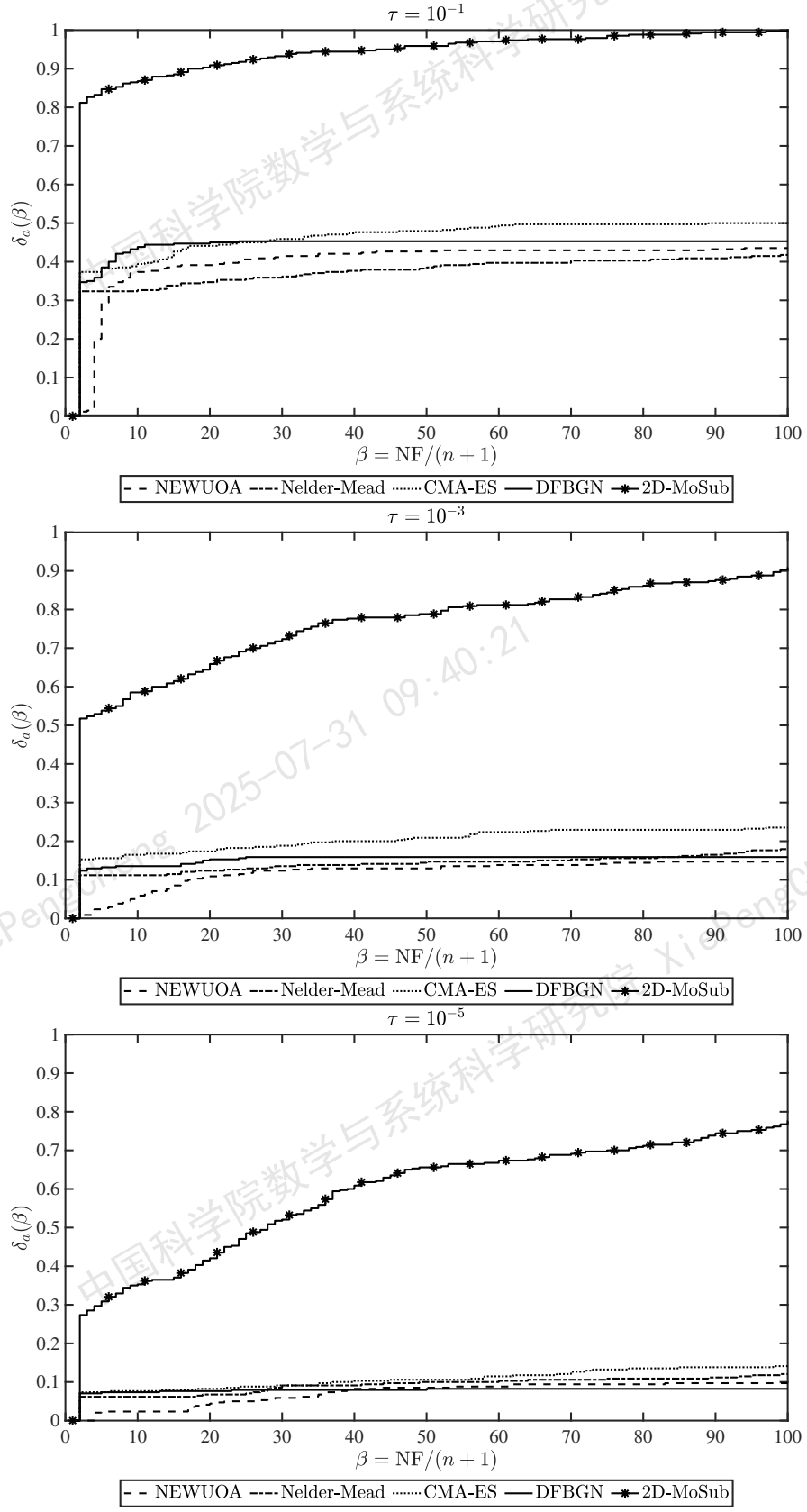


图 4-5 求解测试大规模问题的 Data Profile

Figure 4-5 Data Profile of solving test large-scale problems

4.2 结合线搜索和信赖域技术的无导数优化算法

加速减速 (SUSD) 方向是一种新的方向, 其已被证明在一定条件下趋于梯度下降方向. 本节将提出把基于插值点协方差矩阵的 SUSD 方向和当前迭代的插值模型函数的信赖域子问题解相结合的无导数优化算法 SUSD-TR. 我们将分析算法 SUSD-TR 的优化动力系统和搜索方向的稳定性. 给出试探步和结构步的详细信息. 数值结果将显示我们算法的优势, 并且数值上的对比表明 SUSD-TR 极大地提高了仅沿 SUSD 方向搜索迭代的方法的性能. 我们的算法与最先进的无导数优化算法相比是具有竞争力的.

4.2.1 背景和动机

考虑无约束优化问题

$$\min_{\boldsymbol{x}\in\mathfrak{R}^n} f(\boldsymbol{x}), \tag{4-29}$$

其中 $f:\mathfrak{R}^n\to\mathfrak{R}$ 是目标函数, 如前文所述, 我们没有它的任何导数信息. 我们知道, 线搜索方法和信赖域方法被广泛用于求解优化问题. 线搜索方法沿当前搜索方向寻找 (最优) 步长 (对于基于 SUSD 方向的算法来说是指这一组中每个点的步长). 然而, 当搜索方向不是很有效时, 线搜索方法可能导致收敛速度变慢. 相比之下, 信赖域方法通过构造局部二次模型函数来寻找模型在信赖域内的极小点, 可以在一定程度上克服线搜索方法中无效搜索方向的问题. 然而, 求解信赖域子问题需要更高的计算成本. Nocedal 和 Yuan [208] 讨论了在使用目标函数的梯度信息的优化中将线搜索方法和信赖域方法进行结合的方法.

对于一组探测点沿着共同的 SUSD 方向 [117] 的移动和迭代流程, 我们更倾向于将一组点的迭代过程称为大规模步. 事实上, 迭代点的内部结构 (尤其在大规模步中) 没有获得好的改进. 简而言之, 使用 SUSD 方向的算法通常在每一步 (同时) 探测一组迭代点, 而后根据这组点的分布确定出一个 SUSD 方向作为共同的前进方向, 同时根据这组点各自的函数值选取这组点中每个点沿该方向的前进步长.

我们的计划是在 (每次) 继续沿 SUSD 方向迭代之前, 通过在信赖域内选择一个更好的点并舍弃一个点, 修改这一组迭代点的结构, 并寻找局部插值模型在信赖域内的极小点. 这个新点可以修正甚至反转一组点的迭代方向, 尤其是在这样的方向产生较大偏离和误差的情况下. 我们考虑在大规模移动之外, 根据插值模型来修正迭代, 这样的做法结合了线搜索技术和信赖域技术.

后面部分的内容组织如下. 我们在第 4.2.2 节给出了 SUSD 方向与信赖域插值相结合的算法 SUSD-TR. 第 4.2.3 节给出了算法 SUSD-TR 的相应优化迭代的动力系统和搜索方向的稳定性. 在第 4.2.4 节中, 我们展示了算法所包含的试探步和结构步的更多细节.

4.2.2　SUSD 方向与信赖域插值的结合

我们的算法主要使用两大步骤来更新迭代点: 信赖域步和线搜索步. 信赖域步通过求解每一步的插值模型函数的信赖域子问题来得到. 线搜索步旨在沿 SUSD 方向推进这组迭代点, 后续讨论中记为 $\boldsymbol{v}_1$. 注意, 我们在本节会用到搜索点这一术语, 它其实就是当前迭代的那组探测点.

假设有 m 个搜索点, 每个搜索点是候选解 $\boldsymbol{x}_i \in \Re^n$, $i = 1, \cdots, m$. 我们要求 $m \geq n$, 其中 n 是优化问题 (4-29) 的维数. 我们定义协方差矩阵 $\boldsymbol{C} \in \Re^{n\times n}$ 为

$$\boldsymbol{C} = \sum_{i=1}^{m} \left(\boldsymbol{x}_i - \boldsymbol{x}_c\right)\left(\boldsymbol{x}_i - \boldsymbol{x}_c\right)^{\top}, \tag{4-30}$$

其中 $\boldsymbol{x}_c = \frac{1}{m}\sum_{i=1}^{m} \boldsymbol{x}_i$ 是搜索点的中心. 令 $\boldsymbol{v}_1, \cdots, \boldsymbol{v}_n$ 表示协方差矩阵 $\boldsymbol{C}$ 的单位特征向量, 它们与特征值 $\mu_1, \cdots, \mu_n$ 逐一对应, 注意, 这里我们按照从最小特征值 μ_1 到最大特征值 μ_n 进行排序 (假设 $\mu_1 \neq 0$). 向量 $\boldsymbol{v}_1$ 是 SUSD 方向 [117]. 算法 12 给出了基于 SUSD 方向和插值及信赖域技术的无导数优化算法 SUSD-TR 的算法流程. 在算法 12 中, $\bar{f}^{(k)}$ 是第 k 步插值点 (搜索点) 中的最小函数值. 这里的 k 用于表示迭代步, 我们将一些参数作为 k 的函数写出.

算法 12 SUSD-TR 算法

输入: 搜索点数 m, 初始点 $\boldsymbol{x}_i^{(0)}, i = 1, \cdots, m$, 参数 β, Δ_0, 终止参数 P, ε; $k = 0$

while $|\bar{f}^{(k)} - \frac{1}{P}\sum_{h=1}^{P} \bar{f}^{(k-h)}| > \varepsilon$ **do**

　用传统 PCA 算法计算 $\boldsymbol{C}^{(k)}$ 和 $\boldsymbol{v}_1^{(k)}$.

　for $i = 1, \cdots, m$ **do**

　　获取 $f_i^{(k)} := f(\boldsymbol{x}_i^{(k)})$.

　end for

　计算 $\Delta_k = \max_i(\|\boldsymbol{x}_i^{(k)} - \boldsymbol{x}_c^{(k)}\|_2)$.

　if $\Delta_k > \kappa\Delta_{k-1}$ **then**

　　结构步: 根据算法 13 用 $\boldsymbol{x}_{\text{new}}^{(k)}$ 替换距离信赖域中心最远的点 $\boldsymbol{x}_d^{(k)}$;

　else

　　模型改进步: 通过调用模型改进步 (即 Conn、Scheinberg 和 Vicente 著作 [20] 中的算法 6.3) 检查并改进插值集的适定性.

　　基于 m 个最新的搜索点生成当前插值集 $\mathcal{X}_k$, 并根据 (4-31) 或 (4-32) 构造线性插值模型函数 $L_k(\boldsymbol{x})$ 或 (欠定) 二次插值模型函数 $Q_k(\boldsymbol{x})$.

　　试探步: 使用截断共轭梯度法求解信赖域子问题

$$\begin{aligned} &\min_{\boldsymbol{x}} L_k(\boldsymbol{x}) \text{ 或 } Q_k(\boldsymbol{x}), \\ &\text{s. t. } \left\|\boldsymbol{x} - \boldsymbol{x}_c^{(k)}\right\|_2 \leq \Delta_k \end{aligned}$$

后, 用该子问题的数值解 $\boldsymbol{x}_{\text{new}}^{(k)}$ 替换 $\boldsymbol{x}_d^{(k)}$, 然后更新信赖域半径 Δ_k. 注意, 这里的 $\boldsymbol{x}_d^{(k)}$ 是当前步中函数值最大的迭代点, 它被 $\boldsymbol{x}_{\text{new}}^{(k)}$ 替换.

end if

计算 $\bar{f}^{(k)} := \min_i f_i^{(k)}$.

for $i = 1, \cdots, m$ **do**

计算 $\alpha(\boldsymbol{x}_i^{(k)}) = \beta[1 - \exp(\bar{f} - f(\boldsymbol{x}_i^{(k)}))]$.

线搜索步: 更新 $\boldsymbol{x}_i^{(k+1)} = \boldsymbol{x}_i^{(k)} + \alpha(\boldsymbol{x}_i^{(k)})\boldsymbol{v}_1^{(k)}$.

end for

令 $k = k + 1$.

end while

输出: $\boldsymbol{x}^* = \boldsymbol{x}_i$, 其中 $\boldsymbol{x}_i$ 是最后一步迭代点中函数值最小的点.

可以看到, SUSD-TR 是一种将线搜索和信赖域技术相结合的无导数优化算法. 这组搜索点会沿着方向 $\boldsymbol{v}_1$ 进行搜索, 这属于线搜索类型; 然后算法将求解信赖域子问题以进行修正, 进而形成算法的循环.

SUSD-TR 具有以下几个优点. 首先, 如果我们在不同的计算节点之间传输数据 (包括函数值等), 它可以实现分布式或并行化, 这是因为算法在线搜索步中可以同时在不同的 m 个搜索点处探测函数值. 这减小了函数值探测的时间成本, 特别是对于探测时间成本高的问题而言. 其次, 算法 12 不依赖于传统的梯度估计, 因此即使没有显式的梯度估计, 它仍然可以运行. 另外, 我们可以给出优化过程的动力系统表达, 进而借助连续化这一工具给出更多理论结果, 这在无导数优化中并不常见, 但却是新颖和重要的. 最后但同样重要的是, 与提供传统有限差分估计的点相比, 这些搜索点可以分布在更灵活的区域内.

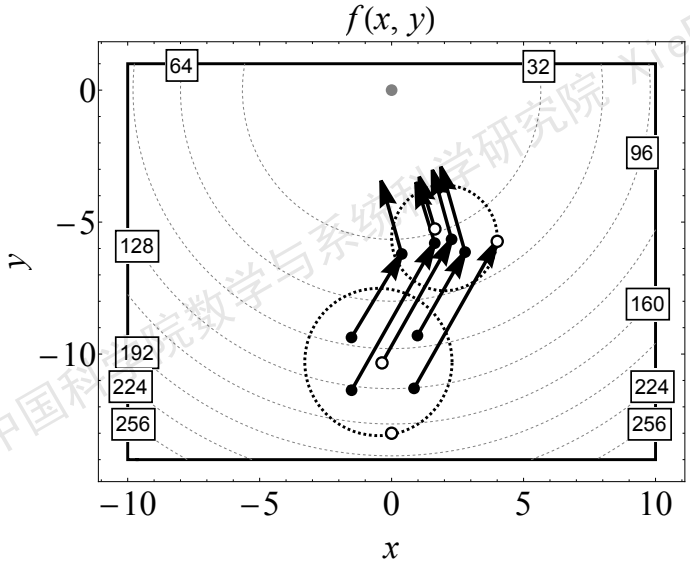


图 4-6 SUSD-TR 在 2 维问题中的一般框架示意图

Figure 4-6 Illustration of the general framework of SUSD-TR for a 2-dimensional problem

图 4-6 展示了算法 SUSD-TR 的迭代过程, 其中一个空圈点表示每一步中被舍弃的点 $\boldsymbol{x}_d$. 在第 k 步, 如果每次迭代的搜索 (样本) 点数少于 $\frac{1}{2}(n+1)(n+2)$, 通

过求解前文所提及过的子问题

$$\begin{aligned} &\min_{Q\in\mathcal{Q}} \left\|\nabla^2 Q - \nabla^2 Q_{k-1}\right\|_F^2 \\ &\text{s. t. } Q(\boldsymbol{x}_i) = f(\boldsymbol{x}_i),\ \forall\, \boldsymbol{x}_i \in \mathcal{X}_k \end{aligned} \tag{4-31}$$

可得到二次插值模型 Q_k; 如果我们分别获得和使用确定的线性或二次插值模型, 则可通过求解方程

$$\begin{aligned} f(\boldsymbol{x}_i) &= L_k(\boldsymbol{x}_i),\ \forall\, \boldsymbol{x}_i \in \mathcal{X}_k \\ \text{或 } f(\boldsymbol{x}_i) &= Q_k(\boldsymbol{x}_i),\ \forall\, \boldsymbol{x}_i \in \mathcal{X}_k \end{aligned} \tag{4-32}$$

来获得第 k 次模型函数 L_k 或 Q_k. 在数值实验中, 我们给出了使用欠定二次模型的 SUSD-TR 的对应结果. 请注意, 简明起见, 我们在这里和接下来的讨论中可能省略用以表示第 k 次迭代的符号 (k).

4.2.3　SUSD-TR 算法迭代方向的稳定性分析

算法 12 的优化过程的动力系统, 或者我们称之为梯度流, 可以转化为

$$\begin{cases} \dot{\boldsymbol{x}}_i = \alpha(\boldsymbol{x}_i)\boldsymbol{v}_1,\ i = 1, \cdots, d-1, d+1, \cdots, m, \\ \dot{\boldsymbol{x}}_d = \left(\alpha(\boldsymbol{x}_d) + \varepsilon_1\right)\left(\boldsymbol{v}_1 + \boldsymbol{\varepsilon}_2\right), \end{cases} \tag{4-33}$$

其中 $\varepsilon_1 \in \mathfrak{R}$ 和 $\boldsymbol{\varepsilon}_2 \in \mathfrak{R}^n$ 是 $\boldsymbol{x}_d$ 的扰动参数, 表示将通过信赖域技术更新为 $\boldsymbol{x}_{\text{new}}$. (4-33) 中的上标点表示对连续时间 t 的导数 (对应迭代 k). 步长 $\alpha : \mathfrak{R} \to \mathfrak{R}$ 是一个指数类型的映射[2] [117], 即

$$\alpha(\boldsymbol{x}_i) = \beta\left[1 - \exp\left(\bar{f} - f(\boldsymbol{x}_i)\right)\right],\ i = 1, \cdots, m, \tag{4-34}$$

其中 $\beta \in \mathfrak{R}$ 是一个正常数, $\bar{f}$ 是当前迭代中所有插值/搜索点的最小函数值. 简明起见, 我们将 $\alpha(\boldsymbol{x}_i)$ 记为 α_i.

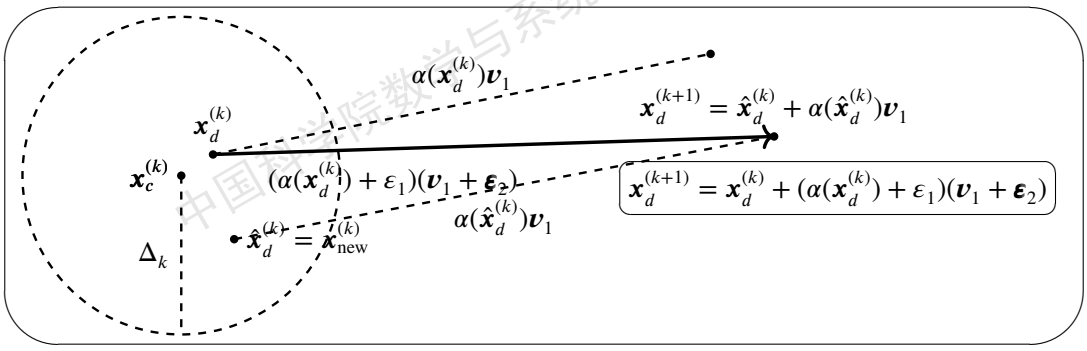


图 4-7 (4-33) 中的扰动

Figure 4-7 The disturbance in (4-33)

[2]步长也可以选择为线性类型的映射 [117], 本节分析的是使用指数类型的映射的情形. 线性类型的步长的对应结果与这里的结果整体相同, 因此我们不再提供更多细节.

图 4-7 刻画了具有离散迭代 (时间) 的动力系统 (4-33), 其中 $\boldsymbol{x}_d^{(k)}$ 在信赖域步中被 $\hat{\boldsymbol{x}}_d^{(k)}$ 替换, 然后 $\hat{\boldsymbol{x}}_d^{(k)}$ 在线搜索步中前进到 $\boldsymbol{x}_d^{(k+1)}$. 因此, 新的迭代点可以表示为

$$\boldsymbol{x}_d^{(k+1)} = \boldsymbol{x}_d^{(k)} + \left(\alpha(\boldsymbol{x}_d^{(k)}) + \varepsilon_1\right)\left(\boldsymbol{v}_1 + \boldsymbol{\varepsilon}_2\right),$$

其中 ε_1 表示步长的扰动, $\boldsymbol{\varepsilon}_2$ 表示 $\boldsymbol{x}_d^{(k)}$ 的方向扰动. 这与动力系统 (4-33) 对应. 接下来, 我们使用连续动力系统进行分析.

注 4.4. 对于使用 (4-34) 的指数映射步长, 当前步中具有最小函数值的点的下一步前进步长为 0, 这与使用线性映射步长的情况不同. 注意, 图 4-6 旨在给出一个 SUSD-TR 的一般框架的示例.

引理 4.15. *(4-33) 对应的 SUSD-TR 方向的动力系统为*

$$\dot{\boldsymbol{v}}_1 = \left(\sum_{j=2}^{n} \frac{1}{\mu_1 - \mu_j} \boldsymbol{v}_j \boldsymbol{v}_j^{\top}\right)\left[\sum_{i=1}^{m} \left(\alpha_i - \alpha_a\right)\left(\boldsymbol{x}_i - \boldsymbol{x}_c\right) + \varepsilon_1\left(\boldsymbol{x}_d - \boldsymbol{x}_c\right) + \Phi \boldsymbol{v}_1\right], \tag{4-35}$$

其中 $\boldsymbol{v}_j$ 是矩阵 $\boldsymbol{C}$ 的第 j 个单位特征向量,

$$\alpha_a = \frac{1}{m}\sum_{i=1}^{m} \alpha_i + \frac{\varepsilon_1}{m},$$

其中 $\alpha_i = \alpha(\boldsymbol{x}_i)$, 另

$$\Phi = \left(\alpha_d \boldsymbol{\varepsilon}_2 + \varepsilon_1 \boldsymbol{\varepsilon}_2\right)\left(\boldsymbol{x}_d - \boldsymbol{x}_c\right)^{\top} + \left(\boldsymbol{x}_d - \boldsymbol{x}_c\right)\left(\alpha_d \boldsymbol{\varepsilon}_2 + \varepsilon_1 \boldsymbol{\varepsilon}_2\right)^{\top}.$$

证明. 注意到 $\boldsymbol{x}_c = \frac{1}{m}\sum\limits_{i=1}^{m} \boldsymbol{x}_i$. 根据动力系统方程 (4-33), 我们得到

$$\dot{\boldsymbol{x}}_c = \alpha_a \boldsymbol{v}_1 + \frac{1}{m}\left(\alpha_d \boldsymbol{\varepsilon}_2 + \varepsilon_1 \boldsymbol{\varepsilon}_2\right),$$

其中 $\alpha_a = \frac{1}{m}\sum\limits_{i=1}^{m} \alpha_i + \frac{\varepsilon_1}{m}$. 取 (4-30) 对时间的导数, 我们得到

$$\begin{aligned}
\dot{\boldsymbol{C}} = & \sum_{i=1}^{m} \left(\alpha_i - \alpha_a\right)\left[\boldsymbol{v}_1\left(\boldsymbol{x}_i - \boldsymbol{x}_c\right)^{\top} + \left(\boldsymbol{x}_i - \boldsymbol{x}_c\right)\boldsymbol{v}_1^{\top}\right] \\
& - \frac{\alpha_d + \varepsilon_1}{m}\sum_{i=1}^{m}\left[\boldsymbol{\varepsilon}_2\left(\boldsymbol{x}_i - \boldsymbol{x}_c\right)^{\top} + \left(\boldsymbol{x}_i - \boldsymbol{x}_c\right)\boldsymbol{\varepsilon}_2^{\top}\right] \\
& + \left(\varepsilon_1 \boldsymbol{v}_1 + \alpha_d \boldsymbol{\varepsilon}_2 + \varepsilon_1 \boldsymbol{\varepsilon}_2\right)\left(\boldsymbol{x}_d - \boldsymbol{x}_c\right)^{\top} + \left(\boldsymbol{x}_d - \boldsymbol{x}_c\right)\left(\varepsilon_1 \boldsymbol{v}_1 + \alpha_d \boldsymbol{\varepsilon}_2 + \varepsilon_1 \boldsymbol{\varepsilon}_2\right)^{\top} \\
= & \sum_{i=1}^{m} \left(\alpha_i - \alpha_a\right)\left[\boldsymbol{v}_1\left(\boldsymbol{x}_i - \boldsymbol{x}_c\right)^{\top} + \left(\boldsymbol{x}_i - \boldsymbol{x}_c\right)\boldsymbol{v}_1^{\top}\right] \\
& + \left(\varepsilon_1 \boldsymbol{v}_1 + \alpha_d \boldsymbol{\varepsilon}_2 + \varepsilon_1 \boldsymbol{\varepsilon}_2\right)\left(\boldsymbol{x}_d - \boldsymbol{x}_c\right)^{\top} + \left(\boldsymbol{x}_d - \boldsymbol{x}_c\right)\left(\varepsilon_1 \boldsymbol{v}_1 + \alpha_d \boldsymbol{\varepsilon}_2 + \varepsilon_1 \boldsymbol{\varepsilon}_2\right)^{\top}.
\end{aligned} \tag{4-36}$$

另外, 基于 $\boldsymbol{C}\boldsymbol{v}_1 = \mu_1\boldsymbol{v}_1$, 我们有

$$\dot{\boldsymbol{C}}\boldsymbol{v}_1 + \boldsymbol{C}\dot{\boldsymbol{v}}_1 = \dot{\mu}_1\boldsymbol{v}_1 + \mu_1\dot{\boldsymbol{v}}_1.$$

我们有

$$\boldsymbol{v}_j^{\top}\dot{\boldsymbol{C}}\boldsymbol{v}_1 + \boldsymbol{v}_j^{\top}\boldsymbol{C}\dot{\boldsymbol{v}}_1 = \dot{\mu}_1\boldsymbol{v}_j^{\top}\boldsymbol{v}_1 + \mu_1\boldsymbol{v}_j^{\top}\dot{\boldsymbol{v}}_1, \tag{4-37}$$

并且矩阵 $\dot{\boldsymbol{C}}$ 是对称的. 这意味着

$$\boldsymbol{v}_j^{\top}\boldsymbol{C}\dot{\boldsymbol{v}}_1 = \left(\boldsymbol{C}\boldsymbol{v}_j\right)^{\top}\dot{\boldsymbol{v}}_1 = \mu_j\boldsymbol{v}_j^{\top}\dot{\boldsymbol{v}}_1.$$

另外, $\boldsymbol{v}_j^{\top}\boldsymbol{v}_1 = \boldsymbol{v}_1^{\top}\boldsymbol{v}_j = 0$, 我们从 (4-37) 可得

$$\boldsymbol{v}_j^{\top}\dot{\boldsymbol{v}}_1 = \frac{1}{\mu_1 - \mu_j}\boldsymbol{v}_j^{\top}\dot{\boldsymbol{C}}\boldsymbol{v}_1. \tag{4-38}$$

由于矩阵 $\boldsymbol{C}$ 是对称的, 我们有

$$\dot{\boldsymbol{v}}_1 = \sum_{j=2}^{n}\boldsymbol{v}_j^{\top}\dot{\boldsymbol{v}}_1\boldsymbol{v}_j. \tag{4-39}$$

将 (4-36) 代入 (4-38), 并根据 (4-39) 以及 $\boldsymbol{v}_j^{\top}\boldsymbol{v}_1(\boldsymbol{x}_i - \boldsymbol{x}_c)^{\top}\boldsymbol{v}_1 = 0$, 命题得证. □

设 $\alpha_c = \alpha(f(\boldsymbol{x}_c))$, 定义梯度 $\nabla\alpha = \nabla\alpha(\boldsymbol{x}_c)$. 我们使用在中心 $\boldsymbol{x}_c$ 的 Taylor 展开来近似 $\alpha_i = \alpha(\boldsymbol{x}_i)$, 即

$$\alpha_i - \alpha_c = \left(\boldsymbol{x}_i - \boldsymbol{x}_c\right)^{\top}\nabla\alpha + r_i, \tag{4-40}$$

其中 $\alpha_c = \alpha(\boldsymbol{x}_c)$ 并且 $r_i = \mathcal{O}\left(\|\boldsymbol{x}_i - \boldsymbol{x}_c\|_2^2\right)$. 假设 $f_c = f(\boldsymbol{x}_c)$. 设 $\nabla f = \nabla f(\boldsymbol{x}_c)$ 为函数 f 在中心 $\boldsymbol{x}_c$ 处的梯度. 我们得到以下引理.

引理 4.16. *根据 (4-33) 和 Taylor 展开, 我们得到*

$$\begin{aligned}\dot{\boldsymbol{v}}_1 = &\sum_{j=2}^{n}\frac{\mu_j}{\mu_1 - \mu_j}\boldsymbol{v}_j\boldsymbol{v}_j^{\top}\nabla\alpha + \boldsymbol{r} + \sum_{j=2}^{n}\frac{\varepsilon_1}{\mu_1 - \mu_j}\boldsymbol{v}_j\boldsymbol{v}_j^{\top}\left(\boldsymbol{x}_d - \boldsymbol{x}_c\right)\\ &+ \sum_{j=2}^{n}\frac{1}{\mu_1 - \mu_j}\boldsymbol{v}_j\boldsymbol{v}_j^{\top}\left(\Phi\boldsymbol{v}_1\right),\end{aligned} \tag{4-41}$$

其中

$$\boldsymbol{r} = \left(\sum_{j=2}^{n}\frac{1}{\mu_1 - \mu_j}\boldsymbol{v}_j\boldsymbol{v}_j^{\top}\right)\left[\sum_{i=1}^{m}r_i\left(\boldsymbol{x}_i - \boldsymbol{x}_c\right)\right].$$

证明. 假设 $r_a = \frac{1}{m}\sum_{i=1}^{m}r_i$. α_a 的定义和 (4-40) 蕴含 $\alpha_a = \alpha_c + r_a + \frac{\varepsilon_1}{m}$. 进而我们根据 (4-40) 得到

$$\sum_{i=1}^{m}\left(\alpha_i - \alpha_a\right)\left(\boldsymbol{x}_i - \boldsymbol{x}_c\right) = \boldsymbol{C}\nabla\alpha + \sum_{i=1}^{m}r_i\left(\boldsymbol{x}_i - \boldsymbol{x}_c\right). \tag{4-42}$$

将 (4-42) 代入 (4-35) 中, 并结合 $\boldsymbol{v}_j^{\top}\boldsymbol{C} = \mu_j\boldsymbol{v}_j^{\top}$, 我们得到 (4-41). □

我们给出以下引理, 这里我们继续用 ∇f 表示 $\nabla f(\boldsymbol{x}_c)$.

引理 4.17. *根据 (4-33) 和指数类型的步长, 我们得到动力系统*

$$\begin{cases} \dot{f}_c = \left\{ \dfrac{\varepsilon_1}{m} + \dfrac{\beta}{m} \displaystyle\sum_{i=1}^{m} \left[1 - \exp(\bar{f} - f(\boldsymbol{x}_i(t))\right] \right\} (\nabla f)^{\top} \boldsymbol{v}_1 \\ \quad + \left\{ \dfrac{\beta}{m} \left[1 - \exp\left(\bar{f} - f(\boldsymbol{x}_d)\right)\right] + \dfrac{\varepsilon_1}{m} \right\} (\nabla f)^{\top} \boldsymbol{\varepsilon}_2, \\ \dot{\boldsymbol{v}}_1 = \beta \exp\left(\bar{f} - f_c\right) \displaystyle\sum_{j=2}^{n} \frac{\mu_j}{\mu_1 - \mu_j} \boldsymbol{v}_j \boldsymbol{v}_j^{\top} \nabla f + \boldsymbol{r} + \sum_{j=2}^{n} \frac{\varepsilon_1}{\mu_1 - \mu_j} \boldsymbol{v}_j \boldsymbol{v}_j^{\top} \left(\boldsymbol{x}_d - \boldsymbol{x}_c\right) \\ \quad + \displaystyle\sum_{j=2}^{n} \frac{1}{\mu_1 - \mu_j} \boldsymbol{v}_j \boldsymbol{v}_j^{\top} \left(\Phi \boldsymbol{v}_1\right). \end{cases} \tag{4-43}$$

证明. 根据计算, 我们有

$$\begin{aligned} \dot{\boldsymbol{x}}_c &= \frac{1}{m} \sum_{i=1}^{m} \alpha_i \boldsymbol{v}_1 + \frac{1}{m} \left(\alpha_d \boldsymbol{\varepsilon}_2 + \varepsilon_1 \boldsymbol{\varepsilon}_2 + \varepsilon_1 \boldsymbol{v}_1\right) \\ &= \left\{ \frac{\varepsilon_1}{m} + \frac{\beta}{m} \sum_{i=1}^{m} \left[1 - \exp\left(\bar{f} - f(\boldsymbol{x}_i)\right)\right] \right\} \boldsymbol{v}_1 \\ &\quad + \left\{ \frac{\beta}{m} \left[1 - \exp\left(\bar{f} - f(\boldsymbol{x}_d)\right)\right] + \frac{\varepsilon_1}{m} \right\} \boldsymbol{\varepsilon}_2. \end{aligned}$$

我们在 $\dot{f}(\boldsymbol{x}_c) = \left(\nabla f(\boldsymbol{x}_c)\right)^{\top} \dot{\boldsymbol{x}}_c$ 中代入上述内容, 然后我们得到 (4-43) 的第一个方程. 此外, 我们有

$$\nabla \alpha(\boldsymbol{x}_c) = \frac{d\alpha}{df} \nabla f(\boldsymbol{x}_c) = \beta \exp\left(\bar{f} - f(\boldsymbol{x}_c)\right) \nabla f(\boldsymbol{x}_c).$$

因此, 我们通过在 (4-41) 中按上述替换 $\nabla \alpha(\boldsymbol{x}_c)$ 可得到 (4-43) 的第二个方程. □

上述内容分析了我们算法对应的优化动力系统. 接下来我们来分析 SUSD-TR 方向的稳定性.

我们首先给出控制和动力系统理论中稳定、渐近稳定和输入至状态稳定 (input-to-state stable, ISS) 的基本定义 [209], 这些将用于我们的稳定性分析.

定义 4.18 (稳定)**.** *若对于任意 $\bar{\varepsilon} > 0$, 存在 $\bar{\delta}(t, \bar{\varepsilon})$ 使得在 $|\eta(t_0)| < \bar{\delta}$ 时对于所有 $t > t_0$ 有 $|\eta(t)| < \bar{\varepsilon}$ 成立, 则称 η 稳定.*

定义 4.19 (渐近稳定)**.** *若在存在 $\bar{\delta}(t_0)$ 使得 $|\eta(t_0)| < \bar{\delta}$ 的情况下 $\lim\limits_{t \to \infty} \eta(t) = 0$, 则称 η 渐近稳定.*

定义 4.20 ($\mathcal{K}$ 类函数)**.** *对于定义在 $r \in [0, a)$ 上的标量连续函数 $g_1(r)$, 若它严格递增并且 $g_1(0) = 0$, 则称其属于 $\mathcal{K}$ 类.*

定义 4.21 ($\mathcal{KL}$ 类函数)**.** 对于定义在 $r \in [0,a)$ 和 $s \in [0,\infty)$ 上的标量连续函数 $g_2(r,s)$, 若对于每个固定的 s, 映射 $g_2(r,s)$ 关于 r 属于 $\mathcal{K}$ 类, 对于每个固定的 r, 映射 $g_2(r,s)$ 关于 s 是递减的, 并且当 $s \to \infty$ 时 $g_2(r,s) \to 0$, 则称其属于 $\mathcal{KL}$ 类.

定义 4.22 (输入至状态稳定)**.** 若存在 $\mathcal{KL}$ 类函数 f_1 和 $\mathcal{K}$ 类函数 f_2, 使得对于任意初始状态 $\eta(t_0) \in [0,2)$ 和任意满足 $|\delta(t)| \leq U$ 的有界输入 $\delta(t)$, 系统 $\dot{\eta} = \psi(t,\eta,\delta)$ 的解 $\eta(t)$ 对所有 $t > t_0$ 有定义, 并且满足不等式

$$|\eta(t)| \leq f_1\left(\left|\eta(t_0)\right|, t - t_0\right) + f_2 \sup_{t_0 \leq \tau \leq t} |\delta(\tau)|,$$

则称该系统关于平衡点 $\eta^* = 0$、邻域 $\eta \in [0,2)$ 和输入上界 U 是输入至状态稳定的.

下面的定理将用于建立与我们算法相关的输入至状态稳定性质.

定理 4.23 (Khalil 的著作 [209] 中的定理 4.19)**.** 设 $\mathcal{V}(t,\eta) : [0,\infty) \times [0,2) \to \mathfrak{R}$ 为一个连续可微函数. 设 $\alpha_1(\eta), \alpha_2(\eta)$ 为 $[0,2)$ 上的 $\mathcal{K}$ 类函数, $\rho(|\delta|)$ 为 $[0,U]$ 上的 $\mathcal{K}$ 类函数, 且 $\alpha_3(\eta)$ 为 $[0,2)$ 上的连续正函数. 假设对于所有 $(t,\eta,\delta) \in [0,\infty) \times [0,2) \times [-U,U]$, 函数 $\mathcal{V}$ 满足

$$\alpha_1(|\eta|) \leq \mathcal{V}(t,\eta) \leq \alpha_2(|\eta|)$$

且当 $|\eta| \geq \rho(|\delta|) > 0$ 时, $\mathcal{V}$ 满足

$$\frac{\partial \mathcal{V}}{\partial t} + \frac{\partial \mathcal{V}}{\partial \eta}\psi(t,\eta,\delta) \leq -\alpha_3(\eta).$$

则系统 $\dot{\eta} = \psi(t,\eta,\delta)$ 是输入至状态稳定的.

注意, 简明起见, 以上定义和定理中的符号与以下内容独立. 假设 $\boldsymbol{g} = \frac{\nabla f}{\|\nabla f\|_2}$. 我们现在来说明 SUSD-TR 方向 $\boldsymbol{v}_1$ 在一定情况下会趋于 $-\boldsymbol{g}$. 定义 $\eta = 1 + \boldsymbol{v}_1^{\top}\boldsymbol{g}$, 其中 $\eta = 0$ 当且仅当 $\boldsymbol{v}_1 = -\boldsymbol{g}$. 我们可以得到关于 η 的以下推论.

推论 4.24. *根据 (4-33) 和指数类型的步长, 我们得到搜索迭代的动力系统*

$$\dot{\eta} = \beta \exp\left(\bar{f} - f_c\right) \|\nabla f\|_2 \sum_{j=2}^{n} \frac{\mu_j}{\mu_1 - \mu_j}\left(\boldsymbol{g}^{\top}\boldsymbol{v}_j\right)^2 + \delta := \psi(t,\eta,\delta), \tag{4-44}$$

其中

$$\delta = \boldsymbol{r}^{\top}\boldsymbol{g} + \boldsymbol{g}^{\top}\left(\sum_{j=2}^{n} \frac{\boldsymbol{v}_j\boldsymbol{v}_j^{\top}}{\mu_1 - \mu_j}\right)\left[\varepsilon_1\left(\boldsymbol{x}_d - \boldsymbol{x}_c\right) + \Phi\boldsymbol{v}_1\right] + \boldsymbol{v}_1^{\top}\dot{\boldsymbol{g}}.$$

证明. 通过计算 $\dot{\boldsymbol{v}}_1^{\top}\boldsymbol{g}$, 可以直接由 (4-43) 得到 (4-44). □

参数 δ 是函数的非线性性 (无法由搜索点控制) 对 η 造成的外部扰动, 这其中包含了高阶项. 注意, 在 $\varepsilon_1 = 0$ 且 $\boldsymbol{\varepsilon}_2 = \mathbf{0}$ 的情况下, $\delta = \boldsymbol{r}^\top \boldsymbol{g} + \boldsymbol{v}_1^\top \dot{\boldsymbol{g}}$, 此时恰好对应 SUSD 算法, 即没有 SUSD-TR 算法中信赖域迭代这一步、迭代仅沿着 SUSD 方向推进的方法.

下面的结果给出了更多关于 $\boldsymbol{v}_1$ 何时以及如何趋于 $-\boldsymbol{g}$ 的细节.

定理 4.25. *假设 $\|\nabla f(\boldsymbol{x}_c)\|_2 > \xi$, 其中 ξ 是一个正常数. 那么, 对于 (4-44), 系统 $\dot{\eta} = \psi(t, \eta, 0)$ 关于平衡点 $\eta = 0$ 是渐近稳定的, 其中在 $\eta(0) \in [0, 2)$ 时, 随着 $t \to \infty$ 有 $\eta(t) \to 0$, 这里的 t 指连续迭代 (时间). 另外, 对于满足 $|\delta| < \beta \exp(\bar{f} - f_c) M \frac{\mu_1}{\mu_n - \mu_1} \xi$, $M \in (0, 1)$ 的扰动, 系统 $\psi(t, \eta, \delta)$ 关于平衡点 $\eta = 0$ 是局部输入至状态稳定的.*

证明. 证明与 Al-Abri 等的工作 [117] 中定理 1 的证明相同. □

4.2.4 试探步和结构步

在本节中, 我们将介绍信赖域步中包含的试探步和结构步的更多细节, 这些步骤是算法 12 实现中的一部分.

试探步可以被理解为是 SUSD-TR 算法的一个小规模修正. 在这一步, 算法通过求解信赖域内的插值模型子问题来获得新点, 即求解

$$\begin{aligned} &\min_{\boldsymbol{x}} L_k(\boldsymbol{x}) \text{ 或 } Q_k(\boldsymbol{x}) \\ &\text{s. t. } \left\| \boldsymbol{x} - \boldsymbol{x}_c^{(k)} \right\|_2 \leq \Delta_k, \end{aligned}$$

并替换掉当前步迭代点中函数值最大的一个点.

在理论分析中, 方向 $\boldsymbol{v}_1$ 存在不收敛或不稳定的可能性. 在这种情况下, 我们称 $\boldsymbol{v}_1$ 是失败的. 下面的命题介绍了在失败的情况下, 即 $\boldsymbol{v}_1$ 转向梯度上升方向 $\boldsymbol{g}$ 时, 我们在 SUSD-TR 算法中加入试探步的优势.

命题 4.26. *假设在给定迭代的 m 个搜索点处有动力系统*

$$\begin{cases} \dot{\boldsymbol{x}}_i = \boldsymbol{g}, \ i = 1, \cdots, d-1, d+1, \cdots, m, \\ \dot{\boldsymbol{x}}_d = -\bar{\alpha}_d \boldsymbol{g}, \end{cases} \tag{4-45}$$

且 $\bar{\alpha}_d > m - 1$. 则中心点将朝向梯度下降方向移动.

证明. 根据 (4-45), 我们得到 $\boldsymbol{x}_c$ 的动力系统:

$$\dot{\boldsymbol{x}}_c = \frac{m - 1 - \bar{\alpha}_d}{m} \boldsymbol{g},$$

进而我们可以直接得出结论. □

上述内容显示, 试探步可以将搜索点群中心的搜索方向从梯度上升方向拉向梯度下降方向.

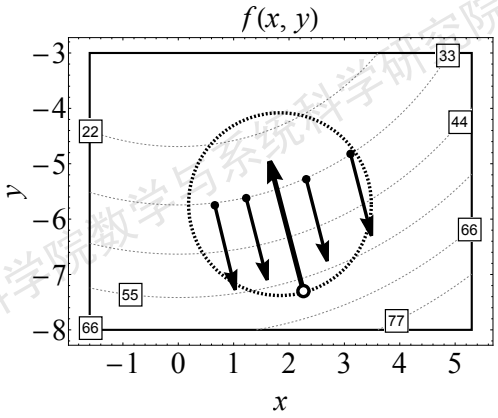


图 4-8 引导至梯度下降方向的 “逆行” 点

Figure 4-8 The “antidromic” point leading a gradient descent direction

算法的当前实现版本会适时地在使用模型前检查和改进插值集的适定性. 这是为了通过调用模型改进步来考虑迭代/插值/搜索点的位置或分布的适定性, 以获得一个良好的插值模型函数. 可以参考 Conn、Scheinberg 和 Vicente 的著作 [95] 了解更一般的讨论.

除了我们上面讨论的试探步之外, 我们还设计了结构改进步. 其动机是扩大局部输入至状态稳定的吸引区域 [209], 使搜索方向更易稳定地趋于梯度下降方向. 事实上, 当我们给出定理 4.25 时, 有一个假设是函数在中心点的梯度的范数应该大于 ξ. 可以注意到, 在定理 4.25 中, 有 ξ 的下界.

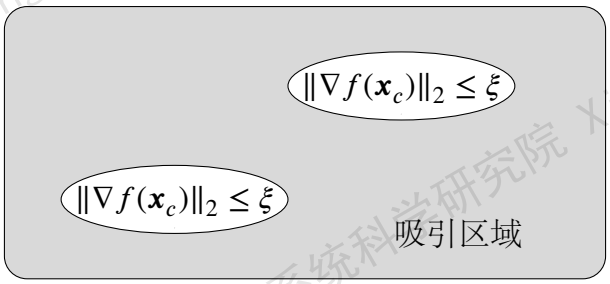


图 4-9 吸引区域 ($\xi > \frac{|\delta|(\mu_n-\mu_1)}{\beta\exp(\bar{f}-f_c)M\mu_1}$)

Figure 4-9 Attraction region ($\xi > \frac{|\delta|(\mu_n-\mu_1)}{\beta\exp(\bar{f}-f_c)M\mu_1}$)

注 4.5. 图 4-9 使用阴影区域表示定理 4.25 ($\mu_1 \neq 0$) 中局部输入至状态稳定平衡的吸引区域.

可以看出, 当矩阵 $\boldsymbol{C}$ 的条件数较小时, 这种下界会减小. 在算法实现中, 我们试图通过减小最大特征值来启发式地使特征值彼此接近. 这个步骤是一个具有良好数值结果的启发式步骤, 我们并不期望通过这一步严格保证协方差矩阵的良好条件. 对于这里的协方差矩阵 $\boldsymbol{C}$, 放射状分布的搜索点可以使 μ_1 和 μ_n 彼此接近, 这通常发生在插值点集适定的情况下, 这些处理在模型改进步中被考虑了. 此外,

假设 $\boldsymbol{v} \in \mathfrak{R}^n$ 是 $\boldsymbol{C}$ 的一个非零特征向量, 对应于最大特征值 μ_n, 我们得到

$$\begin{aligned}\mu_n &= \max_{\|\boldsymbol{v}\|_2=1} \boldsymbol{v}^\top \boldsymbol{C}\boldsymbol{v} \\ &= \max_{\|\boldsymbol{v}\|_2=1} \boldsymbol{v}^\top \left[\sum_{i=1}^{m} \left(\boldsymbol{x}_i - \boldsymbol{x}_c\right)\left(\boldsymbol{x}_i - \boldsymbol{x}_c\right)^\top\right] \boldsymbol{v} \\ &= \max_{\|\boldsymbol{v}\|_2=1} \sum_{i=1}^{m} \left[\left(\boldsymbol{x}_i - \boldsymbol{x}_c\right)^\top \boldsymbol{v}\right]^2.\end{aligned}$$

然后我们可以得到其上界和下界, 即

$$\sum_{i=1}^{m} \left\|\boldsymbol{x}_i - \boldsymbol{x}_c\right\|_2^2 \geq \max_{\|\boldsymbol{v}\|_2=1} \sum_{i=1}^{m} \left[\left(\boldsymbol{x}_i - \boldsymbol{x}_c\right)^\top \boldsymbol{v}\right]^2 \geq \max_i \left\|\boldsymbol{x}_i - \boldsymbol{x}_c\right\|_2^2.$$

同时, 算法 13 可以降低协方差矩阵 $\boldsymbol{C}$ 最大特征值的上界和下界. 这里我们省略了迭代指标 k.

算法 13 结构步

1: 在搜索点集中舍弃最远点 $\boldsymbol{x}_{\text{far}} := \arg\max_{\boldsymbol{x}_i} \left\|\boldsymbol{x}_i - \boldsymbol{x}_c\right\|_2^2$.

2: 在搜索点集中添加新点 $\boldsymbol{x}_{\text{new}} = \frac{1}{m-1} \sum_{i \neq \text{far}} \boldsymbol{x}_i$ (替换 $\boldsymbol{x}_{\text{far}}$).

上文讨论了 SUSD-TR 算法中的试探步和结构步.

4.2.5 数值结果

本节展示数值结果, 主要包括求解两个测试问题的结果, 以及通过观察求解测试问题集的相关性能表现来与其他无导数优化算法进行对比的情况.

例 4.1. 我们实现了对应方法的 MATLAB 代码来极小化 2 维 Rosenbrock 函数

$$f(x_1, x_2) = \left(1 - x_1\right)^2 + 100\left(x_2 - x_1^2\right)^2, \tag{4-46}$$

及函数

$$f(x_1, x_2) = \left(1 - x_1\right)^2 + \left(x_2 - x_1^2\right)^2, \tag{4-47}$$

以此来测试和对比 SUSD 算法 [117] (仅执行 SUSD-TR 算法中的线搜索步) 和我们的 SUSD-TR 算法, 其中 $(x_1, x_2)^\top$ 表示变量 $\boldsymbol{x} \in \mathfrak{R}^2$. 图 4-10 展示了使用 SUSD 算法和 SUSD-TR 算法求解上述两个问题的迭代轨迹. 空心圆点代表迭代过程中的中心点. 我们观察到 SUSD 算法的迭代方向不稳定, $\boldsymbol{v}_1$ 不收敛, 最终错过了极小点 $(1, 1)^\top$. 然而, 如果我们应用 SUSD-TR 算法, 可以发现它能够成功收敛到极小点. 在求解这两个例子时, 我们设置算法参数为 $\kappa = 1.2, \Delta_k = 5, \beta = 1$, 5 个初始搜索点是

$$\boldsymbol{x}_1^{(0)} = \begin{pmatrix} 20 \\ 0 \end{pmatrix}, \boldsymbol{x}_2^{(0)} = \begin{pmatrix} 23 \\ 4 \end{pmatrix}, \boldsymbol{x}_3^{(0)} = \begin{pmatrix} 23 \\ -4 \end{pmatrix}, \boldsymbol{x}_4^{(0)} = \begin{pmatrix} 17 \\ 4 \end{pmatrix}, \boldsymbol{x}_5^{(0)} = \begin{pmatrix} 17 \\ -4 \end{pmatrix},$$

SUSD-TR 算法中的模型函数是欠定二次插值模型. 总之, 在求解这里的例子时, 对于 SUSD 算法, 迭代有时不收敛, 搜索点有时沿梯度上升方向前进. 然而, 我们的 SUSD-TR 算法可以有效地收敛到极小点, 搜索点基本上沿着函数值下降方向行进, 并仅需不到 SUSD 算法所需的一半的函数值探测次数. 原因之一即是在当前迭代中总是存在一个点沿着二次插值模型的梯度下降方向行进, 当模型准确时 (模型至少是完全线性的), 这也与目标函数的梯度下降方向靠近.

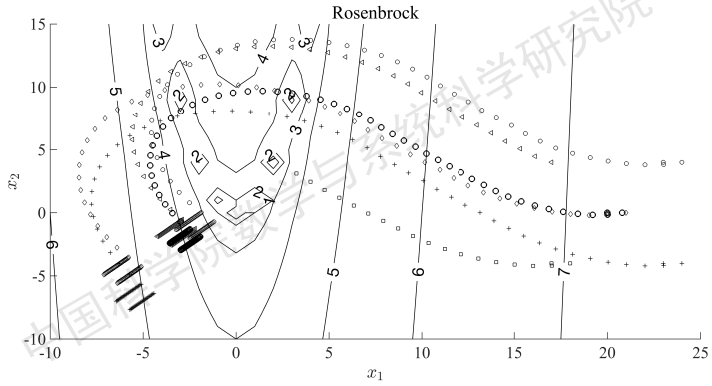
表 4-4 图 4-11 和图 4-12 对应的测试问题

Table 4-4 Test problems for Figure 4-11 and Figure 4-12

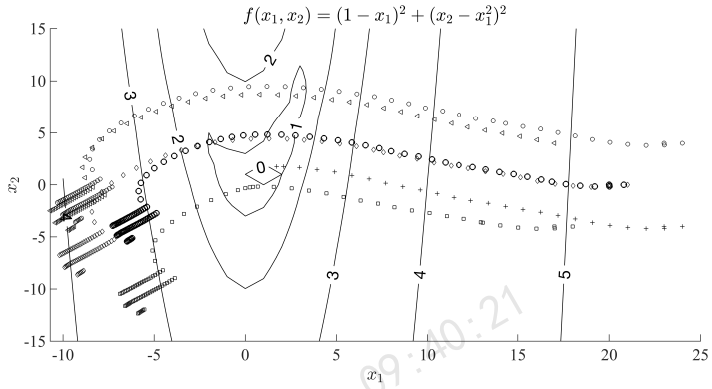
ARGLINA	ARGLINA4	ARGLINB	ARGLINC	ARGTRIG
ARWHEAD	BDQRTIC	BDQRTICP	BDVALUE	BROWNAL
BROYDN3D	BROYDN7D	BRYBND	CHAINWOO	CHEBQUAD
CHNROSNBZ	CHPOWELLB	CHPOWELLS	CHROSEN	COSINE
CUBE	CURLY10	CURLY20	CURLY30	DIXMAANE
DIXMAANF	DIXMAANG	DIXMAANH	DIXMAANI	DIXMAANJ
DIXMAANK	DIXMAANL	DIXMAANM	DIXMAANN	DIXMAANO
DIXMAANP	DQRTIC	EDENSCH	ENGVAL1	ERRINROS
EXPSUM	EXTROSNB	EXTTET	FIROSE	FLETCBV2
FLETCBV3	FLETCHCR	FMINSRF2	FREUROTH	GENBROWN
GENHUMPS	GENROSE	INDEF	INTEGREQ	LIARWHD
LILIFUN3	LILIFUN4	MOREBV	MOREBVL	NCB20
NCB20B	NONCVXU2	NONCVXUN	NONDIA	NONDQUAR
PENALTY1	PENALTY2	PENALTY3	PENALTY3P	POWELLSG
POWER	ROSENBROCK	SBRYBND	SBRYBNDL	SCHMVETT
SCOSINE	SCOSINEL	SEROSE	SINQUAD	SPARSINE
SPARSQUR	SPHRPTS	SPMSRTLS	SROSENBR	STMOD
TOINTGSS	TOINTTRIG	TQUARTIC	TRIGSABS	-

求解上述经典例子展示了我们算法的优势. 为了进一步比较, 我们将我们的算法与基于 SUSD 方向的无导数优化算法 [117] 及有代表性的 Nelder-Mead 方法 [51] 和 NEWUOA [94] 进行了比较. 表 4-4 中的测试问题维数从 2 到 120, 取自经典无约束优化测试函数集 [177, 178, 180, 181, 185, 186], 相应的数值结果如图 4-11 和图 4-12 所示.

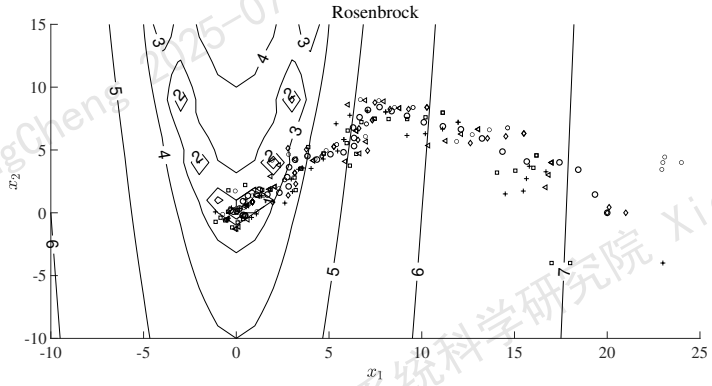
我们的算法从 $m = 2n + 1$ 个随机选择的初始点开始, 参数分别设定为 $\beta = 1, P = 5, \varepsilon = 10^{-6}, \kappa = 1.2$ 和精度 τ 为 $10^{-1}, 10^{-3}, 10^{-5}$. 我们从图 4-11 和图 4-12 可以观察到, 对于所测试的问题, SUSD-TR 算法在一定精度下比 Nelder-Mead 方法更有效, 可以接近 NEWUOA 算法, 实现对这些测试问题的高效求解, 其数值表



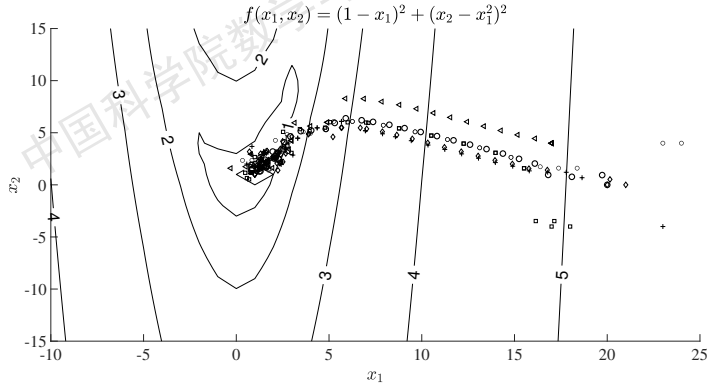
(a) 使用 SUSD 算法极小化 (4-46)



(b) 使用 SUSD 算法极小化 (4-47)



(c) 使用 SUSD-TR 算法极小化 (4-46)



(d) 使用 SUSD-TR 算法极小化 (4-47)

图 4-10 使用 SUSD 和 SUSD-TR 求解 2 维测试问题

Figure 4-10 Solving the 2-dimensional test problems by SUSD and SUSD-TR

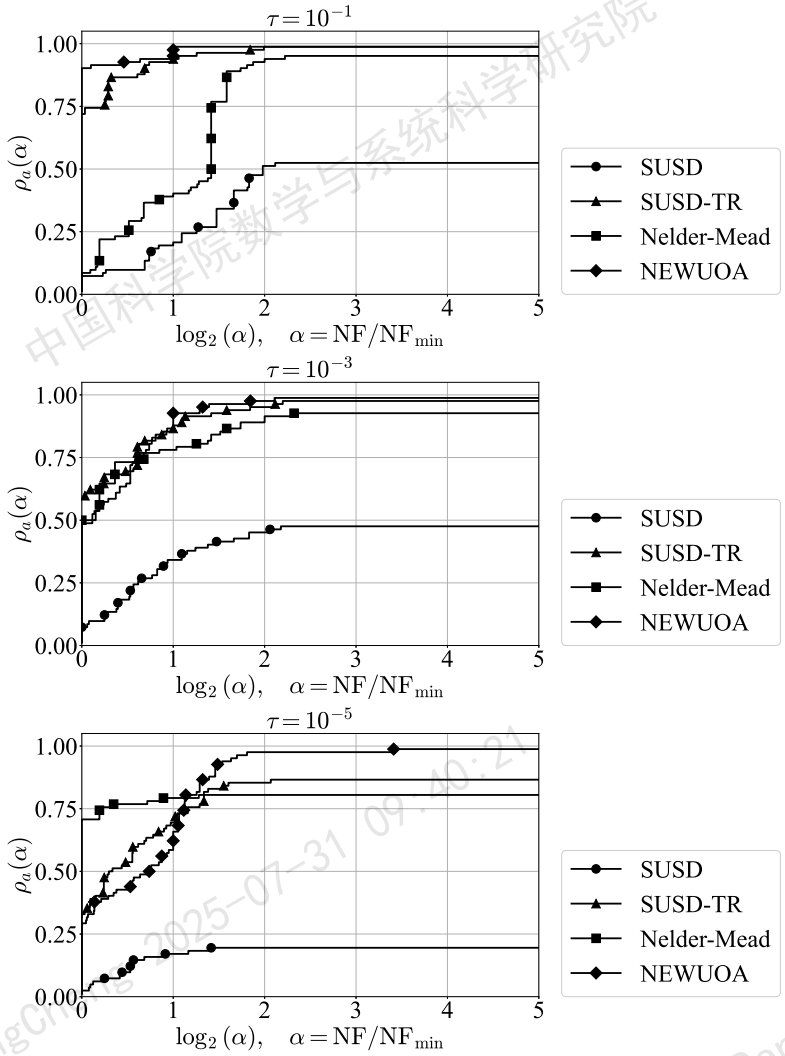


图 4-11 求解测试问题的 Performance Profile

Figure 4-11 Performance Profile for solving the test problems

现明显优于基于 SUSD 方向的方法 (即纯线搜索类型), 其他算法均选取同规模默认参数.

此外, SUSD-TR 算法和 SUSD 算法可以并行化地处理探测时间成本高的问题, 这是相比于其他算法的一个主要优势 (它可以在线搜索步同时探测 m 个点).

基于上述数值测试结果, 可以看到与仅使用 SUSD 方向相比, 我们的算法通过结合 SUSD 方向和信赖域技术, 成功提升了求解问题时的表现.

4.2.6　小结

本节提出了 SUSD-TR 算法, 它结合了求解模型函数的信赖域子问题的过程和沿 SUSD 方向传输点的过程. 我们展示了 SUSD-TR 算法的动力系统及其对应的并行搜索前进方向的稳定性. 数值结果展示了我们的算法 SUSD-TR 的优势. 在未来的研究中, 我们将考虑更有效的步长、更好的组合方式和不同的 SUSD 方向. 此外, 我们可以进而研究对多个点进行小规模修正的方法, 求解带约束问题的

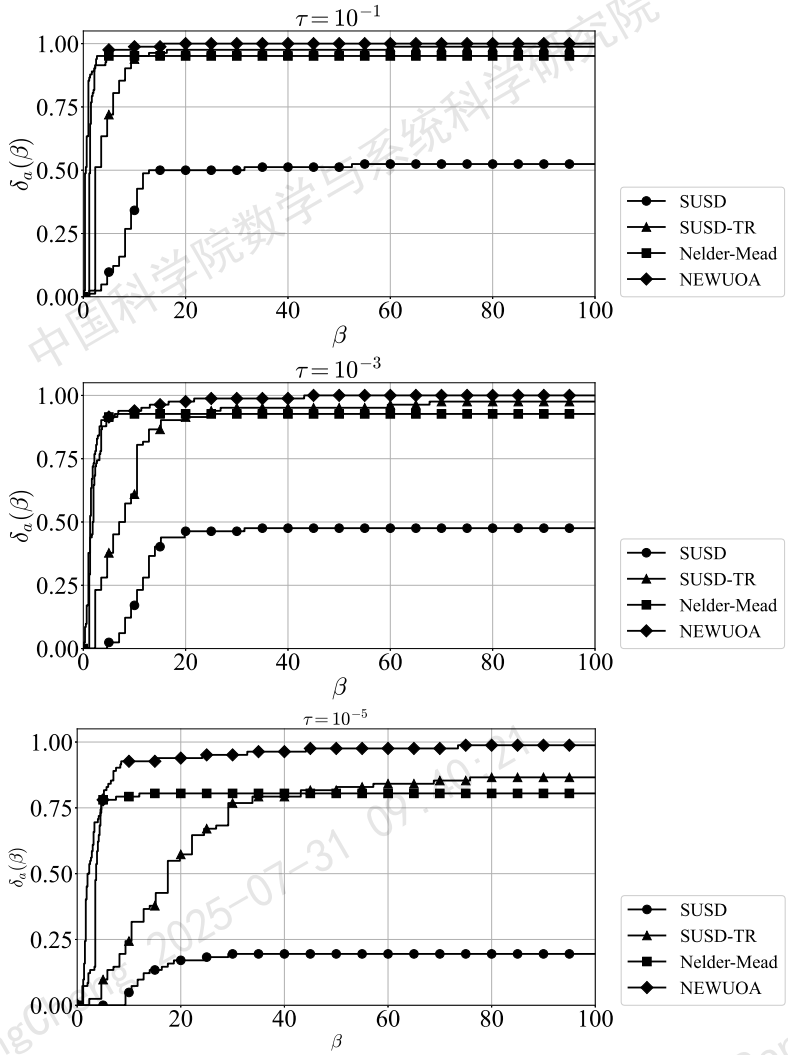


图 4-12 求解测试问题的 Data Profile

Figure 4-12 Data Profile for solving the test problems

相应方法也是未来的一个研究方向.

第 5 章　总结与展望

许多来自科学、工程以及人工智能、机器学习应用的优化问题都存在着导数不可用或不可靠的情况. 在这些问题中, 目标函数只能作为黑箱输出, 而无法提供导数信息. 面对这样的情况, 就需要用到无导数优化方法来求解. 无导数优化是计算科学和工程中最重要、最开放和最具挑战性的领域之一, 具有巨大的实际潜力. 设计无导数优化方法的目标是通过使用尽量少的函数值探测次数来实现优化. 我们知道, 信赖域方法是非线性优化中一类著名的算法. 信赖域算法通过在接近当前迭代点的区域内极小化二次模型来生成新的迭代点. 在无导数情形中, 相应的模型通常是通过多项式插值、回归或其他逼近技术构造的, 这类优化方法被称为基于模型的无导数优化方法. 本论文的第 2 章到第 4 章针对求解无约束的无导数优化问题进行了研究.

第 2 章探讨了如何设计更好的逼近模型, 并深入讨论了逼近与优化之间的关系. 我们知道, 最有效的基于模型的无导数优化方法之一是基于欠定二次插值模型的信赖域算法. 在迭代过程中, 使用不同的技术更新二次模型会产生不同的插值模型. Powell、Conn 和 Toint 等提出的最小范数 (更新) 二次模型 [171, 210] 是近 20 年在国际上领先的模型. 本论文通过以下两种方式改进了这些模型.

首先, 我们提出通过极小化新二次模型与旧模型之间的变化量的 H^2 范数来获取二次模型函数, 以此减小插值点或方程的数量下界. 这种方法具有相应的投影性质和误差界, 我们给出了一个易于实现的模型更新公式. 并且通过给出最小加权 H^2 范数更新二次模型的权重系数的重心进一步找到了最佳权重系数, 同时我们给出了理论分析和数值结果.

其次, 考虑到信赖域迭代步可以在基于模型的无导数信赖域方法中提供相应的模型函数最优性信息, 我们给出了理解和分析 Conn 和 Toint 的模型的一个新视角, 并提出了一个易于实现的新模型. 文章还介绍了使用这种模型的理论原因和动机, 据作者所知, 这是首个通过考虑信赖域迭代性质来为无导数优化方法构造欠定二次插值模型的工作.

相关的未来工作包括对逼近与优化关系的进一步研究. 本论文的工作已经揭示了逼近对于优化的重要性, 我们还可以利用本论文所探索的对最小 H^2 范数更新二次模型和函数最优性的讨论来开发更多无导数信赖域算法, 特别是给出具有更好收敛性保证的算法. 例如, 针对具有不同结构的问题设计自适应的 H^2 范数更新二次模型的权重系数. 此外, 我们还可以从其他角度比较最小加权 H^2 范数更新二次模型的权重系数. 另外, 还需要考虑更多细节和进一步工作: 例如, 我们是否可以通过仅使用 “比较函数值大小” 的探测机制 (而不是使用函数值探测机制) 来构造一个考虑了模型最优性的更好模型? 事实上, 本论文关于 “优化和逼近” 的研究也在试图推动这些重要学科的融合. 值得注意的是, 我们的最新发现表明无导数优化正是逼近和优化的交叉点. “用于逼近的优化” 和 “用于优化的逼

近” 这两条探索路径在数学理论和实际算法应用方面有着巨大的潜力. 除此之外, 寻找每步迭代中更好的插值点数量是另一个将来可以尝试研究的工作.

第 3 章讨论了带变换目标函数无导数优化. 我们提出了一种求解带变换目标函数优化问题的无导数优化方法, 并给出了相应的探测方案. 对于严格凸模型以及在信赖域内具有唯一极小点的模型函数, 我们证明了除了平移变换之外的保模型最优性变换的存在性、给出了与保模型最优性变换对应的变换函数值的一个充分必要条件. 我们获得了仿射变换目标函数的相应二次模型函数, 并证明了一些正单调变换 (甚至是具有正乘法系数的仿射变换) 不是保模型最优性变换. 我们还对相应的模型函数进行了插值误差分析, 给出了仿射变换目标函数的情况, 在理论上给出了一阶临界点的收敛性分析. 并给出了测试问题和实际问题的数值结果. 如第 3 章所述, 带变换的无导数优化还有很多待研究的内容. 例如, 我们可以尝试将构造最小 Frobenius 范数更新二次模型的方法应用于实际的带变换的无导数优化中 (例如, 带有噪声添加机制或隐私保护机制的黑箱优化). 此外, 在更弱的假设下求解带有变换目标函数的问题时的算法的收敛性分析仍然是一个有待解决的、具有挑战性的问题. 本论文所提出的有关 (不使用导数) 极小化 “移动靶” 目标函数的开放问题也是有趣且富有挑战的. 我们认为, 带变换的无导数优化已经开始在理论和应用领域显示出了潜在的影响. 事实上, 它从新的角度刻画了带有噪声的优化问题和情形, 并推导出了新的理论结果. 因此, 我们认为对这类问题开展进一步的理论分析和算法设计是有价值的. 同时, 带变换的无导数优化中的带噪声黑箱实际问题和机器学习问题也密切相关, 故而很多人工智能领域的相关应用值得关注.

第 4 章分为两个部分. 第一部分讨论了求解大规模无导数优化问题的子空间方法, 第二部分探讨了将线搜索框架和信赖域框架相结合的并行无导数优化算法. 在当前的无导数优化问题中, 大规模问题仍然是一个瓶颈, 这是因为当问题的变量维数很高时, 构造局部 (多项式) 模型的计算成本和插值误差可能会很高. 我们将这种情况视为无导数优化中的维数灾难. 为了应对这一挑战, 我们提出了一种新的求解大规模黑箱问题的子空间优化方法, 该方法利用子空间技术和二次插值模型来高效地搜索极小点. 我们的新方法 2D-MoSub 通过迭代地在 2 维子空间中使用 2 维二次模型来寻找新点、进行更新, 其具有良好的逼近误差和收敛性质. 接下来可以开展的相关工作包括 (使用或不使用导数地) 求解更大规模的问题. 此外, 我们也将探索更多技术, 包括子空间方法、并行化、随机化技术等, 以用来处理大规模问题. 我们还计划设计新的子空间选择策略, 并研究大规模带约束问题. 在实际应用中, 求解大规模问题对于各种应用非常关键, 包括机器学习和其他实际需求. 我们发现 2D-MoSub 能够为求解对优化和数值领域非常重要的大规模问题带来希望. 此外, 我们还将考虑利用高性能计算进行大规模分析和优化, 进一步探索相应的随机子空间方法.

在第 4 章的第二部分中, 我们介绍了一种使用二次模型改进线搜索方法的并行方法 SUSD-TR. 该方法将由插值点的协方差矩阵导出的 SUSD 方向与在当前

迭代步骤中二次插值模型对应的信赖域子问题的解相结合. 我们对 SUSD-TR 算法优化过程的相关动力系统和迭代搜索方向的性质进行了分析, 数值结果展示了该算法的高效性. 下一步的工作包括对并行化方法的进一步研究. 我们还将考虑改进步长、改进两种方法的结合方式以及对不同 SUSD 方向的进一步探索. 此外, 我们还计划在多个搜索点上同步进行小规模修正, 并将相应的方法应用到带约束优化问题上. 事实上, SUSD-TR 可以被视为线搜索框架和信赖域框架的组合方法的并行和无导数版本. 它试图利用和结合这两种传统迭代优化框架的优势. 下一步, 我们还将继续研究基于模型的无导数方法和直接搜索类方法的比较和组合, 并探索更多的特定应用, 如分布式源搜索场景下的应用等.

中国科学院数学与系统科学研究院 XiePengCheng 2025-07-31 09:40:21

中国科学院数学与系统科学研究院 XiePengCheng 2025-07-31 09:40:21

参考文献

[1] 袁亚湘, 孙文瑜. 最优化理论与方法 [M]. 科学出版社, 1997.

[2] 袁亚湘. 非线性规划数值方法 [M]. 上海科学技术出版社, 1993.

[3] Nocedal J, Wright S J. Numerical Optimization [M]. Berlin: Springer, 2006.

[4] Bertsekas D P. Nonlinear Programming [M]. Belmont, MA, USA: Athena Scientific, 1999.

[5] 戴彧虹, 刘新为. 线性与非线性规划算法与理论 [J]. 运筹学学报, 2014, 18(1): 69-92.

[6] 郭田德, 韩丛英, 唐思琦. 组合优化机器学习方法 [M]. 科学出版社, 2019.

[7] 刘浩洋, 户将, 李勇锋, 等. 最优化：建模、算法与理论 [M]. 高等教育出版社, 2020.

[8] 刘歆, 刘亚锋. 凸优化 [M]. 科学出版社, 2024.

[9] Ferris M C, Pang J S. Engineering and economic applications of complementarity problems [J]. SIAM Review, 1997, 39(4): 669-713.

[10] Biswas P, Lian T C, Wang T C, et al. Semidefinite programming based algorithms for sensor network localization [J]. ACM Transactions on Sensor Networks, 2006, 2(2): 188-220.

[11] Liu X, Wang X, Wen Z, et al. On the convergence of the self-consistent field iteration in kohn–sham density functional theory [J]. SIAM Journal on Matrix Analysis and Applications, 2014, 35(2): 546-558.

[12] Bottou L, Curtis F E, Nocedal J. Optimization methods for large-scale machine learning [J]. SIAM review, 2018, 60(2): 223-311.

[13] Li Z, Zhang S, Wang Y, et al. Alignment of molecular networks by integer quadratic programming [J]. Bioinformatics, 2007, 23(13): 1631-1639.

[14] Wen Z, Yin W. A feasible method for optimization with orthogonality constraints [J]. Mathematical Programming, 2013, 142(1): 397-434.

[15] 刘歆. 强关联多电子体系的优化模型与算法 [J]. 计算数学, 2023, 45(2): 141-159.

[16] Li T, Xie J, Lu S, et al. Duopoly game of callable products in airline revenue management [J]. European Journal of Operational Research, 2016, 254(3): 925-934.

[17] Yin J, Li Q. A semismooth Newton method for support vector classification and regression [J]. Computational Optimization and Applications, 2019, 73(2): 477-508.

[18] Xia Y, Yuan Y X. A new linearization method for quadratic assignment problems [J]. Optimisation Methods and Software, 2006, 21(5): 805-818.

[19] Xu D, Du D. The k-level facility location game [J]. Operations Research Letters, 2006, 34 (4): 421-426.

[20] Conn A R, Scheinberg K, Vicente L N. Introduction to Derivative-free Optimization [M]. Philadelphia: SIAM, 2009.

[21] Audet C, Hare W. Derivative-free and Blackbox Optimization [M]. Heidelberg: Springer, 2017.

[22] Audet C, Orban D. Finding optimal algorithmic parameters using derivative-free optimization [J]. SIAM Journal on Optimization, 2006, 17(3): 642-664.

[23] Aly A, Guadagni G, Dugan J B. Derivative-free optimization of neural networks using local search [C]//2019 IEEE 10th Annual Ubiquitous Computing, Electronics Mobile Communication Conference (UEMCON). Piscataway: IEEE, 2019: 293-299.

[24] Higham N J. Accuracy and Stability of Numerical Algorithms [M]. Philadelphia: SIAM, 2002.

[25] Levina T, Levin Y, McGill J, et al. Dynamic pricing with online learning and strategic consumers: An application of the aggregating algorithm [J]. Operations Research, 2009, 57(2): 327-341.

[26] Li S, Xie P, Zhou Z, et al. Simulation of interaction of folded waveguide space traveling wave tubes with derivative-free mixedinteger based NEWUOA algorithm [C]//2021 7th International Conference on Computer and Communications. Piscataway: IEEE, 2021: 1215-1219.

[27] Booker A, Frank P, Dennis J E, et al. Managing surrogate objectives to optimize a helicopter rotor design-further experiments [C]//7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization. 1998: 4717.

[28] Booker A J, Dennis J E, Frank P D, et al. Optimization using surrogate objectives on a helicopter test example [M]//Computational Methods for Optimal Design and Control. Boston: Springer, 1998: 49-58.

[29] Serafini D B. A Framework for Managing Models in Nonlinear Optimization of Computationally Expensive Functions [M]. Houston: Rice University, 1999.

[30] Audet C, Dennis J E. A pattern search filter method for nonlinear programming without derivatives [J]. SIAM Journal on Optimization, 2004, 14(4): 980-1010.

[31] Marsden A L. Aerodynamic Noise Control by Optimal Shape Design [M]. Stanford: Stanford University, 2005.

[32] Marsden A L, Wang M, Dennis J E, et al. Optimal aeroacoustic shape design using the surrogate management framework [J]. Optimization and Engineering, 2004, 5(2): 235-262.

[33] Duvigneau R, Visonneau M. Hydrodynamic design using a derivative-free method [J]. Structural and Multidisciplinary Optimization, 2004, 28: 195-205.

[34] Green J E J. Faster and more accurate testing [J]. Advanced Materials and Processes, 1987, 131: 72, 75-76, 79.

[35] Gu T, Li W, Zhao A, et al. BBGP-sDFO: Batch Bayesian and Gaussian process enhanced subspace derivative free optimization for high-dimensional analog circuit synthesis [J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2024, 43(2): 417-430.

[36] Martins J R, Lambe A B. Multidisciplinary design optimization: A survey of architectures [J]. AIAA Journal, 2013, 51(9): 2049-2075.

[37] Ragonneau T M. Model-based derivative-free optimization methods and software [D]. Hong Kong Polytechnic University, 2023.

[38] Ghanbari H, Scheinberg K. Black-box optimization in machine learning with trust region based derivative free algorithm [R]. 2017.

[39] Wilson Z T, Sahinidis N V. The ALAMO approach to machine learning [J]. Computers & Chemical Engineering, 2017, 106: 785-795.

[40] Ughi G, Abrol V, Tanner J. An empirical study of derivative-free-optimization algorithms for targeted black-box attacks in deep neural networks [J]. Optimization and Engineering, 2022, 23(3): 1319-1346.

[41] Tett S F B, Gregory J M, Freychet N, et al. Does model calibration reduce uncertainty in climate projections? [J]. Journal of Climate, 2022, 35(8): 2585 - 2602.

[42] 丁晓东. 基于插值模型的无导数优化方法及其应用 [D]. 北京: 中国科学院数学与系统科学研究院, 2010.

[43] Meza J C, Martinez M L. Direct search methods for the molecular conformation problem [J]. Journal of Computational Chemistry, 1994, 15(6): 627-632.

[44] Alberto P, Nogueira F, Rocha H, et al. Pattern search methods for user-provided points: Application to molecular geometry problems [J]. SIAM Journal on Optimization, 2004, 14 (4): 1216-1236.
[45] Davis P. Looking beyond the black box in optimization [J]. SIAM News, 2016, 49(8).
[46] 张在坤. 无导数优化方法的研究 [D]. 北京: 中国科学院数学与系统科学研究院, 2012.
[47] Wright M H. Direct search methods: Once scorned, now respectable [J]. Pitman Research Notes in Mathematics Series, 1996: 191-208.
[48] Lewis R M, Torczon V J, Trosset M W. Direct search methods: Then and now [J]. Journal of Computational and Applied Mathematics, 2000, 124(1-2): 191-207.
[49] Kolda T, Lewis R, Torczon V J. Optimization by direct search: New perspectives on some classical and modern methods [J]. SIAM Review, 2003, 45: 385-482.
[50] Hooke R, Jeeves T A. "Direct search" solution of numerical and statistical problems [J]. Journal of the ACM, 1961, 8(2): 212-229.
[51] Nelder J A, Mead R. A simplex method for function minimization [J]. The Computer Journal, 1965, 7(4): 308-313.
[52] Tseng P. Fortified-descent simplicial search method: A general approach [J]. SIAM Journal on Optimization, 1999, 10: 269-288.
[53] Fermi E, Metropolis N. Numerical solution of a minimization problem, Los Alamos unclassified report ls-492 [R]. Los Alamos, USA: Alamos National Laboratory, 1952.
[54] Box G E P. Evolutionary operation: A method for increasing industrial productivity [J]. Applied Statistics, 1957, 6(2): 81-101.
[55] Torczon V J. Multi-directional search: A direct search algorithm for parallel machines [D]. Houston, TX, USA: Rice University, 1989.
[56] Dennis J E, Torczon V J. Direct search methods on parallel machines [J]. SIAM Journal on Optimization, 1991, 1(4): 448-474.
[57] Torczon V J. On the convergence of the multidirectional search algorithm [J]. SIAM Journal on Optimization, 1991, 1(1): 123-145.
[58] Lewis R M, Torczon V J. Rank ordering and positive bases in pattern search algorithms: ICASE Technical report TR 96-71 [R]. Hampton, USA: NASA Langley Research Center, 1996.
[59] Torczon V J. On the convergence of pattern search algorithms [J]. SIAM Journal on optimization, 1997, 7(1): 1-25.
[60] Lewis R M, Torczon V J. Pattern search algorithms for bound constrained minimization [J]. SIAM Journal on Optimization, 1999, 9(4): 1082-1099.
[61] Lewis R M, Torczon V J. Pattern search methods for linearly constrained minimization [J]. SIAM Journal on Optimization, 2000, 10(3): 917-941.
[62] Audet C, Dennis J E. Analysis of generalized pattern searches [J]. SIAM Journal on Optimization, 2000, 13.
[63] Hough P D, Kolda T G, Torczon V J. Asynchronous parallel pattern search for nonlinear optimization [J]. SIAM Journal on Scientific Computing, 2002, 23(1): 134-156.
[64] Kolda T G. Revisiting asynchronous parallel pattern search for nonlinear optimization [J]. SIAM Journal on Optimization, 2006, 16(2): 563-586.
[65] Abramson M A, Audet C, Dennis J E. Nonlinear programming by mesh adaptive direct searches [J]. SIAG/OPT Views-and-News, 2006, 17: 2-11.
[66] Audet C, Le Digabel S, Tribes C. NOMAD user guide [R]. Les Cahiers du GERAD, 2009.

[67] 邓乃扬. 计算方法丛书: 无约束最优化计算方法 [M]. 科学出版社, 1982.
[68] McKinnon K I M. Convergence of the Nelder-Mead simplex method to a nonstationary point [J]. SIAM Journal on Optimization, 1998, 9(1): 148-158.
[69] Kelley C T. Detection and remediation of stagnation in the Nelder–Mead algorithm using a sufficient decrease condition [J]. SIAM Journal on Optimization, 1999, 10(1): 43-55.
[70] Nazareth L, Tseng P. Gilding the lily: A variant of the Nelder-Mead algorithm based on golden-section search [J]. Computational Optimization and Applications, 2002, 22(1): 133-144.
[71] Price C J, Coope I D, Byatt D. A convergent variant of the Nelder-Mead algorithm [J]. Journal of Optimization Theory and Applications, 2002, 113(1): 5-19.
[72] Rosenbrock H H. An automatic method for finding the greatest or least value of a function [J]. The Computer Journal, 1960, 3(3): 175-184.
[73] Swann W H. Direct search methods [M]//Murray W. Numerical Methods for Unconstrained Optimization. London: Academic Press, 1972: 13-28.
[74] Smith C S. The automatic computation of maximum likelihood estimates [R]. Scientific Department, National Coal Board, 1962.
[75] Powell M J D. An efficient method for finding the minimum of a function of several variables without calculating derivatives [J]. The Computer Journal, 1964, 7(2): 155-162.
[76] Stewart III G W. A modification of Davidon's minimization method to accept difference approximations of derivatives [J]. Journal of the ACM (JACM), 1967, 14(1): 72-83.
[77] Gill P E, Murray W. Quasi-Newton methods for unconstrained optimization [J]. IMA Journal of Applied Mathematics, 1972, 9(1): 91-108.
[78] Gill P E, Murray W, Saunders M A, et al. Computing forward-difference intervals for numerical optimization [J]. SIAM Journal on Scientific and Statistical Computing, 1983, 4(2): 310-321.
[79] Nesterov Y, Spokoiny V. Random gradient-free minimization of convex functions [J]. Foundations of Computational Mathematics, 2017, 17(2): 527-566.
[80] Duchi J C, Jordan M I, Wainwright M J, et al. Optimal rates for zero-order convex optimization: The power of two function evaluations [J]. IEEE Transactions on Information Theory, 2015, 61(5): 2788-2806.
[81] Scheinberg K. Finite difference gradient approximation: To randomize or not? [J]. INFORMS Journal on Computing, 2022, 34(5): 2384-2388.
[82] Zhigljavsky A A. Theory of Global Random Search [M]. Heidelberg: Springer Science & Business Media, 2012.
[83] Berahas A S, Cao L, Choromanski K, et al. A theoretical and empirical comparison of gradient approximations in derivative-free optimization [J]. Foundations of Computational Mathematics, 2022, 22(2): 507-560.
[84] Diniz-Ehrhardt M A, Martínez J M, Raydán M. A derivative-free nonmonotone line-search technique for unconstrained optimization [J]. Journal of Computational and Applied Mathematics, 2008, 219(2): 383-397.
[85] Zangwill W I. Minimizing a function without calculating derivatives [J]. The Computer Journal, 1967, 10(3): 293-296.
[86] Gilmore P, Kelley C T. An implicit filtering algorithm for optimization of functions with many local minima [J]. SIAM Journal on Optimization, 1995, 5(2): 269-285.

[87] Kelley C T. A brief introduction to implicit filtering [R]. North Carolina State University. Center for Research in Scientific Computation, 2002.

[88] Kelley C T. Implicit filtering [M]. Philadelphia: SIAM, 2011.

[89] Greenstadt J. A quasi-Newton method with no derivatives [J]. Mathematics of Computation, 1972, 26(117): 145-166.

[90] Greenstadt J. Revision of a derivative-free quasi-Newton method [J]. Mathematics of Computation, 1978, 32(141): 201-221.

[91] Winfield D. Function and functional optimization by interpolation in data tables [D]. Harvard University, 1969.

[92] Powell M J D. On trust region methods for unconstrained minimization without derivatives [J]. Mathematical Programming, 2003, 97: 605-623.

[93] Powell M J D. Least Frobenius norm updating of quadratic models that satisfy interpolation conditions [J]. Mathematical Programming, 2004, 100: 183-215.

[94] Powell M J D. The NEWUOA software for unconstrained optimization without derivatives [M]//Large-scale Nonlinear Optimization. Boston: Springer, 2006: 255-297.

[95] Conn A, Scheinberg K, Vicente L N. Geometry of sample sets in derivative free optimization. part ii: polynomial regression and underdetermined interpolation [J]. IMA Journal of Numerical Analysis, 2008, 28: 721-748.

[96] Conn A, Scheinberg K, Vicente L N. Global convergence of general derivative-free trust-region algorithms to first- and second-order critical points [J]. SIAM Journal on Optimization, 2009, 20: 387-415.

[97] Xie P, Yuan Y. Least H^2 norm updating quadratic interpolation model function for derivative-free trust-region algorithms [R]. 2023.

[98] Xie P, Yuan Y. A new two-dimensional model-based subspace method for large-scale unconstrained derivative-free optimization: 2D-MoSub [R]. 2023.

[99] Xie P. Sufficient conditions for distance reduction between the minimizers of non-convex quadratic functions in the trust region [R]. 2023.

[100] Björkman M, Holmström K. Global optimization of costly nonconvex functions using radial basis functions [J]. Optimization and Engineering, 2000, 1: 373-397.

[101] Wild S M, Regis R G, Shoemaker C A. ORBIT: Optimization by radial basis function interpolation in trust-regions [J]. SIAM Journal on Scientific Computing, 2008, 30(6): 3197-3219.

[102] Xie P, Yuan Y. A derivative-free optimization algorithm combining line-search and trust-region techniques [J]. Chinese Annals of Mathematics, Series B, 2023, 44(5): 719-734.

[103] Bandeira A S, Scheinberg K, Vicente L N. Convergence of trust-region methods based on probabilistic models [J]. SIAM Journal on Optimization, 2014, 24(3): 1238-1264.

[104] Gratton S, Royer C W, Vicente L N, et al. Complexity and global rates of trust-region methods based on probabilistic models [J]. IMA Journal of Numerical Analysis, 2017, 38(3): 1579-1597.

[105] Van Laarhoven P J M, Aarts E H L. Simulated Annealing: Theory and Applications: volume 37 [M]. Dordrecht: Springer Netherlands, 1987.

[106] Goldberg D E. Genetic algorithms in search, optimization and machine learning [M]. Boston: Addison−Wesley Longman, 1989.

[107] Goldberg D E, Holland J H. Genetic algorithms and machine learning [J]. Machine Learning, 1988, 3(2-3): 95-99.

[108] Holland J H. Adaptation in natural and artificial systems: Introductory analysis with applications to biology, control, and artificial intelligence [M]. Cambridge, MA: MIT Press, 1992.

[109] Kivkpatrick S. Optimization by simulated annealing: Quantitative studies [J]. Journal of Statistical Physics, 1984, 34(5/6): 975-986.

[110] Kirkpatrick S, Gelatt C, Vecchi M. Optimization by simulated annealing [J]. Science (New York, N.Y.), 1983, 220: 671-80.

[111] Cartis C, Gould N I M, Toint Ph L. On the oracle complexity of first-order and derivative-free algorithms for smooth nonconvex minimization [J]. SIAM Journal on Optimization, 2012, 22(1): 66-86.

[112] Sarker R, Mohammadian M, Yao X. Evolutionary Optimization: volume 48 [M]. Kluwer Academic Pub, 2002.

[113] Hansen N, Ostermeier A. Adapting arbitrary normal mutation distributions in evolution strategies: The covariance matrix adaptation [C]//Proceedings of IEEE International Conference on Evolutionary Computation. IEEE, 1996: 312-317.

[114] Hansen N, Ostermeier A. Completely derandomized self-adaptation in evolution strategies [J]. Evolutionary Computation, 2001, 9(2): 159-195.

[115] Hansen N. The CMA evolution strategy: A tutorial [R]. 2016.

[116] Wu W, Zhang F. A speeding-up and slowing-down strategy for distributed source seeking with robustness analysis [J]. IEEE Transactions on Control of Network Systems, 2015, 3(3): 231-240.

[117] Al-Abri S, Lin T X, Tao M, et al. A derivative-free optimization method with application to functions with exploding and vanishing gradients [J]. IEEE Control Systems Letters, 2020, 5(2): 587-592.

[118] Zhang H, Conn A R, Scheinberg K. A derivative-free algorithm for least-squares minimization [J]. SIAM Journal on Optimization, 2010, 20(6): 3555-3576.

[119] Cartis C, Roberts L. A derivative-free Gauss–Newton method [J]. Mathematical Programming Computation, 2019, 11(4): 631-674.

[120] Grapiglia G N, Yuan J, Yuan Y. A derivative-free trust-region algorithm for composite nonsmooth optimization [J]. Computational and Applied Mathematics, 2016, 35(2): 475-499.

[121] Liu S, Wang L, Xiao N, et al. An inexact preconditioned zeroth-order proximal method for composite optimization [Z]. 2024.

[122] Xie P. A derivative-free trust-region method for optimization on the ellipsoid [J]. Journal of Physics: Conference Series, 2023, 2620(1): 012007.

[123] Tang Y, Li N. Distributed zero-order algorithms for nonconvex multi-agent optimization [C]//2019 57th Annual Allerton Conference on Communication, Control, and Computing (Allerton). 2019: 781-786.

[124] Pelikan M, Goldberg D E, Cantú-Paz E. BOA: The Bayesian optimization algorithm [C]// Proceedings of the Genetic and Evolutionary Computation Conference GECCO-99: volume 1. Burlington: Morgan Kaufmann Publishers, 1999: 525-532.

[125] Gratton S, Royer C W, Vicente L N, et al. Direct search based on probabilistic descent [J]. SIAM Journal on Optimization, 2015, 25(3): 1515-1541.

[126] Ghadimi S, Lan G. Stochastic first-and zeroth-order methods for nonconvex stochastic programming [J]. SIAM Journal on Optimization, 2013, 23(4): 2341-2368.

[127] Larson J, Wild S M. A batch, derivative-free algorithm for finding multiple local minima [J]. Optimization and Engineering, 2016, 17(1): 205-228.

[128] Larson J, Menickelly M, Wild S M. Derivative-free optimization methods [J]. Acta Numerica, 2019, 28: 287-404.

[129] 张在坤. 无导数优化 [M]//中国学科发展战略：数学优化. 科学出版社, 2021: 84-92.

[130] Rios L M, Sahinidis N V. Derivative-free optimization: a review of algorithms and comparison of software implementations [J]. Journal of Global Optimization, 2013, 56(3): 1247-1293.

[131] Hansen N, Müller S D, Koumoutsakos P. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES) [J]. Evolutionary Computation, 2003, 11(1): 1-18.

[132] Scheinberg K. Manual for Fortran software package DFO version 2.0 [R]. Tech. rep., 2003.

[133] Kelley C T. Users guide for IMFIL version 1.0 [R]. 2011.

[134] Ge D, Liu T, Liu J, et al. SOLNP+: A derivative-free solver for constrained nonlinear optimization [R]. 2022.

[135] Powell M J D. A tolerant algorithm for linearly constrained optimization calculations [J]. Mathematical Programming, 1989, 45: 547-566.

[136] Powell M J D. A direct search optimization method that models the objective and constraint functions by linear interpolation [M]//Advances in Optimization and Numerical Analysis. Dordrecht: Springer, 1994: 51-67.

[137] Powell M J D. UOBYQA: unconstrained optimization by quadratic approximation [J]. Mathematical Programming, 2002, 92(3): 555-582.

[138] Powell M J D. The BOBYQA algorithm for bound constrained optimization without derivatives [J]. Cambridge NA Report NA2009/06, University of Cambridge, Cambridge, 2009: 26-46.

[139] Powell M J D. On fast trust region methods for quadratic models with linear constraints [J]. Mathematical Programming Computation, 2015, 7(3): 237-267.

[140] Cartis C, Fiala J, Marteau B, et al. Improving the flexibility and robustness of model-based derivative-free optimization solvers [J]. ACM Transactions on Mathematical Software, 2019, 45(3): 1-41.

[141] Cartis C, Roberts L. Scalable subspace methods for derivative-free nonlinear least-squares optimization [J]. Mathematical Programming, 2023, 199(1-2): 461-524.

[142] Ragonneau T M, Zhang Z. PDFO: a cross-platform package for Powell's derivative-free optimization solvers [R]. 2023.

[143] Ragonneau T M. Model-based derivative-free optimization methods and software [D]. Hong Kong: Department of Applied Mathematics, The Hong Kong Polytechnic University, 2022.

[144] Xie P. NEWUOA-Matlab-Version-2.0 [R/OL]. 2022. https://github.com/PengchengXieLSEC/NEWUOA-Matlab-Version-2.0/releases/tag/2.0.

[145] Xie P. BOBYQA-Matlab-Version-1.0 [R/OL]. 2023. https://github.com/PengchengXieLSEC/BOBYQA-Matlab-Version-1.0/releases/tag/Version-1.0.

[146] Zhang Z. PRIMA: Reference implementation for Powell's methods with modernization and amelioration [R]. 2023.

[147] Spendley W, Hext G, Himsworth F. Sequential application of simplex designs in optimization and evolutionary operation [J]. Technometrics, 1962, 4: 441-461.

[148] Winfield D. Function minimization by interpolation in a data table [J]. IMA Journal of Applied Mathematics, 1973, 12.

[149] Tröltzsch A, Gratton S, Toint Ph L. A model-based trust-region algorithm for DFO and its adaptation to handle noisy functions and gradients [C]//The 21st International Symposium on Mathematical Programming. 2012.

[150] Xie P, Yuan Y. Derivative-free optimization with transformed objective functions and the algorithm based on the least Frobenius norm updating quadratic model [J]. Journal of the Operations Research Society of China, 2024: 1-37.

[151] Cartis C, Gould N I M, Toint Ph L. On the oracle complexity of first-order and derivative-free algorithms for smooth nonconvex minimization [J]. SIAM Journal on Optimization, 2012, 22(1): 66-86.

[152] Conn A R, Gould N I, Toint Ph. L. Trust Region Methods [M]. Philadelphia: SIAM, 2000.

[153] Bandeira A S, Scheinberg K, Vicente L N. Computation of sparse low degree interpolating polynomials and their application to derivative-free optimization [J]. Mathematical Programming, 2012, 134(1): 223-257.

[154] Zhang Z. Sobolev seminorm of quadratic functions with applications to derivative-free optimization [J]. Mathematical Programming, 2014, 146(1-2): 77-96.

[155] Berghen F V, Bersini H. CONDOR, a new parallel, constrained extension of Powell's UOBYQA algorithm: Experimental results and comparison with the DFO algorithm [J]. Journal of Computational and Applied Mathematics, 2005, 181(1): 157-175.

[156] Conn A R, Scheinberg K, Toint Ph L. A derivative free optimization algorithm in practice [C]//7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization. Reston: AIAA, 1998: 4718.

[157] Wild S M. MNH: A derivative-free optimization algorithm using minimal norm Hessians [C]//The Tenth Copper Mountain Conference on Iterative Methods. 2008.

[158] Oeuvray R, Bierlaire M. Boosters: A derivative-free algorithm based on radial basis functions [J]. International Journal of Modelling and Simulation, 2009, 29(1): 26-36.

[159] Marazzi M, Nocedal J. Wedge trust region methods for derivative free optimization [J]. Mathematical Programming, 2002, 91(2): 289-305.

[160] Conn A R, Scheinberg K, Toint Ph L. On the convergence of derivative-free methods for unconstrained optimization [M]//Approximation Theory and Optimization: Tributes to M.J.D. Powell. 1997: 83-108.

[161] Steihaug T. The conjugate gradient method and trust regions in large scale optimization [J]. SIAM Journal on Numerical Analysis, 1983, 20(3): 626-637.

[162] Toint Ph L. Towards an efficient sparsity exploiting newton method for minimization [M]// Sparse matrices and their uses. Academic Press, 1981: 57-88.

[163] Yuan Y. On the truncated conjugate gradient method [J]. Mathematical Programming, 2000, 87: 561-573.

[164] Dolan E D, Moré J J. Benchmarking optimization software with performance profiles [J]. Mathematical Programming, 2002, 91(2): 201-213.

[165] Moré J J, Wild S M. Benchmarking derivative-free optimization algorithms [J]. SIAM Journal on Optimization, 2009, 20(1): 172-191.

[166] Yuan Y. Subspace techniques for nonlinear optimization [M]//Jeltsch R, Li D Q, Sloan I H. Some Topics in Industrial and Applied Mathematics. Beijing: Higher Education Press, 2007: 206-218.

[167] Yuan Y. Subspace methods for large scale nonlinear equations and nonlinear least squares [J]. Optimization and Engineering, 2009, 10(2): 207-218.

[168] Yuan Y. A review on subspace methods for nonlinear optimization [C]//Proceedings of the International Congress of Mathematics. 2014: 807-827.

[169] Berglund E, Khirirat S, Wang X. Zeroth-order randomized subspace newton methods [C]// 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2022: 6002-6006.

[170] Powell M J D. Beyond symmetric Broyden for updating quadratic models in minimization without derivatives [J]. Mathematical Programming, 2013, 138(1): 475-500.

[171] Conn A R, Toint Ph L. An algorithm using quadratic interpolation for unconstrained derivative free optimization [M]//Di Pillo G, Giannessi F. Nonlinear Optimization and Applications. Boston: Springer, 1996: 27-47.

[172] Conn A R, Scheinberg K, Toint Ph L. A derivative free optimization algorithm in practice [C]//Proceedings of the 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization. AIAA, 1998: 129-139.

[173] Evans L C. Partial differential equations [M]. Providence, R.I.: American Mathematical Society, 2010.

[174] Powell M J D. On updating the inverse of a KKT matrix [J]. Numerical Linear Algebra and Optimization, ed. Yaxiang Yuan, Science Press (Beijing), 2004: 56-78.

[175] Conn A R, Scheinberg K, Vicente L N. Geometry of interpolation sets in derivative free optimization [J]. Mathematical Programming, 2008, 111(1): 141-172.

[176] Powell M J D. On the Lagrange functions of quadratic models that are defined by interpolation [J]. Optimization Methods and Software, 2001, 16(1-4): 289-309.

[177] Gould N I M, Orban D, Toint Ph L. CUTEr and SifDec: A constrained and unconstrained testing environment, revisited [J]. ACM Transactions on Mathematical Software, 2003, 29 (4): 373-394.

[178] Moré J J, Garbow B S, Hillstrom K E. Testing unconstrained optimization software [J]. ACM Transactions on Mathematical Software, 1981, 7(1): 17-41.

[179] Conn A R, Gould N I M, Toint Ph L. Testing a class of methods for solving minimization problems with simple bounds on the variables [J]. Mathematics of Computation, 1988, 50 (182): 399-430.

[180] Lukšan L, Matonoha C, Vlcek J. Modified CUTE problems for sparse unconstrained optimization [R]. 2010.

[181] Li Y J, Li D H. Truncated regularized Newton method for convex minimizations [J]. Computational Optimization and Applications, 2009, 43: 119-131.

[182] Jarre F. EXPSUM Dataset [R/OL]. 2015. http://www.opt.uni-duesseldorf.de/~jarre/dot/f_cx.m.

[183] Conn A, Gould N, Lescrenier M, et al. Performance of a multifrontal scheme for partially separable optimization [M]//Advances in Optimization and Numerical Analysis. Dordrecht: Springer, 1994: 79-96.

[184] Toint Ph L. Some numerical results using a sparse matrix updating formula in unconstrained optimization [J]. Mathematics of Computation, 1978, 32(143): 839-851.

[185] Andrei N. An unconstrained optimization test functions collection [J]. Advanced Modeling and Optimization, 2008, 10(1): 147-161.

[186] Li G. The secant/finite difference algorithm for solving sparse nonlinear systems of equations [J]. SIAM Journal on Numerical Analysis, 1988, 25: 1181-1196.

[187] Robinson S M. Quadratic interpolation is risky [J]. SIAM Journal on Numerical Analysis, 1979, 16(3): 377-379.

[188] Conn A R, Scheinberg K, Toint Ph L. Recent progress in unconstrained nonlinear optimization without derivatives [J]. Mathematical Programming, 1997, 79(1): 397-414.

[189] The MathWorks Inc. MATLAB (R2023b) [M]. Natick, Massachusetts, 2023.

[190] Lagarias J C, Reeds J A, Wright M H, et al. Convergence properties of the Nelder-Mead simplex method in low dimensions [J]. SIAM Journal on Optimization, 1998, 9(1): 112-147.

[191] Higham N J. Optimization by direct search in matrix computations [J]. SIAM Journal on Matrix Analysis and Applications, 1993, 14(2): 317-333.

[192] Kelley C T. Iterative Methods for Optimization [M]. Philadelphia: SIAM, 1999.

[193] Higham N J. The matrix computation toolbox [R]. 2002.

[194] Dinur I, Nissim K. Revealing information while preserving privacy [C]//Proceedings of the 22nd ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems. 2003: 202-210.

[195] Dwork C, Nissim K. Privacy-preserving datamining on vertically partitioned databases [C]// Annual International Cryptology Conference. Springer, 2004: 528-544.

[196] Dwork C, McSherry F, Nissim K, et al. Calibrating noise to sensitivity in private data analysis [C]//Halevi S, Rabin T. Theory of Cryptography: Third Theory of Cryptography Conference. Berlin: Springer, 2006: 265-284.

[197] Nissim K, Raskhodnikova S, Smith A. Smooth sensitivity and sampling in private data analysis [C]//Proceedings of the 39th Annual ACM Symposium on Theory of Computing. 2007: 75-84.

[198] Kasiviswanathan S P, Lee H K, Nissim K, et al. What can we learn privately? [J]. SIAM Journal on Computing, 2011, 40(3): 793-826.

[199] McSherry F, Talwar K. Mechanism design via differential privacy [C]//48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07). IEEE, 2007: 94-103.

[200] Wang Y, Hale M, Egerstedt M, et al. Differentially private objective functions in distributed cloud-based optimization [C]//2016 IEEE 55th Conference on Decision and Control (CDC). IEEE, 2016: 3688-3694.

[201] Liu J, Huang X, Liu J K. Secure sharing of personal health records in cloud computing: Ciphertext-policy attribute-based signcryption [J]. Future Generation Computer Systems, 2015, 52: 67-76.

[202] Kusner M, Gardner J, Garnett R, et al. Differentially private Bayesian optimization [C]// International Conference on Machine Learning. PMLR, 2015: 918-927.

[203] Deng G, Ferris M C. Adaptation of the UOBYQA algorithm for noisy functions [C]// Proceedings of the 2006 Winter Simulation Conference. Piscataway: IEEE, 2006: 312-319.

[204] Jamieson K G, Nowak R, Recht B. Query complexity of derivative-free optimization [C]// Pereira F, Burges C, Bottou L, et al. Advances in Neural Information Processing Systems: volume 25. New York: Curran Associates, Inc., 2012.

[205] Wilson J D, Wintucky E G, Vaden K R, et al. Advances in space traveling-wave tubes for NASA missions [J]. Proceedings of the IEEE, 2007, 95(10): 1958-1967.

[206] Levush B. The design and manufacture of vacuum electronic amplifiers: Progress and challenges [C]//2019 International Vacuum Electronics Conference (IVEC). IEEE, 2019: 1-5.

[207] Gould N I M, Orban D, Toint Ph L. CUTEst: a constrained and unconstrained testing environment with safe threads for mathematical optimization [J]. Computational Optimization and Applications, 2015, 60: 545-557.

[208] Nocedal J, Yuan Y. Combining trust region and line search techniques [C]//In: Advances in Nonlinear Programming. Boston: Springer, 1998: 153-175.

[209] Khalil H K. Nonlinear Systems [M]. Upper Saddle River: Prentice Hall, 2002.

[210] Powell M J D. Least Frobenius norm updating of quadratic models that satisfy interpolation conditions [J]. Mathematical Programming, 2004, 100(1): 183-215.

中国科学院数学与系统科学研究院 XiePengCheng 2025-07-31 09:40:21

中国科学院数学与系统科学研究院 XiePengCheng 2025-07-31 09:40:21

致 谢

本文在我的导师袁亚湘院士的悉心指导下完成. 袁老师对我的培养是用心的、全方面的, 大至理想、信念, 小至定理、标点, 用学术和生活中的言传身教带我成长, 我永怀感激. 学术方面, 袁老师对我博士期间的每一项研究都给予了重要关注、指导和鼓励: 周末加班为我改文章、带我在办公室黑板前推导计算、在国际大会上亲自宣讲同我的合作工作、给我修改学术报告的词稿和内容. 点滴间尽显大师风范, 言行中倾注无限心血. 交流方面, 袁老师鼓励我勇于探讨、大胆交流: 带领和支持我积极参与到国际会议、学术竞赛、学生学术组织建设中去. 在袁老师的支持下, 我曾在 ICIAM2023 做报告，曾先后同三位 SIAM 主席面对面交流. 生活方面, 袁老师带我爬山、自驾、打牌, 寓教于乐, 时刻给予着父亲般的关爱.

我深知: 是袁老师培塑了我的学术观和人生观. 我会用实际行动践行袁老师的谆谆教诲, 努力向成为一名理论顶天、应用立地的计算数学者而奋斗. 同时，我也要衷心感谢我的师母对我一直以来的关爱和照顾，让我在北京求学期间时刻感受着家的温暖, 让我能够开心地学习和研究.

感谢戴彧虹老师在我科研和学习上的帮助, 戴老师的建议、见解和指导总是兼具创新性和可行性, 大大开阔了我的科研思路、给予了我科研信心. 戴老师常常教育我打开脑洞、夯实基础、学会应用, 还多次给我介绍重要的无导数优化应用问题, 给我成长的平台和机会. 感谢郭田德老师带我走进国科大, 郭老师在雁栖湖的动员、在视频授课时的教导我历历在目, 郭老师是我们国科大数学学子的恩师和榜样. 感谢课题组的刘歆老师在学术和生活上的指引, 我有幸聆听刘老师的凸分析课程，收获颇丰. 刘老师用智慧持续带动着我多多思考、用跑步健身启发着我要带着强健的体魄来学习, 提升了我的学研自信. 感谢课题组的刘亚锋老师, 我们办公室同楼层，和刘老师在讨论班以及平时的每次相遇都让我受益匪浅, 刘老师给了我很多科研和生活上的鼓励. 感谢课题组的马俊杰老师, 马老师在科研方面给了我很多建议和帮助. 感谢课题组的孙聪老师, 孙老师给予了我很多关心和鼓励. 感谢课题组的史斌老师, 史老师给我介绍了很多现代优化知识和前沿研究. 感谢课题组的高斌老师, 高老师在多方面帮助和推动了我的进步. 感谢课题组的陈亮老师，陈老师为我提出了很多重要的建议.

特别感谢我的师爷 Michael Powell 院士留给世人一系列无导数优化算法, 这是一笔宝藏, 是吸引我、引导我的光束. 同时，感谢 Coralia Cartis 教授和张在坤老师为我的无导数优化研究提供了重要帮助.

感谢谢金星老师、王彦飞老师、徐大川老师、文再文老师、夏勇老师、徐姿老师、牛凌峰老师给予的指导和帮助. 感谢王晓老师、李庆娜老师、姜波老师、崔春风老师给予的照顾和鼓励.

感谢课题组的顾然、高斌、赵亮、周睿智、陈伟坤、王小玉、陈亮、傅凯、张瑞、肖纳川、吴宇宸、杨沐明、陈雅丹、赵浩天、张瑞进、吉振远、刘为、黄磊、

姜博鸥、陈圣杰、张吾帅君、王磊、汪思维、陈硕、裴骞、王圣超、张跃、赵成、胡雨宽、武哲宇、张亦、章煜海、胡雨婷、刘上琳、王子岳、彭任锋、李博文、汤宇扬、苏昭纲、李冠达、姜林硕、郑浩然、范熙来、杨俨、胡威、刁若愉、罗舟行、张宇航、岳艺双、张思远、徐勐、李新鹏、李雨芯、金泽龙、王宇扬、黄辰飞、张博洋等师兄弟姐妹们, 共同奋斗的点滴令我难忘. 感谢博士后张国涵、黄娜、张婷、刘泽显、戴金雨、王姝、徐加樑、李成梁、于腾腾、张旭、王嘉妮、张凯丽、章丽、魏奇远、张帆等师兄师姐, 他们给予我的学研建议是我的宝贵财富.

感谢父母、家人. 特别感谢我的爷爷，他是一名优秀的中学数学教师，为我在数学上启蒙. 谨以此文献给所有关心帮助过我的人.

2024 年 6 月

中国科学院数学与系统科学研究院 XiePengCheng 2025-07-31 09:40:21

作者简历及攻读学位期间发表的学术论文与其他相关学术成果

作者简历:

谢鹏程, 男, 出生于 1998 年 10 月.

2015 年 8 月—2019 年 6 月, 在西安交通大学数学与统计学院获得学士学位.

2019 年 8 月—2024 年 6 月, 在中国科学院数学与系统科学研究院攻读博士学位.

攻读博士学位期间的部分已发表学术论文:

(1) Pengcheng Xie, Ya-xiang Yuan. Derivative-free optimization with transformed objective functions and the algorithm based on the least Frobenius norm updating quadratic model [J]. Journal of the Operations Research Society of China (2024).

(2) Pengcheng Xie, Ya-xiang Yuan. A derivative-free optimization algorithm combining line-search and trust-region techniques [J]. Chinese Annals of Mathematics, Series B, 44, 719–734 (2023).

(3) Pengcheng Xie. A derivative-free trust-region method for optimization on the ellipsoid [J]//Journal of Physics: Conference Series. IOP Publishing, 2023, 2620(1): 012007.

(4) Shuoran Li, Pengcheng Xie, Zihao Zhou, et al. Simulation of interaction of folded waveguide space traveling wave tubes with derivative-free mixedinteger based NEWUOA algorithm [C]//2021 7th International Conference on Computer and Communications (ICCC), Chengdu, China, 2021: 1215-1219.

部分获奖情况:

(1) 2023 年 9 月　中国科学院数学与系统科学研究院院长奖学金
(2) 2021 年 12 月　中国研究生数学建模竞赛全国一等奖
(3) 2021 年 10 月　全国向上向善好青年
(4) 2021 年 11 月　国家基础学科拔尖学生培养计划优秀学生
(5) 2021 年 5 月　北京市三好学生
(6) 2020 年 12 月　国家奖学金
(7) 2020 年 4 月　中国科学院优秀共青团员
(8) 2019 年 9 月　中国科学院数学与系统科学研究院华罗庚奖学金

中国科学院数学与系统科学研究院 XiePengCheng 2025-07-31 09:40:21