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

博士学位论文

正交约束优化：理论、算法与应用

作者姓名：高 斌

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：应用数学

培养单位：中国科学院数学与系统科学研究院

2019 年 6 月

Optimization with Orthogonality Constraints: Theory, Algorithms and Applications

A dissertation submitted to the

University of Chinese Academy of Sciences

in partial fulfillment of the requirement

for the degree of

Doctor of Philosophy

in Applied Mathematics

By

Bin Gao

Supervisor: Professor Ya-xiang Yuan

Institute of Computational Mathematics and Scientific/Engineering Computing

Academy of Mathematics and Systems Science

Chinese Academy of Sciences

June, 2019

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。尽我所知，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已经发表或撰写过的研究成果。对论文所涉及的研究工作做出贡献的其他个人和集体，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保存和使用学位论文的规定，即中国科学院大学有权保留送交学位论文的副本，允许该论文被查阅，可以按照学术研究公开原则和保护知识产权的原则公布该论文的全部或部分内容，可以采用影印、缩印或其他复制手段保存、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：

献给我的父亲母亲

摘 要

正交约束优化问题是指变量为矩阵且满足正交性约束的最优化问题, 其可行集也被称为 Stiefel 流形. 这类优化模型在科学计算、材料科学及数据科学等领域有着广泛的应用. 由于正交化过程的高计算代价和低可扩展性, 正交约束优化算法的研究遇到了新的瓶颈和挑战. 本文系统地研究了正交约束优化问题, 设计了三类优化算法, 并将新算法应用于实际问题中.

首先我们系统地研究了正交约束优化问题. 我们从黎曼流形优化和欧式空间约束优化两个角度, 分别推导了问题的最优性条件. 其中, 通过分析一大类黎曼度量下的黎曼梯度, 我们得到了问题各种描述下一阶最优性条件的对应关系. 此外, 我们还证明了在任意一阶稳定点处, 正交约束优化问题的 Lagrange 乘子具有显式表达式. 这些性质的刻画将对本文的算法设计起到至关重要的作用.

接着我们针对一大类正交约束优化问题, 提出了乘子校正算法框架. 其主要包含两个步骤, 分别是函数值下降步和乘子校正步. 不同于以往的经典算法, 在我们的算法框架中, 函数值下降步采用标准的欧式负梯度方向, 而不是 Stiefel 流形的切空间方向. 另一方面, 我们构造的乘子校正步进一步使函数值下降, 同时也保证了乘子的对称性. 基于此算法框架, 我们提出了两大类算法. 第一类是梯度下降方法, 其中包括梯度反射法和梯度投影法. 第二类采用以列为块的块坐标下降方法. 进一步, 我们证明了算法的全局收敛性. 数值实验表明我们的算法框架具有很大的潜力.

然后我们将乘子校正步推广到一般的 Stiefel 流形收缩类算法, 得到了子空间加速的收缩类算法. 通过利用乘子校正步的子空间最优性质, 我们设计了一类两阶段的子空间加速算法. 第一阶段是函数值下降法, 第二阶段是子空间加速步. 将 Stiefel 流形的收缩类线搜索算法应用于第一阶段, 我们证明了加速算法的全局收敛性和局部线性收敛速度. 数值实验展示了加速技术的有效性.

之后针对一般的正交约束优化问题, 我们提出了基于增广 Lagrange 函数的并行算法. 由于正交化过程的可扩展性较低, 我们考虑不可行方法并采用增广 Lagrange 罚函数. 不同于经典的增广 Lagrange 函数法, 在我们的算法中, 原始变量的更新通过极小化增广 Lagrange 函数的邻近点线性化逼近得到. 同时, Lagrange 乘子由其一阶稳定点处的显式表达式更新得到. 由此, 算法的主要步骤都可以很自然地进行矩阵计算层面的并行化. 进一步, 我们建立了算法的全局收敛性, 分析

了算法的最坏情况复杂度以及局部收敛速度. 此外, 为了减弱算法对罚参数的敏感性, 我们提出了改进的可并行列极小化算法. 串行的数值实验说明乘子的新更新方式显著加速了算法的收敛速度, 并且数值表现与已有的可行方法不相上下. 并行环境下的数值实验验证了我们的算法具有较高的可扩展性.

最后我们将乘子校正算法和基于增广 Lagrange 函数的并行算法应用于电子结构计算. 我们考虑了电子结构计算中的 Kohn-Sham 密度泛函理论, 其离散模型通常表述为一个带有正交约束的优化问题. 在串行环境下, 我们测试了 18 个不同的分子结构, 数值实验显示了我们的新算法优于已有的经典算法. 在并行环境下, 我们测试了简化的 Kohn-Sham 总能量极小化问题, 数值结果显示我们提出的并行算法具有较高的可扩展性.

关键词： 正交约束, Stiefel 流形, 块坐标下降法, 收缩类方法, 增广 Lagrange 函数, 并行计算, 电子结构计算

Abstract

Optimization problems with orthogonality constraints is a class of matrix optimization problems such that the matrix variable satisfies orthogonality, and the feasible region of which is also known as Stiefel manifold. This type of problems has many applications in scientific computing, material sciences and data science. Due to the high computational cost and low scalability of orthonormalization procedure, there is a huge demand for efficient solvers of these problems. In this dissertation, we systemically study the orthogonally constrained optimization problems, propose three kinds of algorithms, and apply these methods to an application problem.

Firstly, we systemically study the optimization problems with orthogonality constraints. It can be regarded as Riemannian optimization and also constrained optimization in Euclidean space. We obtain different optimality conditions from these two points of view, respectively. Through a class of Riemannian gradient, we establish the relationship between these optimality conditions. Moreover, we prove that the Lagrangian multipliers have a closed-form expression at any first-order stationary point. These propositions enlighten us to design our new algorithms.

Secondly, a new first-order framework is proposed for solving a class of optimization problems with orthogonality constraints. Our new framework combines a function value reduction step with a correction step. Different from the existing approaches, the function value reduction step searches along the standard Euclidean descent directions instead of in the tangent space of the Stiefel manifold, and the correction step further reduces the function value and guarantees a symmetric dual variable at the same time. We construct two types of algorithms based on this new framework. The first type is based on gradient reduction including the gradient reflection and the gradient projection approaches. Another one adopts a column-wise block coordinate descent scheme. Moreover, we prove the global convergence of our algorithmic framework. Numerical experiments illustrate that our new framework is of great potential.

Thirdly, we propose a class of retraction-based methods with subspace acceleration, by extending the correction step in the first-order framework to general retraction-based methods on Stiefel manifold. In fact, the correction step can be regarded as an optimiza-

tion problem in a subspace. Based on this, we construct a two-stage algorithm including function value reduction step and subspace acceleration step. We apply the retraction-based line-search method on Stiefel manifold to the function value reduction step, and prove the global convergence and local linear convergence rate of corresponding algorithm. Numerical experiments verify that our new acceleration technique is useful.

Fourthly, a class of parallel approaches based on the augmented Lagrangian function is proposed for solving optimization problems with orthogonality constraints. Unlike the classical augmented Lagrangian methods, in our algorithm (PLAM), the prime variables are updated by minimizing a proximal linearized approximation of the augmented Lagrangian function. Meanwhile, the dual variables are updated by a closed-form expression which holds at any first-order stationary point. Consequently, the main parts of the proposed algorithm can be parallelized naturally. We establish the global subsequence convergence, and analyze the worst-case complexity and local convergence rate for PLAM under some mild assumptions. To reduce the sensitivity of the penalty parameter, we put forward a modification of PLAM, which is called PCAL. Numerical experiments in serial illustrate that the novel updating rule for the Lagrangian multipliers significantly accelerates the convergence of PLAM. Under parallel environment, numerical tests demonstrate that PCAL attains good performance and high scalability.

Finally, we apply the new first-order framework and parallel approaches to electronic structure calculation. In this field, Kohn-Sham density functional theory (KSDFT) is known to be an important topic. The last step of KSDFT is an orthogonally constrained optimization problem. Under serial setting, we test 18 molecules in the KSSOLV platform. Numerical results illustrate that our algorithms outperform the existing methods. Besides, parallel experiments show that our approaches obtain a high scalability.

Keywords: Orthogonality constraints, Stiefel manifold, Block coordinate descent, Retraction, Augmented Lagrangian method, Parallel computing, Electronic structure calculation

目 录

图形列表

表格列表

符号列表

字符

$\mathbb{R}^{n\times p}$	欧式空间
$\mathbb{SR}^{p\times p}$	实对称矩阵全体
$\mathbb{D}^{p\times p}$	实对角矩阵全体
I	单位矩阵
X_i	矩阵 X 的第 i 列
X^k	变量 X 的第 k 次迭代
$\mathcal{M}$	黎曼流形
$\mathcal{S}_{n,p}$	Stiefel 流形
$\mathcal{T}_X\mathcal{M}$	黎曼流形 $\mathcal{M}$ 在 X 点处的切空间

算子

$X^\dagger$	矩阵 X 的伪逆
$\mathrm{tr}(A)$	矩阵 A 的迹
$\mathrm{diag}(A)$	矩阵 A 对角元构成的向量
$\mathrm{Diag}(x)$	向量 x 构成的对角矩阵
$\lambda_{\min}(A)$	矩阵 A 的最小特征值
$\sigma_{\min}(A)$	矩阵 A 的最小奇异值
$\Psi(A)$	矩阵 A 的对称化: $\frac{1}{2}(A+A^\top)$
$\Phi(A)$	矩阵 A 的对角元构成的对角矩阵: $\mathrm{Diag}(\mathrm{diag}(A))$

缩写

ADMM	交替方向乘子法
ALM	增广 Lagrange 函数法
BCD	块坐标下降法
BLAS	基础线性代数子程序库
KSDFT	Kohn-Sham 密度泛函理论

第 1 章　引言

最优化也被称为数学规划是运筹学的一个重要分支. 这一学科所研究的问题是讨论在众多的决策方案中, 什么样的决策最优以及怎样找出最优决策. 在构建寻找最优决策算法的过程中, 其理论与计算方法都得到了大力发展. 如今, 最优化理论与方法在国防、经济、金融、工程、管理、生物、医疗等许多领域有着广泛的应用. 在科技飞速发展的今天, 很多来自人工智能、机器学习等领域的问题最终也可以归结为最优化问题.

最优化的思想可以追溯到 17 世纪 Newton 和 Leibniz 创立微积分的时代, 那时极值问题就已被提出. 在 1947 年 Dantzig 提出求解线性规划的单纯形法之后, 最优化逐渐发展成为一门独立的学科. 随着计算机技术的发展以及实际应用的需要, 最优化理论与方法也在日新月异的不断进步. 最优化各个分支学科的研究也不断壮大起来, 例如线性规划, 非线性规划, 整数规划, 随机优化, 多目标优化, 矩阵优化, 流形优化等.

正交约束优化问题是指变量为矩阵且满足正交性约束条件的最优化问题. 其作为一类特殊的最优化问题在数值代数、材料科学、机器学习等领域中有着广泛的应用. 由于满足正交约束的矩阵全体构成了一个矩阵流形, 称为 Stiefel 流形, 因而此类问题既可以看成是一般的非线性规划, 也可以作为流形优化的一个特例. 随着大数据时代的到来, 实际应用问题中的数据规模和结构都发生了巨大变化, 如何高效的利用这些信息对于正交约束优化问题的求解至关重要. 因此, 越来越多的研究者开始关注此类问题. 本论文的出发点是将正交约束优化问题看成一般的非线性规划问题, 从欧式空间的角度, 直接对问题本身进行求解. 这样做的好处是避免了在 Stiefel 流形上的复杂计算, 而这恰好是已有流形优化收缩类算法的主要计算代价之一. 本文主要研究了欧式空间中的非收缩可行和不可行方法, 分别提出了三类算法框架, 并从理论和数值上验证了这些算法的有效性. 此外, 本文还考虑了一类正交约束优化的实际应用问题: 电子结构计算.

作为全文的引言, 接下来我们简要介绍论文所需的一些基本概念. 在 1.1 节, 我们引入欧式空间中的最优化问题, 并给出最优解及最优性条件的定义. 接着, 1.2 节介绍了黎曼流形优化的理论与方法. 然后, 我们在 1.3 节介绍并行计算的基本概念. 最后, 我们将本文的主要工作概括论述在 1.4 节.

1.1 欧式空间中的非线性规划

在本小节, 我们介绍欧式空间 $\mathbb{R}^n$ 中的非线性规划, 若非特别提及, 这里所涉及的主要结果大都来自于专著 [1–4].

1.1.1 最优化问题

在欧式空间 $\mathbb{R}^n$ 中, 非线性规划问题可以写作如下的形式,

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & f(x) \\ \text{s.t.} \quad & c_i(x) = 0, \quad i \in \mathcal{E}, \\ & c_i(x) \geq 0, \quad i \in \mathcal{I}, \end{aligned} \tag{1.1}$$

其中 f 和 c_i 都是定义在 $\mathbb{R}^n$ 上的实值函数, $\mathcal{E}$ 和 $\mathcal{I}$ 分别表示不同的有限指标集. 若我们定义满足约束条件的所有 x 构成的集合为问题 (1.1) 的可行域 Ω, 即

$$\Omega := \left\{x \in \mathbb{R}^n \mid c_i(x) = 0, i \in \mathcal{E}; c_i(x) \geq 0, i \in \mathcal{I}\right\},$$

则我们可以将问题 (1.1) 重写为更紧凑的形式,

$$\min_{x\in\Omega} \ f(x). \tag{1.2}$$

特别地, 如果 $\Omega = \mathbb{R}^n$, 我们称问题 (1.2) 为无约束优化问题, 否则称其为约束优化问题. 进一步, 如果可行域 Ω 是 $\mathbb{R}^n$ 中的凸集且目标函数 f 关于 x 是凸函数, 则我们称问题 (1.2) 为凸问题, 否则称其为非凸问题. 接下来我们给出全局最优点和局部最优点的定义.

定义 1.1 (全局最优点). *我们称 x^* 为问题 (1.2) 的全局最优点, 如果 $x^* \in \Omega$ 并且*

$$f(x) \geq f(x^*), \quad \forall x \in \Omega.$$

定义 1.2 (局部最优点). *我们称 x^* 为问题 (1.2) 的局部最优点, 如果 $x^* \in \Omega$ 并且存在 x^* 的一个领域 $\mathcal{N}(x^*)$ 使得*

$$f(x) \geq f(x^*), \quad \forall x \in \mathcal{N}(x^*) \cap \Omega.$$

进一步, 如果不等式 $f(x) > f(x^*)$ 对任意的 $x \in \mathcal{N}(x^*) \cap \Omega$ 都成立, 则 x^* 被称为问题 (1.2) 的严格局部最优点.

显然, 全局最优点也是局部最优点, 反之则不一定成立. 特别地, 当问题 (1.2) 为凸问题时, 由凸优化理论 [5, 6] 可知, 局部最优点也是全局最优点.

1.1.2 最优性条件

当 $\mathcal{E} = \mathcal{I} = \emptyset$ 时, $\Omega = \mathbb{R}^n$, 此时问题 (1.1) 退化为无约束优化问题

$$\min_{x \in \mathbb{R}^n} \ f(x), \tag{1.3}$$

其最优性的判定相对简单.

定理 1.1. *假设 x^* 是问题 (1.1) 的局部最优点, 函数 f 在 x^* 处连续可微, 则必有*

$$\nabla f(x^*) = 0.$$

接下来, 我们考虑约束优化问题 (1.2). 首先, 我们定义问题在任意可行点 x ($x \in \Omega$) 处的积极集 (active set) 为

$$\mathcal{A}(x) := \mathcal{E} \cup \left\{ i \in \mathcal{I} \mid c_i(x) = 0 \right\}.$$

积极集 $\mathcal{A}(x)$ 反映了问题在可行点 x 处的约束满足情况.

通常我们在算法设计的时候需要考虑目标函数 $f(x)$ 及其约束 $c_i(x)$ 的逼近形式, 为了保证一阶 Taylor 展开, 也就是线性逼近的有效性, 我们需要对约束函数 c_i 做一些自然且必要的假设. 约束规范性 (constraint qualification) 就是这样一类假设, 它保证了可行域 Ω 的线性逼近与原可行域的相似性. 下面我们给出一类广泛使用的约束规范性条件.

定义 1.3 (LICQ)**.** *给定可行点 x 及其积极集 $\mathcal{A}(x)$, 我们称线性独立约束规范条件 (Linear Independence Constraint Qualification, 简称 LICQ) 成立当且仅当集合 $\left\{ \nabla c_i(x), i \in \mathcal{A}(x) \right\}$ 线性独立.*

接下来我们给出 x^* 成为局部最优点的一阶必要性条件. 在陈述条件之前, 我们先定义问题 (1.1) 的 Lagrange 函数为

$$\mathcal{L}(x, \lambda) := f(x) - \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i c_i(x), \tag{1.4}$$

其中 $\lambda = (\lambda_i)_{i \in \mathcal{E} \cup \mathcal{I}}$, 且 λ_i 称为约束 $c_i(x)$ 对应的 Lagrange 乘子.

定理 1.2 (一阶必要性条件). *假设 x^* 是问题 (1.1) 的局部最优点, 函数 f 和 c_i 连续可微并且在 x^* 处 LICQ 约束规范性条件成立. 则存在 Lagrange 乘子 $\lambda^* =$*

$\left(\lambda_i^*\right)_{i\in\mathcal{E}\cup\mathcal{I}}$, 使得下式在 (x^*,λ^*) 处成立,

$$\nabla_x\mathcal{L}(x^*,\lambda^*)=0, \tag{1.5a}$$

$$c_i(x^*)=0,\quad \forall\, i\in\mathcal{E}, \tag{1.5b}$$

$$c_i(x^*)\geq 0,\quad \forall\, i\in\mathcal{I}, \tag{1.5c}$$

$$\lambda_i^*\geq 0,\quad \forall\, i\in\mathcal{I}, \tag{1.5d}$$

$$\lambda_i^* c_i(x^*)=0,\quad \forall\, i\in\mathcal{E}\cup\mathcal{I}. \tag{1.5e}$$

(1.5) 式通常被称为 Karush-Kuhn-Tucker 条件, 简称 KKT 条件. 在本文的算法设计中, KKT 条件占有重要的地位, 因为本文最主要的贡献之一是提出了新的 Lagrange 乘子显式更新公式, 而其恰好是由 KKT 条件直接推导而来.

为了描述问题 (1.1) 的二阶最优性条件, 我们首先引入切锥 (tangent cone) 的概念. 假设 LICQ 约束规范条件在给定可行点 x 处成立, 并且其积极集为 $\mathcal{A}(x)$, 则约束集的切锥等价于如下的线性化可行方向集

$$\mathcal{F}(x):=\left\{d\in\mathbb{R}^n\;\middle|\;\begin{array}{ll} d^\top\nabla c_i(x)=0, & \forall\, i\in\mathcal{E},\\ d^\top\nabla c_i(x)=0, & \forall\, i\in\mathcal{A}(x)\cap\mathcal{I}\end{array}\right\}.$$

值得说明的是, 切锥的定义并不依赖于可行域 Ω 的代数形式, 事实上, 它只依赖于可行域的几何性质. 这里, 线性化可行方向集 $\mathcal{F}(x)$ 包含两种类型的向量. 第一种, 当 $w\in\mathcal{F}(x)$ 满足 $w^\top\nabla f(x)\neq 0$ 时, 我们可以很自然地得到函数值 f 的可行下降方向. 然而, 当 $w\in\mathcal{F}(x)$ 且满足 $w^\top\nabla f(x)=0$ 时, 约束集的一阶信息并不足以刻画沿此方向的函数值是否上升或下降. 因此, 在可行点 x^* 及其满足 KKT 条件 (1.5) 的 Lagrange 乘子 λ^* 处, 我们将 $\mathcal{F}(x^*)$ 中所有满足 $w^\top\nabla f(x^*)=0$ 的第二种向量 w 收集起来, 定义为临近锥 (critical cone)

$$C(x^*,\lambda^*)=\left\{w\in\mathcal{F}(x^*)\mid \nabla c_i(x^*)^\top w=0,\forall\, i\in\mathcal{A}(x)\cap\mathcal{I}\ \text{且}\ \lambda_i^*>0\right\}.$$

利用目标函数 f 和约束 c_i 的二阶信息, 我们得到当 x^* 为局部最优点时, Lagrange 函数的 Hesse 矩阵在集合 $C(x^*,\lambda^*)$ 上具有非负曲率. 更具体的, 我们引入如下的二阶必要性和充分性条件.

定理 1.3 (二阶必要性条件). *假设 x^* 是问题 (1.1) 的局部最优点, 并且在 x^* 处 LICQ 约束规范性条件成立. 令 λ^* 是满足 KKT 条件 (1.5) 的 Lagrange 乘子. 则下式成立,*

$$w^\top\nabla^*_{xx}\mathcal{L}(x^*,\lambda^*)w\geq 0,\quad \forall\, w\in C(x^*,\lambda^*).$$

定理 1.4 (二阶充分性条件). *假设对于可行点 $x^* \in \mathbb{R}^n$, 存在 Lagrange 乘子 λ^* 使得 KKT 条件 (1.5) 满足. 同时假设下式成立,*

$$w^\top \nabla^*_{xx} \mathcal{L}(x^*, \lambda^*) w > 0, \quad \forall\, w \in C(x^*, \lambda^*), w \neq 0.$$

则 x^* 是问题 (1.1) 的严格局部最优点.

1.1.3 无约束优化问题的梯度法

除了问题本身已有显式解或其他极少数特殊情况外, 无约束优化问题 (1.3) 的求解都是使用迭代法. 即给定初始值 $x^0 \in \mathbb{R}^n$, 然后由算法生成迭代点列 $x^k (k = 1, 2, \dots)$, 直到其收敛到满足最优性条件的解. 线搜索方法 (line search) 是最常见的一类迭代法. 它的主要思想是在当前迭代点选取一个搜索方向, 接着在此方向选取合适的搜索步长, 由此得到下一步迭代点. 梯度法是一种特殊的线搜索方法, 它采用迭代公式

$$x^{k+1} = x^k - \alpha_k \nabla f(x^k), \tag{1.6}$$

其中 $-\nabla f(x^k)$ 作为搜索方向, $\alpha_k > 0$ 作为搜索步长. 不同的步长选取导致了不同的梯度方法.

最简单的梯度法是最速下降法 (steepest descent), 由 Cauchy [7] 在 1847 年提出, 它的步长通过精确线搜索得到, 也就是

$$\alpha_k^{\mathrm{SD}} = \arg\min_{\alpha > 0} f(x^k - \alpha \nabla f(x^k)).$$

通常, 当问题的条件数很大时, 应用最速下降法, 迭代点列会产生 “zigzag” 现象 [8, 9], 这极大限制了梯度法的有效性.

1988 年, Barzilai 和 Borwein (BB) 的工作 [10] 很大程度上改变了我们对于梯度类方法的看法. 在他们的工作中, 迭代公式 (1.6) 被重新描述为

$$x^{k+1} = x^k - D^k \nabla f(x^k),$$

其中 $D_k = \alpha_k I$. 我们知道拟牛顿方法 (quasi-Newton) [11] 中的割线方程 (secant equation) 满足如下形式,

$$B^k s^{k-1} = y^{k-1}, \tag{1.7}$$

其中 $s^{k-1} = x^k - x^{k-1}$, $y^{k-1} = \nabla f(x^k) - \nabla f(x^{k-1})$, 拟牛顿矩阵 B_k 是 Hesse 矩阵 $\nabla^2 f(x^k)$ 的某种近似. 拟牛顿法具有良好的收敛性质, 其主要原因是拟牛顿矩阵用

割线近似切线, 从而得到针对牛顿方向很好的近似. 因此我们也希望矩阵 $(D^k)^{-1}$ 满足割线方程 (1.7). 文献 [10] 提出了可使 α 极小化的最小二乘问题,

$$\min_{\alpha} \left\|(\alpha I)^{-1} s^{k-1} - y^{k-1}\right\|,$$

其解为

$$\alpha_k^{\mathrm{BB1}} = \frac{(s^{k-1})^\top s^{k-1}}{(s^{k-1})^\top y^{k-1}}.$$

另一方面, 我们也可以对应地考虑问题

$$\min_{\alpha} \left\|s^{k-1} - (\alpha I) y^{k-1}\right\|.$$

由此, 我们得到另一个步长选取方式

$$\alpha_k^{\mathrm{BB2}} = \frac{(s^{k-1})^\top y^{k-1}}{(y^{k-1})^\top y^{k-1}}.$$

当 $(s^{k-1})^\top y^{k-1} > 0$ 时, 由 Cauchy–Schwarz 不等式 $((s^{k-1})^\top y^{k-1})^2 \le \left\|s^{k-1}\right\|^2 \left\|y^{k-1}\right\|^2$, 我们得到 $\alpha_k^{\mathrm{BB1}} \ge \alpha_k^{\mathrm{BB2}}$. 因此, α_k^{BB1} 是比 α_k^{BB2} 更大的步长选择. 在很多实际问题中, 长步长 α_k^{BB1} 的数值表现优于短步长 α_k^{BB2}, 具体可参考 [12–15].

带有 BB 步长的梯度法我们统称为 BB 方法. 在实际中, BB 方法是一类非单调方法, 其针对一般目标函数的收敛性目前还没有得到彻底解决. 但对于特殊问题, 例如严格凸二次问题, 其收敛性分析可见 [10, 16–18]. 虽然 BB 方法针对无约束优化提出, 但对于特殊的约束优化问题, 例如盒子约束二次规划, 我们仍然可以通过投影梯度构造相应的 BB 方法 [19].

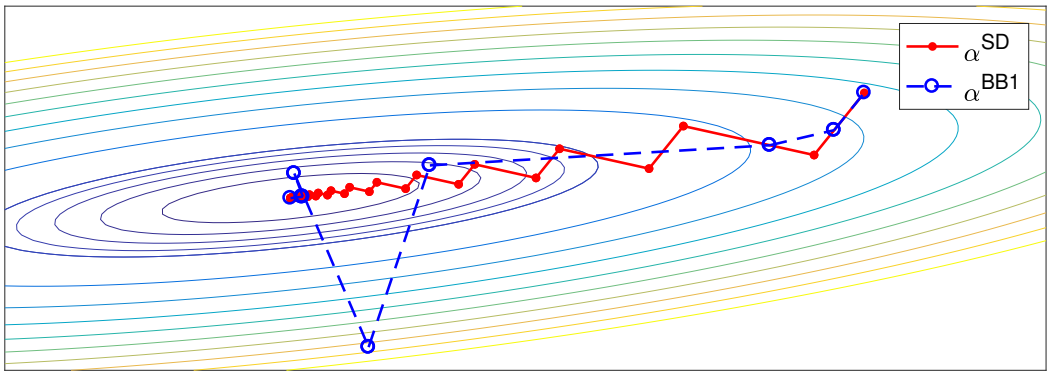


图 1.1 最速下降法与 BB 方法的数值比较 (目标函数: $f(x) = x^2 + 4y^2 - 3xy - 2x$)

Figure 1.1 Numerical comparison on the steepest descent method and BB method (objective function: $f(x) = x^2 + 4y^2 - 3xy - 2x$)

为了展示 BB 方法对最速下降法的巨大改进, 我们设计了一个小实验. 对于目标函数 $f(x) = x^2 + 4y^2 - 3xy - 2x$, 我们分别采用最速下降法与 BB 方法对

问题进行迭代求解. 从图 1.1 中我们可以明显观察到, 最速下降法在靠近最优点附近出现了 “zigzag” 现象, 而选取长 BB 步长 α_k^{BB1} 的梯度法在迭代几步之后就靠近了最优点. 由于 BB 方法在实际中的表现经常优于普通的梯度方法, 因此它得到了许多研究者的关注, 逐渐的发展出了许多基于 BB 步长的不同方法, 例如 [12, 15, 19, 20]. 在本文的算法设计中, 步长的选择是影响算法有效性的一个重要因素. 大量的数值实验显示采用如下的交替 BB 步长方法 [19],

$$\alpha_k^{\mathrm{ABB}} = \begin{cases} \alpha_k^{\mathrm{BB1}}, & k\text{是奇数}, \\ \alpha_k^{\mathrm{BB2}}, & k\text{是偶数}, \end{cases} \tag{1.8}$$

算法具有最好的表现. 因此, 在大部分的算法中, 我们都采取上述的交替 BB 步长.

1.2 黎曼流形优化

黎曼流形优化 (Riemannian optimization) 起源于上世纪八十年代, 近二十年来受到研究者们的广泛关注. 在许多实际应用中, 例如线性代数、信号处理、数据挖掘、统计分析、流形学习等, 问题的变量或目标函数通常满足一些不变性质. 如果我们合理利用这些特殊结构, 则问题可以等价转化为如下的最优化问题,

$$\min_{X\in\mathcal{M}} \ f(X), \tag{1.9}$$

其中, 搜索空间 $\mathcal{M}$ 是一个配备了黎曼结构的微分流形 [21], f 是定义在 $\mathcal{M}$ 上的光滑函数. 我们称问题 (1.9) 为黎曼流形优化问题. 图 1.2[1] 简要介绍了微分流形的定义, 即流形局部与欧式空间中的开邻域光滑同胚. 我们介绍一些黎曼流形优化的具体应用, 感兴趣的读者可以参考以下文献, 非线性特征值问题 [22], 电子结构计算 [23, 24], 低秩矩阵/张量填充 [25, 26], 弹性形状分析 [27], 独立成分分析 [28], 3D 视频稳定 [29], 流形学习 [30–32] 等.

1.2.1 矩阵流形优化问题

通常, 我们所考虑的变量为向量或矩阵, 其中由矩阵构成的流形我们称为矩阵流形. 表 1.1 总结了一些常见的矩阵流形[2]. 其中 $X \succ 0$ 表示矩阵 X 对称正定.

一般来讲, 黎曼流形 $\mathcal{M}$ 可以被包含在 (映射意义下) 一个更大的欧式空间中, 由 1.1 节可知, 黎曼流形优化也可看成是欧式空间中的约束优化. 考虑到黎曼流形的特殊结构, 如果我们将全空间直接选取为流形 $\mathcal{M}$, 则最优化问题 (1.9) 可以

[1] 图片来源: https://en.wikipedia.org/wiki/Differentiable_manifold.

[2] 参考 https://www.manopt.org.

流形名称	数学表示
欧式空间 (复)	$\mathbb{R}^{m\times n}, \mathbb{C}^{m\times n}$
对称矩阵	$\{X \in \mathbb{R}^{n\times n} : X = X^\top\}$
反对称矩阵	$\{X \in \mathbb{R}^{n\times n} : X + X^\top = 0\}$
Centerd 矩阵	$\{X \in \mathbb{R}^{m\times n} : X\mathbf{1}_n = 0_m\}$
单位球	$\{X \in \mathbb{R}^{m\times n} : \|X\|_{\mathrm{F}} = 1\}$
对称单位球	$\{X \in \mathbb{R}^{n\times n} : \|X\|_{\mathrm{F}} = 1, X = X^\top\}$
复单位球	$\{X \in \mathbb{C}^{m\times n} : \|X\|_{\mathrm{F}} = 1\}$
Oblique 流形	$\{X \in \mathbb{R}^{m\times n} : \|X_1\|_{\mathrm{F}} = \cdots = \|X_n\|_{\mathrm{F}} = 1\}$
复 Oblique 流形	$\{X \in \mathbb{C}^{m\times n} : \|X_1\|_{\mathrm{F}} = \cdots = \|X_n\|_{\mathrm{F}} = 1\}$
复圆	$\{z \in \mathbb{C}^n : \|z_1\| = \cdots = \|z_n\| = 1\}$
实 DFT 的相位	$\left\{z \in \mathbb{C}^n : \|z_k\| = 1, z_{1+\mathrm{mod}(k,n)} = \bar{z}_{1+\mathrm{mod}(n-k,n)}, \forall\, k\right\}$
Stiefle 流形	$\{X \in \mathbb{R}^{n\times p} : X^\top X = I\}$
复 Stiefle 流形	$\{X \in \mathbb{C}^{n\times p} : X^* X = I\}$
广义 Stiefle 流形	$\{X \in \mathbb{R}^{n\times p} : X^\top B X = I, B \succ 0\}$
堆积 Stiefle 流形	$\left\{X \in \mathbb{R}^{md\times k} : (XX^\top)_{ii} = I_d\right\}$
Grassmann 流形	$\left\{\mathrm{span}(X) : X \in \mathbb{R}^{n\times p}, X^\top X = I\right\}$
复 Grassmann 流形	$\left\{\mathrm{span}(X) : X \in \mathbb{C}^{n\times p}, X^* X = I\right\}$
广义 Grassmann 流形	$\left\{\mathrm{span}(X) : X \in \mathbb{R}^{n\times p}, X^\top B X = I, B \succ 0\right\}$
旋转群	$\left\{R \in \mathbb{R}^{n\times n} : R^\top R = I, \det(R) = 1\right\}$
特殊欧式群	$\left\{(R,t) \in \mathbb{R}^{n\times n} \times \mathbb{R}^n : R^\top R = I, \det(R) = 1\right\}$
Essential 流形	$SO(3)^2/SO(2)$
定秩流形	$\left\{X \in \mathbb{R}^{m\times n} : \mathrm{rank}(X) = k\right\}$
定秩张量	Tucker 形式的固定多线性秩张量
严格正矩阵	$\left\{X \in \mathbb{R}^{m\times n} : X_{ij} > 0, \forall\, i, j\right\}$
对称正定矩阵	$\{X \in \mathbb{R}^{n\times n} : X = X^\top, X \succ 0\}$
-	$\left\{X \in \mathbb{R}^{n\times n} : X = X^\top \succeq 0, \mathrm{rank}(X) = k\right\}$
-	$\left\{X \in \mathbb{R}^{n\times n} : X = X^\top, X \succ 0, \mathrm{rank}(X) = k, \mathrm{diag}(X) = 1\right\}$
-	$\left\{X \in \mathbb{R}^{n\times n} : X = X^\top, X \succ 0, \mathrm{rank}(X) = k, \mathrm{trace}(X) = 1\right\}$
严格单纯形	$\left\{X \in \mathbb{R}^{m\times n} : X_{ij} > 0, \forall\, i, j \text{ 且 } X\mathbf{1}_n = \mathbf{1}_m\right\}$
-	$\left\{X \in \mathbb{R}^{n\times n} : X_{ij} > 0, \forall\, i, j \text{ 且 } X\mathbf{1}_n = \mathbf{1}_n, X^\top\mathbf{1}_n = \mathbf{1}_n\right\}$
-	$\left\{X \in \mathbb{R}^{n\times n} : X_{ij} > 0, \forall\, i, j \text{ 且 } X\mathbf{1}_n = \mathbf{1}_n, X = X^\top\right\}$
常数矩阵流形	$\{A\}$

表 1.1 矩阵流形集

Table 1.1 Collection of matrix manifold

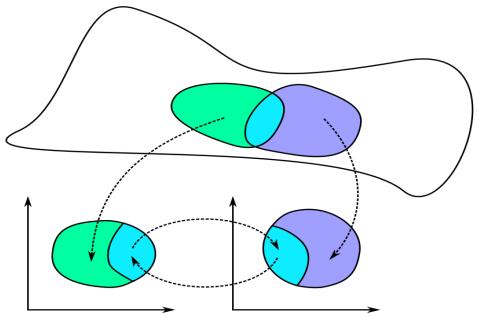


图 1.2 微分流形及其局部坐标卡

Figure 1.2 Differentiable manifold and its charts

看成是黎曼流形 $\mathcal{M}$ 上的无约束优化问题. 由此欧式空间中的经典优化方法大部分都可以推广到黎曼流形优化中 [33].

在给出不同的矩阵流形优化算法之前, 我们首先得确定流形优化问题解所满足的最优性条件. 假设流形 $\mathcal{M}$ 在 $X \in \mathcal{M}$ 点处配备有黎曼度量[3] $\langle \cdot, \cdot \rangle_X$, 记 $\mathcal{T}_X\mathcal{M}$ 为黎曼流形在点 X 处的切空间[3]. 对于黎曼流形 $\mathcal{M}$ 上无约束优化问题 (1.9), 我们定义光滑实值函数 f 在 X 点处的黎曼梯度 $\mathrm{grad} f(X)$, 为切空间 $\mathcal{T}_X\mathcal{M}$ 中满足如下条件的唯一元素,

$$\left\langle \mathrm{grad} f(X), Z \right\rangle_X = \mathcal{D} f(X)[Z], \quad \forall Z \in \mathcal{T}_X\mathcal{M}, \tag{1.10}$$

其中 $\mathcal{D} f(X)[Z]$ 表示 f 定义在流形切空间上的方向导数. 若全空间为欧式空间 $\mathbb{R}^{n\times p}$, 则其定义为

$$\mathcal{D} f(X)[Z] := \mathrm{tr}(\nabla f(X)^\top Z).$$

类似于欧式空间无约束优化问题的一阶最优性条件 (定理 1.1), 对于流形上的无约束优化问题, 我们有如下定理.

定理 1.5 ([33]). *假设 $X^* \in \mathcal{M}$ 是问题 (1.9) 的局部最优点, 则必有*

$$\mathrm{grad} f(X^*) = 0.$$

1.2.2 收缩类方法

在本小节中, 我们介绍一类常用的流形优化方法, 称为收缩类方法. 这类方法也是我们在本文的算法设计与比较中的主要对比算法. 收缩 (retraction) 的概念来源于代数拓扑 [34], 它可以看做是一般的流形测地线的推广. 受到 [35] 的启发, 矩

[3] 关于黎曼度量以及切空间的定义, 读者请参考 [21, 33].

阵流形优化算法得到不断发展. 文献 [36] 和 [37] 分别将收缩的概念应用到动态系统及流形优化中. 在本文中, 收缩的定义来自于专著 [33, Definition 4.1.1]. 下面我们给出收缩映射的具体定义.

定义 1.4 (收缩映射). 如果光滑映射 $\mathcal{R}_X : \mathcal{T}_X\mathcal{M} \to \mathcal{M}$ 满足

(1) $\mathcal{R}_X(0_X) = X$, 其中 0_X 是切空间 $\mathcal{T}_X\mathcal{M}$ 的零元素;

(2) (局部严格条件) $\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{R}_X(tZ)|_{t=0} = Z, \forall Z \in \mathcal{T}_X\mathcal{M}$,

则我们称 $\mathcal{R}_X$ 为收缩映射.

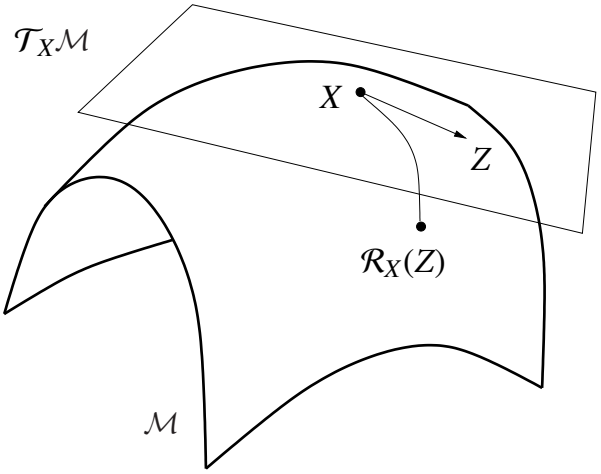


图 1.3 收缩映射

Figure 1.3 Retraction

由上述定义可知, 收缩映射在局部建立了切空间 $\mathcal{T}_X\mathcal{M}$ 与流形 $\mathcal{M}$ 的对应关系. 示意图 1.3[4] 展示了这种类似于 “投影” 的对应关系. 另一方面, 收缩映射的定义使得我们可以在黎曼度量引入的向量内积空间 $\mathcal{T}_X\mathcal{M}$ 中考虑目标函数, 也就是将只定义在流形 $\mathcal{M}$ 上的函数 f “拉回” 到向量空间,

$$\hat{f}_X = f \circ \mathcal{R}_X : \mathcal{T}_X\mathcal{M} \to \mathbb{R}.$$

在 1.1.3 小节我们提到了欧式空间中的线搜索方法,

$$x^{k+1} = x^k + \alpha_k d^k, \tag{1.11}$$

其中 $\alpha_k > 0$ 是搜索步长, d^k 表示搜索方向. 当我们将其应用到矩阵流形优化时, 一个很重要的变化是, 新得到的迭代点 x^{k+1} 在大多数情况下并不继续满足流形

[4]图片来源: [33].

结构, 例如球面和 Stiefel 流形. 另一方面, 搜索方向 d^k 的选取也不再是全空间, 因为根据流形的定义, 其只能限制在当前点的切空间. 因此, 我们必须重新定义流形上的线搜索. 利用收缩映射, 类似于欧式空间的线搜索 (1.11), 我们可以定义流形上的线搜索更新公式 [33],

$$X^{k+1} = \mathcal{R}_{X^k}(\alpha_k D^k),$$

其中 $\alpha_k > 0$ 是搜索步长, $D^k \in \mathcal{T}_{X^k}\mathcal{M}$ 表示搜索方向. 为了得到线搜索方法的全局收敛性, 我们需要对 α_k 和 D_k 作如下假设.

定义 1.5 (**梯度相关序列** [33, Definition 4.2.1])**.** 给定黎曼流形 $\mathcal{M}$ 上的目标函数 f, 切向量序列 $\{D^k\}$, 其中 $D^k \in \mathcal{T}_{X^k}\mathcal{M}$. 如果点列 $\{X^k \in \mathcal{M}\}$ 的任意聚点 X^* 都满足 $\operatorname{grad} f(X^*) \neq 0$, 并且相应的切向量子列 $\{D^k\}_{k\in\mathcal{K}}$ 有界,

$$\overline{\lim}_{k\to\infty, k\in\mathcal{K}} \left\langle \operatorname{grad} f(X^k), D^k \right\rangle_X < 0$$

成立. 则我们称序列 $\{D^k\}$ 梯度相关.

梯度相关序列本质上保证了我们所选取的方向 D^k 是函数值下降方向, 结合如下的非精确线搜索, 我们可以得到流形上的线搜索方法.

定义 1.6 (Armijo **回溯线搜索** [38])**.** 给定黎曼流形 $\mathcal{M}$ 上的目标函数 f 和收缩映射 $\mathcal{R}$. 对于给定的点 $X \in \mathcal{M}$, 切向量 $D \in \mathcal{T}_X\mathcal{M}$, 以及常数 $\bar{\alpha} > 0, \beta, \sigma \in (0,1)$. 我们定义 *Armijo* 点为 $D^{\mathrm{A}} = \alpha^{\mathrm{A}} D = \beta^m \bar{\alpha} D$, 其中 m 是使得不等式

$$f(X) - f(\mathcal{R}_X(\beta^m \bar{\alpha} D)) \geq -\sigma \left\langle \operatorname{grad} f(X), \beta^m \bar{\alpha} D \right\rangle_X$$

成立的最小非负整数. 我们称 α^{A} 为 *Armijo* 步长.

注 1.1. 在 *1.1.3* 小节中, 我们提到了两类步长选择: α^{SD} 和 α^{BB}. 这两种步长有显式的更新方式, 因此不需要进行额外的计算. 在 *Armijo* 回溯线搜索中, 步长 α^{A} 是通过不断的试探步, 比较函数值的下降迭代得到的, 我们称之为非精确线搜索. 在本文中, 我们约定线搜索都是指需要进行试探步计算的非精确线搜索, 而带有 α^{SD} 和 α^{BB} 显式步长的方法我们称为非线搜索方法.

受工作 [39] 的启发, 结合上述定义, 针对流行优化问题 (1.9), 文献 [40] 提出了如下的加速线搜索方法, 也就是流形上的收缩类线搜索方法 (算法 1). 此方法是可行方法, 即每一步迭代点 X^k 都在流形 $\mathcal{M}$ 上, 与之相反的我们称为不可行方法. 在算法 1 中我们发现, 只要给定黎曼流形 $\mathcal{M}$ 的结构, 需要我们去做的就是分

算法 1: 收缩类线搜索方法 [33]

给定黎曼流形 $\mathcal{M}$ 上的收缩映射 $\mathcal{R}$, 常数 $\bar{\alpha} > 0, c, \beta, \sigma \in (0, 1)$.

初始化: $X^0 \in \mathcal{M}$; 令 $k := 0$

while 停机准则不满足 **do**

  选取搜索方向 $D^k \in \mathcal{T}_{X^k}\mathcal{M}$, 使其满足梯度相关条件 (定义 1.5).

  选取 $X^{k+1} \in \mathcal{M}$ 使得不等式

$$f(X^k) - f(X^{k+1}) \geq c\left(f(X^k) - f(\mathcal{R}_{X^k}(\alpha^{\mathrm{A}} D^k))\right)$$

  成立, 其中 α^{A} 是 Armijo 步长 (定义 1.6).

  令 $k := k + 1$.

返回 X^k.

别选取合适的搜索方向以及有效的收缩映射, 不同的收缩映射导致了不同的收缩线搜索方法. 在第二章中, 我们将会详细介绍 Stiefel 流形上收缩映射的选取.

关于此类算法的收敛性分析我们不予讨论, 感兴趣的读者请参考 [33, 40]. 事实上, 在流形优化领域, 还有很多基于收缩映射的算法研究, 例如收缩类信赖域法, 收缩类拟牛顿法等, 感兴趣的读者可以阅读 [33, 41–43] 及其参考文献.

1.3 并行计算

随着信息技术的不断发展, 大数据 (Big Data) 时代已经到来. 一方面, 不同于以往的数据生成方式, 如今数据的产生逐渐由手动低效率模式转变为高效自动化. 与此同时产生的是海量增长的数据, 其规模呈现几何级数的增长. 例如, 在 24 小时内完成 48 小时天气预报[5], 至少需要计算 635 万个网格点, 内存需求大于 1TB, 计算性能[6]要求高达 25Tflops/s. 另一方面, 人类的活动也越来越依赖互联网. 目前, 全世界有超过 30 亿人连入互联网, 如此庞大的用户量会产生大量的数据. 由于大数据往往具有分布式采集和分布式存储的特点, 因此传统的串行算法并不能直接用于处理这类数据. 在这样的背景下, 高性能计算及并行计算孕育而生, 为我们处理大数据时代下的问题带来了机遇.

[5]数据来源: [44].

[6]Tflops/s 指每秒 1 万亿次浮点指令. 参考: https://en.wikipedia.org/wiki/FLOPS.

1.3.1 并行计算简介

简单地讲, 并行计算就是在并行计算机或分布式计算机等高性能计算系统上所作的超级计算. 并行计算主要的研究方向分为: 高性能计算系统 (硬件), 并行算法, 并行程序设计与应用, 并行计算性能评测等. 在本文的算法设计中, 我们主要侧重于并行算法的设计与实现.

通常, 串行算法是指在计算机上由单个进程[7]按一定次序完成的任务. 随着计算机技术的发展, 通过对处理器的多核架构进行设计, 人们利用多个运算核心来提高处理器的性能. 由此, 我们可以将同一个计算任务分配在不同的多个进程按照一定的次序并行计算完成, 这也就是并行算法. 这里, 我们以稠密矩阵向量乘法举例说明. 假设矩阵 $A \in \mathbb{R}^{m\times n}$, 向量 $x \in \mathbb{R}^n$, 接下来我们介绍一种矩阵向量乘法 $b = Ax$ 的并行计算方法. 从图 1.4 我们可以看到, 矩阵向量乘法 Ax 可以并行在多个 CPU 核上同时进行计算, 其中每一个小的任务都是计算向量内积.

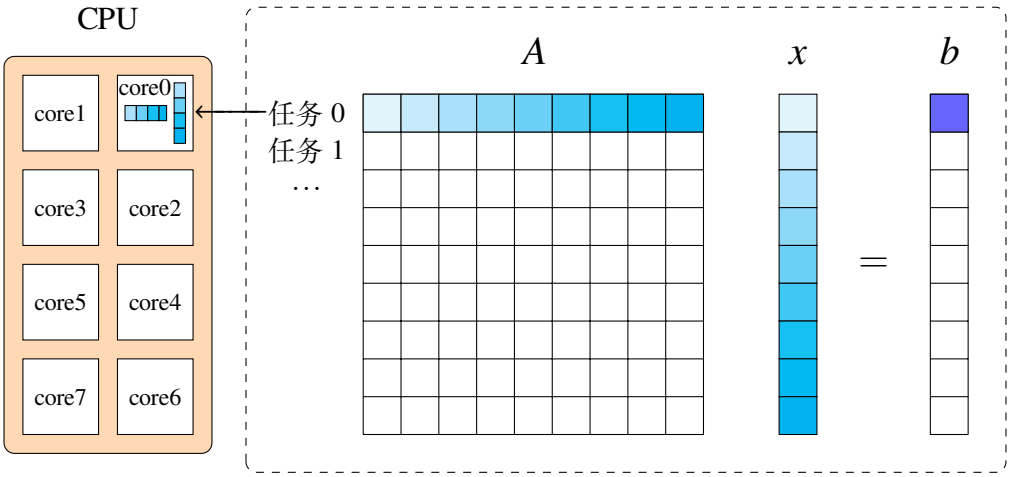


图 1.4 矩阵向量乘法的并行计算

Figure 1.4 Parallel computing for matrix-vector multiplication

关于并行程序的设计, 通常我们需要借助特定的并行编程模型. 它用于并行环境实现的接口或函数库, 或者是对已有程序语言的并行扩展. 常见的并行编程模型包括 MPI (Message Passing Interface)、Pthreads (POSIX threads) 和 OpenMP. 在本文中, 我们不对这些并行编程模型进行详细介绍, 感兴趣的读者请参考 [45–47].

上面我们已经介绍了并行计算所需的算法和程序, 为了衡量和比较不同并行算法之间的区别, 我们需要引入如下的概念. 首先, 比较不同的算法, 我们可以很自然地通过程序的运行时间来衡量算法的优劣. 对于串行算法而言, CPU 运行

[7]进程: 计算机任务的划分单位.

时间即为程序运行的时间. 然而对于并行算法而言, 其程序同时运行在多个 CPU 核上, 如果直接将每个 CPU 核的时间相加, 这显然不能反映程序的真实效率. 因此, 我们需要考虑程序在现实世界中真实运行的时间, 也就是墙上时间 (wall-clock time), 它不单独依赖于每个 CPU 核, 而只与整个程序的运行时间有关. 除此之外, 同一个并行程序在不同 CPU 核下运行的时间也可能不同. 因此, 我们引入如下的并行加速比 (parallel speedup factor) 和并行效率 (parallel efficiency) 的概念 [44].

$$\begin{aligned} \text{speedup-factor}(m) &= \frac{m_0 T_{m_0}}{T_m}, \\ \text{efficiency}(m) &= \frac{m_0 T_{m_0}}{m T_m}. \end{aligned}$$

在上式中, m 表示程序运行的 CPU 核数, T_m 表示程序在 m 核下运行的墙上时间. 通常, 我们取 $m_0 = 1$, 也就是单核运行. 一般而言, 我们有 $\text{speedup-factor}(m) < m, \text{efficiency}(m) < 1$. 如果并行加速比越接近 m, 并行效率越靠近 1, 则说明程序的并行可扩展性越好.

1.3.2 分布式/并行优化算法

通常来讲, 分布式/并行优化计算主要分为两大类. 第一类做法是数值代数层面的并行, 也就是将已有的高效串行算法进行并行化. 其中基础的线性代数操作 (Basic Linear Algebra Subprograms, 简称为 BLAS[8]), 例如矩阵向量乘法 (BLAS2), 矩阵矩阵乘法 (BLAS3) 等都可被高效的并行计算. 这类算法的最大优点是不需要对串行算法本身进行变动, 而这恰好也限制了其进一步的发展. 另一类计算方法是优化模型或者优化算法层面的并行, 也就是将一个大规模的问题分解为若干个小规模容易计算的问题, 这些问题相互独立或不独立. 根据子问题的求解次序, 我们还可以将并行算法分为异步并行或者同步并行. 示意图 1.5 展示了分布式/并行优化算法设计的两种不同思路.

随着数据规模的不断增长, 原有的一些优化模型已不足以完整地描述问题, 与此同时原有的优化算法也面临着相同的挑战. 在优化领域, FISTA [48], EigPen [49] 等算法根据第一种思路已经并行化实现. 而第二种思路则是目前并行优化算法领域的热点问题之一. 该类方法主要包括并行块坐标下降法 (PBCD) [50]、交替方向乘子法 (ADMM) [51]、并行子空间校正法 (PSC) [52] 等, 这些算法各有利弊. 关于其他可分布式/并行优化算法, 感兴趣的读者请参考 [53–57] . 接下来, 我们详细介绍在本文算法设计与比较中用到的块坐标下降方法.

[8] 请参考 http://www.netlib.org/blas/.

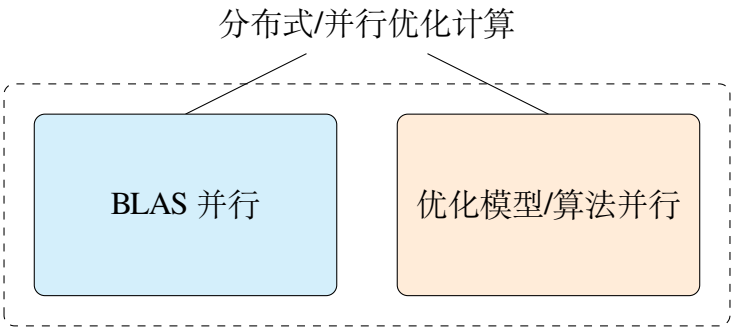


图 1.5 分布式/并行优化计算

Figure 1.5 Distributed/Parallel optimization

块坐标下降法 (BCD) 和 ADMM 方法是大数据时代最受关注的两类算法. 其中 BCD 方法的主要思想是将变量进行拆分, 得到若干块小变量. 在算法迭代过程中, 每次只选取一块进行更新, 其余块作为常量固定, 由此原问题的规模得到了显著降低. 块坐标下降的思想最早可以追溯到求解线性方程组的 Jacob 迭代和 Gauss-Seidel 迭代 [58]. 特别地, 我们考虑如下的优化问题,

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & f(x_1, x_2, \ldots, x_p) \\ \text{s.t.} \quad & \textstyle\sum_{i=1}^{p} A_i x_i = b, \end{aligned}$$

其中 $x = [x_1, x_2, \ldots, x_p]$, $x_i \in \mathbb{R}^{n_i}$, $\sum_{i=1}^{p} n_i = n$. 应用 BCD 方法求解此问题, 我们得到算法 2.

算法 2: 块坐标下降法 (BCD)

1 初始化: $x^0 \in \mathbb{R}^n$; 令 $k := 0$

2 **while** 停机准则不满足 **do**

3 依据某种规则求解如下的 p 个 BCD 子问题,

$$\begin{aligned} x_i^{k+1} := \arg\min_{y\in\mathbb{R}^{n_i}} \quad & f_i(y) := f(x_1^k, \ldots, x_{i-1}^k, y, x_{i+1}^k, x_p^k) \\ \text{s.t.} \quad & A_i y = b - \textstyle\sum_{j\neq i}^{p} A_j x_j^k. \end{aligned}$$

令 $k := k + 1$.

4 返回 $x^k = [x_1^k, x_2^k, \ldots, x_p^k]$.

注 1.2. *值得说明的是, 分布式/并行计算并不是 BCD 方法最大的特征. 事实上, BCD 方法有很多变种. 一方面, 如果我们考虑块变量的更新, 在算法 2 中, 如果我们分别独立计算 p 个 BCD 子问题, 则其迭代本质上是 Jacob 迭代, 称为并行的*

BCD 方法 (PBCD). 如果我们选取 $f_i(y) := f(x_1^{k+1}, \ldots, x_{i-1}^{k+1}, y, x_{i+1}^k, x_p^k)$, 即每个块变量计算后及时更新, 则迭代为 *Gauss-Seidel* 型迭代. 另一方面, 如果考虑 p 个子问题的求解顺序, 则我们有如下的不同选取方式:

- 随机选取 *(有放回的):* 每次从集合 $\{1, 2, \ldots, p\}$ 中随机选取一个数, 以此为指标, 求解相应的子问题;
- 随机选取 *(无放回的):* 每次生成集合 $\{1, 2, \ldots, p\}$ 的随机排列, 以此顺序求解相应的子问题;
- 贪婪策略: 根据子问题的求解程度, 每次对子问题 $\{1, 2, \ldots, p\}$ 中最不精确的那个进行求解;
- 按序循环更新: 按集合 $\{1, 2, \ldots, p\}$ 的顺序循环求解子问题.

事实上, 每一种不同的选择都会导致不同的算法, 其对应的可并行程度也有所不同.

在 BCD 方法中, 我们可以考虑的并行策略是 p 个子问题分别独立求解, 因此可并行的 BCD 方法 PBCD 是模型和算法上的并行. 目前, 关于 BCD 方法研究的热点主要包括两个方面: 其一, 考虑利用 Nesterov 加速技巧 [59–61] 设计加速的 BCD 方法. 其二, 随机更新选取策略下 BCD 方法的理论收敛性质 [62–65]. 关于这两点, 本文不予介绍, 感兴趣的读者请阅读参考文献.

1.4 本文主要内容

本文主要从理论、算法与应用三个角度, 系统的研究了正交约束优化问题. 余下部分的内容安排如下.

在第 2 章中, 我们系统地研究了正交约束优化问题. 从黎曼流形优化和欧式空间约束优化两个角度, 我们分别推导了问题的最优性条件. 其中, 通过分析一大类黎曼度量下的黎曼梯度, 我们得到了问题各种描述下一阶最优性条件的对应关系. 此外, 我们还证明了在任意一阶稳定点处, 正交约束优化问题的 Lagrange 乘子具有显式表达式. 这些性质的刻画启发了本文的算法设计.

在第 3 章中, 针对一大类正交约束优化问题, 我们提出了一类非收缩算法框架. 其主要包含两个步骤, 分别是函数值下降步和乘子校正步. 不同于以往的经典算法, 在我们的算法框架中, 函数值下降步采用标准的欧式负梯度方向, 而不是 Stiefel 流形的切空间方向. 另一方面, 我们构造的乘子校正步进一步使函数值下降, 同时也保证了乘子的对称性. 基于此算法框架, 我们提出了两大类算法. 第一

类是梯度下降方法, 其中包括梯度反射法和梯度投影法. 第二类采用以列为块的块坐标下降方法, 同时我们也提出了一个新的方法用于非精确求解对应的块坐标下降子问题. 进一步, 我们证明了算法的全局收敛性. 数值实验表明我们的算法框架具有很大的潜力.

在第 4 章中, 我们将乘子校正步推广到一般的 Stiefel 流形收缩类算法, 得到了子空间加速的收缩类算法. 通过利用乘子校正步的子空间最优性质, 我们设计了一类两阶段的子空间加速算法. 第一阶段是函数值下降法, 第二阶段是子空间加速步. 将 Stiefel 流形的收缩类线搜索算法应用于第一阶段, 我们证明了加速算法的全局收敛性及局部线性收敛速度. 数值实验展示了加速技术的有效性.

在第 5 章中, 针对一般的正交约束优化问题, 我们提出了基于增广 Lagrange 函数的并行算法. 考虑到正交化过程的低可扩展性, 我们采用不可行方法并利用增广 Lagrange 罚函数. 不同于经典的增广 Lagrange 函数法, 在我们的算法中, 原始变量的更新通过极小化增广 Lagrange 函数的邻近点线性化逼近得到. 同时, Lagrange 乘子由其在一阶稳定点处的显式解更新得到. 由此, 算法的主要步骤都可以很自然地进行矩阵计算层面的并行化. 进一步, 我们分析了算法的全局收敛性, 最坏情况下的复杂度以及局部收敛速度. 此外, 为了减弱算法对罚参数的敏感性, 我们提出了改进的可并行列极小化算法. 串行的数值实验说明了乘子的新更新方式显著加速了算法的收敛速度, 并且数值表现与已有的可行方法不相上下. 并行环境下的数值实验验证了我们的算法具有较高的可扩展性.

在第 6 章中, 我们将乘子校正算法和基于增广 Lagrange 函数的并行算法应用于电子结构计算. 我们考虑了电子结构计算中的 Kohn-Sham 密度泛函理论, 其模型通常表述为一个带有正交约束的优化问题. 在串行环境下, 我们测试了 18 个不同的分子结构, 数值实验显示了我们的新算法优于已有的经典算法. 在并行环境下, 我们测试了简化的 Kohn-Sham 总能量极小化问题, 数值结果显示我们提出的并行算法具有较高的可扩展性.

在最后一章中, 我们对全文作以总结并给出对未来工作的展望.

第 2 章　正交约束优化问题

正交约束优化问题是指变量满足正交性的矩阵优化问题,

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & f(X) \\ \text{s.t.} \quad & X^\top X = I_p, \end{aligned} \tag{2.1}$$

其中函数 f 可微, $p \le n$. 特别地, 问题 (2.1) 的可行域也被称为 Stiefel 流形 [66], 记做

$$\mathcal{S}_{n,p} := \left\{ X \in \mathbb{R}^{n\times p} \mid X^\top X = I \right\}.$$

由此, 正交约束优化问题既可以看作欧式空间 $\mathbb{R}^{n\times p}$ 中的约束优化问题, 也可以看成是 Stiefel 流形上的矩阵流形优化问题,

$$\min_{X\in\mathcal{S}_{n,p}} \quad f(X). \tag{2.2}$$

在图 2.1 中, 我们考虑最简单的正交约束, 即 $p = 1$ 时的球面约束. 我们观察到二

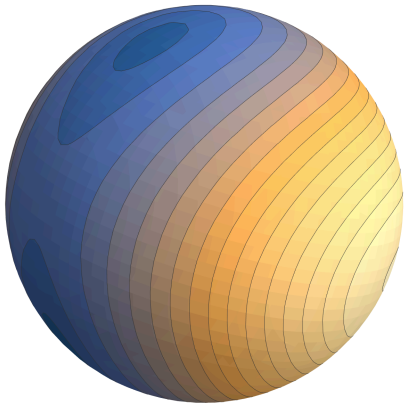


图 2.1 目标函数 $f(x,y,z) = x^2 + 5y^2 - 3z^2 + 5x$ 在球面上的等值线图

Figure 2.1 Contour of function $f(x,y,z) = x^2 + 5y^2 - 3z^2 + 5x$ on a sphere

次函数 $f(x,y,z) = x^2+5y^2-3z^2+5x$ 在球面约束下, 有两个局部极小值点, 一个局部极大值点. 一方面, 由于正交约束的非凸性以及正交化过程的高计算代价, 问题 (2.1) 在实际中并不容易求解. 另一方面, 当目标函数 f 取特定形式时, 问题 (2.1) 是 NP-难的, 例如最大割问题 [67], 干扰极小化问题 [68] 以及 Bose–Einstein 凝聚 [69]. 因此, 面对这些困难和挑战, 问题 (2.1) 被越来越多的学者所关注. 在本章中, 我们主要介绍正交约束优化问题的应用背景, 一阶最优性条件的刻画, Stiefel 流形的结构和性质, 以及一些已有的优化算法.

2.1 问题背景及应用

正交约束优化问题在科学工程计算以及数据科学等领域都有着广泛的应用, 例如电子结构计算 [70–74], 变分问题的压缩模型 [75, 76], Bose-Einstein 凝聚 [69], 线性特征值问题 [49, 77, 78], 非线性特征值问题 [79–81], 奇异值分解 [58, 82], 低秩相关矩阵 [83–85], 稀疏主成分分析 [86, 87], 正交 Procrustes 问题 [88, 89], 异构四次函数和的极小化 [90, 91], 干扰极小化问题 [68, 92] 以及联合对角化问题 [28, 93] 等.

接下来, 我们具体介绍几类应用问题.

1. 特征值问题

求解实对称矩阵 $A \in \mathbb{R}^{n\times n}$ 的 p 个最小特征值及其特征向量, 可以用如下的迹极小化问题 [94, 95] 描述,

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & \mathrm{tr}(X^\top AX) \\ \text{s.t.} \quad & X^\top X = I_p. \end{aligned} \tag{2.3}$$

上述问题的最优解 X^* 张成了 A 的不变特征子空间, 其正交基对应的特征值为矩阵 A 的 p 个最小特征值. 具体来说, 若 A 的特征值为 $\lambda_1 \le \lambda_2 \le \cdots \le \lambda_n$, 则我们有

$$AX^* = X^*\Lambda^*,$$

其中 $\Lambda^* = \mathrm{Diag}(\lambda_1, \lambda_2, \ldots, \lambda_p)$.

2. 广义特征值问题

给定矩阵对 (A, B), 其中 $A, B \in \mathbb{R}^{n\times n}$ 且 $B \succ 0$, 求解矩阵对 (A, B) 的 p 个最小广义特征值的问题

$$Av = \lambda Bv,$$

可以等价的描述为如下的广义 Rayleigh 商极小化问题,

$$\begin{aligned} \min_{Y\in\mathbb{R}^{n\times p}} \quad & \mathrm{tr}(Y^\top AY(Y^\top BY)^{-1}) \\ \text{s.t.} \quad & Y^\top BY = I_p. \end{aligned}$$

其等价性可参考文献 [33, Proposition 2.1.1].

3. 奇异值分解

给定矩阵 $A \in \mathbb{R}^{m\times n}(m \ge n)$, 记 A 的奇异值为 $0 \le \sigma_1 \le \sigma_2 \le \cdots \le \sigma_n$. 由奇异值分解的定义 [58, 96] 以及 Eckart-Young-Mirsky 定理 ([97], [58, Theorem 2.4.8])

可知, 求解矩阵 A 最小 p 个奇异值对应的奇异向量, 可由如下的最佳秩 p 逼近得到,

$$A^p \triangleq \sum_{i=1}^{p} \sigma_i u_i v_i^\top = \underset{\mathrm{rank}(W)\le p}{\arg\min} \|A - W\|_{\mathrm{F}}^2 . \tag{2.4}$$

等价的, 也就是如下的截断奇异值分解,

$$A^p = U\Sigma V^\top,$$

其中 $U \in \mathcal{S}_{m,p}$, $V \in \mathcal{S}_{n,p}$ 且 $\Sigma = \mathrm{Diag}(\sigma_1, \ldots, \sigma_p)$. 接下来, 我们介绍三种不同的求解截断奇异值分解的模型, 它们都是正交约束优化问题, 并且相互等价.

3a) 由定义, 此问题可以等价的转化为计算矩阵 $AA^\top$ 或 $A^\top A$ 的特征值, 也就是

$$\begin{aligned} \min_{U\in\mathbb{R}^{m\times p}} \quad & \mathrm{tr}(U^\top AA^\top U) \\ \text{s.t.} \quad & U^\top U = I_p, \end{aligned} \quad \text{或} \quad \begin{aligned} \min_{V\in\mathbb{R}^{n\times p}} \quad & \mathrm{tr}(V^\top A^\top AV) \\ \text{s.t.} \quad & V^\top V = I_p. \end{aligned}$$

上述问题的解分别为矩阵 A 最小 p 个奇异值对应的左奇异向量矩阵 U 或右奇异向量矩阵 V, 由此我们得到矩阵 A 的截断奇异值分解.

3b) 由 (2.4) 式, 我们可以得到截断奇异值分解的另一种等价模型,

$$\begin{aligned} \min_{X\in\mathbb{R}^{m\times p}, Y\in\mathbb{R}^{n\times p}} \quad & \|A - XY^\top\|_{\mathrm{F}}^2 \\ \text{s.t.} \quad & X^\top X = I_p. \end{aligned}$$

3c) 基于 Stiefel 流形的乘积空间, 文献 [98] 提出了截断奇异值分解的另一种等价模型,

$$\begin{aligned} \min_{U\in\mathbb{R}^{m\times p}, V\in\mathbb{R}^{n\times p}} \quad & \mathrm{tr}(U^\top AVN) \\ \text{s.t.} \quad & U^\top U = I_p, \\ & V^\top V = I_p, \end{aligned}$$

其中 $N = \mathrm{Diag}(\mu_1, \mu_2, \ldots, \mu_p)$, 并且 $\mu_p > \mu_{p-1} > \cdots > \mu_1 > 0$.

4. *正交 Procrustes 问题*

给定矩阵 $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times p}$, 正交 Procrustes 问题 [88, 89] 实际上是 Stiefel 流形上的最小二乘逼近问题,

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & \|AX - B\|_{\mathrm{F}}^2 \\ \text{s.t.} \quad & X^\top X = I_p. \end{aligned}$$

5. 二次指派问题

二次指派问题是组合优化中的经典问题, 刻画了不同设施在距离和权重极小化下的选址问题. 文献 [99] 等价的考虑了 Stiefel 流形上带有非负约束的二次指派问题,

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & \mathrm{tr}\left(A^\top (X\odot X) B (X\odot X)^\top\right) \\ \text{s.t.} \quad & X^\top X = I_p, \\ & X \geq 0, \end{aligned}$$

其中 $A\in\mathbb{R}^{n\times n}$ 表示距离矩阵, $B\in\mathbb{R}^{n\times n}$ 表示权重矩阵, $\odot$ 表示 Hadamard 积, $X\geq 0$ 表示矩阵 X 的所有元素非负.

6. *Bose-Einstein* 凝聚态问题

Bose-Einstein 凝聚 [100] 是玻色子原子在冷却到接近绝对零度所呈现出的一种气态的、超流性的物质状态[1]. 其物理模型是球面约束下的能量泛函极小化问题,

$$\phi_g = \arg\min_{\phi\in\mathcal{S}} \quad E(\phi),$$

其中能量泛函 $E(\phi)$ 和球面约束 $\mathcal{S}$ 分别定义为

$$\begin{aligned} E(\phi) &= \int_{\mathbb{R}^d} \left[\frac{1}{2}|\nabla\phi(\mathbf{x})|^2 + V(\mathbf{x})|\phi(\mathbf{x})|^2 + \frac{\beta}{2}|\phi(\mathbf{x})|^4 - \Omega\bar{\phi}(\mathbf{x})L_z\phi(\mathbf{x})\right] d\mathbf{x}, \\ S &= \left\{\phi \mid E(\phi)<\infty, \int_{\mathbb{R}^d} |\phi(\mathbf{x})|^2 d\mathbf{x} = 1\right\}. \end{aligned}$$

我们将上述问题进行离散化处理 [69], 得到如下的离散模型,

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & \tfrac{1}{2}x^\top A x + \tfrac{\beta}{2}\sum_{i=1}^n x_i^4 \\ \text{s.t.} \quad & \|x\|_2 = 1. \end{aligned}$$

此问题本质上是一个四次函数在球面上的极小化问题, 也正是正交约束优化问题退化到 $p=1$ 的特殊情形.

7. 图的稳定数

给定图 $\mathcal{G}=(V,E)$, V 是定点集合, E 为边构成的集合. V 的最大稳定子集 (互不相交的顶点) 的维数定义为稳定数 $\zeta(\mathcal{G})$. 则由文献 [101] 可知,

$$\zeta(\mathcal{G})^{-1} = \min_{\|x\|_2=1} \sum_{i=1}^n x_i^4 + 2\sum_{(i,j)\in E} x_i^2 x_j^2.$$

这也是 $p=1$ 时的正交约束优化问题.

[1]参考 https://en.wikipedia.org/wiki/Bose%E2%80%93Einstein_condensate.

8. 主成分分析

在数据科学中, 主成分分析 [102] 是一种统计方法, 其通过正交变换将一组可能存在相关性的变量转换成一组线性不相关的变量, 转换后的这组变量称为主成分. 通常, 这个过程也被称为数据降维. 给定观测矩阵 $A \in \mathbb{R}^{n\times m}$, 我们需要在 m 个特征中筛选出 p 个主要的特征. 文献 [30, 103, 104] 引入了如下的正交约束优化模型,

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & -\tfrac{1}{m}\mathrm{tr}\left(X^\top (A-\bar{A})(A-\bar{A})^\top X\right) \\ \text{s.t.} \quad & X^\top X = I_p, \end{aligned}$$

其中 $\bar{A} = \frac{1}{m}\sum_{i=1}^{m} A_i \mathbf{1}^\top$, $\mathbf{1} = (1,\ldots,1)^\top \in \mathbb{R}^n$. 本质上, 主成分分析等价于问题 3 中的截断奇异值分解.

9. *Kohn-Sham* 总能量极小化问题

Kohn-Sham 密度泛函理论 (KSDFT) [105] 是材料科学中的一个重要问题. 其最后一步是在正交约束下极小化离散的 Kohn-Sham 总能量泛函 [70],

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & E(X) \\ \text{s.t.} \quad & X^\top X = I_p. \end{aligned} \tag{2.5}$$

在第 6 章中, 我们将会详细介绍此问题.

10. 薛定谔方程的压缩模型

压缩模型 [75] 用来刻画孤立粒子薛定谔方程的空间解. 文献 [75] 利用变分法得到了如下的离散 ℓ_1 正则化优化模型,

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & \mathrm{tr}(X^\top H X) + \tfrac{1}{\kappa}|X|_1 \\ \text{s.t.} \quad & X^\top X = I_p, \end{aligned}$$

这里 $\kappa > 0$ 是一个预设的参数, $H \in \mathbb{SR}^{n\times n}$ 表示电子结构的离散 Hamilton 算子, $|X|_1 := \sum_{i=1,j=1}^{i=n,j=p} |X_{ij}|$.

2.2 最优性条件

在本节, 我们从欧式空间中的正交约束优化问题 (2.1) 以及 Stiefel 流形上的无约束优化问题 (2.2) 两个角度出发, 分别引入问题的最优性条件和基本性质. 为了刻画最优性条件, 首先我们给出约束集 Stiefel 流形的黎曼结构.

2.2.1 Stiefel 流形

由于许多算法的设计依赖于 Stiefel 流形的结构, 因此在本小节我们介绍一些 Stiefel 流形的基本性质, 其中部分结果来自于专著 [33], 具体证明由本文作者补

充. 此外, 我们还首次研究了一大类黎曼度量下的黎曼梯度.

性质 2.1. *Stiefel 流形 $\mathcal{S}_{n,p}$ 满足如下性质:*

1) $\mathcal{S}_{n,p}$ 是有界闭集.

2) $\mathcal{S}_{n,p}$ 是欧式空间 $\mathbb{R}^{n\times p}$ 的黎曼嵌入子流形.

3) 当 $p=1$ 时, $\mathcal{S}_{n,p}$ 退化为 $\mathbb{R}^n$ 中的单位球 $\mathcal{S}^{n-1}=\{x\in\mathbb{R}^n \mid \|x\|_2=1\}$.

4) 当 $p=n$ 时, $\mathcal{S}_{n,p}$ 等于正交群 $\mathcal{O}_n=\{X\in\mathbb{R}^{n\times n} \mid X^\top X=I_n\}$.

5) $\dim(\mathcal{S}_{n,p})=np-\frac{1}{2}p(p+1)$.

证明. 由 Stiefel 流形 $\mathcal{S}_{n,p}$ 的定义, 性质 1)-4) 立即可得. 下面我们考虑性质 5).

首先我们构造映射

$$\begin{aligned} F:\mathbb{R}^{n\times p} &\longrightarrow \mathbb{SR}^{p\times p} \\ X &\longmapsto X^\top X-I_p. \end{aligned}$$

其中 $\mathbb{SR}^{p\times p}$ 表示 p 阶对称矩阵全体, 并且 $\dim(\mathbb{SR}^{p\times p})=p(p+1)/2$. 由 F 的定义, 我们得到

$$\mathcal{S}_{n,p}=F^{-1}(0). \tag{2.6}$$

下面我们证明映射 F 的秩为 $p(p+1)/2$, 也就是对于任意的 $\bar{Z}\in\mathbb{SR}^{p\times p}$, 存在 $Z\in\mathbb{R}^{n\times p}$ 使得

$$\mathcal{D}F(X)[Z]=\bar{Z},\quad \forall X\in\mathcal{S}_{n,p},$$

即 $\mathcal{D}F(X)$ 对于任意的 $X\in\mathcal{S}_{n,p}$ 都是满射. 由定义可知

$$\mathcal{D}F(X)[Z]=X^\top Z+Z^\top X,\quad \forall X\in\mathcal{S}_{n,p},\forall Z\in\mathbb{R}^{n\times p}.$$

取 $Z=\frac{1}{2}X\bar{Z}$, 代入上式我们得到,

$$\mathcal{D}F(X)[\frac{1}{2}X\bar{Z}]=\frac{1}{2}X^\top X\bar{Z}+\frac{1}{2}\bar{Z}^\top X^\top X=\bar{Z},\quad \forall X\in\mathcal{S}_{n,p}.$$

由此我们得到 F 的秩等于 $\dim(\mathbb{SR}^{p\times p})$, 即 $p(p+1)/2$.

结合 (2.6) 式以及微分流形中的淹没定理 [33, Proposition 3.3.3], 我们有

$$\dim(\mathcal{S}_{n,p})=\dim(F^{-1}(0))=np-\frac{1}{2}p(p+1).$$

□

接下来, 我们给出 Stiefel 流形切空间 $\mathcal{T}_X\mathcal{S}_{n,p}$ 的具体表达形式.

性质 2.2 (切空间). *Stiefel 流形在 $X \in \mathcal{S}_{n,p}$ 点处的切空间 $\mathcal{T}_X\mathcal{S}_{n,p}$ 有如下形式,*

$$\begin{aligned}\mathcal{T}_X\mathcal{S}_{n,p} &= \{Z \in \mathbb{R}^{n\times p} : X^\top Z + Z^\top X = 0\} && (2.7)\\ &= \{XW + X_\perp K : W^\top + W = 0, W \in \mathbb{R}^{p\times p}, K \in \mathbb{R}^{(n-p)\times p}\} && (2.8)\\ &= \{AX : A^\top + A = 0, A \in \mathbb{R}^{n\times n}\}, && (2.9)\end{aligned}$$

其中 $X_\perp \in \mathbb{R}^{n\times(n-p)}$ 表示子空间 $\mathrm{span}(X)$ 的正交补空间, 也就是 $XX^\top + X_\perp X_\perp^\top = I$, $W \in \mathbb{R}^{p\times p}$, $A \in \mathbb{R}^{n\times n}$ 都是反对称矩阵.

证明. 首先我们证明 (2.7) 式成立. 构造映射

$$\begin{aligned}F : \mathbb{R}^{n\times p} &\longrightarrow \mathbb{SR}^{p\times p}\\ X &\longmapsto X^\top X.\end{aligned}$$

类似于性质 2.1 中 5) 的证明, 我们有 $\mathcal{S}_{n,p} = F^{-1}(I_p)$, 并且映射 F 的秩为 $p(p+1)/2$. 由 [33, (3.19) 式], 我们得到

$$\mathcal{T}_X\mathcal{S}_{n,p} = \ker(\mathcal{D}F(X)) = \{Z \in \mathbb{R}^{n\times p} : X^\top Z + Z^\top X = 0\}.$$

除此之外, 我们还得到

$$\dim(\mathcal{T}_X\mathcal{S}_{n,p}) = \dim\left(\ker(\mathcal{D}F(X))\right) = np - \frac{p(p+1)}{2}. \tag{2.10}$$

定义 Stiefel 流形上的曲线 $X(t) : t \longmapsto X(t)$, 由 $X(t)^\top X(t) = I$, 我们有

$$\dot{X}(t)^\top X(t) + X(t)^\top \dot{X} = 0. \tag{2.11}$$

又 $X(t) \in \mathcal{S}_{n,p}$, 则 $[X(t)\ X_\perp(t)] \in \mathbb{R}^{n\times n}$ 满秩, 故对于 $\dot{X}(t) \in \mathbb{R}^{n\times n}$, 我们可以令

$$\dot{X}(t) = X(t)W(t) + X_\perp(t)K(t),$$

其中 $W(t) \in \mathbb{R}^{p\times p}$, $K \in \mathbb{R}^{(n-p)\times p}$. 将上式代入 (2.11) 式, 得到

$$W^\top(t) + W(t) = 0,$$

从而 (2.8) 式成立.

假设 $A \in \mathbb{R}^{n\times n}$ 满足 $A^\top + A = 0$, 显然我们有

$$\{AX : A^\top + A = 0, A \in \mathbb{R}^{n\times n}\} \subset \{Z \in \mathbb{R}^{n\times p} : X^\top Z + Z^\top X = 0\} = \mathcal{T}_X\mathcal{S}_{n,p}. \tag{2.12}$$

另一方面, 对于 $X \in \mathcal{S}_{n,p}$, 存在正交矩阵 $Q \in \mathbb{R}^{n\times n}$ 使得 $X = Q[I_p\ 0]^\top$. 记 $B = Q^\top AQ = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}$, 则

$$\begin{aligned} & \dim\{AX : A^\top + A = 0, A \in \mathbb{R}^{n\times n}\} \\ = \ & \dim\{Q^\top AX : A^\top + A = 0, A \in \mathbb{R}^{n\times n}\} \\ = \ & \dim\{B[I_p\ 0]^\top : B^\top + B = 0, B \in \mathbb{R}^{n\times n}\} \\ = \ & \dim\{[B_{11}^\top\ B_{21}^\top]^\top : B_{11}^\top + B_{11} = 0, B_{11} \in \mathbb{R}^{p\times p}, B_{21} \in \mathbb{R}^{(n-p)\times p}\} \\ = \ & p^2 - \frac{p(p+1)}{2} + (n-p)p = np - \frac{p(p+1)}{2}, \end{aligned}$$

其中最后一个等式利用了 $\dim(\mathbb{SR}^{p\times p}) = p(p+1)/2$. 结合上式, (2.10) 式和 (2.12) 式, 我们可知 (2.9) 成立. □

定义了 Stiefel 流形的切空间之后, 我们就可以装备切空间的黎曼度量. 在 $\mathbb{R}^{n\times p}$ 中, 对于 $Z_1, Z_2 \in \mathbb{R}^{n\times p}$, 经典的欧式内积定义为

$$\langle Z_1, Z_2\rangle := \mathrm{tr}(Z_1^\top Z_2).$$

如果我们对 Stiefel 流形配备上述诱导度量作为其黎曼度量, 则我们得到黎曼流形 $\mathcal{S}_{n,p}$. 为方便起见, 我们简称为流形 $\mathcal{S}_{n,p}$. 由此, 我们可以给出 Stiefel 流形的法空间性质.

性质 2.3 (法空间). *Stiefel 流形在 $X \in \mathcal{S}_{n,p}$ 点处的法空间有如下形式,*

$$(\mathcal{T}_X\mathcal{S}_{n,p})^\perp = \left\{XS : S \in \mathbb{SR}^{p\times p}\right\}.$$

证明. 由性质 2.2 中的 (2.8) 式, 并利用性质 $\mathcal{T}_X\mathcal{S}_{n,p} \oplus (\mathcal{T}_X\mathcal{S}_{n,p})^\perp = \mathbb{R}^{n\times p}$ ($\oplus$ 表示直和), 显然我们有

$$(\mathcal{T}_X\mathcal{S}_{n,p})^\perp = \left\{XN : N \in \mathbb{R}^{p\times p}\right\}.$$

由定义可知, 对于任意的 $XN \in (\mathcal{T}_X\mathcal{S}_{n,p})^\perp$,

$$\langle Z, XN\rangle = 0, \quad \forall Z \in \mathcal{T}_X\mathcal{S}_{n,p}.$$

根据 (2.8) 式, 我们令 $Z = XW_Z + X_\perp K_Z$, 其中 W_Z 是反对称矩阵, 将 Z 代入上式得到

$$\langle Z, XN\rangle = \mathrm{tr}(W_Z^\top N) = 0.$$

因为 $\mathrm{tr}(W_Z^\top S) = \langle W_Z, S\rangle = 0$ 对于任意的 $S \in \mathbb{SR}^{p\times p}$ 都成立, 又由 W_Z 的任意性, 故 $N \in \mathbb{SR}^{p\times p}$. 由此性质成立. □

接下来, 我们给出闭凸集上正交投影的定义, 其定义与黎曼度量 $\langle \cdot, \cdot \rangle_X$ 有关, 也就是不同的度量诱导了不同的投影算子.

定义 2.1. 给定闭凸集 $\mathcal{A} \subset \mathbb{R}^{n\times p}$, 我们定义点 $Y \in \mathbb{R}^{n\times p}$ 到 $\mathcal{A}$ 的正交投影为

$$\mathcal{P}_{\mathcal{A}}(Y) = \operatorname*{arg\,min}_{X\in\mathcal{A}} \|Y - X\|_{\mathrm{F}}.$$

由性质 2.2 和 2.3, 我们可以很自然定义任意 $Y \in \mathbb{R}^{n\times p}$ 在 Stiefel 流形的切空间和法空间的正交投影.

性质 2.4. 给定任意的 $Y \in \mathbb{R}^{n\times p}$, 我们定义其在 *Stiefel* 流形切空间和法空间的正交投影为

$$\begin{aligned} \mathcal{P}_{\mathcal{T}_X \mathcal{S}_{n,p}}(Y) &= (I - XX^\top)Y + X\mathrm{skew}(X^\top Y), \\ \mathcal{P}^{\perp}_{\mathcal{T}_X \mathcal{S}_{n,p}}(Y) &= X\mathrm{sym}(X^\top Y), \end{aligned}$$

其中 $\mathrm{sym}(A) := \frac{1}{2}(A + A^\top)$, $\mathrm{skew}(A) := \frac{1}{2}(A - A^\top)$.

证明. 由性质 2.2, 2.3 和正交投影的定义 2.1, 假设 $Y = XW + X_\perp K$ ($W \in \mathbb{R}^{n\times p}, K \in \mathbb{R}^{(n-p)\times p}$), 我们不难得到

$$\mathcal{P}^{\perp}_{\mathcal{T}_X \mathcal{S}_{n,p}}(Y) = X \cdot \operatorname*{arg\,min}_{S\in\mathbb{SR}^{p\times p}} \|Y - XS\|_{\mathrm{F}} = X \cdot \operatorname*{arg\,min}_{S\in\mathbb{SR}^{p\times p}} \|W - S\|_{\mathrm{F}},$$

又因为 $W = X^\top Y$, 从而

$$S^* = \operatorname*{arg\,min}_{S\in\mathbb{SR}^{p\times p}} \|W - S\|_{\mathrm{F}} = \mathrm{sym}(X^\top Y),$$

即 $\mathcal{P}^{\perp}_{\mathcal{T}_X \mathcal{S}_{n,p}}(Y) = X\mathrm{sym}(X^\top Y)$. 又因为 $Y = \mathcal{P}_{\mathcal{T}_X \mathcal{S}_{n,p}}(Y) + \mathcal{P}^{\perp}_{\mathcal{T}_X \mathcal{S}_{n,p}}(Y)$, 故性质成立. □

接下来, 我们给出 Stiefel 流形黎曼梯度 (1.10) 的具体形式.

引理 2.1. 对于 *Stiefel* 流形优化问题 (2.2), 我们有

$$\mathrm{grad} f(X) = \mathcal{P}_{\mathcal{T}_X \mathcal{S}_{n,p}}(\nabla f(X)).$$

证明. 由性质 2.4, 对于任意的 $Z \in \mathcal{T}_X \mathcal{S}_{n,p}$, 我们有

$$\begin{aligned} \left\langle \mathcal{P}_{\mathcal{T}_X \mathcal{S}_{n,p}}(\nabla f(X)), Z \right\rangle &= \left\langle \nabla f(X) - \mathcal{P}^{\perp}_{\mathcal{T}_X \mathcal{S}_{n,p}}(\nabla f(X)), Z \right\rangle \\ &= \left\langle \nabla f(X), Z \right\rangle \\ &= \mathcal{D} f(X)[Z]. \end{aligned}$$

根据黎曼梯度 $\mathrm{grad} f(X)$ 的定义 (1.10), 引理得证. □

综上, 我们可以得到 Stiefel 流形优化问题 (2.2) 的一阶最优性条件.

定理 2.1. *假设 $X^* \in \mathcal{M}$ 是 Stiefel 流形优化问题 (2.2) 的局部最优点, 则必有*

$$\mathrm{grad} f(X^*) = (I - X^* X^{*\top})\nabla f(X^*) + X^* \mathrm{skew}(X^{*\top}\nabla f(X^*)) = 0, \tag{2.13}$$

这里 $\mathrm{grad} f$ 表示欧式度量下的黎曼梯度.

证明. 定理 1.5, 性质 2.4 和引理 2.1 的直接推论. □

由性质 2.4 和引理 2.1 可知, 不同的黎曼度量 $\langle \cdot, \cdot \rangle_X$ 诱导了不同的投影算子, 进而导致了不同的黎曼梯度 $\mathrm{grad} f$. 在经典的欧式度量下定理 2.1 成立. 一个很自然的问题是, Stiefel 流形优化问题的最优点是其本质性质, 不应与流形选取的黎曼度量有关. 也就是说, 在其他不同的度量下, 定理 2.1 应该依然成立. 下面我们具体介绍一类黎曼度量, 并给与说明.

定义

$$\langle Z_1, Z_2 \rangle_\rho := \mathrm{tr}(Z_1^\top (I - (1 - \frac{1}{\rho}) X X^\top) Z_2), \quad \forall Z_1, Z_2 \in \mathbb{R}^{n\times p}. \tag{2.14}$$

显然当 $\rho > 0$ 时, 矩阵 $(I - (1 - \frac{1}{\rho}) X X^\top)$ 对称正定, 由此不难验证 $\langle \cdot, \cdot \rangle_\rho$ 为黎曼度量.

假设在此度量下的黎曼梯度记为 $\mathrm{grad}_\rho f(X) \in \mathcal{T}_X \mathcal{S}_{n,p}$, 并且假设 $\mathrm{grad}_\rho f(X) = XW_\rho + X_\perp K_\rho$, 其中 $W \in \mathbb{R}^{p\times p}$ 是反对称矩阵, $K_\rho \in \mathbb{R}^{(n-p)\times p}$. 则对于任意的 $Z = XW_Z + X_\perp K_Z \in \mathcal{T}_X \mathcal{S}_{n,p}$ (W_Z 是反对称矩阵), 通过简单计算, 我们可以推出

$$\begin{aligned} \left\langle \mathrm{grad}_\rho f(X), Z \right\rangle_\rho &= \mathrm{tr}(Z^\top (I - (1 - \frac{1}{\rho}) X X^\top) \mathrm{grad}_\rho f(X)) \\ &= \frac{1}{\rho}\mathrm{tr}(W_\rho^\top W_Z) + \mathrm{tr}(K_\rho^\top K_Z). \end{aligned} \tag{2.15}$$

由黎曼梯度的定义 (1.10), 并假设 $\nabla f(X) = XW + X_\perp K$, 其中 $W \in \mathbb{R}^{p\times p}, K \in \mathbb{R}^{(n-p)\times p}$, 类似的我们有

$$\begin{aligned} \left\langle \mathrm{grad}_\rho f(X), Z \right\rangle_\rho &= \mathcal{D} f(X)[Z] \\ &= \mathrm{tr}(\nabla f(X)^\top Z) \\ &= \mathrm{tr}(W^\top W_Z) + \mathrm{tr}(K^\top K_Z). \\ &= \mathrm{tr}(\mathrm{skew}(W)^\top W_Z) + \mathrm{tr}(K^\top K_Z). \end{aligned} \tag{2.16}$$

其中最后一步利用了 $W_\rho = \mathrm{skew}(W_\rho) + \mathrm{sym}(W_\rho)$ 和 W_Z 的反对称性. 由于 (2.15) 式和 (2.16) 式对于任意的 $Z \in \mathcal{T}_X\mathcal{S}_{n,p}$ 都成立, 比较两式, 我们令

$$W_\rho = \rho \cdot \mathrm{skew}(W), \quad K_\rho = K,$$

由此我们得到了黎曼梯度 $\mathrm{grad}_\rho f(X)$ 的表达式 (唯一性显然),

$$\begin{aligned} \mathrm{grad}_\rho f(X) &= \rho \cdot X\mathrm{skew}(W) + X_\perp K \\ &= \rho \cdot X\mathrm{skew}(W) + \nabla f(X) - XW \\ &= \rho \cdot X\mathrm{skew}(X^\top \nabla f(X)) + \nabla f(X) - XX^\top \nabla f(X) \\ &= (I - XX^\top)\nabla f(X) + \rho \cdot X\mathrm{skew}(X^\top \nabla f(X)). \end{aligned} \tag{2.17}$$

其中用到了 $\nabla f(X) = XW + X_\perp K$ 以及 $W = X^\top \nabla f(X)$.

综上所述, 我们得到了如下的更一般的定理.

定理 2.2. 假设 $X^* \in \mathcal{M}$ 是 *Stiefel* 流形优化问题 (2.2) 的局部最优点, 则必有

$$\mathrm{grad}_\rho f(X^*) = (I - X^*X^{*\top})\nabla f(X^*) + \rho \cdot X^*\mathrm{skew}(X^{*\top}\nabla f(X^*)) = 0,$$

这里 $\mathrm{grad}_\rho f$ 表示黎曼度量 $\langle \cdot, \cdot \rangle_\rho$ 下的黎曼梯度 *($\rho > 0$)*. 特别地,

1) 当 $\rho = 1$ 时, 度量为欧式度量, 此时

$$\mathrm{grad}_1 f(X) = \mathrm{grad} f(X). \tag{2.18}$$

2) 当 $\rho = 2$ 时, 度量为 *Canonical* 度量, 此时

$$\mathrm{grad}_2 f(X) = \nabla f(X) - X\nabla f(X)^\top X. \tag{2.19}$$

3) 当 $\rho > 0$ 时, 我们有

$$\mathrm{grad}_\rho f(X) = \left(I - (1 - \frac{\rho}{2})XX^\top\right) \mathrm{grad}_2 f(X). \tag{2.20}$$

证明. 直接验算可得. □

注 2.1. 文献 *[99, Subsection 4.1], [106, Lemma 2.1], [104, Proposition 1]* 和 *[30, (2.6)]* 分别提出了满足一阶最优性条件的最优解集 $\{X^*\}$ 的刻画, 这与定理 2.2 的结果等价. 在本小节的推导中, 我们给出了一大类黎曼度量的定义, 并且根据其形式, 第一次详细推导了在此度量下成立的黎曼梯度 $\mathrm{grad}_\rho f(X)$ 的具体表达式. 由此我们得到了最优解集的刻画.

在性质 2.2 中, 我们给出了 Stiefel 流形切空间的三种不同刻画. 由于 $\mathrm{grad}_\rho f(X) \in \mathcal{T}_X\mathcal{S}_{n,p}$, 我们发现在定理 2.2 中, $\mathrm{grad}_\rho f(X)$ 恰好是 (2.8) 式的具体形式. 事实上, 根据 (2.9) 式, 我们也有如下的黎曼梯度刻画.

性质 2.5. *对于任意的 $X \in \mathcal{S}_{n,p}$, 我们有*

$$\mathrm{grad}_\rho f(X) = A_\rho X, \tag{2.21}$$

其中 $A_\rho = (P_X\nabla f(X))X^\top - X(P_X\nabla f(X))^\top$, $P_X = I - (1 - \frac{\rho}{2})XX^\top$.

证明. 直接验算可得. □

上述性质可以看做是文献 [99] 中黎曼梯度的直接推广.

2.2.2 正交约束

在上一小节, 通过对 Stiefel 流形结构的刻画, 我们得到了 Stiefel 流形优化问题 (2.2) 的一阶最优性条件. 在本小节中, 我们从欧式空间优化的角度出发, 重新刻画了正交约束优化问题 (2.1) 的最优性条件, 得到了 Lagrange 乘子所满足显式表达式, 这是本文的主要贡献之一, 也是本文部分算法设计的核心.

首先, 我们定义问题 (2.1) 的一阶稳定点 [4].

定义 2.2. *给定点 $X \in \mathbb{R}^{n\times p}$, 如果*

$$\begin{cases} \mathrm{tr}(Y^\top\nabla f(X)) \geq 0; \\ X^\top X = I_p, \end{cases} \tag{2.22}$$

对任意 $Y \in \mathcal{T}_X\mathcal{S}_{n,p}$ 都成立, 则我们称 X 为问题 (2.1) 的一阶稳定点. 我们记包含所有一阶稳定点的集合为 Ω_{FON}.

文献 [107] 将矩阵优化问题 (2.1) 等价转化为 $\mathbb{R}^{np}$ 中的优化问题, 并给出了问题所满足的一阶最优性条件. 其中, 在 [107, Theorem 2.1] 中证明了正交约束 ($X^\top X = I_p$) 满足 LICQ 约束规范性条件 (定义 1.3).

接下来, 根据 (1.4) 式, 我们定义约束优化问题 (2.1) 的 Lagrange 函数为

$$\mathcal{L}(X, \Lambda) := f(X) - \left\langle \Lambda, X^\top X - I \right\rangle. \tag{2.23}$$

其中 $\Lambda \in \mathbb{SR}^{p\times p}$ 是正交约束所对应的 Lagrange 乘子.

下面我们给出正交约束优化问题 (2.1) 的一阶最优性条件.

定理 2.3 (一阶最优性条件). 假设 X^* 是正交约束优化问题 (2.1) 的局部最优点, 函数 f 连续可微, 则 X^* 满足如下的 *KKT* 条件:

$$\begin{cases} (I_n - X^*X^{*\top})\nabla f(X^*) &= 0; & (\text{次稳定性}) \\ X^{*\top}\nabla f(X^*) &= \nabla f(X^*)^\top X^*; & (\text{对称性}) \\ X^{*\top}X^* &= I_p. & (\text{可行性}) \end{cases} \tag{2.24}$$

证明. 由于正交约束满足 LICQ 约束规范性条件 [107, Theorem 2.1], 根据 Lagrange 函数的定义 (2.23) 以及定理 1.2, 我们有

$$\nabla_X \mathcal{L}(X^*, \Lambda^*) = \nabla f(X^*) - X^*\Lambda = 0,$$

从而 $\Lambda^* = X^{*\top}\nabla f(X^*)$. 将其代入上式, 我们得到

$$(I_n - X^*X^{*\top})\nabla f(X^*) = 0.$$

又因为正交约束满足对称性, 故 Lagrange 乘子 Λ^* 对称. 由此定理得证. □

由于一阶稳定点条件 (定义 2.2) 无法被数值验证, 因此我们证明了如下的等价结论.

引理 2.2. 点 $X \in \mathbb{R}^{n\times p}$ 是问题 (2.1) 的一阶稳定点当且仅当 *KKT* 条件 (2.24) 成立.

证明. 由性质 2.2, 对于任意的 $Y \in \mathcal{T}_X\mathcal{S}_{n,p}$, 都能被唯一的分解为 $Y = XS + K$, 这里 $S \in \mathbb{R}^{p\times p}$ 是反对称矩阵 ($S^\top + S = 0$) 且 $K \in \mathbb{R}^{n\times p}$ 满足 $K^\top X = 0$, 这等价于 $K = (I_n - XX^\top)K$. 同样的, 任意具有 $XS + K$ 形式的矩阵都属于 $\mathcal{T}_X\mathcal{S}_{n,p}$.

由于 S 和 K 都是任意的, 故条件 (2.22) 等价于如下的关系式

$$\operatorname{tr}(S^\top X^\top \nabla f(X)) \geq 0, \qquad \forall\, S \in \mathbb{R}^{p\times p} \text{ 且 } S^\top + S = I_p, \tag{2.25}$$

$$\operatorname{tr}(K^\top \nabla f(X)) \geq 0, \qquad \forall\, K \in \mathbb{R}^{n\times p} \text{ 且 } K^\top X = 0, \tag{2.26}$$

$$X^\top X = I_p.$$

记 $Q := X^\top \nabla f(X)$, 利用 (2.25) 式和 $Q^\top - Q$ 的反对称性, 我们有

$$\operatorname{tr}((Q - Q^\top)Q) \geq 0. \tag{2.27}$$

由 (2.27) 式可得

$$\begin{aligned} 0 &\leq \operatorname{tr}((Q - Q^\top)Q) + \operatorname{tr}((Q - Q^\top)Q) = \operatorname{tr}(QQ - Q^\top Q) + \operatorname{tr}(Q^\top(Q^\top - Q)) \\ &= \operatorname{tr}(QQ - Q^\top Q) + \operatorname{tr}((Q^\top - Q)Q^\top) = \operatorname{tr}(QQ - Q^\top Q + Q^\top Q^\top - QQ^\top) \\ &= \operatorname{tr}((Q - Q^\top)(Q - Q^\top)) = -\operatorname{tr}((Q - Q^\top)^\top(Q - Q^\top)) \leq 0, \end{aligned}$$

也就是 $Q = Q^\top$.

另一方面, 如果 $X^\top \nabla f(X)$ 是对称的, 等式 $\mathrm{tr}(S^\top X^\top \nabla f(X)) = 0$ 对任意反对称矩阵 S 都成立. 因此, (2.25) 式等价于 $X^\top \nabla f(X)$ 的对称性.

由性质 $K = (I_n - XX^\top)K$ 和 K 的任意性, 我们很容易得知 (2.26) 式和 $(I_n - XX^\top)\nabla f(X) = 0$ 的等价性. 由此引理得证. □

因此, 我们得到了一阶稳定点条件与问题 KKT 条件的等价性, 即局部最优解也是问题的一阶稳定点. 另一方面, 在定理 2.3 的证明中, 我们可以很自然地得到局部最优解 X^* 所满足的 Lagrange 乘子的表达式.

推论 2.1. *假设 X 是问题 (2.1) 的一阶稳定点, 此时正交约束的 Lagrange 乘子满足*

$$\Lambda = X^\top \nabla f(X) = \nabla f(X)^\top X. \tag{2.28}$$

注 2.2. *通常来讲, 在一些基于 Lagrange 函数的算法中, 乘子的更新对于整个算法的表现至关重要. 推论 2.1 给了我们很好的启发, 有时候我们不需要迭代的进行乘子更新, 而是直接采用 (2.28) 式给出的显式更新方式. 在第 5 章中, 我们采用这种思想, 设计了一种新算法.*

最后, 由欧式空间中的二阶最优性条件 (定理 1.3 和 1.4), 我们给出正交约束优化问题 (2.1) 所满足的二阶最优性条件.

定义 2.3 ([99, Lemma 2])**.** *若 X 是问题 (2.1) 的一阶稳定点且满足*

$$\mathrm{tr}(Z^\top \nabla^2 f(X)[Z] - \Lambda Z^\top Z) \geq 0, \quad \forall\, Z \in \mathcal{T}_X \mathcal{S}_{n,p}. \tag{2.29}$$

其中 Λ 由 (2.28) 式定义, 则我们称其为二阶稳定点.

性质 2.6 ([99, Lemma 2])**.** *若 X 是问题 (2.1) 的一个局部极小值点, 则其一定是二阶稳定点. 若 X 是问题 (2.1) 的一个严格局部极小值点, 当且仅当 X 是一阶稳定点并且满足*

$$\mathrm{tr}(Y^\top \nabla^2 f(X)[Y] - \Lambda Y^\top Y) > 0, \quad \forall\, 0 \neq Y \in \mathcal{T}_X \mathcal{S}_{n,p}, \tag{2.30}$$

其中 Λ 由 (2.28) 式定义.

2.2.3 判断准则

至此, 我们分别得到了 Stiefel 流形优化问题 (2.2) 和正交约束优化问题 (2.1) 的最优性条件. 在实际中, 我们经常需要判断算法生成的迭代点列是否已经近似满足问题的最优性条件. 因此, 在本小节, 我们比较不同最优性条件之间的区别.

首先, 我们可以很容易验证欧式空间中的一阶最优性条件 (2.24) 恰好等于文献 [99, Lemma 1] 中提到的一阶最优性条件,

$$\begin{cases} \nabla f(X) - X\nabla f(X)^\top X &= 0; \\ X^\top X &= I_p. \end{cases}$$

假设由算法生成的迭代点列满足可行性, 也就是正交约束 $X^\top X = I_p$ 始终成立, 则我们可以通过计算

$$\left\|\nabla f(X) - X\nabla f(X)^\top X\right\|_{\mathrm{F}}$$

来判断迭代点列是否应该终止. 这恰好就是 (2.19) 式定义的黎曼梯度范数

$$\left\|\mathrm{grad}_2\, f(X)\right\|_{\mathrm{F}}.$$

通过计算, 我们发现

$$\begin{aligned} & \left\|\nabla f(X) - X\nabla f(X)^\top X\right\|_{\mathrm{F}}^2 \\ = \; & \left\|\nabla f(X) - XX^\top \nabla f(X)\right\|_{\mathrm{F}}^2 + \left\|X^\top \nabla f(X) - \nabla f(X)^\top X\right\|_{\mathrm{F}}^2. \end{aligned} \tag{2.31}$$

此式中, 第一项恰好是 KKT 条件 (2.24) 中的次稳定性违反度, 第二项正好是 (2.24) 中 Lagrange 乘子 Λ 的对称性违反度.

受到上述讨论的启发, 对于算法生成的可行迭代点列, 如果我们用 KKT 条件 (2.24) 来判断迭代点列是否满足最优性条件, 则我们需要同时计算 $\left\|\nabla f(X) - XX^\top \nabla f(X)\right\|_{\mathrm{F}}^2$ 和 $\left\|X^\top \nabla f(X) - \nabla f(X)^\top X\right\|_{\mathrm{F}}^2$, 也就是次稳定性和对称性的违反度. 针对这两项, 我们给予不同的权重组合, 由此得到

$$\left\|\nabla f(X) - XX^\top \nabla f(X)\right\|_{\mathrm{F}}^2 + \rho^2 \left\|X^\top \nabla f(X) - \nabla f(X)^\top X\right\|_{\mathrm{F}}^2,$$

其中 $\rho > 0$.

另一方面, 考虑黎曼梯度 (2.17) 式, 我们有

$$\begin{aligned} & \left\|\mathrm{grad}_\rho\, f(X)\right\|_{\mathrm{F}}^2 \\ = \; & \left\|(I - XX^\top)\nabla f(X) + \rho X \mathrm{skew}(X^\top \nabla f(X))\right\|_{\mathrm{F}}^2 \\ = \; & \left\|\nabla f(X) - XX^\top \nabla f(X)\right\|_{\mathrm{F}}^2 + \rho^2 \left\|X^\top \nabla f(X) - \nabla f(X)^\top X\right\|_{\mathrm{F}}^2 \end{aligned}$$

这恰好与 KKT 条件 (2.24) 特殊定义的判断准则完全相同. 由此我们建立了黎曼梯度 $\mathrm{grad}_\rho f(X)$ 和欧式空间一阶最优性条件 (2.24) 的内在关系.

注 2.3. *我们考虑 Stiefel 流形的黎曼梯度*

$$\mathrm{grad}_\rho f(X) = (I - XX^\top)\nabla f(X) + \rho \cdot X\mathrm{skew}(X^\top \nabla f(X)).$$

由上述讨论可知, $\mathrm{grad}_\rho f(X)$ 中的第一项对应于欧式空间一阶最优性条件 (2.24) 中的次稳定性. 第二项对应于 (2.24) 中的 Lagrange 乘子对称性. 这两种最优性条件的本质区别在于, Stiefel 流形优化的一阶最优性条件在不同的黎曼度量下有不同的权重组合, 而欧式空间的一阶最优性条件 (2.24) 则更为本质, 不依赖于权重的选取. 为了方便起见, 在本文的实际计算中, 我们采用 (2.31) 式是否足够小作为算法的终止准则.

2.3 算法综述

在本小节, 我们简要介绍一些已有的正交约束优化问题 (2.1) 的求解算法, 其中包括收缩类方法和不可行方法等.

2.3.1 收缩类方法

在 1.2.2 小节, 我们详细介绍了一般黎曼流形 $\mathcal{M}$ 上的收缩类线搜索方法. 根据算法 1, 我们需要做的就是在切空间 $\mathcal{T}_X\mathcal{M}$ 中选取合适的下降搜索方向 D^k, 以及有效的收缩映射 $\mathcal{R}_X$. 根据不同的下降方向和算法框架, 我们有诸如梯度类方法 [108–110]、共轭梯度法 [110, 111]、信赖域方法 [41, 112]、牛顿法 [111] 和拟牛顿法 [43, 113, 114] 等收缩类算法. 特别地, 如果我们取 $\mathcal{M}$ 为 Stiefel 流形, 则这些方法也可完全适用于问题 (2.2).

值得说明的是, 上述这些方法都基于收缩映射 (定义 1.4), 它定义了一种保持正交性的更新准则, 也就是说这些方法都是可行方法. 对于 Stiefel 流形来说, 收缩映射 $\mathcal{R}_X$ 的选取有很多种, 大致上我们可以将其分为两类, 测地线类 [33, 110, 111] 和投影类 [33, 42, 108], 相应的收缩类线搜索算法分别被称为测地线类方法和投影类方法.

接下来我们分别具体介绍这两类方法. 给定点 $X \in \mathcal{S}_{n,p}$, $D \in \mathcal{T}_X\mathcal{S}_{n,p}$ 为下降搜索方向. $\mathcal{R}_X(tD)$ 表示收缩后的点, 其中 t 为搜索步长. 由定义 1.4 可知, $\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{R}_X(tD)|_{t=0} = D$.

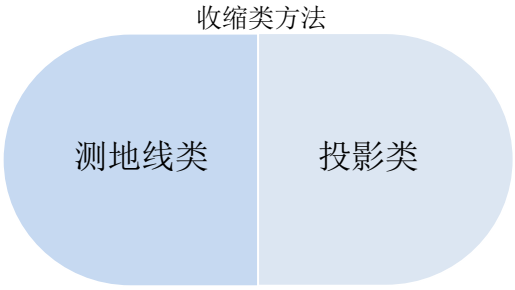


图 2.2 收缩类方法: 测地线类和投影类

Figure 2.2 Retraction-based methods: geodesic-type and projection-type

2.3.1.1 测地线类

测地线类方法, 顾名思义, 就是沿着 Stiefel 流形的测地线搜索合适的试探步.

文献 [111] 的工作开辟了正交约束优化问题新的研究领域, 他们详细刻画了 Stiefel 流形的几何结构, 并给出了测地线的具体表达形式. 由此, 他们还提出了如下的可行搜索轨迹,

$$\mathcal{R}_X^{\mathrm{geoe}}(tD) = [X, Q] \exp\left(t \begin{bmatrix} -X^\top D & -R^\top \\ R & 0 \end{bmatrix} \right) \begin{bmatrix} I_p \\ 0 \end{bmatrix}, \tag{2.32}$$

其中 $QR = -(I - XX^\top)D$ 表示矩阵 $-(I - XX^\top)D$ 的部分 QR 分解[2], $\exp(X)$ 表示矩阵指数函数[3]. 在上述更新轨迹中, 我们观察到每一次搜索都需要计算一个 $2p$ 阶的矩阵指数, 这将会花费大量计算代价.

事实上, 更新轨迹 (2.32) 式是由欧式度量推导得到的. 文献 [110] 根据 Canonical 度量, 提出了另一类测地线更新公式,

$$\mathcal{R}_X^{\mathrm{geoc}}(tD) = \exp(-tA)X, \tag{2.33}$$

其中 $A \in \mathbb{R}^{n\times n}$ 是反对称矩阵. 在 [110] 中, A 取为 $\nabla f(X)X^\top - X\nabla f(X)^\top$ (事实上, 就是性质 2.5 中的 A_2), 此时真正的搜索方向 $D = -\mathrm{grad}_2\, f(X) = -A_2X$. 这里每一次更新都需要计算一个 n 阶矩阵指数, 当 $p \geq n/2$ 时, 迭代公式 (2.33) 的矩阵指数计算量要少于公式 (2.32).

由更新公式 (2.32) 和 (2.33) 可知, 计算测地线需要涉及矩阵指数运算, 这会

[2]列满秩矩阵 $X \in \mathbb{R}^{n\times p}$ 的部分 QR 分解表示为 $X = QR$, 其中 $Q \in \mathbb{R}^{n\times p}$ 正交并且 $R \in \mathbb{R}^{p\times p}$ 是具有正对角元的上三角矩阵.

[3]矩阵指数的定义请参考 [58, Subsection 9.3].

带来计算上的困难. 文献 [109] 基于 Cayley 变换[4]提出了一种拟测地线更新公式,

$$\mathcal{R}_X^{\mathrm{qgeo}}(tD) = \left(I + \frac{t}{2}A\right)^{-1}\left(I - \frac{t}{2}A\right)X, \tag{2.34}$$

其中 $A \in \mathbb{R}^{n\times n}$ 是反对称矩阵, A 若取为 $\nabla f(X)X^\top - X\nabla f(X)^\top$, 则 $D = -\mathrm{grad}_2\, f(X) = -A_2X$. 在实际计算中, 拟测地线公式每一步迭代的运算量是求解一个 n 阶线性方程组.

文献 [99] 提出了一个保持约束可行的有效方法. 基于工作 [115], 他们将求解偏微分方程组的 Crank-Nicholson 技术推广到了矩阵变量, 提出了一类曲线搜索方法. 事实上, Crank-Nicholson 技术在矩阵计算中恰好对应于 Cayley 变换, 因此他们提出了一种与 Cayley 变换更新公式 (2.34) 完全等价的曲线搜索策略,

$$\mathcal{R}_X^{\mathrm{wy}}(tD) = X - tU\left(I_{2p} + \frac{t}{2}V^\top U\right)^{-1}V^\top X, \tag{2.35}$$

其中 $U = [P_XD, X]$, $V = [X, -P_XD]$, $P_X = (I_n - \frac{1}{2}XX^\top)$. 在迭代更新中, 此公式每步需求解一个 $2p$ 阶线性方程组, 而不是原 Cayley 变换 (2.34) 式中的 n 阶线性方程组. 当 $p \le n/2$ 时, 其计算量得到了大幅降低. 另一方面, 通过结合 Barzilai-Borwein[5]非单调线搜索策略 [116], 更新公式 (2.35) 在实际数值计算中表现更加优异, 并且已经实际应用于多个领域, 例如电子结构计算 [49].

2.3.1.2 投影类

投影类方法主要包含两个步骤, 先在切空间进行线搜索, 接着 “投影[6]” 回 Stiefel 流形. 矩阵的正交化过程, 包括奇异值分解、QR 分解和极分解 (polar decomposition), 都可以用来计算 $\mathbb{R}^{n\times p}$ 中的点到 Stiefel 流形的 “投影”.

在定义 2.1 中, 我们可以得到任意点到闭凸集的投影. 尽管 Stiefel 流形是非凸的, 我们仍然可以定义其正交投影如下.

性质 2.7 ([117, Theorem 1]). *给定任意的满秩矩阵 $Y \in \mathbb{R}^{n\times p}$, 其在 Stiefel 流形上的正交投影为*

$$\mathcal{P}_{\mathcal{S}_{n,p}}(Y) = \operatorname*{arg\,min}_{X\in\mathcal{S}_{n,p}} \|Y - X\|_{\mathrm{F}} = UV^\top = Y(Y^\top Y)^{-1/2},$$

这里 Y 的奇异值分解为 $Y = U\Sigma V^\top$, 其中 $U \in \mathbb{R}^{n\times p}$ 为正交矩阵, $\Sigma \in \mathbb{R}^{p\times p}$ 是对角阵, $V \in \mathbb{R}^{p\times p}$. $C^{1/2}$ 表示对称正定矩阵 $C \in \mathbb{R}^{p\times p}$ 的平方根[7].

[4]假设 $S \in \mathbb{R}^{n\times n}$ 是反对称矩阵, 即 $S^\top = -S$, 则 $I - S$ 可逆并且 $(I - S)^{-1}(I + S)$ 正交, $(I - S)^{-1}(I + S)$ 称为矩阵 S 的 Cayley 变换.

[5]参见 1.1.3 小节.

[6]这里的投影只是一种形象的描述, 并不仅仅指正交投影 $\mathcal{P}_{\mathcal{S}_{n,p}}$.

[7]矩阵平方根的定义请参考 [58, Subsection 4.2.4].

利用性质 2.7, 我们可以将凸约束优化问题中的梯度投影方法 [3] 应用到正交约束优化问题 (2.1) 中. 由此得到的迭代更新公式如下,

$$\mathcal{R}_X^{\mathrm{pg}}(tD) = \mathcal{P}_{\mathcal{S}_{n,p}}(X - t\nabla f(X)). \tag{2.36}$$

我们注意到, 这里的搜索方向是 $-\nabla f(X)$. 事实上, 通过计算我们发现

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{P}_{\mathcal{S}_{n,p}}(X - t\nabla f(X))|_{t=0} = -\mathrm{grad}\, f(X),$$

因此, 这里 $D = -\mathrm{grad}\, f(X)$.

文献 [108] 提出了 Stiefel 流形上的标准投影方法,

$$\mathcal{R}_X^{\mathrm{pj}}(tD) = \mathcal{P}_{\mathcal{S}_{n,p}}(X - tD). \tag{2.37}$$

这里 $D \in \mathcal{T}_X\mathcal{S}_{n,p}$, 此公式等价于计算矩阵 $X - tD$ 的奇异值分解.

基于 QR 分解和极分解, 文献 [33] 提出了两种投影类收缩方法, 具体如下

$$\mathcal{R}_X^{\mathrm{qr}}(tD) = \mathrm{qr}(X - tD), \tag{2.38}$$

$$\mathcal{R}_X^{\mathrm{pd}}(tD) = (X - tD)(I_p + t^2D^\top D)^{-1/2}. \tag{2.39}$$

其中 $\mathrm{qr}(X)$ 表示矩阵 X 的部分 QR 分解的 Q 矩阵. 事实上, 极分解投影方法 (2.39) 本质上等价于奇异值分解, 也就和标准投影方法 (2.37) 完全等价. 两者的区别在于极分解的计算量要小于直接对 $X - tD$ 做奇异值分解.

在本文第 3 章中, 我们提出了一个新的乘子校正算法框架, 其中包含梯度投影法和梯度反射法. 一般来讲, 我们的新算法框架并不属于收缩类方法, 因此我们称其为非收缩类方法. 但是特别地, 若当前点 X 满足对称性条件 $X^\top\nabla f(X) = \nabla f(X)^\top X$, 则此时我们定义的梯度投影步 (3.20) 等价于收缩算子 $\mathcal{R}_X^{\mathrm{pg}}$. 梯度反射步

$$\begin{cases} V = X^k - t\nabla f(X^k), \quad \text{取定 } t \in (0, \rho^{-1}); \\ Y_{\mathrm{GR}}(t; X) = \bar{X}_{\mathrm{GR}} = (-I_n + 2V(V^\top V)^\dagger V^\top)X^k. \end{cases}$$

恰好满足收缩映射的定义, 其中 $\frac{\mathrm{d}}{\mathrm{d}t}Y_{\mathrm{GR}}(t; X)|_{t=0} = -2\mathrm{grad}_0\, f(X)$. 因此, 在本文中, 我们也定义了一类只在特殊条件下成立的收缩方法.

文献 [106] 推广了 [99] 的想法, 通过在可行点 $X \in \mathcal{S}_{n,p}$ 处对全空间进行子空间分解, 定义搜索轨迹为

$$Y_{\mathrm{jd}}(t; X) = XR(t) + WN(t), \tag{2.40}$$

其中 $XR(t)$ 在 X 的值空间, $WN(t)$ 在 $X^\top$ 的零空间. [106] 的主要思想是寻找合适的 $R(t)$ 和 $N(t)$, 使得轨迹 $Y_{\rm jd}(t;X)$ 始终保持可行性. 也就是对于任意的 $t>0$, 都有 $Y_{\rm jd}(t;X)\in\mathcal{S}_{n,p}$. 他们根据不定系数法提出了一大类收缩的算法框架, 并且上述的某些测地线类和投影类的方法都可以看做是他们算法框架的特例, 例如 $\mathcal{R}_X^{\rm geoe}$, $\mathcal{R}_X^{\rm wy}$, $\mathcal{R}_X^{\rm pg}$, $\mathcal{R}_X^{\rm pj}$, $\mathcal{R}_X^{\rm qr}$ 以及 $\mathcal{R}_X^{\rm pd}$. 结合 BB 方法, 他们具体提出了一种自适应可行的类 BB 方法,

$$\begin{cases} W=-(I_n-XX^\top)D, \\ J(t)=I_p+\frac{t^2}{4}WW^\top+g(t)X^\top D, \\ \mathcal{R}_X^{\rm jd}(tD)=(2X+tW)J(t)^{-1}-X. \end{cases}$$

其中 $D\in\mathcal{T}_X\mathcal{S}_{n,p}$ 是任意给定的切空间方向, $g(t)$ 是任意满足 $g(0)=0, g'(0)=1/2$ 的函数. 值得说明, 满足这个更新公式的方法仍是收缩类方法.

注 2.4. 当 $p=1$ 时, 正交约束退化为球约束, 此时投影类方法与测地线类方法有相同的迭代轨迹. 我们可以很容易验证上述更新公式都是收缩映射, 由此上述方法都可以归类为收缩类方法. 在实际中, 测地线类和投影类方法, 都需要结合具体的线搜索策略, 在测地线或切空间上进行搜索, 例如 *Armijo* 非精确线搜索 *(*定义 *1.6)* 或者非单调线搜索 *[116]*. 线搜索策略用来保证算法的全局收敛性, 但与此同时也会增加额外的函数值, 梯度值估计, 从而增加计算代价.

在本文的第 3 章中, 我们将提出一个新的可行算法框架, 新方法显著的减少了每步迭代所需的计算量, 并且不需要进行线搜索. 值得说明的是, 我们的新可行算法框架并不属于收缩类方法, 由此我们构建了一个新的非收缩类方法研究领域.

2.3.2 不可行方法

在上一小节中, 我们主要介绍了一大类可行方法, 其每步迭代至少都需要进行一次矩阵分解, 或者线性方程组求解, 或者矩阵指数计算. 一方面, 这些操作将会带来巨大的计算开销, 尤其是再考虑线搜索算法, 计算量更会显著增加. 另一方面, 上面提到的这些线性代数计算都很难并行实现, 这对于大规模问题而言, 将会成为计算高效性的主要瓶颈. 因此, 综合上面的考虑, 一些学者开始研究正交约束优化问题的不可行方法.

文献 [49] 针对迹极小化问题 (2.3), 提出了如下的 Courant 罚函数模型 [118],

$$\min_{X\in\mathbb{R}^{n\times p}}\ \frac{1}{2}{\rm tr}(X^\top AX)+\frac{\mu}{4}\|X^\top X-I\|_{\rm F}^2. \tag{2.41}$$

其中 $\mu > 0$ 是罚参数. 在适当选取 μ 的条件下, 他们证明了上述无约束优化问题等价于最初的迹极小化问题 (2.3). 结合 BB 非单调线搜索技术, 他们提出了针对无约束优化问题 (2.41) 的梯度方法 (类似于 1.1.3 小节的讨论), 其具体更新形式为

$$X^{k+1} = X^k - \alpha_k(AX + \mu X(X^\top X - I)).$$

由于整个迭代过程最主要的计算量只涉及矩阵的乘法运算, 因此文献 [49] 采用并行编程模型 OpenMP 对算法进行并行化处理. 由此得到的算法在特征值问题的测试中, 表现优异并且具有良好的可扩展性.

文献 [119] 基于 [120] 中提出的无约束优化模型, 考虑了更一般的无约束 β 次优化模型,

$$\min_{X\in\mathbb{R}^{n\times p}} \frac{\theta}{\beta}\left\|X^\top X\right\|_{\mathrm{F}}^{\frac{\beta}{2}} + \frac{1}{2}\mathrm{tr}(X^\top(A-\mu I)X).$$

其中 $\beta > 2, \theta > 0, \mu \in \mathbb{R}$ 表示平移参数. 特别地, 当 $\beta = 4, \theta = 1$ 时, 上述问题与模型 (2.41) 等价. 采用和 [49] 类似的方法, 文献 [119] 的数值结果验证了其模型的有效性.

文献 [121] 将如下的对称低秩乘积模型

$$\min_{X\in\mathbb{R}^{n\times p}} \frac{1}{2}\left\|XX^\top - A\right\|_{\mathrm{F}}^2,$$

用于求解迹极小化问题 (2.3). 将上述问题看成是最小二乘模型, 他们提出了一个高效的 Gauss-Newton 算法. 大量的数值实验表明, 该算法的表现优于已有的一些 Krylov 子空间方法 [122, 123].

上述这些方法都只能应用于特征值问题的求解. 最近, 针对一般的正交约束优化问题 (2.1), 文献 [117] 利用时下研究非常活跃的交替方向乘子法 (ADMM) [51] 和分裂的 Bregman 迭代 [124], 通过引入辅助变量将目标函数与正交约束分离, 得到了如下的优化问题,

$$\begin{aligned} \min_{X,Y\in\mathbb{R}^{n\times p}} \quad & f(X) \\ \text{s.t.} \quad & X = Y, \\ & Y^\top Y = I_p. \end{aligned}$$

同时, [117] 还提出了求解上述问题的 Bregman 迭代方法. 关于该算法的数值表现, 我们将会在第 5 章中详细讨论.

文献 [125] 利用临近点交替极小化策略 [126] 以及增广 Lagrange 方法, 针对

如下非光滑迹极小化问题,

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & g(X)+\mu\cdot \mathrm{tr}(X^\top HX) \\ \text{s.t.} \quad & X^\top MX = I_p, \end{aligned}$$

提出了一类有效的不可行方法. 其中 $H\in\mathbb{SR}^{n\times n}$, 矩阵 $M\succ 0$, 并且 $g(X)$ 是非光滑凸函数 (例如 ℓ_1 正则化).

文献 [76] 将 [125] 的结果推广到更一般的非凸非光滑正交约束优化问题,

$$\begin{aligned} \min_{X,Y\in\mathbb{R}^{n\times p}} \quad & f(X)+g(Y)+h(X,Y) \\ \text{s.t.} \quad & AX+BY=C, \\ & X^\top MX = I_p. \end{aligned}$$

其中 $A,B\in\mathbb{R}^{m\times n}$ 且 B 满秩, 矩阵 $M\succeq 0$, f 和 g 是下半连续的真凸函数, h 是连续函数.

据我们所知, 目前还没有能够高效求解一般的正交约束优化问题的不可行方法. 在本文的第 5 章中, 针对一般的正交约束优化问题, 我们将会引入一个新的不可行算法框架. 除此之外, 新方法还可以采用高效的并行计算, 这将极大的拓展新算法求解问题的规模和能力.

2.3.3 其它算法及软件包

近两年来, 在收缩类方法框架下, 研究者还提出了一些有效的改进牛顿方法 [24, 127]. 这些方法即利用了牛顿法在局部的二次收敛速度, 又结合了一些全局化的修正技术, 确保了算法的全局收敛性. 最近, 文献 [128] 将采用随机扩散的全局优化技术应用到正交约束优化问题中, 提出了正交约束优化问题的全局优化算法.

文献 [30] 将工作 [106] 推广到如下带有正交约束的经验风险最小化问题:

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & f(X):=\tfrac{1}{m}\textstyle\sum_{i=1}^m f_i(X) \\ \text{s.t.} \quad & X^\top X = I_p. \end{aligned}$$

其中, 他们提出了一种不需要引入额外流形计算的一般流形上的随机方差下降梯度方法. 大量的数值实验表明, 针对主成分分析问题和矩阵完整化问题, 该算法十分有效.

目前, 被研究人员所广泛使用的矩阵流形优化软件包是 Manopt[8] [129], 这是一个 Matlab 语言的软件包, 整合了时下最新的黎曼流形优化算法. 常用微分几

[8] https://www.manopt.org

何的运算都内嵌在软件包中, 这使得整个算法包更便于研究者以及低门槛用户使用. 另一个比较高效的软件包 OptM[9]基于文献 [99] 开发, 它是针对一般的正交约束优化问题的 Matlab 语言软件包. 该软件包调用简单并且在大部分正交约束优化问题中表现稳定且高效. ARNT[10] 是文献 [24] 基于二阶优化方法所开发的黎曼流形优化求解器, 它具有较快的局部收敛速度, 并且在大量的测试问题中表现优异. ROPTLIB[11] [130] 是一个 C++ 语言的黎曼流形优化库, 内部包含了许多已有的黎曼流形算法. 除此之外, ROPTLIB 库还可以分别在 Matlab 和 Julia 环境下独立运行, 其效率要优于仅在 Matlab 和 Julia 中运行的代码.

最后, 我们将一些常用的正交约束优化求解器总结在表 2.1 中.

名称	编程语言	简要描述	参考文献
unit_opt	MATLAB	酉矩阵约束优化求解器[12]	[110, 110]
GenRTR	MATLAB	黎曼信赖域优化算法包[13]	[41, 131]
OptM	MATLAB	正交约束优化求解器[9]	[99]
Manopt	MATLAB	矩阵流形优化工具箱[8]	[129]
ROPTLIB	C++	黎曼流形优化库[11]	[130]
ARNT	MATLAB	黎曼流形二阶优化求解器[10]	[24]
McTorch	Python	深度学习 (PyTorch) 黎曼优化库[14]	[132]

表 2.1 正交约束优化求解器

Table 2.1 Solvers for optimization with orthogonality constraints

2.4 小结

在本章中, 我们详细介绍了正交约束优化问题的应用背景. 接着, 我们分别从 Stiefel 流形优化和欧式空间约束优化两个不同角度, 推导了问题的最优性条件. 其中通过分析一大类黎曼度量下的黎曼梯度, 我们得到了 Stiefel 流形优化和欧式空间约束优化的一阶最优性条件的对应关系. 除此之外, 我们还得到在任意一阶稳定点处, 正交约束优化问题的 Lagrange 乘子具有显式表达式. 最后, 我们详细介绍了已有的正交约束优化算法.

[9] https://github.com/wenstone/OptM

[10] https://github.com/wenstone/ARNT

[11] https://www.math.fsu.edu/~whuang2/Indices/index_ROPTLIB.html

[12] http://legacy.spa.aalto.fi/sig-legacy/unitary_optimization/index.html

[13] https://www.math.fsu.edu/~cbaker/GenRTR/

[14] https://github.com/mctorch/mctorch

第 3 章　乘子校正算法

在本章中, 我们考虑一大类具有广泛应用的正交约束优化问题. 针对这类问题, 我们提出了新的非收缩类算法框架, 其主要包含两个步骤, 分别是函数值下降步和乘子校正步. 不同于第 2 章中介绍的已有方法, 在我们的算法框架中, 函数值下降步沿标准的欧式负梯度下降方向进行搜索, 而不是 Stiefel 流形切空间的方向. 另一方面, 我们构造的乘子校正步进一步使函数值下降, 同时保证了乘子, 也就是对偶变量的对称性. 基于此算法框架, 我们提出了两大类算法. 第一类是梯度下降方法, 其中包括梯度反射法 (GR) 和梯度投影法 (GP). 第二类采用以列为块的块坐标下降方法 (CBCD), 同时我们也提出了一个新的方法用于非精确求解对应的块坐标下降子问题. 我们证明了常数步长的梯度反射法和梯度投影法, 还有块坐标下降法都包含在我们的算法框架中, 进而证明了由算法生成的相应迭代点列的聚点都是一阶稳定点. 数值实验表明我们的算法框架具有很大的潜力.

3.1　引言

在本章中, 我们考虑如下的带有正交约束的矩阵变量优化问题,

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & f(X) := h(X) + \mathrm{tr}(G^\top X) \\ \text{s.t.} \quad & X^\top X = I_p, \end{aligned} \tag{3.1}$$

其中 I_p 表示 p 阶单位矩阵, $p < n$, 目标函数 $f : \mathbb{R}^{n\times p} \longrightarrow \mathbb{R}$, 满足如下假设.

假设 3.1. *1. f 二次可微. 我们定义 ρ 为*

$$\rho := \sup_{X\in\tilde{\mathcal{S}}} \left\|\nabla^2 f(X)\right\|_2,$$

这里 $\tilde{\mathcal{S}} := \{Y \mid ||Y||_{\mathrm{F}}^2 < p+1\}$[1].

2. $f(X) = h(X) + \mathrm{tr}(G^\top X)$, 其中 $G \in \mathbb{R}^{n\times p}$, $h(X)$ 满足正交不变性, 也就是, $h(XQ) = h(X)$ 对任意 $Q \in \mathcal{S}_{p,p}$ 都成立, 并且有 $\nabla h(X) = H(X)X$, 这里 $H : \mathbb{R}^{n\times p} \longrightarrow \mathbb{SR}^{n\times n}$ 是一个矩阵映射.

在实际中, ρ 的值经常不可知并且难以估计. 在本章中, 我们将会给出几种估计方法.

[1]事实上, $\tilde{S}$ 可以为包含 $\mathcal{S}_{n,p} = \{Y \in \mathbb{R}^{n\times p} \mid Y^\top Y = I_p\}$ 的任意给定有界开集.

注 3.1. 若 $\rho = 0$, 目标函数 $f(X)$ 退化为线性函数 $\mathrm{tr}(G^\top X)$. 在这种情况下, 问题 (3.1) 本质上等价于正交投影 $\mathcal{P}_{\mathcal{S}_{n,p}}(-G) = \arg\min_{X\in\mathcal{S}_{n,p}} \|-G - X\|_{\mathrm{F}}$. 由性质 2.7 可知, 问题 (3.1) 的解可显式表达为 $X = -UV^\top$, 这里 $U\Sigma V^\top$ 是 G 的奇异值分解. 在本章中, 我们不讨论这种特殊情况.

事实上, 在我们新提出的算法框架中, 只需要假设 3.1 就可完全得到算法的全局收敛性. 因此, 在本章中我们不考虑如何弱化这个充分条件. 另一方面, 由于我们假设了问题的结构, 即目标函数 $f(X)$ 所满足的性质, 很自然地我们要问: 这个假设的适用性是否广泛. 实际上, 在许多重要的实际问题中, 假设 3.1 都成立. 这里我们给出两个例子予以说明.

例 3.1.

$$f(X) := \frac{1}{2}\mathrm{tr}(X^\top AX) + \mathrm{tr}(G^\top X),$$

其中 $A \in \mathbb{SR}^{n\times n}$. 在这个例子中, 我们有

$$\nabla f(X) = AX + G, \qquad H(X) = A.$$

我们注意到如果例 3.1 中的目标函数取 $G = 0$, 相应的正交约束优化问题将会退化为 Rayleigh-Ritz 极小化问题, 而其恰好是特征值问题的优化模型 (2.3). 然而当 $G \neq 0$, 问题将变得难以求解, 甚至于当 A 正定的时候亦是如此. 实际上, 例 3.1 也是信赖域方法用于求解正交约束优化问题的关键子问题 (参见 [71]).

例 3.2.

$$f(X) := \frac{1}{2}\mathrm{tr}(X^\top AX) + \frac{1}{2}\sum_{i=1}^{m} q_i(z),$$

其中 $z = \mathrm{diag}(XX^\top)$, $q_i : \mathbb{R}^n \to \mathbb{R}$ $(i = 1, \cdots, m)$ 并且 $A \in \mathbb{SR}^{n\times n}$. 在这个例子中, 我们有

$$\nabla f(X) = \left(A + \sum_{i=1}^{m} \mathrm{Diag}(\nabla q_i(z))\right) X.$$

此时, 对应于问题 (3.1) 的线性项系数 $G = 0$.

事实上, 例 3.2 可以看成是 Kohn-Sham 总能量极小化问题 (2.5) 的一个简化特例.

3.2 乘子校正算法

在本节中, 我们提出正交约束优化问题的一阶算法框架. 其中包含函数值下降步以及乘子校正步.

3.2.1 最优性条件

根据 2.2 小节的讨论, 首先我们回顾问题 (3.1) 的一阶最优性条件 (定理 2.3) 如下,

$$\begin{cases} (I_n - XX^\top)\nabla f(X) &= 0; & (\text{次稳定性}) \\ X^\top \nabla f(X) &= \nabla f(X)^\top X; & (\text{对称性}) \\ X^\top X &= I_p. & (\text{可行性}) \end{cases} \tag{3.2}$$

由定理 2.2 和 (2.31) 式可知, 当 $X \in \mathcal{S}_{n,p}$, 即 $X^\top X = I_p$ 时, 上述一阶最优性条件等价于

$$\begin{aligned} 0 &= \left\|\mathrm{grad}_2\, f(X)\right\|_{\mathrm{F}}^2 = \left\|\nabla f(X) - X\nabla f(X)^\top X\right\|_{\mathrm{F}}^2 \\ &= \left\|\nabla f(X) - XX^\top \nabla f(X)\right\|_{\mathrm{F}}^2 + \left\|X^\top \nabla f(X) - \nabla f(X)^\top X\right\|_{\mathrm{F}}^2. \end{aligned} \tag{3.3}$$

上式右端第一项恰好为一阶最优性条件 (3.2) 中次稳定性的违反度, 也可看做是 $\nabla f(X)$ 在 X 的零空间的投影. 另一方面, 由推论 2.1 可知, 第二项为 Lagrange 乘子对称性的违反度. 一般来讲, 我们可以设计下降算法, 使得迭代点列的每一步都满足正交性, 并且使得 (3.3) 式的右端第一项逐渐趋于 0. 在此基础上, 算法设计的难点就变成了如何使得新迭代点可行并且第二项也逐渐消失或者始终为 0.

3.2.2 校正步和算法框架

受到关系式 (3.3) 的启发, 为了使得 $\nabla f(X)$ 在 X 的零空间的投影等于 0, 我们可以采用如下的两个步骤. 首先, 从当前点开始, 选取一个可行试探点使得函数值以投影梯度范数的量级下降. 接着以此试探点为基, 搜索使得对称性成立并不使函数增加的点为下一个迭代点. 重复此过程直至收敛. 在这两个步骤中, 我们希望可行性始终能够得到满足. 由此, 我们提出了如下的算法框架.

假设当前迭代点为 X^k, 在第一步中, 我们找到一个中间点 $\bar{X} \in \mathcal{S}_{n,p}$ 使得如下的函数值充分下降条件满足,

$$f(X^k) - f(\bar{X}) \geq C_1 \cdot \left\|(I_n - X^k X^{k\top})\nabla f(X^k)\right\|_{\mathrm{F}}^2, \tag{3.4}$$

其中 $C_1 > 0$ 是一个正常数. 不等式 (3.4) 的右边度量了在点 X^k 处, 一阶最优性条件 (3.2) 次稳定性的违反度.

尽管中间点 $\bar{X} \in \mathcal{S}_{n,p}$ 满足不等式 (3.4), 但它并不满足 (3.2) 式中的乘子对称性. 在第二部分, 我们考虑构造如下校正步使得对称性得以满足并且不增加函数值.

由假设 3.1, 我们有

$$\bar{X}^\top \nabla f(\bar{X}) = \bar{X}^\top H(\bar{X})\bar{X} + \bar{X}^\top G. \tag{3.5}$$

其中 $\bar{X}^\top H(\bar{X})\bar{X}$ 是对称的. 因此, 当 $\bar{X}^\top G$ 对称时, 下一步迭代 X^{k+1} 我们可以取 $\bar{X}$. 否则, 我们需要使得迭代点 X^{k+1} 满足对称性条件 ${X^{k+1}}^\top G = G^\top X^{k+1}$. 在这种情况下, 为实现对称性, 我们使用旋转校正, 也就是 $X^{k+1} = -\bar{X}UT^\top$, 这里正交矩阵 U 和 T 来自于 p 阶矩阵

$$\bar{X}^\top G = U\Lambda T^\top \tag{3.6}$$

的奇异值分解.

事实上我们发现, 这样一个校正步实际上是去寻找一个 p 阶矩阵 Q 使得目标函数 $f(\bar{X}Q)$ 极小化, 即

$$\min_{Q\in\mathcal{S}_{p,p}} f(\bar{X}Q),$$

并且令 $X^{k+1} = \bar{X}Q^*$. 由假设 3.1, 上述问题等价于

$$\min_{Q\in\mathcal{S}_{p,p}} \mathrm{tr}((\bar{X}Q)^\top G).$$

根据注 3.1 的讨论, 我们可以很容易得到上述问题的全局极小值点 $Q^* = -UT^\top$. 因此, 我们取下一步迭代为

$$X^{k+1} = \begin{cases} \bar{X}, & \text{当 } \bar{X}^\top G = G^\top \bar{X}; \\ -\bar{X}UT^\top, & \text{否则.} \end{cases} \tag{3.7}$$

值得说明的是这个校正步使得矩阵 ${X^{k+1}}^\top G$ 的所有特征值为负. 当 $\bar{X}^\top G$ 的所有特征值为负时, 我们无需进行校正.

对于这样的校正步 X^{k+1}, 我们有如下性质.

引理 3.1. 假设 $\bar{X} \in \mathcal{S}_{n,p}$. X^{k+1} 由 (3.7) 式计算得到, 这里 U 和 T 由校正步 (3.6) 式决定. 则我们有 $X^{k+1} \in \mathcal{S}_{n,p}$ 并且 ${X^{k+1}}^\top \nabla f(X^{k+1})$ 满足对称性. 进一步, 我们有

$$8\theta\left(f(\bar{X}) - f(X^{k+1})\right) \geq \left\|\bar{X}^\top \nabla f(\bar{X}) - \nabla f(\bar{X})^\top \bar{X}\right\|_{\mathrm{F}}^2, \tag{3.8}$$

其中

$$\theta := ||G||_2. \tag{3.9}$$

证明. 首先, X^{k+1} 的正交性和 ${X^{k+1}}^\top \nabla f(X^{k+1})$ 的对称性都都可由公式 (3.7) 直接推出. 接下来, 我们证明不等式 (3.8). 如果 $\theta = 0$, 这也就意味着 $\nabla f(X) = H(X)X$, 由此 $\bar{X}^\top \nabla f(\bar{X})$ 对称, 故 (3.8) 式成立.

若 $\theta \neq 0$. 一方面, 根据假设 3.1, 我们有

$$\begin{aligned} f(\bar{X}) - f(X^{k+1}) &= h(\bar{X}) + \mathrm{tr}(G^\top \bar{X}) - h(X^{k+1}) - \mathrm{tr}(G^\top X^{k+1}) \\ &= \mathrm{tr}(G^\top \bar{X} - G^\top X^{k+1}) \\ &= \mathrm{tr}(U\Lambda T^\top + \Lambda) = \mathrm{tr}(B + \Lambda), \end{aligned} \tag{3.10}$$

其中 $B = (\Lambda T^\top U + U^\top T \Lambda)/2$. 另一方面,

$$\begin{aligned} & \left\| \bar{X}^\top \nabla f(\bar{X}) - \nabla f(\bar{X})^\top \bar{X} \right\|_{\mathrm{F}}^2 = \left\| \bar{X}^\top G - G^\top \bar{X} \right\|_{\mathrm{F}}^2 \\ = & \left\| U\Lambda T^\top - T\Lambda U^\top \right\|_{\mathrm{F}}^2 = 2\mathrm{tr}(\Lambda^2) - 2\mathrm{tr}(\Lambda T^\top U \Lambda T^\top U) \\ = & 4\mathrm{tr}(\Lambda^2 - B^2), \end{aligned} \tag{3.11}$$

其中最后一个等式利用了

$$\mathrm{tr}(B^2) = \frac{1}{2}\mathrm{tr}(\Lambda^2) + \frac{1}{2}\mathrm{tr}(\Lambda T^\top U \Lambda T^\top U).$$

进一步, 我们有

$$\begin{aligned} & \mathrm{tr}(\Lambda^2 - B^2) \leq \sum_{i=1}^{p} (\Lambda_{ii}^2 - B_i^\top B_i) \leq \sum_{i=1}^{p} (\Lambda_{ii}^2 - B_{ii}^2) = \sum_{i=1}^{p} (\Lambda_{ii} - B_{ii})(\Lambda_{ii} + B_{ii}) \\ \leq & \sum_{i=1}^{p} 2\Lambda_{ii}(\Lambda_{ii} + B_{ii}) \leq 2||\Lambda||_2 \cdot \sum_{i=1}^{p} (\Lambda_{ii} + B_{ii}) = 2||\Lambda||_2 \cdot \mathrm{tr}(\Lambda + B) \\ \leq & 2||G||_2 \cdot \mathrm{tr}(\Lambda + B) = 2\theta \cdot \mathrm{tr}(\Lambda + B). \end{aligned} \tag{3.12}$$

这里, 第三个不等式利用了性质

$$|B_{ii}| = \Lambda_{ii} \cdot |T_i^\top U_i| \leq \Lambda_{ii}.$$

结合 (3.10)-(3.12) 式, 引理得证. □

综上所述, 完整的乘子校正算法框架如下 (算法 3). 由于一阶最优性条件 (3.2) 的对称性和可行性在每一步成立, 故我们取 $\left\| c(X) \right\|_{\mathrm{F}} \leq \epsilon$ 作为算法的终止准则, 其中

$$c(X) := (I_n - XX^\top)\nabla f(X). \tag{3.13}$$

算法 3: 正交约束优化问题的一阶算法框架

令终止误差 $\epsilon > 0$; 初始化: $X^0 \in \mathcal{S}_{n,p}$; 令 $k := 0$

while $||c(X^k)||_{\mathrm{F}} > \epsilon$ **do**

  基于 X^k, 寻找可行点 $\bar{X}$ 满足函数值充分下降条件 (3.4);

  基于 $\bar{X}$, 由乘子校正公式 (3.7) 计算可行点 X^{k+1};

  令 $k := k + 1$.

返回 X^k.

3.3 从迭代点 X^k 到 $\bar{X}$ 的算法

在上节, 我们提出了一个新的算法框架, 但是我们并没有具体给出满足函数值充分下降条件 (3.4) 的 $\bar{X}$ 的取法. 在本节, 我们提出两类具体的方法用以实现算法 3 中的第 3 步. 前两小节, 我们介绍第一类基于欧式空间的梯度下降法. 接着我们引入第二类以列为块的块坐标下降法. 最后列出了我们的算法与已有算法的每步计算量比较.

3.3.1 梯度类方法

在欧式空间使函数值下降的一个直观方法是取负梯度下降方向进行迭代. 在本文的第 1 章中, 我们详细讨论了此类方法. 然而, 在本章的问题中, 由欧式梯度下降步得到的试探点并不满足正交约束. 因此, 在本节我们考虑两种不同的方法, 使得试探步可以被 “拉回” 到 Stiefel 流形上. 与此同时, 我们证明了每一种方法都满足算法 3 中第 3 步的条件.

两种方法都是基于如下的观察.

引理 3.2. *对任意的 $Y \in \mathcal{B}_{X,\tau} := \mathcal{B}(X - \tau\nabla f(X), \tau||\nabla f(X)||_{\mathrm{F}})$, 其中 $\tau \in (0, \rho^{-1})$, 下式成立,*

$$f(X) - f(Y) \geq \frac{1 - \rho\tau}{2\tau} \cdot ||X - Y||_{\mathrm{F}}^2. \tag{3.14}$$

证明. 对任意的 $Y \in \mathcal{B}_{X,\tau}$, 我们有

$$\left\langle Y - X, Y - X + 2\tau\nabla f(X) \right\rangle \leq 0,$$

由此得到

$$\begin{aligned} f(Y) &\leq f(X) + \langle Y - X, \nabla f(X) \rangle + \frac{\rho}{2}||Y - X||_{\mathrm{F}}^2 \\ &= f(X) + \frac{1}{2\tau} \cdot \langle Y - X, Y - X + 2\tau \nabla f(X) \rangle - \frac{\tau^{-1} - \rho}{2} \cdot ||Y - X||_{\mathrm{F}}^2 \\ &\leq f(X) - \frac{\tau^{-1} - \rho}{2} \cdot ||Y - X||_{\mathrm{F}}^2. \end{aligned}$$

因此, 引理得证. □

为了更直观的展示可行域, 当前迭代点, 梯度步和辅助球的关系, 我们作图 3.1 如下. 引理 3.2 告诉我们, 在辅助球 $\mathcal{B}_{X,\tau}$ 内的任意一点都满足函数值的下降

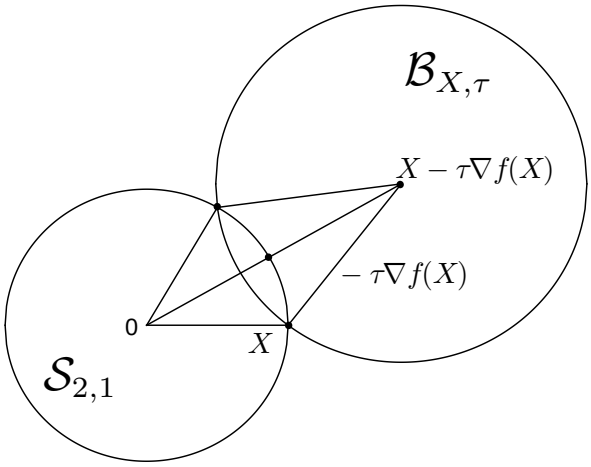


图 3.1 梯度类方法

Figure 3.1 Gradient type method

性 (3.14). 同时, 我们又要求新的迭代点满足正交性, 即迭代点在球面 $\mathcal{S}_{2,1}$ 上. 因此, 我们可以很自然从 $\mathcal{B}_{X,\tau}$ 与球面 $\mathcal{S}_{2,1}$ 相交的区域选取下一步的迭代点. 考虑到计算的有效性, 我们采用图中所示的两类方法, 并将其推广到一般的 Stiefel 流形 $\mathcal{S}_{n,p}$.

3.3.1.1 梯度反射法

可行试探点的第一种可能选择是当前迭代点 X^k 沿 $X^k - \tau\nabla f(X^k)$ 的零空间反射, 事实上, 这个点可以通过 Householder 变换计算得到.

$$\text{GR:} \quad \begin{cases} V = X^k - \tau\nabla f(X^k), \quad \text{取定} \, \tau \in (0, \rho^{-1}); \\ \bar{X}_{\mathrm{GR}} = (-I_n + 2V(V^\top V)^\dagger V^\top)X^k. \end{cases} \tag{3.15}$$

从示意图 3.1 我们可以观察到, $\bar{X}_{\mathrm{GR}}$ 是 X 的镜面反射点, 如果在算法 3 的步 3 采用 (3.15) 式得到 $\bar{X} := \bar{X}_{\mathrm{GR}}$, 那么我们称此算法为梯度反射法 (Gradient Reflection, 简称为 GR).

接下来我们证明由 (3.15) 式定义的中间点 $\bar{X}_{\mathrm{GR}}$ 是可行点, 并且满足函数值的充分下降条件 (3.4).

引理 3.3. *令 $X^k \in \mathcal{S}_{n,p}$ 且 $\bar{X}_{\mathrm{GR}}$ 由 (3.15) 式定义. 则 $\bar{X}_{\mathrm{GR}} \in \mathcal{S}_{n,p}$ 并且我们有*

$$f(X^k) - f(\bar{X}_{\mathrm{GR}}) \geq \frac{2(\tau^{-1} - \rho)}{(\tau^{-1} + \rho + \theta)^2} \cdot \left\| (I_n - X^k X^{k\top}) \nabla f(X^k) \right\|_{\mathrm{F}}^2 \tag{3.16}$$

成立. 其中 $\tau \in (0, \rho^{-1})$, ρ 和 θ 分别由假设 3.1 和等式 (3.9) 得到.

证明. 为了简便起见, 在证明中我们省略上标 k, 并用 X 表示 X^k. 首先, 通过计算, 我们有

$$\bar{X}_{\mathrm{GR}}^{\top} \bar{X}_{\mathrm{GR}} = X^{\top} \left(-I_n + 2V(V^{\top}V)^{\dagger} V^{\top} \right)^{\top} \left(-I_n + 2V(V^{\top}V)^{\dagger} V^{\top} \right) X = I_p,$$

即 $\bar{X}_{\mathrm{GR}} \in \mathcal{S}_{n,p}$.

令 $RSQ^{\top}$ 表示矩阵 V 的奇异值分解. 当 $S = 0$ 时, 我们有 $X = \tau \nabla f(X)$, 由此推出 $(I_n - XX^{\top}) \nabla f(X) = 0$ 并且不等式 (3.16) 成立. 接下来, 我们考虑 $S \neq 0$ 的情况. 根据 V 的分解我们有

$$\begin{aligned}
& \|\bar{X}_{\mathrm{GR}} - X\|_{\mathrm{F}} = 2 \left\| \left(I_n - V(V^{\top}V)^{\dagger} V^{\top} \right) X \right\|_{\mathrm{F}} \\
= \;& 2 \left\| (I_n - RR^{\top}) X \right\|_{\mathrm{F}} = 2\sqrt{p - \|R^{\top}X\|_{\mathrm{F}}^2} = 2 \left\| (I_n - XX^{\top}) R \right\|_{\mathrm{F}} \\
\geq \;& 2 \left\| (I_n - XX^{\top}) VQS^{\dagger} \right\|_{\mathrm{F}} \geq 2 \left\| (I_n - XX^{\top}) VQ \right\|_{\mathrm{F}} \cdot \lambda_{\min}^{+}(S^{\dagger}) \\
= \;& 2 \left\| (I_n - XX^{\top}) V \right\|_{\mathrm{F}} / \|S\|_2 = 2\tau \left\| c(X) \right\|_{\mathrm{F}} / \|V\|_2 \\
\geq \;& \frac{2}{\tau^{-1} + \rho + \theta} \left\| c(X) \right\|_{\mathrm{F}},
\end{aligned} \tag{3.17}$$

其中 $\lambda_{\min}^{+}$ 表示最小的正特征值, $c(X)$ 由 (3.13) 式定义. 根据 $RS = VQ$, 对于任意的 j 满足 $S_{jj} = 0$, 我们有 VQ 的第 j 列的所有元素都为 0, 故第二个不等式成立. (3.17) 的最后一个不等式由

$$\begin{aligned}
\|\nabla f(X)\|_2 &\leq \|H(X)X\|_2 + \|G\|_2 \leq \rho + \theta, \\
\|V\|_2 &\leq \|X\|_2 + \tau \|\nabla f(X)\|_2 \leq 1 + \tau(\rho + \theta)
\end{aligned} \tag{3.18}$$

得到. 令 $Y = \bar{X}_{\mathrm{GR}}$, 并将不等式 (3.17) 代入引理 3.2 的 (3.14) 式中, 我们得到

$$\begin{aligned}
f(X) - f(\bar{X}_{\mathrm{GR}}) &\geq \frac{4}{(\tau^{-1} + \rho + \theta)^2} \cdot \frac{\tau^{-1} - \rho}{2} \cdot \|(I_n - XX^{\top}) \nabla f(X)\|_{\mathrm{F}}^2 \\
&= \frac{2(\tau^{-1} - \rho)}{(\tau^{-1} + \rho + \theta)^2} \cdot \|(I_n - XX^{\top}) \nabla f(X)\|_{\mathrm{F}}^2,
\end{aligned} \tag{3.19}$$

由此引理得证. □

3.3.1.2 梯度投影法

观察示意图 3.1 可知, 另一个可能的可行试探点是 $X^k - \tau\nabla f(X^k)$ 在 Stiefel 流形上的投影点, 也就是

$$\text{GP:} \quad \begin{cases} V = X^k - \tau\nabla f(X^k), \quad \text{取定 } \tau \in (0, \rho^{-1}); \\ \bar{X}_{\text{GP}} = \mathcal{P}_{\mathcal{S}_{n,p}}(V). \end{cases} \tag{3.20}$$

其中投影算子 $\mathcal{P}_{\mathcal{S}_{n,p}}$ 由性质 2.7 可得. 如果在算法 3 的步骤 3 选用 (3.20) 得到 $\bar{X} := \bar{X}_{\text{GP}}$, 那么我们称此算法为梯度投影法 (Gradient Projection, 简称为 GP). 同样的, 我们可以证明 $\bar{X}_{\text{GP}}$ 满足可行性以及函数值的充分下降性.

引理 3.4. *令 $X^k \in \mathcal{S}_{n,p}$ 且 $\bar{X}_{\text{GP}}$ 由 (3.20) 定义. 则 $\bar{X}_{\text{GP}} \in \mathcal{S}_{n,p}$ 并且我们有*

$$f(X^k) - f(\bar{X}_{\text{GP}}) \geq \frac{\tau^{-1} - \rho}{2(\tau^{-1} + \rho + \theta)^2} \cdot \left\|(I_n - X^k X^{k\top})\nabla f(X^k)\right\|_{\text{F}}^2, \tag{3.21}$$

成立. 其中 $\tau \in (0, \rho^{-1})$, ρ 和 θ 分别由假设 3.1 和 (3.9) 式得到.

证明. 第一部分的证明由引理 3.3 同理可得. 接下来, 我们证明 (3.21).

利用奇异值分解 $V = RSQ^\top$ 和 (3.17) 的第二个等式, 我们得到

$$\begin{aligned} & ||\bar{X}_{\text{GP}} - X^k||_{\text{F}}^2 - \frac{1}{4}||\bar{X}_{\text{GR}} - X||_{\text{F}}^2 = ||RQ^\top - X^k||_{\text{F}}^2 - ||(I_n - RR^\top)X^k||_{\text{F}}^2 \\ = \;& \text{tr}(I_p) - 2\text{tr}(QR^\top X^k) + \text{tr}(I_p) - \text{tr}(I_p) + 2\text{tr}(X^{k\top} RR^\top X^k) - ||R^\top X^k||_{\text{F}}^2 \\ = \;& p - 2\text{tr}(QR^\top X^k) + ||R^\top X^k||_{\text{F}}^2 = ||Q^\top - R^\top X^k||_{\text{F}}^2 \geq 0, \end{aligned}$$

由此推出

$$||\bar{X}_{\text{GP}} - X^k||_{\text{F}} \geq \frac{1}{2}||\bar{X}_{\text{GR}} - X^k||_{\text{F}} \geq \frac{1}{\tau^{-1} + \rho + \theta}||c(X)||_{\text{F}}.$$

进一步, 利用证明 (3.19) 式同样的思路, 我们可以得到 (3.21) 式. 由此引理得证. □

3.3.2 以列为块的块坐标下降方法

作为另一类常用的一阶方法, 在第 1 章中, 我们详细介绍了块坐标下降法. 对于正交约束优化问题而言, 自然的想法是划分变量的每一列为一块. 然而, 在非凸问题中, 已有的结果并不能保证块变量相互耦合的块坐标下降法的收敛性. 因此, 我们值得去研究正交约束优化问题的以列为块的块坐标下降方法. 在本小节, 我

们考虑用块坐标下降法求解算法 3 的第 3 步, 并给出有效求解子问题的方法. 除此之外, 我们还证明了新的块坐标下降法满足算法 3 的条件.

利用算法 2 中块坐标下降的思想, 在原正交约束优化问题 (3.1) 中, 我们固定变量 X 的 $p-1$ 列, 只留下第 i 列作为未知变量, 由此得到如下的块坐标下降子问题

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & f_{i,X}(x) \\ \text{s.t.} \quad & ||x||_2 = 1, \\ & X_{\bar{i}}^\top x = 0, \end{aligned} \tag{3.22}$$

其中 $f_{i,X}(x) := f(X_{i,x})$, $X_{i,x}$ 表示矩阵 X 的第 i 列是变量, 其余列都固定, $X_{\bar{i}}$ 则表示矩阵 $[X_1,\ldots,X_{i-1},X_{i+1},\ldots,X_p]$.

假设我们能够得到上述子问题的解或是找到一个可行点使得 $f_{i,X}(X_i)$ 满足函数值的充分下降性, 则我们可以利用这个可行点以 Gauss-Seidel 列更新的方式进行迭代. 具体来讲, 如果 X 是当前迭代点, 则试探点 $\bar{X}$ 可由如下的块坐标下降法计算得到. 这里, X_{i,x^+} 表示矩阵 X 的第 i 列替换为 x^+, 即 $X_{i,x^+} = [X_1,\ldots,X_{i-1},x^+,X_{i+1},\ldots,X_p]$.

算法 4: 以列为块的块坐标下降法 (CBCD)

1 令 $W^0 = X, i := 1$;

2 **while** $i \le p$ **do**

3 令 W^{i-1} 替换 X, 求解子问题 (3.22), 得到可行点 x^+, 使其满足如下的函数值充分下降条件和渐进小的步长保证

$$f_{i,W^{i-1}}(X_i) - f_{i,W^{i-1}}(x^+) \ge k_1||X_i - x^+||_2^2, \tag{3.23}$$

$$||X_i - x^+||_2 \ge k_2 \left\|(I_n - W^{i-1}W^{i-1\top})\nabla f_{i,W^{i-1}}(X_i)\right\|_2; \tag{3.24}$$

令 $W^i = W_{i,x^+}^{i-1}, i := i+1$;

4 **返回** $\bar{X} = W^p$.

注 3.2. *算法 4 实际上提供了一个以列为块的块坐标下降循环算法, 也就是列的更新是以序列周期循环进行的 ($i = 1,\ldots,p$). 同样的, 我们可以实现注 1.2 中提到的贪婪, 随机选取 (有放回的), 随机排列 (无放回的) 等块坐标下降法中经典的更新方式. 然而, 在数值实验一节, 我们展示了这些策略并不能帮助提升序列循环块坐标下降方法的数值表现. 因此, 我们省略对这些不同策略的具体分析.*

在我们证明算法 4 能够找到满足算法 3 第 3 步的 $\bar{X}$ 之前, 我们需要回答两个问题: 第一, 我们能不能以较低计算量得到一个解或是可行点满足函数值充分下降条件和渐进小的步长保证 (3.23)-(3.24) ? 第二, 算法 4 是否提供了一个满足函数充分下降条件 (3.4) 的可行点? 接下来, 我们分别回答这两个问题.

3.3.2.1 求解块坐标下降子问题

在本小节, 我们讨论如何有效的得到子问题 (3.22) 的一个可行试探点. 我们注意到子问题 (3.22) 的第二个约束限制了变量 x 只能在 $X_{\bar{i}}$ 的零空间里. 因此, 我们可以利用变量替换 $x = (I_n - X_{\bar{i}}X_{\bar{i}}^\top)x$ 来减少约束.

首先, 我们有 $X_{\bar{i}}^\top x = 0$ 成立当且仅当 $x = (I_n - X_{\bar{i}}X_{\bar{i}}^\top)x$. 因此, 子问题 (3.22) 等价于如下问题

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & f_{i,X}((I_n - X_{\bar{i}}X_{\bar{i}}^\top)x) \\ \text{s.t.} \quad & \left\|(I_n - X_{\bar{i}}X_{\bar{i}}^\top)x\right\|_2 = 1. \end{aligned} \tag{3.25}$$

进一步, 当问题 (3.25) 限制在 $X_{\bar{i}}$ 的零空间里, 我们有如下结论.

性质 3.1. *当 $X_{\bar{i}}^\top x = 0$ 对任意 $x \in \mathcal{D}$ 都成立. 则限制在子空间 $\mathcal{D}$ 的问题 (3.22) 和如下的球约束问题等价.*

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & q_i(x) := f_{i,X}((I_n - X_{\bar{i}}X_{\bar{i}}^\top)x) \\ \text{s.t.} \quad & ||x||_2 = 1, \\ & x \in \mathcal{D}. \end{aligned} \tag{3.26}$$

证明. 对于任意的 $x \in \mathcal{D}$, 我们有 $x = (I_n - X_{\bar{i}}X_{\bar{i}}^\top)x$. 故问题 (3.26) 和限制在子空间 $\mathcal{D}$ 的问题 (3.22) 等价. 利用问题 (3.25) 和 (3.22) 的等价性, 性质成立. □

性质 3.1 告诉我们, 如果可以找到一个合适的子空间 $\mathcal{D}$, 则我们可以通过求解问题 (3.26), 得到一个子问题 (3.22) 的满足函数值充分下降条件的可行点.

我们注意到 X_i 和 $\nabla q_i(X_i) = (I_n - X_{\bar{i}}X_{\bar{i}}^\top)\nabla f_{i,X}((I_n - X_{\bar{i}}X_{\bar{i}}^\top)X_i)$ 都在 $X_{\bar{i}}$ 的零空间中. 因此, 子空间 $\operatorname{span}\left\{X_i, \nabla q_i(X_i)\right\}$ 的任意点都满足正交性 $X_{\bar{i}}^\top x = 0$. 由此, 子空间 $\operatorname{span}\left\{X_i, \nabla q_i(X_i)\right\}$ 是满足性质 3.1 的子空间 $\mathcal{D}$ 的一个选择. 我们可以把限制在子空间

$$\mathcal{D} = \operatorname{span}\left\{X_i, \nabla q_i(X_i)\right\}$$

的子问题 (3.26) 看做是原始正交约束优化问题 (3.1) 的一个特例, 这里 $p = 1$. 此时, 我们推荐使用上节中的梯度反射法 (3.15) 和梯度投影法 (3.20) 来计算 x^+. 事实上, 在实际计算中, 我们可以非精确的求解子问题, 甚至只采用一步迭代.

我们可以验证如果子问题 (3.26) 采用梯度反射法 (3.15) 或梯度投影法 (3.20), 则得到的解 x^+ 都满足函数值的充分下降性 (3.23) 和渐进小的步长保证 (3.24).

引理 3.5. 令 $x^+ = (-1 + 2v(v^\top v)^{-1}v^\top)X_i$ 或者 $x^+ = (v^\top v)^{-\frac{1}{2}}v$, 这里 $v = X_i - \tau \cdot \nabla q_i(X_i)$, $\tau \in (0, \rho^{-1})$. 则 x^+ 满足子问题 (3.22) 的约束并且满足条件 (3.23) 和 (3.24).

引理 3.5 的证明可由引理 3.2, 3.3, 3.4 和 $I_n - XX^\top = (I_n - X_iX_i^\top)(I_n - X_{\bar{i}}X_{\bar{i}}^\top)$ 直接得到.

注 3.3. 特别地, 如果 $f_{i,X}$ 是二次的, 则限制在子空间 $\text{span}\left\{X_i, \nabla q_i(X_i)\right\}$ 的子问题 (3.26) 等价于求解一个四次方程的根. 这些根可由显示表达式得到. 在这种情况下, 限制在子空间 $\text{span}\left\{X_i, \nabla q_i(X_i)\right\}$ 的子问题 (3.26) 的全局极小值点可作为 x^+ 的另一种选择. 具体来讲, 当目标函数 $f_{i,X}$ 是二次的, 我们有如下子问题

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & \tfrac{1}{2}x^\top \tilde{A}x + \tilde{g}^\top x \\ \text{s.t.} \quad & ||x||_2 = 1, \\ & x \in \text{span}\left\{X_i, \nabla q_i(X_i)\right\}. \end{aligned}$$

等价地, 我们得到二维球面上的二次函数极小化问题, 也就等价于四次方程求根.

3.3.2.2 函数值充分下降性

在这一小节, 我们证明由算法 4 得到的 $\bar{X}$ 是问题 (3.1) 的一个可行点, 并且满足函数值充分下降条件 (3.4).

引理 3.6. 令 $X \in \mathcal{S}_{n,p}$ 且 $\bar{X}$ 由算法 *4* 计算得到. 则我们有 $\bar{X} \in \mathcal{S}_{n,p}$ 并且

$$f(X) - f(\bar{X}) \geq \frac{k_1 k_2^2}{\left(1 + (p-1)k_2\left((1+\sqrt{2})\rho + \sqrt{2}\theta\right)\right)^2} \left\|(I_n - XX^\top)\nabla f(X)\right\|_{\text{F}}^2. \quad (3.27)$$

证明. $\bar{X}$ 的可行性可由 Gauss-Seidel 型更新和子问题 (3.22) 的约束直接得到. 接下来我们证明第二部分. 首先, 我们有

$$\begin{aligned} f(X) - f(\bar{X}) &= f(W^0) - f(W^p) \\ &= \sum_{i=1}^{p}\left(f(W^{i-1}) - f(W^i)\right) \\ &= \sum_{i=1}^{p}\left(f_{i,W^{i-1}}(W_i^{i-1}) - f_{i,W^{i-1}}(W_i^i)\right), \end{aligned} \quad (3.28)$$

并且

$$f_{i,W^{i-1}}(W_i^{i-1}) - f_{i,W^{i-1}}(W_i^i) \geq k_1 k_2^2 \left\| (I_n - W^{i-1}W^{i-1\top})\nabla f_{i,W^{i-1}}(W_i^{i-1}) \right\|_2^2. \quad (3.29)$$

利用梯度 (3.18) 的 Lipschitz 连续性和有界性, 我们得到

$$\begin{aligned}
& \left\| (I_n - W^{i-1}W^{i-1\top})\nabla f_{i,W^{i-1}}(W_i^{i-1}) \right\|_2 \\
\geq\ & \left\| (I_n - W^0 W^{0\top})\nabla f_{i,W^{i-1}}(W_i^{i-1}) \right\|_2 - \left\| (W^{i-1}W^{i-1\top} - W^0 W^{0\top})\nabla f_{i,W^{i-1}}(W_i^{i-1}) \right\|_2 \\
\geq\ & (I_n - XX^\top)\nabla f_{i,W^{i-1}}(X_i)_2 - \sum_{j=1}^{i-1} \left\| W^j W^{j\top} - W^{j-1}W^{j-1\top} \right\|_{\mathrm{F}} \cdot \left\| \nabla f_{i,W^{i-1}}(W_i^{i-1}) \right\|_2 \\
\geq\ & \left\| (I_n - XX^\top)\nabla f_{i,W^0}(X_i) \right\|_2 - \left\| (I_n - XX^\top)(\nabla f_{i,W^0}(X_i) - \nabla f_{i,W^{i-1}}(X_i)) \right\|_2 \\
& -(\rho+\theta)\sum_{j=1}^{i-1} \left(\sqrt{2 - 2\left(W_j^{j\top} W_j^{j-1} \right)^2} \right) \\
\geq\ & \left\| (I_n - XX^\top)\nabla f_{i,X}(X_i) \right\|_2 - \nabla f_{i,W^0}(X_i) - \nabla f_{i,W^{i-1}}(X_i)_2 \\
& -\sqrt{2}(\rho+\theta)\cdot \sum_{j=1}^{i-1} \left(\sqrt{2 - 2\left(W_j^{j\top} W_j^{j-1} \right)} \right) \\
\geq\ & (I_n - XX^\top)\nabla f_{i,X}(X_i)_2 - \rho \cdot \left\| W^0 - W^{i-1} \right\|_2 - \sqrt{2}(\rho+\theta)\sum_{j=1}^{i-1} \left\| W_j^j - W_j^{j-1} \right\|_2 \qquad (3.30) \\
\geq\ & \left\| (I_n - XX^\top)\nabla f_{i,X}(X_i) \right\|_2 - ((1+\sqrt{2})\rho + \sqrt{2}\theta)\sqrt{k_1}^{-1} \sum_{j=1}^{i-1} \sqrt{f_{j,W^{j-1}}(W_j^{j-1}) - f_{j,W^{j-1}}(W_j^j)},
\end{aligned}$$

其中第三个不等式利用了 $\left| W_i^{i-1\top} W_i^i \right| \leq 1$. 结合 (3.29) 式, 我们有

$$\begin{aligned}
& \sqrt{f_{i,W^{i-1}}(W_i^{i-1}) - f_{i,W^{i-1}}(W_i^i)} \geq \sqrt{k_1}k_2 \left\| (I_n - W^{i-1}W^{i-1\top})\nabla f_{i,W^{i-1}}(W_i^{i-1}) \right\|_2 \qquad (3.31) \\
\geq\ & \sqrt{k_1}k_2 \left\| (I_n - XX^\top)\nabla f_{i,X}(X_i) \right\|_2 - k_2((1+\sqrt{2})\rho + \sqrt{2}\theta)\sum_{j=1}^{i-1} \sqrt{f_{j,W^{j-1}}(W_j^{j-1}) - f_{j,W^{j-1}}(W_j^j)}.
\end{aligned}$$

令 $\delta_j := \sqrt{f_{j,W^{j-1}}(W_j^{j-1}) - f_{j,W^{j-1}}(W_j^j)}$, $c := k_2((1+\sqrt{2})\rho + \sqrt{2}\theta)$, 并代入关系式 (3.31), 我们得到

$$\left(\delta_i + c\sum_{j=1}^{i-1} \delta_j \right)^2 \leq (1 + (i-1)c)\,\delta_i^2 + \sum_{j=1}^{i-1} c\,(1 + (i-1)c)\,\delta_j^2,$$

由此推出

$$(1 + (i-1)c)\,\delta_i^2 + \sum_{j=1}^{i-1} c\,(1 + (i-1)c)\,\delta_j^2 \geq k_1 k_2^2 \left\| (I_n - XX^\top)\nabla f_{i,X}(X_i) \right\|_2^2. \quad (3.32)$$

将不等式 (3.32) 从 $i=1$ 到 p 进行累加, 再结合 (3.28) 式, 我们得到

$$\begin{aligned}&\left(1+(p-1)k_2((1+\sqrt{2})\rho+\sqrt{2}\theta)\right)^2\sum_{i=1}^{p}\left(\sqrt{f_{i,W^{i-1}}(W_i^{i-1})-f_{i,W^{i-1}}(W_i^i)}\right)^2\\ \geq\ &\sum_{i=1}^{p}k_1k_2^2\cdot\left\|(I_n-XX^\top)\nabla f_{i,X}(X_i)\right\|_2^2=k_1k_2^2\cdot\left\|(I_n-XX^\top)\nabla f(X)\right\|_{\mathrm{F}}^2,\end{aligned}\tag{3.33}$$

也就是

$$f(X)-f(\bar{X})\geq\frac{k_1k_2^2}{\left(1+(p-1)k_2\left((1+\sqrt{2})\rho+\sqrt{2}\theta\right)\right)^2}\cdot\left\|(I_n-XX^\top)\nabla f(X)\right\|_{\mathrm{F}}^2,$$

由此引理得证. □

引理 3.6 的一个副产物是对以列为块的块坐标下降法, 如下的渐进小步长保证性质得到满足.

推论 3.1. *令 $X\in\mathcal{S}_{n,p}$ 且 $\bar{X}$ 由算法 4 计算得到. 则我们有*

$$||X-\bar{X}||_{\mathrm{F}}\geq\frac{k_2}{1+(p-1)k_2((1+\sqrt{2})\rho+\sqrt{2}\theta)}\cdot||(I_n-XX^\top)\nabla f(X)||_{\mathrm{F}}.\tag{3.34}$$

证明. 利用条件 (3.24) 和 (3.30) 的倒数第二个不等式, 采用不等式 (3.31) 和 (3.33) 同样的推导方式, 推论得证. □

3.3.3 计算量比较

在第 2 章 2.3 小节中, 我们详细介绍了一些针对正交约束优化问题的已有算法. 在本小节, 我们比较新提出的算法框架与已有算法的每步计算量. 首先, 我们约定基础的线性代数操作 (BLAS) 的计算量如下. 给定 $A\in\mathbb{R}^{n\times n}$, B_1, $B_2\in\mathbb{R}^{n\times p}$, S_1, $S_2\in\mathbb{R}^{p\times p}$ 和 $x\in\mathbb{R}^n$, 计算矩阵乘法 $B_1^\top B_2$, $B_1^\top B_1$, B_1S_1, 和 S_1S_2 分别需要 $2np^2$, np^2+np, $2np^2$ 和 $2p^3$ 个浮点运算. 计算 A^{-1} 和 S^{-1} 分别需要 $8n^3/3$ 和 $8p^3/3$ 个浮点运算. 计算矩阵向量乘法 Ax 需要 $2n^2$ 个浮点运算. 计算矩阵 $S\in\mathbb{R}^{p\times p}$ 的奇异值分解到给定的精度需要 $O(p^3)$ 个浮点运算 [133]. 我们假设 $\nabla f(X)$ 的计算结果已经得到, 因此 $\nabla f(X)$ 的计算并不统计在每步计算量中. 其他的一些设定与文献 [106, Table 1] 类似. 具体比较结果如表 3.1 所示.

在表 3.1中, “首次 t” 和 “重复 t” 分别表示仅在当前点计算初始试探点和接下来试探点的计算量. 这里, 函数值的额外计算并没有统计在内. 值得说明的是, 我们的两大类算法都无需进行线搜索, 并且梯度反射法和梯度投影法都可在固定常数步长的假设下收敛. 此外, 块坐标下降法的子问题也只需要通过一步迭代就可

更新策略		计算量	
		首次 t	重复 t
		测地线类算法	
$\mathcal{R}_X^{\text{geoe}}$	[110]	$O(n^3)$	$O(n^3)$
$\mathcal{R}_X^{\text{qgeo}}$	[109]	$O(n^3)$	$O(n^3)$
$\mathcal{R}_X^{\text{geoe}}$	[111]	$10np^2+2np+O(p^3)$	$4np^2+O(p^3)$
$\mathcal{R}_X^{\text{wy}}$	[99]	$7np^2+2np+O(p^3)$	$4np^2+\ np+O(p^3)$
		投影类算法	
$\mathcal{R}_X^{\text{qr}}$	[33]	$6np^2+3np+O(p^3)$	$2np^2+2np$
$\mathcal{R}_X^{\text{pd}}$	[33]	$7np^2+4np+O(p^3)$	$2np^2+2np+O(p^3)$
$\mathcal{R}_X^{\text{pj}}$	[108]	$7np^2+4np+O(p^3)$	$3np^2+3np+O(p^3)$
$\mathcal{R}_X^{\text{jd}}$	[106]	$7np^2+3np+O(p^3)$	$2np^2+3np+O(p^3)$
		我们的算法	
GR		$9np^2+4np+O(p^3)$	
GP		$7np^2+3np+O(p^3)$	
CBCD-GR		$4np^2+8np+O(p^3)$	
CBCD-GP		$4np^2+5np+O(p^3)$	

表 3.1 计算量的比较

Table 3.1 Comparison on computational cost

进行非精确求解. 因此, 一般来讲我们算法的每步计算量要少于收缩类算法. 虽然如此, 我们还是要指出实际的计算时间不仅依赖于总浮点运算数, 还和 BLAS 的高效计算息息相关.

进一步, CBCD-GR 和 CBCD-GP 分别表示第 3 步采用块坐标下降法的算法 3, 并在算法 4 的第 3 步中采用梯度反射法和梯度投影法. 我们注意到函数值梯度 $\nabla f_{i,X}((I_n - W_{\bar{i}}^{i-1}W_{\bar{i}}^{i-1\top})X_i)$ 的计算并没有考虑在内, 原因是当 $W_{\bar{i}}^{i-1\top}X_i = 0$ 时, 其等价于 $\nabla f_{i,X}(X_i)$. 当 $f_{i,X}(X_i)$ $(i = 1, ..., p)$ 是二次函数时, 我们可以得到限制在子空间 $\text{span}\left\{X_i, \nabla q_i(X_i)\right\}$ 上的子问题 (3.26) 的全局最优解, 此时每步需要的计算量为 $12np^2 + 3np + O(p^3)$.

3.4 收敛性分析

在本节, 我们建立新算法框架 3 的收敛性. 首先, 函数值的收敛性结果如下.

引理 3.7. 令 $\{X^k\}$ 是由算法 *3* 从初始点 $X^0 \in \mathcal{S}_{n,p}$ 生成的迭代点列, 则 $\left\{f(X^k)\right\}$ 收敛.

证明. 根据算法 3 中第 3 步 $\bar{X}$ 的构造, 结合引理 3.1, 我们得到

$$\begin{aligned} & f(X^k) - f(X^{k+1}) = f(X^k) - f(\bar{X}) + f(\bar{X}) - f(X^{k+1}) \\ \geq\ & C_1 \left\| \nabla f(X^k) - X^k X^{k\top} \nabla f(X^k) \right\|_{\mathrm{F}}^2 + \frac{1}{8\theta + 1} \left\| \bar{X}^\top \nabla f(\bar{X}) - \nabla f(\bar{X})^\top \bar{X} \right\|_{\mathrm{F}}^2 \\ \geq\ & C_1 \cdot \left\| \nabla f(X^k) - X^k \nabla f(X^k)^\top X^k \right\|_{\mathrm{F}}^2. \end{aligned} \tag{3.35}$$

因此, $\{f(X^k)\}$ 单调下降. 又由于 $\mathcal{S}_{n,p}$ 是紧集, $\{f(X^k)\}$ 下有界, 故我们有 $\{f(X^k)\}$ 收敛. □

接下来, 我们证明迭代点列的子列收敛性.

定理 3.1. 令 $\{X^k\}$ 是由算法 *3* 从初始点 $X^0 \in \mathcal{S}_{n,p}$ 生成的迭代点列. 则 $\{X^k\}$ 存在一个收敛子列, 并且点列 $\{X^k\}$ 的每一个聚点 X^* 都满足问题 (3.1) 的一阶最优性条件 (3.2).

证明. 由迭代点 X^k 的可行性可得点列 $\{X^k\}$ 有界, 故有收敛子序列. 令 X^* 是点列 $\{X^k\}$ 的任意聚点. 由 X^k 的可行性, 我们推出 X^* 满足一阶最优性条件 (3.2) 中的可行性.

利用引理 3.7 证明中的不等式 (3.35) 和 $\{f(X^k)\}$ 的有界性, 我们得到

$$\lim_{k \to +\infty} \left\| \nabla f(X^k) - X^k \nabla f(X^k)^\top X^k \right\|_{\mathrm{F}} = 0,$$

由此推出 $\left\| \nabla f(X^*) - X^* \nabla f(X^*)^\top X^* \right\|_{\mathrm{F}} = 0$. 根据 (3.3) 式, X^* 满足一阶最优性条件 (3.2) 的前两个条件. 结合引理 2.2, 定理得证. □

引理 3.7 和定理 3.1 保证了在 $\{X^k\}$ 的所有聚点集合上, $f(X)$ 是一个常数, 我们记为 f^*, 并记

$$\Omega_{\mathrm{FON}}^{f^*} = \Omega_{\mathrm{FON}} \cap \{X \mid f(X) = f^*\}, \tag{3.36}$$

其中 Ω_{FON} 参见定义 2.2.

接下来, 我们证明 X^k 和 $\Omega_{\mathrm{FON}}^{f^*}$ 的距离趋于 0.

推论 3.2. 令 $\{X^k\}$ 是由算法 3 从初始点 $X^0 \in \mathcal{S}_{n,p}$ 生成的迭代点列, 则我们有

$$f(X^k) \geq f^*, \qquad \forall\, k = 1, \cdots \tag{3.37}$$

且

$$\lim_{k\to\infty} \operatorname{dist}(X^k, \Omega_{\mathrm{FON}}^{f^*}) = 0. \tag{3.38}$$

证明. 由于 $\{f(X^k)\}$ 是非增的, 故关系式 (3.37) 成立. 接下来, 我们假设 (3.38) 不成立. 则存在 $\delta > 0$ 和 $\{X^k\}$ 的一个子序列, 记为 $\{X^{k_j}\}$, 使得

$$\operatorname{dist}(X^{k_j}, \Omega_{\mathrm{FON}}^{f^*}) \geq \delta. \tag{3.39}$$

由于 $\{X^{k_j}\}$ 有界, 故 $\{X^{k_j}\}$ 存在一个收敛子序列并且任意聚点都满足一阶最优性条件. 这与 (3.39) 矛盾, 由此推论成立. □

3.5 数值实验

在本节, 我们测试算法 3 的数值表现. 基于例 3.1, 我们选取了一大类测试问题. 关于例 3.2 的测试请见第 6 章. 本节的所有数值实验都在一台戴尔 Optiplex 9020 个人电脑运行, 处理器是两个 3.6GHz 的 Intel® Core™ i7-4790, 内存是 8GB. 数值实验的运行环境是 MATLAB R2016a.

3.5.1 算法的实现细节

在引理 3.3 和 3.4 中, 我们证明了当固定常数步长 τ 小于 ρ^{-1} 时, 梯度反射法和梯度投影法都满足函数值的充分下降性 (3.14). 然而, 我们很难准确的估计 ρ 的值, 并且 ρ^{-1} 的值可能会非常小, 也就导致了收敛速度过慢. 在实际中, 我们采取文献 [19] 提出的交替 Barzilai-Borwein 步长方法 (本文 1.1.3 小节), 这也是文献 [99] 所采用的方法. 具体来讲, τ 的更新规则如下.

$$\tau := \begin{cases} \tau^{\mathrm{BB1}}, & k\text{是奇数}, \\ \tau^{\mathrm{BB2}}, & k\text{是偶数}. \end{cases} \tag{3.40}$$

在上式中,

$$\tau^{\mathrm{BB1}} := \frac{\langle J^{k-1}, J^{k-1}\rangle}{|\langle J^{k-1}, K^{k-1}\rangle|}, \qquad \tau^{\mathrm{BB2}} := \frac{|\langle J^{k-1}, K^{k-1}\rangle|}{\langle K^{k-1}, K^{k-1}\rangle},$$
$$J^{k-1} = X^k - X^{k-1}, \qquad K^{k-1} = c(X^k) - c(X^{k-1}).$$

我们称带有 (3.40) 步长 τ 的梯度反射法和梯度投影法分别为 GR-BB 和 GP-BB. 相应的, 带有常数步长 τ 的梯度反射法和梯度投影法被称为 GR-F 和 GP-F.

以列为块的块坐标下降法只用来测试二次问题 (3.1). 因为在每一步内迭代中, 限制在二维子空间 $\mathrm{span}\left\{X_i, \nabla q_i(X_i)\right\}$ 的子问题 (3.26) 可以被精确求解到全局最优点. 然而, 在每一步外迭代中, 根据列更新顺序的不同产生了不同类型的算法. 通常, 有如下四种顺序:

1) 顺序循环: $j_i = i$, 对于 $i = 1, 2, ..., p$;

2) 随机抽样: $j_i = \lceil p \cdot \mathrm{rand}(1, 1) \rceil$, 对于 $i = 1, 2, ..., p$ (有放回的抽样);

3) 随机排列: $\{j_1, j_2, ..., j_p\}$ 是 $\{1, 2, \cdots, p\}$ 的一个随机排列 (无放回的抽样);

4) 贪婪: 对于 $i = 1, 2, ..., p$,

$$j_i := \arg\max_{j=1,\cdots,p} \left\| (I_n - W^{i-1} W^{i-1\top}) \nabla f_j(X_j) \right\|_2 .$$

相应的以列为块的块坐标下降法分别记作 CBCD-C, CBCD-R1, CBCD-R2 和 CBCD-G.

我们已经证明了由新算法框架 3 生成的任意迭代点都满足 (3.2) 中的对称性和可行性. 因此, 对于停机准则, 我们只需检查投影梯度 $||(I_n - XX^\top)\nabla f(X)||_{\mathrm{F}}$ 即可. 更具体的, 停机准则描述如下,

$$\left\| (I_n - XX^\top) \nabla f(X) \right\|_{\mathrm{F}} < \epsilon \left\| \nabla f(X^0) - X^0 \nabla f(X^0)^\top X^0 \right\|_{\mathrm{F}}, \tag{3.41}$$

其中 $\epsilon > 0$ 是给定的精度. (3.41) 式的右端项与初始的投影梯度大小相匹配. 另一方面, 一阶算法的收敛性可能会随着迭代点趋于一阶稳定点而变慢, 故我们也采用了文献 [99] 中的判断准则来探测和判断迭代点列的收敛, 具体如下.

$$\mathrm{tol}_k^x := \frac{\|X^k - X^{k+1}\|_{\mathrm{F}}}{\sqrt{n}} < \epsilon_x \quad \text{并且} \quad \mathrm{tol}_k^f := \frac{|f(X^k) - f(X^{k+1})|}{|f(X^k)| + 1} < \epsilon_f, \tag{3.42}$$

$$\begin{aligned} &\mathrm{mean}([\mathrm{tol}^x_{k-\min\{k,T\}+1}, \ldots, \mathrm{tol}^x_k]) < 10\epsilon_x, \\ \text{并且} \quad &\mathrm{mean}([\mathrm{tol}^f_{k-\min\{k,T\}+1}, \ldots, \mathrm{tol}^f_k]) < 10\epsilon_f. \end{aligned} \tag{3.43}$$

其中 $\mathrm{mean}(\cdot)$ 表示求平均值.

当上述三个停机准则 (3.41)-(3.43) 之一满足, 或迭代达到了最大迭代步数 MaxIter 时, 我们终止我们的算法. 默认的停机参数选取为 $\epsilon = 10^{-5}$, $\epsilon_x = 10^{-6}$, $\epsilon_f = 10^{-10}$, $T = 5$ 和 $\mathrm{MaxIter} = 3000$.

3.5.2 测试问题

在本小节, 我们引入一大类基于例 3.1 的测试问题. 首先, 考虑如下的带有正交约束的二次优化问题.

$$\begin{aligned} \min_{X \in \mathbb{R}^{n\times p}} \quad & \tfrac{1}{2}\mathrm{tr}(X^\top A X) + \mathrm{tr}(G^\top X) \\ \text{s.t.} \quad & X^\top X = I_p, \end{aligned} \tag{3.44}$$

其中矩阵 $A \in \mathbb{R}^{n\times n}$ 和 $G \in \mathbb{R}^{n\times p}$ 如下随机生成.

$$A := P\Lambda P^{\top}, \tag{3.45}$$

$$G := \alpha \cdot QD, \tag{3.46}$$

这里矩阵 $P = \mathrm{qr}\left(\mathrm{rand}(n,n)\right) \in \mathbb{R}^{n\times n}$, $\tilde{Q} = \mathrm{rand}(n,p) \in \mathbb{R}^{n\times p}$, $Q \in \mathbb{R}^{n\times p}$, $Q_i = \tilde{Q}_i/||\tilde{Q}_i||_2$ $(i = 1,2,...,p)$, 并且矩阵 $\Lambda \in \mathbb{R}^{n\times n}$ 和 $D \in \mathbb{R}^{p\times p}$ 是对角矩阵, 满足

$$\Lambda_{ii} := \begin{cases} \beta^{1-i}, & \text{当 } \omega_i < \xi, \\ -\beta^{1-i}, & \text{否则}, \end{cases} \quad \text{对于所有的 } i = 1,2,\ldots,n, \tag{3.47}$$

$$D_{jj} := \zeta^{j-1}, \quad \text{对于所有的 } j = 1,2,\ldots,p, \tag{3.48}$$

其中 $\omega_i \in [0,1]$ $(i = 1,2,...,n)$ 是随机生成的数. $n \times p$ 是变量大小; $\beta \geq 1$ 是决定 A 的特征值衰减的一个参数; $\zeta \geq 1$ 用来刻画 G 的每一列范数的增长率. 参数 $\alpha > 0$ 表示二次项和线性项之间的数量关系. 当 α 很大时, 线性项将会主导函数值的大小. 参数 $\xi \in [0,1]$ 用来决定矩阵 A 是否正定. 当 $\xi = 1$, 矩阵 A 正定, 而 $\xi = 0$ 则意味着 A 负定. 如无特殊说明, 这些参数的默认值为 $n = 3000, p = 60, \alpha = 1, \beta = 1.01, \zeta = 1.2, \xi = 1$. 我们选取 $X^0 = \mathrm{qr}\left(\mathrm{rand}(n,p)\right) \in \mathbb{R}^{n\times p}$ 作为算法的初始值.

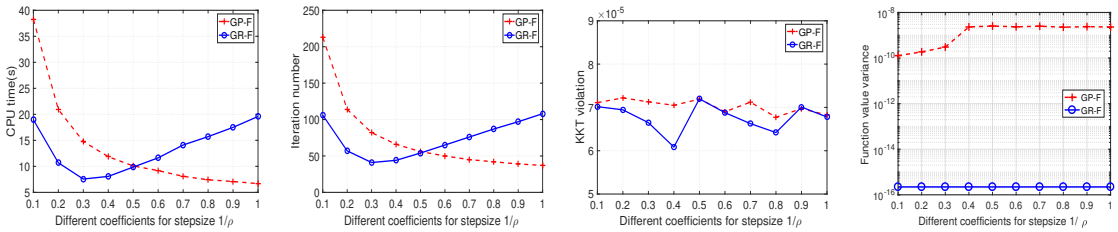
3.5.3 算法默认参数选取设置

在本小节, 我们通过数值实验选取梯度反射法, 梯度投影法和以列为块的块坐标下降法的默认设置.

首先我们比较在不同的固定步长下 GR-F 和 GP-F 的数值表现. 在实验中, 参数 p 选取为 $10\%n$, 参数 ζ 选取为 1.01, 其他参数选择为默认设置. 我们通过以下四个方面比较不同算法的数值表现: CPU 时间 (秒为单位)、总迭代数、一阶最优性条件 (KKT) 的违反度 (2.2.3 小节)、还有函数值. 在这里, 由于我们求解的是非凸问题, 为了比较不同算法之间的函数值, 我们定义 $f_{\min}$ 为不同算法得到的绝对值最小的函数值, 记 f_s 为算法 s 得到的函数值. 由此我们定义相对函数值如下,

$$\frac{|f_s - f_{\min}|}{1 + |f_{\min}|} + \mathrm{eps}, \tag{3.49}$$

这里 eps $= 2.2204e$-16 是 MATLAB 的机器精度. 为了得到以 log 为底的 y-轴的函数值, 我们在相对误差中加上机器精度 eps 来确保函数的比较值为正. 由于收缩类算法和我们的算法都是可行方法, 故我们不需要统计可行性的违反度 $||I - X^{\top}X||_{\mathrm{F}}$. 数值比较结果分别展示在图 3.2 的 (a)-(d) 中. 我们从 $0.1\rho^{-1}$ 到 ρ^{-1} 选取不同的步长 τ.



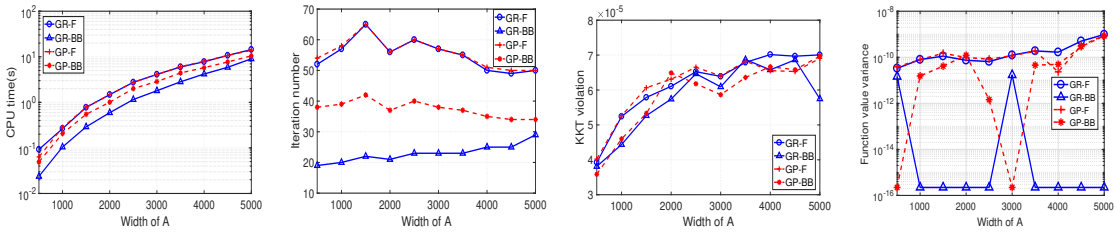
(a) CPU 时间　(b) 总迭代数　(c) KKT 违反度　(d) 相对函数值

图 3.2 不同固定步长的 GR-F 和 GP-F 的数值比较

Figure 3.2 Performance of GR-F and GP-F with different stepsizes

从图 3.2中, 我们观察到对于 GR-F 和 GP-F, $\tau = 1/3\rho$ 和 $1/\rho$ 分别是其最好的选择. 因此, 我们选取它们作为和 GR-BB/GP-BB 比较的默认步长.

接下来, 我们测试 10 个随机生成的问题, 其中 n 的大小由 500 变化到 5000, 变量的列数 p 选为 $10\%n$. 参数 ζ 为 1.01, 其他参数选取它们的默认值. 数值结果呈现在图 3.3 中.



(a) CPU 时间　(b) 总迭代数　(c) KKT 违反度　(d) 相对函数值

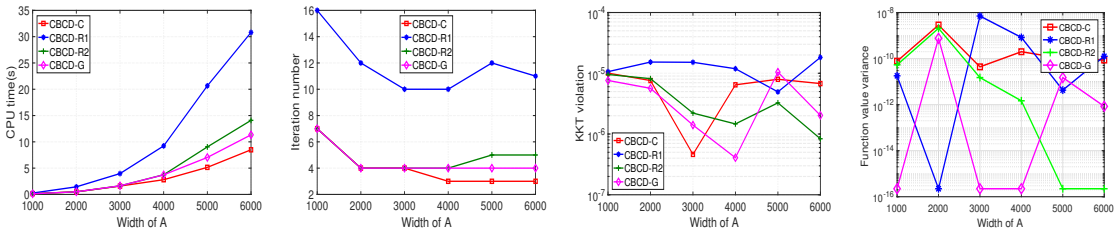
图 3.3 梯度类算法的数值比较

Figure 3.3 Performance of gradient based algorithms

从图 3.3 中, 我们注意到 GR-BB 和 GP-BB 比 GR-F 和 GP-F 需要更少的迭代数和更少的 CPU 时间, 同时它们也得到了同量级的 KKT 违反度. 不仅如此, 在大多数情况下, 我们发现 GR-BB 在 CPU 时间和总迭代数方面都优于 GP-BB. 因此, 在接下来的数值比较中, 我们选取 GR-BB 作为梯度类算法的代表.

接着, 我们比较在不同列更新顺序下, 以列为块的块坐标下降法的数值表现. 我们测试 CBCD-C, CBCD-R1, CBCD-R2 和 CBCD-RG 用来求解不同大小的问题, 其中 n 的大小由 1000 变化到 6000, 变量的列数 p 设为 $2\%n$, 其他参数默认. 数值结果如图 3.4 所示.

从图 3.4 我们观察到 CBCD-C, CBCD-G 和 CBCD-R2 有相似的表现, 并且 CPU 时间和总迭代数都优于 CBCD-R1. 在 CBCD-C, CBCD-G 和 CBCD-R2 中, CBCD-C



(a) CPU 时间　(b) 总迭代数　(c) KKT 违反度　(d) 相对函数值

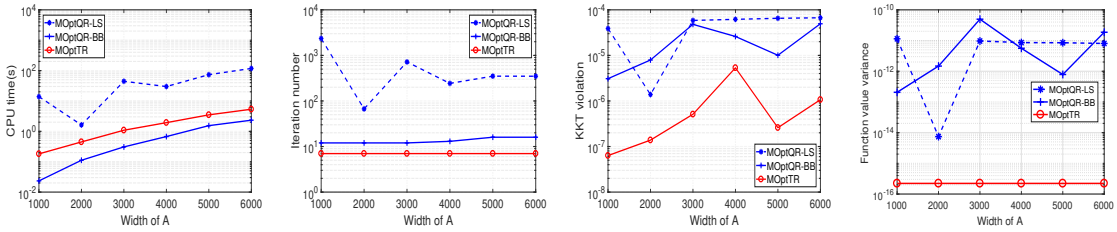
图 3.4 不同列更新顺序的块坐标下降法的数值比较

Figure 3.4 Performance of CBCD with different types of choosing working index

表现稳定且易于实现. 故在接下来的数值比较中, 我们采用 CBCD-C 作为以列为块的块坐标下降法的代表.

3.5.4 随机生成二次问题的数值比较

在本小节, 我们通过测试一大类问题 (3.44), 比较我们的算法 GR-BB, CBCD-C 和两种已有的高效算法的数值表现 (我们从表 2.1 中, 选取了两类具有最好表现的算法). 首先, 我们选取基于文献 [99] 的求解器 OptM[2]. 对其他已有的算法比较, 我们选取 MOptQR-LS (带有线搜索的 QR 流形收缩算法[3] [33]), MOptQR-BB (为了公平起见, 我们也实现了交替 BB 步长的 QR 流形收缩算法) 和 MOptTR (流形信赖域算法[3] [33]). 以上算法在第 1 章和第 2 章中都有详细介绍. 针对默认的问题 (3.44), 我们比较 MOptQR-LS, MOptQR-BB 和 MOptTR 的数值表现. 数值结果如图 3.5 所示.



(a) CPU 时间　(b) 总迭代数　(c) KKT 违反度　(d) 相对函数值

图 3.5 不同步长的 MOptQR 的数值比较

Figure 3.5 Performance of MOptQR with different type stepsize

从图 3.5 我们得知, 在此问题中 MOptQR-BB 优于其他两种方法. 因此, 我们选取 MOptQR-BB 作为另一个比较算法, 简记为 MOptQR.

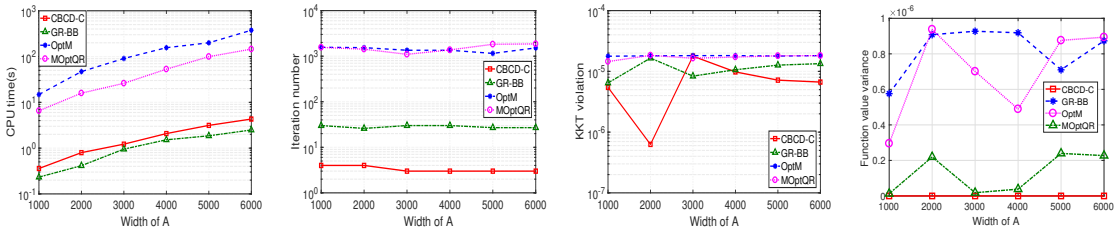
[2] https://github.com/wenstone/OptM

[3] http://www.manopt.org

在接下来的数值实验中, 我们只比较 GR-BB, CBCD-C, OptM 和 MOptQR 的数值表现. 采用小节 3.5.1 中同样的停机准则, 参数选取为默认值, 我们设计了六组测试问题, 在每组问题中, 只有一个变化的参数. 具体定义如下所示.

- 变量的行数, $n = 1000j$, 其中 $j = \mathbf{1}, \mathbf{2}, \mathbf{3}, \mathbf{4}, \mathbf{5}, \mathbf{6}$;
- 变量的列数, $p = 20j$, 其中 $j = \mathbf{1}, \mathbf{2}, \mathbf{3}, \mathbf{4}, \mathbf{5}, \mathbf{6}$;
- A 的特征值的衰减率, $\beta = 1.01 + 0.03j$, 其中 $j = 0, \mathbf{1}, 2, 3, \mathbf{4}, 5, 6, \mathbf{7}, 8$;
- G 每列范数的变化率, $\zeta = 1.01 + 0.03j$, 其中 $j = 0, \mathbf{1}, 2, 3, \mathbf{4}, 5, 6, \mathbf{7}, 8$;
- 线性项占比, $\alpha = 10^{-2}, \mathbf{10^{-1}}, \mathbf{1}, \mathbf{10}, 10^{2}$;
- A 的正定性, $\xi = 0.2(j-1)$, 其中 $j = \mathbf{1}, 2, \mathbf{3}, 4, \mathbf{5}, 6$.

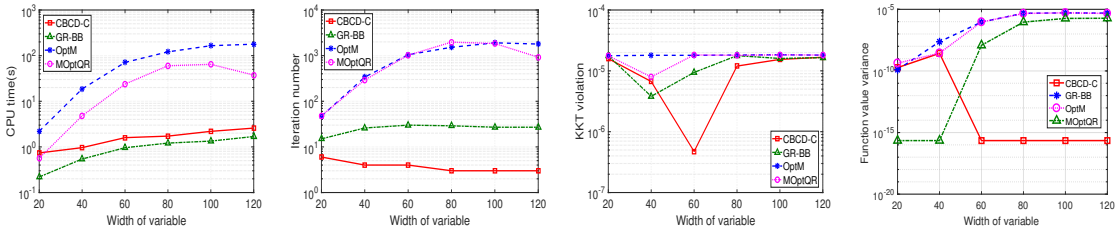
这些参数由等式 (3.45)-(3.48) 定义. 线性特征值问题, 也就是取 $\alpha = 0$ 的问题 (3.44), 不在我们的考虑内. 因为对于一般的特征值问题求解器, 还需考虑很多数值代数问题. 并且我们提出的梯度反射法和梯度投影法都需要针对不同的问题调整不同的步长, 因此在实际特征值计算中, 将会变得不实用. 虽然已有许多专门考虑线性特征值问题的求解器, 但他们很难处理一般的正交约束优化问题, 而这恰好是我们提出的新算法的优势所在. 这六组测试问题的数值结果分别如图 3.6-3.11 所示.



(a) CPU 时间　(b) 总迭代数　(c) KKT 违反度　(d) 相对函数值

图 3.6 变量行数 n 的数值比较

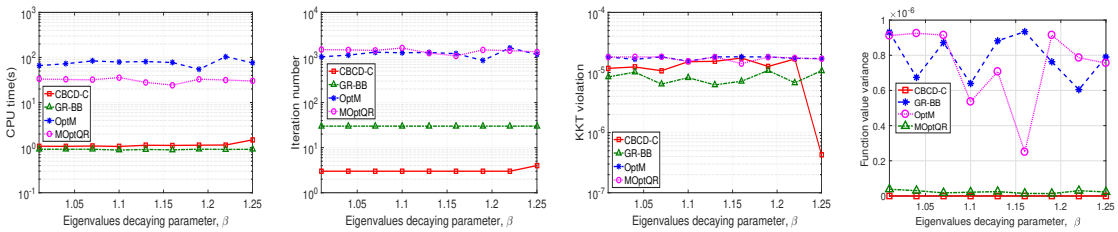
Figure 3.6 Comparison with varying matrix dimension n



(a) CPU 时间　(b) 总迭代数　(c) KKT 违反度　(d) 相对函数值

图 3.7 变量列数 p 的数值比较

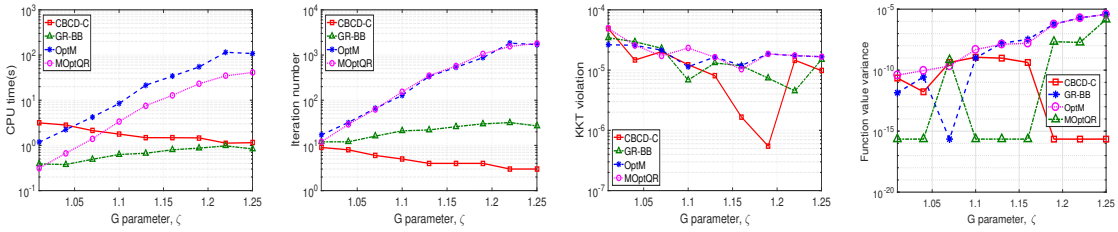
Figure 3.7 Comparison with varying width of variable p



(a) CPU 时间　(b) 总迭代数　(c) KKT 违反度　(d) 相对函数值

图 3.8 A 特征值的衰减率 β 的数值比较

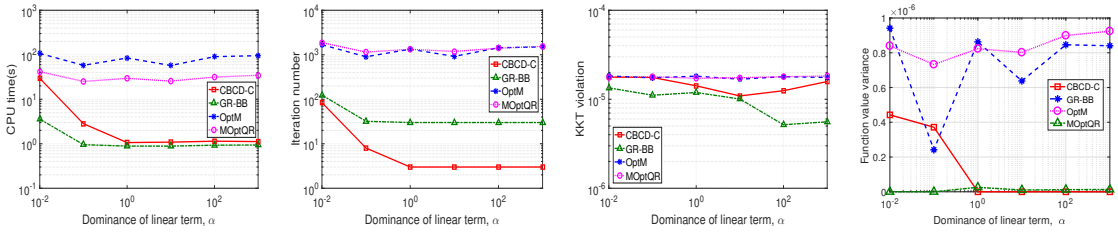
Figure 3.8 Comparison with varying decay parameter β



(a) CPU 时间　(b) 总迭代数　(c) KKT 违反度　(d) 相对函数值

图 3.9 G 每列范数的变化率 ζ 的数值比较

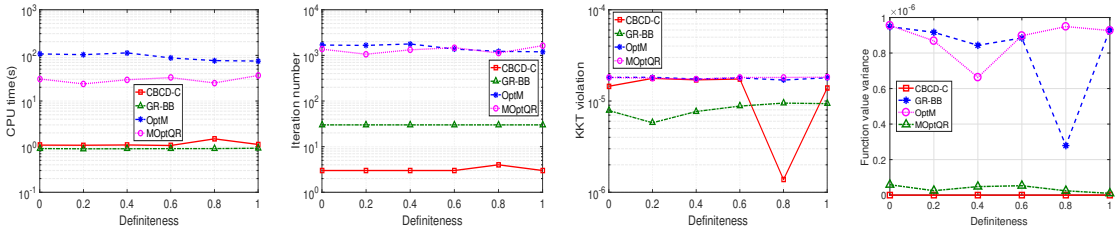
Figure 3.9 Comparison with varying G parameter ζ



(a) CPU 时间　(b) 总迭代数　(c) KKT 违反度　(d) 相对函数值

图 3.10 线性项占比 α 的数值比较

Figure 3.10 Comparison with varying dominance of linear term, α



(a) CPU 时间　(b) 总迭代数　(c) KKT 违反度　(d) 相对函数值

图 3.11 A 的正当性 ξ 的数值比较

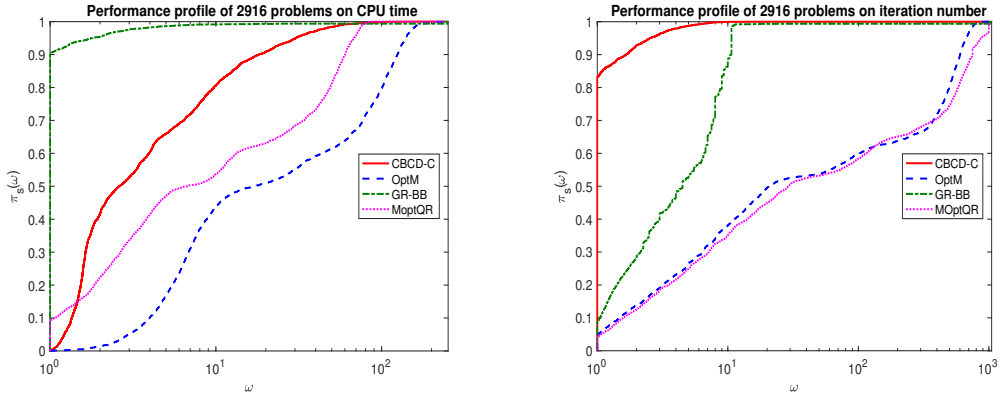
Figure 3.11 Comparison with varying nonnegtivity of A, ξ

通过这些数值比较，我们有如下的观察. 所有的求解器从相同的初始点出发都能得到相同的函数值. 并且它们具有 10^{-5} 量级的近似的 KKT 违反度. 在大多数实验中，GR-BB 和 CBCD-C 有比其他两个求解器较低的 KKT 违反度. 在所有四个算法中，CBCD-C 在几乎所有问题中都有最少的迭代步数，而 GR-BB 则有最少的 CPU 时间. 除了一些极端情况，CBCD-C 在 CPU 时间上表现仅次于 GR-BB.

最后，我们选取在问题描述中的所有加黑的问题，并在这些问题上进行算法的综合性能 [134] 比较. 其中，总计有 $6\times6\times3\times3\times3\times3 = 2916$ 个随机生成的问题. 综合性能比较能够消除一小部分难问题对算法的整体影响，也能减少算法结果对不同判断条件的敏感度，同时还提供了一个可视化的方法来直观了解不同算法之间的数值表现. 接下来，我们介绍一些此测试的关键参数. 对于问题 m 和求解器 s，我们令 $t_{m,s}$ 表示 CPU 时间或者总迭代数. 性能比定义为 $r_{m,s} := t_{m,s}/\min_s\{t_{m,s}\}$. 如果求解器 s 没有解出问题 m，我们令其性能比等于无穷或者某个充分大的数. 最后，求解器 s 的综合性能定义为

$$\pi_s(\omega) := \frac{\text{满足 } r_{m,s} \le \omega \text{ 的问题数}}{\text{问题总数}}.$$

它表示求解器在 $\omega \min_s t_{m,s}$ 秒时间内可以求解的测试问题集的百分比. 自然的，当 π_s 越靠近 1，求解器 s 的表现就越好. 综合性能关于 CPU 时间和总迭代步数的数值比较结果如图 3.12 所示.



(a) CPU 时间　　(b) 总迭代数

图 3.12 综合性能的数值比较

Figure 3.12 Performance profile

我们注意到在求解这 2916 测试问题时，GR-BB 在 CPU 时间上表现最好，CBCD-C 仅次于 GR-BB 且需要最少的迭代步数. 额外的，在表 3.2 中，我们还展示了不同求解器在这些测试问题上的平均 KKT，可行性违反度和相对函数值

的比较. 表 3.2 显示了所有求解器都得到近似同量级的平均 KKT 违反度, 可行性

	CBCD-C	OptM	GR-BB	MOptQR
KKT 违反度	1.6075e-05	2.1730e-05	1.9501e-05	2.5072e-05
可行性违反度	1.9172e-14	1.5276e-14	1.9350e-14	2.1006e-15
相对函数值变化	6.5780e-06	8.1754e-06	3.0417e-06	7.9584e-06

表 3.2 平均 KKT, 可行性违反度和相对函数值的数值比较

Table 3.2 Average KKT, feasibility violation and function value

违反度和相对函数值. 这里, 求解器 s 求解 m 问题时的相对函数值定义与 (3.49) 类似, 具体来说,

$$z_{m,s} := \frac{\left|f_{m,s} - \min_s\{f_{m,s}\}\right|}{1 + \left|\min_s\{f_{m,s}\}\right|}.$$

3.5.5 以列为块的块坐标下降法的全局性质

通过以上实验, 我们发现了一个有趣的现象: 虽然我们的问题 (3.1) 是非凸的, 但所有的求解器从随机初始点出发都得到了同样的函数值. 因此, 我们设计了一个新实验来研究算法的全局性质. 我们构造如下问题,

$$\begin{aligned} \min_{X\in\mathbb{R}^{3\times 2}} \quad & \tfrac{1}{2}\mathrm{tr}\left((X - X^*)^\top A(X - X^*)\right) \\ \text{s.t.} \quad & X^\top X = I_2, \end{aligned}$$

其中 $A = \begin{bmatrix} 13/2 & 2 & 0 \\ 2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$. 对于这个特殊问题, 我们可以很容易验证 $X^* = \begin{bmatrix} 3/5 & 0 \\ 4/5 & 0 \\ 0 & 1 \end{bmatrix}$ 是此问题的全局极小值点, 而

$$X^{\mathrm{I}} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}, \quad X^{\mathrm{II}} = \begin{bmatrix} 3/5 & 0 \\ 4/5 & 0 \\ 0 & -1 \end{bmatrix}, \quad X^{\mathrm{III}} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & -1 \end{bmatrix}$$

是其他一阶稳定点. 其中 X^{I} 是局部极小值点, 其他两个都是鞍点. 接下来, 我们令算法分别从这三个一阶稳定点附近初始, 测试 GR-BB, CBCD-C, OptM 和 MOptQR 的不同表现. 初始点定义为,

$$\begin{aligned} X^0 &:= \mathcal{P}_{\mathcal{S}_{n,p}}(X^i + \mu \cdot \mathrm{randn}(3,2)), \quad i = \mathrm{I}, \mathrm{II}, \mathrm{III} \\ X^0 &:= \mathcal{P}_{\mathcal{S}_{n,p}}(\mathrm{randn}(3,2)), \end{aligned}$$

其中 $\mu > 0$ 用来控制 X^0 到 X^i 之间的距离, $i = \mathrm{I}, \mathrm{II}, \mathrm{III}$. 我们令 $\mu = 10^{-4}$ 并分别测试从这四个不同初始点出发的不同算法的数值表现. 我们重复实验 1000 次, 记录每个求解器求得的不同解的次数. 实验结果汇总在表 3.3, 3.4, 3.5 和 3.6 中.

测试方法	X^*	X^{I}	X^{II}	X^{III}	成功率
CBCD-C	1000	0	0	0	100%
GR-BB	0	1000	0	0	0%
OptM	0	1000	0	0	0%
MOptQR	0	1000	0	0	0%

表 3.3 从 X^{I} 附近初始的数值结果

Table 3.3 Test results near X^{I}

测试方法	X^*	X^{I}	X^{II}	X^{III}	成功率
CBCD-C	1000	0	0	0	100%
GR-BB	0	1000	0	0	0%
OptM	729	0	271	0	72.9%
MOptQR	78	0	922	0	7.8%

表 3.4 从 X^{II} 附近初始的数值结果

Table 3.4 Test results near X^{II}

测试方法	X^*	X^{I}	X^{II}	X^{III}	成功率
CBCD-C	1000	0	0	0	100%
GR-BB	1000	0	0	0	100%
OptM	338	28	634	0	33.8%
MOptQR	5	5	990	0	0.5%

表 3.5 从 X^{III} 附近初始的数值结果

Table 3.5 Test results near X^{III}

测试方法	X^*	X^{I}	X^{II}	X^{III}	成功率
CBCD-C	1000	0	0	0	100%
GR-BB	656	344	0	0	65.6%
OptM	864	136	0	0	86.4%
MOptQR	774	226	0	0	77.4%

表 3.6 完全随机初始化的数值结果

Table 3.6 Test results with random initial guesses

我们从这些表中发现四个算法并不一定收敛到同一个稳定点. 在我们的测试中, CBCD-C 总能找到全局极小值点. 我们不确定这是否是巧合, 或者是 CBCD-C 能以大概率收敛到全局极小值点. 在以后的工作中, 我们将会做进一步的研究. 另一方面, 我们发现完全随机初始化可以增加其他三个算法找到全局极小值点的概率.

3.6 小结

在本章中, 对于正交约束优化问题 (3.1), 我们提出了一个新的一阶算法框架. 其主要包含两个步骤. 第一步, 我们选取一个函数值下降方法使得函数值减少并同时保持迭代点的可行性, 因此关于 Stiefel 流形的切空间计算可以被省略. 第二步, 我们利用乘子校正步来保证迭代点列的任意聚点都是一阶稳定点. 进一步, 对于一些特殊情况, 此校正步可以被省略. 我们提出了两类算法, 不同点在于算

法框架的第一步. 我们首先提出了一个梯度类方法, 其常数步长的全局收敛性可以得到保证, 因此线搜索将不再需要. 我们推荐了两个具体的算法, 梯度反射法和梯度投影法. 第二类算法是以列为块的块坐标下降法, 其列坐标的更新顺序由 Gauss-Seidel 类型决定. 同时, 我们也提出了一个新的方法用来非精确的高效求解子问题, 并保证了算法的全局收敛性. 针对一大类不同的测试问题, 数值实验显示了我们的新算法具有巨大潜力.

第 4 章　子空间加速的收缩类算法

在本章中, 我们考虑一大类正交约束优化问题, 其目标函数具有特定结构. 通过观察第 3 章中引入的乘子校正算法, 我们发现乘子校正步实际上是一个较小规模的优化问题的解. 利用子空间加速的思想, 我们将乘子校正步推广到一般的 Stiefel 流形收缩类算法, 由此得到了加速的收缩类算法. 进一步, 我们证明了算法的全局收敛性, 并给出了算法的局部收敛速度. 数值实验展示了新的加速技术具有巨大潜力.

4.1　引言

在本章中, 我们继续考虑第 3 章引入的正交约束优化问题,

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & f(X) := h(X) + \operatorname{tr}(G^\top X) \\ \text{s.t.} \quad & X^\top X = I_p, \end{aligned} \tag{4.1}$$

其中目标函数 $f : \mathbb{R}^{n\times p} \longrightarrow \mathbb{R}$ 满足假设 3.1, 也就是说 $G \in \mathbb{R}^{n\times p}$, $h(X)$ 满足正交不变性, 并且 $\nabla h(X) = H(X)X$, 这里 $H : \mathbb{R}^{n\times p} \longrightarrow \mathbb{SR}^{n\times n}$ 是一个矩阵函数.

在第 3 章中, 针对正交约束优化问题 (4.1), 我们提出了一个新的乘子校正算法框架. 其中, 乘子校正步的目的是使得一阶最优性条件 (3.2) 中的乘子对称性满足. 假设其中间点为 $\bar{X} \in \mathcal{S}_{n,p}$, 则乘子校正步如下所示,

$$X^{k+1} = \begin{cases} \bar{X}, & \text{当 } \bar{X}^\top G = G^\top \bar{X}; \\ -\bar{X}UT^\top, & \text{否则.} \end{cases} \tag{4.2}$$

其中 U 和 T 来自于 p 阶矩阵 $\bar{X}^\top G$ 的奇异值分解 $\bar{X}^\top G = U\Lambda T^\top$. 由 3.2.2 小节的讨论可知, 乘子校正步 (4.2) 实际上可由如下的最优化问题确定,

$$\min_{Q\in\mathcal{S}_{p,p}} f(\bar{X}Q).$$

求得上述问题的最优解后, 我们令 $X^{k+1} = \bar{X}Q^*$ 为下一步迭代点, 这与乘子校正步等价.

事实上, 上述问题也可表示为在 $\bar{X}$ 的正交不变子空间上寻找 $f(X)$ 的最优解, 即

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & f(X) \\ \text{s.t.} \quad & X \in \mathcal{D}_{\bar{X}}, \end{aligned} \tag{4.3}$$

其中 $\mathcal{D}_{\bar{X}} := \left\{\bar{X}Q : Q \in \mathcal{S}_{p,p}\right\}$ 是 $\bar{X}$ 的正交不变子空间. 因此, 我们将一个 $\mathcal{S}_{n,p}$ 上的优化问题限制在一个更小规模的子空间 $\mathcal{D}_{\bar{X}}$ 上. 这样做的好处是极大程度降低了计算的复杂度, 如果不考虑乘子的对称性, 则乘子校正步也可理解为优化中的子空间加速技术 [135].

综合上述讨论, 我们将问题 (4.3) 作为子空间加速的子问题. 当目标函数满足假设 3.1 时, 问题 (4.3) 有显式解

$$X^{k+1} = -\bar{X}UT^{\top}, \tag{4.4}$$

其中 $\bar{X}^{\top}G$ 的奇异值分解为 $\bar{X}^{\top}G = U\Lambda T^{\top}$. 如果我们利用上式对任意可行下降方法进行子空间加速, 则我们可以很自然地得到一个两阶段的加速方法. 考虑问题 (4.1), 假设当前点 $X^k \in \mathcal{S}_{n,p}$, 则加速算法框架如下所示,

$$X^k \xrightarrow{\text{下降阶段: 任意可行下降方法}} \bar{X} \xrightarrow{\text{加速阶段: (4.4) 式}} X^{k+1}.$$

特别地, 如果在函数值下降阶段, 采用第 3 章提出的梯度类方法或者以列为块的块坐标下降法, 则上述过程就是算法 3.

接下来, 我们考虑下降阶段的可行方法.

4.2 加速的收缩类算法

在第 2 章中, 我们详细介绍了 Stiefel 流形上的收缩类算法. 很自然地, 我们可以将其应用到加速算法框架中的下降阶段. 根据上一节的讨论, 并结合一般黎曼流形上的收缩类线搜索算法 1, 我们提出如下的子空间加速收缩类算法 (算法 5).

注 4.1. 一般来讲, 加速阶段的计算量会显著影响一个加速算法的数值表现. 如果加速过程得到的收益不足以弥补新增的计算代价, 则加速策略将没有实际意义. 在算法 5 中, 根据不同收缩映射 $\mathcal{R}$ 的选取, 一般来讲函数值下降阶段的计算量至少为 $O(np^2)$ 量级. 而我们提出的乘子校正加速步本质上只需要计算一个 p 阶的奇异值分解, 其计算量为 $O(p^3)$. 当 $n \gg p$ 时, 加速步的计算可以忽略不计.

从算法 5 中我们可以看到, 在下降阶段, 为了保证算法的全局收敛性, 我们需要进行线搜索计算. 搜索得到的迭代点至少要以一定的量级优于 Armijo 线搜索. 事实上, 我们可以直接取 $\bar{X}$ 为 Armijo 线搜索得到的点. 在实际计算中, 考虑到梯度类方法在靠近解处会收敛缓慢, 我们采用 1.1.3 小节提到的交替 BB 步长方法 (1.8), 由此线搜索过程得到了省略.

算法 5: 子空间加速的收缩类线搜索算法

1 给定 Stiefel 流形 $\mathcal{S}_{n,p}$ 上的收缩映射 $\mathcal{R}$, 常数 $\bar{\alpha} > 0, c, \beta, \sigma \in (0,1)$.

2 初始化: $X^0 \in \mathcal{S}_{n,p}$; 令 $k := 0$.

3 **while** 停机准则不满足 **do**

4 (函数值下降阶段) 选取搜索方向 $D^k \in \mathcal{T}_{X^k}\mathcal{S}_{n,p}$, 使其满足梯度相关条件 (定义 1.5). 接着选取 $\bar{X} \in \mathcal{S}_{n,p}$ 使得不等式

$$f(X^k) - f(\bar{X}) \geq c\left(f(X^k) - f(\mathcal{R}_{X^k}(\alpha^{\mathrm{A}} D^k))\right) \tag{4.5}$$

成立, 其中 α^{A} 是 Armijo 步长 (定义 1.6).

5 (加速阶段) 基于 $\bar{X}$, 由子空间问题的解 (4.4) 计算可行点 X^{k+1}.

6 令 $k := k + 1$.

7 返回 X^k.

4.3 收敛性分析

首先, 我们给出算法 5 的子列收敛性.

定理 4.1. *令 $\{X^k\}$ 是由算法 5 从初始点 $X^0 \in \mathcal{S}_{n,p}$ 生成的迭代点列. 则 $\{X^k\}$ 存在一个收敛子列, 并且点列 $\{X^k\}$ 的每一个聚点 X^* 都满足问题 (3.1) 的一阶最优性条件 (3.2).*

证明. 由引理 3.1 可知, 乘子校正步满足

$$8\theta\left(f(\bar{X}) - f(X^{k+1})\right) \geq \left\|\bar{X}^{\top}\nabla f(\bar{X}) - \nabla f(\bar{X})^{\top}\bar{X}\right\|_{\mathrm{F}}^2.$$

结合不等式 (4.5), 我们得到

$$f(X^k) - f(X^{k+1}) \geq c\left(f(X^k) - f(\mathcal{R}_{X^k}(\alpha^{\mathrm{A}} D^k))\right).$$

由收缩类线搜索算法的收敛性结果 [33, Theorem 4.3.1], 定理得证. □

从上述证明中我们发现, 实际上由子空间加速得到的 X^{k+1} 就是算法 1 中新迭代点的一种选取方式. 因此, 算法 5 也可看作是算法 1 在 Stiefel 流形上的一种具体实现.

特别的, 如果我们在算法 5 中取搜索方向 $D^k = -\mathrm{grad} f(X^k)$ (此时, 由定义 1.5, D^k 显然满足梯度相关条件), 则我们可以得到如下的局部线性收敛速度.

定理 4.2 ([33, Theorem 4.5.6]). *假设算法 5 中的搜索方向 $D^k = -\mathrm{grad} f(X^k)$. 由定理 4.1, 假设 $\{X^k\}$ 是由算法 5 从初始点 $X^0 \in \mathcal{S}_{n,p}$ 生成的迭代点列, 并收敛到 $X^* \in \mathcal{S}_{n,p}$. 假设目标函数 f 在 X^* 处的 Hesse 矩阵 H 正定, 则 X^* 为局部极小值点. 进一步, 给定 $r \in (r_*, 1)$, 其中 $r_* = 1 - \min\left(2\sigma\bar{\alpha}\lambda_{\min}(H), 4\sigma(1-\sigma)\beta\frac{\lambda_{\min}(H)}{\lambda_{\max}(H)}\right)$, 则存在 $K \geq 0$ 使得*

$$f(X^{k+1}) - f(X^*) \leq (r + (1-r)(1-c))(f(X^k) - f(X^*))$$

对于所有的 $k \geq K$ 都成立, 即算法 5 具有局部线性收敛速度.

值得说明的是, 上述性质只说明了子空间加速有不比原算法差的理论结果, 但未回答子空间加速为何会有更好的的数值表现.

4.4 数值实验

在本节我们研究子空间加速后的收缩类算法的数值表现. 本节的数值实验环境和停机准则与第 3 章相同. 其中, 停机参数选取如下, $\epsilon = 10^{-8}$, $\epsilon_x = 10^{-6}$, $\epsilon_f = 10^{-10}$, $T = 5$ 和 MaxIter $= 3000$.

在数值实验中, 我们考虑如下的测试问题,

$$\begin{aligned} \min_{X \in \mathbb{R}^{n\times p}} \quad & \tfrac{1}{2}\mathrm{tr}(X^\top A X) + \mathrm{tr}(G^\top X) \\ \text{s.t.} \quad & X^\top X = I_p, \end{aligned}$$

其中矩阵 $A \in \mathbb{R}^{n\times n}$ 和 $G \in \mathbb{R}^{n\times p}$ 的随机生成与 3.5.2 小节相同. 我们选取默认的问题参数为 $n = 3000, p = 60, \alpha = 1, \beta = 1.01, \zeta = 1.04, \xi = 1$.

由算法 5 可知, 如果想要具体实现算法, 我们必须选取 Stiefel 流形上合适的收缩映射. 基于第 2 章的讨论, 在本章的数值比较中, 我们选取投影和 QR 分解作为收缩映射, 即

$$\mathcal{R}_X^{\mathrm{pj}}(tD) = \mathcal{P}_{\mathcal{S}_{n,p}}(X - tD), \tag{4.6}$$

$$\mathcal{R}_X^{\mathrm{qr}}(tD) = \mathrm{qr}(X - tD). \tag{4.7}$$

在实际计算中, 我们选取

$$D = \mathrm{grad}_2 f(X) = \nabla f(X) - X\nabla f(X)^\top X.$$

关于步长的选取, 我们采用 1.1.3 小节提到的交替 BB 步长方法 (1.8), 应用到正交约束优化问题, 就是 (3.40) 式. 作为对照实验, 我们还考虑了未加速的标准投影和

QR 分解收缩类算法. 以上这些算法分别称为 ManPJ-BB, ManQR-BB, ManPJ-BB-C 和 ManQR-BB-C. 其中 “PJ” 和 “QR” 分别表示由投影收缩映射 (4.6) 和 QR 分解收缩映射 (4.7) 得到的算法, “-C” 表示带有子空间加速过程的算法, 反之则不加速. 作为比较测试, 我们还选取了第 3 章提出的基于梯度反射的乘子校正算法, GR-BB. 算法的默认设置和参数选择与第 3 章的数值实验相同.

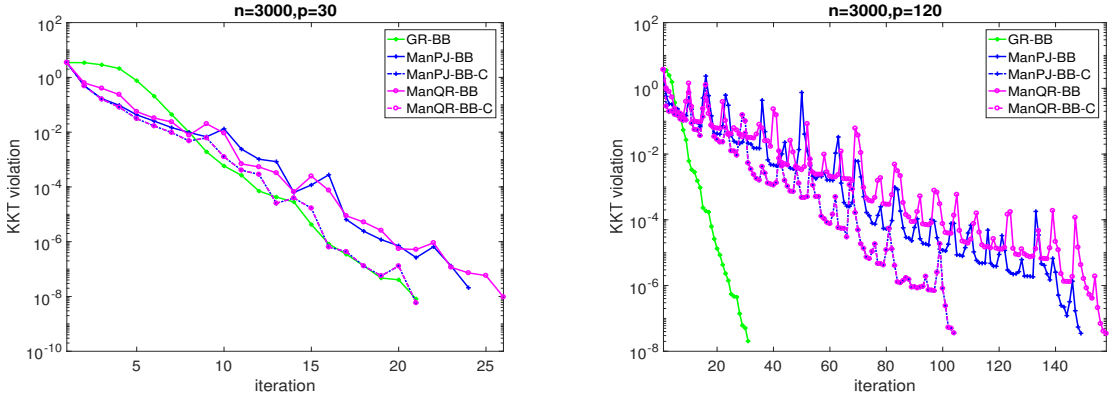
通过观察第 3 章中 ManQR-BB 的数值表现, 我们选取了三组对照问题用来测试.

- 变量的列数, $p = 30, 120$;
- G 每列范数的变化率, $\zeta = 1.01, 1.1$;
- A 的正定性, $\xi = 0, 0.5$.

在每组问题中, 其他未给定问题参数均选取其默认值.

图 4.1, 4.2 和 4.3 展示了不同算法的 KKT 违反度的变化情况. 从图中, 我们可以看出, 加速技术明显提升了原有收缩类算法的收敛速度. 特别地, 在图 4.3 中, 我们发现加速后的收缩类算法的迭代步数要少于原有的收缩类算法, 甚至也少于乘子校正算法 GR-BB. 除此之外, 我们还发现尽管乘子校正算法 GR-BB 与加速后的收缩类算法有相同的校正步, 但其数值表现仍然不同于收缩类算法. 在大部分的测试问题中, GR-BB 都具有更好的表现. 而投影收缩算法和 QR 分解收缩算法的表现基本类似. 从图中, 我们还发现了一个有趣的现象, 经过加速后的投影和 QR 分解收缩类算法有完全相同的迭代点列, 也就是说加速后的两类算法等价. 事实上, 我们选取的收缩映射 (4.6) 和 (4.7) 本质上都是在 $X - tD$ 的值空间选取正交矩阵, 也就是 $\mathcal{R}_X^{\mathrm{pj}}(tD), \mathcal{R}_X^{\mathrm{qr}}(tD) \in \mathcal{D}_{\mathcal{P}_{St_{n,p}}(X-tD)}$. 因此两类收缩算法生成的子问题 (4.3) 相同, 从而得到了一样的加速点.

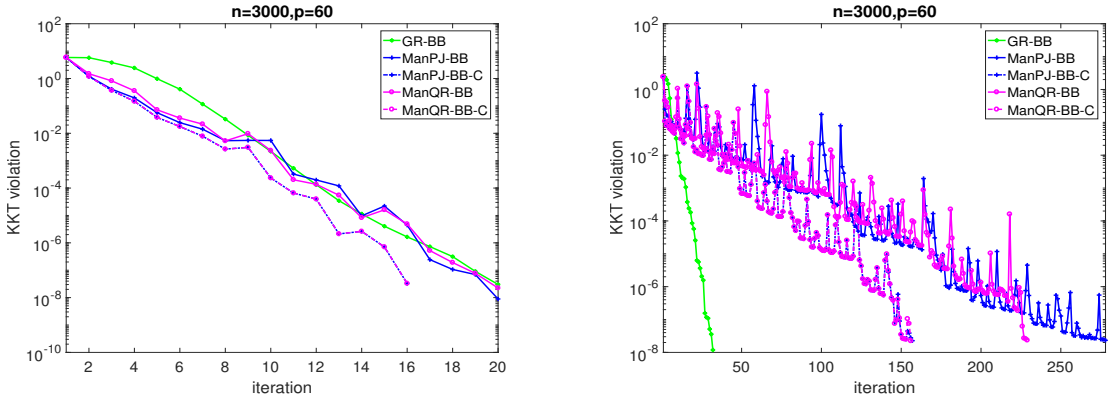
我们在注 4.1 中提到, 尽管加速技术减少了原有算法的迭代步数, 但是我们还应考虑其带来的额外计算. 因此, 在表 4.1 中, 我们列出了不同算法详细的数值结果. 从表中, 我们发现所有算法都可以求解到同样的函数值, 同时也有相同量级的 KKT 违反度. 从迭代步数和 CPU 时间上, 我们发现尽管加速步引入了额外的计算量, 但是其带来的迭代步数降低是显著的. 在总的 CPU 时间上, 加速后的收缩类算法相较于原收缩算法需要更少的运行时间. 这说明我们提出的子空间加速技术有效且实用.



(a) $p = 30$ (b) $p = 120$

图 4.1 加速收缩类算法的数值比较: 变量的列数 p

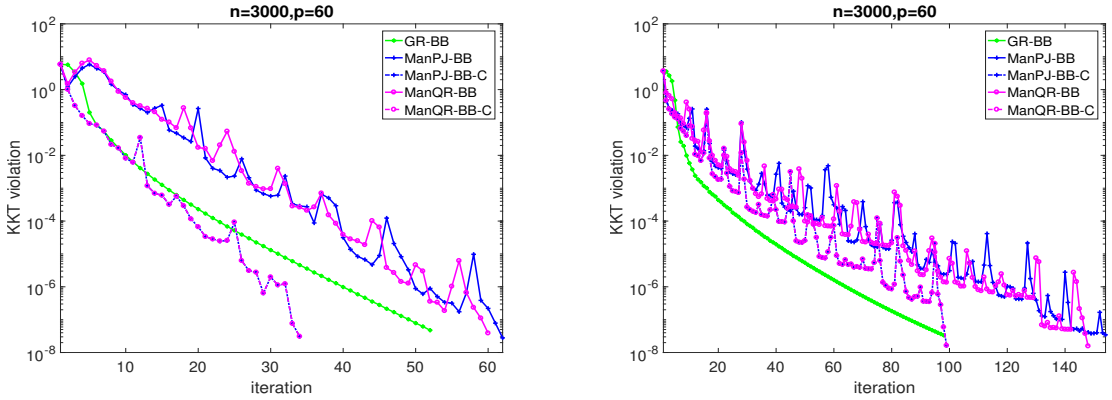
Figure 4.1 Performance of accelerated retraction-based methods: width of variable p



(a) $\zeta = 1.01$ (b) $\zeta = 1.1$

图 4.2 加速收缩类算法的数值比较: G 列范数的变化率 ζ

Figure 4.2 Performance of accelerated retraction-based methods: G parameter ζ



(a) $\xi = 0$ (b) $\xi = 0.5$

图 4.3 加速收缩类算法的数值比较: A 的正定性 ξ

Figure 4.3 Performance of accelerated retraction-based methods: nonnegtivity of A, ξ

变化参数	测试方法	函数值	KKT 违反度	总迭代数	CPU 时间 (s)
$p = 30$	GR-BB	-17.7869766760	8.15e-09	21	0.35
	ManPJ-BB	-17.7869766760	2.09e-08	24	0.36
	ManPJ-BB-C	-17.7869766760	5.84e-09	21	0.34
	Man-BB	–17.7869766760	9.58e-09	26	0.38
	ManQR-BB-C	-17.7869766760	5.84e-08	21	0.35
$p = 120$	GR-BB	-25.5806718521	2.04e-08	31	1.80
	ManPJ-BB	-25.5806718521	3.51e-08	149	5.76
	ManPJ-BB-C	-25.5806718521	3.60e-08	104	5.24
	ManQR-BB	-25.5806718521	3.45e-08	158	6.22
	ManQR-BB-C	-25.5806718521	3.60e-08	104	5.18
$\zeta = 1.01$	GR-BB	-43.6334383053	3.02e-08	20	0.62
	ManPJ-BB	-43.6334383053	8.93e-09	20	0.45
	ManPJ-BB-C	-43.6334383053	3.29e-08	16	0.43
	ManQR-BB	-43.6334383053	2.25e-08	20	0.44
	ManQR-BB-C	-43.6334383053	3.29e-08	16	0.42
$\zeta = 1.1$	GR-BB	-10.9204582842	1.19e-08	32	0.98
	ManPJ-BB	-10.9204582842	2.33e-08	278	5.91
	ManPJ-BB-C	-10.9204582842	2.27e-08	157	4.07
	ManQR-BB	-10.9204582842	2.34e-08	229	4.82
	ManQR-BB-C	-10.9204582842	2.24e-08	156	4.00
$\xi = 0$	GR-BB	-44.7900645660	4.76e-08	52	1.59
	ManPJ-BB	-44.7900645660	2.79e-08	62	1.37
	ManPJ-BB-C	-44.7900645660	3.08e-08	34	0.90
	ManQR-BB	-44.7900645660	3.89e-08	60	1.26
	ManQR-BB-C	-44.7900645660	3.08e-08	34	0.90
$\xi = 0.5$	GR-BB	-23.3612295898	3.33e-08	98	2.89
	ManPJ-BB	-23.3612295898	3.46e-08	154	3.27
	ManPJ-BB-C	-23.3612295898	1.64e-08	99	2.53
	ManQR-BB	-23.3612295898	1.57e-08	148	3.06
	ManQR-BB-C	-23.3612295898	1.64e-08	99	2.57

表 4.1 加速收缩类算法的数值比较结果

Table 4.1 Comparison on accelerated retraction-based methods

4.5 小结

在本章中, 对于正交约束优化问题 (4.1), 考虑其目标函数的特殊结构, 我们提出了一个新的子空间加速算法. 其中, 子空间加速的思想主要来自于第 3 章的乘子校正步. 通过考虑一个更小的限制在子空间的优化问题, 我们可以将原有的可行下降算法进行加速. 其中, 我们特别考虑了收缩类线搜索算法, 由此得到了加速的收缩类算法. 进一步, 我们给出了算法的全局收敛性以及局部线性收敛速度. 数值实验说明了我们的加速技术高效且实用.

第 5 章　基于增广 Lagrange 函数的并行算法

众所周知, 正交化过程具有较低的可扩展性, 由此正交约束优化问题的可并行算法设计变得极其艰难. 尽管如此, 在一些应用中, 可并行算法的需求却变得越来越大, 例如材料计算. 在本章中, 我们提出了一个用于求解正交约束优化问题的邻近点线性化增广 Lagrange 算法 (PLAM) . 不同于经典的增广 Lagrange 函数法, 在我们的算法中, 原始变量的更新通过极小化增广 Lagrange 函数的邻近点线性化逼近得到. 同时, 对偶变量由其一阶稳定点处的显式解更新得到. 当问题的求解需要高精度的可行性时, 正交化过程只会出现在算法的最后一步, 作为后处理. 由此, 新算法的主要步骤都可以很自然地进行矩阵计算层面的并行化. 进一步, 我们在一些较弱假设条件下证明了算法的全局子列收敛性, 最坏情况下的复杂度以及 PLAM 的局部收敛速度. 此外, 为了减弱罚参数对算法的敏感性, 我们提出了一个改进的 PLAM 方法, 叫作 PLAM 的可并行列极小化算法 (PCAL). 串行的数值实验结果说明了算法 PLAM 的新乘子更新方式显著加速了算法的收敛速度, 并且数值表现与已有的针对正交约束优化问题的可行方法不相上下. 同时, PCAL 的数值表现也并不很依赖于罚参数的选取. 并行环境下的数值实验验证了我们的算法 PCAL 有较好的数值表现并且具有较高的可扩展性.

5.1　引言

在本章中, 我们考虑如下的正交约束优化问题,

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & f(X) \\ \text{s.t.} \quad & X^\top X = I_p, \end{aligned} \tag{5.1}$$

其中 I_p 是 p 阶单位矩阵, $2p \le n$. 对于目标函数 f, 我们仅有如下的连续可微假设. 在一些理论分析中, 如果需要函数 f 的二次可微性, 我们将会特别提及.

假设 5.1. *函数 f 连续可微.*

在第 2 章中, 我们详细介绍了针对正交约束优化问题 (5.1) 的实际应用和已有优化算法. 其中, Kohn-Sham 总能量泛函极小化问题的变量规模经常非常巨大, 因此对于正交约束优化问题的高效求解器的开发变得至关重要. 目前已有的正交约束优化算法都只能处理一些较小规模的问题. 然而, 当变量矩阵的列数 p 变得很大的时候, 求解正交约束优化问题的主要瓶颈是其算法并发性的缺乏, 也就是

不易被并行计算. 因此, 并行算法的研究是目前优化领域中的迫切需求. 尽管高可扩展性的算法在近些年越来越得到 Kohn-Sham 密度泛函理论领域的关注, 但是目前除了 [57] 之外还没有其他成功的尝试.

事实上, 我们发现并行化对于正交约束优化问题而言特别困难, 其主要原因是无论考虑任何算法, 正交化计算过程的可扩展性都会很低. 目前几乎所有的有效算法都需要可行的迭代点列. 也就是说, 为了实现可行性, 这些算法或多或少都需要进行显式或隐式的正交化计算. 这种计算通常缺乏可扩展性, 因此在相应的算法中, 正交化过程计算量占比巨大. 例如, 我们考虑离散的 Kohn-Sham 总能量极小化问题, 在每一步迭代计算中, 函数值和一阶导数的计算需要 $O(n\log n+np)$ 或者 $O(np)$ 个浮点运算, 不同的计算量取决于使用平面波离散还是有限差分离散. 一般来讲, 在求解问题的任意优化算法中, 每步主要的迭代计算包括: 用于 BLAS3 的 $O(np^2)$ 个浮点运算, 以及正交化过程的 $O(p^3)$ 个浮点运算. 其中, BLAS3 的计算可以被高效并行, 而正交化过程的并行计算却很难实现.

为了突破这个瓶颈, 我们建议采用不可行方法来代替可行方法. 参考 2.3.2 小节的讨论, 针对一般的正交约束优化问题, 目前还没有实际有效的不可行方法. 已有的不可行方法, 要不就是对于特殊问题 (Rayleigh-Ritz 迹) 的极小化 (2.3) [49, 121], 要不就是采用交替方向乘子法 (ADMM) 框架, 引入辅助变量来分离目标函数与正交约束 [117]. 第一种方法很难推广到一般的目标函数, 而第二种方法总体而言表现不佳, 具体的数值例子可参见本章数值实验一节.

接下来, 我们考虑新的不可行算法框架.

5.2 乘子显式更新算法

5.2.1 增广 Lagrange 函数法

如果我们不需要每步迭代点都满足可行性, 那么一个直接的方法是罚函数法. [49] 考虑了针对特征值问题的 Courant 罚函数 [118], 我们简单实现后发现其很难推广到一般的正交约束优化问题. 因此, 我们考虑另一类常用的罚函数法: 增广 Lagrange 函数法 (ALM) [3, 136, 137], 也称为乘子法.

首先, 我们定义问题 (5.1) 的增广 Lagrange 函数,

$$\begin{aligned}\mathcal{L}_\beta(X,\Lambda) &= f(X)-\frac{1}{2}\langle\Lambda, X^\top X-I_p\rangle+\frac{\beta}{4}||X^\top X-I_p||_{\mathrm{F}}^2\\ &= f(X)+\frac{\beta}{4}\left\|X^\top X-\left(I_p+\frac{1}{\beta}\Lambda\right)\right\|_{\mathrm{F}}^2-\frac{1}{4\beta}||\Lambda||_{\mathrm{F}}^2,\end{aligned}\tag{5.2}$$

其中 β 是罚参数, $\Lambda \in \mathbb{SR}^{p,p}$ 表示正交约束对应的 Lagrange 乘子. 增广 Lagrange 函数法的步骤如算法 6 所示.

算法 6: 增广 Lagrange 函数法 (ALM)

1 初始化: $\Lambda^0 \in \mathbb{R}^{p\times p}$; 令 $k := 0$

2 **while** 停机准则不满足 **do**

3 关于原始变量 X 极小化增广 Lagrange 函数, 得到

$$X^{k+1} := \underset{X\in\mathbb{R}^{n\times p}}{\arg\min} \quad \mathcal{L}_\beta(X, \Lambda^k), \tag{5.3}$$

4 更新 Lagrange 乘子

$$\Lambda^{k+1} := \Lambda^k - \beta({X^{k+1}}^\top X^{k+1} - I_p). \tag{5.4}$$

5 必要时更新罚因子 β.

6 令 $k := k + 1$.

7 返回 X^k.

注 5.1. *我们知道, 当 Lagrange 乘子恰当并且罚参数 β 足够大时, 增广 Lagrange 函数是精确罚函数 [3], 也就是说, 原问题的解也是增广 Lagrange 函数无约束优化问题的解. 在这种情况下, 我们必须要求 Lagrange 乘子更新满足, 对偶梯度上升法 (5.4) 或者极小化 KKT 系统得到的线性最小二乘问题的解. 一般来讲, 算法 6 对于带有线性约束的问题具有良好的表现, 然而对于非线性约束的优化问题, 目前在实际中我们还不清楚如何有效的选取罚参数 β, 使其对算法的数值表现不敏感.*

本章的主要目标是针对问题 (5.1), 寻找一种不可行算法, 使其和已有的可行方法计算代价相似. 否则, 我们很难从并行化计算的过程获得较高的收益. 基于此种目的, 我们非常细致的测试了经典的算法 6, 并且极尽可能的去调整罚参数 β 的选取. 不幸的是, 对于正交约束优化问题 (5.1), 经典的增广 Lagrange 函数法远不能满足我们求解实际问题的需要. 因此, 我们需要采取一种新的方法来改进经典的增广 Lagrange 函数法.

5.2.2 乘子显式更新的增广 Lagrange 函数法

回顾问题 (5.1) 的一阶最优性条件,

$$\begin{cases} (I_n - XX^\top)\nabla f(X) &= 0; & (\text{次稳定性}) \\ X^\top \nabla f(X) &= \nabla f(X)^\top X; & (\text{对称性}) \\ X^\top X &= I_p. & (\text{可行性}) \end{cases} \tag{5.5}$$

根据推论 2.1, 我们知道 Lagrange 乘子 Λ 在任意一阶稳定点处, 都有如下的显式表达式

$$\Lambda = \nabla f(X)^\top X = X^\top \nabla f(X). \tag{5.6}$$

因此, 在算法 6 中, 最直接也是最自然的想法就是采取如下的乘子对称化形式作为新的乘子更新公式.

$$\Lambda = \Psi(\nabla f(X)^\top X), \tag{5.7}$$

其中 $\Psi(A) := (A + A^\top)/2$. 值得说明的是, 对称化过程是必要的, 因为表达式 $\nabla f(X)^\top X$ 的对称性在每一步迭代中并不能够得到保证.

在接下来的引理和理论分析中, 我们证明了如果算法 6 采用 (5.7) 式所示的乘子更新方式, 则其对应的增广 Lagrange 函数仍是精确罚函数, 并且罚参数 β 的精确下界可以得到估计. 由此, 我们可以省去罚参数 β 的更新.

引理 5.1. 令 X^* 是如下问题的二阶稳定点

$$\min_{X \in \mathbb{R}^{n\times p}} \mathcal{L}_\beta(X, \Lambda^*) \tag{5.8}$$

其中 $\Lambda^* = \Psi(\nabla f(X^*)^\top X^*)$. 假设 $\beta > \lambda_{\max}(\nabla^2 f(X^*))$, 则 X^* 也是问题 (5.1) 的二阶稳定点. 进一步, 在 X^* 处, 最优性条件 (5.5) 成立.

证明. 首先, 我们有

$$\nabla_X \mathcal{L}_\beta(X^*, \Lambda^*) = \nabla f(X^*) + \beta X^* \left(X^{*\top} X^* - \left(I_p + \frac{1}{\beta}\Lambda^* \right) \right); \tag{5.9}$$

$$\begin{aligned} \nabla^2_{XX} \mathcal{L}_\beta(X^*, \Lambda^*)[S] &= \nabla^2 f(X^*)[S] + \beta S \left(X^{*\top} X^* - \left(I_p + \frac{1}{\beta}\Lambda^* \right) \right) \\ &\quad + \beta X^* (S^\top X^* + X^{*\top} S). \end{aligned} \tag{5.10}$$

因为 X^* 是问题 (5.8) 在 $\Lambda^* = \Psi(\nabla f(X^*)^\top X^*)$ 下的二阶稳定点, 由定义 2.3, 我们有

$$\nabla \mathcal{L}_\beta(X^*, \Lambda^*) = 0; \tag{5.11}$$

$$\langle S, \nabla^2_{XX} \mathcal{L}_\beta(X^*, \Lambda^*)[S] \rangle \geq 0, \quad \forall S \neq 0. \tag{5.12}$$

接着将 (5.9) 式带入 (5.11) 式, 我们得到

$$\nabla f(X^*) - X^*\Lambda^* - \beta X^*(I_p - X^{*\top}X^*) = 0. \tag{5.13}$$

在 (5.13) 式的两边左乘 $X^{*\top}$, 我们有

$$X^{*\top}\nabla f(X^*) = X^{*\top}X^*\Lambda^* + \beta X^{*\top}X^*(I_p - X^{*\top}X^*). \tag{5.14}$$

假设 $X^* = U\Sigma V^\top$ 是 X^* 的奇异值分解, 其中 $U \in \mathcal{S}_{n,p}$, $\Sigma \in \mathbb{D}^p$ 且 $V \in \mathcal{S}_{p,p}$. 由此我们推出, $X^{*\top}X^* = V\Sigma^2V^\top$. 进一步, 我们有

$$X^{*\top}\nabla f(X^*) - \beta V\Sigma^2V^\top = V\Sigma^2V^\top\Lambda^* - \beta V\Sigma^4V^\top.$$

在等式两边左乘 $V^\top$ 并且右乘 V, 我们得到

$$V^\top X^{*\top}\nabla f(X^*)V - \beta\Sigma^2 = \Sigma^2(V^\top\Lambda^*V - \beta\Sigma^2).$$

两边取算子 $\Phi(\,\cdot\,) = \mathrm{Diag}(\mathrm{diag}(\,\cdot\,))$ 并利用

$$\mathrm{diag}(V^\top X^{*\top}\nabla f(X^*)V) = \mathrm{diag}(V^\top\nabla f(X^*)^\top X^*V) = \mathrm{diag}(V^\top\Lambda^*V), \tag{5.15}$$

我们得到

$$(I_p - \Sigma^2)(\Phi(V^\top\Lambda^*V) - \beta\Sigma^2) = 0, \tag{5.16}$$

由此推出

$$D(\Phi(V^\top\Lambda^*V) - \beta\Sigma^2) = 0, \tag{5.17}$$

其中矩阵 $D \in \mathbb{D}^p$ 满足

$$D_{ii} = \begin{cases} 0, & \text{如果 } (I_p - \Sigma^2)_{ii} = 0; \\ 1, & \text{否则,} \end{cases} \qquad \forall i = 1, ..., p.$$

另一方面, 因为 $n \geq 2p$, 故存在 $\tilde{U} \in \mathcal{S}_{n,p}$ 使得 $\tilde{U}^\top U = 0$. 令 $S = \tilde{U}DV^\top$, 如果 $S \neq 0$, 我们将 S 带入 (5.10) 式得到

$$\begin{aligned} & \langle S, \nabla^2_{XX}\mathcal{L}_\beta(X^*, \Lambda^*)[S]\rangle \\ = \;& \mathrm{tr}(S^\top\nabla^2 f(X^*)[S]) - \beta\mathrm{tr}(S^\top S) - \mathrm{tr}(S^\top S(\Lambda^* - \beta X^{*\top}X^*)) \\ = \;& \mathrm{tr}\left(S^\top(\nabla^2 f(X^*) - \beta I)[S]\right) - \mathrm{tr}(V^\top S^\top SVV^\top(\Lambda^* - \beta V\Sigma^2V^\top)V) \\ = \;& \mathrm{tr}\left(S^\top(\nabla^2 f(X^*) - \beta I)[S]\right) - \mathrm{tr}(D^2(V^\top\Lambda^*V - \beta\Sigma^2)) \\ = \;& \mathrm{tr}\left(S^\top(\nabla^2 f(X^*) - \beta I)[S]\right) - \mathrm{tr}(D^2(\Phi(V^\top\Lambda^*V) - \beta\Sigma^2)). \end{aligned}$$

其中 I 表示从 $\mathbb{R}^{n\times p}$ 到 $\mathbb{R}^{n\times p}$ 的恒等映射. 结合二阶最优性条件 (5.12), 关系式 (5.17) 和 β 的假设, 我们得到

$$0 \le \langle S, \nabla^2_{XX}\mathcal{L}_\beta(X^*,\Lambda^*)[S]\rangle = \mathrm{tr}\left(S^\top(\nabla^2 f(X^*) - \beta I)[S]\right) < 0, \tag{5.18}$$

由此得到矛盾. 故 $S = 0$, 从而 $\Sigma = I_p$. 因此, 我们有 $X^* \in \mathcal{S}_{n,p}$. 结合 (5.11) 式和 (5.12) 式, 我们可以很容易验证最优性条件 (5.5) 成立. 由此引理得证. □

引理 5.1 保证了使用乘子精确更新公式 (5.7) 的增广 Lagrange 函数仍是一个精确罚函数. 进一步, 为了得到一阶方法的收敛性, 我们需要建立一阶版本的引理 5.1. 此外, 为了得到算法的全局收敛速度, 我们还需要建立可行性违反度和一阶最优性条件之间的关系.

引理 5.2. 对于任意满足 $\sigma_{\min}(X^*) > 0$ 的 X^*, 假设 $\beta > \frac{||\nabla f(X^*)||_2\cdot||X^*||_2+\delta}{\sigma^2_{\min}(X^*)}$, 其中 $\delta > 0$. 则下式成立

$$||X^{*\top}X^* - I_p||_{\mathrm{F}} \le \frac{||X^*||_2}{\delta}\cdot||\nabla_X\mathcal{L}_\beta(X^*,\Lambda^*)||_{\mathrm{F}}, \tag{5.19}$$

其中 $\Lambda^* = \Psi(\nabla f(X^*)^\top X^*)$. 特别的, 如果 X^* 还是如下问题

$$\min_{X\in\mathbb{R}^{n\times p}} \mathcal{L}_\beta(X,\Lambda^*)$$

的一阶稳定点, 其中 $\Lambda^* = \Psi(\nabla f(X^*)^\top X^*)$, 则 X^* 也是原问题 (5.1) 的一阶稳定点.

证明. 为了简便起见, 我们记 $G = \nabla_X\mathcal{L}_\beta(X^*,\Lambda^*)$. 在 (5.9) 式两端左乘 $X^{*\top}$, 并利用 X^* 的奇异值分解 $X^* = U\Sigma V^\top$, 我们得到

$$X^{*\top}G = X^{*\top}\nabla f(X^*) - \beta V\Sigma^2V^\top - V\Sigma^2V^\top\Lambda^* + \beta V\Sigma^4V^\top.$$

对上式两端左乘 $V^\top$ 并右乘 V, 我们得到

$$V^\top X^{*\top}GV = V^\top X^{*\top}\nabla f(X^*)V - \beta\Sigma^2 - \Sigma^2\left(V^\top\Lambda^*V - \beta\Sigma^2\right).$$

对上式取算子 Φ 且利用 (5.15) 式, 我们有

$$\Phi(V^\top X^{*\top}GV) = (I_p - \Sigma^2)(\Phi(V^\top\Lambda^*V) - \beta\Sigma^2). \tag{5.20}$$

因为 $\beta > \left(||\nabla f(X^*)||_{\mathrm{F}}\cdot||X^*||_2 + \delta\right)/\sigma^2_{\min}(X^*)$, 我们得到

$$\beta\sigma^2_{\min}(X^*) \ge ||\nabla f(X^*)||_2\cdot||X^*||_2 + \delta,$$

由此推出

$$\sigma_{\min}(\beta\Sigma^2) \geq ||V^\top \Lambda^* V||_2 + \delta \geq ||\Phi(V^\top \Lambda^* V)||_2 + \delta.$$

从而,

$$\sigma_{\min}\left(\beta\Sigma^2 - \Phi(V^\top \Lambda^* V)\right) \geq \delta \tag{5.21}$$

成立. 将 (5.21) 式代入 (5.20) 式, 我们得到

$$\begin{aligned} ||X^*||_2||G||_{\mathrm{F}} &\geq ||\Phi(V^\top X^{*\top} G V)||_{\mathrm{F}} = ||(I_p - \Sigma^2)(\Phi(V^\top \Lambda^* V) - \beta\Sigma^2)||_{\mathrm{F}} \\ &\geq ||I_p - \Sigma^2||_{\mathrm{F}} \cdot \sigma_{\min}\left(\beta\Sigma^2 - \Phi(V^\top \Lambda^* V)\right) \geq ||I_p - X^{*\top} X^*||_{\mathrm{F}} \cdot \delta. \end{aligned}$$

因此, 引理得证. □

5.3 可并行算法

在本节中, 针对正交约束优化问题 (5.1), 我们提出一个可并行算法以及它的一个变种. 这两种方法都基于增广 Lagrange 函数 (5.2), 并且都采用了新的乘子显式更新来替代经典算法 6 中的乘子对偶上升步.

我们的算法与经典的增广 Lagrange 函数法的另一个不同之处是关于原始变量的更新, 在我们的算法中, 增广 Lagrange 子问题的目标函数由其邻近点线性化逼近得到.

5.3.1 邻近点线性化增广 Lagrange 算法

通常来讲, 在算法 6 中, 增广 Lagrange 子问题 (5.3) 一般都没有显式解. 有时, 甚至求解问题 (5.3) 都变得十分困难. 因此, 我们考虑非精确求解子问题 (5.3), 其主要思想是构造增广 Lagrange 函数的简单逼近. 给定当前迭代点 X^k, 我们定义如下的邻近点线性化增广 Lagrange 函数,

$$\tilde{\mathcal{L}}_\beta(X) = \mathrm{tr}(\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)^\top (X - X^k)) + \frac{\eta^k}{2}||X - X^k||_{\mathrm{F}}^2. \tag{5.22}$$

利用上述逼近以及新的乘子显式更新格式, 我们可以得到如下的邻近点线性化增广 Lagrange 算法.

算法 7 的主要计算代价集中在第 3 步和第 4 步. 其中, 步 3 只引入了基础的 BLAS3 (矩阵乘法) 代数运算. 而步 4 中的极小化子问题 (5.24) 本质上是一个梯度

算法 7: 邻近点线性化增广 Lagrange 算法 (PLAM)

1 **初始化:** 取 $X^0 \in \mathbb{R}^{n\times p}$, 令 $k := 0$;

2 **while** 停机准则不满足 **do**

3 计算 Lagrange 乘子

$$\Lambda^k := \Psi(\nabla f(X^k)^\top X^k). \tag{5.23}$$

4 极小化邻近点线性化增广 Lagrange 函数, 得到

$$X^{k+1} := \operatorname*{arg\,min}_{X\in\mathbb{R}^{n\times p}} \tilde{\mathcal{L}}_\beta(X). \tag{5.24}$$

5 令 $k := k+1$.

6 **返回:** X^k.

下降步,

$$\begin{aligned} X^{k+1} &= X^k - \frac{1}{\eta^k}\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k) \\ &= X^k - \frac{1}{\eta^k}\left(\nabla f(X^k) + \beta X^k\left(X^{k\top}X^k - I_p - \frac{1}{\beta}\Lambda^k\right)\right) \\ &= X^k - \frac{1}{\eta^k}\left(\nabla f(X^k) - X^k\Psi(\nabla f(X^k)^\top X^k) + \beta X^k(X^{k\top}X^k - I_p)\right), \end{aligned} \tag{5.25}$$

其中最后一步根据公式 (5.23) 得到. 显然, (5.25) 式中引入的代数运算也同样属于 BLAS3.

我们注意到 $1/\eta^k$ 本质上就是梯度步的步长. 因此, 邻近点参数 η^k 的选取方式可以和经典梯度法 (1.1.3 小节) 的步长选取类似. 在数值实验部分, 我们将详细讨论 η^k 的选取.

5.3.2 可并行的列极小化算法

算法 PLAM 一个明显的缺陷是, 当我们对罚参数 β 和邻近点参数 η^k 不加任何限制时, 迭代点列的有界性将很难得到保证. 理论上来讲, 为了保证算法的全局收敛性, 参数 β 的值需要充分大. 相应的, η^k 的值也要充分大才行, 这也就意味着我们采取了充分小的步长, 这将会导致算法收敛速度变慢. 事实上, 根据一些经验性的观察, 我们发现 PLAM 的表现与参数 β 和 η^k 的选取密切相关. 总而言之, 如何选取这两个参数使得算法 7 表现优异并不是一件很容易的事.

因此, 我们提出一个算法 PLAM 的改进版本. 新的算法基于 PLAM, 但是在

步 4 增加了冗余的列单位球约束. 由此, 算法得到的迭代点列将会限制在一个紧集上, 从而迭代点列有界. 首先, 我们注意到 $\tilde{\mathcal{L}}_\beta(X) = \sum_{i=1}^p \tilde{\mathcal{L}}_\beta^{(i)}(X_i)$, 其中

$$\tilde{\mathcal{L}}_\beta^{(i)}(x) := \nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)_i^\top (x - X_i^k) + \frac{\eta^k}{2}||x - X_i^k||_2^2.$$

算法 8 展示了改进后的 PLAM 算法.

算法 8: PLAM 的可并行列极小化算法 (PCAL)

1 **初始化:** 取 $X^0 \in \mathbb{R}^{n\times p}$, 令 $k := 0$;

2 **while** 停机准则不满足 **do**

3 由 (5.23) 式或

$$\Lambda^k := \Psi(\nabla f(X^k)^\top X^k) + \Phi\left(X^{k\top} \nabla_X L_\beta(X^k, \Psi(\nabla f(X^k)^\top X^k))\right), \quad (5.26)$$

计算 Lagrange 乘子.

4 **for** $i = 1, ..., p$ **do**

5 极小化邻近点线性化增广 Lagrange 函数

$$\begin{aligned} X_i^{k+1} := \arg\min_{x\in\mathbb{R}^n} \quad & \tilde{\mathcal{L}}_\beta^{(i)}(x) \\ \text{s. t.} \quad & ||x||_2 = 1. \end{aligned} \quad (5.27)$$

6 更新 $X^{k+1} = [X_1^{k+1}, \ldots, X_p^{k+1}]$, 令 $k := k + 1$.

7 **返回:** X^k.

算法 8 中的子问题 (5.27) 能以列并行化的方式并行求解. 事实上, 每个子问题都有显式解

$$X_i^{k+1} = \frac{X_i^k - \frac{1}{\eta^k}\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)_i}{\left\| X_i^k - \frac{1}{\eta^k}\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)_i \right\|_2}.$$

对于算法 PCAL, 我们可以采用 PLAM 同样的 Lagrange 乘子更新方式, 也就是公式 (5.23). 为了得到更好的算法表现, 我们也采取了启发式的更新公式 (5.26). 其具体形式由如下观察得到. 在一阶最优性条件 (5.5) 中, 我们强加一组额外的球约束, 由此得到

$$\begin{cases} \nabla f(X) = X\Lambda + XD; \\ X^\top X = I_p, \end{cases} \quad (5.28)$$

其中 $D \in \mathbb{D}^p$, 并且 D 由子问题 (5.27) 中变量 X_i 的 Lagrange 乘子决定.

5.3.3 计算量比较

<table>
<tr><td colspan="5">函数计算</td></tr>
<tr><td rowspan="3">$f(X) := \frac{1}{2}\mathrm{tr}(X^\top AX) + \mathrm{tr}(G^\top X)$</td><td rowspan="2">$AX$</td><td>$A$: 稠密</td><td>$A$: 稀疏</td><td>$A$: 稀疏</td></tr>
<tr><td>$2n^2p$</td><td>$O(np)$</td><td rowspan="3">$O(np)$</td></tr>
<tr><td>$\nabla f(X) = AX + G$</td><td colspan="2">np</td></tr>
<tr><td>$\frac{1}{2}\mathrm{tr}(X^\top AX) + \mathrm{tr}(G^\top X)$</td><td colspan="2">$4np$</td></tr>
<tr><td colspan="5">KKT: $\nabla f(X) - X\nabla f(X)^\top X$</td></tr>
<tr><td>$\nabla f(X)^\top X$</td><td colspan="3">$2np^2$</td><td rowspan="2">$4np^2 + np$</td></tr>
<tr><td>$X(\nabla f(X)^\top X)$</td><td colspan="3">$2np^2$</td></tr>
<tr><td colspan="5">可行性: $X^\top X - I$</td></tr>
<tr><td>$X^\top X$</td><td colspan="3">np^2</td><td>$np^2 + np$</td></tr>
<tr><td colspan="5">算法</td></tr>
<tr><td rowspan="2">PLAM</td><td>$X(X^\top X - I)$</td><td colspan="2">$2np^2$</td><td rowspan="2">$4np^2 + O(np)$</td></tr>
<tr><td>$X\Psi(\nabla f(X)^\top X)$</td><td colspan="2">$2np^2$</td></tr>
<tr><td rowspan="4">PCAL</td><td>$X(X^\top X - I)$</td><td colspan="2">$2np^2$</td><td rowspan="4">$4np^2 + O(np)$</td></tr>
<tr><td>$X\Psi(\nabla f(X)^\top X)$</td><td colspan="2">$2np^2$</td></tr>
<tr><td>$\Phi\left(X^{k\top}\nabla_X L_\beta(X^k, \Psi(\nabla f(X^k)^\top X^k))\right)$</td><td colspan="2">$O(np)$</td></tr>
<tr><td>$X\Lambda = X\Psi(\cdot) + X\Phi(\cdot)$</td><td colspan="2">$O(np)$</td></tr>
<tr><td rowspan="4">MOptQR (cholesky $LL^\top$)</td><td>$V := X - \tau(\nabla f(X) - X\nabla f(X)^\top X)$</td><td colspan="2">$2np$</td><td rowspan="4">$3np^2 + \boxed{O(p^3)} + O(np)$</td></tr>
<tr><td>$V^\top V$</td><td colspan="2">np^2</td></tr>
<tr><td>$\mathrm{chol}(V^\top V) = LL^\top$</td><td colspan="2">$\boxed{p^3/3}$</td></tr>
<tr><td>$VL^{-\top}$</td><td colspan="2">$2np^2 + \boxed{O(p^3)}$</td></tr>
<tr><td>MOptQR (Gram-Schmidt)</td><td colspan="3">$2np^2$</td><td>$\boxed{2np^2 + O(np)}$</td></tr>
<tr><td colspan="5">总计</td></tr>
<tr><td>PLAM</td><td colspan="4">$7np^2 + O(np)$</td></tr>
<tr><td>PCAL</td><td colspan="4">$7np^2 + O(np)$</td></tr>
<tr><td>MOptQR</td><td colspan="4">cholesky 算法: $7np^2 + \boxed{O(p^3)} + O(np)$, Gram-Schmidt 算法: $4np^2 + \boxed{2np^2 + O(np)}$</td></tr>
</table>

表 5.1 计算量的比较

Table 5.1 The comparison of computational cost

在本小节中, 我们比较不同算法之间的每步迭代计算量, 基础的线性代数操作计算量和总计算量列在表 5.1 中.

其中, 用方框标记的项都是无法进行并行计算的. 从表 5.1 中我们发现, 在理论的并行可扩展性方面, 我们的新算法比已有的可行方法更具优势. 在实际中, 对 KKT 的计算, 我们采用 $X\Psi(\nabla f(X)^\top X)$ 而不是 $X(\nabla f(X)^\top X)$, 这是由于在任意一阶稳定点附近它们已经非常相近, 由此, 我们可以节省 $2np^2$ 个浮点运算的计算量.

5.4 收敛性分析

在本节中我们主要讨论算法 PLAM 的收敛性分析. 在相应的合理假设下, 我们分析了算法的全局收敛性, 最坏情况下的复杂度和 Q-线性局部收敛速度.

5.4.1 PLAM 的全局收敛性

为了证明算法 7 的收敛性, 除了假设 5.1 之外, 我们还需对初始值及参数 β 和 η^k 做一些合理的假设. 为了叙述方便, 接下来我们先给出这些条件.

假设 5.2. 对于给定的 X^0, 如果存在 $\underline{\sigma} \in (0,1)$ 使得

$$\sigma_{\min}(X^0) \geq \underline{\sigma}, \qquad 0 < ||X^{0\top}X^0 - I_p||_{\mathrm{F}} \leq 1 - \underline{\sigma}^2,$$

则我们称其为一个合格的初始值.

上述假设的要求并不苛刻, 在实际中可以很容易构造满足假设的初始点 X^0. 接下来, 我们给出两类初始点的选取方法.

例 5.1. 第一类: $X^0 = Q\Sigma$, 其中 $Q \in \mathcal{S}_{n,p}$, $\Sigma = \mathrm{Diag}(\underbrace{1, ..., 1}_{p-1}, \underline{\sigma})$, 任意给定 $\underline{\sigma} \in (0,1)$. 可以验证 $\sigma_{\min}(X^0) = \underline{\sigma}$ 且 $||X^{0\top}X^0 - I_p||_{\mathrm{F}} = 1 - \underline{\sigma}^2 > 0$.

例 5.2. 第二类: $X^0 \notin \mathcal{S}_{n,p}$ 满足 $\sigma^2_{\min}(X^0) > 1 - \frac{1}{\sqrt{p}}$ 且 $\sigma^2_{\max}(X^0) < 1 + \frac{1}{\sqrt{p}}$. 在这种情况下, $0 < ||X^{0\top}X^0 - I_p||_{\mathrm{F}}$ 立即成立. 令

$$\underline{\sigma} = \sqrt{\min\left\{1 - \frac{1}{\sqrt{p}},\ \sqrt{p}\left(1 + \frac{1}{\sqrt{p}} - \sigma^2_{\max}(X^0)\right),\ \sqrt{p}\left(\sigma^2_{\min}(X^0) - 1 + \frac{1}{\sqrt{p}}\right)\right\}},$$

则不难推出 $\sigma_{\min}(X^0) > \underline{\sigma} > 0$. 进一步, 我们有

$$\begin{aligned} ||X^{0\top}X^0 - I_p||_{\mathrm{F}} &\leq \sqrt{p} \cdot ||X^{0\top}X^0 - I_p||_2 \\ &\leq \sqrt{p} \cdot \sqrt{\max\{\lambda^2_{\max}(X^{0\top}X^0 - I_p),\ \lambda^2_{\min}(X^{0\top}X^0 - I_p)\}}. \end{aligned}$$

由

$$\begin{aligned} \lambda_{\max}(X^{0\top}X^0 - I) &= \lambda_{\max}(X^{0\top}X^0) - 1 = \sigma^2_{\max}(X^0) - 1 \leq \frac{1}{\sqrt{p}} - \frac{\underline{\sigma}^2}{\sqrt{p}}, \\ \lambda_{\min}(X^{0\top}X^0 - I) &= \lambda_{\min}(X^{0\top}X^0) - 1 = \sigma^2_{\min}(X^0) - 1 \geq \frac{\underline{\sigma}^2}{\sqrt{p}} - \frac{1}{\sqrt{p}}, \end{aligned}$$

我们得到 $||X^{0\top}X^0 - I_p||_{\mathrm{F}} \leq \sqrt{p} \cdot \left(\frac{1}{\sqrt{p}} - \frac{\underline{\sigma}^2}{\sqrt{p}}\right) = 1 - \underline{\sigma}^2$.

接下来, 我们列出在本节中用到的特殊记号.

$$\begin{aligned} &R = ||X^{0\top}X^0 - I_p||_{\mathrm{F}}; \quad C = \{X \mid ||X^{\top}X - I_p||_{\mathrm{F}} \leq R\}; \quad \underline{f} = \min_{X \in C} f(X); \\ &M = \max_{X \in C} ||X||_2; \quad N = \max_{X \in C} ||\nabla f(X)||_{\mathrm{F}}; \quad L = \max_{X \in C} ||\nabla^2 f(X)||_2. \end{aligned} \tag{5.29}$$

我们引入如下的评价函数

$$h(X) := f(X) - \frac{1}{2}\left\langle \Psi(\nabla f(X)^\top X), X^\top X - I_p \right\rangle + \frac{\beta}{4}||X^\top X - I_p||_{\mathrm{F}}^2. \tag{5.30}$$

假设函数 $f(X)$ 二次连续可微, 则我们有 $\nabla f(X)$ 在紧集 C 上 Lipschitz 连续. 也就是, 存在常数 $L_h > 0$, 与 β 有关, 使得

$$||\nabla h(X) - \nabla h(y)||_{\mathrm{F}} \le L_h||X - Y||_{\mathrm{F}}, \quad \forall\, X, Y \in C. \tag{5.31}$$

算法的参数 β 和 η^k, 以及其他在证明中用到的常数定义如下.

假设 5.3.

$$c_1 \in \left(0, \frac{1}{2}\right);\ \beta > \max\left\{\frac{MN}{\underline{\sigma}^2} + \sqrt{\frac{M^2N^2}{\underline{\sigma}^4} + \frac{(N+LM)^2}{4\underline{\sigma}^2(1-2c_1)}}, \frac{MN}{\underline{\sigma}}, \frac{4MN}{\underline{\sigma}^2}\right\} \tag{5.32}$$

$$c_2 \in \left(0, \frac{R^2(\beta\underline{\sigma}^2 - 4MN)}{2N_L^2}\right]; \quad \eta^k \in \left[\underline{\eta}, \bar{\eta}\right], \tag{5.33}$$

$$\text{其中 } \underline{\eta} = \max\left\{\frac{L_h}{2c_1}, \frac{2N_LM + N_L\sqrt{4M^2+2R}}{R}, \frac{R+2M^2}{c_2}\right\},$$

$$N_L = (1+M^2)N + \beta RM, \quad \bar{\eta} \ge \underline{\eta}.$$

注 5.2. 假设 *5.3* 中的条件适用于理论分析. 在实际中, 满足条件的参数 β 和 η^k 将会非常局限.

接下来我们列出算法的收敛性证明框架. 假设 $\{X^k\}$ 是由算法 7 生成的迭代点列. 收敛性证明的步骤包括:

1) 任意迭代点 X^k 都属于 C, 且 $\underline{\sigma}$ 是迭代点 X^k 奇异值的一个一致下界;

2) 评价函数 $h(X)$ 下有界;

3) $\{h(X^k)\}$ 单调下降, 因此收敛;

4) $\{X^k\}$ 的任意聚点, 记为 X^*, 都是增广 Lagrange 问题 (5.8) 的一阶稳定点, 其中 $\Lambda^* = \Psi(\nabla f(X^*)^\top X^*)$.

5) $\{X^k\}$ 的任意聚点, 记为 X^*, 都是原正交约束优化问题 (5.1) 的一阶稳定点.

接下来, 我们给出引理和推论来说明上述证明步骤成立.

引理 5.3. 令 $\{X^k\}$ 是由算法 *7* 从初始点 X^0 生成的迭代点列, 其中初始点 X^0 满足假设 *5.2* 并且问题的参数满足假设 *5.3*. 则我们有

$$\sigma_{\min}(X^k) \ge \underline{\sigma}, \quad X^k \in C. \tag{5.34}$$

证明. 我们采用数学归纳法. 首先, 由假设 5.2 我们立即可得条件 (5.34) 对于 X^0 成立. 接下来我们研究当条件 (5.34) 对 X^k 成立时, (5.34) 是否对 X^{k+1} 成立. 我们分两种情况讨论.

情况一, $||{X^k}^\top X^k - I_p||_{\mathrm{F}} \le \frac{R}{2}$. 这时, 我们有

$$\begin{aligned} & ||{X^{k+1}}^\top X^{k+1} - I_p||_{\mathrm{F}} \\ = & \left\| \left(X^k - \frac{1}{\eta^k}\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k) \right)^\top \left(X^k - \frac{1}{\eta^k}\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k) \right) - I_p \right\|_{\mathrm{F}} \\ \le & ||{X^k}^\top X^k - I_p||_{\mathrm{F}} + \frac{2}{\eta^k}||X^k||_2||\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)||_{\mathrm{F}} + \frac{1}{(\eta^k)^2}||\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)||_{\mathrm{F}}^2. \end{aligned}$$

通过验算, 我们不难得到

$$\begin{aligned} ||\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)||_{\mathrm{F}} &= \left\| \nabla f(X^k) - X^k \Psi(\nabla f(X^k)^\top X^k) + \beta X^k ({X^k}^\top X^k - I_p) \right\|_{\mathrm{F}} \\ &\le (1 + M^2)N + \beta RM = N_L \end{aligned}$$

对任意的 $X^k \in C$ 都成立. 利用 $X^k \in C$, (5.29) 和 (5.33), 我们有

$$\frac{2}{\eta^k}||X^k||_2||\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)||_{\mathrm{F}} + \frac{1}{(\eta^k)^2}||\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)||_{\mathrm{F}}^2 \le \frac{R}{2},$$

由此推出 $||{X^{k+1}}^\top X^{k+1} - I_p||_{\mathrm{F}} \le R$. 这也就证明了 (5.34) 对 $k+1$ 成立.

情况二, $||{X^k}^\top X^k - I_p||_{\mathrm{F}} > \frac{R}{2}$. 为了简便起见, 我们记 $c(X) = \frac{1}{2}||X^\top X - I_p||_{\mathrm{F}}^2$,

$$d = \nabla f(X^k) - X^k \Lambda^k, \quad C = {X^k}^\top X^k - I_p, \quad \delta = X^k C. \tag{5.35}$$

根据 $\sigma_{\min}(X^k) \ge \underline{\sigma}$ 和 $X^k \in C$, 我们有

$$\|\delta\|_{\mathrm{F}} > \frac{R\underline{\sigma}}{2}. \tag{5.36}$$

又因为当 A 对称时, $\mathrm{tr}(AB) = \mathrm{tr}(AB^\top)$ 成立, 从而我们有

$$\mathrm{tr}(C{X^k}^\top \nabla f(X^k)) = \mathrm{tr}(C\nabla f(X^k)^\top X^k) = \mathrm{tr}(C\Lambda^k).$$

因此, 我们推出

$$\begin{aligned} \langle d, \delta \rangle &= \mathrm{tr}(C{X^k}^\top \nabla f(X^k) - C{X^k}^\top X^k \Lambda^k) \\ &= \mathrm{tr}(C{X^k}^\top \nabla f(X^k) - C(C + I_p)\Lambda^k) = -\mathrm{tr}(C^2 \Lambda^k). \end{aligned} \tag{5.37}$$

注意到 $L_c = 2R + 4M^2$ 是 $\nabla c(X)$ 在 C 上的 Lipschitz 常数. 结合 (5.33), (5.36) 和 (5.37) 式, 我们有

$$\begin{aligned} & \operatorname{tr}\left(\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)^\top \nabla c(X^k)\right) - c_2 ||\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)||_{\mathrm{F}}^2 \\ \geq & \ 2\langle d + \beta\delta, \delta\rangle - c_2 N_L^2 = 2\beta \|\delta\|_{\mathrm{F}}^2 + 2\langle d, \delta\rangle - c_2 N_L^2 \\ > & \ \frac{\beta R^2 \underline{\sigma}^2}{2} - 2\|C\|_{\mathrm{F}}^2 \cdot \operatorname{tr}(\Lambda^k) - c_2 N_L^2 \\ \geq & \ \frac{\beta R^2 \underline{\sigma}^2}{2} - 2R^2 MN - c_2 N_L^2 \geq 0. \end{aligned}$$

根据 Taylor 展开, 我们得到

$$\begin{aligned} & c(X^{k+1}) = c\left(X^k - \frac{1}{\eta^k}\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)\right) \\ \leq & \ c(X^k) - \frac{1}{\eta^k}\left\langle \nabla_X \mathcal{L}_\beta(X^k, \Lambda^k), \nabla c(X^k)\right\rangle + \frac{L_c}{2(\eta^k)^2}||\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)||_{\mathrm{F}}^2 \\ < & \ c(X^k) - \left(\frac{c_2}{\bar{\eta}} - \frac{L_c}{2\bar{\eta}^2}\right) \cdot ||\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)||_{\mathrm{F}}^2 \leq c(X^k). \end{aligned}$$

结合假设以及 $R \leq 1 - \underline{\sigma}^2$, 我们不难得到 $\sigma_{\min}(X^{k+1}) \geq \underline{\sigma}$. 由此引理得证. □

引理 5.4. 由 (5.30) 式定义的评价函数 $h(X)$ 在 C 上下有界.

此引理可由 $h(X)$ 的连续可微性以及 C 的紧性立得, 故我们省略证明过程.

引理 5.5. 令 $\{X^k\}$ 是由算法 7 从初始点 X^0 生成的迭代点列, 其中初始点 X^0 满足假设 5.2 并且问题的参数满足假设 5.3. $h(X)$ 由 (5.30) 式定义. 则我们有

$$h(X^k) - h(X^{k+1}) \geq c_3 \cdot ||\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)||_{\mathrm{F}}^2, \tag{5.38}$$

其中 $c_3 = \frac{c_1}{\bar{\eta}} - \frac{L_h}{2\bar{\eta}^2} > 0$.

证明. 首先, 我们注意到

$$\nabla h(X) = \nabla_X \mathcal{L}_\beta(X, \Psi(\nabla f(X)^\top X)) - \frac{1}{2}(\nabla^2 f(X)[X] + \nabla f(X))(X^\top X - I_p).$$

在此引理的证明中, 我们继续使用 (5.35) 的记号. 通过计算, 我们发现

$$\begin{aligned} & ||\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)||_{\mathrm{F}}^2 - \frac{1}{1 - 2c_1}||\nabla h(X^k) - \nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)||_{\mathrm{F}}^2 \\ \geq & \ ||d + \beta\delta||_{\mathrm{F}}^2 - \frac{(N + LM)^2}{4(1 - 2c_1)}||C||_{\mathrm{F}}^2 \ \geq \ 2\beta\langle d, \delta\rangle + \beta^2||\delta||_{\mathrm{F}}^2 - \frac{(N + LM)^2}{4(1 - 2c_1)}||C||_{\mathrm{F}}^2 \\ \geq & \ -\beta\|C\|_{\mathrm{F}}^2 \cdot \operatorname{tr}(\Lambda^k) + \left(\beta^2 \underline{\sigma}^2 - \frac{(N + LM)^2}{4(1 - 2c_1)}\right) \cdot ||C||_{\mathrm{F}}^2 \\ \geq & \ -2\beta MN||C||_{\mathrm{F}}^2 + \left(\beta^2 \underline{\sigma}^2 - \frac{(N + LM)^2}{4(1 - 2c_1)}\right) \cdot ||C||_{\mathrm{F}}^2 \geq 0, \end{aligned}$$

其中, 倒数第二个不等式由关系式 (5.37) 推出. 因此, 我们得到

$$\langle \nabla_X \mathcal{L}_\beta(X^k, \Lambda^k), \nabla h(X^k) \rangle \geq c_1 ||\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)||_{\mathrm{F}}^2. \tag{5.39}$$

将 (5.33) 和 (5.39) 式代入 Taylor 展开, 我们有

$$\begin{aligned} & h(X^{k+1}) = h\left(X^k - \frac{1}{\eta^k}\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)\right) \\ \leq\ & h(X^k) - \frac{1}{\eta^k}\left\langle \nabla_X \nabla h(X^k), \mathcal{L}_\beta(X^k, \Lambda^k)\right\rangle + \frac{L_h}{2(\eta^k)^2}||\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)||_{\mathrm{F}}^2 \\ \leq\ & h(X^k) - \left(\frac{c_1}{\bar{\eta}} - \frac{L_h}{2\bar{\eta}^2}\right) \cdot ||\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)||_{\mathrm{F}}^2. \end{aligned}$$

由此引理得证. □

结合 $h(X)$ 在 C 上的有界性, 引理 5.5 可直接推出 $\{h(X^k)\}$ 的收敛性. 具体来讲, 我们有如下的推论.

推论 5.1. *令 $\{X^k\}$ 是由算法 7 从初始点 X^0 生成的迭代点列, 其中初始点 X^0 满足假设 5.2 并且问题的参数满足假设 5.3. 则算法 k 步有限终止得到 $\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k) = 0$, 或者*

$$\lim_{k\to+\infty} \nabla_X \mathcal{L}_\beta(X^k, \Lambda^k) = 0.$$

进一步, $\{X^k\}$ 至少有一个收敛子序列. $\{X^k\}$ 的任意聚点, 记做 X^*, 都是增广 Lagrange 问题当 $\Lambda^* = \Psi(\nabla f(X^*)^\top X^*)$ 时的一阶稳定点.

证明. 引理 5.3 和 5.5 的直接推论. □

最后, 我们给出算法 PLAM 的全局收敛速度, 也就是最坏情况下的复杂度.

定理 5.1. *令 $\{X^k\}$ 是由算法 7 从初始点 X^0 生成的迭代点列, 其中初始点 X^0 满足假设 5.2 并且问题的参数满足假设 5.3. 则序列 $\{X^k\}$ 至少有一个聚点, 且任意聚点都是原正交约束优化问题 (5.1) 的一阶稳定点. 进一步, 对于任意的 $K > 1$, 下式成立,*

$$\min_{k=0,\ldots,K-1} ||\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)||_{\mathrm{F}} < \sqrt{\frac{f(X^0) - \underline{f} + MNR + \beta R^2/4}{c_3 K}}. \tag{5.40}$$

证明. 定理的第一部分可由推论 5.1 和引理 5.2 直接得到. 在引理 5.5 中, 我们有

$$\begin{aligned} h(X^0) - \min_{X\in C} h(X) &\geq h(X^0) - h(X^K) \geq \sum_{k=0}^{K-1} c_3 ||\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)||_{\mathrm{F}}^2 \\ &\geq c_3 K \cdot \min_{k=0,\ldots,K-1} ||\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k)||_{\mathrm{F}}^2. \end{aligned} \tag{5.41}$$

进一步, 我们得到

$$h(X^0) \leq f(X^0) + \frac{1}{2}MNR + \frac{\beta}{4}R^2, \quad \min_{X\in C} h(X) \geq \underline{f} - \frac{1}{2}MNR. \tag{5.42}$$

结合不等式 (5.41) 和 (5.42), 我们有 (5.40) 式成立. □

推论 5.2. *若定理 5.1 的所有假设都满足, 并且对给定的 $\beta > (MN+\delta)/\underline{\sigma}$, $\delta > 0$ 定理成立. 则我们有*

$$\begin{aligned} &\min_{k=0,\ldots,K-1} \max\left\{||I_p - X^{k\top}X^k||_{\mathrm{F}}, ||\nabla_X \mathcal{L}_\beta(X^k,\Lambda^k)||_{\mathrm{F}}\right\} \\ < \;& \max\left\{\frac{M}{\delta}, 1\right\}\sqrt{\frac{f(X^0) - \underline{f} + MNR + \beta R^2/4}{c_3 K}}. \end{aligned}$$

证明. 引理 5.2 和定理 5.1的直接推论. □

注 5.3. *推论 5.2 中的次线性收敛速度告诉我们, 当停机准则设为*

$$\max\left\{||I_p - X^{k\top}X^k||_{\mathrm{F}}, ||\nabla_X \mathcal{L}_\beta(X^k,\Lambda^k)||_{\mathrm{F}}\right\} < \epsilon$$

时, 算法 7 至多需要 $O(1/\epsilon^2)$ 次迭代后终止.

5.4.2 PLAM 和 PCAL 的局部收敛速度

在本小节, 我们考虑当正交约束优化问题 (5.1) 有一个孤立的局部极小值点时, 算法 PLAM 的局部收敛性.

定理 5.2. *假设 X^* 是问题 (5.1) 的一个孤立局部极小值点, 同时我们记*

$$\tau := \inf_{0\neq Y\in\mathcal{T}_X\mathcal{S}_{n,p}} \frac{\mathrm{tr}(Y^\top\nabla^2 f(X)[Y] - \Lambda Y^\top Y)}{||Y||_{\mathrm{F}}^2}.$$

算法的参数满足 $\beta \geq \frac{L+MN+\tau}{2}$ 和 $\eta^k \in [\underline{\eta}, \bar{\eta}]$, 其中 $\bar{\eta} \geq \underline{\eta} \geq L + MN + 2\beta$. 则存在 $\varepsilon > 0$ 使得, 算法 7 从任意满足 $||X^0 - X^*||_{\mathrm{F}} < \varepsilon$ 的初始点 X^0 出发生成的迭代点列 $\{X^k\}$, 以 Q-线性速度收敛到 X^* .

证明. 首先, 我们考虑迭代点更新公式 (5.24).

$$\begin{aligned} X^{k+1} &= X^k - \frac{1}{\eta^k}\nabla_X \mathcal{L}_\beta(X^k, \Psi(\nabla f(X^k)^\top X^k)); \\ X^* &= X^* - \frac{1}{\eta^k}\nabla_X \mathcal{L}_\beta(X^*, \Psi(\nabla f(X^*)^\top X^*)). \end{aligned}$$

记 $\delta^k = X^k - X^*$, 通过一式减去二式并利用 Taylor 展开, 我们得到

$$\delta^{k+1} = \delta^k - \frac{1}{\eta^k}\nabla^2_{XX}\mathcal{L}_\beta(X^*, \Psi(\nabla f(X^*)^\top X^*))[\delta^k] + o(||\delta^k||), \tag{5.43}$$

结合 Hesse 算子的表达式 (5.12), 性质 $\nabla f(X^*)^\top X^* = \Psi(\nabla f(X^*)^\top X^*)$ 以及 η 的假设, 我们推出

$$\left\| \frac{1}{\eta^k} \nabla^2_{XX} \mathcal{L}_\beta(X^*, \nabla f(X^*)^\top X^*)[\delta^k] \right\|_{\mathrm{F}} \leq ||\delta^k||_{\mathrm{F}}. \tag{5.44}$$

另一方面, δ^k 能被分解为三项的和,

$$\delta^k = X^* S + X^* W + K, \tag{5.45}$$

其中 $S \in \mathbb{R}^{p\times p}$ 对称, $W \in \mathbb{R}^{p\times p}$ 反对称, $K \in \mathbb{R}^{n\times p}$ 与 X^* 垂直. 因为 X^* 是严格局部极小值点, 且 $\mathcal{T}_X\mathcal{S}_{n,p}$ 是闭集, 则我们有 $\tau > 0$. 因此, 当 $X^*W + K \in \mathcal{T}_X\mathcal{S}_{n,p}$, 下式成立

$$\mathrm{tr}\left((X^*W + K)^\top \nabla^2_{XX} \mathcal{L}_\beta(X^*, \nabla f(X^*)^\top X^*)[X^*W + K]\right) \geq \tau ||X^*W + K||^2_{\mathrm{F}}, \tag{5.46}$$

进一步, 由 β 的假设我们可以推出

$$\begin{aligned} & \mathrm{tr}\left((X^*S)^\top \nabla^2_{XX} \mathcal{L}_\beta(X^*, \nabla f(X^*)^\top X^*)[X^*S]\right) \\ = \ & \mathrm{tr}(SX^*\nabla^2 f(X^*)X^*S - S^2 \nabla f(X^*)^\top X^* + 2\beta S^2) \geq \tau ||X^*S||^2_{\mathrm{F}}. \end{aligned} \tag{5.47}$$

结合 (5.46), (5.47), S 的对称性, W 的反对称性, $K^\top X^* = 0$ 和 η 的假设, 我们得到

$$\begin{aligned} & \mathrm{tr}\left(\delta^{k\top} \nabla^2_{XX} \mathcal{L}_\beta(X^*, \nabla f(X^*)^\top X^*)[\delta^k]\right) \\ = \ & \mathrm{tr}\left((X^*W + K)^\top \nabla^2_{XX} \mathcal{L}_\beta(X^*, \nabla f(X^*)^\top X^*)[X^*W + K]\right) \\ & + \mathrm{tr}\left((X^*W + K)^\top \nabla^2_{XX} \mathcal{L}_\beta(X^*, \nabla f(X^*)^\top X^*)[X^*S]\right) \\ & + \mathrm{tr}\left((X^*S)^\top \nabla^2_{XX} \mathcal{L}_\beta(X^*, \nabla f(X^*)^\top X^*)[X^*W + K]\right) \\ & + \mathrm{tr}\left((X^*S)^\top \nabla^2_{XX} \mathcal{L}_\beta(X^*, \nabla f(X^*)^\top X^*)[X^*S]\right) \\ \geq \ & \tau ||X^*W + K||^2_{\mathrm{F}} + \tau ||X^*S||^2_{\mathrm{F}} = \tau ||\delta^k||^2_{\mathrm{F}}. \end{aligned} \tag{5.48}$$

我们注意到 (5.44) 式可推出线性算子

$$I - \frac{1}{\eta^k} \nabla^2_{XX} \mathcal{L}_\beta(X^*, \nabla f(X^*)^\top X^*)$$

的半正定性. 结合 (5.48) 式, 我们得出

$$||\delta^{k+1}||_{\mathrm{F}} \leq (1 - \tau)||\delta^k||_{\mathrm{F}} + o(||\delta^k||),$$

由此定理得证. □

注 5.4. *当算法 PCAL 的乘子更新采用和 PLAM 相同的更新公式 (5.23) 时, 同理可得 PCAL 的全局收敛性和局部收敛性.*

5.5 数值实验

在本节, 我们测试算法 PLAM 和 PCAL 的数值表现. 首先, 在 5.5.1 和 5.5.2 小节中, 我们分别介绍了算法的实现细节和测试问题. 接着我们从如下的两个方面展示我们的数值实验.

在第一部分, 也就是 5.5.3 小节, 我们主要在串行环境下确定新算法的默认设置, 其中包括参数的不同选取. 相应的测试环境是一台工作站, 其中处理器是一块 2.70GHz×12, 30M 缓存的 Intel® E5-2697 v2, 内存是 128GB. 数值实验的运行环境是 Ubuntu 12.04 下的 MATLAB R2016b.

第二部分, 在并行环境下, 通过和并行版本的 MOptQR 进行比较, 我们研究了算法 PCAL 的并行效率. 5.5.5 小节的全部数值实验都是在 LSSC-IV 集群[1]的一个单节点编译并运行的. 中国科学院“科学与工程计算国家重点实验室”的 LSSC-IV 四号集群系统是一个高性能计算平台. LSSC-IV 的操作系统为 Red Hat Enterprise Linux Server 7.3. 我们测试所使用的节点称为 “b01” , 其包含有两块 2.20GHz×24, 60M 缓存的 Intel® E7-8890 v4 处理器, 并拥有 4TB 的共享内存和总计 96 个处理器核.

5.5.1 算法实现细节

在我们的算法 PLAM 和 PCAL 中有两个参数需要选取. 根据定理 5.1, PLAM 的罚参数选取需要充分大. 尽管我们给出了满足定理假设的参数 β 的合理估计, 但在实际计算中, 这个值还是过大, 影响了算法的收敛速度. 因此, 在接下来的数值实验中, 对于 PLAM, 我们选取 β 为 $s := ||\nabla^2 f(0)||_2$ 的一个上界. 对于 PCAL, 我们令 β 为常数 1.

在算法 7 和 8 中, 另一个重要的参数是邻近点参数 η, 其倒数恰好是梯度步的步长. 类似于 β 的选取, 我们不宜直接采用理论分析中的严格限制. 例如, 当迭代点列靠近最优点的邻域时, 参数 η^k 的值经常可以用 $||\nabla^2_X \mathcal{L}_\beta(X^k, \Lambda^k)||_{\mathrm{F}}$ 来逼近. 在实际中, 基于不同的选取方式, 我们有如下的 η^k 选取策略:

(i) $\eta^k_{\mathrm{C}} := \gamma$, 其中 $\gamma > 0$ 是一个充分大的常数.

(ii) 差分逼近:

$$\eta^k_{\mathrm{D}} := \frac{||\nabla_X \mathcal{L}_\beta(X^k, \Lambda^k) - \nabla_X \mathcal{L}_\beta(X^{k-1}, \Lambda^{k-1})||_F}{||X^k - X^{k-1}||_{\mathrm{F}}}.$$

[1]更多信息请参考 http://lsec.cc.ac.cn/chinese/lsec/LSSC-IVintroduction.pdf.

(iii) BB 方法 (参考 1.1.3 小节):

$$\eta_{\mathrm{BB1}}^k := \frac{\left|\left\langle S^{k-1}, Y^{k-1}\right\rangle\right|}{\left\langle S^{k-1}, S^{k-1}\right\rangle}, \quad \text{或} \quad \eta_{\mathrm{BB2}}^k := \frac{\left\langle Y^{k-1}, Y^{k-1}\right\rangle}{\left|\left\langle S^{k-1}, Y^{k-1}\right\rangle\right|},$$

其中

$$S^k = X^k - X^{k-1}, \quad Y^k = \nabla_X \mathcal{L}_\beta(X^k, \Lambda^k) - \nabla_X \mathcal{L}_\beta(X^{k-1}, \Lambda^{k-1}).$$

(iv) 交替 BB 步长 (3.40):

$$\eta_{\mathrm{ABB}}^k := \begin{cases} \eta_{\mathrm{BB1}}^k, & k\text{是奇数}, \\ \eta_{\mathrm{BB2}}^k, & k\text{是偶数}. \end{cases}$$

在没有特别提及时, 串行和并行实验的停机准则都如下所示.

$$\frac{\left\|\nabla f(X) - X\nabla f(X)^\top X\right\|_{\mathrm{F}}}{\left\|\nabla f(X^0) - X^0\nabla f(X^0)^\top X^0\right\|_{\mathrm{F}}} < 10^{-8}.$$

所有算法的最大总迭代数都设为 3000.

5.5.2 测试问题

在本小节, 我们介绍用于数值实验的四类不同的测试问题.

问题 1: 简化的离散 Kohn-Sham 总能量极小化问题.

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & \tfrac{1}{2}\mathrm{tr}(X^\top L X) + \tfrac{\alpha}{4}\rho(X)^\top L^\dagger \rho(X) \\ \text{s. t.} \quad & X^\top X = I_p, \end{aligned} \tag{5.49}$$

其中矩阵 $L \in \mathbb{S}^n$ 且 $\rho(X) := \mathrm{diag}(XX^\top)$. 在数值实验中, 我们令 $\alpha = 1$, 并且 L 由高斯分布随机生成, 也就是采用 MATLAB 函数, `L=randn(n)`. 此外, 我们还对 L 进行对称化处理, $L := \frac{1}{2}(L + L^\top)$. 在这个例子中, 我们取 $s = \|L\|_2$.

问题 2: 一类带有正交约束的二次规划.

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & \tfrac{1}{2}\mathrm{tr}(X^\top A X) + \mathrm{tr}(G^\top X) \\ \text{s.t.} \quad & X^\top X = I_p, \end{aligned} \tag{5.50}$$

其中矩阵 $A \in \mathbb{S}^n$ 和 $G \in \mathbb{R}^{n\times p}$. 此问题已在第 3 章中有过详细讨论. 在本章的数值实验中, 我们选取和第 3 章相同的问题生成方式. 在这个例子中, 我们取 $s = \|A\|_2$.

问题 3: Rayleigh-Ritz 迹极小化, 这也是问题 2 的特殊情况.

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & \tfrac{1}{2}\mathrm{tr}(X^\top A X) \\ \text{s.t.} \quad & X^\top X = I_p, \end{aligned} \tag{5.51}$$

其中矩阵 $A \in \mathbb{S}^n$. 在数值实验中, 矩阵 A 的生成与问题 2 相同. 在这个例子中, 我们取 $s = \|A\|_2$.

问题 4: 另一类带有正交约束的二次规划.

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & \tfrac{1}{2}\mathrm{tr}(A^\top X B X^\top) \\ \text{s.t.} \quad & X^\top X = I_p. \end{aligned} \tag{5.52}$$

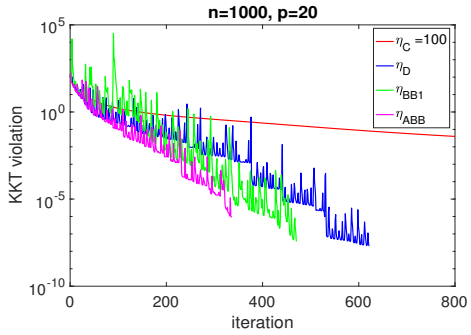
其中矩阵 $A \in \mathbb{S}^n$ 且 $B \in \mathbb{S}^p$. 通过观察我们发现, 尽管此问题并不满足第 3 章的问题假设, 但我们仍可使用本章的算法 PLAM 或者 PCAL. 矩阵 A 和 B 由高斯分布随机生成, 也就是, $A =$`randn(n)`, $A := \frac{1}{2}(A+A^\top)$ 且 $B =$`randn(p)`, $B := \frac{1}{2}(B+B^\top)$. 在这个例子中, 我们取 $s = \|A\|_2 \cdot \|B\|_2$.

5.5.3 算法默认参数选取设置

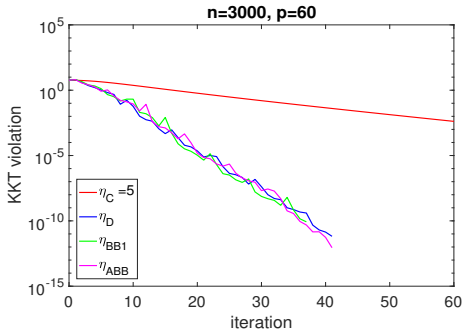
在本小节, 我们确定新算法 PLAM 和 PCAL 的默认参数设置.

在第一个实验中, 我们测试 η^k 采用四种不同选取方式的算法 PLAM 和 PCAL, 测试问题是问题 1-4. 在策略 (iii) 中, 由于采用 η_{BB1} 的算法数值表现优于 η_{BB2}, 故在此策略中我们只展示 η_{BB1} 的数值结果. 这里, 罚参数被固定选取为 $\beta = s + 0.1$. 图 5.1 展示了 PLAM 和 PCAL 在不同 η^k 选取下的数值结果. 从子图 (a)-(d) 中, 我们观察到采用 η_{ABB} 选取的 PLAM 有相对较好的表现. 在同样的设定下, 关于 PCAL 采用不同 η^k 选取方式的结果展示在子图 (e)-(h) 中. 我们注意到采用 η_{ABB} 的 PCAL 算法数值表现优于其他选取. 综上, 我们选取 η_{ABB} 作为算法 PLAM 和 PCAL 的默认设置.

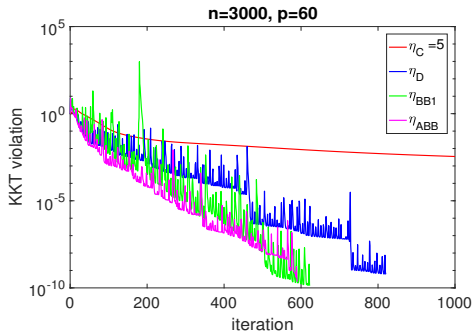
接下来我们比较不同参数 β 设定下算法 PLAM 和 PCAL 的数值表现. 在实验中, 我们令 β 的选取集合是 $\{0, 0.01s, 0.1s, s+0.1, 10s+1\}$. 为了平衡不同选取之间的差别, 在初始设定中我们选取一个相对大的 s 作为我们的默认参数. 例如, 在问题 1 中, 当 $\|L\|_2 > 1$ 时, 我们选取 $s = \|L\|_2^2 + 0.5$, 否则我们选取 $s = 1/\|L\|_2 + 0.1$. 邻近点参数固定为默认值 $\eta = \eta_{\mathrm{ABB}}$. 所有的数值结果都呈现在图 5.2 中. 我们注意到在子图 (a)-(d) 中, 若 PLAM 选取相对较小的 β 值, 则其在某些例子将不会收敛. 若选取较大的 β 值, 则会导致更慢的收敛速度. 因此, 在实际中一个恰好合适的 β 值对于 PLAM 的数值表现至关重要. 另一方面, 算法 PCAL 对于 β 的依赖性可以从子图 (e)-(h) 中发现. 在某些例子中, 采用越小的 β 值 PCAL 的表现越好, 在另一些例子中, PCAL 的表现则对 β 的选取完全不敏感. 为了更深入的研究 PLAM 和 PCAL 的区别, 我们又进行了另一项数值实验, 如图 5.3 所示. 可以看到, 在同一个问题中, 采用不同的 β 选取, PCAL 的表现并不会随着 β 的变化而剧烈



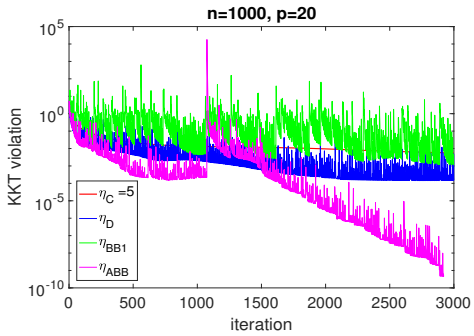
(a) 问题 1



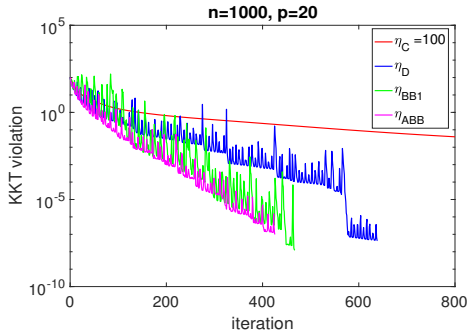
(b) 问题 2



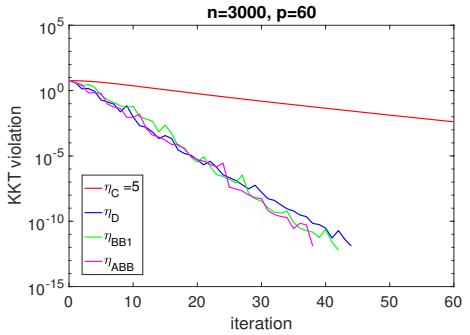
(c) 问题 3



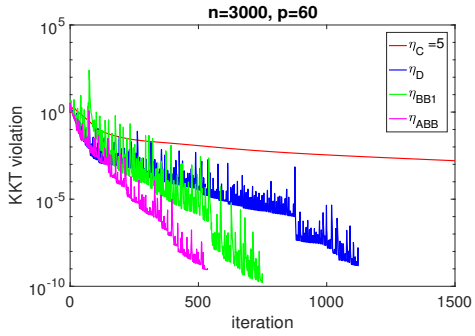
(d) 问题 4



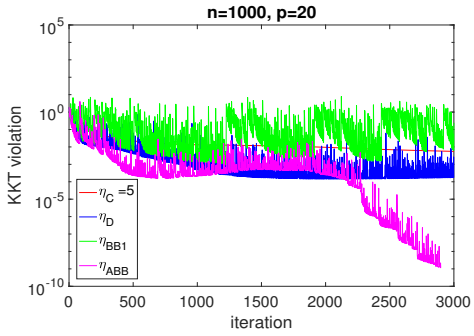
(e) 问题 1



(f) 问题 2



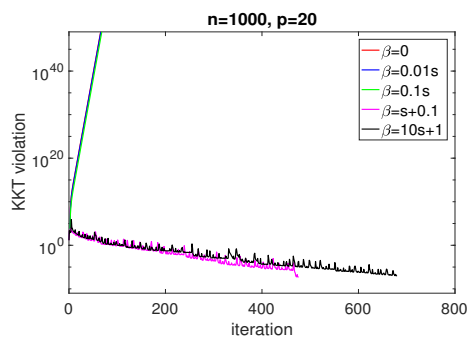
(g) 问题 3



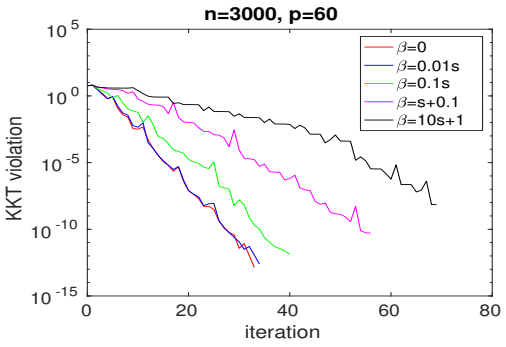
(h) 问题 4

图 5.1 不同邻近点参数 η 选取下 KKT 违反度的数值比较: PLAM (a)-(d), PCAL (e)-(h) ($\beta = s + 0.1$)

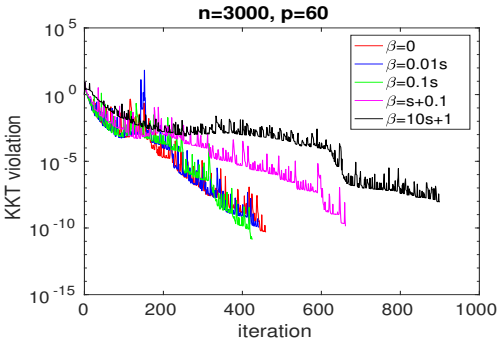
Figure 5.1 A comparison of KKT violation for PLAM (a)-(d) and PCAL (e)-(h) with different η ($\beta = s + 0.1$)



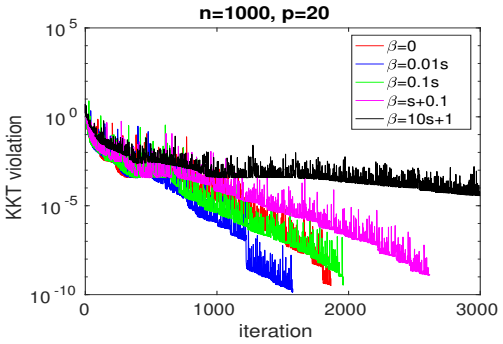
(a) 问题 1



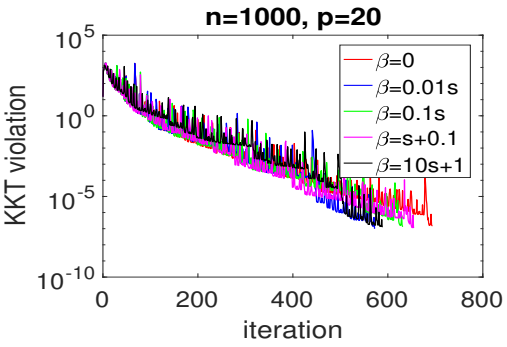
(b) 问题 2



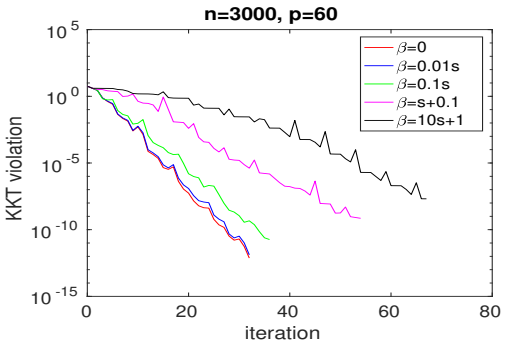
(c) 问题 3



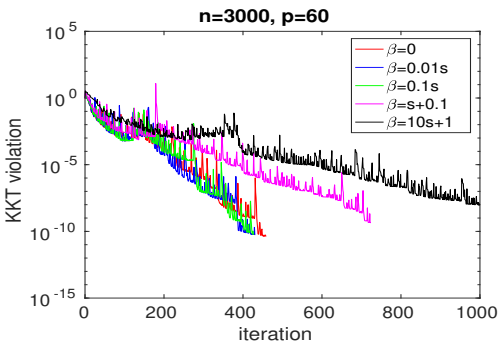
(d) 问题 4



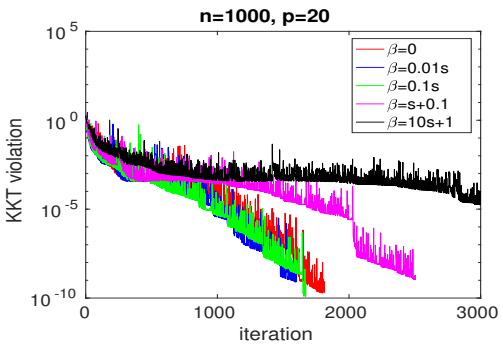
(e) 问题 1



(f) 问题 2



(g) 问题 3

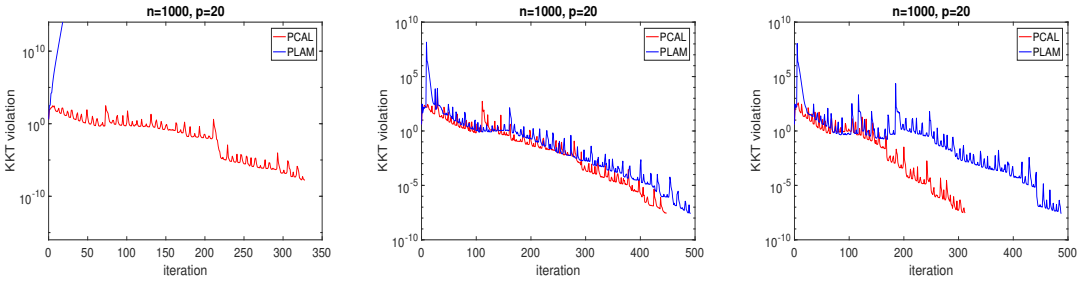


(h) 问题 4

图 5.2 不同罚参数 β 选取下 KKT 违反度的数值比较: PLAM (a)-(d), PCAL (e)-(h) ($\eta = \eta_{\mathrm{ABB}}$)

Figure 5.2 A comparison of KKT violation for PLAM (a)-(d) and PCAL (e)-(h) with different β ($\eta = \eta_{\mathrm{ABB}}$)

变化, 而 PLAM 却有明显的不同. 因此, 在实际中, 我们建议算法 PLAM 采用 s 的逼近值作为参数 β 的默认设定, 而 PCAL 采用常数值 1. 由于 PCAL 对于参数 β 不敏感, 故在并行实验环节, 我们采用 PCAL 作为我们的默认算法代表.



(a) $\beta = 0.01s$ (b) $\beta = s + 0.1$ (c) $\beta = 10s + 1$

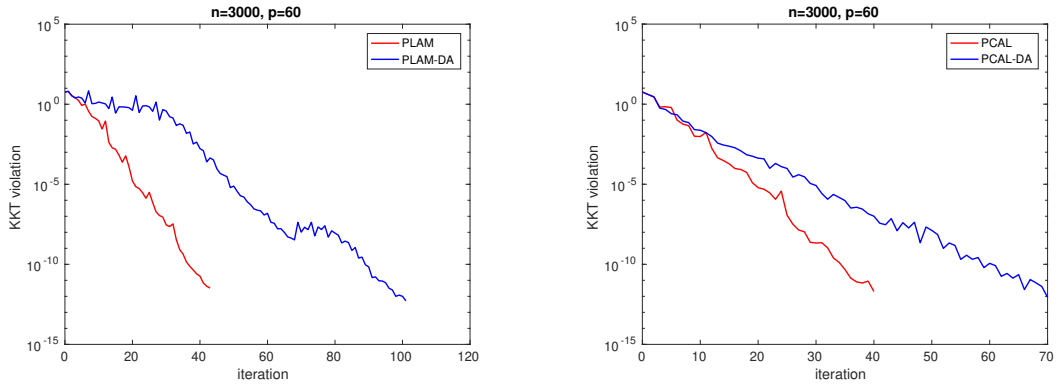
图 5.3 不同罚参数 β 选取的 PLAM 和 PCAL 的数值比较 (问题 1)

Figure 5.3 A comparison between PLAM and PCAL with different β on Problem 1

算法 PLAM 和经典的增广 Lagrange 函数法有两个重要的区别. 第一, 在原始变量的更新中, 我们采用一个简单的梯度步代替增广 Lagrange 子问题的求解. 第二, 对于对偶变量也就是 Lagrange 乘子的更新, 我们采用一个显式的表达式, 而不是增广 Lagrange 函数法中的对偶上升步. 为了验证我们提出的新乘子更新公式的有效性, 我们测试了不同乘子更新方式的算法 PLAM 和 PCAL. 我们分别用 PLAM-DA 和 PCAL-DA 表示第 3 步采用对偶上升步的算法 7 和 8. 数值实验结果呈现在图 5.4 中. 通过观察, 我们发现在求解正交约束优化问题时, 我们提出的新乘子显式更新公式的数值表现要优于传统的对偶上升步. 事实上, 这些结果与经典的增广 Lagrange 函数法的理论并不矛盾. 因为在这里, 我们并没有精确求解 Lagrange 子问题, 而是只进行了一步梯度下降. 除此之外, 对于罚参数 β 的更新, 我们并没有动态调节, 只是采用固定常数值. 在已有的测试中, 我们也尝试实现了经典的增广 Lagrange 函数法, 其中子问题求解到固定精度, 并且罚参数也动态调节. 值得说明的是, 我们的算法 PLAM 和 PCAL 的数值表现仍然优于经典的增广 Lagrange 函数法.

5.5.4 后处理过程

在本节的最后, 我们研究在迭代进行的过程中, KKT 和可行性的违反度的数值变化. 这里, 我们仅对问题 1 进行测试. 数值结果如图 5.5 所示. 我们注意到可行性违反度的下降是非单调的, 并且和 KKT 违反度的下降有相似的趋势. 这个结果恰好与引理 5.2 的理论分析结果一致. 在实际应用中, 如果我们想得到一个更



(a) 问题 1　　(b) 问题 2

图 5.4 不同乘子选取的 PLAM 和 PCAL 的数值比较

Figure 5.4 A comparion bewteen PLAM and PCAL on multilplier

高可行精度的解, 可以采取一个适当的 KKT 终止误差并增加一个后处理过程, 也就是对算法求得的解 X^* 进行正交化.

$$\mathrm{orth}(X^*) := \mathcal{P}_{\mathcal{S}_{n,p}}(X^*) = UV^\top, \tag{5.53}$$

其中 X^* 的奇异值分解为 $X^* = U\Sigma V^\top$, $U \in \mathcal{S}_{n,p}$, $\Sigma \in \mathbb{D}^p$ 且 $V \in \mathcal{S}_{p,p}$.

首先我们需要对上述正交化过程进行理论分析, 来保证其并不会对解的质量产生很大的影响, 特别当 δ 充分大时, KKT 违反度并不会随着后处理过程 (5.53) 而剧烈变化.

性质 5.1. *假设目标函数 $f(X)$ 满足 (5.29) 并且引理 5.2 中的假设都成立. 令 $\tilde{X} = \mathrm{orth}(X^*)$, 其中 orth 由 (5.53) 式定义, 则我们有*

$$||\nabla_X \mathcal{L}_\beta(\tilde{X}, \tilde{\Lambda})||_{\mathrm{F}} \le \left(1 + \frac{(2L + (N+\beta)||X^*||_2 + N)\,||X^*||_2}{\delta}\right) ||\nabla_X \mathcal{L}_\beta(X^*, \Lambda^*)||_{\mathrm{F}} \tag{5.54}$$

其中 $\Lambda^* = \Psi(\nabla f(X^*)^\top X^*)$ 且 $\tilde{\Lambda} = \Psi(\nabla f(\tilde{X})^\top \tilde{X})$.

证明. 利用 $||\Sigma - I||_{\mathrm{F}} \le ||(\Sigma - I)(\Sigma + I)||_{\mathrm{F}} = ||\Sigma^2 - I||_{\mathrm{F}}$, 我们得到

$$\begin{aligned}
& ||\nabla_X \mathcal{L}_\beta(X^*, \Lambda^*) - \nabla_X \mathcal{L}_\beta(\tilde{X}, \tilde{\Lambda})||_{\mathrm{F}} \\
\le\ & ||\nabla f(X) - \nabla f(\tilde{X})||_{\mathrm{F}} + ||X^* \nabla f(X^*)^\top X^* - X^* \nabla f(X^*)^\top \tilde{X}||_{\mathrm{F}} \\
& + ||X^* \nabla f(X^*)^\top \tilde{X} - \tilde{X} \nabla f(X^*)^\top \tilde{X}||_{\mathrm{F}} + ||\tilde{X} \nabla f(X^*)^\top \tilde{X} - \tilde{X} \nabla f(\tilde{X})^\top \tilde{X}||_{\mathrm{F}} \\
& + \beta ||X^* (X^{*\top} X^* - I_p)||_{\mathrm{F}}
\end{aligned}$$

$$
\begin{aligned}
&\leq L||X^* - \tilde{X}||_{\mathrm{F}} + N||X^*||_2||X^* - \tilde{X}||_{\mathrm{F}} + N||X^* - \tilde{X}||_{\mathrm{F}} \\
&\quad + L||X^* - \tilde{X}||_{\mathrm{F}} + \beta||X^*||_2||X^{*\top}X^* - I_p||_{\mathrm{F}} \\
&= (2L + N||X^*||_2 + N)||U\Sigma V^\top - UV^\top||_{\mathrm{F}} + \beta||X^*||_2||X^{*\top}X^* - I_p||_{\mathrm{F}} \\
&\leq (2L + (N+\beta)||X^*||_2 + N) \cdot ||X^{*\top}X^* - I_p||_{\mathrm{F}}.
\end{aligned}
$$

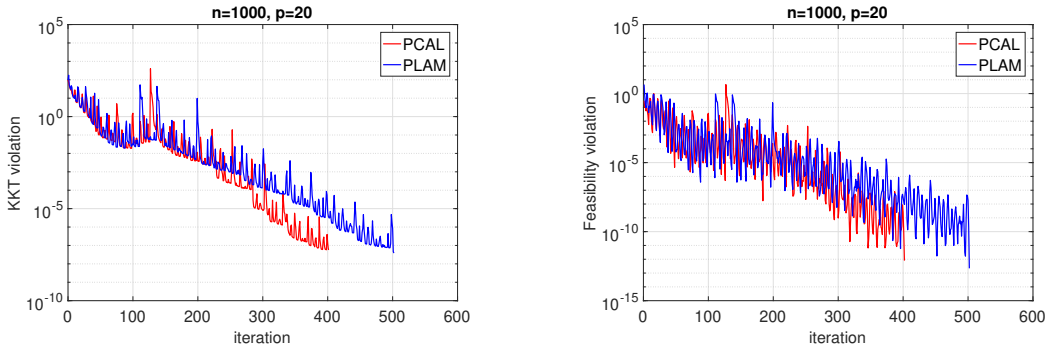
结合上述不等式和 (5.19), 我们推出

$$
\begin{aligned}
&||\nabla_X \mathcal{L}_\beta(\tilde{X}, \tilde{\Lambda})||_{\mathrm{F}} - ||\nabla_X \mathcal{L}_\beta(X^*, \Lambda^*)||_{\mathrm{F}} \leq ||\nabla_X \mathcal{L}_\beta(X^*, \Lambda^*) - \nabla_X \mathcal{L}_\beta(\tilde{X}, \tilde{\Lambda})||_{\mathrm{F}} \\
\leq\ & (2L + (N+\beta)||X^*||_2 + N) \cdot ||X^{*\top}X^* - I_p||_{\mathrm{F}} \\
\leq\ & \frac{(2L + (N+\beta)||X^*||_2 + N)\,||X^*||_2}{\delta} \cdot ||\nabla_X \mathcal{L}_\beta(X^*, \Lambda^*)||_{\mathrm{F}},
\end{aligned}
$$

进而不等式 (5.54) 成立. □

注 5.5. 假设 β 足够接近 $\left(||\nabla f(X^*)||_2 \cdot ||X^*||_2 + \delta\right)/\sigma^2_{\min}(X^*)$ 且 δ 充分大, 则不等式 (5.54) 的系数接近于 $\left(1 + \frac{||X^*||_2^2}{\sigma^2_{\min}(X^*)}\right)$. 当 X^* 近乎正交时, 系数进一步逼近 2.

另一方面, 我们也从数值角度验证了后处理过程的有效性, 即其在不影响 KKT 违反度的前提下提高解的可行性. 表 5.2 给出了数值实验结果. 其中列出了算法求解到终止点 X^* 的函数信息以及进行正交化后的函数信息. 以下的正交化过程都由 MATLAB 的内建函数 `svd`$(\cdot)$ 给出, 并且后处理过程将作为我们的默认设置.



(a) KKT 违反度　　(b) 可行性违反度

图 5.5 PLAM 和 PCAL 的 KKT 和可行性违反度的数值变化比较 (问题 1)

Figure 5.5 The results of KKT and feasibility violation for PLAM and PCAL on Problem 1

5.5.5 并行效率

在本小节, 我们研究算法 PLAM 和 PCAL 的并行效率. 为了得到算法的并行可扩展性, 我们首先需要在较少的 CPU 核数下进行大规模问题的测试, 这将会花

算法		函数值	KKT 违反度	可行性违反度
		$n = 1000, p = 20, \alpha = 1$		
PLAM	X^*	-4.205530767124e+02	8.74e-06	2.56e-09
	orth(X^*)	-4.205530767662e+02	8.74e-06	5.61e-15
PCAL	X^*	-4.205530767773e+02	6.01e-06	1.13e-08
	orth(X^*)	-4.205530767665e+02	6.00e-06	2.00e-14

表 5.2 算法 PLAM 和 PCAL 后处理过程的数值比较 (问题 1)

Table 5.2 The results of orthogonal step for PLAM and PCAL on Problem 1

费大量时间. 因此, 为了避免一些无意义重复的测试和比较, 我们只测试两类代表算法 PCAL 和 MOptQR 的数值表现. 其中 QR 收缩类算法 MOptQR 的参数选取与第 3 章相同.

所有的算法都是由 C++ 编程语言实现并使用 OpenM 进行并行计算. 我们在实验中使用的线性代数库是 Eigen[2] (版本 3.3.4). Eigen 是一个开源的并且被广泛使用的矩阵计算 C++ 模板库.

BLAS3, 也就是矩阵和矩阵的乘法运算, 在算法 PCAL 和 MOptQR 的总计算量中占有很大的比重. 因此, 一种高效的并行策略对于 BLAS3 的计算至关重要, 如果采用更好的并行策略将会节省更多的程序运行时间. 我们首先通过一些测试来选取哪种并行策略更适合我们的实际计算. 我们考虑两种不同的选择. 第一种是模板库 Eigen 默认自带的多线程并行运算[3], 它利用 OpenMP 对稠密的矩阵乘法和行存储模式的稀疏向量/矩阵乘法进行并行. 第二种策略是直接以列乘积的方式对 BLAS3 计算进行并行. 也就是说, 当我们计算两个矩阵的乘积 AB 时, 用并行的方式分别在不同的核中独立计算矩阵 A 与矩阵 B 的每一列的乘积. 具体方式如图 5.6 所示.

接下来我们测试两种不同策略的并行效率. 首先我们生成矩阵 A 和 B,

```
A=Random(1000,10000),      B=Random(10000,1000),
```

其中 “`Random(·,·)`” 是由 Eigen 内部提供的矩阵生成函数. 我们分别在 1, 2, 4, 8, 16, 32, 64 和 96 核下运行程序, 矩阵矩阵乘法 AB 的数值结果如图 5.7 所示. 其中 “Eigen” 和 “Column-wise” 分别代表默认的并行策略和列乘积的并行策略. 我们观

[2]参考: http://eigen.tuxfamily.org/index.php?title=Main_Page.

[3]更多信息请参见 http://eigen.tuxfamily.org/dox/TopicMultiThreading.html.

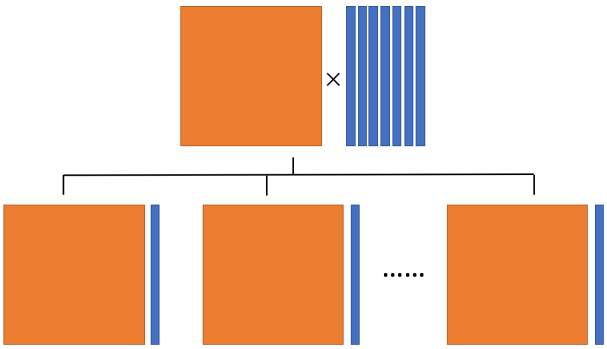
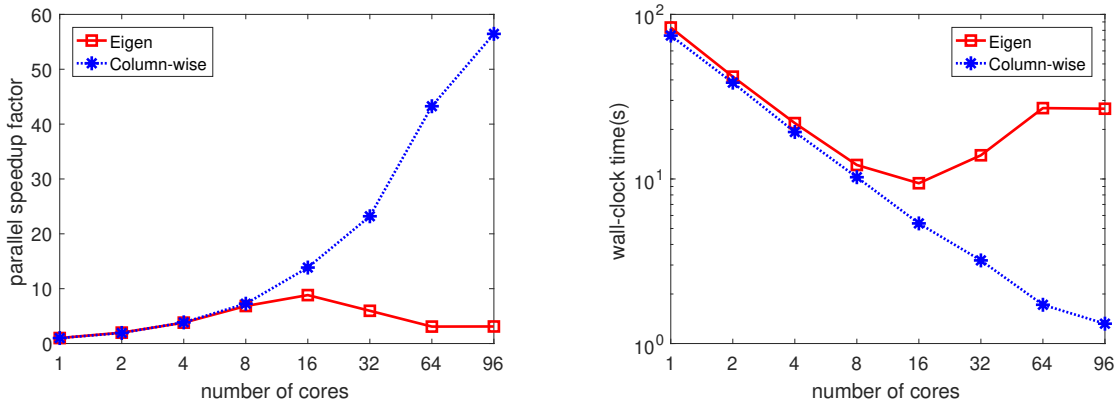


图 5.6 按列相乘的并行策略

Figure 5.6 Parallel strategy for matrix-matrix multiplication

察到列乘积的并行策略明显优于 Eigen 默认的多线程计算. 因此, 在接下来的实验中对于 BLAS3 计算我们采用列乘积形式的并行策略.



(a) 加速比　　(b) 时间

图 5.7 稠密 BLAS3 计算比较: $A^{1000\times10000}B^{10000\times1000}$

Figure 5.7 The results of dense-dense BLAS3: $A^{1000\times10000}B^{10000\times1000}$

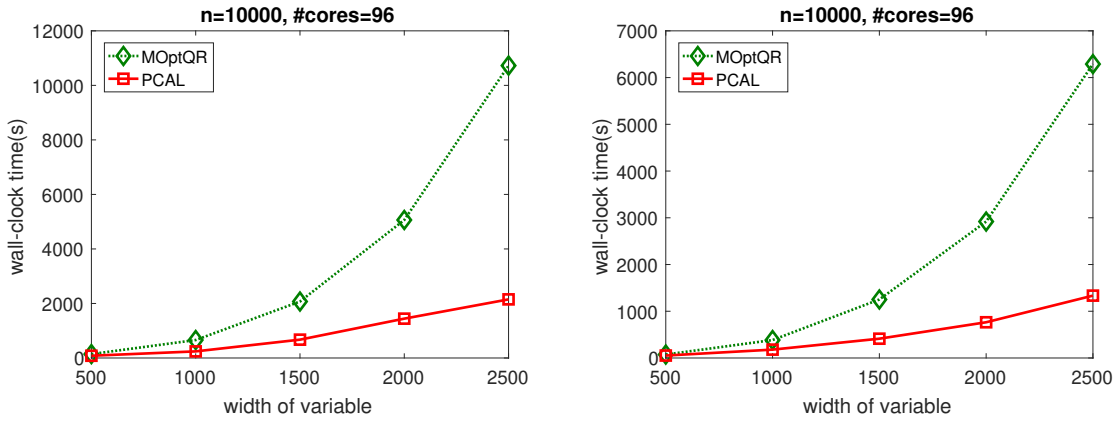
我们测试了 MOptQR 和新算法 PCAL 的并行效率. 由于 MOptQR 在每一步迭代中都需要进行 QR 分解, 因此矩阵分解算法的选取对其数值表现至关重要. 根据 Eigen 已有的数值报告[4], 我们选取 Eigen 中的 "LLT" 类作为 QR 分解的计算方式. 其正交化过程的计算包含一个小规模 (p 阶) 的 Cholesky 分解和一个 p 阶线性方程组的求解. 测试中所有的算法参数都选取其默认值. 初始点 X^0 由 $X^0 =$`random(n,p)` 和 $X^0 =$`qr`(X^0) 生成.

我们首先测试问题 1 和 2. 在问题 1 中, 我们令 L 是一个块对角矩阵, 具体来讲, $L = \mathrm{Diag}(L_1, \ldots, L_s)$, 其中 $L_i \in \mathbb{R}^{5\times5}$ 都是以 2 为主对角元以 -1 为次对角元的

[4]请参见 http://eigen.tuxfamily.org/dox/group__TutorialLinearAlgebra.html.

三对角矩阵. 我们令系数 $\alpha = 1$. 在问题 2 中, 我们令 A 是以 2 为主对角元以 -1 为次对角元的三对角矩阵, 令 G=Random(n,p). 这样的问题生成是为了使函数值和梯度的计算可以并行处理. 除此之外, 也使得函数的计算变得稀疏和结构化, 由此得到的并行加速比并不会太依赖于函数值和梯度的计算.

在第一组测试中, 我们研究在多核环境下算法 MOptQR 和 PCAL 随着变量列数增加的数值表现. 这里, 我们选取 $n = 10000$ 且 p 在 $500, 1000, 1500, 2000, 2500$ 中变化. 所有的算法都是在 96 核的环境下运行的. 总的运行时间如图 5.8 所示, 其中 “#cores” 表示计算所使用的 CPU 核数. 从图中我们发现, 随着变量列数的线性增加, PCAL 所需的计算时间并不会剧烈增加, 而 MOptQR 则以非线性的趋势变化.



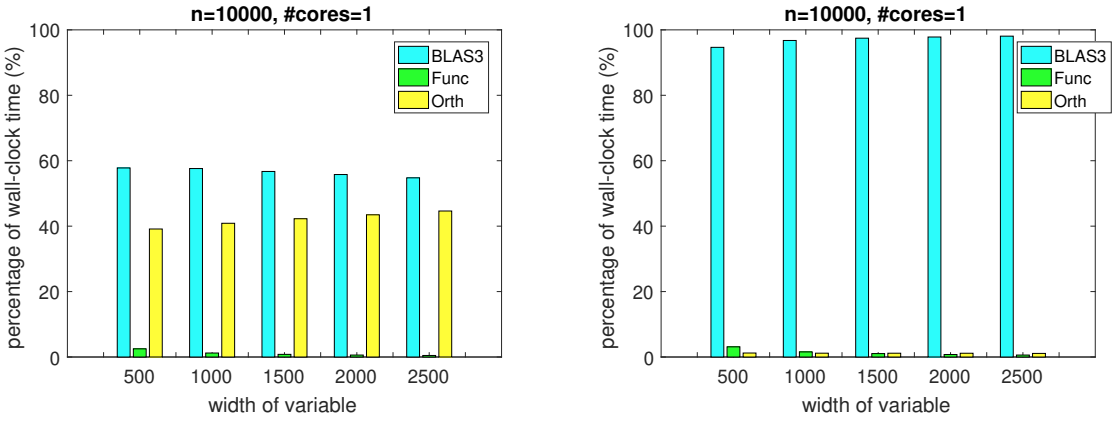
(a) 问题 1　　(b) 问题 2

图 5.8 变量列数变化下的运行总时间比较

Figure 5.8 The wall-clock time results on varying width of the matrix variable

在图 5.9 中, 我们测试了在单核环境下算法不同种类计算时间所占总时间的百分比. 具体分为三大类: “BLAS3” (稠密矩阵乘法), “Func” (函数值与梯度计算), “Orth” (正交化过程: MOptQR 中的 QR 分解和 PCAL 的后处理). 这三类计算几乎占用了算法运行的全部时间. 需要阐明的, 我们只是在算法的层级内进行了相关计算的时间统计, 例如任何函数值和梯度的计算并不包含在 “BLAS3” 类别中, 尽管其中有一些计算本质上属于 “BLAS3”. 其次, 这样一种分类的正确与否值得讨论, 但这并不影响我们对于算法的整体印象, 因为从实际结果来看, “BLAS3” 类的计算要远远比 “Orth” 类的计算更适合并行. 从图 5.9 中我们可以看到, 对于 PCAL 而言, “BLAS3” 类的运行时间几乎主导了 PCAL 的计算. 随着列数 p 的增加, “BLAS3” 类的计算占比越来越大, 而 “Func” 类的计算越来越小, 并且 “Orth”

的运行时间几乎可以忽略. 然而对于 MOptQR, “BLAS3” 类的计算时间大约占总时间的 60%, 并且正交化计算 “Orth” 的运行时间占比随着 p 的增加逐渐增大.



(a) MOptQR (b) PCAL

图 5.9 单核环境下不同种类的计算占比 (问题 2)

Figure 5.9 A comparison of timing profile on a single core for Problem 2

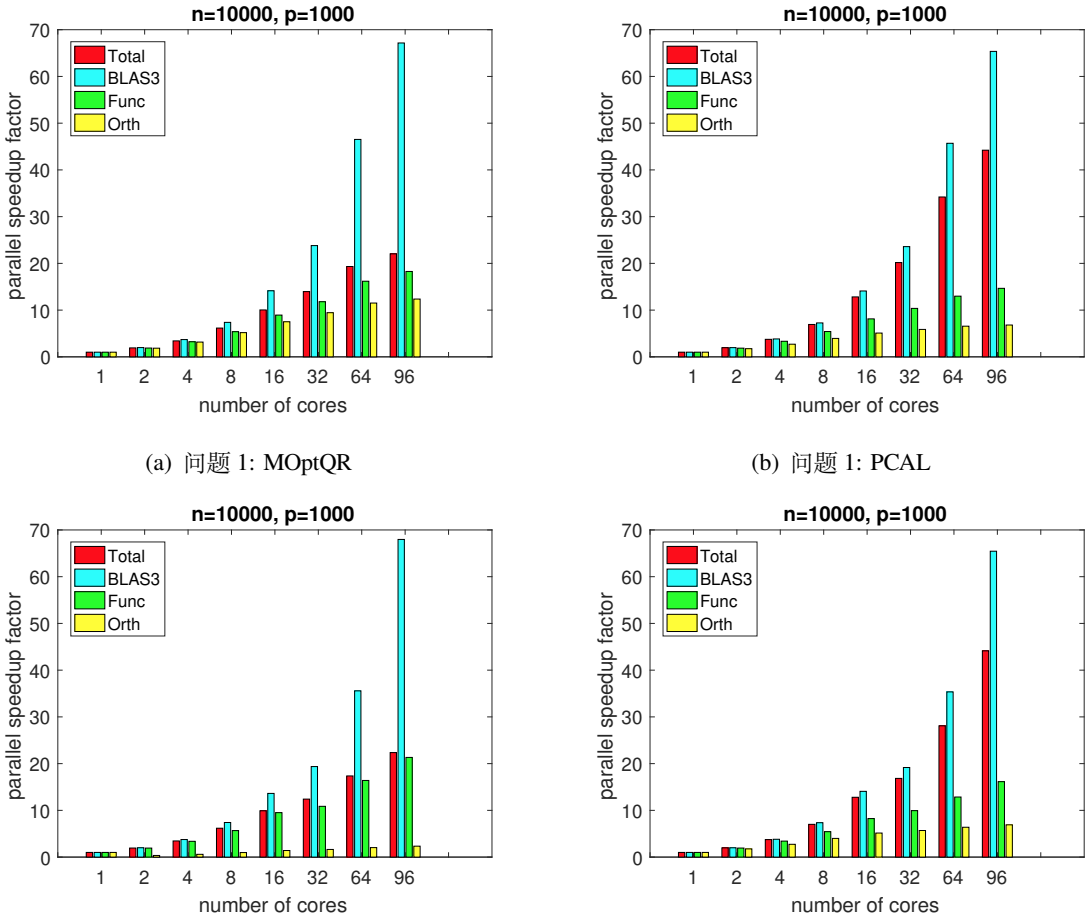
接下来, 我们令 $n = 10000$ 且 $p = 1000, 2000$, 测试算法 PCAL 和 MOptQR 在不同核数下的并行加速比, 其中核数分别取为 1, 2, 4, 8, 16, 32, 64 和 96. 图 5.10 和 5.11 展示了总运行时间, “BLAS3” 类, “Func” 类和 “Orth” 类的并行加速比. 从图中我们可以看出, BLAS3 类的计算具有很高的可扩展性, 结合上一个实验, 我们得出 PCAL 的可扩展性要远远优于 MOptQR. 进一步, 随着问题规模的增大, 例如当矩阵变量的列数增加时, PCAL 在并行效率方面的优势会越来越明显.

5.5.6 PCAL 与 ADMM 的比较

在第 3 章中, 我们详细介绍了求解正交约束优化问题的不可行方法. 其中, 交替方向乘子法 (ADMM) 在实际应用中有较好的数值表现. 因此, 在本节的最后, 我们在串行环境下进行 PCAL 与 ADMM 类算法的数值比较.

文献 [117] 提出了一类分离正交约束算法 (SOC), 其本质可看成是 ADMM 类型的算法. 在这一小节, 我们实现了 SOC 算法, 并在测试问题 1-4 上进行了数值比较. 问题的参数设定与 5.5.3 小节相同. 对于算法 PCAL, 我们采取其默认设置. 在 SOC 算法中, 参数 r 对于其数值表现极其敏感. 因此, 通过多次试验, 我们选取针对不同问题表现最好的参数, 即 $r = 90, 1, 5, 5$. 算法 SOC 的子问题求解精度设为 10^{-8}.

图 5.12 和 5.13 展示了 PCAL 和 SOC 的 KKT 和可行性违反度的数值变化. 图 5.14 显示了 PCAL 和 SOC 每步所需的内迭代数. 从图中我们发现, PCAL 在测试



(a) 问题 1: MOptQR

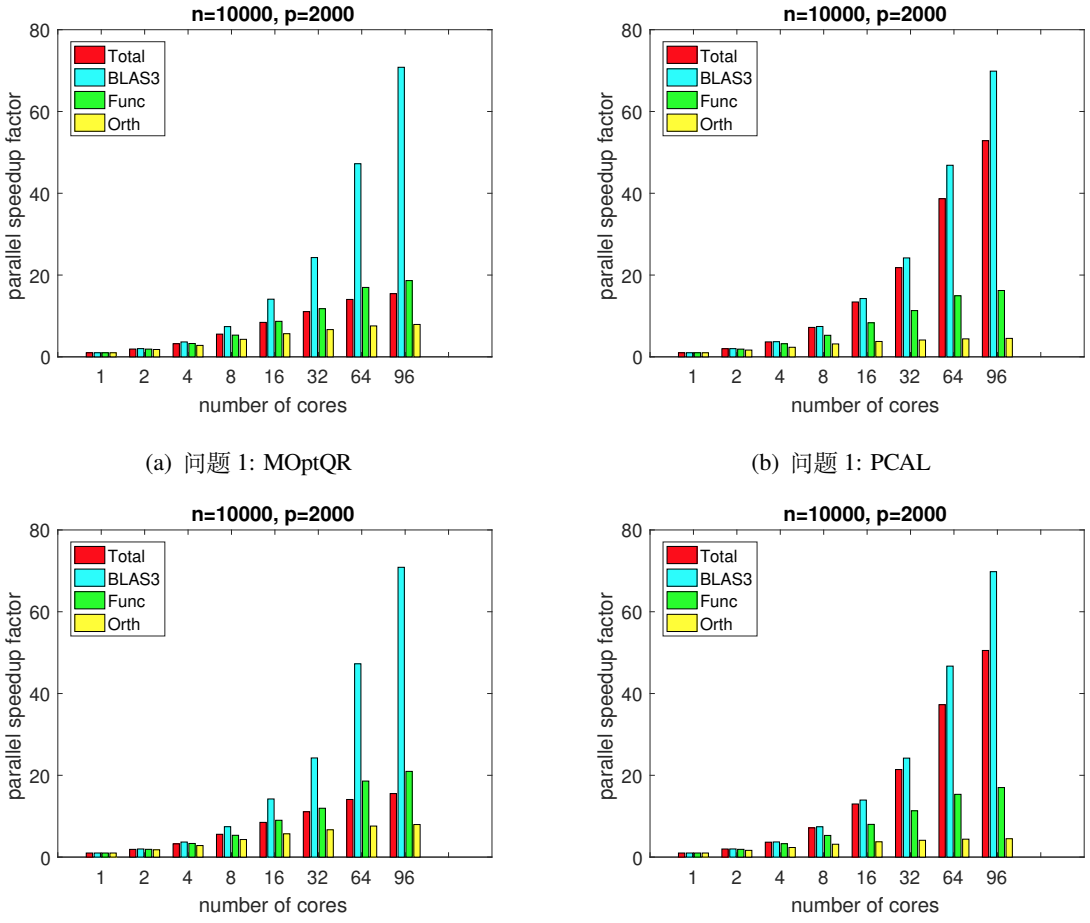
(b) 问题 1: PCAL

(c) 问题 2: MOptQR

(d) 问题 2: PCAL

图 5.10 MOptQR 和 PCAL 的并行加速比比较 ($p = 1000$)

Figure 5.10 A comparison of speedup factor among MOptQR and PCAL ($p = 1000$)



(a) 问题 1: MOptQR

(b) 问题 1: PCAL

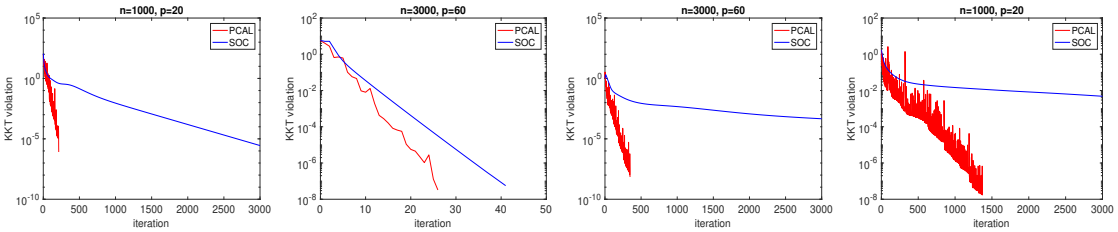
(c) 问题 2: MOptQR

(d) 问题 2: PCAL

图 5.11 MOptQR 和 PCAL 的并行加速比比较 ($p = 2000$)

Figure 5.11 A comparison of speedup factor among MOptQR and PCAL ($p = 2000$)

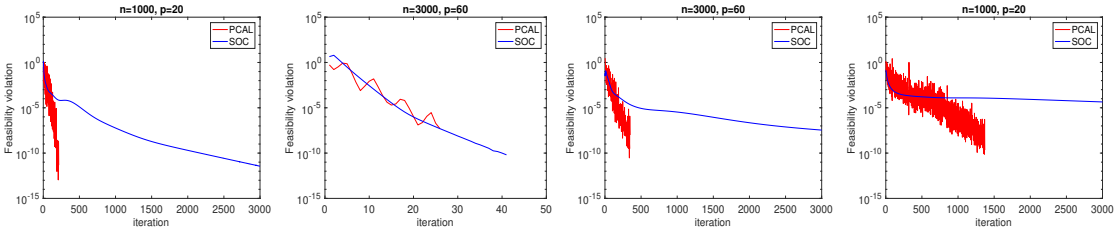
问题 1-4 上表现优于 SOC, 并且 PCAL 每步迭代只需要很少的计算代价, 而 SOC 则需要每步做一次正交化. 综上, 我们的新算法 PCAL 数值表现优于 ADMM 类型的算法.



(a) 问题 1　(b) 问题 2　(c) 问题 3　(d) 问题 4

图 5.12 PCAL 和 SOC 的 KKT 违反度的数值变化比较

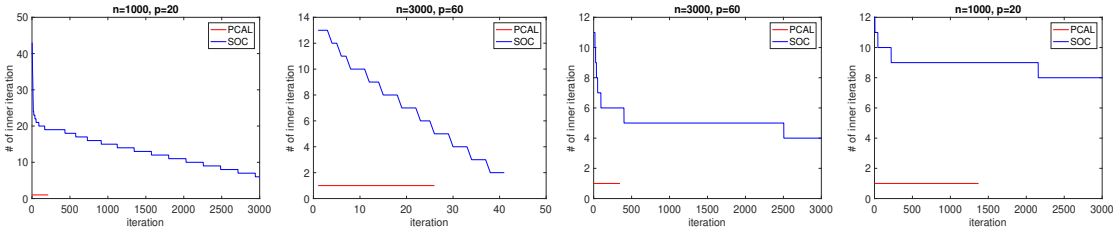
Figure 5.12 Comparison on KKT violation of PCAL and SOC



(a) 问题 1　(b) 问题 2　(c) 问题 3　(d) 问题 4

图 5.13 PCAL 和 SOC 的可行性违反度的数值变化比较

Figure 5.13 Comparison on feasibility violation of PCAL and SOC



(a) 问题 1　(b) 问题 2　(c) 问题 3　(d) 问题 4

图 5.14 PCAL 和 SOC 的内迭代数变化比较

Figure 5.14 Comparison on inner iteration of PCAL and SOC

5.6 小结

正交约束优化已有的算法大部分都基于流形收缩类算法, 由于每一步都需要保持迭代点可行, 这些方法在矩阵变量列数 p 相对较小时表现优异. 然而, 在实际计算中, 随着列数 p 的增加, 正交化过程的低可扩展性成为了已有算法的主要瓶

颈. 为了解决这个问题, 我们考虑使用不可行方法. 但是, 以往经典的不可行方法, 例如增广 Lagrange 函数法, 在效率远不如已有的收缩类可行方法. 尽管我们利用并行化的策略可以减少增广 Lagrange 方法的运行时间, 但是总体上而言, 其效率还是不如已有的算法. 因此, 在本章中针对正交约束优化问题, 我们提出了一个更实际可行更高效的可并行算法. 通过利用 Lagrange 乘子在任意一阶稳定点处的显式表达式, 我们给出了新的改进后的增广 Lagrange 算法. 在每一步迭代中, 子问题的求解我们只需要一个增广 Lagrange 梯度步, 对于对偶变量, 我们采取了新的显式更新方式. 进一步, 考虑到迭代点列的有界性, 我们还提出了一个改进的算法版本. 在实际计算中, 我们并行实现了新算法, 在并行环境下, 我们算法的可扩展性要远远优于已有的收缩类可行方法. 由此可见, 我们的算法在实际应用中具有巨大的潜力.

第 6 章　正交约束优化在电子结构计算中的应用

在电子结构计算中, Kohn-Sham 密度泛函理论 (KSDFT) 是一类重要的研究方法, 已经被广泛应用于凝聚态物理、生物大分子及纳米材料等体系的模拟. KSDFT 的模型通常表述为一个带有正交约束的优化问题或者被称为 Kohn-Sham 方程的非线性特征值问题. 通常, 这类问题的变量规模非常巨大, 并且正交化过程可扩展性差, 因此对于正交约束优化问题的高效求解器的开发变得至关重要. 在本章中, 我们分别应用乘子校正算法和基于增广 Lagrange 的并行算法求解 Kohn-Sham 总能量极小化问题. 在串行环境下, 我们测试了 18 个不同的分子结构, 数值实验显示了我们算法的数值表现优于已有的经典算法. 在并行环境下, 我们测试了一个简化的 Kohn-Sham 总能量极小化问题, 数值结果显示我们提出的并行算法 PCAL 具有较高的可扩展性.

6.1　引言

物质的电子结构 (即电荷分布) 本质上决定了物质的力学、光学、磁学等性质. 第一原理电子结构计算模拟以量子力学为理论基础, 可以研究那些理论难以分析的规律和实验不容易观测到的现象. 因此, 电子结构计算方法对于我们理解世界有着重要的意义.

不同于经典力学, 在量子力学中, 粒子所处的状态由 “波函数” 来描述, 而波函数模的平方则表示了粒子在空间中的概率密度. 在非相对论情况下, 多电子体系的波函数满足 Schrödinger 方程. 因此, 电子结构计算的主要目标是求解 Schrödinger 方程从而得到电子波函数. 我们考虑包含 N 个电子的体系, 此时 Schrödinger 方程是 $3N$ 维的特征值问题, 数值上来讲, 属于不可计算模型 [138]. 因此, 研究者们考虑了近似简化模型, 其中最为广泛使用的是上世纪 60 年代发展起来的密度泛函理论.

密度泛函理论 (DFT) 的基本思想是用 3 维的电子密度代替 $3N$ 维的电子波函数, 从而使得问题的规模显著降低. 1965 年, Kohn 和 Sham [105] 考虑将原有的多体 Schrödinger 方程转化为有效的单体问题, 并通过交换关联能表示电子之间的交换和关联作用, 得到了一个易于计算的模型. Kohn-Sham 密度泛函理论的模型通常表述为一个带有正交约束的优化问题或者被称为 Kohn-Sham 方程的非线性特征值问题. 尽管交换关联能的具体表达式未知, 但在实际计算中, 基于 KSDFT 的

数值方法表现优异. 因此, KSDFT 已成为电子结构计算中非常重要的研究方向.

6.2 Kohn-Sham 密度泛函理论

6.2.1 Kohn-Sham 总能量极小化

首先, 我们定义 Kohn-Sham 总能量泛函 [139]

$$\begin{aligned} E_{\text{total}}^{\text{KS}}(\psi_1,\ldots,\psi_p) \;:=\; & \frac{1}{2}\sum_{i=1}^{p}\int_{\Omega}\left\|\nabla\psi_i(r)\right\|^2 \mathrm{d}r + \int_{\Omega}\rho(r)V_{\text{ion}}(r)\mathrm{d}r + \\ & \frac{1}{2}\int_{\Omega}\int_{\Omega}\frac{\rho(r)\rho(r')}{\|r-r'\|}\mathrm{d}r\mathrm{d}r' + E_{\text{xc}}(\rho), \end{aligned} \tag{6.1}$$

其中 $\Omega \in \mathbb{R}^3$, $\psi_i (i = 1,\ldots,p)$ 是单粒子的波函数.

$$\rho(r) = \sum_{i=1}^{p}\psi_i^*(r)\psi_i(r)$$

表示电子密度. 函数 $V_{\text{ion}}(r) = \sum_{j=1}^{n_u} zj/\left\|r-\hat{r}_j\right\|$ 表示原子核引入的离子势能, 其中 n_u 为原子核数. $E_{\text{xc}}(\rho)$ 表示交换关联能函数.

Kohn-Sham 总能量极小化问题是指在波函数满足正交性的约束下极小化总能量泛函, 即

$$\begin{aligned} \min_{\psi_1,\ldots,\psi_p} \quad & E_{\text{total}}^{\text{KS}}(\psi_1,\ldots,\psi_p) \\ \text{s.t.} \quad & \psi_i^*\psi_j = \delta_{ij}, \end{aligned} \tag{6.2}$$

其中 $\delta_{ij} = \begin{cases} 1, & \text{若 } i = j; \\ 0, & \text{否则.} \end{cases}$

为了数值求解上述泛函极小化问题, 并且考虑到问题规模和计算复杂度, 研究者们发展了多种高效的离散方法. 其中, 主要的离散方法包含三类: 平面波方法、局部基集法和实空间方法. 平面波方法也被称为倒空间方法, 使用 Fourier 基底离散. 在本章中, 我们主要考虑平面波方法.

6.2.2 离散问题

平面波离散后的 Kohn-Sham 总能量 (6.1) 有如下表示,

$$E(X) := \frac{1}{4}\text{tr}(X^\top LX) + \frac{1}{2}\text{tr}(X^\top V_{\text{ion}}X) + \frac{1}{4}\rho^\top L^\dagger\rho + \frac{1}{2}\rho^\top\epsilon_{\text{xc}}(\rho), \tag{6.3}$$

其中 $\rho(X) := \text{diag}(XX^\top)$ 表示电子密度, L 是平面波基底下拉普拉斯算子的有限维表示. 离散的局部离子势由对角矩阵 V_{ion} 表示. 矩阵 $L^\dagger$ 表示 Hartree 势离散形

式 L 的伪逆. 交换关联能函数 ϵ_{xc} 用来刻画电子间的非经典和量子关系. 由此, 我们可以得到离散的 Kohn-Sham 总能量极小化问题,

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times p}} \quad & E(X) \\ \text{s.t.} \quad & X^\top X = I_p. \end{aligned} \tag{6.4}$$

我们发现, 离散的总能量极小化问题恰好是正交约束优化问题 (2.1) 的一个特例. 最近几十年间, 研究者们针对离散的 Kohn-Sham 总能量极小化问题已经提出了一些有效的优化方法 [23, 57, 70–72, 127, 139–143]. 尽管如此, 考虑到问题规模以及正交约束的低可扩展性, 高效的可并行的算法仍然值得研究.

不难验证 $\nabla E(X) = H(X)X$, 其中

$$H(X) = L/2 + V_{ion} + \text{Diag}(L^\dagger \varrho(X)) + \text{Diag}(\mu_{\text{xc}}(\varrho(X)))$$

是 Kohn-Sham 哈密尔顿量且 $\mu_{\text{xc}}(\varrho(X)) = d\epsilon_{\text{xc}}/d\varrho(X)$, $H(X)$ 满足正交不变性和对称性. 我们发现总能量函数 $E(X)$ 满足第 3 章中问题的假设 3.1. 特别地, $E(X)$ 并没有假设 3.1 中 X 的线性项. 因此, 根据第 2 章中正交约束优化问题最优性条件的讨论, 我们可以很容易得到问题 (6.4) 的一阶最优性条件,

$$\begin{cases} H(X)X &= X\Lambda; \\ X^\top X &= I, \end{cases} \tag{6.5}$$

其中 $\Lambda \in \mathbb{SR}^{p\times p}$ 表示 Lagrange 乘子. 事实上, 上述非线性特征值问题也被称为 Kohn-Sham 方程.

注 6.1. *由一阶最优性条件 (6.5) 和 $H(X)$ 的定义, 我们得到对于任意的 $X \in \mathcal{S}_{n,p}$, Lagrange 乘子 $\Lambda = X^\top H(X)X$ 满足对称性. 因此, 针对问题 (6.4), 我们在第 3 章中提到的乘子校正算法无需进行乘子校正步.*

电子结构计算领域最广泛使用的算法是自洽场迭代 (SCF), SCF 本质上是将一个非线性特征值问题转化为一系列线性特征值问题, 每一步相当于在正交约束下极小化一个原始能量泛函的二次替代函数 [73]. 具体来讲, 给定当前迭代点 X^k, 求解如下的线性特征值问题,

$$\begin{cases} H(X^k)X &= X\Lambda; \\ X^\top X &= I. \end{cases} \tag{6.6}$$

令其解为下一步迭代点 X^{k+1}. 其中 Λ 包含 $H(X^k)$ 的 p 个最小特征值.

接下来, 我们应用本文提出的新算法求解 Kohn-Sham 总能量极小化问题, 并与已有的算法进行比较, 其中包括 SCF.

6.3 数值实验

6.3.1 测试平台及算法

串行环境下, 我们的测试基于 MATLAB 工具包 KSSOLV[1] [72]. KSSOLV 是专门针对电子结构计算领域的计算平台, 它允许研究者可以较为容易验证自己算法的有效性, 而不需要过多的关注电子结构计算的建模以及内部计算. KSSOLV 中交换关联能函数 ϵ_{xc} 采用文献 [144] 提出的被广泛接受的局域密度近似 (LDA) 显式形式.

KSSOLV 内部提供了几种算法可供用户自由选取和比较, 其中包括自洽场迭代 (SCF) 和信赖域直接优化算法 (TRDCM) [71]. SCF 及其变种是目前 Kohn-Sham 密度泛函理论中最广泛使用的一类方法. 另一类方法 TRDCM 基于传统的信赖域优化方法, 并结合了自洽场迭代用于求解信赖域子问题. 除了 SCF 和 TRDCM 之外, 我们还选取了两种目前最有效的针对一般正交约束优化问题的算法. 第一种是由文献 [99] 提出的保正交约束可行方法 OptM[2]. 第二种用于比较的算法是 QR 收缩算法 [33]. 算法的原始版本是 MOptQR-LS (带有线搜索的流形 QR 收缩算法[3]). 为了公平起见, 我们也同样实现了交替 BB 步长版本的 MOptQR-LS, 记做 MOptQR. 在每一步迭代中, MOptQR 都需要进行一个 QR 分解. 关于上述算法的详细介绍, 请参见 2.3.1 小节. 在本文提出的算法中, 我们选取了乘子校正算法 GR-BB 以及不可行方法 PLAM 和 PCAL.

对于以上所有的方法, 我们设定同样的停机准则 $\left\|(I_n - XX^\top)\nabla E(X)\right\|_{\text{F}} < 10^{-5}$. 并且对于 SCF 和 TRDCM, 最大迭代数设为 MaxIter $=$ 200 , 而对于 MOptQR, OptM, GR-BB, PLAM 和 PCAL 设为 1000. PLAM 的罚参数 β_{PLAM} 选取其具有最好数值表现的值, 而 PCAL 的罚参数 β_{PCAL} 取为常数 1. 其他参数设定与第 3 和第 5 章相同. 对于所有测试算法, 我们通过 KSSOLV 的内建函数 "`getX0`" 选取相同的初始点 X^0.

并行环境下, 我们测试了流形收缩类算法 MOptQR 和本文提出的可并行算法 PCAL. 算法参数和测试平台与 5.5.5 小节相同.

6.3.2 测试问题

串行环境下, 我们选取由工具包 KSSOLV 提供的关于不同分子的 18 个算例. 具体请见表 6.1

[1] http://crd-legacy.lbl.gov/~chao/KSSOLV/

[2] http://optman.blogs.rice.edu

[3] http://www.manopt.org

问题	$n \times p$	问题	$n \times p$	问题	$n \times p$
al	16879×12	ctube661	12599×48	nic	251×7
alanine	12671×18	glutamine	16517×29	pentacene	44791×51
benzene	8407×15	graphene16	3071×37	ptnio	4609×43
c2h6	2103×7	graphene30	12279×67	qdot	2103×8
c12h26	5709×37	h2o	2103×4	si2h4	2103×6
co2	2103×8	hnco	2103×8	sih4	2103×4

表 6.1 KSSOLV 测试问题集

Table 6.1 Testing problems in KSSOLV

并行环境下, 我们测试了如下的离散 Kohn-Sham 总能量极小化问题

$$\begin{aligned} \min_{X \in \mathbb{R}^{n \times p}} \quad & \tfrac{1}{2}\mathrm{tr}(X^\top L X) + \tfrac{1}{2}\rho(X)^\top L^\dagger \rho(X) - \tfrac{3}{4}\gamma \rho(X)^\top \rho(X)^{\frac{1}{3}} \\ \text{s. t.} \quad & X^\top X = I_p, \end{aligned} \tag{6.7}$$

其中矩阵 $L \in \mathbb{R}^{n \times n}$ 且 $\rho(X) := \mathrm{diag}(XX^\top)$. 参数 $\gamma = 2(\frac{3}{\pi})^{1/3}$, 并且 $\rho(X)^{\frac{1}{3}}$ 表示向量 $\rho(X)$ 对应元素的三次方根构成的向量. 这个问题采用了一种特殊的交换函数 $-\frac{3}{4}\gamma\rho(X)^\top\rho(X)^{\frac{1}{3}}$ (其中关联项被忽略), 具体可参考文献 [74]. L 是一个块对角矩阵, 具体来讲, 就是 $L = \mathrm{Diag}(L_1, \ldots, L_s)$, 其中 $L_i \in \mathbb{R}^{5\times 5}$ 都是以 2 为主对角元以 -1 为次对角元的三对角矩阵.

6.3.3 乘子校正算法的数值结果

在本小节, 我们比较第 3 章提出的算法 GR-BB 和已有算法的数值表现. 值得说明的是, 我们的算法 GR-BB 不需要进行乘子校正步. 本小节的实验平台于第 3 章相同.

具体的数值实验结果如表 6.2, 6.3 和 6.4 所示. 在表中, "E_{tot}", "KKT 违反度" 分别表示总能量函数值以及 $\left\|(I_n - XX^\top)H(X)X\right\|_{\mathrm{F}}$ 的值. 通过观察, 我们发现 GR-BB 的表现优于其他算法, 在大多数的例子中, GR-BB 总能得到相同的总能量函数值和较低的 KKT 违反度. 特别地, 在较大规模的问题 "ctube661" 中, GR-BB 在达到相似总能量函数值和同量级 KKT 违反度的情况下, 需要更少的 CPU 时间.

6.3.4 PLAM 和 PCAL 的数值结果

在本小节, 我们将第 5 章提出的算法 PLAM 和 PCAL 应用于求解 Kohn-Sham 总能量极小化问题 (6.4). 此实验的目的是测试我们提出的不可行方法和已有的没有考虑并行计算的可行方法之间的数值表现. 相应的测试环境与第 5 章相同.

算法	E_{tot}	KKT 违反度	总迭代数	CPU 时间 (s)
al, $n = 16879$, $p = 12$				
SCF	-1.5799906179e+01	8.68e-03	200	2509.48
TRDCM	-1.5803817595e+01	8.15e-06	184	1595.83
MOptQR	-1.5802118775e+01	8.42e-03	1000	2017.61
GR-BB	-1.5802922328e+01	2.05e-03	1000	2070.80
alanine, $n = 12671$, $p = 18$				
SCF	-6.1161921213e+01	9.70e-07	15	204.20
TRDCM	-6.1161921213e+01	5.91e-06	16	147.84
MOptQR	-6.1161921213e+01	8.14e-06	65	142.70
GR-BB	-6.1161921212e+01	9.78e-06	63	142.36
benzene, $n = 8407$, $p = 15$				
SCF	-3.7225751363e+01	7.85e-07	12	85.52
TRDCM	-3.7225751363e+01	7.33e-06	14	71.13
MOptQR	-3.7225751363e+01	8.38e-06	127	154.06
GR-BB	-3.7225751362e+01	9.69e-06	50	60.38
c2h6, $n = 2103$, $p = 7$				
SCF	-1.4420491322e+01	1.12e-06	11	10.09
TRDCM	-1.4420491322e+01	5.00e-06	12	7.61
MOptQR	-1.4420491322e+01	5.56e-06	49	8.53
GR-BB	-1.4420491321e+01	9.84e-06	43	7.58
c12h26, $n = 5709$, $p = 37$				
SCF	-8.1536091936e+01	1.52e-06	16	288.09
TRDCM	-8.1536091937e+01	9.48e-06	15	171.38
MOptQR	-8.1536091935e+01	9.51e-06	442	1296.05
GR-BB	-8.1536091936e+01	8.85e-06	50	157.02
co2, $n = 2103$, $p = 8$				
SCF	-3.5124395801e+01	1.50e-06	11	11.92
TRDCM	-3.5124395801e+01	7.63e-06	13	8.72
MOptQR	-3.5124395800e+01	9.03e-06	39	7.53
GR-BB	-3.5124395801e+01	6.94e-06	39	7.52

表 6.2 Kohn-Sham 总能量极小化问题的数值比较

Table 6.2 The results in Kohn-Sham total energy minimization

算法	E_{tot}	KKT 违反度	总迭代数	CPU 时间 (s)
ctube661, $n = 12599$, $p = 48$				
SCF	-1.3463843176e+02	2.80e-06	13	532.25
TRDCM	-1.3463843176e+02	5.77e-06	22	787.58
MOptQR	-1.3463843177e+02	5.06e-06	533	3817.95
GR-BB	-1.3463843176e+02	9.27e-06	68	493.53
glutamine, $n = 16517$, $p = 29$				
SCF	-9.1839425243e+01	2.88e-06	17	616.73
TRDCM	-9.1839425244e+01	8.49e-06	15	479.34
MOptQR	-9.1839425243e+01	7.26e-06	87	570.86
GR-BB	-9.1839425243e+01	9.76e-06	75	499.92
graphene16, $n = 3071$, $p = 37$				
SCF	-9.3873673630e+01	5.28e-03	200	2008.61
TRDCM	-9.4046217545e+01	6.12e-06	43	313.88
MOptQR	-9.4046217540e+01	9.56e-06	693	1110.39
GR-BB	-9.4046217543e+01	8.35e-06	321	513.45
graphene30, $n = 12279$, $p = 67$				
SCF	-1.7358503892e+02	3.18e-03	200	15344.80
TRDCM	-1.7359510505e+02	9.77e-06	62	3768.22
MOptQR	-1.6908746446e+02	3.87e+00	1000	11930.80
GR-BB	-1.7359510453e+02	1.97e-04	1000	12027.63
h2o, $n = 2103$, $p = 4$				
SCF	-1.6440507246e+01	7.78e-07	9	5.48
TRDCM	-1.6440507246e+01	8.22e-06	11	4.55
MOptQR	-1.6440507245e+01	8.43e-06	44	5.13
GR-BB	-1.6440507245e+01	9.89e-06	42	4.53
hnco, $n = 2103$, $p = 8$				
SCF	-1.6440507246e+01	7.08e-07	9	5.52
TRDCM	-1.6440507246e+01	9.64e-06	11	4.27
MOptQR	-1.6440507245e+01	9.20e-06	82	10.41
GR-BB	-1.6440507246e+01	8.64e-06	40	5.11

表 6.3 Kohn-Sham 总能量极小化问题的数值比较

Table 6.3 The results in Kohn-Sham total energy minimization

算法	E_{tot}	KKT 违反度	总迭代数	CPU 时间 (s)
nic, $n = 251$, $p = 7$				
SCF	-2.3543529955e+01	1.10e-06	12	3.13
TRDCM	-2.3543529955e+01	9.33e-06	49	5.50
MOptQR	-2.3543529955e+01	8.26e-06	100	2.84
GR-BB	-2.3543529955e+01	9.56e-06	39	0.88
pentacene, $n = 44791$, $p = 51$				
SCF	-1.3189029495e+02	9.83e-07	15	2448.72
TRDCM	-1.3189029495e+02	9.67e-06	23	2706.14
MOptQR	-1.3189029495e+02	7.02e-06	355	9145.66
GR-BB	-1.3189029495e+02	9.54e-06	100	2606.81
ptnio, $n = 4609$, $p = 43$				
SCF	-2.2678884273e+02	8.25e-07	70	1079.14
TRDCM	-2.2678882962e+02	2.93e-04	200	1957.89
MOptQR	-2.2678884235e+02	2.33e-05	1000	2281.22
GR-BB	-2.2678884272e+02	9.68e-06	512	1159.91
qdot, $n = 2103$, $p = 8$				
SCF	2.7702342351e+01	3.91e-02	200	175.16
TRDCM	2.7699896368e+01	2.72e-03	200	104.80
MOptQR	3.1736592205e+01	3.96e+00	1000	135.88
GR-BB	2.7700280932e+01	7.90e-04	1000	138.98
si2h4, $n = 2103$, $p = 6$				
SCF	-6.3009750460e+00	4.98e-07	13	12.42
TRDCM	-6.3009750459e+00	7.39e-06	16	9.09
MOptQR	-6.3009750460e+00	3.83e-06	75	11.67
GR-BB	-6.3009750457e+00	6.58e-06	58	8.97
sih4, $n = 2103$, $p = 4$				
SCF	-6.1769279851e+00	8.83e-07	10	5.80
TRDCM	-6.1769279850e+00	9.59e-06	10	4.50
MOptQR	-6.1769279851e+00	3.76e-06	42	5.14
GR-BB	-6.1769279850e+00	9.03e-06	36	4.41

表 6.4 Kohn-Sham 总能量极小化问题的数值比较

Table 6.4 The results in Kohn-Sham total energy minimization

详细的数值结果如表 6.5, 6.6 和 6.7 所示. 从表中我们观察到 PCAL 整体上相较于其他算法有更好的表现, 并且在大多数的例子中, PCAL 在得到相近函数值的前提下有更小的 KKT 违反度. 特别的, 在大规模的问题 “graphene30” 中, PCAL 所需的时间远少于其他算法. 在问题 “qdot” 中, 我们注意到只有 PLAM 和 PCAL 能够输出满足预设 KKT 精度的解, 而其他算法都会异常终止. 因此, 我们得出结论, 算法 PCAL 和 PLAM 在求解离散的 Kohn-Sham 总能量极小化问题时, 优于其他已有的可行方法.

上面列出的详细数值结果并不能直观展示一个算法的整体表现. 类似于 3.5.4 小节, 我们采用文献 [134] 提出的综合性能比较. 其关于 CPU 时间的数值比较结果如图 6.1 所示. 对于 Kohn-Sham 总能量极小化问题, 通过观察我们发现在所有 6 个算法中, PCAL 在 CPU 时间上表现最好.

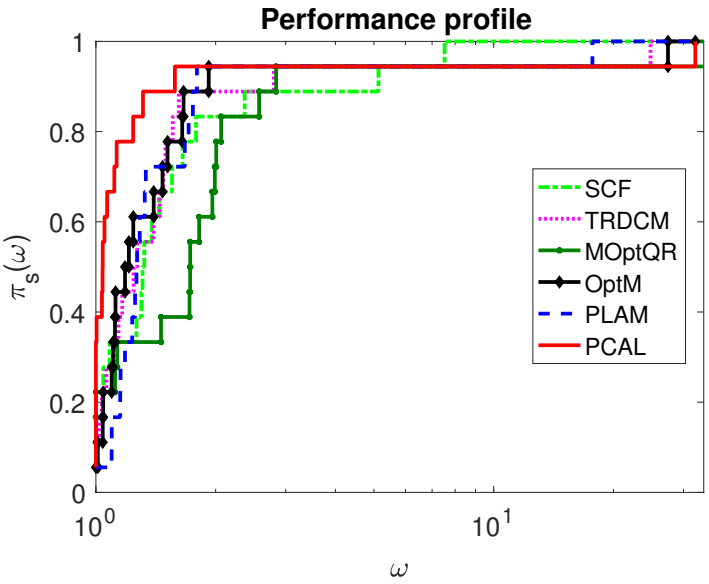


图 6.1 CPU 时间的综合性能比较

Figure 6.1 Performance profile in CPU time

6.3.5 PCAL 的并行测试

在本节最后, 我们在并行环境下测试算法 PCAL 和 MOptQR 应用于问题 (6.7) 的数值表现. 在本问题中, 我们取 $n = 10000$, $p = 1000$. 本小节的全部数值实验都是在 LSSC-IV 集群的一个单节点编译并运行的. 关于其具体信息, 请参考 5.5 节.

图 6.2 展示了不同算法之间的并行加速比. 其中 “Total” 表示算法总运行时间的加速比. 我们发现, PCAL 的总体可扩展性要远远优于流形收缩类算法 MOptQR.

算法	E_{tot}	KKT 违反度	总迭代数	可行性违反度	CPU 时间
	al, $n = 16879, p = 12$		($\beta_{\text{PLAM}} = 10, \beta_{\text{PCAL}} = 1$)		
SCF	-1.5789379003e+01	4.88e-03	200	6.53e-15	539.51
TRDCM	-1.5803791151e+01	6.36e-06	154	4.94e-15	336.79
MOptQR	-1.5803814080e+01	1.88e-04	1000	1.33e-14	393.54
OptM	-1.5803791098e+01	2.38e-05	1000	3.19e-14	378.80
PLAM	-1.5803790675e+01	1.29e-05	1000	3.34e-07	399.80
PCAL	-1.5803791055e+01	8.96e-06	596	5.95e-15	228.06
	alanine, $n = 12671, p = 18$		($\beta_{\text{PLAM}} = 13, \beta_{\text{PCAL}} = 1$)		
SCF	-6.1161921212e+01	3.80e-07	13	7.20e-15	21.46
TRDCM	-6.1161921213e+01	6.02e-06	15	5.20e-15	16.97
MOptQR	-6.1161921213e+01	7.52e-06	64	6.77e-15	14.89
OptM	-6.1161921213e+01	2.27e-06	69	4.03e-14	16.44
PLAM	-6.1161921212e+01	9.50e-06	76	7.90e-15	17.14
PCAL	-6.1161921213e+01	4.14e-06	61	7.19e-15	15.89
	benzene, $n = 8407, p = 15$		($\beta_{\text{PLAM}} = 10, \beta_{\text{PCAL}} = 1$)		
SCF	-3.7225751349e+01	2.10e-07	10	7.82e-15	10.07
TRDCM	-3.7225751363e+01	9.23e-06	15	7.12e-15	9.83
MOptQR	-3.7225751362e+01	8.12e-06	146	7.24e-15	19.91
OptM	-3.7225751363e+01	2.50e-06	70	1.54e-14	9.61
PLAM	-3.7225751362e+01	9.37e-06	71	4.62e-15	9.55
PCAL	-3.7225751362e+01	9.22e-06	50	5.15e-15	7.74
	c2h6, $n = 2103, p = 7$		($\beta_{\text{PLAM}} = 10, \beta_{\text{PCAL}} = 1$)		
SCF	-1.4420491315e+01	3.70e-09	10	3.66e-15	3.40
TRDCM	-1.4420491322e+01	8.75e-06	13	2.76e-15	4.01
MOptQR	-1.4420491321e+01	8.59e-06	47	2.58e-15	2.57
OptM	-1.4420491322e+01	2.62e-06	55	1.18e-14	2.87
PLAM	-1.4420491322e+01	7.91e-06	69	2.92e-15	3.41
PCAL	-1.4420491322e+01	4.91e-06	45	2.33e-15	2.58
	c12h26, $n = 5709, p = 37$		($\beta_{\text{PLAM}} = 10, \beta_{\text{PCAL}} = 1$)		
SCF	-8.1536091894e+01	4.95e-08	14	1.40e-14	30.08
TRDCM	-8.1536091937e+01	4.84e-06	16	1.17e-14	21.77
MOptQR	-8.1536091936e+01	6.68e-06	147	1.43e-14	39.57
OptM	-8.1536091937e+01	1.07e-06	83	7.10e-14	22.65
PLAM	-8.1536091936e+01	5.88e-06	96	1.55e-14	25.11
PCAL	-8.1536091936e+01	8.75e-06	70	1.45e-14	22.88
	co2, $n = 2103, p = 8$		($\beta_{\text{PLAM}} = 10, \beta_{\text{PCAL}} = 1$)		
SCF	-3.5124395789e+01	6.17e-08	10	2.53e-15	2.61
TRDCM	-3.5124395801e+01	4.14e-06	14	4.11e-15	2.09
MOptQR	-3.5124395800e+01	9.30e-06	88	2.35e-15	2.90
OptM	-3.5124395801e+01	1.70e-06	48	3.55e-14	1.68
PLAM	-3.5124395801e+01	7.92e-06	57	2.30e-15	1.84
PCAL	-3.5124395801e+01	9.15e-06	43	2.11e-15	1.74

表 6.5 Kohn-Sham 总能量极小化问题的数值比较

Table 6.5 The results in Kohn-Sham total energy minimization

算法	E_{tot}	KKT 违反度	总迭代数	可行性违反度	CPU 时间
	ctube661, $n = 12599$, $p = 48$		($\beta_{\text{PLAM}} = 13, \beta_{\text{PCAL}} = 1$)		
SCF	-1.3463843175e+02	3.88e-07	11	1.43e-14	56.43
TRDCM	-1.3463843176e+02	6.85e-06	23	1.09e-14	87.41
MOptQR	-1.3463843176e+02	7.21e-06	152	1.78e-14	107.62
OptM	-1.3463843176e+02	2.35e-06	82	2.15e-14	59.23
PLAM	-1.3463843176e+02	4.34e-06	107	2.37e-14	72.18
PCAL	-1.3463843176e+02	9.68e-06	65	1.95e-14	54.07
	glutamine, $n = 16517$, $p = 29$		($\beta_{\text{PLAM}} = 13, \beta_{\text{PCAL}} = 1$)		
SCF	-9.1839425202e+01	1.12e-07	15	1.07e-14	67.40
TRDCM	-9.1839425244e+01	3.23e-06	16	7.00e-15	54.65
MOptQR	-9.1839425243e+01	9.83e-06	78	9.07e-15	51.46
OptM	-9.1839425244e+01	2.47e-06	87	9.73e-15	57.65
PLAM	-9.1839425243e+01	8.72e-06	104	9.26e-15	66.31
PCAL	-9.1839425243e+01	6.28e-06	74	9.33e-15	53.53
	graphene16, $n = 3071$, $p = 37$		($\beta_{\text{PLAM}} = 10, \beta_{\text{PCAL}} = 1$)		
SCF	-9.4023322108e+01	2.07e-03	200	1.32e-14	309.33
TRDCM	-9.4046217545e+01	8.85e-06	45	1.08e-14	47.87
MOptQR	-9.4046217225e+01	9.90e-06	422	1.15e-14	80.67
OptM	-9.4046217545e+01	2.27e-06	245	1.03e-14	48.66
PLAM	-9.4046217854e+01	9.52e-06	278	1.34e-14	51.57
PCAL	-9.4046217542e+01	8.68e-06	176	1.17e-14	41.11
	graphene30, $n = 12279$, $p = 67$		($\beta_{\text{PLAM}} = 13, \beta_{\text{PCAL}} = 1$)		
SCF	-1.7358453985e+02	5.19e-03	200	1.93e-14	2815.79
TRDCM	-1.7359510506e+02	4.80e-06	71	1.42e-14	765.92
MOptQR	-1.7359510505e+02	9.92e-06	456	2.59e-14	800.08
OptM	-1.7359510506e+02	2.47e-06	472	2.49e-14	904.44
PLAM	-1.7359510505e+02	8.88e-06	330	2.75e-14	601.41
PCAL	-1.7359510505e+02	8.52e-06	253	2.62e-14	548.70
	h2o, $n = 2103$, $p = 4$		($\beta_{\text{PLAM}} = 10, \beta_{\text{PCAL}} = 1$)		
SCF	-1.6440507245e+01	1.16e-08	8	1.15e-15	1.29
TRDCM	-1.6440507246e+01	6.48e-06	11	1.11e-15	1.02
MOptQR	-1.6440507246e+01	3.84e-06	49	9.30e-16	1.14
OptM	-1.6440507246e+01	2.01e-06	61	6.40e-15	1.50
PLAM	-1.6440507245e+01	6.43e-06	56	2.37e-15	1.29
PCAL	-1.6440507246e+01	7.42e-06	42	1.86e-15	1.06
	hnco, $n = 2103$, $p = 8$		($\beta_{\text{PLAM}} = 10, \beta_{\text{PCAL}} = 1$)		
SCF	-2.8634664360e+01	9.44e-08	12	3.82e-15	4.32
TRDCM	-2.8634664365e+01	9.54e-06	13	3.47e-15	4.47
MOptQR	-2.8634664363e+01	9.74e-06	163	3.17e-15	12.26
OptM	-2.8634664365e+01	5.30e-06	117	2.26e-15	8.30
PLAM	-2.8634664364e+01	9.95e-06	105	3.18e-15	7.39
PCAL	-2.8634664364e+01	9.03e-06	70	2.60e-15	5.36

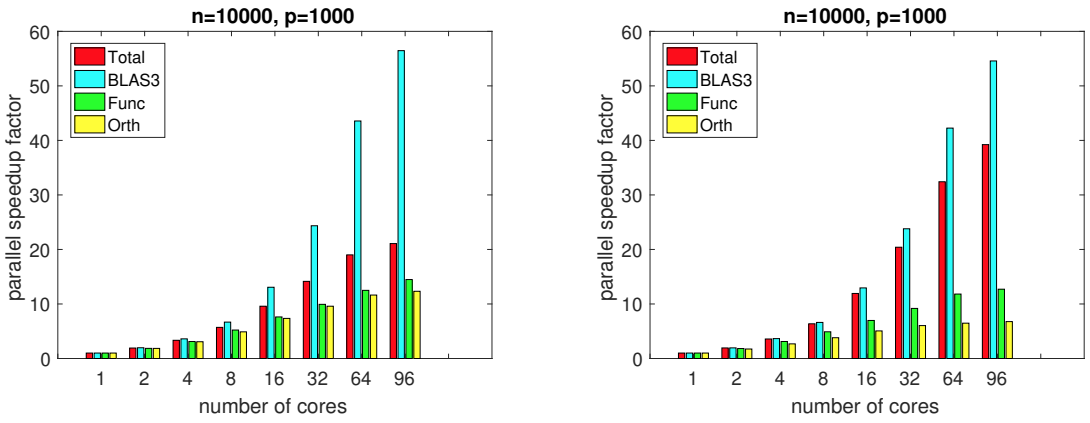
表 6.6 Kohn-Sham 总能量极小化问题的数值比较

Table 6.6 The results in Kohn-Sham total energy minimization

算法	E_{tot}	KKT 违反度	总迭代数	可行性违反度	CPU 时间
	nic, $n = 251$, $p = 7$		($\beta_{\text{PLAM}} = 10, \beta_{\text{PCAL}} = 1$)		
SCF	-2.3543529950e+01	2.13e-10	11	2.99e-15	1.47
TRDCM	-2.3543529955e+01	7.94e-06	15	4.49e-15	0.99
MOptQR	-2.3543529955e+01	3.04e-06	111	2.73e-15	1.53
OptM	-2.3543529955e+01	3.86e-07	63	8.80e-15	0.90
PLAM	-2.3543529955e+01	4.02e-06	67	1.39e-15	0.89
PCAL	-2.3543529955e+01	8.42e-06	52	1.88e-15	0.99
	pentacene, $n = 44791$, $p = 51$		($\beta_{\text{PLAM}} = 13, \beta_{\text{PCAL}} = 1$)		
SCF	-1.3189029494e+02	5.76e-07	13	1.58e-14	293.68
TRDCM	-1.3189029495e+02	7.60e-06	22	1.08e-14	276.25
MOptQR	-1.3189029495e+02	7.78e-06	112	3.21e-14	306.97
OptM	-1.3189029495e+02	1.39e-06	97	3.39e-14	283.02
PLAM	-1.3189029495e+02	8.66e-06	123	3.52e-14	321.04
PCAL	-1.3189029495e+02	7.67e-06	89	3.08e-14	271.32
	ptnio, $n = 4069$, $p = 43$		($\beta_{\text{PLAM}} = 13, \beta_{\text{PCAL}} = 1$)		
SCF	-2.2678884268e+02	1.09e-05	53	1.46e-14	168.25
TRDCM	-2.2678882693e+02	2.81e-04	200	1.07e-14	471.34
MOptQR	-2.2678884271e+02	9.57e-06	786	1.06e-14	347.38
OptM	-2.2678884273e+02	9.52e-06	508	1.14e-14	203.63
PLAM	-2.2678884271e+02	9.00e-06	579	1.01e-14	213.60
PCAL	-2.2678884271e+02	8.55e-06	386	1.19e-14	189.70
	qdot, $n = 2103$, $p = 8$		($\beta_{\text{PLAM}} = 10, \beta_{\text{PCAL}} = 1$)		
SCF	2.7700280133e+01	6.70e-03	5	2.92e-15	1.09
TRDCM	2.7699537080e+01	1.43e-02	200	2.73e-15	27.01
MOptQR	1.0483319768e+02	3.45e+01	1000	1.77e-15	28.72
OptM	2.7699807230e+01	1.45e-04	1000	2.39e-15	29.89
PLAM	2.7699800860e+01	9.68e-06	678	1.98e-15	19.30
PCAL	2.7699800851e+01	5.41e-06	962	2.88e-15	35.01
	si2h4, $n = 2103$, $p = 6$		($\beta_{\text{PLAM}} = 10, \beta_{\text{PCAL}} = 1$)		
SCF	-6.3009750375e+00	5.25e-07	11	3.62e-15	2.97
TRDCM	-6.3009750459e+00	8.24e-06	16	3.12e-15	4.30
MOptQR	-6.3009750460e+00	3.70e-06	116	2.00e-15	5.96
OptM	-6.3009750459e+00	9.60e-06	68	1.41e-14	4.15
PLAM	-6.3009750455e+00	7.27e-06	89	1.58e-15	5.33
PCAL	-6.3009750459e+00	4.33e-06	62	2.42e-15	3.90
	sih4, $n = 2103$, $p = 4$		($\beta_{\text{PLAM}} = 10, \beta_{\text{PCAL}} = 1$)		
SCF	-6.1769279820e+00	2.07e-08	8	1.75e-15	1.91
TRDCM	-6.1769279850e+00	9.53e-06	10	1.14e-15	1.60
MOptQR	-6.1769279851e+00	4.32e-06	34	1.58e-15	1.07
OptM	-6.1769279851e+00	8.18e-06	46	8.52e-16	1.62
PLAM	-6.1769279849e+00	7.37e-06	56	1.99e-15	1.79
PCAL	-6.1769279847e+00	9.16e-06	47	1.55e-15	1.69

表 6.7 Kohn-Sham 总能量极小化问题的数值比较

Table 6.7 The results in Kohn-Sham total energy minimization



(a) MOptQR (b) PCAL

图 6.2 MOptQR 和 PCAL 的并行加速比比较 (Kohn-Sham 总能量极小化问题)

Figure 6.2 A comparison of speedup factor among MOptQR and PCAL on the simplified Kohn-Sham total energy minimization

6.4 小结

在本章中, 我们考虑了正交约束优化问题的一个具体应用: 电子结构计算. 其中, Kohn-Sham 总能量极小化问题是描述电子结构的一种重要方式. 这类问题通常具有较大规模, 并且正交化过程的可扩展性差. 因此, 我们考虑将本文第 3, 5 章提出的高效算法 GR-BB, PLAM/PCAL 应用于此类问题. 通过 Kohn-Sham 总能量极小化问题的数值求解, 我们发现我们提出的新算法相较于已有的算法而言, 数值表现优异, 并且算法 PCAL 还具有较高的可扩展性. 从而, 我们从实际应用的角度验证了本文的算法高效且实用.

第 7 章　总结与展望

本文分别从理论、算法与应用三个角度, 系统的研究了正交约束优化问题. 主要内容包括:

在第 2 章中, 我们详细介绍了正交约束优化问题的应用背景. 接着, 我们分别从 Stiefel 流形优化和欧式空间约束优化两个不同角度, 推导了问题的最优性条件. 其中通过分析一大类黎曼度量下的黎曼梯度, 我们得到了 Stiefel 流形优化和欧式空间中约束优化的一阶最优性条件的对应关系. 除此之外, 我们证明了在任意一阶稳定点处, 正交约束优化问题的 Lagrange 乘子具有显式表达式. 这些性质和推论启发了本文的部分算法设计.

在第 3 章中, 对于一大类正交约束优化问题, 我们提出了一个新的一阶算法框架. 其包含两个步骤. 第一步, 我们选取函数值下降方法使得函数值减少并同时保持迭代点的可行性, 因此关于 Stiefel 流形的复杂计算可以被省略. 第二步, 我们利用乘子校正步来保证迭代点列的任意聚点都是一阶稳定点. 进一步, 对于一些特殊情况, 此校正步可以被省略. 基于此算法框架, 我们提出了两大类算法. 第一类是梯度下降方法, 其中包括梯度反射法和梯度投影法. 第二类采用以列为块的块坐标下降方法, 其列坐标的更新顺序由 Gauss-Seidel 类型决定. 同时, 我们也提出了一个新的方法用来非精确高效的求解子问题, 并保证了算法的全局收敛性. 针对一大类不同的测试问题, 数值实验显示了我们的新算法具有巨大潜力.

在第 4 章中, 我们将乘子校正步推广到一般的 Stiefel 流形收缩类算法, 得到了子空间加速的收缩类算法. 通过考虑一个更小的限制在子空间的优化问题, 我们可以将原有的可行下降算法进行加速. 其中, 我们特别考虑了收缩类线搜索算法, 由此得到了加速的收缩类算法. 进一步, 我们给出了算法的全局收敛性以及局部线性收敛速度. 数值实验说明了我们的加速技术高效且实用.

在第 5 章中, 针对一般的正交约束优化问题, 我们提出了基于增广 Lagrange 函数的并行算法. 考虑到正交化过程的低可扩展性. 我们采用不可行方法. 基于增广 Lagrange 罚函数, 我们提出了一个更实际可行更高效的可并行算法. 通过利用 Lagrange 乘子在任意一阶稳定点处的显式表达式, 我们给出了改进后的增广 Lagrange 算法. 在每一步迭代中, 子问题的求解我们只需要一次增广 Lagrange 梯度步, 对于对偶变量, 我们采取了新的显式更新方式. 进一步, 考虑到迭代点列的有界性, 我们还提出了一个改进的算法版本. 在实际计算中, 我们并行实现了新算

法, 在并行环境下, 我们算法的可扩展性远远优于已有的收缩类可行方法. 由此可见, 我们的算法在实际应用中具有巨大的潜力.

在第 6 章中, 我们考虑了正交约束优化问题的一个具体应用: 电子结构计算. 其中, Kohn-Sham 总能量极小化问题是描述电子结构的一种重要方式. 这类问题通常具有较大规模, 因此, 我们考虑将本文第 3, 5 章提出的高效算法 GR-BB, PLAM/PCAL 应用于此类问题. 通过 Kohn-Sham 总能量极小化问题的数值求解, 我们发现新算法相较于已有的算法, 数值表现优异, 并且算法 PCAL 还具有较高的可扩展性. 由此, 我们从实际应用的角度说明了本文的算法高效且实用.

除了本文的研究内容外, 正交约束优化问题还有许多值得研究的课题. 例如:

如何设计二阶方法进一步提升算法的表现. 如何调整算法或者利用全局化的策略获得问题的全局最优解. 考虑到正交化过程的低可扩展性, 如何设计可并行的 Jacobi 类型的块坐标下降法.

目前, 本文的加速技术只适用于具有特殊结构的正交约束优化问题, 如何将其推广到一般的正交约束优化问题是个值得考虑的研究方向. 此外, 我们还可以考虑将子空间加速技术应用于 Stiefel 流形优化中更多的可行下降算法.

除此之外, 如何调整我们的算法使其更适于特征值问题的计算也是另一个需要考虑的方向.

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. 非线性优化计算方法[M]. 科学出版社, 2008.

[3] NOCEDAL J, WRIGHT S J. Numerical optimization, 2nd[M]. Springer, 2006.

[4] SUN W, YUAN Y X. Optimization theory and methods: nonlinear programming: volume 1 [M]. Springer Science & Business Media, 2006.

[5] ROCKAFELLAR R T. Convex analysis[M]. Princeton university press, 2015.

[6] BOYD S, VANDENBERGHE L. Convex optimization[M]. Cambridge university press, 2004.

[7] CAUCHY A. Méthode générale pour la résolution des systemes d’équations simultanées[J]. Comp. Rend. Sci. Paris, 1847, 25(1847):536-538.

[8] AKAIKE H. On a successive transformation of probability distribution and its application to the analysis of the optimum gradient method[J]. Annals of the Institute of Statistical Mathematics, 1959, 11(1):1-16.

[9] NOCEDAL J, SARTENAER A, ZHU C. On the behavior of the gradient norm in the steepest descent method[J]. Computational Optimization and Applications, 2002, 22(1):5-35.

[10] BARZILAI J, BORWEIN J M. Two-point step size gradient methods[J]. IMA Journal of Numerical Analysis, 1988, 8(1):141-148.

[11] DENNIS J E, Jr, MORÉ J J. Quasi-Newton methods, motivation and theory[J]. SIAM review, 1977, 19(1):46-89.

[12] RAYDAN M. The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem[J]. SIAM Journal on Optimization, 1997, 7(1):26-33.

[13] FLETCHER R. On the Barzilai-borwein method[M]//Optimization and control with applications. Springer, 2005: 235-256.

[14] BIRGIN E G, MARTÍNEZ J M, RAYDAN M. Spectral projected gradient methods: review and perspectives[J]. J. Stat. Softw, 2014, 60(3):1-21.

[15] DAI Y H, HUANG Y, LIU X W. A family of spectral gradient methods for optimization[J]. arXiv preprint arXiv:1812.02974, 2018.

[16] DAI Y H. A new analysis on the Barzilai-Borwein gradient method[J]. Journal of the operations Research Society of China, 2013, 1(2):187-198.

[17] RAYDAN M. On the Barzilai and Borwein choice of steplength for the gradient method[J]. IMA Journal of Numerical Analysis, 1993, 13(3):321-326.

[18] DAI Y H, LIAO L Z. R-linear convergence of the Barzilai and Borwein gradient method[J]. IMA Journal of Numerical Analysis, 2002, 22(1):1-10.

[19] DAI Y H, FLETCHER R. Projected Barzilai-Borwein methods for large-scale box-constrained quadratic programming[J]. Numerische Mathematik, 2005, 100(1):21-47.

[20] BIRGIN E G, MARTÍNEZ J M, RAYDAN M. Nonmonotone spectral projected gradient methods on convex sets[J]. SIAM Journal on Optimization, 2000, 10(4):1196-1211.

[21] BOOTHBY W M. An introduction to differentiable manifolds and Riemannian geometry: volume 120[M]. Academic press, 1986.

[22] ZHAO Z, BAI Z J, JIN X Q. A Riemannian Newton algorithm for nonlinear eigenvalue problems[J]. SIAM Journal on Matrix Analysis and Applications, 2015, 36(2):752-774.

[23] ZHANG X, ZHU J, WEN Z, ZHOU A. Gradient type optimization methods for electronic structure calculations[J]. SIAM Journal on Scientific Computing, 2014, 36(3):C265-C289.

[24] HU J, MILZAREK A, WEN Z, YUAN Y. Adaptive quadratically regularized Newton method for Riemannian optimization[J]. SIAM Journal on Matrix Analysis and Applications, 2018, 39 (3):1181-1207.

[25] VANDEREYCKEN B. Low-rank matrix completion by Riemannian optimization[J]. SIAM Journal on Optimization, 2013, 23(2):1214-1236.

[26] KRESSNER D, STEINLECHNER M, VANDEREYCKEN B. Low-rank tensor completion by Riemannian optimization[J]. BIT Numerical Mathematics, 2014, 54(2):447-468.

[27] HUANG W, GALLIVAN K A, SRIVASTAVA A, ABSIL P A. Riemannian optimization for registration of curves in elastic shape analysis[J]. Journal of Mathematical Imaging and Vision, 2016, 54(3):320-343.

[28] THEIS F J, CASON T P, ABSIL P A. Soft dimension reduction for ICA by joint diagonalization on the Stiefel manifold[C]//International Conference on Independent Component Analysis and Signal Separation. Springer, 2009: 354-361.

[29] JIA C, EVANS B L. 3D rotational video stabilization using manifold optimization[C]//2013 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 2013: 2493-2497.

[30] JIANG B, MA S, SO A M C, ZHANG S. Vector transport-free SVRG with general retraction for Riemannian optimization: Complexity analysis and practical implementation[J]. arXiv preprint arXiv:1705.09059, 2017.

[31] CHERIAN A, SRA S. Riemannian dictionary learning and sparse coding for positive definite matrices[J]. IEEE transactions on neural networks and learning systems, 2017, 28(12):2859-2871.

[32] HOSSEINI R, SRA S. Matrix manifold optimization for Gaussian mixtures[C]//Advances in Neural Information Processing Systems. 2015: 910-918.

[33] ABSIL P A, MAHONY R, SEPULCHRE R. Optimization algorithms on matrix manifolds[M]. Princeton University Press, 2009.

[34] HIRSCH M W. Differential topology: volume 33[M]. Springer Science & Business Media, 2012.

[35] HELMKE U, MOORE J B. Optimization and dynamical systems[M]. Springer-Verlag„ 1994.

[36] SHUB M. Some remarks on dynamical systems and numerical analysis[J]. Proc. VII ELAM.(L. Lara-Carrero and J. Lewowicz, eds.), Equinoccio, U. Simón Bolıvar, Caracas, 1986:69-92.

[37] ADLER R L, DEDIEU J P, MARGULIES J Y, MARTENS M, SHUB M. Newton's method on Riemannian manifolds and a geometric model for the human spine[J]. IMA Journal of Numerical Analysis, 2002, 22(3):359-390.

[38] ARMIJO L. Minimization of functions having Lipschitz continuous first partial derivatives[J]. Pacific Journal of mathematics, 1966, 16(1):1-3.

[39] LUENBERGER D G. The gradient projection method along geodesics[J]. Management Science, 1972, 18(11):620-631.

[40] ABSIL P A, GALLIVAN K A. Accelerated line-search and trust-region methods[J]. SIAM Journal on Numerical Analysis, 2009, 47(2):997-1018.

[41] ABSIL P A, BAKER C G, GALLIVAN K A. Trust-region methods on Riemannian manifolds [J]. Foundations of Computational Mathematics, 2007, 7(3):303-330.

[42] ABSIL P A, MALICK J. Projection-like retractions on matrix manifolds[J]. SIAM Journal on Optimization, 2012, 22(1):135-158.

[43] HUANG W. Optimization algorithms on Riemannian manifolds with applications[J]. Ph.D. Thesis, 2013.

[44] 陈国良. 并行计算: 结构 · 算法 · 编程[M]. 高等教育出版社, 2011.

[45] GROPP W D, GROPP W, LUSK E, SKJELLUM A, LUSK A D F E E. Using MPI: portable parallel programming with the message-passing interface: volume 1[M]. MIT press, 1999.

[46] NICHOLS B, BUTTLAR D, FARRELL J, FARRELL J. Pthreads programming: A POSIX standard for better multiprocessing[M]. O'Reilly Media, Inc., 1996.

[47] CHAPMAN B, JOST G, VAN DER PAS R. Using OpenMP: portable shared memory parallel programming: volume 10[M]. MIT press, 2008.

[48] BECK A, TEBOULLE M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems[J]. SIAM journal on imaging sciences, 2009, 2(1):183-202.

[49] WEN Z, YANG C, LIU X, ZHANG Y. Trace-penalty minimization for large-scale eigenspace computation[J]. Journal of Scientific Computing, 2016, 66(3):1175-1203.

[50] FERCOQ O, RICHTÁRIK P. Accelerated, parallel, and proximal coordinate descent[J]. SIAM Journal on Optimization, 2015, 25(4):1997-2023.

[51] BOYD S, PARIKH N, CHU E, PELEATO B, ECKSTEIN J. Distributed optimization and statistical learning via the alternating direction method of multipliers[J]. Foundations and Trends® in Machine learning, 2011, 3(1):1-122.

[52] DONG Q, LIU X, WEN Z W, YUAN Y X. A parallel line search subspace correction method for composite convex optimization[J]. Journal of the Operations Research Society of China, 2015, 3(2):163-187.

[53] RECHT B, RE C, WRIGHT S, NIU F. Hogwild: A lock-free approach to parallelizing stochastic gradient descent[C]//Advances in neural information processing systems. 2011: 693-701.

[54] PENG Z, YAN M, YIN W. Parallel and distributed sparse optimization[C]//Signals, Systems and Computers, 2013 Asilomar Conference on. IEEE, 2013: 659-646.

[55] LIU J, WRIGHT S J, RÉ C, BITTORF V, SRIDHAR S. An asynchronous parallel stochastic coordinate descent algorithm[J]. The Journal of Machine Learning Research, 2015, 16(1):285-322.

[56] PENG Z, XU Y, YAN M, YIN W. Arock: an algorithmic framework for asynchronous parallel coordinate updates[J]. SIAM Journal on Scientific Computing, 2016, 38(5):A2851-A2879.

[57] DAI X, LIU Z, ZHANG L, ZHOU A. A conjugate gradient method for electronic structure calculations[J]. SIAM Journal on Scientific Computing, 2017, 39(6):A2702-A2740.

[58] GOLUB G H, VAN LOAN C F. Matrix computations: volume 3[M]. Johns Hopkins University Press, 2012.

[59] NESTEROV Y. Gradient methods for minimizing composite functions[J]. Mathematical Programming, 2013, 140(1):125-161.

[60] NESTEROV Y. Efficiency of coordinate descent methods on huge-scale optimization problems [J]. SIAM Journal on Optimization, 2012, 22(2):341-362.

[61] ZHAO T, YU M, WANG Y, ARORA R, LIU H. Accelerated mini-batch randomized block coordinate descent method[C]//Advances in neural information processing systems. 2014: 3329-3337.

[62] BECK A, TETRUASHVILI L. On the convergence of block coordinate descent type methods [J]. SIAM journal on Optimization, 2013, 23(4):2037-2060.

[63] RICHTÁRIK P, TAKÁČ M. Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function[J]. Mathematical Programming, 2014, 144(1-2): 1-38.

[64] LU Z, XIAO L. On the complexity analysis of randomized block-coordinate descent methods [J]. Mathematical Programming, 2015, 152(1-2):615-642.

[65] XU Y, YIN W. Block stochastic gradient iteration for convex and nonconvex optimization[J]. SIAM Journal on Optimization, 2015, 25(3):1686-1716.

[66] STIEFEL E. Richtungsfelder und fernparallelismus in n-dimensionalen mannigfaltigkeiten[J]. Commentarii Mathematici Helvetici, 1935, 8(1):305-353.

[67] GAREY M R, JOHNSON D S. Computers and intractability: A guide to the theory of NP-completeness[J]. Bull. Amer. Math. Soc, 1980, 3:898-904.

[68] LIU Y F, DAI Y H, LUO Z Q. On the complexity of leakage interference minimization for interference alignment[C]//2011 IEEE 12th international workshop on signal processing advances in wireless communications. IEEE, 2011: 471-475.

[69] HU J, JIANG B, LIU X, WEN Z. A note on semidefinite programming relaxations for polynomial optimization over a single sphere[J]. Science China Mathematics, 2016, 59(8):1543-1560.

[70] YANG C, MEZA J C, WANG L W. A constrained optimization algorithm for total energy

minimization in electronic structure calculations[J]. Journal of Computational Physics, 2006, 217(2):709-721.

[71] YANG C, MEZA J C, WANG L W. A trust region direct constrained minimization algorithm for the Kohn–Sham equation[J]. SIAM Journal on Scientific Computing, 2007, 29(5):1854-1875.

[72] YANG C, MEZA J C, LEE B, WANG L W. KSSOLV–a MATLAB toolbox for solving the Kohn–Sham equations[J]. ACM Transactions on Mathematical Software (TOMS), 2009, 36 (2):10.

[73] LIU X, WANG X, WEN Z, YUAN Y. 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.

[74] LIU X, WEN Z, WANG X, ULBRICH M, YUAN Y. On the analysis of the discretized Kohn–Sham density functional theory[J]. SIAM Journal on Numerical Analysis, 2015, 53(4):1758-1785.

[75] OZOLIŅŠ V, LAI R, CAFLISCH R, OSHER S. Compressed modes for variational problems in mathematics and physics[J]. Proceedings of the National Academy of Sciences, 2013, 110 (46):18368-18373.

[76] ZHU H, ZHANG X, CHU D, LIAO L Z. Nonconvex and nonsmooth optimization with generalized orthogonality constraints: An approximate augmented Lagrangian method[J]. Journal of Scientific Computing, 2017, 72(1):331-372.

[77] CABOUSSAT A, GLOWINSKI R, PONS V. An augmented Lagrangian approach to the numerical solution of a non-smooth eigenvalue problem[J]. Journal of Numerical Mathematics, 2009, 17(1):3-26.

[78] LIU X, HAO C, CHENG M. A sequential subspace projection method for linear symmetric eigenvalue problem[J]. Asia Pacific Journal of Operational Research, 2013, 30(3):1340003.

[79] GAO W, YANG C, MEZA J C. Solving a class of nonlinear eigenvalue problems by Newton's method[J]. Technical Reports, 2009.

[80] BAI Z, SLEIJPEN G, VAN DER VORST H, LIPPERT R, EDELMAN A. 9. nonlinear eigenvalue problems[M/OL]. 2000: 281-314. DOI: 10.1137/1.9780898719581.ch9.

[81] BAI Z, LU D, VANDEREYCKEN B. Robust Rayleigh quotient minimization and nonlinear eigenvalue problems[J]. SIAM Journal on Scientific Computing, 2018, 40(5):A3495-A3522.

[82] LIU X, WEN Z, ZHANG Y. Limited memory block Krylov subspace optimization for computing dominant singular value decompositions[J]. SIAM Journal on Scientific Computing, 2013, 35(3):A1641-A1668.

[83] PIETERSZ 4 R, GROENEN P J. Rank reduction of correlation matrices by majorization[J]. Quantitative Finance, 2004, 4(6):649-662.

[84] GRUBIŠIĆ I, PIETERSZ R. Efficient rank reduction of correlation matrices[J]. Linear algebra and its applications, 2007, 422(2):629-653.

[85] REBONATO R, JÄCKEL P. The most general methodology to create a valid correlation matrix for risk management and option pricing purposes[J]. Available at SSRN 1969689, 2011.

[86] ZOU H, HASTIE T, TIBSHIRANI R. Sparse principal component analysis[J]. Journal of computational and graphical statistics, 2006, 15(2):265-286.

[87] D'ASPREMONT A, EL GHAOUI L, JORDAN M I, LANCKRIET G R. A direct formulation for sparse PCA using semidefinite programming[J]. SIAM review, 2007, 49(3):434-448.

[88] SCHÖNEMANN P H. A generalized solution of the orthogonal Procrustes problem[J]. Psychometrika, 1966, 31(1):1-10.

[89] ELDÉN L, PARK H. A Procrustes problem on the Stiefel manifold[J]. Numerische Mathematik, 1999, 82(4):599-619.

[90] BOLLA M, MICHALETZKY G, TUSNÁDY G, ZIERMANN M. Extrema of sums of heterogeneous quadratic forms[J]. Linear Algebra and its Applications, 1998, 269(1-3):331-365.

[91] RAPCSÁK T. On minimization of sums of heterogeneous quadratic functions on Stiefel manifolds[M]//From local to global optimization. Springer, 2001: 277-290.

[92] PETERS S W, HEATH R W. Interference alignment via alternating minimization[C]//2009 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 2009: 2445-2448.

[93] JOHO M, MATHIS H. Joint diagonalization of correlation matrices by using gradient methods with application to blind signal separation[C]//Sensor Array and Multichannel Signal Processing Workshop Proceedings, 2002. IEEE, 2002: 273-277.

[94] SAMEH A H, WISNIEWSKI J A. A trace minimization algorithm for the generalized eigenvalue problem[J]. SIAM Journal on Numerical Analysis, 1982, 19(6):1243-1259.

[95] SAMEH A, TONG Z. The trace minimization method for the symmetric generalized eigenvalue problem[J]. Journal of Computational and Applied Mathematics, 2000, 123(1-2):155-175.

[96] DONGARRA J, GATES M, HAIDAR A, KURZAK J, LUSZCZEK P, TOMOV S, YAMAZAKI I. The singular value decomposition: Anatomy of optimizing an algorithm for extreme scale[J]. SIAM Review, 2018, 60(4):808-865.

[97] ECKART C, YOUNG G. The approximation of one matrix by another of lower rank[J]. Psychometrika, 1936, 1(3):211-218.

[98] SATO H, IWAI T. A Riemannian optimization approach to the matrix singular value decomposition[J]. SIAM Journal on Optimization, 2013, 23(1):188-212.

[99] WEN Z, YIN W. A feasible method for optimization with orthogonality constraints[J]. Mathematical Programming, 2013, 142(1-2):397-434.

[100] GRIFFIN A, SNOKE D W, STRINGARI S. Bose-Einstein condensation[M]. Cambridge University Press, 1996.

[101] NIE J. Regularization methods for sum of squares relaxations in large scale polynomial optimization[J]. Submitted for publication., September, 2009.

[102] JOLLIFFE I. Principal component analysis[M]. Springer, 2011.

[103] ZHANG H, REDDI S J, SRA S. Riemannian SVRG: Fast stochastic optimization on Riemannian manifolds[C]//Advances in Neural Information Processing Systems. 2016: 4592-4600.

[104] LIU H, WU W, SO A M C. Quadratic optimization with orthogonality constraints: Explicit Lojasiewicz exponent and linear convergence of line-search methods[C]//International Conference on Machine Learning. 2016: 1158-1167.

[105] KOHN W, SHAM L J. Self-consistent equations including exchange and correlation effects[J]. Physical review, 1965, 140(4A):A1133.

[106] JIANG B, DAI Y H. A framework of constraint preserving update schemes for optimization on Stiefel manifold[J]. Mathematical Programming, 2015, 153(2):535-575.

[107] RAPCSÁK T. On minimization on Stiefel manifolds[J]. European Journal of Operational Research, 2002, 143(2):365-376.

[108] MANTON J H. Optimization algorithms exploiting unitary constraints[J]. IEEE Transactions on Signal Processing, 2002, 50(3):635-650.

[109] NISHIMORI Y, AKAHO S. Learning algorithms utilizing quasi-geodesic flows on the Stiefel manifold[J]. Neurocomputing, 2005, 67:106-135.

[110] ABRUDAN T E, ERIKSSON J, KOIVUNEN V. Conjugate gradient algorithm for optimization under unitary matrix constraint[J]. Signal Processing, 2009, 89(9):1704-1714.

[111] EDELMAN A, ARIAS T A, SMITH S T. The geometry of algorithms with orthogonality constraints[J]. SIAM journal on Matrix Analysis and Applications, 1998, 20(2):303-353.

[112] ABSIL P A, BAKER C G, GALLIVAN K A. Trust-region methods on Riemannian manifolds with applications in numerical linear algebra[C]//Proceedings of the 16th International Symposium on Mathematical Theory of Networks and Systems (MTNS2004), Leuven, Belgium. 2004: 5-9.

[113] HUANG W, GALLIVAN K A, ABSIL P A. A Broyden class of quasi-Newton methods for Riemannian optimization[J]. SIAM Journal on Optimization, 2015, 25(3):1660-1685.

[114] HUANG W, ABSIL P A, GALLIVAN K A. A Riemannian BFGS method for nonconvex optimization problems[M]//Numerical Mathematics and Advanced Applications ENUMATH 2015. Springer, 2016: 627-634.

[115] GOLDFARB D, WEN Z, YIN W. A curvilinear search method for p-harmonic flows on spheres [J]. SIAM Journal on Imaging Sciences, 2009, 2(1):84-109.

[116] ZHANG H, HAGER W W. A nonmonotone line search technique and its application to unconstrained optimization[J]. SIAM journal on Optimization, 2004, 14(4):1043-1056.

[117] LAI R, OSHER S. A splitting method for orthogonality constrained problems[J]. Journal of Scientific Computing, 2014, 58(2):431-449.

[118] COURANT R. Variational methods for the solution of problems of equilibrium and vibrations [M]. Verlag nicht ermittelbar, 1943.

[119] JIANG B, CUI C, DAI Y H. Unconstrained optimization models for computing several extreme eigenpairs of real symmetric matrices[J]. Pacific Journal of Optimzation, 2014, 10(1):55-71.

[120] AUCHMUTY G. Unconstrained variational principles for eigenvalues of real symmetric matrices[J]. SIAM Journal on Mathematical Analysis, 1989, 20(5):1186-1207.

[121] LIU X, WEN Z, ZHANG Y. An efficient Gauss–Newton algorithm for symmetric low-rank product matrix approximations[J]. SIAM Journal on Optimization, 2015, 25(3):1571-1608.

[122] SORENSEN D C. Numerical methods for large eigenvalue problems[J]. Acta Numerica, 2002, 11:519-584.

[123] SAAD Y. Numerical methods for large eigenvalue problems: revised edition: volume 66[M]. Siam, 2011.

[124] GOLDSTEIN T, OSHER S. The split Bregman method for L1-regularized problems[J]. SIAM journal on imaging sciences, 2009, 2(2):323-343.

[125] CHEN W, JI H, YOU Y. An augmented Lagrangian method for ℓ_1-regularized optimization problems with orthogonality constraints[J]. SIAM Journal on Scientific Computing, 2016, 38 (4):B570-B592.

[126] BOLTE J, SABACH S, TEBOULLE M. Proximal alternating linearized minimization or nonconvex and nonsmooth problems[J]. Mathematical Programming, 2014, 146(1-2):459-494.

[127] LI Y, WEN Z, YANG C, YUAN Y. A semi-smooth Newton method for solving semidefinite programs in electronic structure calculations[J]. arXiv:1708.08048, 2017.

[128] YUAN H, GU X, LAI R, WEN Z. Global optimization with orthogonality constraints via stochastic diffusion on manifold[J]. arXiv preprint arXiv:1707.02126, 2017.

[129] BOUMAL N, MISHRA B, ABSIL P A, SEPULCHRE R. Manopt, a Matlab toolbox for optimization on manifolds[J]. The Journal of Machine Learning Research, 2014, 15(1):1455-1459.

[130] HUANG W, ABSIL P A, GALLIVAN K A, HAND P. ROPTLIB: an object-oriented C++ library for optimization on Riemannian manifolds[J]. ACM Transactions on Mathematical Software (TOMS), 2018, 44(4):43.

[131] BAKER C G, ABSIL P A, GALLIVAN K A. An implicit trust-region method on Riemannian manifolds[J]. IMA journal of numerical analysis, 2008, 28(4):665-689.

[132] MEGHWANSHI M, JAWANPURIA P, KUNCHUKUTTAN A, KASAI H, MISHRA B. McTorch, a manifold optimization library for deep learning[R]. arXiv preprint arXiv:1810.01811, 2018.

[133] TREFETHEN L N, BAU III D. Numerical linear algebra: volume 50[M]. Siam, 1997.

[134] DOLAN E D, MORÉ J J. Benchmarking optimization software with performance profiles[J]. Mathematical programming, 2002, 91(2):201-213.

[135] YUAN Y X. A review on subspace methods for nonlinear optimization[C]//Proceedings of the International Congress of Mathematics. 2014: 807-827.

[136] POWELL M J. A method for nonlinear constraints in minimization problems[J]. Optimization, 1969:283-298.

[137] BERTSEKAS D P. Constrained optimization and Lagrange multiplier methods[M]. Academic press, 2014.

[138] DIRAC P A M. Quantum mechanics of many-electron systems[J]. Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character, 1929, 123(792):714-733.

[139] 刘壮. 第一原理电子结构计算的优化算法若干研究[D]. 博士学位论文. 北京: 中国科学院大学, 2017.

[140] YANG C, GAO W, MEZA J C. On the convergence of the self-consistent field iteration for a class of nonlinear eigenvalue problems[J]. SIAM Journal on Matrix Analysis and Applications, 2009, 30(4):1773-1788.

[141] WEN Z, MILZAREK A, ULBRICH M, ZHANG H. Adaptive regularized self-consistent field iteration with exact Z for electronic structure calculation[J]. SIAM Journal on Scientific Computing, 2013, 35(3):A1299-A1324.

[142] ULBRICH M, WEN Z, YANG C, KLOCKNER D, LU Z. A proximal gradient method for ensemble density functional theory[J]. SIAM Journal on Scientific Computing, 2015, 37(4): A1975-A2002.

[143] 姜波. 若干特殊优化问题以及应用[D]. 博士学位论文. 北京: 中国科学院大学, 2013.

[144] PERDEW J P, ZUNGER A. Self-interaction correction to density-functional approximations for many-electron systems[J]. Physical Review B, 1981, 23(10):5048.

作者简历及攻读学位期间发表的学术论文与研究成果

作者简历

高斌, 男, 甘肃省兰州市人, 出生于 1991 年 10 月.

2010 年 9 月至 2014 年 6 月, 就读于四川大学数学学院, 基础数学专业, 获理学学士学位.

2014 年 9 月至 2019 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, 导师为袁亚湘院士.

Email: gaobin@lsec.cc.ac.cn 个人主页: https://www.gaobin.cc/

已发表 (或正式接受) 的学术论文:

[1] Bin Gao, Xin Liu and Ya-xiang Yuan, *Parallelizable Algorithms for Optimization Problems with Orthogonality Constraints*, SIAM Journal on Scientific Computing, accepted.

[2] Bin Gao, Xin Liu, Xiaojun Chen and Ya-xiang Yuan, *A New First-order Algorithmic Framework for Optimization Problems with Orthogonality Constraints*, SIAM Journal on Optimization, 28-1(2018), 302–332.

[3] 高斌, 刘歆, 袁亚湘. 正交约束优化问题的一阶算法. 运筹学学报, 21-4(2017), 57-68.

已完成的学术论文:

[1] Bin Gao, Xin Liu, Xiaojun Chen and Ya-xiang Yuan, *On the Łojasiewicz Exponent of the Quadratic Sphere Constrained Optimization Problem*, arXiv:1611.08781.

国际学术报告:

[1] Parallelizable Approaches for Optimization Problems with Orthogonality Constraints, *The 23nd International Symposium on Mathematical Programming*, Bordeaux, France, July 5, 2018, Cluster Talk.

[2] Parallelizable Approaches for Optimization Problems with Orthogonality Constraints, *The 11th International Conference on Numerical Optimization and Numerical Lin-*

ear Algebra, Yinchuan, China, August 9, 2017, Contributed Talk.

[3] Column-wise BCD Method for Orthogonal Constrained Optimization Problems. *The 11th East Asia SIAM Conference*, Macao SAR, China, June 20, 2016, Contributed Talk.

参加的研究项目及获奖情况:

2018 年　中国工业与应用数学学会第 16 届年会优秀学生论文奖

2018 年　中国科学院院长特别奖

2017 年　国家奖学金 (博士)

2016 年　International Workshop on Modern Optimizaion and Application “Honor Student Award”

2016 年　第二届中国运筹学会数学规划分会研究生论坛 “优秀成果奖”

2016 年　中国科学院数学与系统科学研究院 “三好学生”

致　谢

玉渊潭的樱花又一次盛开, 我的博士生涯也即将结束. 回顾五年来的点点滴滴, 心中不免百感交集. 笔落至此, 除了感激和不舍, 再没有其他话语.

感谢恩师, 我最敬爱的袁老师. 第一次走进袁老师办公室的场景仿佛昨日, 袁老师的和蔼可亲与平易近人给我留下了深刻的印象. 在科研上, 袁老师给予我充分的自由, 每次和袁老师讨论后, 我总是茅塞顿开. 我最崇拜的是袁老师总能看清问题的本质, 并且能给出最直观、最形象的解释. 耳濡目染下, 我也学着袁老师严谨、多角度地思考问题. 同时, 我要感谢袁老师给我很多外出参加学术会议的机会, 让我开阔了视野, 结识了朋友. 在生活上, 袁老师就是我的偶像, 为人大度风趣且爱好丰富, 对学生的关心和照顾就像对自己儿女一般. 我总是希望自己能像袁老师一样, 在科研和生活上永不止步、永不服输. 在人生的十字路口, 袁老师像父亲般指引我, 鼓励我, 给我建议, 帮助我走出迷茫, 最终坚定我的信念. 我想说, 成为袁老师的学生是我这辈子最幸福和最幸运的事. 感谢袁老师照亮了我的人生, 以后的路, 我还要更加努力.

感谢兄长, 我最亲爱的刘歆师兄. 作为兄长, 师兄亲力亲为、事无巨细地在科研上帮助我, 小到语句格式、代码编写, 大到单独为我讲一遍报告. 每当我科研遇到瓶颈的时候, 聪明的师兄总能提出新的思路. 我还记得那些深夜共同奋战在办公室的日子, 那是我博士期间最难忘的时光. 作为朋友, 师兄和我无话不谈. 一路走来, 有过欢笑, 有过泪水. 在我稍有松懈的时候, 师兄严厉地批评了我, 那句 “逆水行舟, 不进则退” 让我刻骨铭心, 师兄的学术态度永远是我学习的榜样. 此外, 还要感谢师兄对我生活上的照顾, 让我学也快乐, 玩也开心.

感谢香港理工大学的陈小君老师在论文合作期间对我的指导和帮助. 感谢课题组的戴彧虹老师, 戴老师在讨论班上的建议让我受益良多. 同时, 戴老师对于生活的热爱和对他人的公益慈善之心也值得我学习. 感谢课题组的刘亚锋师兄, 亚锋师兄总能给予师弟师妹们朋友般的关心和照顾, 讨论起学术来也是热情洋溢、一丝不苟. 感谢北京邮电大学的孙聪师姐, 每次和师姐一同外出开会, 我总能学到如何最优化行程, 同时也要感谢师姐对我科研道路的关心和鼓励. 感谢已在外工作曾给予我帮助的师兄师姐们: 王彦飞, 文再文, 夏勇, 马士谦, 张在坤, 吴乐秦, 姜波等师兄, 范金燕, 徐玲玲, 牛凌峰, 王晓等师姐.

感谢课题组同期的师兄弟姐妹们: 感谢盛镇醴, 刘田香, 张睿燕, 董乾, 王树

雄, 崔春风, 顾然, 康冬, 董志龙, 陈诚, 曾燎原等师兄师姐们对我的关心与照顾. 还要感谢王小玉, 陈亮, 傅凯, 金哲, 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, 姜博鸥, 黄磊, 陈圣杰, 张吾帅君, 王磊, 汪思维等师弟师妹们在课题组的朝夕相伴以及博士后李志保, 黄亚魁, 贲树军, 黄娜, 张婷, 张国涵对我的帮助. 特别感谢黄亚魁, 周睿智, 肖纳川, 赵浩天审阅我的论文初稿并提出了许多宝贵的建议. 感谢一同入学的赵亮, 周睿智和陈伟坤, 有你们的相伴科研生活才能如此快乐. 此外, 我还要感谢数学院其他给予我关心和帮助的同学们, 是你们的陪伴让我不再孤单.

感谢刘颖, 吴继萍, 丁如娟, 张继平, 钱莹, 魏敏, 尹永华, 刘霞等办公室老师们的帮助, 因为你们, 我的学习生活顺利而又多姿多彩.

感谢我的家人, 一路走来, 我的父母给予我世上最无私的爱, 你们尊重我的每一个决定并且毫无保留地支持我. 养育之恩无以为报, 希望我的努力能够让你们自豪.

感谢陪伴我整个研究生阶段的梦伊, 愿未来的日子里, 我们可以继续牵手看想看的世界.

最后的最后, 感谢我爱的人和爱我的人. 新的远航才刚刚开始: “朝闻道, 夕死可矣”.

于北京保福寺