

Newton Spectral Gradient Method of Unconstrained Optimization Problems

Tingxia Song, Zhensheng Yu

College of Science, University of Shanghai for Science and Technology, Shanghai
Email: 1273766703@qq.com, zhsh-yu@163.com

Received: Jan. 24th, 2020; accepted: Feb. 7th, 2020; published: Feb. 14th, 2020

Abstract

Based on the Newton method and the combination of the steepest descent method, Newton method and spectral gradient method are combined to solve unconstrained optimization problems. This method is effectively applied to the problem that the Hessian matrix of the objective function is not positive definite or the initial point is not close to the minimum point, and the non-monotone line search is used to solve the step size. The global convergence and superlinear convergence of this method are established under relatively mild conditions, and the numerical experiments show that the algorithm has good numerical experimental results.

Keywords

Newton Direction, Spectral Gradient Direction, Non-Monotone, Convex Combination

无约束优化问题的牛顿谱梯度法

宋婷霞, 宇振盛

上海理工大学理学院, 上海
Email: 1273766703@qq.com, zhsh-yu@163.com

收稿日期: 2020年1月24日; 录用日期: 2020年2月7日; 发布日期: 2020年2月14日

摘要

本文在Newton法和最速下降法的组合方法的基础上提出求解无约束优化问题的Newton法与谱梯度法的组合方法, 该方法有效地应用于目标函数的Hessian矩阵不正定或初始点不接近极小点的问题, 并利用非单调线搜索求解步长。在较温和的条件下建立了该方法的全局收敛性和超线性收敛性, 并且数值实

验证了该算法具有很好的数值实验效果。

关键词

牛顿方向, 谱梯度方向, 非单调, 凸组合

Copyright © 2020 by author(s) and Hans Publishers Inc.

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

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



Open Access

1. 引言

本文主要用于研究如下非线性无约束最优化问题

$$\min_{x \in \mathbb{R}^n} f(x) \quad (1.1)$$

其中 $f: \mathbb{R}^n \rightarrow \mathbb{R}$ 是连续可微函数。本文的方法是通过更新迭代式

$$x_{k+1} = x_k + \alpha_k d_k \quad (1.2)$$

来解决问题(1.1), d_k 是搜索方向, $\alpha_k > 0$ 是步长。

牛顿法是解无约束优化最经典也是最早的算法之一, 作为最基本的求解无约束优化问题的方法, 牛顿法及其各种改进方法一直受到广泛关注(见文献[1] [2] [3] [4])。下面是牛顿法的原理:

假设 x_k 是当前迭代点, 计算 x_k 处目标函数的梯度 $g_k = \nabla f(x_k)$ 和 Hessian 矩阵 $G_k = \nabla^2 f(x_k)$ 。若 $g_k = \nabla f(x_k) \neq 0$, 则根据目标函数的二次可微性知, 目标函数在 x_k 附近是严格凸函数。基于目标函数的二次泰勒展式, 在当前迭代点 x_k 附近:

$$f(x_k + d) \approx f(x_k) + d^T \nabla f(x_k) + \frac{1}{2} d^T \nabla^2 f(x_k) d \quad (1.3)$$

所以, 考虑

$$\min_{d \in \mathbb{R}^n} \varphi_k(d) = d^T \nabla f(x_k) + \frac{1}{2} d^T \nabla^2 f(x_k) d \quad (1.4)$$

假定 $\nabla^2 f(x_k)$ 是正定的, 则知该式的解是

$$d_k = -[\nabla^2 f(x_k)]^{-1} \nabla f(x_k) \quad (1.5)$$

上式就是牛顿搜索方向。牛顿方向 d_k 被附加到(1.2)的搜索方向而不是直接替代主要基于以下几个考虑: 首先(1.1)对于[5]中的问题, 函数 $f: \mathbb{R}^n \rightarrow \mathbb{R}$ 的 Hess 矩阵可能不正定, 即使我们假设 f 在 \mathbb{R}^n 上连续可微, 也可能初始点的选取没有足够靠近“极小点”, 牛顿法的一些理想性质不在初始点附近, 其次通过结合谱梯度法, 能够避免求得的(1.1)式的解被困在局部最小值中。

BB 步是由 Barzilai 和 Barwein 在 1988 年提出的两点步长梯度法[6], 其基本思想是利用迭代当前点以及前一点的信息来确定步长因子。1993 年和 2002 年 Raydan 和 Dai & Liao 证明了广义 n 维严格凸二次型 BB 方法的全局收敛性和 R-线性收敛性。随后, BB 方法也扩展以解决一般的非线性优化问题。通过将 Grippo 等提出的非单调线搜索[7] [8] [9]结合起来, 1997 年 Marcos Raydan 在文献[10]中提出了一种加入全局策略的 GBB 算法, 2018 年 Yu-Hong Dai 在文章[11]中, 提出了一族谱梯度法, 并证明了此类算法在

任意维情况下都具有 R 阶超线性收敛性。下面是该方法的原理:

他们是把迭代公式 $x_{k+1} = x_k + \lambda_k d_k$ 看成是

$$x_{k+1} = x_k - D_k g_k \quad (1.6)$$

其中 $g_k = g(x_k) = \nabla f(x_k)$, $D_k = \lambda_k I$. 为了使矩阵 D_k 具有“拟牛顿”性质, 计算 λ_k 使得

$$\min \|s_{k-1} - D_k y_{k-1}\|_2 \quad (1.7)$$

或者

$$\min \|D_k^{-1} s_{k-1} - y_{k-1}\|_2 \quad (1.8)$$

其中 $s_{k-1} = x_k - x_{k-1}$, $y_{k-1} = g_k - g_{k-1}$, 由上两式分别求得

$$\lambda_1 = s_{k-1}^T y_{k-1} / \|y_{k-1}\|_2^2 \quad (1.9)$$

和

$$\lambda_2 = \|s_{k-1}\|_2^2 / s_{k-1}^T y_{k-1} \quad (1.10)$$

上面两式就是谱梯度步长。在一般的非二次情况下, λ_1 , λ_2 可以是令人无法接受的大或小(甚至对于非凸函数是负的), 因此我们必须假设逐步计算出的 λ 是通过(1.10)或(1.11)进行修改, 以满足形式的一个条件

$$0 < \min\left(\eta, \frac{1}{\tau}\right) \leq \lambda_k \leq \max\left(\frac{1}{\eta}, \tau\right) \quad (1.11)$$

若不然, 则令 $\lambda = \tau$ 。

现在我们给出一个简短的分析, 形成了一个针对问题(1.1)的牛顿谱梯度法。对任意的 $x \in \mathfrak{R}^n$ 当 $g_k = \nabla f(x_k) \neq 0$ 时, 设 $G_k = \nabla^2 f(x_k)$ 是目标函数的 Hessian 矩阵。由于目标函数 f 是连续可微的, 所以如果下列方程

$$G_k d + g_k = 0 \quad (1.12)$$

有解 d_N , 则 d_N 在 x_k 处的方向为下降方向, 并称为 f 在 x_k 的牛顿方向。且由论文[12]得, 序列 $\{x_k\}$ 的任何极限点 x^* 都是 f 的稳定点, 既 $g(x^*) = 0$ 。

谱梯度法因其几乎对所有初始值都是成立的而得到广泛的应用。从而弥补了牛顿法对初始点的苛刻要求, 并且该方法不需要求解目标函数 f 的 Hessian 矩阵。因此有如下谱梯度方向

$$d_B = -\lambda g_k \quad (1.13)$$

在本文中, 我们结合(1.6)和(1.12)引入牛顿方向和谱梯度方向的凸组合来最大化利用这两种方法的优势。对于任给的 $x \in \mathfrak{R}^n$ 设

$$d_k = t d_N + (1-t) d_B \quad (1.14)$$

其中谱梯度方向 d_B 有一个缺点是该方向不总是下降的。为了克服这一缺点, 本文结合 Grippo 等提出的非单调线搜索, 从而避免了谱梯度法不是单调下降的问题。

$$f(x_k + \beta^{m_k} d_k) \leq \max_{0 \leq j \leq M} f(x_{k-j}) + \sigma \beta^{m_k} g_k^T d_k \quad (1.15)$$

其中步长 $\alpha_k = \beta^{m_k}$ 。

文章剩下的部分组织如下: 第二节给出了搜索方向的性质并描述了具体的牛顿谱梯度方法; 第三节

证明了算法的收敛性; 第四节给出了数值实验, 第五节进行了总结。

2. 搜索方向的性质及牛顿谱梯度法

算法 2.1 (牛顿谱梯度算法)

步 0 给定初始点 $x_0 \in R^n$, 终止误差 $0 \leq \varepsilon \ll 1$ 。令 $k := 0$

步 1 计算 $g_k = \nabla f(x_k)$ 。若 $\|g_k\| \leq \varepsilon$, 停算, 输出 x_k 作为近似极小点。

步 2 由(1.10)或(1.11)给出满足条件(1.12)的

$$d_B = -\lambda g_k.$$

步 3 计算 $G_k = \nabla^2 f(x_k)$, 解方程组

$$G_k d + g_k = 0 \tag{2.1}$$

若(2.1)有解 d_N , 且满足 $g_k^T d_N < 0$, 转步 3; 否则, 令 $d_k = d_B$ 转步 5。

步 4 $d_k = t d_N + (1-t) d_B$ 。

步 5 由非单调线搜索技术确定步长因子 $\alpha_k = \beta^{m_k}$ 。(m_k 是满足条件的最小非负整数)

$$f(x_k + \beta^{m_k} d_k) \leq \max_{0 \leq j \leq \min(k, M)} f(x_{k-j}) + \sigma \beta^{m_k} g_k^T d_k \tag{2.2}$$

步 6 令 $x_{k+1} = x_k + \alpha_k d_k$, $k := k + 1$, 转步 1。

首先, 我们将研究 d_k 的一些性质, 且我们假定目标函数 f 在 R^n 上总是二次连续可微的, 且满足下列假设 1:

假设 1 目标函数 $f: R^n \rightarrow R$ 二次连续可微, 对任意给定 $x_0 \in R^n$, f 在水平集

$$L(x_0) = \{x \mid f(x) \leq f(x_0)\} \tag{2.3}$$

上有界的。

假设 2 存在 x^* 的一个邻域 $N(x^*, \delta_1)$, 使得 $g(x)$ 在该邻域内 Lipschitz 连续, 既存在一个常数 $0 < K < \frac{1}{3\lambda}$

使得

$$\|g(x) - g(x^*)\| \leq K \|x - x^*\|, \forall x \in N(x^*, \delta_1)$$

假设 3 存在 x^* 的一个邻域 $N(x^*, \delta)$, 使得 $G(x)$ 在该邻域内 Lipschitz 连续, 既存在一个常数 $L > 0$, 使得

$$\|G(x) - G(x^*)\| \leq L \|x - x^*\|, \forall x \in N(x^*, \delta)$$

为了证明收敛性定理, 首先给出如下性质: 下列性质证明该算法的方向是下降方向。

定理 2.1 假设 $x \in R^n$ 不是(1.1)的稳定点, 由式(2.1)计算得到的 d_N 存在且 $\sigma \in (0, 1)$ 。则存在 α_k 使得, d_k 是 f 在 x 处的下降方向且满足(2.2)式。

证明: 因为 f 是连续可微的, 对于给定的 $t \in [0, 1]$

$$d_k = t d_N + (1-t) d_B$$

通过(1.12)和(1.6)有

$$\begin{aligned} \nabla f(x)^T d_k &= \nabla f(x)^T (t d_N + (1-t) d_B) \\ &= t \nabla f(x)^T \cdot (-G_k^{-1} g_k) - (1-t) \nabla f(x)^T \lambda g_k \\ &= -t g_k^T G_k^{-1} g_k - \lambda (1-t) g_k^T g_k \end{aligned}$$

根据正定矩阵的逆矩阵仍然是正定矩阵的性质, 显然有

$$\nabla f(x)^T d_k < 0. \quad (2.4)$$

对任给的 $0 < \alpha_k \leq 1$, 有

$$f(x + \alpha_k d_k) = f(x) + \nabla f(x)^T d_k + o(\alpha_k) \quad (2.5)$$

且对任给的 k 都有

$$f(x_k + \alpha_k d_k) \leq \max_{0 \leq j \leq M} f(x_{k-j})$$

故有

$$f(x_k + \alpha_k d_k) \leq \max_{0 \leq j \leq M} f(x_{k-j}) + \nabla f(x)^T d_k + o(\alpha_k)$$

结合(2.4), 对任给的 $\sigma \in (0, 1)$, (2.2)成立。

目前为止, 我们已经考虑了非稳定点 x_k 处的搜索方向 d_k 的一些全局性质, 为了证明我们算法的超线性收敛性, 我们接下来考虑解点周围的方向的性质。

定理 2.2 假设 x^* 是 $g(x) = 0$ 的解, 存在 $(0, 1)$ 一个常数 κ , 对任给的 $\alpha_k \in (0, 1]$,

$$0 < \min\left(\eta, \frac{1}{\tau}\right) \leq \lambda_k \leq \max\left(\frac{1}{\eta}, \frac{1}{\tau}\right)$$

且对于 $x_k \in \mathfrak{R}^n$, $x_k \rightarrow x^*$, 有

$$\nabla f(x_k)^T d_k = -2\alpha_k f(x_k) + \alpha_k o(f(x_k))$$

证明: 详见参考文献[13]定理 3.2。

3. 收敛性分析

针对我们的牛顿谱梯度法, d_N 是通过求解线性系统(2.1)得到, 如果(2.1)是不可解的或者矩阵 G_k 是高度病态的, 则 $d_N = -\lambda g_k$, 这意味着 $d_k = -\lambda g_k$ 。其次, 通过定理 2.1, 如果 x_k 不是(1.1)的稳定点, 且算法 2.1 的步骤 5 有很好的定义, 则我们的算法要么停止在一个稳定点, 要么生成一个无穷序列 $\{x_k\}$ 。

定理 3.1 设 $\{x_k\}$ 是由算法 2.1 生成的序列, 且满足假设 1, 则下列性质成立:

- (a) 序列 $\{x_k\}$ 的任何极限点 x^* 都是 f 的稳定点, 既满足 $g(x^*) = 0$;
- (b) $\{x_k\}$ 的任何极限点 x^* 都不是 f 的极大值点;
- (c) 如果 $\{x_k\}$ 只有有限个稳定点, 则序列 $\{x_k\}$ 本身收敛。

证明: 为了证明性质(a), 我们结合定理 2.1 使用了([7], p709)中的收敛性证明部分。

我们定义 $m(k) = \min(k, M)$, 显然 $m(0) = 0$, $0 \leq m(k) \leq \min(m(k-1) + 1, M)$ 对所有的 $k \geq 1$, 另外, 对任给的 k 有 $\alpha_k \in (0, 1]$, 最后存在正数 c_1 和 c_2 使得搜索方向 d_k 满足

$$g_k^T d_k \leq -c_1 \|g_k\|^2, \quad \|d_k\| \leq c_2 \|g_k\|$$

实际上, 在牛顿谱梯度法中 $d_k = t d_N + (1-t) d_B$, 对于所有的 $k \geq 1$

$$\begin{aligned} \nabla f(x)^T d_k &= \nabla f(x)^T (t d_N + (1-t) d_B) \\ &= t \nabla f(x)^T \cdot (-G_k^{-1} g_k) - (1-t) \nabla f(x)^T \lambda g_k \\ &= -t g_k^T G_k^{-1} g_k - \lambda (1-t) g_k^T g_k \end{aligned}$$

因为 G_k 正定, 故 G_k^{-1} 正定, 即对一切 $0 \neq h \in \mathbb{R}^n$ 有 $h^T G_k^{-1} h > 0$, 而 $g(x_k) \neq 0$, 所以存在一个正数 p , 使得 $g_k^T G_k^{-1} g_k > p \|g_k\|^2$, 从而有 $g_k^T d_k \leq (tp - \lambda(1-t)) \|g_k\|^2$, 另外,

$$\begin{aligned} \|d_k\| &= \|td_N + (1-t)d_B\| \\ &= \|-tG_k^{-1}g_k - (1-t)\lambda g_k\| \\ &\leq \|-tG_k^{-1}g_k\| + \|(1-t)\lambda g_k\| \end{aligned}$$

由 G_k 的正定性及 f 二次连续可微可知, 存在 x^* 的一个领域 $U(x^*)$, 使得对任意的 $x \in U(x^*)$, 都有 $G(x)$ 是一致正定的。特别的, $\|G(x)^{-1}\|$ 在上有限, 既存在常数 $C > 0$, 使得 $\|G(x)^{-1}\| \leq C$, 故而 $\|d_k\| \leq (tC + (1-t)\lambda) \|g_k\|$ 。

对任给的 $t \in [0, 1]$, $\lambda \in (0, 1]$, 令 $c_1 = tp - \lambda(1-t)$, $c_2 = tC + \lambda(1-t)$ 。

因此, 根据([9], p711)中的等式(14), 我们得到

$$\lim_{k \rightarrow \infty} \alpha_k \|g_k\| = 0$$

因为 $\alpha_k = \beta^{m_k} > 0$, (a)式得证。

(b)和(c)的证明直接由[9]中的收敛性定理给出。

定理 3.2 设函数 $f(x)$ 二次连续可微, $\{x_k\}$ 由算法 2.1 产生的迭代序列, 且满足假设 2, 设 $\{x_k\}$ 聚点 x^* 是 f 的解。 $G(x^*)$ 正定, 则 $\{x_k\}$ 超线性收敛于 f 的全局极小点 x^* 。

证明: 由 $G(x^*)$ 的正定性及 f 二次连续可微可知, 存在 x^* 的一个领域 $U(x^*)$, 使得对任意的 $x \in U(x^*)$, 都有 $G(x)$ 是一致正定的。特别的, $\|G(x)^{-1}\|$ 在上有限, 既存在常数 $C > 0$, 使得 $\|G(x)^{-1}\| \leq C$ 。又由 $G(x)$ 的连续性知, 存在邻域 $U(x^*)$, 使得

$$\|G(x) - G(x^*)\| \leq \frac{1}{4C}, \forall x \in U(x^*) \subseteq U_1(x^*)$$

因此, 当 $x_k \in U(x^*)$ 时, 有

$$\begin{aligned} \|x_{k+1} - x^*\| &= \|x_k + \alpha_k (-tG_k^{-1}g_k - (1-t)\lambda g_k) - x^*\| \\ &= \|G_k^{-1} \cdot \|\alpha_k t(g_k - g^*) - G_k(x_k - x^*) + \alpha_k(1-t)\lambda G_k(g_k - g^*)\| \\ &\leq \|G_k^{-1}\| \cdot \|\alpha_k t(g_k - g^*) - G_k(x_k - x^*)\| + \|\alpha_k(1-t)\lambda(g_k - g^*)\| \\ &\leq C \left\| \int_0^1 G(x^* + \mu(x_k - x^*)) (x_k - x^*) d\mu - G_k(x_k - x^*) \right\| + K\lambda \|x_k - x^*\| \\ &\leq C \int_0^1 \|G(x^* + \mu(x_k - x^*)) - G_k\| \cdot \|x_k - x^*\| d\mu + K\lambda \|x_k - x^*\| \\ &\leq C \left(\int_0^1 \|G(x^* + \mu(x_k - x^*)) - G_k\| d\mu \right. \\ &\quad \left. + \int_0^1 \|G_k - G(x^*)\| \|x_k - x^*\| d\mu \right) + K\lambda \|x_k - x^*\| \\ &\leq \frac{1}{2} \|x_k - x^*\| + K\lambda \|x_k - x^*\| \\ &= \left(\frac{1}{2} + K\lambda \right) \|x_k - x^*\| \\ &\leq \frac{5}{6} \|x_k - x^*\| \end{aligned}$$

从而有 $\|x_{k+1} - x^*\| = o\|x_k - x^*\|$, 既 $\{x_k\}$ 超线性收敛于 x^* 。

4. 数值实验

本文从文献[5]中取出 38 个标准测试问题进行数值实验, 目的是比较本文算法与几种经典无约束优化算法的数值效果以及对本文不同 M 值的算法的试验效果。所有数值实验都是在 WINDOWS10 操作系统, CPU 频率为 2.0 GHz, 内存为 2 GB 的个人计算机上进行。各参数选取如下: $\varepsilon = 10^{-8}$, $\sigma = 0.4$, $\rho = 0.55$, $\eta = 10^{-10}$ 。其中如果步 2 中 λ_k 不满足(1.12), 则参数 τ 由下列式子给出:

$$\tau = \begin{cases} 1, & \|g_k\|_2 > 1 \\ \|g_k\|_2^{-1}, & 10^{-5} \leq \|g_k\|_2 \leq 1 \\ 10^5, & \|g_k\|_2 < 10^{-5} \end{cases} \quad (4.1)$$

表 1 中列出了各计算结果, 其中各列的代表含义如下: k : 迭代次数; flag : 函数值的计算次数; f^* : 算法求得的最优值。文中的初始点 x_0 采用文献[5]中给定的用来试验函数正确性的点作为初始点; 终止条件采用 $\|g_k\| \leq 10^{-8}$ 。

表 1 选取 $M = 1, M = 2, M = 5, M = 10$ 对本文算法进行比较, 计算出不同的 M 值函数的迭代次数 k , 函数值的计算次数 flag 和函数值 f^* , 并进行了比较。凸组合参数 t 的选取是通过比较 $t = 0.4, 0.5, 0.55, 0.6, 0.618$ 这几个参数选取的, 通过比较发现当 $t = 0.618$ 时, 表现效果较佳, 因此选择 $t = 0.618$ 。其次从表中可以看出当 $M = 10$ 时, 该算法在最优函数值, 函数值的计算次数以及迭代次数上对更多函数都是好的, 因此选用 $M = 10$ 时的函数计算效果与其他方法进行比较。表 2 中主要把共轭梯度法, BFGS 拟牛顿法同本文中的算法进行比较。

Table 1. Different algorithms with the comparison of this algorithm

表 1. 不同算法同本文算法的比较

问题	nmsg 方法 M1			nmsg 方法 M2			nmsg 方法 M5			nmsg 方法 M10		
	flag	f^*	k	flag	f^*	k	flag	f^*	k	flag	f^*	k
fun	170	9.30E-22	52	64	7.68E-20	52	15	3.64E-18	54	14	1.74E-26	56
fun202	64	4.90E+01	22	29	1.04E-20	21	12	3.98E-21	28	16	2.24E-25	32
fun204	1	1.84E-01	9	0	1.84E-01	8	0	1.84E-01	8	0	1.84E-01	8
fun208	170	9.30E-22	52	64	7.71E-20	52	15	3.64E-18	54	14	1.74E-26	56
fun243	8	7.97E-01	13	4	7.97E-01	12	4	7.97E-01	12	4	7.97E-01	12
fun256	902	1.71E-14	221	302	1.13E-18	83	44	2.57E-19	59	18	2.18E-20	67
fun266	6	1.00E+00	14	4	1	13	3	1	13	3	1.00E+00	13
fun271	0	9.97E-21	10	0	9.97E-21	10	0	9.97E-21	10	0	9.97E-21	10
fun273	0	8.17E-20	15	0	8.17E-20	15	0	8.17E-20	15	0	8.17E-20	15
fun275	7	2.94E-15	10	5	3.54E-16	11	0	1.89E-18	11	0	1.89E-18	11
fun276	13	4.73E-16	13	3	6.78E-16	12	0	7.02E-19	12	0	7.02E-19	12
fun286	170	9.30E-21	52	64	1.02E-19	53	15	3.64E-17	54	14	1.74E-25	56
fun288	582	4.62E-15	139	399	4.06E-18	105	44	1.28E-18	59	18	1.09E-19	67
fun290	5	1.50E-12	21	5	1.50E-12	21	5	1.50E-12	21	5	1.50E-12	21
fun291	12	1.62E-12	25	12	1.62E-12	25	12	1.62E-12	25	12	1.62E-12	25
fun292	17	8.01E-13	28	17	8.01E-13	28	17	8.01E-13	28	17	8.01E-13	28
fun293	20	6.32E-13	30	20	6.32E-13	30	20	6.32E-13	30	20	6.32E-13	30

Continued

fun294	3	5.64E+00	500	60	1.67E-20	63	47	3.30E-20	59	9	3.85E-20	54
fun295	14	9.59E+00	500	20	9.52E+00	500	78	3.40E-20	91	28	2.73E-20	79
fun296	369	4.75E-18	139	12	1.55E+01	500	113	2.28E-20	137	47	6.19E-21	107
fun298	599	1.10E-17	221	355	1.11E-20	296	178	4.87E-20	211	165	8.19E-22	241
fun300	273	-20	64	18	-20	27	4	-20	24	4	-2.00E+01	24
fun301	265	-50	53	22	-50	30	9	-50	26	9	-50	26
fun302	421	-100	68	31	-100	33	18	-100	30	16	-1.00E+02	27
fun308	9	7.73E-01	15	6	7.73E-01	14	6	7.73E-01	14	6	7.73E-01	14
fun314	8	1.69E-01	8	5	1.69E-01	9	5	1.69E-01	9	5	1.69E-01	9
fun352	28	9.03E+02	27	15	9.03E+02	21	8	9.03E+02	18	8	9.03E+02	18
fun379	192	4.01E-02	125	83	4.01E-02	93	34	4.01E-02	157	29	4.01E-02	51

Table 2. Comparing different M value algorithm
表 2. 不同 M 取值的算法比较

问题	bfgs			cg			$M = 10$		
	flag	f^*	k	flag	f^*	k	flag	f^*	k
fun	28	7.25E-23	33	2139	2.82E-17	224	14	1.74E-26	56
fun202	22	4.90E+01	14	50883	4.90E+01	5000	16	2.24E-25	32
fun204	8	1.84E-01	3	62	1.84E-01	16	0	1.84E-01	8
fun208	28	7.25E-23	33	1791	1.15E-16	191	14	1.74E-26	56
fun243	12	7.97E-01	9	657	7.97E-01	136	4	7.97E-01	12
fun256	32	1.08E-12	42	33923	4.49E-11	5000	18	2.18E-20	67
fun266	19	1	13	305	1	84	3	1.00E+00	13
fun271	57	3.08E-30	11	160	1.68E-19	17	0	9.97E-21	10
fun273	95	1.81E-19	39	207	2.30E-20	19	0	8.17E-20	15
fun275	2	5.61E-16	26	26	2.55E-13	2807	0	1.89E-18	11
fun276	2	4.15E-13	23	16	4.36E-13	399	0	7.02E-19	12
fun286	213	3.78E-18	154	828	5.07E-17	84	14	1.74E-25	56
fun288	96	6.13E-13	198	2485	1.83E-12	342	18	1.09E-19	67
fun290	8	1.70E-12	39	12	1.56E-10	5000	5	1.50E-12	21
fun291	102	8.40E-13	76	51	1.76E-11	5000	12	1.62E-12	25
fun292	430	5.70E-13	166	117	3.39E-12	4358	17	8.01E-13	28
fun293	3	NAN	5000	0	NAN	5000	20	6.32E-13	30
fun294	86	3.97E+00	45	5861	5.85E-17	561	9	3.85E-20	54
fun295	141	3.99E+00	74	2808	6.00E-17	255	28	2.73E-20	79
fun296	222	3.99E+00	103	1489	NAN	5000	47	6.19E-21	107
fun298	597	4.45E-21	249	14285	4.68E-18	1010	165	8.19E-22	241
fun300	20	-20	25	3264	-20	894	4	-2.00E+01	24
fun301	52	-50	57	11513	-5.00E+01	3134	9	-50	26
fun302	100	-100	107	18129	-1.00E+02	5000	16	-1.00E+02	27
fun308	11	7.73E-01	13	54	7.73E-01	17	6	7.73E-01	14
fun314	9	1.69E-01	6	729	1.69E-01	128	5	1.69E-01	9
fun352	23	9.03E+02	9	37703	9.03E+02	5000	8	9.03E+02	18
fun379	43	4.01E-02	56	9501	4.01E-02	1902	29	4.01E-02	51

从表 2 中能明显地看出, 本文的算法在很大一部分函数中都有很好的计算效果, 在 27 个函数中, 其中对最优函数值表现好的有 20 个, 对迭代次数表现好的有 14 个, 而从函数值的计算次数来看, 几乎处处表现良好。因此我们得到了一个相对比较好的算法。

5. 总结

本文通过介绍牛顿法与谱梯度法的凸组合搜索方向提出一种求解无约束优化问题的牛顿 - 谱梯度法, 该算法实现了好的数值实验结果, 即使在牛顿方向奇异的情况。我们在数值试验过程中还发现该方法在某些情况下依然会带有牛顿法中无法达到最优点的问题, 但是当添加适当的选择条件时可能会取得更好的实验结果。

参考文献

- [1] An, H.B., Mo, Z.Y. and Liu, X.P.J. (2007) A Choice of Forcing Terms in Inexact Newton Method. *Journal of Computational and Applied Mathematics*, **200**, 47-60. <https://doi.org/10.1016/j.cam.2005.12.030>
- [2] 唐声平. 非线性偏微分方程多解计算的加速增广部分牛顿法[D]: [硕士学位论文]. 长沙: 湖南师范大学, 2017.
- [3] Qi, L. and Sun, J.J. (1993) A Nonsmooth Version of Newton's Method. *Mathematical Programming*, **58**, 353-367. <https://doi.org/10.1007/BF01581275>
- [4] Li, Y.J. and Li, D.H.J. (2009) Truncated Regularized Newton Method for Convexminimizations. *Computational Optimization & Applications*, **43**, 119-131. <https://doi.org/10.1007/s10589-007-9128-7>
- [5] Mccrae, B. and Stacey, K. (1987) *More Test Examples for Nonlinear Programming Codes*. Springer, Berlin, 1-271.
- [6] Barzilai, J. and Borwein, J.M.J. (1988) Two-Point Step Size Gradient Methods. *IMA Journal of Numerical Analysis*, **8**, 141-148. <https://doi.org/10.1093/imanum/8.1.141>
- [7] Grippo, L. and Lucidi, F.L.J. (1986) A Nonmonotone Line Search Technique for Newton's Method. *SIAM Journal on Numerical Analysis*, **23**, 707-716. <https://doi.org/10.1137/0723046>
- [8] Grippo, L., Lampariello, F. and Lucidi, S.J. (1991) A Class of Nonmonotone Stabilization Method in Unconstrained Optimization. *Numerische Mathematik*, **59**, 779-805. <https://doi.org/10.1007/BF01385810>
- [9] Grippo, L. and Sciandrone, M.J. (2002) Nonmonotone Globalization Techniques for the Barzilai-Borwein Gradient Method. *Computational Optimization and Applications*, **23**, 143-169. <https://doi.org/10.1023/A:1020587701058>
- [10] Raydan, M. (1997) The Barzilai and Borwein Gradient Method for the Large Scale Unconstrained Minimization Problem. *Society for Industrial and Applied Mathematics*, **7**, 26-33. <https://doi.org/10.1137/S1052623494266365>
- [11] Dai, Y.H., Huang, Y. and Liu, X.W.J. (2019) A Family of Spectral Gradient Methods for Optimization. *Computational Optimization and Applications*, **74**, 43-65. <https://doi.org/10.1007/s10589-019-00107-8>
- [12] Sun, D., Womersley, R.S. and Qi, H.J. (2002) A Feasible Semismooth Asymptotically Newton Method for Mixed Complementarity Problems. *Mathematical Programming*, **94**, 167-187. <https://doi.org/10.1007/s10107-002-0305-2>
- [13] Dai, Y.-H.J. (2002) R-Linear Convergence of the Barzilai and Borwein Gradient Method. *IMA Journal of Numerical Analysis*, **22**, 1-10. <https://doi.org/10.1093/imanum/22.1.1>