

# Sparse Nonnegative Solution of Underdetermined Linear Equations by Active Set Method

Peng Zhang\*, Zhensheng Yu#

College of Science, University of Shanghai for Science and Technology, Shanghai  
Email: zpwork2019@163.com, #zhsh-yu@163.com

Received: Jul. 9<sup>th</sup>, 2020; accepted: Jul. 24<sup>th</sup>, 2020; published: Jul. 31<sup>st</sup>, 2020

---

## Abstract

In this paper, for acquiring the sparse nonnegative solution of underdetermined linear equations, the original problem is relaxed into a  $l_0$  regularized optimization model. An active set identification technique is developed to accurately identify the zero components in a neighbourhood of the strict  $L$ -stationary point. Based on the active set identification technique, we propose an active set Barzilar-Borwein method to solve a  $l_0$  regularized minimization model. Numerical results show that the algorithm can effectively solve the sparse nonnegative solutions of underdetermined linear equations.

## Keywords

Sparse Nonnegative Solution,  $l_0$  Regularized Optimization Model, Strict  $L$ -Stationary Point, Active Set Barzilar-Borwein Method

---

# 有效集方法求解欠定线性方程组的稀疏非负解

张 鹏\*, 宇振盛#

上海理工大学理学院, 上海  
Email: zpwork2019@163.com, #zhsh-yu@163.com

收稿日期: 2020年7月9日; 录用日期: 2020年7月24日; 发布日期: 2020年7月31日

---

\*第一作者。

#通讯作者。

## 摘要

针对欠定线性方程组稀疏非负解的求解问题, 本文首先将原问题松弛为  $l_0$  正则优化模型。随之提出有效集方法识别严格  $L$ -稳定点邻域内的零分量, 基于这种快速识别技术, 设计了有效集 Barzilar-Borwein 算法求解  $l_0$  正则极小化模型。最后的数据实验证明该算法可以快速有效地求解欠定线性方程组的稀疏非负解。

## 关键词

稀疏非负解,  $l_0$  正则优化模型, 严格  $L$ -稳定点, 有效集 Barzilar-Borwein 算法

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

基于欠定线性方程组  $Ax = b$ , 这里  $b \in R^m$ ,  $x \in R^n$ ,  $A$  是  $m \times n$  ( $m < n$ ) 的矩阵,  $b$  是已知的,  $x$  是未知的, 本文只考虑非负解  $x \geq 0$  的部分。寻找欠定线性方程组的稀疏非负解, 就是求解如下的优化问题:

$$\begin{aligned} \min \|x\|_0 \\ \text{s.t. } Ax = b, x \geq 0. \end{aligned} \quad (1)$$

其中  $\|x\|_0$  指向量  $x$  中非零分量的个数。这类问题在信号处理, 图像恢复和多光谱数据处理等领域引起了广泛的研究[1] [2] [3]。

在现有的研究成果中, 主要用贪婪法和松弛法来解决问题(1)。贪婪法主要包括正交匹配追踪算法[4], 正则化正交匹配追踪算法[5]以及压缩采样匹配追踪[6]和子空间追踪[7]。这种类型的算法由于其相对较快的计算速度, 而在实践中得到了广泛的应用。然而, 当信号不是很稀疏或观测噪声水平较高时, 贪婪法的性能无法得到保证。

松弛法是通过用一定的非负连续函数代替  $l_0$  范数, 将  $l_0$  极小化模型转化为更易于处理的模型。常用于逼近  $l_0$  极小化模型的是  $l_1$  极小化模型。 $l_1$  极小化是一个凸优化问题, 在矩阵  $A$  满足适当的条件下,  $l_1$  极小化可以精确地恢复稀疏信号。因此,  $l_1$  极小化被认为是解决稀疏性问题的非常有用的工具并得到了广泛的运用。此外, 一些非凸函数  $l_p$  ( $0 < p < 1$ ) 范数被提出作为  $l_0$  范数的松弛。与  $l_1$  极小化方法相比, 非凸松弛模型通常具有更好的稀疏性, 但通常较难求解。

求解约束优化问题(1)主要有两种算法。第一种算法是迭代重加权算法, 其中两个最重要的迭代重加权算法是重加权  $l_1$  极小化[8]和迭代重加权最小二乘算法[9]。另一种方法通常称为正则化方法, 它通过引入正则化参数将问题(1)转化为下面的优化问题:

$$\begin{aligned} \min \phi(x) := f(x) + \mu \|x\|_0 \\ \text{s.t. } x \geq 0. \end{aligned} \quad (2)$$

其中  $f(x) = \frac{1}{2} \|Ax - b\|_2^2$ , 且  $\mu > 0$  是给定的正则化参数。

本文的其他内容如下:首先给出一些符号定义和关于问题(2)的一阶优化条件。基于 Cheng 和 Chen [10] 的研究,我们在加入非负约束  $x \geq 0$  后,重新定义了四种关于问题(2)的稳定点:基本稳定点,强基本稳定点,  $L$ -稳定点,严格  $L$ -稳定点,并说明了它们之间的关系。随之提出了有效集方法求解问题(2),这种方法能够精准地找出  $L$ -稳定点邻域内的值为零的分量。接着提出了有效集 Barzilar-Borwein 算法求解问题(2)。最后进行了数据实验,证明了该算法可以有效的求解欠定线性方程组的稀疏非负解。

## 2. 符号说明

在本节中,我们将定义一些符号说明。假设  $x^*$  是问题(2)的最优解,我们定义有效集  $\gamma(x^*)$  是  $x^*$  中零分量的下标集合,无效集  $\tau(x^*)$  是  $x^*$  中非零分量的下标集合:

$$\gamma(x^*) = \{i : x_i^* = 0\} \text{ 和 } \tau(x^*) = \{i : x_i^* > 0\}.$$

**定义 2.1** [11] 令  $v \in R^n$ ,  $\alpha > 0$ , 若向量值函数  $h_0(v, \alpha)$  满足

$$h_0(v, \alpha)_i = \begin{cases} v_i, & |v_i| > \sqrt{2\alpha}, \\ 0, & |v_i| \leq \sqrt{2\alpha}. \end{cases} \quad i = 1, 2, \dots, n.$$

则称  $h_0(v, \alpha)$  为硬阈值算子。

**注 2.1** 硬阈值算子是  $l_0$  极小化问题的最优解, 即

$$h_0(v, \alpha) = \arg \min_{u \in R^n} \left\{ \|u\|_1 : u = \arg \min_{x \in R^n} \left\{ \alpha \|x\|_0 + \frac{1}{2} \|x - v\|^2 \right\} \right\}.$$

在最近的研究中, Beck 和 Hallak [12] 提出了下面问题的一些基本定义

$$\min \{ f(x) + \mu \|x\|_0 : x \in B \}. \quad (3)$$

其中  $\mu > 0$ ,  $B \subseteq R^n$ 。

**定义 2.2** (基本稳定点) 如果  $B = R_+^n$ , 则有:

$$g_i(x^*) = 0, \forall i \in \tau(x^*). \quad (4)$$

其中  $g_i(x^*)$  表示  $f(x^*)$  在第  $i$  个分量的梯度。那我们称满足条件(4)的  $x^*$  为问题(2)的基本稳定点。

**定义 2.3** ( $L$ -稳定点) 向量  $x^*$  称作问题(3)的  $L$ -稳定点需满足:

$$\begin{aligned} x^* &\in \text{prox}_{\frac{\mu}{L} g_B} \left( x^* - \frac{1}{L} g(x^*) \right) \\ &= \arg \min_{x \in B} \left\{ \frac{\mu}{L} \|x\|_0 + \frac{1}{2} \left\| x - \left( x^* - \frac{1}{L} g(x^*) \right) \right\|^2 \right\}. \end{aligned}$$

其中  $L > 0$ ,  $g_B(x) = \delta_B(x) + \|x\|_0$  且  $\delta_B(x)$  是集合  $B$  的指标函数, 而指标函数也就是

$$\delta_B(x) = \begin{cases} 0, & x \in B; \\ 1, & x \notin B. \end{cases}$$

当  $B = R_+^n$  时, 问题(3)等价于问题(2), 则上述条件等价于下面的条件:

$$\begin{cases} 0 \leq g_i(x^*) \leq \sqrt{2\mu L}, i \in \gamma(x^*); \\ x_i^* \geq \sqrt{\frac{2\mu}{L}} \text{ 且 } g_i(x^*) = 0, i \in \tau(x^*). \end{cases}$$

向量  $x^*$  满足如下条件就称为问题(2)的严格  $L$ -稳定点:

$$\begin{cases} 0 \leq g_i(x^*) \leq \sqrt{2\mu L}, i \in \gamma(x^*); \\ x_i^* > \sqrt{\frac{2\mu}{L}} \text{ 且 } g_i(x^*) = 0, i \in \tau(x^*). \end{cases} \quad (5)$$

向量  $x^*$  满足如下条件就称为问题(2)的强基本稳定点:

$$\begin{cases} 0 \leq g_i(x^*) \leq \sqrt{2\mu L}, i \in \gamma(x^*); \\ g_i(x^*) = 0, i \in \tau(x^*). \end{cases}$$

由上述条件不难得出由问题(2)的严格  $L$ -稳定点可以推出问题(2)的  $L$ -稳定点, 强基本稳定点和基本稳定点。

对于函数  $\phi(x)$ , 称  $d \in R^n$  是函数  $\phi(x)$  在  $x$  处的下降方向, 如果常数  $\delta > 0$ , 使得

$$\phi(x + \alpha d) < \phi(x), \forall \alpha \in (0, \delta).$$

我们用  $D(x)$  表示函数  $\phi(x)$  的所有下降方向的集合:

$$D(x) = \{d \in R^n \mid \phi(x + \alpha d) < \phi(x), \forall \alpha \in (0, \delta)\}.$$

类似于文献[10]的定理 2.1, 下面的定理将给出集合  $D(x)$  更精确的表达形式, 这将有利于我们设计线性搜索方法解决问题(2)。

**定理 2.1** 假设  $x > 0$ , 由于  $f(x)$  是连续可微的, 集合  $D(x)$  可表示为:

$$D(x) = \left\{ d \in R^n \mid g_{\tau(x)}^T d_{\tau(x)} < 0, d_{\gamma(x)} = 0 \right\}.$$

### 3. 有效集估计

通常来说, 有效集的效率很大程度上取决于如何找到最优点处的有效约束。接下来, 将介绍有效集的估计并用它来逼近原问题的严格  $L$ -稳定点。我们用集合  $A(x)$  逼近  $\gamma(x^*)$ , 集合  $F(x)$  逼近  $\tau(x^*)$ :

$$A(x) = \left\{ i : 0 \leq x_i \leq \frac{1}{L} g_i(x) + \sqrt{\frac{2\mu}{L}} \right\},$$

和

$$F(x) = \left\{ i : x_i > \frac{1}{L} g_i(x) + \sqrt{\frac{2\mu}{L}} \right\}.$$

下面的定理表明, 当点  $x^k$  足够接近严格  $L$ -稳定点时, 则上述有效集的逼近集合  $A(x^k)$  是集合  $\gamma(x^*)$  的子集。

**定理 3.1** 假设  $x^*$  是问题(2)的严格  $L$ -稳定点, 序列  $\{x^k\}$  收敛于  $x^*$ , 则对于足够大的  $k$  有:

$$A(x^k) \subseteq \gamma(x^*) \text{ 且 } \tau(x^*) \subseteq F(x^k).$$

证明: 假设  $x^*$  是问题(2)的严格  $L$ -稳定点, 如果  $\tau(x^*) = \emptyset$ , 由于

$$A(x^k) \cup F(x^k) = \gamma(x^*) \cup \tau(x^*) = \{1, 2, \dots, n\}.$$

所以结论成立。假设  $\tau(x^*) \neq \emptyset$ , 对于任意的  $i \in \tau(x^*)$ , 由(5)可知:

$$x_i^* - \frac{1}{L} g_i(x^*) = x_i^* > \sqrt{\frac{2\mu}{L}}.$$

因为  $g(x)$  是连续的, 故有:

$$x_i^k - \frac{1}{L} g_i(x^k) > \sqrt{\frac{2\mu}{L}}.$$

因此,  $i \in F(x^k)$  也就是  $\tau(x^*) \subseteq F(x^k)$ 。

另一方面, 对于任意的  $i \in A(x^k)$ , 有

$$0 \leq x_i^k \leq \frac{1}{L} g_i(x^k) + \sqrt{\frac{2\mu}{L}}.$$

由于序列  $\{x^k\}$  收敛于  $x^*$ , 可得:

$$0 \leq x_i^* \leq \frac{1}{L} g_i(x^*) + \sqrt{\frac{2\mu}{L}}.$$

对于足够大的  $k$ , 可得  $x_i^* = 0$ 。再由最优性条件, 我们有  $0 \leq g_i(x^*) \leq \sqrt{2\mu L}$ 。因此,  $i \in \gamma(x^*)$  也就是  $A(x^k) \subseteq \gamma(x^*)$ , 证毕。

下面的定理表明, 当  $x^k$  是问题(2)的严格  $L$ -稳定点时, 我们可以得到集合  $A(x^k) = \gamma(x^k)$ 。

**定理 3.2** 假设  $x^k$  是问题的严格  $L$ -稳定点, 则有:

$$A(x^k) = \gamma(x^k), F(x^k) = \tau(x^k).$$

证明: 假设  $x^k$  是问题的严格  $L$ -稳定点, 对于任意的  $i \in \gamma(x^k)$ , 由条件(5)可得  $i \in A(x^k)$ 。而对于任意的  $i \in A(x^k)$ , 我们有  $x_i^k = 0$ 。但事实上, 如果存在  $x_i^k \neq 0$ , 由条件(5)可得  $g_i(x^k) = 0$  且有:

$$x_i^k \leq \sqrt{\frac{2\mu}{L}} \Rightarrow x_i^k = 0.$$

由此可得  $i \in \gamma(x^k)$ , 故  $A(x^k) = \gamma(x^k)$ 。同理可得  $F(x^k) = \tau(x^k)$ , 证毕。

## 4. 有效集算法

在本节中, 首先给出搜索方向的定义和一些相关性质, 然后介绍一种有效集 Barzilar-Borwein 算法来求解问题(2), 并分析了该算法的全局收敛性。

### 4.1. 搜索方向的一些性质

令  $x^k \in \Omega$  是第  $k$  个迭代点, 为了简化表述, 令

$$A^k = A(x^k), F^k = \{i \in F(x^k) : x_i > 0\}.$$

为了定义搜索方向, 进一步将  $A^k$  分成两个部分:

$$A_1^k = \{i \in A^k : x_i^k g_i(x^k) \geq 0\}, A_2^k = \{i \in A^k : x_i^k g_i(x^k) < 0\}.$$

搜索方向  $d^k = (d_{A_1^k}^k, d_{A_2^k}^k, d_{F^k}^k)$ , 定义  $d_{F^k}^k$  为:

$$d_i^k = -\lambda^k g_i^k, i \in F^k. \quad (6)$$

其中  $0 < \lambda_{\min} \leq \lambda^k < \lambda_{\max}$  是 Barzilar-Borwein 算法的步长, 其精确的定义将在 4.2 节中给出。集合  $A_1^k$  表示零分量的下标集合, 且包含满足条件(5)的变量的下标集合。集合  $A_2^k$  包含不满足条件(5)的变量的下标集合。因此, 给出定义:

$$d_i^k = -x_i^k, i \in A_1^k. \quad (7)$$

$$d_i^k = -g_i^k, i \in A_2^k. \quad (8)$$

下面的定理表明  $d^k = 0$  当且仅当  $x^k$  是问题(2)的一个严格  $L$ -稳定点。

**定理 4.1** 假设  $d^k$  由上述定义式(6)~(8)所表示。  $d^k = 0$  当且仅当  $x^k$  是问题(2)的一个严格  $L$ -稳定点。

证明: 令  $d^k = 0$ , 由  $A_2^k$  的定义和搜索方向  $d_i^k = -g_i^k = 0$ , 可得  $A_2^k = \emptyset$ 。如果  $i \in A_1^k$ , 由  $A_1^k$  的定义和  $d_i^k = -x_i^k = 0$  可得:

$$x_i^k = 0 \text{ 且 } g_i(x^k) \leq \sqrt{2\mu L}.$$

如果  $i \in F^k$ , 由  $F^k$  的定义可知:

$$x_i^k > \sqrt{\frac{2\mu}{L}} \text{ 且 } g_i(x^k) = 0.$$

反过来, 假设  $x^k$  是问题(2)的严格  $L$ -稳定点。如果  $i \in A^k$ , 我们称  $x_i^k = 0$ ,  $d_i^k = -x_i^k = 0$  且  $A_2^k = \emptyset$ 。事实上, 如果  $x_i^k \neq 0$ , 由式(5)可得:

$$g_i(x^k) = 0 \text{ 且 } x_i^k \leq \sqrt{\frac{2\mu}{L}} \Rightarrow x_i^k = 0.$$

这与假设条件矛盾。因此, 对  $\forall i \in F^k$  有:

$$g_i(x^k) = 0 \text{ 且 } d_i^k = 0.$$

证毕。

**定理 4.2** 假设  $x^k \rightarrow x^*$  且  $d^k \rightarrow 0$ , 其中  $d^k$  由式(6)~(8)所表示, 则  $x^*$  是问题(2)的一个  $L$ -稳定点。

证明: 对于  $\forall i \in A_1^k$ , 当  $x^k \rightarrow x^*$  时, 由于  $d_i^k = -x_i^k \rightarrow 0$ , 可得  $x_i^* = 0$ 。再由  $A_1^k$  的定义, 我们可得:

$$x_i^* = 0 \text{ 且 } g_i(x^*) \leq \sqrt{2\mu L},$$

对于  $\forall i \in A_2^k$ , 由  $d_i^k = -g_i^k \rightarrow 0$ , 可得:

$$g_i(x^*) = 0.$$

对于  $\forall i \in F^k$ , 由于  $g(x)$  的连续性和  $d^k \rightarrow 0$ , 有

$$x_i^* \geq \sqrt{\frac{2\mu}{L}} \text{ 且 } g_i(x^*) = 0.$$

证毕。

下面的定理表明, 如果  $x^k$  是问题(2)的强基本稳定点, 由(6)~(8)定义的  $d^k$  是  $f$  在  $x^k$  处的下降方向。

**定理 4.3** 假设  $d^k$  由(6)~(8)式定义, 可得

$$(g^k)^T d^k \leq 0. \quad (9)$$

此外, 等号成立当且仅当  $x^k$  是问题(2)的强基本稳定点。

证明: 由  $A_1^k$  的定义可知:

$$g_i^k d_i^k = -g_i^k x_i^k \leq 0, \forall i \in A_1^k.$$

再由  $A_2^k$  的定义可知:

$$g_i^k d_i^k = -(g_i^k)^2 < 0.$$

对于任意的  $i \in F^k$ , 我们有:

$$(g_i^k)^T d_i^k \leq -\lambda_{\min} \left\| (g_i^k)^T \right\|^2 \leq 0.$$

综上所述, 我们可以得到  $(g^k)^T d^k \leq 0$ 。

假设  $x^k$  是问题(2)的强基本稳定点。对于任意的  $i = 1, 2, \dots, n$ , 有  $x_i^k g_i(x^k) = 0$ 。由此, 对于任意的  $i \in A_1^k$ , 有  $d_i^k = -x_i^k$ , 则有  $(g^k)^T d^k = 0$ 。由  $A_2^k$  的定义可知  $A_2^k = \emptyset$ 。对于任意的  $i \in F^k$ , 我们有  $g_i^k = 0$ 。相反地, 假设不等式(9)的左边等于 0, 我们有:

$$g_i(x^k) = 0 \text{ 且 } x_i^k > \sqrt{\frac{2\mu}{L}}, \forall i \in F^k.$$

由  $A_2^k$  的定义可知  $A_2^k = \emptyset$ 。由  $A_1^k$  的定义可得:

$$x_i^k = 0, 0 \leq g_i^k \leq \sqrt{2\mu L} \text{ 或 } 0 < x_i^k \leq \sqrt{\frac{2\mu}{L}}, g_i^k = 0.$$

由此可知,  $x^k$  是问题(2)的强基本稳定点, 证毕。

## 4.2. 算法

基于上述讨论和非单调线性搜索[13], 类似于文献[10], 我们提出了一种有效集 Barzilar-Borwein 算法来求解问题(2)。但不同于文献[10]的是我们在加入非负约束后, 重新定义了有效集和有效集部分的搜索方向, 产生了不同的迭代点, 加快了算法的收敛速度。

**算法 4.1** (有效集 Barzilar-Borwein 算法, 简记 ABB 算法)

步 0 初始化  $x^0 \in R_+^n$ , 取  $\eta, \delta \in (0, 1)$ ,  $\lambda^0 = 1$  且  $\lambda_{\max} > \lambda_{\min} > 0$ , 正整数  $M$ , 取  $k = 0$ 。

步 1 进行收敛性检验, 如果满足停止条件, 则用近似解  $x^k$  终止。

步 2 计算  $\lambda^k: \lambda_{\min} \leq \lambda^k \leq \lambda_{\max}$  且  $d^k$  由式(6)~(8)确定。

步 3 如果  $\alpha^k = 1$ , 满足不等式

$$\phi(x^k + \alpha^k d^k) \leq \phi(x^k) - \delta (\alpha^k \|d^k\|)^2. \quad (10)$$

执行步 5, 否则执行步 4。

步 4 确定  $\alpha^k := \max\{\eta^j, j = 0, 1, \dots\}$  满足不等式

$$f(x^k + \alpha^k d^k) \leq \max_{0 \leq j \leq m(k)} f(x^{k-j}) - \delta (\alpha^k \|d^k\|)^2. \quad (11)$$

其中  $m(0) = 0$ ,  $0 \leq m(k) \leq \min[m(k-1) + 1, M]$ , 执行步 5。

步 5 计算  $x^{k+1} := x^k + \alpha^k d^k$ , 取  $k := k + 1$ , 返回步 1。

接下来, 将给出  $\lambda^k$  的显性形式, 运用了 Barzilar-Borwein (BB) 算法[14]的基本思想。类似于 Barzilar-Borwein 步的思想, 通过求解下面的优化问题:

$$\min_{\lambda \in R} \|D(\lambda) s_{F^k}^{k-1} - y_{F^k}^{k-1}\|^2.$$

其中  $s_{F^k}^{k-1} = x_{F^k}^k - x_{F^k}^{k-1}$  且  $y_{F^k}^{k-1} = g_{F^k}^k - g_{F^k}^{k-1}$ , 我们得到 Barzilar-Borwein 步长如下:

$$\lambda_{BB}^k = \frac{\|s_{F^k}^{k-1}\|^2}{(s_{F^k}^{k-1})^T y_{F^k}^{k-1}}.$$

为了控制  $\lambda_{BB}^k$  的大小值, 我们用区间  $[\lambda_{\min}, \lambda_{\max}]$  来限制, 其中  $\lambda_{\min} < \lambda_{\max}$  是给定的正常数。因此, 我们设

$$\lambda^k = \max\left\{\lambda_{\min}, \min\left\{\lambda_{\max}, \lambda_{BB}^k\right\}\right\} \quad (12)$$

为了简单起见, 我们将步长(12)代入式(6), 简化了算法 4.1。

### 4.3. 全局收敛性分析

下面将分析算法 4.1 的全局收敛性。类似于文献[13]中的分析, 可以得到了如下的引理。

**引理 4.1** 假设序列  $\{x^k\}$  是由算法 4.1 产生。如果存在一个正整数  $k_1$ , 对于所有  $k > k_1$ , 满足线性搜索(11), 我们有:

$$\lim_{k \rightarrow \infty} \alpha^k d^k = 0. \quad (13)$$

下面的定理和定理 4.2 表明序列  $\{x^k\}$  的任意聚点都是问题(2)的  $L$ -稳定点和强基本稳定点。

**定理 4.4** 假设序列  $\{x^k\}$  是由算法 4.1 产生。如果对于所有的  $k$  都有  $d^k \neq 0$ , 则序列  $\{x^k\}$  的任意聚点  $x^*$  都是问题(2)的  $L$ -稳定点或强基本稳定点。

证明: 假设对于所有的  $k$  都有  $d^k \neq 0$ , 且  $x^*$  为序列  $\{x^k\}$  的任意聚点。则存在一个无限指标集合  $K$ , 满足  $k \in K$  使得:

$$\lim_{k \rightarrow \infty} x^k = x^*.$$

考虑到不同集合  $A_1^k, A_2^k, F^k$  的数量是有限的, 对于任意的  $k \in K$ , 我们考虑两种情况。情形(1): 如果存在一个无限指标集合  $K_1 \subseteq K$ , 对于所有的  $k \in K_1$ , 线搜索(11)都满足。通过对不等式(10)求和, 可得

$$\lim_{k \rightarrow \infty} d^k = 0.$$

则由定理 4.2 可知结论成立。情形(2): 如果存在一个正整数  $k_1$ , 对于所有的  $k > k_1$ , 线搜索(11)都满足。如果序列  $\{\alpha^k\}_{k \in K}$  有一个非零的极限值, 我们由(13)式可知

$$\lim_{k \rightarrow \infty} d^k = 0.$$

则由定理 4.2 可得知结论成立。否则, 如果我们有  $\lim_{k \rightarrow \infty} \alpha^k = 0$ , 对于所有的  $k \geq 1$ , 由线搜索(11)规则可得

$$f\left(x^k + \frac{\alpha^k}{\eta} d^k\right) - f(x^k) \geq f\left(x^k + \frac{\alpha^k}{\eta} d^k\right) - \max_{0 \leq j \leq m(k)} f(x^{k-j}) > -\delta \left(\frac{\alpha^k}{\eta} \|d^k\|\right)^2. \quad (14)$$

另一方面, 对于充分大的  $k \in K$ , 由中值定理可得

$$f\left(x^k + \frac{\alpha^k}{\eta} d^k\right) - f(x^k) = \frac{\alpha^k}{\eta} \left( g\left(x^k + \theta^k \frac{\alpha^k}{\eta} d^k\right) - g(x^k) \right)^T d^k + \frac{\alpha^k}{\eta} g(x^k)^T d^k.$$

其中  $\theta^k \in (0, 1)$ 。将上述等式带入(14), 可得

$$\left( g\left(x^k + \theta^k \frac{\alpha^k}{\eta} d^k\right) - g(x^k) \right)^T d^k + g(x^k)^T d^k > -\delta \frac{\alpha^k}{\eta} \|d^k\|^2.$$

由于序列  $\{d^k\}_{k \in K}$  是有界的, 不失一般性地, 存在  $d^*$  使得  $\lim_{k \rightarrow \infty} d^k = d^*$ , 结合引理 4.1 可知,  $\lim_{k \rightarrow \infty} \alpha^k = 0$ 。再由最后一个不等式取极限得  $g(x^*)^T d^* > 0$ 。由定理 4.3 得出矛盾, 故结论成立, 证毕。



## 5. 数值实验

下面将进行一些数值实验来验证算法 4.1 对于求解欠线性方程组的稀疏非负解是有效的。所有的实验结果都是在联想笔记本电脑(2.50 GHz, 8.00 GB of RAM)上实现的, 数学软件采用 MATLAB R2016a。

基于文献[15]的实验证实了  $l_0$  优化问题可连续的有效性, 我们将算法中嵌入连续化过程。具体方法就是在考虑求解问题(2)前, 先求解如下系列问题

$$x_{\mu_k}^* := \arg \min_{x \in \mathbb{R}_+^n} \{ \phi_{\mu_k}(x) = f(x) + \mu_k \|x\|_0 \}.$$

其中  $\mu_0 > \mu_1 > \dots > \mu_N$  且  $N$  是一个正整数。我们用问题的解  $x_{\mu_{k-1}}^*$  作为下一个问题的初始迭代点。取  $\mu_0 = \frac{1}{2} \|A^T b\|_\infty^2$ ,  $\mu_N = 10^{-15} \mu_0$ , 区间  $[\mu_N, \mu_0]$  经过取对数化处理后, 再将其分成  $N$  个均匀分布的子区间。我们选择序列  $\{\mu_k\}$  作为子区间的端点, 并对  $\mu_k$  做了一个处理, 即令  $\mu_k = \max\{\mu_k, 0.005\}$ 。算法中涉及的参数设计如下:

$\lambda^0 = 1, M = 10, \lambda_{\min} = 10^{-3}, \lambda_{\max} = 10, \delta = 10^{-2}, \eta = 0.5$ 。首先用标准高斯分布的独立样本填充  $m \times n$  的度量矩阵  $A$ , 并且对这些行进行正交化。而真实解  $x^*$  是一个具有有效集  $A^*$  的  $T$ -稀疏信号, 观测向量  $b$  定义为:

$$b = Ax^* + e$$

其中  $e$  是根据均值为零且方差为  $10^{-2}$  的高斯分布绘制的。

在实验中, 我们研究了参数  $N$  和  $L$  对算法的影响, 设  $n = 5 \times 10^3$  和  $n = 10^4$ ,  $m = \text{round}(0.2 * n)$ , 并考虑了稀疏度的范围:  $x^*$  的非零分量的数量  $T$  在 1 到 60 都是可行的。算法的初始点是零向量, 当满足  $\|g(x^k)_{\tau(x^k)}\|_\infty \leq 10^{-5}$  或  $\|g(x^k)_{\gamma(x^k)}\|_\infty \leq 0.05$  条件时, 停止运算。下面将从平均迭代步数(Iter)、平均误差 ( $\|x^k - x^*\|_\infty$ ) 和运算时间(Time)三个方面来对 ABB 算法的性能进行评估。

**Table 1.**  $n = 5 \times 10^3, L = 1/5$ , the results of ABB algorithm to solve the problem

**表 1.**  $n = 5 \times 10^3, L = 1/5$  时 ABB 算法求解问题的计算结果

| $(N, T)$    | Iter  | $\ x^k - x^*\ _\infty$ | Time (sec.) |
|-------------|-------|------------------------|-------------|
| (10, 1:30)  | 26.86 | 4.14e-02               | 18.85       |
| (15, 1:30)  | 31.62 | 4.47e-02               | 19.21       |
| (20, 1:30)  | 35.55 | 4.60e-02               | 22.03       |
| (10, 31:60) | 30.47 | 5.56e-02               | 20.35       |
| (15, 31:60) | 35.53 | 5.62e-02               | 21.48       |
| (20, 31:60) | 39.55 | 5.49e-02               | 24.86       |

**Table 2.**  $n = 5 \times 10^3, L = 1/4$ , the results of ABB algorithm to solve the problem

**表 2.**  $n = 5 \times 10^3, L = 1/4$  时 ABB 算法求解问题的计算结果

| $(N, T)$   | Iter  | $\ x^k - x^*\ _\infty$ | Time (sec.) |
|------------|-------|------------------------|-------------|
| (10, 1:30) | 26.15 | 4.14e-02               | 18.03       |
| (15, 1:30) | 31.04 | 4.47e-02               | 19.08       |
| (20, 1:30) | 35.22 | 4.60e-02               | 20.67       |

Continued

|             |       |          |       |
|-------------|-------|----------|-------|
| (10, 31:60) | 30.50 | 5.56e-02 | 19.33 |
| (15, 31:60) | 35.80 | 5.62e-02 | 20.59 |
| (20, 31:60) | 39.53 | 5.49e-02 | 24.70 |

**Table 3.**  $n = 5 \times 10^3, L = 1/2$ , the results of ABB algorithm to solve the problem

**表 3.**  $n = 5 \times 10^3, L = 1/2$  时 ABB 算法求解问题的计算结果

| $(N, T)$    | Iter  | $\ x^k - x^*\ _\infty$ | Time (sec.) |
|-------------|-------|------------------------|-------------|
| (10, 1:30)  | 26.85 | 4.14e-02               | 18.87       |
| (15, 1:30)  | 31.89 | 4.47e-02               | 19.09       |
| (20, 1:30)  | 35.63 | 4.60e-02               | 21.33       |
| (10, 31:60) | 32.47 | 5.56e-02               | 24.51       |
| (15, 31:60) | 35.90 | 5.62e-02               | 25.47       |
| (20, 31:60) | 39.90 | 5.49e-02               | 27.64       |

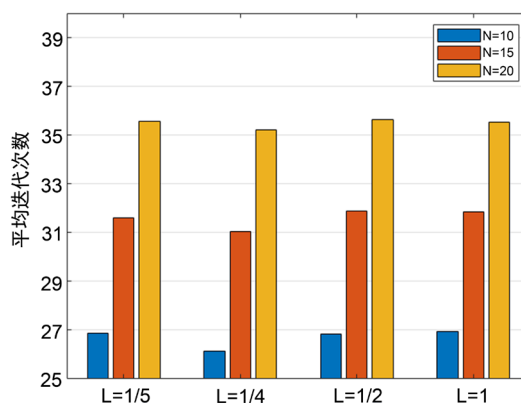
**Table 4.**  $n = 5 \times 10^3, L = 1$ , the results of ABB algorithm to solve the problem

**表 4.**  $n = 5 \times 10^3, L = 1$  时 ABB 算法求解问题的计算结果

| $(N, T)$    | Iter  | $\ x^k - x^*\ _\infty$ | Time (sec.) |
|-------------|-------|------------------------|-------------|
| (10, 1:30)  | 26.96 | 4.14e-02               | 19.03       |
| (15, 1:30)  | 31.85 | 4.47e-02               | 20.55       |
| (20, 1:30)  | 35.52 | 4.60e-02               | 21.10       |
| (10, 31:60) | 34.17 | 5.56e-02               | 23.65       |
| (15, 31:60) | 36.10 | 5.62e-02               | 24.88       |
| (20, 31:60) | 39.73 | 5.49e-02               | 26.78       |

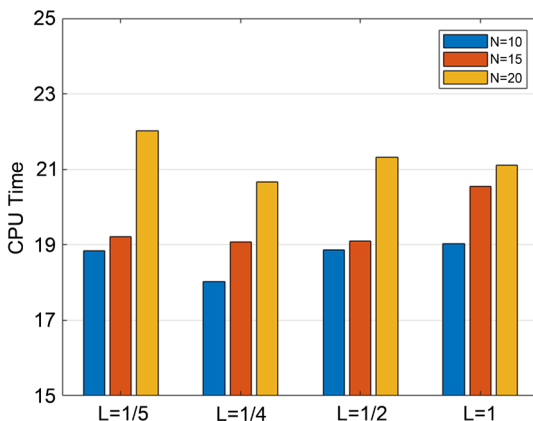
由上述表 1~4 可以看出, 当问题的规模取  $n = 5 \times 10^3$  时,  $T = 31:60, N = 15$ , ABB 算法产生了较大的平均误差。在  $T = 1:30, N = 10$  时, ABB 算法产生的平均误差较小。而  $L$  的取值变化对于 ABB 算法产生的平均误差影响很小。

对于上述表格中计算的数据, 我们将绘图分析在  $T = 1:30$  时,  $L$  和  $N$  的取值变化对于 ABB 算法求解问题的平均迭代次数和运算时间产生的影响。

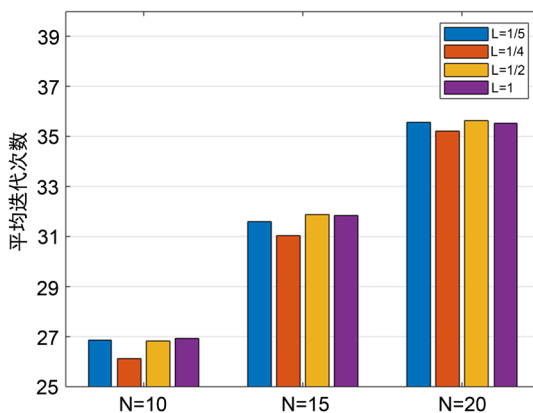


**Figure 1.** The influence of the change of  $N$  value on the average number of iterations at different  $L$  levels

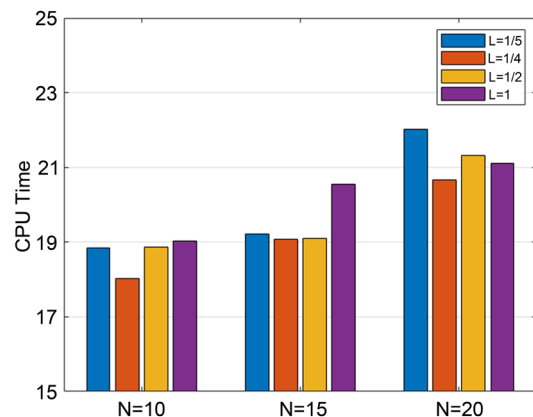
**图 1.** 不同  $L$  水平下,  $N$  取值变化对平均迭代次数的影响



**Figure 2.** The influence of the change of  $N$  value on operation time at different  $L$  levels  
**图 2.** 不同  $L$  水平下,  $N$  取值变化对运算时间的影响



**Figure 3.** The influence of the change of  $L$  value on the average number of iterations at different  $N$  levels  
**图 3.** 不同  $N$  水平下,  $L$  取值变化对平均迭代次数的影响



**Figure 4.** The influence of the change of  $L$  value on operation time at different  $N$  levels  
**图 4.** 不同  $N$  水平下,  $L$  取值变化对运算时间的影响

由图 1 和图 2 可以明显看出, 当  $L$  的值一定时, 随着  $N$  的取值变大, 迭代步数和运算时间都会增加。同样的, 由图 3 和图 4 可以明显看出, 当  $N$  的取值一定时, 可以看出  $L=1/4$  时, 平均迭代步数和运算时间都是最小的。由此可知, 当  $L=1/4, N=10$  时, 可以得到最优的实验结果。

**Table 5.**  $n = 10^4, L = 1/5$ , the results of ABB algorithm to solve the problem**表 5.**  $n = 10^4, L = 1/5$  时 ABB 算法求解问题的计算结果

| $(N, T)$    | Iter  | $\ x^k - x^*\ _\infty$ | Time (sec.) |
|-------------|-------|------------------------|-------------|
| (10, 1:30)  | 26.86 | 4.77e-02               | 76.62       |
| (15, 1:30)  | 30.62 | 4.70e-02               | 77.02       |
| (20, 1:30)  | 34.55 | 4.60e-02               | 79.56       |
| (10, 31:60) | 28.47 | 5.53e-02               | 80.35       |
| (15, 31:60) | 33.53 | 5.72e-02               | 81.46       |
| (20, 31:60) | 39.27 | 5.48e-02               | 84.06       |

**Table 6.**  $n = 10^4, L = 1/4$ , the results of ABB algorithm to solve the problem**表 6.**  $n = 10^4, L = 1/4$  时 ABB 算法求解问题的计算结果

| $(N, T)$    | Iter  | $\ x^k - x^*\ _\infty$ | Time (sec.) |
|-------------|-------|------------------------|-------------|
| (10, 1:30)  | 26.41 | 4.77e-02               | 75.03       |
| (15, 1:30)  | 30.59 | 4.70e-02               | 76.08       |
| (20, 1:30)  | 34.62 | 4.60e-02               | 78.77       |
| (10, 31:60) | 27.70 | 5.53e-02               | 77.53       |
| (15, 31:60) | 32.03 | 5.72e-02               | 80.49       |
| (20, 31:60) | 39.23 | 5.48e-02               | 82.50       |

**Table 7.**  $n = 10^4, L = 1/2$ , the results of ABB algorithm to solve the problem**表 7.**  $n = 10^4, L = 1/2$  时 ABB 算法求解问题的计算结果

| $(N, T)$    | Iter  | $\ x^k - x^*\ _\infty$ | Time (sec.) |
|-------------|-------|------------------------|-------------|
| (10, 1:30)  | 26.84 | 4.77e-02               | 76.97       |
| (15, 1:30)  | 30.90 | 4.70e-02               | 80.09       |
| (20, 1:30)  | 34.69 | 4.60e-02               | 81.53       |
| (10, 31:60) | 30.27 | 5.53e-02               | 80.51       |
| (15, 31:60) | 33.80 | 5.72e-02               | 81.67       |
| (20, 31:60) | 40.30 | 5.48e-02               | 83.54       |

**Table 8.**  $n = 10^4, L = 1$ , the results of ABB algorithm to solve the problem**表 8.**  $n = 10^4, L = 1$  时 ABB 算法求解问题的计算结果

| $(N, T)$    | Iter  | $\ x^k - x^*\ _\infty$ | Time (sec.) |
|-------------|-------|------------------------|-------------|
| (10, 1:30)  | 26.98 | 4.77e-02               | 77.03       |
| (15, 1:30)  | 30.93 | 4.70e-02               | 79.55       |
| (20, 1:30)  | 34.66 | 4.60e-02               | 81.10       |
| (10, 31:60) | 31.43 | 5.53e-02               | 80.65       |
| (15, 31:60) | 34.07 | 5.72e-02               | 82.68       |
| (20, 31:60) | 39.90 | 5.48e-02               | 83.78       |

由表 5-8 中的结果可知, 当问题规模达到  $n = 10^4$  时, 我们依旧可以得到相同的结论。当  $L$  的值一定时, 随着  $N$  的取值变大, 迭代步数和运算时间都会增加。当  $N$  的取值一定时, 可以看出  $L = 1/4$  时, 平均迭代步数和运算时间都是最小的。由此可知, 当  $L = 1/4, N = 10$  时, 可以得到最优的实验结果。综上可知, ABB 算法可以快速有效地求解欠定线性方程组的稀疏非负解。

## 6. 小结

本文主要讨论了求解欠定线性方程组的稀疏非负解问题。首先将线性约束条件惩罚到目标函数上, 给出了  $l_0$  正则优化模型的严格  $L$ -稳定点定义。随之提出有效集方法识别严格  $L$ -稳定点邻域内的零分量, 基于这种快速精准的识别技术, 设计了有效集 Barzilar-Borwein 算法, 并且证明了由该算法产生的序列的任意聚点都是强稳定点。最后进行了数据实验, 证明了该算法可以快速有效地求解欠定线性方程组的稀疏非负解。

## 参考文献

- [1] Donoho, D.L. (2006) Compressed Sensing. *IEEE Transactions on Information Theory*, **52**, 1289-1306. <https://doi.org/10.1109/TIT.2006.871582>
- [2] Bruckstein, A.M., Donoho, D.L. and Elad, M. (2009) From Sparse Solutions of Systems of Equations to Sparse Modeling of Signals and Images. *SIAM Review*, **51**, 34-81. <https://doi.org/10.1137/060657704>
- [3] Candès, E. and Tao, T. (2006) Near Optimal Signal Recovery from Random Projections: Universal Encoding Strategies. *IEEE Transactions on Information Theory*, **52**, 5406-5425. <https://doi.org/10.1109/TIT.2006.885507>
- [4] Tropp, J.A. and Gilbert, A. (2007) Signal Recovery from Random Measurements via Orthogonal Matching Pursuit. *IEEE Transactions on Information Theory*, **53**, 4655-4666. <https://doi.org/10.1109/TIT.2007.909108>
- [5] Needell, D. and Vershynim, R. (2010) Signal Recovery from Incomplete and Inaccurate Measurements via Regularized Orthogonal Matching Pursuit. *IEEE Journal of Selected Topics in Signal Processing*, **4**, 310-316. <https://doi.org/10.1109/JSTSP.2010.2042412>
- [6] Needell, D. and Tropp, J.A. (2008) CoSaMP: Iterative Signal Recovery from Incomplete and Inaccurate Samples. *Applied and Computational Harmonic Analysis*, **26**, 301-321. <https://doi.org/10.1016/j.acha.2008.07.002>
- [7] Dai, W. and Milenkovic, O. (2009) Subspace Pursuit for Compressive Sensing Signal Reconstruction. *IEEE Transactions on Information Theory*, **55**, 2230-2249. <https://doi.org/10.1109/TIT.2009.2016006>
- [8] Candès, E., Wakin, M.B. and Boyd, S.P. (2008) Enhancing Sparsity by Reweighted L1 Minimization. *Journal of Fourier Analysis and Applications*, **14**, 877-905. <https://doi.org/10.1007/s00041-008-9045-x>
- [9] Gorodnitsky, I.F. and Rao, B.D. (1997) Sparse Signal Reconstruction from Limited Data Using FOCUSS: A Reweighted Minimum Norm Algorithm. *IEEE Transactions on Signal Processing*, **45**, 600-616. <https://doi.org/10.1109/78.558475>
- [10] Cheng, W., Chen, Z. and Hu, Q. (2019) An Active Set Barzilar-Borwein Algorithm for L0 Regularized Optimization. *Journal of Global Optimization*, **76**, 769-791. <https://doi.org/10.1007/s10898-019-00830-w>
- [11] Zeng, J.S., Lin, S.B. and Xu, Z.B. (2014) Sparse Solution of Underdetermined Linear Equations via Adaptively Iterative Thresholding. *Signal Processing*, **97**, 152-161. <https://doi.org/10.1016/j.sigpro.2013.10.031>
- [12] Beck, A. and Hallak, N. (2018) Proximal Mapping for Symmetric Penalty and Sparsity. *SIAM Journal on Optimization*, **28**, 496-527. <https://doi.org/10.1137/17M1116544>
- [13] Grippo, L., Lampariello, F. and Lucidi, S. (1986) A Non-Monotone Line Search Technique for Newton's Method. *SIAM Journal of Numerical Analysis*, **23**, 707-716. <https://doi.org/10.1137/0723046>
- [14] Barzilai, J. and Borwein, J.M. (1988) Two Point Step Size Gradient Method. *IMA Journal of Numerical Analysis*, **8**, 141-148. <https://doi.org/10.1093/imanum/8.1.141>
- [15] Jiao, Y.L., Jin, B.J. and Lu, X.L. (2015) A Primal Dual Active Set with Continuation Algorithm for the L0 Regularized Optimization Problem. *Applied and Computational Harmonic Analysis*, **39**, 400-426. <https://doi.org/10.1016/j.acha.2014.10.001>