第 36 卷第 10 期　　电 子 与 信 息 学 报　　Vol.36No.10
2014 年 10 月　　Journal of Electronics & Information Technology　　Oct. 2014

稀疏信号重构的阈值化迭代检测估计

宋和平*[①]　　王国利[②]
[①](江苏大学计算机科学与通信工程学院　镇江　212013)
[②](中山大学信息科学与技术学院　广州　510006)

摘　要：研究压缩传感(Compressed Sensing, CS)的稀疏信号重构算法，该文提出一种新的算法框架 —— 阈值化迭代检测估计(Iterative Detection Estimation with Thresholding, IDET)。算法框架包括两个方面：选择单阶段阈值化(One-Stage Thresholding, OST)算法的迭代步作为支持集检测的参考；根据稀疏信号的特征设计支持集检测方法。同时，提出该算法框架的实现算法，实现算法先检测由迭代硬阈值化(Iterative Hard Thresholding, IHT)迭代步得到一个支持集，然后通过求解支持集上的最小二乘问题来估计待重构的稀疏信号，迭代上述两个步骤直至满足条件停止。IDET 算法的关键在于支持集检测，该文提出 3 种适用于快速衰减信号的支持集检测方法。实验结果表明，IDET 稀疏重构性能优于 IHT 的其他加速算法。

关键词：压缩传感；稀疏信号重构；贪婪算法；支持集检测；迭代检测估计
中图分类号：　TN911.7　　**文献标识码**：　A　　**文章编号**：1009-5896(2014)10-2431-07
DOI: 10.3724/SP.J.1146.2013.01696

Sparse Signal Recovery via Iterative Detection Estimation with Thresholding

Song He-ping[①]　　Wang Guo-li[②]
[①](*School of Computer Science and Telecommunication Engineering*, *Jiangsu University*, *Zhenjiang* 212013, *China*)
[②](*School of Information Science and Technology*, *Sun Yat-Sen University*, *Guangzhou* 510006, *China*)

Abstract: This paper devotes efforts to develope sparse signal recovery algorithms for Compressed Sensing (CS). The proposed algorithmic framework is called as Iterative Detection Estimation with Thresholding (IDET). IDET takes the One-Stage Thresholding (OST) as the reference for support detection, and devises support detection methods depended on the character of sparse signals. This study presents an implementation of IDET, which detects a support set by thresholding the result of the Iterative Hard Thresholding (IHT) iteration and estimates the reconstructed signal by solving a truncated least-squares problem on the support set, and it iterates these two steps until stop condition is met. The key to IDET lies in support detection, so this study explores three support detection strategies for fast decaying signals. The experimental results show IDET exhibits superior reconstruction performance than other accelerated algorithm for IHT.

Key words: Compressed Sensing (CS); Sparse signal recovery; Greedy algorithms; Support detection; Iterative detection estimation

1　引言

近年来，稀疏信号处理成为信号处理和应用数学等领域的一个新研究热点，特别是新兴发展的压缩传感(Compressed sensing, CS)理论[1,2]。压缩传感为稀疏信号处理提供了一个新的方向，可以通过欠采样的线性测量重构稀疏信号。压缩传感开创了很多新的应用，为此，多个国际期刊专门为压缩传感出版了专刊，总结了当前的研究进展[3−5]。

压缩传感由欠采样的线性测量重构未知稀疏信号，其稀疏信号重构是求解一个带稀疏约束的欠定线性问题即 ℓ_0 优化问题。然而，ℓ_0 问题是个 NP 难的非凸组合问题。为此，压缩传感的奠基工作[1,2]提出用 ℓ_1 问题代替 ℓ_0 问题，即基追踪(Basis Pursuit, BP)问题[6]。此后，很多研究人员提出各种稀疏信号重构的计算方法[7]。一般地可以分为 3 种：凸松弛算法、贝叶斯推理和贪婪算法。凸松弛算法将稀疏信号重构的 ℓ_0 问题替换为凸优化问题，如 BP 算法[6]等。贝叶斯推理算法由稀疏先验模型求解最大后验

2013-11-04 收到，2014-04-30 改回
国家自然科学基金(61375080, 61170126)，江苏省科技支撑计划项目(BE2013696)和江苏大学高级人才科研基金(12JDG050)资助课题
*通信作者：宋和平　songhp@ujs.edu.cn

估计，如贝叶斯压缩传感(Bayesian Compressive Sensing, BCS)[8]等。凸松弛算法依赖于最优化算法的实现，相对来说计算比较复杂耗时；贝叶斯推理算法依赖于信号的先验假设，而且算法参数比较多。贪婪算法具有易于设计实现、计算复杂度相对低的特点，吸引了很多研究者的关注。常见的算法有正交匹配追踪(Orthogonal Matching Pursuit, OMP)[9]、子空间追踪(Subspace Pursuit, SP)[10]、压缩采样匹配追踪(Compressive Sampling Matching Pursuit, CoSaMP)[11]、分阶段正交匹配追踪(Stagewise Orthogonal Matching Pursuit, StOMP)[12]和迭代硬阈值化算法(Iterative Hard Thresholding, IHT)[13]等。迭代硬阈值化是一种简单实用的稀疏信号重构方法，但其收敛速度比较慢。最近，文献[14]提出一种新的加速算法——双超松弛阈值化算法(Double OverRElaxation thresholding, DORE)。DORE 算法是一种基于条件期望值最大化(Expectation Conditional Maximization Either, ECME)迭代的加速算法，其中 ECME 由方差未知的高斯模型推导。对稀疏度 k 未知情况，文献[14]提出一种自动双超松弛阈值化算法(Automatic Double OverRElaxation thresholding, ADORE)。借用文献[15]的概念，本文把这一类算法分为单阶段阈值化(One-Stage Thresholding, OST)和双阶段阈值化(Two-Stage Thresholding, TST)[1)]。

鉴于单次求解 BP 问题的稀疏重构结果不够理想，特别是对测量次数相对较少的情况，文献[16]提出一种迭代改进 BP 的算法——迭代支持集检测(Iterative Support Detection, ISD)，其思想是通过迭代支持集检测来改善 BP 算法的重构结果。ISD 是一个算法框架，具体实现依赖于支持集检测方法。本文借鉴 ISD 算法中支持集检测的思想，把 ISD 算法应用于贪婪算法，提出一种新的稀疏信号重构算法框架——阈值化迭代检测估计(Iterative Detection Estimation with Thresholding, IDET)。IDET 算法框架为设计新的贪婪算法提供了思路，设计新的贪婪算法包括两个方面：(1)选择 OST 算法的迭代步作为支持集检测的参考；(2)根据稀疏信号的特征设计支持集检测方法。同时，本文提出该算法框架的实现算法。该实现算法首先检测 IHT 算法的迭代步得到一个支持集，然后通过求解支持集上的最小二乘问题来估计待重构的稀疏信号，迭代上述两个步骤直至满足停止条件。IDET 算法的关键在于支持集检测，本文提出 3 种适用于快速衰减信号[16]的支持集检测方法。快速衰减信号是指信号的绝对值从大到小衰减很快，常见的有稀疏高斯信号、稀疏拉普拉斯信号和一些幂律衰减信号。

2 基于阈值化的稀疏重构算法

本文关注的是迭代贪婪算法，借用文献[15]的概念，把这一类算法分为单阶段阈值化 OST 和双阶段阈值化 TST 算法。OST 算法是本文对应文献[15]中 TST 提出的概念，是指对算法迭代步进行一次阈值化就能重构稀疏信号的算法。下面将分别介绍 OST 算法和 TST 算法。另外，本节将介绍相关的迭代支持集检测 ISD 算法。

2.1 单阶段阈值化

文献[14]由高斯概率框架推导出 ECME 迭代：

$$\boldsymbol{x}^{(t)}=\boldsymbol{H}_k(\boldsymbol{x}^{(t-1)}+\boldsymbol{A}^{\mathrm{T}}(\boldsymbol{A}\boldsymbol{A}^{\mathrm{T}})^{-1}(\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}^{(t-1)})) \tag{1}$$

其中 $\boldsymbol{x}\in\mathbb{R}^n$ 为稀疏度为 k (即 $\|\boldsymbol{x}\|_0=k$，$\|\boldsymbol{x}\|_0$ 表示 $\boldsymbol{x}$ 的 ℓ_0 范数即非零元素个数)的未知信号，$\boldsymbol{A}\in\mathbb{R}^{m\times n}$ 是测量矩阵，$\boldsymbol{y}$ 是测量向量，$\boldsymbol{H}_k(\boldsymbol{x})$ 是硬阈值化即保留 $\boldsymbol{x}$ 绝对值前 k 个最大值其余赋值为 0。从式(1)可以看出，当测量矩阵 $\boldsymbol{A}$ 行正交时(即 $(\boldsymbol{A}\boldsymbol{A}^{\mathrm{T}})$ 是单位矩阵)，ECME 迭代步变成 IHT 迭代步。其它的对迭代步进行一次阈值化就能重构稀疏信号的算法还有迭代软阈值化[13]、Bregman 迭代算法[17]和迭代收缩阈值化算法[18]等。ECME 和 IHT 都可以通过有限次迭代得到稀疏解，但其收敛速度比较慢(线性收敛)[19]。为此，文献[14]提出一种加速算法——双超松弛阈值化 DORE。DORE 算法先利用两次超松弛步骤：

$$\boldsymbol{x}_1^{(t)}=\boldsymbol{x}^{(t)}+\alpha_1(\boldsymbol{x}^{(t)}-\boldsymbol{x}^{(t-1)}) \tag{2}$$

$$\boldsymbol{x}_2^{(t)}=\boldsymbol{x}_1^{(t)}+\alpha_2(\boldsymbol{x}_1^{(t)}-\boldsymbol{x}^{(t-2)}) \tag{3}$$

其中 α_1,α_2 是线搜索参数。最后，由一个硬阈值化步骤 $\boldsymbol{x}^{(t)}=\boldsymbol{H}_k(\boldsymbol{x}_2^{(t)})$ 来保证重构的稀疏信号是 k 稀疏的。

2.2 双阶段阈值化

考虑压缩传感的稀疏信号重构问题，双阶段阈值化 TST 算法的目的是检测一个支持集 I 来近似测量向量 $\boldsymbol{y}:\boldsymbol{y}=\boldsymbol{A}_I\boldsymbol{x}_I$。待重构的稀疏信号初始赋值为 $\boldsymbol{x}=\boldsymbol{0}$，TST 算法迭代计算下面两个步骤：

(1)支持集检测：　检测信号 $\boldsymbol{x}$ 的支持集 I，即选择测量矩阵 $\boldsymbol{A}$ 的哪些列去生成测量向量 $\boldsymbol{y}$，或者说确定用信号 $\boldsymbol{x}$ 的哪些分量来稀疏表示 $\boldsymbol{x}$。

(2)信号估计：　通过求解支持集 I 上的最小二乘问题来更新待重构的稀疏信号 $\boldsymbol{x}$，

---

1)阈值化(Thresholding)是一种非线性函数，如 MATLAB 函数 wthresh，定义硬阈值化 $\boldsymbol{H}_\theta(\boldsymbol{x})=\boldsymbol{x}\cdot(|\boldsymbol{x}|>\theta)$；软阈值化 $\boldsymbol{S}_\theta(\boldsymbol{x})=\mathrm{sgn}(\boldsymbol{x})(|\boldsymbol{x}|-\theta)_+$。本文所述的双阶段阈值化是文献[15]中 TST 概念的推广。

$$\boldsymbol{x}_I = \arg\min_{\boldsymbol{z}} \left\{ \left\| \boldsymbol{y} - \boldsymbol{A}_I \boldsymbol{z} \right\|_2^2, \operatorname{supp}(\boldsymbol{z}) \subseteq I \right\},\ \boldsymbol{x}_{[1,n]\backslash I} = \boldsymbol{0}$$

其中支持集定义为 $\operatorname{supp}(\boldsymbol{x}) = \{i : \boldsymbol{x}_i \neq 0\}$, $[1,n]\backslash I$ 表示集合 I 在 $\{1,2,\cdots,n\}$ 上的补集。TST 算法的支持集检测和信号估计分别被认为是一次阈值化，不同算法在支持集检测上各不相同，如 OMP, SP, CoSaMP, StOMP 等算法分别采用不同的策略进行支持集检测；信号估计都采用支持集上的最小二乘法。

2.3 迭代支持集检测

迭代支持集检测 ISD 算法[16]的提出是基于 BP 算法的稀疏重构结果，其思想是通过迭代支持集检测来改善 BP 算法的重构性能。初始一个支持集 $I = \varnothing$, ISD 算法迭代计算下面两个步骤：

(1) 支持集补集上的基追踪(BP)：定义 $T = I^C$ ，求解部分基追踪 BP 问题：

$$\boldsymbol{x} = \arg\min_{\boldsymbol{x}} \left\| \boldsymbol{x}_T \right\|_1, \quad \text{s.t. } \boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}$$

(2)支持集检测：　以第(1)步重构结果 $\boldsymbol{x}$ 作为参考检测得到一个支持集 I 。

迭代支持集检测(ISD)是一个算法框架，具体实现依赖于支持集检测方法。文献[16]针对快速衰减信号提出几种有效的支持集检测方法。这里只介绍其中一种简单实用的阈值化方法(如不作特别说明，下文中 ISD 算法特指由式(4)定义的方法)。

$$I^{(t)} = \left\{ i : |\boldsymbol{x}_i^{(t)}| > \beta^t \max \left| \boldsymbol{x}^{(t)} \right| \right\}, \quad \beta \in (0,1) \qquad (4)$$

3　阈值化迭代检测估计

鉴于 ISD 算法在理论和应用上的成功，本文把 ISD 算法中支持集检测的思想应用到双阶段阈值化 TST 算法，提出一种新的稀疏信号重构算法框架 — 阈值化迭代检测估计(IDET)，其算法流程如图 1 所示。阈值化迭代检测估计 IDET 属于 TST 算法，它利用 OST 算法作为其中支持集检测的参考。IDET 算法设计的关键在于：(1)选择合适的 OST 算法迭代步作为支持集检测的参考；(2)根据稀疏信号的特征设计支持集检测方法。由于 ECME 的迭代步需要计算存储逆矩阵 $(\boldsymbol{A}\boldsymbol{A}^{\mathrm{T}})^{-1}$ ，特别不适用于大规模应用，本文采用较为简单的 IHT 迭代步作为支持集检测的参考，设计 IDET 的实现算法。该实现算法首先检测 IHT 算法的迭代步得到一个支持集 I ，然后通过求解支持集 I 上的最小二乘问题来估计待重构的稀疏信号，迭代上述两个步骤直至满足停止条件。

IDET 算法的关键在于支持集检测，根据文献[16]对稀疏信号特征的观察，本文相应的只设计适用于快速衰减信号的支持集检测方法，提出 3 种不同的方法。综上，提出的 IDET 实现算法描述为：

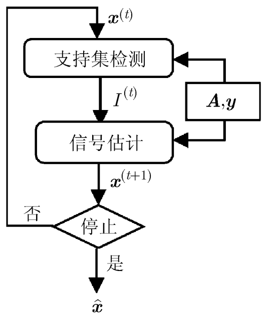

图 1 阈值化迭代检测估计(IDET)的算法流程图

(1)初始化：　初始化待重构的稀疏信号 $\boldsymbol{x}^{(0)} = \boldsymbol{0}$ ；初始化残差信号 $\boldsymbol{r}^{(0)} = \boldsymbol{y}$ ；初始化支持集 $I^{(0)} = \varnothing$ ；设迭代次数 $t = 1$ 。

(2)支持集检测：　更新信号估计：

$$\boldsymbol{x}^{(t)} = \boldsymbol{x}^{(t-1)} + \boldsymbol{A}^{\mathrm{T}} \boldsymbol{r}^{(t-1)}$$

检测支持集 I ：

方法 1：$I^{(t)} = \operatorname{supp}(\boldsymbol{H}_k(\boldsymbol{x}^{(t)}))$ ；

方法 2：$I^{(t)} = \left\{ i : \left| \boldsymbol{x}_i^{(t)} \right| > \beta^t \max \left| \boldsymbol{x}^{(t)} \right| \right\}$ ；

方法 3：$I^{(t)} = I^{(t-1)} \cup \{ i : |\boldsymbol{x}_i^{(t)}| \geq \gamma \max |\boldsymbol{x}^{(t)}|; i \notin I^{(t-1)} \}$ 。

(3)信号估计：　估计信号：

$$\boldsymbol{x}_{I^{(t)}}^{(t)} = \arg\min_{\boldsymbol{z}} \left\{ \left\| \boldsymbol{y} - \boldsymbol{A}_{I^{(t)}} \boldsymbol{z} \right\|_2^2,\ \operatorname{supp}(\boldsymbol{z}) \subseteq I^{(t)} \right\}$$

$$\boldsymbol{x}_{[1,n]\backslash I^{(t)}}^{(t)} = \boldsymbol{0}$$

更新残差信号：$\boldsymbol{r}^{(t)} = \boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}^{(t)}$ 。

(4)算法停止验证：　检查是否满足算法停止条件，若不满足，$t = t+1$，进入算法第(2)步；否则算法停止。待重构的稀疏信号 $\boldsymbol{x}$ 的非零分量在支持集 $I^{(t)}$ 上，相应的支持向量为 $\boldsymbol{x}_{I^{(t)}}^{(t)}$ 。

本文提出了 3 种支持集检测方法，相应的 IDET 实现算法命名为 IDET-k , IDET-β 和 IDET-γ ，其中 $\beta \in (0,\ 1)$， $\gamma \in (0,\ 1]$是阈值化参数。

评注：

(1)IDET 算法是本文借鉴 ISD 算法思想提出的算法框架，其实现算法采用 OST 算法 IHT 的迭代步作为参考来检测支持集 I 。文献[20]是本文算法框架的特例，该算法由基于 Bernoulli-Gaussian 模型的二值假设检验推导的迭代步也是 IHT 的迭代步，但其不涉及支持集检测方法的设计。OMP, SP, CoSaMP 等方法采用残差信号与测量矩阵的相关度($\boldsymbol{A}^{\mathrm{T}} \boldsymbol{r}^{(t)}$)作为检测支持集的参考，而 $\boldsymbol{A}^{\mathrm{T}} \boldsymbol{r}^{(t-1)}$ 是最小二乘的负梯度。IDET 实现算法中支持集检测参考的是最小二乘的梯度下降迭代步 $\boldsymbol{x}^{(t)} + \boldsymbol{A}^{\mathrm{T}} \boldsymbol{r}^{(t)}$ ，梯度下降迭代步由于提供了稀疏信号估计在算法设计上更为灵活，可以根据稀疏信号的不同特点设计不同的支持集检测方法。

(2)IDET-β 算法的支持集检测方法与 ISD 算法类似，都是通过与指数递减的阈值相比保留绝对值较大的分量。IDET-γ 算法的支持集检测方法则保留当前检测到的支持集 I，下次迭代在支持集 I 的补集上作检测。这两种支持集检测方法都不像 IDET-k 一样每次迭代确定支持集的势为 k，适用于稀疏度 k 未知的应用。

(3)IDET 算法思想与 ISD 算法类似，关键的区别在于支持集检测方法。它们最大的不同之处在于支持集检测参考的是不同的稀疏重构方法。IDET 算法支持集检测参考的是 OST，而 ISD 算法支持集检测参考的是 BP。由此可见 IDET 算法比 ISD 算法在计算复杂度上更具优势。IDET 估计信号采用的是支持集 I 上的最小二乘法，而 ISD 估计信号是在支持集 I 的补集上求解 BP。由此可见 IDET 算法比 ISD 算法更适合于大规模应用。

(4)IDET 算法可认为是一种与 DORE 不同的 OST 加速算法。DORE 先通过 IHT 估计稀疏信号 $\boldsymbol{x}^{(t)}$，然后通过两步超松弛加速算法，最后进行一次硬阈值化来保证重构的稀疏信号是 k 稀疏的。DORE 需要知道先验稀疏度 k，而 IDET-β，IDET-γ 不需要这个强条件，只需给定一个阈值化参数即可。采用自动双超松弛阈值化 ADORE 需要很多次的 DORE 来估计稀疏度 k。

(5)阈值化迭代检测估计的 IDET-β 和 IDET-γ 跟阈值化参数有关。阈值化参数越大稀疏重构效果越好，相应地其支持集 I 增加地越慢，这样需要更多的迭代次数(见实验部分)。

(6)与其他贪婪算法，如 OMP, SP, CoSaMP 等不同，阈值化迭代检测估计的 IDET-k 每次迭代确定支持集的势为 k，而不是逐次增加 1 或多个(k，$2k$)索引到支持集 I 后再进行一次硬阈值化。IDET-β 的支持集 I 是非递增或嵌套的，而 IDET-γ 的支持集 I 是递增或嵌套的。

(7)正如 ISD 算法所讨论，阈值化迭代检测估计的 IDET-β 和 IDET-γ 只适用于快速衰减信号，对那些信号绝对值衰减很慢甚至不衰减(如二值 0/+1、三值 0/−1/+1 信号)情况不适用，阈值化迭代检测估计的 IDET-k 没有这个限制。总体来说，阈值化迭代检测估计 IDET 算法对非快速衰减信号没有显示出比 ℓ_1 最优化更好的稀疏重构性能[15,21]，但可以通过线性或非线性映射把非快速衰减信号转化为适用于 IDET 算法的快速衰减信号[22]。

4 计算复杂度分析

阈值化迭代检测估计的 IDET-k，IDET-β 和 IDET-γ 算法主要的计算步骤为矩阵向量乘法(matrix-vector multiplication)。迭代硬阈值化 IHT 迭代的计算复杂度为 $\mathcal{O}(mn)$，那么算法 IDET-k，IDET-β 和 IDET-γ 的更新稀疏信号的计算复杂度为 $\mathcal{O}(mn)$。阈值化步骤的计算复杂度为 $\mathcal{O}(n\lg k)$ 2)，求解支持集上的最小二乘所采用的近似共轭梯度法的计算复杂度为 $\mathcal{O}(ckm)$ (c 为近似共轭梯度法迭代次数)3)，更新残差信号的计算复杂度为 $\mathcal{O}(km)$，那么每次迭代所需计算复杂度总和为 $\mathcal{O}(mn+n\lg k+ckm+km)$。双超松弛阈值化法(DORE)算法两次硬阈值化的计算复杂度为 $\mathcal{O}(2n\lg k)$，两次超松弛步骤的计算复杂度为 $\mathcal{O}(2mn)$，那么每次迭代所需计算复杂度总和为 $\mathcal{O}(3mn+2n\lg k)$。IDET 算法和 DORE 算法需存储测量矩阵 $\boldsymbol{A}$，其所需的存储量为 $\mathcal{O}(mn)$。OMP, SP 等 TST 算法的计算复杂度分析参考文献[23]。

5 实验结果

本节介绍仿真实验，所有代码均由 MATLAB 实现 4)，部分采用了文献[14,21]提供的代码，运行环境为 MATLAB v7.6，计算机操作系统为 Windows XP, 2.53 GHz Intel Celeron CPU, 2 GB 内存。实验比较算法选择 OST 的加速算法 DORE 算法和 TST 算法中性能较好的 SP 算法[15,21]，IDET 算法不与 ℓ_1 最优化(如 ISD 算法)和其他 OST/TST 算法(如 OMP 算法等)比较，这些算法已经在文献[14,15,21]中有详细的比较。算法评价有两组：相变(phase transitions)曲线和 Shepp-Logan Phantom 图像重构，更多的实验比较见文献[23]。

5.1 相变曲线

在实验中，所有测量矩阵 $\boldsymbol{A}$ 都采用均匀球面矩阵集(Uniform spherical ensemble)，即测量矩阵 $\boldsymbol{A}$ 是随机高斯矩阵并对 $\boldsymbol{A}$ 归一化。实验所用 k 稀疏的信号 $\boldsymbol{x}$ 是随机选择势为 k 的支持集，相应的元素值由高斯分布均匀生成。为公平比较算法，实验中迭代停止条件设最大迭代次数为 100 或相对误差为：$\left\|\boldsymbol{y}-\boldsymbol{r}^{(t)}\right\|_2/\|\boldsymbol{y}\|_2\le 10^{-6}$。定义 $\rho=k/m$ 为稀疏性度量(sparsity)，$\delta=m/n$ 为不确定性度量(indeterminacy)，那么得到评价稀疏信号重构的 2 维相平面 $(\delta,\rho)\in[0,1]^2$。对每一个 δ，通过插值找到稀疏重构概率大于 0.50 的所有 ρ，这样得到相变曲线。曲线之下表示稀疏重构成功，反之则失败。越

2)算法 IDET-k 的支持集检测需找出前 k 个最大值即计算复杂度为 $\mathcal{O}(n\lg k)$，IDET-β 和 IDET-γ 的支持集检测只需两次遍历查找即计算复杂度为 $\mathcal{O}(2n)$

3)算法 IDET-β 和 IDET-γ 支持集 I 变化不一，但 $|I|\le k$，这里以 $|I|\le k$ 计算。实验部分近似共轭梯度法迭代次数设为 20。

4)所有代码可在 https://github.com/songhp/IDET/下载。

高的相变曲线表示算法重构能力越好。本实验采用文献[21]所用代码生成相变曲线。确定稀疏信号维数 n=400，$\delta\in[0.05, 0.50]$，$\rho\in[0.05, 0.50]$，线性选取 16 对(ρ,δ)。每对(δ,ρ)实验运行 100 次。采用 $\|\boldsymbol{x}_0-\boldsymbol{x}\|_2/\|\boldsymbol{x}_0\|_2<0.01$($\boldsymbol{x}_0$ 是真实稀疏信号)判定稀疏重构准确。

实验测试阈值化参数 β 对算法 IDET-β 的影响，设 β=0.7,0.8,0.9，选取 δ=0.05, 0.20, 0.35, 0.50，阈值化参数 β 对算法 IDET-β 的影响如图 2 所示，相应的相变曲线如图 3 所示。从图中可以看出，β 越大稀疏信号重构的准确重构率越高，稀疏重构性能越好。同样地，测试阈值化参数 γ 对算法 IDET-γ 的影响，设 γ=0.7,0.8,0.9，选取 δ=0.05, 0.20, 0.35, 0.50，阈值化参数 γ 对算法 IDET-γ 的影响如图 4 所示，相应的相变曲线如图 3 所示。从图中可以看出，γ 越大稀疏信号重构的准确重构率越高，稀疏重构性能越好。图 5 实验对比了 DORE, SP 算法与 IDET-k 算法的准确重构率，可以看出 IDET-k 比 DORE 准确重构率高，稀疏重构性能更好；在 m/n 比较大时，SP 比 IDET-k 稀疏重构性能更好。从图 3 可以看出，所有算法中 DORE 算法稀疏重构性能最差，阈值化迭代检测估计的 IDET-β , IDET-γ 依赖于阈值化参数，参数越大算法稀疏重构性能越好；SP 算法在 m/n>0.4 时显示出更好的稀疏重构性能。

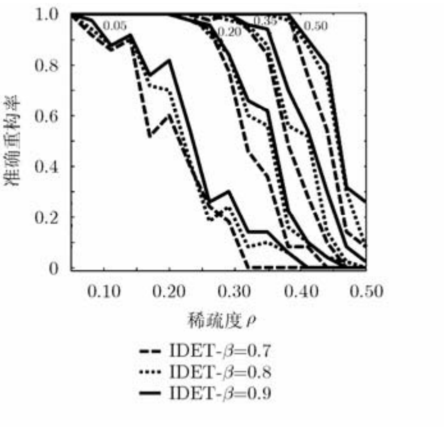

图 2 算法 IDET-β 不同阈值化参数的准确重构率比较

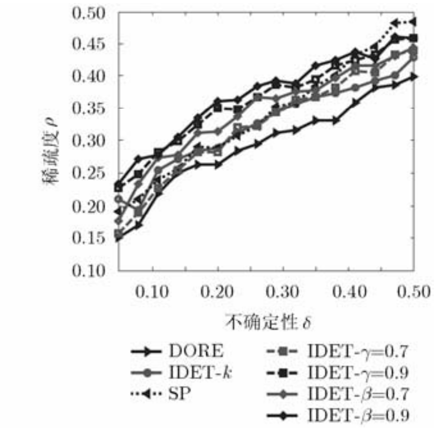

图 3 算法 DORE, SP, IDET 的相变曲线比较

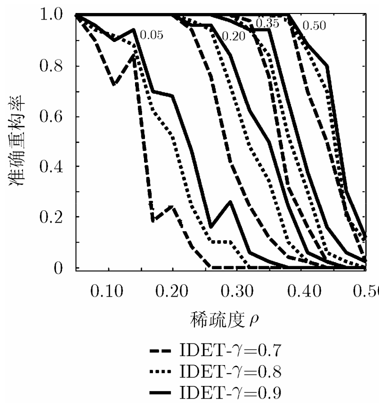

图 4 算法 IDET-γ 不同阈值化参数的准确重构率比较

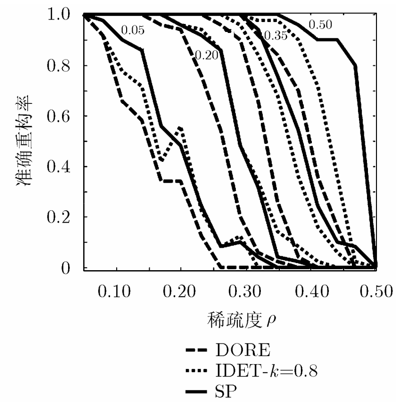

图 5 算法 DORE, SP 与 IDET-k 的准确重构率比较

5.2 图像重构

本小节实验比较 IDET 算法与 DORE, SP 算法重构 Shepp-Logan Phantom 图像。测量向量 $\boldsymbol{y}$ 是 2 维离散傅里叶变换射线采样系数。测量矩阵 $\boldsymbol{A}=\boldsymbol{\Phi\Psi}$，$\boldsymbol{\Phi}\in\mathbb{R}^{m\times n}$ 是采样矩阵(部分离散傅里叶变换矩阵)，$\boldsymbol{\Psi}\in\mathbb{R}^{m\times n}$ 是稀疏化矩阵(逆 Harr 小波变换)。Shepp-Logan Phantom 图像的 Harr 小波变换系数是稀疏的。稀疏重构性能评价标准采用峰值信噪比(Peak Signal-to-Noise Ratio, PSNR)、迭代次数和 CPU 运行时间。峰值信噪比 PSNR(dB)= $20\lg\left((\boldsymbol{\Psi x}_0)_{\max}-(\boldsymbol{\Psi x}_0)_{\min}\right)/\|\boldsymbol{x}_0-\boldsymbol{x}\|_2$，其中 $\boldsymbol{x}_0$ 是 Harr 小波变换系数，$(\boldsymbol{\Psi x}_0)_{\max}$, $(\boldsymbol{\Psi x}_0)_{\min}$ 分别表示 Shepp-Logan Phantom 图像的最大值、最小值。为公平比较算法，实验中迭代停止条件设最大迭代次

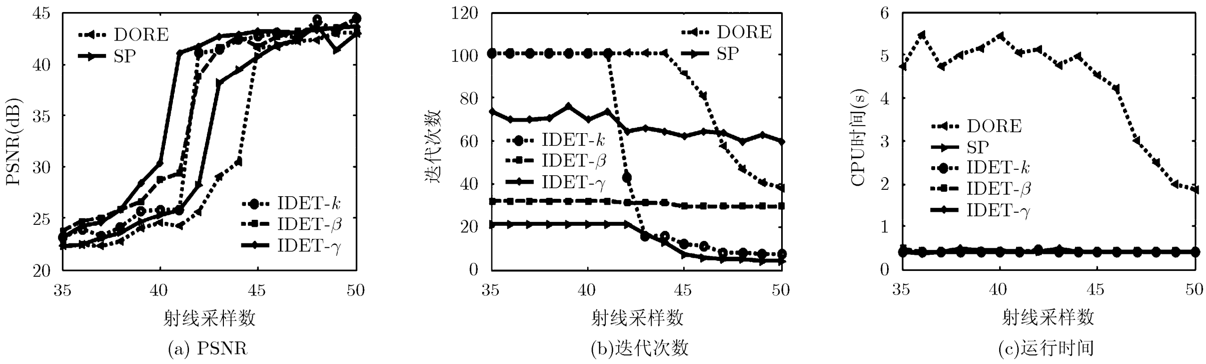

图 6 重构 Shepp-Logan Phantom 图像的峰值信噪比 PSNR、迭代次数和 CPU 运行时间比较

数为 1000 或相对误差为：$\|\boldsymbol{y}-\boldsymbol{r}^{(t)}\|_2/\|\boldsymbol{y}\|_2 \le e$。对图像大小为 256×256，设 $e=0.01$, $\beta=0.8$, $\gamma=0.8$。图 6显示的是图像射线采样数为35～50时IDET算法与 DORE, SP 算法的重构结果比较。IDET 的 3 个实现算法 IDET-k，IDET-β，IDET-γ 重构图像的 PSNR 都要比 DORE, SP 算法大。算法 SP, IDET-β 迭代次数要比其他算法少，IDET-k 在射线采样数比较高时，其迭代次数下降很明显。算法 IDET-γ 和 DORE 在很多情况下维持比较高的迭代次数，但就 CPU 运行时间而言，IDET-γ 跟 SP, IDET-β，IDET-k 一样要比 DORE 算法少很多。

6 结束语

本文研究压缩传感的稀疏信号重构算法，把贪婪算法分为两类：OST 与 TST。受 ISD 算法思想的启发，本文提出一种结合 OST 和 TST 的贪婪算法框架——IDET。IDET 算法与传统的 TST 不同之处在于采用 OST 作为支持集检测的参考，可以根据稀疏信号的特征设计相应的支持集检测方法。针对快速衰减信号，IDET 的实现算法设计了 3 种检测 IHT 迭代步的方法。实验结果表明，IDET 实现算法比 DORE 算法、SP 算法(大多数情况下)具有更好的稀疏重构性能。贪婪算法以其快速、易于实现(特别是对大规模应用)的特点在压缩传感研究领域得到广泛的关注，如国内的研究[24]。IHT 之类的 OST 算法实现简单，但其收敛比较慢。本文提出的 IDET 实现算法是一种比 DORE 算法更具优势的 IHT 加速算法。根据 IDET 算法思想，我们可以设计更多的 OST 加速算法。ISD 算法在理论与实践上表明重构相同稀疏度的稀疏信号所需的测量次数比 BP 算法更少，为此，研究 IDET 算法在理论上减少测量次数成为新的方向。

参 考 文 献

[1] Candes E, Romberg J, and Tao T. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information[J]. *IEEE Transactions on Information Theory*, 2006, 52(2): 489–509.

[2] Donoho D. Compressed sensing[J]. *IEEE Transactions on Information Theory*, 2006, 52(4): 1289–1306.

[3] Bajwa W, Leus G, Scaglione A, *et al.*. Guest editorial: special issue on compressive sensing in communications[J]. *Physical Communication*, 2012, 5(2): 61–63.

[4] Allstot D, Rovatti R, and Setti G. Editorial: special issue of compressive sensing[J]. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 2012, 2(3): 337–339.

[5] Ahmad F, Arce G, Narayanan R, *et al.*. Special section guest editorial: compressive sensing for imaging[J]. *Journal of Electronic Imaging*, 2013, 22(2): 020901-1–020901-2.

[6] Chen S, Donoho D, and Saunders M. Atomic decomposition by basis pursuit[J]. *SIAM Review*, 2001, 43(1): 129–159.

[7] Qaisar S, Bilal R, Iqbal W, *et al.*. Compressive sensing: from theory to applications, a survey[J]. *Journal of Communications and Networks*, 2013, 15(5): 443–456.

[8] Ji S, Xue Y, and Carin L. Bayesian compressive sensing[J]. *IEEE Transactions on Signal Processing*, 2008, 56(6): 2346–2356.

[9] Tropp T and Gilbert A. Signal recovery from random measurements via orthogonal matching pursuit[J]. *IEEE Transactions on Information Theory*, 2007, 53(12): 4655–4666.

[10] Dai W and Milenkovic O. Subspace pursuit for compressive sensing signal reconstruction[J]. *IEEE Transactions on Information Theory*, 2009, 55(5): 2230–2249.

[11] Needell D and Tropp J. CoSaMP: iterative signal recovery from incomplete and inaccurate samples[J]. *Communications of the ACM*, 2010, 53(12): 93–100.

[12] Donoho D, Tsaig Y, Drori I, *et al.*. Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit[J]. *IEEE Transactions on Information Theory*, 2012, 58(2): 1094–1121.

[13] Blumensath B and Davies M. Iterative hard thresholding for

compressed sensing[J]. *Applied and Computational Harmonic Analysis*, 2009, 27(3): 265–274.

[14] Qiu K and Dogandzic A. Sparse signal reconstruction via ECME hard thresholding[J]. *IEEE Transactions Signal Processing*, 2012, 60(9): 4551–4569.

[15] Maleki A and Donoho D. Optimally tuned iterative reconstruction algorithms for compressed sensing[J]. *IEEE Journal of Selected Topics in Signal Processing*, 2010, 4(2): 330–341.

[16] Wang Y and Yin W. Sparse signal reconstruction via iterative support detection[J]. *SIAM Journal on Imaging Sciences*, 2010, 3(3): 462–491.

[17] Yin W, Osher S, Goldfarb D, *et al.*. Bregman iterative algorithms for l_1-minimization with applications to compressed sensing[J]. *SIAM Journal on Imaging Sciences*, 2008, 1(1): 143–168.

[18] Beck A and Teboulle M. A fast iterative shrinkage thresholding algorithm for linear inverse problems[J]. *SIAM Journal on Imaging Sciences*, 2009, 2(1): 183–202.

[19] Blumensath T. Accelerated iterative hard thresholding[J]. *Signal Processing*, 2012, 92(3): 752–756.

[20] Amini A, Babaie-Zadeh M, and Jutten C. A fast method for sparse component analysis based on iterative detection-estimation[C]. International Workshop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering, Paris, France, 2006: 123–130.

[21] Sturm B. Sparse vector distributions and recovery from compressed sensing[OL]. http://arxiv.org/pdf/1103.6246v2, 2013.10.

[22] Zhang X, Wen J, Han Y, *et al.*. An improved compressive sensing reconstruction algorithm using linear non-linear mapping[C]. Proceedings of Information Theory and Applications Workshop, La Jolla, CA, USA, 2011: 1–7.

[23] 宋和平. 稀疏信号重构的贪婪算法及其应用[D]. [博士论文], 中山大学, 2011.
Song He-ping. Greedy algorithms with application to sparse signal recovery[D]. [Ph.D.dissertation], Sun Yat-Sen University, 2011.

[24] 刘记红, 黎湘, 徐少坤, 等. 基于改进正交匹配追踪算法的压缩感知雷达成像方法[J]. 电子与信息学报, 2012, 34(6): 1344–1350.
Liu Ji-hong, Li Xiang, Xu Shao-kun, *et al.*. Compressed sensing radar imaging methods based on modified orthogonal matching pursuit algorithms[J]. *Journal of Electronics & Information Technology*, 2012, 34(6): 1344–1350.

宋和平： 男，1983 年生，博士，研究方向为智能信息处理.

王国利： 男，1965 年生，教授，研究方向为压缩传感支配的智能感知.