**SCIENCE CHINA**
Information Sciences

• **Supplementary File** •

# Sparse signal reconstruction via generalized two-stage thresholding

Heping SONG[1,2*], Zehong AI[1], Yuping LAI[3*], Hongying MENG[4] & Qirong MAO[1]

[1]*School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang* 212013*, China;*
[2]*Jiangsu Key Laboratory of Security Technology for Industrial Cyberspace, Jiangsu University, Zhenjiang* 212013*, China;*
[3]*School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing* 100875*, China;*
[4]*Department of Electronic and Computer Engineering, Brunel University London, Middlesex UB8 3PH, UK*

## Appendix A Algorithm

**Algorithm A1** GTST-$\alpha$ Algorithm

**Require:** $A$, y, $\alpha$, $t$=1, $\mathrm{x}^0$=$\mathbf{0}$, $\mathrm{r}^0$=y, $\mathrm{d}^0$=$A'\mathrm{r}^0$, $I^0$=$\emptyset$, $T^0$=$[1,n]$.
**Ensure:** The reconstructed signal x.
**while** stopping condition false **do**
  $\mathrm{x}^t = \mathrm{x}^{t-1} + \mathrm{d}^{t-1}$
  $I^t = \{i : |\mathrm{x}_i^t| \geqslant \alpha \ \max|\mathrm{x}_j^t|;\ i \in [1,n],\ j \in T^{t-1},\ \alpha \in (0,1]\}$
  $T^t = [1,n] \setminus I^t$
  $\mathrm{x}_{I^t}^t = A_{I^t}^{\dagger}\mathrm{y}$
  $\mathrm{x}_{T^t}^t = \mathbf{0}$.
  $\mathrm{r}^t = \mathrm{y} - A\mathrm{x}^t$
  $\mathrm{d}_{T^t}^t = A'_{T^t}\mathrm{r}^t$
  $\mathrm{d}_{I^t}^t = \mathbf{0}$
  $t = t + 1$
**end while**
**return** x

## Appendix B Experiments

In this section, we conduct numerical experiments via computer simulations to illustrate the effectiveness of the proposed approaches. We implemented GTST-$\alpha$, GraDes [1]and SDAR [2] in MATLAB running on Windows 10 with 2.8-GHz Intel i7-7700HQ Quad Core CPU and 16-GB memory. The code of GraDes and SDAR is available on the author's homepage. In all the experiments, the unknown sparse signals were initialized with zero vectors. The nonzero entries of sparse signals were chosen as a random support set of elements, and their values were sampled uniformly. The sampling/measurement matrix $A$ was randomly generated from the standard i.i.d. Gaussian distribution in each trial; each column was then normalized to have unit norm. The tested algorithms were stopped upon the relative error fell below stopping tolerance $10^{-6}$ or the number of iterations is greater than 100. In all the cases, the performance of the algorithms was quantified by the probability of exact reconstruction, i.e., the rate of exact reconstruction on Monte Carlo simulation repeated 200 times. A relative error of less than $10^{-2}$ was counted as the exact reconstruction.

First, we present a demo of the signal reconstruction performed in [3]. The sparse signal was generated with sparsity $k$=25, length $n$=200. We set $m$=60, created a $m \times n$ sampling/measurement matrix $A$, and set y := $A$x. As observed by [3], a sparse signal with 25 nonzeros cannot be easily reconstructed using only 60 measurements with signal dimension $n$=200. The top left panel of Figure C1 shows the perfect reconstruction using GTST-$\alpha$ with $\alpha$=0.8 in merely 14 iterations. However, SDAR exhibits a number of false nonzeros and missed nonzeros when using the same setting as GTST (As shown in the top right panel of Figure C1). The dynamic thresholding introduced in GTST-$\alpha$ is important since it allows GTST to exactly reconstruct the unknown sparse signal using fewer measurements than SDAR.

We turn now to test the performance comparisons of GTST, SDAR, and GraDes in terms of the probability of exact reconstruction. First, we fixed the signal dimension $n$=1000, and sample size $m$=400 and varied the sparsity level $k$=50:10:240. Here, $k$ =50:10:240 means the sparsity level ranges from 50 to 240 with an increment of 10. The results are plotted in the bottom left panel of Figure C1. It is clear that the performance of all the three methods worsened as $k$ increased. With smaller sparsity level $k$=50:100, all three methods performed well in recovering the true signal. As $k$ increased, GraDes was the first one that failed to reconstruct signal and vanished when $k > 160$. However, GTST performed well even when $k$=160. Next, we tested the comparisons

* Corresponding author (email: songhp@ujs.edu.cn, laiyuping@qq.com)

by fixing the signal dimension $n$=500, sparsity level $k$=20 and varying the sample size $m$=20:10:200. As shown in the bottom right panel of Figure C1, the probabilities of the exact reconstruction of sparse signals of GTST and SDAR are higher than GraDes as $m$ increases. In summary, these experimental results indicate that GTST achieves better reconstruction performance with fewer measurements than SDAR and GraDes.

## Appendix C Figure

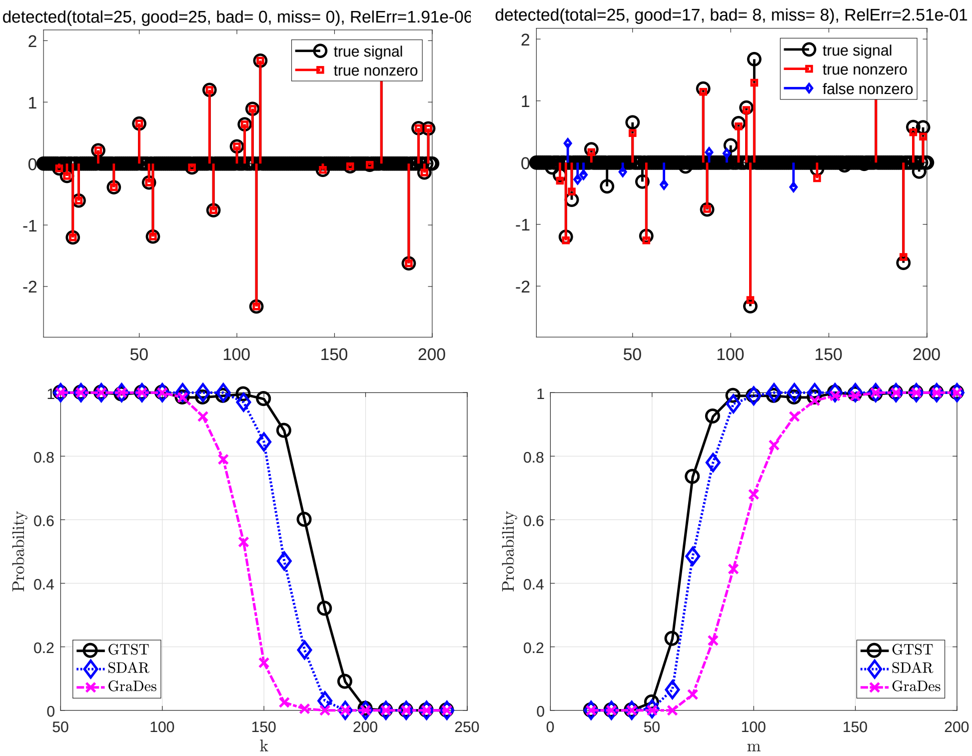


**Figure C1** (Color online) Numerical experiments. A demo of sparse signal reconstruction using GTST (top left panel) and SDAR (top right panel). Performance comparison in terms of the probability of exact reconstruction by varying sparsity level $k$ (bottom left panel) and sample size $m$ (bottom right panel).


**References**

1 Garg R, Khandekar R. Gradient descent with sparsification: An iterative algorithm for sparse recovery with restricted isometry property. ICML, 2009, 337-344

2 Huang J, Jiao Y, Liu Y, et al. A constructive approach to L0 penalized regression. J Mach Learn Res, 2018, 19: 1-37

3 Wang Y, Yin W. Sparse signal reconstruction via iterative support detection. SIAM J Imaging Sci, 2010, 3: 462-491