

# An Evolutionary Multiobjective Approach to Sparse Reconstruction

Lin Li, *Student Member, IEEE*, Xin Yao, *Fellow, IEEE*, Rustam Stolkin, *Member, IEEE*,  
Maoguo Gong, *Member, IEEE*, and Shan He, *Member, IEEE*

**Abstract**—This paper addresses the problem of finding sparse solutions to linear systems. Although this problem involves two competing cost function terms (measurement error and a sparsity-inducing term), previous approaches combine these into a single cost term and solve the problem using conventional numerical optimization methods. In contrast, the main contribution of this paper is to use a multiobjective approach. The paper begins by investigating the sparse reconstruction problem, and presents data to show that knee regions do exist on the Pareto front (PF) for this problem and that optimal solutions can be found in these knee regions. Another contribution of the paper, a new soft-thresholding evolutionary multiobjective algorithm (StEMO), is then presented, which uses a soft-thresholding technique to incorporate two additional heuristics: one with greater chance to increase speed of convergence toward the PF, and another with higher probability to improve the spread of solutions along the PF, enabling an optimal solution to be found in the knee region. Experiments are presented, which show that StEMO significantly outperforms five other well known techniques that are commonly used for sparse reconstruction. Practical applications are also demonstrated to fundamental problems of recovering signals and images from noisy data.

**Index Terms**—Compressed sensing, evolutionary algorithm, knee region, multiobjective optimization, Pareto front, sparse reconstruction, zero norm.

## I. INTRODUCTION

SPARSE reconstruction algorithms have been widely used to solve problems involving large under-determined systems of linear equations [1]–[3]. They have found

Manuscript received August 15, 2012; revised December 24, 2012 and August 13, 2013; accepted October 16, 2013. Date of publication October 24, 2013; date of current version November 26, 2014. This work was supported in part by the EU FP7-PEOPLE-2009-IRSES project on Nature Inspired Computation and its Applications under Grant 247619, in part by the National Natural Science Foundation of China under Grant 61371201, Grant 61001202, and Grant 61273317, and in part by the National Top Youth Talents Support Program of China. This paper was recommended by Associate Editor K. Deb.

L. Li and M. Gong are with the Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, International Research Center for Intelligent Perception and Computation, Xidian University, Xi'an 710071, China (e-mail: xdlinli86@163.com; gong@ieee.org).

X. Yao and S. He are with CERCIA, School of Computer Science, University of Birmingham, Edgbaston, Birmingham B15 2TT, U.K. (e-mail: x.yao@cs.bham.ac.uk; s.he.1@cs.bham.ac.uk).

R. Stolkin is with School of Mechanical Engineering, University of Birmingham, Edgbaston, Birmingham B15 2TT, U.K. (e-mail: r.stolkin@cs.bham.ac.uk).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TEVC.2013.2287153

many applications in signal processing and computer vision [4]–[6]. More specifically, consider an under-determined system of linear equations  $\mathbf{y} = \mathbf{Ax}$  where matrix  $A \in R^{M \times N}$  with  $M \leq N$ ,  $\mathbf{y} \in R^M$  and  $\mathbf{x} \in R^N$ . The unknown sparse signal  $\mathbf{x}_{tr}$  can be recovered from the under-determined linear system, where  $\mathbf{y}$  is called the measurement vector and the full-rank and over-complete matrix  $A$  is the sensing matrix. Several algorithms, e.g.,[7], [8], have been proposed to solve this problem based on the prior knowledge that  $\mathbf{x}_{tr}$  contains many zero elements. Hence, the aim is to recover the original signal  $\mathbf{x}_{tr}$  from the following under-determined system of the linear equations (1), under the constraint that the vector  $\mathbf{x}$  is very sparse

$$\mathbf{y} = \mathbf{Ax}. \quad (1)$$

If noise is involved during the measurement process, (1) can be reformatted as

$$\mathbf{y} = \mathbf{Ax} + \mathbf{n} \quad (2)$$

where vector  $\mathbf{n} \in R^M$  represents the additive noise during the observation process.

According to compressed sensing theory [1], sparse solutions to (1) and (2) are approximations to the ground-truth signal  $\mathbf{x}_{tr}$ . The sparsity of a signal is represented by its  $l_0$ -norm, which counts the number of nonzero entries in it. In the noise-free or noisy cases, the following equations are obtained respectively:

$$\min_{\mathbf{x}} \|\mathbf{x}\|_0 \quad s.t. \quad \mathbf{y} = \mathbf{Ax} \quad (3)$$

$$\min_{\mathbf{x}} \|\mathbf{x}\|_0 \quad s.t. \quad \|\mathbf{y} - \mathbf{Ax}\|_2^2 \leq \sigma. \quad (4)$$

Equation (3) represents the noiseless case, whereas (4) tries to recover  $\mathbf{x}_{tr}$  under measurement noise, where  $\sigma$  is a nonnegative parameter to estimate the noise level in the data. Solving the above equations, which involves  $l_0$ -norm for general  $\mathbf{x}$ , are proven to be NP-hard optimization problems [9], and several different approaches are commonly employed to deal with this NP-hard problem, including:

- 1) greedy algorithms, typically as orthogonal matching pursuit [10], [11];
- 2) iterative hard thresholding methods [8], [12];
- 3) relaxation algorithms, which replace  $\|\mathbf{x}\|_0$  by a differentiable function  $\|\mathbf{x}\|_1$  or  $l_p$ -norm with  $0 < p < 1$  [13];
- 4) algorithms that take advantage of the statistical characterization of the signal [14].

The  $l_1$ -minimization algorithms, which employ  $\|\mathbf{x}\|_1$  to induce sparsity, are among the most popular approaches to sparse reconstruction. They include gradient projection methods [7], homotopy methods [15], [16], and iterative shrinkage-thresholding (IST) [17]. For a comprehensive review of fast  $l_1$ -minimization algorithms, see [6].

Problems (3) and (4) are constrained optimization problems [18], which are typically solved using penalty functions. These methods transform the two objectives into one by multiplying each objective function by a weighting factor and then summing up all contributions, i.e., they take a weighted sum of the objective functions. The choice of these weighting factors has great impact on the resulting reconstructions; however, a shortcoming of these penalty function methods is that it is not obvious how to optimally choose these key parameters so as to maximize performance.

As an example, consider the  $l_1$ -minimization algorithm IST [17], which solves the following optimization problem:

$$\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{Ax} - \mathbf{y}\|_2^2 + \lambda \|\mathbf{x}\|_1. \quad (5)$$

Clearly, different choices for  $\lambda$  in the above equation will yield different optimal solutions and it is not easy to know the optimal  $\lambda$ , which will result in the best estimation of the ground-truth signal. Fig. 1 illustrates a simple experiment to demonstrate how the parameter  $\lambda$  affects the results of the algorithm. In Fig. 1, the horizontal axis represents possible choices of  $\lambda$ , and the vertical axis plots error in terms of the difference between the ground-truth signal and the reconstructed solutions (normalized by dividing by the magnitude of the original signal). This experiment illustrates how strongly such algorithms can be affected by a poor choice of  $\lambda$ . In this example, the best solution is obtained when the parameter  $\lambda$  is about 0.01. For this experiment, the sparse ground-truth signal was randomly generated, with a sparsity ratio 0.1 and its nonzero values drawn from the standard normal distribution. The subset of its nonzero coefficients was randomly chosen. The sensing matrix  $A \in R^{M \times N}$  is a Gaussian random dictionary, whose elements are independent and identically distributed (i.i.d.) as the standard normal distribution, with the ambient dimension  $N = 2000$  and the projection dimension  $M = 1200$ . Ten independent trials were performed for each choice of  $\lambda$ , and the graph shows the average of these. In each trial, the measurement vector  $\mathbf{y}$  was corrupted by an additive white noise term whose coefficients are i.i.d. distributed as  $N(0, 0.01)$ .

One way of avoiding the difficulties of penalty methods, is to convert the constrained problems into multiobjective problems (MOPs). MOPs have gained increasing interest during the past ten years [19]–[21] and have been shown to be useful for solving constrained problems [22]. Evolutionary algorithms and other population-based approaches are among the most popular methods for handling MOPs [23]–[27]. In this paper, we present an improved evolutionary multiobjective method, and also show how it can be used for solving the sparse reconstruction problem, which has usually been solved using penalty functions, without a multiobjective approach.

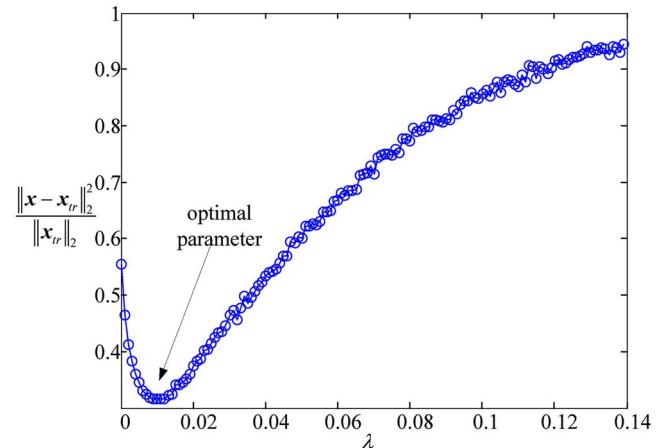


Fig. 1. Variation in the performance of the IST algorithm with different choices of the parameter  $\lambda$ .

Since penalty optimization algorithms transform MOPs into scalar problems through weighting factors, their optimal solutions are actually included in the Pareto set of the MOPs. Each solution for a particular choice of weighting factor corresponds to one of the solutions in the Pareto set. All the solutions in the Pareto set are optima of the MOPs and they represent different compromises between the competing objectives. Therefore, we also examine the problem of determining which solution in a Pareto set will generate the best reconstruction to the original signal in the sparse reconstruction problem. MOP researchers have studied this problem of tradeoff of multiple objectives and found some interesting regions, such as knee areas [28], where further improvement in one objective causes a rapid degradation in other objectives. Since MOPs always involve contradictory objective functions, decision makers often prefer their solutions to lie in these knee areas. Additionally, the best reconstruction of a ground-truth signal may also correspond to solutions in this region. Knee areas [29], [30] are solutions that have the maximum marginal rates of return, i.e., for which an improvement in one objective causes a severe degradation in another. Knee regions have attracted considerable interest in the study of MOPs and decision makers have been shown to prefer solutions that lie in knee areas in many applications [28], [29], [31]. Note that, in the worst case, even if a knee region solution in a particular problem does not turn out to provide the best possible approximation for the ground-truth signal, then the solution will still be a Pareto solution, which means that these solutions are still optimal in the sense of multiobjective optimization. Such solutions still offer an optimal tradeoff between reconstruction error and a sparsity requirement, and they can still provide satisfactory approximations to the ground-truth signals, according to sparse reconstruction theory.

This paper is concerned with the parameters in sparse reconstruction models (sparsity and measurement error). It is hard to choose the relative weightings for these parameters properly so that the solutions yielded by the model are optimal. Similar to other MOPs, our experiments show that knee areas in sparse signal models also indicate the best balance between the measurement error and the sparsity constraint, and give the best performance in signal reconstruction.

Therefore, we first present a novel method, soft-thresholding evolutionary multiobjective algorithm (StEMO), for solving zero-norm sparse reconstruction models using a multiobjective approach. We then show how knee regions can be used to find an optimal tradeoff between competing objective functions, which will yield the best possible estimation of the original signal. The experimental results show that the best approximation to the ground-truth signal of the sparse reconstruction problems lies in the knee areas, and the method for finding the knee area for this problem is also introduced in Section II in detail.

The remainder of this paper is organized as follows. In Section II, the proposed algorithm is presented in detail. Section III presents experimental data, showing how the performance of StEMO compares against five other commonly used techniques. Concluding remarks are given in Section IV.

## II. MULTIOBJECTIVE APPROACH TO SPARSE RECONSTRUCTION

Evolutionary algorithms have been widely used to solve MOPs [21], [23], as well as for seeing applications in other areas where multiple contradictory objectives are involved, for example digital filter design [32], software engineering [33], and machine learning [24], [26]. In this paper, our contributions are: first to present an improved evolutionary algorithm which incorporates a soft-thresholding step to achieve multiobjective optimization; second, to show how this MOP approach can be applied to sparse reconstruction to yield better results than the use of conventional single objective methods.

Considering the sparse constraint and measurement error as two objectives, the sparse reconstruction problem can be formulated as a bi-objective optimization problem

$$\min_{\mathbf{x}} \mathbf{F}(\mathbf{x}) = \min_{\mathbf{x}} (\|\mathbf{x}\|_0, \|\mathbf{Ax} - \mathbf{y}\|_2^2). \quad (6)$$

A new multiobjective evolutionary algorithm StEMO (see Table I) based on NSGA-II [23] is presented to solve this problem. First, we improve the performance of NSGA-II, with respect to better convergence and accuracy, by incorporating a problem-depended local search and a k-nearest neighbor product-based vicinity distance [34]. Next, using  $PF$  obtained by the improved multiobjective evolutionary algorithm, an angle-based method is used to obtain the knee areas, where a solution at the knee region can be found.

The procedure of the proposed algorithm is given in Table I.  $t$  is the current generation number and  $P^t$  is the current population.  $n_{mutation}$  and  $n_{crossover}$  are the number of individuals for mutation and crossover operators, respectively, which are obtained by the corresponding mutation and crossover rates. The mutation and crossover operators are applied to the current population to obtain new solutions. Then, a problem-dependent local search operator (described in Section II-A) is used to improve the performance of the algorithm and the k-nearest neighbor list (kNNL) [35] is employed to obtain the next generation. Once the termination test is met, we stop the algorithm and get the optimal  $PF$ . For decision making we use the angle-based method (described in Section II-C) to find the knee points of the problem. Here, we use nonuniform mutation

TABLE I  
PROCEDURE OF STEMO

<b>Step1</b>	<b>Initialization:</b> Set $t=0$ , generate the initial population $P^0$ randomly, and set the termination criterion.
<b>Step2</b>	<b>Implementation of mutation operator:</b> Choose $n_{mutation}$ individuals from population $P^t$ randomly. Apply the mutation operator on these individuals, and replace them by the offspring ones. Population $P_{mutation}^t$ is obtained.
<b>Step3</b>	<b>Implementation of crossover operator:</b> Choose $n_{crossover}$ individuals from $P_{mutation}^t$ randomly. Apply the crossover operator on these solutions and obtain a sub-population $P_{crossover}^t$ .
<b>Step4</b>	<b>Evaluation and obtain the non-dominated set:</b> Calculate the fitness of $P_{crossover}^t$ . Combine $P_{crossover}^t$ and $P^t$ , i.e. $P_{combine}^t = P_{crossover}^t \cup P^t$ . Obtain the non-dominated set of $P_{combine}^t$ which is denoted as $P_{non\_dominated}^t$ and the dominated set $P_{dominated}^t$ .
<b>Step5</b>	<b>Implementation of local search operator:</b> Apply the local search operator on $P_{non\_dominated}^t$ and $P_{dominated}^t$ , and obtain a sub-population $P_{local\_search}^t$ .
<b>Step6</b>	<b>Implementation of selection operator:</b> Combine $P_{local\_search}^t$ , $P_{crossover}^t$ , and $P^t$ . Then one can obtain $P_{cmb\_pop}^t = P_{local\_search}^t \cup P_{crossover}^t \cup P^t$ . Select the next population $P^{t+1}$ from $P_{cmb\_pop}^t$ by applying the selection operator.
<b>Step7</b>	<b>Termination Test:</b> If termination criterion is satisfied, stop the algorithm and output the results; if not set $t = t + 1$ then turn to Step2.
<b>Step8</b>	<b>Obtainment of knee region:</b> Angle-based method [29] is used to obtain the knee regions.

[36] and BLX- $\alpha$  [37] as the mutation and crossover operators (the parameter  $b$  to determine the strength of the mutation operator is usually  $3 \sim 5$ ). Moreover, given the sparsity of the signal in this problem, we improved the nonuniform mutation to increase the chance to obtain more sparse solutions. For a parent solution  $\mathbf{x} = (x_1, \dots, x_k, \dots, x_N)$ ,  $x_k$  is selected to perform the mutation and  $x'_k$  is its mutated result.  $x'_k$  is modified as  $x''_k$  to insure the sparsity of the solution after mutation

$$x''_k = \begin{cases} 0 & \text{if } r > 0.9 \\ x'_k & \text{otherwise} \end{cases} \quad (7)$$

where  $r$  is a random number between 0 and 1. Other operators, such as soft-thresholding local search, selection operator, and the method for finding the knee points, will be discussed in the following subsections.

### A. Soft-Thresholding Local Search

In this section, we describe Step 5 of the StEMO algorithm as shown in Table I. In order to obtain better solutions efficiently, we employ an additional step, beyond crossover and mutation, in which individuals are further perturbed using a soft-thresholding local search strategy improved from IST [17]. For clarity, we first briefly introduce the IST algorithm before describing how the local search operator is incorporated within an evolutionary algorithm.

1) *Introduction of IST:* IST is one of the  $l_1$ -minimization algorithms that can solve the problem of (5).  $\lambda$  is a Lagrangian multiplier that balances the tradeoff between the two-norm of measurement error and the sparsity-enforcing one-norm of the solution. IST treats the convex problem (5) as a special case of the following unconstrained optimization problem:

$$G(\mathbf{x}) = f(\mathbf{x}) + \lambda p(\mathbf{x}) \quad (8)$$

where  $f : R^N \rightarrow R$  is a smooth and convex function, and  $p : R^N \rightarrow R$  is a separable function but not necessarily

smooth nor convex. Function  $p(\mathbf{x})$  is usually referred to as the regularization term and it is finite for all  $\mathbf{x} \in R^N$ .  $p(\mathbf{x})$  is separable, if it satisfies

$$p(\mathbf{x}) = \sum_{i=1}^N p_i(x_i). \quad (9)$$

It is obvious that  $p(\mathbf{x}) = \|\mathbf{x}\|_1$  in (5) is separable, and  $f(\mathbf{x}) = \frac{1}{2}\|\mathbf{Ax} - \mathbf{y}\|_2^2$ . The algorithm solves the problem through a sequence of iterations  $\{\mathbf{x}^k, k = 0, 1, 2, \dots, n\}$ , where  $\mathbf{x}^{k+1}$  is obtained from the previous solution  $\mathbf{x}^k$  by optimizing the following sub-problem:

$$\begin{aligned} \mathbf{x}^{k+1} &= \min_{\mathbf{x}} f(\mathbf{x}^k) + (\mathbf{x} - \mathbf{x}^k)^T \nabla f(\mathbf{x}^k) \\ &\quad + \frac{1}{2}(\mathbf{x} - \mathbf{x}^k)^T \cdot \nabla^2 f(\mathbf{x}^k) \cdot (\mathbf{x} - \mathbf{x}^k) + \lambda p(\mathbf{x}) \\ &\approx \min_{\mathbf{x}} (\mathbf{x} - \mathbf{x}^k)^T \nabla f(\mathbf{x}^k) + \frac{\beta^k}{2} \|\mathbf{x} - \mathbf{x}^k\|_2^2 + \lambda p(\mathbf{x}) \\ &= \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{x} - \mathbf{v}^k\|_2^2 + \frac{\lambda}{\beta^k} p(\mathbf{x}) \end{aligned} \quad (10)$$

where the diagonal matrix  $\beta^k I$  is an estimate of the Hessian matrix  $\nabla^2 f(\mathbf{x}^k)$  and

$$\mathbf{v}^k = \mathbf{x}^k - \frac{1}{\beta^k} \nabla f(\mathbf{x}^k). \quad (11)$$

Now, the following equation can be obtained:

$$\begin{aligned} x_i^{k+1} &= \min_{x_i} \frac{1}{2} \|x_i - v_i^k\|_2^2 + \frac{\lambda}{\beta^k} p_i(x_i) \\ &= \min_{x_i} \frac{1}{2} \|x_i - v_i^k\|_2^2 + \frac{\lambda}{\beta^k} |x_i| \\ &= \text{soft}(v_i^k, \frac{\lambda}{\beta^k}) \end{aligned} \quad (12)$$

where

$$\begin{aligned} \text{soft}(u, a) &= \text{sgn}(u) \max\{|u| - a, 0\} \\ &= \begin{cases} \text{sgn}(u)(|u| - a) & \text{if } |u| > a \\ 0 & \text{otherwise} \end{cases} \end{aligned} \quad (13)$$

is the soft-thresholding function.

Different IST algorithms have different strategies for choosing the regularizing coefficient  $\lambda$  and the parameter  $\beta^k$ . For example, the regularizing coefficient is usually generated by a decreasing sequence of  $\{\lambda_k, k = 0, 1, \dots, n\}$  where  $\lambda_0 > \lambda_1 \dots > \lambda_n$  and  $\lambda_n \rightarrow 0$ . The parameter  $\beta^k$  in [38] is chosen by optimizing the so-called Barzilai–Borwein equation

$$\begin{aligned} \beta^k &= \min_{\beta} \|\beta(\mathbf{x}^k - \mathbf{x}^{k-1}) - (\nabla f(\mathbf{x}^k) - \nabla f(\mathbf{x}^{k-1}))\|_2^2 \\ &= \frac{(\mathbf{x}^k - \mathbf{x}^{k-1})^T (\nabla f(\mathbf{x}^k) - \nabla f(\mathbf{x}^{k-1}))}{(\mathbf{x}^k - \mathbf{x}^{k-1})^T (\mathbf{x}^k - \mathbf{x}^{k-1})}. \end{aligned} \quad (14)$$

The parameters have great effect on the performance of the algorithm. Especially for the regularizing coefficient  $\lambda$ , and a lot of strategies have been proposed to improve the performance of the algorithm by a continuation procedure in  $\lambda$  [7], [39].

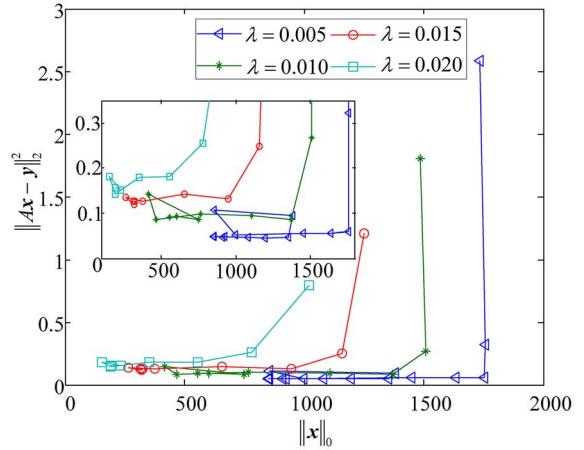


Fig. 2. Outputs of successive iterations of the soft-thresholding algorithm with different  $\lambda$  values. In the beginning of the sequences, there is more of a chance to obtain solutions that dominate the previous ones. As the iterations increase, solutions are getting closer to the optimal. If we consider that we can obtain different optimal solutions with different  $\lambda$  and these optimal solutions lie on the PF, then the solutions of successive iterations tend to get closer to the PF. The closer the solutions are to the PF, the higher probability there is to achieve solutions that are nondominated to the previous ones (see the enlarged graph).

2) *Incorporating Soft-Thresholding Into Evolutionary Algorithm:* We now discuss how to best choose values for  $\mathbf{x}^k$  and  $\mathbf{x}^{k-1}$ . In MOPs, it is important to ensure, first, that each solution converges to the Pareto-optimal set, while, second, also ensuring maintenance of a good spread of solutions over the final set. To do this, we need to look at how we could make the soft-thresholding method work as a local search strategy in our evolutionary algorithm. As shown in (10), an updated solution  $\mathbf{x}^{k+1}$  is computed by minimizing the linearized function of the previous solution  $\mathbf{x}^k$ . We can prove that  $\mathbf{x}^{k+1}$  always dominates or is nondominated to the previous solution  $\mathbf{x}^k$ , which means that  $\mathbf{x}^{k+1}$  is never worse than  $\mathbf{x}^k$  (Fig. 2).

*Theorem 1:* In the sequence of  $\mathbf{x}$ ,  $\{\mathbf{x}^k, k = 0, 1, 2, \dots, n\}$ , solution  $\mathbf{x}^{k+1}$  always dominates  $\mathbf{x}^k$ , or  $\mathbf{x}^{k+1}$  and  $\mathbf{x}^k$  are nondominated. We denote the relationship between  $\mathbf{x}^{k+1}$  and  $\mathbf{x}^k$  as  $\mathbf{x}^{k+1} \preceq \mathbf{x}^k$ .

*Proof:* The sequence of  $\{\mathbf{x}^k, k = 0, 1, 2, \dots, n\}$  is derived to minimize (8), which means

$$f(\mathbf{x}^{k+1}) + \lambda p(\mathbf{x}^{k+1}) \leq f(\mathbf{x}^k) + \lambda p(\mathbf{x}^k). \quad (15)$$

We assume that  $\mathbf{x}^{k+1} \neq \mathbf{x}^k$ . Then, we get

$$\begin{aligned} f(\mathbf{x}^{k+1}) + \lambda p(\mathbf{x}^{k+1}) &< f(\mathbf{x}^k) + \lambda p(\mathbf{x}^k) \\ \Rightarrow (f(\mathbf{x}^{k+1}) - f(\mathbf{x}^k)) + \lambda(p(\mathbf{x}^{k+1}) - p(\mathbf{x}^k)) &< 0. \end{aligned} \quad (16)$$

Because  $\lambda > 0$ , at least one of the equations in  $f(\mathbf{x}^{k+1}) - f(\mathbf{x}^k) < 0$  and  $p(\mathbf{x}^{k+1}) - p(\mathbf{x}^k) < 0$  holds, that is  $\exists f(\mathbf{x}^{k+1}) < f(\mathbf{x}^k)$  or  $p(\mathbf{x}^{k+1}) < p(\mathbf{x}^k)$ . Then, three cases are involved:

- a)  $\forall f(\mathbf{x}^{k+1}) \leq f(\mathbf{x}^k)$  and  $p(\mathbf{x}^{k+1}) \leq p(\mathbf{x}^k)$  hold, and  $\exists f(\mathbf{x}^{k+1}) < f(\mathbf{x}^k)$  or  $p(\mathbf{x}^{k+1}) < p(\mathbf{x}^k)$ , which means  $\mathbf{x}^{k+1}$  dominates  $\mathbf{x}^k$ , denoted as  $\mathbf{x}^{k+1} \prec \mathbf{x}^k$ ;
- b)  $\forall f(\mathbf{x}^{k+1}) < f(\mathbf{x}^k)$  and  $p(\mathbf{x}^{k+1}) > p(\mathbf{x}^k)$  hold, which means  $\mathbf{x}^{k+1}$  and  $\mathbf{x}^k$  are nondominated;

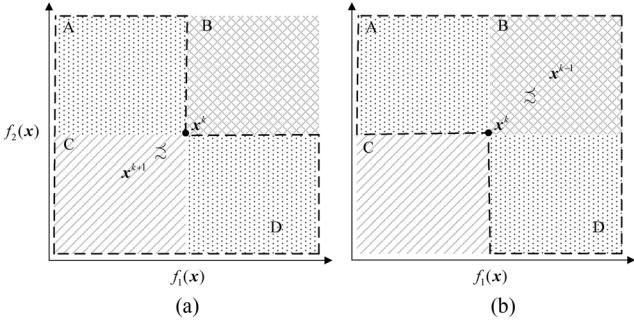


Fig. 3. Relationship between solutions in sequence  $\{x^k, k = 0, \dots, n\}$ . The relationship between  $x^k$  and its two nearest solutions  $x^{k+1}$  and  $x^{k-1}$  is used to illustrate the relationship between different solutions in the sequence. Solutions in area A (upper-left part of the graph) and area D (lower-right part of graph) are nondominated to  $x^k$ . Solutions in area C dominate solution  $x^k$ . Solutions in area B are dominated by  $x^k$ . (a) Dashed area (areas A, C, and D) shows the possible area for the updated solution  $x^{k+1}$ . (b) All possible areas (A, B and D) for the previous solution  $x^{k-1}$ .

- c)  $\forall f(x^{k+1}) > f(x^k)$  and  $p(x^{k+1}) < p(x^k)$  hold, which also means  $x^{k+1}$  and  $x^k$  are nondominated.

So  $x^{k+1} \precsim x^k$ .  $\blacksquare$

According to update (12), Theorem 1, and Figs. 2 and 3, we conclude that: 1) the updated solution  $x^{k+1}$  can be generated from two other solutions  $x^k$  and  $x^{k-1}$  ( $x^{k-1}$  is used to approximate the Hessian matrix of  $x^k$ ), and 2) the relationship of solutions in the sequence is

$$\{x^{k+1} \precsim x^k \precsim x^{k-1}, k = 1, \dots, n - 1\}. \quad (17)$$

In order to incorporate this soft-thresholding method into the evolutionary algorithm as a local search strategy, we need to select solutions from the current generation (generated by the evolutionary algorithm), which have the properties necessary for Theorem 1 to hold true. However, first, we need two solutions and then we need to make sure the relationship between the two solutions is in accordance with the solutions in the sequence of (17). It is easy to get two solutions in the current generation. Considering that the two solutions  $x^k$  and  $x^{k-1}$  for obtaining the updated solution  $x^{k+1}$  should be nondominated or  $x^k \prec x^{k-1}$ , and the nondominated and the dominated sets in the current generation are already calculated in the previous step of local search (Step 4 in Table I), we select the  $x^k$  and  $x^{k-1}$  through the following two methods (we shall refer to these two as case 1 and case 2, respectively, see Fig. 4). In case 1, we choose both  $x^k$  and  $x^{k-1}$  from the set of nondominated solutions [see Fig. 4(a)]. In case 2,  $x^k$  is chosen from the set of nondominated solutions but  $x^{k-1}$  is deliberately chosen from the set of dominated solutions [see Fig. 4(b)]. The reasons for these two ways to choose  $x^k$  and  $x^{k-1}$  are that, first we can make sure that  $x^k \precsim x^{k-1}$  and no extra computation time is needed to calculate the domination relationship between  $x^k$  and  $x^{k-1}$ .

Now, we discuss how these two cases are used in our algorithm. There is more chance to obtain solutions that dominate the previous ones in early iterations than in later iterations because the start points may not be very good, which means that there is higher probability to improve both the objective functions (see Fig. 2), so the relationship between the updated

TABLE II  
PROCEDURE OF THE SOFT-THRESHOLDING LOCAL SEARCH

<i>Local search procedure</i>
<b>If</b> $n_{dominated} \neq 0$
<b>If</b> $n_{non\_dominated} \geq popsize/2$ and $mod(t, 2) = 0$
Split the current non-dominated set into two equal sub-sets in number of solutions randomly.
<b>For</b> each of the solution $x^k$ in one of the sub-sets
Execute case 1. $x^{k-1}$ is selected randomly from the other sub-set. Obtain the new solution $x^{k+1}$ .
<b>EndFor</b>
<b>For</b> each of the solution $x^k$ in the other sub-sets
Execute case 2. $x^{k-1}$ is selected randomly from $P_{dominated}^t$ . Obtain the new solution $x^{k+1}$ .
<b>EndFor</b>
<b>Else</b>
<b>For</b> each solution $x^k$ in the non-dominated set
Execute case 2. $x^{k-1}$ is randomly selected from the dominated set $P_{dominated}^t$ . Obtain the new solution $x^{k+1}$ .
<b>EndFor</b>
<b>EndIf</b>
<b>Else</b>
<b>For</b> each of the solution $x^k$ in the $P_{non\_dominated}^t$ .
Execute case 1. $x^{k-1}$ is selected randomly from the left solutions in $P_{non\_dominated}^t$ . Obtain the new solution $x^{k+1}$ .
<b>EndFor</b>
<b>EndIf</b>

and the previous solutions are usually  $x^{k+1} \prec x^k$  ( $x^{k+1}$  lies in the area C in Fig. 3) also  $x^k \prec x^{k-1}$  ( $x^{k-1}$  lies in the area B in Fig. 3), and more nondominated solutions are generated in the early stage of the iterations. In contrast, there are also many dominated solutions and even the nondominated solutions are not the global optimal solutions in the early evolutionary stage. So the properties of the solutions in the early stage of evolutionary method correspond with the properties of the solutions in early iterations of the soft-thresholding method, and that is the reason why case 2 is applied (see the first *Else* in Table II) to obtain more nondominated solutions, thus improving the latest estimate of the *PF* and pushing the latest generation of individuals rapidly toward the true *PF* (as more solutions  $x^{k+1}$  in the bottom-left part in Fig. 2 or area C in Fig. 3 are obtained). There will be more and more nondominated solutions in the population with the increase in generations, as the selection operator prefers the nondominated individuals and our aim is to get the *PF*. Then there is a greater chance to apply case 1 (see the second *If* and second *Else* in Table II). We argue that there is higher probability for case 1 to obtain updated solutions that are nondominated relative to the previous ones. Assume that the current population is full of nondominated solutions and that they lie on the true *PF*. There is no chance to get  $x^{k+1}$  which dominates  $x^k$  because  $x^k$  is one of the optimal nondominated solutions. However, since  $x^{k+1} \precsim x^k$  from Theorem 1, we can deduce that  $x^{k+1}$  is nondominated to  $x^k$  ( $x^{k+1}$  lies in areas A and D in Fig. 3). So there is a greater chance for case 1 to disperse the solutions on the *PF* and higher probability for case 2 to get improved nondominated solutions.

In our modified procedure of local search, case 1 or case 2 are executed depending on whether there are dominated solutions in the current population and the proportion of the nondominated solutions. When the nondominated set only

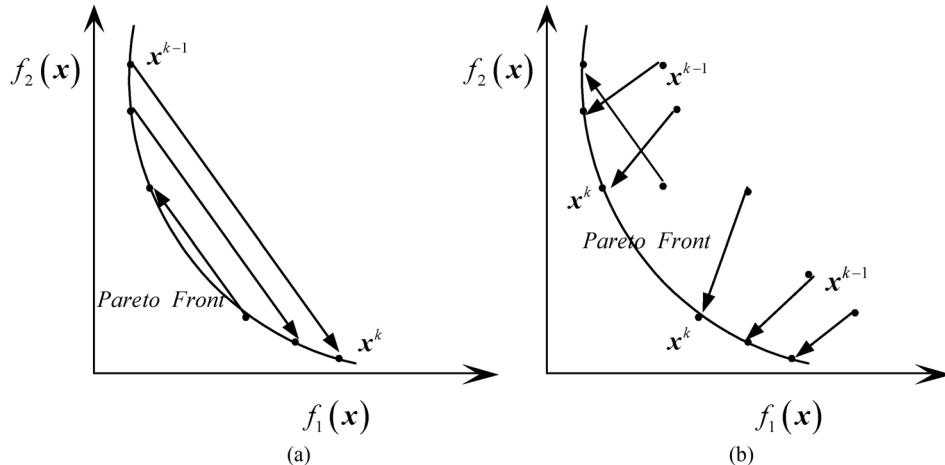


Fig. 4. Graphical explanation for local search under two cases. (a) Both  $x^k$  and  $x^{k-1}$  are selected from the nondominated set with greater chance to obtain solutions, which tend to spread solutions along the PF. (b)  $x^k$  is from the nondominated set while  $x^{k-1}$  is selected from the dominated set with higher probability to obtain updated solutions which dominate the current population, thus converging rapidly toward the true PF.

TABLE III  
PROCEDURE OF THE SELECTION OPERATOR

<b>Selection procedure</b>	
<b>Input:</b>	$P^t$ (the current population)
	$P_{crossover}^t$ (sub-population after the crossover operator)
<b>Do:</b>	$P_{local\_search}^t$ (sub-population obtained through local search) Combine the three input populations and denote the combined population by $P_{cmb\_pop}^t$ . Select from $P_{cmb\_pop}^t$ using kNNL-based diversity maintenance [35] and obtain the population $P^{t+1}$ for the next generation.
<b>Output:</b>	$P^{t+1}$ (the next population)

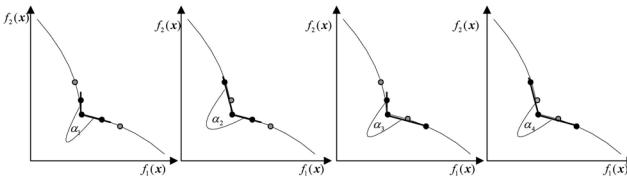


Fig. 5. Angle-based method.  $\alpha_1 \sim \alpha_4$  are the four possible angles between a solution and its four neighbors. The maximum of these four angles is calculated to represent the tradeoff for this solution. Then, the solution with maximum tradeoff is taken to denote the knee region.

contains less than half of the total solutions, case 2 is executed to generate more nondominated solutions. When the current population is entirely composed of nondominated individuals, case 1 is applied to obtain a better spread on the nondominated set. Otherwise, case 1 or case 2 are applied to achieve both better convergence and better spread.

The detailed procedure of local search is summarized in Table II.  $n_{dominated}$  is the cardinality of  $P_{dominated}^t$ .  $t$  denotes the  $t$ th generation in the evolutionary process.  $n_{non\_dominated}$  counts solutions in the nondominated set. Parameter  $popsize$  is the size of the current population. New solutions  $x^{k+1}$  are obtained by equations expressed in (12), (13), and (14).

This procedure is useful because it guarantees improvement in two different senses: one is an improved spread within the set of nondominated solutions and another is a tendency to move from dominated toward nondominated solutions. Both of

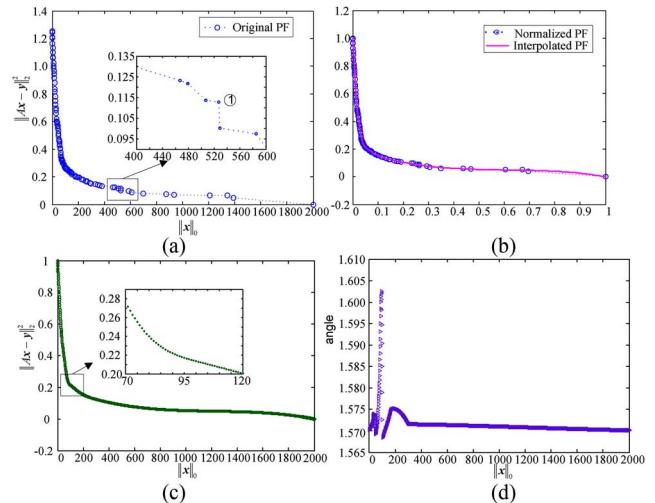


Fig. 6. Illustration of how solutions, which are not the optimal nondominated ones, affect the process for finding knee points. The PF curve is not as smooth as it looks. It is evident that solutions at the enlarged figure (top left) would disturb the angle-based method to find the knee area. The top right plot demonstrates the process that smoothes the PF by interpolation. The bottom left figure shows the process to sample the interpolated PF uniformly and calculate the angles based on PF obtained by sampling. This paper takes four neighbors into account when calculating the angles. The  $x$ -axis is enlarged into the original range for a better view in the bottom plots. (a) Original PF before interpolation. (b) Normalizing PF and interpolating PF with B-splines. (c) Sampling interpolated PF with uniform interval on  $x$ -axis. (d) Angle information based on sampled points.

these improvement directions are of importance in obtaining optimal nondominated solutions on the PF.

We now refer the reader to Appendix A, which presents a set of experiments and data to demonstrate the advantages of using case 1 and case 2 methods for choosing  $x^k$  and  $x^{k-1}$  over random choice during local search.

### B. Selection Operator

In this paper, we adopt a selection operator, which prunes solutions in the combined population  $P_{cmb\_pop}^t$ . Many crowding distances measurement and diversity maintenance

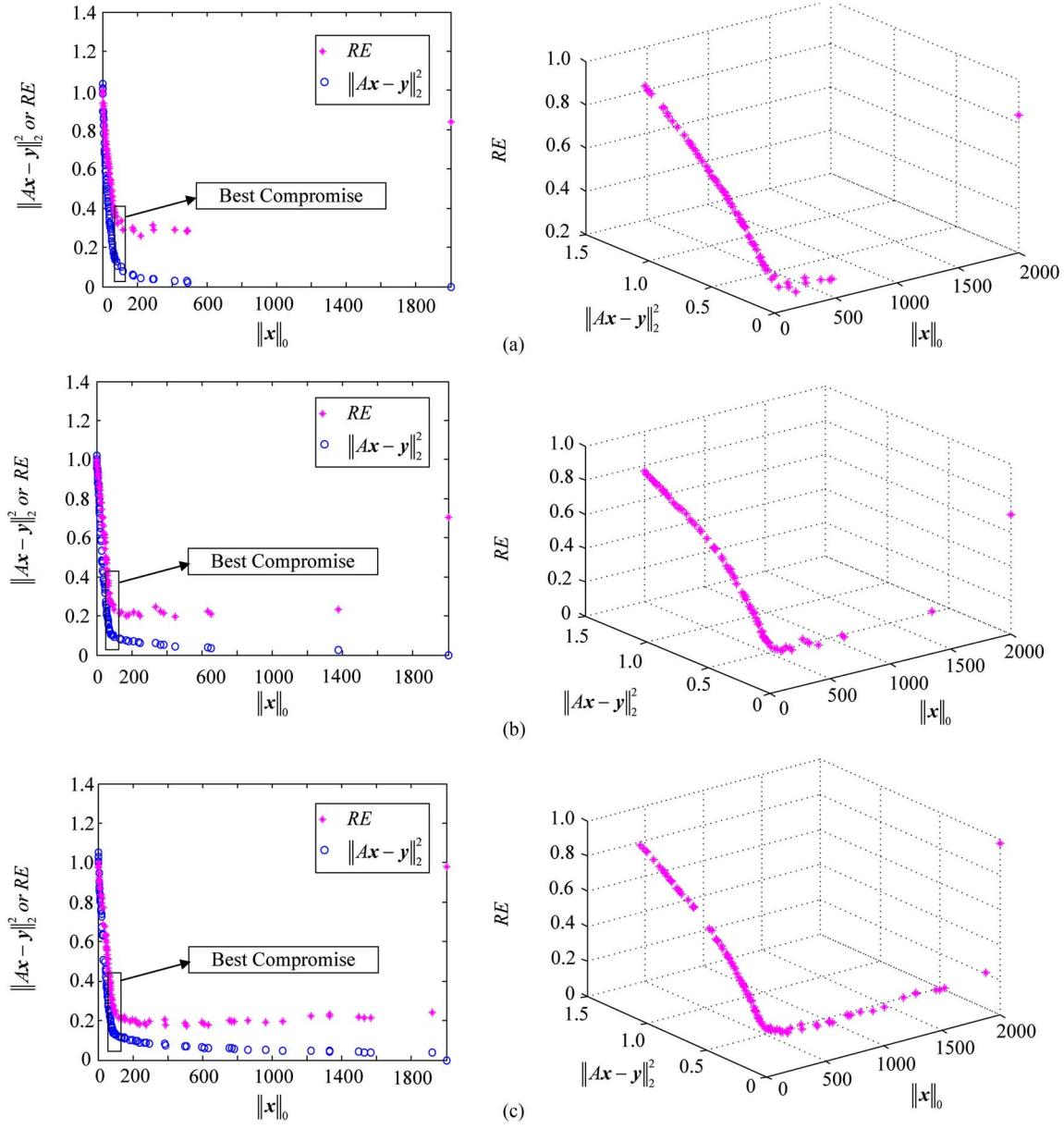


Fig. 7. Connection between reconstruction error, measurement error, and the sparsity of the solutions on *PF*, and the best compromise region between measurement error and the sparsity of the solutions with different dimension in the Gaussian random projection. (a)  $M = 600$ . (b)  $M = 1200$ . (c)  $M = 1800$ .

methods have been proposed to improve the performance of NSGA-II, such as harmonic average distance strategy in [40]. Here, diversity maintenance based on k-nearest neighbor list (kNNL) [35], [34], is used to select solutions for the next generation. This selection procedure is described in Table III.

### C. Finding Knee Areas

Knee regions consist of solutions for which a small improvement in one objective would cause a severe deterioration in another. These are usually the most interesting regions along the *PF* in MOPs. Two methods, for locating the knee regions, are proposed in [29]. Because the model in this paper is a bi-objective optimization, we can use an angle-based method [29] for seeking the knee points on the *PF* and four points are considered in this paper (see Fig. 5).

In the angle-based method, straight lines are drawn to connect a solution with each of its nearest neighbors on the

*PF*. The gradients of these lines represent the tradeoffs in each objective. As shown in Fig. 5, four neighbors, which are the closest and second closest either side of an individual, are taken into account and thus four angles ( $\alpha_1 \sim \alpha_4$ ) are computed. The maximum of these four angles is considered the tradeoff of the solution. The knee regions then can be obtained by comparing the tradeoffs of all solutions along the *PF*, to find that solution with maximum tradeoff angle.

Note that the problem being tackled in this paper is more difficult than the problems described in [29]. The first difficulty is that because the problem is NP-hard, it is hard to ensure that the estimated *PF* produced by the algorithm converges to the true *PF*. If some of the solutions, of a population, do not truly lie on the *PF* then they can distort the calculations used to find the knee region, leading to an incorrect estimate of the globally optimal solution (solutions in the global knee region). As shown in the enlarged view of Fig. 6(a), the point ① is

TABLE IV  
OBTAINING KNEE POINTS

<i>Procedure for finding the knee region on a PF</i>	
<b>Input:</b>	Initial estimate of $PF$ - the set of solutions from the converged evolutionary algorithm.
<b>Do:</b>	Normalize $PF$ . Fit B-splines to $PF$ to generate a smoother curve $PF'$ . Sample $PF'$ evenly. Find knee point on $PF'$ using the angle-based method.
<b>Output:</b>	Knee points on $PF'$ .

not an optimal nondominated solution and it will disturb the process of finding knee areas. The second difficulty is that two objective functions  $\|x\|_0$  and  $\|Ax - y\|_2^2$  are involved in the  $PF$ , but the ranges of these two functions usually have greatly different magnitudes. This can diminish the significance of the differences in tradeoff angles between neighboring solutions. The third difficulty is that it is not easy to obtain a set of solutions that adequately sample the full range of the  $PF$  because the population size is fixed.

In order to overcome these difficulties and achieve accurate estimates of knee regions, first we normalize the  $PF$  by its maximum  $\|Ax - y\|_2^2$  value and  $\|x\|_0$  value; thus, avoiding the second difficulty mentioned above. Next, we perform smoothing by interpolating the  $PF$  using B-splines and then evenly resampling from the smooth spline to deal with the first and third difficulties. Finally, the knee points can be found from this interpolated curve  $PF'$  using the angle-based method [29]. The procedure for finding the knee regions is summarized in Table IV. After finding the knee region on  $PF'$ , we can estimate the knee area on the original  $PF$  by finding the point on  $PF$  that is closest to the knee region of  $PF'$ .

### III. EXPERIMENTS AND DISCUSSION

Our experiments fall into two parts. Section III-A presents data that demonstrate the existence of knee regions in sparse reconstruction problems, and show that these provide solutions which best satisfy the tradeoff between enforcing sparsity and reducing measurement error. Additionally, we use the example of an image processing problem to show the effect of choosing different solutions along a particular  $PF$ . Section III-B compares the performance of StEMO against five other algorithms from the literature, which are commonly used for sparse reconstruction.

#### A. Existence of Knee Areas and Best Compromise Between Two Conflicting Objectives

To demonstrate the existence of a knee area for this problem, four sets of experiments have been conducted. Graphs and statistical box plots are given to show the existence of the knee region, and the factors that affect the position of the knee area are also analyzed. Then, an application to a sparse image recovery problem is given to show the different reconstructions of images corresponding to different solutions on a  $PF$ .

In these experiments, simulations were conducted with random sparse signals. The ground-truth signal  $x_{tr}$  was generated as follows: 1) a subset of nonzero coefficients is chosen randomly and the cardinality of the subset represents

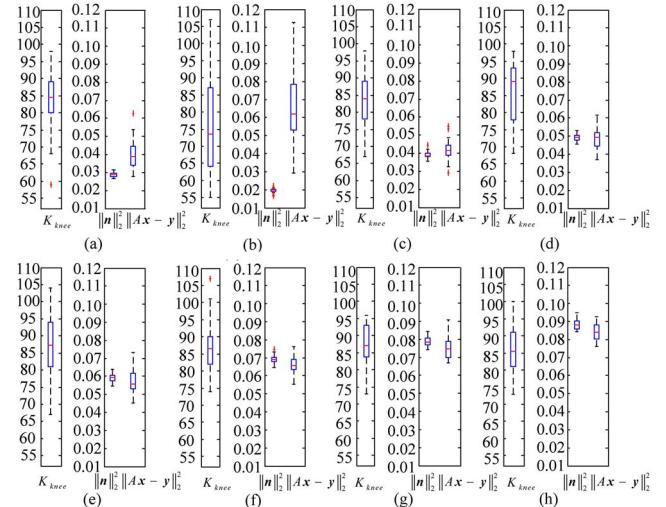


Fig. 8. Box-plots for various different values of  $M$ . For each projection dimension, three box-plots are shown: 1) the zero-norm of the solution at the knee area; 2) the measurement error; and 3) norm of the additive white noise. Each box-plot is generated by running 30 independent trials. (a)  $M = 400$ . (b)  $M = 600$ . (c)  $M = 800$ . (d)  $M = 1000$ . (e)  $M = 1200$ . (f)  $M = 1400$ . (g)  $M = 1600$ . (h)  $M = 1800$ .

the total nonzero elements in  $x_{tr}$ ; 2) their values are drawn from the standard normal distribution; and 3)  $x_{tr}$  is normalized to have unit length.

The measurement matrix  $A$  is a Gaussian random matrix with coefficients randomly generated from the standard normal distribution. The measurement vector  $y$  is corrupted by white noise  $n$ . Elements in  $n$  are generated from a normal distribution with mean 0 and a standard deviation, which takes different values in different experiments. The length of the signal is 2000. The *popsize* is set to 100 and the maximum generation is 120 in this experiment. The mutation rate is 0.1 and the crossover rate is 0.5.

1) *Effect of Projection Dimension on Knee Region:* This section describes an experiment to investigate which factors influence the position of knee areas. The experiment reveals connections between the dimension of the Gaussian random projection and the knee areas of the sparse model. The size of the nonzero elements  $K$  in the ground-truth signal is 100. Eight different cases are investigated, where the dimension ( $M$ ) of the Gaussian random projection is varied from 400 to 1800, in intervals of 200. Thirty independent trials were undertaken for each value of  $M$ .

Fig. 7 graphs results for three of the eight test cases where the dimension ( $M$ ) of the Gaussian random projection is set to 600, 1200, and 1800, labeled Fig. 7(a), (b), and (c), respectively. In each case, the right-hand graph is a 3-D plot, graphing the relationship between the measurement error  $\|Ax - y\|_2^2$ , the sparsity of the signal, and the reconstruction error. The left-hand graph shows two 2-D views of the data; variation of  $\|Ax - y\|_2^2$  (circle data points) and reconstruction error (star data points) with change in sparsity. Reconstruction error (RE) is the average difference between  $x$  and the ground-truth signal. Each graph of Fig. 7 shows results for one example trial; space does not permit showing 30 examples of each graph for all 30 trials of all eight  $M$  values, but the

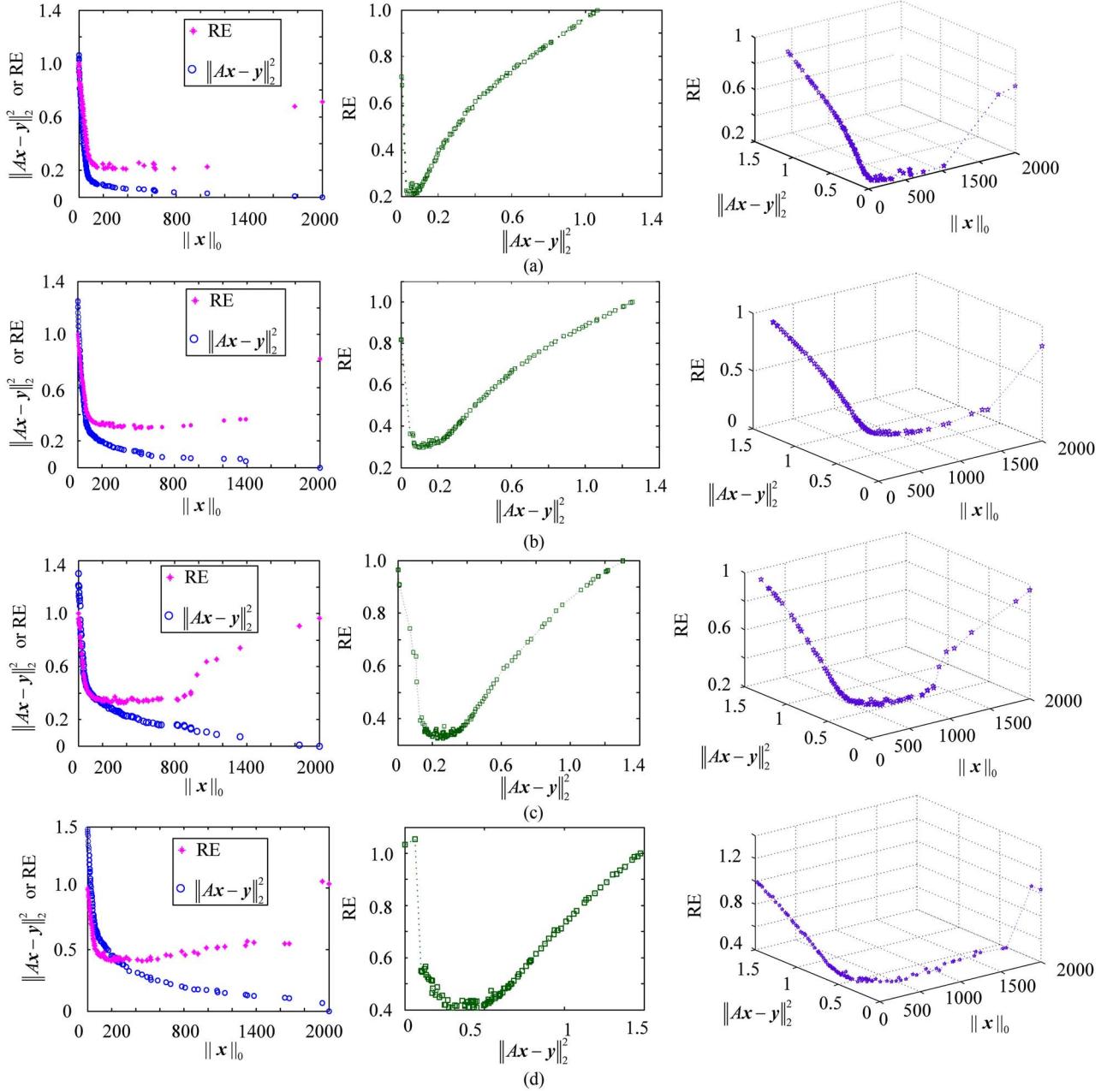


Fig. 9. Relationships between reconstruction error, measurement error, and the sparsity of the solutions on  $PF$ , for four different magnitudes of additive noise. Left-hand figures show variation of measurement error and  $RE$  with change in sparsity, middle figures show the change of  $RE$  with measurement error, and right-hand figures show 3-D plot of these three terms. (a)  $\delta = 0.008$ . (b)  $\delta = 0.012$ . (c)  $\delta = 0.016$ . (d)  $\delta = 0.020$ .

graphs shown are typical of the data and usefully illustrate important observed trends.

Fig. 8 uses box-plots to show statistical results of the change in knee regions with varying numbers of projection dimensions. For each value of projection dimension, three box-plots of the 30 trials are presented, showing: 1) the zero-norm of the solution at the knee area, denoted  $K_{knee}$ ; 2) the measurement error  $\|Ax - y\|_2^2$ ; and 3) the norm of the additive white noise, denoted by  $n_2$ .

From Fig. 7, we can observe some important and useful properties: 1) for all values of  $M$ , the  $PF$  exhibits an obvious knee region; 2) the horizontal position of the knee area (i.e., the correspondence with sparsity) does not appear to vary much for different numbers of dimensions  $M$  typically

corresponding to an  $\|x\|_0$  value of around 90; 3) these knee areas provide an optimal solution to the sparse reconstruction problem, because any further attempt to decrease reconstruction error will result in rapid deterioration in sparsity for only very small increases in accuracy.

From Fig. 8 we can again see that the zero-norm of the solutions at the knee point are consistently about 90, and do not change appreciably with different numbers of projection dimensions.

The second and third box-plots in each subfigure of Fig. 8 show the squares of  $l_2$ -norm of the added noise  $n_2^2$  and the reconstruction error  $\|Ax - y\|_2^2$ . It can easily be shown that if these two terms are equal to each other, and the zero-norm of the corresponding solution coincides with the ground-truth

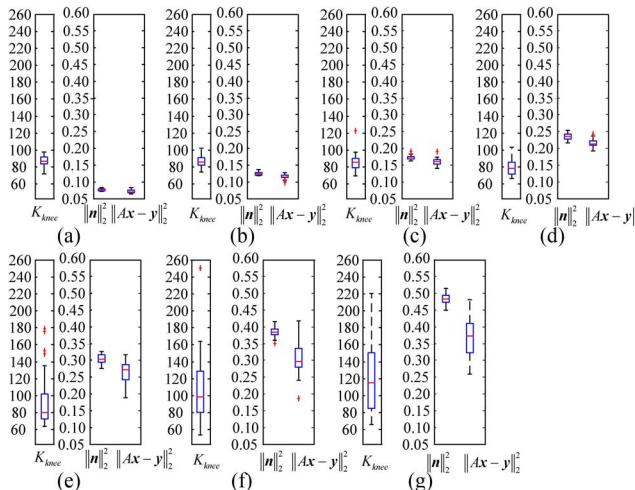


Fig. 10. Box-plots for various different values of additive white noise, standard deviation varies  $\delta = 0.008 \sim 0.02$ . For each value of noise, three box-plots are shown: 1) the zero-norm of the solution at the knee area; 2) the measurement error; and 3) norm of the additive white noise. Each box-plot is generated by running 30 independent trials. (a)  $\delta = 0.008$ . (b)  $\delta = 0.10$ . (c)  $\delta = 0.12$ . (d)  $\delta = 0.14$ . (e)  $\delta = 0.16$ . (f)  $\delta = 0.18$ . (g)  $\delta = 0.020$ .

signal, then one can exactly reconstruct the signal from the observation vector despite it being polluted by noise. Unfortunately, in sparse reconstruction problems,  $n_2^2$  and the zero-norm of the original signal are typically unknown; however, Figs. 7 and 8 suggest a useful way of approximating these important parameters:  $\|Ax - y\|_2^2$  can be used to approximate  $n_2^2$ , and  $\|\mathbf{x}\|_0$  can be estimated as the position of the knee region,  $K$ .

Note that  $n_2^2$  increases with the growth in the projection dimension  $M$ . This is because, with growing  $M$ , there are more elements in the original signal which can be corrupted by noise during the observation process. Measurement error can be used as an approximation of the noise. The better this approximation is, the more precisely the original signal can be estimated. Fig. 8 shows that the measurement error is very close to the noise when  $M$  is in the range of 800–1400, suggesting that the approximation can be used to generate accurate reconstructions in this region. As shown in the reconstruction error curves of Fig. 7, the best approximation to the ground-truth signal can be produced in this range of  $M$ . Note that, while the position of the knee with respect to sparsity (horizontal axis in Fig. 7) is relatively independent of  $M$ , the position of the knee with respect to both measurement error and reconstruction error (vertical axis in Fig. 7) does vary significantly with different values of  $M$ .

2) *Effect of Noise on Knee Region:* We now explore how the position of the knee area varies with the standard deviation of measurement noise. In this experiment, the size of the Gaussian observation matrix  $A$  is fixed at  $1200 \times 2000$ . The number of nonzero elements is set to 100. Noise is simulated by adding a normally distributed random variable to each element of the ground-truth observation matrix, such that each random variable is drawn from the distribution  $N(0, \delta^2)$ . We have gathered data for seven different cases where  $\delta$  varies from  $0.008 - 0.02$  in increments of size 0.002. For each value of  $\delta$ , we performed 30 independent trials, with new values for

additive noise being randomly generated for each trial, so that each trial uses different noise values.

Fig. 9 shows the relationship between the reconstruction error, the measurement error, and the number of nonzero elements in the corresponding solutions. Axes in Fig. 9 represent the same variables as in Fig. 7. The left-hand column of graphs shows 2-D plots of the variance of measurement error and the RE with change in sparsity. The middle column of graphs depict the change in the reconstruction error with  $\|Ax - y\|_2^2$ . The right-hand graphs provide 3-D views of the three variables together.

The graphs of Fig. 9 reveal several important trends: 1) it is evident that a knee region does exist for these problems, for all levels of noise; 2) the PF becomes smoother with increasing levels of noise, i.e., the knee becomes less sharp and pronounced; 3) the position of the knee along the horizontal (sparsity) axis does not substantially vary with changing noise levels, remaining approximately constant in the region of 90–100; 4) the measurement error does vary with noise magnitude (because the measurement error is an estimate of noise magnitude, as discussed earlier); and 5) the knee region represents the best compromise between  $\|Ax - y\|_2^2$  and  $\|\mathbf{x}\|_0$ , i.e., beyond this point very little additional reconstruction accuracy can be gained while the sparsity constraint becomes rapidly violated. Thus, the knee region provides the optimal estimate of the original signal that can be achieved by a sparse model.

Fig. 10 shows box-plots of  $K_{knee}$ ,  $\|n_2^2\|$ , and  $\|Ax - y\|_2^2$  for each of the seven different noise levels. These plots suggest that, for a given sparsity, the position of the knee region does not vary appreciably with different noise levels.

This experiment suggests that: 1) horizontal position of knee regions is independent of the magnitude of noise; 2) the sparsity of solutions at the knee region reflects the sparsity of the ground-truth signal; 3) the measurement errors of solutions at knee regions approximate the norms of the noise; and 4) solutions at the knee region are those that best approximate both the sparsity of the original signal and the noise.

3) *Effect of Sparsity on Knee Region:* In this section, we describe an experiment to investigate the influence of the sparsity of the ground-truth signal on the position of knee regions. The size of the Gaussian observation matrix  $A$  is fixed at  $1200 \times 2000$ . The standard deviation of the noise is fixed at 0.01. We examine four different cases, where the number of nonzero coefficients in  $\mathbf{x}_{tr}$  varies from 100 to 400 in increments of 100. For each case, 30 independent trials were performed. For each trial, the locations of nonzero elements were randomly chosen and their values were randomly sampled from a normal distribution. As it is harder to solve zero-norm sparse models with the increase in  $K$ , *popsize* is increased to 200 and the maximum generation is 200 for better performance and a better description of the PF.

Fig. 11 plots the relationship between  $\|Ax - y\|_2^2$ ,  $\|\mathbf{x}\|_0$ , and the reconstruction error when the sparsity of the ground-truth signal changes. It is found that the horizontal position (sparsity axis) of knee points varies with different values of  $K$ . The zero-norm of solutions in the knee areas are consistent with the true sparsity of the original signal, suggesting that the knee areas provide good solutions. Since the sparsity estimate

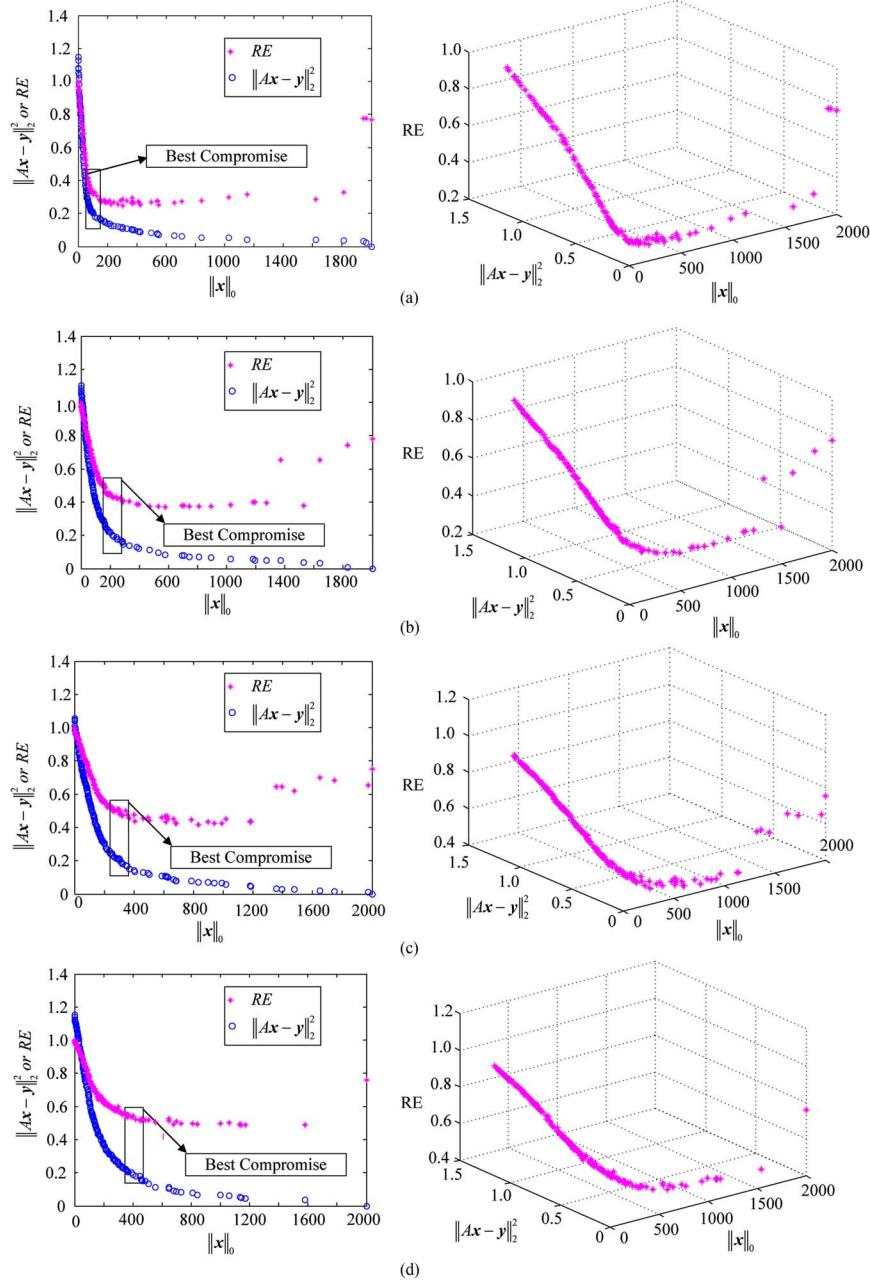


Fig. 11. Relationship between reconstruction error, measurement error, and the sparsity of the solutions on  $PF$  and the position of knee regions.  $K$  denotes the size of nonzero elements in  $\|x\|_0$ . (a)  $K=100$ . (b)  $K=200$ . (c)  $K=300$ . (d)  $K=400$ .

rapidly deteriorates for points to the right of the knee region, while providing very little improvement in reconstruction error, it is clear that the knee areas provide optimal solutions.

Fig. 12 shows statistical data for the zero-norm of the solutions and measurement errors at knee areas. These plots show that  $K_{knee}$  provides a good estimate of the zero-norm of the true signal, but also suggests that the accuracy of this estimate deteriorates as  $K$  increases.

4) *Application of StEMO to Reconstructing Noise-Corrupted Images:* This section demonstrates the efficacy of the StEMO algorithm with a practical application to a fundamental image processing problem. Fig. 13(a) shows a ground-truth image of a white circle against a black background. This can be represented by a set of Haar

wavelets [41] as shown in Fig. 13(c). The StEMO algorithm is tasked with reconstructing the original image from a noise-corrupted image [see Fig. 13(b)] with corresponding Haar wavelets [see Fig. 13(d)]. In this experiment, the original image was corrupted with additive Gaussian white noise of zero mean and 0.05 variance. Sampling the wavelet coefficients of the noisy image by a Gaussian random matrix  $A$  with  $M = 3/4N$ , where  $N$  is the length of the signal, we generate the measurement vector  $y$ . Then, StEMO is used to reconstruct the truth-ground signal from  $y$ .

Fig. 14 shows the  $PF$  solutions, generated by StEMO. Examples of reconstructed images, and corresponding Haar wavelets, are shown for the knee solution, and solutions to the left and right of the knee, which are overly sparse or

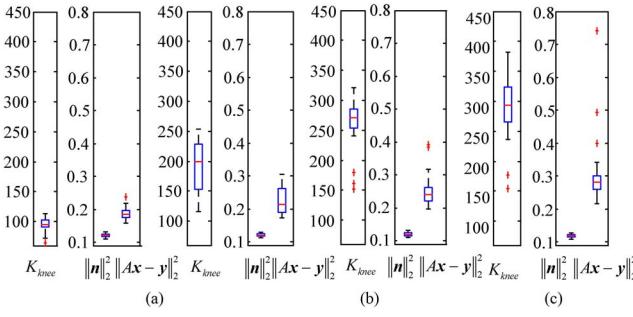


Fig. 12. Box-plots of  $K_{knee}$ ,  $\|n\|_2^2$  and  $\|Ax - y\|_2^2$ , for four different values of  $K$ , where  $K$  is the number of nonzero elements in the ground-truth signal. Thirty independent trials were carried out for each value of  $K$ . (a)  $K = 100$ . (b)  $K = 200$ . (c)  $K = 300$ . (d)  $K = 400$ .

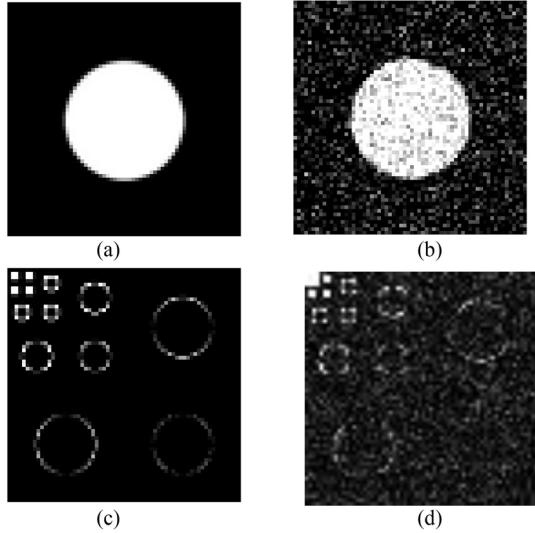


Fig. 13. Sparse image example. (a) Original image. (b) Image corrupted by noise. (c) Wavelet coefficients of the original image. (d) Wavelet coefficients of the corrupted image.

under-sparse, respectively. Clearly, the knee solution results in a reconstructed image, which is closest to the ground-truth original.

As shown in Fig. 14, the solutions on the upper part of the *PF* are sparser but with larger measurement errors. The images show an intuitive meaning of this tradeoff; although these solutions offer a superior amount of denoising (visible as a despeckling effect in the reconstructed image) they cannot accurately reconstruct the shape of the original signal. In contrast, solutions on the lower part of the *PF* provide a truer reconstruction of the original image shape, but are much more severely corrupted by noise (visible in the image as salt and pepper type noise). Clearly, the solution at the knee offers the best compromise. It reconstructs the original image better than solutions to the left of the knee, while providing a better denoising effect than the solutions to the right of the *PF*.

5) *Summary of Observations:* The main observations of these experiments are that: 1) a knee region exists on *PF* of the sparse reconstruction model; 2) solutions in the knee region represent the best estimate of the original signal; and 3) the position of the knee area on the  $\|x\|_0$  axis is closely related to the sparsity of the original signal and its position on the  $\|Ax - y\|_2^2$  and RE axes are related to the observation noise and projection dimensions.

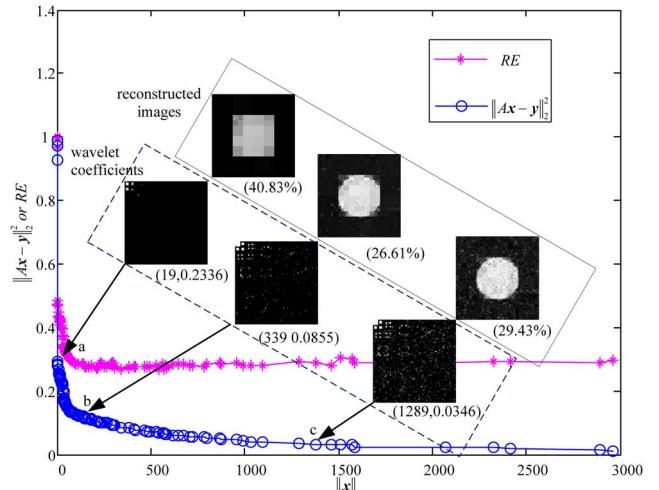


Fig. 14. Variation of reconstruction error and measurement error, for varying amounts of sparsity. Point a represents a solution that has little noise but is overly sparse (poorly reconstructing the shape). Point c represents a solution which is not very sparse (uses a large number of wavelets to give a good shape estimate) but suffers from a large amount of noise. Point b, in the knee region, represents the optimal tradeoff.

### B. Comparison of StEMO Against Other Methods

In this section, we demonstrate the benefit of the multiobjective approach, by comparing the performance of the multiobjective StEMO algorithm against five single objective approaches from the literature. The five single objective algorithms are OMP [10], basis pursuit (BP) [42], homotopy method [15], [16], IST [17], [43], and alternating direction method (ADM) [44]. Toolboxes for these methods can be found in [45]. These methods have been chosen because they are popular choices in the literature for solving sparse reconstruction problems.

1) *Simulation Experiments:* In this experiment, a ground-truth signal was generated in the same way as in Section III-A with a length of  $N = 2000$ . The performance of StEMO and the five comparison methods are shown in Fig. 15. In the left-hand graph, performance is plotted against increasing projection dimension, and in the right-hand graph against increasing sparsity ratio. In all experiments, the measurement vector is corrupted by additive noise with elements distributed as  $N(0, 0.01)$ . Each data point on each graph is the average error obtained from 20 independent runs of each algorithm. Values of tolerance  $\sigma = 0.3$  and  $\lambda = 0.02$  were used for the comparison algorithms.

In Fig. 15(a), the sparsity ratio was fixed at  $k/n = 0.1$  while the dimension of the measurement matrix was varied over the range  $d = 800$ – $1800$ . StEMO outperforms other algorithms, obtaining lower average estimation errors, except for OMP in the projection dimension range of 1200–1600. Note that the OMP method obtains good solutions here because the parameter  $\sigma$  greatly affects the performance of OMP, and the chosen value  $\sigma = 0.3$  for these experiments happens to be the optimal choice for this example, but might perform poorly on other examples.

In Fig. 15(b), the sampling rate is fixed where  $n=2000$  and  $d=1200$ , and the sparsity ratio changes from 0.1 to 0.5. The reconstruction error increases for all methods as the

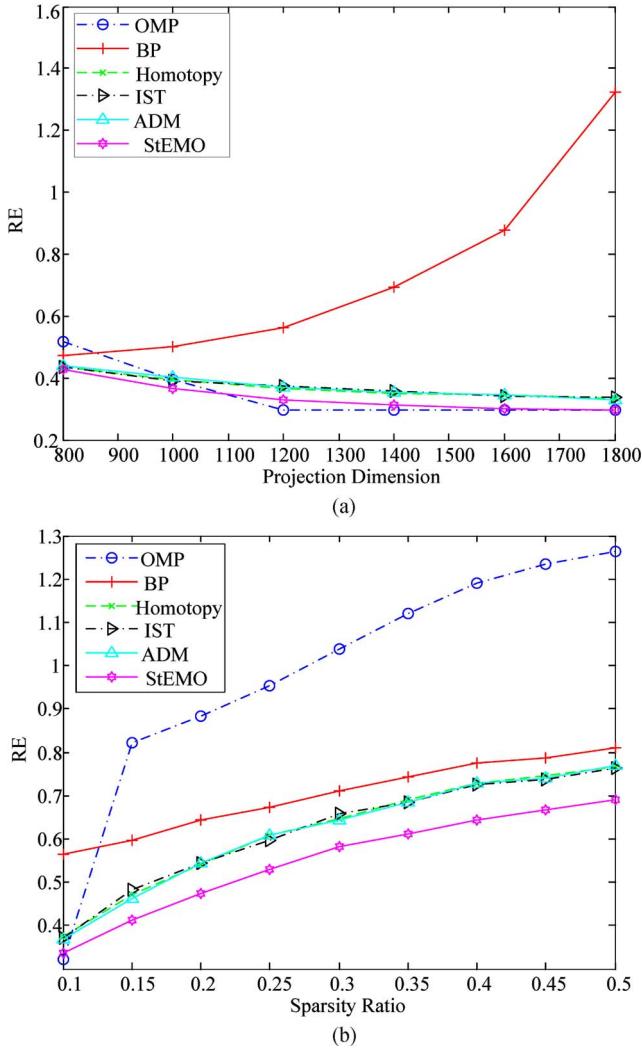


Fig. 15. Comparisons of StEMO against other algorithms. (a) Varying projection dimensions  $d = 800\text{--}1800$ . (b) Varying sparsity ratio  $k/n = 0.1 \sim 0.5$ .

sparsity ratio gets larger, but StEMO consistently obtains lower average estimation errors than the other algorithms. These two experiments show that the proposed method can produce better performance than well known single objective methods.

2) *Benchmark Problems:* The toolbox Sparco [46] contains a suite of problems for sparse signal reconstruction. Five signals (sgnspike, gausspike, cosspike, gcoospikes, and jitter) from Sparco are used to compare the proposed StEMO algorithm against five other algorithms taken from the literature. Sparse signals, such as sgnspike and gausspike, shown in Fig. 21, can be sampled by a measurement matrix  $A$  and recovered by  $\mathbf{y} = A\mathbf{x}$ . While signals such as cosspike, gcoospikes, and jitter (see Figs. 24, 26, and 28) are not sparse themselves, they have a sparse representation  $\mathbf{x}$  with respect to a set of basis vectors  $B$ , i.e.,  $f = B\mathbf{x}$ , where  $\mathbf{x}$  is sparse in its coefficients. If we sample such signals  $f$ , by a measurement matrix,  $A$ , then the observed vector  $\mathbf{y} = Af = AB\mathbf{x}$ . Let  $M = AB$ , then  $\mathbf{y} = M\mathbf{x}$ . Once one recovers  $\mathbf{x}$ , one can obtain  $f$  from  $f = B\mathbf{x}$ . Therefore, for these cases (see Figs. 25, 27, and 29) we show both the reconstructed coefficients as well as the actual signals. The tolerance  $\sigma$  is set to 0.1 and  $\lambda$  for the single-objective algorithms. Figs. 21–29 give the original signals and

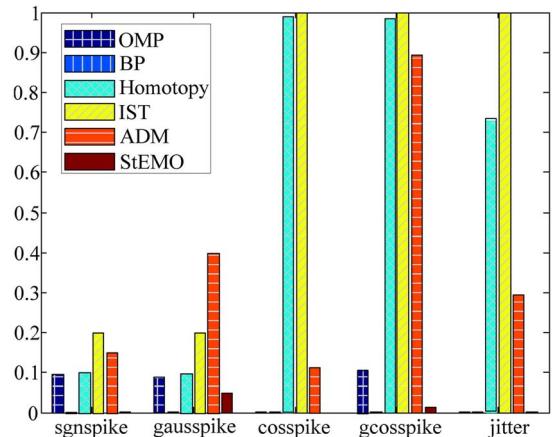


Fig. 16. Average reconstruction errors of the six algorithms on the five problems.

the reconstruction results produced by each algorithm for the five benchmark problems.

Fig. 16 gives the average reconstruction errors of the six methods on the five problems. All of the six algorithms show good performances on the cosspike problem. For gcoospikes, all algorithms except IST and ADM can recover the signal. OMP, BP, and StEMO reconstructed the coefficients and signals of sgnspike, gausspike, and jitter better than other algorithms. StEMO and BP show the best performances over all of the five problems. Note that BP works very well on these five problems and it obtained even less average error than StEMO on gausspike problem. On one hand, this may be because BP benefits from being given optimal parameter values in this experiment. However, in general it is usually not possible to know the optimal choices of BP parameters, when solving many problems. In contrast, BP did not outperform the other algorithms in the simulation problems presented in Section III-B1, which illustrate how the performance of BP can be severely affected by suboptimal parameter choices. On the other hand, as we have discussed in Section II-C, it is not easy to find the exact PF and this can sometimes have an effect on how accurately we can find the knee region. The reason that StEMO did not perform better than BP on this signal may be that the method for detecting knee points was misled by the inclusion of a few suboptimal solutions on the estimated PF.

3) *Summary:* We compared the proposed method with the other commonly used algorithms in compressed sensing from both simulation experiments and image reconstruction experiments. As shown in Fig. 1, the performance of the other algorithms can be severely affected by the particular choice of parameters and it is usually not easy to know the optimal value for these parameters for different problems. In contrast, StEMO avoids the difficulty of choosing such parameters by locating the solutions on the knee regions of the problems. Furthermore, our experimental results show that StEMO achieves better reconstruction results for most of the problems.

#### IV. CONCLUSION

This paper has made several contributions. First, a Pareto-based multiobjective optimization method StEMO has been

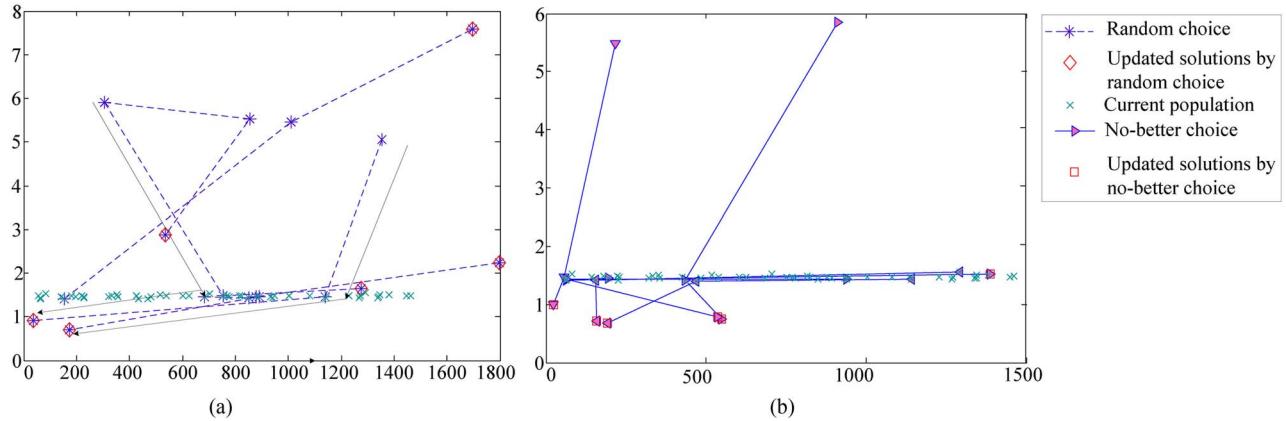


Fig. 17. Updated solutions generated by random-choice and no-better-choice methods in one early generation when case 2 ( $x^k \prec x^{k-1}$ ) in no-better-choice is applied.  $x^{k-1}$ ,  $x^k$  and  $x^{k+1}$  are shown in the figures, (a) with dashed star lines for random-choice and (b) solid triangle lines for no-better-choice. Diamonds and squares on the end of each line denote the updated solutions  $x^{k+1}$ , solutions on the other end of each line denote  $x^{k-1}$  and solutions in the middle of each line are  $x^k$ .

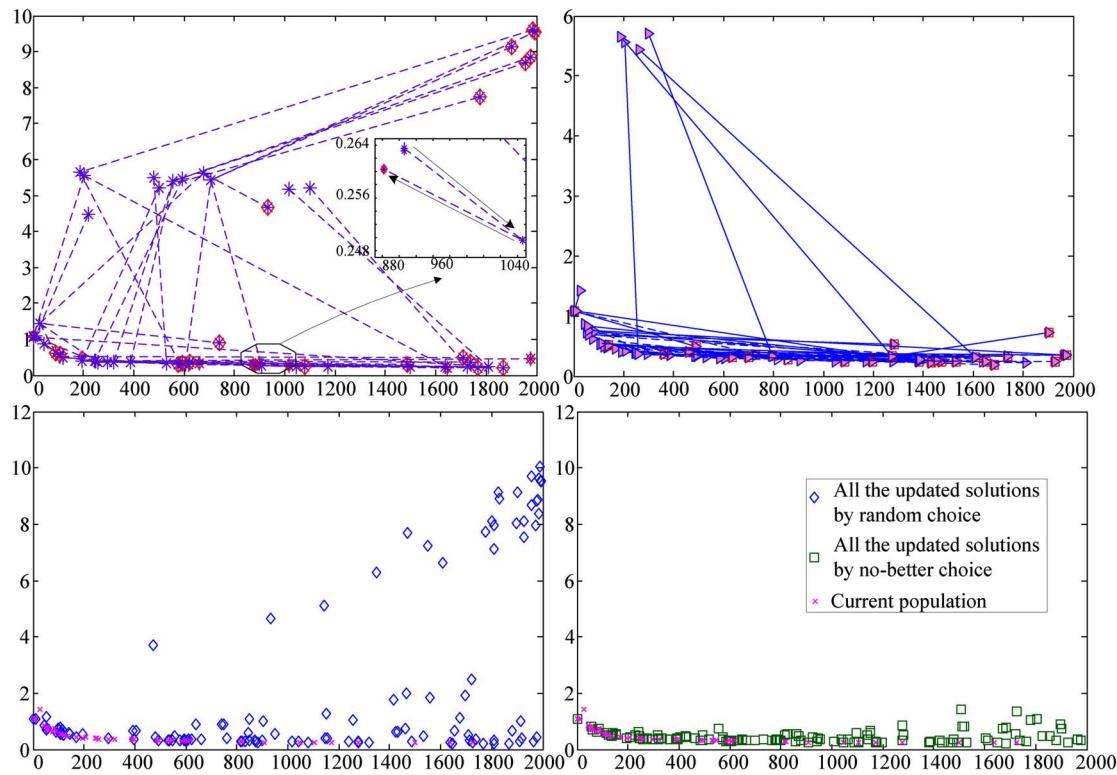


Fig. 18. Updated solutions generated by random-choice and no-better-choice in a late generation when both case 1 ( $x^{k-1}$  and  $x^k$  are nondominated to each other) and case 2 are applied. Top left figure shows solutions  $x^{k-1}$ ,  $x^k$ , and the updated solution  $x^{k+1}$  generated by the random-choice. Top right figure shows solutions  $x^{k-1}$ ,  $x^k$ , and the updated solution  $x^{k+1}$  generated by the no-better-choice (start and end points denoted as in Fig. 17). Bottom left figure shows the current population (red crosses) and new solutions  $x^{k+1}$  after one iteration of random-choice local search (blue diamonds). Bottom right figure shows current population (red crosses) and new solutions  $x^{k+1}$  after one iteration of no-better-choice local search (green squares).

proposed, for finding optimal tradeoff solutions to the sparse reconstruction problem.

Second, an experimental investigation has been presented, which explores the *PF* tradeoff between measurement error and sparse-inducing terms, and some new insights into the zero-norm sparse reconstruction algorithms have been gained by analyzing the *PF*.

Third, our experiments have shown that knee regions do exist for the bi-objective problem  $\mathbf{F}(\mathbf{x}) = (\|\mathbf{x}\|_0, \|\mathbf{Ax} - \mathbf{y}\|_2^2)$ , and that these knee regions do provide optimal solutions, which achieve the best possible tradeoff between the competing cost functions.

In addition, we have shown how important information, such as the best approximation to the noise, and the best approximation of the sparsity of the original signal, can be extracted from the position of the knee on the *PF*.

The principal contribution of the paper is to show that a multiobjective approach can often solve these kinds of sparse reconstruction problems better than the single-objective approaches, which have commonly been applied to such problems. The new algorithm, StEMO, incorporating a modified evolutionary algorithm, which generates improved estimates of *PF* by inclusion of two heuristics, via a soft-thresholding approach, can increase both the spread and the

convergence speed of solutions. The paper has given a variety of experimental results, which show the advantages of the StEMO algorithm over other commonly used techniques, and also show how it can be applied to practical problems.

Future work will be to investigate more robust methods [30] for finding knee points when the *PF* is discontinuous or nonconvex. Since it is hard to find exact optimal nondominated solutions for NP-hard problems, our method for detecting knee points can sometimes be misled by inclusion of a few suboptimal solutions on the estimated *PF*. Our ongoing work is to develop a robust knee finding method, which can accurately locate the knee region, even when information about the *PF* is incomplete. Future work will also seek interesting and useful applications of our method for solving sparse reconstruction problems in engineering and other fields.

#### APPENDIX A COMPARISON OF NO-BETTER-CHOICE WITH RANDOM-CHOICE

In Section II-A2, we analyzed the advantages of applying the two cases (Fig. 4, case 1 and case 2) to choose  $\mathbf{x}^k$  and  $\mathbf{x}^{k-1}$ . In this section, we present experimental test data to show the superior performance of using case 1 and case 2 decisions over random choices for  $\mathbf{x}^k$  and  $\mathbf{x}^{k-1}$ . Note that the purpose of these methods (case 1 and case 2) for choosing  $\mathbf{x}^k$  and  $\mathbf{x}^{k-1}$  is to mimic the relationship between the two solutions  $\mathbf{x}^k \prec \mathbf{x}^{k-1}$  in the soft-thresholding iterations. We refer to the use of these two methods as no-better-choice (see Theorem 1 in Section II-A2) and the conventional approach as *random-choice*.

Two experiments are carried out: 1) both no-better-choice and random-choice are applied on the same population, and the outputs of each method are compared, and 2) no-better-choice and random-choice are each incorporated into the local search algorithm (described in Step 5 of Table I) separately and the performance of the local search algorithm in each case is compared. Note that for the two experiments, the ground-truth signal, the measurement matrix  $A$ , and the measurement vector  $y$  are the same as described in Section III. The standard deviation of the noise is fixed to 0.1.

##### A. Effect of No-Better-Choice and Random-Choice on the Same Population

In this experiment, both the no-better-choice and random-choice for  $\mathbf{x}^k$  and  $\mathbf{x}^{k-1}$  are performed on the same population in every generation to see which one could obtain better updated solutions. In order to do this, both the no-better-choice and random-choice local searches are applied on  $P'_{\text{nondominated}}$  and  $P'_{\text{dominated}}$  in Step 5 of Table I, and the performance of each method is compared. The size of the population is 50 in this experiment, and both the subpopulations generated by random-choice and no-better-choice are combined for the selection operator (Step 6 in Table I) to ensure that both of these methods performances are compared on identical populations. As explained in Section II-A2, the two selection techniques (case 1 and case 2) for choosing  $\mathbf{x}^k$  and  $\mathbf{x}^{k-1}$  in no-better-choice have different roles and are applied according to a decision criterion. The performance of no-better-choice and random-choice are compared in Figs. 17 and 18.

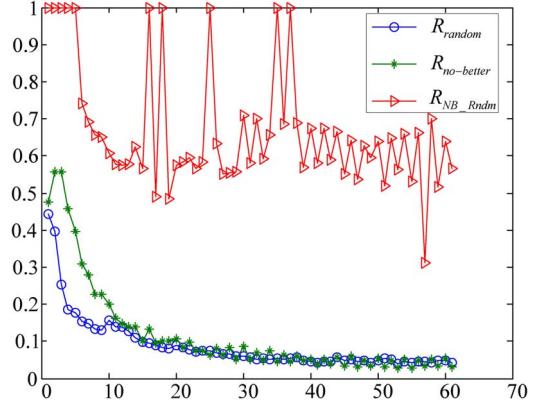


Fig. 19. Comparison of nondominated solutions generated by no-better-choice and random-choice.  $R_{\text{random}}$  denotes the proportion of solutions generated by random-choice that dominate the solutions in the current population.  $R_{\text{no-better}}$  denotes the proportion of the solutions generated by no-better-choice that dominate the current population. Consider the subpopulation combined by all the solutions generated by both choices as  $P'_{\text{NR}}$ .  $R_{\text{NB\_Rndm}}$  represents the ratio of the number of nondominated solutions generated by the no-better-choice in  $P'_{\text{NR}}$  to all the nondominated solutions in  $P'_{\text{NR}}$ . The vertical axis denotes the ratios and horizontal axis denotes successive generations.

In Fig. 17, more new solutions dominate the previous solutions and the current population (see the rectangles in the lower-left part in right subfigure in Fig. 17). Compared to no-better-choice, random-choice generated worse solutions as some of the solutions divert toward the top-right region of the objective space, which means that the solutions deteriorate in both objective functions. But random-choice can also sometimes result in improved solutions, as shown in the arrowed lines in the left sub-figure in Fig. 17, where two nondominated solutions are obtained. It is obvious that the relationships between the previous solutions, which generated these two (random) improved solutions are  $\mathbf{x}^k \prec \mathbf{x}^{k-1}$  and so we see that the success here has come about by random choices accidentally replicating the case 2 criterion of the no-better-choice method. Therefore, the data of Fig. 17 supports the rationale of choosing  $\mathbf{x}^k \prec \mathbf{x}^{k-1}$  (case 2) when we apply the soft-thresholding as a local search within our evolutionary algorithm.

Fig. 18 shows another example, chosen from a later stage in the evolutionary algorithm, where some of the no-better-choice decisions are case 1 and some are case 2 (line 2 in Table II explains how these two options are chosen by our algorithm). Clearly, solutions obtained by the no-better-choice method are located in the lower part of the space (close to the current population which means they are more likely to be nondominated or to dominate the current population). In contrast, some solutions generated by the random-choice method are near the current population but many other solutions appear toward the upper-right part of the space (as compared to the current population) which means they are inferior dominated solutions. Also, an enlarged graph in the top-left figure is given to show one of the nondominated solutions generated by random-choice for  $\mathbf{x}^{k-1}$  and  $\mathbf{x}^k$ . Note that, in this case,  $\mathbf{x}^{k-1}$  and  $\mathbf{x}^k$  are two nondominated solutions and  $\mathbf{x}^{k+1}$  is nondominated to  $\mathbf{x}^k$ , which corresponds to case 1.

Fig. 19 shows the results of random-choice and no-better-choice based on 30 independent trials with 60 generations in each trial.  $R_{\text{random}}$ ,  $R_{\text{no-better}}$ , and  $R_{\text{NB\_Rndm}}$  are the average

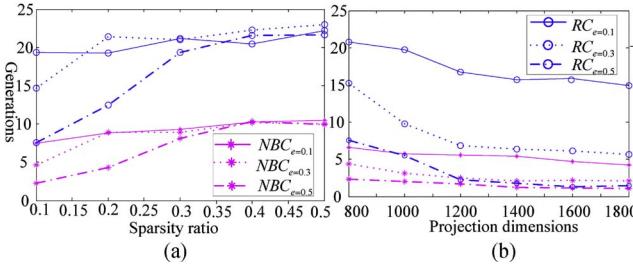


Fig. 20. Minimum generations to obtain solutions with the three different amounts of final error from the ground-truth signal.  $NBC_{e=0.1}$ ,  $NBC_{e=0.3}$ , and  $NBC_{e=0.5}$ , denote the minimum generations for evolutionary algorithm using no-better-choice method to obtain solutions with final errors  $e = 0.1$ ,  $0.3$ , and  $0.5$ , respectively.  $RC_{e=0.1}$ ,  $RC_{e=0.3}$ , and  $RC_{e=0.5}$ , denote the minimum generations for evolutionary algorithm using the random-choice method to obtain solutions with final errors  $e = 0.1$ ,  $0.3$ , and  $0.5$ , respectively. (a) Sparsity ratio (the proportion of nonzero elements in the ground-truth signal) varies from  $0.1$  to  $0.5$ . (b) Projection dimensions  $M$  vary from  $800$  to  $1800$ . Each point is obtained by averaging the results from  $30$  independent trials.

results of the  $30$  independent trials. It is obvious that no-better-choice shows better performance in early stages of the evolutionary procedure; however, it seems that random-choice catches up with no-better-choice in later stages in terms of its ability to generate nondominated solutions. Since the ratio of nondominated solutions in current solutions increases anyway with successive generations, it is not surprising that random-choice improves its performance in later generations in terms of nondominated solutions ( $R_{random}$  and  $R_{no-better}$ ). More nondominated solutions in the population means case 1 is applied with higher probability (thereby emphasizing spreading solutions along the  $PF$  rather than seeking nondominated solutions) and it also means that there is more of a chance for random-choice to get two non-dominated solutions (case 1) for obtaining the updated solution. Although random-choice achieves similar performance in terms of generating nondominated solutions in late stages of the evolutionary procedure,  $R_{NB\_Rndm}$  shows that no-better-choice usually produces a greater number of nondominated solutions than random-choice for most generations. If  $R_{NB\_Rndm} > 0.5$ , then no-better-choice obtained more solutions that dominate the solutions generated by random-choice; otherwise, random-choice achieves more solutions that dominated the ones obtained by no-better-choice.  $R_{NB\_Rndm} = 1$  means all the solutions obtained by no-better-choice dominate all the solutions generated by random-choice. Fig. 19 also shows no-better-choice achieves better performance than random-choice.

### B. Incorporate No-Better-Choice and Random-Choice Into the Local Search Algorithm

In Fig. 20, for each fixed error ( $0.1$ ,  $0.3$  or  $0.5$ ), the evolutionary algorithm using the no-better-choice method obtains a satisfactory solution with fewer generations than the random-choice method, regardless of different sparsity ratios or projection dimensions. This illustrates the superior performance of no-better-choice and supports the decision to employ no-better-choice instead of random-choice for choosing  $\mathbf{x}^k$  and  $\mathbf{x}^{k-1}$ .

## APPENDIX B FIGURES PRODUCED BY EACH ALGORITHM FOR THE BENCHMARK PROBLEMS

In this section, the original signal and the reconstruction results (Figs. 21–29) produced by each algorithm for the benchmark problems are shown.

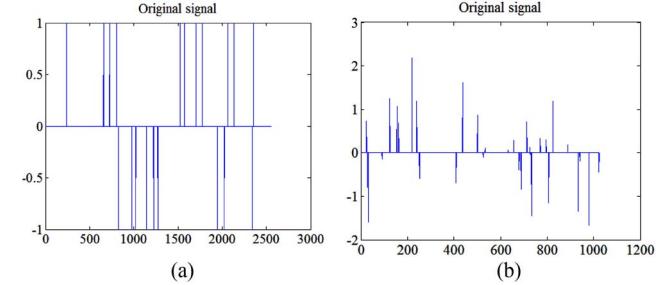


Fig. 21. Original signals of (a) sgnspike and (b) gausspike.

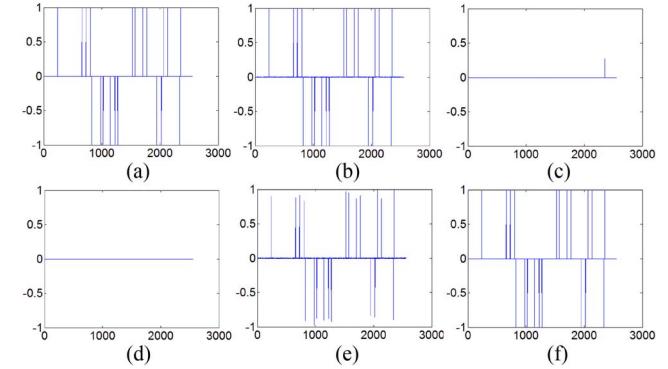


Fig. 22. Reconstructed signals of sgnspike by each of the six algorithms. (a) sgnspike by OMP. (b) sgnspike by BP. (c) sgnspike by Homotopy. (d) sgnspike by IST. (e) sgnspike by ADM. (f) sgnspike by StEMO.

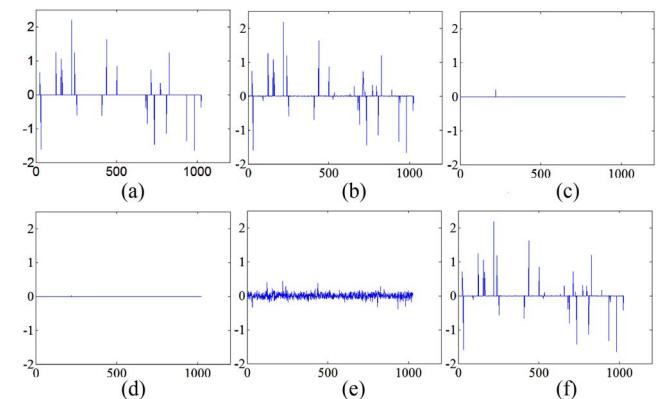


Fig. 23. Reconstructed signals of gausspike produced by the six algorithms. (a) gausspike by OMP. (b) gausspike by BP. (c) gausspike by Homotopy. (d) gausspike by IST. (e) gausspike by ADM. (f) gausspike by StEMO.

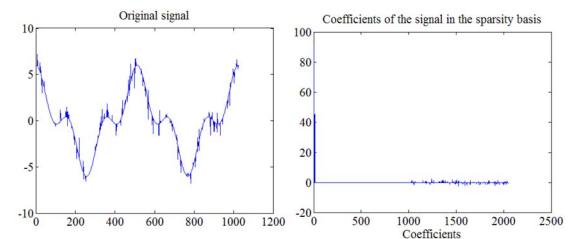


Fig. 24. Original cosspike signal and its coefficients in the sparsity basis.

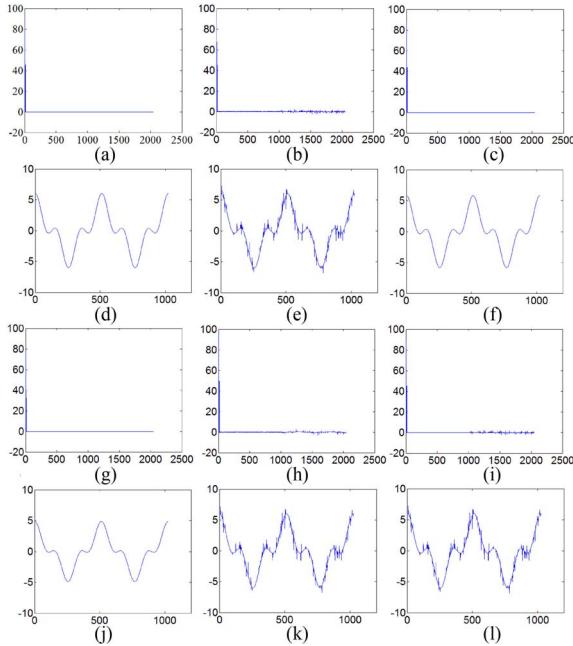


Fig. 25. Reconstructed coefficients and signals of cosspike by the six algorithms. (a) Coefficients by OMP. (b) Coefficients by BP. (c) Coefficients by Homotopy. (d) Signal by OMP. (e) Signal by BP. (f) Signal by Homotopy. (g) Coefficients by IST. (h) Coefficients by ADM. (i) Coefficients by StEMO. (j) Signal by IST. (k) Signal by ADM. (l) Signal by StEMO.

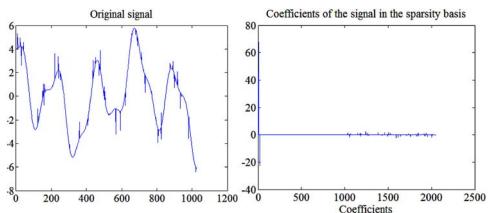


Fig. 26. Original gcosspike signal and its coefficients in the sparsity basis.

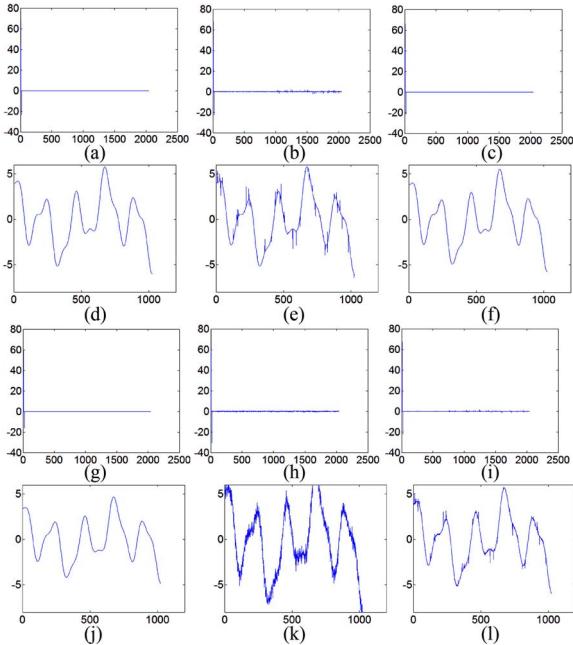


Fig. 27. Reconstructed coefficients and signals of gcosspike by the six algorithms. (a) Coefficients by OMP. (b) Coefficients by BP. (c) Coefficients by Homotopy. (d) Signal by OMP. (e) Signal by BP. (f) Signal by Homotopy. (g) Coefficients by IS. (h) Coefficients by ADM. (i) Coefficients by StEMO. (j) Signal by IST. (k) Signal by ADM. (l) Signal by StEMO.

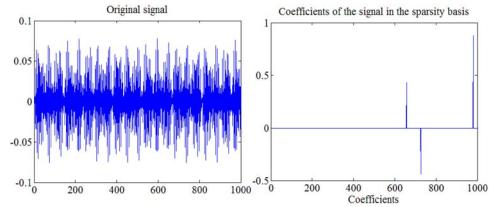


Fig. 28. Original jitter signal and its coefficients in the sparsity basis.

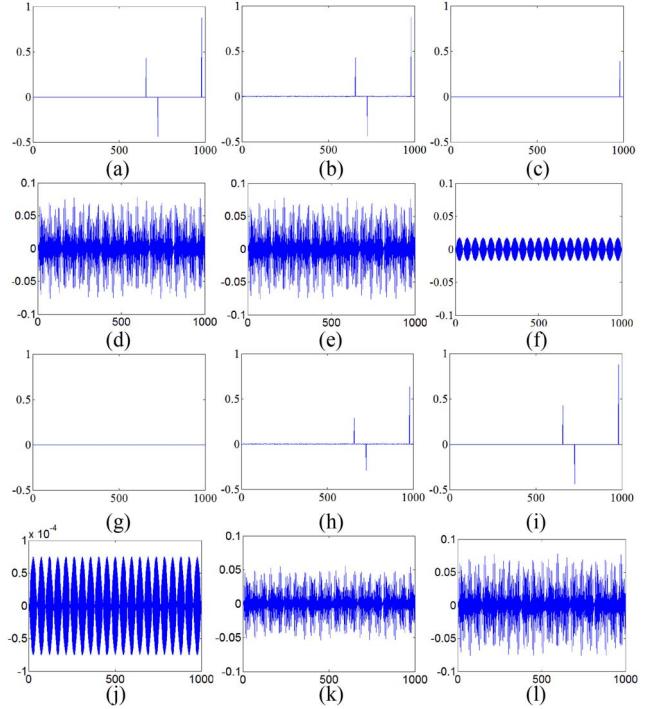


Fig. 29. Reconstructed coefficients and signals of jitter by the six algorithms. (a) Coefficients by OMP. (b) Coefficients by BP. (c) Coefficients by Homotopy. (d) Signal by OMP. (e) Signal by BP. (f) Signal by Homotopy. (g) Coefficients by IST. (h) Coefficients by ADM. (i) Coefficients by StEMO. (j) Signal by IST. (k) Signal by ADM. (l) Signal by StEMO.

## REFERENCES

- [1] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
- [2] E. J. Candès and Y. Plan, "Near-ideal model selection by  $l_1$  minimization," *Ann. Stat.*, vol. 37, no. 5A, pp. 2145–2177, 2009.
- [3] E. J. Candès, "The restricted isometry property and its implications for compressed sensing," *Comptes Rendus Math.*, vol. 346, no. 9, pp. 589–592, 2008.
- [4] J. Wu, F. Liu, L. C. Jiao, and X. Wang, "Compressive sensing SAR image reconstruction based on Bayesian framework and evolutionary computation," *IEEE Trans. Image Process.*, vol. 20, no. 7, pp. 1904–1911, Jul. 2011.
- [5] H. Lee, A. Battle, R. Raina, and A. Y. Ng, "Efficient sparse coding algorithms," in *Proc. 19th Conf. Adv. Neural Inf. Process. Syst.*, vol. 19, pp. 801–808, 2007.
- [6] A. Y. Yang, S. S. Sastry, A. Ganesh, and Y. Ma, "Fast  $l_1$ -minimization algorithms and an application in robust face recognition: A review," in *Proc. 17th IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 1849–1852.
- [7] M. A. T. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," *IEEE J. Sel. Topics Signal Process.*, vol. 1, no. 4, pp. 586–597, Dec. 2007.
- [8] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *Appl. Comput. Harmonic Anal.*, vol. 27, no. 3, pp. 265–274, 2009.
- [9] G. Davis, "Adaptive nonlinear approximations," Ph.D. dissertation, Dept. Math., Courant Inst. of Math. Sci., New York Univ., New York, NY, USA, 1994.

- [10] G. Davis, S. Mallat, and M. Avellaneda, "Adaptive greedy approximations," *Constructive Approx.*, vol. 13, no. 1, pp. 57–98, 1997.
- [11] V. N. Temlyakov, "The best m-term approximation and greedy algorithms," *Adv. Comput. Math.*, vol. 8, no. 3, pp. 249–265, 1998.
- [12] T. Blumensath and M. Davies, "Normalized iterative hard thresholding: Guaranteed stability and performance," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 298–309, Apr. 2010.
- [13] Z. Xu, X. Chang, F. Xu, and H. Zhang, "L<sub>1</sub>/2 regularization: A thresholding representation theory and a fast solver," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 7, pp. 1013–1027, Jul. 2012.
- [14] S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing," *IEEE Trans. Signal Process.*, vol. 56, no. 6, pp. 2346–2356, Jun. 2008.
- [15] D. L. Donoho and Y. Tsaig, "Fast solution of l1 norm minimization problems when the solution may be sparse," *IEEE Trans. Inf. Theory*, vol. 54, no. 11, pp. 4789–4812, Nov. 2008.
- [16] D. M. Malioutov, M. Cetin, and A. S. Willsky, "Homotopy continuation for sparse signal representation," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Process.*, Mar. 2005, vol. 5, pp. 733–736.
- [17] K. K. Herrity, A. C. Gilbert, and J. A. Tropp, "Sparse approximation via iterative thresholding," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Process.*, May 2006, vol. 3, pp. 624–627.
- [18] L. Jiao, L. Li, R. Shang, F. Liu, and R. Storkin, "A novel selection evolutionary strategy for constrained optimization," *Inf. Sci.*, vol. 239, no. 1, pp. 122–141, 2013.
- [19] D. Lim, Y. S. Ong, M. H. Lim, and Y. Jin, "Single/multi-objective inverse robust evolutionary design methodology in the presence of uncertainty," in *Evolutionary Computation in Dynamic and Uncertain Environments*, vol. 51, S. Yang, Y. Ong, and Y. Jin, Eds. Berlin, Germany: Springer, 2007, pp. 437–456.
- [20] H. Wang, S. Kwong, Y. Jin, W. Wei, and K. Man, "Multi-objective hierarchical genetic algorithm for interpretable fuzzy rule-based knowledge extraction," *Fuzzy Sets Syst.*, vol. 149, no. 1, pp. 149–186, 2005.
- [21] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007.
- [22] Z. Cai and Y. Wang, "A multiobjective optimization-based evolutionary algorithm for constrained optimization," *IEEE Trans. Evol. Comput.*, vol. 10, no. 6, pp. 658–675, Dec. 2006.
- [23] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.
- [24] Y. Jin and B. Sendhoff, "Pareto-based multiobjective machine learning: An overview and case studies," *IEEE Trans. Syst., Man, Cybern.C, Appl. Rev.*, vol. 38, no. 3, pp. 397–415, May 2008.
- [25] Y. Jin, *Multi-objective machine learning*. Berlin, Germany: Springer Verlag, 2006, vol. 16.
- [26] A. Chandra and X. Yao, "Evolving hybrid ensembles of learning machines for better generalisation," *Neurocomputing*, vol. 69, nos. 7–9, pp. 686–700, 2006.
- [27] H. Chen and X. Yao, "Multiobjective neural network ensembles based on regularized negative correlation learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 12, pp. 1738–1751, Dec. 2010.
- [28] L. Rachmawati and D. Srinivasan, "Multiobjective evolutionary algorithm with controllable focus on the knees of the Pareto front," *IEEE Trans. Evol. Comput.*, vol. 13, no. 4, pp. 810–824, Aug. 2009.
- [29] J. Branke, K. Deb, H. Dierolf, and M. Osswald, "Finding knees in multi-objective optimization," in *Proc. 8th Conf. PPSN*, 2004, LNCS 3242, pp. 722–731.
- [30] K. Deb and S. Gupta, "Understanding knee points in bicriteria problems and their implications as preferred solution principles," *Eng. Optimiz.*, vol. 43, no. 11, pp. 1175–1204, 2011.
- [31] I. Das, "On characterizing the 'knee' of the Pareto curve based on normal-boundary intersection," *Structural Multidisciplinary Optimiz.*, vol. 18, no. 2, pp. 107–115, 1999.
- [32] T. Schnier, X. Yao, and P. Liu, "Digital filter design using multiple Pareto fronts," *Soft Comput.*, vol. 8, no. 5, pp. 332–343, 2004.
- [33] K. Praditwong, M. Harman, and X. Yao, "Software module clustering as a multi-objective search problem," *IEEE Trans. Softw. Eng.*, vol. 37, no. 2, pp. 264–282, Mar.–Apr. 2011.
- [34] S. Kukkonen and K. Deb, "Improved pruning of non-dominated solutions based on crowding distance for bi-objective optimization problems," in *Proc. WCCI*, 2006, pp. 1179–1186.
- [35] D. Yang, L. Jiao, M. Gong, and J. Feng, "Adaptive ranks clone and k-nearest neighbour list based immune multi-objective optimization," *Comput. Intell.*, vol. 26, no. 4, pp. 359–385, 2010.
- [36] A. Neubauer, "A theoretical analysis of the non-uniform mutation operator for the modified genetic algorithm," in *Proc. IEEE Int. Conf. Evol. Comput.*, Apr. 1997, pp. 93–96.
- [37] L. J. Eshelman, "Real-coded genetic algorithms and interval-schemata," in *Proc. Foundations Genetic Algorithms II*, D. L. Whitley, Ed., San Mateo, CA, pp. 187–202, 1993.
- [38] S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2479–2493, Jun. 2009.
- [39] E. T. Hale, W. Yin, and Y. Zhang, "A fixed-point continuation method for l1-regularized minimization with applications to compressed sensing," Dept. Comput. Appl. Math., Rice Univ., Houston, TX, USA, Tech. Rep. TR07-07, 2007.
- [40] Z. Wang, K. Tang, and X. Yao, "Multi-objective approaches to optimal testing resource allocation in modular software systems," *IEEE Trans. Reliab.*, vol. 59, no. 3, pp. 563–575, Sep. 2010.
- [41] A. Graps, "An introduction to wavelets," *IEEE Comput. Sci. Eng.*, vol. 2, no. 2, pp. 50–61, 1995.
- [42] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 1, pp. 129–159, 2001.
- [43] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, 2009.
- [44] J. Yang and Y. Zhang, "Alternating direction algorithms for l1-problems in compressive sensing," *SIAM J. Sci. Comput.*, vol. 33, no. 1, pp. 250–278, 2011.
- [45] A. Y. Yang *et al.* (2009). *L1 Benchmark Package* [Online]. Available: <http://www.eecs.berkeley.edu/yang/software/l1benchmark/l1benchmark.zip>
- [46] E. V. Berg, M. P. Friedlander, G. Hennenfent, F. Herrmann, R. Saab, and Ö. Yilmaz, "Sparco: A testing framework for sparse reconstruction," Dept. Computer Science, Univ. Brit. Columbia, Vancouver, U.K., Tech. Rep. TR-2007-20, Oct. 2007.



**Lin Li** (S'10) received the B.Sc. degree in information and computation science from Xidian University, Xian, China, in 2009. She is currently pursuing the Ph.D. degree in electronic engineering at Xidian University.

Her current research interests include computational intelligence, machine learning, numerical optimization, signal processing and image processing.



**Xin Yao** (M'91–SM'96–F'03) received the B.Sc. degree from the University of Science and Technology of China (USTC), Hefei, China, in 1982, the M.Sc. degree from the North China Institute of Computing Technology, Beijing, China, in 1985, and the Ph.D. degree from USTC, in 1990.

Since 1999, he has been a Professor (Chair) of computer science with the University of Birmingham, Birmingham, U.K., where he is currently the Director of the Centre of Excellence for Research in Computational Intelligence and Applications. He

is also a Distinguished Visiting Professor (Grand Master Professorship) of USTC. He has over 350 refereed publications. His research interests include evolutionary computation, neural network ensembles and multiple classifiers, meta-heuristic algorithms, data mining, computational complexity of evolutionary algorithms, and real-world applications.

Dr. Yao was a recipient of the President's Award for Outstanding Thesis by the Chinese Academy of Sciences for his Ph.D. work on simulated annealing and evolutionary algorithms in 1989. He won the 2001 IEEE Donald G. Fink Prize Paper Award for his work on evolutionary artificial neural networks. He was the Editor-in-Chief of the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION from 2003 to 2008, an Associate Editor or Editorial Board Member of 12 other journals, and the Editor of the World Scientific Book Series on *Advances in Natural Computation*. He has given over 60 invited keynote and plenary speeches at conferences and workshops worldwide.



**Rustam Stolkin** (M'12) received the undergraduate and master's degrees in engineering science from the University of Oxford, Oxford, U.K., in 1998, and the Ph.D. degree in robotic vision from University College London, London, U.K., in 2004.

From 2004 to 2008, he was a Research Assistant Professor with the Stevens Institute of Technology, Hoboken, NJ, USA, where he worked on sensor systems for maritime security, before returning to the U.K. to focus his work on robotics and machine intelligence at the University of Birmingham, Birmingham, U.K. His research has always been diverse, spanning areas of science and engineering outside of robotics, as well as manipulation with robotic arms and hands, novel robotic vehicles, computer vision and other kinds of autonomous sensing, and a long-term interest in engineering and science education. He is also keen on collaborations between academia and industry, with current applied projects including robotic decommissioning of nuclear waste, and robotics for bomb disposal.

Dr. Stolkins is a Senior Birmingham Fellow with the School of Mechanical Engineering, University of Birmingham. He has been a member of the Robotics and Automation Society, the Marine Technology Society and the American Society for Engineering Education.



**Maoguo Gong** (M'07) received the B.S. degree in electronic engineering (with first class honors) and the Ph.D. degree in electronic science and technology from Xidian University, Xian, China, in 2003 and 2009, respectively.

Since 2006, he has been a Teacher with Xidian University. His research interests include computational intelligence with applications.

Dr. Gong is a member of the IEEE Computational Intelligence Society, an Executive Committee Member of the Natural Computation Society of Chinese Association for Artificial Intelligence, and a Senior Member of the Chinese Computer Federation.



**Shan He** (M'07) received the M.Sc. Eng. degree (with distinction) in intelligence engineering and the Ph.D. degree from the Department of Electrical Engineering and Electronics, University of Liverpool, Liverpool, U.K., in 2002 and 2006, respectively.

He is currently a Lecturer in Computational Biology with the School of Computer Science, University of Birmingham. He is also with the Centre for Systems Biology, School of Biosciences. His research interests include swarm intelligence, neural networks, evolutionary computation, systems biology, and animal behavior.