An Interior-Point Method for Large-Scale ℓ_1-Regularized Least Squares

Seung-Jean Kim, *Member, IEEE*, K. Koh, M. Lustig, Stephen Boyd, *Fellow, IEEE*, and Dimitry Gorinevsky, *Fellow, IEEE*

***Abstract*—Recently, a lot of attention has been paid to ℓ_1 regularization based methods for sparse signal reconstruction (e.g., basis pursuit denoising and compressed sensing) and feature selection (e.g., the Lasso algorithm) in signal processing, statistics, and related fields. These problems can be cast as ℓ_1-regularized least-squares programs (LSPs), which can be reformulated as convex quadratic programs, and then solved by several standard methods such as interior-point methods, at least for small and medium size problems. In this paper, we describe a specialized interior-point method for solving large-scale ℓ_1-regularized LSPs that uses the preconditioned conjugate gradients algorithm to compute the search direction. The interior-point method can solve large sparse problems, with a million variables and observations, in a few tens of minutes on a PC. It can efficiently solve large dense problems, that arise in sparse signal recovery with orthogonal transforms, by exploiting fast algorithms for these transforms. The method is illustrated on a magnetic resonance imaging data set.**

***Index Terms*—Basis pursuit denoising, compressive sampling, compressed sensing, convex optimization, interior-point methods, least squares, preconditioned conjugate gradients, ℓ_1 regularization.**

I. Introduction

We consider a linear model of the form

$$y = Ax + v,$$

where $x \in \mathbf{R}^n$ is the vector of unknowns, $y \in \mathbf{R}^m$ is the vector of observations, $v \in \mathbf{R}^m$ is the noise, and $A \in \mathbf{R}^{m \times n}$ is the data matrix.

When $m \geq n$ and the columns of A are linearly independent, we can determine x by solving the least squares problem of minimizing the quadratic loss $||Ax - y||_2^2$, where $||u||_2 = (\sum_i u_i^2)^{1/2}$ denotes the ℓ_2 norm of u.

When m, the number of observations, is not large enough compared to n, simple least-squares regression leads to over-fit.

Manuscript received January 30, 2007; revised August 30, 2007. This work was supported by the Focus Center Research Program Center for Circuit and System Solutions award 2003-CT-888, by JPL award I291856, by the Precourt Institute on Energy Efficiency, by Army award W911NF-07-1- 0029, by NSF awards ECS-0423905 and 0529426, by DARPA award N66001-06-C-2021, by NASA award NNX07AEIIA, by AFOSR award FA9550-06-1-0514, and by AFOSR award FA9550-06-1-0312. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Yonina Eldar.

The authors are with the Information Systems Lab, Department of Electrical Engineering, Stanford University, Stanford, CA 94305 USA (e-mail: sjkim@stanford.edu; deneb1@stanford.edu; mlustig@stanford.edu; boyd@stanford.edu; gorin@stanford.edu).

Digital Object Identifier 10.1109/JSTSP.2007.910971

A. ℓ_2-Regularized Least Squares

A standard technique to prevent over-fitting is ℓ_2 or Tikhonov regularization [38], which can be written as

$$\text{minimize } ||Ax - y||_2^2 + \lambda ||x||_2^2 \tag{1}$$

where $\lambda > 0$ is the regularization parameter. The Tikhonov regularization problem or ℓ_2-regularized least-squares program (LSP) has the analytic solution

$$x^{l2} = (A^T A + \lambda I)^{-1} A^T y. \tag{2}$$

We list some basic properties of Tikhonov regularization, which we refer to later when we compare it to ℓ_1-regularized least squares.

- *Linearity*. From (2), we see that the solution x^{l2} to the Tikhonov regularization problem is a linear function of y.
- *Limiting behavior as* $\lambda \to 0$. As $\lambda \to 0, x^{l2}$ converges to the Moore–Penrose solution $A^\dagger y$, where $A^\dagger$ is the Moore–Penrose pseudoinverse of A. The limit point has the minimum ℓ_2-norm among all points that satisfy $A^T(Ax - y) = 0$:

$$A^\dagger y = \arg \min_{A^T(Ax-y)=0} ||x||_2.$$

- *Convergence to zero as* $\lambda \to \infty$. The optimal solution x^{l2} tends to zero, as $\lambda \to \infty$.
- *Regularization path*. The optimal solution x^{l2} is a smooth function of the regularization parameter λ, as it varies over $[0, \infty)$. As λ decreases to zero, x^{l2} converges to the Moore–Penrose solution; as λ increases, x^{l2} converges to zero.

The solution to the Tikhonov regularization problem can be computed by direct methods, which require $O(n^3)$ flops (assuming that m is order n or less), when no structure is exploited. The solution can also be computed by applying iterative (nondirect) methods (e.g., the conjugate gradients method) to the linear system of equations $(A^T A + \lambda I)x = A^T y$. Iterative methods are efficient especially when there are fast algorithms for the matrix-vector multiplications with the data matrix A and its transpose A^T (i.e., Au and $A^T v$ with $u \in \mathbf{R}^n$ and $v \in \mathbf{R}^m$), which is the case when A is sparse or has a special form such as partial Fourier and wavelet matrices.

B. ℓ_1-Regularized Least Squares

In ℓ_1-regularized least squares (LS), we substitute a sum of absolute values for the sum of squares used in Tikhonov regularization, to obtain

$$\text{minimize } \|Ax-y\|_2^2+\lambda\|x\|_1 \tag{3}$$

where $\|x\|_1=\sum_i^n |x_i|$ denotes the ℓ_1 norm of x and $\lambda>0$ is the regularization parameter. We call (3) an ℓ_1-regularized LSP. This problem always has a solution, but it need not be unique.

We list some basic properties of ℓ_1-regularized LS, pointing out similarities and differences with ℓ_2-regularized LS.

- *Nonlinearity*. From (2), we see that Tikhonov regularization yields a vector x, which is a linear function of the vector of observations y. By contrast, ℓ_1-regularized least squares yields a vector x, which is not linear in y.
- *Limiting behavior as* $\lambda\to 0$. ℓ_1-regularized LS shows a different limiting behavior with ℓ_1-regularized LS, as $\lambda\to 0$. In ℓ_1-regularized LS, the limiting point has the minimum ℓ_1 norm among all points that satisfy $A^T(Ax-y)=0$.
- *Finite convergence to zero as* $\lambda\to\infty$. As in Tikhonov regularization, the optimal solution tends to zero, as $\lambda\to\infty$. For ℓ_1-regularized LS, however, the convergence occurs for a finite value of λ:

$$\lambda\geq\lambda_{\max}=\|2A^Ty\|_\infty \tag{4}$$

  where $\|u\|_\infty=\max_i|u_i|$ denotes the ℓ_∞ norm of the vector u. For $\lambda\geq\lambda_{\max}$, the optimal solution of the ℓ_1-regularized LSP (3) is 0. In contrast, the optimal solution to the Tikhonov regularization problem is zero only in the limit as $\lambda\to\infty$. (The derivation of the formula (4) for $\lambda_{\max}$ is given in Section III-A.)
- *Regularization path* The solution x^{l2} to the Tikhonov regularization problem varies smoothly as the regularization parameter λ varies over $[0,\infty)$. By contrast, the regularization path of the ℓ_1-regularized LSP (3), i.e., the family of solutions as λ varies over $(0,\infty)$, has the piecewise-linear solution path property [15]: There are values $\lambda_1,\ldots,\lambda_k$, with $0=\lambda_k<\cdots<\lambda_1=\lambda_{\max}$, such that the regularization path is a piecewise linear curve on $\mathbf{R}^n$

$$x^{11}=\frac{\lambda_i-\lambda}{\lambda_i-\lambda_{i+1}}x^{(i+1)}+\frac{\lambda-\lambda_{i+1}}{\lambda_i-\lambda_{i+1}}x^{(i)},\quad \lambda_{i+1}\leq\lambda\leq\lambda_i,\quad i=1,\ldots,k-1$$

  where $x^{(i)}$ solves the ℓ_1-regularized LSP (3) with $\lambda=\lambda_i$. (So $x^{(1)}=0$ and $x^{11}=0$ when $\lambda\geq\lambda_1$.)

More importantly, ℓ_1-regularized LS typically yields a *sparse* vector x, that is x that has relatively few nonzero coefficients. (As λ decreases, it tends to be sparser but not necessarily [26], [52].) In contrast, the solution x^{l2} to the Tikhonov regularization problem typically has all coefficients nonzero.

Recently, the idea of ℓ_1 regularization has been receiving a lot of interest in signal processing and statistics. In signal processing, the idea of ℓ_1 regularization comes up in several contexts including basis pursuit denoising [8] and a signal recovery method from incomplete measurements (e.g., [4], [7], [6], [11], [12], [54], [53]). In statistics, the idea of ℓ_1 regularization is used in the well-known Lasso algorithm [52] for feature selection and its extensions including the elastic net [63].

Some of these problems do not have the standard form (3) but have a more general form

$$\text{minimize}\quad \|Ax-y\|_2^2+\sum_{i=1}^n\lambda_i|x_i| \tag{5}$$

where $\lambda_i\geq 0$ are regularization parameters. (The variables x_i that correspond to $\lambda_i=0$ are not regularized.) This general problem can be reformulated as a problem of the form (3).

We now turn to the computational aspect of ℓ_1-regularized LS, the main topic of this paper. There is no analytic formula or expression for the optimal solution to the ℓ_1-regularized LSP (3), analogous to (2); its solution must be computed numerically. The objective function in the ℓ_1-regularized LSP (3) is convex but not differentiable, so solving it is more of a computational challenge than solving the ℓ_2-regularized LRP (1).

Generic methods for nondifferentiable convex problems, such as the ellipsoid method or subgradient methods [51], [43], can be used to solve the ℓ_1-regularized LSP (3). These methods are often very slow.

The ℓ_1-regularized LSP (3) can be transformed to a convex quadratic problem, with linear inequality constraints. The equivalent quadratic program (QP) can be solved by standard convex optimization methods such as interior-point methods [32], [37], [57], [58]. Standard interior-point methods are implemented in general purpose solvers including `MOSEK` [34], which can readily handle small and medium size problems. Standard methods cannot handle large problems in which there are fast algorithms for the matrix-vector operations with A and A^T. Specialized interior-point methods that exploit such algorithms can scale to large problems, as demonstrated in [8], [27]. High-quality implementations of specialized interior-point methods include `l1-magic` [5] and `PDCO` [50], which use iterative algorithms, such as the conjugate gradients (CG) or LSQR algorithm [42], to compute the search step.

Recently, several researchers have proposed homotopy methods and variants for solving ℓ_1-regularized LSPs [14], [24], [15], [46], [40]. Using the piecewise linear property of the regularization path, path-following methods can compute efficiently the entire solution path in an ℓ_1-regularized LSP. When the solution of (13) is extremely sparse, these methods can be very fast, since the number of kinks the methods need to find is modest [14]. Otherwise, the path-following methods can be slow, which is often the case for large-scale problems. Other recently developed computational methods for ℓ_1-regularized LSPs include coordinate-wise descent methods [19], a fixed-point continuation method [23], Bregman iterative regularization based methods [41], [59], sequential subspace optimization methods [35], bound optimization methods [17], iterated shrinkage methods [9], [16], gradient methods [36], and gradient projection algorithms [18]. Some of these methods including the gradient projection algorithms [18] can efficiently handle very large problems.

C. Outline

The main purpose of this paper is to describe a specialized interior-point method for solving large ℓ_1-regularized LSPs that uses the preconditioned conjugate gradients algorithm to compute the search step. The method which we will describe in Section IV can solve large sparse problems with m and n on the order of a million in a few tens of minutes on a PC. It can efficiently solve large dense problems, that arise in sparse signal recovery with orthogonal transforms, by exploiting fast algorithms for these transforms. The specialized method is far more efficient than interior-point methods that use direct or CG methods to compute the search step, as demonstrated with several examples in Section V. Compared with first-order methods such as coordinate-descent methods, our method is comparable in solving large problems with modest accuracy, but is able to solve them with high accuracy with relatively small additional computational cost.

We illustrate the method on a real magnetic resonance imaging data set in Section V. The method can solve the QP that arises in reconstructing the object of interest with adequate accuracy from partially sampled Fourier coefficients, within around 100 preconditioned conjugate gradients (PCG) steps, which amounts to performing a few hundred fast Fourier transform (FFT) operations on the object.

Although the interior-point method (we will describe in Section IV) is tailored toward ℓ_1-regularized LSPs, the main idea behind the method can be generalized to other convex problems involving ℓ_1 regularization. We describe some generalizations of the method in Section VI. We also show how the method can easily handle the general ℓ_1-regularized LSP (5), without forming explicitly the data matrix, which is typically dense, of its equivalent formulation of the form (3).

II. ℓ_1 Regularization in Signal Processing and Statistics

Several signal processing and estimation methods based on the idea of ℓ_1 regularization can be formulated as problems of the form (3) or (5). In this section, we consider two important ℓ_1 regularization based methods. One of these is related to the magnetic resonance imaging (MRI) example given in Section V.

A. Compressed Sensing/Compressive Sampling

Let z be an unknown vector in $\mathbf{R}^n$ (which can represent a 2- or 3-D object of interest). Suppose that we have m linear measurements of an unknown signal $z \in \mathbf{R}^n$

$$y_i = \langle \phi_i, z \rangle + v_i, \quad i = 1, \ldots, m$$

where $\langle \cdot, \cdot \rangle$ denotes the usual inner product, $v \in \mathbf{R}^m$ is the noise, and $\phi_i \in \mathbf{R}^n$ are known signals. Standard reconstruction methods require at least n samples. Suppose we know *a priori* that z is compressible or has a sparse representation in a transform domain, described by $W \in \mathbf{R}^{n \times n}$ (after expanding the real and imaginary parts if necessary). In this case, if ϕ_i are well chosen, then the number of measurements m can be dramatically smaller than the size n usually considered necessary [4], [11].

Compressed sensing [11] or compressive sampling [4] attempts to exploit the sparsity or compressibility of the true signal in the transform domain by solving a problem of the form

$$\text{minimize} \quad \|\Phi z - y\|_2^2 + \lambda \|Wz\|_1 \tag{6}$$

where the variable is $z \in \mathbf{R}^n$. Here, $\Phi = [\phi_1 \cdots \phi_m]^T \in \mathbf{R}^{m \times n}$ is called the compressed sensing matrix, $\lambda > 0$ is the regularization parameter, and W is called the sparsifying transform. Compressed sensing has a variety of potential applications including analog-to-information conversion and sparse MRI.

When W is invertible, we can reformulate the compressed sensing problem (6) as the ℓ_1-regularized LSP

$$\text{minimize} \quad \|Ax - y\|_2^2 + \lambda \|x\|_1 \tag{7}$$

where the variable is $x \in \mathbf{R}^n$ and the problem data or parameters are $A = \Phi W^{-1} \in \mathbf{R}^{m \times n}$ and $y \in \mathbf{R}^m$. [If $x^\star$ solves (7), then $W^{-1}x^\star$ solves (6), and conversely, if $z^\star$ solves (6), then $x^\star = Wz^\star$ solves (7).] This problem is a basis pursuit denoising (BPDN) problem with the dictionary matrix A; see, e.g., [8] for more on basis-pursuit denoising.

Before proceeding, we should mention an important feature of the ℓ_1-regularized LSPs that arise in compressed sensing. The data matrix A in the ℓ_1-regularized LSP (7) is typically fully dense, and so its equivalent QP formulation is fully dense. But the equivalent dense QP has an important difference from general dense QPs: there are fast algorithms for the matrix-vector operations with A and A^T, based on fast algorithms for the sparsifying transform and its inverse transform.

B. ℓ_1 Regularized Linear Regression

Let $u \in \mathbf{R}^n$ denote a vector of explanatory or feature variables, and $y \in \mathbf{R}$ denote the associated output. A linear model predicts the output as

$$\hat{y} = \beta^T u + \beta_{\text{bias}},$$

where $\beta_{\text{bias}} \in \mathbf{R}$ is the bias or intercept and $\beta \in \mathbf{R}^n$ is the weight vector.

Suppose we are given a set of (observed or training) examples, $(u_i, y_i) \in \mathbf{R}^n \times \mathbf{R}, i = 1, \ldots, m$. To estimate the weight coefficients and intercept, the Lasso algorithm [52] solves the ℓ_1-regularized LSP

$$\text{minimize} \quad \sum_{i=1}^{m} (\beta^T u_i + \beta_{\text{bias}} - y_i)^2 + \sum_{i=1}^{n} \lambda |\beta_i| \tag{8}$$

with variables $\beta_{\text{bias}} \in \mathbf{R}$ and $\beta \in \mathbf{R}^n$. The Lasso problem (8) has the form (5) with the variables $x = (\beta, \beta_{\text{bias}}) \in \mathbf{R}^{n+1}$. Here, the bias β_{bias} is not regularized.

Extensive research has shown that ℓ_1-regularized linear regression can outperform ℓ_2-regularized linear regression (also

called ridge regression), especially when the number of observations is smaller than the number of features [15], [52]. Recently, theoretical properties of ℓ_1-regularized linear regression have been studied by several researchers; see, e.g., [20], [29], [33], [56], [62], [61].

III. Optimality Conditions and Dual Problem

In this section, we give some preliminaries needed later.

A. Optimality Conditions

The objective function of the ℓ_1-regularized LSP (3) is convex but not differentiable, so we use a first-order optimality condition based on subdifferential calculus. We can obtain the following necessary and sufficient conditions for x to be optimal for the ℓ_1-regularized LSP (3):

$$(2A^T(Ax-y))_i \in \begin{cases} \{+\lambda_i\} & x_i > 0, \\ \{-\lambda_i\} & x_i < 0, \\ [-\lambda_i, \lambda_i] & x_i = 0, \end{cases} \quad i = 1, \ldots, n.$$

We can now derive the formula (4) for $\lambda_{\max}$. The condition that 0 is optimal is that $(2A^Ty)_i \in [-\lambda, \lambda]$ for $i = 1, \ldots, n$, i.e., $\|2A^Ty\|_\infty \leq \lambda$.

B. Dual Problem and Suboptimality Bound

We derive a Lagrange dual of the ℓ_1-regularized LSP (3). We start by introducing a new variable $z \in \mathbf{R}^m$, as well as new equality constraints $z = Ax - y$, to obtain the equivalent problem

$$\begin{array}{ll} \text{minimize} & z^Tz + \lambda_i\|x\|_1 \\ \text{subject to} & z = Ax - y. \end{array} \tag{9}$$

Associating dual variables $\nu_i \in \mathbf{R}, i = 1, \ldots, m$ with the equality constraints $z_i = (Ax-y)_i$, the Lagrangian is

$$L(x, z, \nu) = z^Tz + \lambda\|x\|_1 + \nu^T(Ax - y - z).$$

The dual function is

$$\inf_{x,z} L(x, z, \nu) = \begin{cases} -(1/4)\nu^T\nu - \nu^Ty, & |(A^T\nu)_i| \leq \lambda_i, \quad i = 1, \ldots, m \\ -\infty, & \text{otherwise.} \end{cases}$$

The Lagrange dual of (9) is therefore

$$\begin{array}{ll} \text{maximize} & G(\nu) \\ \text{subject to} & |(A^T\nu)_i| \leq \lambda_i, \quad i = 1, \ldots, m \end{array} \tag{10}$$

where the dual objective $G(\nu)$ is

$$G(\nu) = -(1/4)\nu^T\nu - \nu^Ty.$$

(See, e.g., [3, ch. 5] or [2] for more on convex duality.) The dual problem (10) is a convex optimization problem with variable $\nu \in \mathbf{R}^m$. We say that $\nu \in \mathbf{R}^m$ is dual feasible if it satisfies the constraints of (10), i.e., $|(A^T\nu)_i| \leq \lambda_i, i = 1, \ldots, m$.

Any dual feasible point ν gives a lower bound on the optimal value $p^\star$ of the primal problem (3), i.e., $G(\nu) \leq p^\star$, which is called weak duality. Furthermore, the optimal values of the primal and dual are equal since the primal problem (3) satisfies Slater's condition, which is called strong duality [3].

An important property of the ℓ_1-regularized LSP (3) is that from an arbitrary x, we can derive an easily computed bound on the suboptimality of x, by constructing a dual feasible point

$$\begin{aligned} &\nu = 2s(Ax - y), \\ &s = \min\{\lambda/|2((A^TAx)_i - 2y_i| \mid i = 1, \ldots, m\}. \end{aligned} \tag{11}$$

The point ν is dual feasible, so $G(\nu)$ is a lower bound on $p^\star$, the optimal value of the ℓ_1-regularized LSP (3).

The difference between the primal objective value of x and the associated lower bound $G(\nu)$ is called the *duality gap*. We use η to denote the gap

$$\eta = \|Ax - y\|_2^2 + \lambda\|x\|_1 - G(\nu). \tag{12}$$

The duality gap is always nonnegative by weak duality, and x is no more than η-suboptimal. At an optimal point, the duality gap is zero, i.e., strong duality holds.

IV. A Truncated Newton Interior-Point Method

The ℓ_1-regularized LSP (3) can be transformed to a convex quadratic problem, with linear inequality constraints

$$\begin{array}{ll} \text{minimize} & \|Ax - y\|_2^2 + \lambda\sum_{i=1}^n u_i \\ \text{subject to} & -u_i \leq x_i \leq u_i, \quad i = 1, \ldots, n \end{array} \tag{13}$$

where the variables are $x \in \mathbf{R}^n$ and $u \in \mathbf{R}^n$. In this section, we describe an interior-point method for solving this equivalent QP.

A. A Custom Interior-Point Method

We start by defining the logarithmic barrier for the bound constraints $-u_i \leq x_i \leq u_i$ in (13)

$$\Phi(x, u) = -\sum_{i=1}^n \log(u_i + x_i) - \sum_{i=1}^n \log(u_i - x_i)$$

defined over $\mathbf{dom}\Phi = \{(x, u) \in \mathbf{R}^n \times \mathbf{R}^n \mid |x_i| < u_i, i = 1, \ldots, n\}$. The central path consists of the unique minimizer $(x^\star(t), u^\star(t))$ of the convex function

$$\phi_t(x, u) = t\|Ax - y\|_2^2 + t\sum_{i=1}^n \lambda u_i + \Phi(x, u)$$

as the parameter t varies from 0 to ∞. With each point $(x^\star(t), u^\star(t))$ on the central path we associate $\nu^\star(t) = 2(Ax^\star(t) - y)$, which can be shown to be dual feasible. (Indeed, it coincides with the dual feasible point ν constructed from $x^\star(t)$ using the method of Section III-B.) In particular, $(x^\star(t), u^\star(t))$ is no more than $2n/t$-suboptimal, so the central path leads to an optimal solution.

In a primal interior-point method, we compute a sequence of points on the central path, for an increasing sequence of values of t, starting from the previously computed central point. (A typical method uses the sequence $t = t_0, \mu t_0, \mu^2 t_0, \ldots$, where μ is between 2 and 50 [3, Sec. 11.3]. The method can be terminated when $2n/t \leq \epsilon$, where ϵ is the target duality gap, since then we can guarantee ϵ-suboptimality of $(x^\star(t), u^\star(t))$. (The reader is referred to [3, Ch. 11] for more on the primal barrier method.) In the primal barrier method, Newton's method is used to minimize

ϕ_t, i.e., the search direction is computed as the exact solution to the Newton system

$$H\begin{bmatrix}\Delta x\\ \Delta u\end{bmatrix} = -g \tag{14}$$

where $H = \nabla^2\phi_t(x,u) \in \mathbf{R}^{2n\times 2n}$ is the Hessian and $g = \nabla\phi_t(x,u) \in \mathbf{R}^{2n}$ is the gradient at the current iterate (x,u).

For a large ℓ_1-regularized LSP, solving the Newton system (14) exactly is not computationally practical. We need to find a search direction which gives a good trade-off of computational effort versus the convergence rate it provides. In the method described below, the search direction is computed as an approximate solution to the Newton system (14), using PCG. When an iterative method is used to approximately solve the Newton system, the overall method is called a truncated Newton method. Truncated Newton methods have been applied to interior-point methods; see, e.g., [27], [30], [55], and [44].

In the primal barrier method, the parameter t is held constant until ϕ_t is (approximately) minimized, i.e., $\|\nabla\phi_t\|_2$ is small. For faster convergence, however, we can update the parameter t at each iteration, based on the current duality gap, computed using the dual feasible point constructed as described in Section III-B. This leads to the following algorithm.

TRUNCATED NEWTON IPM FOR ℓ_1-REGULARIZED LSPS.

---

given relative tolerance $\epsilon_{\rm rel} > 0$.

Initialize. $t := 1/\lambda, x := 0, u := \mathbf{1} = (1,\ldots,1) \in \mathbf{R}^n$.

repeat

1. Compute the search direction $(\Delta x, \Delta u)$ as an approximate solution to the Newton system (14).
2. Compute the step size s by backtracking line search.
3. Update the iterate by $(x,u) := (x,u) + s(\Delta x, \Delta u)$.
4. Construct a dual feasible point ν from (11).
5. Evaluate the duality gap η from (12).
6. **quit** if $\eta/G(\nu) \le \epsilon_{\rm rel}$.
7. Update t.

As a stopping criterion, the method uses the duality gap divided by the dual objective value. By weak duality, the ratio is an upper bound on the relative suboptimality

$$\frac{f(x)-p^\star}{p^\star} \le \frac{\eta}{G(\nu)},$$

where $p^\star$ is the optimal value of the ℓ_1-regularized LSP (3) and $f(x)$ is the primal objective computed with the point x (computed in step 3). Therefore, the method solves the problem to guaranteed relative accuracy $\epsilon_{\rm rel}$.

Given the search direction $(\Delta x, \Delta u)$, the new point is $(x,u) + s(\Delta x, \Delta u)$, where $s \in \mathbf{R}_+$, the step size, is to be computed. In the backtracking line search, the step size is taken as $s = \beta^k$, where $k \ge 0$ is the smallest integer that satisfies

$$\phi_t(x+\beta^k\Delta x, u+\beta^k\Delta u) \le \phi_t(x,u) + \alpha\beta^k\nabla\phi_t(x,u)^T\,[\Delta x\ \Delta u]$$

where $\alpha \in (0,1/2)$ and $\beta \in (0,1)$ are algorithm parameters. (Typical values for the line search parameters are $\alpha = 0.01, \beta = 0.5$.) The reader is referred to [3, ch. 9] for more on the backtracking line search.

We use the update rule

$$t := \begin{cases}\max\{\mu\min\{2n/\eta, t\}, t\}, & s \ge s_{\min}\\ t, & s < s_{\min}\end{cases}$$

where $\mu > 1$ and $s_{\min} \in (0,1]$ are parameters to be chosen. The same type of update rule was used in the custom interior-point method for ℓ_1-regularized logistic regression described in [30]. The choice of $\mu = 2$ and $s_{\min} = 0.5$ appears to give good performance for a wide range of problems. This update rule uses the step length s as a crude measure of proximity to the central path. In particular, $2n/t$ is the value of t for which the associated central point has the same duality gap as the current point. The update rule appears to be quite robust and work well when combined with the PCG algorithm we will describe soon. The reader is referred to [30] for an informal justification of convergence of the interior-point method based on this update rule (with exact search directions).

B. Search Direction via PCGs

We can find compact representations of the Hessian and gradient. The Hessian can be written as

$$H = t\nabla^2\|Ax-y\|_2^2 + \nabla^2\Phi(x,u) = \begin{bmatrix}2tA^TA + D_1 & D_2\\ D_2 & D_1\end{bmatrix}$$

where

$$D_1 = \operatorname{diag}\left(\frac{2\left(u_1^2+x_1^2\right)}{\left(u_1^2-x_1^2\right)^2},\ldots,\frac{2\left(u_n^2+x_n^2\right)}{\left(u_n^2-x_n^2\right)^2}\right) \in \mathbf{R}^n$$

$$D_2 = \operatorname{diag}\left(\frac{-4u_1x_1}{\left(u_1^2-x_1^2\right)^2},\ldots,\frac{-4u_nx_n}{\left(u_n^2-x_n^2\right)^2}\right) \in \mathbf{R}^n.$$

Here, we use $\operatorname{diag}(A_1,\ldots,A_p)$ to denote the diagonal matrix with diagonal blocks $A_1,\ldots,A_p$. The Hessian H is symmetric and positive definite. The gradient can be written as

$$g = \begin{bmatrix}g_1\\ g_2\end{bmatrix} \in \mathbf{R}^{2n}$$

where

$$g_1 = \nabla_x\phi_t(x,u) = 2tA^T(Ax-y) + \begin{bmatrix}2x_1/(u_1^2-x_1^2)\\ \vdots\\ 2x_n/(u_n^2-x_n^2)\end{bmatrix} \in \mathbf{R}^n$$

$$g_2 = \nabla_u\phi_t(x,u) = t\lambda\mathbf{1} - \begin{bmatrix}2u_1/(u_1^2-x_1^2)\\ \vdots\\ 2u_n/(u_n^2-x_n^2)\end{bmatrix} \in \mathbf{R}^n.$$

We compute the search direction approximately, applying the PCG algorithm [10, Sect. 6.6] to the Newton system (14). The PCG algorithm uses a preconditioner $P \in \mathbf{R}^{2n\times 2n}$, which is symmetric and positive definite. We will not go into the details of the PCG algorithm, and refer the reader to [28], [48, Sect. 6.7], or [[39, Sect. 5].

The preconditioner used in the PCG algorithm approximates the Hessian of $t\|Ax-y\|_2^2$ with its diagonal entries, while retaining the Hessian of the logarithmic barrier $\Phi(x,u)$

$$P = \begin{bmatrix} 2t\,\mathrm{diag}(A^TA) & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} D_1 & D_2 \\ D_2 & D_1 \end{bmatrix}. \qquad (15)$$

Here, $\mathrm{diag}(S)$ is the diagonal matrix obtained by setting the off-diagonal entries of the matrix S to zero.

The cost of computing the diagonal entries of A^TA can be amortized over all interior-point iterations and multiple problems with the same data matrix and different observation vectors, since we need to compute them only once. When the amortized cost is still expensive, we can approximate the diagonal matrix $\mathrm{diag}(A^TA)$ with a scaled identity matrix to obtain the preconditioner

$$P = \begin{bmatrix} 2\tau tI & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} D_1 & D_2 \\ D_2 & D_1 \end{bmatrix} \qquad (16)$$

where τ is a positive constant. This preconditioner performs well especially when the diagonal elements of A^TA show relatively small variations.

The PCG algorithm needs a good initial search direction and an effective truncation rule.

- *Initial point* There are many choices for the initial search direction, e.g., 0, the negative gradient, and the search direction found in the previous step. Numerical experiments suggest that initializing the PCG algorithm with the previous search direction appears to have a small advantage over the other two.
- *Truncation rule* The truncation rule for the PCG algorithm gives the condition for terminating the algorithm. The truncation rule in our implementation is simple: the PCG algorithm stops when the cumulative number of PCG steps exceeds the given limit N_{pcg} or we compute a point with relative tolerance less than ϵ_{pcg}. We use the adaptive relative tolerance change rule

$$\epsilon_{\mathrm{pcg}} = \min\{0.1, \xi\eta/\|g\|_2\},$$

where η is the duality gap at the current iterate and ξ is an algorithm parameter. (The choice of $\xi = 0.01$ appears to work well for a wide range of problems.) In other words, we solve the Newton system with low accuracy (but never worse than 10%) at early iterations, and solve it more accurately as the duality gap decreases.

Each iteration of the PCG algorithm involves a handful of inner products, a matrix-vector product Hp with $p = (p_1, p_2) \in \mathbf{R}^{2n}$ and a solve step with P in computing $P^{-1}r$ with $r = (r_1, r_2) \in \mathbf{R}^{2n}$. (Here, we use the convention that for column vectors a and b, (a,b) is the column vector obtained by stacking a on top of b.) The solve step $P^{-1}r$ can be computed as

$$P^{-1}r = \begin{bmatrix} (D_1D_3 - D_2^2)^{-1}(D_1r_1 - D_2r_2) \\ (D_1D_3 - D_2^2)^{-1}(-D_2r_1 + D_3r_2) \end{bmatrix}$$

where $D_3 = 2t\,\mathrm{diag}(A^TA) + D_1$ for the preconditioner (15) and $D_3 = 2\tau tI + D_1$ for the preconditioner (16). The computational cost is $O(n)$ flops.

The computationally most expensive operation for a PCG step is therefore the matrix-vector product Hp with $p = (p_1, p_2) \in \mathbf{R}^{2n}$. The product can be computed as

$$Hp = \begin{bmatrix} 2tA^Tv + D_1p_2 \\ D_2p_1 + D_1p_2 \end{bmatrix},$$

where $v = Ap_1$. The complexity of Hp depends on the problem data and determines the cost of a PCG step. We consider two cases.

- *Sparse problems*. The cost of computing Hp primarily depends on the number of nonzeros in the matrix A. The cost is $O(p)$ flops when A has p nonzero elements.
- *Dense problems with fast matrix-vector multiplications algorithms*. The cost of computing Hp is $O(nm)$ if no structure is exploited. If fast algorithms are available for the matrix-vector multiplications with A and A^T, the cost can be reduced substantially. As an example we consider the compressed sensing problem (6) where Φ is a Fourier matrix and W is an orthogonal wavelet matrix ($W^{-1} = W^T$). The matrix-vector multiplication $Au = \Phi W^{-1}u$ with $u \in \mathbf{R}^n$ can be done efficiently using the fast algorithms for the inverse wavelet transform and the DFT. The multiplication $A^Tv = W^{-T}\Phi^Tv = W\Phi^Tv$ can be computed efficiently in a similar manner using the fast algorithm for the wavelet transform. For any $v \in \mathbf{R}^m$, we can efficiently compute $\Phi^T \in \mathbf{R}^m$ using zero-filling and then performing the inverse Fourier transform. The cost of a PCG step is therefore $O(n \log n)$. (The Hessian-vector product Hp involves one FFT, one inverse FFT, one discrete wavelet transform (DWT), and one inverse DWT operation.)

C. Performance

Since the memory requirement of the truncated Newton interior-point method is modest, the method is able to solve very large problems, for which forming the Hessian H (let alone computing the search direction) would be prohibitively expensive. The runtime of the truncated Newton interior-point method is determined by the product of the total number of PCG steps required over all iterations and the cost of a PCG step. The total number of PCG iterations required by the truncated Newton interior-point method depends on the value of the regularization parameter λ and the relative tolerance ϵ_{rel}. In particular, for very small values of λ (which lead to solutions with relatively large nonzero coefficients), the truncated Newton interior-point method requires a larger total number of PCG steps.

In extensive testing, we found that the total number of PCG steps ranges between a few tens (for medium size problems) and several hundreds (for very large problems) to compute a solution which is never worse than 1% suboptimal, i.e., with relative tolerance 0.01. In particular, we observed that the total number of PCG steps remains modest (around a few hundred) nearly irrespective of the size of the problem, when the mutual coherence of the data matrix A is small, i.e., the off-diagonal elements of A^TA are relatively small compared with the diagonal elements. (This observation is not very surprising, since as the mutual coherence tends to zero, the Hessian in the Newton system (14) becomes more and more diagonally dominant.)

V. Numerical Examples

In this section, we give some numerical examples to illustrate the performance of the truncated Newton interior-point method (TNIPM) described in Section IV. The algorithm parameters are taken as

$$\alpha = 0.01, \quad \beta = 0.5, \quad s_{\min} = 0.5$$
$$\mu = 2, \quad \xi = 0.01, \quad N_{\mathrm{pcg}} = 200.$$

Since the iteration limit N_{pcg} in the PCG algorithm is set to be large enough, it was never reached in our experiments.

We compare the performance of the Matlab implementation the truncated Newton interior-point method (available from http://www.stanford.edu/~boyd/l1_ls/) and the following five existing solvers: `MOSEK`, `PDCO-CHOL`, `PDCO-LSQR`, `11-MAGIC`, and `HOMOTOPY`. `MOSEK` [34] is a C implementation of a primal-dual interior-point method with Matlab interface. `PDCO-CHOL` is a Matlab implementation of a primal-dual interior-point method [50] that uses the Cholesky factorization algorithm to compute the search direction. `PDCO-LSQR` is a Matlab implementation of a primal-dual interior-point method [50] that uses the LSQR algorithm [42] to compute the search direction. (`PDCO-CHOL` and `PDCO-LSQR` are available in `SparseLab` [13], a library of Matlab routines for finding sparse solutions to underdetermined systems.) `11-MAGIC` [5] is a Matlab package devoted to compressed sensing problems. `PDCO-LSQR` and `11-MAGIC` implement interior-point methods that use the CG or LSQR method to compute the search direction and hence are similar in spirit to our method. `Homotopy` is an implementation of the homotopy method [14], available in `Sparselab`.

The existing methods and TNIPM were run on an AMD 270 under Linux.

A. A Sparse Signal Recovery Example

As an example, we consider a sparse signal recovery problem with a signal $x \in \mathbf{R}^{4096}$ which consists of 160 spikes with amplitude ± 1, shown at the top plot of Fig. 1. The measurement matrix $A \in \mathbf{R}^{1024\times 4096}$ is created by first generating a matrix of size $\mathbf{R}^{1024\times 4096}$ with entries generated independently and identically according to the standard normal distribution and then orthogonalizing the rows, as with the example in [5, Sect. 3.1]. In the problem, we have

$$y = Ax + v$$

where v is drawn according to the Gaussian distribution $\mathcal{N}(0, 0.01^2 I)$ on $\mathbf{R}^{1024}$.

Our method and the five existing methods above were run to find a point which is not worse than 1% suboptimal, i.e., with relative tolerance 0.01. The regularization parameter was taken as $\lambda = 0.01\lambda_{\max}$, where the value of $\lambda_{\max}$ was computed using the formula given in (4). Table I compares the runtimes of the Matlab implementation of our method and the five existing methods. The truncated Newton interior-point method is most efficient for this medium problem.

Fig. 1 shows the reconstruction results. The top plot shows the original signal x. The middle plot shows the signal $x^{\mathrm{me}} = A^{\dagger}y$, called the minimum energy reconstruction, (which is the point in the set $\{x \in \mathbf{R}^{4096} \mid A^T Ax = A^T y\}$ closest to the origin). The bottom plot shows the signal x^{l1} obtained by solving the BPDN problem. The ℓ_1-regularization based method finds exactly the positions of nonzeros in the original signal (after ap-

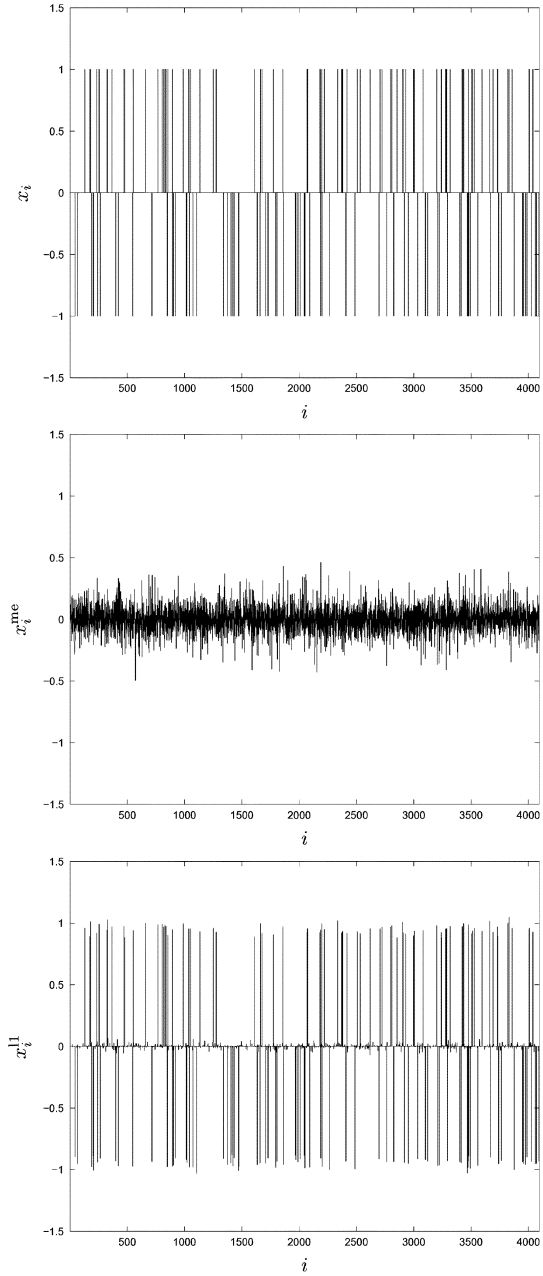


Fig. 1. Sparse signal reconstruction. Top: original signal. Middle: minimum energy reconstruction. Bottom: BPDN reconstruction.

TABLE I
Runtimes of the Truncated Newton Interior-Point Method (TNIPM) and Five Existing Methods, for a Signal Denoising Problem

Method	Runtime (sec)
`TNIPM`	7
`MOSEK`	141
`PDCO-CHOL`	72
`PDCO-LSQR`	109
`l1-magic`	538
`Homotopy`	11

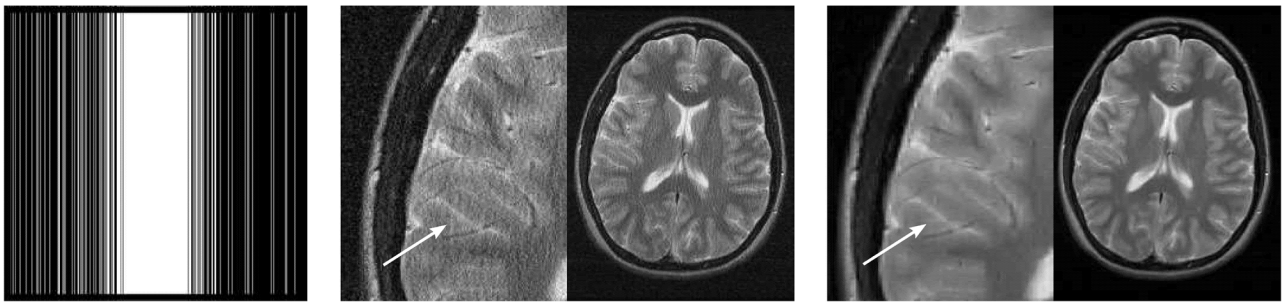


Fig. 2. Brain image reconstruction results. *Left*. Collected partial Fourier coefficients (in white). *Middle*. Linear reconstruction. *Right*. Compressed sensing reconstruction.

propriate thresholding), although the number of measurements is far less than the number of unknowns. (After identifying the nonzero positions in the reconstructed signal, we can use a standard least-squares method to estimate the values in the nonzero positions, over all signals with the given nonzero positions.) The minimum energy reconstruction method does not identify the nonzero positions at all. We applied Tikhonov regularization with a wide range of the regularization parameters to estimate the signal but observed similar results.

B. A Compressed Sensing Example in Rapid MRI

As another example, we consider a sparse signal recovery problem in MRI, using the idea of compressed sensing. We scanned the brain of a healthy volunteer, obtaining 205 out of 512 possible parallel lines in the spatial frequency of the image. The lines were chosen randomly with higher density sampling at low frequency, achieving a 2.5 scan-time reduction factor, as illustrated in the left panel of Fig. 2. The compressed sensing matrix Φ in (6) is therefore a matrix obtained by randomly removing many rows of the 2-D DFT matrix, called a partial Fourier ensemble. Brain images have a sparse representation in the wavelet domain. In the example shown, we use the Daubechies 4 wavelet transform as the sparsifying transform W in (6).

We compared the compressed sensing or sparse MRI method with a linear reconstruction method, which sets unobserved Fourier coefficients to zero and then performs the inverse Fourier transform. For compressed sensing reconstruction, we solved the problem with relative tolerance $\epsilon_{\rm rel} = 0.05$ and the regularization parameter $\lambda = 0.01$. Fig. 2 shows the two reconstruction results. The linear reconstruction suffers from incoherent noise-like streaking artifacts (pointed by the arrow) due to undersampling, whereas the artifacts are much less noticeable in the compressed sensing reconstruction.

The compressed sensing reconstruction problem can be reformulated as a QP, which has around $4 \times 512^2 \approx 10^6$ variables. (One half are the real and imaginary wavelet coefficients and the other half are new variables added in transforming the compressed sensing problem into a QP.) The run time of the Matlab implementation of our interior-point method to solve the QP with $\epsilon_{\rm rel} = 0.05$ was a few minutes, and the total number of PCG steps required over all interior-point iterations was 137. The exact relative tolerance from the optimal objective value (which was computed using our method with very small relative tolerance 10^{-6}) was below 1%. `MOSEK` could not handle the problem, since forming the Hessian H (let alone computing the search direction) is prohibitively expensive for direct methods. Some of the existing solvers mentioned above could handle the problem but were much slower than our method, by two orders of magnitude.

C. Scalability

To examine the effect of problem size on the runtime of the truncated Newton interior-point method, we generated a family of ten sparse data sets, with the number of features n varying from one thousand to one million, and $m = 0.1n$ examples, i.e., 10 times more features than examples. In doing so, the sparsity of A was controlled so that the total number of nonzero entries in A was $p \approx 30$m. The elements of A were independent and identically distributed, drawn from the standard normal distribution. For each data set, x was generated to be sparse with $0.25n$ nonzero elements. The measurements Ax were corrupted by white noise with zero mean and variance $0.01^2 I$. The regularization parameter was taken as $\lambda = 0.1\lambda_{\max}$.

Fig. 3 summarizes the scalability comparison results for the regularization parameter, when our method and the four existing methods except for `Homotopy` were run to find a point which is not worse than 1% suboptimal, i.e., with relative tolerance 0.01.(`Homotopy` was excluded in the scalability comparison, since the current implementation available in [5] does not handle sparse data effectively.) Evidently the truncated Newton interior-point method is more efficient for small problems, and far more efficient for medium and large problems. By fitting an exponent to the data over the range from $n = 1000$ to the largest problem successfully solved by each method, we found that the empirical complexities of our method and the four existing methods. The empirical complexity of `TNIPM` was $O(n^{1.2})$ and that of `l1-magic` was $O(n^{1.3})$. Empirical complexities of other solvers were more than quadratic.

VI. Extensions and Variations

We can solve the general ℓ_1-regularized LSP (5), using The Truncated Newton interior-point method described in Section IV. We can extend the idea behind this method to some other problems involving ℓ_1 regularization. The most important part in the extensions is to find a preconditioner that gives a good trade-off between the computational complexity and the accelerated convergence it provides. We will focus on this part,

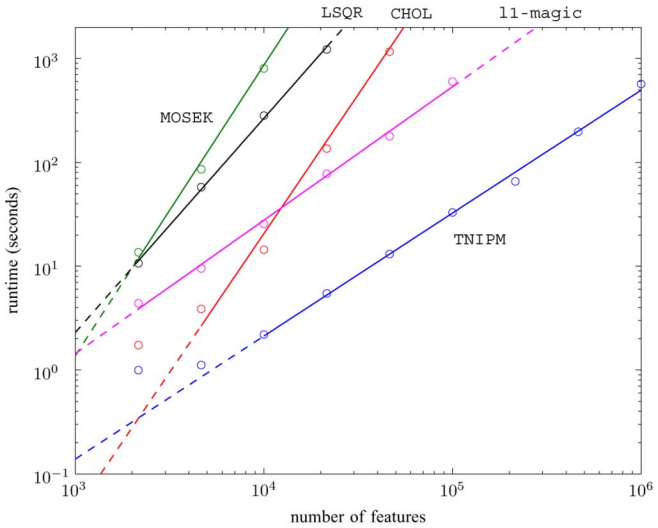


Fig. 3. Runtime of the proposed truncated Newton interior-point method (`TNIPM`), `MOSEK`, `PDCO-CHOL` (abbreviated as CHOL), `PDCOLSQR` (abbreviated as LSQR), and `l1-magic`, for ten randomly generated sparse problems, with the regularization parameter.

and will not go into the details of other parts which are rather straightforward.

A. General ℓ_1-Regularized LSPs

The general ℓ_1-regularized LSP (5) can be reformulated as

$$\text{minimize} \quad \inf_{x_i, i\in J} \|Ax-y\|_2^2 + \sum_{i\notin J} \lambda_i |x_i|$$

where $J = \{i \in \{1,\dots,n\} \,|\, \lambda_i = 0\}$ and the variables are $x_i, i \notin J$. The first term in the objective can be evaluated analytically and is in fact a quadratic function of the variables $x_i, i \notin J$. This problem can be cast as a problem of the form (3) via a simple transform of variables, $z_i = (\lambda_i/\lambda)x_i$, and hence can be solved using the truncated Newton interior-point method described above.

We should point out that the method does not need to form explicitly the data matrix of the equivalent ℓ_1-regularized LSP; it needs an algorithm for multiplying a vector by the data matrix of the equivalent formulation and an algorithm for multiplying a vector by its transpose. These matrix-vector multiplications can be performed with the nearly same computational cost as those with the original data matrix A and its transpose A^T, provided that the number of unregularized variables (i.e., $x_i, i \in J$) is small.

As an example, we consider the Lasso problem (8). It can be transformed to a problem of the form

$$\text{minimize} \quad \|Ax - y + (1/m)\mathbf{1}\mathbf{1}^T y\|_2^2 + \lambda\|x\|_1$$

where the variables are $x = \beta \in \mathbf{R}^n$ and the problem data are

$$A = \begin{bmatrix} u_1^T \\ \vdots \\ u_m^T \end{bmatrix} - \frac{1}{m}\mathbf{1}\sum_{i=1}^m u_i^T \in \mathbf{R}^{m\times n}.$$

(Here, $\mathbf{1}$ is the vector of all ones whose dimension is clear from the context.) The data matrix A in the equivalent formulation is the sum of the original data matrix and a rank-one matrix, so the matrix-vector multiplications with A and A^T can be performed with the same computational cost as those with the original data matrix and its transpose.

B. ℓ_1-Regularized LSPS With Nonnegativity Constraints

Suppose that the variable $x \in \mathbf{R}^n$ is known to be nonnegative. We add the nonnegativity constraint to the ℓ_1-regularized LSP (3), to obtain

$$\begin{aligned} &\text{minimize} \quad \|Ax-y\|_2^2 + \lambda\|x\|_1 = \|Ax-y\|_2^2 + \lambda\sum_{i=1}^n x_i \\ &\text{subject to} \quad x_i \geq 0, \quad i = 1,\dots,n. \end{aligned}$$

The associated centering problem is to minimize the weighted objective function augmented by the logarithmic barrier for the constraints $x_i \geq 0$

$$t\|Ax-y\|_2^2 + t\lambda\sum_{i=1}^n x_i - \sum_{i=1}^n \log x_i.$$

The Newton system for the centering problem is

$$(2tA^TA + D)\Delta x_{\text{nt}} = -g$$

where

$$D = \begin{bmatrix} 1/x_1^2 \\ \vdots \\ 1/x_1^2 \end{bmatrix}, \quad g = 2tA^T(Ax-y) + \begin{bmatrix} t\lambda_1 - 1/x_1 \\ \vdots \\ t\lambda_n - 1/x_n \end{bmatrix}.$$

The diagonal preconditioner of the form

$$P = \text{diag}(2tA^TA) + D$$

works well for this problem, especially when the optimal solution is sparse.

C. Isotonic Regression With ℓ_1 Regularization

We consider a variation of the ℓ_1-regularized LSP (3) subject to the monotonicity constraints

$$x_1 \leq \cdots \leq x_n$$

which can be written as

$$\begin{aligned} &\text{minimize} \quad \|Ax-y\|_2^2 + \lambda\|x\|_1 \\ &\text{subject to} \quad x_1 \leq \cdots \leq x_n. \end{aligned}$$

This problem is related to the monotone Lasso [25], which is an extension of isotonic regression which has been extensively studied in statistics [1], [45].

The ℓ_1-regularized isotonic regression problem arises in several contexts. As an example, the problem of finding monotone trends in a Gaussian setting can be cast as a problem of this form [49], [21]. As another example, the problem of estimating accumulating damage trend from a series of structural health monitoring (SHM) images can be formulated as a problem of the

form in which the variables are 3-D (a time series of images) [22].

We take the change of variables

$$z_1 = x_1, \quad z_k = x_k - x_{k-1}, \quad k = 2, \ldots, n$$

to reformulate the isotonic regression problem as an ℓ_1-regularized LSP with monotonicity constraints

$$\begin{array}{ll} \text{minimize} & \|y - \bar{A}z\|_2^2 + \sum_{k=2}^{n} \lambda z_k \\ \text{subject to} & z_k \geq 0, \quad k = 1, \ldots, n \end{array}$$

where the new data matrix is

$$\bar{A} = AG, \quad G = \begin{bmatrix} 1 & & & & \\ 1 & 1 & & & \\ 1 & 1 & \ddots & & \\ \vdots & \vdots & \ddots & 1 & \\ 1 & 1 & \cdots & 1 & 1 \end{bmatrix} \in \mathbf{R}^{n \times n}.$$

(The entries of G not shown above are zero.) This problem can be solved by a truncated Newton interior-point method with the preconditioner described above, exploiting the structure of the new data matrix. The two operations, multiplying a vector by $\bar{A}$ and multiplying a vector by its transpose, can be done as efficiently as those with A, using the fact that the corresponding operations with G and G^T can be done in $O(n)$ flops.

D. ℓ_1-Regularized LSPs With Complex Variables

The truncated Newton interior-point method can be extended to a problem of the form

$$\text{minimize} \quad \|Az - b\|_2^2 + \lambda \|z\|_1 \tag{17}$$

where the variable is $z \in \mathbf{C}^n$ and the problem data are $b \in \mathbf{C}^n$ and $A \in \mathbf{C}^{m \times n}$. Here $\|z\|_1$, the ℓ_1 norm of the complex vector z, is defined by

$$\|z\|_1 = \sum_{i=1}^{n} (x_i^2 + y_i^2)^{1/2} \tag{18}$$

where x is the real part of z and y is the imaginary part.

We note that (17) cannot be cast as an ℓ_1-regularized LSP of the form (3). Instead, it can be reformulated as the convex problem

$$\begin{array}{ll} \text{minimize} & \left\| \tilde{A} \begin{bmatrix} x \\ y \end{bmatrix} - \tilde{b} \right\|_2^2 + \lambda \mathbf{1}^T u \\ \text{subject to} & \sqrt{x_i^2 + y_i^2} \leq u_i, \quad i = 1, \ldots, n \end{array} \tag{19}$$

where the variables are $x, y, u \in \mathbf{R}^n$ and the problem data are

$$\tilde{A} = \begin{bmatrix} \mathbf{Re}A & -\mathbf{Im}A \\ \mathbf{Im}A & \mathbf{Re}A \end{bmatrix} \in \mathbf{R}^{2m \times 2n},$$
$$\tilde{b} = \begin{bmatrix} \mathbf{Re}b \\ \mathbf{Im}b \end{bmatrix} \in \mathbf{R}^{2m}.$$

This problem is a second-order cone program (SOCP) and can be solved by standard interior-point methods; see, e.g., [31] for more on SOCPs.

We show how the truncated Newton interior-point method can be extended to the SOCP (19). Using the standard barrier function for second-order cone constraints, the associated centering problem can be formulated as

$$\text{minimize} \quad t \left\| \tilde{A} \begin{bmatrix} x \\ y \end{bmatrix} - \tilde{b} \right\|_2^2 + t\lambda \mathbf{1}^T u + \Phi(x, y, u) \tag{20}$$

where the variables are $x, y, u \in \mathbf{R}^n$ and $\Phi(x, y, u)$ is the barrier function

$$\Phi(x, y, u) = -\sum_{i=1}^{n} \log \left(u_i^2 - x_i^2 - y_i^2 \right)$$

for the constraints of (19). The Hessian of this problem is

$$H = t\tilde{A}^T \tilde{A} + \nabla^2 \Phi(x, y, u).$$

As before, we consider the preconditioner that approximates the Hessian of the first term in the objective of the centering problem (20) with its diagonal entries, while retaining the Hessian of the logarithmic barrier

$$P = t \operatorname{diag}(\tilde{A}^T \tilde{A}) + \nabla^2 \Phi(x, y, u).$$

After appropriate reordering, this preconditioner is a block diagonal matrix consisting of n 3×3 diagonal matrices, and so the computational effort of $P^{-1}r$ is $O(n)$. This preconditioner appears to be quite effective in solving the centering problem (20).

In compressed sensing, a problem of the form (17) arises in case when the entries of the matrices Φ and W are complex and we do not expand the real and imaginary parts of the matrices Φ and W. In this case, we have $A = \Phi W^{-1}$, which is a matrix with complex entries. The resulting formulation is different from the formulation, considered in Section II-A, obtained by expanding the real and imaginary parts of z, Φ, and W, with the ℓ_1 penalty function $\sum_{i=1}^{n} |x_i| + |y_i|$. Compared to the formulation described in Section II-A, the formulation based on the penalty function (18) couples together the real and imaginary parts of entries of z, so the optimal solution found tends to have more simultaneous zero real and imaginary entries in the entries. (The idea behind the penalty function (18) is used in the grouped Lasso [60] and in total variation minimization with two- or higher-dimensional data [7], [47].)

VII. Conclusion

In this paper, we have described a specialized interior-point method for solving large-scale ℓ_1-regularized LSPs. The method is similar in spirit to the specialized interior-point method for basis pursuit denoising described in [8]. A major difference is that our method uses the PCG algorithm to compute the search direction, whereas the method described in [8] uses the LSQR without preconditioning. Another difference lies in the truncation rule for the iterative algorithm: the truncation rule for the iterative algorithm used in our method is more aggressive to find

a search direction which gives a good tradeoff of computational effort versus the convergence rate it provides.

Our method can be generalized to handle a variety of extensions, as described in Section VI. The extension to total variation minimization with two- or three-dimensional data is more involved, since it is not straightforward to find a simple but effective preconditioner for the associated centering problem. We hope to extend the method to the total variation minimization problem in future work.

In many applications, the choice of an appropriate value of the regularization parameter λ involves computing a portion of the regularization path, or at the very least solving ℓ_1-regularized LSPs with multiple, and often many, values of λ. Incorporating a warm start technique into the truncated Newton interior-point method, we can compute a good approximation of the regularization path much more efficiently than by solving multiple problems independently. The idea has been successfully used in ℓ_1-regularized logistic regression [30].

ACKNOWLEDGMENT

The authors are grateful to the anonymous reviewers, E. Candès, M. Calder, J. Duchi, and M. Grant, for helpful comments.

REFERENCES

[1] R. Barlow, D. Bartholomew, J. Bremner, and H. Brunk, *Statistical Inference Under Order Restrictions: The Theory and Application of Isotonic Regression*. New York: Wiley, 1972.

[2] J. Borwein and A. Lewis, *Convex Analysis and Nonlinear Optimization*. New York: Springer, 2000.

[3] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge University Press, 2004.

[4] E. Candès, "Compressive sampling," *Proc. Int. Congr. Mathematics*, 2006.

[5] E. Candès and J. Romberg, *l_1-magic: A Collection of MATLAB Routines for Solving the Convex Optimization Programs Central to Compressive Sampling* 2006 [Online]. Available: www.acm.caltech.edu/l1magic/

[6] E. Candès, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Commun. Pure Appl. Math.*, vol. 59, no. 8, pp. 1207–1223, 2005.

[7] E. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[8] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 1, pp. 129–159, 2001.

[9] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Commun. Pure Appl. Mathe.*, vol. 57, pp. 1413–1541, 2004.

[10] J. Demmel, *Applied Numerical Linear Algebra*. Cambridge, U.K.: Cambidge Univ. Press, 1997.

[11] D. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[12] D. Donoho, M. Elad, and V. Temlyakov, "Stable recovery of sparse overcomplete representations in the presence of noise," *IEEE Trans. Inf. Theory*, vol. 52, no. 1, pp. 6–18, Jan. 2006.

[13] D. Donoho, V. Stodden, and Y. Tsaig, SparseLab: Seeking Sparse Solutions to Linear Systems of Equations 2006 [Online]. Available: http://sparselab.stanford.edu/

[14] D. Donoho and Y. Tsaig, "Fast solution of l_1-norm minimization problems when the solution may be sparse," *Manuscript* 2006 [Online]. Available: http://www.stanford.edu/

[15] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *Ann. Statist.*, vol. 32, no. 2, pp. 407–499, 2004.

[16] M. Elad, B. Matalon, and M. Zibulevsky, "Image denoising with shrinkage and redundant representations," in *Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR)*, 2006, vol. 2, pp. 1924–1931.

[17] M. Figueiredo and R. Nowak, "A bound optimization approach to wavelet-based image deconvolution," in *Proc. IEEE Int. Conf. Image Processing (ICIP)*, 2005, pp. 782–785.

[18] M. Figueiredo, R. Nowak, and S. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," *IEEE J. Select. Topics Signal Process.*, 2007.

[19] J. Friedman, T. Hastie T, and R. Tibshirani, Pathwise Coordinate Optimization 2007 [Online]. Available: www-stat.stanford.edu/hastie/pub.htm

[20] J. Fuchs, "Recovery of exact sparse representations in the presence of noise," *IEEE Trans. Inf. Theory*, vol. 51, no. 10, pp. 3601–3608, Oct. 2005.

[21] D. Gorinevsky, "Monotonic regression filters for trending gradual deterioration faults," in *Proc. American Control Conf. (ACC)*, 2004, pp. 5394–5399.

[22] D. Gorinevsky, S.-J. Kim, S. Bear, S. Boyd, and G. Gordon, "Optimal estimation of deterioration from diagnostic image sequence," *IEEE Trans. Signal Process.* 2007 [Online]. Available: http://www.stanford.edu/gorin/papers/MonoImage07twocol.pdf, Available from, Submitted to

[23] E. Hale, W. Yin, and Y. Zhang, "A fixed-point continuation method for ℓ_1 regularized minimization with applications to compressed sensing," *Manuscript* 2007 [Online]. Available: http://www.dsp.ece.rice.edu/cs/

[24] T. Hastie, S. Rosset, R. Tibshirani, and J. Zhu, "The entire regularization path for the support vector machine," *J. Mach. Learning Res.*, vol. 5, pp. 1391–1415, 2004.

[25] T. Hastie, J. Taylor, R. Tibshirani, and G. Walther, "Forward stagewise regression and the monotone lasso," *Electron. J. Statist.*, vol. 1, no. 1–29, 2007.

[26] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. New York: Springer-Verlag, 2001, Springer Series in Statistics.

[27] C. Johnson, J. Seidel, and A. Sofer, "Interior point methodology for 3-D PET reconstruction," *IEEE Trans. Med. Imag.*, vol. 19, no. 4, pp. 271–285, 2000.

[28] C. T. Kelley, "Iterative methods for linear and nonlinear equations," *Frontiers in Applied Mathematics*, vol. 16, 1995, SIAM: Philadelphia, PA.

[29] K. Knight and W. Fu, "Asymptotics for lasso-type estimators," *Ann. Statist.*, vol. 28, no. 5, pp. 1356–1378, 2000.

[30] K. Koh, S.-J. Kim, and S. Boyd, "An interior-point method for ℓ_1-regularized logistic regression," *J. Mach. Learning Res.*, vol. 8, pp. 1519–1555, 2007.

[31] M. Lobo, L. Vandenberghe, S. Boyd, and H. Lebret, "Applications of second-order cone programming," *Linear algebra and its Applications*, vol. 284, pp. 193–228, 1998.

[32] D. Luenberger, *Linear and Nonlinear Programming*, 2nd ed. Reading, MA: Addison-Wesley, 1984.

[33] N. Meinshausen and P. Büuhlmann, "High-dimensional graphs and variable selection with the lasso," *Ann. Statist.*, vol. 34, no. 3, pp. 1436–1462, 2006.

[34] The MOSEK Optimization Tools Version 2.5. User's Manual and Reference 2002 [Online]. Available: www.mosek.com, MOSEK ApS Available from

[35] G. Narkiss and M. Zibulevsky, Sequential Subspace Optimization Method for Large-Scale Unconstrained Problems The Technion, Haifa, Israel, Tech. Rep. CCIT No 559, 2005.

[36] Y. Nesterov, "Gradient methods for minimizing composite objective function," 2007, CORE Discussion Paper 2007/76 [Online]. Available: http://www.optimization-online.org/DB_HTML/2007/09/1784.html

[37] Y. Nesterov and A. Nemirovsky, "Interior-point polynomial methods in convex programming," *Studies in Applied Mathematics*, vol. 13, 1994, SIAM: Philadelphia, PA.

[38] A. Neumaier, "Solving ill-conditioned and singular linear systems: A tutorial on regularization," *SIAM Rev.*, vol. 40, no. 3, pp. 636–666, 1998.

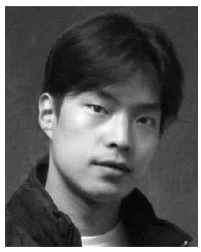
[39] J. Nocedal and S. Wright, "Numerical optimization," *Springer Series in Operations Research*, 1999, Springer.

[40] M. Osborne, B. Presnell, and B. Turlach, "A new approach to variable selection in least squares problems," *IMA J. Numer. Anal.*, vol. 20, no. 3, pp. 389–403, 2000.

[41] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin, "An iterative regularization method for total variation based image restoration," *SIAM J. Multiscale Modeling and Simulation*, vol. 4, no. 2, pp. 460–489, 2005.
[42] C. Paige and M. Saunders, "LSQR: An algorithm for sparse linear equations and sparse least squares," *ACM Trans. Mathemat. Software*, vol. 8, no. 1, pp. 43–71, 1982.
[43] B. Polyak, "Introduction to optimization," *Optimization Software*, 1987, Translated from Russian.
[44] L. Portugal, M. Resende, G. Veiga, and J. Júdice, "A truncated primal-infeasible dual-feasible network interior point method," *Networks*, vol. 35, no. 2, pp. 91–108, 2000.
[45] T. Robertson, F. Wright, R. , and Dykstra, *Order Restricted Statistical Inference*. New York: Wiley, 1988.
[46] S. Rosset, , L. Saul, Y. Weiss, and L. Bottou, Eds., "Tracking curved regularized optimization solution paths," in *Advances in Neural Information Processing Systems 17*. Cambridge, MA: MIT Press, 2005.
[47] L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physics D*, vol. 60, pp. 259–268, 1992.
[48] Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd ed. Philadelphia, PA: SIAM, 2003.
[49] S. Samar, D. Gorinevsky, and S. Boyd, "Moving horizon filter for monotonic trends," in *Proc. 43rd IEEE Conf. Decision and Control*, 2004, vol. 3, pp. 3115–3120.
[50] M. Saunders, PDCO: Primal-Dual Interior Method for Convex Objectives 2002 [Online]. Available: http://www.stanford.edu/group/SOL/software/pdco.html
[51] N. Z. Shor, "Minimization methods for non-differentiable functions," *Springer Series in Computational Mathematics*, 1985, Springer.
[52] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Statist. Soc.*, ser. B, vol. 58, no. 1, pp. 267–288, 1996.
[53] J. Tropp, "Just relax: Convex programming methods for identifying sparse signals in noise," *IEEE Trans. Inf. Theory*, vol. 52, no. 3, pp. 1030–1051, Mar. 2006.
[54] Y. Tsaig and D. Donoho, "Extensions of compressed sensing," *Signal Process.*, vol. 86, no. 3, pp. 549–571, 2006.
[55] L. Vandenberghe and S. Boyd, "A primal-dual potential reduction method for problems involving matrix inequalities," *Mathemat. Programm.*, ser. B, vol. 69, pp. 205–236, 1995.
[56] M. Wainwright, "Sharp thresholds for high-dimensional and noisy recovery of sparsity," in *Proc. 44th Annu. Allerton Conf. Communication, Control, and Computing*, 2006.
[57] S. Wright, "Primal-dual interior-point methods," *Society for Industrial and Applied Mathematics*, 1997, SIAM: Philadelphia, PA.
[58] Y. Ye, *Interior Point Algorithms: Theory and Analysis*. New York: Wiley, 1997.
[59] W. Yin, S. Osher, J. Darbon, and D. Goldfarb, *Bregman Iterative Algorithms for Compessed Sensing and Related Problems* 2007 [Online]. Available: http://www.math.ucla.edu/applied/cam/index.shtml
[60] M. Yuan and L. Lin, "Model selection and estimation in regression with grouped variables," *J. Roy. Statist. Soc.*, ser. B, vol. 68, no. 1, pp. 49–67, 2006.
[61] P. Zhao and B. Yu, "On model selection consistency of lasso," *J. Mach. Learning Res.*, vol. 7, pp. 2541–2563, 2006.
[62] H. Zou, "The adaptive lasso and its oracle properties," *J. Amer. Statist. Assoc.*, vol. 101, no. 476, pp. 1418–1429, 2006.
[63] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Roy. Statist. Soc.*, ser. B, vol. 67, no. 2, pp. 301–320, 2005.

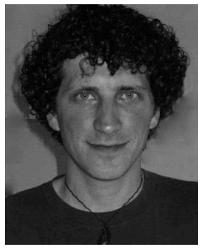
Seung-Jean Kim (M'02) received the Ph.D. degree in electrical engineering from Seoul National University, Seoul, Korea.

Since 2002, he has been with the Information Systems Laboratory, Department of Electrical Engineering, Stanford University, Stanford, CA, where he is currently a Consulting Assistant Professor. His current research interests include convex optimization with engineering applications, large-scale optimization, robust optimization, computational finance, machine learning, and statistics.



Kwangmoo Koh received the B.S. degree in electrical engineering from Seoul National University, Seoul, Korea, in 1999 and the M.S. degree in electrical engineering from in 2004 from Stanford University, Stanford, CA, where he is currently pursuing the Ph.D. degree in electrical engineering.

After a three-year stint as a member of technical staff with Humax Co., Ltd., Korea, and as a Research Assistant in Real-Time Operating System Laboratory, Seoul National University, he resumed his graduate studies at Stanford University. His research interests include convex optimization, machine learning, computer systems, and time-series analysis.



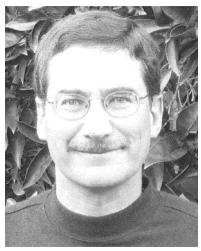
Michael Lustig received the B.Sc. degree from the Electrical Engineering Department, Technion-Israel Institute of Technology, Haifa, Israel, in 2001, and the M.Sc. degree in 2004 from the Electrical Engineering Department, Stanford University, Stanford, CA, where he is currently pursuing the Ph.D. degree, working on the application of compressed sensing to rapid MRI.

His research interests include medical imaging reconstruction techniques and MR pulse-sequence design.

Stephen Boyd (S'82–M'85–SM'97–F'99) received the A.B. degree in mathematics (summa cum laude) from Harvard University, Cambridge, MA, in 1980, and the Ph.D. degree in electrical engineering and computer science from the University of California at Berkeley in 1985.

He is the Samsung Professor of Engineering in the Information Systems Laboratory, Electrical Engineering Department, Stanford University, Stanford, CA. His current interests include convex programming applications in control, signal processing, and circuit design. He is the author of *Linear Controller Design: Limits of Performance* (with Craig Barratt, 1991), *Linear Matrix Inequalities in System and Control Theory* (with L. El Ghaoui, E. Feron, and V. Balakrishnan, 1994), and *Convex Optimization* (with Lieven Vandenberghe, 2004) .

Dr. Boyd received an Office of Naval Research (ONR) Young Investigator Award, a Presidential Young Investigator Award, the 1992 AACC Donald P. Eckman Award, the Perrin Award for Outstanding Undergraduate Teaching in the School of Engineering, an ASSU Graduate Teaching Award, and the 2003 AACC Ragazzini Education award. He is a Distinguished Lecturer of the IEEE Control Systems Society and holds an honorary Ph.D. degree from Royal Institute of Technology (KTH), Stockholm, Sweden.



Dimitry Gorinevsky (M'91–SM'98–F'06) received the Ph.D. degree from Moscow (Lomonosov) University, Moscow, Russia, and the M.Sc. degree from the Moscow Institute of Physics and Technology.

He is a Consulting Professor of Electrical Engineering at Stanford University, Stanford, CA, and an Independent Consultant to NASA. He worked for Honeywell for ten years. Prior to that, he held research, engineering, and academic positions in Moscow, Russia; Munich, Germany; and Toronto and Vancouver, Canada. His research interests are in decision and control systems applications across many industries. He has authored a book, more than 140 reviewed technical papers, and a dozen patents.

Dr. Gorinevsky is an Associate Editor of the IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY. He is a recipient of Control Systems Technology Award (2002) and an IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY Outstanding Paper Award (2004) of the IEEE Control Systems Society.