Direction of arrival estimation using robust complex Lasso

Esa Ollila
Department of Signal Processing and Acoustics, Aalto University
P.O.Box 13000, FI-00076 Aalto, Finland

***Abstract*—The Lasso (Least Absolute Shrinkage and Selection Operator) has been a popular technique for simultaneous linear regression estimation and variable selection. In this paper, we propose a new novel approach for robust Lasso that follows the spirit of M-estimation. We define M-Lasso estimates of regression and scale as solutions to generalized zero subgradient equations. Another unique feature of this paper is that we consider complex-valued measurements and regression parameters, which requires careful mathematical characterization of the problem. An explicit and efficient algorithm for computing the M-Lasso solution is proposed that has comparable computational complexity as state-of-the-art algorithm for computing the Lasso solution. Usefulness of the M-Lasso method is illustrated for direction-of-arrival (DoA) estimation with sensor arrays in a single snapshot case.**

***Index Terms*—Compressive sensing, beamforming, DoA estimation, Lasso, sparsity**

I. Introduction

We consider the complex-valued linear model $\mathbf{y} = \mathbf{\Phi}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, where $\mathbf{\Phi}$ is a known $n \times p$ complex-valued measurement matrix (or matrix of predictors), $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^\top$ is the unknown vector of complex-valued regression coefficients (or system parameters) and $\boldsymbol{\varepsilon} \in \mathbb{C}^n$ denotes the additive noise. For ease of exposition, we consider the centered linear model (i.e., we assume that the intercept is equal to zero). The primary interest is to estimate the unknown parameter $\boldsymbol{\beta}$ given $\mathbf{y} \in \mathbb{C}^n$ and $\mathbf{\Phi} \in \mathbb{C}^{n \times p}$. However, in many practical applications, the linear system is *underdetermined* ($p > n$) or $p \approx n$ and the least squares estimate (LSE) does not have a unique solution or is subject to a very high variance. Furthermore, for large number of predictors, we would like to identify the ones that exhibit the strongest effects. Hence we wish to find a *sparse solution* $\hat{\boldsymbol{\beta}}$, which sets weights for irrelevant predictors equal to 0. In these cases one needs to regularize the regression coefficients (i.e., to control how large they can grow). Another problem with the LSE arises when there are outliers or the noise follows a heavy-tailed non-Gaussian distribution. Then robust estimation [1] is upmost importance for reliable estimation of the unknown parameters.

The complex version of the popular Lasso [2] solves an ℓ_1-penalized LS regression problem,

$$\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{\Phi}\boldsymbol{\beta}\|_2^2 + 2\lambda\|\boldsymbol{\beta}\|_1 \tag{1}$$

where $\lambda > 0$ is the shrinkage (penalty) parameter. As $\lambda \in (0, \infty)$ varies, the solution $\hat{\boldsymbol{\beta}}_\lambda$ traces out a path in $\mathbb{C}^p$, with $\hat{\boldsymbol{\beta}}_{\lambda \to 0}$ then corresponding to the conventional LSE. We refer the reader to [3] for a comprehensive account on Lasso. The larger the value of λ the greater is the amount of shrinkage for the coefficients (some of which can be shrunk all the way to zero).

Robust Lasso is needed in case of heavy-tailed errors or severe outliers. A popular focus in the literature for obtaining robust Lasso estimates is to use a robust criterion in place of the least squares (LS) criterion. Most robust loss functions require a preliminary estimate of the scale of the error terms. An accurate estimate of scale is difficult to obtain since significant predictors are unknown ($\boldsymbol{\beta}$ is sparse and possibly $n < p$). Therefore a joint estimation of regression and scale becomes a necessity. In this paper, we propose a new approach for robust Lasso that follows the spirit of M-estimation. We define M-Lasso estimates of regression and scale as solutions to generalized zero subgradient equations which are based on general loss function. These equations are a sufficient and necessary condition of a solution to the Lasso problem (1) given that the loss function is the LS-loss. A unique feature of this paper is that we consider complex-valued measurements and regression parameters. This requires careful mathematical characterization of the problem and proper tools from complex function theory. A simple and efficient algorithm for computing the M-Lasso solution is also developed.

We illustrate how the proposed M-Lasso can be used for DoA estimation of source signals using sensor arrays when only a single snapshot is available. Indeed sparse regression approaches for DoA estimation has been an active research field; see [4], [5], [6], [7] and references therein. Our examples illustrate that M-Lasso based on Huber loss function has similar performance in DoA finding as Lasso (1) in complex Gaussian noise, but superior performance in heavy-tailed non-Gaussian noise or in face of outliers.

Let us offer a brief outline of the paper. Robust loss functions and their properties in complex-valued case are outlined in Section II. Also the notion of pseudo-residual vector is introduced which will be elemental in our developments. In Section III, we recall the zero subgradient estimating equations for Lasso solution and then define M-Lasso estimates of regression and scale as solutions to generalized subgradient equations. A highly efficient algorithm for computing the M-Lasso estimates is also proposed. Finally, we consider the direction finding application with sensor arrays in Section IV. Section V concludes.

Notations. The vector space $\mathbb{C}^n$ is equipped with the usual Hermitian inner product, $\langle \mathbf{a}, \mathbf{b} \rangle = \mathbf{a}^{\mathsf{H}}\mathbf{b}$, where $(\cdot)^{\mathsf{H}} = [(\cdot)^*]^\top$ denotes the Hermitian (complex conjugate) transpose. This induces the conventional (Hermitian) ℓ_2-norm $\|\mathbf{a}\|_2 = \sqrt{\mathbf{a}^{\mathsf{H}}\mathbf{a}}$. The ℓ_1-norm is the defined as $\|\mathbf{a}\|_1 = \sum_{i=1}^n |a_i|$, where $|a| = \sqrt{a^* a} = \sqrt{a_R^2 + a_I^2}$ denotes the modulus of a complex number $a = a_R + \jmath a_I$. For a matrix $\mathbf{A} \in \mathbb{C}^{n\times p}$, we denote by $\mathbf{a}_i \in \mathbb{C}^n$ its i^{th} column vector and $\mathbf{a}_{(i)} \in \mathbb{C}^p$ denotes the Hermitian transpose of its i^{th} row vector. Hence, this means that we may write the measurement matrix $\mathbf{\Phi} \in \mathbb{C}^{n\times p}$ as $\mathbf{\Phi} = \begin{pmatrix} \boldsymbol{\phi}_1 & \cdots & \boldsymbol{\phi}_p \end{pmatrix} = \begin{pmatrix} \boldsymbol{\phi}_{(1)} & \cdots & \boldsymbol{\phi}_{(n)} \end{pmatrix}^{\mathsf{H}}$.

II. Robust loss functions and pseudo-residuals

Suppose that the error terms ε_i are i.i.d. continuous random variables from a circular distribution [8] with p.d.f. $f(e) = (1/\sigma) f_0(e/\sigma)$, where $f_0(e)$ denotes the standard form of the density and $\sigma > 0$ is the scale parameter. Robust loss functions commonly require knowledge of σ in order to properly downweight outlying observations. Hence the unknown scale σ needs to be estimated jointly with the regression coefficient as a preliminary robust scale estimate is generally not available.

We adopt the definition of *loss function* to complex-valued case from [9]. Namely, we call $\rho : \mathbb{C} \to \mathbb{R}_0^+$ a loss function if it is circularly symmetric, $\mathbb{R}$-differentiable convex function which satisfies $\rho(0) = 0$. Due to circularity assumption (implying that $\rho(e^{\jmath\theta}x) = \rho(x) \forall\, \theta \in \mathbb{R}$) it follows that $\rho(x) = \rho_0(|x|)$ for some $\rho_0 : \mathbb{R}_0^+ \to \mathbb{R}_0^+$. This illustrates that ρ is not $\mathbb{C}$-differentiable (i.e., holomorphic function) since only functions that are *both* holomorphic *and* real-valued are constants. The complex derivative [10] of ρ w.r.t. $x^* = (x_R + \jmath x_I)^*$ is

$$\psi(x) = \frac{\partial}{\partial x^*}\rho(x) = \frac{1}{2}\left(\frac{\partial \rho}{\partial x_R} + \jmath \frac{\partial \rho}{\partial x_I}\right) = \frac{1}{2}\rho_0'(|x|)\mathrm{sign}(x),$$

where

$$\mathrm{sign}(e) = \begin{cases} e/|e|, & \text{for } e \neq 0 \\ 0, & \text{for } e = 0 \end{cases}$$

is the complex signum function and ρ_0' denotes the real derivative of the real-valued function ρ_0. Function $\psi : \mathbb{C} \to \mathbb{C}$ will be referred in the sequel as *score function*.

For obtaining robust estimates, the utilized loss function $\rho(e)$ should assign smaller weights for large errors e than the *LS (or ℓ_2-)loss* $\rho(e) = |e|^2$. One most commonly used robust loss function is due to Huber [11]. In the complex-valued case, *Huber's loss function* can be defined as follows [9]:

$$\rho_{H,c}(e) = \begin{cases} |e|^2, & \text{for } |e| \leq c \\ 2c|e| - c^2, & \text{for } |e| > c, \end{cases} \tag{2}$$

where c is a user-defined *threshold* that influences the degree of robustness and efficiency of the method. Hence similar to the real-valued case, Huber's loss function is a hybrid of ℓ_2 and ℓ_1 loss functions $\rho(e) = |e|^2$ and $\rho(e) = |e|$, respectively, using ℓ_2-loss for relatively small errors and ℓ_1-loss for relatively large errors. Moreover, it is convex and verifies the conditions imposed on the loss function ($\mathbb{R}$-differentiability and circular symmetry). Huber's score function becomes

$$\psi_{H,c}(e) = \begin{cases} e, & \text{for } |e| \leq c \\ c\,\mathrm{sign}(e), & \text{for } |e| > c \end{cases}.$$

Note that Huber's ψ is a winsorizing (clipping) function: the smaller the c, the more clipping is actioned on the residuals.

Now recall that in robust regression, the loss function acts on standardized residual vector $\mathbf{r}/\sigma$, where $\mathbf{r} \equiv \mathbf{r}(\boldsymbol{\beta}) = \mathbf{y} - \mathbf{\Phi}\boldsymbol{\beta}$ denotes the residual vector for some candidate $\boldsymbol{\beta} \in \mathbb{C}^p$ of regression coefficient vector and σ is the scale. The loss function then defines a *pseudo-residual*, defined as

$$\mathbf{r}_\psi \equiv \mathbf{r}_\psi(\boldsymbol{\beta}, \sigma) = \psi\left(\frac{\mathbf{y} - \mathbf{\Phi}\boldsymbol{\beta}}{\sigma}\right)\sigma \tag{3}$$

where ψ-function acts coordinate-wise to vector $\mathbf{r}/\sigma$, so $[\psi(\mathbf{r}/\sigma)]_i = \psi(r_i/\sigma)$. Some remarks of definition (3) are in order. First, note that if $\rho(\cdot)$ is the conventional LS-loss, $\rho(e) = |e|^2$, then $\psi(e) = e$, and $\mathbf{r}_\psi$ is equal with the conventional residual vector, so $\mathbf{r}_\psi = \mathbf{y} - \mathbf{\Phi}\boldsymbol{\beta} = \mathbf{r}$. Second, for Huber's loss function, pseudo-residual vector $\mathbf{r}_\psi$ has i^{th} element equal to r_i if $|r_i| < c\sigma$ and equal to $(c\sigma)\mathrm{sign}(r_i)$ otherwise. In other words, residuals that are farther apart from zero than c times the scale σ are trimmed (downweighted). This is the underlying principle for robustness of Huber's loss function. Third, note that the multiplier σ in (3) is essential in bringing the residuals back to the original scale of the data.

Since σ is unknown in practise, robust loss function $\rho(e)$, such as Huber's loss above, require a preliminary robust scale estimate $\hat{\sigma}$ in order to determine if e should be downweighted or not. In sparse regression problems, obtaining such an estimate is more difficult task than in the conventional regression problem since now the significant predictors (columns of $\mathbf{\Phi}$) are not known ($\boldsymbol{\beta}$ is sparse and possibly $n < p$). Suppose that a preliminary robust regression estimate $\hat{\boldsymbol{\beta}}_{init}$ can be computed ($n > p$ case) and then used to compute a robust scale statistic $\hat{\sigma}$ (e.g., median absolute deviation) based on $\hat{\mathbf{r}} = \mathbf{y} - \mathbf{\Phi}\hat{\boldsymbol{\beta}}_{init}$. Such a scale estimate is biased and can significantly underestimate the true σ due to overfitting when the number of predictors p is large. This then implies too severe downweighting. So robust loss function and underestimated $\hat{\sigma}$ results in pseudo-residuals which can severely downweight 'good' residuals (not just outliers). Similarly, all residuals can be left intact if $\hat{\sigma}$ is an overestimate (too large).

III. Robust complex M-Lasso

The earlier approaches for robust Lasso are based on an idea of adding an ℓ_1-penalty to a robust criterion function since the LS criterion function, $J_{\ell_2}(\boldsymbol{\beta}) = \|\mathbf{y} - \mathbf{\Phi}\boldsymbol{\beta}\|_2^2$, is sensitive to outliers. For example, [12] utilize the least absolute deviation (LAD) criterion, $J_{\ell_1}(\boldsymbol{\beta}) = \|\mathbf{y} - \mathbf{\Phi}\boldsymbol{\beta}\|_1$, LTS-Lasso of [13] is based on the least trimmed squares (LTS) criterion, whereas [14] utilized Huber's criterion function $Q(\boldsymbol{\beta}, \sigma)$ in (9).

Our approach is different from these earlier approaches. Namely, we define Lasso estimator as a solution to generalized zero subgradient equations that is based on general loss function $\rho(e)$. Our approach follows the spirit of M-estimation [11], [1] where the principal idea is to define an estimator as a solution to generalized maximum likelihood (ML-)estimating equations.

We start by recalling the zero subgradient equation for complex-valued Lasso problem (1). Note that the utilized LS criterion function $J_{\ell_2}(\boldsymbol{\beta})$ in Lasso problem (1) is convex (in fact strictly convex if $n > p$) and $\mathbb{R}$-differentiable but the ℓ_1-penalty function $\|\boldsymbol{\beta}\|_1$ is not $\mathbb{R}$-differentiable at a point where at least one coordinate β_j is zero. However, we can resort to generalization of notion of gradient applicable for convex functions, called the *subdifferential* [15]. For a complex function $f : \mathbb{C}^p \to \mathbb{R}$ we can define subdifferential at a point $\boldsymbol{\beta}$ as

$$\partial f(\boldsymbol{\beta}) = \{\mathbf{z} \in \mathbb{C}^p : f(\boldsymbol{\beta}') \geq f(\boldsymbol{\beta}) + 2\mathrm{Re}(\langle \mathbf{z}, \boldsymbol{\beta}' - \boldsymbol{\beta}\rangle) \\ \text{for all } \boldsymbol{\beta}' \in \mathbb{C}^p\}.$$

Any element $\mathbf{z} \in \partial f(\boldsymbol{\beta})$ is then called a *subgradient* of f at $\boldsymbol{\beta}$. The subdifferential of the modulus $|\beta_j|$ is

$$\partial|\beta_j| = \begin{cases} \frac{1}{2}\mathrm{sign}(\beta_j), & \text{for } \beta_j \neq 0 \\ \frac{1}{2}s & \text{for } \beta_j = 0 \end{cases}$$

where s is some complex number verifying $|s| \leq 1$. Thus subdifferential of $|\beta_j|$ is the usual complex derivative when $\beta_j \neq 0$, i.e., $\partial|\beta_j| = \frac{\partial}{\partial\beta_j^*}|\beta_j|$ for $\beta_j \neq 0$. Then a necessary and sufficient condition for a solution to the Lasso problem (1) is that $\partial(J_{\ell_2}(\boldsymbol{\beta}) + 2\lambda\|\boldsymbol{\beta}\|_1) \in \mathbf{0}$ which gives zero subgradient equation

$$-\boldsymbol{\phi}_i^{\mathsf{H}}\big(\mathbf{y} - \boldsymbol{\Phi}\hat{\boldsymbol{\beta}}\big) + \lambda\hat{s}_j = 0 \quad \text{for } j = 1, \ldots, p \tag{4}$$

where $\hat{s}_j$ is 2 times an element of the subdifferential of $|\beta_j|$ evaluated at $\hat{\beta}_j$, i.e., equal to $\mathrm{sign}(\hat{\beta}_j)$ if $\hat{\beta}_j \neq 0$ and some complex number lying inside the unit complex circle otherwise. Given the Lasso solution $\hat{\boldsymbol{\beta}}$, the natural scale estimate $\hat{\sigma}^2$ is then

$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}\big|y_i - \boldsymbol{\phi}_{(i)}^{\mathsf{H}}\hat{\boldsymbol{\beta}}\big|^2 = \frac{1}{n}\|\hat{\mathbf{r}}\|_2^2 \tag{5}$$

where $\hat{\mathbf{r}} = \mathbf{y} - \boldsymbol{\Phi}\hat{\boldsymbol{\beta}}$ denote the residual vector at the solution.

Given the considerations in Section II, it appears wise to estimate the unknown parameters $\boldsymbol{\beta} \in \mathbb{C}^p$ and $\sigma > 0$ jointly. Thus we seek for a pair $(\hat{\boldsymbol{\beta}}, \hat{\sigma})$ which verify the generalized (zero subgradient) estimating equations, which we refer to as *Lasso M-estimating equations*, of the form

$$-\boldsymbol{\phi}_i^{\mathsf{H}}\mathbf{r}_\psi(\hat{\boldsymbol{\beta}}, \hat{\sigma}) + \lambda\hat{s}_j = 0 \quad \text{for } j = 1, \ldots, p \tag{6}$$

$$\alpha n - \sum_{i=1}^{n}\chi\left(\frac{|y_i - \boldsymbol{\phi}_{(i)}^{\mathsf{H}}\hat{\boldsymbol{\beta}}|}{\hat{\sigma}}\right) = 0 \tag{7}$$

where $\alpha > 0$ is a fixed scaling factor (described later) and function $\chi : \mathbb{R}_0^+ \to \mathbb{R}_0^+$ is defined as

$$\chi(t) = \rho_0'(t)t - \rho_0(t). \tag{8}$$

Recall that $\rho(x) = \rho_0(|x|)$. To simplify notation we write the pseudo-residual vector $\mathbf{r}_\psi(\hat{\boldsymbol{\beta}}, \hat{\sigma})$ in (6) as $\hat{\mathbf{r}}_\psi$.

Some remarks of this definition are in order. First, consider the conventional choice, i.e., the LS-loss $\rho(e) = |e|^2$. In this case, $\hat{\mathbf{r}}_\psi = \hat{\mathbf{r}}$, so (6) reduces to (4). Furthermore, since $\rho_0(t) = t^2$ and $\rho_0'(t) = 2t$, the χ-function in (8) is $\chi(t) = t^2$, and (7) reduces to (5). In other words, for LS-loss, the M-Lasso solution $(\hat{\boldsymbol{\beta}}, \hat{\sigma})$ to (6)-(7) is the conventional Lasso estimate (so $\hat{\boldsymbol{\beta}}$ is a solution to (1)) and $\hat{\sigma}$ equals the standard scale statistic in (5). Second, if $\lambda = 0$ (so no penalization and $n > p$), then the solution to (6) and (7) is the unique solution to the convex optimization problem

$$\arg\min_{\boldsymbol{\beta},\sigma}\left\{Q(\boldsymbol{\beta}, \sigma) = \alpha n\sigma + \sum_{i=1}^{n}\rho\left(\frac{y_i - \boldsymbol{\phi}_{(i)}^{\mathsf{H}}\boldsymbol{\beta}}{\sigma}\right)\sigma\right\}. \tag{9}$$

Important feature of the objective function $Q(\boldsymbol{\beta}, \sigma)$ above is that it is jointly convex in $(\boldsymbol{\beta}, \sigma)$ given that ρ is convex. In other words, for $\lambda = 0$ (and $n > p$), equations (6) and (7) are necessary and sufficient condition for a solution to problem (9). This objective function was originally studied by Huber [1] in the real-valued case. Lasso penalized Huber's criterion was considered by Owen [14] and ℓ_0-penalization in real-valued and complex-valued case in [16], [9], respectively.

Next we note that (6) can be written after recalling the definition (3) more compactly as $\langle \boldsymbol{\phi}_j, \hat{\mathbf{r}}_\psi\rangle = \lambda\hat{s}$ for $j = 1, \ldots, p$. This mean that (after taking modulus of both sides) the following holds

$$|\langle \boldsymbol{\phi}_j, \hat{\mathbf{r}}_\psi\rangle| = \lambda, \quad \text{if } \hat{\beta}_j \neq 0 \tag{10}$$

$$|\langle \boldsymbol{\phi}_j, \hat{\mathbf{r}}_\psi\rangle| \leq \lambda, \quad \text{if } \hat{\beta}_j = 0 \tag{11}$$

i.e., whenever a component, say $\hat{\beta}_j$, of $\hat{\boldsymbol{\beta}}$ becomes non-zero, the corresponding absolute correlation between the pseudo-residual $\hat{\mathbf{r}}_\psi$ and column $\boldsymbol{\phi}_j$ of $\boldsymbol{\Phi}$, $|\langle \boldsymbol{\phi}_j, \hat{\mathbf{r}}_\psi\rangle|$, meets the boundary λ in magnitude, where $\lambda > 0$ is the penalty parameter. This is well-known property of Lasso; see e.g., [3] or [7] in the complex-valued case. This property is then fulfilled by M-Lasso estimates by definition. In the real-valued case, [14] considered minimization of penalized Huber's criterion $Q_\lambda(\boldsymbol{\beta}, \sigma) = Q(\boldsymbol{\beta}, \sigma) + \lambda\|\boldsymbol{\beta}\|_1$. The solution of $\min_{\boldsymbol{\beta},\sigma} Q_\lambda(\boldsymbol{\beta}, \sigma)$, however, is different from solutions to (6)-(7). This can be verified by noting that the zero subgradient equation $\partial_{\boldsymbol{\beta}}Q_\lambda(\boldsymbol{\beta}, \sigma) = \mathbf{0}$ is different from (6). This also means that solution for penalized Huber's criterion based on LS-loss function $\rho(e) = |e|^2$ is not the Lasso solution (1). This is somewhat counterintuitive. This equivalence with Lasso and M-Lasso for LS-loss, however, holds.

The scaling factor α in (7) is chosen so that the obtained scale estimate $\hat{\sigma}$ is Fisher-consistent for the unknown scale σ when $\{\varepsilon_i\}_{i=1}^n \overset{iid}{\sim} \mathbb{C}\mathcal{N}(0, \sigma^2)$. Due to (7), it is chosen so that $\alpha = \mathbb{E}[\chi(e)]$, when $e \sim \mathbb{C}\mathcal{N}(0, 1)$, holds. For many loss functions, α can be computed in closed-form. For example, for Huber's function (2) the χ-function in (8) becomes

$$\chi_{H,c}(|e|) = |\psi_{H,c}(e)|^2 = \begin{cases} |e|^2, & \text{for } |e| \leq c \\ c^2, & \text{for } |e| > c \end{cases}. \tag{12}$$

In this case the estimating equation (7) can be written as

$$\sum_{i=1}\left|\psi_{H,c}\left(\frac{y_i - \boldsymbol{\phi}_{(i)}^{\mathsf{H}}\hat{\boldsymbol{\beta}}}{\hat{\sigma}}\right)\hat{\sigma}\right|^2 = \hat{\sigma}^2 n\alpha \Leftrightarrow \hat{\sigma}^2 = \frac{1}{n\alpha}\|\hat{\mathbf{r}}_\psi\|^2$$

where $\hat{\mathbf{r}}_\psi = \mathbf{r}_\psi(\hat{\boldsymbol{\beta}}, \hat{\sigma})$. The consistency factor $\alpha = \alpha(c)$ can be computed in closed-form as

$$\alpha = c^2(1 - F_{\chi^2_2}(2c^2)) + F_{\chi^2_4}(2c^2). \qquad (13)$$

Note that α depends on the threshold c. We will choose threshold c as $c^2 = (1/2)F^{-1}_{\chi^2_2}(q)$ for $q \in (0,1)$. See [9].

Next we propose an explicit and efficient algorithm for computing the M-Lasso solution. The algorithm follows the idea of state-of-the-art algorithm, the cyclic coordinate descent (CCD) [17], for computing the Lasso solution. Our algorithm is a generalization of it in two aspects. First, it adapts it to the complex-valued case and second, it generalizes the algorithm to the robust M-estimation scenario. First recall that CCD algorithm repeatedly cycles through the predictors updating one parameter (coordinate) β_j at a time ($j = 1, \ldots, p$) while keeping others fixed at their current iterate values. At jth step, the update for $\hat{\beta}_j$ is obtained by soft-thresholding a conventional coordinate descent update $\hat{\beta}_j + \langle \boldsymbol{\phi}_j, \hat{\mathbf{r}}\rangle$, where $\hat{\mathbf{r}}$ denotes the residual vector $\hat{\mathbf{r}} = \mathbf{r}(\hat{\boldsymbol{\beta}})$ at current estimate $\hat{\boldsymbol{\beta}}$. For M-Lasso, similar updates are performed but $\hat{\mathbf{r}}$ replaced by pseudo-residual vector $\hat{\mathbf{r}}_\psi$ and the update for scale is obtained prior to cycling through the coefficients. The M-Lasso algorithm proceeds as follows:

```
1) Update the scale $\hat{\sigma}^2 \leftarrow \frac{\hat{\sigma}^2}{\alpha n}\sum_{i=1}^{n}\chi\left(\frac{y_i - \boldsymbol{\phi}_{(i)}^{\mathsf{H}}\hat{\boldsymbol{\beta}}}{\hat{\sigma}}\right)$
2) For $j = 1, \ldots, p$ do
    a) Update the pseudoresidual: $\hat{\mathbf{r}}_\psi \leftarrow \psi\left(\frac{\mathbf{y} - \boldsymbol{\Phi}\hat{\boldsymbol{\beta}}}{\hat{\sigma}}\right)\hat{\sigma}$
    b) Update the coefficient: $\hat{\beta}_j \leftarrow S_\lambda\big(\hat{\beta}_j + \langle \boldsymbol{\phi}_j, \hat{\mathbf{r}}_\psi\rangle\big)$
3) Repeat Steps 1 and 2 until convergence
```

Above $S_\lambda(x) = \operatorname{sign}(x)(|x| - \lambda)_+$, $x \in \mathbb{C}$, is the complex soft-thresholding operator and $(t)_+$ denotes the positive part of $t \in \mathbb{R}$: $(t)_+ = t$ if $t > 0$ and 0 otherwise. The M-Lasso algorithm has comparable computational complexity as state-of-the art algorithm for computing the Lasso solution (1).

IV. Single snapshot DoA estimation

We consider uniform linear array (ULA) consisting of n sensors with half a wavelength inter-element spacing that receives k narrowband incoherent farfield plane-wave sources from a point source ($n > k$). At discrete time t, the array output (called *snapshot*) $\mathbf{y} \in \mathbb{C}^n$ is a weighted linear combination of the signal waveforms $\mathbf{s} = (s_1, \ldots, s_k)^\top$ corrupted by additive noise $\mathbf{e} \in \mathbb{C}^n$, $\mathbf{y}(t) = \mathbf{A}(\boldsymbol{\theta})\mathbf{s} + \mathbf{e}$, where $\mathbf{A} = \mathbf{A}(\boldsymbol{\theta})$ is the $n \times k$ *steering matrix* parametrized by the vector $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_k)^\top$ of (distinct) unknown direction-of-arrivals (DoA's) of the sources. We assume that only a single snapshot is available. Each column vector $\mathbf{a}(\theta_i)$, called the *steering vector*, represents a point in known array manifold, $\mathbf{a}(\theta) = \frac{1}{\sqrt{p}}(1, e^{-\jmath\pi\sin(\theta)}, \cdots, e^{-\jmath\pi(n-1)\sin(\theta)})^\top$. The objective of sensor array source localization is to find the DoA's of the sources, i.e., to identify the steering matrix $\mathbf{A}(\boldsymbol{\theta})$ parametrized by $\boldsymbol{\theta}$.

As in [4], we cast the source localization problem as a sparse regression problem. We construct an angular grid (look directions of interest) of size $p \gg k$, $[\theta] = \{\theta_{(i)} \in [-\pi/2, \pi/2) \ : \ \theta_{(1)} > \cdots > \theta_{(p)}\}$. If $[\theta]$ contains the true DoA's θ_i, $i = 1, \ldots, k$, then the snapshot follows sparse linear regression model, $\mathbf{y} = \boldsymbol{\Phi}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, where the measurement matrix $\boldsymbol{\Phi} \in \mathbb{C}^{n\times p}$ has as its columns the steering vectors at considered look directions, i.e., $\boldsymbol{\phi}_i = \mathbf{a}(\theta_{(i)})$. Thus identifying the true DoA's is equivalent to identifying the non-zero elements of β_j. Thus (M-)Lasso estimation becomes necessary since often $p > n$ and the LSE does not provide sparse solutions. Note also that even if $[\theta]$ does not contain the true DoA's but has reasonably fine grid, one can identify good estimates of true DoA's as locations in the angular grid corresponding to k largest coefficients of M-Lasso solution (given $\mathbf{y}$ and $\boldsymbol{\Phi}$) $\hat{\boldsymbol{\beta}}_\lambda$, where λ is such that the solution has ≥ 3 nonzero coefficients. If the number of sources k is known, then more obvious approach is to obtain the M-Lasso estimate $\hat{\boldsymbol{\beta}}_\lambda$ for a penalty parameter λ that results in k-nonzero elements. Let us denote the largest λ value that produces the desired k non-zero coefficients by λ^*. The locations of the nonzero coefficients of $\hat{\boldsymbol{\beta}}_{\lambda^*}$ in the angular grid $[\theta]$ then give natural M-Lasso DoA estimates.

The simulation set-up is described next. The ULA receives $k = 3$ sources at DoA's $\theta_1 = -5$, $\theta_2 = 0$ and $\theta_3 = 20$ degrees and the noise $\boldsymbol{\varepsilon}$ has i.i.d. elements from $\mathbb{C}\mathcal{N}(0, \sigma^2)$ distribution. The amplitudes of the sources are $|s_1| = 1$, $|s_2| = 0.6$ and $|s_3| = 0.2$ and the noise variance σ^2 is chosen such that the SNR $= 10\log_{10}(\bar{s}^2/\sigma^2) = 15$dB, where $\bar{s}^2 = \frac{1}{3}(|s_1|^2 + |s_2|^2 + |s_3|^2) = 0.4667$ denotes the average source power. The phase of each source $s_i \in \mathbb{C}$ is randomly generated from $Unif(0, 2\pi)$ distribution. We consider angular grid $[\theta] = (-90, -85, \ldots, 80, 85)$ with 5 degree spacing. Thus the simulation set-up closely follows that of [7]. We compare results of regular Lasso ($=$ M-Lasso using LS-loss function) to the results of robust M-Lasso using Huber's loss function $\rho_{H,c}(\cdot)$ with threshold $c = 1.3774$ corresponding to $q = 0.85$. To compare robustness of the methods, we compute the estimates also for corrupted data in which magnitude of one measurement, y_1, is scaled by a factor of 100. To depict the M-Lasso solution paths, we compute the solution on a grid of 200 values in $(0, \lambda_{\max})$ with equal spacings in the logarithmic scale, where λ_{max} denotes the smallest penalty value that shrinks all the coefficients of M-Lasso estimates to zero.

Left hand side column of Figure 1 shows the results for the original data and the right hand side column for the corrupted data. For each method, the upper row depicts the M-Lasso coefficient paths, i.e., the graphs of $|\hat{\beta}_{\lambda,j}|$ for $j = 1, \ldots, p$ versus normalized $\|\hat{\boldsymbol{\beta}}_\lambda\|$. The dotted vertical line identifies the solution $\hat{\boldsymbol{\beta}}_{\lambda^*}$ which is then used in the lower row plots. As can be seen the coefficient paths of Lasso and Huber's M-Lasso for original data are closely similar. For corrupted data, however, the Lasso coefficient paths completely change whereas the solution path for Huber's M-Lasso remains practically unaffected by the large outlier. For original data both methods yield a solution $\hat{\boldsymbol{\beta}}_{\lambda^*}$ that identify the true DoA's. However, for corrupted data, Lasso yields estimates 20^o, 45^o, and 55^o degrees. Thus curiously, only the source s_3 (from DoA $\theta_3 = 20^o$) with lowest power (SNR) is correctly identified whereas the two higher power sources ($\theta_1 = -5^o$ and $\theta_2 = 0^o$) are not. Huber's M-Lasso, however, correctly identifies the DoA's of the true sources as well as the order of the magnitudes. For each method, the lower row in Figure 1 plots $|\langle \mathbf{a}(\theta_{(i)}), \hat{\mathbf{r}}_\psi \rangle|$ versus $\theta_{(i)}$ on the angular grid $[\theta]$. The horizontal line indicates the value λ^* (giving 3 nonzero coefficients) used and the dotted vertical lines identify the true DoA's of the sources. These plots also illustrate that equations (10)-(11) hold, so the M-Lasso algorithm has correctly found the solutions to (6) and (7). To conclude, for original data, both methods produce similar plots and the same correct DoA estimates, but for corrupted data, only the robust Huber's M-Lasso provides reliable estimates.

V. Conclusions

The robust M-Lasso estimates of regression and scale are defined as solutions to generalized zero subgradient equations in the spirit of M-estimation. An explicit and efficient algorithm for computing the solution was proposed. The usefulness of complex M-Lasso in DoA estimation of sources with sensor arrays was illustrated using a simulated data set. Due to fast algorithm, we recommend using M-Lasso in practical big data applications due to its robustness properties.

References

[1] P. J. Huber, *Robust Statistics*. New York: Wiley, 1981.

[2] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Royal Stat. Soc., Ser. B*, vol. 58, pp. 267–288, 1996.

[3] T. Hastie, R. Tibshirani, and M. Wainwright, *Statistical Learning with Sparsity: The Lasso and Generalizations*. CRC Press, 2015.

[4] D. Malioutov, M. Cetin, and A. S. Willsky, "A sparse signal reconstruction perspective for source localization with sensor arrays," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 3010–3022, 2005.

[5] S. Fortunati, R. Grasso, F. Gini, M. S. Greco, and K. LePage, "Single-snapshot DOA estimation by using compressed sensing," *EURASIP J. Adv. Signal Process.*, vol. 2014, no. 1, pp. 1–17, 2014.

[6] A. Xenaki, P. Gerstoft, and K. Mosegaard, "Compressive beamforming," *J. Acoust. Soc. Am.*, vol. 136, no. 1, pp. 260–271, 2014.

[7] P. Gerstoft, A. Xenaki, and C. Mecklenbrauker, "Multiple and single snapshot compressive beamforming," *J. Acoust. Soc. Am.*, vol. 138, no. 4, pp. 2003–2014, 2015.

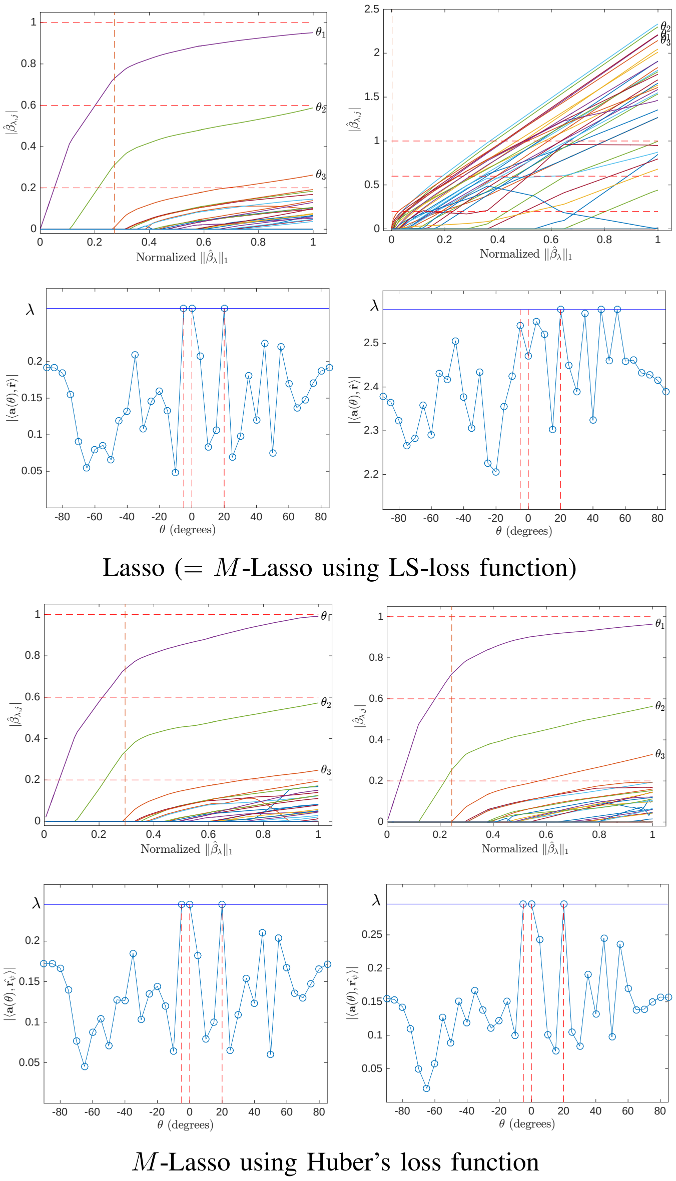
[8] E. Ollila, J. Eriksson, and V. Koivunen, "Complex elliptically symmetric random variables – generation, characterization, and circularity tests," *IEEE Trans. Signal Process.*, vol. 59, no. 1, pp. 58–69, 2011.

[9] E. Ollila, "Multichannel sparse recovery of complex-valued signals using Huber's criterion," in *Proc. Compressed Sensing Theory and its Applications to Radar, Sonar and Remote Sensing (CoSeRa'15)*, Pisa, Italy, Jun. 16 – 19, 2015, pp. 32–36.

[10] J. Eriksson, E. Ollila, and V. Koivunen, "Essential statistics and tools for complex random variables," *IEEE Trans. Signal Process.*, vol. 58, no. 10, pp. 5400–5408, 2010.

[11] P. Huber, "Robust estimation of a location parameter," *The Annals of Mathematical Statistics*, vol. 35, no. 1, pp. 73–101, 1964.

[12] H. Wang, G. Li, and G. Jiang, "Robust regression shrinkage and consistent variable selection through the LAD-Lasso," *J. Bus. Econ. Stat.*, vol. 25, pp. 347–355, 2007.



Lasso (= M-Lasso using LS-loss function)

M-Lasso using Huber's loss function

Fig. 1. Results for M-Lasso based on LS-loss and Huber's loss function. Left columns show results for the original data and the right column for the corrupted data. For both methods, first row shows the coefficient paths. The dotted vertical line identifies the solution $\hat{\boldsymbol{\beta}}_\lambda$ which is used in the plots below and the dotted horizontal lines indicate the magnitudes $|\beta_j|$ of the true sources. The second row depicts $|\langle \mathbf{a}(\theta_{(i)}), \hat{\mathbf{r}}_\psi \rangle|$ on the angular grid $[\theta]$. The horizontal line indicates the value of λ used and the dotted vertical lines identify the true DoA's of the sources.

[13] A. Alfons, C. Croux, and S. Gelper, "Sparse least trimmed squares regression for analyzing high-dimensional large data sets," *Ann. Appl. Stat.*, vol. 7, no. 1, pp. 226–248, 2013.

[14] A. B. Owen, "A robust hybrid of lasso and ridge regression," *Contemporary Mathematics*, vol. 443, pp. 59–72, 2007.

[15] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[16] E. Ollila, H.-J. Kim, and V. Koivunen, "Robust iterative hard thresholding for compressed sensing," in *Proc. IEEE Int'l Symp. Communications, Control, and Signal Processing (ISCCSP'14)*, Athens, Greece, May 21 – 23, 2014, pp. 226–229.

[17] J. Friedman, T. Hastie, H. Höfling, and R. Tibshirani, "Pathwise coordinate optimization," *Ann. Appl. Stat.*, vol. 1, no. 2, pp. 302–332, 2007.