Robust Linear Estimation using M-estimation and Weighted L1 Regularization: Model Selection and Recursive Implementation

Z. G. Zhang [1, 2], S. C. Chan [2], Y. Zhou [2], and Y. Hu [1]

[1] Department of Orthopaedics and Traumatology, the University of Hong Kong, Pokfulam Road, Hong Kong

[2] Department of Electrical and Electronic Engineering, the University of Hong Kong, Pokfulam Road, Hong Kong

{zgzhang; scchan; yizhou}@eee.hku.hk; yhud@hku.hk

***Abstract*—This paper studies an M-estimation-based method for linear estimation with weighted L1 regularization and its recursive implementation. Motivated by the sensitivity of conventional least-squares-based L1-regularized linear estimation (Lasso) in impulsive noise environment, an M-estimator-based Lasso (M-Lasso) method is introduced to restrain the outliers and an iterative re-weighted least-squares (IRLS) algorithm is proposed to solve this M-estimation problem. Moreover, instead of using the matrix inversion formula, QR decomposition (QRD) is employed in the M-Lasso for recursive implementation with a lower arithmetic complexity. Simulation results show that the M-estimation-based Lasso performs considerably better than the traditional LS-based Lasso in suppressing the impulsive noise, and its recursive QRD algorithm has a good performance in online processing.**

I. INTRODUCTION

The linear absolute shrinkage and selection operator (Lasso) method was first proposed in [1] by Tibshirani for estimation in linear regression model. Since then, the statistical properties of Lasso have been thoroughly studied in statistics community [2]-[4], and a number of techniques have been developed to solve and improve the Lasso method [5]-[8]. The Lasso algorithm employs an L1 regularization on the regression coefficients and it tends to produce extremely sparse solutions, which makes it a powerful technique to address the variable selection problem. Therefore, the Lasso method is attracting more and more interests in a wide variety of research areas, and its practical implications have been reported in model selection [9]-[11], basis pursuit denoising [12], sparse signal recovery and compressed sensing [13]-[14], etc. The traditional Lasso method estimates the linear regression model in a least-squares (LS) sense, i.e., the estimate is obtained by minimizing the sum of squared residual errors. Since the LS estimation implicitly assumes that the additive noise is Gaussian, its performance will be considerably degraded in impulsive noise environment. To address this problem, a robust Lasso is proposed in this study and it employs an M-estimation function instead of the conventional quadratic LS function so that the impulsive noise or outliers can be effectively suppressed.

The conventional Lasso and the M-estimation-based Lasso (M-Lasso) were intended for batch processing of a block of observations. In this paper, a new time-recursive QR-decomposition (QRD)-based algorithm is proposed for the Lasso to reduce the arithmetic complexity for online processing. Furthermore, to combat the adverse effect of impulsive noise to the recursive QRD-based Lasso algorithm, M-estimation is employed in the recursive algorithm and it yields the proposal QRD-M-Lasso algorithm. Finally, the proposed M-Lasso and the QRD-M-Lasso are applied to a sparse signal recovery problem to evaluate their performances in contaminated-Gaussian noise environments.

The paper is organized as follows. In Section II, the basic principle of the conventional Lasso method is revisited. Section III introduces the M-estimation-based Lasso for suppressing impulsive noise or outliers. The QRD-based recursive Lasso is developed in Section IV. Experimental results on sparse signal recovery and comparisons to other conventional methods are presented in Section V. Finally, conclusions are drawn in Section VI.

II. L1-REGULARIZED LS (LASSO)

Consider a typical linear regression model in the form

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}\,, \tag{1}$$

where $\mathbf{y}=[y(1),y(2),\cdots,y(N)]^T$ is the observation, $\mathbf{x}=[x_1,x_2,\cdots,x_M]^T$ is the regression coefficients, $\mathbf{A}=[A^T(1),A^T(2),\cdots,A^T(N)]^T \in \mathbf{R}^{N\times M}$ is the design matrix, $\mathbf{e}=[e(1),e(2),\cdots,e(N)]^T$ is the additive noise, N and M are the number of observations and dimension of regression coefficients, respectively.

If $N \geq M$, the least-squares (LS) solution to (1) can be found by minimizing the quadratic loss function:

$$\hat{\mathbf{x}}_{LS} = \arg\min_{\mathbf{x}} \|\mathbf{y}-\mathbf{A}\mathbf{x}\|_2^2 = (\mathbf{A}^T\mathbf{A})^{-1}(\mathbf{A}^T\mathbf{y})\,, \tag{2}$$

where $\|\cdot\|_2^2$ denotes the L2 norm. It implicitly assumes the error $\boldsymbol{e}$ is white Gaussian distributed. When the number of observations is less than the dimension of coefficients (i.e., $N<M$), the problem is underdetermined with matrix $\mathbf{A}^T\mathbf{A}$ singular and the LS solution is unavailable.

To address the problem, an L2 regularization is often imposed on the regression coefficient, and the objective function of the L2-regularized linear regression consists of two parts, a quadratic loss function and an L2 constraint:

$$\begin{aligned}\hat{\mathbf{x}}_{L2} &= \arg\min_{\mathbf{x}}[\|\mathbf{y}-\mathbf{A}\mathbf{x}\|_2^2 + \mu\|\mathbf{x}\|_2^2]\,,\\ &= (\mathbf{A}^T\mathbf{A}+\mu\mathbf{I})^{-1}(\mathbf{A}^T\mathbf{y})\,,\end{aligned} \tag{3}$$

This study was partially supported by a grant from the Research Grants Council of the Hong Kong SAR, China and the University of Hong Kong CRCG Seed Fund.

where μ>0 is a regularization parameter. The L2-regularized linear regression method is also called the ridge regression.

Unlike the ridge algorithm, the Lasso algorithm imposes an L1 regularization on the regression coefficient and yields the following estimate:

$$\hat{\mathbf{x}}_{L1} = \arg\min_{\mathbf{x}}[\|\mathbf{y}-\mathbf{A}\mathbf{x}\|_2^2 + \mu\|\mathbf{x}\|_1]. \quad (4)$$

Note that the objective function in (4) though convex is a non-differentiable function and it is difficult to obtain its solution in an analytic form. An approximate closed form solution can be obtained by rewriting (4) as [1]:

$$\hat{\mathbf{x}}_{L1} = \arg\min_{\mathbf{x}}[\|\mathbf{y}-\mathbf{A}\mathbf{x}\|_2^2 + \mu\sum_{m=1}^{M}(x_m^2/|x_m|)], \quad (5)$$

where the L1 regularization had a similar form to an L2 norm with a weight $|x_m|$. (5) suggests computing the ridge regression iteratively to yield an iteratively re-weighted least-squares (IRLS) algorithm for the Lasso estimate:

$$\hat{\mathbf{x}}_{L1}^{(i+1)} = (\mathbf{A}^T\mathbf{A} + \mu\mathbf{\Lambda}^{(i)})^{-1}(\mathbf{A}^T\mathbf{y}), \quad (6)$$

where $i = 0,1,...$ is the iteration number, $\mathbf{\Lambda}^{(i)}$ is the generalized inverse (pseudo-inverse) of matrix $diag\{|x_{L1,1}^{(i)}|,|x_{L1,2}^{(i)}|,\cdots,|x_{L1,M}^{(i)}|\}$. A good initial value for (6) is the ridge regression estimate of (3). The iteration will stop until a maximum number is reached or the difference between two successive iteration steps is small enough.

III. M-ESTIMATION-BASED LASSO (M-LASSO)

"M-estimation" refers to "generalized maximum likelihood (ML) estimation", which is a formal approach to robust estimation [15]. In ML estimation, the objective function $\rho(e)$ to be minimized is -ln $p(e)$, where $p(e)$ is the probability density function (pdf) of the noise. However, it is rather difficult to estimate $p(e)$ accurately in practice. In M-estimation, $\rho(e)$ is chosen as an appropriate but fixed function to reduce the sensitivity of the estimator to outliers. In the context of Lasso, an M-estimate function $\rho(e)$ should be employed as the loss function instead of the conventional quadratic loss function $\|e\|_2^2$ to combat possible outliers. The resulting objective function reads:

$$\hat{\mathbf{x}}_{ME;L1} = \arg\min_{\mathbf{x}}[\rho(\mathbf{y}-\mathbf{A}\mathbf{x}) + \mu\|\mathbf{x}\|_1]. \quad (7)$$

The M-estimate function can be chosen as the Huber function:

$$\rho(e) = \begin{cases} e^2/2 & 0 \le |e| < \xi, \\ |e| - \xi + \xi^2/2 & |e| \ge \xi, \end{cases} \quad (8)$$

$$\Psi(e) = \begin{cases} e & 0 \le |e| < \xi, \\ \mathrm{sgn}(e)\cdot\xi & |e| \ge \xi, \end{cases} \quad (9)$$

or the Hampel's three part redescending function:

$$\rho(e) = \begin{cases} e^2/2 & 0 \le |e| < \xi_a, \\ \xi_a|e| - \xi_a^2/2 & \xi_a \le |e| < \xi_b, \\ \xi_a[(\xi_b+\xi_c) - \xi_a + (|e|-\xi_c)^2/(\xi_b-\xi_c)]/2 & \xi_b \le |e| < \xi_c, \\ \xi_a(\xi_b+\xi_c)/2 - \xi_a^2/2 & \xi_c \le |e|, \end{cases} \quad (10)$$

$$\Psi(e) = \begin{cases} e & 0 < |e| < \xi_a, \\ \mathrm{sgn}(e)\cdot\xi_a & \xi_a \le |e| < \xi_b, \\ \mathrm{sgn}(e)\cdot[(|e|-\xi_c)\xi_a/(\xi_b-\xi_c)] & \xi_b \le |e| < \xi_c, \\ 0 & |e| \ge \xi_c, \end{cases} \quad (11)$$

where $\Psi(e) = \partial\rho(e)/\partial e$ is the derivative of the M-estimation function $\rho(e)$, ξ, ξ_a, ξ_b, and ξ_c are the threshold parameters used to suppress the effect of outlier when the estimation error e is very large. We can see that the M-estimation function $\rho(e)$ behaves like a quadratic function when the error is below the thresholds ξ or ξ_a. The effect of errors with large amplitudes will be reduced substantially beyond the thresholds, and the Hampel's $\mathbf{\Psi}$ function will even assign zero weight to the outliers with large amplitudes.

By comparing Figs. 1 (a) and (b), we can see that the Huber function can only suppress the impulse noise to some extent, while the Hampel's redescending function can annihilate impulsive noise with large amplitudes. As a result, the Hampel's redescending function usually gives a better performance than the Huber function. In this paper, we propose to use the Hampel's function in the M-estimation-based Lasso (M-Lasso). A generalized Lasso similar to our M-Lasso was also proposed in [2] and [8] to improve the robustness of Lasso, but they only consider the Huber function. Our simulation results will show that the Hampel's function can achieve better performances than the Huber function in various impulsive noise environment.

The threshold parameters in the M-estimate functions are used to control the degree of outlier suppression and they need to be estimated based on the variance of the "impulse-free" estimation error, $\hat{\sigma}^2$. A simple estimate of these parameters is $\xi = 1.96\cdot\hat{\sigma}$ for Huber function, and $\xi_a = 1.96\cdot\hat{\sigma}$, $\xi_b = 2.24\cdot\hat{\sigma}$, $\xi_c = 2.58\cdot\hat{\sigma}$ for Hampel's function. The quantity $\hat{\sigma}$ can be estimated using the robust noise variance estimator given in [16] as:

$$\hat{\sigma} = \{median(|y(n) - y(n-1)|)\}/(\sqrt{2}\cdot 0.6745), \quad (12)$$

$n = 2,\cdots,N$. Without the L1 constraint, (7) can be solved by an IRLS algorithm. But, unlike the IRLS for L1 regularization that adds a weight matrix $\mathbf{\Lambda}^{(i)}$ to the coefficients $\mathbf{x}$, the IRLS for M-estimation assign a weighting matrix $\mathbf{W}^{(i)}$ to the design matrix **A** and observation **y**. More precisely, by differentiating the M-estimate function ρ(**y-Ax**) with respect to **x** and setting the derivative to zero, one gets,

$$\partial\rho/\partial\mathbf{x} = \sum_{n=1}^{N}\Psi(e(n)) = \sum_{n=1}^{N}[\Psi(e(n))/e(n)]\cdot e(n) = 0, \quad (13)$$

where $e(n) = y(n) - A(n)\mathbf{x}$. Suppose the coefficient estimate at i-th iteration is $\mathbf{x}_{ME}^{(i)}$ and the estimation error is $e^{(i)}$. Denote $\mathbf{W} = diag\{w(e_1),\cdots,w(e_m)\}$ where $w(e) = \begin{cases}\Psi(e)/e & \text{for } e \ne 0 \\ \Psi(0) & \text{for } e = 0\end{cases}$, the robust coefficient estimate at the (i+1)-th iteration is:

$$\hat{\mathbf{x}}_{ME}^{(i+1)} = (\mathbf{A}^T\mathbf{W}^{(i)}\mathbf{A})^{-1}(\mathbf{A}^T\mathbf{W}^{(i)}\mathbf{y}). \quad (14)$$

Since both the M-estimation loss function and the L1 regularization can be handled using the IRLS algorithms, the M-Lasso problem in (7) can be solved iteratively as

$$\hat{\mathbf{x}}_{ME;L1}^{(i+1)} = (\mathbf{A}^T\mathbf{W}^{(i)}\mathbf{A} + \mu\mathbf{\Lambda}^{(i)})^{-1}(\mathbf{A}^T\mathbf{W}^{(i)}\mathbf{y}). \quad (15)$$

The initial value for the M-Lasso can be chosen as the LS-based ridge regression result $\hat{\mathbf{x}}_{L2}$. Simulation results show that the IRLS for M-Lasso usually converges in 5-10 iterations. The algorithm is also readily applicable to weighted L1 norm.

IV. QRD-BASED M-LASSO

The M-Lasso method in Section III is a batch-processing algorithm, which works on a batch of observations. For online application and slowly time-varying parameter estimation, QRD-based time-recursive algorithm is attractive for its lower arithmetic complexity. Moreover, QRD-based method is numerically more stable than using the matrix inversion formula for solving LS problems in finite wordlength. To this ends, suppose that **x** changes with time, (1) becomes:

$$y(n) = A(n)\mathbf{x}(n) + e(n),\ n = 1,2,\cdots,N. \quad (16)$$

This time-varying linear estimation can be solved by several classical adaptive filtering algorithms, such as the recursive least-squares (RLS). The procedure of the QRD-based RLS algorithm is summarized in Table I. During each recursion, the current observation $y(n)$ and its design vector $A(n)$ are respectively appended to the matrices $\mathbf{R}(n\text{-}1)$ and $\mathbf{U}(n\text{-}1)$ to produce the updated $\mathbf{R}(n)$ and $\mathbf{U}(n)$ by a QRD. The coefficient $\mathbf{x}(n)$ is then estimated from $\mathbf{R}(n)$ and $\mathbf{U}(n)$ by back-substitution. Compared with batch processing with a $O(M^3)$ complexity, the QRD-RLS has a lower arithmetic complexity of $O(M^2)$ at each time instant. In the proposed recursive Lasso algorithm, the objective function at time n is:

$$\begin{aligned}\hat{\mathbf{x}}(n) &= \arg\min_{\mathbf{x}}[\| y(n) - A(n)\mathbf{x}(n)\|_2^2 + \kappa\|\mathbf{x}(n)\|_1] \\ &= \arg\min_{\mathbf{x}}[\| y(n) - A(n)\mathbf{x}(n)\|_2^2 + \kappa\sum_{i=1}^{M}\frac{x_m^{\ 2}(n)}{|\hat{x}_m(n-1)|}], \\ &= \arg\min_{\mathbf{x}}[\| D(n) - C(n)\mathbf{x}(n)\|_2^2, \end{aligned} \quad (17)$$

where $D(n) = \begin{bmatrix} y(n) \\ 0 \end{bmatrix}$, $C(n) = \begin{bmatrix} A(n) \\ \sqrt{\kappa}\mathbf{\Lambda}(n-1) \end{bmatrix}$, and $\mathbf{\Lambda}(n-1)$ is the generalized inverse of $diag\{|\hat{x}_1(n-1)|,\cdots,|\hat{x}_M(n-1)|\}$.

However, the Lasso cannot be solved directly by the recursive QRD in Table I due to the L1 regularization term. Inspired by the QRD-based recursive Levenberg-Marquardt algorithm developed in our previous work [17], a possible method to overcome the difficulty is to append successively the row vectors $\sqrt{\kappa M}E_l$ of the weighting matrix $\mathbf{\Lambda}$, instead of the whole matrix, to the previous QRD, where E_l is the l-th row of $\mathbf{\Lambda}$ and l=(n mod M)+1. More precisely, at each time instant, the algorithm (i) in Table I is executed once for the vector $[A(n), y(n)]$ and again for the vector $[\sqrt{\kappa M}E_l, 0]$.

The M-estimation can also be incorporated into the QRD-Lasso. The idea is to compute the estimate error $\hat{e}(n) = y(n) - A(n)\hat{\mathbf{x}}(n-1)$ first and then impose a weighting factor $w(n) = \Psi(\hat{e}(n))/\hat{e}(n)$ on the current observation $y(n)$ and design vector $A(n)$ during the QRD, where $\Psi(\hat{e}(n))$ is an M-estimation function. That is, the vector $[w(n)A(n), w(n)y(n)]$ will be appended to $\mathbf{R}(n-1)$ and $\mathbf{U}(n-1)$ to update the QRD, instead of $[A(n), y(n)]$.

To estimate the variance of the "impulsive-free" noise recursively for the M-estimation function, the formula proposed in [17] can be used:

$$\hat{\sigma}^2(n) = \lambda_\sigma\hat{\sigma}^2(n-1) + c_1(1-\lambda_\sigma)\text{median}[A_e(n)], \quad (18)$$

The forgetting factor λ_σ is a positive real number close to but smaller than one, $A_e(n) = \{\hat{e}^2(n),\cdots,\hat{e}^2(n-N_w+1)\}$, $c_1 = 1.483(1+5/(N_w-1))$ is a finite sample correction factor, and N_w is the length of the data set. With the M-estimation function and appropriate thresholds, the effects of the impulsive noise can be restrained or even removed in the QRD-M-Lasso method by assigning them with small or zero weights.

TABLE I. QRD-RLS ALGORITHM

Initialization:
$\mathbf{R}(0) = \sqrt{\delta}\mathbf{I}$, δ is a small positive constant;
$\mathbf{U}(0) = \mathbf{0}$; $\mathbf{x}(0) = \mathbf{0}$;
Recursion:
Given $\mathbf{R}(n-1)$, $\mathbf{U}(n-1)$, $\mathbf{x}(n-1)$, $A(n)$ and $y(n)$, compute at time n:
(i). $\begin{bmatrix} \mathbf{R}(n) & \mathbf{U}(n) \\ \mathbf{0} & \tilde{e}(n) \end{bmatrix} = \mathbf{Q}(n)\begin{bmatrix} \sqrt{\lambda}\mathbf{R}(n-1) & \sqrt{\lambda}\mathbf{U}(n-1) \\ A(n) & y(n) \end{bmatrix}$,
where $\mathbf{Q}(n)$ is calculated by Givens rotation and λ is the forgetting factor.
(ii). $\mathbf{x}(n) = \mathbf{R}^{-1}(n)\mathbf{U}(n)$ or a back-substitution solving.

V. SIMULATION RESULTS

We evaluate and compare the performances of the proposed M-Lasso and other related algorithms in a sparse signal reconstruction problem, which aims to recover a finite signal from a very limited number of noisy observations. In this simulation, the sparse signal **x** had a dimension M=512 and consists of 10 randomly-distributed spikes with amplitude ±1 and zeros elsewhere. The number of observations $y(n)$ was N=256 and the design vector $A(n)$ was generated from a normal distribution with zero mean and unit variance. The additive noise was simulated using the contaminated Gaussian noise model:

$$p(e) \sim (1-\eta)\mathcal{N}(0,\sigma_g) + \eta\mathcal{N}(0,\sigma_{im}), \quad (19)$$

where $\mathcal{N}(0,\sigma_g)$ and $\mathcal{N}(0,\sigma_{im})$ were two i.i.d. zero mean Gaussian processes with variance σ_g^2 and σ_{im}^2, and η denoted the occurrence probability of the impulsive noise with variance σ_{im}^2. The variance of the additive Gaussian component was $\sigma_g^2 = 10^{-2}$, resulting in a SNR of 30dB. The impulsive outliers with a much larger variance of $\sigma_{im}^2 = 10^4$ (the SNR is around -30dB) were randomly located in the 256 observations. Four methods, ridge, Lasso, M-Lasso with Huber function, and M-Lasso with Hampel's function, were tested in impulsive noise environment with different impulsive probability η. The regularization parameter for these methods was set as μ=0.01 and the maximum iteration number was 20.

One example of the sparse signal, the noisy observations (with η=0.02), and the recovery results are given in Fig. 2. It can be seen that only the proposed M-Lasso with Hampel's function can achieve a satisfactory result, although the M-Lasso with Huber function can also recover the signal to some extent. The mean squared deviation (MSD) criterion was used to evaluate the performances of the estimation, and it was given by $MSD = 10\log_{10}[\frac{1}{M}\sum_{m=1}^{M}(x_m - \hat{x}_m)^2]$, where $\hat{x}_m$ is the estimated signal. The MSD results shown in Fig. 3 were averages of 1000 independent Monte-Carlo runs. We can find that, all the Lasso-based method had good results when impulsive noise was absent. When the occurrence probabilities of impulsive noise are increased, only the M-Lasso with Hampel's function maintains a stable and satisfactory performance, while other methods were degraded considerably. We also tested the proposed QRD-M-Lasso method in tracking of a sparse signal with time-varying amplitudes. The experimental settings and parameters are the same to the previous experiment, except that the time-varying sparse signal $x_m(n)$ was generated by $x_m(n)=x_m(n\text{-}1)+\delta_m(n)$, where $\delta_m(n)$ is a random process with zero mean and variance 10^{-4}. The forgetting factor for QRD is set as λ=0.99, and the parameters for estimating the noise variance are λ_σ=0.9 and N_w=16. Two impulsive noise components with variance $\sigma_{im}^2=10^4$ were added at time 120 and 200 for visualization. Fig. 4 shows the tracking results of QRD-Lasso and QRD-M-Lasso. We can see that the recursive QRD-M-Lasso had a much better tracking performance than the QRD-Lasso under impulsive noises.

VI. Conclusion

An M-estimation-based Lasso (M-Lasso) algorithm was presented to improve the robustness of the conventional LS-based Lasso in impulsive noise environment. A recursive QRD-based M-Lasso was also presented for online applications. Simulations showed the M-Lasso method and its recursive implementations are more robust than their LS counterpart. It will find applications in model selection, compressed sensing, etc.

References

[1] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Royal. Statist. Soc B.*, vol. 58, no. 1, pp. 267–288, 1996.

[2] J. Fan and R. Li, "Variable selection via nonconcave penalized likelihood and its oracle properties,". *J. Am. Stat. Assoc.*, vol. 96, no. 456, pp. 1348–1360, Dec. 2001.

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

[4] H. Zou, "The adaptive lasso and its oracle properties," *J. Amer. Statist. Assoc.*, vol. 101, no. 476, pp. 1418–1429, 2006.

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

[6] S. J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, "An interior-point method for large-scale l1-regularized least squares," *IEEE J. Sel. Top. Signal Process.*, vol. 1, no. 4, pp. 606–617, Dec. 2007.

[7] M. Schmidt, G. Fung, and R. Rosales, "Fast optimization methods for L1-regularization: A comparative study and 2 new approaches," in *Proc. 18th European Conference on Machine Learning (ECML)*, Warshaw, Poland, 17–21 Sep., 2007.

[8] V. Roth, "The generalized Lasso," *IEEE Trans. Neural Netw.*, vol. 15, no. 1, pp. 16–28, Jan. 2004.

[9] P. Zhao and B. Yu, "On model selection consistency of lasso," *J. Mach. Learning Res.*, vol. 7, pp. 2541–2563, 2006.

[10] C. Leng, Y. Lin, and G. Wahba, "A note on the lasso and related procedures in model selection," *Stat. Sin.*, vol. 16, pp. 1273–1284, 2006.

[11] H. S. Wang, G. D. Li, and C. L. Tsai, "Regression coefficient and autoregressive order shrinkage and selection via the lasso," *J. Royal. Statist. Soc. B.*, vol. 69, no. 1, pp. 63–78, 2007.

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

[13] 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.

[14] 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.

[15] P. J. Huber, *Robust Statistics*, NY: John Wiley, 1981.

[16] Z. G. Zhang, S. C. Chan, K. L. Ho, and K. C. Ho, "On bandwidth selection in local polynomial regression analysis and its application to multi-resolution analysis of non-uniform data," *J. Signal Process. Syst.*, vol. 52, no. 3, pp. 263–280, Sept. 2008.

[17] S. C. Chan, Y. Zhou, and W. Y. Lau, "Approximate QR-based algorithms for recursive nonlinear least squares estimation," in *Proc. IEEE International Symposium on Circuits and Systems (ISCAS2005)*, Kobe, Japan, 23–26 May, 2005, vol. 5, pp. 4333–4336.

[18] S. C. Chan and Y. X. Zou, "A recursive least M-estimate algorithm for robust adaptive filtering in impulse noise: fast algorithm and convergence performance analysis," *IEEE Trans. Signal Process.*, vol. 2, no. 4, pp. 975–991, Apr. 2004.

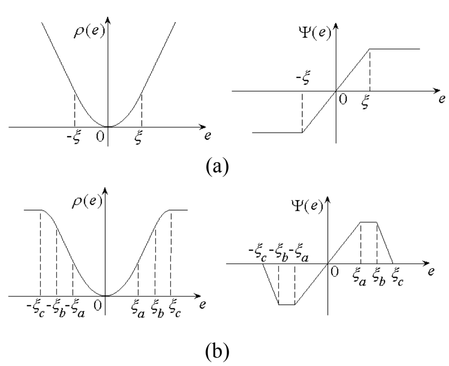

Figure 1. (a) The Huber M-estimate function, and (b) the Hampel's three part redescending M-estimate function.

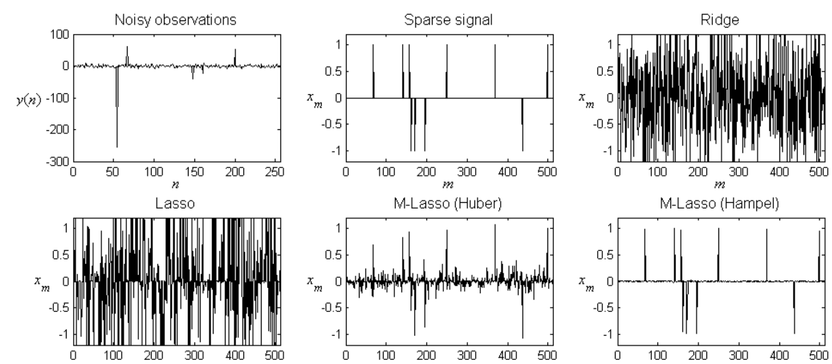

Figure 2. One sparse signal recovery example using different methods in impulsive noise enviroment (occurrence probabilities of impulsive noise is η=0.02).

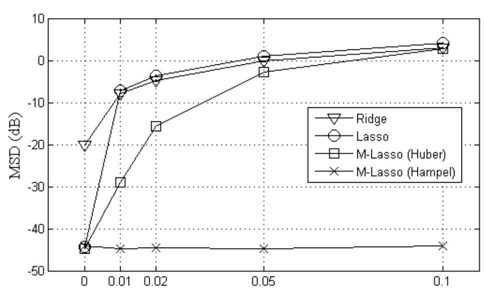

Figure 3. Comparision of different signal recovery methods under different occurrence probabilities of impulsive noise.

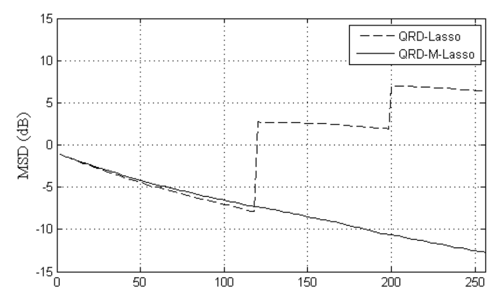

Figure 4. Comparision of QRD-Lasso and QRD-M-Lasso for time-varying sparse signal recovery in impulsive noise enviroment (two outliers were added at time instants 120 and 200).