Phase Retrieval via Coordinate Descent

Hing Cheung So

http://www.ee.cityu.edu.hk/~hcso

Department of Electronic Engineering, City University of Hong Kong

W.-J. Zeng and H. C. So, "Phase retrieval using coordinate descent techniques," filed with U.S. Patent Office (No. 15/344,279), Nov. 2016

W.-J. Zeng and H. C. So, "Coordinate descent algorithms for phase retrieval," arXiv, https://128.84.21.199/abs/1706.03474

Outline

- Introduction
- Coordinate Descent for Phase Retrieval
- Extension to Sparse Signals
- Application to Blind Equalization
- Numerical Examples
- Concluding Remarks
- List of References

Introduction

What is Phase Retrieval?

The aim is to recover a signal-of-interest using the magnitude-square or intensity observations of its linear transformation.

Mathematically, the problem is to find $\boldsymbol{x} \in \mathbb{C}^N$ (more precisely, up to a global phase $\phi \in [0, 2\pi)$ because $\mathrm{e}^{\mathrm{j}\phi}\boldsymbol{x}$ is also a solution) from M phaseless observations $b_m \in \mathbb{R}$:

$$b_m = \left|\boldsymbol{a}_m^H \boldsymbol{x}\right|^2 + \nu_m, \quad m = 1, \cdots, M$$

where $\boldsymbol{a}_m \in \mathbb{C}^N$ are known sampling vectors, $\nu_m \in \mathbb{R}$ are additive zero-mean noise terms and $M > N$.

Why Phase Retrieval is Important?

Many real-world problems can be boiled down to phase retrieval:

- Astronomy
- Computational Biology
- Crystallography
- Digital Communications
- Electron Microscopy
- Neutron Radiography
- Optical Imaging

Note that in some applications, $\boldsymbol{a}_m$ is the discrete Fourier transform vector, although it can be generalized to any linear mappings.

How to Perform Phase Retrieval?

Adopting least squares (LS) criterion, $\boldsymbol{x}$ is determined from:

$$\min_{\boldsymbol{x}\in\mathbb{C}^N} f(\boldsymbol{x}) := \sum_{m=1}^{M} \left(\left| \boldsymbol{a}_m^H \boldsymbol{x} \right|^2 - b_m \right)^2$$

It is a nonconvex optimization problem where minimizing a multivariate fourth-order polynomial is required, which is generally NP-hard.

Conventional methods include

- Gerchberg-Saxton algorithm (GSA): solve the nonconvex problem via alternating projection.

- Wirtinger flow (WF): solve the nonconvex problem via gradient descent.

- PhaseLift and PhaseCut: relax the nonconvex problem to a convex program.

However, these methods have the drawbacks of requiring:

- Lengthy observations or large M.

- Large number of iterations.

- High computational complexity.

Coordinate Descent for Phase Retrieval

The key idea is to apply coordinate descent (CD): a single unknown is solved at each iteration while all other variables are kept fixed, which results in minimizing a univariate quartic polynomial only.

Using real-valued representation, the LS minimization is:

$$\min_{\bar{\boldsymbol{x}} \in \mathbb{R}^{2N}} f(\bar{\boldsymbol{x}}) := \sum_{m=1}^{M} \left(q_m(\bar{\boldsymbol{x}}) - b_m\right)^2$$

where

$$\bar{\boldsymbol{x}} = \begin{bmatrix} \mathrm{Re}(\boldsymbol{x}) \\ \mathrm{Im}(\boldsymbol{x}) \end{bmatrix}, \; q_m(\bar{\boldsymbol{x}}) = \bar{\boldsymbol{x}}^T \bar{\boldsymbol{A}}_m \bar{\boldsymbol{x}}, \; \bar{\boldsymbol{A}}_m = \begin{bmatrix} \mathrm{Re}(\boldsymbol{A}_m) & -\mathrm{Im}(\boldsymbol{A}_m) \\ \mathrm{Im}(\boldsymbol{A}_m) & \mathrm{Re}(\boldsymbol{A}_m) \end{bmatrix}, \; \boldsymbol{A}_m = \boldsymbol{a}_m \boldsymbol{a}_m^H$$

CD is an iterative procedure that successively minimizes the objective function along coordinate directions.

Denote the result of the kth iteration as $\bar{\boldsymbol{x}}^k = [\bar{x}_1^k \ \cdots \ \bar{x}_{2N}^k]^T$.

At kth iteration, we minimize $f(\bar{\boldsymbol{x}}^k)$ w.r.t. i_kth ($i_k \in \{1, \cdots, 2N\}$) variable while keeping the remaining $2N-1$ $\{\bar{x}_i^k\}_{i \neq i_k}$ fixed.

This is equivalent to performing 1-D search along i_k th coordinate:

$$\alpha_k = \arg\min_{\alpha \in \mathbb{R}} f\left(\bar{\boldsymbol{x}}^k + \alpha \boldsymbol{e}_{i_k}\right)$$

where $\boldsymbol{e}_{i_k}$ is the unit vector with the i_kth entry being one and all other entries being zero.

Then $\bar{\boldsymbol{x}}$ is updated by

$$\bar{\boldsymbol{x}}^{k+1} = \bar{\boldsymbol{x}}^{k} + \alpha_k \boldsymbol{e}_{i_k}$$

which implies that only the i_kth component is adjusted:

$$\bar{x}_{i_k}^{k+1} \leftarrow \bar{x}_{i_k}^{k} + \alpha_k$$

while all other components remain unchanged.

Since $\bar{\boldsymbol{x}}^k$ is known, $f\left(\bar{\boldsymbol{x}}^k + \alpha \boldsymbol{e}_{i_k}\right)$ is a univariate function of α.

Thus, finding α_k is 1-D minimization problem.

High-Level Algorithm

The proposed CD is outlined in Algorithm 1:

Algorithm 1 CD for Phase Retrieval

Initialization: Determine $\bar{\boldsymbol{x}}^0 \in \mathbb{R}^{2N}$ using the spectral method [2].

for $k = 0, 1 \cdots,$ **do**

Choose index $i_k \in \{1, \cdots, 2N\}$;

$\alpha_k = \arg\min_{\alpha \in \mathbb{R}} f\left(\bar{\boldsymbol{x}}^k + \alpha \boldsymbol{e}_{i_k}\right)$;

$\bar{x}_{i_k}^{k+1} \leftarrow \bar{x}_{i_k}^k + \alpha_k$;

Stop if $f\left(\bar{\boldsymbol{x}}^k\right) - f\left(\bar{\boldsymbol{x}}^{k+1}\right) < \epsilon$ is satisfied, where $\epsilon > 0$ is a small tolerance parameter.

end for

Selection rules for the coordinate index i_k include:

- Cyclic rule (CCD): i_k takes value cyclically from $\{1, \cdots, 2N\}$, and thus one cycle corresponds to $2N$ iterations.

- Random rule (RCD): i_k is randomly selected from $\{1, \cdots, 2N\}$ with equal probability.

- Greedy rule (GCD): i_k is chosen as

$$i_k = \arg\max_i \left|\nabla f_i(\bar{\boldsymbol{x}}^k)\right|, \quad \nabla f_i(\bar{\boldsymbol{x}}^k) = \frac{\partial f(\bar{\boldsymbol{x}}^k)}{\partial \bar{x}_i^k}$$

i.e., coordinate with largest absolute value of the partial derivative, and full gradient at each iteration is needed.

Closed-form solution for α_k is derived as follows.

Let

$$\varphi(\alpha) = f(\bar{\boldsymbol{x}} + \alpha \boldsymbol{e}_i) = \sum_{m=1}^{M} \left(q_m(\bar{\boldsymbol{x}} + \alpha \boldsymbol{e}_i) - b_m\right)^2$$

where the mth term is

$$\varphi_m(\alpha) = \left(q_m(\bar{\boldsymbol{x}} + \alpha \boldsymbol{e}_i) - b_m\right)^2$$

Expanding $q_m(\bar{\boldsymbol{x}} + \alpha \boldsymbol{e}_i)$ results in:

$$q_m(\bar{\boldsymbol{x}} + \alpha \boldsymbol{e}_i) = \alpha^2 \boldsymbol{e}_i^T \bar{\boldsymbol{A}}_m \boldsymbol{e}_i + 2\alpha \boldsymbol{e}_i^T \bar{\boldsymbol{A}}_m \bar{\boldsymbol{x}} + \bar{\boldsymbol{x}}^T \bar{\boldsymbol{A}}_m \bar{\boldsymbol{x}} \triangleq c_{2,i}^m \alpha^2 + c_{1,i}^m \alpha + c_0^m$$

where $c_{2,i}^m$, $c_{1,i}^m$ and c_0^m are coefficients of univariate quadratic polynomial.

Further manipulation yields:

$$c_{2,i}^m = \boldsymbol{e}_i^T \bar{\boldsymbol{A}}_m \boldsymbol{e}_i = \left[\bar{\boldsymbol{A}}_m\right]_{i,i} = \begin{cases} \left|[\boldsymbol{a}_m]_i\right|^2, \ i = 1, \cdots, N \\ \left|[\boldsymbol{a}_m]_{i-N}\right|^2, \ i = N+1, \cdots, 2N. \end{cases}$$

$$c_{1,i}^m = 2\boldsymbol{e}_i^T \bar{\boldsymbol{A}}_m \bar{\boldsymbol{x}} = \begin{cases} \mathrm{Re}\left(\left(\boldsymbol{a}_m^H \boldsymbol{x}\right) [\boldsymbol{a}_m]_i\right), \ i = 1, \cdots, N \\ \mathrm{Im}\left(\left(\boldsymbol{a}_m^H \boldsymbol{x}\right) [\boldsymbol{a}_m]_{i-N}\right), \ i = N+1, \cdots, 2N \end{cases}$$

because

$$\bar{\boldsymbol{A}}_m \bar{\boldsymbol{x}} = \begin{bmatrix} \mathrm{Re}(\boldsymbol{A}_m \boldsymbol{x}) \\ \mathrm{Im}(\boldsymbol{A}_m \boldsymbol{x}) \end{bmatrix} = \begin{bmatrix} \mathrm{Re}\left(\left(\boldsymbol{a}_m^H \boldsymbol{x}\right) \boldsymbol{a}_m\right) \\ \mathrm{Im}\left(\left(\boldsymbol{a}_m^H \boldsymbol{x}\right) \boldsymbol{a}_m\right) \end{bmatrix}$$

$$c_0^m = \bar{\boldsymbol{x}}^T \bar{\boldsymbol{A}}_m \bar{\boldsymbol{x}} = \boldsymbol{x}^H \boldsymbol{A}_m \boldsymbol{x} = \left|\boldsymbol{a}_m^H \boldsymbol{x}\right|^2$$

Since $q_m(\bar{\boldsymbol{x}} + \alpha \boldsymbol{e}_i)$ is quadratic, $\varphi_m(\alpha)$ is a univariate quartic polynomial of α:

$$\varphi_m(\alpha) = d_{4,i}^m \alpha^4 + d_{3,i}^m \alpha^3 + d_{2,i}^m \alpha^2 + d_{1,i}^m \alpha + d_0^m$$

where

$$\begin{aligned}
d_{4,i}^m &= \left(c_{2,i}^m\right)^2 \\
d_{3,i}^m &= 2c_{2,i}^m c_{1,i}^m \\
d_{2,i}^m &= \left(c_{1,i}^m\right)^2 + 2c_{2,i}^m \left(c_0^m - b_m\right) \\
d_{1,i}^m &= 2c_{1,i}^m \left(c_0^m - b_m\right) \\
d_0^m &= \left(c_0^m - b_m\right)^2
\end{aligned}$$

Recall

$$\varphi(\alpha) = \sum_{m=1}^{M} \varphi_m(\alpha)$$

We have:

$$\varphi(\alpha) = d_{4,i}\alpha^4 + d_{3,i}\alpha^3 + d_{2,i}\alpha^2 + d_{1,i}\alpha + d_0$$

where

$$d_0 = \sum_{m=1}^{M} d_0^m, \quad d_{j,i} = \sum_{m=1}^{M} d_{j,i}^m, \quad j = 1, \cdots, 4$$

Hence α is easily solved with closed-form expression from:

$$\varphi'(\alpha) = 4d_{4,i}\alpha^3 + 3d_{3,i}\alpha^2 + 2d_{2,i}\alpha + d_{1,i} = 0$$

whose complexity is merely $\mathcal{O}(1)$.

Computational Complexity

- $\mathcal{O}(M)$ per iteration for CCD and RCD; note that one cycle corresponds to $2N$ iterations.

- $\mathcal{O}(MN)$ per iteration for GCD.

Convergence Analysis

- Three CD algorithms converge to a stationary point regardless of the initial value.

- RCD locally converges to the global minimum and achieves exact retrieval at geometric rate with high probability provided that M is large enough.

Extension to Sparse Signals

If $\boldsymbol{x}$ is sparse, then the real-valued $\bar{\boldsymbol{x}}$ is also sparse.

The sparse signal retrieval problem is formulated as:

$$\min_{\bar{\boldsymbol{x}} \in \mathbb{R}^{2N}} \sum_{m=1}^{M} \left(\bar{\boldsymbol{x}}^T \bar{\boldsymbol{A}}_m \bar{\boldsymbol{x}} - b_m \right)^2, \quad \text{subject to } \|\bar{\boldsymbol{x}}\|_0 \le s$$

That is, the number of nonzero elements in $\bar{\boldsymbol{x}}$ is at most s.

Recall in linear compressed sensing, the ℓ_0 -norm is approximated by the ℓ_1-norm so that the resultant problem becomes a convex optimization problem.

Widely-used methods include the least absolute shrinkage and selection operator (LASSO):

$$\min \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2^2, \quad \text{subject to } \|\boldsymbol{x}\|_1 \leq \delta, \quad \delta \geq 0$$

basis pursuit (BP):

$$\min \|\boldsymbol{x}\|_1, \quad \text{subject to } \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2^2 \leq \epsilon, \quad \epsilon \geq 0$$

and ℓ_1-regularization:

$$\min \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \tau \|\boldsymbol{x}\|_1, \quad \tau \geq 0$$

Adopting ℓ_1-regularization, sparse phase retrieval is

$$\min_{\bar{\boldsymbol{x}} \in \mathbb{R}^{2N}} g(\bar{\boldsymbol{x}}) := \sum_{m=1}^{M} \left(\bar{\boldsymbol{x}}^T \bar{\boldsymbol{A}}_m \bar{\boldsymbol{x}} - b_m \right)^2 + \tau \|\bar{\boldsymbol{x}}\|_1, \quad \|\bar{\boldsymbol{x}}\|_1 = \sum_{i=1}^{2N} |\bar{x}_i|$$

High-Level Algorithm

The CD for sparse signals is outlined in Algorithm 2:

Algorithm 2 CD for Sparse Phase Retrieval

Initialization: Determine $\bar{\boldsymbol{x}}^0 \in \mathbb{R}^{2N}$ using the spectral method [2].

for $k = 0, 1 \ \cdots \ ,$ **do**

Choose index $i_k \in \{1, \cdots, 2N\}$;

$$\alpha_k = \arg\min_{\alpha \in \mathbb{R}} f\left(\bar{\boldsymbol{x}}^k + \alpha \mathbf{e}_{i_k}\right) + \tau \|\bar{\boldsymbol{x}} + \alpha \mathbf{e}_i\|_1;$$

$$\bar{x}_{i_k}^{k+1} \leftarrow \bar{x}_{i_k}^k + \alpha_k;$$

Stop if $f\left(\bar{\boldsymbol{x}}^k\right) - f\left(\bar{\boldsymbol{x}}^{k+1}\right) < \epsilon$ is satisfied, where $\epsilon > 0$ is a small tolerance parameter.

end for

As there is no gradient for $g(\bar{\boldsymbol{x}})$, GCD is not implementable because it requires gradient for index selection.

We only present the CCD and RCD for the ℓ_1-regularization, and they are referred to as ℓ_1-CCD and ℓ_1-RCD.

The steps of the CD for solving $g(\bar{\boldsymbol{x}})$ are similar to those in Algorithm 1 except that an ℓ_1-norm term is added to the scaler minimization problem of:

$$\min_{\alpha \in \mathbb{R}} \psi(\alpha) := \varphi(\alpha) + \tau \|\bar{\boldsymbol{x}} + \alpha \boldsymbol{e}_i\|_1$$

Ignoring the terms independent to α yields

$$\min_{\alpha \in \mathbb{R}} \psi(\alpha) := \varphi(\alpha) + \tau |\alpha + \bar{x}_i|$$

Making a change of variable $\beta = \alpha + \bar{x}_i$ and ignoring the irrelevant components, we obtain an equivalent scalar minimization problem:

$$\min_{\beta \in \mathbb{R}} \psi(\beta) := u_4 \beta^4 + u_3 \beta^3 + u_2 \beta^2 + u_1 \beta + \tau |\beta|$$

where

$$\begin{aligned} u_4 &= d_{4,i} \\ u_3 &= d_{3,i} - 4\bar{x}_i d_{4,i} \\ u_2 &= d_{2,i} - 3\bar{x}_i d_{3,i} + 6\bar{x}_i^2 d_{4,i} \\ u_1 &= d_{1,i} - 2\bar{x}_i d_{2,i} + 3\bar{x}_i^2 d_{3,i} - 4\bar{x}_i^3 d_{4,i} \end{aligned}$$

Although $\psi(\beta)$ is non-smooth due to the absolute term, there is a closed-form solution.

We study $\psi(\beta)$ in two intervals: $[0,\infty)$ and $(-\infty,0)$.

Define $\mathcal{S}^+$ containing the stationary points of $\psi(\beta)$ in the interval $[0,\infty)$, i.e., $\mathcal{S}^+$ is the set of real positive roots of:

$$4u_4\beta^3 + 3u_3\beta^2 + 2u_2\beta + (u_1 + \tau) = 0, \;\; \beta \geq 0$$

$\mathcal{S}^+$ can be empty, or has at most 3 positive elements.

Similarly, $\mathcal{S}^-$ is the set that contains the stationary points of $\psi(\beta)$ in $(-\infty,0)$, i.e., real negative roots of

$$4u_4\beta^3 + 3u_3\beta^2 + 2u_2\beta + (u_1 - \tau) = 0, \;\; \beta < 0$$

Again, $\mathcal{S}^-$ can be empty, or has at most 3 entries.

The minimizer of $\psi(\beta)$ in $[0,\infty)$ and $(-\infty,0)$ must be an element of $\mathcal{S}^+$ and $\mathcal{S}^-$, respectively.

Minimizer in $[0,\infty)$ must also include the boundary, i.e., 0.

Hence

$$\beta^\star = \arg\min_{\beta} \psi(\beta), \ \beta \in \{0 \cup \mathcal{S}^+ \cup \mathcal{S}^-\}$$

We only need to evaluate $\psi(\beta)$ with at most 7 elements.

The coordinate of ℓ_1-regularized CD is updated as:

$$\bar{x}_{i_k}^{k+1} \leftarrow \beta^\star$$

If $\mathcal{S}^+ \cup \mathcal{S}^- = \emptyset$, then $\bar{x}_{i_k}^{k+1} = \beta^\star = 0$, making the solution sparse.

Application to Blind Equalization

Consider a communication system with discrete-time complex baseband signal model:

$$r(n) = s(n) \otimes h(n) + \nu(n)$$

where

$r(n)$ is received signal
$s(n)$ is transmitted data symbol
$h(n)$ is channel impulse response
$\nu(n)$ is additive white noise

Blind equalization aims at recovering $s(n)$ without knowing $h(n)$.

Define the equalizer with P coefficients:

$$\boldsymbol{w} = [w_0 \ \cdots \ w_{P-1}]^T$$

and

$$\boldsymbol{r}_n = [r(n) \ \cdots \ r(n-P+1)]^T$$

The equalizer output is

$$y(n) = \sum_{i=0}^{P-1} w_i^* r(n-i) = \boldsymbol{w}^H \boldsymbol{r}_n$$

As many modulated signals such as PSK, FM, and PM, are of constant modulus, we apply the constant modulus criterion:

$$\min_{\boldsymbol{w}} \sum_n \left(|\boldsymbol{w}^H \boldsymbol{r}_n|^2 - \kappa \right)^2, \quad \kappa = \frac{\mathbb{E}\left[|s(n)|^4\right]}{\mathbb{E}\left[|s(n)|^2\right]}$$

where κ is the known dispersion constant.

Numerical Examples

All methods use the same initial value obtained from the spectral method.

The measurement vectors $\{\boldsymbol{a}_m\}$ satisfy a complex standard i.i.d. Gaussian distribution.

Convergence Behavior and Statistical Performance

$\boldsymbol{x}$ and noise ν_m are i.i.d. Gaussian while $N = 64$ and $M = 6N$.

Note that it is fair to compare $2N$ iterations (one cycle) for the CD with one iteration of WF because the computational complexity of the CCD and RCD per cycle is the same as the WF per iteration:

N	64	128	256	512	1024
CCD	2.84×10^{-4}	1.83×10^{-3}	1.07×10^{-2}	3.95×10^{-2}	1.56×10^{-1}
WF	2.34×10^{-4}	1.61×10^{-3}	9.23×10^{-3}	3.68×10^{-2}	1.44×10^{-1}

Runtime Comparison (in sec.)

Reduction of the objective function normalized w.r.t. $||\boldsymbol{b}||^2$:

$$\frac{f(\bar{\boldsymbol{x}}^k) - f(\bar{\boldsymbol{x}}^\star)}{||\boldsymbol{b}||^2}, \quad \boldsymbol{b} = [b_1 \ \cdots \ b_M]^T$$

SNR is defined as

$$\text{SNR} = \frac{\mathbb{E}\left[||\boldsymbol{b}||^2\right]}{M\sigma_\nu^2}$$

Relative recovery error:

$$\frac{\|\bar{\boldsymbol{x}}^k - T_{\phi_k}(\bar{\boldsymbol{x}}^\star)\|^2}{\|\bar{\boldsymbol{x}}^\star\|^2}$$

where $T_{\phi_k}(\bar{\boldsymbol{x}}^\star)$ is extracted from $\mathrm{e}^{\mathrm{j}\phi_k}\boldsymbol{x}^\star$ such that $\|\boldsymbol{x}^k - \mathrm{e}^{\mathrm{j}\phi_k}\boldsymbol{x}^\star\|$ is minimum, which reflects the convergence speed.

Successful recovery means:

$$\frac{\|\bar{\boldsymbol{x}}^k - T_{\phi_k}(\bar{\boldsymbol{x}}^\star)\|^2}{\|\bar{\boldsymbol{x}}^\star\|^2} < 10^{-5}$$

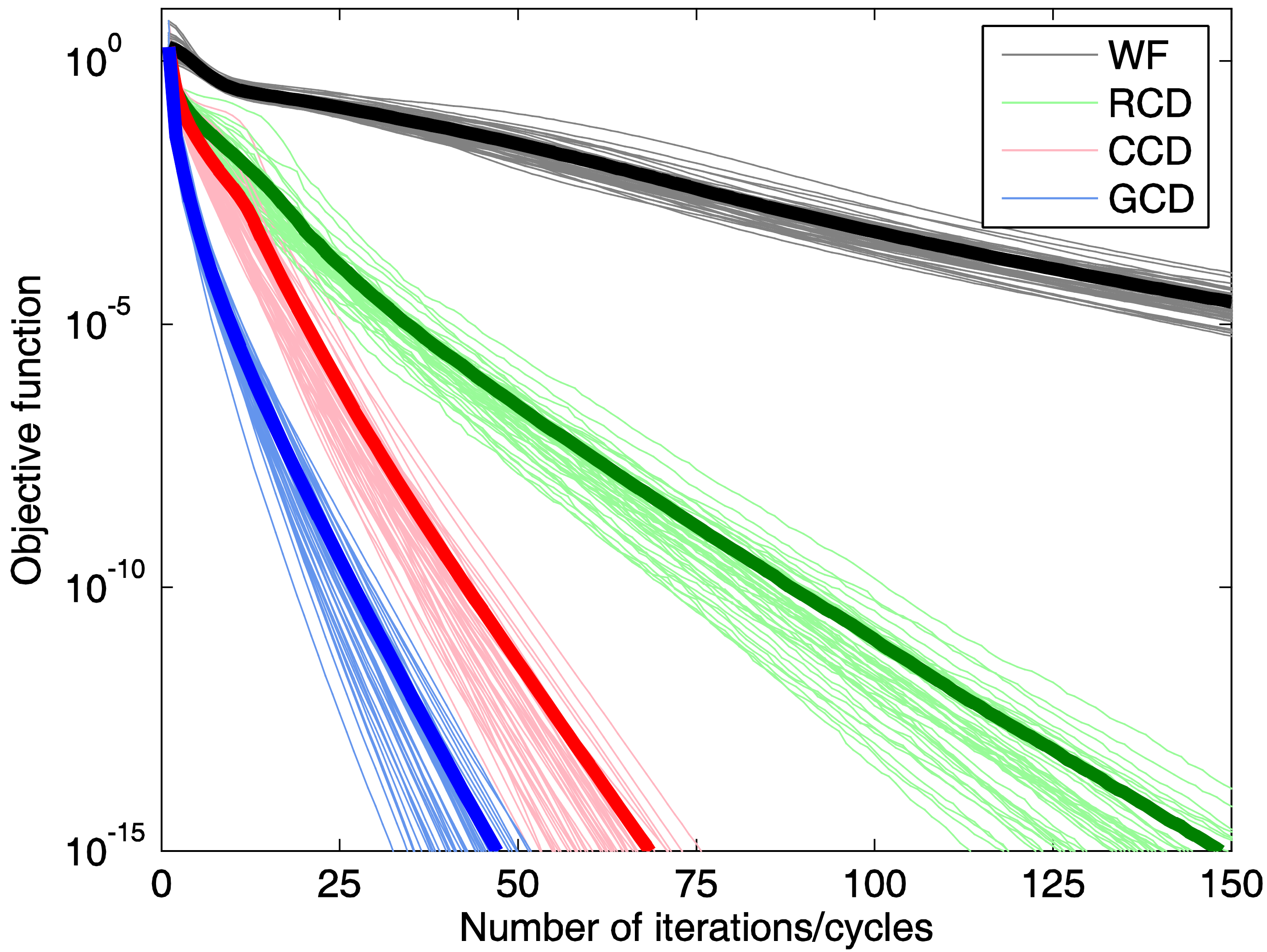

Figure 1: Normalized reduction of objective function versus number of iterations/cycles at SNR = 20 dB

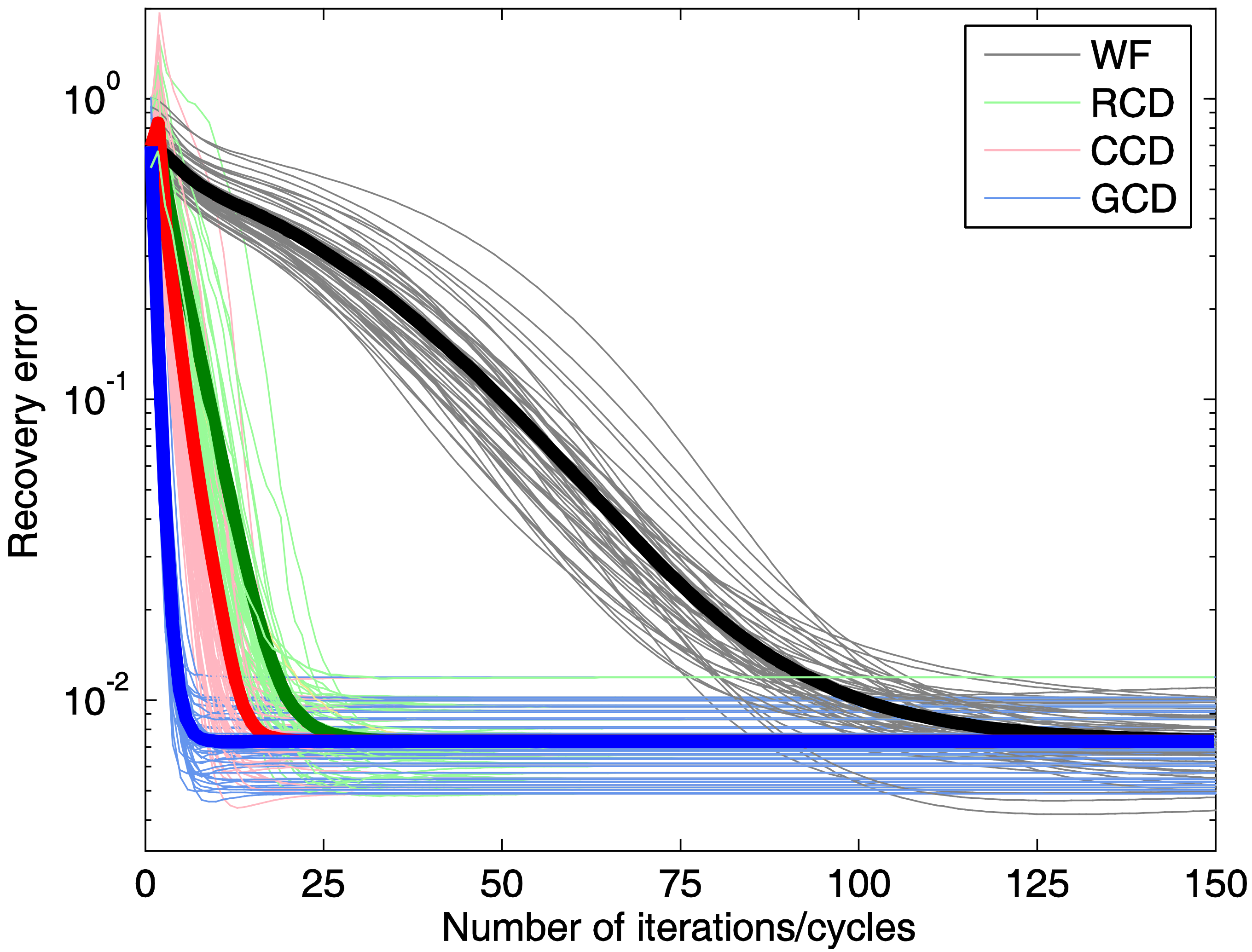

Figure 2: Relative recovery error versus number of iterations/cycles at SNR = 20 dB

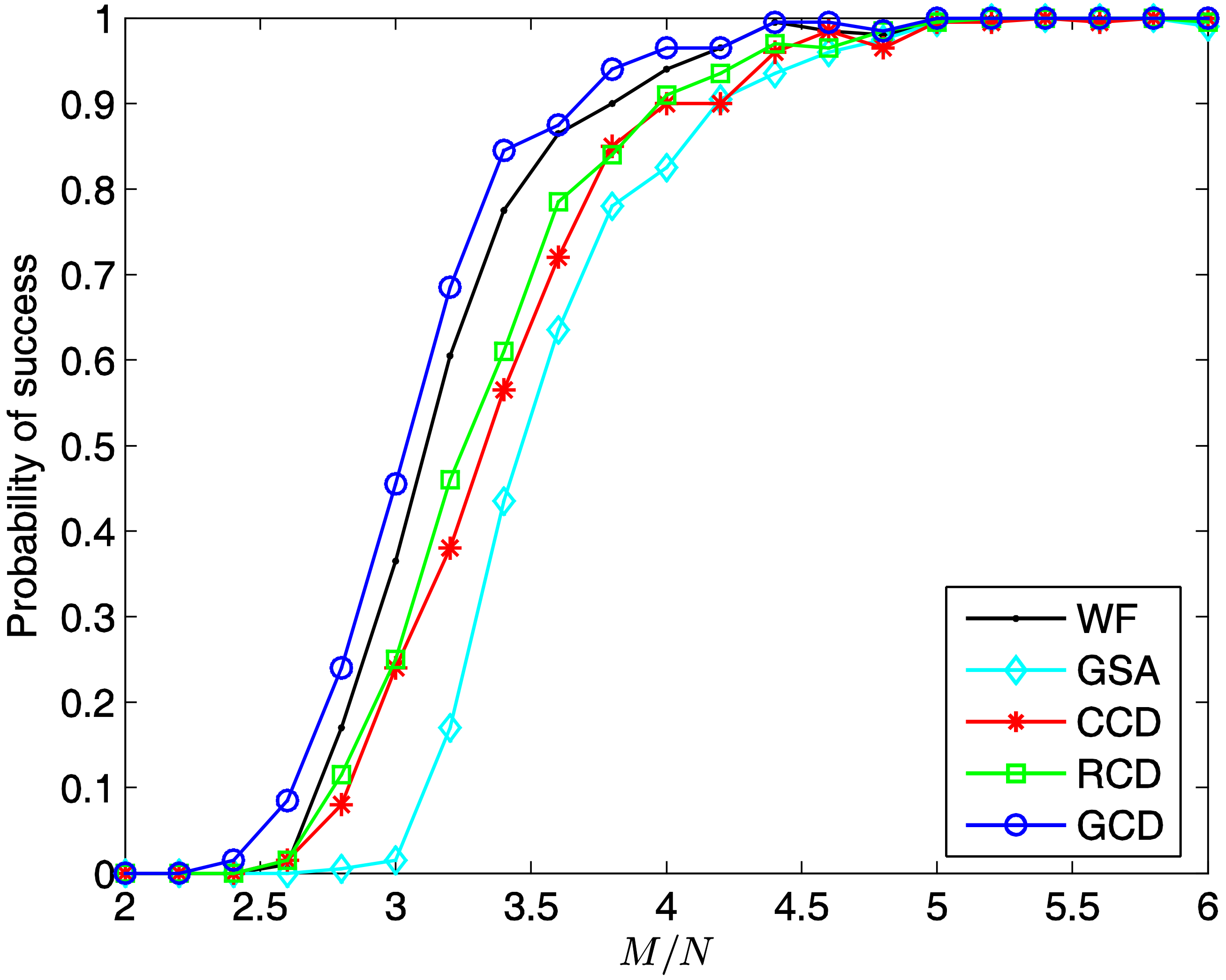

Figure 3: Empirical probability of success versus number of noise-free measurements

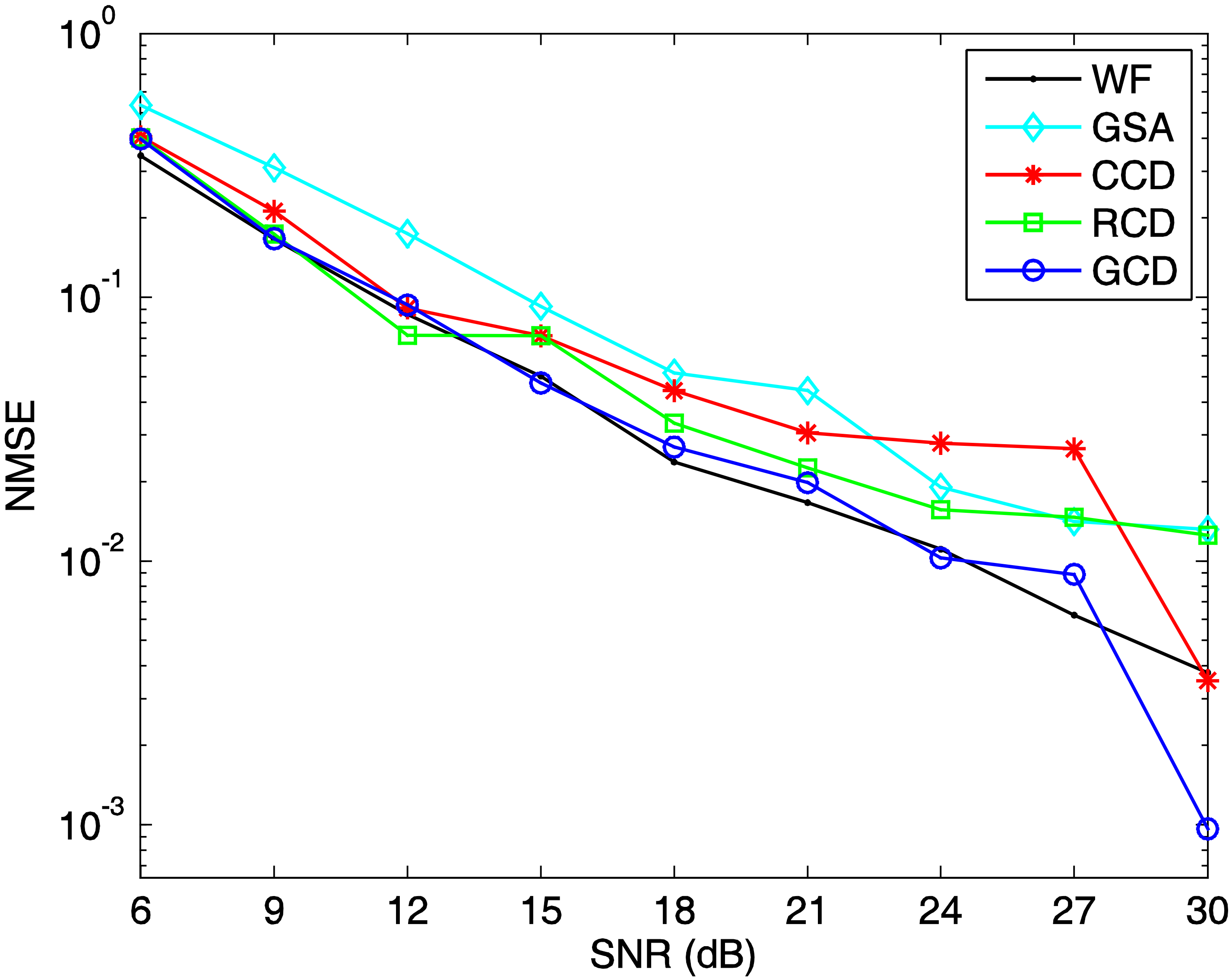

Figure 4: NMSE of recovered signal versus SNR

Sparse Phase Retrieval Performance

$\tau = 2.35M$ is used for ℓ_1-CCD and ℓ_1-RCD.

Support of sparse signal is randomly selected from $[1, N]$ where $N = 64$, while $K = 5$ and $M = 2N$.

The real and imaginary parts of the nonzero coefficients of $\boldsymbol{x}$ are drawn as random uniform variables in the range $[-2/\sqrt{2}, -1/\sqrt{2}] \cup [1/\sqrt{2}, 2/\sqrt{2}]$.

Comparison with WF and sparse GSA using hard-thresholding is included.

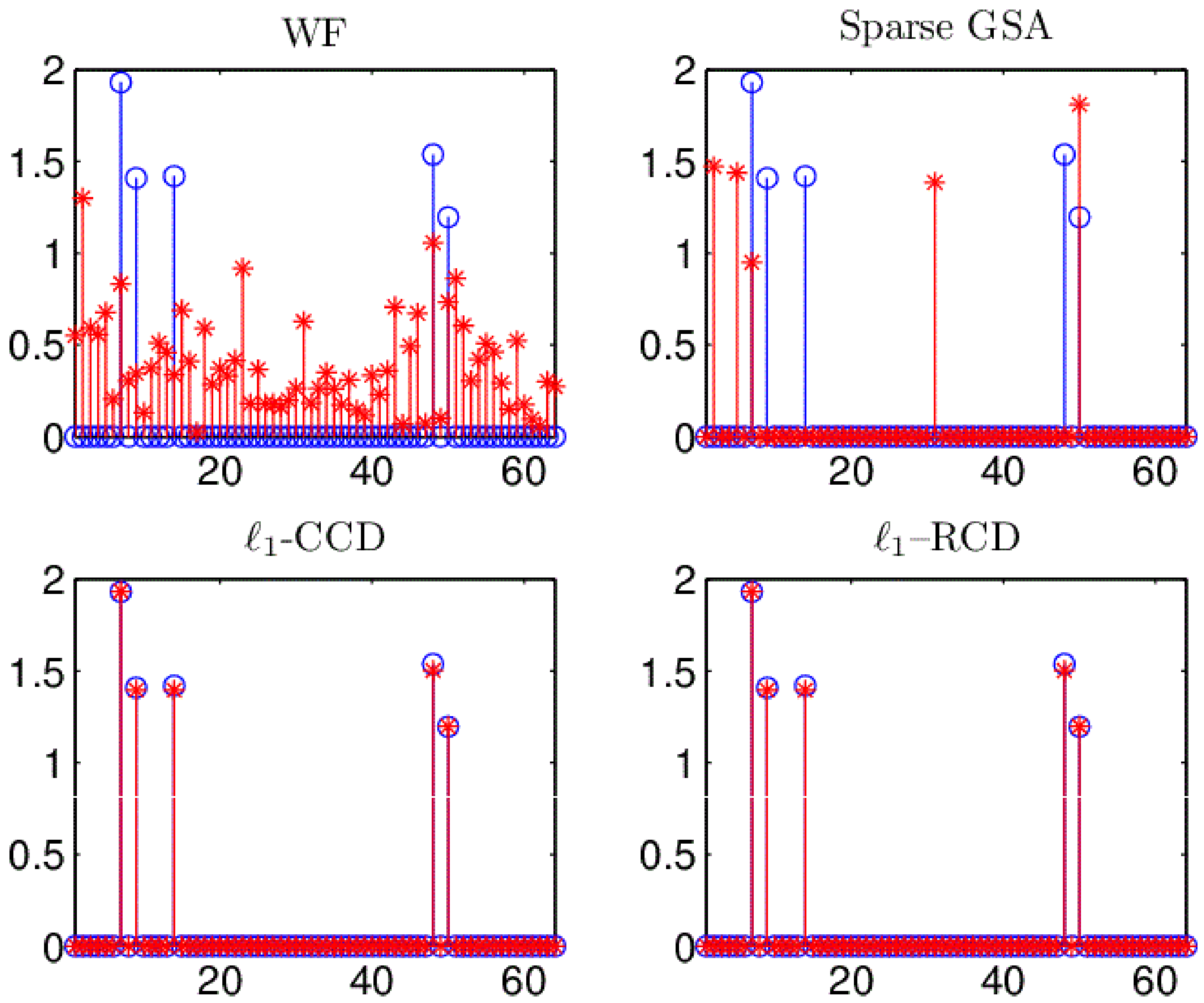

Figure 5: Magnitudes of recovered signals

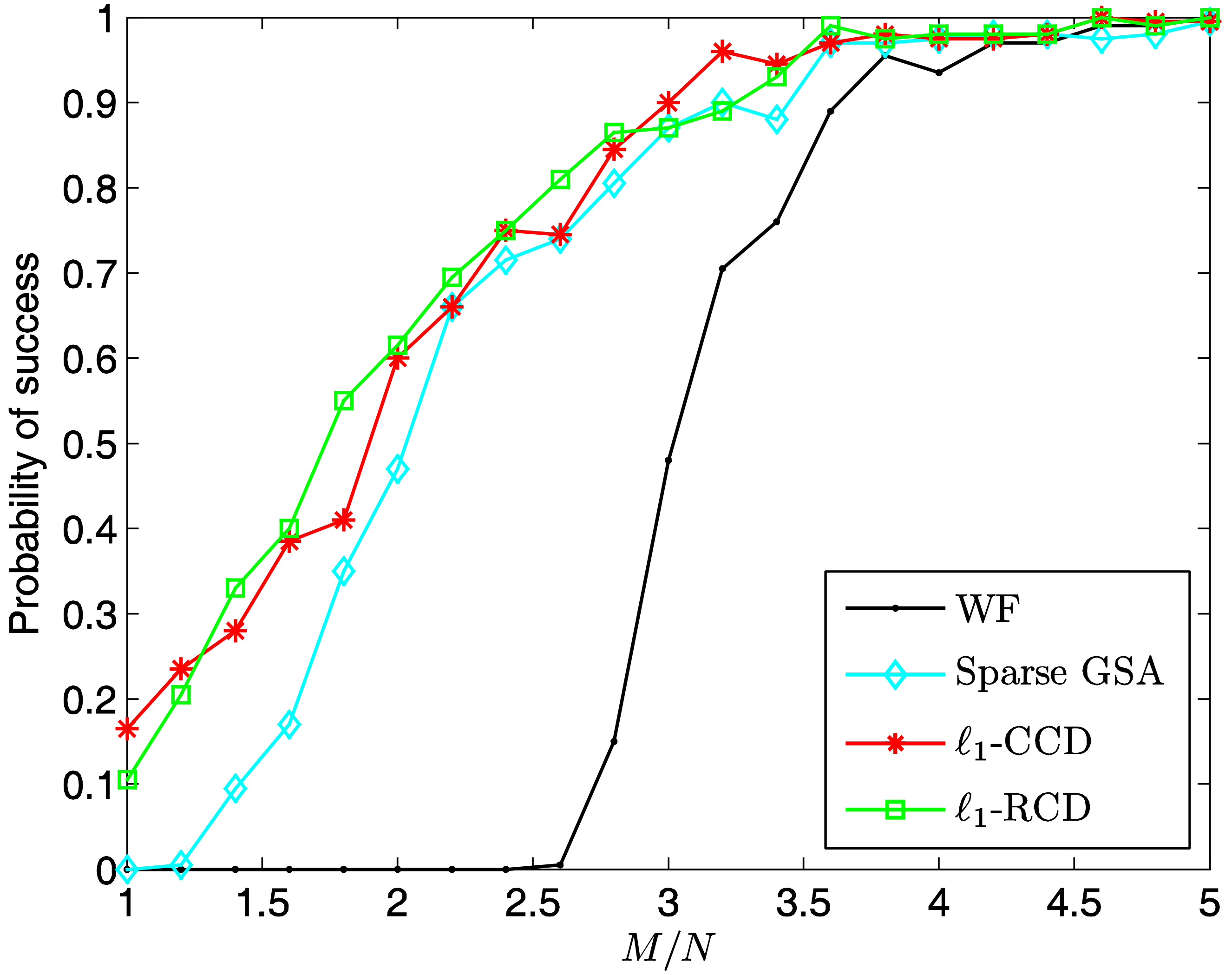

Figure 6: Probability of success versus number of noise-free measurements for sparse phase retrieval

Blind Equalization Performance

QPSK: $s(n) \in \{1, -1, \mathrm{j}, -\mathrm{j}\} \Rightarrow \kappa = 1$

FIR communication channel: $\{0.4, 1, -0.7, 0.6, 0.3, -0.4, 0.1\}$

Equalization quality is measured using the quantified inter-symbol interference (ISI):

$$\mathrm{ISI} = \frac{\sum_n |v(n)|^2 - \max_n |v(n)|^2}{\max_n |v(n)|^2}, \quad v(n) = h(n) \otimes w(n)$$

Comparison with WF and conventional super-exponential algorithm (SEA) is included.

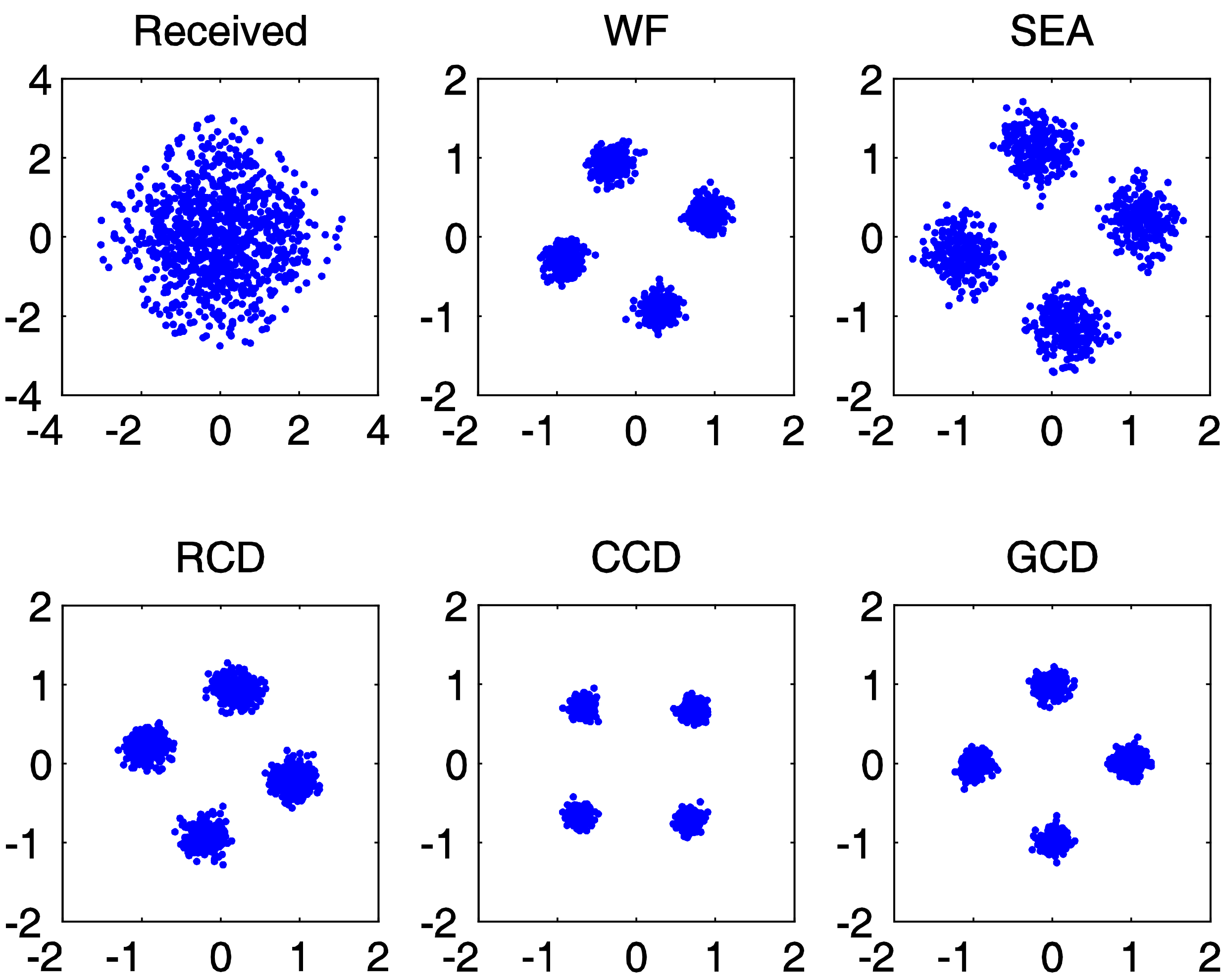

Figure 7: Scatter plots of constellations of received signal and equalizer outputs

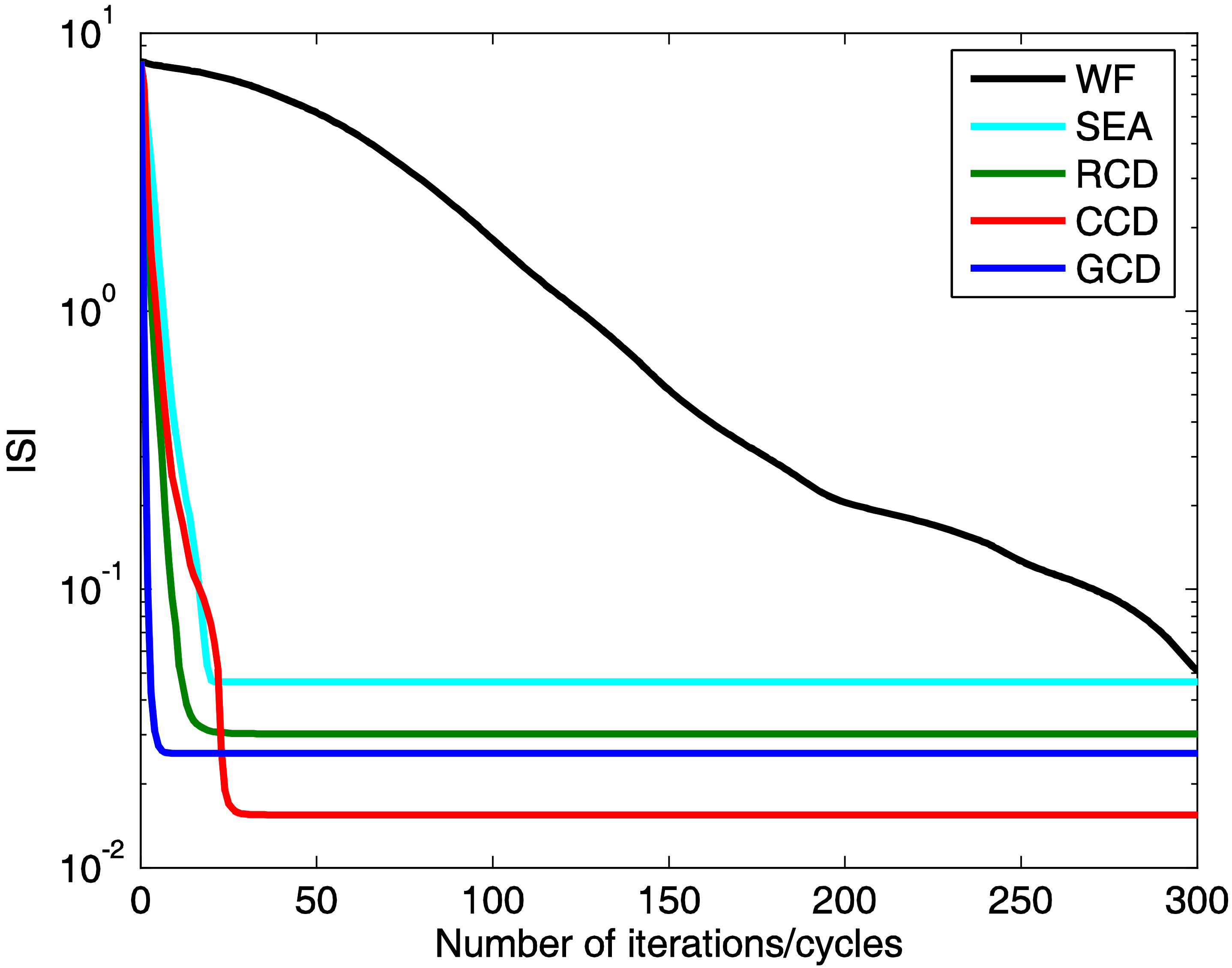

Figure 8: ISI versus number of iterations/cycles

Concluding Remarks

- Making use of CD where only one variable is updated at each iteration, multivariate fourth-order polynomial minimization of phase retrieval is converted to univariate fourth-order polynomial minimization.

- Cyclic, random, and greedy CD selection rules have been considered. All converge faster than WF and GCD has the fastest convergence at the expense of higher computational requirement.

- CCD and RCD has been extended to phase retrieval of sparse signals.

- Application to blind equalization is demonstrated.

List of References

[1] R. Gerchberg and W. Saxton, "A practical algorithm for the determination of phase from image and diffraction plane pictures," *Optik*, vol. 35, pp. 237–246, 1972.

[2] E. J. Candès, X. Li, and M. Soltanolkotabi, "Phase retrieval via Wirtinger flow: Theory and algorithms," *IEEE Trans. Inf. Theory*, vol. 61, no. 4, pp. 1985–2007, Apr. 2015.

[3] E. J. Candès, T. Strohmer, and V. Voroninski, "PhaseLift: Exact and stable signal recovery from magnitude measurements via convex programming," *Commun. Pure Appl. Math.*, vol. 66, no. 8, pp. 1241–1274, 2013.

[4] I. Waldspurger, A. d'Aspremont, and S. Mallat, "Phase recovery, MaxCut and complex semidefinite

programming," *Math. Program., Ser. A*, vol. 149, no. 1, pp. 47–81, Feb. 2015.

[5] Y. Shechtman, Y. C. Eldar, O. Cohen, H. N. Chapman, J. Miao and M. Segev, "Phase retrieval with application to optical imaging," *IEEE Signal Process. Mag*., vol. 32, no. 3, pp. 87–109, May 2015.

[6] O. Shalvi and E. Weinstein, "Super-exponential method for blind deconvolution," *IEEE Trans. Inf. Theory*, vol. 39, no. 2, pp. 504–519, Mar. 1993.