Multiplication-free coordinate descent iteration for ℓ_1-regularized least squares

Nguyen T. Thao*, Dominik Rzepka†
*Dept. of EE, City College of New York, CUNY, New York, USA, tnguyen@ccny.cuny.edu
† Dept. of Measurement and Electronics, AGH Univ. of Science and Tech., Kraków, Poland, drzepka@agh.edu.pl

***Abstract*—We propose a coordinate descent iteration for ℓ_1-regularized least-squares optimization that is free of multiplications. Although a suboptimal version of the ideal coordinate descent algorithm of Li and Osher [1], it contributes to major computational savings for only slight convergence degradations.**

***Index Terms*—least squares, ℓ_1-regularization, basis pursuit denoising, lasso, coordinate descent, relaxation, binary scaling, bit shift, compressed sensing, sparse recovery.**

I. Introduction

Sparse representation has become a fundamental tool for modern machine learning and signal processing [2]. Finding sparse approximate solutions of $\mathrm{A}\mathbf{x} = \mathbf{b}$ (where A is an $M \times N$ matrix) can be performed by minimizing

$$F_{\ell_0}(\mathbf{x}) = \|\mathrm{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda\|\mathbf{x}\|_0. \tag{1}$$

While finding a minimizer of $F_{\ell_0}(\mathbf{x})$ is an NP-complete problem, it is possible to obtain efficiently a good approximation of it. Two main approaches for this are greedy methods (related to matching pursuit [3], [4]) and relaxation methods (e.g. [5], [6], see also survey [2]). The latter rely on an ℓ_1-penalized least squares optimization by minimizing

$$F(\mathbf{x}) = \|\mathrm{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda\|\mathbf{x}\|_1. \tag{2}$$

Greedy methods are fast, but not accurate in finding the correct support of $\mathbf{x}$. With ℓ_1-regularized methods, finding the correct support is guaranteed as long as the matrix A obeys certain conditions, but they tend to be slower. In [1], Li and Osher proposed a relaxation-based coordinate descent (CD) approach to this optimization, where at each iteration, $F(\mathbf{x})$ is minimized along one of the coordinate dimensions of $\mathbf{x}$. CD methods are appealing for their low complexity per iteration. Recently proposed improvements of CD methods are parallelization [7], [8] and inexact iterations [9].

Assuming that the column vectors of A are normalized, we push further the complexity reduction of the CD ℓ_1-regularization by limiting all multiplications to mere binary scalings. This is made possible by not aiming at an ideal minimization of $F(\mathbf{x})$ along any given coordinate dimension, but instead by identifying coordinate displacements that are powers of 2 while still guaranteeing a decrease of $F(\mathbf{x})$. Numerical experiments show that the rate of convergence of our multiplication-free iteration is only slightly degraded compared to the ideal CD minimization. Meanwhile, the absence of multiplications allows for substantial computational savings in hardware (FPGA, ASIC etc.) implementations of sparse recovery methods [10], [11], [12], [13], [14].

D. Rzepka was supported by the Polish National Center of Science under Grant DEC-2017/27/B/ST7/03082.

II. Strategy of coordinate descent

The goal is to approach a minimizer of $F(\mathbf{x})$ with an algorithm of the type

$$\mathbf{x}^{(n+1)} = \mathbf{x}^{(n)} + \alpha_n\, \mathbf{e}_{i_n} \tag{3}$$

where $\mathbf{e}_i$ denotes the ith coordinate vector of $\mathbb{R}^N$, $(i_n)_{n\geq 0}$ is some sequence of $I := \{1, \cdots, N\}$ and $(\alpha_n)_{n\geq 0}$ is some sequence in the set

$$B := \big\{ \pm 2^k : k \in \mathbb{Z} \big\} \cup \{0\}.$$

Our strategy is to take

$$\alpha_n = \delta_{i_n}(\mathbf{x}^{(n)}) \tag{4}$$

where for each $i \in I$, $\delta_i(\mathbf{x})$ is a real function of $\mathbb{R}^N$ satisfying the following conditions:

(a) for all $\mathbf{x} \in \mathbb{R}^N$,

$$F\big(\mathbf{x} + \delta_i(\mathbf{x})\, \mathbf{e}_i\big) \leq F(\mathbf{x}) \tag{5}$$

with a strict inequality when $F(\mathbf{x} + \alpha\mathbf{e}_i)$ is not the minimized at $\alpha = 0$,

(b) for all $\mathbf{x} \in \mathbb{R}^N$, $\delta_i(\mathbf{x}) \in B$,

(c) the computation of $\alpha_n = \delta_{i_n}(\mathbf{x}^{(n)})$ in the iteration (3) can be implemented without any multiplication or division but only with binary scalings (besides additions and elementary binary operations).

III. One-dimensional case

We start with the simple case $N = 1$. The function $F(\mathbf{x})$ is reduced to

$$f(x) := \tfrac{1}{2}\|\mathbf{a}\, x - \mathbf{b}\|_2^2 + \lambda|x| \tag{6}$$

where $\mathbf{a}, \mathbf{b} \in \mathbb{R}^M$ with $\|\mathbf{a}\|_2 = 1$, $\lambda \geq 0$ and $x \in \mathbb{R}$. Our goal is to find a real function $\delta(x)$ satisfying conditions (a) and (b), which here have the simpler form,

(a) for all $x \in \mathbb{R}$,

$$f(x + \delta(x)) \leq f(x) \tag{7}$$

with a strict inequality when $f(x + \alpha)$ is not minimized at $\alpha = 0$.

(b) for all $x \in \mathbb{R}$, $\delta(x) \in B$.

A. Revisiting the shrinkage function

Let $f^+(x)$ and $f^-(x)$ be the two quadratic functions defined by

$$f^{\pm}(x) = \tfrac{1}{2}\|\mathbf{a}\,x - \mathbf{b}\|_2^2 \pm \lambda\, x. \tag{8}$$

Then,

$$f(x) = \left\{ \begin{array}{ll} f^+(x), & x \geq 0 \\ f^-(x), & x \leq 0 \end{array} \right. = \max\big(f^-(x), f^+(x)\big). \tag{9}$$

Let

$$\breve{x} := \underset{x\in\mathbb{R}}{\operatorname{argmin}}\ f(x) \qquad \text{and} \qquad \breve{x}^{\pm} := \underset{x\in\mathbb{R}}{\operatorname{argmin}}\ f^{\pm}(x).$$

Since $\|\mathbf{a}\|_2 = 1$, then

$$f^{\pm}(x) = \tfrac{1}{2}\,x^2 - (\mathbf{a}^\top\mathbf{b} - (\pm\lambda))x + \tfrac{1}{2}\|\mathbf{b}\|_2^2,$$

and hence

$$\breve{x}^{\pm} = \mathbf{a}^\top\mathbf{b} - (\pm\lambda). \tag{10}$$

Note that $\breve{x}^- \geq \breve{x}^+$.

Proposition 3.1:

$$\breve{x} = \left\{ \begin{array}{ll} \breve{x}^+, & \breve{x}^+ > 0 \\ \breve{x}^-, & \breve{x}^- < 0 \\ 0, & \text{otherwise} \end{array} \right. . \tag{11}$$

Proof: If $\breve{x}^+ > 0$, then $f\big(\breve{x}^+\big) = f^+\big(\breve{x}^+\big) \leq f^+(\alpha) \leq f(\alpha)$ for all $\alpha \in \mathbb{R}$, so $\breve{x} = \breve{x}^+$. Similarly, $\breve{x} = \breve{x}^-$ when $\breve{x}^- < 0$. Since $\breve{x}^- \geq \breve{x}^+$, the remaining case is when $\breve{x}^+ \leq 0 \leq \breve{x}^-$. By strict convexity, $f^{\pm}(\alpha)$ is decreasing on $\big(-\infty, \breve{x}^{\pm}\big]$ and increasing on $\big[\breve{x}^{\pm}, \infty\big)$. As $f(\alpha) = f^-(\alpha)$ on $(-\infty, 0]$ which is included in $\big(-\infty, \breve{x}^-\big]$, then $f(\alpha)$ is decreasing on $(-\infty, 0]$. Similarly, $f(\alpha)$ is increasing on $[0, \infty)$. So $\breve{x} = 0$. ■

By injecting (10) into (11), we then retrieve the well-known function expression [15]

$$\breve{x} = \operatorname{shrink}(\mathbf{a}^\top\mathbf{b}, \lambda)$$

where

$$\operatorname{shrink}(\beta, \lambda) := \left\{ \begin{array}{ll} \beta-\lambda, & \beta-\lambda > 0 \\ \beta+\lambda, & \beta+\lambda < 0 \\ 0, & \text{otherwise} \end{array} \right. .$$

B. Generic displacement function $\delta(x)$

It is clear that condition (a) is satisfied when $\delta(x) = \breve{x} - x$. It follows from (11) that

$$\breve{x} - x = \left\{ \begin{array}{ll} \breve{x}^+ - x, & \breve{x}^+ > 0 \\ \breve{x}^- - x, & \breve{x}^- < 0 \\ -x, & \text{otherwise} \end{array} \right. . \tag{12}$$

Consider the more general displacement function

$$\delta(x) := \left\{ \begin{array}{ll} \rho(\breve{x}^+ - x), & x + \rho(\breve{x}^+ - x) > 0 \\ \rho(\breve{x}^- - x), & x + \rho(\breve{x}^- - x) < 0 \\ -\hat{\rho}(x), & \text{otherwise} \end{array} \right. \tag{13}$$

where $\rho(\cdot)$ and $\hat{\rho}(\cdot)$ are some real monotonically increasing functions. Note that $\delta(x) = \breve{x} - x$ when $\rho(x) = \hat{\rho}(x) = x$ for all $x \in \mathbb{R}$. Note also in the general case that there is no conflict between the first two cases of (13) since $\rho(\cdot)$ is monotonically increasing and $\breve{x}^- \geq \breve{x}^+$. Our goal is to find a condition on $\rho(\cdot)$ and $\hat{\rho}(\cdot)$ for condition (a) to be satisfied by $\delta(x)$. The following proposition will be determinant in this search.

Proposition 3.2: Let $g(x)$ be a quadratic function of curvature $c > 0$, minimized at $\breve{x}$. For any $x \in \mathbb{R}$,

$$g\big(x + \mu\,(\breve{x} - x)\big) < g(x) \quad \Leftrightarrow \quad x \neq \breve{x} \text{ and } \mu \in (0, 2).$$

Proof: Explicitly, $g(x) = g(\breve{x}) + c(x - \breve{x})^2$ for all $x \in \mathbb{R}$. So $g\big(x + \mu\,(\breve{x} - x)\big) = g(\breve{x}) + c(1-\mu)^2\,(x - \breve{x})^2$. Then, assuming that $x \neq \breve{x}$, $g\big(x + \mu\,(\breve{x} - x)\big) < g(x)$ is successively equivalent to $(1-\mu)^2 < (1-0)^2$, $|1-\mu| < 1$ and finally to $\mu \in (0, 2)$. ■

Theorem 3.3: Assume that $\rho(\cdot)$ and $\hat{\rho}(\cdot)$ are monotonically increasing such that $\rho(0) = \hat{\rho}(0) = 0$, $\rho(x)/x \in (0, 2)$ and $\hat{\rho}(x)/x \in (0, 1]$ for all $x \neq 0$. Then the function $\delta(x)$ defined by (13) satisfies condition (a).

Proof: This proof is complex due to the multiplicity of cases. In particular, the three cases of (11) and (13) do not match, except when $x = \breve{x}$. We start with this particular value of x and show that $\delta(\breve{x}) = 0$. In the first case of (11), we have $\breve{x} = \breve{x}^+ > 0$. Hence, $\rho(\breve{x}^+ - x) + \breve{x} = \rho(0) + \breve{x} = \breve{x} > 0$. This falls in the first case of (13) which implies that $\delta(\breve{x}) = \rho(0) = 0$. In a similar manner, we obtain $\delta(\breve{x}) = 0$ in the second case of (11). In the third case of (11), we have $\breve{x}^+ \leq 0 \leq \breve{x}^-$ and $\breve{x} = 0$. By assumption on $\rho(\cdot)$, $\rho(x)$ and x have the same sign. So $\rho(\breve{x}^+) \leq 0 \leq \rho(\breve{x}^-)$. This falls in the third case of (13) for $x = \breve{x}$, since $\breve{x} = 0$. Hence $\delta(\breve{x}) = -\hat{\rho}(\breve{x}) = 0$. Therefore, (7) is trivially satisfied at $x = \breve{x}$.

We now consider $x \neq \breve{x}$. From the assumptions on $\rho(\cdot)$ and $\hat{\rho}(\cdot)$, we have for all $u \neq 0$,

$$\exists \mu \in (0, 2), \hat{\mu} \in (0, 1], \quad \rho(u) = \mu\, u \ \text{ and } \ \hat{\rho}(u) = \hat{\mu}\, u. \tag{14}$$

Note that μ and $\hat{\mu}$ depend on u. This is also true at $u = 0$ where one can take $\mu = \hat{\mu} = 1$ for example. Let us label the three cases of (13) 'case 1', 'case 2' and 'case 3', respectively. We are going to show that the following three properties,

$$x + \delta(x) \geq 0, \tag{15}$$

$$\exists \mu \in (0, 2), \quad \delta(x) = \mu\,(\breve{x}^+ - x), \tag{16}$$

$$x \neq \breve{x}^+, \tag{17}$$

are satisfied in case 1 and in case 3 with $x > 0$.

Assume case 1. We have $x + \rho(\breve{x}^+ - x) > 0$ and $\delta(x) = \rho(\breve{x}^+ - x)$. So (15) is straightforward and (16) results directly from (14). If $x = \breve{x}^+$, we obtain successively that $\rho(\breve{x}^+ - x) = 0$, $\breve{x}^+ = x > 0$, $\breve{x} = \breve{x}^+$ according to (11), and finally $x = \breve{x}$. So our assumption that $x \neq \breve{x}$ implies (17).

Assume now case 3 with $x > 0$. From case 3, we have

$$x + \rho(\breve{x}^+ - x) \leq 0 \leq x + \rho(\breve{x}^- - x) \text{ and } \delta(x) = -\hat{\rho}(x). \tag{18}$$

Condition (15) is true since $x + \delta(x) = x - \hat{\rho}(x)$ which has the same sign as x due to (14). Since $x > 0$, $0 < \hat{\rho}(x) \leq x \leq$

$-\rho(\breve{x}^+ - x) = -\mu\,(\breve{x}^+ - x)$ for some $\mu \in (0,2)$ according to (14). Then, there exists $\mu' \in (0,\mu] \subset (0,2)$ such that $\hat{\rho}(x) = -\mu'\,(\breve{x}^+ - x)$. This leads to (16). Since $x > 0$, assuming that $x = \breve{x}^+$ would imply that $\breve{x} = \breve{x}^+$ according to (11), which contradicts the assumption that $x \neq \breve{x}$. This justifies (17).

It follows from (9) and (15) that

$$f(x+\delta(x)) = f^+(x+\delta(x)) \quad \text{and} \quad f^+(x) \leq f(x). \tag{19}$$

It results from Proposition 3.2 with $g = f^+$, (16) and (17) that

$$f^+(x+\delta(x)) < f^+(x). \tag{20}$$

We thus obtain the strict inequality of (7) by combining (19) and (20).

One obtains in case 2 and in case 3 with $x < 0$ equations similar to (15)-(20), where the inequality is reversed in (15), and the superscript $^+$ is replaced by $^-$ in (16)-(20). This also leads to the strict inequality of (7). What remains is case 3 with $x = 0$. From (18), we have $\rho(\breve{x}^+) \leq 0 \leq \rho(\breve{x}^-)$. According to (14), $\mu^+\breve{x}^+ \leq 0 \leq \mu^-\breve{x}^-$ for some $\mu^\pm > 0$. This implies that $\breve{x} = 0$ according to (11), which contradicts the assumption that $x \neq \breve{x}$. So this case is not possible. ∎

C. Binary relaxed displacement

To achieve condition (b), we need to force $\rho(\cdot)$ and $\hat{\rho}(\cdot)$ to have values in B while satisfying the condition of Theorem 3.3. There are various ways to do so. As a simple solution, we propose to take for $\rho(\cdot)$ the odd function such that

$$\rho(u) = \max_{2^k \leq u} 2^k \tag{21}$$

for all $u > 0$, and set $\hat{\rho}(\cdot) = \rho(\cdot)$. By construction, $\rho(0) = 0$ and for any $u \neq 0$, $\rho(u) = \operatorname{sign}(u)\, 2^k$ where k is the position of the most significant 1 in the binary expansion of $|u|$. When $u > 0$, it is clear that $\rho(u) \leq u < 2\rho(u)$. So $u/\rho(u) \in [1,2)$, or equivalently, $\rho(u)/u \in (\frac{1}{2}, 1]$. This remains true for all $u \neq 0$ since $\rho(u)$ has the same sign as u.

IV. N-DIMENSIONAL CASE

We return to the minimization of the general function $F(\mathbf{x})$ of (2). Calling $\mathbf{a}_i$ the ith column vector of A, we assume $\|\mathbf{a}_i\|_2 = 1$ for all $i \in I$.

A. Design of $\delta_i(\mathbf{x})$

Let $\mathbf{x} = (x_1, \cdots, x_N) \in \mathbb{R}^N$ and $i \in I$ be given. We can write

$$\mathrm{A}\mathbf{x} - \mathbf{b} = \mathbf{a}_i\, x_i - \mathbf{b}_i \tag{22}$$

where $\mathbf{b}_i := \mathbf{b} - \sum_{j \in I\setminus\{i\}} \mathbf{a}_j\, x_j$. We can therefore express $F(\mathbf{x})$ from (2) as

$$F(\mathbf{x}) = f_i(x_i) + c_i \tag{23}$$

where for any $x \in \mathbb{R}$,

$$f_i(x) := \tfrac{1}{2}\|\mathbf{a}_i\, x - \mathbf{b}_i\|_2^2 + \lambda|x|$$

and $c_i := \lambda \sum_{j \in I\setminus\{i\}} |x_j|$. Although $\mathbf{b}_i$ and c_i depend on $\mathbf{x}$, note that they do not depend on its ith component. Therefore,

$$F(\mathbf{x} + \alpha\mathbf{e}_i) = f_i(x_i + \alpha) + c_i$$

for any $\alpha \in \mathbb{R}$. By applying Theorem 3.3 to $f_i(x)$, we have

$$f_i(x_i + \alpha_i) \leq f_i(x_i) \tag{24}$$

where

$$\alpha_i := \begin{cases} \rho(\breve{x}_i^+ - x_i), & x_i + \rho(\breve{x}_i^+ - x_i) > 0 \\ \rho(\breve{x}_i^- - x_i), & x_i + \rho(\breve{x}_i^- - x_i) < 0 \\ -\hat{\rho}(x_i), & \text{otherwise} \end{cases} \tag{25}$$

and

$$\breve{x}_i^\pm = \mathbf{a}_i^\top \mathbf{b}_i - (\pm\lambda).$$

The inequality of (24) is moreover strict when $x_i \neq \breve{x}_i := \operatorname{argmin}_x f_i(x)$. It then follows from (23) that

$$F(\mathbf{x} + \alpha_i \mathbf{e}_i) \leq F(\mathbf{x})$$

with a strict inequality when $F(\mathbf{x} + \alpha\mathbf{e}_i)$ is not minimized at $\alpha = 0$. Since $\mathbf{a}_i^\top \mathbf{a}_i = 1$, we have

$$\breve{x}_i^\pm - x_i = \mathbf{a}_i^\top \mathbf{b}_i - (\pm\lambda) - \mathbf{a}_i^\top \mathbf{a}_i\, x_i = r_i - (\pm\lambda) \tag{26}$$

where $r_i := \mathbf{a}_i^\top(\mathbf{b}_i - \mathbf{a}_i x_i) = \mathbf{a}_i^\top(\mathbf{b} - \mathrm{A}\mathbf{x})$ due to (22). Define

$$\mathrm{G} := \mathrm{A}^\top \mathrm{A} \qquad \text{and} \qquad \mathbf{c} := \mathrm{A}^\top \mathbf{b}.$$

Then, r_i is the ith component of the vector

$$\mathbf{r} := \mathrm{R}\mathbf{x} \qquad \text{where} \qquad \mathrm{R}\mathbf{x} := \mathbf{c} - \mathrm{G}\mathbf{x}. \tag{27}$$

Define the function

$$d_i(\mathbf{x}, \mathbf{r}) := \begin{cases} \alpha^+, & x_i + \alpha^+ > 0 \\ \alpha^-, & x_i + \alpha^- < 0 \\ -\hat{\rho}(x_i), & \text{otherwise} \end{cases} \tag{28}$$

where

$$\alpha^\pm := \rho\big(r_i - (\pm\lambda)\big). \tag{29}$$

By injecting (26) into (25), we have $\alpha_i = d_i(\mathbf{x}, \mathbf{r})$ where $\mathbf{r} := \mathrm{R}\mathbf{x}$. Condition (a) is then satisfied by the function

$$\delta_i(\mathbf{x}) := d_i(\mathbf{x}, \mathrm{R}\mathbf{x}). \tag{30}$$

Condition (b) is met by specifically taking $\rho(\cdot)$ defined in Section III-C and $\hat{\rho}(\cdot) = \rho(\cdot)$.

B. Multiplication-free iteration

The function $\delta_i(\mathbf{x})$ does not directly satisfy condition (c) as $\alpha_n = \delta_{i_n}(\mathbf{x}^{(n)}) = d_{i_n}(\mathbf{x}^{(n)}, \mathbf{r}^{(n)})$ where

$$\mathbf{r}^{(n)} := \mathrm{R}\mathbf{x}^{(n)}, \tag{31}$$

which involves matrix multiplications. However, as $\mathbf{x}^{(n)}$ satisfies the recursion (3), $\mathbf{r}^{(n)}$ yields a simple recursive relation. Indeed, after subtracting $\mathbf{b}$ from the members of the relation $\mathrm{G}\mathbf{x}^{(n+1)} = \mathrm{G}\mathbf{x}^{(n)} + \alpha_n \mathbf{g}_{i_n}$ where

$$\mathbf{g}_i := \mathrm{G}\mathbf{e}_i = \mathrm{A}^\top \mathbf{a}_i,$$

(31) and (27) imply that $\mathbf{r}^{(n+1)} = \mathbf{r}^{(n)} - \alpha_n \mathbf{g}_{i_n}$. The iteration of (3) and (4) can then be equivalently implemented by the system

$$\alpha_n = d_{i_n}(\mathbf{x}^{(n)}, \mathbf{r}^{(n)}) \tag{32a}$$

$$\mathbf{x}^{(n+1)} = \mathbf{x}^{(n)} + \alpha_n\, \mathbf{e}_{i_n} \tag{32b}$$

$$\mathbf{r}^{(n+1)} = \mathbf{r}^{(n)} - \alpha_n\, \mathbf{g}_{i_n}. \tag{32c}$$

Starting with $\mathbf{x}^{(0)} = \mathbf{0}$, we have $\mathbf{r}^{(0)} = \mathbf{c}$. Every multiplication in this iteration is a binary scaling.

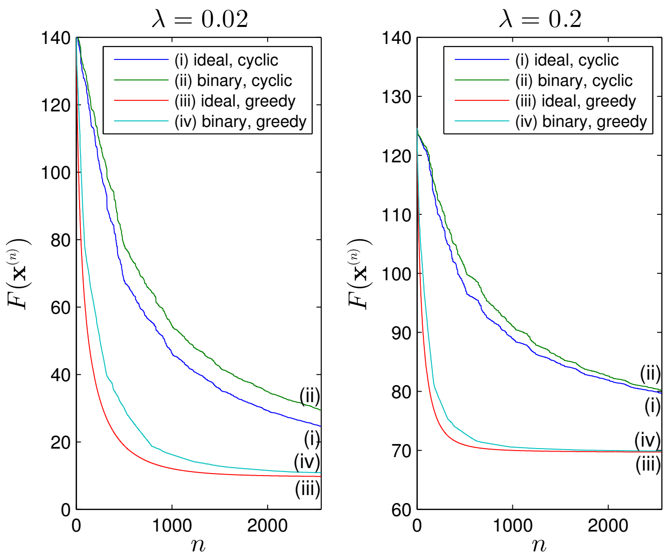

Fig. 1. Numerical results

C. Cyclic and greedy control

Until now, we have not specified the choice of control sequence $(i_n)_{n\geq 0}$. The most basic option is the cyclic control $i_n := (n \bmod N) + 1$. Better results are obtained with control sequences that are adaptive with the evolution of the current estimate $\mathbf{x}^{(n)}$. In the case of ideal coordinate descent, it was proposed in [1] to choose at each iteration n the index $i_n \in I$ that maximizes $\|\mathbf{x}^{(n+1)} - \mathbf{x}^{(n)}\|_2$. With our multiplication-free iteration, this control amounts to taking

$$i_n := \underset{i\in I}{\operatorname{argmax}} \left|d_{i_n}(\mathbf{x}^{(n)}, \mathbf{r}^{(n)})\right|. \tag{33}$$

This control which we will label as "greedy" in this paper does not involve any multiplication.

V. Experiments

As in [1], we performed our numerical tests with a 256×512 matrix A whose entries are randomly and uniformly drawn in $[0, 1]$ before the column vectors are normalized. In Figure 1, we plot the evolution of $F(\mathbf{x}^{(n)})$ with respect to the iteration number n, where $\mathbf{x}^{(n)}$ results from the iteration of the system (32), for various choices of sequence $(i_n)_{n\geq 0}$ and functions $\rho(\cdot)$ and $\hat{\rho}(\cdot)$ in the definition (28) of $d_i(\mathbf{x}, \mathbf{r})$. The cyclic control $i_n := (n \bmod N) + 1$ is used in (i) and (ii), while the greedy control of (33) is used in (iii) and (iv). The functions $\rho(x) = \hat{\rho}(x) = x$ are used in (i) and (ii). They reproduce the ideal CD minimization of [1]. Meanwhile, the binary valued functions $\rho(\cdot)$ and $\hat{\rho}(\cdot)$ defined in Section III-C are used in (iii) and (iv). In this case, all multiplications are reduced to binary scalings. In spite of this outstanding complexity reduction, the consequent degradation in the minimization of $F(\mathbf{x})$ appears to be quite limited (compare (ii) and (iv) with (i) and (iii), respectively).

VI. Block coordinate descent

A block version of (32) can be achieved by iterating the more general system

$$\boldsymbol{\alpha}_n = \mathrm{D}_n(\mathbf{x}^{(n)}, \mathbf{r}^{(n)}) \tag{34a}$$
$$\mathbf{x}^{(n+1)} = \mathbf{x}^{(n)} + \boldsymbol{\alpha}_n \tag{34b}$$
$$\mathbf{r}^{(n+1)} = \mathbf{r}^{(n)} - \mathrm{G}\,\boldsymbol{\alpha}_n \tag{34c}$$

where

$$\mathrm{D}_n(\mathbf{x}, \mathbf{r}) := \textstyle\sum_{i\in I} w_i^{(n)}\, d_i(\mathbf{x}, \mathbf{r})\, \mathbf{e}_i$$

for some non-negative coefficients $w_i^{(n)}$. As (34c) still results from (31), it follows from (30) that

$$\mathbf{x}^{(n+1)} = \mathbf{x}^{(n)} + \textstyle\sum_{i\in I} w_i^{(n)}\, \delta_i(\mathbf{x}^{(n)})\, \mathbf{e}_i.$$

By having $\sum_{i\in I} w_i^{(n)} = 1$ for each $n \geq 0$, one easily shows that $F(\mathbf{x}^{(n+1)}) \leq F(\mathbf{x}^{(n)})$ given that F is convex and (5) is satisfied for all $i \in I$. The system (34) is then made free of multiplication by adding the constraint that $w_i^{(n)}$ is a power of 2 for every i and n.

References

[1] Y. Li and S. Osher, "Coordinate descent optimization for ℓ^1 minimization with application to compressed sensing; a greedy algorithm," *Inverse Problems and Imaging*, vol. 3, no. 3, pp. 487–503, 2009.

[2] Z. Zhang, Y. Xu, J. Yang, X. Li, and D. Zhang, "A survey of sparse representation: algorithms and applications," *IEEE access*, vol. 3, pp. 490–530, 2015.

[3] S. Mallat and Z. Zhang, "Matching pursuit with time-frequency dictionaries," tech. rep., Courant Institute of Mathematical Sciences New York United States, 1993.

[4] D. L. Donoho, Y. Tsaig, I. Drori, and J.-L. Starck, "Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 58, no. 2, pp. 1094–1121, 2012.

[5] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM journal on imaging sciences*, vol. 2, no. 1, pp. 183–202, 2009.

[6] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine learning*, vol. 3, no. 1, pp. 1–122, 2011.

[7] S. J. Wright, "Coordinate descent algorithms," *Mathematical Programming*, vol. 151, no. 1, pp. 3–34, 2015.

[8] P. Richtárik and M. Takáč, "Parallel coordinate descent methods for big data optimization," *Mathematical Programming*, vol. 156, no. 1-2, pp. 433–484, 2016.

[9] R. Tappenden, P. Richtárik, and J. Gondzio, "Inexact coordinate descent: complexity and preconditioning," *Journal of Optimization Theory and Applications*, vol. 170, no. 1, pp. 144–176, 2016.

[10] F. Ren, R. Dorrace, W. Xu, and D. Marković, "A single-precision compressive sensing signal reconstruction engine on FPGAs," in *Field Programmable Logic and Applications (FPL), 2013 23rd International Conference on*, pp. 1–4, IEEE, 2013.

[11] H. D. R. Aniles, "FPGA-based compressed sensing reconstruction of sparse signals," Master's thesis, Instituto Nacional de Astrofísica, Óptica y Electrónica, 2014.

[12] Y. Quan, Y. Li, X. Gao, and M. Xing, "FPGA implementation of real-time compressive sensing with partial fourier dictionary," *International Journal of Antennas and Propagation*, vol. 2016, 2016.

[13] T. Remez, O. Litany, S. Yoseff, H. Haim, and A. Bronstein, "FPGA system for real-time computational extended depth of field imaging using phase aperture coding," *arXiv preprint arXiv:1608.01074*, 2016.

[14] S. Kim, U. Yun, J. Jang, G. Seo, J. Kang, H.-N. Lee, and M. Lee, "Reduced computational complexity orthogonal matching pursuit using a novel partitioned inversion technique for compressive sensing," *Electronics*, vol. 7, no. 9, p. 206, 2018.

[15] W. J. Fu, "Penalized regressions: the bridge versus the lasso," *J. Comput. Graph. Statist.*, vol. 7, no. 3, pp. 397–416, 1998.