TWO VARIANTS OF ALTERNATING DIRECTION METHOD OF MULTIPLIERS WITHOUT CERTAIN INNER ITERATIONS AND THEIR APPLICATION TO IMAGE SUPER-RESOLUTION

Masao Yamagishi, Shunsuke Ono, and Isao Yamada**

Department of Communications and Integrated Systems, Tokyo Institute of Technology, Japan.
E-mail: {myamagi, isao}@sp.ss.titech.ac.jp, ono@net.ss.titech.ac.jp

ABSTRACT

We propose variants of *Alternating Direction Method of Multipliers (ADMM)* employing simplified updates under additional assumptions. ADMM iteratively solves the minimization of the sum of two nonsmooth convex functions. Each iterations of ADMM itself consists of solving a certain convex optimization problem which often requires the use of some iterative solver. Such inner iterations cause slow convergence. Our proposed algorithms avoid some of inner iterations by employing simplified updates. An efficacy of the proposed algorithm is shown in an image super-resolution problem. In this application, the resultant algorithm does not require matrix inversion which causes inner iterations of the original ADMM. A numerical example in the image super-resolution setting demonstrates that our proposed algorithms reduce CPU time to about 70–80 percent of the original ADMM.

***Index Terms*—** minimization methods, iterative methods, image enhancement

1. INTRODUCTION

Convex optimization problems arise in many signal and image processing applications. Recently many researchers have special interest in algorithms for nonsmooth convex optimization because in many situations the nonsmooth convex optimization offers more suitable formulation than the smooth convex optimization.

One candidate of such a desired algorithm is the *Alternating Direction Method of Multipliers (ADMM)* [1, 2, 3]. ADMM solves the following minimization problem:

$$\begin{aligned} &\text{Find } (\boldsymbol{x}^*, \boldsymbol{z}^*) \in \mathcal{S} \\ &\mathcal{S} := \mathop{\arg\min}_{\forall(\boldsymbol{x},\boldsymbol{z})\in\mathbb{R}^N\times\mathbb{R}^M} \{f(\boldsymbol{x}) + g(\boldsymbol{z}) \mid \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{z} = \boldsymbol{c}\}, \end{aligned} \tag{1}$$

where $f\colon \mathbb{R}^N \to (-\infty, \infty]$ and $g\colon \mathbb{R}^M \to (-\infty, \infty]$ are proper (possibly nonsmooth) lower semicontinuous convex functions (see for example [4]), and matrices $\boldsymbol{A} \in \mathbb{R}^{P\times N}, \boldsymbol{B} \in \mathbb{R}^{P\times M}$ and a vector $\boldsymbol{c} \in \mathbb{R}^P$ are given. ADMM generates sequences $(\boldsymbol{x}_k)_{k\geq 0}$ and $(\boldsymbol{z}_k)_{k\geq 0}$, which achieve convergence of the objective function value to the minimum level. The alternating updates of $\boldsymbol{x}_k$ and $\boldsymbol{z}_k$ are defined by solving separately certain minimization problems related to f or g.

An efficacy of ADMM can be seen in the minimization problem of $f(\cdot) + g(\boldsymbol{A}(\cdot))$. That is, the case where $p = M$, $\boldsymbol{B} = -\boldsymbol{I}_M$ ($\boldsymbol{I}_M \in \mathbb{R}^{M\times M}$ is an identity matrix), and $\boldsymbol{c} = \boldsymbol{0}$ is considered. Then the iteration of ADMM essentially requires only the *proximity operator* [5, 6] of g to update $\boldsymbol{z}_k$ while the iterations in the direct use of other conventional algorithms [6, 7, 8, 9, 10, 11, 12] require the proximity operator of $g(\boldsymbol{A}(\cdot))$ that is often hard to compute. The remaining issue of ADMM is difficulty of the update of $\boldsymbol{x}_k$: Unfortunately, the update of $\boldsymbol{x}_k$ often requires iterative solvers on each iteration; these inner iterations cause slow convergence.

In this paper, as candidates of the decisive solution of inner iterations, we propose two variants of ADMM. These variants employ simplified updates of $\boldsymbol{x}_k$ under the following assumptions:

(C1) $f\colon \mathbb{R}^N \to \mathbb{R}$ is a differentiable convex function and its gradient is Lipschitz continuous with Lipschitz constant $\mathcal{L}(f) > 0$, i.e.[1],

$$\|\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{y})\| \leq \mathcal{L}(f)\|\boldsymbol{x} - \boldsymbol{y}\|, \forall \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^N,$$

and $(\mathcal{L}(f)\boldsymbol{I}_N + \rho\boldsymbol{A}^T\boldsymbol{A})^{-1}$ does not require inner iteration,

(C2) The proximity operator of $f\colon \mathbb{R}^N \to (-\infty, \infty]$ is easy to compute and an upper bound of the largest eigenvalue of $\boldsymbol{A}^T\boldsymbol{A}$ is known.

Under the case (C1), the update of $\boldsymbol{x}_k$ is realized by the unique solution of a certain linear equation. Under the case (C2), the update of $\boldsymbol{x}_k$ is realized by the proximity operator of f. These updates do not require inner iterations in many cases even if the update of the original ADMM does.

We consider a single image super-resolution problem as an application of our proposed algorithms. The single image super-resolution is a technique to estimate a high-resolution image from a low-resolution image (see e.g. [13] for single image super-resolution and [14] for an application of ADMM to multi image super-resolution problems). The resultant algorithms have closed form expressions to update $\boldsymbol{x}_k$ while a direct application of ADMM requires a matrix inversion which causes inner iterations. A numerical example in the image super-resolution setting demonstrates that our proposed algorithms work appropriately and reduce CPU time to about 70–80 percent of the original ADMM.

2. ALTERNATING DIRECTION METHOD OF MULTIPLIERS (ADMM)

The Alternating Direction Method of Multipliers (ADMM) approximates the solution of problem (1). The iterates of ADMM are summarized as

$$\boldsymbol{x}_{k+1} \in \mathop{\arg\min}_{\boldsymbol{x}\in\mathbb{R}^N} L_\rho(\boldsymbol{x}, \boldsymbol{z}_k, \boldsymbol{y}_k) \tag{2}$$

$$\boldsymbol{z}_{k+1} \in \mathop{\arg\min}_{\boldsymbol{z}\in\mathbb{R}^M} L_\rho(\boldsymbol{x}_{k+1}, \boldsymbol{z}, \boldsymbol{y}_k) \tag{3}$$

$$\boldsymbol{y}_{k+1} = \boldsymbol{y}_k + \rho(\boldsymbol{A}\boldsymbol{x}_{k+1} + \boldsymbol{B}\boldsymbol{z}_{k+1} - \boldsymbol{c}), \tag{4}$$

*This work was partially supported by SCAT (Support Center for Advanced Telecommunications) and JSPS Grants-in-Aid (B-21300091, 20760252, 09J09539)

[1] In this paper, we denote the standard inner product by $\langle\cdot,\cdot\rangle$ and its induced norm by $\|\cdot\|$.

where L_ρ is the augmented Lagrangian of problem (1), i.e.,

$$L_\rho(\boldsymbol{x},\boldsymbol{z},\boldsymbol{y}) := f(\boldsymbol{x})+g(\boldsymbol{z})+\langle \boldsymbol{y}, \boldsymbol{A}\boldsymbol{x}+\boldsymbol{B}\boldsymbol{z}-\boldsymbol{c}\rangle+\frac{\rho}{2}\|\boldsymbol{A}\boldsymbol{x}+\boldsymbol{B}\boldsymbol{z}-\boldsymbol{c}\|^2.$$

Under the assumption of the existence of a saddle point[2], say $(\boldsymbol{x}^*,\boldsymbol{z}^*,\boldsymbol{y}^*)$, of the unaugmented Lagrangian L_0, ADMM satisfies the following [3]:
$(\boldsymbol{x}^*,\boldsymbol{z}^*)$ is the solution of problem (1), s.t.

(P1) Residual convergence: $\lim_{k\to\infty}\|\boldsymbol{A}\boldsymbol{x}_k+\boldsymbol{B}\boldsymbol{z}_k-\boldsymbol{c}\|=0$ as $k\to\infty$, i.e., the iterates approach feasibility.

(P2) Objective convergence: $\lim_{k\to\infty} f(\boldsymbol{x}_k)+g(\boldsymbol{z}_k)=f(\boldsymbol{x}^*)+g(\boldsymbol{z}^*)$ as $k\to\infty$, i.e., the objective function of the iterates approaches the optimal value.

Two minimization problems (2) and (3) are difficult to solve in general. For example, under a certain mild condition, the calculation of $\boldsymbol{x}_{k+1}$ is equivalent to find a vector satisfying

$$\partial f(\boldsymbol{x}_{k+1})+\boldsymbol{A}^T\boldsymbol{y}_k+\rho\boldsymbol{A}^T(\boldsymbol{A}\boldsymbol{x}_{k+1}+\boldsymbol{B}\boldsymbol{z}_k-\boldsymbol{c})\ni\boldsymbol{0}, \tag{5}$$

where ∂f is the subdifferential[3] of f. This problem is more demanding than the computation of the proximity operator[4] of f: for each $\boldsymbol{z}\in\mathbb{R}^N$, $\mathrm{prox}_{\gamma f}(\boldsymbol{z})$ is characterized by the unique $\boldsymbol{x}$ satisfying

$$\partial f(\boldsymbol{x})+\gamma^{-1}(\boldsymbol{x}-\boldsymbol{z})\ni\boldsymbol{0}.$$

Therefore, application of ADMM is restricted to the case where each solution of (2) and (3) is obtained easily. Otherwise, (2) and (3) are solved by some iterative solvers (such a technique can be seen in [15]).

3. PROPOSED METHODS

3.1. ADM-type algorithm with a simplified update of $\boldsymbol{x}_k$ under (C1)

We introduce a simplified update by employing the minimization of an upper bound of the augmented Lagrangian L_ρ. We adopt a convex upper bound $\hat{L}_\rho$ of L_ρ under the assumption (C1), i.e.

$$L_\rho(\boldsymbol{x},\boldsymbol{z}_k,\boldsymbol{y}_k) \le \hat{L}_\rho(\boldsymbol{x},\boldsymbol{z}_k,\boldsymbol{y}_k),\ \forall \boldsymbol{x}\in\mathbb{R}^N, \tag{6}$$

where[5]

$$\begin{aligned}\hat{L}_\rho(\boldsymbol{x},\boldsymbol{z}_k,\boldsymbol{y}_k) \ :=\ & L_\rho(\boldsymbol{x},\boldsymbol{z}_k,\boldsymbol{y}_k)-f(\boldsymbol{x})+f(\boldsymbol{x}_k)\\ &+\langle\nabla f(\boldsymbol{x}_k),\boldsymbol{x}-\boldsymbol{x}_k\rangle+\frac{\mathcal{L}(f)}{2}\|\boldsymbol{x}-\boldsymbol{x}_k\|^2.\end{aligned}$$

[2] A pair $(\bar{\boldsymbol{x}},\bar{\boldsymbol{z}},\bar{\boldsymbol{y}})$ is a saddle point of the unaugmented Lagrangian L_0 if and only if

$$L_0(\bar{\boldsymbol{x}},\bar{\boldsymbol{z}},\boldsymbol{y})\le L_0(\bar{\boldsymbol{x}},\bar{\boldsymbol{z}},\bar{\boldsymbol{y}})\le L_0(\boldsymbol{x},\boldsymbol{z},\bar{\boldsymbol{y}}),$$

for any $(\boldsymbol{x},\boldsymbol{z},\boldsymbol{y})\in\mathbb{R}^N\times\mathbb{R}^M\times\mathbb{R}^P$.

[3] For a given proper lower semicontinuous convex function $f\colon\mathbb{R}^N\to(\infty,\infty]$, the subdifferential of f is a set-valued function defined by

$$\partial f(\boldsymbol{x}) := \{\boldsymbol{v}\in\mathbb{R}^N \mid f(\boldsymbol{x})+\langle\boldsymbol{y}-\boldsymbol{x},\boldsymbol{v}\rangle\le f(\boldsymbol{y}),\ \forall\boldsymbol{y}\in\mathbb{R}^N\}.$$

[4] For a given proper lower semicontinuous convex function $f\colon\mathbb{R}^N\to(\infty,\infty]$, the proximity operator of f (of level $\gamma>0$) is defined by

$$\mathrm{prox}_{\gamma f}(\boldsymbol{z}) := \operatorname*{arg\,min}_{\boldsymbol{x}\in\mathbb{R}^N}\left(f(\boldsymbol{x})+\frac{1}{2\gamma}\|\boldsymbol{x}-\boldsymbol{z}\|^2\right).$$

[5] The inequality (6) is guaranteed by a well-known upper bound of f under (C1) (see e.g. [16]):

$$f(\boldsymbol{x})\le f(\boldsymbol{y})+\langle\nabla f(\boldsymbol{y}),\boldsymbol{x}-\boldsymbol{y}\rangle+\frac{\mathcal{L}(f)}{2}\|\boldsymbol{x}-\boldsymbol{y}\|^2,\ \forall\boldsymbol{x},\boldsymbol{y}\in\mathbb{R}^N.$$

This leads the following algorithm:

$$\boldsymbol{x}_{k+1}=\operatorname*{arg\,min}_{\boldsymbol{x}\in\mathbb{R}^N}\hat{L}_\rho(\boldsymbol{x},\boldsymbol{z}_{k+1},\boldsymbol{y}_k) \tag{7}$$

$$\boldsymbol{z}_{k+1}\in\operatorname*{arg\,min}_{\boldsymbol{z}\in\mathbb{R}^M}L_\rho(\boldsymbol{x}_{k+1},\boldsymbol{z},\boldsymbol{y}_k) \tag{8}$$

$$\boldsymbol{y}_{k+1}=\boldsymbol{y}_k+\rho(\boldsymbol{A}\boldsymbol{x}_{k+1}+\boldsymbol{B}\boldsymbol{z}_{k+1}-\boldsymbol{c}). \tag{9}$$

Note that $\boldsymbol{x}_{k+1}$ is characterized by the following linear equation:

$$\nabla f(\boldsymbol{x}_k)+\mathcal{L}(f)(\boldsymbol{x}_{k+1}-\boldsymbol{x}_k)+\boldsymbol{A}^T\boldsymbol{y}_k+\rho\boldsymbol{A}^T(\boldsymbol{A}\boldsymbol{x}_{k+1}+\boldsymbol{B}\boldsymbol{z}_k-\boldsymbol{c})=\boldsymbol{0}, \tag{10}$$

which is easier than the nonlinear equation (5) to obtain a solution of (2).

The algorithm (7–9) guarantees the existence of $(u_k)_{k\ge0}$ and $(v_k)_{k\ge0}$ satisfying (11) in the following theorem and therefore satisfies (P1) and (P2).

Theorem 3.1 ([17]) *Suppose that the unaugmented Lagrangian L_0 of problem (1) has a saddle point $(\boldsymbol{x}^*,\boldsymbol{z}^*,\boldsymbol{y}^*)$. Let $(\boldsymbol{x}_k)_{k\ge0}\subset\mathbb{R}^N$ be a sequence which is not necessarily defined by (2). For $(\boldsymbol{z}_0,\boldsymbol{y}_0)\in\mathbb{R}^M\times\mathbb{R}^P$, define the sequences $(\boldsymbol{z}_k)_{k\ge0}$ and $(\boldsymbol{y}_k)_{k\ge0}$ by (3) and (4). If there exists some nonnegative sequences $(u_k)_{k\ge0}$ and $(v_k)_{k\ge1}$ such that*

$$\begin{aligned}f(\boldsymbol{x}_{k+1}) \ \le\ & f(\boldsymbol{x}^*)+u_k-u_{k+1}-v_{k+1}\\ &-\langle\boldsymbol{y}_{k+1},\boldsymbol{A}\boldsymbol{x}_{k+1}-\boldsymbol{A}\boldsymbol{x}^*\rangle\\ &-\rho\langle\boldsymbol{B}\boldsymbol{z}_k-\boldsymbol{B}\boldsymbol{z}_{k+1},\boldsymbol{A}\boldsymbol{x}_{k+1}-\boldsymbol{A}\boldsymbol{x}^*\rangle\end{aligned} \tag{11}$$

for every $k\ge1$, then the residual convergence (P1) and the objective convergence (P2) are guaranteed.

3.2. ADM-type algorithm with a simplified update of $\boldsymbol{x}_k$ under (C2)

We introduce a simplified update by replacing the term $\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{x}_{k+1}$ of (5) by $\lambda\boldsymbol{x}_{k+1}-\boldsymbol{p}_k$; Here, $\boldsymbol{p}_k$ is introduced to guarantee (11) and $\lambda>0$ is any value larger than the largest eigenvalue of $\boldsymbol{A}^T\boldsymbol{A}$ for example $\lambda=\mathrm{tr}(\boldsymbol{A}^T\boldsymbol{A})$. Hence, $\boldsymbol{x}_{k+1}$ is realized by

$$\partial f(\boldsymbol{x}_{k+1})+\boldsymbol{A}^T\boldsymbol{y}_k+\rho(\lambda\boldsymbol{x}_{k+1}-\boldsymbol{p}_k+\boldsymbol{A}^T\boldsymbol{B}\boldsymbol{z}_k-\boldsymbol{A}^T\boldsymbol{c})\ni\boldsymbol{0}.$$

This leads our second proposed algorithm:

$$\boldsymbol{x}_{k+1}=\mathrm{prox}_{(\rho\lambda)^{-1}f}(\lambda^{-1}(\boldsymbol{p}_k-\boldsymbol{A}^T(\boldsymbol{B}\boldsymbol{z}_k-\boldsymbol{c}+\rho^{-1}\boldsymbol{y}_k))) \tag{12}$$

$$\boldsymbol{p}_{k+1}=(\lambda\boldsymbol{I}_N-\boldsymbol{A}^T\boldsymbol{A})\boldsymbol{x}_{k+1} \tag{13}$$

$$\boldsymbol{z}_{k+1}\in\operatorname*{arg\,min}_{\boldsymbol{z}\in\mathbb{R}^M}L_\rho(\boldsymbol{x}_{k+1},\boldsymbol{z},\boldsymbol{y}_k) \tag{14}$$

$$\boldsymbol{y}_{k+1}=\boldsymbol{y}_k+\rho(\boldsymbol{A}\boldsymbol{x}_{k+1}+\boldsymbol{B}\boldsymbol{z}_{k+1}-\boldsymbol{c}) \tag{15}$$

with an arbitrarily chosen $\boldsymbol{p}_0\in\mathbb{R}^N$.

If a saddle point of L_0 exists, the algorithm (12–15) fulfills the condition (11) and satisfies (P1) and (P2) [17].

4. APPLICATION TO SINGLE IMAGE SUPER-RESOLUTION WITH ℓ_1 MINIMIZATION

Single image super-resolution problem is stated as the inversion of the following linear system

$$\boldsymbol{r}=\boldsymbol{D}\boldsymbol{R}\hat{\boldsymbol{x}}$$

where $\hat{\boldsymbol{x}}\in\mathbb{R}^N$ is the unknown high-resolution image with N pixels, $\boldsymbol{r}\in\mathbb{R}^M$ is a target image with $M<<N$ pixels, $\boldsymbol{D}\colon\mathbb{R}^N\to\mathbb{R}^M$

is a down sample operator, and $\boldsymbol{R}\colon \mathbb{R}^N \to \mathbb{R}^N$ is a blur operator. Since $\boldsymbol{DR}$ is degraded, a priori information is required to solve this inverse problem satisfactorily. There are mainly two type problem formulations to exploit the sparsity of high-frequency components of the high-resolution image as a priori information.

The first problem formulation is

$$\min_{\boldsymbol{x}\in\mathbb{R}^N} \frac{1}{2}||\boldsymbol{DRx}-\boldsymbol{r}||^2+\mu||\boldsymbol{Fx}||_1, \tag{16}$$

where $||\cdot||_1$ denotes the ℓ_1 norm, $\mu \geq 0$ is the regularization parameter, and $\boldsymbol{F}:\mathbb{R}^N \to \mathbb{R}^K$ $(K \geq N)$ is a certain tight frame which extracts sparsity of the high-resolution image and satisfies $\boldsymbol{F}^T\boldsymbol{F} = \boldsymbol{I}_N$. We can see that (16) is a special case of (1) under condition (C1) with $f(\boldsymbol{x}) = \frac{1}{2}||\boldsymbol{DRx}-\boldsymbol{r}||^2$, $g(\boldsymbol{z}) = \mu||\boldsymbol{Fz}||_1$, $\boldsymbol{A} = \boldsymbol{I}_N$, $\boldsymbol{B} = -\boldsymbol{I}_N$, and $\boldsymbol{c} = \boldsymbol{0}$.

Then the first proposed algorithm (7–9) specialized to (16) is given as

$$\boldsymbol{x}_{k+1}=\frac{1}{\mathcal{L}(f)+\rho}\left(\mathcal{L}(f)\boldsymbol{x}_k-\boldsymbol{R}^T\boldsymbol{D}^T(\boldsymbol{DRx}_k-\boldsymbol{r})-\boldsymbol{y}_k+\rho\boldsymbol{z}_k\right) \tag{17}$$

$$\boldsymbol{z}_{k+1} = \boldsymbol{F}^T\mathrm{prox}_{\rho^{-1}\mu\|\cdot\|_1}(\boldsymbol{F}(\boldsymbol{x}_{k+1}+\rho^{-1}\boldsymbol{y}_k)) \tag{18}$$

$$\boldsymbol{y}_{k+1} = \boldsymbol{y}_k+\rho(\boldsymbol{x}_{k+1}-\boldsymbol{z}_{k+1}). \tag{19}$$

As you see above, the first proposed algorithm for (16) does not require inner iterations due to no use of any matrix inversion. Note that the proximity operator of $\|\cdot\|_1$ has an closed-form expression with $\mathcal{O}(K)$ complexity. This operator is highly utilized in context of sparsity-aware signal processing (e.g., [6, 9, 10, 18]).

On the other hand, the original ADMM algorithm for (16) is given as

$$\boldsymbol{x}_{k+1} = \left(\rho\boldsymbol{I}_N+\boldsymbol{R}^T\boldsymbol{D}^T\boldsymbol{DR}\right)^{-1}\left(\boldsymbol{R}^T\boldsymbol{D}^T\boldsymbol{r}-\boldsymbol{y}_k+\rho\boldsymbol{z}_k\right) \tag{20}$$

$$\boldsymbol{z}_{k+1} = \boldsymbol{F}^T\mathrm{prox}_{\rho^{-1}\mu\|\cdot\|_1}(\boldsymbol{F}(\boldsymbol{x}_{k+1}+\rho^{-1}\boldsymbol{y}_k))$$

$$\boldsymbol{y}_{k+1} = \boldsymbol{y}_k+\rho(\boldsymbol{x}_{k+1}-\boldsymbol{z}_{k+1}).$$

The calculation of inverse of $\rho\boldsymbol{I}_N+\boldsymbol{R}^T\boldsymbol{D}^T\boldsymbol{DR}$ in (20) causes inner iterations because $\rho\boldsymbol{I}_N+\boldsymbol{R}^T\boldsymbol{D}^T\boldsymbol{DR}$ is not circulant or Toeplitz matrix.

The second problem formulation is

$$\min_{\boldsymbol{x}\in\mathbb{R}^N} \iota_C(\boldsymbol{DRx})+||\boldsymbol{Fx}||_1, \tag{21}$$

where

$$\iota_C(\boldsymbol{x}) := \begin{cases} 0, & \text{if } \boldsymbol{x}\in C, \\ \infty, & \text{otherwise,} \end{cases}$$

denotes the indicator function of the nonempty closed convex set

$$C := \{\boldsymbol{x}\in\mathbb{R}^M \mid ||\boldsymbol{x}-\boldsymbol{r}|| \leq \varepsilon\}.$$

Problem (21) is also special case of (1) with $f(\boldsymbol{x}) = ||\boldsymbol{Fx}||_1$, $g(\boldsymbol{z}) = \iota_C(\boldsymbol{z})$, $\boldsymbol{A} = \boldsymbol{DR}$, $\boldsymbol{B} = -\boldsymbol{I}_M$, and $\boldsymbol{c} = \boldsymbol{0}$. Note that relationships of f, g, and $\boldsymbol{A}$ are different from the first problem formulation. In this case, we can assume that condition (C2) holds, i.e., the proximity operator of f is computed efficiently and an upper bound λ of the largest eigenvalue of $\boldsymbol{A}^T\boldsymbol{A} = \boldsymbol{R}^T\boldsymbol{D}^T\boldsymbol{DR}$ is known (e.g. $\lambda = \mathrm{tr}(\boldsymbol{A}^T\boldsymbol{A})$).

Table 1. Comparison of CPU time (sec) of proposed algorithms and the original ADMM. Proposed algorithms reduce CPU time to about 70–80 percent of the original ADMM.

	1st formulation (16)		2nd formulation (21)	
	1st algo. (17–19)	ADMM	2nd algo. (22–25)	ADMM
512×512	172	248	184	253
1024×1024	887	1144	920	1207

Then the second proposed algorithm (12–15) can be applicable to (21) as

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}^T\mathrm{prox}_{(\rho\lambda)^{-1}\|\cdot\|_1}(\boldsymbol{F}(\lambda^{-1}(\boldsymbol{p}_k+\boldsymbol{R}^T\boldsymbol{D}^T(\boldsymbol{z}_k-\rho^{-1}\boldsymbol{y}_k)))) \tag{22}$$

$$\boldsymbol{p}_{k+1} = (\lambda\boldsymbol{I}_N-\boldsymbol{R}^T\boldsymbol{D}^T\boldsymbol{DR})\boldsymbol{x}_{k+1} \tag{23}$$

$$\boldsymbol{z}_{k+1} = \mathrm{prox}_{\rho^{-1}\iota_C}(\boldsymbol{x}_{k+1}+\rho^{-1}\boldsymbol{y}_k) \tag{24}$$

$$\boldsymbol{y}_{k+1} = \boldsymbol{y}_k+\rho(\boldsymbol{DRx}_{k+1}-\boldsymbol{z}_{k+1}). \tag{25}$$

Similarly as the first proposed algorithm, the second proposed algorithm (22–25) does not require matrix inversion while the original ADMM algorithm for (21) does.

5. NUMERICAL EXPERIMENTS

We show performance of the proposed algorithms through image super-resolution described in Section 4. We use 'Lena' image (256×256 [pixels]; thus $N = 65,536$) for a target image $\boldsymbol{r}$ shown in Fig. 1(a), and we consider to make 512×512, and 1024×1024 high-resolution images. In this setting, $\boldsymbol{D}$ is the down sample operator by 1 pixel per 4 and 16 pixels respectively, and $\boldsymbol{R} = \boldsymbol{P}^H\boldsymbol{HP}$ is the blur operator where $\boldsymbol{P}$ is the discrete Fourier transform matrix (and its complex conjugate transpose $\boldsymbol{P}^H$) and $\boldsymbol{H}$ is a diagonal matrix determined by 2×2 and 4×4 uniform blur kernels respectively. We employ a shift invariant redundant Haar wavelet transform with four levels as $\boldsymbol{F}$. Parameters are set as $\mu = 0.003$, $\rho = 10$, $\lambda = 1$, and $\varepsilon = 0$. For the all algorithms, iteration number is fixed as 100.

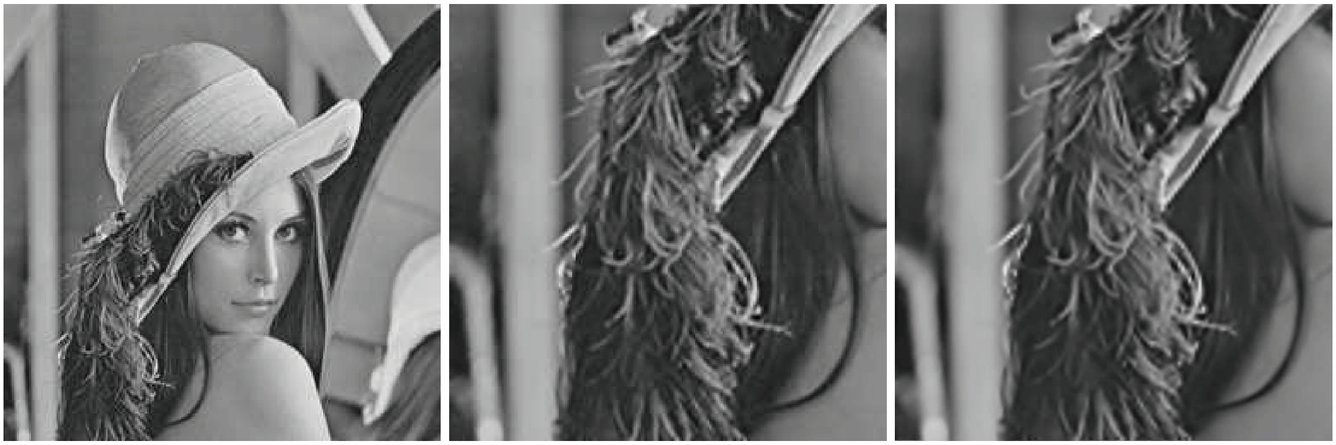
Fig. 1(b)-(e) show resultant images. The proposed algorithms work appropriately for interpolating from 256×256 to 512×512, and 1024×1024.

Table 1 shows a comparison of CPU time measured on a desktop computer equipped with an Intel Core i7 2.8-GHz processor and 8 GB of RAM. Here, ADMM approximates matrix inversion of the update of $\boldsymbol{x}_{k+1}$ by the conjugate gradient method[6] with accuracy 10^{-20}. In this case, proposed algorithms overcome the original ADMM; proposed algorithms reduce CPU time to about 70–80 percent of ADMM.

6. CONCLUSION

We have proposed two variants of ADMM employing simplified updates of $\boldsymbol{x}_k$ under some additional assumptions. Each update often avoids inner iterations for the update of $\boldsymbol{x}_k$. Hence the proposed algorithms are applicable to many problems which cause inner iterations in the original ADMM. The efficacy have been examined in a single image super resolution setting as an example. A numerical example have demonstrated that proposed algorithms reduce CPU time to about 70–80 percent of the original ADMM.

[6] For a given invertible matrix $A \in \mathbb{R}^{N\times N}$ and $\boldsymbol{b} \in \mathbb{R}^N$, we approximate $A^{-1}\boldsymbol{b}$ by $\bar{\boldsymbol{x}} \in \mathbb{R}^N$ such that $\frac{\|A\bar{\boldsymbol{x}}-\boldsymbol{b}\|}{\|\boldsymbol{b}\|} \leq \delta$ with user-defined accuracy $\delta > 0$. $\bar{\boldsymbol{x}}$ is obtained by the use of iterative solvers. This technique can be seen in [15].



(a) Target image, 256×256 [pixels] (b) 1st algorithm (17–19), 512×512. (c) 1st algorithm (17–19), 1024×1024.

(a') Target image (portion) (d) 2nd algorithm (22–25), 512×512. (e) 2nd algorithm (22–25), 1024×1024.

Fig. 1. Portions of image super-resolution results on the test image 'Lena'.

7. REFERENCES

[1] R. Glowinski and A. Marrocco, "Sur l'approximation, par elements finis d'ordre un, et la resolution, par penalisation-dualité, d'une classe de problems de Dirichlet nonlineares," *Revue Francaise d'Automatique, Informatique, et Recherche Opérationelle*, vol. 9, pp. 41–76, 1975.

[2] D. Gabay and B. Mercier, "A dual algorithm for the solution of nonlinear variational problems via finite element approximations," *Computers and Mathematics with Applications*, vol. 2, pp. 17–40, 1976.

[3] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "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.

[4] J. B. Hiriart-Urruty and C. Lemarechal, *Convex analysis and minimization algorithms*, vol. 1 and 2, Springer-Verlag, 1993.

[5] J. J. Moreau, "Fonctions convexes duales et points proximaux dans un espace hilbertien," *C. R. Acad. Sci. Paris Sér. A Math.*, vol. 255, pp. 2897–2899, 1962.

[6] P. L. Combettes and V. R. Wajs, "Signal recovery by proximal forward-backward splitting," *Multiscale Modeling and Simulation*, vol. 4, no. 4, pp. 1168–1200, 2005.

[7] B. Mercier, *Lectures on topics in finite element solution of elliptic problems*, vol. 63 of *Tata Institute of Fundamental Research Lectures on Mathematics and Physics*, Tata Institute of Fundamental Research, Bombay, 1979, With notes by G. Vijayasundaram.

[8] G. B. Passty, "Ergodic convergence to a zero of the sum of monotone operators in Hilbert space," *J. Math. Anal. Appl.*, vol. 72, pp. 383–290, 1979.

[9] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.

[10] M. Yamagishi and I. Yamada, "Over-relaxation of the fast iterative shrinkage-thresholding algorithm with variable stepsize," *Inverse Problems*, vol. 27, 2011.

[11] J. Douglas and H. H. Rachford, "On the numerical solution of heat conduction problems in two or three space variables," *Trans. Amer. Math. Soc.*, vol. 82, pp. 421–439, 1956.

[12] H. H. Bauschke and P. L. Combettes, "A Dykstra-like algorithm for two monotone operators," *Pacific J. Optim*, vol. 4, pp. 383–391, 2008.

[13] D. Glasner, S. Bagon, and M. Irani, "Super-resolution from a single image," in *Proc. of International Conference on Computer Vision*, 2009.

[14] L. L. Huang, L. Xiao, and Z. H. Wei, "Efficient and effective total variation image super-resolution: A preconditioned operator splitting approach," *Mathematical Problems in Engineering*, 2011.

[15] T. Goldstein and S. Osher, "The split bregman method for l1-regularized problems," *SIAM J. on Imaging Sciences*, vol. 2, no. 2, pp. 323–343, 2009.

[16] Y. E. Nesterov, *Introductory lectures on convex optimization: a basic course*, vol. 87 of *Applied Optimization Series*, Kluwer Academic Publishers, 2003.

[17] M. Yamagishi, S. Ono, and I. Yamada, "Variants of alternative direction method of multipliers without certain inner iterations," *in preparation*, 2012.

[18] D. Donoho and I. Johnstone, "Ideal spatial adaptation via wavelet shrinkage," *Biometrika*, vol. 81, pp. 425–455, 1994.