

An Alternating Direction Nonmonotone Approximate Newton Algorithm for Inverse Problems

Zhuhan Zhang, Zhensheng Yu, Xinyue Gan

University of Shanghai for Science and Technology, Shanghai, China
Email: 419797498@qq.com, zhsh-yu@163.com, 1335892166@qq.com

How to cite this paper: Zhang, Z.H., Yu, Z.S. and Gan, X.Y. (2016) An Alternating Direction Nonmonotone Approximate Newton Algorithm for Inverse Problems. *Journal of Applied Mathematics and Physics*, 4, 2069-2078.

<http://dx.doi.org/10.4236/jamp.2016.411206>

Received: October 28, 2016

Accepted: November 27, 2016

Published: November 30, 2016

Copyright © 2016 by authors and Scientific Research Publishing Inc. This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

In this paper, an alternating direction nonmonotone approximate Newton algorithm (ADNAN) based on nonmonotone line search is developed for solving inverse problems. It is shown that ADNAN converges to a solution of the inverse problems and numerical results provide the effectiveness of the proposed algorithm.

Keywords

Nonmonotone Line Search, Alternating Direction Method, Bound-Constraints, Newton Method

1. Introduction

We consider inverse problems that can be expressed in the form

$$\min \frac{1}{2} \|Ax - b\|^2 + \phi(Bx), \quad (1)$$

where $A \in R^{m \times n}$, $B \in R^{m \times n}$, $\phi: R^m \rightarrow (-\infty, \infty)$ is convex, and $b \in R^m$. The emphasis of our work is on problems where A and B have a specific structure. It can be applied to many applications, especially in machine learning [1] [2], image reconstruction [3] [4] or model reduction [5]. We assume that the functions in (1) are strictly convex, so both problems has an unique solution x^* .

In Hong-Chao Zhang's paper [6], he uses the Alternating Direction Approximate Newton method (ADAN) based on Alternating Direction Method (ADMM) which originally in [7] to solve (1). He employs the BB approximation to increase the iterations. In many applications, the optimization problems in ADMM are either easily resolvable, since ADMM iterations can be performed at a low computational cost. Besides, com-

bine different Newton-based methods with ADMM have become a trend, see [6] [8] [9], since those methods may achieve the high convergent speed.

In alternating direction nonmonotone approximate Newton (ADNAN) algorithm developed in this paper, we adopt the nonmonotone line search to replace the traditional Armijo line search in ADAN, because the nonmonotone schemes can improve the likelihood of finding a global optimum and improve convergence speed in cases where a monotone scheme is forced to creep along the bottom of a narrow curved valley in [10].

In the latter context, the first subproblem is to solve the unconstrained minimization problems with Alternating Direction Nonmonotone Approximate Newton algorithm. The purpose is to accelerate the speed of convergence, and then to project or the scale the unconstrained minimizer into the box $\{w \in R^m : l \leq w \leq u\}$, The second subproblem is a bound-constrained optimization problem.

The rest of the paper is organized as follows. In Section 2, we give a review of the alternating direction approximate Newton method. In Section 3, we introduce the new algorithm ADNAN. In Section 4, we introduce the gradient-based algorithm of the second subproblem. A preliminary convergence analysis for ADNAN and gradient-based algorithm (GRAD) is given in Section 5. Numerical results presented in Section 6 explain the effectiveness of ADNAN and GRAD.

2. Review of Alternating Direction Approximate Newton Algorithm

In this section, we briefly review the well-known Alternating Direction Approximate Newton (ADAN) method which has been studied in the areas of convex programming and image reconstruction see [4] [6] and references therein.

We introduce a new variable w to obtain the split formulation of (1):

$$\min_{x,w} \frac{1}{2} \|Ax - b\|^2 + \phi(w) \quad \text{s.t.} \quad w = Bx, x \in R^n, w \in R^m \tag{2}$$

The augmented Lagrangian function associated with (2) is

$$L(x, w, \lambda) = \frac{1}{2} \|Ax - b\|^2 + \phi(w) + \lambda(Bx - w) + \frac{\beta}{2} \|Bx - w\|^2 \tag{3}$$

where $\beta > 0$ is the penalty parameter, $b \in R^m$ is a Lagrangian multiplier associated with the constraint $w = Bx$. In ADMM, each iteration minimizes over x holding w fixed, minimizes over w holding x fixed, and updates an estimate for the multiplier b . More specifically, if λ^k is the current approximation to the multiplier, then ADMM [10] [11] applied to the split formulation (3) is given by the iteration:

$$x^{k+1} = \arg \min_x L(x, w^k, \lambda^k) \tag{4}$$

$$w^{k+1} = \arg \min_w L(x^{k+1}, w, \lambda^k) \tag{5}$$

$$\lambda^{k+1} = \lambda^k + \beta(w^k - Bx^k) \tag{6}$$

And (4) can be written as follows:

$$x^{k+1} = \arg \min_x f(x)$$

$$f(x) = \frac{1}{2} \|Ax - b\|^2 + \frac{\beta}{2} \|Bx - w^k + \beta^{-1} \lambda^k\|^2 \tag{7}$$

For any Hermitian matrix $Q \in \mathbb{R}^{N \times N}$, we define $\|x\|_Q^2 = \langle x, Qx \rangle$, if Q is a positive definite matrix, then $\|\bullet\|_Q$ is a norm. The proximal version of (4) is

$$x^{k+1} = \arg \min_x f(x)$$

$$f(x) = \frac{1}{2} \|Ax - b\|^2 + \frac{\beta}{2} \|Bx - w^k + \beta^{-1} \lambda^k\|^2 + \|x - x^k\|_Q \tag{8}$$

In ADAN, the choice $Q = \delta I - A^T A$ will cancel the $\|Ax\|^2$ term in this iteration while $\|Bx\|^2$ is retained. We replace δ by δ_k , where $\delta_k I$ is a Barzilai-Borwein (BB) [8] [12] approximation to $A^T A$. We can observe the fast convergence of BB approximation in the experiments of Raydan and Svaiter [13]. Moreover, δ_k is strictly smaller than the largest eigenvalue of $A^T A$, and $Q = \delta_k I - A^T A$ is indefinite, so the new convergence analysis is needed. As a result, the updated version for x given in (4) can be expressed as follows:

$$x^{k+1} = \arg \min_x f(x) = x^k - (A^T A + \beta B^T B)^{-1} \nabla f_k \tag{9}$$

$$\nabla f_k := A^T (Ax^k - b) + \beta B^T (Bx^k - w^k + \beta^{-1} \lambda^k) \tag{10}$$

Here, $(\cdot)^{-1}$ is the generalized inverse, $A^T A + \beta B^T B$ is the Hessian of the objective f , and ∇f is the gradient of f at x^k , $g_k = \nabla f$. The formula for x^{k+1} in (2) is exactly the same formula that we would have gotten if we performed a single iteration of Newton's method on the equation $\nabla f(x) = 0$ with starting guess x^k . We employ the BB approximation $A^T A \approx \delta_k I$, [14] where

$$\begin{aligned} \delta_k &= \arg \min \left\{ \left\| A(x^k - x^{k-1}) - \delta(x^k - x^{k-1}) \right\|^2 : \delta \geq \delta_{\min} \right\} \\ &= \max \left\{ \delta_{\min}, \frac{\|A(x^k - x^{k-1})\|^2}{\|x^k - x^{k-1}\|^2} \right\} \end{aligned}$$

and $\delta_{\min} > 0$ is a positive lower bound for δ_k . Hence, the Hessian is approximated by $\delta_k I + \beta B^T B$. Since a Fourier transform can be inverted in $O(N \log N)$ flops. The inversion of $\delta_k I + \beta B^T B$ can be accomplished relatively quickly. After replacing $A^T A$ by $\delta_k I$, the iteration becomes $x^{k+1} = x^k + d_k$ where $d_k = -(\delta_k I + \beta B^T B)^{-1} \nabla f_k$

Note that by substituting $Q = \delta_k I - A^T A$ in (2) and solving for the minimizer, we would get exactly the same formula for the minimizer as that given in (5). When the search direction is determined suitable step size α_k along this direction should be found to determine the next iterative point such as $x^{k+1} = x^k + \alpha_k d_k$.

The inner product between d_k and the objective gradient at x_k is

$$\langle \nabla f_k, d_k \rangle = -\left(\delta_k \|d_k\|^2 + \beta \|Bd_k\|^2 \right).$$

It follows that d_k is a descent direction. Since f is a quadratic, the Taylor expansion of $f(x^{k+1})$ around x^k is as follows:

$$f(x^{k+1}) = f(x^k) + \langle \nabla f, x^{k+1} - x^k \rangle + \frac{1}{2} \|x^{k+1} - x^k\|_H^2$$

where $H = A^T A + \beta B^T B$

Algorithm 1. Nonmonotone Line search

In this section, we adopt a nonmonotone line search method [9]. The step size α_k is chosen in an ordinary Armijo line search which could not admit the more faster speed in unconstrained problems [12]. In contrast, nonmonotone schemes can not only improve the likelihood of finding a global optimum but also improve convergence speed.

Initialization: Choose starting guess x_0 , and parameters $0 \leq \eta_{\min} \leq \eta_{\max} \leq 1$, $0 < m < \sigma < 1 < \rho$, and $u > 0$. Set $C_0 = f(x_0)$, $Q_0 = 1$, and $k = 0$.

Convergence test: If $\|\nabla f(x_k)\|$ sufficiently small, then stop.

line search update: set $x_{k+1} = x_k + \alpha_k d_k, d_k = 0$, where α_k satisfies either the (non-monotone) Wolfe conditions:

$$f(x_k + \alpha_k d_k) \leq C_k + m\alpha_k \nabla f(x_k) d_k, \tag{11}$$

$$\nabla f(x_k + \alpha_k d_k) d_k \geq \sigma \nabla f(x_k) d_k \tag{12}$$

or the (nonmonotone) Armijo conditions: $\alpha_k = \bar{\alpha} \rho^{h_k}$, where $\bar{\alpha}_k > 0$ is the trial step, and h_k is the largest integer such that (11) holds and $\alpha_k \leq \mu$.

Cost update: Choose $\eta_k \in [\eta_{\min}, \eta_{\max}]$, and update

$$Q_{k+1} = \eta_k Q_k + 1, C_{k+1} = (\eta_k Q_k C_k + f(x_{k+1})) / Q_{k+1} \tag{13}$$

Observe that C_{k+1} is a convex combination of C_k and $f(x_{k+1})$. Since $C_0 = f(x_0)$ it follows that C_k is a convex combination of the function values $f(x_0), \dots, f(x_k)$. The choice of η_k controls the degree of nonmonotonicity. If $\eta_k = 0$ for each k , then the line search is the usual monotone Wolfe or Armijo line search. If $\eta_k = 1$ for each k , then $C_k = A_k$ where $A_k = \frac{1}{k+1} \sum_{i=0}^k f_i, f_i = f(x_i)$, is the average function value. The scheme with $C_k = A_k$ was proposed by Yu-Hong Dai [15]. As η_k approaches 0, the line search closely approximates the usual monotone line search, and as η_k approaches 1, the scheme becomes more nonmonotone, treating all the previous function values with equal weight when we compute the average cost value C_k .

Lemma 2.1 If $\nabla f(x_k) d_k \leq 0$ for each k , then for the iterates generated by the nonmonotone line search algorithm, we have $f_k \leq C_k \leq A_k$ for each k . Moreover, if $\nabla f(x_k) d_k < 0$ and $f(x)$ is bounded from below, then there exists α_k satisfying either the Wolfe or Armijo conditions of the line search update.

3. Alternating Direction Nonmonotone Approximate Newton Algorithm

In Algorithm 1, we could get the x at each iteration which can be combined with Algorithm 2. Then, we use the Algorithm 2 to solve the first subproblem in this paper which

is an unconstrained minimization problem with ADNAN, then to project or the scale the unconstrained minimizer into the box

$$\min_{l \leq w \leq u} \frac{1}{2} \|Ax - b\|^2 + \phi(Bx), x \in R^n \tag{14}$$

the iteration is as follows:

$$x^{k+1} = \arg \min_x L(x, w^k, \lambda^k) \tag{15}$$

$$w^{k+1} = \arg \min_{l \leq w \leq u} L(x^{k+1}, w, \lambda^k) \tag{16}$$

$$\lambda^{k+1} = \lambda^k + \beta(w^k - Bx^k) \tag{17}$$

Later we give the existence and uniqueness result for (1).

The solution w^k to (5) has the closed-form means

$$w^k = \arg \min_{l \leq w \leq u} L(x^k, w, b^k) = P\left(Bx^k - \frac{\lambda^k}{\beta}\right)$$

with P being the projection map onto the box $[l, u]$.

Algorithm 2. Alternating Direction Nonmonotone Approximate Newton algorithm.

Parameter: $0.5 < \gamma < 1 < \tau, \rho > 0, 0 < \delta_{\min} < \delta_0$, Initialize $k = 1$.

Step 1: If $\|\nabla f(x_k)\|$ sufficiently small, then set $x^{k+1} = x^k, d^k = 0, \delta_k = \delta_{k-1}$, and branch to Step 4.

$$\delta_k = \max \left\{ \delta_{\min}, \frac{\|A(x^k - x^{k-1})\|^2}{\|x^k - x^{k-1}\|^2} \right\}, d_k = -(\delta_k I + \beta B^T B)^{-1} \nabla f_k.$$

Step 2: If $\delta_k \alpha_k > \delta_{k-1} \alpha_k, \delta_k > \max\{\delta_{\min}, \delta_{k-1}\}$ then $\delta_{\min} = \tau \delta_{\min}$.

Step 3: If α_k accomplish the Wolfe conditions, then $\alpha_k := \tau \alpha_{\max}$.

Step 4: Update x which generated from Algorithm 1.

Step 5: $w^{k+1} = P\left(Bx^k - \frac{\lambda^k}{\beta}\right).$

Step 6: $b^{k+1} = b^k + \beta(Bx^{k+1} - w^{k+1}).$

Step 7: If a stopping criterion is satisfied, terminate the algorithm, Otherwise $k = k + 1$ and go to Step 1.

Lemma 3.1: we show some criteria that are only satisfied a finite number of times, so δ_{\min} converge to positive limits. An upper bound for δ_{\min} is the following:

Uniformly in k , we have $\delta_{\min,k} \leq \delta_k \leq \max\{\bar{\delta}, \tau \|A\|\}$ where $\delta_{\min,k}$ is the value of δ_{\min} at the start of iteration k and $\bar{\delta} = \delta_{\min,1}$ is the starting δ_{\min} in ADNAN.

Lemma 3.2: If $Ad_k = Bd_k = 0$, then x^k minimizes $f(x_k)$.

4. Algorithm 3: Gradient-Based Algorithm (GRAD)

Next, we consider the second subproblem which is about bound-constrained optimization problem as

$$\min_{l \leq x \leq u} \frac{1}{2} \|Ax - b\|^2 + \phi(Bx), w \in R^n \tag{18}$$

And the iteration is similar with (4) (5) (6) as follows:

$$x^{k+1} = \arg \min_{l \leq x \leq u} L(x, w^k, \lambda^k) \tag{19}$$

$$w^{k+1} = \arg \min_w L(x^{k+1}, w, \lambda^k) \tag{20}$$

$$\lambda^{k+1} = \lambda^k + \beta(w^k - Bx^k) \tag{21}$$

Compute the solution x^k from (19), $x^k = B^{-1}(\lambda^k + P^{-1}w^k)$

Compute the solution w^k from (20), $(A^T A + B^T B)w^k = A^T b + \beta w^{k-1} + \lambda^k$

Set $\lambda^{k+1} = \lambda^k + \beta(w^k - Bx^k)$

5. Convergence Analysis

In this section, we show the convergence of proposed algorithms. Obviously, the proofs of the two algorithms are almost the same, and we only prove the convergence of algorithm 2.

Lemma 3.1: Let L be the function in (3). The vector $(x^*, w^*) \in R^n \times [l, u]$ solves (2) if and only if there exists $\lambda^* \in R^n$ such that (x^*, w^*, λ^*) solves $L(x^*, w^*, \lambda) \leq L(x^*, w^*, \lambda^*) \leq L(x, w, \lambda^*), \forall (x, w, \lambda) \in R^n \times [l, u] \times R^n$.

Lemma 3.2: Let L be the function in (3), (x^*, w^*, λ^*) be a saddle-point of L , Then the sequence (x^k, w^k) satisfies $\lim_{k \rightarrow \infty} (x^k, w^k) = (x^*, w^*)$.

Theorem 3.1: Let (x_k, w_k) be the sequence of iterates produced by the algorithm 2. then $\lim_{k \rightarrow \infty} x_k = x^*$, $\lim_{k \rightarrow \infty} w_k = w^*$ and (x^*, w^*) is the optimal point for problem (14)

Proof From Lemma 3.1, 3.2, we obtain that

$$\lim \|w^{k+1} - w^k\| = 0, \lim \|w^k - x^k\| = 0 \tag{22}$$

Since we have a unique minimizer in $[l, u]$, so we have $\lim_{k \rightarrow \infty} x^k = x^*$, Then, (22) gives $\lim_{k \rightarrow \infty} w_k = w^*$ which completes the proof.

6. Numerical Experiments

6.1. Parameter Settings

In Algorithm 2, the parameters β , the penalty in the augmented Lagrangian (3), are common to these two algorithms, ADAN and ADNAN. Besides β has a vital impact on convergence speed. We choose $\beta = 10^{-4}$ based on the results from W. Hager [6]. The choice $\tau = 1.2$ is a compromise between speed and stability, $\delta_{\min} = 0.001$ is large enough to ensure invertibility.

The search directions were generated by the L-BFGS method developed by Nocedal in [16] and Liu and Nocedal in [1]. We choose the step size α_k to satisfy the Wolfe conditions with $m = 0.09$ and $\sigma = 0.9$. In addition we employ a fixed value $\eta_k = 0.47$ which could get the reasonable results. To obtain a good estimate for the optimal objec-

tive in (1), we ran them for 100,000 iterations. The optimal objective values for the three data sets were

$$\Psi = 0.1275$$

$$\Psi = 0.2456$$

$$\Psi = 0.9627$$

In addition, we timed how long it took ADNAN to reduce the objective error to within 1% of the optimal objective value. The algorithms are coded in MATLAB, version 2011b, and run on a Dell version 4510U with a 2.8 GHz Intel i7 processor.

In Algorithm 3, a 256-by-256 gray-scale image was considered, which is compared to the experiment by J. Zhang [8]. The dimensions of the inverse problems are $m = n = 65536$ and the constraints are $l = -\infty$ and $u = (255, \dots, 255)^T$. The experiments on image deblurring problems show that GRAD algorithm is also effective in terms of quality of the image resolution.

6.2. Experiments Results

This section compares the performance of the ADNAN to ADAN. The main difference between the ADNAN algorithm and the ADAN algorithm is the computation of x^{k+1} . In ADAN $x^{k+1} = x^k + \alpha_k d_k$ where α_k is the step size. In ADNAN, x generated from Algorithm 1, if the convergence condition in ADAN is satisfied, then the update $x^{k+1} = x^k + \alpha_k d_k$ could be performed. Here δ_k is the same choice for them. Hence, there seems to be a significant benefit from using a value for δ_k smaller than the largest eigenvalue of $A^T A$.

The initial guess for x^1 , w^1 and λ^1 was zero for two algorithms. **Figures 1-3** show the objective values and objective error as a function of CPU time. Moreover, we give the comparison of objective values and objective error versus CPU time/s for different Ψ conditions. It is observed that ADNAN is slightly stable than ADAN although ADNAN and ADAN are competitive. The ADNAN not only could get more smaller objective error but also get more fast convergence speed (see **Figure 3**). In addition, ADNAN objective value could get more smaller after a few iterations than ADAN. As a whole, the effect of ADNAN is superior to ADAN.

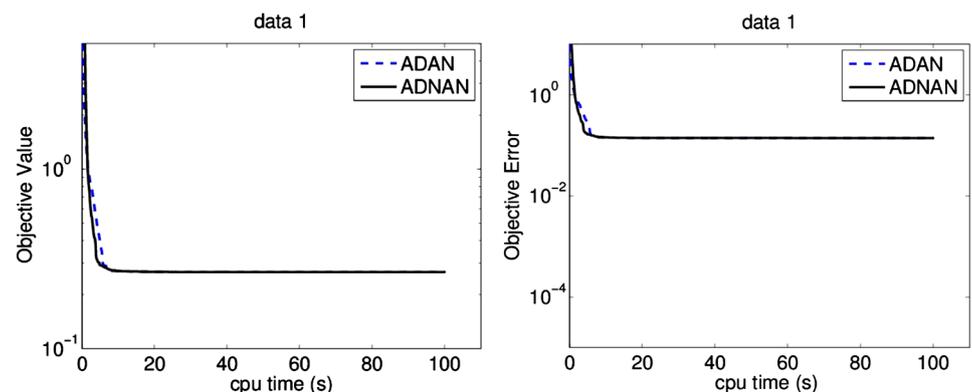


Figure 1. $\Psi = 0.1275$.

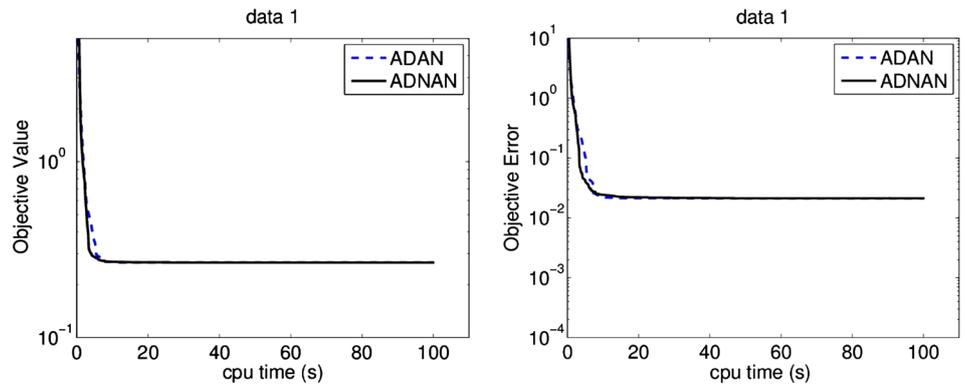


Figure 2. $\Psi = 0.2456$.

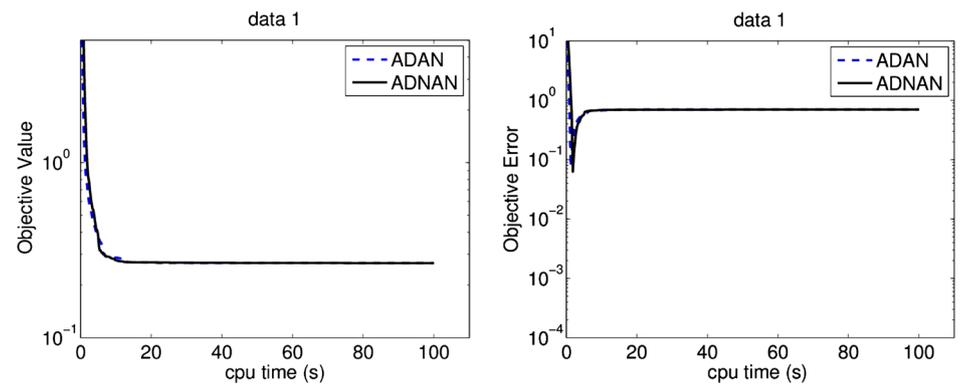


Figure 3. $\Psi = 0.9627$.

On the other hand, the experiment results about Algorithm 3 are as follows:

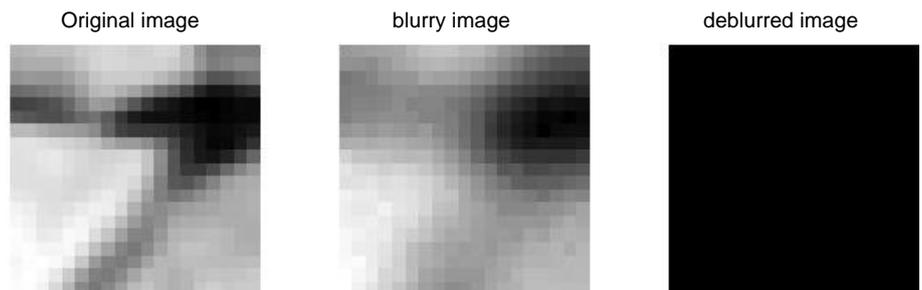


Figure 4. $l = -255, u = 0$.

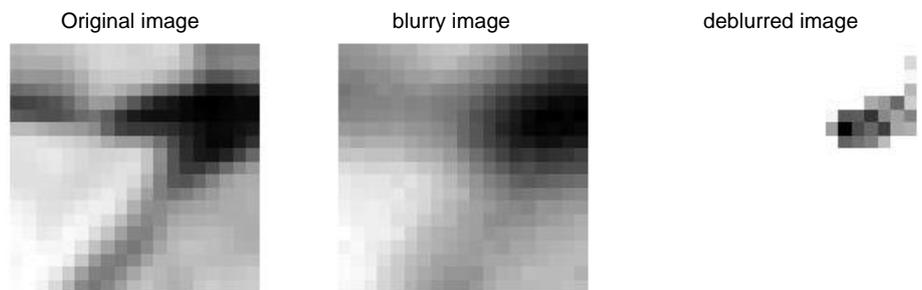


Figure 5. $l = -255, u = 150$.

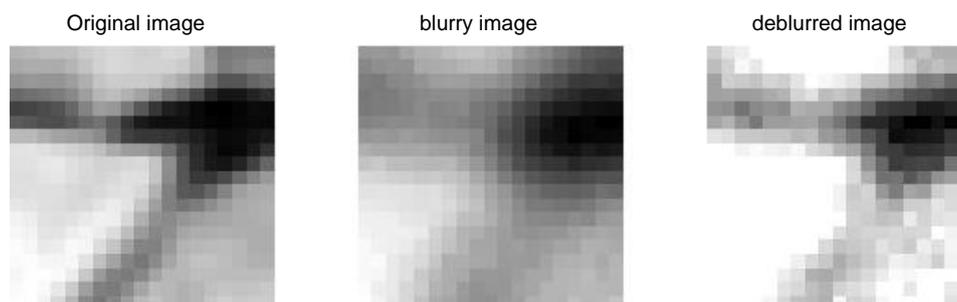


Figure 6. $l = -255, u = 200$.

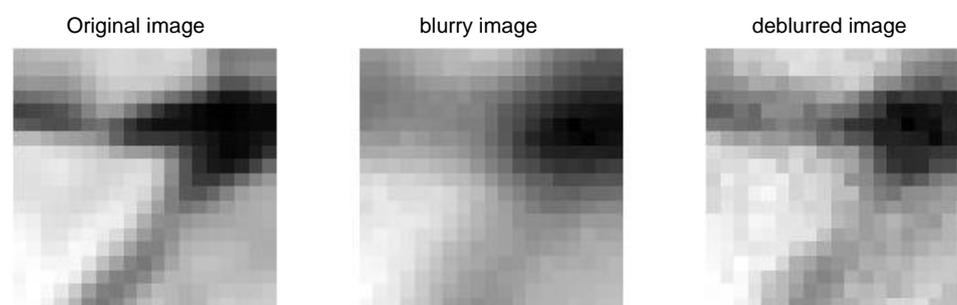


Figure 7. $l = -255, u = 255$.

7. Conclusions

According to the **Figures 1-3**, we can conclude that the nonmonotone line search could accelerate the convergence speed, furthermore ADNAN could get the objective values more stable and fast during the iterations when compared to ADAN.

On the other hand, the validness of GRAD is verified. Experiments results on image deblurring problems in **Figures 4-7** show that difference constraints on x can also get effective deblurred images.

Acknowledgements

This work is supported by Innovation Program of Shanghai Municipal Education Commission (No. 14YZ094).

References

- [1] Amit, Y. (2007) Uncovering Shared Structures in Multiclass Classification. *Machine Learning, Twenty-fourth International Conference*, **227**, 17-24.
<https://doi.org/10.1145/1273496.1273499>
- [2] Argyriou, A., Evgeniou, T. and Pontil, M. (2007) Multi-Task Feature Learning. *Advances in Neural Information Processing Systems*, **19**, 41-48.
- [3] Chambolle, A. (2004) An Algorithm for Total Variation Minimization and Applications. *Journal of Mathematical Imaging and Vision*, **20**, 89-97.
- [4] Chambolle, A. (2011) A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging. *Journal of Mathematical Imaging and Vision*, **40**, 120-145.
- [5] Zhang, L. and Vandenberghe, L. (2009) Interior-Point Method for Nuclear Norm Ap-

- proximation with Application to System Identification. *SIAM Journal on Matrix Analysis and Applications*, **31**, 1235-1256.
- [6] Hager, W. and Zhang, H.-C. (2015) An Alternating Direction Approximate Newton Algorithm for Ill-Conditioned Inverse Problems with Application to Parallel MRI. *Journal of the Operational Research Society of China*, **3**, 139-162.
- [7] Gabay, D. and Mercier, B. (1976) A Dual Algorithm for the Solution of Nonlinear Variational Problems via Finite Element Approximations. *Computers & mathematics with applications*, **2**, 17-40.
- [8] Zhang, J.J. (2013) Solving Regularized Linear Least-Squares Problems by the Alternating Direction Method with Applications to Image Restoration. *Electronic Transactions on Numerical Analysis*, **40**, 356-372.
- [9] Zhang, H.C. and Hager, W. (2004) A Nonmonotone Line Search Technique and Its Application to Unconstrained Optimization. *Society for Industrial and Applied Mathematics*, **14**, 1043-1056.
- [10] Grippo, L., Lampariello, F. and Lucidi, S. (1986) A Nonmonotone Line Search Technique for Newton's Method. *SIAM Journal on Mathematical Analysis*, **23**, 717-716.
- [11] Eckstein, J. and Bertsekas, D. (1992) On the Douglas-Rachford Splitting Method and the Proximal Point Algorithm for Maximal Monotone Operators. *Mathematical Programming*, **55**, 293-318. <https://doi.org/10.1007/BF01581204>
- [12] Barzilai, J. and Borwein, J. (1988) Two Point Step Size Gradient Methods. *IMA Journal of Numerical Analysis*, **8**, 141-148. <https://doi.org/10.1093/imanum/8.1.141>
- [13] Raydan, M. and Svaiter, B.F. (2002) Relaxed Steepest Descent and Cauchy-Barzilai-Borwein Method. *Computational Optimization and Applications*, **21**, 155-167. <https://doi.org/10.1023/A:1013708715892>
- [14] Block, K.T., Uecker, M. and Frahm J. (2007) Undersampled Radial MRI with Multiple Coils: Iterative Image Reconstruction Using a Total Variation Constraint. *Magnetic Resonance in Medicine*, **57**, 1086-1098. <https://doi.org/10.1002/mrm.21236>
- [15] Dai, Y.H. (2002) On the Nonmonotone Line Search. *Journal of Control Theory and Application*, **112**, 315-330.
- [16] Grippo, L., Lampariello, F. and Lucidi, S. (1989) A Truncated Newton Method with Nonmonotone Line Search for Unconstrained Optimization. *Journal of Optimization Theory and Applications*, **6**, 401-419. <https://doi.org/10.1007/BF00940345>



Submit or recommend next manuscript to SCIRP and we will provide best service for you:

Accepting pre-submission inquiries through Email, Facebook, LinkedIn, Twitter, etc.

A wide selection of journals (inclusive of 9 subjects, more than 200 journals)

Providing 24-hour high-quality service

User-friendly online submission system

Fair and swift peer-review system

Efficient typesetting and proofreading procedure

Display of the result of downloads and visits, as well as the number of cited articles

Maximum dissemination of your research work

Submit your manuscript at: <http://papersubmission.scirp.org/>

Or contact jamp@scirp.org