Affine scaling methods for image deblurring problems

Raymond H. Chan[*], Benedetta Morini[†] and Margherita Porcelli[**]

[*]*Department of Mathematics, The Chinese University of Hong Kong, Shatin, NT, Hong Kong.*
[†]*Dipartimento di Energetica "S. Stecco", Università di Firenze, via C. Lombroso 6/17, 50134 Firenze, Italia.*
[**]*Department of Mathematics, University of Namur, 61, rue de Bruxelles, B-5000 Namur, Belgium.*

Abstract. Blur in images can be removed by solving a series of box-constrained least-squares problems. In this paper, we compare two recent approaches for solving these problems using affine-scaling methods. Both approaches aim at solving a nonlinear system arising from the Karush-Kuhn-Tucker condition. One approach is to identify the active set and update the inactive components of the iterates by using a Newton-like method. The other is to iteratively solve the nonlinear system entry-wise by a Quasi-Newton method.

INTRODUCTION

Deblurring refers to the problem of finding the original image from a blurred and noisy image. The problem can be modeled algebraically in many ways depending on the smoothness of the image sought and the noise model [7]. For Gaussian noise, one common model is the bound-constrained linear least-squares problem:

$$\min_{l \le x \le u} \frac{1}{2} \|Ax - b\|^2, \tag{1}$$

where $\|\cdot\|$ is the 2-norm, $A \in R^{m \times n}$ is a blurring operator, $b \in R^m$ is the observed image, $x \in R^n$ the restored image to be sought, and the bounds $l \in (R \cup \{-\infty\})^n$, $u \in (R \cup \{+\infty\})^n$ $(m \ge n)$ represent the dynamic range of the image. For 8-bit gray-scale images, we have $l_i = 0$ and $u_i = 255$ for all $1 \le i \le n$. The inequalities $l \le x \le u$ are to be interpreted entry-wise. It is well-known that these problems are ill-posed as the blurring operators A are discretized integral operator [7]. Hence it is necessary to incorporate further information about the desired solution in order to stabilize the problem. The pervasive Tikhonov regularization defines the solution as the minimizer of a combination of the norm of both the residual and the solution, giving rise to the regularized box-constrained linear least-squares problem

$$\min_{l \le x \le u} q(x) = \min_{l \le x \le u} \left\{ \frac{1}{2} \|Ax - b\|^2 + \frac{1}{2} \lambda^2 \|Bx\|^2 \right\}. \tag{2}$$

Here $B \in R^{m \times n}$ may be the identity matrix or a regularization operator and $\lambda \in R$ is a regularization parameter that balances the weight of the norm of the solution relative to the norm of the residual.

Since Tikhonov approach can oversmooth images, especially when B is a high order differential operator, the total-variation (TV) is a prominent alternative regularization procedure, [9]. Interestingly, problem (2) may arise also in the solution of TV regularization; in fact the deblurring problem with TV regularization can be solved through a sequence of subproblems which consist of two parts: a TV denoising part and a Tikhonov-like deblurring part as in (2), see [6].

Recently, several algorithms have been proposed for the solution of (2), see e.g. [1, 3, 5, 8]. In this paper we focus on the numerical performance of affine-scaling procedures [4]. They are based on the Karush-Kuhn-Tucker (KKT) condition for (2) which can be written as a product of ∇q and a scaling matrix. Since the KKT condition is nonlinear, one can solve it by Newton-like methods. However, because of the size of the problem, specific consideration has to be taken to reduce the cost of computing the Newton steps. Here we consider and compare two different approaches, one is to solve the Newton step entry-wise, and the other is to solve it block-wise. The Affine Scaling Cyclic Barzilai-Borwein (AS_CBB) method in [5] belongs to the former class. It solves the KKT condition entrywise by a Quasi-Newton method and obtains a search direction without solving any linear systems. In contrast, the Reduced-Newton (RN) method in [8] combines a Newton method with an active set strategy. The Newton iteration only updates the inactive components, and hence reduces the size of the linear system to solve in each iteration. In this short communication, we compare the numerical performance of these two different approaches.

TWO AFFINE SCALING METHODS

In this section we give a brief description of the RN method and the AS_CBB method and refer to [5, 8] for details. We assume that q is strictly convex and the bounds are finite, which is indeed the case in image deblurring. The solution of problem (2) is unique and will be indicated as x^*. The superscript k denotes the iteration number in the algorithms, $g(x) = A^T(Ax - b) + \lambda^2 B^T Bx$ is the gradient of q, $g_i(x)$ represents the ith component of g. For a vector x, either x_i or $(x)_i$ stand for the ith component of x and $P(\cdot)$ denotes the projection map onto the box $[l, u]$.

The KKT conditions for (2) can be formulated as

$$D(x)g(x) = 0, \tag{3}$$

where $D(x) = \text{diag}(d_1(x), \ldots, d_n(x))$ has entries

$$d_i(x) = u_i - x_i, \ \text{if} \ g_i(x) < 0, \qquad d_i(x) = x_i - l_i, \ \text{if} \ g_i(x) > 0, \qquad d_i(x) = \min\{x_i - l_i, u_i - x_i\} \ \text{otherwise}.$$

We consider Newton method for the solution of (3) and let x^k be a strictly feasible iterate, i.e. $l < x^k < u$. Then the Newton equation, i.e. the linear system to be solved at each iteration, takes the form

$$[D(x^k)(A^T A + \lambda^2 B^T B) + E(x^k)]p^k = -D(x^k)g(x^k), \tag{4}$$

where the coefficient matrix is obtained by formal application of the product rule.

The RN method in [8] exploits an active set approach to reduce the dimension of the following linear system which is equivalent to (4):

$$M(x^k)p^k = -g(x^k), \qquad M(x^k) = (A^T A + \lambda^2 B^T B) + D(x^k)^{-1} E(x^k). \tag{5}$$

Here, to handle degenerate solution, matrix $E(x) = \text{diag}(e_1(x), \ldots, e_n(x))$ has the form

$$e_i(x) = |g_i(x)|, \ \text{if} \ |g_i(x)| < \min\{x_i - l_i, u_i - x_i\}^2 \ \text{or} \ |g_i(x)|^2 > \min\{x_i - l_i, u_i - x_i\}, \quad e_i(x) = 0 \ \text{otherwise}.$$

The active set strategy is designed as follows. Let $\delta^k = \min\{\delta, \sqrt{\|P(x^k - g(x^k)) - x^k\|}\}$ for a fixed positive $\delta \leq \min_i |u_i - l_i|/2$ and identify the active set $\mathscr{A}_k$ at x^k by setting $\mathscr{A}_k = \{i : (x^k)_i - l_i \leq \delta^k \ \text{or} \ u_i - (x^k)_i \leq \delta^k\}$. For the component of x^k in $\mathscr{A}_k$, the step to the nearest bound is taken. This amounts to forming the subvector $(p^k)_{\mathscr{A}_k}$ of p^k having components $(p^k)_i$, $i \in \mathscr{A}_k$, given by

$$(p^k)_i = u_i - (x^k)_i, \quad \text{if} \quad u_i - (x^k)_i \leq \delta^k, \qquad (p^k)_i = l_i - (x^k)_i \quad \text{otherwise}.$$

Now, consider the inactive set $\mathscr{I}_k = \{1, \ldots, n\} \setminus \mathscr{A}_k$ at x^k. The vector $(p^k)_{\mathscr{I}_k}$ which represents the components of $(p^k)_i$ of p^k for $i \in \mathscr{I}_k$ is computed by the Newton equation (5) as follows. Let $(M(x^k))_{\mathscr{T}\mathscr{V}}$ be the submatrix of $M(x^k)$ with elements m_{ij}^k, $i, \in \mathscr{T}$, $j \in \mathscr{V}$, and omit permutations for simplicity. The linear system (5) can be formulated as

$$\begin{pmatrix} (M(x^k))_{\mathscr{I}_k \mathscr{I}_k} & (M(x^k))_{\mathscr{I}_k \mathscr{A}_k} \\ (M(x^k))_{\mathscr{A}_k \mathscr{I}_k} & (M(x^k))_{\mathscr{A}_k \mathscr{A}_k} \end{pmatrix} \begin{pmatrix} (p^k)_{\mathscr{I}_k} \\ (p^k)_{\mathscr{A}_k} \end{pmatrix} = - \begin{pmatrix} (g(x^k))_{\mathscr{I}_k} \\ (g(x^k))_{\mathscr{A}_k} \end{pmatrix}.$$

Since the vector $(p^k)_{\mathscr{A}_k}$ is fixed, $(p^k)_{\mathscr{I}_k}$ solves the first block row:

$$(M(x^k))_{\mathscr{I}_k \mathscr{I}_k} (p^k)_{\mathscr{I}_k} = -(g^k)_{\mathscr{I}_k} - (M(x^k))_{\mathscr{I}_k \mathscr{A}_k} (p^k)_{\mathscr{A}_k}.$$

Finally, since the definition of matrix $M(\cdot)$ needs strict feasibility of the iterates, we set $\hat{p}^k = \max\{\sigma, 1 - \|P(x^k + p^k) - x^k\|\} (P(x^k + p^k) - x^k)$ with $\sigma \in (0, 1)$ and then we set $x^{k+1} = x^k + \hat{p}^k$.

The theoretical properties of the RN method are as follows, see [8]. First, if x^k is sufficiently close to the solution x^* of (2), then $\mathscr{A}_k = \{i \in \{1, \ldots, n\} : x_i^* \in \{l_i, u_i\}\}$. Second, if $\mathscr{A}_k \neq \emptyset$, then by Cauchy Interlace Theorem, $(M(x^k))_{\mathscr{I}_k \mathscr{I}_k}$ is better conditioned than $M(x^k)$. Third, if x^0 is sufficiently close to x^*, the generated sequence $\{x^k\}$ converges to x^* quadratically even in the case of degeneracy. The procedure is embedded into a globalization strategy as described in [8, Section 5].

The AS_CBB method replaces the matrix $[D(x^k)(A^T A + \lambda^2 B^T B) + E(x^k)]$ in (4) by the diagonal matrix $[\mu_k D(x^k) + E(x^k)]$ where μ_k is a positive scalar computed by a Quasi-Newton rule. Thus, the ith component $(p^k)_i$ of p^k is given by

$$(p^k)_i = -\left(\frac{1}{\mu_k + |g_i(x^k)|/d_i(x^k)} \right) g_i(x^k). \tag{6}$$

The positive scalar μ_k is computed using a cyclic version of the Barzilai-Borwein (BB) stepsize rule. Let

$$\mu_0^{BB} = \max\{\bar{\mu}, \|g(x^0)\|_\infty\}, \quad \mu_k^{BB} = \arg\min_{\mu \geq \bar{\mu}} \|\mu s^{k-1} - y^{k-1}\| = \max\left\{\bar{\mu}, \frac{(s^{k-1})^T y^{k-1}}{(s^{k-1})^T s^{k-1}},\right\}, \quad k \geq 1,$$

where $\bar{\mu}$ is a fixed positive parameter, $s^{k-1} = x^k - x^{k-1}$, $y^{k-1} = g(x^k) - g(x^{k-1})$. The cyclic BB strategy consists in re-using the BB stepsize for several iterations, i.e. letting $c \geq 1$ be the cycle length and $l \geq 0$ be the cycle number, the value of the scalars μ_k is assigned by the rule $\mu_{cl+i} = \mu_{cl+1}^{BB}$, $i = 1, \ldots, c$. Finally, since the BB method does not monotonically reduce the value of the objective function, AS_CBB method generates a new iterate of the form $x^{k+1} = x^k + \zeta_k p^k$, where the stepsize $\zeta_k \in (0,1]$ is computed by a non-monotone line-search strategy. The generated sequence is strictly feasible, see [5, Lemma 3.4], and converges R-linearly to a nondegenerate solution of (2).

NUMERICAL EXPERIMENTS

The RN method and the AS_CBB method have been implemented in Matlab and run on an Intel Xeon (TM) 3.4 Ghz, 1GB RAM using Matlab 7.6 with machine precision $\varepsilon_m \approx 2 \cdot 10^{-16}$. Both procedures have been implemented in a matrix-free manner. In particular, the linear systems arising in the RN method have been solved by the Conjugate Gradient (CG) method with an accuracy determined by an adaptive rule and providing fast asymptotic convergence. In the RN method the action of $M(x^k)$ times a vector was computed by performing sparse mode products using the equalities

$$(M(x^k))_{\mathscr{I}_k \mathscr{I}_k} = (I)_{\mathscr{I}_k \mathscr{N}} M(x^k) (I)_{\mathscr{N} \mathscr{I}_k}, \qquad (M(x^k))_{\mathscr{I}_k \mathscr{A}_k} = (I)_{\mathscr{I}_k \mathscr{N}} M(x^k) (I)_{\mathscr{N} \mathscr{A}_k}, \qquad \mathscr{N} = \{1, \ldots, n\},$$

where I is the identity matrix and $\mathscr{N} = \{1, \ldots, n\}$. The parameter values of the RN method and AS_CBB method are those used for the implementations in [5, 8].

We considered the four test images in Figure 1 denoted as Satellite, Surf, Zebra, Bear which have several pixels with values equal to 0 or 255. Specifically, there are 89.81%, 21.51%, 17.62% and 3.83% active pixels in the true Satellite, Surf, Zebra and Bear images respectively. The images have dimension 256×256 and thus $m = n = 65536$ in (2). In (2), we fixed $A \in R^{n \times n}$ to be the out-of-focus blur with radius 3 and $B \in R^{n \times n}$ to be the gradient matrix. Moreover we let the observed image b be such that $b = Ax_{true} + \eta$ where x_{true} is the true image and η is the Gaussian white noise with standard derivation 1. The parameter λ is chosen by trial and error such that it maximizes the Peak Signal to Noise Ratio (PSNR) value of the reconstructed image.


Source: Berkeley Segmentation Dataset

FIGURE 1. The true images: Satellite, Surf, Zebra, Bear.

For both the procedures, a strictly feasible initial guess x_0 is obtained by computing the unconstrained minimizer of (2). In particular, the normal equations for (2) are solved by using the CG method until the residual is less than 10^{-7}, and then projecting the computed solution onto the box $[w, 254w]$, with $w = (1, \ldots, 1)^T \in R^n$.

The real components of the solutions returned by the two algorithms are rounded to integers to get the pixels values of the restore images. For this reason, very accurate solutions are pointlessly expensive and not useful for the quality of the image as they will give the same pixel values of less accurate solutions. Then, after a new iterate was computed we used the following stopping criterion in both iterative procedures $\|P(x^k + g(x^k)) - x^k\| \leq n \, 10^{-4}$.

The solutions computed by the RN method gave a lower value of the objective function q than those computed by the AS_CBB method and a more accurate restored image. In particular, the gain of the RN method over the AS_CBB

method in terms of the PSNR value varied between 0.26dB and 0.66dB. Hence we made further runs with the AS_CBB method to obtain an image with a PSNR value that is at least as good as the PSNR attained by the RN method. The results obtained are summarized in Table 1 where we report the number of iterations (iter), the total number of CG iterations (CG iter) for the RN method, the value of the objective function at the computed solution x_c ($q(x_c)$), the time in seconds (time) and the PSNR value of the restored images. We note that the PSNR values are equal for the two methods and that AS_CBB requires a high number of iterations. This depends on the R-linear convergence rate of the procedure. Interestingly, the computational time is in favour of the RN method and the gain over AS_CBB method is between 4.95 and 24.22 seconds. This fact clearly shows that the RN method is more competitive than the AS_CBB method. This is especially true when the image has a quite large number of active pixels; for then the RN method can take advantage of its fast convergence properties and of the active set strategy which reduces considerably the size of the linear systems to be solved.
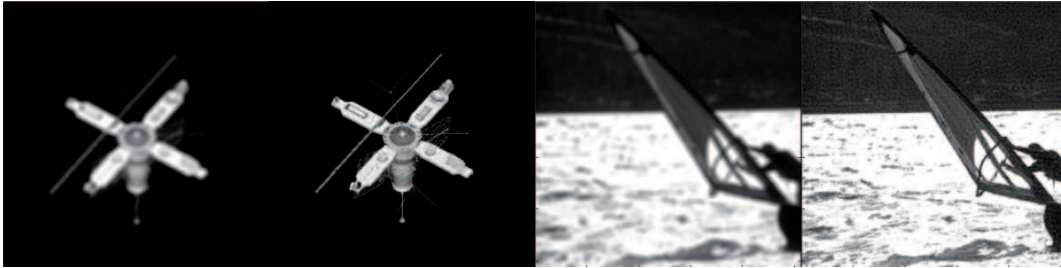
FIGURE 2. Restoration results. Satellite: Blurred image PSNR=25.23dB, (left); RN method PSNR=34.41dB, (right). Surf: Blurred image PSNR=21.49dB, (left); RN method PSNR=28.96dB, (right)

TABLE 1. Number of iterations, the total number of CG iterations for the RN method, value of the objective function, time in seconds and PSNR value of the restored images.

Image	iter	iter CG	time	PSNR	iter	time	PSNR
		RN			AS_CBB		
Satellite	5	44	30.02	34.41	91	36.08	34.43
Surf	5	65	37.64	28.96	111	42.59	28.96
Zebra	5	95	49.05	24.10	171	73.27	24.10
Bear	3	32	20.38	27.19	72	29.08	27.19

ACKNOWLEDGMENTS

REFERENCES

1. S. Bellavia, J. Gondzio, and B. Morini, *Numer. Linear Algebra Appl.*, to appear, DOI:10.1002/nla.732 (2010).
2. S. Bellavia, M. Macconi, and B. Morini, *Numer. Linear Algebra Appl.*, **13**, pp. 825–846 (2006).
3. D. Chen, and R. J. Plemmons, "Nonnegativity constraints in numerical analysis", in *The Birth of Numerical Analysis*, A. Bultheel and R. Cools Eds., World Scientific Press, River Edge, NJ, USA, 2009, pp. 109–140.
4. T. F. Coleman, Y. Li, *SIAM J. Optim*, **6**, pp. 418–445, (1996).
5. W. Hager, B. Mair, and H. Zhang, *Math. Program.*, **119**, pp. 1–32 (2009).
6. Y. Huang, M. Ng, and Y. Wen, *Multiscale Model. Simul.*, **7**, pp. 774–795 (2008).
7. P. C. Hansen, *Numer. Algor.*, **6**, pp. 1–35 (1994).
8. B. Morini, M. Porcelli, and R. H. Chan, *J. Comput. Appl. Math.*, **233**, pp. 2200–2212 (2010).
9. L. Rudin, S. Osher, and E. Fatemi, *Physica D*, **60**, pp. 259–268, (1992).