UPRE Method for Total Variation Parameter Selection

Youzuo Lin*

School of Mathematical and Statistical Sciences, Arizona State University, Tempe, AZ 85287 USA.

Brendt Wohlberg[1,]

T-5, Los Alamos National Laboratory, Los Alamos, NM 87545, USA.

Hongbin Guo

School of Mathematical and Statistical Sciences, Arizona State University, Tempe, AZ 85287 USA.

Abstract

Total Variation (TV) regularization is a popular method for solving a wide variety of inverse problems in image processing. In order to optimize the reconstructed image, it is important to choose a good regularization parameter. The Unbiased Predictive Risk Estimator (UPRE) has been shown to give a good estimate of this parameter for Tikhonov regularization. In this paper we propose an extension of the UPRE method to the TV problem. Since direct computation of the extended UPRE is impractical in the case of inverse problems such as deblurring, due to the large scale of the associated linear problem, we also propose a method which provides a good approximation of this large scale problem, while significantly reducing computational requirements.

Keywords: Parameter Selection, Total Variation Regularization, Large Scale Problem, Inverse Problem.

*Corresponding author

Email addresses: youzuo.lin@asu.edu (Youzuo Lin), brendt@lanl.gov (Brendt Wohlberg), hguo1@asu.edu (Hongbin Guo)

[1]This research was supported by the U.S. Department of Energy through the LANL/LDRD Program.

1. Introduction

Many image restoration tasks can be posed as linear inverse problems of the form

$$K\mathbf{x} = \mathbf{b} + \boldsymbol{\nu}, \tag{1}$$

where $\mathbf{b}$ represents the measured data, $\boldsymbol{\nu}$ represents noise, K is a linear transform (e.g. a convolution operator in the case of a deconvolution problem, and the identity in the case of denoising), and $\mathbf{x}$ represents the vectorised form of the recovered image. Regularization provides a method for controlling the noise and possible poor-conditioning of the operator K, prominent examples being the classical Tikhonov regularization [1],

$$\arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|K\mathbf{x} - \mathbf{b}\|_2^2 + \frac{\lambda}{2}\|H\mathbf{x}\|_2^2 \right\}, \tag{2}$$

where the matrix H is usually defined as a high-pass filtering operator, or identity matrix. The more recent TV regularization [2],

$$\arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|K\mathbf{x} - \mathbf{b}\|_2^2 + \lambda \|\mathbf{x}\|_{\mathrm{TV}} \right\}, \tag{3}$$

where the TV norm $\|\mathbf{x}\|_{\mathrm{TV}}$ is defined as $\left\| \sqrt{(D_x\mathbf{x})^2 + (D_y\mathbf{x})^2} \right\|_1$, with scalar operations applied to a vector considered to be applied element-wise, and the horizontal and vertical derivative operators written as D_x and D_y respectively. These two methods differ in the regularization term; TV regularization is more difficult to compute, but usually provides superior results.

Effective application of these regularization methods depends critically on correct selection of the regularization parameter λ. While it is common practice for the user to simply try various values until the solution looks reasonable, the preferred approach is to estimate the λ value which optimizes some objective measure of image quality, such as the Signal to Noise Ratio (SNR) of the reconstructed image with respect to the original undegraded image. There are several existing parameter selection methods for Tikhonov regularization [3, 4]: (1) those requiring some knowledge of the noise $\boldsymbol{\nu}$, such as the *Discrepancy Principle* [5], and the *UPRE* [4], and (2) those that do not, such as *Generalized Cross-Validation* (GCV) [6, 7] and the *L-Curve Method* [8]. Optimal parameter selection for TV regularization, in contrast, has received surprisingly little attention. To the best of our

knowledge, there are very few papers discussing this issue under the TV framework [9, 10, 11, 12, 13].

We chose to extend the UPRE method to TV regularization, based on its good performance in the Tikhonov case [14], as well as the conceptual simplicity of the extension. Since the direct extension is only able to deal with relatively small-scale problems, we also discuss how to bypass this obstacle by using a Krylov subspace method. Experimental results are provided to demonstrate the efficacy of our approach.

2. Unbiased Predictive Risk Estimator

The UPRE approach, also known as the C_L method, was first proposed [15] for regression problems, and then extended [4] to optimal parameter selection for Tikhonov problems. Define the *predictive error* $\mathbf{p}_\lambda = K\mathbf{x}_\lambda - K\mathbf{x}_{\text{true}}$, where $\mathbf{x}_\lambda \in \mathbb{R}^n$ is the computed solution for parameter λ, and $\mathbf{x}_{\text{true}} \in \mathbb{R}^n$ is the ground truth solution. According to the UPRE method, the optimal parameter λ as the minimizer of the *predictive risk* $(1/n)||\mathbf{p}_\lambda||^2$, which is statistically estimated since $\mathbf{x}_{\text{true}}$ is, in general, unknown. The full derivation [4, Sec. 7.1], which is too lengthy to reproduce here, depends on the ability to express the regularized solution as having linear dependence on the data, $\mathbf{x}_\lambda = R_{\text{TK},\lambda}\mathbf{b}$, where the *regularization matrix* $R_{\text{TK},\lambda} = (K^TK + \lambda I)^{-1}K^T$. Defining the *regularized residual* $\mathbf{r}_\lambda = K\mathbf{x}_\lambda - \mathbf{b}$, and the *influence matrix* $A_{\text{TK},\lambda} = K(K^TK + \lambda I)^{-1}K^T$, the optimal parameter λ is the minimizer of

$$\text{UPRE}_{\text{TK}}(\lambda) = \frac{1}{n}||\mathbf{r}_\lambda||^2 + \frac{2\sigma^2}{n}\text{trace}(A_{\text{TK},\lambda}) - \sigma^2, \tag{4}$$

where σ^2 is the noise variance. The primary computational cost of evaluating the function UPRE_{TK} at λ consists of solving the Tikhonov problem at λ to obtain $\mathbf{x}_\lambda$, from which $\mathbf{r}_\lambda$ is obtained, and, more significantly the computation of $\text{trace}(A_{\text{TK},\lambda})$.

2.1. Extension of UPRE to Total Variation Regularization

Extension of the UPRE to TV regularization is complicated by the absence of a linear equation $\mathbf{x}_\lambda = R_{\text{TV},\lambda}\mathbf{b}$ for the solution in terms of the data. Following part of the derivation of the Lagged Diffusivity algorithm [4, Sec. 8.2], we approximate the TV term $\|\mathbf{x}\|_{\text{TV}}$ by $\|\psi((D_x\mathbf{x})^2 + (D_y\mathbf{x})^2)\|_1$, where $\psi(x) = \sqrt{x + \beta^2}$ provides differentiability at the origin. Correspondingly,

the gradient of the TV term, $\nabla(\|\mathbf{x}\|_{\mathrm{TV}})$, at x_λ can be written as $L(\mathbf{x}_\lambda)\mathbf{x}$, where

$$L(\mathbf{x}_\lambda) = D_x^T \mathrm{diag}(\psi'(\mathbf{x}_\lambda))D_x + D_y^T \mathrm{diag}(\psi'(\mathbf{x}_\lambda))D_y,$$

allowing one to define

$$R_{\mathrm{TV},\lambda} = (K^T K + \lambda L(\mathbf{x}_\lambda))^{-1} K^T,$$

which is in the required form except for the dependence of matrix $L(\mathbf{x}_\lambda)$ on $\mathbf{x}_\lambda$.

Followed this idea, the influence matrix in the TV case can be written as,

$$A_{\mathrm{TV},\lambda} = K(K^T K + \lambda L(x_\lambda))^{-1} K^T. \tag{5}$$

The derivation (which is too lengthy to reproduce here, please refer to [4] for more details) of $\mathrm{UPRE}_{\mathrm{TK}}(\lambda)$ depends on the symmetry of $A_{\mathrm{TK},\lambda}$ and the Trace Lemma [4]. Since $A_{\mathrm{TV},\lambda}$ is also symmetric, the functional for $\mathrm{UPRE}_{\mathrm{TV}}(\lambda)$ can be derived in a similar way, and it can be shown that the UPRE for TV method shares the same form of expression as the Tikhonov method, with UPRE functional

$$\mathrm{UPRE}_{\mathrm{TV}}(\lambda) = \frac{1}{n}||\mathbf{r}_\lambda||^2 + \frac{2\sigma^2}{n}\mathrm{trace}(A_{\mathrm{TV},\lambda}) - \sigma^2. \tag{6}$$

2.2. Computational Limitations

In the Tikhonov case the computation of $\mathrm{trace}(A_{\mathrm{TK},\lambda})$ in (4) is straightforward if the Singular Value Decomposition (SVD) of A is available, but in many large scale problems it is too expensive to compute the SVD of A. In [16] an approximation method is proposed to approximate the value of $\mathrm{trace}(A_{\mathrm{TK},\lambda})$ and related work can be found in [17, 18, 19, 20].

The primary difficulty in implementing the UPRE in the TV case is the computation of $\mathrm{trace}(A_{\mathrm{TV},\lambda})$ in (6), since the linear approximation of the regularization term in the TV case further complicates the computation of UPRE in comparison with the Tikhonov case. Direct computation of the UPRE imposes very severe limits on the problem size due to computation time and memory requirements. In the following sections, we will introduce an algorithm which computes an approximation of the UPRE with vastly reduced computational cost, allowing application of this method to standard image sizes. In implementing this approximation, an enormous reduction in memory requirements is achieved by avoiding explicit construction of matrices such as A, D_x, D_x^T, D_y and D_y^T, the algorithm implementation requiring only matrix-vector products involving these matrices.

2.3. Extension of UPRE to Large Scale Total Variation

In the computation of (6), the most expensive part, as mentioned above, is the trace of the influence matrix, trace$\{K(K^TK + \lambda L(x_\lambda))^{-1}K^T\}$, since we need to deal with an inverse first then find the trace value. Applying the approach of Hutchinson [21], we can approximate trace$(f(M))$ by the unbiased trace estimator

$$E(\mathbf{u}^T f(M)\mathbf{u}) \simeq \text{trace}(f(M)), \tag{7}$$

where $\mathbf{u}$ is a discrete multivariate random variable, which takes each entry the values -1 and +1 with probability 0.5, and the matrix M is symmetric positive definite (SPD).

Define the eigenvalue decomposition of M as $M = Q^T\Lambda Q$, where Q is an orthogonal matrix and Λ is a diagonal matrix of eigenvalues ρ_i in increasing order. Then, following [19, 20], it can be shown that

$$\begin{aligned} \mathbf{u^T} f(M)\mathbf{u} &= \sum_{i=1}^{n} f(\rho_i)\tilde{\mathrm{u}}_i^2 \\ &= \int_a^b f(\rho)d\mu(\rho), \end{aligned} \tag{8}$$

where $\tilde{\mathbf{u}} = Q\mathbf{u}$, $\tilde{u}_i$ are the components of $\tilde{\mathbf{u}}$, and the measure $\mu(\rho)$ is defined as

$$\mu(\rho) = \begin{cases} 0 & \text{if } a \le \rho < \rho_1 \\ \sum_{j=1}^{i} \tilde{u}_j^2 & \text{if } \rho_i \le \rho < \rho_{i+1}, \quad 1 < i < n \\ \sum_{j=1}^{n} \tilde{u}_j^2 & \text{if } \rho_n \le \rho < b. \end{cases} \tag{9}$$

In order to compute the Riemann-Stieltjes integral in (8), Gauss quadrature is used,

$$\int_a^b f(\rho)d\mu(\rho) \approx \sum_{i=1}^{k} \omega_i f(\theta_i), \tag{10}$$

where the weights ω_i and the nodes θ_i are unknown. Golub *et al.* [19] mentioned a way to determine ω_i and θ_i, by constructing a sequence of orthogonal polynomials $p_k(\rho)_{k=0}^N$, based on the measure $\mu(\rho)$ in (9), subject to

$$\int_a^b p_i(\rho)p_j(\rho)d\mu(\rho) = \delta_{ij}.$$

and satisfying a three term recurrence relation

$$\gamma_k p_k(\rho) = (\rho - \alpha_k)p_{k-1}(\rho) - \gamma_{k-1}p_{k-2}(\rho),$$

where $k \in \{1, ..., N\}$, $p_{-1}(\rho) = 0$, and $p_0(\rho) = 1$.

The equivalent matrix form of the above can be written as,

$$\rho \mathbf{P}(\rho) = T_N \mathbf{P}(\rho) + \gamma_N \mathbf{P}(\rho)\mathbf{e_N},$$

where $\mathbf{P^T}(\rho) = [p_0(\rho), ..., p_{N-1}(\rho)]$, $\mathbf{e_N^T} = [0, ..., 0, 1]$, and

$$T_N = \begin{pmatrix} \alpha_1 & \gamma_1 & & & \\ \gamma_1 & \alpha_2 & \gamma_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \gamma_{N-2} & \alpha_{N-1} & \gamma_{N-1} \\ & & & \gamma_{N-1} & \alpha_N \end{pmatrix}. \tag{11}$$

It has been shown in [22] that θ_i are the eigenvalues of T_N, and ω_i are squares of the first components of the normalized eigenvectors of T_N, and this observation is utilized as Golub-Welsch algorithm. Golub [19] points out that these eigenvalues/vectors can be computed by applying the Lanczos procedure to M with initial vector $\mathbf{x}^{(0)} = \mathbf{u}/\|\mathbf{u}\|_2$, and such approximation works best for sparse matrices. Further details of the procedure outlined above can be found in [19, 20].

In our problem, the linear transform matrix K is often reasonably sparse, and more importantly, the matrix $K^TK + \lambda L(x_\lambda)$ is SPD. We can adapt the above algorithm by setting $f(x) = 1/x$ and deriving the trace of $\mathrm{A_{TV,\lambda}}$ as

$$\begin{aligned} \mathrm{trace}(A_{\mathrm{TV},\lambda}) &\approx E(\mathbf{u^T}K((K^TK + \lambda L(x_\lambda))^{-1})K^T\mathbf{u}), \\ &= E(\mathbf{v^T}((K^TK + \lambda L(x_\lambda))^{-1})\mathbf{v}), \end{aligned} \tag{12}$$

where $\mathbf{v} = K^T\mathbf{u}$. Related to the eq. (7) above, the matrix $M = K^TK + \lambda L(x_\lambda)$ and $f(x) = x^{-1}$. The tridiagonal matrix T_N in eq. (11) can be derived correspondingly under this transformation, and the weights ω_i and the nodes θ_i can be assured according to the "Golub-Welsch" algorithm as mentioned. Numerically, we need to implement the Lanczos procedure on the newly defined matrix M, but with a slightly different starting vector. Instead of using $\mathbf{x}^{(0)} = \mathbf{u}/\|\mathbf{u}\|_2$ as previously, we now need to utilize the corresponding initial vector for this case, i.e., $\mathbf{x}^{(0)} = \mathbf{v}/\|\mathbf{v}\|_2$.

If provided with a confidence probability p $= 1 - \alpha$ and the required accuracy δ of the approximated trace value, it has been shown [20] that we can find out the sample size, N, of the Monte Carlo test which needs to be processed to achieve this accuracy of δ. Specifically, by denoting AppTR_j as the jth approximation to the trace, we have

$$N \geq \left(\frac{\rho_\alpha}{\delta}\right)^2 \left(\frac{\sigma_t}{m_t}\right)^2, \tag{13}$$

where m_t and σ_t are the mean and standard deviation respectively of $\{\mathrm{AppTR}_j | j \in \{1, 2, \ldots, t\}\}$.

2.4. General Algorithm

For Monte Carlo test sample size N given by (13), the general algorithm for computing the approximated trace is illustrated as follows:

for (j = 1 **to** N)
  Setup the tolerance, TOL;
  Setup the initial vector for the Lanczos procedure as:
  $\mathbf{x}^{(0)} = \mathbf{v}/\|\mathbf{v}\|_2$ as in (12);
  Setup the Lanczos level, k = 1;
  while (RelErr > TOL)
    $\alpha_k = \mathbf{x}^{(k-1)T}(K^T K + \lambda L(x_\lambda))\mathbf{x}^{(k-1)}$;
    $\mathbf{z} = (K^T K + \lambda L(x_\lambda))\mathbf{x}^{(k-1)}$;
    $\mathbf{z} = \mathbf{z} - \alpha_k \mathbf{x}^{(k-1)} - \gamma_{k-1}\mathbf{x}^{(k-2)}$;
    $\gamma_k = \|\mathbf{z}\|_2$;
    $\mathbf{x}^{(k)} = \mathbf{z}/\gamma_k$;
    Construct the matrix T according to (11);
    Compute the Ritz values θ_i and the first elements ω_i
    of Ritz vectors of T;
    Compute $I_G = \sum_{i=1}^{LEV} \omega_i f(\theta_i)$ by using Gauss
    quadrature as described in (10);
    Update the relative error RelErr from two successive
    approximations of I_G;
    $k = k + 1$;
  end (while)
  $\mathrm{AppTR}_j = \|\mathbf{v}\|_2^2 I_G$;

Approx. Trace $= 1/j \sum_{p=1}^{j} \mathrm{AppTR}_p$;
Update the Monte Carlo test sample size N according to (13);
end (for)

It is worth pointing out that the test sample size N given by (13) is updated each time along with a new approximation of AppTR_j is computed.

Once we have the trace value, the UPRE functional value can be found using (6), and in our algorithm, we use two ways to locate the minimizer of the UPRE functional. One is by exhaustive search, which means we set up a grid of points of λ, and sweep through all of them and find out the corresponding λ value with respect to to the minima. The other method is based on the Golden Section search [23]. From numerical experiments, we have found that calculation of the UPRE functional is significantly slower for large λ (due to the corresponding computational cost of the TV solver at large λ). The Golden Search method is much faster than the exhaustive search, as it computes the UPRE functional at fewer λ values, and, when the target λ is small, is able to avoid more than one computation for large λ.

3. Computational Results

In this section, we present four numerical tests: one denoising problem, two deblurring problems, and some comparison results, all of which are implemented in Matlab. The blurring kernel is chosen to be Gaussian, with additive Gaussian white noise. We assume that the variances of the added noise are known for all our tests below, for those applications where the variances are unknown, certain techniques to estimate the noise [24, 25] can be applied before performing our proposed algorithm. Since we are using synthetic data, we can find out the true optimal parameter λ by sweeping over a wide enough range of possible values of $[\lambda_{\min}, \lambda_{\max}]$, and compare the corresponding SNR values. Similarly, we can also compute the *predictive risk*. For the purpose of showing the effectiveness of our approximation algorithm, we compute the exact UPRE value at the same time by exactly constructing the explicit matrix A_λ in (4). We evaluate the performance of our approximated UPRE approach by comparing to the above three values.

3.1. Test Problem 1 - Deblurring: Lena Image with Size 32 × 32

To demonstrate the strength of our approximation algorithm, we set up a small test problem, so that we can compute both the direct method, by

using the explicit matrix for the UPRE value, and the approximated method proposed above. We compare our approximated UPRE with the *predictive risk*, the exact UPRE and the actual result using the optimal parameter. The test image is a 32×32 image cut from the original 256×256 version of the well-known "Lena" image. The test image is blurred using an 11×11 Gaussian kernel before adding zero-mean white noise with a variance of 63.0. The corresponding blurring matrix has size 1024×1024. Table 1 shows the results of estimated parameter λ and the estimated SNR value of the recovered image using the approximated UPRE, the true predictive risk and the exact UPRE functional as well as the optimal values. In this example, the Exact UPRE is a good estimate of the True Predictive Risk, and the Approximate UPRE is a good estimate of Exact UPRE (in fact, in this small example, these values are the same to displayed precision), and the resulting λ estimate is very close to the optimal λ (the same to displayed precision).

	Optimal	True Pred. Risk	Exact UPRE	Approx. UPRE
SNR (dB)	14.70	14.70	14.70	14.70
λ	94.90	94.90	94.90	94.90

Table 1: Performance results for Test Problem 1. Image size: 32×32, Lena; blurring matrix size: 1024×1024, consisting of 11×11 Gaussian kernel with $\sigma = 0.46$; Gaussian noise with 0 mean and variance 63.0; SNR of the original noisy image: 13.24 dB; accuracy of the trace approximation in (13): $\delta = 0.02$.

3.2. Test Problem 2 - Denoising: Lena Image with Size 256 × 256

In order to show the feasibility of UPRE for denoising, in our first example we generate a denoising problem by setting $K = \mathrm{I}$ in (3). Gaussian noise with mean 0 and variance 37.9 was added to the image. The UPRE plot and the SNR plot are given separately in Figure 1 and in Figure 2. The optimal λ estimated by UPRE gives a very promising result, since in Figure 2 the estimated optimal SNR value is very close to the maximum SNR. The TV solver used in this testing problem is lagged-diffusivity [4], but this is not a requirement of the method.

3.3. Test Problem 3 - Deblurring: Lena Image with Size 256 × 256

In this test problem, we use the whole 256×256 pixel Lena image, again adding Gaussian white noise after blurring with a Gaussian kernel. The blurring matrix under this setup has size 65536×65536, which is too large

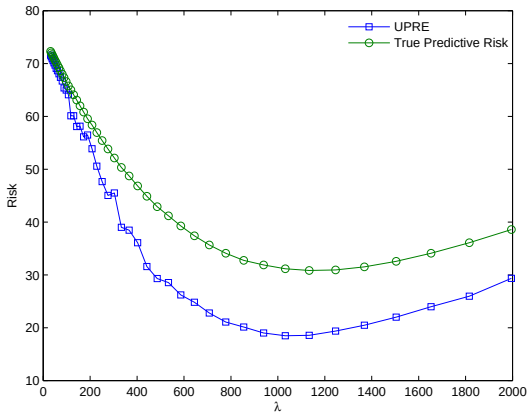

Figure 1: Comparison of *predictive risk* and approximate UPRE for the 256×256 denoising test. Note that location of the minima are so close that they are virtually indistinguishable.

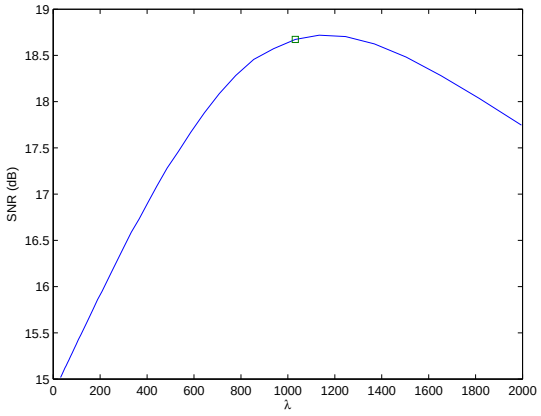

Figure 2: Variation of reconstruction SNR (with respect to known ground truth image) against λ. Note the proximity of the maximum SNR region to the minimum of the Approximate UPRE in Figure 1.

to be able to compute the SVD, so the traditional UPRE is impractical due to memory limitations. However by using our approximated algorithm, we can still process this image and the numerical results in Table 2 show the good accuracy of our approximated algorithm by comparing column 5 and column 2. Figure 3 compares the *predictive risk* with the UPRE value, from which we can see that the minima of those two curves are located close together. It is worth pointing out that, what we want here is the location of the minimum of the function, so the functional value is irrelevant. In Fig. (4), the box is the SNR value achieved by using the UPRE approach, which is close to the maximum of the SNR value.

	Optimal	True Pred. Risk	Exact UPRE	Approx. UPRE
SNR (dB)	15.27	15.27	-	15.25
λ	257.85	257.85	-	228.61

Table 2: Performance results for Test Problem 3. Image size: 256×256, Lena; blurring matrix size: 65536×65536, consisting of 11×11 Gaussian kernel with $\sigma = 0.91$; Gaussian noise with 0 mean and variance 27.93; SNR of the original degraded image: 13.13 dB; accuracy of the trace approximation in (13): $\delta = 0.02$.

3.4. Test Problem 4 - Comparison Result

For the denoising problem, we compare our algorithm performance with that of Gilboa *et al.* [10], which appears to be the most effective of the existing methods, [11], [12], and [13] (of which there are surprisingly few, given the prominence of TV as a regularization method). From Figure 5, we can see that our estimation provides a much closer match to the "Optimal SNR" curve; the average difference of our estimation and optimal value is 0.02 dB, while the average loss of the method of Gilboa *et al.* is 0.76 dB.

Similar performance results for the deblurring problem are provided in Figure 6, of which nine different deblurring tests with original SNR ranging from 11.36 dB to 29.49 dB are constructed and tested using UPRE approach. In this comparison test, we are unable to provide results for other existing approach, as all of those of which we are aware are restricted to the denoising problem. Again our algorithm gives very good results, with an average loss of 0.03 dB.

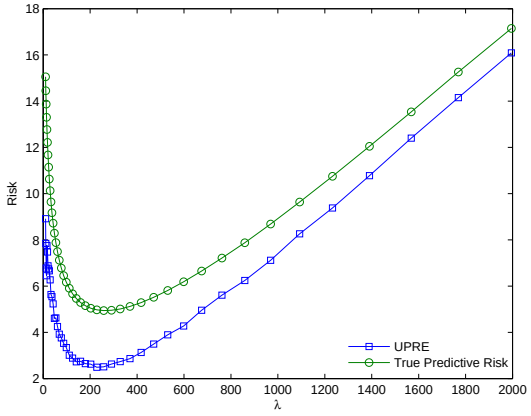

Figure 3: Comparison of *predictive risk* and approximate UPRE for the 256×256 deblurring test. Note that the location of the minima at the curve are very close.

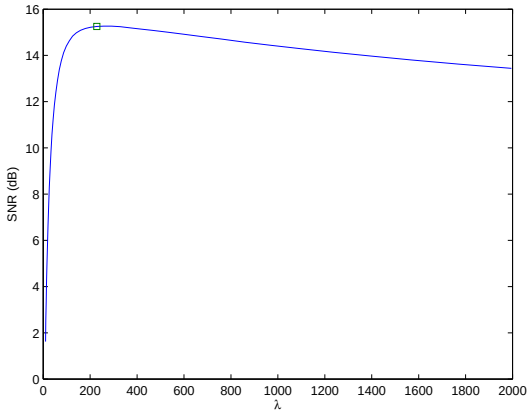

Figure 4: Variation of reconstruction SNR (with respect to known ground truth image) against λ. Note the proximity of the maximum SNR region to the minimum of the Approximate UPRE in Figure 3.

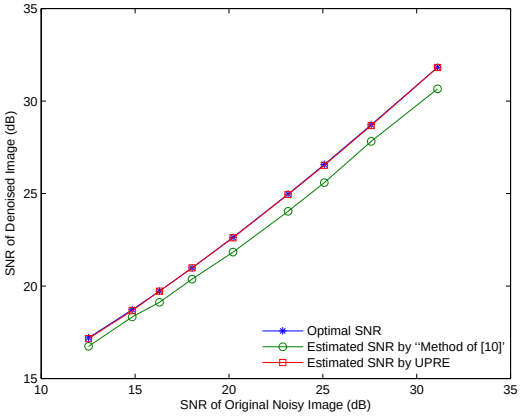

Figure 5: Plot of Optimal SNR and Estimated SNR versus original noisy SNR.

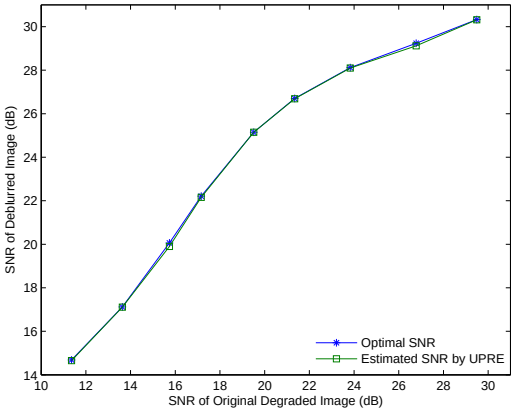

Figure 6: Plot of Optimal SNR and Estimated SNR versus original noisy SNR.

4. Conclusions

Our method for computing the UPRE for Total Variation regularization, via the approximated trace, gives a good approximation to the UPRE computed using the exact trace value, while very significantly reducing computational requirements. The approximated value may be computed without explicit representation of very large matrices, and therefore also avoids memory limitations which prevent the application of the direct computation to problems involving images of practical size. In addition to providing better accuracy than the method of Gilboa *et al.* [10] for the denoising problem, the proposed approach is applicable to more general problems, such as deconvolution.

[1] A. N. Tikhonov, Regularization of incorrectly posed problems, Soviet mathematics - Doklady 4 (1963) 1624–1627.

[2] L. Rudin, S. Osher, E. Fatemi, Nonlinear total variation based noise removal algorithms, Physica D 60 (1992) 259–268.

[3] P. C. Hansen, Rank-Deficient and Discrete Ill-Posed Problems, SIAM, 1997.

[4] C. R. Vogel, Computational Methods for Inverse Problems, SIAM, 2002.

[5] V. A. Morozov, Methods for solving Incorrectly posed problems, Springer-Verlag, 1984.

[6] G. H. Golub, M. T. Heath, G. Wahba, Generalized cross-validation as a method for choosing a good ridge parameter, Technometrics 21 (2) (1979) 215–223.

[7] G. Wahba, Practical approximate solutions to linear operator equations when the data are noisy, SIAM Journal on Numerical Analysis 14 (4) (1977) 651–667.

[8] P. C. Hansen, D. P. O'Leary, The use of the L-curve in the regularization of discrete ill-posed problem, SIAM Journal on Scientific Computing 14 (6) (1993) 1487–1503.

[9] S. D. Babacan, R. Molina, A. K. Katsaggelos, Parameter estimation in TV image restoration using variational distribution approximation, IEEE Transactions on Image Processing 17 (3) (2008) 326–339.

[10] G. Gilboa, N. Sochen, Y. Y. Zeevi, Estimation of optimal PDE-based denoising in the SNR sense, IEEE Transactions on Image Processing 15 (8) (2006) 2269–2280.

[11] D. M. Strong, T. F. Chan, Relation of regularization parameter and scale in total variation based image denoising, Tech. rep., UCLA Computational and Applied Mathematics (CAM) Report (1996).

[12] J.-F. Aujol, G. Gilboa, T. F. Chan, S. J. Osher, Structure-texture image decomposition-modeling, algorithms, and parameter selection, International Journal of Computer Vision 67 (1) (2006) 111–136.

[13] V. Solo, Selection of regularisation parameters for total variation denoising, in: IEEE International Conference on Acoustics, Speech, and Signal Processing, Vol. 3, 1999, pp. 1653–1655.

[14] J. Mead, R. A. Renaut, A Newton root-finding algorithm for estimating the regularization parameter for solving ill-conditioned least squares problems, Inverse Problems 25 (2) (2009) 025002 (19pp).

[15] C. L. Mallows, Some comments on C_p, Technometrics 15 (4) (1973) 661–675.

[16] Y. Lin, B. Wohlberg, Application of the UPRE method to optimal parameter selection for large scale regularization problems, 2008, pp. 89–92.

[17] M. E. Kilmer, D. P. O'Leary, Choosing regularization parameters in iterative methods for ill-posed problems, Journal on Matrix Analysis and Applications 22 (4) (2001) 1204–1221.

[18] D. A. Girard, The fast Monte-Carlo cross-validation and C_L procedures: Comments, new results, and application to image recovery problems, Computational Statistics 10 (1995) 205–231.

[19] G. H. Golub, U. von Matt, Generalized cross-validation for large-scale problems, Journal of Computational and Graphical Statistics 6 (1) (1997) 1–34.

[20] H. Guo, Computing trace of function of matrix, Numerical Mathematics 9 (2) (2000) 204–215.

[21] M. F. Hutchinson, A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines, Communications in Statistics, Simulation, and Computation 18 (3) (1989) 1059–1076.

[22] G. H. Golub, J. H. Welsch, Calculation of gauss quadrature rules, Mathematics of Computation 23 (1969) 221–230.

[23] R. P. Brent, Algorithms for Minimization without Derivatives, Prentice-Hall, 1973.

[24] S. I. Olsen, Noise variance estimation in images, in: Proceedings of the 8th Scandinavian Conference on Image Analysis, 1993, pp. 25–28.

[25] T. Gasser, L. Sroka, C. Jennen-Steinmetz, Residual variance and residual pattern in nonlinear regression, Biometrika 73 (3) (1986) 625 – 633.