Consiglio Nazionale delle Ricerche

Generalized Cross-Validation applied to Conjugate Gradient for discrete ill-posed problems

P.Favati, G.Lotti, O.Menchi , F.Romani

IIT TR-09/2013

Technical report

maggio 2013

Istituto di Informatica e Telematica

Generalized Cross-Validation applied to Conjugate Gradient for discrete ill-posed problems

P. Favati [a], G. Lotti [b], O. Menchi [c,*], F. Romani [c]

[a] *IIT - CNR Via G. Moruzzi 1, 56124 Pisa, Italy.*

[b] *Dip. di Matematica, University of Parma, Parco Area delle Scienze 53/A, 43124 Parma, Italy.*

[c] *Dip. di Informatica, University of Pisa, Largo Pontecorvo 3, 56127 Pisa, Italy.*

Abstract

To apply the Generalized Cross-Validation (GCV) as a stopping rule for an iterative method, we must estimate the trace of the so-called influence matrix which appears in the denominator of the GCV function. In the case of conjugate gradient, unlike what happens with stationary iterative methods, the regularized solution has a nonlinear dependence on the noise which affects the data of the problem. This fact is often pointed out as a cause of poor performance of GCV. To overcome this drawback, in this paper we propose a new method which linearizes the dependence by computing the derivatives through iterative formulas along the lines of Perry and Reeves (1994) and Bardsley (2008). We compare the proposed method with other methods suggested in the literature by an extensive numerical experimentation both on 1D and on 2D test problems.

Key words: Regularization Problems, Conjugate Gradient, Generalized Cross-Validation

* Corresponding author.
Email addresses: paola.favati@iit.cnr.it (P. Favati), grazia.lotti@unipr.it (G. Lotti), menchi@di.unipi.it (O. Menchi), romani@di.unipi.it (F. Romani).

1 Introduction

Given a matrix $A \in \mathbf{R}^{n\times n}$ and a vector $\boldsymbol{b} \in \mathbf{R}^n$, we consider the system

$$A\boldsymbol{x} = \boldsymbol{b}. \tag{1}$$

We assume that A is a large full rank matrix, having singular values which gradually decay to zero, so that it is difficult to determine its numerical rank. In many applications the available right-hand side of the system is contaminated by a noise $\boldsymbol{\eta}$ accounting for both the measurement errors and the process involved in the construction of the discrete model describing the underlying continuous phenomenon, i.e.

$$\boldsymbol{b} = \boldsymbol{b}^* + \boldsymbol{\eta}.$$

The vectors $\boldsymbol{b}^*$ and $\boldsymbol{x}^*$, such that $A\boldsymbol{x}^* = \boldsymbol{b}^*$, are considered the exact right-hand side and the exact solution of the system. Classical examples of this kind of problems arise from the discretization of Fredholm integral equations of the first kind, as for instance in the imaging deconvolution, where A represents an imaging system, $\boldsymbol{x}^*$ an object, $\boldsymbol{b}^*$ the noise-free image of the object and $\boldsymbol{b}$ the noisy image.

Due to the ill-conditioning of the matrix and the presence of the noise, the solution $\widetilde{x} = A^{-1}\boldsymbol{b}$ is often a poor approximation of $\boldsymbol{x}^*$ even if the magnitude of $\boldsymbol{\eta}$ is small, and the problem of finding a good approximation of $\boldsymbol{x}^*$ turns out to be a discrete ill-posed problem [7]. Special techniques called *regularization* methods are required to deal with this kind of problems. Both direct methods (as Tikhonov method) and iterative methods can be used to this aim. Iterative methods are suggested for large matrices A without particular structure properties. The iterative method has to enjoy the semi-convergence property, i.e. in presence of the noise it reconstructs first the low-frequency components, which correspond to the largest singular values of A. The iteration should be stopped before the high-frequency components of the noise start to enter the computed solution. In this sense the iteration number plays the role of the regularization parameter.

Among the classical semi-convergent methods we consider here the *conjugate gradient* method (CG). The regularizing properties of CG are well known (see for example [14]). CG has in general a good convergence rate and finds quickly an optimal vector $\boldsymbol{x}_{opt}$ which minimizes the error with respect to $\boldsymbol{x}^*$. This behavior can be a disadvantage in the regularization context, because also the high-frequency components enter quickly the computed solution and the error increases sharply after the optimal number k_{opt} of steps. As a matter of fact,

the determination of k_{opt} is very sensitive to the perturbation of the right-hand side [3]. As a consequence, the regularizing efficiency of CG depends heavily on the effectiveness of the stopping rule employed. In this paper we focus our attention on the *Generalized Cross-Validation* rule (GCV) [7,16,17], which is a widely used stopping technique and has the advantage over other popular stopping rules like the discrepancy principle or the UPRE, of not requiring information on the noise variance.

The stopping index is estimated through the minimum of the GCV function, whose denominator requires the computation of the trace of the CG influence matrix. GCV has been shown to be very effective when applied to iterative methods whose influence matrix does not depend on the noise, i.e. when the regularized solution depends linearly on the right-hand side of the system. However, this is not the case of CG, and some techniques have been proposed to overcome this drawback [4,6,7,15]. In order to approximate the denominator of the GCV function, we propose a new method which linearizes the dependence of the regularized solution on the noise as suggested in [15]. Instead of approximating the required derivatives by finite differences, we compute them by means of iterative formulas, along the lines of [1,13]. Our aim is to compare the effectiveness of this method with other ones proposed in the literature through an extensive numerical experimentation both on 1D and on 2D test problems.

The outline of the paper is the following: preliminary definitions and the GCV function are given in Section 2. In Section 3 the CG code is recalled in order to derive the expressions used for computing the trace of the influence matrix. Unless the matrix A has some special structure, the direct application of these expressions is impracticable for large dimensions, so a stochastic implementation based on the trace lemma is given. The special case of a circulant matrix A is examined in Section 4. The numerical experimentation, described in Section 5, shows that the different approximations of the denominator of the GCV function are in general not very critical in detecting an acceptable stopping index. Anyway, a reasonable ranking of them can be obtained for the examples we consider.

Throughout the paper, $\boldsymbol{\eta}$ is assumed to be an uncorrelated Gaussian white noise, i.e. with distribution $\mathcal{N}(0, \sigma^2 I)$, and $\|\boldsymbol{v}\|$ denotes the Euclidean norm of a vector $\boldsymbol{v}$.

2 The regularized solution

Let $A = U\Sigma V^T$ be the singular value decomposition of A, where $U = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n]$ $\in \mathbf{R}^{n\times n}$ and $V = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n] \in \mathbf{R}^{n\times n}$ have orthonormal columns, i.e. $U^T U =$

$V^TV = I$, and $\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_n)$, where the σ_i for $i = 1, \ldots, n$ are the singular values of A, gradually decaying toward zero. In practice, the last ones settle to values of the same magnitude of the machine precision.

The expansions of $\boldsymbol{b}^*$ and $\boldsymbol{\eta}$ in the basis U are

$$\boldsymbol{b}^* = \sum_i b_i^* \, \boldsymbol{u}_i, \quad \boldsymbol{\eta} = \sum_i \eta_i \, \boldsymbol{u}_i, \quad \text{where} \quad b_i^* = \boldsymbol{u}_i^T \boldsymbol{b}^*, \quad \eta_i = \boldsymbol{u}_i^T \boldsymbol{\eta}.$$

Then

$$\boldsymbol{x}^* = A^{-1}\boldsymbol{b}^* = \sum_i x_i^* \, \boldsymbol{v}_i, \quad \text{where} \quad x_i^* = \frac{b_i^*}{\sigma_i}, \tag{2}$$

and

$$\widetilde{\boldsymbol{x}} = \boldsymbol{x}^* + A^{-1}\boldsymbol{\eta} = \boldsymbol{x}^* + \sum_i \frac{\eta_i}{\sigma_i} \, \boldsymbol{v}_i = \sum_i \widetilde{x}_i \, \boldsymbol{v}_i, \quad \text{where} \quad \widetilde{x}_i = x_i^* + \frac{\eta_i}{\sigma_i}. \tag{3}$$

The coefficients η_i are typically of the same order for all i, with $|\eta_i| \sim \|\boldsymbol{\eta}\|/n$. If the last σ_i's are much smaller than the corresponding $|\eta_i|$, the quantities η_i/σ_i greatly increase with i. It follows that the low-frequency components of $\widetilde{\boldsymbol{x}}$ and $\boldsymbol{x}^*$ do not differ much, while the high-frequency components of $\widetilde{\boldsymbol{x}}$ are disastrously dominated by the high-frequency components of the noise and $\widetilde{\boldsymbol{x}}$ can be affected by a large error with respect to $\boldsymbol{x}^*$. The contribution of the high-frequency components of the noise should be damped in the regularized solution. Acceptable approximations can be obtained only if the $|b_i^*|$ decay to zero with i faster than the corresponding σ_i (this condition is known as Picard discrete condition), at least until the numerical levelling of the singular values.

The vector $\boldsymbol{x}_k$, computed at the kth step of the iterative method used to solve (1), can be considered as a regularized solution with the index k acting as the regularization parameter. We write the expansion of $\boldsymbol{x}_k$ in the basis V in the form

$$\boldsymbol{x}_k = \sum_i \varphi_{k,i} \, \widetilde{x}_i \, \boldsymbol{v}_i, \tag{4}$$

where the coefficients $\varphi_{k,i}$ are called *filter factors* at the kth step. The *solution error* of $\boldsymbol{x}_k$ is

$$\boldsymbol{\epsilon}_k = \boldsymbol{x}^* - \boldsymbol{x}_k = \sum_i \left((1 - \varphi_{k,i}) \, \frac{b_i^*}{\sigma_i} - \varphi_{k,i} \, \frac{\eta_i}{\sigma_i} \right) \boldsymbol{v}_i. \tag{5}$$

Let k_{opt} be the index which minimizes $\|\boldsymbol{\epsilon}_k\|$. Ideally, the iteration should be stopped at the k_{opt}th step, but the solution error is not directly computable.

If the iteration is stopped after the k_{opt}th step, too many filter factors would approach 1 and too many high-frequency components of the noise would enter, with a quick raise of $\|\boldsymbol{\epsilon}_k\|$. An approximation of k_{opt} can be sought through the minimum k_π of the *predictive* error

$$\boldsymbol{\pi}_k = A\boldsymbol{x}^* - A\boldsymbol{x}_k = \sum\nolimits_i \left((1-\varphi_{k,i})\, b_i^* - \varphi_{k,i}\, \eta_i\right) \boldsymbol{u}_i. \tag{6}$$

The main difference between minimizing the solution error and the predictive error is that the increase of $\|\boldsymbol{\pi}_k\|$ after the k_πth step is very weak, because in (6) the coefficients η_i are not divided by small σ_i. Figure 1 illustrates a typical case. It is obtained by applying CG to a 1D problem taken from [8], considered in the experimentation of Section 5. The right-hand side has been contaminated by white Gaussian noise $\boldsymbol{\eta}$ with relative level $\|\boldsymbol{\eta}\|/\|\boldsymbol{b}^*\| = 3.2\%$. The log plot of the histories $\|\boldsymbol{\epsilon}_k\|$ (solid line) and $\|\boldsymbol{\pi}_k\|$ (dashed line) varying

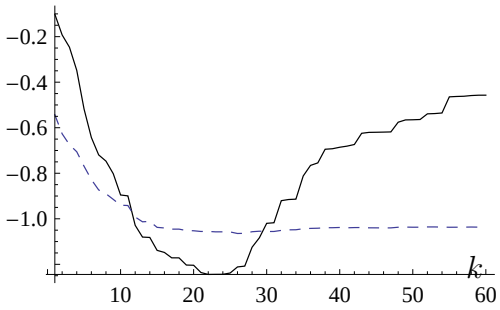

Fig. 1. Histories $\|\boldsymbol{\epsilon}_k\|$ (solid line) and $\|\boldsymbol{\pi}_k\|$ (dashed line).

k are plotted. The curve of $\|\boldsymbol{\pi}_k\|$ displays a characteristic L-shape, with a corner separating a nearly vertical part from a nearly horizontal part. The flat behavior of this second part implies that the determination of the right index through the minimum of $\|\boldsymbol{\pi}_k\|$ may be subjected to numerical difficulties. An exhaustive comparison between the behaviors of $\|\boldsymbol{\epsilon}_k\|$ and $\|\boldsymbol{\pi}_k\|$ will be made in Section 5.

Naturally, also the predictive error is not directly computable. However, methods (like UPRE and GCV) have been suggested to estimate it stochastically through the norm of the residual vector $\boldsymbol{r}_k = \boldsymbol{b} - A\boldsymbol{x}_k$, which is an available quantity. The *Generalized Cross-Validation* (GCV) approach, introduced in [5,17], is a popular method for practical problems with discrete data and stochastic noise. It derives from the ordinary cross-validation method, which considers all the leave-one-out regularized solutions and chooses the parameter that minimizes the average of the squared prediction errors in using each solution to predict the missing data value. By applying a suitable weighting, Wahba [17] derived the GCV method, which has the advantage of being invariant under orthogonal transformations of the data.

Usually the vector $\boldsymbol{x}_k$ is expressed through a regularization operator R_k in the

form

$$\boldsymbol{x}_k = R_k(\boldsymbol{b}).$$

In the original formulation [17], GCV has been defined for an operator R_k such that $R_k(\boldsymbol{b}) = R_k\,\boldsymbol{b}$, with R_k not depending on $\boldsymbol{b}$, i.e. when the regularized solution depends linearly on the right-hand side of the system. The matrix $A_k = AR_k$ is called *influence matrix* and describes how well the vector $\boldsymbol{x}_k$ predicts the right-hand side $\boldsymbol{b}$, i.e. $A\boldsymbol{x}_k = A_k\boldsymbol{b}$. For this case the GCV function is defined as

$$V(k) = \frac{n\|\boldsymbol{r}_k\|^2}{\left(\mathrm{tr}\,(I - A_k)\right)^2}, \tag{7}$$

and represents an estimate of the mean predictive error $\|\boldsymbol{\pi}_k\|^2/n$. The minimizer of $V(k)$ estimates k_π and can be used as an approximation of k_{opt}. From a statistical point of view the quantity $\mathrm{tr}\,(I - A_k)$ in the denominator of (7) can be considered as the effective number of degrees of freedom.

Subsequently, GCV has been applied as a stopping rule also in connection with iterative methods for which the regularized solution does not depend linearly on the right-hand side, like the CG [4,6,7,15].

3 Approximating the trace of the influence matrix for CG

When A is not a symmetric positive definite matrix, CG is applied to the *normal equations*

$$M\boldsymbol{x} = A^T\boldsymbol{b}, \quad \text{where} \quad M = A^TA, \tag{8}$$

and takes on the name CGLS, but for simplicity we continue to use here the name CG. In its essential form, CG computes $\boldsymbol{x}_k$ recursively, using two auxiliary vectors $\boldsymbol{p}_j$ (the search direction) and $\boldsymbol{q}_j = A^T\boldsymbol{b} - M\boldsymbol{x}_j$ (the residual vector of $\boldsymbol{x}_j$ in system (8)) and two scalars α_j and β_j for $j = 1, \ldots, k$. At the

jth step the algorithm computes

$$
\begin{aligned}
&\boldsymbol{z}_{j-1} = A\boldsymbol{p}_{j-1},\\
&\alpha_{j-1} = \|\boldsymbol{q}_{j-1}\|^2/\|\boldsymbol{z}_{j-1}\|^2,\\
&\boldsymbol{x}_j = \boldsymbol{x}_{j-1} + \alpha_{j-1}\,\boldsymbol{p}_{j-1},\\
&\boldsymbol{r}_j = \boldsymbol{b} - A\,\boldsymbol{x}_j,\\
&\boldsymbol{q}_j = A^T\boldsymbol{r}_j,\\
&\beta_{j-1} = \|\boldsymbol{q}_j\|^2/\|\boldsymbol{q}_{j-1}\|^2,\\
&\boldsymbol{p}_j = \boldsymbol{q}_j + \beta_{j-1}\boldsymbol{p}_{j-1}.
\end{aligned}
\tag{9}
$$

The initial positions are $\boldsymbol{x}_0 = \boldsymbol{0}$ and $\boldsymbol{p}_0 = \boldsymbol{q}_0 = A^T\boldsymbol{b}$. The recursive computation of $\boldsymbol{q}_j$ with $\boldsymbol{q}_j = \boldsymbol{q}_{j-1} - \alpha_{j-1}\,M\boldsymbol{p}_{j-1}$ is often suggested for reducing the computational cost, but we prefer to use here the original more stable definition.

We examine now three different approaches to compute the trace of A_k required by the denominator of the GCV function (7).

The filter approach. CG is a projection method on the Krylov subspace

$$\mathcal{K}_k(M, \boldsymbol{q}_0) = \mathrm{span}(\boldsymbol{q}_0, M\,\boldsymbol{q}_0, M^2\,\boldsymbol{q}_0, \ldots, M^{k-1}\,\boldsymbol{q}_0),$$

i.e. $\boldsymbol{x}_k$ minimizes $\phi(\boldsymbol{x}) = \|\boldsymbol{b} - A\boldsymbol{x}\|^2$ on $\mathcal{K}_k(M, \boldsymbol{q}_0)$ and $\boldsymbol{q}_k$ is orthogonal to $\mathcal{K}_k(M, \boldsymbol{q}_0)$. An orthonormal basis of $\mathcal{K}_k(M, \boldsymbol{q}_0)$ can be obtained by applying k steps of the Lanczos tridiagonalization algorithm to the symmetric positive definite matrix M. Starting with the vector $\boldsymbol{w}_1 = \boldsymbol{q}_0/||\boldsymbol{q}_0||$, the algorithm produces the orthonormal sequence $\boldsymbol{w}_j = (-1)^{j-1}\boldsymbol{q}_{j-1}/\|\boldsymbol{q}_{j-1}\|$ and the scalars γ_j and δ_j, $j = 1, \ldots, k$, such that the $n \times k$ matrix $W_k = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_k]$ and the $k \times k$ symmetric matrix $T = \mathrm{tridiag}(\delta_j, \gamma_j, \delta_{j+1})$ verify $T_k = W_k^T M W_k$. T_k is viewed as the representation of M projected on $\mathcal{K}_k(M, \boldsymbol{q}_0)$. Its eigenvalues $\theta_{k,1}, \ldots, \theta_{k,k}$ are called the *Ritz values* of M at the kth step and converge to the singular values σ_i when $k \to \infty$. The filter factors of CG are shown [7] to have the form

$$\varphi_{k,i} = 1 - \prod_{j=1}^{k} (1 - \frac{\sigma_i^2}{\theta_{k,j}}).$$

From (4) we have

$$\boldsymbol{x}_k = V\Phi_k\Sigma^{-1}U^T\boldsymbol{b}, \quad \text{where} \quad \Phi_k = \mathrm{diag}(\varphi_{k,1},\ldots,\varphi_{k,n}),$$

then

$$A_k = U\Phi_k U^T \quad \text{and} \quad \mathrm{tr}\, A_k = \sum_{i=1}^{n}\varphi_{k,i}. \tag{10}$$

Formula (10) gives a theoretical means for computing the denominator of (7), but is impracticable from a computational point of view when n is large because it requires the knowledge of the singular values of A. For this reason, formula (10) will not be taken into consideration in the experimentation of Section 5.

The KA approach. Another theoretical expression for $\mathrm{tr}\, A_k$ is obtained by recalling that

$$\boldsymbol{x}_k = P_k\,\widetilde{\boldsymbol{x}},$$

where $P_k = W_k(W_k^TMW_k)^{-1}W_k^TM$ is the projector on $\mathcal{K}_k(M,\boldsymbol{q}_0)$, orthogonal with respect to the inner product $<\ \cdot\ ,\ \cdot\ >_M$. Then

$$\boldsymbol{x}_k = W_k(W_k^TMW_k)^{-1}W_k^TA^T\,\boldsymbol{b}. \tag{11}$$

The influence matrix is

$$A_k = AW_k(W_k^TA^TAW_k)^{-1}W_k^TA^T = AW_k(AW_k)^{\dagger}.$$

Denoting by $AW_k = \widehat{U}\widehat{\Sigma}\widehat{V}^T$ the singular value decomposition of AW_k, we have $A_k = \widehat{U}\widehat{\Sigma}\widehat{\Sigma}^{\dagger}\widehat{U}^T$. Then, since AW_k has full rank,

$$\mathrm{tr}\, A_k = k. \tag{12}$$

For its simplicity, formula (12) is appealing. But the use of this formula, which in Section 5 will be identified as KA approach, is arguable when k_{opt} is not small compared with n, because the vectors $\boldsymbol{q}_j$ computed by CG in finite arithmetic loose their orthogonality properties after few steps, leading to nonorthogonal bases W_k. Moreover, with increasing of k multiple approximations of the σ_i are generated. It follows that, even in the case that the Lanczos process approximates the singular values in their natural order, the difference between k and the number of the converged Ritz values increases with k and

the quantity $n-k$ in general cannot be considered as the correct degrees of freedom for GCV.

The approach based on linearization. Since the influence matrix of CG depends on $\boldsymbol{b}$, in [15] it is suggested to replace A_k by $\widehat{A}_k = A\,J_k^*$, where $J_k^* = J_k(\boldsymbol{b}^*)$ is the Jacobian matrix of $R_k(\boldsymbol{b})$ respect to $\boldsymbol{b}$, evaluated in $\boldsymbol{b}^*$. A further approximation is required in practice: replace J_k^* by J_k, the Jacobian matrix in $\boldsymbol{b}$, that is the matrix of the derivatives of the vector $\boldsymbol{x}_k$ effectively computed by the iterative method. We propose to evaluate $\operatorname{tr}\widehat{A}_k$ by an iterative technique along the lines suggested in [1,13] for other iterative methods. Using the notations

$$\widehat{A}_j = A\Big[\frac{\partial(\boldsymbol{x}_j)_i}{\partial b_h}\Big], \quad \widehat{P}_j = A\Big[\frac{\partial(\boldsymbol{p}_j)_i}{\partial b_h}\Big], \quad \widehat{Q}_j = A\Big[\frac{\partial(\boldsymbol{q}_j)_i}{\partial b_h}\Big],$$

from (9) we have

$$\begin{aligned}
&\widehat{A}_j = \widehat{A}_{j-1} + \alpha_{j-1}\,\widehat{P}_{j-1} + \boldsymbol{z}_{j-1}\boldsymbol{\zeta}_{j-1}^T, \quad \text{where} \quad (\boldsymbol{\zeta}_{j-1})_h = \frac{\partial \alpha_{j-1}}{\partial b_h},\\
&\widehat{Q}_j = AA^T(I - \widehat{A}_j),\\
&\widehat{P}_j = \widehat{Q}_j + \beta_{j-1}\,\widehat{P}_{j-1} + \boldsymbol{z}_{j-1}\boldsymbol{\xi}_{j-1}^T, \quad \text{where} \quad (\boldsymbol{\xi}_{j-1})_h = \frac{\partial \beta_{j-1}}{\partial b_h},\\
&\boldsymbol{\zeta}_j = 2\,\alpha_j\,\Big(\widehat{Q}_j^T\frac{\boldsymbol{r}_j}{\|\boldsymbol{q}_j\|^2} - \widehat{P}_j^T\frac{\boldsymbol{z}_j}{\|\boldsymbol{z}_j\|^2}\Big),\\
&\boldsymbol{\xi}_j = 2\,\beta_j\,\Big(\widehat{Q}_{j+1}^T\frac{\boldsymbol{r}_{j+1}}{\|\boldsymbol{q}_{j+1}\|^2} - \widehat{Q}_j^T\frac{\boldsymbol{r}_j}{\|\boldsymbol{q}_j\|^2}\Big).
\end{aligned} \tag{13}$$

The initial positions are $\widehat{A}_0 = O$, $\widehat{P}_0 = \widehat{Q}_0 = AA^T$. In the following this "complete derivative" approach will be called CD approach.

The computation of formulas (13) cannot be restricted to the single traces of the matrices, because the whole $\widehat{A}_j$ is required to compute the trace of $\widehat{Q}_j$. So, the use of (13) for the direct computation of $\operatorname{tr}\widehat{A}_k$ is impracticable for large n and we resort to a randomized approach which exploits the following

Trace Lemma: Given a matrix B of size n and a vector $\boldsymbol{v}$ of n independent samples of a normal random variable with zero mean and variance σ^2, then $\sigma^2 \operatorname{tr} B = \mathcal{E}(\boldsymbol{v}^T B\boldsymbol{v})$.

By applying this lemma to matrix $\widehat{A}_k$ and to a vector $\boldsymbol{v}$ of n independent samples of a normal random variable with zero mean and unit variance, we have

$$\operatorname{tr}\widehat{A}_k = \mathcal{E}(\boldsymbol{v}^T\,\boldsymbol{x}_k'), \quad \text{where} \quad \boldsymbol{x}_k' = AJ_k\boldsymbol{v} = \widehat{A}_k\boldsymbol{v}. \tag{14}$$

The vector $\boldsymbol{x}'_k$ can be computed recursively. Setting $\boldsymbol{x}'_j = \widehat{A}_j \boldsymbol{v}$, $\boldsymbol{p}'_j = \widehat{P}_j \boldsymbol{v}$, $\boldsymbol{q}'_j = \widehat{Q}_j \boldsymbol{v}$, $\zeta'_j = \boldsymbol{\zeta}_j^T \boldsymbol{v}$ and $\xi'_j = \boldsymbol{\xi}_j^T \boldsymbol{v}$, from (13) we have

$$\begin{aligned}
&\boldsymbol{x}'_j = \boldsymbol{x}'_{j-1} + \alpha_{j-1}\, \boldsymbol{p}'_{j-1} + \zeta'_{j-1} \boldsymbol{z}_{j-1}, \\
&\boldsymbol{q}'_j = AA^T(\boldsymbol{v} - \boldsymbol{x}'_j), \\
&\boldsymbol{p}'_j = \boldsymbol{q}'_j + \beta_{j-1}\, \boldsymbol{p}'_{j-1} + \xi'_{j-1} \boldsymbol{z}_{j-1}, \\
&\zeta'_j = 2\, \alpha_j \Big(\frac{\boldsymbol{r}_j^T \boldsymbol{q}'_j}{\|\boldsymbol{q}_j\|^2} - \frac{\boldsymbol{z}_j^T \boldsymbol{p}'_j}{\|\boldsymbol{z}_j\|^2} \Big), \\
&\xi'_j = 2\, \beta_j \Big(\frac{\boldsymbol{r}_{j+1}^T \boldsymbol{q}'_{j+1}}{\|\boldsymbol{q}_{j+1}\|^2} - \frac{\boldsymbol{r}_j^T \boldsymbol{q}'_j}{\|\boldsymbol{q}_j\|^2} \Big).
\end{aligned} \tag{15}$$

The initial positions are $\boldsymbol{x}'_0 = \boldsymbol{0}$, $\boldsymbol{p}'_0 = \boldsymbol{q}'_0 = AA^T \boldsymbol{v}$. The computation is carried out with the following procedure.

Procedure P_1: m independent realizations of a vector $\boldsymbol{v}$ with distribution $\mathcal{N} = \mathcal{N}(0, I)$ are generated and the trace of $\widehat{A}_k$ is approximated by averaging the m values $\boldsymbol{v}^T \boldsymbol{x}'_k$ obtained applying (15). In [4] it is shown that when n is large enough and suitable hypotheses on the distribution of the eigenvalues of the matrix are satisfied, only few vectors $\boldsymbol{v}$ are needed to get a reliable estimate. In [10] it is shown that the variance of this estimate is minimized when the components of $\boldsymbol{v}$ are generated independently and take on the values $+1$ and -1 with equal probability $1/2$. In the following we denote this distribution by $\mathcal{U}$.

The magnitudes of the vectors $\boldsymbol{\zeta}_j$ and $\boldsymbol{\xi}_j$ for $j = 1, \ldots, k-1$ measure how much A_k depends on $\boldsymbol{b}$. If these magnitudes are sufficiently small, we may assume that the dependence of α_j and β_j on $\boldsymbol{b}$ is weak and enables us to ignore it. In this "incomplete derivative" approach, which in the following will be called ID approach, A_k is approximated by the matrix $\widehat{A}_k$, obtained setting in (13) $\boldsymbol{\zeta}_j$ and $\boldsymbol{\xi}_j$ to $\boldsymbol{0}$, i.e.

$$\widehat{A}_j = \widehat{A}_{j-1} + \alpha_{j-1}\, \widehat{P}_{j-1}, \quad \widehat{Q}_j = AA^T(I - \widehat{A}_j), \quad \widehat{P}_j = \widehat{Q}_j + \beta_{j-1}\, \widehat{P}_{j-1}, \tag{16}$$

with initial positions $\widehat{A}_0 = O$, $\widehat{P}_0 = \widehat{Q}_0 = AA^T$. The vector $\boldsymbol{x}'_k$ required in (14) is now computed by

$$\boldsymbol{x}'_j = \boldsymbol{x}'_{j-1} + \alpha_{j-1}\, \boldsymbol{p}'_{j-1}, \quad \boldsymbol{q}'_j = AA^T(\boldsymbol{v} - \boldsymbol{x}'_j), \quad \boldsymbol{p}'_j = \boldsymbol{q}'_j + \beta_{j-1}\, \boldsymbol{p}'_{j-1}. \tag{17}$$

The computation is carried out according to

Procedure P_2: it works like P_1 with relations (15) replaced by (17). Having used in recurrences (17) the same α_j and β_j of (9), it is immediate to see that the vector $\boldsymbol{x}'_k$ so obtained is equal to $A\boldsymbol{x}_k$, where $\boldsymbol{x}_k$ is the vector we would obtain by applying CG to system (8) with right-hand side $A^T\boldsymbol{v}$. Then this approach coincides with the *Monte-Carlo gcv* approach, suggested in [4].

We consider also a third procedure, proposed in [15], which computes $\boldsymbol{x}'_k$ in (14) by replacing the derivatives used in the Jacobian with finite difference approximations.

Procedure P_3: denoting by $x_k(\boldsymbol{b})$ the kth vector obtained by applying CG to right-hand side $A^T\boldsymbol{b}$, we have

$$J_k\boldsymbol{v} \sim \frac{\boldsymbol{x}_k(\boldsymbol{b}+\delta\,\boldsymbol{v}) - \boldsymbol{x}_k(\boldsymbol{b}-\delta\,\boldsymbol{v})}{2\delta},$$

where δ is a small constant. At each step of CG the three vectors $\boldsymbol{x}_j(\boldsymbol{b})$, $\boldsymbol{x}_j(\boldsymbol{b}+\delta\boldsymbol{v})$ and $\boldsymbol{x}_j(\boldsymbol{b}-\delta\boldsymbol{v})$ are computed. The truncation error of a central approximation is of the second order, but we must take into consideration also the influence of the noise. Then it is not worthwhile to choose δ too small. Moreover in [7] it is observed that this approach can be very sensitive to the choice of δ.

Taking into account the computational cost, the two procedures P_1 and P_2 can be considered equivalent, in the sense that the computation of ζ'_j and ξ'_j which are present in (15) but not in (17) do not alter the cost, dominated by the matrix-vector product for $\boldsymbol{q}'_j$. Procedure P_3 doubles the computational cost. Of course, the cost of the three procedures increases with the number m of random vectors $\boldsymbol{v}$. We expect that the larger m the better estimate of $\operatorname{tr}\widehat{A}_k$, but it remains to be established if this leads to a substantially improvement in the reconstructed solution.

In Section 5, we will examine how much the use of GCV as a stopping rule for CG is influenced by these different ways of approximating the trace in the denominator of (7). As already noted, when n is large an effective implementation of formulas (13) and (16) is in general too demanding in terms of computational cost. In Section 5 we will use them as a theoretical basis for testing the effectiveness of procedures $P_1 - P_3$.

4 The case of the circulant matrix

There are problems where a spectral decomposition of A is available at a low cost also for large n. This is true, for example, in the case of the restoration of images blurred by a shift-invariant PSF. In this case, setting suitable boundary conditions, the matrix A has a 2D circulant structure and its eigenvalues can be computed by the use of FFT, as we see presently.

Let $\boldsymbol{v}$ be a vector of size n and A the $n \times n$ circulant matrix whose first row is $\boldsymbol{a}^T$. Circulant matrices of size n are diagonalized by the Fourier matrix $\mathcal{F}$, whose elements are

$$f_{r,s} = \frac{1}{\sqrt{n}}\,\omega^{rs}, \quad r,s = 0,\ldots,n-1, \quad \text{with} \quad \omega = \exp(2\pi \boldsymbol{i}/n),$$

i.e. $A = \sqrt{n}\,\mathcal{F}\,\mathrm{diag}(\mathcal{F}\,\boldsymbol{a})\,\mathcal{F}^*$. Then the product $\boldsymbol{z} = A\boldsymbol{v}$ can be so computed

$$\widetilde{\boldsymbol{a}} = \mathcal{F}\boldsymbol{a}, \quad \widetilde{\boldsymbol{v}} = \mathcal{F}\boldsymbol{v}, \quad \boldsymbol{t} = \widetilde{\boldsymbol{r}} \odot \widetilde{\boldsymbol{v}}^*, \quad \boldsymbol{z} = \sqrt{n}\,\mathcal{F}\,\boldsymbol{t},$$

where $\odot$ indicates the element-wise product of two vectors of the same size and $\widetilde{\boldsymbol{v}}^{\,*}$ is the conjugate of $\widetilde{\boldsymbol{v}}$. To obtain the product $\boldsymbol{z} = A^T\boldsymbol{v}$ it is sufficient to take the first column of A as $\boldsymbol{a}$ or to replace $\widetilde{\boldsymbol{a}}$ by its conjugate $\widetilde{\boldsymbol{a}}^{\,*}$. The multiplications by $\mathcal{F}$ and $\mathcal{F}^*$ can be efficiently computed by calling two FFT routines, with a computational cost of order $O(n \log n)$.

A substantial reduction of the cost when A is circulant is obtained by shifting to the Fourier domain, i.e. by replacing the matrices and vectors involved in the computation with their transformations in the Fourier basis. Let $\boldsymbol{a}^T$ be the first row of A, $\widetilde{\boldsymbol{a}} = \mathcal{F}\boldsymbol{a}$, $\widetilde{\boldsymbol{b}} = \mathcal{F}\boldsymbol{b}$, $\widetilde{\boldsymbol{c}} = \sqrt{n}\;\widetilde{\boldsymbol{a}} \odot \widetilde{\boldsymbol{b}}$ and $\widetilde{\boldsymbol{d}} = n\;\widetilde{\boldsymbol{a}}^{\,*} \odot \widetilde{\boldsymbol{a}}$. The CG algorithm becomes

$$\begin{aligned}
&\widetilde{\boldsymbol{z}}_{j-1} = \sqrt{n}\;\widetilde{\boldsymbol{a}} \odot \widetilde{\boldsymbol{p}}_{j-1},\\
&\alpha_{j-1} = \|\widetilde{\boldsymbol{q}}_{j-1}\|^2 / \|\widetilde{\boldsymbol{z}}_{j-1}\|^2,\\
&\widetilde{\boldsymbol{x}}_j = \widetilde{\boldsymbol{x}}_{j-1} + \alpha_{j-1}\,\widetilde{\boldsymbol{p}}_{j-1},\\
&\widetilde{\boldsymbol{q}}_j = \widetilde{\boldsymbol{c}} - \widetilde{\boldsymbol{d}} \odot \widetilde{\boldsymbol{x}}_j,\\
&\beta_{j-1} = \|\widetilde{\boldsymbol{q}}_j\|^2 / \|\widetilde{\boldsymbol{q}}_{j-1}\|^2,\\
&\widetilde{\boldsymbol{p}}_j = \widetilde{\boldsymbol{q}}_j + \beta_{j-1}\widetilde{\boldsymbol{p}}_{j-1}.
\end{aligned} \tag{18}$$

The initial positions are $\widetilde{\boldsymbol{x}}_0 = \boldsymbol{0}$ and $\widetilde{\boldsymbol{p}}_0 = \widetilde{\boldsymbol{q}}_0 = \widetilde{\boldsymbol{c}}$. At the end $\boldsymbol{x}_k = \mathcal{F}^*\widetilde{\boldsymbol{x}}_k$.

The cost per iteration is now reduced to the order of $O(n)$. If relations (16) are used, the computation of $\mathrm{tr}A_k$ can be easily performed, because $\widehat{A}_j$, $\widehat{P}_j$ and $\widehat{Q}_j$ still keep the circulant structure. Denoting $\widetilde{\boldsymbol{a}}_j = \mathcal{F}\,\widehat{\boldsymbol{a}}_j$, $\widetilde{\boldsymbol{p}}_j = \mathcal{F}\,\widehat{\boldsymbol{p}}_j$ and $\widetilde{\boldsymbol{q}}_j = \mathcal{F}\,\widehat{\boldsymbol{q}}_j$ where $\widehat{\boldsymbol{a}}_j^T$, $\widehat{\boldsymbol{p}}_j^T$ and $\widehat{\boldsymbol{q}}_j^T$ are the first rows of $\widehat{A}_j$, $\widehat{P}_j$ and $\widehat{Q}_j$, we have

$$\widetilde{\boldsymbol{a}}_j = \widetilde{\boldsymbol{a}}_{j-1} + \alpha_{j-1}\widetilde{\boldsymbol{p}}_{j-1}, \quad \widetilde{\boldsymbol{q}}_j = \widetilde{\boldsymbol{a}}^{\,*} - \widetilde{\boldsymbol{d}} \odot \widetilde{\boldsymbol{a}}_j, \quad \widetilde{\boldsymbol{p}}_j = \widetilde{\boldsymbol{q}}_j + \beta_{j-1}\widetilde{\boldsymbol{p}}_{j-1},$$

with initial positions $\widetilde{\boldsymbol{a}}_0 = \boldsymbol{0}$, $\widetilde{\boldsymbol{p}}_0 = \widetilde{\boldsymbol{q}}_0 = \widetilde{\boldsymbol{a}} \odot \widetilde{\boldsymbol{a}}^{\,*}$ and $\mathrm{tr}\,\widehat{A}_k = n\,\sum_i (\widetilde{\boldsymbol{a}}_k)_i$. Then

$$V(k) = \frac{n\,\|\widetilde{\boldsymbol{r}}_k\|^2}{(n - \mathrm{tr}\,\widehat{A}_k)^2}\,, \quad \text{where} \quad \widetilde{\boldsymbol{r}}_k = \widetilde{\boldsymbol{b}}_k - \sqrt{n}\;\widetilde{\boldsymbol{a}}^{\,*} \odot \widetilde{\boldsymbol{x}}_k.$$

Analogous formulas can be derived for the 2-level circulant matrices, which frequently occur in 2D problems of image deconvolution.

Unfortunately, such a forward shifting to the Fourier domain does not hold for relations (13), because $\boldsymbol{z}_{j-1}\boldsymbol{\zeta}_{j-1}^T$ and $\boldsymbol{z}_{j-1}\boldsymbol{\xi}_{j-1}^T$ are not circulant.

5 Numerical experimentation

The numerical experimentation is carried out in *Mathematica* with machine-epsilon 2^{-53} arithmetic both on 1D and on 2D test problems. For each problem the matrix A and the solution $\boldsymbol{x}^*$ are given, then the vector $\boldsymbol{b}^* = A\boldsymbol{x}^*$ is computed and white Gaussian noises are generated, with relative magnitudes $\|\boldsymbol{\eta}\|/\|\boldsymbol{b}^*\|$ ranging from 0.5% to 10%.

CG method is applied to each sample, i.e. a triple (A, $\boldsymbol{b}^*$, $\boldsymbol{\eta}$), and the optimum index k_{opt} is derived from the history $\|\boldsymbol{\epsilon}_k\|$. For any k the ratio

$$E(k) = \frac{\|\boldsymbol{\epsilon}_k\| - \|\boldsymbol{\epsilon}_{k_{opt}}\|}{\|\boldsymbol{\epsilon}_{k_{opt}}\|} \tag{19}$$

of the solution error (5) of $\boldsymbol{x}_k$ with respect to the optimal error is considered. If taken as the regularized solution, $\boldsymbol{x}_k$ is oversmoothed if $k < k_{opt}$ and undersmoothed if $k > k_{opt}$. Its level of smoothing is judged by comparing $E(k)$ with a preassigned value ℓ. If $E(k) \le \ell$ then $\boldsymbol{x}_k$ is called acceptable, if $\ell < E(k) \le 10\,\ell$ then $\boldsymbol{x}_k$ is called mildly over- undersmoothed, if $E(k) > 10\,\ell$ then $\boldsymbol{x}_k$ is called severely over- undersmoothed. Since our aim is to evaluate how the different procedures described in Section 3 for estimating $\mathrm{tr}A_k$ affect the GCV efficiency, the chosen value of ℓ should allow to establish a clear performance ranking of the procedures.

For a deeper understanding of the numerical results, another parameter should be taken into consideration, namely the difficulty of the various problems. From a theoretical point of view, we expect this difficulty to depend on the decay of the singular values of A and the spectral components of the vectors $\boldsymbol{b}$ and $\boldsymbol{\eta}$. But to measure it with specific regards to the regularization by means of CG, we think that formula (19) may serve the purpose through the ratios $E(k_{opt}-1)$ and $E(k_{opt}+1)$ which monitor the consequence of missing the optimal index by only one.

Anyway, we recall that in practice finding the minimum of $V(k)$ might fail to detect a proper estimate of k_{opt} because of the nearly flat behavior of $V(k)$ followed by multiple local minima due to the noise and the accumulation of the rounding errors which can transform a slowly decreasing behavior into a slowly increasing one or viceversa.

5.1 1D experimentation

The 1D experimentation concerns some problems of [8] at size $n = 256$, namely `baart`, `deriv2`, `foxgood`, `heat` (with `kappa`= 1.1, 2, 3), `ilaplace` (the four listed examples), `phillips`, `shaw`, `wing`. The problems, which are obtained from the discretization of Fredholm integral equations of the first kind, have in general severely ill-conditioned matrices, with more than half the singular values below the machine precision. For these problems the acceptability level $\ell = 0.1$ is chosen.

The aim of a first set of experiments is the comparison of the behaviors of $\boldsymbol{\epsilon}_k$ and $\boldsymbol{\pi}_k$. So we apply CG to each sample and find the index k_π of the minimum of $\|\boldsymbol{\pi}_k\|$. The first line of Table 1 shows the percentages of acceptable cases, of severely or mildly oversmoothed cases and of severely or mildly undersmoothed cases over the total number of samples, according to the ratio $E(k)$ defined in (19) with $k = k_\pi$. It appears that the probability of getting acceptable reconstructions for our set of samples is sufficiently high when k_{opt} is estimated through k_π. The two following lines refer to the difficulty of the 1D problems, showing that a large percentage of problems can be classified as difficult. The last three lines of the table show the effect of the different approximations of the GCV function in detecting the stopping index. The comparison is made in a theoretical context using directly (12) (KA approach), (13) (CD approach), or (16) (ID approach).

The CD and ID approaches share a common behavior. This suggests that ignoring the dependence of A_k on $\boldsymbol{b}$ does not produce very serious effects. The KA approach does not behave too badly for these problems, where k_{opt} is in general small. In fact, a small number of iterations does not leave enough room

	sev. over.	mild over.	acceptable	mild under.	sev. under.
k_π	0%	2.3%	86.0%	10.0%	1.7%
$k_{opt}-1$	19.0%	40.6%	40.4%	0%	0%
$k_{opt}+1$	0%	0%	40.8%	29.2%	30.0%
KA	3.9%	20.2%	49.0%	9.6%	17.3%
CD	4.1%	15.4%	64.4%	11.9%	4.2%
ID	4.8%	16.4%	61.5%	13.1%	4.2%

Table 1
1D problems: percentages of cases of over- undersmoothing for the predictive error and the different approaches to the approximation of the GCV function with acceptability level $\ell = 0.1$.

for revealing the inherent instability of the approach.

Next, procedures P_1 and P_2 are applied with $m = 1$ and $m = 5$ random vectors $\boldsymbol{v}$ generated in both the normal and uniform distributions $\mathcal{N}$ and $\mathcal{U}$. Table 2 lists the percentages of acceptable cases, of severely or mildly oversmoothed cases and of severely or mildly undersmoothed cases.

	m	sev. over.	mild over.	acceptable	mild under.	sev. under.
$P_1\ \mathcal{N}$	1	5.0%	15.0%	61.6%	11.7%	6.7%
	5	3.5%	17.3%	67.9%	9.0%	2.3%
$P_1\ \mathcal{U}$	1	2.3%	16.2%	59.8%	10.0%	11.7%
	5	3.6%	14.8%	63.5%	11.0%	7.1%
$P_2\ \mathcal{N}$	1	5.4%	15.2%	61.3%	10.8%	7.3%
	5	3.1%	18.6%	65.4%	10.6%	2.3%
$P_2\ \mathcal{U}$	1	2.1%	17.9%	58.1%	10.6%	11.3%
	5	5.6%	14.2%	61.6%	12.1%	6.5%

Table 2
1D problems: percentages of cases of over- undersmoothing for procedures P_1 and P_2 with acceptability level $\ell = 0.1$.

We see that the two procedures P_1 and P_2 behave similarly, in confirmation of what seen in the theoretical context, i.e. that ignoring the dependence of A_k on $\boldsymbol{b}$ does not produce too serious effects. Anyway, taking into account that the computational costs of the two procedures is the same, the slight prevalence of P_1 suggests its use against P_2.

To evaluate the influence of the two distributions $\mathcal{N}$ and $\mathcal{U}$, we consider the figures in the acceptable column and the sum of the figures in the "sev. over." and "sev. under." columns. Choosing the random vectors in the distribution $\mathcal{N}$ produces better results.

Finally, we evaluate the effects of choosing just one random vector $\boldsymbol{v}$ or a larger number m of vectors $\boldsymbol{v}$, used for estimating the trace through (14). It is evident that the cheapest choice $m = 1$ cannot be suggested against the more computational expensive choice $m = 5$, because of the larger risk of undersmoothing.

Table 3 lists the analogous percentages for procedure P_3. Three values are given to δ, namely 0.1 (large), 0.01 (medium) and 0.001 (small). Also with this procedure $m = 1$ and $m = 5$ random vectors $\boldsymbol{v}$ are generated in the normal and uniform distributions $\mathcal{N}$ and $\mathcal{U}$, but the table shows only the results of $\mathcal{N}$ because of their appreciable superiority, in confirmation of what seen for procedures P_1 and P_2. Also with procedure P_3, the choice $m = 1$ gives the worst results. At first glance, it might seem that procedure P_3 has an overall performance comparable with that of procedure P_1. However, due to its irregular behavior varying δ and its larger computational cost, we do not feel like suggesting P_3 against P_1.

δ	m	sev. over.	mild over.	acceptable	mild under.	sev. under.
0.1	1	4.8%	17.9%	61.3%	10.0%	6.0%
	5	3.6%	20.6%	68.1%	5.4%	2.3%
0.01	1	5.4%	21.0%	59.6%	8.8%	5.2%
	5	3.3%	21.9%	67.5%	5.4%	1.9%
0.001	1	5.0%	16.5%	63.1%	9.6%	5.8%
	5	3.5%	19.6%	67.3%	6.9%	2.7%

Table 3
1D problems: percentages of cases of over- undersmoothing for procedure P_3 with acceptability level $\ell = 0.1$.

5.2 2D experimentation

The 2D experimentation deals with two images of astronomical interest (the spiral galaxy NGC 1288 [2] and an image of satellite [11]) and two of medical interest (the synthetic Shepp-Logan phantom [18] and a Hoffman phantom [9]), widely used in the literature for testing image deconvolution algorithms. All the images have $n = 256^2$ number of pixels. The matrix A which performs the blur is a 2-level circulant matrix generated by a positive space invariant bandlimited PSF with a bandwidth $\nu = 15$, normalized in such a way that the sum of the elements is equal to 1. We consider four exponential PSFs of the following types.

(a) Gaussian PSFs represented by masks with entries $m_{i,j} = \exp(-\alpha\, i^2 - \beta j^2)$, $-\nu \leq i, j \leq \nu$, where $\alpha = 0.3$, $\beta = 0.25$ and $\alpha = 0.1$, $\beta = 0.1$.

(b) Motion-type PSFs represented by masks with entries $m_{i,j} = \exp(-\alpha(i+j)^2 - \beta(i-j)^2)$, $-\nu \leq i, j \leq \nu$, where $\alpha = 0.02$, $\beta = 0.01$ and $\alpha = 0.04$, $\beta = 0.02$.

These 2D problems are less difficult than the 1D problems for what concerns the regularization by means of CG. Actually, with the same level $\ell = 0.1$ used for the 1D case, almost the totality of the problems would give acceptable results. Hence for these problems the lower acceptability level $\ell = 0.002$ is fixed. Due to the large dimension of the problems, CD approach is not applied and ID approach is implemented as described in Section 4, i.e. by exploiting the circulant structure. The results are shown in Table 4.

	sev. over.	mild over.	acceptable	mild under.	sev. under.
k_π	0%	7.3%	90.8%	1.9%	0%
$k_{opt} - 1$	6.7%	18.1%	75.2%	0%	0%
$k_{opt} + 1$	0%	0%	78.0%	12.0%	10.0%
KA	0%	0%	1.3%	2.8%	95.9%
ID	0%	8.3%	91.1%	0.6%	0%

Table 4
2D problems: percentages of cases of over- undersmoothing for the predictive error and the KA and ID approaches to the approximation of the GCV function with acceptability level $\ell = 0.002$.

The similarity of the percentages of the ID approach and those of k_π suggests that also in the 2D case ignoring the dependence of A_k on $\boldsymbol{b}$ does not produce very serious effects. On the contrary, here KA approach is not reliable, because in most cases the GCV function computed by setting $\mathrm{tr}A_k = k$ has no minimum.

Table 5 lists the percentages of the different cases obtained by applying procedures P_1 and P_2. The two procedures behave as with the 1D problems, except that the choice of the two distributions $\mathcal{N}$ and $\mathcal{U}$ appears now to be less significant.

Table 6 lists the analogous percentages for procedure P_3. Also for this procedure the choice of the two distributions $\mathcal{N}$ and $\mathcal{U}$ appears to be irrelevant, hence the table shows the results of $\mathcal{N}$. Unlike the 1D case, the percentage of acceptable cases increases for decreasing δ. Anyway, procedure P_3 is always outperformed by both P_1 and P_2.

	m	sev. over.	mild over.	acceptable	mild under.	sev. under.
P_1 $\mathcal{N}$	1	0%	9.7%	89.4%	0.9%	0%
	5	0%	8.5%	90.9%	0.6%	0%
P_1 $\mathcal{U}$	1	0%	9.1%	90.1%	0.8%	0%
	5	0%	9.4%	90.1%	0.5%	0%
P_2 $\mathcal{N}$	1	0%	9.8%	88.9%	1.3%	0%
	5	0%	8.8%	90.6%	0.6%	0%
P_2 $\mathcal{U}$	1	0%	9.2%	90.0%	0.8%	0%
	5	0%	9.5%	90.0%	0.5%	0%

Table 5
2D problems: percentages of cases of over- undersmoothing for procedures P_1 and P_2 with acceptability level $\ell = 0.002$.

δ	m	sev. over.	mild over.	acceptable	mild under.	sev. under.
0.1	1	2.5%	23.7%	66.9%	6.9%	0%
	5	2.5%	21.7%	71.3%	4.5%	0%
0.01	1	0%	23.1%	76.0%	0.9%	0%
	5	0%	22.2%	77.3%	0.5%	0%
0.001	1	0%	21.7%	77.4%	0.9%	0%
	5	0%	19.1%	80.3%	0.6%	0%

Table 6
2D problems: percentages of cases of over- undersmoothing for procedure P_3 with acceptability level $\ell = 0.002$.

6 Conclusions

Various techniques have been considered to estimate the denominator of the GCV function in connection with CG regularization. In the literature, the following techniques are suggested: the simple KA approach, i.e. $\operatorname{tr} A_k = k$; the *Monte-Carlo gcv* approach [4]; the approach proposed in [15], which deals with the dependance of the influence matrix on the noise by approximating the Jacobian with finite differences. A new approach, denoted as procedure P_1, has been introduced which also deals with the dependance of the influence matrix on the noise, but explicitly computes the Jacobian by means of recurrences. We have also considered a procedure P_2, obtained by simplifying the derivatives of the recursions in P_1, which actually coincides with the Monte-Carlo gcv approach. The procedure based on [15] has been here called procedure P_3.

The indicator $E(k)$ has been introduced to measure the effectiveness of the choice of $\boldsymbol{x}_k$ as a regularized solution. On the basis of the experiments carried

out on 1D and 2D test problems, we can draw some conclusions.

(1) Considering the results of the 2D experimentation (Table 4), we do not feel like suggesting the KA approach.
(2) The sensitivity of procedure P_3 to the choice of the parameter δ, its irregular behavior on the 1D problems (Table 3), its poorer performance with respect to procedures P_1 and P_2 in the 2D case (Tables 5, 6) and its larger computational cost do not suggest its use.
(3) Procedures P_1 and P_2 behave similarly and share approximately the same computational cost, but P_1 is theoretically more accurate. Considering the results of the 1D experimentation (Table 2), which are not contradicted by the results of the 2D experimentation (Table 5), we suggest the use of the new procedure P_1 together with few independent realizations of a vector $\boldsymbol{v}$ with distribution $\mathcal{N}(0, I)$.

References

[1] J.M. Bardsley, *Stopping Rules for a Nonnegatively Constrained Iterative Method for Ill-Posed Poisson Imaging Problems*, BIT Numerical Mathematics, 46 (2008) pp. 651–664.

[2] M. Bertero and P. Boccacci, *Image restoraton methods for the Large Binocular Telescope (LBT)*, *Astron. Astrophys. Suppl. Ser.*, 147 (2000) pp. 323–333.

[3] B. Eicke, A.K. Louis and R. Plato, *The instability of some gradient methods for ill-posed problems*, Numer. Math., 58 (1990) pp. 129–134.

[4] D.A. Girard, *A fast 'Monte-Carlo Cross-Validation' procedure for large least squares problems with noisy data*, Numer. Math., 56 (1989) pp. 1–23.

[5] G.H. Golub, M. Heath and G. Wahba, *Generalized cross-validation as a method for choosing a good ridge parameter*, Technometrics, 21 (1979) pp. 215-223.

[6] M. Hanke and P.C. Hansen, *Regularization methods for large-scale problems*, Surv. Math. Ind., 3 (1993) 253-315 .

[7] P.C. Hansen, *Rank-Deficient and Discrete Ill-Posed Problems*, SIAM Monographs on Mathematical Modeling and Computation 1998.

[8] P.C. Hansen, *Regularization tools*, Numer. Algo., 46 (2007) pp. 189–194, available at http://www2.imm.dtu.dk/∼pch/Regutools/RTv4manual.pdf

[9] E.J. Hoffman, P.D. Cutler, W.M. Digby and J.C. Mazziotta, *3-D phantom to simulate cerebral blood flow and metabolic images for PET*, IEEE Trans. Nucl. Sci., 37 (1990) pp. 616–620.

[10] M.F. Hutchinson, *A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines*, Communications in Statistics - Simulation and Computation, 19 (1990) pp. 433–450.

[11] K.P. Lee, J.C. Nagy and L. Perrone *Iterative methods or image restoration: a Matlab object oriented approach* (2002), available at http://www.mathcs.emory.edu/∼nagy/RestoreTools

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

[13] K.M. Perry and S.J. Reeves, *Generalized Cross-Validation as a Stopping Rule for the Richardson-Lucy Algorithm*, in The Restoration of HST Images and Spectra II. R.J. Hanisch and R.L. White, Eds., Space Telescope Science Institute, (1994) pp. 97-103.

[14] R. Plato, Optimal algorithms for linear ill-posed problems yield regularization methods, *Num. Funct. Anal. and Optimiz.*, 11 (1990) pp. 111–118.

[15] R.J. Santos and A.R. De Pierro, *The effect of the nonlinearity on GCV applied to Conjugate Gradients in computerized tomography*, Comput. Appl. Math., 25 (2006) pp. 111–128.

[16] C. Vogel, Computational Methods for Inverse Problems, SIAM Frontier in Applied Mathematics, Philadelphia, 2002.

[17] G. Wahba, Practical approximate solutions to linear operator equations when the data are noisy, SIAM J. Numer. Anal., 14 (1977) pp. 651–667.

[18] http://www.oersted.dtu.dk/ftp/jaj/31655/ct_programs/