A Multigrid method for image restoration

MARCO DONATELLI
(marco.donatelli@uninsubria.it)

Università dell'Insubria, Dip. CCFFMM - Sede di Como
Via Valleggio 11 - 22100 Como, Italy

Outline

- Boundary Conditions and structured linear systems
- Multigrid methods: MGM vs AMG
- Algebraic Multigrid (AMG) for Circulant matrices
- Reconstruction without regularization
- Two Tikhonov regularization techniques
- Numerical examples and comparisons
- Conclusion and future work

Boundary Conditions (BCs)

- The restored image $\mathbf{f}$ is obtained from the observed image $\mathbf{g}$ solving

$$A\mathbf{f} = \mathbf{g}$$

where A is a two level structured matrix, usually numerically banded at each level.

Boundary Conditions (BCs)

- The restored image $\mathbf{f}$ is obtained from the observed image $\mathbf{g}$ solving

$$A\mathbf{f} = \mathbf{g}$$

where A is a two level structured matrix, usually numerically banded at each level.

- Type of BCs:
  - Dirichlet $\Rightarrow A = T_{n,m}(z)$ (block Toeplitz - Toeplitz block),
  - Periodic $\Rightarrow A = C_{n,m}(z)$ (block Circulant - Circulant block),
  - Different BCs (Reflective, Anti-Reflective, $\ldots$) lead to matrices diagonalized by fast discrete transformations (cosine, sine, $\ldots$).

Toeplitz matrices

- Banded Toeplitz matrix:

$$T_n(z) = \begin{bmatrix} a_0 & a_1 & \cdots & a_b & & & \\ a_{-1} & \ddots & \ddots & & \ddots & & \\ \vdots & \ddots & \ddots & \ddots & & \ddots & \\ a_{-c} & & \ddots & \ddots & \ddots & & a_b \\ & \ddots & & \ddots & \ddots & \ddots & \vdots \\ & & \ddots & & \ddots & \ddots & a_1 \\ & & & a_{-c} & \cdots & a_{-1} & a_0 \end{bmatrix}_{n\times n}$$

Toeplitz matrices

- Banded **Toeplitz** matrix:

$$T_n(z) = \begin{bmatrix} a_0 & a_1 & \cdots & a_b & & & \\ a_{-1} & \ddots & \ddots & & \ddots & & \\ \vdots & \ddots & \ddots & \ddots & & \ddots & \\ a_{-c} & & \ddots & \ddots & \ddots & & a_b \\ & \ddots & & \ddots & \ddots & \ddots & \vdots \\ & & \ddots & & \ddots & \ddots & a_1 \\ & & & a_{-c} & \cdots & a_{-1} & a_0 \end{bmatrix}_{n \times n}$$

- Many properties of the matrix follow from properties of the associated generating function z. For instance:

$$z \geq 0 \quad \Longrightarrow \quad T_n(z) \text{ positive definite.}$$

Toeplitz matrices

- Banded Toeplitz matrix:

$$T_n(z) = \begin{bmatrix} a_0 & a_1 & \cdots & a_b & & & \\ a_{-1} & \ddots & \ddots & & \ddots & & \\ \vdots & \ddots & \ddots & \ddots & & \ddots & \\ a_{-c} & & \ddots & \ddots & \ddots & & a_b \\ & \ddots & & \ddots & \ddots & \ddots & \vdots \\ & & \ddots & & \ddots & \ddots & a_1 \\ & & & a_{-c} & \cdots & a_{-1} & a_0 \end{bmatrix}_{n \times n}$$

- Many properties of the matrix follow from properties of the associated generating function z. For instance:

$$z \geq 0 \quad \Longrightarrow \quad T_n(z) \text{ positive definite.}$$

- $T_{n,m}(z)$ is a two-level Toeplitz matrix where $z(x, y)$ is a bivariate function.

Circulant matrices

- Circulant matrix:

$$C_n(z) = \begin{bmatrix} a_0 & a_1 & \cdots & a_{n-1} \\ a_{n-1} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & a_1 \\ a_1 & \dots & a_{n-1} & a_0 \end{bmatrix}_{n \times n} .$$

Circulant matrices

- Circulant matrix:

$$C_n(z) = \begin{bmatrix} a_0 & a_1 & \cdots & a_{n-1} \\ a_{n-1} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & a_1 \\ a_1 & \dots & a_{n-1} & a_0 \end{bmatrix}_{n\times n}.$$

- Can be diagonalized by the Fast Fourier Transform F_n:

$$\mathcal{C}_n = \{F_n D_n F_n^H : \ D_n = \mathrm{diag}_{j=0,\dots,n-1}(\lambda_j)\}.$$

Circulant matrices

- Circulant matrix:

$$C_n(z) = \begin{bmatrix} a_0 & a_1 & \cdots & a_{n-1} \\ a_{n-1} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & a_1 \\ a_1 & \ldots & a_{n-1} & a_0 \end{bmatrix}_{n \times n}.$$

- Can be diagonalized by the Fast Fourier Transform F_n:

$$\mathcal{C}_n = \{F_n D_n F_n^H : \; D_n = \mathrm{diag}_{j=0,\ldots,n-1}(\lambda_j)\}.$$

- Let $C_n(z)$ be the Circulant matrix generated by $z(x)$ its eigenvalues are $\lambda_j = z(\frac{2\pi j}{n})$, $j = 0, \ldots, n-1$.

Multigrid: MGM vs AMG

- The Multigrid idea: to project recursively the problem in another of small dimension maintaining as much information as possible.

Multigrid: MGM vs AMG

- The Multigrid idea: to project recursively the problem in another of small dimension maintaining as much information as possible.
- In the Geometric Multigrid (MGM) the projector is fixed (the linear interpolator) and the smoother is chosen to obtain a smooth error.

Multigrid: MGM vs AMG

- The Multigrid idea: to project recursively the problem in another of small dimension maintaining as much information as possible.
- In the Geometric Multigrid (MGM) the projector is fixed (the linear interpolator) and the smoother is chosen to obtain a smooth error.
- In the Algebraic Multigrid (AMG) the smoother is fixed and the projector is chosen to project into the subspace where the smoother is ineffective.

Multigrid: MGM vs AMG

- The Multigrid idea: to project recursively the problem in another of small dimension maintaining as much information as possible.
- In the Geometric Multigrid (MGM) the projector is fixed (the linear interpolator) and the smoother is chosen to obtain a smooth error.
- In the Algebraic Multigrid (AMG) the smoother is fixed and the projector is chosen to project into the subspace where the smoother is ineffective.
- Many classical iterative methods (Richardson, Jacobi, Gauss-Seidel) have a similar spectral behavior. Therefore, for the AMG the choice of the projector is crucial.

Multigrid: MGM vs AMG

- The Multigrid idea: to project recursively the problem in another of small dimension maintaining as much information as possible.
- In the Geometric Multigrid (MGM) the projector is fixed (the linear interpolator) and the smoother is chosen to obtain a smooth error.
- In the Algebraic Multigrid (AMG) the smoother is fixed and the projector is chosen to project into the subspace where the smoother is ineffective.
- Many classical iterative methods (Richardson, Jacobi, Gauss-Seidel) have a similar spectral behavior. Therefore, for the AMG the choice of the projector is crucial.
- The proposed AMG is optimal (the computational cost for solving the linear system has the same order as the matrix-vector product) for generating function with zeros of finite order (Arico', Donatelli, Serra Capizzano, SIMAX to appear).

AMG for the Circulant algebra

- The smoother is the relaxed Richardson: $R_i = I_{n_i^2} - \omega_i C_{n_i,n_i}(z_i)$.

AMG for the Circulant algebra

- The smoother is the relaxed Richardson: $R_i = I_{n_i^2} - \omega_i C_{n_i,n_i}(z_i)$.
- The projector at level i is defined as $P_{i+1}^i = U_{i+1}^i \cdot C_{n_i,n_i}(p_i)$ where

$$U_{i+1}^i = K_{i+1}^i \otimes K_{i+1}^i, \qquad K_{i+1}^i = \begin{bmatrix} 1 & 0 & & & & & \\ & 0 & 1 & 0 & & & \\ & & & \ddots & \ddots & \ddots & \\ & & & & 0 & 1 & 0 \end{bmatrix}_{n_{i+1} \times n_i}$$

AMG for the Circulant algebra

- The smoother is the relaxed Richardson: $R_i = I_{n_i^2} - \omega_i C_{n_i,n_i}(z_i)$.
- The projector at level i is defined as $P_{i+1}^i = U_{i+1}^i \cdot C_{n_i,n_i}(p_i)$ where

$$U_{i+1}^i = K_{i+1}^i \otimes K_{i+1}^i, \qquad K_{i+1}^i = \begin{bmatrix} 1 & 0 & & & & & \\ & 0 & 1 & 0 & & & \\ & & & \ddots & \ddots & \ddots & \\ & & & & 0 & 1 & 0 \end{bmatrix}_{n_{i+1}\times n_i}$$

and let $w_0 = (x_0, y_0)$ be the unique zeros of z_i of order $2q$, p_i is such that:

$$\begin{array}{lll} (1) & \limsup_{w\to w_0} \left|\dfrac{p_i(\hat{w})}{z_i(w)}\right| < +\infty, & \hat{w} \in M(w), \\ (2) & 0 < \displaystyle\sum_{\hat{w}\in M(w)\cup\{w\}} p_i^2(\hat{w}), & i = 0, \ldots, m-1, \end{array}$$

where $M(w) = \{(\pi + x,\, y),\ (x,\, \pi + y),\ (\pi + x,\, \pi + y)\}$.

Computational issue

- For each level $i = 0, \ldots, m$:

$$U_{i+1}^{i} \Longrightarrow A_i \in \text{Circulant}$$

$p_i \Longrightarrow$ projects in the subspace where the smoother is ineffective

Computational issue

- For each level $i = 0, \dots, m$:

$$U^i_{i+1} \Longrightarrow A_i \in \text{Circulant}$$

$p_i \Longrightarrow$ projects in the subspace where the smoother is ineffective

- Using a Galerkin strategy ($A_{i+1} = P^i_{i+1} A_i (P^i_{i+1})^T$) the matrices A_i and P^i_{i+1} are computed in a Setup phase before applying the AMG with a logarithmic computational cost in the dimension.

Computational issue

- For each level $i = 0, \ldots, m$:

$$U_{i+1}^{i} \Longrightarrow A_i \in \text{Circulant}$$

$p_i \Longrightarrow$ projects in the subspace where the smoother is ineffective

- Using a Galerkin strategy ($A_{i+1} = P_{i+1}^{i} A_i (P_{i+1}^{i})^T$) the matrices A_i and P_{i+1}^{i} are computed in a Setup phase before applying the AMG with a logarithmic computational cost in the dimension.

- At each recursion level the A_i bandwidth is halved like the dimension and tends to the double of the P_{i+1}^{i} bandwidth (which is small: it is a function of the ill-conditioning of A_0).

Computational issue

- For each level $i = 0, \ldots, m$:

$$U_{i+1}^{i} \Longrightarrow A_i \in \text{Circulant}$$

$p_i \Longrightarrow$ projects in the subspace where the smoother is ineffective

- Using a Galerkin strategy ($A_{i+1} = P_{i+1}^{i} A_i (P_{i+1}^{i})^T$) the matrices A_i and P_{i+1}^{i} are computed in a Setup phase before applying the AMG with a logarithmic computational cost in the dimension.

- At each recursion level the A_i bandwidth is halved like the dimension and tends to the double of the P_{i+1}^{i} bandwidth (which is small: it is a function of the ill-conditioning of A_0).

- Let $\mathcal{AMG}$ be the iteration matrix, we have proved that

$$\rho(\mathcal{AMG}) \leq c < 1$$

where c is independent from $n \Longrightarrow$ increasing n the number of iterations tends to a constant value.

PSF and noise

- We add 2% of noise to the blurred image $\hat{\mathbf{g}}$ solving

$$C_{n,n}(z)\mathbf{f} = \mathbf{g},$$

where $\mathbf{g} = \hat{\mathbf{g}} + \mathbf{n}$, $\mathbf{n} = 2\frac{\|\hat{\mathbf{g}}\|_2}{\|\mathbf{a}\|_2}\mathbf{a}$ and $\mathbf{a} = \mathrm{rand}(n^2, 1)$.

PSF and noise

- We add 2% of noise to the blurred image $\hat{\mathbf{g}}$ solving

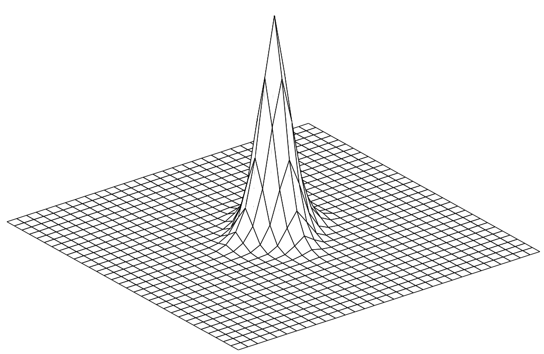
$$C_{n,n}(z)\mathbf{f} = \mathbf{g},$$

  where $\mathbf{g} = \hat{\mathbf{g}} + \mathbf{n}$, $\mathbf{n} = 2\frac{\|\hat{\mathbf{g}}\|_2}{\|\mathbf{a}\|_2}\mathbf{a}$ and $\mathbf{a} = \mathrm{rand}(n^2, 1)$.

- The PSF is generated by $z(x, y) \geq 0$ and is close to zero in a neighborhood of (π, π):

$$z(x, y) = \frac{1}{c}F(x, y)\psi(x, y),$$

  where $F(x, y) = (2 + \cos(x) + \cos(y))^3$ is the "kernel" that vanishes in (π, π) with order 6, $\psi(x, y) > 0$ with nonnegative Fourier coefficients and c is a normalization constant.

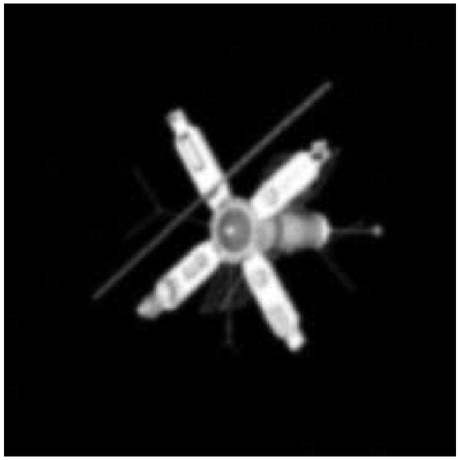


Reconstruction without regularization

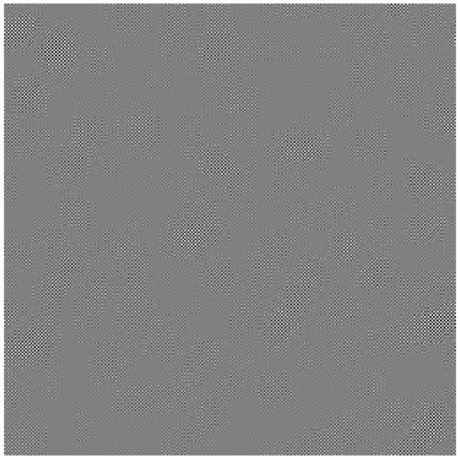
- Pre-smoother: two iterations of relaxed Richardson with $\omega = \max(z(x,y))^{-1}$.
  Post-smoother: two iterations of Conjugate Gradient without preconditioning.

Reconstruction without regularization

- Pre-smoother: two iterations of relaxed Richardson with $\omega = \max(z(x,y))^{-1}$.
  Post-smoother: two iterations of Conjugate Gradient without preconditioning.
- Our AMG does not have any regularization property.



Blurred image with 2% of noise.



Restored image after 85 iterations.

#(Iter.)	$\|\text{error}\|_2$
1	1.876715E+06
15	2.691198E+06
30	2.702400E+06
45	2.702924E+06
60	2.702964E+06
85	2.702967E+06

Relative error in $\|\cdot\|_2$.

- It approximates the solution in the whole frequency space and not only in a low frequency subspace: it is disturbed by noise at each iteration.

Two regularization techniques

- Tikhonov regularization:

$$\min_{\mathbf{f}\in\mathbb{R}^{n^2}} \left\{ \|C_{n,n}(z)\mathbf{f} - \mathbf{g}\|_2^2 + \mu \, \|\mathbf{f}\|_2^2 \right\},$$

which leads to

$$C_{n,n}(z^2+\mu)\mathbf{f} = C_{n,n}(z)\mathbf{g},$$

but it doubles the condition number.

Two regularization techniques

- Tikhonov regularization:

$$\min_{\mathbf{f}\in\mathbb{R}^{n^2}} \left\{ \|C_{n,n}(z)\mathbf{f} - \mathbf{g}\|_2^2 + \mu \|\mathbf{f}\|_2^2 \right\},$$

  which leads to

$$C_{n,n}(z^2+\mu)\mathbf{f} = C_{n,n}(z)\mathbf{g},$$

  but it doubles the condition number.

- Riley regularization:

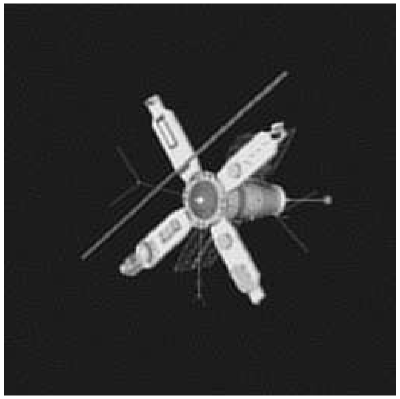
$$C_{n,n}(z+\theta)\mathbf{f} = \mathbf{g},$$

  which is equivalent to

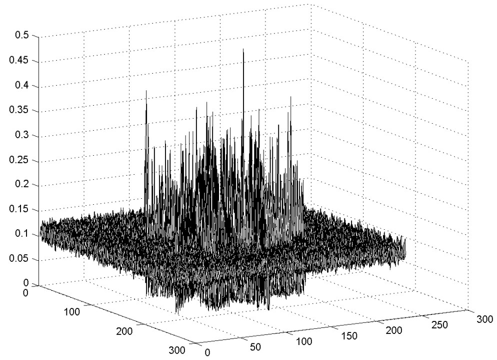
$$\min_{\mathbf{f}\in\mathbb{R}^{n^2}} \left\{ \left\| C_{n,n}(z)^{\frac{1}{2}}\mathbf{f} - C_{n,n}(z)^{-\frac{1}{2}}\mathbf{g} \right\|_2^2 + \theta \|\mathbf{f}\|_2^2 \right\}.$$

Numerical experiments

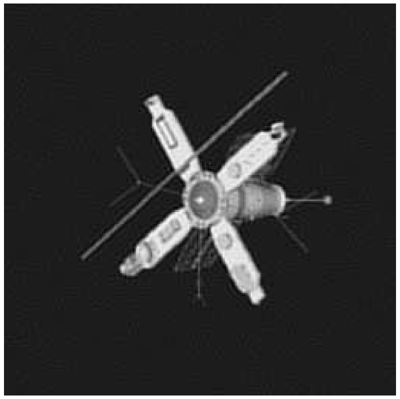
- Restored image μ_{opt}. **Tikhonov** Error.



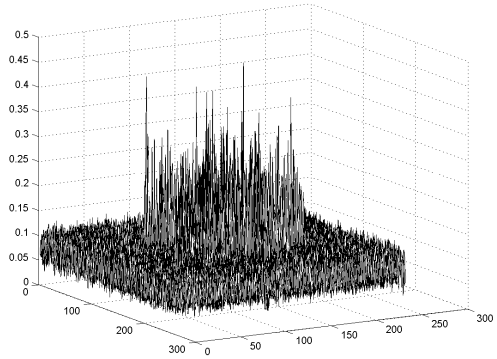
$\|\text{error}\|_2 = 0.1578$



- Restored image θ_{opt}. **Riley** Error.



$\|\text{error}\|_2 = 0.1948$



AMG and MGM: comparison

- R. Chan, T.Chan et al. proposed a MGM for image restoration using Riley's regularization and Preconditioned Conjugated Gradient as smoother.

AMG and MGM: comparison

- R. Chan, T.Chan et al. proposed a MGM for image restoration using Riley's regularization and Preconditioned Conjugated Gradient as smoother.
- Also with more simple smoothers our AMG is optimal:

  decreasing θ the number of iterations tends to a constant value without exploding.

AMG and MGM: comparison

- R. Chan, T.Chan et al. proposed a MGM for image restoration using Riley's regularization and Preconditioned Conjugated Gradient as smoother.
- Also with more simple smoothers our AMG is optimal:

  decreasing θ the number of iterations tends to a constant value without exploding.

- Number of iterations for AMG and MGM using the Riley's strategy with Richardson and Conjugate Gradient as smoothers:

θ	10^{-2}	10^{-3}	10^{-4}	10^{-5}
$p(x,y) = (2 - 2\cos(x))^2(2 - 2\cos(y))^2$ our projector	15	49	73	73
$p(x,y) = (2 + 2\cos(x))(2 + 2\cos(y))$ linear interpolator	15	45	286	> 1000

$\Longrightarrow$ R. Chan, T.Chan et al.
T. Huckle et al.

Conclusion

- Our AMG does not have any regularization property.
- We must use the Tikhonov regularization technique.
- Decreasing the regularization parameter, our AMG requires the same number of iterations.

Future work

- To test our AMG with a PSF generated by a Gaussian function.
- To extend the work to other Boundary Conditions (Reflective, Anti-Reflective).
- Will it be possible to define a regularization property or a regularization parameter estimation inside the AMG?