ADMM Penalty Parameter Selection with Krylov Subspace Recycling Technique for Sparse Coding

ICIP 2017 2017-10-20 Beijing

Youzuo Lin, Brendt Wohlberg, and Velimir V. Vesselinov

Los Alamos National Laboratory
Los Alamos, NM 87545
USA

ADMM

- The general form for an ADMM problem is

$$\arg\min_{\mathbf{x},\mathbf{y}} f(\mathbf{x}) + g(\mathbf{y}) \;\; \text{such that} \;\; A\mathbf{x} + B\mathbf{y} = \mathbf{c}$$

- ADMM iterations:

$$\mathbf{x}^{(k+1)} = \arg\min_{\mathbf{x}} f(\mathbf{x}) + \frac{\rho}{2}\left\|A\mathbf{x} + B\mathbf{y}^{(k)} - \mathbf{c} + \mathbf{u}^{(k)}\right\|_2^2$$
$$\mathbf{y}^{(k+1)} = \arg\min_{\mathbf{y}} g(\mathbf{y}) + \frac{\rho}{2}\left\|A\mathbf{x}^{(k+1)} + B\mathbf{y} - \mathbf{c} + \mathbf{u}^{(k)}\right\|_2^2$$
$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + A\mathbf{x}^{(k+1)} + B\mathbf{y}^{(k+1)} - \mathbf{c}$$

- Convergence is guaranteed under relatively mild conditions.
- In practice ADMM works well for a wide range of problems.

ADMM Penalty Parameter I

- **But** the convergence rate of the algorithm depends strongly on the penalty parameter ρ.
- A variety of penalty parameter selection methods have been proposed
  - Theoretically optimal parameters derived for a restricted class of problems, e.g. Raghunathan et al. (2014), Ghadimi et al. (2015)
  - Heuristic methods that do not provide good performance in all contexts, e.g. He et al. (2000), Wohlberg (2017), Xu et al. (2017)
  - Theory based methods that are broadly applicable but complex or expensive to implement, e.g. Nishihara et al. (2015)

ADMM Penalty Parameter II

- Proposed approach:
  - Applicable to problems in which the main linear system is solved via iterative methods
  - The selection principle itself is simple: solve the linear system for a number of different ρ values, selecting the value that delivers the smallest value of functional to be minimized
  - The additional solutions of the linear system can be computed at very small additional cost by exploiting Krylov subspace methods

Imaging Inverse Problems

- We are interested in addressing problems of the form

$$\arg\min_{\mathbf{x}} (1/2) \|F\mathbf{x} - \mathbf{s}\|_2^2 + R(\mathbf{x})$$

  where F is the forward operator and R is the regularization term.
- Such problems can be solved within the ADMM framework via *variable splitting*

$$\arg\min_{\mathbf{x},\mathbf{y}} (1/2) \|F\mathbf{x} - \mathbf{s}\|_2^2 + R(\mathbf{y}) \quad \text{s.t.} \quad \mathbf{x} = \mathbf{y}$$

ADMM Solution

- The ADMM iterations for this problem are:

$$\mathbf{x}^{(k+1)} = \arg\min_{\mathbf{x}} \, (1/2) \left\| F\mathbf{x} - \mathbf{s} \right\|_2^2 + \frac{\rho}{2} \left\| \mathbf{x} - \mathbf{y}^{(k)} + \mathbf{u}^{(k)} \right\|_2^2 \quad (1)$$

$$\mathbf{y}^{(k+1)} = \arg\min_{\mathbf{y}} R(\mathbf{y}) + \frac{\rho}{2} \left\| \mathbf{x}^{(k+1)} - \mathbf{y} + \mathbf{u}^{(k)} \right\|_2^2 \quad (2)$$

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \mathbf{x}^{(k+1)} - \mathbf{y}^{(k+1)} \quad (3)$$

- One of the advantages of ADMM is the decoupling of the data fidelity and regularization terms:
  - The solution to (2) depends on the form of $R(\cdot)$.
  - Solving (1) requires solving the linear system

$$(F^T F + \rho I)\mathbf{x} = F^T \mathbf{s} + \rho(\mathbf{y}^{(k)} - \mathbf{u}^{(k)}) \quad \Rightarrow$$

Solving the Linear System

- The major computational cost of an ADMM algorithm is often in solving this linear system (repeated here)

$$(F^T F + \rho I)\mathbf{x} = F^T \mathbf{s} + \rho(\mathbf{y}^{(k)} - \mathbf{u}^{(k)})$$

- When F is an explicit matrix, an LU or Cholesky pre-factorization of $F^T F + \rho I$ can be used for an efficient solution via direct methods.
- For many inverse problems (e.g. tomography), F is represented as a transform operator: we need to use iterative methods (e.g. CG, LSQR) to solve this linear system.
- The LSQR algorithm has some very useful properties for this problem.

LSQR I

- LSQR is an iterative linear solver with good performance on large-scale ill-posed problems.
- It belongs to the family for *Krylov subspace* techniques.
- For least squares problem with $A \in \mathbb{R}^{N \times N}$

$$\arg\min_{\mathbf{x}} (1/2) \|A\mathbf{x} - \mathbf{b}\|_2^2$$

the order-n Krylov subspace is

$$\mathcal{K}_n(A, \mathbf{b}) = \mathrm{span}\{\mathbf{b}, A\mathbf{b}, A^{(2)}\mathbf{b}, \ldots, A^{(n-1)}\mathbf{b}\}$$

- If $A \in \mathbb{R}^{N \times M}$, the Krylov subspace is generated for the normal equations $A^T A\mathbf{x} = A^T\mathbf{b}$

$$\mathcal{K}_n(A^T A, A^T\mathbf{b}) = \mathrm{span}\{A^T\mathbf{b}, A^T A\mathbf{b}, (A^T A)^{(2)}\mathbf{b}, \ldots, (A^T A)^{(n-1)}\mathbf{b}\}$$

LSQR II

- LSQR is based on the Golub-Kahan-Lanczos (GKL) bidiagonalization technique: a computationally efficient iterative algorithm for constructing an orthogonal basis for the Krylov subspace.
- The major computational cost is in the computation of a pair of orthogonal bases $U^{(k+1)}$ and $V^{(k)}$.
- Given the bases, solving the problem via the bidiagonal decomposition is very cheap

$$(U^{(k+1)})^T A V^{(k)} = \begin{bmatrix} \alpha^{(1)} & & & \\ \beta^{(2)} & \alpha^{(2)} & & \\ & \beta^{(3)} & \ddots & \\ & & \ddots & \alpha^{(k)} \\ & & & \beta^{(k+1)} \end{bmatrix}$$

Linear Problem Transformation

- Returning to the ADMM subproblem, we need to solve

$$\mathbf{x}^{(k+1)} = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2} \|F\mathbf{x} - \mathbf{s}\|_2^2 + \frac{\rho}{2} \left\| \mathbf{x} - (\mathbf{y}^{(k)} - \mathbf{u}^{(k)}) \right\|_2^2 \right\}$$

- Need to transform into a least squares problem so that we can apply LSQR
- Standard form transformation

$$\tilde{\mathbf{x}} = \mathbf{x} - (\mathbf{y}^{(k)} - \mathbf{u}^{(k)})$$
$$\tilde{\mathbf{s}} = \mathbf{s} - F(\mathbf{y}^{(k)} - \mathbf{u}^{(k)})$$

gives

$$\tilde{\mathbf{x}}^{(k+1)} = \arg\min_{\tilde{\mathbf{x}}} \left\{ \frac{1}{2} \|F\tilde{\mathbf{x}} - \tilde{\mathbf{s}}\|_2^2 + \frac{\rho}{2} \|\tilde{\mathbf{x}}\|_2^2 \right\}$$

Equivalent Least Squares Form

Using

$$\|\mathbf{x}\|_2^2 + \|\mathbf{y}\|_2^2 = \left\| \begin{pmatrix} \mathbf{x}^T & \mathbf{y}^T \end{pmatrix}^T \right\|_2^2$$

the standard form

$$\tilde{\mathbf{x}}^{(k+1)} = \arg\min_{\tilde{\mathbf{x}}} \left\{ \frac{1}{2} \|F\tilde{\mathbf{x}} - \tilde{\mathbf{s}}\|_2^2 + \frac{\rho}{2} \|\tilde{\mathbf{x}}\|_2^2 \right\}$$

can be written in the equivalent least squares form

$$\tilde{\mathbf{x}}^{(k+1)} = \arg\min_{\tilde{\mathbf{x}}} \left\| \begin{bmatrix} F \\ \sqrt{\rho} I \end{bmatrix} \tilde{\mathbf{x}} - \begin{bmatrix} \tilde{\mathbf{s}} \\ 0 \end{bmatrix} \right\|_2^2$$

Normal equations are

$$(F^T F + \rho I)\tilde{\mathbf{x}} = F^T \tilde{\mathbf{s}}$$

Krylov Subspace Invariance I

- Krylov subspace at the n^{th} step of the GKL method

$$\begin{aligned}
&\mathcal{K}_n\{F^T F + \rho I, F^T \tilde{\mathbf{s}}\} \\
=\ & \text{span}\{F^T \tilde{\mathbf{s}}, (F^T F + \rho I)\, F^T \tilde{\mathbf{s}}, (F^T F + \rho I)^2\, F^T \tilde{\mathbf{s}}, \ldots\} \\
=\ & \text{span}\{F^T \tilde{\mathbf{s}}, F^T F F^T \tilde{\mathbf{s}} + \rho F^T \tilde{\mathbf{s}}, (F^T F)^2 F^T \tilde{\mathbf{s}} + 2\rho F^T F F^T \tilde{\mathbf{s}} \\
&\qquad + \rho^2 F^T \tilde{\mathbf{s}}, \ldots\} \\
=\ & \text{span}\{F^T \tilde{\mathbf{s}}, F^T F F^T \tilde{\mathbf{s}}, (F^T F)^2 F^T \tilde{\mathbf{s}}, \ldots\} = \mathcal{K}_n\{F^T F, F^T \tilde{\mathbf{s}}\}
\end{aligned}$$

- The Krylov subspaces for

$$(F^T F + \rho I)\tilde{\mathbf{x}} = F^T \tilde{\mathbf{s}}$$

and

$$F^T F \tilde{\mathbf{x}} = F^T \tilde{\mathbf{s}}$$

are the same.

Krylov Subspace Invariance II

- It can also be shown that the bases generated by the GKL method are the same for problems

$$F^T F\tilde{\mathbf{x}} = F^T\tilde{\mathbf{s}} \tag{4}$$

and

$$(F^T F + \rho I)\tilde{\mathbf{x}} = F^T\tilde{\mathbf{s}} \tag{5}$$

- *Subspace Recycling*: We can compute the Krylov subspace for problem (4) and then use it to cheaply compute the solution to problem (5) with multiple ρ values

Robust ADMM Penalty Parameter Selection

- Generate N logarithmically spaced ADMM penalty parameters between 10^a and 10^b: $\rho_1, \ldots, \rho_N$
- Compute the GKL bases for $F^T F \tilde{\mathbf{x}} = F^T \tilde{\mathbf{s}}$ (seed system)
- Use the GKL bases to efficiently solve $(F^T F + \rho_n I)\tilde{\mathbf{x}} = F^T \tilde{\mathbf{s}}$ $\forall n \in \{1, 2, \ldots, N\}$ (non-seed systems)
- Select the ρ_n value that minimizes the problem functional

$$\rho_{\text{optimal}} = \underset{\rho \in \{\rho_1, \rho_2, \ldots, \rho_N\}}{\arg\min} \; (1/2) \left\| F\mathbf{x}(\rho) - \mathbf{s} \right\|_2^2 + R(\mathbf{x}(\rho))$$

- Use the corresponding $\mathbf{x}(\rho_{\text{optimal}})$ value as the solution of the ADMM $\mathbf{x}$ step, and proceed to compute the ADMM $\mathbf{y}$ and $\mathbf{u}$ steps

Test Problem: Sparse Coding

- Consider sparse coding via Basis Pursuit DeNoising (BPDN)

$$\arg\min_{\mathbf{x}} \{(1/2)\|D\mathbf{x} - \mathbf{s}\|_2 + \lambda\|\mathbf{x}\|_1\}$$

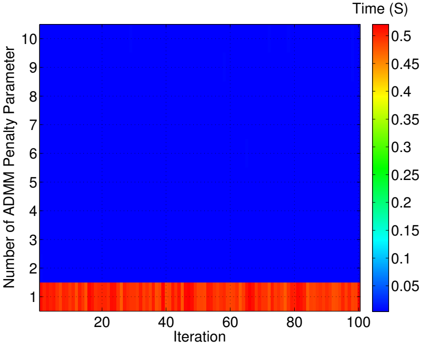
  with dictionary D.

- This problem fits within our general framework with

$$F = D \qquad\qquad R(\mathbf{x}) = \lambda\|\mathbf{x}\|_1$$

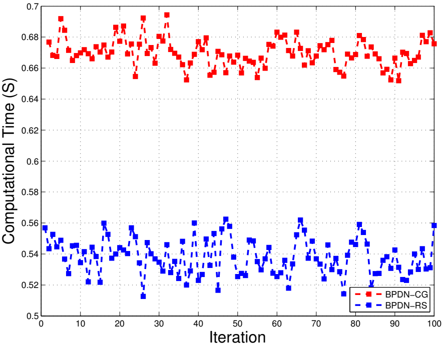
- Address a Gaussian white noise image denoising problem.
- Use the $2\times$ overcomplete Discrete Cosine Transform (DCT) basis as the dictionary.

Linear Solver Efficiency I



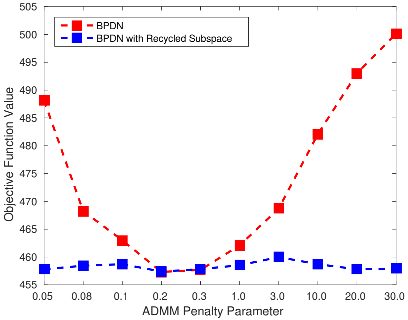
- Computation time cost in solving the inverse step using $N = 10$ penalty parameters for our method.
- Solving each non-seed system is about an order of magnitude faster than solving the seed system.

Linear Solver Efficiency II



- The overall computation time at each iteration using BPDN with CG solver (in red) and BPDN with our subspace recycling technique (in blue).
- Our method is consistently more efficient than BPDN-CG method.

Parameter Selection Efficacy



- Solve multiple times with different initial penalty parameters $\rho_{\text{init}} \in \{0.05, 0.08, 0.1, 0.2, 0.3, 1.0, 3.0, 10.0, 20.0 \text{ and } 30.0\}$
- Our method yields objective function value consistently close to the optimal value regardless of ρ_{init}

Conclusions and Future Work

- We have developed a computationally efficient ADMM penalty parameter selection technique using Krylov subspace recycling.
- Initial experiments in using a sparse coding problem for image denoising indicate that it is very effective in selecting close to optimal penalty parameters.
- Future work:
  - Test the method on a wider range of image reconstructions problems.
  - Compare with alternative parameter selection techniques.