Lecture 14: ICD, Preconditioning, and Conjugate Gradient

Chapter 5.4–5.5

Charles A. Bouman

Purdue University

Friday, September 25, 2026

Outline

1. **Iterative coordinate descent**
   Each pixel is updated to its exact minimizer.
2. **Making ICD practical**
   The error is stored as a state variable, and the LSI case gives a simple update.
3. **Why ICD behaves differently**
   ICD is compared with gradient descent as a function of spatial frequency.
4. **Conjugate gradient (supplementary)**
   A method whose speed does not depend on the condition number.

Reading: Chapter 5.4–5.5, *Foundations of Computational Imaging*.

Today's Big Ideas

- ICD updates one pixel to its exact minimizer; no step size.
- Storing the error $e = y - Ax$ makes each update local and cheap.
- With no blur, the update averages the measurement and neighbors.
- ICD converges fast at high frequencies: edges come first.

Chapter 5.5, conjugate gradient, follows as supplementary material.

Coordinate Descent

Coordinate descent is a sequential greedy update of pixels.

Advantages:

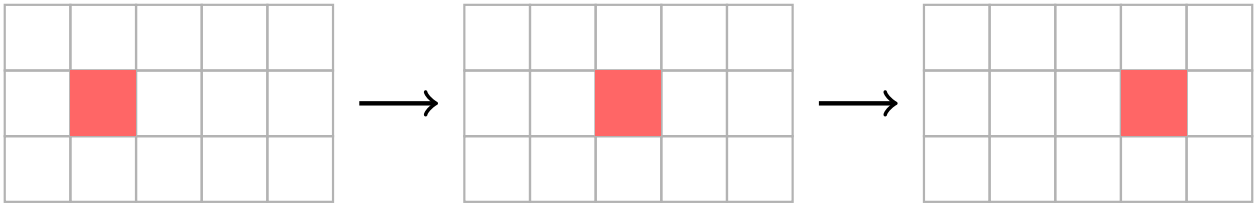
- Does not require the selection of a step size.
- Faster convergence at high frequencies.

Disadvantages:

- More difficult to parallelize.

Iterative coordinate descent: one pixel at a time

Iterative coordinate descent (ICD) updates each pixel in sequence while the remaining pixels are held fixed. Each update solves a one-variable minimization exactly.



update x_s then the next and the next

The ICD update is most easily written in the assignment notation of a programming language,

$$x_s \leftarrow \underset{x_s \in \mathbb{R}}{\arg\min}\, f(x) \ ,$$

where the pixels x_r for $r \neq s$ are unchanged. One full iteration updates each pixel exactly once.

The ICD step α

The result of an ICD update of the pixel s can be compactly expressed as $x + \alpha\varepsilon_s$, where ε_s is a vector that is one for element s and zero otherwise. Using this notation,

$$x_s \leftarrow x_s + \arg\min_{\alpha \in \mathbb{R}} f(x + \alpha\varepsilon_s) \ .$$

The ICD update can then be computed by solving the equation

$$0 \ = \ \frac{\partial f(x + \alpha\varepsilon_s)}{\partial \alpha} \ = \ \left[\nabla f(x + \alpha\varepsilon_s)\right]^t \varepsilon_s \ ,$$

which results in

$$\alpha = \frac{(y - Ax)^t A_{*,s} - \sigma^2 x^t B_{*,s}}{||A_{*,s}||^2 + \sigma^2 B_{s,s}} \ ,$$

where $A_{*,s}$ and $B_{*,s}$ denote the s^{th} columns of A and B, respectively.

The minimization along the coordinate direction is **exact**, so there is no step-size parameter and no approximation.

The ICD update equation

Writing the prior term out with the GMRF prediction filter gives the ICD update equation

$$x_s \leftarrow x_s + \frac{(y - Ax)^t A_{*,s} - \lambda \left(x_s - \sum_{r \in \partial s} g_{s-r} x_r \right)}{||A_{*,s}||^2 + \lambda} \ ,$$

where $\lambda = \frac{\sigma^2}{\sigma_x^2}$ is the noise-to-signal ratio of Lecture 13.

- The first numerator term accounts for the forward model error.
- The second term accounts for the non-causal prediction error.

The computational problem

Direct implementation requires evaluation of the term $y - Ax$ with each pixel update. Typically, this requires too much computation.

Fast ICD: keep the error as a state variable

Fortunately, there is a simple method for speeding the computation by keeping a state variable which stores the error term $e = y - Ax$.

ICD Algorithm:

Initialize $e \leftarrow y - Ax$

For K iterations $\{$

  For each pixel $s \in S$ $\{$

$$
\begin{aligned}
v &\leftarrow x_s \\
x_s &\leftarrow x_s + \frac{e^t A_{*,s} - \lambda \left(x_s - \sum_{r \in \partial s} g_{s-r} x_r \right)}{||A_{*,s}||^2 + \lambda} \\
e &\leftarrow e - A_{*,s}(x_s - v)
\end{aligned}
$$

  $\}$

$\}$

The error is updated incrementally, using only the column $A_{*,s}$ of the pixel that changed.

The LSI special case

As with gradient descent, when A represents an LSI filter a_s, then the update has the form

$$x_s \leftarrow x_s + \frac{a_{-s} * (y_s - a_s * x_s) - \lambda \left(x_s - \sum_{r \in \partial s} g_{s-r} x_r \right)}{\sum_s a_s^2 + \lambda} \ .$$

- The denominator $\sum_s a_s^2 + \lambda$ is a constant, computed once.

No blur: the update is a weighted average

When the point-spread function is $a_s = \delta_s$, there is no blurring, and the update has a simpler form:

$$\boxed{x_s \leftarrow \frac{y_s + \lambda \sum_{r \in \partial s} g_{s-r} x_r}{1 + \lambda}}.$$

The update is a **weighted average** of the measurement y_s and the noncausal prediction from the neighbors. The weight on the measurement is 1 and the weight on the prediction is λ.

measurement y_s ⟵ x_s ⟶ neighbors

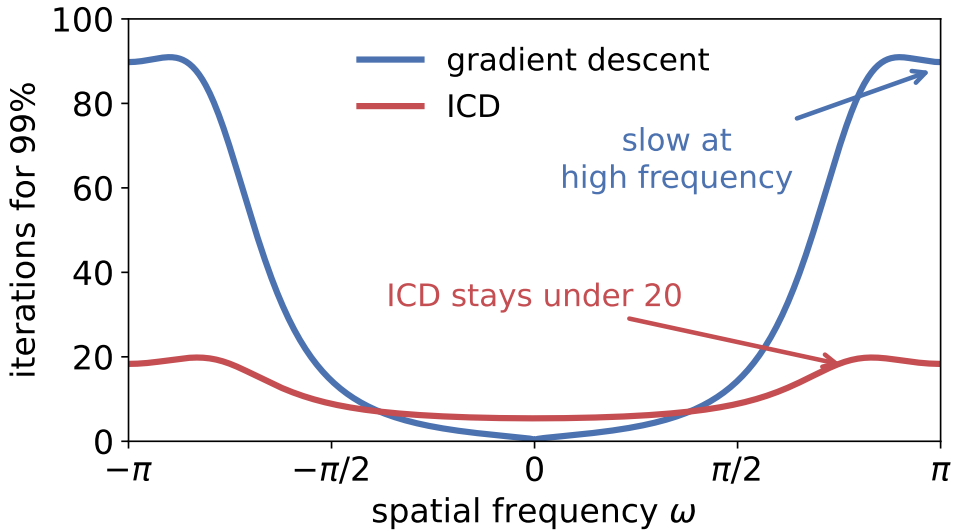
clean data — noisy data

$\lambda \to 0$ — $\lambda \to \infty$

- Weighting set by the signal-to-noise ratio: $\lambda = \sigma^2 / \sigma_x^2$.

Convergence of ICD

- ICD convergence is analyzed like gradient descent.
- Refer to the book for details.
- Bottom line:
  - ICD tends to converge faster at high spatial frequencies
  - This is important because high frequencies require more computation.



Example 5.2: ICD and gradient descent on the same 1-D deconvolution problem.

Other Commonly used optimization methods

The MAP estimate is the minimizer of a quadratic cost function:

$$\hat{x}_{MAP} = \arg\min_{x \in \mathbb{R}^N} f(x) \ , \qquad f(x) = \frac{1}{2\sigma^2}||y - Ax||^2 + \frac{1}{2}x^t Bx \ .$$

- Four commonly used methods:
  - **Gradient descent**
  - **Line search**
  - **Preconditioning**
  - **Conjugate gradient**

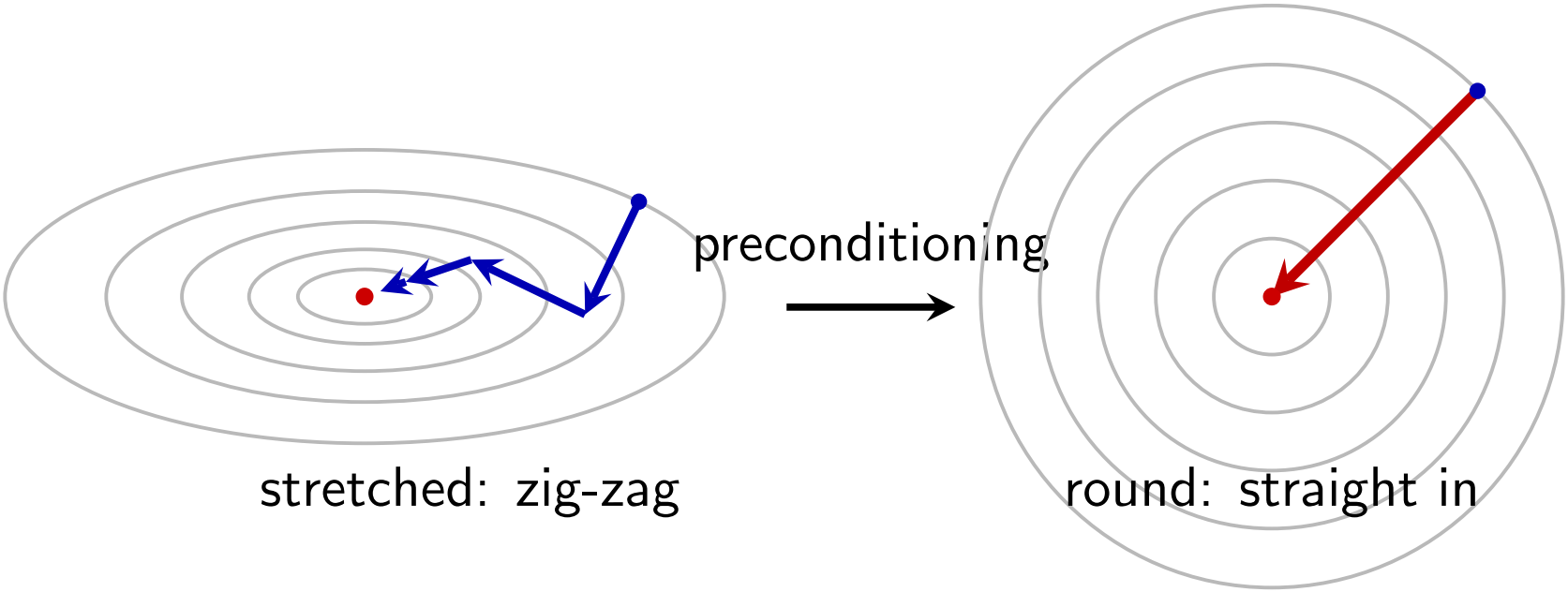
These can be mixed and matched depending on the application.

Preconditioned gradient descent

Preconditioning multiplies the gradient by $M^{-1} \approx H^{-1}$:

$$x^{(k+1)} = x^{(k)} - \alpha\, M^{-1} \nabla f(x^{(k)}) \ .$$

It rounds the contours, so descent heads almost straight to the minimum.



M must be positive definite

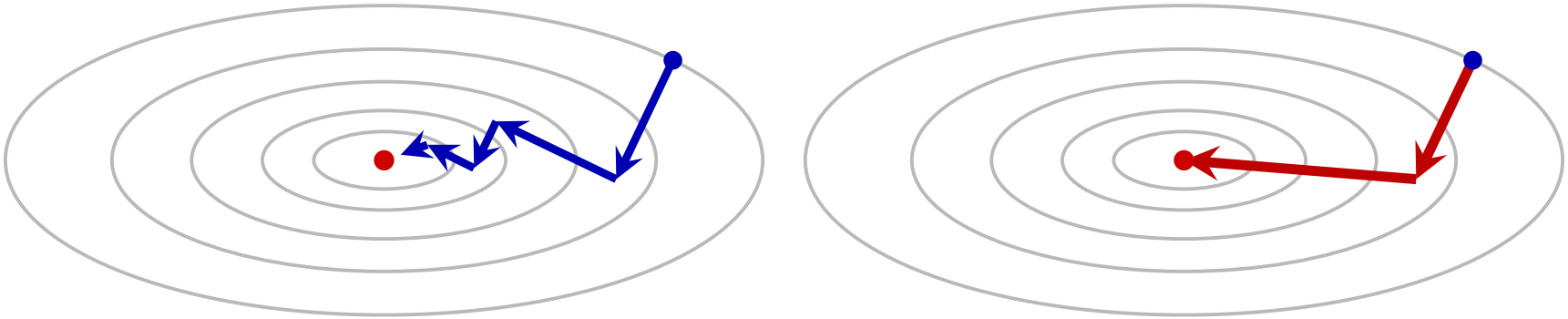
Writing $g = \nabla f(x^{(k)})$, the preconditioned step is $x^{(k+1)} = x^{(k)} - \alpha M^{-1} g$, so a first-order expansion in α gives

$$f(x^{(k+1)}) \approx f(x^{(k)}) - \alpha\, g^t M^{-1} g \ .$$

- So for the cost to fall for small $\alpha > 0$, M must be **positive definite**.
- However, making $M \approx H$ speeds convergence.
- To ensure this, M is often diagonal with strictly positive entries.

Conjugate gradient

Two directions are **conjugate** when $u^t H v = 0$. Conjugate gradient makes each search direction conjugate to the previous ones, which avoids the zig-zag.



gradient descent

conjugate gradient

Conjugate gradient reaches the minimum in two steps.
Gradient descent still zig-zags.

The conjugate gradient iteration

Initialization. Set $h^{(0)} = d^{(0)} = -\nabla f(x^{(0)})$, where $x^{(0)}$ is the initial image and $\nabla f(x) = -\frac{1}{\sigma^2} A^t(y - Ax) + Bx$.

Iterate until convergence:

$$
\begin{aligned}
d^{(k)} &= -\nabla f(x^{(k)}) \ , \\
\gamma_k &= \frac{(d^{(k)} - d^{(k-1)})^t d^{(k)}}{(d^{(k-1)})^t d^{(k-1)}} \ , \\
h^{(k)} &= d^{(k)} + \gamma_k h^{(k-1)} \ , \\
\alpha^{(k)} &= \underset{\alpha \geq 0}{\arg\min}\, f(x^{(k)} + \alpha h^{(k)}) \ , \\
x^{(k+1)} &= x^{(k)} + \alpha^{(k)} h^{(k)} \ .
\end{aligned}
$$

There is no step-size parameter, and for a quadratic cost the method reaches the exact minimum in at most N steps.

Chapter synthesis: choosing an optimizer

Method	Strengths	Weaknesses
Gradient descent	The algorithm is simple. Each update is one gradient evaluation.	The step size must be chosen: too large is unstable, too small is slow. The slowest mode decays as $(1 - 1/\kappa(H))^k$, so high frequencies are slow.
GD with line search	There is no step-size parameter, and the cost decreases monotonically, so convergence is stable.	The line search can dominate the computation, and it is still a gradient method limited by $\kappa(H)$.
ICD	There is no step-size parameter. Each pixel is updated exactly. Convergence is fast at high spatial frequencies, where the edges are.	Convergence is slow at low spatial frequencies. Updates are sequential, so ICD is harder to parallelize.
Precondi-tioned GD	Multiplying the gradient by $M^{-1} \approx H^{-1}$ rounds the contours, so convergence depends much less on $\kappa(H)$.	A step size is still needed. M must be positive definite, chosen, and cheap to invert.
Conjugate gradient	Each direction is conjugate to the previous ones, which mitigates the condition-number limit. No step size, and exact in at most N steps.	Running N iterations is often impractical.

Summary

- ICD updates one pixel to its exact minimizer; no step size.
- The error $e = y - Ax$ is kept as a state variable, so each update is local.
- With no blur, the update is a weighted average of the measurement and the neighbors, $x_s \leftarrow \frac{y_s + \lambda \sum_{r \in \partial s} g_{s-r} x_r}{1+\lambda}$.
- ICD: under 20 iterations at high frequencies; gradient descent: 80+.

Reading: Chapter 5.4. Conjugate gradient is supplementary material.