Carnegie Mellon

18-660: Numerical Methods for Engineering Design and Optimization

Xin Li
Department of ECE
Carnegie Mellon University
Pittsburgh, PA 15213

Electrical & Computer ENGINEERING

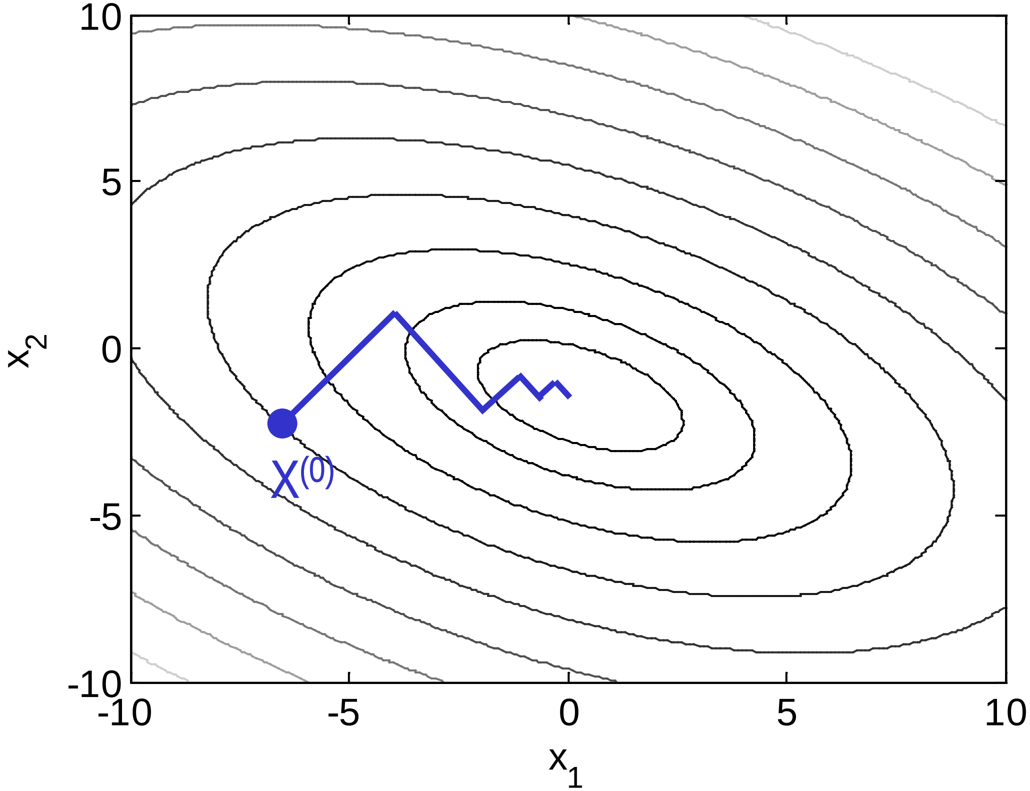
Overview

- Conjugate Gradient Method (Part 2)
  - Conjugate search direction
  - Gram-Schmidt conjugation
  - Conjugate gradient method

Quadratic Programming

- Solve linear equation by quadratic programming

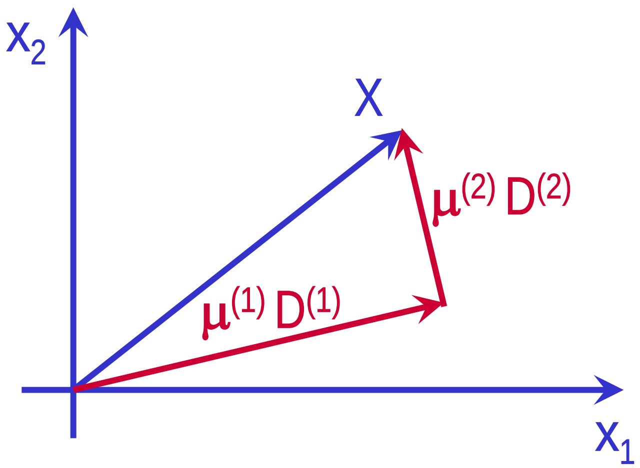
$$AX = B \quad \Rightarrow \quad \min_{X} \; f(X) = \frac{1}{2} X^{T} A X - B^{T} X + C$$



Gradient method has slow convergence

Orthogonal Search Direction

- Ideally, we want to select a set of orthogonal search directions



$$X^{(k+1)} = X^{(k)} + \mu^{(k)} D^{(k)}$$

$$D^{(i)T} D^{(j)} = 0$$

$$\Delta^{(k)} = X^{(k)} - X$$

$$\mu^{(k)} = -\frac{D^{(k)T}\Delta^{(k)}}{D^{(k)T} D^{(k)}}$$

However, we do not know $\Delta^{(k)}$ – otherwise, we know $X = X^{(k)} - \Delta^{(k)}$

Conjugate Search Direction

- We do not know $\Delta^{(k)}$, but we can easily calculate $A\Delta^{(k)}$

$$AX = B \qquad R^{(k)} = B - AX^{(k)} \qquad \Delta^{(k)} = X^{(k)} - X$$

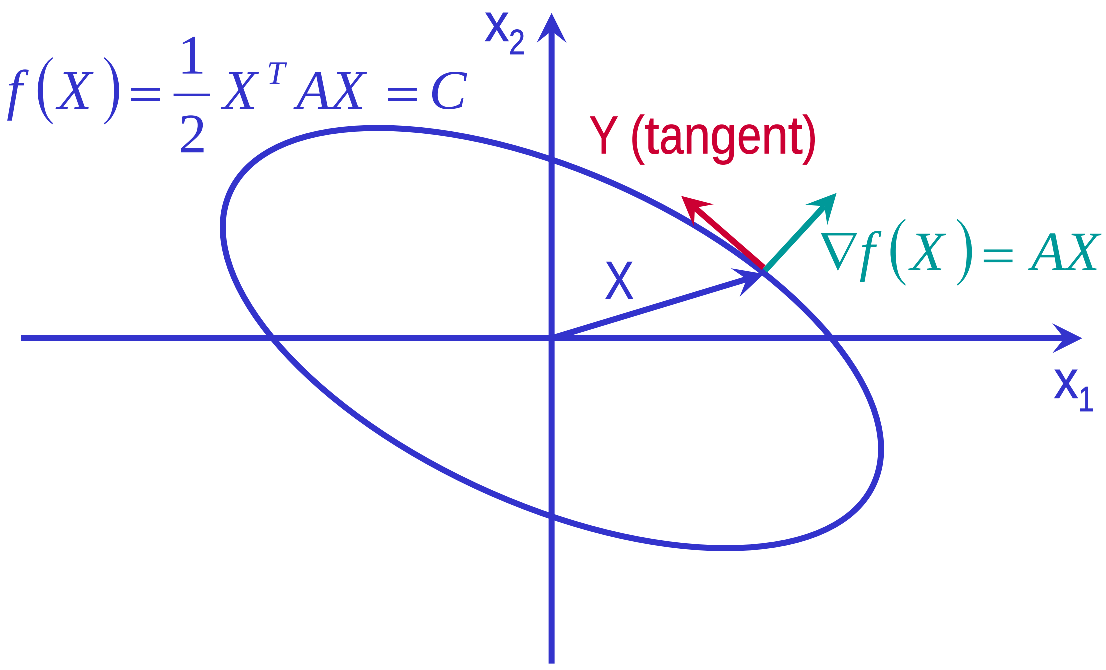
$$A\Delta^{(k)} = AX^{(k)} - AX = AX^{(k)} - B = -R^{(k)}$$

- Instead of using orthogonal directions $D^{(k)}$, we make search directions conjugate (or equivalently A-orthogonal)

Conjugate Search Direction

- Two vectors $D^{(i)}$ and $D^{(j)}$ are conjugate (or A-orthogonal) if $D^{(i)T}AD^{(j)} = 0$ ($i \neq j$)

- Geometrical interpretation

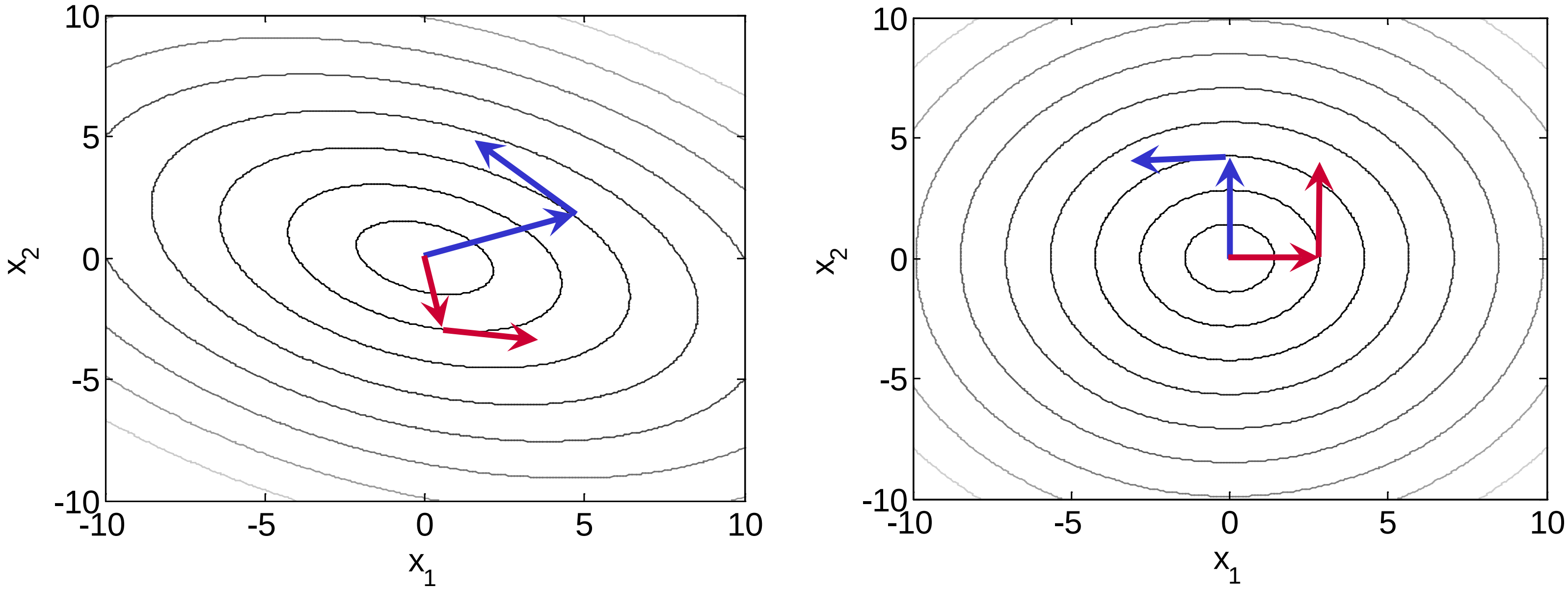


$$Y^T \cdot \nabla f(X) = 0$$

$$Y^T AX = 0$$

Conjugate Example

- "Conjugate" = "orthogonal" if A = identity matrix



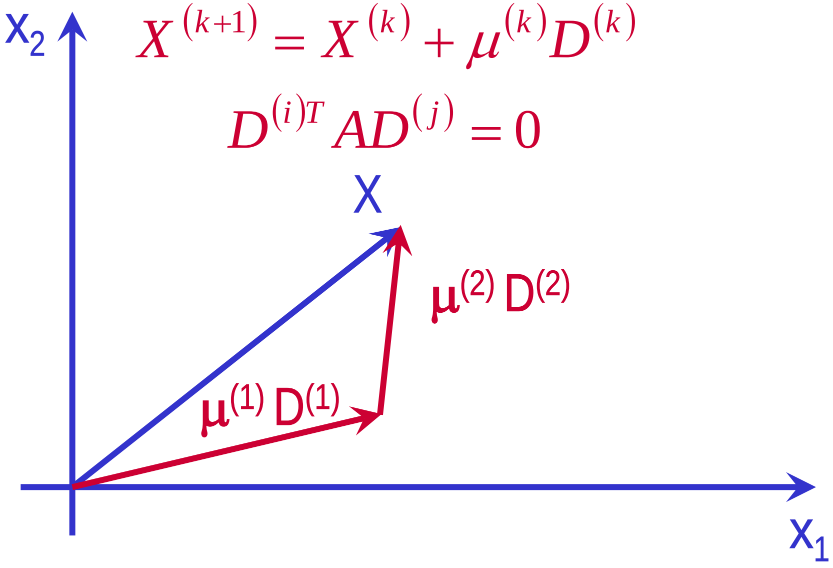
$$A = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 2 \end{bmatrix} \qquad A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Conjugate Search Direction

- Iteration scheme with conjugate search directions
  - Select a set of conjugate search directions $D^{(k)}$
  - Take exactly one iteration step for each direction
  - After at most N steps, we get the solution X
  - (N is the problem size, i.e., $A \in R^{N \times N}$)

$$X^{(k+1)} = X^{(k)} + \mu^{(k)} D^{(k)}$$

$$D^{(i)T} A D^{(j)} = 0$$



How do we decide $\mu^{(k)}$ and $D^{(k)}$?

Conjugate Search Direction

- Determine step size $\mu^{(k)}$

$$D^{(i)T} A D^{(j)} = 0 \qquad X^{(k+1)} = X^{(k)} + \mu^{(k)} D^{(k)} \qquad X = X^{(0)} + \mu^{(0)} D^{(0)} + \cdots + \mu^{(N-1)} D^{(N-1)}$$

$$X^{(k+1)} = X^{(0)} + \mu^{(0)} D^{(0)} + \cdots + \mu^{(k)} D^{(k)}$$

$$\Delta^{(k+1)} = X^{(k+1)} - X = -\mu^{(k+1)} D^{(k+1)} - \cdots - \mu^{(N-1)} D^{(N-1)}$$

$$D^{(k)T} A \Delta^{(k+1)} = D^{(k)T} A \cdot \left[-\mu^{(k+1)} D^{(k+1)} - \cdots - \mu^{(N-1)} D^{(N-1)} \right] = 0$$

$\Delta^{(k+1)}$ and $D^{(k)}$ are conjugate

Conjugate Search Direction

- Determine step size $\mu^{(k)}$

$$A\Delta^{(k+1)} = -R^{(k+1)} \qquad D^{(k)T} A\Delta^{(k+1)} = 0$$

$$D^{(k)T} R^{(k+1)} = 0$$

$R^{(k+1)}$ and $D^{(k)}$ are orthogonal

Conjugate Search Direction

- Determine step size $\mu^{(k)}$

$$X^{(k+1)} = X^{(k)} + \mu^{(k)} D^{(k)} \qquad R^{(k)} = B - AX^{(k)} \qquad D^{(k)T} R^{(k+1)} = 0$$

$$D^{(k)T} \cdot \left[B - AX^{(k+1)}\right] = D^{(k)T} \cdot \left[B - AX^{(k)} - \mu^{(k)} AD^{(k)}\right] = 0$$

$$D^{(k)T} \cdot \left[R^{(k)} - \mu^{(k)} AD^{(k)}\right] = 0$$

$$D^{(k)T} R^{(k)} - \mu^{(k)} D^{(k)T} AD^{(k)} = 0$$

$$\mu^{(k)} = \frac{D^{(k)T} R^{(k)}}{D^{(k)T} AD^{(k)}}$$

Conjugate Search Direction

- $\mu^{(k)}$ minimizes $f[X^{(k+1)}]$ along the direction $D^{(k)}$

$$f(X) = \frac{1}{2} X^T A X - B^T X + C \qquad R^{(k)} = B - AX^{(k)} \qquad X^{(k+1)} = X^{(k)} + \mu^{(k)} D^{(k)}$$

$$\min_{\mu^{(k)}} \quad f\left[X^{(k+1)}\right] = \frac{1}{2} X^{(k+1)T} A X^{(k+1)} - B^T X^{(k+1)} + C$$

$$\frac{d}{d\mu^{(k)}} f\left[X^{(k+1)}\right] = \left[\frac{\partial f}{\partial X^{(k+1)}}\right]^T \cdot \frac{\partial X^{(k+1)}}{\partial \mu^{(k)}} = \left[AX^{(k+1)} - B\right]^T \cdot D^{(k)} = 0$$

$$R^{(k+1)T} \cdot D^{(k)} = 0$$

$R^{(k+1)}$ and $D^{(k)}$ are orthogonal

Conjugate Search Direction

- Important equations about conjugate search direction

$$AX = B$$ → Linear equation

$$\min_X \quad f(X) = \frac{1}{2} X^T A X - B^T X + C$$ → Equivalent optimization

$$R^{(k)} = B - AX^{(k)}$$ → Residual definition

$$\nabla f\left[X^{(k)}\right] = AX^{(k)} - B = -R^{(k)}$$ → Residual vs. gradient

Conjugate Search Direction

- Important equations about conjugate search direction

$$X^{(k+1)} = X^{(k)} + \mu^{(k)} D^{(k)}$$

$$\mu^{(k)} = \frac{D^{(k)T} R^{(k)}}{D^{(k)T} A D^{(k)}}$$

→ Iteration scheme

$$D^{(i)T} A D^{(j)} = 0$$

→ Conjugate search directions

$$D^{(k)T} R^{(k+1)} = 0$$

→ Orthogonal residual

Gradient Method vs. Conjugate Search Direction

Gradient method

$$X^{(k+1)} = X^{(k)} + \mu^{(k)} R^{(k)}$$

$$\mu^{(k)} = \frac{R^{(k)T} R^{(k)}}{R^{(k)T} A R^{(k)}}$$

- Use $R^{(k)}$ as search direction
- $\mu^{(k)}$ is optimized to minimize cost

Conjugate gradient method

$$X^{(k+1)} = X^{(k)} + \mu^{(k)} D^{(k)}$$

$$\mu^{(k)} = \frac{D^{(k)T} R^{(k)}}{D^{(k)T} A D^{(k)}}$$

- Use $D^{(k)}$ as search direction
- $\mu^{(k)}$ is optimized to minimize cost

We need to further develop an algorithm to generate $D^{(k)}$'s that are conjugate

Conjugate Gradient Method

- Construct search directions by conjugation of residuals
  - Residual is directly related to gradient
  - Search directions are defined by conjugation of gradients

$$\min_{X} \quad f(X) = \frac{1}{2} X^T A X - B^T X + C$$

$$\Downarrow$$

$$\nabla f\left[X^{(k)}\right] = A X^{(k)} - B = -R^{(k)}$$

Conjugate Gradient Method

- Define subspace

$$S^{(k)} = span\left\{R^{(0)}, R^{(1)}, \cdots, R^{(K-1)}\right\}$$

K-dimensional space with K basis vectors

- Gradient method directly uses $R^{(k)}$ as search direction

- Conjugate gradient method uses conjugation of $R^{(k)}$ so that each iteration step searches along a different direction
  - Given a set of basis vectors, how do we calculate the conjugation of them?
  - Introduce the algorithm of Gram-Schmidt conjugation

Gram-Schmidt Conjugation

$$D^{(0)} = R^{(0)}$$

$$D^{(1)} = R^{(1)} + \beta_{10} D^{(0)} \qquad D^{(0)T} A D^{(1)} = 0$$

$$D^{(0)T} A D^{(1)} = D^{(0)T} A \cdot \left[R^{(1)} + \beta_{10} D^{(0)} \right] = 0$$

$$D^{(0)T} A R^{(1)} + \beta_{10} D^{(0)T} A D^{(0)} = 0$$

$$\beta_{10} = -\frac{D^{(0)T} A R^{(1)}}{D^{(0)T} A D^{(0)}}$$

Gram-Schmidt Conjugation

$$D^{(k)} = R^{(k)} + \sum_{i=0}^{k-1} \beta_{ki} D^{(i)} \qquad D^{(i)T} A D^{(j)} = 0$$

$$D^{(i)T} A D^{(k)} = D^{(i)T} A \cdot \left[R^{(k)} + \sum_{j=0}^{k-1} \beta_{kj} D^{(j)} \right] = D^{(i)T} A R^{(k)} + \beta_{ki} D^{(i)T} A D^{(i)} = 0$$

$$\beta_{ki} = -\frac{D^{(i)T} A R^{(k)}}{D^{(i)T} A D^{(i)}}$$

Conjugate Gradient Method

- Step 1: start from an initial guess $X^{(0)}$, and set k = 0
- Step 2: calculate

$$D^{(0)} = R^{(0)} = B - AX^{(0)}$$

- Step 3: update solution

$$X^{(k+1)} = X^{(k)} + \mu^{(k)} D^{(k)} \quad \text{where} \quad \mu^{(k)} = \frac{D^{(k)T} R^{(k)}}{D^{(k)T} A D^{(k)}}$$

- Step 4: calculate residual

$$R^{(k+1)} = B - AX^{(k+1)}$$

- Step 5: determine search direction

$$D^{(k+1)} = R^{(k+1)} + \sum_{i=0}^{k} \beta_{k+1,i} D^{(i)} \quad \text{where} \quad \beta_{k+1,i} = -\frac{D^{(i)T} A R^{(k+1)}}{D^{(i)T} A D^{(i)}}$$

- Step 6: set k = k + 1 and go to Step 3

Conjugate Gradient Method

- This simple implementation is not numerically efficient
- There are a number of numerical tricks that we can apply to reduce computational complexity
- Key idea
  - $X^{(k)}$, $D^{(k)}$ and $R^{(k)}$ are strongly correlated
  - They can be computed in many different ways – we should use the most efficient algorithm in our implementation
  - More details in next lecture

Summary

- Conjugate gradient method (Part 2)
  - Conjugate search direction
  - Gram-Schmidt conjugation
  - Conjugate gradient method