THE UNIVERSITY OF UTAH

SCHOOL OF COMPUTING

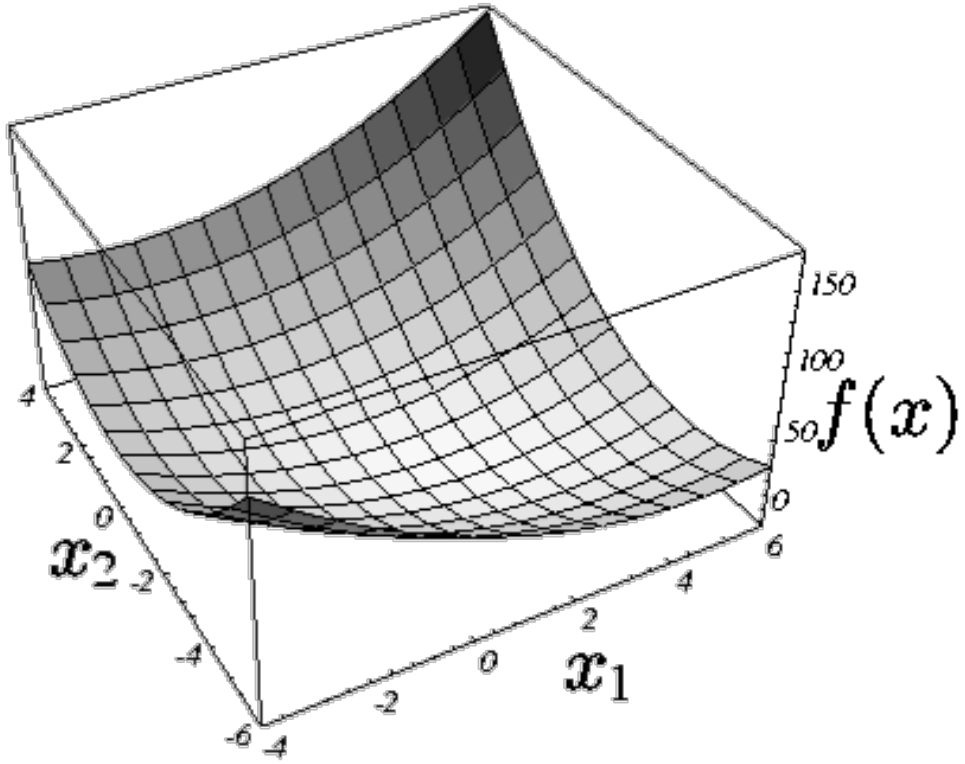
CS 3200 – Introduction to Scientific Computing

Instructor: Chris Johnson

Topic: Conjugate Gradient (CG) Method

For Symmetric Positive-Definite Matrix $\boldsymbol{A}$

- Solve the system $\boldsymbol{A}x = b$
- Graph of the quadratic form $f(x) = \frac{1}{2}x^T\boldsymbol{A}x - b^Tx + c$:



- The minimum point of the surface is the solution to $\boldsymbol{A}x = b$

from J. R. Shewchuk “Painless CG"

Steepest Descent Method

- Quadratic function of vector x:

$$f(x) = \frac{1}{2}x^T \mathbf{A}x - b^T x + c$$

- Derivative:

$$f'(x) = \frac{1}{2}\mathbf{A}^T x + \frac{1}{2}\mathbf{A}x - b$$

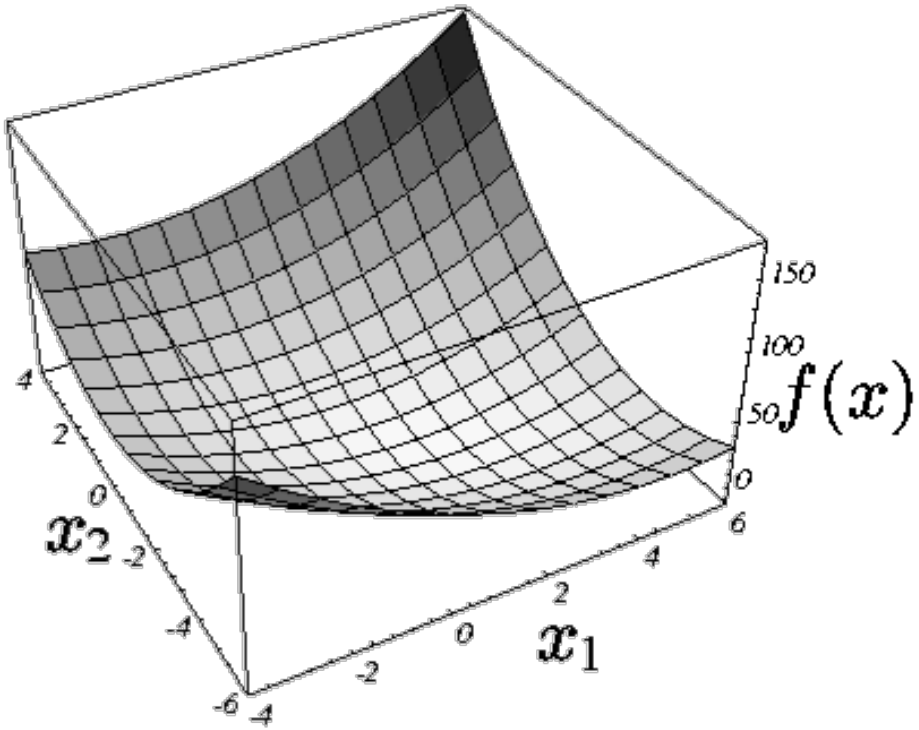
- If $\boldsymbol{A}$ is symmetric

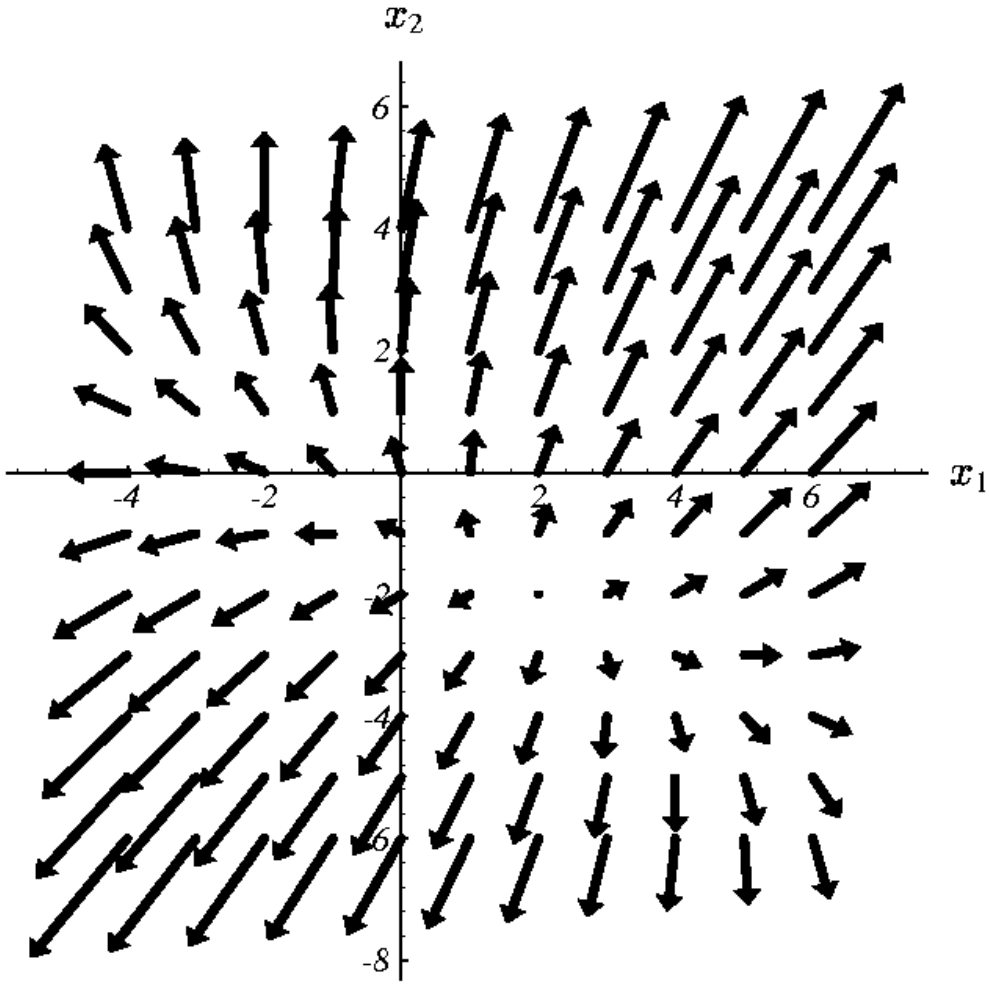
$$f'(x) = \mathbf{A}x - b$$

- If $\boldsymbol{A}$ is also positive-definite (i.e., $x^T \mathbf{A}x > 0$ for any nonzero vector x)
  - Minimize $f(x)$ by setting $f'(x)$ to 0

$$\mathbf{A}x = b$$

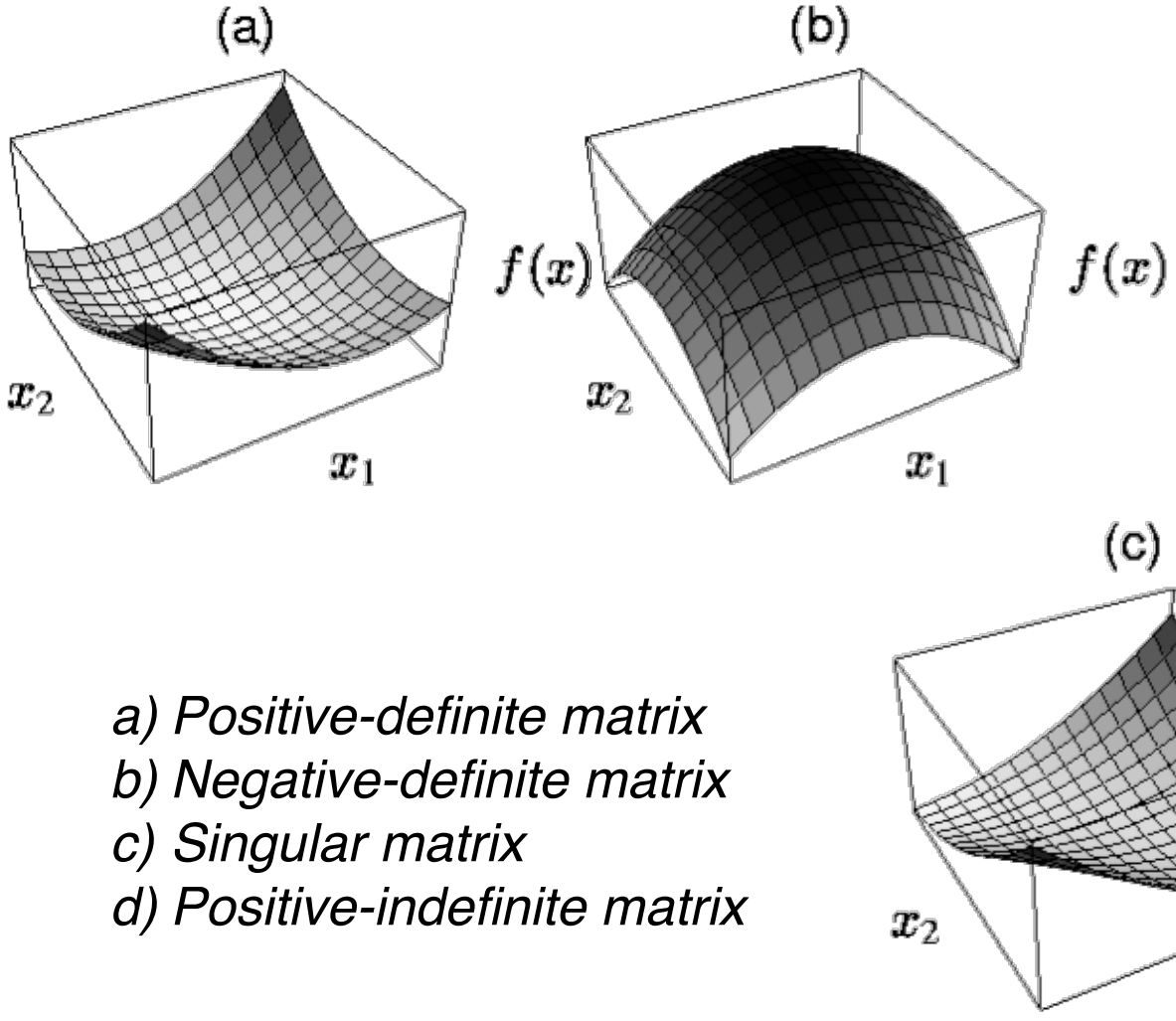
Gradient of Quadratic Form

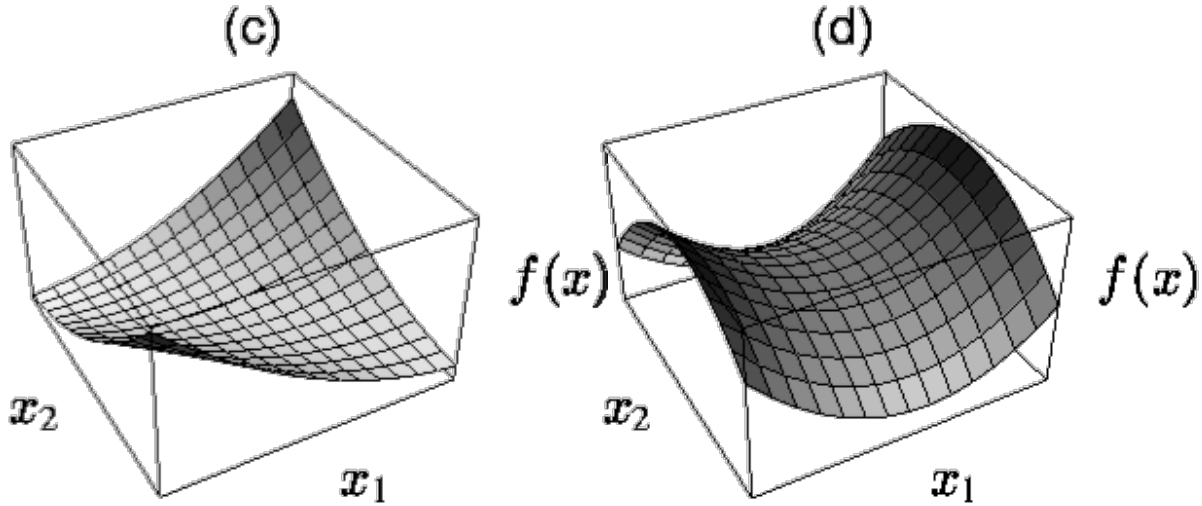




from J. R. Shewchuk “Painless CG"

For A Not Positive-Definite





a) Positive-definite matrix
b) Negative-definite matrix
c) Singular matrix
d) Positive-indefinite matrix

from J. R. Shewchuk “Painless CG"

Non-Stationary Iterative Method

- Start from initial guess x_0; adjust until close enough to the exact solution:

$$x_{i+1} = x_i + a_i p_i$$

  - p_i - *Adjustment Direction*
  - a_i - *Step Size*

- How to choose direction and step size?

Steepest Descent Method

- Choose the direction in which f decreases most quickly – the direction opposite to $f'(x_i)$:

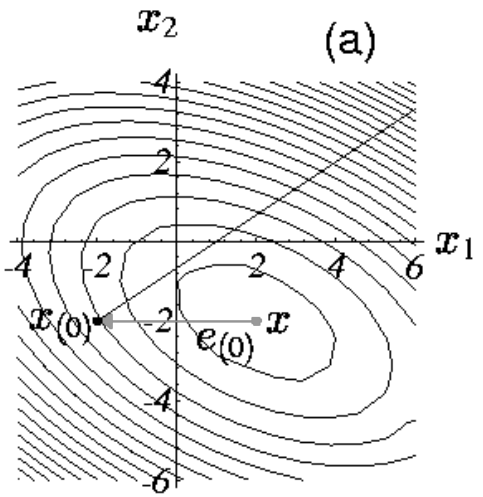
$$r_i = -f'(x_i) = b - Ax_i$$

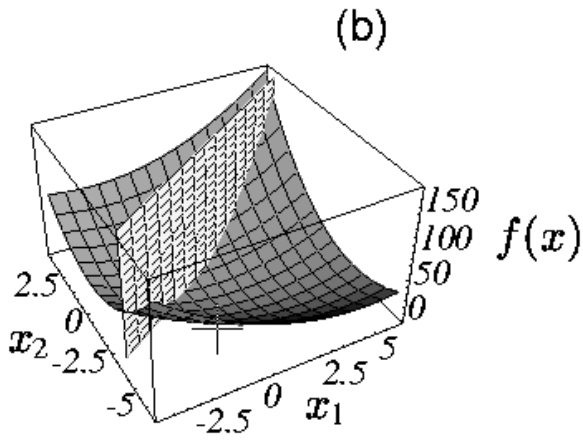
- Which is also the direction of the residual:

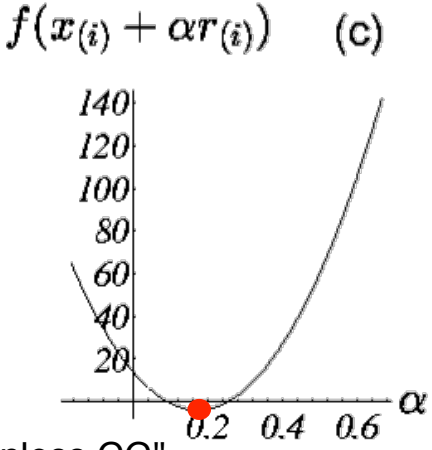
$$x_{i+1} = x_i + a_i r_i$$

Steepest Descent Method

- Direction of steepest descent is plane in 3 dimensions
- Intersection of plane with function produces parabola
- Minimum of parabola determines desired step size







from J. R. Shewchuk “Painless CG"

Steepest Descent Method

- How to choose step size?
  - a_i should minimize f, along the direction of r_i, which means

$$\frac{d}{da} f(x_{i+1}) = 0$$

$$\frac{d}{da} f(x_{i+1}) = f'(x_{i+1})^T \frac{d}{da} x_{i+1} = f'(x_{i+1})^T r_i = 0$$

$$\Rightarrow {r_{i+1}}^T r_i = 0$$

$$\Rightarrow (b - Ax_{i+1})^T r_i = 0$$

$$\Rightarrow (b - A(x_i + a_i r_i))^T r_i = 0$$

$$\Rightarrow (b - Ax_i)^T r_i = a_i (Ar_i)^T r_i$$

$$\Rightarrow \boxed{a_i = \frac{{r_i}^T r_i}{{r_i}^T A r_i}}$$

Steepest Descent Algorithm

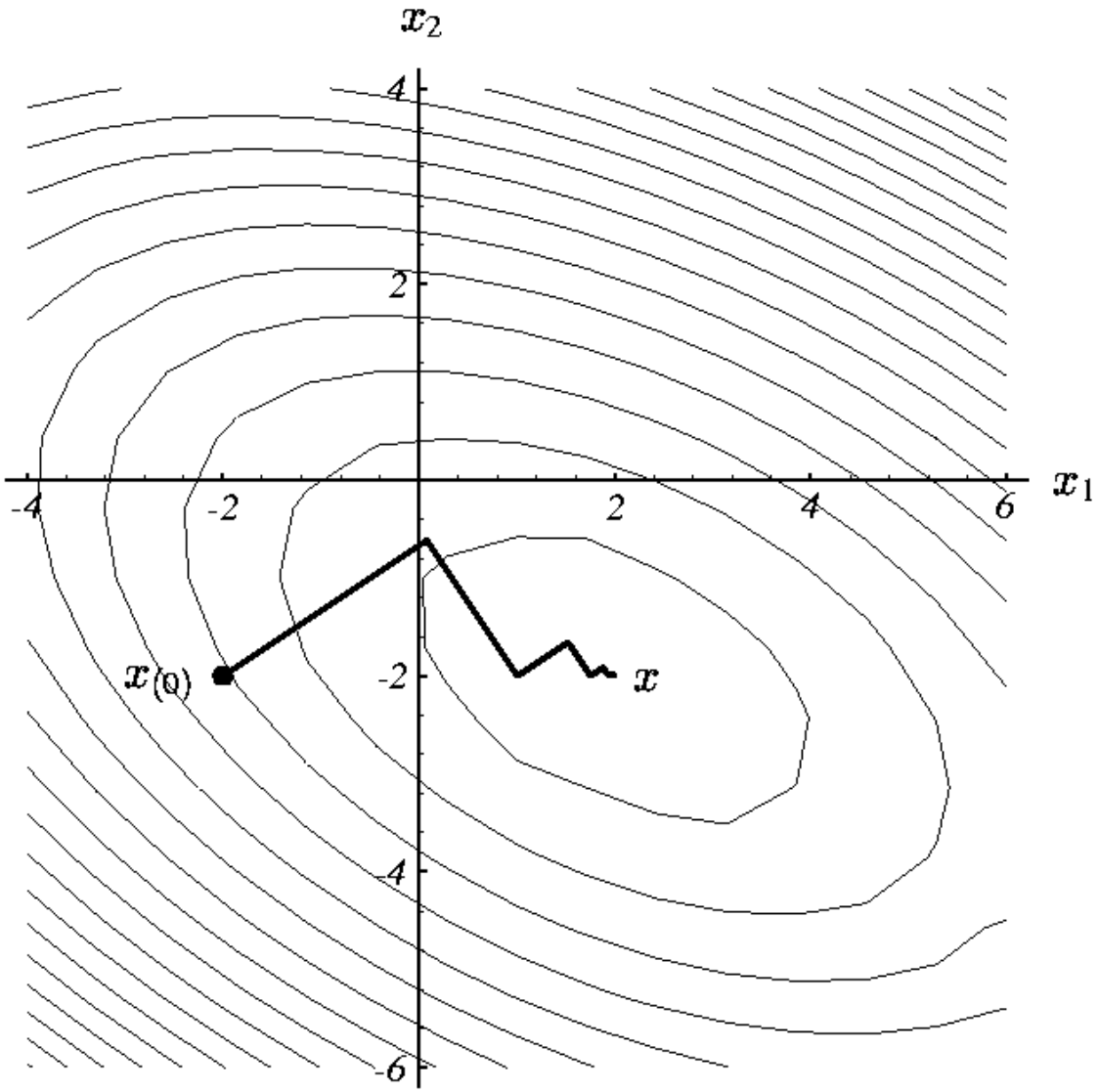
- Given x_0, iterate until the residue is smaller than the error tolerance:

$$r_i = b - Ax_i$$

$$a_i = \frac{{r_i}^T r_i}{{r_i}^T A r_i}$$

$$x_{i+1} = x_i + a_i r_i$$

Iterations of Steepest Descent Method

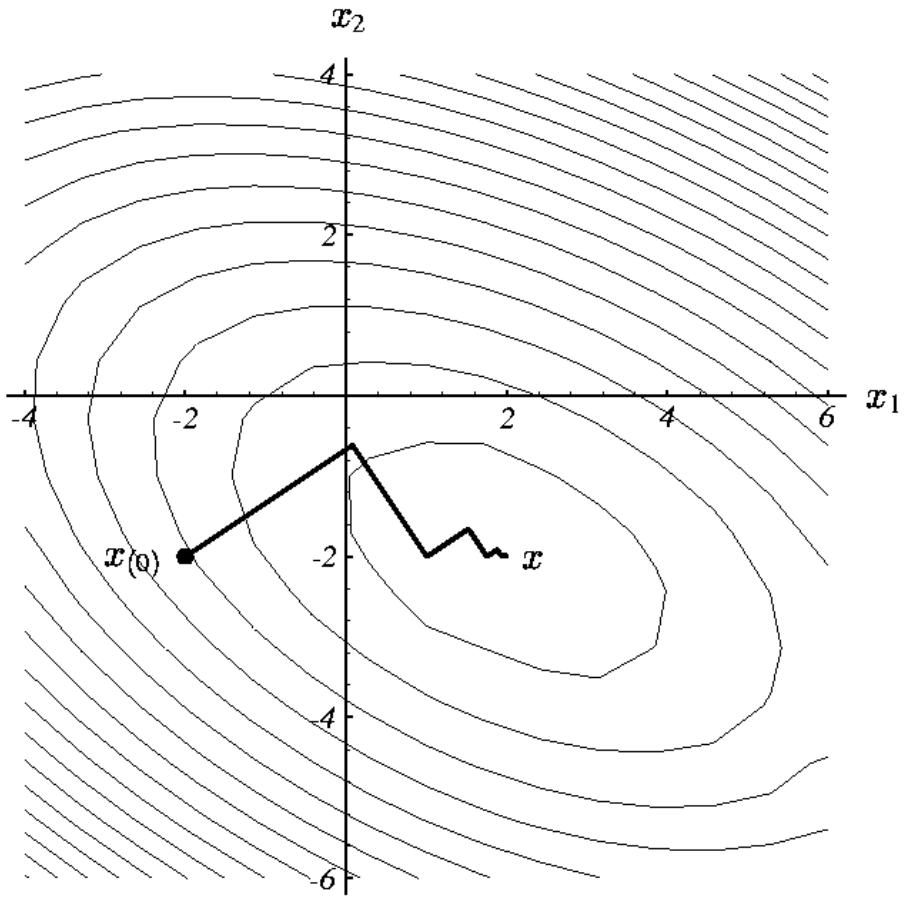


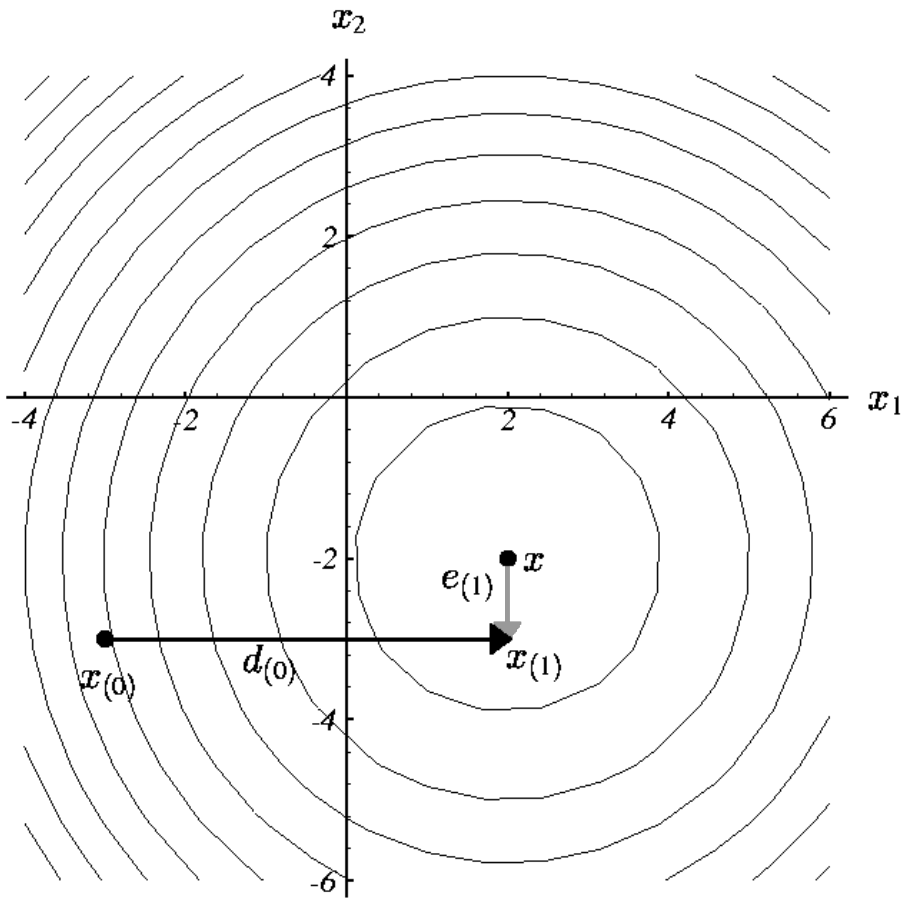
from J. R. Shewchuk “Painless CG"

Steepest Descent Method

- The steepest descent method is very reliable - it can always make progress provided the gradient is nonzero
- Depending on the function, however, steepest descents can take a long time to converge to the true solution
- Instead of looking at a gradient direction, we can speed up minimization by searching in the conjugate direction
  - The error after i steps is orthogonal to all previous search directions

Conjugate Gradient (CG) Method





from J. R. Shewchuk “Painless CG"

Conjugate Gradient Method

- The search directions are orthogonal (so-called A-orthogonal or conjugate)
- The error is minimal over the space spanned by the search directions
- Minimum error implies reaching an exact solution in at most n steps
  - Because of rounding errors, there is a loss of orthogonality

Orthogonal Directions

- Pick orthogonal search directions: $d_0, d_1, \ldots, d_{n-1}$

$$x_{i+1} = x_i + a_i d_i$$

$$d_i^T e_{i+1} = 0$$

$$d_i^T (e_i + a_i d_i) = 0$$

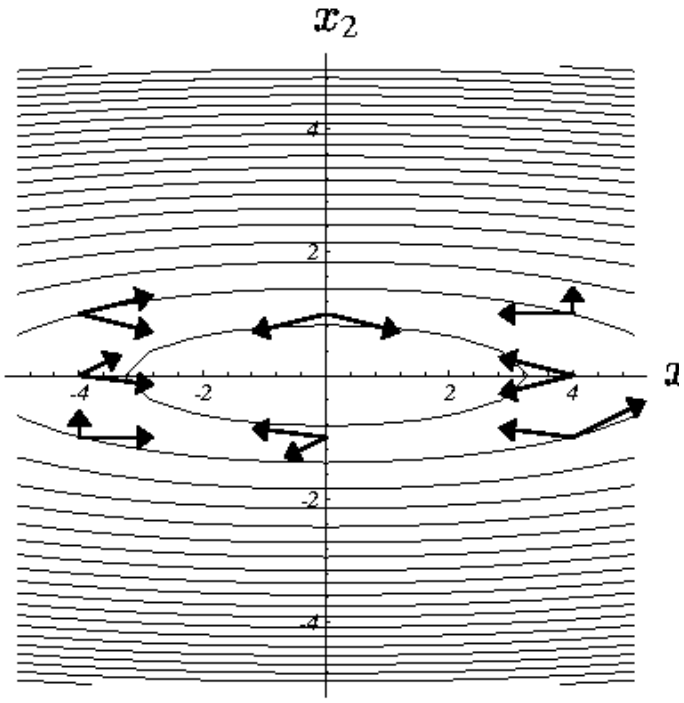
$$a_i = -\frac{d_i^T e_i}{d_i^T d_i}$$

- Problem – we don't know e_i

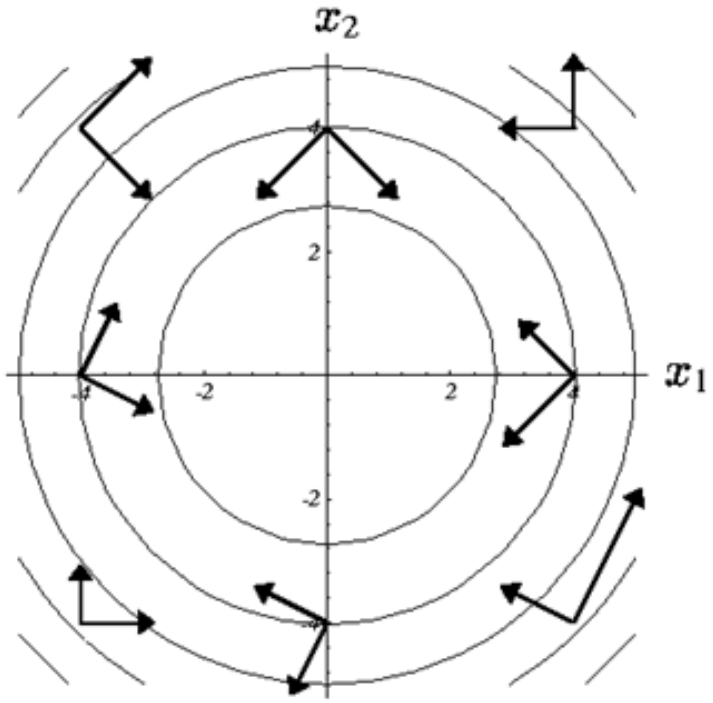
A-Orthogonal Directions

- Instead of orthogonal search directions, we make the search directions *A–orthogonal* (*conjugate*)

$$r_i^T A d_i = 0$$



These pairs of vectors are A-orthogonal . . .



. . . because these pairs of vectors are orthogonal.

from J. R. Shewchuk “Painless CG"

Conjugate Gradient Method

- New search directions:

$$x_{i+1} = x_i + \alpha_i d_i$$

- Optimum orthogonality:

$$0 = d_i^T e_{i+1}$$

- We say that two non-zero vectors u and v are conjugate (with respect to $\boldsymbol{A}$) if $u^T A v = 0$
- Look at the conjugation:

$$0 = d_i^T A e_{i+1}$$

Search Step Size

- $\frac{d}{da} f(x_{i+1}) = f'(x_{i+1})^T \frac{d}{da} x_{i+1} = f'(x_{i+1})^T d_i = 0$

$$\Rightarrow r_{i+1}^T d_i = 0$$

$$\Rightarrow (b - Ax_{i+1})^T d_i = 0$$

$$\Rightarrow (b - A(x_i + a_i d_i))^T d_i = 0$$

$$\Rightarrow (b - Ax_i)^T d_i = a_i (Ad_i)^T d_i$$

$$\Rightarrow \boxed{a_i = \frac{r_i^T d_i}{d_i^T A d_i}}$$

Conjugate Gradient Algorithm

- Start with

$$d_0 = r_0 = b - \boldsymbol{A}x_0$$

- Iterate:

$$\alpha_i = \frac{d_i^T r_i}{d_i^T \boldsymbol{A} d_i} \quad \leftarrow \text{scalar}$$

$$x_{i+1} = x_i + \alpha_i d_i \quad \leftarrow \text{vector}$$

$$r_{i+1} = r_i - \alpha_i \boldsymbol{A} d_i \quad \leftarrow \text{vector}$$

$$\beta_{i+1} = \frac{r_{i+1}^T r_{i+1}}{r_i^T r_i} \quad \leftarrow \text{scalar}$$

$$d_{i+1} = r_{i+1} + \beta_{i+1} d_i \quad \leftarrow \text{vector}$$

Conjugate Gradient Algorithm

- Initialization:

$$x_0 = [0 \quad 0]'$$

$$d_0 = r_0 = b - Ax_0$$

- Iteration:

$$\alpha_0 = \frac{d_0^T r_0}{d_0^T A d_0}$$

$$x_1 = x_0 + \alpha_0 d_0$$

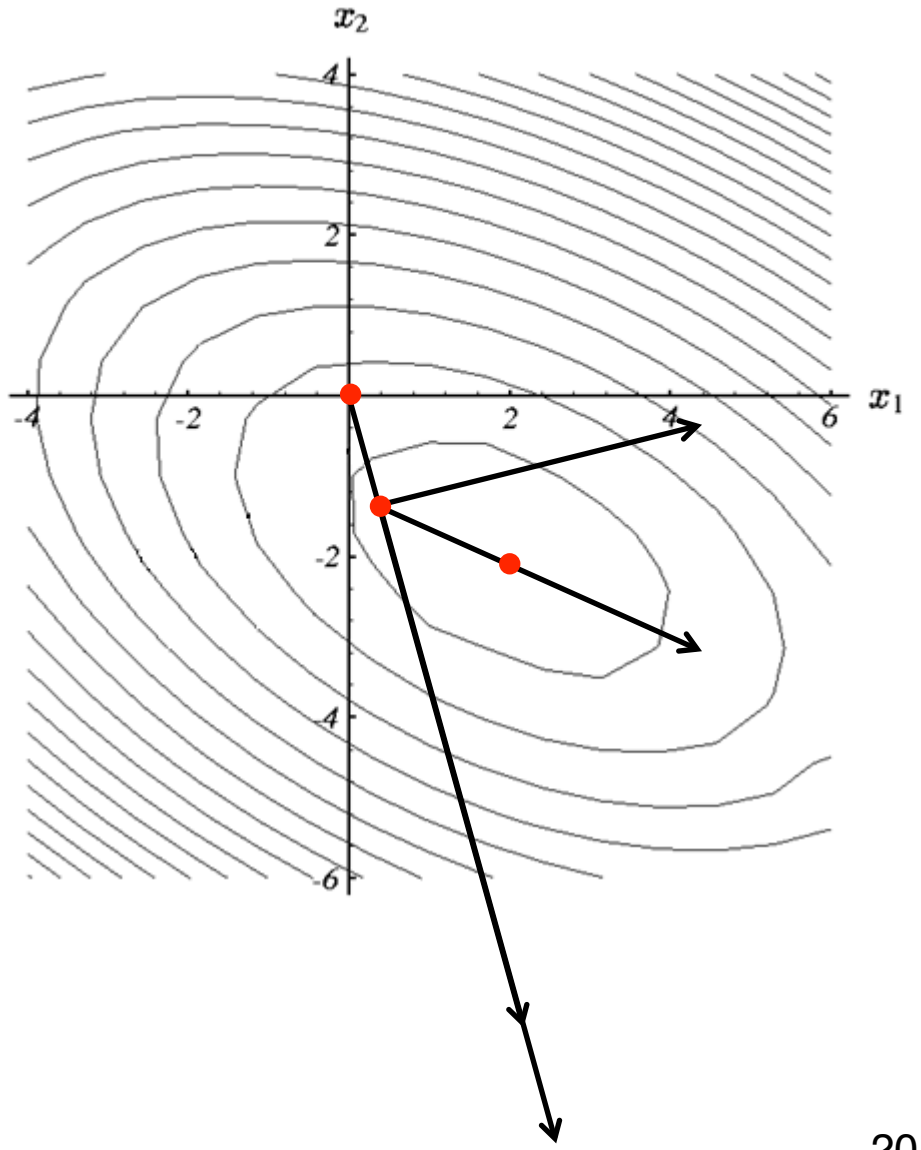
$$r_1 = r_0 - \alpha_0 A d_0$$

$$\beta_1 = \frac{r_1^T r_1}{r_0^T r_0}$$

$$d_1 = r_1 + \beta_1 d_0$$

$$\alpha_1 = \frac{d_1^T r_1}{d_1^T A d_1}$$

$$x_2 = x_1 + \alpha_1 d_1$$



Preconditioning

- In exact arithmetic, CG converges in n steps (completely unrealistic)

- The convergence rate of CG can be accelerated by preconditioning

- Apply the CG algorithm to $M^{-1}A$
  - Choose M so that $M^{-1}A$ is better conditioned and systems of the form $My = z$ are easily solved
  - Then solve $(M^{-1}A)x = M^{-1}b$

- Possible choices for preconditioning include
  - Diagonalization
  - Incomplete LU factorization
  - Approximate inverse
  - SSOR

Conjugate Gradient – Other resources

- Shewchuk's introduction to steepest descent and CG (cited in these slides):
  - http://www.cs.cmu.edu/~quake-papers/painless-conjugate-gradient.pdf
- Another set of slides on CG, with information on preconditioners:
  - http://www.stanford.edu/class/ee364b/lectures/conj_grad_slides.pdf
- A good online text for iterative methods (see section 5.3 for steepest descent & 6.7 for CG):
  - http://www-users.cs.umn.edu/~saad/IterMethBook_2ndEd.pdf

Conjugate Gradient – Other resources

- Applets illustrating the steepest descent and CG methods, from a site with an extensive collection of applets for scientific computing concepts and methods:
  - http://www.cse.illinois.edu/iem/optimization/SteepestDescent/
  - http://www.cse.illinois.edu/iem/optimization/ConjugateGradient/

Steepest Descent

- Suppose that we want to find a local minimum for the scalar function f of the vector variable x, starting from an initial point x_0. Picking an appropriate x_0 is crucial but also very problem-dependent. We start from x_0 and go downhill. At every step of the way, we must make the following decisions:
  - When to stop?
  - In what direction to proceed?
  - How long a step to take?

Steepest Descent: An Optimization Problem

- Quadratic function of vector x

$$f(x) = \frac{1}{2} x^T A x - b^T x + c$$

- If $x^T \mathbf{A} x > 0$ for any nonzero vector x
  - Matrix $\boldsymbol{A}$ is positive-definite

- If $\boldsymbol{A}$ is symmetric and positive-definite
  - $f(x)$ is minimized by the solution $\mathbf{A}x = b$

Symmetric Positive-Definite Matrix

- If A is symmetric and positive-definite:
  - p is an arbitrary point
  - x is the solution point

$$x = A^{-1}b$$

$$f(p) = f(x) + \frac{1}{2}(p - x)^T A(p - x)$$

  - Since

$$\frac{1}{2}(p - x)^T A(p - x) > 0$$

  - We have

$$f(p) > f(x)$$

Steepest Descent: Example

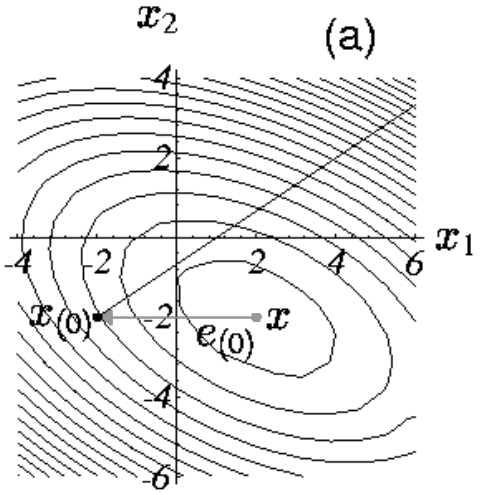
- $\begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2 \\ -8 \end{bmatrix}$

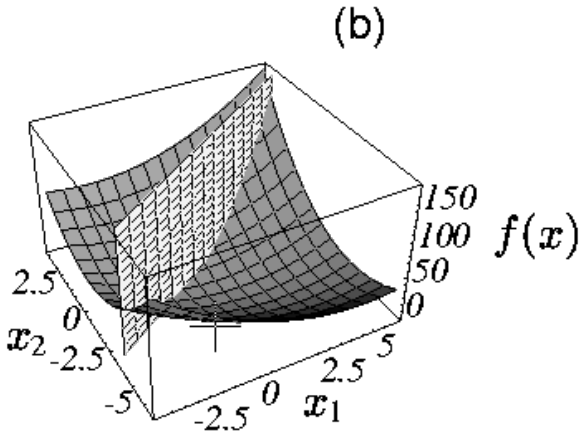
a) Starting at $(-2, -2)$, take the direction of steepest descent of f

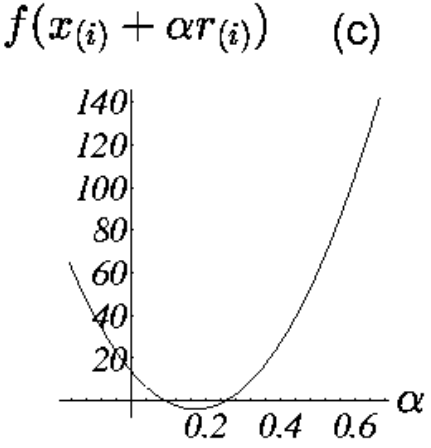
b) Find the point on the intersection of these two surfaces that minimizes f

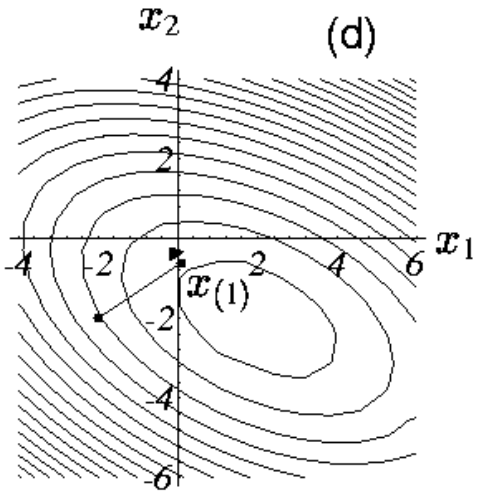
c) Intersection of surfaces

d) The gradient at the bottommost point is orthogonal to the gradient of the previous step





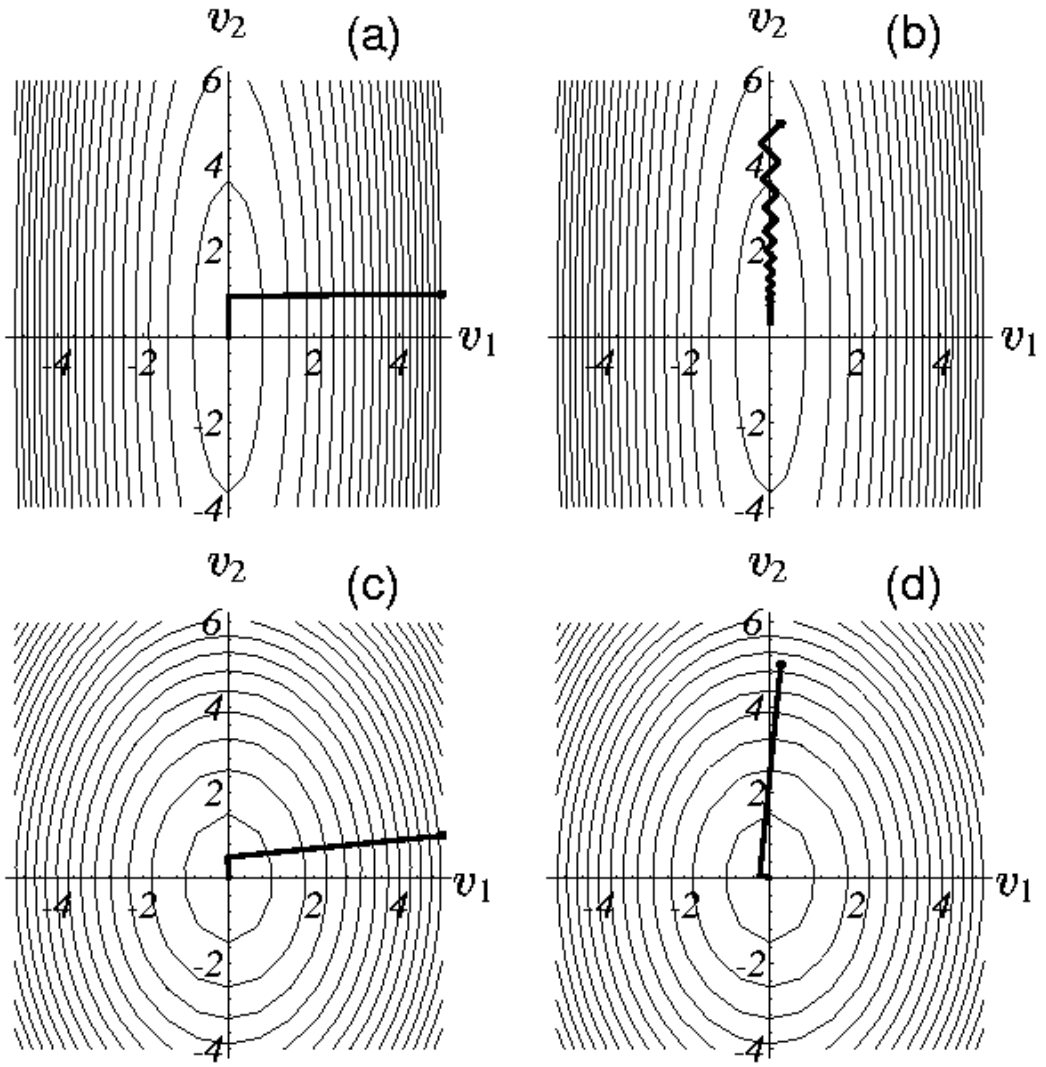




Case Study

-

a) Large κ, small μ
b) An example of poor convergence – κ and μ and both large
c) Small κ and μ
d) Small κ, large μ



from J. R. Shewchuk “Painless CG"

Other Krylov Subspace Methods

- Nonsymmetric linear systems:
  - GMRES:
    for $i = 1, 2, 3, \ldots$
    find $x_i \in K_i(A, b)$ such that $r_i = (Ax_i - b) \perp K_i(A, b)$
    But, no short recurrence => save old vectors => lots more space (usually "restarted" every k iterations to use less space)
  - BiCGStab, QMR, etc.:
    Two spaces $K_i(A, b)$ and $K_i(A^T, b)$ w/ mutually orthogonal bases Short recurrences => O(n) space, but less robust
  - Convergence and preconditioning more delicate than CG
  - Active area of current research
- Eigenvalues: Lanczos (symmetric), Arnoldi (nonsymmetric)

from J. R. Shewchuk "Painless CG"

Conjugate Gradient: Convergence

- In exact arithmetic, CG converges in n steps (completely unrealistic)
- Accuracy after k steps of CG is related to:
  - Consider polynomial of degree k that is equal to 1 at step 0
  - How small can such a polynomial be at all the eigenvalues of A?
- Condition number:

$$\kappa(A) = \|A\|_2 \|A^{-1}\|_2 = \lambda_{max}(A)/\lambda_{min}(A)$$

- Residual is reduced by a constant factor over $\mathcal{O}(\mathrm{K}^{1/2}(A))$ iterations of CG

from J. R. Shewchuk “Painless CG"

Preconditioning

- The convergence rate of CG can be accelerated by preconditioning
- Apply the CG algorithm to $M^{-1}A$ where M is chosen so that $M^{-1}A$ is better conditioned and systems in the form of Mz=y are easily solved
- Possible choices for preconditioners include:
  - Diagonal
  - Incomplete LU factorization
  - Approximate inverse
  - SSOR

Preconditioners

- Suppose you had a matrix B such that:
  1. Condition number $\kappa(B^{-1}A)$ is small
  2. By = z is easy to solve

- Then you could solve $(B^{-1}A)x = B^{-1}b$ instead of $Ax = b$

- B = A is great for (1), not for (2)
- B = I is great for (2), not for (1)
- Domain-specific approximations sometimes work
- B = diagonal of A sometimes works

- Better: blend in some direct-methods ideas. . .

from J. R. Shewchuk “Painless CG"

Preconditioned Conjugate Gradient Iteration

$x_0 = 0,\ r_0 = b,\ d_0 = B^{-1}r_0,\ y_0 = B^{-1}r_0$

for $k = 1,2,3, \ldots$

$\alpha k = (y_{k-1}^T r_{k-1}) / (d_{k-1}^T A d_{k-1})$	step length
$\mathrm{x_k} = x_{k-1} + \alpha_k d_{k-1}$	approx solution
$\mathrm{r}_k = r_{k-1} - \alpha_k A d_{k-1}$	residual
$\mathrm{y_k} = B^{-1} r_k$	preconditioning solve
$\beta_k = (y_k^T r_k)/(y_{k-1}^T r_{k-1})$	improvement
$\mathrm{d_k} = y_k + \beta_k d_{k-1}$	search direction

- One matrix-vector multiplication per iteration
- One solve with preconditioner per iteration

from J. R. Shewchuk “Painless CG"

Enter the Conjugate Gradient Method

- The conjugate gradient method was originally proposed in Magnus R. Hestenes and Eduard Stiefel (1952), Methods of conjugate gradients for solving linear systems, J. Research Nat. Bur. Standards 49, 409–436.