Optimization

Eung Je Woo

Department of Biomedical Engineering

Impedance Imaging Research Center (IIRC)

Kyung Hee University

Korea

ejwoo@khu.ac.kr

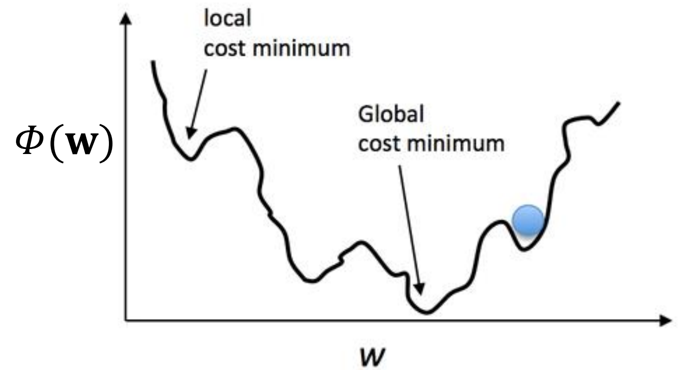
Optimization

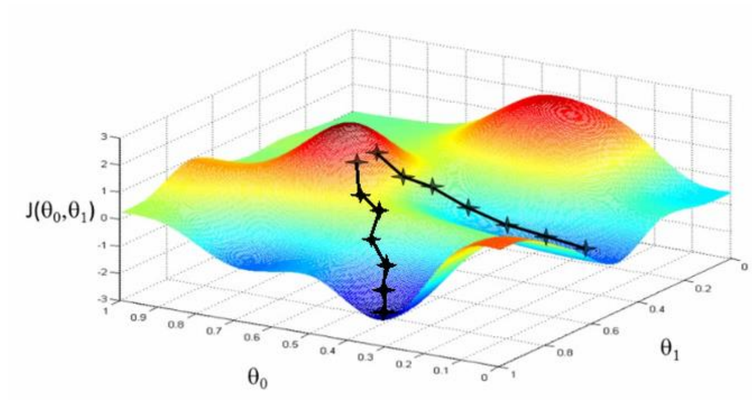
- Generic minimization problem: $\min_{\mathbf{w}} \Phi(\mathbf{w})$ or $\mathbf{w}^* = \underset{\mathbf{w}}{\operatorname{argmin}}\, \Phi(\mathbf{w})$

- $\mathbf{u}^*$ is a local minimum if $\nabla\Phi(\mathbf{u}^*) = \mathbf{0}$ and $\nabla^2\Phi(\mathbf{u}^*) > 0$ (positive definite) where

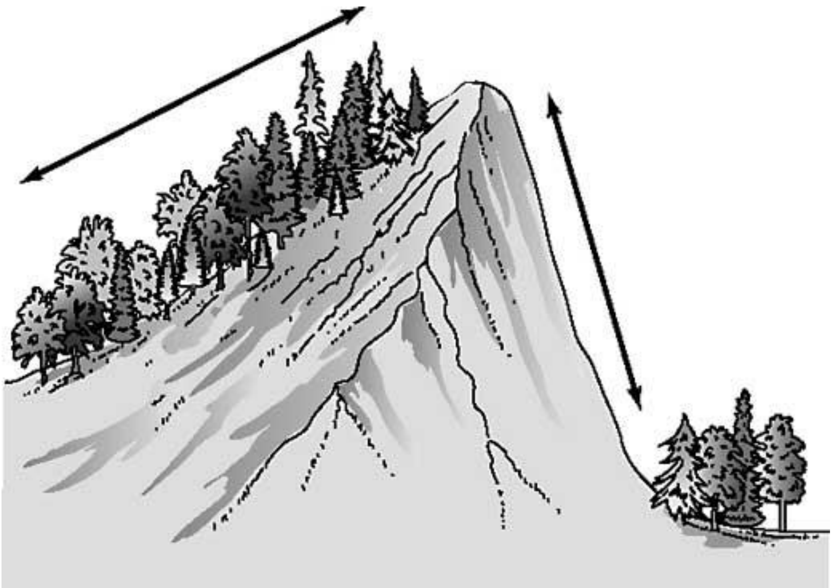
$$\nabla\Phi(\mathbf{u}) = \begin{bmatrix} \frac{\partial\Phi}{\partial u_1} & \frac{\partial\Phi}{\partial u_2} & \cdots & \frac{\partial\Phi}{\partial u_m} \end{bmatrix}^T$$

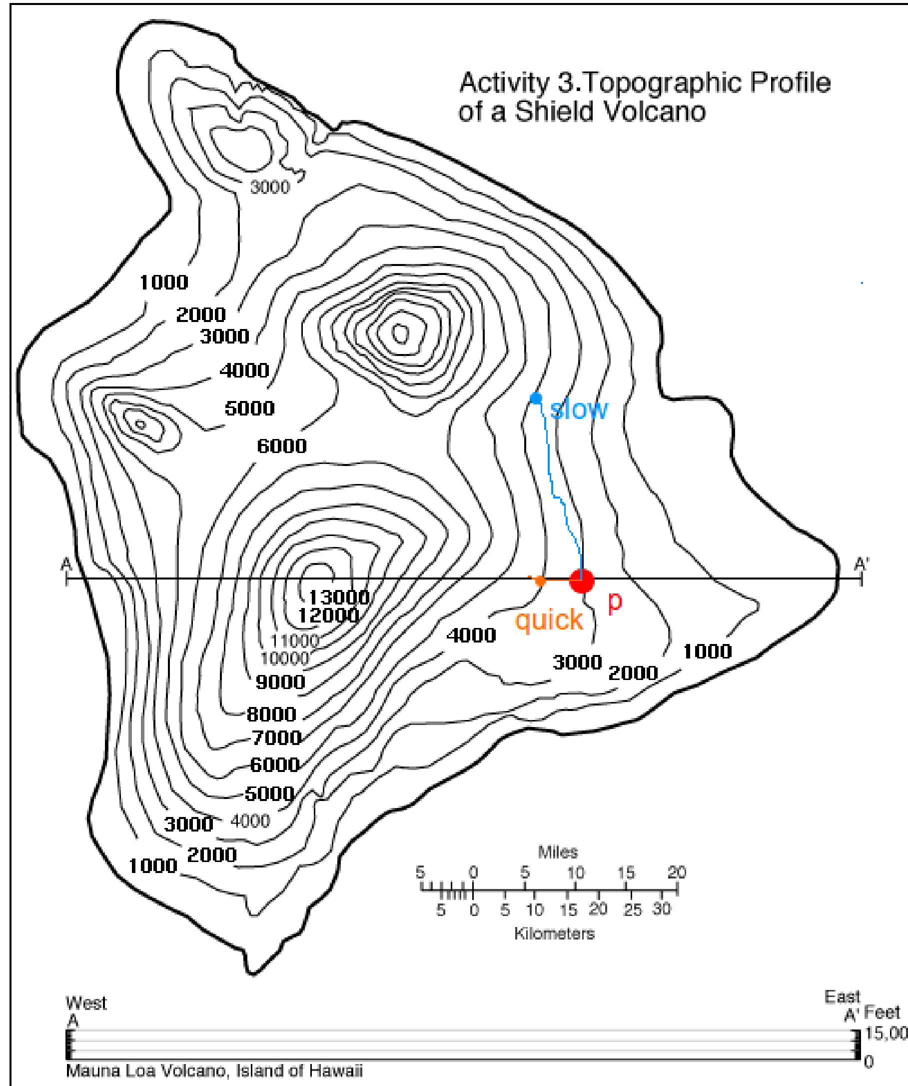
$$\nabla^2\Phi(\mathbf{u}) = \begin{bmatrix} \frac{\partial^2\Phi}{\partial u_1^2} & \frac{\partial^2\Phi}{\partial u_1 \partial u_2} & \cdots & \frac{\partial^2\Phi}{\partial u_1 \partial u_m} \\ \frac{\partial^2\Phi}{\partial u_2 \partial u_1} & \frac{\partial^2\Phi}{\partial u_2^2} & \cdots & \frac{\partial^2\Phi}{\partial u_2 \partial u_m} \\ \vdots & \vdots & & \vdots \\ \frac{\partial^2\Phi}{\partial u_m \partial u_1} & \frac{\partial^2\Phi}{\partial u_m \partial u_2} & \cdots & \frac{\partial^2\Phi}{\partial u_m^2} \end{bmatrix}$$





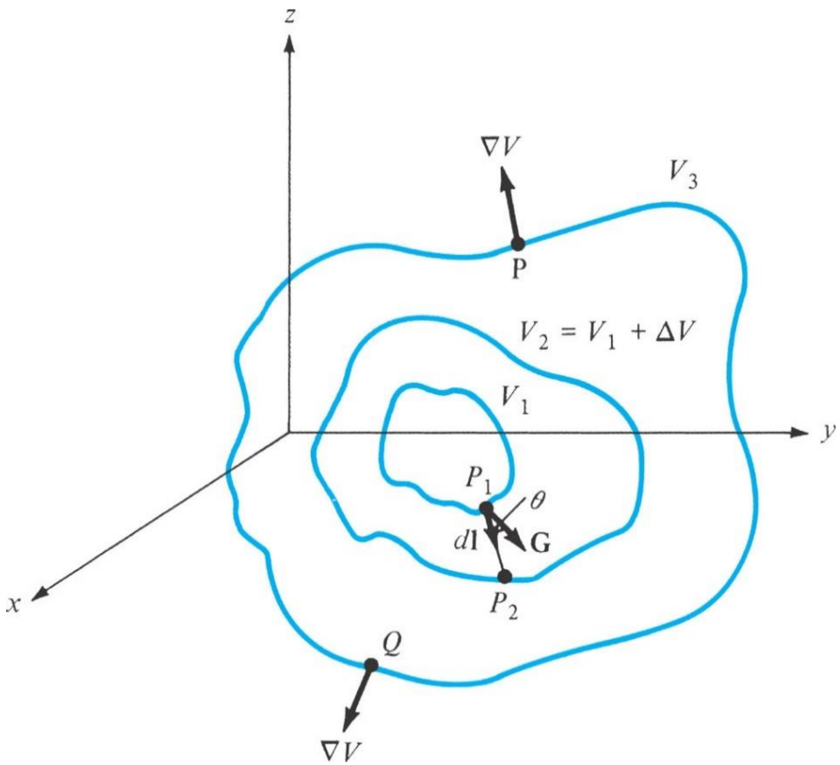
Slope in Three Dimension

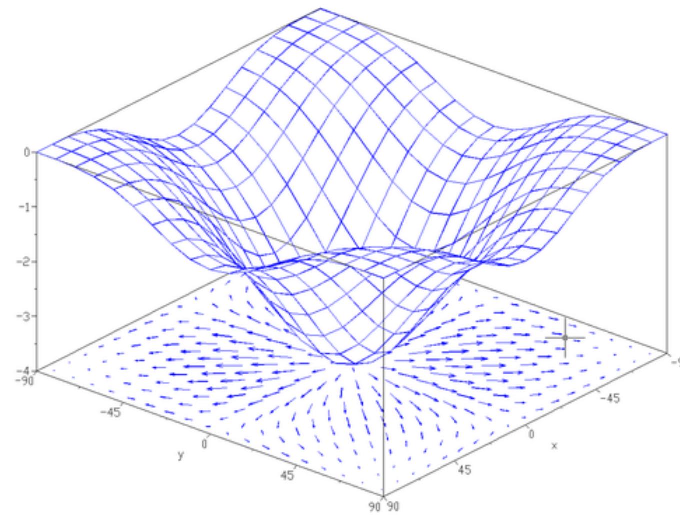


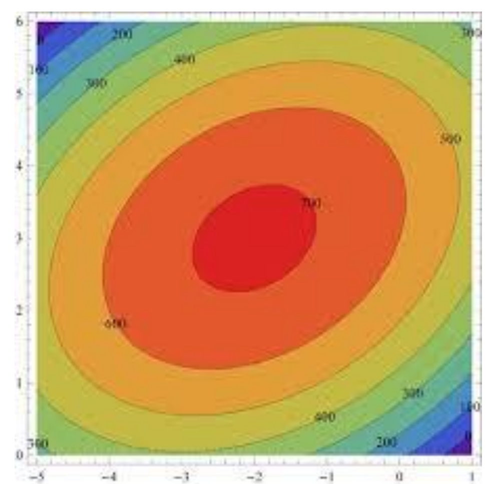


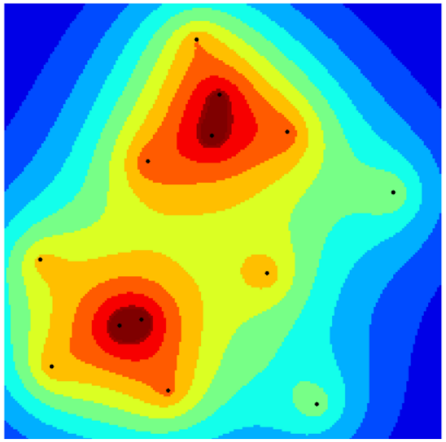
Gradient

$$\text{grad}\, V = \nabla V = \frac{\partial V}{\partial x}\vec{a}_x + \frac{\partial V}{\partial y}\vec{a}_y + \frac{\partial V}{\partial z}\vec{a}_z$$



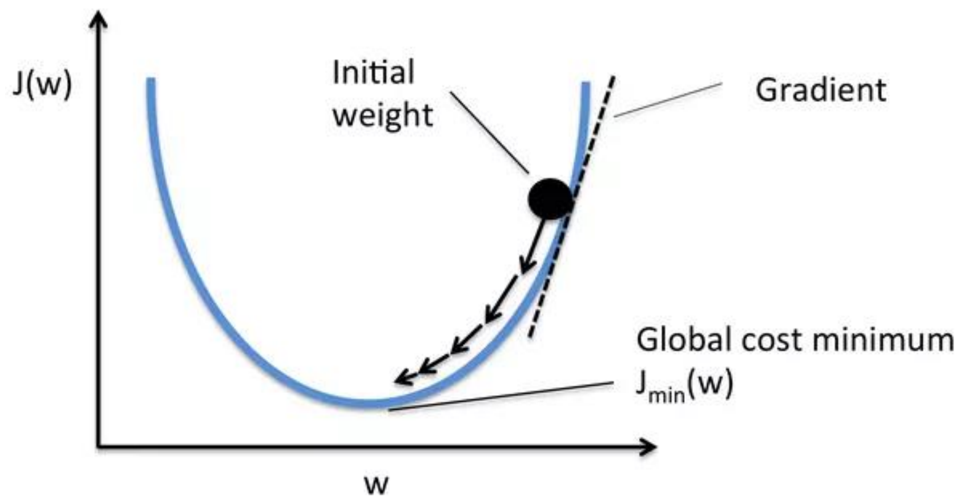






Steepest Descent Algorithm

- Gradient search algorithm
- Initial guess $\mathbf{u}^0$
- Iterate $\mathbf{u}^{k+1} = \mathbf{u}^k - h\frac{\nabla\Phi(\mathbf{u}^k)}{\|\nabla\Phi(\mathbf{u}^k)\|}$ for a chosen step size h
- How to stop?



Iterative Search to a Local Minimum

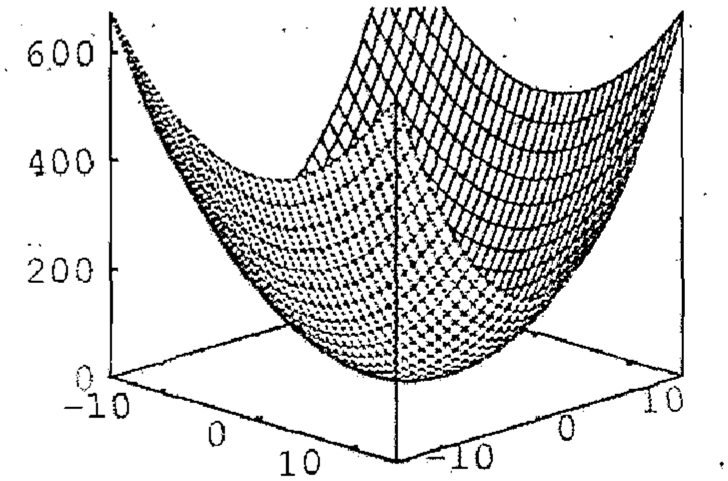


Fig. 1.5: The graph of $f(x, y) = x^2 + 2y^2$.

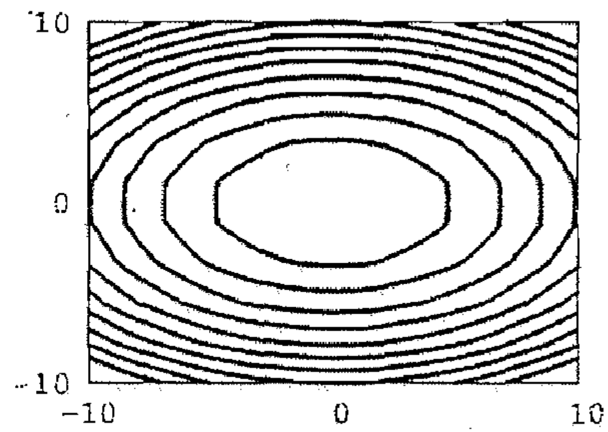
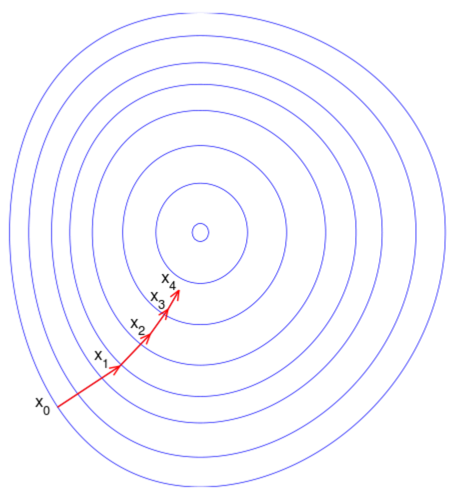
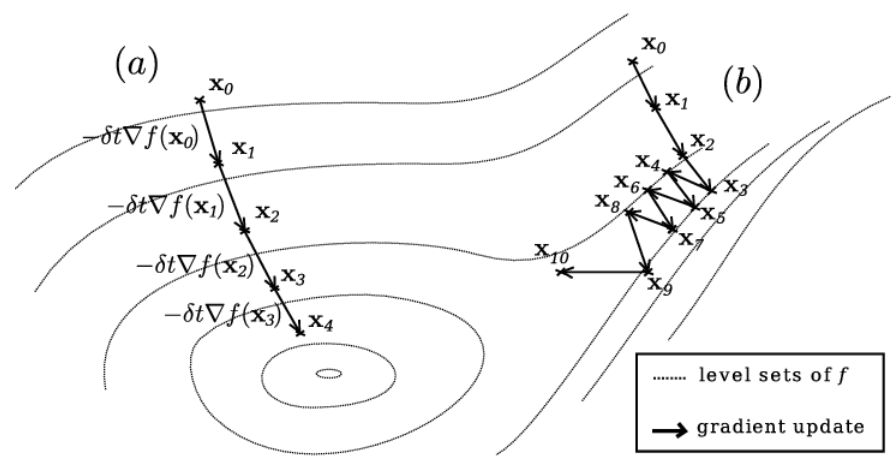


Fig. 1.6: Level curves of $f(x, y) = x^2 + 2y^2 = c$ for $c = 25, 50, \ldots, 250$.





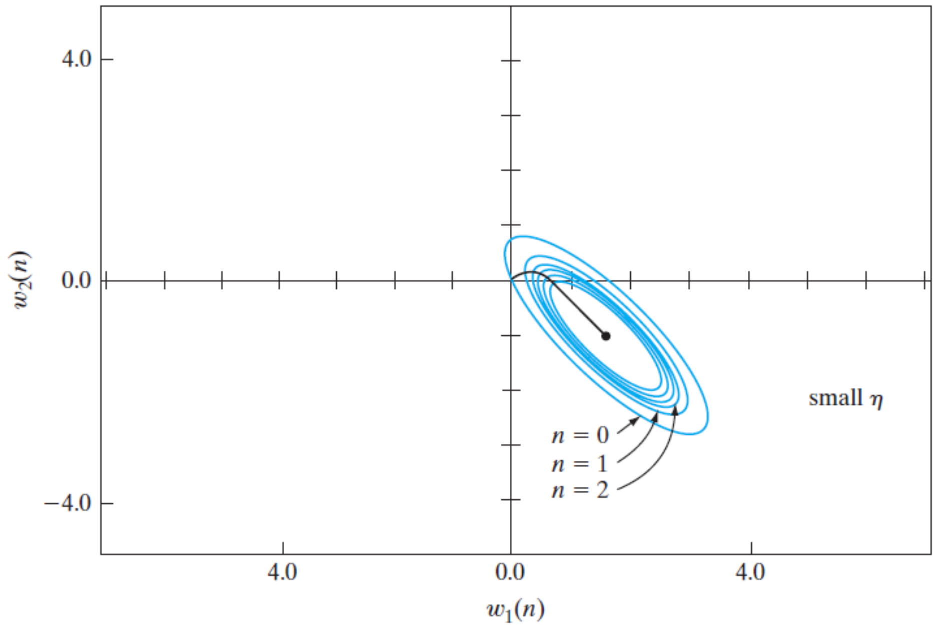
Newton-Raphson Algorithm

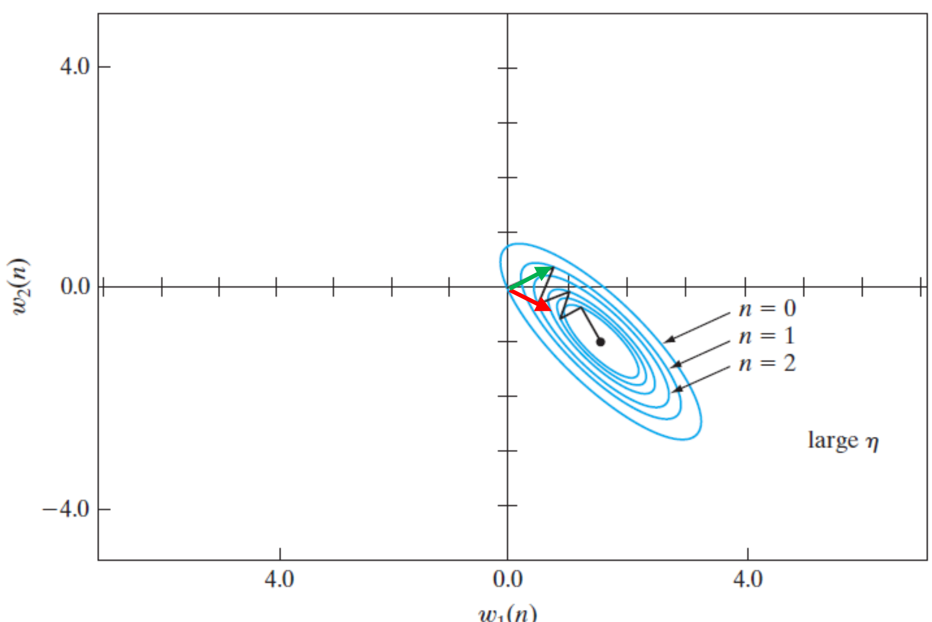
- Quadratic approximation:

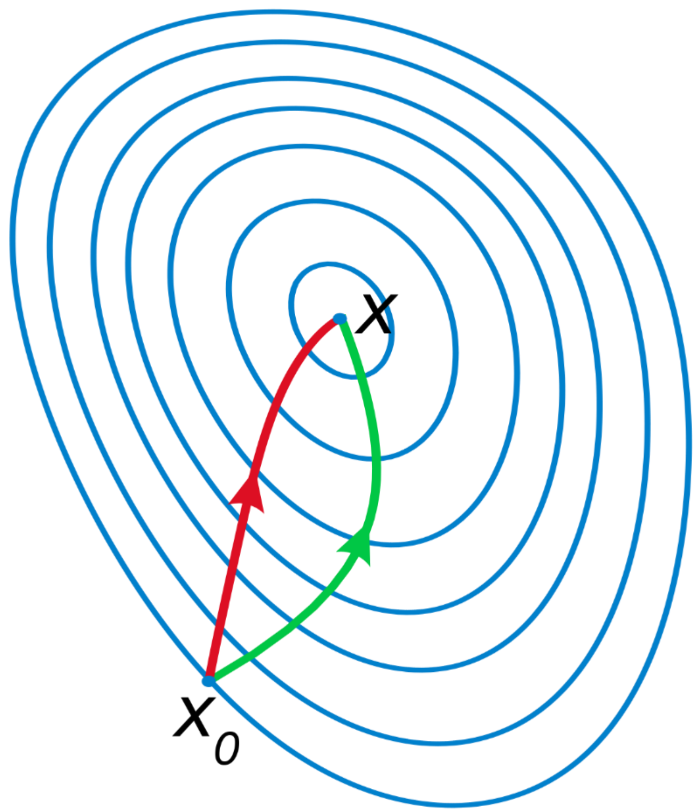
$$\Phi\left(\mathbf{u}^k + \Delta\mathbf{u}\right) = \Phi\left(\mathbf{u}^k\right) + \left[\nabla\Phi\left(\mathbf{u}^k\right)\right]^T \Delta\mathbf{u} + \frac{1}{2}\Delta\mathbf{u}^T\left[\nabla^2\Phi\left(\mathbf{u}^k\right)\right]\Delta\mathbf{u} + \mathrm{h.o.t}$$

- Initial guess $\mathbf{u}^0$
- Iterate $\mathbf{u}^{k+1} = \mathbf{u}^k - \left[\nabla^2\Phi\left(\mathbf{u}^k\right)\right]^{-1}\nabla\Phi\left(\mathbf{u}^k\right)$
- How to stop?

Steepest Descent vs. Newton-Raphson







Gauss-Newton Algorithm

- Approximate Hessian as

$$\left[\nabla^2 \Phi\left(\mathbf{u}^k\right)\right] \approx \left[\nabla \Phi\left(\mathbf{u}^k\right)\right]\left[\nabla \Phi\left(\mathbf{u}^k\right)\right]^T = \mathbf{H}$$

- Initial guess $\mathbf{u}^0$

- Iterate $\mathbf{u}^{k+1} = \mathbf{u}^k - \mathbf{H}^{-1}\nabla \Phi\left(\mathbf{u}^k\right)$

  or $\mathbf{u}^{k+1} = \mathbf{u}^k - [\mathbf{H} + \lambda\mathbf{I}]^{-1}\nabla \Phi\left(\mathbf{u}^k\right)$

- How to stop?

- How about conjugate gradient algorithm?

EOD