Numerical Optimization with Python

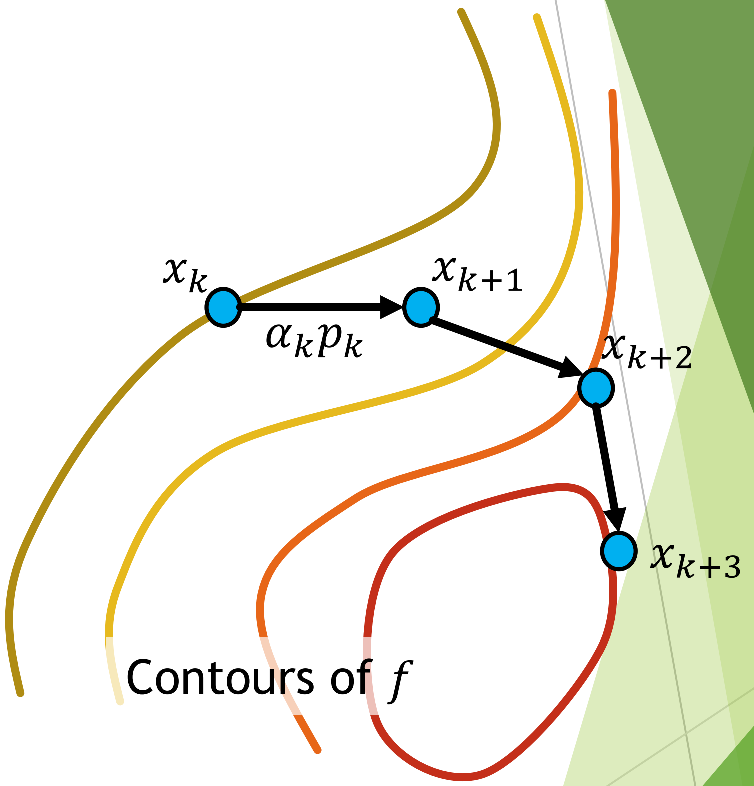
Lecture 3: Unconstrained Optimization (Part 2/2)

Lecture 03: Unconstrained Optimization (Part 2/2)

- Line search methods: gradient descent and Newton directions
- Choosing the step size: Wolfe conditions for sufficient decrease
- Convergence analysis
- An overview of quasi-Newton methods

Line Search Methods: Steepest Descent and Newton Directions

- A general framework for line search methods:
  - At each iteration - compute a search direction p_k
  - Decide how far to move along that direction
  - The iteration update rule is given by: $x_{k+1} = x_k + \alpha_k p_k$
  - The positive scalar α_k is called the *step length*



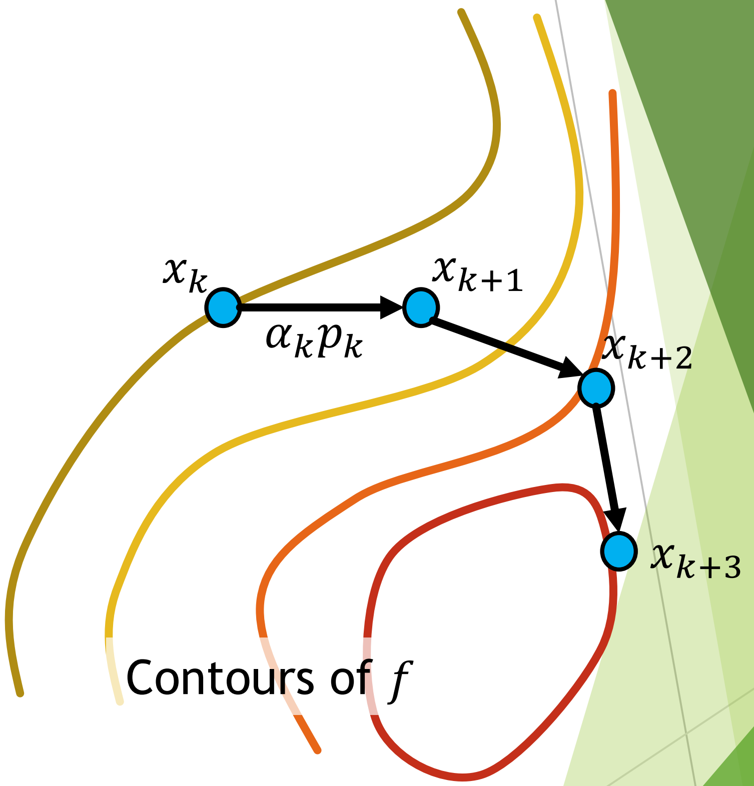
- Questions: is it literally a step length? How is our naïve gradient descent from HW01 and previous lecture a special case of the above?

Line Search Methods: Steepest Descent and Newton Directions

- We will focus on p_k of the following types:
  - The search direction will typically be required to be a descent direction, namely: $p_k^T \nabla f_k < 0$
  - The search direction often will have the form:

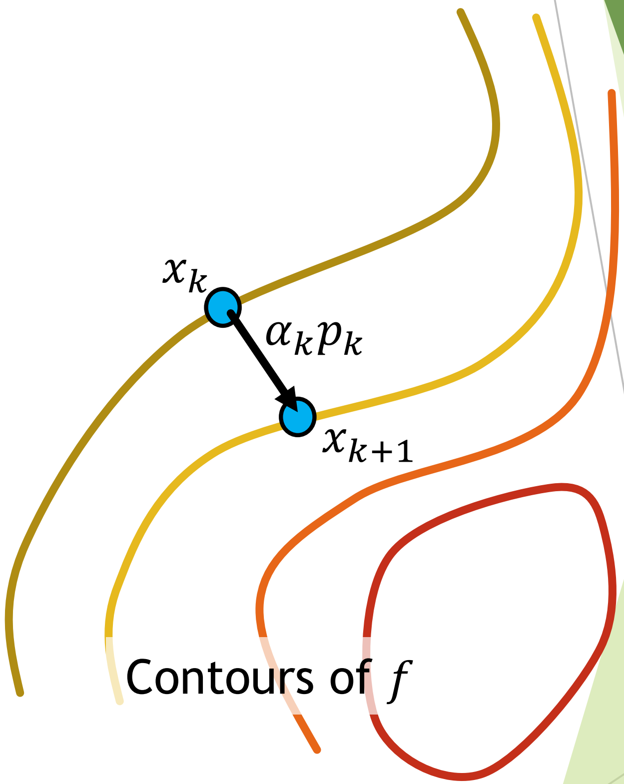
$$p_k = -B_k^{-1} \nabla f_k$$

where B_k is a symmetric and non-singular matrix (we will see several examples for how this form arises)



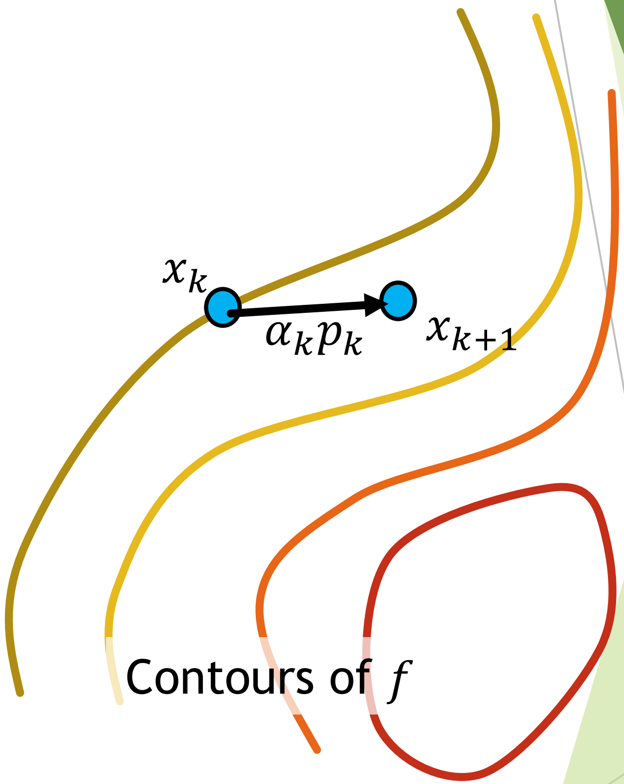
Line Search Methods: Steepest Descent and Newton Directions

- Gradient descent direction:
  - We have reviewed the fact from Multivariate Calculus, that $-\nabla f(x)$ is the direction of steepest descent
  - This is a local fact: at x, the directional derivative is minimal in the direction $-\nabla f(x)$
  - In the line search terminology ($p_k = -B_k^{-1} \nabla f_k$): $p_k = -\nabla f_k$ and $B_k = I$



Line Search Methods: Steepest Descent and Newton Directions

- Newton direction:
  - In the line search terminology ($p_k = -B_k^{-1} \nabla f_k$): $p_k = -\nabla^2 f_k^{-1} \nabla f_k$ and $B_k = \nabla^2 f_k$ (the Hessian)
  - Far from the minimizer – Newton direction might not be a descent direction!
  - Why is the Newton direction defined this way?

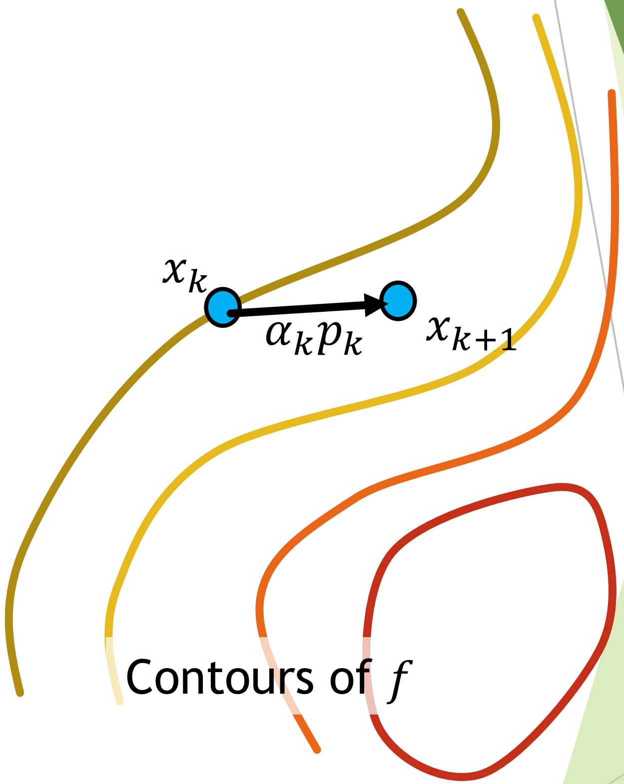


Line Search Methods: Steepest Descent and Newton Directions

- Newton direction derivation - motivation:
  - Consider the easy case where f is quadratic
  - To minimize $f(x) = \frac{1}{2}x^T Bx + a^T x + c$, differentiate:

$$\nabla f(x) = Bx + a$$

Requiring $\nabla f(x) = 0$ yields $x = -B^{-1}a$, and in the case of B positive definite (f convex) x is indeed a minimizer



Line Search Methods: Steepest Descent and Newton Directions

- Newton direction – obtained as the minimization of the quadratic model:
  - Now f is not quadratic but consider its best quadratic model - 2nd order Taylor approximation, at x_k:

$$m_k(x_k + p) = f(x_k) + \nabla f(x_k)^T p + \frac{1}{2} p^T \nabla^2 f(x_k) p$$

  - Here x_k is constant (the current direction)
  - p is the unknown and will be defined as the Newton step
  - Differentiating: $\nabla m_k(x_k + p) = \nabla f(x_k) + \nabla^2 f(x_k) p$
  - Requiring $\nabla m_k(x_k + p) = 0$ yields $p = -\nabla^2 f(x_k)^{-1} \nabla f(x_k)$

Line Search Methods: Steepest Descent and Newton Directions

- Newton direction may be undefined if $\nabla^2 f(x_k)$ is not invertible
- Note how from the derivation we have an associated natural step size of 1
- Note how Newton direction might not be a descent direction (it marches to a stationary point of the quadratic model. That's it!):

$$\nabla f(x_k)^T p = \nabla f(x_k)^T[-\nabla^2 f(x_k)^{-1}\nabla f(x_k)] = -\nabla f(x_k)^T \nabla^2 f(x_k)^{-1} \nabla f(x_k)$$

We have shown: the *directional derivative* is the negative of the Hessian's (inverse) quadratic form. It is not guaranteed to be negative. If the Hessian is positive definite we have guarantee.

Line Search Methods: Steepest Descent and Newton Directions

- A reminder from linear algebra: what are the eigenvalues of the inverse of a positive definite (or any symmetric, non-singular matrix)?
- A is symmetric and PD, then we can write: $A = V^T D V$ (V orthogonal and $D = \mathrm{diag}[\delta_1, \dots, \delta_n]$)
- Consider $V^T D^{-1} V$ where $D^{-1} := \mathrm{diag}\left[\frac{1}{\delta_1}, \dots, \frac{1}{\delta_n}\right]$. Then:

$$V^T D V V^T D^{-1} V = V^T D D^{-1} V = V^T V = \mathrm{Id}$$

- We have shown that $A^{-1} = V^T D^{-1} V$, and hence the eigenvalues are $\frac{1}{\delta_1}, \dots, \frac{1}{\delta_n}$

Line Search Methods: Steepest Descent and Newton Directions

- A technique for overcoming situations where the Newton direction is not a descent direction: Hessian modification
- In practice, each iteration involves solving the linear system:

$$\nabla^2 f(x_k) p_k^N = -\nabla f(x_k)$$

  where p_k^N is the unknown (Newton direction)

- The idea: replace the coefficient matrix $\nabla^2 f(x_k)$ with a positive definite approximation

Line Search Methods: Steepest Descent and Newton Directions

- Possible modifications:
  - A multiple of the identity: find a scalar $\tau > 0$ such that $\nabla^2 f(x_k) + \tau I$ is sufficiently positive definite
  - Modified *Cholesky Factorization*: attempt to decompose $\nabla^2 f(x_k) = LDL^T$ and upon failure, update the computed elements of D such that they are positive

(If you are not familiar with Cholesky Factorization: every symmetric positive-definite matrix A can be written in the form $A = LDL^T$, where L is lower triangular with unit diagonal and D is diagonal matrix with positive elements)

Line Search Methods: Steepest Descent and Newton Directions

- NOTE: the form LDL^T is convenient for the above described modification procedure. In other contexts you usually encounter Cholesky decomposition in the form $A = LL^T$, but these are equivalent since we can use $LD^{\frac{1}{2}}$ (well defined since all diagonal elements are positive)

Choosing the Step Size: Wolfe Conditions for Sufficient Decrease

- The ideal choice of step length α_k would be the global minimizer of the univariate problem: $\phi(\alpha) = f(x_k + \alpha p_k), \alpha > 0$
- In general, this procedure (referred to as *exact line search*) is too expensive

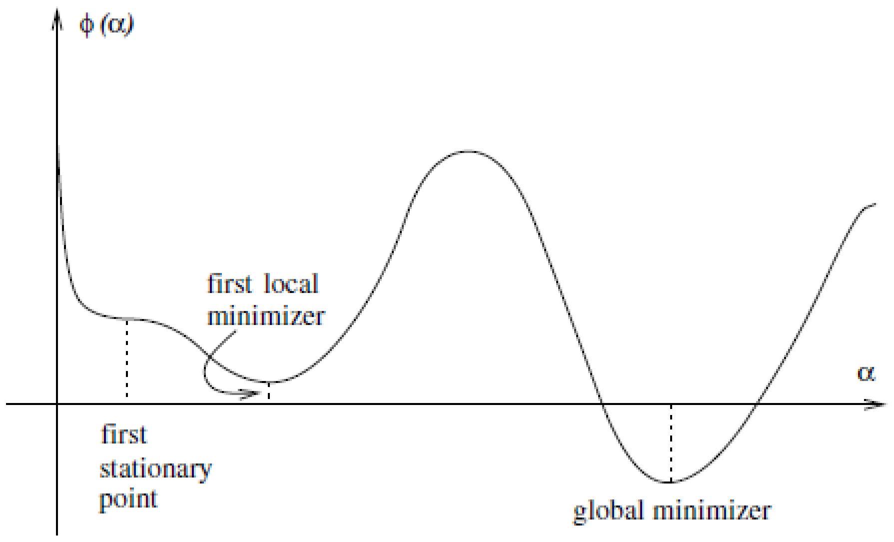


Fig: Naucedel & Wright Ch03

Choosing the Step Size: Wolfe Conditions for Sufficient Decrease

- Instead: inexact line search to identify step length with adequate reduction of f at low cost
- A naïve requirement might be decrease in objective values:

$$f(x_k + \alpha_k p_k) < f(x_k)$$

- Easy: construct an example of a sequence that decreases but is bounded away from the minimizer
- So – a more strict requirement is needed

Choosing the Step Size: Wolfe Conditions for Sufficient Decrease

- The Wolfe conditions: sufficient decrease in function values, as measured by the inequality:

$f(x_k + \alpha p_k) \leq f(x_k) + c_1 \alpha \nabla f(x_k)^T p_k$, for some constant $c_1 \in (0, 1)$

Choosing the Step Size: Wolfe Conditions for Sufficient Decrease

- The Wolfe conditions: sufficient decrease in function values, as measured by the inequality:

$$f(x_k + \alpha p_k) \leq f(x_k) + c_1 \alpha \nabla f(x_k)^T p_k, \text{ for some constant } c_1 \in (0, 1)$$

Function values at the next iterate, the selected location along the line p_k

Choosing the Step Size: Wolfe Conditions for Sufficient Decrease

- The Wolfe conditions: sufficient decrease in function values, as measured by the inequality:

$$f(x_k + \alpha p_k) \le f(x_k) + c_1 \alpha \nabla f(x_k)^T p_k, \text{ for some constant } c_1 \in (0,1)$$

Function values at the next iterate, the selected location along the line p_k

A linear function of the step length α, coinciding with f at $\alpha = 0$ (namely at x_k) with negative but less negative than f along p_k at x_k

Choosing the Step Size: Wolfe Conditions for Sufficient Decrease

- The Wolfe conditions: sufficient decrease in function values, as measured by the inequality: $f(x_k + \alpha p_k) \leq f(x_k) + c_1 \alpha \nabla f(x_k)^T p_k$, for constant $c_1 \in (0, 1)$

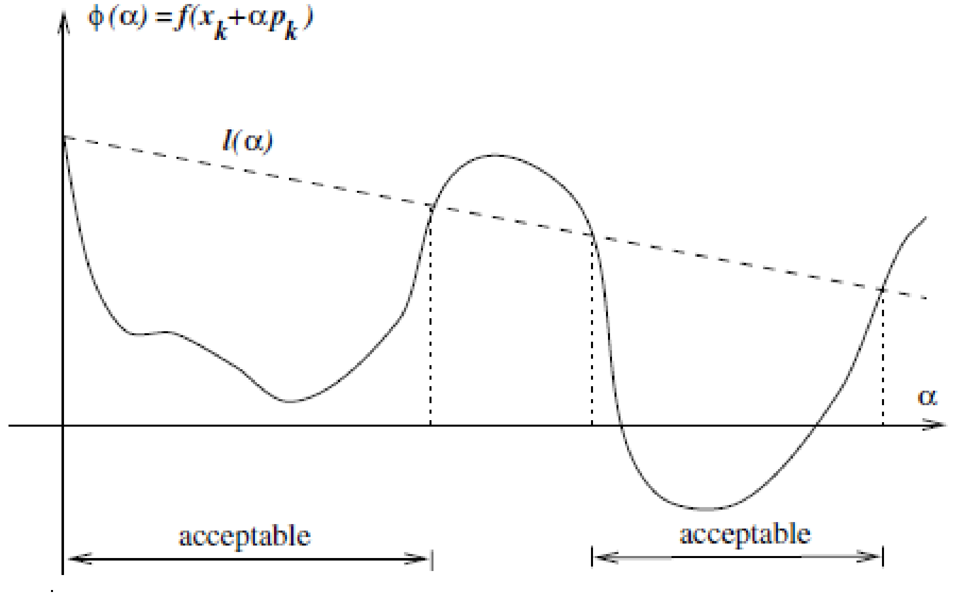


Fig: Naucedel & Wright Ch03

Choosing the Step Size: Wolfe Conditions for Sufficient Decrease

- Problem: the condition is easily satisfied by all sufficiently small values of α, and the algorithm might not make reasonable progress if taking very small steps

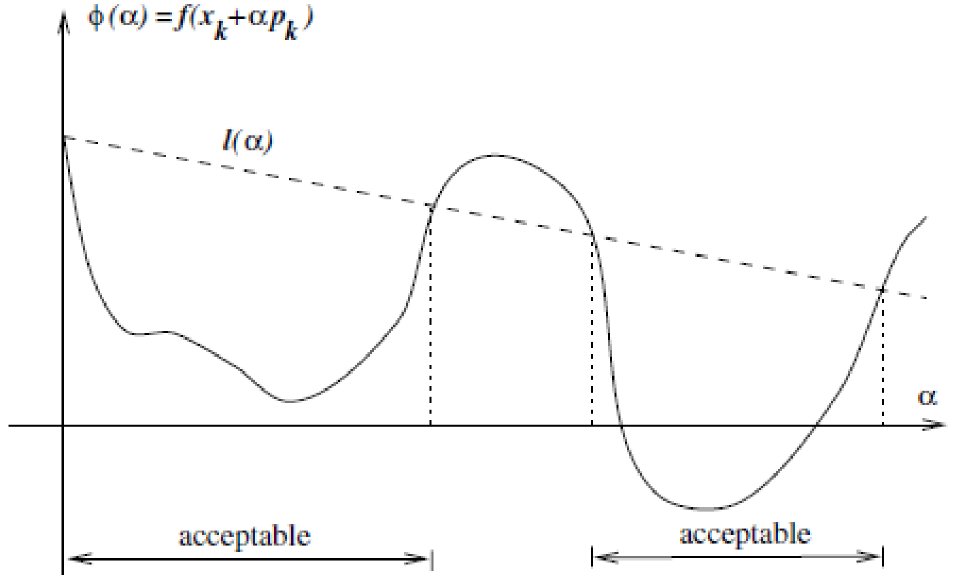


Fig: Naucedel & Wright Ch03

Choosing the Step Size: Wolfe Conditions for Sufficient Decrease

- Thus we introduce a second requirement – the curvature condition:

$$\nabla f(x_k + \alpha_k p_k)^T p_k \geq c_2 \nabla f(x_k)^T p_k \text{ for some constant } c_2 \in (0,1)$$

Choosing the Step Size: Wolfe Conditions for Sufficient Decrease

- Thus we introduce a second requirement – the curvature condition:

$\nabla f(x_k + \alpha_k p_k)^T p_k \geq c_2 \nabla f(x_k)^T p_k$ for some constant $c_2 \in (0,1)$

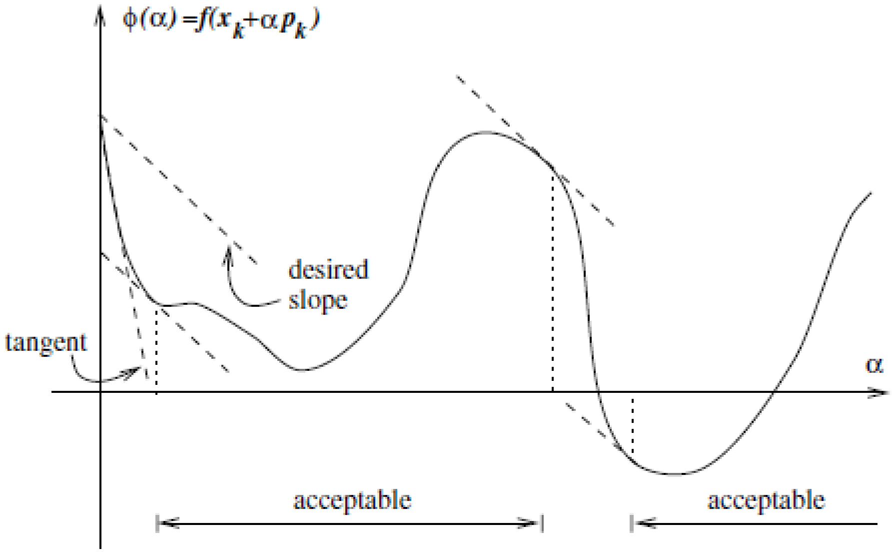
The slope $\phi'(\alpha_k)$

c_2 times the slope $\phi'(0)$

Choosing the Step Size: Wolfe Conditions for Sufficient Decrease

- Thus we introduce a second requirement – the curvature condition:

$$\nabla f(x_k + \alpha_k p_k)^T p_k \geq c_2 \nabla f(x_k)^T p_k \text{ for some constant } c_2 \in (0,1)$$



Choosing the Step Size: Wolfe Conditions for Sufficient Decrease

- The underlying idea: if the slope $\phi'(\alpha)$ is “strongly negative” we may attain significant decrease in f by moving further along the search direction.
- If the slope $\phi'(\alpha)$ is only slightly negative, on the other hand, it makes sense to terminate the search
- Summarizing, we require: $\begin{cases} f(x_k + \alpha p_k) \leq f(x_k) + c_1 \alpha \nabla f(x_k)^T p_k \\ \nabla f(x_k + \alpha_k p_k)^T p_k \geq c_2 \nabla f(x_k)^T p_k \end{cases}$

With $0 < c_1 < c_2 < 1$

Choosing the Step Size: Wolfe Conditions for Sufficient Decrease

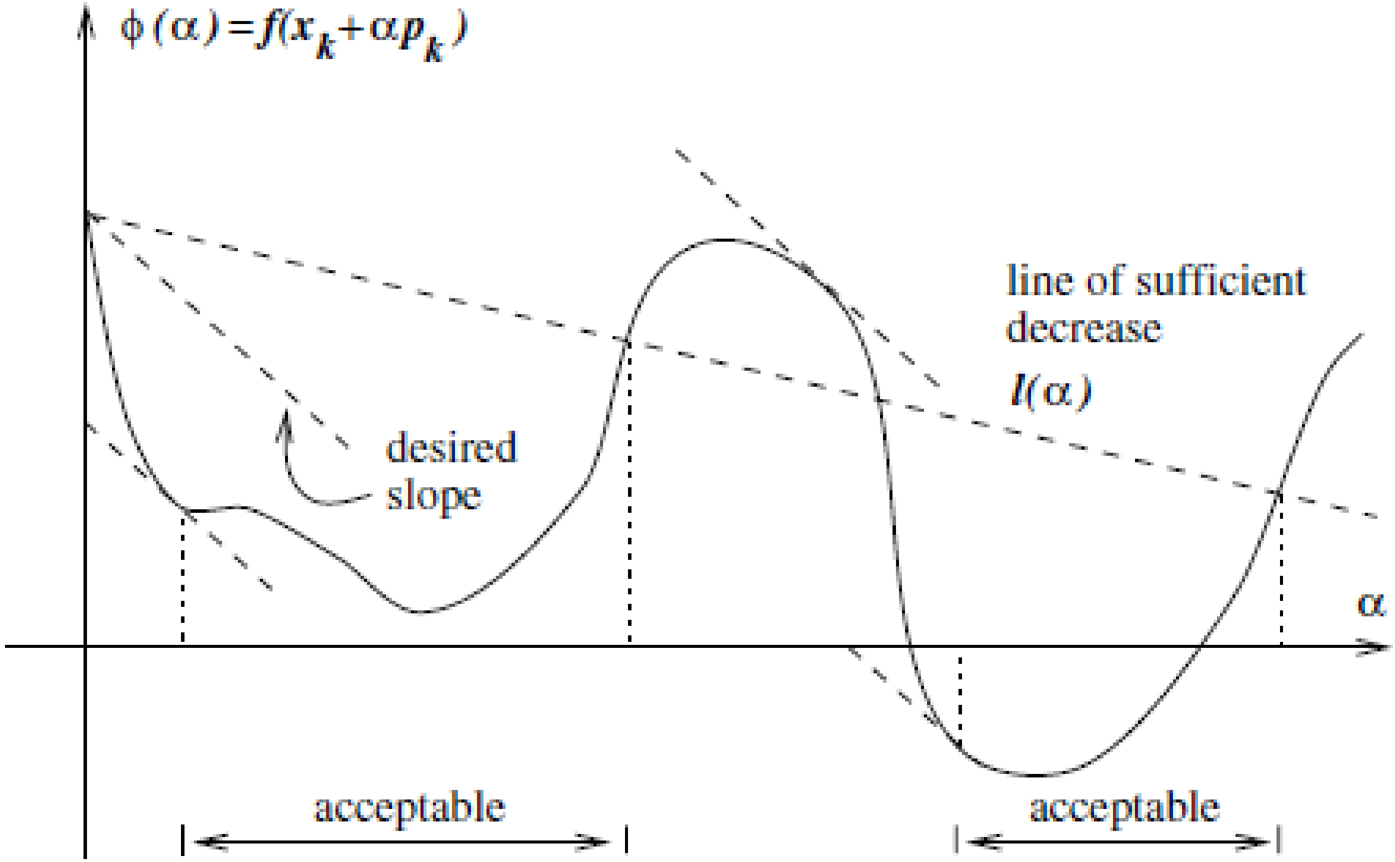


Fig: Naucedel & Wright Ch03

Choosing the Step Size: Wolfe Conditions for Sufficient Decrease

- Question: is it guaranteed that such intervals can be found?

- **Lemma:** if p_k is a descent direction and f is bounded below along the ray $x_k + \alpha p_k$, $\alpha > 0$, then there exist intervals satisfying the Wolfe conditions.

- Proof ingredients: continuity and mean value theorems. See Naucedel & Wright Lemma 3.1, Ch03.

- Practical technique for finding α: backtracking $\alpha \leftarrow \rho\alpha$ from initial $\bar{\alpha}$ and $\rho \in (0,1)$

Convergence Analysis

- The theoretical result states that under appropriate assumptions (typically not checked in concrete situations), steepest descent and Newton's methods converge to stationary points: $\lim_{k\to\infty} \|\nabla f(x_k)\|$
- The above relies on a technical result: *Zoutendijk's Theorem* (see Naucedel & Wright, Theorem 3.2).
- Geometrically, conditions are made to ensure that search directions are bounded away from orthogonality to the gradient, and that step lengths are chosen according to Wolfe conditions.

Convergence Analysis

- Rate of convergence is linear for steepest descent
- Rate of convergence is quadratic for Newton’s method, provided that the starting point is sufficiently close to the minimizer
- (the above properties are typical in the sense that for quadratic convergence we are required the cost of evaluating second derivatives, and hence the name first order/second order methods)

An Overview of Quasi-Newton Methods

- In order not to compute the Hessian but still enjoy super-linear convergence: $\nabla^2 f(x_k)$ is replaced with an approximation B_k, typically devised via the change in gradient from one location to the next
- Examples of two possible Hessian approximations: SR1 and BFGS, described next
- We would like to make use of the fact that $\nabla^2 f(x_k)(x_{k+1} - x_k)$ is an approximation for $\nabla f(x_{k+1}) - \nabla f(x_k)$ (why?)

An Overview of Quasi-Newton Methods

- To obtain our B_{k+1}, the Hessian approximation in the next step, we require it satisfies the following condition, called the *secant equation*:

$$B_{k+1}s_k = y_k$$

Where $s_k = x_{k+1} - x_k$ and $y_k = \nabla f(x_{k+1}) - \nabla f(x_k)$ (attempting to mimic the linear approximation via derivatives)

- Sometimes further conditions are imposed on B_{k+1} such as symmetry (as in the exact Hessian) and low rank of the difference $B_{k+1} - B_k$

An Overview of Quasi-Newton Methods

- *SR1 (Symmetric Rank One)* update formula:

$$B_{k+1} = B_k + \frac{(y_k - B_k s_k)(y_k - B_k s_k)^T}{(y_k - B_k s_k)^T s_k}$$

- *BFGS (Broyden, Fletcher, Goldfarb and Shanno)* update formula:

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{y_k^T s_k}$$

An Overview of Quasi-Newton Methods

- Properties:
  - The update has rank 1 in SR1 and rank 2 in BFGS
  - Both updates satisfy the Secant equation
  - Both maintain symmetry
  - If B_0 is positive definite, and if $s_k^T y_k > 0$, BFGS produces positive definite approximations
- The direction is then defined by $p_k = -B_k^{-1} \nabla f(x_k)$ (namely use B_k in place of the exact Hessian)

An Overview of Quasi-Newton Methods

Some further remarks and points for discussion:

- Frozen Hessians: use same Hessian for several iterations
- Exact Update every few iterations and low rank update in the rest
- Are we inverting matrices at each iteration to obtain $p_k = -B_k^{-1}\nabla f(x_k)$?
- Why are low rank updates interesting?