CS Space

Unconstrained Nonlinear Optimization
Lecture 2: Line Search Methods 1

Feodor Pisnitchenko

CMCC/UFABC

Momoe Sakamori

IMECC/UNICAMP

Bibliography

- Nocedal, J. and Wright, S. *Numerical Optimization* (2nd ed.). Springer, New York, 2006.
- Bertsekas, D. P. *Nonlinear Programming* (2nd ed.). Athena Scientific, Belmont MA, 1999.
- Luenberger, D. G. and Ye, Y. *Linear and Nonlinear Programming* (3rd ed.). Springer, New York, 2008.
- Dennis, J. E., Jr. and Schnabel, R. B. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Classics in Applied Mathematics 16. SIAM, Philadelphia, 1996.

Line Search Algorithms: Core Idea

- **Choose a descent direction** p_k at iteration k.
- **Update** the iterate via

$$x_{k+1} = x_k + \alpha_k p_k,$$

  where $\alpha_k > 0$ is the *step length*.
- **(Inexactly) minimise** the univariate function $\phi(\alpha) = f(x_k + \alpha p_k)$ to obtain α_k. Common techniques:
  - Backtracking line search (Armijo test)
  - Two–phase line search with ZOOM (Wolfe / strong Wolfe tests)

Line Search Methods: Search Directions

$$x_{k+1} = x_k + \alpha_k p_k$$

Steepest descent (gradient) method

$$p_k = -\nabla f_k$$

Descent direction

Any p_k such that

$$p_k^T \nabla f_k < 0,$$

e.g., where the angle θ between p_k and $-\nabla f_k$ is less than $\frac{\pi}{2}$

Newton's direction

$$p_k = -(\nabla^2 f_k)^{-1} \nabla f_k$$

Alternative Choices for Search Directions

Diagonally scaled steepest descent

$$p_k = -D_k \nabla f_k, \quad D_k \approx (\nabla^2 f_k)^{-1} \text{ diagonal}$$

Modified Newton's (constant Hessian)

$$p_k = -(\nabla^2 f_0)^{-1} \nabla f_k$$

Finite-difference Newton's

$$p_k = -H_k^{-1} \nabla f_k, \quad H_k \approx \nabla^2 f_k \text{ via finite differences}$$

Descent Property of a Direction

From Taylor's theorem with remainder, there exists some $t \in (0, \alpha)$ such that:

$$f(x_k + \alpha p) = f(x_k) + \alpha p^T \nabla f(x_k + tp).$$

Suppose that the direction p satisfies:

$$p^T \nabla f(x_k) < 0.$$

Then, by continuity of ∇f, for sufficiently small $\alpha > 0$, we also have:

$$p^T \nabla f(x_k + tp) < 0.$$

Therefore,

$$f(x_k + \alpha p) < f(x_k),$$

which means that p is a **descent direction** at x_k.

Steepest Descent Direction: Maximum Rate of Decrease

The steepest descent direction minimizes the directional derivative:

$$\min_p p^T \nabla f_k \quad \text{s.t.} \quad \|p\| = 1$$

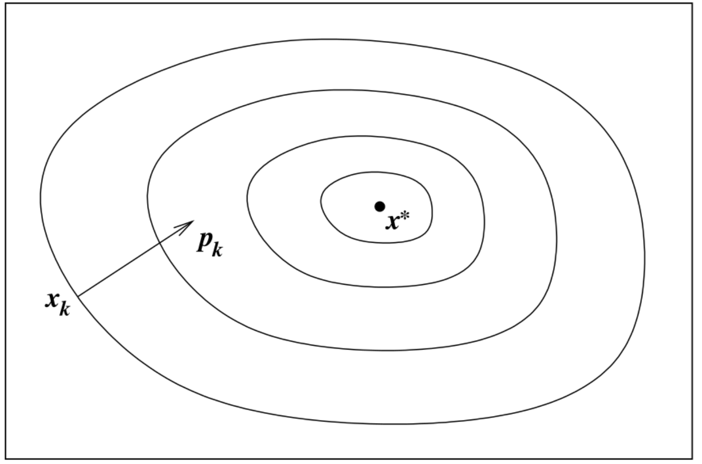
Since:

$$p^T \nabla f_k = \|p\| \|\nabla f_k\| \cos\theta = \|\nabla f_k\| \cos\theta$$

The solution is:

$$\cos\theta = -1, \quad p = -\frac{\nabla f_k}{\|\nabla f_k\|}$$

Steepest descent direction



Newton's Direction

Applying second-order Taylor approximation to $f(x_k + p)$ yields:

$$f(x_k + p) \approx f_k + p^T \nabla f_k + \frac{1}{2} p^T \nabla^2 f_k p \stackrel{\text{def}}{=} m_k(p)$$

To minimize $m_k(p)$, assuming $\nabla^2 f_k \succ 0$, set the gradient to zero:

$$p_k^N = -(\nabla^2 f_k)^{-1} \nabla f_k$$

Newton's direction

Applying Taylor second-order series approximation to $f(x_k + p)$,

$$f(x_k + p) \approx f_k + p^T \nabla f_k + \frac{1}{2} p^T \nabla^2 f_k p \stackrel{\text{def}}{=} m_k(p).$$

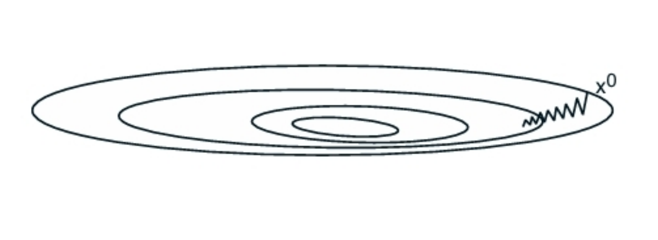
Assuming that $\nabla^2 f_k$ is positive definite, we obtain p that minimizes $m_k(p)$ by setting its derivative to zero,

$$p_k^N = -(\nabla^2 f_k)^{-1} \nabla f_k$$

Steepest Descent vs. Newton's Method

Steepest descent:

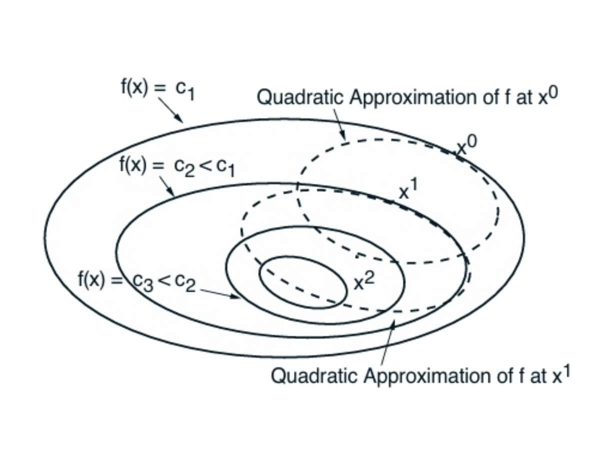
- Slow convergence
- Requires only first order information of f



Steepest Descent vs. Newton's Method

Newton's method:

- Fast convergence
- Requires second-order information



Quasi-Newton Search Directions

We replace the true Hessian $\nabla^2 f_k$ with a symmetric approximation B_k and define the model:

$$m_k(p) = f_k + p^T \nabla f_k + \frac{1}{2} p^T B_k p.$$

The search direction is:

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

To update B_k, we use the **secant equation**:

$$B_{k+1} s_k = y_k, \quad s_k = x_{k+1} - x_k, \quad y_k = \nabla f_{k+1} - \nabla f_k.$$

Most Quasi-Newton directions uses the updates of B_k that satisfy the secant condition and preserve symmetry.

Quasi-Newton search directions

SR1 - Symmetric-rank-one 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 formula

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

BFGS guarantees positive definiteness if $B_0 \succ 0$ and $s_k^T y_k > 0$.

Inverse Updates in Quasi-Newton Methods

It is often more efficient to update the inverse approximation $H_k = B_k^{-1}$ directly.

SR1 inverse update

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

BFGS inverse update

$$H_{k+1} = (I - \rho_k s_k y_k^T) H_k (I - \rho_k y_k s_k^T) + \rho_k s_k s_k^T, \quad \rho_k = \frac{1}{y_k^T s_k}$$

The search direction is computed as:

$$p_k = -H_k \nabla f_k$$

Step Length: General Idea

- Given a descent direction p_k, we search for a new iterate x_{k+1} with lower objective value:

$$x_{k+1} = x_k + \alpha_k p_k,$$

  where $\alpha_k > 0$ is the **step length**.

- The goal is to reduce $f(x)$ as much as possible along the direction p_k.

Search Direction Structure

Typically, the search direction p_k has the form:

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

where B_k is a symmetric and nonsingular matrix.

- Steepest descent method: $B_k = I$
- Newton's method: $B_k = \nabla^2 f(x_k)$
- Quasi-Newton method: $B_k \approx \nabla^2 f(x_k)$

Ideal Step Length

Let

$$\phi(\alpha) = f(x_k + \alpha p_k),$$

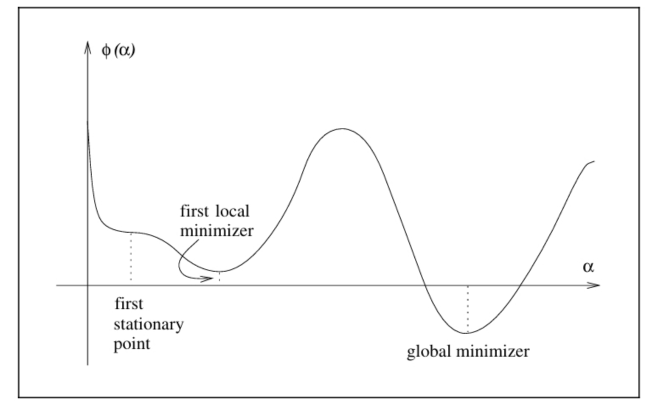
which is a univariate function of the step length α. The ideal step length is the global minimizer of $\phi(\alpha)$:

$$\alpha_k^* = \arg\min_{\alpha>0} \phi(\alpha).$$

However, exact line search is often computationally expensive.

In practice: Use inexact line search with sufficient decrease and curvature conditions.

Step Length



$\phi(\alpha)$
first local
minimizer
α
first
stationary
point
global minimizer

Motivation for Inexact Line Search

- Computing the exact minimizer of $\phi(\alpha)$ is often impractical.
- Instead, we accept α_k if it leads to:
  - Sufficient decrease in f
  - Adequate curvature change

These conditions form the basis of inexact line search methods.

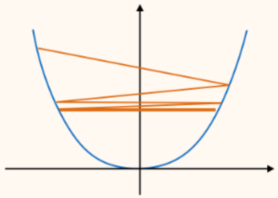
Basic Requirement for Step Length

The step length α_k should satisfy at least:

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

However, such a condition alone may accept steps that are too small or unproductive.

$$\begin{aligned} \min \quad & f(x) = x^2 \\ x_{k+1} &= x_k + \alpha_k p_k, \\ x_0 = -1, \quad & p_k = (-1)^k, \\ \alpha_k &= \left(2 - \frac{1}{4^{k+1}}\right)|x_k| \end{aligned}$$



Armijo Condition (Sufficient Decrease)

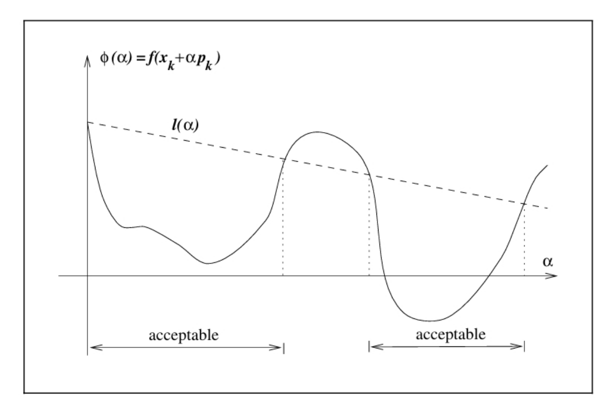
Armijo condition

$$f(x_k + \alpha p_k) \leq f(x_k) + c_1 \alpha \nabla f_k^T p_k,$$

for some $c_1 \in (0, 1)$.

- Ensures a sufficient decrease relative to the linear approximation.
- $\nabla f_k^T p_k < 0$ ensures descent.
- Typical value: $c_1 = 10^{-4}$

Geometric Interpretation of Armijo Condition



- Let $\ell(\alpha) = f(x_k) + c_1 \alpha \nabla f_k^T p_k$.
- Accept α if $\phi(\alpha) \leq \ell(\alpha)$.

Curvature Condition

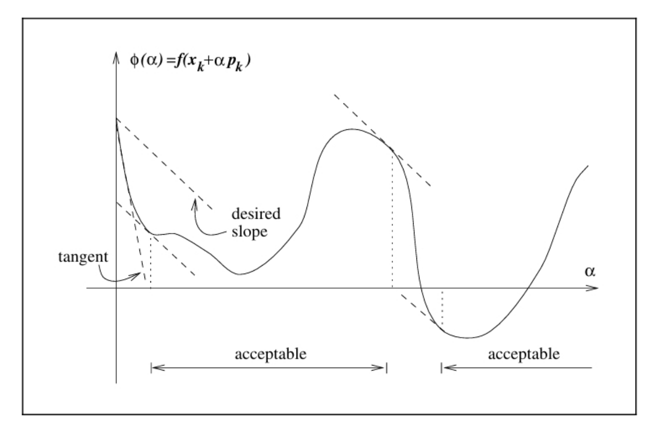
Curvature condition

$$\nabla f(x_k + \alpha_k p_k)^T p_k \geq c_2 \nabla f_k^T p_k,$$

for some $c_2 \in (c_1, 1)$.

- Ensures that the slope is not too steep at the new point.
- Equivalent to: $\phi'(\alpha_k) \geq c_2 \phi'(0)$
- Often paired with the Armijo condition.

Curvature Condition - Illustration



Wolfe and Strong Wolfe Conditions

Wolfe conditions

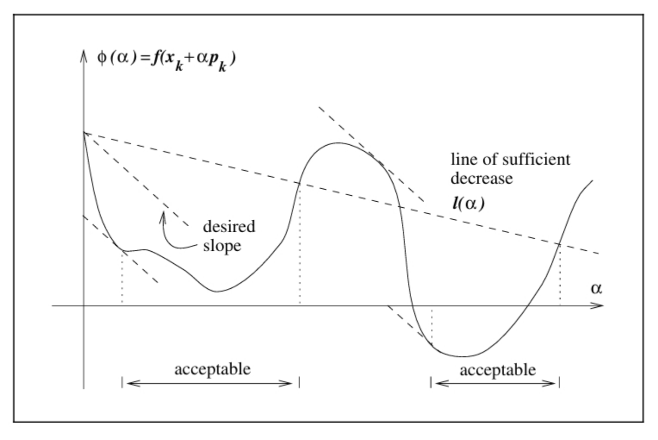
$$f(x_k + \alpha p_k) \le f(x_k) + c_1 \alpha \nabla f_k^T p_k$$
$$\nabla f(x_k + \alpha p_k)^T p_k \ge c_2 \nabla f_k^T p_k$$

Strong Wolfe conditions

$$f(x_k + \alpha p_k) \le f(x_k) + c_1 \alpha \nabla f_k^T p_k$$
$$\left|\nabla f(x_k + \alpha p_k)^T p_k\right| \le c_2 \left|\nabla f_k^T p_k\right|.$$

with $0 < c_1 < c_2 < 1.$

Wolfe Conditions – Illustration



Existence of Acceptable Step Length

Lemma

Let $f : \mathbb{R}^n \to \mathbb{R}$ be continuously differentiable, and p_k be a descent direction at x_k. If f is bounded below along the ray $\{x_k + \alpha p_k : \alpha > 0\}$, then for any $0 < c_1 < c_2 < 1$, there exists an interval of step lengths satisfying:

- Wolfe conditions
- Strong Wolfe conditions

Goldstein Conditions

Goldstein conditions

$$f_k + (1-c)\alpha\nabla f_k^T p_k \leq f(x_k + \alpha p_k) \leq f_k + c\alpha\nabla f_k^T p_k,$$

for $c \in (0, 0.5)$.

- Ensures both lower and upper bounds on decrease.
- May exclude the minimizer of $\phi(\alpha)$.
- Often used in Newton-type methods.

Summary of Step Length Conditions

- **Armijo (sufficient decrease)**:

$$f(x_k + \alpha p_k) \leq f(x_k) + c_1 \alpha \nabla f_k^T p_k$$

- **Curvature condition**:

$$\nabla f(x_k + \alpha p_k)^T p_k \geq c_2 \nabla f_k^T p_k$$

- **Strong Wolfe**:

$$\left| \nabla f(x_k + \alpha p_k)^T p_k \right| \leq c_2 \left| \nabla f_k^T p_k \right|$$

- **Goldstein**:

$$(1 - c)\alpha \nabla f_k^T p_k \leq f(x_k + \alpha p_k) - f_k \leq c \alpha \nabla f_k^T p_k$$

Backtracking Line Search

ALGORITHM 1: Backtracking Line Search

Choose $\bar{\alpha} > 0, \rho \in (0,1), c \in (0,1)$;

Set $\alpha \leftarrow \bar{\alpha}$;

repeat until $f(x_k + \alpha p_k) \le f(x_k) + c\alpha \nabla f_k^T p_k$

$\qquad \alpha \leftarrow \rho\alpha$;

end (repeat)

Terminate with $\alpha_k = \alpha$.

Backtracking Line Search: Parameters choices

Typical Initial Step Size $\bar{\alpha}$:

- **Newton / Quasi-Newton methods:** $\bar{\alpha} = 1$ (often leads to fast convergence near the solution)
- **Steepest descent:** smaller values (e.g., $\bar{\alpha} = 0.1$) help avoid overshooting when far from the minimum

Common Parameter Choices:

- $\rho \in [0.1, 0.8]$: step size reduction factor
- $c \in \{10^{-4},\ 10^{-2}\}$: Armijo parameter (as suggested by Nocedal and Wright)

Introduction to Hessian Modification

- Away from the solution, the Hessian $\nabla^2 f(x_k)$ may not be positive definite. Consequently, the Newton direction p_k^N defined by
$$\nabla^2 f(x_k) p_k^N = -\nabla f(x_k)$$
may not be a descent direction.

- To address this issue, we modify the Hessian by solving:
$$(\nabla^2 f(x_k) + E_k) p_k = -\nabla f(x_k),$$
where E_k is a positive diagonal matrix or a full matrix chosen to ensure the modified Hessian is positive definite.

Algorithm: Modified Newton's Method

ALGORITHM 2: Line Search Newton with Modification

Initialize x_0;
for $k = 0, 1, 2, \ldots$
Factorize $B_k = \nabla^2 f(x_k) + E_k$, where $E_k = 0$ if $\nabla^2 f(x_k)$ is sufficiently positive definite; otherwise, choose E_k to ensure B_k is sufficiently positive definite;
Solve $B_k p_k = -\nabla f(x_k)$;
Set $x_{k+1} \leftarrow x_k + \alpha_k p_k$, where α_k satisfies the Wolfe, Goldstein, or Armijo backtracking conditions;
end for

Convergence Theorem for Modified Newton's Method

Bounded Modified Factorization Property

The sequence of matrices $\{B_k\}$ has a bounded condition number whenever the sequence of Hessians $\{\nabla^2 f(x_k)\}$ is bounded:

$$\mathcal{K}(B_k) = \|B_k\|\|B_k^{-1}\| \leq C, \quad C > 0, \quad \forall k \geq 0$$

Theorem

Let f be twice continuously differentiable on an open set D, and assume the starting point x_0 of Algorithm 2 is such that the level set $\mathcal{L} = \{x \in D : f(x) \leq f(x_0)\}$ is compact. If the bounded modified factorization property holds, then:

$$\lim_{k \to \infty} \nabla f(x_k) = 0$$

Techniques for Hessian Modification

Hessian modification techniques:

- Eigenvalue modification
- Adding a multiple of the identity
- Modified Cholesky factorization
- Modified symmetric indefinite factorization

Overview of Step-Length Selection Algorithms

- Step-length selection is crucial for line search methods.
- Analytical solution exists for convex quadratics:

$$\alpha_k = -\frac{\nabla f_k^T p_k}{p_k^T Q p_k},$$

  while general nonlinear functions require iterative methods.
- Incorporating derivative information enhances efficiency and aids in satisfying Wolfe or Goldstein conditions.
- Near the solution, initial α often satisfies these conditions.

Step-Length Selection Algorithms

- Line search procedures start with an initial estimate α_0 and generate a sequence $\{\alpha_i\}$ that either terminates with α_i satisfying the conditions (e.g., Wolfe conditions) or determines no such step length exists.
- Typical procedures consist of two phases:
  1. A bracketing phase that finds an interval $[\bar{a}, \bar{b}]$ containing acceptable step lengths.
  2. A selection phase that zooms in to locate the final step length.

Interpolation for Step-Length Selection

Our goal is to find a minimum of the one-dimensional function:

$$\phi(\alpha) = f(x_k + \alpha p_k)$$

The sufficient decrease condition is:

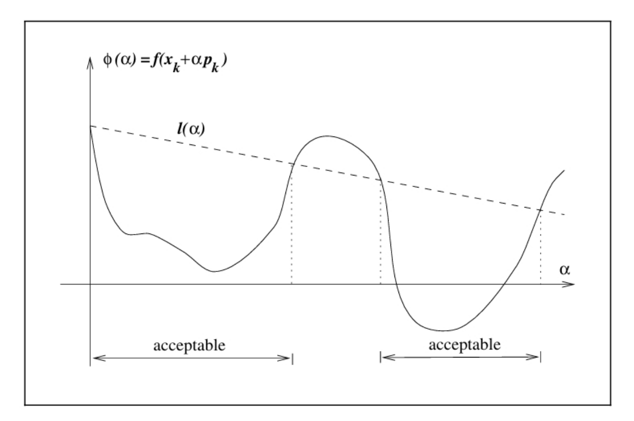
$$\phi(\alpha_k) \le \phi(0) + c_1 \alpha_k \phi'(0)$$

This condition ensures a decrease in f proportional to α_k and $\phi'(0)$. Typically, c_1 is small (e.g., 10^{-4}).

Interpolation for Step-Length Selection

Given an initial α_0:

- If $\phi(\alpha_0) \le \phi(0) + c_1\alpha_0\phi'(0)$, then α_0 is acceptable, and the search terminates.
- Otherwise, acceptable step lengths lie in $[0, \alpha_0]$.



Interpolation Techniques for Step-Length Selectic

We construct a quadratic approximation $\phi_q(\alpha)$ using $\phi(0)$, $\phi'(0)$, and $\phi(\alpha_0)$:

$$\phi_q(\alpha) = \left(\frac{\phi(\alpha_0) - \phi(0) - \alpha_0\phi'(0)}{\alpha_0^2}\right)\alpha^2 + \phi'(0)\alpha + \phi(0)$$

The minimizer α_1 of ϕ_q is:

$$\alpha_1 = -\frac{\phi'(0)\alpha_0^2}{2[\phi(\alpha_0) - \phi(0) - \phi'(0)\alpha_0]}$$

Interpolation Techniques for Step-Length Selectio

If α_1 does not satisfy the condition, use cubic interpolation with $\phi(0)$, $\phi'(0)$, $\phi(\alpha_0)$, and $\phi(\alpha_1)$:

$$\phi_c(\alpha) = a\alpha^3 + b\alpha^2 + \phi'(0)\alpha + \phi(0)$$

where:

$$\begin{bmatrix} a \\ b \end{bmatrix} = \frac{1}{\alpha_0^2\alpha_1^2(\alpha_1 - \alpha_0)} \begin{bmatrix} \alpha_0^2 & -\alpha_1^2 \\ -\alpha_0^3 & \alpha_1^3 \end{bmatrix} \begin{bmatrix} \phi(\alpha_1) - \phi(0) - \phi'(0)\alpha_1 \\ \phi(\alpha_0) - \phi(0) - \phi'(0)\alpha_0 \end{bmatrix}$$

Interpolation Techniques for Step-Length Selectio

The minimizer $\alpha_2 \in [0, \alpha_1]$ of ϕ_c is:

$$\alpha_2 = \frac{-b + \sqrt{b^2 - 3a\phi'(0)}}{3a}$$

If α_2 satisfies the condition, terminate. Otherwise, repeat with $\phi(0)$, $\phi'(0)$, $\phi(\alpha_1)$, and $\phi(\alpha_2)$.
Safeguard: If α_i is too close to or much smaller than α_{i-1}, set $\alpha_i = \alpha_{i-1}/2$.

Strategies for Initial Step Length

- For Newton and quasi-Newton methods, set $\alpha_0 = 1$.
- For steepest descent, one strategy assumes similar first-order change:

$$\alpha_0 = \alpha_{k-1} \frac{\nabla f_{k-1}^T p_{k-1}}{\nabla f_k^T p_k}$$

- Another strategy uses quadratic interpolation with $f(x_{k-1})$, $f(x_k)$, and $\nabla f_{k-1}^T p_{k-1}$:

$$\alpha_0 = \frac{2(f_k - f_{k-1})}{\phi'(0)}$$

Line Search Algorithm for Wolfe Conditions

- The Wolfe (or strong Wolfe) conditions are widely used for termination.
- The algorithm has two stages:
  1. Start with a trial α_1, increasing it until finding an acceptable step or bracketing interval.
  2. If bracketing, use the **zoom** function to decrease the interval until an acceptable step is found.

Line Search Algorithm for Wolfe Conditions

Strong Wolfe Conditions

$$\begin{aligned} \phi(\alpha) &\leq \phi(0) + c_1 \alpha \phi'(0) \\ |\phi'(\alpha)| &\leq c_2 |\phi'(0)| \end{aligned}$$

with $0 < c_1 < c_2 < 1$.

Line Search Algorithm for Wolfe Conditions

ALGORITHM 3: Line Search Algorithm

Set $\alpha_0 \leftarrow 0$, choose $\alpha_{\max} > 0$ and $\alpha_1 \in (0, \alpha_{\max})$;
Set $i \leftarrow 1$;
repeat
    Evaluate $\phi(\alpha_i)$;
    if $\phi(\alpha_i) > \phi(0) + c_1\alpha_i\phi'(0)$ or $[\phi(\alpha_i) \geq \phi(\alpha_{i-1})$ and $i > 1]$
        Set $\alpha_* \leftarrow$ **zoom**(α_{i-1}, α_i) and **stop**;
    Evaluate $\phi'(\alpha_i)$;
    if $|\phi'(\alpha_i)| \leq -c_2\phi'(0)$
        Set $\alpha_* \leftarrow \alpha_i$ and **stop**;
    if $\phi'(\alpha_i) \geq 0$
        Set $\alpha_* \leftarrow$ **zoom**(α_i, α_{i-1}) and **stop**;
    Choose $\alpha_{i+1} \in (\alpha_i, \alpha_{\max})$;
    Set $i \leftarrow i + 1$;
end repeat

Zoom Procedure for Line Search

- The sequence $\{\alpha_i\}$ is monotonically increasing.
- The **zoom** procedure is invoked when:
  (i) α_i violates the sufficient decrease condition;
  (ii) $\phi(\alpha_i) \geq \phi(\alpha_{i-1})$;
  (iii) $\phi'(\alpha_i) \geq 0$.
- Extrapolation (e.g., interpolation or $\alpha_{i+1} = M\alpha_i$ with $M > 1$) finds the next trial α_{i+1}.

Zoom Procedure for Line Search

The **zoom** function $(\alpha_{lo}, \alpha_{hi})$ satisfies:

(a) The interval $[\alpha_{lo}, \alpha_{hi}]$ contains step lengths satisfying the strong Wolfe conditions.

(b) α_{lo} has the smallest function value among those satisfying the sufficient decrease condition.

(c) α_{hi} is chosen such that $\phi'(\alpha_{lo})(\alpha_{hi} - \alpha_{lo}) < 0$.

Each iteration generates $\alpha_j \in (\alpha_{lo}, \alpha_{hi})$, updating endpoints to maintain these properties.

Zoom Procedure for Line Search

Algorithm 4: Zoom

repeat

Interpolate (quadratic, cubic, or bisection) to find a trial step length α_j between α_{lo} and α_{hi};

Evaluate $\phi(\alpha_j)$;

if $\phi(\alpha_j) > \phi(0) + c_1\alpha_j\phi'(0)$ or $\phi(\alpha_j) \geq \phi(\alpha_{lo})$

Set $\alpha_{hi} \leftarrow \alpha_j$;

else

Evaluate $\phi'(\alpha_j)$;

if $|\phi'(\alpha_j)| \leq -c_2\phi'(0)$

Set $\alpha_* \leftarrow \alpha_j$ and **stop**;

if $\phi'(\alpha_j)(\alpha_{hi} - \alpha_{lo}) \geq 0$

Set $\alpha_{hi} \leftarrow \alpha_{lo}$;

Set $\alpha_{lo} \leftarrow \alpha_j$;

end repeat

Zoom Procedure for Line Search

- If α_j satisfies the strong Wolfe conditions, set $\alpha_* = \alpha_j$ and terminate.
- Otherwise, if α_j satisfies the sufficient decrease condition and $\phi(\alpha_j) < \phi(\alpha_{lo})$, set $\alpha_{lo} \leftarrow \alpha_j$. If this violates condition (c), set $\alpha_{hi} \leftarrow \alpha_{lo}$.