Unconstrained minimization
Lectures for PHD course on
Numerical optimization

Enrico Bertolazzi

DIMS – Università di Trento

November 21 – December 14, 2011
1. General iterative scheme

2. Backtracking Armijo line-search
   - Global convergence of backtracking Armijo line-search
   - Global convergence of steepest descent

3. Wolfe–Zoutendijk global convergence
   - The Wolfe conditions
   - The Armijo-Goldstein conditions

4. Algorithms for line-search
   - Armijo Parabolic-Cubic search
   - Wolfe linesearch
Given \( f : \mathbb{R}^n \mapsto \mathbb{R} \):

\[
\minimize_{x \in \mathbb{R}^n} f(x)
\]

the following regularity about \( f(x) \) is assumed in the following:

**Assumption (Regularity assumption)**

We assume \( f \in C^1(\mathbb{R}^n) \) with Lipschitz continuous gradient, i.e.

there exists \( \gamma > 0 \) such that

\[
\| \nabla f(x)^T - \nabla f(y)^T \| \leq \gamma \| x - y \| , \quad \forall x, y \in \mathbb{R}^n
\]
The problem

**Definition (Global minimum)**

*Given* $f : \mathbb{R}^n \mapsto \mathbb{R}$ a point $x_\star \in \mathbb{R}^n$ is a **global minimum** if

$$f(x_\star) \leq f(x), \quad \forall x \in \mathbb{R}^n.$$ 

**Definition (Local minimum)**

*Given* $f : \mathbb{R}^n \mapsto \mathbb{R}$ a point $x_\star \in \mathbb{R}^n$ is a **local minimum** if

$$f(x_\star) \leq f(x), \quad \forall x \in B(x_\star; \delta).$$

Obviously a global minimum is a local minimum. Find a global minimum in general is not an easy task. The algorithms presented in the sequel will approximate local minima’s.
Definition (Strict global minimum)

Given \( f : \mathbb{R}^n \mapsto \mathbb{R} \) a point \( x^* \in \mathbb{R}^n \) is a **strict global minimum** if

\[
f(x^*) < f(x), \quad \forall x \in \mathbb{R}^n \setminus \{x^*\}.
\]

Definition (Strict local minimum)

Given \( f : \mathbb{R}^n \mapsto \mathbb{R} \) a point \( x^* \in \mathbb{R}^n \) is a **strict local minimum** if

\[
f(x^*) < f(x), \quad \forall x \in B(x^*; \delta) \setminus \{x^*\}.
\]

Obviously a strict global minimum is a strict local minimum.
Lemma (First order Necessary condition for local minimum)

Given $f : \mathbb{R}^n \mapsto \mathbb{R}$ satisfying the regularity assumption. If a point $x_\star \in \mathbb{R}^n$ is a local minimum then

$$\nabla f(x_\star)^T = 0.$$ 

Proof.

Consider a generic direction $d$, then for $\delta$ small enough we have

$$\lambda^{-1}(f(x_\star + \lambda d) - f(x_\star)) \geq 0, \quad 0 < \lambda < \delta$$

so that

$$\lim_{\lambda \to 0} \lambda^{-1}(f(x_\star + \lambda d) - f(x_\star)) = \nabla f(x_\star)d \geq 0,$$

because $d$ is a generic direction we have $\nabla f(x_\star)^T = 0.$
1. The first order necessary condition do not discriminate maximum, minimum, or saddle points.

2. To discriminate maximum and minimum we need more information, e.g. second order derivative of $f(x)$.

3. With second order derivative we can build necessary and sufficient condition for a minima.

4. In general using only first and second order derivative at the point $x_\star$ it is not possible to deduce a necessary and sufficient condition for a minima.
Lemma (Second order Necessary condition for local minimum)

Given \( f \in C^2(\mathbb{R}^n) \) if a point \( \mathbf{x}_* \in \mathbb{R}^n \) is a local minimum then \( \nabla f(\mathbf{x}_*)^T = 0 \) and \( \nabla^2 f(\mathbf{x}_*) \) is semi-definite positive, i.e.

\[
d^T \nabla^2 f(\mathbf{x}_*) d \geq 0, \quad \forall d \in \mathbb{R}^n
\]

Example

This condition is only, necessary, in fact consider \( f(\mathbf{x}) = x_1^2 - x_2^3 \),

\[
\nabla f(\mathbf{x}) = (2x_1, -3x_2^2), \quad \nabla^2 f(\mathbf{x}) = \begin{pmatrix} 2 & 0 \\ 0 & -6x_2 \end{pmatrix}
\]

for the point \( \mathbf{x}_* = 0 \) we have \( \nabla f(0) = 0 \) and \( \nabla^2 f(0) \) semi-definite positive, but \( 0 \) is a saddle point not a minimum.
Proof.

The condition $\nabla f(x_\star)^T = 0$ comes from first order necessary conditions. Consider now a generic direction $d$, and the finite difference:

$$
\frac{f(x_\star + \lambda d) - 2f(x_\star) + f(x_\star - \lambda d)}{\lambda^2} \geq 0
$$

by using Taylor expansion for $f(x)$

$$
f(x_\star \pm \lambda d) = f(x_\star) \pm \nabla f(x_\star) \lambda d + \frac{\lambda^2}{2} d^T \nabla^2 f(x_\star) d + o(\lambda^2)
$$

and from the previous inequality

$$
d^T \nabla^2 f(x_\star) d + 2o(\lambda^2)/\lambda^2 \geq 0
$$

taking the limit $\lambda \to 0$ and form the arbitrariness of $d$ we have that $\nabla^2 f(x_\star)$ must be semi-definite positive.
Lemma (Second order sufficient condition for local minimum)

Given \( f \in C^2(\mathbb{R}^n) \) if a point \( x_\star \in \mathbb{R}^n \) satisfy:

1. \( \nabla f(x_\star)^T = 0 \);
2. \( \nabla^2 f(x_\star) \) is definite positive; i.e.

\[
d^T \nabla^2 f(x_\star) d > 0, \quad \forall d \in \mathbb{R}^n \setminus \{x_\star\}
\]

then \( x_\star \in \mathbb{R}^n \) is a strict local minimum.

Remark

Because \( \nabla^2 f(x_\star) \) is symmetric we can write

\[
\lambda_{\min} d^T d \leq d^T \nabla^2 f(x_\star) d \leq \lambda_{\max} d^T d
\]

If \( \nabla^2 f(x_\star) \) is positive definite we have \( \lambda_{\min} > 0 \).
Proof.

Consider now a generic direction $d$, and the Taylor expansion for $f(x)$

$$f(x_\star + d) = f(x_\star) + \nabla f(x_\star) d + \frac{1}{2} d^T \nabla^2 f(x_\star) d + o(\|d\|^2)$$

$$\geq f(x_\star) + \frac{1}{2} \lambda_{min} \|d\|^2 + o(\|d\|^2)$$

$$\geq f(x_\star) + \frac{1}{2} \lambda_{min} \|d\|^2 \left(1 + o(\|d\|^2)/\|d\|^2\right)$$

choosing $d$ small enough we can write

$$f(x_\star + d) \geq f(x_\star) + \frac{1}{4} \lambda_{min} \|d\|^2 > f(x_\star), \quad d \neq 0, \|d\| \leq \delta.$$

i.e. $x_\star$ is a strict minimum.
Outline

1. General iterative scheme

2. Backtracking Armijo line-search
   - Global convergence of backtracking Armijo line-search
   - Global convergence of steepest descent

3. Wolfe–Zoutendijk global convergence
   - The Wolfe conditions
   - The Armijo-Goldstein conditions

4. Algorithms for line-search
   - Armijo Parabolic-Cubic search
   - Wolfe linesearch
General iterative scheme

How to find a minimum

Given \( f : \mathbb{R}^n \mapsto \mathbb{R} \): minimize \( \min_{x \in \mathbb{R}^n} f(x) \).

1. We can solve the problem by solving the necessary condition. i.e by solving the nonlinear systems
   \[
   \nabla f(x)^T = 0.
   \]

2. Using such an approach we looses the information about \( f(x) \).

3. Moreover such an approach can find solution corresponding to a maximum or saddle points.

4. A better approach is to use all the information and try to build minimizing procedure, i.e. procedures that, starting from a point \( x_0 \) build a sequence \( \{x_k\} \) such that \( f(x_{k+1}) \leq f(x_k) \). In this way, at least, we avoid to converge to a strict maximum.
in practice very rare to be able to provide explicit minimizer.

iterative method: given starting guess $\mathbf{x}_0$, generate the sequence,

$$\{\mathbf{x}_k\}, \quad k = 1, 2, \ldots$$

**AIM:** ensure that (a subsequence) has some favorable limiting properties:

- satisfies first-order necessary conditions
- satisfies second-order necessary conditions
A generic iterative minimization procedure can be sketched as follows:

- calculate a search direction $p_k$ from $x_k$
- ensure that this direction is a descent direction, i.e.

\[ \nabla f(x_k) p_k < 0, \quad \text{whenever } \nabla f(x_k)^T \neq 0 \]

so that, at least for small steps along $p_k$, the objective function $f(x)$ will be reduced.

- use line-search to calculate a suitable step-length $\alpha_k > 0$ so that

\[ f(x_k + \alpha_k p_k) < f(x_k). \]

- Update the point:

\[ x_{k+1} = x_k + \alpha_k p_k \]
Written with a pseudo-code the minimization procedure is the following algorithm:

**Generic minimization algorithm**

Given an initial guess $x_0$, let $k = 0$;

while not converged do

   Find a descent direction $p_k$ at $x_k$;
   Compute a step size $\alpha_k$ using a line-search along $p_k$.

   Set $x_{k+1} = x_k + \alpha_k p_k$ and increase $k$ by 1.

end while

The crucial points which differentiate the algorithms are:

1. The computation of the direction $p_k$;
2. The computation of the step size $\alpha_k$. 
The first developed minimization algorithms try to solve

$$\alpha_k = \arg \min_{\alpha > 0} f(x_k + \alpha p_k)$$

- performing exact line-search by univariate minimization;
- rather expensive and certainly not cost effective.

Modern methods implements inexact line-search:
- ensure steps are neither too long nor too short
- try to pick *useful* initial step size for fast convergence
- best methods are based on:
  - backtracking–Armijo search;
  - Armijo–Goldstein search;
  - Franke–Wolfe search;
To obtain a monotone decreasing sequence we can use the following algorithm:

**Backtracking line-search**

Given $\alpha_{\text{init}}$ (e.g., $\alpha_{\text{init}} = 1$);
Given $\tau \in (0, 1)$ typically $\tau = 0.5$;
Let $\alpha^{(0)} = \alpha_{\text{init}}$;
while not $f(x_k + \alpha^{(\ell)} p_k) < f(x_k)$ do
    set $\alpha^{(\ell+1)} = \tau \alpha^{(\ell)}$;
    increase $\ell$ by 1;
end while
Set $\alpha_k = \alpha^{(\ell)}$.

To be effective the previous algorithm should terminate in a finite number of steps. The next lemma assure that if $p_k$ is a descent direction then the algorithm terminate.
Outline

1. General iterative scheme

2. Backtracking Armijo line-search
   - Global convergence of backtracking Armijo line-search
   - Global convergence of steepest descent

3. Wolfe–Zoutendijk global convergence
   - The Wolfe conditions
   - The Armijo-Goldstein conditions

4. Algorithms for line-search
   - Armijo Parabolic-Cubic search
   - Wolfe linesearch
To prevent large steps relative to the decreasing of $f(x)$ we require that

$$f(x_k + \alpha_k p_k) \leq f(x_k) + \alpha_k \beta \nabla f(x_k) p_k$$

for some $\beta \in (0, 1)$. Typical values of $\beta$ ranges form $10^{-4}$ to 0.1.
Backtracking Armijo line-search

Given $\alpha_{\text{init}}$ (e.g., $\alpha_{\text{init}} = 1$);
Given $\tau \in (0, 1)$ typically $\tau = 0.5$;
Let $\alpha^{(0)} = \alpha_{\text{init}}$;

while not $f(x_k + \alpha^{(\ell)} p_k) \leq f(x_k) + \alpha^{(\ell)} \beta \nabla f(x_k) p_k$ do
  set $\alpha^{(\ell+1)} = \tau \alpha^{(\ell)}$;
  increase $\ell$ by 1;
end while
Set $\alpha_k = \alpha^{(\ell)}$.

- Backtracking Armijo line-search prevents the step from getting too large.
- Now the question is: will the backtracking Armijo line-search terminate in a finite number of steps?
Theorem (Finite termination of Armijo line-search)

Suppose that $f(x)$ satisfy the standard assumptions and $\beta \in (0, 1)$ and that $p_k$ is a descent direction at $x_k$. Then the Armijo condition

$$f(x_k + \alpha_k p_k) \leq f(x_k) + \alpha_k \beta \nabla f(x_k) p_k$$

is satisfied for all $\alpha_k \in [0, \omega_k]$ where

$$\omega_k = \frac{2(\beta - 1) \nabla f(x_k) p_k}{\gamma \|p_k\|^2}$$

Assumption (Regularity assumption)

We assume $f \in C^1(\mathbb{R}^n)$ with Lipschitz continuous gradient, i.e. there exists $\gamma > 0$ such that

$$\|\nabla f(x) - \nabla f(y)\| \leq \gamma \|x - y\|, \quad \forall x, y \in \mathbb{R}^n$$
Finite termination of Armijo line-search

To prove finite termination we need the following Taylor expansion due to the regularity assumption:

$$f(x + \alpha p) = f(x) + \alpha \nabla f(x)p + E \quad \text{where} \quad |E| \leq \frac{\gamma}{2} \alpha^2 \|p\|^2$$

Proof.

If $\alpha \leq \omega_k$ we have $\alpha \gamma \|p_k\|^2 \leq 2(\beta - 1) \nabla f(x_k)p_k$ and by using Taylor expansion

$$f(x_k + \alpha p_k) \leq f(x_k) + \alpha \nabla f(x_k)p_k + \frac{\gamma}{2} \alpha^2 \|p_k\|^2$$

$$\leq f(x_k) + \alpha \nabla f(x_k)p_k + \alpha(\beta - 1) \nabla f(x_k)p_k$$

$$\leq f(x_k) + \alpha \beta \nabla f(x_k)p_k$$
Corollary (Finite termination of Armijo linesearch)

Suppose that $f(x)$ satisfy the standard assumptions and $\beta \in (0, 1)$ and that $p_k$ is a descent direction at $x_k$. Then the step-size generated by the backtracking-Armijo line-search terminates with

$$\alpha_k \geq \min \{\alpha_{\text{init}}, \tau \omega_k\}, \quad \omega_k = 2(\beta - 1)\nabla f(x_k)p_k / (\gamma \|p_k\|^2)$$

Proof.

Line-search will terminate as soon as $\alpha^{(\ell)} \leq \omega_k$:

1. May be that $\alpha_{\text{init}}$ satisfies the Armijo condition $\Rightarrow \alpha_k = \alpha_{\text{init}}$.
2. Otherwise in the last line-search iteration we have

$$\alpha^{(\ell-1)} > \omega_k, \quad \alpha_k = \alpha^{(\ell)} = \tau \alpha^{(\ell-1)} > \tau \omega_k.$$

Combining these 2 cases gives the required result.
The previous analysis permit to say that Backtracking-Armijo line-search ends in a finite number of steps.

The line-search produce a step length not too long due to the condition

$$f(x_k + \alpha_k p_k) \leq f(x_k) + \alpha_k \beta \nabla f(x_k) p_k$$

The line-search produce a step length not too short due to the finite termination theorem.

Armijo line-search can be improved by adding some further requirements on the step length acceptance criteria.
Theorem (Global convergence)

Suppose that $f(x)$ satisfy the standard assumptions, then, for the iterates generated by the **Generic minimization algorithm with backtracking Armijo line-search** either:

1. $\nabla f(x_k)^T = 0$ for some $k \geq 0$;
2. $\lim_{k \to \infty} f(x_k) = -\infty$;
3. $\lim_{k \to \infty} |\nabla f(x_k)p_k| \min \left\{ 1, \|p_k\|^{-1} \right\} = 0$.

Remark

If the theorem, point 1 means that we found a stationary point in a finite number of steps. Point 2 means that function $f(x)$ is unbounded below, so that a minimum does not exists. Point 3 alone do not imply convergence, but if $\nabla f(x_k)$ and $p_k$ do not become orthogonal and $\|p_k\| \not\to 0$ then $\|\nabla f(x_k)\| \to 0$. 

Unconstrained minimization
Proof. Assume points 1 and 2 are not satisfied, then we prove point 3. Consider
\[ f(x_{k+1}) \leq f(x_k) + \alpha_k \beta \nabla f(x_k) p_k \leq f(x_0) + \sum_{j=0}^{k} \alpha_j \beta \nabla f(x_j) p_j \]
by the fact that \( p_k \) is a descent direction we have that the series:
\[ \sum_{j=0}^\infty \alpha_j |\nabla f(x_j) p_j| \leq \beta^{-1} \lim_{k \to \infty} [f(x_0) - f(x_{k+1})] < \infty \]
and then
\[ \lim_{j \to \infty} \alpha_j |\nabla f(x_j) p_j| = 0 \]
Proof. \((2/3)\).

Recall that

\[
\alpha_k \geq \min \{ \alpha_{\text{init}}, \tau \omega_k \}, \quad \omega_k = 2(\beta - 1)\nabla f(x_k)p_k / (\gamma \|p_k\|^2)
\]

and consider the two index set:

\[
\mathcal{K}_1 = \{ k \mid \alpha_k = \alpha_{\text{init}} \}, \quad \mathcal{K}_2 = \{ k \mid \alpha_k < \alpha_{\text{init}} \},
\]

Obviously \( \mathbb{N} = \mathcal{K}_1 \cup \mathcal{K}_2 \) and from \( \lim_{k \to \infty} \alpha_k |\nabla f(x_k)p_k| = 0 \) we have

\[
\lim_{k \in \mathcal{K}_1 \to \infty} \alpha_k |\nabla f(x_k)p_k| = 0, \quad (A)
\]

\[
\lim_{k \in \mathcal{K}_2 \to \infty} \alpha_k |\nabla f(x_k)p_k| = 0, \quad (B)
\]
Proof. (3/3).

For $k \in K_1$ we have $\alpha_k = \alpha_{\text{init}}$ and

$$\alpha_k |\nabla f(x_k)p_k| = \alpha_{\text{init}} |\nabla f(x_k)p_k|$$

and from (A) we have

$$\lim_{k \in K_1 \to \infty} |\nabla f(x_k)p_k| = 0 \tag{\star}$$

For $k \in K_2$ we have $\tau \omega_k \leq \alpha_k \leq \omega_k$ so

$$\alpha_k |\nabla f(x_k)p_k| \geq \tau \omega_k |\nabla f(x_k)p_k| \geq 2\tau(1 - \beta) \frac{|\nabla f(x_k)p_k|^2}{\gamma \|p_k\|^2}$$

and from (B) we have

$$\lim_{k \in K_1 \to \infty} \frac{|\nabla f(x_k)p_k|}{\|p_k\|} = 0 \tag{\star\star}$$

Combining (\star) and (\star\star) gives the required result.
Given an initial guess $x_0$, let $k = 0$;

while not converged do

Compute a step-size $\alpha_k$ using a line-search along $-\nabla f(x_k)^T$.

Set $x_{k+1} = x_k - \alpha_k \nabla f(x_k)^T$ and increase $k$ by 1.

end while

- The steepest descent algorithm is simply the generic minimization algorithm with search direction the opposite of the gradient in $x_k$.
- The search direction $-\nabla f(x_k)^T$ is always a descent direction unless the point $x_k$ is a stationary point.
Global convergence of steepest descent

Corollary (Global convergence of steepest descent)

Suppose that $f(x)$ satisfy the standard assumptions, then, for the iterates generated by the steepest descent algorithm with backtracking Armijo line-search either:

1. $\nabla f(x_k)^T = 0$ for some $k \geq 0$;
2. $\lim_{k \to \infty} f(x_k) = -\infty$;
3. $\lim_{k \to \infty} \nabla f(x_k)^T = 0$. 
Although the **steepest descent** scheme is globally convergent it can be very slow!

A classical example is the Rosenbrock function:

\[ f(x, y) = 100 (y - x^2)^2 + (x - 1)^2 \]
This function has a unique minimum at \((1, 1)^T\) inside a banana shaped valley.
After 100 iteration starting from \((-1.2, 1)^T\) the approximate minimum is far from the solution.
The steepest descent is a slow method, not only on a difficult test case like the Rosenbrock example.

Given the function $f(x, y) = \frac{1}{2}x^2 + \frac{9}{2}y^2$ starting from $x_0 = (9, 1)^T$ we have the zig-zag pattern toward $(0, 0)^T$. 
Outline

1. General iterative scheme

2. Backtracking Armijo line-search
   - Global convergence of backtracking Armijo line-search
   - Global convergence of steepest descent

3. Wolfe–Zoutendijk global convergence
   - The Wolfe conditions
   - The Armijo-Goldstein conditions

4. Algorithms for line-search
   - Armijo Parabolic-Cubic search
   - Wolfe linesearch
The simple condition of descent step is in general not enough for the convergence of an iterative minimization scheme.

The condition of sufficient decrease of backtracking Armijo line-search may be insufficient on general inexact line-search algorithm.

Adding another condition to the sufficient decrease condition such that we avoid too short step length we obtain globally convergent numerical procedure.

Depending on which additional condition is added we obtain the:

1. Wolfe conditions;
2. Armijo Goldstein conditions.
The Wolfe conditions

Let $c_1$ and $c_2$ two constant such that $0 < c_1 < c_2 < 1$. We say that the step length $\alpha_k$ satisfy the Wolfe conditions if $\alpha_k$ satisfy:

1. **sufficient decrease**: $f(x_k + \alpha_k p_k) \leq f(x_k) + c_1 \alpha_k \nabla f(x_k) p_k$;
2. **curvature condition**: $\nabla f(x_k + \alpha_k p_k) p_k \geq c_2 \nabla f(x_k) p_k$.

![Diagram illustrating the Wolfe conditions](image-url)
The strong Wolfe conditions

Let $c_1$ and $c_2$ two constant such that $0 < c_1 < c_2 < 1$. We say that the step length $\alpha_k$ satisfy the strong Wolfe conditions if $\alpha_k$ satisfy:

1. sufficient decrease: $f(x_k + \alpha_k p_k) \leq f(x_k) + c_1 \alpha_k \nabla f(x_k) p_k$;
2. curvature condition: $|\nabla f(x_k + \alpha_k p_k)p_k| \leq c_2 |\nabla f(x_k)p_k|$. 

Unconstrained minimization
The Wolfe condition seems quite restrictive.

The next lemma answers to the question if a step length satisfying Wolfe conditions does exist.

**Lemma (strong Wolfe step length)**

Let $f : \mathbb{R}^n \mapsto \mathbb{R}$ satisfying the regularity assumption. If the following conditions are satisfied:

1. $p_k$ is a descent direction for the point $x_k$, i.e. $\nabla f(x_k)p_k < 0$;
2. $f(x_k + \alpha p_k)$ is bounded from below, i.e. $\lim_{\alpha \to \infty} f(x_k + \alpha p_k) > -\infty$.

then for any $0 < c_1 < c_2 < 1$ there exists an interval $[a, b]$ such that all $\alpha_k \in [a, b]$ satisfy the strong Wolfe conditions.
Proof.

Define $\ell(\alpha) = f(x_k) + \alpha c_1 \nabla f(x_k)p_k$ and $g(\alpha) = f(x_k + \alpha p_k)$. From $\lim_{\alpha \to \infty} \ell(\alpha) = -\infty$ and from condition 1 it follows that there exists $\alpha_* > 0$ such that

$$\ell(\alpha_*) = g(\alpha_*) \quad \text{and} \quad \ell(\alpha) > g(\alpha), \quad \forall \alpha \in (0, \alpha_*)$$

so that all step length $\alpha \in (0, \alpha_*)$ satisfy strong Wolfe condition 1. Because $\ell(0) = g(0)$ from Cauchy-Rolle theorem there exists $\alpha_{**} \in (0, \alpha_*)$ such that

$$g'(\alpha_{**}) = \ell'(\alpha_{**}) \quad \Rightarrow$$

$$\nabla f(x_k + \alpha_{**}p_k)p_k = c_1 \nabla f(x_k)p_k > c_2 \nabla f(x_k)p_k$$

by continuity we find an interval around $\alpha_{**}$ with step lengths satisfying strong Wolfe conditions.
The Zoutendijk condition

Theorem (Zoutendijk)

Let \( f : \mathbb{R}^n \mapsto \mathbb{R} \) satisfying the regularity assumption and bounded from below, i.e.

\[
\inf_{x \in \mathbb{R}^n} f(x) > -\infty
\]

Let \( \{x_k\}, k = 0, 1, \ldots, \infty \) generated by a generic minimization algorithm where line-search satisfy Wolfe conditions, then

\[
\sum_{k=1}^{\infty} (\cos \theta_k)^2 \| \nabla f(x_k)^T \|^2 < +\infty
\]

where

\[
\cos \theta_k = \frac{-\nabla f(x_k)p_k}{\| \nabla f(x_k)^T \| \| p_k \|}
\]
Proof. (1/3).

Using the second condition of Wolfe

\[ \nabla f(x_k + \alpha_k p_k) p_k \geq c_2 \nabla f(x_k) p_k \]

\[ (\nabla f(x_k + \alpha_k p_k) - \nabla f(x_k)) p_k \geq (c_2 - 1) \nabla f(x_k) p_k \]

by using Lipschitz regularity

\[ \left| (\nabla f(x_k + \alpha_k p_k) - \nabla f(x_k)) p_k \right| \leq \gamma \| x_{k+1} - x_k \| \| p_k \| \]

\[ = \alpha_k \gamma \| p_k \|^2 \]

and using both inequality we obtain the estimate for \( \alpha_k \):

\[ \alpha_k \geq \frac{c_2 - 1}{\gamma \| p_k \|^2} \nabla f(x_k) p_k \]
Proof. (2/3).

Using the first condition of Wolfe and estimate of $\alpha_k$

\[
f(x_k + \alpha_k p_k) \leq f(x_k) + \alpha_k c_1 \nabla f(x_k) p_k\]

\[
\leq f(x_k) - \frac{c_1(1 - c_2)}{\gamma \|p_k\|^2} (\nabla f(x_k) p_k)^2
\]

setting $A = c_1(1 - c_2)/\gamma$ and using the definition of $\cos \theta_k$

\[
f(x_{k+1}) = f(x_k + \alpha_k p_k) \leq f(x_k) - A (\cos \theta_k)^2 \left\| \nabla f(x_k) \right\|_2^2
\]

and by induction

\[
f(x_{k+1}) \leq f(x_1) - A \sum_{j=1}^{k} (\cos \theta_j)^2 \left\| \nabla f(x_j) \right\|_2^2
\]
Proof. The function $f(x)$ is bounded from below, i.e.

$$\inf_{x \in \mathbb{R}^n} f(x) > -\infty$$

so that

$$A \sum_{j=1}^{k} (\cos \theta_j)^2 \left\| \nabla f(x_j)^T \right\|^2 \leq f(x_1) - f(x_{k+1})$$

and

$$A \sum_{j=1}^{\infty} (\cos \theta_j)^2 \left\| \nabla f(x_j)^T \right\|^2 \leq f(x_1) - \lim_{k \to \infty} f(x_{k+1}) < +\infty$$
Corollary (Zoutendijk condition)

Let $f : \mathbb{R}^n \mapsto \mathbb{R}$ satisfying the regularity assumption and bounded from below. Let $\{x_k\},\ k = 0, 1, \ldots, \infty$ generated by a generic minimization algorithm where line-search satisfy Wolfe conditions, then

$$\cos \theta_k \left\| \nabla f(x_k)^T \right\| \to 0$$

where

$$\cos \theta_k = \frac{-\nabla f(x_k) p_k}{\left\| \nabla f(x_k)^T \right\| \left\| p_k \right\|}$$

Remark

If $\cos \theta_k \geq \delta > 0$ for all $k$ from the Zoutendijk condition we have:

$$\left\| \nabla f(x_k)^T \right\| \to 0$$

i.e. the generic minimization algorithm where line-search satisfy Wolfe conditions converge to a stationary point.
The Armijo-Goldstein conditions

Let $c_1$ and $c_2$ two constant such that $0 < c_1 < c_2 < 1$. We say that the step length $\alpha_k$ satisfy the Wolfe conditions if $\alpha_k$ satisfy:

1. $f(x_k + \alpha_k p_k) \leq f(x_k) + c_1 \alpha_k \nabla f(x_k) p_k$;
2. $f(x_k + \alpha_k p_k) \geq f(x_k) + c_2 \alpha_k \nabla f(x_k) p_k$;
The Armijo-Goldstein conditions

1. Armijo-Goldstein conditions has very similar theoretical properties like the Wolfe conditions.

2. Global convergence theorems can be established.

3. The weakness of Armijo-Goldstein conditions respect to Wolfe conditions is that the former can exclude local minima’s from the step length as you can see in the figure below.

\[ f(x_k) + \alpha c_1 \nabla f(x_k) p_k \]

\[ f(x_k + \alpha p_k) \]

\[ f(x_k) + \alpha c_2 \nabla f(x_k) p_k \]
Outline

1. General iterative scheme

2. Backtracking Armijo line-search
   - Global convergence of backtracking Armijo line-search
   - Global convergence of steepest descent

3. Wolfe–Zoutendijk global convergence
   - The Wolfe conditions
   - The Armijo-Goldstein conditions

4. Algorithms for line-search
   - Armijo Parabolic-Cubic search
   - Wolfe linesearch
Armijo Parabolic-Cubic search

1. Backtracking-Armijo line-search can be slow if a large number of reduction must be performed to satisfy Armijo condition.

2. A better performance is obtained if instead of reducing by a fixed factor we use polynomial interpolation to estimate the location of the minimum.

3. Assuming that that $f(x_k)$ and $\nabla f(x_k)p_k$ are known at the first step we know also $f(x_k + \lambda p_k)$ if $\lambda$ is the first trial step.

4. In this case a parabolic interpolation can be used to estimate the minimum.

5. If we store the last trial step length, in the successive iteration we can use cubic interpolation to estimate the minima’s.

6. The resulting algorithm is in the following slides.
Algorithm (Armijo Parabolic-Cubic search (1/3))

\[ \text{armijo_linesearch}(f, x, p, c_1) \]

\[ f_0 \leftarrow f(x); \ \nabla f_0 \leftarrow \nabla f(x)p; \ \lambda \leftarrow 1; \]

\textbf{while} \( \lambda \geq \lambda_{\text{min}} \) \textbf{do}

\[ f_\lambda \leftarrow f(x + \lambda p); \]

\textbf{if} \( f_\lambda \leq f_0 + \lambda c_1 \nabla f_0 \) \textbf{then}

\[ \text{return} \ \lambda; \ \text{successful search} \]

\textbf{else}

\textbf{if} \( \lambda = 1 \) \textbf{then}

\[ \lambda_{\text{tmp}} \leftarrow \nabla f_0 / [2(f_0 + \nabla f_0 - f_\lambda)]; \]

\textbf{else}

\[ \lambda_{\text{tmp}} \leftarrow \text{cubic}(f_0, \nabla f_0, f_\lambda, \lambda, f_p, \lambda_p); \]

\textbf{end if}

\[ \lambda_p \leftarrow \lambda; \ f_p \leftarrow f_\lambda; \ \lambda \leftarrow \text{range}(\lambda_{\text{tmp}}, \lambda/10, \lambda/2); \]

\textbf{end if}

\textbf{end while}

\[ \text{return} \ \lambda_{\text{min}}; \ \text{failed search} \]
Algorithm (Armijo Parabolic-Cubic search)

\[ \text{range}(\lambda, a, b) \]

if \( \lambda < a \) then
  return \( a \);
else if \( \lambda > b \) then
  return \( b \);
else
  return \( \lambda \);
end if
Algorithm (Armijo Parabolic-Cubic search) (3/3)

\textit{cubic}(f_0, \nabla f_0, f_\lambda, \lambda, f_p, \lambda_p)

Evaluate:

\[
\begin{pmatrix}
  a \\
  b \\
\end{pmatrix} = \frac{1}{\lambda^2 \lambda_p^2 (\lambda - \lambda_p)} \begin{pmatrix}
  \lambda_p^2 & -\lambda^2 \\
  -\lambda_p^3 & \lambda^3 \\
\end{pmatrix} \begin{pmatrix}
  f_\lambda - f_0 - \lambda \nabla f_0 \\
  f_p - f_0 - \lambda_p \nabla f_0 \\
\end{pmatrix}
\]

\textbf{if} \ a = 0 \ \textbf{then}

\textbf{return} \ -\nabla f_0/(2b); \quad \textit{cubic is a quadratic}

\textbf{else}

\[d \leftarrow b^2 - 3a \nabla f_0;\]

\textbf{return} \ (-b + \sqrt{d})/(3a); \quad \textit{discriminant legitimate cubic}

\textbf{end if}
Wolfe linesearch is identical to the Armijo Parabolic-Cubic search, until a point satisfying the first condition is found.

At this point the Armijo algorithm stop while Wolfe search try to refine the search until the second condition is satisfied.

If the step estimated is too short then is is enlarged until it contains a minimum.

If the step estimated is too long it is reduced until the second condition is satisfied.
Algorithm (Wolfe linesearch)

\texttt{wolfe_linesearch}(f, x, p, c_1, c_2)

\[ f_0 \leftarrow f(x); \quad \nabla f_0 \leftarrow \nabla f(x)p; \quad \lambda \leftarrow 1; \]

\textbf{while} \( \lambda \geq \lambda_{\text{min}} \) \textbf{do}

\[ f_\lambda \leftarrow f(x + \lambda p); \]

\textbf{if} \( f_\lambda \leq f_0 + \lambda c_1 \nabla f_0 \) \textbf{then}

\hspace{1em} \textbf{go to} \ ZOOM; \textit{found a } \lambda \text{ satisfying condition 1}

\textbf{else}

\hspace{1em} \textbf{if} \( \lambda = 1 \) \textbf{then}

\hspace{2em} \lambda_{\text{tmp}} \leftarrow \nabla f_0 / [2(f_0 + \nabla f_0 - f_\lambda)];

\hspace{1em} \textbf{else}

\hspace{3em} \lambda_{\text{tmp}} \leftarrow \texttt{cubic}(f_0, \nabla f_0, f_\lambda, \lambda, f_p, \lambda_p);

\hspace{1em} \textbf{end if}

\hspace{1em} \lambda_p \leftarrow \lambda; \quad f_p \leftarrow f_\lambda; \quad \lambda \leftarrow \texttt{range}(\lambda_{\text{tmp}}, \lambda/10, \lambda/2);

\textbf{end if}

\textbf{end while}

\textbf{return} \( \lambda_{\text{min}} \); \textit{failed search}
Algorithm (Wolfe linesearch)

**ZOOM:**
\[ \nabla f_x \leftarrow \nabla f(x + \lambda p)p; \]
if \( \nabla f_x \geq c_2 \nabla f_0 \) then return \( \lambda; \) found Wolfe point!
if \( \lambda = 1 \) then
*forward search of an interval bracketing a minimum*

while \( \lambda \leq \lambda_{max} \) do
\[ \{\lambda_p, f_p\} \leftarrow \{\lambda, f_x\}; \]
\( \lambda \leftarrow 2\lambda; \) \( f_x \leftarrow f(x + \lambda p); \)
if not \( f_x \leq f_0 + \lambda c_1 \nabla f_0 \) then
\[ \{\lambda_p, f_p\} \leftarrow \{\lambda, f_x\}; \text{ go to REFINE; swap values} \]
end if
\[ \nabla f_x \leftarrow \nabla f(x + \lambda p)p; \]
if \( \nabla f_x \geq c_2 \nabla f_0 \) then return \( \lambda; \) found Wolfe point!
end while

return \( \lambda_{max} \); failed search
Algorithm (Wolfe linesearch (3/3))

REFINE:
\[
\{\lambda_{lo}, f_{lo}, \nabla f_{lo}\} \leftarrow \{\lambda, f_\lambda, \nabla f_\lambda\}; \quad \Delta \leftarrow \lambda_p - \lambda_{lo};
\]
while \(\Delta > \epsilon\) do
\[
\delta \lambda \leftarrow \Delta^2 \nabla f_{lo} / \left[2(f_{lo} + \nabla f_{lo}\Delta - f_p)\right];
\]
\[
\delta \lambda \leftarrow \text{range}(\delta \lambda, 0.2\Delta, 0.8\Delta);
\]
\[
\lambda \leftarrow \lambda_{lo} + \delta \lambda; \quad f_\lambda \leftarrow f(x + \lambda p);
\]
if \(f_\lambda \leq f_0 + \lambda c_1 \nabla f_0\) then
\[
\nabla f_\lambda \leftarrow \nabla f(x + \lambda p)p;
\]
if \(\nabla f_\lambda \geq c_2 \nabla f_0\) then
return \(\lambda\);
\[
\{\lambda_{lo}, f_{lo}, \nabla f_{lo}\} \leftarrow \{\lambda, f_\lambda, \nabla f_\lambda\}; \quad \Delta \leftarrow \Delta - \delta \lambda;
\]
else
\[
\{\lambda_p, f_p\} \leftarrow \{\lambda, f_\lambda\}; \quad \Delta \leftarrow \delta \lambda;
\]
end if
end while
return \(\lambda\); failed search

found Wolfe point!
References

J. Stoer and R. Bulirsch
Introduction to numerical analysis

J. E. Dennis, Jr. and Robert B. Schnabel
Numerical Methods for Unconstrained Optimization and Nonlinear Equations