Chapter 4. Unconstrained optimization

Version: 31-10-2011

Material: *(for details see)*

- **Chapter 11 in [FKS] (pp.251-276)**

A reference e.g. L.11.2 refers to the corresponding Lemma in the book [FKS]

PDF-file of the Book [FKS]: ***Faigle/Kern/Still, Algorithmic principles of Mathematical Programming.***
on: ***http://wwwhome.math.utwente.nl/∼stillgj/priv/***

4.1 Introduction

Consider the nonlinear minimization problem with $f \in C^1$ or $f \in C^2$:

$$(P) \qquad \min_{x \in \mathbb{R}^n} f(x) \; ,$$

Recall: Usually in (*nonconvex*) ***Unconstrained Optimization*** we try to find a local minimizer

Global minimization is 'much more' difficult.

Theoretical method: (based on optimality conditions)

- Find a point $\overline{x}$ satisfying

$$\nabla f(\overline{x}) = 0 \qquad \textit{(critical point)}$$

- Check whether $\nabla^2 f(\overline{x}) \succ 0$.

CONCEPTUAL ALGORITHM: Choose $x_0 \in \mathbb{R}^n$. Iterate

- **step k:** Given $x_k \in \mathbb{R}^n$, find a new point x_{k+1} with $f(x_{k+1}) < f(x_k)$.

We hope that: $x_k \to \overline{x}$ with $\overline{x}$ a local mininimizer.

Def. Let $x_k \to \overline{x}$ for $k \to \infty$. The sequence (x_k) has a:

- *linear convergence* if with a constant $0 \leq C < 1$ and some $K \in \mathbb{N}$:

$$\|x_{k+1} - \overline{x}\| \leq C\|x_k - \overline{x}\|, \qquad \forall k \geq K.$$

C is the convergence factor.

- *quadratic convergence* if with a constant $c \geq 0$,

$$\|x_{k+1} - \overline{x}\| \leq c\|x_k - \overline{x}\|^2, \qquad k \in \mathbb{N}.$$

- *superlinear convergence* if

$$\lim_{k\to\infty} \frac{\|x_{k+1} - \overline{x}\|}{\|x_k - \overline{x}\|} = 0 .$$

4.2 General descent method ($f \in C^1$)

Def. A vector $d_k \in \mathbb{R}^n$ is called a *descent direction* in x_k if

$$\nabla f(x_k)^T d_k < 0 \qquad (*)$$

Rem. *If* $(*)$ *holds then for any* $t > 0$ *small enough:* $f(x_k + td_k) < f(x_k)$

Abbreviation: $g(x) := \nabla f(x), \quad g_k := g(x_k)$

Conceptual DESCENT METHOD: Choose a starting point $x_0 \in \mathbb{R}^n$ and $\epsilon > 0$. Iterate

step k: Given $x_k \in \mathbb{R}^n$, proceed as follows:

- if $||g(x_k)|| < \epsilon$, stop with $\overline{x} \approx x_k$.
- Choose a descent direction d_k in x_k: $\quad g_k^T d_k < 0$
- Find a solution t_k of the (one-dimensional) minimization problem

$$\min_{t \geq 0} f(x_k + td_k) \quad \text{and put } x_{k+1} = x_k + t_k d_k. \qquad (**)$$

Rem. Minimization in $\mathbb{R}^n$ is reduced to (line) minimization in $\mathbb{R}$ (in each step k).

Steepest descent method: Use in the *descent method* as descent direction (*see Ex.11.7*): $\boxed{d_k = -\nabla f(x_k)}$

Ex.11.7 Assuming $\nabla f(x_k) \neq 0$, show that $d_k = -[\nabla f(x_k)]/\|\nabla f(x_k)\|$ solves the problem:

$$\min_{d \in \mathbb{R}^n} \nabla f(x_k)^T d \qquad \text{s.t.} \qquad \|d\| = 1$$

Convergence behavior:

L.11.3 In the line-minimization step (**) we have

$$\nabla f(x_{k+1})^T d_k = 0$$

For the steepest descent method this means:

$$d_{k+1}^T d_k = 0 \qquad \textit{(ziggzagging)}$$

Th.11.1 Let $f \in C^1$. Apply the steepest descent method. If the iterates x_k converge, *i.e.*, $x_k \to \overline{x}$, then $\boxed{\nabla f(\overline{x}) = 0}$

Rem. The next example shows that in general *(even for min of quadratic functions)*, the steepest descent method cannot be expected to converge better than *linearly*.

Ex.11.9. Apply the steepest descent method to

$$q(x) = x^T \begin{pmatrix} 1 & 0 \\ 0 & r \end{pmatrix} x \quad , \quad r \geq 1$$

Then with $x_0 = (r, 1)$ it follows

$$x_k = \left(\frac{r-1}{r+1} \right)^k (r, (-1)^k) \ .$$

(*Linear convergence to* $\overline{x} = 0$ *with factor* $C = (r-1)/(r+1)$.)

HINT: Make use of [FKS,L.11.8] and apply induction wrt. k.

4.3 Method of conjugate directions

Aim: *Find an algorithm which (at least for quadratic functions) has "better convergence" than steepest descent.*

4.3.1 Case: $f(x) = q(x) := \frac{1}{2}x^T Ax + b^T x$, $A \succ 0$ **(pd.)**

Idea. Try to generate d_k's such that
(*not only* $\nabla q(x_{k+1})^T d_k = 0$ *but***)**

$$\nabla q(x_{k+1})^T d_j = 0 \quad \forall 0 \leq j \leq k$$

Then, after n steps we have

$$\nabla q(x_n)^T d_j = 0 \quad \forall 0 \leq j \leq n-1$$

and (*if the d_j's are lin. indep.***)** $\nabla q(x_n) = 0$**. So** $x_n = -A^{-1}b$ **is the minimizer of q.**

L. 11.4 Apply the descent method to $q(x)$. The following are equivalent:

(i) $g_{j+1}^T d_i = 0 \quad \text{for all } 0 \le i \le j \le k;$

(ii) $d_j^T A d_i = 0 \quad \text{for all } 0 \le i < j \le k.$

Definition. Vectors $d_0, \dots, d_{n-1} \neq 0$ are called A-conjugate (or A-orthogonal) if: $d_j^T A d_i = 0 \quad \forall i \neq j$.

Ex. A collection of A-conjugate vectors $d_0, \dots, d_{n-1} \neq 0$ in $\mathbb{R}^n$ are linearly independent.

Construction To obtain the conditions in L.11.4, simply try

$$\boxed{d_k = -g_k + \alpha_k d_{k-1}}$$

Then $d_k^T A d_{k-1} = 0$ implies $\alpha_k = \frac{g_k^T A d_{k-1}}{d_{k-1}^T A d_{k-1}}$.

Th.11.3 Apply the descent method to $q(x)$ with

$$d_k = -g_k + \alpha_k d_{k-1} \ , \quad \alpha_k = \frac{g_k^T A d_{k-1}}{d_{k-1}^T A d_{k-1}}$$

Then the d_k's are A-conjugate. In particular, the algorithm stops after *(at most)* n steps with the unique minimizer $\overline{x} = -A^{-1}b$ of q.

Conjugate Gradient Method (CG)

```
INIT: Choose $x_0 \in \mathbb{R}^n$, $\varepsilon > 0$, $d_0 := -g_0$;
ITER: WHILE $\|g_k\| \geq \varepsilon$ DO
BEGIN
    Determine a solution $t_k$ for the problem
    $(*) \qquad \min_{t \geq 0} f(x_k + t d_k)$
    Set $x_{k+1} = x_k + t_k d_k$ .
    Set $d_{k+1} = -g_{k+1} + \alpha_{k+1} d_k$ .
END
```

Ex.11.10 Under the assumptions of Th.11.3, show that the iteration point x_{k+1} is the (global) minimizer of the quadratic function q on the affine subspace

$$S_k = \{x_0 + \gamma_0 d_0 + .. + \gamma_k d_k \mid \gamma_0, .., \gamma_k \in \mathbb{R}\}$$

4.3.2 Case: non-quadratic functions $f(x)$

Note that for quadratic function $f = q$ we have:

$$\begin{aligned}\alpha_{k+1} &= \frac{g_{k+1}^T A d_k}{d_k^T A d_k} = \frac{g_{k+1}^T (g_{k+1} - g_k)}{d_k^T (g_{k+1} - g_k)} \\ &= \frac{g_{k+1}^T (g_{k+1} - g_k)}{\|g_k\|^2} = \frac{\|g_{k+1}\|^2}{\|g_k\|^2}\end{aligned}$$

So, for non-quadratic $f(x)$ in the CG-algorithm we can can use $d_{k+1} = -g_{k+1} + \alpha_{k+1} d_k$ with:

Hestenes-Stiefel (1952): $\alpha_{k+1} = \dfrac{g_{k+1}^T (g_{k+1} - g_k)}{d_k^T (g_{k+1} - g_k)}$

Fletcher-Reeves (1964): $\alpha_{k+1} = \dfrac{\|g_{k+1}\|^2}{\|g_k\|^2}$

Polak-Ribiere (1969): $\alpha_{k+1} = \dfrac{g_{k+1}^T (g_{k+1} - g_k)}{\|g_k\|^2}$

Application to sparse systems $Ax = b, \quad A \succ 0$

Def. $A = (a_{ij})$ is sparse if less than

α% of the a_{ij}-s are $\neq 0$ with *(say)* $\alpha \leq 5$

CG-method: apply the CG-method to

$$\min \frac{1}{2} x^T A x - b^T x \quad \text{with solution } \overline{x} = A^{-1} b$$

CG Method for sparse linear systems $Ax = b,\ A \succ 0$

```
INIT: Choose $x_0 \in \mathbb{R}^n$ and $\varepsilon > 0$ and set $d_0 = -g_0$;
ITER: WHILE $\|g_k\| \geq \varepsilon$ DO
BEGIN
    Set $x_{k+1} = x_k + t_k d_k$ with $t_k = -\frac{g_k^T d_k}{d_k^T A d_k}$
    Set $g_{k+1} = g_k + t_k A d_k$
    Set $d_{k+1} = -g_{k+1} + \alpha_{k+1} d_k$ with $\alpha_{k+1} = \frac{g_{k+1}^T g_{k+1}}{g_k^T g_k}$
END
```

Rem. **Complexity:** $\approx \frac{\alpha}{100} n^2$ **flop's (***floating point operations***) per ITER.**

4.4 Line minimization

In the general descent method (see Ch.4.2) we have to repeatedly solve:

$$\min_{t\geq 0} \ h(t) \quad \text{with } h(t) = f(x_k + td_k)$$

where $h'(0) < 0$.

This can be done by:

- ***'exact line minimization'* of numerical analysis**
  e.g., bisection, golden section, Newton-,
  secant method (see Ch.4.3, Ch.11.4.1)
- **or more "efficiently" by *'inexact line search'***
  Goldstein-, Goldstein-Wolfe test (see Ch.11.4.2)

4.5 Newton's method:

General remark: *Newton's method for solving systems of nonlinear equations is one of the most important tools of applied mathematics.*

Newtons's Iteration: **For solving $F(x) = 0$**

(a system of n equations in n unknowns $\mathbf{x} = (x_1, \ldots, x_n)$)

$$\boxed{\mathbf{x}_{k+1} = \mathbf{x}_k - [\nabla F(\mathbf{x}_k)]^{-1} F(\mathbf{x}_k)}$$

Th.11.4 (***local convergence of Newton's method***)

Given $F : \mathbb{R}^n \to \mathbb{R}^n$, $F \in C^2$ such that

$$F(\overline{x}) = 0 \quad \text{and} \quad \nabla F(\overline{x}) \text{ is non-singular.}$$

Then the Newton iteration x_k converges quadratically to $\overline{x}$ for any x_0 sufficiently close to $\overline{x}$.

'Newton' for solving : $\min f(x)$ *or* $F(x) := \nabla f(x) = 0$,

$$\boxed{x_{k+1} = x_k - [\nabla^2 f(x_k)]^{-1} \nabla f(x_k)}$$

(local) quadratic convergence to $\overline{x}$ if:

$f \in C^3$, $\nabla f(\overline{x}) = 0$ with $\nabla^2 f(\overline{x})$ non-singular.

Problems with this Newton method for: $\min f(x)$

- $x_k \to \overline{x}$ possibly a loc. maximizer.
- $x_k \to x_{k+1}$ with *increasing "f"*

Newton descent method: The 'Newton direction'

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

is a *descent direction at* x_k ($g_k^T d_k < 0$) if (*assume* $\nabla f(x_k) \neq 0$):

$[\nabla^2 f(x_k)]^{-1}$ or equivalently $\nabla^2 f(x_k)$ is positive definite.

Algorithm: *(Levenberg-Marquardt variant)*

```
step k:   Given $x_k \in \mathbb{R}^n$ with $g_k \neq 0$.
  1. determine $\sigma_k > 0$ such that $(\nabla^2 f(x_k) + \sigma_k I) \succ 0$,
       compute   $d_k = -\big(\nabla^2 f(x_k) + \sigma_k I\big)^{-1} g_k \quad (*)$
  2. Find a minimizer $t_k$ of $\min_{t \geq 0} f(x_k + t d_k)$
       and put $x_{k+1} = x_k + t_k d_k$.
```

Ex.11.n1 *[connection with the 'trust region method']*
Consider the quadratic Taylor approximation of f near x_k:

$$\begin{aligned} q(x) \quad := \quad & f(x_k) + \nabla f(x_k)^T (x - x_k) \\ & + 1/2 (x - x_k)^T \nabla^2 f(x_k)(x - x_k) \end{aligned}$$

Compute the descent step d_k according to $(*)$ *(Levenberg-Marquardt)* **and put $x_{k+1} = x_k + d_k$, $\tau := \|d_k\|$. Show that x_{k+1} is a local minimizer of the** ***trust region problem*****:** $\min\ q(x) \quad$ s.t. $\quad \|x - x_k\| \leq \tau$

Disadvantage of the Newton methods:

- $\nabla^2 f(x_k)$ needed
- work per step: linear system
  $F_k x = b_k$ $\approx n^3$ flop's

4.6 Quasi-Newton method.

Find a method *which only makes use of first derivatives and only needs $O(n^2)$ flop's per iter.*

Consider the descent method with:

$$d_k = -H_k g_k$$

desired properties for H_k:

- i $H_k \succ 0$
- ii $H_{k+1} = H_k + E_k$ simple update rule
- iii for quadratic f → conjugate directions d_j
- iv the ***Quasi-Newton condition***:

$$(x_{k+1} - x_k) = H_{k+1}(g_{k+1} - g_k)$$

Notation: $\delta_k := (x_{k+1} - x_k) \ , \quad \gamma_k := (g_{k+1} - g_k)$

```
*Quasi-Newton Method*
INIT: Choose some $x_0 \in \mathbb{R}^n$, $H_0 \succ 0$, $\varepsilon > 0$
ITER: WHILE $\|g_k\| \geq \varepsilon$ DO
BEGIN
    Set $d_k = -H_k g_k$,
    Determine a solution $t_k$ for the problem
        $\min_{t \geq 0} f(x_k + t d_k)$
    Set $x_{k+1} = x_k + t_k d_k$ and update
      $H_{k+1} = H_k + E_k$ .
END
```

For the *update* $H_k + E_k$ we try: with $\alpha, \beta, \mu \in \mathbb{R}$

$$E_k = \alpha u u^T + \beta v v^T + \mu(u v^T + v u^T) \quad (\star)$$

where $u := \delta_k$, $v := H_k \gamma_k$

Note that E_k is symmetric with rank ≤ 2.

L.11.5 Apply the Quasi-Newton method to $q(x) = \frac{1}{2}x^TAx + b^Tx$, $A \succ 0$ with

E_k of the form $(\star)$ and

H_{k+1} satisfying iv: $\delta_k = H_{k+1}\gamma_k$

Then the directions d_j are A-conjugate :

$$d_j^TAd_i = 0 \qquad 0 \leq i < j \leq k$$

Last step in the construction of E_k: Find α, β, μ in $(\star)$ such that (iv) holds. This leads to the following *update.*

Broyden family: with $\Phi \in \mathbb{R}$

$$H_{k+1} = H_k + \boxed{\frac{\delta_k \delta_k^T}{\delta_k^T \gamma_k} - \frac{H_k \gamma_k \gamma_k^T H_k}{\gamma_k^T H_k \gamma_k} + \Phi\, w w^T} \qquad (**)$$

where $\quad w := \left(\frac{\delta_k}{\delta_k^T \gamma_k} - \frac{H_k \gamma_k}{\gamma_k^T H_k \gamma_k} \right) (\gamma_k^T H_k \gamma_k)^{\frac{1}{2}}.$

As special cases we obtain:

$\Phi = 0$, the *DFP-method* (1963)
(Davidon, Fletcher, Powell)

$\Phi = 1$, the *BFGS-method* (1970)
(Broyden, Fletcher, Goldfarb, Shanno)

Finally we show that property i), $H_k \succ 0$*, is preserved.*

L.11.6 In the Quasi-Newton method, if we use $(**)$ with $\Phi \geq 0$, then: $\quad H_k \succ 0 \quad \Rightarrow \quad H_{k+1} \succ 0$