

4TE3/6TE3
Algorithms for
Continuous Optimization
(Search directions-I)

Tamás TERLAKY
Computing and Software
McMaster University

Hamilton, January 2004

terlaky@mcmaster.ca

Tel: 27780

Generic Algorithm

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & x \in \mathcal{C}. \end{array}$$

Input:

$\epsilon > 0$ is the accuracy parameter;
 x^0 is a given (relative interior) feasible point;

Step 0: $x := x^0, k = 0;$

Step 1: Find **search direction** s^k

$$\text{s.t. } \delta f(x^k, s^k) < 0;$$

(This should be a descending feasible direction
in the constrained case.)

Step 1a: If no such direction exists **STOP**,
optimum found.

Step 2: Line search : find $\lambda^k = \min_{\lambda} f(x^k + \lambda s^k);$

Step 3: $x^{k+1} = x^k + \lambda^k s^k, k = k + 1;$

Step 4: If **stopping criteria** satisfied **STOP**,
else GOTO Step 1.

Search direction

Gradient method

$$s = -\nabla f(x^k)$$

Steepest descent direction!

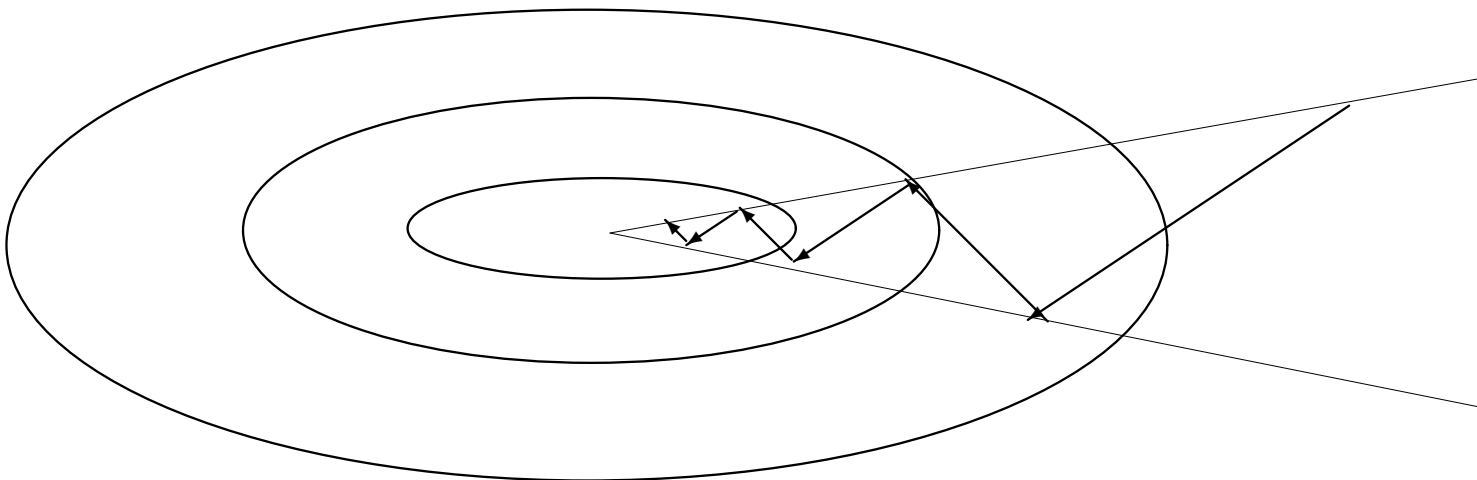
$$\begin{aligned} \delta f(x, -\nabla f(x)) &= -\nabla f(x)^T \nabla f(x) \\ &= \min_{\|s\|=\|\nabla f(x)\|} \{\nabla f(x)^T s\}. \end{aligned}$$

Cost of one iteration: $O(n)$ + line search.

The (negative) gradient is orthogonal to the level curves.

Not a finite algorithm, even not for quadratic functions.

Slow convergence, “zigg-zagging” is possible.



Convergence

of the gradient method

Theorem 1 *Let f be continuously differentiable. Starting from the initial point x^0 using exact line search the gradient method produces a decreasing sequence x^0, x^1, x^2, \dots such that $f(x^k) > f(x^{k+1})$ for $k = 0, 1, 2, \dots$. Assume that the level set $D = \{x : f(x) \leq f(x^0)\}$ is compact, then any accumulation point \bar{x} of the generated sequence $x^0, x^1, x^2, \dots, x^k, \dots$ is a stationary point (i.e. $\nabla f(\bar{x}) = 0$) of f . If the function f is a convex function, then \bar{x} is a global minimizer of f .*

Proof: Since D is compact and f is continuous we have that f is bounded on D , hence we have a convergent subsequence $x^{k_j} \rightarrow \bar{x}$ with $f(x^{k_j}) \rightarrow f^*$ as $k_j \rightarrow \infty$. By continuity of f we have $f(\bar{x}) = f^*$. Since the search direction is the gradient of f we have

$$\bar{s} = \lim_{k_j \rightarrow \infty} s^{k_j} = - \lim_{k_j \rightarrow \infty} \nabla f(x^{k_j}) = -\nabla f(\bar{x}).$$

Multiplying by $\nabla f(\bar{x})$ we have

$$\bar{s}^T \nabla f(\bar{x}) = -\nabla f(\bar{x})^T \nabla f(\bar{x}) \leq 0. \quad (*)$$

On the other hand using the construction of the iteration sequence and the convergent subsequence we write

$$f(x^{k_{j+1}}) \leq f(x^{k_j+1}) \leq f(x^{k_j} + \lambda s^{k_j}).$$

Taking the limit in the last inequality we have

$$f(\bar{x}) \leq f(\bar{x} + \lambda \bar{s})$$

which leads to $\delta f(\bar{x}, \bar{s}) = \bar{s}^T \nabla f(\bar{x}) \geq 0$. Combining this result with (*) we have $\nabla f(\bar{x}) = 0$, and the theorem is proved. \square

The order of convergence

The order of convergence is only linear, speed depends on the conditioning of the Hessian. Let $q(x) = \frac{1}{2}x^T Ax - b^T x$ with $A : n \times n$ symmetric and positive definite, $b \in R^n$.

Let $0 < \mu_1 \leq \mu_2 \leq \dots \leq \mu_n$ be the eigenvalues of A .

$$\text{Let } r = \frac{\mu_n}{\mu_1}$$

(condition number of A), degree of difficulty.

Let $E(x) = q(x) + \frac{1}{2}E(x^*) = \frac{1}{2}(x - x^*)^T A(x - x^*)$, where x^* minimizes the function $q(x)$.

Theorem 2 *Let x^0 be an arbitrary starting point of the steepest decent method applied to minimize $E(x)$. Then the steepest descent method converges to the unique minimum x^* of $E(x)$ (and $q(x)$), further*

$$\text{as } E(x^{k+1}) \leq \left(\frac{r-1}{r+1}\right)^2 E(x^k) = \left(\frac{\mu_n - \mu_1}{\mu_n + \mu_1}\right)^2 E(x^k).$$

Newton's method

f is twice continuously differentiable and strictly convex.

Newton's method is based on minimizing the second order approximation of f .

$$q(x) := f(x^k) + \nabla f(x^k)^T(x - x^k) + \frac{1}{2}(x - x^k)^T \nabla^2 f(x^k)(x - x^k).$$

The Hessian $\nabla^2 f(x^k)$ is positive definite (PD), thus $q(x)$ is strictly convex. Hence the minimum is attained when

$$\nabla q(x) = \nabla f(x^k) + \nabla^2 f(x^k)(x - x^k) = 0.$$

Thus,

$$x^{k+1} = x^k - (\nabla^2 f(x^k))^{-1} \nabla f(x^k).$$

Local quadratic convergence with full step (without any line search!).

Good starting point is essential.

If line search is applied \rightarrow *damped Newton* method.

Cost of one iteration: $O(n^3)$ + line search.

To reduce computational costs:

quasi-Newton methods,

methods of conjugate directions.

Hessian not PD: then do *Trust-region method*.

Newton's method for

nonlinear equations:

$$F(x) = 0$$

Linearize at x^k : $F(x) \approx F(x^k) + JF(x^k)(x - x^k)$

$$JF(x)_{ij} = \frac{\partial F_i(x)}{\partial x_j} \quad \text{where } i = 1, \dots, m; \quad j = 1, \dots, n.$$

$$JF(x^k)(x^{k+1} - x^k) = -F(x^k).$$

Minimize $f(x) \Leftrightarrow \nabla f(x) = 0$.

$$\nabla^2 f(x^k)(x^{k+1} - x^k) = -\nabla f(x^k).$$

The Jacobian of the gradient is exactly the Hessian of the function $f(x)$ hence it is positive definite and we have

$$x^{k+1} = x^k - (\nabla^2 f(x^k))^{-1} \nabla f(x^k)$$

as we have seen above.

Descent directions (once more)

Hessian/modified Hessian

At a given point x in the directions s the directional derivative is:

$$\delta f(x, s) = \nabla f(x)^T s.$$

Let $s = -H\nabla f(x)$ (like in Newton's method), then s is a descent direction, i.e.,

$$\delta f(x, s) = \nabla f(x)^T s = -\nabla f(x)^T H \nabla f(x) < 0$$

iff H is positive definite. That is why a positive definite Hessian is needed for Newton's method.

Observe: ANY positive definite matrix H gives a descent direction!

- If $H = I$ then we got the gradient method.
- If $H = (\nabla^2 f(x))^{-1}$ then we have Newton's method.
- If $H = (\nabla^2 f(x) + \alpha I)^{-1}$ then we got the Trust Region method.

Fiacco-McCormick's modification

Newton-like algorithm for Nonconvex functions

Let $LDL^T = (\nabla^2 f(x))^{-1}$, if it exists, where D is a diagonal and L is a lower triangular matrix.

Choose the search direction as follows:

- If the factorization exists and $\text{diag}(D)$ is positive, then use Newton, i.e., $s = -(\nabla^2 f(x))^{-1} \nabla f(x)$.
- If the factorization exists and D is not positive, then let

$$y = -\text{sign}(\text{sign}(\text{diag}(D)) - I).$$

Then solve

$$L^T s = y$$

which imply

$$s^T (\nabla^2)^{-1} f(x) s = s^T LDL^T s < 0.$$

If $\delta f(x, s) < 0$ then use s ;

If $\delta f(x, -s) < 0$ then use $-s$;

If $\delta f(x, s) = 0$ then use $s = -\nabla f(x, s)$.

- If the factorization does not exist then use steepest descent, i.e., $s = -\nabla f(x)$.

Trust-region method - I

If the function $f(x)$ is not strictly convex, or if the Hessian is ill-conditioned the Hessian is not (or hardly) invertible.

Remedy: **trust-region method.**

Quadratic approximation of the function $f(x)$:

$$q(x) = f(x^k) + \nabla f(x^k)^T(x - x^k) + \frac{1}{2}(x - x^k)^T \nabla^2 f(x^k)^T(x - x^k).$$

If the Hessian is not PD, we perturb it to PD:

$$Q(x) = f(x^k) + \nabla f(x^k)^T(x - x^k) + \frac{1}{2}(x - x^k)^T (\nabla^2 f(x^k)^T + \alpha I)(x - x^k).$$

$$Q(x) = q(x) + \frac{\alpha}{2} \|x - x^k\|^2$$

The minimum of this (observe $\nabla^2 f(x^k)$ is replaced by $(\nabla^2 f(x) + \alpha I)$ comparing Newton);

$$s^k(\alpha) = x^{k+1} - x^k = -(\nabla^2 f(x^k) + \alpha I)^{-1} \nabla f(x^k).$$

Then $x^k + s^k(\alpha)$ minimizes $Q(x)$, but we want a decrease in $q(x)$ as well:

$$\begin{aligned} Q(x^k + s^k(\alpha)) &\leq Q(x) \\ q(x^k + s^k(\alpha)) + \frac{\alpha}{2} \|s^k(\alpha)\|^2 &\leq q(x) + \frac{\alpha}{2} \|x - x^k\|^2 \\ q(x^k + s^k(\alpha)) &\leq q(x) \quad \text{if } \|x - x^k\| \leq \|s^k(\alpha)\| =: r_\alpha \end{aligned}$$

Trust -region method - II

$$r_\alpha := \|s^k(\alpha)\| = \left\| - \left(\nabla^2 f(x^k) + \alpha I \right)^{-1} \nabla f(x^k) \right\|$$

Lemma 1 r_α is a non-increasing function of α .

Proof: Let z_1, \dots, z_n be an orthonormal eigenvector system of $\nabla^2 f(x^k)$ with the ordered eigenvalues $\mu_1 \leq \dots \leq \mu_n$. Further, let $\nabla f(x^k) = \sum_{i=1}^n c_i z_i$.

Then

$$s^k(\alpha) = - \sum_{i=1}^n \frac{c_i}{\mu_i + \alpha} z_i$$

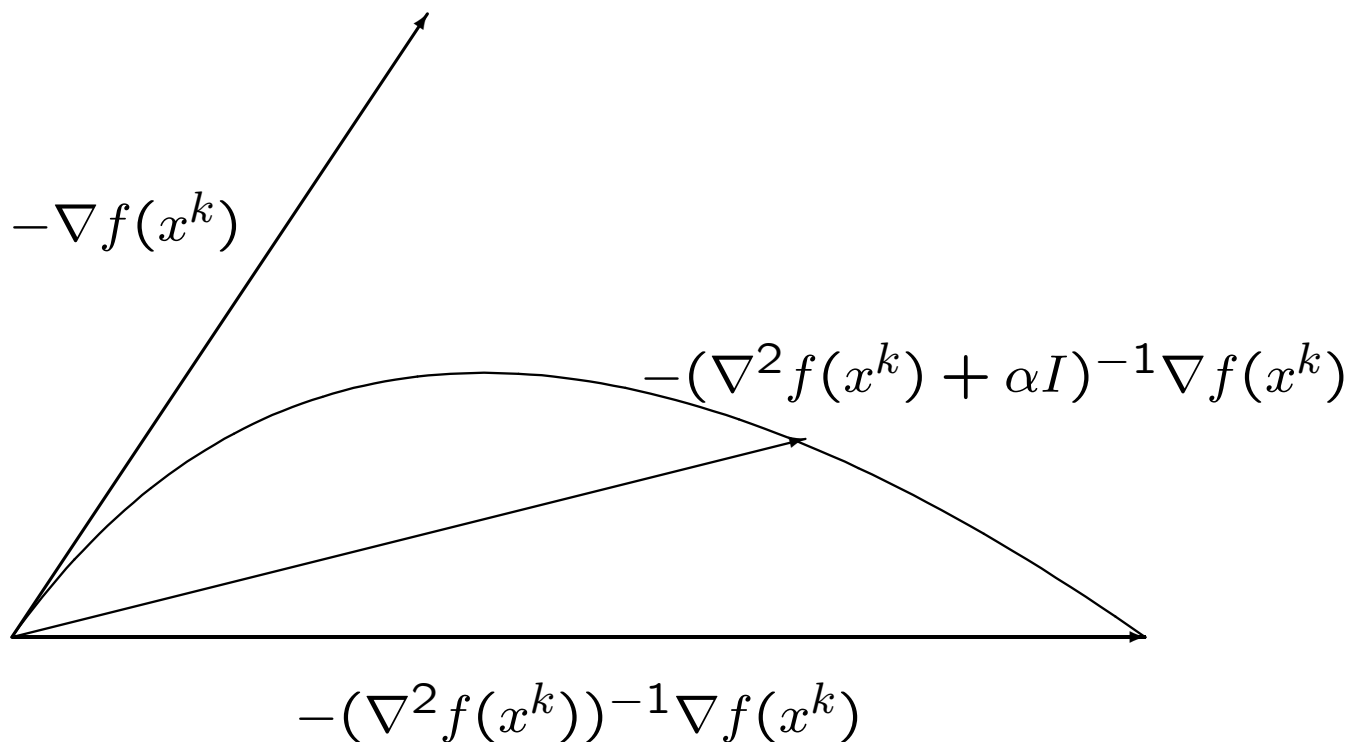
$$r_\alpha = \sqrt{\sum_{i=1}^n \frac{c_i^2}{(\mu_i + \alpha)^2}}$$

is clearly non-increasing with α increasing. \square

The bigger the α , the smaller the step!

Trust region method - III

The Trust-Region (TR) direction is a compromise between the gradient and Newton directions:



If $\alpha = 0$ then we have the Newton step,
as $\alpha \rightarrow \infty$ then we approach a small multiple of the
negative gradient.

Trust region method - IV

The update of α :

Let x^k and $\alpha > 0$ be the current iterate.

Let $0 < \mu < \eta < 1$ and $0 < \gamma_1 < 1 < \gamma_2$ be given.

Typical values: $\mu = \frac{1}{4}$, $\eta = \frac{3}{4}$, $\gamma_1 = \frac{1}{2}$, $\gamma_2 = 2$.

A usual starting value is $\alpha = 1$.

If x^k satisfies termination criteria, then **STOP**.

Calculate $s^k(\alpha)$.

Compute

$$\rho^k = \frac{f(x^k) - f(x^k + s^k(\alpha))}{f(x^k) - q(x^k + s^k(\alpha))} = \frac{\text{actual decrease}}{\text{predicted decrease}}$$

If $\rho^k < \mu$ **then** (step failed)

$$x^{k+1} := x^k \text{ and } \alpha := \gamma_2 \alpha.$$

If $\mu \leq \rho^k \leq \eta$ **then** (step as predicted)

$$x^{k+1} := x^k + s^k(\alpha) \text{ and } \alpha := \alpha.$$

If $\rho^k > \eta$ **then** (step is very good)

$$x^{k+1} := x^k + s^k(\alpha) \text{ and } \alpha := \gamma_1 \alpha.$$

In order to avoid exact line search

α is dynamically increased and decreased.

Trust region method - V

Why is it called TRUST-REGION?

Newton step was given by minimizing the quadratic approximation:

$$q(x) = f(x^k) + \nabla f(x^k)^T (x - x^k) + \frac{1}{2} (x - x^k)^T \nabla^2 f(x^k) (x - x^k).$$

How far you TRUST that $q(x)$ is a good approximation of $f(x)$? Not far, thus you replace the Newton sub-problem by the TR subproblem

$$\begin{array}{ll} \min & q(x) \\ \text{s.t.} & \|x - x^k\|^2 \leq \Delta_k^2. \end{array}$$

As we will see later (duality, Lagrange function) the optimum of this problem is obtained by minimizing the Lagrange function

$$Q(x) = f(x^k) + \nabla f(x^k)^T (x - x^k) + \frac{1}{2} (x - x^k)^T (\nabla^2 f(x^k) + \alpha I) (x - x^k),$$

where α is an appropriate Lagrange multiplier. This results in

$$x^{k+1} = x^k - (\nabla^2 f(x^k) + \alpha I)^{-1} \nabla f(x^k).$$

It can be verified:

The bigger the Δ_k is, the smaller α is.

Thus if you solve the TR subproblem directly in a TR algorithm, you should decrease Δ_k when α is increased and increase Δ_k when α is decreased.