Part 2: Linesearch methods for unconstrained optimization

Nick Gould (RAL)

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad f(x)$$

MSc course on nonlinear optimization

UNCONSTRAINED MINIMIZATION

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \;\; f(x)$$

where the **objective function** $f : \mathbb{R}^n \longrightarrow \mathbb{R}$

- ⊙ assume that $f \in C^1$ (sometimes C^2) and Lipschitz
- ⊙ often in practice this assumption violated, but not necessary

ITERATIVE METHODS

- in practice very rare to be able to provide explicit minimizer
- iterative method: given starting "guess" x_0, generate sequence

$$\{x_k\}, \quad k = 1, 2, \ldots$$

- **AIM:** ensure that (a subsequence) has some favourable limiting properties:
  - satisfies first-order necessary conditions
  - satisfies second-order necessary conditions

Notation: $f_k = f(x_k)$, $g_k = g(x_k)$, $H_k = H(x_k)$.

LINESEARCH METHODS

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

  $$g_k^T p_k < 0 \quad \text{if} \quad g_k \neq 0$$

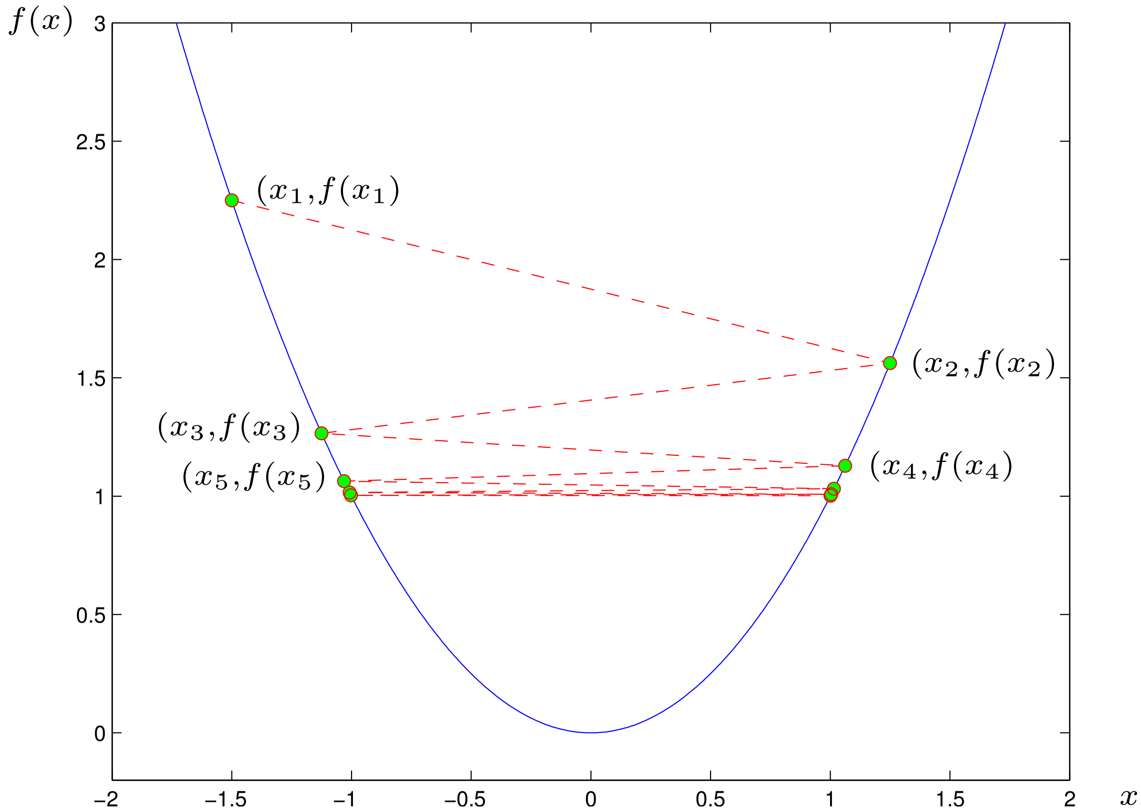
  so that, for small steps along p_k, the objective function **will** be reduced
- calculate a suitable **steplength** $\alpha_k > 0$ so that

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

- computation of α_k is the **linesearch**—may itself be an iteration
- generic linesearch method:

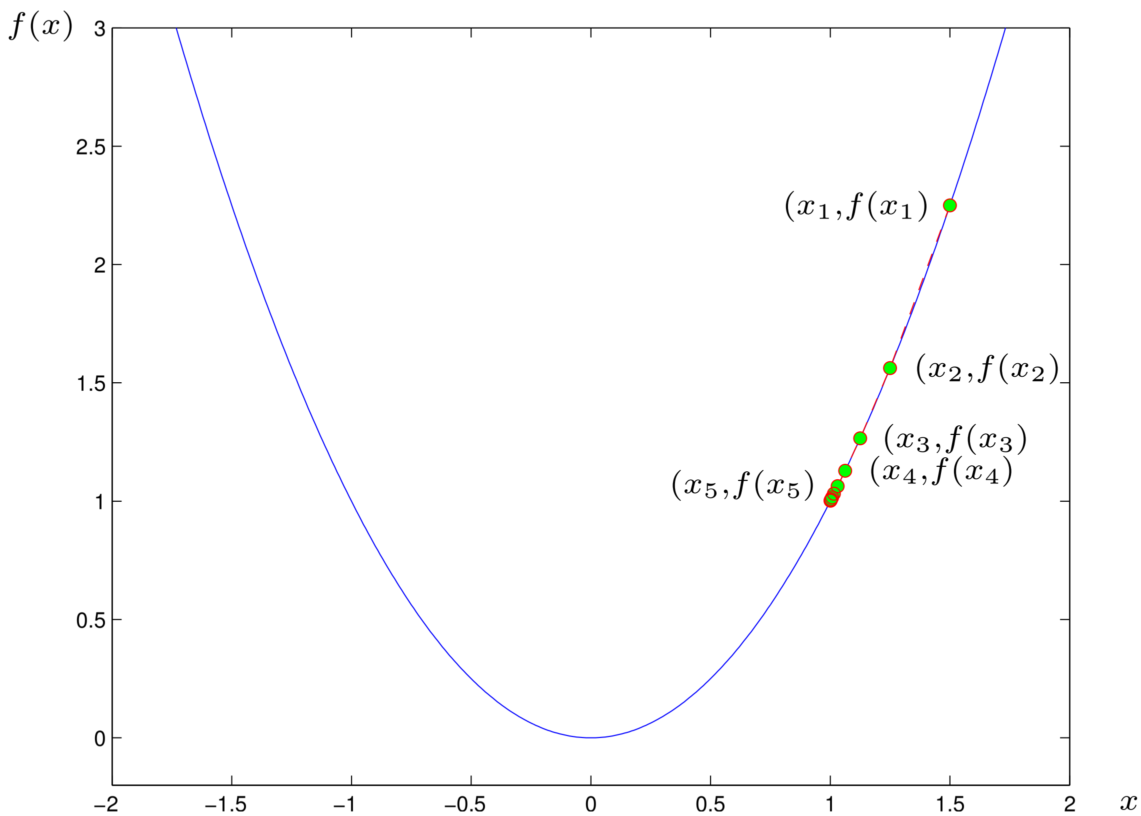
  $$x_{k+1} = x_k + \alpha_k p_k$$

STEPS MIGHT BE TOO LONG



The objective function $f(x) = x^2$ and the iterates $x_{k+1} = x_k + \alpha_k p_k$ generated by the descent directions $p_k = (-1)^{k+1}$ and steps $\alpha_k = 2 + 3/2^{k+1}$ from $x_0 = 2$

STEPS MIGHT BE TOO SHORT



The objective function $f(x) = x^2$ and the iterates $x_{k+1} = x_k + \alpha_k p_k$ generated by the descent directions $p_k = -1$ and steps $\alpha_k = 1/2^{k+1}$ from $x_0 = 2$

PRACTICAL LINESEARCH METHODS

- in early days, pick α_k to minimize

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

  - **exact** linesearch—univariate minimization
  - rather expensive and certainly not cost effective

- modern methods: **inexact** linesearch
  - ensure steps are neither too long nor too short
  - try to pick "useful" initial stepsize for fast convergence
  - best methods are either
    - "backtracking- Armijo" or
    - "Armijo-Goldstein"

    based

BACKTRACKING LINESEARCH

Procedure to find the stepsize α_k:

Given $\alpha_{\text{init}} > 0$ (e.g., $\alpha_{\text{init}} = 1$)
let $\alpha^{(0)} = \alpha_{\text{init}}$ and $l = 0$
Until $f(x_k + \alpha^{(l)} p_k)$ "$<$" f_k
    set $\alpha^{(l+1)} = \tau \alpha^{(l)}$, where $\tau \in (0, 1)$ (e.g., $\tau = \frac{1}{2}$)
    and increase l by 1
Set $\alpha_k = \alpha^{(l)}$

- this prevents the step from getting too small ... but does not prevent too large steps relative to decrease in f
- need to tighten requirement

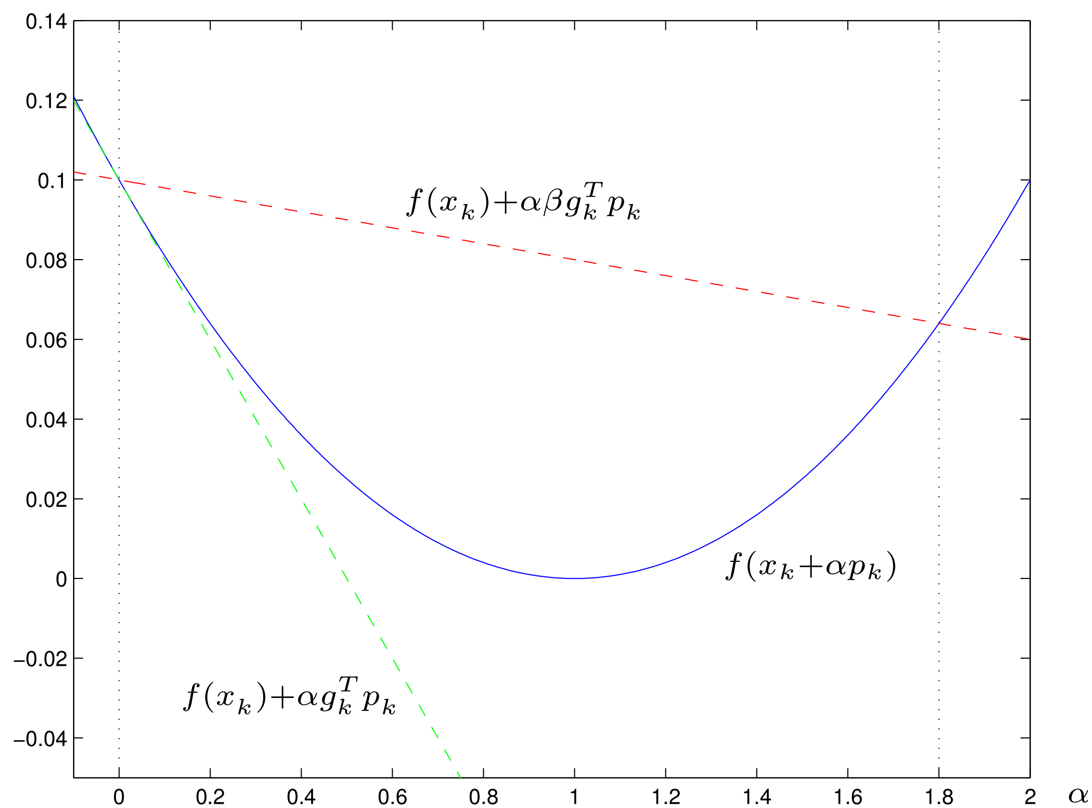
$$f(x_k + \alpha^{(l)} p_k) \text{ ``}<\text{''} f_k$$

ARMIJO CONDITION

In order to prevent large steps relative to decrease in f, instead require

$$f(x_k + \alpha_k p_k) \leq f(x_k) + \alpha_k \beta g_k^T p_k$$

for some $\beta \in (0,1)$ (e.g., $\beta = 0.1$ or even $\beta = 0.0001$)



BACKTRACKING-ARMIJO LINESEARCH

Procedure to find the stepsize α_k:

Given $\alpha_{\text{init}} > 0$ (e.g., $\alpha_{\text{init}} = 1$)
let $\alpha^{(0)} = \alpha_{\text{init}}$ and $l = 0$
Until $f(x_k + \alpha^{(l)} p_k) \leq f(x_k) + \alpha^{(l)} \beta g_k^T p_k$
    set $\alpha^{(l+1)} = \tau \alpha^{(l)}$, where $\tau \in (0, 1)$ (e.g., $\tau = \frac{1}{2}$)
    and increase l by 1
Set $\alpha_k = \alpha^{(l)}$

SATISFYING THE ARMIJO CONDITION

Theorem 2.1. Suppose that $f \in C^1$, that $g(x)$ is Lipschitz continuous with Lipschitz constant $\gamma(x)$, that $\beta \in (0,1)$ and that p is a descent direction at x. Then the Armijo condition

$$f(x + \alpha p) \leq f(x) + \alpha\beta g(x)^T p$$

is satisfied for all $\alpha \in [0, \alpha_{\max(x)}]$, where

$$\alpha_{\max} = \frac{2(\beta - 1)g(x)^T p}{\gamma(x)\|p\|_2^2}$$

PROOF OF THEOREM 2.1

Taylor's theorem (Theorem 1.1) +

$$\alpha \leq \frac{2(\beta - 1) g(x)^T p}{\gamma(x) \|p\|_2^2},$$

$\Longrightarrow$

$$\begin{aligned} f(x + \alpha p) &\leq f(x) + \alpha g(x)^T p + \tfrac{1}{2} \gamma(x) \alpha^2 \|p\|^2 \\ &\leq f(x) + \alpha g(x)^T p + \alpha(\beta - 1) g(x)^T p \\ &= f(x) + \alpha \beta g(x)^T p \end{aligned}$$

THE ARMIJO LINESEARCH TERMINATES

Corollary 2.2. Suppose that $f \in C^1$, that $g(x)$ is Lipschitz continuous with Lipschitz constant γ_k at x_k, that $\beta \in (0,1)$ and that p_k is a descent direction at x_k. Then the stepsize generated by the backtracking-Armijo linesearch terminates with

$$\alpha_k \geq \min\left(\alpha_{\text{init}}, \frac{2\tau(\beta-1)g_k^T p_k}{\gamma_k \|p_k\|_2^2}\right)$$

PROOF OF COROLLARY 2.2

Theorem 2.1 $\Longrightarrow$ linesearch will terminate as soon as $\alpha^{(l)} \leq \alpha_{\max}$.

2 cases to consider:

1. May be that α_{init} satisfies the Armijo condition $\Longrightarrow$ $\alpha_k = \alpha_{\text{init}}$.
2. Otherwise, must be a last linesearch iteration (the l-th) for which

$$\alpha^{(l)} > \alpha_{\max} \quad \Longrightarrow \quad \alpha_k \geq \alpha^{(l+1)} = \tau \alpha^{(l)} > \tau \alpha_{\max}$$

Combining these 2 cases gives required result.

GENERIC LINESEARCH METHOD

Given an initial guess x_0, let $k = 0$
Until convergence:
  Find a descent direction p_k at x_k
  Compute a stepsize α_k using a
    backtracking-Armijo linesearch along p_k
  Set $x_{k+1} = x_k + \alpha_k p_k$, and increase k by 1

GLOBAL CONVERGENCE THEOREM

Theorem 2.3. Suppose that $f \in C^1$ and that g is Lipschitz continuous on $\mathbb{R}^n$. Then, for the iterates generated by the Generic Linesearch Method,

either

$$g_l = 0 \;\text{ for some }\; l \geq 0$$

or

$$\lim_{k \to \infty} f_k = -\infty$$

or

$$\lim_{k \to \infty} \min\left(|p_k^T g_k|, |p_k^T g_k| / \|p_k\|_2\right) = 0.$$

PROOF OF THEOREM 2.3

Suppose that $g_k \neq 0$ for all k and that $\lim_{k\to\infty} f_k > -\infty$. Armijo $\Longrightarrow$

$$f_{k+1} - f_k \leq \alpha_k \beta p_k^T g_k$$

for all $k \Longrightarrow$ summing over first j iterations

$$f_{j+1} - f_0 \leq \sum_{k=0}^{j} \alpha_k \beta p_k^T g_k.$$

LHS bounded below by assumption $\Longrightarrow$ RHS bounded below. Sum composed of -ve terms $\Longrightarrow$

$$\lim_{k\to\infty} \alpha_k |p_k^T g_k| = 0$$

Let

$$\mathcal{K}_1 \stackrel{\text{def}}{=} \left\{ k \mid \alpha_{\text{init}} > \frac{2\tau(\beta-1) g_k^T p_k}{\gamma \|p_k\|_2^2} \right\} \;\; \& \;\; \mathcal{K}_2 \stackrel{\text{def}}{=} \{1, 2, \ldots\} \setminus \mathcal{K}_1$$

where γ is the assumed uniform Lipschitz constant.

For $k \in \mathcal{K}_1$,

$$\alpha_k \geq \frac{2\tau(\beta - 1) g_k^T p_k}{\gamma \|p_k\|_2^2}$$

$\implies$

$$\alpha_k p_k^T g_k \leq \frac{2\tau(\beta - 1)}{\gamma} \left(\frac{g_k^T p_k}{\|p_k\|} \right)^2 < 0$$

$\implies$

$$\lim_{k \in \mathcal{K}_1 \to \infty} \frac{|p_k^T g_k|}{\|p_k\|_2} = 0. \tag{1}$$

For $k \in \mathcal{K}_2$,

$$\alpha_k \geq \alpha_{\text{init}}$$

$\implies$

$$\lim_{k \in \mathcal{K}_2 \to \infty} |p_k^T g_k| = 0. \tag{2}$$

Combining (1) and (2) gives the required result.

EXAMPLES

Steepest-descent direction. $p_k = -g_k$

$$\lim_{k\to\infty} \min\left(|p_k^T g_k|, |p_k^T g_k|/\|p_k\|_2\right) = 0 \implies \lim_{k\to\infty} g_k = 0$$

Newton-like direction: $p_k = -B_k^{-1} g_k$

$$\lim_{k\to\infty} \min\left(|p_k^T g_k|, |p_k^T g_k|/\|p_k\|_2\right) = 0 \implies \lim_{k\to\infty} g_k = 0$$

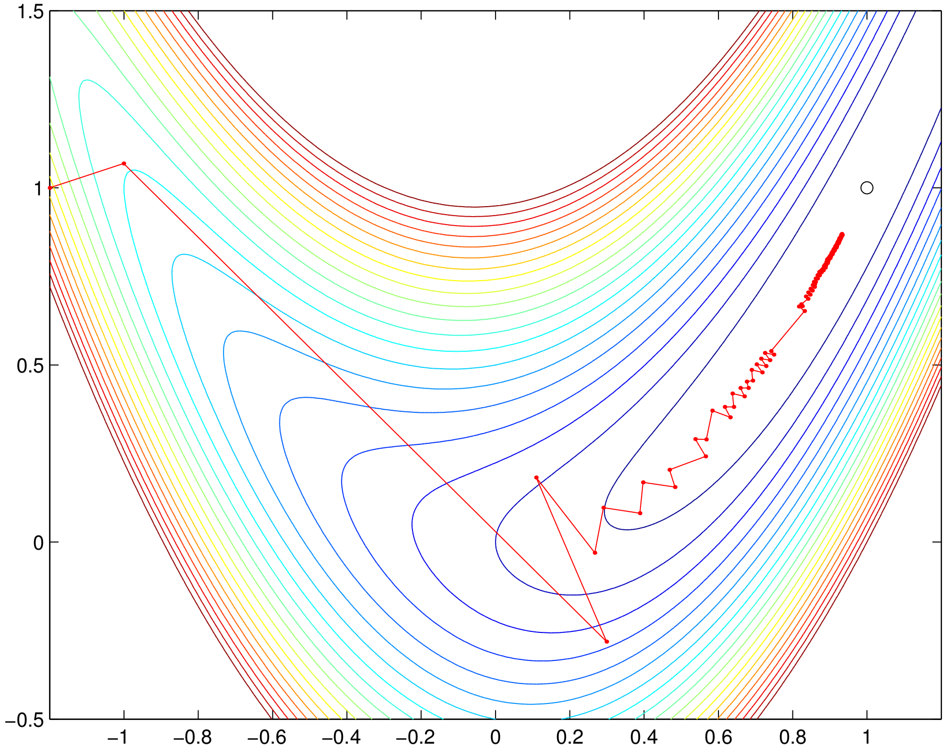
provided B_k is uniformly positive definite

Conjugate-gradient direction: p_k = any conjugate-gradient approximation to minimizer of $f_k + p^T g_k + \frac{1}{2} p^T B_k p \approx f(x_k + p)$

$$\lim_{k\to\infty} \min\left(|p_k^T g_k|, |p_k^T g_k|/\|p_k\|_2\right) = 0 \implies \lim_{k\to\infty} g_k = 0$$

provided B_k is uniformly positive definite

STEEPEST DESCENT EXAMPLE

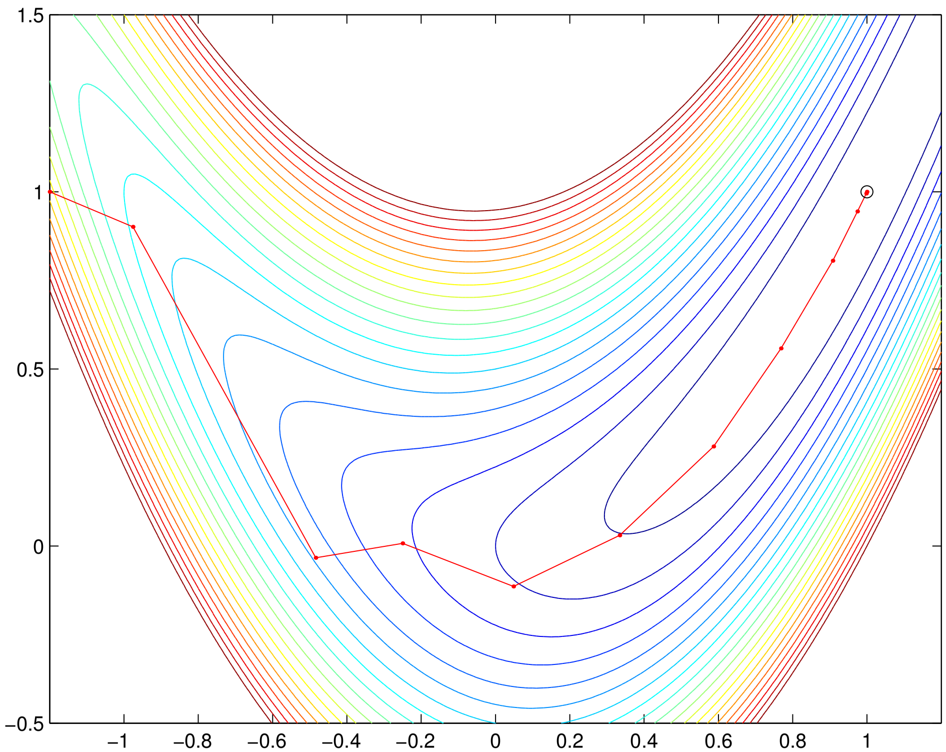


Contours for the objective function $f(x, y) = 10(y - x^2)^2 + (x - 1)^2$, and the iterates generated by the Generic Linesearch steepest-descent method

METHOD OF STEEPEST DESCENT (cont.)

- archetypical globally convergent method
- many other methods resort to steepest descent in bad cases
- not scale invariant
- convergence is usually very (very!) slow (linear)
- numerically often not convergent at all

NEWTON METHOD EXAMPLE



Contours for the objective function $f(x, y) = 10(y - x^2)^2 + (x - 1)^2$, and the iterates generated by the Generic Linesearch Newton method

MORE GENERAL DESCENT METHODS (cont.)

- ⊙ may be viewed as "scaled" steepest descent
- ⊙ convergence is often faster than steepest descent
- ⊙ can be made scale invariant for suitable B_k