Nonsmooth Optimization: Bundle Methods

Kaisa Joki

kjjoki@utu.fi

University of Turku

5.11.2013

Turun yliopisto
University of Turku

Outline

Turun yliopisto
University of Turku

Nonsmooth Optimization and Application Areas

- In nonsmooth optimization (NSO) functions don't need to be differentiable
- The general problem is that we are minimizing functions that are typically not differentiable at their minimizers
- This type of problems arise in many fields of applications
  - Economics
  - Mechanics
  - Engineering
  - Computational chemistry and biology
  - Optimal control
  - Data mining

Cause of Nonsmoothness

- **Inherent**: Original phenomenon contains various discontinuities and irregularities.
- **Technological**: Caused by some extra technological constraints which may cause a nonsmooth dependence between variables and functions.
- **Methodological**: Some algorithms for constrained optimization may lead to a nonsmooth problem (for example, the exact penalty function method).
- **Numerical**: So called "stiff problems" which are analytically smooth but numerically unstable and behave like nonsmooth problems.

Turun yliopisto
University of Turku

Difficulties Caused by Nonsmoothness

The gradient does not exist at every point so we

- can't utilize the classical theory of optimization because it requires certain differentiability and strong regularity assumptions
- can't use smooth (gradient based) methods because they may lead failure in convergence, in optimality test or in gradient approximation
- have difficulties defining a descent direction

Nonsmooth Optimization Problem

General problem

Lets consider a nonsmooth optimization problem of the form

$$\begin{cases} \min & f(\boldsymbol{x}) \\ \text{s. t.} & \boldsymbol{x} \in X, \end{cases}$$

where

- Set $X \subseteq \mathbb{R}^n$ is a set of feasible solutions
- Objective function $f : \mathbb{R}^n \to \mathbb{R}$ is
  - not required to have continuous derivatives
  - supposed to be locally Lipschitz continuous on the set X

In the following the objective function f is assumed to be convex.

Convex Analysis

Definition 1

Let function $f : \mathbb{R}^n \to \mathbb{R}$ be convex. The *subdifferential* of f at $\boldsymbol{x} \in \mathbb{R}^n$ is a set

$$\partial f(\boldsymbol{x}) = \left\{\boldsymbol{\xi} \in \mathbb{R}^n \mid f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \boldsymbol{\xi}^T(\boldsymbol{y} - \boldsymbol{x}) \text{ for all } \boldsymbol{y} \in \mathbb{R}^n\right\}.$$

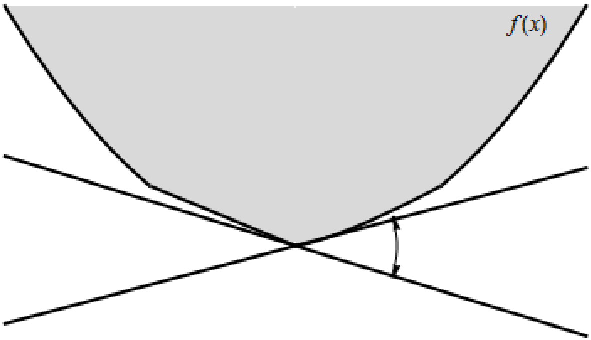
Each vector $\boldsymbol{\xi} \in \partial f(\boldsymbol{x})$ is called a *subgradient* of f at point $\boldsymbol{x}$.

The subdifferential $\partial f(\boldsymbol{x})$ is

- a nonempty, convex and compact set
- a generalization of a classical derivative because if $f : \mathbb{R}^n \to \mathbb{R}$ is convex and differentiable at $\boldsymbol{x} \in \mathbb{R}^n$, then $\partial f(\boldsymbol{x}) = \{\nabla f(\boldsymbol{x})\}$

Turun yliopisto
University of Turku

Convex Analysis



Subdifferential

Convex Analysis

Theorem 2

If $f : \mathbb{R}^n \to \mathbb{R}$ is convex then for all $\boldsymbol{y} \in \mathbb{R}^n$

$$f(\boldsymbol{y}) = \max\left\{ f(\boldsymbol{x}) + \boldsymbol{\xi}^T(\boldsymbol{y} - \boldsymbol{x}) \,|\, \boldsymbol{x} \in \mathbb{R}^n, \boldsymbol{\xi} \in \partial f(\boldsymbol{x}) \right\}.$$

Theorem 3

The direction $\boldsymbol{d} \in \mathbb{R}^n$ is a descent direction for a convex function $f : \mathbb{R}^n \to \mathbb{R}$ at $\boldsymbol{x} \in \mathbb{R}^n$ if

$$\boldsymbol{\xi}^T\boldsymbol{d} < 0 \text{ for all } \boldsymbol{\xi} \in \partial f(\boldsymbol{x}).$$

Turun yliopisto
University of Turku

Optimality Condition

For convex functions we have the following necessary and sufficient optimality condition:

Theorem 4

A convex function $f : \mathbb{R}^n \to \mathbb{R}$ attains its global minimum at point $\boldsymbol{x}$, if and only if

$$\mathbf{0} \in \partial f(\boldsymbol{x}).$$

Methods for Nonsmooth Optimization

The main problem: We usually don't know the whole subdifferential of the function but only one arbitrary subgradient at each point.

Different methods to solve a nonsmooth optimization problem

- Bundle Methods
- Derivative Free Methods
- Subgradient Methods
- Gradient Sampling Methods
- Hybrid Methods
- Special Methods

About Standard Bundle Method

- We consider an unconstrained convex nonsmooth problem

$$\begin{cases} \min & f(\boldsymbol{x}) \\ \text{s. t.} & \boldsymbol{x} \in \mathbb{R}^n \end{cases}$$

- Assumption: At every point $\boldsymbol{x} \in \mathbb{R}^n$ we can evaluate the value $f(\boldsymbol{x})$ and one arbitrary $\boldsymbol{\xi} \in \partial f(\boldsymbol{x})$
- Converges to the global minimum of f (if it exists)

Turun yliopisto
University of Turku

The main idea:

- Approximate the subdifferential of the objective function with a *bundle*
- Bundle consists of subgradients from previous iterations
- Subgradient information is used to construct a piecewise linear approximation to the objective function
- This approximation is used to determine a descent direction
- If approximation is not adequate then we add more information to the bundle

Bundle

- At iteration k in the current iteration point $\boldsymbol{x}_k$ our *bundle* is

$$\mathcal{B}_k = \{(\boldsymbol{y}_j, f(\boldsymbol{y}_j), \boldsymbol{\xi}_j) \mid j \in J_k\}.$$

  where
  - $\boldsymbol{y}_j \in \mathbb{R}^n$ is a trial point
  - $\boldsymbol{\xi}_j \in \partial f(\boldsymbol{y}_j)$ is a subgradient
  - J_k is a nonempty subset of $\{1, 2, \ldots, k\}$

- By using the bundle we can construct a *cutting plane model* which is a piecewise linear approximation of function f

Cutting Plane Model

- The cutting plane model is

$$\hat{f}_k(\boldsymbol{x}) = \max_{j \in J_k} \left\{ f(\boldsymbol{y}_j) + \boldsymbol{\xi}_j^T (\boldsymbol{x} - \boldsymbol{y}_j) \right\} \quad \text{for all } \boldsymbol{x} \in \mathbb{R}^n.$$

- It is a convex function and $\hat{f}_k(\boldsymbol{x}) \leq f(\boldsymbol{x})$.
- This approximation can be written in equivalent form

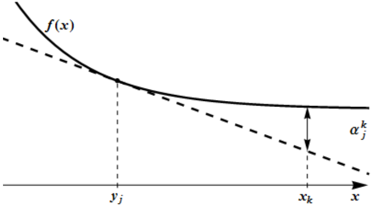
$$\hat{f}_k(\boldsymbol{x}) = \max_{j \in J_k} \left\{ f(\boldsymbol{x}_k) + \boldsymbol{\xi}_j^T (\boldsymbol{x} - \boldsymbol{x}_k) - \alpha_j^k \right\}$$

with the *linearization error*

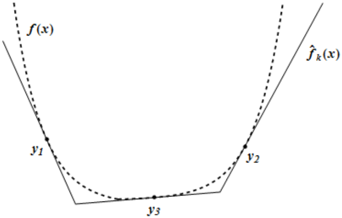
$$\alpha_j^k = f(\boldsymbol{x}_k) - f(\boldsymbol{y}_j) - \boldsymbol{\xi}_j^T (\boldsymbol{x}_k - \boldsymbol{y}_j) \geq 0 \quad \text{for all } j \in J_k.$$

Turun yliopisto
University of Turku

Cutting Plane Model



Linearization error



Cutting plane model

Algorithm: Direction Finding Problem

To determine the search direction $\boldsymbol{d}_k$ we need to solve

$$\min_{\boldsymbol{d}\in\mathbb{R}^n} \left\{ \hat{f}_k(\boldsymbol{x}_k + \boldsymbol{d}) + \frac{1}{2}\boldsymbol{d}^T \boldsymbol{M}_k \boldsymbol{d} \right\}$$

where

- $\boldsymbol{M}_k$ is a positive definite and symmetric $n \times n$ matrix.
- $\frac{1}{2}\boldsymbol{d}^T \boldsymbol{M}_k \boldsymbol{d}$ is a stabilizing term which
  - guarantees existence of the unique solution $\boldsymbol{d}_k$
  - keeps approximation local enough

The search direction $\boldsymbol{d}_k$ is also a descent direction to the original objective

Algorithm: Quadratic Direction Finding Problem

$$\min_{\boldsymbol{d}\in\mathbb{R}^n} \left\{ \max_{j\in J_k} \left\{ f(\boldsymbol{x}_k) + \boldsymbol{\xi}_j^T \boldsymbol{d} - \alpha_j^k \right\} + \frac{1}{2} \boldsymbol{d}^T \boldsymbol{M}_k \boldsymbol{d} \right\} \tag{1}$$

- The problem (1) can be rewritten as a *smooth quadratic direction finding problem*

$$\begin{cases} \min & v + \frac{1}{2} \boldsymbol{d}^T \boldsymbol{M}_k \boldsymbol{d} \\ \text{s. t.} & \boldsymbol{\xi}_j^T \boldsymbol{d} - \alpha_j^k \le v \quad \forall j \in J_k, \\ & v \in \mathbb{R},\ \boldsymbol{d} \in \mathbb{R}^n \end{cases} \tag{2}$$

Turun yliopisto
University of Turku

Algorithm: Search Direction and Stopping Condition

- The solution $(v_k, \boldsymbol{d}_k)$ of (2) can also be calculated from the dual problem
- The next iteration candidate is $\boldsymbol{y}_{k+1} = \boldsymbol{x}_k + \boldsymbol{d}_k$
- Value

$$v_k = \hat{f}_k(\boldsymbol{y}_{k+1}) - f(\boldsymbol{x}_k)$$

  is the predicted descent of f at $\boldsymbol{y}_{k+1}$
- If $v_k = 0$ then the current point $\boldsymbol{x}_k$ is the global minimum
- It is convenient to stop algorithm when $-v_k \leq \varepsilon$ where $\varepsilon > 0$ is a final accuracy tolerance

Turun yliopisto
University of Turku

Algorithm: Serious and Null Step

- Now we perform either a serious step or a null step
- A *serious step*

  $$\boldsymbol{x}_{k+1} = \boldsymbol{y}_{k+1}$$

  is performed if

  $$f(\boldsymbol{y}_{k+1}) - f(\boldsymbol{x}_k) \leq m v_k$$

  where $m \in (0, 1/2)$ is a line search parameter.
- Otherwise we make a *null step*

  $$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k$$

Turun yliopisto University of Turku

Algorithm: Updating the Bundle

- In both steps we improve the approximation by adding

$$(\boldsymbol{y}_{k+1}, f(\boldsymbol{y}_{k+1}), \boldsymbol{\xi}_{k+1})$$

  into the bundle where $\boldsymbol{\xi}_{k+1} \in \partial f(\boldsymbol{y}_{k+1})$
- Easiest way to do this is to set $J_{k+1} = J_k \cup \{k+1\}$
  - stores all subgradients
  - causes difficulties with storage and computation
- By using *the subgradient aggregation strategy* we can keep the size of the bundle bounded

Turun yliopisto
University of Turku

Nonconvex Bundle Methods

- Objective function is only supposed to be locally Lipschitz continuous
- Cannot guarantee even local optimality of a solution without some convexity assumption
- Not as efficient methods as convex bundle methods
- Best nonconvex bundle methods are generalizations of convex bundle methods with suitable modifications
- Not developed from the nonconvex perspective

Difference of two Convex functions

Definition 5

A function $f : \mathbb{R}^n \to \mathbb{R}$ is called a *DC function* if it can be written in the form

$$f(\boldsymbol{x}) = f_1(\boldsymbol{x}) - f_2(\boldsymbol{x})$$

where f_1 and f_2 are convex functions on $\mathbb{R}^n$.

- If a DC function f is nonsmooth then at least one of the functions f_1 and f_2 is nonsmooth
- For DC functions it is still possible to utilize convex analysis and convex optimization theory to some extent
- Many problems of nonconvex optimization can be described by using DC functions

Future Work

Only a few efficient nonconvex nonsmooth optimization methods exist so my goal is

- Develop a local bundle method for unconstrained nonconvex nonsmooth problems where the objective is a DC function
- Add good features of gradient sampling methods to possibly get a better method
- Extend the method to constrained case
- Modify the local method so that we get a global solution both in unconstrained and constrained case

References

[1] Bagirov, A.M. and Ugon, J.: *Codifferential method for minimizing nonsmooth DC functions*. Journal of Global Optimization, Vol. 50(1), 2011, pages 3–22.

[2] Haarala, M.: *Large-Scale Nonsmooth Optimization: Variable metric bundle method with limited memory*. Doctoral Thesis, University of Jyväskylä, 2004.

[3] Mäkelä, M.M.: *Survey of bundle methods for nonsmooth optimization*. Optimization Methods and Software, Vol. 17(1), 2002, pages 1–29.

[4] Mäkelä, M.M. and Neittaanmäki, P.: *Nonsmooth Optimization: Analysis and Algorithms with Applications to Optimal Control*. World Scientific Publishing Co., Singapore, 1992.

Thank you for your attention!