Constrained Scaled Conjugate Gradient Based Direct Search

G.Chandra Mouli
(Email: gcmouli1@gmail.com)

Supervisor: Prof. Vishnu Narayanan
(Email: vishnu@iitb.ac.in)

IEOR, IIT Bombay

December 16, 2018

ORSI-2018, Mumbai

Overview

Definition

Consider the optimization problem of form

$$\min_{x \in \Omega} f(x),$$

where $\Omega \subset \mathbb{R}^n$ is the feasible region and $f : \Omega \subset \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$.
We define such a problem as Derivative Free Optimization (DFO) if:

- Nature of f is unknown and is generally computed through a computer simulation.
- Derivative information about f is unknown.
- Sometimes knowledge about constraints defining Ω is also unknown.

Growing Importance of DFO

- Many real world optimization problems are quite complex in nature. They lack gradient information and are often plagued by noise.
- Automatic differentiation is not possible with many commercial software as they don't provide source code.
- Technological advancements in computer hardware and increasing sophistication in software leads to new opportunities for optimization.

Applications of DFO

Some examples which are modelled as DFO [2] are:

- Parameter tuning of algorithms e.g. tuning parameters in nonlinear optimization, MINLP solvers etc.
- Error analysis of stability and accuracy for a numerical computation e.g. estimation of matrix condition number, analysis of numerical stability for fast matrix inversion.
- Design of various engineering problems for e.g. design of rotor blade of helicopter, aeroacoustic shape design and hydrodynamic design.

Expectations from a DFO Algorithm

Following properties are desired from a DFO algorithm:

- Should be able to solve smooth and nonsmooth problems.
- Should have good convergence properties.
- Since most DFO problems are quite expensive to compute, the algorithm should achieve good results in a very limited budget.
- Should be scalable to high dimensional (>100) problems.
- A low computational time (excluding function evaluation time) is desired.

Algorithms for DFO

Algorithms for DFO can be broadly classified into following three approaches:

- Direct Search Methods (Nelder-Mead Simplex, Hooke-Jeeves method, Generalized Pattern Search and Mesh Adaptive Direction Search etc).
- Model based Methods or Surrogate based Methods (NEWUOA, ORBIT etc).
- A combination of above two approaches (NOMAD, SID-PSM etc).

Structure of Direct Search Algorithm

Each iteration comprise of two steps.

- Search
  - Flexible step.
  - Doesn't affect convergence directly.
  - Good implementation can greatly enhance the speed of algorithm.
- Poll
  - More rigid step.
  - Ensures theoretical convergence.
  - Search local vicinity of current incumbent solution.

The algorithm first calls the search step and creates trial points. If search fails, it goes to poll step.

Search Step

- Generation of only finite number of trial points is allowed.
- Searches the feasible space in global sense.
- Flexible step. Problem specific search procedure can be implemented.
- Allows usage of generic search strategies like speculative search, VNS, LH, and particle swarm etc.

Acceptance Condition

Let x_k be the current incumbent solution. Then the new point x_{k+1} is accepted only if [2, 3]

$$f(x_{k+1}) < f(x_k) - \rho(\Delta_k),$$

where Δ_k is the step length and $\rho : \mathbb{R}_+ \to \mathbb{R}_+$ is a continuous function which satisfies $\lim\limits_{y \to 0^+} \frac{\rho(y)}{y} = 0$ and $\rho(y_1) \leq \rho(y_2)$ if $y_1 < y_2$.

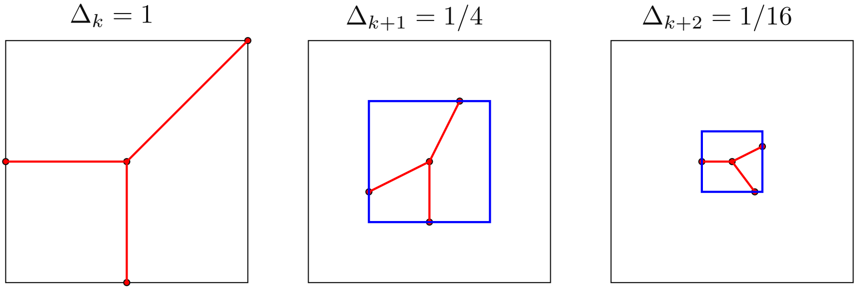
Poll Step

Poll Set

$$P_k = \{x_k + \Delta_k d \ : \ d \in D_k\}$$

- Poll Directions D_k [1] A distinct positive spanning set (different from previous iteration) is constructed from nonnegative integer combination of directions of D (set of positive bases).
- Evaluation of function at points in P_k.
- Determine whether iteration was successful or not.
- If successful we update Δ_k as:
  - $\Delta_{k+1} \to 4\Delta_k$.
- If unsuccessful:
  - $\Delta_{k+1} \to \Delta_k/4$.
- With increasing iterations, the set of poll directions, being normalized, grow dense in unit sphere. Thus any direction in $\mathbb{R}^n$ can be approximated with arbitrary precision.

Illustration of Poll Step



Poll steps during failure and corresponding shrinks

Motivation

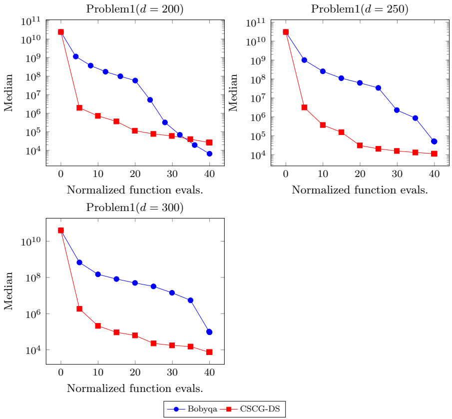
- Direct search methods are robust but slow.
- Model based methods are very effective on smooth problems as they tend to utilize underlying structural properties of the given function.
- An approach which is effective as well as robust, is desired.
- Gives good solution within a low computational budget.
- Can handle large dimensional problems.

Our Approach

- Comprises of poll step and search step.
- We generate $n+1$ uniform angled directions and corresponding trial points across a given point.
- After every poll step, a quadratic model is fitted across the trial points and its minimum is computed (only if model is positive definite).
- If model fitting step is successful, algorithm restarts from the quadratic minima.
- We evaluate simplex gradient about current point using above trial points.
- Apply scaled conjugate gradient method using simplex gradient.

Results on Chained Rosenbrock Function [4]

$$f(x) = \sum_{i=2}^{n} [100(x_{i-1}^2 - x_i)^2 + (x_{i-1} - 1)^2]$$



Progress Curve for Chained Rosenbrock Function

Future Work

- Extend to handle general constraints.
- Add ability to handle discrete and categorical variables.
- Add more model search approaches like RBF etc.
- Extend to global optimization.

Software

- Written completely in C.
- Dependencies: Lapack and Blas libraries.
- Works on linux based machines only.
- One can download my code from https://github.com/gcmouli1/CSCG-DS
- Instructions for its usage are given in the website.
- Email-id of author: gcmouli1@gmail.com

References

[1] Charles Audet and John E Dennis Jr. Mesh adaptive direct search algorithms for constrained optimization. *SIAM Journal on optimization*, 17(1):188–217, 2006.

[2] Andrew R Conn, Katya Scheinberg, and Luis N Vicente. *Introduction to derivative-free optimization*, volume 8. Siam, 2009.

[3] Giovanni Fasano, Giampaolo Liuzzi, Stefano Lucidi, and Francesco Rinaldi. A linesearch-based derivative-free approach for nonsmooth constrained optimization. *SIAM Journal on Optimization*, 24(3):959–992, 2014.

[4] Ladislav Lukšan and Jan Vlcek. Test problems for unconstrained optimization. *Technical Report*, 897, 2018.