PRINCIPLES OF NOISY NONLINEAR OPTIMIZATION

Jorge Nocedal

with Yuchen Luo and Shigeng Sun

Huatulco, Jan 2023

Northwestern

Introduction

50 years of research in deterministic nonlinear optimization
Adopted in wide range of applications
Can be quite complex for constrained problems:
IPOPT, KNITRO, SNOPT, MINOS,…

Growing interest in stochastic optimization problems.
Noise or computational error

$$\tilde{f}(x) = f(x) + \epsilon$$

Central Question:

- should existing methods be drastically redesigned to be robust to noise?

or

- do relatively small changes suffice?

Main Thesis

- Can design effective methods by preserving underlying properties of current methods
- Making judicious modifications following

Three Design Principles

Based on the observation: the only operations that lead to difficulties are:

1. Comparisons of noisy function values
2. Computation of differences of noisy function values
3. Computation of differences of noisy gradients

In addition, robust stop tests can be difficult in the noisy settings.
Relevant to methods including inner and outer iterations. (Dezfulian, Waechter)

We do not argue…

That the only way to design nonlinear optimization methods that are robust to noise is to adapt existing deterministic methods

May be preferable to design methods from scratch following new ideas

But the sophistication of many methods makes it alluring to build upon their foundations as much as possible.

Example: Inequality Constrained Problems:

1. If a good estimate of active set is known, it is attractive to use an active-set approach (SQP)

2. Interior point methods very effective for large problems with network structure. Hope to retain this strength in the noisy setting

.

Main goal of this talk

- Discuss the three design principles
- Review recent research on how to implement them in practice
- Illustrate via a case study involving engineering design

N.B. Argue that we need an estimate of the noise to guarantee a reasonable solution

Before doing this:
- What do we mean by noise?
- What are realistic applications?

$$\tilde{f}(x) = f(x) + \epsilon$$

Some References

Curtis, Robinson, Roger, et al. (constrained setting)
Scheinberg, Paquette, et al. (unconstrained)
Berahas and Northwestern team (unconstrained, constrained)
Bollapragada (dynamic sampling, unconstrained, constrained)

Before that:
More' and Wilde

Before that:
Polyak (robust control)

Noise

Computational error:

- Roundoff, Mixed Precision
- Deterministic, repeated evaluations give same results

$$\tilde{f}(x) = f(x) + \epsilon$$

- Computational error arises in scientific computing
- Inexact linear solves, adaptive integration schemes
- More'-Wild

Stochastic noise

- Monte Carlo simulation, etc

We assume that noise is persistent and that it cannot be controlled or diminished in any way

Acceptable solutions

Given a noise level, we can define acceptable solutions (neighborhoods)

- Can algorithms compute them?
- What information about the noise is needed?
- In the algorithms discussed today noise estimation is an integral part of the iteration

Having said all of this, do our codes really fail when we inject noise?

Failure of classical methods

- Design optimization problem involving PDEs, Willcox et al
- Some physical parameters are uncertain, Monte Carlo
- Standard optimization packages failed
- Resort to a derivative free optimization code (Powell's BOBYQA)
  - Why? For another talk…

$$\min f(x)$$
$$\text{s.t. } c(x) \leq 0$$

Moving forward

- Noisy finite differences are enough to cause failure
- Prefer: interior point, augmented Lagrangian, etc?

Principle I: robust comparisons

Comparisons are performed when gauging progress in a line search or trust region approach, both for unconstrained and constrained problems (objective or penalty function).

Claim: We can retain the logic of the algorithms

$$\tilde{f}(x_{k+1}) < \tilde{f}(x_k)$$

Comparisons should be:

- relaxed based on noise level, or
- should be avoided altogether (no need for noise estimate)

Option A: Avoiding the line search

Reasons for avoiding a line search

- Measuring progress with some confidence we may be too expensive.

$$\tilde{f}(x_{k+1}) < \tilde{f}(x_k) \quad \text{vs } f(x_{k+1}) - f(x_k)$$

- If search direction is very noisy and poorly scaled, it is unproduction to try control the length of each step; better to rely on expected behavior

- Forcing sample consistency not useful in the very noisy regime

Steplength can control noise and displacement simultaneously

Avoiding the line search: SGD

Hallmark of Neural Network
Perceptron algorithm, Rummelhart, LeCun, etc. Bertsekas

Responsibility falls on tuning or steplength rule

- Predetermined diminishing steplength $\alpha_k = O(1/k)$
- Adaptive/manual step-wise reduction (current practice)
- Fixed steplength

For stochastic problem: $\min F(w) \equiv \mathbb{E}[f(w;\xi)]$

$$\mathbb{E}[F(w_{k+1}) - F(w_k)] \leq -\alpha_k \|\nabla F(w_k)\|_2^2 + \alpha_k^2 \mathbb{E}\|\nabla f(w_k, \xi_k)\|^2$$

Option B: Performing a line search

If line search is performed, safeguard sufficient decrease condition (Berahas)

$$f(x_k + \alpha_k d_k) \leq f(x_k) + c_1 \alpha_k \nabla f(x_k)^T d_k + \epsilon_A$$

Will never fail if $\epsilon_A = 2 \max \epsilon_f$

- Guarantee convergence by setting $\epsilon_A = 2\epsilon_f$
- If noise is not bounded, set 2 times std deviation
- Not provably convergent, one can expect it in practice. Scheinberg et al.

Interested in preserving line searches; common in nonlinear optimization algorithms

Trust region method

Create a model of the objective

$$m_k(d) = \tilde{f}(x_k) + \nabla \tilde{f}(x_k)^T d + \frac{1}{2} d^T B d$$

Accept the step and update trust region according to ratio

$$\frac{f(x_k) - f(x_k + d_k) + \epsilon}{m(0) - m(d_k) + \epsilon}$$

Can establish convergence to a neighborhood

Sun and Nocedal 2021

Noise-tolerant first-order line search method

Problem: $\min f(x)$ Observe: $\tilde{f}(x)$; stochastic approx: $\tilde{g}_k$

1. Compute $\tilde{g}_k$
2. $p_k = -\tilde{g}_k$
2. Find α_k such that

$$\tilde{f}(x_k + \alpha p_k) \leq \tilde{f}(x_k) + c_0 \alpha_k \tilde{g}_k^T p + 2\epsilon_f$$

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

Algorithm can be run repeatedly for smaller values of ϵ_f

Bounded Errors Assumption

Assume bounded errors (noise) for simplicity

$$|\tilde{f}(x) - f(x)| \le \epsilon_f \qquad \|\tilde{c}(x) - c(x)\|_1 \le \epsilon_c$$

$$\|\tilde{g}(x) - g(x)\| \le \epsilon_g \qquad \|\tilde{J}(x) - J(x)\|_{1,2} \le \epsilon_J$$

$$\min\ f(x) \quad \text{s.t.} \quad c(x) = 0$$

A convergence result

Before the iterates enter the region where errors dominate, true function values converge at an R-linear rate to a neighborhood of the solution

Theorem. Let

$$N = \{\, x : \| \nabla \phi(x) \| \le \max\{ A \frac{\sqrt{M \epsilon_f}}{\beta}, B \frac{\epsilon_g}{\beta} \} \,\}$$

Let K be the first iterate that enters N. Then for all $k < K$

$$\phi(x_k) - \phi(x_*) \le \rho[\phi(x_0) - \phi(x_*)] + 2\epsilon_f$$

$$\phi(x) \leftarrow f(x)$$

Berahas et al
Oztoprak

if $(\epsilon_f, \epsilon_g) > 0$, then K is finite

If iterate enters N all subsequent iterates cannot stray too far

Solving practical problems

How to compute gradient approximations? $\tilde{g}_k$
(Noisy) Automatic Differentiation

→ Noise-aware finite differences More'-Wild (2002)

Principles of Noisy Optimization. Part II: Function Differences

- Comparisons of function values
- Differences of Functions or gradient values: noise-aware derivative estimation

$$\frac{\tilde{f}(x_k + h) - \tilde{f}(x_k)}{h} \qquad h = 8^{1/4}\sqrt{\frac{\epsilon}{L}} \qquad L = \max_I |\phi''(x)|$$

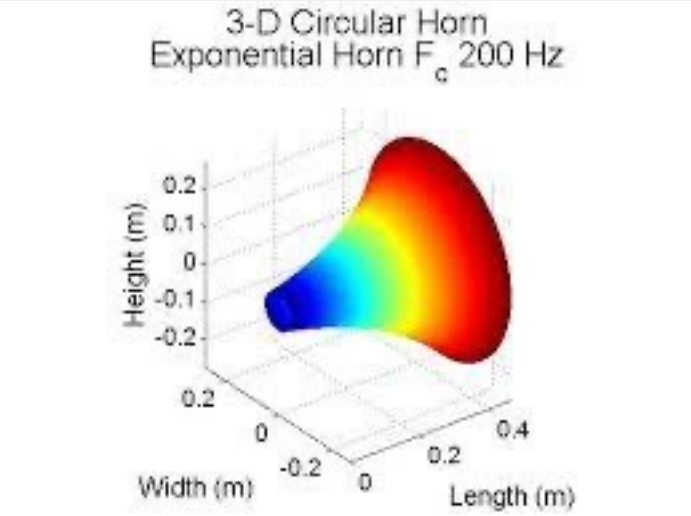
Adaptive estimation of *L* is important — Shi, Xie, Xuan 2022

Need a reasonable estimate of noise level

Complexity guarantees for Gaussian directions — Nesterov, Spokoiny

We can now tackle a practical problem…

Acoustic design

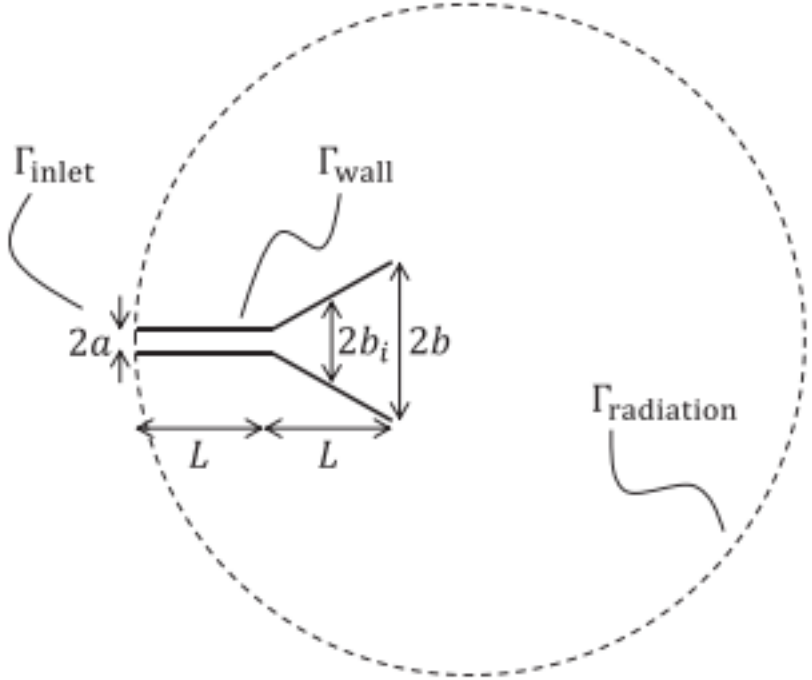


An incoming wave enters horn through inlet; exits the outlet into exterior domain with an absorbing boundary

Goal: optimize efficiency

Some of the properties of the metal are unknown

High fidelity model is a finite-element model of the Helmholtz equation leading to system of 39,895 equations and unknowns. This systems gives pressures which are then used to compute the reflection coefficient



Design variables: $b_1, b_2, \ldots, b_6$
uncertain parameters:
impedances, wave numbers

Figure 3

The uncertain model parameters are given as

ple.

Random variable	Distribution	Lower bound	Upper bound	Mean	Standard Deviation
$k(\omega)$	Uniform	1.3	1.5	–	–
$z_u(\omega)$	Normal	–	–	50	3
$z_l(\omega)$	Normal	–	–	50	3

$z_{upper}(\omega)$: upper horn wall impedance

Formulation

The optimization problem

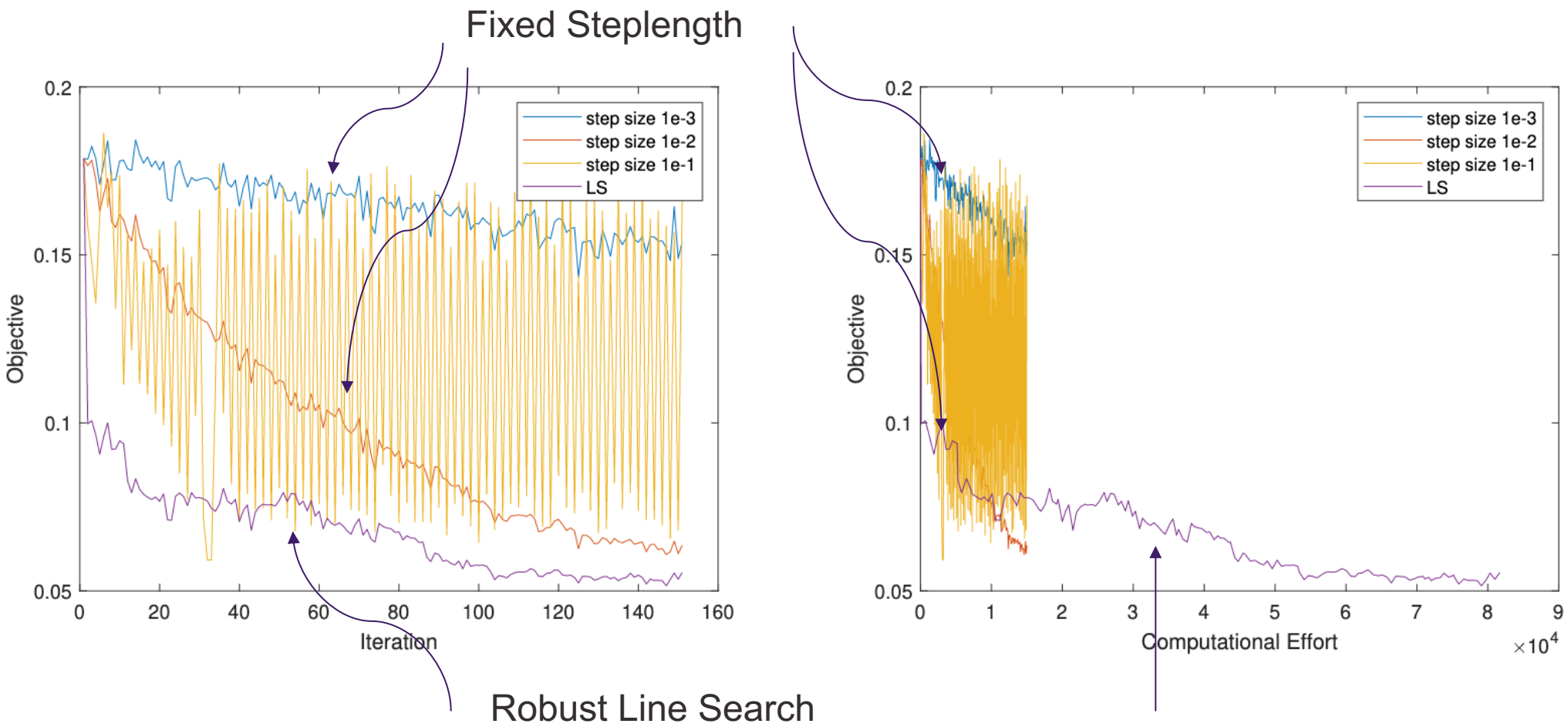
$$\min_{b_L \le b \le b_u} f(b) = \mathbb{E}[s(b,\omega))] + 3\sqrt{var[s(b,\omega)]}$$

Bound constrained stochastic nonlinear optimization problem
Use gradient projection method with relaxed line search
Using noise-aware finite difference approximations to gradient

- Estimate noise level via sampling
- Fairly constant throughout optimization

Solution of acoustic design problem

Sample size = 100



Conclusion: feature to be added to codes

Add module for predetermined steplength selection rule

Interesting alternative: step search technique
Supported by probabilistic convergence theory

Scheinberg et al. 2022

Deterministic variant of horn problem

The optimization problem

$$\min_{b_L \le b \le b_u} f(b) = \mathbb{E}[s(b,\omega))] + 3\sqrt{var[s(b,\omega)]} \quad \Rightarrow \quad \min_{b_l \le b \le \le b_u} s(b)$$

Deterministic variant of horn problem

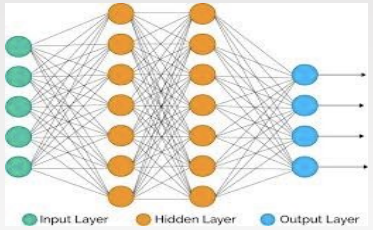
$$\min_{b_l \leq b \leq \leq b_u} s(b)$$

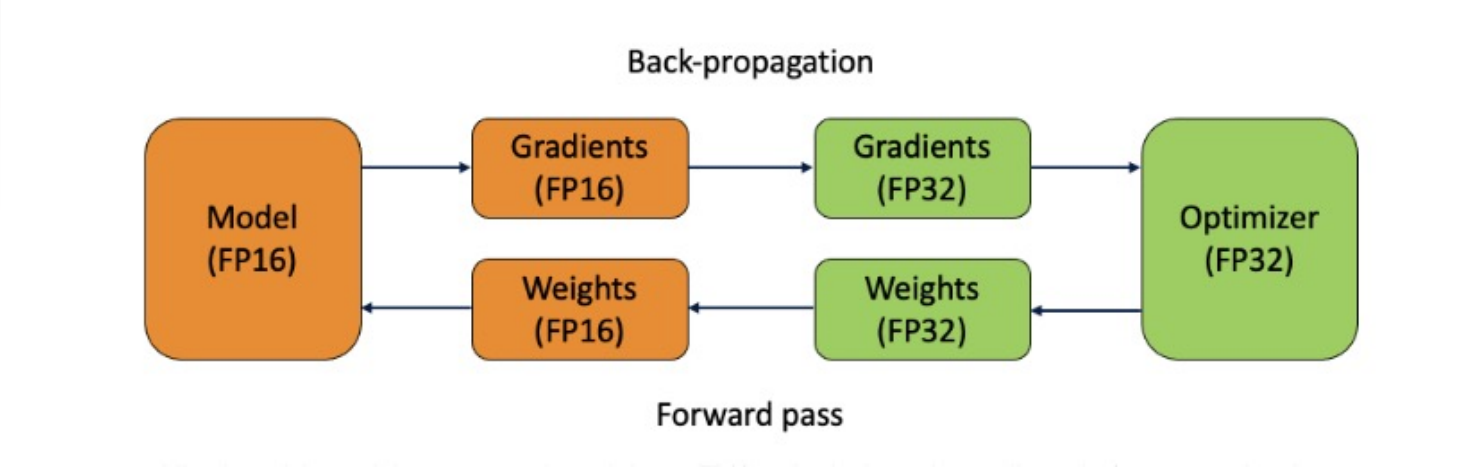
With analytic derivatives computed in lower precision arithmetic

- Example of mixed precision arithmetic promoted in deep learning
- Speedups, memory and energy savings
- Computational noise
- Lower precision noise: multiplicative noise $x(1 + \epsilon_{mach})$

Mixed Precision Arithmetic in Deep Learning

Our optimization packages written in FP64
Training neural networks. Inference.
Weight update: FP32
Forward and Back-propagation FP16



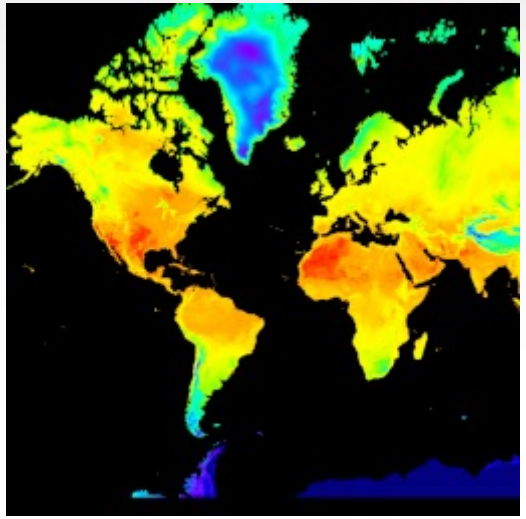



```
opt = tf.train.AdamOptimizer()
opt = tf.train.experimental.enable_mixed_precision_graph_rewrite(opt)
```

Mixed Precision in Data Assimilation

Evolution can be described by the Navier-Stokes equations.

$$x_i = M_i(x_{i-1}), \quad i = 1, 2, \ldots, N; \qquad x_0 = \text{given}.$$



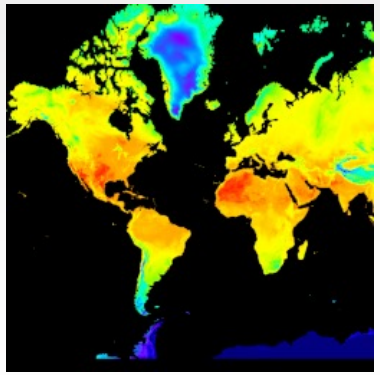
However,

- Initial atmospheric state only partially known,
- Estimate using an optimization algorithm
- Maximize goodness of fit between the simulated states and actual observations in assimilation window.

Optimal initial state used to produce a 10-15 day weather forecast.

Data assimilation model

- Using lower precision throughout gives reasonable results
- More promising: mixed precision
- Gradient computation (adjoint) in lower precision
- Gradients (Jacobian) already computed using a lower fidelity model



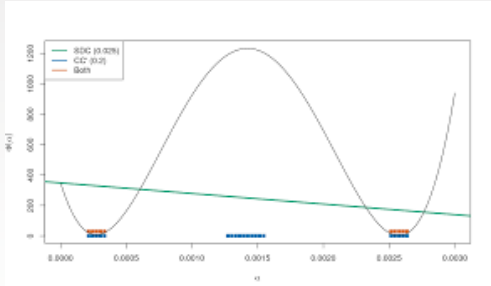
$$J(x_0, x) = 1/2(x_0 - x^b)^T B^{-1}(x_0 - x^b) + 1/2 \sum_{i=0}^{N} (x_i - y_i)^T R_i^{-1}(x_i - y_i),$$

x^b = given background state,

B and the R_i are error covariance matrices and

Principles of Noisy Optimization. Part 3:

Differences in gradients: performing quasi-Newton updates



Armijo-Wolfe line search

- BFGS and L-BFGS widely used for unconstrained and constrained problems

$$x_{k+1} = x_k - \alpha H_k \nabla \tilde{f}(x_k)$$

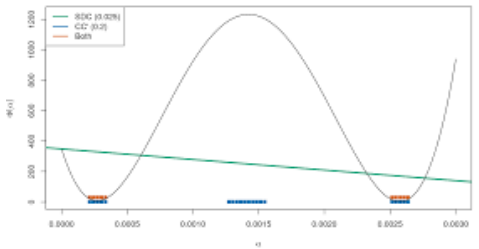
- Work in conjunction with line search yields convex approximation

$$y_k = \nabla f(x_{k+1}) - \nabla f(x_k) \quad s_k = x_{k+1} - x_k \quad s_k^T y_k > 0$$

Armijo-Wolfe line search

$$f(x_k + \alpha p) \le f(x_k) + \alpha c_1 g(x_k)^T p$$ Armijo

$$g(x_k + \alpha p)^T p \ge c_2 g(x_k)^T p$$ Wolfe



Armijo-Wolfe line search

Quasi-Newton methods and noise

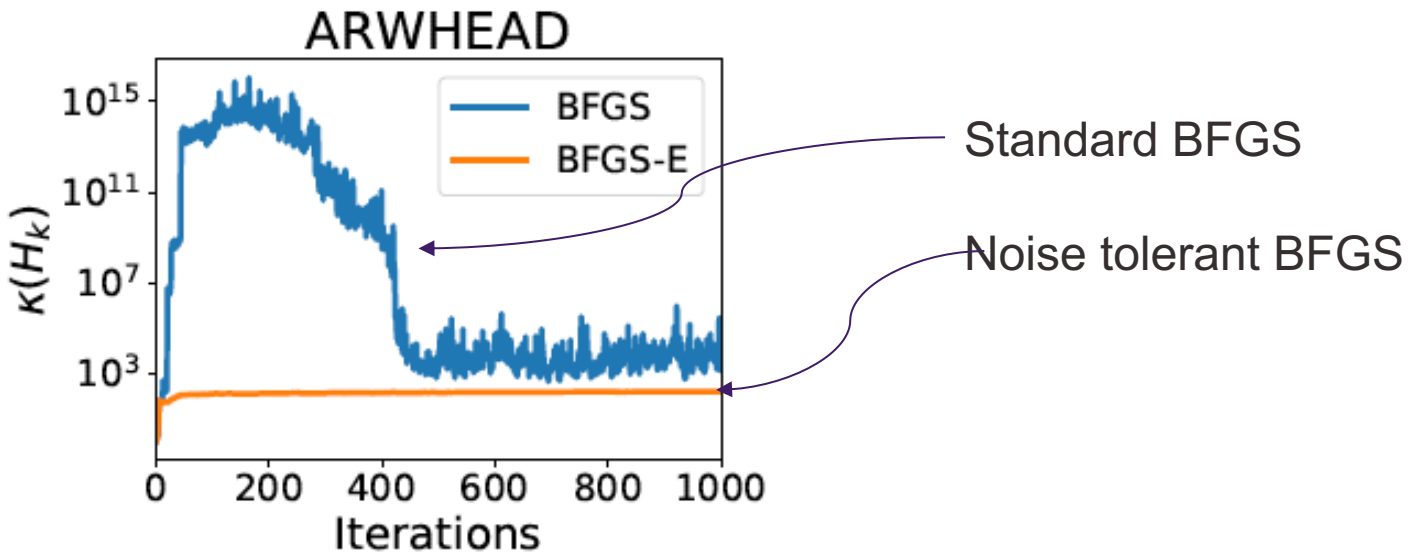
- Break down with noise
- Gradient differences corrupted

$$x_{k+1} = x_k - \alpha H_k \nabla \tilde{f}(x_k)$$

$$y_k = \nabla \tilde{f}(x_{k+1}) - \nabla \tilde{f}(x_k) \quad s_k = x_{k+1} - x_k \quad s_k^T y_k > 0$$

- Needs reliable curvature estimates H_k
- We propose a way to achieve this

Condition number of Hessian approximation



After entering regime where noise dominates new Hessian approximations stable

Robust BFGS update

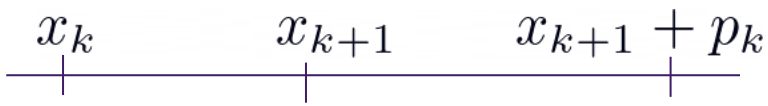
1. Compute step as usual $x_{k+1} = x_k - \alpha H_k \nabla \tilde{f}(x_k)$

2. measure curvature over a sufficiently large interval; lengthen

$$y_k = \nabla \tilde{f}(x_{k+1} + p_k) - \nabla \tilde{f}(x_k) \qquad s_k = [x_{k+1} + p_k] - x_k$$

3. Hessian update

$$H_{k+1} = (I - \rho s_k y_k^T) H_k (I - \rho y_k s_k^T) + \rho s_k s_k^T \qquad \rho = 1/y_k^T s_k$$



Sufficiently long interval

$$\ell = O(\epsilon_g / m)$$

$$\epsilon_g = \text{ error in gradient}$$
$$m = \text{convexity parameter}$$

$$\text{If } \|x_{k+1} - x_k\| \geq \ell \text{ continue}$$

$$\text{Else } y_k = \nabla \tilde{f}(x_k + \delta) - \nabla \tilde{f}(x_k) \text{ with } \delta = \ell p_k / \|p_k\|$$

- Knowledge of *m* not needed
- Can be estimated adaptively

Convergence Theory

Classical convergence theory (Dennis-More', Powell, Byrd,…,)

Analysis is complex

- Step affects Hessian approx. and vice versa.
- Line search essential role
- Bounding condition number of H_k not possible without first proving convergence

$$x_{k+1} = x_k - \alpha H_k \nabla \tilde{f}(x_k)$$

Use fundamental result about BFGS updating

Byrd-Nocedal 1989

- as long as curvature estimates are reliable…
- a large fraction of all steps are strongly descent directions

A fundamental result of BFGS updating

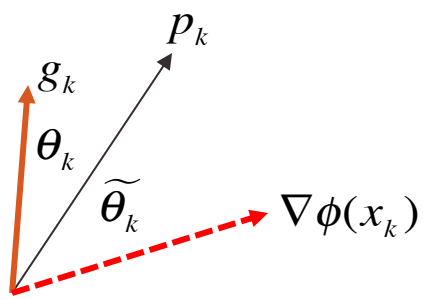
Theorem. [Byrd, N. (1989)] Let $H_0 > 0$ and $H_k = B_k^{-1}$ generated by BFGS updating using any pairs (s_k, y_k) s.t.

$$\frac{y_k^T s_k}{s_k^T s_k} \geq \hat{m} \qquad \frac{y_k^T y_k}{y_k^T s_k} \leq \hat{M} \quad \forall k \qquad (*)$$

Fix $q \in (0,1)$ (say $q = 0.9$). Define $\cos\theta_k$ angle between s_k and $B_k s_k$

Then a fixed fraction (say 0.9) of search directions make an acute angle
With the steepest descent direction

$$\theta_k = \angle(-p_k, g_k), \qquad \widetilde{\theta}_k = \angle(-p_k, \nabla\phi(x_k))$$

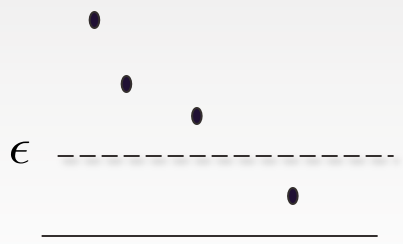


New Convergence theory: noisy case

Yuchen Xie

Identify a region where noise is not dominant and show

- Existence of steplengths (conditional)
- Same steplength works for true objective
- Lengthening guarantees stable updating
- Existence of good iterates: noisy case
- Function decrease for good iterates



Byrd, Xie, N. 2019

Linear Convergence

Before the iterates enter the region where errors dominate, true function values converge at an R-linear rate to a neighborhood of the solution

Theorem. Let

$$N = \{\, x : \| \nabla \phi(x) \| \le \max\{ A \frac{\sqrt{M \epsilon_f}}{\beta}, B \frac{\epsilon_g}{\beta} \} \,\}$$

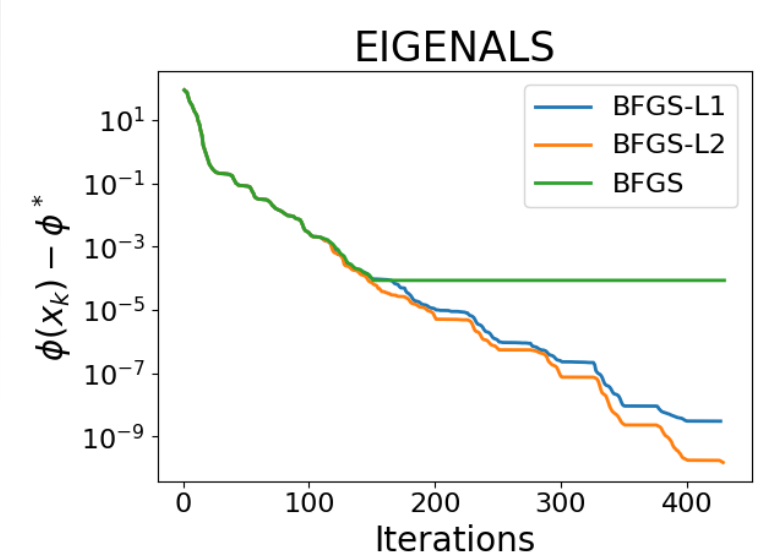
Let K be the first iterate that enters N. Then for all $k < K$

$$\phi(x_k) - \phi(x_*) \le \rho[\phi(x_0) - \phi(x_*)] + 2\epsilon_f$$

Other Results:

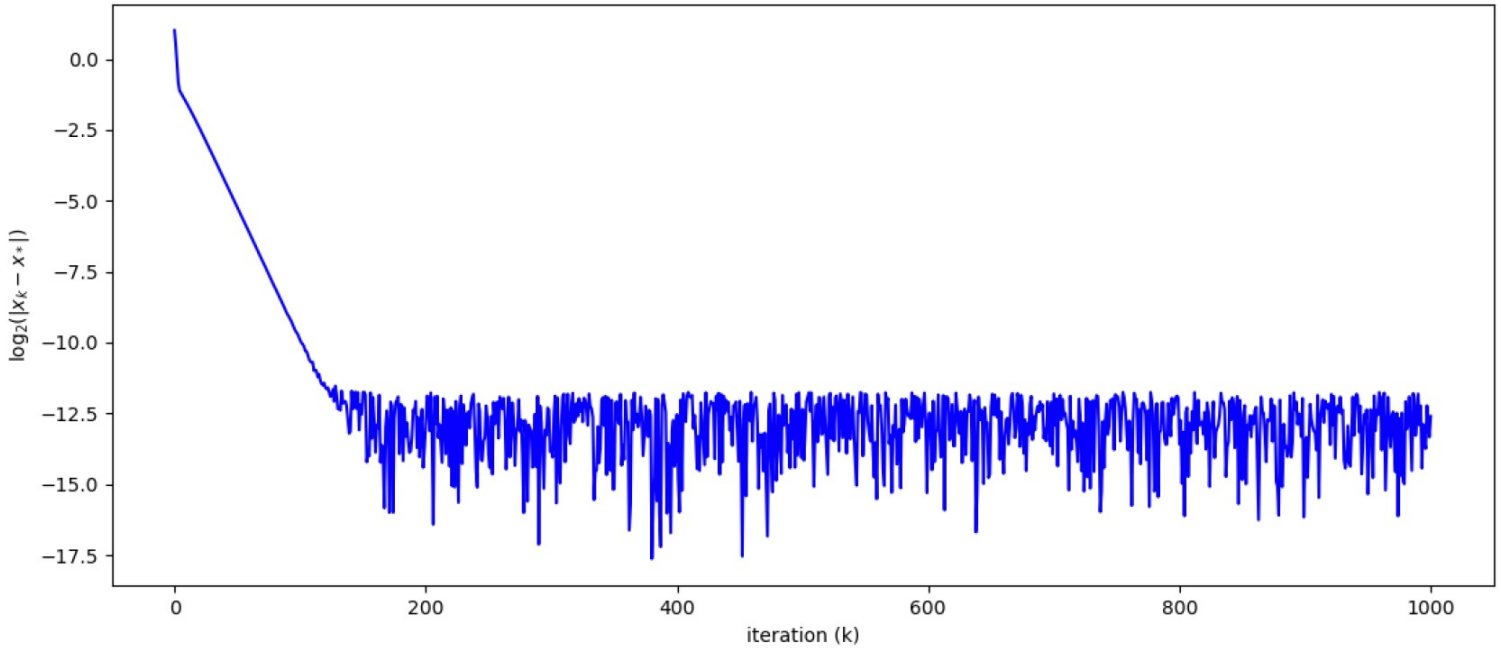
1. If $(\epsilon_f, \epsilon_g) > 0$ then K is finite
2. If an iterate enters noise neighborhood N, all subsequent iterates cannot stray too far away $(2\epsilon_f)$
3. For all good iterates sufficiently away from N lengthening is not necessary

- Success in the presence of intermittent noise



Numerical Results

Figure 4.1: Distance to optimality ($\log_2(\|x_k - x_*\|)$) vs iteration number for $\epsilon_1 = \epsilon_2 = 10^{-3}$



(a) HS7. $\epsilon_f = 1E-3, \epsilon_c = 1E-3, \epsilon_g = 1.41E-3, \epsilon_J = 1.41e-3$

The Algorithm

Input: x_0, $H_0 > 0$, lengthening parameter ℓ

For $k = 0,1,\ldots$

$p_k \leftarrow -H_k g_k$

Attempt to find α that satisfies the Armijo-Wolfe for (f, g)

If succeeded: $\alpha_k \leftarrow \alpha$

else $\alpha_k \leftarrow 0$

If $\| \alpha_k p_k \| \geq \ell$

$s_k \leftarrow \alpha_k p_k$, $\quad y_k \leftarrow g(x_k + s_k) - g(x_k)$ [usual]

else

$s_k \leftarrow \ell \dfrac{p_k}{\| p_k \|}$, $\quad y_k = g(x_k + s_k) - g(x_k)$ [lengthening, extra gradient]

end if

Update inverse Hessian approx; compute new iterate

$H_{k+1} = BFGS(H_k, s_k, y_k) \qquad x_{k+1} \leftarrow x_k + \alpha_k p_k$ [could be zero]

end for

Three application classes

Monte Carlo simulation of physical model with uncertainties

- optimize engineering system modeled by differential equations
- in which some physical parameters are uncertain.
- Monte Carlo
- Objective is an expectation

Mixed Precision Arithmetic and Adjoints

- for atmospheric and ocean sciences.
- The gradient is based on a lower fidelity model; the objective
- is noisy. These problems are similar in nature to parameter
- identification problems.

Empirical risk minimization problem in machine learning

- Multi-class logistic regression or neural networks.

THE END