Proximal Trust Region Methods for Nonsmooth, Nonconvex Inverse Problems

Robert Baraldi

CSGF Annual Meeting

Overview I

Inverse Problems in Optimization

- Key facet of scientific computing is inverse problem solving:
  - Numerically computing parameters by minimizing a cost function
  - Parameters are used to estimate unobserved data
- Just a few applications...
  - Tomography interpolation
  - Neuron Firing/biological parameter fitting
  - Machine Learning - neural networks training
  - Seismic Denoising

Inverse Problem Cost Functions and Regularizers I

- Separable cost functions:
  - Sum of two functions (with constraints) with exploitable characteristics; (non)smoothness, (non)convexity

$$\underset{x}{\text{minimize}} \; f(x) + h(x) \tag{1}$$

- Smooth term f- contains derivative information
  - Usually data misfit
  - Nonconvex in nonlinear functions - PDE/ODE inverse problems, ML, etc

Inverse Problem Cost Functions and Regularizers II

- Nonsmooth term h - regularizers that often promote sparsity for ill-conditioned problems
  - Large datasets encourage overfitting
  - Model-complexity is moderated by sparsity-inducing functions, but these lack derivatives
  - Examples: sparse regression, matrix completion (rank), phase retrieval, TV regularization
  - In literature: usually convex approximations of nonconvex functions

Nonconvexity and nonsmoothness: Why?

- Useful but difficult
  - Problems: no global optimum, lack derivatives
  - Theory/software exists for smoothed/convex counterparts
  - Nonlinear problems have rich history, but require differentiability
- Talk Goals:
  1. Find algorithms for broad class of nonsmooth, nonconvex functions and regularizers
  2. Bridge gap between classic nonlinear opt. methods and structured nonsmooth, nonconvex problems

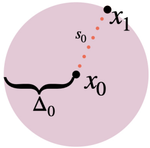
Focus: Quasi-Newton PG + TR Method

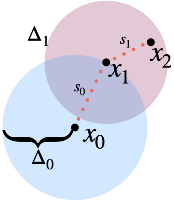
- Two broad optimization camps: linesearch vs trust regions (TR)
  - Linesearch: pick direction, find step along that direction
  - TR: build model, optimize over restricted region
- TR methods: Fast/efficient, smooth models over ℓ_2 regions
- Extensions to nonsmooth settings are limited
- **Contribution**: Extend TR methods to entire nonsmooth class $f + h$, provide implementation
  - Tools: Proximal Gradient (PG), TR

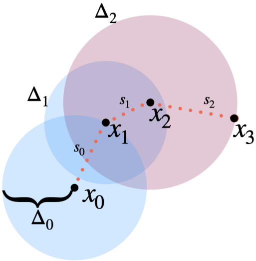
Brief Trust Region Introduction

- Trust-region methods: numerically efficient approximations of nonlinear functions.
- k^{th} iteration uses surrogate quadratic model of smooth f:
  - Gradient ∇f, Hessian approximation $B_k = B_k^T$
  - Valid within a region determined by quadratic model performance and accuracy
- Saves numerical cost for expensive forward solutions.
- Nonsmooth TR exists, but is restrictive or impractical.

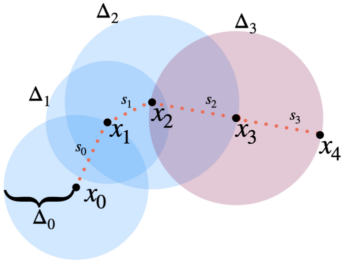
Figure 1: TR Example Path



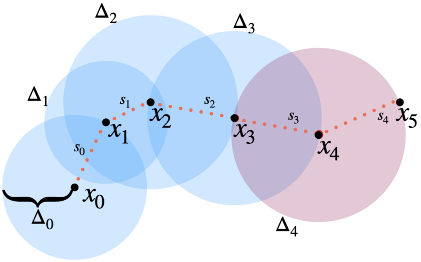

Δ_1
s_1
x_2
x_1
s_0
x_0
Δ_0



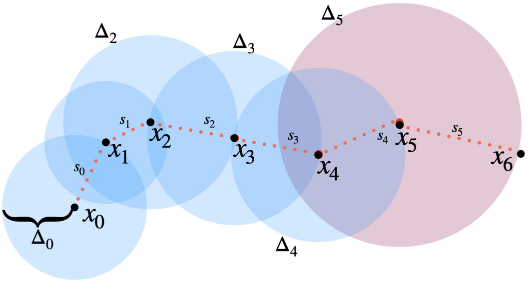
Δ_2
Δ_1
s_1
s_2
x_2
x_1
x_3
s_0
x_0
Δ_0



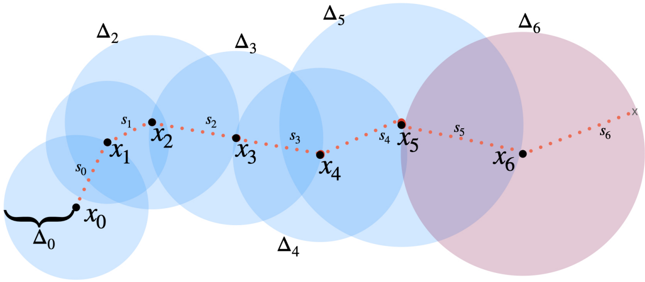
Δ_2
Δ_3
Δ_1
s_1
s_2
s_3
x_2
x_1
x_3
x_4
s_0
x_0
Δ_0

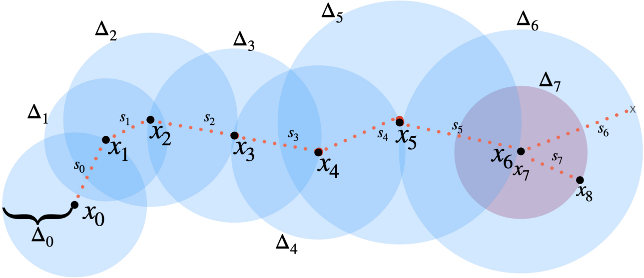


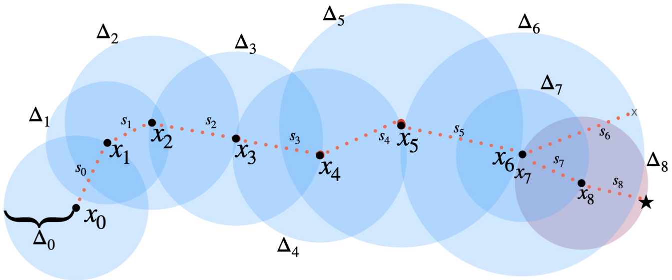
Δ_2
Δ_3
Δ_1
s_1
x_2
s_2
s_3
s_4
x_5
x_1
x_3
x_4
s_0
x_0
Δ_0
Δ_4



Δ5
Δ2
Δ3
s1
s2
s3
s4
s5
x1
x2
x3
x4
x5
x6
s0
x0
Δ0
Δ4


Δ_2
Δ_3
Δ_5
Δ_6
Δ_0
Δ_4
x_0
x_1
x_2
x_3
x_4
x_5
x_6
s_0
s_1
s_2
s_3
s_4
s_5
s_6
x


Δ_0
Δ_1
Δ_2
Δ_3
Δ_4
Δ_5
Δ_6
Δ_7
x_0
x_1
x_2
x_3
x_4
x_5
x_6
x_7
x_8
s_0
s_1
s_2
s_3
s_4
s_5
s_6
s_7
x


Δ_0
Δ_1
Δ_2
Δ_3
Δ_4
Δ_5
Δ_6
Δ_7
Δ_8
x_0
x_1
x_2
x_3
x_4
x_5
x_6
x_7
x_8
s_0
s_1
s_2
s_3
s_4
s_5
s_6
s_7
s_8

Recap: Problem Class and Goals

- Problem Statement: $\min_x \; f(x) + h(x)$
  - $f \in \mathcal{C}^1$, h proper, lsc.
- Contribution: TR method where **steps** are computed by minimizing simpler nonsmooth models based on PG.
- Results:
  - Global convergence
  - $O(1/\epsilon^2)$ worst-case complexity - equivalent to smooth cases
  - Comparisons between PG and QR method

TR Analysis Outline

1. Assume that we generated s_k that optimizes $m_k(s; x_k)$ via PG. How do we extend TR theory to nsmth ncvx case?
2. How do we generate s_k via PG?

Tricky: we have an outer/overall TR problem and an inner s_k problem!

TR Theoretical Results - # 1

- Adapting model assumptions to nonsmooth case $\Rightarrow$ similar convergence of the smooth-case trust-region algorithms!
  - Monotonic Decrease in objective value
  - Eventually get an s within the trust-region (a *successful* iteration)
  - $\mathcal{O}(1/\epsilon^2)$ iteration complexity
  - $\lim_{k\to\infty} f(x_k) + h(x_k) \to -\infty$ or $\lim_{k\to\infty} \xi(\Delta_0; x_k) = 0$: i.e. eventual first-order convergence

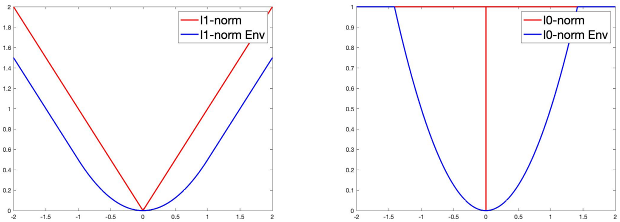
Proximal Operator - #2

Proper, lsc function $h : \mathbb{R}^n \to \overline{\mathbb{R}}$, $\nu > 0$, the *Moreau envelope* $e_{\nu h}$ and the *proximal mapping* $\text{prox}_{\nu h}$ are defined by

$$e_{\nu h}(x) := \inf_w \frac{1}{2\nu}\|w - x\|^2 + h(w), \tag{2a}$$

$$\underset{\nu h}{\text{prox}}(x) := \arg\min_w \frac{1}{2\nu}\|w - x\|^2 + h(w). \tag{2b}$$

Figure 2: Two common proximal operators and their envelopes ($\nu = 1$)



Determining Subproblem Solutions

- To produce an s, we need to solve

$$\underset{s}{\text{minimize}} \;\; m_k := \varphi(s) + \psi(s) + \chi(s), \tag{3}$$

- Tool: Proximal gradient updates
- Initialized a $s_0 = 0$ where $\psi + \chi$ is finite, it generates iterates according to

$$s_{j+1} \in \underset{\nu(\psi+\chi)}{\text{prox}} (s_j - \nu \nabla \varphi(s_j)), \quad j \geq 0, \tag{4}$$

where $\nu > 0$ is a (sometimes) fixed step size.

Every PG-step Decreases Surrogate Models I

- Descent for every inner step (Bolte, Sabach, and Teboulle, 2014)
- PG converges sublinearly to a stationary point of $\varphi + \psi$.
- Results: We eventually arrive at $0 \in \partial(\varphi + \psi + \chi)(s_k)$ - i.e. a stationary point of the surrogate model

Theoretical Conclusions; Numerical Comparisons

- Theory:
  - Outer/TR Method: s_k created by nonsmooth means (PG) still converges to critical point of $f + h$
  - Inner/PG Method: PG will create an s_k, eventually reaches critical point of model
- Next:
  - Perform model reduction on nonlinear inverse problem
  - Compare against two similar methods: PANOC and ZeroFPR

Classical ODE Inverse Problem

We would like to solve

$$\min_x \|F(x) - b\|_2^2 + h(x). \tag{5}$$

where $F(x)$ is the solution of a system of ODEs.

Fitzhugh-Nagumo Model

The Fitzhugh-Nagumo model for neuron activation is given by

$$\frac{dV}{dt} = (V - V^3/3 - W + x_1)x_2^{-1} \quad (6a)$$

$$\frac{dW}{dt} = x_2(x_3 V - x_4 W + x_5). \quad (6b)$$

For $x_1 = x_4 = x_5 = 0$, it becomes the Van-der-Pol oscillator

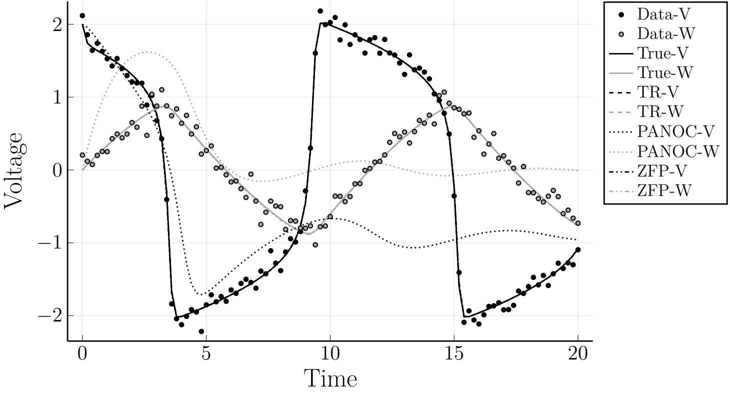
$$\frac{dV}{dt} = (V - V^3/3 - W)x_2^{-1} \quad (7a)$$

$$\frac{dW}{dt} = x_2(x_3 V). \quad (7b)$$

- Highly nonlinear and ill-conditioned
- LBFGS for $h(x) = \lambda\|x\|_0$ and an ℓ_∞-norm TR ball
- Goal: Fit data, exactly enforce $x_1 = x_4 = x_5 = 0$

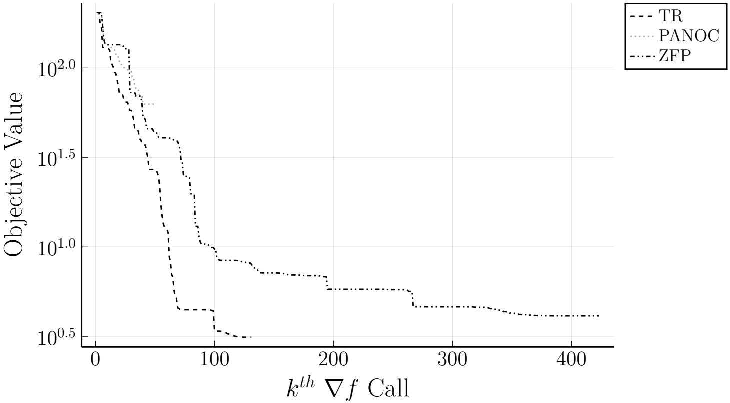
TR Results I

Figure 3: Fitzhugh-Nagumo solution ((6a), (6b)) for $h(x) = \lambda\|x\|_0$ in (5) with ℓ_∞-norm TR and LBFGS approximation.



a Solution Comparisons

TR Results II



b Objective Descent

Conclusions & Current Work

- Theoretical
  - General Prox Operator computation?
  - Extension to penalty methods
  - Different B_k operators - LBFGS, LSR1, Gauss-Newton/NLS
- Practical
  - Finalize numerical Julia packages/tests (https://github.com/UW-AMO/TRNC) - extensions to C++
  - Add in constraints/barrier methods
  - Implementation for harder PDE examples

Future Directions

- Inexact methods for PDE-constrained optimization
  - Imprecise gradient, subgradients
  - Inexact prox solution for incomputable proxes
  - Semismooth regularizer specifics
- Fast linear algebra for ν_k computation
- Fidelity-tuning for numerical simulations
- Applications to PDE-constrained inversion in CFD, earth/climate modeling, ... huge host of national lab resources
- Numerical software/HPC implementation - Trilinos/ROL, Dakota, GPU compatibility

Thank you!

- Questions?

References I

J. Bolte, S. Sabach, and M. Teboulle. Proximal alternating linearized minimization for nonconvex and nonsmooth problems. 1(146):459—-494, 2014. DOI: 10.1007/s10107-013-0701-9.