Second-Order Type Optimization Methods For Data Analysis

Zaiwen Wen

Beijing International Center For Mathematical Research
Peking University

Overview

- Xiantao Xiao, Yongfeng Li, Zaiwen Wen, Liwei Zhang, Semi-Smooth Second-order Type Methods for Composite Convex Programs, Journal of Scientific Computing
- Yongfeng Li, Zaiwen Wen, Chao Yang, Yaxiang Yuan, A Semi-smooth Newton Method for Semidefinite programming in electronic structure calculation, `https://arxiv.org/abs/1708.08048`
- Andre Milzarek, Xiantao Xiao, Shicong Cen, Zaiwen Wen, Michael Ulbrich, A Stochastic Semismooth Newton Method for Nonsmooth Nonconvex Optimization
  `https://arxiv.org/abs/1803.03466`
- Jiang Hu, Andre Milzarek, Zaiwen Wen, Yaxiang Yuan, Adaptive regularized Newton method for optimization on Riemannian manifold, SIAM Journal on Matrix Analysis and Applications
- Chao Ma, Xin Liu, Zaiwen Wen, Globally Convergent Levenberg-Marquardt Method For Phase Retrieval

Outline

1. Basic Concepts of Semi-smooth Newton method
2. Semi-smooth Newton method for SDP
3. Stochastic Semi-smooth Newton Method
4. Regularized Newton Method for Optimization on Manifold
5. Modified Levenberg-Marquardt Method For Phase Retrieval

Composite convex program

Consider the following composite convex program

$$\min_{x \in \mathbb{R}^n} \quad f(x) + h(x),$$

where f and h are convex, f is differentiable but h may not

Many applications:

- Sparse and low rank optimization: $h(x) = \|x\|_1$ or $\|X\|_*$ and many other forms.
- Regularized risk minimization: $f(x) = \sum_i f_i(x)$ is a loss function of some misfit and h is a regularization term.
- Constrained program: h is an indicator function of a convex set.

A General Recipe

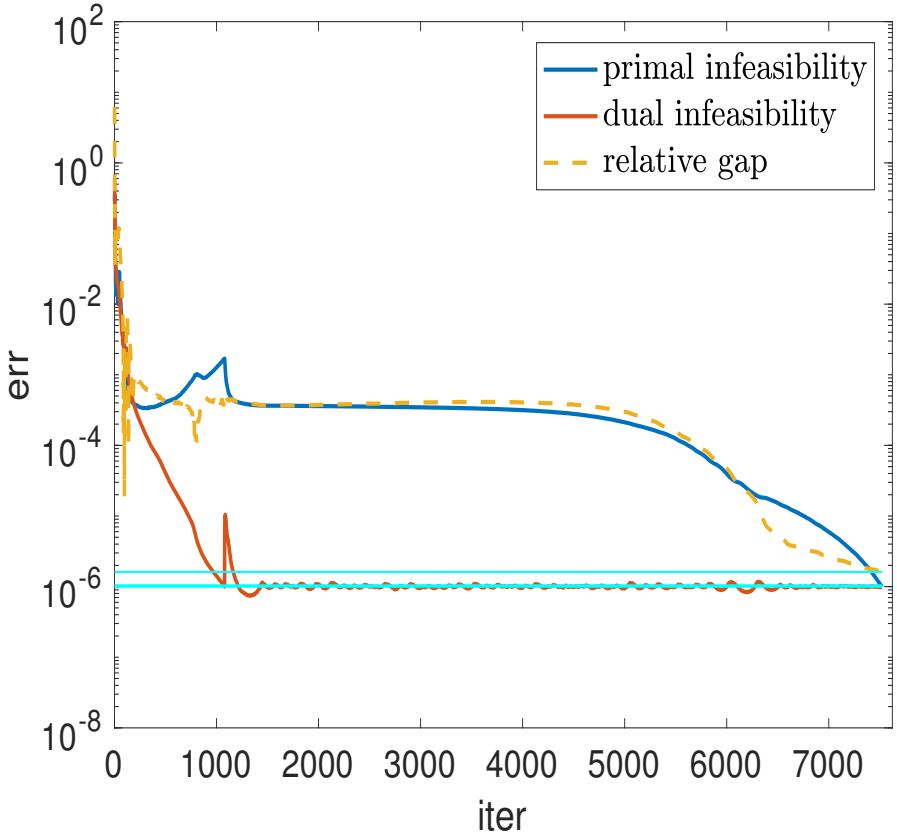
Goal: study approaches to bridge the gap between first-order and second-order type methods for composite convex programs.

key observations:

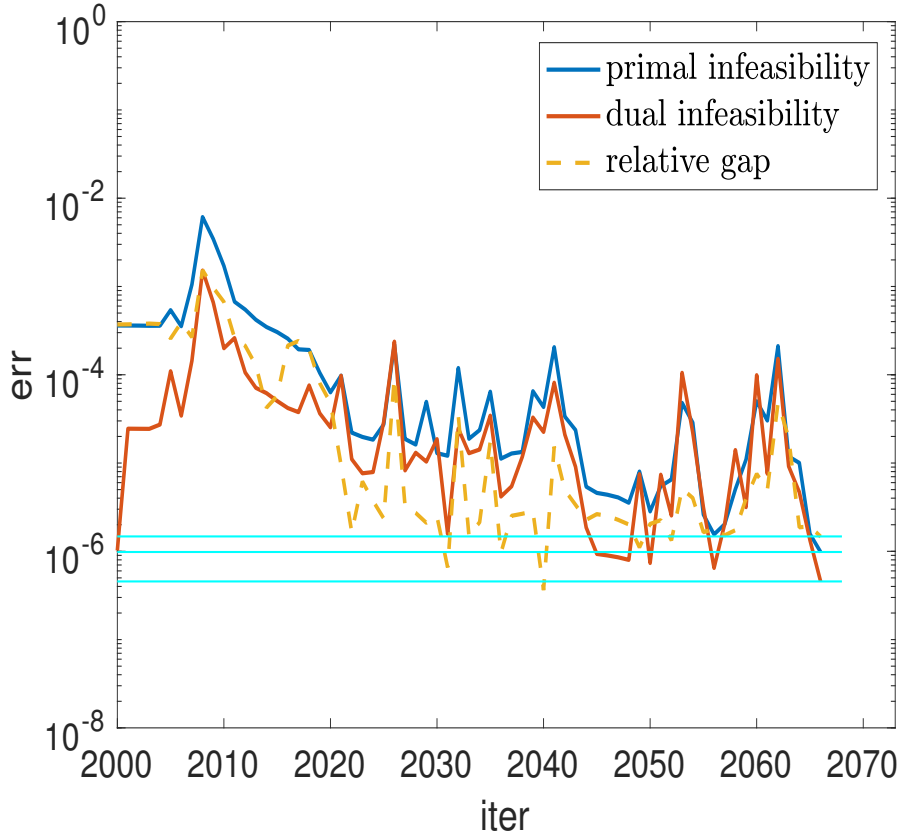
- Many popular first-order methods can be equivalent to some fixed-point iterations: $x^{k+1} = T(x^k)$;
  - **Advantages:** easy to implement; converge fast to a solution with moderate accuracy.
  - **Disadvantages:** slow tail convergence.
- The original problem is equivalent to the system $F(x) := (I - T)(x) = 0$.
- Newton-type method since $F(x)$ is semi-smooth in many cases
- Computational costs can be controlled reasonably well

An SDP From Electronic Structure Calculation

system: BeO



(a) ADMM, CPU: 2003



(b) Semi-smooth Newton, CPU: 635

Forward-backward splitting (FBS)

- *proximal mapping*:

$$\mathrm{prox}_{th}(x) := \underset{u\in\mathbb{R}^n}{\mathrm{argmin}}\ \{h(u) + \frac{1}{2t}\|u - x\|_2^2\}.$$

- FBS is the iteration

$$\begin{aligned} x^{k+1} &= \mathrm{prox}_{th}(x^k - t\nabla f(x^k)), k = 0, 1, \cdots, \\ &= \arg\min_x \left\langle \nabla f(x^k), x - x^k \right\rangle + \frac{1}{2t}\|x - x^k\|_2^2 + h(x) \end{aligned}$$

- Equivalent to a fixed-point iteration

$$x^{k+1} = T_{\mathrm{FBS}}(x^k).$$

where

$$T_{\mathrm{FBS}} := \mathrm{prox}_{th} \circ (I - t\nabla f).$$

Douglas-Rachford splitting (DRS)/ADMM

- DRS is the following update:

$$\begin{aligned} x^{k+1} &= \mathrm{prox}_{th}(z^k), \\ y^{k+1} &= \mathrm{prox}_{tf}(2x^{k+1} - z^k), \\ z^{k+1} &= z^k + y^{k+1} - x^{k+1}. \end{aligned}$$

- Equivalent to a fixed-point iteration

$$z^{k+1} = T_{\mathrm{DRS}}(z^k),$$

where

$$T_{\mathrm{DRS}} := I + \mathrm{prox}_{tf} \circ (2\mathrm{prox}_{th} - I) - \mathrm{prox}_{th}.$$

- The ADMM to the primal is equivalent to the DRS to the dual

Semi-smoothness

- Solving the system
$$F(z) = 0,$$
where $F(z) = T(z) - z$ and $T(z)$ is a fixed-point mapping.
- $F(z)$ fails to be differentiable in many interesting applications.
- but $F(z)$ is (strongly) semi-smooth and monotone.
  (a) F is directionally differentiable at x; and
  (b) for any $d \in \mathbb{R}^n$ and $J \in \partial F(x + d)$,
$$\|F(x + d) - F(x) - Jd\|_2 = o(\|d\|_2) \quad \text{as } d \to 0.$$

A regularized semi-smooth Newton method

- The Jacobian $J_k \in \partial_B F(z^k)$ is positive semidefinite
- Let $\mu_k = \lambda_k \|F^k\|_2$. Constructe a Newton system:

$$(J_k + \mu_k I)d = -F^k,$$

- Solving the Newton system inexactly:

$$r^k := (J_k + \mu_k I)d^k + F^k.$$

We seek a step d^k approximately such that

$$\|r^k\|_2 \leq \tau \min\{1, \lambda_k \|F^k\|_2 \|d^k\|_2\}, \quad \text{where } 0 < \tau < 1$$

- Newton Step: $z^{k+1} = z^k + d^k$
- Faster local convergence is ensured

Outline

1. Basic Concepts of Semi-smooth Newton method
2. Semi-smooth Newton method for SDP
3. Stochastic Semi-smooth Newton Method
4. Regularized Newton Method for Optimization on Manifold
5. Modified Levenberg-Marquardt Method For Phase Retrieval

Semidefinite Programming

Consider the SDP

$$\min \langle C, X \rangle, \text{s.t.} \quad \mathcal{A}X = b, X \succeq 0$$

- $f(X) = \langle C, X \rangle + 1_{\{\mathcal{A}X=b\}}(X)$.
- $h(X) = 1_K(X)$, where $K = \{X : X \succeq 0\}$.
- Proximal Operator: $\text{prox}_{th}(Z) = \arg\min_X \frac{1}{2}\|X - Z\|_F^2 + th(X)$
- Let $Z = Q\Sigma Q^T$ be the spectral decomposition

$$\begin{aligned} \text{prox}_{tf}(Y) &= (Y + tC) - \mathcal{A}^*(\mathcal{A}Y + t\mathcal{A}C - b), \\ \text{prox}_{th}(Z) &= Q_\alpha \Sigma_\alpha Q_\alpha^T, \end{aligned}$$

- Fixed-point mapping from DRS:

$$F(Z) = \text{prox}_{th}(Z) - \text{prox}_{tf}(2\text{prox}_{th}(Z) - Z) = 0.$$

Semi-smooth Newton System

- assumption: $\mathcal{A}\mathcal{A}^* = I$
- The SMW theorem yields the inverse matrix

$$(J_k + \mu_k I)^{-1} = H^{-1} + H^{-1}A^T(I - AWH^{-1}A^T)^{-1}AWH^{-1}$$

$$= \frac{1}{\mu(\mu+1)}(\mu I + T)(I + A^\top(\frac{\mu^2}{2\mu+1}I + ATA^\top)^{-1}A(\frac{\mu}{2\mu+1}I - T)).$$

- $ATA^\top d = \mathcal{A}Q(\Omega_0 \circ (Q^T(D)Q))Q^T$, where $D = \mathcal{A}^* d$,

$$\Omega_0 = \begin{bmatrix} E_{\alpha\alpha} & I_{\alpha\bar{\alpha}} \\ I_{\alpha\bar{\alpha}}^T & 0 \end{bmatrix},$$

  and $E_{\alpha\alpha}$ is a matrix of ones and $I_{ij} = \frac{\mu k_{ij}}{\mu+1-k_{ij}}$

- computational cost $O(|\alpha|n^2)$

Semi-smooth Newton method

- Select $0 < \nu < 1$, $0 < \eta_1 \leq \eta_2 < 1$ and $1 < \gamma_1 \leq \gamma_2$. $\underline{\lambda} > 0$
- A trial point $U^k = Z^k + S^k$
- Define a ratio

$$\rho_k = \frac{-\left\langle F(U^k), S^k \right\rangle}{\|S^k\|_F^2}.$$

- Update the point

$$Z^{k+1} = \begin{cases} U^k, \text{ if } \|F(U^k)\|_F \leq \nu \max\limits_{\max(1,k-\zeta+1)\leq j\leq k} \|F(Z^j)\|_F, & \text{[Newton]} \\ Z^k, \text{otherwise.} & \text{[failed]} \end{cases}$$

- Update the regularization prameter

$$\lambda_{k+1} \in \begin{cases} (\underline{\lambda}, \lambda_k), & \text{if } \rho_k \geq \eta_2, \\ [\lambda_k, \gamma_1 \lambda_k], & \text{if } \eta_1 \leq \rho_k < \eta_2, \\ (\gamma_1 \lambda_k, \gamma_2 \lambda_k], & \text{otherwise,}. \end{cases}$$

Switching between the ADMM and Newton steps

the reduced ratios of primal and dual infeasibilities

$$\omega_{\eta_p}^k = \frac{\text{mean}_{k-5\le j\le k}\eta_p^j}{\text{mean}_{k-25\le j\le k-20}\eta_p^j} \text{ and } \omega_{\eta_q}^k = \frac{\text{mean}_{k-5\le j\le k}\eta_q^j}{\text{mean}_{k-25\le j\le k-20}\eta_q^j}.$$

Repeat:

- Semi-smooth Newton steps (doSSN == 1)
  Compute $U^k = Z^k + S^k$. Then update Z^{k+1} and λ_{k+1}.
  If Newton step is failed, set $N_f = N_f + 1$.
  If $N_f \ge \bar{N}_f$ or the Newton step performs bad
    Set doSSN = 0 and parameters for the ADMM steps
- ADMM steps (doSSN == 0)
  Perform an ADMM step.
  If the ADMM step performs bad
    Set doSSN = 1, $N_f = 0$ and parameters of the Newton steps

Global Convergence

Theorem

Suppose that $\{Z^k\}$ is a sequence generated by the semismooth Newton method. Then the residuals of $\{Z^k\}$ converge to 0, i.e., $\lim_{k\to\infty} \|F(Z^k)\| = 0$.

- If $\{Z^k\}$ is bounded, Then any accumulation point of $\{Z^k\}$ converges to some point $\bar{Z}$ such that $F(\bar{Z}) = 0$.
- This algorithm can solve the general composite optimization.

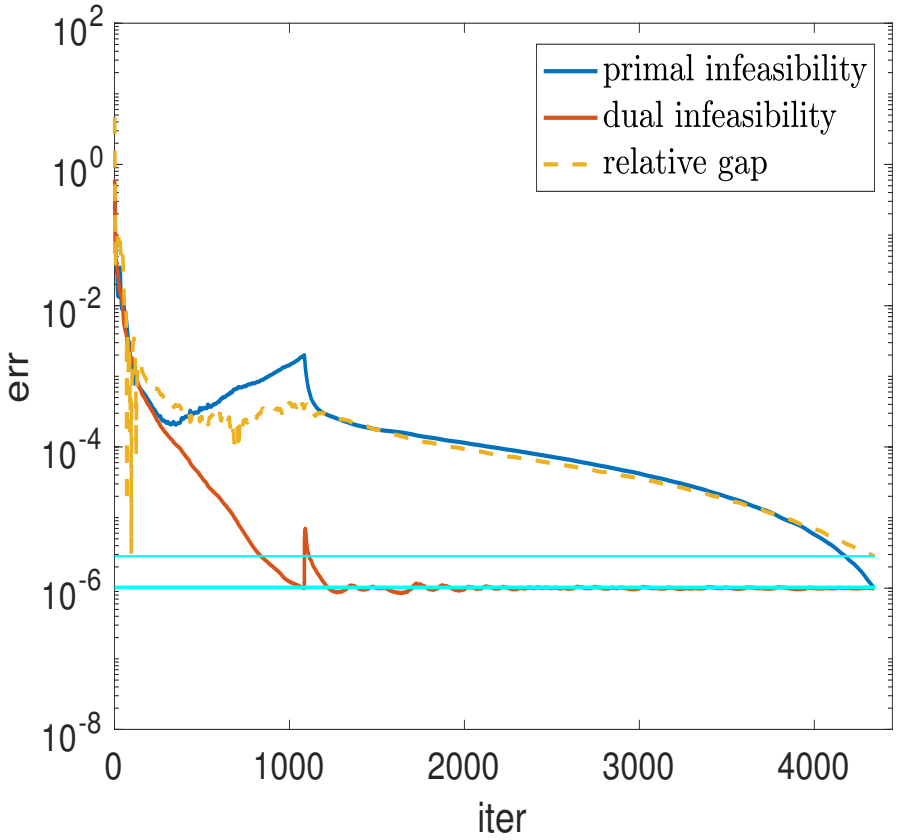
Comparison on electronic structure calculation

- The data set are used in the paper of Nakata, et al. Thanks Prof. Nakata Maho and Prof. Mituhiro Fukuta for sharing all data sets on 2RDM
- solver:
  - SDPNAL: Newton-CG Augmented Lagrangian Method proposed by Zhao, Sun and Toh
  - SDPNAL+: Enhanced version of SDPNAL by Yang, Sun and Toh
  - SSNSDP: the semi-smooth Newton method using stop rules $\eta_p < 3 \times 10^{-6}$ and $\eta_d < 3 \times 10^{-7}$.
- all experiments were performed on a computing cluster with an Intel Xeon 2.40GHz CPU that processes 28 cores and 256GB RAM.
- main criteria:

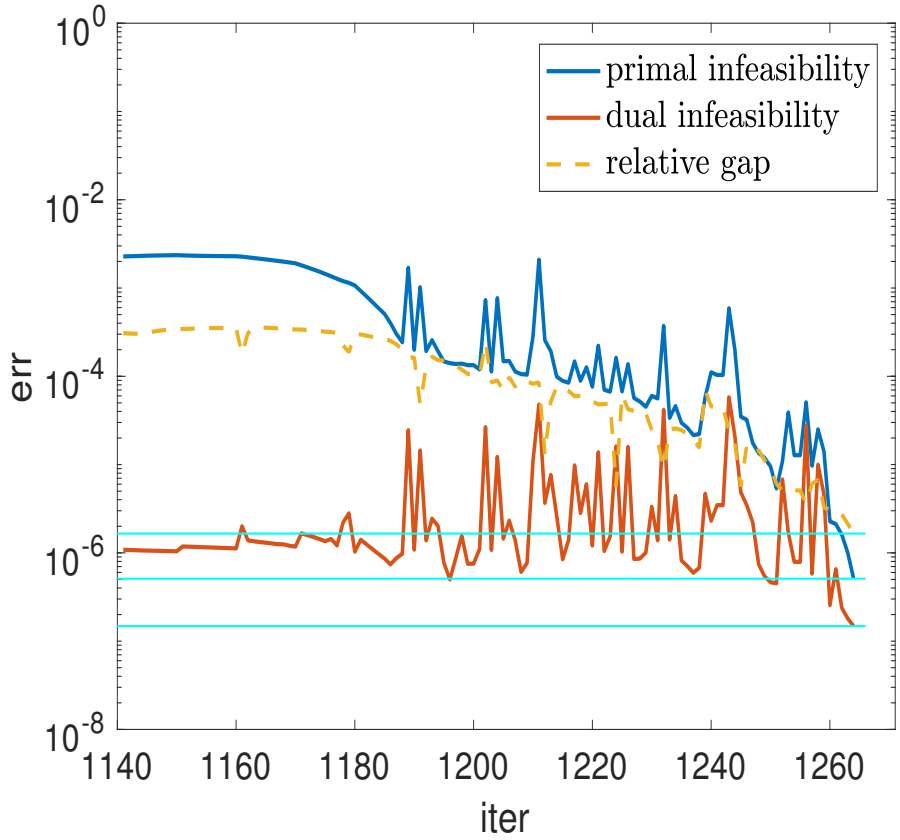
$$\eta_p = \frac{\|\mathcal{A}(X) - b\|_2}{\max(1, \|b\|_2)} \quad \eta_d = \frac{\|\mathcal{A}^* y - C - S\|_F}{\max(1, \|C\|_F)}$$

$$\eta_g = \frac{|b^T y - \mathrm{tr}(C^T X)|}{\max(1, \mathrm{tr}(C^T X))} \quad \mathrm{err} = b^T y - \mathrm{energy}_{\mathrm{fullCI}}$$

Compuational Results: C2

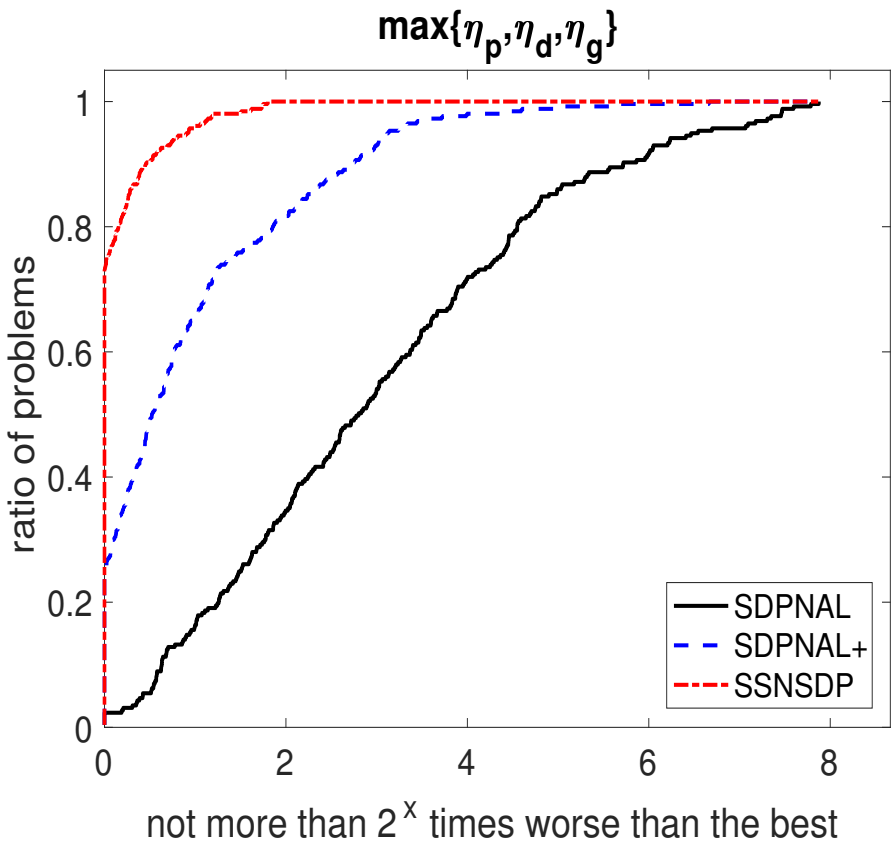


(c) ADMM, CPU: 41694

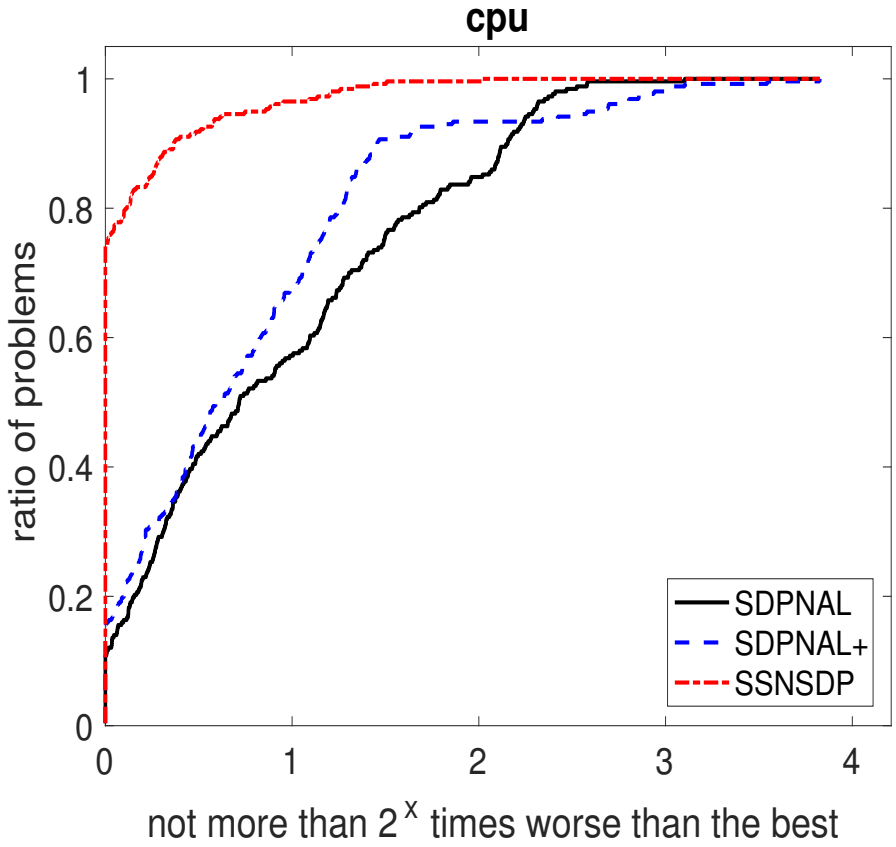


(d) Semi-smooth Newton, CPU: 14074

Comparison on electronic structure calculation



(e) max(η_p,η_d,η_g)



(f) cpu time

Comparison on electronic structure calculation

success: $\max\{\eta_p, \eta_d\} \leq 10^{-6}$

case	SSNSDP		SDPNAL		SDPNAL+	
	number	percentage	number	percentage	number	percentage
success	276	100%	53	19.2%	265	96%
fastest	205	74.3%	30	10.9%	41	14.9%
fastest under success	232	84.1%	3	1.09%	41	14.9%
not slower 1.2 times	236	85.5%	71	25.7%	87	31.5%
not slower 1.2 times under success	251	90.9%	5	1.81%	87	31.5%

Figure: Comparison between SDPNAL, SDPNAL+ and SSNSDP

Outline

1. Basic Concepts of Semi-smooth Newton method
2. Semi-smooth Newton method for SDP
3. Stochastic Semi-smooth Newton Method
4. Regularized Newton Method for Optimization on Manifold
5. Modified Levenberg-Marquardt Method For Phase Retrieval

Examples and Applications

- Consider

$$\min_x \quad f(x) + h(x) =: \psi(x)$$

- Expected and Empirical Risk Minimization:

$$f(x) := \mathbb{E}[F(x,\xi)] = \int_\Omega F(x,\xi(\omega))\, d\mathbb{P}(\omega), \quad f(x) := \frac{1}{N}\sum_{i=1}^{N} f_i(x)$$

Applications and Typical Situation:

- Large-scale machine learning problems, LASSO, sparse and bilinear logistic regression, low-rank matrix completion, sparse dictionary learning, ...
- $\mathbb{P}$ is not known (completely) or N is very large.

$\rightsquigarrow$ Full evaluation of f and ∇f is impractical or even not possible.

$\rightsquigarrow$ Use stoch. optimization techniques and sampling strategies!

Algorithmic Idea

Basic idea based on $x^{k+1} = T_{\mathrm{FBS}}(x^k) = \mathrm{prox}_h^{\Lambda}(x^k - t\nabla f(x^k))$.

- The proximity operator [*Moreau* '65]

$$\mathrm{prox}_h^{\Lambda}(y) := \mathrm{argmin}_z \; h(z) + \frac{1}{2}\|y - z\|_{\Lambda}^2.$$

- We incorporate second order information and use stochastic Hessian oracles ($\mathcal{SSO}$)

$$H(x^k; t^k) \approx \nabla^2 f(x^k)$$

  to estimate the Hessian $\nabla^2 f$ and compute the Newton step.
- The sample collections s^k and t^k are chosen independently of each other and of the other batches s^ℓ, t^ℓ, $\ell \in \mathcal{N}_0 \setminus \{k\}$.
- Let $G : \mathbb{R}^n \times \Xi \to \mathbb{R}^n$ and $H : \mathbb{R}^n \times \Xi \to \mathbb{S}^n$ be Carathéodory functions. We work with the following $\mathcal{SFO}$ and $\mathcal{SSO}$:

$$\mathcal{G}_{s^k}(x) := \frac{1}{\mathbf{n}_k^{\mathrm{g}}} \sum_{i=1}^{\mathbf{n}_k^{\mathrm{g}}} G(x; s_i^k) \quad \text{and} \quad \mathcal{H}_{t^k}(x) := \frac{1}{\mathbf{n}_k^{\mathrm{h}}} \sum_{j=1}^{\mathbf{n}_k^{\mathrm{h}}} H(x; t_j^k).$$

Stochastic Semi-smooth Newton Method: Idea

To accelerate the stochastic proximal gradient method, we want to augment it by a stochastic Newton-type step, obtained from the (sub-sampled) optimality condition:

$$F_s^\Lambda(x) = x - \mathrm{prox}_h^\Lambda(x - \Lambda^{-1}\mathcal{G}_s(x)) \approx 0.$$

The semi-smooth Newton step is given by

$$M_k\, d^k = -F_{s^k}^\Lambda(x^k), \quad x^{k+1} = x^k + d^k,$$

with sample batches s^k, t^k and $M_k \in \mathcal{M}_{s^k,t^k}^{\Lambda_k}(x^k)$,

$$\mathcal{M}_{s,t}^\Lambda(x) := \{M = I - D + D\Lambda^{-1}\mathcal{H}_t(x) : D \in \partial\mathrm{prox}_h^\Lambda(u_s^\Lambda(x))\}$$

and $u_s^\Lambda(x) := x - \Lambda^{-1}\mathcal{G}_s(x)$.

$\rightsquigarrow$ Aim: Utilize fast local convergence to stationary points!

Stochastic Semismooth Newton Method

Sub-sampled Semi-smooth Newton Method (S4N)

0. Choose $x^0 \in \operatorname{dom} h$, batch sizes $(\mathbf{n}_k^g)$, $(\mathbf{n}_k^h)$, matrices (Λ_k), and step sizes (α_k). Select ind. batches s^0, t^0. Set $k := 0$.

   While "**not converged**" do:

1. Compute $F_{s^k}^{\Lambda_k}(x^k)$ and choose $M_k \in \mathcal{M}_{s^k,t^k}^{\Lambda_k}(x^k)$. Select new sample batches s^{k+1}, t^{k+1}.
2. Compute the semismooth Newton step via

$$M_k d^k = -F_{s^k}^{\Lambda_k}(x^k).$$

   If this is not possible, go to step 4.

3. Set $z^k := x^k + d^k$. If $z^k \in \operatorname{dom} h$ and z^k satisfies the growth conditions $(\star)$, set $x^{k+1} := z^k$ and go to step 5.
4. Compute a proximal gradient step $x^{k+1} := x^k - \alpha_k F_{s^k}^{\Lambda_k}(x^k)$.
5. Increment k and go to step 1.

Algorithmic Framework (Cont')

We use the following growth conditions ($\star$) in step 3:

$$\|F_{s^{k+1}}^{\Lambda_{k+1}}(z^k)\| \le (\eta + \nu_k)\cdot\theta_k + \varepsilon_k^1, \tag{G.1}$$

$$\psi(z^k) \le \psi(x^k) + \beta\cdot\theta_k^{1/2}\|F_{s^{k+1}}^{\Lambda_{k+1}}(z^k)\|^{1/2} + \varepsilon_k^2, \tag{G.2}$$

where $\eta \in (0,1)$, $\beta > 0$, and $(\nu_k), (\varepsilon_k^2) \in \ell_+^1$, $(\varepsilon_k^1) \in \ell_+^{1/2}$.

We set θ_{k+1} to $\|F_{s^{k+1}}^{\Lambda_{k+1}}(x^{k+1})\|$ if x^{k+1} was obtained in step 3.

Remark:

- Calculating $F_{s^{k+1}}^{\Lambda_{k+1}}(z^k)$ requires evaluation of $\mathcal{G}_{s^{k+1}}(z^k)$. This information can be reused in the next iteration if $z^k \rightsquigarrow x^{k+1}$ is accepted as new iterate.

Global Convergence: Assumptions

Basic Assumptions:

(A.1) ∇f is Lipschitz continuous on $\mathbb{R}^n$ with constant L.

(A.2) The matrices $(\Lambda_k) \subset \mathbb{S}^n_{++}$ satisfy $\lambda_M I \geq \Lambda_k \geq \lambda_m I$ for all k.

(A.3) ψ is bounded from below on **dom** h.

Stochastic Assumptions:

(S.1) For all $k \in \mathcal{N}$, there exists $\sigma_k \geq 0$ such that

$$\mathbb{E}[\|\nabla f(x^k) - \mathcal{G}_{s^k}(x^k)\|^2] \leq \sigma_k^2.$$

(S.2) The matrices M_k, chosen in step 1, are random operators.

Global Convergence

Theorem: Global Convergence [*MXCW*, '17]

Suppose that (A.1)–(A.3) and (S.1)–(S.2) are fulfilled. Then, under the additional conditions, $\alpha_k \leq \overline{\alpha} := \min\{1, \lambda_m/L\}$,

$$(\alpha_k) \text{ is nonincreasing }, \quad \sum \alpha_k = \infty, \quad \sum \alpha_k \sigma_k^2 < \infty$$

it holds $\liminf_{k\to\infty} \mathbb{E}[\|F^\Lambda(x^k)\|^2] = 0$ and $\liminf_{k\to\infty} F^\Lambda(x^k) = 0$ a.s. for any $\Lambda \in \mathbb{S}^n_{++}$.

- Verify that (x^k) actually defines an adapted stochastic process.
- The batch s^k and the iterate x^k are not independent.
- Derive approximate and uniform descent estimates for the terms $\psi(x^k) - \psi(x^{k+1})$.

For strongly convex case: $\lim_{k\to\infty} \mathbb{E}[\|F^\Lambda(x^k)\|^2] = 0$ and $\lim_{k\to\infty} F^\Lambda(x^k) = 0$ a.s. for any $\Lambda \in \mathbb{S}^n_{++}$.

Numerical Results: Sparse Logistic Regression

We consider the following ℓ_1-regularized logistic regression problem

$$\min_x \quad \frac{1}{N}\sum_{i=1}^{N} f_i(x) + \mu\|x\|_1, \qquad f_i(x) := \log(1+\exp(-b_i \cdot a_i^\top x))$$

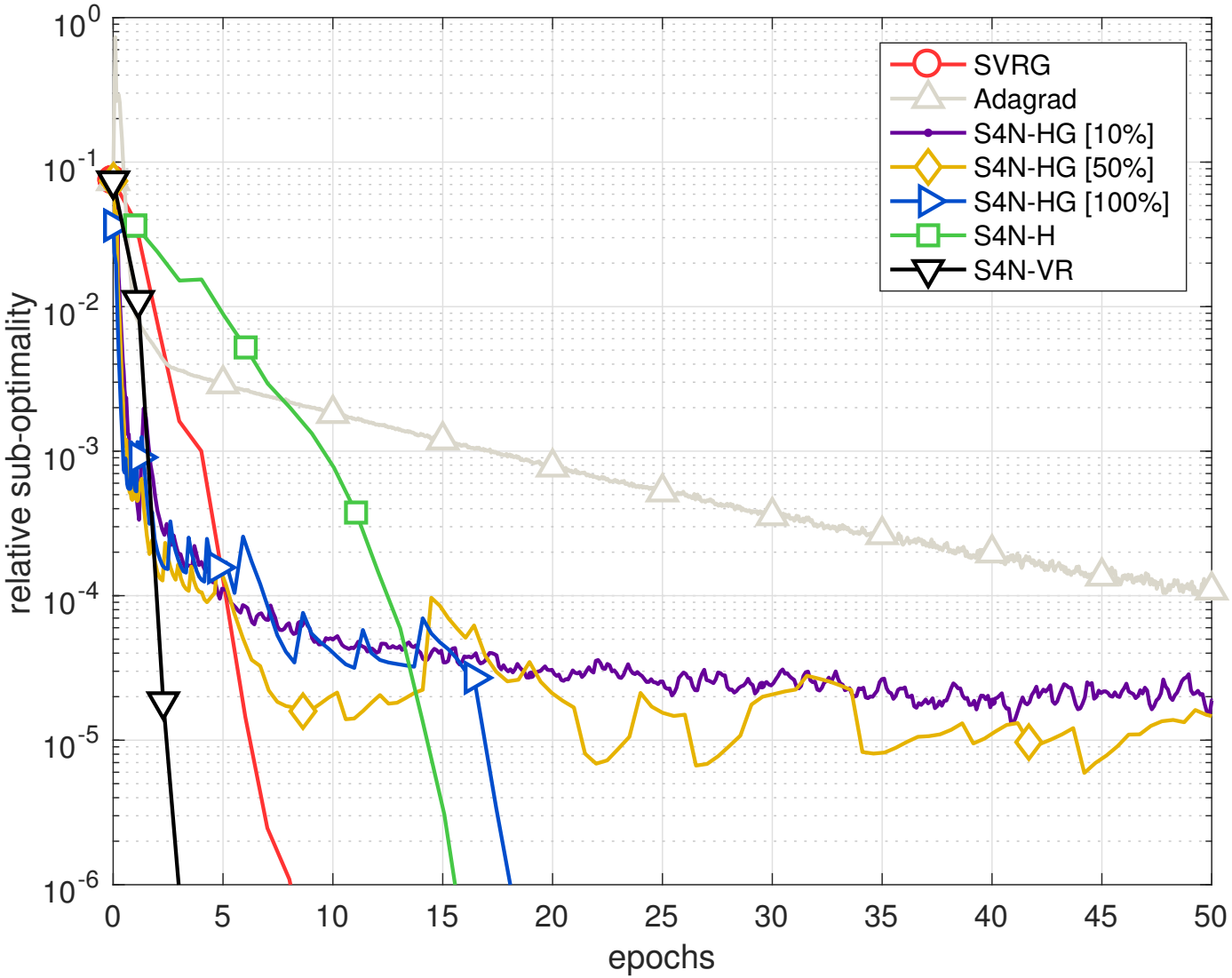
where $a_i^\top \in \mathbb{R}^n$ denotes the ith row of the data matrix $A \in \mathbb{R}^{N\times n}$ and $b \in \{-1,1\}^N$ is a binary vector.

Specifications of the test framework:[1]

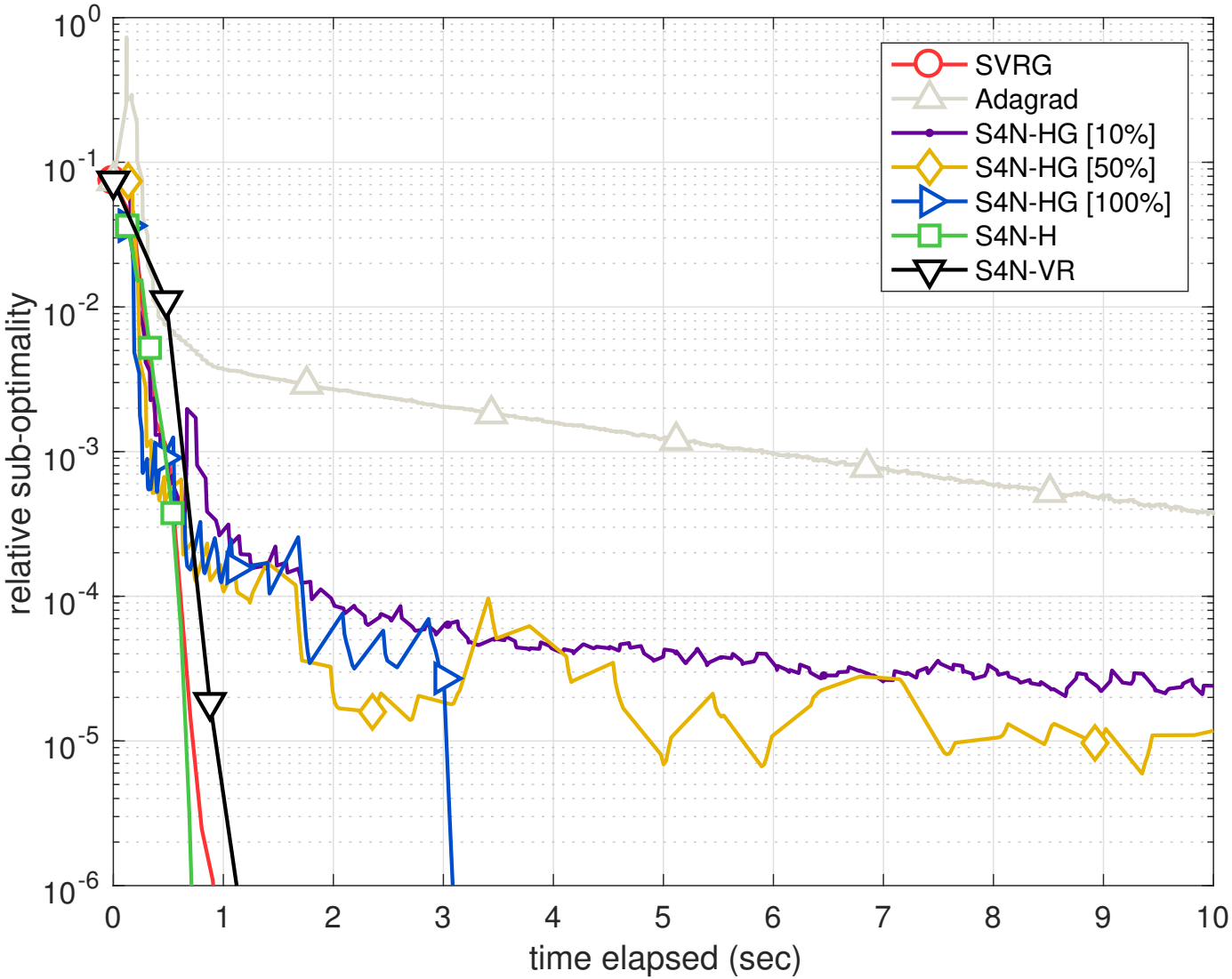
dataset	data points N	features n	
covtype	581 012	54	$\mu =$ 5e-3
gisette	6 000	5 000	$\mu =$ 5e-2
rcv1	20 242	47 236	$\mu =$ 1e-3

[1] LIBSVM - www.csie.ntu.edu.tw/ cjlin/libsvm/

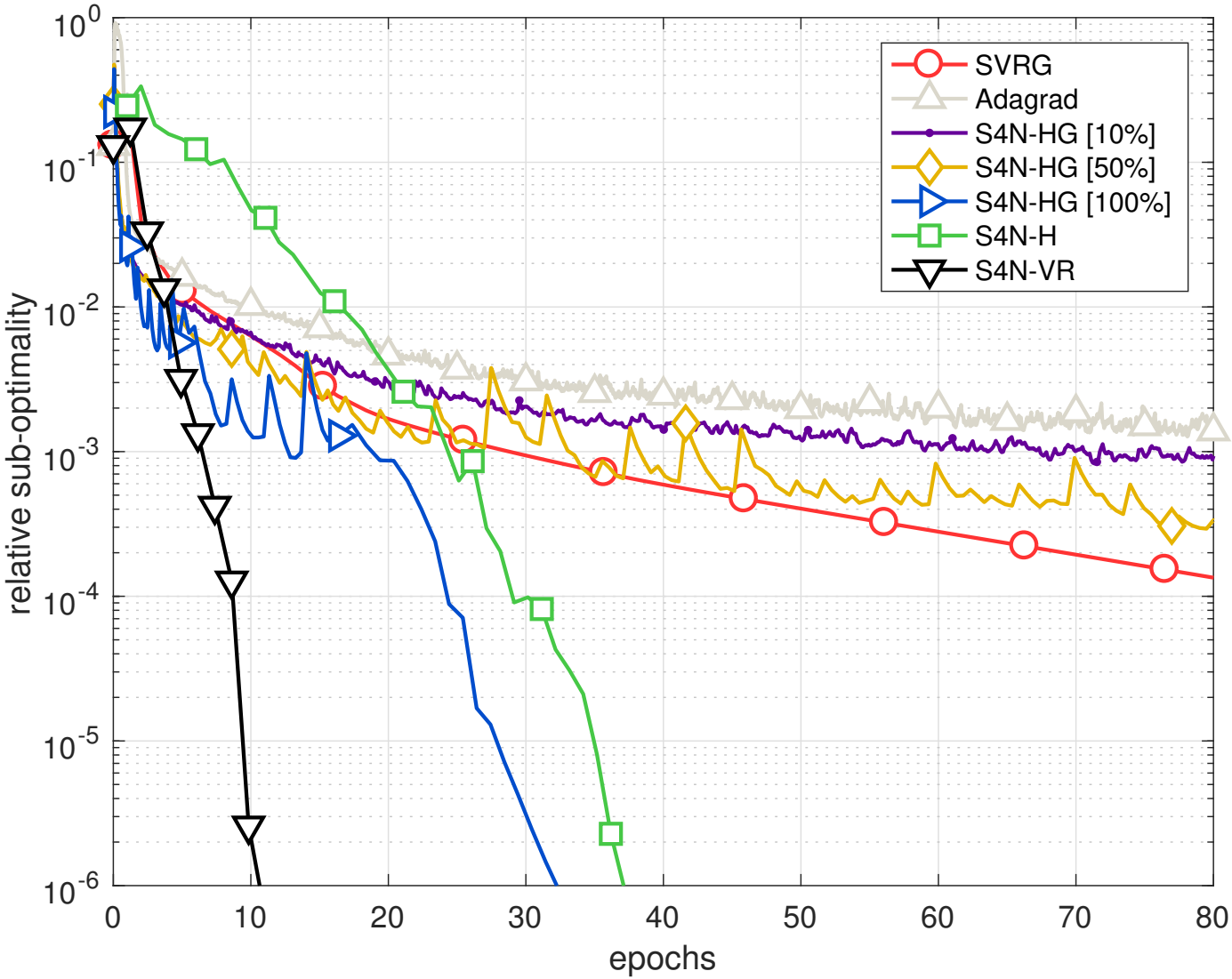
Numerical Comparisons - `covtype`, Epochs



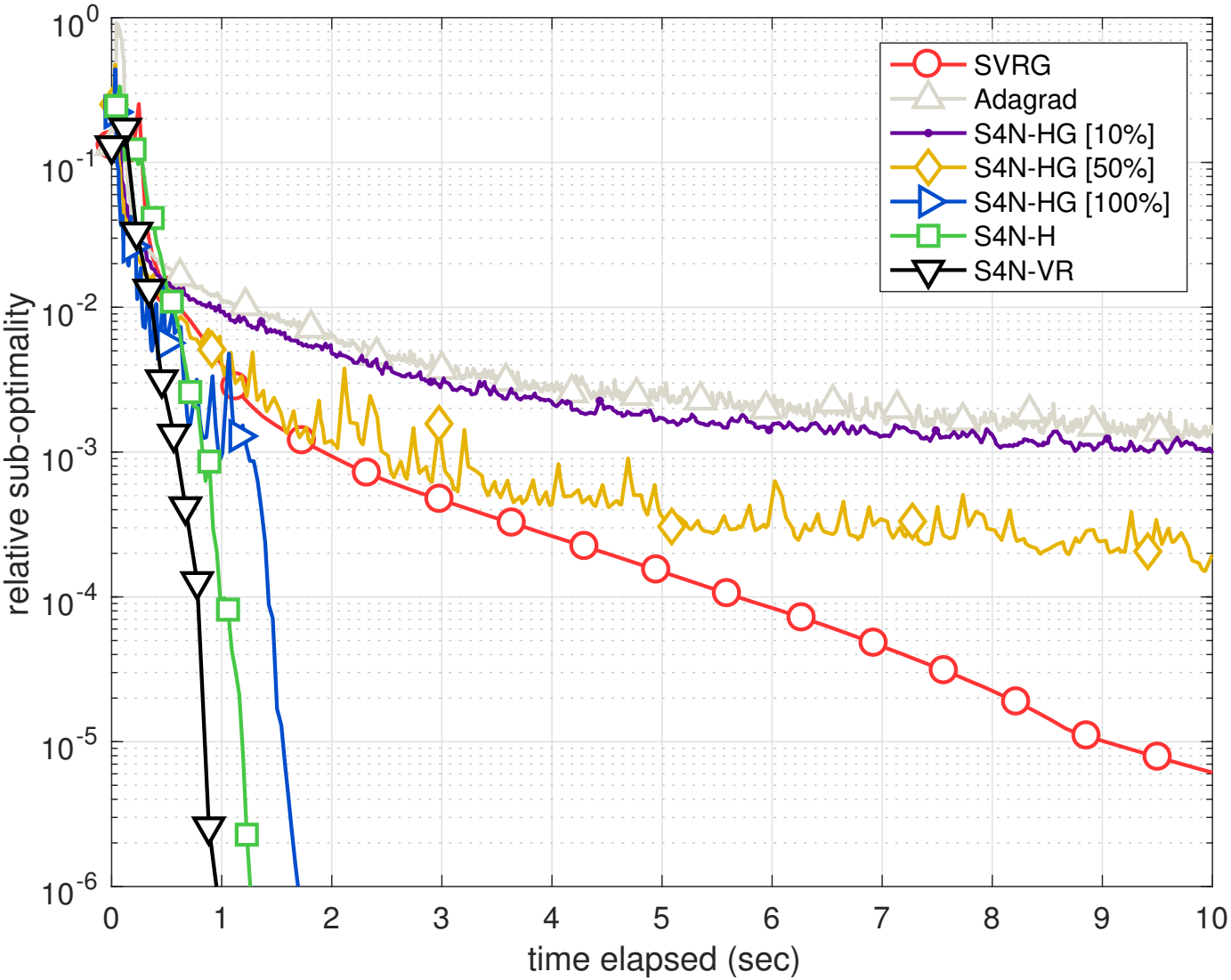
Numerical Comparisons - `covtype`, Time



Numerical Comparisons - `gisette`, Epochs



Numerical Comparisons - `gisette`, Time



Outline

1. Basic Concepts of Semi-smooth Newton method
2. Semi-smooth Newton method for SDP
3. Stochastic Semi-smooth Newton Method
4. Regularized Newton Method for Optimization on Manifold
5. Modified Levenberg-Marquardt Method For Phase Retrieval

Optimization on Riemannian Manifold

Problem definition

$$\min_{\mathbf{x}\in\mathcal{M}} \quad f(x),$$

where $\mathcal{M}$ is a Riemannian manifold.

- Common matrix manifolds
  - Stiefel Manifold: $St(p,n) \triangleq \{X \in \mathbb{R}^{n\times p} \mid X^\top X = I_p\}$
  - Grassmann manifold: $Grass(p,n)$ denote the set of all p–dimensional subspaces of $\mathbb{R}^n$
  - Oblique manifold: $\{X \in \mathbb{R}^{n\times p} \mid \mathrm{diag}(X^T X) = I_p\}$
  - Rank-p manifold: $\{X \in \mathbb{R}^{m\times n} : \mathrm{rank}(X) = p\}$

Electronic Structure Calculation

- Total energy minimization:

$$\min_{X^*X=I} E_{kinetic}(X) + E_{ion}(X) + E_{Hartree}(X) + E_{xc}(X) + E_{fock}(X),$$

where

$$\begin{aligned}
E_{kinetic}(X) &= \frac{1}{2}\mathrm{tr}(X^*LX) \\
E_{ion}(X) &= \mathrm{tr}(X^*V_{ion}X) \\
E_{Hartree}(X) &= \frac{1}{2}\rho(X)^\top L^\dagger \rho(X) \\
E_{xc}(X) &= \rho(X)^\top \mu_{xc}(\rho(X)) \\
\rho(X) &= \mathrm{diag}(D(X)), \quad D(X) = XX^* \\
E_{fock}(X) &= \langle V(D)X, X\rangle
\end{aligned}$$

- Nonlinear eigenvalue problem (looks like the KKT condtions):

$$\begin{aligned}
H(X)X &= X\Lambda \\
X^*X &= I
\end{aligned}$$

Bose-Einstein condensates

Minimization problem

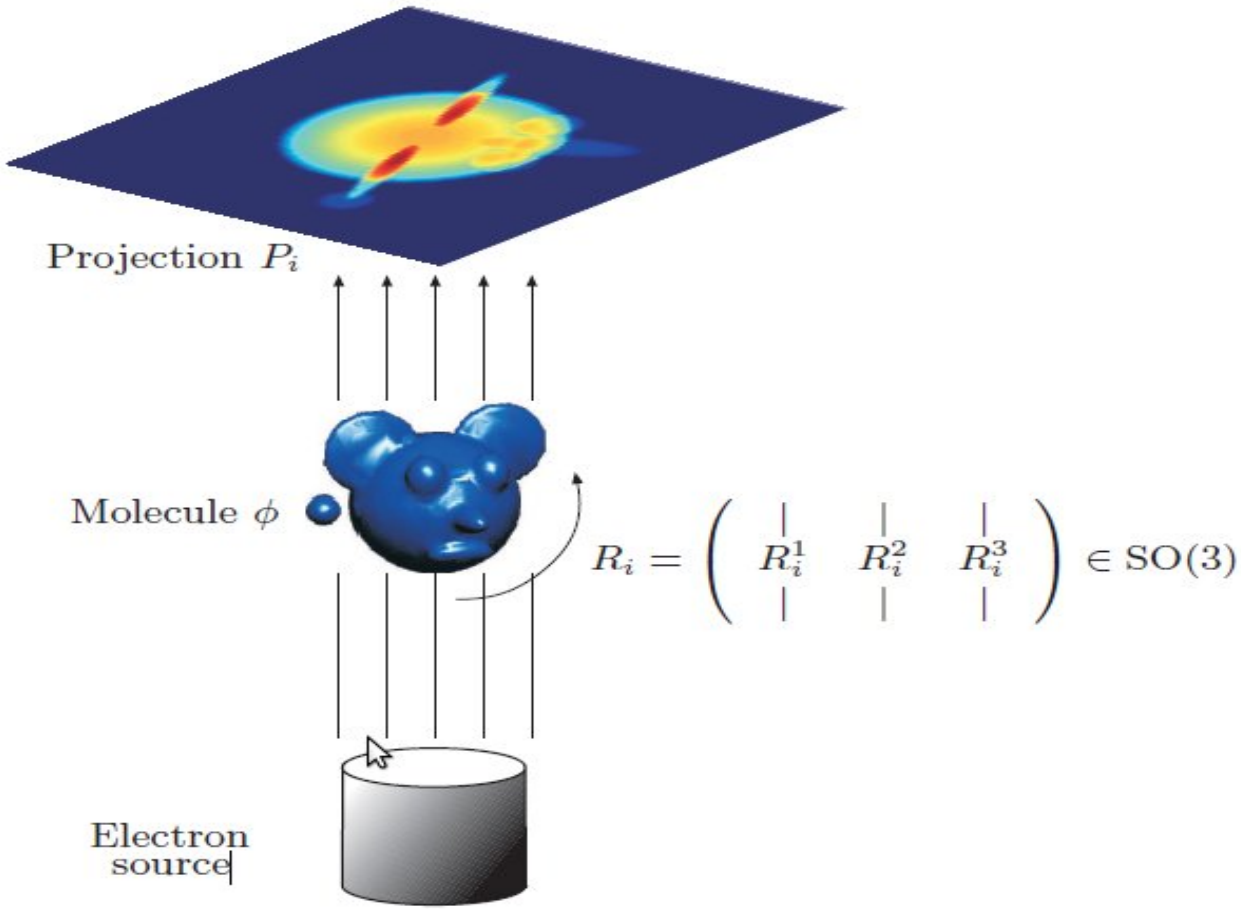
$$\min_{\phi \in S} E(\phi),$$

$$\begin{aligned} E(\phi) &= \int_{\mathbb{R}^d} \left[\frac{1}{2} |\nabla \phi(\mathbf{x})|^2 + V_d(\mathbf{x}) |\phi(\mathbf{x})|^2 + \frac{\beta_d}{2} |\phi(\mathbf{x})|^4 - \Omega \bar{\phi}(\mathbf{x}) L_z \phi(\mathbf{x}) \right] d\mathbf{x} \\ S &= \left\{ \phi \mid E(\phi) < \infty, \int_{\mathbb{R}^d} |\phi(\mathbf{x})|^2 d\mathbf{x} = 1 \right\}. \end{aligned}$$

Discretized problem

$$\min_{X \in \mathbf{C}^N} \mathcal{F}(X) := \frac{1}{2} X^* A X + \alpha \sum_{i=1}^{N} |X_i|^4, \text{ s.t. } \|X\|_F = 1.$$

Cryo-electron microscopy reconstruction



Find 3D structure given samples of 2D projections images

Thanks: Amit Singer

Regularized Newton Method

- Our new adaptively regularized Newton (ARNT) method:

$$\begin{cases} \min & m_k(x) := \langle \nabla f(x_k), x - x_k \rangle + \frac{1}{2} \langle H_k[x - x_k], x - x_k \rangle + \frac{\sigma_k}{2} \|x - x_k\|^2, \\ \text{s.t.} & x \in \mathcal{M}, \end{cases}$$

  where $\nabla f(x_k)$ and H_k are the Euclidean gradient Hessian.

- Regularized parameter update (trust-region-like strategy):
  - ratio: $\rho_k = \frac{f(z_k) - f(x_k)}{m_k(z_k)}$.
  - regularization parameter σ_k:

$$\sigma_{k+1} \in \begin{cases} (0, \sigma_k] & \text{if } \rho_k > \eta_2, \quad \Rightarrow \boxed{x_{k+1} = z_k} \\ [\sigma_k, \gamma_1 \sigma_k] & \text{if } \eta_1 \le \rho_k \le \eta_2, \quad \Rightarrow \boxed{x_{k+1} = z_k} \\ [\gamma_1 \sigma_k, \gamma_2 \sigma_k] & \text{otherwise.} \quad \Rightarrow \boxed{x_{k+1} = x_k} \end{cases}$$

  where $0 < \eta_1 \le \eta_2 < 1$ and $1 < \gamma_1 \le \gamma_2$.

Solvers for subproblem

- The subproblem implicitly preserve the Lagrangian multipliers

$$\mathrm{Hess} m_k(x_k)[\xi] = \mathbf{P}_{x_k}(H_k[\xi] - U\mathrm{sym}((x_k)^*\nabla f(x_k))) + \tau_k \xi,$$

- Riemannian Gradient method with BB step size.
- Newton system for the subproblem

$$\mathrm{grad}\, m_k(x_k) + \mathrm{Hess} m_k(x_k)[\xi] = 0.$$

- Modified CG method

$$\xi_k = \begin{cases} s_k + \tau_k d_k & \text{if } d_k \neq 0, \\ s_k & \text{if } d_k = 0, \end{cases} \quad \text{with} \quad \tau_k := \frac{\langle d_k, \mathrm{grad}\, m_k(x_k)\rangle_{x_k}}{\langle d_k, \mathrm{Hess} m_k(x_k)[d_k]\rangle_{x_k}}$$

  - d_k represents and transports the negative curvature information
  - s^k corresponds to the "usual" output of the CG method.

Hartree-Fock total energy minimization

Solver	fval	nrmG	its	time	fval	nrmG	its	time
	ctube661				glutamine			
ACE	-1.43e+2	9.2e-7	8(2.8)	795	-1.04e+2	3.9e-7	10(3.0)	229
GBBN	-1.43e+2	6.5e-7	10(26.3)	1399	-1.04e+2	8.4e-7	11(13.3)	256
ARN	-1.43e+2	6.0e-7	9(14.1)	832	-1.04e+2	8.8e-7	10(9.5)	209
ARQN	-1.43e+2	2.0e-7	8(13.2)	777	-1.04e+2	1.5e-7	8(10.1)	182
AKQN	-1.43e+2	6.1e-7	17(10.3)	1502	-1.04e+2	9.1e-7	25(6.0)	515
RQN	-1.43e+2	7.2e-6	59	6509.0	-1.04e+2	2.9e-6	57	1532
	graphene16				graphene30			
ACE	-1.01e+2	7.6e-7	13(3.4)	367	-1.87e+2	8.6e-7	58(4.2)	14992
GBBN	-1.01e+2	4.2e-7	14(42.1)	659	-1.87e+2	8.9e-7	29(72.2)	19701
ARN	-1.01e+2	4.5e-7	14(23.0)	403	-1.87e+2	9.0e-7	45(35.6)	14860
ARQN	-1.01e+2	4.9e-7	11(20.2)	357	-1.87e+2	7.6e-7	15(26.5)	6183
AKQN	-1.01e+2	7.9e-7	49(15.1)	1011	-1.87e+2	8.0e-7	39(12.3)	9770
RQN	-1.01e+2	1.0e-3	74	2978	-1.87e+2	1.5e-5	110	39091

Bose-Einstein condensates

solver	f	its	nrmG	time	f	its	nrmG	time
β =500								
	$\Omega = 0.70$				$\Omega = 0.80$			
OptM	6.9731	340	1.0e-4	56.3	6.1016	386	1.0e-4	65.2
TRQH	6.9731	7(55)	2.0e-4	61.6	6.1016	5(64)	2.0e-4	83.1
ARNT	6.9731	10(99)	8.7e-5	44.4	6.1016	10(104)	8.7e-5	70.6
RTR	6.9731	99(118)	9.3e-5	234.2	6.1016	18(130)	7.7e-5	130.1
	$\Omega = 0.90$				$\Omega = 0.95$			
OptM	4.7784	10000	1.2e-3	243.6	3.7419	10000	7.4e-4	241.6
TRQH	4.7778	277(176)	2.0e-4	1090.9	3.7416	363(181)	2.0e-4	1185.1
ARNT	4.7777	147(132)	9.6e-5	413.3	3.7414	500(147)	2.6e-4	1204.0
RTR	4.7777	500(147)	8.5e-4	1250.4	3.7415	500(172)	9.7e-4	1419.0
β =1000								
	$\Omega = 0.70$				$\Omega = 0.80$			
OptM	9.5283	990	1.0e-4	63.7	8.2627	10000	5.5e-4	231.9
TRQH	9.5301	102(156)	2.0e-4	404.1	8.2610	453(177)	2.0e-4	1427.0
ARNT	9.5301	60(81)	9.3e-5	140.4	8.2610	202(105)	6.7e-5	412.7
RTR	9.5301	293(91)	8.6e-5	478.8	8.2610	500(113)	5.5e-4	972.7
	$\Omega = 0.90$				$\Omega = 0.95$			
OptM	6.3611	10000	3.0e-3	230.8	4.8856	10000	5.2e-4	241.4
TRQH	6.3607	142(170)	2.0e-4	595.6	4.8831	172(178)	2.0e-4	708.1
ARNT	6.3607	500(110)	2.8e-3	931.5	4.8822	500(121)	1.5e-3	1015.8
RTR	6.3607	500(122)	7.6e-4	1010.8	4.8823	500(137)	1.9e-3	1103.8

Outline

1. Basic Concepts of Semi-smooth Newton method
2. Semi-smooth Newton method for SDP
3. Stochastic Semi-smooth Newton Method
4. Regularized Newton Method for Optimization on Manifold
5. Modified Levenberg-Marquardt Method For Phase Retrieval

Phase retrieval

Detectors record intensities of diffracted rays $\Longrightarrow$ phaseless data only!

- Recover x from phaseless measurements about $x \in \mathbf{C}^n$

$$\text{find } x, \text{ s.t. } |Ax| = b.$$

- An equivalent model

$$\min_{x \in \mathbf{C}^n, y \in \mathbb{R}^m} \quad \frac{1}{2}\|Ax - y\|_2^2, \text{ s.t. } |y| = b.$$

Applications: Hubble Space Telescope, X-ray crystallography

Phase retrieval by non-convex optimization

Solve the system of quadratic equations:

$$y_r = |\langle a_r, x\rangle|^2, \quad r = 1, 2, ..., m.$$

- **Gaussian model:**

$$a_r \in \mathbb{C}^n \overset{i.i.d.}{\sim} \mathcal{N}(0, I/2) + i\mathcal{N}(0, I/2).$$

Nonlinear least square problem

$$\min_{z \in \mathbb{C}^n} \quad f(z) = \frac{1}{4m} \sum_{k=1}^{m} (y_k - |\langle a_k, z\rangle|^2)^2$$

f is nonconvex, many local minima

Wirtinger flow: Candes, Li and Soltanolkotabi ('14)

- **Spectral Initialization:**
  1. Input measurements $\{a_r\}$ and observation $\{y_r\}(r = 1, 2, ..., m)$.
  2. Calculate z_0 to be the leading eigenvector of $Y = \frac{1}{m} \sum_{r=1}^{m} y_r a_r a_r^*$.
  3. Normalize z_0 such that $\|z_0\|^2 = n \frac{\sum_r y_r}{\sum_r \|a_r\|^2}$.

- **Iteration via Wirtinger derivatives:** for $\tau = 0, 1, \ldots$

$$z_{\tau+1} = z_\tau - \frac{\mu_{\tau+1}}{\|z_0\|^2} \nabla f(z_\tau)$$

The Modified LM method for Phase Retrieval

Levenberg-Marquardt Iteration:

$$\mathbf{z}_{k+1} = \mathbf{z}_k - \left(\Psi(z_k) + \mu_k I\right)^{-1} g(z_k)$$

Algorithm

1. **Input:** Measurements $\{a_r\}$, observations $\{y_r\}$. Set $\epsilon \geq 0$.
2. Construct z_0 using the spectral initialization algorithms.
3. **While** $\|g(z_k)\| \geq \epsilon$ **do**
   - Compute s_k by solving equation

$$\Psi_{z_k}^{\mu_k} s_k = \left(\Psi(z_k) + \mu_k I\right) s_k = -g(z_k).$$

     until

$$\left\|\Psi_{z_k}^{\mu_k} s_k + g(z_k)\right\| \leq \eta_k \|g(z_k)\|.$$

   - Set $\mathbf{z}_{k+1} = \mathbf{z}_k + s_k$ and $k := k + 1$.
3. **Output:** z_k.

Convergence of the Gaussian Model

Theorem

If the measurements follow the Gaussian model, the LM equation is solved accurately ($\eta_k = 0$ for all k), and the following conditions hold:

- $m \geq cn \log n$, *where c is sufficiently large;*
- *If* $f(z_k) \geq \frac{\|z_k\|^2}{900n}$, *let* $\mu_k = 70000n\sqrt{nf(z_k)}$; *if else, let* $\mu_k = \sqrt{f(z_k)}$.

Then, with probability at least $1 - 15e^{-\gamma n} - 8/n^2 - me^{-1.5n}$, *we have* $dist(z_0, x) \leq (1/8)\|x\|$, *and*

$$\mathrm{dist}(z_{k+1}, x) \leq c_1 dist(z_k, x),$$

Meanwhile, once $f(z_s) < \frac{\|z_s\|^2}{900n}$, *for any* $k \geq s$ *we have*

$$dist(z_{k+1}, x) < c_2 dist(z_k, x)^2.$$

Key to proof

Lower bound of GN matrix's second smallest eigenvalue

For any $y,\ z \in \mathbb{C}^n$, $Im(y^* z) = 0$, we have:

$$\mathbf{y}^* \Psi(z) \mathbf{y} \geq \|y\|^2 \|z\|^2,$$

holds with high probability.

$$Im(y^* z) = 0 \quad \Rightarrow \quad \|(\Psi_z^\mu)^{-1} \mathbf{y}\| \leq \frac{2}{\|z\|^2 + \mu} \|\mathbf{y}\|.$$

Key to proof

Local error bound property

$$\frac{1}{4}\mathrm{dist}(z,x)^2 \le f(z) \le 8.04\mathrm{dist}(z,x)^2 + 6.06n\mathrm{dist}(z,x)^4,$$

holds for any z satisfying $\mathrm{dist}(z,x) \le \frac{1}{8}$.

Regularity condition

$$\mu(z)\mathbf{h}^*\left(\Psi_z^\mu\right)^{-1} g(\mathbf{z}) \ge \frac{1}{16}\|\mathbf{h}\|^2 + \frac{1}{64100n\|h\|}\|g(\mathbf{z})\|^2$$

holds for any $z = x + h$, $\|h\| \le \frac{1}{8}$, and $f(z) \ge \frac{\|z\|^2}{900n}$.

Numerical Result: Natural Image

Figure: The Milky Way Galaxy. Image size is 1080×1920 pixels. For the ALM method, the CPU time is 20240.64s, with a final relative error to be 2.44×10^{-16}; for the ILM algorithm, the CPU time is 4733.43s, and the final relative error is 2.42×10^{-16}; for WF algorithm, the CPU time is 5211.35s, while the final relative error is 4.91×10^{-16}

Numerical Result: Natural Image

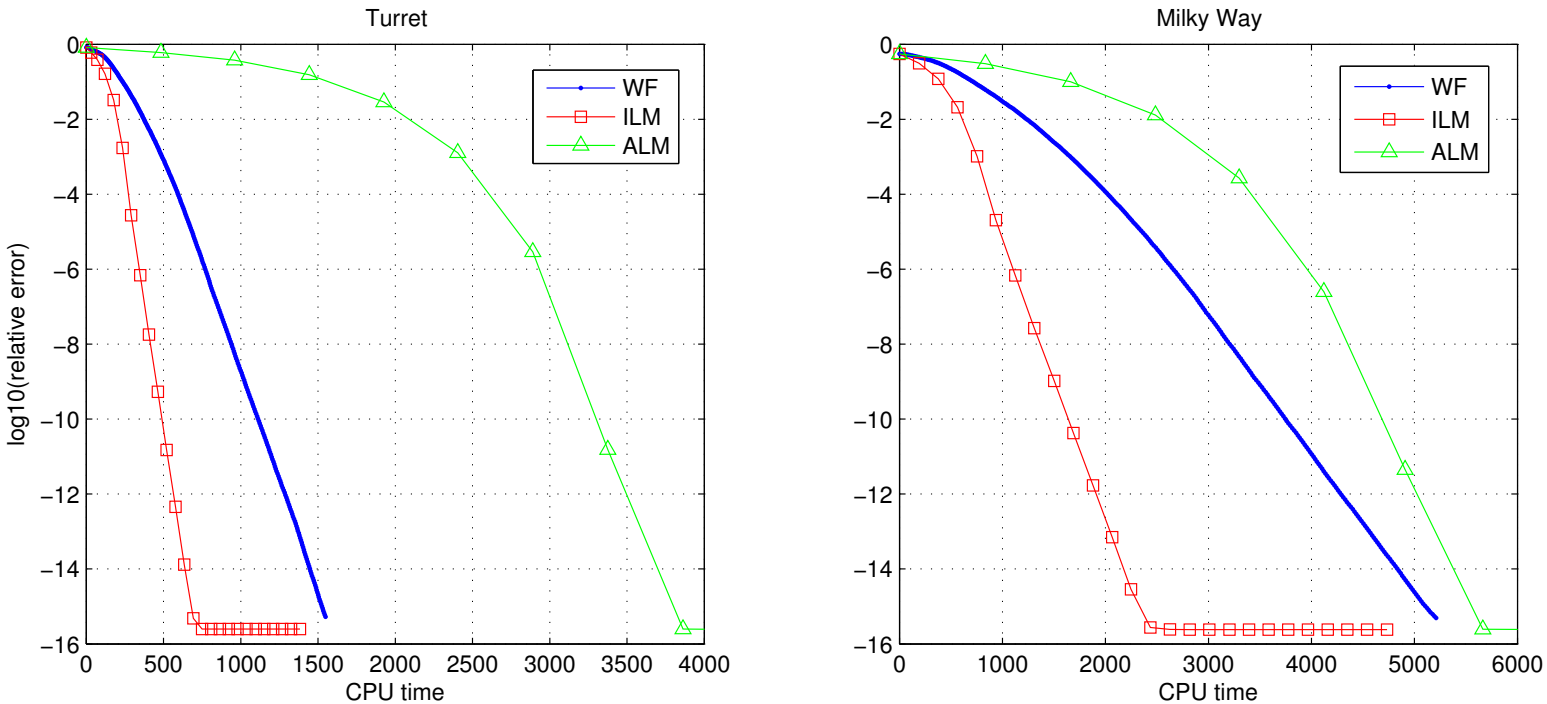


Figure: Relation between relative error and CPU time used for natural images recovery.

Contact Information

Many Thanks For Your Attention!

- Looking for Ph.D students and Postdoc
  Competitive salary as U.S and Europe
- `http://bicmr.pku.edu.cn/~wenzw`
- E-mail: wenzw@pku.edu.cn
- Office phone: 86-10-62744125