Semi-smooth Newton Methods for Semi-definite Programming and Stochastic Optimization

Zaiwen Wen

Beijing International Center For Mathematical Research
Peking University

2018 International Workshop
on Modern Optimization and Applications

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`

Outline

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

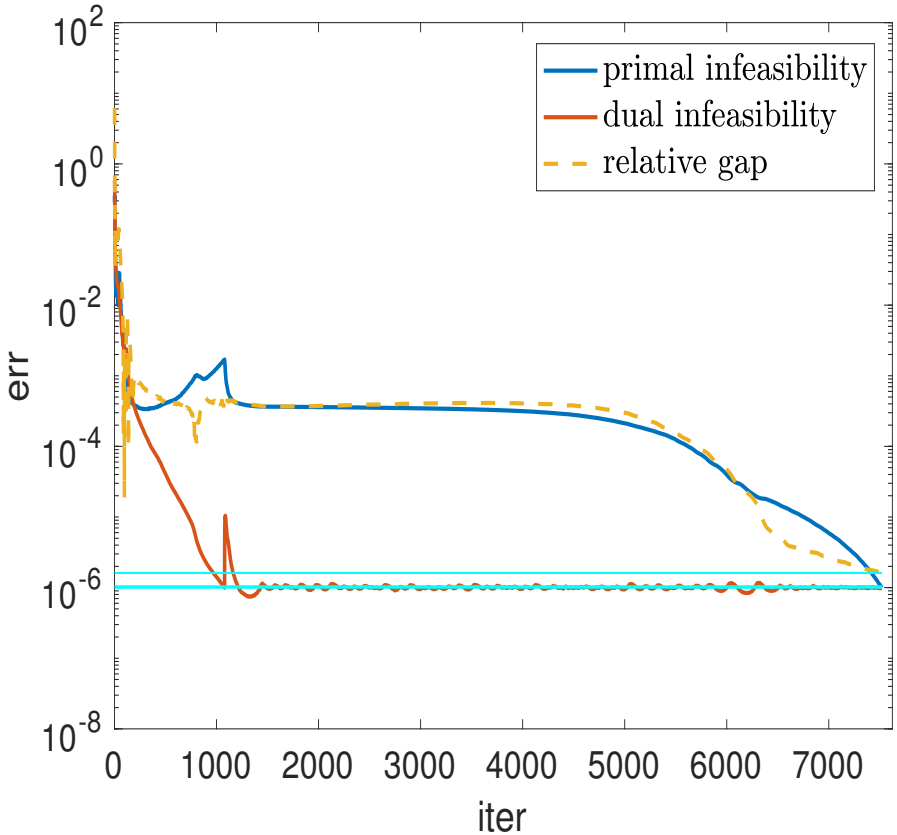
Goal: study approaches to bridge the gap between first-order and second-order type methods for composite convex programs.

key observations:

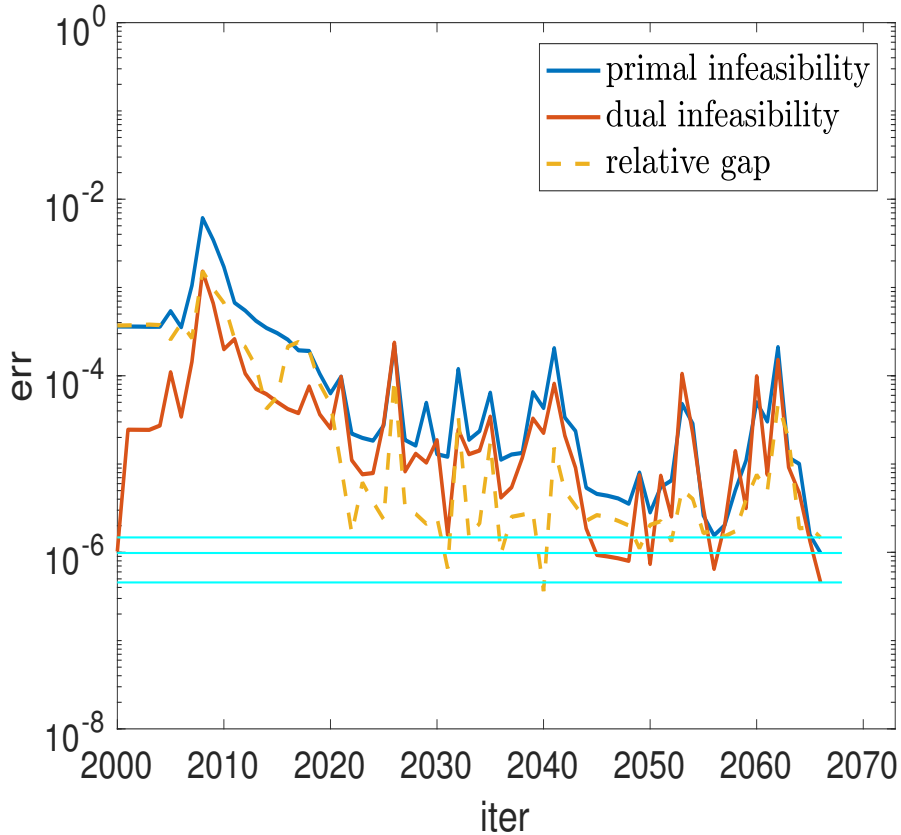
- Many popular first-order methods can be equivalent to some fixed-point iterations: $x^{k+1} = T(x^k)$;
  - **Advantages:** easy to implement; converge fast to a solution with moderate accuracy.
  - **Disadvantages:** slow tail convergence.
- The original problem is equivalent to the system $F(x) := (I - T)(x) = 0$.
- 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



(b) Semi-smooth Newton

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

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.

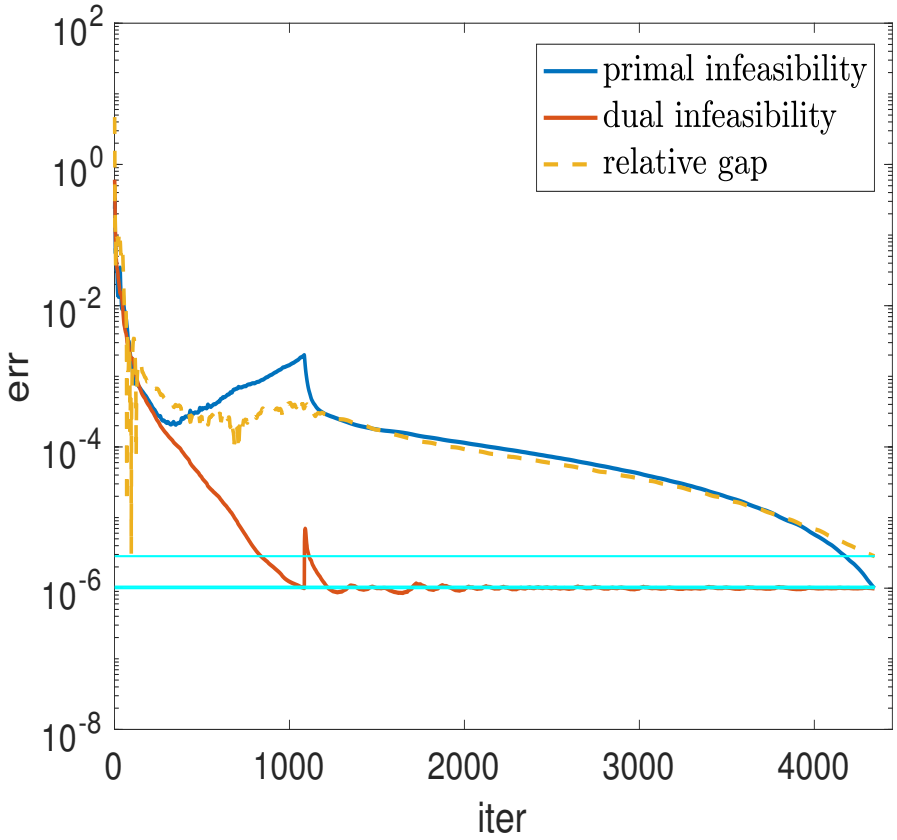
Comparison on electronic structure calculation

- 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:

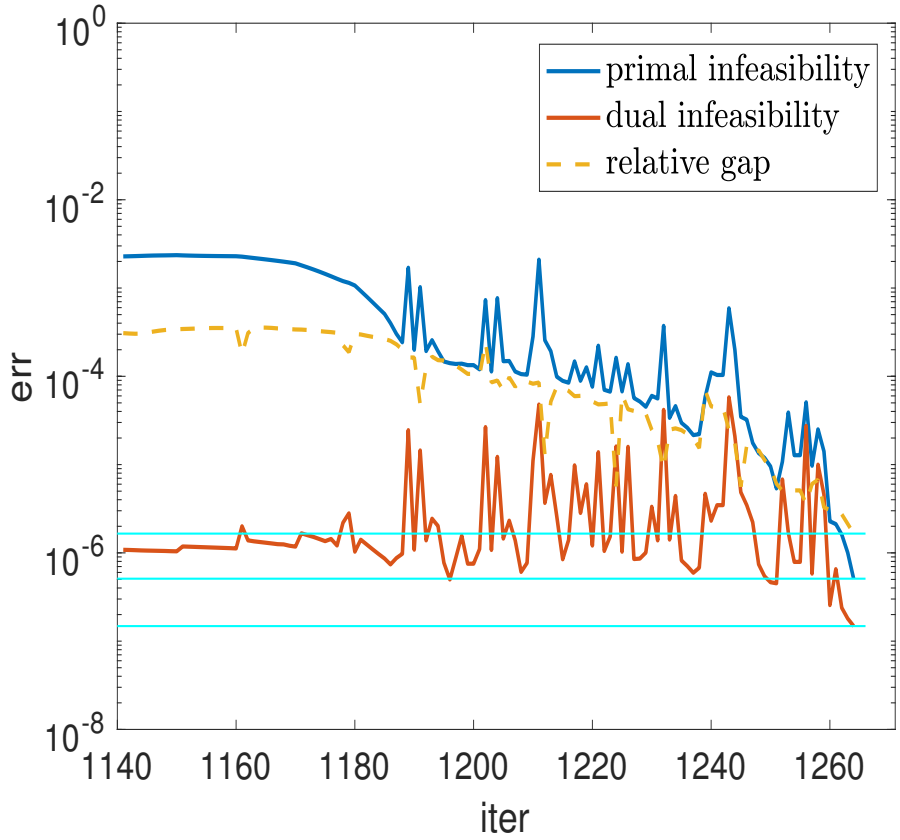
$$\eta_p = \frac{\|\mathcal{A}(X) - b\|_2}{\max(1, \|b\|_2)} \quad \eta_d = \frac{\|\mathcal{A}^* y - C - S\|_F}{\max(1, \|C\|_F)}$$

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

Compuational Results: C2

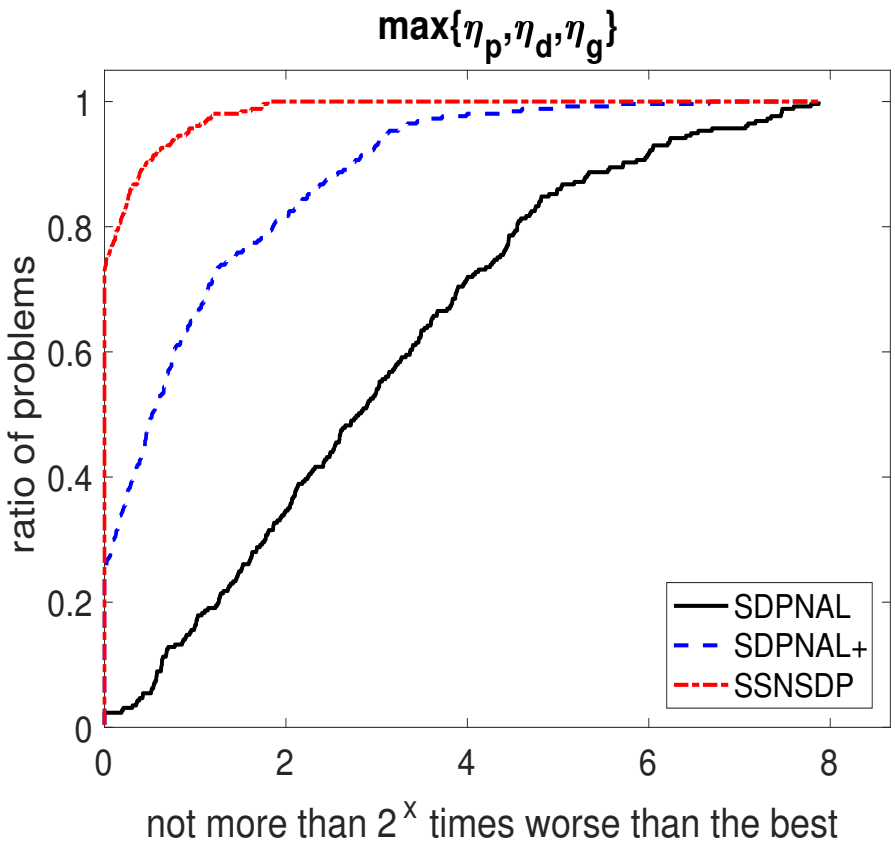


(c) ADMM

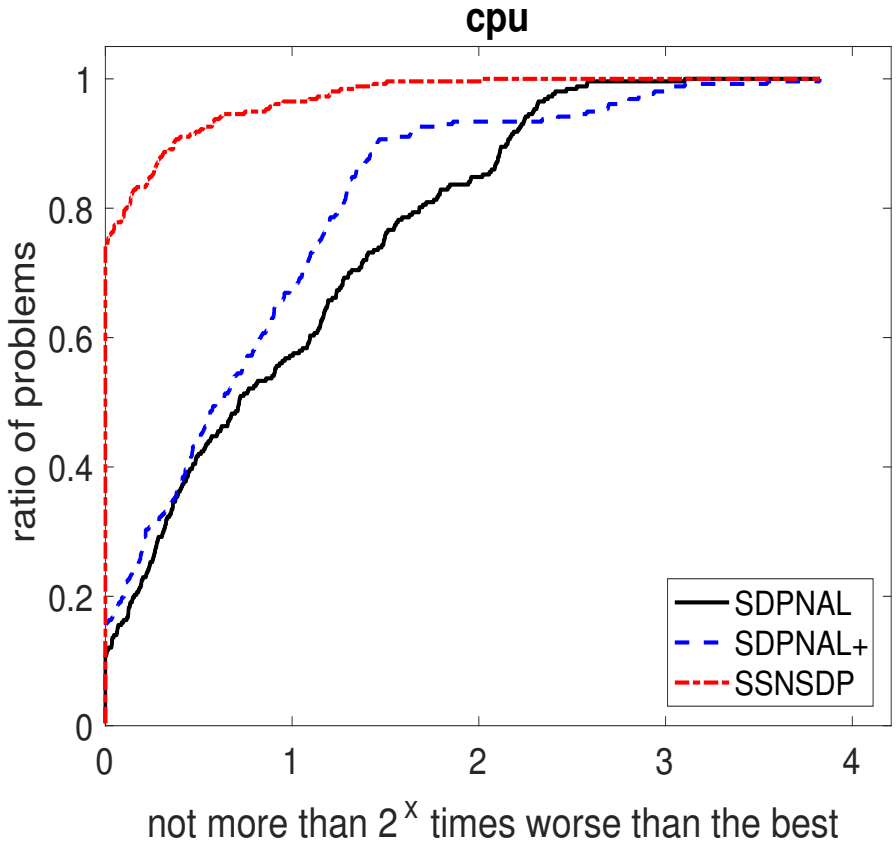


(d) Semi-smooth Newton

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

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!

First-Order Optimality Conditions

First Order Optimality Conditions for $\min_x \psi(x)$:

A point $x \in \operatorname{dom} h$ is a stationary point iff

$$0 \in \nabla f(x) + \partial h(x),$$

where $\partial h(x)$ is the convex subdifferential of h at x.

The optimality conditions can be rewritten as follows:

Optimality Conditions as Nonsmooth Equation:

$$F^\Lambda(x) := x - \operatorname{prox}_h^\Lambda(x - \Lambda^{-1}\nabla f(x)) = 0,$$

where $\mathbb{R}^{n\times n} \ni \Lambda \succ 0$ is symmetric and positive definite and $\operatorname{prox}\Lambda h$ is the proximity operator [*Moreau* '65]:

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

Algorithmic Background

- In [*Fukushima, Mine* '81; *Tseng, Yun* '09] $-F^\Lambda(x)$ is used as a descent direction and an Armijo-type line search is included to ensure global convergence.

$\rightsquigarrow$ This defines the basic proximal gradient method.

- In our setting, f and ∇f have to be estimated or sub-sampled.
- General idea of first-order stochastic optimization methods: use a stochastic oracle ($\mathcal{SFO}$) to estimate the gradient:

$$G(x^k; s^k) \approx \nabla f(x^k),$$

where s^k is a collection of random variables or samples.

$\rightsquigarrow$ This leads to stochastic variants of the mapping F^Λ:

$$F_s^\Lambda(x) = x - \operatorname{prox}_h^\Lambda(x - \Lambda^{-1} G(x; s))$$

and to different stochastic proximal gradient methods.

Related Work and Literature

First Order Methods:

- [*Robbins, Monro* '51]: Foundations of the classical SGD.
- [*Polyak* '90; *Nemirovski et al.* '09; *Friedlander, Schmidt* '12], [...]
- [*Ghadimi, Lan (et al.)* '13, '16, '16]: (Accelerated) Proximal gradient schemes for nonconvex problems.
- [*Xiao, Zhang* '14; *Reddi et al.* '16]: Proximal SVRG and SAGA.
- [*Xu, Yin* '15]: Block prox-SGD for nonconvex problems.

Quasi-Newton Methods:

- [*Schraudolph et al.* '07; *Mokhtari, Ribeiro* '14, '15]: Sub-sampled (L)BFGS for online optimization.
- [*Byrd et al.* '11, '16; *Gower et al.* '16]: Stochastic Quasi-Newton.
- [*Wang et al.* '17]: Stochastic Quasi-Newton for nonconvex prob.

Related Work and Literature (Cont')

Second Order Methods:

- [*Agarwal et al.* '16]: LiSSA; Hessian sampling for convex prob.
- [*Bollapragada et al.* '16]: Sub-sampled Newton; convergence in expectation; f_i strongly convex.
- [*Xu et al.* '16]: Sub-sampled Newton with nonuniform sampling.
- [*Roosta-K., Mahoney* '16; *Xu et al.* '17, '17]: Sub-sampled Newton; convergence results in probability.
- [*Pilanci, Wainwright* '17]: Newton sketch.
- [*Ye et al.* '17]: Local conv. of approximate Newton methods.

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:

$$\|F^{\Lambda_{k+1}}_{s^{k+1}}(z^k)\| \le (\eta + \nu_k)\cdot\theta_k + \varepsilon^1_k, \tag{G.1}$$

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

where $\eta \in (0,1)$, $\beta > 0$, and $(\nu_k), (\varepsilon^2_k) \in \ell^1_+$, $(\varepsilon^1_k) \in \ell^{1/2}_+$.

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

Remark:

- Calculating $F^{\Lambda_{k+1}}_{s^{k+1}}(z^k)$ 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 $\Lambda \in \mathbb{S}^n_{++}$.

Assumptions for Local Convergence

$\rightsquigarrow$ We study local conv. of a sequence of iterates (trajectory) (x^k) of a single run of **S4N**.

Assumptions:

Let x^* and Λ_* be accumulation points of (x^k) and (Λ_k). Let us assume that there exists $\bar{\varepsilon} > 0$ such that:

(C.1) There exists $\bar{k} \in \mathcal{N}$ such that $\Lambda_k = \Lambda_*$ for all $k \geq \bar{k}$.

(C.2) There exist $\nu_*, K_* > 0$ such that

$$\lambda_{\min}(\nabla^2 f(x)) \geq \nu_*, \quad \lambda_{\max}(\nabla^2 f(x)) \leq K_*, \quad \forall\ x \in B_{\bar{\varepsilon}}(x^*).$$

(C.3) ψ is Lipschitz continuous on $B_{\bar{\varepsilon}}(x^*)$ with constant L_ψ.

(C.4) The residual mapping F^{Λ_*} is semismooth at x^*.

- (C.2) is satisfied if the Hessian $\nabla^2 f(x^*)$ is positive definite.

Refined Stochastic Assumptions

We define the error terms

$$\mathcal{E}^{\mathrm{g}}_{s_i^k}(x) := \nabla f(x) - G(x; s_i^k), \quad \mathcal{E}^{\mathrm{h}}_{t_j^k}(x) := \nabla^2 f(x) - H(x; t_j^k).$$

Stochastic Assumptions (Extended):

We now assume that the oracles $G(\cdot; s_i^k)$ and $H(\cdot; t_j^k)$ are unbiased estimators of the gradient and Hessian, i.e., for all i, j, and k,

$$\mathbb{E}[\mathcal{E}^{\mathrm{g}}_{s_i^k}(x)] = 0, \quad \mathbb{E}[\mathcal{E}^{\mathrm{h}}_{t_j^k}(x)] = 0, \quad \forall\, x \in \mathbb{R}^n.$$

(S.3) There are $\bar{\sigma}, \bar{\rho}$, such that for all i, j, k and $x \in \mathbb{R}^n$, it holds

$$\mathbb{E}[\exp(\|\mathcal{E}^{\mathrm{g}}_{s_i^k}(x)\|^2/\bar{\sigma}^2)] \le e, \quad \mathbb{E}[\exp(\|\mathcal{E}^{\mathrm{h}}_{t_j^k}(x)\|^2/\bar{\rho}^2)] \le e.$$

Theorem: Transition to Fast Convergence [*MXCW*, '17]

Setup: Let $(\delta_k) \subset (0,1)$ be given. Let us set $C := (2\lambda_M + 3K_*)/\nu_*$,

$$\mathcal{R}_k := \min\{\varepsilon_k^1/(2L_F C), [\varepsilon_k^2]^2\}, \quad \Gamma_k := \min\{\mathcal{R}_{k-1}, \mathcal{R}_k\}.$$

- Let the cond. (A.1)–(A.3), (S.2)–(S.3), and (C.1)–(C.4) be satisfied for some acc. points x^*, Λ_* of (x^k) and (Λ_k).
- Assume that the step sizes (α_k) are bounded $\alpha_k \in [\underline{\alpha}, \overline{\alpha}]$.

Then, there exists a constant $\gamma \in (0,1)$ that does not depend on (ε_k^1) and (ε_k^2) such that, if

$$\mathbf{n}_k^{\mathrm{g}} \geq \left[\left(1 + \sqrt{3\log(\delta_k^{-1})}\right)\frac{2\bar{\sigma}}{\lambda_m \Gamma_k}\right]^2, \quad \mathbf{n}_k^{\mathrm{h}} \geq \frac{3\log(2n\delta_k^{-1})\bar{\rho}^2}{\gamma^2},$$

for all $k \geq \bar{\ell}$, we have:

Transition to Fast Convergence (Cont')

- The point x^* is a stationary point,
- There exists ℓ_* such that x^k results from a stoch. semi-smooth Newton step for all $k \geq \ell_*$,
- The whole sequence (x^k) converges to x^*,

with probability $\delta_* := \prod_{k=\bar{\ell}}^{\infty}(1-2\delta_k)(1-\delta_k)$.

Remarks:

- Based on concentration ineq. for vector- and matrix-valued martingales [*Juditsky, Nemirovski* '09; *Tropp* '12; ...]
- If the batch size $\mathbf{n}_k^{\mathrm{g}}$ increases geometrically, i.e., if for some $\eta_\gamma \in (0,1)$, we redefine $\Gamma_k := \min\{\mathcal{R}_{k-1}, \mathcal{R}_k\}$ and

$$\mathcal{R}_k := \min\{\min\{\varepsilon_k^1, \eta_\gamma^{k-\bar{\ell}}\}/(2L_F C), [\varepsilon_k^2]^2\},$$

then (x^k) converges r-linearly to x^* with rate η_γ.

Further Remarks and an Example

Remarks (Cont'):

- ⇝ Extension to r-superlinear convergence is possible!
- ⇝ If the full gradient is used eventually, we can obtain q-linear and q-superlinear convergence with high probability.

Example:

Let $c_1, c_2, \varpi > 0$ be given and let us set $\varepsilon_1^k = c_1 \cdot k^{-(2+\frac{\varpi}{4})}$ and $\varepsilon_2^k = c_2 \cdot k^{-(1+\frac{\varpi}{8})}$ for all $k \in \mathcal{N}$. Then, setting

$$\mathbf{n}_k^{\mathrm{g}} = k^{4+\varpi} \log(k) \quad \text{and} \quad \mathbf{n}_k^{\mathrm{h}} = \log(k)^{1+\varpi},$$

the local convergence results hold with probability $\delta_* \geq 99\%$.

Numerical Results: Sparse Logistic Regression

We consider the following ℓ_1-regularized logistic regression problem

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

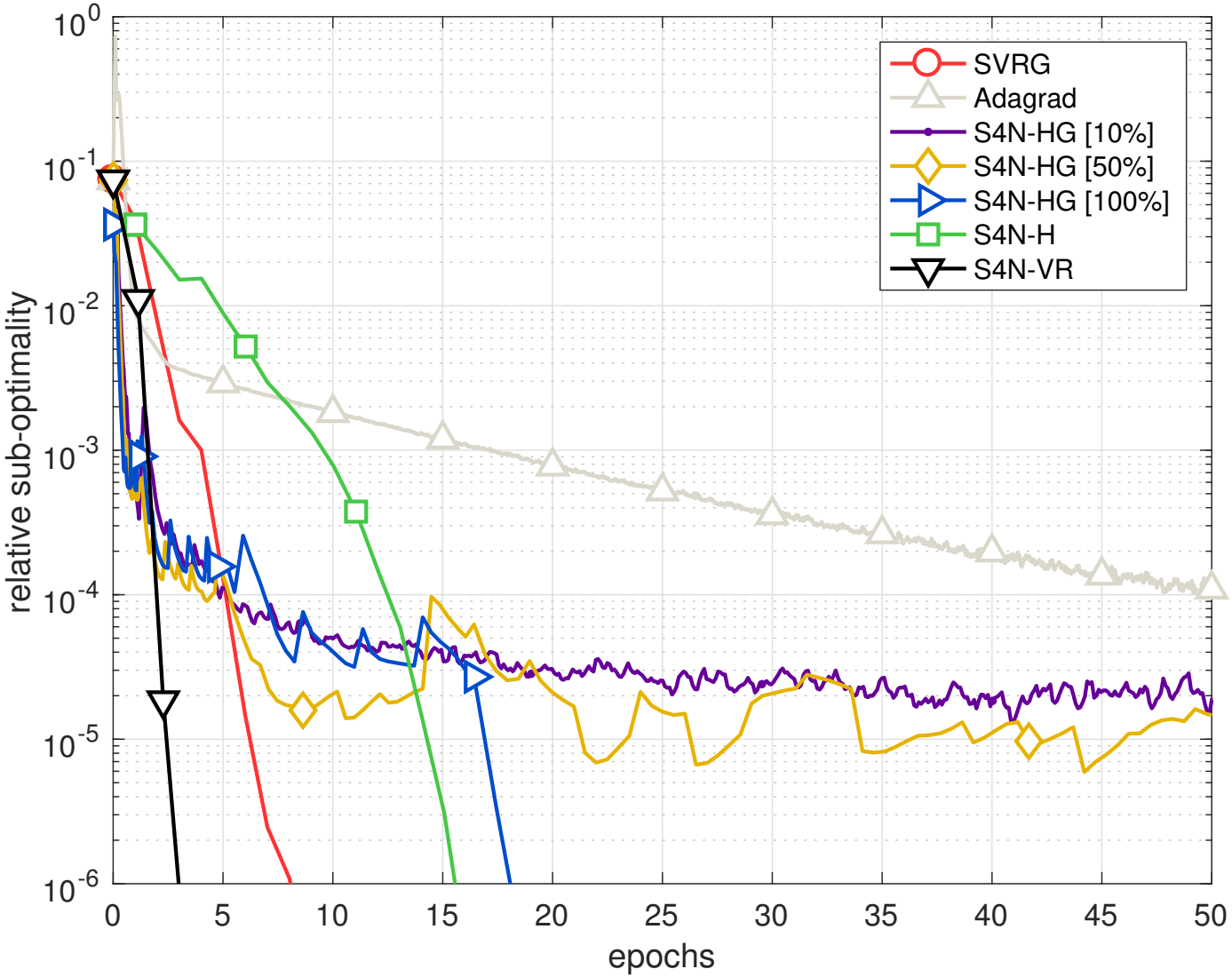
where $a_i^\top \in \mathbb{R}^n$ denotes the ith row of the data matrix $A \in \mathbb{R}^{N\times n}$ and $b \in \{-1,1\}^N$ is a binary vector.

Specifications of the test framework:[1]

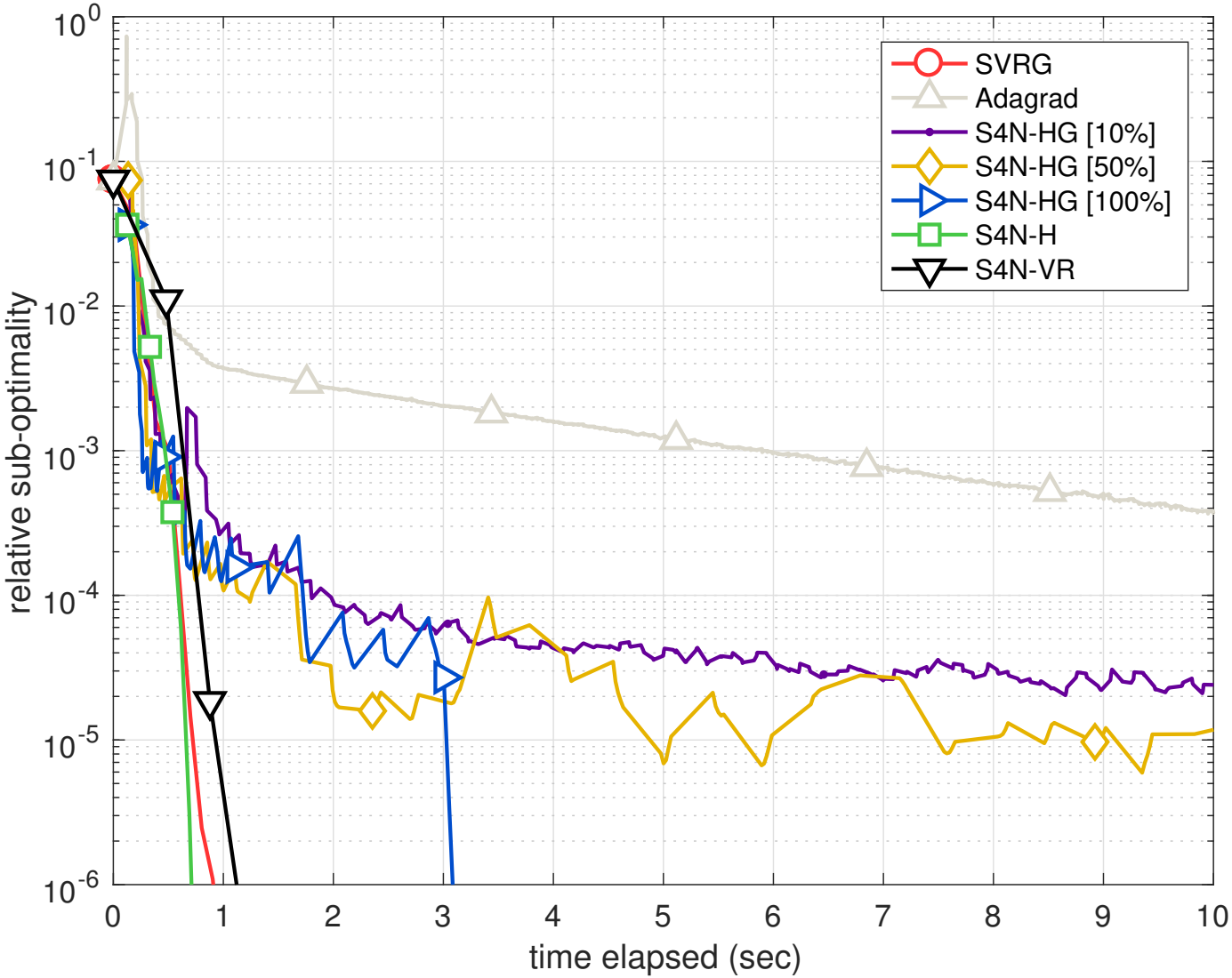
dataset	data points N	features n	
covtype	581 012	54	$\mu =$ 5e-3
gisette	6 000	5 000	$\mu =$ 5e-2
rcv1	20 242	47 236	$\mu =$ 1e-3

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

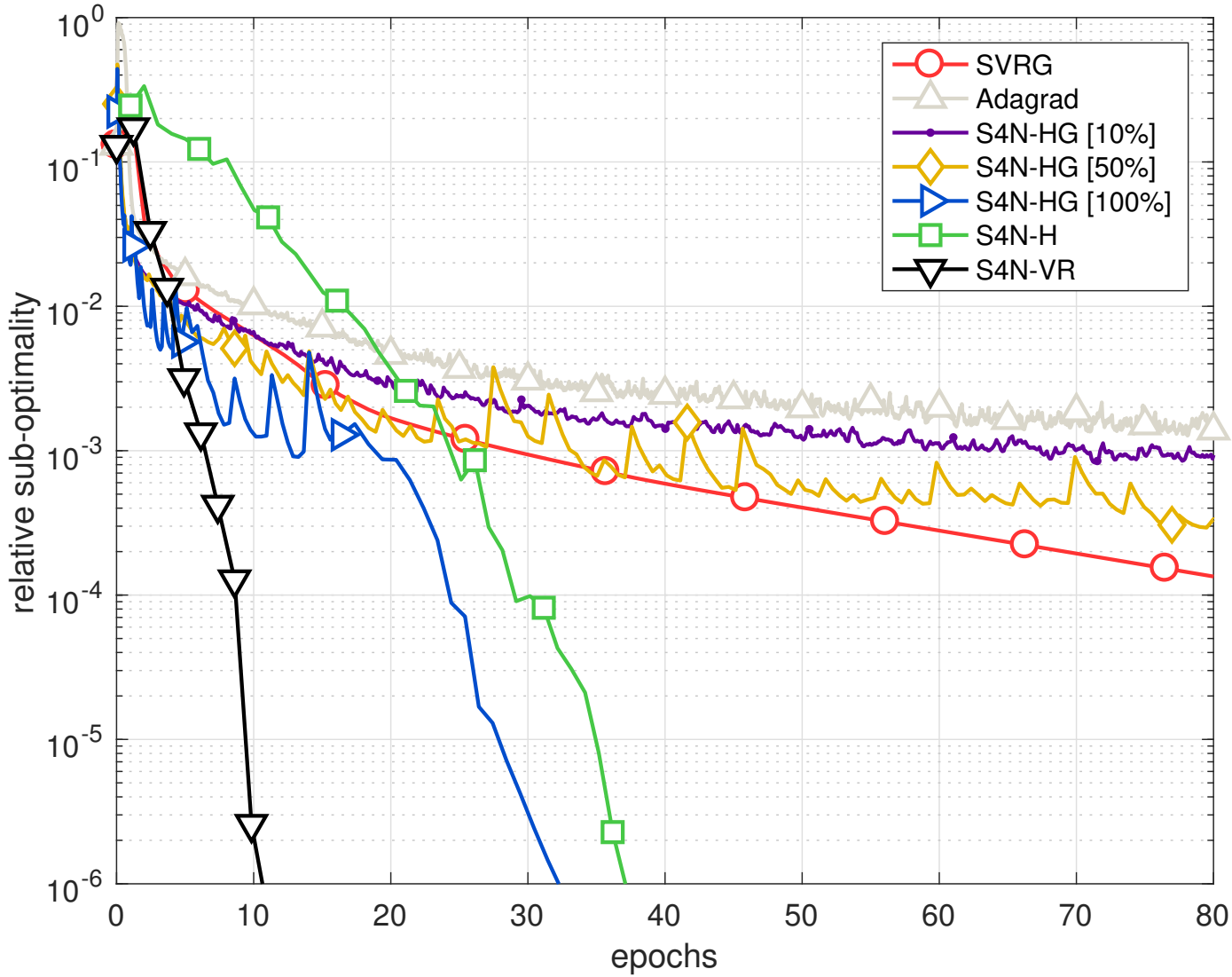
Numerical Comparisons - `covtype`, Epochs



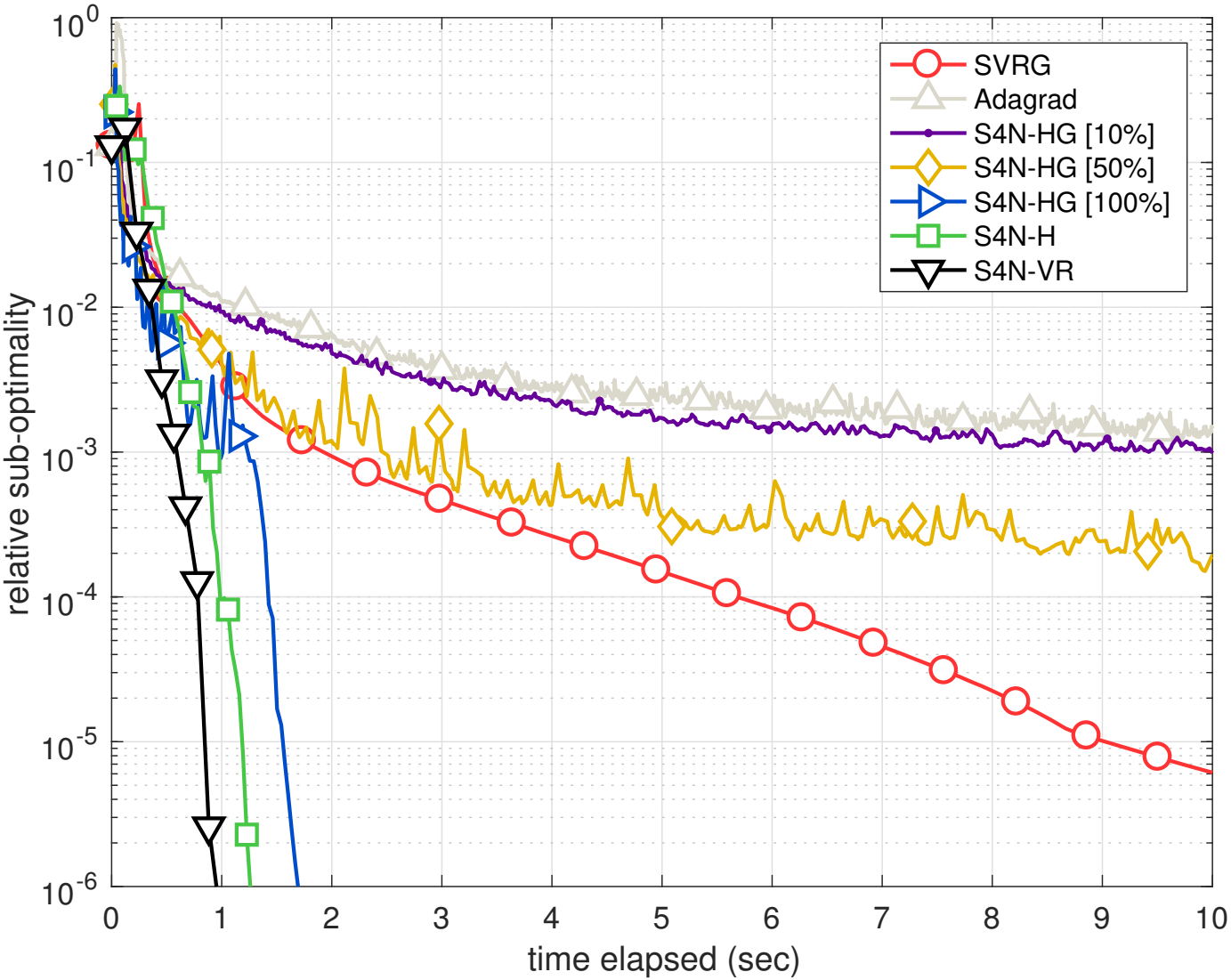
Numerical Comparisons - `covtype`, Time



Numerical Comparisons - `gisette`, Epochs



Numerical Comparisons - `gisette`, Time



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