Variance-Reduced Stochastic Quasi-Newton Methods for Decentralized Learning

Jiaojiao Zhang

Department of Systems Engineering and Engineering Management,
The Chinese University of Hong Kong

Joint work with
Huikang Liu (ICL), Anthony Man-Cho So (CUHK) and Qing Ling (SYSU)

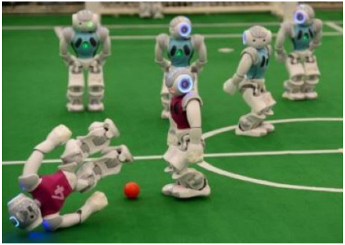
SIAM Conference on Optimization (OP21)

Outline

Background

Machine learning over networks

- large-scale learning
- privacy-preserving learning
- decentralized system control



Background

- Consider the decentralized convex optimization problem

$$x^* = \arg\min_{x \in \mathbb{R}^d} F(x) \triangleq \frac{1}{n}\sum_{i=1}^{n} f_i(x), \quad \text{where } f_i(x) \triangleq \frac{1}{m_i}\sum_{l=1}^{m_i} f_{i,l}(x).$$

$f_{i,l}$ is the l-th sample cost on node i, assumed to be convex and differentiable.

- We consider the case that m_i is very **large**.
- An **equivalent decentralized** formulation is

$$\mathrm{x}^* = \arg\min_{\mathrm{x}=[x_1;\cdots;x_n]\in\mathbb{R}^{nd}} f(\mathrm{x}) \triangleq \frac{1}{n}\sum_{i=1}^{n} f_i\left(x_i\right), \text{ s.t. } x_i = x_j, \ \forall j \in \mathcal{N}_i, \ \forall i.$$

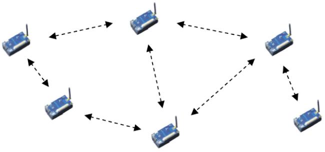


Fig 1. Decentralized network

- Data are distributed across a connected network of n nodes.
- Each node is only allowed to send/receive information to/from its neighboring nodes.
- All the nodes cooperate to obtain a common variable x^*.

Decentralized Deterministic Algorithms

First-order methods

- penalized gradient methods [Nedic 2009, Yuan 2016]
- EXTRA/PG-EXTRA [Shi 2015a, Shi 2015b]
- primal-dual methods [Shi 2014, Chang 2015]
- gradient tracking [Di Lorenzo 2015, Qu 2017, Nedic 2017, Sun 2019]

Second-order methods

- penalized methods [Mokhtari 2016, Bajovic 2017, Mansoori 2019]
- primal-dual methods [Mokhtari 2016, Zhang 2021]
- approximate Newton-type method [Li 2020]
- cubically-regularized Newton method [Daneshmand 2021]
- adaptive Newton method with a finite-time consensus inner loop [Zhang 2020]

Decentralized Stochastic Algorithms

First-order methods

- stochastic gradient [Chen 2012, Tang 2018, Pu 2019]
- variance-reduced gradient [Mokhtari 2016, Xin 2020, Li 2020, Pu 2021]

Second-order methods

- seldom investigated

Can we propose computationally affordable decentralized stochastic second-order methods?

Outline

A General Framework

Assumption 1: The mixing matrix W is nonnegative with $w_{ij} \geq 0$. The weight $w_{ij} = 0$ if and only if $j \notin \mathcal{N}_i$. W is symmetric and doubly stochastic, i.e., $W = W^T$ and $W1_n = 1_n$. The null space of $I_n - W$ is span (1_n).

- "Averaging" property of the mixing step

$$\|\mathrm{W}\mathrm{x}^k - \mathrm{W}_\infty \mathrm{x}^k\| \leq \sigma \|\mathrm{x}^k - \mathrm{W}_\infty \mathrm{x}^k\|,$$

  where $\sigma = \|W - \frac{1}{n}1_n 1_n^T\|_2 < 1$, $\mathrm{W} = W \otimes I_d$ and $\mathrm{W}_\infty = \frac{1_n 1_n^T}{n} \otimes I_d$.
- σ represents the connectedness of the network.

A General Framework

- Node i updates x_i^{k+1} according to the following decentralized stochastic quasi-Newton step
$$x_i^{k+1} = \sum_{j=1}^{n} w_{ij} x_j^k - \alpha d_i^k.$$
- In centralized setting, an ideal d_i^{k+1} is the global negative Newton direction
$$\left(\frac{1}{n}\sum_{i=1}^{n} \nabla^2 f_i(\overline{\mathrm{x}}^{k+1})\right)^{-1} \left(\frac{1}{n}\sum_{i=1}^{n} \nabla f_i(\overline{\mathrm{x}}^{k+1})\right),$$
where $\overline{\mathrm{x}}^{k+1} = \frac{1}{n}\sum_{i=1}^{n} x_i^{k+1}$.
- However, computing the global negative Newton direction is expensive: lack of information from the entire network / large m_i.
- **Our idea**: We update d_i^{k+1} with Hessian **inverse** approximation H_i^{k+1} and gradient approximation g_i^{k+1}, given by
$$d_i^{k+1} = H_i^{k+1} g_i^{k+1}.$$
We will construct g_i^{k+1} below and H_i^{k+1} in the next part.

Construct g_i^{k+1}

- Node i obtains a corrected stochastic gradient v_i^{k+1} with SVRG, as

$$v_i^{k+1} = \frac{1}{b_i} \sum_{l \in S_i^{k+1}} \left(\nabla f_{i,l}(x_i^{k+1}) - \nabla f_{i,l}(\tau_i^{k+1}) \right) + \nabla f_i(\tau_i^{k+1}),$$

  where $S_i^{k+1} \subseteq \{1, \ldots, m_i\}$ with batch size b_i, while $\tau_i^{k+1} = \tau_i^k$ or $\tau_i^{k+1} = x_i^{k+1}$ if $\mathrm{mod}(k+1, T) = 0$. We have $\mathbb{E}[v_i^k] = \nabla f_i(x_i^k)$.
- g_i^{k+1} is constructed with a dynamic average consensus (DAC) step

$$g_i^{k+1} = \sum_{j=1}^{n} w_{ij} g_j^k + v_i^{k+1} - v_i^k,$$

  with initialization $g_i^0 = v_i^0 = \nabla f_i(x_i^0)$. We have $\frac{1}{n}\sum_{i=1}^n g_i^k = \frac{1}{n}\sum_{i=1}^n v_i^k, \forall k$.
- With SVRG, $v_i^k \approx \nabla f_i(x_i^k)$. Therefore, with DAC, when x_i^k are almost consensual, $g_i^k \approx \frac{1}{n}\sum_{i=1}^n \nabla f_i(\bar{x}^k)$.

A General Framework

Proposed general framework

$$
\begin{cases}
x_i^{k+1} = \sum_{j=1}^{n} w_{ij} x_j^k - \alpha d_i^k, \\
v_i^{k+1} = \frac{1}{b_i} \sum_{l \in S_i^{k+1}} \left(\nabla f_{i,l}(x_i^{k+1}) - \nabla f_{i,l}(\tau_i^{k+1}) \right) \\
\qquad\quad + \nabla f_i(\tau_i^{k+1}), \\
g_i^{k+1} = \sum_{j=1}^{n} w_{ij} g_j^k + v_i^{k+1} - v_i^k, \\
d_i^{k+1} = H_i^{k+1} g_i^{k+1}.
\end{cases}
$$

Flow on node i

$$
\begin{array}{l}
x_i^{k+1} \\
\downarrow \\
\nabla f_{i,l}(x_i^{k+1}),\ \forall l \in S_i^{k+1} \\
\downarrow \\
v_i^{k+1} \approx \nabla f_i(x_i^{k+1}) \\
\downarrow \\
g_i^{k+1} \approx \frac{1}{n} \sum_{i=1}^{n} \nabla f_i(\overline{x}^{k+1}) \\
\downarrow \\
H_i^{k+1}\ ? \\
\downarrow \\
d_i^{k+1}
\end{array}
$$

- In the proposed general framework, H_i^k is constructed locally given g_i^k and x_i^k, **without** extra sampling or communication.

Linear Convergence of General Framework

Assumption 2: There exist two constants M_1 and M_2 with $0 < M_1 \leq M_2 < \infty$ such that

$$M_1 I_d \preceq H_i^k \preceq M_2 I_d, \ \forall i = 1, \ldots, n, \ \forall k \geq 0.$$

- We will check Assumption 2 in the next part.

Assumption 3: Each local sample cost $f_{i,l}$ is convex and has Lipschitz continuous gradients, i.e.,

$$f_{i,l}(y) \leq f_{i,l}(x) + \nabla f_{i,l}(x)^T (y - x) + \frac{L}{2} \|y - x\|^2.$$

Assumption 4: The global cost function F is strongly convex, i.e.,

$$F(y) \geq F(x) + \nabla F(x)^T (y - x) + \frac{\mu}{2} \|y - x\|^2.$$

Linear Convergence of General Framework

Theorem 1

Under Assumptions 1–4, if the parameters satisfy

$$\alpha \leq \frac{(1-\sigma^2)^2 \mu M_1}{200 L^2 M_2^2}, B \leq \frac{1}{160} \min\left\{1, \frac{\zeta(1-\sigma^2)^2}{\gamma^2}\right\}, T \geq \frac{2\log(280/(\zeta(1-\sigma^2)^2))}{\zeta\widetilde{\alpha}},$$

where $\zeta = \left(\frac{\mu}{L}\right)^2 \left(\frac{M_1}{M_2}\right)^2, \gamma = 1 - \frac{M_1}{M_2}$, *and* $\widetilde{\alpha} = \frac{M_2^2 L^2}{M_1 \mu}\alpha$. *Then, the proposed general framework converges linearly to the optimal solution, such that*

$$\|\mathrm{u}^{(t+1)T}\|_\infty^{\mathrm{q}} \leq 0.9 \|\mathrm{u}^{tT}\|_\infty^{\mathrm{q}}.$$

- Define the non-sampling rate $B = \max_{i\in\{1,\ldots,n\}} \left\{\frac{m_i - b_i}{(m_i-1)b_i}\right\} < 1$.
- Define $\kappa_F = L/\mu$ and $\kappa_H = M_2/M_1$.
- The total number of stochastic gradient evaluations is

$$\mathcal{O}\left(\left(\max_i\{m_i\} + \frac{\max_i\{b_i\} \cdot \kappa_F^2 \kappa_H^2 \log \frac{\kappa_F \kappa_H}{1-\sigma^2}}{(1-\sigma^2)^2}\right) \log \frac{1}{\epsilon}\right).$$

Outline

Two Quasi-Newton Methods

- Recall that H_i^k is constructed locally given g_i^k and x_i^k without extra sampling or communication, and is assumed to satisfy

$$M_1 I_d \preceq H_i^k \preceq M_2 I_d.$$

- How to construct Hessian approximation with gradient information?

  — **Quasi-Newton methods!**

Two Quasi-Newton Methods

- In centralized setting, two well-known quasi-Newton methods are

$$(DFP)\ H^{k+1} = H^k + \frac{s^k (s^k)^T}{(s^k)^T y^k} - \frac{H^k y^k (y^k)^T H^k}{(H^k y^k)^T y^k},$$

$$(BFGS)\ H^{k+1} = \left(I_d - \frac{s^k (y^k)^T}{(s^k)^T y^k}\right) H^k \left(I_d - \frac{y^k (s^k)^T}{(s^k)^T y^k}\right) + \frac{s^k (s^k)^T}{(s^k)^T y^k}.$$

  Here, s^k and y^k are defined as

$$s^k = x^{k+1} - x^k \text{ and } y^k = \nabla F(x^{k+1}) - \nabla F(x^k).$$

- Curvature condition $(s^k)^T y^k > 0$ holds due to strong convexity and thus $H^k \succ 0$.
- **However, g_i^k are noisy due to stochastic gradient noise and disagreement among the nodes, how to preserve $M_1 I_d \preceq H_i^k \preceq M_2 I_d$?**

  — Damping technique and limited-memory technique!

Damped Regularized Limited-memory DFP: Step I

- To guarantee $\lambda_{\min}(H_i^{k+1}) > 0$, we use **damping technique** and add a regularization

$$H_i^{k+1} = H_i^k + \frac{\hat{s}_i^k(\hat{s}_i^k)^T}{(\hat{s}_i^k)^T\hat{y}_i^k} - \frac{H_i^k\hat{y}_i^k(\hat{y}_i^k)^T H_i^k}{(\hat{y}_i^k)^T H_i^k \hat{y}_i^k} + \rho I_d,$$

where $s_i^k = x_i^{k+1} - x_i^k$, $y_i^k = g_i^{k+1} - g_i^k$, $\hat{s}_i^k = s_i^k - \rho y_i^k$, $\hat{y}_i^k = \theta_i^k y_i^k + (1-\theta_i^k)(H_i^0 + \epsilon I_d)^{-1}\hat{s}_i^k$,

$$\theta_i^k = \min\left\{\tilde{\theta}_i^k, \frac{\tilde{L}\|\hat{s}_i^k\|}{\|y_i^k\|}\right\} \text{ and } \tilde{\theta}_i^k = \begin{cases} \frac{0.75(\hat{s}_i^k)^T\left(H_i^0+\epsilon I_d\right)^{-1}\hat{s}_i^k}{(\hat{s}_i^k)^T\left(H_i^0+\epsilon I_d\right)^{-1}\hat{s}_i^k-(\hat{s}_i^k)^T y_i^k}, \\ \quad \text{if } (\hat{s}_i^k)^T y_i^k \leq 0.25(\hat{s}_i^k)^T\left(H_i^0+\epsilon I_d\right)^{-1}\hat{s}_i^k, \\ 1, \text{ otherwise.} \end{cases}$$

- With the corrected $\hat{y}_i^k$ by the damping technique, $(\hat{s}_i^k)^T\hat{y}_i^k > 0$ and thus $H_i^k \succ 0$.
- In addition, the regularization guarantees $\lambda_{\min}(H_i^{k+1}) > \rho$.

Damped Regularized Limited-memory DFP: Step II

- To guarantee $\lambda_{\max}(H_i^{k+1}) < \infty$, we further use **limited-memory technique**.
- Use a fixed moving window of M past variations

$$\{\hat{s}_i^{k+1-M}, \hat{s}_i^{k-M}, \ldots, \hat{s}_i^{k}\} \quad \text{and} \quad \{\hat{y}_i^{k+1-M}, \hat{y}_i^{k-M}, \ldots, \hat{y}_i^{k}\}.$$

- Recursively use M past variations and update H_i^{k+1} as

$$H_i^{k,(t+1)} = H_i^{k,(t)} + \frac{\hat{s}_i^p(\hat{s}_i^p)^T}{(\hat{s}_i^p)^T\hat{y}_i^p} - \frac{H_i^{k,(t)}\hat{y}_i^p(\hat{y}_i^p)^T H_i^{k,(t)}}{(\hat{y}_i^p)^T H_i^{k,(t)}\hat{y}_i^p} + \rho I_d,$$

  where $t = 0, \ldots, \tilde{M}-1$, $p = k+1-\tilde{M}+t$, and $\tilde{M} = \min\{k+1, M\}$.
- Restart after $\tilde{M}$ iterations with initialization $\beta I_d \preceq H_i^{k,(0)} \preceq \mathcal{B} I_d$.
- Computation cost per iteration is $O(Md^2)$, storage is $O(d^2 + Md)$.

Damped Limited-memory BFGS: Step I

- To guarantee $\lambda_{\min}(H_i^{k+1}) > 0$, we use **damping technique**.

$$H_i^{k+1} = \left(I_d - \frac{s_i^k (\hat{y}_i^k)^T}{(s_i^k)^T \hat{y}_i^k}\right) H_i^k \left(I_d - \frac{\hat{y}_i^k (s_i^k)^T}{(s_i^k)^T \hat{y}_i^k}\right) + \frac{s_i^k (s_i^k)^T}{(s_i^k)^T \hat{y}_i^k}.$$

where $\hat{y}_i^k = \theta y_i^k + (1-\theta)(H_i^0 + \epsilon I)^{-1} s_i^k$, and

$$\theta_i^k = \min\left\{\tilde{\theta}_i^k, \frac{\tilde{L}\|s_i^k\|}{\|y_i^k\|}\right\}, \ \tilde{\theta}_i^k = \begin{cases} \frac{0.75(s_i^k)^T\left(H_i^0+\epsilon I_d\right)^{-1}s_i^k}{(s_i^k)^T\left(H_i^0+\epsilon I_d\right)^{-1}s_i^k-(s_i^k)^T y_i^k}, \\ \quad \text{if } (s_i^k)^T y_i^k \leq 0.25 (s_i^k)^T \left(H_i^0 + \epsilon I_d\right)^{-1} s_i^k, \\ 1, \ \text{otherwise.} \end{cases}$$

- With the corrected $\hat{y}_i^k$ by the damping technique, $(s_i^k)^T \hat{y}_i^k > 0$ and thus $H_i^k \succ 0$.

Damped Limited-memory BFGS: Step II

- To guarantee $\lambda_{\max}(H_i^{k+1}) < \infty$, we further use **limited-memory technique**.

Algorithm 1: Two-loop recursion

Result: $H_i^{k+1} g_i^{k+1} \leftarrow r_i$
Set $q_i \leftarrow g_i^{k+1}$;
for $\underline{p = k, k-1, \ldots, k+1-\tilde{M}}$ **do**
  $\alpha_i^p \leftarrow \frac{(s_i^p)^T q_i}{(s_i^p)^T \hat{y}_i^p}$;
  $q_i \leftarrow q_i - \alpha_i^p \hat{y}_i^p$;
end
$r_i \leftarrow H_i^{k,(0)} q_i$;
for $\underline{p = k+1-\tilde{M}, k-\tilde{M}, \ldots, k}$ **do**
  $\beta_i \leftarrow \frac{(\hat{y}_i^p)^T r_i}{(s_i^p)^T \hat{y}_i^p}$;
  $r_i \leftarrow r_i + s_i^p(\alpha_i^p - \beta_i)$;
end

- Store M past variations $\{s_i^{k+1-M}, s_i^{k-M}, \ldots, s_i^k\}$ and $\{\hat{y}_i^{k+1-M}, \hat{y}_i^{k-M}, \ldots, \hat{y}_i^k\}$.
- Instead of generating H_i^{k+1} explicitly, update $H_i^{k+1} g_i^{k+1}$ by two-loop recursion.
- Restart after $\tilde{M}$ iterations with $\beta I_d \preceq H_i^{k,(0)} \preceq \mathcal{B} I_d$.
- Computation cost per iteration is $O(Md)$, storage is $O(Md)$.

Analysis of Proposed DFP and BFGS methods

Theorem 2 (DFP)

The proposed damped regularized limited-memory DFP satisfies

$$M_1 I_d \preceq H_i^k \preceq M_2 I_d, \forall i,$$

where $M_1 = \rho + (1+\omega)^{-2M}\left(\frac{1}{\beta} + \frac{1}{4(\mathcal{B}+\epsilon)}\right)^{-1}$, $M_2 = \mathcal{B} + M(4\mathcal{B} + 4\epsilon + \rho)$ *and* $\omega = 4(\mathcal{B}+\epsilon)\left(\tilde{L} + \frac{1}{\beta+\epsilon}\right)$.

Theorem 3 (BFGS)

The proposed damped limited-memory BFGS method satisfies

$$M_1 I_d \preceq H_i^k \preceq M_2 I_d,$$

where $M_1 = \left(\frac{1}{\beta} + \frac{M\omega^2}{4(\mathcal{B}+\epsilon)}\right)^{-1}$ *and* $M_2 = (1+\omega)^{2M}\left(\mathcal{B} + \frac{1}{\tilde{L}(\omega+2)}\right)$.

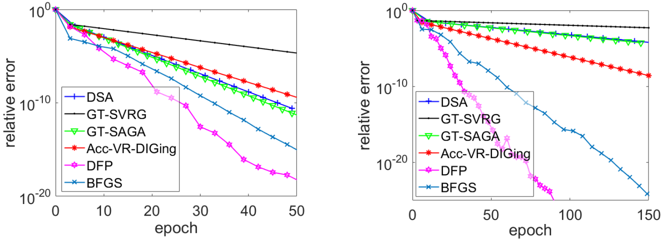
Outline

Effects of Condition Number: Synthetic Data

- Consider a least-squares problem

$$\min_{x \in \mathbb{R}^d} \frac{1}{2} \sum_{i=1}^{n} \|A_i x - b_i\|^2.$$

- Define $\kappa_{LS} = \frac{\lambda_{\max}(A^T A)}{\lambda_{\min}(A^T A)}$, where $A = [A_1; \cdots; A_n] \in \mathbb{R}^{nm \times d}$.
- The performance metric: relative error $= \frac{\|x^k - x^*\|^2}{n\|x^0 - x^*\|^2}$.



(a) $\kappa_{LS} = 10$ (b) $\kappa_{LS} = 2000$

Comparison with First-order Algorithms: Real Datasets

- Consider a logistic regression problem

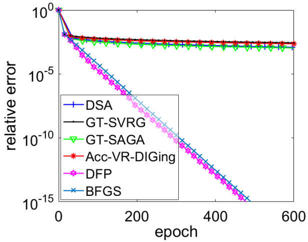
$$\min_{x \in \mathbb{R}^d} \frac{\iota}{2}\|x\|^2 + \frac{1}{n}\sum_{i=1}^{n} \frac{1}{m_i}\sum_{j=1}^{m_i} \ln\left(1 + \exp\left(-\left(\mathrm{o}_{ij}^T x\right)\mathrm{p}_{ij}\right)\right),$$

  where node i owns m_i training samples $(\mathrm{o}_{il}, \mathrm{p}_{il}) \in \mathbb{R}^d \times \{-1, +1\}$.
- We normalize each sample such that $\|\mathrm{o}_{il}\| = 1, \forall i, l$.

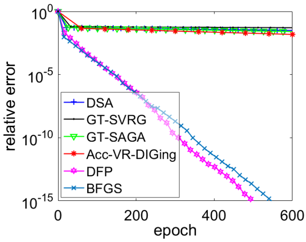
Tabela 1: Datasets used in numerical experiments.

Dataset	**# of Samples ($\sum_{i=1}^{n} m_i$)**	**# of Features (d)**
covtype	40000	54
cod-rna	52000	8
a6a	11220	123
a9a	32560	123
ijcnn1	91700	22

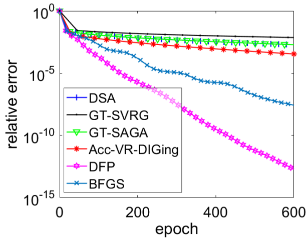
Comparison with First-order Algorithms: Real Datasets



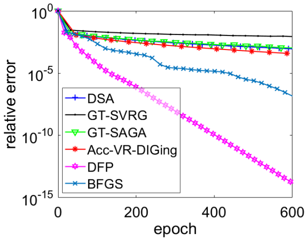
(c) covtype



(d) cod-rna



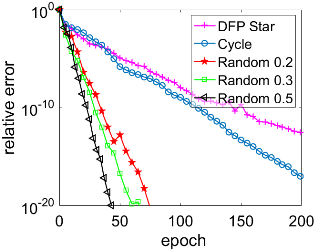
(e) a6a



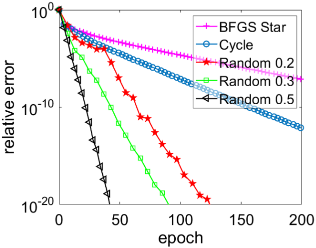
(f) a9a

Effects of Topology on ijcnn1

- σ of the five graphs are $0.967, 0.950, 0.863, 0.797$, and 0.569, respectively.



(g) Topology of DFP



(h) Topology of BFGS

Decentralized Stochastic Quasi-Newton Methods

Summary

- We propose a general framework of decentralized stochastic quasi-Newton methods, which converges linearly to the optimal solution.
- We specify two fully decentralized stochastic quasi-Newton methods to locally construct Hessian inverse approximations.

Future work

- Improve the theoretical results (with more communication or better initialization).

Thank you!

Two Lemmas

Lemma 4 (DFP)

With the corrected $\hat{y}_i^p$ *by the damping technique, we have*

$$0 < \theta_i^p \le 1 \text{ and } (\hat{s}_i^p)^T \hat{y}_i^p \ge 0.25(\hat{s}_i^p)^T (H_i^{k,(0)} + \epsilon I)^{-1} \hat{s}_i^p.$$

Moreover, H_i^{k+1} *keeps positive definite, such that* $\lambda_{\min}(H_i^{k+1}) > \rho$.

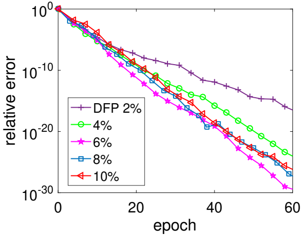
Lemma 5 (BFGS)

With the corrected $\hat{y}_i^p$ *by the damping technique, we have*

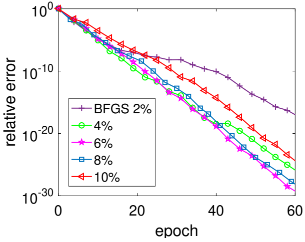
$$0 < \theta_i^p \le 1 \text{ and } (s_i^p)^T \hat{y}_i^p \ge 0.25(s_i^p)^T (H_i^{k,(0)} + \epsilon I_d)^{-1} s_i^p.$$

Moreover, H_i^{k+1} *keeps positive definite and* $\lambda_{\min}(H_i^{k+1}) > 0$.

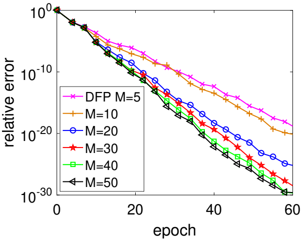
Effects of Batch Size and Memory Size on ijcnn1



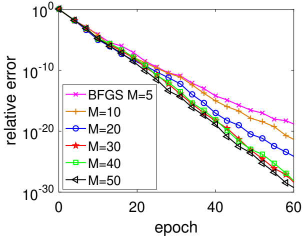
(i) Batch size of DFP



(j) Batch size of BFGS



(k) Memory size of DFP



(l) Memory size of BFGS