A Mini-Batch Quasi-Newton Proximal Method for Constrained Total-Variation Nonlinear Image Reconstruction

Tao Hong

FMRI Lab
University of Michigan, Ann Arbor
Email: tahong@umich.edu
Website: hongtao-argmin.github.io

Outline

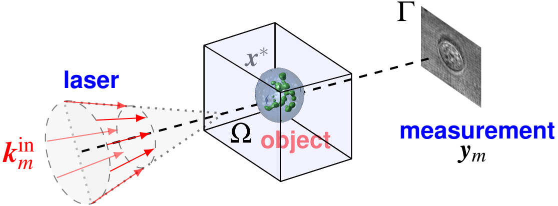
Inverse Problem – Optical Diffraction Tomography

Competing Algorithms

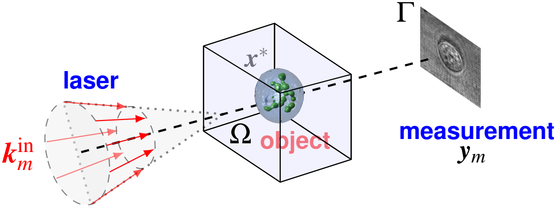
Mini-Batch Quasi-Newton Proximal Methods

Numerical Results

Optical Diffraction Tomography (ODT)



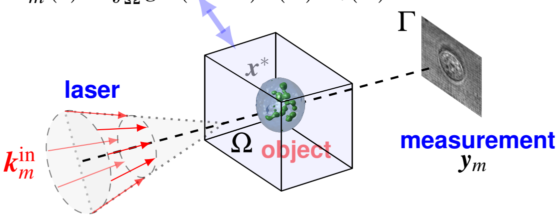
Optical Diffraction Tomography (ODT)



Nonlinear forward model ($\mathcal{H}_m$): Lippmann-Schwinger equation

Optical Diffraction Tomography (ODT)

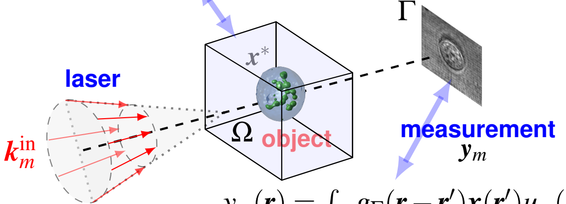
$$u_m(\boldsymbol{r}) = u_m^{\text{in}}(\boldsymbol{r}) + \int_\Omega g_\Omega(\boldsymbol{r}-\boldsymbol{r}')\boldsymbol{x}(\boldsymbol{r}')u_m(\boldsymbol{r}')d\boldsymbol{r}'$$



Nonlinear forward model ($\mathcal{H}_m$): Lippmann-Schwinger equation

Optical Diffraction Tomography (ODT)

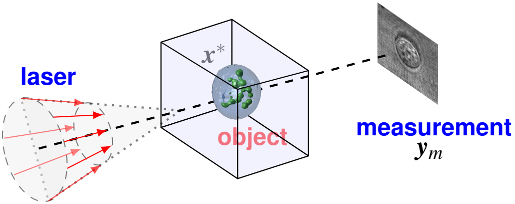
$$u_m(\boldsymbol{r}) = u_m^{\text{in}}(\boldsymbol{r}) + \int_\Omega g_\Omega(\boldsymbol{r} - \boldsymbol{r}')\boldsymbol{x}(\boldsymbol{r}')u_m(\boldsymbol{r}')d\boldsymbol{r}'$$



$$y_m(\boldsymbol{r}) = \int_\Omega g_\Gamma(\boldsymbol{r} - \boldsymbol{r}')\boldsymbol{x}(\boldsymbol{r}')u_m(\boldsymbol{r}')d\boldsymbol{r}' + u_m^{\text{in}}(\boldsymbol{r})|_\Gamma$$

Nonlinear forward model ($\mathcal{H}_m$): Lippmann-Schwinger equation

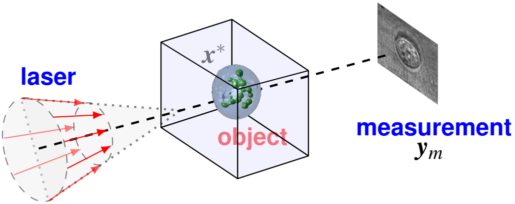
Optical Diffraction Tomography



Nonlinear forward model ($\mathcal{H}_m$): Lippmann-Schwinger equation

$$x^* = \arg\min_{x \in \mathcal{C}} \frac{1}{M} \sum_{m=1}^{M} \underbrace{\frac{1}{2} \| \mathcal{H}_m(x) - y_m \|_2^2}_{f_m} + R(x)$$

Optical Diffraction Tomography

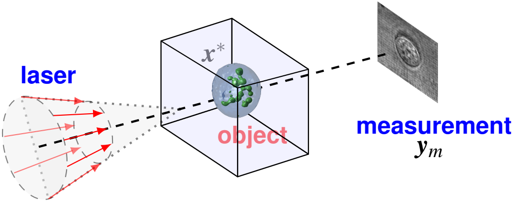


Nonlinear forward model ($\mathcal{H}_m$): Lippmann-Schwinger equation

$$x^* = \arg\min_{x \in \mathcal{C}} \frac{1}{M} \sum_{m=1}^{M} \underbrace{\frac{1}{2} \| \mathcal{H}_m(x) - y_m \|_2^2}_{f_m} + R(x)$$

∇f_m expensive, $M \gg 1$, $\mathcal{C}, f_m$ nonconvex, $R(\cdot)$ nonsmooth, ...

Optical Diffraction Tomography



Nonlinear forward model ($\mathcal{H}_m$): Lippmann-Schwinger equation

$$x^* = \arg\min_{x \in \mathcal{C}} \frac{1}{M} \sum_{m=1}^{M} \underbrace{\frac{1}{2} \| \mathcal{H}_m(x) - y_m \|_2^2}_{f_m} + R(x)$$

∇f_m expensive, $M \gg 1$, $\mathcal{C}, f_m$ nonconvex, $R(\cdot)$ nonsmooth, ...

We consider $R(\cdot) = \mathrm{TV}(\cdot)$

Solvers: $\min_{\boldsymbol{x} \in \mathcal{C}} \frac{1}{M} \sum_{m=1}^{M} f_m + R(\boldsymbol{x})$

APM, ADMM, primal-dual etc.

Solvers: $\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{M}\sum_{m=1}^{M} f_m + R(\boldsymbol{x})$

APM, ADMM, primal-dual etc.

Proximal gradient descent kth iter.

$$\boldsymbol{x}_{k+1} =$$
$$\underbrace{\arg\min_{\boldsymbol{u}\in\mathcal{C}} \left(\sum_m f_m(\boldsymbol{x}_k)\right) + \left\langle \sum_m \nabla f_m(\boldsymbol{x}_k), \boldsymbol{u}-\boldsymbol{x}_k \right\rangle + \frac{1}{2a_k}\|\boldsymbol{u}-\boldsymbol{x}_k\|_2^2 + M\cdot R(\boldsymbol{u})}$$

Solvers: $\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{M}\sum_{m=1}^{M} f_m + R(\boldsymbol{x})$

APM, ADMM, primal-dual etc.

Proximal gradient descent kth iter.

$$\boldsymbol{x}_{k+1} = \underbrace{\arg\min_{\boldsymbol{u}\in\mathcal{C}} \left(\sum_m f_m(\boldsymbol{x}_k)\right) + \left\langle \sum_m \nabla f_m(\boldsymbol{x}_k), \boldsymbol{u}-\boldsymbol{x}_k \right\rangle + \frac{1}{2a_k}\|\boldsymbol{u}-\boldsymbol{x}_k\|_2^2 + M\cdot R(\boldsymbol{u})}_{\operatorname{prox}_{a_k MR}(\boldsymbol{x}) = \arg\min_{\boldsymbol{u}\in\mathcal{C}} a_k MR(\boldsymbol{u}) + \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{x}\|_2^2}$$

Solvers: $\min_{\boldsymbol{x} \in \mathcal{C}} \frac{1}{M} \sum_{m=1}^{M} f_m + R(\boldsymbol{x})$

APM, ADMM, primal-dual etc.

Proximal gradient descent kth iter.

$$\boldsymbol{x}_{k+1} =$$
$$\underbrace{\arg\min_{\boldsymbol{u} \in \mathcal{C}} \left(\sum_m f_m(\boldsymbol{x}_k)\right) + \left\langle \sum_m \nabla f_m(\boldsymbol{x}_k), \boldsymbol{u} - \boldsymbol{x}_k \right\rangle + \frac{1}{2a_k} \|\boldsymbol{u} - \boldsymbol{x}_k\|_2^2 + M \cdot R(\boldsymbol{u})}_{\operatorname{prox}_{a_k MR}(\boldsymbol{x}) = \arg\min_{\boldsymbol{u} \in \mathcal{C}} a_k MR(\boldsymbol{u}) + \frac{1}{2}\|\boldsymbol{u} - \boldsymbol{x}\|_2^2}$$

Acceleration

$$\begin{cases} \boldsymbol{x}_{k+1} & = & \operatorname{prox}_{a_k MR}(\boldsymbol{v}_k - a_k \sum_m \nabla f_m(\boldsymbol{v}_k)) \\ \boldsymbol{v}_{k+1} & = & \boldsymbol{x}_{k+1} + c_k(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k) \end{cases}$$

Solvers: $\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{M}\sum_{m=1}^{M} f_m + R(\boldsymbol{x})$

APM, ADMM, primal-dual etc.

Proximal gradient descent kth iter.

$$\boldsymbol{x}_{k+1} = \underbrace{\arg\min_{\boldsymbol{u}\in\mathcal{C}} \left(\sum_m f_m(\boldsymbol{x}_k)\right) + \left\langle \sum_m \nabla f_m(\boldsymbol{x}_k), \boldsymbol{u}-\boldsymbol{x}_k \right\rangle + \frac{1}{2a_k}\|\boldsymbol{u}-\boldsymbol{x}_k\|_2^2 + M\cdot R(\boldsymbol{u})}_{\mathrm{prox}_{a_k MR}(\boldsymbol{x}) = \arg\min_{\boldsymbol{u}\in\mathcal{C}} a_k MR(\boldsymbol{u}) + \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{x}\|_2^2}$$

Acceleration

$$\begin{cases} \boldsymbol{x}_{k+1} &= \mathrm{prox}_{a_k MR}(\boldsymbol{v}_k - a_k \sum_m \nabla f_m(\boldsymbol{v}_k)) \\ \boldsymbol{v}_{k+1} &= \boldsymbol{x}_{k+1} + c_k(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k) \end{cases}$$

Stochastic? Choose some m as an estimation for $\sum_m \nabla f_m(\boldsymbol{x}_k)$

Solvers: $\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{M}\sum_{m=1}^{M} f_m + R(\boldsymbol{x})$

APM, ADMM, primal-dual etc.

Proximal gradient descent kth iter.

$$\boldsymbol{x}_{k+1} = \underbrace{\arg\min_{\boldsymbol{u}\in\mathcal{C}} \left(\sum_m f_m(\boldsymbol{x}_k)\right) + \left\langle \sum_m \nabla f_m(\boldsymbol{x}_k), \boldsymbol{u}-\boldsymbol{x}_k \right\rangle + \frac{1}{2a_k}\|\boldsymbol{u}-\boldsymbol{x}_k\|_2^2 + M\cdot R(\boldsymbol{u})}_{\operatorname{prox}_{a_k MR}(\boldsymbol{x}) = \arg\min_{\boldsymbol{u}\in\mathcal{C}} a_k MR(\boldsymbol{u}) + \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{x}\|_2^2}$$

Acceleration

$$\begin{cases} \boldsymbol{x}_{k+1} & = & \operatorname{prox}_{a_k MR}(\boldsymbol{v}_k - a_k \sum_m \nabla f_m(\boldsymbol{v}_k)) \\ \boldsymbol{v}_{k+1} & = & \boldsymbol{x}_{k+1} + c_k(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k) \end{cases}$$

Stochastic? Choose some m as an estimation for $\sum_m \nabla f_m(\boldsymbol{x}_k)$

$\rightarrow$ At each iteration computation is independent of M.

Solvers: $\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{M}\sum_{m=1}^{M} f_m + R(\boldsymbol{x})$

APM, ADMM, primal-dual etc.

Proximal gradient descent kth iter.

$$\boldsymbol{x}_{k+1} =$$

$$\underbrace{\arg\min_{\boldsymbol{u}\in\mathcal{C}} \Big(\sum_m f_m(\boldsymbol{x}_k)\Big) + \Big\langle \sum_m \nabla f_m(\boldsymbol{x}_k), \boldsymbol{u}-\boldsymbol{x}_k \Big\rangle + \frac{1}{2a_k}\|\boldsymbol{u}-\boldsymbol{x}_k\|_2^2 + M\cdot R(\boldsymbol{u})}_{\mathrm{prox}_{a_k MR}(\boldsymbol{x}) = \arg\min_{\boldsymbol{u}\in\mathcal{C}} a_k MR(\boldsymbol{u}) + \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{x}\|_2^2}$$

Acceleration

$$\begin{cases} \boldsymbol{x}_{k+1} &= \mathrm{prox}_{a_k MR}(\boldsymbol{v}_k - a_k \sum_m \nabla f_m(\boldsymbol{v}_k)) \\ \boldsymbol{v}_{k+1} &= \boldsymbol{x}_{k+1} + c_k(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k) \end{cases}$$

Stochastic? Choose some m as an estimation for $\sum_m \nabla f_m(\boldsymbol{x}_k)$

$\rightarrow$ At each iteration computation is independent of M.

Remark: for $R(\boldsymbol{D}\boldsymbol{x})$ (e.g., TV), we need iterative solvers for $\mathrm{prox}_R(\boldsymbol{x})$.

Inverse Problem – Optical Diffraction Tomography

Competing Algorithms

Mini-Batch Quasi-Newton Proximal Methods

Numerical Results

Our Suggestion

$$\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{M}\sum_m f_m(\boldsymbol{x}) + R(\boldsymbol{x})$$

$$\boldsymbol{x}_{k+1} =$$

$$\underbrace{\arg\min_{\boldsymbol{u}\in\mathcal{C}} \Big(\sum_m f_m(\boldsymbol{x}_k)\Big) + \Big\langle \sum_m \nabla f_m(\boldsymbol{x}_k), \boldsymbol{u}-\boldsymbol{x}_k \Big\rangle + \frac{1}{2a_k}(\boldsymbol{u}-\boldsymbol{x}_k)^{\mathcal{H}}\boldsymbol{I}(\boldsymbol{u}-\boldsymbol{x}_k) + M\cdot R(\boldsymbol{u})}_{\mathrm{prox}^{\boldsymbol{W}}_{a_k MR}(\boldsymbol{x}) = \arg\min_{\boldsymbol{u}\in\mathcal{C}} a_k MR(\boldsymbol{u}) + \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{x}\|_2^2,\ \boldsymbol{W}=\boldsymbol{I}}$$

Our Suggestion

$$\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{M}\sum_m f_m(\boldsymbol{x}) + R(\boldsymbol{x})$$

$$\boldsymbol{x}_{k+1} =$$

$$\underbrace{\arg\min_{\boldsymbol{u}\in\mathcal{C}} \Big(\sum_m f_m(\boldsymbol{x}_k)\Big) + \Big\langle \sum_m \nabla f_m(\boldsymbol{x}_k), \boldsymbol{u}-\boldsymbol{x}_k \Big\rangle + \frac{1}{2a_k}(\boldsymbol{u}-\boldsymbol{x}_k)^{\mathcal{H}} \boldsymbol{B}_k (\boldsymbol{u}-\boldsymbol{x}_k) + M\cdot R(\boldsymbol{u})}_{\mathrm{prox}_{a_k MR}^{\boldsymbol{W}}(\boldsymbol{x}) = \arg\min_{\boldsymbol{u}\in\mathcal{C}} a_k MR(\boldsymbol{u}) + \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{x}\|_{\boldsymbol{W}}^2,\ \boldsymbol{W}\succ 0 := \boldsymbol{B}_k}$$

Our Suggestion

$$\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{M}\sum_m f_m(\boldsymbol{x}) + R(\boldsymbol{x})$$

$$\boldsymbol{x}_{k+1} =$$

$$\underbrace{\arg\min_{\boldsymbol{u}\in\mathcal{C}} \Big(\sum_m f_m(\boldsymbol{x}_k)\Big) + \Big\langle \sum_m \nabla f_m(\boldsymbol{x}_k), \boldsymbol{u}-\boldsymbol{x}_k \Big\rangle + \frac{1}{2a_k}(\boldsymbol{u}-\boldsymbol{x}_k)^{\mathcal{H}} \boldsymbol{B}_k (\boldsymbol{u}-\boldsymbol{x}_k) + M\cdot R(\boldsymbol{u})}_{\mathrm{prox}^{\boldsymbol{W}}_{a_k MR}(\boldsymbol{x}) = \arg\min_{\boldsymbol{u}\in\mathcal{C}} a_k MR(\boldsymbol{u}) + \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{x}\|^2_{\boldsymbol{W}},\ \boldsymbol{W}\succ 0 := \boldsymbol{B}_k}$$

$$\boldsymbol{x}_{k+1} = \mathrm{prox}^{\boldsymbol{B}_k}_{a_k MR}\Big(\boldsymbol{x}_k - a_k \boldsymbol{B}_k^{-1} \sum_m \nabla f_m(\boldsymbol{x}_k)\Big)$$

Our Suggestion

$$\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{M}\sum_m f_m(\boldsymbol{x}) + R(\boldsymbol{x})$$

$$\boldsymbol{x}_{k+1} = \underbrace{\arg\min_{\boldsymbol{u}\in\mathcal{C}} \left(\sum_m f_m(\boldsymbol{x}_k)\right) + \left\langle \sum_m \nabla f_m(\boldsymbol{x}_k), \boldsymbol{u}-\boldsymbol{x}_k \right\rangle + \frac{1}{2a_k}(\boldsymbol{u}-\boldsymbol{x}_k)^{\mathcal{H}} \boldsymbol{B}_k (\boldsymbol{u}-\boldsymbol{x}_k) + M\cdot R(\boldsymbol{u})}_{\operatorname{prox}^{\boldsymbol{W}}_{a_k MR}(\boldsymbol{x}) = \arg\min_{\boldsymbol{u}\in\mathcal{C}} a_k MR(\boldsymbol{u}) + \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{x}\|^2_{\boldsymbol{W}},\ \boldsymbol{W}\succ 0 := \boldsymbol{B}_k}$$

$$\boldsymbol{x}_{k+1} = \operatorname{prox}^{\boldsymbol{B}_k}_{a_k MR}\left(\boldsymbol{x}_k - a_k \boldsymbol{B}_k^{-1} \sum_m \nabla f_m(\boldsymbol{x}_k)\right)$$

Second-order methods converge faster than APG $\rightarrow$ iterations

Our Suggestion

$$\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{M}\sum_m f_m(\boldsymbol{x}) + R(\boldsymbol{x})$$

$$\boldsymbol{x}_{k+1} =$$

$$\underbrace{\arg\min_{\boldsymbol{u}\in\mathcal{C}} \Big(\sum_m f_m(\boldsymbol{x}_k)\Big) + \Big\langle \sum_m \nabla f_m(\boldsymbol{x}_k), \boldsymbol{u}-\boldsymbol{x}_k \Big\rangle + \frac{1}{2a_k}(\boldsymbol{u}-\boldsymbol{x}_k)^{\mathcal{H}} \boldsymbol{B}_k (\boldsymbol{u}-\boldsymbol{x}_k) + M\cdot R(\boldsymbol{u})}_{\mathrm{prox}_{a_k MR}^{\boldsymbol{W}}(\boldsymbol{x}) = \arg\min_{\boldsymbol{u}\in\mathcal{C}} a_k MR(\boldsymbol{u}) + \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{x}\|_{\boldsymbol{W}}^2,\ \boldsymbol{W}\succ 0 := \boldsymbol{B}_k}$$

$$\boldsymbol{x}_{k+1} = \mathrm{prox}_{a_k MR}^{\boldsymbol{B}_k}\Big(\boldsymbol{x}_k - a_k \boldsymbol{B}_k^{-1} \sum_m \nabla f_m(\boldsymbol{x}_k)\Big)$$

Second-order methods converge faster than APG $\rightarrow$ iterations

Wall (CPU or GPU) time?

Our Suggestion

$$\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{M}\sum_m f_m(\boldsymbol{x}) + R(\boldsymbol{x})$$

$$\boldsymbol{x}_{k+1} =$$

$$\underbrace{\arg\min_{\boldsymbol{u}\in\mathcal{C}} \left(\sum_m f_m(\boldsymbol{x}_k)\right) + \left\langle \sum_m \nabla f_m(\boldsymbol{x}_k), \boldsymbol{u}-\boldsymbol{x}_k \right\rangle + \frac{1}{2a_k}(\boldsymbol{u}-\boldsymbol{x}_k)^{\mathcal{H}} \boldsymbol{B}_k (\boldsymbol{u}-\boldsymbol{x}_k) + M \cdot R(\boldsymbol{u})}_{\mathrm{prox}^{\boldsymbol{W}}_{a_k MR}(\boldsymbol{x}) = \arg\min_{\boldsymbol{u}\in\mathcal{C}} a_k MR(\boldsymbol{u}) + \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{x}\|^2_{\boldsymbol{W}},\ \boldsymbol{W}\succ 0 := \boldsymbol{B}_k}$$

$$\boldsymbol{x}_{k+1} = \mathrm{prox}^{\boldsymbol{B}_k}_{a_k MR}\left(\boldsymbol{x}_k - a_k \boldsymbol{B}_k^{-1} \sum_m \nabla f_m(\boldsymbol{x}_k)\right)$$

Second-order methods converge faster than APG $\rightarrow$ iterations

Wall (CPU or GPU) time?

Questions & Answers:

1. Q: Slow because of $\mathrm{prox}^{\boldsymbol{B}_k}_{R}$ A:

Our Suggestion

$$\min_{\boldsymbol{x} \in \mathcal{C}} \frac{1}{M} \sum_m f_m(\boldsymbol{x}) + R(\boldsymbol{x})$$

$$\boldsymbol{x}_{k+1} =$$

$$\underbrace{\arg\min_{\boldsymbol{u} \in \mathcal{C}} \left(\sum_m f_m(\boldsymbol{x}_k)\right) + \left\langle \sum_m \nabla f_m(\boldsymbol{x}_k), \boldsymbol{u} - \boldsymbol{x}_k \right\rangle + \frac{1}{2a_k} (\boldsymbol{u} - \boldsymbol{x}_k)^{\mathcal{H}} \boldsymbol{B}_k (\boldsymbol{u} - \boldsymbol{x}_k) + M \cdot R(\boldsymbol{u})}_{\mathrm{prox}_{a_k MR}^{\boldsymbol{W}}(\boldsymbol{x}) = \arg\min_{\boldsymbol{u} \in \mathcal{C}} a_k MR(\boldsymbol{u}) + \frac{1}{2} \|\boldsymbol{u} - \boldsymbol{x}\|_{\boldsymbol{W}}^2, \ \boldsymbol{W} \succ 0 := \boldsymbol{B}_k}$$

$$\boldsymbol{x}_{k+1} = \mathrm{prox}_{a_k MR}^{\boldsymbol{B}_k}\left(\boldsymbol{x}_k - a_k \boldsymbol{B}_k^{-1} \sum_m \nabla f_m(\boldsymbol{x}_k)\right)$$

Second-order methods converge faster than APG $\rightarrow$ iterations

Wall (CPU or GPU) time?

Questions & Answers:

1. Q: Slow because of $\mathrm{prox}_R^{\boldsymbol{B}_k}$ A: Make it fast

Our Suggestion

$$\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{M}\sum_m f_m(\boldsymbol{x}) + R(\boldsymbol{x})$$

$$\boldsymbol{x}_{k+1} = \underbrace{\arg\min_{\boldsymbol{u}\in\mathcal{C}} \left(\sum_m f_m(\boldsymbol{x}_k)\right) + \left\langle \sum_m \nabla f_m(\boldsymbol{x}_k), \boldsymbol{u}-\boldsymbol{x}_k \right\rangle + \frac{1}{2a_k}(\boldsymbol{u}-\boldsymbol{x}_k)^{\mathcal{H}} \boldsymbol{B}_k (\boldsymbol{u}-\boldsymbol{x}_k) + M\cdot R(\boldsymbol{u})}_{\mathrm{prox}_{a_k MR}^{\boldsymbol{W}}(\boldsymbol{x}) = \arg\min_{\boldsymbol{u}\in\mathcal{C}} a_k MR(\boldsymbol{u}) + \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{x}\|_{\boldsymbol{W}}^2,\ \boldsymbol{W}\succ 0 := \boldsymbol{B}_k}$$

$$\boldsymbol{x}_{k+1} = \mathrm{prox}_{a_k MR}^{\boldsymbol{B}_k}\left(\boldsymbol{x}_k - a_k \boldsymbol{B}_k^{-1} \sum_m \nabla f_m(\boldsymbol{x}_k)\right)$$

Second-order methods converge faster than APG $\rightarrow$ iterations

Wall (CPU or GPU) time?

Questions & Answers:

1. Q: Slow because of $\mathrm{prox}_R^{\boldsymbol{B}_k}$ A: Make it fast
2. Q: Too many m? A:

Our Suggestion

$$\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{M}\sum_m f_m(\boldsymbol{x}) + R(\boldsymbol{x})$$

$$\boldsymbol{x}_{k+1} = \underbrace{\arg\min_{\boldsymbol{u}\in\mathcal{C}} \Big(\sum_m f_m(\boldsymbol{x}_k)\Big) + \Big\langle \sum_m \nabla f_m(\boldsymbol{x}_k), \boldsymbol{u}-\boldsymbol{x}_k \Big\rangle + \frac{1}{2a_k}(\boldsymbol{u}-\boldsymbol{x}_k)^{\mathcal{H}}\boldsymbol{B}_k(\boldsymbol{u}-\boldsymbol{x}_k) + M\cdot R(\boldsymbol{u})}_{\mathrm{prox}^{\boldsymbol{W}}_{a_k MR}(\boldsymbol{x}) = \arg\min_{\boldsymbol{u}\in\mathcal{C}} a_k MR(\boldsymbol{u}) + \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{x}\|^2_{\boldsymbol{W}},\ \boldsymbol{W}\succ 0 := \boldsymbol{B}_k}$$

$$\boldsymbol{x}_{k+1} = \mathrm{prox}^{\boldsymbol{B}_k}_{a_k MR}\Big(\boldsymbol{x}_k - a_k \boldsymbol{B}_k^{-1}\sum_m \nabla f_m(\boldsymbol{x}_k)\Big)$$

Second-order methods converge faster than APG $\rightarrow$ iterations
Wall (CPU or GPU) time?

Questions & Answers:

1. Q: Slow because of $\mathrm{prox}_R^{\boldsymbol{B}_k}$ A: Make it fast
2. Q: Too many m? A: Stochastic behavior

Our Suggestion

$$\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{M}\sum_m f_m(\boldsymbol{x}) + R(\boldsymbol{x})$$

$$\boldsymbol{x}_{k+1} = \underbrace{\arg\min_{\boldsymbol{u}\in\mathcal{C}} \left(\sum_m f_m(\boldsymbol{x}_k)\right) + \left\langle \sum_m \nabla f_m(\boldsymbol{x}_k), \boldsymbol{u}-\boldsymbol{x}_k \right\rangle + \frac{1}{2a_k}(\boldsymbol{u}-\boldsymbol{x}_k)^{\mathcal{H}}\boldsymbol{B}_k(\boldsymbol{u}-\boldsymbol{x}_k) + M\cdot R(\boldsymbol{u})}_{\operatorname{prox}^{\boldsymbol{W}}_{a_k MR}(\boldsymbol{x}) = \arg\min_{\boldsymbol{u}\in\mathcal{C}} a_k MR(\boldsymbol{u}) + \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{x}\|^2_{\boldsymbol{W}},\ \boldsymbol{W}\succ 0 := \boldsymbol{B}_k}$$

$$\boldsymbol{x}_{k+1} = \operatorname{prox}^{\boldsymbol{B}_k}_{a_k MR}\left(\boldsymbol{x}_k - a_k\boldsymbol{B}_k^{-1}\sum_m \nabla f_m(\boldsymbol{x}_k)\right)$$

Second-order methods converge faster than APG $\rightarrow$ iterations

Wall (CPU or GPU) time?

Questions & Answers:

1. Q: Slow because of $\operatorname{prox}^{\boldsymbol{B}_k}_R$ A: Make it fast
2. Q: Too many m? A: Stochastic behavior
3. Q: But then how to get $\boldsymbol{B}_k$? A:

Our Suggestion

$$\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{M}\sum_m f_m(\boldsymbol{x}) + R(\boldsymbol{x})$$

$$\boldsymbol{x}_{k+1} = \underbrace{\arg\min_{\boldsymbol{u}\in\mathcal{C}} \left(\sum_m f_m(\boldsymbol{x}_k)\right) + \left\langle \sum_m \nabla f_m(\boldsymbol{x}_k), \boldsymbol{u}-\boldsymbol{x}_k \right\rangle + \frac{1}{2a_k}(\boldsymbol{u}-\boldsymbol{x}_k)^{\mathcal{H}}\boldsymbol{B}_k(\boldsymbol{u}-\boldsymbol{x}_k) + M\cdot R(\boldsymbol{u})}_{\mathrm{prox}^{\boldsymbol{W}}_{a_k MR}(\boldsymbol{x}) = \arg\min_{\boldsymbol{u}\in\mathcal{C}} a_k MR(\boldsymbol{u}) + \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{x}\|^2_{\boldsymbol{W}},\ \boldsymbol{W}\succ 0 := \boldsymbol{B}_k}$$

$$\boldsymbol{x}_{k+1} = \mathrm{prox}^{\boldsymbol{B}_k}_{a_k MR}\left(\boldsymbol{x}_k - a_k\boldsymbol{B}_k^{-1}\sum_m \nabla f_m(\boldsymbol{x}_k)\right)$$

Second-order methods converge faster than APG $\rightarrow$ iterations

Wall (CPU or GPU) time?

Questions & Answers:

1. Q: Slow because of $\mathrm{prox}^{\boldsymbol{B}_k}_R$ A: Make it fast
2. Q: Too many m? A: Stochastic behavior
3. Q: But then how to get $\boldsymbol{B}_k$? A: Recall quasi-Newton methods

Our Solution: Mini-batch & Second-order

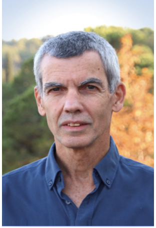
$$\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{M}\sum_m f_m(\boldsymbol{x}) + \lambda \mathrm{TV}(\boldsymbol{x}), \quad \lambda > 0$$

A Mini-Batch Quasi-Newton Proximal Method for Constrained Total-Variation Nonlinear Image Reconstruction

Tao Hong, Thanh-an Pham, Irad Yavneh, and Michael Unser *Fellow, IEEE*

arXiv: 2307.02043

***Abstract*—Over the years, computational imaging with accu-** and the tradeoff parameter $\lambda > 0$ balances these two terms. The



Challenging Issues (I) – Estimate $\boldsymbol{B}_k$ (Mini-Batch)

Steps: $\min_{\boldsymbol{x} \in \mathcal{C}} \frac{1}{M} \sum_m f_m(\boldsymbol{x}) + \lambda \mathrm{TV}(\boldsymbol{x})$

Challenging Issues (I) – Estimate $\boldsymbol{B}_k$ (Mini-Batch)

Steps: $\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{M}\sum_m f_m(\boldsymbol{x}) + \lambda \mathrm{TV}(\boldsymbol{x})$

1. Split $\{1,2,\cdots,M\}$ into K_s subsets $\{\mathcal{S}_t\}_{t=1}^{K_s}$ that $M=\sum_{t=1}^{K_s}|\mathcal{S}_t|$.

Challenging Issues (I) – Estimate $\boldsymbol{B}_k$ (Mini-Batch)

Steps: $\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{M}\sum_m f_m(\boldsymbol{x}) + \lambda \mathrm{TV}(\boldsymbol{x})$

1. Split $\{1,2,\cdots,M\}$ into K_s subsets $\{\mathcal{S}_t\}_{t=1}^{K_s}$ that $M = \sum_{t=1}^{K_s} |\mathcal{S}_t|$.

   Then $\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{K_s}\sum_t F_t(\boldsymbol{x}) + \lambda \mathrm{TV}(\boldsymbol{x})$ with $F_t(\boldsymbol{x}) = \frac{1}{|\mathcal{S}_t|}\sum_{m\in\mathcal{S}_t} f_m(\boldsymbol{x})$

Challenging Issues (I) – Estimate $\boldsymbol{B}_k$ (Mini-Batch)

Steps: $\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{M}\sum_m f_m(\boldsymbol{x}) + \lambda \mathrm{TV}(\boldsymbol{x})$

1. Split $\{1, 2, \cdots, M\}$ into K_s subsets $\{\mathcal{S}_t\}_{t=1}^{K_s}$ that $M = \sum_{t=1}^{K_s} |\mathcal{S}_t|$.

   Then $\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{K_s}\sum_t F_t(\boldsymbol{x}) + \lambda \mathrm{TV}(\boldsymbol{x})$ with $F_t(\boldsymbol{x}) = \frac{1}{|\mathcal{S}_t|}\sum_{m\in\mathcal{S}_t} f_m(\boldsymbol{x})$

2. Build second-order approximation for each $F_t(\boldsymbol{x})$ locally that we actually solve ($k \geq K_s$)

$$\boldsymbol{x}_k = \arg\min_{\boldsymbol{x}\in\mathcal{C}} \left(\sum_t \Big(\underbrace{\left\langle \nabla F_{\kappa_{K_s}(k,t)}(\boldsymbol{x}_{k-t}), \boldsymbol{x} - \boldsymbol{x}_{k-t} \right\rangle}_{F_t(\boldsymbol{x})} \right.$$

$$\left. \underbrace{+ \frac{1}{2a_k} (\boldsymbol{x} - \boldsymbol{x}_{k-t})^{\mathcal{T}} \boldsymbol{B}_{k-t}^{\kappa_{K_s}(k,t)} (\boldsymbol{x} - \boldsymbol{x}_{k-t}) \Big)}_{F_t(\boldsymbol{x})} + K_s \lambda \mathrm{TV}(\boldsymbol{x}) \right),$$

where $\kappa_{K_s}(k,t) = \mathrm{mod}(k - 1 - t, K_s) + 1$.

Challenging Issues (I) – Estimate $\boldsymbol{B}_k$ (Mini-Batch)

Steps: $\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{M}\sum_m f_m(\boldsymbol{x}) + \lambda \mathrm{TV}(\boldsymbol{x})$

1. Split $\{1, 2, \cdots, M\}$ into K_s subsets $\{\mathcal{S}_t\}_{t=1}^{K_s}$ that $M = \sum_{t=1}^{K_s} |\mathcal{S}_t|$.

   Then $\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{K_s}\sum_t F_t(\boldsymbol{x}) + \lambda \mathrm{TV}(\boldsymbol{x})$ with $F_t(\boldsymbol{x}) = \frac{1}{|\mathcal{S}_t|}\sum_{m\in\mathcal{S}_t} f_m(\boldsymbol{x})$

2. Build second-order approximation for each $F_t(\boldsymbol{x})$ locally that we actually solve ($k \geq K_s$)

$$\boldsymbol{x}_k = \arg\min_{\boldsymbol{x}\in\mathcal{C}} \left(\sum_t \Bigg(\underbrace{\left\langle \nabla F_{\kappa_{K_s}(k,t)}(\boldsymbol{x}_{k-t}), \boldsymbol{x} - \boldsymbol{x}_{k-t} \right\rangle}_{F_t(\boldsymbol{x})} \right.$$

$$\left. \underbrace{+ \frac{1}{2a_k}(\boldsymbol{x} - \boldsymbol{x}_{k-t})^{\mathcal{T}} \boldsymbol{B}_{k-t}^{\kappa_{K_s}(k,t)} (\boldsymbol{x} - \boldsymbol{x}_{k-t}) \Bigg)}_{F_t(\boldsymbol{x})} + K_s \lambda \mathrm{TV}(\boldsymbol{x}) \right),$$

where $\kappa_{K_s}(k,t) = \mathrm{mod}(k - 1 - t, K_s) + 1$.

Estimate $\boldsymbol{B}_k$ Continue: SR1

$$\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{K_s}\sum_t F_t(\boldsymbol{x}) + \lambda \mathrm{TV}(\boldsymbol{x})$$

$\boxed{\boldsymbol{x}_{k-6}, \nabla F_{\kappa_3(k,3)}(\boldsymbol{x}_{k-6})}$ $\boxed{\boldsymbol{x}_{k-5}, \nabla F_{\kappa_3(k,2)}(\boldsymbol{x}_{k-5})}$ $\boxed{\boldsymbol{x}_{k-4}, \nabla F_{\kappa_3(k,1)}(\boldsymbol{x}_{k-4})}$

$\boxed{\boldsymbol{x}_{k-3}, \nabla F_{\kappa_3(k,3)}(\boldsymbol{x}_{k-3})}$ $\boxed{\boldsymbol{x}_{k-2}, \nabla F_{\kappa_3(k,2)}(\boldsymbol{x}_{k-2})}$ $\boxed{\boldsymbol{x}_{k-1}, \nabla F_{\kappa_3(k,1)}(\boldsymbol{x}_{k-1})}$

Update $\boldsymbol{x}_k$ with $K_s = 3$

$$\kappa_{K_s}(k,t) = \mathrm{mod}(k-1-t, K_s) + 1$$

Estimate $\boldsymbol{B}_k$ Continue: SR1

$$\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{K_s}\sum_t F_t(\boldsymbol{x}) + \lambda \mathrm{TV}(\boldsymbol{x})$$

$$\begin{tikzcd}
\boxed{\boldsymbol{x}_{k-6}, \nabla F_{\kappa_3(k,3)}(\boldsymbol{x}_{k-6})} \arrow[d] & \boxed{\boldsymbol{x}_{k-5}, \nabla F_{\kappa_3(k,2)}(\boldsymbol{x}_{k-5})} \arrow[d] & \boxed{\boldsymbol{x}_{k-4}, \nabla F_{\kappa_3(k,1)}(\boldsymbol{x}_{k-4})} \arrow[d] \\
\boldsymbol{B}_{k-3}^{\kappa_3(k,3)} & \boldsymbol{B}_{k-2}^{\kappa_3(k,2)} & \boldsymbol{B}_{k-1}^{\kappa_3(k,1)} \\
\boxed{\boldsymbol{x}_{k-3}, \nabla F_{\kappa_3(k,3)}(\boldsymbol{x}_{k-3})} \arrow[u] & \boxed{\boldsymbol{x}_{k-2}, \nabla F_{\kappa_3(k,2)}(\boldsymbol{x}_{k-2})} \arrow[u] & \boxed{\boldsymbol{x}_{k-1}, \nabla F_{\kappa_3(k,1)}(\boldsymbol{x}_{k-1})} \arrow[u]
\end{tikzcd}$$

$\boxed{\text{Update } \boldsymbol{x}_k \text{ with } K_s = 3}$ $\quad \left\{\boldsymbol{B}_{k-t}^{\kappa_3(k,t)} \succ 0 := \boldsymbol{D}_t^k \pm \boldsymbol{u}_t^k (\boldsymbol{u}_t^k)^{\mathcal{T}}\right\}_t$: SR1

$$\kappa_{K_s}(k,t) = \mathrm{mod}(k-1-t, K_s) + 1$$

Estimate $\boldsymbol{B}_k$ Continue: SR1

$$\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{K_s}\sum_t F_t(\boldsymbol{x}) + \lambda \mathrm{TV}(\boldsymbol{x})$$

$$\begin{tikzcd}
\boxed{\boldsymbol{x}_{k-6}, \nabla F_{\kappa_3(k,3)}(\boldsymbol{x}_{k-6})} \arrow[d] & \boxed{\boldsymbol{x}_{k-5}, \nabla F_{\kappa_3(k,2)}(\boldsymbol{x}_{k-5})} \arrow[d] & \boxed{\boldsymbol{x}_{k-4}, \nabla F_{\kappa_3(k,1)}(\boldsymbol{x}_{k-4})} \arrow[d] \\
\boldsymbol{B}_{k-3}^{\kappa_3(k,3)} \arrow[d, dotted] & \boldsymbol{B}_{k-2}^{\kappa_3(k,2)} \arrow[dl, dotted] & \boldsymbol{B}_{k-1}^{\kappa_3(k,1)} \arrow[dll, dotted] \\
\boxed{\boldsymbol{x}_{k-3}, \nabla F_{\kappa_3(k,3)}(\boldsymbol{x}_{k-3})} \arrow[u] & \boxed{\boldsymbol{x}_{k-2}, \nabla F_{\kappa_3(k,2)}(\boldsymbol{x}_{k-2})} \arrow[u] & \boxed{\boldsymbol{x}_{k-1}, \nabla F_{\kappa_3(k,1)}(\boldsymbol{x}_{k-1})} \arrow[u] \\
\boxed{\text{Update } \boldsymbol{x}_k \text{ with } K_s = 3} & &
\end{tikzcd}$$

$$\left\{\boldsymbol{B}_{k-t}^{\kappa_3(k,t)} \succ 0 := \boldsymbol{D}_t^k \pm \boldsymbol{u}_t^k (\boldsymbol{u}_t^k)^{\mathcal{T}}\right\}_t : \text{SR1}$$

$$\kappa_{K_s}(k,t) = \mathrm{mod}(k-1-t, K_s) + 1$$

Estimate $\boldsymbol{B}_k$ Continue: SR1

$$\min_{\boldsymbol{x}\in\mathcal{C}} \frac{1}{K_s}\sum_t F_t(\boldsymbol{x}) + \lambda \mathrm{TV}(\boldsymbol{x})$$

$\boldsymbol{x}_{k-6}, \nabla F_{\kappa_3(k,3)}(\boldsymbol{x}_{k-6})$ | $\boldsymbol{x}_{k-5}, \nabla F_{\kappa_3(k,2)}(\boldsymbol{x}_{k-5})$ | $\boldsymbol{x}_{k-4}, \nabla F_{\kappa_3(k,1)}(\boldsymbol{x}_{k-4})$

$\boldsymbol{B}_{k-3}^{\kappa_3(k,3)}$ | $\boldsymbol{B}_{k-2}^{\kappa_3(k,2)}$ | $\boldsymbol{B}_{k-1}^{\kappa_3(k,1)}$

$\boldsymbol{x}_{k-3}, \nabla F_{\kappa_3(k,3)}(\boldsymbol{x}_{k-3})$ | $\boldsymbol{x}_{k-2}, \nabla F_{\kappa_3(k,2)}(\boldsymbol{x}_{k-2})$ | $\boldsymbol{x}_{k-1}, \nabla F_{\kappa_3(k,1)}(\boldsymbol{x}_{k-1})$

Update $\boldsymbol{x}_k$ with $K_s = 3$

$\left\{\boldsymbol{B}_{k-t}^{\kappa_3(k,t)} \succ 0 := \boldsymbol{D}_t^k \pm \boldsymbol{u}_t^k (\boldsymbol{u}_t^k)^{\mathcal{T}}\right\}_t$: SR1

$$\kappa_{K_s}(k,t) = \mathrm{mod}(k-1-t, K_s) + 1$$

Estimate $\boldsymbol{B}_k$ Continue: $\boldsymbol{B}_k = \sum_t = \because \pm \vdots \cdots$

Recall

$$
\begin{aligned}
\boldsymbol{x}_k = & \arg\min_{\boldsymbol{x}\in\mathcal{C}} \Bigg(\sum_t \Big(\left\langle \nabla F_{\kappa_{K_s}(k,t)}(\boldsymbol{x}_{k-t}), \boldsymbol{x} - \boldsymbol{x}_{k-t} \right\rangle \\
& \qquad + \frac{1}{2a_k} (\boldsymbol{x} - \boldsymbol{x}_{k-t})^{\mathcal{T}} \boldsymbol{B}_{k-t}^{\kappa_{K_s}(k,t)} (\boldsymbol{x} - \boldsymbol{x}_{k-t}) \Big) + K_s \lambda \mathrm{TV}(\boldsymbol{x}) \Bigg) \\
= & \underbrace{\arg\min_{\boldsymbol{x}\in\mathcal{C}} \left(\boxed{ \tfrac{1}{2} \|\boldsymbol{x} - \boldsymbol{v}_k\|_{\boldsymbol{B}_k}^2 + a_k K_s \lambda \mathrm{TV}(\boldsymbol{x}) } \right)}_{\mathrm{prox}_{a_k K_s \lambda \mathrm{TV}(\cdot)}^{\boldsymbol{B}_k}(\boldsymbol{v}_k)}
\end{aligned}
$$

Note $\boldsymbol{B}_k = \sum_t \boldsymbol{B}_{k-t}^{\kappa_{K_s}(k,t)}$ ($\boldsymbol{B}_k = \boldsymbol{D}_k \pm \boldsymbol{U}_k \boldsymbol{U}_k^{\mathcal{T}}$) and

$$
\boldsymbol{v}_k = (\boldsymbol{B}_k)^{-1} \sum_t \left(\boldsymbol{B}_{k-t}^{\kappa_{K_s}}(k,t) \boldsymbol{x}_{k-t} - a_k \nabla F_{\kappa_{K_s}(k,t)}(\boldsymbol{x}_{k-t}) \right).
$$

Challenging Issues (II) – Compute $\mathrm{prox}_{\mathrm{TV}(\cdot)}^{\boldsymbol{B}_k}(\boldsymbol{x})$ Efficiently

$$\mathrm{prox}_{\mathrm{TV}(\cdot)}^{\boldsymbol{B}_k}(\boldsymbol{x}) = \arg\min_{\boldsymbol{u}} \underbrace{\mathrm{TV}(\boldsymbol{u})}_{R(\boldsymbol{D}\boldsymbol{u})} + \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{x}\|_{\boldsymbol{B}_k}^2$$

Challenging Issues (II) – Compute $\mathrm{prox}_{\mathrm{TV}(\cdot)}^{\boldsymbol{B}_k}(\boldsymbol{x})$ Efficiently

$$\mathrm{prox}_{\mathrm{TV}(\cdot)}^{\boldsymbol{B}_k}(\boldsymbol{x}) = \arg\min_{\boldsymbol{u}} \underbrace{\mathrm{TV}(\boldsymbol{u})}_{R(\boldsymbol{D}\boldsymbol{u})} + \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{x}\|_{\boldsymbol{B}_k}^2$$

Dual formulation

Challenging Issues (II) – Compute $\mathrm{prox}_{\mathrm{TV}(\cdot)}^{\boldsymbol{B}_k}(\boldsymbol{x})$ Efficiently

$$\mathrm{prox}_{\mathrm{TV}(\cdot)}^{\boldsymbol{B}_k}(\boldsymbol{x}) = \arg\min_{\boldsymbol{u}} \underbrace{\mathrm{TV}(\boldsymbol{u})}_{R(\boldsymbol{D}\boldsymbol{u})} + \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{x}\|_{\boldsymbol{B}_k}^2$$

Dual formulation

Define $\mathcal{L} : \mathbb{R}^{(I-1)\times J} \times \mathbb{R}^{I\times(J-1)} \rightarrow \mathbb{R}^{I\times J}$ that

$$\mathcal{L}(\boldsymbol{P},\boldsymbol{Q})_{i,j} = \boldsymbol{P}_{i,j} + \boldsymbol{Q}_{i,j} - \boldsymbol{P}_{i-1,j} - \boldsymbol{Q}_{i,j-1}, \forall i,j,$$

The adjoint operator of $\mathcal{L} : \mathbb{R}^{I\times J} \rightarrow \mathbb{R}^{(I-1)\times J} \times \mathbb{R}^{I\times(J-1)}$ is

$$\mathcal{L}^{\mathcal{T}}(\boldsymbol{X}) = (\boldsymbol{P},\boldsymbol{Q}),$$

that $\boldsymbol{P}_{i,j} = \boldsymbol{X}_{i,j} - \boldsymbol{X}_{i+1,j}, \boldsymbol{Q}_{i,j} = \boldsymbol{X}_{i,j} - \boldsymbol{X}_{i,j+1}, \forall i,j$.

Challenging Issues (II) – Compute $\mathrm{prox}_{\mathrm{TV}(\cdot)}^{\boldsymbol{B}_k}(\boldsymbol{x})$ Efficiently

Primal:

$$\boldsymbol{x}_k = \arg\min_{\boldsymbol{x}\in\mathcal{C}} \left(\frac{1}{2}\|\boldsymbol{x}-\boldsymbol{v}_k\|_{\boldsymbol{B}_k}^2 + a_k K_s \lambda \mathrm{TV}(\boldsymbol{x}) \right),$$

Challenging Issues (II) – Compute $\mathrm{prox}_{\mathrm{TV}(\cdot)}^{\boldsymbol{B}_k}(\boldsymbol{x})$ Efficiently

Primal:

$$\boldsymbol{x}_k = \arg\min_{\boldsymbol{x}\in\mathcal{C}} \left(\frac{1}{2}\|\boldsymbol{x}-\boldsymbol{v}_k\|_{\boldsymbol{B}_k}^2 + a_k K_s \lambda \mathrm{TV}(\boldsymbol{x}) \right),$$

Dual:

$$(\boldsymbol{P}^*,\boldsymbol{Q}^*) = \arg\min_{(\boldsymbol{P},\boldsymbol{Q})\in\mathcal{P}} \Big(-\|\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q}) - \mathrm{prox}_{\delta_{\mathcal{C}}}^{\boldsymbol{B}_k}(\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q}))\|_{\boldsymbol{B}_k}^2 + \|\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q})\|_{\boldsymbol{B}_k}^2 \Big),$$

where $\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q}) = \boldsymbol{v}_k - a_k K_s \lambda (\boldsymbol{B}_k)^{-1} \mathcal{L}(\boldsymbol{P},\boldsymbol{Q})$.

Challenging Issues (II) – Compute $\mathrm{prox}_{\mathrm{TV}(\cdot)}^{\boldsymbol{B}_k}(\boldsymbol{x})$ Efficiently

Primal:

$$\boldsymbol{x}_k = \arg\min_{\boldsymbol{x}\in\mathcal{C}} \left(\frac{1}{2}\|\boldsymbol{x}-\boldsymbol{v}_k\|_{\boldsymbol{B}_k}^2 + a_k K_s \lambda \mathrm{TV}(\boldsymbol{x}) \right),$$

Dual:

$$(\boldsymbol{P}^*,\boldsymbol{Q}^*) = \arg\min_{(\boldsymbol{P},\boldsymbol{Q})\in\mathcal{P}} \Big(-\|\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q}) - \mathrm{prox}_{\delta_{\mathcal{C}}}^{\boldsymbol{B}_k}(\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q}))\|_{\boldsymbol{B}_k}^2 + \|\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q})\|_{\boldsymbol{B}_k}^2 \Big),$$

where $\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q}) = \boldsymbol{v}_k - a_k K_s \lambda (\boldsymbol{B}_k)^{-1} \mathcal{L}(\boldsymbol{P},\boldsymbol{Q})$.

Then $\boldsymbol{x}_k = \boxed{\mathrm{prox}_{\delta_{\mathcal{C}}}^{\boldsymbol{B}_k}(\boldsymbol{w}_k(\boldsymbol{P}^*,\boldsymbol{Q}^*))}$

Challenging Issues (II) – Compute $\mathrm{prox}_{\mathrm{TV}(\cdot)}^{\boldsymbol{B}_k}(\boldsymbol{x})$ Continue

$$(\boldsymbol{P}^*, \boldsymbol{Q}^*) = \arg \min_{(\boldsymbol{P},\boldsymbol{Q}) \in \mathcal{P}} \Big(-\| \boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q}) - \mathrm{prox}_{\delta_C}^{\boldsymbol{B}_k}(\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q})) \|_{\boldsymbol{B}_k}^2 + \| \boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q}) \|_{\boldsymbol{B}_k}^2 \Big),$$

Challenging Issues (II) – Compute $\mathrm{prox}_{\mathrm{TV}(\cdot)}^{\boldsymbol{B}_k}(\boldsymbol{x})$ Continue

$$(\boldsymbol{P}^*,\boldsymbol{Q}^*) = \arg\min_{(\boldsymbol{P},\boldsymbol{Q})\in\mathcal{P}} \Big(-\|\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q}) - \mathrm{prox}_{\delta_C}^{\boldsymbol{B}_k}(\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q}))\|_{\boldsymbol{B}_k}^2 + \|\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q})\|_{\boldsymbol{B}_k}^2 \Big),$$

Gradient: $-2a_k K_s \lambda \mathcal{L}^{\mathcal{T}}\left(\mathrm{prox}_{\delta_C}^{\boldsymbol{B}_k}(\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q}))\right)$

Lipschitz Constant: $16\omega_{\min} a_k^2 K_s^2 \lambda^2$ ($24\omega_{\min} a_k^2 K_s^2 \lambda^2$) for 2D (3D) &

$\omega_{\min}$ smallest eigenvalue $\boldsymbol{B}_k$.

Challenging Issues (II) – Compute $\mathrm{prox}_{\mathrm{TV}(\cdot)}^{\boldsymbol{B}_k}(\boldsymbol{x})$ Continue

$$(\boldsymbol{P}^*,\boldsymbol{Q}^*) = \arg\min_{(\boldsymbol{P},\boldsymbol{Q})\in\mathcal{P}} \Big(-\|\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q}) - \mathrm{prox}_{\delta_C}^{\boldsymbol{B}_k}(\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q}))\|_{\boldsymbol{B}_k}^2 + \|\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q})\|_{\boldsymbol{B}_k}^2 \Big),$$

Gradient: $-2a_k K_s \lambda \mathcal{L}^{\mathcal{T}}\left(\mathrm{prox}_{\delta_C}^{\boldsymbol{B}_k}(\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q}))\right)$

Lipschitz Constant: $16\omega_{\min}a_k^2K_s^2\lambda^2$ ($24\omega_{\min}a_k^2K_s^2\lambda^2$) for 2D (3D) &

$\omega_{\min}$ smallest eigenvalue $\boldsymbol{B}_k$.

$$\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q}) = \boldsymbol{v}_k - a_k K_s \lambda (\boldsymbol{B}_k)^{-1}\mathcal{L}(\boldsymbol{P},\boldsymbol{Q})$$

$$\boldsymbol{v}_k = (\boldsymbol{B}_k)^{-1}\sum_t \left(\boldsymbol{B}_{k-t}^{\kappa_{K_s}}(k,t)\boldsymbol{x}_{k-t} - a_k \nabla F_{\kappa_{K_s}(k,t)}(\boldsymbol{x}_{k-t})\right)$$

Challenging Issues (II) – Compute $\mathrm{prox}_{\mathrm{TV}(\cdot)}^{\boldsymbol{B}_k}(\boldsymbol{x})$ Continue

$$(\boldsymbol{P}^*,\boldsymbol{Q}^*) = \arg\min_{(\boldsymbol{P},\boldsymbol{Q})\in\mathcal{P}} \Big(-\|\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q}) - \mathrm{prox}_{\delta_C}^{\boldsymbol{B}_k}(\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q}))\|_{\boldsymbol{B}_k}^2 + \|\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q})\|_{\boldsymbol{B}_k}^2 \Big),$$

Gradient: $-2a_k K_s \lambda \mathcal{L}^{\mathcal{T}}\left(\mathrm{prox}_{\delta_C}^{\boldsymbol{B}_k}(\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q}))\right)$

Lipschitz Constant: $16\omega_{\min} a_k^2 K_s^2 \lambda^2$ ($24\omega_{\min} a_k^2 K_s^2 \lambda^2$) for 2D (3D) &

$\omega_{\min}$ smallest eigenvalue $\boldsymbol{B}_k$.

$$\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q}) = \boldsymbol{v}_k - a_k K_s \lambda (\boldsymbol{B}_k)^{-1} \mathcal{L}(\boldsymbol{P},\boldsymbol{Q})$$

$$\boldsymbol{v}_k = (\boldsymbol{B}_k)^{-1} \sum_t \left(\boldsymbol{B}_{k-t}^{\kappa_{K_s}}(k,t)\boldsymbol{x}_{k-t} - a_k \nabla F_{\kappa_{K_s}(k,t)}(\boldsymbol{x}_{k-t}) \right)$$

Recall $\left\{ \boldsymbol{B}_{k-t}^{\kappa_{K_s}(k,t)} \succ 0 := \boldsymbol{D}_t^k \pm \boldsymbol{u}_t^k (\boldsymbol{u}_t^k)^{\mathcal{T}} \right\}_t \to \boldsymbol{B}_k = \sum_t \boldsymbol{B}_{k-t}^{\kappa_{K_s}(k,t)}$

Woodbury matrix identity

Challenging Issues (II) – Compute $\mathrm{prox}^{\boldsymbol{B}_k}_{\mathrm{TV}(\cdot)}(\boldsymbol{x})$ Continue

Gradient: $-2a_k K_s \lambda \mathcal{L}^{\mathcal{T}}\left(\mathrm{prox}^{\boldsymbol{B}_k}_{\delta_C}\left(\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q})\right)\right)$

Challenging Issues (II) – Compute $\mathrm{prox}_{\mathrm{TV}(\cdot)}^{\boldsymbol{B}_k}(\boldsymbol{x})$ Continue

Gradient: $-2a_k K_s \lambda \mathcal{L}^{\mathcal{T}} \left(\mathrm{prox}_{\delta_{\mathcal{C}}}^{\boldsymbol{B}_k} (\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q})) \right)$

Note $\boldsymbol{B}_k = \sum_t \boldsymbol{B}_{k-t}^{\kappa_{K_s}(k,t)} : \boldsymbol{D} \pm \boldsymbol{U}\boldsymbol{U}^{\mathcal{T}}$

Challenging Issues (II) – Compute $\mathrm{prox}_{\mathrm{TV}(\cdot)}^{\boldsymbol{B}_k}(\boldsymbol{x})$ Continue

Gradient: $-2a_k K_s \lambda \mathcal{L}^{\mathcal{T}}\left(\mathrm{prox}_{\delta_C}^{\boldsymbol{B}_k}(\boldsymbol{w}_k(\boldsymbol{P},\boldsymbol{Q}))\right)$

Note $\boldsymbol{B}_k = \sum_t \boldsymbol{B}_{k-t}^{\kappa_{K_s}(k,t)} : \boldsymbol{D} \pm \boldsymbol{U}\boldsymbol{U}^{\mathcal{T}}$

Theorem (Becker19 SIAMOPT)

Let $\boldsymbol{W} \in \mathbb{R}^{N\times N} := \boldsymbol{D} \pm \boldsymbol{U}\boldsymbol{U}^{\mathcal{T}}$ *with* $\boldsymbol{U} \in \mathbb{R}^{N\times r}$ *and* $N \gg r$*. Then,*

$$\mathrm{prox}_{\lambda R}^{\boldsymbol{W}}(\boldsymbol{x}) = \mathrm{prox}_{\lambda R}^{\boldsymbol{D}}(\boldsymbol{x} \mp \boldsymbol{D}^{-1}\boldsymbol{U}\boldsymbol{\beta}^*),$$

where $\boldsymbol{\beta}^* \in \mathbb{R}^r$ *is the unique zero vector of the following nonlinear equations*

$$\mathbb{J}(\beta) : \boldsymbol{U}^{\mathcal{T}}\left(\boldsymbol{x} - \mathrm{prox}_{\lambda R}^{\boldsymbol{D}}(\boldsymbol{x} \mp \boldsymbol{D}^{-1}\boldsymbol{U}\boldsymbol{\beta})\right) + \boldsymbol{\beta}.$$

Inverse Problem – Optical Diffraction Tomography

Competing Algorithms

Mini-Batch Quasi-Newton Proximal Methods

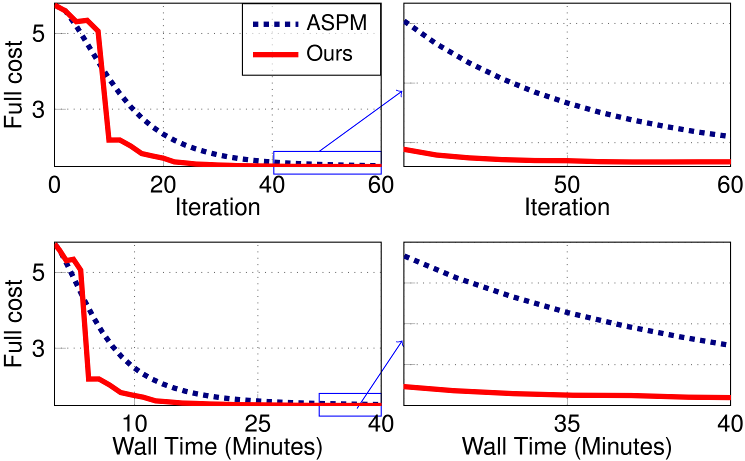
Numerical Results

Numerical Results - Real Data

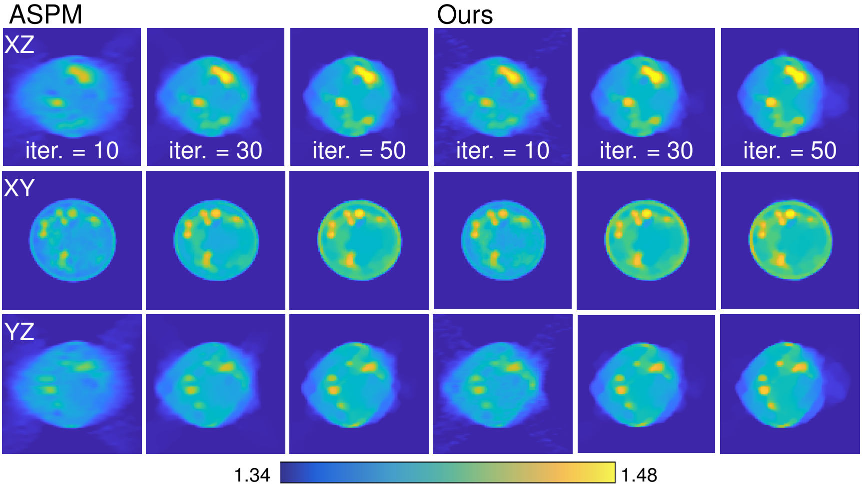
- yeast cell & water (RI=1.338)
- $M = 60$ incident plane waves (532nm) & cone illumination half angle 35°
- Image size 96^3 & $\boldsymbol{y}_m \in \mathbb{C}^{150\times 150}$ & $K_s = 5$
- Compare with acc. stochastic prox. method (ASPM) & GPU

Numerical Results - Real Data

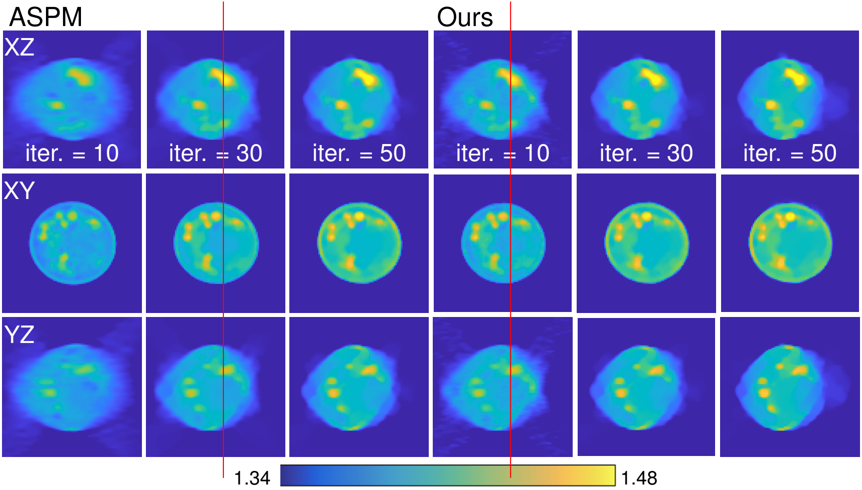
- yeast cell & water (RI=1.338)
- $M = 60$ incident plane waves (532nm) & cone illumination half angle 35°
- Image size 96^3 & $\boldsymbol{y}_m \in \mathbb{C}^{150\times150}$ & $K_s = 5$
- Compare with acc. stochastic prox. method (ASPM) & GPU



Reconstructed Images



Reconstructed Images



Scan Me

Thanks & Questions?