Supplementary Material for "DEs-Inspired Accelerated Unfolded Linearized ADMM Networks for Inverse Problems"

There are some detailed explanations for the main paper. Firstly, we give the proofs of Lemma 2 and Theorem 1 in Section A. Secondly, the convergence analysis of the explicit Trapezoid LADMM scheme is given in Section B. Finally, we provide more experimental details and results in Section C.

A. DEs Form and Error Analysis

I. Proof of Lemma 2

Proof. Our implicit Trapezoid LADMM scheme (15) can be rewritten as the following minimizing problems:

$$\begin{cases} \mathbf{x}_{k+1} = \underset{\mathbf{x}}{\operatorname{argmin}} \Big\{ f(\mathbf{x}) + \dfrac{\theta_k}{2h} \big\| \mathbf{x} - \mathbf{x}_k + \dfrac{h\beta_k}{2\theta_k} \big(F_k(\mathbf{x}_k) + F_k(\mathbf{x}_{k+1}) \big) \big\|^2 \Big\}, \\ \mathbf{y}_{k+1} = \underset{\mathbf{y}}{\operatorname{argmin}} \Big\{ g(\mathbf{y}) + \dfrac{\eta_k}{2h} \big\| \mathbf{y} - \mathbf{y}_k + \dfrac{h}{2\eta_k} \big(G_k(\mathbf{y}_k) + G_k(\mathbf{y}_{k+1}) \big) \big\|^2 \Big\}, \\ \boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + h\beta_k(\mathbf{A}\mathbf{x}_{k+1} + \mathbf{y}_{k+1} - \mathbf{b}). \end{cases} \tag{A.1}$$

In the similar way, we can have the first-order optimality conditions of (A.1):

$$\begin{cases} 0 \in \partial f(\mathbf{x}_{k+1}) + \dfrac{\theta_k}{h} \Big(\mathbf{x}_{k+1} - \mathbf{x}_k + \dfrac{h\beta_k}{2\theta_k} \big(F_k(\mathbf{x}_k) + F_k(\mathbf{x}_{k+1}) \big) \Big), \\ 0 \in \partial g(\mathbf{y}_{k+1}) + \dfrac{\eta_k}{h} \Big(\mathbf{y}_{k+1} - \mathbf{y}_k + \dfrac{h}{2\eta_k} \big(G_k(\mathbf{y}_k) + G_k(\mathbf{y}_{k+1}) \big) \Big), \\ 0 = \boldsymbol{\lambda}_{k+1} - \boldsymbol{\lambda}_k - h\beta_k(\mathbf{A}\mathbf{x}_{k+1} + \mathbf{y}_{k+1} - \mathbf{b}). \end{cases} \tag{A.2}$$

From the first inclusion, we obtain

$$0 \in \partial f(\mathbf{x}_{k+1}) + \theta_k \frac{\mathbf{x}_{k+1} - \mathbf{x}_k}{h} + \frac{1}{2} \left(\mathbf{W}_k^\top \left(\boldsymbol{\lambda}_k + \beta_k(\mathbf{A}\mathbf{x}_k + \mathbf{y}_k - \mathbf{b}) \right) + \mathbf{W}_k^\top \left(\boldsymbol{\lambda}_k + \beta_k(\mathbf{A}\mathbf{x}_{k+1} + \mathbf{y}_k - \mathbf{b}) \right) \right). \tag{A.3}$$

We also see that $\lim_{h\to 0} \frac{\mathbf{x}_{k+1} - \mathbf{x}_k}{h} = \dot{\mathbf{X}}(t)$, $\mathbf{x}_{k+1} = \mathbf{X}(t+h) \xrightarrow{h\to 0} \mathbf{X}(t)$, $\mathbf{y}_{k+1} = \mathbf{Y}(t+h) \xrightarrow{h\to 0} \mathbf{Y}(t)$, and $\boldsymbol{\lambda}_{k+1} = \boldsymbol{\Lambda}(t+h) \xrightarrow{h\to 0} \boldsymbol{\Lambda}(t)$. Then,

$$\text{(A.3)} \xrightarrow{h\to 0} 0 \in \partial f(\mathbf{X}(t)) + \theta \dot{\mathbf{X}}(t) - F(\mathbf{X}(t)). \tag{A.4}$$

And $0 \in \partial g(\mathbf{Y}(t)) + \eta \dot{\mathbf{Y}}(t) - G(\mathbf{Y}(t))$ can be obtained in the same way. Finally, about $\boldsymbol{\lambda}$, we can also obtain $\dot{\boldsymbol{\Lambda}}(t) - \beta(\mathbf{A}\mathbf{X}(t) + \mathbf{Y}(t) - \mathbf{b}) = 0$. We consider the Moreau-Yosida approximations $f_{\mu_1}(\mathbf{x})$ and $g_{\mu_2}(\mathbf{y})$ of objective $f(\mathbf{x})$ and $g(\mathbf{y})$ with $\mu_1, \mu_2 > 0$. Then the implicit Trapezoid LADMM scheme (15) also corresponds to solving the approximating DEs (8). In this case, the accuracy or error of the two schemes can be compared. Conversely, if our Euler LADMM scheme and Trapezoid LADMM scheme do not correspond to solving the same DEs, then the Trapezoid LADMM scheme may not generate an unfolded network with faster convergence despite its higher precision. □

II. Proof of Theorem 1

Proof. In this subsection, we give the local and global error bound analysis of our Euler LADMM scheme (7) and Trapezoid LADMM scheme (15), and our analysis refers to [1]. We first review the following notations.

1. $\mathcal{X}_{k+1} = (\mathbf{x}_{k+1}^\top, \mathbf{y}_{k+1}^\top, \boldsymbol{\lambda}_{k+1}^\top)^\top$ is an iterative solution;
2. The optimal trajectory function is $\mathcal{X}(t) = (\mathbf{X}(t)^\top, \mathbf{Y}(t)^\top, \boldsymbol{\Lambda}(t)^\top)^\top$ and initial value $\mathcal{X}(0) = (\mathbf{x}_0^\top, \mathbf{y}_0^\top, \boldsymbol{\lambda}_0^\top)^\top$;
3. Let $\mathbf{P}(t, \Theta, \mathcal{X}) = (\frac{1}{\theta}(F(\mathbf{X}) - \nabla f(\mathbf{X}))^\top, \frac{1}{\eta}(G(\mathbf{Y}) - \nabla g(\mathbf{Y}))^\top, \beta(\mathbf{A}\mathbf{X} + \mathbf{Y} - \mathbf{b})^\top)^\top$, omitting t, and $\Theta = (\mathbf{W}, \theta, \eta, \beta)$.

Under the assumptions mentioned in Theorem 1 and these definitions, we can obatin a differential equation w.r.t. $\mathcal{X}$, i.e., $\dot{\mathcal{X}} = \mathbf{P}(t, \Theta, \mathcal{X})$, with the initial condition $\mathcal{X}(0) = \mathcal{X}_0$.

Besides, the optimal value at t_k is defined as $\mathcal{X}(t_k)$, $t_k \in [0, T]$. The global error bound from optimal trajectory is defined as $\boldsymbol{\varepsilon}_{k+1} = \mathcal{X}(t_{k+1}) - \mathcal{X}_{k+1}$; the local error bound from optimal trajectory is defined as $\mathbf{e}_{k+1} = \mathcal{X}(t_{k+1}) - \mathcal{X}_{k+1}^*$, where $\mathcal{X}_{k+1}^* = \mathcal{K}\big(\mathcal{X}(t_k) + h\mathbf{P}(t_k, \mathcal{X}(t_k))\big)$ for the Euler LADMM scheme (7) and $\mathcal{X}_{k+1}^* = \mathcal{K}\big(\mathcal{X}(t_k) + \frac{h}{2}\left[\mathbf{P}(t_k, \Theta_k, \mathcal{X}(t_k)) + \mathbf{P}(t_{k+1}, \Theta_k, \mathcal{X}(t_{k+1}))\right]\big)$ for the Trapezoid LADMM scheme (15), non-linear transformation $\mathcal{K}(\cdot) = (\mathcal{F}_f(\cdot); \mathcal{G}_g(\cdot); \mathbf{I}(\cdot))$, and $\|\cdot\|$ represents the vector ℓ_2-norm.

The Error Bound of Our Euler LADMM scheme: We start from Euler LADMM scheme (7) and consider local error as follows:

$$\begin{aligned}\|\mathbf{e}_{k+1}\| &= \|\mathcal{X}(t_{k+1}) - \mathcal{X}^*_{k+1}\| \\ &= \|\mathcal{K}\Big(\mathcal{X}(t_k) + \int_{t_k}^{t_{k+1}} \mathbf{P}(t,\Theta,\mathcal{X}(t))\mathrm{d}t\Big) - \mathcal{K}\Big(\mathcal{X}(t_k) + h\mathbf{P}(t_k,\Theta_k,\mathcal{X}(t_k))\Big)\| \\ &\overset{\text{(b)}}{\leq} \|\mathcal{X}(t_k) + \int_{t_k}^{t_{k+1}} \mathbf{P}(t,\Theta,\mathcal{X}(t))\mathrm{d}t - [\mathcal{X}(t_k) + h\mathbf{P}(t_k,\Theta_k,\mathcal{X}(t_k))]\| \\ &= \|\int_{t_k}^{t_{k+1}} \mathbf{P}(t,\Theta,\mathcal{X}(t))\mathrm{d}t - h\mathbf{P}(t_k,\Theta_k,\mathcal{X}(t_k))\|\end{aligned} \tag{A.5}$$

where $\overset{\text{(b)}}{\leq}$ holds because $\mathcal{F}_f(\cdot)$ and $\mathcal{G}_g(\cdot)$ are non-expansive mappings, and the rest of the equation holds due to [1, Theorem 12.2]. Moreover, let us estimate the local error bound:

$$\|\mathbf{e}_{k+1}\| = \|\int_{t_k}^{t_{k+1}} \Big[\dot{\mathcal{X}}(t) - \dot{\mathcal{X}}(t_k)\Big]\,\mathrm{d}t\| = \|\int_{t_k}^{t_{k+1}} \ddot{\mathcal{X}}(t_k + \xi(t-t_k))(t-t_k)\mathrm{d}t\| = \|\frac{1}{2}h^2\ddot{\mathcal{X}}(t_k + \xi(\bar{t}-t_k))\| \tag{A.6}$$

where $0 < \xi < 1, \bar{t} \in (t_k, t_{k+1})$, and the second equation holds due to the Mean Value Theorem, and the rest of the equation holds due to [1, Theorem 12.2]. Set $Q_1 = \max_{t_0\leq t\leq T}\|\ddot{\mathcal{X}}(t)\|$, then $\|\mathbf{e}_{k+1}\| \leq \frac{1}{2}Q_1h^2$, that is, the local error bound is $\mathcal{O}(h^2)$. Next, the global error bound of the Euler LADMM scheme (7) is:

$$\begin{aligned}\|\boldsymbol{\varepsilon}_{k+1}\| &= \|\mathcal{X}(t_{k+1}) - \mathcal{X}_{k+1}\| \\ &\leq \|\boldsymbol{\varepsilon}_k + \int_{t_k}^{t_{k+1}} \mathbf{P}(t,\Theta,\mathcal{X}(t))\mathrm{d}t - \int_{t_k}^{t_{k+1}} \mathbf{P}(t_k,\Theta_k,\mathcal{X}(t_k))\mathrm{d}t + \int_{t_k}^{t_{k+1}} \mathbf{P}(t_k,\Theta_k,\mathcal{X}(t_k))\mathrm{d}t - \int_{t_k}^{t_{k+1}} \mathbf{P}(t_k,\Theta_k,\mathcal{X}_k)\mathrm{d}t\| \\ &= \|\boldsymbol{\varepsilon}_k + \mathbf{e}_{k+1} + \int_{t_k}^{t_{k+1}} [\mathbf{P}(t_k,\Theta_k,\mathcal{X}(t_k)) - \mathbf{P}(t_k,\Theta_k,\mathcal{X}_k)]\,\mathrm{d}t\| \\ &\leq \|\boldsymbol{\varepsilon}_k\| + \frac{1}{2}Q_1h^2 + \int_{t_k}^{t_{k+1}} \|\mathbf{P}(t_k,\Theta_k,\mathcal{X}(t_k)) - \mathbf{P}(t_k,\Theta_k,\mathcal{X}_k)\|\,\mathrm{d}t\end{aligned} \tag{A.7}$$

where the first inequality holds due to non-expansive mappings $\mathcal{F}_f(\cdot)$ and $\mathcal{G}_g(\cdot)$ and [1, Theorem 12.2].

The functions f and g are L_f-smooth and L_g-smooth respectively, so $\|\nabla f(\mathbf{x}_1) - \nabla f(\mathbf{x}_2)\| \leq L_f\|\mathbf{x}_1 - \mathbf{x}_2\|$ and $\|\nabla g(\mathbf{y}_1) - \nabla g(\mathbf{y}_2)\| \leq L_g\|\mathbf{y}_1 - \mathbf{y}_2\|$. In such case, we can omit μ_1 and μ_2 and just use the gradients of the functions f and g. In addition, F_k and G_k are Lipschitz-continuous with respect to $\mathbf{x}$ and $\mathbf{y}$, respectively, and thus $\mathbf{P}(\cdot)$ with respect to $\mathcal{X}(t)$ satisfies the Lipschitz continuous condition, i.e., there is a constant $L_1 > \max\{L_f, L_g\}$ such that $\|\mathbf{P}(t,\Theta,\mathcal{X}_1) - \mathbf{P}(t,\Theta,\mathcal{X}_2)\| \leq L_1\|\mathcal{X}_1 - \mathcal{X}_2\|$.

Especially, when f or g is ℓ_1-norm, from our analysis in the main paper, there exists a constant $L_{f_{\mu_1}}$ such that $\|\nabla f_{\mu_1}(\mathbf{x}_1) - \nabla f_{\mu_1}(\mathbf{x}_2)\| \leq L_{f_{\mu_1}}\|\mathbf{x}_1 - \mathbf{x}_2\|$ as well as g. Thus there is always a constant L_2 such that $\mathbf{P}(\cdot)$ satisfies $\|\mathbf{P}(t,\Theta,\mathcal{X}_1) - \mathbf{P}(t,\Theta,\mathcal{X}_2)\| \leq L_2\|\mathcal{X}_1 - \mathcal{X}_2\|$ at discontinuous points, then $\|\mathbf{P}(t,\Theta,\mathcal{X}_1) - \mathbf{P}(t,\Theta,\mathcal{X}_2)\| \leq L\|\mathcal{X}_1 - \mathcal{X}_2\|$ holds, where $L = \max\{L_1, L_2\}$. In practice, our experiments also verify that our networks work on ℓ_1-norm problem models. Plugging this Lipschitz continuous condition into (A.7) yields that:

$$\begin{aligned}\|\boldsymbol{\varepsilon}_{k+1}\| &\leq (1+hL)\,\|\boldsymbol{\varepsilon}_k\| + \frac{1}{2}Q_1h^2 \\ &= (1+hL)^2\,\|\boldsymbol{\varepsilon}_{k-1}\| + (1+hL)\frac{1}{2}Q_1h^2 + \frac{1}{2}Q_1h^2 \\ &\leq \cdots \\ &\leq (1+hL)^{k+1}\,\|\boldsymbol{\varepsilon}_0\| + \left[(1+hL)^k + (1+hL)^{k-1} + \cdots + 1\right]\frac{1}{2}Q_1h^2.\end{aligned} \tag{A.8}$$

More generally,

$$\|\boldsymbol{\varepsilon}_k\| \leq (1+hL)^k\,\|\boldsymbol{\varepsilon}_0\| + \left[\sum_{j=0}^{k}(1+hL)^j\right]\frac{1}{2}Q_1h^2 \leq (1+hL)^k\,\|\boldsymbol{\varepsilon}_0\| + \frac{Q_1h^2}{2hL}\left[(1+hL)^k - 1\right], \quad (k = 1, 2, \cdots). \tag{A.9}$$

Due to $hL > 0$, then $e^{hL} > 1 + hL$, $e^{khL} > (1+hL)^k$, the global error bound of the Euler LADMM scheme (7) from the optimal trajectory is:

$$\|\boldsymbol{\varepsilon}_k\| \leq e^{khL}\,\|\boldsymbol{\varepsilon}_0\| + \frac{Q_1h^2}{2hL}\left[e^{khL} - 1\right]\|\boldsymbol{\varepsilon}_k\| \leq e^{(T-t_0)L}\,\|\boldsymbol{\varepsilon}_0\| + \frac{Q_1h^2}{2hL}\left[e^{(T-t_0)L} - 1\right] \triangleq \mathcal{O}(h). \tag{A.10}$$

The Error Bound of Our Implicit Trapezoid LADMM Scheme: About implicit Trapezoid LADMM scheme (15), we consider its local error bound analysis.

$$\begin{aligned}\|\mathbf{e}_{k+1}\| &= \|\mathcal{X}(t_{k+1}) - \mathcal{X}^*_{k+1}\| \\ &= \|\mathcal{K}\big(\mathcal{X}(t_k) + \int_{t_k}^{t_{k+1}} \mathbf{P}(t,\Theta,\mathcal{X}(t))\mathrm{d}t\big) - \mathcal{K}\big(\mathcal{X}(t_k) + \frac{h}{2}\left[\mathbf{P}(t_k,\Theta_k,\mathcal{X}(t_k)) + \mathbf{P}(t_{k+1},\Theta_k,\mathcal{X}(t_{k+1}))\right]\big)\| \\ &\overset{\text{(b)}}{\leq} \|\mathcal{X}(t_k) + \int_{t_k}^{t_{k+1}} \mathbf{P}(t,\Theta,\mathcal{X}(t))\mathrm{d}t - \mathcal{X}(t_k) - \frac{h}{2}\left[\mathbf{P}(t_k,\Theta_k,\mathcal{X}(t_k)) + \mathbf{P}(t_{k+1},\Theta_k,\mathcal{X}(t_{k+1}))\right]\| \\ &= \|\int_{t_k}^{t_{k+1}} \left\{\mathbf{P}(t,\Theta,\mathcal{X}(t)) - \left[\frac{t-t_{k+1}}{t_k-t_{k+1}}\mathbf{P}(t_k,\Theta_k,\mathcal{X}(t_k)) + \frac{t-t_k}{t_{k+1}-t_k}\mathbf{P}(t_{k+1},\Theta_k,\mathcal{X}(t_{k+1}))\right]\right\}\mathrm{d}t\| \\ &= \|\int_{t_k}^{t_{k+1}} \left[\mathbf{P}(t,\Theta,\mathcal{X}(t)) - P_1(t)\right]\mathrm{d}t\|\end{aligned} \tag{A.11}$$

where $\overset{\text{(b)}}{\leq}$ holds because $\mathcal{F}_f(\cdot)$ and $\mathcal{G}_g(\cdot)$ are non-expansive mappings and so does $\mathcal{K}(\cdot)$, P_1 is the two-point interpolation polynomial of $\mathbf{P}(t,\Theta,\mathcal{X}(t))$, and the other equations hold due to the same reason as in [1]. From the remainder term of interpolation, we obtain

$$\begin{aligned}\|\mathbf{e}_{k+1}\| &= \|\int_{t_k}^{t_{k+1}} \frac{1}{2!}\ddot{\mathbf{P}}(t_k+\xi h)(t-t_k)(t-t_{k+1})\mathrm{d}t\| \\ &= \|\ddot{\mathbf{P}}(t_k+\xi h)\int_{t_k}^{t_{k+1}} \frac{1}{2!}(t-t_k)(t-t_{k+1})\mathrm{d}t\| \\ &= \|-\frac{h^3}{12}\ddot{\mathbf{P}}(t_k+\xi h)\| = \|-\frac{h^3}{12}\mathcal{X}^{(3)}(t_k+\xi h)\| \leq \frac{1}{12}Q_2h^3\end{aligned} \tag{A.12}$$

where $0<\xi<1$, $Q_2 = \max_{t_0\leq t\leq T}\|\mathcal{X}^{(3)}(t)\|$, and $\mathcal{X}^{(3)}$ is the third derivative of $\mathcal{X}(t)$. Therefore, the local error bound of the implicit Trapezoid LADMM scheme is $\mathcal{O}(h^3)$. Next, the global error bound of the implicit Trapezoid LADMM scheme (15) is:

$$\begin{aligned}\|\boldsymbol{\varepsilon}_{k+1}\| =& \|\mathcal{X}(t_{k+1}) - \mathcal{X}_{k+1}\| \\ \overset{\text{(b)}}{\leq}& \|\mathcal{X}(t_k) + \int_{t_k}^{t_{k+1}} \mathbf{P}(t,\Theta,\mathcal{X}(t))\mathrm{d}t - \left[\mathcal{X}_k + \frac{h}{2}\left(\mathbf{P}(t_k,\Theta_k,\mathcal{X}_k) + \mathbf{P}(t_{k+1},\Theta_k,\mathcal{X}_{k+1})\right)\right]\| \\ \leq& \|\boldsymbol{\varepsilon}_k\| + \frac{1}{12}Q_2h^3 + \int_{t_k}^{t_{k+1}} \|\frac{t-t_{k+1}}{t_k-t_{k+1}}\mathbf{P}(t_k,\Theta_k,\mathcal{X}(t_k)) + \frac{t-t_k}{t_{k+1}-t_k}\mathbf{P}(t_{k+1},\Theta_k,\mathcal{X}(t_{k+1})) \\ & - \frac{t-t_{k+1}}{t_k-t_{k+1}}\mathbf{P}(t_k,\Theta_k,\mathcal{X}_k) - \frac{t-t_k}{t_{k+1}-t_k}\mathbf{P}(t_{k+1},\Theta_k,\mathcal{X}_{k+1})\|\mathrm{d}t\end{aligned} \tag{A.13}$$

where $\overset{\text{(b)}}{\leq}$ holds because $\mathcal{F}_f(\cdot)$ and $\mathcal{G}_g(\cdot)$ are non-expansive mappings, and the last inequality holds due to the same reason as in [1]. Due to $|\frac{t-t_{k+1}}{t_k-t_{k+1}}| \leq 1$, we can get $\|\frac{t-t_{k+1}}{t_k-t_{k+1}}(\mathbf{P}(t,\Theta,\mathcal{X}_1) - \mathbf{P}(t,\Theta,\mathcal{X}_2))\| \leq L\|\mathcal{X}_1-\mathcal{X}_2\|$ similarly. Thus,

$$\|\boldsymbol{\varepsilon}_{k+1}\| \leq \left(1+\frac{hL}{2}\right)\|\boldsymbol{\varepsilon}_k\| + \frac{hL}{2}\|\boldsymbol{\varepsilon}_{k+1}\| + \frac{1}{12}Q_2h^3. \tag{A.14}$$

Then, setting $1-\frac{hL}{2}>0$, we have

$$\begin{aligned}\|\boldsymbol{\varepsilon}_{k+1}\| \leq& \frac{1+\frac{hL}{2}}{1-\frac{hL}{2}}\|\boldsymbol{\varepsilon}_k\| + \frac{1}{\left(1-\frac{hL}{2}\right)}\frac{1}{12}Q_2h^3 \\ \leq& \left(\frac{1+\frac{hL}{2}}{1-\frac{hL}{2}}\right)^{k+1}\|\boldsymbol{\varepsilon}_0\| + \left[\left(\frac{1+\frac{hL}{2}}{1-\frac{hL}{2}}\right)^{k} + \cdots + \frac{1+\frac{hL}{2}}{1-\frac{hL}{2}} + 1\right]\frac{1}{\left(1-\frac{hL}{2}\right)}\frac{1}{12}Q_2h^3.\end{aligned} \tag{A.15}$$

More generally,

$$\|\boldsymbol{\varepsilon}_k\| \leq \left(\frac{1+\frac{hL}{2}}{1-\frac{hL}{2}}\right)^{k}\|\boldsymbol{\varepsilon}_0\| + \frac{Q_2h^3}{12hL}\left[\left(\frac{1+\frac{hL}{2}}{1-\frac{hL}{2}}\right)^{k} - 1\right] \quad (k=1,2,\cdots). \tag{A.16}$$

We set $x = \frac{1}{hL}-\frac{1}{2}$, so $\left(\frac{1+\frac{hL}{2}}{1-\frac{hL}{2}}\right)^k = (1+\frac{1}{\frac{1}{hL}-\frac{1}{2}})^{\frac{T-t_0}{h}} = (1+\frac{1}{x})^{\frac{(T-t_0)L}{2}}(1+\frac{1}{x})^{(T-t_0)Lx} \leq e^{(T-t_0)L}$. Then, the global error bound of implicit Trapezoid LADMM scheme (15) from the optimal trajectory can be estimated as follows:

$$\|\boldsymbol{\varepsilon}_k\| \leq e^{(T-t_0)L}\|\boldsymbol{\varepsilon}_0\| + \frac{Q_2h^3}{12hL}\left[e^{(T-t_0)L-1}\right] \triangleq \mathcal{O}(h^2), \quad (k=1,2,\cdots). \tag{A.17}$$

We finish the proof. □

Accelerated unfolded Trapezoid LADMM scheme. Following the idea of the Trapezoid LADMM scheme (15), we intuitively design the accelerated unfolded Trapezoid LADMM scheme as follows:

$$\begin{cases} \widetilde{\mathbf{x}}_k = \mathbf{x}_k + \dfrac{1}{h\theta_k+1}(\mathbf{x}_k - \mathbf{x}_{k-1}), \\ \mathbf{x}_{k+1} = \mathcal{F}_f\big(\widetilde{\mathbf{x}}_k + \dfrac{h^2\beta_k}{2(1+h\theta_k)}(F_k(\widetilde{\mathbf{x}}_k) + F_k(\mathbf{x}_{k+1}))\big), \\ \widetilde{\mathbf{y}}_k = \mathbf{y}_k + \dfrac{1}{h\eta_k+1}(\mathbf{y}_k - \mathbf{y}_{k-1}), \\ \mathbf{y}_{k+1} = \mathcal{G}_g\big(\widetilde{\mathbf{y}}_k + \dfrac{h^2}{2(1+h\eta_k)}(G_k(\widetilde{\mathbf{y}}_k) + G_k(\mathbf{y}_{k+1}))\big), \\ \widetilde{\boldsymbol{\lambda}}_k = \boldsymbol{\lambda}_k + \dfrac{\beta_k}{\beta_k+h}(\boldsymbol{\lambda}_k - \boldsymbol{\lambda}_{k-1}), \\ \boldsymbol{\lambda}_{k+1} = \widetilde{\boldsymbol{\lambda}}_k + \dfrac{h^2\beta_k}{\beta_k+h}(\mathbf{A}\mathbf{x}_{k+1} + \mathbf{y}_{k+1} - \mathbf{b}). \end{cases} \tag{A.18}$$

Similarly, since the accelerated Trapezoid LADMM scheme (A.18) is implicit, we also give an explicit version through the prediction-correction strategy in Algorithm 1 in the main paper.

B. Convergence Analysis of Our Trapezoid LADMM Scheme

In this section, we give the convergence analysis of our non-accelerated explicit Trapezoid LADMM scheme. Firstly, we introduce some definitions and assumptions. Secondly, we give Lemma B.1-B.3 in turn as an assistant to the proof of our Theorem B.1. Thirdly, we prove the convergence of implicit Trapezoid LADMM scheme, i.e., Theorem B.1. And then we analyze the convergence of our explicit scheme. Finally, we analyze the convergence rate of our Trapezoid LADMM schemes. Our proofs in this section refer to [2].

We introduce the variational inequality $\mathrm{VI}(\Omega, \mathbf{F}, \vartheta) := \vartheta(\mathbf{u}) - \vartheta(\mathbf{u}^*) + \langle \boldsymbol{\omega} - \boldsymbol{\omega}^*, \mathbf{F}^*(\boldsymbol{\omega}^*)\rangle \geq 0$, $\forall \boldsymbol{\omega} \in \Omega$, $\boldsymbol{\omega}^* \in \Omega^*$ as a convergence criterion, where $\mathbf{u} = (\mathbf{x}, \mathbf{y})^\top, \boldsymbol{\omega} = (\mathbf{x}, \mathbf{y}, -\boldsymbol{\lambda})^\top, \mathbf{F}^*(\boldsymbol{\omega}) = (\mathbf{A}^\top\boldsymbol{\lambda}, \boldsymbol{\lambda}, \mathbf{A}\mathbf{x}+\mathbf{y}-\mathbf{b})^\top$, $\vartheta(\mathbf{u}) = f(\mathbf{x}) + g(\mathbf{y})$, and Ω^* is the solution set of Problem (2). We define a matrix $D_k = \frac{\theta_k}{h}\mathbf{I} - \frac{\beta_k}{2}\mathbf{W}_k^\top\mathbf{A}$. The convergence of the Trapezoid LADMM scheme necessitates the positive semi-definiteness of the matrix D_k and thus we also define the set $\mathcal{S}(\epsilon) \triangleq \{(\mathbf{W}, \theta, \beta, \eta, h) | \|\mathbf{W}-\mathbf{A}\|_F \leq \epsilon, D \succ 0, \beta, \theta, \eta, h > 0\}$ as a limitation, where $\|\cdot\|_F$ is Frobenius norm. According to these definitions and assumptions, we give Lemma B.1-B.3 in turn as an assistant to the proof of our Theorem B.1.

Lemma B.1. *Let the sequence $\{\boldsymbol{\omega}_k = (\mathbf{x}_k, \mathbf{y}_k, -\boldsymbol{\lambda}_k)\}$ be generated by implicit Trapezoid LADMM scheme (15), then we have the same result as in [2]:*

$$\langle \boldsymbol{\lambda}_{k+1} - \boldsymbol{\lambda}_k, \mathbf{y}_k - \mathbf{y}_{k+1}\rangle \geq 0. \tag{B.19}$$

Lemma B.2. *Let $\{\boldsymbol{\omega}_k = (\mathbf{x}_k, \mathbf{y}_k, -\boldsymbol{\lambda}_k)^\top\}$ be the sequence generated by the implicit Trapezoid LADMM scheme (15), then we have*

$$\vartheta(\mathbf{u}) - \vartheta(\mathbf{u}_{k+1}) + (\boldsymbol{\omega} - \boldsymbol{\omega}_{k+1})^\top[\mathbf{F}_k(\boldsymbol{\omega}_{k+1}) + \mathbf{G}_k(\mathbf{y}_k - \mathbf{y}_{k+1}) + \mathbf{H}_k(\boldsymbol{\omega}_{k+1} - \boldsymbol{\omega}_k)] \geq 0 \tag{B.20}$$

where $\mathbf{F}_k(\boldsymbol{\omega}) = (\mathbf{W}_k^\top\boldsymbol{\lambda}, \boldsymbol{\lambda}, \mathbf{A}\mathbf{x}+\mathbf{y}-\mathbf{b})^\top$, $\mathbf{G}_k(\mathbf{y}) = (\beta_k\mathbf{W}_k^\top\mathbf{y}, \beta_k\mathbf{y}, \mathbf{0})^\top$, and $\mathbf{H}_k$ is defined as:

$$\mathbf{H}_k = \begin{pmatrix} D_k & 0 & 0 \\ 0 & \beta_k\mathbf{I} & 0 \\ 0 & 0 & \frac{1}{\beta_k}\mathbf{I} \end{pmatrix}. \tag{B.21}$$

Note that this conclusion is similar to D-LADMM [2, Lemma 4.1], but our $D_k = \theta_k\mathbf{I} - \frac{\beta_k}{2}\mathbf{W}_k^\top\mathbf{A}$. Lemma B.2 indicates that the quality $\|\boldsymbol{\omega}_k - \boldsymbol{\omega}_{k+1}\|_{\mathbf{H}_k}^2$ can be used to measure how accurate $\boldsymbol{\omega}_{k+1}$ is for being a solution of $\mathrm{VI}(\Omega, \mathbf{F}, \vartheta)$, where $\|\boldsymbol{\omega}\|_{\mathbf{H}}^2 = \langle \boldsymbol{\omega}, \mathbf{H}\boldsymbol{\omega}\rangle$. Since $\mathbf{H}_k$ is positive semi-definite, if $\|\boldsymbol{\omega}_k - \boldsymbol{\omega}_{k+1}\|_{\mathbf{H}_k}^2 = 0$, we can conclude that $\mathbf{H}_k(\boldsymbol{\omega}_{k+1} - \boldsymbol{\omega}_k) = 0$ and $\mathbf{G}_k(\mathbf{y}_k - \mathbf{y}_{k+1}) = 0$, so for $\forall \boldsymbol{\omega}$, $\vartheta(\mathbf{u}) - \vartheta(\mathbf{u}_{k+1}) + \langle \boldsymbol{\omega} - \boldsymbol{\omega}_{k+1}, \mathbf{F}^*(\boldsymbol{\omega}_{k+1})\rangle \geq 0$ on the condition of $\mathbf{W}_k = \mathbf{A}$, which means $\boldsymbol{\omega}_{k+1}$ is a solution of $\mathrm{VI}(\Omega, \mathbf{F}, \vartheta)$, i.e., the solution of Problem (2). Thus, we consider the bound of $\|\boldsymbol{\omega}_k - \boldsymbol{\omega}_{k+1}\|_{\mathbf{H}_k}^2$ in Lemma B.3 below.

Lemma B.3. *Let the sequence $\{\boldsymbol{\omega}_k = (\mathbf{x}_k, \mathbf{y}_k, -\boldsymbol{\lambda}_k)\}$ be generated by the implicit Trapezoid LADMM scheme (15). Suppose that, for any point $\boldsymbol{\omega}^* \in \Omega^*$, there exists suitable parameters $\boldsymbol{\Theta} = \{\mathbf{W}_k, \theta_k, \eta_k, \beta_k\}_{k=1}^K \in \mathcal{S}(\epsilon)$ such that:*

$$\langle \boldsymbol{\omega}_{k+1} - \boldsymbol{\omega}^*, \mathbf{H}_k(\boldsymbol{\omega}_k - \boldsymbol{\omega}_{k+1})\rangle \geq 0, \ \forall k \geq 0 \tag{B.22}$$

where $\mathbf{H}_k$ is given in (B.21), then $\|\boldsymbol{\omega}_k\| < \infty$ holds for all k, and we have:

$$\|\boldsymbol{\omega}_k - \boldsymbol{\omega}^*\|_{\mathbf{H}_k}^2 \geq \|\boldsymbol{\omega}_{k+1} - \boldsymbol{\omega}^*\|_{\mathbf{H}_k}^2 + \|\boldsymbol{\omega}_k - \boldsymbol{\omega}_{k+1}\|_{\mathbf{H}_k}^2. \tag{B.23}$$

The conclusion of our Lemma B.3 is similar to D-LADMM [2, Lemma 4.2]. Lemma B.3 shows that there exist proper learnable parameters that make $\boldsymbol{\omega}_k$ strictly contractive with respect to the solution set Ω^*, which plays a key role in the convergence analysis below.

Theorem B.1 (Convergence of implicit Trapezoid LADMM scheme)**.** *Let the sequence $\{\boldsymbol{\omega}_k = (\mathbf{x}_k, \mathbf{y}_k, -\boldsymbol{\lambda}_k)^\top\}$ be generated by the implicit Trapezoid LADMM scheme (15), then there exists $\boldsymbol{\Theta} \in \mathcal{S}(\epsilon)$ such that $\{\boldsymbol{\omega}_k\}$ converges to a solution $\boldsymbol{\omega}^*$ of Problem (2).*

From the proof of our Theorem B.1, we know that our implicit Trapezoid LADMM scheme (15) converges to the solution of Problem (2). And according to our conference version [3, Eq. (9)], our explicit Trapezoid LADMM scheme, i.e., **Case 1** in Algorithm 1, converges to implicit Trapezoid LADMM scheme when i is large enough. Thus our explicit Trapezoid LADMM scheme achieves convergence. To further prove the convergence rate of the explicit Trapezoid LADMM scheme, we give Theorem B.2.

Theorem B.2 (Convergence Rate of the explicit Trapezoid LADMM scheme)**.** *Let the sequence $\{\boldsymbol{\omega}_k = (\mathbf{x}_k, \mathbf{y}_k, -\boldsymbol{\lambda}_k)^\top\}$ be generated by **Case 1** in Algorithm 1 (non-accelerated explicit Trapezoid LADMM scheme). Suppose that there exist $(\mathbf{A}, \theta^*, \eta^*, \beta^*)$ and $K_0 > 0$ such that for any $k \geq K_0$, the similar EBC in [2] holds. Then there exist suitable parameters $\boldsymbol{\Theta} = \{\mathbf{W}_k, \theta_k, \eta_k, \beta_k\}_{k=1}^K \in \mathcal{S}(\epsilon)$ such that:*

$$dist^2_{\mathbf{H}_{k+1}}(\boldsymbol{\omega}_{k+1}, \Omega^*) < \gamma \; dist^2_{\mathbf{H}_k}(\boldsymbol{\omega}_k, \Omega^*) \tag{B.24}$$

where $dist^2_{\mathbf{H}}(\boldsymbol{\omega}, \Omega^*) = \min_{\boldsymbol{\omega}^* \in \Omega^*} \|\boldsymbol{\omega} - \boldsymbol{\omega}^*\|^2_{\mathbf{H}}$ and γ is a positive constant smaller than 1.

The conclusion of our Theorem B.2 is similiar to [2]. D-LADMM can find appropriate parameters to construct a solution that is closer to Ω^* than the solution produced by fixed parameters at each iteration. Hence, from our Theorem B.2, it is entirely possible for the proposed implicit Trapezoid LADMM scheme to achieve the similar linear convergence rate as D-LADMM [2, Theorem 3], which will be also confirmed in the experiments.

I. Proof of Lemma B.1

Proof. We know that [3, Eq. (9)] converges to implicit Trapezoid LADMM scheme. Thus, we only needs to prove the convergence of implicit Trapezoid LADMM scheme (15). Without loss of generality, we assume $h \equiv 1$ and similar results can be obtained for other values of h. About $\mathbf{y}$-subproblem in (15), by the proximal operator $\text{Prox}_{g\frac{\beta_k}{2}}(\mathbf{z}) = \arg\min_{\mathbf{y}}\{\frac{\beta_k}{4}\|\mathbf{y} - \mathbf{z}\|^2 + g(\mathbf{y})\}$ w.r.t. g, it can be written as:

$$\underset{\mathbf{y}}{\text{argmin}} \left\{ g(\mathbf{y}) + \frac{\beta_k}{4}\|\mathbf{y} - \mathbf{y}_k + \frac{1}{\eta_k}\left[\frac{1}{2}(\mathbf{y}_k + \mathbf{y}_{k+1}) - \mathbf{b} + \mathbf{A}\mathbf{x}_{k+1} + \frac{\boldsymbol{\lambda}_k}{\beta_k}\right]\|^2 \right\}. \tag{B.25}$$

By deriving the optimality conditions of the (B.25), we have

$$g(\mathbf{y}) - g(\mathbf{y}_{k+1}) + \left\langle \mathbf{y} - \mathbf{y}_{k+1}, \frac{\beta_k}{2}(\mathbf{y}_{k+1} - \mathbf{y}_k) + \frac{\beta_k}{2\eta_k}\left[\frac{1}{2}(\mathbf{y}_k + \mathbf{y}_{k+1}) - \mathbf{b} + \mathbf{A}\mathbf{x}_{k+1} + \frac{\boldsymbol{\lambda}_k}{\beta_k}\right] \right\rangle \geq 0. \tag{B.26}$$

By setting $\eta_k = \frac{1}{2}$ and combining $\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + \beta_k\big(\mathbf{A}\mathbf{x}_{k+1} + \mathbf{y}_{k+1} - \mathbf{b}\big)$, we yield that

$$g(\mathbf{y}) - g(\mathbf{y}_{k+1}) + \langle \mathbf{y} - \mathbf{y}_{k+1}, \boldsymbol{\lambda}_{k+1} \rangle \geq 0, \forall \mathbf{y} \in \mathbb{R}^m. \tag{B.27}$$

Obviously, analogous to (B.27), for $\mathbf{y}_k \in \mathbb{R}^m$, we have

$$g(\mathbf{y}) - g(\mathbf{y}_k) + \langle \mathbf{y} - \mathbf{y}_k, \boldsymbol{\lambda}_k \rangle \geq 0, \forall \mathbf{y} \in \mathbb{R}^m. \tag{B.28}$$

Setting $\mathbf{y} = \mathbf{y}_k$ and $\mathbf{y} = \mathbf{y}_{k+1}$ in (B.27) and (B.28), respectively, and combining (B.27) and (B.28), we can get

$$\langle \boldsymbol{\lambda}_{k+1} - \boldsymbol{\lambda}_k, \mathbf{y}_k - \mathbf{y}_{k+1} \rangle \geq 0. \tag{B.29}$$

We finish the proof.

□

II. Proof of Lemma B.2

Proof. Similarly, we take $h \equiv 1$ as an example and define the proximal operator w.r.t f as $\text{Prox}_{f\theta_k}(\mathbf{z}) = \arg\min_{\mathbf{x}}\{\frac{\theta_k}{2}\|\mathbf{x} - \mathbf{z}\|^2 + f(\mathbf{x})\}$. The $\mathbf{x}$-subproblem in the implicit Trapezoid LADMM scheme (15) can be written as:

$$\mathbf{x}_{k+1} = \underset{\mathbf{x}}{\text{argmin}} \left\{ f(\mathbf{x}) + \frac{\theta_k}{2}\|\mathbf{x} - \mathbf{x}_k + \frac{1}{\theta_k}\mathbf{W}_k^\top\big(\boldsymbol{\lambda}_k + \beta_k(\frac{1}{2}\mathbf{A}(\mathbf{x}_k + \mathbf{x}_{k+1}) + \mathbf{y}_k - \mathbf{b})\big)\|^2 \right\}. \tag{B.30}$$

By deriving the optimality conditions of (B.30), we have

$$f(\mathbf{x}) - f(\mathbf{x}_{k+1}) + \left\langle \mathbf{x} - \mathbf{x}_{k+1}, \theta_k(\mathbf{x}_{k+1} - \mathbf{x}_k) + \mathbf{W}_k^\top \left(\boldsymbol{\lambda}_k + \beta_k(\frac{1}{2}\mathbf{A}(\mathbf{x}_k + \mathbf{x}_{k+1}) + \mathbf{y}_k - \mathbf{b})\right) \right\rangle \geq 0. \tag{B.31}$$

Combining $\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + \beta_k\big(\mathbf{A}\mathbf{x}_{k+1} + \mathbf{y}_{k+1} - \mathbf{b}\big)$ yields that:

$$f(\mathbf{x}) - f(\mathbf{x}_{k+1}) + \left\langle \mathbf{x} - \mathbf{x}_{k+1}, \theta_k(\mathbf{x}_{k+1} - \mathbf{x}_k) + \mathbf{W}_k^\top \left(\boldsymbol{\lambda}_{k+1} + \beta_k(\mathbf{y}_k - \mathbf{y}_{k+1}) + \frac{\beta_k}{2}\mathbf{A}(\mathbf{x}_k - \mathbf{x}_{k+1})\right) \right\rangle \geq 0. \tag{B.32}$$

From the $\boldsymbol{\lambda}$-subproblem in the implicit Trapezoid LADMM scheme (15), we can see

$$\frac{1}{\beta_k}(\boldsymbol{\lambda}_{k+1} - \boldsymbol{\lambda}_k) - (\mathbf{A}\mathbf{x}_{k+1} + \mathbf{y}_{k+1} - \mathbf{b}) = \mathbf{0}. \tag{B.33}$$

In summary, combining (B.32), (B.33) and (B.27), we can get

$$\begin{aligned} &\vartheta(\mathbf{u}) - \vartheta(\mathbf{u}_{k+1}) + \begin{pmatrix} \mathbf{x} - \mathbf{x}_{k+1} \\ \mathbf{y} - \mathbf{y}_{k+1} \\ \boldsymbol{\lambda}_{k+1} - \boldsymbol{\lambda} \end{pmatrix} \circ \\ &\left\{ \begin{pmatrix} \mathbf{W}_k^\top \boldsymbol{\lambda}_{k+1} \\ \boldsymbol{\lambda}_{k+1} \\ (\mathbf{A}\mathbf{x}_{k+1} + \mathbf{y}_{k+1} - \mathbf{b}) \end{pmatrix} + \begin{pmatrix} \beta_k \mathbf{W}_k^\top (\mathbf{y}_k - \mathbf{y}_{k+1}) \\ \beta_k(\mathbf{y}_k - \mathbf{y}_{k+1}) \\ \mathbf{0} \end{pmatrix} + \begin{pmatrix} D_k(\mathbf{x}_{k+1} - \mathbf{x}_k) \\ \beta_k(\mathbf{y}_{k+1} - \mathbf{y}_k) \\ \frac{1}{\beta_k}(\boldsymbol{\lambda}_k - \boldsymbol{\lambda}_{k+1}) \end{pmatrix} \right\} \geq 0, \forall \boldsymbol{\omega} \in \Omega. \end{aligned} \tag{B.34}$$

Using the notations of $\boldsymbol{\omega}$, $\mathbf{F}_k(\boldsymbol{\omega})$, $\mathbf{G}_k(\mathbf{y})$ and $\mathbf{H}_k$, we can obtain the assertion (B.20) immediately. Note that our $D_k = \theta_k \mathbf{I} - \frac{\beta_k}{2}\mathbf{W}_k^\top \mathbf{A}$.

□

III. Proof of Lemma B.3

Proof. Please see D-LADMM [2, Lemma 4.2]. □

IV. Proof of Theorem B.1

Proof. From the Lemma B.3, given $\boldsymbol{\omega}^* \in \Omega^*$, there exists proper $\boldsymbol{\Theta} = \{\mathbf{W}_k, \theta_k, \eta_k, \beta_k\}_{k=1}^K$ such that:

$$\sum_{k=0}^{\infty} \|\boldsymbol{\omega}_k - \boldsymbol{\omega}_{k+1}\|_{\mathbf{H}_k}^2 \leq \sum_{k=0}^{\infty} \|\boldsymbol{\omega}_k - \boldsymbol{\omega}^*\|_{\mathbf{H}_k}^2 - \|\boldsymbol{\omega}_{k+1} - \boldsymbol{\omega}^*\|_{\mathbf{H}_k}^2 \leq \|\boldsymbol{\omega}_0 - \boldsymbol{\omega}^*\|_{\mathbf{H}_0}^2 + \sum_{k=0}^{\infty} \left| \|\boldsymbol{\omega}_{k+1} - \boldsymbol{\omega}^*\|_{(\mathbf{H}_{k+1} - \mathbf{H}_k)}^2 \right|. \tag{B.35}$$

This conclusion is the same as D-LADMM too. If we define some large enough $(\theta^*, \eta^*, \beta^*)$ and let $\mathbf{W}_k \to \mathbf{A}$, $\theta_k \to \theta^*, \beta_k \to \beta^*$ following Section B.3 in D-LADMM, then $\sum_{k=0}^{\infty} \left| \|\boldsymbol{\omega}_{k+1} - \boldsymbol{\omega}^*\|_{(\mathbf{H}_{k+1} - \mathbf{H}_k)}^2 \right| < \infty$ and further $\sum_{k=0}^{\infty} \|\boldsymbol{\omega}_k - \boldsymbol{\omega}_{k+1}\|_{\mathbf{H}_k}^2 < \infty$. Thus, we know that the sequence $\{\boldsymbol{\omega}_k\}$ is bounded and there exists a subsequence of $\boldsymbol{\omega}_k$ converges to $\boldsymbol{\omega}_\infty$. Then following [2, Theorem 1], we can obtain $\vartheta(\mathbf{u}) - \vartheta(\mathbf{u}_\infty) + \langle \boldsymbol{\omega} - \boldsymbol{\omega}_\infty, \mathbf{F}^*(\boldsymbol{\omega}_\infty)\rangle \geq 0$ and $\boldsymbol{\omega}_k \to \boldsymbol{\omega}_\infty$ as $k \to \infty$ on the condition of $\mathbf{H}_k \succ 0$, where $\boldsymbol{\omega}_\infty \in \Omega^*$. Thus, our implicit Trapezoid LADMM scheme (15) converges to the solution of Problem (2). □

V. Proof of Theorem B.2

Proof. Without loss of generality, we assume that there exists some $\{\mathbf{W}_k, \theta_k, \eta_k, \beta_k\}$ to make the $\boldsymbol{\omega}_{k+1} \neq \boldsymbol{\omega}_k$. Otherwise we can perturb $(\mathbf{W}_k, \eta_k, \theta_k, \beta_k)$ to make $\boldsymbol{\omega}_{k+1} \neq \boldsymbol{\omega}_k$. Due to $\|\boldsymbol{\omega}_{k+1} - \boldsymbol{\omega}_k\|_{\mathbf{H}_k}^2 \neq 0$, there exists $\kappa_k > 0$ such that:

$$dist_{\mathbf{H}_k}^2(\boldsymbol{\omega}_{k+1}, \Omega^*) \leq \kappa_k \|\boldsymbol{\omega}_{k+1} - \boldsymbol{\omega}_k\|_{\mathbf{H}_k}^2. \tag{B.36}$$

Following (B.23), we have

$$dist_{\mathbf{H}_k}^2(\boldsymbol{\omega}_{k+1}, \Omega^*) \leq dist_{\mathbf{H}_k}^2(\boldsymbol{\omega}_k, \Omega^*) - \|\boldsymbol{\omega}_k - \boldsymbol{\omega}_{k+1}\|_{\mathbf{H}_k}^2. \tag{B.37}$$

Combing the above inequality with (B.36), we get:

$$dist_{\mathbf{H}_k}^2(\boldsymbol{\omega}_{k+1}, \Omega^*) \leq \left(1 + \frac{1}{\kappa_k}\right)^{-1} \left[dist_{\mathbf{H}_k}^2(\boldsymbol{\omega}_k, \Omega^*)\right]. \tag{B.38}$$

According to [2, Theorem 2], we obtain:

$$dist_{\mathbf{H}_{k+1}}^2(\boldsymbol{\omega}_{k+1}, \Omega^*) \leq (1 + \frac{1}{c\alpha_k}) dist_{\mathbf{H}_k}^2(\boldsymbol{\omega}_{k+1}, \boldsymbol{\Omega}^*) \tag{B.39}$$

where $c > 0$ is a constant and $\alpha_k > 1$. Combining (B.38) and (B.39), we can also obtain the monotonically decreasing property:

$$dist_{\mathbf{H}_{k+1}}^2(\boldsymbol{\omega}_{k+1}, \Omega^*) < dist_{\mathbf{H}_k}^2(\boldsymbol{\omega}_k, \Omega^*). \tag{B.40}$$

TABLE C.1
COMPARISON OF THE DENOISING RESULTS IN TERMS OF PSNR (DB) ON 12 IMAGES IN THE WATERLOO BRAGZONE GREYSCALE SETS WITH SALT-AND-PEPPER NOISE RATE 5%. THE BEST, SECOND BEST, AND THIRD BEST RESULTS ARE HIGHLIGHTED IN RED, BLUE, AND GREEN COLORS, RESPECTIVELY.

	Barb	Boat	Bridge	Couple	Finger	Goldhill	Lena	Man	Mandrill	Peppers	Washsat	Zelda	Ave.	Time(s)
D-LADMM ($K=15$ [2])	34.68	33.69	29.05	33.46	32.57	33.88	35.91	33.26	27.17	34.88	35.19	38.62	33.53	0.2733
ELADMM ($K=15$, Ours)	34.92	33.55	29.86	33.57	32.58	33.61	35.85	33.57	27.64	34.96	35.95	38.88	33.75	0.2741
A-ELADMM ($K=15$, Ours)	34.83	35.24	29.59	35.29	36.41	36.39	38.41	34.62	27.18	36.26	39.00	41.65	35.40	0.2868
TLADMM ($K=8$, Ours)	36.69	35.79	30.61	36.30	36.68	37.35	39.54	35.66	26.66	36.91	39.63	41.93	36.15	0.2868
TLADMM ($K=15$, Ours)	36.89	36.30	30.32	36.79	37.18	37.24	39.81	36.00	26.84	37.14	40.14	42.07	36.39	0.5601
A-TLADMM ($K=8$, Ours)	36.70	35.66	30.77	36.06	36.29	37.07	38.39	35.75	28.24	36.95	38.94	41.30	36.01	0.2899
A-TLADMM ($K=15$, Ours)	37.25	35.81	30.81	36.91	37.39	37.93	39.80	36.57	28.45	37.47	40.12	42.22	36.72	0.5635

Then, there are two cases to be discussed.

Case 1: When $k \geq K_0$, under the similar EBC condition: $\text{dist}^2_{\mathbf{H}^*}(\widetilde{\boldsymbol{\omega}}, \Omega^*) \leq \kappa \|\widetilde{\boldsymbol{\omega}} - \boldsymbol{\omega}_k\|^2_{\mathbf{H}^*}$ as in [2], where κ is a positive constant and $\mathbf{H}^*$ is given in (B.21) by setting $(\mathbf{W}_k, \theta_k, \eta_k, \beta_k)$ as $(\mathbf{A}, \theta^*, \eta^*, \beta^*)$, we can get:

$$dist^2_{\mathbf{H}_{k+1}}(\boldsymbol{\omega}_{k+1}, \Omega^*) < \kappa_1 \|\boldsymbol{\omega}_{k+1} - \boldsymbol{\omega}_k\|^2_{\mathbf{H}_k}. \tag{B.41}$$

Case 2: When $k < K_0$, from the convergence of our Trapezoid LADMM scheme in Theorem B.1 and the inequality (B.23), we know that $dist^2_{\mathbf{H}_{k+1}}(\boldsymbol{\omega}_{k+1}, \Omega^*) < \|\boldsymbol{\omega}_0 - \boldsymbol{\omega}^*\|^2_{\mathbf{H}_0} + \sum_{k=0}^{K_0} \left| \|\boldsymbol{\omega}_{k+1} - \boldsymbol{\omega}^*\|^2_{(\mathbf{H}_{k+1} - \mathbf{H}_k)} \right| < \infty$. Hence there exists one constant $C > 0$ such that $dist^2_{\mathbf{H}_{k+1}}(\boldsymbol{\omega}_{k+1}, \Omega^*) < C$. Since $\|\boldsymbol{\omega}_k - \boldsymbol{\omega}_{k+1}\|^2_{\mathbf{H}_k} \neq 0$, there exists one constant $\epsilon > 0$ such that $\|\boldsymbol{\omega}_k - \boldsymbol{\omega}_{k+1}\|^2_{\mathbf{H}_k} > \epsilon$. We immediately have:

$$dist^2_{\mathbf{H}_{k+1}}(\boldsymbol{\omega}_{k+1}, \Omega^*) < \frac{C}{\epsilon} \|\boldsymbol{\omega}_{k+1} - \boldsymbol{\omega}_k\|^2_{\mathbf{H}_k}. \tag{B.42}$$

Letting $\kappa = \max\{\frac{C}{\epsilon}, \kappa_1\}$ and combining (B.37) and (B.40), we get

$$dist^2_{\mathbf{H}_{k+1}}(\boldsymbol{\omega}_{k+1}, \Omega^*) < \left(1 + \frac{1}{\kappa}\right)^{-1} dist^2_{\mathbf{H}_k}(\boldsymbol{\omega}_k, \Omega^*).$$

To sum up, $dist^2_{\mathbf{H}_k}(\boldsymbol{\omega}, \Omega^*)$ converges to zero linearly. Furthermore, combining [3, Eq. (9)], our explicit Trapezoid LADMM, i.e., **Case 1** in Algorithm 1, converges to implicit Trapezoid LADMM scheme (15) when i is large enough. Thus, there exists a set of learnable parameters that helps Algorithm 1 achieve the same linear convergence, which will be confirmed in the experiments. We finish the proof.

□

C. MORE EXPERIMENTAL DETAILS AND RESULTS

In this section, we display the detailed execution of the experiments and more experimental results. All methods are implemented on the NVIDIA GeForce RTX 2080Ti and PyTorch platform.

I. Simulation Experiments

In the simulation experiments, we set $m = 250$ and $d = 500$. For training, we set the batch size to 16 and adopt the stochastic gradient descent (SGD) [4] algorithm with a learning rate $lr = 0.0001$ to train all the networks. The numbers of training and testing samples are set to 10,000 and 1,000, respectively. Each entry in the matrix $\mathbf{A}$ is sampled from i.i.d. Gaussian distribution, namely $A_{i,j} \sim \mathcal{N}(0, 1/m)$, and then we normalize its columns so that they have ℓ_2-norm units. The generation of $\mathbf{x}$ and $\mathbf{y}$ is similar to [2]. For a fair comparison, the matrix $\mathbf{A}$ is the same in all the methods.

II. Natural Image Denoising

Experiment Setting. In this experiment, dictionary $\mathbf{A} \in \mathbb{R}^{256\times512}$ is obtained by clean images [5]. $\mathbf{b}$ in the training set contains 10,000 noisy image blocks with patch size 16×16 for LADMM-type methods. We adopt Adam optimizer [6] with a learning rate $lr = 0.0002$ to train all the methods with a batch size of 16 and we set the number of training epoch to 20 for D-LADMM, our (A-)ELADMM and our (A-)TLADMM. Note that we use $Loss_2$ as training loss function in all of our networks. The test dataset contains 1,024 image blocks for LADMM-type methods and "Time" in Table C.1 - C.3 refers to the GPU time used to restore all test image blocks.

Additional Results. We further show the denoising performance of individual images in dataset WBZG at different noise ratios, and the experimental results are shown in Table C.1 - C.3, which all demonstrate the advantages of our algorithms.

TABLE C.2
COMPARISON OF THE DENOISING RESULTS IN TERMS OF PSNR (DB) ON 12 IMAGES IN THE WATERLOO BRAGZONE GREYSCALE SETS WITH SALT-AND-PEPPER NOISE RATE 10%.

Algorithms	Barb	Boat	Bridge	Couple	Finger	Goldhill	Lena	Man	Mandrill	Peppers	Washsat	Zelda	Ave.	Time (s)
D-LADMM ($K=15$, [2])	32.12	31.16	26.36	31.63	31.44	32.53	35.23	31.06	24.75	34.66	34.82	37.82	31.97	0.2756
D-LADMM ($K=30$)	30.55	30.23	25.67	30.78	30.11	31.46	34.50	30.12	23.24	32.12	34.13	35.62	30.71	0.5685
ELADMM ($K=15$, Ours)	32.07	31.38	26.45	31.49	31.68	32.37	35.67	30.88	23.96	34.20	34.32	37.92	31.87	0.2748
TLADMM ($K=8$, Ours)	33.36	33.29	27.94	32.98	33.65	34.39	37.75	32.94	24.58	34.13	36.43	39.30	33.39	0.2701
TLADMM ($K=15$, Ours)	34.46	33.40	28.26	33.65	34.30	34.58	39.33	33.24	25.07	34.92	37.06	40.27	34.04	0.5031
A-ELADMM ($K=15$, Ours)	32.99	32.89	27.92	32.82	33.53	33.89	37.49	32.56	25.39	34.45	36.47	39.38	33.32	0.2905
A-TLADMM ($K=15$, Ours)	34.38	33.47	28.38	33.59	34.86	34.68	39.63	33.31	25.83	34.18	37.88	40.32	34.21	0.5621

TABLE C.3
COMPARISON OF THE DENOISING RESULTS IN TERMS OF PSNR (DB) ON 12 IMAGES IN THE WATERLOO BRAGZONE GREYSCALE SETS WITH SALT-AND-PEPPER NOISE RATE 15%.

	Barb	Boat	Bridge	Couple	Finger	Goldhill	Lena	Man	Mandrill	Peppers	Washsat	Zelda	Ave.	Time(s)
D-LADMM ($K=15$ [2])	29.11	29.43	23.36	29.55	28.91	30.93	32.72	29.35	20.28	31.84	34.26	36.71	29.70	0.2754
ELADMM ($K=15$, Ours)	29.63	29.54	23.66	29.51	28.77	30.85	32.72	29.82	20.61	31.78	34.85	36.64	29.86	0.2745
A-ELADMM ($K=15$, Ours)	29.51	30.56	24.92	30.29	30.92	31.61	33.48	29.91	23.08	31.72	35.83	37.64	30.79	0.2749
TLADMM ($K=8$, Ours)	29.83	30.52	24.14	29.62	29.54	32.78	34.13	31.32	22.25	33.14	36.06	38.62	31.00	0.2701
TLADMM ($K=15$, Ours)	31.24	31.56	25.81	30.88	32.26	32.93	34.54	30.86	22.28	32.87	36.09	38.69	31.67	0.5022
A-TLADMM ($K=8$, Ours)	31.33	31.34	26.75	31.46	31.98	32.97	34.82	31.36	24.18	32.59	35.54	38.58	31.91	0.2711
A-TLADMM ($K=15$, Ours)	31.78	31.59	26.58	31.97	32.88	33.09	35.16	31.65	24.17	32.77	36.72	39.21	32.30	0.5031

Comparison with MPRNet [7]: We tested the performance of MPRNet on our salt-and-pepper denoising task, and the experimental results are shown in Table C.4. It can be seen that the performance of MPRNet is not as good as our algorithms. This is mainly because the test dataset contains only gray-scale images, while MPRNet requires the input in RGB color space. Using gray-scale images as input to MPRNet will increase interference information. Thus, we conduct the image denoising task on a color FFHQ 256×256-1k dataset [8]. We add the same salt-and-pepper noise on the FFHQ 256×256-1k dataset for our TLADMM, A-TLADMM, and MPRNet [7] on this denoising task and denoising results are shown in Table C.5, where we implemented the source code of MPRNet and D-LADMM as baselines.

TABLE C.4
COMPARISON OF THE PSNR (DB) RESULTS IN THE NATURAL IMAGE DENOISING TASK ON 12 IMAGES IN THE WATERLOO BRAGZONE GREYSCALE SET AT SALT-AND-PEPPER NOISE RATE 15%.

Algorithms	Barb	Boat	Bridge	Couple	Finger	Goldhill	Lena	Man	Mandrill	Peppers	Washsat	Zelda	Ave.
MPRNet [7]	20.48	22.05	19.35	19.89	20.89	19.10	21.76	22.77	18.46	22.26	23.48	22.58	21.09
ELADMM	29.63	29.54	23.66	29.51	28.77	30.85	32.72	29.82	20.61	31.78	34.85	36.64	29.86
A-ELADMM	29.51	30.56	24.92	30.29	30.92	31.61	33.48	29.91	23.08	31.72	35.83	37.64	30.79
TLADMM	31.24	31.56	25.81	30.88	32.26	32.93	34.54	30.86	22.28	32.87	36.09	38.69	31.67
A-TLADMM	31.78	31.59	26.58	31.97	32.88	33.09	35.16	31.65	24.17	32.77	36.72	39.21	32.30

TABLE C.5
COMPARISON OF THE PSNR AND SSIM RESULTS ON THE COLOR FFHQ 256 × 256-1K DATASET [8] AT SALT-AND-PEPPER NOISE RATIOS 5%, 10% AND 15%.

Algorithms	5%		10%		15%		#Params (Millions)
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	
MPRNet [7]	27.03	0.7870	24.27	0.7182	22.65	0.6710	20.1
D-LADMM [2]	34.37	0.9118	32.01	0.8489	28.77	0.7591	2.09
TLADMM (Ours)	36.45	0.9204	33.25	0.8801	30.35	0.8132	1.97
A-TLADMM (Ours)	37.02	0.9258	33.61	0.8886	29.62	0.7942	1.97

From Table C.5, in the case of sparse salt-and-pepper noise, our methods outperform MPRNet at different noise ratios, while MPRNet [7] is a proven efficiency method on the smartphone image denoising datasets such as SIDD https://www.eecs.yorku.ca/ kamel/sidd/dataset.php, but it is a black-box model. In contrast, our algorithms are inspired by traditional optimization, are white-box, provable, and have stronger interpretability like the work [2]. Moreover, the number of parameters of our methods is much less than that of MPRNet, which makes them more adaptable to small-scale datasets.

Ablation study on the FFHQ 256×256-1k dataset: For more robust and concrete evidence of the effectiveness, we also conduct an ablation study on the FFHQ 256 × 256-1k dataset to assess how much our loss $Loss_2$ and trapezoid structure each contribute, and the results are shown in Table C.6. It can be found that only changing the loss function can still maintain the

performance of D-LADMM, which indicates that taking the objective function as the training loss can impose strict constraints on the training procedure and it can be regarded as a substitute for no ground truth. Furthermore, we changed the network structure to our trapezoid structure, and this improvement (1.3dB) is far more significant than above. It is also verified that the trapezoid structure plays a more important role than our loss function.

TABLE C.6
COMPARISON OF DENOISING RESULTS PSNR WITH DIFFERENT K ON THE FFHQ 256×256-1K DATASET AT 10% SALT-AND-PEPPER NOISE.

PSNR \ Layers / Algorithms	$K=9$	$K=12$	$K=15$	$K=18$	Ave.
Original D-LADMM [2]	31.10	32.07	32.01	31.55	31.68
D-LADMM, our $Loss_2$	31.39	32.52	32.30	30.82	31.76
TLADMM, our $Loss_2$	32.63	33.78	33.25	32.81	33.12

III. Natural Image Inpainting

Experiment Setting. We divide the images in the BSDS500 dataset [9] into image blocks and randomly select $N = 50,000$ and $N = 1,000$ 8×8 image patches for training and validation, respectively. We implement other algorithms by ourselves and implement our methods based on the source code of LFISTA [10]. The training batch size is set to 256 and the SGD optimizer is used. The input to our network is triplets $\{\mathbf{b}_i, \mathbf{M}_i, \mathbf{x}^*\}_{i=1}^N$ of the corrupt train patches $\mathbf{b}_i$, their corresponding mask $\mathbf{M}_i$, and the solutions $\mathbf{x}^*$ is generated by 300 iterations of the FISTA on the corrupt signals. For Problem (25), the specific structure of the k-th layer of our TLADMM is as follows:

$$\begin{cases} \mathbf{x}_{k+1}^0 = \mathcal{F}_f\big(\mathbf{x}_k + \dfrac{h\beta_k}{\theta_k} F_k(\mathbf{x}_k)\big), \\ \mathbf{x}_{k+1} = \mathcal{F}_f\big(\mathbf{x}_k + \dfrac{h\beta_k}{2\theta_k}\left[F_k(\mathbf{x}_k) + F_k(\mathbf{x}_{k+1}^0)\right]\big), \\ \mathbf{y}_{k+1}^0 = \dfrac{\beta_k}{1+\beta_k}\big(\mathbf{y}_k - \dfrac{h}{\eta_k}(\mathbf{M}\mathbf{D}\mathbf{x}_{k+1} + \mathbf{y}_k - \mathbf{b} + \dfrac{\boldsymbol{\lambda}_k}{\beta_k})\big), \\ \mathbf{y}_{k+1} = \dfrac{\beta_k}{1+\beta_k}\big(\mathbf{y}_k - \dfrac{h}{2\eta_k}\big((\mathbf{M}\mathbf{D}\mathbf{x}_{k+1} + \mathbf{y}_k - \mathbf{b} + \dfrac{\boldsymbol{\lambda}_k}{\beta_k}) + (\mathbf{M}\mathbf{D}\mathbf{x}_{k+1} + \mathbf{y}_{k+1}^0 - \mathbf{b} + \dfrac{\boldsymbol{\lambda}_k}{\beta_k})\big)\big), \\ \boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + h\beta_k\left(\mathbf{y}_{k+1} + \mathbf{M}\mathbf{D}\mathbf{x}_{k+1} - \mathbf{b}\right) \end{cases} \tag{C.43}$$

where $F_k(\mathbf{x}) = -\left(\mathbf{M}\mathbf{W}_k\right)^\top\left(\mathbf{M}\mathbf{D}\mathbf{x} + \mathbf{y}_k - \mathbf{b} + \frac{\boldsymbol{\lambda}_k}{\beta_k}\right)$, $\mathbf{W}_k$ is a learnable matrix, initialized to $\mathbf{D}$, $\mathbf{A} = \mathbf{M}\mathbf{D}$, $\mathcal{F}_f(\cdot) = ST(\cdot, \tau_1)$, and $\mathcal{G}_g(\cdot) = \frac{\beta_k}{1+\beta_k}\mathcal{I}(\cdot)$. Note that the $\mathbf{y}$-update generalizes its closed solution (i.e., $\mathbf{y}_{k+1} = \frac{\beta_k}{1+\beta_k}\big(-(\mathbf{M}\mathbf{D}\mathbf{x}_{k+1} - \mathbf{b} + \frac{\boldsymbol{\lambda}_k}{\beta_k})\big)$). Furthermore, by adding extrapolation steps, we can obtain the network structure of our A-TLADMM:

$$\begin{cases} \widetilde{\mathbf{x}}_k = \mathbf{x}_k + \frac{1}{h\theta_k+1}(\mathbf{x}_k - \mathbf{x}_{k-1}), \\ \mathbf{x}_{k+1}^0 = \mathcal{F}_f\big(\widetilde{\mathbf{x}}_k + \frac{\beta_k h^2}{1+h\theta_k} F_k(\widetilde{\mathbf{x}}_k)\big), \\ \mathbf{x}_{k+1} = \mathcal{F}_f\big(\widetilde{\mathbf{x}}_k + \frac{\beta_k h^2}{2(1+h\theta_k)}\left[F_k(\widetilde{\mathbf{x}}_k) + F_k(\mathbf{x}_{k+1}^0)\right]\big), \\ \widetilde{\mathbf{y}}_k = \mathbf{y}_k + \frac{1}{h\eta_k+1}(\mathbf{y}_k - \mathbf{y}_{k-1}), \\ \mathbf{y}_{k+1}^0 = \frac{\beta_k}{1+\beta_k}\big(\widetilde{\mathbf{y}}_k - \frac{h^2}{1+h\eta_k}(\mathbf{M}\mathbf{D}\mathbf{x}_{k+1} + \widetilde{\mathbf{y}}_k - \mathbf{b} + \frac{\boldsymbol{\lambda}_k}{\beta_k})\big), \\ \mathbf{y}_{k+1} = \frac{\beta_k}{1+\beta_k}\big(\widetilde{\mathbf{y}}_k - \frac{h^2}{2(1+h\eta_k)}\big((\mathbf{M}\mathbf{D}\mathbf{x}_{k+1} + \widetilde{\mathbf{y}}_k - \mathbf{b} + \frac{\boldsymbol{\lambda}_k}{\beta_k}) + (\mathbf{M}\mathbf{D}\mathbf{x}_{k+1} + \mathbf{y}_{k+1}^0 - \mathbf{b} + \frac{\boldsymbol{\lambda}_k}{\beta_k})\big)\big), \\ \widetilde{\boldsymbol{\lambda}}_k = \boldsymbol{\lambda}_k + \frac{\beta_k}{\beta_k+h}(\boldsymbol{\lambda}_k - \boldsymbol{\lambda}_{k-1}), \\ \boldsymbol{\lambda}_{k+1} = \widetilde{\boldsymbol{\lambda}}_k + \frac{\beta_k h^2}{\beta_k+h}\left(\mathbf{y}_{k+1} + \mathbf{M}\mathbf{D}\mathbf{x}_{k+1} - \mathbf{b}\right). \end{cases} \tag{C.44}$$

Additional Results. More recovered results of some methods for the image inpainting task with 50% missing pixels are shown in Fig. C.2 - C.7. All of these demonstrate the strengths of our algorithms.

IV. Compressive Sensing for Natural Images

In this subsection, we describe some experimental details about our (A-)ELADMM-Net and (A-)TLADMM-Net. To reduce complexity, we avoided the matrix inversions due to closed solutions in the CS model by the following derivation. About $\mathbf{x}$ update, our networks actually solve the $\mathbf{x}$-subproblem as follows:

$$\mathbf{x}_{k+1} = \arg\min_{\mathbf{x}}\{\frac{1}{2}\|\mathbf{c} - \mathbf{\Phi}\mathbf{x}\|^2 + \frac{\beta_k}{2}\|\mathbf{x} - \mathbf{y}_k + \frac{\boldsymbol{\lambda}_k}{\beta_k}\|^2\}. \tag{C.45}$$

We linearize the quadratic terms in (C.45) at $\mathbf{x}_k$ by the Taylor's formula and add a proximal term $\frac{1}{2}\|\mathbf{x}-\mathbf{x}_k\|^2$, thus (C.45) can be approximated as:

$$\mathbf{x}_{k+1} = \arg\min_{\mathbf{x}}\{\frac{1}{2}\|\mathbf{x}-\mathbf{x}_k-\boldsymbol{\Phi}^\top(\mathbf{c}-\boldsymbol{\Phi}\mathbf{x}_k)+\beta_k(\mathbf{x}_k-\mathbf{y}_k+\frac{\boldsymbol{\lambda}_k}{\beta_k})\|^2\}. \tag{C.46}$$

Thus, we can obtain the k-th layer of our TLADMM-Net:

$$\begin{cases} \mathbf{x}_{k+1}^0 = \boldsymbol{\Phi}^\top(\mathbf{c}-\boldsymbol{\Phi}\mathbf{x}_k)+\mathbf{x}_k-h\beta_k\big(\mathbf{x}_k-\mathbf{y}_k+\frac{\boldsymbol{\lambda}_k}{\beta_k}\big), \\ \mathbf{x}_{k+1} = \boldsymbol{\Phi}^\top(\mathbf{c}-\boldsymbol{\Phi}\mathbf{x}_k)+\mathbf{x}_k-\frac{h\beta_k}{2}\big(\mathbf{x}_k-\mathbf{y}_k+\frac{\boldsymbol{\lambda}_k}{\beta_k}+\mathbf{x}_{k+1}^0-\mathbf{y}_k+\frac{\boldsymbol{\lambda}_k}{\beta_k}\big), \\ \mathbf{y}_{k+1}^0 = \mathcal{G}_g\big(\mathbf{y}_k-\frac{h}{\eta_k}\big(\mathbf{x}_{k+1}-\mathbf{y}_k+\frac{\boldsymbol{\lambda}_k}{\beta_k}\big)\big), \\ \mathbf{y}_{k+1} = \mathcal{G}_g\big(\mathbf{y}_k-\frac{h}{2\eta_k}\big(\mathbf{x}_{k+1}-\mathbf{y}_k+\frac{\boldsymbol{\lambda}_k}{\beta_k}+\mathbf{x}_{k+1}-\mathbf{y}_{k+1}^0+\frac{\boldsymbol{\lambda}_k}{\beta_k}\big)\big), \\ \boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k+h\beta_k\big(\mathbf{x}_{k+1}-\mathbf{y}_{k+1}\big) \end{cases} \tag{C.47}$$

where $\mathcal{G}_g(\cdot) = \widetilde{\mathcal{T}}\big(ST(\mathcal{T}(\cdot),\tau_2)\big)$, $\widetilde{\mathcal{T}}$ is the inverse transformation of $\mathcal{T}$, and $\mathcal{F}_f(\cdot) = \boldsymbol{\Phi}^\top(\mathbf{c}-\boldsymbol{\Phi}\mathbf{x}_k)+(\cdot)$. Note that the transformation $\mathcal{T}$ is the same as in [11]. Similarly, the structure of our ELADMM-Net can be obtained. In our experiments, we also initialize β_k as a small value to allow us to find the next point on a larger scale. From (C.45) to (C.47), we successfully avoid the matrix inversions about $\mathbf{x}$-update, which contributes to reducing reconstruction time with little loss of accuracy. Furthermore, by adding the extrapolation steps for $(\mathbf{x},\mathbf{y},\boldsymbol{\lambda})$, the network structure of A-TLADMM-Net can be also obtained.

Calculation of the parameter quantities of our (A-)TLADMM-Net and (A-)ELADMM-Net: for example, under CS ratio $\gamma = 30\%$, the number of parameters of our ELADMM-Net is $(32\times32\times3\times3\times2+32\times3\times3\times2+3)\times10+327\times33\times33 = 546,213$, our A-ELADMM-Net is 546,243, our TLADMM-Net is $(32\times32\times3\times3\times2+32\times3\times3\times2+7)\times10+327\times33\times33 = 546,253$, our A-TLADMM-Net is 546,283, while the number of parameters of ISTA-Net^{++} is $760,220$, COAST is $1,122,056$, and DPC-DUN is about $1,100,000$.

Experiment Setting. We set the size of the image block to 33×33. Then, for a given CS ratio, the corresponding measurement matrix $\boldsymbol{\Phi}$ is constructed by generating a random Gaussian matrix and then orthogonalizing its rows, i.e., $\boldsymbol{\Phi}\boldsymbol{\Phi}^\top = \mathbf{I}$. We initialize $\mathbf{x}_0 = \boldsymbol{\Phi}^\top\mathbf{c}$ as well as $\mathbf{y}_0$. For training, we use the Adam optimizer and train all the methods to 400 epochs with batch size 64. We set a learning rate $lr = 0.0001$ to train our networks for a fair comparison. More comparison results are shown in Table C.8.

An Ablation Experiment for Loss Functions. In this compressive sensing task, we actually have verified the performance of some compared algorithms trained by our $Loss_1$ as shown in Table C.7. It can be seen that MAC-Net and DPC-DUN trained with our $Loss_1$ perform clearly worse than their original versions, respectively. ISTA-Net^{++} and COAST with our $Loss_1$ perform relatively well, but they can only achieve similar levels as their original methods. Based on this analysis, we compared the source code results of the compared algorithms in our main paper.

TABLE C.7
COMPARISON OF NATURAL IMAGE COMPRESSIVE SENSING RESULTS IN TERMS OF PSNR (DB) WITH DIFFERENT LOSS FUNCTIONS AT SAMPLED RATIOS γ =10%, 20%, 30%, 40% AND 50% ON THE TEST DATASETS, BSD68 AND SET11. WE **BOLD** THE HIGHER PSNR FOR THE SAME NETWORK.

Datasets / Algorithms	BSD68						Set11					
	10%	20%	30%	40%	50%	Avg.	10%	20%	30%	40%	50%	Avg.
Original MAC-Net (ECCV2020, [12])	**25.70**	**28.23**	**30.10**	**31.89**	**33.37**	**29.86**	**27.92**	**31.54**	**33.87**	**36.18**	**37.76**	**33.45**
MAC-Net with our $Loss_1$	25.35	27.88	29.85	31.54	33.00	29.52	27.35	30.98	33.30	35.77	37.28	32.94
Original ISTA-Net^{++} (ICME2021, [13])	**26.25**	**29.00**	31.10	33.00	34.85	30.84	**28.34**	**32.33**	34.86	36.94	38.73	**34.24**
ISTA-Net^{++} with our $Loss_1$	26.09	28.93	**31.12**	**33.08**	**34.98**	30.84	27.82	32.11	**34.92**	**37.16**	**39.05**	34.21
Original COAST (TIP2021, [14])	**26.28**	**29.00**	**32.10**	**32.93**	**34.74**	**31.01**	**28.69**	**32.53**	**35.04**	**37.13**	**38.94**	**34.47**
COAST with our $Loss_1$	26.18	28.95	31.99	32.92	34.71	30.95	28.21	32.27	34.89	37.01	38.87	34.25
Original DPC-DUN (TIP 2023, [15])	**26.82**	**29.66**	**31.81**	**33.75**	**35.68**	**31.54**	**29.33**	**32.86**	**35.80**	**37.88**	**39.79**	**35.13**
DPC-DUN with our $Loss_1$	25.78	28.59	30.77	32.61	34.41	30.43	27.62	31.17	34.07	36.01	37.89	33.35

V. Compressive Sensing on Speech Data

For speech data, the network structures of our (A-)TLADMM-Net are the same as those in natural image compressive sensing. Following [24], we add zero-mean Gaussian noise with standard deviation std $= 10^{-4}$ to the measurements and choose the CS ratio 25% and 40% for analysis. We use a column orthogonal matrix $\boldsymbol{\Phi}$ to downsample raw speech data and treat soft-thresholding parameters as trainable values.

An Ablation Experiment for Loss Functions. In this compressive sensing on speech data experiment, when training the compared algorithms ISTA-Net^{+}, HSSE and ADMM-DAD, the experimental performance using our $Loss_1$ is slightly better than that of using the original loss, as shown in Table C.9. For a fair comparison, all the methods use the $Loss_1$ as the loss function.

TABLE C.8
COMPARISON OF IMAGE COMPRESSIVE SENSING RESULTS IN TERMS OF PSNR (DB) UNDER DIFFERENT SAMPLED RATIOS $\gamma = 10\%, 20\%, 30\%, 40\%$ AND 50% ON THE BSD68 AND SET11 DATASETS. AS WE CAN SEE, OUR NETWORKS ACHIEVE MUCH BETTER RESULTS THAN OTHER METHODS IN THE CASES OF ALL THE SAMPLED RATIOS.

Algorithms \ Datasets	BSD68						Set11					
	10%	20%	30%	40%	50%	Avg.	10%	20%	30%	40%	50%	Avg.
LDAMP (NeurIPS2017, [16])	23.94	27.74	30.28	32.12	32.89	29.39	24.71	30.65	33.87	36.03	36.60	32.37
ISTA-Net$^+$ (CVPR2018, [11])	25.24	28.00	30.20	32.10	33.93	29.89	26.57	30.85	33.74	36.05	38.05	33.05
DPDNN (TPAMI2019, [17])	24.81	27.28	29.22	30.99	32.74	29.01	26.09	29.75	32.37	34.69	36.83	31.95
GDN (TCI2019, [18])	25.19	27.95	29.88	32.07	34.09	29.84	26.03	30.16	32.95	35.25	37.60	32.40
SCSNet (CVPR2019, [19])	27.28	29.01	31.87	33.86	35.77	31.56	28.48	31.95	34.62	36.92	39.01	34.20
DPA-Net (TIP2020, [20])	25.33	-	29.58	-	-	-	27.66	-	33.60	-	-	-
MAC-Net (ECCV2020, [12])	25.70	28.23	30.10	31.89	33.37	29.86	27.92	31.54	33.87	36.18	37.76	33.45
COAST (TIP2021, [14])	26.28	29.00	32.10	32.93	34.74	31.01	28.69	32.53	35.04	37.13	38.94	34.47
ISTA-Net^{++} (ICME2021, [13])	26.25	29.00	31.10	33.00	34.85	30.84	28.34	32.33	34.86	36.94	38.73	34.24
GPX-ADMM-Net (EUSIPCO2021, [21])	25.30	27.79	29.32	31.99	33.25	29.53	27.46	31.36	33.85	36.28	38.32	33.45
HSSE (TNNLS2022, [22])	26.29	28.99	32.01	32.75	34.75	30.96	28.69	-	34.92	37.04	38.92	-
LGSR (TNNLS2023, [23])	26.33	29.78	32.30	34.32	36.33	31.81	28.24	-	34.93	37.10	38.99	-
DPC-DUN (TIP 2023, [15])	26.82	29.66	31.81	33.75	35.68	31.54	29.33	32.86	35.80	37.88	39.79	35.13
ELADMM-Net ($K=20$, Ours)	27.01	29.53	32.01	33.89	35.82	31.65	28.34	32.51	34.72	37.32	38.71	34.32
A-ELADMM-Net ($K=10$, Ours)	27.33	29.81	32.40	34.10	36.22	31.97	28.54	32.70	34.95	37.56	38.89	34.53
TLADMM-Net ($K=10$, Ours)	27.76	30.38	32.68	34.78	36.82	32.48	28.95	32.81	35.73	38.18	40.32	35.19
A-TLADMM-Net ($K=10$, Ours)	27.67	30.60	32.82	34.90	36.95	32.59	29.20	33.02	35.87	38.24	40.42	35.35
A-TLADMM-Net ($K=20$, Ours)	27.94	30.86	33.21	35.36	37.41	32.96	29.43	33.35	36.14	38.61	40.89	35.68

TABLE C.9
COMPARISON OF THE TEST MSE RESULTS ($\times 10^{-2}$ AND $\times 10^{-4}$) ON THE TIMIT AND SPEECHCOMMANDS DATASETS WITH DIFFERENT LOSS FUNCTIONS AT CS RATIOS $\gamma = 25\%, 40\%$.

Algorithms \ Datasets	TIMIT		SpeechCommands	
	25%	40%	25%	40%
Original ISTA-Net$^+$ [11]	2.245	2.049	0.589	0.474
ISTA-Net$^+$ with our $Loss_1$	**2.232**	**2.029**	**0.584**	**0.462**
Original HSSE [22]	1.214	0.931	0.698	**0.347**
HSSE with our $Loss_1$	**0.911**	**0.829**	**0.476**	0.362
Original ADMM-DAD [24]	0.794	0.431	0.254	0.146
ADMM-DAD with our $Loss_1$	**0.791**	**0.424**	**0.252**	**0.134**

More visual results of the spectrograms are shown in Fig. C.8 and the reconstruction examples of the raw speech samples under $\gamma = 40\%$ on dataset TIMIT are provided in another folder in https://github.com/Weixin-An/A-TLADMM-Net/tree/master/Speech%20CS. All of these verify that our algorithms can distinguish more frequencies than baseline methods.

VI. Compressive Sensing MRI

About CS MRI, the iterations of (A-)ELADMM-Net and (A-)TLADMM-Net can also be derived in the same way. The sampling mode is used pseudo radial sampling. We evaluate the Structure Similarity Index Measure (SSIM) as follows:

$$\mathrm{SSIM}(\mathbf{x}_K, \mathbf{x}^*) = \frac{(2\mu_{\mathbf{x}_K}\mu_{\mathbf{x}^*} + c_3)(2\delta_{\mathbf{x}_K\mathbf{x}^*} + c_4)}{(\mu^2_{\mathbf{x}_K}\mu^2_{\mathbf{x}^*} + c_3)(\delta^2_{\mathbf{x}_K}\delta^2_{\mathbf{x}^*} + c_4)},$$

between the network output $\mathbf{x}_K$ and the ground truth, where $\mu_{\mathbf{x}_K}$ and $\mu_{\mathbf{x}^*}$ represent the mean values of $\mathbf{x}_K$ and $\mathbf{x}^*$ respectively; $\delta^2_{\mathbf{x}_K}$ and $\delta^2_{\mathbf{x}^*}$ represent the variances of $\mathbf{x}_K$ and $\mathbf{x}^*$ respectively; $\delta_{\mathbf{x}_K\mathbf{x}^*}$ represents the covariance of $\mathbf{x}_K$ and $\mathbf{x}^*$, and the constant c_3, c_4 prevents the exception of dividing by 0.

An Ablation Experiment for Loss Functions. In this MRI CS experiment, ADMM-Net uses the averaged normalized root mean square error (NRMSE). For a fair comparison, we replace its loss function with our $Loss_1$ and the experimental results are almost the same, as shown in Table C.10. As for ISTA-type methods, we replace their ℓ_2-norm loss with our $Loss_1$, and the experimental performance is slightly worse. Thus, we still used their original loss functions to train them.

We also test all the methods at the same time cost and the results are shown in Fig. C.1. At relatively high CS ratios, our A-TLADMM-Net always reconstructs higher quality images than the compared methods. Moreover, our networks still perform much better than ADMM-Net at the same time cost. Our TLADMM-Net also enjoys better performance than ELADMM-Net. At relatively low CS ratios, our TLADMM-Net and A-TLADMM-Net are competitive with the reconstruction results of ISTA-Net$^+$.

More visual results are shown in Fig. C.9. This experiment verified that the (accelerated) Trapezoid LADMM schemes are more accurate than the (accelerated) Euler LADMM schemes, which also inspires us to try more accurate numerical discretizations, such as the linear multi-step discretization, to improve performance on the CS MRI task.

TABLE C.10
COMPARISON OF TEST PSNR (DB) RESULTS FOR COMPRESSIVE SENSING MRI WITH DIFFERENT LOSS FUNCTIONS.

Algorithms	CS Ratio γ			
	20%	30%	40%	50%
Original ADMM-Net [25]	37.17	39.84	**41.56**	**43.00**
ADMM-Net with our $Loss_1$	**37.31**	**39.92**	41.55	42.98
Original ISTA-Net [11]	**38.30**	**40.52**	**42.12**	**43.60**
ISTA-Net with our $Loss_1$	38.11	40.37	42.04	43.55
Original ISTA-Net$^+$ [11]	38.73	**40.89**	42.52	**44.09**
ISTA-Net$^+$ with our $Loss_1$	**38.78**	40.75	**42.53**	44.05

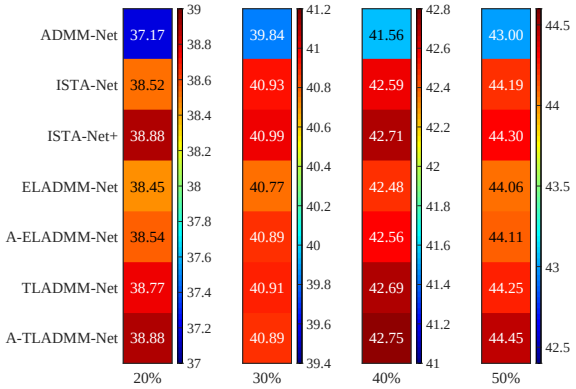


Fig. C.1. Comparison of test PSNR (dB) results for CS-MRI on the brain dataset at the same time cost.

REFERENCES

[1] J. Stoer, R. Bartels, W. Gautschi, R. Bulirsch, and C. Witzgall, *Introduction to Numerical Analysis*. Springer New York, 2013.
[2] X. Xie, J. Wu, G. Liu, Z. Zhong, and Z. Lin, "Differentiable linearized admm," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6902–6911.
[3] W. An, Y. Yue, Y. Liu, F. Shang, and H. Liu, "A numerical des perspective on unfolded linearized admm networks for inverse problems," in *Proc. ACM Int. Conf. Multimed.*, 2022, pp. 5065–5073.
[4] H. Robbins and S. Monro, "A stochastic approximation method," *The annals of mathematical statistics*, pp. 400–407, 1951.
[5] Y. Xu and W. Yin, "A fast patch-dictionary method for whole image recovery," *Inverse Probl. Imaging*, vol. 10, no. 2, pp. 563–583, 2016.
[6] K. Cho, B. van Merrienboer, C. Gulcehre, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," in *Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, 2014.
[7] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, M.-H. Yang, and L. Shao, "Multi-stage progressive image restoration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 14 821–14 831.
[8] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4401–4410.
[9] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2, 2001, pp. 416–423.
[10] A. Aberdam, A. Golts, and M. Elad, "Ada-lista: Learned solvers adaptive to varying models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 12, pp. 9222–9235, 2021.
[11] J. Zhang and B. Ghanem, "Ista-net: Interpretable optimization-inspired deep network for image compressive sensing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1828–1837.
[12] J. Chen, Y. Sun, Q. Liu, and R. Huang, "Learning memory augmented cascading network for compressed sensing of images," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 513–529.
[13] D. You, J. Xie, and J. Zhang, "Ista-net++: flexible deep unfolding network for compressive sensing," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2021, pp. 1–6.
[14] D. You, J. Zhang, J. Xie, B. Chen, and S. Ma, "Coast: Controllable arbitrary-sampling network for compressive sensing," *IEEE Trans. Image Process.*, vol. 30, pp. 6066–6080, 2021.
[15] J. Song, B. Chen, and J. Zhang, "Dynamic path-controllable deep unfolding network for compressive sensing," *IEEE Trans. Image Process.*, vol. 32, pp. 2202–2214, 2023.
[16] C. A. Metzler, A. Mousavi, and R. G. Baraniuk, "Learned d-amp: principled neural network based compressive image recovery," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1770–1781.
[17] W. Dong, P. Wang, W. Yin, G. Shi, F. Wu, and X. Lu, "Denoising prior driven deep neural network for image restoration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 10, pp. 2305–2318, 2018.
[18] D. Gilton, G. Ongie, and R. Willett, "Neumann networks for linear inverse problems in imaging," *IEEE Trans. Comput. Imaging*, vol. 6, pp. 328–343, 2019.
[19] W. Shi, F. Jiang, S. Liu, and D. Zhao, "Scalable convolutional neural network for image compressed sensing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 12 290–12 299.
[20] Y. Sun, J. Chen, Q. Liu, B. Liu, and G. Guo, "Dual-path attention network for compressed sensing image reconstruction," *IEEE Trans. Image Process.*, vol. 29, pp. 9482–9495, 2020.
[21] S.-W. Hu, G.-X. Lin, and C.-S. Lu, "Gpx-admm-net: Admm-based neural network with generalized proximal operator," in *European Signal Proces. Conf.*, 2021, pp. 2055–2059.
[22] Z. Zha, B. Wen, X. Yuan, J. Zhou, C. Zhu, and A. C. Kot, "A hybrid structural sparsification error model for image restoration," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 33, no. 9, pp. 4451–4465, 2022.
[23] Zha, Zhiyuan and Wen, Bihan and Yuan, Xin and Zhou, Jiantao and Zhu, Ce and Kot, Alex Chichung, "Low-rankness guided group sparse representation for image restoration," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 34, no. 10, pp. 7593–7607, 2023.

Original image Corrupted, 11.1 D-LADMM, 26.51 ELADMM, 26.54 A-ELADMM, 28.36 TLADMM, 27.08 A-TLADMM, 30.00

Fig. C.2. The recovered results (dB) of the methods for image inpainting tasks with 50% missing pixels on Barbara.

Original image Corrupted, 10.6 D-LADMM, 30.91 ELADMM, 30.84 A-ELADMM, 31.86 TLADMM, 31.58 A-TLADMM, 32.86

Fig. C.3. The recovered results (dB) of the methods for image inpainting tasks with 50% missing pixels on Boat.

Original image Corrupted, 10.2 D-LADMM, 34.11 ELADMM, 34.15 A-ELADMM, 33.90 TLADMM, 34.75 A-TLADMM, 35.26

Fig. C.4. The recovered results (dB) of the methods for image inpainting tasks with 50% missing pixels on House.

Original image Corrupted, 10.9 D-LADMM, 34.31 ELADMM, 34.55 A-ELADMM, 31.31 TLADMM, 35.08 A-TLADMM, 32.36

Fig. C.5. The recovered results (dB) of the methods for image inpainting tasks with 50% missing pixels on Lena.

Original image Corrupted, 10.9 D-LADMM, 29.84 ELADMM, 29.75 A-ELADMM, 29.87 TLADMM, 30.09 A-TLADMM, 30.75

Fig. C.6. The recovered results (dB) of the methods for image inpainting tasks with 50% missing pixels on Peppers.

Original image Corrupted, 11.2 D-LADMM, 31.14 ELADMM, 31.38 A-ELADMM, 31.34 TLADMM, 31.94 A-TLADMM, 32.28

Fig. C.7. The recovered results (dB) of the methods for image inpainting tasks with 50% missing pixels on Couple.

[24] V. Kouni, G. Paraskevopoulos, H. Rauhut, and G. C. Alexandropoulos, "Admm-dad net: a deep unfolding network for analysis compressed sensing," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2022, pp. 1506–1510.

[25] Y. Yang, J. Sun, H. Li, and Z. Xu, "Deep admm-net for compressive sensing mri," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 10–18.

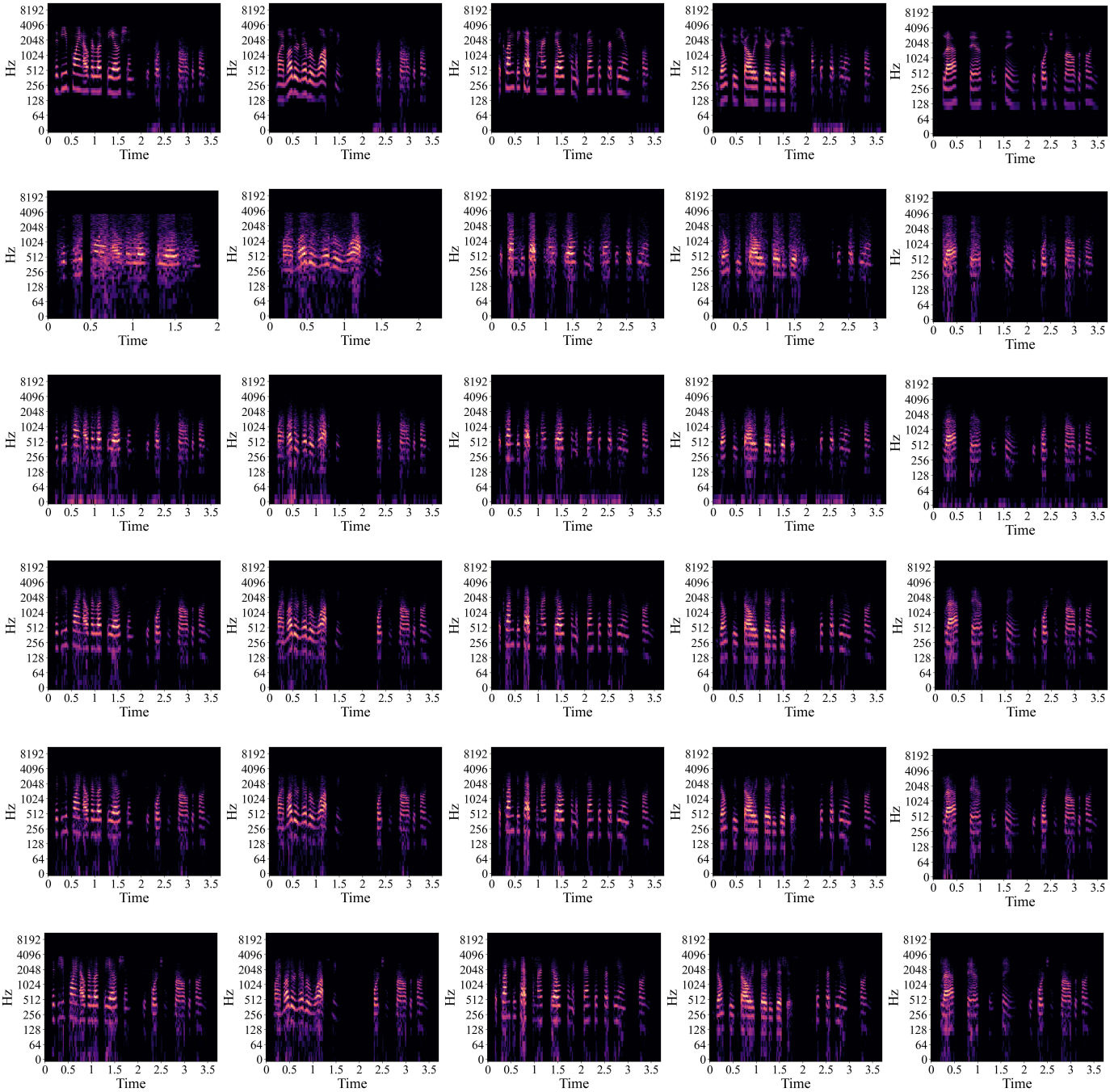
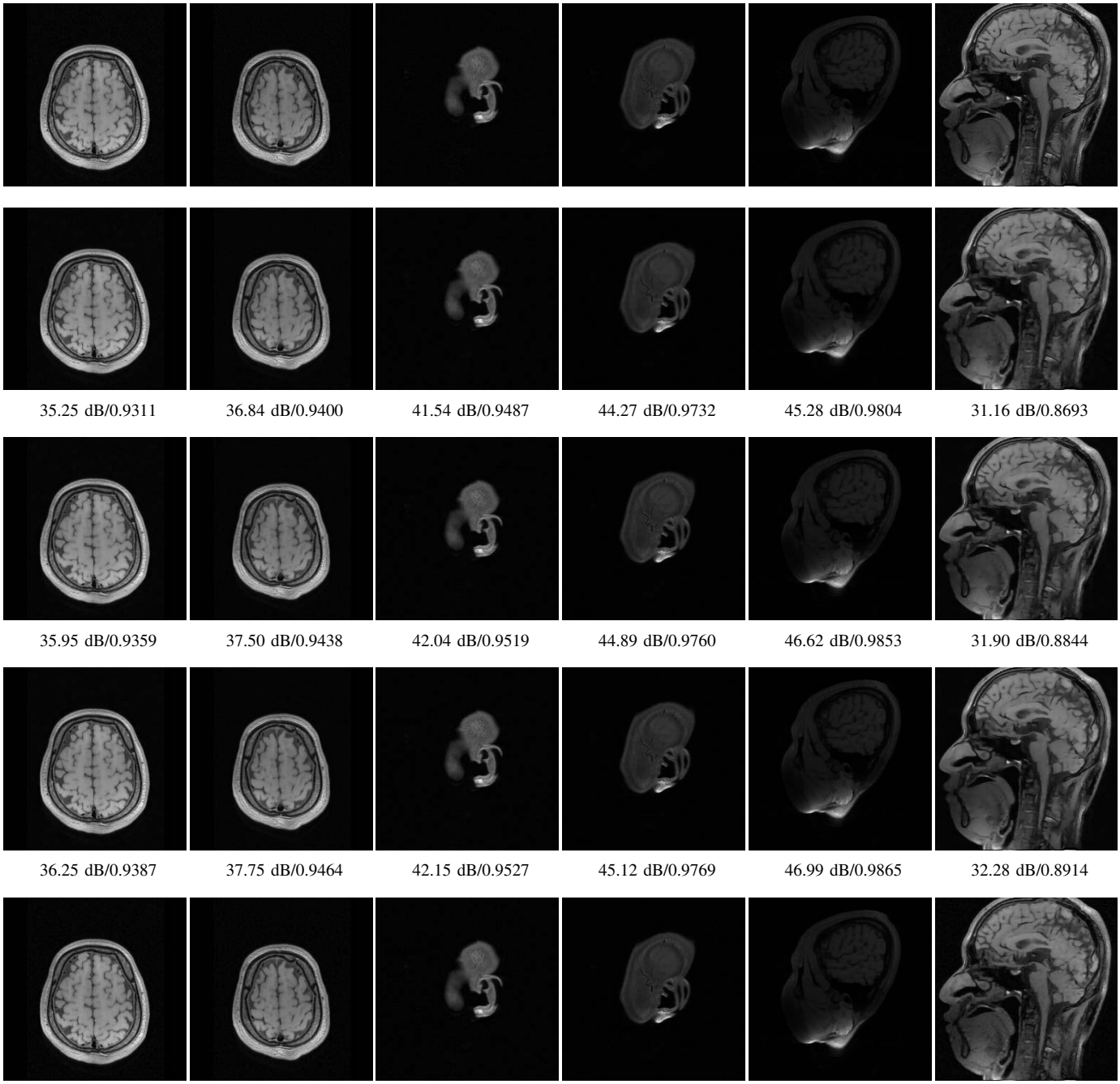


Fig. C.8. Comparison of the visual results for the speech CS task at $\gamma = 40\%$ on TIMIT. From top to bottom, the spectrograms of Ground Truth, ADMM-DAD [24], ELADMM-Net (Ours), A-ELADMM-Net (Ours), TLADMM-Net (Ours), and A-TLADMM-Net (Ours).



35.25 dB/0.9311 | 36.84 dB/0.9400 | 41.54 dB/0.9487 | 44.27 dB/0.9732 | 45.28 dB/0.9804 | 31.16 dB/0.8693

35.95 dB/0.9359 | 37.50 dB/0.9438 | 42.04 dB/0.9519 | 44.89 dB/0.9760 | 46.62 dB/0.9853 | 31.90 dB/0.8844

36.25 dB/0.9387 | 37.75 dB/0.9464 | 42.15 dB/0.9527 | 45.12 dB/0.9769 | 46.99 dB/0.9865 | 32.28 dB/0.8914

36.85 dB/0.9260 | 38.26 dB/0.9377 | 42.55 dB/0.9561 | 45.25 dB/0.9768 | 46.87 dB/0.9855 | 32.84 dB/0.9000

Fig. C.9. Examples of visual results and PSNR/SSIM of the MRI compressive sensing task on the brain dataset with CS ratio $\gamma = 20\%$. From top to bottom, the results of Ground Truth, ELADMM-Net (Ours), A-ELADMM-Net (Ours), TLADMM-Net (Ours), and A-TLADMM-Net (Ours).