A Primal-Dual Algorithm for Image Reconstruction with Input-Convex Neural Network Regularizers

Matthias J. Ehrhardt

Department of Mathematical Sciences, University of Bath, UK

5 May, 2026

Joint work with:

H. S. Wong (Bath), S. Mukherjee (Kharagpur)

Hok Shing Wong

Engineering and Physical Sciences Research Council

m4DL

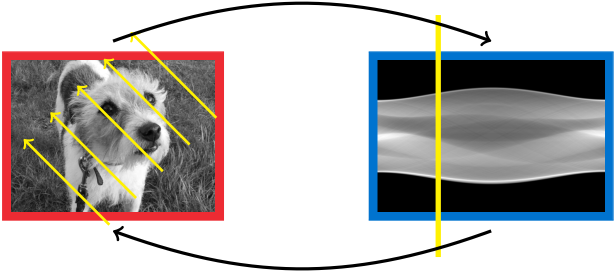
UNIVERSITY OF BATH

Inverse Problems

desired solution

mathematical model $\rightarrow$ $Ax = b$ $\leftarrow$ observed data

Forward problem: compute b given x



Inverse problem: recover x given b

Ray transform (e.g. CT, PET): $Ax(L) = \int_L x(r)dr$

Variational Regularization

Variational regularization approximate x^* via

$$\hat{x} \in \arg\min_x \left\{ \mathcal{D}(Ax, b) + \lambda \mathcal{R}(x) \right\}$$

$\mathcal{D}$ **data fidelity**: related to noise statistics

$\mathcal{R}$ **regularizer**: penalizes unwanted features, stability

$\lambda \geq 0$ **regularization parameter**: weights data and regularizer

The **regularizer** can be

- hand-crafted: e.g., Tikhonov, TV, TGV, ...
- data-driven: learned regularization parameter, learned regularizer

Examples of Regularisers

Total Variation (TV) Rudin et al. 1992

$$\mathcal{R}(x) = \int \|\nabla x(r)\|_2 dr = W\sigma(Vx)$$

noisy

TV regularized

- Denoising without oversmoothing edges
- Results cartoonish

Examples of Regularisers

Total Variation (TV) Rudin et al. 1992

$$\mathcal{R}(x) = \int \|\nabla x(r)\|_2 dr = W\sigma(Vx)$$

Fields-of-Experts (FoE) Roth and Black '05

$$\mathcal{R}_\theta(x) = \sum_{k=1}^{K} \alpha_k \int \phi_{\gamma_k}((\kappa_k * x)(r))dr = W\sigma(Vx)$$

To make $\mathcal{R}_\theta$ convex: ϕ_{γ_k} convex, $\alpha_k \geq 0$

Examples of Regularisers

Total Variation (TV) Rudin et al. 1992

$$\mathcal{R}(x) = \int \|\nabla x(r)\|_2 dr = W\sigma(Vx)$$

Fields-of-Experts (FoE) Roth and Black '05

$$\mathcal{R}_\theta(x) = \sum_{k=1}^{K} \alpha_k \int \phi_{\gamma_k}((\kappa_k * x)(r))dr = W\sigma(Vx)$$

Input Convex Neural Network (ICNN) Amos et al. '17

$$z_1 = \sigma(V_1 x + b_1)$$

$$z_k = \sigma(W_k z_{k-1} + V_k x + b_k) \quad k = 2, \ldots, K$$

$$\mathcal{R}_\theta(x) = z_K$$

To make $\mathcal{R}_\theta$ convex: σ convex and nondecreasing, $W_k \geq 0$

Examples of Regularisers

Total Variation (TV) Rudin et al. 1992

$$\mathcal{R}(x) = \int \|\nabla x(r)\|_2 dr = W\sigma(Vx)$$

Fields-of-Experts (FoE) Roth and Black '05

$$\mathcal{R}_\theta(x) = \sum_{k=1}^{K} \alpha_k \int \phi_{\gamma_k}((\kappa_k * x)(r))dr = W\sigma(Vx)$$

Input Convex Neural Network (ICNN) Amos et al. '17

$$z_1 = \sigma(V_1 x + b_1)$$

$$z_k = \sigma(W_k z_{k-1} + V_k x + b_k) \quad k = 2, \ldots, K$$

$$\mathcal{R}_\theta(x) = z_K$$

To make $\mathcal{R}_\theta$ convex: σ convex and nondecreasing, $W_k \geq 0$

CRR Goujon et al. '23, CNN Lunz et al. '18, TDV Kobler et al. '21, wCRR Goujon et al. '24, wICNN Shumaylov et al. '24, IDCNN Zhang and Leong '25 ...

How to solve variational problem?

Given $\mathcal{R}_\theta$, how to solve

$$\min_x \{\mathcal{D}(Ax, b) + \lambda \mathcal{R}_\theta(x)\}$$

Properties

- nonsmooth (data fit, activation functions)
- subgradient methods are slow
- deep network (vanishing gradients)
- regularizer not prox-friendly

$$\mathrm{prox}_{\tau \mathcal{R}_\theta}(z) = \arg\min_x \left\{ \frac{1}{2} \|x - z\|_2^2 + \tau \mathcal{R}_\theta(x) \right\}$$

Goal: Can we do better by exploiting the specific ICNN structure?

Generalized ICNN

Input Convex Neural Network (ICNN)

$$
\begin{aligned}
z_1 &= \sigma(V_1 x + b_1) \\
z_k &= \sigma(W_k z_{k-1} + V_k x + b_k) \quad k = 2, \ldots, K \\
\mathcal{R}_\theta(x) &= z_K
\end{aligned}
$$

constraints on σ and W_k

Generalized ICNN

Input Convex Neural Network (ICNN)

$$
\begin{aligned}
z_1 &= \sigma(V_1 x + b_1) \\
z_k &= \sigma(W_k z_{k-1} + V_k x + b_k) \quad k = 2, \ldots, K \\
\mathcal{R}_\theta(x) &= z_K
\end{aligned}
$$

constraints on σ and W_k

Generalized ICNN

$$
\begin{aligned}
z_1 &= \psi_1(x) \\
z_k &= \psi_k(x, z_1, \ldots, z_{k-1}) \quad k = 2, \ldots, K \\
\mathcal{R}_\theta(x) &= z_K
\end{aligned}
$$

Special cases:

- ICNN
- skip connections: $z_k = z_{k-1} + \sigma(W_k z_{k-1} + V_k x + b_k)$

Generalized ICNN

Generalized ICNN

$$
\begin{aligned}
z_1 &= \psi_1(x) \\
z_k &= \psi_k(x, z_1, \dots, z_{k-1}) \quad k = 2, \dots, K \\
\mathcal{R}_\theta(x) &= z_K
\end{aligned}
$$

Assumption: 1) ψ_k convex;
2) $\psi_k^x := \psi_k^x(\cdot)$ nondecreasing: $v \leq w \Rightarrow \psi_k^x(v) \leq \psi_k^x(w)$

Prop: Under the Assumption, $\mathcal{R}_\theta$ is convex.

Reformulation 1

Option 0:

$$\min_{x} \{\mathcal{D}(Ax, y) + \lambda \mathcal{R}_\theta(x)\}$$

Option 1:

$$\min_{x,z} \{D(Ax, y) + \alpha z_K\}$$

$$\text{s.t.} \quad z_1 = \psi_1(x)$$

$$z_k = \psi_k(x, z_1, \dots, z_{k-1}) \quad k = 2, \dots, K$$

Carreira-Perpinan and Wang '14, Taylor et al. '16, Zhang and Brand '17, Askari et al. '18, Li et al. '19, Wang and Benning '23

Reformulation 1

Option 0:

$$\min_x \{\mathcal{D}(Ax, y) + \lambda \mathcal{R}_\theta(x)\}$$

Option 1:

$$\min_{x,z} \{D(Ax, y) + \alpha z_K\}$$

$$\text{s.t.} \quad z_1 = \psi_1(x)$$

$$z_k = \psi_k(x, z_1, \dots, z_{k-1}) \quad k = 2, \dots, K$$

Carreira-Perpinan and Wang '14, Taylor et al. '16, Zhang and Brand '17, Askari et al. '18, Li et al. '19, Wang and Benning '23

- Constraints: $(x, z_1) \in \mathrm{grph}(\psi_1) = \{t, s \mid s = \psi_1(t)\}$ etc
- loss of convexity
  - Option 0: convex if datafit is convex in x
  - Option 1: nonconvex

Reformulation 2

Option 1:
$$\min_{x,z} \{D(Ax, y) + \alpha z_K\}$$
$$\text{s.t.} \quad z_1 = \psi_1(x)$$
$$z_k = \psi_k(x, z_1, \ldots, z_{k-1}) \quad k = 2, \ldots, K$$

Constraint: $(x, z_1) \in \text{grph}(\psi_1) = \{t, s \mid s = \psi_1(t)\}$ etc

Reformulation 2

Option 1:

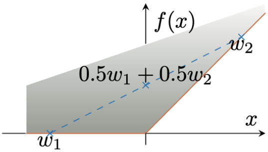
$$\min_{x,z} \{D(Ax, y) + \alpha z_K\}$$

$$\text{s.t.} \quad z_1 = \psi_1(x)$$

$$z_k = \psi_k(x, z_1, \ldots, z_{k-1}) \quad k = 2, \ldots, K$$

Constraint: $(x, z_1) \in \text{grph}(\psi_1) = \{t, s \mid s = \psi_1(t)\}$ etc

Idea epigraphs of convex functions are convex: relax constraint



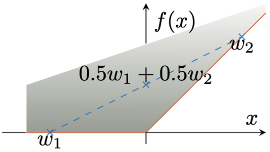
Reformulation 2

Option 1:

$$\min_{x,z} \{D(Ax, y) + \alpha z_K\}$$
$$\text{s.t.} \quad z_1 = \psi_1(x)$$
$$z_k = \psi_k(x, z_1, \ldots, z_{k-1}) \quad k = 2, \ldots, K$$

Constraint: $(x, z_1) \in \text{grph}(\psi_1) = \{t, s \mid s = \psi_1(t)\}$ etc

Idea epigraphs of convex functions are convex: relax constraint



Option 2:

$$\min_{x,z} \{D(Ax, y) + \alpha z_K\}$$
$$\text{s.t.} \quad z_1 \geq \psi_1(x)$$
$$z_k \geq \psi_k(x, z_1, \ldots, z_{k-1}) \quad k = 2, \ldots, K$$

Constraints: $(x, z_1) \in \text{epi}(\psi_1) = \{t, s \mid s \geq \psi_1(t)\}$ etc

Properties of Reformulation

Option 1:

$$\begin{aligned} \mathcal{R}_{\text{eq}}(x) = \min_{z}\, & z_K \\ \text{s.t.}\quad & z_1 = \psi_1(x) \\ & z_k = \psi_k(x, z_1, \ldots, z_{k-1}), k = 2, \ldots, K \end{aligned}$$

Option 2:

$$\begin{aligned} \mathcal{R}_{\text{ineq}}(x) = \min_{z}\, & z_K \\ \text{s.t.}\quad & z_1 \geq \psi_1(x) \\ & z_k \geq \psi_k(x, z_1, \ldots, z_{k-1}), k = 2, \ldots, K \end{aligned}$$

Prop: $\mathcal{R}_{\text{eq}}(x) \geq \mathcal{R}_{\text{ineq}}(x)$

Prop: Under the Assumption, $\mathcal{R}_{\text{eq}}(x) = \mathcal{R}_{\text{ineq}}(x)$

Corollary: Under the Assumption, Option 2 is convex and equivalent to Option 1 (and thus to Option 0).

Numerical Approach

Option 2:

$$\min_{x,z} \{\mathcal{D}(Ax, y) + \alpha z_K\}$$
$$\text{s.t.} \quad z_1 \geq \sigma(V_1 x + b_1)$$
$$z_k \geq \sigma(W_k z_{k-1} + V_k x + b_k) \quad k = 2, \ldots, K$$

Epigraph reformulation:

$$\min_{x,z} \{\mathcal{D}(Ax, y) + \alpha z_K\}$$
$$\text{s.t.} \quad (V_1 x + b_1, z_1) \in \text{epi}(\sigma)$$
$$(W_k z_{k-1} + V_k x + b_k, z_k) \in \text{epi}(\sigma) \quad k = 2, \ldots, K$$

Numerical Approach

Epigraph reformulation:

$$\min_{x,z} \{\mathcal{D}(Ax, y) + \alpha z_K\}$$

$$\begin{aligned} \text{s.t.} \quad & (V_1 x + b_1, z_1) \in \text{epi}(\sigma) \\ & (W_k z_{k-1} + V_k x + b_k, z_k) \in \text{epi}(\sigma) \quad k = 2, \dots, K \end{aligned}$$

Numerical Approach

Epigraph reformulation:

$$\min_{x,z} \{\mathcal{D}(Ax, y) + \alpha z_K\}$$

$$\text{s.t.} \quad (V_1 x + b_1, z_1) \in \text{epi}(\sigma)$$

$$(W_k z_{k-1} + V_k x + b_k, z_k) \in \text{epi}(\sigma) \quad k = 2, \dots, K$$

Optimization template:

$$\min_{w} \{\mathcal{F}(Kw) + \mathcal{H}(w) + \mathcal{G}(w)\}$$

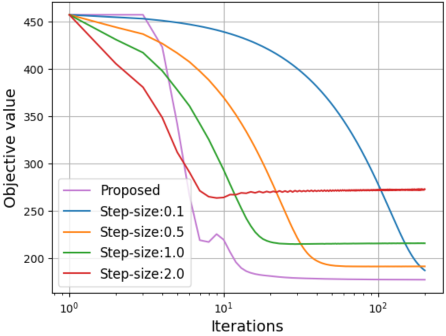
- $w = (x, z)$
- $\mathcal{F}$ related to the indicator function of the epigraph of σ
- K related to the weight matrices
- can be approached via standard primal-dual algorithms, e.g., PDHG Chambolle and Pock 2011, CV Condat '13, Vũ '13 . . . ; implementation details see paper

Example: Salt & Pepper denoising

$$\min_x \{\mathcal{F}(x) = \|x - y\|_1 + \lambda \mathcal{R}_\theta(x)\},$$

Compare with **constant step-size** subgradient method:

$$x^{k+1} = x^k - \eta g^k, \quad g^k \in \partial \mathcal{F}(x^k)$$



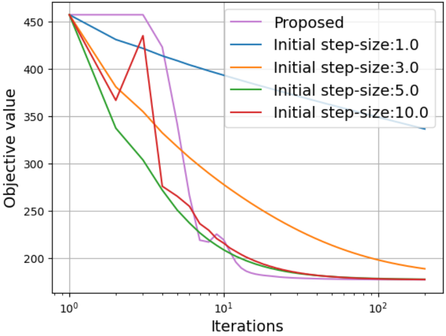
- Fast and stable convergence
- SM-C: large step-size converges to suboptimal solution
- SM-C: small step-size converges but is slow

Example: Salt & Pepper denoising

$$\min_x \{\mathcal{F}(x) = \|x - y\|_1 + \lambda \mathcal{R}_\theta(x)\},$$

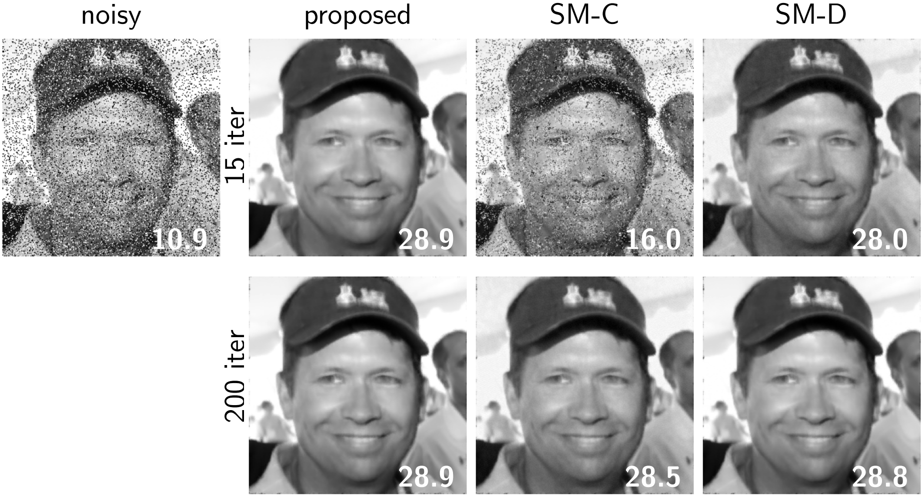
Compare with **diminishing step-size** subgradient method:

$$x^{k+1} = x^k - \frac{\eta}{k} g^k, \quad g^k \in \partial F(x^k)$$



- similar but slightly faster

Example: Salt & Pepper denoising



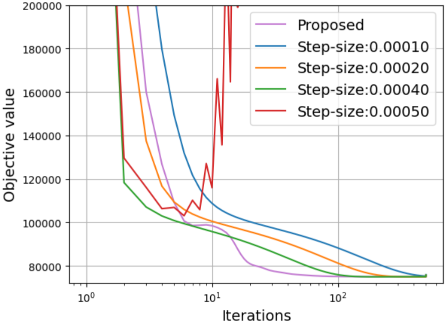
- max PSNR already reached after 15 iterations
- SM-C struggles

Example: CT

$$\min_{x} \left\{ \mathbf{1}^T (Ax - y + r) + y^T \log\left(\frac{y}{Ax + r}\right) + \lambda \mathcal{R}_\theta(x) + \iota_{[0,\infty)}(x) \right\}$$

Compare with **constant step-size** subgradient method:

$$x^{k+1} = x^k - \eta g^k, \quad g^k \in \partial \mathcal{F}(x^k)$$



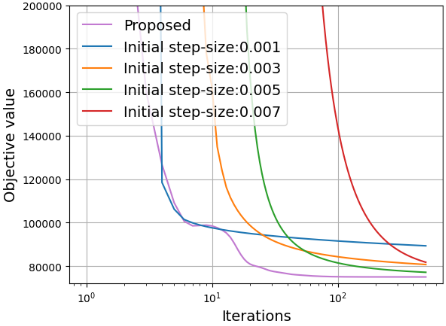
- SM-C with large step-size diverges

Example: CT

$$\min_x \left\{ \mathbf{1}^T (Ax - y + r) + y^T \log\left(\frac{y}{Ax + r}\right) + \lambda \mathcal{R}_\theta(x) + \iota_{[0,\infty)}(x) \right\}$$

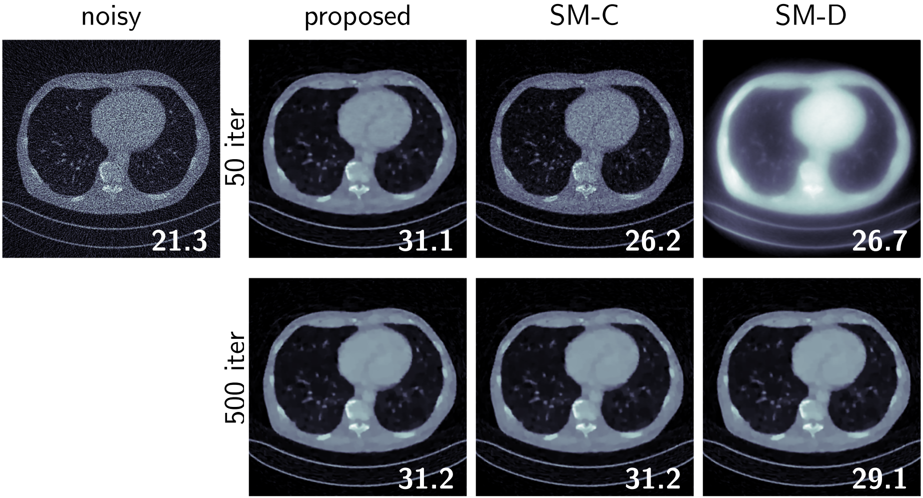
Compare with **diminishing step-size** subgradient method:

$$x^{k+1} = x^k - \frac{\eta}{k} g^k, \quad g^k \in \partial \mathcal{F}(x^k)$$



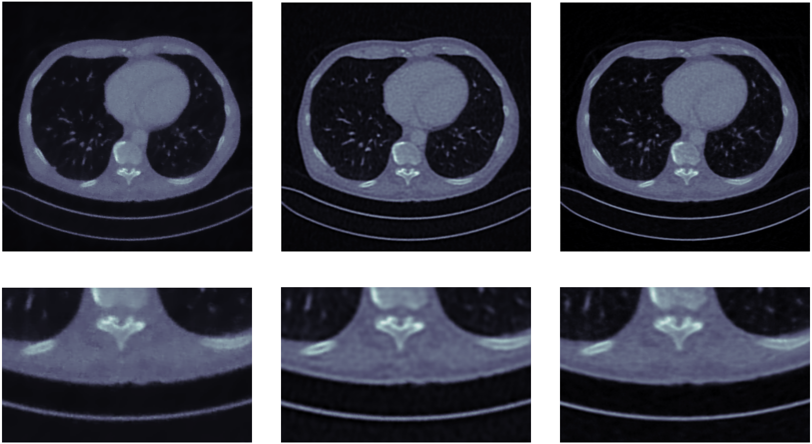
- SM-D does not improve SM-C in this example

Example: CT



- max PSNR almost reached after 50 iterations
- SM-C slower
- SM-D does not reach max PSNR

Bilevel Learning for ICNN



ICNN-AR, PSNR=29.3 ICNN-Bilevel, 31.4 LPD, 34.2

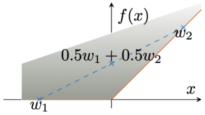
- much better performance with end-to-end learning

Bogensperger et al. '25, Mukherjee et al. '24, Adler and Öktem '18

Summary and Outlook

Summary:

- ICNN-type regularizer for inverse problems
- Epigraph reformulation preserves convexity
- Fast and robust primal-dual framework for inference



Outlook:

- Training of ICNNs Bogensperger et al. '25
- Architectures beyond ICNN?
- Non-convexity?