RED: Regularization by Denoising

The Little Engine that Could

Yaniv Romano

Technion - Israel Institute of Technology

SIAM Annual Meeting 2017

Joint work with:

Michael Elad

Peyman Milanfar

TECHNION Israel Institute of Technology

Google Research

The research leading to these results has been received funding from the European union's Seventh Framework Program (FP/2007-2013) ERC grant Agreement ERC-SPARSE- 320649

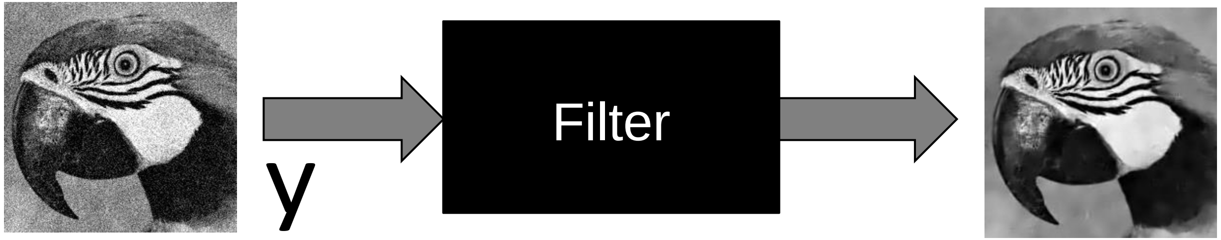
Background and Main Objective

Image Denoising: Definition

This is the "simplest" inverse problem

$$y = x + e$$

Measured — Clean — Noise (AWGN)



An output as close as possible to

x

Image Denoising – Can we do More?

Instead of improving
image denoising algorithms
lets seek ways to leverage
these "*engines*" in order to solve
OTHER (INVERSE) PROBLEMS

Prior-Art 1:

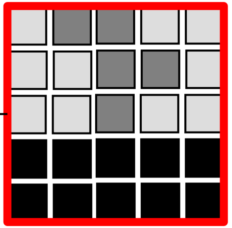
Laplacian Regularization

[Elmoataz, Lezoray, & Bougleux 2008] [Szlam, Maggioni, & Coifman 2008]
[Peyre, Bougleux & Cohen 2011] [Milanfar 2013] [Kheradmand & Milanfar 2014]
[Liu, Zhai, Zhao, Zhai, & Gao 2014] [Haque, Pai, & Govindu 2014] [Romano & Elad 2015]…

Pseudo-Linear Denoising

- Often times we can describe our denoiser as a pseudo-linear Filter

$$\text{Filter}\{x\} = \mathbf{W}(x)x$$



- True for K-SVD, EPLL, NLM, BM3D and other algorithms, where the overall processing is divided into a non-linear stage of decisions, followed by a linear filtering

- We may propose an image-adaptive Laplacian:

$$\text{Laplacian}\{x\} = x - \mathbf{W}(x)x$$

The “residual”

$$= \left(\mathbf{I} - \mathbf{W}(x)\right)x = \mathbf{L}(x)x$$

Laplacians as Regularization

$$\min_{x} \ \ell(x,y) + \frac{\lambda}{2} x^T \mathbf{L} x$$

Laplacian Regularization

The problems with this line of work are that:

1. The regularization term is hard to work with since **L**/**W** is a function of x. This is circumvented by cheating and assuming a fixed **W** per each iteration
2. If so, what is really the underlying energy that is being minimized?
3. When the denoiser cannot admit a pseudo-linear interpretation of **W**(x)x, this term is not possible to use

Prior-Art 2:

The Plug-and-Play-Prior (P^3) Scheme

[Venkatakrishnan, Wohlberg & Bouman, 2013]

The P^3 Scheme

- Use a denoiser to solve general inverse problems
- Main idea: Use ADMM to minimize the MAP energy

$$\hat{x}_{MAP} = \min_{x} \ \ell(x,y) + \frac{\lambda}{2}\, \rho(x)$$

- The ADMM translates this problem (difficult to solve) into 2 simple sub-problems:

1. Solve a linear system of equations, followed by
2. A denoising step

P^3 Shortcomings

- The P^3 scheme is an excellent idea, as one can use ANY denoiser, even if $\rho(\cdot)$ is not known, but…
  - Parameter tuning is **TOUGH** when using a general denoiser
  - This method is tightly tied to **ADMM** without an option for changing this scheme
  - **CONVERGENCE** ? Unclear (steady-state at best)
  - For an arbitrarily denoiser, no underlying & consistent **COST FUNCTION**
- In this work we propose an alternative which is closely related to the above ideas (both Laplacian regularization and P^3) which overcomes the mentioned problems: RED

RED: First Steps

Regularization by Denoising [RED]

$$\rho(x) = \frac{1}{2} x^T (x - \mathbf{W}x)$$

Regularization by Denoising [RED]

We suggest: $\rho(x) = \frac{1}{2} x^T (x - f(x))$

... for an arbitrary denoiser f(x)

$$\rho(x) = 0 \quad \Longrightarrow \quad \begin{array}{l} 1.\ x = 0 \\ 2.\ x = f(x) \\ 3.\ \text{Orthogonality} \end{array}$$

[Romano, Elad & Milanfar, 2016]

Which f(x) to Use ?

Almost any algorithm you want may be used here, from the simplest Median (see later), all the way to the state-of-the-art CNN-like methods

We shall require f(x) to satisfy several properties as follows …

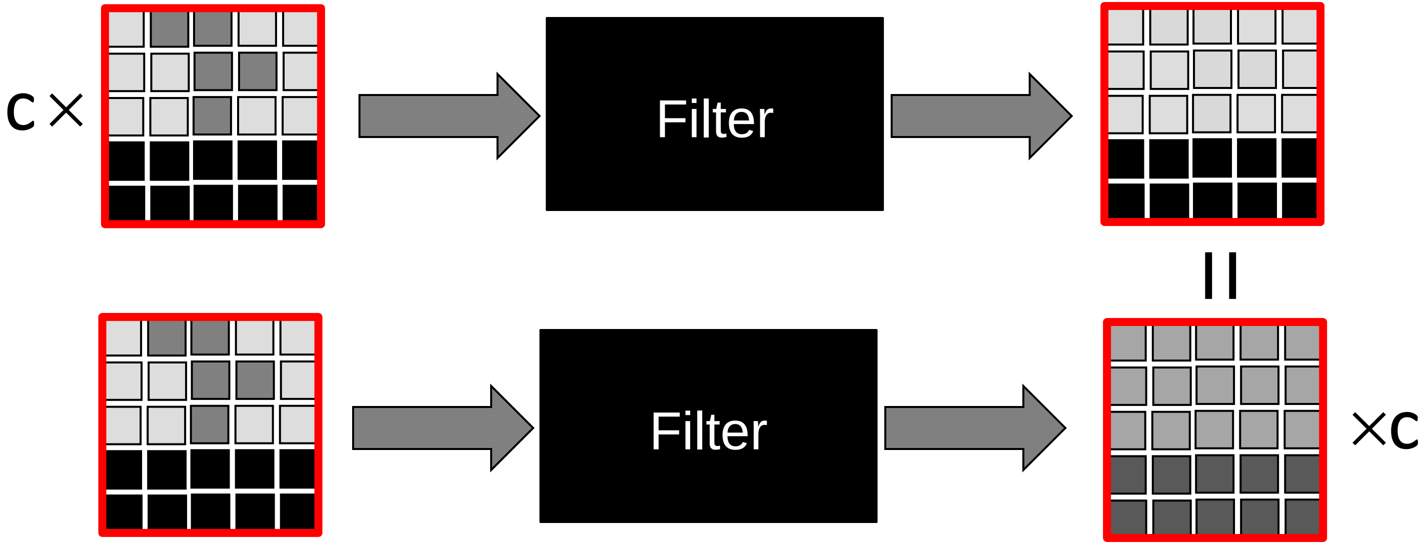
Denoising Filter Property I

$$f(x):[0,1]^n \rightarrow [0,1]^n$$

- **Differentiability:**
  - Some filters obey this requirement (NLM, Bilateral, Kernel Regression, TNRD)
  - Others can be ε-modified to satisfy this (Median, K-SVD, BM3D, EPLL, CNN, …)

Denoising Filter Property II

- Local Homogeneity: for $|c-1| \leq \varepsilon << 1$, we have that $f(cx) = cf(x)$



Denoising Filter Property II

- Local Homogeneity: for $|c-1| \leq \varepsilon << 1$, we have that $f(cx) = cf(x)$

Holds for state-of-the-art algorithms such as K-SVD, NLM, BM3D, EPLL & TNRD…

Implication (1)

- Directional Derivative:

$$\nabla_x f\left(x\right)\cdot d \;= \lim_{\varepsilon\to 0}\frac{f(x+\varepsilon d)-f(x)}{\varepsilon}$$

$d = x$

$$\nabla_x f\left(x\right)\cdot x = \lim_{\varepsilon\to 0}\frac{f(x+\varepsilon x)-f(x)}{\varepsilon}$$

$$= \lim_{\varepsilon\to 0}\frac{(1+\varepsilon)f(x)-f(x)}{\varepsilon} = f(x)$$

↑ Homogeneity

Looks Familiar ?

- We got the property $\nabla_x f(x) \cdot x = f(x)$

  ↑ n×n Matrix

- This is much more general than $f(x) = \mathbf{W}(x)x$ and applies to any denoiser satisfying the above conditions

Implication (2)

- For small $\|h\|_2$

$$f(x+h) \cong f(x) + \nabla_x f(x) \cdot h$$

Directional Derivative →

$$= \nabla_x f(x) \cdot x + \nabla_x f(x) \cdot h$$

$$= \nabla_x f(x) \cdot (x+h)$$

- Implication: Filter stability. Small additive perturbations of the input don't change the filter matrix

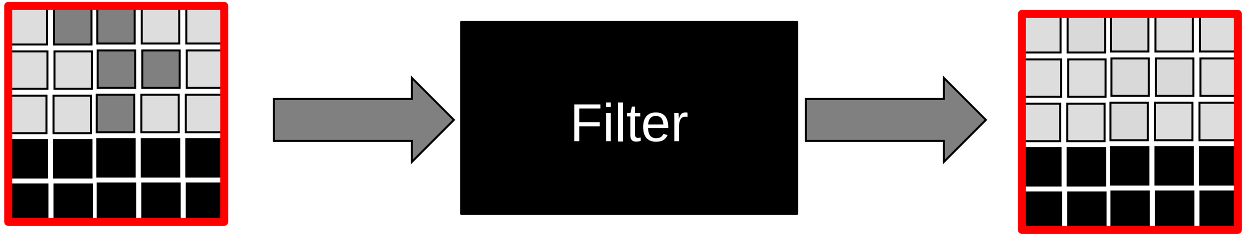
Denoising Filter Property III

- **Passivity** via the spectral radius:

$$r\{\nabla_x f(x)\} = \max |\lambda(\nabla_x f(x))| \leq 1$$

$$\Rightarrow \quad \|x\| \geq \|f(x)\|$$

↑ Passivity



Denoising Filter Property III

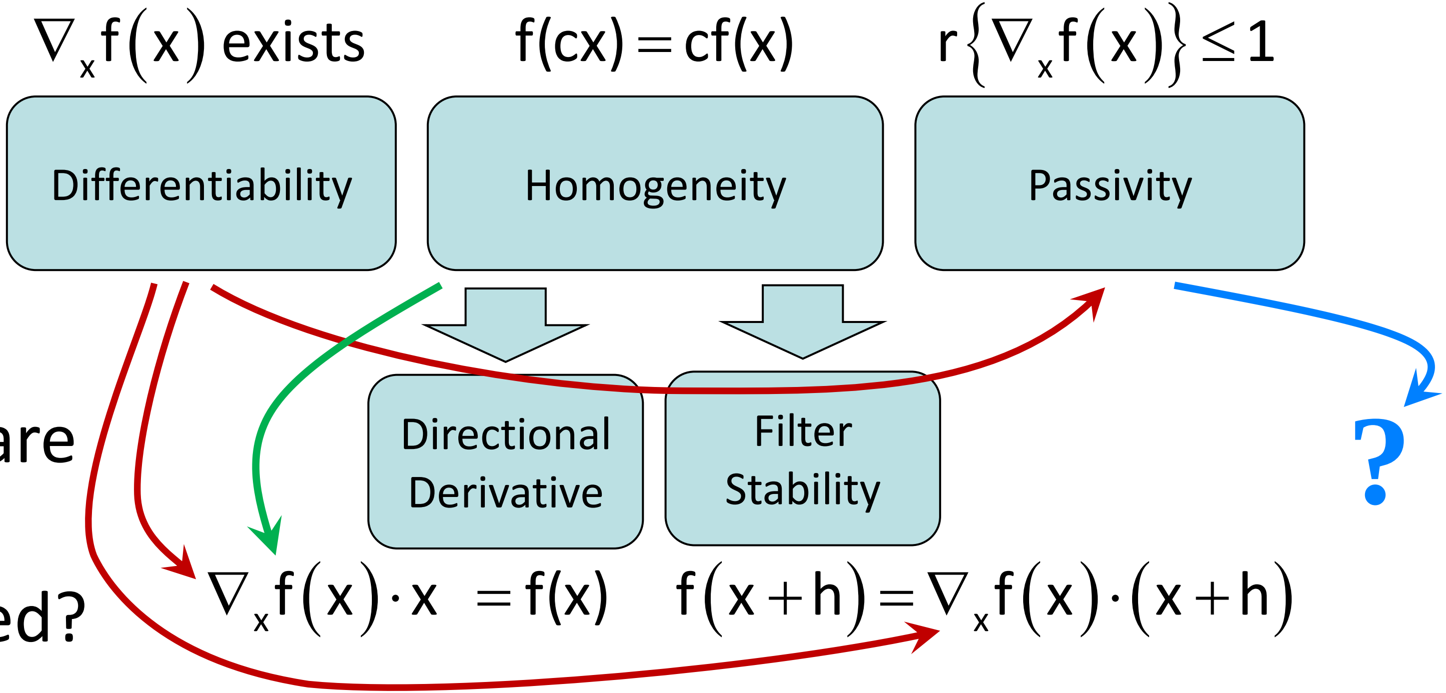
- Passivity via the spectral radius:

$$r\left\{\nabla_x f\left(x\right)\right\} = \max \left|\lambda\left(\nabla_x f\left(x\right)\right)\right| \leq 1$$

Holds for state-of-the-art algorithms such as K-SVD, NLM, BM3D, EPLL & TNRD…

Summary of Properties

- The 3 properties that f(x) should follow:



- Why are these needed?

RED: Advancing

Regularization by Denoising (RED)

$$\rho(x)=\frac{1}{2}x^{T}\left(x-f\left(x\right)\right)^{*}$$

Surprisingly, this expression is differentiable:

$$\nabla\rho(x)=x-\frac{1}{2}\nabla\left\{x^{T}f\left(x\right)\right\}$$

$$=x-\frac{1}{2}\left(f(x)+\nabla f(x)x\right)=x-f(x)\ \text{the residual}$$

$\nabla_x f\left(x\right)\cdot x\ =f(x)$ and Homogeneity

* Why not $\rho(x)=\frac{1}{2}\left\|x-f\left(x\right)\right\|_2^2$?

Regularization by Denoising (RED)

$$\rho(x) = \frac{1}{2} x^T \left(x - f\left(x\right)\right)$$

$$\Downarrow$$

$$\nabla \rho(x) = x - f\left(x\right)$$

$$\Downarrow$$

$$\nabla \left\{ \nabla \rho(x) \right\} = \mathbf{I} - \nabla_x f\left(x\right) \succcurlyeq 0$$

Relying on the differentiability

$$r\left\{ \nabla_x f\left(x\right) \right\} \leq 1$$

Passivity guarantees positive definiteness of the Hessian and hence convexity

RED for Linear Inverse Problems

$$\min_{x} \frac{1}{2}\|\mathbf{H}x - y\|_2^2 + \frac{\lambda}{2} x^T\left(x - f(x)\right)$$

L_2-based Data Fidelity　　　Regularization

This energy-function is convex

⇩

Any reasonable optimization algorithm will get to the global minimum if applied correctly

Numerical Approach

$$\min_{x} \frac{1}{2}\left\|\mathbf{H}x - y\right\|_2^2 + \frac{\lambda}{2}x^T\left(x - f(x)\right)$$

We proposed three ways to minimize this objective

1. Steepest Descent – simple but slow
2. ADMM – reveals the differences between the P^3 and RED
3. Fixed Point – the most efficient method

Will are about to concentrate on the last one

Numerical Approach: Fixed Point

$$\min_{x} \frac{1}{2}\|\mathbf{H}x - y\|_2^2 + \frac{\lambda}{2}x^T\left(x - f(x)\right)$$

$$\mathbf{H}^T\left(\mathbf{H}x - y\right) + \lambda\left(x - f(x)\right) = 0$$

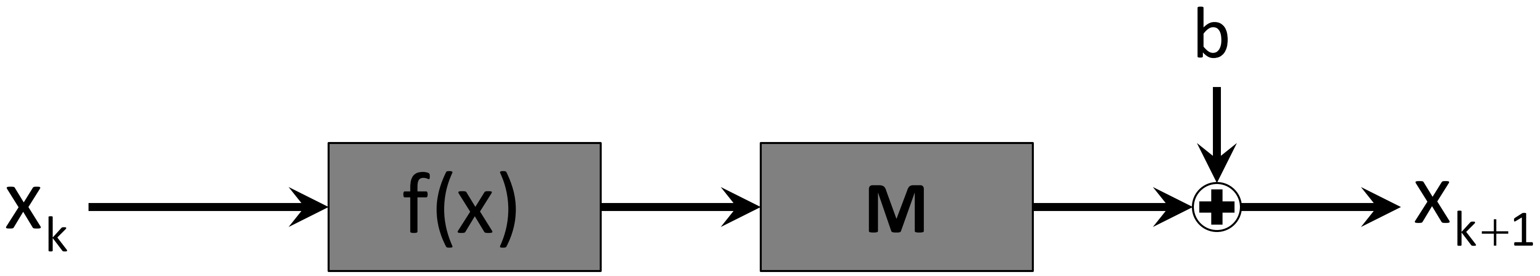
$$\mathbf{H}^T\left(\mathbf{H}x_{k+1} - y\right) + \lambda\left(x_{k+1} - f(x_k)\right) = 0$$

$$x_{k+1} = \left(\mathbf{H}^T\mathbf{H} + \lambda\mathbf{I}\right)^{-1}\left(\mathbf{H}^T y + \lambda f(x_k)\right)$$

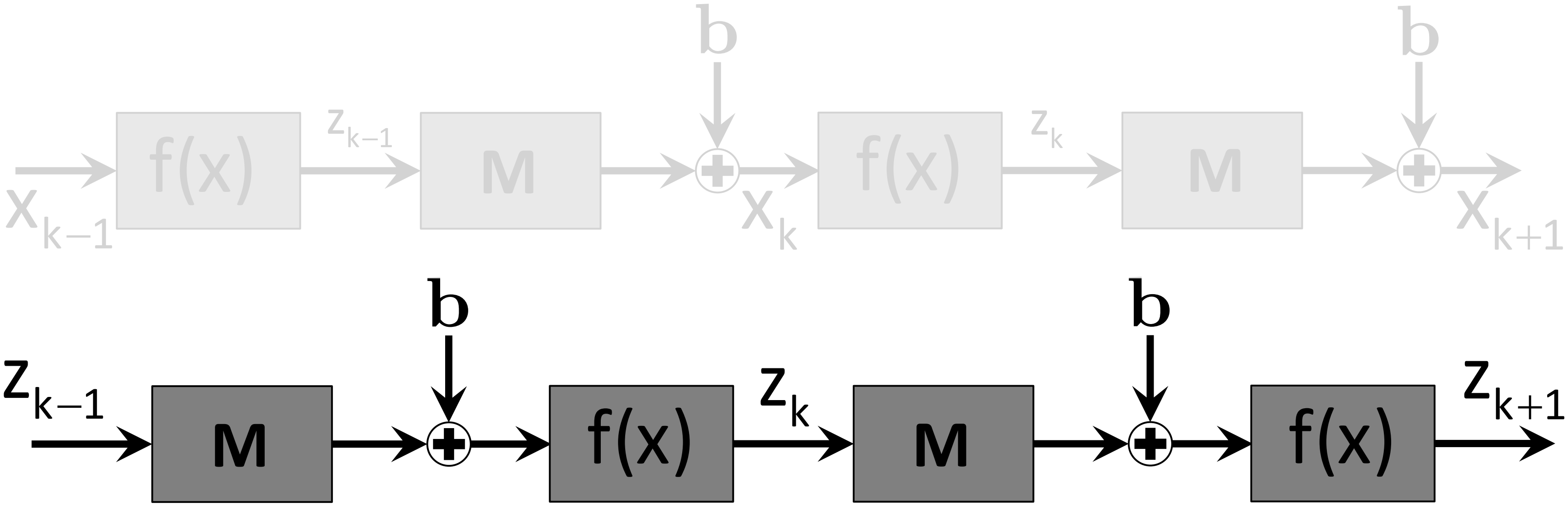
Guaranteed to converge due to the passivity of f(x)

Numerical Approach III: Fixed Point

$$x_{k+1} = \underbrace{\left(\mathbf{H}^T\mathbf{H} + \lambda\mathbf{I}\right)^{-1}\mathbf{H}^T y}_{b} + \underbrace{\lambda\left(\mathbf{H}^T\mathbf{H} + \lambda\mathbf{I}\right)^{-1}}_{\mathbf{M}} f(x_k)$$



A connection to CNN



$$z_{k+1} = f\left(\mathbf{M}z_k + b\right)$$

- While CNN use a trivial and weak non-linearity f(●), we propose a very aggressive and image-aware denoiser
- Our scheme is guaranteed to minimize a clear and relevant objective function

So… Again, Which f(x) to Use ?

- Almost any algorithm you want may be used here, from the simplest Median (see later), all the way to the state-of-the-art CNN-like methods

Comment: Our approach has one hidden parameter – the level of the noise (σ) the denoiser targets. We simply fix this parameter for now. But more work is required to investigate its effect

RED in Practice

Examples: Deblurring

Uniform 9×9 kernel and WAGN with σ^2=2

Ground Truth

Input 20.83dB

RED+Median 25.87dB

NCSR 28.39dB

P^3 +TNRD 28.43dB

RED+TNRD 28.82dB

Examples: 3x Super-Resolution

Bicubic 20.68dB

NCSR 26.79dB

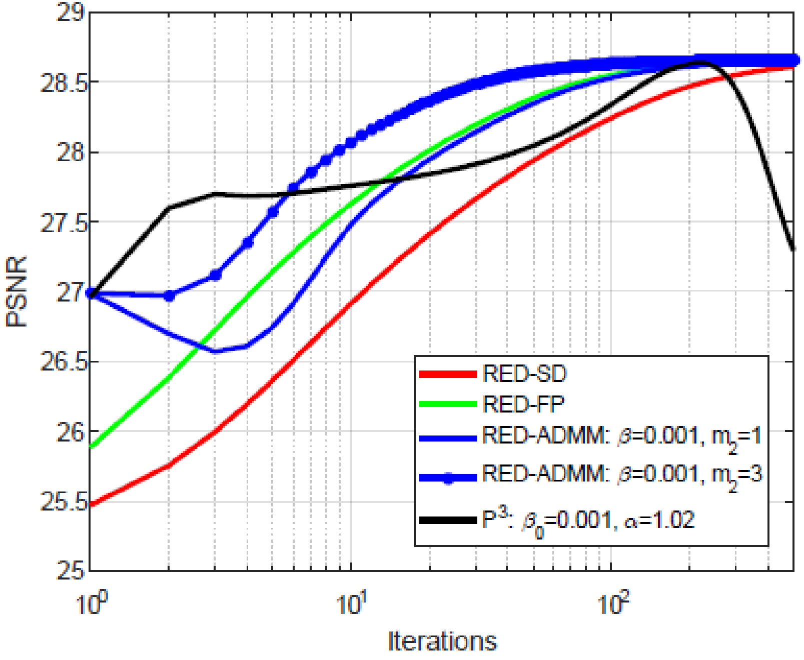
P^3 +TNRD 26.61dB

RED+TNRD 27.39dB

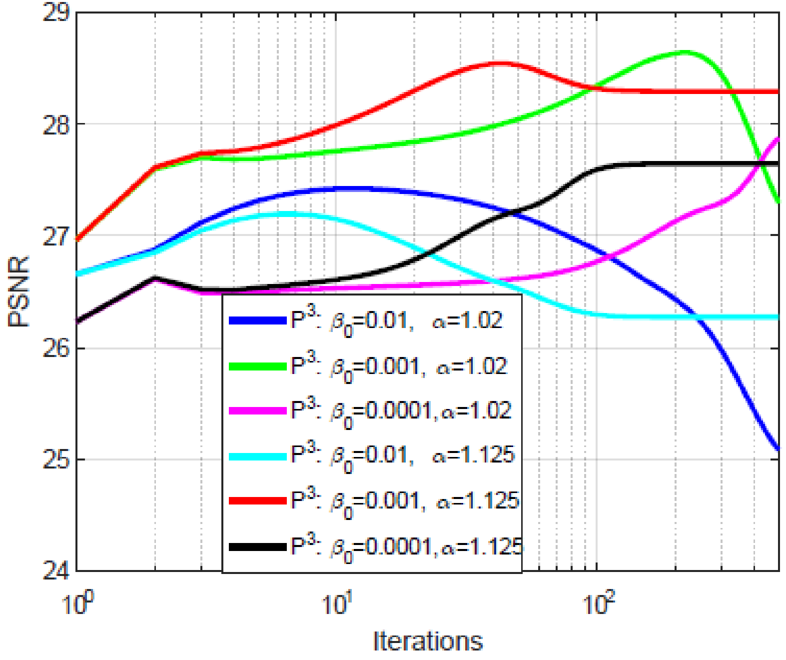
Degradation:

- A Gaussian 7×7 blur with width 1.6
- A 3:1 down-sampling and
- WAGN with σ=5

Sensitivity to Parameters

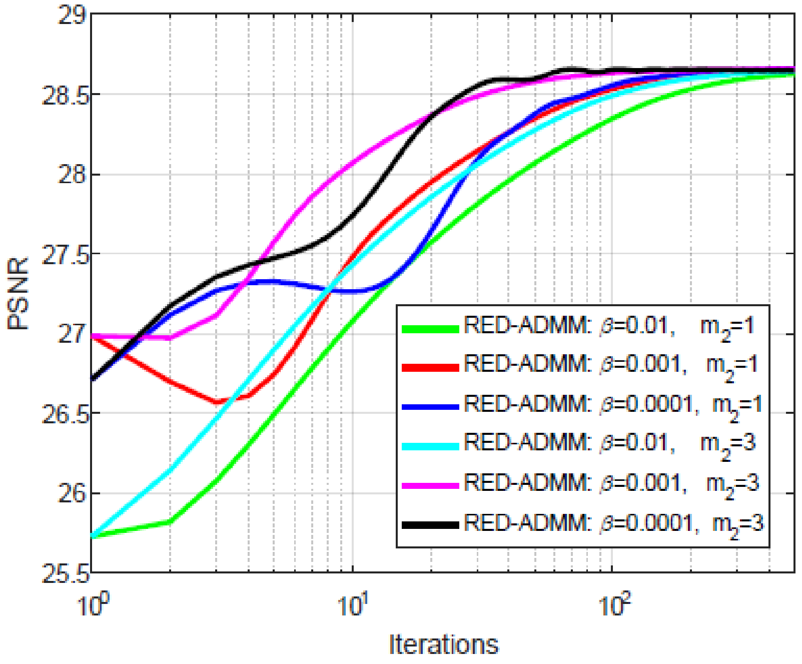


RED versus P^3

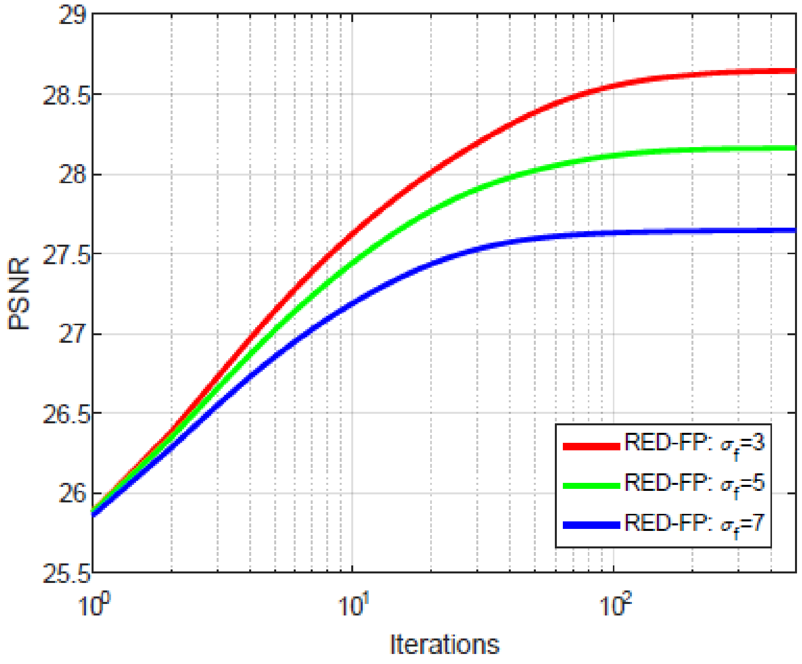


P^3: Sensitivity to parameter changes in the ADMM

Sensitivity to Parameters



RED: Robustness to parameter changes in the ADMM



RED: The effect of the input noise-level to f(·)

Conclusions

What Have We Seen Today ?

- RED – a method to take a denoiser and use it sequentially for solving inverse problems
- Main benefits: Clear objective being minimized, Convexity, flexibility to use almost any denoiser and any optimization scheme
- One could refer to RED as a way to substantiate earlier methods (Laplacian-Regularization and the P^3) and fix them
- Challenges: Trainable version? Compression?

Relevant Reading

1. "Regularization by Denoising", Y. Romano, M. Elad, and P. Milanfar, To appear, SIAM J. on Imaging Science.
2. "A Tour of Modern Image Filtering", P. Milanfar, IEEE Signal Processing Magazine, no. 30, pp. 106–128, Jan. 2013
3. "A General Framework for Regularized, Similarity-based Image Restoration", A. Kheradmand, and P. Milanfar, IEEE Trans on Image Processing, vol. 23, no. 12, Dec. 2014
4. "Boosting of Image Denoising Algorithms", Y. Romano, M. Elad, SIAM J. on Image Science, vol. 8, no. 2, 2015
5. "How to SAIF-ly Improve Denoising Performance", H. Talebi , X. Zhu, P. Milanfar, IEEE Trans. On Image Proc., vol. 22, no. 4, 2013
6. "Plug-and-Play Priors for Model-Based Reconstruction", S.V. Venkatakakrishnan, C.A. Bouman, B. Wohlberg, GlobalSIP, 2013
7. "BM3D-AMP: A New Image Recovery Algorithm Based on BM3D Denoising", C.A. Metzler, A. Maleki, and R.G. Baraniuk, ICIP 2015
8. "Is Denoising Dead?", P. Chatterjee, P. Milanfar, IEEE Trans. On Image Proc., vol. 19, no. 4, 2010
10. "Symmetrizing Smoothing Filters", P. Milanfar, SIAM Journal on Imaging Sciences, Vol. 6, No. 1, pp. 263–284
11. "What Regularized Auto-Encoders Learn from The Data-Generating Distribution", G. Alain, and Y. Bengio, JMLR, vol.15, 2014,
12. "Representation Learning: A Review and New Perspectives", Y. Bengio, A. Courville, and P. Vincent, IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 35, no. 8, Aug. 2013


"That's all Folks!"
Thank You