Solving RED with Weighted Proximal Methods

Tao Hong
Joint work with Irad Yavneh and Michael Zibulevsky

Computer Science Department
Technion - Israel Institute of Technology
Outline

Background
   RED and Its Properties
   Existing Solvers in RED

Weighted Proximal Methods (WPMs)
   Proximal Methods and Its Acceleration
   How? and Why? – WPMs

Numerical Results
   Image Deblurring
   Image Super-Resolution (SR)
   Additional Results
Background

RED and Its Properties
Existing Solvers in RED

Weighted Proximal Methods (WPMs)
Proximal Methods and Its Acceleration
How? and Why? – WPMs

Numerical Results
Image Deblurring
Image Super-Resolution (SR)
Additional Results
Inverse Problems – Optimization Problem

Image Denoising - “Simplest” Inverse:

\[
\begin{align*}
\hat{y} & = \hat{x} + \hat{n} \\
\text{Measured} & & \text{Clean} & & \text{Noise (AWGN)}
\end{align*}
\]
Inverse Problems – Optimization Problem

Image Denoising - “Simplest” Inverse:

\[ y = x + n \]

Measured \hspace{1cm} Clean \hspace{1cm} Noise (AWGN)

How?
Inverse Problems – Optimization Problem

Image Denoising - “Simplest” Inverse:

\[
\begin{align*}
\mathbf{y} & = \mathbf{x} + \mathbf{n} \\
\text{Measured} & \quad \text{Clean} \quad \text{Noise (AWGN)}
\end{align*}
\]

How?

Maximum a Posteriori Probability (MAP) – Prior

Optimization Problem

\[
\hat{x} = \arg \min_{x} \ell(x, y) + \lambda \rho(x)
\]

Data Fidelity \quad Prior

\[
\ell(x, y) : \text{linear or nonlinear} \rightarrow \frac{1}{2\sigma^2} \|x - y\|^2_2 \quad \text{or} \quad \frac{1}{2\sigma^2} \|Hx - y\|^2_2
\]

\[
\rho(\cdot) : \text{TV, sparsity, low-rank, CNN etc.}
\]
Inverse Problems – Optimization Problem

Image Denoising - “Simplest” Inverse:

\[ y = x + n \]

Measured Clean Noise (AWGN)

How?

Maximum a Posteriori Probability (MAP) – Prior

Optimization Problem

\[ \hat{x} = \arg \min_x \ell(x, y) + \lambda \rho(x) \]

Data Fidelity Prior

\[ \ell(x, y) : \text{linear or nonlinear} \rightarrow \frac{1}{2\sigma^2} \| x - y \|_2^2 \text{ or } \frac{1}{2\sigma^2} \| Hx - y \|_2^2 \]

\[ \rho(\cdot) : \text{TV, sparsity, low-rank, CNN etc.} \]

Finding effective \( \rho(\cdot) \)?
What is RED? – REgularized by Denoising

Can we utilize these denoising algorithms as the prior? and How?
What is RED? – REgularized by Denoising

Can we utilize these denoising algorithms as the prior? and How?

semi-positive
What is RED? – REgularized by Denoising

Can we utilize these denoising algorithms as the prior? and How?

semi-positive

Romano et al. [REM17] → RED:

\[ \rho(x) = \frac{1}{2} x^T (x - f(x)) \]

\( f(x) \): abstract image denoising algorithms
What is RED? – REgularized by Denoising

Can we utilize these denoising algorithms as the prior? and How?

semi-positive

Romano et al. [REM17] → RED:

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

$$f(x) : \text{abstract image denoising algorithms}$$

$$\hat{x} = \arg \min_x \phi(x) \triangleq \ell(x, y) + \frac{\lambda}{2} x^T (x - f(x))$$
What is RED? – REgularized by Denoising

Can we utilize these denoising algorithms as the prior? and How?

semi-positive

Romano et al. [REM17] → RED:

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

$$f(\mathbf{x}) : \text{abstract image denoising algorithms}$$

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \phi(\mathbf{x}) \triangleq \ell(\mathbf{x}, \mathbf{y}) + \frac{\lambda}{2} \mathbf{x}^T (\mathbf{x} - f(\mathbf{x}))$$

How to minimize $$\phi(\mathbf{x})$$? It is weird → $$f(\mathbf{x})$$. 
The Properties of RED

Assumptions:

- Differentiability: $f(x) : [0, 1]^n \rightarrow [0, 1]^n$
- Local Homogeneity: $f(cx) = cf(x)$, if $|c - 1| \leq \varepsilon \ll 1$
- Passivity: $\|f(x)\| \leq \|x\|$
The Properties of RED

Assumptions:

▶ Differentiability: \( f(x) : [0, 1]^n \to [0, 1]^n \)

▶ Local Homogeneity: \( f(cx) = cf(x) \), if \( |c - 1| \leq \varepsilon \ll 1 \)

▶ Passivity: \( \|f(x)\| \leq \|x\| \)

\[
\nabla_x f(x) \cdot x = \lim_{\varepsilon \to 0} \frac{f(x+\varepsilon x)-f(x)}{\varepsilon} = \lim_{\varepsilon \to 0} \frac{(1+\varepsilon)f(x)-f(x)}{\varepsilon} = f(x)
\]

\[
\frac{\partial}{\partial x} \left( \frac{1}{2} x^T (x - f(x)) \right) = x - \frac{1}{2} f(x) - \frac{1}{2} \nabla_x f(x) \cdot x = x - f(x)
\]

\[
\frac{\partial^2}{\partial x \partial x^T} \left( \frac{1}{2} x^T (x - f(x)) \right) \geq 0
\]
The Properties of RED

Assumptions:

- **Differentiability:** \( f(x) : [0, 1]^n \rightarrow [0, 1]^n \)
- **Local Homogeneity:** \( f(cx) = cf(x) \), if \(|c - 1| \leq \varepsilon \ll 1\)
- **Passivity:** \( \|f(x)\| \leq \|x\| \)

\[
\nabla_x f(x) \cdot x = \lim_{\varepsilon \to 0} \frac{f(x+\varepsilon x) - f(x)}{\varepsilon} = \lim_{\varepsilon \to 0} \frac{(1+\varepsilon)f(x) - f(x)}{\varepsilon} = f(x)
\]

\[
\frac{\partial}{\partial x} \left( \frac{1}{2} x^T (x - f(x)) \right) = x - \frac{1}{2} f(x) - \frac{1}{2} \nabla_x f(x) \cdot x = \boxed{x - f(x)}
\]

\[
\frac{\partial^2}{\partial x \partial x^T} \left( \frac{1}{2} x^T (x - f(x)) \right) \geq 0
\]

Conclusions:

- \( \rho(x) \) is convex. If \( \ell(x, y) \) is convex, \( \phi(x) \) is **convex**.
- Evaluate one time gradient or \( \phi(x) \), call one time \( f(x) \).
How Many Algorithms Satisfy these Assumptions?

[REM17]: We have many, some of them are state-of-the-art.

NLM, Bilateral, kernal regression, TNRD etc.

Others $\varepsilon$-modified: Median, K-svd, BM3D, EPLL, CNN etc.
Background
RED and Its Properties
Existing Solvers in RED

Weighted Proximal Methods (WPMs)
Proximal Methods and Its Acceleration
How? and Why? – WPMs

Numerical Results
Image Deblurring
Image Super-Resolution (SR)
Additional Results
Solvers in RED – $\ell(x, y) + \frac{\lambda}{2} x^T (x - f(x))$

- gradient based methods: gradient descent/Nesterov Acceleration, conjugate gradient, BFGS, LBFGS etc. – line search? [NW06]

...
Solvers in RED – $\ell(x, y) + \frac{\lambda}{2} x^T (x - f(x))$

- gradient based methods: gradient descent/Nesterov Acceleration, conjugate gradient, BFGS, LBFGS etc. – line search? [NW06]
- ADMM [BPC\textsuperscript{+} 11]
Solvers in RED – $\ell(x, y) + \frac{\lambda}{2} x^T (x - f(x))$

- gradient based methods: gradient descent/Nesterov Acceleration, conjugate gradient, BFGS, LBFGS etc. – line search? [NW06]
- ADMM [BPC^+11]
- fixed-point (FP) [REM17]

\[
\frac{1}{\sigma^2} H^T (Hx_{k+1} - y) + \lambda (x_{k+1} - f(x_k)) = 0 \quad \text{Fourier or CG}
\]
Solvers in RED – $\ell(x, y) + \frac{\lambda}{2} x^T (x - f(x))$

- gradient based methods: gradient descent/Nesterov Acceleration, conjugate gradient, BFGS, LBFGS etc. – line search? [NW06]
- ADMM [BPC+11]
- fixed-point (FP) [REM17]

$$\frac{1}{\sigma^2} H^T (Hx_{k+1} - y) + \lambda (x_{k+1} - f(x_k)) = 0$$  Fourier or CG

- Vector Extrapolation (VE) [HRE19]
Solvers in RED – $\ell(x, y) + \frac{\lambda}{2} x^T (x - f(x))$

- gradient based methods: gradient descent/Nesterov Acceleration, conjugate gradient, BFGS, LBFGS etc. – line search? [NW06]
- ADMM [BPC+11]
- fixed-point (FP) [REM17]

\[
\frac{1}{\sigma^2} H^T(Hx_{k+1} - y) + \lambda (x_{k+1} - f(x_k)) = 0 \quad \text{Fourier or CG}
\]

- Vector Extrapolation (VE) [HRE19]
- Accelerated Proximal Gradient (APG) [RS19]
- ...

In practice: $\text{APG} \geq \text{VE} > \text{FP} > \text{ADMM} > \text{gradient based}$
Solvers in RED – $\ell(x, y) + \frac{\lambda}{2} x^T (x - f(x))$

- gradient based methods: gradient descent/Nesterov Acceleration, conjugate gradient, BFGS, LBFGS etc. – line search? [NW06]
- ADMM [BPC+11]
- fixed-point (FP) [REM17]

$$\frac{1}{\sigma^2} H^T (Hx_{k+1} - y) + \lambda (x_{k+1} - f(x_k)) = 0$$ Fourier or CG

- Vector Extrapolation (VE) [HRE19]
- Accelerated Proximal Gradient (APG) [RS19]
- ...

In practice: APG $\geq$ VE $>$ FP $>$ ADMM $>$ gradient based

But, but, but
Solvers in RED – $\ell(x, y) + \frac{\lambda}{2} x^T (x - f(x))$

- gradient based methods: gradient descent/Nesterov Acceleration, conjugate gradient, BFGS, LBFGS etc. – line search? [NW06]
- ADMM [BPC\(^+\)11]
- fixed-point (FP) [REM17]

\[
\frac{1}{\sigma^2} H^T (Hx_{k+1} - y) + \lambda (x_{k+1} - f(x_k)) = 0 \quad \text{Fourier or CG}
\]

- Vector Extrapolation (VE) [HRE19]
- Accelerated Proximal Gradient (APG) [RS19]
- ...

In practice: APG $\succeq$ VE $>$ FP $>$ ADMM $>$ gradient based

But, but, but

Weighted Proximal Methods can do better. :-(
Background
RED and Its Properties
Existing Solvers in RED

Weighted Proximal Methods (WPMs)
Proximal Methods and Its Acceleration
How? and Why? – WPMs

Numerical Results
Image Deblurring
Image Super-Resolution (SR)
Additional Results
Development of the Proximal Gradient Methods

Composite problem:

\[
\min_x \phi(x) \triangleq g(x) + h(x)
\]

\(g(x)\): convex, differentiability
\(h(x)\): convex, can be nonsmooth
The solution is nonempty.

\(^1\) Euclidean distance here. Bregman distance in general, [Bec17].
Development of the Proximal Gradient Methods

Composite problem:

$$\min_x \phi(x) \triangleq g(x) + h(x)$$

$g(x)$: convex, differentiability
$h(x)$: convex, can be nonsmooth
The solution is nonempty.

$x_k$: solution at $k$th iteration
Linearizing $g(x)$ at $x_k$ \footnote{Euclidean distance here. Bregman distance in general, [Bec17].}:

$$g(x) + h(x) \leq \hat{\phi}(x, x_k) \triangleq g(x_k) + \langle \nabla_x g(x_k), x - x_k \rangle + \frac{L}{2} \| x - x_k \|_2^2 + h(x)$$

$$\nabla^2_x g(x) \preceq L$$
Development of the Proximal Gradient Methods

Composite problem:

\[
\min_x \phi(x) \triangleq g(x) + h(x)
\]

\(g(x)\): convex, differentiability
\(h(x)\): convex, can be nonsmooth
The solution is nonempty.

\(x_k\): solution at \(k\)th iteration

Linearizing \(g(x)\) at \(x_k\):

\[
g(x) + h(x) \leq \hat{\phi}(x, x_k) \triangleq g(x_k) + \langle \nabla_x g(x_k), x - x_k \rangle + \frac{L}{2} \| x - x_k \|^2 + h(x)
\]

\[
\nabla^2_x g(x) \preceq L
\]

Minimize \(\hat{\phi}(x, x_k)\) instead of minimizing \(\phi(x)\) at \((k + 1)\)th iteration:

\[
\text{Prox}_{\frac{1}{L}} h(v_k) = \arg \min_x \frac{1}{2} \| x - v_k \|^2 + \frac{1}{L} h(x) : \text{Closed-Form}
\]

\[
v_k = x_k - \frac{1}{L} \nabla_x g(x_k)
\]

\(^1\) Euclidean distance here. Bregman distance in general, [Bec17].
Acceleration – Nesterov/FISTA

Set $y_1 = x_0$ and $t_1 = 1$ and repeat the following at step $k \geq 1$

- $x_k = \text{Prox}_{\frac{1}{L}h}(y_k - \frac{1}{L} \nabla x g(y_k))$

- $t_{k+1} = \frac{1 + \sqrt{1 + 4t_k^2}}{2}$

- $y_{k+1} = x_k + \frac{t_k - 1}{t_{k+1}} (x_k - x_{k-1})$
Acceleration – Nesterov/FISTA

Set \( y_1 = x_0 \) and \( t_1 = 1 \) and repeat the following at step \( k \geq 1 \)

\[
\begin{align*}
\triangleright \quad x_k &= \text{Prox}_{\frac{1}{L} h} \left( y_k - \frac{1}{L} \nabla x g(y_k) \right) \\
\triangleright \quad t_{k+1} &= \frac{1+\sqrt{1+4t_k^2}}{2} \\
\triangleright \quad y_{k+1} &= x_k + \frac{t_k - 1}{t_{k+1}} \left( x_k - x_{k-1} \right)
\end{align*}
\]

Convergence Speed:

Proximal: \( O\left(\frac{1}{k}\right) \)

Acceleration: \( O\left(\frac{1}{k^2}\right) \)
Acceleration – Nesterov/FISTA

Set $y_1 = x_0$ and $t_1 = 1$ and repeat the following at step $k \geq 1$

1. $x_k = \text{Prox}_{\frac{1}{L}h}(y_k - \frac{1}{L} \nabla x g(y_k))$

2. $t_{k+1} = \frac{1 + \sqrt{1 + 4 t_k^2}}{2}$

3. $y_{k+1} = x_k + \frac{t_k - 1}{t_{k+1}} (x_k - x_{k-1})$

Convergence Speed:

Proximal: $O\left(\frac{1}{k}\right)$

Acceleration: $O\left(\frac{1}{k^2}\right)$

Can we do better?

No closed-form —— $\text{Prox}_{\frac{1}{L}h}(\cdot)$ —— RED
Background
   RED and Its Properties
   Existing Solvers in RED

Weighted Proximal Methods (WPMs)
   Proximal Methods and Its Acceleration
   How? and Why? – WPMs

Numerical Results
   Image Deblurring
   Image Super-Resolution (SR)
   Additional Results
Weighted Proximal Methods

Linearizing $g(x)$ with higher accuracy:

$$g(x) + h(x) \leq \hat{\phi}(x, x_k)$$

$$\hat{\phi}(x, x_k) \triangleq g(x_k) + \langle \nabla_x g(x_k), x - x_k \rangle + \frac{1}{2a_k} (x - x_k)^T B_k (x - x_k) + h(x)$$

$a_k$ stepsize or use 1 and $B_k \succ 0$. Define

$$\text{Prox}^{WPM}_{a_k h}(v_k) = \arg \min_x \frac{1}{2} \| x - v_k \|_{B_k}^2 + a_k h(x) : \text{No Closed-Form}$$

$$v_k = x_k - a_k B_k^{-1} \nabla_x g(x_k)$$
Weighted Proximal Methods

Linearizing $g(x)$ with higher accuracy:

$$g(x) + h(x) \leq \hat{\phi}(x, x_k)$$

$$\hat{\phi}(x, x_k) \triangleq g(x_k) + \langle \nabla_x g(x_k), x - x_k \rangle + \frac{1}{2a_k} (x - x_k)^T B_k (x - x_k) + h(x)$$

$a_k$ stepsize or use 1 and $B_k \succ 0$. Define

$$\text{Prox}_{a_k h}^{WPM}(v_k) = \arg \min_x \frac{1}{2} \|x - v_k\|_{B_k}^2 + a_k h(x) : \text{No Closed-Form}$$

$$v_k = x_k - a_k B_k^{-1} \nabla_x g(x_k)$$

In RED:

Remind the denoising $f(x)$ in RED: high complexity
Weighted Proximal Methods

Linearizing $g(x)$ with higher accuracy:

$$g(x) + h(x) \leq \hat{\phi}(x, x_k)$$

$$\hat{\phi}(x, x_k) \triangleq g(x_k) + \langle \nabla_x g(x_k), x - x_k \rangle + \frac{1}{2a_k} (x - x_k)^T B_k (x - x_k) + h(x)$$

$a_k$ stepsize or use 1 and $B_k \succ 0$. Define

$$\text{Prox}_{a_k h}^{WPM}(v_k) = \arg \min_x \frac{1}{2} \| x - v_k \|_{B_k}^2 + a_k h(x) : \text{No Closed-Form}$$

$$v_k = x_k - a_k B_k^{-1} \nabla_x g(x_k)$$

In RED:

Remind the denoising $f(x)$ in RED: high complexity

To reduce the calling of $f(x)$, we set

$$g(x) = \lambda \rho(x)$$

and

$$h(x) = \ell(x, y)$$
The Choice of $B_k - \underbrace{\ell(x, y)}_{h(x)} + \underbrace{\frac{\lambda}{2} x^T (x - f(x))}_{g(x)}$
The Choice of $B_k - \ell(x, y) + \frac{\lambda}{2} x^T (x - f(x))$

Proximal method:

$$\text{Prox}_{\frac{1}{L} h}(v_k) = \arg \min_x \frac{1}{2} \|x - v_k\|^2 + \frac{1}{L} h(x) : \text{Closed-Form}$$

$$v_k = x_k - \frac{1}{L} \nabla_x g(x_k)$$

WPMs:

$$\text{Prox}_{a_k h}^{WPM}(v_k) = \arg \min_x \frac{1}{2} \|x - v_k\|^2 + a_k h(x) : \text{No Closed-Form}$$

$$v_k = x_k - a_k B_k^{-1} \nabla_x g(x_k)$$

$\triangleright B_k = \lambda I$ and $a_k = 1$: recover proximal method
The Choice of $B_k - \ell(x, y) + \frac{\lambda}{2} x^T (x - f(x))$

Proximal method:

$$\text{Prox}_{\frac{1}{L}}(v_k) = \arg \min_x \frac{1}{2} \|x - v_k\|^2 + \frac{1}{L} h(x) : \text{Closed-Form}$$

$$v_k = x_k - \frac{1}{L} \nabla_x g(x_k)$$

WPMs:

$$\text{Prox}_{a_k h}^{WPM}(v_k) = \arg \min_x \frac{1}{2} \|x - v_k\|^2_{B_k} + a_k h(x) : \text{No Closed-Form}$$

$$v_k = x_k - a_k B_k^{-1} \nabla_x g(x_k)$$

- $B_k = \lambda I$ and $a_k = 1$: recover proximal method
- $B_k$: the Hessian of $g(x) \rightarrow$ Quasi-Newton [NW06]
- ...
Algorithm 1 SR1

Initialization: $k = 1$, $\gamma = 1.25$, $\delta = 10^{-8}$, $x_k, x_{k-1}, \nabla g(x_k), \nabla g(x_{k-1})$.

1: if $k = 1$ then
2:  \( B_k \leftarrow \alpha I \)
3: else
4:  Set \( s_k \leftarrow x_k - x_{k-1} \) and \( m_k \leftarrow \nabla g(x_k) - \nabla g(x_{k-1}) \)
5:  Calculate $\tau \leftarrow \gamma \frac{\|m_k\|_2}{\langle s_k, m_k \rangle}$
6:  if $\tau < 0$ then
7:  \( B_k \leftarrow \alpha I \)
8:  else
9: \( H_0 \leftarrow \tau I \)
10: \( \text{if } |\langle m_k - H_0 s_k, s_k \rangle| \leq \delta \|s_k\|_2 \|m_k - H_0 s_k\|_2 \text{ then} \)
11: \( u_k \leftarrow 0 \)
12: \( \text{else} \)
13: \( u_k \leftarrow \frac{m_k - H_0 s_k}{\sqrt{\langle m_k - H_0 s_k, s_k \rangle}} \)
14: \( \text{end if} \)
15: \( B_k \leftarrow H_0 + u_k u_k^T \)
16: \( \text{end if} \)
17: end if
18: Return: $B_k$
Background
  RED and Its Properties
  Existing Solvers in RED

Weighted Proximal Methods (WPMs)
  Proximal Methods and Its Acceleration
  How? and Why? – WPMs

Numerical Results
  Image Deblurring
  Image Super-Resolution (SR)
  Additional Results
Image Deblurring - Uniform

![Graph of Starfish-UniformBlur showing PSNR vs Denoiser Evaluations and Seconds for different denoisers: FP, FP-MPE, APG, WPM. The graphs show the relationship between the number of denoiser evaluations and the time taken to achieve a certain PSNR value.]
Image Deblurring - Gaussian

Starfish-GaussianBlur

PSNR

Starfish-GaussianBlur

PSNR
Background
RED and Its Properties
Existing Solvers in RED

Weighted Proximal Methods (WPMs)
Proximal Methods and Its Acceleration
How? and Why? – WPMs

Numerical Results
Image Deblurring
Image Super-Resolution (SR)
Additional Results
Image Super-Resolution (SR)

Plants-Downscale

PSNR vs. Denoiser Evaluations

PSNR vs. Seconds
Background
  RED and Its Properties
  Existing Solvers in RED

Weighted Proximal Methods (WPMs)
  Proximal Methods and Its Acceleration
  How? and Why? – WPMs

Numerical Results
  Image Deblurring
  Image Super-Resolution (SR)

Additional Results
More Results [HYZ19]

FP – 200 denoiser evaluations

Denoiser evaluations, other methods – comparable PSNR

1st and 2nd row: deblurring with uniform and Gaussian blurs.

3rd row: SR

<table>
<thead>
<tr>
<th>Image</th>
<th>FP-MPE</th>
<th>APG</th>
<th>WPM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Butterfly</td>
<td>54</td>
<td>34</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>54</td>
<td>26</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>51</td>
<td>26</td>
</tr>
<tr>
<td>Boats</td>
<td>24</td>
<td>20</td>
<td>21</td>
</tr>
<tr>
<td></td>
<td>60</td>
<td>34</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td>36</td>
<td>20</td>
<td>12</td>
</tr>
<tr>
<td>House</td>
<td>24</td>
<td>18</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>62</td>
<td>26</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>18</td>
<td>15</td>
<td>10</td>
</tr>
<tr>
<td>Parrot</td>
<td>39</td>
<td>30</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>52</td>
<td>40</td>
<td>36</td>
</tr>
<tr>
<td></td>
<td>49</td>
<td>31</td>
<td>28</td>
</tr>
<tr>
<td>Lena</td>
<td>48</td>
<td>34</td>
<td>29</td>
</tr>
<tr>
<td></td>
<td>47</td>
<td>16</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>37</td>
<td>26</td>
<td>18</td>
</tr>
<tr>
<td>Barbara</td>
<td>14</td>
<td>12</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>48</td>
<td>23</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>17</td>
<td>15</td>
<td>11</td>
</tr>
<tr>
<td>Peppers</td>
<td>42</td>
<td>29</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td>41</td>
<td>40</td>
<td>34</td>
</tr>
<tr>
<td></td>
<td>38</td>
<td>30</td>
<td>28</td>
</tr>
<tr>
<td>Leaves</td>
<td>50</td>
<td>41</td>
<td>34</td>
</tr>
<tr>
<td></td>
<td>36</td>
<td>18</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>60</td>
<td>41</td>
<td>12</td>
</tr>
</tbody>
</table>
WPMs are good if no closed-form solution exists for the proximal operator.
Thanks & Questions?
Amir Beck.  

Distributed optimization and statistical learning via the alternating direction method of multipliers.  

Tao Hong, Yaniv Romano, and Michael Elad.  
Acceleration of red via vector extrapolation.  

Tao Hong, Irad Yavneh, and Michael Zibulevsky.  
Solving red via weighted proximal methods.  

Jorge Nocedal and Stephen J. Wright.  
*Numerical Optimization*. 