Constraints versus penalties for edge-preserving full-waveform inversion

Felix J. Herrmann
Constraints versus penalties for edge-preserving full-waveform inversion

Felix J. Herrmann & Bas Peters

Bas Peters and Felix J. Herrmann, “Constraints versus penalties for edge-preserving full-waveform inversion”. 2016. Submitted to TLE.
Motivation

**Full-waveform inversion (FWI):**
- hampered by poor data & parasitic local minima
- ill-posed <=> missing frequencies & finite aperture
- should benefit from bounds & structure-promoting priors (TV- or $\ell_1$-norms)

**Efforts met w/ limited success:**
- unpredictable dependence on (unnecessary) hyper parameters
- poor conditioning of structure promoting regularization
- difficulties handling multiple pieces of prior information
FWI

Unconstrained optimization problem:

\[
\begin{align*}
\text{minimize} & \quad f(m) = \frac{1}{2} \sum_{i=1}^{N_s} \|F(m)q_i - d_i\|_2^2 \\
\text{subject to} & \quad m \in \mathbb{R}^m
\end{align*}
\]

Local derivative information is used to update the model:

\[
m_{k+1} = m_k - \gamma \nabla_m f(m_k)
\]

- no insurance model iterates remain (physically/geologically) feasible
- no mitigation of inversion artifacts by controlling model’s complexity
Stylized example

Forward model:

\[ d = F(c)q \equiv c \ast q \]

Unconstrained inversion:

\[
\min_{c \in \mathbb{R}^m} \frac{1}{2} \| F(c)q - d \|_2^2
\]

- when source \( q \) misses low frequencies
- w/o regularization
Stylized example w/ constraints

Regularization via constraints on model:

\[
\minimize_{c \geq c_0} \frac{1}{2} \| F(c)q - d \|^2 \quad \text{subject to} \quad Dc \geq 0
\]

- minimal velocity
- monotonic increasing gradient of the velocity

Leads to successful recovery...
Inversion w/o constraints

velocity

data

inversion

data fit
Inversion w/ constraints
Tikhonov regularization

Add quadratic penalty terms:

$$\min_{m} f(m) + \frac{\alpha}{2} \|R_1 m\|^2 + \frac{\beta}{2} \|R_2 m\|^2$$

- well-known & successful technique
- is differentiable
- not an exact penalty
- regularization may adversely affect gradient & Hessian
- requires non-trivial choices for hyper parameters
- not easily extended to edge-preserving $\ell_1$ - norms
- no guarantees that all model iterates are regularized
Regularization w/ constraints

Add multiple constraints:

\[
\begin{align*}
\text{minimize } f(m) & \quad \text{subject to } m \in C_1 \cap C_2 \\
\end{align*}
\]

- not well-known in our community
- requires understanding of latest optimization techniques
- does not affect gradient & Hessian
- easier parameterization
- able to uniquely project onto intersection of multiple constraint sets
- constraints do not need to be differentiable
- constraints are satisfied at every model iterate

Jean Jacques Moreau
1923–2014
POCS vs. best approximation

“Projection”-onto-convex-sets solves convex feasibility problems:

\[
\text{find } \mathbf{m} \in C_1 \cap C_2
\]

Instead, we solve convex projection problems:

\[
\text{minimize } \|\mathbf{m} - \mathbf{x}\|_2^2 \quad \text{subject to } \mathbf{m} \in C_1 \cap C_2
\]

- obtain optimal (also feasible) approximations
- project uniquely w/ DYKSTRA onto intersections of convex sets
Feasibility vs optimal

Optimal (red) --> unique solution
Feasible (green) --> depends on order

Intersection of circle & square
Feasible (green) --> depends on order

Optimal (red) --> unique solution
Our constraints

**bounds:** $C_1 = \{ m : m \in \text{Box} \}$ where $m \in \text{Box}$ means

$$l_{i,j} \leq m_{i,j} \leq u_{i,j} \quad \forall i,j$$

**total-variation norm ball:** $C_2 = \{ m : \text{TV}(m) \leq \tau \}$

$$\text{TV}(m) = \frac{1}{h} \sum_{ij} \sqrt{(m_{i+1,j} - m_{i,j})^2 + (m_{i,j+1} - m_{i,j})^2}$$
**Proximal projection**

Find a model $\mathbf{m}$, closest to $\mathbf{x}$, such that it satisfies the constraints:

$$
\mathcal{P}_C(\mathbf{x}) = \arg\min_{\mathbf{m}} \| \mathbf{m} - \mathbf{x} \|_2^2 \quad \text{subject to} \quad \mathbf{m} \in \mathcal{C}_1 \cap \mathcal{C}_2
$$

yields

- nonlinear “minimal complexity” best approximation of $\mathbf{m}$
- $\mathbf{m} \rightarrow \mathbf{x}$ when $\tau \rightarrow \tau_0 = \text{TV}(\mathbf{x})$ and $\mathbf{m} \in \text{Box}$
- edge preserving
Best approximations

0.15\tau_0  0.25\tau_0  0.5\tau_0  0.75\tau_0  \tau_0
FWI w/ non-differentiable penalties

Nonlinear least-squares objective for FWI w/ TV:

$$\min_{m} \|d^{obs} - d^{sim}(m)\|^2 + \alpha TV(m)$$

- $TV(m)$ is not differentiable so no access to $\nabla f(m)$ and $\nabla^2 f(m)$
- gradient-descent/quasi-Newton/(Gauss-Newton) solvers need local derivative information
FWI w/ non-differentiable penalties

Nonlinear least-squares objective for FWI w/ TV:

\[
\begin{align*}
\text{minimize} & \quad \| d^{\text{obs}} - d^{\text{sim}}(m) \|^2 + \alpha \text{TV}(m) \\
\end{align*}
\]

- TV\((m)\) is not differentiable so no access to \(\nabla f(m)\) and \(\nabla^2 f(m)\)
- gradient-descent/quasi-Newton/(Gauss-Newton) solvers need local derivative information
Possible solution $\epsilon$-smoothing of TV:

$$TV_\epsilon(m) = \frac{1}{h} \sum_{ij} \sqrt{(m_{i+1,j} - m_{i,j})^2 + (m_{i,j+1} - m_{i,j})^2 + \epsilon^2}$$

Differentiable objective with penalty-parameter $\alpha$:

$$\min_{m} f(m) + \alpha TV_\epsilon(m)$$

**Problem**: need to select 2 unintuitive hyper parameters
Numerical example 1

- FWI w/ smoothed TV-penalty & box constraints
- data w/ zero-mean random noise, $\frac{\|\text{noise}\|_2}{\|\text{signal}\|_2} = 0.25$
- starting model = smoothed true model
- frequency batches from 3 Hz to 10 Hz
increased "blockiness"

FWI results using smoothed TV for various $\alpha$, $\epsilon$ combinations
Constrained formulation

Problem statement:

\[
\min_{\mathbf{m}} \ f(\mathbf{m}) \ \text{subject to} \ \mathbf{m} \in \text{Box} \ \text{and} \ \text{TV}(\mathbf{m}) \leq \tau
\]

Our approach: solve this problem directly.

There are many ways to solve it.
Algorithm design – wish list

- application of constraints should not require additional expensive gradient & objective calculations
- updated models need to satisfy all constraints after each iteration
- arbitrary number of constraints should be handled as long as their intersection is non-empty
- manual tuning of parameters should be limited to bare minimum
- constraints should work w/ black-box gradients & objectives
Nested optimization strategy

**Constrained optimization:**

\[
\minimize_{\mathbf{m}} f(\mathbf{m}) \quad \text{subject to} \quad \mathbf{m} \in C = \bigcap_{i=1}^{p} C_i
\]

via 3 levels of nested optimization:

1. Projected gradients = expensive step
2. Dykstra’s algorithm*
3. Projection onto each set separately (closed form or w/ ADMM)

* parameter free
Projected gradients

Algorithm:

\[ m_{k+1} = \mathcal{P}_C(m_k - \nabla m f(m_k)) \]
Constrained formulation

Algorithm:

\[ m_{k+1} = \mathcal{P}_C(m_k - \nabla_m f(m_k)) \]

projection onto constraint set \hspace{1cm} gradient step (proposed model)

\[ \mathcal{P}_C(m) = \arg \min_x \|x - m\|_2 \quad \text{s.t.} \quad x \in \bigcap_{i=1}^p C_i. \]

intersection of constraint sets
Constrained formulation

Projection onto intersection:

\[ P_C(m) = \arg\min_x \|x - m\|_2 \quad \text{s.t.} \quad x \in \bigcap_{i=1}^{p} C_i. \]

Typically no closed-form solution -> use Dykstra’s algorithm.

Requires:
- projections onto each set separately
- vector additions
Dykstra splitting

Toy example:
find projection onto intersection of circle & square

Algorithm 1 Dykstra.

\[ x_0 = \mathbf{m}, \ p_0 = 0, \ q_0 = 0 \]

For \( k = 0, 1, \ldots \)

\[ y_k = \mathcal{P}_{C_1}(x_k + p_k) \]

\[ p_{k+1} = x_k + p_k - y_k \]

\[ x_{k+1} = \mathcal{P}_{C_2}(y_k + q_k) \]

\[ q_{k+1} = y_k + q_k - x_{k+1} \]

End

Only needs projections onto each set separately!
Individual projections

Projection onto bounds:

\[ P_{C_1}(m_i) = \text{median}\{l_i, m_i, u_i\} \quad \forall i \]

Projection onto TV-ball: split variables, then use ADMM for \( x \) & \( z \)

\[ P_C(m) = \arg \min_x \frac{1}{2} \|x - m\|_2^2 \quad \text{s.t.} \quad x \in C \]

\[ = \arg \min_x \frac{1}{2} \|x - m\|_2^2 \quad \text{s.t.} \quad \|\nabla x\|_1 \leq \sigma \]

\[ = \arg \min_x \frac{1}{2} \|x - m\|_2^2 \quad \text{s.t.} \quad \|z\|_1 \leq \sigma \ , \ \nabla x = z \]
Workflow

User provided code for FWI

model update direction

- gradient
- function value

Optimization algorithm which handles projections (projected & proximal algorithms)

vector

projected vector

- Projector onto set 1
- Projector onto set 2
- Projector onto set p

Algorithm to compute projection onto intersection
User provided code for FWI

Gradient

Function value

Projected gradient algorithm

Dykstra’s algorithm

Bounds: closed-form solution

TV: ADMM

Projector onto set $p$
Workflow

\[ m_{k+1} = P_C(m_k - \nabla_m f(m_k)) \]

User provided code for FWI

\[ f(m_k) \rightarrow \nabla f(m_k) \rightarrow m_{k+1} \]

Bounds:
- Closed-form solution
- Projector onto set \( p \)

\[ P_C(m_i) = \text{median}\{l_i, m_i, u_i\} \]

\[ P_C(m) = \arg \min_x \frac{1}{2} \| x - m \|_2^2 \quad \text{s.t.} \quad \| z \|_1 \leq \sigma, \nabla x = z \]

\[ P_C(m) = \arg \min \| x - m \|_2 \quad \text{s.t.} \quad x \in \bigcap_{i=1}^p C_i \]

Projector onto set \( p \)
Numerical example 1 – revisited

- FWI w/ TV-norm & bound constraints
- data w/ zero-mean random noise, \[ \frac{\| \text{noise} \|_2}{\| \text{signal} \|_2} = 0.25 \]
- starting model = smoothed true model
- frequency batches from 3Hz to 10 Hz
FWI w/ constraints

$0.15\tau_0$

$0.25\tau_0$

$0.5\tau_0$

$0.75\tau_0$
Penalties vs. constraints

Regularization w/ penalties:
- inversion results for various $\alpha$, $\epsilon$ combinations behave unpredictably
- challenges proper parameter settings
- offers no guarantees of feasibility for each model iterate

Regularization w/ constraints:
- inversion results behave predictably for increasing $\tau$
- edges are preserved for not too large $\tau$
- inversion artifacts appear for too large $\tau$

Suggests cooling technique w/ warm starts where $\tau$ is increased slowly...
Case study. Improve delineation of salt for a good starting model but poor (6 dB) data...
Reduced BP model – modelling parameters

- number of sources: 132; number of receivers: 311
- receiver spacing: 40m, source spacing: 80m, max offset 11.5 km
- grid size: 20 m
- known Ricker wavelet sources with 15Hz peak frequency
- data available starting at 3 Hz
- 8 simultaneous shots w/ Gaussian weights w/ redraws
- starting model = smoothed true model
- inversion crime but poor data $||\text{noise}||_2/||\text{signal}||_2 = 0.5$
True velocity model – reduced by a factor of 2.5
Starting model
Adjoint-state w/ noisy data \( \|\text{noise}\|_2/\|\text{signal}\|_2 = 0.5 \)
WRI /w TV-constraints

\[ \frac{\|\text{noise}\|_2}{\|\text{signal}\|_2} = 0.5 \]
Heuristic

Multiple frequency cycles:
- warm starts
- increasingly relaxed TV constraints & fixed bound constraints
- starts w/ relaxed TV-norm of starting model

Extend search space:
- make sure data is fitted
- optimize over model & (source) wavefields
- jointly fit data & wave equation (physics)
- use noise level to automatically select trade-off data & PDE (physics) fit
Wavefield Reconstruction Inversion – gradient


**WRI method:**

for each source $i$

solve \[
\begin{pmatrix}
P_i \\
\lambda A_i(m)
\end{pmatrix}
\begin{pmatrix}
u_{\lambda,i} \\
\end{pmatrix}
\approx
\begin{pmatrix}
d_i \\
\lambda q_i
\end{pmatrix}
\]

$g = g + \lambda^2 \omega^2 \text{diag}(\bar{u}_{i,\lambda})^*(A(m)\bar{u}_{i,\lambda} - q_i)$

end

**Adjoint-state method:**

for each source $i$

solve $A(m)u_i = q_i$

solve $A(m)^*v_i = P_i^*(P_iu_i - d_i)$

$g = g + \omega^2 \text{diag}(u_i)^*v_i$

end

Patent application WO2014172787 – A PENALTY METHOD FOR PDE-CONSTRUINED OPTIMIZATION pending
BP model – inversion parameters

**Optimization specs:**
- spectral-projected gradients
- non-monotone linesearch w/ window size of 5
- max 8 DYKSTRA iterations

**Constraint specs:**
- frequency continuation 3–9 Hz in consecutive batches of 2
- 3 warm started frequency sweeps w/ \( \tau^{l+1} = 1.25 \tau^l \) \& \( \tau^0 = 1.00 \times \text{TV}(m_0) \)
- anisotropic TV
1st cycle cycle

\[ \| \text{noise} \|_2 / \| \text{signal} \|_2 = 0.25 \]

**bounds only**

- FWI
- WRI

**bounds & TV**

- FWI
- WRI
2nd cycle

\[ \frac{\| \text{noise} \|_2}{\| \text{signal} \|_2} = 0.25 \]
3rd cycle

\[
\frac{\text{noise}}{\text{signal}} = 0.25
\]

bounds only

FWI

WRI

bounds & TV
bounds only

FWI

WRI

bounds & TV
2nd cycle \[ \frac{\|\text{noise}\|_2}{\|\text{signal}\|_2} = 0.5 \]

bounds only

FWI

50% noise, FWI, bounds only, 2nd cycle

WRI

50% noise, WRI, bounds only, 2nd cycle

bounds & TV

FWI

50% noise, FWI, TV, 2nd cycle

WRI

50% noise, WRI, TV, 2nd cycle
3rd cycle

\[ \frac{\|\text{noise}\|_2}{\|\text{signal}\|_2} = 0.5 \]

bounds only

FWI

WRI

bounds & TV
Conclusions & generalizations

Adding constraints to inversion:

- leaves gradient (and Hessian) untouched
- is robust & behaves predictably w/ model iterates that remain feasible
- intuitive parameterizations of prior information in >2 constraints
- can be accelerated w/ quasi-Newton & parallelized projections
- “black box” solution that work w/ any implementation for FWI/WRI

*Extensions paired w/ constraints are a powerful combination!*
Acknowledgements

This research was carried out as part of the SINBAD project with the support of the member organizations of the SINBAD Consortium.