IWAVE Implementation of Waveform Inversion via Nonlinear DSO

Dong Sun

The Rice Inversion Project

Jan 29, 2010

Introduction

Focus:

- review Nonlinear DS algorithm for Waveform Inversion
- discuss a general implementation of inversion as a simulation driven optimization

  a C++ wrapper of IWAVE becomes the first necessary and important step

Outline

1. Overview of Nonlinear DSO for Waveform Inversion
2. IWave++: a C++ wrapper of IWAVE
3. Numerical Results
4. Summary & Future Work

Waveform Inversion (WI)

The usual set-up:

- $\mathcal{M}$: Model Space (possible models of earth structure)
- $\mathcal{D}$: Data Space
- $\mathcal{F}$: $\mathcal{M} \to \mathcal{D}$: modeling operator (Forward Map)

WI problem:

given $d \in \mathcal{D}$, find $m \in \mathcal{M}$ such that $\mathcal{F}[m] \approx d$

often in the form of **Least Squares Inversion**:

$$\min_{m \in \mathcal{M}} J_{LS} := \frac{1}{2} \|\mathcal{F}[m] - d\|_{\mathcal{D}}^2 + \mathcal{R}(m)$$

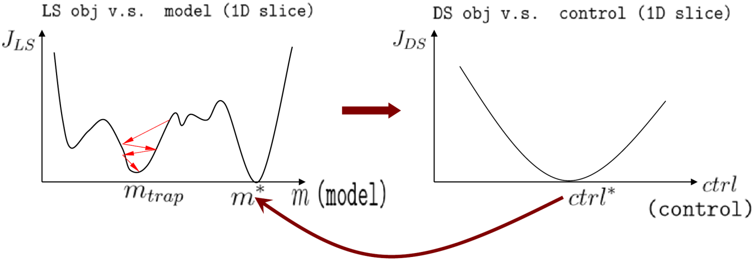
Waveform Inversion is both rewarding and challenging!!!

Motivation of DSO

Waveform Inversion:

- large scale, nonlinear optimization driven by expensive simulation $\Longrightarrow$ only gradient-related methods feasible
- non-convex objective (lots of spurious local minima) for typical band-limited data $\Longrightarrow$ local methods fail without accurate initial model

Differential Semblance addresses local minima issue via convexifying the objective (do inversion over new control space)



General Idea: Extended Modeling

- Data redundancy
  data sets contain sufficient info to produce many partial, overlapping images (extended model) w.r.t. bin/acquisition parameter
- Semblance principle
  correct velocity yields flat/focus image/model gathers, i.e., extended model independent of acquisition parameter
- DS principle (Symes, 1986)
  measure the flatness/"focusness" of extended model by DS operator

Symes (2008): Migration Velocity Analysis is a solution method for the linearized waveform inversion problem. A nonlinear generalization can be formed based on extended modeling.

Nonlinear DSO: Formulation & Key Idea

☞ Waveform Inversion via nonlinear DSO
(Dong's MA thesis (2008) , Dong & Symes (2009))

$$\min_{d_l \in \mathcal{D}_l} J[d_l] := \frac{1}{2} \left\| \frac{\partial m[d_l]}{\partial s} \right\|^2 \qquad \texttt{coherency}$$

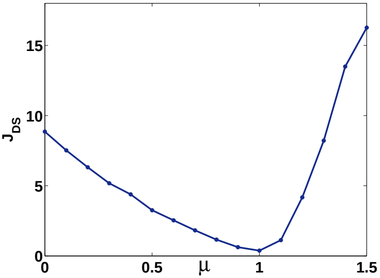
$$\texttt{s.t. } m[d_l] = \underset{m \in \mathcal{M}}{\operatorname{argmin}}\, Q[m] \qquad \texttt{fidelity (data fitting)}$$

$$Q[m] := \frac{1}{2} \left\| \mathcal{F}[m] - d_l - d \right\|^2 + \frac{1}{2}\sigma^2 \left\| \frac{\partial m}{\partial s} \right\|^2 + \mathcal{R}(m)$$

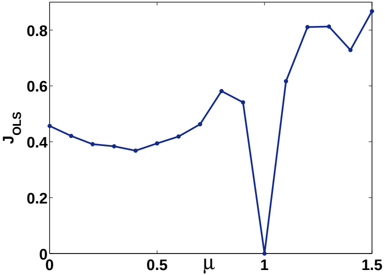
✌ Key idea: low-frequency data components could be a proper control parameter, via updating which to minimize incoherency
(reparametrization of model space with low-frequency data)

Nonlinear DSO: Scan Test

Scan Test: DS objective v.s. LS objective



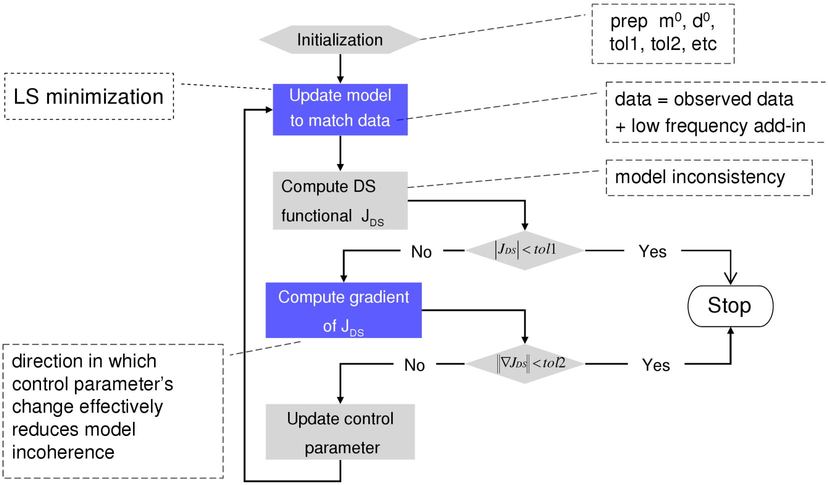
DS objevtive evaluated along a segment in low-frequency control space passing the true missing low-frequency components



LS objevtive evaluated along a segment in model space passing the true model

Nonlinear DSO: Algorithm Flow

nDSO flow



Nonlinear DSO: Main Computation

- Least Squares Inversion

$$\min_{m \in \mathcal{M}} \; Q[m] = \tfrac{1}{2} \left\| \mathcal{F}[m] - d_l - d \right\|^2$$

$$\nabla_m Q = D\mathcal{F}[m]^* \left(\mathcal{F}[m] - (d_o + d_l) \right)$$

$$H = D\mathcal{F}[m]^* \, D\mathcal{F}[m]$$

- Gradient Computation

$$\nabla_{d_l} J = \Pi \; D\mathcal{F}[m] \; H^{-1} \; \left(\frac{\partial}{\partial s} \right)^* \frac{\partial}{\partial s}$$

common blocks: actions of $D\mathcal{F}[m]$ and its adjoint map $D\mathcal{F}[m]^*$

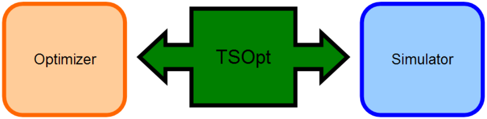
Implementation of Inversion

Implementation requires multiple levels of abstraction

- ✦ simulation depends on physics & its numerical realization
- ✧ linear algebra & optimization algorithms only involve mathematical concepts (vectors, operators, gradient, ...)

Existing packages (Marco)

- RVL (Rice Vector Library): interfaces for optimization and linear algebra algorithms
- Umin: algorithms for unconstrained minimization
- TSOpt: interfaces for timestepping algorithms
- IWAVE: a parallel framework built in C for solving time-dependent partial differential equations with lots of modeling options already implemented for acoustics



First step towards implementing seismic inversion with existing packages: wrap simulation (IWAVE) as a RVL operator, with three common methods, to compute

- value (modeling)
- first derivative (action of Born Map)
- adjoint derivative (adjoint action of Born Map)

This way

- handles multiple levels of abstraction natuarally
- seperates simulation from optimization

Outline

1. Overview of Nonlinear DSO for Waveform Inversion
2. IWave++: a C++ wrapper of IWAVE
3. Numerical Results
4. Summary & Future Work

Model Problem

- Forward Simulation:

$$\frac{1}{\kappa(\mathbf{x})}\frac{\partial p}{\partial t} + \nabla \cdot \mathbf{v} = f(\mathbf{x}, t),$$
$$\frac{1}{b(\mathbf{x})}\frac{\partial \mathbf{v}}{\partial t} + \nabla p = 0$$

  Forward map: $\mathcal{F}[m] := Sp$, S sampling operator

- Born (linearied) map at model m: $D\mathcal{F}[m]\delta m := S\delta p$

$$\frac{1}{\kappa(\mathbf{x})}\frac{\partial \delta p}{\partial t} + \nabla \cdot \delta\mathbf{v} = -\frac{\delta\kappa(\mathbf{x})}{\kappa}\nabla \cdot \mathbf{v},$$
$$\frac{1}{b}\frac{\partial \delta\mathbf{v}}{\partial t} + \nabla \delta p = -\frac{\delta b(\mathbf{x})}{b(\mathbf{x})}\nabla p$$

Time Stepping in Forward & Born Simulations

- Forward Simulation $\mathcal{F}[m] := Sp$

$$
\begin{aligned}
p &= p \;\; -\kappa\, \Delta t\, \nabla \cdot \mathbf{v} \;\; + \texttt{source} \\
\mathbf{v} &= \mathbf{v} \;\; -b\, \Delta t\, \nabla p
\end{aligned}
$$

- Born (linearied) map $D\mathcal{F}[m]\delta m := S\delta p$

$$
\begin{aligned}
\delta p &= \delta p \;\; -\kappa\, \Delta t\, \nabla \cdot \delta\mathbf{v} \;\; -\delta\kappa\, \Delta t\, \nabla \cdot \mathbf{v} \\
\delta\mathbf{v} &= \delta\mathbf{v} \;\; -b\Delta t\, \nabla \delta p \;\; -\delta b\, \Delta t\, \nabla p
\end{aligned}
$$

Adjoint Computation

- Backward Simulation

$$\xi_p^k = \xi_p^{k+1} \; -\kappa\widehat{\Delta t}\,\nabla\cdot\xi_{\mathbf{v}}^{k+\frac{1}{2}} \; + \texttt{source}$$

$$\xi_{\mathbf{v}}^{k-\frac{1}{2}} = \xi_{\mathbf{v}}^{k+\frac{1}{2}} \; -b\,\widehat{\Delta t}\,\nabla_p \xi_p^k$$

- Image Accumulation

$$\mathcal{I}_\kappa^{k-1} = \mathcal{I}_\kappa^k \; -\xi_p^k\,\Delta t \nabla\cdot\mathbf{v}^{k-\frac{1}{2}},$$

$$\mathcal{I}_b^{k-1} = \mathcal{I}_b^k \; -\xi_{\mathbf{v}}^{k-\frac{1}{2}}\,\Delta t \nabla p^{k-1}$$

$$\mathcal{I} = \begin{pmatrix} \frac{1}{\kappa}\mathcal{I}_\kappa^1 \\ \\ \frac{1}{b}\mathcal{I}_b^1 \end{pmatrix}$$

IWAVE++: Forward Simulation

```
IWaveState
-----------
IWAVE iwstate;
TSIndex tsi; ...
```

```
IWaveOp
-----------
apply(x,y);
applyDeriv(x,dx,dy);
applyAdjDeriv(x,dy,dx);
...
```

Simulation flow (IWaveOP::apply)

1. initialization
2. sim loop (wavefield updating)
   (1) update wavefields & exchange info (`iwave_run()`)
   (2) post-step
      ① insert source
      ② sample & record results
      ③ update time

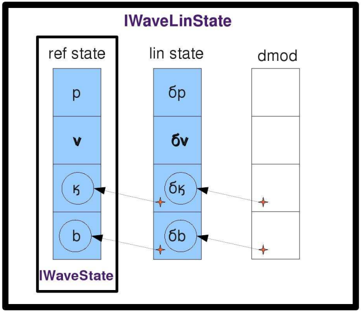
Core Computations: $p = p - \kappa \Delta t \nabla \cdot \mathbf{v}$, $\mathbf{v} = \mathbf{v} - b \Delta t \nabla p$

```
TIMESTEP_FUN ts = (RDOM *dom, int iarr, void *pars);
```

IWave++: Born Simulation

Simulation flow (IWaveOP::applyDeriv)

1. init both ref & pert fields
2. lin sim loop
   (1) update pert-fields
   (2) post-step:
      ① insert born source
      ② sample traces
      ③ update time
   (3) synchronize fields
      run ref sim to current time level of lin fields



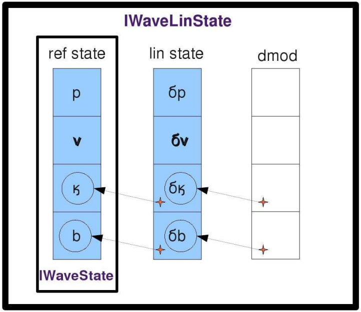
Born Src: $\delta p \mathrel{+}= -\delta\kappa\,\Delta t \nabla\cdot\mathbf{v}$, $\delta\mathbf{v} \mathrel{+}= -\delta b\,\Delta t\nabla p$

```
GEN_TIMESTEP_FUN gts = (RDOM *ddom, RDOM* rdom, RDOM* pdom, int
fwd, ...);
```

IWave++: Adjoint Action of Born Map

Simulation flow (IWaveOP::applyAdjDeriv)

1. init both fwd- & bwd- fields
2. bwd-sim loop
   (1) synchronize fields
   run fwd-sim to current time level of bwd-fields
   (2) update bwd-fields
   (3) post-step:
   ① insert bwd-source
   ② image accumulation
   ③ update bwd-time



Adj Born Src: $I_p \mathrel{+}= -\frac{1}{\kappa}\,\xi_p \Delta t \nabla \cdot \mathbf{v}, \quad I_{\mathbf{b}} \mathrel{+}= -\frac{1}{\mathbf{b}}\,\xi_{\mathbf{v}} \Delta t \nabla p$

```
GEN_TIMESTEP_FUN gts = (RDOM* ddom, RDOM* rdom, RDOM* pdom, int
fwd, ...);
```

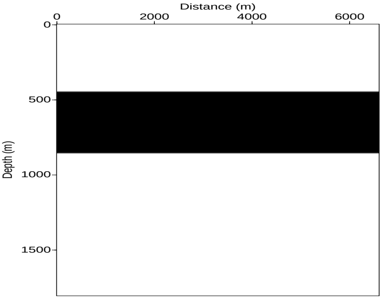
Outline

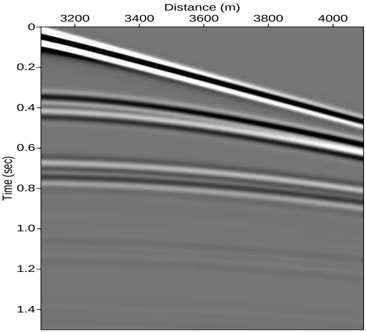
1. Overview of Nonlinear DSO for Waveform Inversion
2. IWave++: a C++ wrapper of IWAVE
3. Numerical Results
4. Summary & Future Work

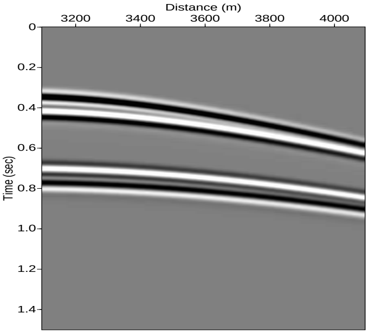
Lin Forward Map

$\kappa = 11109$ MPa, $\rho = 2100\ kg/m^3$, $c = 2.3\ km/s$, $\delta\kappa = 2641$ Mpa, $\delta\rho = 100\ kg/m^3$, $\delta c = 0.2\ km/s$, src $(3000, 40)$, receivers at depth 80

primary reflections around 0.4s and 0.7s; multiple reflections around 0.8s, 1.2s, 1.4s, ...



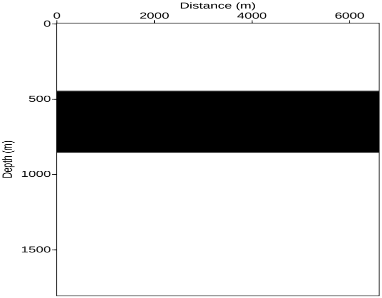


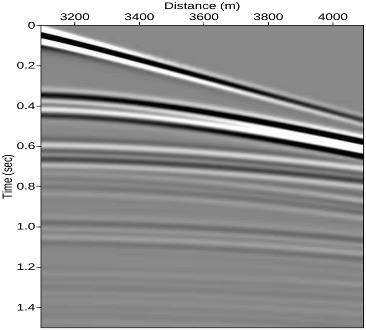


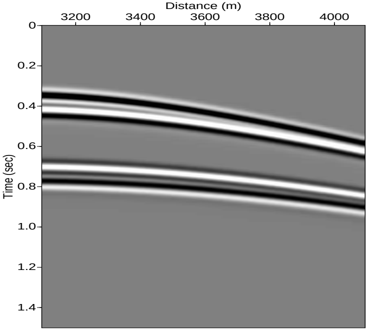
Lin Forward Map

$\kappa = 11109$ MPa, $\rho = 2100\ kg/m^3$, $c = 2.3\ km/s$, $\delta\kappa = 12849$ Mpa, $\delta\rho = 100\ kg/m^3$, $\delta c = 1.0\ km/s$, src $(3000, 40)$, receivers at depth 80

primary reflections around 0.4s and 0.6s; multiple reflections around 0.8s, 1.2s, 1.4s, ...

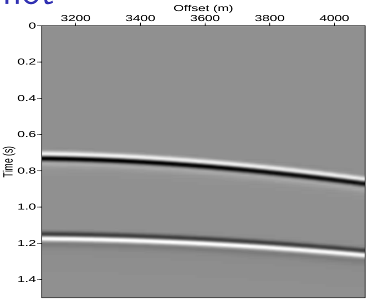


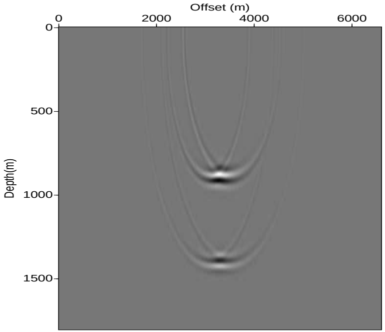


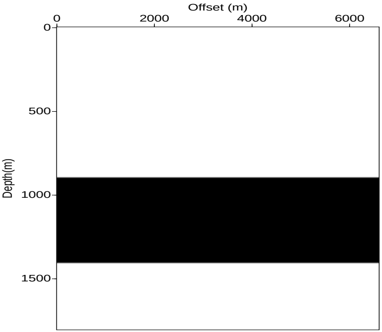


Adjoint Computation: Single Shot

$\kappa = 11109$ MPa, $\rho = 2100\ kg/m^3$, $c = 2.3\ km/s$, $\delta\kappa = 987$ Mpa, $\delta c = 0.1\ km/s$,
sx 3000 at depth 40,
gx 3100 - 4090 at depth 80

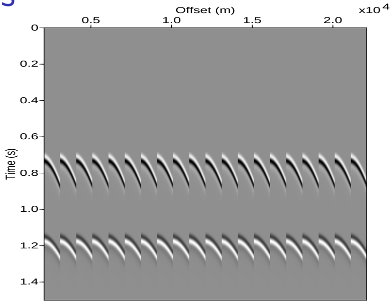


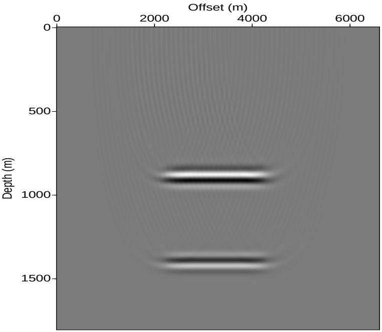


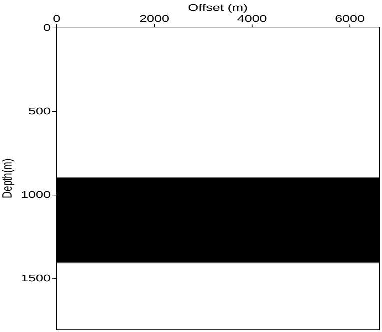


Adjoint Computation: 20 Shots

$\kappa = 11109$ MPa, $\rho = 2100\ kg/m^3$, $c = 2.3\ km/s$, $\delta\kappa = 987$ Mpa, $\delta c = 0.1\ km/s$,
sx 2000 - 3900 at depth 40,
gx 2100 - 4990 at depth 80







Summary & Future Work

Done:

- Review nonlinear DSO
  formulate WI via extended modeling concept
  low-frequency data is a good analog to "macro-model" and used as controls
- introduce IWave++: a wrapper of IWAVE, with three common methods
  seperate simulation from developing optimization algorithms

Doing:

- dot product test

To Do:

- DS operator and its adjoint;
- Various tests on DSO and regular LS inversion

Acknowledgements

- TRIP members
- TRIP sponsors
- NSF DMS 0620821

- Thank you for listening

Appendix

Gradient Computation: derivation

- $J[d_l] = \frac{1}{2} \left\| \frac{\partial \bar{m}[d_l]}{\partial \xi} \right\|^2_{\mathsf{M}} \Rightarrow \delta J = -\left(\frac{\partial^2 \bar{m}}{\partial \xi^2}, \delta \bar{m} \right)_{\mathsf{M}}$
- First-order necessity condition of the sub-minimization problem

$$\nabla_{\bar{m}} Q = 0$$

where

$$\nabla_{\bar{m}} Q = D\overline{\mathcal{F}}[\bar{m}]^T \left(\overline{\mathcal{F}}[\bar{m}] - (d_o + d_l) \right) - \sigma^2 \frac{\partial}{\partial \xi} \triangle^{-1} \frac{\partial}{\partial \xi} \bar{m} + D\mathcal{R}(\bar{m})$$

$\Rightarrow$

$$H_Q \, \delta \bar{m} = D\overline{\mathcal{F}}[\bar{m}]^T \delta d_l$$

where

$$H_Q := D\overline{\mathcal{F}}[\bar{m}]^T D\overline{\mathcal{F}}[\bar{m}] - \sigma^2 \frac{\partial}{\partial \xi} \triangle^{-1} \frac{\partial}{\partial \xi} + D^2 \mathcal{R}$$

$\Rightarrow$

$$\delta \bar{m} = D_{d_l} \bar{m} \, \delta d_l$$

$$D_{d_l} \bar{m} = H_Q^{-1} D\overline{\mathcal{F}}[\bar{m}]^T$$

Gradient Computation: derivation

$$
\begin{aligned}
\delta J &= -\left(\frac{\partial^2 \bar{m}}{\partial \xi^2}, \delta \bar{m}\right)_{\overline{\mathcal{M}}} \\
&= -\left(\frac{\partial^2 \bar{m}}{\partial \xi^2}, D_{d_l} \bar{m} \ \delta d_l\right)_{\overline{\mathcal{M}}} \\
&= -\left((D_{d_l} \ \bar{m})^T \frac{\partial^2 \bar{m}}{\partial \xi^2}, \delta d_l\right)_{\mathcal{D}}
\end{aligned}
$$

Hence,

$$
\begin{aligned}
\nabla J &= -\,(D_{d_l} \ \bar{m})^T \ \tfrac{\partial^2 \bar{m}}{\partial \xi^2} \\
&= -D\overline{\mathcal{F}}[\bar{m}] \ H_Q^{-1} \ \tfrac{\partial^2 \bar{m}}{\partial \xi^2}
\end{aligned}
$$

Recall

$$
H_Q = D\overline{\mathcal{F}}[\bar{m}]^T D\overline{\mathcal{F}}[\bar{m}] - \sigma^2 \frac{\partial}{\partial \xi} \triangle^{-1} \frac{\partial}{\partial \xi} + D^2 \mathcal{R}
$$

Key computation

$$
H_Q \ q = b
$$

such as

$$
H_Q \delta \bar{m} = -\nabla_{\bar{m}} Q \quad , \quad H_Q \delta \bar{m} = D\overline{\mathcal{F}}[\bar{m}]^T \delta d_l
$$

Formulate WI via Extended Modeling

Extended Modeling Concept $\longrightarrow$ a unified view of OLS and MVA (Symes, 2008)

The *extension* of model $\mathcal{F} : \mathcal{M} \longrightarrow \mathcal{D}$ consists of

- $\overline{\mathcal{M}}$: extended model space
- $E : \mathcal{M} \longrightarrow \overline{\mathcal{M}}$: extension operator, one-to-one, $E[\mathcal{M}] \subset \overline{\mathcal{M}}$ ($E[\mathcal{M}]$: the "physical models")
- $\overline{\mathcal{F}} : \overline{\mathcal{M}} \longrightarrow \mathcal{D}$: extended modeling operator, $\mathcal{F}[m] = \overline{\mathcal{F}}[E[m]]$ for any $m \in \mathcal{M}$

Extended inversion:

given $d \in \mathbf{D}$, find $\bar{m} \in \overline{\mathcal{M}}$ such that $\overline{\mathcal{F}}[\bar{m}] \simeq d$

solution $\bar{m}$ physically meaningful only if $\bar{m} \in E[\mathcal{M}]$

Since $\overline{\mathcal{M}}$ has more degrees of freedom, ambiguity is more likely.

Extended Modeling may lead to Effective WI

Extension concept (Symes,2008)

- provides a unified view of WI and MVA
  in linearized extended modeling context, MVA is a solution method to the partially linearized inverse problem

- has lots of familiar extensions
  annihilator A chosen in differential semblance class , lots of successful implementations and theoretical results
  (Symes(1990), Symes & Carazzone(1991), Symes(1999),
  Shen & Calandra(2005),...)

- suggests an approach to nonlinear waveform inversion incorporating elements of MVA
  Symes(1991) proved this problem is equivalent to an unconstrained problem with no local minima and the objective has stable shape independent of source spectrum (under some assumption ...)