The Inverse Regional Ocean Modeling System:

Development and Application to Data Assimilation of Coastal Mesoscale Eddies.

Di Lorenzo, E., Moore, A., H. Arango, B. Chua, B. D. Cornuelle , A. J. Miller and Bennett A.

Goals

•Overview of the ***Inverse Regional Ocean Modeling System***

• ***Implementation*** - How do we assimilate data using the ROMS set of models

• Examples, ***(a)*** Coastal upwelling ***(b)*** mesoscale eddies in the Southern California Current

Inverse Ocean Modeling System (IOMs)

Chua and Bennett (2001)

To implement a ***representer-based*** generalized inverse method to solve weak constraint data assimilation into a non-linear model

↓

NL-ROMS, TL-ROMS, REP-ROMS, AD-ROMS

Moore et al. (2003)

↓

Inverse Regional Ocean Modeling System (IROMS)

a ***4D-variational data assimilation system*** for high-resolution basin-wide and coastal oceanic flows

NL-ROMS: $$\frac{\partial \mathrm{u}}{\partial t} = N(\mathrm{u}) + \mathrm{F}(t)$$

REP-ROMS: $$\frac{\partial \mathrm{u}}{\partial t} = N(\mathrm{u}_B) + \mathrm{A}\left(\mathrm{u} - \mathrm{u}_B\right) + \mathrm{F}(t)$$

def:

$$\boxed{\mathrm{A} \equiv \left.\frac{\partial N}{\partial \mathrm{u}_B}\right|_{\mathrm{u}_B}}$$

Approximation of NONLINEAR DYNAMICS ***(STEP 1)***

do $n = 1 \rightarrow \infty$

$$\mathrm{u}_B \equiv \mathrm{u}^{n-1}$$

$$\frac{\partial \mathrm{u}^n}{\partial t} = N(\mathrm{u}_B) + \mathrm{A}\left(\mathrm{u}^n - \mathrm{u}_B\right) + \mathrm{F}(t)$$

enddo

$$\mathrm{u}^n \longrightarrow \mathrm{u}$$

also referred to as ***Picard Iterations***

REP-ROMS: $$\frac{\partial \mathrm{u}}{\partial t} = N(\mathrm{u}_B) + \mathrm{A}\left(\mathrm{u} - \mathrm{u}_B\right) + \mathrm{F}(t)$$

def:

$$\mathrm{A} \equiv \left.\frac{\partial N}{\partial \mathrm{u}_B}\right|_{\mathrm{u}_B}$$

$$\mathrm{s} = \mathrm{u} - \mathrm{u}_B$$

def:

$$A \equiv \left.\frac{\partial N}{\partial \mathrm{u}_B}\right|_{\mathrm{u}_B}$$

$$\mathrm{s} = \mathrm{u} - \mathrm{u}_B$$

REP-ROMS: $$\frac{\partial \mathrm{u}}{\partial t} = N(\mathrm{u}_B) + A\left(\mathrm{u} - \mathrm{u}_B\right) + F(t)$$

TL-ROMS: $$\frac{\partial \mathrm{s}}{\partial t} = A\mathrm{s}$$

AD-ROMS: $$\frac{\partial \boldsymbol{\lambda}}{\partial t} = A^T \boldsymbol{\lambda}$$

REP-ROMS: $$\frac{\partial \mathrm{u}}{\partial t} = N(\mathrm{u}_B) + \mathrm{A}\left(\mathrm{u} - \mathrm{u}_B\right) + \mathrm{F}(t) + \mathrm{e}(t)$$

TL-ROMS: $$\frac{\partial \mathrm{s}}{\partial t} = \mathrm{A}\mathrm{s} + \mathrm{e}(t)$$

AD-ROMS: $$\frac{\partial \boldsymbol{\lambda}}{\partial t} = \mathrm{A}^T \boldsymbol{\lambda}$$

(STEP 2)

Small Errors

1) *model missing dynamics*
2) *boundary conditions errors*
3) *Initial conditions errors*

REP-ROMS: $$\frac{\partial \mathrm{u}}{\partial t} = N(\mathrm{u}_B) + \mathrm{A}\left(\mathrm{u} - \mathrm{u}_B\right) + \mathrm{F}(t) + \mathrm{e}(t)$$

TL-ROMS: $$\frac{\partial \mathrm{s}}{\partial t} = \mathrm{A}\mathrm{s} + \mathrm{e}(t)$$

AD-ROMS: $$\frac{\partial \boldsymbol{\lambda}}{\partial t} = \mathrm{A}^T \boldsymbol{\lambda}$$

Integral Solutions

REP-ROMS: $$\frac{\partial \mathrm{u}}{\partial t} = N(\mathrm{u}_B) + \mathrm{A}\left(\mathrm{u} - \mathrm{u}_B\right) + \mathrm{F}(t) + \mathrm{e}(t)$$

TL-ROMS: $$\frac{\partial \mathrm{s}}{\partial t} = \mathrm{A}\mathrm{s} + \mathrm{e}(t)$$

$$\mathrm{s}(t_N) = \mathrm{R}(t_0, t_N)\mathrm{s}(t_0) + \int_{t_0}^{t_N} \mathrm{R}(t', t_N)\mathrm{e}(t')dt'$$

AD-ROMS: $$\frac{\partial \lambda}{\partial t} = \mathrm{A}^T \lambda$$

Tangent Linear Propagator

Integral Solutions

REP-ROMS: $$\frac{\partial \mathrm{u}}{\partial t} = N(\mathrm{u}_B) + \mathrm{A}\left(\mathrm{u} - \mathrm{u}_B\right) + \mathrm{F}(t) + \mathrm{e}(t)$$

TL-ROMS: $$\frac{\partial \mathrm{s}}{\partial t} = \mathrm{A}\mathrm{s} + \mathrm{e}(t)$$

$$\mathrm{s}(t_N) = \mathrm{R}(t_0, t_N)\mathrm{s}(t_0) + \int_{t_0}^{t_N} \mathrm{R}(t', t_N)\mathrm{e}(t')dt'$$

AD-ROMS: $$\frac{\partial \boldsymbol{\lambda}}{\partial t} = \mathrm{A}^T \boldsymbol{\lambda}$$

Adjoint Propagator

$$\boldsymbol{\lambda}(t_0) = \mathrm{R}^T(t_N, t_0)\boldsymbol{\lambda}(t_N)$$

Integral Solutions

REP-ROMS:

$$\mathrm{u}(t_N) = \mathrm{R}(t_0, t_N)\mathrm{u}(t_0) + \int_{t_0}^{t_N} \mathrm{R}(t', t_N)\left[N(\mathrm{u}_B) + \mathrm{F}(t') + \mathrm{e}(t')\right] dt'$$

Tangent Linear Propagator

TL-ROMS:

$$\mathrm{s}(t_N) = \mathrm{R}(t_0, t_N)\mathrm{s}(t_0) + \int_{t_0}^{t_N} \mathrm{R}(t', t_N)\mathrm{e}(t') dt'$$

AD-ROMS:

Adjoint Propagator

$$\boldsymbol{\lambda}(t_0) = \mathrm{R}^T(t_N, t_0)\boldsymbol{\lambda}(t_N)$$

Integral Solutions

REP-ROMS:

$$\mathrm{u}(t_N) = \mathrm{R}(t_0, t_N)\mathrm{u}(t_0) + \int_{t_0}^{t_N} \mathrm{R}(t', t_N)\left[N(\mathrm{u}_B) + \mathrm{F}(t') + \mathrm{e}(t')\right]dt'$$

TL-ROMS:

Tangent Linear Propagator

$$\mathrm{s}(t_N) = \mathrm{R}(t_0, t_N)\mathrm{s}(t_0) + \int_{t_0}^{t_N} \mathrm{R}(t', t_N)\mathrm{e}(t')dt'$$

$$\mathrm{R}$$

AD-ROMS:

Adjoint Propagator

$$\boldsymbol{\lambda}(t_0) = \mathrm{R}^T(t_N, t_0)\boldsymbol{\lambda}(t_N)$$

$$\mathrm{R}^T$$

How is the tangent linear model useful for assimilation?

TL-ROMS:

$$\boxed{\mathrm{s}(t_N) = \mathrm{R}(t_0, t_N)\mathrm{s}(t_0) + \int_{t_0}^{t_N} \mathrm{R}(t', t_N)\mathrm{e}(t')dt'}$$

ASSIMILATION (1)

Problem Statement

1) Set of observations $\longrightarrow \mathrm{d}$

2) Model trajectory $\longrightarrow \mathrm{u}(t)$

3) Find $\hat{\mathrm{u}}(t)$ *that minimizes* $\longrightarrow \mathrm{d} - \int_{t_0}^{T} \mathrm{H}(t')\hat{\mathrm{u}}(t)dt'$

Sampling functional

TL-ROMS:

$$\mathrm{s}(t_N) = \mathrm{R}(t_0, t_N)\mathrm{s}(t_0) + \int_{t_0}^{t_N} \mathrm{R}(t', t_N)\mathrm{e}(t')dt'$$

ASSIMILATION (1)

Problem Statement

1) Set of observations $\rightarrow \mathrm{d}$

2) Model trajectory $\rightarrow \mathrm{u}(t)$

3) Find $\hat{\mathrm{u}}(t)$ *that minimizes* $\rightarrow \mathrm{d} - \int_{t_0}^{T} \mathrm{H}(t')\hat{\mathrm{u}}(t)dt'$

Sampling functional

TL-ROMS:

$$\mathrm{s}(t_N) = \mathrm{R}(t_0, t_N)\mathrm{s}(t_0) + \int_{t_0}^{t_N} \mathrm{R}(t', t_N)\mathrm{e}(t')dt'$$

$$\hat{\mathrm{u}}(t) = \mathrm{u}(t) + \mathrm{s}(t)$$

Best Model Estimate **Initial Guess** **Corrections**

ASSIMILATION (2) *Modeling the Corrections*

1) Initial model-data misfit $\rightarrow \hat{\mathrm{d}} \equiv \mathrm{d} - \int_{t_0}^{T} \mathrm{H}(t')\mathrm{u}(t)dt'$

2) Model Tangent Linear trajectory $\rightarrow \mathrm{s}(t)$

3) Find $\mathrm{s}(t)$ *that minimizes* $\rightarrow \hat{\mathrm{d}} - \int_{t_0}^{T} \mathrm{H}(t')\mathrm{s}(t')dt'$

TL-ROMS:

$$\mathrm{s}(t_N) = \mathrm{R}(t_0, t_N)\mathrm{s}(t_0) + \int_{t_0}^{t_N} \mathrm{R}(t', t_N)\mathrm{e}(t')dt'$$

$$\hat{\mathrm{u}}(t) = \mathrm{u}(t) + \mathrm{s}(t)$$

Best Model Estimate **Initial Guess** **Corrections**

ASSIMILATION (2)

Modeling the Corrections

1) Initial model-data misfit $\rightarrow \hat{\mathrm{d}} \equiv \mathrm{d} - \int_{t_0}^{T} \mathrm{H}(t')\mathrm{u}(t)dt'$

2) Model Tangent Linear trajectory $\rightarrow \mathrm{s}(t)$

3) Find $\mathrm{s}(t)$ *that minimizes* $\rightarrow \hat{\mathrm{d}} - \int_{t_0}^{T} \mathrm{H}(t')\mathrm{s}(t')dt'$

TL-ROMS:

$$\mathrm{s}(t_N) = \cancel{\mathrm{R}(t_0, t_N)\mathrm{s}(t_0) +} \int_{t_0}^{t_N} \mathrm{R}(t', t_N)\mathrm{e}(t')dt'$$

$$\mathrm{s}(t_0) = \mathrm{e}_0 \delta(t - t_0) = \mathrm{e}(t_0)$$

ASSIMILATION (2) *Modeling the Corrections*

1) Initial model-data misfit $\rightarrow \hat{\mathrm{d}} \equiv \mathrm{d} - \int_{t_0}^{T} \mathrm{H}(t')\mathrm{u}(t)dt'$

2) Model Tangent Linear trajectory $\rightarrow \mathrm{s}(t)$

3) Find $\mathrm{s}(t)$ *that minimizes* $\rightarrow \hat{\mathrm{d}} - \int_{t_0}^{T} \mathrm{H}(t')\mathrm{s}(t')dt'$

TL-ROMS:

$$\mathrm{s}(t_N) = \cancel{\mathrm{R}(t_0, t_N)\mathrm{s}(t_0) +} \int_{t_0}^{t_N} \mathrm{R}(t', t_N)\mathrm{e}(t')dt'$$

$$\mathrm{s}(t_0) = \mathrm{e}_0 \delta(t - t_0) = \mathrm{e}(t_0)$$

$\mathrm{e}(t_0)$ → *Corrections to initial conditions*

$\mathrm{e}(t)$ → *Corrections to model dynamics and boundary conditions*

ASSIMILATION (2)

Modeling the Corrections

1) Initial model-data misfit $\longrightarrow \hat{\mathrm{d}}$

2) Corrections to Model State $\longrightarrow \mathrm{e}(t)$

3) Find $\mathrm{e}(t)$ *that minimizes* $\longrightarrow \hat{\mathrm{d}} - \int_{t_0}^{T} \mathrm{H}(t') \underbrace{\int_{t_0}^{t'} \mathrm{R}(t'',t')\mathrm{e}(t'')dt''}_{\mathrm{s}(t)} dt'$

$\mathrm{e}(t_0)$ → *Corrections to initial conditions*

$\mathrm{e}(t)$ → *Corrections to model dynamics and boundary conditions*

ASSIMILATION (2)　　*Modeling the Corrections*

1) *Initial model-data misfit* $\rightarrow \hat{\mathrm{d}}$

2) *Correction to Model Initial Guess* $\rightarrow \mathrm{e}(t_0)$

3) *Find* $\mathrm{e}(t_0)$ *that minimizes* $\rightarrow \hat{\mathrm{d}} - \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0, t')\mathrm{e}(t_0)dt'$

Assume we seek to correct only the initial conditions

⬇

STRONG CONSTRAINT

$\mathrm{e}(t_0)$ → *Corrections to initial conditions*

$\cancel{\mathrm{e}(t)}$ → *Corrections to model dynamics and boundary conditions*

ASSIMILATION (2)

Modeling the Corrections

1) Initial model-data misfit $\rightarrow \hat{\mathrm{d}}$

2) Correction to Model Initial Guess $\rightarrow \mathrm{e}(t_0)$

3) Find $\mathrm{e}(t_0)$ *that minimizes* $\rightarrow \hat{\mathrm{d}} - \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0,t')\mathrm{e}(t_0)dt'$

ASSIMILATION (3)

Cost Function

$$J[\mathrm{e}_0] = \left[\hat{\mathrm{d}} - \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0,t')dt'\mathrm{e}_0\right]^T \mathrm{C}_{\boldsymbol{\varepsilon}}^{-1} \left[\hat{\mathrm{d}} - \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0,t')dt'\mathrm{e}_0\right]$$

$$+\mathrm{e}_0^T \mathrm{P}^{-1} \mathrm{e}_0$$

ASSIMILATION (2) *Modeling the Corrections*

1) *Initial model-data misfit* $\longrightarrow \hat{\mathrm{d}}$

2) *Correction to Model Initial Guess* $\longrightarrow \mathrm{e}(t_0)$

3) *Find* $\mathrm{e}(t_0)$ *that minimizes* $\longrightarrow \hat{\mathrm{d}} - \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0,t')\mathrm{e}(t_0)dt'$

ASSIMILATION (3) *Cost Function*

$$J[\mathrm{e}_0] = \left[\hat{\mathrm{d}} - \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0,t')dt'\mathrm{e}_0\right]^T \mathrm{C}_{\varepsilon}^{-1} \left[\hat{\mathrm{d}} - \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0,t')dt'\mathrm{e}_0\right]$$

$$+\mathrm{e}_0^T \mathrm{P}^{-1} \mathrm{e}_0$$

1) corrections should reduce misfit within observational error

2) corrections should not exceed our assumptions about the errors in model initial condition.

ASSIMILATION (3)

Cost Function

$$J[\mathrm{e}_0] = \left[\hat{\mathrm{d}} - \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0, t')dt'\mathrm{e}_0\right]^T \mathrm{C}_{\boldsymbol{\varepsilon}}^{-1} \left[\hat{\mathrm{d}} - \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0, t')dt'\mathrm{e}_0\right]$$

$$+\mathrm{e}_0^T \mathrm{P}^{-1} \mathrm{e}_0$$

G **is a mapping matrix of dimensions**
observations X model space

def:

$$\mathrm{G} = \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0,t')dt'$$

ASSIMILATION (3)

Cost Function

$$J[\mathrm{e}_0] = \left[\hat{\mathrm{d}} - \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0,t')dt'\mathrm{e}_0\right]^T \mathrm{C}_{\boldsymbol{\varepsilon}}^{-1} \left[\hat{\mathrm{d}} - \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0,t')dt'\mathrm{e}_0\right]$$

$$+\mathrm{e}_0^T \mathrm{P}^{-1} \mathrm{e}_0$$

G **is a mapping matrix of dimensions**
observations X model space

def:

$$\mathrm{G} = \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0,t')dt'$$

ASSIMILATION (3)

Cost Function

$$J[\mathrm{e}_0] = \left[\hat{\mathrm{d}} - \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0,t')dt'\mathrm{e}_0\right]^T \mathrm{C}_{\boldsymbol{\varepsilon}}^{-1}\left[\hat{\mathrm{d}} - \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0,t')dt'\mathrm{e}_0\right]$$
$$+\mathrm{e}_0^T\mathrm{P}^{-1}\mathrm{e}_0$$

$$J[\mathrm{e}_0] = \left[\hat{\mathrm{d}} - \mathrm{G}\mathrm{e}_0\right]^T \mathrm{C}_{\boldsymbol{\varepsilon}}^{-1}\left[\hat{\mathrm{d}} - \mathrm{G}\mathrm{e}_0\right] + \mathrm{e}_0^T\mathrm{P}^{-1}\mathrm{e}_0$$

$$\left(\mathrm{G}^T \mathrm{C}_\varepsilon^{-1} \mathrm{G} + \mathrm{P}^{-1}\right) \mathrm{e}_0 - \mathrm{G}^T \mathrm{C}_\varepsilon^{-1} \hat{\mathrm{d}} = 0$$

Minimize *J*

ASSIMILATION (3)

Cost Function

$$J[\mathrm{e}_0] = \left[\hat{\mathrm{d}} - \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0, t')dt'\mathrm{e}_0\right]^T \mathrm{C}_\varepsilon^{-1} \left[\hat{\mathrm{d}} - \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0, t')dt'\mathrm{e}_0\right]$$

$$+\mathrm{e}_0^T \mathrm{P}^{-1} \mathrm{e}_0$$

$$J[\mathrm{e}_0] = \left[\hat{\mathrm{d}} - \mathrm{G}\mathrm{e}_0\right]^T \mathrm{C}_\varepsilon^{-1} \left[\hat{\mathrm{d}} - \mathrm{G}\mathrm{e}_0\right] + \mathrm{e}_0^T \mathrm{P}^{-1} \mathrm{e}_0$$

4DVAR inversion

$$\underbrace{\left(\mathrm{G}^T \mathrm{C}_\varepsilon^{-1} \mathrm{G} + \mathrm{P}^{-1}\right)}_{\hat{\mathrm{H}}} \mathrm{e}_0 - \mathrm{G}^T \mathrm{C}_\varepsilon^{-1} \hat{\mathrm{d}} = 0$$

Hessian Matrix

def:

$$\boxed{\mathrm{G} = \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0, t')\,dt'}$$

4DVAR inversion

$$\underbrace{\left(\mathrm{G}^T\mathrm{C}_\varepsilon^{-1}\mathrm{G}+\mathrm{P}^{-1}\right)}_{\hat{\mathrm{H}}}\mathrm{e}_0-\mathrm{G}^T\mathrm{C}_\varepsilon^{-1}\hat{\mathrm{d}}=0$$

Hessian Matrix

def:

$$\boxed{\mathrm{G}=\int_{t_0}^{T}\mathrm{H}(t')\mathrm{R}(t_0,t')dt'}$$

IOM *representer-based* inversion

$$\underbrace{\left(\mathrm{G}\mathrm{P}\mathrm{G}^T+\mathrm{C}_\varepsilon\right)}_{\hat{\mathrm{P}}}\underbrace{\left(\mathrm{P}\mathrm{G}^T\right)^{-1}\mathrm{e}_0}_{\boldsymbol{\beta}^n}=\hat{\mathrm{d}}$$

4DVAR inversion

$$\underbrace{\left(\mathrm{G}^T \mathrm{C}_\varepsilon^{-1} \mathrm{G} + \mathrm{P}^{-1}\right)}_{\hat{\mathrm{H}}} \mathrm{e}_0 - \mathrm{G}^T \mathrm{C}_\varepsilon^{-1} \hat{\mathrm{d}} = 0$$

Hessian Matrix

def:

$$\boxed{\mathrm{G} = \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0, t')dt'}$$

IOM *representer-based* inversion

$$\underbrace{\left(\mathrm{GPG}^T + \mathrm{C}_\varepsilon\right)}_{\hat{\mathrm{P}}} \underbrace{\left(\mathrm{PG}^T\right)^{-1} \mathrm{e}_0}_{\boldsymbol{\beta}^n} = \hat{\mathrm{d}}$$

***Representer* Coefficients**

Stabilized *Representer Matrix*

Representer Matrix

$$\boxed{\widehat{\mathrm{R}} \equiv \mathrm{GPG}^T}$$

4DVAR inversion

$$\underbrace{\left(\mathrm{G}^T \mathrm{C}_\varepsilon^{-1} \mathrm{G} + \mathrm{P}^{-1}\right)}_{\hat{\mathrm{H}}} \mathrm{e}_0 - \mathrm{G}^T \mathrm{C}_\varepsilon^{-1} \hat{\mathrm{d}} = 0$$

Hessian Matrix

def:

$$\boxed{\mathrm{G} = \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0, t')\,dt'}$$

IOM *representer-based* inversion

$$\underbrace{\left(\mathrm{G}\mathrm{P}\mathrm{G}^T + \mathrm{C}_\varepsilon\right)}_{\hat{\mathrm{P}}} \underbrace{\left(\mathrm{P}\mathrm{G}^T\right)^{-1} \mathrm{e}_0}_{\boldsymbol{\beta}^n} = \hat{\mathrm{d}}$$

***Representer* Coefficients**

Stabilized *Representer Matrix*

Representer Matrix

Data to Data Covariance

$$\boxed{\widehat{\mathrm{R}} \equiv \mathrm{G}\mathrm{P}\mathrm{G}^T}$$

def:

$$G = \int_{t_0}^{T} H(t')R(t_0,t')dt'$$

How to understand the physical meaning of the ***Representer Matrix***?

IOM *representer-based* inversion

$$\underbrace{\left(GPG^T + C_\varepsilon\right)}_{\hat{P}}\underbrace{\left(PG^T\right)^{-1} e_0}_{\boldsymbol{\beta}^n} = \hat{d}$$

***Representer* Coefficients**

Stabilized *Representer Matrix*

Representer Matrix

$$\boxed{\hat{R} \equiv GPG^T}$$

Representer Matrix

$$\widehat{\mathrm{R}} \equiv \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0,t')dt'\,\mathrm{P}\int_{t_0}^{T} \mathrm{R}^{T}(t',t_0)\mathrm{H}^{T}(t')dt'$$

TL-ROMS **AD-ROMS**

Representer Matrix

$$\widehat{\mathrm{R}} \equiv \mathrm{G}\mathrm{P}\mathrm{G}^{T}$$

Representer Matrix

$$\boxed{\widehat{\mathrm{R}} \equiv \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0, t')dt'\,\mathrm{P} \int_{t_0}^{T} \mathrm{R}^T(t', t_0)\mathrm{H}^T(t')dt'}$$

Assume a special assimilation case:

$$\mathrm{H}(t) = \mathrm{I}\delta(t - T)$$

→ *Observations = Full model state at time T*

$$\mathrm{P} = \left\langle \mathrm{s}_0 \mathrm{s}_0{}^T \right\rangle$$

→ *Diagonal Covariance with unit variance*

Representer Matrix

$$\boxed{\widehat{\mathrm{R}} \equiv \int_{t_0}^{T} \mathrm{I}\delta(t'-T)\mathrm{R}(t_0,t')dt'\left\langle \mathrm{s}_0\mathrm{s}_0{}^{T} \right\rangle \int_{t_0}^{T} \mathrm{R}^{T}(t',t_0)\mathrm{I}\delta(t'-T)dt'}$$

Assume a special assimilation case:

$$\mathrm{H}(t) = \mathrm{I}\delta(t-T)$$

→ *Observations = Full model state at time T*

$$\mathrm{P} = \left\langle \mathrm{s}_0\mathrm{s}_0{}^{T} \right\rangle$$

→ *Diagonal Covariance with unit variance*

Representer Matrix

$$\boxed{\widehat{\mathrm{R}} \equiv \mathrm{R}(t_0, T)\left\langle \mathbf{s}_0 \mathbf{s}_0{}^T \right\rangle \mathrm{R}^T(T, t_0)}$$

Representer Matrix

$$\widehat{\mathrm{R}} \equiv \mathrm{R}(t_0, T)\left\langle \mathrm{s}_0 \mathrm{s}_0{}^T \right\rangle \mathrm{R}^T(T, t_0)$$

Assume you want to compute the ***model spatial covariance*** *at time T*

$$\left\langle \mathrm{s}(T)\mathrm{s}^T(T) \right\rangle$$

Representer Matrix

$$\widehat{\mathrm{R}} \equiv \mathrm{R}(t_0, T)\left\langle \mathrm{s}_0 \mathrm{s}_0{}^T \right\rangle \mathrm{R}^T(T, t_0)$$

Assume you want to compute the ***model spatial covariance*** *at time T*

$$\left\langle \mathrm{s}(T)\mathrm{s}^T(T) \right\rangle$$

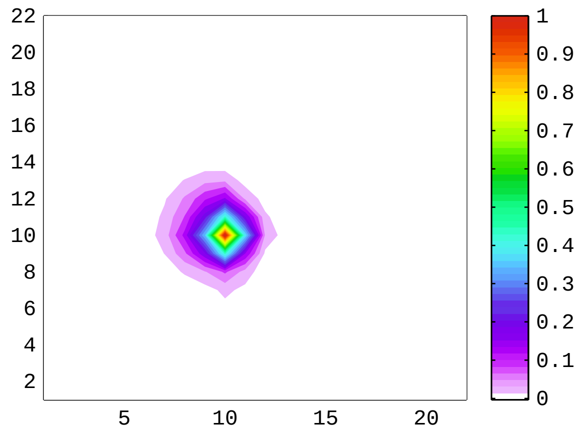
$$\mathrm{s}(T) = \mathrm{R}(T, t_0)\mathrm{s}_0$$

$$\begin{aligned}\left\langle \mathrm{s}(T)\mathrm{s}^T(T) \right\rangle &= \left\langle \left(\mathrm{R}(t_0, T)\mathrm{s}_0\right)\left(\mathrm{R}(t_0, T)\mathrm{s}_0\right)^T \right\rangle \\ &= \mathrm{R}(t_0, T)\left\langle \mathrm{s}_0 \mathrm{s}_0{}^T \right\rangle \mathrm{R}(T, t_0)\end{aligned}$$

Representer Matrix

$$\widehat{\mathrm{R}} \equiv \mathrm{R}(t_0, T)\left\langle \mathrm{s}_0 \mathrm{s}_0{}^{T} \right\rangle \mathrm{R}^{T}(T, t_0) \; = \left\langle \mathrm{s}(T)\mathrm{s}^{T}(T) \right\rangle$$

model *to* ***model*** *covariance*



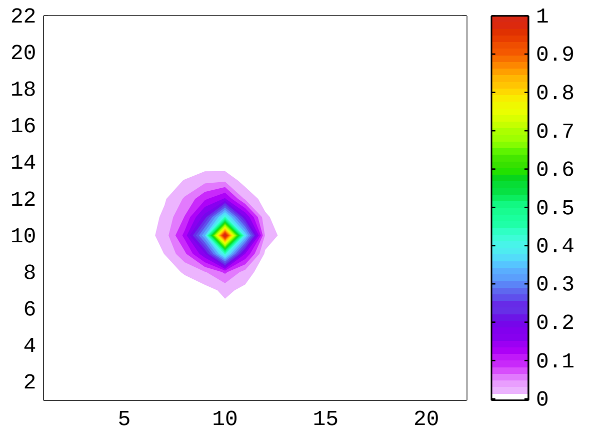
Temperature Temperature Covariance

for grid point n

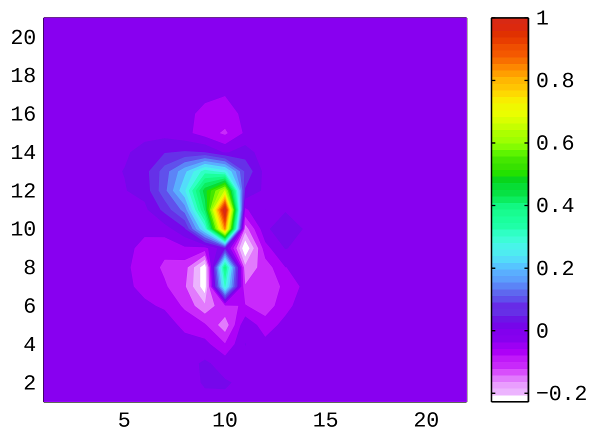
Representer Matrix

$$\widehat{\mathrm{R}} \equiv \mathrm{R}(t_0,T)\left\langle \mathrm{s}_0\mathrm{s}_0^{\;T}\right\rangle \mathrm{R}^T(T,t_0) = \left\langle \mathrm{s}(T)\mathrm{s}^T(T)\right\rangle$$

model *to* ***model*** *covariance*



Temperature Temperature Covariance
for grid point n



Temperature Velocity Covariance
for grid point n

Representer Matrix

$$\widehat{\mathrm{R}} \equiv \mathrm{R}(t_0, T)\left\langle \mathrm{s}_0 \mathrm{s}_0{}^T \right\rangle \mathrm{R}^T(T, t_0) = \left\langle \mathrm{s}(T)\mathrm{s}^T(T) \right\rangle$$

model *to* ***model*** *covariance*

$$\widehat{\mathrm{R}}(t', t'') \equiv \mathrm{R}(t_0, t')\left\langle \mathrm{s}_0 \mathrm{s}_0{}^T \right\rangle \mathrm{R}^T(t'', t_0) = \left\langle \mathrm{s}(t')\mathrm{s}^T(t'') \right\rangle$$

model *to* ***model*** *covariance most general form*

Representer Matrix

$$\widehat{\mathrm{R}} \equiv \mathrm{R}(t_0, T)\left\langle \mathrm{s}_0 \mathrm{s}_0{}^T \right\rangle \mathrm{R}^T(T, t_0) \;\; = \left\langle \mathrm{s}(T)\mathrm{s}^T(T) \right\rangle$$

***model** to **model** covariance*

$$\widehat{\mathrm{R}}(t', t'') \equiv \mathrm{R}(t_0, t')\left\langle \mathrm{s}_0 \mathrm{s}_0{}^T \right\rangle \mathrm{R}^T(t'', t_0) = \left\langle \mathrm{s}(t')\mathrm{s}^T(t'') \right\rangle$$

***model** to **model** covariance most general form*

if we sample at observation locations through $\int_{t_0}^{T} \mathrm{H}(t')(\quad)\,dt'$

$$\widehat{\mathrm{R}} \equiv \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0, t')\,dt'\,\mathrm{P} \int_{t_0}^{T} \mathrm{R}^T(t', t_0)\mathrm{H}^T(t')\,dt' \;\; = \left\langle \hat{\mathrm{d}}\hat{\mathrm{d}}^T \right\rangle$$

***data** to **data** covariance*

… back to the system to invert ….

STRONG CONSTRAINT

4DVAR inversion

def:

$$\boxed{\mathrm{G} = \int_{t_0}^{T} \mathrm{H}(t')\mathrm{R}(t_0,t')dt'}$$

$$\underbrace{\left(\mathrm{G}^T\mathrm{C}_\varepsilon^{-1}\mathrm{G} + \mathrm{P}^{-1}\right)}_{\hat{\mathrm{H}}}\mathrm{e}_0 - \mathrm{G}^T\mathrm{C}_\varepsilon^{-1}\hat{\mathrm{d}} = 0$$

Hessian Matrix

IOM *representer-based* inversion

$$\underbrace{\left(\mathrm{GPG}^T + \mathrm{C}_\varepsilon\right)}_{\hat{\mathrm{P}}}\underbrace{\left(\mathrm{PG}^T\right)^{-1}\mathrm{e}_0}_{\boldsymbol{\beta}^n} = \hat{\mathrm{d}}$$

***Representer* Coefficients**

Stabilized *Representer Matrix*

Representer Matrix

$$\boxed{\widehat{\mathrm{R}} \equiv \mathrm{GPG}^T}$$

WEAK CONSTRAINT

4DVAR inversion

$$\int_{t_0}^{T} \underbrace{\left[\mathrm{G}^T(t)\mathrm{C}_\varepsilon^{-1}\mathrm{G}(t') + \mathrm{C}^{-1}(t,t') \right]}_{\hat{\mathrm{H}}(t,t')} \mathrm{e}(t')\,dt' - \mathrm{G}^T(t)\mathrm{C}_\varepsilon^{-1}\hat{\mathrm{d}} = 0$$

def:

$$\boxed{\mathrm{G}(t) = \int_{t}^{T} \mathrm{H}(t')\mathrm{R}(t,t')\,dt'}$$

Hessian Matrix

IOM *representer-based* inversion

$$\underbrace{\int_{t_0}^{T}\int_{t_0}^{T} \left[\mathrm{G}(t')\mathrm{C}(t',t'')\mathrm{G}^T(t'') + \mathrm{C}_\varepsilon \right] dt'\,dt''}_{\hat{\mathrm{P}}} \underbrace{\int_{t_0}^{T}\int_{t_0}^{T} \left[\mathrm{C}(t',t'')\mathrm{G}^T(t'') \right]^{-1} \mathrm{e}(t')\,dt'\,dt''}_{\boldsymbol{\beta}^n} = \hat{\mathrm{d}}$$

***Representer* Coefficients**

Stabilized *Representer Matrix*

Representer Matrix

$$\boxed{\widehat{\mathrm{R}} \equiv \int_{t_0}^{T}\int_{t_0}^{T} \left[\mathrm{G}(t')\mathrm{C}(t',t'')\mathrm{G}^T(t'') + \mathrm{C}_\varepsilon \right] dt'\,dt''}$$

WEAK CONSTRAINT

How to solve for corrections $\mathrm{e}(t)$?

→ Method of solution in ***IROMS***

IOM *representer-based* inversion

$$\underbrace{\int_{t_0}^{T}\int_{t_0}^{T}\left[\mathrm{G}(t')\mathrm{C}(t',t'')\mathrm{G}^T(t'')+\mathrm{C}_\varepsilon\right]dt'dt''}_{\hat{\mathrm{P}}}\ \underbrace{\int_{t_0}^{T}\int_{t_0}^{T}\left[\mathrm{C}(t',t'')\mathrm{G}^T(t'')\right]^{-1}\mathrm{e}(t')dt'dt''}_{\boldsymbol{\beta}^n}=\hat{\mathrm{d}}$$

Stabilized *Representer Matrix*

***Representer* Coefficients**

$$\mathrm{e}(t)=\int_{t_0}^{t}\mathrm{C}(t,t'')\int_{t''}^{T}\mathrm{R}^T(t',t'')\mathrm{H}^T(t')dt'\ \boldsymbol{\beta}^n dt''$$

WEAK CONSTRAINT

How to solve for corrections $\mathrm{e}(t)$?

→ Method of solution in ***IROMS***

IOM *representer-based* inversion

$$\hat{\mathrm{P}} \quad \mathrm{x} \quad \boldsymbol{\beta}^{n} = \hat{\mathrm{d}}$$

***Representer* Coefficients**

Stabilized *Representer Matrix*

$$\mathrm{e}(t) = \int_{t_0}^{t} \mathrm{C}(t,t'') \int_{t''}^{T} \mathrm{R}^{T}(t',t'')\mathrm{H}^{T}(t')dt' \ \boldsymbol{\beta}^{n} dt''$$

Method of solution in *IROMS*

STEP 1) **Produce background state using nonlinear model starting from initial guess.**

$$\mathrm{u}_B(t) = \mathrm{N}[\hat{\mathrm{u}}_0, \mathrm{F}(t), t_0, t]$$

STEP 2) **Run REP-ROMS linearized around background state to generate first estimate of model trajectory**

$$\mathrm{u}_F^0(t) = \mathrm{R}(t_0,t)\hat{\mathrm{u}}_0 + \int_{t_0}^{t} \mathrm{R}(t',t)\left[N(\mathrm{u}_B) + \mathrm{F}(t')\right]dt'$$

$$\hat{\mathrm{u}}^0(t) = \mathrm{u}_F^0(t)$$

$\mathrm{do} \quad n = 0 \rightarrow N$

outer loop

STEP 3) **Compute model-data misfit**

$$\hat{\mathrm{d}} = \mathrm{d} - \int_{t_0}^{T} \mathrm{H}(t')\hat{\mathrm{u}}^n(t)dt'$$

STEP 4) **Solve for Representer Coeficients**

$$\hat{\mathrm{P}}\boldsymbol{\beta}^n = \hat{\mathrm{d}}$$

STEP 5) **Compute corrections**

$$\mathrm{e}(t) = \int_{t_0}^{t} \mathrm{C}(t,t'')\int_{t''}^{T} \mathrm{R}^T(t',t'')\mathrm{H}^T(t')dt' \,\boldsymbol{\beta}^n dt''$$

STEP 6) **Update model state using REP-ROMS**

$$\hat{\mathrm{u}}^n(t) = \mathrm{R}(t_0,t)\hat{\mathrm{u}}_0 + \int_{t_0}^{t} \mathrm{R}(t',t)\left[N(\mathrm{u}_B) + \mathrm{F}(t')\right]dt' + \int_{t_0}^{t} \mathrm{R}(t',t)\mathrm{e}(t')dt'$$

Method of solution in *IROMS*

STEP 1) **Produce background state using nonlinear model starting from initial guess.**

$$\mathrm{u}_B(t) = \mathrm{N}[\hat{\mathrm{u}}_0, \mathrm{F}(t), t_0, t]$$

STEP 2) **Run REP-ROMS linearized around background state to generate first estimate of model trajectory**

$$\mathrm{u}_F^0(t) = \mathrm{R}(t_0,t)\hat{\mathrm{u}}_0 + \int_{t_0}^{t} \mathrm{R}(t',t)\left[N(\mathrm{u}_B) + \mathrm{F}(t')\right]dt'$$

$$\hat{\mathrm{u}}^0(t) = \mathrm{u}_F^0(t)$$

$\mathrm{do} \quad n = 0 \rightarrow N$

outer loop

STEP 3) **Compute model-data misfit**

$$\hat{\mathrm{d}} = \mathrm{d} - \int_{t_0}^{T} \mathrm{H}(t')\hat{\mathrm{u}}^n(t)dt'$$

inner loop

STEP 4) **Solve for Representer Coeficients**

$$\hat{\mathrm{P}}\boldsymbol{\beta}^n = \hat{\mathrm{d}}$$

STEP 5) **Compute corrections**

$$\mathrm{e}(t) = \int_{t_0}^{t} \mathrm{C}(t,t'') \int_{t''}^{T} \mathrm{R}^T(t',t'')\mathrm{H}^T(t')dt' \; \boldsymbol{\beta}^n dt''$$

STEP 6) **Update model state using REP-ROMS**

$$\hat{\mathrm{u}}^n(t) = \mathrm{R}(t_0,t)\hat{\mathrm{u}}_0 + \int_{t_0}^{t} \mathrm{R}(t',t)\left[N(\mathrm{u}_B) + \mathrm{F}(t')\right]dt' + \int_{t_0}^{t} \mathrm{R}(t',t)\mathrm{e}(t')dt'$$

How to evaluate the action of the stabilized ***Representer Matrix*** $\widehat{\mathrm{P}}$

$$\widehat{\mathrm{P}}\boldsymbol{\beta} = \int_{t_0}^{T} \mathrm{H}(\hat{t}) \int_{t_0}^{\hat{t}} \mathrm{R}(t',\hat{t}) \int_{t_0}^{T} \mathrm{C}(t',t'') \int_{t''}^{T} \mathrm{R}^T(\widehat{t},t'')\mathrm{H}^T(\widehat{t})\boldsymbol{\beta}\, d\widehat{t}\, dt''\, dt'\, d\hat{t} + \mathrm{C}_{\boldsymbol{\varepsilon}}\boldsymbol{\beta}$$

How to evaluate the action of the stabilized ***Representer Matrix*** $\widehat{\mathrm{P}}$

$$\widehat{\mathrm{P}}\boldsymbol{\beta} = \int_{t_0}^{T} \mathrm{H}(\hat{t}) \int_{t_0}^{\hat{t}} \mathrm{R}(t',\hat{t}) \int_{t_0}^{T} \mathrm{C}(t',t'') \underbrace{\int_{t''}^{T} \mathrm{R}^T(\widehat{t},t'')\mathrm{H}^T(\widehat{t})\boldsymbol{\beta}\, d\widehat{t}}_{\boldsymbol{\lambda}(t'')\ \text{Adjoint Solution}}\, dt''\, dt'\, d\hat{t} + \mathrm{C}_{\boldsymbol{\varepsilon}}\boldsymbol{\beta}$$

How to evaluate the action of the stabilized ***Representer Matrix*** $\widehat{\mathrm{P}}$

$$\widehat{\mathrm{P}}\boldsymbol{\beta} = \int_{t_0}^{T} \mathrm{H}(\hat{t}) \int_{t_0}^{\hat{t}} \mathrm{R}(t',\hat{t}) \underbrace{\int_{t_0}^{T} \mathrm{C}(t',t'') \underbrace{\int_{t''}^{T} \mathrm{R}^T(\widehat{t},t'')\mathrm{H}^T(\widehat{t})\boldsymbol{\beta}\, d\widehat{t}}_{\boldsymbol{\lambda}(t'')\ \text{Adjoint Solution}}\, dt''}_{\mathrm{f}(t')\ \text{Convolution of Adjoint Solution}}\, dt'\, d\hat{t} + \mathrm{C}_{\boldsymbol{\varepsilon}}\boldsymbol{\beta}$$

How to evaluate the action of the stabilized ***Representer Matrix*** $\widehat{\mathrm{P}}$

$$\widehat{\mathrm{P}}\boldsymbol{\beta} = \int_{t_0}^{T} \mathrm{H}(\hat{t}) \underbrace{\int_{t_0}^{\hat{t}} \mathrm{R}(t',\hat{t}) \underbrace{\int_{t_0}^{T} \mathrm{C}(t',t'') \underbrace{\int_{t''}^{T} \mathrm{R}^T(\overset{\frown}{t},t'')\mathrm{H}^T(\overset{\frown}{t})\boldsymbol{\beta}\, d\overset{\frown}{t}}_{\boldsymbol{\lambda}(t'')\ \text{Adjoint Solution}}\, dt''}_{\mathrm{f}(t')\ \text{Convolution of Adjoint Solution}}\, dt'}_{\boldsymbol{\tau}(\hat{t})\ \text{Tangent Linear model forced with } \mathrm{f}(t')}\, d\hat{t} + \mathrm{C}_{\boldsymbol{\varepsilon}}\boldsymbol{\beta}$$

How to evaluate the action of the stabilized ***Representer Matrix*** $\hat{\mathrm{P}}$

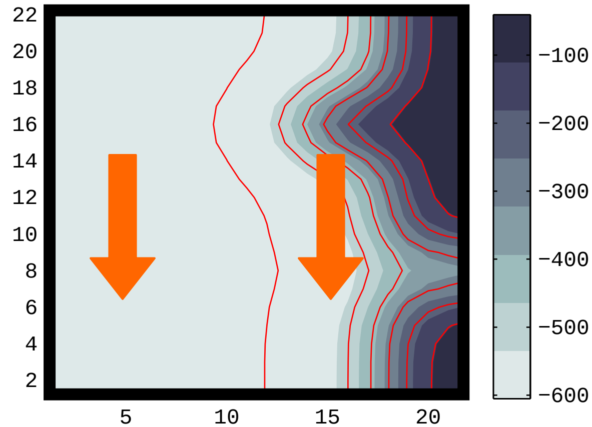
$$\hat{\mathrm{P}}\boldsymbol{\beta} = \underbrace{\int_{t_0}^{T} \mathrm{H}(\hat{t}) \underbrace{\int_{t_0}^{\hat{t}} \mathrm{R}(t',\hat{t}) \underbrace{\int_{t_0}^{T} \mathrm{C}(t',t'') \underbrace{\int_{t''}^{T} \mathrm{R}^{T}(\hat{t},t'')\mathrm{H}^{T}(\hat{t})\boldsymbol{\beta}\, d\hat{t}}_{\boldsymbol{\lambda}(t'')\ \text{Adjoint Solution}}\, dt''}_{\mathrm{f}(t')\ \text{Convolution of Adjoint Solution}}\, dt'}_{\boldsymbol{\tau}(\hat{t})\ \text{Tangent Linear model forced with } \mathrm{f}(t')}\, d\hat{t}}_{\text{Sampling of Tangent Linear solution } \boldsymbol{\tau}(\hat{t})} + \mathrm{C}_{\boldsymbol{\varepsilon}}\boldsymbol{\beta}$$

ROMS Coastal Upwelling

10km res, Baroclinic, Periodic NS, CCS Topographic Slope with Canyons and Seamounts, full options!

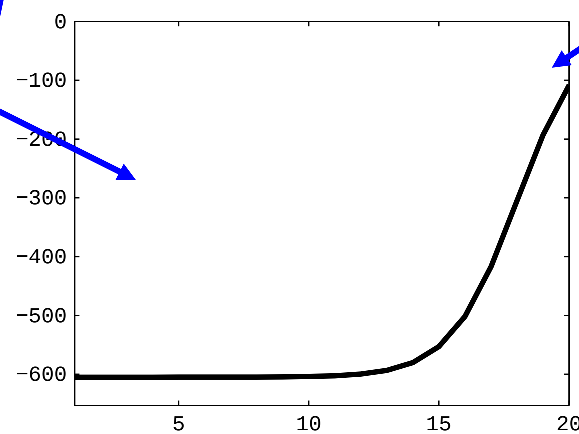
Model Topography

Seasonal Wind Stress



Open Ocean

Shelf



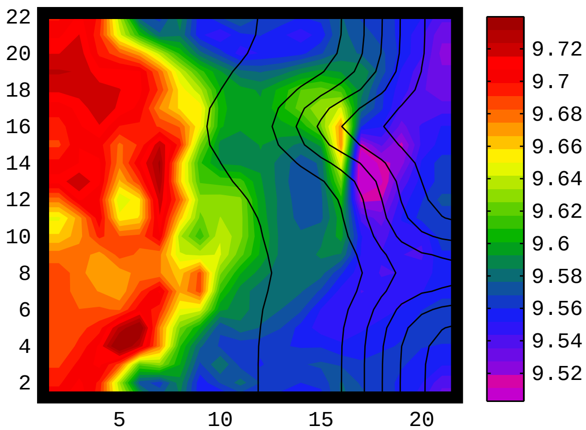
200 km

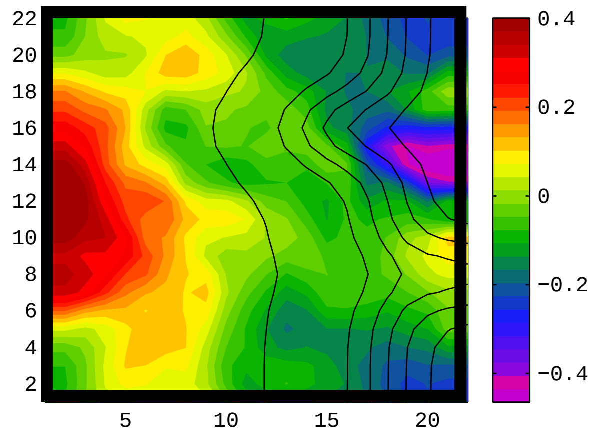
ROMS Coastal Upwelling

10km res, Baroclinic, Periodic NS, CCS Topographic Slope with Canyons and Seamounts, full options!

Temperature

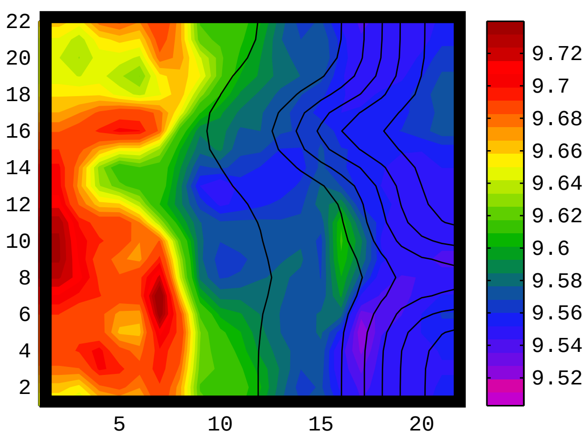
Sea Surface Heigth

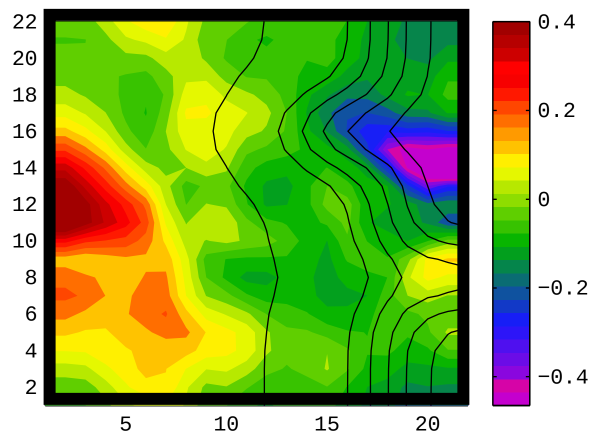




Initial Condition

t_0 **= 1 JAN '06**



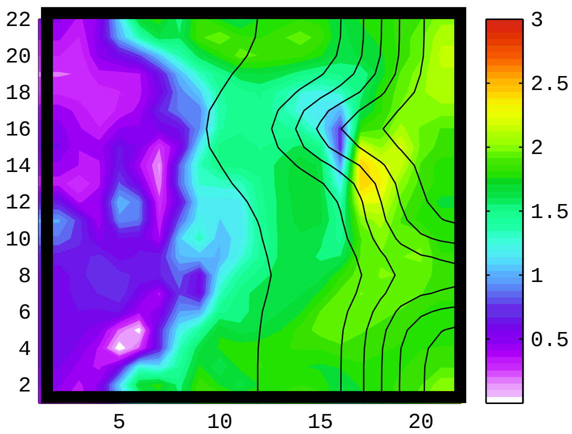


Final Time

t_N **= 4 JAN '06**

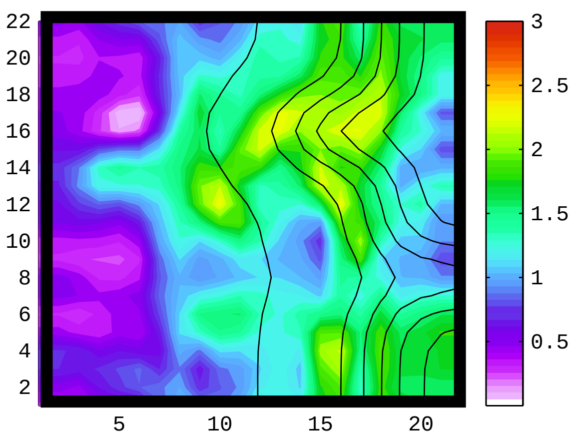
The ASSIMILATION Setup

Passive Tracer



Initial Condition

$$t_0$$



Final Time

$$t_N$$

***Problem*:** *Assimilate a passive tracer, which was perfectly measured.*

Model Dynamics are Correct →

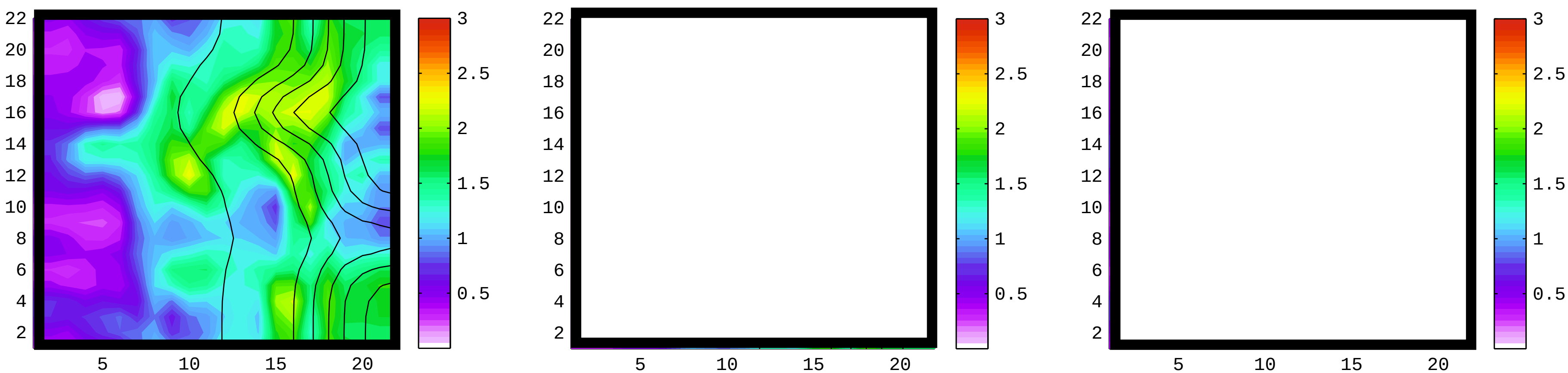
$$J = \mathrm{n}^{T} \mathrm{C}_{\boldsymbol{\varepsilon}}^{-1} \mathrm{n} + \mathrm{e}_0{}^{T} \mathrm{P}^{-1} \mathrm{e}_0$$

$$\frac{\partial T}{\partial t} + \mathrm{u} \nabla T = K_H \nabla_H^2 T + K \frac{\partial^2 T}{\partial z^2}$$

$$T(t_0) = T_0$$

$\mathrm{e}(t_0)$ → *Corrections only to initial state*

t_N **True**



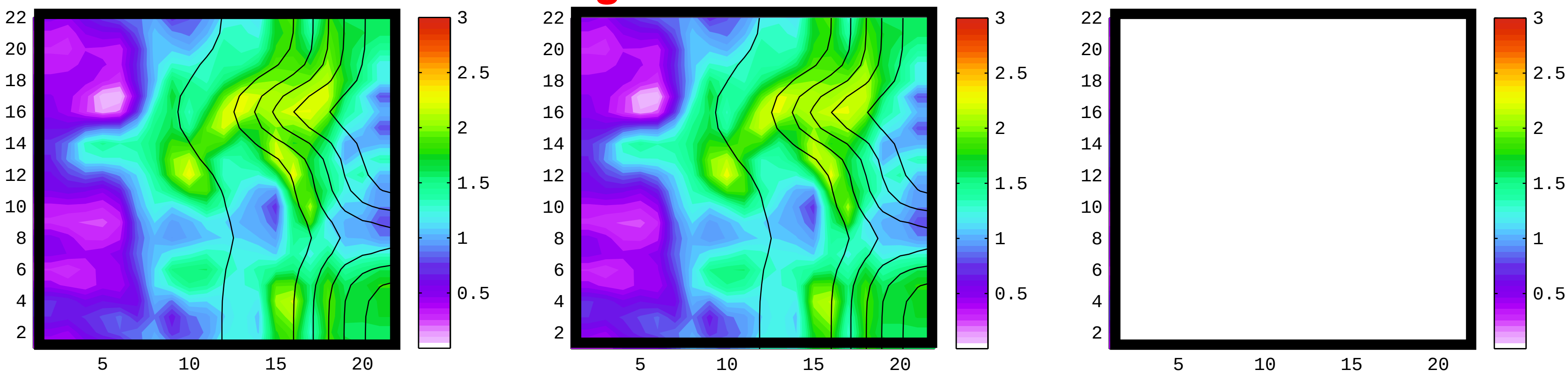
Model Dynamics are Correct

$$J = \mathrm{n}^T \mathrm{C}_{\boldsymbol{\varepsilon}}^{-1} \mathrm{n} + \mathrm{e}_0{}^T \mathrm{P}^{-1} \mathrm{e}_0$$

t_N

True

Diagonal Covariance



Model Dynamics are Correct →

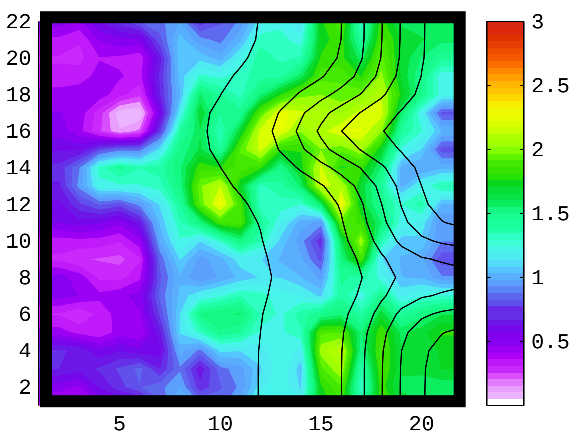
$$J = \mathrm{n}^T \mathrm{C}_{\boldsymbol{\varepsilon}}^{-1} \mathrm{n} + \mathrm{e}_0{}^T \mathrm{P}^{-1} \mathrm{e}_0$$

How about the initial condition?

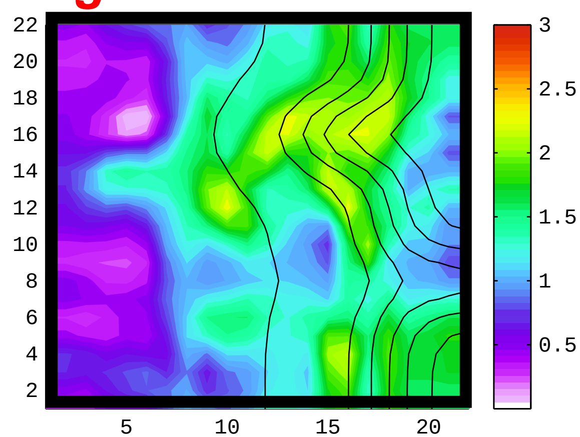
$$t_0$$

t_N

True

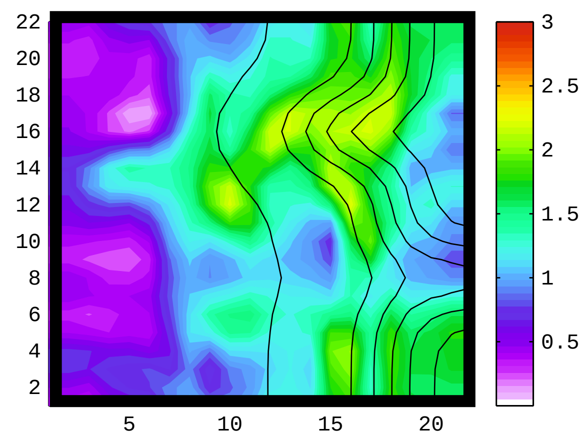


Diagonal Covariance



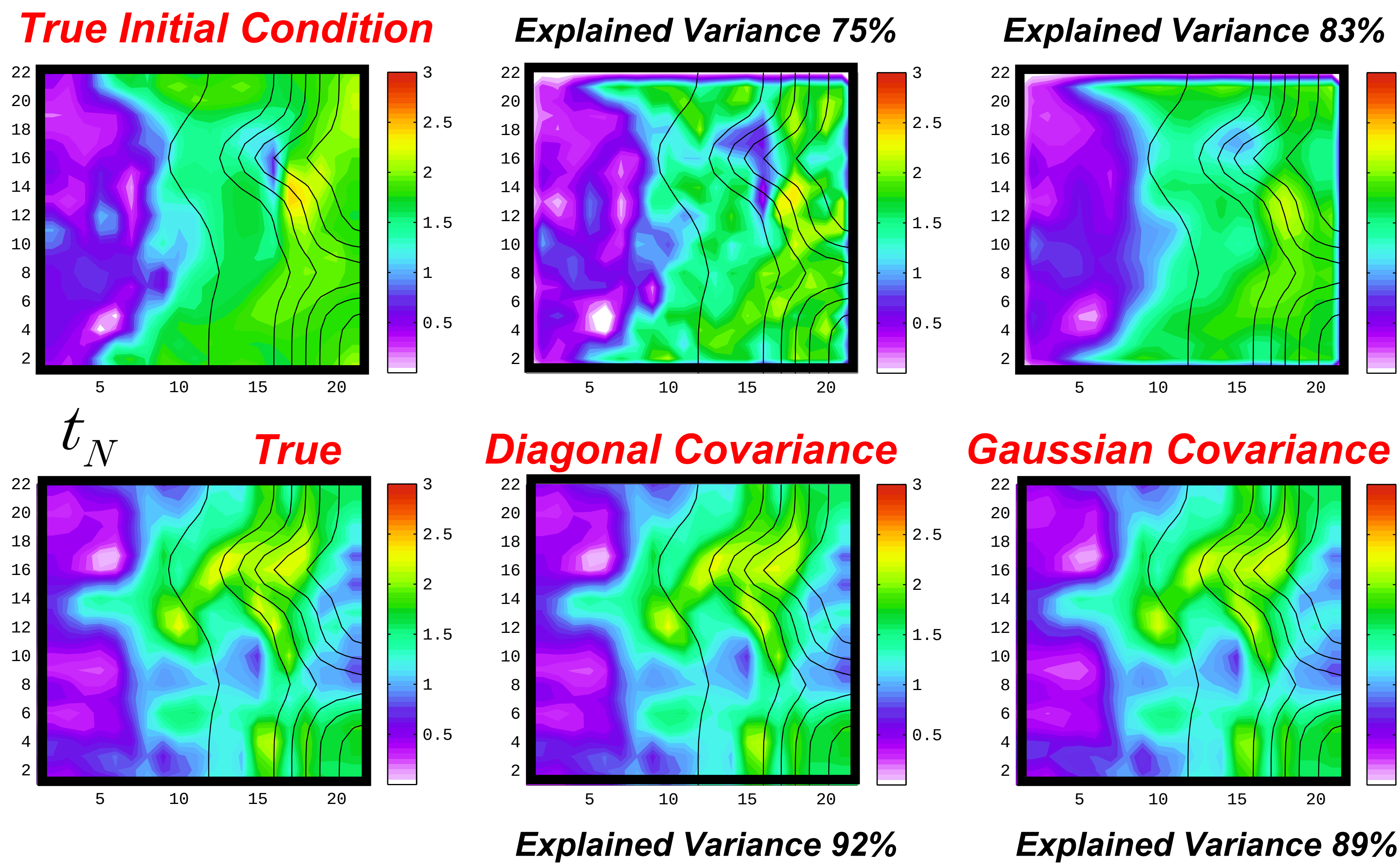
Explained Variance 92%

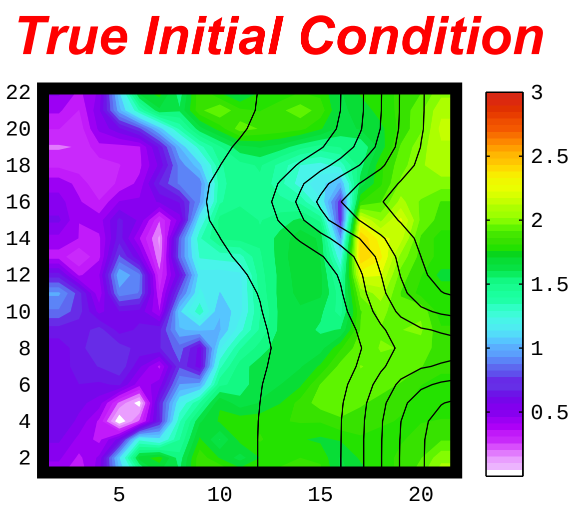
Gaussian Covariance

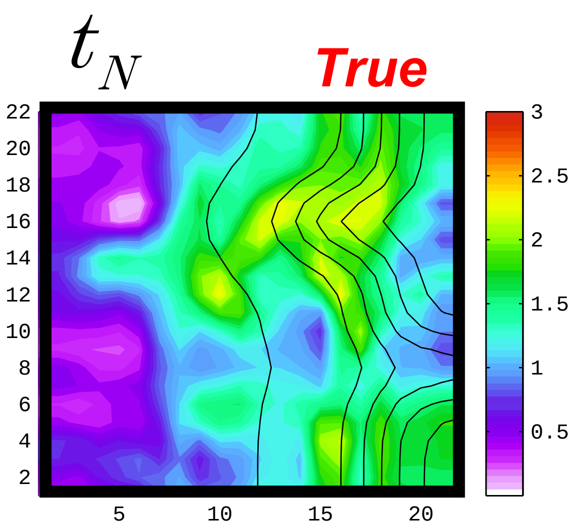


Explained Variance 89%

Model Dynamics are Correct







What if we allow for error in the model dynamics?

$$\frac{\partial T}{\partial t} + \mathrm{u}\nabla T = K_H \nabla_H^2 T + K \frac{\partial^2 T}{\partial z^2}$$

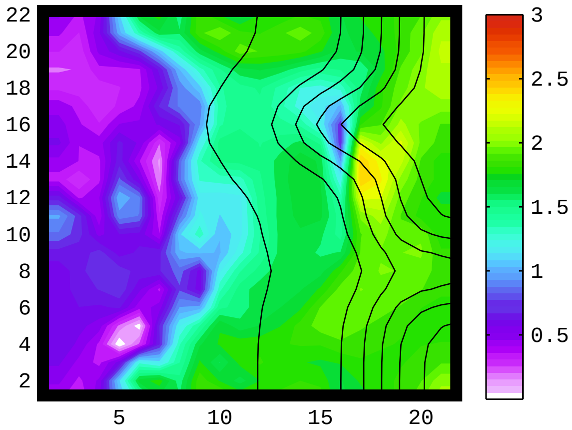
$$T(t_0) = T_0$$

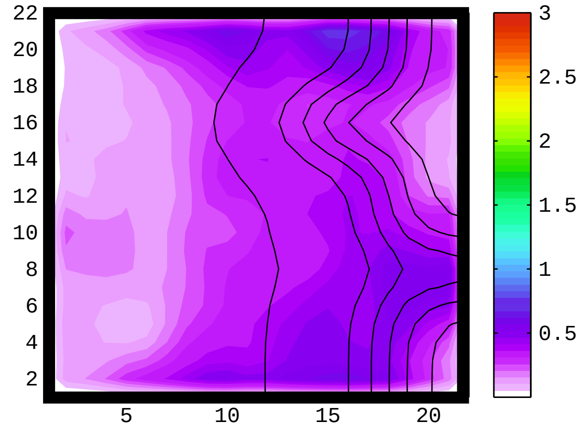
$\mathrm{e}(t)$ → *Time Dependent corrections to model dynamics and boundary conditions*

Weak Constraint

True Initial Condition

Explained Variance 24%

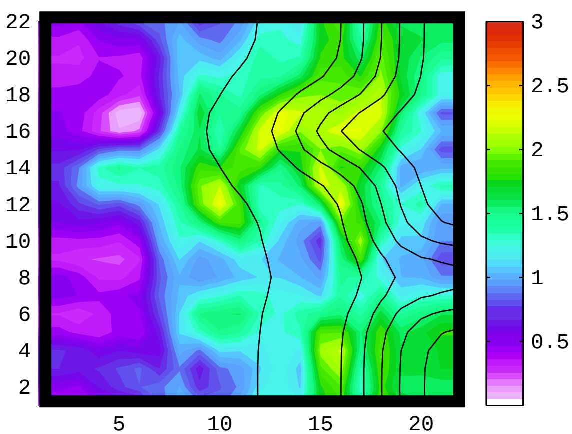


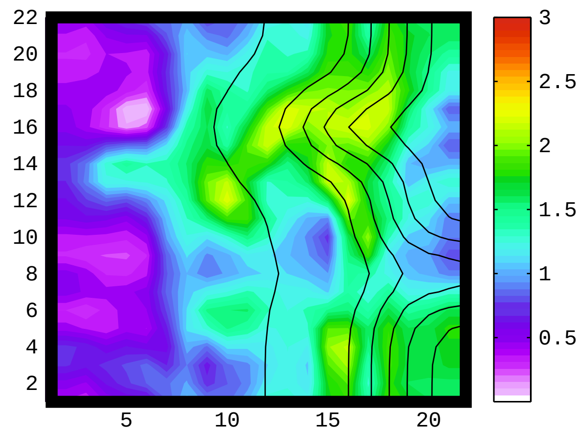


t_N

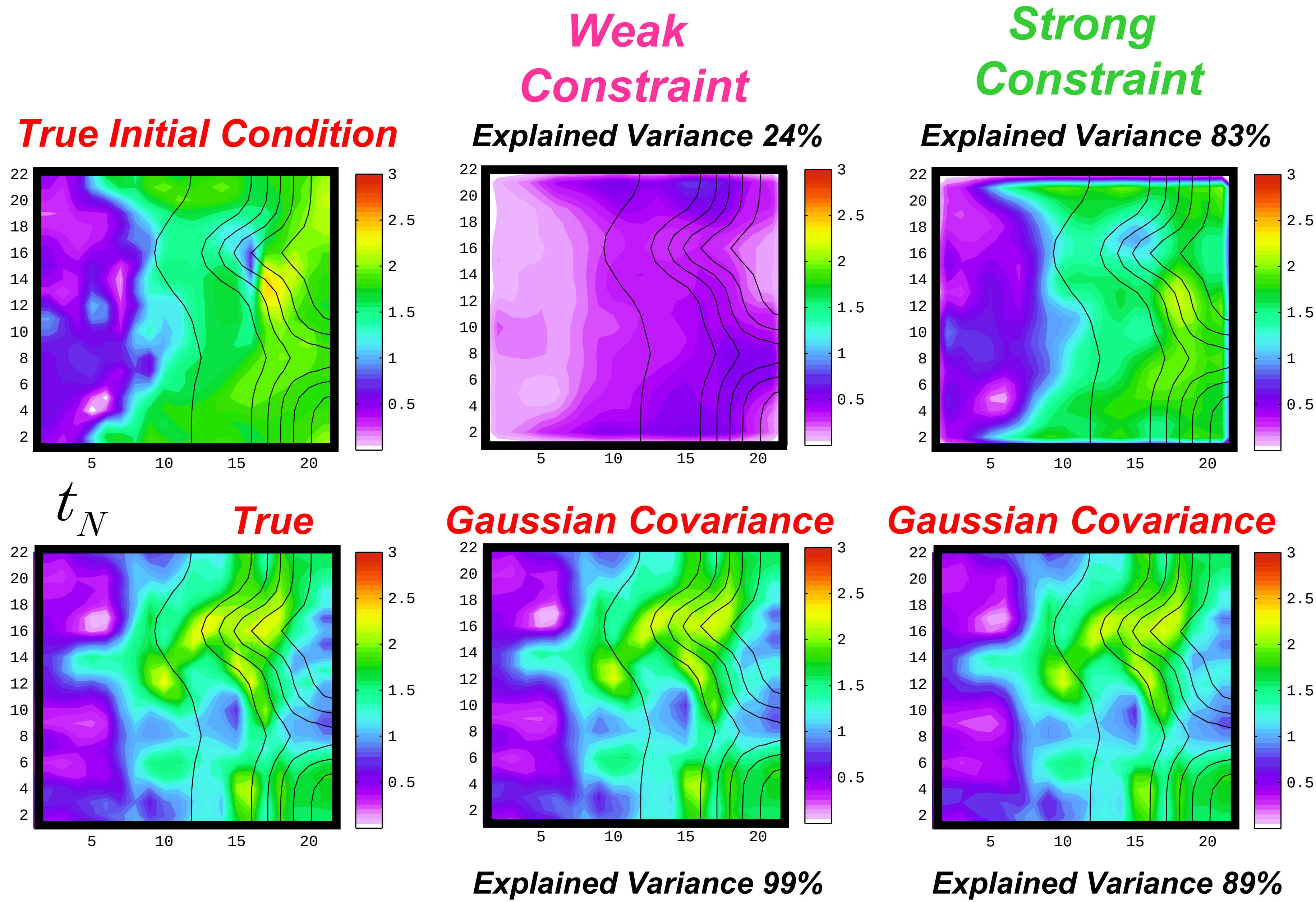
True

Gaussian Covariance

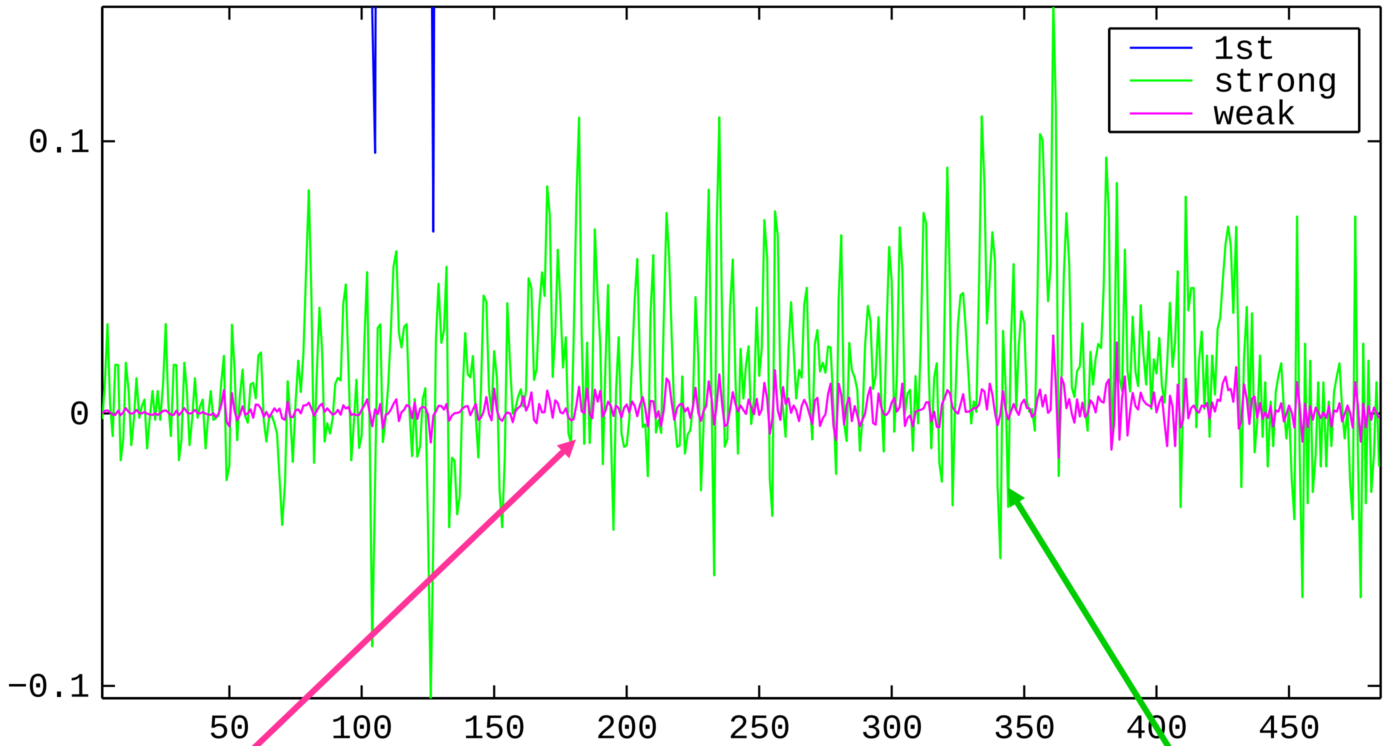




Explained Variance 99%


Weak Constraint
Strong Constraint
True Initial Condition
Explained Variance 24%
Explained Variance 83%
t_N
True
Gaussian Covariance
Gaussian Covariance
Explained Variance 99%
Explained Variance 89%

Data Misfit
(comparison)



Weak Constraint

Strong Constraint

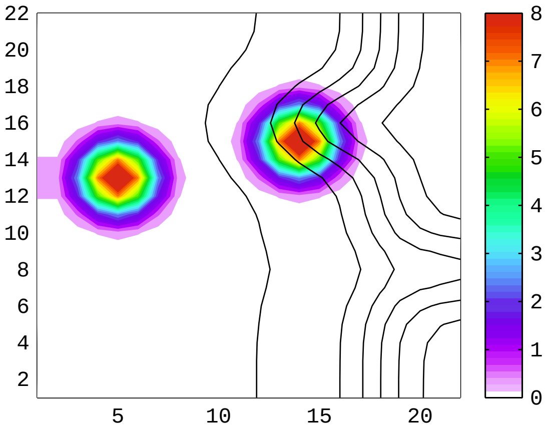
What if we really have substantial model errors, or bias?

$$\frac{\partial T}{\partial t} + \mathbf{u} \nabla T = K_H \nabla_H^2 T + K \frac{\partial^2 T}{\partial z^2}$$

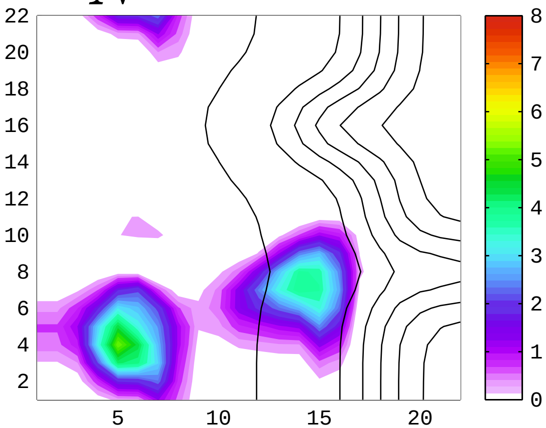
$$T(t_0) = T_0$$

Assimilation of data at time t_N

True Initial Condition

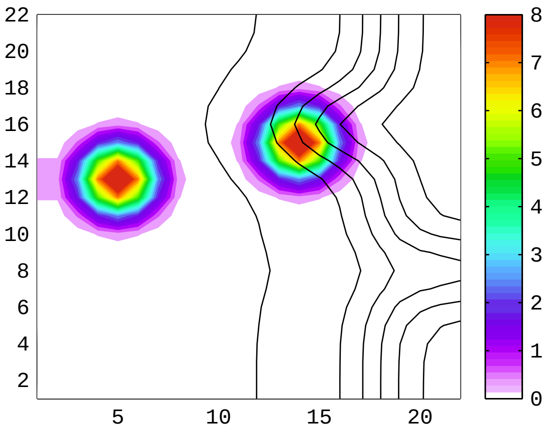


t_N **True**



Assimilation of data at time t_N

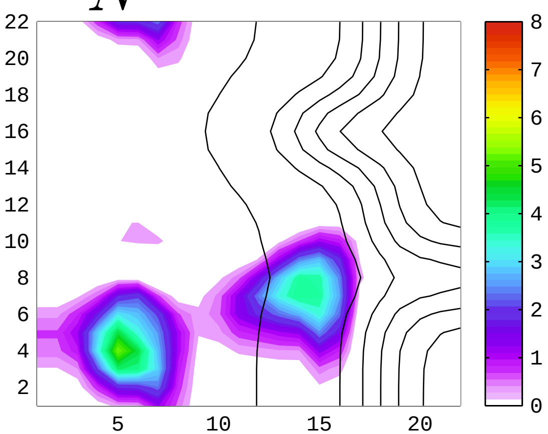
True Initial Condition



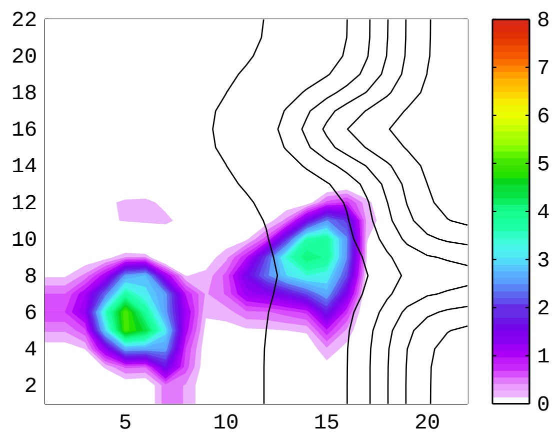
Which is the model with the correct dynamics?

t_N

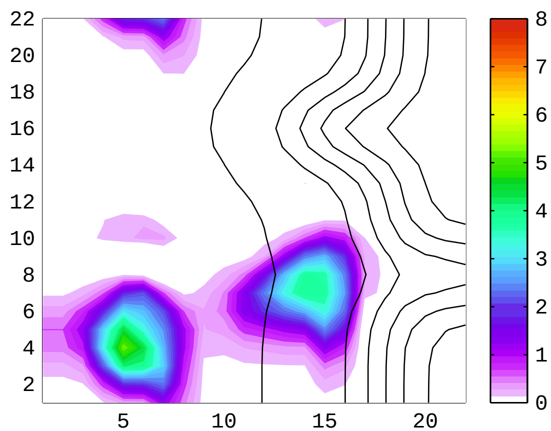
True

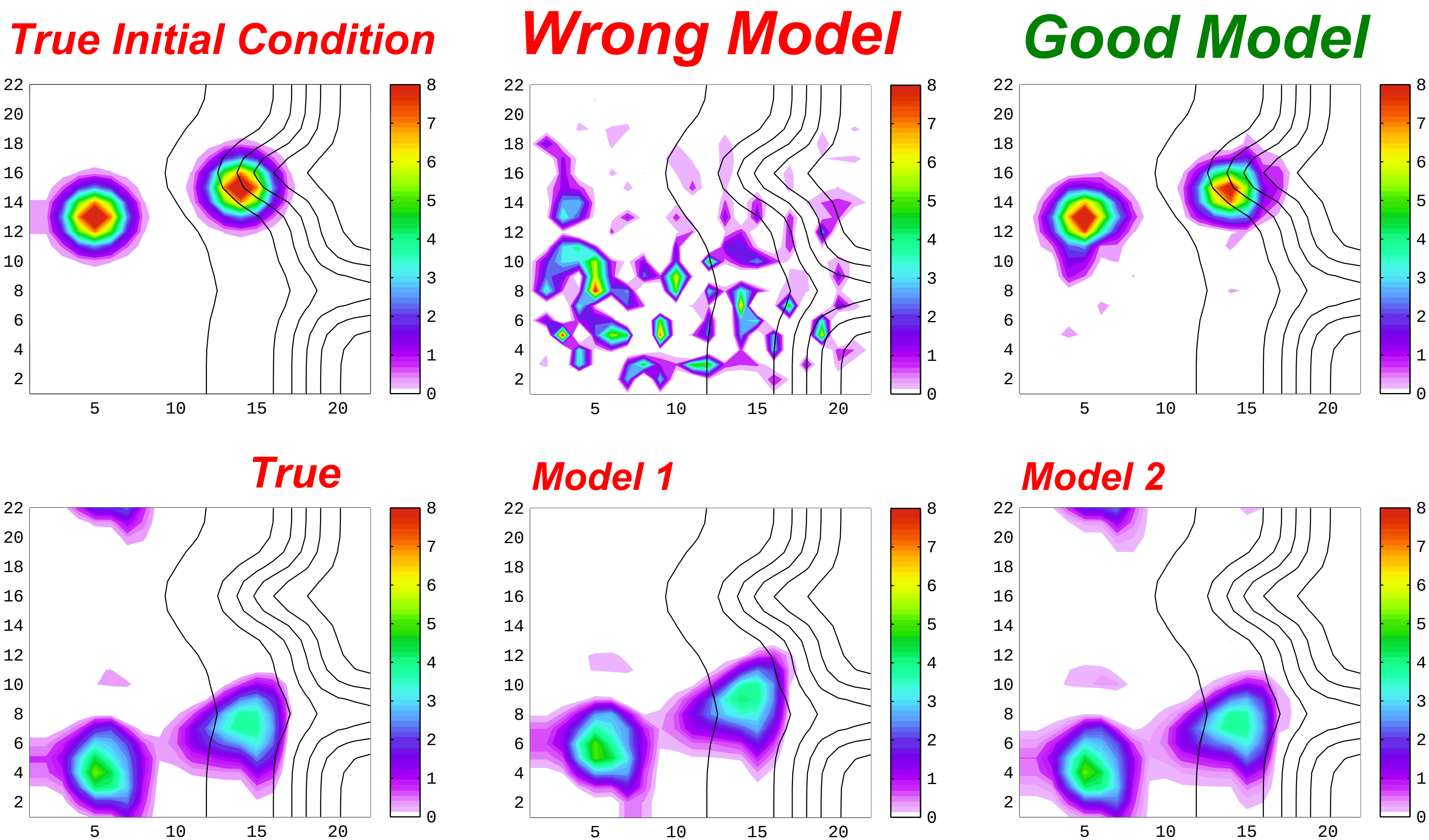


Model 1



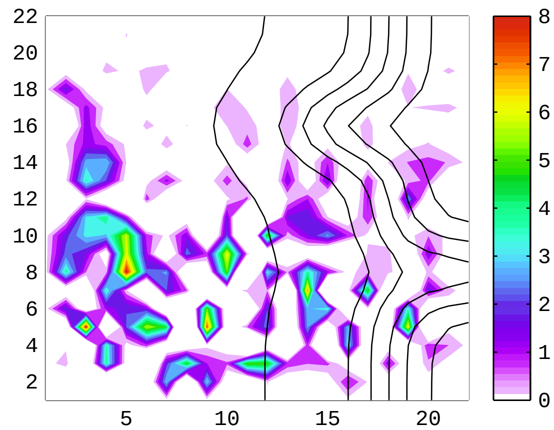
Model 2




True Initial Condition
Wrong Model
Good Model
True
Model 1
Model 2

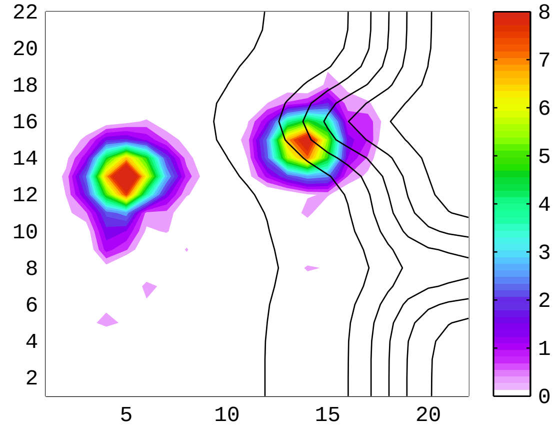
Time Evolution of solutions after assimilation

Wrong Model



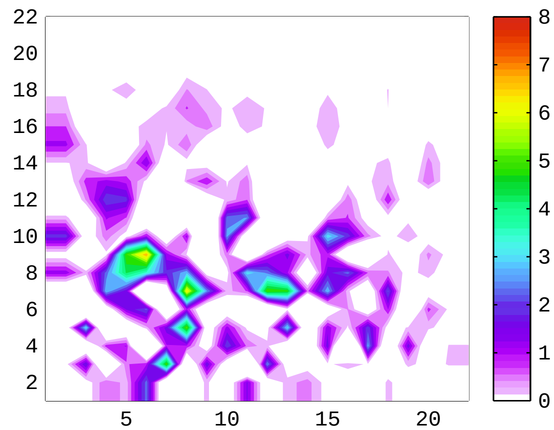
DAY 0

Good Model



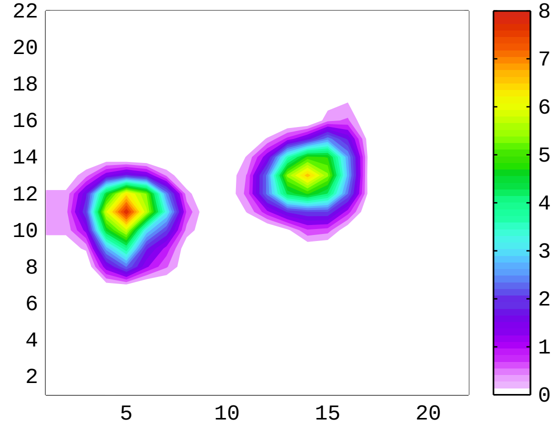
Time Evolution of solutions after assimilation

Wrong Model



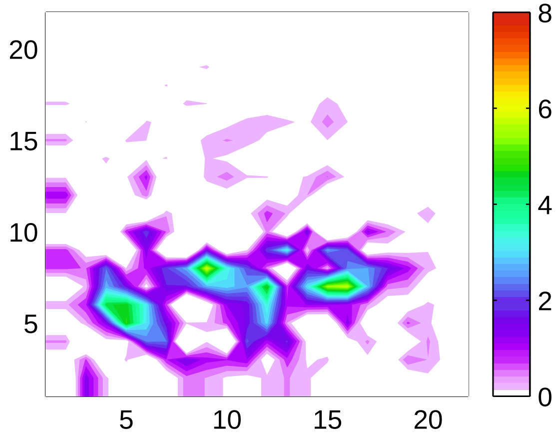
DAY 1

Good Model



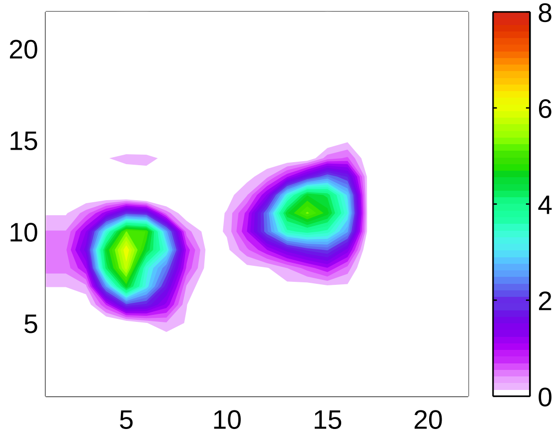
Time Evolution of solutions after assimilation

Wrong Model



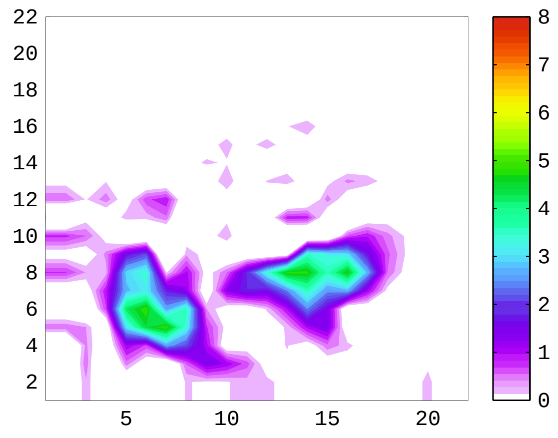
DAY 2

Good Model



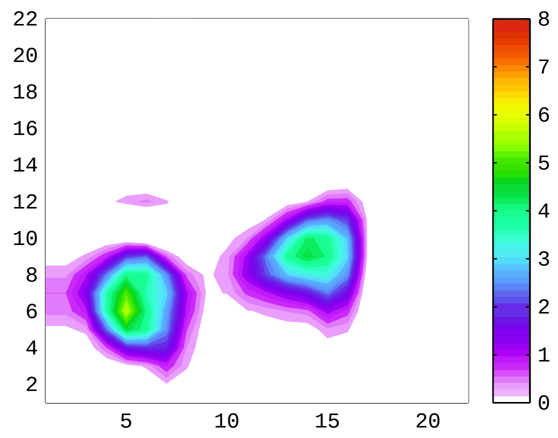
Time Evolution of solutions after assimilation

Wrong Model



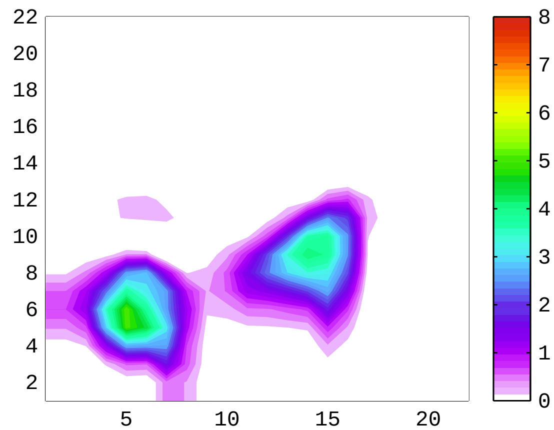
DAY 3

Good Model



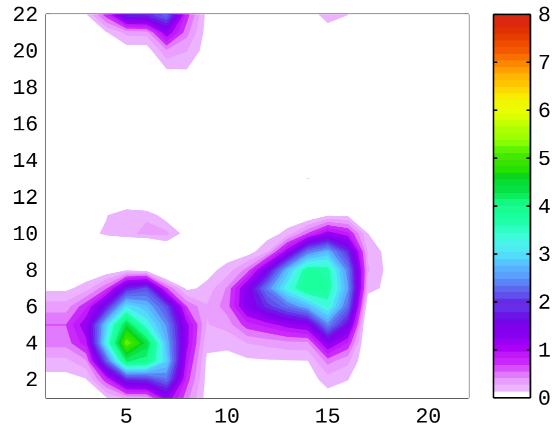
Time Evolution of solutions after assimilation

Wrong Model



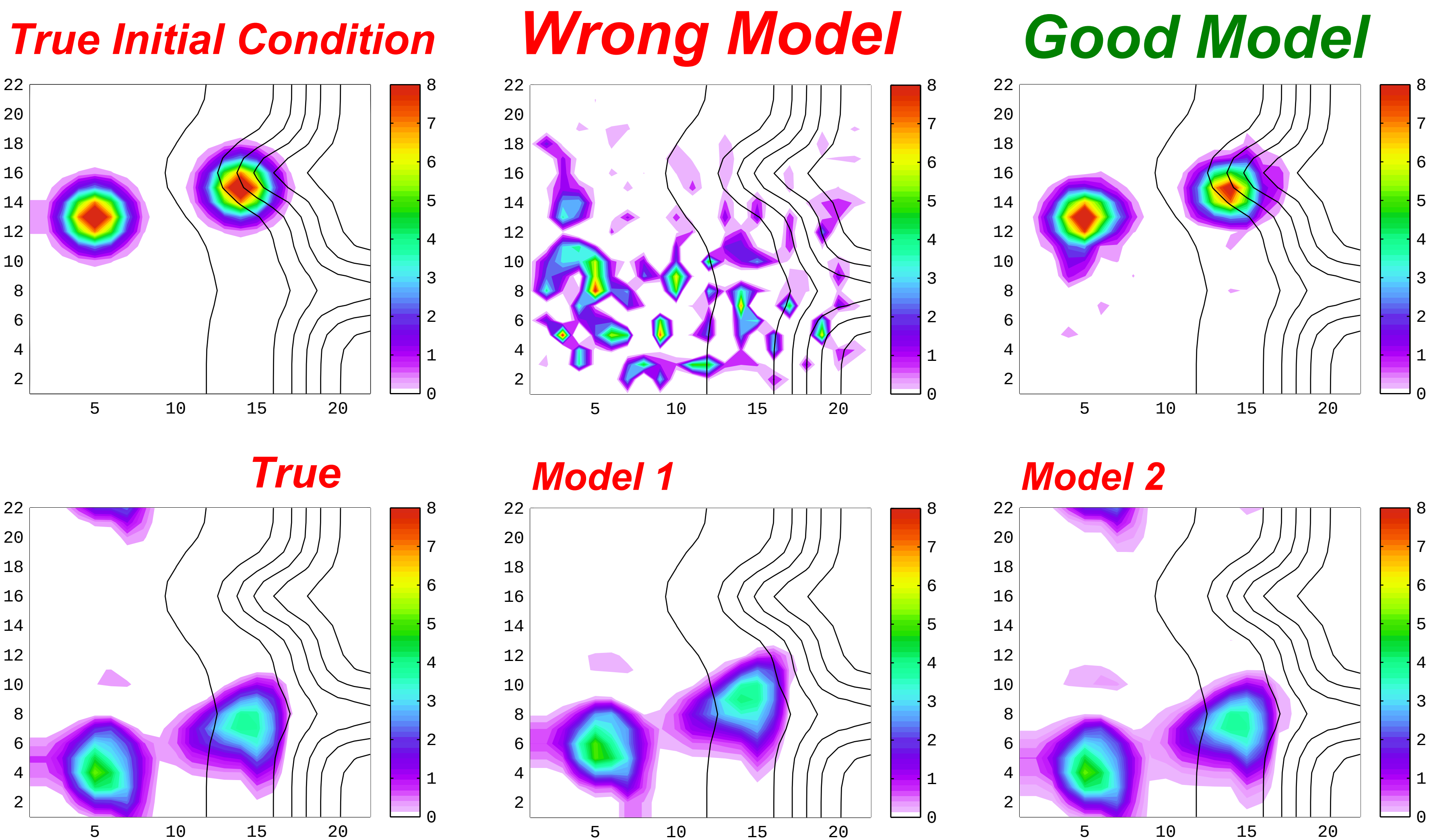
DAY 4

Good Model

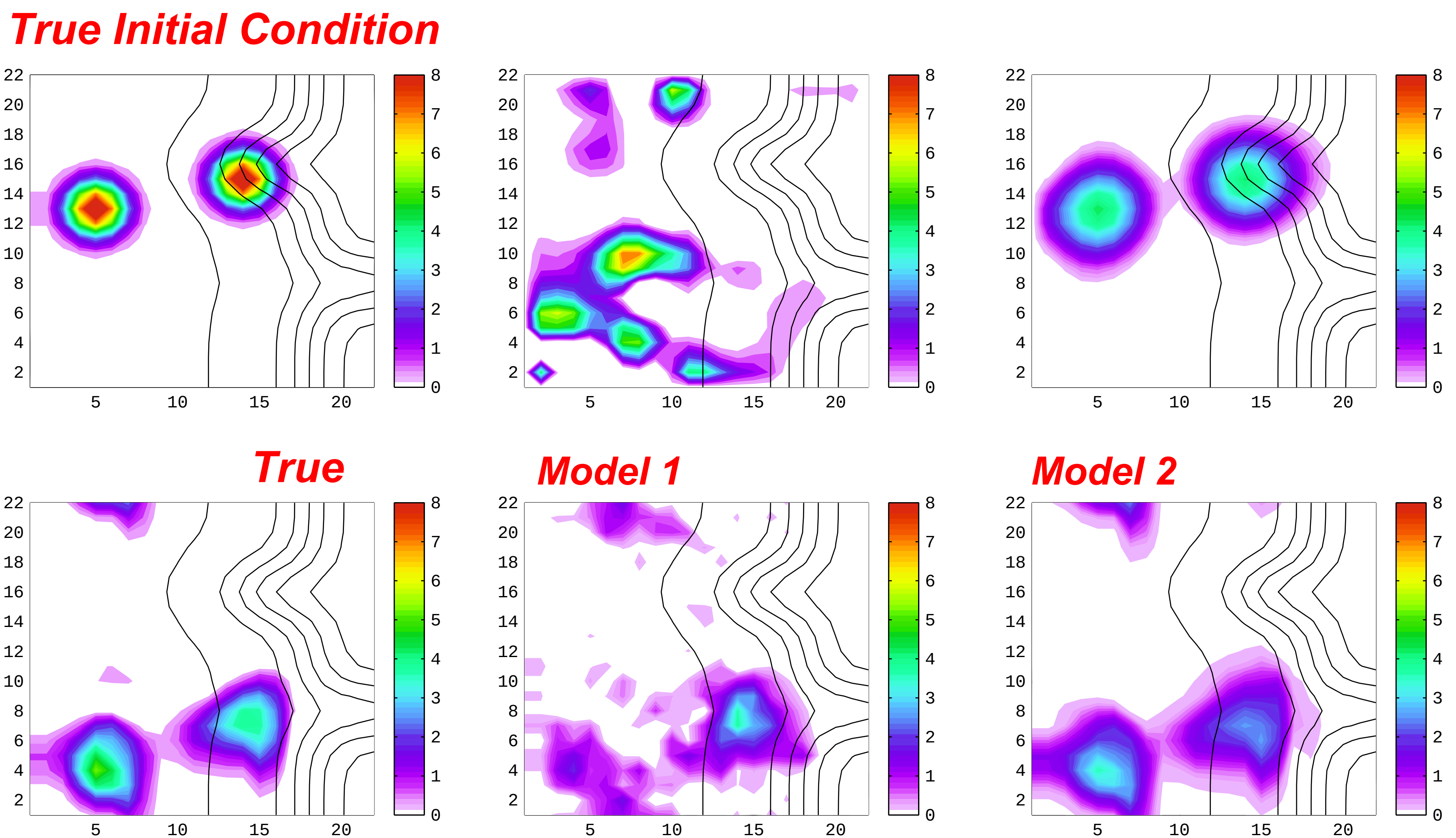


What if we apply more smoothing?

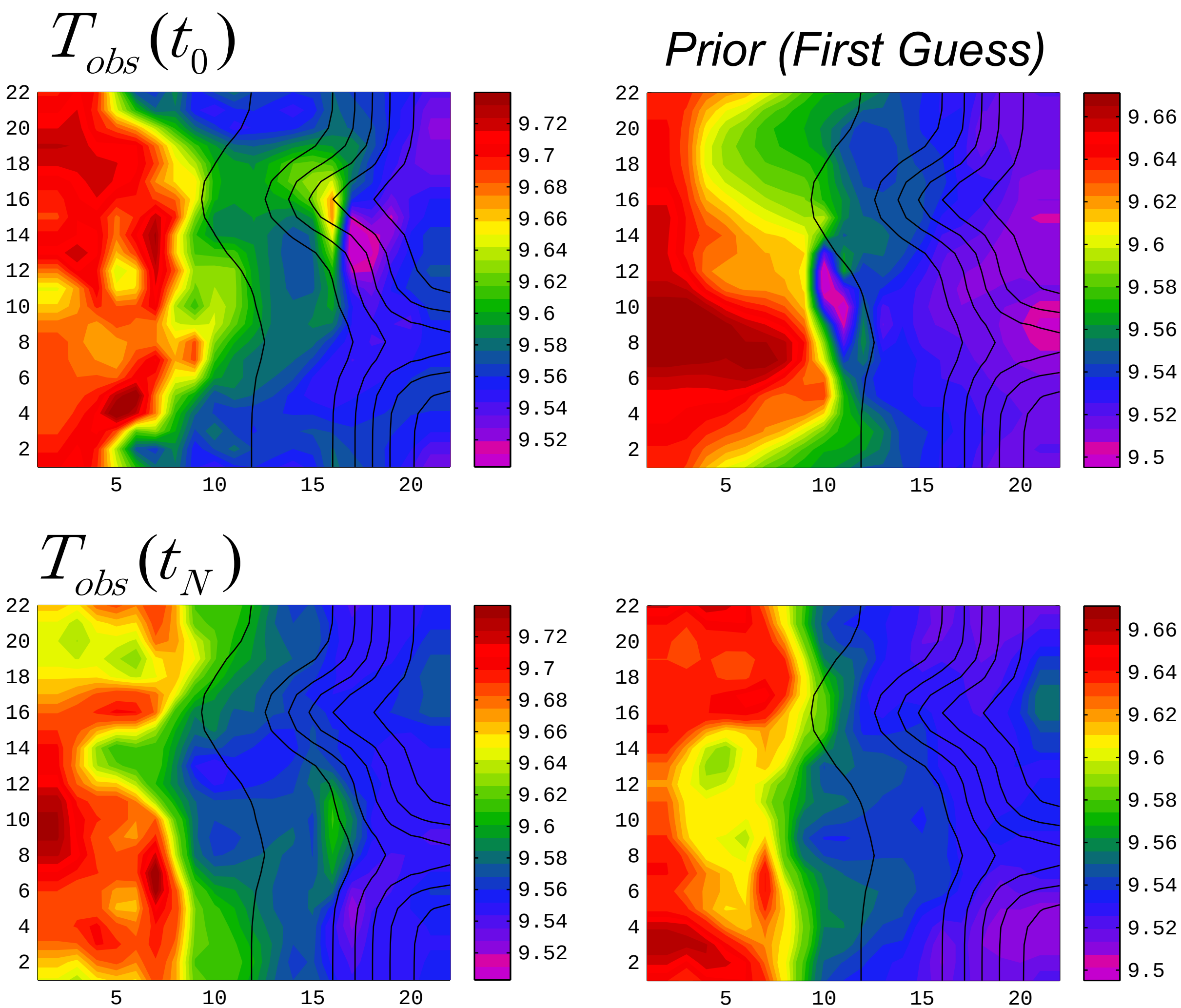
True Initial Condition | Wrong Model | Good Model



Assimilation of data at time t_N

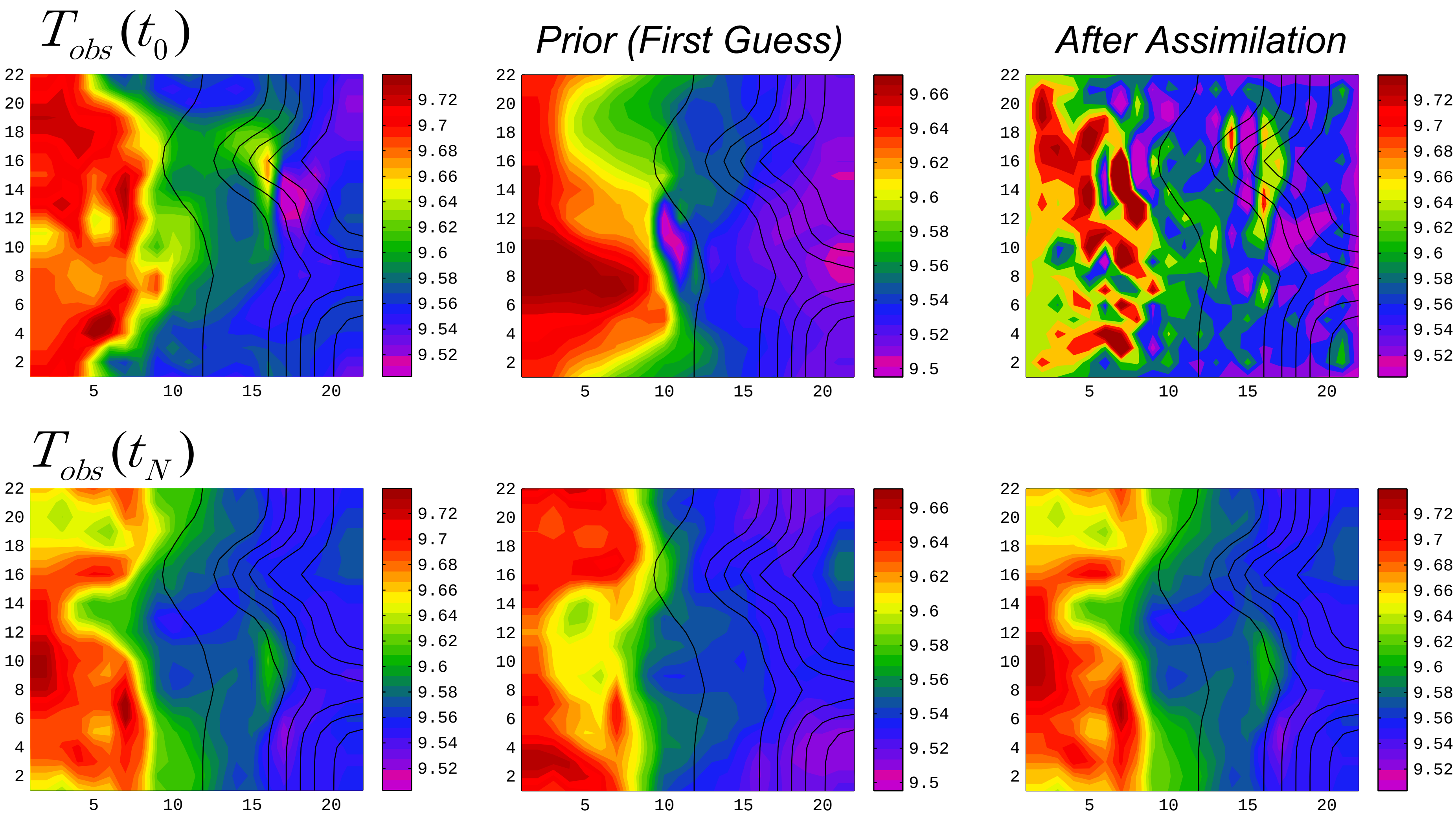


Assimilating *SST*



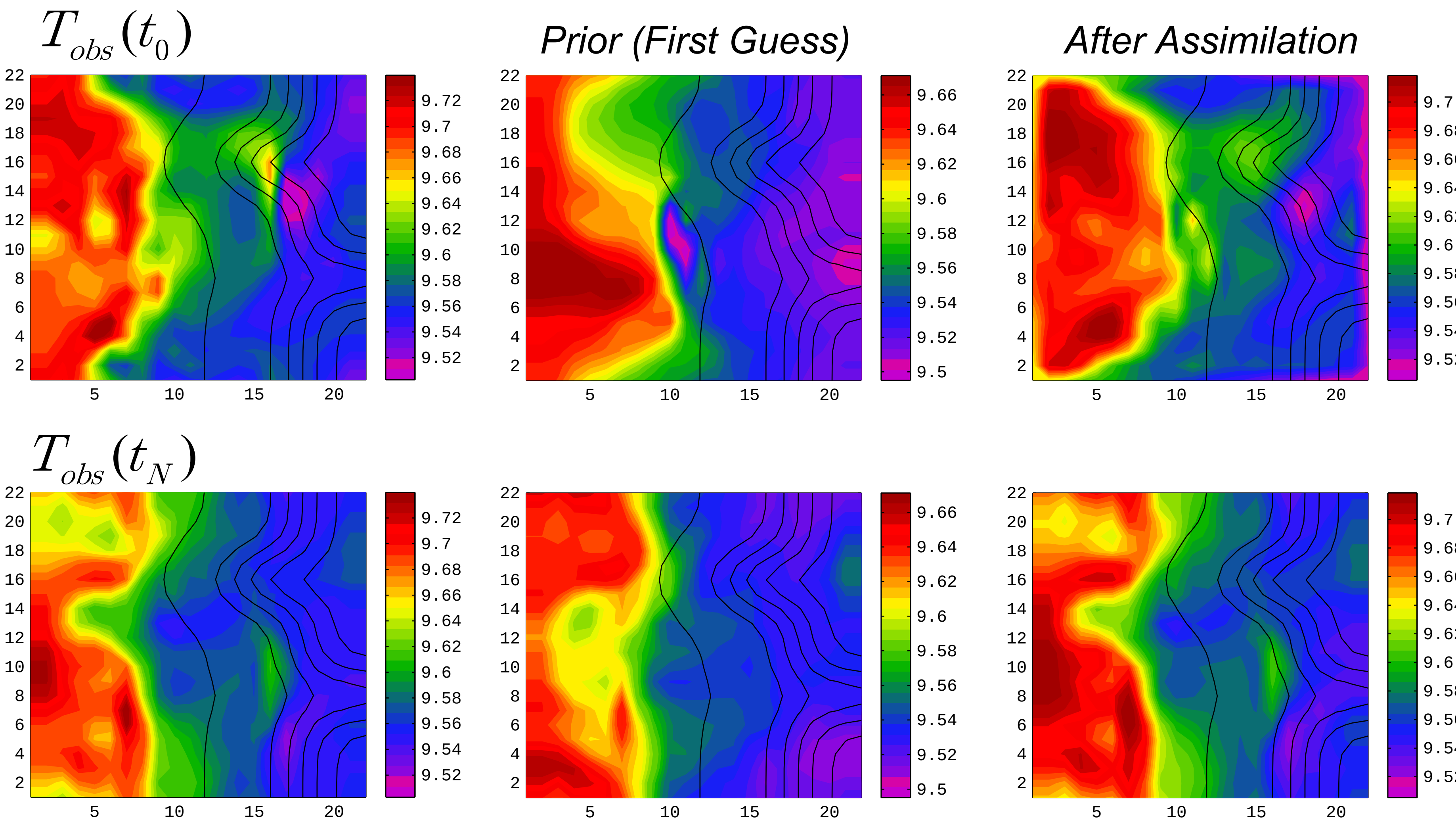
Assimilating *SST*

No smoothing

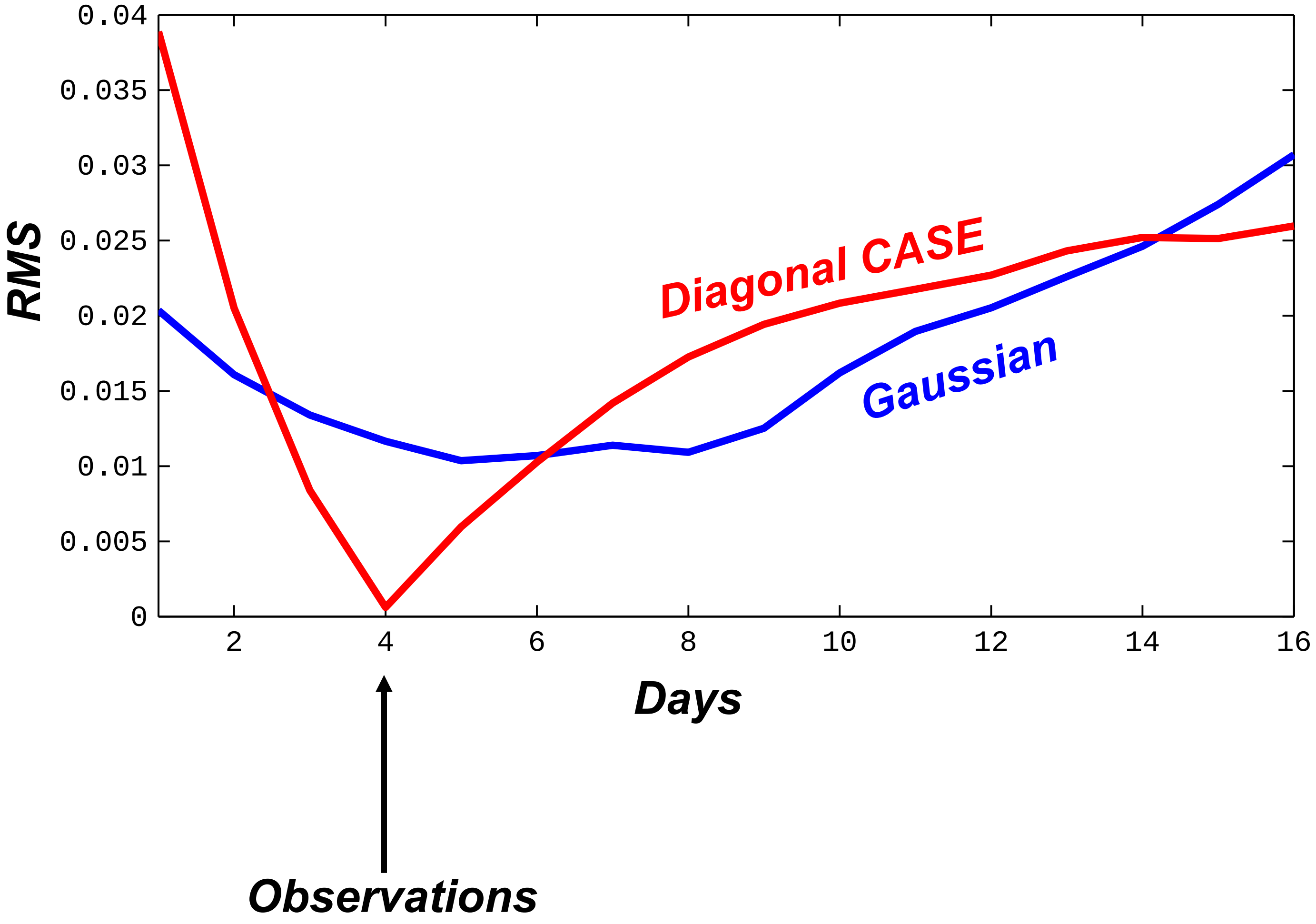


Assimilating *SST*

Gaussian Covariance Smoothing

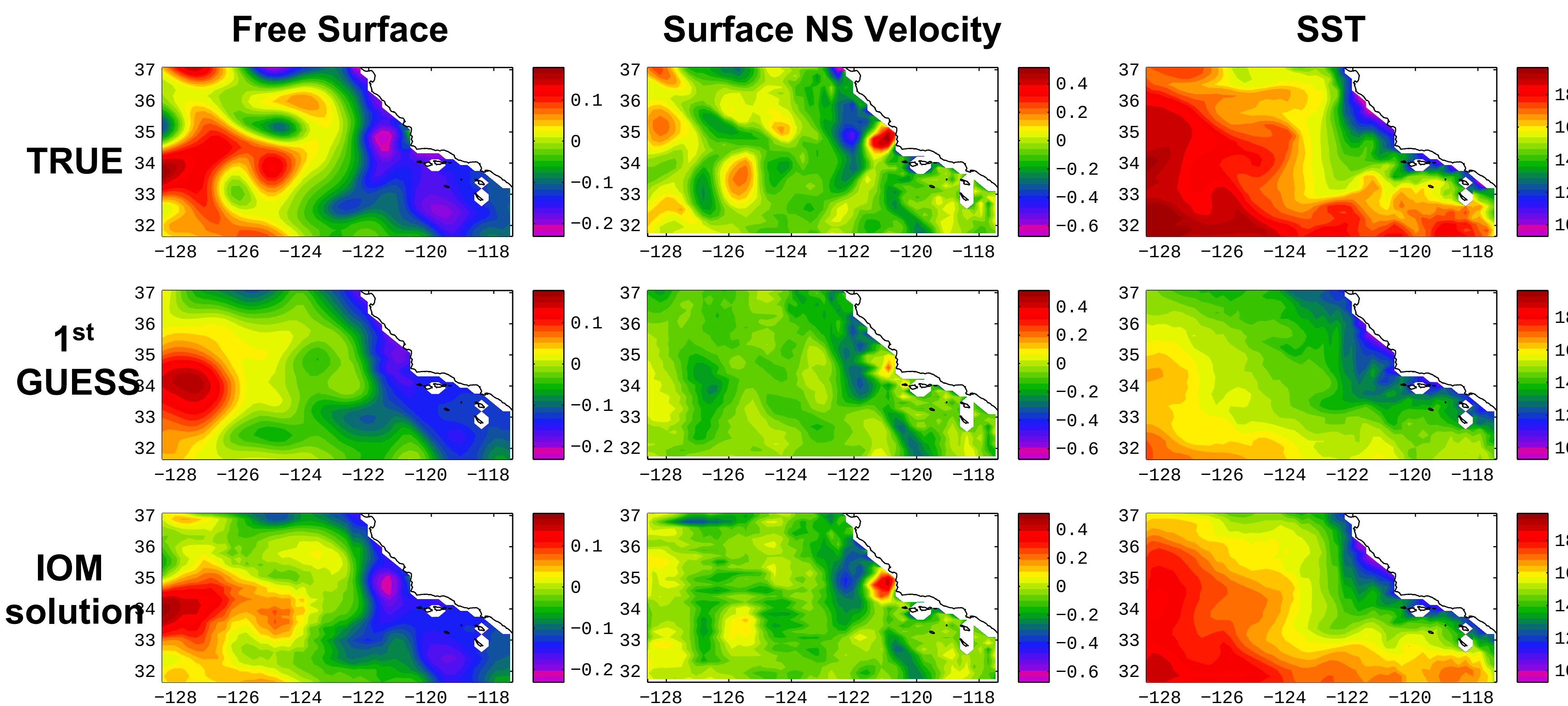


RMS difference from *TRUE*

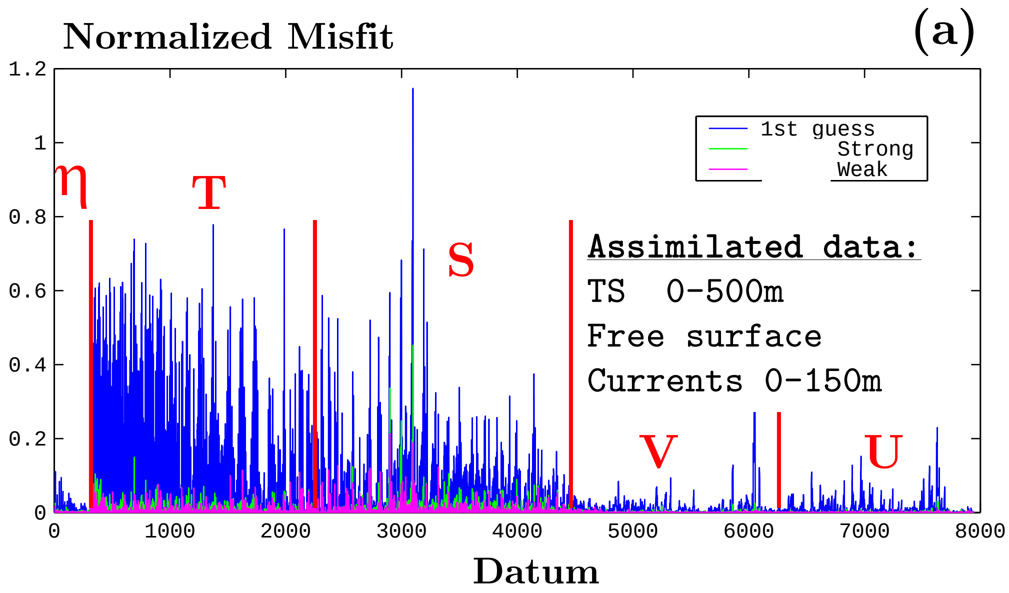


Assimilation of Mesoscale Eddies in the Southern California Bight

<u>Assimilated data:</u>
TS 0-500m
Free surface
Currents 0-150m

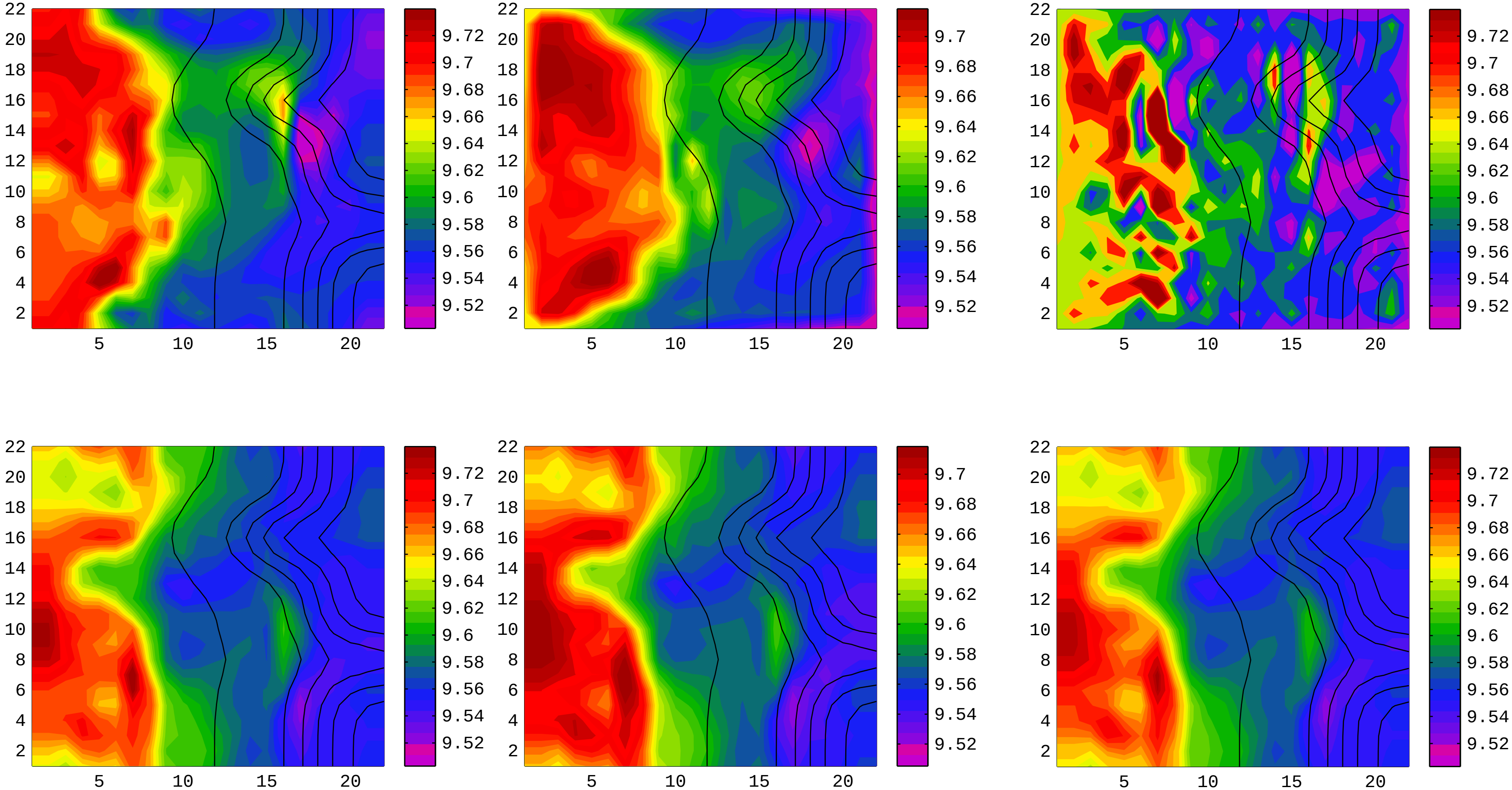


Assimilation of Mesoscale Eddies in the Southern California Bight



Happy data assimilation and forecasting!

Di Lorenzo, E., Moore, A., H. Arango, Chua, B. D. Cornuelle, A. J. Miller and Bennett A. (2005) **The Inverse Regional Ocean Modeling System (IROMS): development and application to data assimilation of coastal mesoscale eddies.** *Ocean Modeling (in preparation)*



22 20 18 16 14 12 10 8 6 4 2
5 10 15 20
9.72 9.7 9.68 9.66 9.64 9.62 9.6 9.58 9.56 9.54 9.52
22 20 18 16 14 12 10 8 6 4 2
5 10 15 20
9.7 9.68 9.66 9.64 9.62 9.6 9.58 9.56 9.54 9.52
22 20 18 16 14 12 10 8 6 4 2
5 10 15 20
9.72 9.7 9.68 9.66 9.64 9.62 9.6 9.58 9.56 9.54 9.52
22 20 18 16 14 12 10 8 6 4 2
5 10 15 20
9.72 9.7 9.68 9.66 9.64 9.62 9.6 9.58 9.56 9.54 9.52
22 20 18 16 14 12 10 8 6 4 2
5 10 15 20
9.7 9.68 9.66 9.64 9.62 9.6 9.58 9.56 9.54 9.52
22 20 18 16 14 12 10 8 6 4 2
5 10 15 20
9.72 9.7 9.68 9.66 9.64 9.62 9.6 9.58 9.56 9.54 9.52