Schur vectors for estimating parameters in filtering algorithm for data assimilation

S.H. Hoang & R. Baraille
Shom, Toulouse

7th Int. Conf. on Sensitivity Analysis of Model Output,
University Nice Sophia Antipolis, Valrose Campus,
Nice, France, 2013

Plan

1. Introduction
2. Reduced-order filter
3. Separation of vertical and horizontal structure of the covariance;
4. Estimation of vertical covariance and of some parameters in horizontal covariance;
5. SPSA for sensitivity analysis
6. Experiment on MICOM
7. Experiment on HYCOM
8. Conclusion

REFERENCES

H.S. Hoang, P. De Mey, O. Talagrand and R. Baraille (1997) « A new reduced-order adaptive filter for state estimation in high dimensional systems, » *Automatica*. 33, pp. 1475-1498.

H.S. Hoang, O. Talagrand and R. Baraille (2001) « On the design of a stable filter for state estimation in high dimensional systems» *Automatica*. 37, 2001, pp. 341-359.

H.S. Hoang, Remy Baraille (2011): Prediction error sampling procedure based on dominant Schur decomposition. Application to state estimation in high dimensional oceanic model. *Applied Mathematics and Computation ,* 218 (7): 3689-3709.

H.S. Hoang, R. Baraille (2011). "On Efficiency of Simultaneous Perturbation Stochastic Approximation Method for Implementation of an Adaptive Filter" , *Computer Technology and Application*, No 2, pp. 948-962.

Spall, J.C. (2000), " Adaptive Stochastic Approximation by the Simultaneous Perturbation Method, *IEEE Transactions on Automatic Control"*, vol. 45, pp. 1839-1853.

H.S. Hoang and R. Barailles (2013) "Adaptive Filter for High Dimensional Inverse Engineering Problems: From Theory to Practical Implementation," *Engineering*, Vol. 5 No. 5A, pp. 70-78. doi: 10.4236/eng.2013.55A010.

I. Introduction

1. Context : Data assimilation
2. Method : Sequential
3. Difficulties : estimating the Prediction Error (PE) covariance matrix (PECM);
4. Approach :
   . adaptive filter;
   . gain parametrization;
   . mean prediction error (MPE);
   . hypothesis on separation of vertical and horizontal structure (SeVHS) for the covariance;
5. Real Schur : for generating PE samples and estimating PECM;
6. « True » PECM is estimated from PE samples;
7. SPSA - Simultaneous Perturbation Stochastic Approximation (SPSA) - a powerful tool for sensitivity analysis and system optimization;
8. Optimal parameters : found by minimizating the mean distance between « true » PECM and that with SeVHS;
9. Experiments : Lorenz system; MICOM; HYCOM

ADAPTIVE FILTERING APPROACH

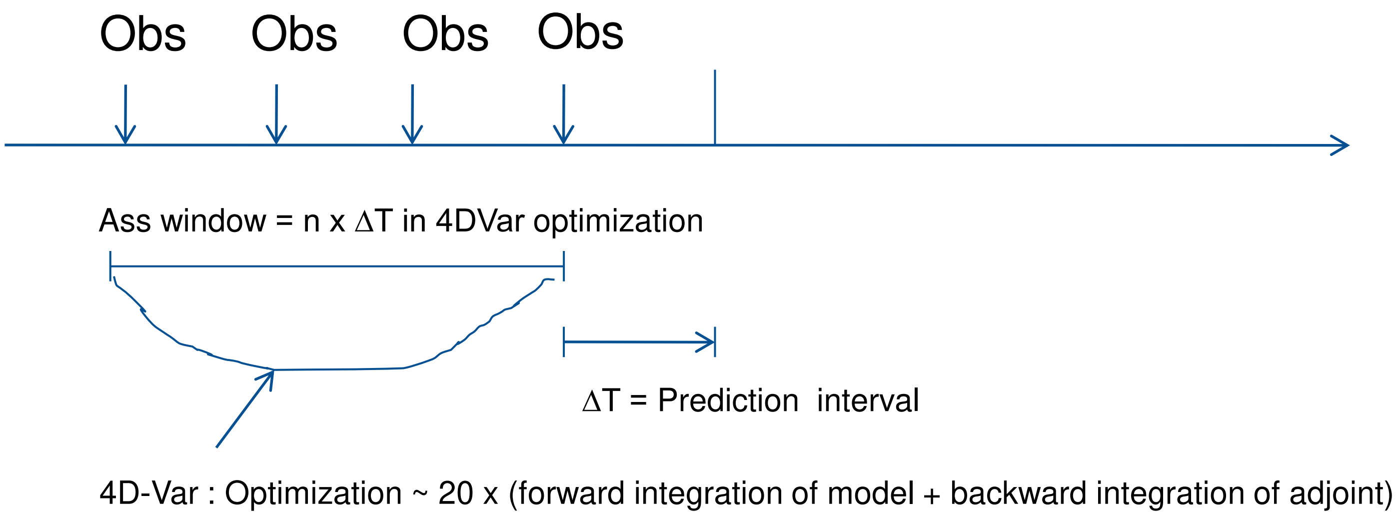
Why Adaptive Filter ?

Advantages:

1. Stability even for unstable dynamics;
2. Computational and memory savings by SPSA (no adjoint code …)
3. Optimality (Mean PE - MPE - of innovation);
4. Dealing with uncertainties:
5. No expensive optimization tool (SPSA)

Why Adaptive Filter ? (continue)

View from 4D-Var optimization



If M is unstable – the error e(0) is amplified by $|\lambda_1|^n$

λ_1 is the first eigenvector of M, $|\lambda_1| > 1$

The same happens for adjoint backward integration

Why Adaptive Filter ? (continue)

AF optimization

AF (without SPSA) : Optimization = forward integration of model
+ backward integration of adjoint over ΔT;
AF (with SPSA) : Optimization = 2 or 3 forward integrations of model over ΔT;

If M is unstable – the error e(0) is amplified by $|\mu_1| < 1$

μ_1 is the first eigenvector of fundamental matrix for the analysis error equation L=(I-KH)M,

Variational

- Mimimizing misfit (obs – **model output**)
- About 20-25 times model integrations over assimilation window

Ensemble Based filter

- Using simulated **error samples**
- 50-100 model integrations over prediction interval

Kalman filtering

- Minimizing in **probabilistic** space
- Number of integrations 10^{12-14}

Differences
between AF and 4D-VAR

Approach	4D-Var	AF
Control vector	Initial state	Gain parameters
Objective	Min Misfit (model output – obs)	Min MPE of system output
Optimization	Batch-vector Iterative	Stochatic approx Sequential
Gradient computation	Integration of model and adjoint over assimilation period	Integration of model at each assimilation instant

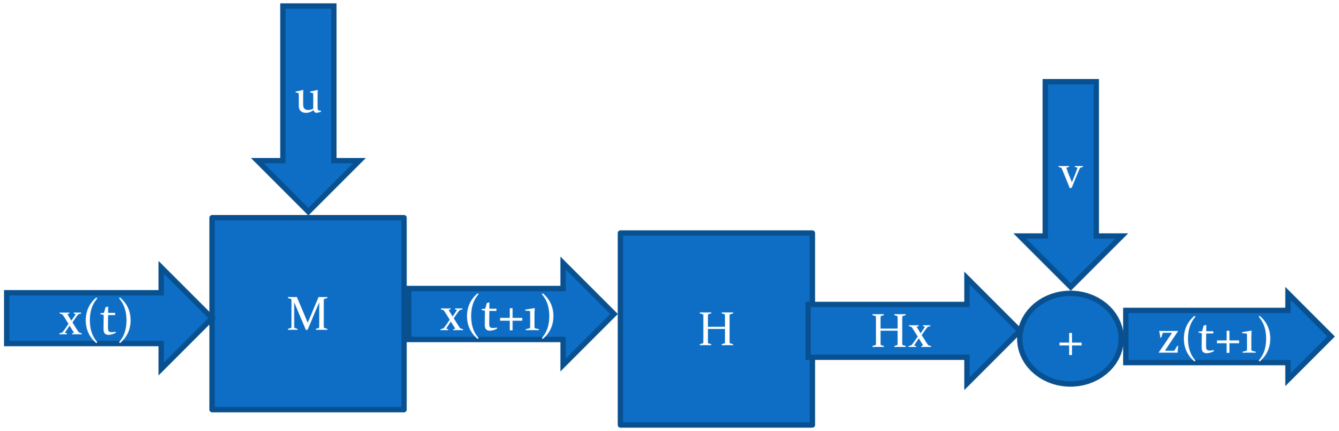
Differences between AF and EnBF

Approach	EnBF	AF
Control vector	Ensemble of samples	Gain parameters
Objective	Approximation to the true ECM	MPE of system output
Optimization	??? Simulating N samples by Monte-Carlo method Sequential	Stochastic approximation (SA) Sequential
Gradient computation	??? N Integrations of model at each assimilation instant	2 ou 3 Integrations of model at each assimilation instant

Principal deas of AF

1. Model: state-space innovation
2. Control vector : gain parameters;
3. Choice of stable filter structure:
   . Corrections in unstable directions;
   . Parametrization of filter gain;
   . Choice of pertinent control vector in as parameters in filter gain;
4. Optimization : minimizing Mean Prediction Error (MPE) of the system output.

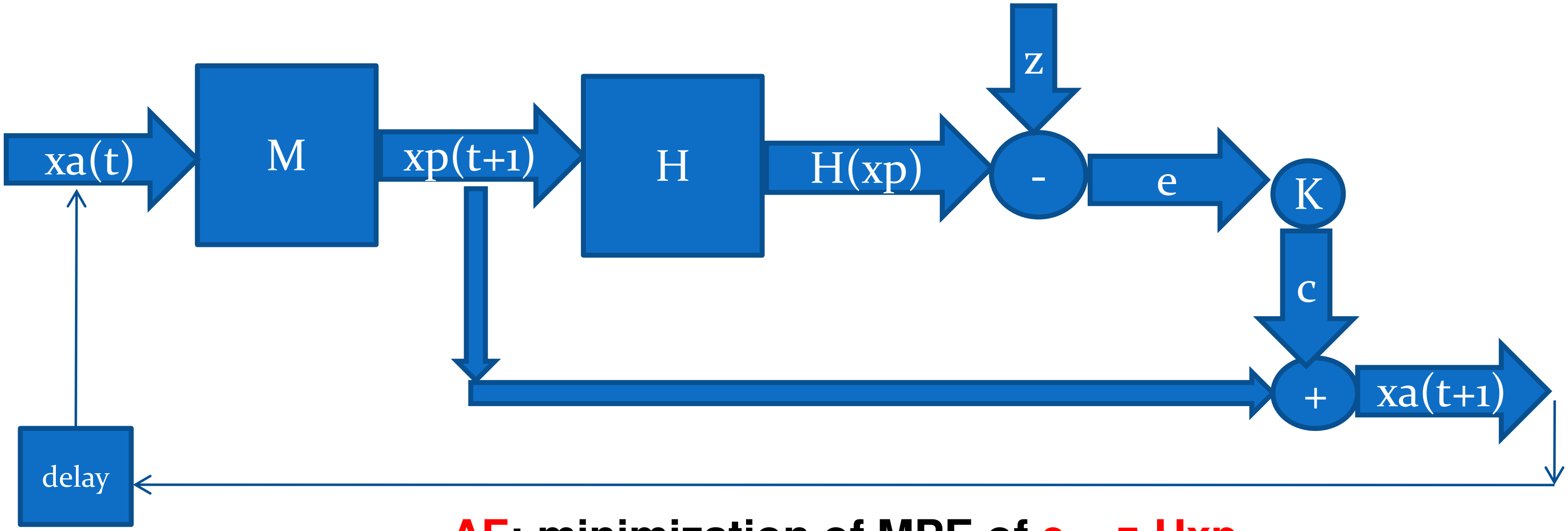
Input-output dynamical system



VAR: minimization of distance between [z(t)] and [x(t)], over assimilation window [t = 1,2,…,N] (control variable = initial state x(0))

x – system state;
u – system input;
M - model;
H – Mesurement Instrument;
v – measurement noise
z - Observations

State space innovation system



AF: minimization of MPE of e = z-Hxp wrt to parameters of K

xa – analysis; **xp** - prediction
M - model; **H** – Mesurement Instrument;
z – Observations
e = innovation
K = gain

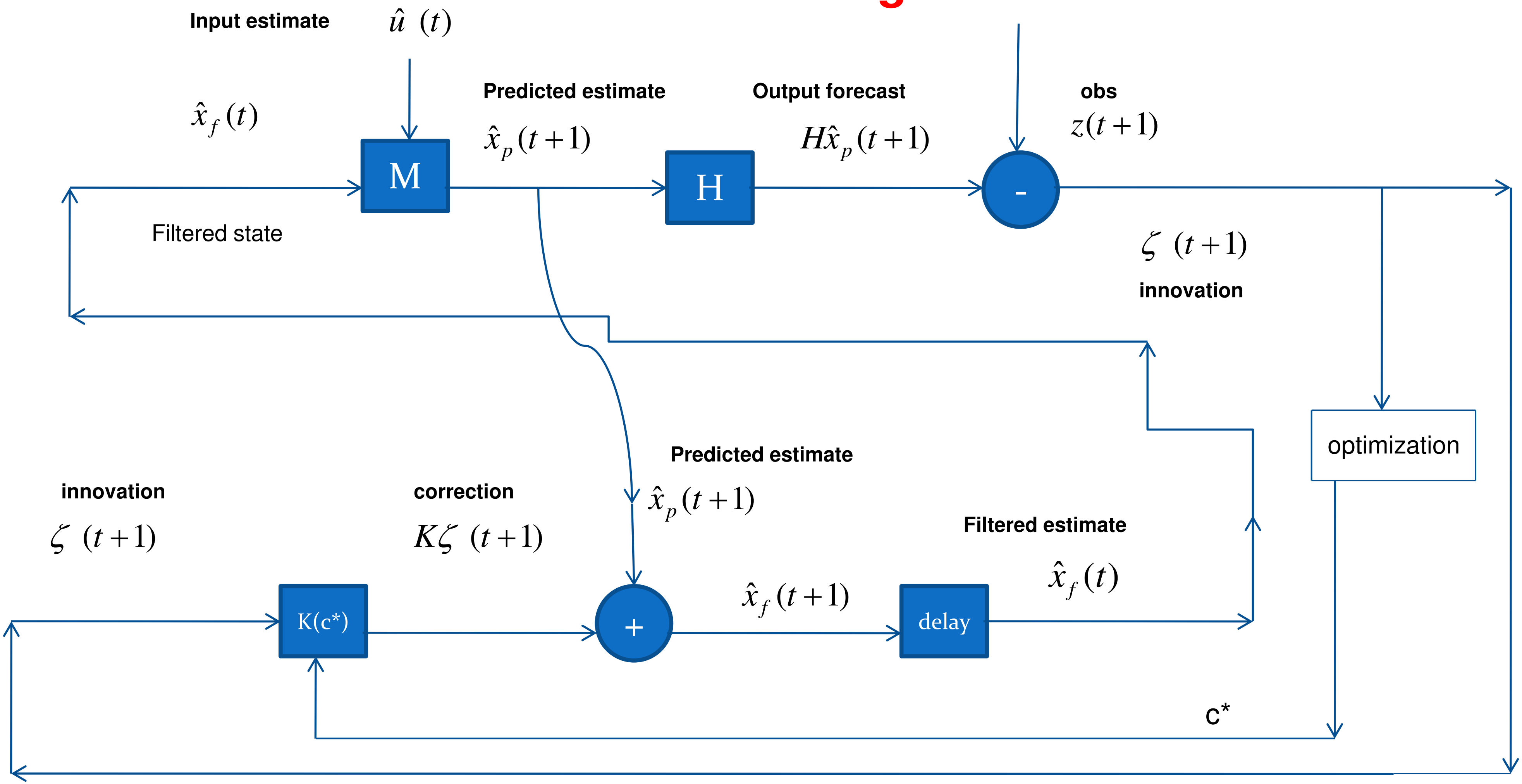
Very important fact !!!

- Equivalence between two representations:

If $e(t)$ is true innovation process (independent sequence) then two representations, process input-output model (PIOM) and state-space innovation representation (SSIR) are equivalent (they produce the same input-output data)

- Example: Kalman filter
- Advantage of SSIR: stability

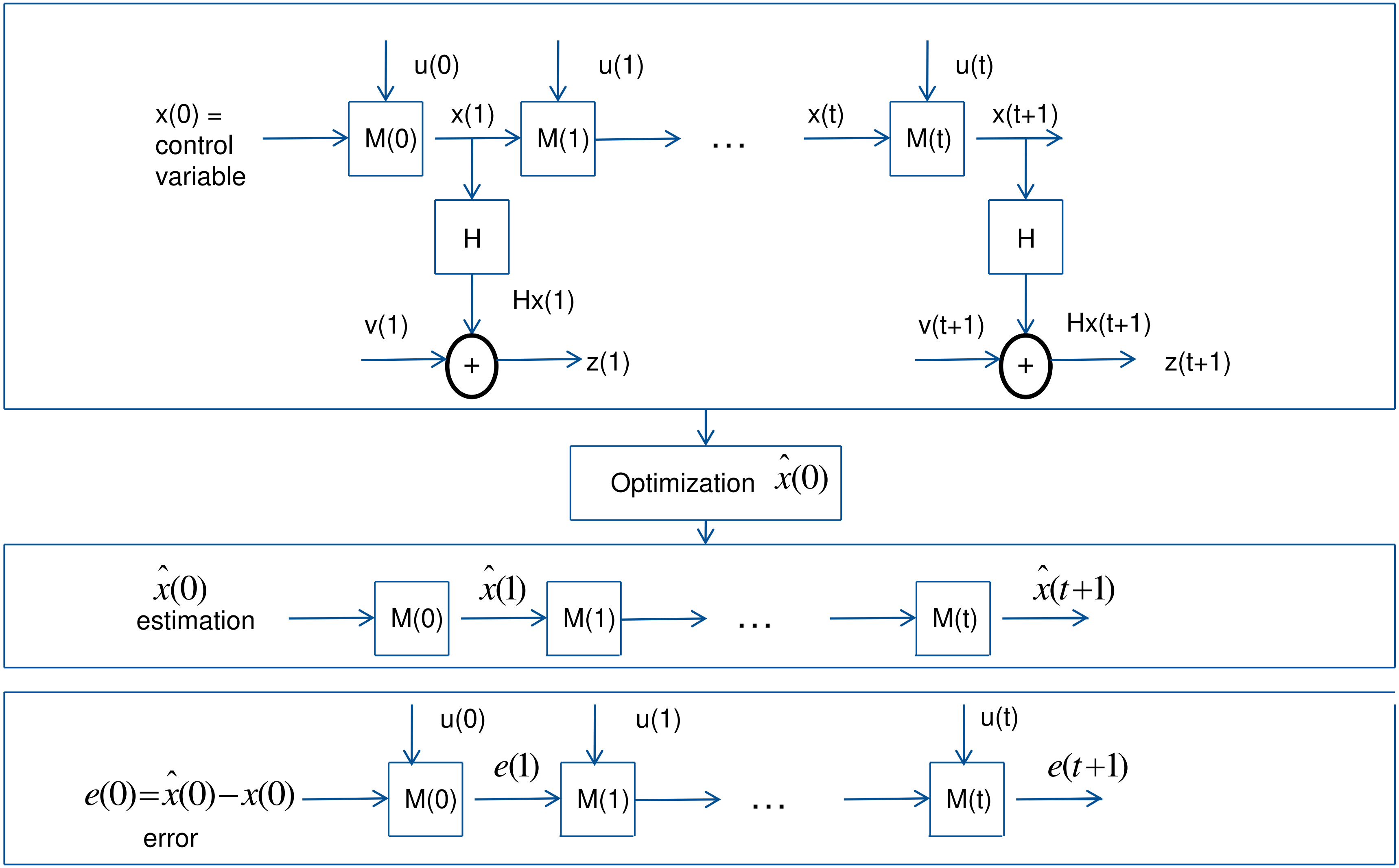
Scheme of AF algorithm



Var = open system; AF = closed system !!!

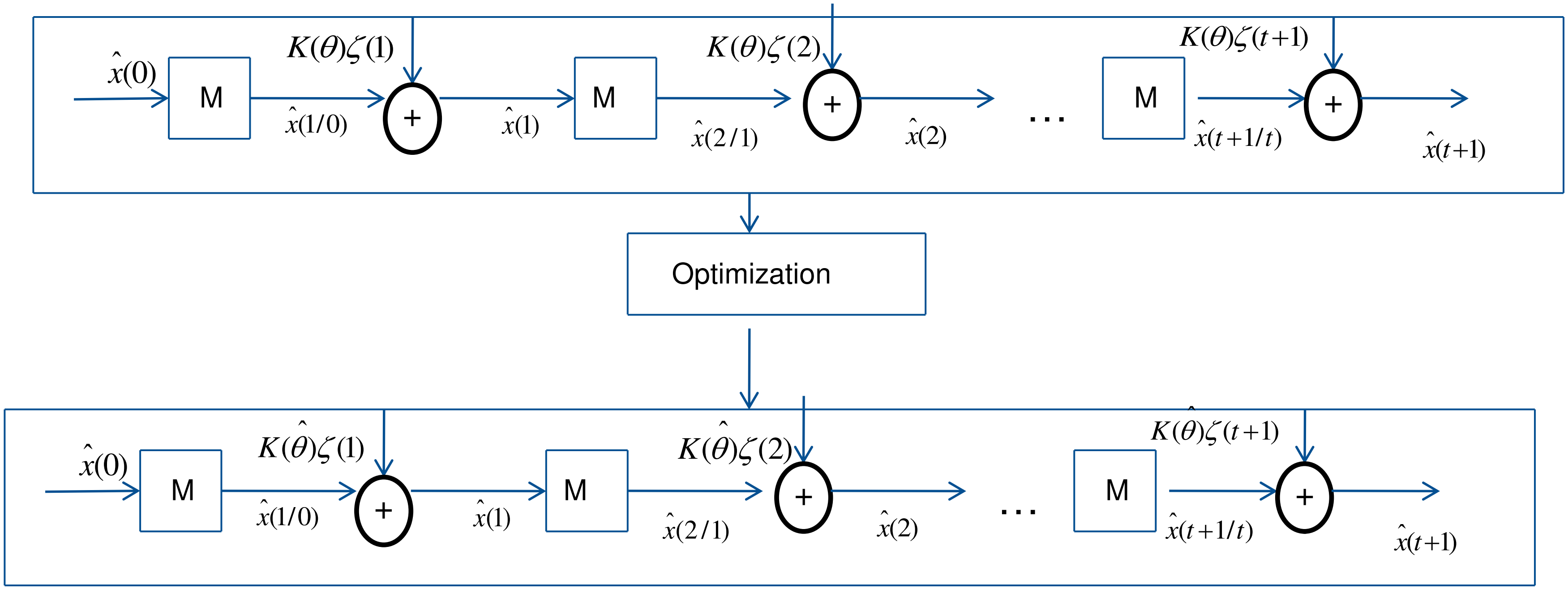
Var = possible unstable; AF : stable

4DVAR : unstable dynamics and unstability



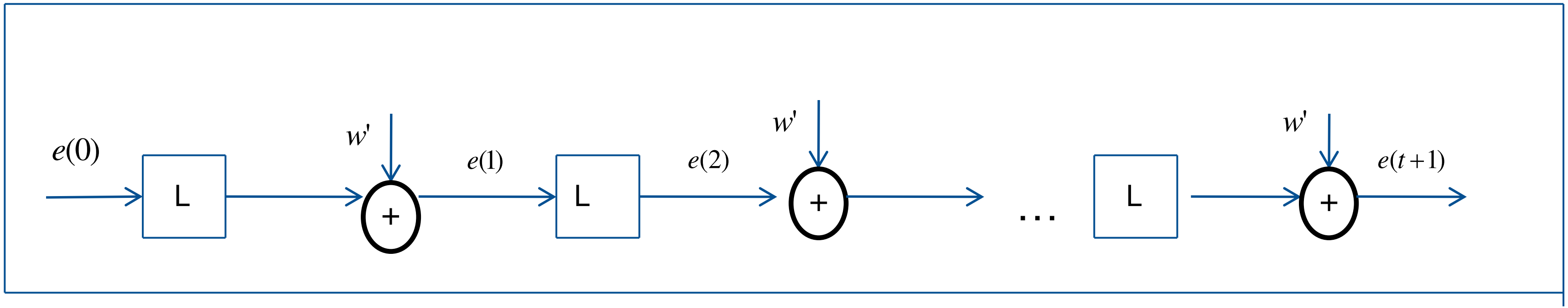
If M is unstable, there is growing of estimation error from e(0) to e(t)

AF : unstable dynamics and stability



$\hat{\theta}$ Belongs to parameter member set ensuring stability of $L=(I-K(\hat{\theta})H]M$

Filtered error equation



$e(t) = \hat{x}(t) - x(t)$	Filtered error
$L(\hat{\theta}) = (I - K(\hat{\theta})H)M$	Fundamental matrix for filtered error dynamics = stable (by construction)
M	Fundamental matrix for model dynamics = may be unstable

If M is unstable, L is stable, the estimation e(t) is bounded

II. ROAF: Standard filtering, stability, adaptation ...

-System dynamics

$$x(t+1) = Mx(t) + w(t)$$

-Observations

$$z(t+1) = Hx(t+1) + v(t)$$

ROAF: Standard method, stability, adaptation ...

Adaptive Filter

$$\hat{x}\left(t+1\right) = A\hat{x}\left(t\right) + K\zeta(t+1) = L\hat{x}\left(t\right) + Kv(t+1),$$

$$\zeta(t+1) := z(t+1) - HA\hat{x}(t),$$

$$K = MH^T\left(HMH^T + R\right)^{-1}, \quad L = (I - KH)A$$

$$J(K) = E\left[\left\|\zeta(t+1)\right\|^2\right] \to \min_K$$

Kalman gain

$$K(t+1) = K(t) - \gamma(t+1)\nabla_{K(t)}\Psi\left[\zeta(t+1)\right]$$

AF gain

Why adaptive filter ?

- Dynamics in VAR = M = unstable (if the system is unstable);

⟹ risk of unstability of optimization, estimation process;

- Dynamics in AF = L = stable (even if M system is unstable);

⟹ stability of optimization and estimation process;

- Optimization in VAR = expensive (~ 20 iterations; 1 iteration := 1 integration of direct model + 1 integration of adjoint model; Integration ~ over assimilation window : 5-10 assimilation intervals)

- Optimization in AF = non-expensive (~ 1 iteration; 1 iteration = 3 integrations of direct model + 0 integration of adjoint model; Integration: ~ over 1 assimilation interval)

Example 1:

Computer capacity: 10**12 operations/s (tera flops)

Model M : (nxn), n = dimension = 10**7

One model time step = dt = matrix-vector multiplication =

dt = M x = n**2 = 10**14 ~ 10**2 (s) = 1.67 mn

5 days prediction = 10**3 iterations of (dt) =

10**3 x 1.67 mn = 27 h ~ 1.16 day

Kalman filter: with Riccati equation,

Matrix-matrix multiplication = M*M = 10**14 x (1.67) mn = 31,7 years !!!

Example 2:

Capacity = most fast (world record) : 10**16 operations/s (10 petaflops)

Model M : (nxn), n = dimension = 10**7

One model time step = dt = matrix-vector multiplication =

dt = M x = n**2 = 10**14 ~ 0.01 (s)

5 days prediction = 10**3 iterations of (dt) =

10**3 x 0.01 s = 10s

Kalman filter: with Riccati equation,

Matrix-matrix multiplication = M*M = 10**14 x 10s= 1150 days = 3.17 years !!!

Why (dominant) Schur vectors ?

- Stability of filter – projection of innovation onto subspace spanned by dominant EVs, SVs or ScVs
- Evs (eigen vectors) – NO : numerical unstability, may be complex …
- SVs (singular vectors) – NO : required linearized system, construction of adjoint.
- ScVs (real Schur vectors) – Yes: stability, no request for linearized and adjoint codes;

⟶

- 4DVAR: Great difficulty in choosing a structured correction to initial state;
- AF : Easy choice for control vector in stabilizing filter gain;

ROAF: Standard method, stability, adaptation

How can we approximate the PECM
In the context of very high dimensional systems ?

Different Methods:

- **Riccati Equation (Kalman Filter)**
- **Specify a-priori analytical form;**
- **EOFs;**
- **EnOI: Simulating PE-samples (differences between model solutions and its mean value);**
- **…**

III. Separation of vertical and horizontal structure of covariance

1. Assuming ***M*** has a separable vertical and horizontal structure

$$M(s_v, s_h; s'_v, s'_h) = M_v(s_v; s'_v) \otimes M_h(s_h; s'_h)$$

$\otimes$ denotes Kronecker product

Separation of vertical and horizontal structure ...

2. Generating a set of PE-samples using Sampling Procedure (Hoang and Baraille, 2011). They develop in the directions of dominant real Schur vectors;

3. Constructing « true » covariance from simulated PE samples;

4. Estimating unknown parameters in vertical and horizontal covariances by minimizing the distance (in Frobenius norm) between « true » covariance and suggested covariance with SeVHS.

IV. Estimation of vertical covariance and of some parameters in horizontal covariance

- « True » covariance

$$M^*(T) = \frac{1}{T}\sum_{k=1}^{T} M(k), M(k) = \frac{1}{p} S(i) S^T(i)$$

$S(i), i = 1,\ldots,T$ obtained from Prediction error Sampling Procedure

$$S(i+1) = MX(i), X(i) = \left[\delta x^1(i),\ldots,\delta x^p(i)\right], i = 0,\ldots,T-1$$

Gram-Schmit orthogonalization: columns of $X(i+1)$ are orthogonormal

$$S(i+1) = X(i+1)G,$$

Estimation of parameters

. Parameters $\theta = (c_{kl}, \rho_x, \rho_y, \rho_{xy})$

.Elements of **vertical** covariance matrix $[c_{kl}]$

. Correlation length :

+ in OX axis and OY axis (ρ_x, ρ_y)

+ common for OX axis and OY axis (ρ_{xy})

Optimization problems

Average cost function

$$J[\theta] = E[\Psi(\delta M(k), \theta)] \Rightarrow \min_\theta$$

$$\Psi\ (\delta M(k)\ , \theta) = \left\| M^*(t) - M_v(c_{lm}) \otimes M_h(\rho_x, \rho_y, L_d) \right\|_F^2$$

$\|\cdot\|_F$ - Frobenius matrix norm

SPSA – Tool for sensitivity analysis and optimization (Spall)

1. J(x) = E[F(x)], x = (x(1),…,x(n)); E(.) – math. expectation
2. Computation of gradient of J wrt to x by perturbing all components of x,
3. x'(i) = x(i) + d(i), d(i) assumes +/- 1 with probability ½
4. dF(i) = (F(x')-F(x))/d(i), i = 1,…,n :

   for very large n, gradient is approximated by

   two model integrations

5. Hessian may be approximated by three model integrations !!!

Algorithm for estimating vertical ECM

$$(c_{kk'})$$

$$\nabla_c \Psi\left[c_{kk'}(t)\right] = \frac{1}{n_u} \sum_{l} \sum_{i,j} \sum_{i',j'} m^l(i,j,k;i',j',k') \exp\left[-\frac{d}{L_d(t)}\right]$$

$$m^l(i,j,k;i',j',k') = \delta x_p^l(i,j,k)\,\delta x_p^l(i',j',k') - C(t)$$

$$C(t) := c_{kk'}(t) \exp\left[-\frac{d}{L_d(t)}\right] \exp\left[-\frac{d}{L_d(t)}\right]$$

Algorithm for estimation of L_d

1. Can be written out as done for $c_{kk'}$

2. The following is preferred to :

Let be given the set of PE samples

$$S[T] = \{\delta x_p^l (s_v, s_h, k), k = 1, \ldots, T\}$$

Step 1. For fix $s_v = l$ and fix k, estimate the correlation function $\hat{C}_h(d)$ from $\delta x_t^l(i,j,k)$

Step 2. Find $\hat{d} := \hat{d}(s_v, k)$ from $\hat{C}_h(\hat{d}) = \frac{1}{e}$

Step 3. $\hat{d} = \frac{1}{T}\sum_{k=1}^{T}\hat{d}(.,k), \hat{d}(.,k) = \frac{1}{N_l}\sum_{l=1}^{N_l}\hat{d}(l,k)$

Step 4. The estimated correlation length is $\hat{L}_d(T) = \hat{d}(T)$

One important results: under certain conditions, SeVHS in Cov implies SeVHS in filter gain

$$K = \begin{bmatrix} k(1) \\ . \\ k(n_L) \end{bmatrix} \otimes K_h \qquad \approx K_h \zeta$$

Example: Altimetry with OI

Lorenz system: assimilation experiment

Model

$$\partial x / dt = \sigma(y - x),$$

$$\partial y / dt = x(\rho - z) - y$$

$$\partial z / dt = xy - \beta z$$

Lorenz used the values $\alpha = 10, \beta = 8/3, \rho = 28$

The system exhibits chaotic behavior for these values

True Initial State

$$x_1^* = 1.50887, x_2^* = -1.531271; x_3^* = 25.46091$$

$$x_1(0) = 5, x_2(0) = 10, x_3(0) = 27$$

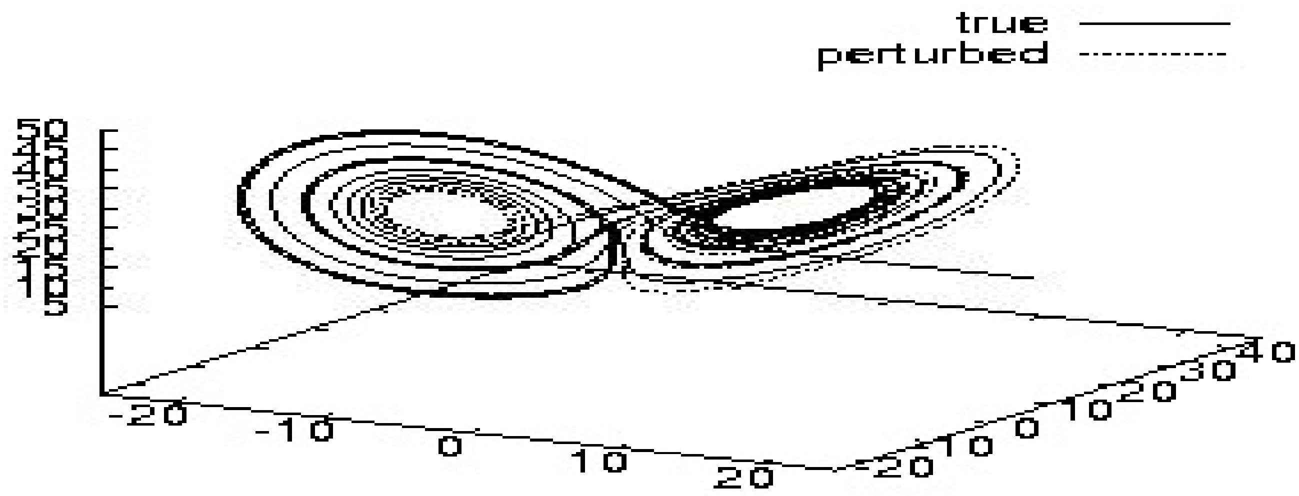
Model time step and obs interval

$$dt = 0.01, DT = 25dt$$

Obs operator and obs noise

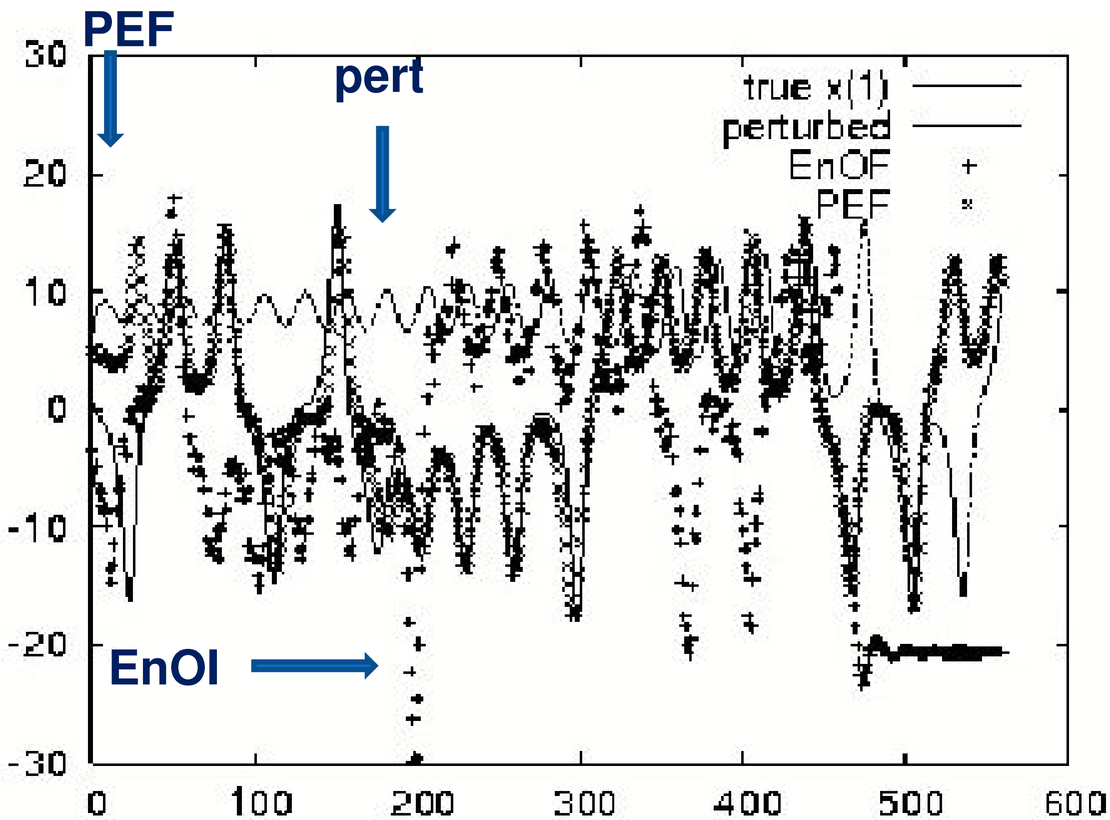
$$H = (1,1,1),\ \sigma_r^2 = 2.$$

Difference between true and perturbed trajectories



Devergence in trajectories due to small
Perturbation in initial state

Assimilation by different methods



First component in Lorenz system:
Pert – with perturbed initial state
EnOI – Ensemble OI
PEF – Prediction Error Filter

SPSA – Powerful tool for sensitivity analysis and optimization (Spall, 2000)

Example: Local methods for sensitivity analysis

1. Local methods involve taking the partial derivative of the output *Y* with respect to an input factor *X(i)* taken at some fixed point in the space of the input (hence the 'local' in the name of the class).

2. Adjoint modelling and Automated Differentiation are methods in this class. They examine small perturbations, typically yields partial derivative of one variable (*Y* = *J* = cost function) at a time wrt to components of *X* (control variables).

3. SPSA allows to obtain the partial derivatives of any component of the output vector *Y(l),l = 1,…,p* with respect to any input component of *X(i)*, *i = 1,…,n.* All derivatives *[dY(l)/dX(i)]* are obtained at the same time by simultaneous perturbation of all components of *X (two integrations of model).*

Example: Local methods for sensitivity analysis (continue)

Suppose $y = F(x)$, $y = (y(1),..,y(p))$, $x = (x(1),\ldots,x(n))$;

Algorithm (for estimating all partial derivatives (derivative of each component of $y(k)$ wrt to any component $x(l)$ of X by perturbing all components of X)

1. $x'(l,m) = x(l) + d\_m(l)$, $d(l)$, $l = 1,2,\ldots,n$ are independent assuming +/- 1 with probability ½, m is number of samples (iteration)
2. Integrate $Y' = F(X')$,
3. $dy(k,m)/dx(l,m) = E[(y'(k)-y(k))/d(l)] \sim (1/N)\sum_{m}[(y'(k,m)-y(k))/d(l)]$, $l= 1,\ldots,n$; $E(.)$ – math expectation.

for large (p,*n*), gradient is approximated only by two model integrations at each iteration

4. Hessian matrix may be approximated by three model integrations !!!

Numerical experiment

In practice we are given usually the situation (for very high dimensional complex system) $y = Ax$, where Ax' symbolizes the computer code which yields the value of *y'* given *x'*. Usually *A* is unknown.

Let x^* be a solution of the problem $z = Ax^*$. Given *z* our task is to find x^*. For the experiment let the true unknown *A*, x^* and *z* be

$$A = [a(i,j)],\ i, j = 1,2,\ a(1,1) = 1, a(1,2) = 2,$$
$$A(2,1) = 3,\ a(2,2) = 4,\ x^* = (1,2),\ z = (5,11).$$

Algorithm: To solve this problem, we will use the SPSA.

1. Identify *A* by estimating all partial derivatives of *y* wrt *x* using SPSA. These partial derivatives give *Ae* – estimation for *A*; The procedure is

$$dY(k)/dX(l) = E[(Y(k + db(k)) - Y(k))/db(k)],$$

$E(.)$ **–** mathematical expectation

$db(k),\ k = 1,\ldots,n$ are i.i.d Bernulli distributed (+/-1 with prob ½).

2. Solve the minimization problem :

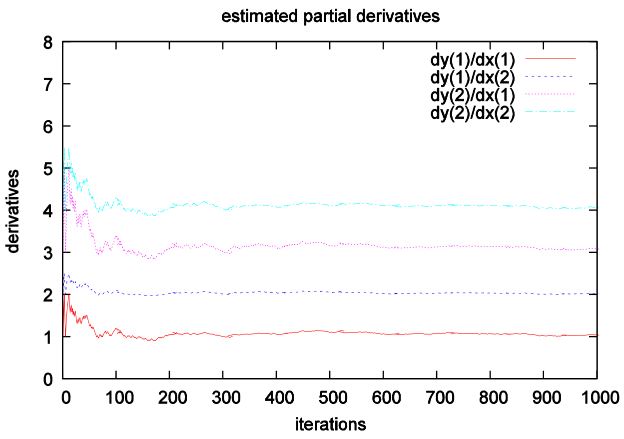
$$J(x) = E<z - Ae\ x, z-Ae\ x> \rightarrow \arg\min(x)$$

by estimating the gradient *G* of *J* wrt to *x* using SPSA. At $(i+1)$ iteration,

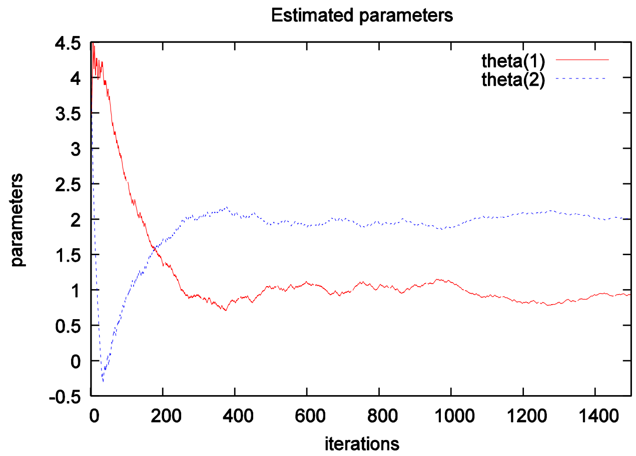
$$theta'(i+1) = theta'(i) - G(i)/((i+1)^{**}0.7)$$

$$G(i) = [J(theta'(i)+d(i)) - J(theta'(i))]/d(i),\ d(i) = db(i)/[i^{**}0.2]$$

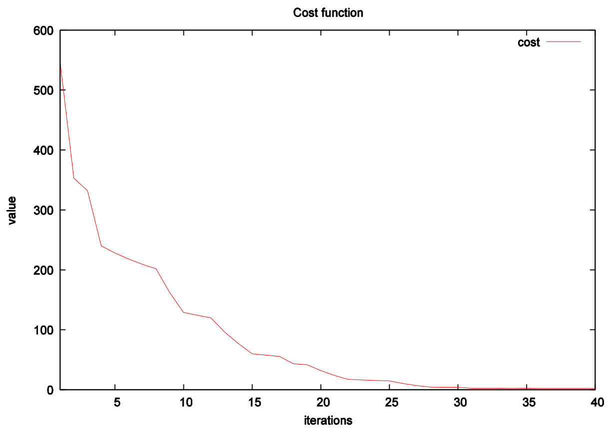
$$theta(i+1) = theta(i) + [theta'(i+1) - theta(i)]/(i+1)$$



Partial derivatives : convergence to elements of Ae to A after about 200 iterations

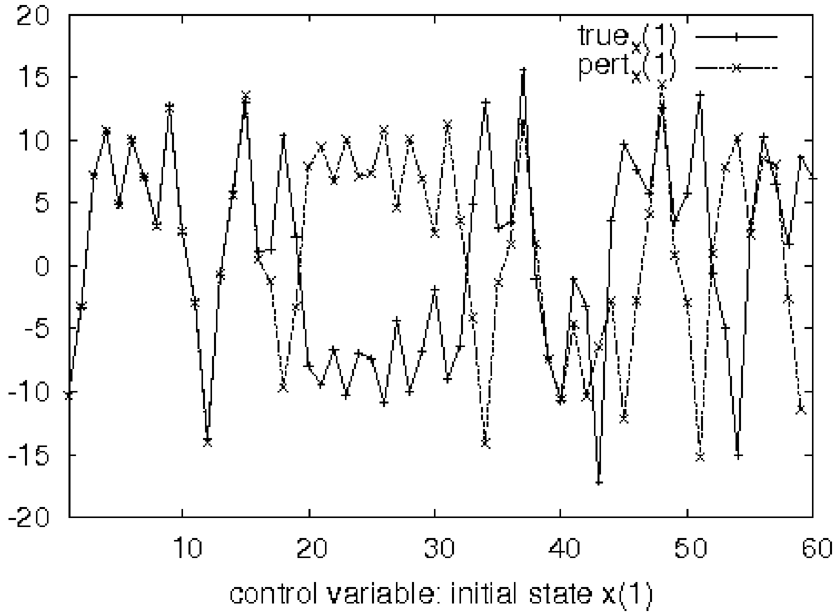


Convergence of estimated solution to the true solution X*



Cost function

Chaos in Lorenz system
sensitivity

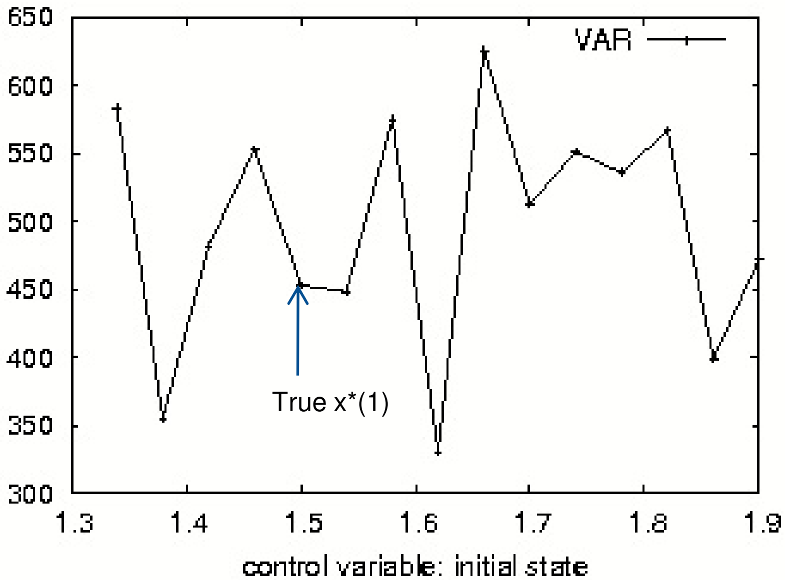


$x_1^* = 1.50887, x_2^* = -1.531271; x_3^* = 25.46091 \quad x_1(0) = 1.48, x_2(0) = x_2^*, x_3(0) = x_3^*$

Lorenz system: very small perturbation of the first component of the initial state : True trajectory and perturbed one are indistinguishable up to T= 16 DT and after become completely divergent

Sensitivity of cost function to control variable : Var

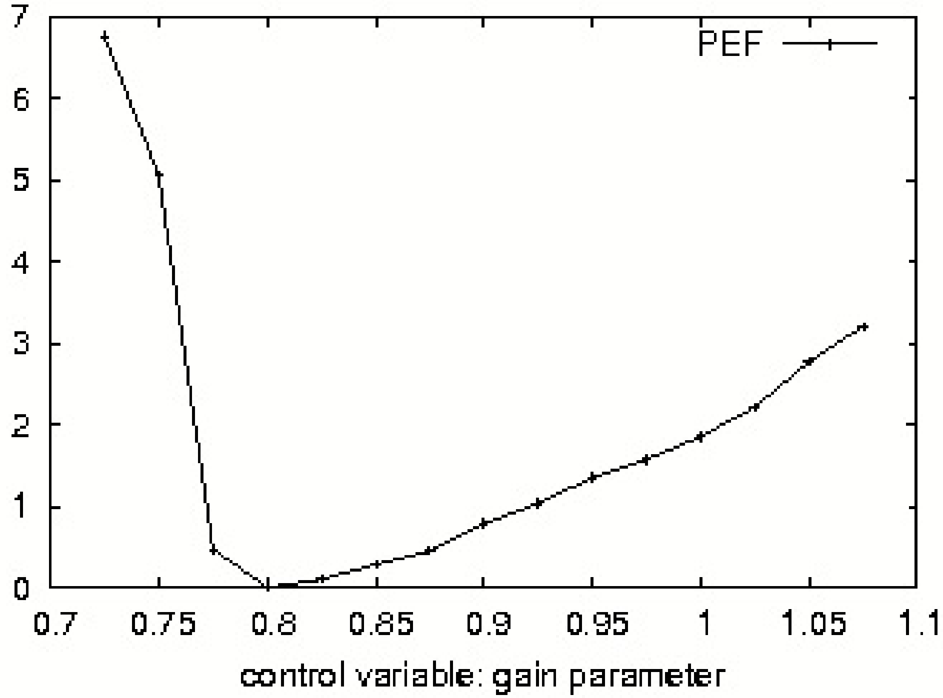
Var approach : control variable = initial state x(1)



Cost function in Var approach as function of initial x(1):
Experiment with Lorenz system, dt = 0.01, DT = 25 dt
X(1) = x*(1)+ dx, x(2)=x*(2), x(3)=x*(3)
It is practically impossible to find x*(1) !!!

Sensitivity of cost function to control variable : AF

AF approach : control variable = gain parameter



Cost function in AF approach: Experiment with Lorenz system,
dt = 0.01, DT = 25 dt
X(1) = x*(1)+ dx, x(2)=x*(2), x(3)=x*(3)
In gain space it is easy to find an optimal parameter

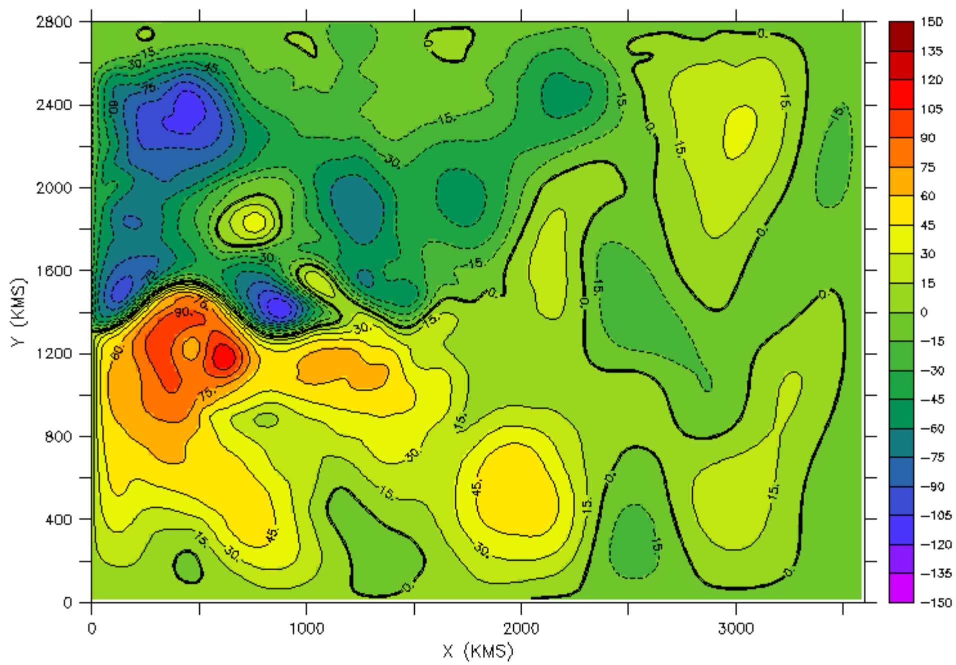
MICOM

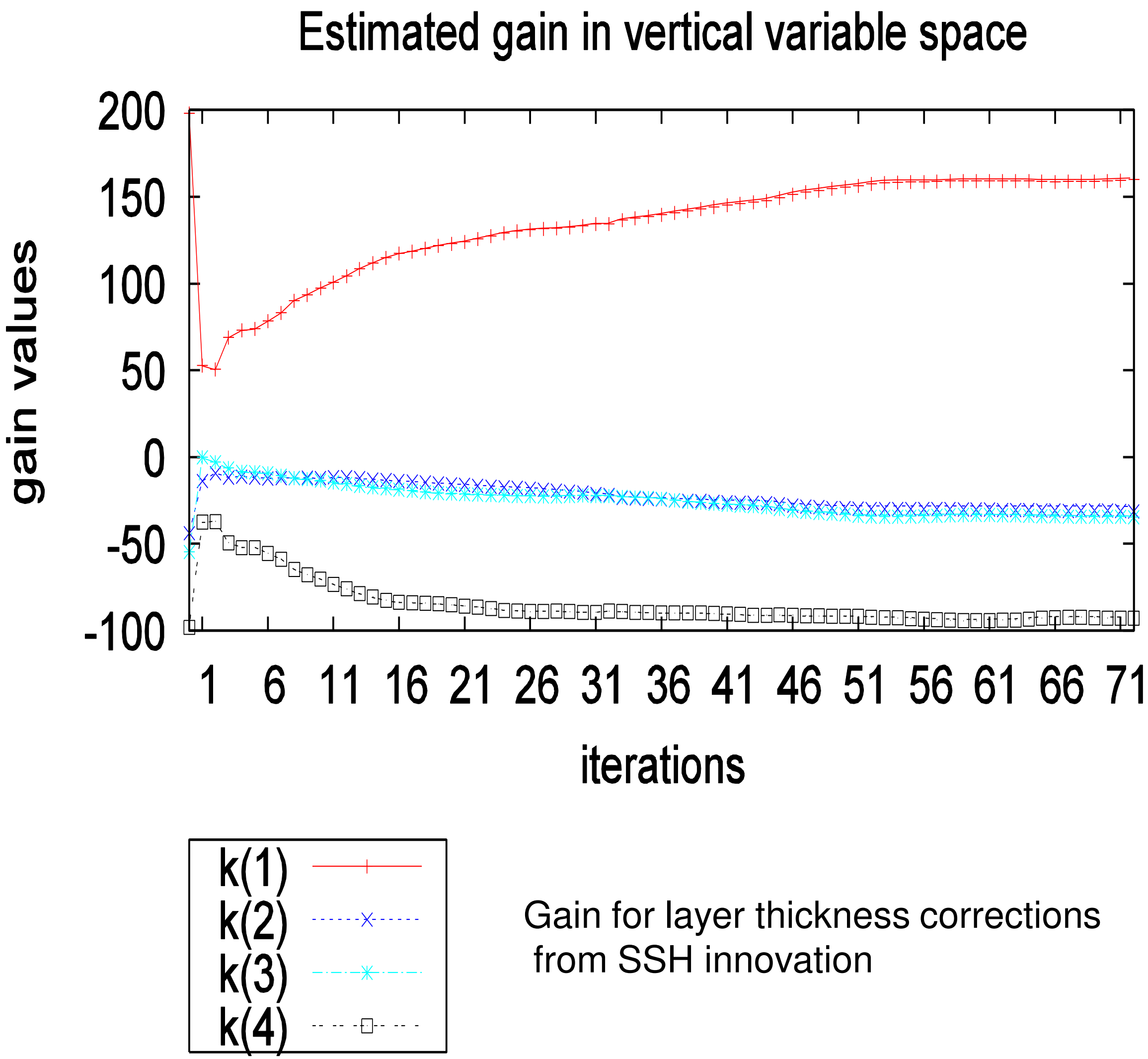
North Atlantics (30°N-60°N, 80°W-44°W)

(h,u,v), grid (140 x 180 x 4)

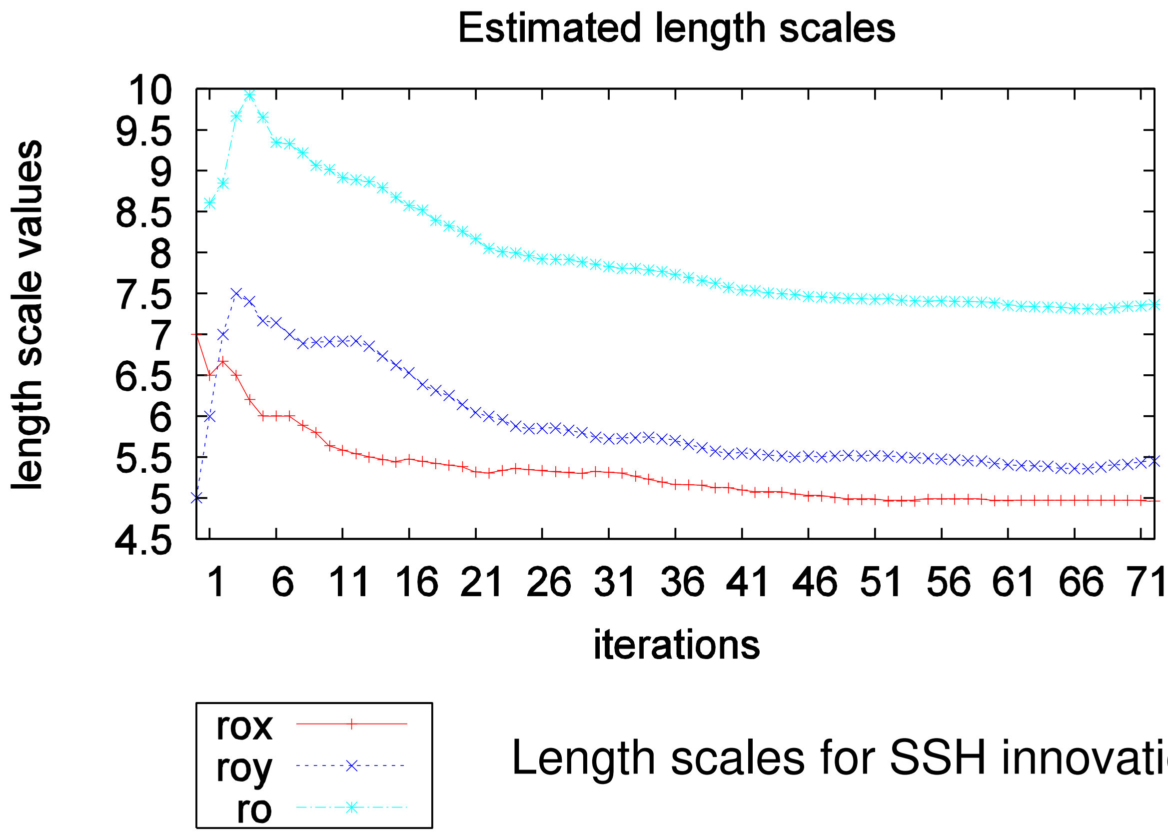
SSH Obs each 10 days

- MICOM adiabatique
- double gyre generated by symmetric wind
- jet is propagated to the Est
- tourbillons
- 4 layers (average layers: 440 m, 608 m, 978 m, 2974 m)
- β-plan (40°N)



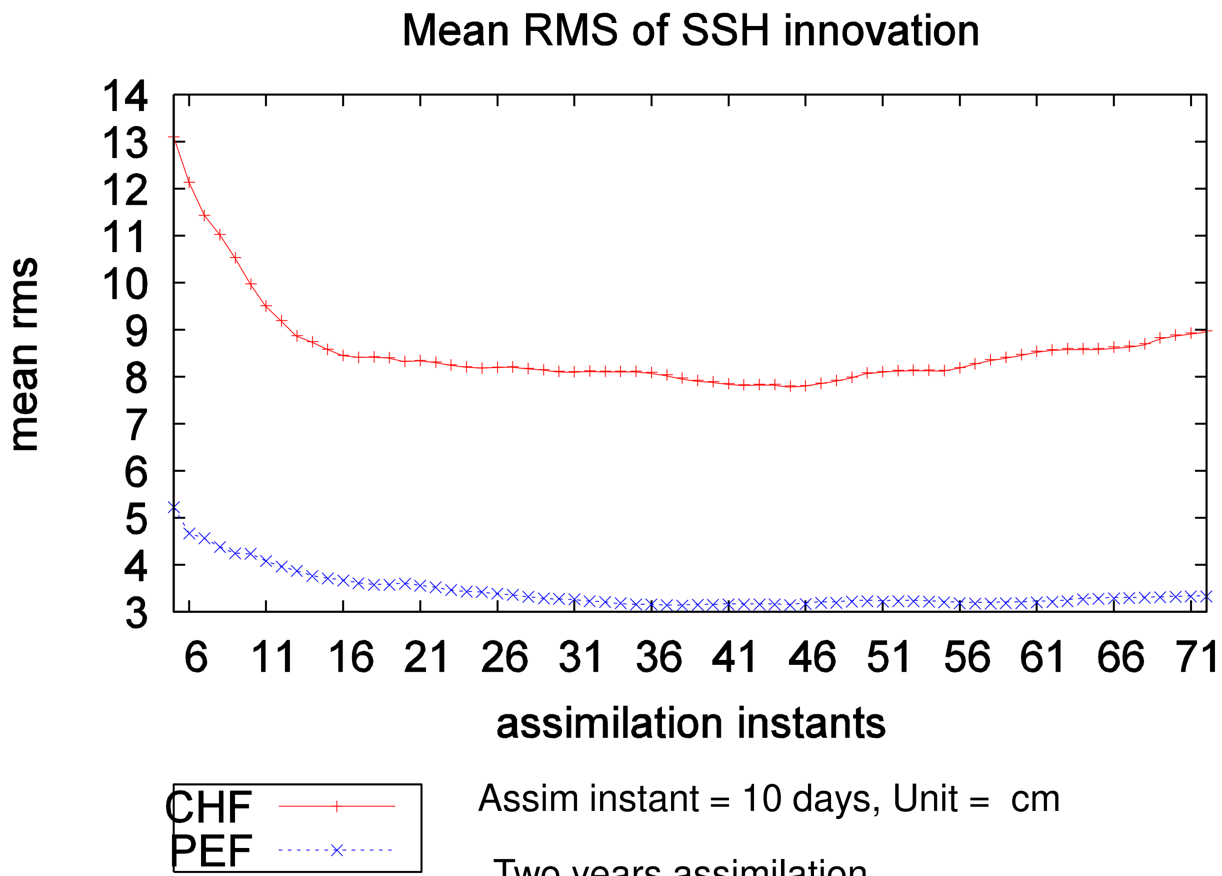


Gain for layer thickness corrections
from SSH innovation



Length scales for SSH innovation

Performance comparison: Sea surface Height



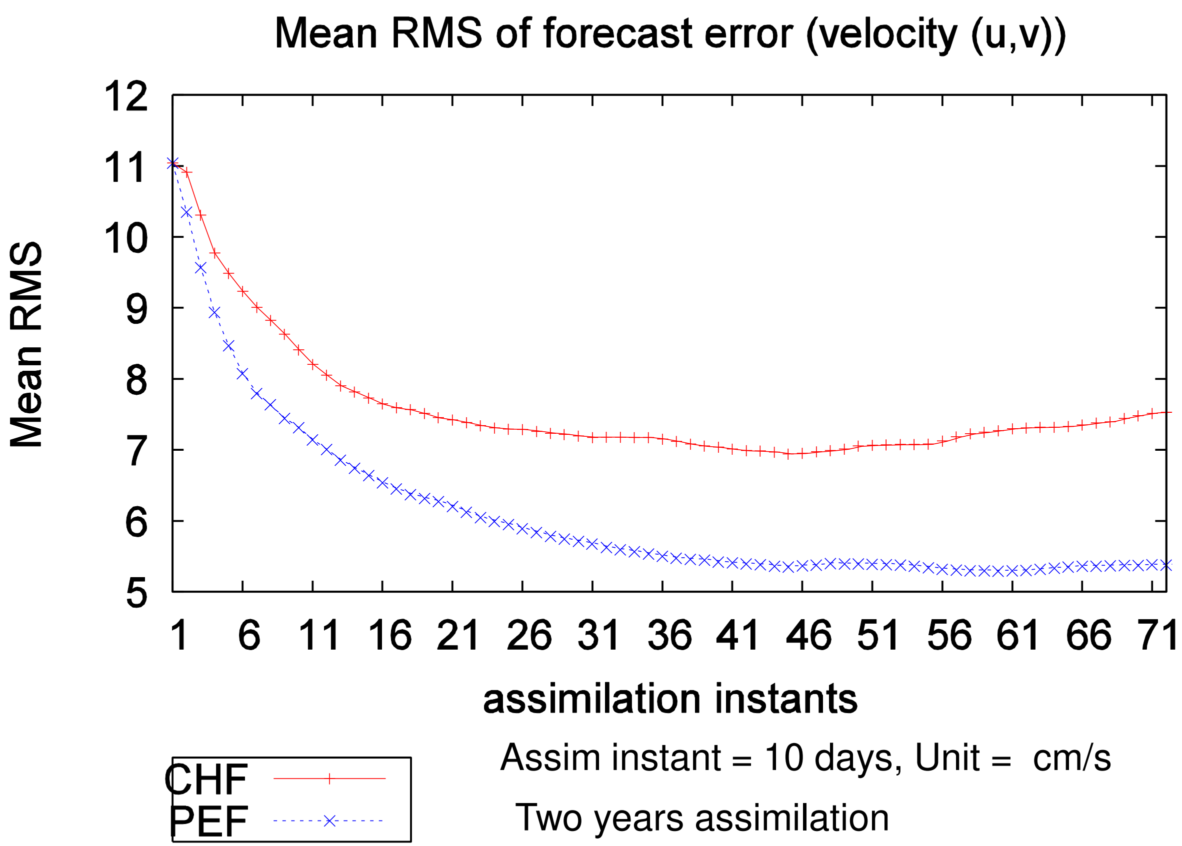
Assim instant = 10 days, Unit = cm

Two years assimilation

CHF – Cooper-Haines Filter; PEF – Prediction Error Filter

CHF is used in the Mercator prediction system

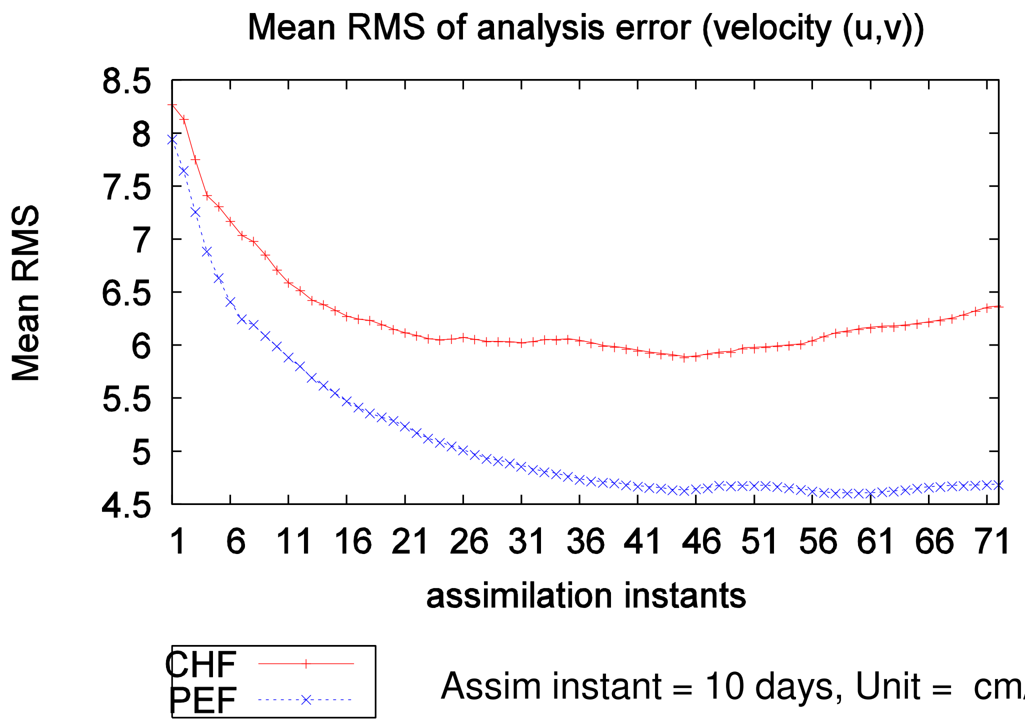
Performance comparision, velocity



CHF – Cooper-Haines Filter; PEF – Prediction Error Filter

CHF is used in the Mercator prediction system

Performance comparision, velocity



Assim instant = 10 days, Unit = cm/s

Two years assimilation

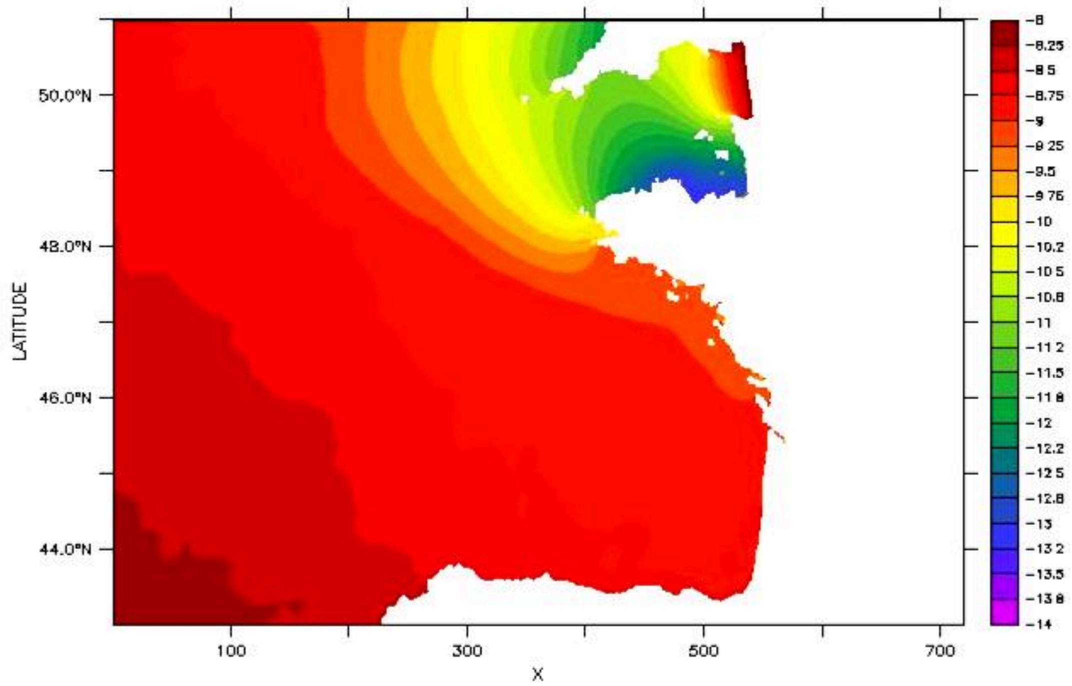
CHF – Cooper-Haines Filter; PEF – Prediction Error Filter

CHF is used in the Mercator prediction system

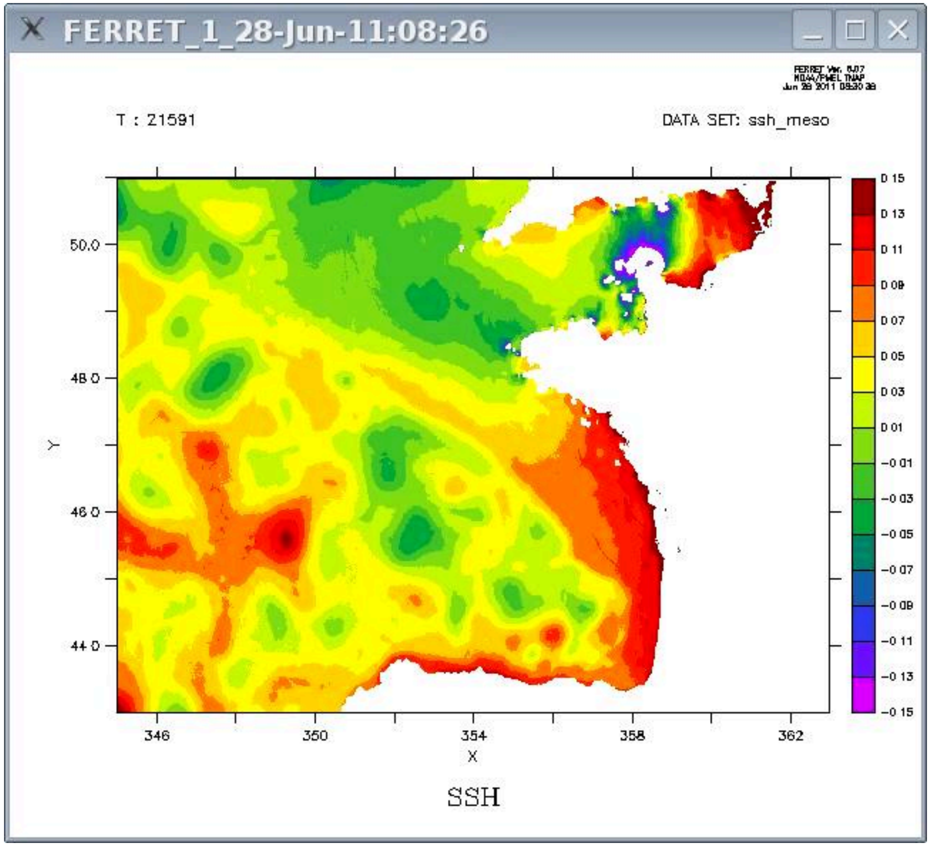
HYCOM Assimilation (SSH+SST+insitu (T+S))
(HYbrid Coordinate Ocean Model)

- Z-coordinate near surface
- Isopycnal coordinate in the deep ocean
- Terrain following near the bottom
- Variables = (h,u,v,T,S)
- Grid = (720 x 471 x 40)
- 1,8 km resolution
- Forced: boundary mesoscale from basin model MERCATOR
- Tides
- Real ECM atmospheric forcing fields at surface
- Regional model of the Bay Biscay

Forecast SSH – 28/06/ 2010 – Tides + mesoscale

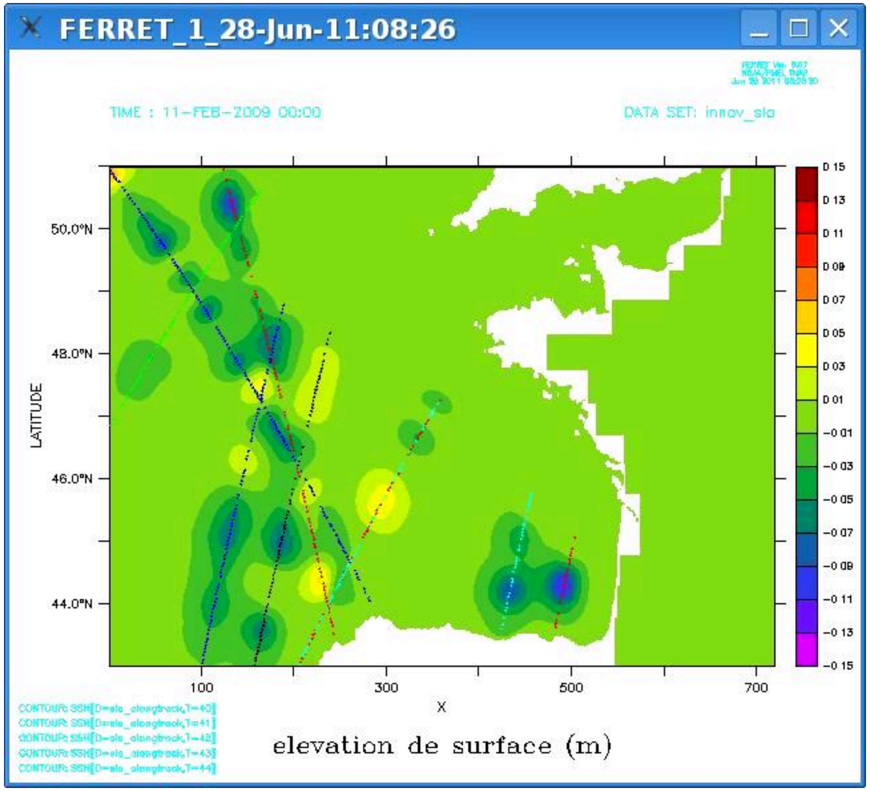


SSH – 28/06/2010 – mesoscale:
SSH forecast obtained are filtering tide

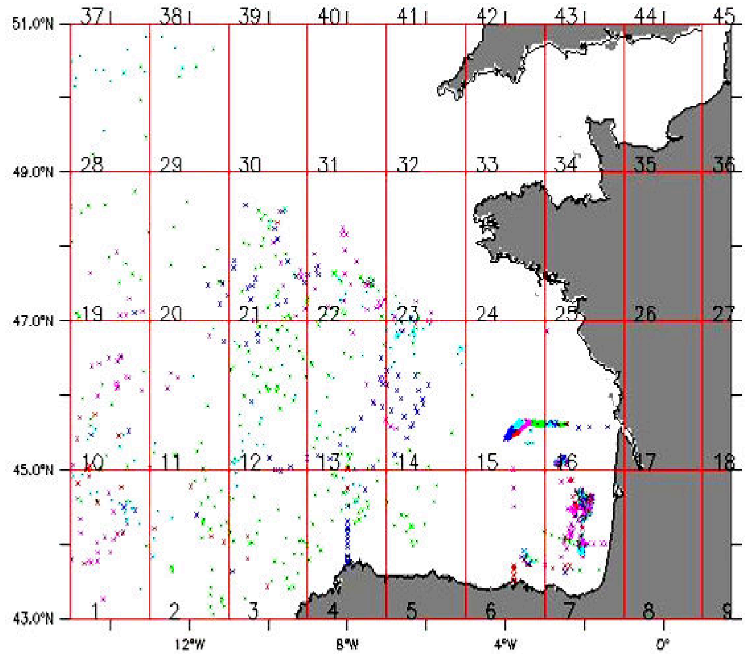


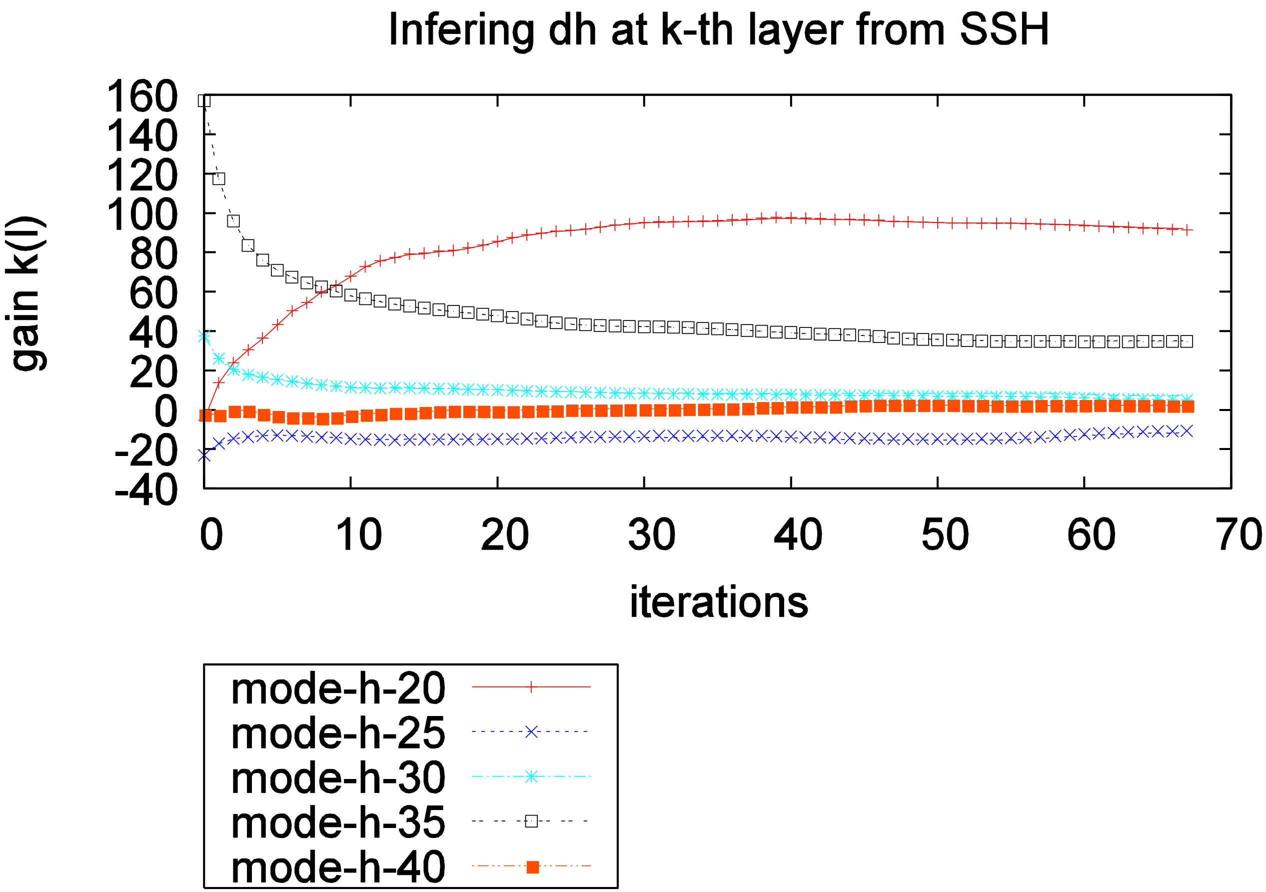
SSH – 28/06/2010 –

altimetry tracks and interpolated values

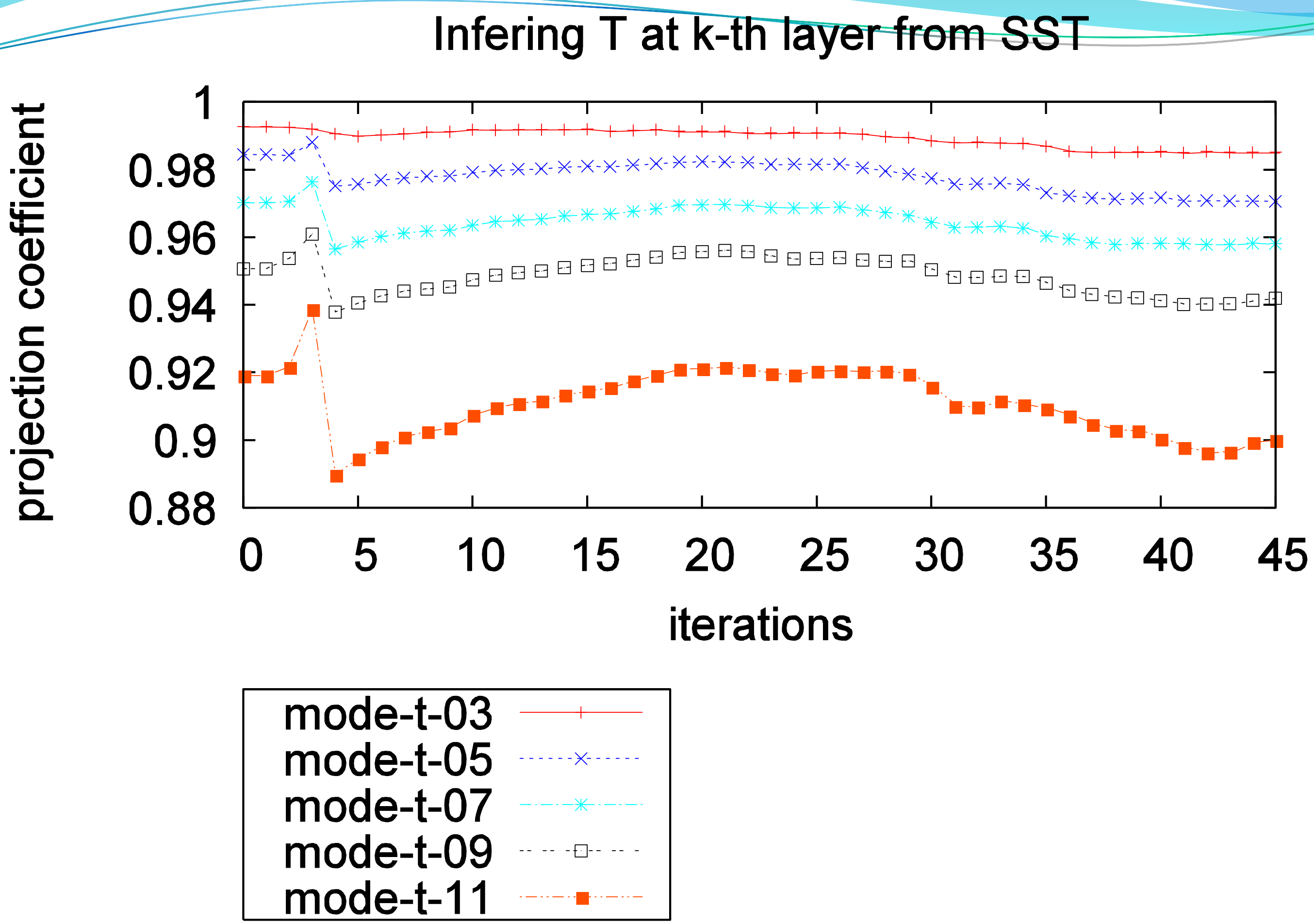


In-situ observation locations from 0 to 2000m, T + S, 2010



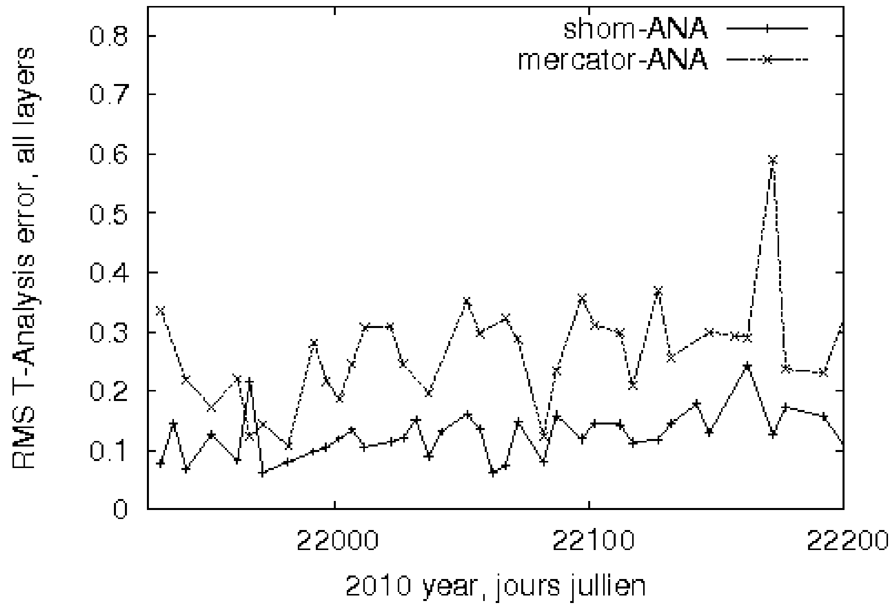


Gain coefficients for SSH produced by Vertical covariances using Schur vectors and SeVHS hypothesis



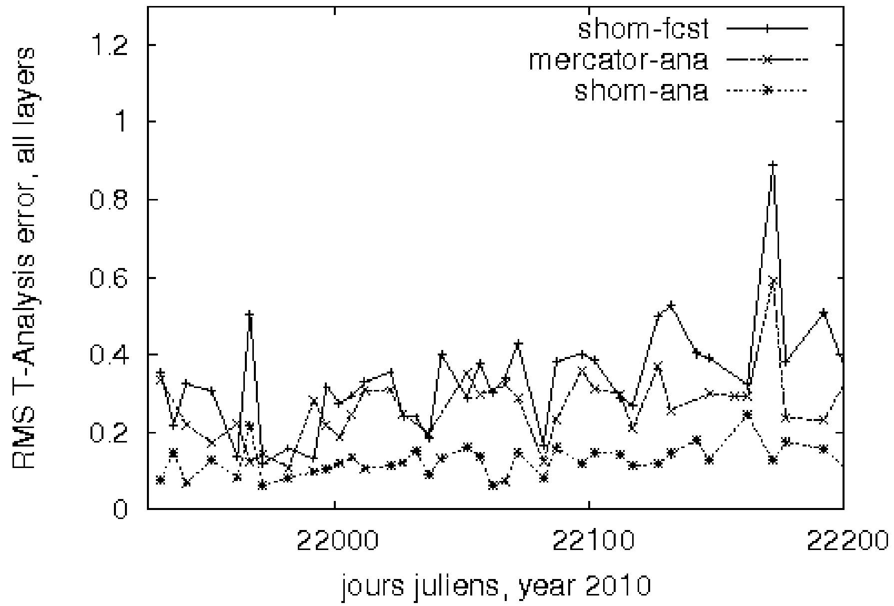
Gain coefficients for infering T at different layers from SST produced by Vertical covariances using Schur vectors and SeVHS hypothesis

Comparison of two assimilation systems
SHOM-MERCATOR
(2010, each 5ds)



Performance of Shom and Mercator assimilation systems. RMS is calculated at all layers, from difference between obs profile (T) and analysis estimates

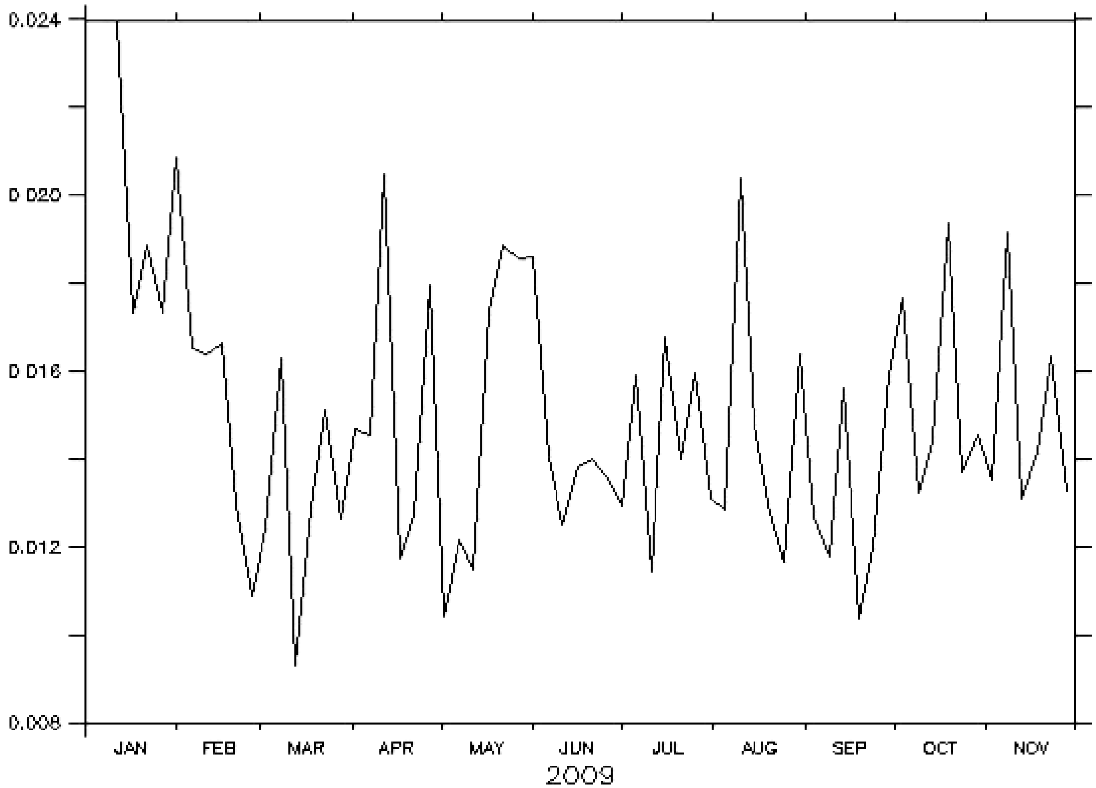
Comparison of two assimilation systems
SHOM-MERCATOR
(2010, each 5 days)



Performance of SHOM and MERCATOR assimilation systems. RMS is calculated at all layers, from difference between obs profile and analysis (forecast) estimates

RMS of SSH innovation
year 2010 – 5 day under track SSH

(in m)



Parameter identification

SHOM-HYCOM experiment (M. Boutet, SHOM, PhD student)

Estimation of friction coefficient in a simplified HYCOM barotropic version (regional model Bay Biscay; Fig. 2).

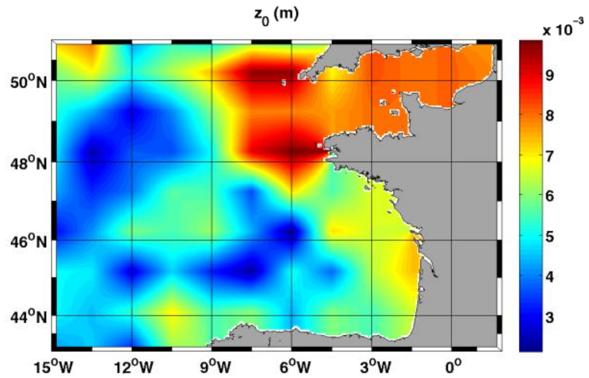


Fig. 1. Estimation of friction coeffcient: (1) true coeffcient z0 = 8mm; (2) One tide wave M2 used; (3) 2000 iterations (SPSA algorithm). The coeffcient is well estimated in near coast region. For far coast region there is a difficulty to well it estimate (certaintly due to insensitivity of this region to friction coeffcient; to be confirmed by a future sensitivity study).

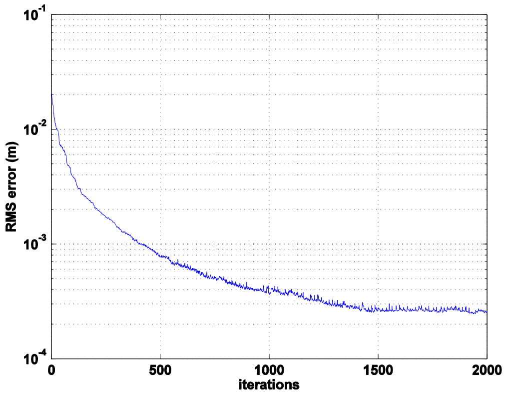


Fig. 2. Value of the sample cost function as iteration progress; Optimization by SPSA algorithm.

Conclusions

- **PIOM, SSIR, MPE of innovation, SPSA**
- **Filter stability: dominant EVs, SVs, ScVs;**
- **Control vector: gain parameters**
- **Reduced-order filter;**

Conclusions (continue)

Hypothesis : SeVHS

- **Estimation of vertical Cov and some parameters of horizontal Cov (correlation length)**
- **Optimization: Frobenius norm of difference between true Cov and Cov with SeVHS**
- **True Cov is obtained from Dominant Schur vectors of the system dynamics**
- **Lorenz system: sensitivity analysis, assimilation;**
- **MICOM: Twin-Experiment with SSH observations**
- **HYCOM: Real obs (SSH, SST, insitu (T,S))**