THE DIFFUSION KERNEL IN PREDICTION PROBLEMS

PAUL KRAUSE
Math. Arizona

8^{th} ADJOINT WORKSHOP
Tannersville, PA
May, 18-22, 2009

THANKS TO ORGANIZERS

(1) GLANCE:

working with

- NON-GAUSSIAN FILTERING methods for
- continuous-time deterministic models
- discrete-time noisy data

$$
\begin{aligned}
\text{Model}: &\quad \frac{d}{dt}\Phi = f(\Phi) \\
\text{Initial}: &\quad \Phi_0 = \text{random} \\
\text{Data}: &\quad y_k = h(\Phi_{t_k}) + \text{noise}
\end{aligned}
$$

issue is

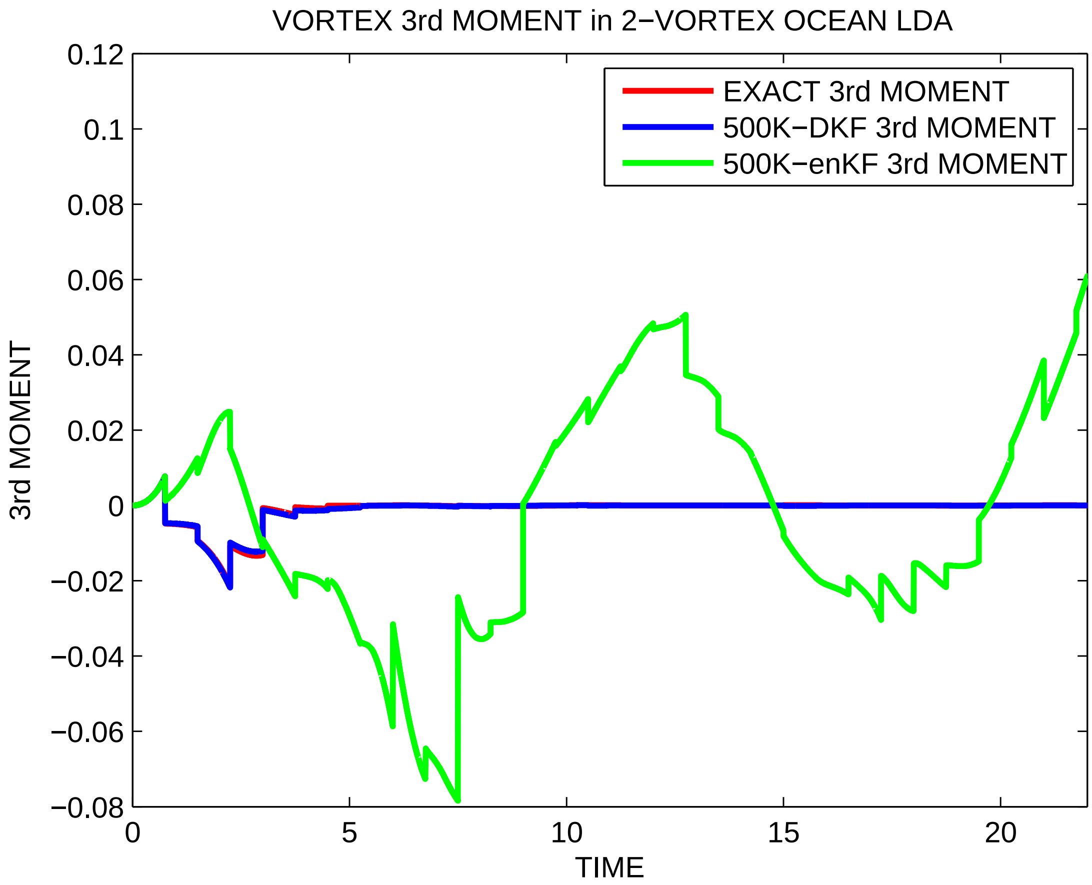
- COST effectiveness

while preserving

- MATH consistency
- PHYS consistency

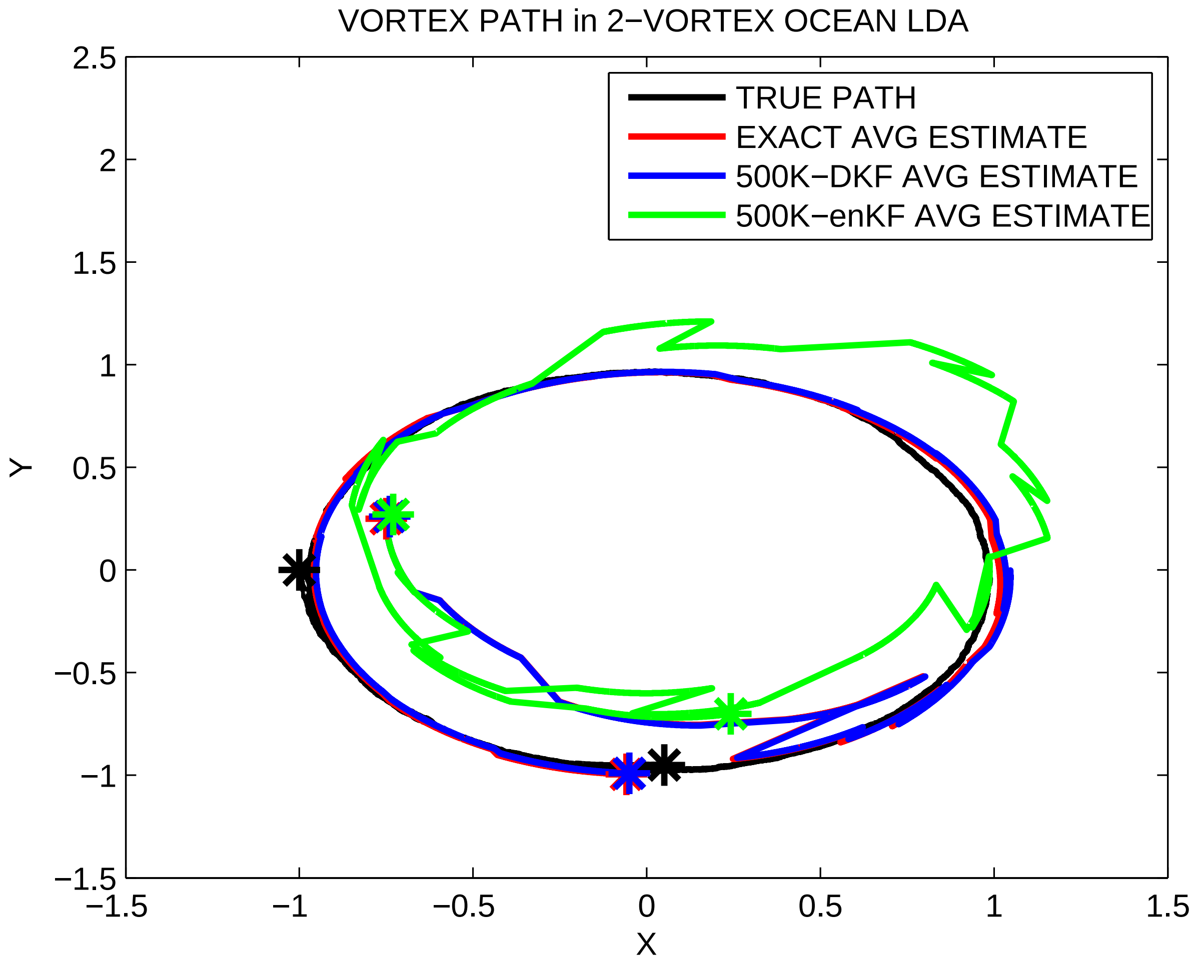
- MATH consistency

Are we closely solving for $\Phi|y$?



Reference:
The Diffusion Kernel Filter applied to Lagrangian Data Assimilation; w/ J.M. Restrepo; MWR, (2009)

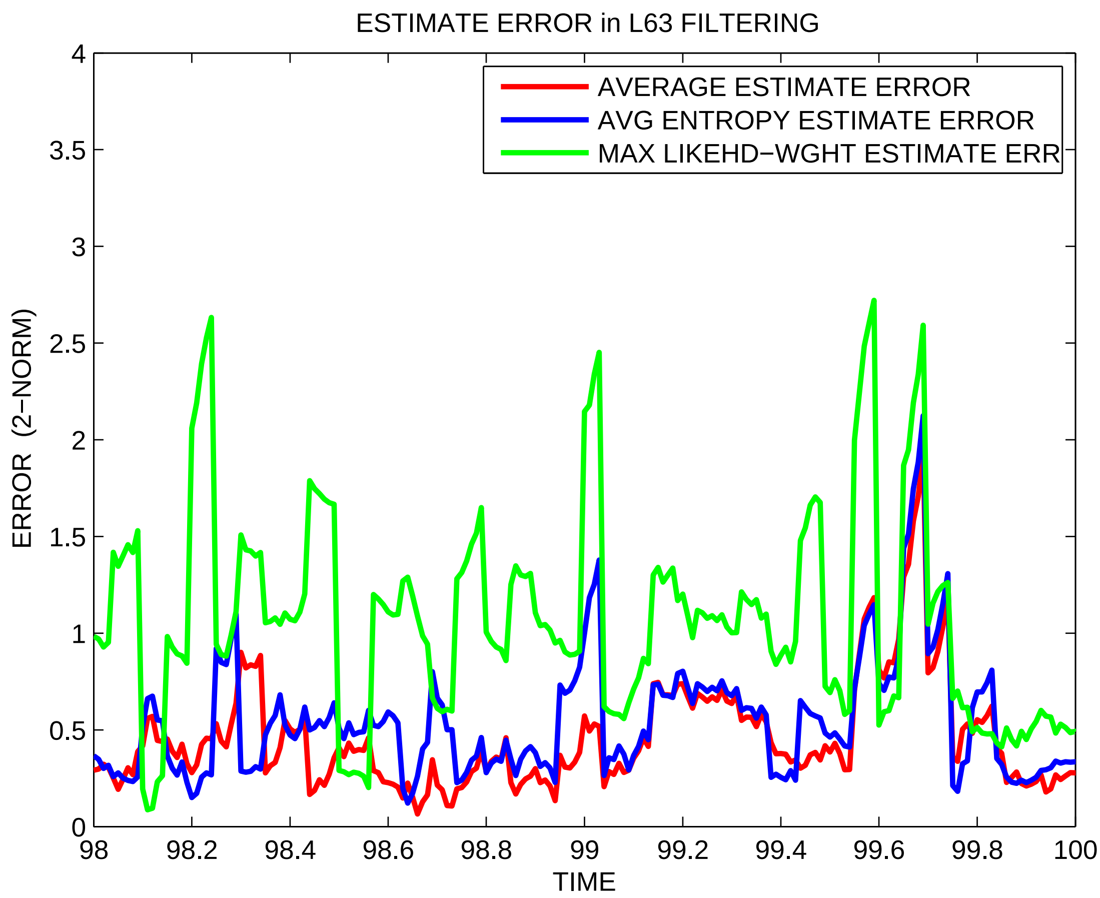
Is that needed?



Reference:
The Diffusion Kernel Filter applied to Lagrangian Data Assimilation; w/ J.M. Restrepo; MWR, (2009)

- PHYS consistency

Are we closely sampling from the deterministic model dynamics ?



Reference:
The Diffusion Kernel Filter; J. Stat. Phy., 134:2 (2009), pp. 365-380

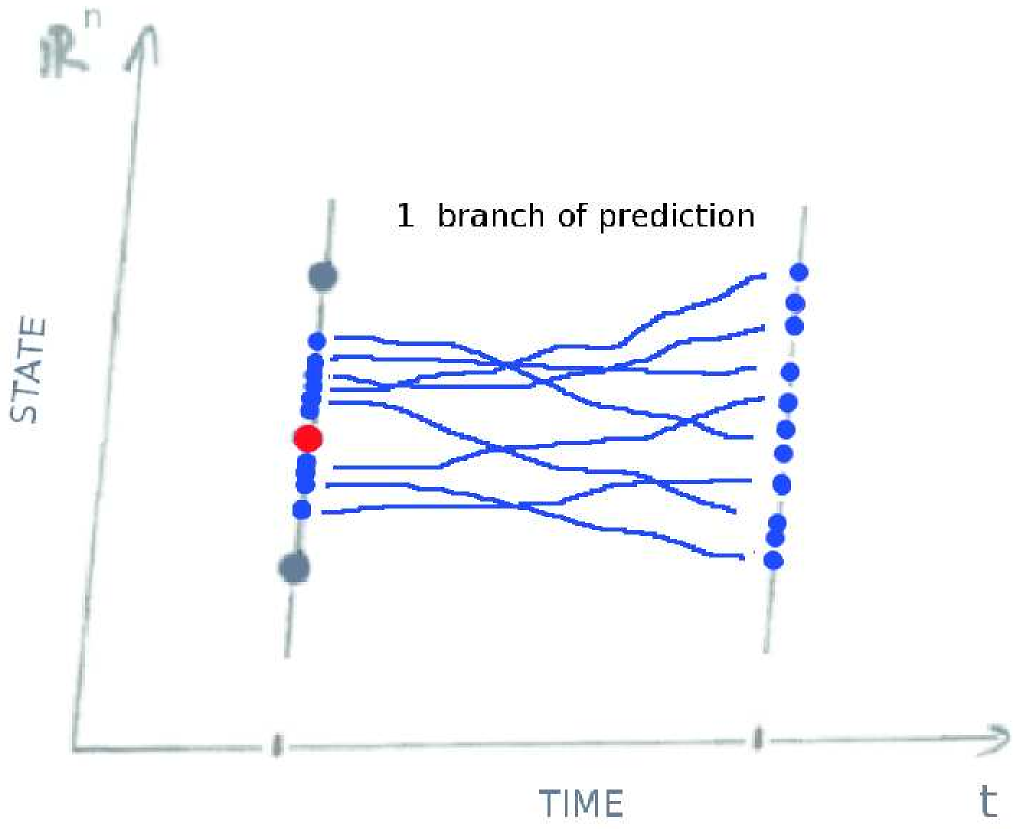
(2) METHOD:

- STEP 1: cast non-Gaussian initial uncertainty into Gaussian mixture of initial uncertainties

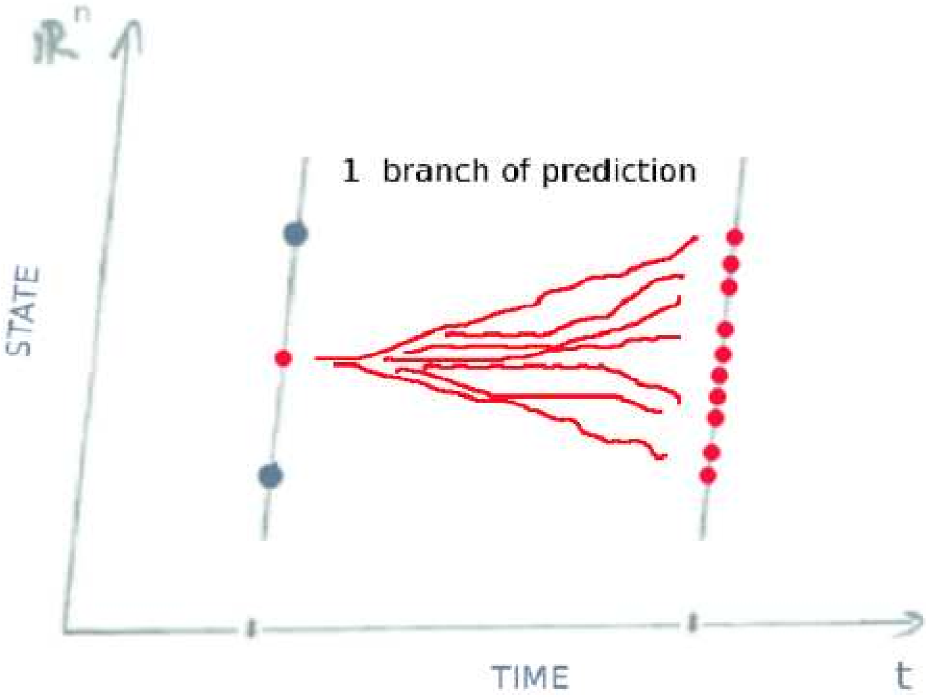
- **STEP 2: cast Gaussian initial uncertainty into model noise and solve the filtering problem for Gaussian branches with Kalman analysis**

- STEP 2': parallelize reduced Tangent Operator-based forecasts, not sample-based forecasts

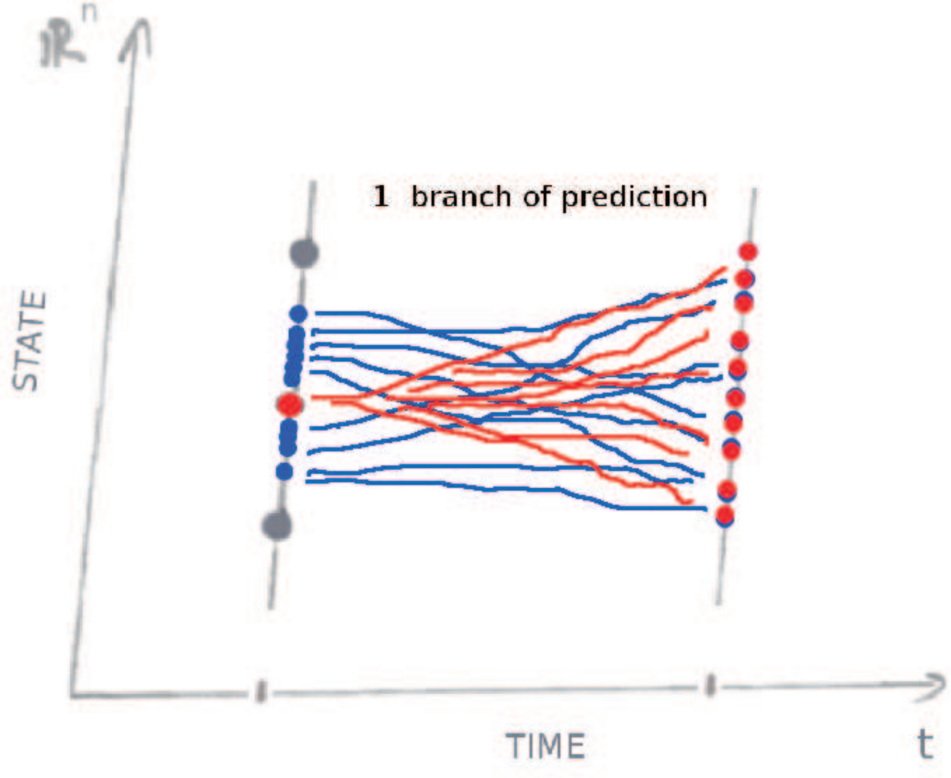
- **STEP 2: cast Gaussian initial uncertainty into model noise**



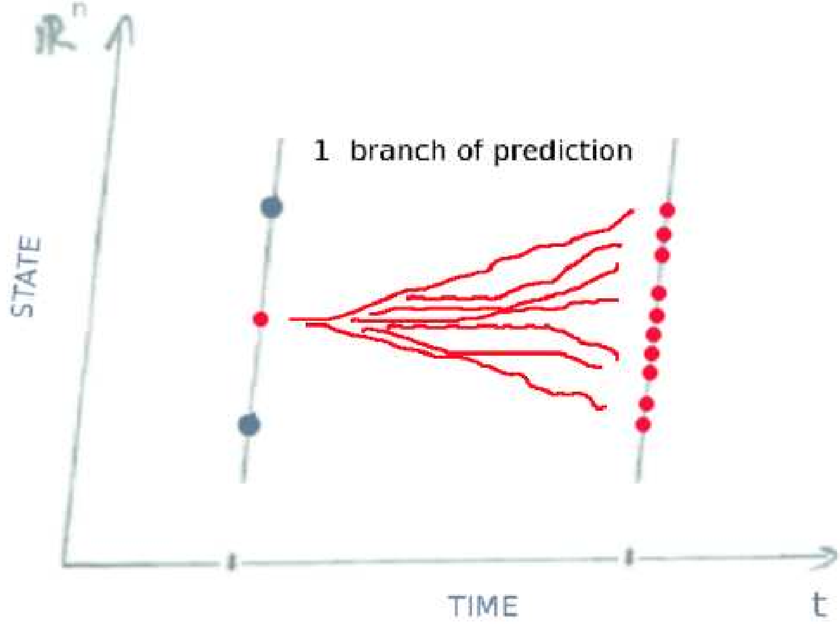
$$\frac{d}{dt}\Phi = f(\Phi)$$
$$\Phi_0 \sim N(\phi_0, C_0)$$



$$d\Phi = f(\Phi)\, dt + g\, dw$$
$$(dw/dt = \text{white})$$
$$\Phi_0 \sim Dirac(\phi_0)$$



(in distribution)

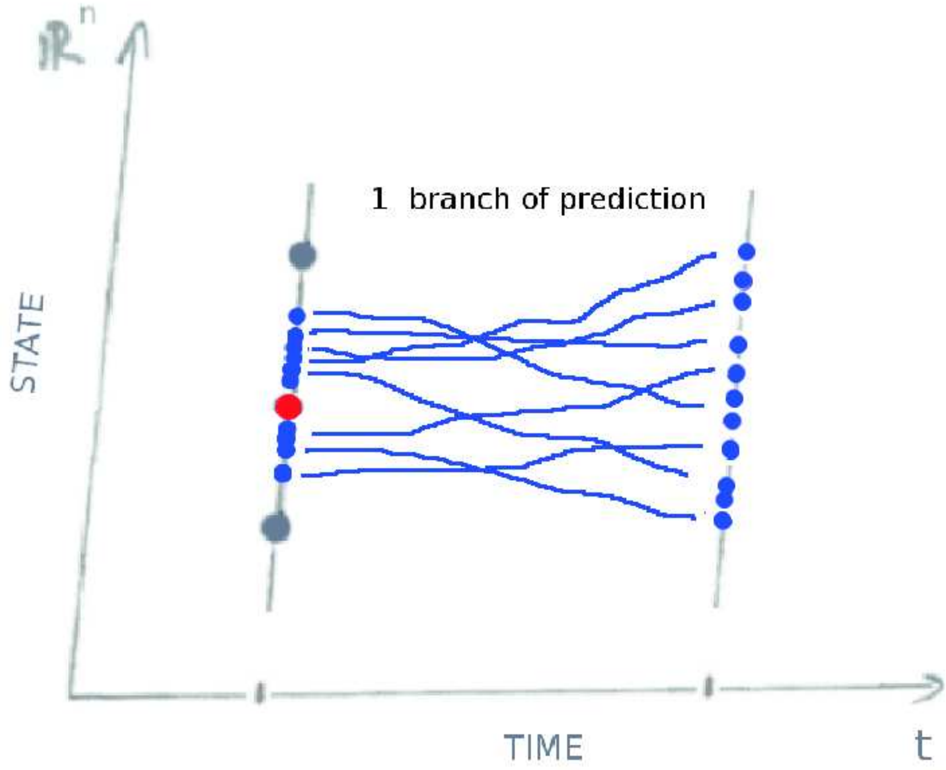


RESULT: linear regime

$$P_t = \mathrm{M}_t (t g g^{\mathrm{T}}) \mathrm{M}_t^{\mathrm{T}}, \text{ where}$$

M_t := Tangent Operator obtained from TLM about ϕ_t

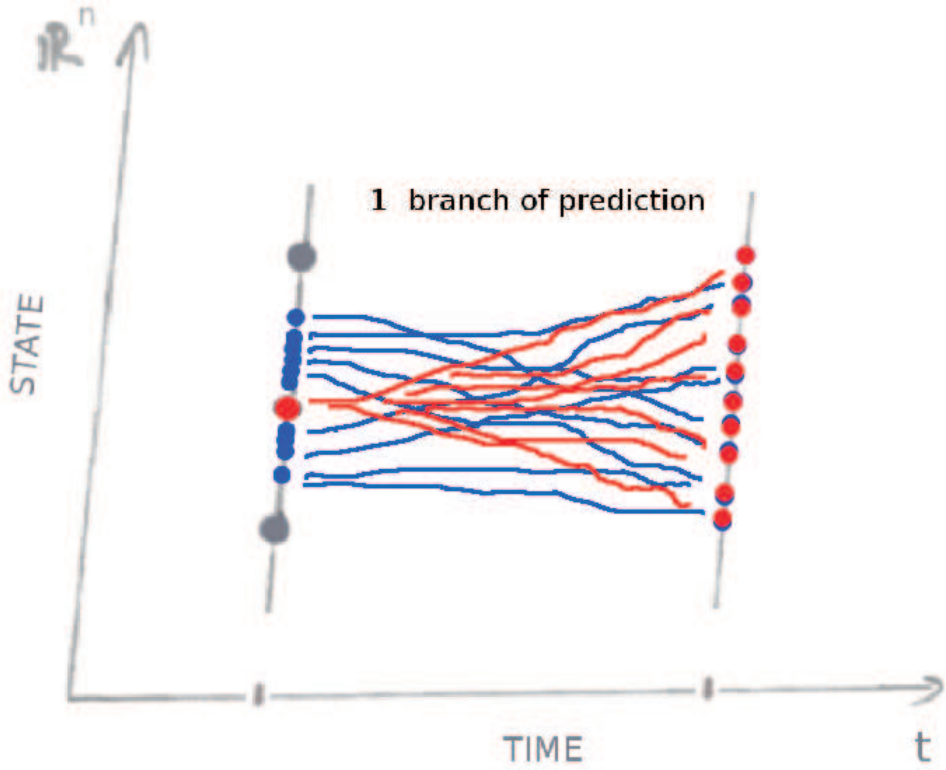
Reference:
The Diffusion Kernel Filter; J. Stat. Phy., 134:2 (2009), pp. 365-380

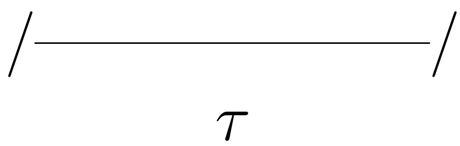


SINCE: linear regime

$$C_t = \mathrm{M}_t C_0 \mathrm{M}_t^{\mathrm{T}}, \text{ where}$$

$\mathrm{M}_t :=$ Tangent Operator obtained from TLM about ϕ_t





$$C_\tau = P_\tau \text{ for } gg^{\mathrm{T}} := \frac{C_0}{\tau}$$

ANSATZ: REDUCED FORMULA

$$\widehat{C}_t := \widehat{\mathrm{M}}_t(t g_e(t) g_e(t)^{\mathrm{T}})\widehat{\mathrm{M}}_t^{\mathrm{T}}, \text{ where}$$

$$g_e(t) := (\textstyle\int_{t_k}^{t} D\hat{f}(\phi_s)ds)\, g + \hat{g}$$

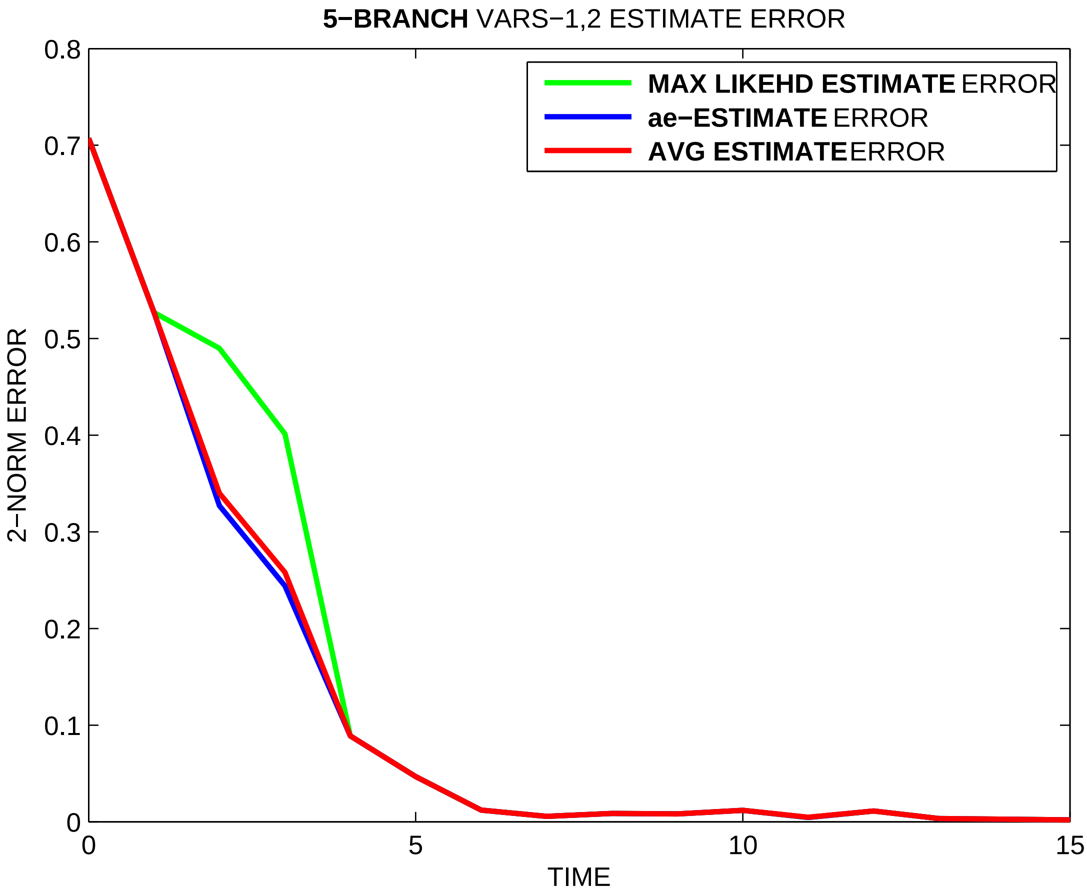
- g_e models the effect of the unresolved components of C_t onto $\widehat{C}_t$
- $\tau g_e(\tau) g_e(\tau)$ at filtering time t_{k+1} involves $\tau g g^{\mathrm{T}} = \mathrm{diag}(\widehat{C}_k, \widetilde{C}_k)$, where $\widetilde{C}_k$ is the value at time t_k of the unresolved block of C_t within the prediction interval $[t_k, t_{k+1}]$
- $\widetilde{C}_k$ would be roughly estimated on a fast side process. Here, it is set to $\widetilde{R}$ (w/ the corresponding analysis vars set to data values, in all branches).

(3) TESTS:

Lorenz-63 flows with 1 time-unit long prediction steps and observation function $h : \mathbb{R}\times\mathbb{R}\times\mathbb{R} \to \mathbb{R}\times\mathbb{R}$, $h(\Phi_1, \Phi_2, \Phi_3) = (\Phi_1^2 + \Phi_2^2, \Phi_3)$, with error $\epsilon \sim N(0, \mathrm{diag}([10^{-2}, 10^{-4}]))$

STEP 1: launch $\widehat{C}_0$ with different caliber for different branches

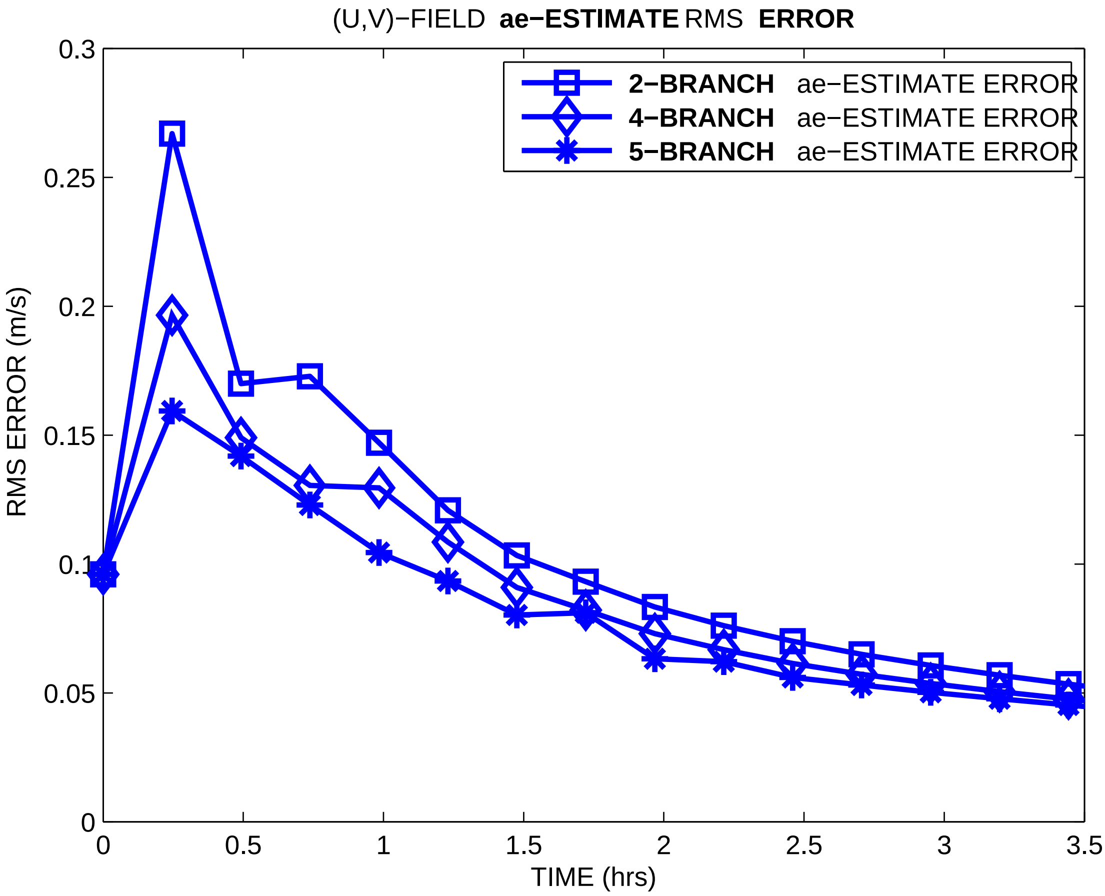
STEP 2: TLM is solved just for the first two columns of M_t

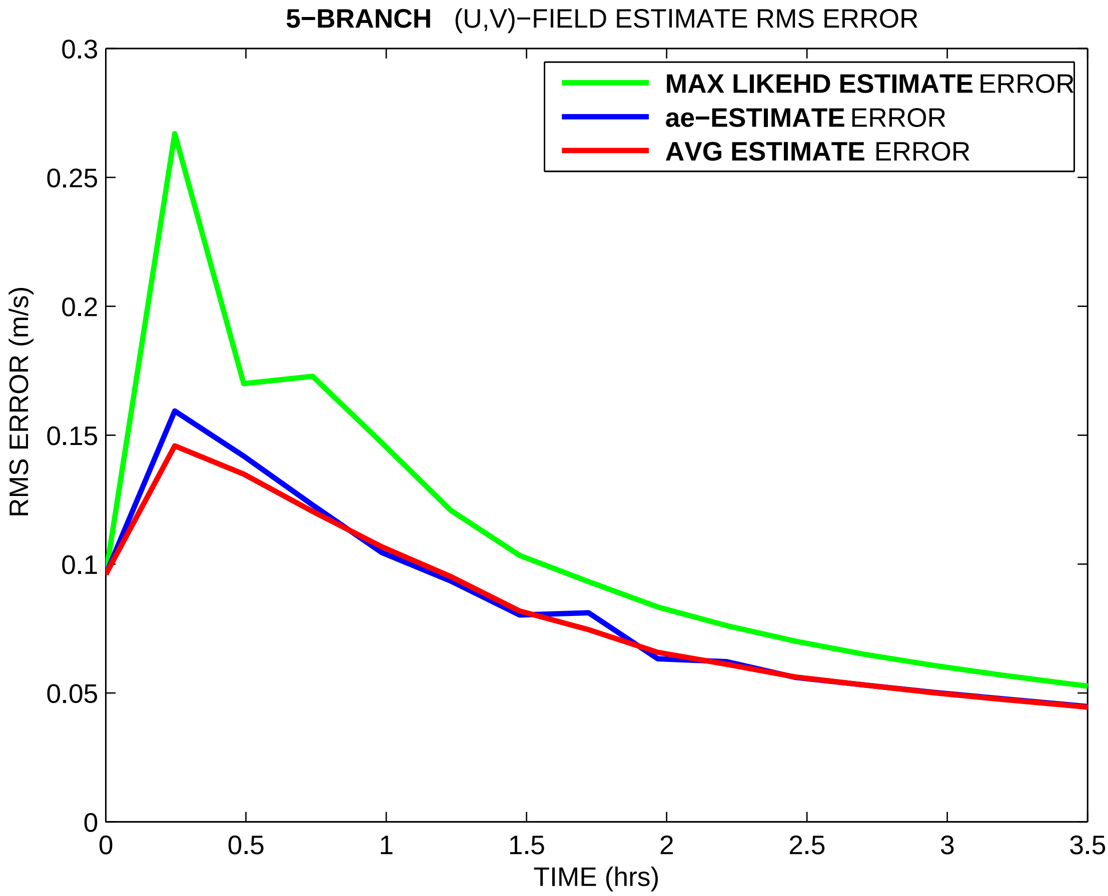

5−BRANCH VARS−1,2 ESTIMATE ERROR
MAX LIKEHD ESTIMATE ERROR
ae−ESTIMATE ERROR
AVG ESTIMATE ERROR
2−NORM ERROR
TIME
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
5
10
15

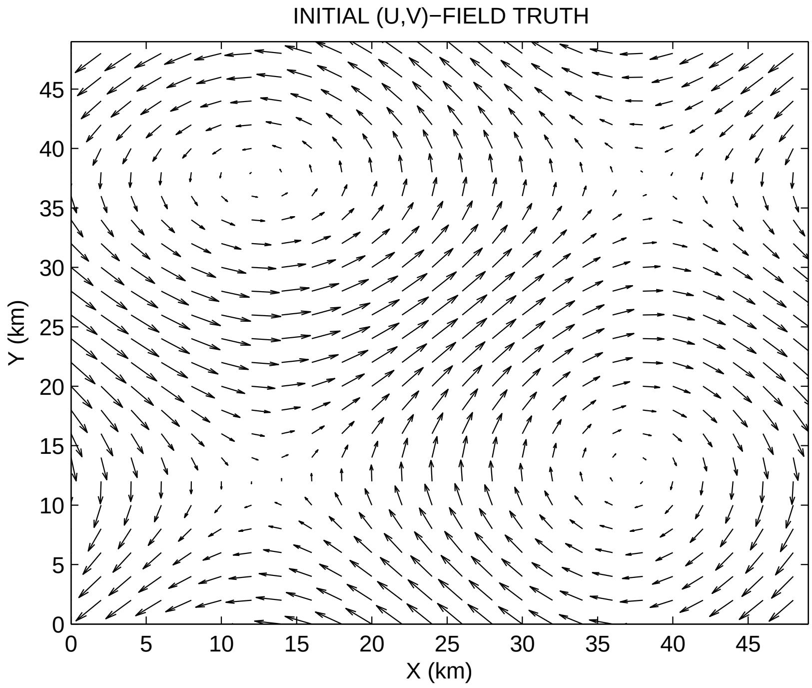
2D periodic rotating shallow water flows with 15 mins long prediction steps on a 50 km $\times$ 50 km domain covered by a 50×50 grid with observation function $h : \mathbb{R}^{50\cdot 50} \times \mathbb{R}^{50\cdot 50} \times \mathbb{R}^{50\cdot 50} \rightarrow \mathbb{R}^{50\cdot 50} \times \mathbb{R}^{50\cdot 50}$, $h(u, v, \eta) = (u^2 + v^2, \eta)$, with error $\epsilon \sim N(0, \mathrm{diag}(10^{-4}))$ (similar results were obtained with 1 hr long prediction steps on a 70×70 grid)

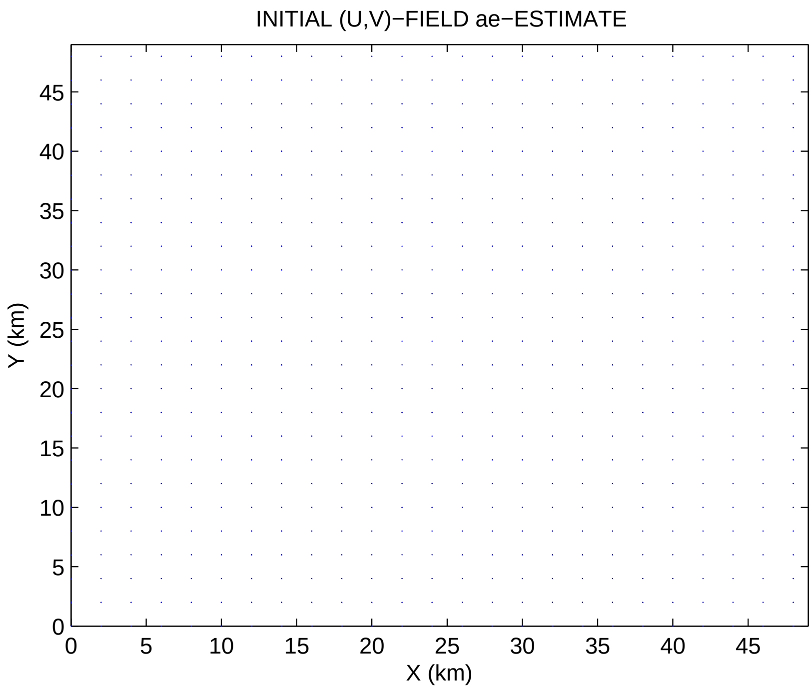
STEP 1: launch $\widehat{C}_0$ with different caliber for different branches

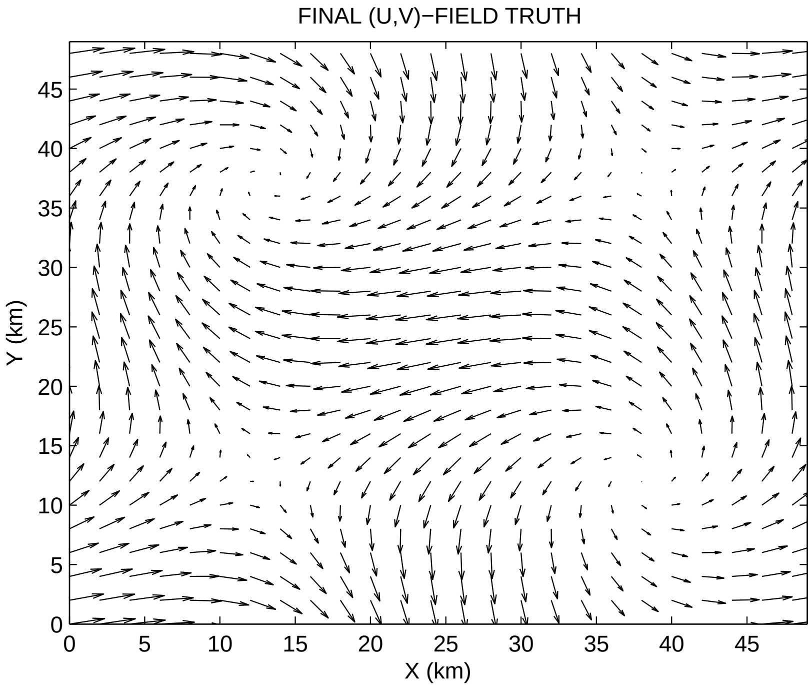
STEP 2: TLM is solved just for the u, v columns of M_t

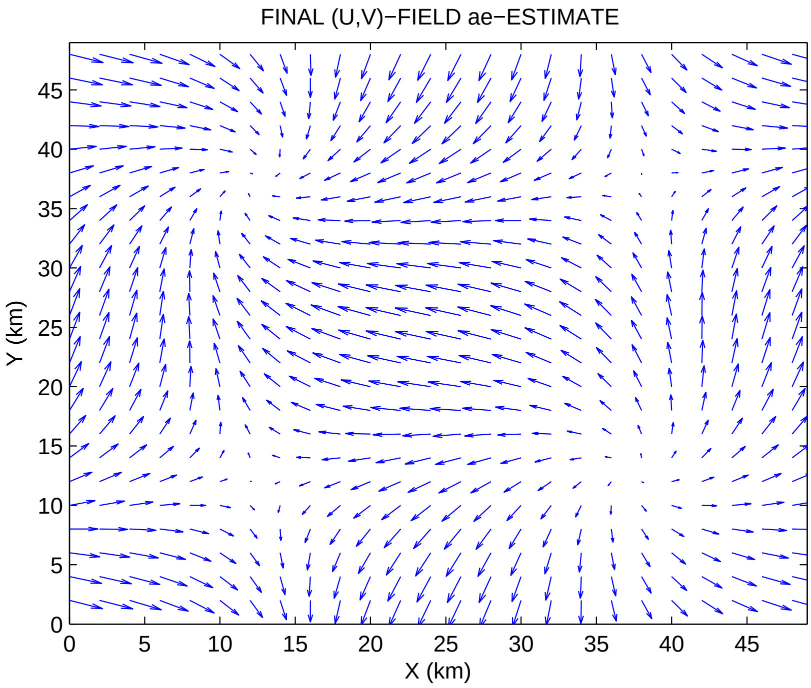

(U,V)−FIELD **ae−ESTIMATE** RMS **ERROR**
2−BRANCH ae−ESTIMATE ERROR
4−BRANCH ae−ESTIMATE ERROR
5−BRANCH ae−ESTIMATE ERROR
RMS ERROR (m/s)
TIME (hrs)
0
0.05
0.1
0.15
0.2
0.25
0.3
0.5
1
1.5
2
2.5
3
3.5

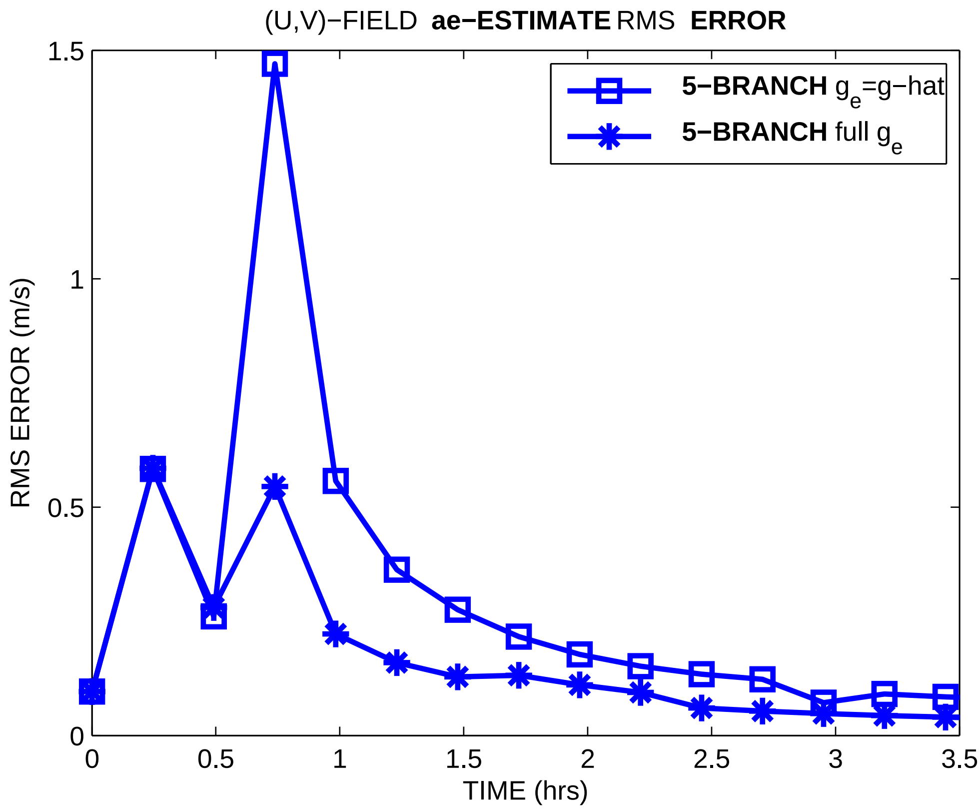

5−BRANCH (U,V)−FIELD ESTIMATE RMS ERROR
MAX LIKEHD ESTIMATE ERROR
ae−ESTIMATE ERROR
AVG ESTIMATE ERROR
RMS ERROR (m/s)
TIME (hrs)
0
0.05
0.1
0.15
0.2
0.25
0.3
0
0.5
1
1.5
2
2.5
3
3.5


INITIAL (U,V)−FIELD TRUTH
Y (km)
X (km)
0
5
10
15
20
25
30
35
40
45


INITIAL (U,V)−FIELD ae−ESTIMATE
Y (km)
X (km)
0
5
10
15
20
25
30
35
40
45


FINAL (U,V)−FIELD TRUTH
Y (km)
X (km)
0
5
10
15
20
25
30
35
40
45


FINAL (U,V)−FIELD ae−ESTIMATE
Y (km)
X (km)
0
5
10
15
20
25
30
35
40
45



(U,V)−FIELD **ae−ESTIMATE** RMS **ERROR**
5−BRANCH g_e=g−hat
5−BRANCH full g_e
RMS ERROR (m/s)
TIME (hrs)
0
0.5
1
1.5
2
2.5
3
3.5

(4) CONCLUSION:

we've got

- NON-GAUSSIAN "**LINEAR**" FILTER
- COST effective? (reduced formula)
- MATH consistent (within settings where the formula applies)
- PHYS consistent (ae-estimate)

THANKS!