GPU Ocean: Efficient simplified ocean models and non-linear data assimilation

Håvard Heitlo Holm[1,2], havard.heitlo.holm@sintef.no

Joint work with André R. Brodtkorb[1,3], Knut-Andreas Lie[1,2], Martin L. Sætra[4], Kai H. Christensen[4], Göran Broström[4], Peter Jan van Leeuwen[5]

1) SINTEF Digital, 2) Norwegian University of Science and Technology, 3) Oslo Metropolitan University, 4) Norwegian Meteorological Institute, 5) University of Reading

Problem description

The predictions of ocean currents in a short time range are typically dominated by location of ocean eddies, which are rotating high and low pressure systems. Compared to their atmospheric counterparts, there are few direct measurements of these eddies, and because of their relative small sizes, a small misplacement of the eddies leads to large errors in predicted currents (see Figure 1). Because of the large complexity of traditional ocean models, it is not feasible to cover the substantial uncertainties through large ensemble simulations.

Our approach is to study the uncertainty in the ocean through large ensembles of simplified ocean models, which can be efficiently simulated using Graphical Processing Units (GPUs). We aim on assimilating available observations of the ocean into the ensemble through non-linear data assimilation methods such as particle filters.

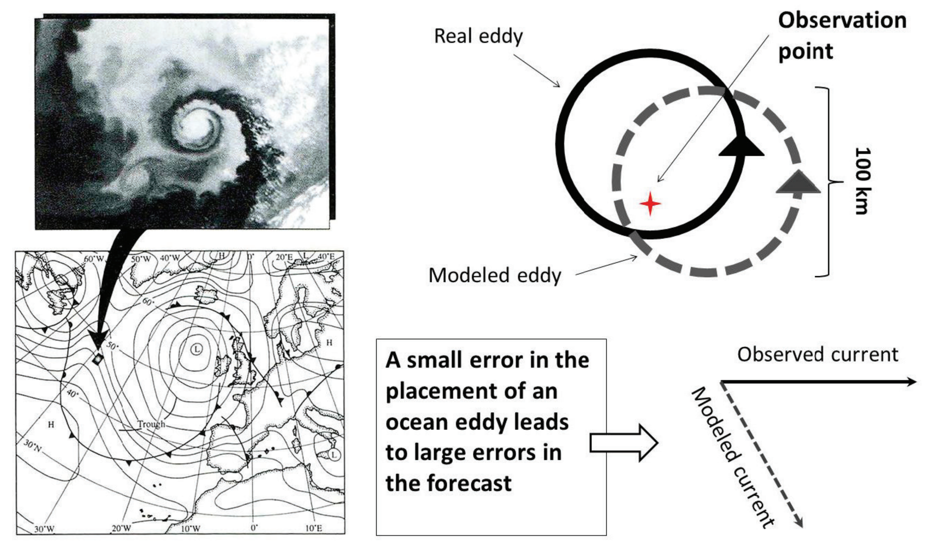

Figure 1: Small misplacements of ocean eddies can lead to a large error in predicted current

Perturbing ensembles of ocean models

As a simplified ocean model we consider the rotating shallow water equations

$$\begin{bmatrix} h \\ hu \\ hv \end{bmatrix}_t + \begin{bmatrix} hu \\ hu^2 + \frac{1}{2}gh^2 \\ huv \end{bmatrix}_x + \begin{bmatrix} hv \\ huv \\ hv^2 + \frac{1}{2}gh^2 \end{bmatrix}_y = \begin{bmatrix} 0 \\ fhv \\ -fhu \end{bmatrix} + \begin{bmatrix} 0 \\ ghH_x \\ ghH_y \end{bmatrix}.$$

Ocean depth, and volume transport in x- and y-direction are the conserved ocean state $\psi = [h, hu, hv]^T$. Further, g is the gravitational force, f is the Coriolis parameter, and $H(x,y)$ describes equilibrium depth. Surface elevation is $\eta = h - H$. These equations are solved with a suitable explicit numerical scheme [1], here denoted by the function $M(\psi)$.

At each time step, random additive model error $\beta \sim N(0, Q)$ is applied to the model, representing realizations of the error related to missing physics and discretization errors. The model evolution from time t_n to t_{n+1} is then given by

$$\psi_j^{n+1} = M(\psi_j^n) + \beta_j^n.$$

The covariance structure Q is based on local small scale perturbations applied in two steps, $Q^{1/2} = Q_{GB}^{1/2} Q_{SOAR}^{1/2}$. The SOAR operator ensures a smooth surface elevation, before $Q_{GB}^{1/2}$ adds momentum corresponding to geostrophic balance, see Figure 2. These operations are well-suited for the computational model found on GPUs, as illustrated by Figure 3.

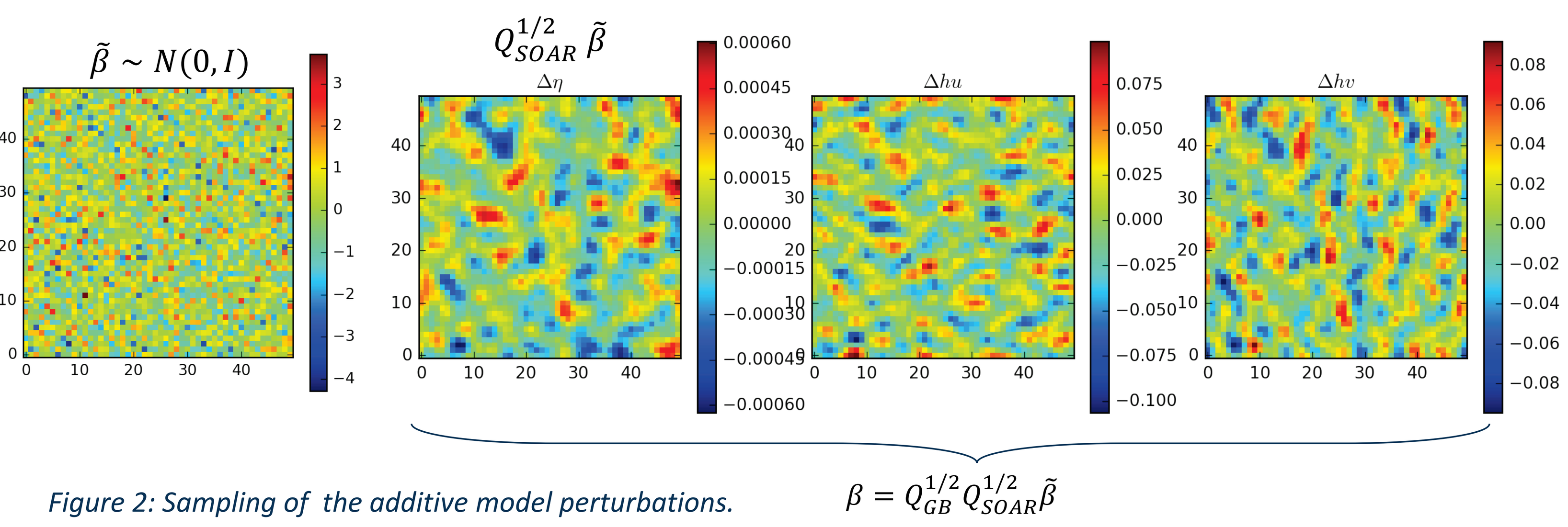

Figure 2: Sampling of the additive model perturbations.

Figure 3: Conceptual comparison of stencil computations and GPU hardware architectures

Observations

Currently, we consider observations of the ocean current at a drifter position (x_d, y_d) at time t_n as

$$y^n = [u(x_d, y_d), v(x_d, y_d)]^T.$$

This gives us sparsely distributed point measurements of the system.

Other types of observations that we are interested in, but are currently not using, are:

- High Frequency (HF) Radar (see Figure 4)
- Synthetic-aperture radar (SAR) satellite data
- Satellite altimetry
- Acoustic profile

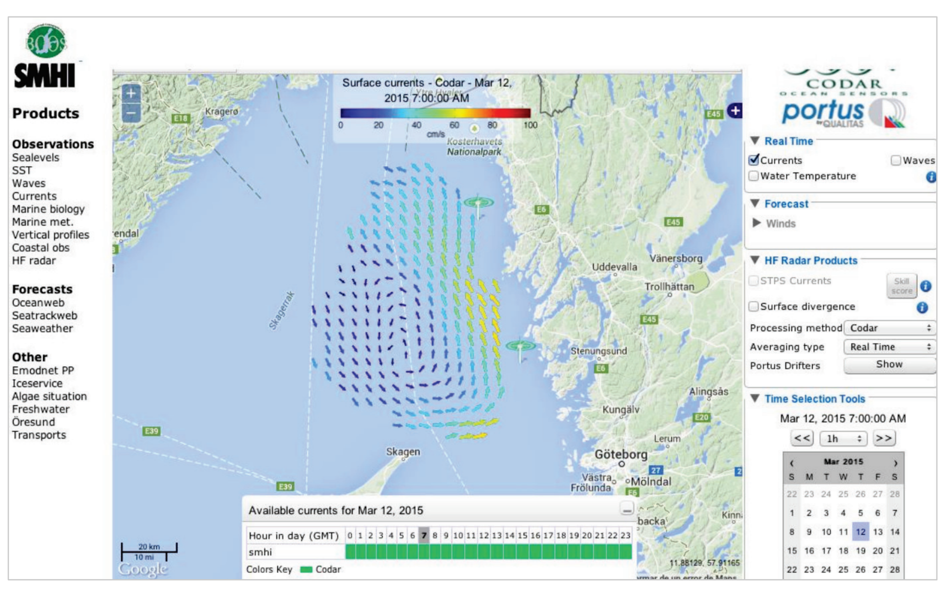

Figure 4: Observations of the ocean current obtained from HF radar.

Non-linear data assimilation on the GPU

The drifter observations y^n are assimilated according to the *implicit equal-weight particle filter* (IEWPF) [2,3], in which we have designed all terms to consist of local operation suitable for GPU implementation. Each ensemble member j is at observation time updated according to

$$\psi_j^n = M(\psi_j^{n-1}) + K_j + \alpha_j^{1/2} P^{1/2}\, \xi_j,$$

in which K_j is a Kalman gain type term, pulling the ensemble member towards the observation, and $\alpha_j^{1/2} P^{1/2}\, \xi_j$ is a random perturbation that ensures that all ensemble members are equally probable.

Figure 5 illustrates how each of the terms in IEWPF acts on an arbitrarily chosen particle, based on observations from three drifters.

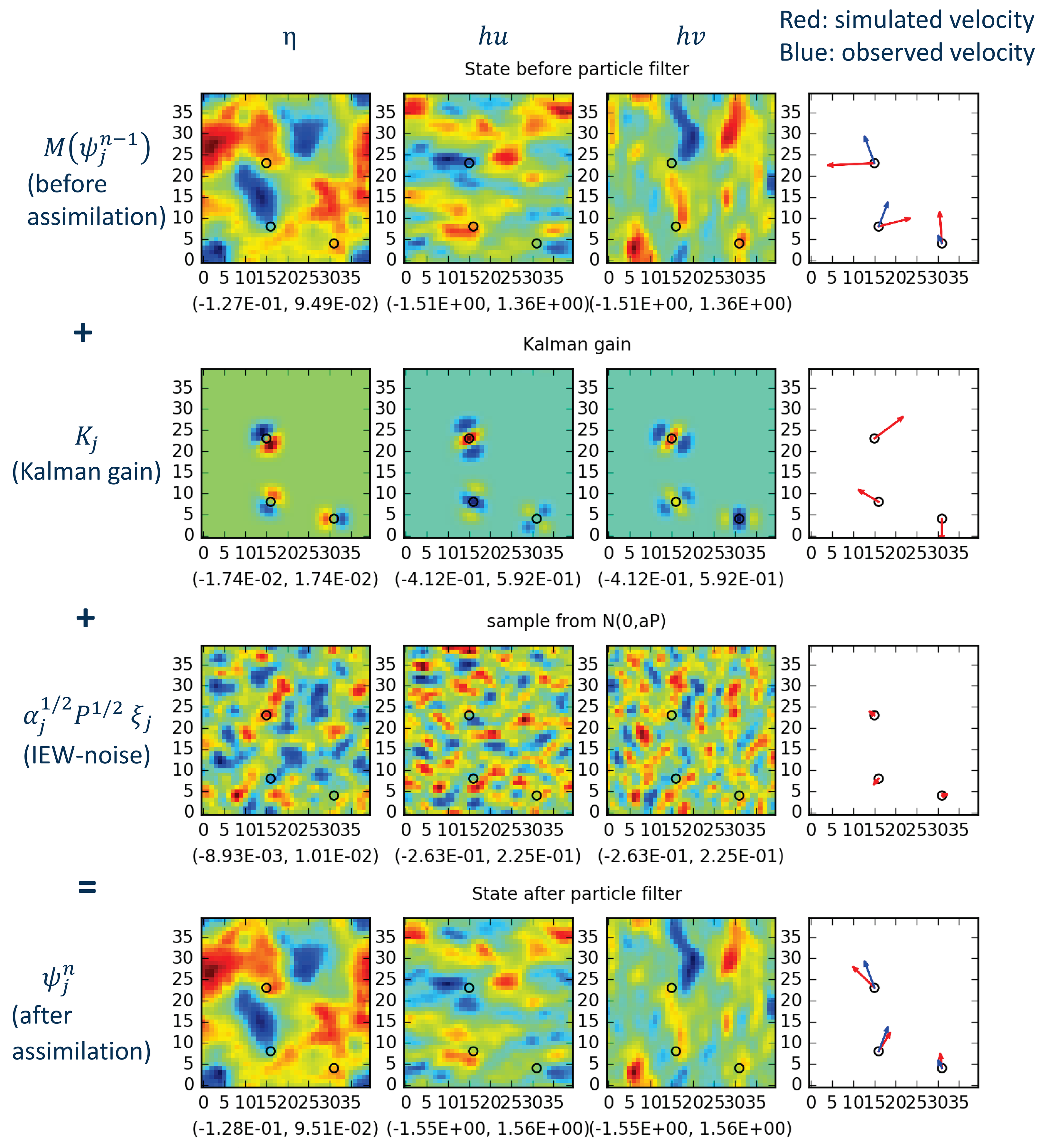

Figure 5: Illustration of how the implicit equal-weight particle filter acts on an ensemble member

Further reading

[1] H. Holm, A. Brodtkorb, G. Broström, K. Christensen and M. Sætra (submitted 2018) 'Test Cases for Rotational Shallow-Water Schemes' (preprint available)

[2] H. Holm. P. van Leeuwen and M. Sætra (in preparation) 'Efficient particle filter for assimilating ocean velocity designed for GPU's'

[3] S. Vetra-Carvalho, P. van Leeuwen, L. Nerger, A. Barth, M. Altag, P. Brasseur, P. Kirchgessner and J.-M. Beckers (2018), 'State-of-the-art stochastic data assimilation for high-dimensional non-Gaussian problems' *Tellus A: Dynamic Meteorology and Oceanography* 70

Source code: https://github.com/metno/gpu-ocean/ Project website: https://www.met.no/en/projects/gpu-ocean

This work is supported by the Research Council of Norway under the GPU Ocean project, grant number 250935/O70.