Proposing A Particle Filter Method for Glider Data Assimilation

Elaine Spiller

Marquette University

International Conference on Data Assimilation

July 14, 2011

Gliders and semi-lagrangian data

Eriksen and Perry
Oceanography
Vol 22, No 2, 2009

Gliders and semi-lagrangian data

Eriksen and Perry
Oceanography
Vol 22, No 2, 2009

How data is collected:

- “fly” through ocean on programmed path

Eriksen and Perry
Oceanography
Vol 22, No 2, 2009

How data is collected:

- "fly" through ocean on programmed path
- surface to report data collected in transit

Gliders and semi-lagrangian data

Eriksen and Perry
Oceanography
Vol 22, No 2, 2009

How data is collected:

- “fly” through ocean on programmed path
- surface to report data collected in transit
- programmed for next leg of flight, repeat

Gliders and semi-lagrangian data

Eriksen and Perry
Oceanography
Vol 22, No 2, 2009

How data is collected:

- “fly” through ocean on programmed path
- surface to report data collected in transit
- programmed for next leg of flight, repeat

What data looks like:

- oceanic variables (biological, chemical optical), pressure, salinity, temperature
- 3-D time-series

Eriksen and Perry
Oceanography
Vol 22, No 2, 2009

How data is collected:

- "fly" through ocean on programmed path
- surface to report data collected in transit
- programmed for next leg of flight, repeat

What data looks like:

- oceanic variables (biological, chemical optical), pressure, salinity, temperature
- 3-D time-series

Glider paths depend on both ocean currents & "driven" motion
Control problem depends on estimates of ocean velocities

Outline

Goal:

- Seeking sequential method that can handle non-linear data

Background:

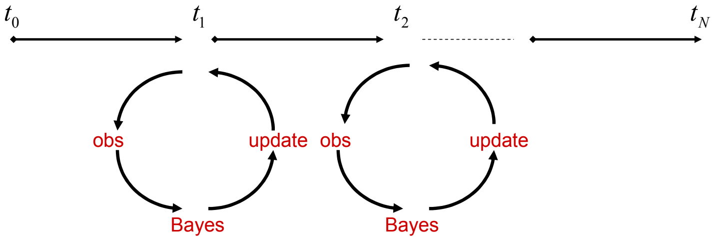
- Particle filters — how to apply

Proposed test problem:

- Test problem — point-vortex system w/controlled tracers

- Strategy for semi-Lagrangian data assimilation
  - Brownian bridge

Bayesian view of sequential DA



$x =$ state

$Y =$ obs $\quad p(x \mid Y) \propto p(Y \mid x)\, p(x)$

Key question: how do we obtain the distributions on RHS?

Particle filters: from t_{j-1} to t_j

prediction step:

$$\pi(x_j | Y_{0,j-1}) = \{x_j, w_j^p(x_j) : w_j^p(x_j) = w_{j-1}(x_{j-1}) \text{ where } x_{j-1} \xrightarrow{\text{SDE}} x_j\}$$

Particle filters: from t_{j-1} to t_j

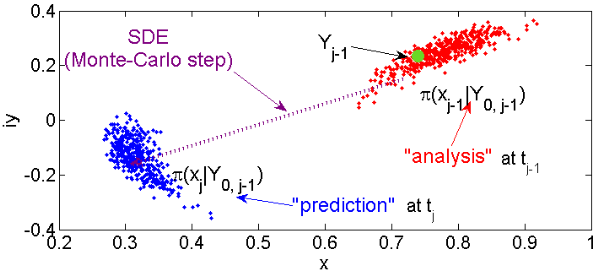
prediction step:

$$\pi(x_j|Y_{0,j-1}) = \{x_j, w_j^p(x_j) : w_j^p(x_j) = w_{j-1}(x_{j-1}) \text{ where } x_{j-1} \underrightarrow{\text{SDE}} x_j\}$$

discrete approx:

Particles are the support of the discrete approximations to these distributions

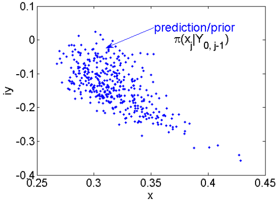
Each particle is associated with a state weight, $w_j(x_j)$



Particle filters: update/analysis at $t = t_j$

Know (discrete approximation):

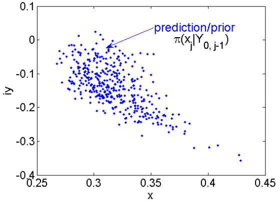
$\pi(x_j|Y_{0,j-1})$ (from last page)



Particle filters: update/analysis at $t = t_j$

Know (discrete approximation):

$$\pi(x_j|Y_{0,j-1}) \text{ (from last page)}$$



Bayes:

$$\pi(x_j|Y_{0,j}) \propto R(x_j, Y_j)\pi(x_j|Y_{0,j-1})$$

Particle filters: update/analysis at $t = t_j$

Know (discrete approximation):

$$\pi(x_j | Y_{0,j-1}) \text{ (from last page)}$$

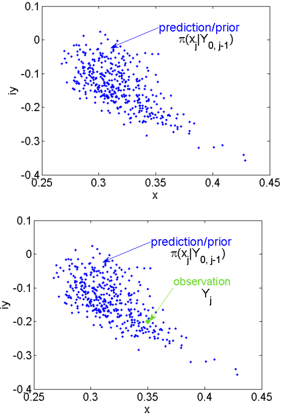
Bayes:

$$\pi(x_j | Y_{0,j}) \propto R(x_j, Y_j)\pi(x_j | Y_{0,j-1})$$

Likelihood:

$$R(x, Y) = \exp[\frac{H(x) \cdot Y}{\theta^2} - \frac{|H(x)|^2}{2\theta^2}]$$

($H(x)$ – observation operator)



Particle filters: update/analysis at $t = t_j$

Know (discrete approximation):

$$\pi(x_j|Y_{0,j-1}) \text{ (from last page)}$$

Bayes:

$$\pi(x_j|Y_{0,j}) \propto R(x_j, Y_j)\pi(x_j|Y_{0,j-1})$$

Likelihood:

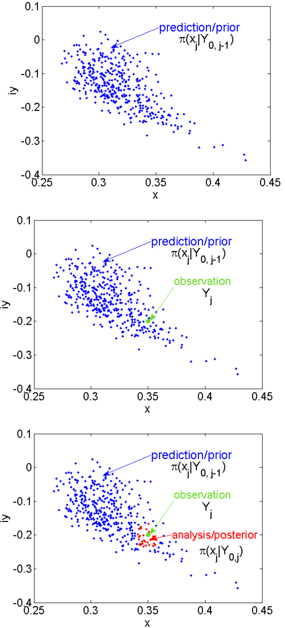
$$R(x, Y) = \exp[\frac{H(x)\cdot Y}{\theta^2} - \frac{|H(x)|^2}{2\theta^2}]$$

($H(x)$ – observation operator)

Update (discrete Bayes):

$$w_j(x_j) \propto R(x_j, Y_j)w_j^p(x_j)$$

$$\pi(x_j|Y_{0,j}) = \{x_j, w_j(x_j)\}$$



Testbed problem: point-vortex flows (2 vortices, 1 tracer)

vortices:

$$\frac{dz_1^*}{dt} = \frac{i}{2\pi} \frac{\Gamma_1}{z_1 - z_2}$$

$$\frac{dz_2^*}{dt} = \frac{i}{2\pi} \frac{\Gamma_2}{z_2 - z_1}$$

Testbed problem: point-vortex flows (2 vortices, 1 tracer)

vortices:

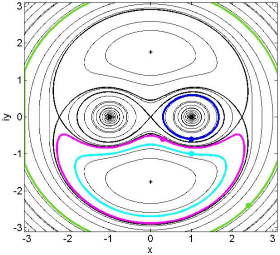
$$\frac{dz_1^*}{dt} = \frac{i}{2\pi}\frac{\Gamma_1}{z_1 - z_2}$$

$$\frac{dz_2^*}{dt} = \frac{i}{2\pi}\frac{\Gamma_2}{z_2 - z_1}$$

tracer:

$$\frac{d\xi^*}{dt} = \frac{i}{2\pi}\frac{\Gamma_1}{\xi - z_1} + \frac{i}{2\pi}\frac{\Gamma_2}{\xi - z_2}$$

Stream function



$\mathbf{x} = \{z_1, z_2, \xi\}$ — state variable

Γ — circulation strength

Testbed problem: point-vortex flows (2 vortices, 1 tracer)

vortices:

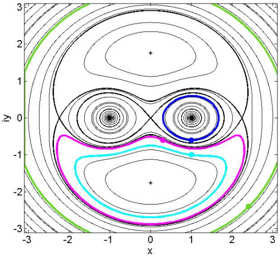
$$\frac{dz_1^*}{dt} = \frac{i}{2\pi}\frac{\Gamma_1}{z_1 - z_2}$$

$$\frac{dz_2^*}{dt} = \frac{i}{2\pi}\frac{\Gamma_2}{z_2 - z_1}$$

tracer:

$$\frac{d\xi^*}{dt} = \frac{i}{2\pi}\frac{\Gamma_1}{\xi - z_1} + \frac{i}{2\pi}\frac{\Gamma_2}{\xi - z_2}$$

Stream function



$\mathbf{x} = \{z_1, z_2, \xi\}$ — state variable

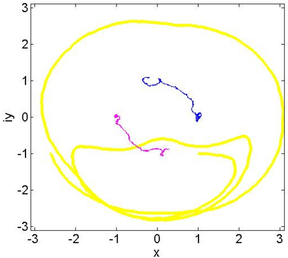
Γ — circulation strength

noisy flow: $dX_t = M(X_t, t)dt + G(X_t, t)dW_t$

- additive model noise $G(X_t, t)dW_t = \sigma dW_t$ with
  – unresolved small scale effects & uncertainty

Lagrangian DA: passive tracer

goal: Estimate vortex paths (their locations determine dynamics)



passive tracer twin experiment:

- generate one “truth run”
- observe tracer locations periodically ($t_j = j\Delta t$),
- $Y_{obs,j} = \xi^{obs}(t_j) = \xi_j^{true} + \theta\eta_j$ with $\eta_j \sim N(0, \mathbf{I})$
- use DA to infer vortex locations

assimilated vortex paths, $\tilde{z}_1$ & $\tilde{z}_2$:

- $\pi(\xi, z_1, z_2)$ state distribution updated w/tracer observations
- $\tilde{z}_1(t) = E_\pi[\tilde{Z}_1] = \sum w_j \tilde{Z}_{1,j}$
- $\tilde{z}_2(t) = E_\pi[\tilde{Z}_2] = \sum w_j \tilde{Z}_{2,j}$

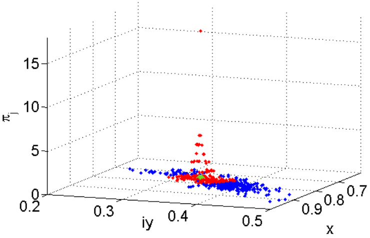
Resampling

problem: degeneracy

- all the weight gets centered on a few particles
- well known and studied

idea:

- pick subset of "best" particles $k = 1, \ldots, M$
- make m_k copies of each particle where $m_k \propto w_j(x_j^{(k)})$ where $\sum m_k = N$
- prior weight of resampled cloud is $1/N$



reasonable:

- doesn't add sampling error
- stochastic evolution to t_{j+1} "spreads out" cloud

Semi-lagrangian DA:

Idea:

- We know where glider surfaces via GPS
- Actual glider path and proposed glider path will differ — proposed path based on forecast of velocity field
- Surfaced location likely different from targeted surfacing x_{target}, y_{target}
- *How* to use time-series nature of data between surfacing?

Semi-lagrangian DA:

Idea:

- We know where glider surfaces via GPS
- Actual glider path and proposed glider path will differ — proposed path based on forecast of velocity field
- Surfaced location likely different from targeted surfacing x_{target}, y_{target}
- *How* to use time-series nature of data between surfacing?

Scientific goal:

- like to recover actual path taken and provide velocity forecast for next step

Semi-lagrangian DA:

Idea:

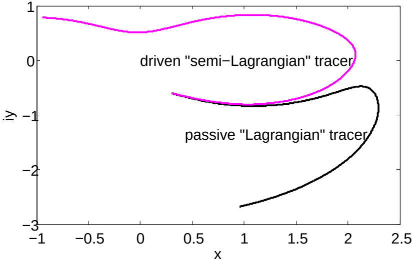
- We know where glider surfaces via GPS
- Actual glider path and proposed glider path will differ — proposed path based on forecast of velocity field
- Surfaced location likely different from targeted surfacing x_{target}, y_{target}
- *How* to use time-series nature of data between surfacing?

Scientific goal:

- like to recover actual path taken and provide velocity forecast for next step

Current objective:

- identify test problem and DA strategy to deal with semi-lagrangian DA



Equations for vortices (Z_1 and Z_2) remain the same. Glider equations become

$$d\xi = \left\{ \frac{i}{2\pi}\left(\frac{\Gamma_1}{\xi - Z_1} + \frac{\Gamma_2}{\xi - Z_2} \right) + A(\omega t) \right\} dt + \sigma dW_t$$

With $A(\omega t)$ a “streering” function, determined by “pilot”.

Note, particles have same $A(\omega t)$, but individual (Z_1 and Z_2).

How to assimilate?

Note: Observe "surfacing" location.

Observation: Surfacing location (observation) becomes a target for DA.

In some sense, looking to use a 1-step smoother so that glider particles "end up" near the surfacing location.

Proposed Approach: Similar to spirit to

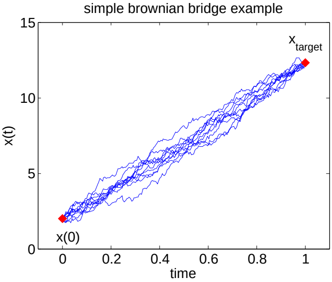
(VanLeeuwen *Efficient* PF, 2010-11; *Implicit* PF Chorin , 2009-10)

Proposed Tool: Brownian bridge

Brownian bridge

How it works: force the random walk to end at x_{target}

$$dX_t = \frac{x_{target} - X_t}{t_{end} - t} dt + \sigma dW_t$$



idea: Use current surfaced location at target, “push” tracer path toward target with Brownian bridge

Semi-lagrangian particle dynamics w/Brownian bridge

$$d\xi^* = \left\{ \frac{Y_{obs} - \xi}{t_j - t} + \frac{i}{2\pi}\left(\frac{\Gamma_1}{\xi - Z_1} + \frac{\Gamma_2}{\xi - Z_2} \right) + A(\omega t) \right\} dt + \sigma dW_t$$

(note, vortices have same dynamics as before)

Tracer semi-lagrangian dynamics:

- black: motion due to vortices (specific to each particle)
- control: steered tracer motion with $A(\omega t)$ (same for each particle)

Particle dynamics for purposes of filtering:

- Brownian bridge: define $b(\xi(t); Y_{obs}(t_j)) = \frac{Y_{obs} - \xi}{t_j - t}$

biasing:

- In filtering, use Brownian bridge and keep track of each particles “bridge magnitude”

biasing:

- In filtering, use Brownian bridge and keep track of each particles “bridge magnitude”

Idea:

- If Brownian bridge had to work hard, hidden vortex states far from truth

biasing:

- In filtering, use Brownian bridge and keep track of each particles “bridge magnitude”

Idea:

- If Brownian bridge had to work hard, hidden vortex states far from truth

Correction:

- Sampling paths w/bridge (from $\tilde{\pi}$, but we want to know how probable those paths would have been w/o bridge)

Filtering semi-Lagrangian dynamics

biasing:

- In filtering, use Brownian bridge and keep track of each particles "bridge magnitude"

Idea:

- If Brownian bridge had to work hard, hidden vortex states far from truth

Correction:

- Sampling paths w/bridge (from $\tilde{\pi}$, but we want to know how probable those paths would have been w/o bridge)

$$\frac{\pi}{\tilde{\pi}} = \frac{\exp(-\int_{t_{j-1}}^{t_j} |\dot{\xi} - g|_\Sigma^2 d\tau)}{\exp(-\int_{t_{j-1}}^{t_j} |(\dot{\xi} - g) - b|_\Sigma^2 d\tau)}$$

Filtering semi-Lagrangian dynamics

biasing:

- In filtering, use Brownian bridge and keep track of each particles "bridge magnitude"

Idea:

- If Brownian bridge had to work hard, hidden vortex states far from truth

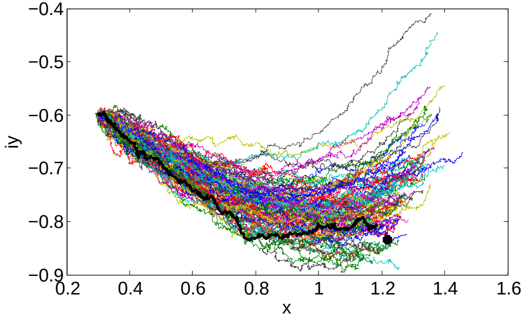
Correction:

- Sampling paths w/bridge (from $\tilde{\pi}$, but we want to know how probable those paths would have been w/o bridge)

$$\frac{\pi}{\tilde{\pi}} = \frac{\exp(-\int_{t_{j-1}}^{t_j} |\dot{\xi} - g|_{\Sigma}^2 d\tau)}{\exp(-\int_{t_{j-1}}^{t_j} |(\dot{\xi} - g) - b|_{\Sigma}^2 d\tau)}$$

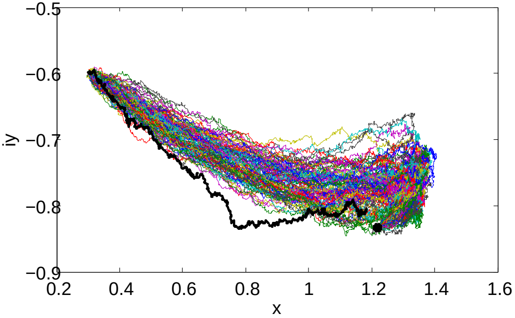
$$R(\xi, Y_{obs}) = \frac{\pi}{\tilde{\pi}} = \exp[-\int_{t_{j-1}}^{t_j} |b(\xi, \tau)|_{\Sigma}^2 d\tau]$$

Particle paths: no Brownian bridge (no biasing)



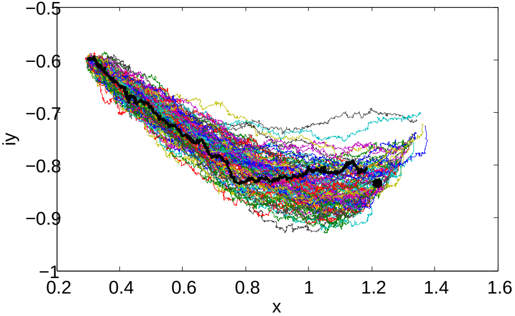
Many particle paths end far from “surfacing” observation

Particle paths: Brownian bridge



Problem: biasing with bridge causes particle paths to overshoot

Particle paths: Modified Brownian bridge



Idea: bias with component of Brownian bridge orthogonal to direction of dynamics (for each particle)

Future directions

Left to do:

- Testing/ computer experiments – what cases “break“ the the filter

Other possibilities:

- Better choices of nudging/biasing term *b*
- A different test model that with a (near) continuous time-series variable/parameter that can be used as data in filtering
- Perhaps use a cluster of gliders for better indentification

Future directions

Left to do:

- Testing/ computer experiments – what cases “break“ the the filter

Other possibilities:

- Better choices of nudging/biasing term *b*
- A different test model that with a (near) continuous time-series variable/parameter that can be used as data in filtering
- Perhaps use a cluster of gliders for better indentification

Thanks!

Supported by an ONR MURI *Ocean 3D+1*