An Implicit Particle Filter for Data Assimilation

Robert N. Miller
Yvette Spitz
Brad Weir
Ethan Atkins
Alexandre Chorin
Matti Morzfeld

Outline

- Introduction: Particle filters, why and why not?
- What's 'implicit' about the implicit filter?
- How it works
- Parameter estimation
- Examples
- Conclusions

Introduction

- Least squares methods (KF, 4DVAR) may not perform well in applications characterized by strong nonlinearity and pdfs that differ greatly from normal
- Particle filters can approximate the evolving pdf of the system, with only the weakest assumptions about the structure of the pdf
- Many proposed particle filters are inefficient in application to problems in high space dimension
- We propose an efficient particle filter (cf, Chorin and Tu (2009,2010); Chorin, Morzfeld and Tu, (2010)), and show first results.

Particle Filters

- A typical particle filter
  - Draw samples (particles) from the estimated initial pdf (the *prior*)
  - Apply the stochastic-dynamic model to each particle.
  - Adjust the sample in some way, in order to account for the information provided by observations
- There are many proposed algorithms for adjusting the prior pdf

The Posterior PDF

- How to adjust the sample to reflect the *posterior* pdf, i.e., the pdf conditioned on observations?
- A typical strategy: Successive Importance Resampling (SIR)
  - Given:
    - An observation $\mathbf{z}$ with error variance R
    - A mapping H from state space to observation space
    - A collection of particles with states $\mathbf{x}_i$

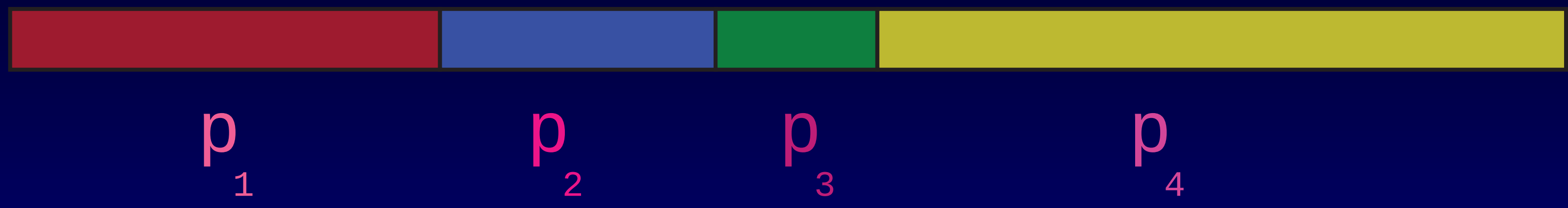
  Calculate

$$p_i \;=\; exp\left(-(\mathbf{z} - H\mathbf{x}_i)^T R^{-1}(\mathbf{z} - H\mathbf{x}_i)\right)$$

  - Use the p_i to determine the ensemble for the next assimilation cycle

The SIR Filter

- Line up the p_i like this:



- Choose a uniform random number between zero and $\sum_1^N p_i$
- Use that random number to choose a particle to participate in the next assimilation cycle
- Repeat this process, with replacement, to choose the sample for the next assimilation cycle
- Particles most likely to be chosen are the ones with the highest probability of giving rise to the observation

The Trouble with SIR

- In high dimension, no particle is very likely to predict an observed value close to the actual observation
- In practice, the sample collapses to a very small number of particles
- This is the problem of “Sample Impoverishment"
- Other strategies (e.g., Metropolis-Hastings) run into the same problem
- An enormous number of particles may be required to maintain a reasonable sample

The Implicit Particle Filter

Proposed solution:
Choose particle paths that are informed by the observation.
Assume a discrete model:

$$\begin{aligned} \mathbf{x}_{j+1} &= \mathbf{x}_j + f(\mathbf{x}_j)\Delta t + (\Delta t)^{1/2} G b \\ &\equiv \mathbf{x}^f_{j+1} + (\Delta t)^{1/2} G b \end{aligned}$$

$b \sim N(0, I)$

The Implicit Particle Filter

So

$$\begin{aligned} \mathbf{x}_{j+1} - \mathbf{x}_j - \Delta t f(\mathbf{x}_j) &\equiv \mathbf{x}_{j+1} - \mathbf{x}_{j+1}^f \\ &\sim N(0, \Delta t G G^T) \end{aligned}$$

For this particle, the variance of the state conditioned on an observation $\mathbf{z}$ at time t_{j+1} is

$$\begin{aligned} \mathbf{F} &= (\mathbf{x} - \mathbf{x}_{j+1}^f)^T (\Delta t G G^T)^{-1} (\mathbf{x} - \mathbf{x}_{j+1}^f)/2 \\ &\quad + (\mathbf{z} - H\mathbf{x})^T R^{-1} (\mathbf{z} - H\mathbf{x})/2 \end{aligned}$$

(As usual, assume observation noise is independent of model noise)

The Implicit Particle Filter

after a bit of algebra $\cdots$

$$\begin{aligned}
\mathbf{F} &= \phi + (\mathbf{x} - m)^T (P^a)^{-1} (\mathbf{x} - m)/2 \\
\phi &= min(\mathbf{F}) \\
&= (\mathbf{z} - H\mathbf{x}^f)^T (HQH^T + R)^{-1} (\mathbf{z} - H\mathbf{x}^f)/2 \\
m &= \mathbf{x}^f_{j+1} + K(\mathbf{z} - H\mathbf{x}^f_{j+1}) \\
Q &= \Delta t G G^T \\
K &= QH^T (HQH^T + R)^{-1} \\
P^a &= (I - KH)Q
\end{aligned}$$

The Implicit Particle Filter

Recipe for the implicit particle filter
Construct the updated value for each particle:

- Generate $\xi_i \sim N(0, I)$, of the dimension of the state space
- Choose the updated state $\mathbf{x}_i$ of the i^{th} particle so that

$$(\mathbf{x}_i - m)^T (P^a)^{-1} (\mathbf{x}_i - m) \;=\; \xi_i \cdot \xi_i$$

The Implicit Particle Filter

- One way to do this: perform a Cholesky decomposition: $P^a = LL^*$ and set

$$\mathbf{x}_i = m + L^{-1}\xi_i$$

- The weight for the i^{th} particle is then $e^{-\phi}J$
  - J is the Jacobian of the map from $\mathbf{x}_i - m$ to $\xi-$space
- The analysis is the weighted sum of the particle states.
- A resampling scheme such as SIR can then be applied

Extensions: Higher Order Numerical Methods

The Klauder-Peterson method for solution of the SDE:

$$d\mathbf{x} = f(\mathbf{x})dt + GdW$$

is given by:

$$\begin{aligned} \mathbf{x}^*_{j+1} &= \mathbf{x}_j + f(\mathbf{x}_j)\Delta t + \Delta t^{1/2} G b^*_{j+1} \\ \mathbf{x}_{j+1} &= \mathbf{x}_j + (\Delta t/2)\left(f(\mathbf{x}_j) + f(\mathbf{x}^*_{j+1})\right) + \Delta t^{1/2} G b_{j+1} \end{aligned}$$

$b^*_{j+1},\ b_{j+1} \sim N(0, I)$ random numbers

Higher Order Methods

Write:

$$
\begin{aligned}
\mathbf{F} &= (\mathbf{x}^*_{j+1} - \Delta t f_j - \mathbf{x}_j)^T Q^{-1} (\mathbf{x}^*_{j+1} - \Delta t f_j - \mathbf{x}_j) + \\
&\quad (\mathbf{x}_{j+1} - \Delta t \bar{f}_j - \mathbf{x}_j)^T Q^{-1} (\mathbf{x}_{j+1} - \Delta t \bar{f}_j - \mathbf{x}_j) + \\
&\quad (\mathbf{z} - H\mathbf{x}_{j+1})^T R^{-1} (\mathbf{z} - H\mathbf{x}_{j+1}) \\
f_j &= f(\mathbf{x}_j) \\
\bar{f}_j &= (f_j + f(x^*_{j+1}))/2
\end{aligned}
$$

- For each particle, find $(\hat{\mathbf{x}}^*_{j+1}, \hat{\mathbf{x}}_{j+1})^T \equiv m$ such that $\mathbf{F}(m) = min(\mathbf{F}) \equiv \phi_0$

Higher Order Methods

- Choose a Gaussian random vector ξ of dimension $2\times$ state dimension and choose $(\mathbf{x}^*_{j+1}, \mathbf{x}_{j+1})$ so that

$$
\begin{aligned}
((\mathbf{x}^*_{j+1}, \mathbf{x}_{j+1})^T &- m)^T (D^2\mathbf{F})^{-1}((\mathbf{x}^*_{j+1}, \mathbf{x}_{j+1})^T - m) \\
&= \xi \cdot \xi
\end{aligned}
$$

- The 2^d approximation may lead to an error in the weight, so:
- Adjust ϕ_0 to account for the difference between $\mathbf{F}$ and its second degree approximation
- Resample

Sparse Observations in Time

Write:

$$
\begin{aligned}
\mathbf{F} \;=\; & (\mathbf{x}^*_{j+1} - \Delta t f_j - \mathbf{x}_j)^T Q^{-1} (\mathbf{x}^*_{j+1} - \Delta t f_j - \mathbf{x}_j) + \\
& (\mathbf{x}_{j+1} - \Delta t \bar{f}_j - \mathbf{x}_j)^T Q^{-1} (\mathbf{x}_{j+1} - \Delta t \bar{f}_j - \mathbf{x}_j) + \\
& (\mathbf{x}^*_{j+2} - \Delta t f_{j+1} - \mathbf{x}_{j+1})^T Q^{-1} \times \\
& \;(\mathbf{x}^*_{j+2} - \Delta t f_{j+1} - \mathbf{x}_{j+1}) + \\
& (\mathbf{x}_{j+2} - \Delta t \bar{f}_{j+1} - \mathbf{x}_{j+1})^T Q^{-1} \times \\
& \;(\mathbf{x}_{j+2} - \Delta t \bar{f}_{j+1} - \mathbf{x}_{j+1}) + \\
& (\mathbf{z} - H\mathbf{x}_{j+2})^T R^{-1} (\mathbf{z} - H\mathbf{x}_{j+2})
\end{aligned}
$$

Sparse Observations in Time

- For each particle, find m, in this case $4\times$ state dimension, that minimizes $\mathbf{F}$
- Choose normally distributed ξ, and $\mathbf{x}^*_{j+1},\ \mathbf{x}_{j+1},\ \mathbf{x}^*_{j+2}$ and $\mathbf{x}_{j+2}$ according to the quadratic approximation of $\mathbf{F}$
- Adjust ϕ and resample
- Generalizations are straightforward:
  - Can add arbitrary numbers of steps in time
  - Can make a fixed-interval smoother by adding intermediate observations

Parameter Estimation

- Can do by linear inverse methods (e.g., Lawson et al., 1995)
- Can introduce new state variables (Dowd, 2006)
- Today: Use successive filter runs to:
  1. Minimize the autocorrelation of the innovation
  2. Minimize a quadratic cost function

An Iterative Minimization Method

- Define the iteration

$$a_{k+1} = a_k - \epsilon_k s_k$$

$$\sum_{k=0}^{\infty} \epsilon_k = \infty$$

$$\sum_{k=0}^{\infty} \epsilon_k^2 < \infty$$

- The obvious choice is $\epsilon_k \propto k^{-p},\ 0.5 < p \leq 1$
- This is the *Robbins-Monro iteration*
- The R-M iteration can be used with any reasonably behaved statistic s that vanishes at optimality, e.g., autocorrelation

The Lotka-Volterra Equations

- A simple ecology model, with a long and colorful history; see numerous web references
- We follow the notation of Lawson et al. (*Bull. Math. Biol. 1995)*

$$\begin{aligned} \frac{dx}{dt} &= x(a_1 + a_2 x + a_3 y) \text{ ``}Prey\text{''} \\ \frac{dy}{dt} &= y(a_4 + a_5 y + a_6 x) \text{ ``}Predator\text{''} \end{aligned}$$

- *For the prey,* $a_1 > 0, \; a_3 < 0$
- *For the predator,* $a_4 < 0, \; a_6 > 0$

Minimizing the Innovation Autocorrelation

Estimate the interaction term a_3

1. For step k:
   - Run the implicit filter through the entire interval of interest
   - Resample by SIR at every assimilation step
2. Calculate lag-1 innovation correlation coefficient C_k
3. If $k = 1$, increment $a_{3,1}$ by 0.5 to form $a_{3,2}$; else
4. Form $a_{3,k+1}$ according to:

$$a_{3,k+1} = a_{3,k} - \frac{1}{k^{0.9}} C_k^{1/4} \frac{a_{3,k} - a_{3,k-1}}{C_k^{1/4} - C_{k-1}^{1/4}}$$

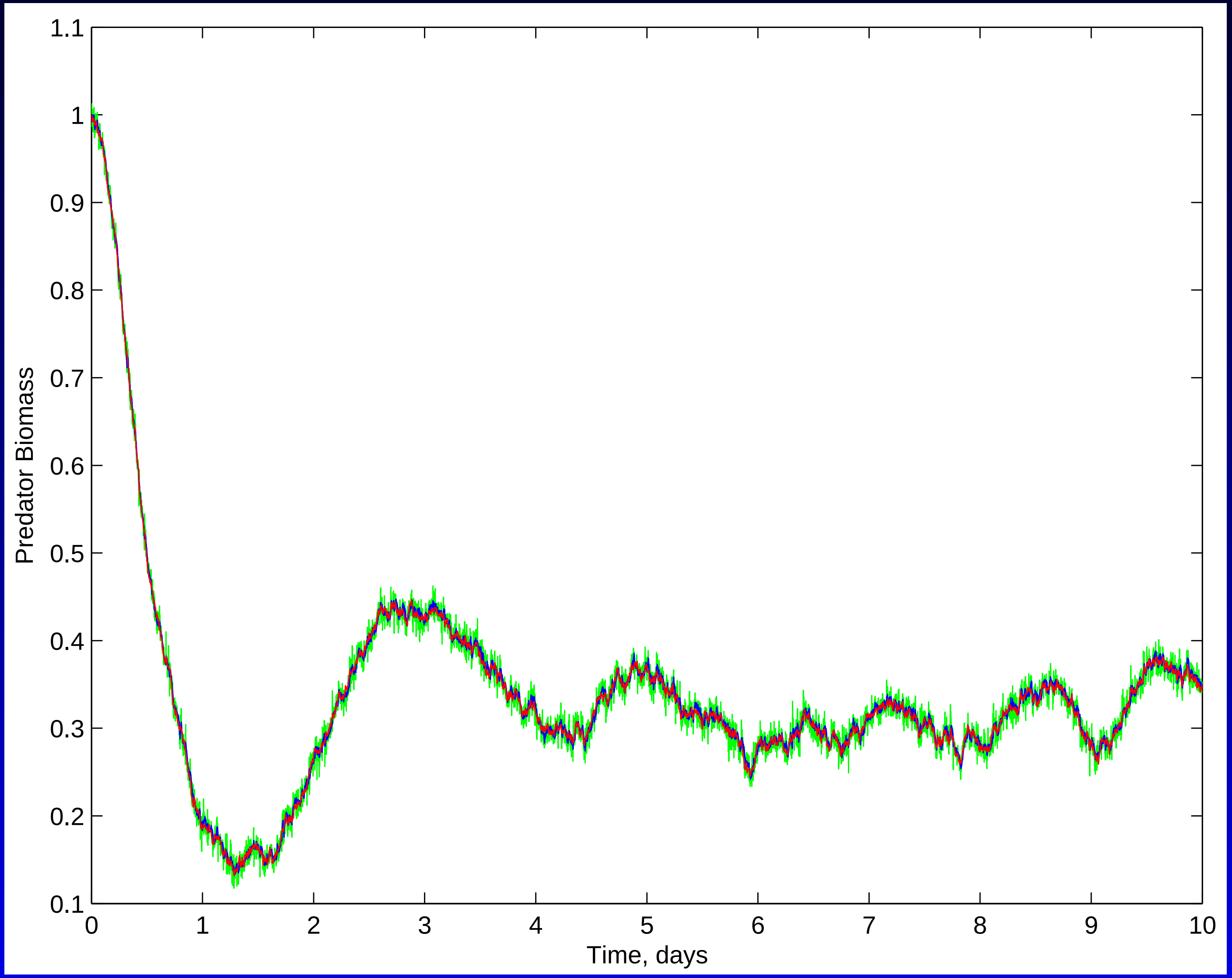
Conditions

1. Each iteration requires a full run of the filter
2. The increment is limited to $2/k^{0.9}$
3. If $C_k^{1/4} - C_{k-1}^{1/4} < 10^{-5}$ the increment is set to zero.

Results for the LV Model

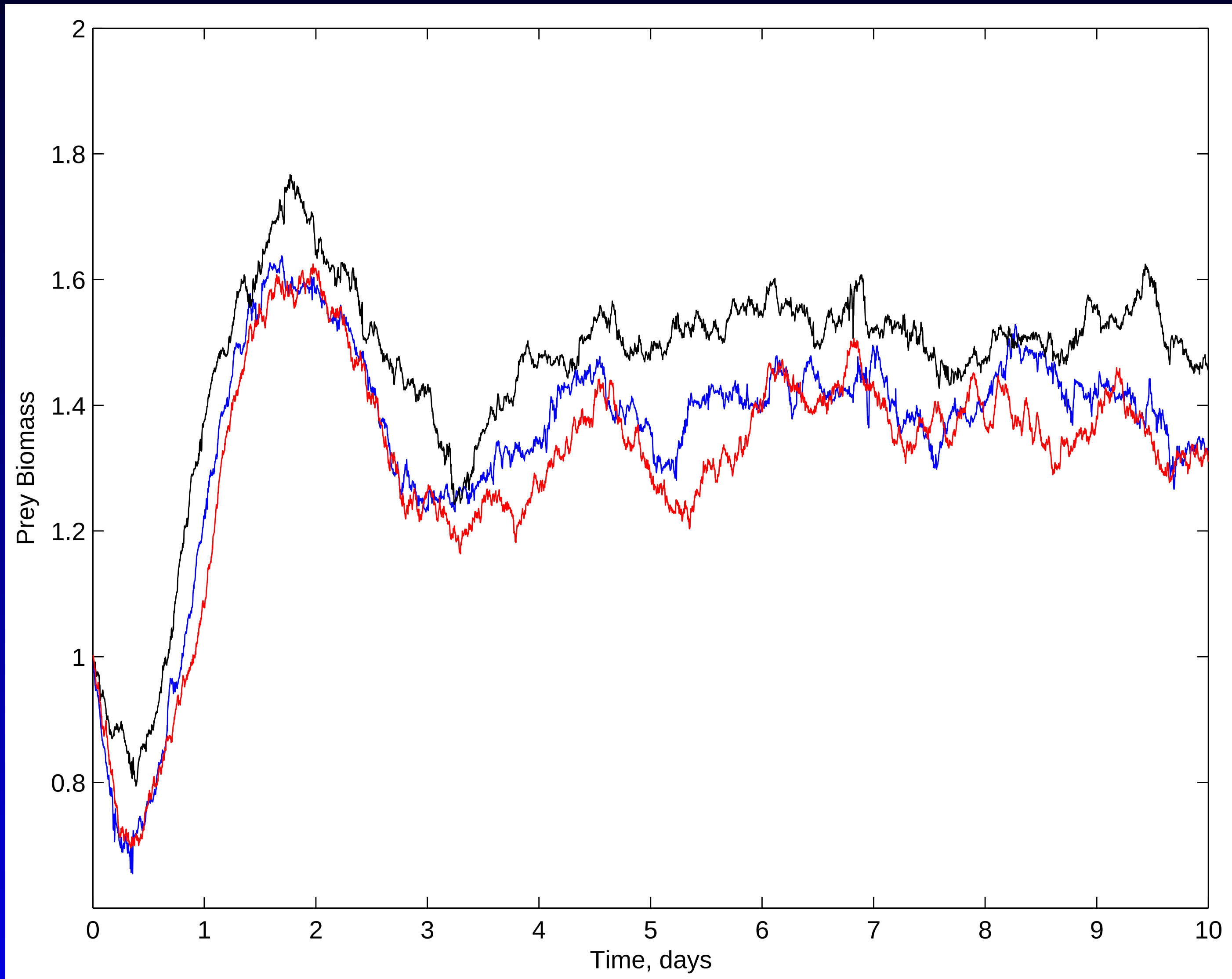
- Generate a reference data set with $a_1 = 4,\ a_2 = -2,\ a_3 = -4$ $a_4 = -6,\ a_5 = 2, a_6 = 4$
- Initial guess: $a_3 = -5$
- Observe the predator y every 2 time steps, $\sigma = 0.01$
- System noise = $\Delta t^{1/2}(0.1, 0.05);\ \Delta t = 0.001$ day.

Predator Biomass



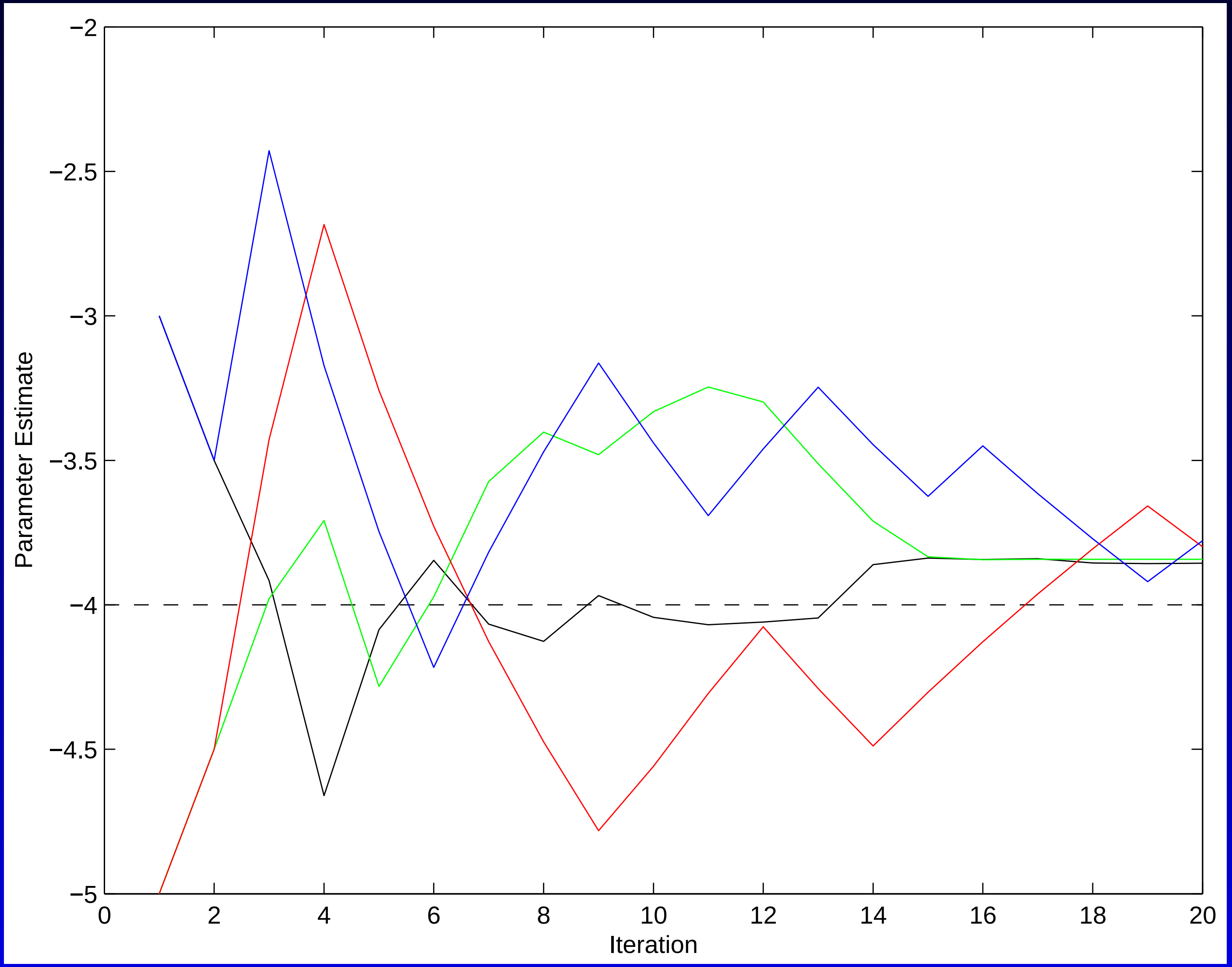
Green: observations; Red: reference solution; Blue: analysis

Prey Biomass



Red: reference solution; Black: analysis, initial parameter; Blue: analysis, final parameter

Parameter Estimation



Estimated parameter vs. iteration for two initial guesses and two noise realizations.

A Generalized LV Model

- No crowding term in the predator equation
- The interaction term takes Michaelis-Menten form, as opposed to mass-action.

$$\frac{dx}{dt} = a_1 x(1 - a_2 x) - \frac{xy}{a_3 + x} \text{ ``}Prey\text{''}$$
$$\frac{dy}{dt} = -a_4 y + a_5 \frac{xy}{a_3 + x} \text{ ``}Predator\text{''}$$

- Fix parameters:
  $a_1 = 1,\ a_2 = 0.5,\ a_3 = 1,\ a_5 = 3$
- There is a subcritical Hopf bifurcation at $a_4 = 1$.
- If $a_4 > 2$, the predators become extinct.

Least-Squares Parameter Estimation

- Write cost function $J(a)$ as the expected value of:

$$\sum_{m=1}^{N} (\mathbf{z}^m - H\mathbf{x}^m(a))^T R^{-1} (\mathbf{z}^m - H\mathbf{x}^m(a))$$

  (e.g. Lawson et al.) Expected value taken over realizations of model noise
- a is a vector of parameters
- The filter generates an approximation $E(\mathbf{x}^m)$ of the expected value of the state vector

Least-Squares Parameter Estimation

- Differentiate the state evolution equation

$$\mathbf{x}_{j+1} = \mathbf{x}_j + f(\mathbf{x}_j)\Delta t + (\Delta t)^{1/2} G b$$

  w.r.t. a,
- Derive an evolution equation for $\nabla_a (\mathbf{z}^m - H\mathbf{x}^m(a))^T R^{-1} (\mathbf{z}^m - H\mathbf{x}^m(a))$
- Use the gradient of the innovations to calculate $\nabla_a J$.
- Each iteration of a gradient descent technique requires a filter run

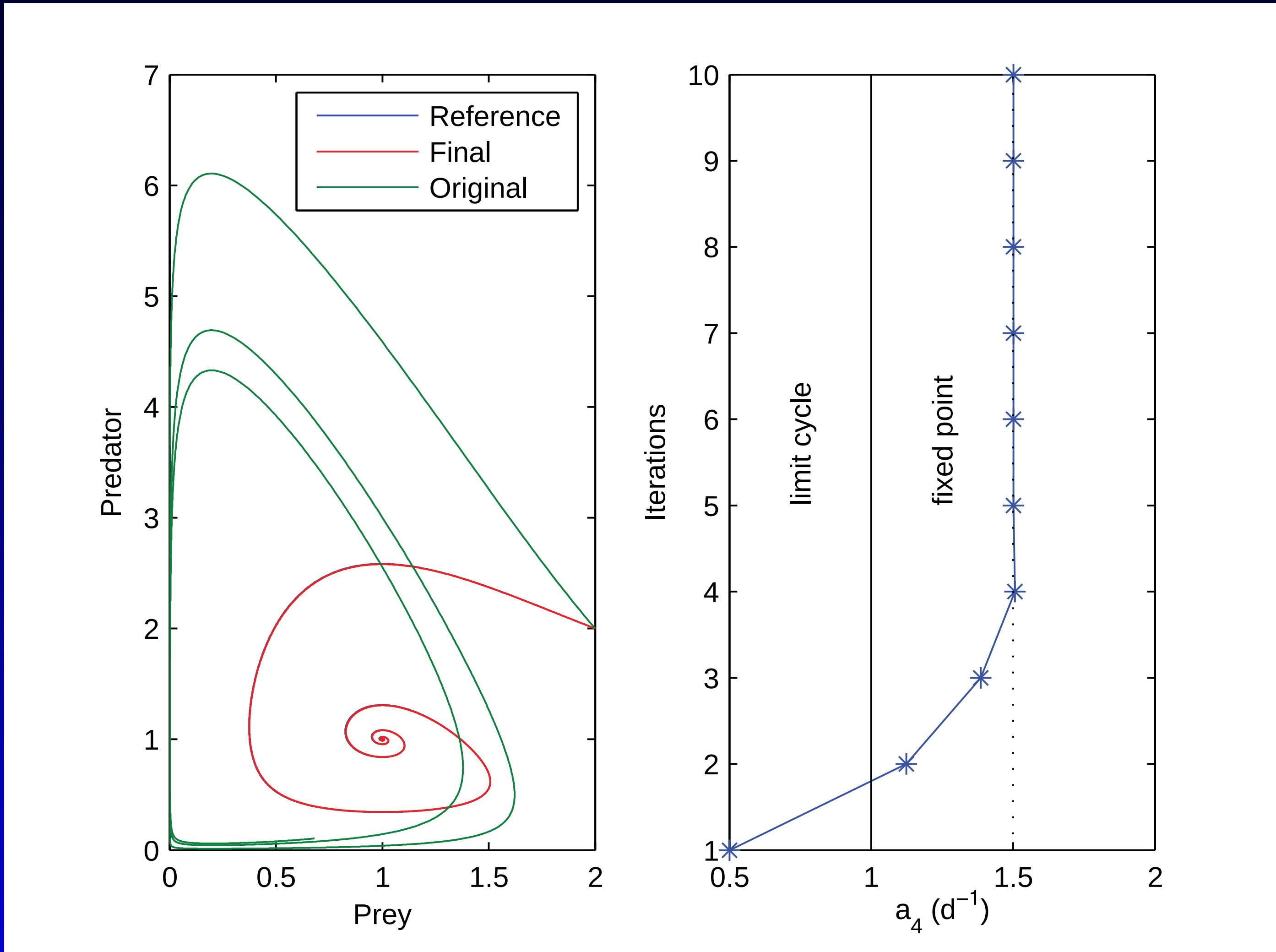
An Iterative Minimization Method

The Robbins-Monro iteration for least-squares

$$
\begin{aligned}
a_{k+1} &= a_k - \epsilon_k \nabla_a J(a) \\
\epsilon_k &\longrightarrow 0
\end{aligned}
$$

(Can loosen the conditions on ϵ_k in the R-M iteration)
Could also use a gradient descent method

Least Squares Parameter Estimation for Generalized LV



The parameter can be estimated accurately, even with an initial guess on the wrong side of the bifurcation (B. Weir)

Remarks on Generalized LV

- A good model and good data lead to good state and the parameter estimates, even if:
  - The initial parameter estimate is on the wrong side of a bifurcation
  - The initial parameter estimate leads to extinction of the predator
- The parameter is still well-estimated for greater values of the model noise, but the results of state estimation are harder to interpret

Conclusions

- The implicit particle filter is a simple method for explicitly estimating the pdf of a stochastic system conditioned on observations
- The implicit particle filter has the possibility of avoiding sample impoverishment,
- The implicit particle filter has proven successful in parameter estimation exercises