Hidden in the Noise: Tracking and Prediction in Complex Systems

Outline

• Particle Filtering Overview
  – Fish Tracking Example
• Selective Resampling Particle Filter
  – Multi-Target Tracking Problem
• Refining Grid Stochastic Filter
  – Performer Tracking & Prediction Problem
• Parameter Estimation in Random Environments
• Infinite Dimensional Exact Filter
  – Pod Tracking Problem
  – Financial Applications
Filtering Overview

Signal:
\[ dX_t = \mu(X_t)dt + \sigma(X_t)dB_t \]

Observation:
\[ Y_t = h(X_t, V_t) \]

Conditional Distribution
\[ P(\ X_t \ | \ \sigma(\ Y_1, \ldots, \ Y_t) ) \]

Optimal Filter
\[ E[X_t \ | \ \sigma(\ Y_1, \ldots, \ Y_t) ] \]

Filtering Procedure

1. Modeling an unobserved signal.
2. Modeling the partial/noisy/distorted observation.
3. Filter outputs a conditional distribution estimate of the signal’s past/present/future state, based on observations.
Non Linear Particle Filters

- Sophisticated Monte-Carlo method that involves modeling signal/observation process
- Particles are copies of signals that evolve independently of one another between observations
- Weight particles based on state and observation based information

What’s Your Best Estimate?

**Weighted Particle Filters**
\[
\sum_{j=1}^{N} W(\xi_i^j) \delta_{\psi}(\cdot) \quad \quad \sum_{j=1}^{N} W(\xi_i^j)
\]

**Optimal Filter**
\[
P(X_t \in \cdot | \sigma(Y_1, \ldots, Y_t))
\]
\[
W(\xi_i^j) = \prod_{j=1}^{t} \frac{1}{\sqrt{\sigma^2}} \exp \left( -\frac{(Y_t^j - h(\xi_i^j))^2}{2\sigma^2} \right)
\]
\[ P(X_t \in A \mid \sigma(Y_t, \ldots, Y_{t+1})) = \frac{\sum_{i=1}^{N} W(\xi_i) \tilde{a}_i^{(A)}}{\sum_{i=1}^{N} W(\xi_i)} \]
Static Grid Filter

Observation: $Y_{t+1}$
Size = Weight

Adaptive Grid Filters

Resampling
Adaptive Interacting Filter

\[ W(\xi_i^i) = \text{weight calculated by } Y_i \]

\[ P(X_r \in \cdot | \sigma(Y_1, ..., Y_r)) = \frac{1}{N} \sum_{i=1}^{N} \delta_{\xi_i}(\cdot) \]

Resampling

Resampling Issues

\[ \frac{1}{N} \sum_{i=1}^{N} \delta_{(\xi_i, ..., \xi_i)}(A_1 \times ... \times A_r) \xrightarrow{N \to \infty} P(X_1 \in A_1, ..., X_r \in A_r | Y_1, ..., Y_r) \]

\[ P(X_1 \in A_1 | Y_1) P(X_2 \in A_2 | Y_2, Y_2) ... P(X_r \in A_r | Y_1, ..., Y_r) \]
Historically Adaptive Grid Filters

\[ \frac{1}{N} \sum_{i=1}^{N} \delta_{(x_{i-1}, ..., x_{i})} \rightarrow_{N}^{\rightarrow m} P(X_i \in A, ..., X_i \in A | Y_i, ..., Y_i) \]

- Mitacs-Pints Branching (MIBR) filter first historically adaptive filter (Fleischmann, Kouritzin 1999)

- Historical Adaptive Interacting Filter (Del Moral, Kouritzin, Miclo 2001) follows similar procedure as Adaptive Interacting Filter

MIBR Filter

Observation \_i

Time
Fish Tracking Problem

Fish Model

\[ dX_t = -\alpha(X_t - \frac{L}{2})dt + \beta d\nu_t + \chi_{\omega D}(X_t)\gamma(X_t)d\xi \]

\[ L = L_1 \times L_2 \]

The MIBR method will track the fish
Selective Resampling Particle Filter
(SERP Filter)

- Most recent particle filter developed at PINTS
- Dramatically outperforms any other particle filter
- Feedback control ň determines extent of resampling
- ň is determined via solution to stochastic control problem
1) Resample Particles

\[ W(\xi'_j) \]

\[ W(\xi'_i) \]

Prob:

\[ \frac{w(\xi'_j)}{w(\xi'_j) + w(\xi'_i)} \]

2

\[ \frac{w(\xi'_i)}{w(\xi'_i) + w(\xi'_j)} \]

2

 Until \( W(\hat{\xi}_i^t) < \bar{w} W(\hat{\xi}_j^t) \) all \( i, j \)

2) Evolve Particles

3) Calculate Weights given new observation

Selective Resampling Particle Filter
(SERP Filter)

- Naturally efficient data structures

- Less resampling noise

- Optimal amount of resampling through feedback mechanism
Multiple Target Tracking

• Goal: Determine the number of targets present in a given area and track them effectively

• Need to be able to model potential interactions between targets

Multiple Ships Lost at Sea

• Sample problem: An unknown number of dinghies in bounded space need to be identified and tracked
Multiple Ships Lost at Sea

• Signal Model:
  • Ship is a nonlinear system with six state components: (x,y) position, orientation, and their respective speeds
  • Have a maximum/minimum speed
    • Speeds affected by type of motion: Rowing, Drifting, Motoring – Seventh state
    • Orientation drifts towards 0 degrees

Weakly Interacting Targets

\[
X_t = \begin{bmatrix}
x \\
y \\
\theta
\end{bmatrix}
\]

\[
X_t' = \begin{bmatrix}
x' \\
y' \\
\theta'
\end{bmatrix}
\]

Attraction-Repulsion Field
**Attraction Repulsion Field**

\[
\kappa(s_1, s_2) = \begin{cases} 
0 & \text{if } s_1 = s_2 \\
\frac{3}{1000N} & \frac{1}{4(\Pi(s_1, s_2) - \varepsilon)}
\end{cases}
\]

\[
\kappa_x(s_1, s_2) = \kappa(s_1, s_2) \frac{\pi_x(s_1) - \pi_x(s_2)}{\Pi(s_1, s_2)}
\]

\[
\kappa_y(s_1, s_2) = \kappa(s_1, s_2) \frac{\pi_y(s_1) - \pi_y(s_2)}{\Pi(s_1, s_2)}
\]

\[
\Pi(s_1, s_2) = \sqrt{\left| \pi_x(s_1) - \pi_x(s_2) \right|^2 + \left| \pi_y(s_1) - \pi_y(s_2) \right|^2}
\]

**Particle Content**

- Particles contain many targets within them
- Can be resampled with particles that contain other targets
Ships Lost at Sea

• Observation Model:
  • We only have noisy discrete-time observations from a high-altitude sensor on a helicopter

\[ Y_t^{l,m} = \int h_{l,m}^{l,m}(X_t) + V_t^{l,m} \]
Refining Grid Stochastic Filter (REST) Filter

- Given a signal that evolves on regular Euclidean subset
- Divide signal state space into a finite number of cells

\[
\begin{array}{cccc}
\text{N}_1 \\
\text{N}_2 \\
\end{array}
\]

In general \( N_1 \times N_2 \times \ldots \times N_d \) cells
Each cell contains:

- Particle count
- Associated Rate

**Refining Grid Stochastic Filter (REST Filter)**

Particles used to approximate unnormalized conditional distribution
Cell Rates

Cell rates are used to calculate net birth (death rate) in a cell.
Rates are determined by cell’s particle count and immediate neighbour’s rates.

Net Birth Rate

Net Birth Rates

Net birth rates are used to mimic signal movement and incorporate observation information.
Dynamic Cell Sizing

Dynamic Cell Sizing Example
Observation: -2 Particles

Splitting Cells...

Tree Node  Cell Node
REST Advantages

• Less simulation noise than particle filters

• Dynamic cell sizing, inherent parameter estimation

• Dynamic domain problems

Performer Problem

- Acoustic tracking system designed to have lighting equipment follow performer on large stage

- Due to mechanical lags, system must be able to predict performer’s future position based on current state
\[ dx_i = f_i \cos(\theta_i) \, dt \]
\[ dy_i = f_i \sin(\theta_i) \, dt \]
\[ dz_i = \alpha \left( \frac{2z_{\max} + z_{\min}}{3} \right) + \sigma \]
\[ df_i = \alpha_y (f_{\text{avg}} - f_i) \, dt \]
\[ d\theta_i = \left\{ -\alpha_1 \theta_i - \alpha_2 \Delta(X_i) \right\} \]
\[ + \sigma_\theta dB_i^\theta \]
Observation Model

\[ h(X_{t_m}, S^l) = \sqrt{(S_x^l - x)^2 + (S_y^l - y)^2 + (S_z^l - z)^2} \]

\[ Y_{t_m} = \begin{cases} 
  h(X_{t_m}, S^l) + \sigma W_{t_m} & \text{if } U_m (0,1) < p \\
  \text{null otherwise} & 
\end{cases} \]
Parameter Estimation for Random Environments

\[ dX_t = \nabla W(X_t) dt + \sigma(X_t) dB_t \]

- Diffusions with Unknown Parameters
  - Combined state and parameter estimation
  - Filter performance
  - Example Problem

- Diffusions in Random Environments
  - Motivation
  - Nonparametric estimation method
  - Particle filters and simulations
Common method for filtering problem

• Filtering Problem

• Common Parameter Estimation Method
  Within Filtering:
  – Conditional likelihood maximization or
    EM algorithm
  – Expensive computation

Example Problem

\[ f(x) = \sum_{i=0}^{n} (i+1) a_i x^i \]
\[ Y_k = h(X_{t_k}) + \sigma X_n(0,1) \]

Estimate parameters via:

• Generalized Method of Moments (GMM)
• Recursive Algorithm
Invariant measure method

• Idea:
  – Find moments of invariant measure by ergodic theorem - GMM
  – Obtain unknown parameters through solving non-singular finite-order linear equations

• Invariant Measure:

Invariant measure method (Cont)

• Ergodic Theorem:

• Expression of Moments of
Simulation result:
parameter estimation

Simulation result:
filtering via branching particle method
Recursive Algorithm

\[
\hat{\beta}_{t+1} = \arg \min \left\{ \frac{1}{t+1} \sum_{i=0}^{t} \| Y_{i+1} - E[h(X_{i+1}^\beta) | \sigma(Y_i, ..., Y_t)] \|^2 \right\}
\]

\[
\hat{\beta}_{t+1} = \hat{\beta}_t + P_t \Phi_t A(Y_{t+1} - \hat{Y}_{t+1})
\]

Example

1 Dimensional Signal

\[
dX_t = -\frac{1}{2} (a_0 + 2a_1 x + 3a_2 x^2 + ... + (d+1)a_d x^d) + \sigma dB_t
\]

\[
+ \sigma^2 \chi_{01} (X_t) d\xi^{(0)} - \sigma^2 \chi_{11} (X_t) d\xi^{(1)}
\]
Particle filters

- Initialization
  - Particles are independently initialized
- Evolution:
  - Particles are evolved according to approximate SDE with current parameter estimate
- Observation Arrival
- Parameter Estimation
  - Calculate moments via solving linear equations or recursive method
- Selection:
  - Particles are branched
Part II: Diffusions in random environments

• Motivation:
  Tracking problem of a dinghy lost on lake
  – In calm lakes dinghy moves as reflecting diffusion process
  – Lake swells provide random drift

• Signal Model:

Noisy observation and filtering problem

• Observation Sequence:

  random vector fields

• Filtering Problem:

  Let . For a continuous function , the filtering problem is to evaluate
Nonparametric estimation method

• Idea:
  – Perform functional estimations for the fixed unknown path of the random medium
  – Use obtained estimators to construct approximate filters
• Ergodic Theorem:

Approximate diffusions

• By trigonometric Fourier series approximation we get smooth estimators for .
• Define approximate diffusions
• Approximate filters
Filter Robustness

• Bhatt & Karandikar (1999) robustness yields

• Budhiraja behavior gives

\[ \text{as } n,T \to \text{ in any way} \]

Particle filters (branching)

• Initialization:
  • Particles are independently initialized

• Evolution:
  • Particles are evolved according to approximate SDE with current parameter estimates

• Observation Arrival

• Functional Estimation:
  • Calculate functional environment estimate

• Selection:
  • Particles are branched
Simulation Results:

Conclusions

• New methods of filtering for reflecting diffusions in random medium or with unknown parameters

• Have no access to true random medium or parameters, but contend with noisy estimates

• Computer workable methods

• Simulations experimentally validate our methods
Further investigations

- Generalize nonparametric estimation method to handle signals in random scenery and fractional BM
- Modify parameter estimation procedure to get estimates at all frequencies
- Examine performance of particle filters on more complicated signal models

Infinite Dimensional Exact Filter (IDEX)

- Kalman, Benes, and Daum exact filters propagate finite dimensional sufficient statistic
- Finite dimensionality limits possible observation models
- Computer efficient convolution allows for more general infinite dimensional exact filters
- Developed using explicit solutions, Feynman-Kac formulas, and Ricatti equations
- Certain nonlinear drift, dispersion coefficients possible
**IDEX Formula**

a) \( \mathcal{E}(x) = P(X_j \mid Y_j)(\Lambda^{-1}_x(x)) \times F(x) \)

b) \( \hat{\mathcal{E}} = FFT(\mathcal{E}) \)

c) \( \hat{\Psi}(\xi) = \left[ \frac{1}{a} \exp(b \xi) \hat{\mathcal{E}}(\xi) \right] \exp\left[ -\frac{\xi^2}{2} \right] \)

d) \( \Psi = IFFT(\hat{\Psi}) \)

e) \( P(X_{+1} \mid Y_j(x)) = \Psi(\nu_x)H_{\nu}(\delta_{t_j}, x) \)

**where** \( H_{\nu}(\delta_{t_j}, x) = \exp\left\{ \eta \times \delta_{t_j} + \int_{\nu}^{x} \mu(z) - m(z)dz \right\} \forall x \in \mathbb{R} \)

**IDEX Advantages**

- No simulation noise

- General observation assumption
Pod Tracking Problem

Stratonovich SDE

\[ dX_s = h(X_s)ds + \sigma(X_s) \circ dW_s \]

\[ X_s = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \]

\[ h : \mathbb{R}^3 \longrightarrow \mathbb{R}^3 \]

\[ \sigma : \mathbb{R}^3 \longrightarrow \mathbb{R}^{3 \times 2} \]

\[ W_s : 2-D \text{ Wiener Process} \]
Vector Fields Constraining Signal to Sphere

\[ f(X_r) - f(X_0) = \sum_{k=0}^{2} \int_{0}^{k} A_k f(X_s) \circ dW_s^k \]

\[ (A_0 f)(x) = \sum_{i=1}^{3} h_i(x) \partial_i f(x) \]

\[ (A_j f)(x) = \sum_{i=1}^{3} \sigma_{ij}(x) \partial_i f(x) \quad j = 1, 2 \]

\[ A_k \tilde{f} \equiv 0 \]

\[ f(X) = x_1^2 + x_2^2 + x_3^2 \]

Pod Observation Model

\[ Y_r = \text{proj}_{\{\tau_1, \tau_2\}}(X_r) + W_r \]

\{ \tau_1, \tau_2 \} = \text{basis of plane normal to } r

r = \text{resting position of pod sensor}
Pod Observation Model

Pod Tracking Developments

- Workable explicit solution
  - Eliminate approximation errors

- More realistic manifolds
  - Cantilever equations
Financial Applications

- Problem: Determine model parameters for option pricing
- Idea: Model Signal = function of volatility – combined tracking and parameter estimation. The signal’s formula is:

\[ Z_t = e^{-\kappa t} \left\{ Z_0 + \int_0^t e^{\kappa s} dW_s \right\} \text{ assuming } Z_0 = f \left( 0, \frac{1}{2\kappa} \right) \]

- with parameter \( \kappa = \) time constant of signal process
- Signal is unobservable volatility in stock price of publicly traded company
- Observations also parameterized with:
  - $\mu$ = measure of mean volatility in log scale
  - $\sigma$ = amplitude of volatility noise
  - $\alpha$ = index of stable law for noise
  - $m$ = drift

Volatility Observation Model

Stock Prices: $P_0, \ldots, P_n$

$Y_k = \log(P_k/P_{k-1})$

$P[g(Z_t)|Y_k, t_k, t]$