# Filtering, drift homotopy and target tracking

## University of Bristol

**Vasileios Maroulas**

**University of Tennessee**
and
**University of Bath**

maroulas@math.utk.edu

# Goal

- Central problem arising in many scientific and engineering applications
- Tracking accurately, efficiently and simultaneously $N$ (large) targets

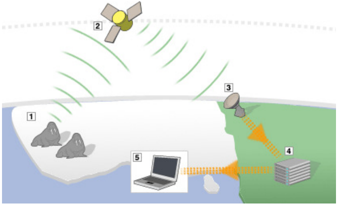
# Example: Tracking Wildlife

- Argos: a satellite–based system collecting data from mobile platforms.

- Ecologists tag and track wildlife through Argos consulting how wildlife behaves.

# Tracking Wildlife

1. Transmitters on animals relay pulses of data
2. Satellite collects data and measures signals' frequencies
3. Satellite relays data to terrestrial receiving sensors
4. Processing center processes data
5. Researchers view information via Internet avenues.



# Goal

- Tracking simultaneously $N$ (large) targets in a fixed domain.

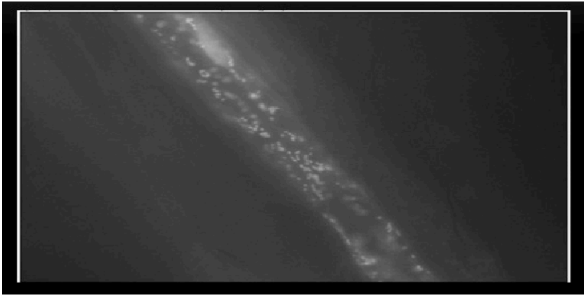


Figure: *Image was captured by Summer REU students mentored by A. Nebenführ and* **VM**

- A plethora of scenarios should be considered.

# Decision on Target Number

# Independent Motion

# Dependent Motion

# Mixed Motion

# Change of Motion and Change of Target Number

# Strategies

- Random Finite Set Filters
  - Consider the targets and associated observations as sets
  - Probability Hypothesis Density (PHD)
  - Cardinalized Probability Hypothesis Density (CPHD)
  - Mahler, Vo, Vo, **VM**...
- Sequential Statistics
  - Sequentially detect and estimate targets
  - Grossi, Lops, **VM**...
- **Particle Filtering**
  - Andrieu, Arulampalam, Bain, Berzuini, Beskos, Crisan, Chopin, Doucet, Gilks, Godsill, Gordon, Fearnhead, Kantas, Latuszynski, Lee, Maskel, Papavasiliou, Papaspiliopoulos, Roberts, Sherlock, Singh, Stinis, Stuart, Whiteley, Weare, West.....

# Single–Object Bayes filtering: Initialization

- $t = 0$: state $x \in \mathbb{R}^N$ distributed according to a priori $f_0(x)$, where $x = (p_x, p_y, p_z, v_x, v_y, v_z, a_x, a_y, a_z)$

# Single–Object Bayes filtering: Initialization

- $t = 0$: state $x \in \mathbb{R}^N$ distributed according to a priori $f_0(x)$, where $x = (p_x, p_y, p_z, v_x, v_y, v_z, a_x, a_y, a_z)$
- If there is good information on the target's position then $f_0$ is a very peaky density

# Single–Object Bayes filtering: Initialization

- $t = 0$: state $x \in \mathbb{R}^N$ distributed according to a priori $f_0(x)$, where $x = (p_x, p_y, p_z, v_x, v_y, v_z, a_x, a_y, a_z)$
- If there is good information on the target's position then $f_0$ is a very peaky density
- If not sufficient knowledge then $f_0$ could be the uniform distribution.

# Single–Object Bayes filtering: Prediction Step

- Object moves between time steps $t$ and $t+1$. Dynamics of the statistical motion of the target captured:

$$X_{t+1} = \phi_t(x', V_t),$$

where $V_t$ is a randomly distributed noise and $\phi_t : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}^N$ is a family of nonlinear, nonsingular functions.

- The *predicted* motion of the object is encapsulated:

$$f_{t+1|t}(x|z_{1:t}) = \int f_{t+1|t}(x|x') f_{t|t}(x'|z_{1:t}) dx', \tag{1}$$

where $f_{t+1|t}(x|x')$ is the Markov transition density and $z_{1:t} \doteq \{z_1, z_2, \cdots, z_t\}$.

# Single–Object Bayes filtering: Update Step

- At recursive time $t+1$ a new observation is collected, $z_{t+1} \in \mathbb{R}^M$.
- (1) needs to be updated using $z_{t+1}$.
- $Z_{t+1} = \eta_{t+1}(x, W_{t+1})$, where $W_{t+1}$ is a randomly distributed noise,$\eta : \mathbb{R}^N \times \mathbb{R}^M \to \mathbb{R}^M$ is a family of nonsingular, nonlinear transformations.
- The *corrected* motion of the object is propagated:

$$f_{t+1|t+1}(x|z_{1:t+1}) \propto f_{t+1}(z_{t+1}|x) f_{t+1|t}(x|z_{1:t}), \qquad (2)$$

where $f_{t+1}(z|x)$ is the likelihood function of the sensor.

# Particle Filter Approach

- Estimate $E[f(X_{T_k})|\{Z_{T_j}\}_{j=1}^k]$ or $p(X_{T_k}|\{Z_{T_j}\}_{j=1}^k)$
- $X_{T_k}$: state vector of our stochastic system.
- $Z_{T_1}, \cdots, Z_{T_K}$: noisy observations of the state of the system at specified instants $T_1, \cdots, T_K$.
- Handle non-linear and/or non-Gaussian cases

# PF Approach

- Computing averages w.r.t. $p(X_{T_k}|\{Z_{T_j}\}_{j=1}^k)$ is difficult
- PF falls in the category of importance sampling.
- Sampling from $q(X_{T_k}|\{Z_{T_j}\}_{j=1}^k)$ which can be easily sampled
- $E[f(X_{T_k})|\{Z_{T_j}\}_{j=1}^k] \approx \frac{1}{N}\sum_{n=1}^N f(X_{T_k}^n)\frac{p(X_{T_k}^n|\{Z_{T_j}\}_{j=1}^k)}{q(X_{T_k}^n|\{Z_{T_j}\}_{j=1}^k)}$

# PF Approach

- Computing averages w.r.t. $p(X_{T_k}|\{Z_{T_j}\}_{j=1}^k)$ is difficult
- PF falls in the category of importance sampling.
- Sampling from $q(X_{T_k}|\{Z_{T_j}\}_{j=1}^k)$ which can be easily sampled
- $E[f(X_{T_k})|\{Z_{T_j}\}_{j=1}^k] \approx \frac{1}{N}\sum_{n=1}^N f(X_{T_k}^n)\frac{p(X_{T_k}^n|\{Z_{T_j}\}_{j=1}^k)}{q(X_{T_k}^n|\{Z_{T_j}\}_{j=1}^k)}$

$$E[f(X_{T_k})|\{Z_{T_j}\}_{j=1}^k] \approx \frac{\sum_{n=1}^N f(X_{T_k}^n)\frac{p(X_{T_k}^n|\{Z_{T_j}\}_{j=1}^k)}{q(X_{T_k}^n|\{Z_{T_j}\}_{j=1}^k)}}{\sum_{n=1}^N \frac{p(X_{T_k}^n|\{Z_{T_j}\}_{j=1}^k)}{q(X_{T_k}^n|\{Z_{T_j}\}_{j=1}^k)}} \qquad (3)$$

- where $N \approx \sum_{n=1}^N \frac{p(X_{T_k}^n|\{Z_{T_j}\}_{j=1}^k)}{q(X_{T_k}^n|\{Z_{T_j}\}_{j=1}^k)}$.

# PF Approach

Filtering is based on the recursion:

$$p(X_{T_k}|\{Z_{T_j}\}_{j=1}^{k}) \propto g(X_{T_k}, Z_{T_k}) p(X_{T_k}|\{Z_{T_j}\}_{j=1}^{k-1}), \qquad (4)$$

where

$$p(X_{T_k}|\{Z_{T_j}\}_{j=1}^{k-1}) = \int p(X_{T_k}|X_{T_{k-1}}) p(X_{T_{k-1}}|\{Z_{T_j}\}_{j=1}^{k-1}) dX_{T_{k-1}}. \qquad (5)$$

Particle filtering is a recursive implementation of the importance sampling approach.

$$q(X_{T_k}|\{Z_{T_j}\}_{j=1}^{k}) = p(X_{T_k}|\{Z_{T_j}\}_{j=1}^{k-1}),$$

then from (4) we get

# PF Approach

$$E[f(X_{T_i})|\{Z_{T_j}\}_{j=1}^{k}] \approx \frac{\sum_{n=1}^{N} f(X_{T_k}^n) g(X_{T_k}^n, Z_{T_k})}{\sum_{n=1}^{N} g(X_{T_k}^n, Z_{T_k})}, \tag{6}$$

$N$ is the number of samples.

- From (6) if we can construct samples from $p(X_{T_k}|\{Z_{T_j}\}_{j=1}^{k-1})$ then we can define the (normalized) weights

$$W_{T_k}^n = \frac{g(X_{T_k}^n, Z_{T_k})}{\sum_{n=1}^{N} g(X_{T_k}^n, Z_{T_k})}.$$

- Weigh the samples and the weighted samples will be distributed according to the posterior distribution $p(X_{T_k}|\{Z_{T_j}\}_{j=1}^{k})$

# A few comments

- Need to associate each target to an observation.

# A few comments

- Need to associate each target to an observation.
- Twofold problem:
  - Combinatorial explosion of the number of possible target-observation arrangements.
  - Targets may come very close or even cross paths requiring the target-observation problem to be solved at every step.
- **Target-observation relies heavily on the accuracy of the underlying filtering algorithm**, i.e. if the filtering algorithm performs poorly at one step then the targets' samples generated at the next step can be off from their true trajectories.

# Classical Algorithm

1. Begin with $N$ unweighted samples $X^n_{T_{k-1}}$ from

$$p(X_{T_{k-1}}|\{Z_{T_j}\}_{j=1}^{k-1}) = \prod_{\lambda=1}^{\Lambda} p(X_{\lambda,T_{k-1}}|\{Z_{\lambda,T_j}\}_{j=1}^{k-1}).$$

2. **Prediction**: Generate $N$ samples $X'^n_{T_k}$ from

$$p(X_{T_k}|X_{T_{k-1}}) = \prod_{\lambda=1}^{\Lambda} p(X_{\lambda,T_k}|X_{\lambda,T_{k-1}}).$$

3. **Target-Observation Association:** Hungarian Algorithm
4. **Update**: Evaluate the weights

$$W^n_{T_k} = \frac{\prod_{\lambda=1}^{\Lambda} g_\lambda(X'^n_{\lambda,T_k}, Z_{\lambda,T_k})}{\sum_{n=1}^{N}\prod_{\lambda=1}^{\Lambda} g_\lambda(X'^n_{\lambda,T_k}, Z_{\lambda,T_k})}.$$

# 1 picture= 1,000 words

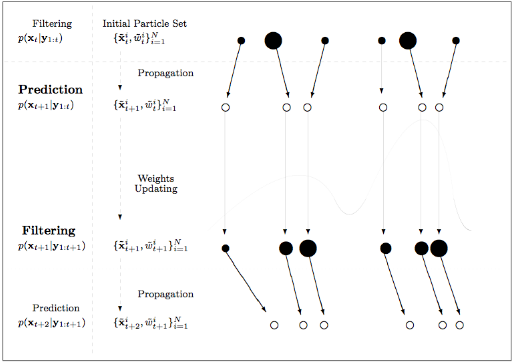



Figure: *Particles evolution in the generic particle filter. Courtesy of Casarin (2004)*

# Classical Algorithm

- Most particles will have a negligible weight with respect to the observation.

5. **Resampling**: Creating more copies of the samples with significant weights based on the current observation.

6. Set $k = k + 1$ and proceed to Step 1.

# 1 picture= 1,000 words

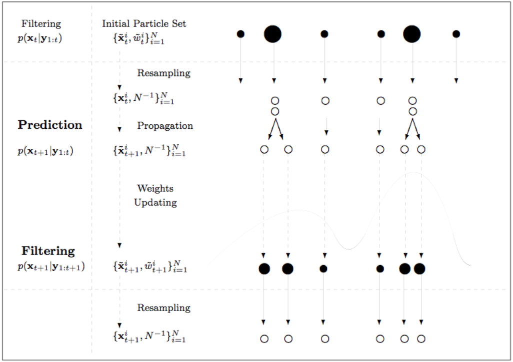



Figure: *Particles evolution in the generic particle filter with resampling. Courtesy of Casarin (2004)*

# Move samples into statistically significant regions

- Particle filters still need a lot of samples to approximate accurately the target distribution.
- One extra step to move samples in statistically significant regions (Gillks-Berzuini 1999, Weare 2009)
- Must preserve the conditional density $p(X_{T_k}|\{Z_{T_j}\}_{j=1}^k)$.

# Move samples into statistically significant regions

- Create more copies not only of the good samples according to the current observation, but also of the values (initial conditions) of the samples at the previous observation.
- These values are the ones which evolved into good samples for the current observation.

# MCMC step appended: a 2-step process:

STEP 1:

**Resampling**: Generate $N$ independent uniform random variables $\{\theta^n\}_{n=1}^N$ in $(0,1)$. For $n = 1, \ldots, N$ let $(X^n_{T_{k-1}}, X^n_{T_k}) = (X'^j_{T_{k-1}}, X'^j_{T_k})$ where

$$\sum_{l=1}^{j-1} W^l_{T_k} \leq \theta^j < \sum_{l=1}^{j} W^l_{T_k},\ j = 1, \cdots, N$$

# MCMC step appended: a 2-step process:

STEP 2:

- Through Bayes rule one can show that the posterior density $p(X_{T_k}|\{Z_{T_j}\}_{j=1}^{k})$ is preserved if sampling from

$$g(X_{T_k}, Z_{T_k})p(X_{T_k}|X_{T_{k-1}}),$$

where $X_{T_{k-1}}$ are given by the modified resampling step.

- This is a problem of conditional sampling.
- *Important issue* is to perform the necessary sampling efficiently

# Drift homotopy

- Consider the signal process: $dX_t = a(X_t)dt + \sigma(X_t)dB_t$

- Consider an SDE system with modified drift

$$dY_t = b(Y_t)dt + \sigma(Y_t)dB_t,$$

  $b(Y_t)$ is suitably chosen to facilitate the conditional sampling problem.

- Consider a collection of $L+1$ modified SDE systems

$$dY_t^\ell = (1-\epsilon_\ell)b(Y_t^\ell)dt + \epsilon_\ell a(Y_t^\ell)dt + \sigma(Y_t^\ell)dB_t,$$

  $\ell = 0, \ldots, L$, with $\epsilon_\ell < \epsilon_{\ell+1}$, $\epsilon_0 = 0$ and $\epsilon_L = 1$.

# Drift homotopy

**Instead** of sampling directly from the density

$$g(X_{T_k}, Z_{T_k})p(X_{T_k}|X_{T_{k-1}}) \tag{7}$$

**Sample** from the density

$$g(Y^0_{T_k}, Z_{T_k})p(Y^0_{T_k}|X_{T_{k-1}})$$

and **gradually morph** the sample into a sample of (7) by sampling from the $\ell$ levels:

$$g(Y^\ell_{T_k}, Z_{T_k})p(Y^\ell_{T_k}|X_{T_{k-1}})$$

# A few comments

- The levels from 0 to $L-1$ are auxiliary and only serve the purpose of providing the sampler at level $L$ with a better initial condition. The final sampling is performed at the $L$th level which corresponds to the original SDE.

# A few comments

- The levels from 0 to $L-1$ are auxiliary and only serve the purpose of providing the sampler at level $L$ with a better initial condition. The final sampling is performed at the $L$th level which corresponds to the original SDE.

- The idea behind drift relaxation resembles the main idea behind *Homotopy Methods* used in deterministic optimization problems.

# A few comments

- The levels from 0 to $L-1$ are auxiliary and only serve the purpose of providing the sampler at level $L$ with a better initial condition. The final sampling is performed at the $L$th level which corresponds to the original SDE.

- The idea behind drift relaxation resembles the main idea behind *Homotopy Methods* used in deterministic optimization problems.

- The drift homotopy algorithm is similar to *Simulated Annealing (SA)* used in equilibrium statistical mechanics. However, instead of modifying a temperature as in SA, here we modify the drift of the system.

# Drift Homotopy algorithm

- Sample through MCMC the density $g(Y^0_{T_k}, Z_{T_k})p(Y^0_{T_k}|X_{T_{k-1}})$.
- For $\ell = 1, ..., L$ take the last sample from the $(\ell - 1)$st SDE and use it as in initial condition for MCMC sampling of the density

$$g(Y^\ell_{T_k}, Z_{T_k})p(Y^\ell_{T_k}|X_{T_{k-1}})$$

at the $\ell$th level.

- Keep the last sample at the $L$th level.

# MCMC step with drift homotopy appended

5. **Resampling:** Based on the current and previous observation.
6. **Drift homotopy MCMC step**: For $n = 1, \dots, N$ and $\lambda = 1, \dots, \Lambda$ choose a modified drift (possibly different for each $n$ and each $\lambda$). Construct through drift homotopy a Markov chain for $Y^n_{T_k}$ with initial value $X^n_{T_k}$ and stationary distribution

$$\prod_{\lambda=1}^{\Lambda} g_\lambda(Y^n_\lambda, Z_{\lambda,T_k}) p_\lambda(Y^n_\lambda | X^n_{\lambda,T_{k-1}}).$$

7. Set $X^n_{T_k} = Y^n_{T_k}$.
8. Set $k = k + 1$ and proceed to Step 1.

# Model: Double-well potential

- Consider the diffusion problem in a double well potential:

$$dX_t = -4X_t(X_t^2 - 1)dt + \frac{1}{2}dB_t \tag{8}$$

- The deterministic part of (8) describes a gradient flow for potential $U(x) = x^4 - 2x^2$ which has two minima at $\pm 1$.
- If the stochastic term is 0 then the solution wanders around one of the minima depending on the value of the initial condition.
- A weak stochastic term leads to rare transitions between the minima of the potential.
- Discretize (8) by an Euler-Maruyama scheme with step size $\Delta t = 10^{-2}$

# Model: Observation

- Observations are considered an additive Gaussian model:

$$Z_{t_k} = X_{t_k} + \xi_{t_k},$$

- Noise $\xi_{t_k} \sim \mathcal{N}(0, .01)$.
- Consider 10 observations in total at $t_k = k = 1, \cdots, 10$.
- Observations alternate between 1 and -1: $Z_{t_k}$ is around 1 if $k$ is odd, and $Z_{t_k}$ is around -1 if $k$ is even.
- Kang and **VM** (2013): $\xi_{t_k} \sim GMM$

# Potential $U$

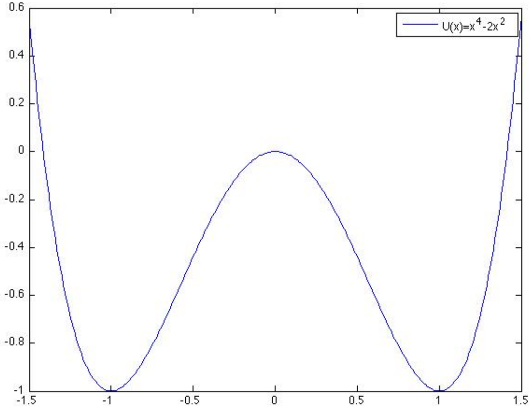



Figure: The potential which corresponds to the deterministic part of (8).

# Drift homotopy

- The difficulty in tracking the observations comes from the rate transitions between the two minima.
- Take $dY_t = b(Y_t)dt + \frac{1}{2}dB_t$.
- Choose $b(Y_t) = -c4Y_t(Y_t^2 - 1)$, where $0 < c < 1$.
- The drift corresponds to the potential $W(y) = c(y^4 - 2y^2)$.
- $W(y)$ has its minima also located at $\pm 1$ but the value at the minima is $-c$.
- This means that the wells corresponding to the minima are shallower
- Transitions between the two wells become more frequent.

# Potential $W$

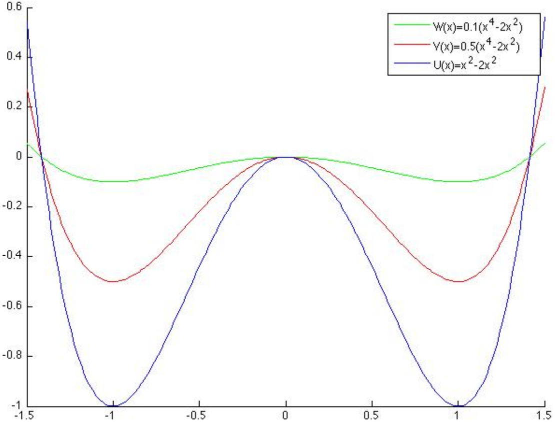



Figure: Potentials which correspond to the modified drift of (8).

# Drift homotopy

Let consider the SDE with the modified drifts and its corresponding $L$ levels,

$$dY_t^\ell = (1-\epsilon_\ell)b(Y_t^\ell)dt + \epsilon_\ell a(Y_t^\ell)dt + \frac{1}{2}dB_t \tag{9}$$

where $\ell = 0, \cdots, L$, $\epsilon_\ell = \frac{\ell}{L}$ for $\ell \neq 0$ and $\epsilon_\ell = 0$ when $\ell = 0$.

# Numerical Results

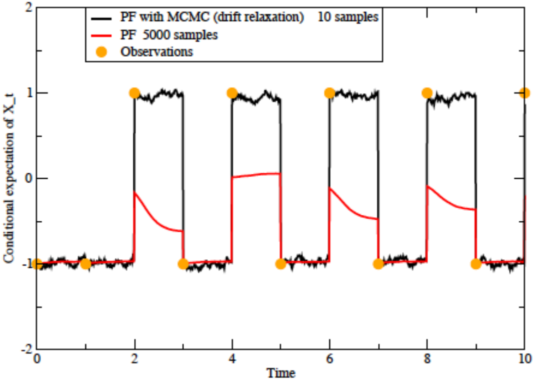



Figure: Comparison of the conditional expectation of $X_t$ computed by the generic PF and the MCMC PF.

# Model: Dynamics

- At each time $t$ we have a total of $\Lambda_t$ targets
- The evolution of the $\lambda$th target ($\lambda = 1, \ldots, \Lambda_t$) is given by the near constant velocity model:

$$\mathbf{x}_{\lambda,t} = \mathbf{A}\mathbf{x}_{\lambda,t-1} + \mathbf{B}\mathbf{v}_{\lambda,t} = [x_{\lambda,t}, \dot{x}_{\lambda,t}, y_{\lambda,t}, \dot{y}_{\lambda,t}]^*,$$

$$\mathbf{A} = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} \text{ and } \mathbf{B} = \begin{bmatrix} T^2/2 & 0 \\ T & 0 \\ 0 & T^2/2 \\ 0 & T \end{bmatrix}, \quad (10)$$

- $T = 1$ is the time between observations.
- $\mathbf{v}_{\lambda,t}$ i.i.d $\mathcal{N}(0, \mathbf{\Sigma_v})$, $\mathbf{\Sigma}_v = diag(\sigma_x^2, \sigma_y^2)$, $\sigma_x^2 = \sigma_y^2 = 1$.

# Model: Observation

- Bearing $\theta$ and range $r$ of a target. Given the $\lambda$th target propagates the $m$th observation

$$\mathbf{Z}_{m,t} = \left[ \begin{array}{c} \arctan(\frac{y_{\lambda,t}}{x_{\lambda,t}}) \\ (x_{\lambda,t}^2 + y_{\lambda,t}^2)^{1/2} \end{array} \right] + \mathbf{w}_{m,t}. \quad (11)$$

- $\mathbf{w}_{m,t} \sim \mathcal{N}(0, \mathbf{\Sigma}_w)$, where $\mathbf{\Sigma}_w = diag(\sigma_\theta^2, \sigma_r^2)$
- For the numerical experiments we chose $\sigma_\theta^2 = 10^{-4}$ and $\sigma_r^2 = 1$.

# Model: Comments

- The synthesized target tracks were created by evolving a number of targets according to (10) and recording the state of each target at each step.
- The observations were created in bearing and range space $\theta, r$ by using (11).
- The number of targets at each observation instant is: $\Lambda_0 = \cdots = \Lambda_{100} = 10$

# Drift homotopy

The dynamics of the targets for the modified drift system at the $\ell$th level are given by

$$\mathbf{y}^{\ell}_{\lambda,t} = \mathbf{A}\mathbf{y}^{\ell}_{\lambda,t-1} + \mathbf{c}^{\ell} + \mathbf{B}\mathbf{v}_{\lambda,t},$$

where $y_{1,\lambda,t}, y_{3,\lambda,t}$ and $y_{2,\lambda,t}, y_{4,\lambda,t}$ are the $xy$ positions and velocities respectively for the $\lambda$th target at time $t$.

The matrix $\mathbf{c}^{\ell}$ is given by

$$\mathbf{c}^{\ell} = (1-\epsilon_l)[\mu_x \frac{T^2}{2}, \mu_x T, \mu_y \frac{T^2}{2}, \mu_y T]^*$$

where $\epsilon_\ell \in [0,1],\ \ell = 0,\dots,L$, with $\epsilon_\ell < \epsilon_{\ell+1}$, $\epsilon_0 = 0$ and $\epsilon_L = 1$. In the numerical experiments we chose $L = 10$ i.e. 10 levels for the drift homotopy.

# Drift homotopy

where $\mu_x^n$ and $\mu_y^n$ for the $n$-th sample as

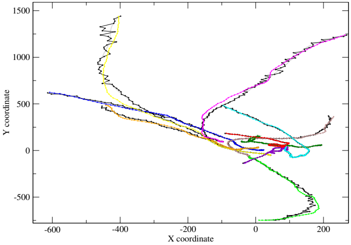
$$\mu_{x,\lambda}^n = \frac{\frac{1}{N}\sum_{n'=1}^{N}(y_{1,\lambda,k-1}^{n'} + y_{2,\lambda,k-1}^{n'}T) - y_{1,\lambda,k-1}^n}{T^2/2} - \frac{2y_{2,\lambda,k-1}^n}{T}$$

and

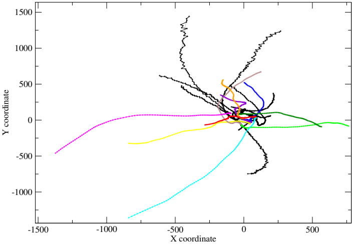
$$\mu_{y,\lambda}^n = \frac{\frac{1}{N}\sum_{n'=1}^{N}(y_{3,\lambda,k-1}^{n'} + y_{4,\lambda,k-1}^{n'}T) - y_{3,\lambda,k-1}^n}{T^2/2} - \frac{2y_{4,\lambda,k-1}^n}{T}.$$

This choice of modified drift corresponds to a mean drift while at the same time offsetting the individual sample's properties. More sophisticated drift choices will be explored in future work.

# Numerical Results




(a) MCMC PF with 20 samples




(b) Generic PF with 5000 samples

Figure: Grey lines: true target, Crosses: observations, Colored lines: estimates

# Error

The RMS error per target (RMSE) is defined with reference to the true target tracks by the formula

$$RMSE(t) = \sqrt{\frac{1}{\Lambda_t} \sum_{k=1}^{\Lambda_t} \| \mathbf{x}_{k,t} - E[\mathbf{x}_{k,t} | Z_1, \ldots, Z_t] \|^2} \qquad (12)$$

$\mathbf{x}_{k,t}$ is the true state vector for target $k$. $E[\mathbf{x}_{k,t} | Z_1, \ldots, Z_t]$ is the conditional expectation estimate calculated with the MCMC or generic particle filter depending on whose filter's performance we want to calculate.

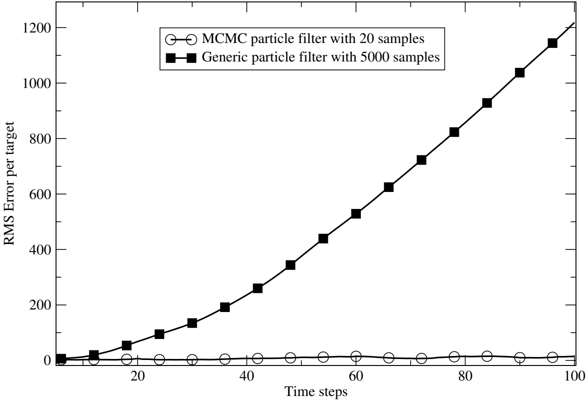



Figure: Comparison of RMS error per target for the MCMC particle filter and the generic particle filter.

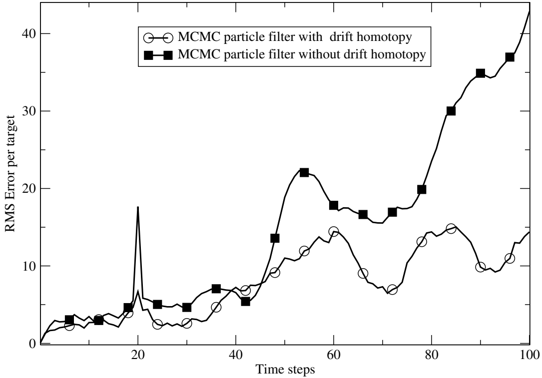



Figure: Comparison of RMS error per target for the MCMC particle filter with drift homotopy ($L = 10$) and the MCMC particle filter without drift homotopy ($L = 0$).

# Conclusion

- Drift homotopy: new MCMC method appended
- MCMC particle filter follows accurately the targets
- There is no ambiguity in the identification of the target tracks.
- The accuracy of the generic particle filter's estimate deteriorates fast.
- Drift homotopy error grows slower in comparison without ($\ell = 0$).

# References


- **VM** and P. Stinis. A drift homotopy Monte Carlo approach to particle filtering for multi-target tracking, *Journal of Computational Physics*, 231(2), pp. 602-611, 2012.
- K. Kang and **VM**. Drift homotopy methods for a nonGaussian filter, *The Proceedings of Information Fusion*, pp. 1088 - 1094, 2013.
- R.P.S. Mahler and **VM**. Tracking spawning objects. *IET Radar, Sonar & Navigation*, 7(3), pp.321-331, 2013.
- **VM** and J. Xiong. Large deviations for optimal filtering with fractional Brownian motion. *Stochastic Processes and their Applications*, 123 (6), pp. 2340-2352, 2013.

# Thanks to my Research Sponsors

The Leverhulme Trust

SIMONS FOUNDATION
Advancing Research in Basic Science and Mathematics

IMA Institute for Mathematics and its Applications

LOCKHEED MARTIN
We never forget who we're working for™

THE UNIVERSITY of TENNESSEE UT KNOXVILLE

NIMBioS
National Institute for Mathematical and Biological Synthesis

# Conditional path sampling

- Consider an SDE: $dX_t = a(X_t)dt + \sigma(X_t)dB_t$
- Discretize the SDE in [0,T] using a step size $\Delta t = T/I$
- We want to construct in the time interval $[0, T]$ sample paths from the SDE such that the endpoints are distributed according to densities $h(X_0)$ and $g(X_T)$, i.e. need to sample the density

$$h(X_{T_0}) \prod_{i=1}^{I} p(X_{T_i}|X_{T\,i-1}) g(X_T)$$

- Assuming that the transitions densities $p(X_{T_i}|X_{T\,i-1})$ can be evaluated then we can use MCMC.

# $\ell$th density for the double-well potential problem

We can replace the sampling the sampling of $g(X_{T_k}, Z_{T_k})p(X_{T_k}|X_{T_{k-1}})$ by sampling from the density

$$\exp\Big[-\frac{(Z_{T_k} - X_T^n(\{\Delta B_i^n\}_{i=0}^{I-1}))^2}{2\sigma_\xi^2} + \sum_{i=0}^{I-1} \frac{(\Delta B_i^n)^2}{2\Delta t}\Big]$$

Instead we use drift homotopy to produce samples by considering the $L$ system of modified SDEs:

$$\exp\Big[-\frac{(Z_{T_k} - Y_T^{\ell,n}(\{\Delta B_i^{\ell,n}\}_{i=0}^{I-1}))^2}{2\sigma_\xi^2} + \sum_{i=0}^{I-1} \frac{(\Delta B_i^{\ell,n})^2}{2\Delta t}\Big]$$

# $\ell$th density for multi-target tracking problem

For the $n$th sample, the density to be sampled at the $l$-th level is

$$\prod_{\lambda=1}^{\Lambda_{t_k}} g_x(z_{\lambda,k}^{\ell,n}, Z_{\theta,\lambda,k}) g_y(z_{\lambda,k}^{\ell,n}, Z_{r,\lambda,k}) p(z_{\lambda,k}^{\ell,n} | z_{\lambda,k-1}^{\ell,n})$$

$$\propto \prod_{\lambda=1}^{\Lambda_{t_k}} \exp\Bigg( -\Bigg\{ \frac{\big(Z_{\theta,\lambda,k} - \arctan(\frac{z_{3,\lambda,k}^{\ell,n}}{z_{1,\lambda,k}^{\ell,n}})\big)^2}{2\sigma_\theta^2} + \frac{\big(Z_{r,\lambda,k} - ({z_{1,\lambda,k}^{\ell,n}}^2 + {z_{3,\lambda,k}^{\ell,n}}^2)^{1/2}\big)^2}{2\sigma_r^2} + \frac{(v_{x,\lambda,k}^n)^2}{2\sigma_x^2} + \frac{(v_{y,\lambda,k}^n)^2}{2\sigma_y^2} \Bigg\} \Bigg), \quad (13)$$