A Geodesic Flow Particle Filter for Non-Thresholded Measurements

Chris Kreucher  
Integrity Applications Inc.  
Ann Arbor, Michigan 48108  
Email: ckreuche@umich.edu

Kristine Bell  
Metron, Inc.  
Reston, Virginia 20190  
Email: bell@metsci.com

Abstract—This paper derives a particle filter with flow induced by a log-homotopy for non-thresholded measurements (i.e., a track-before-detect log-homotopy particle filter). We have elected to use the Geodesic approach to particle flow and study both scaled-identity and Gaussian diffusion. We show the performance of the new filter provides order-of-magnitude tracking improvement over SIR filtering on a 2D tracking problem. While the homotopy approach requires significantly more computations per particle as it numerically solves a stochastic differential to flow each particle, we still show a significant improvement in performance per flop. We expect this performance improvement to widen, perhaps significantly, with higher dimensional state space. The numerics are discussed further in the paper.

I. INTRODUCTION

A particle filter approximates a probability distribution by a set of samples, or particles, and corresponding weights [1], [2]. In the motion update step of Bayesian filtering, the particles are propagated according to the motion model, while the weights remain fixed. In the information update step, the particles remain fixed while the weights are updated by Bayes rule. This involves multiplying the weights by the likelihood function and then normalizing so they sum to one. Particle degeneracy occurs when the likelihood function is concentrated on only a few particles and the information-updated weights are mostly zeros [3]. The standard remedy is resampling, in which the particles with the largest weights are replicated in proportion to their weights, and the weights of the resampled particles are set to be equal [1], [2]. This type of “sampling importance resampling” (SIR) information update is a computationally intensive process that incurs performance degradation in many practical problems.

In a series of papers [3]–[15], Daum has introduced a new homotopy-based method to implement the information update. In this approach, instead of using measurements to simply update the weight of particles from the prior, measurements are used to flow prior particles to a posterior location. Among other things, this addresses the particle degeneracy problem inherent in SIR particle filtering which occurs when the observation likelihood is narrow compared to the prior.

Several practical implementations have been recently reported [16]–[19]. These implementations all use a conventional detect-then-track approach which takes an input data surface and thresholds to a set of detections. The detections (threshold exceedances) are then associated to the tracking filter to produce state estimates over time. In contrast, here we discuss the design and implementation of a track-before-detect log-homotopy flow particle filter, which incorporates the raw(-er) non-thresholded observations directly into the filter.

In this paper, we derive a homotopy-based particle flow for a non-thresholded sensor model and show significant improvement in tracking performance. The paper proceeds as follows. In Section II, we briefly review Daum’s approach and give the details of a special case referred to as Geodesic Flow with scaled-Identity diffusion. Next, Section III gives the details of our pixelated non-thresholded measurement model. Third, in Section IV, we specialize the flow equations specialized to our model. A detailed derivation are omitted for lack of space. Section V provides a simulation which compares the performance of the new filter to an SIR filter on a model problem. Finally, Section VI concludes.

II. A SUMMARY OF THE GEODESIC FLOW APPROACH

The homotopy-based approach to particle filtering [5], [8] starts by defining a flow of the conditional probability density function on state vector \( x \) with respect to a parameter \( \lambda \)

\[
\log p(x, \lambda) = \log g(x) + \lambda \log h(x) - \log K(\lambda),
\]

where \( g(x) \) is the prior density, \( h(x) \) is the likelihood and \( K(\lambda) \) is a normalization. The conditional probability \( p(x, \lambda) \) can be seen to move between the prior when \( \lambda = 0 \) and the posterior when \( \lambda = 1 \).

It is distinguished from the standard particle filtering approach in that particles are flowed from a prior location to a posterior location using the measurements. In particular, the approach further supposes [6] the flow of particles from prior to posterior obeys the Ito stochastic differential equation

\[
dx = f(x, \lambda)d\lambda + dw,
\]

where the covariance of the process noise is given by \( Q(x, \lambda) \).

Daum’s most recent work focuses on a special case [7], [8] which results in the Geodesic flow

\[
f(x, \lambda) = -\left[ \frac{\partial^2 \log p(x, \lambda)}{\partial x^2} \right]^{-1} \left( \frac{\partial \log h(x)}{\partial x} \right)^T.
\]
Coupled with $Q(x, \lambda)$ this function defines the stochastic differential equation (SDE) in variables $x$ and $\lambda$ that flows particles from their prior location to their posterior location.

There are many ways to pick an optimal $Q(x, \lambda)$ as discussed in [14], [15]. We have elected to explore two cases. First, we look at $Q(x, \lambda)$ derived using a Gaussian assumption on the densities [15],

$$Q(x, \lambda) = \left[ P^{-1} + \lambda H^T R^{-1} H \right]^{-1} \lambda H^T R^{-1} H \left[ P^{-1} + \lambda H^T R^{-1} H \right]^{-1},$$

where $P$ comes from the prior distribution and $R$ is a matched to the sensor model.

Next, we investigate the case where $Q(x, \lambda)$ is a positive constant multiple of the identity matrix (given $f(x, \lambda)$, $x$, and $\lambda$) [14]. In this case, the constant is approximately

$$\alpha \approx \frac{2 \left\| \partial \log p(x, \lambda) \partial f(x, \lambda) \right\|}{\left\| \partial \partial_x \left( \frac{\partial p(x, \lambda)}{\partial x} \right) / p(x, \lambda) \right\|}.$$  

(5)

Together equations (3) and (4) or (5) completely specify the flow of particles from prior to posterior.

III. NON-THRESHOLDED MEASUREMENT MODEL

This section defines the sensor likelihood $h(x)$ for our non-thresholded sensor measurements. We specialize here to the case where the raw sensor data has been processed to create an $N \times M$ array of pixelated input data (for example, a Range/Doppler Map or an Infrared Array). The measurements are then a set of intensities on the $MN$ sensor pixels and will be denoted $z = [z_1, z_2, \ldots, z_{MN}]$. The model $h(x)$ defines the statistics of the pixels, and is coarsely described as modeling the expected intensity in each pixel as related to the distance from the pixel to the projection of the target on to sensor space. The sensor impulse response (IPR) defines how the expected intensity falls off as the distance from the target.

The target is characterized by state $x$ and the mapping $m(x) = [i_x \ j_x]^T$ projects the target state $x$ to the sensor space (pixels). In general, this projection will be a nonlinear mapping (for example, from $x = [x \ x \ y \ y]$ coordinates to range and range-rate). The distance between the projection of $x$ and a sensor pixel $(i, j)$ will be denoted

$$\delta_{ij}(x) = m(x) - [i \ j]^T.$$  

(6)

For the purposes of exposition, here we will use simply use $m(x) = x$, i.e., we measure cells with unit thickness in the same state space as the tracker.

We model intensities of the pixels as having Rayleigh statistics. The expected intensity in pixel $(i, j)$ is given by its distance from the projection of the target state $x$, weighted by the target IPR. We elect to use an exponential for the target IPR and define its value in pixel $(i, j)$ as

$$I_{ij}(x) = e^{-\frac{1}{2} \delta_{ij}^2(x) \lambda_{ij}(x)^{-1}}.$$  

(7)

The Rayleigh intensity parameter in pixel $(i, j)$ is then

$$\lambda_{ij}(x) = \lambda_b + (\lambda_t - \lambda_b) I_{ij}(x).$$  

(8)

This model captures the fact that the intensity parameter in pixel $(i, j)$ is $\lambda_b$ (background intensity) in pixels very far from the target, $\lambda_t$ (target intensity) for a pixel centered exactly at the target, and falls off as dictated by the IPR covariance $S$. The following figures show example measurement scans in the $1D$ and $2D$ cases. Each pixel has intensity drawn from a Rayleigh random variable with mode given by eq. (8).

![Example measurement scan](image1)

Fig. 1. An example one-dimensional measurement scan with $\lambda_b = 4$, $\lambda_t = 1$ and $s = 16$. Measurements are distributed Rayleigh with intensity described by the distance from the true target projection and the IPR.

![Example measurement scan](image2)

Fig. 2. An example two-dimensional measurement scan with $\lambda_b = 4$, $\lambda_t = 1$, and $s = 16$. Measurements are distributed Rayleigh with intensity described by the distance from the true target projection and the IPR.

With this as background, we can now write the full non-thresholded measurement model explicitly as

$$h(x) = p(z|x) = \prod_{ij} p(z_{ij}|x) = \prod_{ij} \frac{z_{ij}}{\lambda_{ij}(x)} e^{-\frac{z_{ij}^2}{2 \lambda_{ij}(x)}},$$  

(9)
Particle flow from prior to posterior is effected by solving the stochastic differential equation defined by eq. (3) and eq. (5). We discuss each of these terms in turn.

A. The deterministic component of the flow, \( f(x, \lambda) \)

The flow (eq. 3) for non-thresholded measurements is specified as follows. Using the identity

\[
\frac{\partial^2 \log p(x, \lambda)}{\partial x^2} = \lambda \frac{\partial^2 \log h(x)}{\partial x^2} + \frac{\partial^2 \log g(x)}{\partial x^2},
\]

the flow is seen to be defined through the partials of the likelihood \( h(x) \) and the prior \( g(x) \).

The required partials of \( h(x) \) can be found in a straightforward manner using the likelihood (eq. 9), Rayleigh intensity (eq. 8) and IPR (eq. 7) definitions given earlier. For example the (scalar) partials with respect variables \( a \) and \( b \) are

\[
\frac{\partial \log h(x)}{\partial a} = \sum_{ij} \left( \frac{z_{ij}^2 - 2 \lambda_{ij}(x)}{2 \lambda_{ij}(x)} \right) \frac{\partial \lambda_{ij}(x)}{\partial a},
\]

and

\[
\frac{\partial^2 \log h(x)}{\partial a \partial b} = \sum_{ij} \left( \frac{z_{ij}^2 - 2 \lambda_{ij}(x)}{2 \lambda_{ij}(x)} \right) \frac{\partial^2 \lambda_{ij}(x)}{\partial a \partial b} + \frac{\lambda_{ij}(x) - z_{ij}^2}{\lambda_{ij}^3(x)} \frac{\partial \lambda_{ij}(x)}{\partial a} \frac{\partial \lambda_{ij}(x)}{\partial b},
\]

where the partials of \( \lambda_{ij}(x) \) are scaled versions of the partials of the IPR \( I_{ij}(x) \), i.e.,

\[
\frac{\partial \lambda_{ij}(x)}{\partial a} = (\lambda_t - \lambda_b) \frac{\partial I_{ij}(x)}{\partial a},
\]

and so on. The partials of \( I_{ij}(x) \) come directly from the definition.

Finally, let \( \mu \) and \( P \) be the empirical mean and covariance of the particles used to represent the prior \( g(x) \). Under a Gaussian approximation, we find \( \frac{\partial^2 \log g(x)}{\partial x^2} = -P^{-1} \). This now completely specifies the flow function \( f(x, \lambda) \) for the non-thresholded model.

B. The Diffusion, \( Q(x, \lambda) \)

In the case of scaled-Identity diffusion, the diffusive component of the flow is defined by the scale codified in eq. (5). Its computation requires the specification of

\[
\frac{\partial \log p(x, \lambda)}{\partial x} \frac{\partial f(x, \lambda)}{\partial x}
\]

and

\[
\frac{\partial}{\partial x} \left[ \text{div} \left( \frac{\partial p(x, \lambda)}{\partial x} \right) / p(x, \lambda) \right].
\]

Both terms can be seen to be a function of \( \log p(x, \lambda), \log g(x), \log h(x) \) and their partials with respect to state \( x \). A detailed derivation of these terms is omitted because of lack of space.

In the case of a Gaussian approximation for diffusion, we instead use \( P \) computed from the prior and an \( R \) which approximates the measurement model. Here we use \( R \) matched to the IPR.

V. SIMULATION

This section shows the results of a simulation comparing the new non-thresholded Geodesic flow particle filter with a SIR particle filter. Both trackers use the exact same temporal prediction, measurement model, and measurements. The only distinction between the filters is that in the Geodesic flow filter, the particles are flowed from their prior position to a posterior position by on solving the Geodesic flow SDE. In contrast, in the SIR filter, particles stay at their prior position and have weights updated via Bayes rule.

A. Illustrative Images

We first present images from a 1D simulation which illustrate the internals of the Geodesic flow. Figure 3 illustrates how particles flow from prior to posterior location as the SDE is solved from \( \lambda = 0 \) to \( \lambda = 1 \). The received measurements are shown at right. This illustrates how the relatively broad prior is concentrated to a narrower posterior focused around where the measurements indicate the target is.

![Figure 3. An illustration in 1D of how the Geodesic method flows particles from their prior location to their posterior location as \( \lambda \) goes from 0 to 1.](image-url)
Fig. 4. An Comparison of the posteriors. Top: Measurements and the Rayleigh mode. Mid: The prior and the Geodesic flow posterior. Bot: The prior (identical to the geodesic prior) and the SIR posterior.

Fig. 5. Zoomed to show detail. Since there are no prior particles near the measurement, the SIR puts full support on the closest particle. The Geodesic method flows particles to the correct region and avoids this pathology.

B. On the Numerical Solution of the SDE

The SDE is solved numerically (i.e., particles are flowed from prior to posterior location) using the Euler-Maruyama method [21]. The implementation uses software based on the MatLab SDE toolset [20]. Since most of the important flow happens near $\lambda = 0$, we found it sufficient to simply space the points logarithmically between $1e-5$ and 1. Specifically, the SDE solution was effected by discretizing $\lambda$ log-spaced with 41 points between $\lambda = 10^{-5}$ to $\lambda = 1$ (i.e., $\lambda = 10^{-5} + \delta$, $\delta = 0 \cdots 40$). Selection of the SDE discretization for homotopic flow is an active area of investigation. For example, recent work [13] has focused on adaptively choosing the step size.

C. Simulation Result

The simulation is described as follows. A target is characterized by its 2D $(x, y)$ state. The true state simply diffuses over time. We use a 100 time step simulation. At each time, non-thresholded measurements are made on a 2D grid. The Rayleigh intensity in each grid cell is given by eq. (8), i.e., it is defined by the true target location of the target and the IPR.

The performance of the tracker is measured by the $RMSE$ error between true state and predicted state over the simulation window. We compare the performance as a function of particle count for the two approaches.

Figure 6 shows the result of a Monte Carlo simulation comparing the $RMSE$ performance of the Geodesic particle filter to an SIR particle filter in terms of number of particles. For the simulation we used IPR $S = sI_{2x2}$, with $s = 32$. The sensor makes measurements on a $300 \times 300$ array. We illustrate the performance of both the Gaussian and caled-Identity diffusions.

D. Discussion

The Geodesic approach requires solving a stochastic differential equation (SDE) to propagate each particle. We have chosen to discretize with 31 steps. Each step requires calculation of the flow and diffusion terms as outlined earlier. In contrast, the SIR requires only an evaluation of the likelihood function and a multiply for each particle and then a resample of the entire collection. Thus, on a per-particle basis, the geodesic flow requires significantly more computations. With coarsely optimized MatLab code, the geodesic approach with
Fig. 6. Monte Carlo Comparisons of the tracking performance versus number of particles for the SIR and Geodesic flow approaches for the 2D example. Top: $\lambda_t = 10$. Bottom: $\lambda_t = 4$. Particle diffusion during the flow is done via either the scaled-Identity method [14] or using a Gaussian model [15]. We find that the Geodesic approach needs about $\approx 250x$ fewer particles to reach the RMSE asymptote.

scaled-Identity diffusion requires $\approx 75x$ more computations per particle then the SIR approach and the geodesic approach with approximate Gaussian diffusion requires $\approx 30x$ more computations per particle. Figure 7 shows the performance curve from Figure 6 at $\lambda_t = 10$ now compared against CPU time, as measured using coarsely optimized MatLab code. In light of the significant RMSE improvement shown in Figure 6, the geodesic approach still provides a computational advantage over the SIR method. Earlier work [5] shows that the computational advantage grows as the dimension of the state space.

Due to discretization of $\lambda$, there are some occasions when the steps are too coarse and the flow is poor. In our current implementation, we trap for such events by simply looking at the numerical gradient of the particle position during the solution. When this happens, the particle is re-flowed from a prior position to its posterior state.

VI. CONCLUSION

This paper has developed and illustrated by simulation a track-before-detect log-homotopy particle filter by reducing to practice work by Daum. We have elected to use the Geodesic approach to particle flow coupled with scaled-identity diffusion. We have found the performance of the new filter provides order-of-magnitude tracking improvement over SIR filtering on a per-particle basis in simulation on a 2D tracking problem. While the homotopy approach requires more computations per particle, on balance, it still provides an improvement per flop.

ACKNOWLEDGMENT

The authors would like to thank Dr. Fred Daum for his helpful comments during the initial stages of this investigation.

REFERENCES


