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 Scientific Solutions, 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]. A standard "sampling importance resampling" (SIR) particle filter is comprised of two steps. In the motion update step, particles are propagated according to the motion model, while the weights remain fixed. In the information update step, particles remain fixed while the weights are updated by Bayes rule. This involves multiplying the weights by the likelihood function and normalizing. Particle degeneracy occurs when the likelihood is concentrated on only a few particles and the information-updated weights are mostly zero [3]. The standard remedy is resampling, where particles with the largest weights are replicated in proportion to their weights, and the weights of all particles are set to be equal [1], [2]. This step leads to a 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.

Several practical implementations have been recently reported [16]–[19]. These implementations all use a 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. Our evaluation shows 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. Next, Section III gives the details of our pixelated non-thresholded measurement model. Third, in Section IV, we specialize the flow equations to our model. A detailed derivation is omited for lack of space. Section V provides a simulation comparing the performance of the new filter to an SIR filter. 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 $\mathbf{x}$ with respect to a parameter λ

$$\log p(\mathbf{x}, \lambda) = \log g(\mathbf{x}) + \lambda \log h(\mathbf{x}) - \log K(\lambda), \quad (1)$$

where $g(\mathbf{x})$ is the prior, $h(\mathbf{x})$ is the likelihood and $K(\lambda)$ is a normalization. $p(\mathbf{x}, \lambda)$ can be seen to move between the prior when $\lambda = 0$ and the posterior when $\lambda = 1$.

It is distinguished from standard particle filtering as particles are flowed from a prior to a posterior location using the measurements. The approach supposes [6] the flow of particles obeys the Ito stochastic differential equation

$$d\mathbf{x} = \mathbf{f}(\mathbf{x}, \lambda)d\lambda + d\mathbf{w}, \quad (2)$$

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

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

$$\mathbf{f}(\mathbf{x}, \lambda) = -\left[\frac{\partial^2 \log p(\mathbf{x}, \lambda)}{\partial \mathbf{x}^2}\right]^{-1} \left(\frac{\partial \log h(\mathbf{x})}{\partial \mathbf{x}}\right)^T. \quad (3)$$

Coupled with $Q(\mathbf{x}, \lambda)$ this defines the stochastic differential equation (SDE) in variables $\mathbf{x}$ and λ that flows particles from their prior location to their posterior location.

There are many ways to pick an optimal $Q(\mathbf{x}, \lambda)$ [14]. We explore two cases. First, we look at $Q(\mathbf{x}, \lambda)$ derived using a Gaussian assumption on the densities [15], i.e., $Q(\mathbf{x}, \lambda)$ is

$$[P^{-1} + \lambda H^T R^{-1} H]^{-1} H^T R^{-1} H [P^{-1} + \lambda H^T R^{-1} H]^{-1}, \quad (4)$$

978-1-4673-8823-8/17/$31.00 ©2017 IEEE

where P comes from the prior and R from the sensor model.

We also investigate the case where $Q(\mathbf{x}, \lambda)$ is a positive constant multiple of the identity matrix (given $f(\mathbf{x}, \lambda)$, $\mathbf{x}$, and λ) [14]. In this case, the constant is approximately

$$\alpha \approx \frac{2\left\| \dfrac{\partial \log p(\mathbf{x},\lambda)}{\partial \mathbf{x}} \dfrac{\partial \mathbf{f}(\mathbf{x},\lambda)}{\partial \mathbf{x}} \right\|}{\left\| \dfrac{\partial}{\partial \mathbf{x}} \left[div\left(\dfrac{\partial p(\mathbf{x},\lambda)}{\partial \mathbf{x}} \right) / p(\mathbf{x},\lambda) \right] \right\|}. \tag{5}$$

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 $h(\mathbf{x})$ 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 $z = [z_1, z_2, \cdots, z_{MN}]$. The model $h(\mathbf{x})$ 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 $\mathbf{x}$, and the mapping $m(\mathbf{x}) = [i_{\mathbf{x}} \quad j_{\mathbf{x}}]^T$ projects the target state $\mathbf{x}$ to the sensor space (pixels). In general, this projection will be a nonlinear mapping. The distance between the projection of $\mathbf{x}$ and a sensor pixel (i, j) will be denoted $\delta_{ij}(\mathbf{x}) = m(\mathbf{x}) - [i \quad j]^T$. For the purposes of exposition, we use $m(\mathbf{x}) = \mathbf{x}$, i.e., we measure cells with unit thickness in the same space as $\mathbf{x}$.

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 $\mathbf{x}$, weighted by the target IPR. We elect to use an exponential for the target IPR and define its value in pixel (i, j) as

$$I_{ij}(\mathbf{x}) = e^{-\frac{1}{2}\delta_{ij}^T(\mathbf{x}) S^{-1} \delta_{ij}(\mathbf{x})}. \tag{6}$$

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

$$\lambda_{ij}(\mathbf{x}) = \lambda_b + (\lambda_t - \lambda_b) I_{ij}(\mathbf{x}). \tag{7}$$

This model captures the fact that the intensity parameter in pixel (i, j) is λ_b (the background intensity) in pixels very far from the target. It is λ_t (the target intensity) for a pixel centered exactly at the target, and falls off as dictated by the IPR covariance S.

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

$$h(\mathbf{x}) \doteq p(z|\mathbf{x}) = \prod_{ij} p(z_{ij}|\mathbf{x}) = \prod_{ij} \frac{z_{ij}}{\lambda_{ij}(\mathbf{x})} e^{\frac{-z_{ij}^2}{2\lambda_{ij}(\mathbf{x})}}. \tag{8}$$

The following figures show example scans in $1D$ and $2D$ cases. Each pixel has intensity drawn from a Rayleigh random variable with mode given by eq. (7).

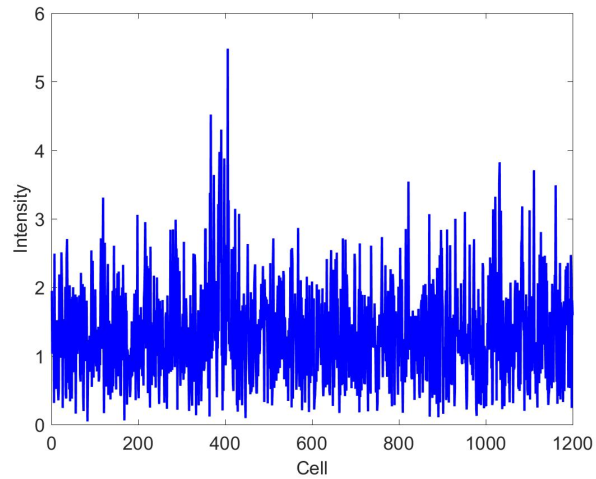

Fig. 1. An example one-dimensional measurement scan with $\lambda_t = 4$, $\lambda_b = 1$ and $s = 16$. Measurements are distributed Rayleigh with intensity described by the distance from the true target projection and the IPR.

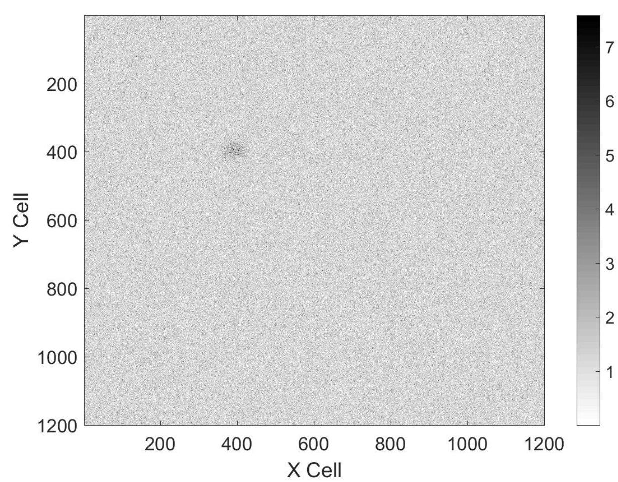

Fig. 2. An example two-dimensional measurement scan with $\lambda_t = 4$, $\lambda_t = 1$, and $s = 16$. Measurements are distributed Rayleigh with intensity described by the distance from the true target projection and the IPR.

IV. Geodesic Flow for Non-thresholded Measurements

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, $\mathbf{f}(\mathbf{x}, \lambda)$

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

$$\frac{\partial^2 \log p(\mathbf{x},\lambda)}{\partial \mathbf{x}^2} = \lambda \frac{\partial^2 \log h(\mathbf{x})}{\partial \mathbf{x}^2} + \frac{\partial^2 \log g(\mathbf{x})}{\partial \mathbf{x}^2}, \tag{9}$$

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

The required partials of $h(\mathbf{x})$ can be found in a straightforward manner using the likelihood (eq. 8), Rayleigh intensity (eq. 7) and IPR (eq. 6) definitions given earlier. For example the (scalar) partials with respect variables a and b are

$$\frac{\partial \log h(\mathbf{x})}{\partial a} = \sum_{ij} \left(\frac{z_{ij}^2 - 2\lambda_{ij}(\mathbf{x})}{2\lambda_{ij}^2(\mathbf{x})} \right) \frac{\partial \lambda_{ij}(\mathbf{x})}{\partial a} \tag{10}$$

and

$$\frac{\partial^2 \log h(\mathbf{x})}{\partial a \partial b} = \sum_{ij} \left(\frac{z_{ij}^2 - 2\lambda_{ij}(\mathbf{x})}{2\lambda_{ij}^2(\mathbf{x})} \right) \frac{\partial^2 \lambda_{ij}(\mathbf{x})}{\partial a \partial b} + \left(\frac{\lambda_{ij}(\mathbf{x}) - z_{ij}^2}{\lambda_{ij}^3(\mathbf{x})} \right) \frac{\partial \lambda_{ij}(\mathbf{x})}{\partial a} \frac{\partial \lambda_{ij}(\mathbf{x})}{\partial b}, \tag{11}$$

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

$$\frac{\partial \lambda_{ij}(\mathbf{x})}{\partial a} = (\lambda_t - \lambda_b) \frac{\partial I_{ij}(\mathbf{x})}{\partial a}, \tag{12}$$

and so on. The partials of $I_{ij}(\mathbf{x})$ come directly from eq. (6).

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

B. The Diffusion, $Q(\mathbf{x}, \lambda)$

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

$$\frac{\partial \log p(\mathbf{x}, \lambda)}{\partial \mathbf{x}} \frac{\partial \mathbf{f}(\mathbf{x}, \lambda)}{\partial \mathbf{x}} \tag{13}$$

and

$$\frac{\partial}{\partial \mathbf{x}} \left[div \left(\frac{\partial p(\mathbf{x}, \lambda)}{\partial \mathbf{x}} \right) / p(\mathbf{x}, \lambda) \right]. \tag{14}$$

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

In the case of a Guassian 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.

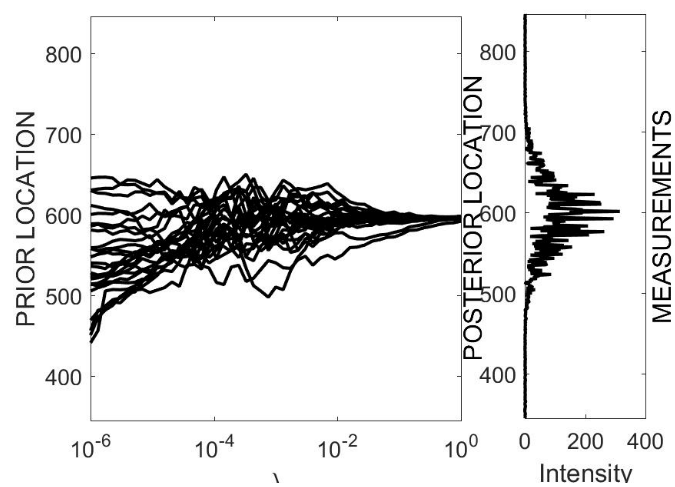

Fig. 3. An illustration in $1D$ of how the Geodesic method flows particles from their prior location to their posterior location as λ goes from 0 to 1.

Figures 4 and 5 illustrate why the Geodesic flow approach improves performance over the SIR approach. These figures show the broad prior/narrow observation likelihood case. We find that the SIR approach has the well-known particle degeneracy issue. Only the one particle that is closest to the true location gets all of the weight. On the other hand, the Geodesic flow method is able to move particles from the prior to the correct posterior location.

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 λ log-spaced with 31 points between $\lambda = 10^{-5}$ to $\lambda = 1$ (i.e., $\lambda = 10^{-5+.167\delta}, \delta = 0 \cdots 30$). Selection of the SDE discretization for homotopic

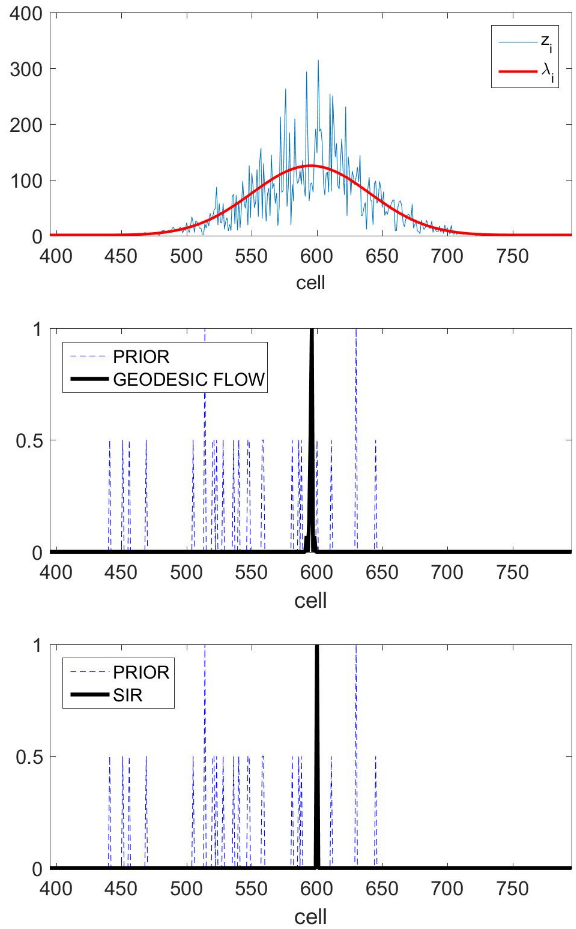

Fig. 4. An Comparison of the posteriors. Top: Measurements and the Rayleigh mode. Mid: The prior and the Geodesic flow posterior. Bot: The prior (identical to the geodesic prior) and the SIR posterior.

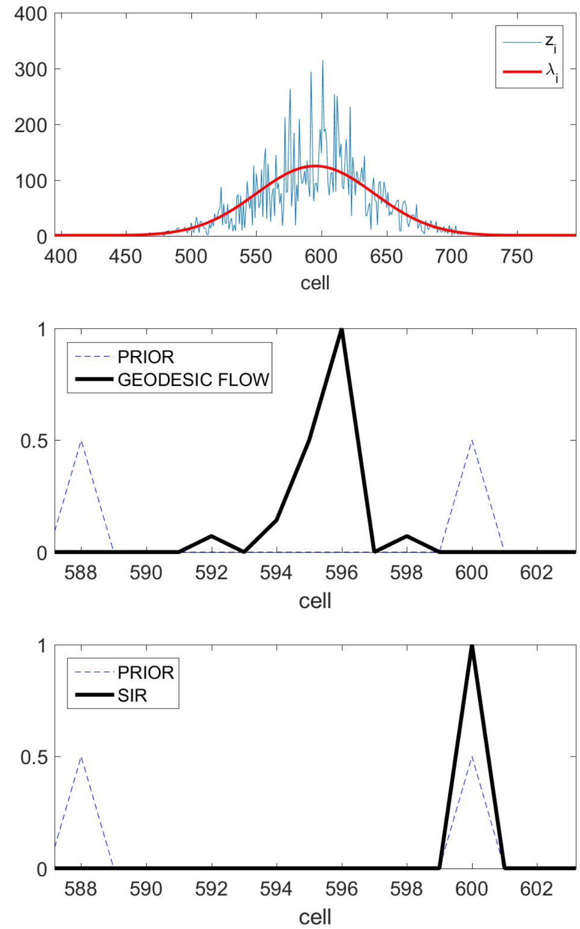

Fig. 5. Zoomed to show detail. Since there are no prior particles near the measurement, the SIR puts full support on the closest particle. The Geodesic method flows particles to the correct region and avoids this pathology.

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. (7), 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 $S = sI_{2\times_2}$, with $s = 32$. The sensor makes measurements on a 300×300 array. We illustrate the performance of both the Gaussian and scaled-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 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. In light of the significant RMSE improvement shown in Figure 6, the geodesic approach still provides a computational

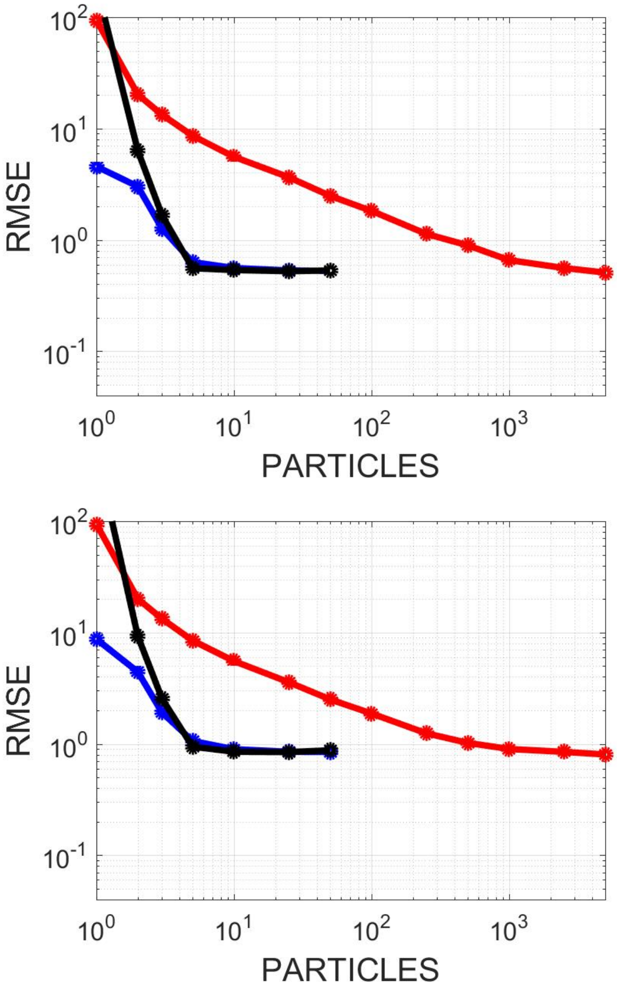

Fig. 6. Monte Carlo Comparisons of the tracking performance versus number of particles for the SIR and Geodesic flow approaches for the $2D$ example. Top: $\lambda_t = 10$. Bottom: $\lambda_t = 4$. Particle diffusion during the flow is done via either the scaled-Identity method [14] or using a Gaussian model [15]. We find that the Geodesic approach needs about $\approx 250x$ fewer particles to reach the RMSE asymptote.

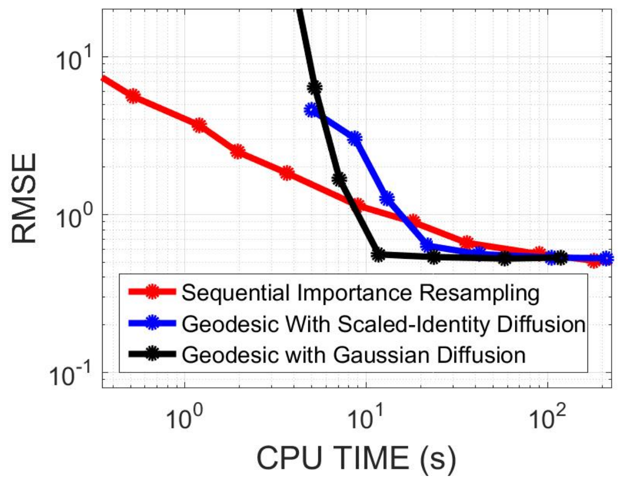

Fig. 7. RMSE performance of the Geodesic flow approach as compared to the SIR approach in terms of CPU time as measured by MatLab. The new Geodesic flow approach provides order-of-magnitude improvement in terms of CPU time for performance.

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 λ, 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 and studied both scaled-identity and Gaussian 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

[1] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter*, Norwood, MA: Artech House, 2004.

[2] L. D. Stone, R. L. Streit, T. L. Corwin, and K. L. Bell, *Bayesian Mutiple Target Tracking, 2nd Ed.*, Norwood, MA: Artech House, 2014.

[3] F. Daum and J. Huang, ”Particle degeneracy: root cause and solution,” *Proc. SPIE 8050, Signal Processing, Sensor Fusion, and Target Recognition XX*, 80500W, May 2011.

[4] F. Daum and J. Huang, ”Nonlinear filters with log-homotopy,” *Proc. SPIE Conf. Signal and Data Processing of Small Targets*, San Diego, CA, Sept. 2007.

[5] F. Daum, J. Huang, and A. Noushin, ”Exact particle flow for nonlinear filters”, *Proc. SPIE 7697, Signal Processing, Sensor Fusion, and Target Recognition XIX*, vol. 7697, April 2010.

[6] F. Daum and J. Huang, ”Particle flow with non-zero diffusion for nonlinear filters”, *Proc. SPIE 8745, Signal Processing, Sensor Fusion, and Target Recognition XXII, 87450* May 2013.

[7] F. Daum and J. Huang, ”Particle flow with non-zero diffusion for nonlinear filters, Bayesian decisions and transport”, *Proc. SPIE Conference on Signal and Data Processing*, San Diego, August 2013.

[8] F. Daum and J. Huang, ”Particle flow for nonlinear filters, Bayesian decisions and transport,” *Proc. IEEE International Conference on Information Fusion*, pp. 1072-1079, 2013.

[9] F. Daum and J. Huang, ”How to avoid normalization of particle flow for nonlinear filters, Bayesian decisions, and transport”, *Proc. SPIE Vol. 9092, 90920B*, 2014.

[10] F. Daum and J. Huang, ”Proof that particle flow corresponds to Bayes rule: necessary and sufficient conditions”, *Proc. SPIE Vol. 9474, 94740I*, 2015.

[11] F. Daum and J. Huang, "A baker's dozen of new particle flows for nonlinear filters, Bayesian decisions and transport", *Proc. SPIE Vol. 9474, 94740J*, 2015.

[12] F. Daum and J. Huang, "A plethora of open problems in particle flow research for nonlinear filters, Bayesian decisions, Bayesian learning, and transport", *Proc. SPIE Vol. 9842, 98420I*, 2016.

[13] S. Mori, F. Daum and J. Douglas, "Adaptive step size approach to homotopy-based particle filtering Bayesian update," *IEEE International Conference on Information Fusion (FUSION)*, Heidelberg, pp. 2035-2042, 2016.

[14] F. Daum, "Seven dubious methods to compute optimal Q for Bayesian stochastic particle flow," *Proc. IEEE International Conference on Information Fusion*, 2016.

[15] F. Daum, J. Huang and A. Noushin, "Gromov's Method for Bayesian Stochastic Particle Flow: A Simple Exact Formula for Q", *Proc. IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, Baden-Baden Germany, Sept. 2016.

[16] S. Choi, P. Willett, F. Daum, and J. Huang, "Discussion and application of the homotopy filter," *Proc. SPIE 8050, Signal Processing, Sensor Fusion, and Target Recognition XX*, 805021, May 2011.

[17] T. Ding and M. J. Coates, "Implementation of the Daum-Huang exact-flow particle filter," *2012 IEEE Statistical Signal Processing Workshop*, Ann Arbor, MI, pp. 257-260, Aug. 2012.

[18] V. P. Jilkov, J. Wu, and H. Chen, "Performance comparison of GPU-accelerated particle flow and particle filters, *Proc. IEEE Fusion Conference*, Istanbul, Turkey, pp. 1095-1102, July 2013.

[19] K. Bell and L. D. Stone, "Implementation of the homotopy particle filter in the JPDA and MAP-PF multi-target tracking algorithms," *Proc. IEEE Conference on Information Fusion*, Salamanca, Spain, July 2014.

[20] A. Horchler, "SDETools: A Matlab Toolbox for the Numerical Solution of Stochastic Differential Equations", Retrieved from https://github.com/horchler/SDETools, 2016.

[21] P. E. Kloeden and E. Platen, *Numerical Solution of Stochastic Differential Equations*. Springer-Verlag, 1992.