18th International Conference on Information Fusion
Washington, DC - July 6-9, 2015

Renormalization group flow in k-space for nonlinear filters, Bayesian decisions and transport

Fred Daum & Jim Huang
Raytheon
Woburn MA USA
daum@raytheon.com

***Abstract*—We derive a new algorithm which avoids normalization of the probability density for particle flow. The algorithm was inspired by renormalization group flow in quantum field theory. In contrast with other particle flow algorithms, this one works in k-space rather than state space. We have roughly 30 or 40 algorithms to compute particle flow, and the three best algorithms avoid computing the normalization of the conditional probability density of the state. We explain why explicit normalization often spoils the flow. This phenomenon has been noticed by other researchers for completely different applications (e.g., weather prediction), but apparently the benefits of avoiding normalization are not well known.**

Keywords- particle filter, nonlinear filter, particle flow, transport problem, extended Kalman filter, Monge-Kantorovich optimal transport

I. INTRODUCTION

We derive a new algorithm which avoids normalization of the conditional probability density for particle flow corresponding to Bayes' rule. This algorithm was inspired by renormalization group flow in quantum field theory, and in contrast with other particle flow algorithms it works in k-space rather than state space. Out of roughly 30 to 40 different algorithms to compute particle flow our 3 favorite methods avoid normalization. It is interesting that these 3 methods avoid normalization in 3 completely different ways. As explained in [24] and [45], explicit computation of normalization often spoils the accuracy of the flow, and hence it is best to avoid such computations altogether. Other researchers have discovered the same phenomenon in diverse applications, such as weather prediction. We will explain both the mathematical and physical significance of normalization. There are extremely interesting and useful analogies with plasmas and classical electromagnetism which are helpful to understand the intuitive ideas involved. It turns out that normalization is equivalent to neutral charge density of our set of particles. Moreover, it is much better to design a set of particles that have charges like normal matter rather than plasmas, which are notoriously unstable. Normal matter made of atoms has exactly neutral charge density both globally and locally (on the scale of one atom). Each atom in normal matter has exactly zero net charge. Furthermore, the charge of electrons and protons is quantized, which greatly facilitates the maintenance of exactly neutral charge density. This connection with electromagnetism is due to the fact that our basic PDE is the same as the Gauss law in divergence form, which is the first of Maxwell's equations. This law guarantees conservation of total probability mass in particle flow. The right hand side of our PDE (6) can be interpreted as charge density of our particles.

Our fundamental equation is a linear PDE with constant coefficients. We know that God intended us to solve linear PDEs with constant coefficients using the Fourier transform. Accordingly, we derived a simple elegant particle flow using the Fourier transform in [50], but unfortunately it does not result in a robust numerical algorithm owing to the use of explicit normalization of the conditional probability density. It is a pity that an elegant mathematical algorithm is spoiled by such numerical issues, and it is the purpose of this paper to fix this problem. Our new algorithm is based on the elegant Fourier transform solution of our PDE, and it avoids normalization.

We start by reviewing particle flow for Bayes' rule in nonlinear filters (section II). The nonlinear filter problem is defined and explored in [6] and [29] from a practical engineering perspective, with an excellent state-of-the-art tutorial on standard particle filters in [21], as well as an encyclopedic treatment of nonlinear filtering in the recent handbook [22], and hence there is no need to repeat such introductory discussions here. We describe three old methods to avoid normalization of the conditional probability density for particle flow in section III, and we explain the physical and mathematical significance of these ideas. The new algorithm for particle flow which avoids normalization is derived in section IV.

There is an intimate connection between the work reported here and so-called optimal transport problems, as explained in [48]. Our particle flow is much simpler and much faster than optimal transport, because we do not solve a variational problem. In particular, we do not pick a unique solution of our

PDE by minimizing a convex functional, but rather we have much simpler methods of picking a unique solution. Moreover, we do not care about the physical effort of transport of our particles, because they are not physical. That is, we do not care how much our particles sweat during the transport. We use non-optimal transport rather than optimal transport, which would be a bad idea in this context. It is much better to minimize computational complexity or maximize stability of the flow rather than minimize an arbitrary and irrelevant convex functional of the transport. We can use any of our particle flow algorithms to solve arbitrary transport problems for smooth nowhere vanishing densities. In particular, we can compute a particle flow between any two smooth nowhere vanishing probability densities by defining the likelihood as the ratio of the posteriori and prior densities. That is, we do not need to have an estimation or decision problem with a given likelihood. Moreover, we can use our particle flow for Bayesian decision problems with arbitrary smooth nowhere vanishing densities. For high dimensional problems, our particle flow is many orders of magnitude faster than standard algorithms for optimal transport and Bayesian decision problems.

II. PARTICLE FLOW FOR BAYES' RULE

The purpose of particle flow is to fix the problem called "particle degeneracy" as explained in [9]. Particle degeneracy is caused by Bayes' rule, and it completely destroys the performance of standard particle filters for high dimensional state vectors. This problem is exacerbated by measurements that are more accurate (e.g., higher signal-to-noise ratio), which is rather counterintuitive, but very easy to understand intuitively by thinking about the cartoon shown in figure 3 of [9]. This problem is fixed by moving the particles to the correct locations in d-dimensional state space. The "correct" locations for particles allows us to accurately represent the probability density of the state vector (x) conditioned on the latest measurement. We do not care which particles move to which locations in state space, as long as we achieve the desired probability density corresponding to Bayes' rule. The flow of particles takes place as the variable λ goes continuously from 0 to 1, and the parameter λ is analogous to time, but it is distinct from time. A "particle" is a point in d-dimensional state space, and we represent the numerical values of the probability densities at each particle in the nonlinear filter; this is the standard approach in essentially all particle filters [21]. We denote the value of each particle in d-dimensional space by the symbol x(λ), and we consider x to be a function of λ. This is a novel viewpoint that is not used in any standard particle filters.

We start by defining a flow of the logarithm of the conditional probability density function with respect to λ:

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

in which p(x, λ) is the conditional probability density of x as a function of λ, h(x) is the likelihood, g(x) is the prior density of x and K(λ) is the normalization of the conditional probability density. The flow of the probability density function defined by (1) corresponds to Bayes' rule. At the start of the flow (λ = 0) apparently p(x, λ) = g(x), and at the end of the flow (λ = 1) we see that p(x, λ) = g(x)h(x)/K(λ), which is exactly what we want to compute. The above flow of the probability density function is called a "homotopy", and it is a well known device in statistics (e.g., see [5] and pages 252-256 in [4]). This homotopy on the logarithm of the conditional density was first used to derive an ODE for Bayes' rule to compute arbitrary statistics in nonlinear filters in [37] and [38]. Several decades later this idea was introduced into particle filtering in [2] and [3], apparently independently of each other. Actually, the formulation in [3] is a much more general flow than the simple line homotopy of the log of the densities shown above. The purpose of this flow of densities was to mitigate particle degeneracy, as explained very clearly in [2] and [3]. As described by Simon Godsill and Tim Clapp: "the task is now to move the particle cloud through this sequence of densities by any available means" (page 144 in [2]); in particular, an algorithm called "annealed importance sampling" was suggested in [2], and similar sampling methods were studied in [3]. However, one would really like to avoid sampling particles in high dimensional state space for arbitrary non-Gaussian densities, owing to the well known difficulties of such algorithms. Therefore, what we really want is to move the particles directly without sampling from any density. This is exactly what particle flow accomplishes. In particular, we want to find a flow of particles corresponding to the flow of probability density defined above by the log-homotopy. We suppose that the flow of particles for Bayes' rule obeys the following Ito stochastic differential equation:

$$dx = f(x,\lambda)d\lambda + dw \quad (2)$$

Moreover, given the flow of the conditional probability density p(x, λ) defined by (1), we can compute the function f(x, λ) by using the Fokker-Planck equation:

$$\frac{\partial p}{\partial \lambda} = -div(fp) + \frac{1}{2}Tr\left[Q\frac{\partial^2 p}{\partial x^2}\right] \quad (3A)$$

$$\frac{\partial p}{\partial \lambda} = -div(fp) + \frac{1}{2}div\left[Q(x)\frac{\partial p}{\partial x}\right] \quad (3B)$$

in which Q is the covariance matrix of the process noise, Tr(.) denotes the trace of (.) and div(.) is the divergence of (.). For simplicity in all of our previous papers we have assumed that the diffusion is zero: Q = 0, but in this paper we develop a new theory of particle flow with non-zero diffusion (see section III for details). Equation (3A) assumes that Q is independent of x, whereas (3B) is more general, and it allows Q(x) to depend on x. As explained in [7], with the assumption of zero diffusion, the flow of particles for Bayes' rule simplifies to an ordinary differential equation:

$$\frac{dx}{d\lambda} = f(x,\lambda) \quad (4)$$

The corresponding Fokker-Planck equation with zero diffusion simplifies to the following:

$$\frac{\partial p}{\partial \lambda} = -div(fp) \quad (5)$$

Combining (1) with (5) and some calculus, we get:

$$div(q) = \eta \quad (6)$$

in which q = pf and we define:

$$\eta = -p(x,\lambda)[\log h(x) - \frac{d \log K(\lambda)}{d\lambda}] \quad (7)$$

In the above we have implicitly assumed that the densities are smooth and nowhere vanishing. Given p(x, λ) and η(x, λ) we want to solve (6) for the function f(x, λ), which will provide the ODE for the desired flow of particles corresponding to Bayes' rule. There are many ways to solve this PDE; in particular, [7], [10], [11] and [58] show roughly 30 to 40 distinct methods to solve (6), but there are also many hybrid methods as well. We will show only three or four such methods in this paper, owing to page limitations. Under mild regularity conditions, one can prove that (6) is a necessary and sufficient condition for the set of particles to obey the probability density corresponding to Baye's rule [57].

The reason that our PDE (6) has so many distinct methods of solution is that it is an extremely simple and nice PDE. In particular, our PDE has the following characteristics: (1) it is a linear PDE in the unknown functions f or q; (2) it is a first order PDE; (3) it has constant coefficients in the unknown function q; (4) there are no boundary conditions or initial conditions on the unknown function f; (5) the PDE is highly underdetermined, because it is only a single scalar-valued PDE, whereas the unknown function is d-dimensional; (6) our PDE has the same form as the Gauss divergence law in electromagnetics; and (7) a necessary and sufficient condition for the existence of a solution to this PDE is that the integral of η with respect to x is zero (see [11] for a discussion of the physical and mathematical meaning of this crucial condition). We can exploit these nice characteristics of our PDE to reduce computational complexity.

The first nice solution of our PDE assumes that the prior (g) and the likelihood (h) are unnormalized Gaussian probability densities. Substituting these densities into (6) and equating like coefficients of x and terms that are quadratic in x results in the following exact solution for the particle flow corresponding to Bayes' rule:

$$\boxed{\frac{dx}{d\lambda} = A(\lambda)x + b(\lambda)} \quad (8)$$

in which

$$A = -\frac{1}{2}PH^T(\lambda HPH^T + R)^{-1}H \quad (9)$$

$$b = (I + 2\lambda A)[(I + \lambda A)PH^T R^{-1} z + A\bar{x}] \quad (10)$$

where P is the covariance matrix of prediction error in x for the Gaussian prior density, and H is the measurement matrix (i.e., z = Hx+v), and R is the covariance matrix of the measurement noise (v). The symbol $\bar{x}$ denotes the predicted value of x, corresponding to the mean value of the prior Gaussian density. Inspection of (9) shows that the particle flow is stable (under very mild conditions). For nonlinear problems one can use this flow by linearizing the measurement equation about each particle (analogous to an EKF) and computing the covariance matrix P from the sample of particles. It turns out that the performance of this filter is orders of magnitude better than the EKF for difficult nonlinear problems, despite the linearization, because we are linearizing about each particle rather than using only one linearization about the estimate of x (e.g., see [8] for many numerical experiments). Moreover, the computation of this filter is roughly ten orders of magnitude faster than standard particle filters for high dimensional problems (see [7] and [8] for details).

The second nice solution of our PDE does not assume that the densities are Gaussian, but rather it only assumes that g and h are smooth and nowhere vanishing. However, we assume that the flow of particles is incompressible, analogous to the well known approximation in fluid dynamics (e.g., subsonic flight in air is well approximated by incompressible flow). That is, we assume that the divergence of f is zero. As shown in [28], this results in the following ODE for the flow of each particle:

$$\boxed{\frac{dx}{d\lambda} = -\frac{\left(\frac{\partial \log p}{\partial x}\right)^T}{\left\|\frac{\partial \log p}{\partial x}\right\|^2} \text{logh}} \quad (11)$$

for non-zero gradient, and the flow is zero otherwise. The gradient of logp can be approximated using a very fast finite difference algorithm in which we compute the set of approximate k-nearest neighbors for each particle (see [27] for details).

The third nice solution of our PDE assumes that the log of the conditional density is twice continuously differentiable, and that there exists a diffusion matrix Q(x) and a flow f(x, λ) such that certain undesirable terms in our PDE sum to zero. Assuming that the Hessian matrix of log p is non-singular, as shown in [45] we obtain the following flow:

$$\boxed{f = -\left[\frac{\partial^2 \log p}{\partial x^2}\right]^{-1}\left(\frac{\partial \log h}{\partial x}\right)^T} \quad (12)$$

This flow is very easy to compute in real time. In particular, we can compute the gradient of the log-likelihood in closed form using calculus, because the function h(x) is typically known as a formula in practical applications. Moreover, the Hessian of log p follows from the definition of log p given by (1) using calculus:

$$\frac{\partial^2 \log p}{\partial x^2} = \frac{\partial^2 \log g}{\partial x^2} + \lambda \frac{\partial^2 \log h}{\partial x^2} \tag{13}$$

We can compute the Hessian of logh in closed form, for the same reason that we can compute the gradient of logh in closed form; moreover, we can compute the Hessian of log g using our fast k-nearest neighbor algorithm, as explained in [27]. Many more details about this flow are given in [45] and [52].

As shown in [44] we can derive a very similar flow using completely different assumptions (i.e., zero diffusion but geodesic flow on a convex subset of Euclidean state space):

$$\frac{dx}{d\lambda} = 2k\left[\frac{\partial^2 \log p}{\partial x^2}\right]^{-1}\left(\frac{\partial \log h}{\partial x}\right)^T \tag{14}$$

An explicit formula for k is given in [44], but in practice one can use the simple approximation $k \approx -1/2$ and obtain excellent results. This flow was derived by solving a vector Riccati equation in closed form, by making a judicious guess at the solution as explained in [44]. However, the solution to the vector Riccati equation is highly underdetermined, because we have a single scalar valued equation in d unknowns. In particular, the solution set is a hyperellipsoid in d dimensions. As suggested in [44] we could pick the vector b to go through the center (or one of the foci) of this hyperellipsoid, but there are many other possible choices, some of which might result in better flows. For example, we could pick f to give the most stable flow in the sense of Lyapunov, similar to the flow derived in [34]. Second, we could pick f to give the least stiff flow, which means the smallest condition number of the Jacobian of the flow, as discussed in [59]. Third, we could pick f to give the minimum L^2 norm solution. Further details will be given in [60]. Moreover, there is a very interesting connection between our geodesic particle flow (14) and the Zel'dovich approximation, which was invented to solve the N-body problem for the big bang in cosmology (see [61] to [64] for details). In particular, the Zel'dovich approximation (ZA) is a simple first order Taylor series for the flow of particles in the big bang, but it gives surprisingly accurate results when compared with numerical solutions of Einstein's field equations using Monte Carlo simulations for the N-body problem, as explained in [64]. However, Brenier has recently shown that the ZA is actually an exact solution to the Monge-Ampère equation for a certain non-standard gravitational model [61]. The Monge-Ampère equation is a highly nonlinear PDE that characterizes the transport of particles from a given probability density to another given probability density, as discussed in [58]. Unfortunately, the Monge-Ampère equation does not obey the N-principle, as explained in [58], but we can derive a new PDE analogous to the Monge-Ampère equation that does satisfy the N-principle (see [58] for details). The connection between the ZA and our particle flow (14) is that both are geodesic flows in a convex subset of Euclidean space, and both flows are extremely accurate and fast to compute. Equation (1) in [62] shows that the ZA is a geodesic flow in a convex subset of Euclidean space. Moreover, ZA is a Lagrangian perturbation which guarantees that the probability density is nowhere vanishing, unlike the usual Eulerian perturbations. Also, ZA provides a simple formula for the probability density along the flow, somewhat similar to Moser's coupling, which also suffers from singularities in finite time [62]. Our geodesic flow (14) has no singularities, unlike ZA, because we use a log-homotopy whereas ZA and Moser do not. As noted by Brenier the physical relevance of the Monge-Ampère gravitational model is open to debate [61], but we don't care about this because our particles are not physical. We only care about having a fast and robust algorithm to move particles from one density to another in d-dimensional space. The ZA was invented in 1970, and it is another example of physics being many decades ahead of statistics, which is the basic theme of this paper.

III. THREE OLD METHODS TO AVOID NORMALIZATION

Notice that our PDE for particle flow (6) explicitly depends on the normalization $K(\lambda)$ of the conditional probability density of x along the flow, but the first three methods to compute particle flow reviewed in this paper do not need to compute $K(\lambda)$. In particular, the Gaussian flow in (8) does not depend explicitly on $K(\lambda)$, and likewise for the incompressible flow (11) or the flow with non-zero diffusion (12), but our geodesic flow (14) is different and very interesting as explained later. This contrasts with many of our other flows which require explicitly computing $K(\lambda)$. For example, our flow based on Coulomb's law computes $K(\lambda)$ using a sample mean of log $h(x, \lambda)$, as explained in [11] and [24]. Similarly, the flow based on the Fourier transform explicitly computes $K(\lambda)$ as explained in [50]. It is not difficult to compute $K(\lambda)$ using the Monte Carlo method given in [11]. In particular, it is easy to show that:

$$\frac{d \log K(\lambda)}{d\lambda} = E[\log h(x)] \tag{15}$$

In which E(.) denotes the expected value of (.) with respect to the probability density $p(x, \lambda)$. We can derive (15) using calculus, as shown in [11], but it is much easier to derive (15) using the Fourier transform. In particular, take the Fourier transform of (6) and evaluate it at $\omega = 0$ as follows:

$$div(pf) = \eta \tag{16}$$

$$i\omega^T F(pf) = \int_\Omega \eta \exp[-i\omega^T x] \mathrm{dx} \tag{17}$$

$$\int_\Omega \eta(x) dx = 0 \tag{18}$$

From which (15) follows immediately.

We can easily compute the expected value of log h(x) using the standard Monte Carlo approximation, as explained in [11] and [24]:

$$\frac{d \log K(\lambda)}{d\lambda} \approx \frac{1}{N}\sum_{j=1}^{N} \log h(x_j) \tag{19}$$

In summary, it is easy to compute an approximation to the normalization term in (6), but the crucial question is: how accurate does this approximation need to be in order to get good filter accuracy? We have learned from our numerical experiments in [24] and [50] and elsewhere that we need to compute this normalization term to very high accuracy. This was rather surprising, and we explain this in the following paragraphs.

First, the condition (18) is a necessary and sufficient condition for the existence of a solution to the PDE (6). For more details on this mathematical fact, see [11] and [24]. If condition (18) is not satisfied, then there is no solution to our PDE. If a solution does not exist for our PDE (6), then there is no reason to expect that the numbers coming out of our computer correspond to what we want in the first place. There are many notorious examples of this in science and math. In the context of particle flow for Bayes' rule, it turns out that the flow would be completely useless, and the same situation obtains for weather prediction, as explained below. Second, if we consider our set of particles as a plasma with electric charge density η, then condition (18) corresponds to neutral charge density over the volume Ω. This physical analogy makes sense because (6) is the Gauss divergence law in electromagnetics for arbitrary dimensions rather than just $d = 3$. It turns out that a very old theorem in classical physics (i.e., Earnshaw's theorem from 1842) says that a dynamical system of particles must have neutral charge density in order to be stable. Plasmas are notoriously unstable, and we want our particle flow to be stable in order to avoid explosive propagation of numerical errors and other errors due to our various approximations. Normal matter is electrically neutral to an extremely high accuracy, otherwise it would explode owing to the Coulomb's force. This is one of Feynman's favorite examples to show how strong electromagnetism is compared with gravity. Atoms are exactly electrically neutral, otherwise they would explode!

Third, other researchers have noticed this phenomenon in completely different contexts than particle filtering. For example, the state-of-the-art weather prediction solves two Poisson's equations in order to compute good locations for nodes for the solution of Navier-Stokes PDEs. This results in a dramatic improvement in accuracy of weather prediction in the UK. However, these researchers have learned by numerical experiments that the condition for zero mass density (18) in Poisson's equation must be computed to machine precision!!! We learned about this related experience at a wonderful talk at Vancouver given by Dr. Emily Walsh [53].

We now summarize the distinct methods that we have used to avoid normalization in our favorite particle flows. Recall that $K(\lambda)$ does not appear explicitly in the first three flows described in sections II and III of this paper. Moreover, the reason that $K(\lambda)$ does not appear in these equations is different for each flow. First, in the Gaussian flow (8), we can derive an ODE for $K(\lambda)$, but it is completely decoupled from the calculation of the flow itself. This is the same situation with the Kalman filter, where we can compute the equations for the evolution of the mean of x and the covariance matrix of errors in x, but the normalization of the conditional density is not needed. In fact, we could easily derive the Kalman filter using the unnormalized conditional density of x, and we can also derive the Beneš filter as well as many other exact finite dimensional filters using the unnormalized density; for details, see [39]. Moreover, the Zakai equation gives the evolution of the conditional density for a very general nonlinear filter problem using the unnormalized density; this results in a linear PDE rather than a nonlinear PDE, and it makes the analysis of existence and uniqueness of solutions for this PDE much easier, and it also makes the numerical solution of the PDE easier. Normalization of the conditional density for the Kalman filter is essentially never discussed in books or papers on Kalman filters.

Inspection of (11) shows that our incompressible flow does not depend on the normalization $K(\lambda)$. This is because the density of the flow is constant for each particle along the flow, and hence the total probability mass does not change along the flow, and therefore there is exactly zero change in $K(\lambda)$ along the flow, and hence the result of (15) is exactly zero. We do not need to compute zero. It was computed very accurately many centuries ago, and we have been using it ever since. This is completely different from the reason for avoiding normalization for the Gaussian flow, in which the change in normalization is not zero, but it is irrelevant to the flow itself.

Inspection of (12) shows that the flow with non-zero diffusion does not depend on the normalization of the conditional density of x. The reason for this is that we killed the normalization by taking the gradient of the log of p in our derivation. But the story about our geodesic flow (14) is different and interesting. Inspection of (14) shows no evidence of $K(\lambda)$, but it is implicitly there, because the formula for k depends on it (see equations (16) and (17) in [44]). On the other hand, our numerical experiments show that we can use a simple approximation for $k \approx -1/2$ which does not depend on $K(\lambda)$. So the geodesic flow (14) is not exactly invariant to $K(\lambda)$, but its dependence is very weak. This is because $K(\lambda)$ enters the geodesic flow only through the speed of the particles rather than the direction, and we know from numerical experiments that the filter accuracy is extremely insensitive to perturbations in the speed. For example, we can typically change the speed by a factor of two with very little change in filter accuracy. We will see that the same thing happens in our new flow using renormalization group flow in section IV. It would be nice to have a theorem or formula or bound that quantifies this situation, and we are working on this also [65]. In particular, taking the Fourier transform of (6) results in a linear algebraic equation for f (as shown in section IV), and we can bound the errors in the flow (f) due to errors in the normalization using the new theory of effective condition number [66], rather than the boring old theory of condition number due to Turing and von Neumann, which results in beautiful formulas that are completely useless. For this new bound, we also need to generalize the interesting results in [67] to use effective condition number rather than boring old condition number.

It is rather curious that this avoidance of normalization is not well known, especially because the solutions of Poisson's equation which require exquisitely accurate normalization are completely ruined by small numerical errors and errors due to other approximations. Recall that for weather prediction the normalization had to be computed to machine precision!

IV. RENORMALIZATION GROUP FLOW IN K-SPACE

We can derive a new particle flow which avoids normalization of the conditional density by combining three ideas: (1) renormalization group flow, inspired by quantum field theory [56]; (2) work in k-space rather than state space by taking the Fourier transform of our PDE, as explained in [50]; and (3) compute the most general solution of our PDE rather than the unique minimum L^2 norm solution. We start by taking the Fourier transform of our PDE (6):

$$div(pf) = \eta \tag{20}$$

$$i\omega^T F(pf) = \hat{\eta} \tag{21}$$

in which F(pf) denotes the Fourier transform of pf; $\hat{\eta}$ denotes the Fourier transform of η, and i = $\sqrt{-1}$. In deriving (21) from (20) we have implicitly assumed that pf decays towards zero sufficiently fast as x approaches the boundary of the set of integration; otherwise, we would have extra terms from the non-zero boundary conditions; this assumption is satisfied for reasonable f(x, λ) (e.g., no worse than polynomial growth in x) because p(x, λ) is a probability density which we assume decays sufficiently rapidly to zero at the boundary. Using the definition of Fourier transform we obtain:

$$i\omega^T \int p(x,\lambda) f(x,\lambda) \exp(-i\omega^T x) dx = \hat{\eta}(\omega) \tag{22}$$

$$\hat{\eta} = \int p(x,\lambda)\left[-\log h(x) + \frac{d\log K}{d\lambda}\right]\exp(-i\omega^T x) dx$$

Both of these integrals can be interpreted as expected values with respect to the probability density p(x, λ), and hence we can compute these integrals using the standard Monte Carlo approximation by averaging over particles:

$$i\omega^T \frac{1}{N}\sum_{j=1}^{N} f(x_j,\lambda)\exp(-\omega^T x_j) \approx \hat{\eta}(\omega) \tag{23}$$

$$\hat{\eta}(\omega) \approx \frac{1}{N}\sum_{j=1}^{N}\left[-\log h(x_j) + \frac{d\log K}{d\lambda}\right]\exp(-i\omega^T x_j)$$

in which N is the number of particles, and x_j is the j^{th} particle. We could approximate the derivative of logK(λ) with respect to λ as shown in equation (19), but we will not do this here for obvious reasons.

Notice that (23) is a single scalar-valued equation for the d-dimensional vector f for each particle; hence, there is no unique solution for f, but we can evaluate (23) at k values of the vector-valued frequency ω, and compute the minimum L^2 norm solution f by using the generalized inverse. First, we can write (23) for the set of values of ω as a very large linear algebraic operator L acting on f:

$$L\tilde{f} \approx \hat{\eta} \tag{24}$$

This large system of linear equations is written explicitly in Figure1 of [50], in which we have used the constraint that f is a real-valued function. Next, we solve (24) for the unique minimum L^2 norm solution using the generalized inverse of L:

$$\tilde{f} \approx L^T (LL^T)^{-1}\hat{\eta} \tag{25}$$

The symbol $\tilde{f}$ used in (25) is defined as the concatenation of d-dimensional f's at each of N particles, and thus the dimension of $\tilde{f}$ is dN, and the dimensions of L are k by dN, and the matrix to be inverted in (25) is a k by k matrix. The matrix L is displayed in its full glory in figure 1 of [50].

We can choose to evaluate (23) at as many vector-valued frequencies as we wish, but to limit the computational complexity to d^3 we use k ≈ cd, in which c is a constant. Furthermore, we can pick any values for the k vector-valued frequencies, and we use the eigenvalues of the sample covariance matrix to compute the appropriate scale for these points in k-space. Theoretical details and numerical experiments and several variance reduction methods are reported in [50].

Next, we can compute the most general solution to (6) as explained in [34], using the generalized inverse and the projection into the d-1 dimensional subspace orthogonal to the minimum L^2 norm flow:

$$\tilde{f} = L^{\#}\hat{\eta} + (I - L^{\#}L)y \tag{26}$$

in which y is an arbitrary d-dimensional vector, and $L^{\#}$ denotes the generalized inverse of L. Our earlier flow (25) corresponds to the minimum L^2 norm solution of (24) with the choice of y = 0. But we can pick y to provide a more robust flow. In particular, we can pick y to give a particle flow that avoids normalization of the conditional density, or minimizes the error in the flow due to errors in the normalization. It is intuitively obvious that the y that does this is given by:

$$y = -L^{\#}\hat{p}\frac{d\log K(\lambda)}{d\lambda} \tag{27}$$

and hence the renormalized group flow is:

$$\boxed{\tilde{f} = L^{\#}\hat{\eta} - (I - L^{\#}L)L^{\#}\hat{p}\frac{d\log K(\lambda)}{d\lambda}} \tag{28}$$

in which $\hat{p}$ denotes the Fourier transform of the conditional probability density p(x,λ). This choice of y subtracts the normalization from the flow in the d-1 dimensional subspace defined by the projection operator orthogonal to $L^{\#}$. One can derive (27) by asking for the y that minimizes the L^2 norm of the derivative of the flow with respect to the normalization:

$$\frac{\partial\left\|\frac{\partial\tilde{f}}{\partial\beta}\right\|^2}{\partial\gamma} = 0 \tag{29}$$

in which

$$\beta = \frac{d \log K(\lambda)}{d\lambda} \tag{30}$$

$$\gamma = \frac{\partial y}{\partial \beta} \tag{31}$$

The idea behind (29) is that we want to pick y so that errors in the normalization K(λ) result in the least perturbation to the flow (f). This is the same basic idea as used by Gell-Mann and Low in their seminal paper on renormalization group flow for quantum field theory [56], in which they wanted to pick a physical law that was invariant with respect to the energy scale. It is not generally possible to make the flow exactly independent of the normalization, in which case the generalized inverse picks the flow with the minimum L^2 error in the approximation to the desired independence. This is because y acts through the projection into the d-1 dimensional subspace orthogonal to $L^{\#}$, and hence we cannot subtract the effect of the normalization in the remaining one dimensional subspace. But this is OK, because the filter accuracy is extremely insensitive to perturbations in the particle speed (as opposed to the direction of particle flow), as emphasized earlier. This is similar to the situation for the algorithm derived in [56]. We use the variables β and γ rather than K(λ) and y for several reasons: (1) we obtain linear equations using β and γ but not K and y; (2) we avoid a singularity in our ODE using β and γ rather than K and y; (3) the resulting ODE does not depend explicitly on K in d-1 dimensions of the flow (but not all d dimensions) because it cancels exactly. It turns out that this is the same choice in renormalization group flow for quantum field theory (see [56] for details); that is, the physicists differentiate with respect to log μ rather than μ, in which μ denotes the scale of energy or mass in quantum field theory. Numerical results for this new flow will be reported elsewhere, owing to page limitations.

ACKNOWLEDGMENTS

We thank Professor Alan Willsky for asking about the normalization of the conditional probability density for particle flow. Alan's question inspired this paper. We thank Dr. Martin Ulehla for explaining the importance of neutral charge density in physics. We thank Dr. Bhashyam Balaji for noting the similarity of renormalization group flow in quantum field theory and our particle flow, as well as interesting and stimulating discussions of quantum field theory and nonlinear filters. We thank Dr. Dan Zwillinger for the outstandingly good advice on how to solve our PDE: "think like a physicist," which is similar to Professor Jun Liu's dictum for MCMC methods: "the physicists are always 20 years ahead of us," at the Harvard Statistics Conference (2009). In this case, however, physics was about 60 years ahead of statistics, if we date the origin of renormalization group flow to the seminal paper by Gell-Mann and Low (1954). We thank Professor Sidney Coleman (posthumously) for his brilliantly clear and honest and hilarious lectures on quantum field theory at Harvard, which are still available on-line today.

REFERENCES

[1] Uwe Hanebeck, Kai Briechle & Andreas Rauh, "Progressive Bayes: a new framework for nonlinear state estimation," Proceedings of SPIE Conference, Orlando Florida, April 2003.

[2] Simon Godsill & Tim Clapp, "Improvement strategies for Monte Carlo particle filters," pages 139-158 in "Sequential Monte Carlo methods in practice," edited by Arnaud Doucet, Nando de Freitas & Neil Gordon, Springer-Verlag, 2001.

[3] Christian Musso, Nadia Oudjane & Francois Le Gland, "Improving regularised particle filters," pages 247-271 in "Sequential Monte Carlo methods in practice," edited by Arnaud Doucet, Nando de Freitas & Neil Gordon, Springer-Verlag, 2001.

[4] Michael Evans & Tim Swartz, "Approximating integrals via Monte Carlo and determinisitc methods," Oxford University Press, 2000.

[5] Andrew Gelman & Xian-Li Meng, "Simulating normalizing constants: from importance sampling to bridge sampling to path sampling," Statistical Science, volume 13 number 2, pages 163-185, May 1998.

[6] Fred Daum, "Nonlinear filters: beyond the Kalman filter," IEEE AES Systems magazine, special tutorial issue, pages 57 to 69, August 2005.

[7] Fred Daum, Jim Huang and Arjang Noushin, "Exact particle flow for nonlinear filters," proceedings of SPIE Conference, Orlando Florida, April 2010.

[8] Fred Daum & Jim Huang, "Numerical experiments for nonlinear filters with exact particle flow induced by log-homotopy," proceedings of SPIE conference, Orlando Florida, April 2010.

[9] Fred Daum & Jim Huang, "Particle degeneracy: root cause and solution," Proceedings of SPIE Conference, Orlando Florida, April 2011.

[10] Fred Daum & Jim Huang, "Exact particle flow for nonlinear filters: seventeen dubious solutions to a first order linear underdetermined PDE," Proceedings of IEEE Asilomar conference, November 2010.

[11] Fred Daum, Jim Huang & Arjang Noushin, "Coulomb's law particle flow for nonlinear filters," Proceedings of SPIE Conference, San Diego CA, August 2011.

[12] Cedric Villani, "Topics in optimal transportation," American Math Society Press, 2003.

[13] Cedric Villani, "Optimal transport," Springer-Verlag, 2009.

[14] Eldad Haber, Tauseef Rehman & Allen Tannenbaum, "An efficient numerical method for the solution of the L^2 optimal mass transfer problem," SIAM Journal of scientific computing, pages 197-211, 2010.

[15] Gian Luca Delzanno and John Finn, "The fluid dynamic approach to equidistribution methods for grid generation and adaptation," Computer Physics Communications, Vol.182(2), pages 330-346, 2011.

[16] Rick Chartrand, Brendt Wohlberg, Kevin Vixie and Erik Bollt, "A gradient descent solution to the Monge-Kantorovich problem," Applied Mathematical Sciences, volume 3, number 22, pages 1071-1080, 2009.

[17] Vladimir Arnold & Boris Khesin, "Topological methods in hydrodynamics, Springer-Verlag, 1998.

[18] Lingji Chen & Raman Mehra, "A study of nonlinear filters with particle flow induced by log-homotopy," Proceedings of SPIE conference, Orlando Florida, April 2011.

[19] Mark Girolami & Ben Calderhead, "Riemann manifold Langevin and Hamiltonian Monte Carlo methods," (with discussion & rejoinder) Journal Royal Statistical Society series B, pages 123-214, volume 73, 2011.

[20] Havard Rue, Sara Martino & Nicolas Chopin, "Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations," (with discussion & rejoinder) Journal Royal Statistical Society series B, pages 319-392, volume 71, 2009.

[21] Arnaud Doucet & A. M. Johansen, "A tutorial on particle filtering and smoothing: fifteen years later," Chapter 24 in "The Oxford handbook of nonlinear filtering," edited by Dan Crisan & Boris Rozovskiĭ, Oxford University Press, 2011.

[22] "The Oxford handbook of nonlinear filtering," edited by Dan Crisan & Boris Rozovskiĭ, Oxford University Press, 2011.

[23] Jonas Hagmar, Mats Jirstrand, Lennart Svensson & Mark Morelande, "Optimal parameterization of posterior densities using homotopy," Proceedings of IEEE fusion conference, Chicago Illinois, July 2011.

[24] Fred Daum, Jim Huang & Arjang Noushin, "Numerical experiments for Coulomb's law particle flow for nonlinear filters," Proceedings of SPIE Conference, San Diego CA, August 2011.

[25] Fred Daum & Jim Huang, "Hollywood log-homotopy: movies of particle flow for nonlinear filters," Proceedings of SPIE conference, Orlando Florida, April 2011.

[26] Fred Daum & Jim Huang, "A fresh perspective on research for nonlinear filters," Proceedings of SPIE conference, Orlando Florida, April 2010.

[27] Fred Daum, Jim Huang, Misha Krichman & Talia Kohen, "Seventeen dubious methods to approximate the gradient for nonlinear filters with particle flow," Proceedings of SPIE conference, San Diego CA, August 2009.

[28] Fred Daum & Jim Huang, "Nonlinear filters with log-homotopy," Proceedings of SPIE Conference on signal & data processing, San Diego California, September 2007.

[29] Branko Ristic, Sanjeev Arulampalam & Neil Gordon, "Beyond the Kalman filter," Artech House publishing, 2004.

[30] Alexander Shnirelman, "Evolution of singularities, generalized Liapunov function and generalized integral for an ideal incompressible fluid," American Journal Math., volume 119, number 3, pages 579-608, 1997.

[31] Chris Budd, Weizhang Huang and Robert Russell, "Adaptivity with moving grids," Acta Numerica, pages 1 to 131, Cambridge University Press, 2009.

[32] Robert Marlow, "Moving mesh methods for solving parabolic partial differential equations," doctoral thesis, University of Leeds, September 2010.

[33] Mohamed Sulman, J. F. Williams and Robert Russell, "Optimal mass transport for higher dimensional adaptive grid generation," Journal of computational physics, pages 3302-3330, 2011.

[34] Fred Daum & Jim Huang, "Generalized particle flow for nonlinear filters," Proceedings of SPIE Conference, Orlando Florida, April 2010.

[35] Jacques-Louis Lions, "Optimal control of systems governed by partial differential equations," translated by Sanjoy Mitter, Springer-Verlag, 1971.

[36] Peter Mathé and Erich Novak, "Simple Monte Carlo and the Metropolis algorithm" Journal of Complexity, volume 23, issues 4-6, pages 673-696, August-December 2007.

[37] Fred Daum, "A new nonlinear filtering formula for discrete time measurements," IEEE CDC Proceedings, Fort Lauderdale Florida, December 1985.

[38] Fred Daum, "A new nonlinear filtering formula for non-Gaussian discrete time measurements," IEEE CDC Proceedings, Athens Greece, December 1986.

[39] Fred Daum, "Nonlinear filters: beyond the Kalman filter," IEEE AES systems magazine, special tutorial issue, pages 57-69, August 2005.

[40] Fred Daum, "Exact finite dimensional filters for cryptodeterministic systems," Proceedings of IEEE Control and Decision Conference, pages 1638-1639, December 1986.

[41] Omar Hijab, "A class of infinite dimensional filters," Proceedings of IEEE Control and Decision Conference, pages 62-65, December 1980.

[42] Fred Daum and Jim Huang, "Friendly rebuttal to Chen and Mehra: incompressible particle flow for nonlinear filters," Proceedings of SPIE Conference, Baltimore, April 2012.

[43] "Monte Carlo and quasi-Monte Carlo methods," edited by Leszek Plaskota and Henryk Wozniakowski, Springer-Verlag, 2012.

[44] Fred Daum and Jim Huang, "Zero curvature particle flow for nonlinear filters," Proceedings of SPIE Conference, Baltimore, April 2013.

[45] Fred Daum and Jim Huang, "Particle flow with non-zero diffusion for nonlinear filters," Proceedings of SPIE Conference, Baltimore, April 2013.

[46] Daniel Rudolph, "Hit-and-run for numerical integration," MCQMC Workshop, Sydney Australia, February 2012.

[47] Daniel Rudolph, "Explicit error bounds for lazy reversible Markov chain Monte Carlo," January 2011.

[48] Fred Daum & Jim Huang, "Particle flow and Monge-Kantorovich transport," Proceedings of FUSION Conference, Singapore, July 2012.

[49] Fred Daum & Jim Huang, "Small curvature particle flow for nonlinear filters," Proceedings of SPIE Conference, Baltimore, May 2012.

[50] Fred Daum and Jim Huang, "Fourier transform particle flow for nonlinear filters," Proceedings of SPIE Conference, Baltimore, April 2013.

[51] Fred Daum and Jim Huang, "Particle flow inspired by Knothe-Rosenblatt transport for nonlinear filters," Proceedings of SPIE Conference, Baltimore, April 2013.

[52] Fred Daum and Jim Huang, "Particle flow with non-zero diffusion for nonlinear filters, Bayesian decisions and transport," Proceedings of SPIE Conference on Signal & Data Processing, San Diego, August 2013.

[53] Emily Walsh and Chris Budd, "Moving mesh methods for problems in meteorology," ICIAM Conference, Vancouver, July 2011.

[54] Bernard Dacorogna, "existence and regularity of solutions of $d\omega = f$ with Dirichlet boundary conditions," 2005.

[55] Thierry de Pauw and Washek Pfeffer, "Distributions for which div $v = F$ has a continuous solution," 2006.

[56] Fred Daum and Jim Huang, "renormalization group flow and other ideas inspired by physics for nonlinear filters, Bayesian decisions and transport," Proceedings of SPIE Conference, Baltimore, May 2014.

[57] Fred Daum and Jim Huang, "proof that particle flow corresponds to Bayes' rule: necessary and sufficient conditions," Proceedings of SPIE Conference, Baltimore, April 2015.

[58] Fred Daum and Jim Huang, "a baker's dozen of new particle flows for nonlinear filters, Bayesian decisions and transport," Proceedings of SPIE Conference, Baltimore, April 2015.

[59] Fred Daum and Jim Huang, "Seven dubious methods to mitigate stiffness in particle flow with non-zero diffusion for nonlinear filters, Bayesian decisions and transport," Proceedings of SPIE Conference on signal processing, Baltimore, April 2014.

[60] Fred Daum, Jim Huang and A. J. Noushin, "Five dubious solutions to the vector Riccati equation for particle flow," work in progress, 2015.

[61] Yann Brenier, "Reconstruction of the early universe, Zeldovich approximation and Monge-Ampère gravitation," available online, 2010.

[62] Johan Hidding, et al., "the Zeldovich approximation: key to understanding cosmic web complexity," Monthly notices of the Royal Astonomical Society, November 2013.

[63] Martin White, "the Zel'dovich approximation," Monthly notices of the Royal Astronomical Society, December 2014.

[64] Ayako Yoshisato, et al., "why is the Zel'dovich approximation so accurate?" the Astrophysical Journal, February 2006.

[65] Fred Daum, Jim Huang and A. J. Noushin, "the N-principle," work in progress, 2015.

[66] Zi-Cai Li, Hung-Tsai Huang, Jeng-Tzong Chen and Yimin Wei, "effective condition number and its applications," Proceedings of 7th World Congress on computational mechanics, Los Angeles California, July 2006.

[67] Siddharth Joshi and Stephen Boyd, "subspaces that minimize the condition number of a matrix," Rejecta Mathematica, volume 1 number 1, pages 4-9, July 2009.