Nonlinear Filters: Beyond the Kalman Filter

FRED DAUM, Member, IEEE
Raytheon

Nonlinear filters can provide estimation accuracy that is vastly superior to extended Kalman filters for some important practical applications. We compare several types of nonlinear filters, including: particle filters (PFs), unscented Kalman filters, extended Kalman filters, batch filters and exact recursive filters. The key practical issue in nonlinear filtering is computational complexity, which is often called "the curse of dimensionality." It has been asserted that PFs avoid the curse of dimensionality, but this is generally incorrect. Well-designed PFs with good proposal densities sometimes avoid the curse of dimensionality, but not otherwise. Future research in nonlinear filtering will exploit recent progress in quasi-Monte Carlo algorithms (rather than boring old Monte Carlo methods), as well as ideas borrowed from physics (e.g., dimensional interpolation) and new mesh-free adjoint methods for solving PDEs. This tutorial was written for normal engineers, who do not have nonlinear filters for breakfast.

Manuscript received January 17, 2004; revised September 17, 2004; released for publication October 4, 2004.

Refereeing of this contribution was handled by P. K. Willett.

Author's address: 318 Acton St., Carlisle, MA 01741-1442, E-mail: (Frederick_E_Daum@raytheon.com).

I. INTRODUCTION

Kalman filters are extremely useful in diverse real world applications, including: robotics, communication systems, GPS, inertial navigation, chemical plant control, predicting the weather, multi-sensor data fusion, tracking of aircraft, satellites, ships, rockets, cars, people, cows, and salmon, as well as some dubious applications (e.g., trying to predict the stock market). Moreover, Kalman filters are relatively easy to design and code, and they often provide good estimation accuracy. On the other hand, Kalman filter accuracy can be surprisingly bad for some practical applications, for several reasons, including: 1) nonlinearities in the equations that describe the physical system, 2) ill-conditioning of the covariance matrix, and 3) inaccurate or incomplete models of the underlying physical problem. For example, Figs. 1 and 2 compare the performance of Kalman filters with various nonlinear filters. In both examples, the nonlinear filters provide vastly superior performance compared with the Kalman filter. This should whet your appetite for nonlinear filters.

The Kalman filter was invented over four decades ago, and since then there has been intense research on nonlinear filters attempting to improve estimation accuracy. Table I summarizes the main types of nonlinear filters that are used in practice. Each of these nonlinear filters is discussed in the following survey. The key practical issue with nonlinear filters is real time computational complexity to achieve a given estimation accuracy. The speed and memory of computers has increased by roughly eight orders of magnitude (per unit cost) relative to computers available in 1960, when Kalman published his famous paper. As a result it is now possible to run nonlinear filters in real time on a single PC for many practical applications. Real time computational complexity depends on several factors, including: 1) dimension (d) of the state vector being estimated, 2) sparseness of the covariance matrix or coupling between distinct components of the state vector, 3) data rate of measurements, 4) desired accuracy of the state vector, 5) degree of nonlinearity, 6) shape of probability densities (e.g., unimodal versus multimodal). For certain low dimensional problems, it is now feasible to run a numerical approximation to the optimal nonlinear filter in real time on a single PC, depending on the data rate, estimation accuracy desired, etc. On the other hand, there are many other applications for which this is still not feasible or cost effective. This has motivated the search for more clever solutions to reduce real time computational complexity. There is no law against using more than one PC to solve a filtering problem. One can imagine using hundreds or thousands of PCs (or microprocessor chips) to speed up computations, but this approach can get rather expensive, complex and bulky.

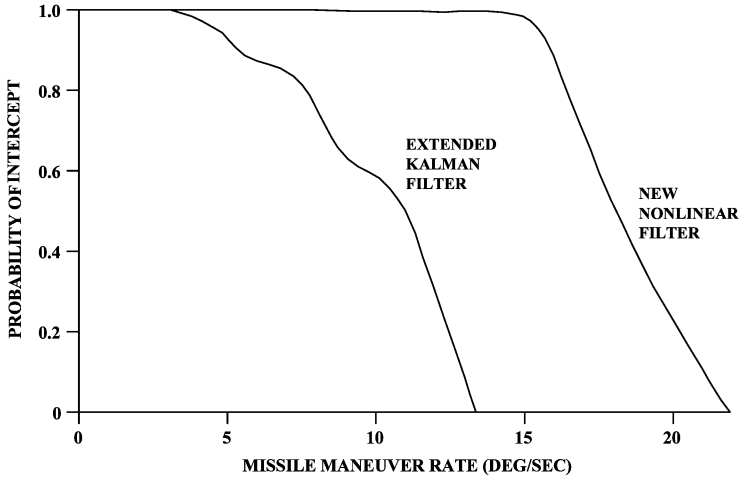

Fig. 1. New nonlinear filter versus EKF (from Schmidt [29]).

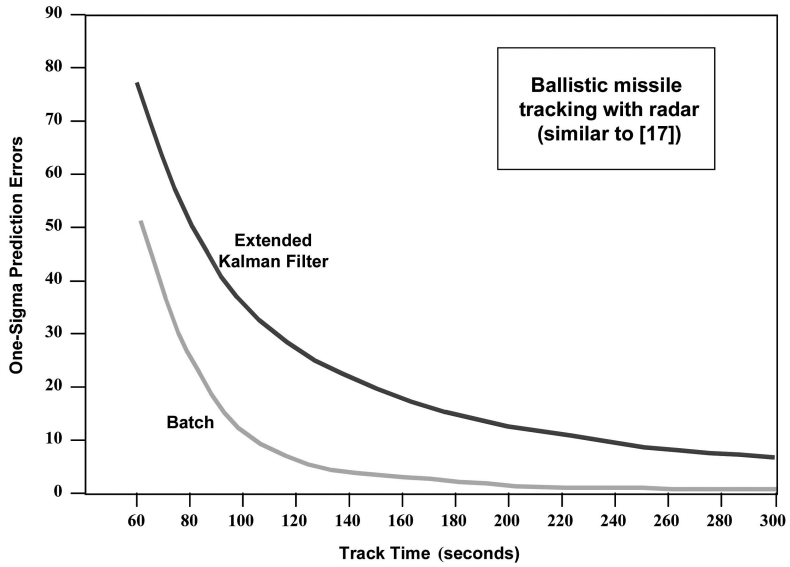

Fig. 2. Batch filter versus EKF.

The nonlinear filtering problem is to estimate the value of the d-dimensional state vector $x(t)$, at time t, given a set of noisy measurements, $Z_k = \{z(t_1), z(t_2), \ldots, z(t_k)\}$. The measurements are made at discrete times: $t_1, t_2, \ldots, t_k$. Each measurement $z(t_k)$ at time t_k is a vector of dimension m. We assume that $x(t)$ evolves in time according to a given differential equation: $dx/dt = f(x,t) + G(x,t)w(t)$, in which $w(t)$ is white process noise. We also assume that the noisy measurements are related to $x(t)$ by a known function: $z(t_k) = h(x(t_k), t_k, v(t_k))$, in which v is measurement noise (usually but not necessarily assumed to be white). The functions f, G and h can be nonlinear in x, which is why we call this a "nonlinear" filtering problem. Engineers often approximate the dynamics of $x(t)$ using a difference equation rather than a differential equation, but this may or may not be a good idea for a given application. Similarly, many academic papers assume continuous time measurements (e.g., [21]), but this is almost always a bad idea for the reasons given in [9]. More details on filtering in general are in Gelb [19], Sorenson [31–32], Jazwinski [38], Tanizaki [35], and Ristic [39].

II. EXTENDED KALMAN FILTERS

The Kalman filter theory applies to linear-Gaussian problems, but most important real world applications are nonlinear and/or non-Gaussian. Engineers use linear approximations to make this theory fit the nonlinear problems that are encountered in the real world. Such linear approximations are extremely easy to use, which explains the popularity of the extended Kalman filter (EKF). A "linear" filter problem means that all functions (f, G, and h) are linear in x, and a "Gaussian" problem means that all noise (both w and v) are Gaussian. The EKF simply approximates f, G, and h with first-order Taylor series evaluated at the current estimate of x. Nothing could be easier than

TABLE I
Comparison of Algorithms for Nonlinear Filtering

Item	Extended Kalman Filters	Unscented Kalman Filters	Particle Filters	Numerical Solution of the Fokker-Planck Equation	Exact Nonlinear Recursive Filters	Batch or Nonrecursive Filters
Statistics Propagated by the algorithm	Mean vector and covariance matrix	Mean vector and covariance matrix	Complete probability density conditioned on the measurements	Complete probability density conditioned on the measurements	Sufficient statistics	Mean vector and covariance matrix
Prediction of statistics from one measurement time to the next	Linear approximation of dynamics	Approximation of the multidimensional integrals using the "unscented transformation"	Monte Carlo integration using importance sampling	Finite difference or other numerical solution of the Fokker-Planck PDE	Numerical integration of ODEs for the sufficient statistics in the exponential family of probability densities	Numerical integration
Correction of statistics at measurement time	Linear approximation of the measurement equations	Approximation of the multidimensional integrals using the "unscented transformation"	Monte Carlo sampling of the conditional density using both importance sampling & resampling	Bayes' rule	Bayes' rule for exponential family of probability densities	Numerical minimization of cost criterion (e.g., log likelihood)
Accuracy of state vector estimate	Sometimes good but often poor compared with theoretically optimal accuracy	Often provides a significant improvement relative to the EKF, but sometimes it does not	Optimal performance for low dimensional problems, but can be highly suboptimal for high dimensions as limited by real time computer speed	Optimal performance if designed carefully, but at the cost of enormous computational complexity for high dimensional problems	Significant improvement relative to the EKF for some applications	Significant improvement relative to the EKF for some applications
Computational complexity of real time algorithm	On the order of d^3 for estimating state vectors of dimension d	Roughly the same as the EKF	Beats the curse of dimensionality for "nice" problems with a carefully designed PF, but not otherwise	Suffers from the curse of dimensionality for fixed grids, but not necessarily for adaptive grids	Roughly the same order as the EKF for most practical problems, and polynomial in d in general	For zero process noise, roughly the same as the EKF, but for non-zero process noise it is much larger than the EKF
References	17, 19, 25, 26, 31, 35 & 38	5, 23 & 24	1, 6, 16, 18, 20 & 22	4, 27, 30 & 38	2, 7, 9, 21 & 29	28, 34 & 36

this. Actually, there is no such thing as "the" EKF, but rather there are hundreds of varieties of EKFs. In particular, different EKFs can be designed using a long list of engineering tricks: 1) different coordinate systems, 2) different factorizations of the covariance matrix, 3) second order (or higher order) Taylor series corrections to the state vector prediction and/or the measurement update, 4) iteration of the state vector update using measurements, 5) different orders with which to use sequential scalar valued measurements to update the state vector, 6) tuning process noise, 7) quasi-decoupling, and 8) combinations of all of the above, as well as other bells and whistles invented by engineers in the hope of improving EKF performance. Sometimes these practical tricks result in significant improvement, but often they result in no improvement or they make performance worse. Table I lists several references that give details on these practical tricks for EKFs. But the literature is very biased; engineers rarely publish papers with examples of methods that don't work. Moreover, there is no quantitative theory that predicts when a given method will work well.

The notion that different coordinate systems (e.g., Cartesian and polar) can result in significantly different accuracy of the EKF is sometimes troublesome for physicists and mathematicians, because the "physics" doesn't depend on what coordinate system is used to describe it, and mathematicians feel that "coordinate free" descriptions are simpler, more general and more powerful. Linear

as well as nonlinear filter problems have exactly the same theoretical optimal estimation accuracy in any coordinate system. Nevertheless, engineers know that EKF accuracy can depend on the specific coordinate system used, owing to nonlinearities and/or ill-conditioning. Moreover, there is no law against using more than one coordinate system in a given EKF. Such hybrid coordinate systems have been used in EKFs with good effect for radars for many decades (see [26]).

III. UNSCENTED KALMAN FILTERS

The EKF is based on a simple linear approximation to the nonlinear equations that describe the physical problem, whereas the unscented Kalman filter (UKF) does not use this simple linear approximation. But rather the UKF uses a more accurate approximation to evaluate the multidimensional integrals required by theory. This approximation is called the "unscented transformation," for obvious reasons, and it is similar to Gauss-Hermite quadrature for numerical approximation of multidimensional integrals. The UKF assumes that the probability density of the state vector is Gaussian, and this density is sampled at a number of carefully chosen points to approximate the multidimensional integrals required. Crassidis and Markley [5] show one application in which the UKF is vastly superior to the EKF. The linear approximation used by the EKF can be extremely bad for some smooth nonpathological nonlinear functions. In particular, the EKF approximates the expected value of $f(x)$ as $E[f(x)] \approx f(E(x))$, which is exact for linear functions, but can be a very bad approximation otherwise. For example, consider the simple nonlinear function $f(x) = x^2$. Assuming that the mean of x is zero, $E(x) = 0$, results in the EKF approximation that $E(x^2) \approx [E(x)]^2 = 0$, whereas the correct value of $E(x^2)$ could be any positive number (the variance of x). That is, the simple linear approximation used by the EKF in this case can be wrong by an arbitrarily large amount! In contrast, the UKF approximation is generally much better than the EKF, and the computational complexity of the UKF and EKF are roughly the same. For more details on UKFs see the references in Table I.

As noted above, UKF estimation accuracy is vastly superior to the EKF for the application studied in [5]. However, such comparisons are often viewed skeptically by hardboiled engineers, who know that there is no such thing as "the" EKF. In particular, one should ask whether this EKF was designed and tuned carefully, by exploiting the litany of bells and whistles described in Section II. Moreover, any comparison based on Monte Carlo simulations should not be trusted on face value, owing to the possibility of bugs in the EKF and vagaries of complex algorithms. One yearns for a simple back-of-the-envelope formula that could explain the results of such Monte Carlo simulations.

One of the mysteries of the UKF is why it is called "unscented." This has never been explained in papers on the UKF, and although I have enjoyed several lectures by the inventors of the UKF, nobody in the audience had the courage to ask this obvious question. But I feel that this should remain a mystery, as it is one of the charming aspects of nonlinear filtering.

TABLE II
Varieties of Particle Filters

Item	Varieties
1. Proposal density	Extended Kalman filter (Gaussian) Unscented filter (Gaussian) Daum filter (exponential family) Gaussian sums Sums of exponential family Other
2. Sampling	Rejection Naïve importance sampling Metropolis-Hastings importance sampling Gibbs sampling Other
3. Re-sampling	Every update Sparse Adaptive Other
4. Representation of conditional density	Particles Smooth Kernels (F. Le Gland, et al.) Other
5. Generation of samples	Deterministic grid Monte Carlo Quasi-Monte Carlo Hybrid Other
6. Variance reduction methods	Stratified Sampling Rao-Blackwellization Control variates Antithetic variables Other

IV. PARTICLE FILTERS

Particle filters (PFs) are a novel class of nonlinear filters announced in a seminal and extremely lucid paper [20] about a decade ago. Since then, there has been an explosion of work on PFs, resulting in a plethora of papers and applications. Like the EKF, however, there is no such thing as "the" particle filter, but rather there is a huge variety of PFs, as shown in Table II. One can code a pretty good PF in a page or two of MATLAB. Also, you don't have to know the finer points of stochastic calculus or the Fokker-Planck equation (FPE) or any fancy numerical

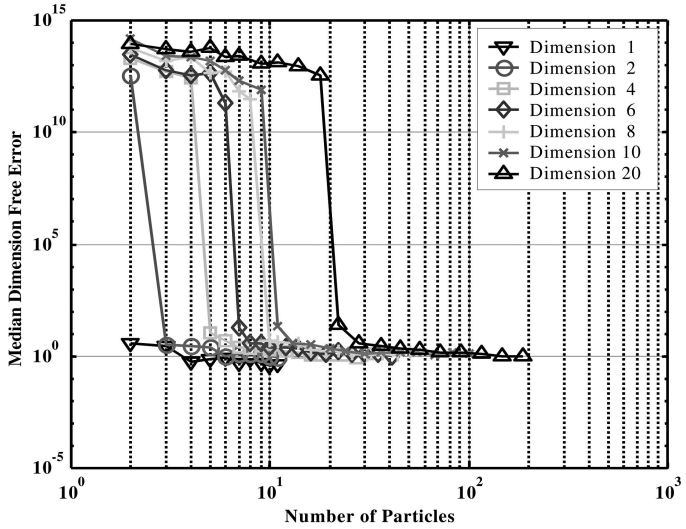

Fig. 3. Dimension-free error versus number of particles for PF with good proposal density.

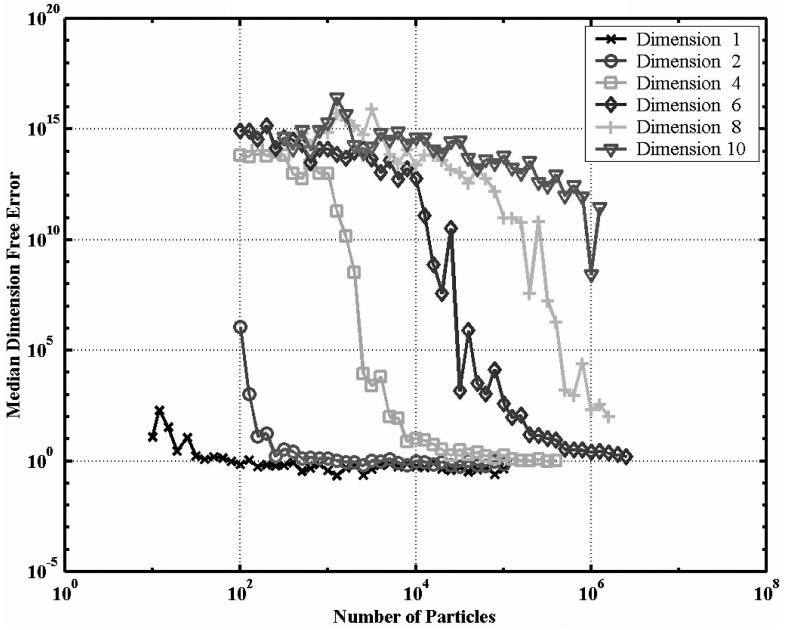

Fig. 4. Dimension-free error versus number of particles for PF with poor proposal density.

methods for solving partial differential equations (PDEs). PFs approximate the complete non-Gaussian probability density of the state vector conditioned on the measurements. Multidimensional integration is approximated using Monte Carlo sampling, which is the hallmark of PFs.

The key issue in nonlinear filters of any kind is the "curse of dimensionality." This is the phrase coined by Richard Bellman over forty years ago to describe the exponential growth of computational complexity as a function of the dimension of the state vector (x). The computational complexity of the Kalman filter grows as the cube of dimension, but for general nonlinear problems using filters that achieve optimal accuracy, the computational complexity grows exponentially in dimension. This is not an academic issue, but rather it is fundamentally important to engineers in the real world, even with computers that run eight orders of magnitude faster per dollar than when Kalman published his seminal paper in 1960. It has been widely asserted that PFs beat the curse of dimensionality; however, this is generally not true. A detailed analysis of the computational complexity of PFs for a given estimation accuracy is in [16]. For low dimensional problems, well-designed PFs achieve optimal estimation accuracy with a computational complexity roughly the same as a EKF, as shown in Fig. 3. But in general for high dimensional problems the computational complexity of the best designed PFs is enormous, as shown in Fig. 4. That is, PFs suffer from the curse of dimensionality in general, contrary to popular assertions. There is now a

vast literature on PFs which is aimed at lowering computational complexity for a given estimation accuracy. For example, [13] uses quasi-Monte Carlo sampling as well as exploiting the smoothness of the underlying probability densities to reduce computational complexity of PFs. There is a simple back-of-the-envelope formula derived in [16] that bounds the computational complexity of carefully designed PFs for "nice" filtering problems.

The key ingredient that makes PFs work for high dimensional problems is a good proposal density, which is used to sample new particles with Monte Carlo methods. In almost all successful PFs, the proposal density is a Gaussian density obtained from an EKF or UKF. If the problem is "nice," then the EKF or UKF will produce a sufficiently accurate Gaussian proposal density to allow the Monte Carlo samples to be concentrated in the important volume of state space, but not otherwise. If we have a poor proposal density (e.g., because the EKF or UKF diverge), then the Monte Carlo samples will be mostly wasted, because they are not concentrated in the relevant volume in state space. In low dimensional problems this is not a significant issue, but in high dimensional problems the vast majority of particles could be sampled in irrelevant volumes in state space. This is the intuitive explanation of the curse of dimensionality for any Monte Carlo method, including PFs. Fig. 3 and 4 compare the performance of PFs for exactly the same problem with a good proposal density and a poor proposal density. The difference in performance is dramatic. The poor proposal density was not chosen with malice aforethought, but rather it is the result of using the algorithm in the seminal paper on PFs [20]. Some researchers call [20] the "classic" PF, but I prefer the term "plain vanilla," to distinguish it from PFs with more bells & whistles (e.g., [13, 18 and 22]). The recent book cleverly entitled *Beyond the Kalman Filter* [39] contains a wealth of new results on PFs, and it is must reading.

V. EXACT NONLINEAR FILTERS

The Kalman filter is "exact" for linear systems with Gaussian noise. More generally, a filter is "exact" if it provides optimal estimation accuracy with an algorithm that only requires solving ordinary differential equations (ODEs) in real time, but it does not require solving PDEs in real time. It would be nice if there were exact filters (like the Kalman filter) for all nonlinear problems, but this is too much to hope for. In theory to solve a nonlinear filtering problem one needs the solution of the FPE, which is a PDE that describes the evolution of the probability density of the state vector conditioned on the measurements. However, accurate numerical approximation of PDEs suffers from the curse of dimensionality in general. We would like to avoid solving high dimensional PDEs in real time. In contrast, engineers have routinely solved ODEs in real time using numerical approximations for many decades. The key idea of "exact" nonlinear filters is to transform a PDE into a system of ODEs exactly. The Kalman filter is one example of this, and there are several others, as explained below.

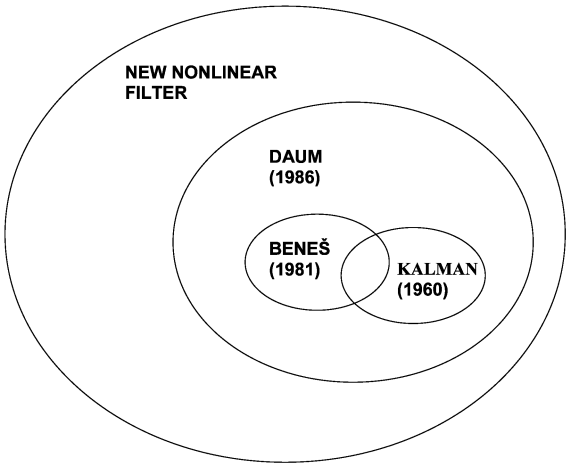

Fig. 5. Relationship of filters.

Two decades ago Beneš published a seminal paper [2] which derives an exact filter for a special class of nonlinear problems. However, the Beneš filter does not solve all linear problems that the Kalman filter solves exactly. Benes' theory was generalized by Daum [8] to handle a much broader class of nonlinear problems as well as solving all Kalman filter problems. Fig. 1 shows the application of this theory for an important real world problem. Fig. 5 and Table III summarize the relationship between the Kalman filter, the Beneš' filter, and four progressively more general exact nonlinear filters derived more recently. The Kalman filter is based on the Gaussian probability density, whereas the more general exact filters use the exponential family of multivariate probability densities. The Gaussian density is a special case of the exponential family, just as the Kalman filter is a special case of the Daum filter [9]. The Kalman filter and Daum filter are compared in Table IV, and a block diagram of the most general Daum filter is given in Fig. 6. A tutorial introduction to exact nonlinear filters is given in [9], with more details in [7], [8], [11], [12], and [14].

Filters that avoid solving PDEs in real time are also called "exact finite dimensional filters," because the state vector of the filter itself has a finite dimension that is fixed (i.e., it does not grow as more measurements are collected). For example, the dimension of the Kalman filter is $M = d + d(d+1)/2$, for estimating a state vector of dimension d. The Kalman filter needs to remember the mean vector (dimension d) and the symmetric covariance matrix,

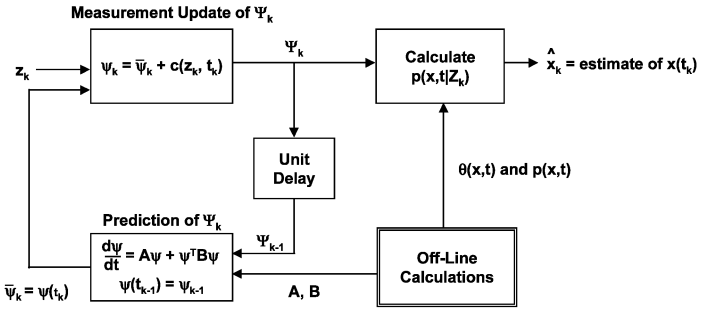

Fig. 6. New nonlinear filter.

TABLE III
Exact Nonlinear Recursive Filters

Filter	Probability Density of x Conditioned on All Measurement $p(x,t \mid Z_k)$	Class of Dynamics	Propagation Equations
1. Kalman	η = Gaussian	$\frac{\partial f}{\partial x} = A(t)$	$\frac{dm}{dt} = Am$ $\frac{dP}{dt} = AP + PA^T + GG^T$
2. Beneš [2]	$\eta \exp\left[\int^x f(x)dx\right]$	$\frac{\partial f}{\partial x} = \left(\frac{\partial f}{\partial x}\right)^T$ and $\|f(x)\|^2 + \mathrm{tr}\left(\frac{\partial f}{\partial x}\right) = x^T Ax + b^T x + c$	$\frac{dm}{dt} = -PAm - \frac{1}{2}Pb$ $\frac{dP}{dt} = I - PAP$
3. Daum [8]	$\eta P_{ss}^{\alpha}(x)$	$f - \alpha Qr^T = Dx + E$ and $\mathrm{tr}\left(\frac{\partial f}{\partial x}\right) + \frac{\alpha}{2} rQr^T = x^T Ax + b^T x + c$ where $r = \frac{\partial}{\partial x}\log P_{ss}(x)$	$\frac{dm}{dt} = 2(\alpha - 1)PAm + Dm + (\alpha - 1)Pb + E$ $\frac{dP}{dt} = 2(\alpha - 1)PAP + DP + PD^T + Q$
4. Daum [8]	$\eta q^{\alpha}(x,t)$	Same as filter 3, but with $r = \frac{\partial}{\partial x}\log q(x,t)$	Same as filter 3
5. Daum [14]	$\eta Q(x,t)$	$\frac{\partial f}{\partial x} - \left(\frac{\partial f}{\partial x}\right)^T = D^T - D$ and $\frac{\partial f}{\partial t} + \dot{D}x + \dot{E} = -\frac{\partial f^T}{\partial x} f - \frac{1}{2}\left[\frac{\partial}{\partial x}\mathrm{tr}\left(\frac{\partial f}{\partial x}\right)\right]^T + (2A + D^T D)x + D^T E + b$	$\frac{dm}{dt} = -(2PA + D)m - E - Pb$ $\frac{dP}{dt} = -2PAP - PD^T - DP + I$
6. Daum [7] and [9]	$p(x,t)\exp[\theta^T(x,t)\psi(Z_k,t)]$	$f = Qr^T + \left(\frac{\partial \theta}{\partial x}\right)^{\#} g + \left(I - \left(\frac{\partial \theta}{\partial x}\right)^{\#}\frac{\partial \theta}{\partial x}\right) y$ # denotes the generalized inverse of the matrix (.)	$\frac{d\psi}{dt} = A^T\psi + \Gamma$ where $\Gamma = (\Gamma_1, \Gamma_2, \ldots, \Gamma_M)^T$ with $\Gamma_j = \psi^T B_j \psi$

which requires remembering another $d(d+1)/2$ numbers. Here the dimension of the Kalman filter M is fixed and finite, and it exactly represents the Gaussian distribution of the state vector conditioned on all measurements. This is a miracle! The number of measurements grows with time, yet the dimension of the exact filter is fixed! Such miracles happen for very special systems, such as linear systems with Gaussian noise, and not otherwise. Intuitively, one should expect that such miracles are rare, just as linear systems are rare (except in textbooks). In the statistics literature, when this miracle happens we say that the problem has "fixed finite dimensional sufficient statistics." About 70 years ago Sir Ronald Fisher (a brilliant British statistician and virulent anti-Bayesian) invented the powerful concept of "sufficient statistic." Fisher suggested that the only probability densities that have fixed finite dimensional sufficient statistics are from the exponential family (which he also invented). However, Fisher did not state or prove this as a theorem. A few years later, Darmois, Koopman and Pitman (working independently of each other on

TABLE IV
Comparison of New Nonlinear Filter and Kalman Filter

Item	Kalman Filter	New Nonlinear Filter
1. Conditional density	Gaussian	Exponential Family
2. Sufficient statistic	x = conditional mean P = covariance matrix	ψ = vector of dimension M
3. Prediction equations	$\frac{d\hat{x}}{dt} = F\hat{x}$ $\frac{dP}{dt} = FP + PF^T + Q$	$\frac{d\psi}{dt} = A\psi + \psi^T B\psi$
4. Measurement update equations	$\hat{x}_k = \bar{x}_k + P_k H_k^T R_k^{-1}(z_k - H_k\bar{x}_k)$ $P_k = M_k - M_k H_k^T (H_k M_k H_k^T + R_k)^{-1} H_k M_k$	$\psi_k = \bar{\psi}_k + c(z_k, t_k)$
5. Dimension of sufficient statistic	$d + d(d+1)/2$	M
6. Dynamic model	linear	nonlinear
7. Measurement model	linear	nonlinear
8. Measurement noise	Gaussian	non-Gaussian
9. Measurements	discrete time	discrete time
10. Dynamics	continuous time	continuous time
11. Computational complexity	numerical integration of ODE	numerical integration of ODE
12. Approximation method	linearization	finite sum decomposition

three different continents) stated and proved a famous theorem that says that the only probability densities for which this miracle can happen must be from the exponential family. The Darmois-Pitman-Koopman theorem assumes that the probability density is smooth and nowhere vanishing, and it assumes a parameter estimation problem, which corresponds to a filtering problem with zero process noise. Researchers have attempted to generalize this theorem to nonlinear filtering problems with non-zero process noise, but without much success. All known examples of nonlinear filtering problems with exact fixed finite dimensional sufficient statistics (assuming smooth nowhere vanishing densities) are from the exponential family, including the Kalman filter, the Beneš filter, and all four Daum filters (see Table III). However, there is no rigorous mathematical proof of this for nonlinear filtering problems with non-zero process noise.

About twenty-five years ago Brockett attempted to understand when miracles can happen for nonlinear filtering problems. Encouraged by the spectacular success of Lie algebras in physics, resulting in many Nobel prizes, Brockett decided to use Lie algebras as the basic mathematical tool. Roughly speaking, Brockett's notion was that a given nonlinear filtering problem would have an exact finite dimensional solution when the dimension of the estimation Lie algebra was finite. Brockett's idea was developed by many smart hardworking researchers for several years, but no new practical nonlinear filters have been discovered using Lie algebras so far (see Hazewinkel [21]). After reading a typical paper on abstract Lie algebras written in the last eighty years, one would never guess that Lie algebras had any practical application. But strange as it may seem, Sophus Lie invented Lie algebras over one hundred thirty years ago in order to understand a very practical problem: when PDEs can be solved exactly. This is at the heart of the nonlinear filtering problem, because if we can solve the FPE exactly in terms of ODEs, then engineers can implement the nonlinear filter in real time. Unfortunately, despite a century of mathematical research, attempts to understand separation of variables for PDEs in terms of Lie algebras have been largely unsuccessful. One reason for this failure is that separation of variables requires finding a preferred coordinate system, whereas Lie algebras are coordinate free. Lie theory has been successful for ODEs but not for PDEs. More details and references on separation of variables and Lie theory are in [9].

The whole point of finding special nonlinear filter problems that can be solved exactly is to provide the basis for a good approximation with reasonable computational complexity. This is analogous to the Kalman filter, which is exact for linear systems with Gaussian noise, and it serves as the basis for the extremely popular approximation called the EKF. Engineers can solve exact problems using numerical approximations, or they can solve approximate problems exactly, and both types of methods are listed in Table I. This point about exact nonlinear filters is often not understood. Some people think that we

are looking for exact filters because we suppose that there are practical problems that can be solved directly with no approximations. But this is not so. Even the Kalman filter is not used without approximations. Moreover, there is no law against solving part of the filtering problem exactly and the remainder with approximations.

VI. NONRECURSIVE AND SEMI-RECURSIVE FILTERS

Gauss invented the least squares method about two hundred years ago, and he successfully applied it to the prediction of the orbit of a newly discovered asteroid, winning fame and fortune. Gauss invented both recursive least squares algorithms (like the Kalman filter) as well as nonrecursive least squares algorithms. The astronomical problems that Gauss solved are highly nonlinear, so Gauss was forced to use a linear approximation much like the EKF. Unlike the Kalman filter, Gauss's nonlinear iterative least squares algorithm was nonrecursive, meaning that Gauss stored all the measurement data, and he processed it as a single "batch" of data, rather than recursively using each discrete time measurement one-by-one. Also, Gauss's algorithm did not model process noise, unlike the Kalman filter. One might think that Gauss's two hundred year old algorithm is obsolete, given PCs with gigaflops of throughput and gigabytes of memory, but this is not so. Gauss's batch least squares is routinely used today, and it often gives accuracy that is superior to the best available EKF (see Fig. 2). The intuitive reason that the batch is superior to the EKF is that the batch relinearizes based on all data in the batch, whereas the EKF only uses the last estimated state vector to compute an improved linearization. Today, with cheap fast memory, storing all measurement data is generally not an issue. In my own experience with phased array radars, I have stored all measurement data for thousands of targets with no problem. That is, the issue of recursive versus nonrecursive filters is a red herring as far as memory is concerned. Moreover, although batch algorithms are generally slower than EKFs for a given application, computers are now eight orders of magnitude faster (per unit cost) compared with 1960, when Kalman published his famous paper. Hence, the slower speed of batch filters (with zero process noise) is also a red herring. On the other hand, batch filters with non-zero process noise can be much much slower than EKFs, but owing to the eight orders of magnitude speedup noted above, this also has become a red herring for most applications. Aubrey Poore has published an excellent paper [28] on practical batch filters with non-zero process noise, and Sorenson's book [34] is very useful for batch filters with zero process noise (so-called "parameter estimation problems"). Gauss' original papers on least squares algorithms have been recently translated (from Latin) and are available along with very interesting historical notes [36]. A few years ago it came to light that Gauss invented the fast Fourier transform (FFT) a century before Professor Runge and almost two centuries before Cooley & Tukey. Similarly, non-Euclidean geometry was invented by Gauss years before other mathematicians published their work. Gauss is famous for inventing some new brilliant seminal gadget, but not publishing it. So we should not be too surprised when [37] arrives showing that Gauss also invented the UKF several hundred years before Julier & Uhlmann.

There is no law against designing an algorithm that is part recursive and part nonrecursive. There are practical filters that in fact combine the best features of both methods. This allows more flexibility in the fundamental tradeoff of accuracy versus computational complexity. I like to call such algorithms "semi-recursive," and [15] provides some theoretical background on such filters. In the academic literature, the term "fixed finite dimensional filter" is used rather than "recursive." The word "dimension" refers to the dimension of the sufficient statistic for a given estimation problem. There is also no law against inventing a filter with a finite but growing (rather than fixed) dimension, and it appears that this is a very promising direction for future research. In particular, the set of all measurements is obviously a sufficient statistic, and its dimension grows with time. This makes sense for discrete time measurements but not for continuous time measurements.

VII. NUMERICAL SOLUTION OF THE FOKKER-PLANCK EQUATION

The FPE is a PDE that governs the evolution of the probability density of the state conditioned on the measurements. If we could approximate the solution of this equation in real time, then the nonlinear filter problem would be solved. However, the computational complexity grows exponentially with dimension of the state vector using standard textbook methods. Something more clever is needed for high dimensional problems. Faster computers help, but the curse of dimensionality still reigns, even in the year 2005. All of the methods previously discussed are attempts to avoid the curse of dimensionality, and sometimes they succeed. On the other hand, for low dimensional problems, numerical solution of the FPE is a viable method (see [40–41]). All textbook methods for numerical solutions of PDEs exploit the smoothness of both the solutions and the PDE themselves. For most practical nonlinear filtering problems there is plenty of smoothness, and thus it is reasonable to expect that numerical methods would have an advantage over PF which do not exploit any smoothness. On the other hand, PF ruthlessly exploit importance sampling, whereas textbook numerical

methods do not. The PFs that work well for high dimensional problems do so because they use a good proposal density to put particles where they will be useful. This is a good adaptive method that works in the real world. Standard textbook numerical methods for solving PDEs use a fixed grid in d-dimensional space, and the computational complexity grows as $N\hat{}d$, where N is the number of grid points in each dimension. With PCs in the year 2005 we can solve problems with low dimension (up to roughly $d = 4$ to 6), but not for higher dimensional fully-coupled problems. To avoid the curse of dimensionality we must use an adaptive grid rather than a fixed grid. Such methods are called "mesh-free" in the literature on PDEs, and significant progress has been made in the last two decades [47]. Unlike PFs, mesh-free numerical methods exploit smoothness, but they also put the nodes in the important places in state space, similar to PFs. Hence, we expect that mesh-free numerical solutions of the FPE should be superior to PFs. Moreover, there is no law against using a good proposal density (e.g. from an EKF or UKF) to position the nodes for a mesh-free numerical solution of the FPE that exploits smoothness; this would be the best of both worlds. There are several key ingredients in this new approach: 1) quantify the error in the solution of the FPE in terms of the mesh (see [42], [45–46] for recent progress); 2) use a good error criterion (e.g., the L1 norm rather than the standard L2 norm); 3) tell the mathematics what you intend to do with the solution of the FPE (e.g., use it to compute the conditional mean of x). All three ingredients are crucial, and none of them are in the old-fashioned textbook numerical methods for PDEs. For image processing the engineering folk theorem is that the L2 norm is a bad way to judge the quality of an image, whereas L1 is generally better, but we are still waiting for the paper that will announce a really good error criterion for images (see [42]). It is pointless to design an adaptive mesh that optimizes the wrong error criterion. A second sin is to use rigorously correct but not very tight bounds on errors in the solution of a PDE to design the adaptive mesh. The rigorous bounds are correct but extremely bad (see [42]). Instead, an engineer should use her intuition and cook-up a good heuristic measure of error, which is unlikely to be rigorously correct, but which could give good results (see [42]). Both [42] and [43] emphasize the importance of telling the mathematics what you are going to do with the solution of the PDE. In particular, for nonlinear filtering we might like to compute the conditional mean of x, which is a linear functional of the solution of the FPE, and this allows us to use the powerful "adjoint method" to estimate errors (see [42], [45–46]). It is interesting that [43] uses Monte Carlo methods, whereas [42] uses adaptive mesh-free numerical solutions of PDEs, but both put great emphasis on the importance of telling the mathematics what you are trying to do. This results in much improved accuracy, and it seems like common sense.

VIII. FUTURE RESEARCH

There is much work to be done on nonlinear filtering. On the practical side, new clever approximations are sure to be invented by engineers, similar to the plethora of engineering tricks used in EKFs and PFs. On the theoretical side, we should develop nonrecursive and semi-recursive filters, as well as exact nonlinear filters with finite but growing dimension. We should prove a generalization of the Darmois-Koopman-Pitman theorem for smooth nonlinear filtering problems with non-zero process noise, as well as for finite but growing dimensional problems, and for problems with less smoothness and less regularity. We should attempt to exploit smoothness in PFs and quasi-Monte Carlo sampling in PFs. Rather than using boring old Gaussian proposal densities, PFs should also use proposal densities from the exponential family and exponential sums (analogous to Gaussian sums). One reason for the difficulty of using Lie algebras and Lie groups for nonlinear filters is that the FPE does not generate a group, but rather it only generates a semi-group, because there is no smooth inverse solution. But we know that two FPEs (one propagating forwards in time and the other backwards in time) are equivalent to two Schrödinger equations (each of which generates a group), so this idea from physics can be exploited to understand when miracles happen for nonlinear filters (see [15]).

Several other ideas can be borrowed from physics to solve the FPE. In particular, physicists have solved the Schrödinger equation using "dimensional interpolation" with good accuracy for high dimensional problems [44]. Dimensional interpolation starts by creating a very large nonphysical dimension for molecules with many atoms, and uses this to approximate the solution of the Schrödinger equation with interpolation (of dimension!) between 1 and infinity (which are both easy problems). This idea sounds crazy on three counts: 1) increasing dimension to make the problem easier (which is the exact opposite of what any sensible person would do); 2) creating a nonphysical dimension for 3-dimensional molecules! and 3) interpolating dimensions! Nevertheless, this crazy idea allows chemists to solve the Schrödinger equation for molecules consisting of many more atoms than before. This crazy idea borrowed from physics can be combined with the idea described above that relates two FPEs with two Schrödinger equations, and we are in business. Better yet, apply dimensional interpolation to the FPE directly.

Another idea from physics is to relate Boltzmann's entropy and the second law of thermodynamics to the evolution of uncertainty in nonlinear filters. I tried to do this myself thirty years ago, without much success, but Sanjoy Mitter has recently succeeded [48]. Mitter untangles the relationship between Shannon's information theoretic entropy with nonlinear filters and thermodynamic entropy. This is particularly exciting, because the exponential family of probability densities is a maximum entropy family, and it is the only family with fixed finite dimensional sufficient statistics. Maybe new exact nonlinear filters can be invented by pursuing Professor Mitter's program.

There is no law against combining ideas from Table I, for example, a semi-recursive PF using quasi-Monte Carlo sampling using a proposal density from the exponential family might be useful. Finally, we do not have a good quantitative theory that tells us when EKFs (or any other kind of approximation) should work well in a given application. This lacuna is a scandalous situation more than four decades after the Kalman filter was invented.

REFERENCES

[1] Arulampalam, M. S., Maskell, S., Gordon, N., and Clapp, T.
A tutorial on particle filters for online nonlinear/non-Gaussian bayesian tracking.
Special Issue of IEEE Transactions on Signal Processing, **50**, 2 (Feb. 2002).

[2] Beneš, V. E.
Exact finite-dimensional filters for certain diffusions with nonlinear drift.
Stochastics, **5** (1981), 65–92.

[3] Brigo, D., Hanzon, B., and LeGland, F.
Approximate nonlinear filtering by projection on exponential manifolds of densities.
Bernoulli (1999), 495–534.

[4] Challa, S., and Bar-Shalom, Y.
Nonlinear filter design using Fokker-Planck-Kolmogorov probability density evolutions.
IEEE Transactions on Aerospace and Electronic Systems, **36**, 1 (Jan. 2000).

[5] Crassidis, J. L., and Markley, F. L.
Unscented filtering for spacecraft attitude estimation.
AIAA Journal on Guidance, Control and Dynamics, (July–Aug. 2003).

[6] Crisan, D., and Doucet, A.
A survey of convergence results on particle filtering methods for practitioners.
Special Issue of IEEE Transactions on Signal Processing (Mar. 2002).

[7] Daum, F. E.
Industrial strength nonlinear filters.
In *Proceedings of Workshop on Estimation, Tracking and Fusion: A Tribute to Yaakov Bar-Shalom*, Monterey, CA, May 2001.

[8] Daum, F. E.
Exact finite dimensional nonlinear filters.
IEEE Transactions on Automatic Control, **AC-31**, 7 (July 1986), 616–622.

[9] Daum, F. E.
New exact nonlinear filters.
In J. C. Spall (Ed.), *Bayesian Analysis of Time Series and Dynamic Models*.
New York: Marcel Dekker, 1988, ch. 8.

[10] Daum, F. E.
Solution of the Zakai equation by separation of variables.
IEEE Transactions on Automatic Control, **AC-32**, 10 (Oct. 1987), 941–943.

[11] Daum, F. E.
New exact nonlinear filters: Theory and applications.
In *Proceedings of SPIE Conference on Signal and Data Processing of Small Targets*, Orlando, FL, Apr. 1994.

[12] Daum, F. E.
Practical nonlinear filtering with the method of virtual measurements.
In *Proceedings of SPIE Conference on Signal and Data Processing*, San Diego, CA, 1997.

[13] Daum, F. E., and Huang, J.
Nonlinear filtering with quasi-Monte Carlo methods.
In *Proceedings of SPIE Conference on Signal and Data Processing*, San Diego, CA, Aug. 2003.

[14] Daum, F. E.
New nonlinear filters and exact solutions of the Fokker-Planck equation.
In *Proceedings of IEEE American Control Conference*, Seattle, WA, 1986, 884–888.

[15] Daum, F. E., and Huang, J.
Physics based computational complexity of nonlinear filters.
In *Proceedings of SPIE Conference on Signal and Data Processing*, Vol. 5428, Orlando, FL, Apr. 2004.

[16] Daum, F. E., and Huang, J.
The curse of dimensionality for particle filters.
In *Proceedings of IEEE Conference on Aerospace*, Big Sky, MT, Mar. 2003.

[17] Daum, F. E., and Fitzgerald, R. J.
Decoupled Kalman filters for phased array radar tracking.
IEEE Transactions on Automatic Control (Mar. 1983).

[18] Doucet, A., de Freitas, N., and Gordon, N. (Eds.)
Sequential Monte Carlo Methods in Practice.
New York: Springer-Verlag, 2001.

[19] Gelb, A., et al.
Applied Optimal Estimation.
Cambridge, MA: MIT Press, 1974.

[20] Gordon, N. J., Salmond, D. J., and Smith, A. F. M.
Novel approach to nonlinear/non-Gaussian Bayesian state estimation.
IEE Proceedings, Pt. F, **140**, 2 (1993), 107–113.

[21] Hazewinkel, M., and Willems, J. C. (Eds.)
Stochastic Systems: The Mathematics of Filtering and Identification and Applications.
Dordrecht, The Netherlands: D. Reidel Publishing Co., 1981.

[22] Special Issue on Monte Carlo Methods for Statistical Signal Processing.
IEEE Transactions on Signal Processing, **50**, 2 (Feb. 2002).

[23] Ito, K., and Xiong, K.
Gaussian filters for nonlinear filtering problems.
IEEE Transactions on Automatic Control (May 2000).

[24] Julier, S., Uhlmann, J., and Durrant-Whyte, H. F.
A new method for the nonlinear transformation of means and covariances in filters and estimators.
IEEE Transactions on Automatic Control (Mar. 2000).

[25] Leskiw, D. M., and Miller, K. S.
Nonlinear estimation with radar observations.
IEEE Transactions on Aerospace and Electronic Systems (Mar. 1982).

[26] Mehra, R. K.
A comparison of several nonlinear filters for reentry vehicle tracking.
IEEE Transactions on Automatic Control (Aug. 1971).

[27] Musick, S., Greenewald, J., Kreucher, C., and Kastella, K.
Comparison of particle method and finite difference nonlinear filters for low SNR target tracking.
In *Proceedings of International Conference on Fusion*, Montreal, Aug. 2001.

[28] Poore, A,. et al.
Batch maximum likelihood and MAP estimation with process noise.
In *Proceedings of SPIE Conference*, Aug. 2003.

[29] Schmidt, G. C.
Designing nonlinear filters based on Daum's theory.
AIAA Journal of Guidance, Control and Dynamics, **16**, 2 (Mar.–Apr. 1993).

[30] Simandl, M., Kralovec, J., and Soderstrom, T.
Anticipative grid design in point-mass approach to nonlinear state estimation.
IEEE Transactions on Automatic Control, **47**, 4 (Apr. 2002).

[31] Sorenson, H. W.
On the development of practical nonlinear filters.
Information Science, **7** (1974), 253–270.

[32] Sorenson, H. W.
Kalman Filtering: Theory and Applications.
New York: IEEE Press, 1985.

[33] Sorenson, H. W.
Recursive estimation for nonlinear dynamic systems.
In J. C. Spall (Ed.) *Bayesian Analysis of Time Series and Dynamic Models*.
New York: Marcel Dekker, 1988, ch. 6.

[34] Sorenson, H. W.
Parameter Estimation.
New York: Marcel Dekker, 1980.

[35] Tanizaki, H.
Nonlinear Filters (2nd ed.).
New York: Springer-Verlag, 1996.

[36] Gauss, C. F.
Theory of the Combination of Observations Least Subject to Errors (translated by G. W. Stewart).
Philadelphia: SIAM Publishers, 1995.

[37] Gauss, C. F.
Unstinken methode der kleinsten quadrate.
Free translation of various works by Gauss on least squares.
Werke, Königlichen Gesellschaft der Wissenschaften zu Göttingen, 1870–1928.

[38] Jazwinski, A. H.
Stochastic Processes and Filtering Theory.
New York: Academic Press, 1970.

[39] Ristic, B., Arulampalam, S., and Gordon, N.
Beyond the Kalman Filter.
Boston: Artech House, 2004.

[40] Kastella, K., and Kreucher, C.
Multiple model nonlinear filtering for low signal ground target applications.
To appear in *IEEE Transactions on Aerospace and Electronic Systems*.

[41] Kastella, K.
Finite difference methods for nonlinear filtering and automatic target recognition.
In Y. Bar-Shalom and W. D. Blair (Eds.), *Multitarget/Multisensor Tracking, Vol. III*, Boston: Artech House, 2000.

[42] Bangerth, W., and Rannacher, R.
Adaptive Finite Element Methods for Differential Equations.
Boston: Birkhauser, 2003.

[43] Mira, A.
Variance reduction in MCMC.
The Second Workshop on Monte Carlo Methods, Harvard University, Aug. 28, 2004.

[44] Herschbach, D. R. (Ed.)
Dimensional scaling in chemical physics.
Boston: Kluwer Academic Publishers, 1993.

[45] Giles, M. B., and Suli, E.
Adjoint methods for PDEs.
Acta Numerica (2002), 145–236.

[46] Pierce, N., and Giles, M. B.
Adjoint recovery of superconvergent functionals from PDE approximations.
SIAM Review, **42**, 2 (2000), 247–264.

[47] Belytschko, T., et al.
Meshless methods: An overview and recent developments.
Computational Methods of Applied Mechanical Engineering (1996), 3–47.

[48] Mitter, S. K., and Newton, N. J.
Information flow and entropy production in the Kalman-Bucy filter.
Journal of Statistical Physics (Jan. 2005).

Fred Daum (S'86—M'86) is a graduate of Harvard University, Cambridge, MA.

He is a senior principal fellow, which is the highest technical rank at Raytheon. He developed, analyzed, and tested the real time algorithms for essentially all the large phased array radars built by the United States in the last three decades, including Cobra Dane, PAVE PAWS, Cobra Judy, and BMEWS. He has also worked on THAAD, UEWR, and ROTHR, as well as several air traffic control systems and shipboard fire control systems. These real time algorithms include: Kalman filters, radar waveform scheduling, Bayesian discrimination, data association, track initiation, discrimination of satellites from missiles, calibration of tropospheric and ionospheric refraction, and target object mapping, for which he was recently given the "Radar Scottie Award" for his work on THAAD. Moreover, he single-handedly developed, analyzed and tested the real time Bayesian discrimination algorithms for Cobra Dane, Cobra Judy, PAVE PAWS and BMEWS. He has given invited lectures at MIT, Harvard, Yale, Brown, Georgia Tech., University of Connecticut, University of Minnesota, University of Illinois at Chicago, and Northeastern University.

He has published nearly one hundred technical papers, and he was awarded the Tom Phillips prize for technical excellence, in recognition of his ability to make complex radar systems work in the real world.