Bayesian particle flow for estimation, decisions & transport

Fred Daum & Jim Huang

23 September 2014

nonlinear filter problem*

dynamical model of state:

$$\begin{cases} dx = F(x,t)dt + G(x,t)dw \quad \text{or} \\ \mathrm{x}(\mathrm{t}_{\mathrm{k+1}}) = F(x(t_k), t_k, w(t_k)) \end{cases}$$

$x(t) =$ state vector at time t

$w(t) =$ process noise vector at time t

$z_k =$ measurement vector at time t_k

$z_k = H(x(t_k), t_k, v_k)$

$v_k =$ measurement noise vector at time t_k

$p(x, t_k | Z_k) =$ probability density of x at time t_k given Z_k

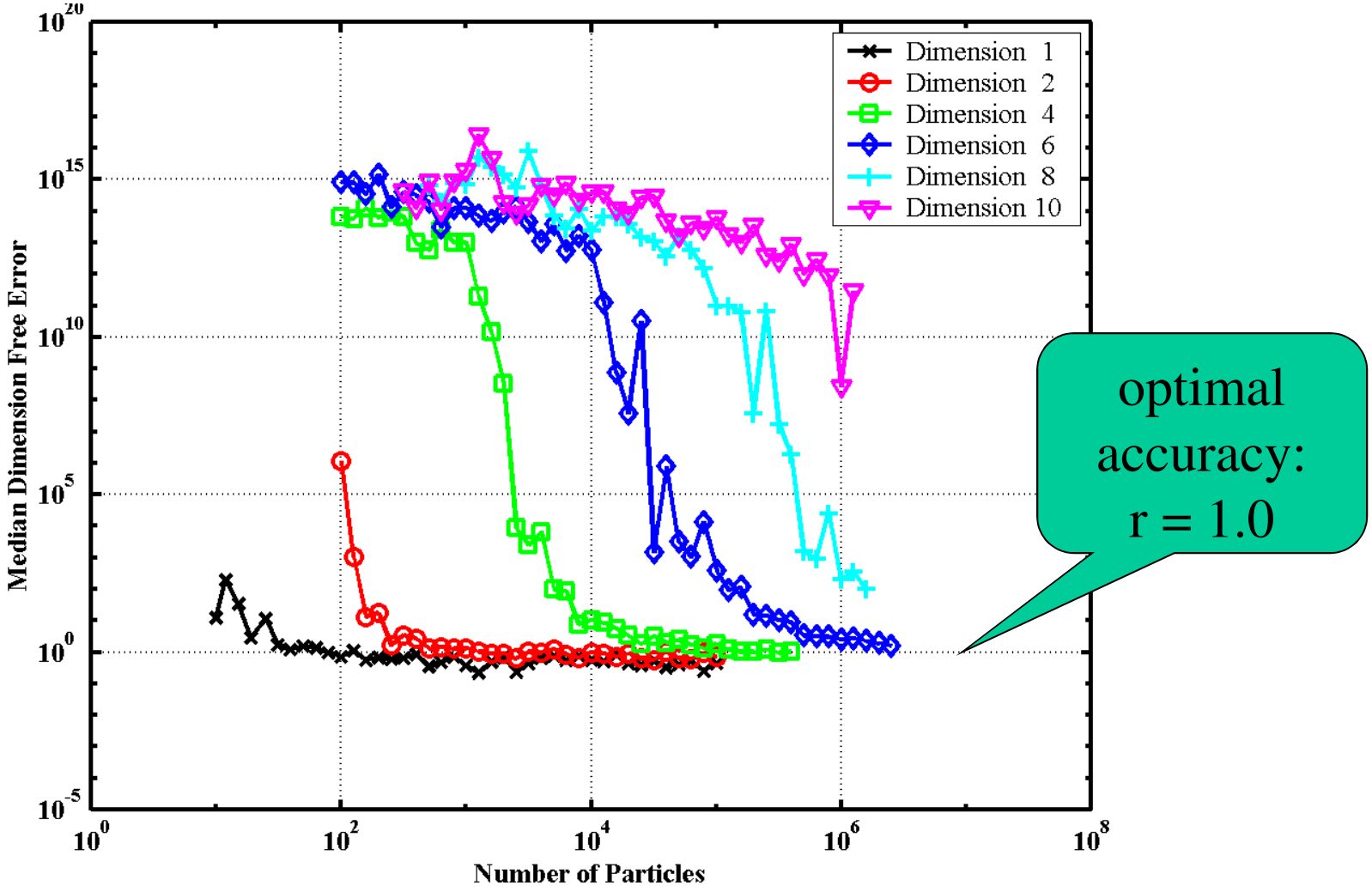
$Z_k =$ set of all measurements

$\mathrm{Z}_\mathrm{k} = \{z_1, z_2, ..., z_k\}$

estimate x given noisy measurements

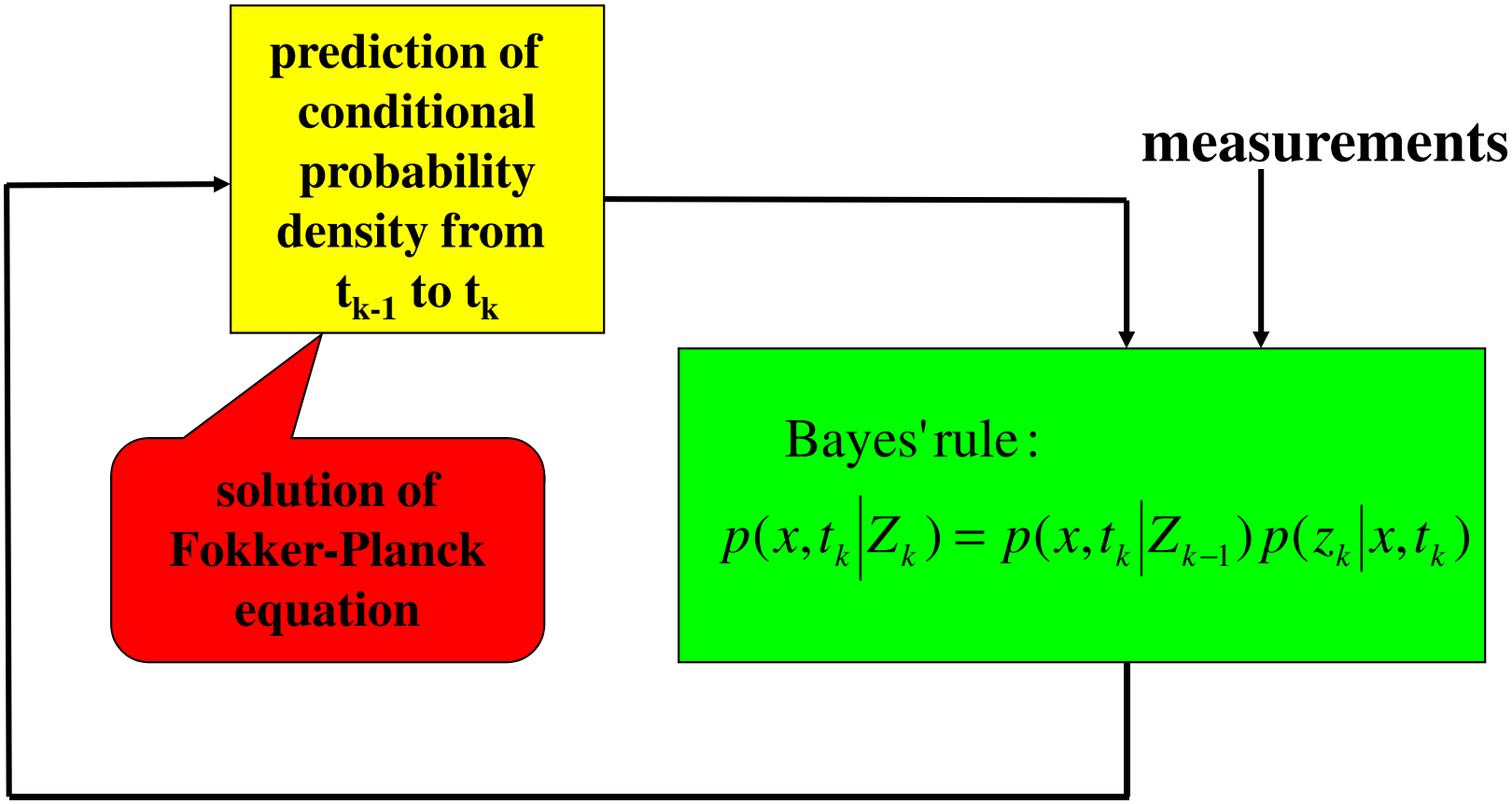
"The Oxford handbook of nonlinear filtering," edited by Dan Crisan and Boris Rozovskii, Oxford University Press, 2011.

curse of dimensionality for classic particle filter*



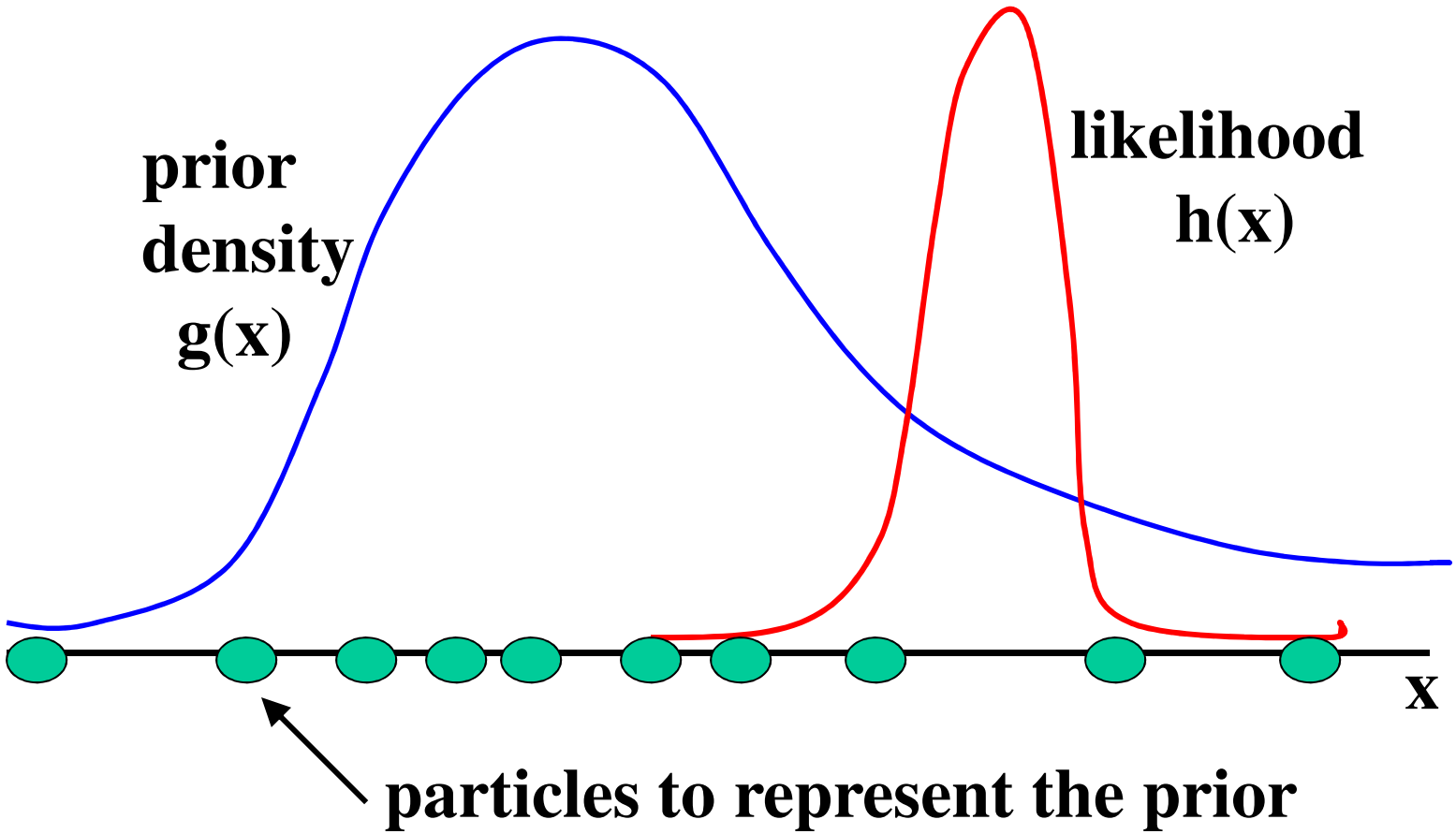
*Daum, IEEE AES Systems Magazine, August 2005.

nonlinear filter*



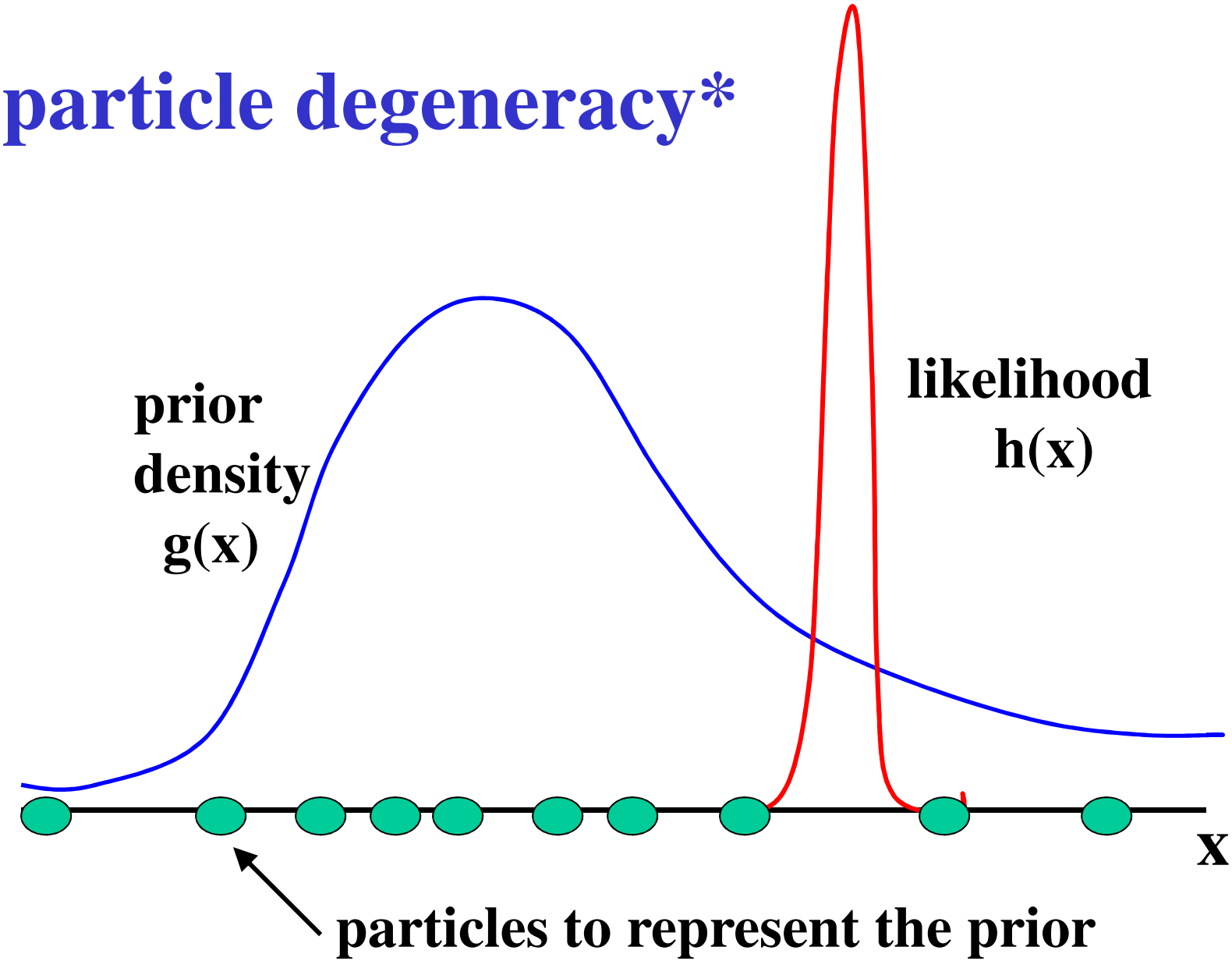
*Yu-Chi Ho & R. C. K. Lee, "A Bayesian approach to problems in stochastic estimation and control," IEEE Transactions on automatic control, pages 333-339, October 1964.

particle degeneracy*



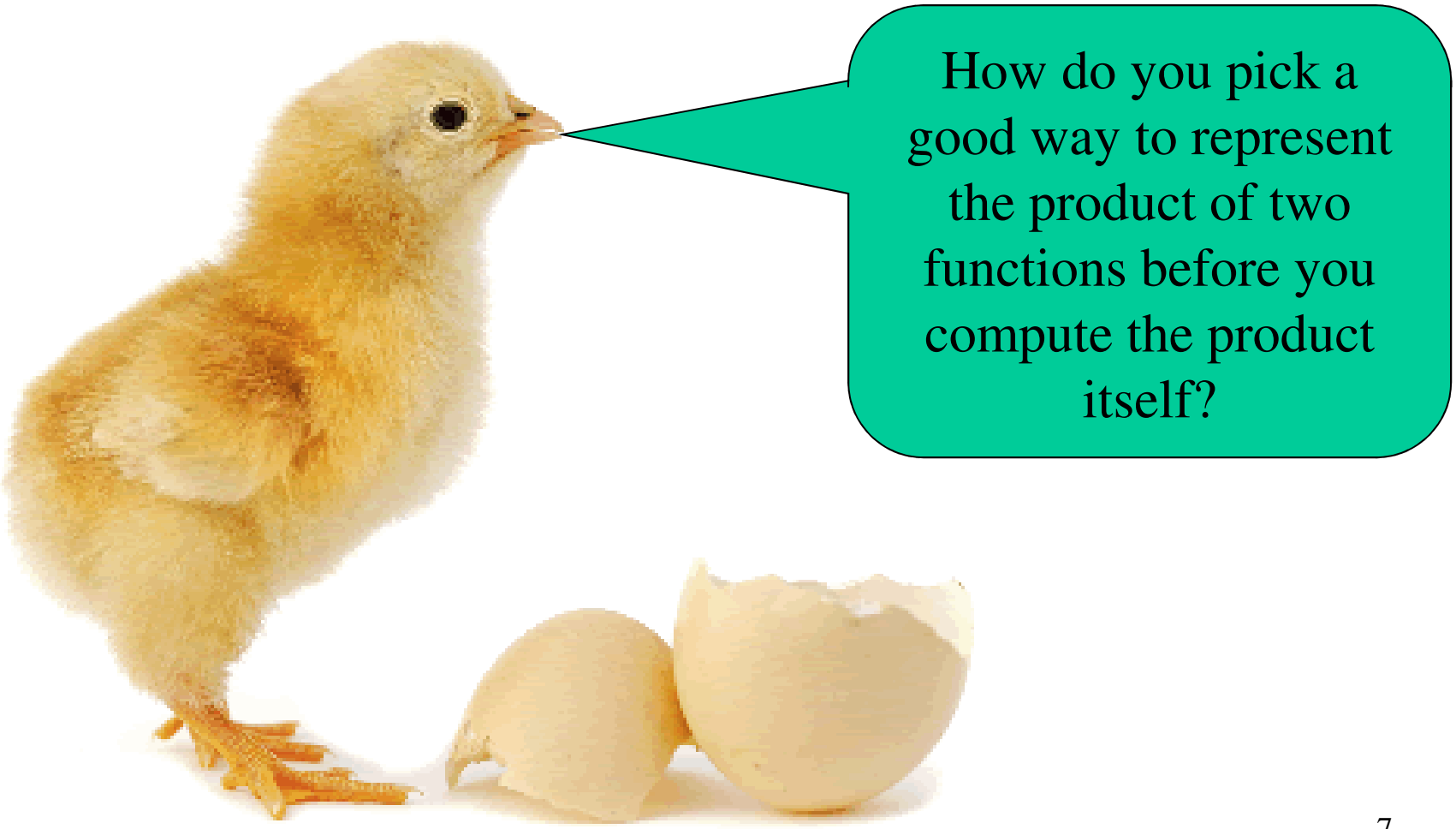
*Daum & Huang, "Particle degeneracy: root cause & solution," SPIE Proceedings 2011.

particle degeneracy*

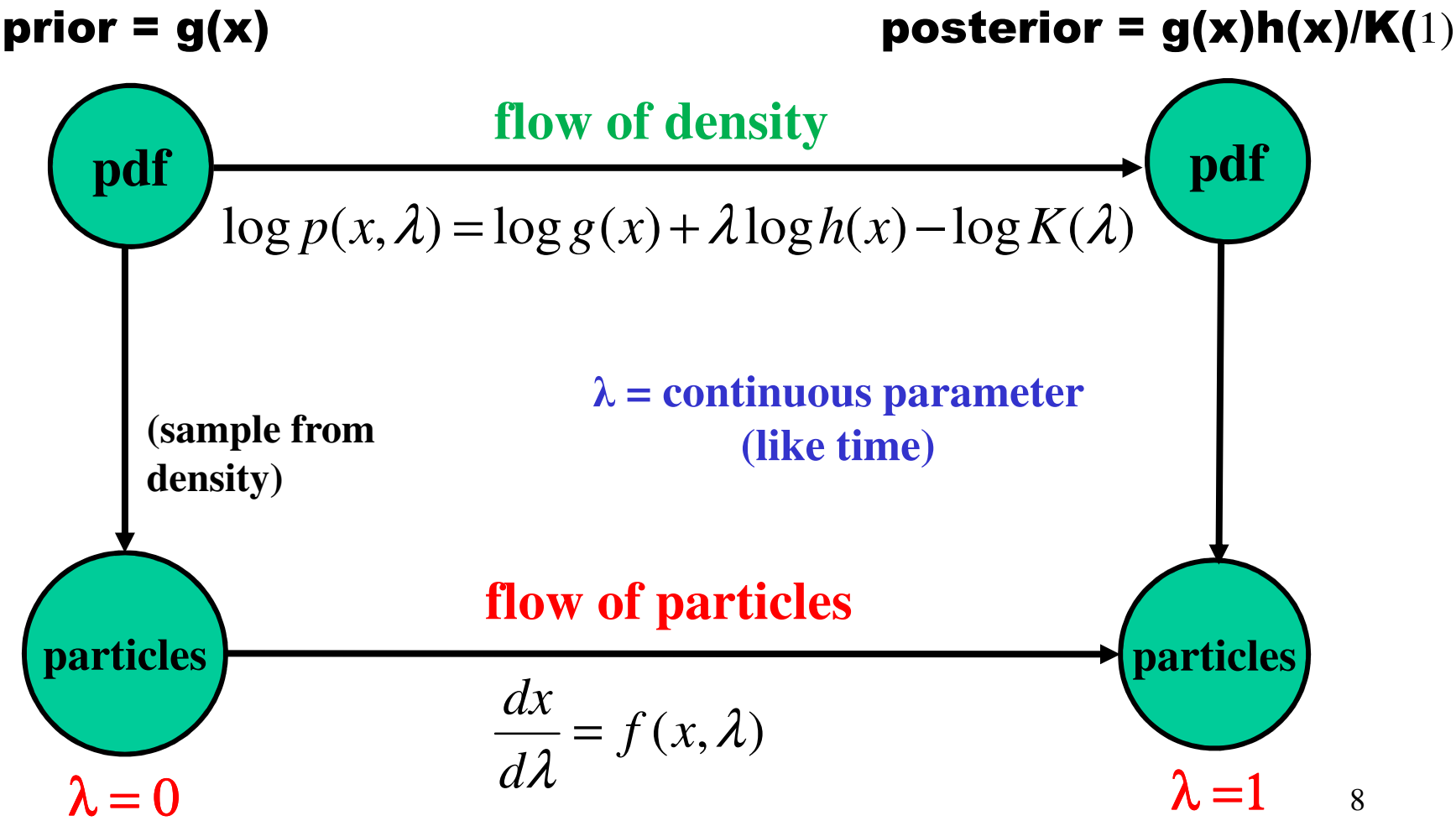


*Daum & Huang, "Particle degeneracy: root cause & solution," SPIE Proceedings 2011.

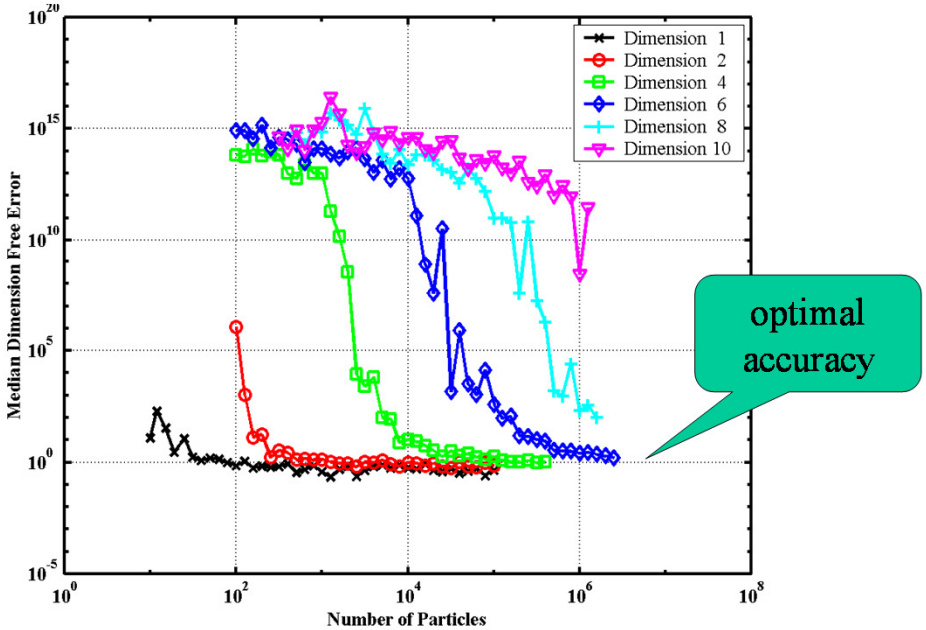
chicken & egg problem



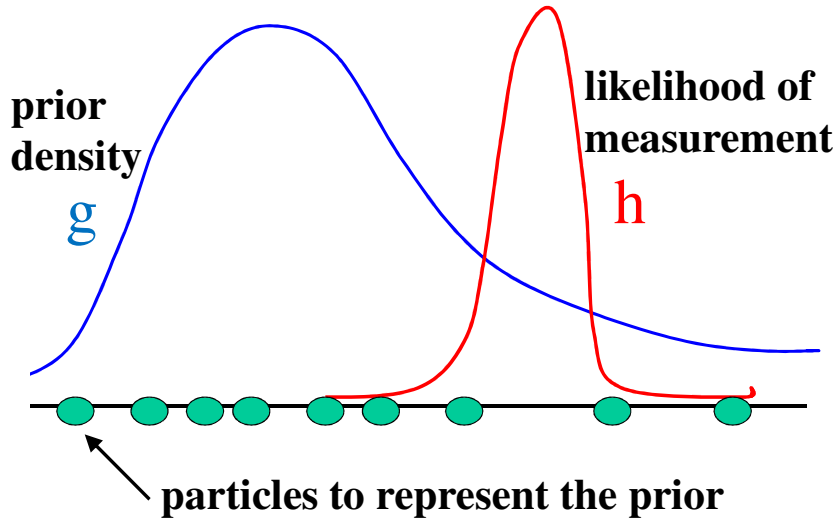
induced flow of particles for Bayes' rule

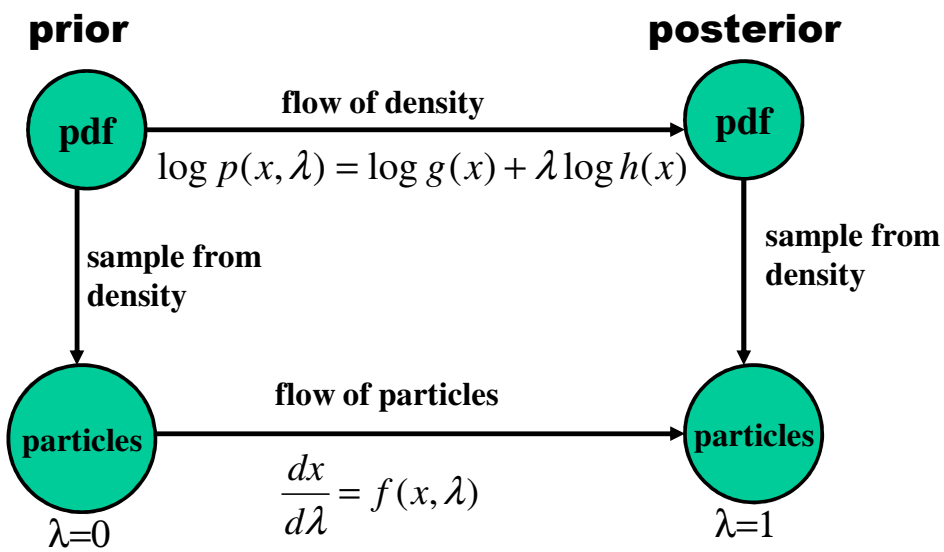


curse of dimensionality:



root cause of
curse of dimensionality:





$$div(pf) = p\left[-\log h + \frac{d\log K}{d\lambda}\right]$$

We design the particle flow by solving the above PDE for f.

$$div(pf) = p\left[-\log h + \frac{d\log K}{d\lambda}\right]$$

$$\text{let } \mathrm{q} = \mathrm{pf}$$

$$\frac{\partial \mathrm{q}_1}{\partial \mathrm{x}_1} + \frac{\partial q_2}{\partial x_2} + \ldots + \frac{\partial q_d}{\partial x_d} = \eta$$

(1) linear PDE in unknown f or q
(2) constant coefficient PDE in q
(3) first order PDE
(4) highly underdetermined PDE
(5) same as the Gauss divergence law in Maxwell's equations
(6) same as Euler's equation in fluid dynamics
(7) existence of solution if and only if volume integral of η is zero
(i.e., neutral charge density for plasma; satisfied automatically)

incompressible flow	irrotational flow	Coulomb's law flow	small curvature flow	constant curvature flows (e.g. zero)
Gaussian densities	exponential family	Knothe-Rosenblatt flow	non-zero diffusion flow	geodesic flows
Fourier transform flow	direct integration	stabilized flows	finite dimensional flow	optimal Monge-Kantorovich transports
method of characteristics	renormalization group flow for log K(λ) inspired by QFT	renormalization group flow for log g(x) inspired by QFT	renormalization group flow for logK(λ) and log g(x)	exponential family with non-zero diffusion
Gibbs sampler like flow (inspired by direct integration)	non-singular Jacobian flow (inspired by proof)	maximum entropy flow	Moser coupling flow	suboptimal Monge-Kantorovich

exact particle flow for Gaussian densities:

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

dx/dλ does not depend on K(λ), despite the fact that the PDE does!

$$\log(h) - \frac{d\log K(\lambda)}{d\lambda} = -div(f) - \frac{\partial \log p}{\partial x} f$$

for g & h Gaussian, we can solve for f exactly:

$$f = Ax + b$$

$$A = -\frac{1}{2} PH^T \left[\lambda HPH^T + R\right]^{-1} H$$

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

automatically stable under very mild conditions & extremely fast

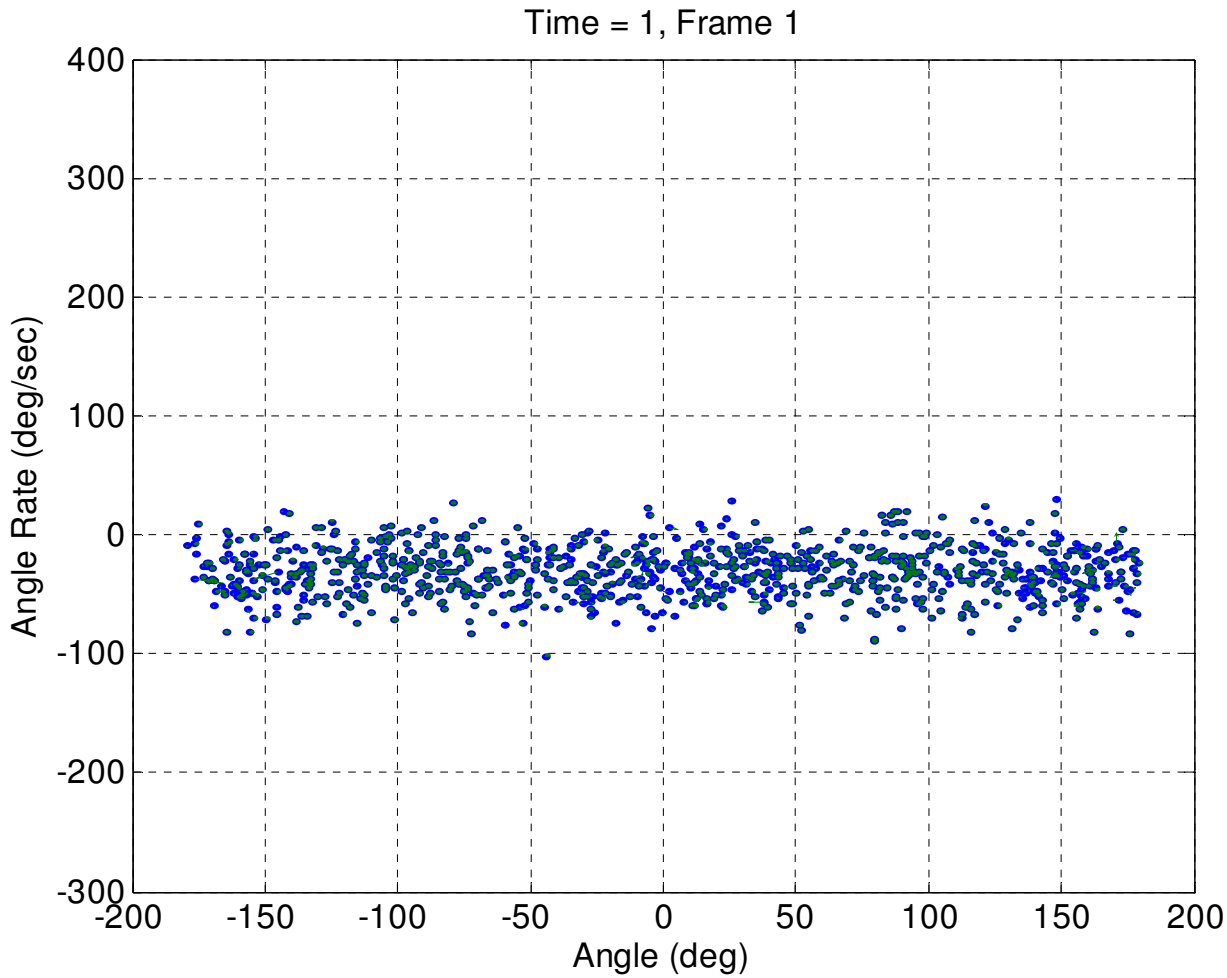
incompressible particle flow

$$\frac{\mathrm{dx}}{\mathrm{d}\lambda} = \begin{cases} -\log(h(x)) \dfrac{\left[\dfrac{\partial \log p(x,\lambda)}{\partial x}\right]^T}{\left\|\dfrac{\partial \log p(x,\lambda)}{\partial x}\right\|^2} & \text{for non - zero gradient} \\ 0 \quad \text{otherwise} & \end{cases}$$

for $d \geq 2$

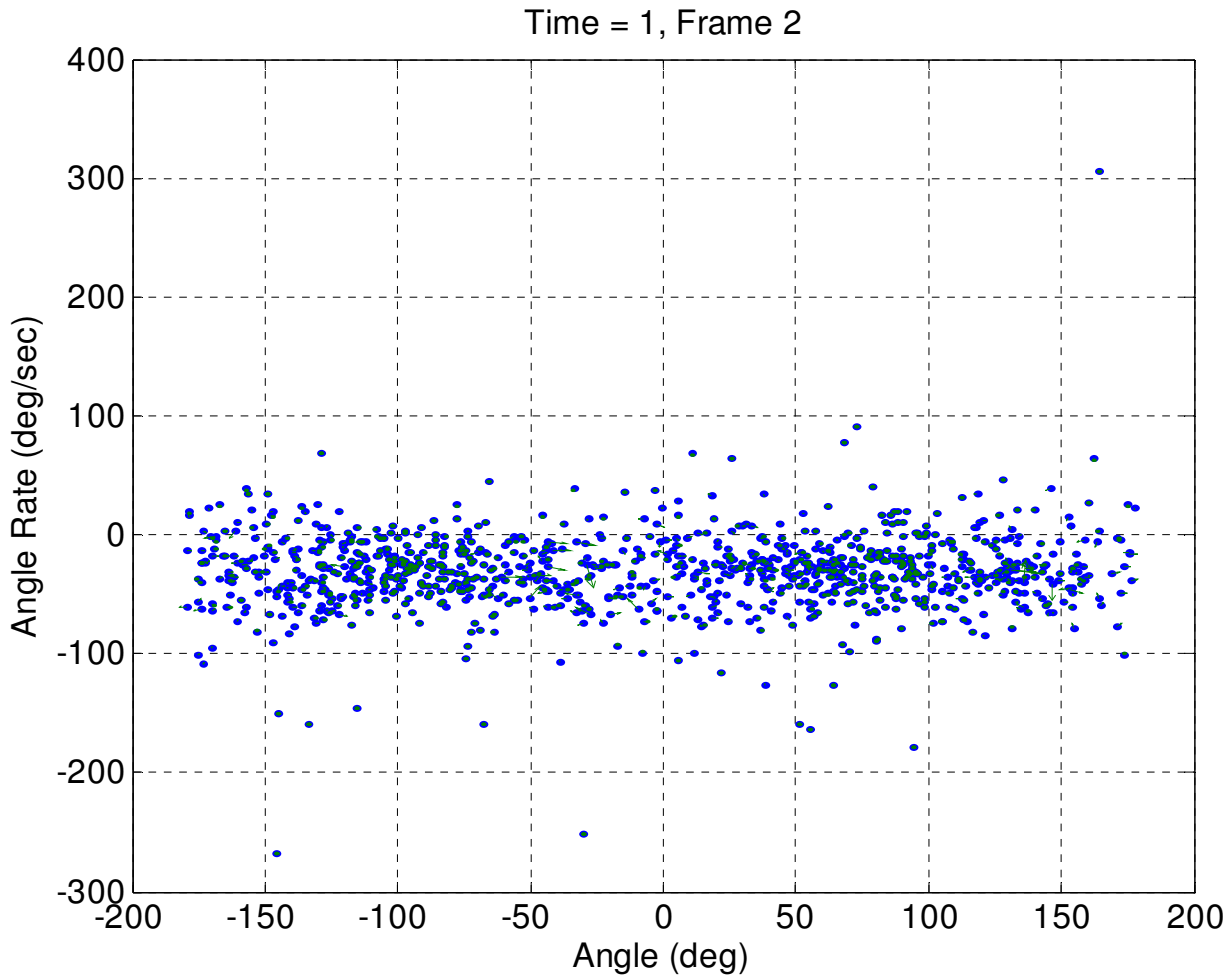
dx/dλ does not depend on K(λ), despite the fact that the PDE does!

initial probability distribution of particles:



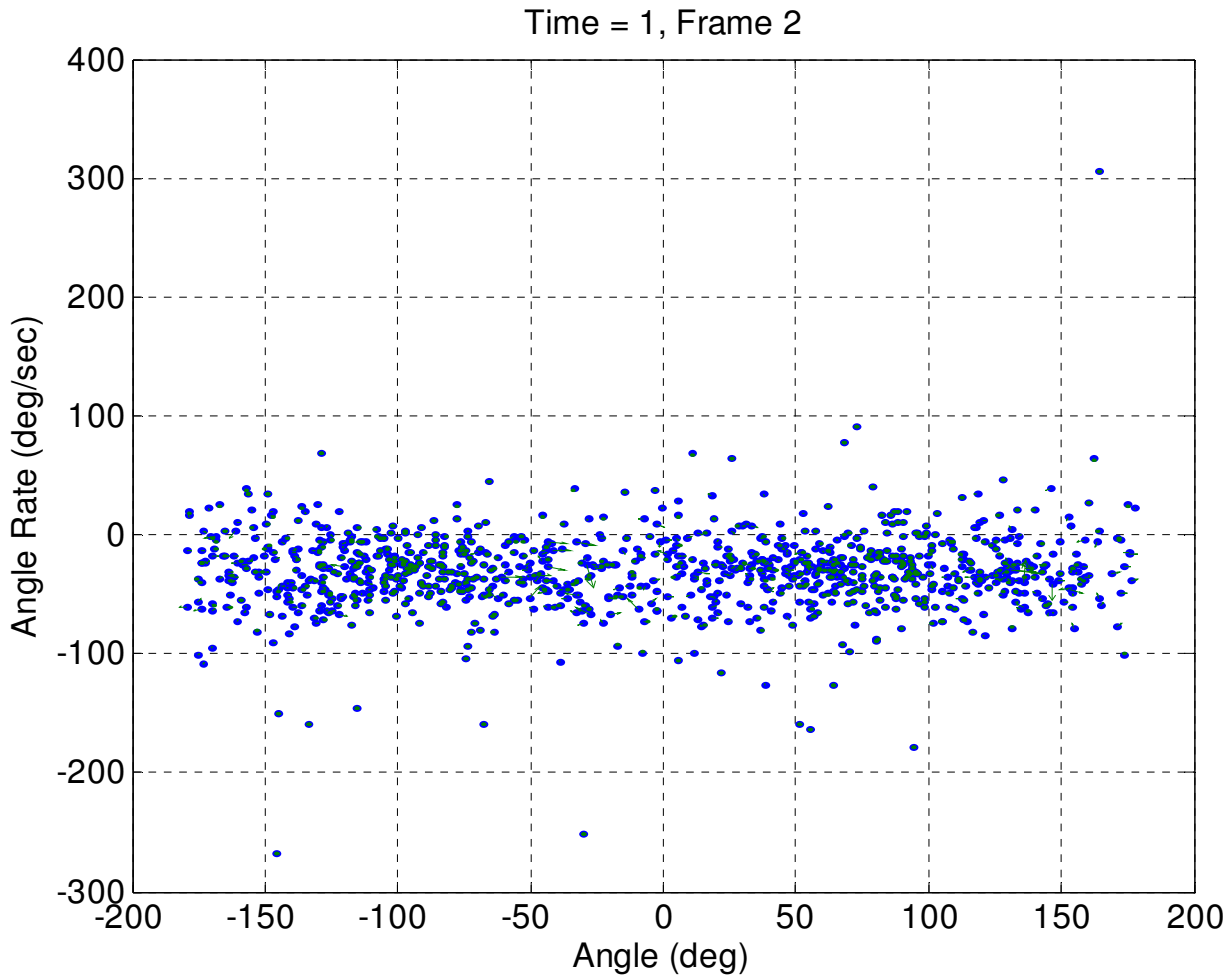
$\lambda = 0.0$

flow of particles (for one noisy measurement of sin(θ) with Bayes' rule):



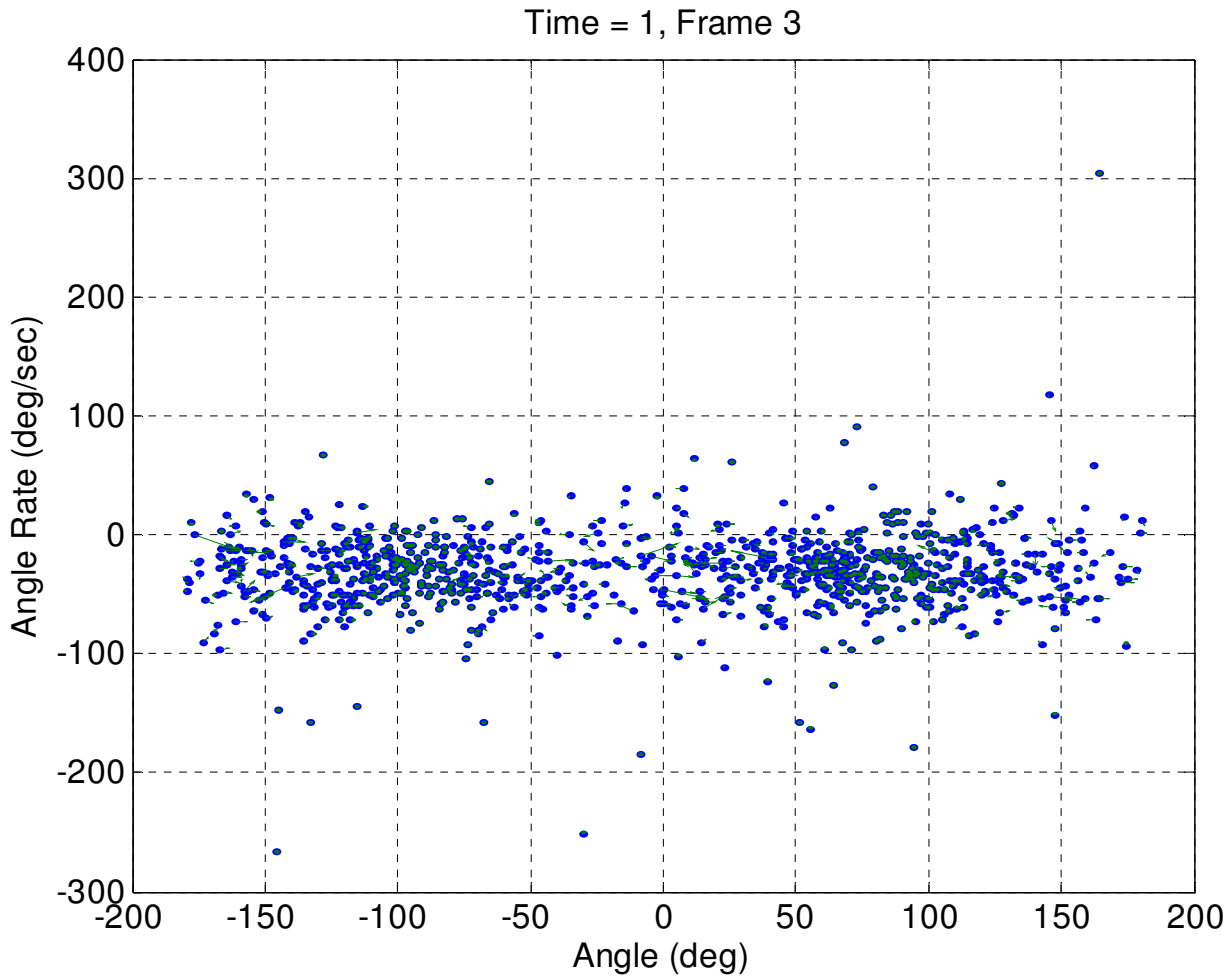
$\lambda = 0.1$

flow of particles (for one noisy measurement of sin(θ) with Bayes' rule):



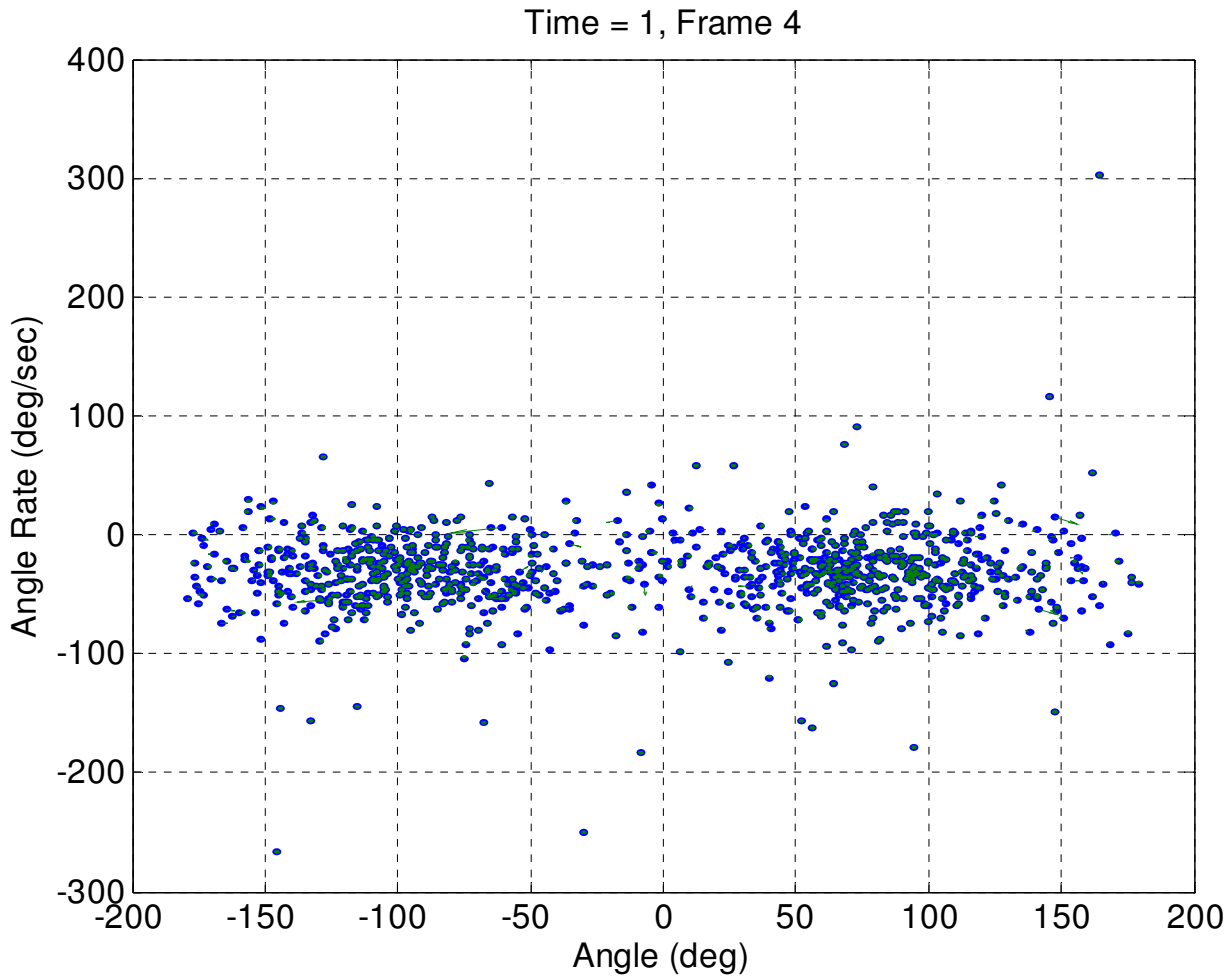
$\lambda = 0.2$

flow of particles (for one noisy measurement of sin(θ) with Bayes' rule):



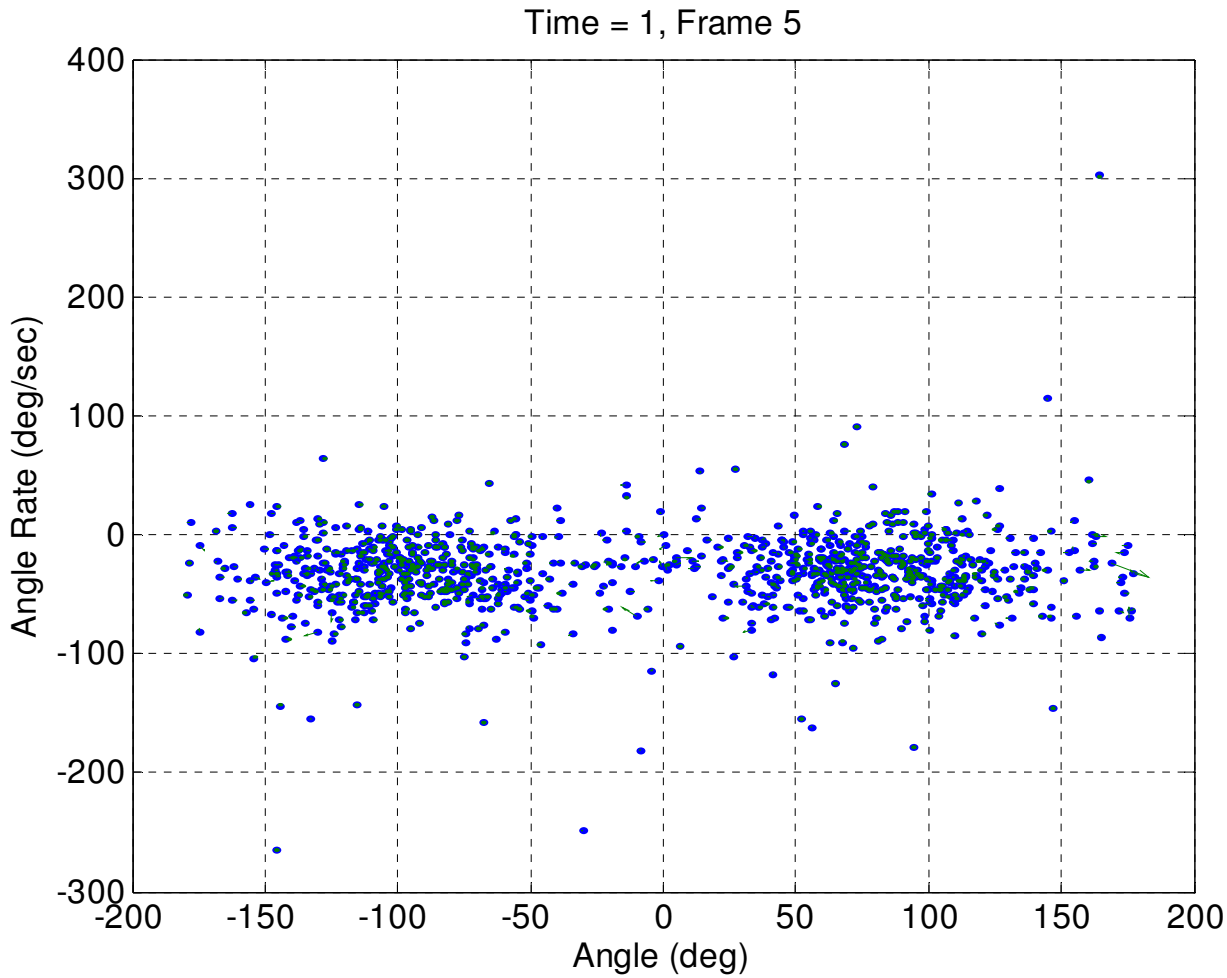
$\lambda = 0.3$

flow of particles (for one noisy measurement of sin(θ) with Bayes' rule):



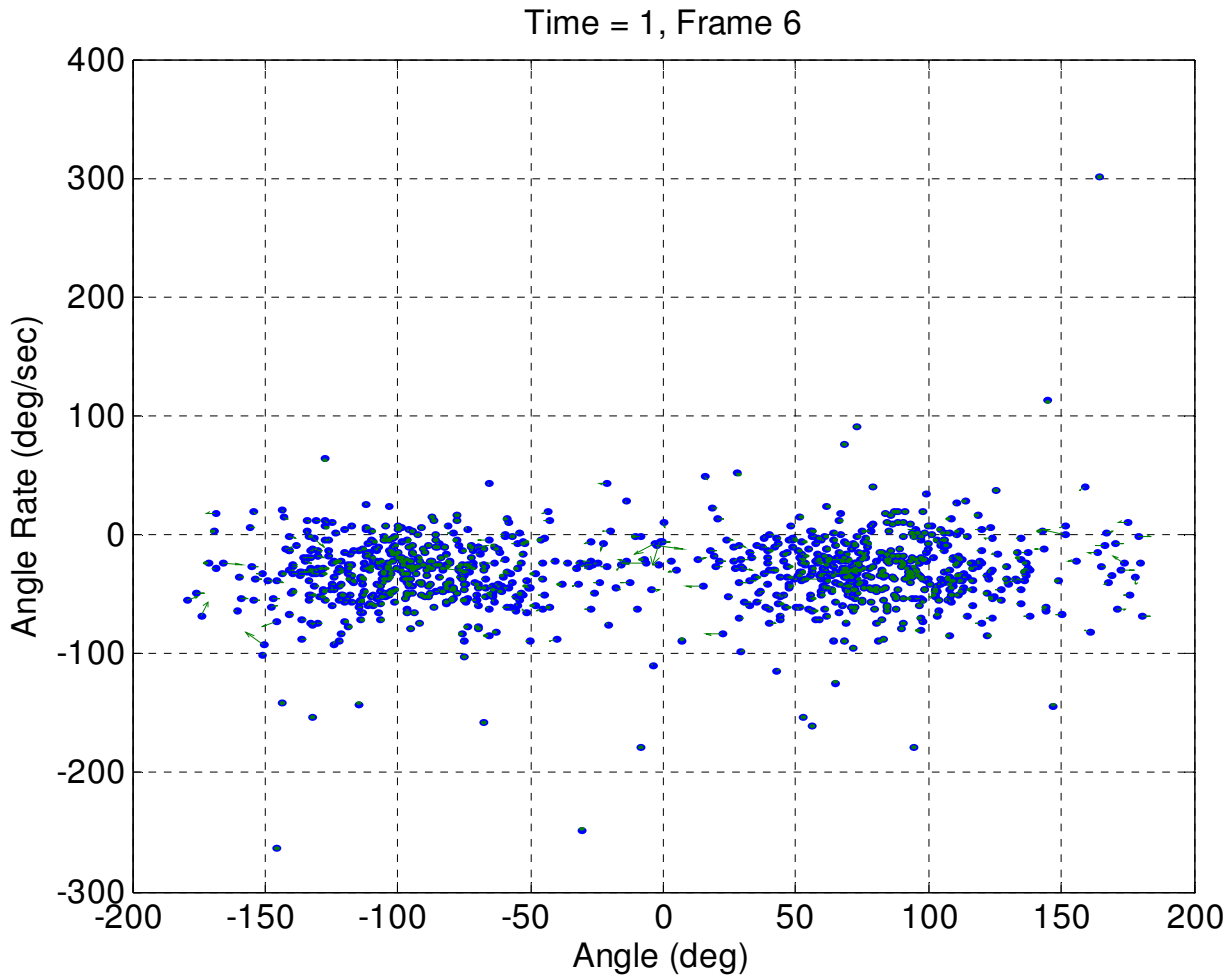
$\lambda = 0.4$

flow of particles (for one noisy measurement of sin(θ) with Bayes' rule):



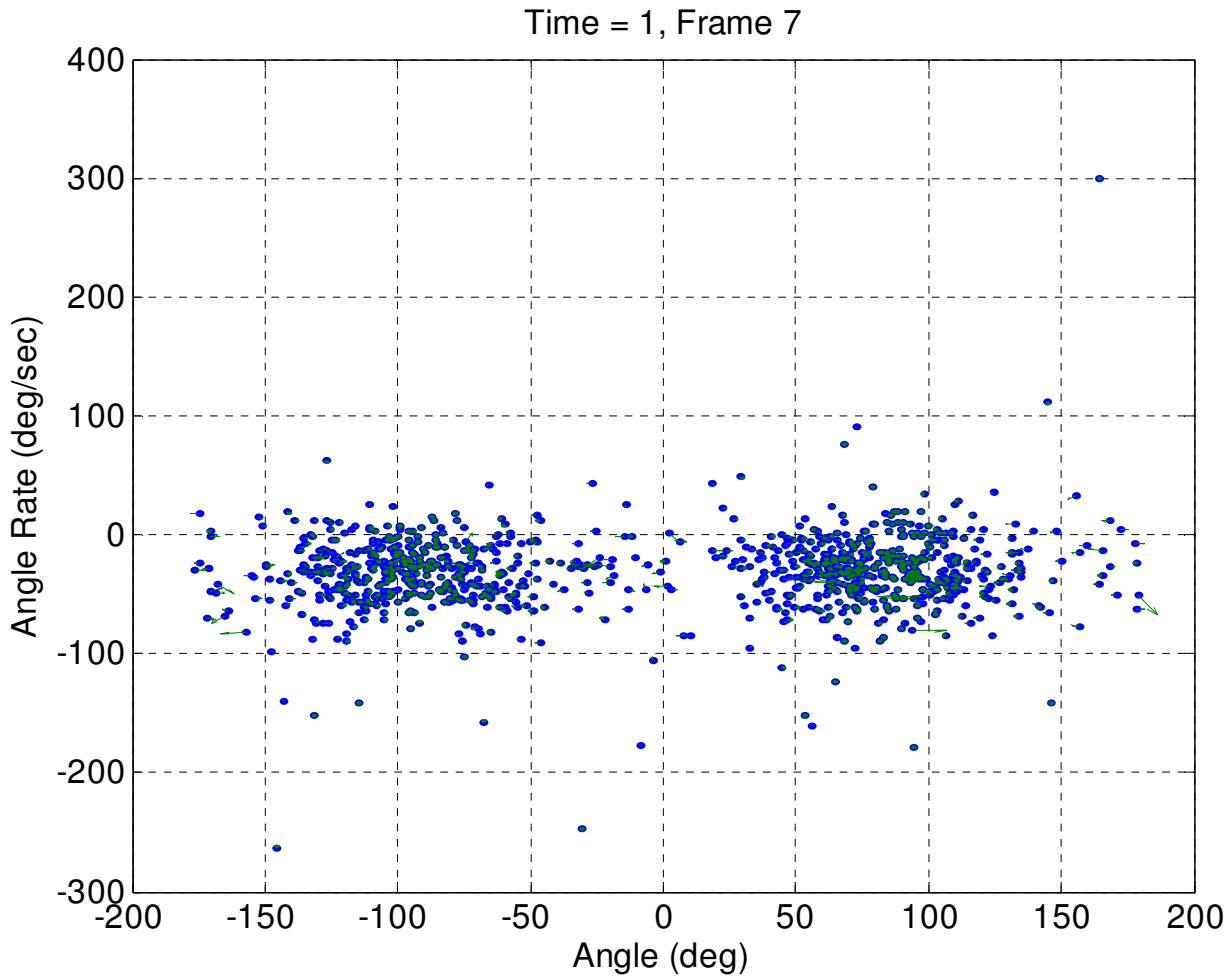
$\lambda = 0.5$

flow of particles (for one noisy measurement of sin(θ) with Bayes' rule):



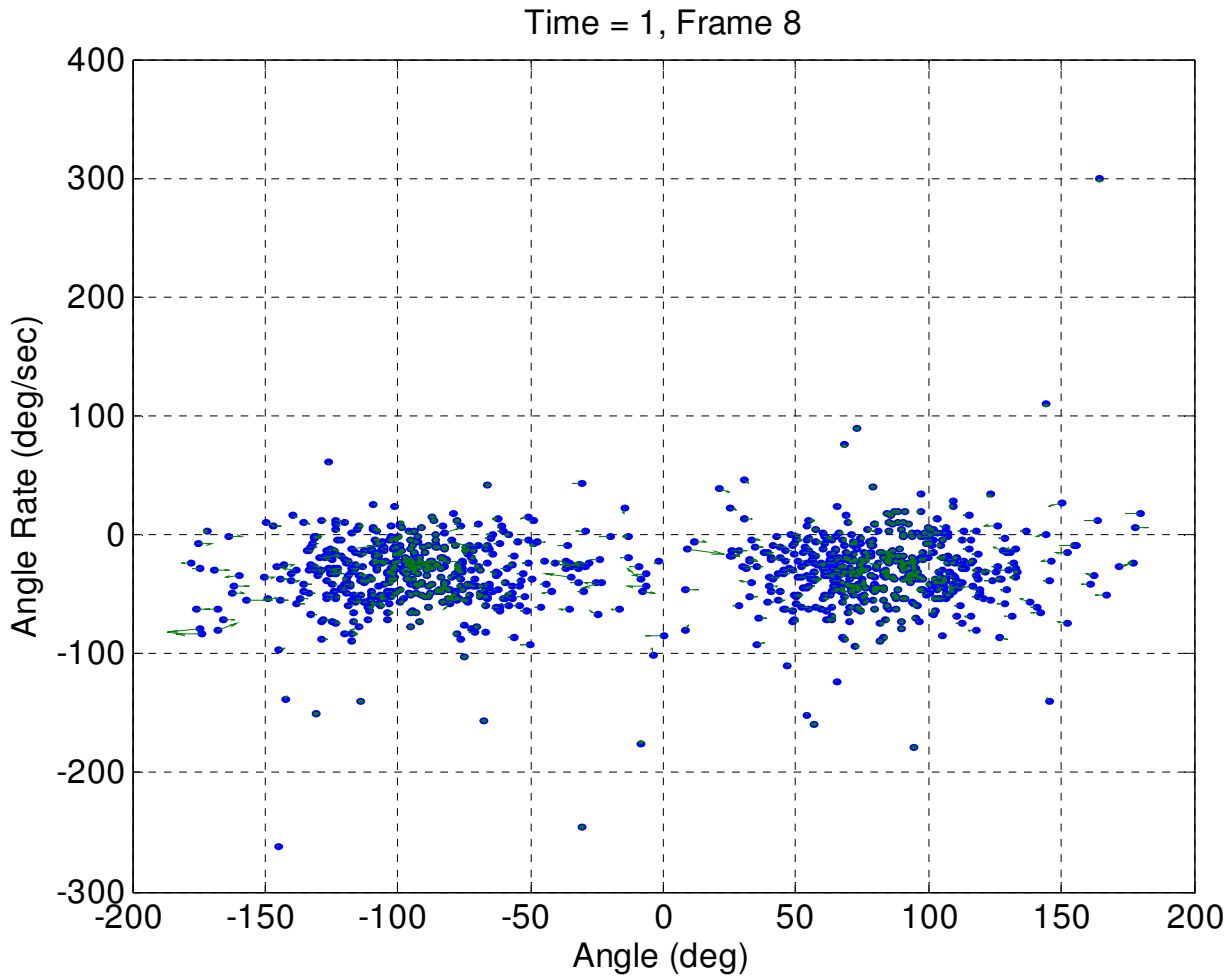
$\lambda = 0.6$

flow of particles (for one noisy measurement of sin(θ) with Bayes' rule):



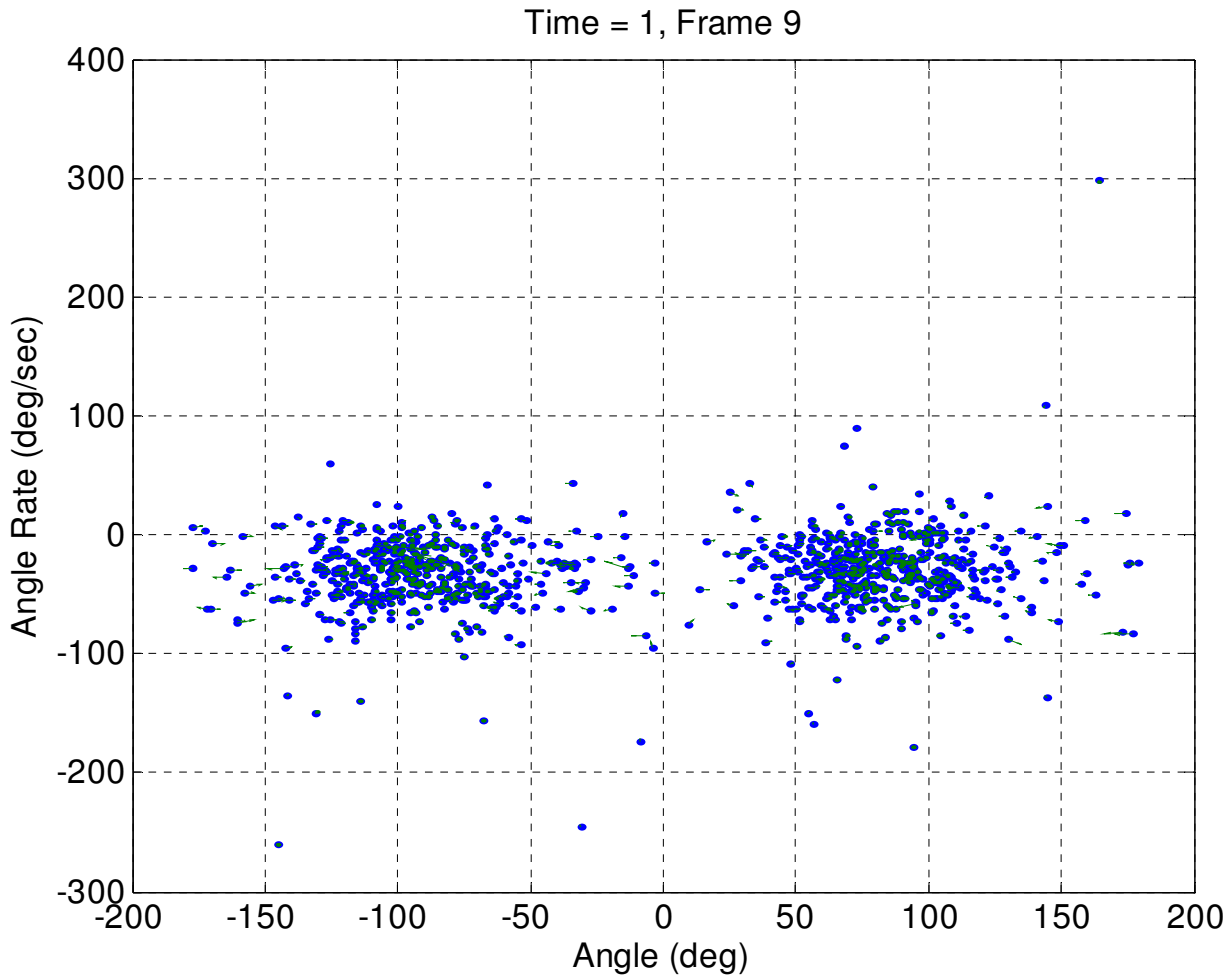
$\lambda = 0.7$

flow of particles (for one noisy measurement of sin(θ) with Bayes' rule):



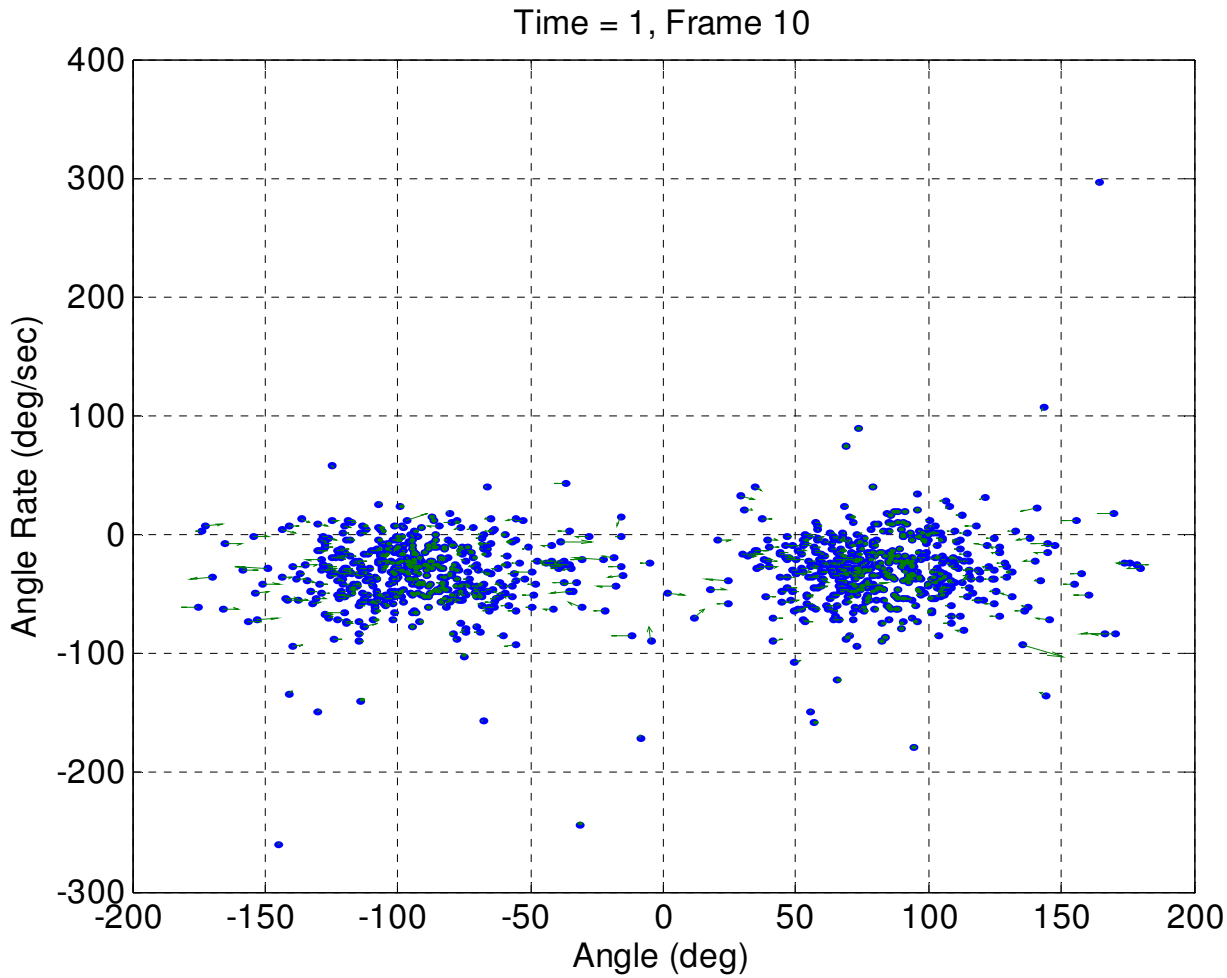
$\lambda = 0.8$

flow of particles (for one noisy measurement of sin(θ) with Bayes' rule):



$\lambda = 0.9$

final probability distribution of particles (resulting from one noisy measurement of sin(θ) with Bayes' rule):



$\lambda = 1$

new particle flow* :

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

dx/dλ does not depend on K(λ), despite the fact that the PDE does!

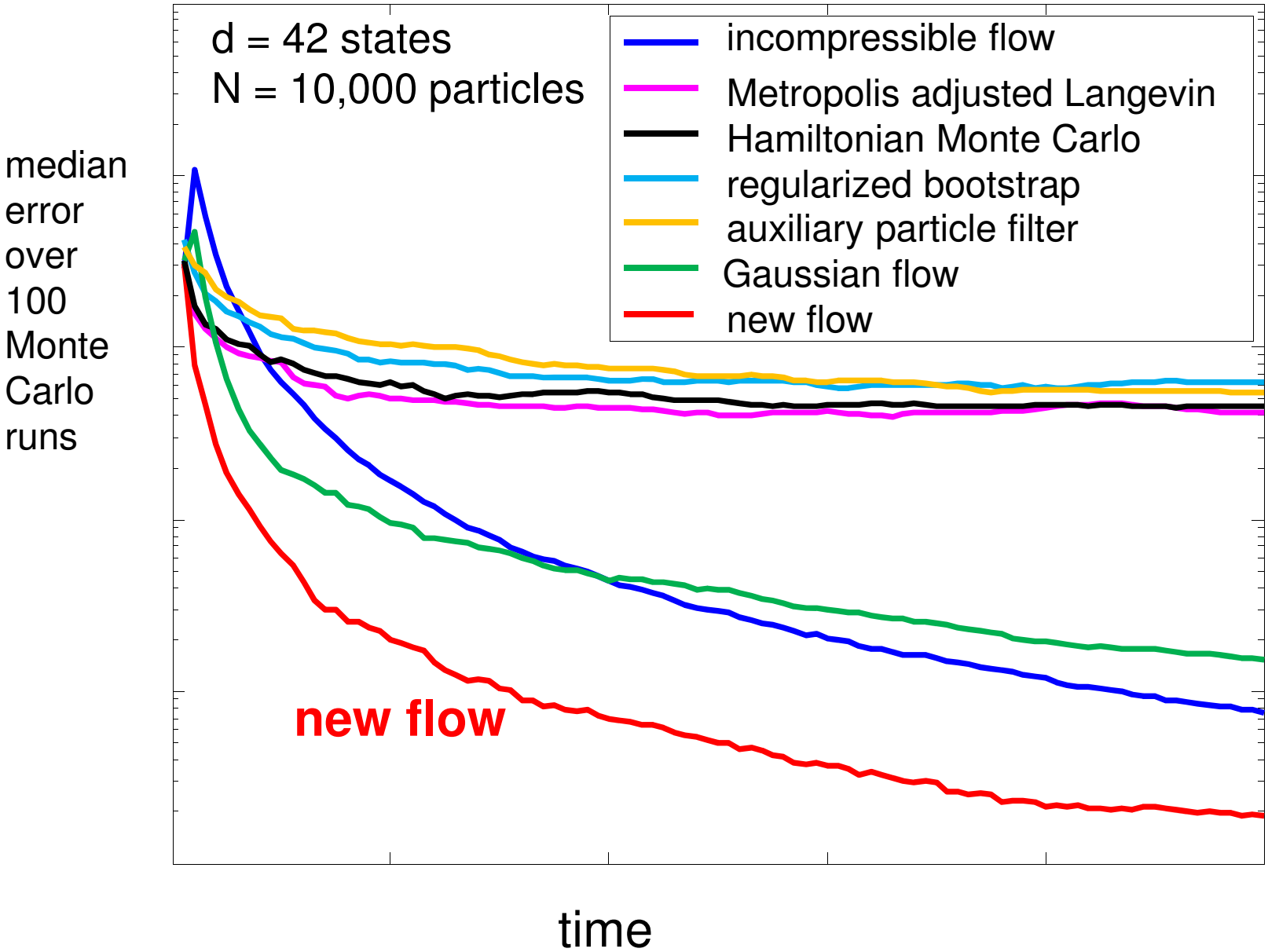
If we approximate the density p as Gaussian, then the observed Fisher information matrix can be computed using the sample covariance matrix (C) over the set of particles:

$$\frac{dx}{d\lambda} \approx C\left(\frac{\partial \log h}{\partial x}\right)^T$$

for Gaussian densities we get the EKF for each particle:

$$\frac{dx}{d\lambda} \approx C\left(\frac{\partial \theta(x)}{\partial x}\right)^T R^{-1}\left(z - \theta(x)\right)$$

*Daum & Huang, "particle flow with non-zero diffusion for nonlinear filters," SPIE conference proceedings, San Diego, August 2013.


d = 42 states
N = 10,000 particles
incompressible flow
Metropolis adjusted Langevin
Hamiltonian Monte Carlo
regularized bootstrap
auxiliary particle filter
Gaussian flow
new flow
new flow
median error over 100 Monte Carlo runs
time

new particle flow :

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

If we approximate the density g(x) from the exponential family, then the flow is:

$$\frac{dx}{d\lambda} \approx -\left[\frac{\partial^2 \log q(x)}{\partial x^2} + \psi^T \frac{\partial^2 \theta(x)}{\partial x^2} + \lambda \frac{\partial^2 \log h}{\partial x^2}\right]^{-1}\left(\frac{\partial \log h}{\partial x}\right)^T$$

in which the unnormalized prior density is:

$$g(x) \approx q(x)\exp\left[\psi^T \theta(x)\right]$$

BIG DIG (17 million cubic yards of dirt, one million truckloads & $24 billion)*

*Daum & Huang, “particle flow & Monge-Kantorovich transport,” proceedings of FUSION conference, Singapore, July 2012.

item	particle flow	transport
1. purpose	fix particle degeneracy caused by Bayes' rule	move physical objects from one probability density to another*
2. optimality criterion	NONE!	minimize convex functional
3. how to pick a unique solution	many different methods (e.g. min L^2 norm, given form, most stable flow, min convex function, more smoothness, max entropy, arbitrary, random, etc.)	minimize convex functional (e.g., dirt mover's metric or Wasserstein metric)
4. computational complexity	numerical integration of ODE for each particle	solution of PDE (Poisson's, Monge-Ampere, HJB, etc.)
5. high dimensional problems	d = 1 to 42	d = 1, 2, 3
6. solution of nice special cases	Gaussian, incompressible, irrotational, geodesic, exponential family, etc.	Moser coupling (1990), Brenier (1991-2011), Knothe-Rosenblatt (1952), etc.
7. homotopy of densities	log-homotopy	homotopy
8. stability of flow considered (e.g., Lyapunov stability)	often explicitly designed into algorithm	rarely
9. existence of flows proved	adapt proofs from transport theory	Shnirelman irrotational flow d > 2, Moser & Dacorogna d >1, Brenier any d, et al.
10. conservation of probability mass along the flow	yes	yes
11. avoiding normalization of probability density is crucial for practical algorithms	yes	no
12. explicitly compute normalized densities	no, log of unnormalized densities	yes
13. stiff ODEs mitigated	yes	no
14. stochastic flows	rarely	rarely

superb books on transport theory

Cédric Villani, "Topics in optimal transportation," AMS Press 2003.

Very clear & accessible introduction; wonderful book!

Cédric Villani, "Optimal transport: old & new," Springer-Verlag 2009. More detailed & rigorous math; free on internet!

new nonlinear filter: particle flow

new particle flow filter	standard particle filters
many orders of magnitude faster than standard particle filters	suffers from curse of dimensionality due to particle degeneracy
3 to 4 orders of magnitude faster code per particle for any $d \geq 3$ problems	requires resampling using a proposal density
3 to 4 orders of magnitude fewer particles required to achieve optimal accuracy for $d \geq 6$ problems	requires millions or billions of particles for high dimensional problems
Bayes' rule is computed using particle flow (like physics)	Bayes' rule is computed using a pointwise multiplication of two functions
no proposal density	depends on proposal density (e.g., Gaussian from EKF or UKF or other)
no resampling of particles	resampling is needed to repair the damage done by Bayes' rule
embarrassingly parallelizable	suffers from bottleneck due to resampling
computes log of unnormalized density	suffers from severe numerical problems due to computation of normalized density

history of mathematics

1. creation of the integers

2. invention of counting

3. invention of addition as a fast method of counting

4. invention of multiplication as a fast method of addition

5. invention of particle flow as a fast method of multiplication*

BACKUP

REFERENCES:

(1) Fred Daum, "Nonlinear filters: beyond the Kalman filter," IEEE Aerospace & Electronic Systems Magazine special tutorial, pages 57-69, August 2005.

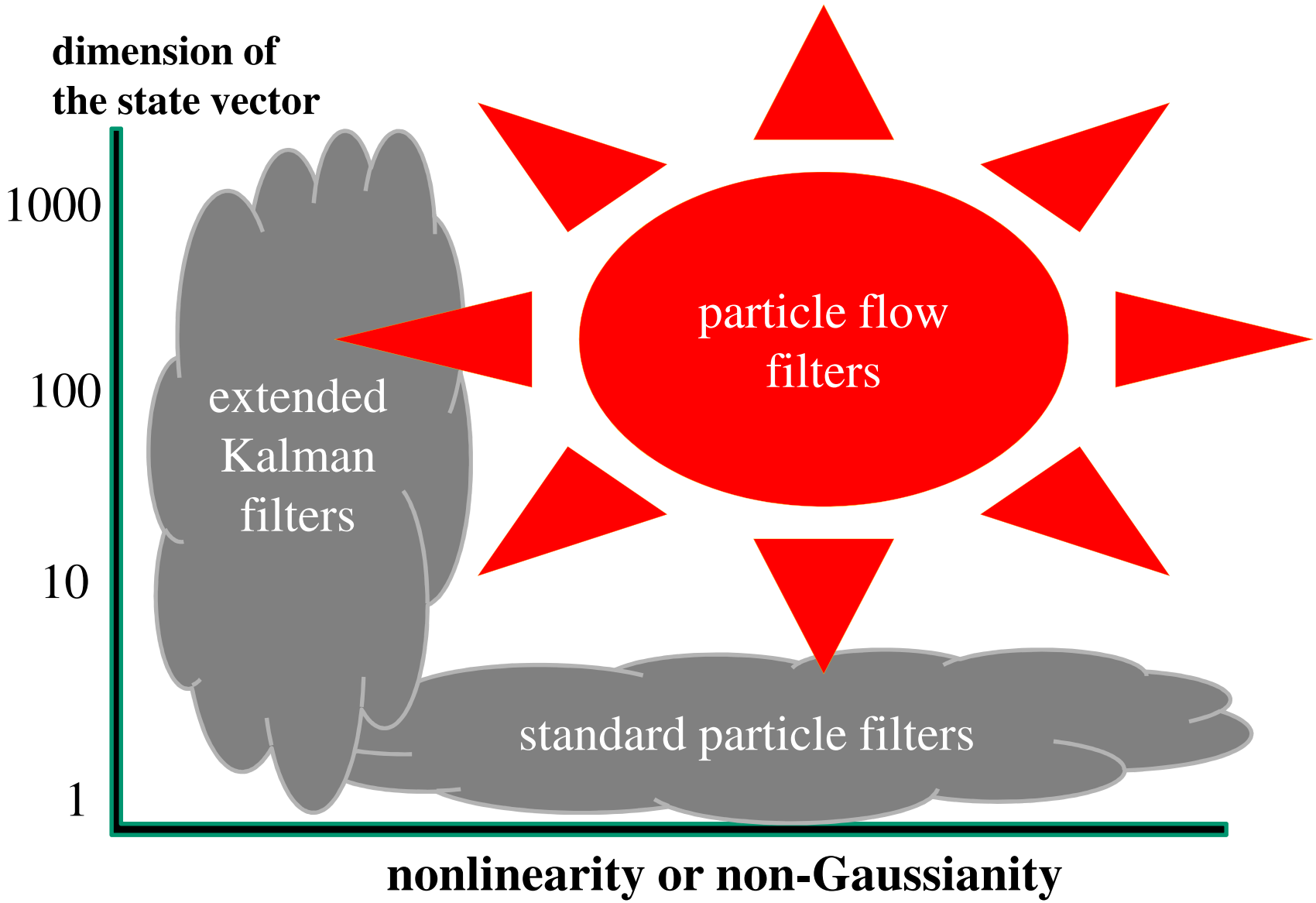
(2) Fred Daum and Jim Huang, "Particle flow with non-zero diffusion for nonlinear filters," Proceedings of SPIE conference, San Diego, August 2013.

(3) Arnaud Doucet and A. M. Johansen, "A tutorial on particle filtering and smoothing: fifteen years later", in "The Oxford handbook of nonlinear filtering," edited by Dan Crisan and Boris Rozovskii, pages 656 to 704, 2011.

(4) Fred Daum and Jim Huang, "Particle flow and Monge-Kantorovich transport," Proceedings of IEEE FUSION Conference, Singapore, July 2012.

(5) Fred Daum & Jim Huang, “How to avoid normalization of particle flow for nonlinear filters, Bayesian decisions and transport,” Proceedings of SPIE conference, Baltimore, May 2014.

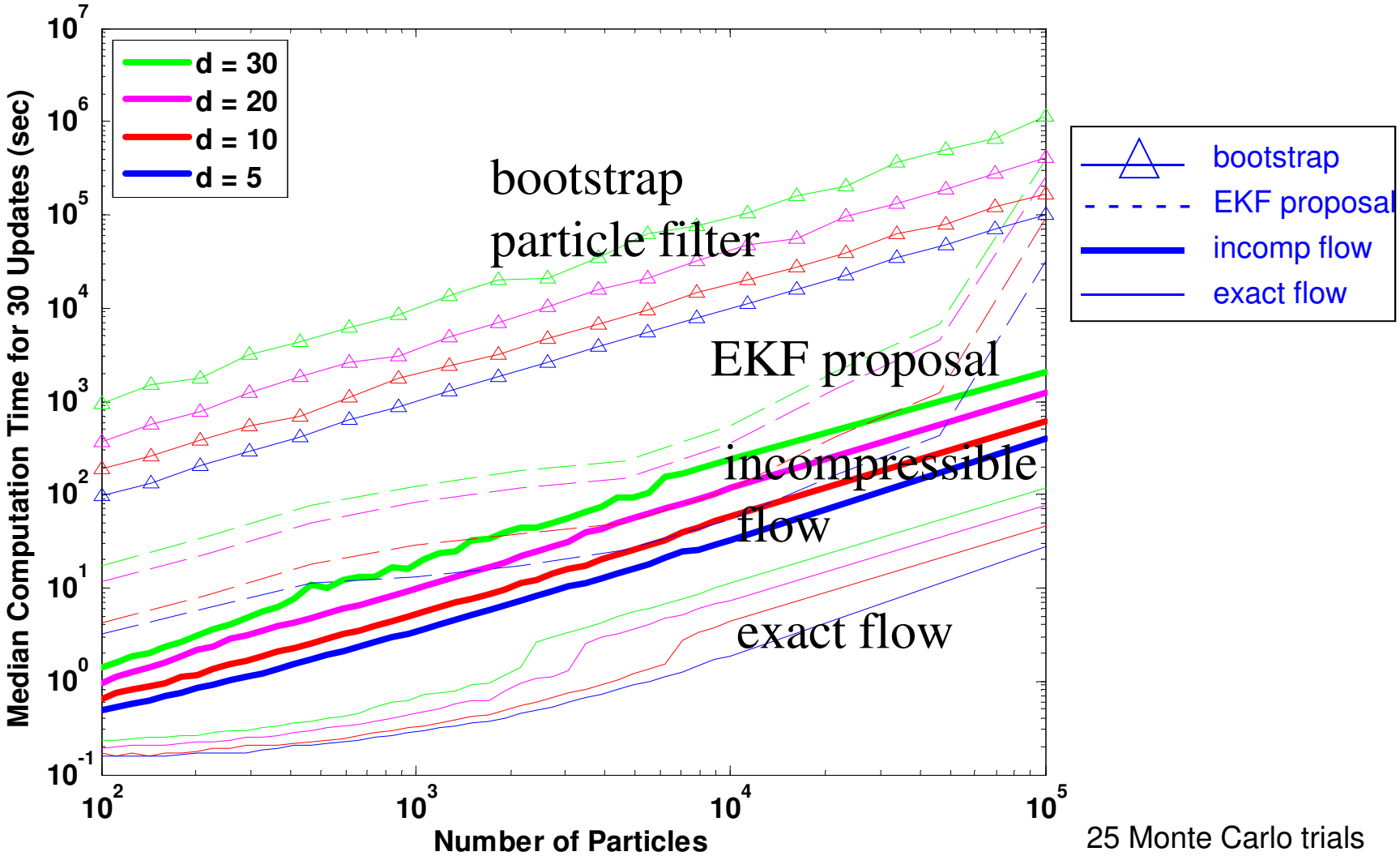
university or company	researchers	topic	papers
Connecticut	Peter Willett & Sora Choi	numerical experiments	2011
McGill	Mark Coates & Ding	numerical experiments	2012
New Orleans	Jilkov & Wu & Chen	GPUs numerical experiments	2013
Melbourne	Mark Morelande	generalization of theory	2011
Goteborg	Svensson, et al.	generalization of theory	2011
Scientific Systems	Lingji Chen & Raman Mehra	analysis of singularities in incompressible flow for d = 1	2010
Mitsubishi	Grover & Sato	theory & numerical experiments	2012
Cambridge	Peter Bunch & Simon Godsill	theory & numerical experiments	2014
Liverpool	Simon Maskell & Flávio De Melo	relation to MCMC & numerical experiments	2014
London	Simon Julier	numerical experiments	---
METRON	Kristine Bell & Larry Stone	numerical experiments & theory	2014
Lockheed Martin	Nima Moshtagh & Moses Chan	numerical experiments	---
STR	Shozo Mori	theory & numerical experiments	---
BU	Castanon, et al.	numerical experiments	---
Toulouse	Marcelo Pereyra	numerical experiments & theory	---
Bonn	Martin Ulmke & Ahmed	numerical experiments	---
Istanbul	Serdar Aslan	numerical experiments	---
Wuhan	Lanlan Pang	numerical experiments	---
Tufts & Hartford	Umarov & Nelson	fractional Brownian motion	2014
Raytheon Boston	Daum & Huang & Noushin	theory & numerical experiments	2007-2014
Raytheon Arizona	Frankot & Reid & Kyle	theory & numerical experiments	---
Raytheon California	Ploplys & Casey	theory & numerical experiments	---


dimension of
the state vector
1000
100
10
1
extended
Kalman
filters
particle flow
filters
standard particle filters
nonlinearity or non-Gaussianity

many applications of particle flow

Bayesian decisions & learning	robotics	communica-tions	control of chemical, mechanical, electrical & nuclear plants
guidance & navigation	tracking	weather & climate prediction	predicting ionosphere, thermosphere, troposphere
science	imaging	medicine (e.g., MRI, surgical planning, drug design, diagnosis)	transport
oil & mineral exploration	financial engineering	adaptive antennas	audio & video signal processing
nonlinear filtering & smoothing	multi-sensor data fusion	compressive sensing	CRYPTO

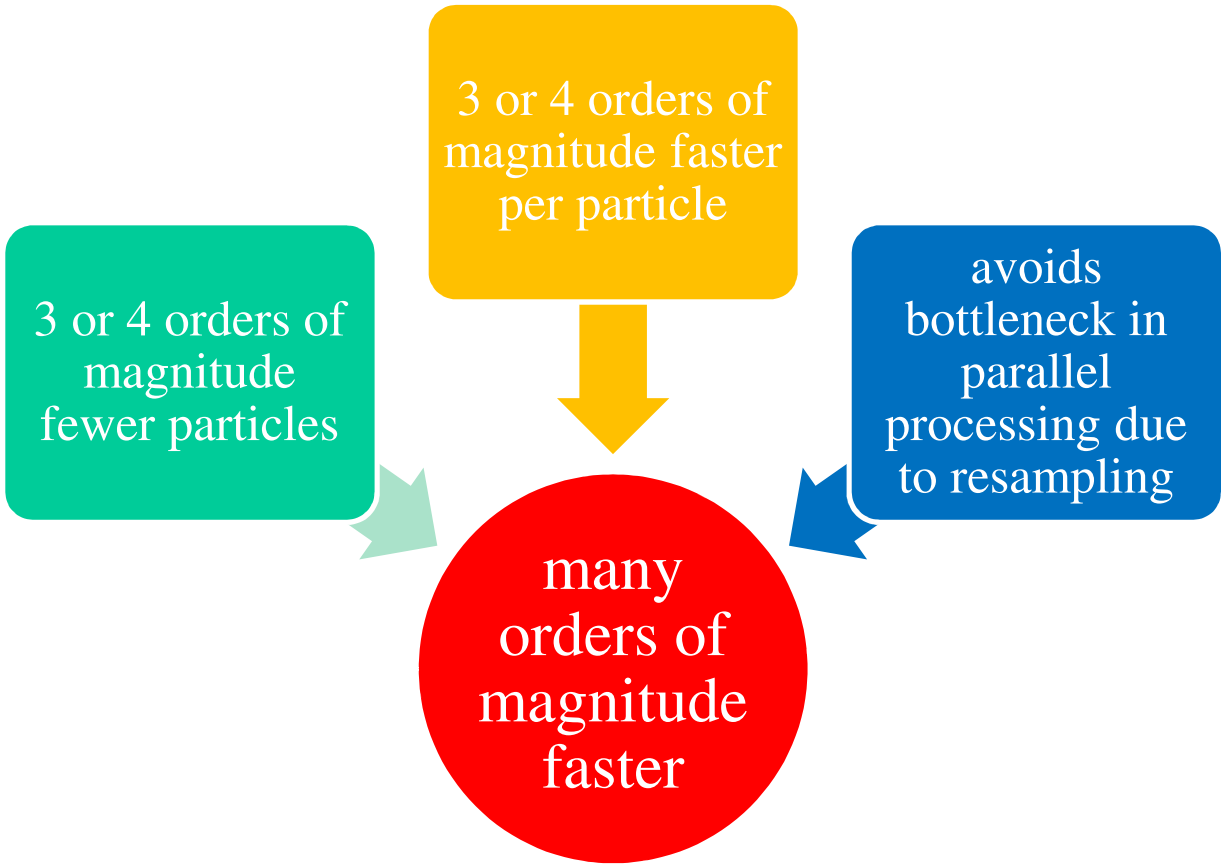
exact flow filter is many orders of magnitude faster per particle than standard particle filters



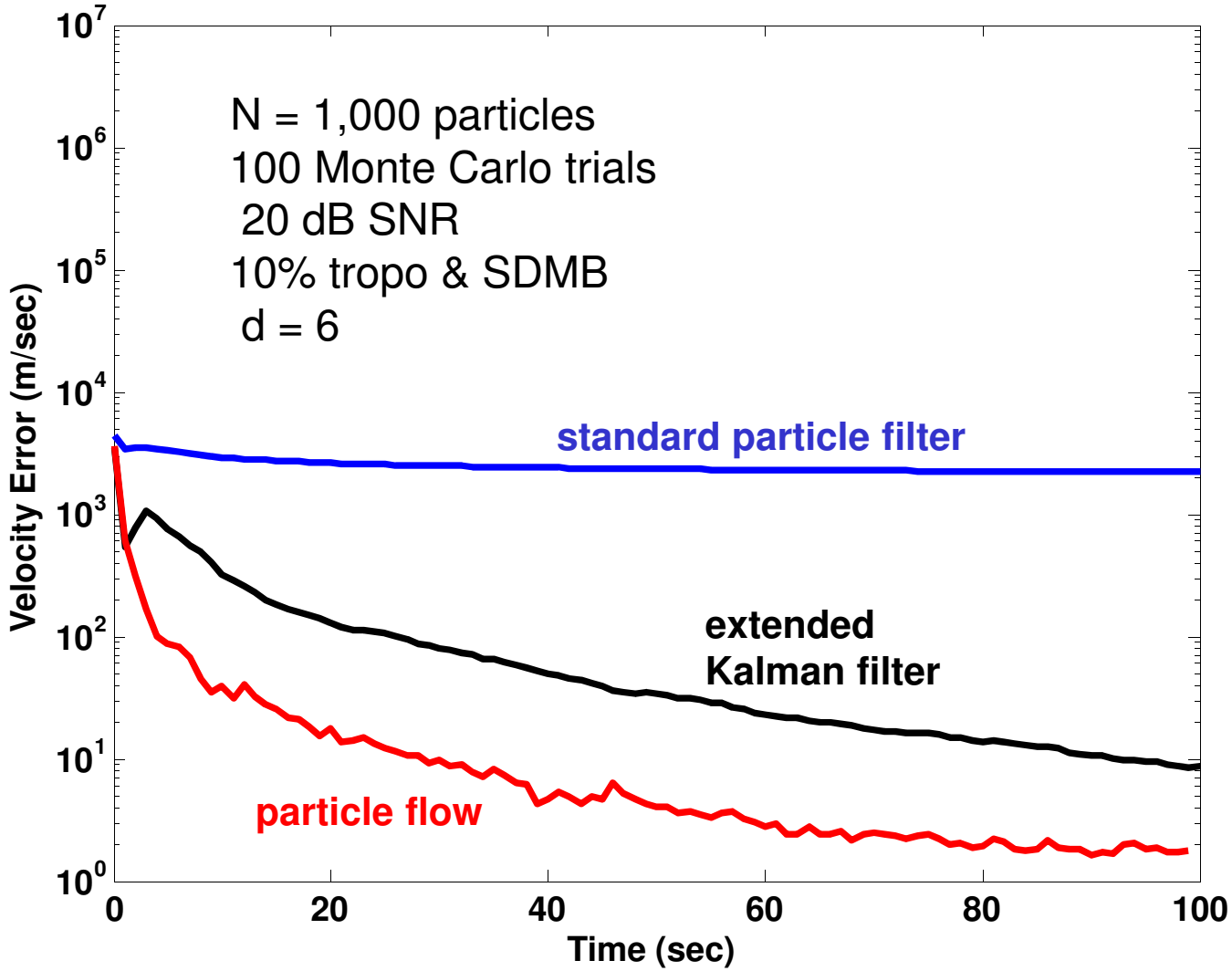
25 Monte Carlo trials

* Intel Corel 2 CPU, 1.86GHz, 0.98GB of RAM, PC-MATLAB version 7.7

particle flow filter is many orders of magnitude faster real time computation (for the same or better estimation accuracy)



comparison of estimation accuracy for three filters:



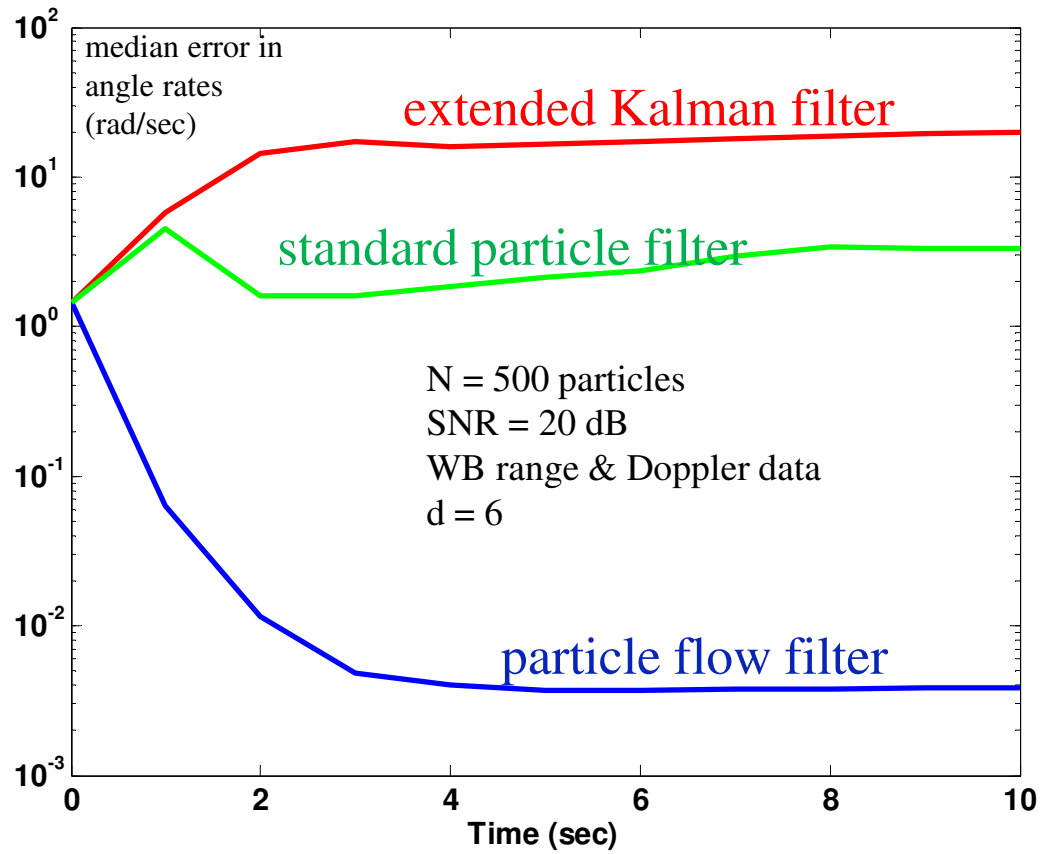
new filter improves angle rate estimation accuracy by two or three orders of magnitude

highly nonlinear dynamics:

$$I_1\dot{\omega}_1 + (I_3 - I_2)\omega_3\omega_2 = M_1$$

$$I_2\dot{\omega}_2 + (I_1 - I_3)\omega_1\omega_3 = M_2$$

$$I_3\dot{\omega}_3 + (I_2 - I_1)\omega_2\omega_1 = M_3$$



extended Kalman filter diverges because it cannot model multimodal conditional probability densities accurately

derivation of PDE for exact particle flow:

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

$$\frac{\partial p(x,\lambda)}{\partial \lambda} = -Tr\left[\frac{\partial (pf)}{\partial x}\right]$$

$$\frac{\partial \log p(x,\lambda)}{\partial \lambda} p(x,\lambda) = -Tr\left[\frac{\partial (pf)}{\partial x}\right]$$

definition of log p(x,λ)

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

$$\left[\log h(x) - \frac{\partial \log K(\lambda)}{\partial \lambda}\right] p(x,\lambda) = -div(pf)$$

$$div(q) = \eta$$

definition of η

$$\eta = -p(x,\lambda)\left[\log h(x) - \frac{\partial \log K(\lambda)}{\partial \lambda}\right]$$

Fokker-Planck equation*

$$\frac{\partial p}{\partial t} = -\frac{\partial p}{\partial x} f - Tr\left(\frac{\partial f}{\partial x}\right) p + \frac{1}{2} Tr\left(\frac{\partial^2 p}{\partial x^2} Q\right)$$

$$p = p(x,t)$$

$$p(x,t) = \text{probability density of x at time t}$$

$$\frac{\mathrm{dx}}{\mathrm{dt}} = f(x,t) + \frac{dw}{dt}$$

$$Q = \text{covariance matrix of white noise w(t)}$$

*Andrew Jazwinski, "Stochastic processes and filtering theory," Dover Books Inc., 1998

Bayes' rule:

$$p(x,t_k|Z_k) = \frac{p(x,t_k|Z_{k-1})\,p(z_k|x,t_k)}{p(z_k|Z_{k-1})}$$

$p(x,t_k|Z_k) =$ probability density of x at time t_k given Z_k

$x =$ state vector

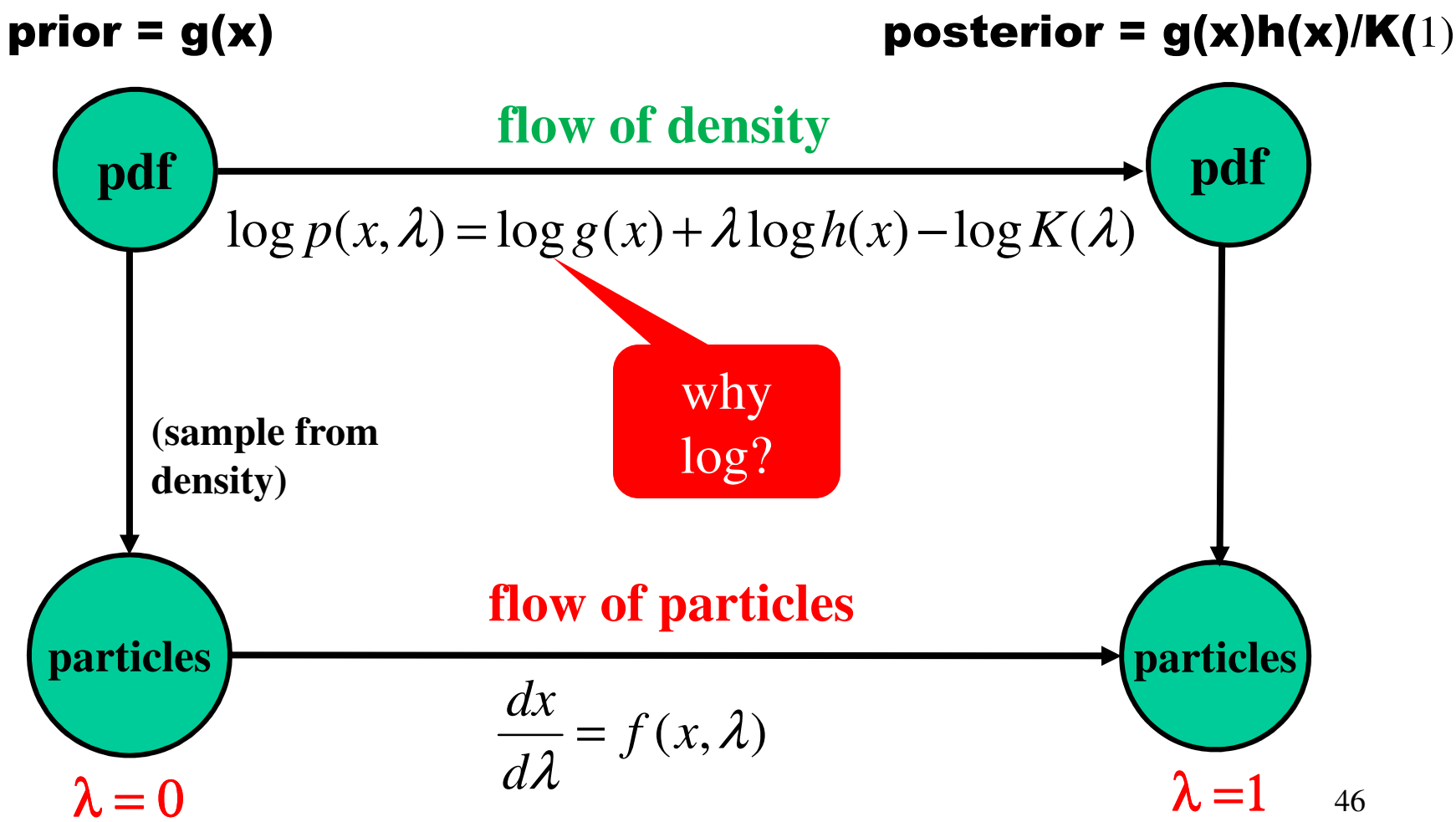
$t_k =$ time of the kth measurement

$z_k = k^{th}$ measurement vector

$Z_k =$ set of all measurements up to & including time t_k

$$Z_k = \{z_1, z_2, z_3, \ldots, z_k\}$$

induced flow of particles for Bayes' rule



convergence with N for particle filters:

$$\sigma^2 \approx c / N$$

N = number of particles

c = so-called "constant" which depends on:

(1) dimension of the state vector (x)
(2) initial uncertainty in the state vector
(3) measurement accuracy
(4) shape of probability densities (e.g., log-concave or multimodal etc.)
(5) Lipschitz constants of log densities
(6) stability of the plant
(7) curvature of nonlinear dynamics & measurements
(8) ill-conditioning of Fisher information matrix
(9) smoothness of densities & dynamics & measurements

Oh's Formula for Monte Carlo errors

$$\sigma^2 \approx \left\{ \left[\frac{1+k}{\sqrt{1+2k}} \right] \exp\left[\frac{\varepsilon^2}{1+2k} \right] \right\}^d / N$$

Assumptions:

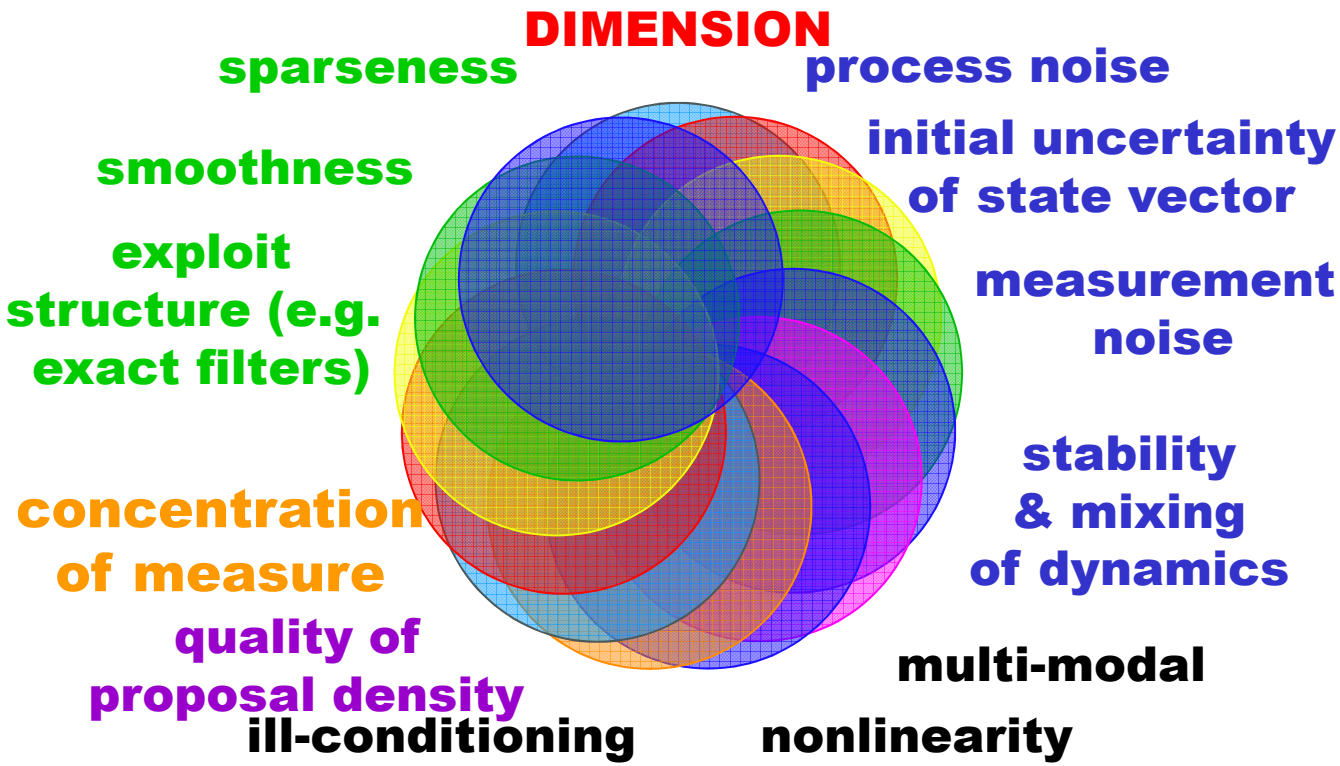
(1) Gaussian density (zero mean & unit covariance matrix)

(2) d-dimensional random variable

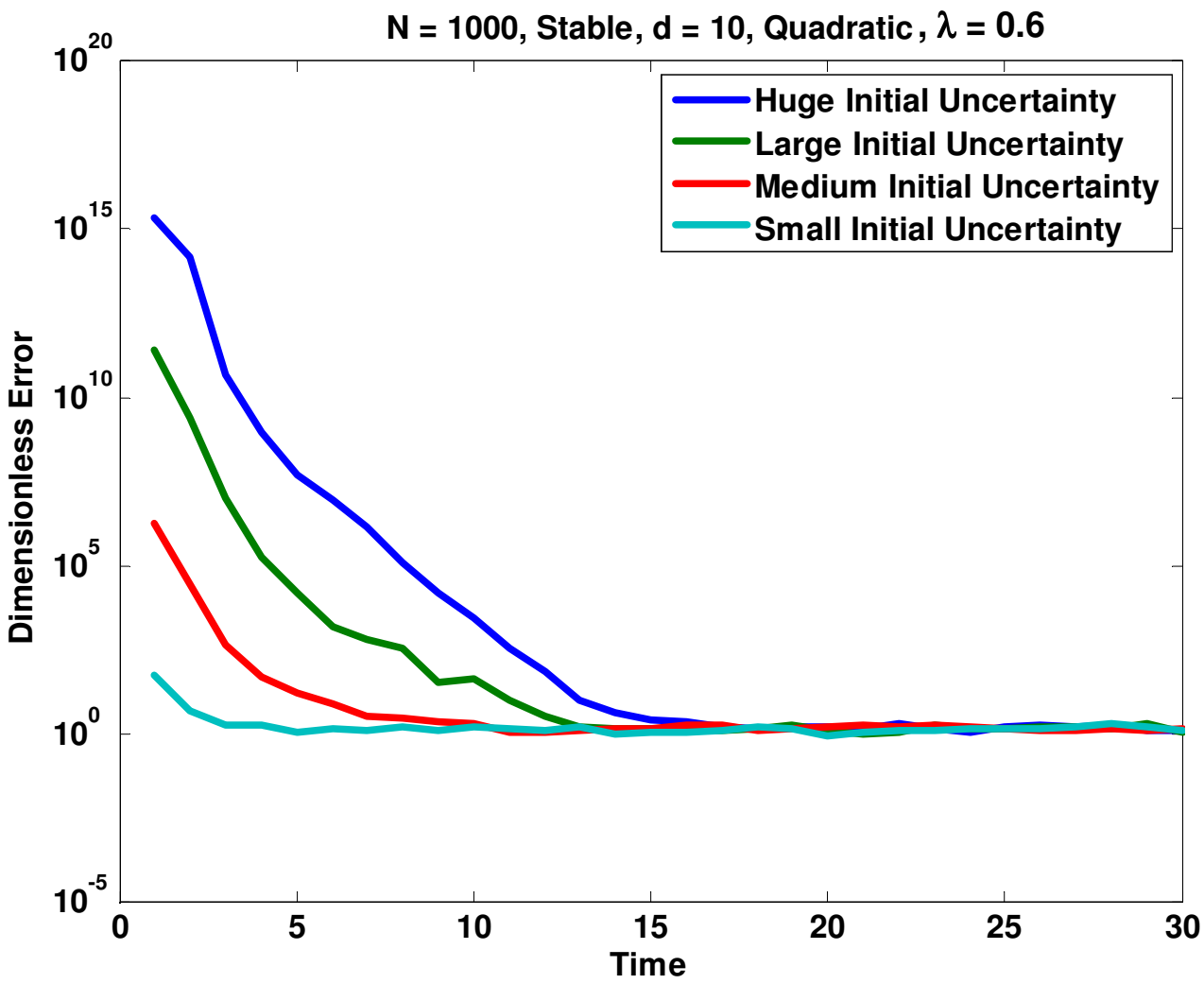
(3) Proposal density is also Gaussian with mean ε and covariance matrix kI, but it is not exact for $k \neq 1$ or $\varepsilon \neq 0$

(4) N = number of Monte Carlo trials

nonlinear filter performance (accuracy wrt optimal & computational complexity)

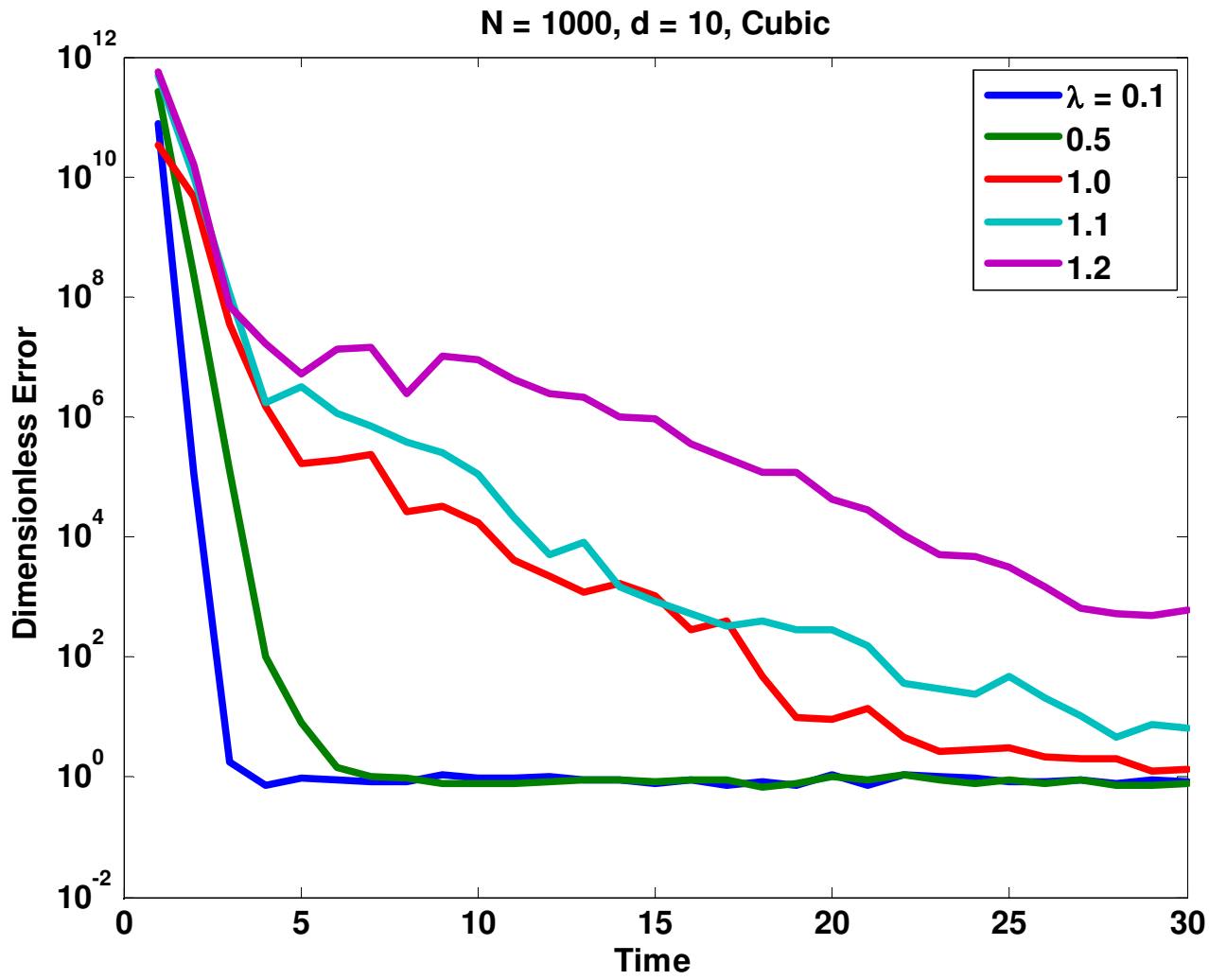


variation in initial uncertainty of x



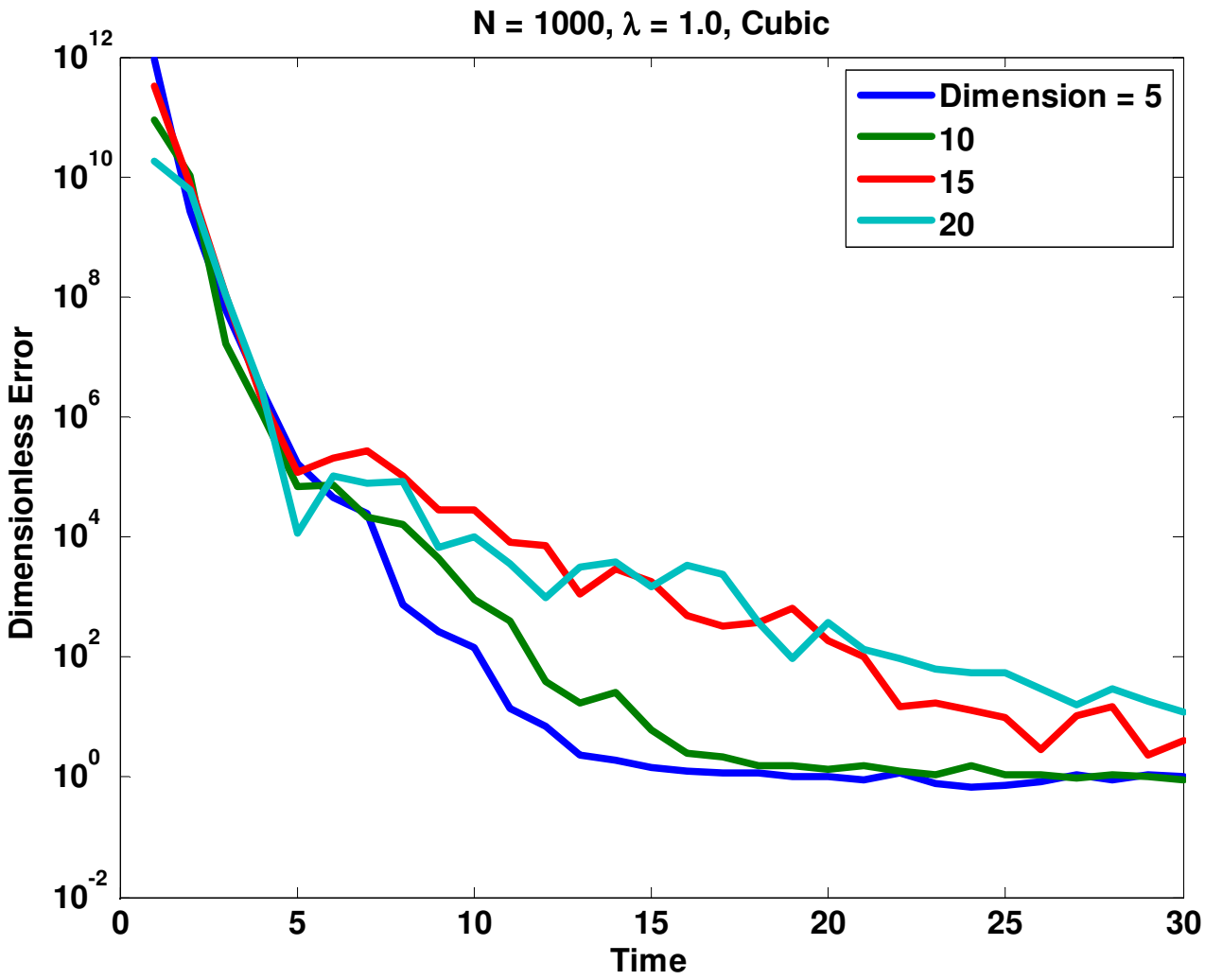
25 Monte Carlo Trials

variation in eigenvalues of the plant (λ)

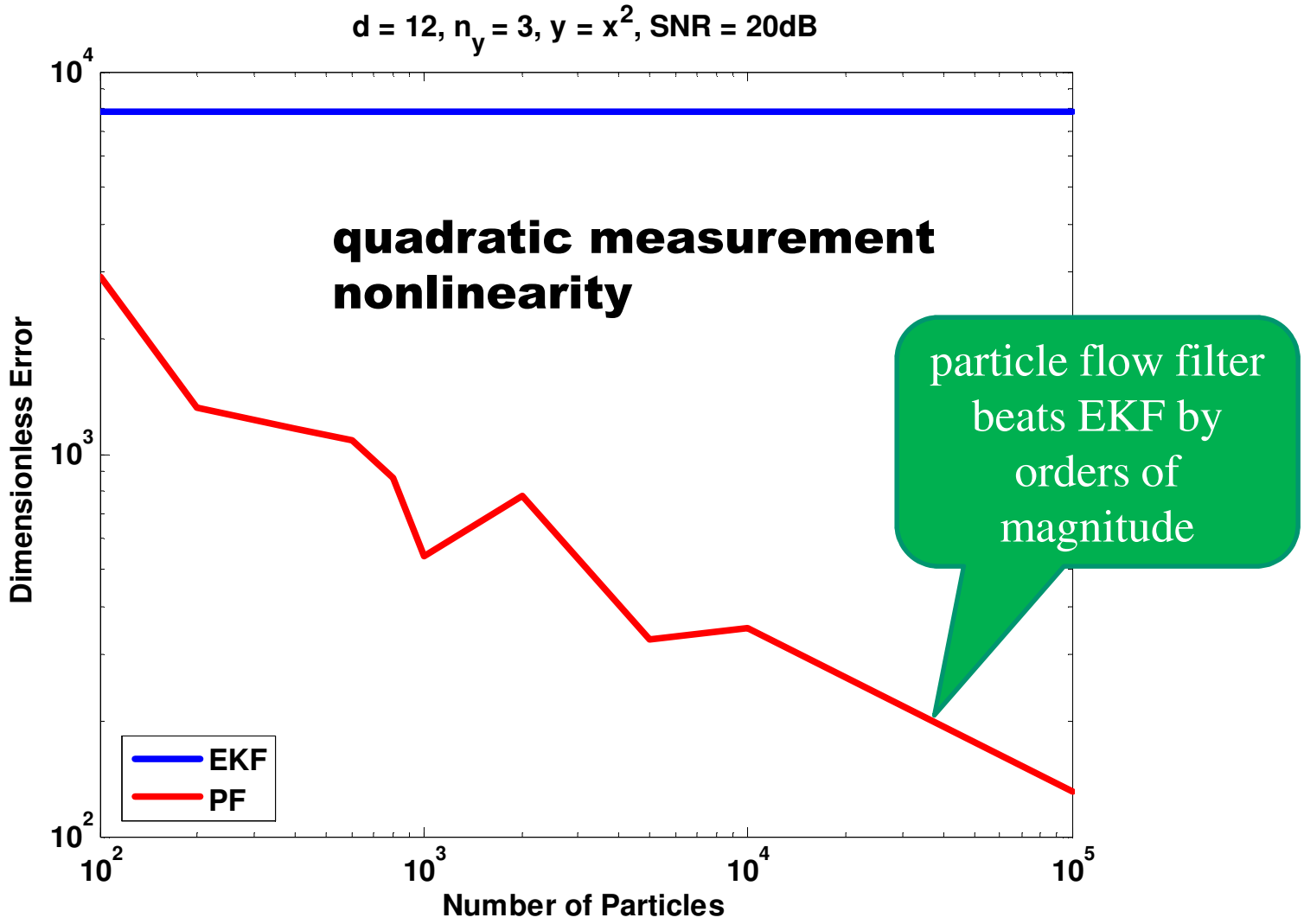


25 Monte Carlo Trials

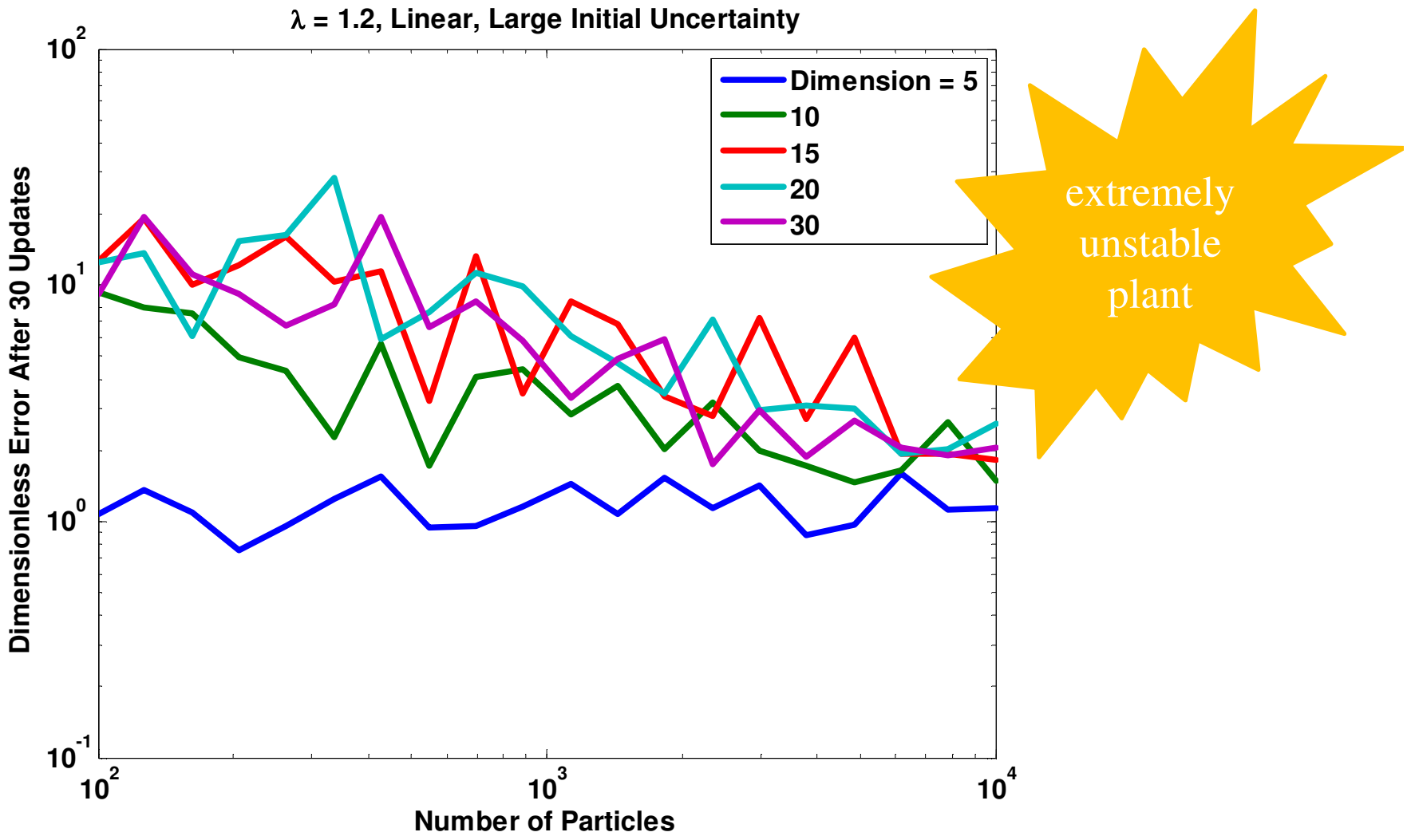
variation in dimension of x



25 Monte Carlo Trials

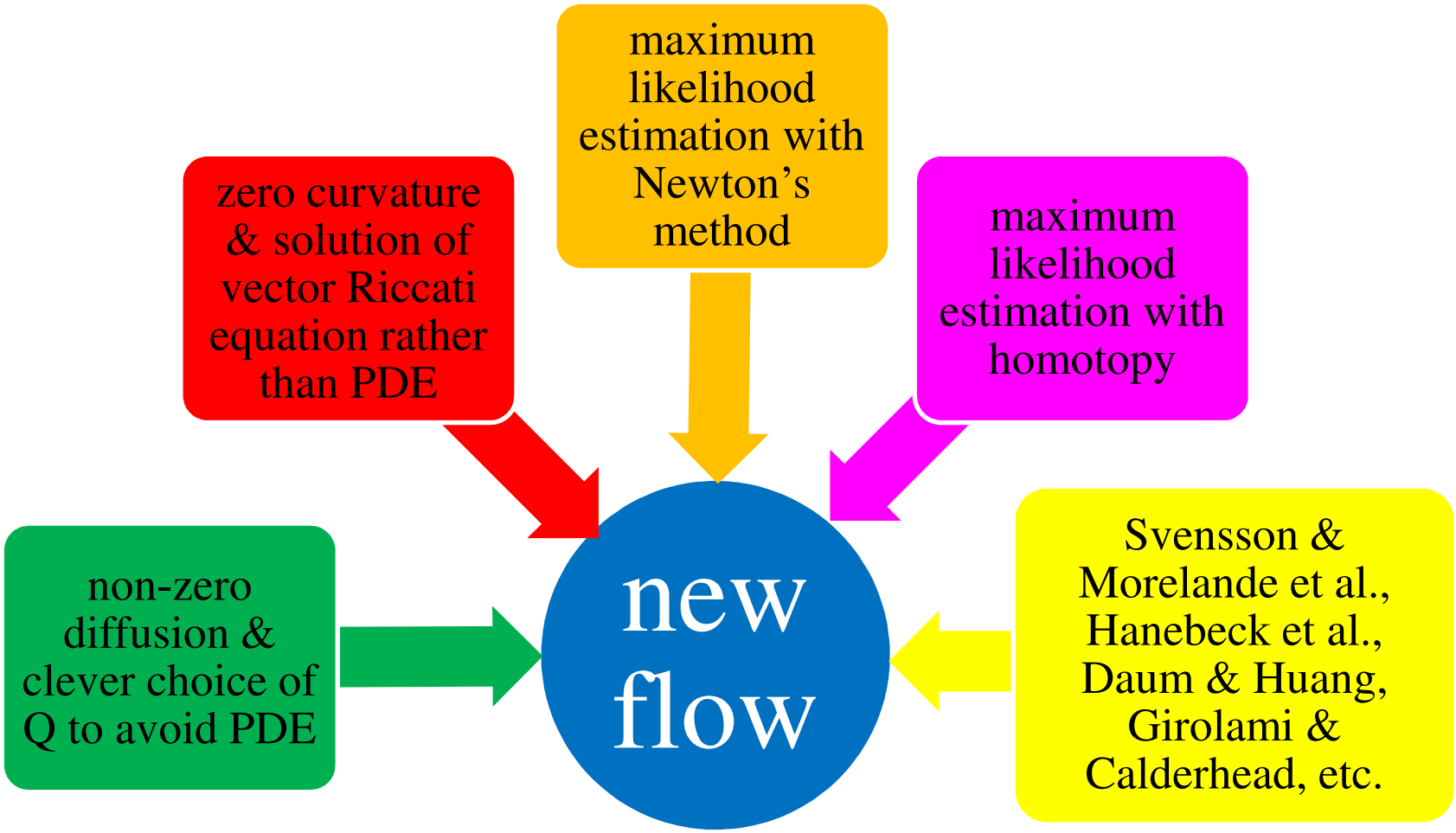

d = 12, n_y = 3, y = x^2, SNR = 20dB
quadratic measurement nonlinearity
Dimensionless Error
Number of Particles
EKF
PF
10^4
10^3
10^2
10^2
10^3
10^4
10^5
particle flow filter beats EKF by orders of magnitude

exact flow: performance vs. number of particles



25 Monte Carlo Trials

all roads lead to new flow:



computing the Hessian of log p:

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

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

can compute Hessians using calculus or 2nd differences

$$\frac{\partial^2 \log p}{\partial x^2} \approx -C^{-1} + \lambda \frac{\partial^2 \log h(x)}{\partial x^2}$$

$C =$ sample covariance matrix of particles for prior $(\lambda = 0)$

with Tychonov regularization; or EKF or UKF covariance matrix

$$\frac{\partial^2 \log p}{\partial x^2} \approx -P^{-1}$$

$P =$ sample covariance matrix of particles for $\mathrm{p}(\mathrm{x}, \lambda)$

with Tychonov regularization; or EKF or UKF covariance matrix

formula that avoids inverse of sample covariance matrix:

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

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

$$\frac{\partial^2 \log p}{\partial x^2} \approx -C^{-1} + \lambda \frac{\partial^2 \log h(x)}{\partial x^2} \quad \text{for g(x)} \approx \text{Gaussian}$$

but Woodbury's matrix inversion lemma gives us :

$$(\mathrm{A}+\mathrm{B})^{-1} = A^{-1} - A^{-1}B(I + A^{-1}B)^{-1}A^{-1} \quad \text{for arbitrary B and non - singular A}$$

hence

$$\left[\frac{\partial^2 \log p}{\partial x^2}\right]^{-1} \approx -C - CB(I - CB)^{-1}C$$

in which

$$\mathrm{B} = \lambda \frac{\partial^2 \log h(x)}{\partial x^2}$$

how to mitigate stiffness in ODEs for certain particle flows*

method	computational complexity	filter accuracy	comments
1. use a stiff ODE solver (e.g., implicit integration rather than explicit)	large to extremely large	uncertain	textbook advice & many papers
2. use very small integration steps everywhere	extremely large	good	brute force solution
3. use very small integration steps only where needed (adaptively determined)	large		
4. use very small integration steps only where needed (determined non-adaptively)	small	2nd best	easy to do with particle flow
5. transform to principal coordinates or approximately principal coordinates	small	best	easy to do for certain applications
6. Battin's trick (i.e., sequential scalar measurement updates)	small	very bad	destroys the benefit of particle flow
7. Tychonov regularization of the Hessian of log p	very small	uncertain	

*Daum & Huang, "seven dubious methods to mitigate stiffness in particle flow for nonlinear filters," Proceedings of SPIE Conference, May 2014.

RED flows are extremely stiff*

incompressible flow	irrotational flow	Coulomb's law flow	small curvature flow
Gaussian densities	exponential family	Fourier transform flow	constant curvature flow (e.g. zero)
Knothe-Rosenblatt flow	non-zero diffusion flow	method of characteristics	geodesic flow
stabilized flows	finite dimensional flow	direct integration	Monge-Kantorovich transport

***non-stiff flows work well with Euler integration $\Delta\lambda = 0.1$**

new particle flow :

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

If we approximate the density p as Gaussian, then the observed Fisher information matrix can be computed using the sample covariance matrix (P) over the set of particles:

$$\frac{dx}{d\lambda} \approx P\left(\frac{\partial \log h}{\partial x}\right)^T$$

for Gaussian densities we get the EKF for each particle:

$$\frac{dx}{d\lambda} \approx P\left(\frac{\partial \theta(x)}{\partial x}\right)^T R^{-1}\left(z - \theta(x)\right)$$

importance of avoiding explicit computation of normalization K(λ)*

(1) our 3 best flows do not explicitly compute the normalization of the conditional density

(2) small errors in computing K(λ) can ruin the filter accuracy (e.g., Coulomb's law flow & Fourier transform flow)

(3) similar effect in numerical weather prediction using transport theory (Bath University); must compute K(λ) to machine precision!

(4) exploit this effect in designing flows

*Daum & Huang, "how to avoid normalization of particle flow for nonlinear filters," Proceedings of SPIE Conference, May 2014.

RED flows do not explicitly compute normalization K(λ)

incompressible flow	irrotational flow	Coulomb's law flow	small curvature flow
Gaussian densities	exponential family	Fourier transform flow	constant curvature flow (e.g. zero curvature)
Knothe-Rosenblatt flow	non-zero diffusion flow	method of characteristics	geodesic flow
stabilized flows	finite dimensional flow	direct integration	Monge-Kantorovich transport

exact particle flow for Gaussian densities:

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

$$\log(h) - \frac{d\log K(\lambda)}{d\lambda} = -div(f) - \frac{\partial \log p}{\partial x} f$$

for g & h Gaussian, we can solve for f exactly:

$$f = Ax + b$$

$$A = -\frac{1}{2} PH^T \left[\lambda HPH^T + R\right]^{-1} H$$

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

f does not depend on K(λ), despite the fact that the PDE does!

incompressible particle flow

$$\frac{\mathrm{dx}}{\mathrm{d}\lambda} = \begin{cases} -\log(h(x)) \dfrac{\left[\dfrac{\partial \log p(x,\lambda)}{\partial x}\right]^T}{\left\|\dfrac{\partial \log p(x,\lambda)}{\partial x}\right\|^2} & \text{for non - zero gradient} \\ 0 \quad \text{otherwise} & \end{cases}$$

for $d \geq 2$

f does not depend on K(λ), despite the fact that the PDE does!

new particle flow :

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

If we approximate the density p as Gaussian, then the observed Fisher information matrix can be computed using the sample covariance matrix (P) over the set of particles:

$$\frac{dx}{d\lambda} \approx P\left(\frac{\partial \log h}{\partial x}\right)^T$$

f does not depend on K(λ), despite the fact that the PDE does!

for Gaussian densities we get the EKF for each particle:

$$\frac{dx}{d\lambda} \approx P\left(\frac{\partial \theta(x)}{\partial x}\right)^T R^{-1}\left(z - \theta(x)\right)$$

most general solution for particle flow:

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

$$\log(h) - \frac{d\log K(\lambda)}{d\lambda} - = -div(f) - \frac{\partial \log p}{\partial x} f$$

the most general solution is:

$$\mathrm{f} = \text{-}\mathrm{C}^{\#}[\log h - \frac{d\log K(\lambda)}{d\lambda}] + (I - C^{\#}C)y$$

in which C is a linear differential operator:

$$\mathrm{C} = \frac{\partial \mathrm{logp}}{\partial \mathrm{x}} + div$$

$C^{\#}$ = generalized inverse of C

y = arbitrary d-dimensional vector

could pick y to robustly stabilize the filter or random or zero or other

idea #1 inspired by renormalization group flow:

$$\frac{\mathrm{dx}}{\mathrm{d}\lambda} = \text{-}\mathrm{C}^{\#}[\log h - \frac{d\log K(\lambda)}{d\lambda}] + (I - C^{\#}C)\,y$$

$$\mathrm{f} = \Gamma + \Pi y$$

$$\Pi = \text{projection into null-space of C}$$

$$\frac{\partial f}{\partial L} = \frac{\partial \Gamma}{\partial L} + \frac{\partial \Pi}{\partial L} y = 0$$

$$y = -\left[\frac{\partial \Pi}{\partial L}\right]^{\#}\left(\frac{\partial \Gamma}{\partial L}\right)$$

$$L = d\log K(\lambda)/d\lambda$$

$$(.)^{\#} = \text{generalized inverse of } (.)$$

L rather than K (just like QFT): linear in L but not K; result does not depend on K itself; avoids singularity; slightly different & it works

idea #2 inspired by renormalization group flow:

$$\frac{\mathrm{dx}}{\mathrm{d}\lambda} = \text{-}\mathrm{C}^{\#}[\log h - \frac{d\log K(\lambda)}{d\lambda}] + (I - C^{\#}C)\,y$$

$$\mathrm{f} = \Gamma + \Pi y$$

$$\Pi = \text{projection into null-space of C}$$

$$\frac{\partial f}{\partial L} = \frac{\partial \Gamma}{\partial L} + \frac{\partial \Pi}{\partial L} y = 0$$

$$y = -\left[\frac{\partial \Pi}{\partial L}\right]^{\#}\left(\frac{\partial \Gamma}{\partial L}\right)$$

$$L = \log g(x)$$

$$(.)^{\#} = \text{generalized inverse of } (.)$$

L rather than g (just like QFT): linear in L but not g; result does not depend on g itself; avoids singularity at g = 0; slightly different & it works

idea #3 inspired by renormalization group flow*

$$\frac{\mathrm{dx}}{\mathrm{d}\lambda} = \text{-}\mathrm{C}^{\#}[\log h - \frac{d\log K(\lambda)}{d\lambda}] + (I - C^{\#}C)\,y$$

$$\mathrm{f} = \Gamma + \Pi y$$

$$\Pi = \text{projection into null-space of C}$$

$$\frac{\partial f}{\partial L} = \frac{\partial \Gamma}{\partial L} + \frac{\partial \Pi}{\partial L}\,y = 0$$

$$y = -\left[\frac{\partial \Pi}{\partial L}\right]^{\#}\left(\frac{\partial \Gamma}{\partial L}\right)$$

$$L = \{\log g(x), d\log K(\lambda)/d\lambda\}^T$$

$$(.)^{\#} = \text{generalized inverse of } (.)$$

*Daum & Huang, "renormalization group flow & other ideas inspired by physics for nonlinear filters," Proceedings of SPIE Conference, May 2014.

computation of normalization using Fourier transform:

$$div(pf) = -p[\log h - \frac{d\log K(\lambda)}{d\lambda}]$$

take the Fourier transform:

$$\mathrm{i}\omega^T \Im(\mathrm{pf}) = -\Im\left\{\mathrm{p}\left[\log\mathrm{h} - \frac{\mathrm{d}\log\mathrm{K}(\lambda)}{\mathrm{d}\lambda}\right]\right\}$$

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

evaluate above at $\omega = 0$ (assuming that E(f) is finite)

$$0 = \int \mathrm{p}(\mathrm{x},\lambda)\left[\log\mathrm{h}(\mathrm{x}) - \frac{\mathrm{d}\log\mathrm{K}(\lambda)}{\mathrm{d}\lambda}\right]dx$$

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

approximate the integral using the Monte Carlo sum over particles:

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

MOVIES

exact particle flow for Gaussian densities:

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

$$\log(h) = -div(f) - \frac{\partial \log p}{\partial x} f$$

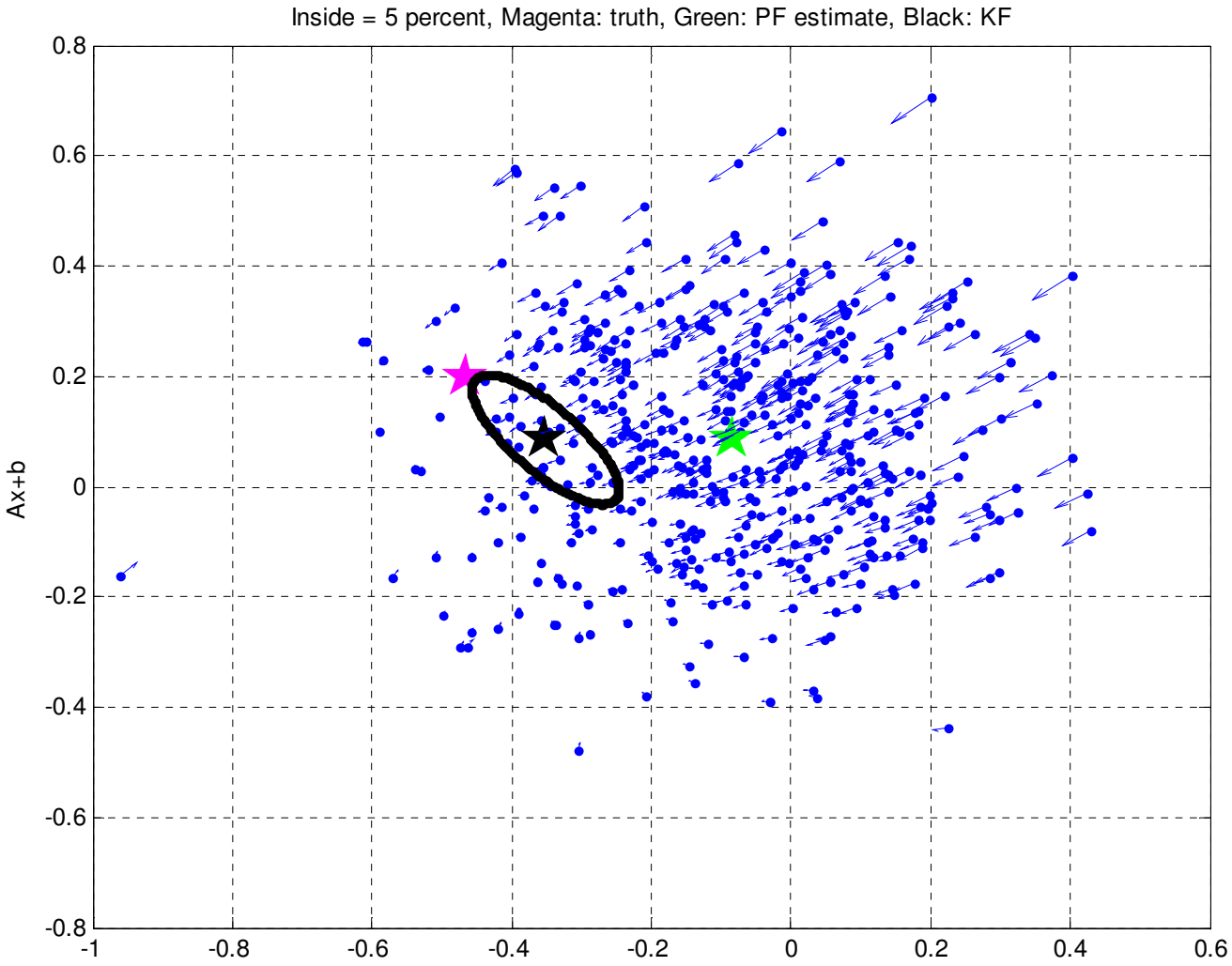
for g & h Gaussian, we can solve for f exactly:

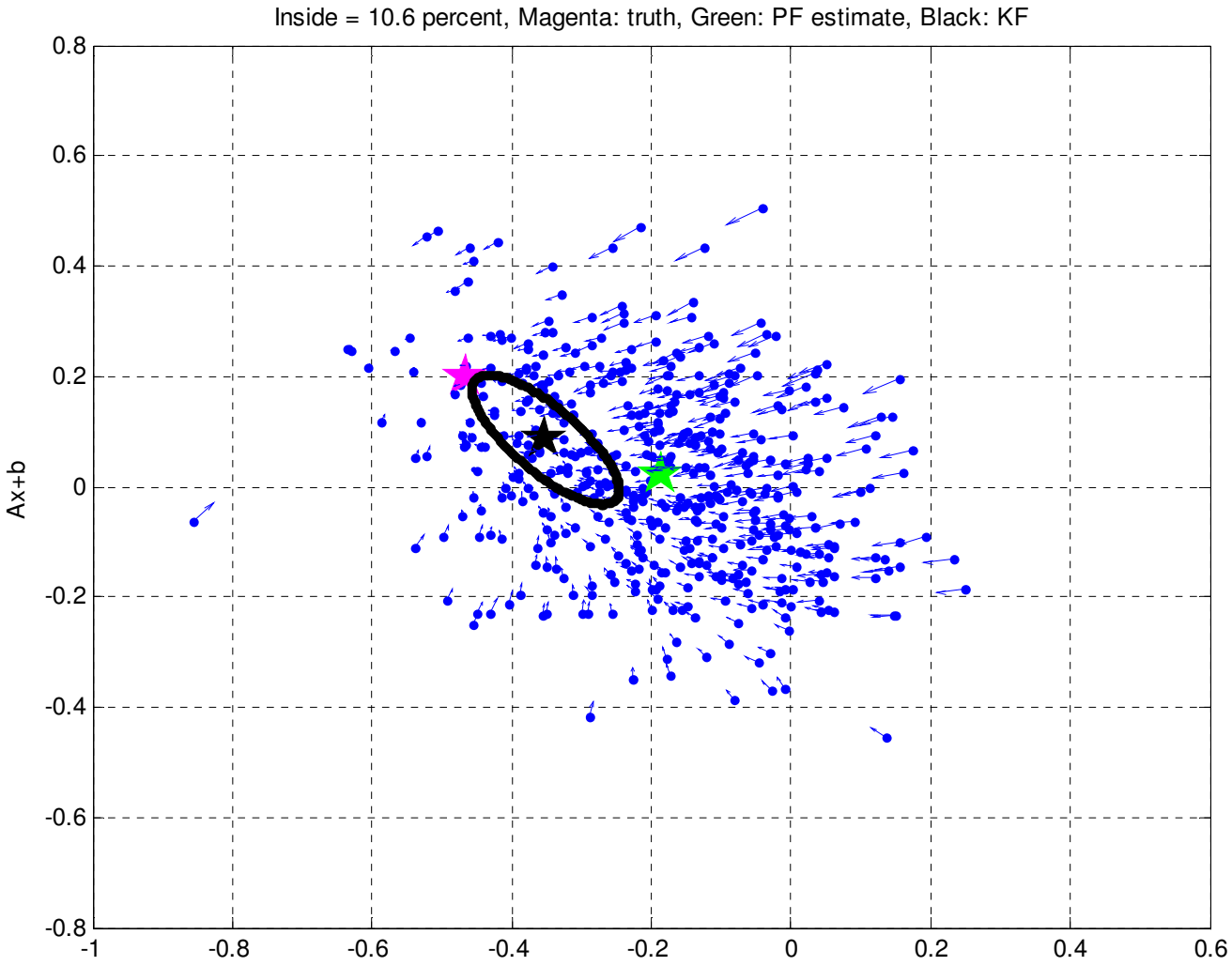
$$f = Ax + b$$

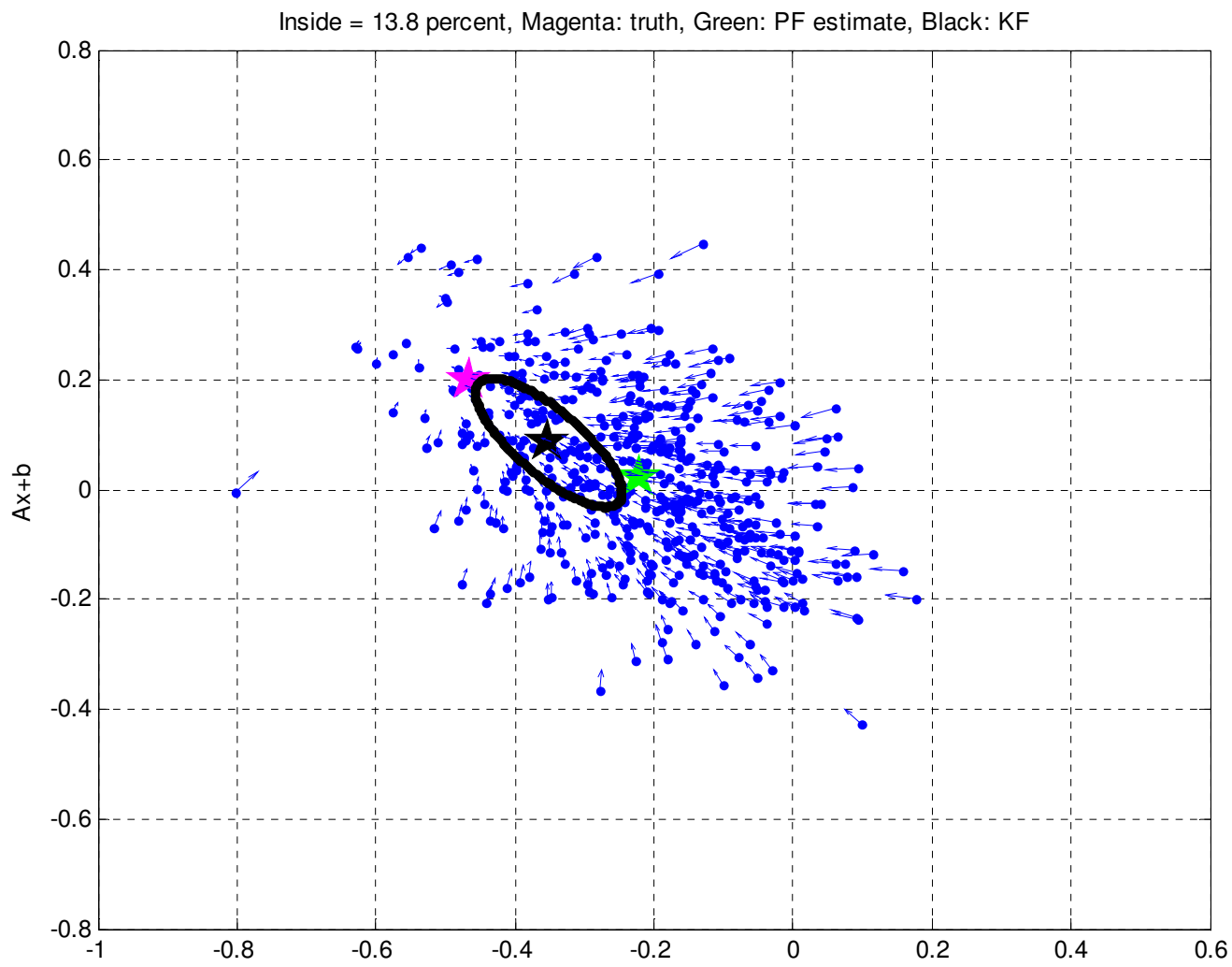
$$A = -\frac{1}{2} PH^T \left[\lambda HPH^T + R\right]^{-1} H$$

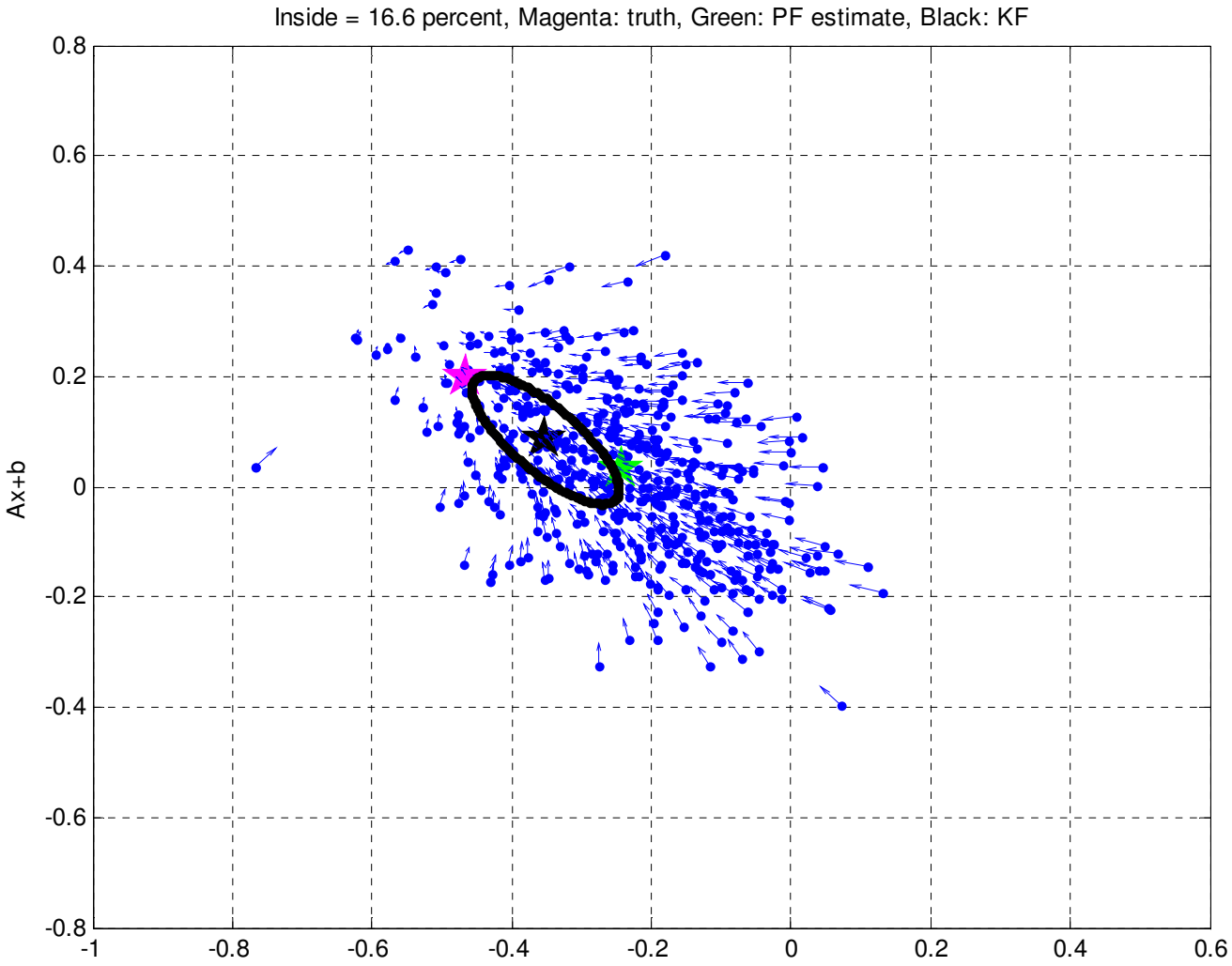
$$b = (I + 2\lambda A)\left[(I + \lambda A)PH^T R^{-1} z + A\bar{x}\right]$$

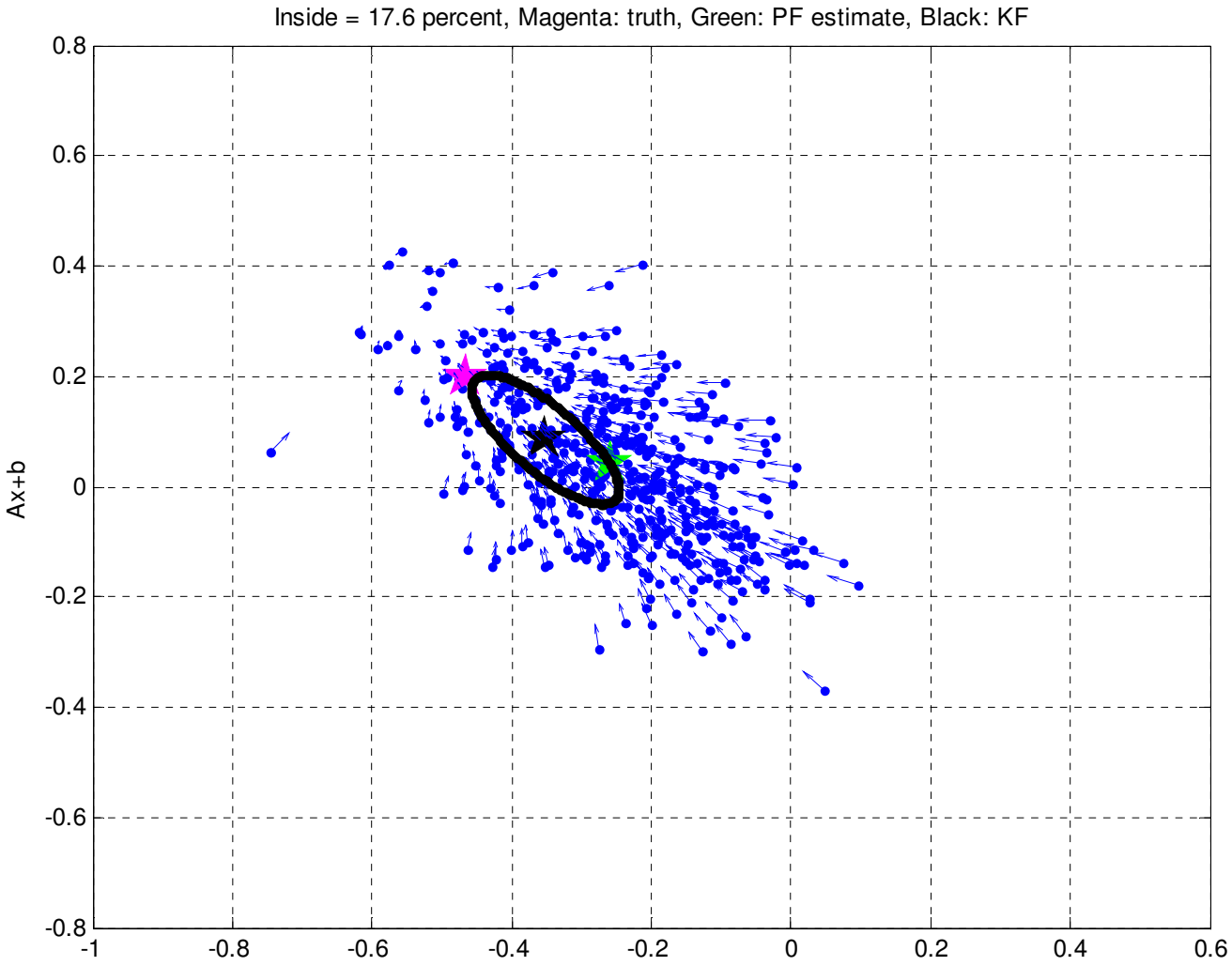
automatically stable under very mild conditions & extremely fast

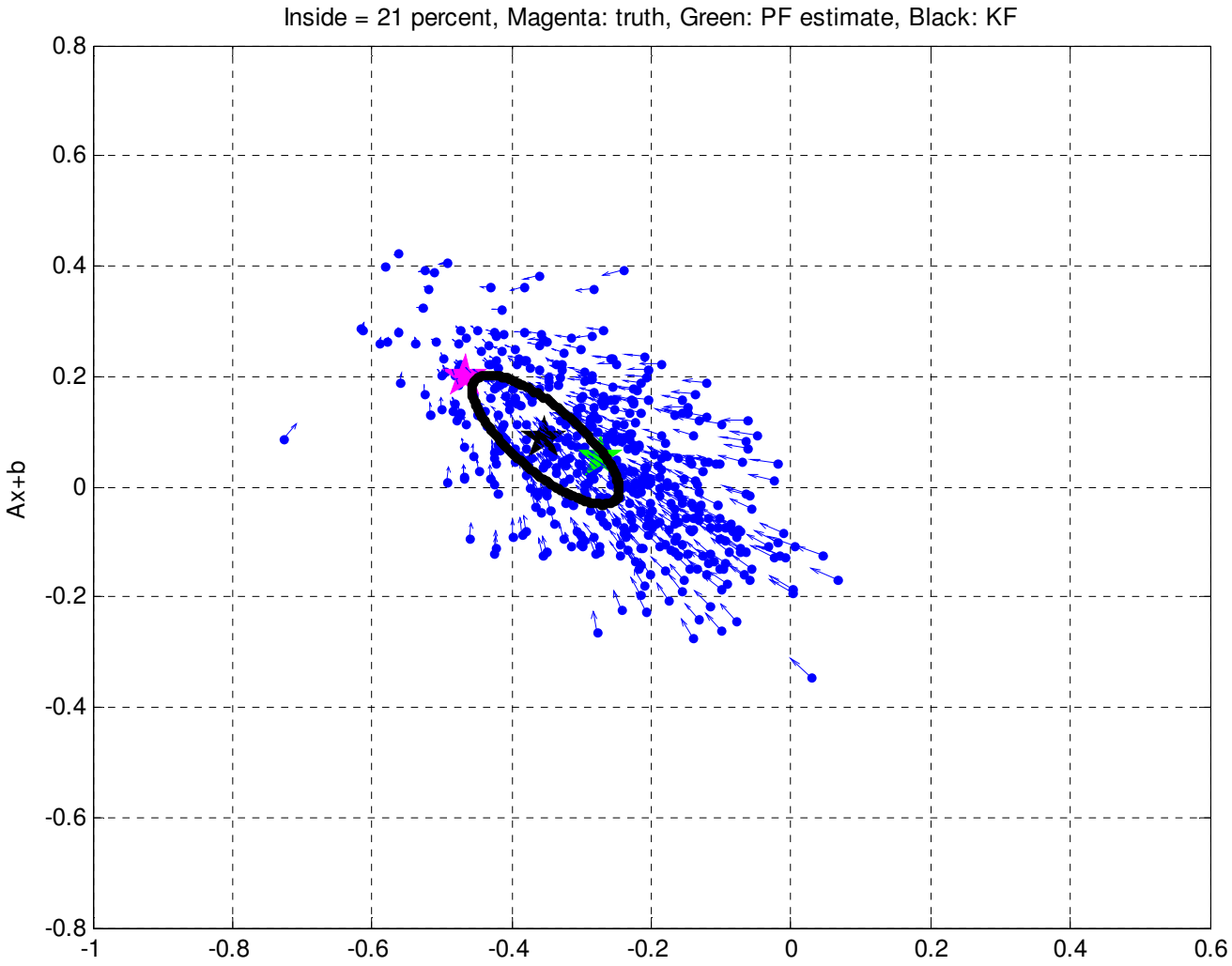

Inside = 5 percent, Magenta: truth, Green: PF estimate, Black: KF
Ax+b
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6

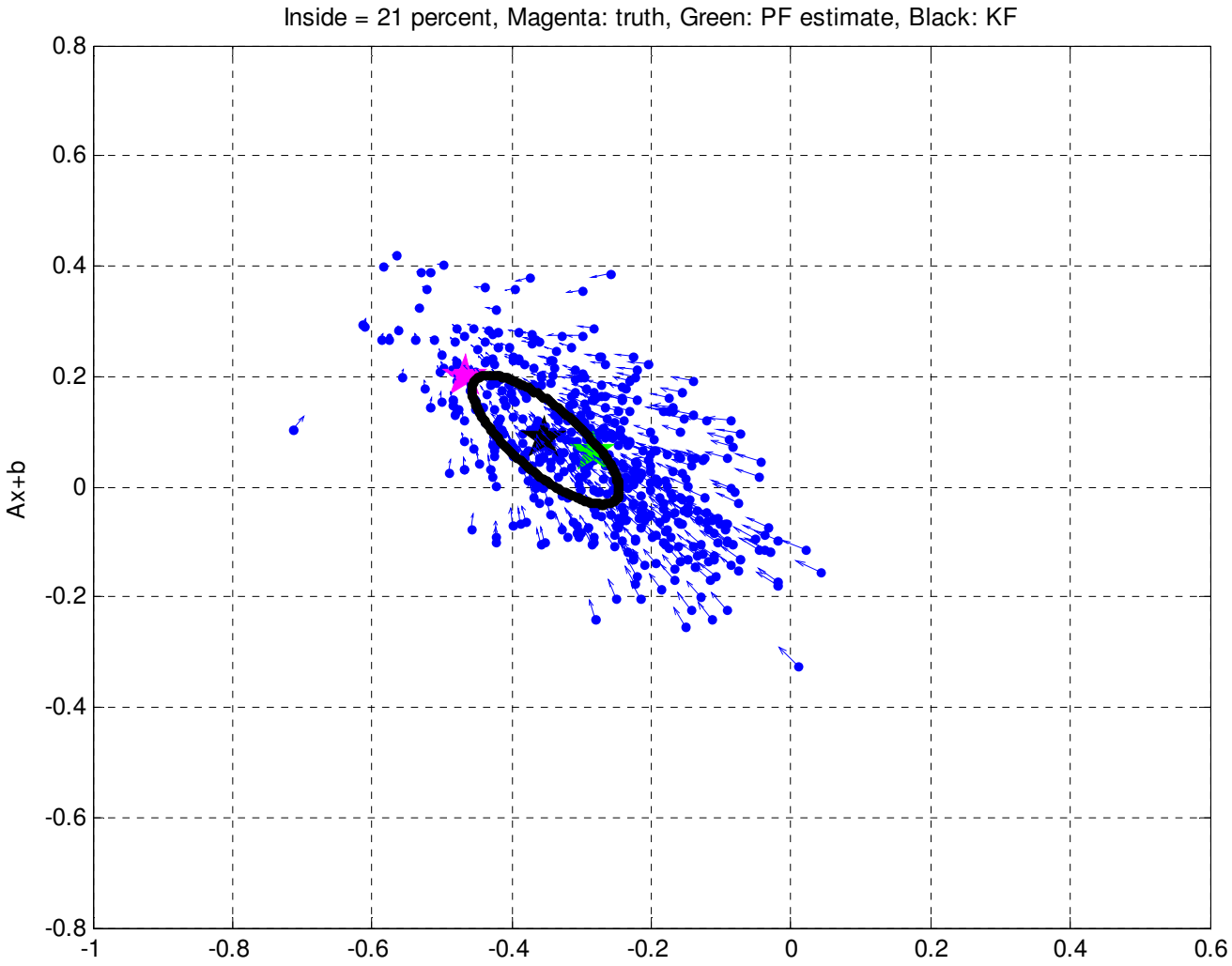

Inside = 10.6 percent, Magenta: truth, Green: PF estimate, Black: KF
Ax+b
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6

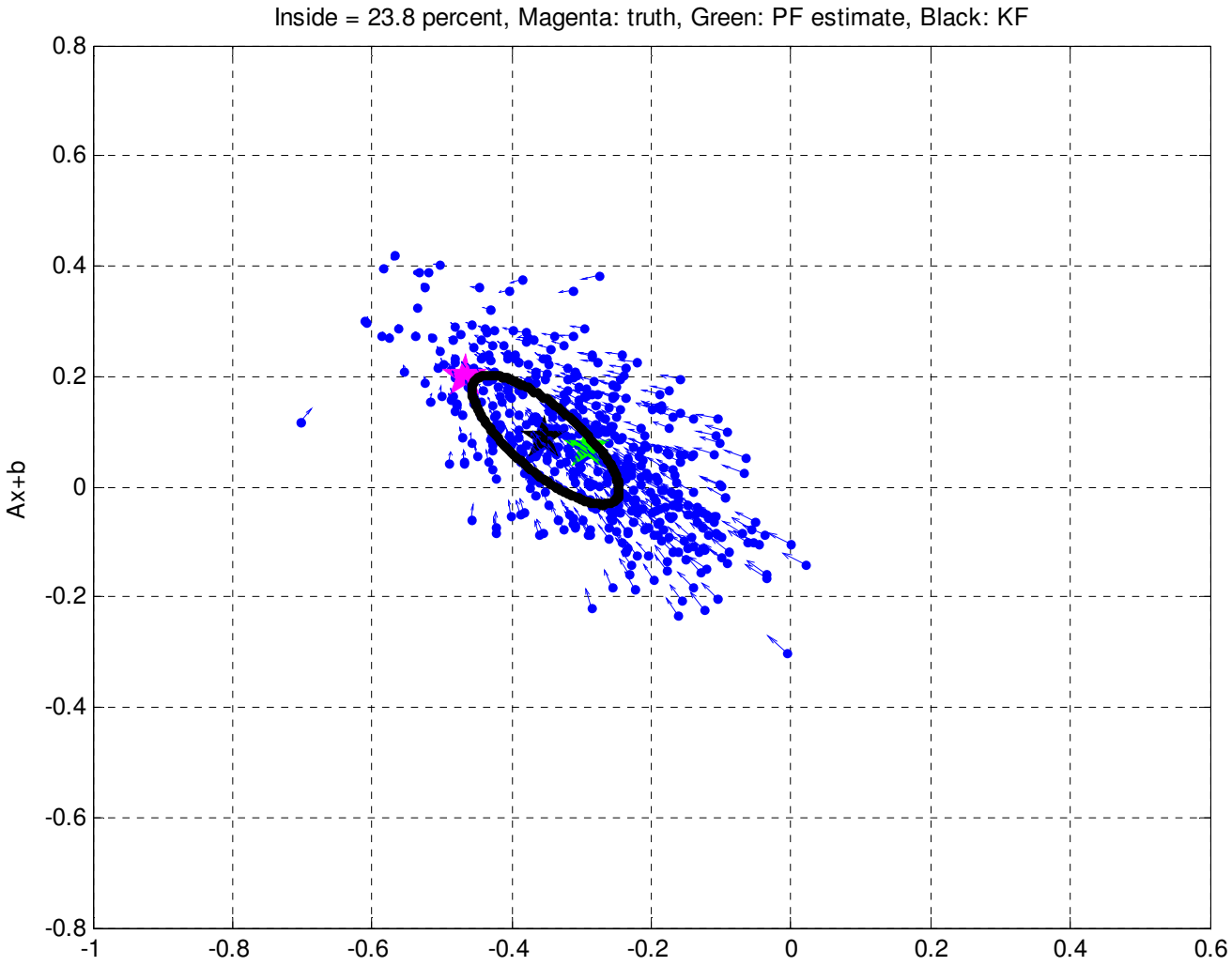

Inside = 13.8 percent, Magenta: truth, Green: PF estimate, Black: KF
Ax+b
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6

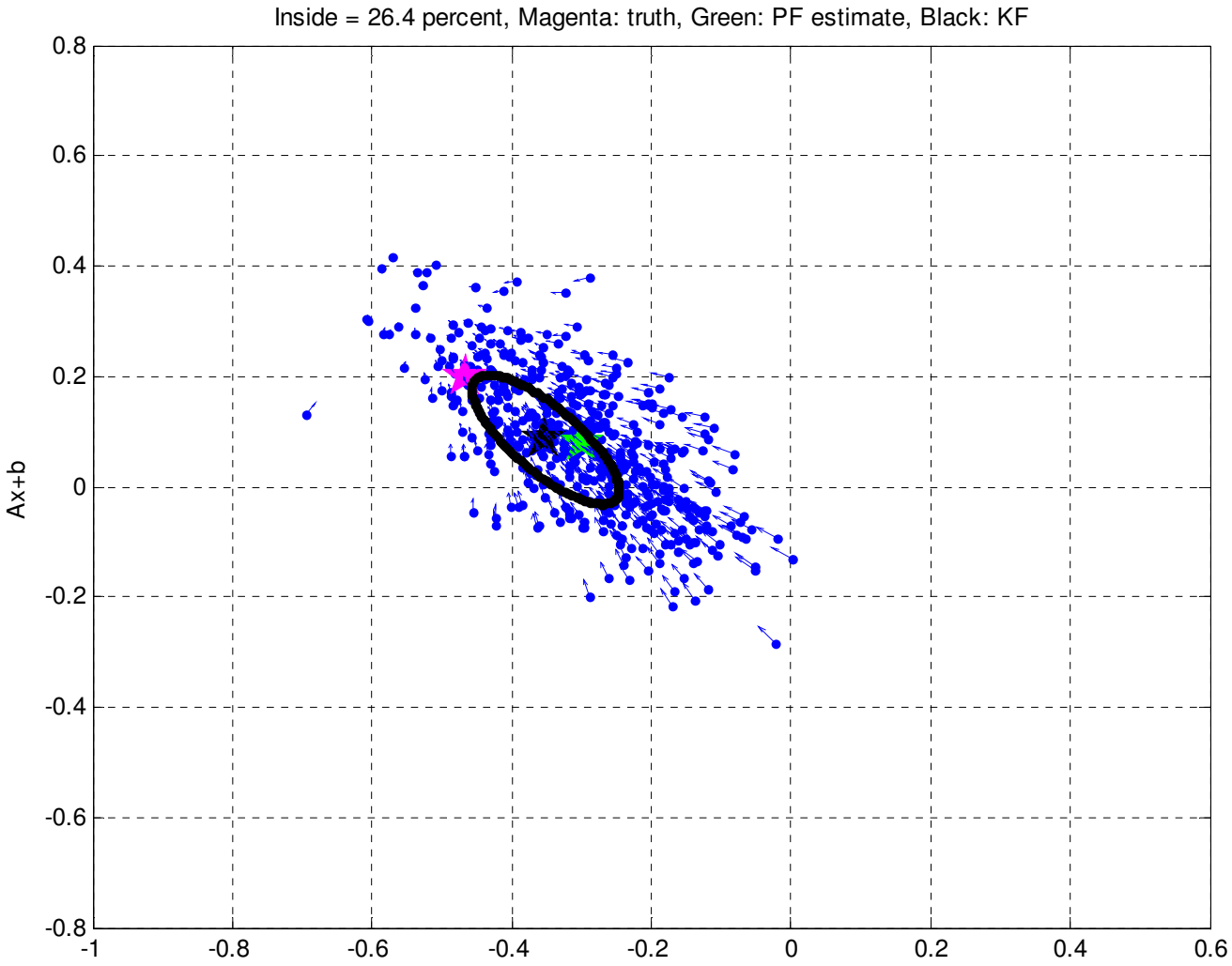

Inside = 16.6 percent, Magenta: truth, Green: PF estimate, Black: KF
Ax+b
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6

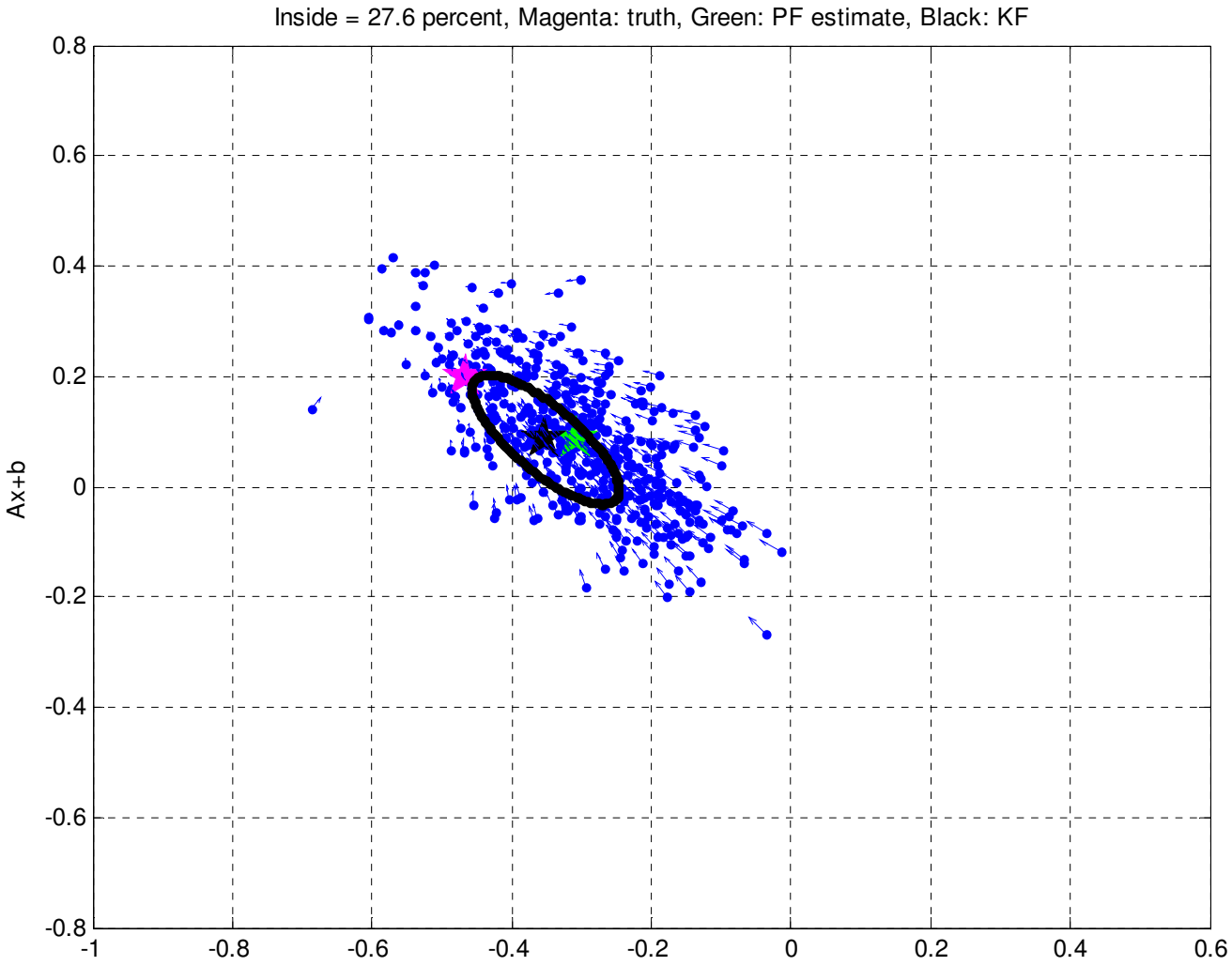

Inside = 17.6 percent, Magenta: truth, Green: PF estimate, Black: KF
Ax+b
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6


Inside = 21 percent, Magenta: truth, Green: PF estimate, Black: KF
Ax+b
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6


Inside = 21 percent, Magenta: truth, Green: PF estimate, Black: KF
Ax+b
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6


Inside = 23.8 percent, Magenta: truth, Green: PF estimate, Black: KF
Ax+b
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6

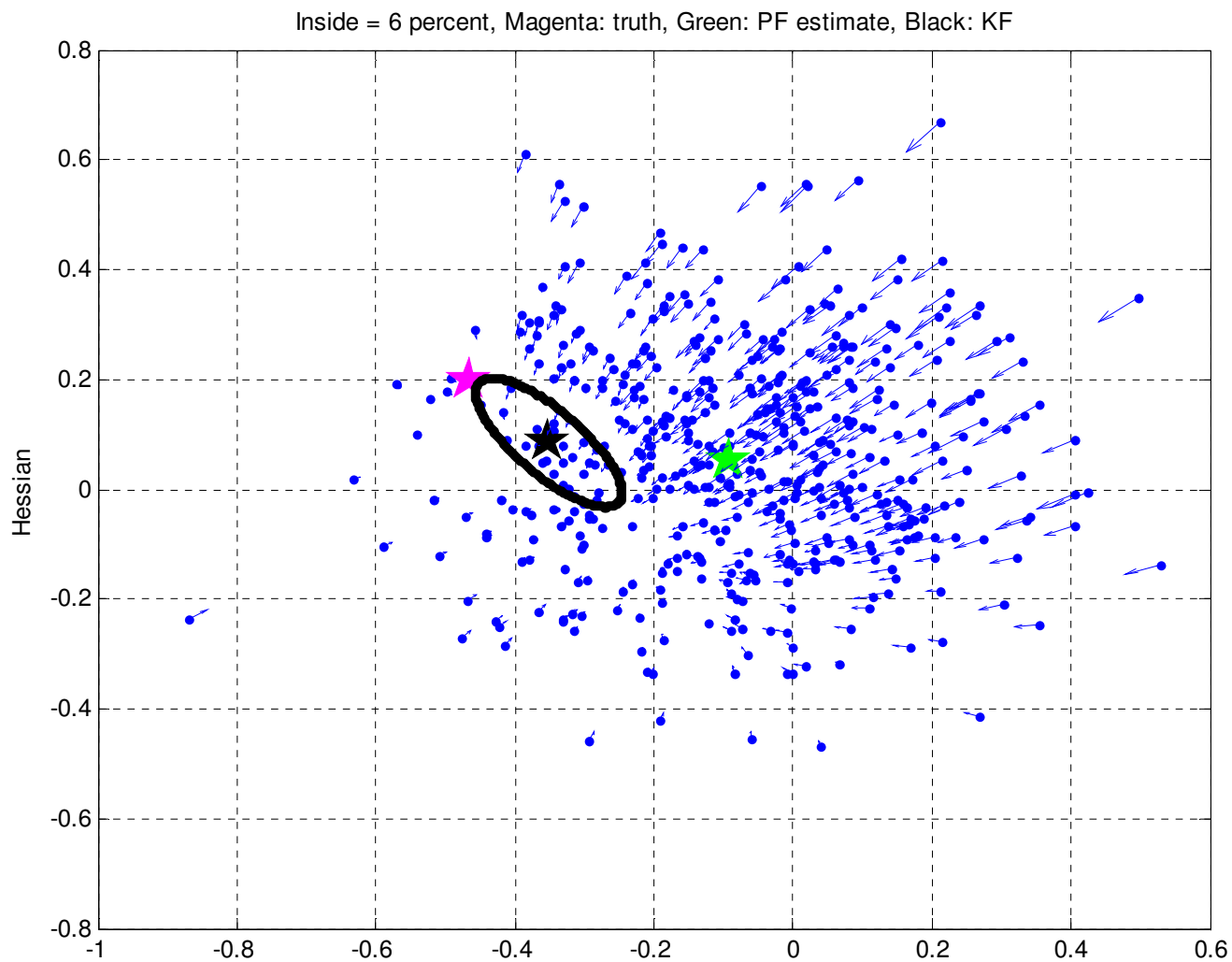

Inside = 26.4 percent, Magenta: truth, Green: PF estimate, Black: KF
Ax+b
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6

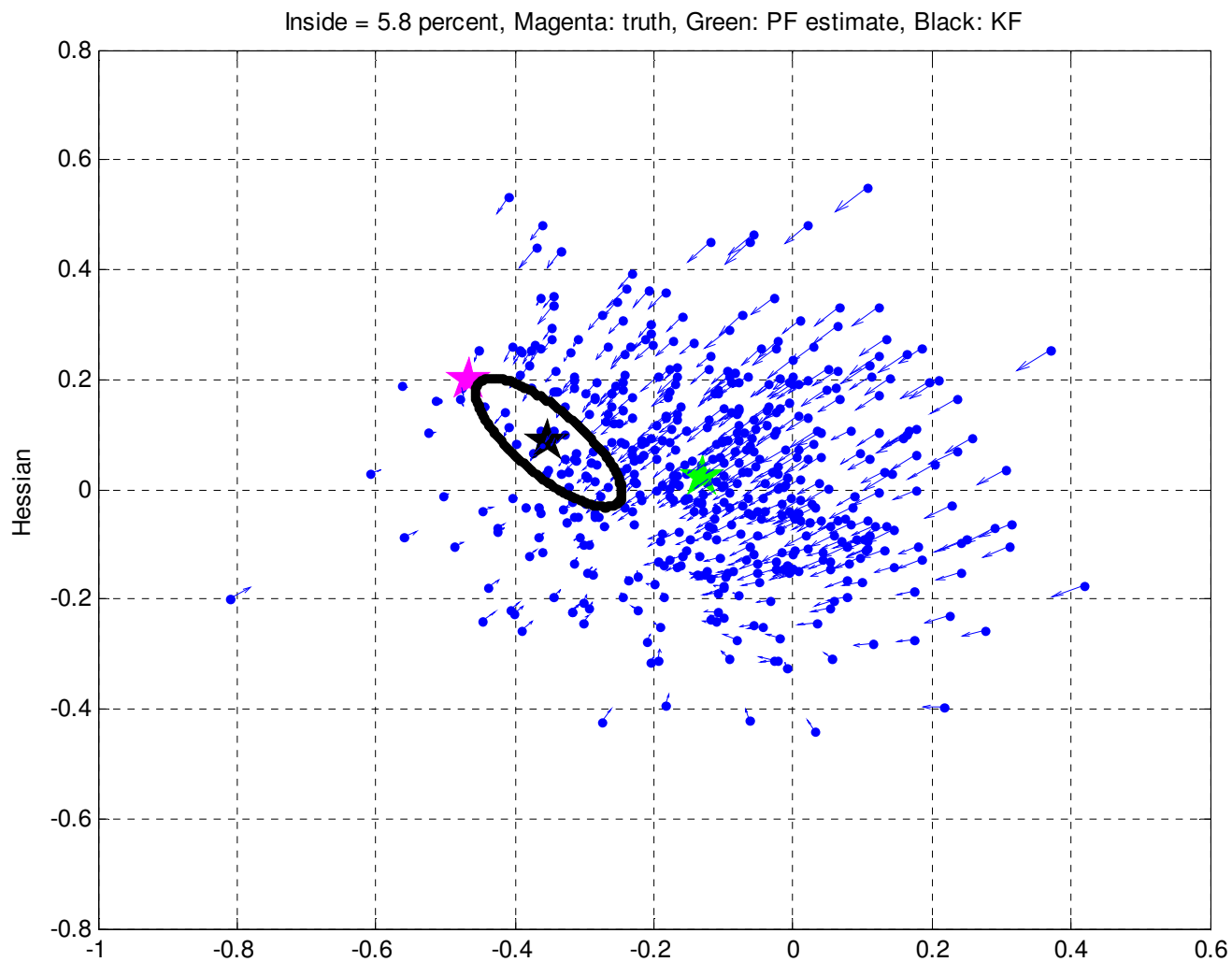

Inside = 27.6 percent, Magenta: truth, Green: PF estimate, Black: KF
Ax+b
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6

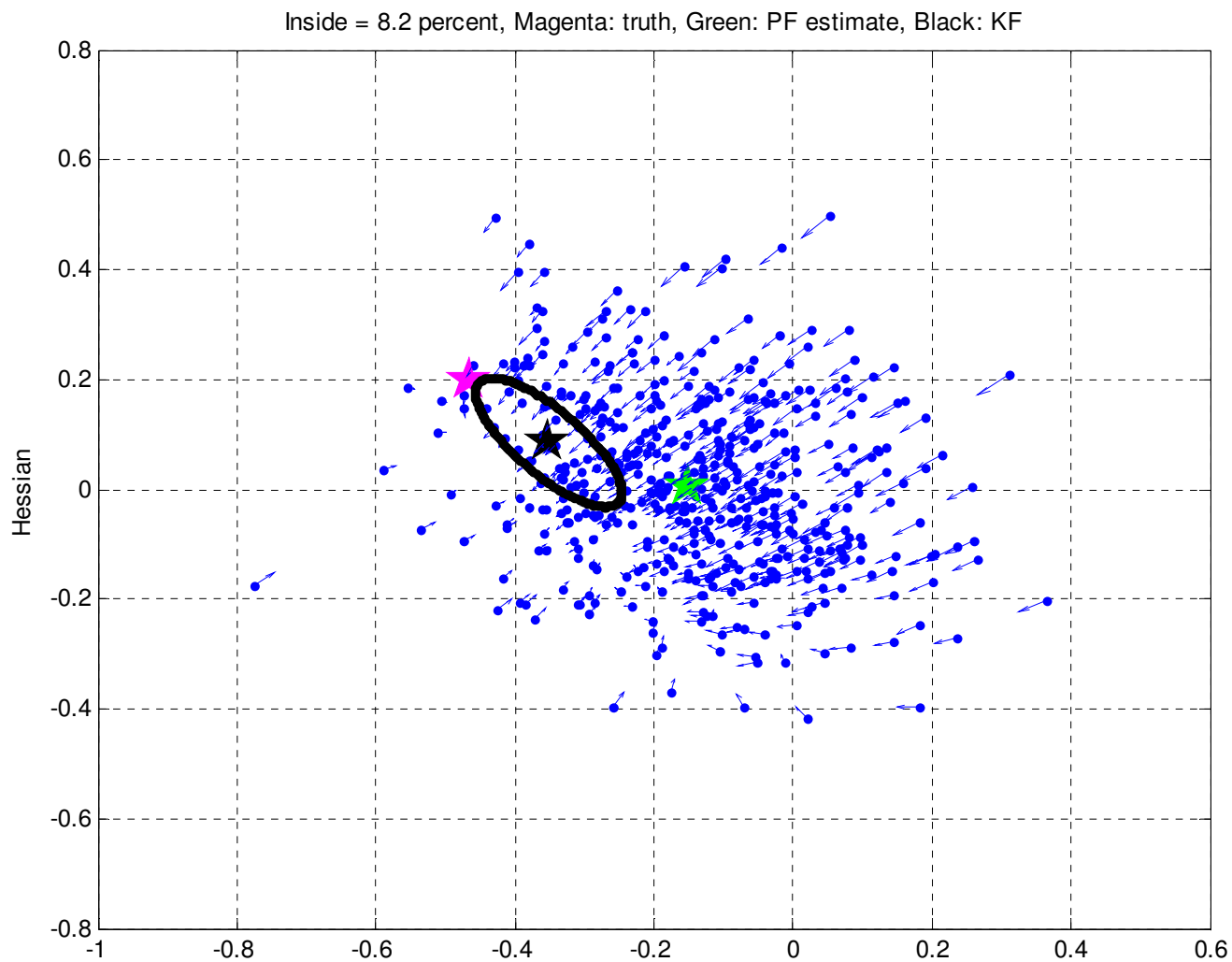
incompressible particle flow

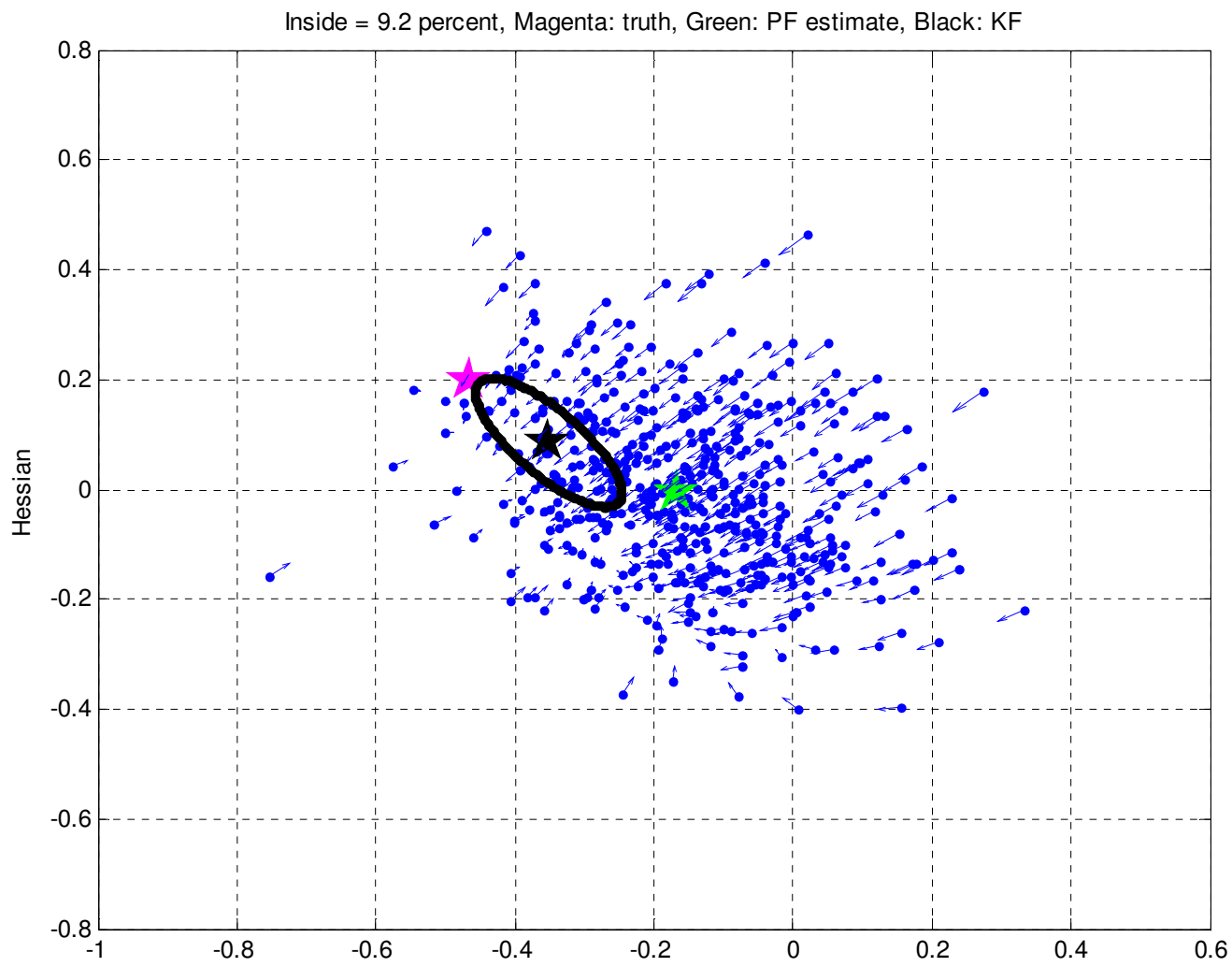
$$\frac{\mathrm{dx}}{\mathrm{d}\lambda} = \frac{-\log(h)\left(\dfrac{\partial \log p}{\partial x}\right)^T}{\left\|\dfrac{\partial \log p}{\partial x}\right\|^2}$$

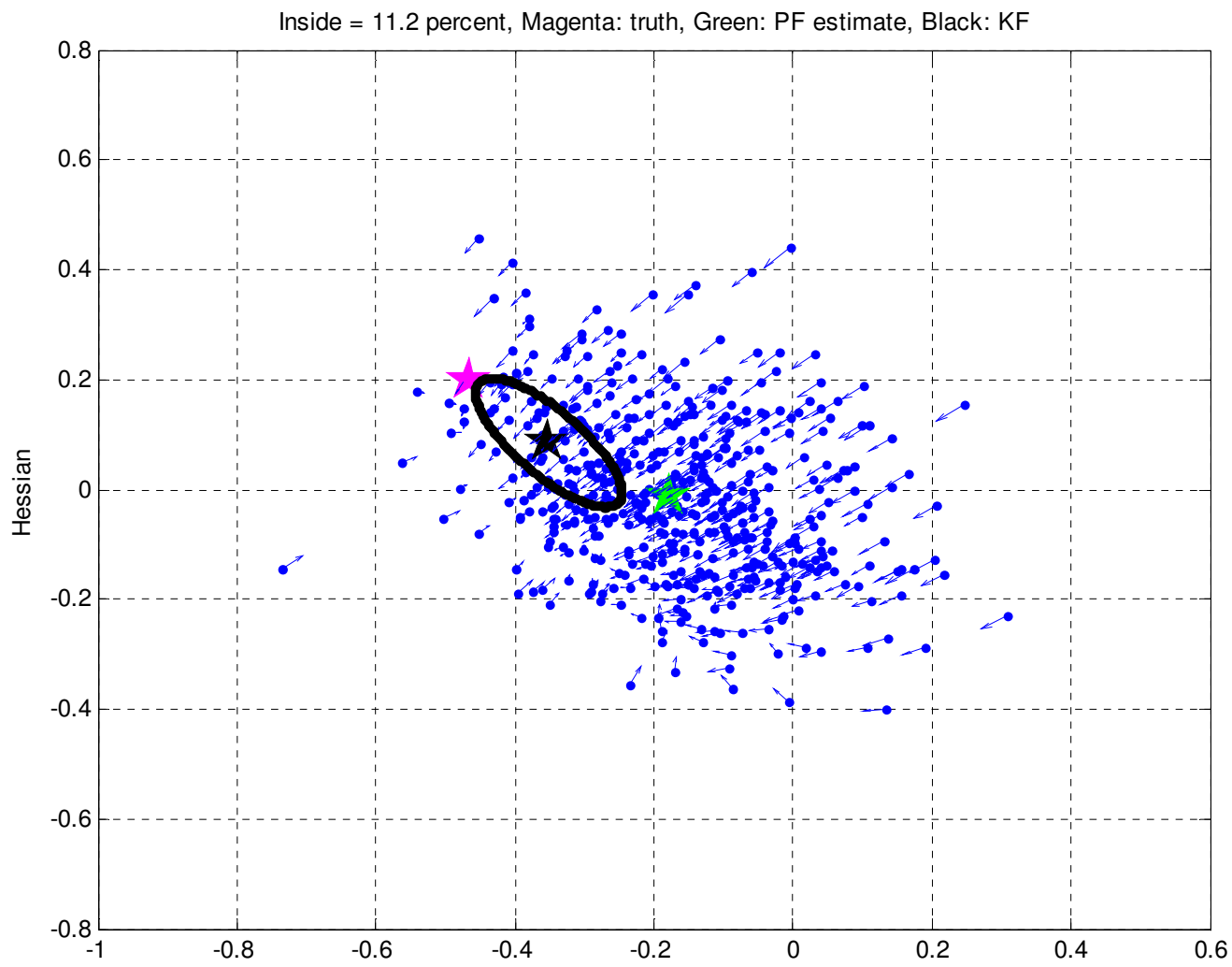
$$\frac{dx}{d\lambda} = 0 \quad \text{for zero gradient}$$

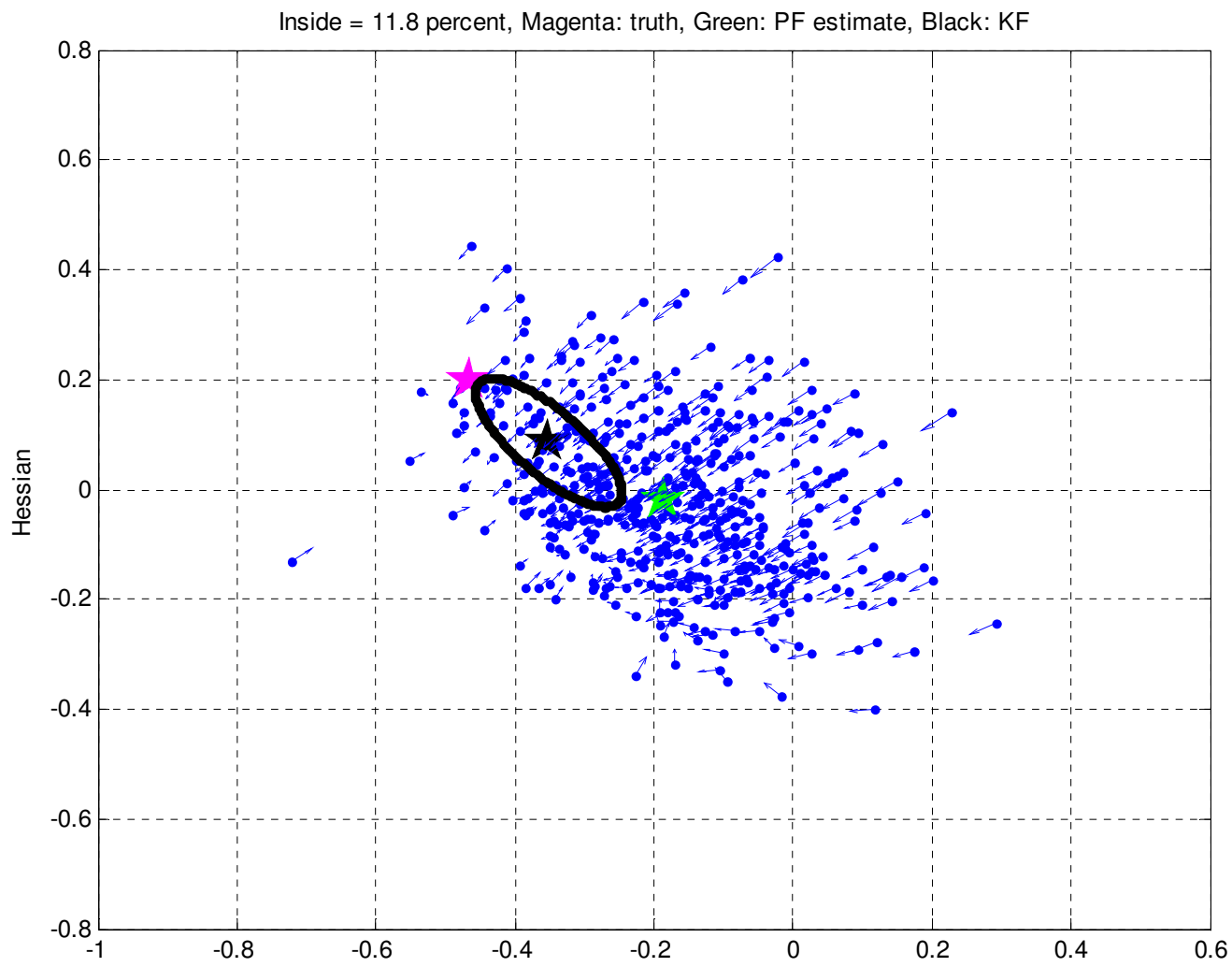

Inside = 6 percent, Magenta: truth, Green: PF estimate, Black: KF
Hessian
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6

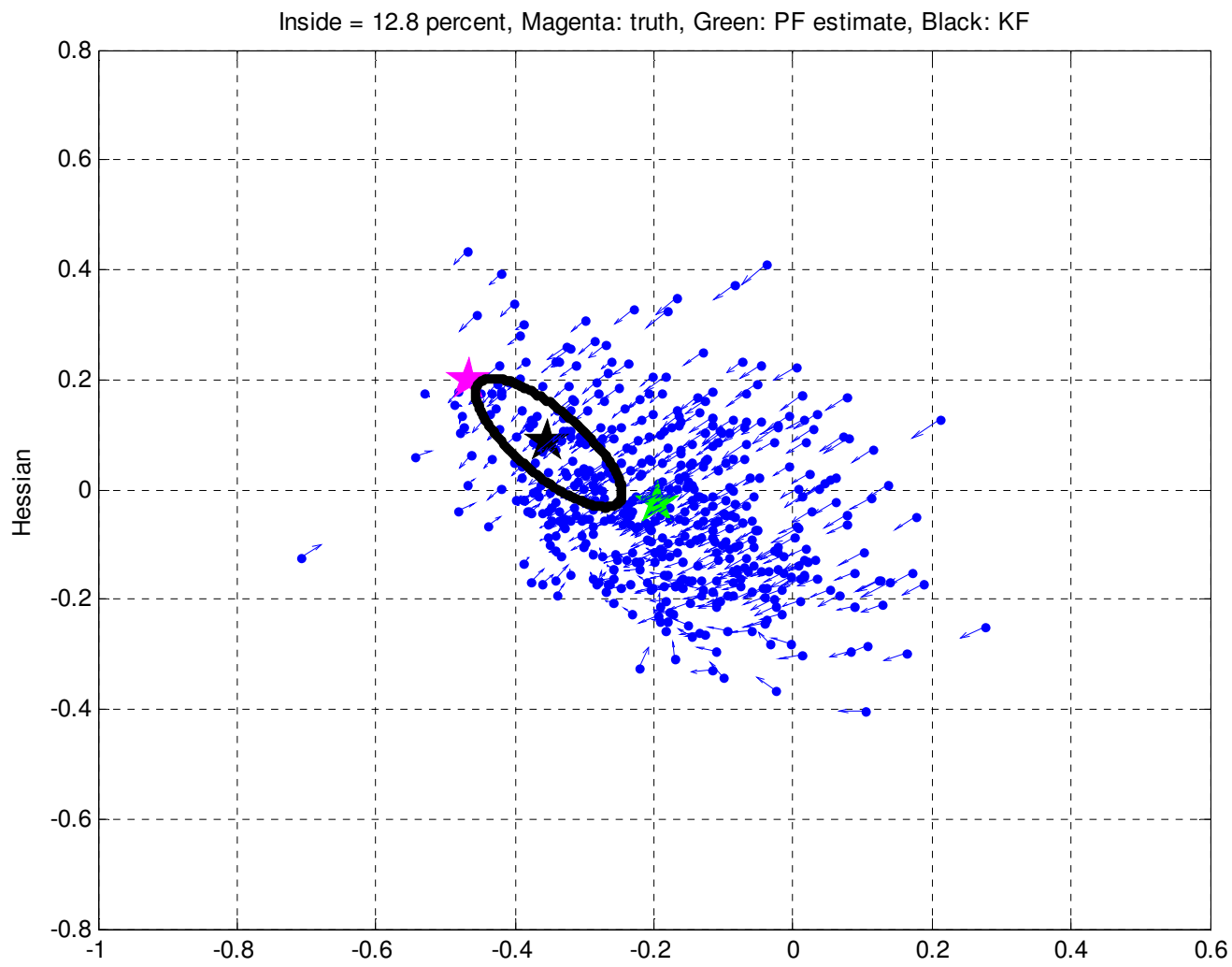

Inside = 5.8 percent, Magenta: truth, Green: PF estimate, Black: KF
Hessian
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6

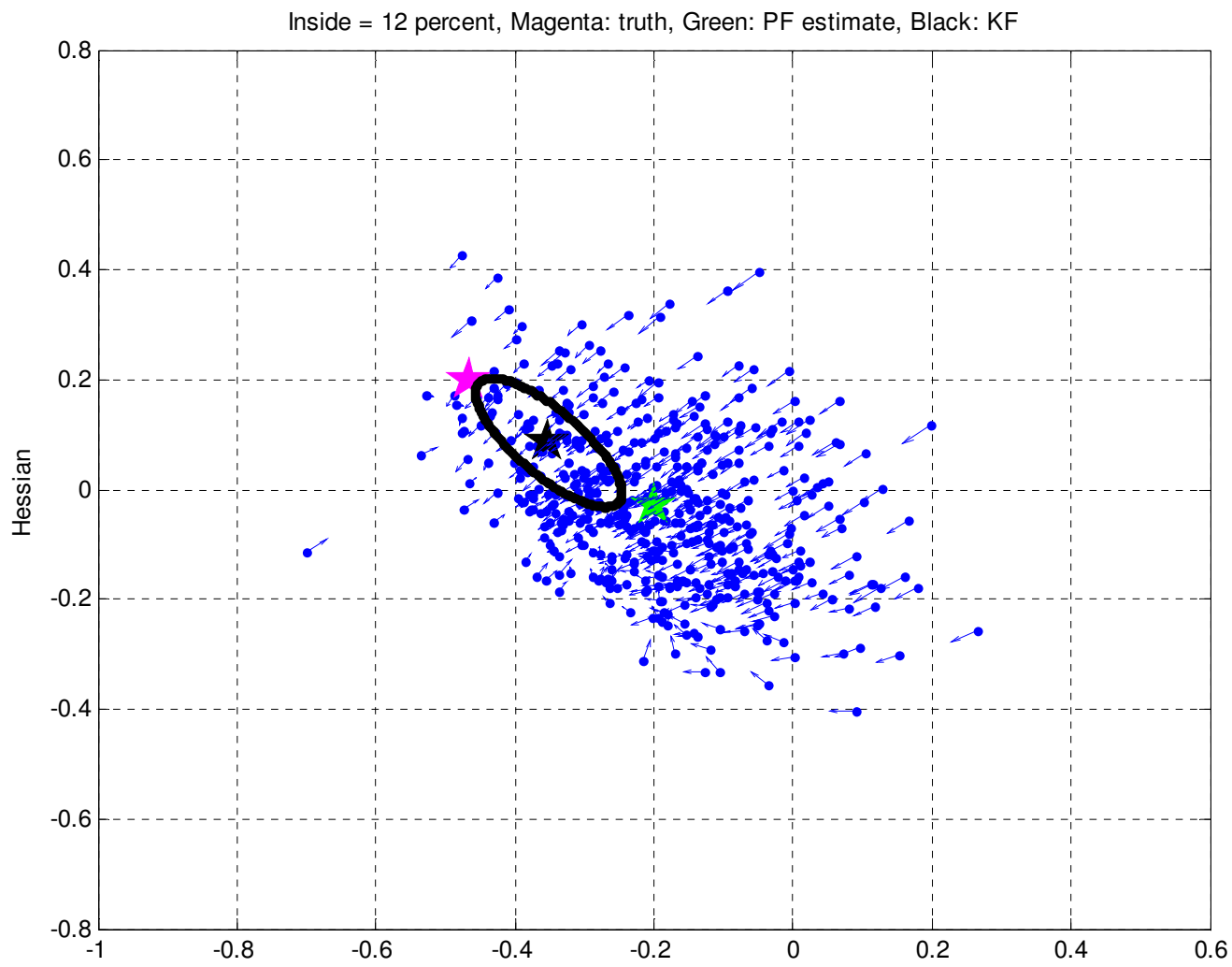

Inside = 8.2 percent, Magenta: truth, Green: PF estimate, Black: KF
Hessian
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6

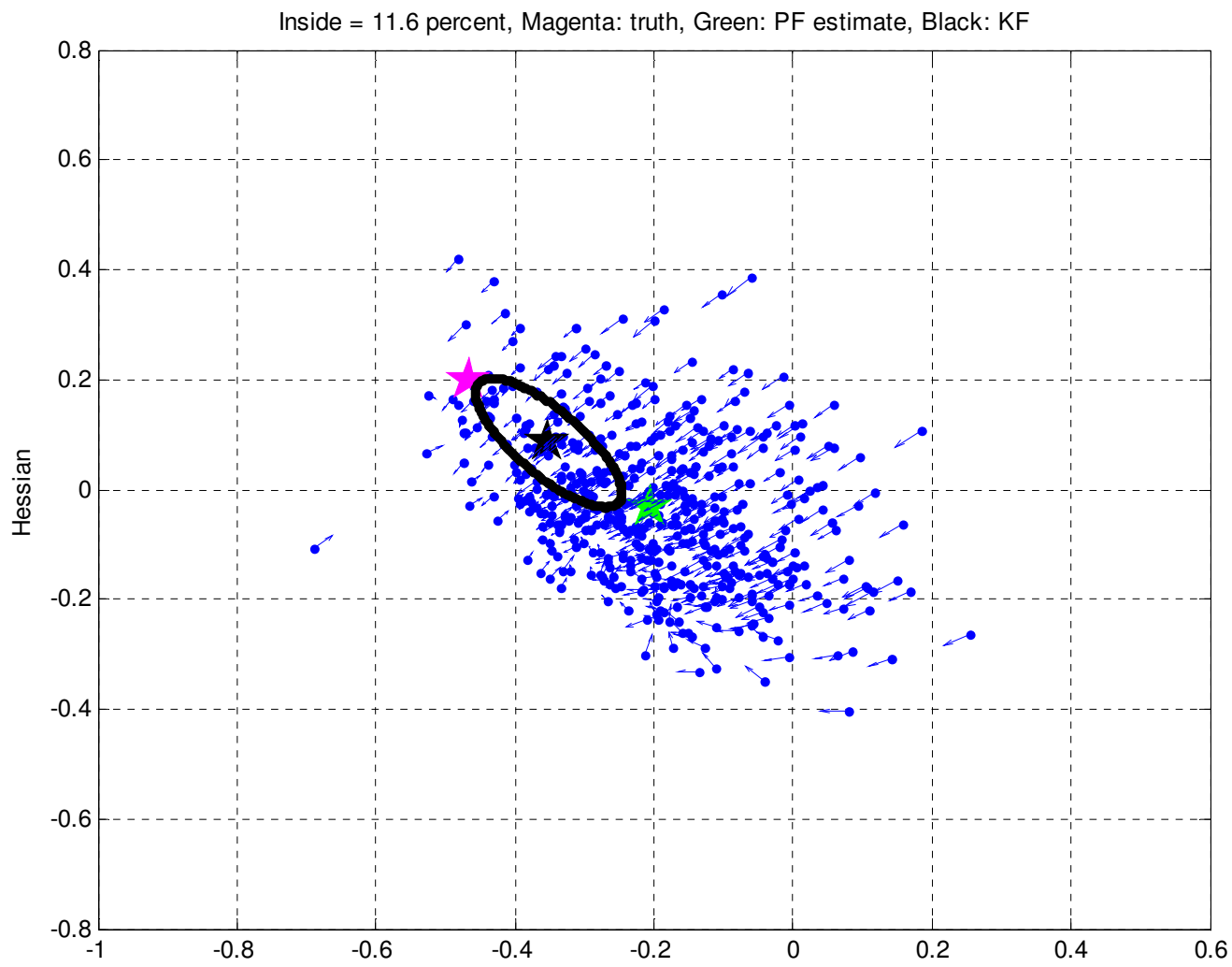

Inside = 9.2 percent, Magenta: truth, Green: PF estimate, Black: KF
Hessian
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6


Inside = 11.2 percent, Magenta: truth, Green: PF estimate, Black: KF
Hessian
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6

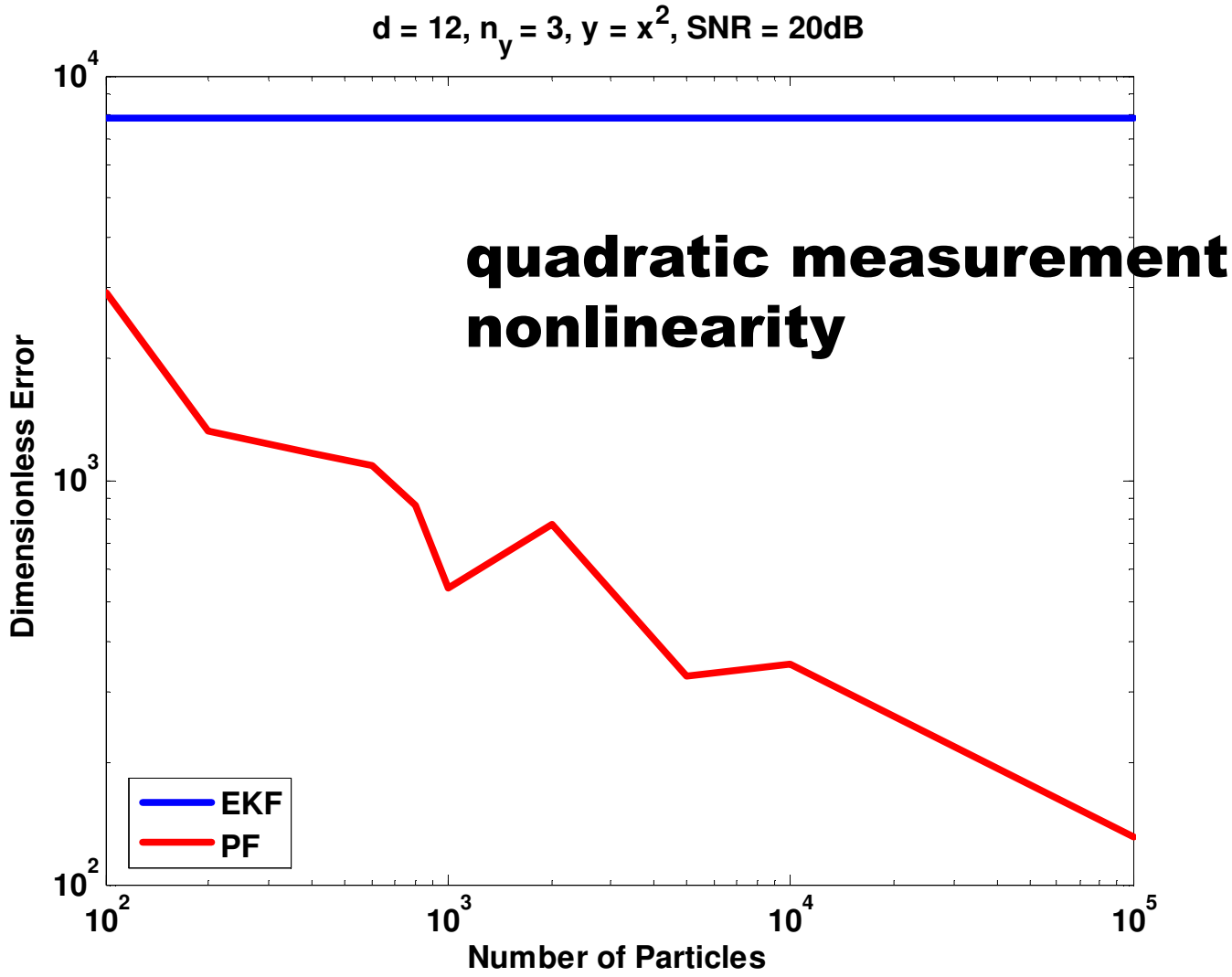

Inside = 11.8 percent, Magenta: truth, Green: PF estimate, Black: KF
Hessian
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6

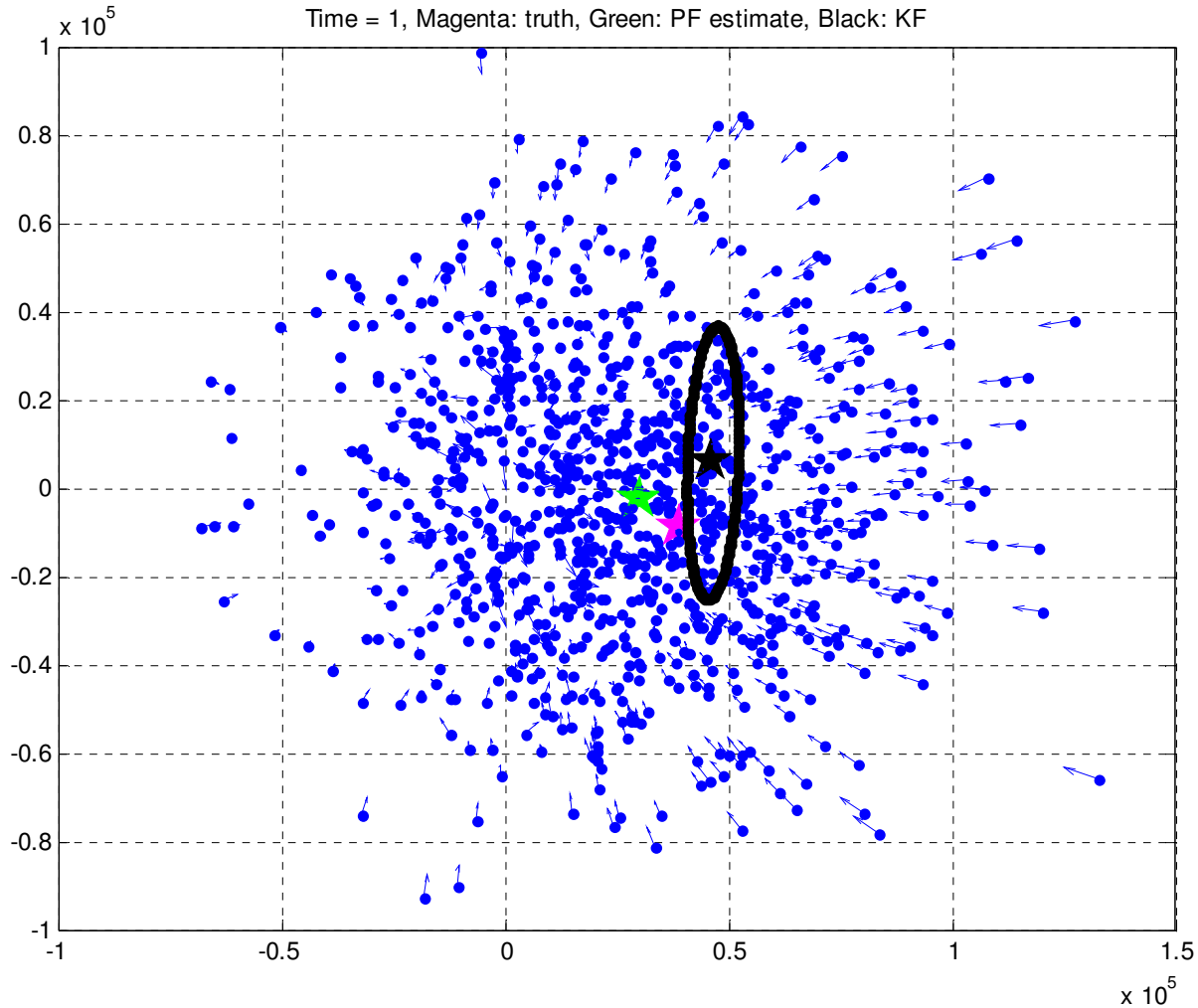

Inside = 12.8 percent, Magenta: truth, Green: PF estimate, Black: KF
Hessian
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6

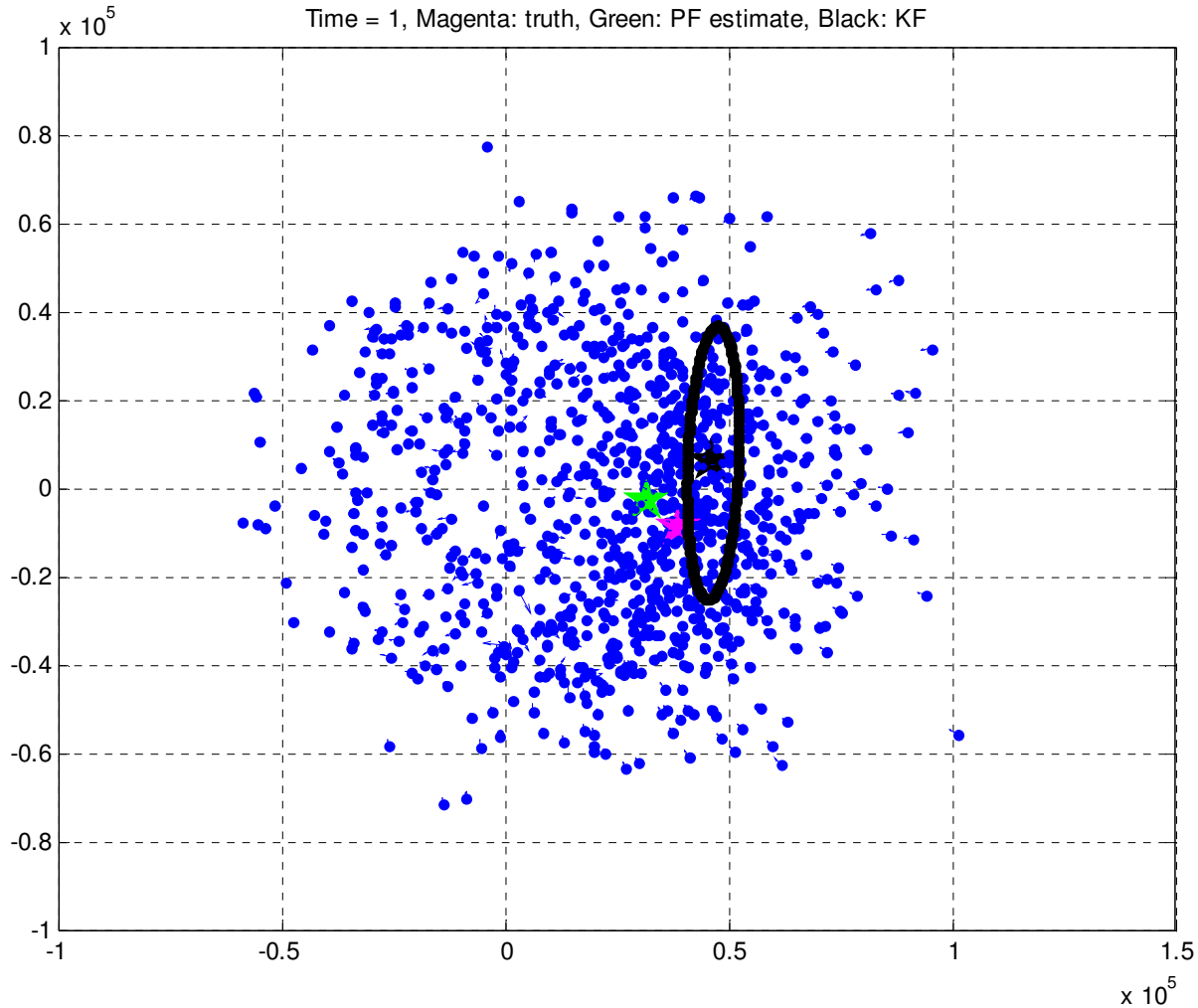

Inside = 12 percent, Magenta: truth, Green: PF estimate, Black: KF
Hessian
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6

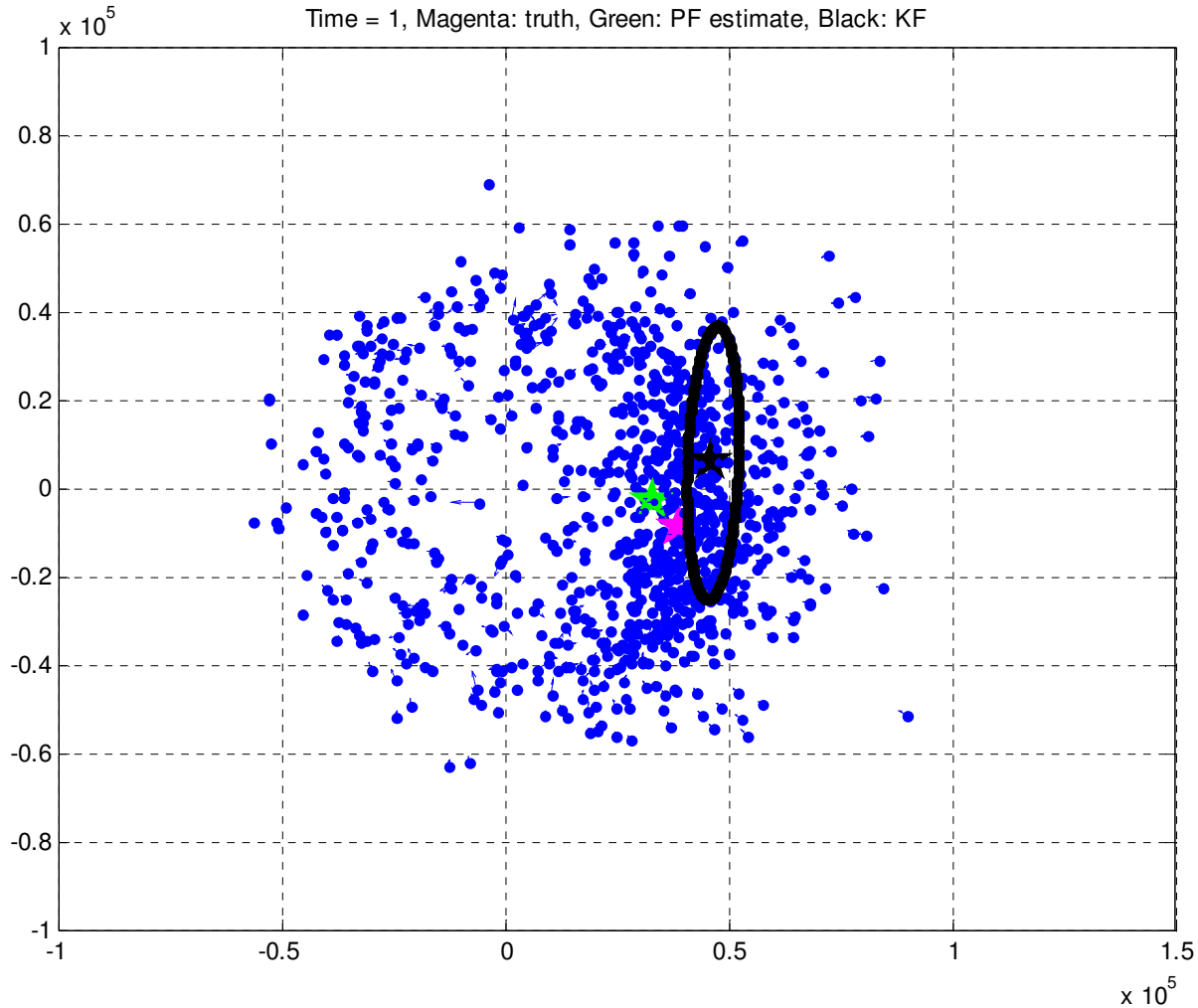

Inside = 11.6 percent, Magenta: truth, Green: PF estimate, Black: KF
Hessian
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6

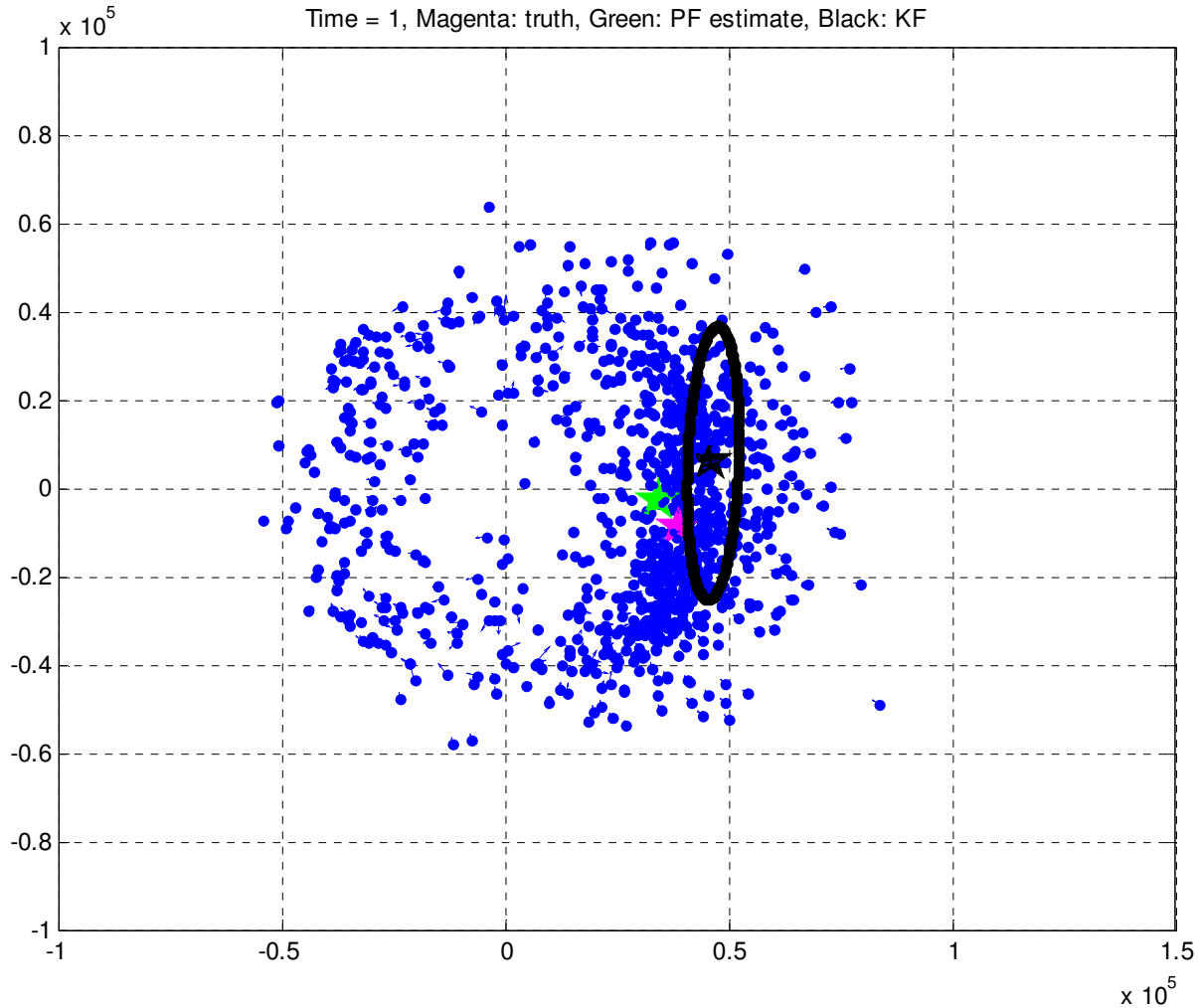
QUADRATIC MEASURE-MENTS

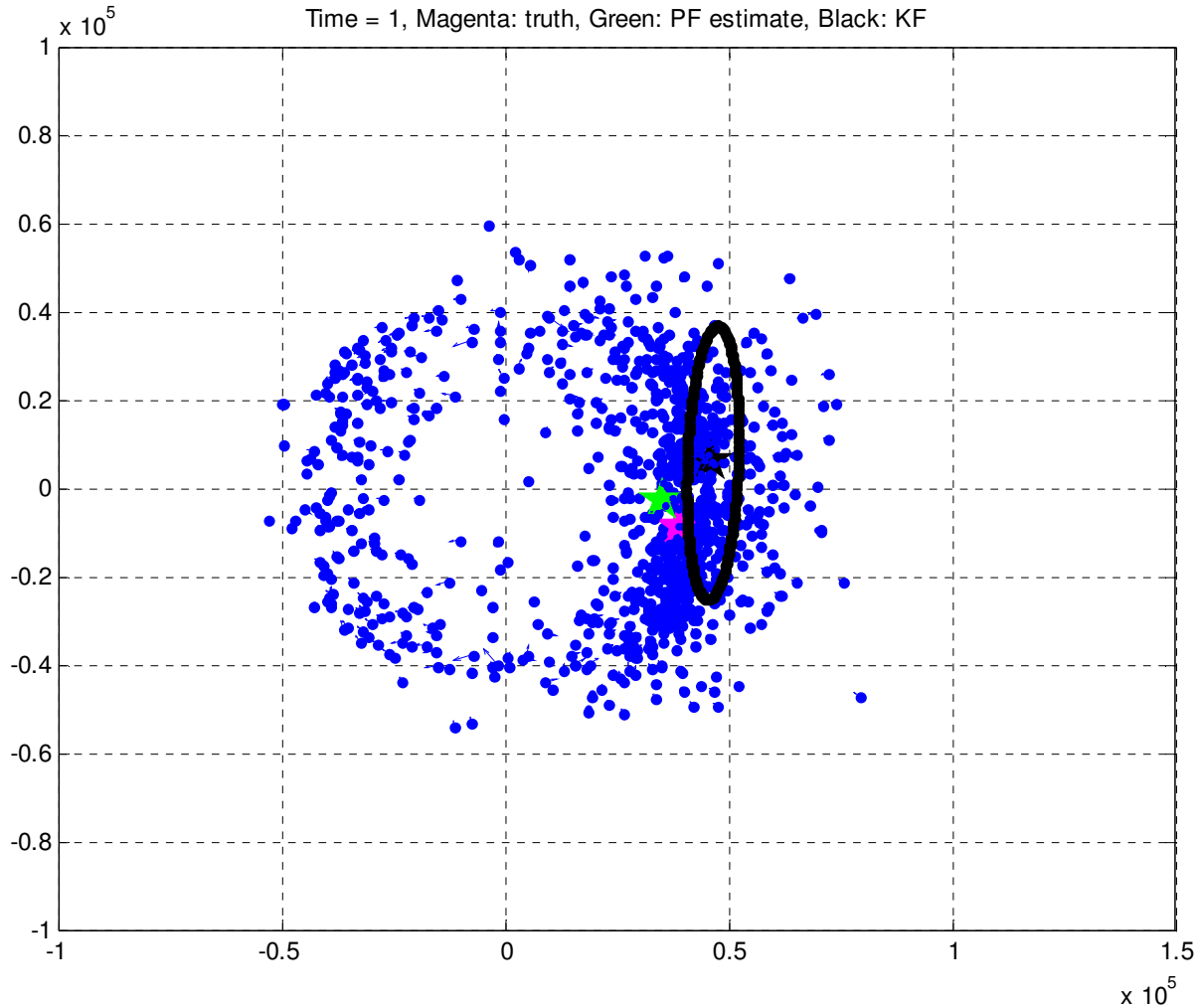

d = 12, n_y = 3, y = x^2, SNR = 20dB
quadratic measurement nonlinearity
Dimensionless Error
Number of Particles
EKF
PF
10^4
10^3
10^2
10^2
10^3
10^4
10^5

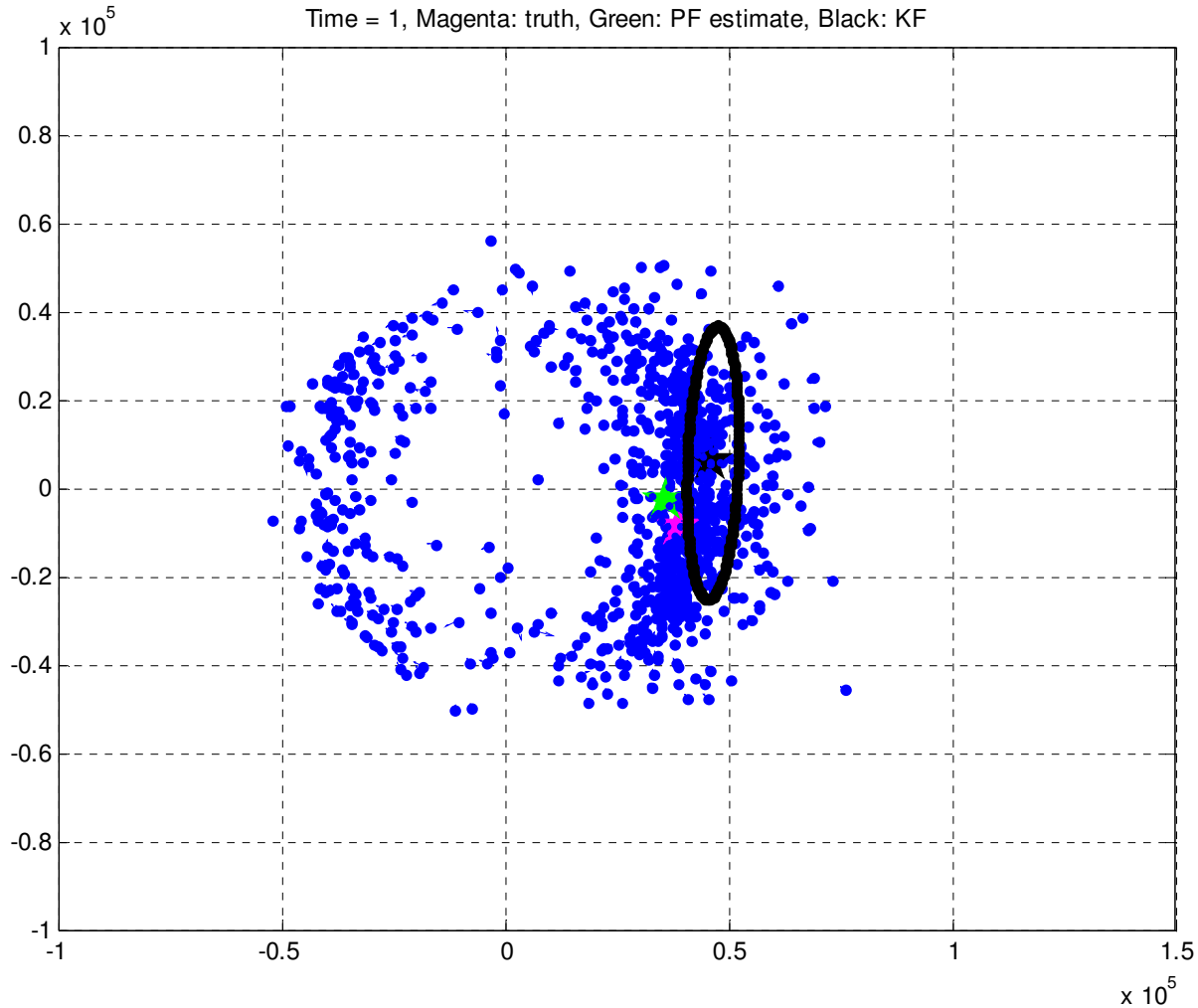

Time = 1, Magenta: truth, Green: PF estimate, Black: KF
x 10^5
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1
-0.5
0
0.5
1
1.5
x 10^5

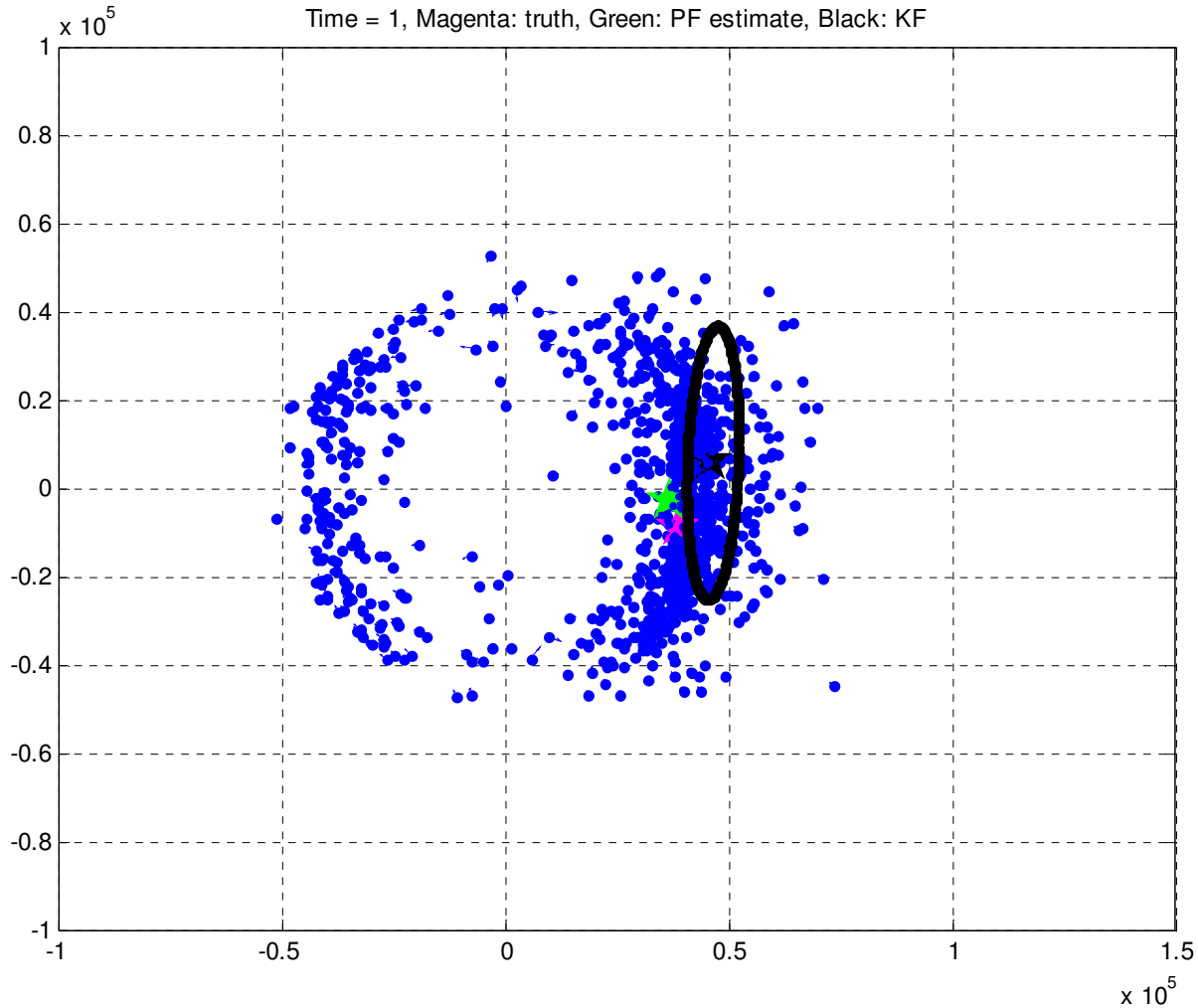

Time = 1, Magenta: truth, Green: PF estimate, Black: KF
x 10^5
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1
-0.5
0
0.5
1
1.5
x 10^5

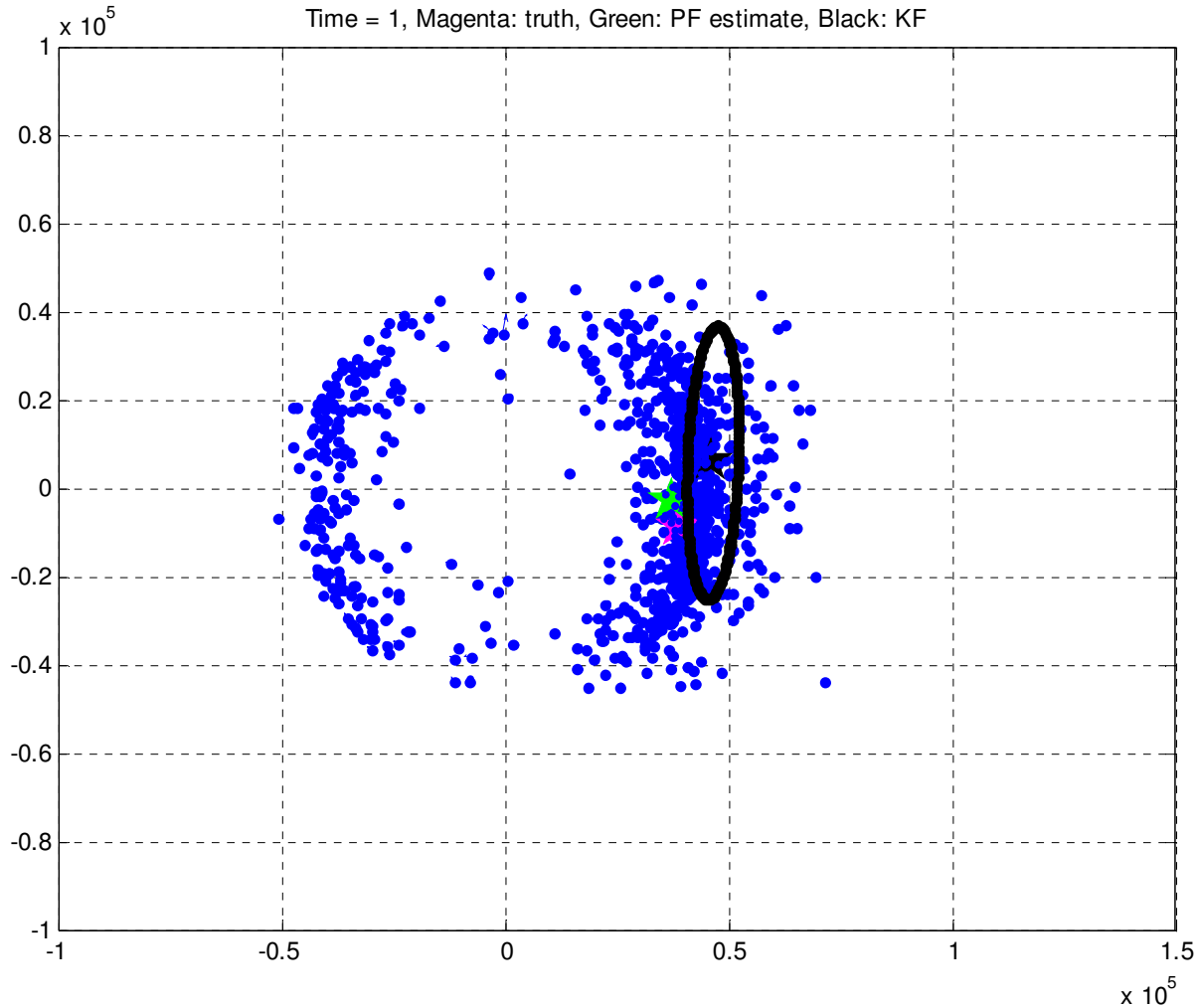

Time = 1, Magenta: truth, Green: PF estimate, Black: KF
x 10^5
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1
-0.5
0
0.5
1
1.5
x 10^5

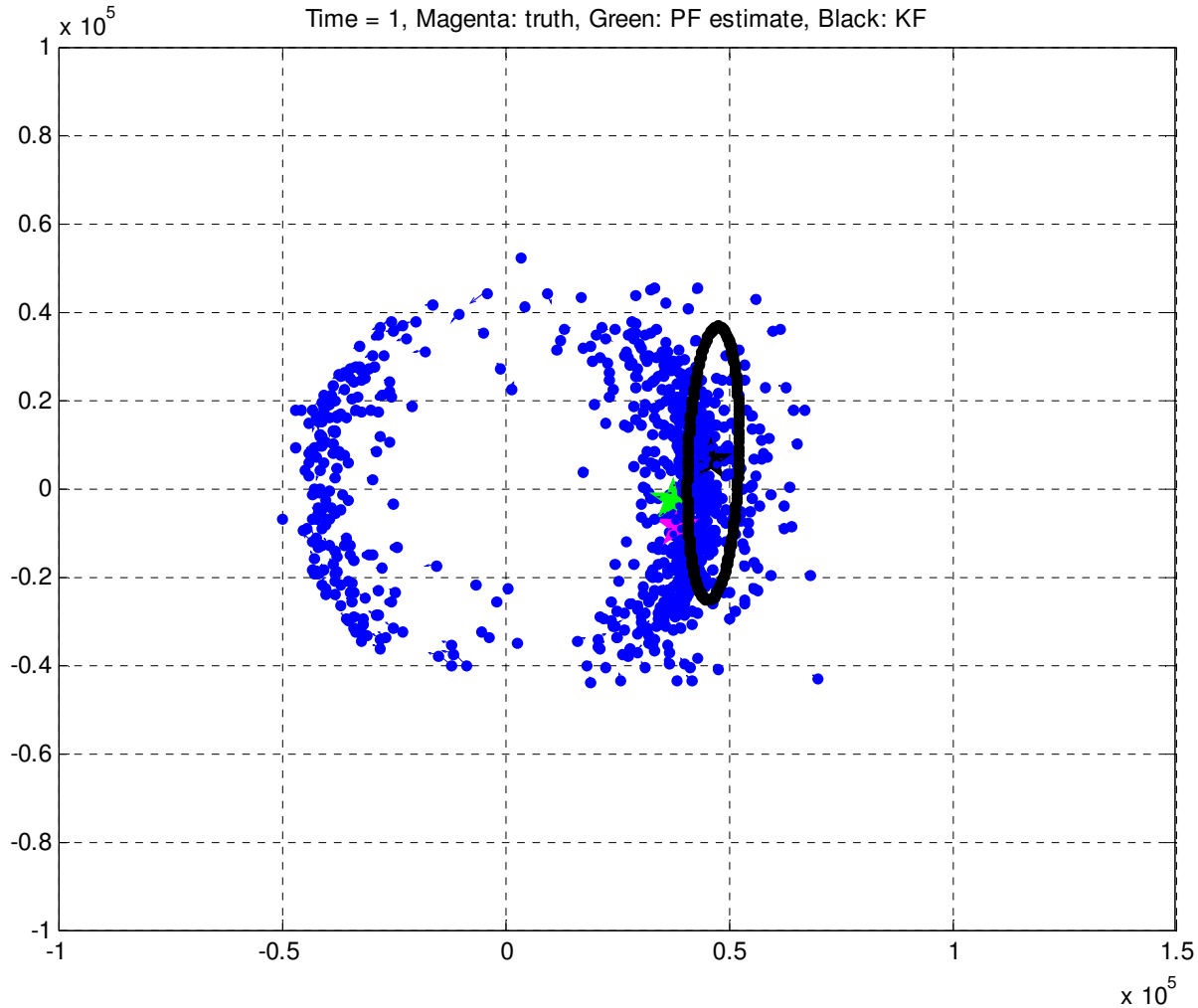

Time = 1, Magenta: truth, Green: PF estimate, Black: KF
x 10^5
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1
-0.5
0
0.5
1
1.5
x 10^5

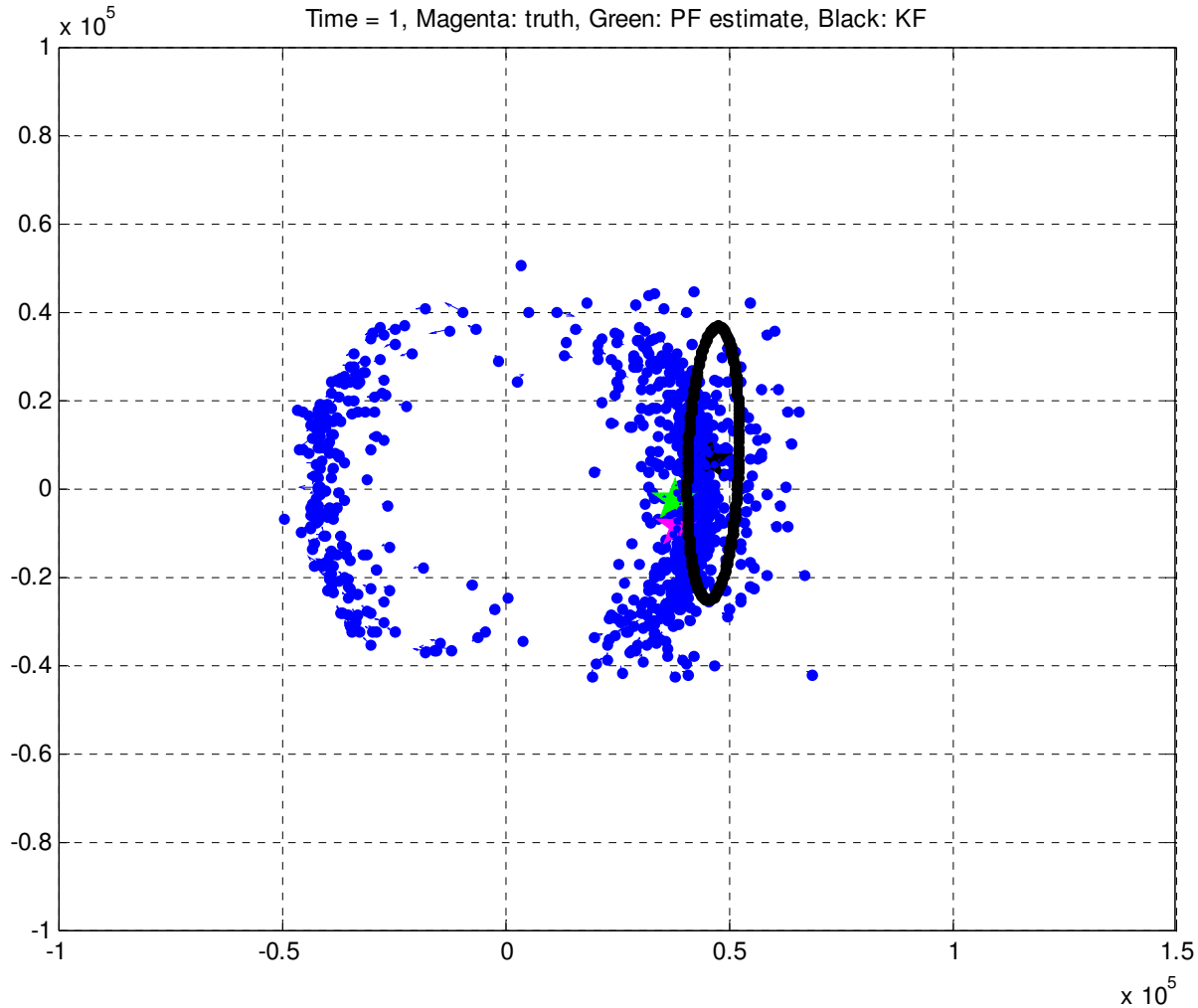

Time = 1, Magenta: truth, Green: PF estimate, Black: KF
x 10^5
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1
-0.5
0
0.5
1
1.5
x 10^5


Time = 1, Magenta: truth, Green: PF estimate, Black: KF
x 10^5
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1
-0.5
0
0.5
1
1.5
x 10^5

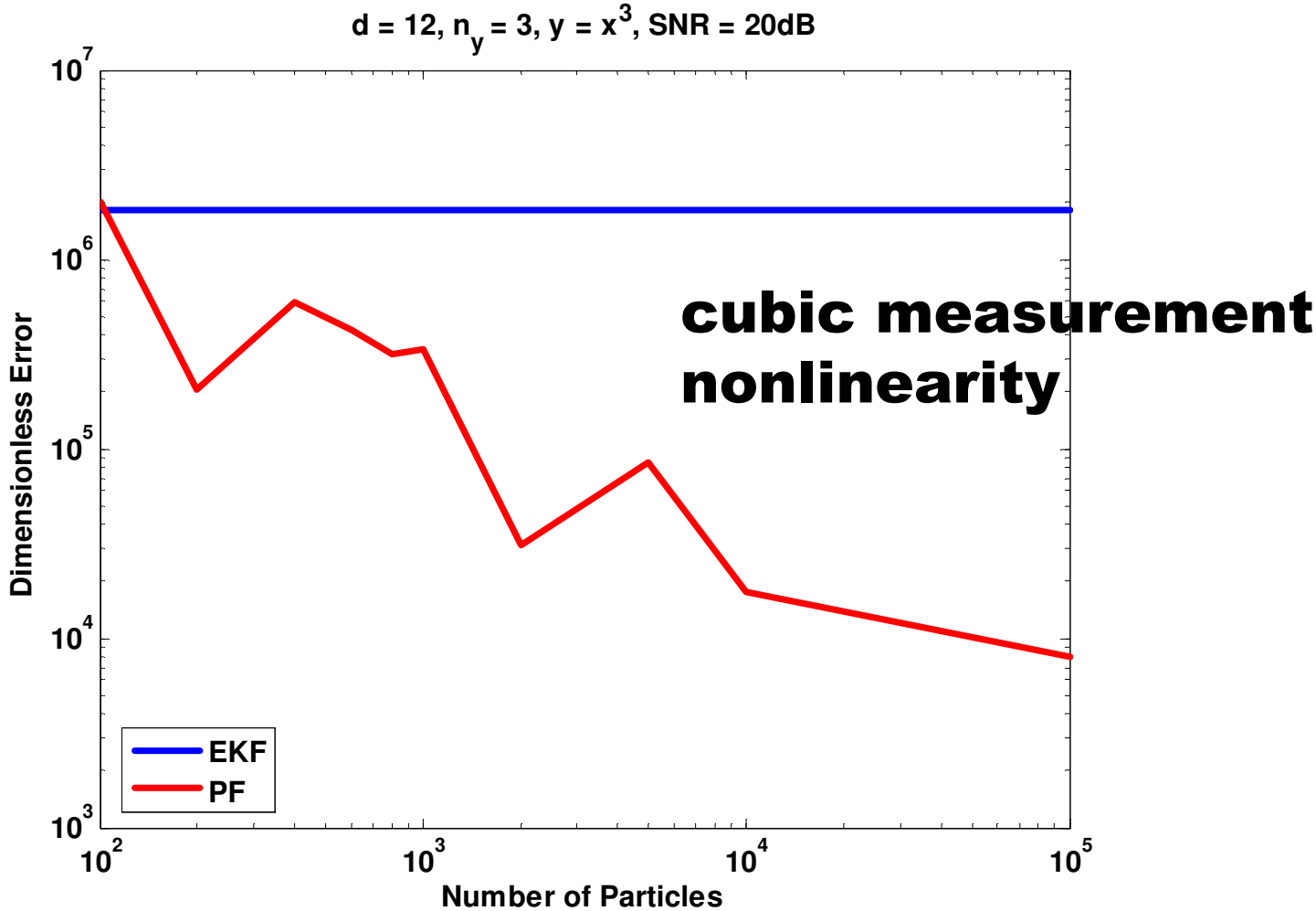

Time = 1, Magenta: truth, Green: PF estimate, Black: KF
x 10^5
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1
-0.5
0
0.5
1
1.5
x 10^5

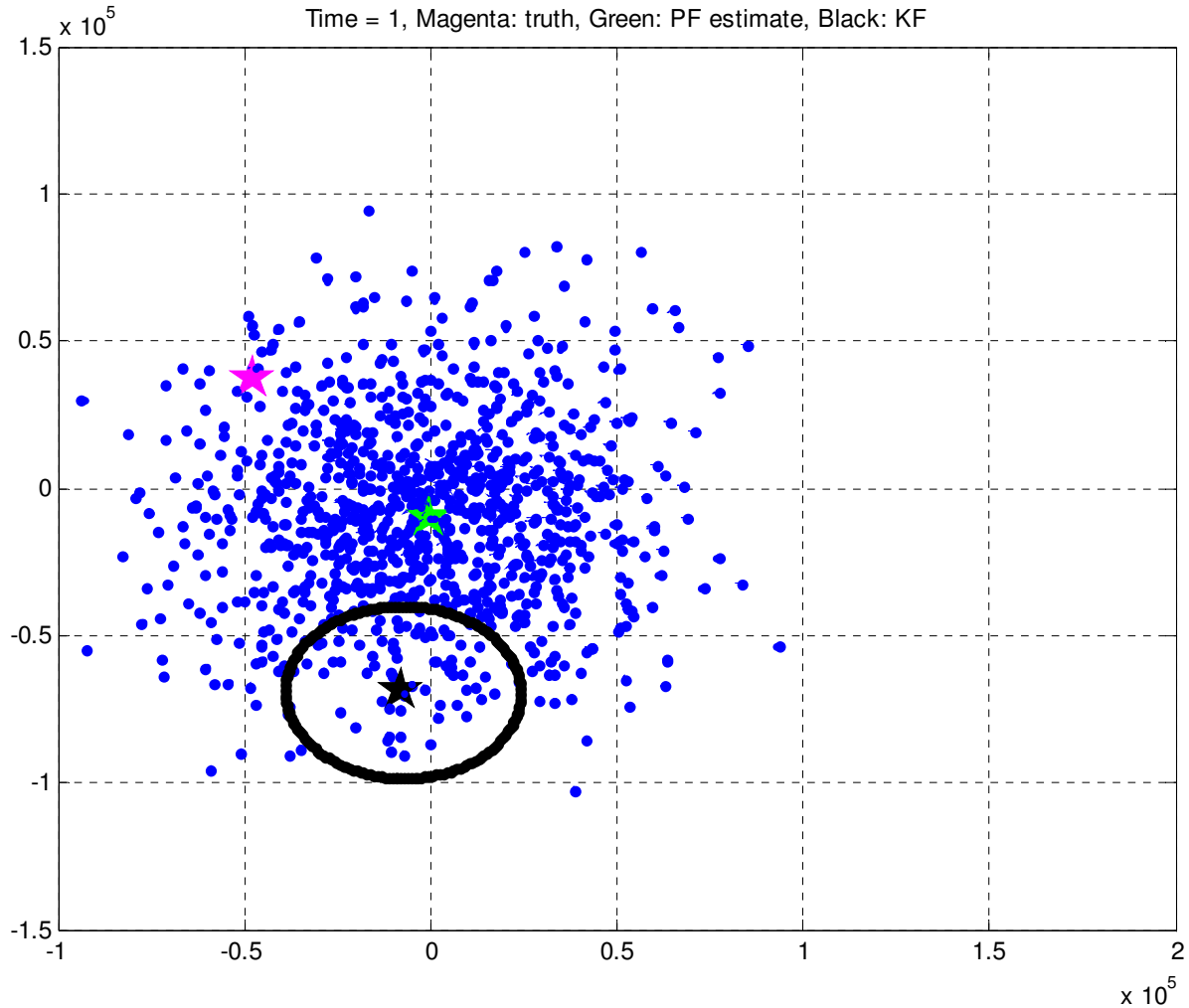

Time = 1, Magenta: truth, Green: PF estimate, Black: KF
x 10^5
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1
-0.5
0
0.5
1
1.5
x 10^5

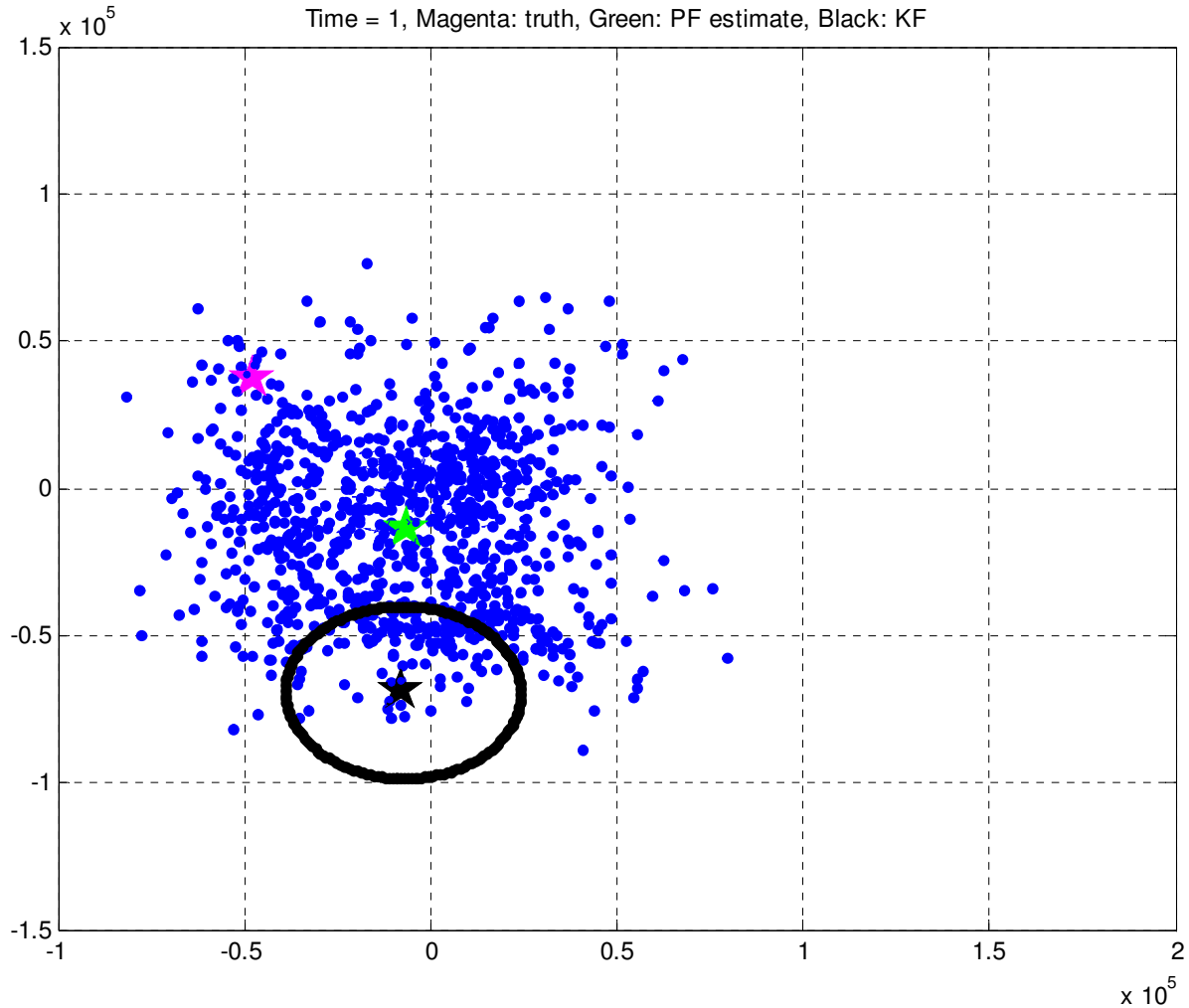

Time = 1, Magenta: truth, Green: PF estimate, Black: KF
x 10^5
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1
-0.5
0
0.5
1
1.5
x 10^5

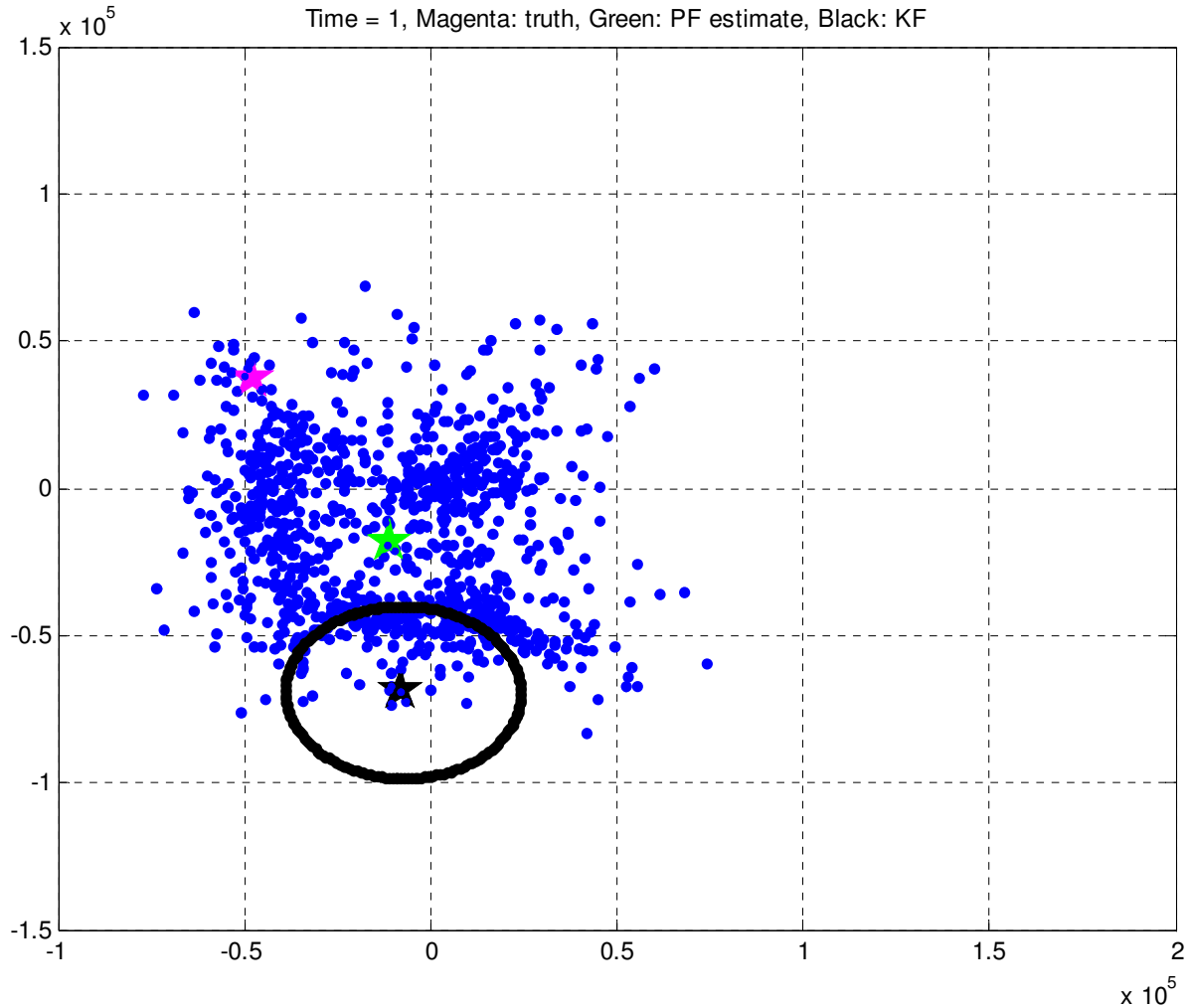

Time = 1, Magenta: truth, Green: PF estimate, Black: KF
x 10^5
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1
-0.5
0
0.5
1
1.5
x 10^5

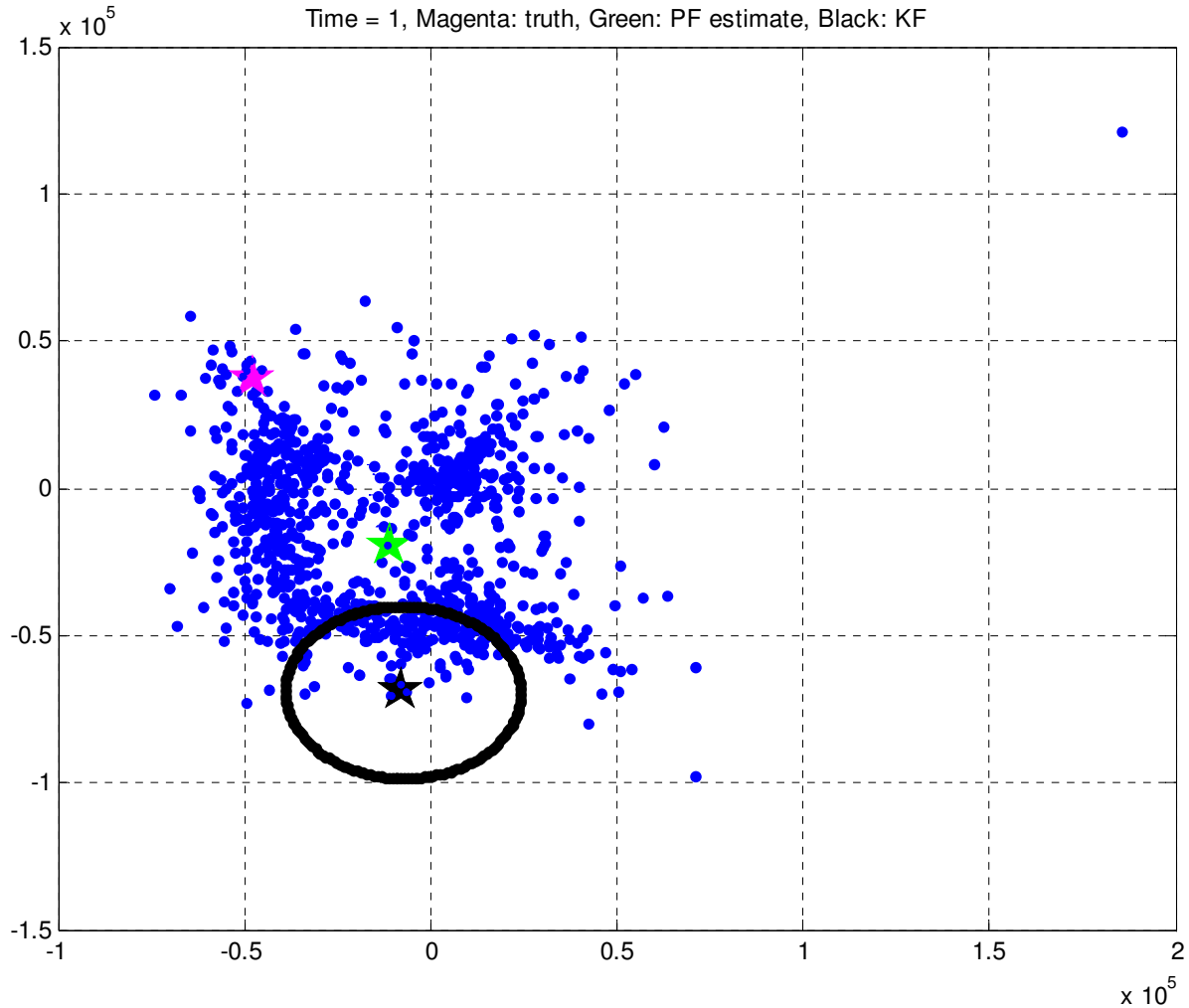
CUBIC MEASURE-MENTS

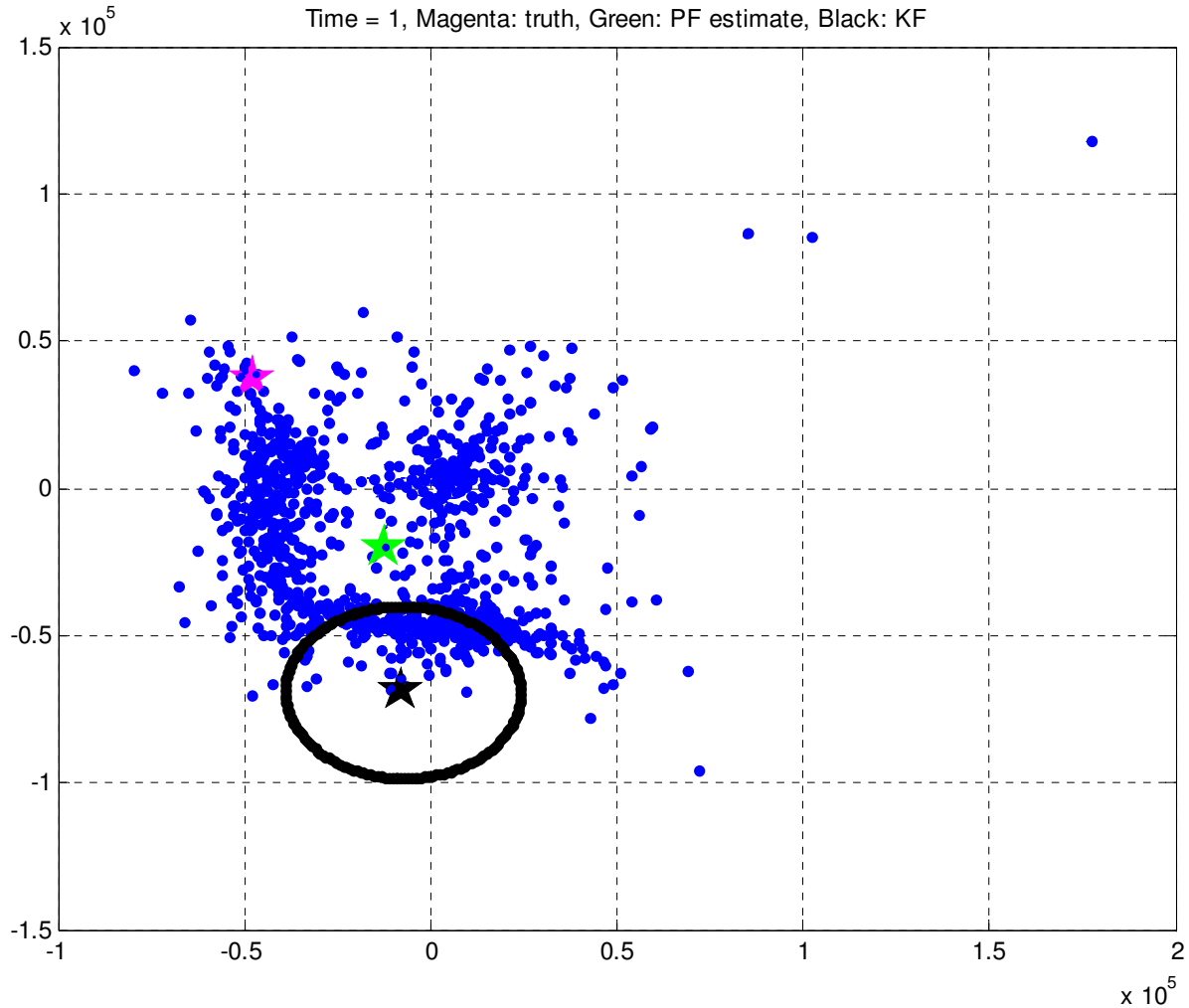

d = 12, n_y = 3, y = x^3, SNR = 20dB
Dimensionless Error
Number of Particles
EKF
PF
cubic measurement nonlinearity

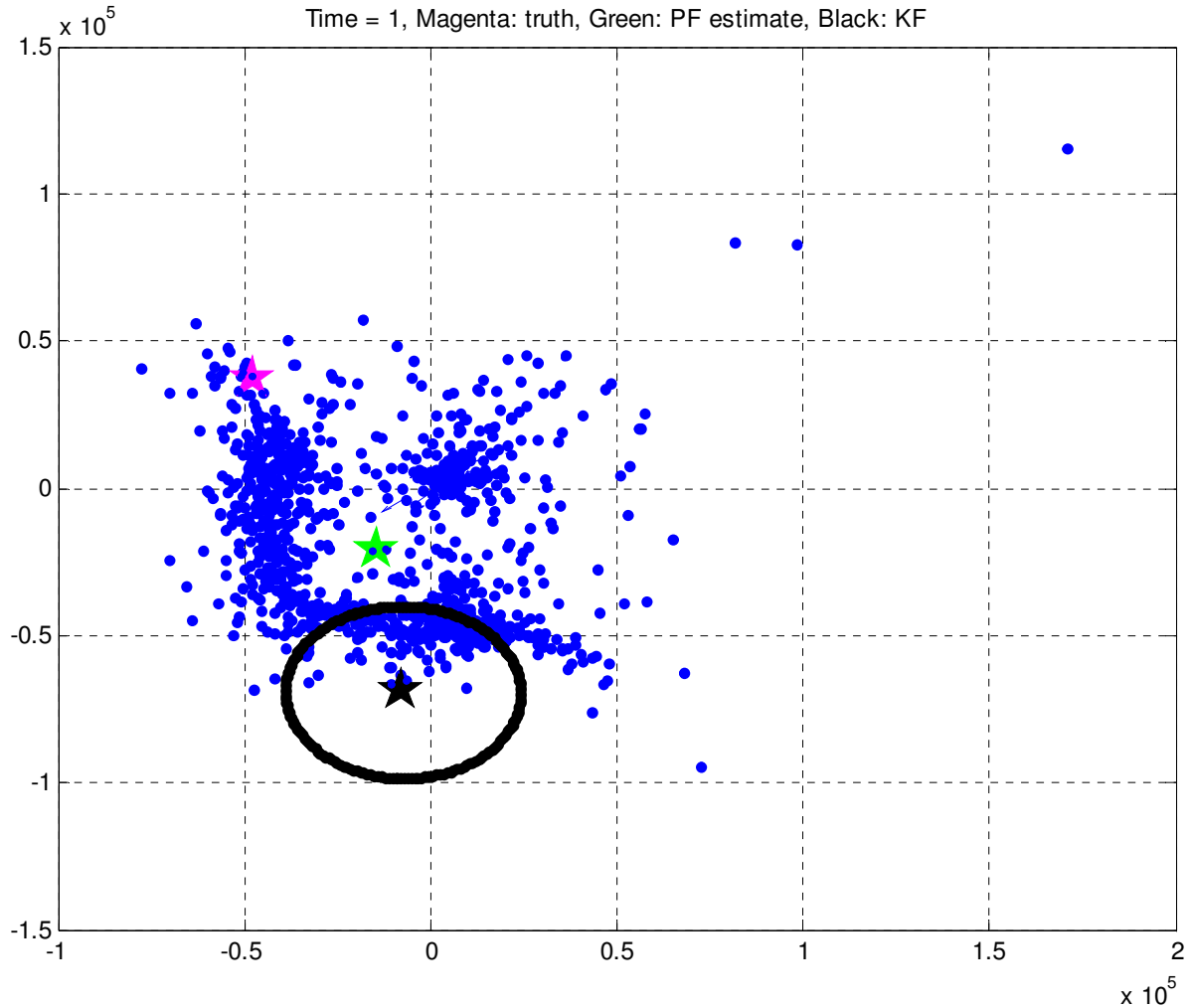

Time = 1, Magenta: truth, Green: PF estimate, Black: KF
x 10^5
1.5
1
0.5
0
-0.5
-1
-1.5
-1
-0.5
0
0.5
1
1.5
2
x 10^5

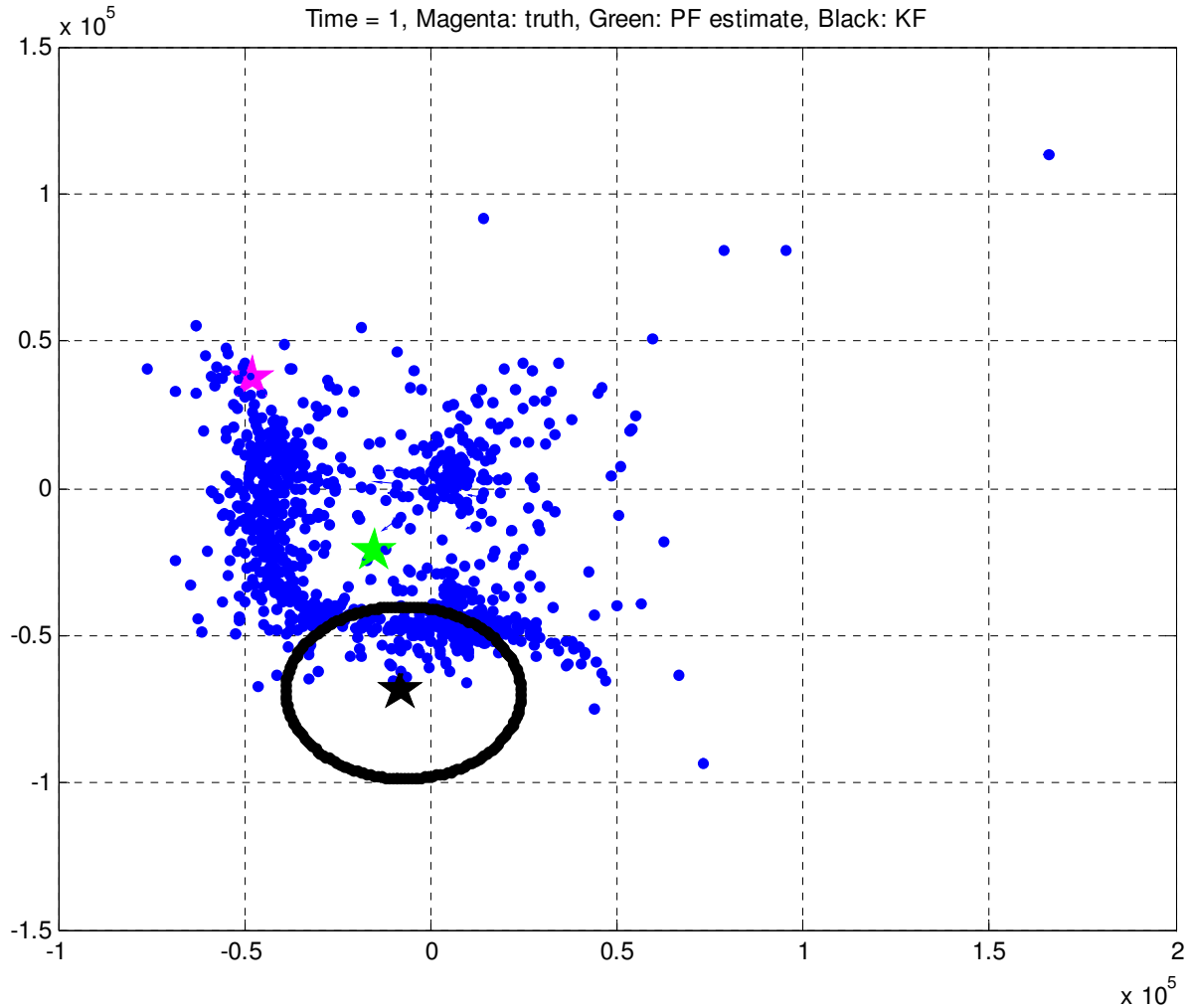

Time = 1, Magenta: truth, Green: PF estimate, Black: KF
x 10^5
1.5
1
0.5
0
-0.5
-1
-1.5
-1
-0.5
0
0.5
1
1.5
2
x 10^5

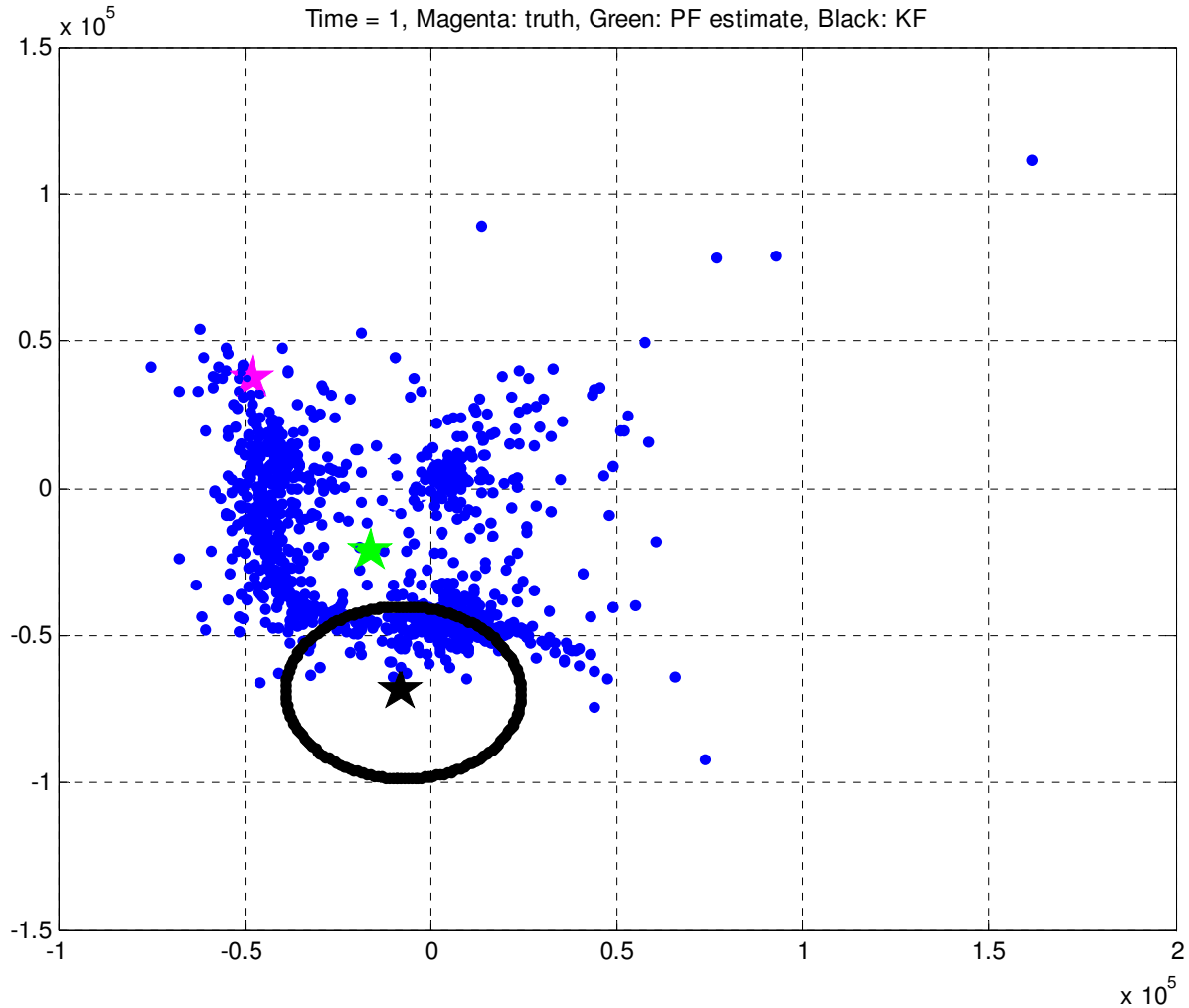

Time = 1, Magenta: truth, Green: PF estimate, Black: KF
x 10^5
1.5
1
0.5
0
-0.5
-1
-1.5
-1
-0.5
0
0.5
1
1.5
2
x 10^5

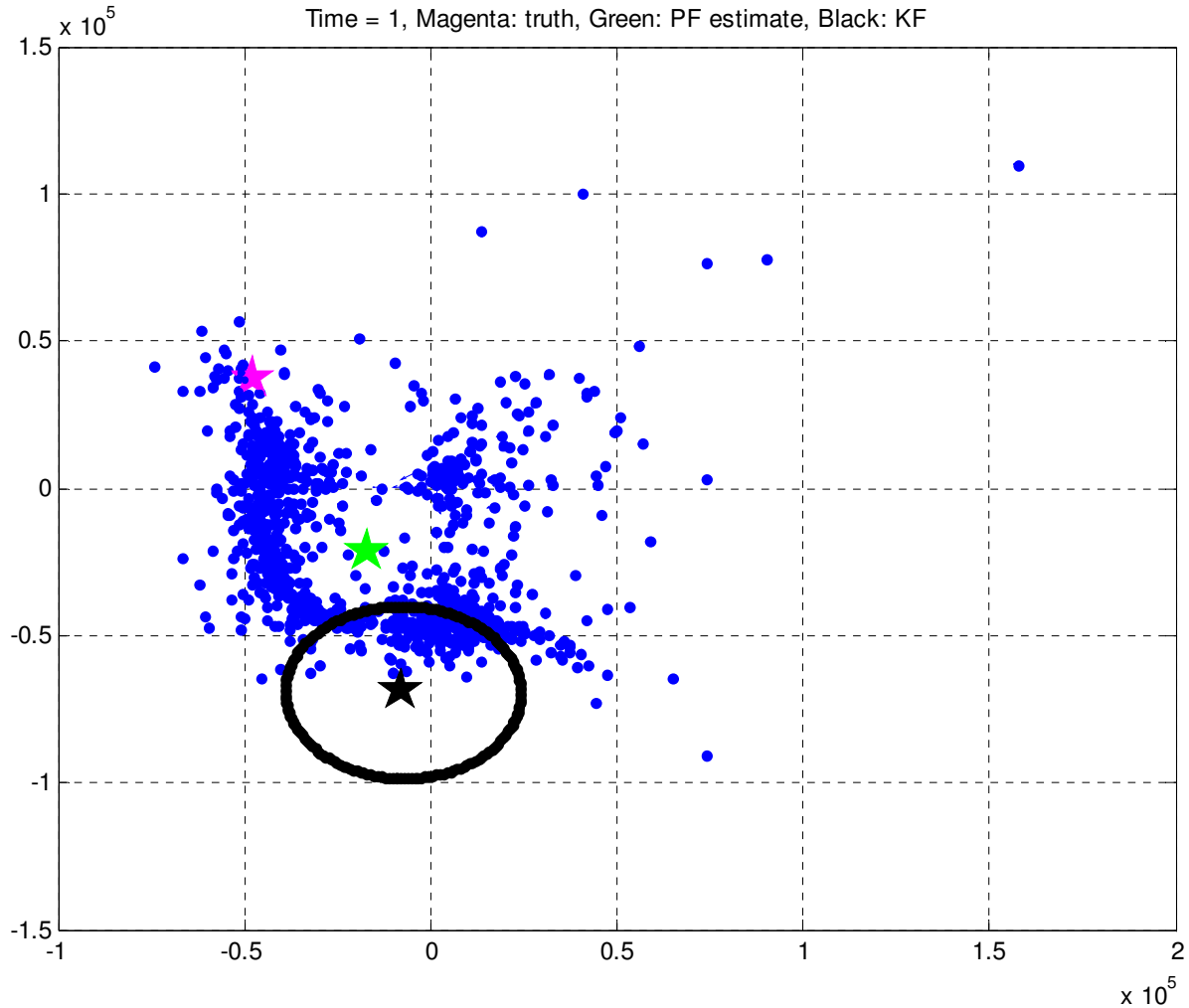

Time = 1, Magenta: truth, Green: PF estimate, Black: KF
x 10^5
1.5
1
0.5
0
-0.5
-1
-1.5
-1
-0.5
0
0.5
1
1.5
2
x 10^5

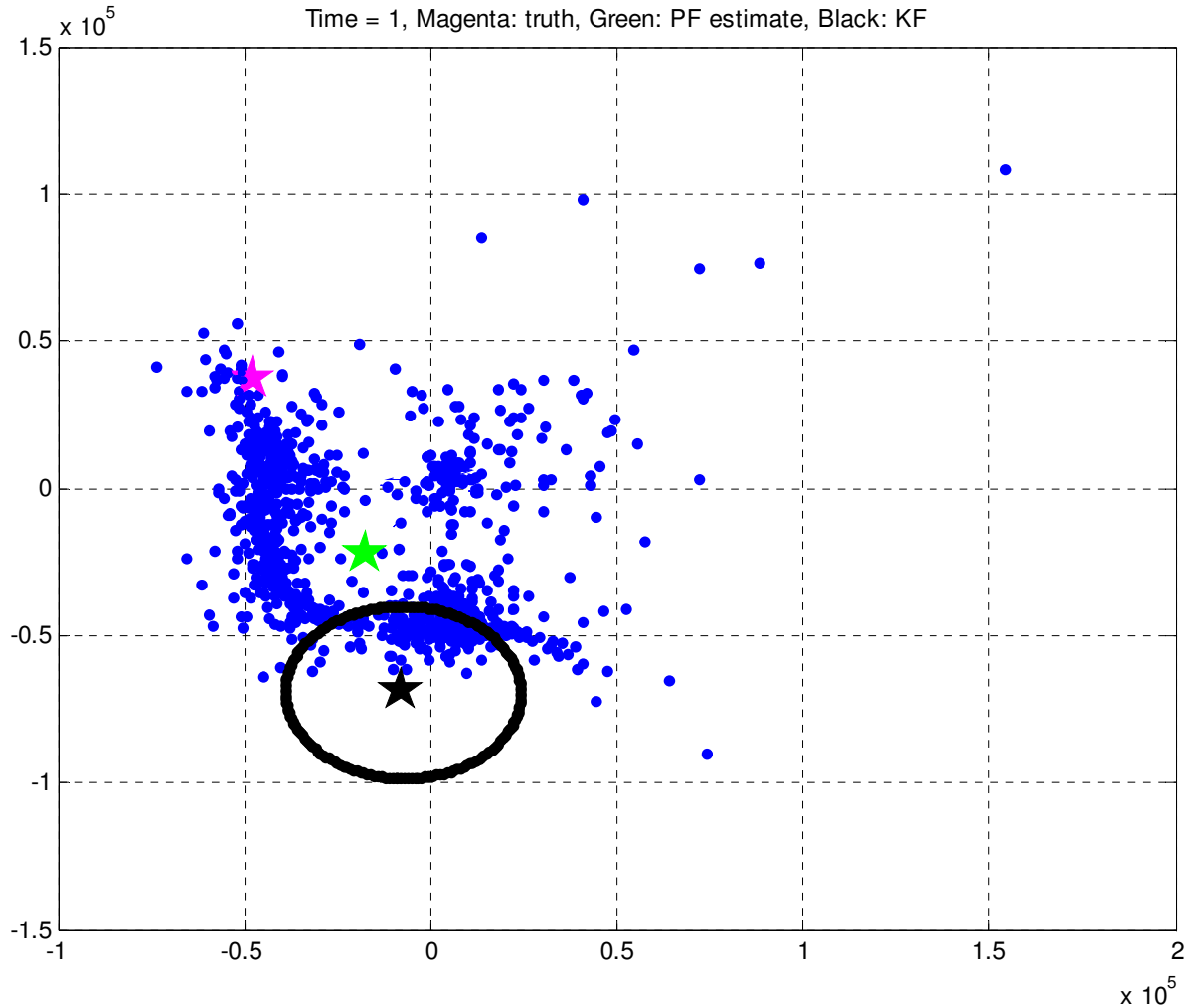

Time = 1, Magenta: truth, Green: PF estimate, Black: KF
x 10^5
1.5
1
0.5
0
-0.5
-1
-1.5
-1
-0.5
0
0.5
1
1.5
2
x 10^5


Time = 1, Magenta: truth, Green: PF estimate, Black: KF
x 10^5
1.5
1
0.5
0
-0.5
-1
-1.5
-1
-0.5
0
0.5
1
1.5
2
x 10^5


Time = 1, Magenta: truth, Green: PF estimate, Black: KF
x 10^5
1.5
1
0.5
0
-0.5
-1
-1.5
-1
-0.5
0
0.5
1
1.5
2
x 10^5


Time = 1, Magenta: truth, Green: PF estimate, Black: KF
x 10^5
1.5
1
0.5
0
-0.5
-1
-1.5
-1
-0.5
0
0.5
1
1.5
2
x 10^5


Time = 1, Magenta: truth, Green: PF estimate, Black: KF
x 10^5
1.5
1
0.5
0
-0.5
-1
-1.5
-1
-0.5
0
0.5
1
1.5
2
x 10^5


Time = 1, Magenta: truth, Green: PF estimate, Black: KF
x 10^5
1.5
1
0.5
0
-0.5
-1
-1.5
-1
-0.5
0
0.5
1
1.5
2
x 10^5

STABILITY

Stability of nonlinear filters

stability of Kalman filter (1963 paper by Kalman)	Kalman filter is stable under very mild conditions (e.g., controllability & observability); Kalman filter is stable for unstable plants*.
stability of extended Kalman filters	EKF is unstable for unstable plants for typical nonlinear examples**
convergence of particle filters	assumes ergodicity, which implies stability of plant
papers by Ramon van Handel & Dan Crisan (2009)	makes extremely strong assumptions (the complete state vector is measured)
incompressible flow of particles	explicitly stabilize filter & flow for unstable plants without measuring complete state vector
exact (compressible) particle flow	automatically stable filter & flow

most general solution for flow:

$$\frac{dx}{d\lambda} = ?$$

by the chain rule :

$$\frac{d \log p(x,\lambda)}{d\lambda} = \frac{\partial \log p(x,\lambda)}{\partial x}\frac{dx}{d\lambda} + \frac{\partial \log p(x,\lambda)}{\partial \lambda} = 0$$

$$\frac{dx}{d\lambda} = -A^{\#}\frac{\partial \log p(x,\lambda)}{\partial \lambda} + [I - A^T A / Tr(A^T A)]y$$

in which $\mathrm{A} = \dfrac{\partial \log p(\mathrm{x}, \lambda)}{\partial \mathrm{x}}$

y = arbitrary d - vector

$\mathrm{A}^{\#} = A^T / (\mathrm{AA}^{\mathrm{T}})$ for $\mathrm{AA}^{\mathrm{T}} > 0$,

and $\mathrm{A}^{\#} = 0$ otherwise.

pick y to maximize stability of filter

Pick y to maximize stability of filter:

$$\frac{dx}{d\lambda} = -A^{\#}\frac{\partial \log p(x,\lambda)}{\partial \lambda} + [I - A^T A / Tr(A^T A)]y$$

Most general solution for flow

For example, linearize about each particle

$$\frac{\mathrm{dx}}{\mathrm{d}\lambda} \approx Bx + Dy \text{ with } \mathrm{y} = \mathrm{Kx}$$

$$\Phi_{filter} \approx \Phi_{plant}\Phi_{\mathrm{Bayes}} \text{ where } \Phi_{Bayes} \approx \exp(\mathrm{B} + \mathrm{DK})$$

pick K to minimize the following stability measure

Schur's inequality

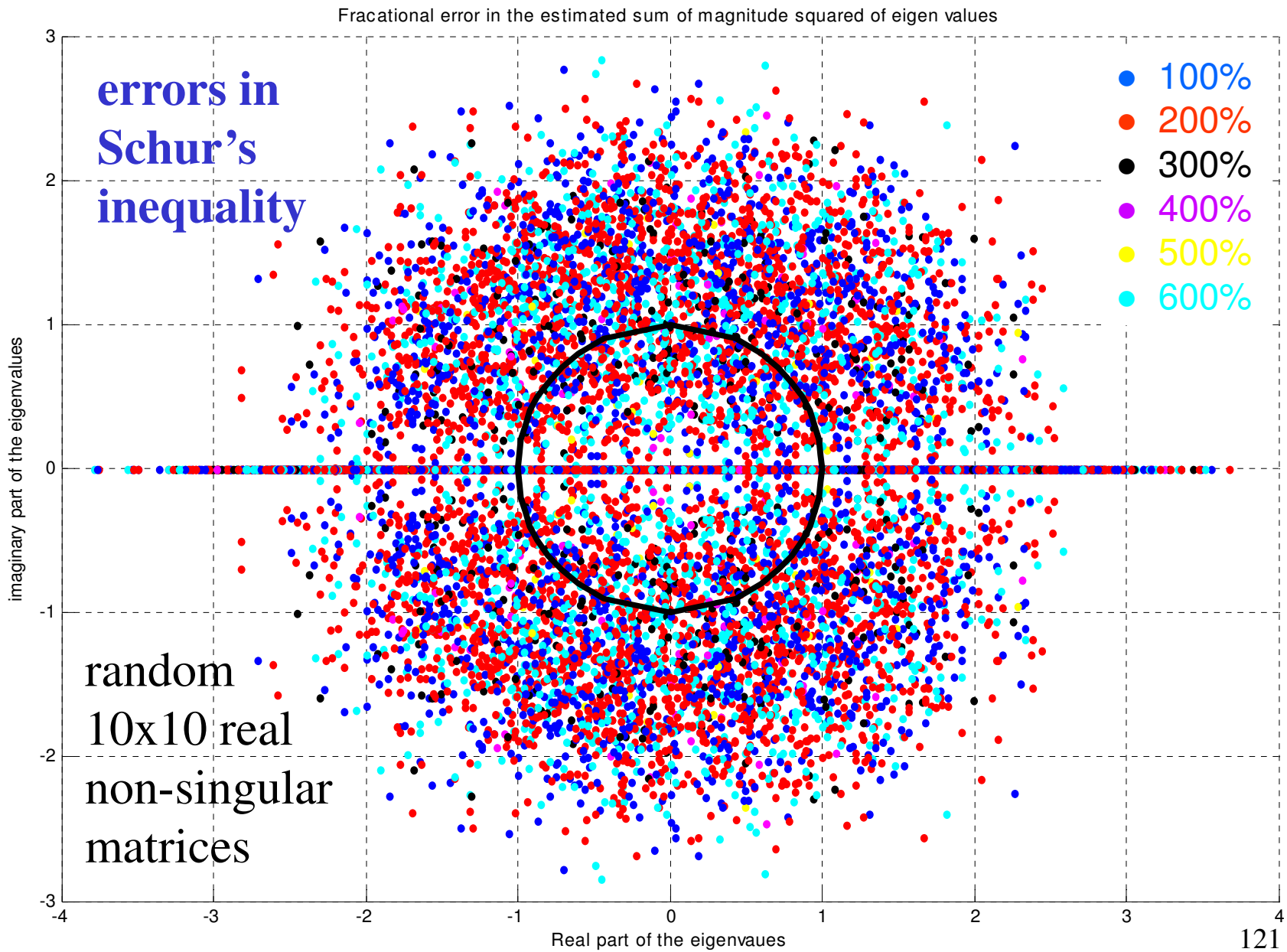
$$\sum_{j=1}^{d}\left|\lambda_j\left(\Phi_{filter}\right)\right|^2 \le Tr\left(\Phi_{plant}\Phi_{plant}^T\Phi_{Bayes}^T\Phi_{Bayes}\right)$$

optimal $\mathrm{K} \approx -(I + B)$ and thus optimal $\mathrm{y} \approx -\mathrm{x} - \mathrm{Bx}$

but we can "delinearize" Bx, resulting in $\mathrm{y} \approx -\mathrm{x} - \{\mathrm{Bx}\}_{\mathrm{delinearized}}$

Hence, optimal $\mathrm{y} \approx -\mathrm{x} + \mathrm{A}^{\#}\frac{\partial \log p(x,\lambda)}{\partial \lambda}$ but $\mathrm{DA}^{\#} = 0$, and thus

Optimal y results in $\frac{\mathrm{dx}}{\mathrm{d}\lambda} \approx -A^{\#}\frac{\partial \log p(x,\lambda)}{\partial \lambda} - Dx$


Fracational error in the estimated sum of magnitude squared of eigen values
errors in Schur's inequality
100%
200%
300%
400%
500%
600%
random 10x10 real non-singular matrices
imaginary part of the eigenvalues
Real part of the eigenvaues
3
2
1
0
-1
-2
-3
-4
-3
-2
-1
0
1
2
3
4

Exactly the same feedback is derived using standard control theory:

$$y = -D^T W^{-1} x$$

in which the controllability Grammian is :

$$W = \int_0^1 \exp(-sB) DD^T \exp(-sB^T) ds$$

where B is the linearization of the particle flow :

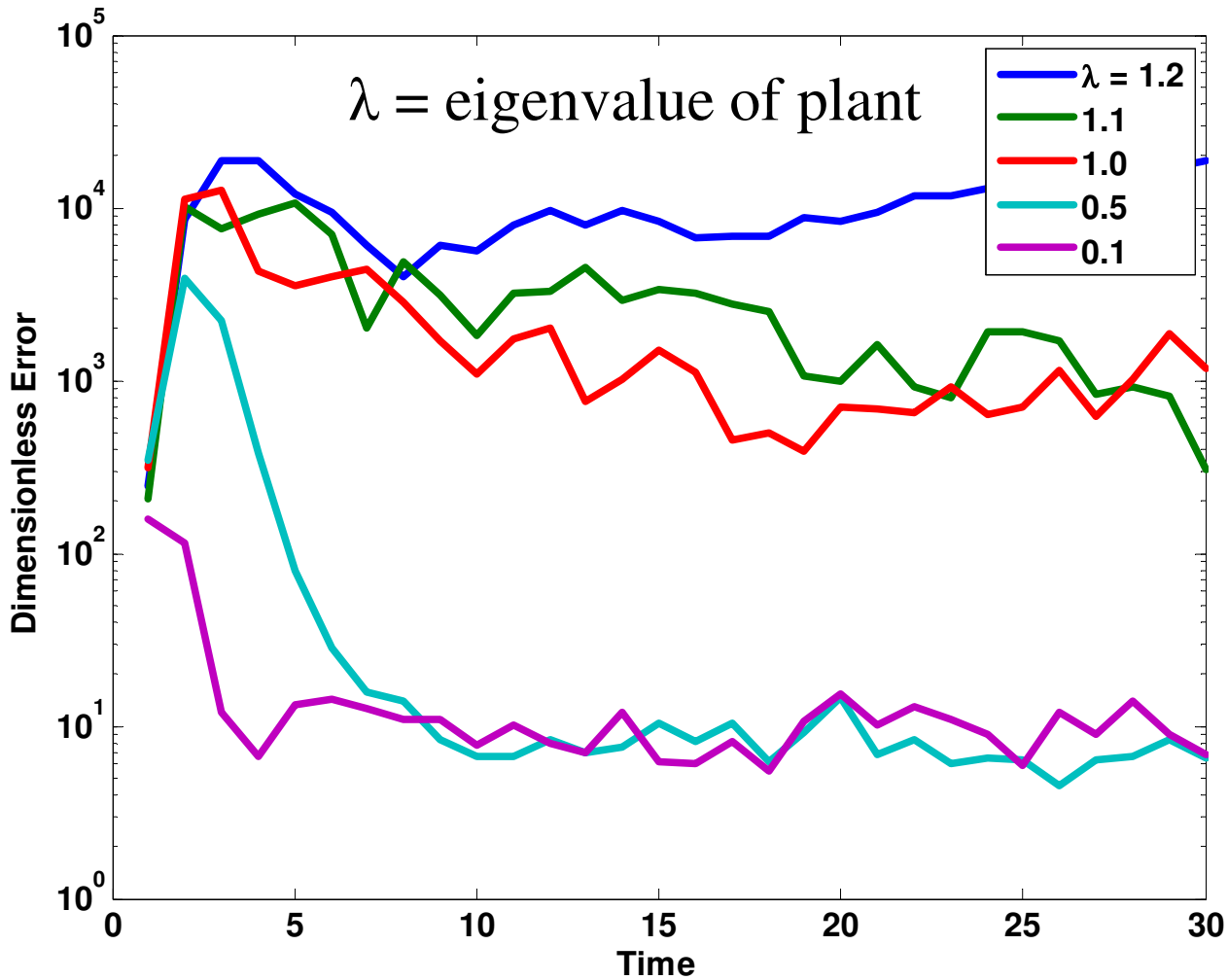
$$\frac{dx}{d\lambda} = -A^{\#} \log(h) + (I - A^T A / AA^T) y$$

$$\frac{dx}{d\lambda} \approx Bx + Dy$$

Using the facts that $D = D^2$ and D kills $A^{\#}$ and a little algebra :

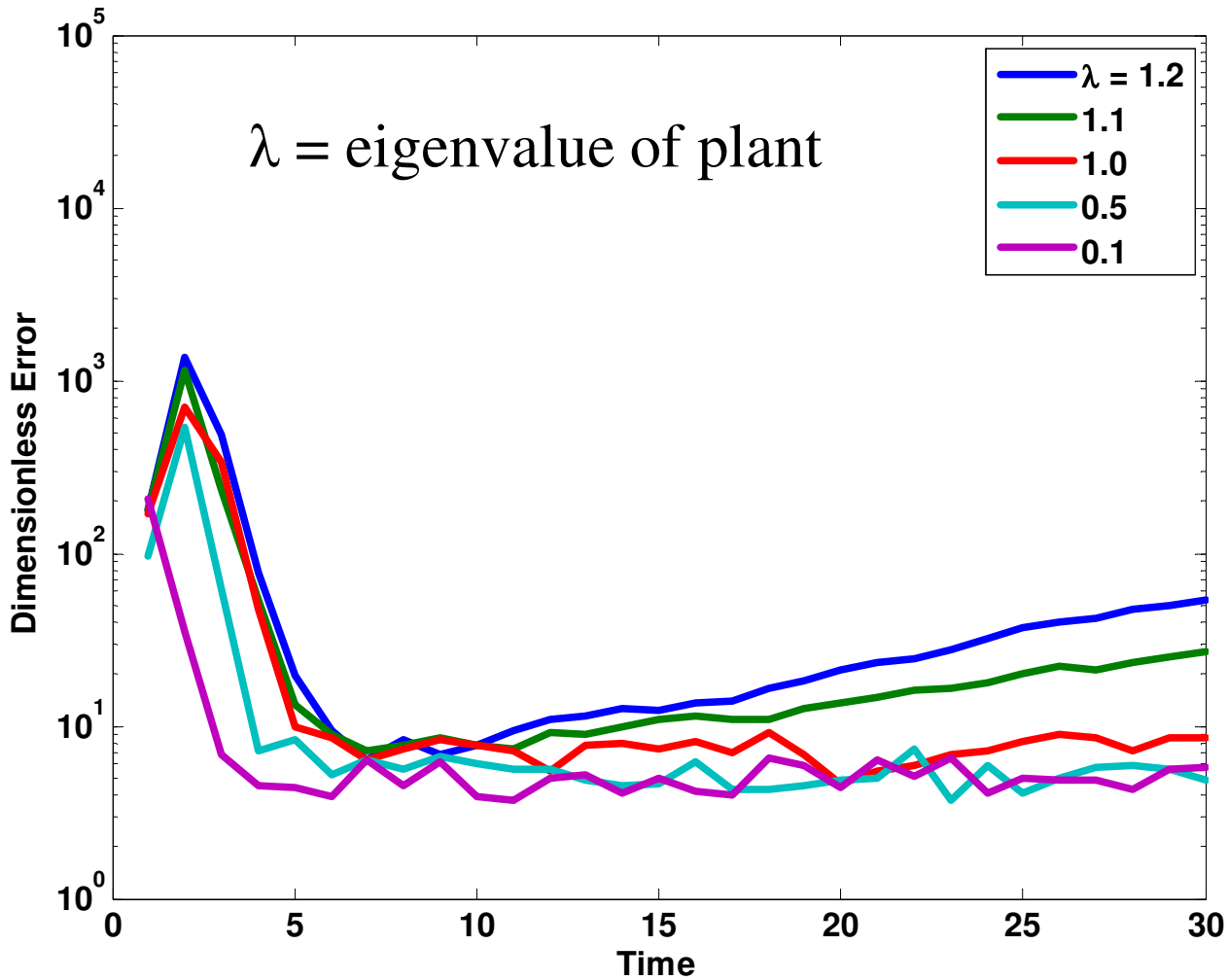
$$y = -Dx$$

Particle filter accuracy depends on the plant stability & mixing



d = 6, ny = 3, N = 500, #(MC) = 10, without -Dx

New theory (general flow) improves filter accuracy dramatically



d = 6, ny = 3, N = 500, #(MC) = 10, with -Dx

item	renormalization group flow in quantum field theory	particle flow for Bayes' rule
1. purpose	avoid infinite integrals at all energy scales (μ)	fix particle degeneracy in particle filters
2. PDE	linear first order PDE	linear first order PDE
3. method	"the trick of doing an integral a little bit at a time" (Tony Zee QFTNS p. 346)	homotopy of log-density
4. efficacy	"the most important conceptual advance in QFT over the last 3 or 4 decades" (Tony Zee QFTNS p. 337)	reduces computational complexity by many orders of magnitude for high dimensional problems
5. algorithm	ODE for motion of particles in N-dimensional space (Tony Zee QFTNS p. 340)	$f = dx/d\lambda$ x = particle in d-dimensions
6. derivation of PDE	$dH(x)/d\mu = 0$ H(x) = Hamiltonian	Fokker-Planck equation & definition of log p
7. new idea for particle flow inspired by RNGF	$dH(x)/d\mu = 0$ (we want H to be scale invariant)	$\partial f/\partial g = 0$ g = prior density

two steps in renormalization group flow & particle flow

step	physics	nonlinear filters
1. regularization	regularization (e.g., cut-off of integral or dimensional regularization d-ε)	homotopy of log-density
2. renormalization	modify effective charge & mass as the energy scale varies from high to low (integrate out degrees of freedom to maintain symmetries & finite number of parameters); scale invariant flow of parameters: $dH/d\mu = 0$	compute flow of particles that is invariant to errors in the prior density & the normalization constant: $\partial f/\partial g = 0$

exact recursive filters*

filter	conditional density	special condition on dynamics dx/dt = f + w
Kalman (1960)	**η = Gaussian**	**∂f/∂x = A(t)**
Beneš (1983)	**η exp(∫f(x)dx)**	**f(x) = ∂V/∂x and** **div(f) + ‖f‖² = x*Ax +bx+c**
Daum (1986)	**exponential family** $p(x\|Z) = p(x)\exp[\theta(x)\Psi(Z)]$	$\frac{\partial \theta}{\partial t} = \frac{\partial \theta}{\partial x}[Qr^T - f] + \frac{1}{2}\xi - A\theta$ $r = \frac{\partial \log p(x)}{\partial x}$ $\quad \xi_j = Tr\left[Q\frac{\partial^2 \theta_j}{\partial x^2}\right]$

*Daum, IEEE AES Systems Magazine, August 2005.

a miracle*

$$p(x,t|Z_t) = p(x,t|\Psi_t(Z_t))$$

dimension of Z grows with time

dimension of Ψ is fixed and finite for all time

*Daum, IEEE AES Systems Magazine, August 2005.

Monge-Ampere highly nonlinear PDE

$$y = T(x)$$

$$p(x)dx = p(y)dy$$

$$p(x) = p(y)\det\left[\frac{\partial y}{\partial x}\right]$$

Let $T(x) = \dfrac{\partial V}{\partial x}$

Hence,

$$p(x) = p(y)\det\left[\frac{\partial^2 V}{\partial x^2}\right]$$

one shot transport requires nonlinear PDE (and we cannot evaluate the functions at good points!), whereas particle flow only needs an extremely simple linear PDE

computing the Hessian of log p:

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

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

can compute Hessians using calculus or 2nd differences

$$\frac{\partial^2 \log p}{\partial x^2} \approx -C^{-1} + \lambda \frac{\partial^2 \log h(x)}{\partial x^2}$$

$C =$ sample covariance matrix of particles for prior $(\lambda = 0)$

with Tychonov regularization; or EKF or UKF covariance matrix

$$\frac{\partial^2 \log p}{\partial x^2} \approx -P^{-1}$$

$P =$ sample covariance matrix of particles for $\mathrm{p}(\mathrm{x}, \lambda)$

with Tychonov regularization; or EKF or UKF covariance matrix

formula that avoids inverse of sample covariance matrix:

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

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

$$\frac{\partial^2 \log p}{\partial x^2} \approx -C^{-1} + \lambda \frac{\partial^2 \log h(x)}{\partial x^2} \quad \text{for g(x)} \approx \text{Gaussian}$$

but Woodbury's matrix inversion lemma gives us :

$$(\mathrm{A}+\mathrm{B})^{-1} = A^{-1} - A^{-1}B(I + A^{-1}B)^{-1}A^{-1} \quad \text{for arbitrary B and non - singular A}$$

hence

$$\left[\frac{\partial^2 \log p}{\partial x^2}\right]^{-1} \approx -C - CB(I - CB)^{-1}C$$

in which

$$\mathrm{B} = \lambda \frac{\partial^2 \log h(x)}{\partial x^2}$$

formula that avoids computing Hessian of g(x):

$$\frac{\partial^2 \log g(x)}{\partial x^2} \approx -\frac{1}{N}\sum_{j=1}^{N}\left(\frac{\partial \log g(x_j)}{\partial x}\right)^T \frac{\partial \log g(x_j)}{\partial x}$$

derivation of the above:

$$\mathrm{E}\left[\frac{\partial^2 \log g(x)}{\partial x^2}\right] = -E\left[\left(\frac{\partial \log g(x)}{\partial x}\right)^T \frac{\partial \log g(x)}{\partial x}\right]$$

$$\frac{\partial^2 \log g(x)}{\partial x^2} \approx E\left[\frac{\partial^2 \log g(x)}{\partial x^2}\right]$$

$$E\left[\left(\frac{\partial \log g(x)}{\partial x}\right)^T \frac{\partial \log g(x)}{\partial x}\right] \approx \frac{1}{N}\sum_{j=1}^{N}\left(\frac{\partial \log g(x_j)}{\partial x}\right)^T \frac{\partial \log g(x_j)}{\partial x}$$

method to solve PDE	how to pick unique solution	comments
1. generalized inverse of linear differential operator	minimum L^2 norm	Coulomb's law or fast Poisson solver
2. Poisson's equation	irrotational flow	Coulomb's law or fast Poisson solver
3. generalized inverse of gradient of log-homotopy	incompressible flow	workhorse for multimodal densities
4. stabilized version of method #3	most robustly stable filter	workhorse for multimodal densities
5. separation of variables (Gaussian)	pick solution of specific form	extremely fast & hard to beat in accuracy
6. separation of variables (exponential family)	pick solution of specific form	generalization of Gaussian flow
7. variational formulation (Gauss & Hertz)	convex function minimization	generalization of minimum L^2 norm
8. optimal transport formulation (Monge-Kantorovich)	convex functional minimization (e.g., least action or Wasserstein metric, etc.)	very high computational complexity (e.g. Monge-Ampere fully nonlinear PDE)
9. direct integration (of first order linear PDE in divergence form)	choice of d-1 arbitrary functions	should work with enforcement of neutral charge density & importance sampling
10. method of characteristics (or generalized method of characteristics)	more conditions (e.g., small curvature or specify curl, or use Lorentz invariance)	can solve any first order linear PDE except for the one of interest to us!
11. another homotopy of the PDE (inspired by Gromov's h-principle)	initial condition of ODE & uniqueness of solution to ODE	like Feynman's perturbation for QED
12. finite dimensional parametric flow (e.g., f = Ax+b with A & b parameters)	non-singular matrix to invert	avoids PDE completely
13. Fourier transform of PDE (divergence form of linear PDE has constant coefficients!)	minimum L^2 norm or most stable flow	generalized inverse & Monte Carlo integration avoids inverse Fourier transform at random points in d dimensions
14. small "curvature" flow	set certain 2nd derivatives of flow to zero	solve d x d system of linear equations or numerically integrate ODE (like Feynman)
15. zero "curvature" flow	set acceleration of particles to zero	solve vector Riccati equation exactly in closed form (rather than solve PDE)!
16. constant "curvature" flow etc. etc.	set acceleration of particle to constant	solve polynomial multivariate equations (rather than PDE); maybe use homotopy
17. upper triangular Jacobian flow	set certain lower triangular terms in Jacobian to zero (but not all terms to zero)	inspired by Knothe-Rosenblatt rearrangement in transport theory
18. non-zero process noise in flow for Bayes' rule, with clever choice of f & Q to avoid PDE	compute gradient of PDE to obtain d equations in d unknowns	Q = covariance matrix of diffusion in flow: $dx = f(x, \lambda)d\lambda + \sqrt{Q}\, dw$

derivation of PDE for particle flow with Q ≠ 0:

$$\frac{dx}{d\lambda} = f(x,\lambda) + \sqrt{Q(x,\lambda)}\,\frac{dw}{d\lambda}$$

$$\frac{\partial p(x,\lambda)}{\partial \lambda} = -div(pf) + \frac{1}{2} div\left[Q(x,\lambda)\frac{\partial p}{\partial x}\right]$$

$$\frac{\partial \log p(x,\lambda)}{\partial \lambda}\, p(x,\lambda) = -div(pf) + \frac{1}{2} div\left[Q\frac{\partial p}{\partial x}\right]$$

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

$$\left[\log h(x) - \frac{d\log K(\lambda)}{d\lambda}\right] p(x,\lambda) = -div(pf) + \frac{1}{2} div\left[Q\frac{\partial p}{\partial x}\right]$$

$$\left[\log h - \frac{d\log K}{d\lambda}\right] p = -p\,div(f) - \frac{\partial p}{\partial x} f + \frac{1}{2} div\left[Q\frac{\partial p}{\partial x}\right]$$

$$\left[\log h - \frac{d\log K}{d\lambda}\right] = -div(f) - \frac{\partial \log p}{\partial x} f + \frac{1}{2p} div\left[Q\frac{\partial p}{\partial x}\right]$$

derivation of first new particle flow with Q ≠ 0:

$$\left[\log h - \frac{d\log K}{d\lambda}\right] = -div(f) - \frac{\partial \log p}{\partial x} f + \frac{1}{2p} div\left[Q(x)\frac{\partial p}{\partial x}\right]$$

$$\frac{\partial \log h}{\partial x} = -f^T \frac{\partial^2 \log p}{\partial x^2} - \frac{\partial div(f)}{\partial x} - \frac{\partial \log p}{\partial x}\frac{\partial f}{\partial x} + \frac{1}{2}\frac{\partial}{\partial x}\left\{div\left[Q(x)\frac{\partial p}{\partial x}\right]/p\right\}$$

pick Q such that the three last terms sum to zero, and solve for f:

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

why does the new flow work so well?

item	new flow	old flows
1. solution for flow	d x d matrix inverse	Moore-Penrose inverse of 1 equation in d unknowns
2. normalization of probability density	we killed the normalization	explicitly computed
3. exploits smoothness of density functions	smoother (2nd derivatives wrt x)	less smooth (only first derivatives wrt x)
4. exploits calculus to compute Hessian & gradient of likelihood	yes	no
5. exploits greater freedom with non-zero diffusion in flow	yes	no
6. depends on Monte Carlo approximation	no	yes
7. generality	more	less

small curvature flow:

Gaussian flow :

$$f = A(\lambda)x + b(\lambda)$$

$$\mathrm{div}(f) = \mathrm{Tr}(A)$$

incompressible flow :

$$\mathrm{div}(f) = 0$$

$$\frac{\partial div(f)}{\partial x} = 0$$

linear first order highly underdetermined PDE:

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

(1) we want a stable flow
(2) we want a full rank flow
(3) we want a fast algorithm to approximate f accurately (roughly 1%)

$$div(pf) = \eta$$

$$\text{let q} = \text{pf} \quad (\text{p} = \text{known} \ \& \ \text{f} = \text{unknown})$$

$$div(q(x,\lambda)) = \eta$$

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

$$\eta = \frac{\partial q_1}{\partial x_1} + \frac{\partial q_2}{\partial x_2} + \ldots + \frac{\partial q_d}{\partial x_d}$$

like Gauss' divergence law in electromagnetics with η analogous to electric charge density

irrotational particle flow:

$$\frac{dx}{d\lambda} = f(x,\lambda) = \left[\frac{\partial V(x,\lambda)}{\partial x}\right]^T / p(x,\lambda)$$

$$Tr\left[\frac{\partial^2 V(x,\lambda)}{\partial x^2}\right] = \eta(x,\lambda)$$

Poisson's equation

$$V(x,\lambda) = -\int \eta(y,\lambda)\frac{c}{\|x-y\|^{d-2}}dy \quad \text{for d} \geq 3$$

$$\mathrm{V}(\mathrm{x},\lambda) = \int \mathrm{p}(\mathrm{y},\lambda)\left[\mathrm{logh}(\mathrm{y})\text{-}\frac{\partial \mathrm{logK}(\lambda)}{\partial\lambda}\right]\frac{c}{\|x-y\|^{d-2}}dy$$

$$\frac{\partial V(x,\lambda)}{\partial x} = \int p(y,\lambda)\left[\log h(y) - \frac{\partial \log K(\lambda)}{\partial\lambda}\right]\frac{c(2-d)(x-y)^T}{\|x-y\|^d}dy$$

$$\frac{\partial V(x,\lambda)}{\partial x} = E\left[(\log h(y) - \frac{\partial \log K(\lambda)}{\partial\lambda})\frac{c(2-d)(x-y)^T}{\|x-y\|^d}\right]$$

$$\frac{\partial V(x_i,\lambda)}{\partial x} \approx \frac{1}{M}\sum_{j\in S_i}\left[(\log h(x_j) - \frac{\partial \log K(\lambda)}{\partial\lambda})\frac{c(2-d)(x_i-x_j)^T}{\|x_i-x_j\|^d}\right]$$

like Coulomb's law

derivation of Fourier transform particle flow:

$$div(pf) = -p[\log h - \frac{d \log K(\lambda)}{d\lambda}]$$

take the Fourier transform:

$$\mathrm{i}\omega^T \Im(\mathrm{pf}) = -\Im\left\{\mathrm{p}\left[\mathrm{logh} - \frac{\mathrm{dlogK}(\lambda)}{\mathrm{d}\lambda}\right]\right\}$$

$$i\omega^T \int p(x,\lambda) f(x,\lambda) \exp(-i\omega^T x) dx = -\int p(x,\lambda)[\log h(x) - E(\log h)] \exp(-i\omega^T x) dx$$

approximate the integrals using the Monte Carlo sum over particles:

$$\boxed{\mathrm{i}\omega^{\mathrm{T}}\left\{\frac{1}{N}\sum_{j=1}^{N} f(x_j,\lambda)\exp(-i\omega^T x_j)\right\} \approx -\frac{1}{N}\sum_{j=1}^{N}\left[\log h(x_j) - E(\log h)\right]\exp(-i\omega^T x_j)}$$

evaluate the above at k points in ω (e.g., k = d or 2d) and write this as a linear operator on the unknown function f :

$$\mathrm{L}(\omega)\mathrm{f} = y(\omega)$$

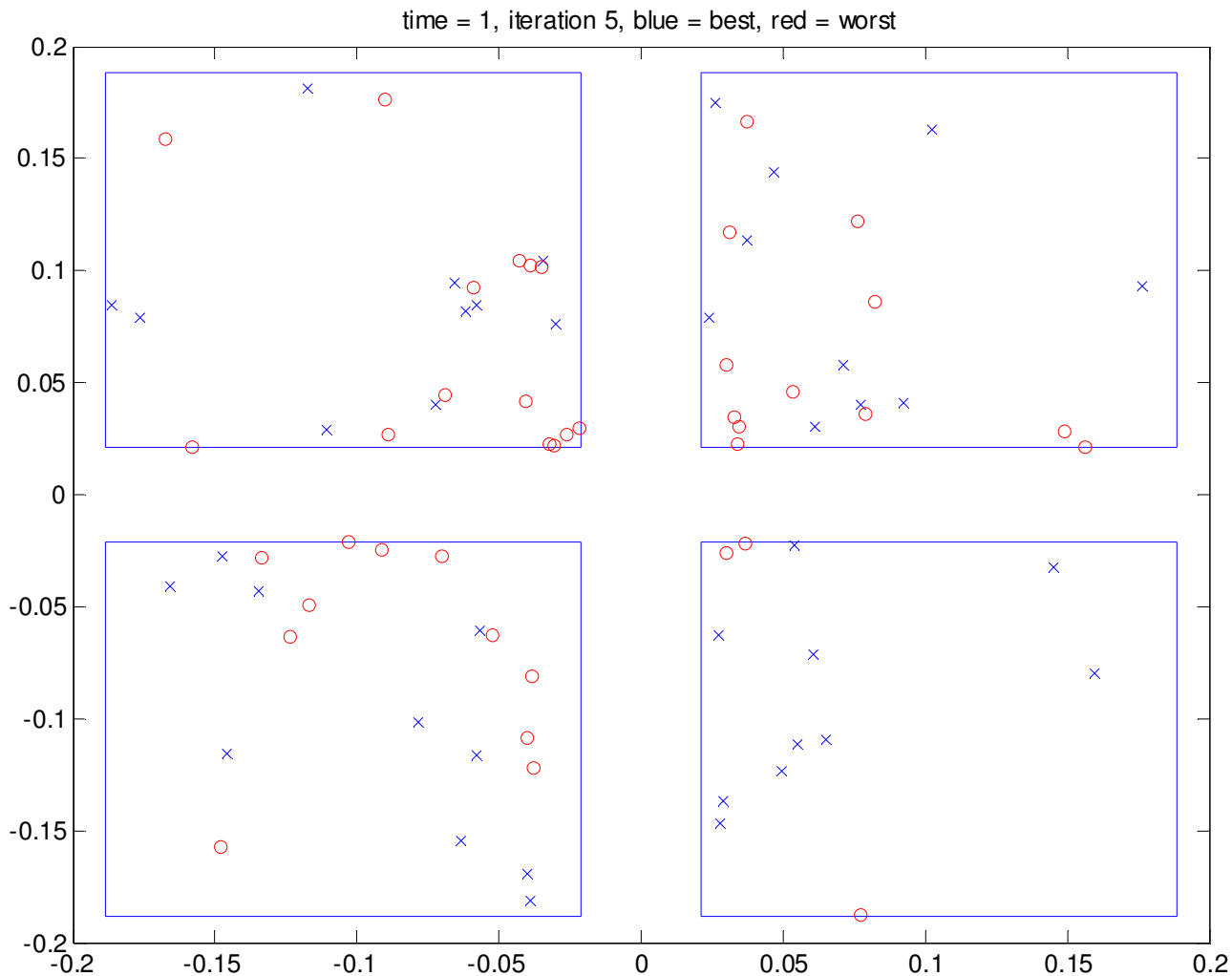
$\mathrm{f(x)} = \mathrm{L}^{\#} y$ in which $\mathrm{L}^{\#}$ = generalized inverse of L

in which $L^{\#} = L^T \left(LL^T\right)^{-1}$

Lf = y written out explicitly:

$$
\begin{bmatrix}
\omega_1^T \sin(\omega_1^T x_1) & \omega_1^T \sin(\omega_1^T x_2) & \cdots & \omega_1^T \sin(\omega_1^T x_N) \\
\omega_1^T \cos(\omega_1^T x_1) & \omega_1^T \cos(\omega_1^T x_2) & \cdots & \omega_1^T \cos(\omega_1^T x_N) \\
\omega_2^T \sin(\omega_2^T x_1) & \omega_2^T \sin(\omega_2^T x_2) & \cdots & \cdots \\
\omega_2^T \cos(\omega_2^T x_1) & \omega_2^T \cos(\omega_2^T x_2) & \cdots & \cdots \\
\vdots & \vdots & \vdots & \vdots \\
\vdots & \vdots & \vdots & \vdots \\
\omega_k^T \sin(\omega_k^T x_1) & \omega_k^T \sin(\omega_k^T x_2) & \vdots & \omega_k^T \sin(\omega_k^T x_N) \\
\omega_k^T \cos(\omega_k^T x_1) & \omega_1^T \cos(\omega_k^T x_2) & \vdots & \omega_k^T \cos(\omega_k^T x_N)
\end{bmatrix}_{2k \times dN}
\begin{bmatrix} f(x_1) \\ f(x_2) \\ \vdots \\ f(x_N) \end{bmatrix}_{dN \times 1}
=
\begin{bmatrix}
-\sum_j \left[\log h(x_j) - E(\log h)\right] \cos(\omega_1^T x_j) \\
\sum_j \left[\log h(x_j) - E(\log h)\right] \sin(\omega_1^T x_j) \\
-\sum_j \left[\log h(x_j) - E(\log h)\right] \cos(\omega_2^T x_j) \\
\sum_j \left[\log h(x_j) - E(\log h)\right] \sin(\omega_2^T x_j) \\
\vdots \\
\vdots \\
-\sum_j \left[\log h(x_j) - E(\log h)\right] \cos(\omega_k^T x_j) \\
\sum_j \left[\log h(x_j) - E(\log h)\right] \sin(\omega_k^T x_j)
\end{bmatrix}_{2k \times 1}
$$

optimization of points in k-space for Fourier transform



most general solution for incompressible flow:

$$\frac{dx}{d\lambda} = ?$$

by the chain rule :

$$\frac{d \log p(x,\lambda)}{d\lambda} = \frac{\partial \log p(x,\lambda)}{\partial x}\frac{dx}{d\lambda} + \frac{\partial \log p(x,\lambda)}{\partial \lambda} = 0$$

$$\frac{dx}{d\lambda} = -A^{\#} \log h(x) + [I - A^{\#}A]y$$

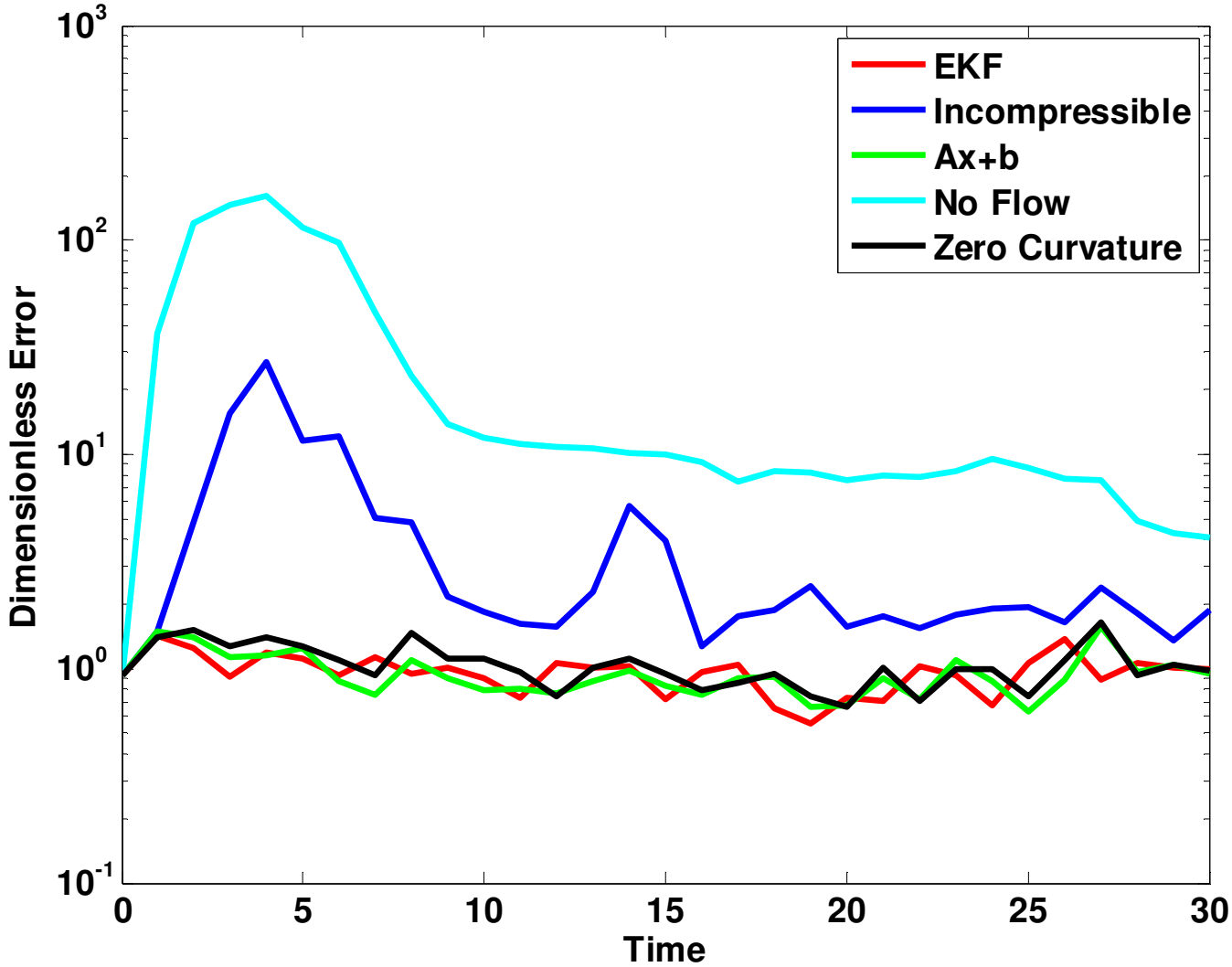
in which $\mathrm{A} = \dfrac{\partial \mathrm{logp(x,}\,\lambda)}{\partial \mathrm{x}}$

$y =$ arbitrary d - vector

pick y to maximize robustness of filter

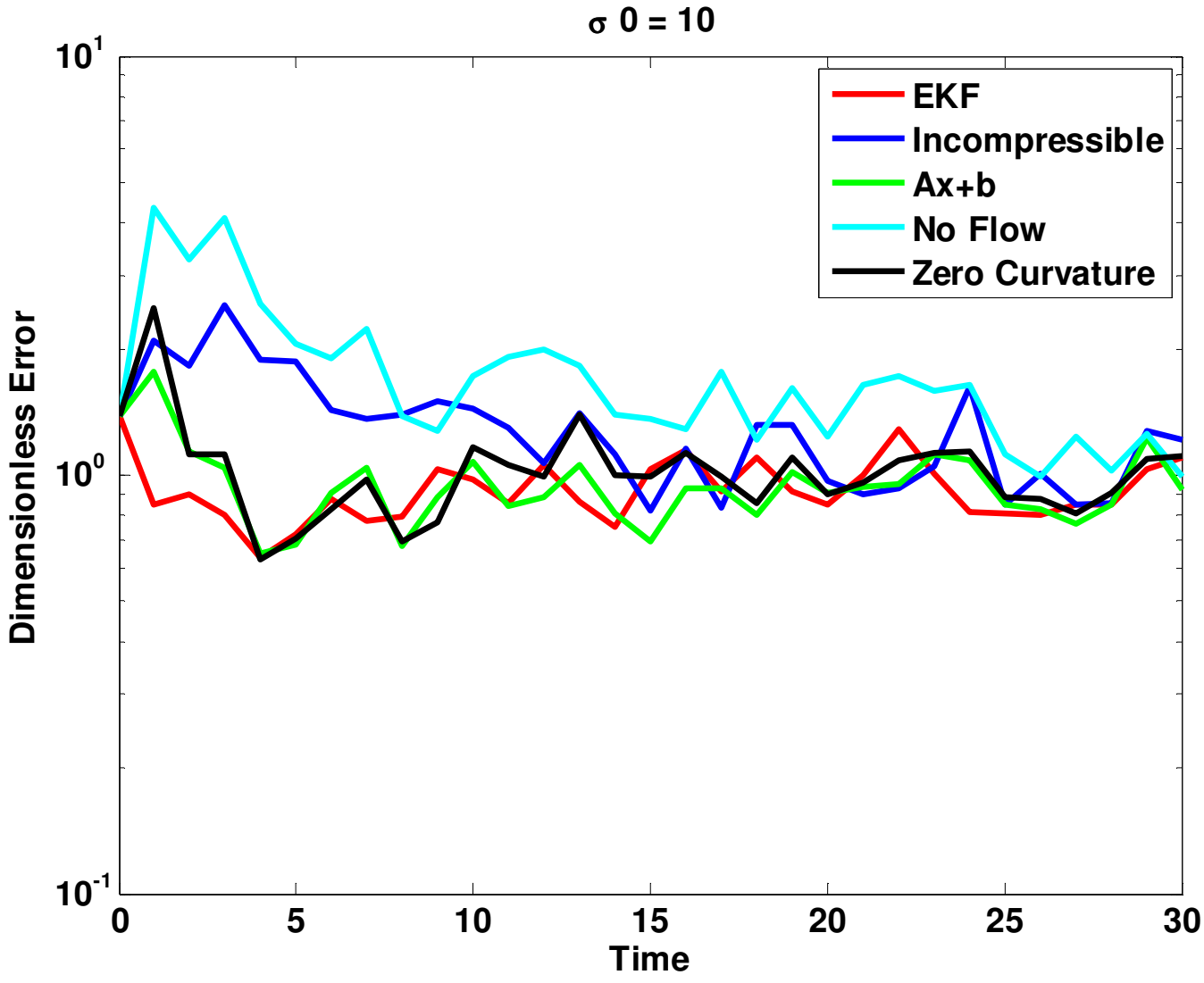
$\mathrm{A}^{\#} = A^{T}/(\mathrm{AA}^{\mathrm{T}})$ for $\mathrm{AA}^{\mathrm{T}} > 0,$

and $\mathrm{A}^{\#} = 0$ otherwise.

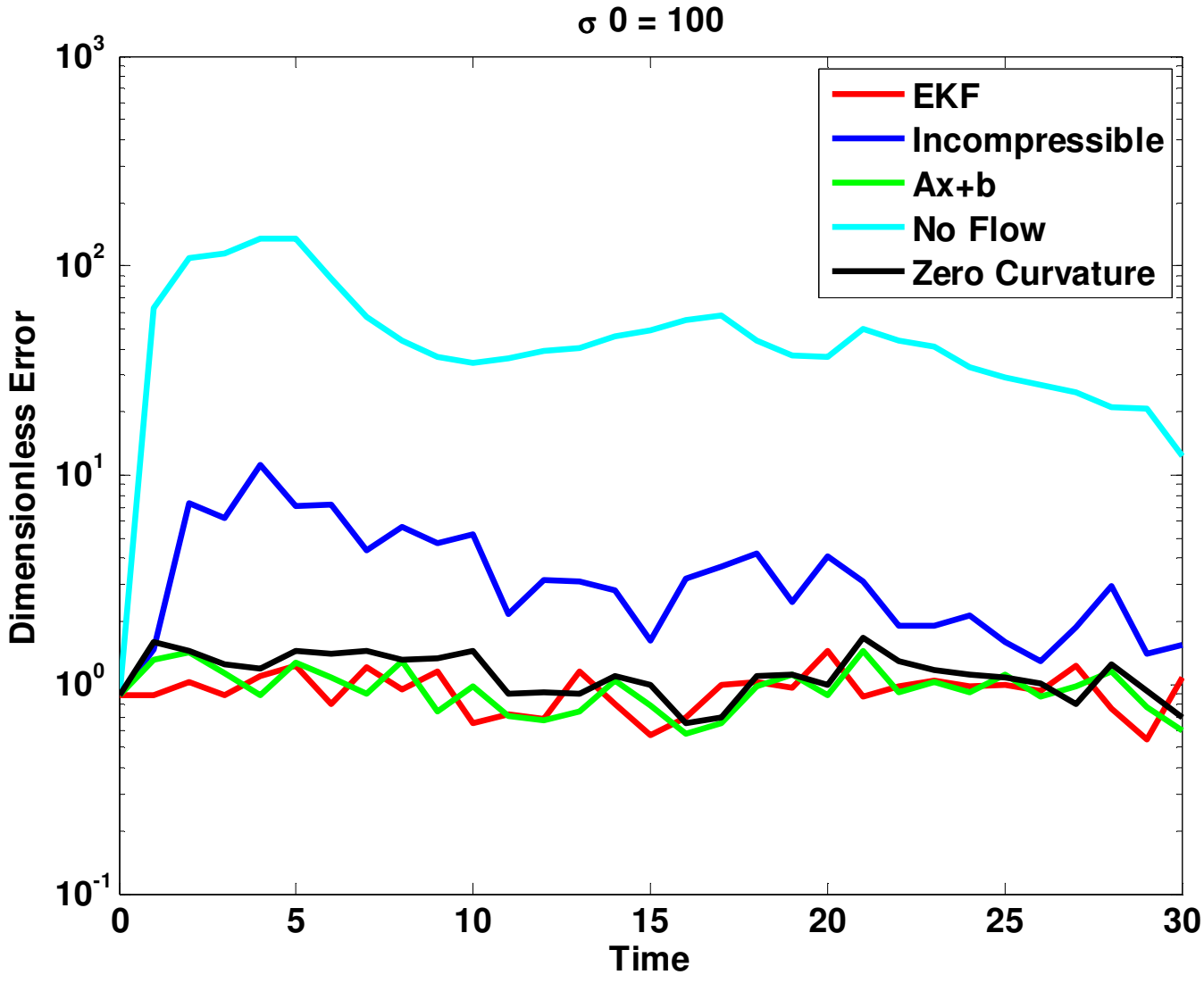


$\sigma 0 = 100$

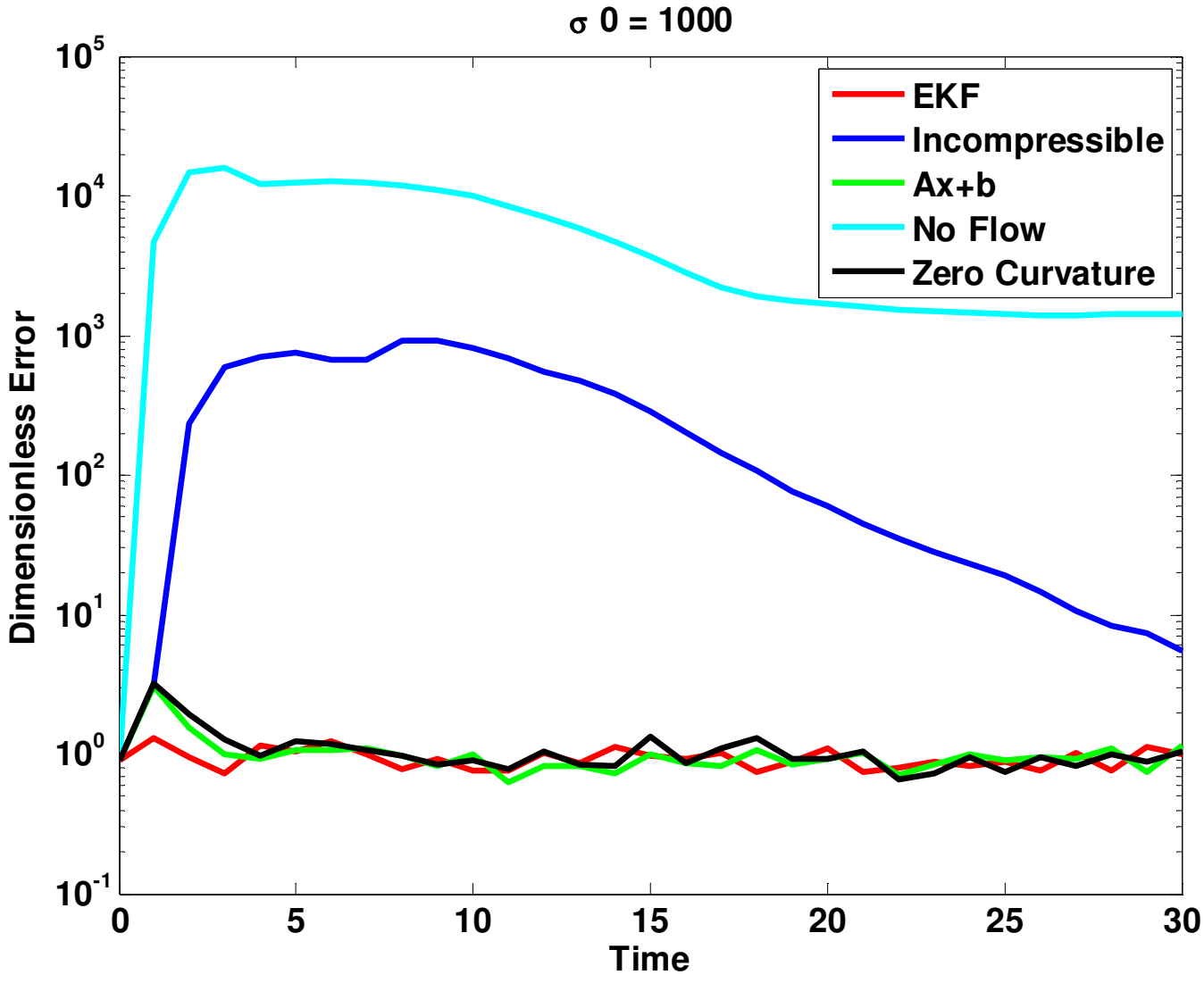
$N = 1000$



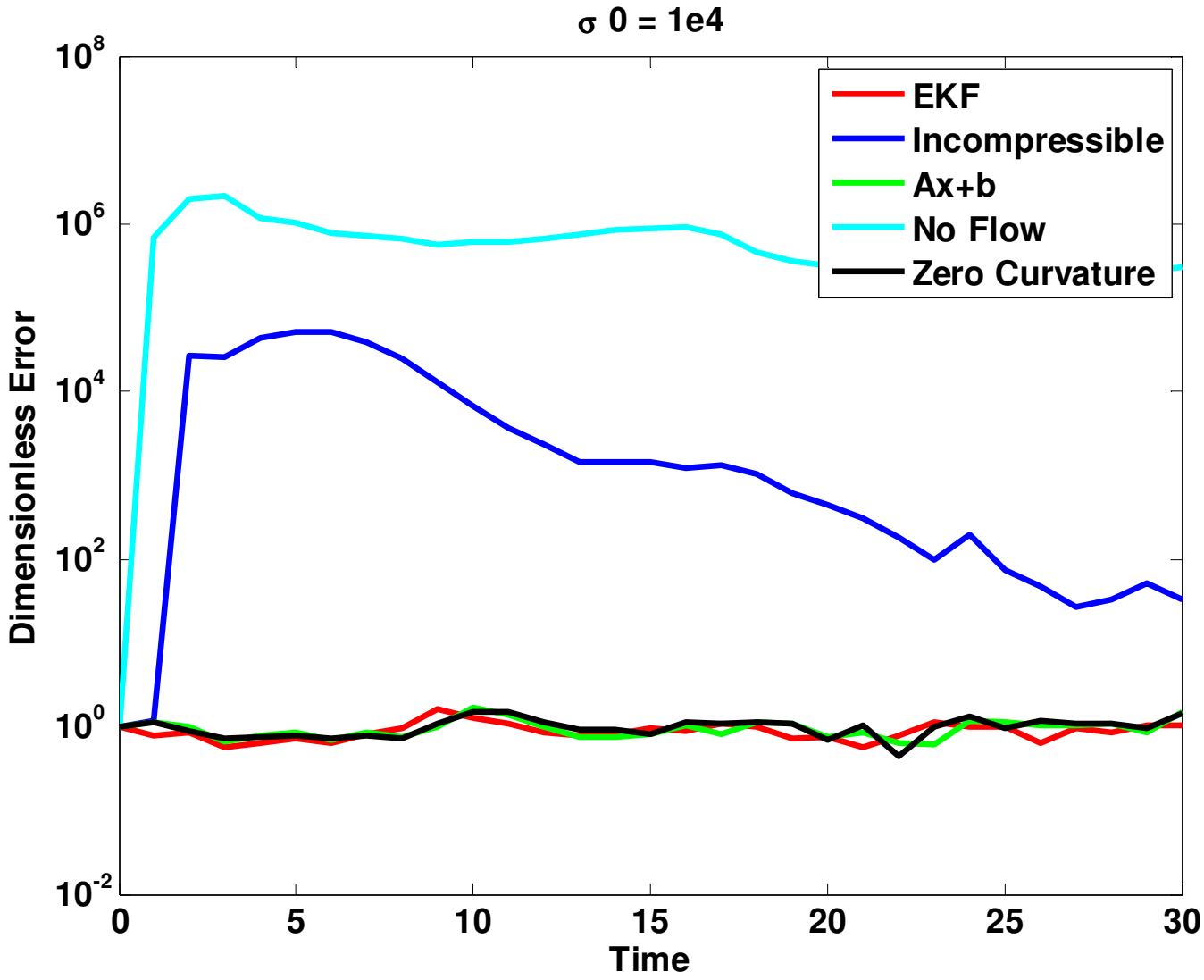
$\Delta\lambda_0$ = 1e-3



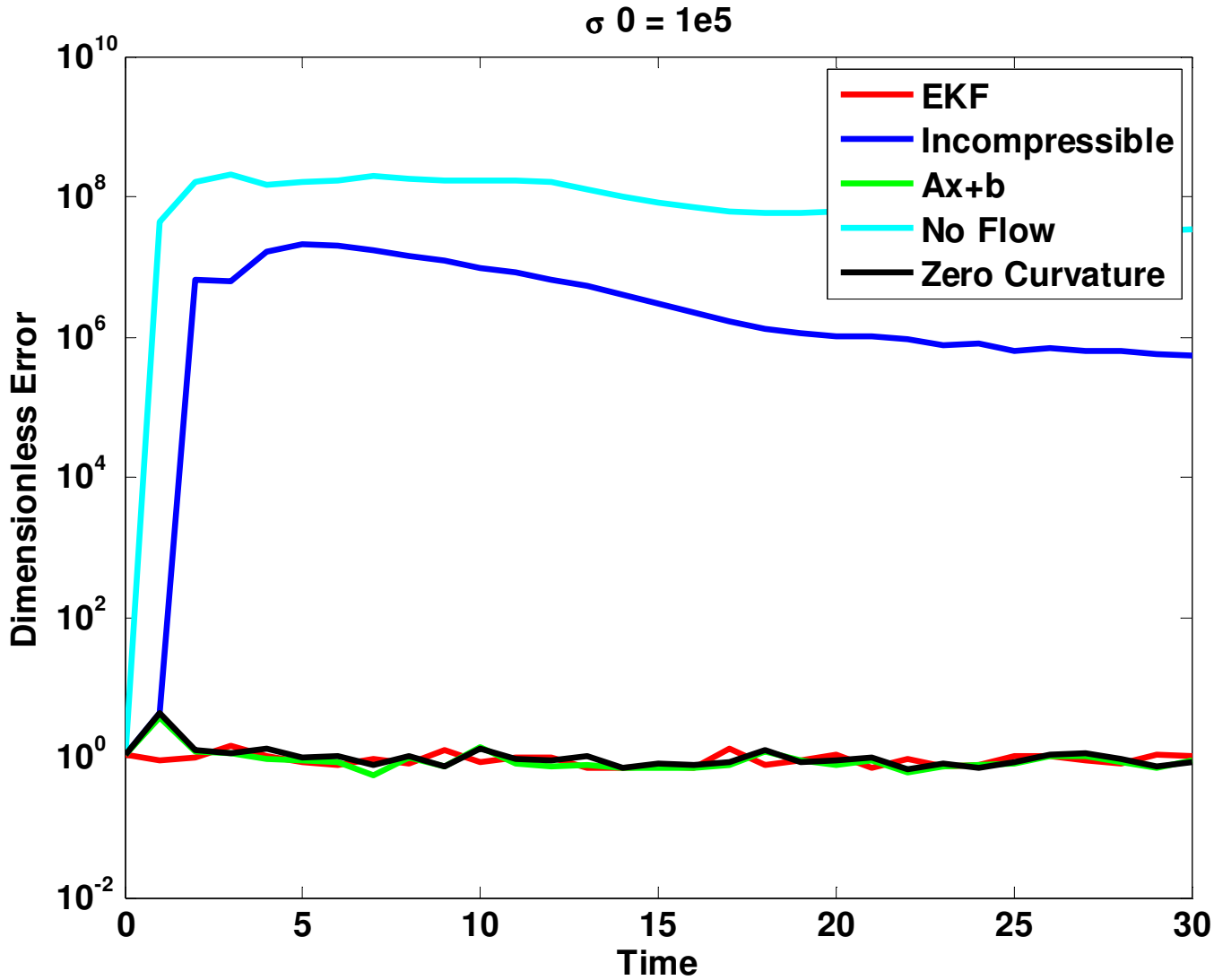
$\Delta\lambda_0$ = 1e-5



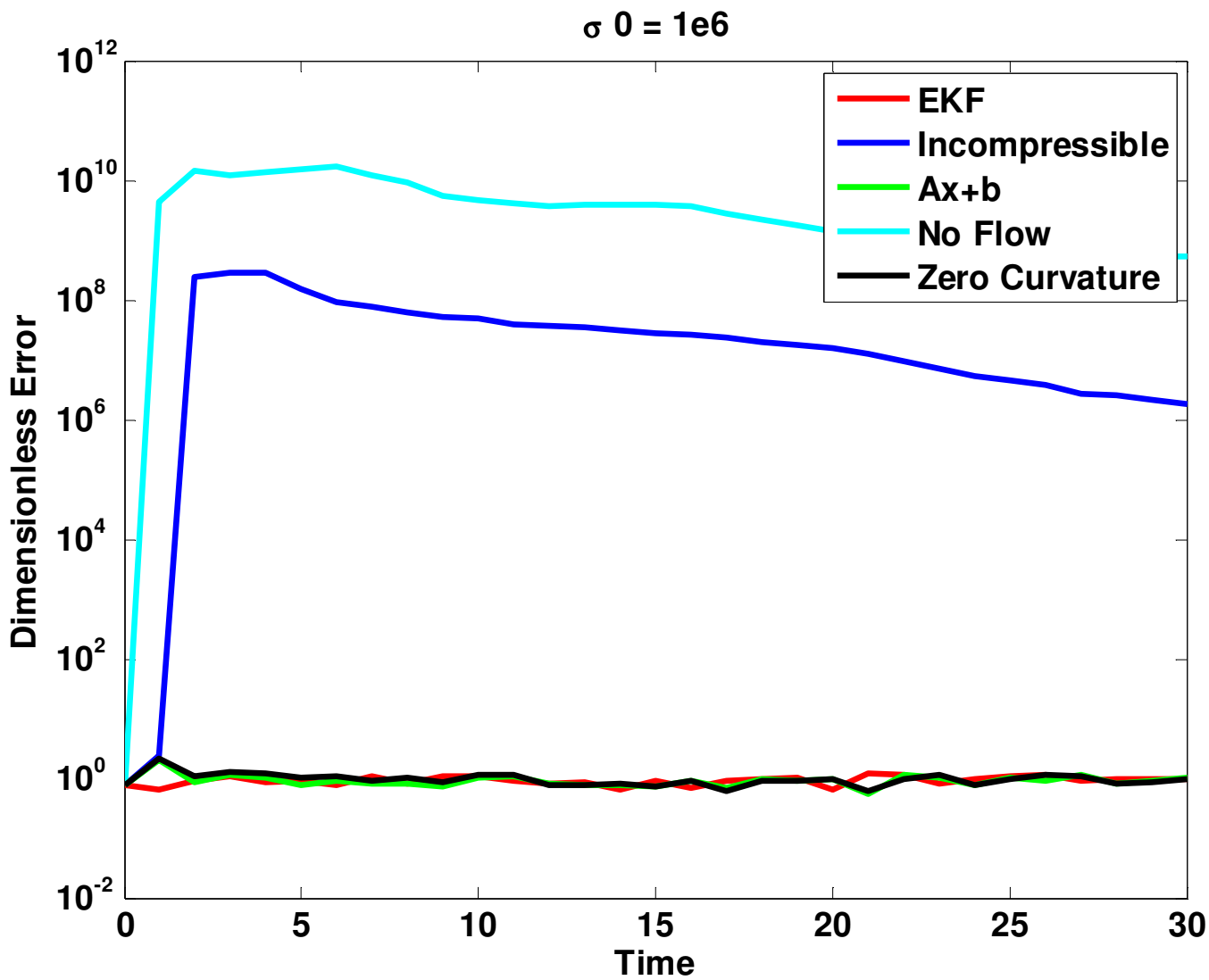
$\Delta\lambda_0$ = 1e-7



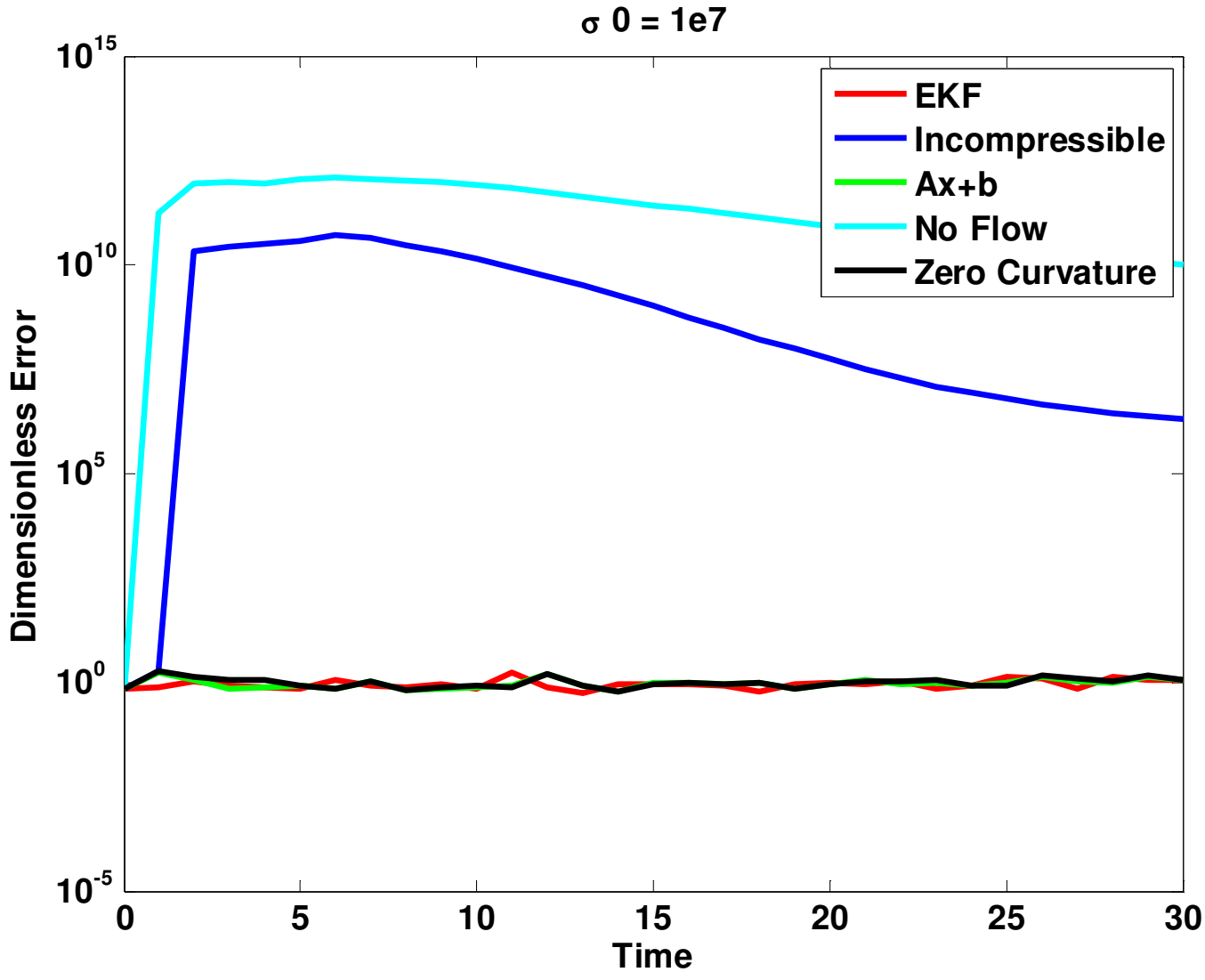
$\Delta\lambda_0 = 1e\text{-}9$



$\Delta\lambda_0$ = 1e-11



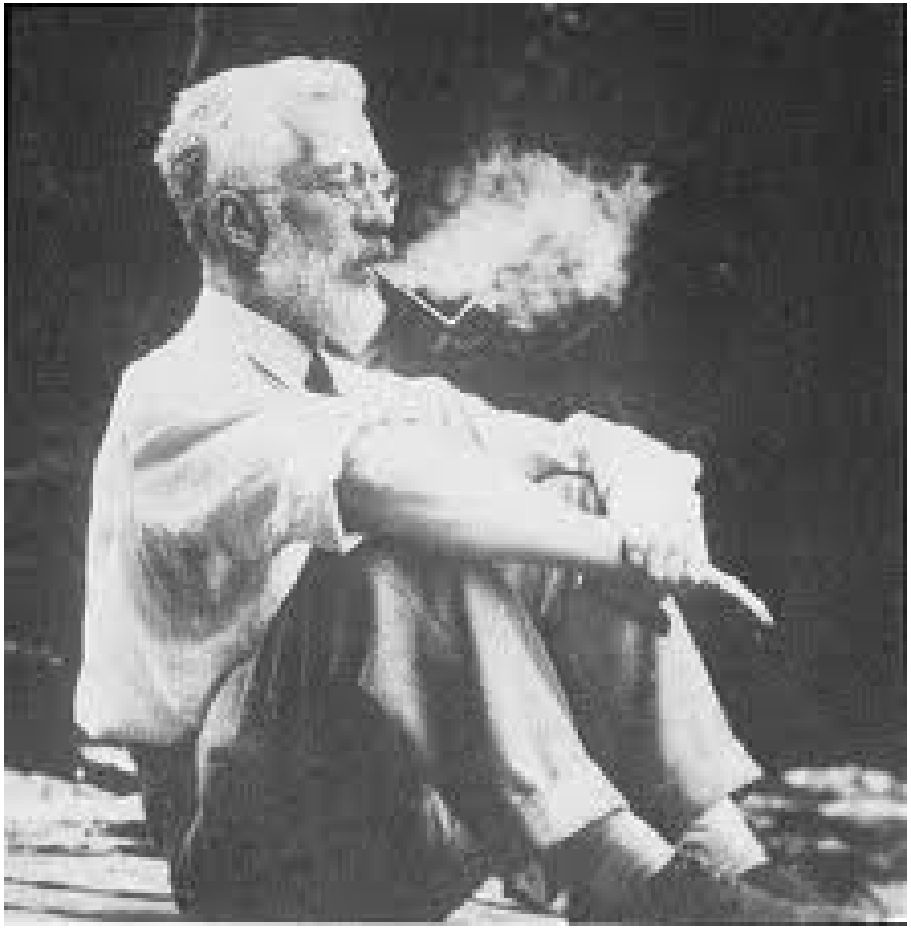
$\Delta\lambda_0$ = 1e-13



$\Delta\lambda_0$ = 1e-15

Fisher information matrix

$$J = -E\left[\frac{\partial^2 \log p}{\partial x^2}\right]$$



zero curvature flow:

we want to solve the following PDE for the flow f :

$$div(f) + \frac{\partial \log p}{\partial x} f = -\log h + \frac{d \log K}{d\lambda}$$

assume that the flow has zero curvature:

$$\frac{d^2 x}{d\lambda^2} = 0$$

but $f = \frac{dx}{d\lambda}$

hence $\frac{df}{d\lambda} = 0$

using this condition (after several pages of calculations) results in :

$$f^T \left[\frac{\partial^2 \log p}{\partial x^2} \right] f + 2 \frac{\partial \log h}{\partial x} f = \frac{d^2 \log K}{d\lambda^2}$$

zero curvature flow:

$$f^T\left[\frac{\partial^2 \log p}{\partial x^2}\right]f + 2\frac{\partial \log h}{\partial x} f = \frac{d^2 \log K}{d\lambda^2}$$

(1) vector Riccati equation for f rather than a PDE for f!
(2) highly underdetermined algebraic equation for f
(3) we can solve for f exactly in closed form!
(4) Hessian of logp is similar to the Fisher information matrix (we can exploit this to solve for f exactly in closed form)
(5) for nonlinear measurements with Gaussian noise, it is easy & fun to solve for f explicitly!

exploit non-singular symmetric pre-Fisher information matrix:

$$f^T\left[\frac{\partial^2 \log p}{\partial x^2}\right]f + 2\frac{\partial \log h}{\partial x} f = \frac{d^2 \log K}{d\lambda^2}$$

we can always write the above in the following canonical form :

$$\left\|\tilde{f}\right\|^2 - 2\frac{\partial \log h}{\partial x}\sqrt{H^{-1}}\,\tilde{f} = -\frac{d^2 \log K}{d\lambda^2}$$

in which :

$$\mathrm{H} = -\frac{\partial^2 \log p}{\partial x^2}$$

$$\tilde{f} = \sqrt{H}\, f$$

solution of general vector Riccati equation:

$$\left\|\tilde{f}-b\right\|^2=0$$

this is a single scalar - valued equation in d unknowns,

but it nevertheless has obvious the unique solution : $\tilde{f}=b$

$$(\tilde{f}-b)^T(\tilde{f}-b)=0$$

$$\left\|\tilde{f}\right\|^2-2b^T\tilde{f}+\left\|b\right\|^2=0$$

obviously this equation also has the unique solution : $\tilde{f}=b$

Encouraged by the above simple example, now consider our equation :

$$\left\|\tilde{f}\right\|^2+b^T\tilde{f}+c=0$$

let $\tilde{f}=kb$ in which k is a scalar

$$k^2\left\|b\right\|^2+k\left\|b\right\|^2+c=0$$

$k^2+k+c/\left\|b\right\|^2=0$ which has the solution : $k=\dfrac{-1\pm\sqrt{1-4c/\left\|b\right\|^2}}{2}$

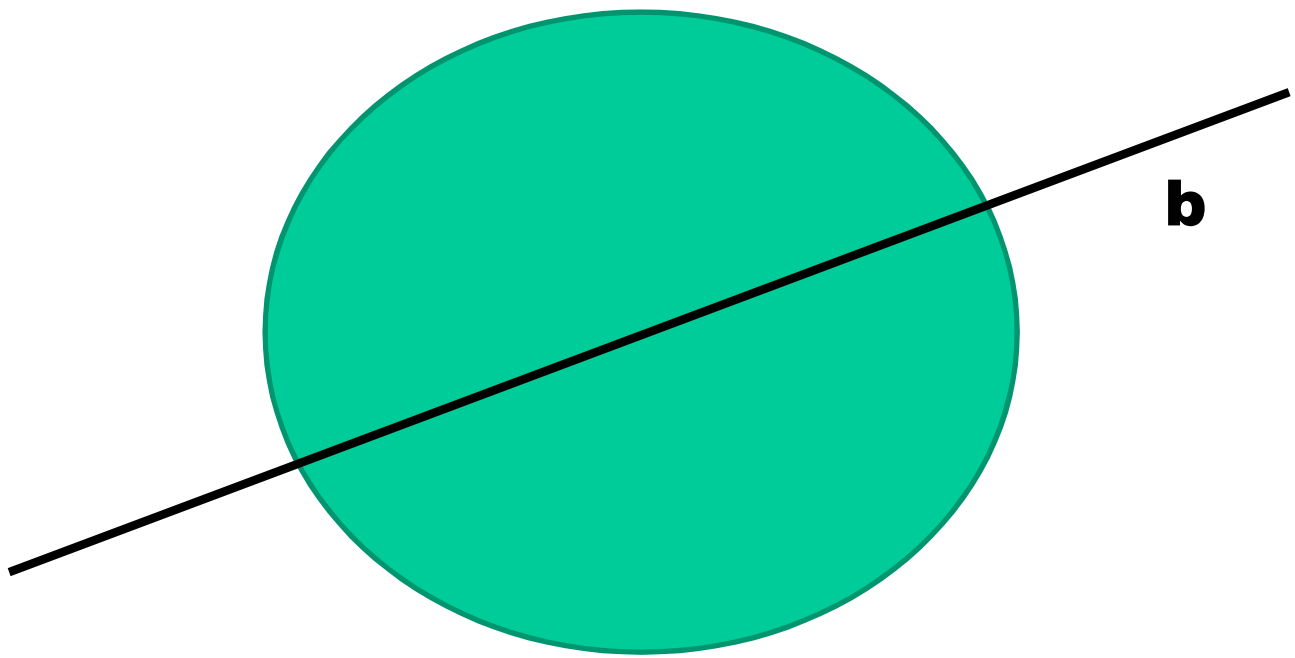
geometrical interpretation of solution:

$$\left\|\tilde{f}\right\|^2 + b^T \tilde{f} + c = 0$$

let $\tilde{f} = kb$ in which k is a scalar

$$k^2\|b\|^2 + k\|b\|^2 + c = 0$$

$$k^2 + k + c/\|b\|^2 = 0 \quad \text{which has the solution :} \quad k = \frac{-1 \pm \sqrt{1 - 4c/\|b\|^2}}{2}$$



solution of our vector Riccati equation:

$$\tilde{f} = \sqrt{H}\, f$$

$$f = \sqrt{H^{-1}}\, \tilde{f}$$

$$\tilde{f} = kb$$

$$b = -2\sqrt{H^{-1}} \left(\frac{\partial \log h}{\partial x} \right)^T$$

$$k = \frac{-1 \pm \sqrt{1 - 4c / \|b\|^2}}{2} \qquad \text{and} \qquad c = \frac{d^2 \log K}{d\lambda^2}$$

$$f = \sqrt{H^{-1}} \left[-2k\sqrt{H^{-1}} \left(\frac{\partial \log h}{\partial x} \right)^T \right]$$

$$f = -2kH^{-1} \left(\frac{\partial \log h}{\partial x} \right)^T$$

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

fast Ewald's method vs. Coulomb's law

item	fast Ewald method in physics & chemistry*	Coulomb's law with fast approximate k-NN	comments
1. dimension of x	d = 3	d = 3 to 30	rapid decay of Coulomb kernel in higher d helps!
2. relative error desired	0.0001 or better	1% to 10%	all Ewald methods the same for 1% accuracy
3. cut-off in x space	fixed distance	random per k-NN	automatic space-taper to weight convolution
4. desired force	on mesh	at particles	big difference!
5. neutral charge	locally enforced	locally enforced	crucial!!!
6. smoothing charge	Gaussian	no explicit smoothing	Debye kernel
7. k-space or real-space	both	real space (x)	no FFT needed for Coulomb

*Shan, Klepeis, Eastwood, Dror & Shaw, "Gaussian split Ewald: a fast Ewald mesh method for molecular simulation," Journal of Chemical Physics, 2005.

item	particle flow	Monge-Kantorovich transport
1. purpose	fix particle degeneracy due to Bayes' rule	move physical objects with minimal effort from one probability density to another
2. conservation of probability mass along flow	yes	yes
3. deterministic	yes*	yes
4. homotopy of density	no	yes
5. log-homotopy of density	yes	no
6. optimality criteria	**none**	Wasserstein metric (dirt mover's metric) or minimum action, etc.
7. how to pick a solution	24 distinct methods	minimize convex functional
8. stability of flow explicitly considered	yes	rarely
9. high dimensional applications	yes ($d \leq 42$)	no ($d = 1, 2$ or 3)
10. computational complexity	numerical integration of ODE for each particle	Poisson's PDE or HJB PDE or Monge-Ampere PDE etc.
11. solution of PDE for nice special cases	incompressible, irrotational, Gaussian, geodesic, etc.	Moser (1965 & 1990), Brenier (1991), Knothe-Rosenblatt (1952)
12. math theory for existence of incompressible flow, etc.	borrow Shnirelman's theorem, and Moser & Dacorogna (1990)	Shnirelman's theorem for $d \geq 3$, Moser & Dacorogna