

Illustrative examples and possible explanation for an unexpected behaviour of the particle filter

Jakob Åslund, Fredrik Gustafsson and Gustaf Hendeby

The self-archived postprint version of this conference paper is available at Linköping University Institutional Repository (DiVA):

<https://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-208594>

N.B.: When citing this work, cite the original publication.

Åslund, J., Gustafsson, F., Hendeby, G., (2024), Illustrative examples and possible explanation for an unexpected behaviour of the particle filter, *2024 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*. <https://doi.org/10.1109/MFI62651.2024.10705780>

Original publication available at:

<https://doi.org/10.1109/MFI62651.2024.10705780>

Copyright: Institute of Electrical and Electronics Engineers (IEEE)

<http://www.ieee.org/>

©2024 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Illustrative examples and possible explanation for an unexpected behaviour of the particle filter

Jakob Åslund

Department of Electrical Engineering
Linköping University, Sweden
jakob.aslund@liu.se

Fredrik Gustafsson

Department of Electrical Engineering
Linköping University, Sweden
fredrik.gustafsson@liu.se

Gustaf Hendeby

Department of Electrical Engineering
Linköping University, Sweden
gustaf.hendeby@liu.se

Abstract—The particle filter (PF) approximates the posterior distribution of the states in filtering problems, and it is well-known that it converges to the true posterior when the number of particles tends to infinity. It would be natural to assume that measures such as mean square error (MSE) decreases monotonically as the number of particles increases. This is, however, not always true. We present a simple two-dimensional linear Gaussian system where the MSE grows initially before it starts to decrease to eventually reach the optimal filter performance, which in this case is provided by the Kalman filter (KF). Other indicators such as the efficient number of particles and trace of the particle covariance show a similar strange behavior.

Inspired by this, we derive a condition for what we term *projected instability*, which means that the particle in the standard SIR PF that gives the best prediction actually increases the state estimation error. For linear systems, this gives an explicit condition in terms of the state space matrices when this situation occurs. Monte Carlo simulations of a large number of random linear systems indicate that everything works as expected as long as the system does not have a projected instability, otherwise the particle filter can perform badly or even diverge.

Index Terms—Particle Filter

I. INTRODUCTION

Since the particle filter (PF) was invented in 1993 [1], there have been plenty of surveys and tutorials [2]–[4] published. They all state that the PF converges to the true Bayesian posterior distribution as the number N of particles tends to infinity in theory. This statement is based on the seminal papers [5]–[7] on convergence analysis. There is much less known of the theoretical performance of the PF for small numbers N and the transient behaviour of how well the posterior is approximated. It might be natural to assume that the performance will monotonically increase with N . After all, a PF with a smaller N is a special case of a larger one, so performance should not be worse. The only thing that can affect this is the mixing step, but it is still rather unintuitive that a larger N can be worse than a smaller one.

To motivate the problem, we start by providing two examples of linear Gaussian systems (where the Kalman filter (KF) provides the ground truth posterior distribution), where the standard SIR PF performs as expected in one of them but in the other example the mean square error as a function of the number of particles, $MSE(N)$, initially increases before starting to decrease to eventually reach the optimal performance

as provided by the KF. There is really nothing strange about the illustrative examples, the Kalman filter has no problem at all, and the signal to noise ratio looks fine for a standard (bootstrap) SIR PF to be applied. One property of the examples is that the state noise is rank deficient, a problem we have studied in earlier contributions [8], [9] from other perspectives.

The difference of the two examples is that the problematic one satisfies a condition we call *projected instability*. This novel condition is based on an analysis of how a particle that after the time update gives a perfect prediction of the measurement actually can increase the state estimation error. That is, particles with high weights can destabilise the PF.

We thus have a tool to detect systems where problems may occur. The stability condition can be seen as an indicator and warning bell of potential issues in the PF. Though we constrain this contribution to examples that are linear, the stability condition is straightforward to generalise to nonlinear systems.

We further run a simulation study, which shows that the studied issue only occurs when the stability condition indicates an unstable system, though we also show that the stability condition is only a necessary condition and not a sufficient one.

Section II gives some brief background theory of the PF. Section III presents two examples that will be compared and provides Monte Carlo simulation results of MSE, as well as the efficient number of particles and trace of the estimated covariance. Section IV introduces the projected instability. Section V provides a broader simulation study of randomized systems, some of which contain a projected instability. Section VI analyses the problem from the theoretical point of view, and gives some insights into the mechanics which occur. Section VII gives a brief view of how the behavior changes if an optimal proposal is used for the PF. Section VIII concludes the paper.

II. BACKGROUND

The PF is a useful filter for estimating the state in many nonlinear systems of the type

$$x_{k+1} = f(x_k) + w_k \quad (1a)$$

$$y_k = h(x_k) + v_k, \quad (1b)$$

where $x_k \in \mathbb{R}^{n_x}$, $y_k \in \mathbb{R}^{n_y}$, f and h are functions $\mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n_x}$ and $\mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n_y}$, respectively, and w_k and v_k are noises of dimensions n_x and n_y , respectively, and with known probability distributions.

The algorithm for the SIR PF with N particles can be seen in Algorithm 1, where q is the proposal. For this work, we will primarily be focusing on the version where the proposal is the prior

$$q(x_k | \bar{x}_{k-1}^{(i)}, y_k) = p(x_k | \bar{x}_{k-1}^{(i)}). \quad (2)$$

Algorithm 1 The standard (bootstrap) SIR PF implementation

0: Initialize particles $i = 1 : N$ as

$$\begin{aligned} x_0^{(i)} &\sim p_{x_0}(x_0) \\ \omega_0^{(i)} &\propto p(y_0 | x_0^{(i)}). \end{aligned}$$

1: Resample according to weights $\omega_{k-1}^{(i)}$

$$\Pr(\bar{x}_{k-1}^{(i)} = x_{k-1}^{(j)}) = \omega_{k-1}^{(j)} \quad (3a)$$

$$\bar{\omega}_{k-1}^{(i)} = 1/N \quad (3b)$$

2: Time Update

$$x_k^{(i)} \sim q(x_k | \bar{x}_{k-1}^{(i)}, y_k) \quad (4)$$

3: Measurement update:

$$\omega_k^{(i)} \propto \bar{\omega}_{k-1}^{(i)} \frac{p(x_k^{(i)} | \bar{x}_{k-1}^{(i)}) p(y_k | x_k^{(i)})}{q(x_k^{(i)} | \bar{x}_{k-1}^{(i)}, y_k)} \quad (5)$$

4: Repeat from 1

The principle is to use samples, referred to as particles, to approximate the probability distribution as

$$p(x_k | y_{0:k}) \approx \sum_{i=1}^N \omega_k^{(i)} \delta(x - x_k^{(i)}), \quad (6)$$

where $x_k^{(i)}$ and $\omega_k^{(i)}$ are the state and assigned weight of the i :th particle, and $y_{0:k}$ are all measurements up until time k . Using this, it is possible to approximate the expected value of any function of x . The estimated value of x_k , given measurements up to time k is thus defined as

$$\hat{x}_{k|k} = \sum_{i=1}^N \omega_k^{(i)} x_k^{(i)}. \quad (7)$$

Since the method is based around approximating the probability distribution with a number of samples, the expectation would be that the more particles one uses, the better results one can expect. This is also common knowledge among practitioners.

In fact, it has been shown that when the number of particles approaches infinity, the approximation in (6) converges towards the true probability $p(x_k | y_{0:k})$ almost surely [7].

III. ILLUSTRATIVE EXAMPLES

There are however cases where the common knowledge fails. To illustrate this, we will use the two systems below:

System 1:

$$x_{k+1} = \underbrace{\begin{pmatrix} 0.6 & -1 \\ 0 & 0.5 \end{pmatrix}}_{F_1} x_k + \underbrace{\begin{pmatrix} 1 \\ 1 \end{pmatrix}}_G w_k \quad (8a)$$

$$y_k = \underbrace{\begin{pmatrix} 1 & 0 \end{pmatrix}}_H x_k + v_k, \quad (8b)$$

System 2:

$$x_{k+1} = \underbrace{\begin{pmatrix} 0.6 & 1 \\ 0 & 0.5 \end{pmatrix}}_{F_2} x_k + \underbrace{\begin{pmatrix} 1 \\ 1 \end{pmatrix}}_G w_k \quad (9a)$$

$$y_k = \underbrace{\begin{pmatrix} 1 & 0 \end{pmatrix}}_H x_k + v_k, \quad (9b)$$

where in both systems w_k and v_k are Gaussian distributions with variances $Q = 1$ and $R = 0.2$ respectively. The initial uncertain of the system P_0 was assumed to be \mathbf{I} . Both systems are stable, and even have the same poles 0.6 and 0.5. A practitioner might notice that the measurement noise is low which is generally not good for the standard PF. Further, the process noise only affects part of the state. While this might cause issues, there are so far no reasons to believe that the above understanding of the particle filter would break.

Note that, as the system is fully linear, the KF can be applied to provide the optimal estimates as a comparison.

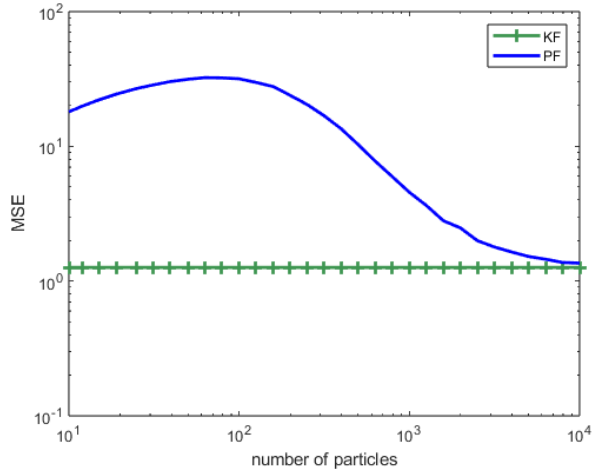
A. MSE

Applying a PF to the systems, however, results in the MSE estimates¹ seen in Figs. 1a and 1b. Clearly something strange is going on here. For System 1, increasing the number of particles results in a larger MSE for small numbers of particles. Further, despite the similarities, System 2 appears to require much fewer particles to approach the lower bound, here calculated by the KF.

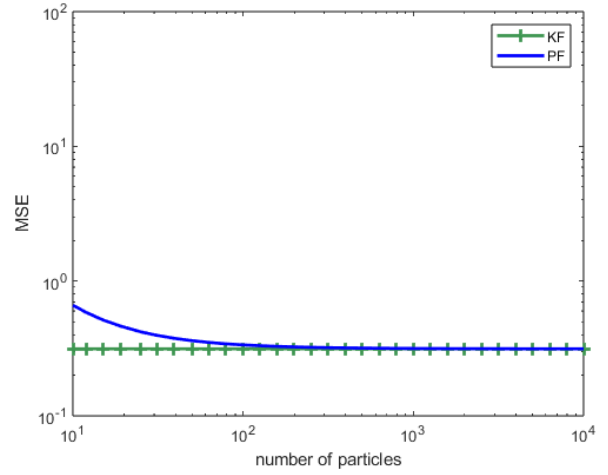
B. Spread of MSE

While observing the average MSE over multiple runs is often a good measure, it can also be interesting to see how the averaged MSE over all time steps differs for different simulations. This is here shown in Figs. 1c and 1d. As before, it is clear that for small numbers of particles for System 1, the MSE increases as the number of particles increases. Further, it is clear that even for larger numbers of particles there is the potential for large outlier errors in the PF, and that the largest errors continue to grow even after the median MSE has decreased.

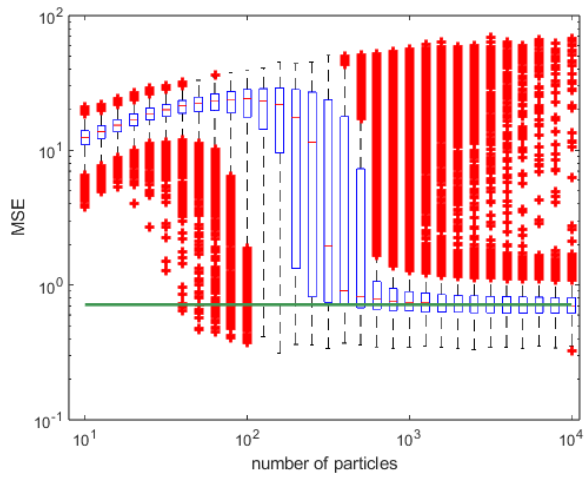
¹Measuring the mean squared error over all simulations and time steps of the filtered estimate $\hat{x}_{k|k}$ over 10000 simulations, with 100 time steps in each simulation.



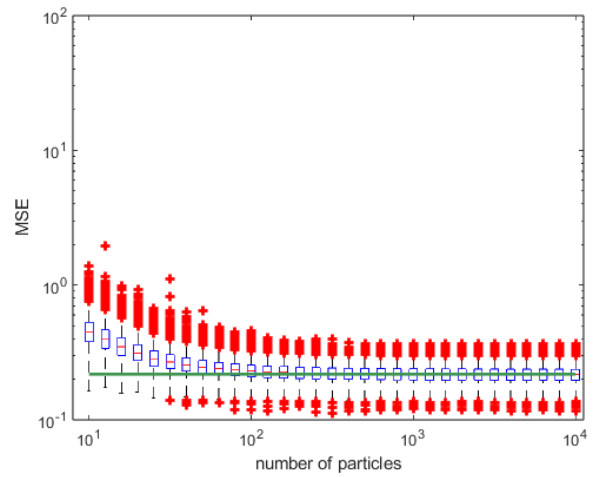
(a) MSE for System 1 in (8).



(b) MSE for System 2 in (9).



(c) Spread of the MSE for System 1 in (8).



(d) Spread of the MSE for System 2 in (9).

Fig. 1: A comparison of the MSE for the two studied systems, averaged over x_1 and x_2 . The green line is the KF MSE.

This can be contrasted with the behaviour for System 2, which behaves as one would expect from a PF. For small numbers of particles the error is larger with a larger spread and as the number of particles increases both the MSE and the spread of the MSE decreases.

C. Number of effective particles

Another method used to analyse how well the PF estimates the system is to look at the number of effective particles N_{eff} of the system, as introduced by [10]. N_{eff} is a tool for estimating how many particles contribute to the approximation at any specific time, and is computed as

$$N_{\text{eff}}(k) \approx \frac{1}{\sum_{i=1}^N \left(\omega_k^{(i)}\right)^2}. \quad (10)$$

Using this measure, we are interested in the spread of the average N_{eff} over all time steps. This is shown in Figs. 2a and 2b. For System 2, the average number of effective particles

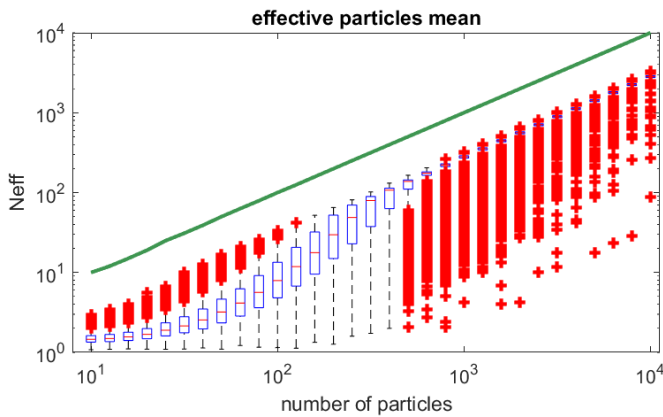
is approximately a constant fraction of the total number of particles. For System 1 however, we can see that for low numbers of particles, the number of effective particles is closer to 1, but then increases to some constant fraction when the number of particles increases. The average number of particles approaching a constant fraction coincides with the median MSE in Figs. 1c and 1d approaching the MSE of the KF. As with the MSE spread, there are also many outliers with very low N_{eff} even for high numbers of particles.

D. Covariance estimation

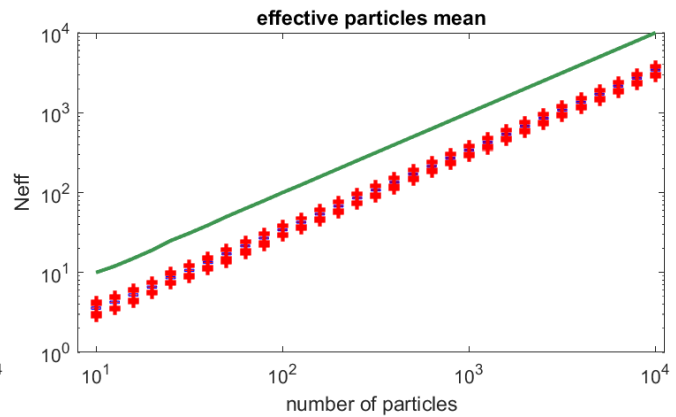
The final measure used in this work, is the estimate of the covariance of the distribution. For a PF this is calculated as

$$\hat{P}_{k|k} = \sum_{i=1}^N \omega_k^{(i)} \left(x_k^{(i)} - \hat{x}_{k|k}\right) \left(x_k^{(i)} - \hat{x}_{k|k}\right)^T, \quad (11)$$

where $\hat{x}_{k|k}$ is the filtered estimate of x_k described in (7). The idea is that for the PF to provide a good estimate of the entire probability distribution, it should also have a good estimate

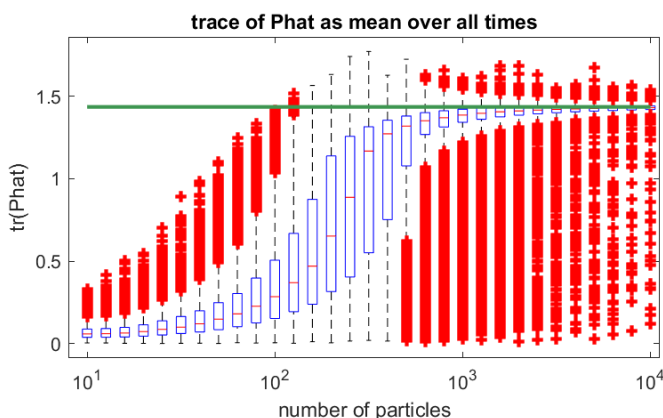


(a) Mean of N_{eff} for System 1 in (8).

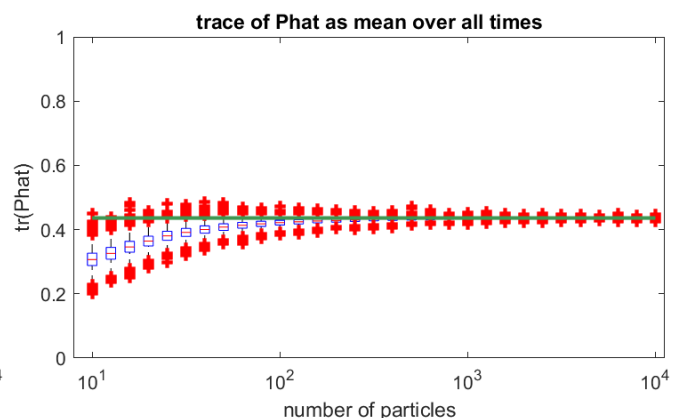


(b) Mean of N_{eff} for System 2 in (9).

Fig. 2: Spread of the number of effective particles of a PF (10), averaged over all time steps, as a function of number of particles. The green line represents the total number of particles.



(a) Trace of the estimated covariance, for System 1 in (8).



(b) Trace of the estimated covariance, for System 2 in (9).

Fig. 3: A comparison of the average estimated trace over all time steps of the covariance (11) of the two systems. The green line represents the true (optimal) covariance.

of the covariance. To simplify the presentation somewhat, we will only present the trace of the covariance, averaged over all time steps, which should give a good general estimate of how spread out the particles are compared to how spread out they should be. These measures are presented in Fig. 3. As with the previous analyses, for System 1, the estimate for small numbers of particles is much worse, and it only starts improving at approximately 300 particles. A connection can also be drawn to the number of effective particles shown previously. Since in many cases there was only 1 particle with any meaningful weight, the covariance must be around 0, which is reflected here. The fact that for System 1 there are large deviations between runs even for large numbers of particles can also be seen.

IV. THE PROJECTED INSTABILITY

In this section we will show the interesting fact that a particle that gives a perfect prediction which fits exactly to the measurement can actually increase the state estimation error under a certain condition.

A. Derivation for Linear Systems

Assume a linear system

$$x_{k+1} = Fx_k + Gw_k \quad (12a)$$

$$y_k = Hx_k + v_k, \quad (12b)$$

where w_k and v_k are noises with covariances Q and R respectively. Next, assume there is state noise $\tilde{w}_k^{(i)}$ that propagates the current particle $\tilde{x}_k^{(i)}$ to a new particle

$$\tilde{x}_{k+1}^{(i)} = F\tilde{x}_k^{(i)} + G\tilde{w}_k^{(i)}, \quad (13)$$

that gives a perfect fit to the measurement

$$y_{k+1} = H\tilde{x}_{k+1}^{(i)}, \quad (14)$$

or, if no such solution exists, one for which the norm of

$$y_{k+1} - H\tilde{x}_{k+1}^{(i)}, \quad (15)$$

is as small as possible. In the SIR-PF, clearly, this noise realization would result in the largest weight possible for that particle in the measurement step.

To find the process noise which satisfies (14), combine the equation with (12) and (13) to get

$$\begin{aligned} H \left(F \tilde{x}_k^{(i)} + G \tilde{w}_k^{(i)} \right) &= H (F x_k + G w_k) + v_{k+1} \\ \iff \tilde{w}_k^{(i)} &= (HG)^\dagger H F \tilde{x}_k^{(i)} + (HG)^\dagger H G w_k + (HG)^\dagger v_k, \end{aligned} \quad (16)$$

where $\tilde{x}_k^{(i)} = x_k - \hat{x}_k^{(i)}$ is the error between the estimate and the true state, and the subscript \dagger denotes the pseudo-inverse. Note that if no solution to (14) exists, the solution provided in (16) instead minimizes the norm of (15).

Inserting this expression back into (13) results in

$$\begin{aligned} \tilde{x}_{k+1}^{(i)} &= x_{k+1} - \hat{x}_{k+1}^{(i)} \\ &= (F - G(HG)^\dagger H F) \tilde{x}_k^{(i)} - G(HG)^\dagger v_k - G(HG)^\dagger H G w_k \end{aligned} \quad (17)$$

The conclusion is that the system in (12) contains what we term as a *projected instability* if

$$\bar{F} = F - G(HG)^\dagger H F, \quad (18)$$

has eigenvalues outside of the unit circle, since this is the requirement for the system in (17) to be unstable.

For the simple case where $H \in \mathbb{R}^{1 \times n}$ and $G \in \mathbb{R}^{n \times 1}$, \bar{F} can be further simplified as

$$\bar{F} = F - \frac{1}{HG} G H F. \quad (19)$$

B. Illustrative Examples

Applying this indicator to systems 1 and 2 from (8) and (9), we get

$$\bar{F}_1 = \begin{pmatrix} 0 & 0 \\ -0.6 & 1.5 \end{pmatrix} \quad (20a)$$

$$\bar{F}_2 = \begin{pmatrix} 0 & 0 \\ -0.6 & -0.5 \end{pmatrix}, \quad (20b)$$

respectively. Clearly, \bar{F}_1 has an eigenvalue in 1.5 (and one in 0) and System 1 therefore contains a projected instability, while System 2 has its eigenvalues in 0 and -0.5 and is therefore does not contain a projected instability. This indicates that the projected instability may be the explanation in the difference of behaviour.

C. Extension to Nonlinear Systems

The derivation of the condition for projected instability can be extended to nonlinear systems of the form

$$x_{k+1} = f(x_k, w_k) \quad (21a)$$

$$y_k = h(x_k, v_k), \quad (21b)$$

for some nonlinear functions $f(x, w)$ and $h(x, v)$.

First, we find the process noise that gives a perfect prediction. To do this extension, one must again assume an estimate \tilde{x}_k propagated in time

$$\tilde{w}_k = \text{sol}\{w : y_{k+1} = h(f(\tilde{x}, w), 0)\}, \quad (22)$$

or, if no such solution exists, select \tilde{w}_k to minimize the norm of

$$y_{k+1} - h(f(\tilde{x}, w), 0), \quad (23)$$

and then plug in this value to get the state estimation error propagation

$$\tilde{x}_{k+1} = f(x_k, w_k) - f(\tilde{x}_k, \tilde{w}_k). \quad (24)$$

To judge if the error increases is not an easy task generally, but linearization can be applied to judge if there is a projected instability locally.

V. SIMULATION STUDY

In order to test when behaviors similar to the one in Section III occur, a simulation study was performed.

For this, we generated 1000 random two-dimensional systems with \bar{F} with eigenvalues between 0.5 and 2, and Q with norms between 1 and 200. The measurement noise R was kept at 1 for all systems. For all systems the state was initialized as $x_0 = \begin{pmatrix} 0 & 0 \end{pmatrix}^T$ and $P_0 = \mathbf{I}$. The matrices G and H were randomized to be unitary column and row matrices respectively, and for reasons which will be discussed later, selected such that $G = H^T$. Finally, F was required to be stable. The exact method can be seen in Algorithm 2.

Algorithm 2 System randomization

- 0: $\lambda_{\bar{F}} \sim U(0.5, 2)$
 $\log(Q_{size}) \sim U(0, \log(200))$
 where U is a uniform distribution.
 - 1: for $i, j = 1, 2$
 $F_{i,j} \sim \mathcal{N}(0, 1)$
 $H_i \sim \mathcal{N}(0, 1)$
 - 2: $H \leftarrow \frac{H}{\|H\|}$
 - 3: $G \leftarrow H^T$
 - 4: $Q \leftarrow Q_{size} G * G^T$
 - 5: $\bar{F} \leftarrow F - G(HG)^\dagger H F$
 - 6: $F \leftarrow \frac{\lambda_{\bar{F}}}{\max(|\lambda(\bar{F})|)} \bar{F}$
 - 7: If the new F is unstable, return to 1.
-

To start with, each system and process noise combination was simulated 100 times for each of 21 particle numbers logarithmically spread between 10 and 1000.

While above, the behavior is shown by the MSE increasing, it is also possible that instead the filtering attempts diverge completely (all weights becoming 0), and the likelihood of this occurring increasing as the number of particles increases. In this case, observing the increase in MSE is difficult as not all simulations even result in an MSE. As such, two criterion were selected to determine if the behavior occurred.

The first was whether the MSE increased. If the average MSE of at least 2 particle numbers greater than 10 is at least 30% larger than the MSE for 10 particles, the behaviour is said to be displayed. If, for any system, there were particle numbers for which the MSE was larger than it was for 10 particles, but the increase was less than 30%, a more thorough investigation with 1000 simulations was performed and the threshold was reduced to 10%.

The second criterion was instead to see if the number of diverged filtering attempts increased. The test was performed

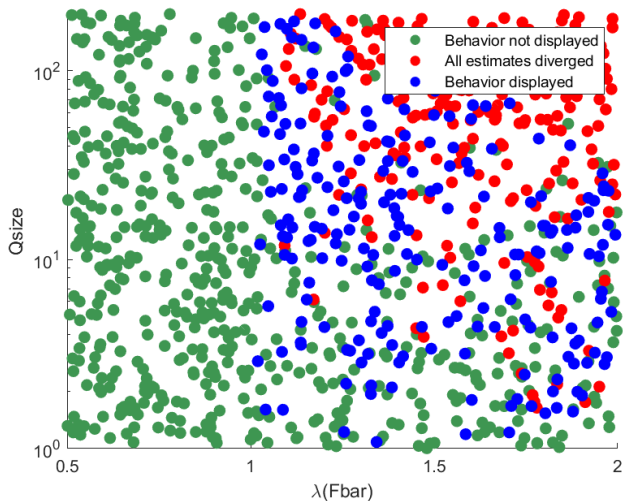


Fig. 4: The PF behaviors for various simulated systems

similarly to the MSE test, but we were instead interested in seeing if, for at least 2 of the simulated particle numbers, the percentage of diverged estimates was at least 30 percentage points larger than it was for 10 particles. As above, if the number increased but did not reach the threshold of 30 percentage points, a new test with 1000 simulations was performed and the threshold was reduced to 10 percentage points.

If either of these criterion were met, the system was deemed to show the described behavior. We have further separated systems for which more than 70% of all filtering attempts diverged for both 10 and 100 particles.

The result can be seen in Fig. 4. First off, it is clear that, independently of the size of Q , no PF displays the behavior or diverges for a system with $\lambda(\bar{F}) < 1$. Secondly, while this behavior is most commonly found in systems with a much smaller measurement noise than process noise, it is also possible for it to occur in systems with a more normal combination of measurement and process noise. Finally, for systems where $\lambda(\bar{F}) > 1$, increasing the size of Q or further increasing $\lambda(\bar{F})$ results in first the studied behavior to increase in likelihood, and then in the likelihood of the filter diverging completely to increase.

An important take away is that both the described behavior and all filters diverging entirely is an outcome which requires the use of much more particles to get an accurate estimate, both of which only occur when \bar{F} is unstable.

VI. ANALYSIS

While we cannot present a proof at this moment, the results in Section V strongly indicate that the projected instability is a necessary condition for this behavior to occur. It is however also shown that it is not a sufficient condition. We will first provide an argument for what the projected instability does in this case, and then present and analyze some other factors which appear to also have an influence on whether the behavior is displayed.

A. Projected instability and the PF

The process described for the projected instability is what would occur if the PF at each point selected the process noise which would lead to the largest weight after the measurement update. Under certain circumstances the PF does something similar to that process. Assuming that multiple particles have been assigned a certain state, these will next be propagated in time with different realizations of the process noise. The particles very close to the hyperplane corresponding to noise-free measurements will be assigned a high weight and therefore kept, while the ones far from it will be given a low weight and discarded in the resampling. This is illustrated in Figs. 5a and 5b. So long as this continues, the error in the particles will grow increasingly larger.

What is limiting this process from spiralling towards infinity is that the further the estimates get from the true state, the larger the process noise will need to be in order to move the particles to the hyperplane. This means that it becomes less likely that any process noise realisation of the PF results in a particle near the hyperplane. The end result of this process will resemble Figs. 5c and 5d, where no particle reaches the hyperplane and the closest one is the only one being kept. When the number of particles is increased, it is more likely that one of the particles appears close to the hyperplane, meaning that the estimate can move further away from the true state.

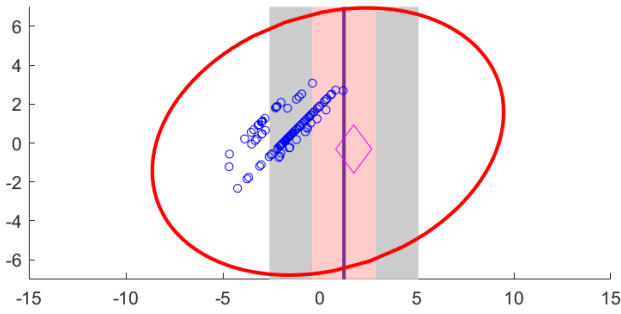
This process however, only occurs when the PF suffers from particle depletion. If at all times, enough particles exist which are close enough to the true measurement to be kept, then the ones which are closer in the unmeasured state will be more likely to have process noise realisations which arrive at the new hyperplane. As such, once the number of particles is large enough, the depletion stops occurring and the PF provides genuinely good results.

B. Noise requirements

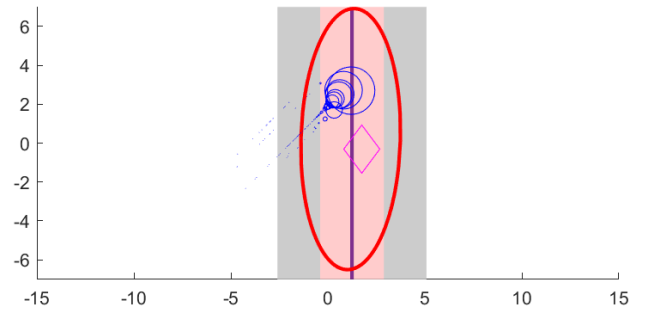
As was observed in Section V, the likelihood of the behavior occurring increases as the proportion of the process noise relative to the measurement noise increases up to the point where the filter starts to diverge.

Using the reasoning in Section VI-A, if the measurement noise is smaller than the process noise, only the particles generated relatively close to the hyperplane corresponding to noise-free measurements will have weights which are not very low, and as such these are the only ones being kept during the resampling step. This is what causes the PF to project the particles onto the hyperplane to begin with. If the covariance of the measurement noise is larger than the process noise, even particles generated relatively far from the hyperplane may be kept, which means that the assumption that the particles are propagated onto the hyperplane does not hold.

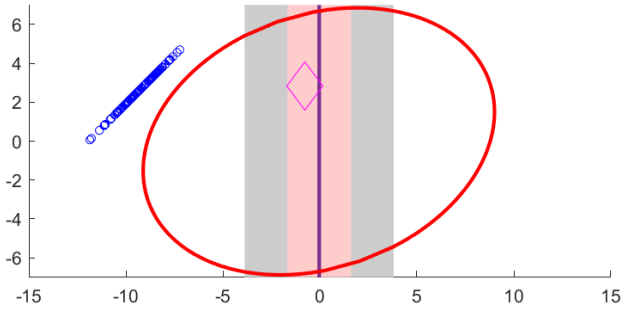
Connected to the requirement on the noise above is the requirement on the direction of the process noise. For the random simulations, the direction of the process noise, G was selected to be H^T . This was based on the reasoning that, in addition to the above requirements, H and G must be such that different w_{k-1} result in large changes in y_k . This occurs most



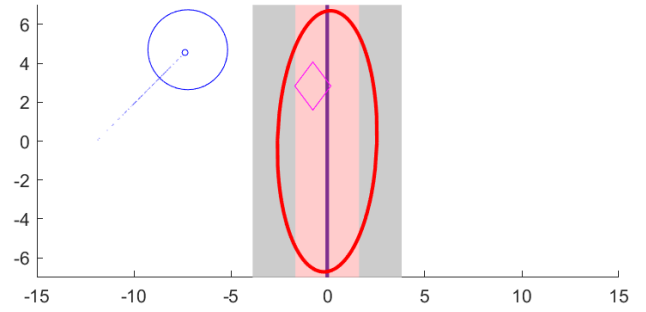
(a) The particles at step 16 before the measurement update.



(b) The particles at step 16 after the measurement update.



(c) The particles at step 39 before the measurement update.



(d) The particles at step 39 after the measurement update.

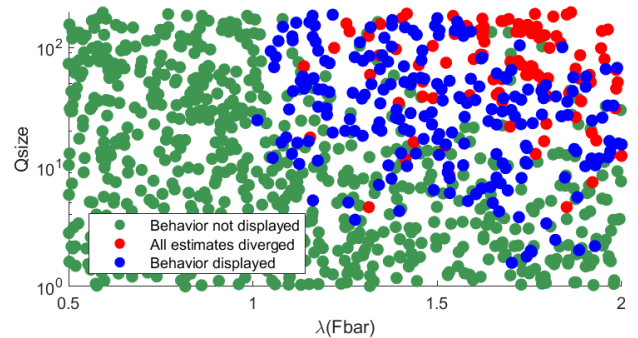
Fig. 5: An illustration of the PF approximation of the state distribution with 100 particles. The blue dots represent the particles, with the size of the dot representing the weight, and the magenta diamond represents the true state. The purple line represents the states corresponding to noise-free measurements and the grey and pink regions correspond to the 95% and 80% confidence region respectively of the current measurement. The red ellipsoid corresponds to the true 95% confidence region given all available measurements.

strongly if the process noise is parallel to the measurement, meaning $G = H^T$. If they are close to orthogonal, all realizations of the process noise will result in approximately the same weight, meaning that the measurement update again will not propagate the particles onto the hyperplane. Any small angle between the measurement and the process noise direction will however suffice, as can be seen with, system 1 in (8).

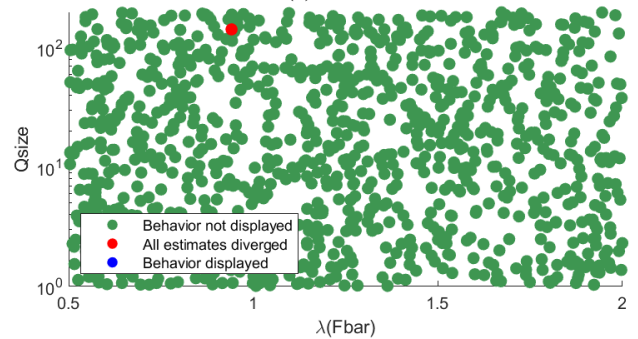
To further test this, systems were generated in the same way as in Section V, with different angles between G and H^T . The results can be seen in Fig. 6. For a 45° angle, the system still often displays the behavior, though it now requires a larger $\lambda(\bar{F})$ or norm of Q than previously for it to occur. For a 90° angle, no system displayed the behavior, and only one system diverged which we believe to be an outlier.

C. Stability

So far, we have always assumed the underlying system to be stable. This is also important to explain the behaviors observed above. The result is that a realization of the system will stay somewhat close to $\mathbf{0}$ at all times. This holds for both the true state, and would hold for the particles, if they were not projected onto the hyperplane by the combined measurement and resampling step. A PF with very few particles would be less likely to generate any particles close to the hyperplane, and the particles would therefore behave closer to a random realization which will result in them spreading around $\mathbf{0}$. Since



(a) 45°



(b) 90°

Fig. 6: The PF behaviors for various simulated systems, with different angles between G and H^T

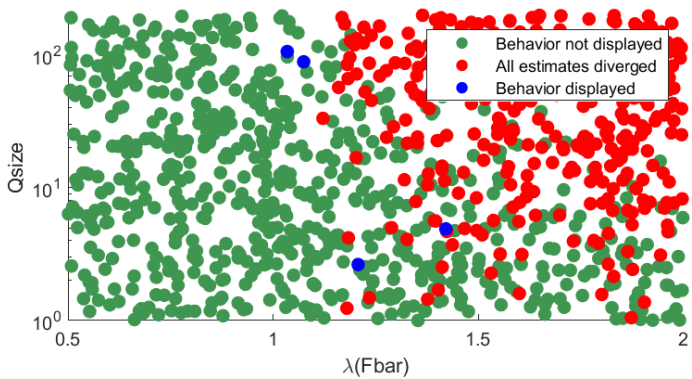


Fig. 7: The PF behaviors for simulated systems with optimal proposal

this is a stable system, this results in an MSE which is not that high. Increasing the number of particles somewhat instead results in the unstable behaviour described above, which takes the estimate further from 0. If the system was not stable, the error for this estimate would diverge, since there is no guarantee that the true state and randomly propagated particles would end up anywhere near each other.

VII. OPTIMAL PROPOSAL

It is also of interest to see how different proposals interact with a projected instability. For this reason we also attempted a run where we instead used the optimal proposal as defined in [11]

$$q(x_k | \bar{x}_{k-1}^{(i)}, y_k) = p(x_k | \bar{x}_{k-1}^{(i)}, y_k), \quad (25)$$

which is easy to compute for a linear system.

The same study as shown in Section V was performed with the optimal proposal, and the results can be seen in Fig. 7. From this, we can see that while the PF now rarely displays the studied behavior, it still has big issues with diverging filters when the system contains a projected instability. This is also reasonable, as the cause for the MSE to increase was that it was unlikely for any particle to be propagated close to the measurement for low numbers of particles with the old proposal. With the optimal proposal particles will be more likely to be placed close to the measurements, meaning that the filter is more likely to diverge instead.

VIII. CONCLUSION

We have in this work explored a strange behavior of the PF. We have introduced the condition of a projected instability, and shown that this condition can in many cases cause issues for the PF, potentially leading to an MSE which does not monotonically decrease, or filters diverging entirely. We have further shown how to calculate when this condition occurs in linear systems, and what other conditions play a part in the behavior occurring.

The main intention of this work is to provide more insight into the behavior of PFs at large, and possibly a new method for analyzing problems with a PF. While this work focuses on the linear case, the derivation of the condition can be easily

extended to nonlinear systems, though all actual calculations become very difficult. We also believe that the projected instability might be of interest in systems which are not stable, though it will likely only result in diverged filters.

A secondary intention is to give an explanation to others, should they come across a system which displays this behavior. In that case, as shown, while switching to an optimal proposal will likely remove said behavior, it is likely that the filter will diverge. While not studied as extensively, initial results indicate other proposals behaving similarly to either the standard proposal or the optimal one. The method which has seen the best performance has been marginalization, though this might not be possible in nonlinear systems.

REFERENCES

- [1] N. Gordon, D. Salmond, and A. Smith, "Novel approach to nonlinear/non-gaussian bayesian state estimation," *IEE Proc. F Radar Signal Process. UK*, vol. 140, no. 2, p. 107, 1993.
- [2] A. Doucet and A. Johansen, "A tutorial on particle filtering and smoothing: Fifteen years later," *Handbook of Nonlinear Filtering*, vol. 12, 01 2009.
- [3] F. Gustafsson, "Particle filter theory and practice with positioning applications," *IEEE Aerospace and Electronic Systems Magazine*, vol. 25, no. 7, pp. 53–82, 2010.
- [4] M. Speekenbrink, "A tutorial on particle filters," *Journal of Mathematical Psychology*, vol. 73, pp. 140–152, 2016.
- [5] P. Del Moral, "Nonlinear filtering: Interacting particle resolution," *Comptes Rendus de l'Académie des Sciences - Series I - Mathematics*, vol. 325, no. 6, pp. 653–658, 1997.
- [6] D. Crisan, P. Del Moral, and T. Lyons, "Discrete filtering using branching and interacting particle systems," *Markov Process. Related Fields*, vol. 5, 03 1999.
- [7] D. Crisan and A. Doucet, "A survey of convergence results on particle filtering methods for practitioners," *IEEE Transactions on Signal Processing*, vol. 50, no. 3, pp. 736–746, 2002.
- [8] J. Åslund, F. Gustafsson, and G. Hendeby, "On covariance matrix degeneration in marginalized particle filters with constant velocity models," in *25th International Conference on Information Fusion (FUSION)*, pp. 1–8, 2022.
- [9] J. Åslund, F. Gustafsson, and G. Hendeby, "When does the marginalized particle filter degenerate?," in *2023 26th International Conference on Information Fusion (FUSION)*, pp. 1–7, 2023.
- [10] A. Kong, J. S. Liu, and W. H. Wong, "Sequential imputations and bayesian missing data problems," *Journal of the American Statistical Association*, vol. 89, no. 425, pp. 278–288, 1994.
- [11] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.