Regular Article

Kalman-Takens filtering in the presence of dynamical noise

Franz Hamilton[1], Tyrus Berry[2], and Timothy Sauer[2,a]

[1] North Carolina State University, Raleigh, NC 27695, USA
[2] George Mason University, Fairfax, VA 22030, USA

Abstract. The use of data assimilation for the merging of observed data with dynamical models is becoming standard in modern physics. If a parametric model is known, methods such as Kalman filtering have been developed for this purpose. If no model is known, a hybrid Kalman-Takens method has been recently introduced, in order to exploit the advantages of optimal filtering in a nonparametric setting. This procedure replaces the parametric model with dynamics reconstructed from delay coordinates, while using the Kalman update formulation to assimilate new observations. In this article, we study the efficacy of this method for identifying underlying dynamics in the presence of dynamical noise. Furthermore, by combining the Kalman-Takens method with an adaptive filtering procedure we are able to estimate the statistics of the observational and dynamical noise. This solves a long-standing problem of separating dynamical and observational noise in time series data, which is especially challenging when no dynamical model is specified.

1 Introduction

Methods of data assimilation are heavily used in geophysics and have become common throughout physics and other sciences. When a parametric, physically motivated model is available, noise filtering and forecasting in a variety of applications are possible. Although the original Kalman filter [1] applies to linear systems, more recent approaches to filtering such as the Extended Kalman Filter (EKF) and Ensemble Kalman Filter (EnKF) [2–6] allow forecasting models to use the nonlinear model equations to compute predictions that are close to optimal.

In some cases, a model is not available, and in other cases, all available models may be inaccurate. In geophysical processes, basic principles may constrain a variable as a function of other driving variables in a known manner, but the driving variables may be unmodeled or modeled with large error [7–10]. Moreover, in numerical weather prediction codes, physics on the large scale is typically sparsely modeled [11,12]. Some recent work has considered the case where only a partial model is known [13,14].

Under circumstances in which models are not available, Takens' method of attractor reconstruction [15–18] has been used to reconstruct physical attractors from

---

a e-mail: tsauer@gmu.edu

data. The dynamical attractor is typically represented by vectors of delay coordinates constructed from time series observations, and approaches to prediction, control, and other time series applications have been proposed [19–21]. In particular, time series prediction algorithms locate the current position in the delay coordinate representation and use analogues from previously recorded data to establish a predictive statistical model [22–35]. However, Takens' method is designed (and proved to work) for strictly deterministic systems, and the effects of noise, both dynamic and observational, is not thoroughly understood.

Recently, a method was introduced that would merge Takens' nonparametric attractor reconstruction approach with Kalman filtering. Since the model equations governing the evolution of the system are unknown, the dynamics are reconstructed nonparametrically using delay-coordinate vectors, and used to replace the model. The Kalman-Takens algorithm [36] is able to filter noisy data with comparable performance to parametric filtering methods that have full access to the exact model. The fidelity of this algorithm follows from the fact that Takens' theorem [15] states roughly that in the large data limit, the equations can be replaced by the data. The surprising fact, shown in [36], is that this fidelity is robust to observational noise which invalidates the basic theory of [15], although a much more complex theory developed in [44] suggests such a robustness. In fact, by implementing the Kalman update, it is suggested in [36] that the nonparametric representation of the dynamics is able to handle substantial observational noise in the data.

In this article we study the effects of dynamic noise (also known as system noise or process noise) on the Kalman-Takens algorithm. We will verify the effectiveness of the method on stochastic differential equations with significant levels of system noise, and show that nonparametric prediction can be improved by the filter almost to the extent of matching the performance of the exact parametric model. In section Section 2 we present the specifics of the Kalman-Takens method. A key to effective application of any Kalman based algorithm is knowing the covariance matrices of the system and observation noise, which are particularly difficult to separate when no model is known. In Section 4 we present an adaptive method for separating the dynamic and observational noise as part of the filtering procedure. The adaptive filtering method naturally complements the Kalman-Takens method by implicitly fitting a fully nonparametric stochastic system to the data. These methods are combined in Section 5, where applications to dynamical data with variable settings of delay coordinates are explored.

2 Kalman-Takens filter

We recall the standard notion of the Kalman filter in the case where the model f and observation function g are known. Consider a nonlinear stochastic system with n-dimensional state vector x and m-dimensional observation vector y defined by

$$
\begin{aligned}
x_{k+1} &= f(x_k, t_k) + w_k \\
y_k &= g(x_k, t_k) + v_k
\end{aligned}
\tag{1}
$$

where w_k and v_k are white noise processes with covariance matrices Q and R, respectively. We begin by describing the filtering procedure in the case when f and g are known.

We are chiefly interested in nonlinear systems, so we will describe a version of Kalman filtering that is common in this case. The ensemble Kalman filter (EnKF) [2] represents a nonlinear system at a given instant as an ensemble of states. Here we initialize the filter with state vector $x_0^+ = 0_{n \times 1}$ and covariance matrix $P_0^+ = I_{n \times n}$.

At the kth step, the filter produces an estimate of the state x_{k-1}^+ and the covariance matrix P_{k-1}^+, which estimates the covariance of the error between the estimate x_{k-1}^+ and the true state. In the unscented version of EnKF [39], the singular value decomposition is used to find the symmetric positive definite square root S_{k-1}^+ of the matrix P_{k-1}^+, allowing us to form an ensemble of E state vectors, where the ith ensemble member is denoted $x_{i,k-1}^+$. The ensemble vectors $x_{i,k-1}^+$ are formed by adding and subtracting the columns of S_{k-1}^+ to the estimate x_{k-1}^+ to produce an ensemble with mean x_{k-1}^+ and covariance P_{k-1}^+ [4,5,39]. In other words, the ensemble statistics match the current filter estimates of the mean and covariance. The ensemble can also be rescaled by introducing weights for each ensemble member as described in [4,5]. Use of the unscented transformation results in an ensemble of $2n+1$ state vectors.

The model f is applied to the ensemble, advancing it one time step, and then observed with function g. The mean of the resulting state ensemble gives the prior state estimate x_k^- and the mean of the observed ensemble is the predicted observation y_k^-. Denoting the covariance matrices P_k^- and P_k^y of the resulting state and observed ensemble, and the cross-covariance matrix P_k^{xy} between the state and observed ensembles, we define

$$P_k^- = \frac{1}{E} \sum_{i=1}^{E} \left(x_{ik}^- - x_k^-\right)\left(x_{ik}^- - x_k^-\right)^T + Q$$

$$P_k^y = \frac{1}{E} \sum_{i=1}^{E} \left(y_{ik}^- - y_k^-\right)\left(y_{ik}^- - y_k^-\right)^T + R$$

$$P_k^{xy} = \frac{1}{E} \sum_{i=1}^{E} \left(x_{ik}^- - x_k^-\right)\left(y_{ik}^- - y_k^-\right)^T. \tag{2}$$

Given the observation y_k, the equations

$$K_k = P_k^{xy}(P_k^y)^{-1}$$
$$P_k^+ = P_k^- - P_k^{xy}(P_k^y)^{-1}P_k^{yx}$$
$$x_k^+ = x_k^- + K_k\left(y_k - y_k^-\right) \tag{3}$$

update the state and covariance estimates. We will refer to this as the parametric filter, since a full set of model equations are assumed to be known. The matrices Q and R are parameters which represent the covariance matrices of the system noise and the observation noise respectively. Often the true statistics of these noise processes are unknown, and examples in [40] have shown that accurate estimates of Q and R are crucial to obtaining a good estimate of the state. In Section 4, we describe an algorithm developed in [40] for adaptive estimation of Q and R which was developed for the case when the dynamical model f and observation function g are known.

Contrary to (1), our assumption in this article is that neither the model f nor observation function g are known, making outright implementation of the EnKF impossible. Instead, the filter described here does not require a model while still leveraging the Kalman update described in (3). The idea is to replace the system evolution, traditionally done through application of f, with advancement of dynamics nonparametrically using delay-coordinate vectors. We describe this method with the assumption that a single variable is observable, say y, but the algorithm can be easily generalized to multiple system observables. In addition, we will assume in our

examples that noise covariances Q and R are unknown and will be updated adaptively as in [40].

The idea behind Takens' method is that given the observable y_k, the delay-coordinate vector

$$x_k = [y_k, y_{k-1}, \ldots, y_{k-d}]$$

accurately captures the *state* of the dynamical system, where d is the number of delays. In the case where the noise variables are removed from (1), the vector x_k provably represents the state of the system for $d + 1 > 2n$ as shown in [15,17] (note that the embedding dimension is $d + 1$ since d delays are added to the original state). An extension of this result to the stochastic system (1) can be found in [44]. Current applications of Takens delay-coordinate reconstruction have been restricted to forecasting; typically by finding historical delay-coordinate states which are close to the current delay-coordinate state and interpolating these historical trajectories. A significant challenge in this method is finding 'good' neighbors, which accurately mirror the current state, especially in real applications when both the current and historical states are corrupted by observational noise. The goal of the Kalman-Takens filter is to quantify the uncertainty in the state and reduce the noise using the Kalman filter.

In order to integrate the Takens reconstruction into the Kalman filter, at each step of the filter an ensemble of delay vectors is formed. The advancement of this ensemble forward in time requires a nonparametric technique to serve as a local proxy $\tilde{f}$ for the unknown model f. Given a specific delay coordinate vector $x_k = [y_k, y_{k-1}, \ldots, y_{k-d}]$, we locate its N nearest neighbors (with respect to Euclidean distance) $x_{i_1}, \ldots, x_{i_N}$ where

$$x_{i_j} = [y_{i_j}, y_{i_j-1}, \ldots, y_{i_j-d}]$$

are found from the noisy training data. Once the neighbors are found, the known one step forecast values $y_{i_1+1}, y_{i_2+1}, \ldots, y_{i_N+1}$ are used with a local model to predict y_{k+1}. In this article, we use a locally constant model which in its most basic form is an average of the nearest neighbors, namely

$$\tilde{f}(x_k) = \left[\frac{y_{i_1+1} + y_{i_2+1} + \ldots + y_{i_N+1}}{N}, y_k, \ldots, y_{k-d+1} \right]. \tag{4}$$

This process is repeated to compute the one step forecast $\tilde{f}$ applied to each member of the ensemble. Once the full ensemble has been advanced forward in time by $\tilde{f}$, the remaining EnKF algorithm is then applied using equation (2) as described above, and the delay-coordinate state vector is updated according to (3). This method was called the *Kalman-Takens filter* in [36].

The prediction (4) can be further refined by considering a weighted average of the nearest neighbors. Throughout the following examples, the weight for neighbor x_{i_j} was defined as

$$w_{i_j} = \frac{e^{-d_{i_j}^2/\sigma^2}}{\sum_{j=1}^{N} e^{-d_{i_j}^2/\sigma^2}}$$

where d_{i_j} is the distance of x_{i_j} to the current delay-coordinate vector and σ is a bandwidth parameter which controls the contribution of each neighbor in the local model. In our results, we set the bandwidth $\sigma = 2$ and use 20 neighbors. We found these parameter settings to be adequate for filtering the data in our examples, although the
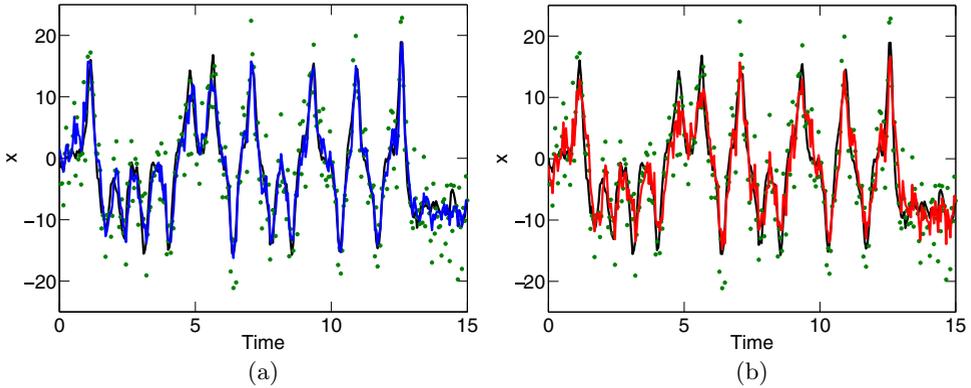
Fig. 1. Filtering comparison of the Lorenz-63 x variable when the system is affected by Gaussian dynamic noise with total variance of 2.4 (black solid line). Observations (green circles) of the stochastic signal perturbed by Gaussian observational noise with variance of 20 (signal RMSE of 4.49). (a) Parametric filter reconstruction (solid blue line) results in an RMSE of 2.34. (b) Kalman-Takens filter with $d = 2$ delays (solid red curve) results in RMSE of 2.95.

results do not change significantly for similar settings. In general, the bandwidth σ could be chosen to be the mean distance to the kth nearest neighbor for some $k < N$.

3 Filtering in the presence of dynamic noise

It was demonstrated in [36] that the Kalman-Takens filter has significant power to reduce observational noise, but no attempt was made to evaluate its effectiveness to handle more general noise. In fact, there is reason to doubt its potential against dynamic noise, due to the fact that as the delay time increases, the fidelity of near neighbors to the current delay coordinate vector may be compromised. In this section, we investigate whether this filter can successfully reconstruct the state under observational and dynamic noise.

A relevant example is an SDE based on the Lorenz-63 system [41]

$$
\begin{aligned}
\dot{x} &= \sigma(y - x) + \xi \dot{W}_x \\
\dot{y} &= x(\rho - z) - y + \xi \dot{W}_y \\
\dot{z} &= xy - \beta z + \xi \dot{W}_z
\end{aligned}
\tag{5}
$$

where $\sigma = 10$, $\rho = 28$, $\beta = 8/3$ and $\dot{W}$ is white noise with unit variance. Assume we have a noisy set of training data points

$$
y(t_k) = x(t_k) + \eta_k
$$

where $k = 1, 2, \ldots, M$ and $y_k = y(t_k)$ is a direct observation of the x variable corrupted by independent Gaussian perturbations η_k with mean zero and variance R^o. Using this data, we want to develop a nonparametric forecasting method to predict future x values of the system. However, due to the presence of the noise η_k, outright application of a prediction method leads to inaccurate forecasts.

If knowledge of (5) were available, the standard parametric filter could be used to assimilate the noisy x observations to the model, generate a denoised estimate of the x variable, and simultaneously estimate the unobserved y and z variables. This
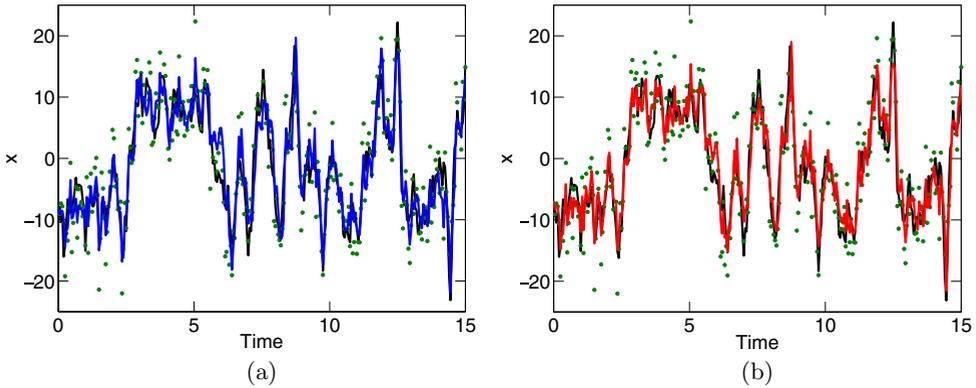
Fig. 2. Filtering comparison of the Lorenz-63 x variable when the system is affected by Gaussian dynamical noise with total variance of 15 (black solid line). Observations (green circles) of the stochastic signal perturbed by Gaussian observational noise with variance of 20 (signal RMSE of 4.49). (a) Parametric filter reconstruction (solid blue line) results in an RMSE of 3.21. (b) Kalman-Takens filter with $d = 2$ delays (solid red curve) results in RMSE of 3.29.

denoised state estimate could then be forecast forward in time using (5), and we refer to this process and the parametric forecast.

In contrast, the Kalman-Takens method assumes no knowledge of the underlying dynamics (5) or even the observation function. Instead, the Kalman-Takens method works directly with the noise observations y_k and defines a proxy for the underlying state by the delay vector $[y_k, y_{k-1}, \ldots, y_{k-d}]$. By applying the Kalman-Takens method to the entire training data set, we reduce the observation noise in the training data which will improve future filtering and also provide better neighbors to improve forecasting.

Figure 1 shows a comparison of ensemble Kalman filtering with and without the model. Figure 1a shows the standard parametric EnKF applied to the x-coordinate of the Lorenz SDE (5) with system noise variance $\xi^2 = 0.8$ (the total variance of the system noise is 2.4 since there are three independent noise variables), and observational noise variance $R^o = 20$. Figure 1b shows the Kalman-Takens filter applied to the same data. Compared to the SDE solution $x(t_k)$ without observational noise (black curve), the parametric filter fits slightly better, but the Kalman-Takens filter does almost as well without knowing the model equations. A training set of $M = 8000$ points were used. Figure 2 shows the same comparison with a higher level of system noise $\xi^2 = 5$ (total system noise variance is 15).

4 Adaptive estimation of Q and R

A persistent problem in time series analysis is distinguishing system noise from observational noise. This is a particularly important issue since observational noise distorts the forecast, and needs to be removed, whereas the system noise affects the future of the system and thus should not be removed. In other words, the state estimate that will give the optimal forecast includes the system noise perturbation, but does not include any of the observation noise perturbations. While obtaining this perfect estimate is typically impossible, getting as close as possible requires knowing the statistics and correlations of the two noise processes. This is captured in the structure of the Kalman equations (2) and (3) which gives the provably optimal estimate of the state for linear systems with additive Gaussian system and observation noise. The
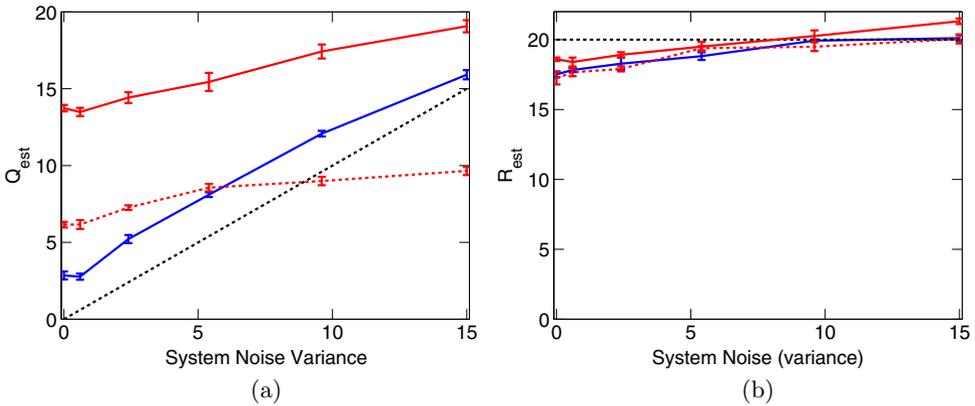
Fig. 3. Adaptive estimation of the filter Q and R matrices. Parametric filter (solid blue curve), Kalman-Takens with 2 delays (solid red curve), Kalman-Takens with 4 delays (dotted red curve) (a) System noise estimate Q_{est} (b) Observation noise estimate R_{est}.

presence of the noise covariance matrices Q and R in the Kalman equations shows how these parameters determine the optimal estimate.

Since we cannot assume that the noise covariance matrices Q and R are known, we used a recently-developed method [40] for the adaptive fitting of these matrices as part of the filtering algorithm. The key difficulty in estimating these covariances is disambiguating the two noise sources. The method of [40] is based on the fact that system noise perturbations affect the state at future times, whereas observation noise perturbations only affect the current time.

The method uses the innovations $\epsilon_k \equiv y_k - y_k^-$ in observation space from (2) to update the estimates Q_k and R_k of the covariances Q and R, respectively, at step k of the filter. In order to estimate these two quantities, we will compute two statistics of the innovations. The first statistic is the outer product $\epsilon_k \epsilon_k^\top$ and the second statistic is the lagged outer product $\epsilon_k \epsilon_{k-1}^\top$. Intuitively, the first statistic only includes information about the observation noise covariance, whereas the second statistic contains information about the system noise covariance.

We produce empirical estimates Q_{k-1}^e and R_{k-1}^e of Q and R based on the innovations at time k and $k-1$ using the formulas

$$P_{k-1}^e = F_{k-1}^{-1} H_k^{-1} \epsilon_k \epsilon_{k-1}^T H_{k-1}^{-T} + K_{k-1} \epsilon_{k-1} \epsilon_{k-1}^T H_{k-1}^{-T}$$
$$Q_{k-1}^e = P_{k-1}^e - F_{k-2} P_{k-2}^a F_{k-2}^T$$
$$R_{k-1}^e = \epsilon_{k-1} \epsilon_{k-1}^T - H_{k-1} P_{k-1}^f H_{k-1}^T. \tag{6}$$

It was shown in [40] that P_{k-1}^e is an empirical estimate of the background covariance at time index $k-1$. Notice that (6) requires local linearizations F_{k-1} of the dynamics and H_{k-1} of the observation function. While there are many methods of finding these linearizations, we use the method introduced in [40] and used in [36] which is based on a linear regression. In particular, to determine F_{k-1} a linear regression is applied between the ensemble before and after the dynamics $\tilde{f}$ are applied. Similarly, to determine H_{k-1} a linear regression is applied between the ensemble before and after the observation function is applied. Notice that this procedure requires us to save the linearizations $F_{k-2}, F_{k-1}, H_{k-1}, H_k$, innovations $\epsilon_{k-1}, \epsilon_k$, and the analysis P_{k-2}^a and Kalman gain matrix, K_{k-1}, from the $k-1$ and $k-2$ steps of the EnKF.

To find stable estimates of Q and R we combine the noisy estimates Q_{k-1}^e and R_{k-1}^e (which are also low-rank by construction) using an exponential moving average

$$Q_k = Q_{k-1} + (Q_{k-1}^e - Q_{k-1})/\tau$$
$$R_k = R_{k-1} + (R_{k-1}^e - R_{k-1})/\tau \qquad (7)$$

where τ is the window of the moving average. For further details on the estimation of Q and R we refer the reader to [36,40].

In Figure 3 we show the final estimates trace(Q_{est}) and R_{est} from applying the adaptive filter to the stochastic Lorenz system (5) with various system noise levels $3\xi^2$ (note that the factor 3 gives the total variance of the three stochastic forcing variables which is compared to trace(Q_{est}) which is also a total variance). In Figure 3b we show that all of the filter methods obtain reasonably accurate estimates of the true observation noise $R^o = 20$, and the results are very robust to the amount of system noise. In Figure 3a we first note that the parametric filter with the perfect model obtains a good estimate (blue, solid curve) of the true system noise (black, dashed curve). Secondly, we note that the trace of Q_{est} for the Kalman-Takens filters both increase with increasing system noise, without distorting the estimate of the observation noise (as shown in Fig. 3b). Notice that the Takens reconstruction corresponds to a highly nonlinear transformation of the state space, and that even the dimension of the state space changes. Since this nonlinear transformation may stretch or contract the state space in a complicated way, it is not particularly surprising that the variance of the stochastic forcing in the delay-embedding space has a different magnitude than the original system noise variance. The exact relationship between these two stochastic forcings is highly nontrivial, and the spatially homogeneous and uncorrelated system noise of (5) may easily become inhomogeneous and correlated in the reconstructed dynamics.

5 Filtering dynamical noise

The Kalman-Takens filter was introduced in [36], which considered deterministic dynamical systems with only observation noise. In section 2, we showed that the filter is also robust to dynamical noise, and in this section we quantify this fact for stochastic systems such as (5) that include both system and observation noise. In Figure 4 we compare the Kalman-Takens filters (red curves) to the parametric filter (blue curve) as a function of the system noise variance $3\xi^2$ for various levels of observation noise. In Figure 4 the observation noise, which is the error between the observed signal and the true state, is denoted by the black dotted line in each subfigure. Each filter obtains a dramatic denoising of the observations relative to the observation noise levels $R^o = 5, 20$, and 60 in Figures 4a–4c respectively. In each case, for large observation noise the Kalman-Takens filter performance is comparable to the parametric filter, and only at very low system noise does the parametric filter have a slight advantage. This illustrates the robustness that the Kalman-Takens filter has to model error since it makes no assumption on the form of the model, unlike the parametric filter which possesses the perfect model.

We also explored the robustness of the Kalman-Takens filter to the number of delays d used in the Takens delay-embedding. For $d = 2$ the embedding dimension is $d + 1 = 3$, which is the minimum dimension needed to embed the attractor of the deterministic Lorenz-63 system. However, the theoretical guarantee of the Takens embedding theory requires $d + 1 > 2n$ where n is the attractor dimension, which requires an embedding dimension of $d + 1 = 5$ (the attractor of the deterministic Lorenz-63 system is slightly larger than 2). As shown in Figures 4a and 4b the $d = 2$
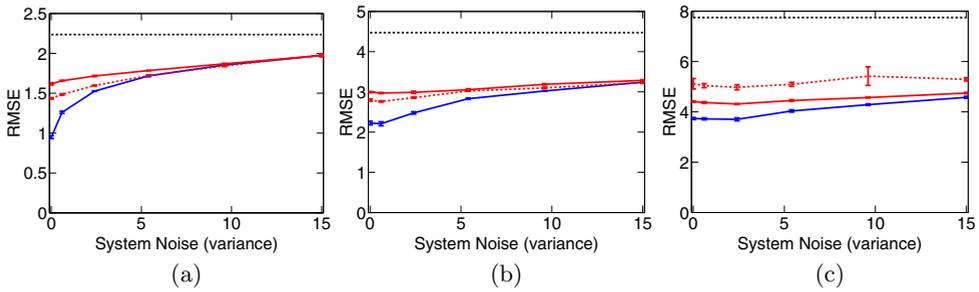
Fig. 4. Reconstructing the noisy Lorenz-63 x variable under increasing levels of dynamical noise. Observations of stochastic signal perturbed by Gaussian observational noise with variance (a) 5, (b) 20 and (c) 60 resulting in signal error level (dotted black line). Parametric filter (solid blue line), Kalman-Takens filter with 2 delays (solid red line) and Kalman-Takens filter with 4 delays (dotted red line) reconstruction accuracy shown. Error bars denote standard error over 5 realizations. As the amount of dynamical noise increases, the Kalman-Takens filter has performance very similar to the parametric filter, which has access to the full model.

(red, solid curve) and $d = 4$ (red, dotted curve) have similar performance except at very low system noise levels, where $d = 4$ has a slight advantage. Notice that when very little noise is present, the longer that two trajectories agree, the closer the corresponding states are, so for small noise, we expect long delays to help find better neighbors. However, in the presence of system noise, this breaks down. Two trajectories may be very close in the past but recent stochastic forcing may cause them to quickly diverge. Moreover, as noise is added, it becomes increasingly difficult to distinguish states, and adding delays is merely adding more dimensions which could coincidentally agree without implying the states are similar. This is shown in Figure 4c where $d = 2$ outperforms $d = 4$ in the presence of large observation noise. Similarly, in all of Figure 4 we note that as the system noise increases, the value of knowing the true model (5) becomes negligible as the $d = 2$ Kalman-Takens performance is indistinguishable from the parametric filter.

Finally, we show in Figure 5 that the Kalman-Takens filter achieves forecast performance comparable to the perfect model. The Kalman-Takens improves forecasting in two ways. First, by running the Kalman-Takens filter on the historical training data, we significantly reduce the observation noise, which allows for better analog forecasting. Second, the Kalman-Takens filter gives a good estimate of the current state, which leads to an improved forecast. Ultimately, Takens based forecasting relies on finding good neighbors in delay-embedding space since these neighbors are interpolated to form the forecast. Finding good neighbors requires both a good estimate of the current state, and historical data with small observation noise; and the Kalman-Takens filter improves both of these.

As a baseline, we used the unfiltered delay-embedding and found neighbors in the raw delay-embedded historical data, and the RMSE of this forecast method is shown in Figure 5 (black solid curve) as a function of the forecast horizon. The time zero forecast is simply the initial estimate, and for the black solid curve the RMSE at time zero corresponds to the observation noise level (since the initial state is simply the unfiltered current observation). The lower RMSE of the filtered estimates (red and blue curves) at time zero in Figure 5 shows how the filter improves the initial estimate of the state. Similarly, the long term forecast converges to the average of the historical training data (since we use an ensemble forecast which are uncorrelated in the long term). The lower RMSE of the filtered forecasts (red curves) compared to the raw data (black curve) in Figure 5 shows that denoising the historical data
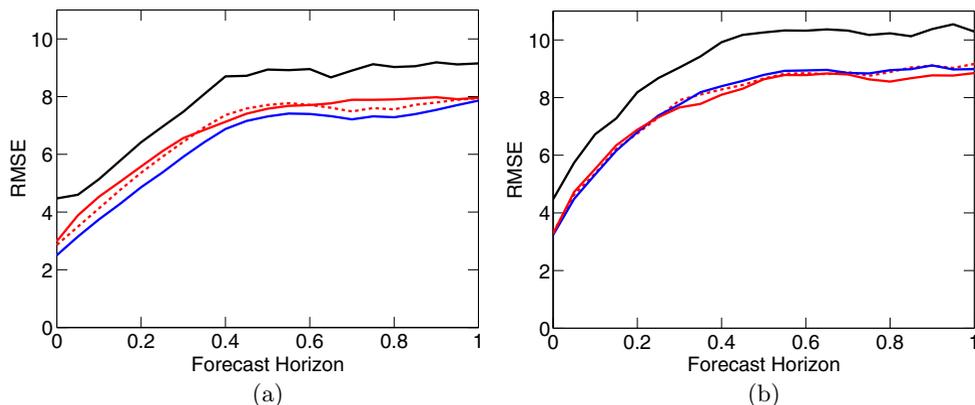
Fig. 5. Forecast accuracy when predicting Lorenz-63 from 8000 noisy training data. (a) System influenced by Gaussian dynamical noise with variance of 0.8. (b) System influenced by Gaussian dynamical noise with variance of 5. Observations perturbed by Gaussian observational noise with variance of 20. Prediction results shown when the training data are not filtered (solid black curve), filtered by the parametric filter (solid blue curve), Kalman-Takens filter with 2 delays (solid red curve) and Kalman-Takens filter with 4 delays (dotted red curve) is used. Results averaged over 1000 realizations.

gives an improved estimate of the long term statistics of the SDE. Moreover, since the red curves are very close to the blue curve in Figure 5 we conclude that the Kalman-Takens is able to improve the historical data and estimate the current state with sufficient accuracy to match the parametric filter and forecast using the perfect model.

6 Summary

Traditional data assimilation is predicated on the existence of an accurate model for the dynamics. In this article, we have shown that the Kalman-Takens filter provides an alternative for stochastic dynamical systems when no model is available. In this approach, the usual role of a fully parametrized model in linear and nonlinear Kalman filtering algorithms is subsumed by a Takens delay-coordinate reconstruction of the dynamics.

Although it might be expected that dynamical noise would be an obstruction to the delay-coordinate embedding that is necessary to reconstruct the attractor, we find by numerical experiment that filtering and forecasting applications are not hampered significantly more than for the parametric filter. In fact, for the stochastic Lorenz attractor, we show that prediction accuracy with Kalman-Takens filtering can approximately match parametric results.

This report is a feasibility study, and leaves open several interesting questions about how to optimally apply the algorithm. In particular, the role of the number of delays, neighborhood size, as well as their relation to the EnKF parameters are still not well understood, and are deserving of further investigation.

References

1. R. Kalman, J. Basic Eng. **82**, 35 (1960)
2. E. Kalnay, *Atmospheric modeling, data assimilation, and predictability* (Cambridge Univ. Press, 2003)
3. G. Evensen, *Data assimilation: the ensemble Kalman filter* (Springer, Heidelberg, 2009)
4. S. Julier, J. Uhlmann, H. Durrant-Whyte, IEEE Trans. Automat. Control **45**, 477 (2000)
5. S. Julier, J. Uhlmann, H. Durrant-Whyte, Proc. IEEE **92**, 401 (2004)
6. S. Schiff, *Neural control engineering* (MIT Press, 2012)
7. R.H. Reichle, R.D. Koster, Geophys. Res. Lett. **32**, 102404 (2005)
8. H. Hersbach, A. Stoffelen, S. De Haan, J. Geophys. Res.: Oceans (1978–2012) **112**, C03006 (2007)
9. H. Arnold, I. Moroz, T. Palmer, Philos. Trans. R. Soc. A: Math. Phys. Eng. Sci. **371**, 20110479 (2013)
10. T. Berry, J. Harlim, Proc. R. Soc. A: Math. Phys. Eng. Sci. **470**, 20140168 (2014)
11. E.N. Lorenz, K.A. Emanuel, J. Atmos. Sci. **55**, 399 (1998)
12. T.N. Palmer, Q. J. R. Met. Soc. **127**, 279 (2001)
13. F. Hamilton, T. Berry, T. Sauer, Phys. Rev. E **92**, 010902 (2015)
14. T. Berry, J. Harlim, J. Comput. Phys. **308**, 305 (2016)
15. F. Takens, in *Lecture Notes in Math* (Springer-Verlag, Berlin, 1981), Vol. 898
16. N. Packard, J. Crutchfield, D. Farmer, R. Shaw, Phys. Rev. Lett. **45**, 712 (1980)
17. T. Sauer, J. Yorke, M. Casdagli, J. Stat. Phys. **65**, 579 (1991)
18. T. Sauer, Phys. Rev. Lett. **93**, 198701 (2004)
19. E. Ott, T. Sauer, J.A. Yorke, *Coping with chaos: analysis of chaotic data and the exploitation of chaotic systems* (Wiley, New York, 1994)
20. H. Abarbanel, *Analysis of observed chaotic data* (Springer-Verlag, New York, 1996)
21. H. Kantz, T. Schreiber, *Nonlinear time series analysis* (Cambridge University Press, 2004)
22. J. Farmer, J. Sidorowich, Phys. Rev. Lett. **59**, 845 (1987)
23. M. Casdagli, Physica D: Nonlinear Phenom. **35**, 335 (1989)
24. G. Sugihara, R.M. May, Nature **344**, 734 (1990)
25. L.A. Smith, Physica D: Nonlinear Phenom. **58**, 50 (1992)
26. J. Jimenez, J. Moreno, G. Ruggeri, Phys. Rev. A **45**, 3553 (1992)
27. T. Sauer, in *Time series prediction: forecasting the future and understanding the past* (Addison-Wesley, 1994), pp. 175–193
28. G. Sugihara, Philos. Trans. R. Soc. A: Math. Phys. Eng. Sci. **348**, 477 (1994)
29. C. G. Schroer, T. Sauer, E. Ott, J.A. Yorke, Phys. Rev. Lett. **80,** 1410 1998
30. D. Kugiumtzis, O. Lingjaerde, N. Christophersen, Physica D: Nonlinear Phenom. **112**, 344 (1998)
31. G. Yuan, M. Lozier, L. Pratt, C. Jones, K. Helfrich, J. Geophys. Res. **109**, C08002 (2004)
32. C.-H. Hsieh, S.M. Glaser, A.J. Lucas, G. Sugihara, Nature **435**, 336 (2005)
33. C.C. Strelioff, A.W. Hubler, Phys. Rev. Lett. **96**, 044101 (2006)
34. S. Regonda, B. Rajagopalan, U. Lall, M. Clark, Y.-I. Moon, Nonlinear Proc. Geophys. **12**, 397 (2005)
35. B. Schelter, M. Winterhalder, J. Timmer, *Handbook of time series analysis: recent theoretical developments and applications* (John Wiley and Sons, 2006)
36. F. Hamilton, T. Berry, T. Sauer, Phys. Rev. X **6**, 011021 (2016)
37. T. Berry, D. Giannakis, J. Harlim, Phys. Rev. E **91**, 032915 (2015)
38. T. Berry, J. Harlim, Physica D **320**, 57 (2016)
39. D. Simon, *Optimal state estimation: Kalman, H^∞ and nonlinear approaches* (John Wiley and Sons, 2006)
40. T. Berry, T. Sauer, Tellus A **65**, 20331 (2013)
41. E. Lorenz, J. Atmos. Sci. **20**, 130 (1963)
42. E.N. Lorenz, in *Proceedings: Seminar on predictability* (AMS, Reading, UK, 1996), pp. 1–18

43. A. Sitz, U. Schwarz, J. Kurths, H. Voss, Phys. Rev. E **66**, 16210 (2002)
44. J. Stark, D.S. Broomhead, M.E. Davies, J. Huke, J. Nonlinear Sci. **13**, 519 (2003)