

Article

Optimal Shadowing Filter for a Positioning and Tracking Methodology with Limited Information

Ayham Zaitouny ^{1,2,*} , Thomas Stemler ¹  and Shannon Dee Algar ¹

¹ Department of Mathematics and Statistics, University of Western Australia, 35 Stirling Highway, Crawley, WA 6009, Australia; thomas.stemler@uwa.edu.au (T.S.); shannon.algar@research.uwa.edu.au (S.D.A.)

² Commonwealth Scientific and Industrial Research Organisation, 26 Dick Perry Ave, Kensington, WA 6151, Australia

* Correspondence: ayham.zaitouny@uwa.edu.au or ayham.zaitouny@csiro.au

Received: 19 January 2019; Accepted: 18 February 2019; Published: 22 February 2019



Abstract: Positioning and tracking a moving target from limited positional information is a frequently-encountered problem. For given noisy observations of the target's position, one wants to estimate the true trajectory and reconstruct the full phase space including velocity and acceleration. The shadowing filter offers a robust methodology to achieve such an estimation and reconstruction. Here, we highlight and validate important merits of this methodology for real-life applications. In particular, we explore the filter's performance when dealing with correlated or uncorrelated noise, irregular sampling in time and how it can be optimised even when the true dynamics of the system are not known.

Keywords: shadowing filter; tracking; positioning; correlated noise; irregular sampling; singularities

1. Introduction

A wide range of filters has been developed for the purposes of tracking moving objects and approximating their trajectories. Currently, tracking is typically achieved using sequential statistical filters, such as Kalman or particle filters, which take only the current and future states into account [1–7]. The classic Kalman filter has been extended to consider converted measurements and nonlinear problems [8]. A review of how to use the particle filter and its capability to handle general models and applications has been introduced in [9,10]. In [11–13], we developed and introduced a new tracking methodology based on shadowing filters. We proved the concept and demonstrated the method's superiority over other tracking methods, as well as its applicability to real-life problems. In this paper, we conduct further investigations to demonstrate and verify the additional advantages of the shadowing filter tracking method, which are important for real applications. Namely, we introduce the two-dimensional solution when the observational error's correlation is taken into account and compare the method's performance when this correlation is considered or ignored. We additionally test the filter's performance when dealing with non-uniform sampling in time and demonstrate the filter's capability to overcome singularities.

As a motivation, consider the frequently-occurring tracking problem that arises when observations are recorded in a non-Cartesian coordinate system. Assume that the bearing and range of an object are observed. These observations are independently determined from some reference point. However, when these measurements are transformed into Cartesian data, they will have correlated errors resulting from this transformation. To make matters worse, we often need to combine multiple measurements. In practice, such problems arise in: (a) active radar locations; (b) satellite interferometry; (c) GPS; and (d) passive sonar locations. The challenges in these applications vary. Applications (a),

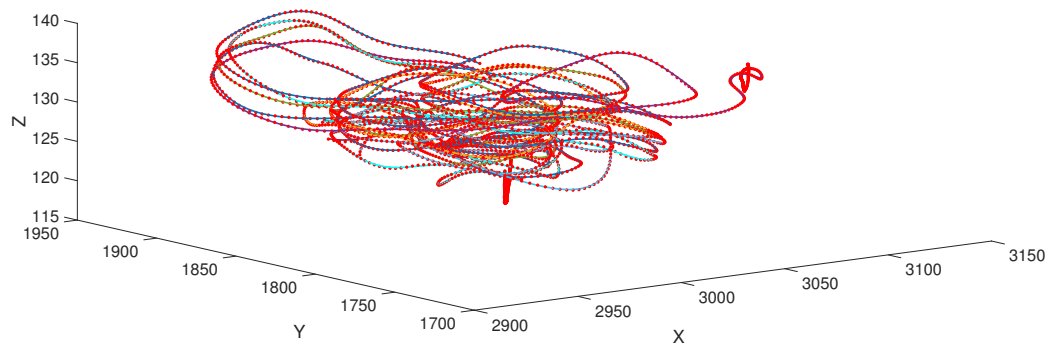
(b) and (d) all use bearing and range data, but in (b), the range is very accurately measured, while the bearing has large errors. In (d), the situation is the opposite, with observations of the bearing being high quality. Application (c), on the other hand, uses only range information, but from several different satellites. All of these cases can be dealt with using the shadowing filter by taking into account the correlation of the observational error. Section 3 of this paper introduces the methodology for approximating the trajectory of an object being tracked with noisy position data and potential correlations in the observational error. The solution in two-dimensions is derived for both uncorrelated and correlated error scenarios. Section 4 utilizes a simple, non-trivial system example to perform a numerical analysis in order to assess whether or not error correlation has to be considered to ensure a good approximation of the object's path and if the extra computational effort required is worth it.

Other common challenges one could face in practice are singularities and non-uniform sampling of data caused by the failure of the recording devices. Section 5 demonstrates the robustness of the shadowing filter tracking methodology to avoid singularity impacts, as well as deal successfully with irregular time resolution. We also apply the tracking technique to a noisy chaotic trajectory, specifically that of the Lorenz model, and analyse the results from the perspective of noise reduction. In the final section, we draw conclusions and make some suggestions for future directions of this work.

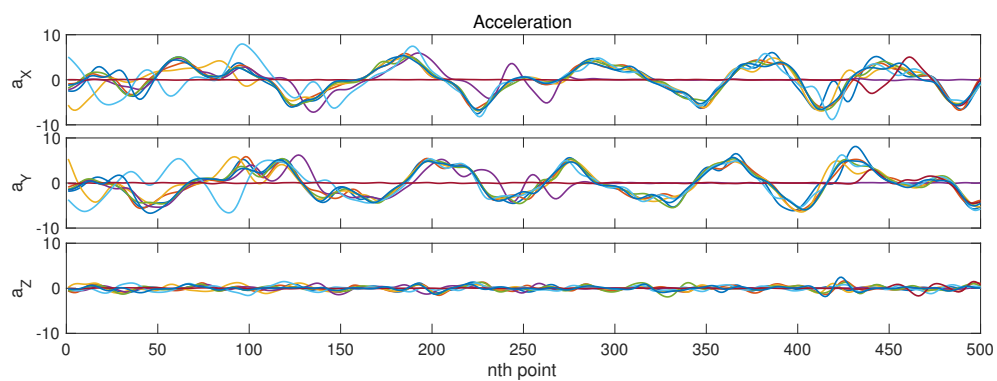
2. Shadowing Filters

The shadowing filter method is unlike sequential filter methods in that it estimates the full trajectory based on all observations simultaneously. In this aspect, our approach is somewhat similar to variational filters [14,15]; however, it does not fall into the trap of local minima [16,17]. This is due to the very simple principle these filters are based on: if the model is good enough, state estimations must be close to the observations and consistent with the model's equations. In practice, this principle imposes a quadratic norm on the filter, which only has one minimum. Specifically regarding the tracking approach, previous investigations showed and verified some merits and advantages of the shadowing filter tracking methodology. In [12,13], we showed the capability of shadowing filter to track single-particle and multi-particle moving objects using GPS data from flying birds; in Figure 1, we show an example of how we successfully used this method to track eight pigeons' trajectories and extracted the corresponding acceleration profiles only from positional GPS data. To track rigid bodies, the method was extended [11] to include rotational motion and moments of inertia. Our method is able to reconstruct the full dynamical phase space from positional information [11–13].

In general, varieties of the shadowing filter have proven to have better performance than Kalman filters [18], particle filters [19], sliding average filters [12] and variational filters [17]. They have been successfully implemented from simple low-dimensional maps and flows [17] up to operational weather models [20]. In [11], we addressed a benchmark with the state-of-art of tracking methods. We conducted direct detailed comparisons with the Kalman filter [21], extended Kalman filter [22] and particle filter [23,24] approaches. Our computational comparisons confirm the superiority of the shadowing filter tracking methodology over these sequential filter tracking methods with respect to accuracy and complexity (computational time). In this manuscript, we will not redo these comprehensive numerical comparisons. However, as the motivation, we confirm the performance compared to existing methods in Figure 2 with systems later investigated in this manuscript. Specifically, correlated observational errors are taken into account in Figure 2a,b and when singularities arise due to recording failure in Figure 2c,d. More detailed explanations and investigations about these scenarios and others are addressed in the following sections.

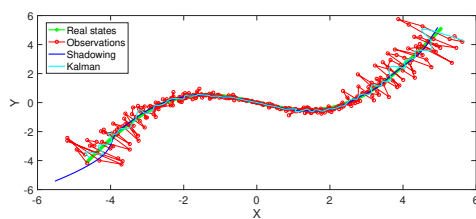


(a) Trajectories

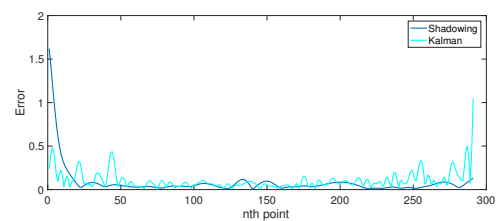


(b) Accelerations

Figure 1. Shadowing filter applied to GPS data from a flock of pigeons. The shadowing filter successfully tracks the eight pigeons. In this demonstration, we refer to our previous works [12,13]. In (a), we track the trajectories of the eight pigeons (the coloured solid lines) from the GPS signals (red dots). In (b), we use the tracking algorithm to estimate the corresponding acceleration profiles for each pigeon (colours refer to different pigeons).



(a) Correlated errors scenario



(b) Error comparison

Figure 2. Cont.

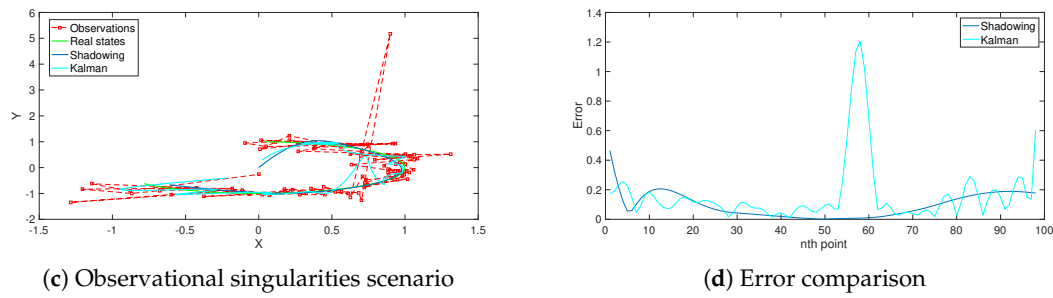


Figure 2. The superiority of the shadowing filter tracking methodology over a sequential filter methodology (Kalman filter). Error correlation has been taken into account in (a) and (b), while the singularity scenario is presented in (c) and (d). The shadowing filter shows obvious superiority, especially where error correlation increases the nonlinearity of the system far from the reference point at the origin in the first scenario and around the singular records in the second scenario. Note that the large error of the shadowing filter estimates at the beginning of the trajectory is expected, as explained in previous investigations [11,12].

3. Summary of the Tracking Methodology

Typically, tracking data will contain only noisy position observations, and so, we are faced with the challenge of having no explicit information about the object's acceleration or velocity. Using the acceleration as a free parameter, we are able to implement the shadowing filter to find a good approximation of the full phase space including acceleration and velocity information based on the position observations. This free parameter can then be tuned in order to optimize the quality of the solution of the approximated states.

We aim to track a point object that is moving in a d -dimensional Euclidean space with Cartesian coordinates, given a sequence of noisy observations. Let $\mathbf{R}_i \in \mathbb{R}^d$ be the true states and $\mathbf{P}_i \in \mathbb{R}^d$ be the noisy observations of the position at time t_i for $i = 0, 1, \dots, n$, where the observational errors have a $d \times d$ covariance matrix $\mathbf{C}_i$ and corresponding information matrix $\mathcal{I}_i = \mathbf{C}_i^{-1}$, which is used to account for the correlations of the observational error.

The object's dynamics are modelled based on its observed position $\mathbf{P}_i \in \mathbb{R}^d$, velocity $\mathbf{v}_i \in \mathbb{R}^d$ and acceleration $\mathbf{a}_i \in \mathbb{R}^d$ for $t_i \leq t \leq t_{i+1}$. Our aim is to approximate $\mathbf{p}_i \in \mathbb{R}^d$ close to the true state $\mathbf{R}_i$. To do this, we minimize the total squared error $\sum_{i=0}^n (\mathbf{P}_i - \mathbf{p}_i)^T \mathcal{I}_i (\mathbf{P}_i - \mathbf{p}_i)$.

We assume that the acceleration is constant over one time interval ($T_i = t_{i+1} - t_i$) and that its magnitude is bounded over the entire trajectory by the relation $(\sum_{i=0}^{n-1} T_i \mathbf{a}_i^T \mathbf{a}_i \leq (t_n - t_0) \xi^2)$. Using Newton's laws and a Galilean transformation, we can solve this optimization problem with the Lagrange multipliers method [25]. The appropriate Lagrange function is:

$$L = \frac{1}{2} \sum_{i=0}^n (\mathbf{P}_i - \mathbf{p}_i)^T \mathcal{I}_i (\mathbf{P}_i - \mathbf{p}_i) \quad (1a)$$

$$+ \sum_{i=0}^{n-1} \lambda_{i+1}^T (\mathbf{p}_{i+1} - \mathbf{p}_i - \mathbf{v}_i T_i - \frac{1}{2} \mathbf{a}_i T_i^2) \quad (1b)$$

$$+ \sum_{i=0}^{n-1} \mu_{i+1}^T (\mathbf{v}_{i+1} - \mathbf{v}_i - \mathbf{a}_i T_i) \quad (1c)$$

$$+ \eta \left(\sum_{i=0}^{n-1} T_i \mathbf{a}_i^T \mathbf{a}_i - (t_n - t_0) \xi^2 \right), \quad (1d)$$

where the Lagrange multipliers $\lambda_i \in \mathbb{R}^d$, $\mu_i \in \mathbb{R}^d$ are d -dimensional column vectors, $\eta \in \mathbb{R}$ and $\xi \in \mathbb{R}$. Equation (1b) represents Newton's first law, Equation (1c) Newton's second law, while Equation (1d) bounds the magnitude of the acceleration. While using the Lagrange multiplier method seems only

natural in the context of a Newtonian model, the optimization could be also implemented using for example gradient descent [16,17] or penalty methods [26].

The solution to the scalar case and rigid-body problems can be found in [11,27]. In this paper, using similar linear algebra methods and appropriately-constructed vector and matrix equations, we extend the solution to higher dimensions, specifically two-dimensional problems, taking into account the correlation of the observational errors. Note that this solution can be generalised for the d -dimensional problem where all vectors and matrices increase in dimension by a factor of d .

3.1. Solution of the Two-Dimensional Problem

3.1.1. Correlated Observational Errors

For higher dimensions, even in two dimensions, the complexity increases as we have to determine whether observational errors are not independent, but correlated. Note that the Weighted Least Squares (WLS) method and its extension, the constrained-WLS, are related to our shadowing filter [28–31]. For the tracking problem, they are able to deal with correlated/uncorrelated noise and irregular sampling in time. However, unlike the shadowing filter, they do not consider explicitly the velocity and acceleration [32,33].

Consider the situation of an object moving in the xy -plane. Given a time resolution of measurements $T_i = t_{i+1} - t_i$ for $i = 0, 1, \dots, n-1$, the true position is $\mathbf{R}_i = (R_{x_i}, R_{y_i})^T \in \mathbb{R}^2$ and the noisy observed position is $\mathbf{P}_i = (X_i, Y_i)^T \in \mathbb{R}^2$. The error in the x -direction has variance $\sigma_{x_i}^2$, and similarly, in the y -direction the error has a variance $\sigma_{y_i}^2$. Therefore, the observational error has the following 2×2 covariance matrix:

$$C_i = \begin{pmatrix} \sigma_{x_i}^2 & \text{cov}_{x_i y_i} \\ \text{cov}_{y_i x_i} & \sigma_{y_i}^2 \end{pmatrix},$$

with the corresponding information matrix:

$$\mathcal{I}_i = C_i^{-1} = \begin{pmatrix} S_{x_i} & S_{x_i y_i} \\ S_{x_i y_i} & S_{y_i} \end{pmatrix},$$

where:

$$\begin{aligned} S_{x_i} &= \frac{\sigma_{y_i}^2}{\sigma_{x_i}^2 \sigma_{y_i}^2 - \text{cov}_{x_i y_i}^2}, \\ S_{y_i} &= \frac{\sigma_{x_i}^2}{\sigma_{x_i}^2 \sigma_{y_i}^2 - \text{cov}_{x_i y_i}^2}, \\ S_{x_i y_i} &= \frac{-\text{cov}_{x_i y_i}}{\sigma_{x_i}^2 \sigma_{y_i}^2 - \text{cov}_{x_i y_i}^2}. \end{aligned}$$

Note that generally, σ_{x_i} and σ_{y_i} can vary with time.

In addition, defining the unknown variables, $\mathbf{p}_i = (x_i, y_i)^T \in \mathbb{R}^2$, $\mathbf{v}_i = (v_{x_i}, v_{y_i})^T \in \mathbb{R}^2$, $\mathbf{a}_i = (a_{x_i}, a_{y_i})^T \in \mathbb{R}^2$, $\boldsymbol{\lambda}_i = (\lambda_{x_i}, \lambda_{y_i})^T \in \mathbb{R}^2$, $\boldsymbol{\mu}_i = (\mu_{x_i}, \mu_{y_i})^T \in \mathbb{R}^2$, $\boldsymbol{\eta} \in \mathbb{R}$ and $\boldsymbol{\xi} \in \mathbb{R}$, then the Lagrangian expressed by Equation (1) can be stated as:

$$L = \frac{1}{2} \sum_{i=0}^n S_{x_i} (X_i - x_i)^2 + 2S_{x_i y_i} (X_i - x_i)(Y_i - y_i) + S_{y_i} (Y_i - y_i)^2 \quad (2a)$$

$$+ \sum_{i=0}^{n-1} [\lambda_{x_{i+1}} (x_{i+1} - x_i - v_{x_i} T_i - \frac{1}{2} a_{x_i} T_i^2) + \lambda_{y_{i+1}} (y_{i+1} - y_i - v_{y_i} T_i - \frac{1}{2} a_{y_i} T_i^2)] \quad (2b)$$

$$+ \sum_{i=0}^{n-1} \mu_{x_{i+1}} (v_{x_{i+1}} - v_{x_i} - a_{x_i} T_i) + \mu_{y_{i+1}} (v_{y_{i+1}} - v_{y_i} - a_{y_i} T_i) \quad (2c)$$

$$+ \eta \left(\sum_{i=0}^{n-1} T_i (a_{x_i}^2 - a_{y_i}^2) - (t_n - t_0) \zeta^2 \right). \quad (2d)$$

Obviously, the optimal solution occurs when all the partial derivatives of L are zero:

$$\frac{\partial L}{\partial x_i} = \frac{\partial L}{\partial y_i} = \frac{\partial L}{\partial v_{x_i}} = \frac{\partial L}{\partial v_{y_i}} = \frac{\partial L}{\partial a_{x_i}} = \frac{\partial L}{\partial a_{y_i}} = \frac{\partial L}{\partial \lambda_{x_i}} = \frac{\partial L}{\partial \lambda_{y_i}} = \frac{\partial L}{\partial \mu_{x_i}} = \frac{\partial L}{\partial \mu_{y_i}} = \frac{\partial L}{\partial \eta} = 0.$$

The partial derivative equations and the complete solution of this situation are detailed in Appendix A. For the purpose of approximating the trajectory, we require only the final equation of the solution given by Equation (A24):

$$(\eta \bar{B} + \bar{A} \mathcal{I}) \mathbf{p} = \bar{A} \mathcal{I} \mathbf{P}$$

where $\bar{A}$ and $\bar{B}$ are numerical matrices used to construct the solution, $\mathcal{I}$ is the information matrix, $\mathbf{P}$ is the observational states vector, $\mathbf{p}$ is the approximated states vector and η is the smoothing parameter of the filter.

3.1.2. Uncorrelated Observational Errors

If the observational errors of the x -component are independent of those in the y -component, then they are uncorrelated and at each time t_i where $i = 0, 1, \dots, n$ the covariance between x_i and y_i equals zero, i.e., $\text{cov}_{x_i y_i} = 0, \forall i = 0, 1, \dots, n$. Therefore, $S_{x_i} = \sigma_{x_i}^{-2}$, $S_{y_i} = \sigma_{y_i}^{-2}$, $S_{x_i y_i} = 0$. Consequently, Equations (A1) and (A2) can be stated as:

$$\frac{\partial L}{\partial x_i} = 0 = \begin{cases} -\sigma_{x_0}^{-2} (X_0 - x_0) - \lambda_{x_1}, & i = 0 \\ -\sigma_{x_i}^{-2} (X_i - x_i) + \lambda_{x_i} - \lambda_{x_{i+1}}, & 0 < i < n \\ -\sigma_{x_n}^{-2} (X_n - x_n) + \lambda_{x_n}, & i = n \end{cases} \quad (3)$$

$$\frac{\partial L}{\partial y_i} = 0 = \begin{cases} -\sigma_{y_0}^{-2} (Y_0 - y_0) - \lambda_{y_1}, & i = 0 \\ -\sigma_{y_i}^{-2} (Y_i - y_i) + \lambda_{y_i} - \lambda_{y_{i+1}}, & 0 < i < n \\ -\sigma_{y_n}^{-2} (Y_n - y_n) + \lambda_{y_n}, & i = n \end{cases} \quad (4)$$

This gives us two independent systems of equations: the x -component system Equations (A3), (A5), (A7), (A9) and (3) and the y -component system Equations (A4), (A6), (A8), (A10) and (4). For each component, the systems are identical to the scalar case, which has been solved previously [11,27], and the optimization problem can be solved independently for x and y .

4. Numerical Investigations: Correlated vs. Uncorrelated Observational Errors

Given a dataset with correlated observational errors, we can either take the correlation into account or ignore it. There are practical reasons why one might choose to ignore the correlation. For instance, the algorithms are more complex with correlation and much more computationally

intensive, as a $d \times 5$ -dimensional system of equations must be solved (see Equations (A1)–(A10)), but if we ignore correlation, the problem is reduced to d independent scalar problems.

We refer herein to these two situations, of taking the Correlation into Account and Ignoring it, as CA and CI, respectively.

In the following investigations, a target's position is tracked in the xy -plane by the position $s = (x, y)^T$ given by the simple, non-trivial observation model:

$$s = \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 10(t/150 - 1/2) \\ 10(t/150 - 1/2) - (5/2) \sin(2\pi(t/150 - 1/2)) \end{pmatrix}, \quad (5)$$

where $0 \leq t \leq 150$, i.e., the length of the time series is $n = 151$.

Note that the choice of this particular path is motivated by the range and bearing problem set up below with the non-linearity in y simplifying the description of the dynamics in polar coordinates.

We introduce correlated observational errors by transforming the position into range and bearing coordinates $q = (r, \theta)^T$ with the inverse of the standard transformation $s = f(q)$:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a + r \cos \theta \\ b + r \sin \theta \end{pmatrix}, \quad (6)$$

where r is the range from a reference point $(a, b)^T$ and θ is the bearing in radians measured anti-clockwise from the x -axis. Thus, the transformed observations are $(r_i, \theta_i)^T$, where $i = 0, 1, \dots, 150$ and the origin $(0, 0)^T$ is the reference point.

We then add white noise to each measurement:

$$\begin{pmatrix} \mathcal{R}_i \\ \Theta_i \end{pmatrix} = \begin{pmatrix} r_i + \sigma_r \chi_i \\ \theta_i + \sigma_\theta \psi_i \end{pmatrix},$$

where $\chi_i \sim N(0, 1)$ is the observational noise of the range component with variance σ_r^2 and $\psi_i \sim N(0, 1)$ is the observational noise of the bearing component with variance σ_θ^2 .

Finally, we obtain the noisy observations, $P_i = (X_i, Y_i)^T$, in Cartesian coordinates, by applying the transformation defined in Equation (6) to $(\mathcal{R}_i, \Theta_i)^T$.

Under the assumption that r is not close to zero and the variances of r and θ are small, the information matrix of $s = (x, y)^T$ can be approximated as [27]:

$$\mathcal{I}_s = \begin{pmatrix} \sigma_r^{-2} \cos^2 \theta + (1/r^2) \sigma_\theta^{-2} \sin^2 \theta & (1/2) \sigma_r^{-2} \sin 2\theta - (1/2r^2) \sigma_\theta^{-2} \sin 2\theta \\ (1/2) \sigma_r^{-2} \sin 2\theta - (1/2r^2) \sigma_\theta^{-2} \sin 2\theta & \sigma_r^{-2} \sin^2 \theta + (1/r^2) \sigma_\theta^{-2} \cos^2 \theta \end{pmatrix}.$$

Given that this information matrix is estimated in advance, we are now able to apply the shadowing technique via Equation (A24) to approximate the trajectory, which we illustrate in Figure 3.

In Figure 3a, the range measurement is four-times more accurate than that of the bearing, as, for this experiment, the standard deviations were chosen to be $\sigma_\theta = 0.60$ and $\sigma_r = 0.15$. Conversely, in Figure 3b, the bearing measurement is four-times more accurate than the range, as $\sigma_r = 0.60$ and $\sigma_\theta = 0.15$. Whilst the filter does work well in either case, note that σ_θ has a significant effect on the observations. This is not the case for σ_r .

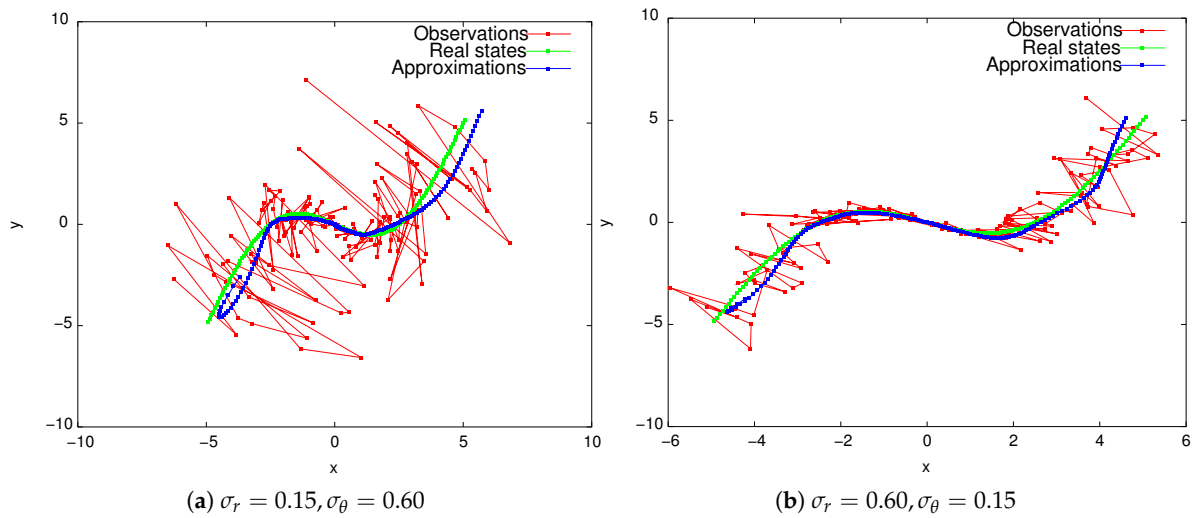


Figure 3. Taking error correlation into account when using range and bearing observations of different accuracies. The smoothing parameter is $\eta = 100$ for both situations.

4.1. Quality Measures

In order to assess the quality of the filter, we must specify the aim of the tracking. Our aim is to find an approximate trajectory from noisy observations of a moving object's positions, which lies optimally close to the true trajectory of the object. In this context, there are two questions that one might hope to answer: (1) Where is the object in a given time interval? (2) What is the current position?

The different nature of these questions is subtle, but profound, with each lending itself to a different class of possible applications. For Question (1), we need to find an approximate trajectory that is close to the true trajectory over the entire interval and therefore shadows the true dynamics. Such a scenario would be required, for example, in order to identify the ship responsible for leaking oil into the ocean based on its trajectory. Question (2) does not need an approximated trajectory that is close to all states, but only close to the last state, for example finding a missing flight. A filter capable of solving both questions will likely need to be optimized differently for the particular case. This could be either “tracking the path that the object travelled” or “knowing the recent position of the object”. Consequently, we require two measures.

The first measure we define details how close the approximated states $\mathbf{p}_i = (x_i, y_i)^T$ are to the true states $\mathbf{R}_i = (R_{x_i}, R_{y_i})^T$ along the whole trajectory for $i = 0, 1, \dots, n$ where $n + 1$ is the length of the time series. For this, we use the root mean squared error defined by:

$$E = \sqrt{\frac{1}{1+n} \sum_{i=0}^n (\|R_{x_i} - x_i\|^2 + \|R_{y_i} - y_i\|^2)}. \quad (7)$$

The second measure details how close the approximated final state, $\mathbf{p}_n = (x_n, y_n)^T$, is to the true final state, $\mathbf{R}_n = (R_{x_n}, R_{y_n})^T$ and can be quantified by the end-point error:

$$e = \sqrt{\|R_{x_n} - x_n\|^2 + \|R_{y_n} - y_n\|^2}. \quad (8)$$

For more accurate results, we use the ensemble averages of these quantities as measures of the filter's performance. We consider N time series, giving N true trajectories, N sequences of observations and therefore N sequences of approximates:

$$\langle E \rangle = \frac{1}{N} \sum_{j=1}^N E_j, \quad (9)$$

$$\langle e \rangle = \frac{1}{N} \sum_{j=1}^N e_j. \quad (10)$$

In the experiments that follow, we assess the quality of the filter for both tracking objectives and optimize parameters such that the respective average errors are minimized.

4.2. Optimization of the Filter Performance

The quality of the solution, i.e., the approximated states, can be optimized by tuning the free parameter of our filter. Generally, the dataset will contain noisy position observations, and so, acceleration can be used as the free parameter. Thus, the smoothing parameter η of the acceleration Lagrangian term in Equation (2d) approximates a suitable value of the object's inertia.

The optimal value is dependent on the time resolution of the time series, the system's dynamics, the noise intensity, etc. Therefore, the following investigation should be understood as a proof of concept only, whereby an optimal value is determined for some specific application. Furthermore, as we only want to show that η does in fact have some optimum value, we can use a rather simple numerical scheme.

The algorithm implemented starts with a broad parameter sweep, $\eta = 0.1, 1, 10, 100, 1000$, and calculates the average errors for each tracking objective ($\langle E \rangle$ and $\langle e \rangle$) from $N = 100$ time series. It then identifies the sub-interval in which the minimum error occurs and divides this η -interval into five new subintervals. Repeating this procedure six times, with a new time series for each narrowing of the parameter range, we get an approximation of the optimal η . This approximation is sufficient to show that an optimal value exists.

Tables A1 and A2 show the results when optimizing η subject to minimizing $\langle E \rangle$ for the CA and CI cases, respectively. The minimum in each interval is indicated in bold. We note that the optimal values for the two types of correlation consideration lie close together, and there does not seem to be a large difference between the resulting errors.

Tables A3 and A4 show the optimization data when our aim is to minimize $\langle e \rangle$ for the CA and CI cases, respectively. Again, minimum values in each interval are indicated in bold. Here, we note a huge difference between the approximated optimal values for the two types of correlation considered. When correlation was ignored, we were required to expand the starting interval as the best η value in Experiment 1 (the zeroth iteration) was the interval's endpoint $\eta = 1000$. Consequently, the almost optimal value of this method $\eta = 1600$ was much higher than the value of $\eta = 91$, found when correlation was considered. On the other hand, the average end-point error was 1.4-times larger when correlation was accounted for.

The results of this investigation are summarized in Figure 4. Blue lines show the results when correlation is accounted for, and red lines show the results when correlation is ignored. The general shape of the graphs—large values of the error for small and large η values—support our initial assumption that an optimal value of η exists. Since our primitive iterative algorithm can only be used to find an approximation of the optimal η , we cannot say anything about whether the found best value is a local or a global property. However, as was pointed out in the beginning, the purpose of this investigation was only to show that the η value can be optimized for a specific application.

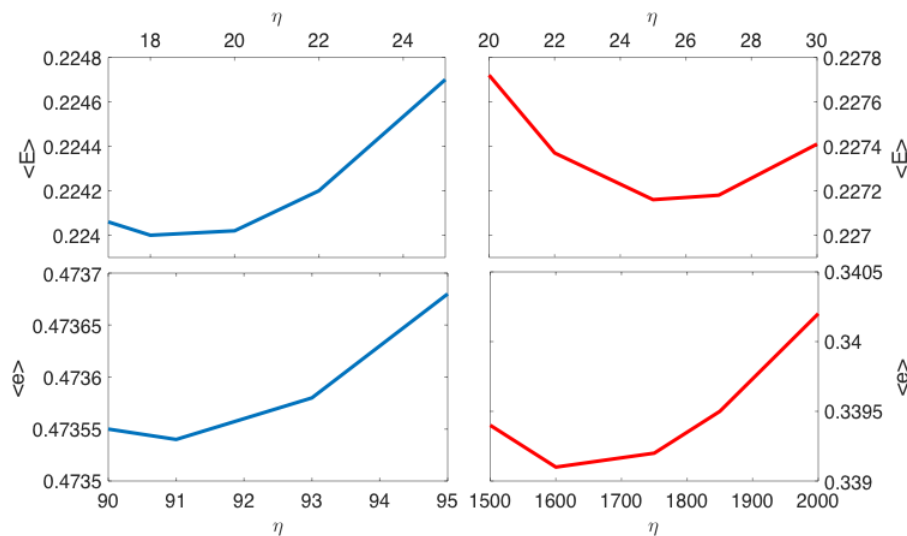
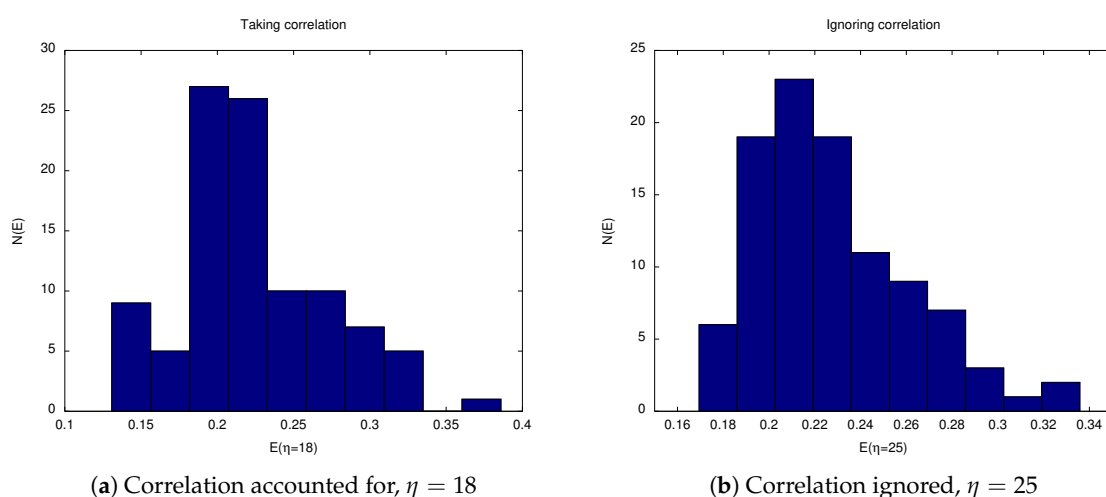


Figure 4. Optimisation of the smoothing parameter η : average root mean squared error $\langle E \rangle$ and average end-point error $\langle e \rangle$ as functions of the smoothing parameter η . We show both scenarios when the error's correlation is taken into account (blue) or completely ignored (red). The minimum illustrates the existence of the optimal value of η that minimises the error and shows that incorporating the error's correlation does not significantly improve the filter's performance.

4.3. Comparison of the CA and CI Cases

Recall that CA refers to the situation where correlation is taken into account, and CI ignores it. Surprisingly, the CI algorithm seems to lead to similar or even better estimates. To get a clearer picture, we need to understand the error distributions, as well.

Figures 5 and 6 show histograms of the two error measures, $\langle E \rangle$ and $\langle e \rangle$, resulting from each correlation algorithm. We used data from 100 time series and chose the approximated optimal η values found previously.



(a) Correlation accounted for, $\eta = 18$

(b) Correlation ignored, $\eta = 25$

Figure 5. Histograms of the root mean squared error E from 100 observation sequences at the optimal value of η in both scenarios when the error's correlation is taken into account or ignored.

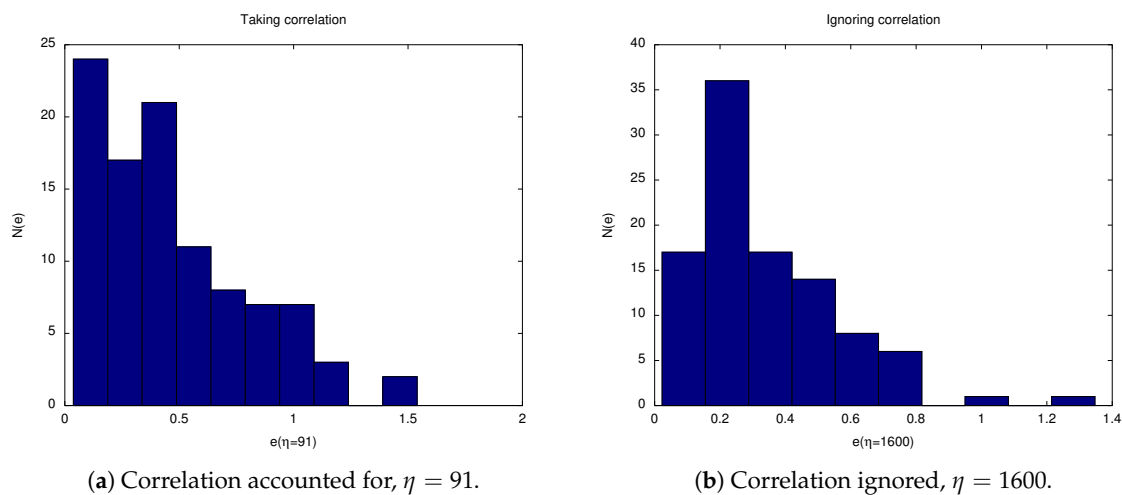


Figure 6. Histograms of the end-point error e from 100 observation sequences at the optimal value of η in both scenarios when error's correlation is taken into account or ignored.

We are not surprised that there was not much difference between the CA and CI algorithms when we were interested in minimizing $\langle E \rangle$, as seen in Figure 5. Both average errors $\langle E \rangle$ were about 0.21, and the distributions were peaked around this value. If there was any difference, it was that the percentage of errors contained in the interval $[0.18, 24]$ was about 10% higher for the CA algorithm. It is not immediately clear whether this is statistically significant, but given that both methods showed similar result, we do not think that investigating this statistical significance is important.

Recall that for the case of minimizing $\langle e \rangle$, unlike for $\langle E \rangle$, our η optimization gave us quite different optimal values depending on which algorithm was used. This situation is depicted in Figure 6. Surprisingly, the resulting distributions have much in common:

- For the CA algorithm: the $\langle e \rangle$ -distribution decayed monotonously, and all errors were contained within the interval $[0.1, 1.3]$; and about 60% of the errors were between $[0.1, 0.5]$.
- For the CI algorithm: the errors were contained within the smaller interval $[0.1, 0.8]$, and the distribution was less broad; and in the subinterval from $[0.1, 0.5]$, we found 70% of the errors.
- In addition, there was a peak around the average value in both cases with $\langle e \rangle \approx 0.2$.

We conclude that the shadowing filter's tracking methodology was robust enough to consider or ignore the correlations of the observational errors. That is, there was not a marked improvement in the shadowing filter's performance if we took correlations into account. This is further illustrated in Figure 7, which shows estimates for both tasks ((a) minimizing average error and (b) minimizing last point error).

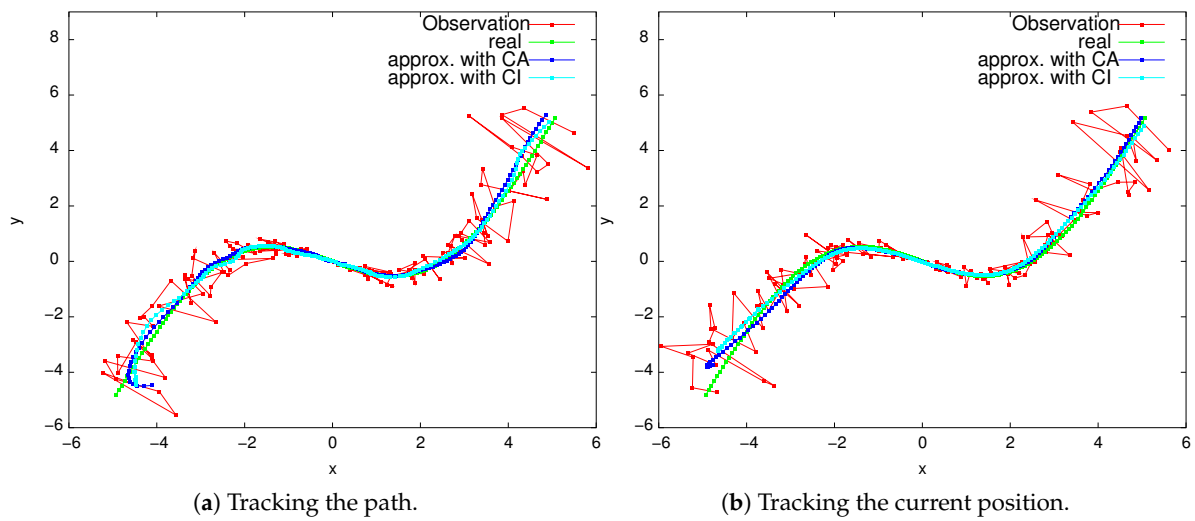


Figure 7. The filter performance in estimating (a) the path and (b) the current position, when error correlation is taken into account. Note that the true dynamics is given by Equation (5).

5. Applications

In this section, we demonstrate some applications with common challenges one frequently faces in practice. Specifically, we emphasise three challenges relevant to tracking: irregular/non-uniform time resolution, records with large singularities, tracking chaotic trajectories and real-world multiple object application.

5.1. Non-Uniform Sampling

Here, we highlight the fact that our shadowing filter is flexible enough to be used even for non-uniformly sampled data. Often, data points are not recorded with the same sampling rate, and so, $t_{h+2} - t_{h+1} = t_{h+1} - t_h = \Delta t$ does not hold for all h in the dataset. Such a situation is quite common for example in GPS tracking, as the GPS device often fails to log one or more data points [34,35]. Our filter can be used with arbitrary values of Δt_i between any two data points; however, we focus here on the sub-problem that some data points are not recorded with each Δt_i a multiple of some sampling time such that $\Delta t_i = m \times \Delta t$.

It has been shown that our tracking methodology is sufficiently robust to ignore the error correlation without effecting performance. Consequently, any d -dimensional problem can be treated as d independent scalar problems. Hence, for our numerical investigation, we will use data generated from the following scalar function that gives true states Y and noisy observations P :

$$\begin{aligned} Y_t &= 25 + 10 \sin \frac{t}{15} + \sqrt{\alpha} \chi_t, \\ P_t &= Y_t + \sqrt{\beta} \epsilon_t, \end{aligned} \quad (11)$$

where $0 \leq t \leq n$, the observation's noise $\epsilon_t \sim N(0,1)$ with ϵ_t a white noise process, χ_t is an independent cumulative white noise process, $\chi_t = \sum_{i=1}^t \hat{\xi}_i$ where $\hat{\xi}_i \sim N(0,1)$, α is the variance of the cumulative white noise and β is the variance of the observational noise.

We generated data with initial length of $n = 500$ using $\beta = 9$ and $\alpha = 1$. From the dataset, we randomly deleted data points using a uniform distribution (cf. Figure 8a) so that the final dataset had $n' \approx 125$ data points. Again, we used our usual measures $\langle E \rangle$ and $\langle e \rangle$ estimated from $N = 100$ datasets for $\eta = 1.3$ or 19 (depending on the tracking objective). To incorporate the non-constant sampling time, we included the sampling times into the matrix Equation (A12). Our numerical investigation shows that for increasing levels of deletion, the error increased relative

to the uniform sampling. Specifically, we found that $\langle E \rangle \approx 3.3$ ($\eta = 1.3$) and $\langle e \rangle \approx 1.5$ ($\eta = 19$). Overall, our filter was still able to find a good enough approximation, as can be seen in Figure 8b.

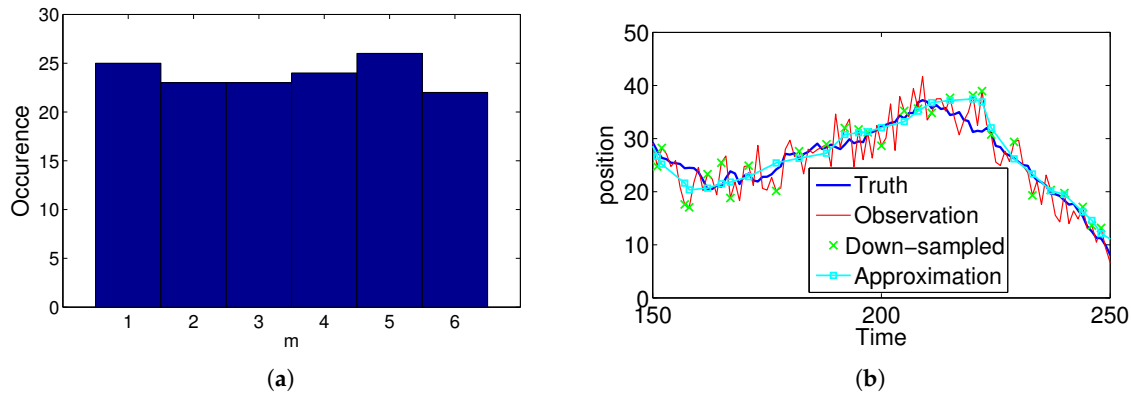


Figure 8. Performance of the filter when using non-uniform sampled data. (a) shows the histogram of sample times in the sequence, and (b) directly compares the observed and estimated trajectory, as well as the observed and non-uniformly down-sampled time series.

5.2. Multiple Bearing Observations: Singularities

In this section, we will study another situation where non-Cartesian observations are used. Consider the case when the target is tracked in the xy -plane, by the position $s = (x, y)$, using bearing observations $q = (\theta, \theta')$ from two distinct reference points (a, b) and (a', b') . The transformation f between the different coordinates is defined by:

$$(x, y) = (a, b) + m(\cos \theta, \sin \theta), \quad (12)$$

$$= (a', b') + m'(\cos \theta', \sin \theta'), \quad (13)$$

where under most circumstances, there exist unique $m, m' \in \mathbb{R}$. Hence, if we have bearing observations, then we need to find the raw position estimates, and that requires calculating m and m' . These can be estimated by solving the linear equations:

$$a + m \cos \theta = a' + m' \cos \theta',$$

$$b + m \sin \theta = b' + m' \sin \theta',$$

which are equivalent to:

$$m \cos \theta - m' \cos \theta' = a' - a,$$

$$m \sin \theta - m' \sin \theta' = b' - b,$$

and these can be expressed in matrix form as follows:

$$\begin{pmatrix} \cos \theta & -\cos \theta' \\ \sin \theta & -\sin \theta' \end{pmatrix} \begin{pmatrix} m \\ m' \end{pmatrix} = \begin{pmatrix} a' - a \\ b' - b \end{pmatrix}. \quad (14)$$

Solving these linear equations enables us to find the raw position estimates from bearing observations using the transformation f [27].

In this situation, it is difficult to compute the Jacobian matrix directly from the transformation f . However, note that the inverse of the transformation f is given by:

$$\theta = \arctan \frac{y-b}{x-a}, \quad (15)$$

$$\theta' = \arctan \frac{y-b'}{x-a'}. \quad (16)$$

Consequently, the matrix K , the Jacobian matrix of f^{-1} , is easily computed as follows:

$$K = \begin{pmatrix} \frac{-(y-b)}{r^2} & \frac{(x-a)}{r^2} \\ \frac{-(y-b')}{r'^2} & \frac{(x-a')}{r'^2} \end{pmatrix},$$

where $r^2 = (x-a)^2 + (y-b)^2$ and $r'^2 = (x-a')^2 + (y-b')^2$, which requires simple computations once an estimate of (x, y) is obtained. Therefore, under the assumption that both r and r' are not close to zero, the information matrix can be approximated using the matrix K as follows:

$$\begin{aligned} \mathcal{I}_s &= K^T \mathcal{I}_q K \\ &= \begin{pmatrix} \frac{-(y-b)}{r^2} & \frac{-(y-b')}{r'^2} \\ \frac{(x-a)}{r^2} & \frac{(x-a')}{r'^2} \end{pmatrix} \begin{pmatrix} \sigma_\theta^{-2} & 0 \\ 0 & \sigma_{\theta'}^{-2} \end{pmatrix} \begin{pmatrix} \frac{-(y-b)}{r^2} & \frac{(x-a)}{r^2} \\ \frac{-(y-b')}{r'^2} & \frac{(x-a')}{r'^2} \end{pmatrix}, \end{aligned}$$

where σ_θ^{-2} is the variance of the observational error in the first bearing component and $\sigma_{\theta'}^{-2}$ is the variance in the second bearing component. It follows that the information matrix is given by:

$$\mathcal{I}_s = \begin{pmatrix} \frac{\sigma_\theta^{-2}(y-b)^2}{r^4} + \frac{\sigma_{\theta'}^{-2}(y-b')^2}{r'^4} & \frac{-\sigma_\theta^{-2}(x-a)(y-b)}{r^4} + \frac{-\sigma_{\theta'}^{-2}(x-a')(y-b')}{r'^4} \\ \frac{-\sigma_\theta^{-2}(x-a)(y-b)}{r^4} + \frac{-\sigma_{\theta'}^{-2}(x-a')(y-b')}{r'^4} & \frac{\sigma_\theta^{-2}(x-a)^2}{r^4} + \frac{\sigma_{\theta'}^{-2}(x-a')^2}{r'^4} \end{pmatrix}.$$

Given this information matrix, we can apply the shadowing filter to find an approximated trajectory.

A problem arises in situations when the sensors and the target are collinear; i.e., $\theta \approx \pm\theta'$. In such situations, the linear Equations (14) are singular or badly conditioned. This can lead to raw position estimates far from their true positions. To solve this problem, we have three options: we can drop the bad observation, replace it with a forecasted position or take into account that the information in this observation is less reliable than in other observations. Since the shadowing filter is estimating a trajectory from a sequence of observations, generally there is no harm in either of these solutions. Although, dropping bad observations leads to a non-uniform time-gap between observations where the shadowing filter still works successfully, as verified in the above investigations. We have chosen here to give our observations different weights, according to their reliability. To do this, we scaled the information matrix by using the `rcond` command in MATLAB. This command calculates the one-norm estimate of the reciprocal condition number as returned by LAPACK. `rcond` was used on the matrix giving information on the angles:

$$\begin{pmatrix} \cos \theta & -\cos \theta' \\ \sin \theta & -\sin \theta' \end{pmatrix}.$$

The output of `rcond` varies between zero and one. For angles that lead to an ill-conditioned Equation (14), we get small weights, but good observations will get maximum weighting.

In the example shown in Figure 9 and [27], we tracked an object using bearing observations provided by two moving sensors. The target moved on a circular path $(\sin(t/25), \cos(t/25))$ for $0 \leq t \leq 100$. It moved in a clockwise direction, starting from the 12 o'clock position. The first sensor moved between $(-3, 3)$ and $(3, 1)$, and the other between $(-3, -2)$ and $(3, 1)$. Both sensors had a constant speed. To generate the bearing observations, we transformed the true states of the object to the bearing coordinates using Equations (15) and (16). We added white noise to both components

with variances $\alpha_\theta = \alpha_{\theta'} = 0.05$ to obtain the desired multiple bearing noisy observations. After that, we transformed these noisy observations to the Cartesian coordinates using the transformation f after computing m and m' from Equation (14). We applied the shadowing filter at $\eta = 1$ on these observations using the scaled information matrices to deal with the bad observations. In Figure 9, we can also see the effect of the different weighting. Note that when the target was between the four and five o'clock position, it was directly between the sensors $\theta = \pm\theta'$. This led to a very bad observation, as seen by the large error. Similarly, at the end of the trajectory, the object was almost directly behind both sensors, which also led to a poorly-conditioned observations. However, using the one-norm estimates as described above enabled us to determine good approximations.

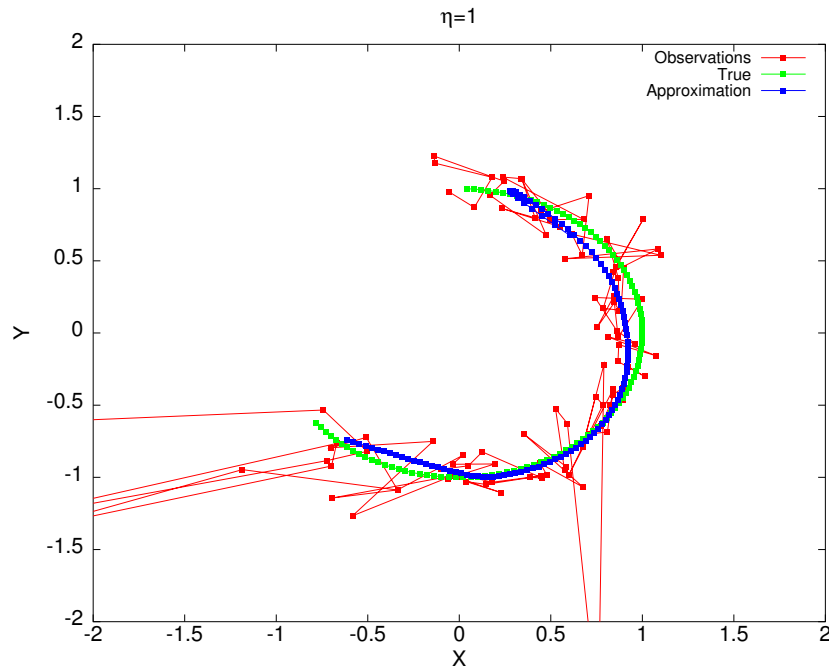


Figure 9. Performance of the shadowing filter in tracking an object using multiple bearing observations with $\eta = 1$, $\alpha_\theta = \alpha_{\theta'} = 0.05$. Note how the filter is able to overcome the singularities in the observations and estimates the true dynamics quite well.

5.3. Chaotic Trajectory: Lorenz Model

In the previous sections and examples, we investigated the performance of the tracking filter for deterministic trajectories in one- and two-dimensional spaces. The natural step from here is to consider the realistic case of a tracking such as the chaotic motions seen in nature in insects, fish or birds.

We now apply the tracking technique to the Lorenz model [36]:

$$\frac{dx}{dt} = \Sigma(y - x) \quad (17a)$$

$$\frac{dy}{dt} = x(\rho - z) - y \quad (17b)$$

$$\frac{dz}{dt} = xy - Bz \quad (17c)$$

Using the standard Lorenz model and typical parameters $\rho = 28$, $\Sigma = 10$ and $B = 8/3$, we initialized the system with $(x_0, y_0, z_0) = (0, 1, 1.05)$ and generated a trajectory containing 2501 data points for $t = [0, 25]$. The data evolved as per Equation (17) and were measured with some uncertainty following the incorporation of noise with variance of $\beta = 1$ to each component as per Equation (11).

The dynamics were tracked using the shadowing filter with $\eta = 0.1$, which was the optimal value for the chosen noise variance as found in Section 4, and a sampling rate of 0.1. We considered uncorrelated errors and treated the system as per the scalar case outlined in Section 3.

Figure 10a–c shows the successful tracking of a target in each of the variables, while Figure 11 shows these details in a projection of the three-dimensional state space. We note that as the shadowing filter did not contain any information about the vector field, the noisy observations were projected down on to the attracting manifold of the Lorenz model. Therefore, our filter acted similar to the other shadowing filters, which had information regarding the vector field encoded in them [17]. Although there were some estimation error, one can see at the outer loops in Figure 11, we note that our filter estimated the true time series most of the time.

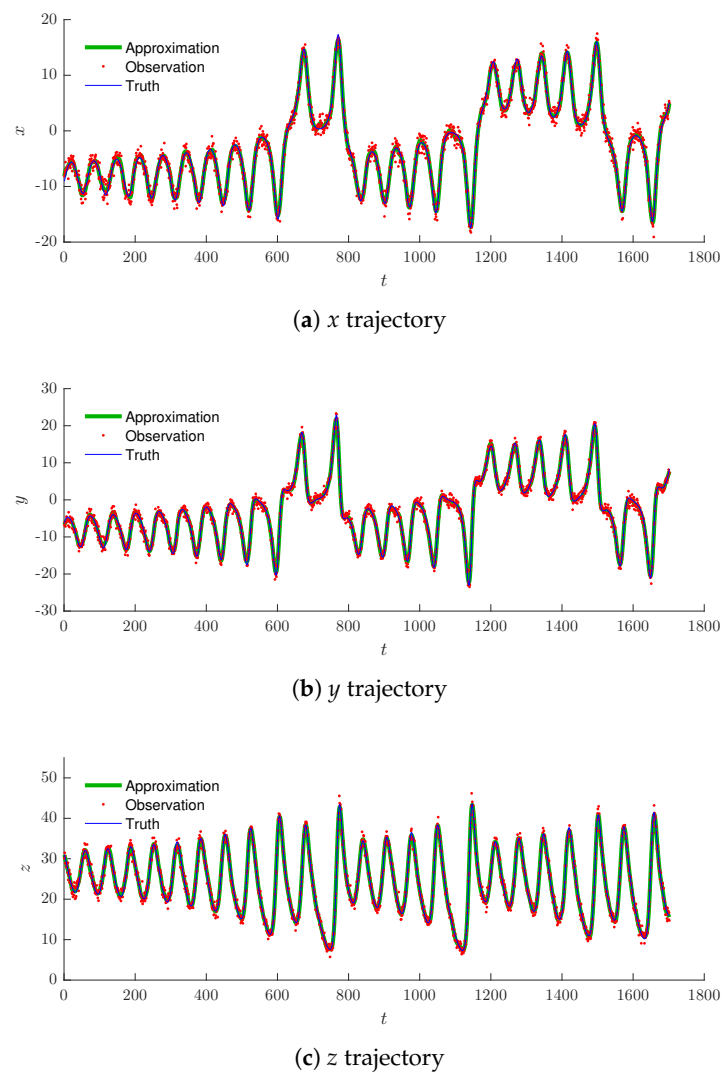


Figure 10. Projections onto each direction for the chaotic Lorenz model.

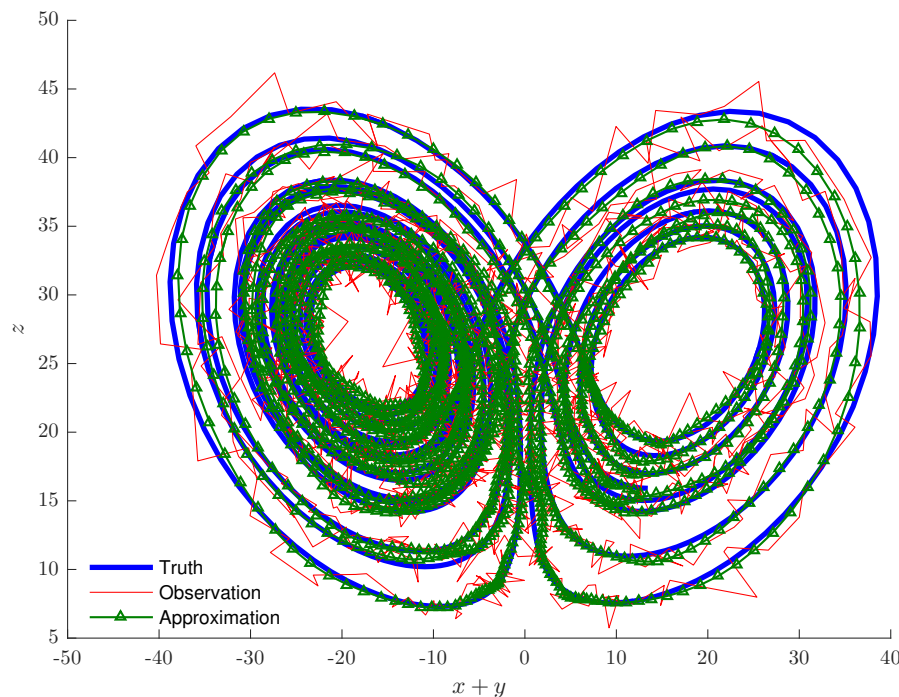


Figure 11. Chaotic Lorenz attractor (blue line), noisy observations (red line) and the tracking technique results (green triangles).

Up until this point, we have been concerned with forecasting. However, this is closely related to noise reduction, and the results of the previous section can be viewed in such a context [37,38].

We assumed that the clean time series of each variable, $\tilde{x}$, $\tilde{y}$ and $\tilde{z}$, is in the presence of measurement noise and aimed to separate these from the observed noisy data in order to restore the structure of the system and improve the quality of predictions [38], i.e., $x_n = \tilde{x}_t + \sqrt{\beta}\epsilon_t$, where ϵ_t is defined as in Equation (11).

For measurement noise applied to Equation (17), the approximated trajectory (green triangles) of Figures 10 and 11 is the reduced noise trajectory. Figure 11 illustrates how the noise (red line) has smeared the chaotic attractor and obscured its structure and how the chaotic dynamics of the clean (blue line) trajectory were successfully captured.

5.4. Real-World Multiple Object Application

As a final application, we present how the shadowing filter can be used to track multiple objects. We used two-dimensional data produced from images of a real flock of ducks moving on a water surface. The dataset was provided by the authors of [39]. We focussed on one of the data files where a turning event occurred. We consider 25 individuals observed at 30 time points. We tracked the trajectories using the positional observations and extracted the corresponding acceleration for each bird. The difference between this application and the tracking of pigeons illustrated in Figure 1 was the size of the flock. Here, we have a larger number of individuals, but a relatively short trajectory. The results are shown in Figure 12, where the circles are the observations and the estimated paths are given as solid lines. The filter accurately tracked the position of each individual. The acceleration of each bird is shown in the lower panels of the figure. Note that this additional information about each individual's state can be used to forecast its future dynamics and could be used to infer that the forces dominate such a collective behaviour [13].

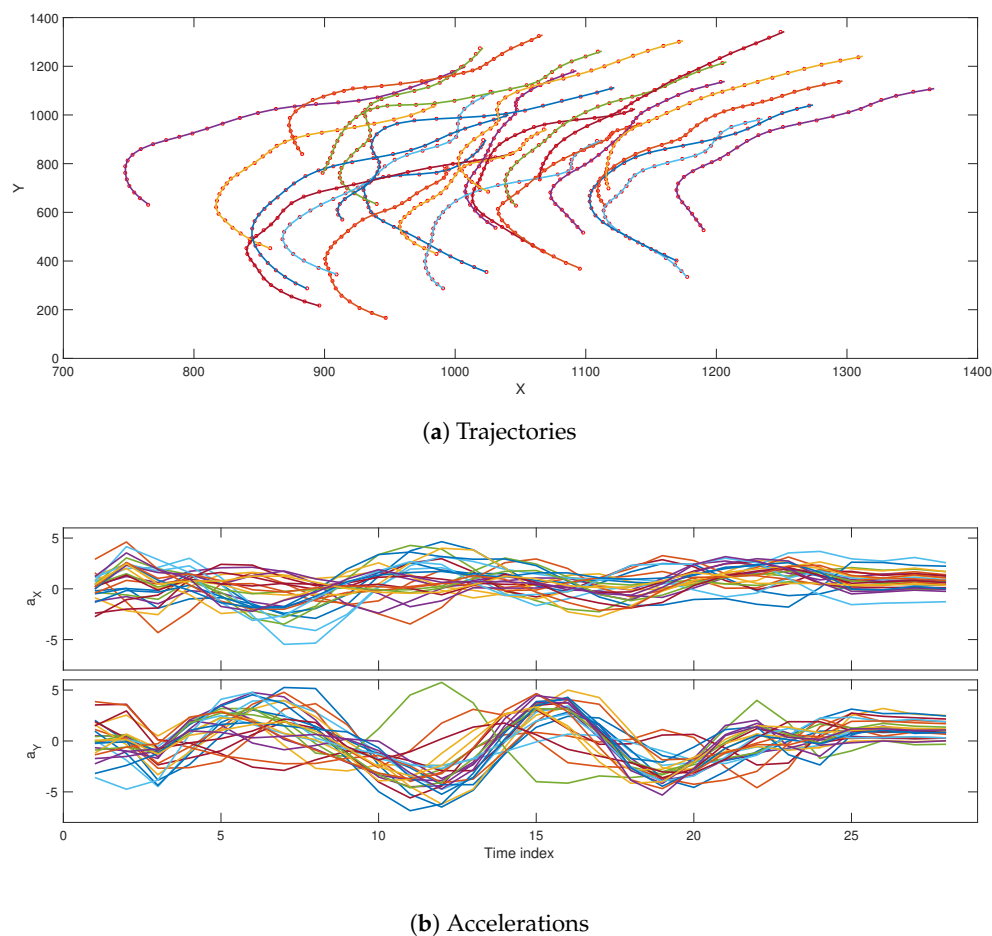


Figure 12. Shadowing filter applied to positional data from a group of ducks moving on the water surface. The shadowing filter successfully tracked the 25 individuals. In (a), we tracked the trajectories of the 25 ducks (the coloured solid lines) from the positional observations (red circles). In (b), we used the tracking algorithm to estimate the corresponding acceleration profiles for each duck (colours refer to different individuals).

6. Conclusions

In this paper and our previous ones [11–13], a new class of shadowing filters, applicable to observational data of objects moving in more than one dimension, was introduced and investigated. Our investigations demonstrate that this approach is highly versatile. First of all, it may be easily adapted to different settings. For instance, it can be implemented to track a single moving particle, a system of multiple objects [13] or be extended to consider rigid bodies instead [11]. Secondly, it is computationally inexpensive in comparison to alternative methods [11]. Finally, it outperformed the established and widely-used Kalman and particle filters in situations where nonlinearities were present [11]. Our tracking technique is applicable to real data and enables us to minimise measurement errors and overcome device failures, as well as allows for a reconstruction of the full dynamical state space from limited position-only data [12,13].

In this paper, specifically, optimization of the filter quality was achieved for two tracking objectives, namely “tracking the path that the object travelled” and “knowing the recent position of the object”, by tuning the smoothing parameter such that the (task-specific) error was minimized. Numerical investigations indicated that our tracking method was sufficiently robust that the benefit of taking error correlation into account was negligible, allowing for many systems to be treated as per the scalar case. Furthermore, we demonstrated that this tracking method can be applied to irregularly-sampled real-world data without additional effort, a situation often encountered when measurement devices

fail to log some measurement points. We highlight the ability of the shadowing filter tracking method to avoid singularities in data, another challenge encountered frequently in practice. The shadowing filter was successfully applied to noisy data from the Lorenz model, and we briefly commented on the relation of this approach to noise reduction algorithms. Finally, a real-world application of multiple object tracking was provided, as we tracked a flock of 25 ducks moving on the water surface.

Lightweight and inexpensive tracking devices are now available and have opened up a new area of tracking possibilities [34,35,40–44], resulting in new observational data, which enable the study of movement, of animals for example, with so far unreached precision. Given these advantages and the resulting big data, it is clear that the new generation of filters needs to be fast and efficient. The outlined benefits of our filter together with its clear optimization possibility make it, in our opinion, a very strong contender for such applications, and we are looking forward to seeing the progress in this area based on our filter.

Author Contributions: A.Z. initiated the ideas, developed the algorithm, performed the computation and wrote the manuscript. T.S. initiated the ideas, discussed the results and revised and improved the manuscript. S.D.A. helped prepare the numerical experiments and revised and improved the manuscript.

Funding: This research received no external funding.

Acknowledgments: We, the authors, would like to thank Kevin Judd and Michael Small for their valuable discussions and suggestions. We also would like to thanks M.Nagy et al. for providing the pigeons data and R. Lukeman et al. for providing the ducks data. A.Z. acknowledges the support of the Australian Research Council through Discovery Grant DP 180100718.

Conflicts of Interest: The authors declare no conflict of interest.

Code Availability

Contact the author for the MATLAB code. A simple code for one-dimensional tracking is provided at: MathWorks-File Exchange-One-Dimensional Tracking Methodology Using Shadowing Filter-Ayham Zaitouny.

Abbreviations

The following abbreviations are used in this manuscript:

- CA Taking observational errors' Correlation into Account
- CI Ignoring observational errors' Correlation

Appendix A. Detailed Solution

$$\frac{\partial L}{\partial x_i} = 0 = \begin{cases} -S_{x_0}(X_0 - x_0) - S_{x_0 y_0}(Y_0 - y_0) - \lambda_{x_1}, & i = 0 \\ -S_{x_i}(X_i - x_i) - S_{x_i y_i}(Y_i - y_i) + \lambda_{x_i} - \lambda_{x_{i+1}}, & 0 < i < n \\ -S_{x_n}(X_n - x_n) - S_{x_n y_n}(Y_n - y_n) + \lambda_{x_n}, & i = n \end{cases} \quad (A1)$$

$$\frac{\partial L}{\partial y_i} = 0 = \begin{cases} -S_{y_0}(Y_0 - y_0) - S_{x_0 y_0}(X_0 - x_0) - \lambda_{y_1}, & i = 0 \\ -S_{y_i}(Y_i - y_i) - S_{x_i y_i}(X_i - x_i) + \lambda_{y_i} - \lambda_{y_{i+1}}, & 0 < i < n \\ -S_{y_n}(Y_n - y_n) - S_{x_n y_n}(X_n - x_n) + \lambda_{y_n}, & i = n \end{cases} \quad (A2)$$

$$\frac{\partial L}{\partial v_{x_i}} = 0 = \begin{cases} -\lambda_{x_1} T_0 - \mu_{x_1}, & i = 0 \\ -\lambda_{x_{i+1}} T_i + \mu_{x_i} - \mu_{x_{i+1}}, & 0 < i < n \\ \mu_{x_n}, & i = n \end{cases} \quad (A3)$$

$$\frac{\partial L}{\partial v_{y_i}} = 0 = \begin{cases} -\lambda_{y_1} T_0 - \mu_{y_1}, & i = 0 \\ -\lambda_{y_{i+1}} T_i + \mu_{y_i} - \mu_{y_{i+1}}, & 0 < i < n \\ \mu_{y_n}, & i = n \end{cases} \quad (A4)$$

$$\frac{\partial L}{\partial a_{x_i}} = 0 = -\frac{1}{2} \lambda_{x_{i+1}} T_i^2 - \mu_{x_{i+1}} T_i + 2\eta T_i a_{x_i} \quad (A5)$$

$$\frac{\partial L}{\partial a_{y_i}} = 0 = -\frac{1}{2} \lambda_{y_{i+1}} T_i^2 - \mu_{y_{i+1}} T_i + 2\eta T_i a_{y_i} \quad (A6)$$

$$\frac{\partial L}{\partial \lambda_{x_i}} = 0 = x_{i+1} - x_i - v_{x_i} T_i - \frac{1}{2} a_{x_i} T_i^2 \quad (A7)$$

$$\frac{\partial L}{\partial \lambda_{y_i}} = 0 = y_{i+1} - y_i - v_{y_i} T_i - \frac{1}{2} a_{y_i} T_i^2 \quad (A8)$$

$$\frac{\partial L}{\partial \mu_{x_i}} = 0 = v_{x_{i+1}} - v_{x_i} - a_{x_i} T_i \quad (A9)$$

$$\frac{\partial L}{\partial \mu_{y_i}} = 0 = v_{y_{i+1}} - v_{y_i} - a_{y_i} T_i \quad (A10)$$

$$\frac{\partial L}{\partial \eta} = 0 = \sum_{i=0}^{n-1} T_i (a_{x_i}^2 - a_{y_i}^2) - (t_n - t_0) \xi^2 \quad (A11)$$

We will show how to solve the problem in two-dimensions when the correlation is taken into account. We will represent the equations using vector and matrix forms. We will define some vectors and matrices; please note that in our definitions, $\mathbf{0}$ refers to a 2×2 matrix of zeros, as follows:

$$\mathbf{P}_{(2n+2) \times (2n+2)} = \begin{pmatrix} X_0 \\ Y_0 \\ \vdots \\ X_n \\ Y_n \end{pmatrix}, \mathbf{p}_{(2n+2) \times (2n+2)} = \begin{pmatrix} x_0 \\ y_0 \\ \vdots \\ x_n \\ y_n \end{pmatrix}, \boldsymbol{\lambda}_{(2n) \times 1} = \begin{pmatrix} \lambda_{x_1} \\ \lambda_{y_1} \\ \vdots \\ \lambda_{x_n} \\ \lambda_{y_n} \end{pmatrix}, \boldsymbol{\mu}_{(2n) \times 1} = \begin{pmatrix} \mu_{x_1} \\ \mu_{y_1} \\ \vdots \\ \mu_{x_n} \\ \mu_{y_n} \end{pmatrix},$$

$$\mathbf{a}_{(2n) \times 1} = \begin{pmatrix} a_{x_0} \\ a_{y_0} \\ \vdots \\ a_{x_{n-1}} \\ a_{y_{n-1}} \end{pmatrix}.$$

Obviously, P denotes the observations, p the estimates, λ and μ our Lagrangian parameters, while a the piecewise constant accelerations. Moreover, to take correlations into account, we define the information matrix as:

$$\mathcal{I}_{(2n+2) \times (2n+2)} = \begin{pmatrix} \begin{bmatrix} S_{x_0} & S_{x_0 y_0} \\ S_{x_0 y_0} & S_{y_0} \end{bmatrix} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \begin{bmatrix} S_{x_n} & S_{x_n y_n} \\ S_{x_n y_n} & S_{y_n} \end{bmatrix} \end{pmatrix}.$$

Similar to the scalar case, we need a time matrix T . However, in the two-dimensional problem, each time resolution is repeated twice along the main diagonal of T , since we have two components:

$$T_{(2n) \times (2n)} = \begin{pmatrix} \begin{bmatrix} T_0 & 0 \\ 0 & T_0 \end{bmatrix} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \begin{bmatrix} T_{n-1} & 0 \\ 0 & T_{n-1} \end{bmatrix} \end{pmatrix}. \quad (\text{A12})$$

To simplify the problem as before to one compact equation, we need two additional matrices, a $2n \times 2n$ matrix D and a $(2n+2) \times 2n$ matrix E defined as:

$$D_{ij} = \begin{cases} -1, & i = j, \\ 1, & i = j + 2, \\ 0, & \text{otherwise,} \end{cases}$$

$$E_{ij} = \begin{cases} -1, & i = j, \\ 1, & i = j + 2, \\ 0, & \text{otherwise.} \end{cases}$$

Using our previous definitions, we can see that Equations (A1) and (A2) can be combined into this compact form:

$$E_{(2n+2) \times 2n} \lambda_{2n \times 1} = \mathcal{I}_{(2n+2) \times (2n+2)} (P_{(2n+2) \times 1} - p_{(2n+2) \times 1}). \quad (\text{A13})$$

Similarly, Equations (A3) and (A4) can be expressed by:

$$D_{2n \times 2n} \mu_{2n \times 1} = T_{2n \times 2n} \lambda_{2n \times 1}. \quad (\text{A14})$$

Following our previous analysis, Equations (A5) and (A6) can be combined by:

$$2\eta a_{2n \times 1} = \frac{1}{2} T_{2n \times 2n} \lambda_{2n \times 1} + \mu_{2n \times 1}. \quad (\text{A15})$$

These three new Equations (A13)–(A15) can be used together to eliminate the dual variables λ_{x_i} , λ_{y_i} , μ_{x_i} and μ_{y_i} . To do that, we need to define two more matrices $M_{2n \times 2n}$ and $L_{2n \times (2n+2)}$ where

the matrix L is similar to the matrix M , but with an extra two columns of zeros at the end, as follows:

$$M_{2n \times 2n} = \begin{pmatrix} \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} & \mathbf{0} & \cdots & \mathbf{0} \\ \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} & \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} & \cdots & \mathbf{0} \\ \vdots & \ddots & \ddots & \vdots \\ \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} & \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} & \cdots & \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \end{pmatrix},$$

$$L_{2n \times (2n+2)} = \begin{pmatrix} \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} & \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} & \cdots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} & \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} & \cdots & \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} & \mathbf{0} \end{pmatrix}.$$

With these definitions, it can be easily verified that $D_{2n \times 2n} M_{2n \times 2n} = I_{2n \times 2n}$, which means that M is the inverse of D . Then, multiplying both sides of this identity by $(T\lambda)$ and using Equation (A14), it follows that:

$$\mu_{2n \times 1} = M_{2n \times 2n} T_{2n \times 2n} \lambda_{2n \times 1} \quad (\text{A16})$$

Similarly, it can be shown that $E_{(2n+2) \times 2n} L_{2n \times (2n+2)} = J_{(2n+2) \times (2n+2)}$, where J is defined as:

$$J_{(2n+2) \times (2n+2)} = \begin{pmatrix} \begin{bmatrix} +1 & 0 \\ 0 & +1 \end{bmatrix} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \begin{bmatrix} +1 & 0 \\ 0 & +1 \end{bmatrix} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \cdots & \cdots & \begin{bmatrix} +1 & 0 \\ 0 & +1 \end{bmatrix} & \mathbf{0} \\ \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} & \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} & \cdots & \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} & \mathbf{0} \end{pmatrix}.$$

Multiplying the identity by:

$$\mathcal{I}(\mathbf{P} - \mathbf{p}) = \begin{pmatrix} \begin{bmatrix} S_{x_0}(X_0 - x_0) + S_{x_0 y_0}(Y_0 - y_0) \\ S_{y_0}(Y_0 - y_0) + S_{x_0 y_0}(X_0 - x_0) \end{bmatrix} \\ \vdots \\ \begin{bmatrix} S_{x_{n+1}}(X_{n+1} - x_{n+1}) + S_{x_{n+1} y_{n+1}}(Y_{n+1} - y_{n+1}) \\ S_{y_{n+1}}(Y_{n+1} - y_{n+1}) + S_{x_{n+1} y_{n+1}}(X_{n+1} - x_{n+1}) \end{bmatrix} \end{pmatrix},$$

we find that:

$$J\mathcal{I}(\mathbf{P} - \mathbf{p}) = \begin{pmatrix} \begin{bmatrix} S_{x_0}(X_0 - x_0) + S_{x_0y_0}(Y_0 - y_0) \\ S_{y_0}(Y_0 - y_0) + S_{x_0y_0}(X_0 - x_0) \end{bmatrix} \\ \vdots \\ \begin{bmatrix} S_{x_{n+1}}(X_{n+1} - x_{n+1}) + S_{x_{n+1}y_{n+1}}(Y_{n+1} - y_{n+1}) \\ S_{y_{n+1}}(Y_{n+1} - y_{n+1}) + S_{x_{n+1}y_{n+1}}(X_{n+1} - x_{n+1}) \end{bmatrix} \\ \begin{bmatrix} \sum_{i=0}^{n-1} -(S_{x_i}(X_i - x_i) + S_{x_iy_i}(Y_i - y_i)) \\ \sum_{i=0}^{n-1} -(S_{y_i}(Y_i - y_i) + S_{x_iy_i}(X_i - x_i)) \end{bmatrix} \end{pmatrix}.$$

Therefore, using the identity of J and Equation (A13), it follows:

$$EL\mathcal{I}(\mathbf{P} - \mathbf{p}) - E\boldsymbol{\lambda} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \begin{bmatrix} -\sum_{i=0}^n (S_{x_i}(X_i - x_i) + S_{x_iy_i}(Y_i - y_i)) \\ -\sum_{i=0}^n (S_{y_i}(Y_i - y_i) + S_{x_iy_i}(X_i - x_i)) \end{bmatrix} \end{pmatrix}.$$

However, $(\sum_{i=0}^n (S_{x_i}(X_i - x_i) + S_{x_iy_i}(Y_i - y_i)) = 0)$ holds; see Equation (A1). Similarly, $(\sum_{i=0}^n (S_{y_i}(Y_i - y_i) + S_{x_iy_i}(X_i - x_i)) = 0)$ holds from Equation (A2). Therefore:

$$\boldsymbol{\lambda}_{2n \times 1} = L_{(2n) \times (2n+2)} \mathcal{I}_{(2n+2) \times (2n+2)} (\mathbf{P} - \mathbf{p})_{(2n+2) \times 1} \quad (\text{A17})$$

Using the two identities for $\boldsymbol{\mu}$ and $\boldsymbol{\lambda}$ (Equations (A16) and (A17)) in Equation (A15), we get:

$$2\eta \mathbf{a} = \left(\frac{1}{2} TL + MTL\right) \mathcal{I}(\mathbf{P} - \mathbf{p}) \quad (\text{A18})$$

Now, Equations (A7) and (A9) can be combined to eliminate v_{x_i} , so that:

$$x_{i+1}T_{i-1} - x_i(T_i + T_{i-1}) + x_{i-1}T_i = \frac{1}{2}(a_{x_i}T_i^2T_{i-1} + a_{x_{i-1}}T_{i-1}^2T_i) \quad (\text{A19})$$

Similarly, from Equations (A8) and (A10), it follows that:

$$y_{i+1}T_{i-1} - y_i(T_i + T_{i-1}) + y_{i-1}T_i = \frac{1}{2}(a_{y_i}T_i^2T_{i-1} + a_{y_{i-1}}T_{i-1}^2T_i) \quad (\text{A20})$$

To combine Equations (A19) and (A20), we need to define two additional matrices $G_{(2n-2) \times 2n}$ and $B_{(2n-2) \times (2n+2)}$ as follows:

$$\begin{aligned}
 G &= \begin{pmatrix} G_{1,1} & G_{2,1} & \cdots & \mathbf{0} \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \cdots & G_{1,n-1} & G_{2,n-1} \end{pmatrix} \\
 G_{1,i} &= \begin{bmatrix} T_{i-1}^2 T_i & 0 \\ 0 & T_{i-1}^2 T_i \end{bmatrix} \\
 G_{2,i} &= \begin{bmatrix} T_{i-1} T_i^2 & 0 \\ 0 & T_{i-1} T_i^2 \end{bmatrix}, \quad \text{where } i = 1, \dots, n-1. \\
 B &= \begin{pmatrix} B_{1,1} & B_{2,1} & B_{1,1-1=0} & \cdots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \cdots & B_{1,n-1} & B_{2,n-1} & B_{1,n-2} \end{pmatrix} \\
 B_{1,i} &= \begin{bmatrix} T_i & 0 \\ 0 & T_i \end{bmatrix} \\
 B_{2,i} &= \begin{bmatrix} -(T_{i-1} + T_i) & 0 \\ 0 & -(T_{i-1} + T_i) \end{bmatrix} \quad \text{where } i = 1, \dots, n-1.
 \end{aligned}$$

Using these two matrices, it is easy to see that Equations (A19) and (A20) are combined as:

$$Bp = \frac{1}{2}Ga \quad (\text{A21})$$

Now, Equations (A18) and (A21) can be combined, and we obtain the equation:

$$(\eta B + AI)p = AIP, \quad (\text{A22})$$

where:

$$A_{(2n-2) \times (2n+2)} = \frac{1}{4}G\left(\frac{1}{2}TL + MTL\right). \quad (\text{A23})$$

Finally, to include the additional two conditions $\sum_{i=0}^n (S_{x_i}(X_i - x_i) + S_{x_i y_i}(Y_i - y_i)) = 0$ and $\sum_{i=0}^n (S_{y_i}(Y_i - y_i) + S_{x_i y_i}(X_i - x_i)) = 0$, we define a $(2n \times (2n+2))$ matrix $\bar{B}$ to be matrix B augmented with two final rows of zeros and a $(2n \times (2n+2))$ matrix $\bar{A}$ to be A augmented with two final rows of 2×2 identity matrices, then we get the final equation:

$$(\eta \bar{B} + \bar{A}I)p = \bar{A}IP \quad (\text{A24})$$

which can be solved by singular-value decomposition to give the approximation trajectory. Similar to the scalar case, to avoid the large errors at the end of the trajectory, we reverse the columns and rows of matrices A and B to obtain the desired solution. Therefore, we expect that the errors at the beginning of our estimations will now be large [11,12].

Appendix B. Numerical Investigation Tables

Table A1. Results of the experiments to identify the almost optimum value of η giving the smallest $\langle E \rangle$ when Correlation is Accounted for (CA).

Experiment 1	η $\langle E \rangle$	0.1 0.34	1 0.28	10 0.24	100 0.26	1000 0.46
Experiment 2	η $\langle E \rangle$	1 0.27	5 0.24	10 0.23	50 0.24	100 0.25
Experiment 3	η $\langle E \rangle$	5 0.235	7 0.230	10 0.227	30 0.228	50 0.237
Experiment 4	η $\langle E \rangle$	7 0.234	10 0.229	15 0.227	20 0.226	30 0.228
Experiment 5	η $\langle E \rangle$	15 0.2361	17 0.2358	20 0.2357	25 0.2364	30 0.2374
Experiment 6	η $\langle E \rangle$	17 0.22406	18 0.22400	20 0.22402	22 0.22420	25 0.22470

Table A2. Results of the experiments to identify the almost optimum value of η giving the smallest $\langle E \rangle$ when Correlation is Ignored (CI).

Experiment 1	η $\langle E \rangle$	0.1 0.37	1 0.28	10 0.23	100 0.24	1000 0.43
Experiment 2	η $\langle E \rangle$	1 0.291	5 0.246	10 0.233	50 0.231	100 0.249
Experiment 3	η $\langle E \rangle$	10 0.23	30 0.22	50 0.23	70 0.23	100 0.25
Experiment 4	η $\langle E \rangle$	10 0.2361	20 0.2299	30 0.2298	40 0.2317	50 0.2345
Experiment 5	η $\langle E \rangle$	20 0.227	25 0.226	30 0.227	35 0.228	40 0.229
Experiment 6	η $\langle E \rangle$	20 0.22772	22 0.22737	25 0.22716	27 0.22718	30 0.22741

Table A3. Results of the experiments to identify the almost optimum value of η giving the smallest $\langle e \rangle$ when Correlation is Accounted for (CA).

Experiment 1	η $\langle e \rangle$	0.1 0.86	1 0.77	10 0.59	100 0.47	1000 0.91
Experiment 2	η $\langle e \rangle$	10 0.55	100 0.47	200 0.51	500 0.70	1000 0.95
Experiment 3	η $\langle e \rangle$	10 0.55	75 0.49	100 0.50	150 0.52	200 0.54
Experiment 4	η $\langle e \rangle$	10 0.5777	50 0.4976	75 0.4863	90 0.4861	100 0.4851
Experiment 5	η $\langle e \rangle$	75 0.4690	80 0.4678	90 0.4664	95 0.4664	100 0.4669
Experiment 6	η $\langle e \rangle$	90 0.47355	91 0.47354	92 0.47356	93 0.47358	95 0.47368

Table A4. Results of the experiments to identify the almost optimum value of η giving the smallest $\langle e \rangle$ when Correlation is Ignored (CI).

Experiment 1	η $\langle e \rangle$	1 0.71	10 0.59	100 0.45	1000 0.36	10,000 0.47
Experiment 2	η $\langle e \rangle$	100 0.40	500 0.36	1000 0.35	5000 0.40	10,000 0.50
Experiment 3	η $\langle e \rangle$	500 0.349	750 0.336	1000 0.329	2500 0.327	5000 0.365
Experiment 4	η $\langle e \rangle$	1000 0.364	2000 0.353	2500 0.356	3500 0.367	5000 0.392
Experiment 5	η $\langle e \rangle$	1000 0.354	1500 0.344	2000 0.340	2250 0.341	2500 0.341
Experiment 6	η $\langle e \rangle$	1500 0.36438	1750 0.36315	2000 0.36319	2100 0.36348	2250 0.36416
Experiment 7	η $\langle e \rangle$	1500 0.3394	1600 0.3391	1750 0.3392	1850 0.3395	2000 0.3402

References

- Bar-Shalom, Y.; Li, X.R.; Kirubarajan, T. *Estimation with Applications to Tracking and Navigation*; John Wiley and Sons: Hoboken, NJ, USA, 2001.
- Bilik, I.; Tabrikian, J. Maneuvering Target Tracking in the Presence of Glint using the Nonlinear Gaussian Mixture Kalman Filter. *IEEE Trans. Aerosp. Electron. Syst.* **2010**, *46*, 246–262. [CrossRef]
- Jazwinski, A.H. *Stochastic Processes and Filtering Theory*; Academic Press: New York, NY, USA, 1970; Volume 64.
- Julier, S.J.; Uhlmann, J.K. New extension of the Kalman filter to nonlinear systems. In Proceedings of the Signal Processing, Sensor Fusion, and Target Recognition VI, Orlando, FL, USA, 28 July 1997; Volume 3068, pp. 182–194. [CrossRef]
- Mahler, R.P.S. *Statistical Multisource-Multitarget Information Fusion*; Artech House: Norwood, MA, USA, 2007.
- Stone, L.D.; Streit, R.L.; Corwin, T.L. *Bayesian Multiple Target Tracking*, 2nd ed.; Artech House Publishers: Boston, MA, USA, 2014.
- Teoh, S.; Braunl, T. A Reliability Point and Kalman Filter-based Vehicle Tracking Technique. In Proceedings of the International Conference on Intelligent Systems (ICIS 2012), Penang, Malaysia, 19 May 2012; pp. 134–138.
- Li, T.C.; Su, J.; Liu, W.; Corchado, J.M. Approximate Gaussian conjugacy: Parametric recursive filtering under nonlinearity, multimodality, uncertainty, and constraint, and beyond. *Front. Inf. Technol. Electron. Eng.* **2017**, *18*, 1913–1939. [CrossRef]
- Doucet, A.; Johansen, A.M. *A Tutorial on Particle Filtering and Smoothing: Fifteen Years Later*; Handbook of nonlinear filtering. Available online: <https://warwick.ac.uk/fac/sci/statistics/staff/academic-research/johansen/publications/dj11.pdf> (accessed on 19 January 2019).
- Martino, L.; Elvira, V.; Camps-Valla, G. Group Importance Sampling for Particle Filtering and MCMC. *Digit. Signal Process.* **2018**, *82*, 133–151. [CrossRef]
- Zaitouny, A.A.; Stemler, T.; Judd, K. Tracking Rigid Bodies Using Only Position Data: A Shadowing Filter Approach Based on Newtonian Dynamics. *Digit. Signal Process.* **2017**, *67*, 81–90. [CrossRef]
- Zaitouny, A.A.; Stemler, T.; Small, M. Tracking a Single Pigeon Using a Shadowing Filter Algorithm. *Ecol. Evolut.* **2017**, *7*, 4419–4431. [CrossRef] [PubMed]
- Zaitouny, A.A.; Stemler, T.; Small, M. Modelling and Tracking the Flight Dynamics of Flocking Pigeons Based on Real GPS Data (Small Flock). *Ecol. Model.* **2017**, *344*, 62–72. [CrossRef]
- Anderson, B.D.O.; Moore, J.B. *Optimal Filtering*; Prentice-Hall: Englewood Cliffs, NJ, USA, 1979.
- Swanson, K.; Vautard, R. Four-dimensional variational assimilation and predictability in a quasi-geostrophic model. *Tellus* **1998**, *50A*, 369–390. [CrossRef]
- Ridout, D.; Judd, K. Convergence properties of gradient descent noise reduction. *Phys. D* **2002**, *165*, 27–48. [CrossRef]

17. Stemler, T.; Judd, K. A guide to shadowing filters for forecasting and state estimation. *Phys. D* **2009**, *238*, 1260–1273. [CrossRef]
18. Judd, K. Bayesian reconstruction of chaotic times series: Right results for the wrong reasons. *Phys. Rev. E* **2003**, *67*, 026212. [CrossRef]
19. Judd, K.; Stemler, T. Failure of sequential bayesian filters and the success of shadowing filters in tracking of nonlinear deterministic and stochastic systems. *Phys. Rev. E* **2009**, *79*, 066206. [CrossRef] [PubMed]
20. Judd, K.; Reynolds, C.A.; Rosmond, T. *Toward Shadowing in Operational Weather Prediction*; Technical Report NRL/MR/7530-04-18; Naval Research Laboratory: Monterey, CA, USA, 2004.
21. Hartikainen, J.; Solin, A.; Särkkä, S. *Optimal Filtering With Kalman Filters and Smoothers*; Department of Biomedical Engineering and Computational Sciences, Aalto University School of Science. Available online: <http://becs.aalto.fi/en/research/bayes/ekfukf/documentation.pdf> (accessed on 19 January 2019).
22. Bugallo, M.; Xu, S.; Djurić, P. Performance Comparison of EKF and Particle Filtering Methods for Maneuvering Targets. *Digit. Signal Process.* **2007**, *17*, 774–786. [CrossRef]
23. Gustafsson, F.; Gunnarsson, F.; Bergman, N.; Forssell, U.; Jansson, J.; Karlsson, R.; Nordlund, P. *Particle Filters for Positioning, Navigation and Tracking*; Linköping University Electronic Press: Linköping, Sweden, 2001.
24. Míguez, J. Analysis of Selection Methods for Cost-reference Particle Filtering with Applications to Maneuvering Target Tracking and Dynamic Optimization. *Digit. Signal Process.* **2007**, *17*, 787–807. [CrossRef]
25. Press, W.H.; Teukolsky, S.A.; Vetterling, W.T.; Flannery, B.P. Numerical Recipes. In *The Art of Scientific Computing*; Cambridge University Press: Cambridge, UK, 2007.
26. Courant, R. Variational methods for the solution of problems of equilibrium and vibrations. *Bull. Am. Math. Soc.* **1943**, *49*, 1–23. [CrossRef]
27. Judd, K. Tracking an Object with Unknown Accelerations using a Shadowing Filter. *arXiv* **2015**, arXiv:1502.07743.
28. Chan, Y.T.; Ho, K.C. A simple and efficient estimator for hyperbolic location. *IEEE Trans. Signal Process.* **1994**, *42*, 1905–1915. [CrossRef]
29. Chandrasekaran, S.; Golub, G.H.; Gu, M.; Sayed, A.H. Parameter estimation in the presence of bounded data uncertainties. *SIAM J. Matrix Anal. Appl.* **1998**, *19*, 235–252. [CrossRef]
30. El Ghaoui, L.; Lebret, H. Robust solutions to least-squares problems with uncertain data. *SIAM J. Matrix Anal. Appl.* **1997**, *18*, 1035–1064. [CrossRef]
31. Huang, Y.; Benesty, J.; Elko, G.W.; Mersereau, R.M. Real-time passive source localization: A practical linear-correction least-squares approach. *IEEE Trans. Speech Audio Process.* **2001**, *9*, 943–956. [CrossRef]
32. Li, T.; Chen, H.; Sun, S.; Corchado, J.M. Joint smoothing and tracking based on continuous-time target trajectory fitting. *IEEE Trans. Autom. Sci. Eng.* **2018**. [CrossRef]
33. Li, T. Single-Road-Constrained Positioning Based on Deterministic Trajectory Geometry. *IEEE Commun. Lett.* **2019**, *23*, 80–83. [CrossRef]
34. DeCesare, N.J.; Squires, J.R.; Kolbe, J.A. Effect of forest canopy on GPS-based movement data. *Wildl. Soc. Bull.* **2005**, *33*, 935–941. [CrossRef]
35. Grémillet, D.; Dell’Omo, G.; Ryan, P.G.; Peters, G.; Ropert-Coudert, Y.; Weeks, S.J. Offshore diplomacy, or how seabirds mitigate intra-specific competition: A case study based on GPS tracking of Cape gannets from neighbouring colonies. *Mar. Ecol. Prog. Ser.* **2004**, *268*, 265–279. [CrossRef]
36. Lorenz, K. Deterministic Nonperiodic Flow. *J. Atmos. Sci.* **1963**, *20*, 130–141. [CrossRef]
37. Kantz, H.; Schreiber, T. *Nonlinear Time Series Analysis*; Cambridge University Press: Cambridge, UK, 1997.
38. Urbanowicz, K.; Holyst, J.A.; Stemler, T.; Benner, H. Noise reduction in chaotic time series by a local projection with nonlinear constraints. *ACTA Phys. Polonica B* **2004**, *35*, 2175–2197.
39. Lukeman, R.; Li, Y.X.; Edelstein-Keshet, L. Inferring individual rules from collective behavior. *Proc. Natl. Acad. Sci. USA* **2010**, *107*, 12576–12580. [CrossRef]
40. Dieck Kattas, G.; Xu, X.K.; Small, M. Dynamical modeling of collective behavior from pigeon flight data: Flock cohesion and dispersion. *PLoS Comput. Biol.* **2012**, *8*, e1002449 [CrossRef]
41. Dieck Kattas, G.; Perez-Barberia, F.J.; Small, M.; Xu, X.K.; Walker, D.M. Unraveling complexity in interspecies interaction through nonlinear dynamical models. *Acta Ethol.* **2013**, *16*, 21–30. [CrossRef]
42. Godley, B.J.; Broderick, A.C.; Glen, F.; Hays, G.C. Post-nesting movements and submergence patterns of loggerhead marine turtles in the Mediterranean assessed by satellite tracking. *J. Exp. Mar. Biol. Ecol.* **2003**, *287*, 119–134. [CrossRef]

43. Ryan, P.G.; Petersen, S.L.; Peters, G.; Grémillet, D. GPS tracking a marine predator: The effects of precision, resolution and sampling rate on foraging tracks of African Penguins. *Mar. Biol. Int. J. Life Oceans Coast. Waters* **2004**, *145*, 215–223. [[CrossRef](#)]
44. Steiner, I.; Bürgi, C.; Werffeli, S.; Dell’Omo, G.; Valenti, P.; Tröster, G.; Wolfer, D.P.; Lipp, H.-P. A GPS logger and software for analysis of homing in pigeons and small mammals. *Physiol. Behav.* **2000**, *71*, 589–596. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).