SPECIAL LatMix COLLECTION

Smoothing and Interpolating Noisy GPS Data with Smoothing Splines

JEFFREY J. EARLY

NorthWest Research Associates, Redmond, Washington

ADAM M. SYKULSKI

Data Science Institute, and Department of Mathematics and Statistics, Lancaster University, Lancaster, United Kingdom

(Manuscript received 21 May 2019, in final form 20 December 2019)

ABSTRACT

A comprehensive method is provided for smoothing noisy, irregularly sampled data with non-Gaussian noise using smoothing splines. We demonstrate how the spline order and tension parameter can be chosen a priori from physical reasoning. We also show how to allow for non-Gaussian noise and outliers that are typical in global positioning system (GPS) signals. We demonstrate the effectiveness of our methods on GPS trajectory data obtained from oceanographic floating instruments known as drifters.

1. Introduction

In 2011 an array of floating ocean surface buoys (drifters) were deployed in the Sargasso Sea to assess the lateral diffusivity of oceanic processes (Shcherbina et al. 2015). Each drifter was equipped with a global positioning system (GPS) receiver recording locations every 30 min. Addressing the primary goal of understanding the processes controlling lateral diffusivity requires significant processing of the drifter positions, including removing mean flow, accounting for the large-scale strain field, and analyzing the residual spectra for hints of a dynamical process. However, it quickly became clear that the GPS position data, which can have accuracies as low as a few meters (Wide Area Augmentation System T and E Team 2016), were contaminated by outliers with position jumps of hundreds of meters or more. Prior to analysis, the position data require removing outliers as well as interpolating gaps to keep the position data synchronized in time across the drifter array.

The basic problem is ubiquitous: observations from GPS receivers return observed positions x_i at times t_i that differ from the true positions $x_{\mathrm{true}}(t_i)$ by some noise $\epsilon_i \equiv x_i - x_{\mathrm{true}}(t_i)$ with variance σ^2. The goal of *smoothing* is to find the true position $x_{\mathrm{true}}(t_i)$ that is not contaminated by the noise, whereas the goal of *interpolating* is to find the true position $x_{\mathrm{true}}(t)$ between observation times. The approach taken here is to use smoothing splines. This approach is relatively broad (Handcock et al. 1994; Nychka 2000), and is related to the methods in Yaremchuk and Coelho (2015) and Elipot et al. (2016) for smoothing drifter trajectories, as discussed later.

Our model for the "true" path $x(t)$ is specified using interpolating B-splines $X^K(t)$ such that

$$x(t) = \sum_{i=1}^{N} \xi_i X_i^K(t), \qquad (1)$$

where K is the order (degree $S = K - 1$) of the spline. For N observations we construct N B-splines such that $x(t_i) = x_i$ for appropriately chosen coefficients ξ_i. To smooth the data, we choose new coefficients $\overline{\xi_i}$ that minimize the penalty function

$$\phi = \frac{1}{N}\sum_{i=1}^{N}\left[\frac{x_i - x(t_i)}{\sigma}\right]^2 + \frac{\lambda_T}{t_N - t_1}\int_{t_1}^{t_N}\left(\frac{d^T x}{dt^T}\right)^2 dt, \qquad (2)$$

Denotes content that is immediately available upon publication as open access.

Corresponding author: Jeffrey J. Early, jearly@nwra.com

DOI: 10.1175/JTECH-D-19-0087.1

for some tension parameter $\lambda_T \geq 0$. If $\lambda_T = 0$ then $\phi = 0$ and $\xi_i = \overline{\xi_i}$ because $x(t_i) = x_i$, but if $\lambda_T \to \infty$ then this forces $x(t)$ to a Tth-order polynomial (e.g., when $T = 2$, the model is forced to be a straight line because it has no second derivative). The resulting path $x(t)$ is known as a smoothing spline and was first introduced in modern form by Reinsch (1967), but according to De Boor (1978) the idea dates back to Whittaker (1923). Once S and T are chosen, the smoothing spline has one free parameter (λ_T) and its optimal value can be found by minimizing the expected mean-square error when the true value of σ is known (Craven and Wahba 1978).

Three issues must be addressed before smoothing splines are applied to GPS data:

1) how to choose S and T—and how these choices affect the recovered power spectrum,
2) how to modify the spline fit to accommodate the non-Gaussian errors of GPS receivers, and
3) how to identify and remove outliers.

To address these issues but also to serve as a practical guide to other practitioners, we review B-splines in section 2 and introduce the canonical interpolating spline as the underlying model for path $x(t)$ in (1). We demonstrate the effect that choosing S has on the high-frequency slope of the power spectrum of the interpolated fit.

Section 3 takes a broad look at smoothing splines and the assumptions they make on the underlying process. Many of the ideas presented in this section are known to the statistics community, so here we present these ideas from a more physical perspective. We show that the penalty function in (2) can be formulated as a maximum-likelihood problem and that applying tension is equivalent to assuming a Gaussian distribution on the tensioned derivative of the underlying process.

Section 4 uses ensembles from synthetic data that mimic the oceanographic data to test a number of choices that must be made. We establish that setting $T = S$ is a reasonable choice. We show how the tension parameter can be chosen a priori (without optimization of the mean-square error) when the *effective sample size* (which we define later) can be estimated from the data. This estimate for effective sample size can be used to reduce the coefficients ξ^i in the spline fit without increasing mean-square error.

The second half of the paper addresses issues specific to GPS position errors. In section 5 we discuss the assumptions of stationarity and isotropy required for bivariate smoothing splines. In section 6 we show that GPS errors are not Gaussian distributed but rather are t distributed, and we show how to modify our method for a t distribution. Section 7 addresses how to modify our method to make smoothing splines robust to outliers. We compare with alternative methods and conclude in sections 8 and 9, respectively.

A major outcome of this work is the implementation of MATLAB classes for generating B-splines, interpolating splines, smoothing splines, and a class specific to smoothing GPS data (https://github.com/JeffreyEarly/GLNumericalModelingKit). These classes are highlighted throughout in relevant sections.

2. Interpolating spline

Assume that we are given N observations of a particle position (t_i, x_i) with no errors. The simplest form of interpolation is a nearest-neighbor method that assigns the position of the particle to the nearest observations in time. The resulting interpolated function $x(t)$ is a polynomial of order $K = 1$ (piecewise constant), shown in the top row of Fig. 1. The next level of sophistication is to assume a constant velocity between any two observations and to use that to interpolate positions between observations, as shown in the second row of Fig. 1. This means we now have a piecewise constant function dx/dt that represents the velocity of the particle, shown in the second row, second column, of Fig. 1. This is a polynomial function of order $K = 2$.

It is less obvious how to proceed to a polynomial of order $K = 3$. With N data points we can construct a piecewise constant acceleration (the second derivative) using the $N - 2$ independent accelerations computed from finite differencing, but where to place *knot points* that define the boundaries of the regions and how to maintain continuity is less clear. The approach taken here is to use B-splines.

a. B-Splines

A B-spline (or basis spline) of order K (degree $S = K - 1$) is a piecewise polynomial that maintains nonzero continuity across S knot points. The knot points are a nondecreasing collection of points in time denoted by τ_i. The basic theory is well documented in De Boor (1978), but here we present a reduced version tailored to our needs.

The mth B-spline of order $K = 1$ is defined as

$$X_m^1(t) \equiv \begin{cases} 1 & \text{if} \quad \tau_m \leq t < \tau_{m+1} \\ 0 & \text{otherwise} \end{cases}. \tag{3}$$

This is the rectangle function as shown in the first row, first column, of Fig. 2. Given P knot points we can construct $P - 1$ B-splines of order $K = 1$, although if a knot point is repeated it results in a spline that is zero everywhere. To represent an interpolating function $x(t)$

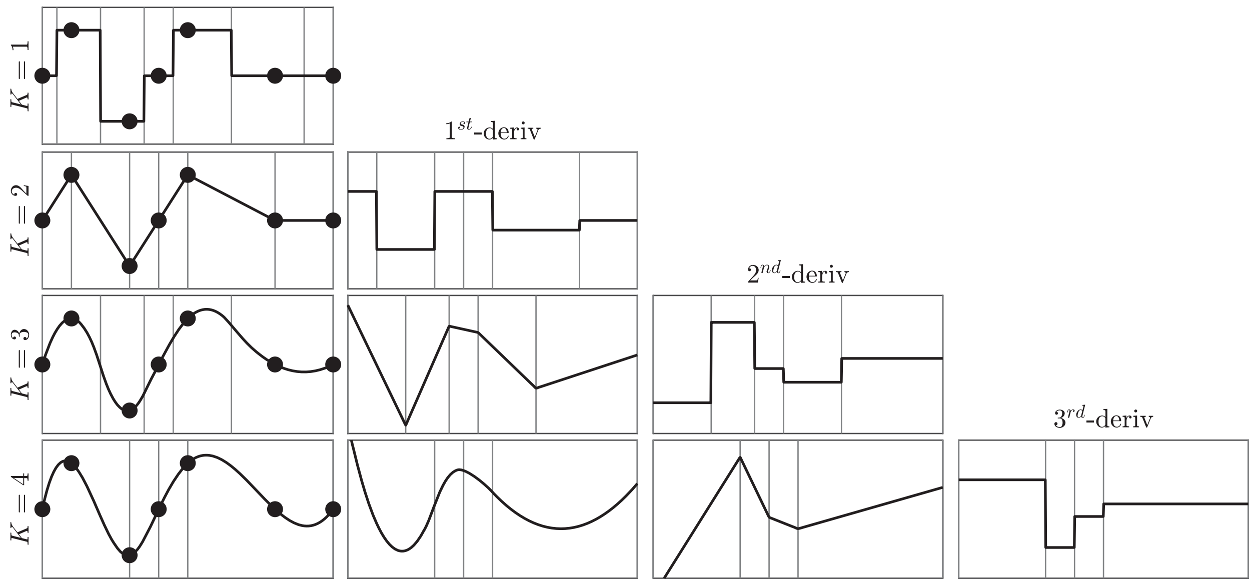

FIG. 1. An example of interpolating between seven data points using a spline function of order K. The data points are shown as circles, and the interpolated function is shown as solid black lines. We show four different orders of interpolation $K = 1, \ldots, 4$ (rows) and their nonzero derivatives (columns). The thin vertical gray lines are the knot points.

for the N observations of a particle position (t_i, x_i) we define $N + 1$ knot points as

$$\tau_m = \begin{cases} t_1 & m = 1 \\ t_{m-1} + \dfrac{t_m - t_{m-1}}{2} & 1 < m \le N. \\ t_N & m > N \end{cases} \quad (4)$$

This creates N independent basis functions that provide support for the region $t_1 \le t \le t_N$ (provided that the last spline is defined to include the last knot point). The interpolating function $x(t)$ is defined as $x(t) \equiv X_m^1(t)\xi^m$, where the coefficients ξ^m are found by solving $X_m^1(t^i)\xi^m = x^i$. The result of this process is shown in Fig. 1 for seven irregularly spaced data points.

All higher-order ($K > 1$) B-splines are defined by recursion,

$$X_m^K(t) \equiv \frac{t - t_m}{t_{m+K-1} - t_m} X_m^{K-1}(t) + \frac{t_{m+K} - t}{t_{m+K} - t_{m+1}} X_{m+1}^{K-1}(t). \quad (5)$$

This creates splines that span across one additional knot point at each order and maintain continuity across one more derivative. Examples are shown in Fig. 2.

Any knot points that are repeated T times result in a total of $T - 1$ splines of order 1 that are everywhere zero. This has the effect of introducing discontinuities in the derivatives for higher-order splines. For our purposes, we use this feature to prevent higher-order splines from crossing boundaries. For $K = 2$ order splines we use $N + 2$ knot points at locations

$$\tau_m = \begin{cases} t_1 & m \le 2 \\ t_{m-1} & 2 < m \le N. \\ t_N & m > N \end{cases} \quad (6)$$

This creates a knot point at every observation point but repeats the first and last knot point. This has the

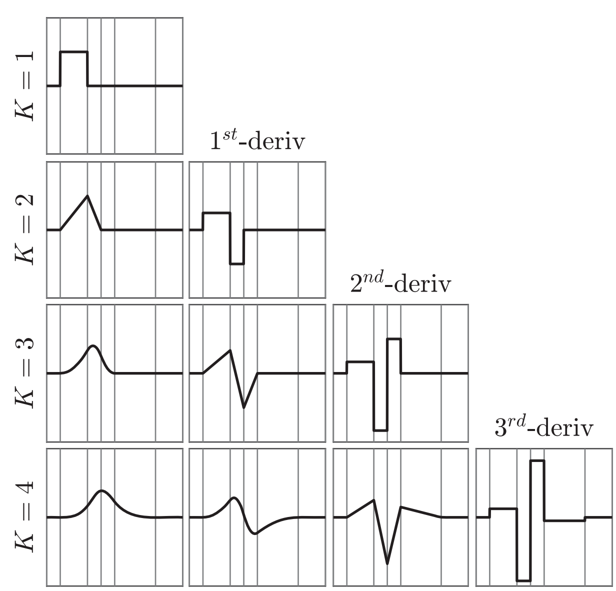

FIG. 2. The B-splines and derivatives (columns) for orders $K = 1, \ldots, 4$ (rows).

effect of terminating the first and last spline at the boundary and creating N second-order B splines, $X_m^2(t)$. The interpolating function $x(t)$ is defined as $x(t) \equiv X_m^2(t)\xi^m$, where the coefficients ξ^m are found by solving $X_m^2(t^i)\xi^m = x^i$. The second row of Fig. 1 shows an example.

This process can be continued to higher-order B-splines. For splines that are of *even* order, we create $N + K$ knot points with

$$\tau_m^{K-\mathrm{even}} = \begin{cases} t_1 & m \le K \\ t_{m-K/2} & K < m \le N \\ t_N & m > N \end{cases}, \tag{7}$$

and, for splines that are *odd* order, we create $N + K$ knot points with

$$\tau_m^{K-\mathrm{odd}} = \begin{cases} t_1 & m \le K \\ t_{m-\frac{K+1}{2}} + \dfrac{t_{m+1-\frac{K+1}{2}} - t_{m-\frac{K+1}{2}}}{2} & K < m \le N \\ t_N & m > N \end{cases} \cdot \tag{8}$$

The knot points are chosen to create N splines for the N data points such that the interpolated function $x(t)$ crosses all N observations (t_i, x_i). The path $x(t)$ is the *canonical interpolating spline* of order K. Examples are shown in Fig. 1.

The knot placements in (7) and (8) are equivalent to the *not-a-knot* boundary conditions described in De Boor (1978) and used in the cubic spline implementation in MATLAB. In the usual formulation of the not-a-knot boundary condition, the knot positions do not change as a function of spline order, and therefore additional constraints must be added at each order—especially the requirement that the highest derivative maintain continuity near the boundaries. In the formulation here, these constraints are implicit in (7) and (8).

b. Numerical implementation

The root class in our suite of MATLAB classes is the BSpline class, which evaluates a complete B-spline basis set given a set of knot points. This class was used to generate Fig. 2.

The interpolating spline used to generate Fig. 1 is implemented in the InterpolatingSpline class—a subclass of BSpline. This class generates interpolating splines of arbitrary order given a set of data points (t_i, x_i), thus generalizing the cubic spline command built in to MATLAB.

c. Synthetic data

Throughout this paper we generate synthetic data for both the signal and the noise. The velocity of the signal is generated from a bivariate Gaussian process known as the Matérn (Lilly et al. 2017). The spectrum of the Matérn is given by

$$S(\omega) = \frac{A^2}{(\omega^2 + \lambda^2)^{p/2}}, \tag{9}$$

where ω is the frequency, A sets the amplitude, $p > 1$ sets the high-frequency slope, and λ sets the frequency below which the signal looks increasingly white. This spectrum has finite amplitude at low frequencies and power-law falloff at high frequencies, two physically realistic properties observed in ocean surface drifters (Sykulski et al. 2016). Trajectories from this velocity spectrum will be generated using the "maternoise" function available in jLab (Lilly 2019). In our experiments, the parameter A is chosen such that the square root of velocity variance in each direction is $u_{\mathrm{rms}} = 0.20\,\mathrm{m\,s^{-1}}$ and the damping scale is $\lambda^{-1} = 30\,\mathrm{min}$. Values of p are varied with $p = 2, 3$, and 4 so that the high-frequency spectrum is proportional to ω^{-2}, ω^{-3}, and ω^{-4}. Velocities are sampled every minute and are integrated to get positions. Figure 3 shows an example velocity spectrum of the signal with $p = 2$.

The position data are contaminated with (white) Gaussian noise with $\sigma = 10\,\mathrm{m}$, a value chosen to resemble GPS errors. For all experiments we use a range of *strides*, that is, subsampled versions of the underlying process as input into the spline fits. A stride of 100 indicates that the signal is subsampled to 1 in every 100 data points. This lets us evaluate the quality of fit against different strides. In analyzing the quality of fits, we use velocities when computing the power spectrum, but report mean-square errors from positions.

d. Spline degree S

We first examine a synthetic signal *uncontaminated* by noise, to examine the role of the spline degree S on the interpolated fit. As noted in Craven and Wahba (1978), the degree of the spline sets its roughness. In terms of the power spectrum, this corresponds to the high-frequency slope as can be seen in Fig. 3, which shows fits with $S = 1, \ldots, 4$. Setting $S = 1$ produces a high-frequency falloff in the spline fit of ω^{-2}. Although this appears to be a desirable feature when fitting to a process with true slope ω^{-2}, the mean-square error is consistently higher (as indicated in the legend of Fig. 3).

The bottom panel of Fig. 3 shows the coherence between the spline fit and the true signal. A coherence of 1 indicates that the signals are perfectly matched at a given frequency, while a coherence of 0 indicates that the signals are unrelated. There is no discernible difference

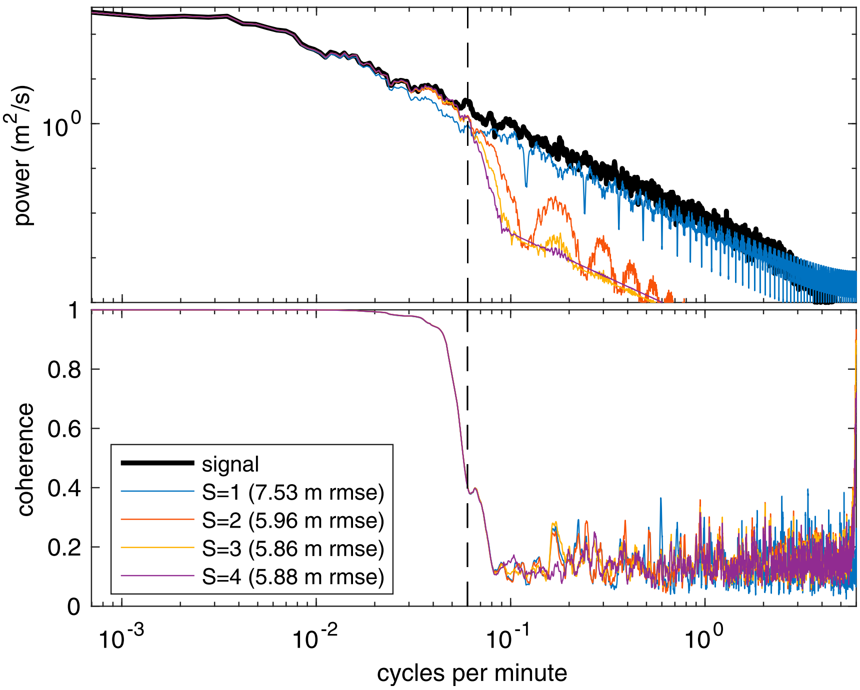

FIG. 3. (top) The velocity spectrum of a synthetic Lagrangian velocity generated from the Matérn (black). The blue, red, and orange lines show the spectrum of the interpolating spline fit to the data with a stride of 100 for $S = 1, \ldots, 4$, respectively. (bottom) The coherence between the smoothed velocities and the true velocity. The dashed vertical line denotes the Nyquist frequency of the strided data.

in coherence between spline fits with $S = 1, \ldots, 4$. The coherence quickly drops to near zero at the same frequency in all three cases. The implication here is that the spline fits are essentially producing noise at frequencies above the loss of coherence. This is why the spline fits with shallower slopes (with more variance at high, incoherent frequencies) produce a larger overall mean-square error than those with steeper slopes (with less variance at high, incoherent frequencies). The conclusion here is that smoother is better: it is better to use an unnecessarily high-order spline to avoid adding extra noise at high frequencies.

3. Smoothing spline

A typical starting point for maximum likelihood is to establish the probability distribution function (PDF) of the errors, $\epsilon_i \equiv x_i - x_{\text{true}}(t_i)$. The canonical example in one dimension (e.g., Press et al. 1992) is to assume errors are independently drawn from a Gaussian with the following probability distribution

$$p_g(\varepsilon|\sigma_g) = \frac{\exp\left(-\frac{1}{2}\frac{\varepsilon^2}{\sigma_g^2}\right)}{\sigma_g(2\pi)^{1/2}}, \qquad (10)$$

where σ_g is the standard deviation. This assumption alone places no assumptions on the signal, only on the structure of the noise.

The probability of the observed data given model $x(t)$ is

$$P = \frac{1}{\sigma\sqrt{2\pi}}\prod_{i=1}^{N}\exp\left\{-\frac{1}{2}\left[\frac{x_i - x(t_i)}{\sigma}\right]^2\right\}, \qquad (11)$$

where we have taken $\sigma = \sigma_g$. Maximizing the probability function in (11) is the same as minimizing its argument—this is the logarithmic likelihood (up to a constant), called the penalty function

$$\phi = \frac{1}{N}\sum_{i=1}^{N}\left[\frac{x_i - x(t_i)}{\sigma}\right]^2. \qquad (12)$$

Stated in this way this is the same as asking for the "least squares" fit of the errors.

a. Smoothing-spline penalty function

The model used here is the canonical interpolating spline of order K described in section 2. We have chosen our knot points such that the model intersects the observations and this certainly maximizes (11) [and minimizes (12)] because all the errors are zero, but the resulting distribution of errors (a delta function at zero) does not resemble the assumed Gaussian distribution. Thus, additional constraints are required if the assumed error distribution is to be recovered.

The smoothing spline augments the penalty function of (12) by adding a global constraint on the Tth derivative of the resulting function as in (2). If $\lambda_T \to 0$ then this reduces to the least squares fit in (12), but if $\lambda_T \to \infty$ then this forces the model to a Tth-order polynomial.

To interpret the first term of (2), consider a motionless particle at true position x_0. Using the N relevant observations x_i, the *sample mean* $\overline{x} = (1/N)\sum x_i$ estimates the particle's position x_0. The unbiased *sample variance* estimates the variance σ^2 of the noise, and is given by $\hat{\sigma}^2 = (1/N-1)\sum(x_i - \overline{x})^2$, the expected value of which is $\langle\hat{\sigma}^2\rangle = [1-(1/N)]\sigma^2$.

Now consider the opposite extreme in which the particle is moving so fast (or the observations are so sparse) that each observation is independent of its neighbors. In this case, each observation must be considered separately, so the sample mean at time t_i is $\overline{x}_i = x_i$ (i.e., we are summing over the single relevant observation). In this scenario we cannot produce a sample variance, because there is only a single relevant observation at time t_i.

In practice, the number of relevant observations is anywhere between 1 and N. Here we use the term *effective sample size*, denoted by n_{eff}, to describe the typical number of observations being used to estimate either the particle's position or the variance of the noise at any given time. In this context, the first term of (2) is proportional to an ensemble of multiple estimates of the sample variance

$$\hat{\sigma}^2 \equiv \frac{1}{N}\sum_{i=1}^{N}[x_i - x(t_i)]^2, \qquad (13)$$

which is expected to scale as

$$\langle\hat{\sigma}^2\rangle = \left(1 - \frac{1}{n_{\text{eff}}^{\text{var}}}\right)\sigma^2, \qquad (14)$$

where $1 < n_{\text{eff}}^{\text{var}} \le N$ is our definition of the effective sample size of the sample variance. Revisiting the limiting cases, as $n_{\text{eff}}^{\text{var}} \to N$ the sample variance matches the true variance; as $n_{\text{eff}}^{\text{var}} \to 1$, the sample variance vanishes.

There is a simple physical interpretation for the second term in (2). Consider the case $T = 1$ so that the smoothing spline is a constraint on velocity. When averaged over the integration time, the integral produces the root-mean-square velocity u_{rms} such that the second term scales as u_{rms}^2. In general, where $x_{\text{rms}}^{(T)}$ is the root-mean-square of the Tth derivative, this means λ_T scales like

$$\lambda_T = \left(1 - \frac{1}{n_{\text{eff}}^{\text{var}}}\right)\frac{1}{\left[x_{\text{rms}}^{(T)}\right]^2}. \qquad (15)$$

The interpretation of the smoothing spline is that the two terms are balanced by a relative weighting of the sample variance of the noise and mean-square of the Tth derivative of the physical process. As discussed in section 4, both $x_{\text{rms}}^{(T)}$ and $n_{\text{eff}}^{\text{var}}$ can be estimated a priori such that a good initial estimate for λ_T can be made.

b. *Smoothing-spline maximum likelihood*

The penalty function in (2) can be restated in terms of maximum likelihood under some conditions (see chapter 3.8 in Green and Silverman 1993). Assume that in addition to knowing the distribution of errors as in (11), we know how the velocity of the underlying physical process is distributed. For example, in geophysical turbulence the velocity probability distribution function is like a Laplace distribution (Bracco et al. 2000). To recover the smoothing spline, we consider the case where the velocity PDF is Gaussian. Stated as maximum likelihood, this means at *any given instant* (and not just the times of observation) we expect the model velocity to be Gaussian. We discretize the problem by sampling the velocity Q times $t_q = t_1 + q\Delta t_q$, where $\Delta t_q = (t_N - t_1)/(Q-1)$ and $q = 0, \ldots, Q-1$. The maximum likelihood is thus stated as

$$P = \prod_{i=1}^{N}\frac{1}{\sigma\sqrt{2\pi}}\exp\left\{-\frac{1}{2}\left[\frac{x_i - x(t_i)}{\sigma}\right]^2\right\} \times \prod_{q=1}^{Q}\frac{\sqrt{\gamma}}{x_{\text{rms}}^{(T)}\sqrt{2\pi}}\exp\left\{-\frac{\gamma}{2}\left[\frac{x^{(T)}(t_q)}{x_{\text{rms}}^{(T)}}\right]^2\right\}, \qquad (16)$$

which is the joint probability of the error distribution from (11) and the velocity distribution of the underlying physical process. We include parameter γ to set the relative weighting between the two distributions, although it could be absorbed into the definition of $x_{\text{rms}}^{(T)}$. Writing (16) as a penalty function (after converting the product of exponentials into exponentials of sums), we have

$$-\log P = \frac{1}{2}\sum_{i=1}^{N}\left[\frac{x_i - x(t_i)}{\sigma}\right]^2 + \frac{\gamma}{2}\sum_{q=1}^{Q}\left[\frac{x^{(T)}(t_q)}{x_{\text{rms}}^{(T)}}\right]^2 + C, \qquad (17)$$

where C is a constant. Setting $\gamma = N/Q$ and renormalizing the penalty function by $2/N$ (which has no effect on the location of its minimum), (17) can be written as

$$\phi = \frac{1}{N}\sum_{i=1}^{N}\left[\frac{x_i - x(t_i)}{\sigma}\right]^2 + \frac{1}{t_N - t_1}\sum_{q=1}^{Q}\left[\frac{x^{(T)}(t_q)}{x_{\mathrm{rms}}^{(T)}}\right]^2 \Delta t_q. \quad (18)$$

Apart from the discretization of the integral, (18) is the same as the penalty function in (2).

There is an important special case when tension is applied at the same order as the spline, $T = S$. In this case the spline is piecewise constant for $x^{(T)}$ with exactly $N - T$ unique values. The parameter $\gamma = N/(N - T) \approx 1$ and (16) can be simplified. This case is appealing because only the $N - T$ unique values of the derivative $x^{(T)}$ that can be computed from N data points are being used for tension, which is not the case when $T < S$.

This maximum-likelihood perspective shows that adding tension to the penalty function is equivalent to assuming a higher-order derivative in the model (e.g., velocity if $T = 1$) is Gaussian. This is therefore making an assumption about the underlying *physical process* of the model. This is in contrast to the first term, which is entirely a statement about *measurement noise*.

Writing the smoothing spline as a maximum-likelihood condition (16), suggests that if the underlying physical process has a nonzero mean value in tension, the fit will not behave as expected. However, smoothing splines can be easily modified to accommodate a mean value in tension, as shown in appendix A.

c. Optimal parameter estimation

For a given choice of T and λ_T, the minimum solution to (2) can be found analytically [see Teanby (2007) and our appendix A]. Once the solution is found the smoothing matrix $\mathbf{S}_\lambda$ is defined as the matrix that takes observations $\mathbf{x}$ and maps them to their smooth values, $\hat{x} = \mathbf{S}_\lambda \mathbf{x}$.

The free parameter λ_T is a relative weighting between the two terms in (2). Choosing its optimal value can be done by minimizing the expected mean-square error (Craven and Wahba 1978),

$$\mathrm{MSE}(\lambda) = \frac{1}{N}\|(\mathbf{S}_\lambda - \mathbf{I})\mathbf{x}\|^2 + \frac{2\sigma^2}{N}\mathrm{Tr}\mathbf{S}_\lambda - \sigma^2, \quad (19)$$

where $\|\ \|^2$ is the Euclidean norm, Tr indicates the trace, and $\mathbf{I}$ is the identity matrix.

A significant amount of the literature on smoothing splines is devoted to minimizing the mean-square error when the variance σ^2 is *not* known. Craven and Wahba (1978) and Wahba (1978) use cross validation to estimate σ and minimize mean-square error. Recent work comparing different estimators shows no single technique to be optimal (Lee 2003). For our application, however, the errors in GPS data can be relatively easily established, as shown in section 6.

The mean-square error in (19) is a combination of the sample variance and the variance of the mean. As already discussed in the context of the penalty function ϕ in section 3a, the first term in (19) is an ensemble of sample variances, and therefore, by combining (13), (14), and (19) we obtain

$$\left(1 - \frac{1}{n_{\mathrm{eff}}^{\mathrm{var}}}\right)\sigma^2 = \frac{1}{N}\|(\mathbf{I} - \mathbf{S}_\lambda)\mathbf{x}\|^2. \quad (20)$$

The second term in (19) is proportional to 2 times the squared standard error, that is, the variance of the sample mean. As discussed in Teanby (2007), the quantity $\mathbf{S}_\lambda \boldsymbol{\Sigma}$ is the covariance matrix with the squared standard error along the diagonal and thus the mean-square standard error is given by $(1/N)\mathrm{Tr}(\mathbf{S}_\lambda \boldsymbol{\Sigma})$. The variance of the sample mean is known to scale inversely with the number of samples being used to estimate the mean. We use this to define the effective sample size of the variance of the mean, $n_{\mathrm{eff}}^{\mathrm{SE}}$ with

$$\frac{\sigma^2}{n_{\mathrm{eff}}^{\mathrm{SE}}} = \frac{1}{N}\mathrm{Tr}(\mathbf{S}_\lambda \boldsymbol{\Sigma}). \quad (21)$$

Taking the measures of effective sample size as functions of λ, the mean-square error can be expressed by combining (19)–(21):

$$\mathrm{MSE}(\lambda) = 2\frac{\sigma^2}{n_{\mathrm{eff}}^{\mathrm{SE}}} - \frac{\sigma^2}{n_{\mathrm{eff}}^{\mathrm{var}}}. \quad (22)$$

If one assumes $n_{\mathrm{eff}}^{\mathrm{var}} = n_{\mathrm{eff}}^{\mathrm{SE}}$, then the expected mean-square error from (19) is equal to $\sigma^2/n_{\mathrm{eff}}$. Although not shown here, in an empirical analysis we find that $n_{\mathrm{eff}}^{\mathrm{var}}$ and $n_{\mathrm{eff}}^{\mathrm{SE}}$ are approximately equal, although $n_{\mathrm{eff}}^{\mathrm{var}}$ becomes highly variable when $n_{\mathrm{eff}}^{\mathrm{SE}}$ approaches 1. These measures of effective sample size can be used to estimate the value of λ_T necessary for optimal tension without minimizing the expected mean-square error.

The definition of effective sample size used here is related to, but not the same as, the notion of degrees of freedom used in Cantoni and Hastie (2002) and references therein.

4. Spline order, tension order, and the spectrum

With a model path [(1)], a penalty function [(2)], and a minimization condition [(19)], we have all of the primary pieces to create a smoothing-spline interpolant to the data. However, a number of choices still must be made. In this section we use synthetically generated data to represent our physical process, and contaminate the process with Gaussian noise as described in section 2c.

TABLE 1. The 68th-percentile range of increase in mean-square error from the optimal fit.

		T				
		1	2	3	4	5
	1	33.8%–80.3%				
	2	14.0%–75.1%	0.8%–12.1%			
S	3	17.1%–77.5%	1.0%–13.1%	0.0%–4.5%		
	4	22.8%–81.9%	1.0%–14.5%	0.0%–4.6%	0.0%–6.3%	
	5	27.6%–91.4%	0.8%–15.4%	0.0%–4.6%	0.0%–6.1%	0.0%–12.8%

We test our ability to recover the signal and examine the effects of changing the spline and tension order on the mean-square error and the resulting spectrum.

The results of this section are empirical, and we acknowledge upfront that any conclusions reached *may* depend on our particular choice of physical model generating the signal. Nevertheless, our expectation is that the conclusions are ‘‘$O(1)$’’ correct and are applicable, at least, to our GPS-tracked drifter dataset.

a. Tension degree T

Given a smoothing spline of degree S, the tension in the penalty function (2) can be applied at any degree $T \leq S$. We use the synthetic data for the three different slopes to empirically establish the relationship between the tension degree T and the spline degree S.

For $S = 1, \ldots, 5$ and all $T \leq S$ we minimize the mean-square error against the true values. The minimization is performed for 200 ensembles of noise and signal with three slopes (ω^{-2}, ω^{-3}, and ω^{-4}) and five different strides. For a given slope, stride, and realization of noise, we identify the minimum mean-square error across S and T and compare all values of S and T as a percentage increase relative to that minimum. After aggregating across slopes, strides, and ensembles, the 68% confidence range is shown in Table 1. The table shows that setting $T = S$ is not always optimal but it is never significantly worse than the optimal choice. Thus for the remainder of the paper we set $T = S$.

b. Loss of coherence

The loss of coherence defines the time scale below which the smoothing spline is not providing useful information. A reasonable hypothesis is that this scale is related to the effective sample size, n_{eff} because this indicates how many points are being used to estimate the true value. Therefore, the loss of coherence occurs at the *effective Nyquist*, which we define as

$$f_s^{\text{eff}} \equiv \frac{1}{2n_{\text{eff}}\Delta t}. \tag{23}$$

In practice, we use $n_{\text{eff}}^{\text{SE}}$ because it is less variable than $n_{\text{eff}}^{\text{var}}$ for values near 1 and is the more direct measure of how many points are being used to estimate the model path. Figure 4 shows the power spectrum and coherence of optimal tension fits for three different strides of the data. In each case (23) indicates the approximate value where the coherence drops below 0.5.

c. Reduced spline coefficients

One practical consideration when working with large datasets is the computational cost of creating the spline fit, which is limited by the rate of solving for the spline coefficients. It is beneficial to reduce knot points (and therefore total splines) where possible. A reasonable strategy is that when the effective sample size is large, as measured by (21), we avoid placing a knot point at every data point—essentially ‘‘skipping’’ data points.

To test this idea, we find the optimal fit over a range of different strides (which varies the effective sample size) and increase the number of skipped knot points until the mean-square error starts to rise. We find that we can skip max[1, floor($2n_{\text{eff}}/3$)] knot points without sacrificing precision. The column labeled ‘‘optimal mse’’ in Table 2 indicates the optimal fit in which one knot point is created for every observation point, whereas the reduced degrees of freedom (‘‘reduced dof’’) column indicates a fit in which the number of knot points is reduced. In some cases the optimal mean-square error improves with fewer knot points. This means that, when handling large datasets, we can reduce the number of splines being used if the effective sample size is large, and we can ‘‘chunk’’ the data (split them into multiple independent pieces) when the effective sample size is small.

d. Interpolation condition

To estimate λ_T from (15), we estimate the mean-square value of a derivative of the process, $x_{\text{rms}}^{(T)}$ (see appendix C) and the effective sample size, n_{eff}. We argue that effective sample size should vary based on the relative size of the measurement errors to the speed of motion. For example, if the position errors are only 1 m, but a particle typically travels 10 m between measurements, then it is hardly justifiable to increase the tension so that the smoothing spline misses the observation

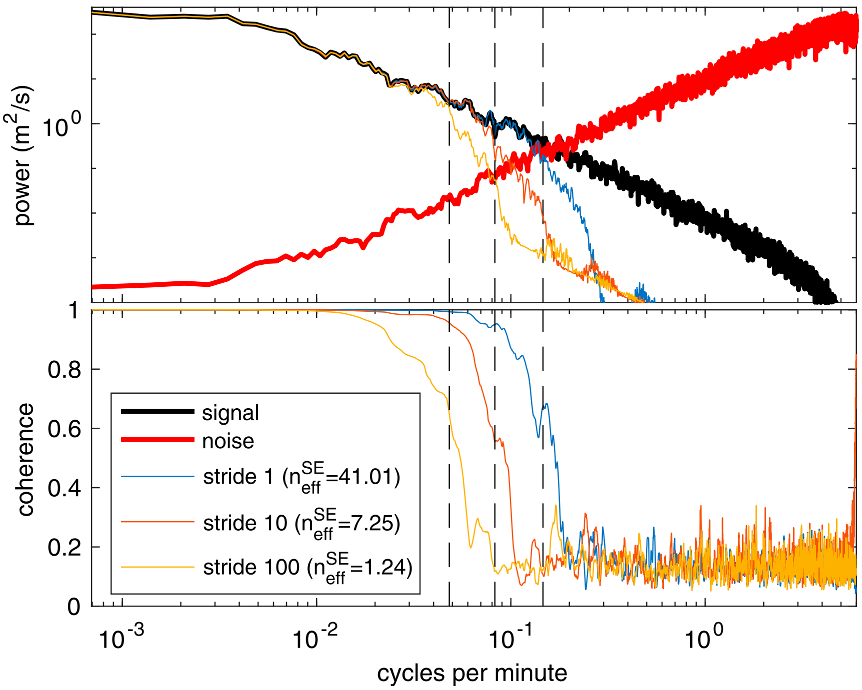

FIG. 4. (top) The uncontaminated velocity spectrum of the signal (black) and velocity spectrum of the noise (red). The observed signal is the sum of the two. The blue, red, and orange lines show the spectrum of the smoothing spline that is best fit to the observations with all, 1/10th, and 1/100th of the data, respectively. (bottom) The coherence between the smoothed signals and the true signal. The vertical dashed lines show the effective Nyquist computed using (23).

points by 1 m. There is not enough statistical evidence to suggest that the particle did not go right through the observation point. On the other hand, if the position errors are 1 m, but the particle typically travels 10 cm between measurements, nearby measurements provide more information about the particle's true position during that time, so our estimate of the particle's true position is closer to a mean of the nearby observations.

This idea can be made more rigorous by stating that a change in position, Δx, is statistically significant if it exceeds the position errors σ by some factor. Assuming the physical process has a characteristic velocity scale, u_{rms}, we use this concept to define Γ as

$$\Gamma \equiv \frac{\sigma}{u_{\mathrm{rms}}\Delta t}, \tag{24}$$

where Δt is the typical time between observations. This argument suggests that effective sample size should be proportional to Γ; that is,

$$n_{\mathrm{eff}}^{\Gamma} = \max(1, C \cdot \Gamma^{m}), \tag{25}$$

where C and m are unknown constants, and we prevent the effective sample size from dropping below 1. Intuitively this means as long as the particle does not move too far between observations, nearby observations help to estimate the true position of the particle.

To test the relationship between Γ and effective sample size, we compute the optimal smoothing spline for a range of values of Γ (created by subsampling the signal) for three different spectral slopes (ω^{-2}, ω^{-3}, and ω^{-4}). The value $n_{\mathrm{eff}}^{\mathrm{SE}}$ is computed from the optimal solution for 50 ensembles and shown in Fig. 5. The fits are remarkably good, but depend on the slope. Processes with shallower slopes (rougher trajectories) yield a smaller effective sample size for a given value of Γ.

Using the interpolation condition Γ to estimate effective sample size, we set $n_{\mathrm{eff}}^{\Gamma} = 14\Gamma^{0.71}$, the empirically determined best fit for slope ω^{-3}. For all spline fits we use

$$\lambda_T^{\mathrm{initial}} = \left(1 - \frac{1}{n_{\mathrm{eff}}^{\Gamma}}\right)\frac{1}{\left[x_{\mathrm{rms}}^{(T)}\right]^2} \tag{26}$$

as an initial estimate for the optimal smoothing parameter, where $x_{\mathrm{rms}}^{(T)}$ in (26) and u_{rms} in (24) are estimated using the method described in appendix C. The scaling law for $n_{\mathrm{eff}}^{\Gamma}$ can be estimated analytically. Let position observations be given by x_i, where

TABLE 2. Mean-square error (mse) and effective sample size for a range of strides and smoothing-spline methods.

Stride	n_{eff}	Optimal mse	Reduced dof	Blind initial	Expected mse
ω^{-2}					
1	8.6	11.5 m^2	0.1%	56.4%	7.4%
2	4.9	20.4 m^2	0.0%	36.3%	2.8%
4	2.9	34.2 m^2	0.1%	20.0%	1.7%
8	1.7	55.9 m^2	0.0%	5.6%	1.0%
16	1.2	81.8 m^2	0.0%	3.6%	0.5%
ω^{-3}					
1	12.5	7.64 m^2	−0.1%	38.6%	6.4%
2	7.1	13.4 m^2	−0.1%	20.4%	3.5%
4	4.1	23.5 m^2	−0.0%	9.8%	2.2%
8	2.3	41.8 m^2	0.0%	1.7%	1.2%
16	1.4	67.9 m^2	0.0%	9.6%	0.6%
ω^{-4}					
1	15.6	5.69 m^2	−0.1%	33.8%	7.9%
2	9.0	10.5 m^2	−0.1%	18.6%	5.1%
4	5.0	18.6 m^2	−0.0%	8.6%	2.4%
8	2.8	33.2 m^2	0.0%	3.2%	1.5%
16	1.6	57.6 m^2	0.0%	15.4%	0.8%

$$x_i = u_{\mathrm{rms}} i\Delta t + \varepsilon_i \quad \text{with} \quad \varepsilon_i = \mathcal{N}(0,\sigma). \tag{27}$$

If the effective sample size is $\langle n\rangle$, then the particle changes position by $\langle n\rangle u_{\mathrm{rms}}\Delta t$ between samples. Applying the two-sample z test, two positions will be considered different for $z > z_{\mathrm{min}}$, where

$$z = \frac{\langle n\rangle u_{\mathrm{rms}}\Delta t}{\sqrt{\dfrac{\sigma^2}{\langle n\rangle} + \dfrac{\sigma^2}{\langle n\rangle}}} \Rightarrow \langle n\rangle = \left(\frac{z\sigma\sqrt{2}}{u_{\mathrm{rms}}\Delta t}\right)^{2/3}. \tag{28}$$

The power law in (28) is close to the empirically derived power laws shown in Fig. 5. This suggests that the coefficient C in (25) can be related to z, a measure of statistical significance.

e. Optimal fits

Table 2 summarizes the key results of this section by applying a smoothing spline ($S = 3$) to 200 ensembles with three different slopes (ω^{-2}, ω^{-3}, and ω^{-4}) and five different strides. When the algorithm uses true values, uncontaminated by noise, we consider the process to be "unblinded," in contrast to "blind" methods, in which the algorithm only uses noisy data. The second and third columns show the effective sample size and average mean-square error when the smoothing spline is applied using the true values (i.e., unblinded) to minimize the mean-square error—this is the lower bound. The fourth column shows average increase in mean-square error when reducing the number of spline coefficients as documented in section 4c. There is almost no change in mean-square error, and therefore, all subsequent methods (whether blind or unblind) use this technique. The fifth column uses (26) from section 4d to provide a (blind) initial guess of the tension parameter. The results are mixed—a typical increase in mean-square error is 30%–50% when the effective sample size is large. While this seems large, this is a small fraction of the total noise variance; for example, an optimal mean-square error of 6 m^2 increases to 8 m^2 when the total variance is 100 m^2. Nearly optimal fits can be found using (19), as shown in the last column of the table.

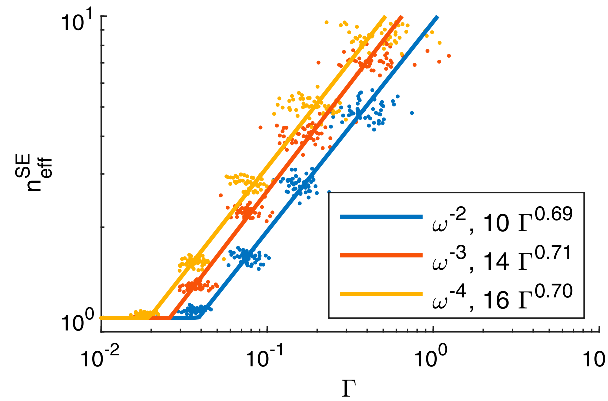

FIG. 5. Effective sample size from the standard error vs Γ.

f. Numerical implementation

The numerical implementation of the methods in this section are available in the SmoothingSpline class, which subclasses BSpline. This class is initialized with three required parameters: a set of data points (t_i, x_i) and an error distribution.

5. Bivariate smoothing splines and stationarity

Up to this point we have considered univariate data, (t_i, x_i), but GPS position data are fundamentally bivariate. The term "bivariate" in the context of splines is often used to denote splines defined on two independent variables—however, in this context we define bivariate to mean two dependent variables (e.g., x and y) and one independent variable (e.g., t).

The trivial approach to working with bivariate data is to treat each direction independently—that is, minimize λ_T^x and λ_T^y independent of each other. However, the underlying physical process is often isotropic. In the context of the maximum-likelihood formulation of smoothing splines in (18), this means we expect $x_{\mathrm{rms}}^{(T)}$ (the rms value of the tensioned variable) to be the same in all directions (invariant under rotation). This however does *not* mean that λ_x should necessarily equal λ_y. To be explicit, if

$$\lambda_T^x = \left(1 - \frac{1}{n_{\text{eff}}^x}\right)\frac{1}{\left[x_{\text{rms}}^{(T)}\right]^2}, \quad \lambda_T^y = \left(1 - \frac{1}{n_{\text{eff}}^y}\right)\frac{1}{\left[y_{\text{rms}}^{(T)}\right]^2}, \tag{29}$$

then, even if $x_{\text{rms}}^{(T)} = y_{\text{rms}}^{(T)}$, the effective sample sizes n_{eff}^x and n_{eff}^y may differ if there is any mean velocity because, as shown in section 4d, effective sample size depends on velocity.

Therefore, to assume isotropy in λ_T and use a bivariate smoothing spline, the mean velocity from the underlying process must be removed. What qualifies as mean and fluctuation rarely has a clear answer, but a reasonable option is letting a polynomial of degree $T + 1$ define the mean. This has the added benefit of removing a constant nonzero tension value, which as shown in section 3b, changes the problem formulation.

It is stationarity, not isotropy, that requires removing the mean velocity. The effective sample size is shown to be dependent on rms velocity, so if velocity varies in time, then the optimal effective sample size varies as well. This means not only do smoothing splines require stationarity in the tensioned variable $x^{(T)}$ as shown in section 3b, but they also require stationarity in the velocity $x^{(1)}$ to be effective. This last requirement can be solved by either removing the mean (as suggested here), or segmenting observations into locally stationary chunks.

a. *Assessing errors*

Removing the mean or some other low-passed version of the data means the total smoothing matrix is a combination of the low-passed and high-passed smoothing matrices. Once this matrix is computed, it can be used to compute the standard errors.

We first create a low pass filter to capture the *mean* component of the flow using a simple polynomial fit $\overline{\mathbf{x}} = \overline{\mathbf{S}}\mathbf{x}$ and then define the residual as our stationary part, $\mathbf{x}' \equiv \mathbf{x} - \overline{\mathbf{x}}$. We now compute the smoothing spline as usual on the residual, $\mathbf{x}'_\lambda = \mathbf{S}_\lambda \mathbf{x}'$. So the total, smoothed path is

$$\hat{\mathbf{x}} = \overline{\mathbf{x}} + \mathbf{x}'_\lambda = \overline{\mathbf{S}}\mathbf{x} + \mathbf{S}_\lambda(\mathbf{x} - \overline{\mathbf{S}}\mathbf{x}) = (\overline{\mathbf{S}} + \mathbf{S}_\lambda - \mathbf{S}_\lambda\overline{\mathbf{S}})\mathbf{x} \equiv \mathbf{S}_T\mathbf{x}. \tag{30}$$

From this we can compute the covariance matrix and the standard error.

b. *Numerical implementation*

The BivariateSmoothingSpline class is initialized with data (t_i, x_i, y_i) and a distribution. For a spline of degree $S = T$, a spline of degree $S + 1$ is used to remove the mean in each direction. With a Gaussian distribution this is simply a least squares polynomial fit. By assumption, the residual data are stationary and isotropic, so the tension parameter λ_T is applied equally in each direction. Minimization is performed on the sum of the expected mean-square errors in each direction.

6. GPS dataset

The primary dataset considered here is nine surface drifters deployed in the Sargasso Sea in the summer of 2011 (Shcherbina et al. 2015). In the past, such drifters used the Argos positioning system, which has significantly poorer temporal coverage and position accuracy (Elipot et al. 2016), but recently most surface drifters have employed GPS receivers and transmitted their data back through Argos or Iridium satellites.

The GPS receiver sits on the surface drifter and collects position data, but because of atmospheric conditions or ocean waves, the receivers are sometimes unable to obtain a position, or when they do, it is highly inaccurate. Despite nominal accuracies of a few meters, it is often the case that some positions are off by more than 1000 m, as can be seen in Fig. 8. Applying a smoothing-spline fit using the method in section 3 produces an extremely poor fit, with clear overshoots to bad data points.

GPS error distribution

We characterize the GPS errors by considering data from a motionless GPS receiver allowed to run for 12 h. The GPS receiver used in this test is not the same as the one used for the drifters (because it was no longer available) but should produce errors similar enough for this analysis.

The position recorded by the motionless GPS are assumed to have isotropic errors with mean zero, which means the positions themselves are the errors. The PDF of the combined x and y position errors are shown in Fig. 6.

The error distribution is first fit to a zero-mean Gaussian PDF (10). The maximum-likelihood fit is found by computing the standard deviation of the sample, which is found to be $\sigma \approx 10$ m and shown as the gray line in Fig. 6. However, it is clear the error distribution shows much longer tails than the Gaussian PDF.

The Student's t distribution is a generalization of the Gaussian that produces longer tails and is defined as

$$p_s(\varepsilon|\nu, \sigma_s^2) = \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma_s\sqrt{\nu\pi}\,\Gamma\left(\frac{\nu}{2}\right)}\left(1 + \frac{\varepsilon^2}{\sigma_s^2\nu}\right)^{-(\nu+1)/2}, \tag{31}$$

where the σ_s parameter scales the distribution width and the ν parameter sets the number of degrees of freedom

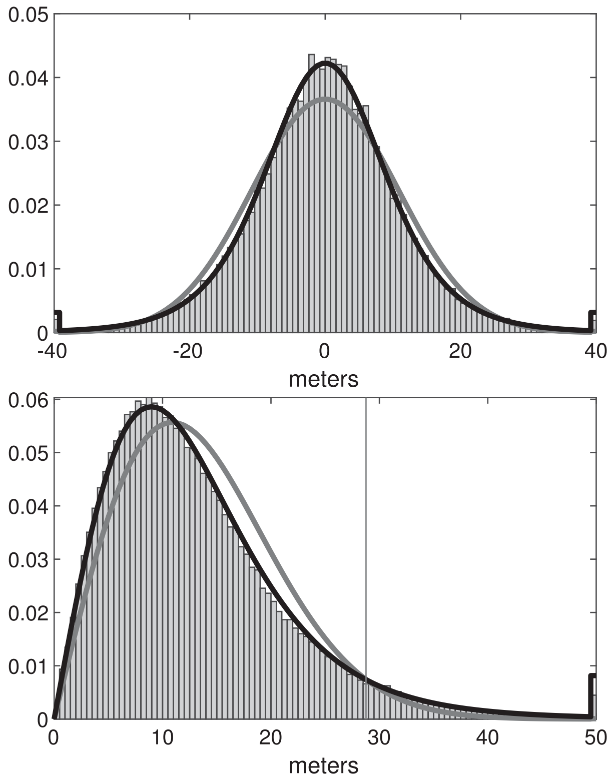

FIG. 6. (top) The position error distribution of the motionless GPS. The gray or black curves are the best-fit Gaussian or t distribution, respectively. (bottom) The distance error distribution with the corresponding expected distributions from the Gaussian and t distributions. The vertical line in the bottom panel shows the 95% error of the t distribution.

[as $\nu \to \infty$, (31) becomes a Gaussian]. The variance is $\sigma^2 = \sigma_s^2[\nu/(\nu-2)]$ and only exists for $\nu > 2$. Minimizing the Anderson–Darling test to find the best-fit t distribution to the data, we find parameters $\sigma_s \approx 8.5$ m and $\nu \approx 4.5$ shown as the black line in Fig. 6. Different choices in GPS receivers and using the Kolmogorov–Smirnoff test results in very similar parameters; that is, $\sigma_s \approx 8$–10 m and $\nu \approx 4$–6.

The *position* error distributions imply a combined *distance* error distribution by computing $\varepsilon_d = (\varepsilon_x^2 + \varepsilon_y^2)^{1/2}$ and is shown in the lower panel of Fig. 6. With Gaussian noise this results in a Rayleigh distribution,

$$p_r(\varepsilon_d|\sigma_g) = \frac{\varepsilon_d}{\sigma_g^2}\exp\left(-\frac{1}{2}\frac{\varepsilon_d^2}{\sigma_g^2}\right), \quad (32)$$

as shown by the gray line. The distance error distribution from t-distributed noise is computed numerically and is shown by the black line. Around 95% of distance errors are within 30 m.

Figure 7 shows the autocorrelation function of the GPS position errors and the 99% confidence intervals.

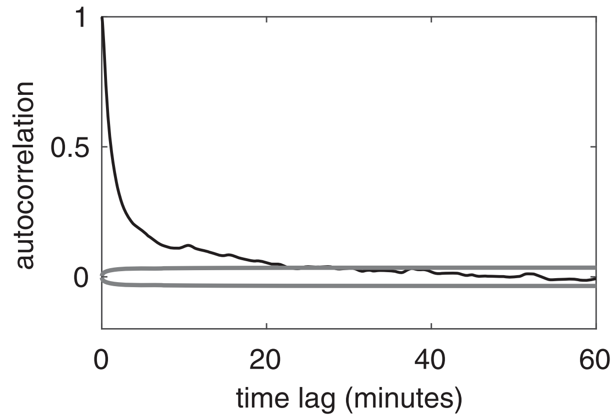

FIG. 7. The autocorrelation function of the GPS positioning error, with 99% confidence intervals shown in gray. The correlation at a drifter sampling period of 30 min is indistinguishable from zero.

We find a rough empirical fit to be $\rho(\tau) = \exp[\max(-\tau/t_0, -\tau/t_1 - 1.35)]$, where $t_0 = 100$ s and $t_1 = 760$ s, which reflects an initially rapid falloff in correlation, followed by a slower decline. The smallest sampling interval of the GPS drifters in question is 30 min and the correlation indistinguishable from zero according to Fig. 7. It is therefore safe to assume the errors are uncorrelated for our real-data example. Although the drifter sampling rate allows us to avoid further discussion of the autocorrelation function of GPS errors, accounting for autocorrelation is a relatively easy extension (and is implemented in the code).

The smoothing-spline algorithms described in section 3 are modified to use the t distribution as described in appendix B. Table 3 shows that the conclusions reached for Gaussian data in section 3 still apply with t-distributed data.

7. Minimization with outliers

The goal here is to find a smooth solution in the presence of outliers—points that do not appear to be of the known error distribution for the GPS receiver shown in section 6. These points are obviously problematic as can be seen in Fig. 8, where individual data points jump hundreds of meters and even several kilometers away from its neighbors. Errors of this size are inconsistent with the noise analysis of the preceding section, so the goal here is to find a model path $x(t)$ robust to this uncharacterized noise. What makes outliers "obvious" to the eye is they appear as unexpectedly large motions, inconsistent with the other motion for that path. The smoothing-spline formulation is therefore useful, as it assumes the motion at some order (e.g., acceleration) is Gaussian, as shown in section 3b. In the nine drifters we

TABLE 3. As in Table 2, but with noise following a t distribution.

Stride	n_{eff}	Optimal mse	Reduced dof	Blind initial	Expected mse
ω^{-2}					
1	8.2	11.8 m^2	0.3%	66.7%	7.7%
2	4.7	20.9 m^2	0.3%	47.3%	6.6%
4	2.8	38.0 m^2	0.1%	24.2%	4.4%
8	1.6	66.3 m^2	0.0%	8.2%	9.3%
16	1.2	101. m^2	0.0%	8.1%	3.7%
ω^{-3}					
1	12.1	7.51 m^2	−0.1%	36.2%	8.8%
2	6.8	13.4 m^2	−0.1%	22.8%	7.0%
4	3.9	26.0 m^2	−0.0%	11.5%	3.8%
8	2.2	47.5 m^2	0.0%	2.2%	3.2%
16	1.3	82.5 m^2	0.0%	12.6%	8.5%
ω^{-4}					
1	14.9	6.01 m^2	−0.2%	35.3%	9.0%
2	8.6	10.5 m^2	−0.2%	24.8%	7.0%
4	4.8	19.1 m^2	−0.1%	7.8%	4.6%
8	2.7	36.4 m^2	0.0%	3.2%	2.7%
16	1.6	69.1 m^2	0.0%	18.9%	11.5%

are analyzing here, one drifter shows no obvious outliers, suggesting the issue may be related to how the antennas are configured. This particular drifter serves as a useful point of comparison.

Minimizing with the expected mean-square error (19) produces a fit so poor it is not worth showing. Because outliers add enormous amounts of variance, the expected mean-square error vastly underestimates the spline tension—essentially chasing every outlier shown in Fig. 8. Because some of the noise is uncharacterized, this suggests using a method such as cross validation might be effective. The orange line in Fig. 8 uses a smoothing-spline fit, assuming Student's t distributed errors, but minimized with cross validation. This fit performs relatively well, but, when compared with the drifter 7, it is clear it still chases some outliers. The goal in this section is to develop a method robust to outliers in cases where we know something about the noise.

The basic problem formulation is as follows: we define a new "robust distribution," p_{robust}, that includes the known noise distribution, p_{noise}, plus an unknown (or assumed) form of an outlier distribution, p_{outlier},

$$p_{\mathrm{robust}}(\varepsilon) = (1-\alpha)p_{\mathrm{noise}}(\varepsilon) + \alpha p_{\mathrm{outlier}}(\varepsilon). \qquad (33)$$

We consider a t distribution for p_{noise} with parameters found from the GPS errors in section 6. The distribution of p_{outlier} is also set to be a t distribution, but with $\nu = 3$ and $\sigma = 50\sigma_{\mathrm{gps}}$, which roughly matches the total variance of the observed outliers. In our tests we varied α from 0 up to 0.25, approximately the range of observed outliers from the drifter datasets.

Throughout our attempts to smooth the noisy GPS data we tried many different approaches to modifying smoothing splines for robustness to outliers, but ultimately found enormous gains are made by simply discarding outliers while minimizing the expected mean-square error (19). The results of this approach are shown in section 7a, and we document our method to reliably estimate the outlier distribution in section 7b.

a. Robust minimization

The challenge with outliers is we do not know their distribution, so minimizing the expected mean-square

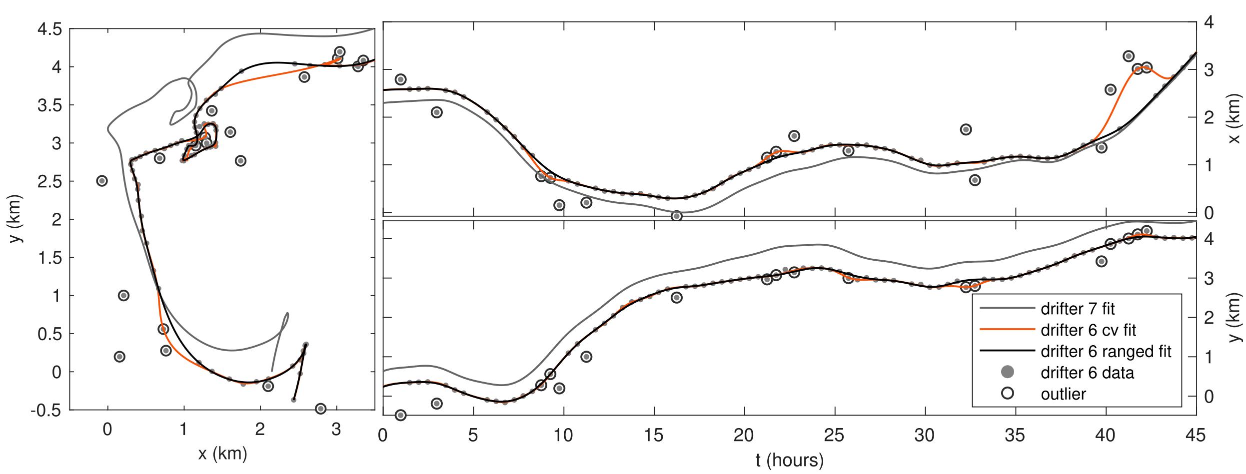

FIG. 8. GPS position data for a 40-h window from drifter 6. The points are the recorded positions, and the black line is the optimal fit using the ranged expected mean-square error. Data points with less than 0.01% chance of occurring are highlighted and are deemed outliers. The light gray line is the optimal smoothing-spline fit for drifter 7, which has no apparent outliers and was released a few hundred meters from drifter 6. The orange line is the smoothing-spline fit assuming t-distributed errors but using cross validation to minimize λ_T.

error using (19) with the expected variance from the robust distribution defined in (33) does not work. Outliers add extra variance and therefore cause the spline to be undertensioned (λ_T too small). Our method excludes the outliers from the calculation of (19), where outliers are defined as points unlikely to arise with the known noise distribution. The *ranged expected mean-square error* replaces σ^2 with

$$\sigma_\beta^2 = \int_{\mathrm{cdf}^{-1}(\beta/2)}^{\mathrm{cdf}^{-1}[1-(\beta/2)]} z^2 p_{\mathrm{noise}}(z)\, dz \tag{34}$$

and discards all rows (and columns) of $\mathbf{S}_\lambda$ where $(\mathbf{S}_\lambda - \mathbf{I})\mathbf{x} < \mathrm{cdf}^{-1}(\beta/2)$ or $(\mathbf{S}_\lambda - \mathbf{I})\mathbf{x} > \mathrm{cdf}^{-1}[1-(\beta/2)]$.

To test this approach we generated data as before but allowed a certain percentage of outliers α to be generated with an outlier distribution following (33). We considered five values of β (1/50, 1/100, 1/200, 1/400, and 1/800) as well as $\beta = 0$, which is just (19). Testing across a number of ensembles with outlier ratios $\alpha = 0.0$, 0.05, 0.10, and 0.25 we found that $\beta = 1/100$ is overall the best choice.

b. Full-tension solution and outlier distribution

The *full-tension* solution is defined as the maximum allowable value of λ given the known noise distribution. That is, the spline fit is pulled away from the observations so that the distribution of observed errors $x_i - x(t_i)$ matches the expected distribution $p_{\mathrm{noise}}(\varepsilon)$. In cases where the effective sample size n_{eff} is large, the full-tension solution approximately matches the optimal (minimal mean-square error) solution. In cases in which the effective sample size is small, the full-tension solution is more akin to a low-pass solution [as increasing λ is equivalent to decreasing $x_{\mathrm{rms}}^{(T)}$].

In the simplest case where there are no outliers, the full-tension solution can be found by requiring the sample variance match the variance of $p_{\mathrm{noise}}(\varepsilon)$. When outliers are present, a more robust method of estimation is required. After some experimentation, we found the most reliable method of achieving full tension is to minimize the Anderson–Darling test of $p_{\mathrm{noise}}(\varepsilon)$ on the interquartile range of observed errors. This method can be used to estimate the outlier distribution and further refine both the full-tension solution and the range over which the expected mean-square error is computed.

The outlier distribution is estimated as follows. We first assume the outlier distribution follows a t distribution with $\nu = 3$ and $\alpha < 0.5$. If the spline is in full tension, then the observed total variance can be used to find σ_o for the outlier distribution. From (33),

$$\mathrm{var}_{\mathrm{total}} = (1-\alpha)\,\mathrm{var}_{\mathrm{noise}} + \alpha 3\sigma_o^2, \tag{35}$$

which, given some α, can be solved for σ_o. Our method uses 100 values of α logarithmically spaced from 0.01 to 0.5 and chooses the value that minimizes the Anderson–Darling test. With an estimate for $p_{\mathrm{robust}}(\varepsilon)$, the full-tension solution can be refined by minimizing the Anderson–Darling test of $p_{\mathrm{robust}}(\varepsilon)$ on the interquartile range of observed errors. This iterative process converges very quickly to a good estimate for the outlier distribution and the full-tension solution.

c. Extension to bivariate data

The strategies in this section are relatively easily extended to bivariate data. All error distributions are assumed isotropic, and the outlier distribution can be estimated by including the errors from both independent directions. The ranged expected mean-square error calculation defined in section 7a uses the *distance* of the error for its cutoff to remain invariant under rotation.

Application of this method to one of the GPS drifters (drifter 6) is shown in Fig. 8. Although it is impossible to know exactly how well the spline fit performed, comparison with drifter 7 (with no apparent outliers) suggests our method successfully avoids chasing outliers.

d. Numerical implementation

The GPSSmoothingSpline inherits from the BivariateSmoothingSpline class and assumes errors follow a t distribution found in section 6. The class projects latitude and longitude using a transverse Mercator projection with the central meridian set to the center of the dataset.

8. Discussion

The methods discussed in this paper are related to other methods used to smooth and interpolate drifter trajectories.

Yaremchuk and Coelho (2015) formulate a cost function, their (9), based on PDFs of the drifter accelerations and the GPS errors. Setting their $\mu = 1$ (they choose $\mu = 0.9$) this is equivalent to the special case of (18) when $S = T = 2$, where they have implicitly chosen λ_T by assuming an infinite effective sample size, n^{eff}. Their method for isolating outliers is nearly equivalent to the iteratively reweighted least squares method detailed in appendix B using a weight function similar to Tukey's biweight, (B6).

Elipot et al. (2016) apply their method to the Argos-tracked surface drifters, which are significantly noisier positions than GPS errors but also follow a t distribution. They assume a linear model for positions, equivalent to assuming $S = T = 2$ with $\lambda_T \to \infty$. In the numerical implementation of this paper, this special case is implemented in the ConstrainedSpline class. The time-dependent weight function used in Elipot et al. (2016) requires manually specifying a weight for each point used, and this method is therefore somewhat different than the approach taken here.

Another technique used for smoothing and interpolating drifter positions is kriging (Hansen and Poulain 1996); however, its relationship to smoothing splines is less clear. In response to a study empirically comparing kriging with smoothing splines (Laslett 1994), Handcock et al. (1994) point out that kriging and smoothing splines are just two specific parameter choices of a more general class of splines defined by their covariance functions. In the context of the maximum-likelihood equation for smoothing splines [(18)], this generalization could be modeled by including a covariance structure on the physical process.

Overall, the method of this paper (in a loose sense) generalizes a number of existing approaches for interpolation, especially in terms of flexibly allowing different levels of smoothness and tension, and in terms of application to non-Gaussian noise structures.

9. Conclusions

The method in this paper solves our problem of finding smoothed, interpolated positions from a noisy GPS drifter dataset with outliers. In more general terms, for signals with second-order structure similar to a Matérn process we found that

1) the spline degree S should be set to a value higher than the high-frequency spectral slope of the process (section 2) and
2) the optimal tension parameter can be estimated a priori (section 4).

For GPS data, there appear to be three key steps for using smoothing splines:

1) using a t distribution for the noise (section 6),
2) removing the mean velocity to make the bivariate data stationary (section 5), and
3) using the ranged expected mean-square error for robustness to outliers (section 7).

Acknowledgments. Thanks are given to Miles Sundermeyer, whose drifters were used in this analysis. The work of J. J. Early was funded by ONR through the Scalable Lateral Mixing and Coherent Turbulence Departmental Research Initiative (LatMix) and National Science Foundation Award 1658564. The work of A. M. Sykulski was funded by the Engineering and Physical Sciences Research Council (Grant EP/R01860X/1).

APPENDIX A

Numerical Implementation

The B-splines are generated using the algorithm described in De Boor (1978) with knot points determined by (7) and (8). The matrix X with components X^i_m denotes the mth B-spline at time t_i. The column vector $\boldsymbol{\xi}^m$ represents the coefficients of the splines such that positions at time t_i are given by $\hat{x}^i$, where $\hat{x}^i = X^i_m \boldsymbol{\xi}^m$.

The smoothing spline condition in (16) can be augmented to include a nonzero mean tension μ_u,

$$\phi = \frac{1}{N}\sum_{i=1}^{N}\left[\frac{x_i - x(t_i)}{\sigma_i}\right]^2 + \frac{1}{Q}\sum_{q=1}^{Q}\left[\frac{u(t_q) - \mu_u}{\sigma_u}\right]^2, \qquad \text{(A1)}$$

where we have taken $T = 1$ for this calculation. The discretized penalty function is

$$\phi = (\mathbf{x} - \mathsf{X}\boldsymbol{\xi})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x} - \mathsf{X}\boldsymbol{\xi}) + \lambda_1(\mathsf{V}\boldsymbol{\xi} - \mu)^{\mathrm{T}}(\mathsf{V}\boldsymbol{\xi} - \mu), \qquad \text{(A2)}$$

where $\boldsymbol{\Sigma}$ denotes the covariance matrix describing the measurement errors and we absorbed several constants into λ_1. The matrix V^q_m is the spline velocity matrix such that $v^q = \mathsf{V}^q_m \xi^m$ is the model velocity at time t_q. To find the coefficients that minimize this function, we take the derivative with respect to $\boldsymbol{\xi}$, set it to zero, and solve for $\boldsymbol{\xi}$,

$$\boldsymbol{\xi} = (\mathsf{X}^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\mathsf{X} + \lambda_1\mathsf{V}^{\mathrm{T}}\mathsf{V})^{-1}(\mathsf{X}^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\mathbf{x} + \mu\lambda_1\mathsf{V}^{\mathrm{T}}\boldsymbol{\iota}), \qquad \text{(A3)}$$

where $\boldsymbol{\iota}$ is a vector of 1s. The operation $\mathsf{V}^{\mathrm{T}}\boldsymbol{\iota}$ essentially integrates the m splines and results in a column vector with the integrated values.

We define the smoothing matrix as the linear operator that takes observations $\mathbf{x}$ to their smoothed values $\hat{\mathbf{x}}$, $\hat{\mathbf{x}} = \mathsf{S}_\lambda\mathbf{x}$. From this definition and (A3),

$$\mathsf{S}_\lambda \equiv \mathsf{X}(\mathsf{X}^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\mathsf{X} + \lambda_1\mathsf{V}^{\mathrm{T}}\mathsf{V})^{-1}\mathsf{X}^{\mathrm{T}}\boldsymbol{\Sigma}^{-1} \qquad \text{(A4)}$$

when $\mu = 0$.

APPENDIX B

Iteratively Reweighted Least Squares

Using the t distribution is challenging because it does not result in a linear solution for the coefficients as in (A3). One solution is to use a search algorithm to directly look for maximum values. Alternatively, one can use iteratively reweighted least squares (IRLS).

The idea with IRLS is to reweight the coefficients of the Gaussian, σ_g in (10), so that the resulting distribution looks like the desired distribution, for example, (31). With the recollection that $\varepsilon_i \equiv x_i - x(t_i, \boldsymbol{\xi})$, the minimization condition $dp_g/d\xi = 0$ implies that

$$\frac{\varepsilon_i}{\sigma_g^2}\frac{\partial x(t_i, \mathbf{x})}{\partial \boldsymbol{\xi}} = 0, \tag{B1}$$

for the Gaussian distribution, whereas for the t distribution this implies

$$\frac{\varepsilon_i}{\sigma_s^2}\frac{\nu+1}{\nu}\left(1+\frac{\varepsilon_i^2}{\nu\sigma_s^2}\right)^{-1}\frac{\partial x(t_i, \mathbf{x})}{\partial \boldsymbol{\xi}} = 0. \tag{B2}$$

This means that one can set

$$\sigma_g^2 = \sigma_s^2\frac{\nu}{\nu+1}\left(1+\frac{\varepsilon_i^2}{\nu\sigma_s^2}\right) \tag{B3}$$

to get a matching distribution. Of course, this is only true if ε_i is already known, which initially it is not. So the method becomes iterative—one starts with ε_i determined from the Gaussian fit and then determines a new ε_i after reweighting σ_g. This method iterates until σ_g stops changing. We can rewrite (B3) as a function of ε_i,

$$w_s(\varepsilon_i) = \sigma_s^2\frac{\nu + (\varepsilon_i^2/\sigma_s^2)}{\nu+1}. \tag{B4}$$

From (B4) it is clear that if $\varepsilon_i < \sigma_s$ then it is reweighted to a smaller value, making the observation point more strongly weighted. On the other hand, if $\varepsilon_i > \sigma_s$, then its relative weighting decreases, and it is treated more as an outlier.

In more general terms, the weight function $w(z)$ for a PDF $p(z)$ is found by setting $-\partial_z \log p(z)$ equal to $-\partial_z \log p_g(z)$ of a Gaussian PDF, where $w(z)$ replaces σ_g^2, and then solving for $w(z)$. The result is

$$\frac{z}{w(z)} = -\frac{\partial_z p}{p} \Rightarrow w(z) = -z\frac{p}{\partial_z p}. \tag{B5}$$

The same strategy could be used to reshape the PDF of a Gaussian to match the desired distribution, but here we simply match the minimization conditions of the PDFs.

As a point of reference, Tukey's biweight is given by

$$\psi(z) = \begin{cases} \dfrac{z}{\sigma_{\mathrm{tb}}^2}\left(1-\dfrac{z^2}{c^2\sigma_{\mathrm{tb}}^2}\right)^2 & |z| < c\sigma_{\mathrm{tb}} \\ 0 & \text{otherwise} \end{cases}, \tag{B6}$$

which as a weight function is

$$w_{\mathrm{tb}}(\varepsilon_i) = \frac{z}{\psi(z)}. \tag{B7}$$

In a practical sense, $\boldsymbol{\Sigma}^{-1}$ in (A4) is replaced with the diagonal matrix $\mathbf{W} \equiv \mathrm{diag}[1/w(\varepsilon_i)]$ populated with the reweighted values for each observation such that

$$\mathbf{S}_\lambda \equiv \mathbf{X}(\mathbf{X}^{\mathrm{T}}\mathbf{W}\mathbf{X} + \lambda_1\mathbf{V}^{\mathrm{T}}\mathbf{V})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{W}. \tag{B8}$$

This operator is used to compute the standard error from the variances, $\mathbf{S}_\lambda\boldsymbol{\Sigma}$, where the variance is assumed to be $\sigma_s^2\nu/(\nu-2)$ for each observation when using a t distribution.

The smoothing-spline solution *does* depend on the initial value of $w(\varepsilon_i)$ used in the IRLS method. However, we find that for uniform initial weightings (e.g., all values start with the square root of the variance), the differences are not statistically significant from other initial values.

APPENDIX C

Estimating the Variance of the Signal

Our method requires good estimates of the root-mean-square velocity u_{rms} of the signal, to determine the effective sample size and variance of the tensioned derivative. Our approach is to compute the power spectrum of the signal at the derivative of interest, and sum the variance that is statistically significantly greater than the expected variance of the noise.

Given a process observed with values x_n at times $t_n = n\Delta$ where $n = 1, \ldots, N$, we estimate the mean of its mth derivative by performing a least squares fit to the polynomial $\bar{x}_n \equiv p_m t_n^m + p_{m-1}t_n^{m-1} + \ldots + p_0$. The *detrended* time series is defined as $\tilde{x}_n \equiv x_n - \bar{x}_n$. The power spectrum of this time series is

$$S_{\mathrm{signal}}(f_k) = \frac{\Delta}{N}\left|\sum_{n=0}^{N-1} x_n \exp(-2\pi i f_k t_n)\right|^2, \tag{C1}$$

where the frequencies f_k are given by $f_k = k/(N\Delta)$. By Plancherel's theorem,

$$\sum_{k=0}^{N-1} S(f_k)\frac{1}{N\Delta} = \frac{1}{N\Delta}\sum_{i=0}^{N-1} x_i^2\Delta. \tag{C2}$$

The power spectrum of the mth derivative of the process is computed as

$$S_{\mathrm{signal}}^{(m)}(f_k) = (2\pi f_k)^{2m} S(f_k). \tag{C3}$$

It is important to detrend the signal prior to computing the derivative because, by assumption, the signal is periodic and has no secular trend.

The noise, ε_i, has total variance $\sigma^2 = (1/N)\sum_{i=1}^{N}\varepsilon_i^2$. Because the noise is assumed to be uncorrelated, the variance distributes evenly across all frequencies. The spectrum of the noise is therefore

$$S_{\text{noise}}(f_k) = \sigma^2\Delta, \tag{C4}$$

which immediately satisfies Plancherel's theorem (C2). The mth derivative of the noise has power spectrum

$$S_{\text{noise}}^{(m)}(f_k) = \sigma^2\Delta(2\pi f_k)^{2m}. \tag{C5}$$

The technique used here sums the variance of the signal for a given frequency if it exceeds the expected variance of the noise at the frequency by some threshold. The estimate of power at each frequency follows a χ^2 distribution with 2 degrees of freedom, so we choose the threshold based on the 95th percentile of the expected distribution. And thus,

$$x_{\text{std}}^{(m)} = \sum_{k=0}^{N-1} S_{\text{signal}}^{(m)}(f_k)[S_{\text{signal}}^{(m)}(f_k) > qS_{\text{noise}}^{(m)}(f_k)]\frac{1}{N\Delta}, \tag{C6}$$

where $q \approx 20$ for the 95th-percentile confidence.

REFERENCES

Bracco, A., J. H. LaCasce, C. Pasquero, and A. Provenzale, 2000: The velocity distribution of barotropic turbulence. *Phys. Fluids*, **12**, 2478, https://doi.org/10.1063/1.1288517.

Cantoni, E., and T. Hastie, 2002: Degrees-of-freedom tests for smoothing splines. *Biometrika*, **89**, 251–263, https://doi.org/10.1093/biomet/89.2.251.

Craven, P., and G. Wahba, 1978: Smoothing noisy data with spline functions. *Numer. Math.*, **31**, 377–403, https://doi.org/10.1007/BF01404567.

De Boor, C., 1978: *A Practical Guide to Splines*. Vol. 27. Springer-Verlag, 348 pp.

Elipot, S., R. Lumpkin, R. Perez, J. J. Early, and A. M. Sykulski, 2016: A global surface drifter data set at hourly resolution. *J. Geophys. Res. Oceans*, **121**, 2937–2966, https://doi.org/10.1002/2016JC011716.

Green, P. J., and B. W. Silverman, 1993: *Nonparametric Regression and Generalized Linear Models: A Roughness Penalty Approach*. Chapman and Hall, 184 pp.

Handcock, M. S., K. Meier, and D. Nychka, 1994: Kriging and splines: An empirical comparison of their predictive performance in some applications: Comment. *J. Amer. Stat. Assoc.*, **89**, 401–403, https://doi.org/10.2307/2290838.

Hansen, D. V., and P.-M. Poulain, 1996: Quality control and interpolations of WOCE-TOGA drifter data. *J. Atmos. Oceanic Technol.*, **13**, 900–909, https://doi.org/10.1175/1520-0426(1996)013<0900:QCAIOW>2.0.CO;2.

Laslett, G. M., 1994: Kriging and splines: An empirical comparison of their predictive performance in some applications. *J. Amer. Stat. Assoc.*, **89**, 391–400, https://doi.org/10.1080/01621459.1994.10476759.

Lee, T. C. M., 2003: Smoothing parameter selection for smoothing splines: A simulation study. *Comput. Stat. Data Anal.*, **42**, 139–148, https://doi.org/10.1016/S0167-9473(02)00159-7.

Lilly, J. M., 2019: A data analysis package for MATLAB, version 1.6.6. Jekyll, http://www.jmlilly.net/jmlsoft.html.

——, A. M. Sykulski, J. J. Early, and S. C. Olhede, 2017: Fractional Brownian motion, the Matérn process, and stochastic modeling of turbulent dispersion. *Nonlinear Processes Geophys.*, **24**, 481–514, https://doi.org/10.5194/npg-24-481-2017.

Nychka, D., 2000: Spatial process estimates as smoothers. *Smoothing and Regression: Approaches, Computation and Application*, M. G. Schimek, Ed., John Wiley and Sons, 393–424.

Press, W. H., S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, 1992: *Numerical Recipes in C: The Art of Scientific Computing*. 2nd ed. Cambridge University Press, 994 pp.

Reinsch, C. H., 1967: Smoothing by spline functions. *Numer. Math.*, **10**, 177–183, https://doi.org/10.1007/BF02162161.

Shcherbina, A. Y., and Coauthors, 2015: The LatMix summer campaign: Submesoscale stirring in the upper ocean. *Bull. Amer. Meteor. Soc.*, **96**, 1257–1279, https://doi.org/10.1175/BAMS-D-14-00015.1.

Sykulski, A. M., S. C. Olhede, J. M. Lilly, and E. Danioux, 2016: Lagrangian time series models for ocean surface drifter trajectories. *J. Roy. Stat. Soc.*, **65**, 29–50, https://doi.org/10.1111/rssc.12112.

Teanby, N. A., 2007: Constrained smoothing of noisy data using splines in tension. *Math. Geol.*, **39**, 419–434, https://doi.org/10.1007/s11004-007-9104-x.

Wahba, G., 1978: Improper priors, spline smoothing and the problem of guarding against model errors in regression. *J. Roy. Stat. Soc.*, **40B**, 364–372, https://doi.org/10.1111/J.2517-6161.1978.TB01050.X.

Whittaker, E. T., 1923: On a new method of graduation. *Proc. Edinburgh Math. Soc.*, **41**, 63–75, https://doi.org/10.1017/S0013091500077853.

Wide Area Augmentation System T and E Team, 2016: Global positioning system (GPS) standard positioning service (SPS) performance analysis report. William J. Hughes Technical Center Tech. Rep. 92, 147 pp., https://www.nstb.tc.faa.gov/reports/PAN92_0116.pdf.

Yaremchuk, M., and E. F. Coelho, 2015: Filtering drifter trajectories sampled at submesoscale resolution. *IEEE J. Oceanic Eng.*, **40**, 497–505, https://doi.org/10.1109/JOE.2014.2353472.