

1 Introduction

Crustal strain rates are fundamentally important quantities for assessing seismic hazard. This is because the locations where strain is rapidly accumulating are the locations where we can expect strain energy to be released seismically. It is then important to develop and improve upon methods for mapping strain in tectonically active regions because such maps could conceivably feed into seismic hazard models such as UCERF3 (Field et al., 2014).

Maps of strain rate can be derived from geodetic measurements of ground displacements, and there are numerous methods for doing so. The classic and simplest way is to assume that the strain rate is constant in time and spatially uniform within subnetworks of the geodetic data. Linear least squares is then used to find the components of the strain rate tensor for each subnetwork (e.g. Frank, 1966; Prescott, 1976; Savage et al., 1986; Feigl et al., 1993; Murray and Lisowski, 2000). Several algorithms have been developed to improve upon this procedure for calculating strain. Shen et al. (1996) and Shen et al. (2015) discuss an algorithm where, instead of using the immediately adjacent stations to calculate strain at a position, the strain is computed with a weighted average over the entire network where the weighting is smaller for more distant stations. Another strategy is to fit a set of interpolating basis functions to the velocity field and then compute the strain from the analytical derivative of the interpolant (e.g. Beavan and Haines, 2001; Tape et al., 2009). The aforementioned studies have all been concerned with estimating long term strain rates. Time dependent strain would be useful for studying geophysical processes which occur over timescales of days to years such as slow slip events, postseismic relaxation, or volcanic deformation. Ohtani et al. (2010) describes a Kalman filter based method for computing time dependent strain by fitting a set of basis functions to a time dependent displacement field and enforcing temporal smoothness in the basis function coefficients.

In essence, estimating strain rates is a matter of numerically calculating the spatial derivative of a geodetically observed velocity field. Any method proposed for calculating strain rates must be able to handle two complications; 1) geodetic velocity estimates are noisy and differentiation will only amplify the noise and 2) velocities are not observed on a regular grid, which prevents the use of standard finite difference methods for computing derivatives. In this paper we demonstrate that both of these complications can be elegantly handled with the recently popularized Radial Basis Function-Finite Difference (RBF-FD) method (Wright and Fornberg, 2006).

The RBF-FD method was introduced as a computationally efficient way to solve large scale partial differential equations over irregular, multi-dimensional domains. The RBF-FD method can be thought of as a generalization of the traditional finite difference method, where the node layout is no longer restricted to regular grids. Indeed, the RBF-FD method can be used to estimate derivatives of discrete data located at arbitrary scattered positions in multi-dimensional space. The RBF-FD method is particularly appealing because it is algorithmically simple, regardless of the domain shape or node layout, and also because the method has performed well in numerous benchmark tests (Fornberg and Flyer, 2015, and references therein).

In this paper, we do not use the RBF-FD method to solve a partial differential equation, but rather we use it to spatially smooth and differentiate GPS derived velocity data. Our smoothing strategy can be viewed as a low-pass filter for scattered data where the degree of smoothness is controlled by a user specified cutoff frequency. This can be contrasted with interpolation based smoothing strategies (e.g. Howell et al., 2016) where the resulting interpolant can be largely and unpredictably controlled by the choice of basis function. After spatially smoothing the velocity field we differentiate it with the RBF-FD method to get a strain rate map. We also demonstrate that this procedure can be used to estimate time dependent strain rates. In that case, we first temporally smooth and differentiate GPS displacement time series to get time dependent velocities. We then spatially smooth and differentiate the resulting velocities for each time epoch.

The method proposed in this paper has numerous advantages which set it apart from other methods for computing strain rates. The method is computationally efficient and stable (there is no inversion of an ill-conditioned matrix). There are no hyper parameters or penalty parameters that need to be tuned for each application. As opposed to interpolation strategies such as Beavan and Haines (2001), Tape et al. (2009), or Ohtani et al. (2010), our method assumes that velocities are locally rather than globally continuous, which allows us to easily handle discontinuities resulting from, for example, a creeping fault.

We begin this paper by summarizing the RBF-FD method and explaining how we construct differentiation matrices for scattered data. We then introduce the smoothing strategy, which is applied to the observed geodetic data prior to differentiation. We then provide two real world demonstrations of our method for calculating strain rates. First we calculate the long term strain rates in Southern California from the CMM3 velocity data set (Shen et al., 2011), and we verify that our results are consistent with other studies. We then calculate time dependent strain rates in Cascadia from the GPS data provided by UNAVCO. In Cascadia, we analyze strain resulting from slow slip events and compare it to the long term tectonic strain accumulation. Slow slip events are found to produce compression in the Olympic Peninsula, which is in addition to the compression resulting from tectonic loading. Further south in Oregon, the slow slip events tend to release the compressional strain that is accumulated tectonically. While similar conclusions have been drawn from fault slip inversions for

slow slip events, it is important to recognize that slip inversion are the product of inverting an ill-conditioned matrix making it difficult to determine whether slip inferences are real or just an artifact of the inversion. The strain rates presented in this paper are more direct observations and can be interpreted with a higher degree of confidence.

2 Method

2.1 Differentiating Scattered Data

In this section we briefly summarize the RBF-FD method and we refer the reader to Wright and Fornberg (2006) or Fornberg and Flyer (2015) for additional details. Consider a set of nodes $\mathbf{x} = \{x_1, \dots, x_N\}$ in \mathbb{R}^d and corresponding observations of a scalar valued functions $\mathbf{u} = \{u(x_1), \dots, u(x_N)\}$. We want to find a differentiation matrix \mathbf{L} , such that $\mathbf{L}\mathbf{u}$ approximates the linear differential operator \mathcal{L} acting on u at nodes x_1, \dots, x_N . For each node x_i we approximate $\mathcal{L}[u(x)]|_{x=x_i}$ as a weighted sum of $\{u(x_j) : j \in \mathcal{S}_i\}$ where \mathcal{S}_i consist of i and the subscripts for the $n - 1$ nearest neighboring nodes to x_i . The approximation can be written as

$$\mathcal{L}[u(x)]|_{x=x_i} \approx \sum_{j \in \mathcal{S}_i} L_{ij} u(x_j) \quad (1)$$

where L_{ij} are the weights making up the differentiation matrix \mathbf{L} . We refer to x_i and its $n - 1$ nearest neighbors as the stencil for x_i , and it is denoted as $\mathbf{x}^i = \{x_j : j \in \mathcal{S}_i\} = \{x_1^i, \dots, x_n^i\}$. The corresponding weights for each node in \mathbf{x}^i are denoted as $\mathbf{w}^i = \{L_{ij} : j \in \mathcal{S}_i\} = \{w_1^i, \dots, w_n^i\}$. We can then equivalently write eq. (1) as

$$\mathcal{L}[u(x)]|_{x=x_i} \approx \sum_{j=1}^n w_j^i u(x_j^i). \quad (2)$$

Following Fornberg and Flyer (2015), we find the components of \mathbf{w}^i , and thus also of \mathbf{L} , by solving the linear system of equations

$$\begin{bmatrix} \phi(\|x_1^i - x_1^i\|) & \cdots & \phi(\|x_n^i - x_1^i\|) & \vdots & \psi_1(x_1^i) & \cdots & \psi_m(x_1^i) \\ \vdots & & \vdots & & \vdots & & \vdots \\ \phi(\|x_1^i - x_n^i\|) & \cdots & \phi(\|x_n^i - x_n^i\|) & \vdots & \psi_1(x_n^i) & \cdots & \psi_m(x_n^i) \\ \hline \psi_1(x_1^i) & \cdots & \psi_1(x_n^i) & 0 & \cdots & 0 \\ \vdots & & \vdots & \vdots & & \vdots \\ \psi_m(x_1^i) & \cdots & \psi_m(x_n^i) & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} w_1^i \\ \vdots \\ w_n^i \\ \lambda_1 \\ \vdots \\ \lambda_m \end{bmatrix} = \begin{bmatrix} \mathcal{L}[\phi(\|x - x_1^i\|)]|_{x=x_i} \\ \vdots \\ \mathcal{L}[\phi(\|x - x_n^i\|)]|_{x=x_i} \\ \hline \mathcal{L}[\psi_1(x)]|_{x=x_i} \\ \vdots \\ \mathcal{L}[\psi_m(x)]|_{x=x_i} \end{bmatrix} \quad (3)$$

for each stencil. In eq. (3), ϕ is a radial basis function (RBF) which we describe below, $\|\bullet\|$ indicates the L_2 norm, ψ_i are monomial basis functions that span the space of all d -dimensional polynomials with a specified degree p (e.g. $\{1, x, y\}$ for $d = 2$ and $p = 1$), and λ_i are parameters that are estimated along with w_j^i when eq. (3) is inverted but they serve no purpose and can be discarded.

Throughout this paper we use a cubic RBF for ϕ ,

$$\phi(r) = r^3. \quad (4)$$

The cubic RBF is an odd degree polyharmonic spline which has the benefit of being scale invariant and thus there is no scaling parameter that needs to be optimized, unlike for many other common choices of RBFs (e.g. Larsson and Fornberg, 2003). We note that the results presented in this paper remain virtually unchanged when we use other polyharmonic splines for ϕ , which is consistent with the findings of Flyer et al. (2016).

We now elaborate on the stencil size, n , and the polynomial degree, p . We choose p to be equal to the degree of the derivative which we are approximating. This choice is based on the analysis of Flyer et al. (2016) and it ensures that eq. (1) will converge to the true derivative as the distance between nodes decreases. The accuracy of eq. (1) also generally improves with larger values of n , but at the expense of computational costs. We then choose n to be large enough for eq. (1) to converge. For the demonstrations in this paper, we find that $n = 30$ is an appropriate choice. It is worth noting that eq. (3) cannot be inverted when the number of nodes is less than the number of monomial basis functions, m . We then have a lower bound on n which is $n \geq m = \binom{p+d}{d}$.

As mentioned, the stencil \mathbf{x}^i consists of the n nearest neighbors to x_i . An obvious assumption in eq. (1) is that $u(x)$ is sufficiently smooth over the footprint of \mathbf{x}^i . If there is a known discontinuity in $u(x)$, then it is still possible for eq. (1) to accurately approximate $\mathcal{L}[u(x)]$ as long as no stencils overlap the discontinuity.

Programatically, this can be easily implemented by redefining the distance norm used to determine the nearest neighbors. When there is a known discontinuity, we define the distance between two nodes to be the L_2 norm, unless the line segment connecting the two nodes intersects the discontinuity. If there is an intersection then the distance between the two nodes is considered infinite. We use this modified distance norm to account for the known creeping segment of the San Andreas fault in Section 3.1.

2.2 RBF-FD Filter

We now introduce the RBF-FD filter which is a method for smoothing irregularly spaced d -dimensional data. Existing methods for performing this task often fall in one of two categories. One approach is to fit a set of basis functions to the observations using least-squares or regularized least-squares (e.g. Fasshauer, 2007). The second category is Gaussian process regression, which is a Bayesian technique where a stochastic prior model is assumed for the underlying signal (e.g. Rasmussen and Williams, 2006). Kriging is among the better known examples of Gaussian process regression (Matheron, 1963). The RBF-FD filter is also an example of Gaussian process regression, and it distinguishes itself from other methods because it is computationally efficient and because it can be conveniently viewed as a low-pass filter with a user defined cut-off frequency. Furthermore, the RBF-FD filter can easily be designed so that the data is not smoothed across known discontinuities. In the following discussion of the RBF-FD filter, we seek to find a smoothed solution, u_{post} , from the irregularly spaced, observed data, u_{obs} . We constrain u_{post} with the observation equation

$$u_{\text{post}} = u_{\text{obs}} + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \mathbf{C}_{\text{obs}}), \quad (5)$$

and the prior model

$$u_{\text{prior}} \sim \mathcal{N}(0, \mathbf{C}_{\text{prior}}), \quad (6)$$

where ϵ and u_{prior} are considered to be Gaussian processes with zero mean and covariances \mathbf{C}_{obs} and $\mathbf{C}_{\text{prior}}$ respectively. The solution for u_{post} minimizes the objective function

$$\|u_{\text{post}} - u_{\text{obs}}\|_{\mathbf{C}_{\text{obs}}}^2 + \|u_{\text{post}}\|_{\mathbf{C}_{\text{prior}}}^2 \quad (7)$$

and is itself a Gaussian process with a distribution described by

$$u_{\text{post}} \sim \mathcal{N}(\bar{u}_{\text{post}}, \mathbf{C}_{\text{post}}). \quad (8)$$

We use \bar{u}_{post} and \mathbf{C}_{post} to denote the mean and covariance of u_{post} respectively. Using Bayesian linear regression (Tarantola, 2005) these values are found to be

$$\begin{aligned} \bar{u}_{\text{post}} &= (\mathbf{C}_{\text{obs}}^{-1} + \mathbf{C}_{\text{prior}}^{-1})^{-1} \mathbf{C}_{\text{obs}}^{-1} u_{\text{obs}} \\ \mathbf{C}_{\text{post}} &= (\mathbf{C}_{\text{obs}}^{-1} + \mathbf{C}_{\text{prior}}^{-1})^{-1}. \end{aligned} \quad (9)$$

\mathbf{C}_{obs} is presumably well known, while $\mathbf{C}_{\text{prior}}$ needs to be chosen based on an understanding of the underlying signal which we are trying to estimate. In Section 2.2.1 we discuss our choice for $\mathbf{C}_{\text{prior}}$ and provide demonstrations with one-dimensional data. The natural extension for $\mathbf{C}_{\text{prior}}$ when dealing with d -dimensional data is discussed in Section 2.2.2.

2.2.1 Filtering in One Dimension

For one-dimensional data we consider a prior which can be stated implicitly as

$$\mathbf{D}_n u_{\text{prior}} = q, \quad q \sim \mathcal{N}(0, \lambda^2), \quad (10)$$

where \mathbf{D}_n is an n 'th order differentiation matrix, and q is white noise with constant variance λ^2 . If we momentarily ignore the fact that \mathbf{D}_n is not invertible then we can explicitly write our prior covariance as

$$\mathbf{C}_{\text{prior}} = \lambda^2 (\mathbf{D}_n^T \mathbf{D}_n)^{-1}. \quad (11)$$

The filtered mean and covariance for the posterior are then

$$\begin{aligned} \bar{u}_{\text{post}} &= (\mathbf{C}_{\text{obs}}^{-1} + \frac{1}{\lambda^2} \mathbf{D}_n^T \mathbf{D}_n)^{-1} \mathbf{C}_{\text{obs}}^{-1} u_{\text{obs}} \\ \mathbf{C}_{\text{post}} &= (\mathbf{C}_{\text{obs}}^{-1} + \frac{1}{\lambda^2} \mathbf{D}_n^T \mathbf{D}_n)^{-1}. \end{aligned} \quad (12)$$

This filtered solution is closely tied to several well established methods of smoothing. For example, one can immediately recognize eq. (12) as an example of Tikhonov regularization (Tikhonov and Arsenin, 1978). We also note a similarity between eq. (12) and smoothing splines (Wahba, 1990). To see this similarity, we first recall that in one-dimension, a smoothing spline is defined as the function, $f(t)$, which minimizes,

$$\sum_{i=1}^P (u_{\text{obs}}^i - f(t_i))^2 + \alpha \int_{t_1}^{t_P} f^{(n)}(t) dt, \quad (13)$$

where u_{obs}^i is an observation at time t_i , P is the number of observations, α is a smoothing parameter, and $f^{(n)}$ denotes the n 'th time derivative of f . If we ignore data uncertainties (i.e. $\mathbf{C}_{\text{obs}} = \mathbf{I}$), we can see from eq. (7) that \bar{u}_{post} is the discrete function which minimizes

$$\|u_{\text{obs}} - \bar{u}_{\text{post}}\|_2^2 + \frac{1}{\lambda^2} \|D_n \bar{u}_{\text{post}}\|_2^2. \quad (14)$$

If u_{obs} is sampled at evenly spaced increments then, with the appropriate choice of smoothing parameters, eq. (14) can be recognized as a discretized form of (13) and we would thus expect $f(t)$ and \bar{u}_{post} to be effectively the same. The similarities break down when dealing with non-uniformly spaced observations.

Whether we use a smoothing spline or eq. (12) to filter u_{obs} , we must choose the appropriate penalty parameter. One common method for choosing an appropriate penalty parameter is generalized cross-validation (Craven and Wahba, 1979), which yields a filtered solution with the maximum predictive power. There is merit to using an entirely objective approach such as cross-validation, and this would be appropriate if there is no prior knowledge of the signal's characteristic wavelength. Otherwise, it may be better to choose a penalty parameter that damps out all the high frequency oscillations which are known to be noise. We elaborate on this point by demonstrating that eq. (12) can also be viewed as a low-pass filter with a cutoff frequency determined by λ .

We wish to transform \bar{u}_{post} in eq. (12) to the frequency domain and in order to do so we assume that u_{obs} has a constant sampling rate. We also require that ϵ and u_{prior} are stationary stochastic processes (i.e. their statistical properties are invariant to time shifts). We require them to be stationary so that matrix multiplication by \mathbf{C}_{obs} and $\mathbf{C}_{\text{prior}}$ can be viewed as convolution in the time domain and thus multiplication in the frequency domain. We then consider the simplifying case where ϵ is white noise with constant variance σ^2 (i.e. $\mathbf{C}_{\text{obs}} = \sigma^2 \mathbf{I}$), and \mathbf{D}_n is the periodic spectral differentiation matrix (e.g. Trefethen, 2000). Under a discrete Fourier transform, \mathbf{D}_n has the properties

$$\mathcal{F}[\mathbf{D}_n g] = (2\pi i \omega)^n \hat{g} \quad (15)$$

and

$$\mathcal{F}[\mathbf{D}_n^T g] = (-2\pi i \omega)^n \hat{g}, \quad (16)$$

where ω is the frequency domain variable, g is an arbitrary vector and \hat{g} is its discrete Fourier transform. The discrete Fourier transform of \bar{u}_{post} is then

$$\hat{u}_{\text{post}}(\omega) = \frac{\frac{1}{\sigma^2}}{\frac{1}{\sigma^2} + \frac{(2\pi\omega)^{2n}}{\lambda^2}} \hat{u}_{\text{obs}}(\omega). \quad (17)$$

We make the change of variables

$$\lambda^2 = (2\pi\omega_c)^{2n} \sigma^2 \quad (18)$$

which changes the hyperparameter from λ to ω_c . The reason for this change of variables becomes apparent when we simplify eq. (17) to

$$\hat{u}_{\text{post}}(\omega) = \frac{1}{1 + \left(\frac{\omega}{\omega_c}\right)^{2n}} \hat{u}_{\text{obs}}(\omega). \quad (19)$$

We can recognize eq. (19) as an n 'th order low-pass Butterworth filter with cut-off frequency ω_c . In the limit as $n \rightarrow \infty$ eq. (19) becomes an ideal low-pass filter which removes all frequencies above ω_c and leaves lower frequencies unaltered. Of course, an ideal low-pass filter is often undesirable because it will tend to produce ringing artifacts in the filtered solutions. When modeling u_{prior} as Brownian motion or integrated Brownian motion, where $n = 1$ and $n = 2$ respectively, the transfer function is tapered across ω_c , which ameliorates ringing in the filtered solution.

By demonstrating that eq. (12) can be made equivalent to a low-pass filter, it may not be clear why we would ever use eq. (12) when it is far more efficient to filter in the frequency domain through the Fast Fourier Transform. In order to make use of the Fast Fourier Transform, the observations must be made at a constant

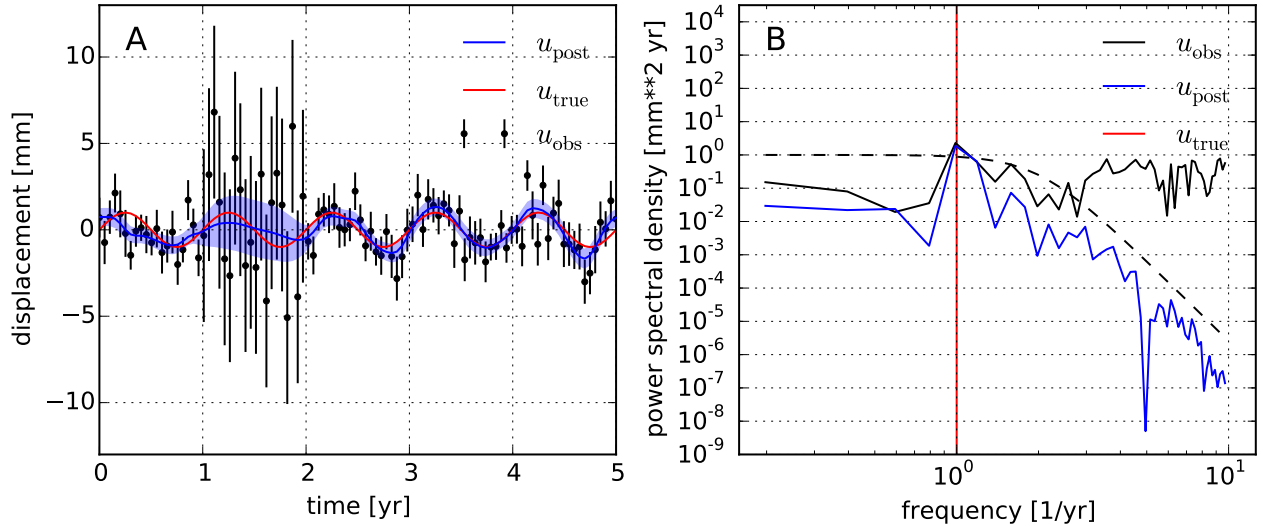


Figure 1: Panel A shows u_{obs} (black scatter points), u_{post} (blue line), and the true signal which we are trying to recover (red line). The lines on each scatter point and the light blue region show the one standard deviation uncertainty for the observations and filtered solution respectively. Panel B shows the estimated power spectral density for the observed, filtered, and true signal. The black dashed line is the squared transfer function in eq. (19) with $\omega_c = 2 \text{ yr}^{-1}$, which roughly indicates how eq. (12) scales the frequency content of u_{obs} . If the data variances were constant then the frequency content of u_{obs} and \bar{u}_{post} would be exactly related by the black dashed line.

sampling rate and the observation noise must be white with constant variance. In contrast, these conditions do not need to be met in order to evaluate eq. (12). The question is then whether eq. (12) still effectively acts as a low-pass filter when the idealized conditions are not met.

We address this question here with two numerical demonstrations. In this first demonstration we generate 100 samples of synthetic data over the time interval $0 < t < 1 \text{ yr}$. The true signal in the synthetic data is a sine wave with a frequency of 1 yr^{-1} and 1 mm amplitude. We obscure the synthetic data with white noise that does not have a constant variance. We randomly pick a variance for each datum from a uniform random distribution ranging from 0.5 to 2 mm^2 . For the period from 1 to 2 years, we assign a variance of 25 mm^2 to the data, effectively making the data uninformative for the filtered solution. The synthetic data and its estimated power spectral density are plotted in Figure 1. Our goal is to use eq. (12) as a low-pass filter which damps out frequencies that are higher than our signal. Since our data does not have constant variance, we cannot choose a cutoff frequency from eq. (18). Instead we define a characteristic data variance as

$$\frac{1}{\bar{\sigma}^2} = \frac{1}{P} \text{tr}(\mathbf{C}_{\text{obs}}^{-1}), \quad (20)$$

where P is the number of observations, and we relate λ to ω_c by

$$\lambda^2 = (2\pi\omega_c)^{2n}\bar{\sigma}^2. \quad (21)$$

In this demonstration D_n is still the spectral differentiation matrix and we use $n = 2$. Our cutoff frequency is chosen to be $\omega_c = 2 \text{ yr}^{-1}$. The filtered solution and its frequency content are plotted Figure 1. Based on panel B, eq. (12) is indeed acting as a low-pass filter with the specified cut-off frequency even though the variances are not constant. However, the frequency content of subsections of \bar{u}_{post} appears to change depending on the variance of u_{obs} . For example, Over the interval $1 < t < 2$, where the variance of u_{obs} is higher, u_{post} lacks the higher frequency oscillations that can be seen in the remainder of the time series. This is desirable behavior because we do not want u_{post} contorting to fit dubious data. It is also worth pointing out that u_{post} asymptotically approaches a solution that is not significantly different from that shown in Figure 1 when the variance for u_{obs} increases to infinity over the interval $1 < t < 2$. This too is desirable behaviour because one could then surmise that an appropriate way to handle missing data with eq. (12) is to assign u_{obs} an infinite variance for the missing time period. With this insight, we can then extend the application of eq. (12) from data smoothing to data interpolation and extrapolation.

We present a similar demonstration to show that eq. (12) effectively acts as a low-pass filter even when the observations are not uniformly spaced. We take samples over the time interval (0, 10) yr, where the sampling rate is 0.05 yr^{-1} for the first 5 years and 0.0125 yr^{-1} for the last 5 years. We use the same underlying signal

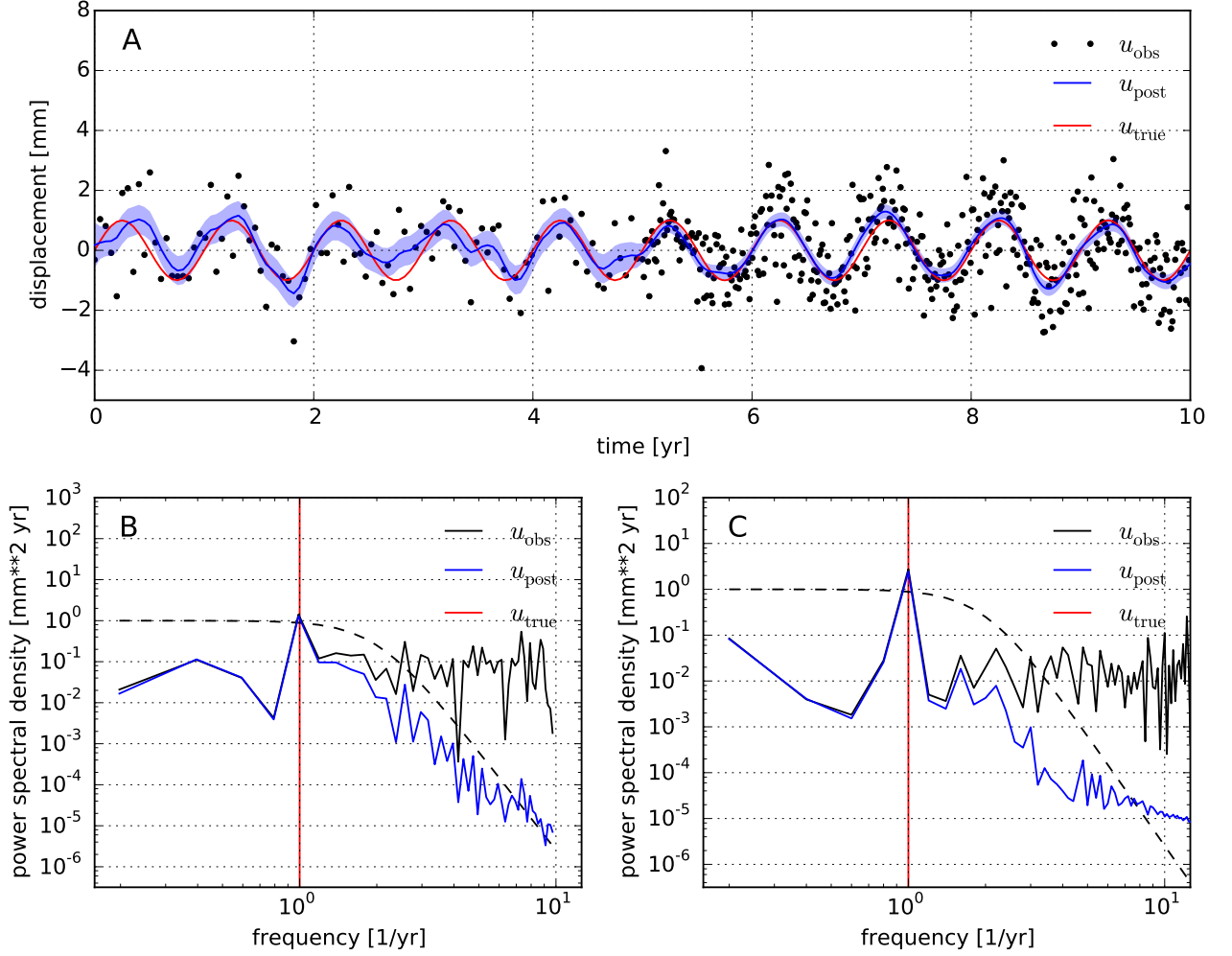


Figure 2: Panel A shows u_{obs} (black scatter points), u_{post} (blue line), and the true signal which we are trying to recover (red line). The light blue region show the one standard deviation uncertainty for u_{post} . The uncertainties for the observations are all 1 mm and are not shown for the sake of clarity. Panel B shows the estimated power spectral density for the observed, filtered, and true signal over the time interval (0,5) yr. Panel C is the same as panel B except for the time interval (5,10) yr. In both panel B and C the black dashed line is the squared transfer function in eq. (19) with $\omega_c = 2 \text{ yr}^{-1}$.

from the previous demonstration and we add noise with a constant variance of 1 mm^2 . We keep $n = 2$ but since we are dealing with non-uniformly spaced data, we can no longer use the periodic spectral differentiation matrix, and \mathbf{D}_n is instead a first-order accurate finite difference matrix for arbitrarily spaced data, which can be computed with traditional finite difference methods Fornberg (1996) or the procedure in Section 2.1. Once again, our cutoff frequency is $\omega_c = 2 \text{ yr}^{-1}$. The synthetic data and filtered solution are shown in panel A of Figure 2. The power spectral density for the time intervals (0, 5) and (5, 10) yr are shown in panel B and C, respectively. In both panels B and C, the spectral density of the filtered solution diverges from the spectral density of the synthetic observations at the expected cutoff frequency of ω_c . For frequencies above $\sim 5 \text{ yr}^{-1}$, the rate that the spectral content decays is slower than what is predicted by the transfer function in eq. (19). This is most pronounced in panel C, and we attribute the discrepancy to our lower order differentiation matrix. Barring the high frequencies which have negligible power in \bar{u}_{post} , the power spectral density of \bar{u}_{post} relative to u_{obs} is accurately described by the transfer function in eq. (19), despite our use of non-uniform spacing.

2.2.2 Filtering in Higher Dimensions

We expand our discussion to smoothing data which is observed in d -dimensional space. The filtered solution is still given by eq. (9), and we consider the prior model

$$\mathbf{L}_n u_{\text{prior}} = q, \quad q \sim \mathcal{N}(0, \lambda^2) \quad (22)$$

where \mathbf{L}_n is a differentiation matrix which approximates the operation

$$\sum_{i=1}^d \frac{\partial^n}{\partial x_i^n} \quad (23)$$

and n is an even integer. In general, we can construct \mathbf{L}_n for arbitrarily spaced data with the strategy described in Section 2.1. The corresponding covariance matrix is then

$$\mathbf{C}_{\text{prior}} = \lambda^2 (\mathbf{L}_n^T \mathbf{L}_n)^{-1}. \quad (24)$$

Using the change of variables from eq. (21), the solution in the time domain is

$$\begin{aligned} \bar{u}_{\text{post}} &= (\mathbf{C}_{\text{obs}}^{-1} + \frac{1}{(2\pi\omega_c)^{2n}\bar{\sigma}^2} \mathbf{L}_n^T \mathbf{L}_n)^{-1} \mathbf{C}_{\text{obs}}^{-1} u_{\text{obs}} \\ \mathbf{C}_{\text{post}} &= (\mathbf{C}_{\text{obs}}^{-1} + \frac{1}{(2\pi\omega_c)^{2n}\bar{\sigma}^2} \mathbf{L}_n^T \mathbf{L}_n)^{-1}. \end{aligned} \quad (25)$$

If we again assume that the observation are regularly spaced, have constant variance, and \mathbf{L}_n is the corresponding spectral differentiation matrix, then the d -dimensional discrete Fourier transform of \bar{u}_{post} is

$$\hat{u}_{\text{post}}(\omega_1, \dots, \omega_d) = \frac{1}{1 + \left(\sum_{i=1}^d \left(\frac{\omega_i}{\omega_c} \right)^n \right)^2} \hat{u}_{\text{obs}}. \quad (26)$$

The transfer function in eq. (26) can once again be recognized as a low-pass filter. Namely, in the limit as $n \rightarrow \infty$, the transfer function becomes a d -dimensional box which is zero for all the frequency tuples, $(\omega_1, \dots, \omega_n)$, which have a component greater than ω_c .

When the data is not regularly spaced, eq. (25) is no longer a low-pass filter, strictly speaking, because the Discrete Fourier Transform is not clearly defined in that case. Nonetheless, numerical tests verify that the shortest feature wavelength retained in the filtered solution is consistent with the specified cutoff frequency, regardless of the data spacing. For this reason, we argue that eq. (25) effectively acts as a low-pass filter for irregularly spaced, d -dimensional data.

3 Applications

3.1 Strain Rate in Southern California

give definition of strain remark on how uncertainties are computed and also how they dont account for error in L

3.2 Time Depenent Strain Rate in Cascadia

4 Discussion and Conclusion

References

- Beavan, J. and Haines, J. (2001). Contemporary horizontal velocity and strain rate fields of the Pacific-Australian plate boundary zone through New Zealand. *Journal of Geophysical Research*, 106(B1):741–770.
- Craven, P. and Wahba, G. (1979). Smoothing noisy data with spline functions: estimating the correct degree of smoothing by the method of generalized cross-validation. *Numerische Mathematik*, 403:377–403.
- Fasshauer, G. E. (2007). *Meshfree Approximation Method with Matlab*. World Scientific Publishing Co., Singapore.
- Feigl, K. L., Agnew, D. C., Bock, Y., and Dong, D. (1993). Space Geodetic Measurement of Crustal Deformation in Central and Southern California, 1984–1992. *Journal of Geophysical Research*, 98(B12):21 677–21 712.
- Field, E. H., Arrowsmith, R. J., Biasi, G. P., Bird, P., Dawson, T. E., Felzer, K. R., Jackson, D. D., Johnson, K. M., Jordan, T. H., Madden, C., Michael, A. J., Milner, K. R., Page, M. T., Parsons, T., Powers, P. M., Shaw, B. E., Thatcher, W. R., Weldon, R. J., and Zeng, Y. (2014). Uniform California Earthquake Rupture Forecast, version 3 (UCERF3) -The time-independent model. *Bulletin of the Seismological Society of America*, 104(3):1122–1180.

- Flyer, N., Fornberg, B., Bayona, V., and Barnett, G. A. (2016). On the role of polynomials in RBF-FD approximations: I. Interpolation and accuracy. *Journal of Computational Physics*, 321:21–38.
- Fornberg, B. (1996). *A practical guide to pseudospectral methods*. Cambridge University Press.
- Fornberg, B. and Flyer, N. (2015). *A Primer on Radial Basis Functions with Applications to the Geosciences*. Society for Industrial and Applied Mathematics, Philadelphia.
- Frank, C. F. (1966). Deduction of earth strains from survey data. *Bulletin of the Seismological Society of America*, 56(1):35–42.
- Howell, S., Smith-Konter, B., Frazer, N., Tong, X., and Sandwell, D. (2016). The vertical fingerprint of earthquake cycle loading in southern California. *Nature Geoscience*, (June).
- Larsson, E. and Fornberg, B. (2003). A numerical study of some radial basis function based solution methods for elliptic PDEs. *Computers and Mathematics with Applications*, 46:891–902.
- Matheron, G. (1963). Principles of geostatistics. *Economic Geology*, 58(8):1246–1266.
- Murray, M. H. and Lisowski, M. (2000). Strain accumulation along the Cascadia subduction zone in western Washington. *Geophysical Research Letters*, 27(22):3631–3634.
- Ohtani, R., McGuire, J. J., and Segall, P. (2010). Network strain filter: A new tool for monitoring and detecting transient deformation signals in GPS arrays. *Journal of Geophysical Research: Solid Earth*, 115(12):1–17.
- Prescott, W. H. (1976). An extension of Frank’s method for obtaining crustal shear strains from survey data. *Bulletin of the Seismological Society of America*, 66(6):1847–1853.
- Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian processes for machine learning*. The MIT Press.
- Savage, J. C., Prescott, W. H., and Gu, G. (1986). Strain accumulation in southern California, 1973–1984. *Journal of Geophysical Research*, 91(B7):7455–7473.
- Shen, Z., Wang, M., Zeng, Y., and Wang, F. (2015). Optimal Interpolation of Spatially Discretized Geodetic Data. *Bulletin of the Seismological Society of America*, 105(4):2117–2127.
- Shen, Z. K., Jackson, D. D., Ge, B. X., and Bob, X. G. (1996). Crustal deformation across and beyond the Los Angeles basin from geodetic measurements. *Journal of Geophysical Research*, 101(B12):27927–27957.
- Shen, Z. K., King, R. W., Agnew, D. C., Wang, M., Herring, T. A., Dong, D., and Fang, P. (2011). A unified analysis of crustal motion in Southern California, 1970–2004: The SCEC crustal motion map. *Journal of Geophysical Research: Solid Earth*, 116(11):1–19.
- Tape, C., Musé, P., Simons, M., Dong, D., and Webb, F. (2009). Multiscale estimation of GPS velocity fields. *Geophysical Journal International*, 179(2):945–971.
- Tarantola, A. (2005). *Inverse problem theory and methods for model parameter estimation*. SIAM.
- Tikhonov, A. N. and Arsenin, V. Y. (1978). Solutions of Ill-Posed Problems . by A . N . Tikhonov ; V . Y . Arsenin Review by : John B . Bell Stable URL : <http://www.jstor.org/stable/2006360> . *American Mathematical Society*, 32(144):1320–1322.
- Trefethen, L. N. (2000). *Spectral Methods in Matlab*. SIAM.
- Wahba, G. (1990). *Spline Models for Observational Data*. Philadelphia.
- Wright, G. B. and Fornberg, B. (2006). Scattered node compact finite difference-type formulas generated from radial basis functions. *Journal of Computational Physics*, 212(1):99–123.