Mathematical Geology, Vol. 38, No. 8, November 2006 (© 2007)
DOI: 10.1007/s11004-006-9056-6

Forward and Inverse Transformations between Cartesian and Channel-fitted Coordinate Systems for Meandering Rivers[1]

Carl J. Legleiter[2,3] and Phaedon C. Kyriakidis[2]

The spatial referencing of river channels is complicated by their meandering planform, which dictates that Euclidean distance in a Cartesian reference frame is not an appropriate metric. Channel-fitted coordinate systems are thus widely used in application-oriented geostatistics as well as theoretical fluid mechanics, where flow patterns are described in terms of a streamwise axis s along the channel centerline and an axis n normal to that centerline. A means of transforming geographic (x, y) coordinates to their equivalents in the (s, n) space and vice versa is needed to relate the two frames of reference, and this paper describes a pair of transformation algorithms that are explicitly intended for reach-scale studies of modern rivers. The forward transformation from Cartesian to channel-fitted coordinates involves parametric description of the centerline using cubic splines, calculation of centerline normal vectors and curvature using results from differential geometry, and an efficient local search to find in-channel data points and compute their (s, n) coordinates. The inverse transformation finds the nearest vertices of a discretized centerline and uses a finite difference approximation to the streamwise rates of change of the centerline's Cartesian coordinates to obtain the geographic equivalent of a point in the (s, n) space. The performance of these algorithms is evaluated using: (i) field data from a gravel-bed river to examine the effects of initial centerline digitization and subsequent filtering; and (ii) analytically-defined centerlines and simulated coordinates to assess transformation accuracy and sensitivity to centerline curvature and discretization. Any discrepancy between a point's known coordinates in one frame of reference and the coordinates produced via transformation from the other coordinate system constitutes a transformation error, and our results indicate that these errors are 2–4% and 0.2–0.5% of the channel width for the field case and simulated centerlines, respectively. The primary sources of transformation error are the initial digitization of the centerline and the relationship between centerline curvature and discretization.

KEY WORDS: river channel, spatial referencing, curvature, orthogonal curvilinear coordinates.

INTRODUCTION

A meandering planform is a defining characteristic of many alluvial rivers, exerting a primary control on patterns of flow, sediment transport, and channel

[1]Received 26 January 2006; accepted 5 May 2006; Published online: 28 February 2007.
[2]Department of Geography, University of California, 3611 Ellison Hall, Santa Barbara, CA 93106-4060, USA.
[3]Yellowstone Ecological Research Center, Bozeman, MT 59718, USA; e-mail carl@geog.ucsb.edu.

Figure 1. Meandering section of Slough Creek, Yellowstone National Park, USA. In areas of high curvature, Euclidean distances between pairs of points underestimate the along-channel path length, as shown in the bends at the left and right. In relatively straight channel segments (center), the discrepancy between Euclidean and along-channel distances is smaller. Flow is from right to left.

migration. Considerable research effort has been devoted toward these compelling phenomena, including flume experiments (e.g., Hook, 1975), detailed field observations (e.g., Dietrich and Smith, 1983, 1984), theoretical models (e.g., Smith and McLean, 1984), and numerical simulations (e.g., Sun, Meakin, and Jossang, 2001a,b). An important issue common to all treatments of meander form and process is that of spatial referencing, which is complicated by the curved, typically irregular channel planform. This non-convex geometry dictates that the conventional Euclidean distance is not an appropriate metric because a straight line drawn between two points will often cross intervening land (i.e., across a point bar on the convex bank, as illustrated in Fig. 1); a similar situation exists in estuaries, as discussed by Little, Edwards, and Porter (1997) and Rathbun (1998). As a result, observations referenced to a standard Cartesian frame will tend to underestimate distances relative to the in-water, along-channel path length, with undesirable consequences for both spatial statistical analysis and physics-based hydraulic modeling. As an example, consider the interpolation of bed topography: neglecting channel curvature would underestimate inter-point distances and bias the covariance model upon which spatial predictions (e.g., kriging estimates) are based. In fluid mechanics, the need to incorporate the effects of curvature on the force balance within a channel bend has long been recognized (e.g., Dietrich, 1987), and the use of an inappropriate distance metric could also affect the calculation of spatial derivatives (e.g., Smith and McLean, 1984).

A logical solution to this problem is to adopt a frame of reference that is fitted to the channel, and the orthogonal curvilinear coordinate system introduced

by Smith and McLean (1984) has been widely used to model flow and sediment transport in meandering rivers (e.g., Dietrich and Smith, 1983, 1984; Nelson, Bennett, and Wiele, 2003). This coordinate system consists of a streamwise axis s defined by the channel centerline and an n axis normal to that centerline. To ensure a right-handed coordinate system with s increasing downstream and z vertically upward, n is positive toward the left bank, facing downstream (Fig. 2). Rendered in this coordinate system, the governing equations for flow through a bend include the effects of curvature-induced centrifugal acceleration and a metrical coefficient that accounts for differences in streamwise arc length across the channel (i.e., along lines of constant n other than the centerline, where $n = 0$; Smith and McLean, 1984). For spatial analysis, transforming observations from Cartesian (x, y) coordinates to the orthogonal curvilinear (s, n) system circumvents the geometric issues associated with the meandering planform and provides a consistent, intuitive frame of reference for describing the spatial structure of channel morphology and hydraulics. In fact, common practice for geostatistical modeling of buried fluvial systems involves a hierarchical series of coordinate transformations accounting for stratigraphy, faulting, channel belt orientation, and 'straightening' of individual channels (e.g., Deutsch and Want, 1996; Deutsch, 2002). These procedures result in a more convenient, transformed (s, n) space within which covariance modeling, spatial prediction, and stochastic simulation can proceed efficiently using Euclidean distances and regular grids. As long as the coordinate transformations are unique and reversible, the results can then be back-transformed to the original (x, y) coordinate system for visualization and interpretation.

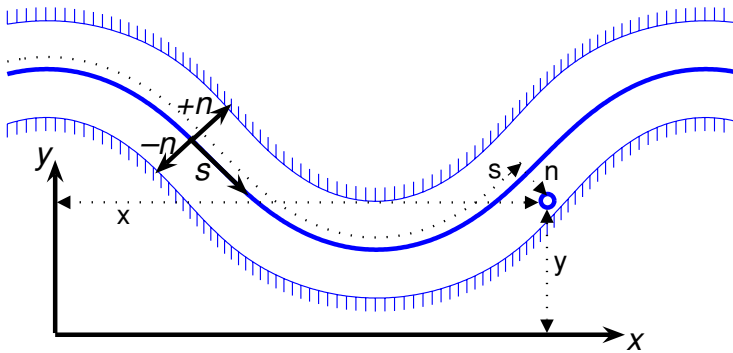

Figure 2. Cartesian (x, y) and channel-centered curvilinear (s, n) coordinate systems. Flow is from left to right, and the s-axis is directed downstream, with the n-axis positive toward the left bank. The thick line represents the channel centerline, the circle represents an arbitrary location within the channel, and the thin dashed lines represent the coordinates in the two frames of reference. Figure after Smith and McLean (1984) and Merwade, Maidment, and Hodges (2005).

Geomorphological and application-oriented geostatistical approaches to the spatial referencing of meandering channels have developed essentially independently of one another, and the problem of unambiguously relating geographic and channel-centered coordinates has been approached from several directions. For example, Smith and McLean (1984) present an analytical expression for transforming (s, n) to (x, y) coordinates, given the Cartesian coordinates of the channel centerline. This equation allows predictions from their fluid mechanical model, expressed in an orthogonal curvilinear frame of reference, to be assigned geographic locations, but an explicit method of defining the (s, n) coordinates of field observations surveyed in a Cartesian system was not provided. Dietrich and colleagues obtained (s, n) coordinates by digitizing a channel centerline from topographic data, establishing cross-sections visually, measuring velocity profiles at several locations along each section, and then using the observed cross-stream discharge to adjust the orientation of the sections as needed to satisfy continuity (Dietrich and Smith, 1983; 1984; Whiting and Dietrich, 1991). Though effective, this technique requires abundant velocity data and closely spaced cross-sections and is thus not practical for larger-scale studies. Conversely, the coordinate transformations introduced by Deutsch and colleagues (Deutsch and Wang, 1996; Deutsch and Tran, 2002; Deutsch, 2002) for geostatistical reservoir characterization emphasize efficient generation of regular grids for large numbers of channels distributed throughout a stratigraphic package. These algorithms thus tend to provide a relatively coarse approximation to channel geometry and were never intended for modern, subaerial rivers. Various non-geometric transformations have also been presented in the statistical literature (e.g., Sampson and Guttorp, 1992; Løland and Høst, 2003), but these approaches fail to provide a clear physical interpretation for the transformed space.

More recently, several methods of transforming coordinates to channel-fitted grids have been proposed, in a range of different contexts. For example, Barabas, Goovaerts, and Adriaens (2001) used grid generation software to 'unbend' a polluted New Jersey river and enable geostatistical modeling of the spatial distribution of contaminated sediments, although an intermediate step was required to interpolate their field observations onto grid nodes, which had different, variable spacings in the streamwise and transverse directions. Whereas Barabas, Goovaers, and Adriaens (2001) relied on generic grid generation software, Goff and Nordfjord (2004) developed a transformation referenced directly to a centerline trace that provided along- and across-channel coordinates analogous to the (s, n) system. Their procedure involved interpreting seismic data to identify points along the channel centerline, filtering these points to avoid gaps and overlaps, and then placing points at specified, constant distances along and locally perpendicular to the centerline to produce a regular, channel-fitted grid. Geographic coordinates were then assigned to these grid nodes by nearest neighbor interpolation such that only those (s, n) points comprising the grid were prospective coordinates for

the transformed (x, y) observations; the resolution of the transformation was thus limited to that of the (s, n) grid. Unlike Goff and Nordfjord (2004), who begin by defining a channel-fitted grid and then relate geographic coordinates to that grid, Merwade, Maidment, and Hodges (2005) recently introduced a transformation algorithm that operates independently of any grid and allows an arbitrary geographic location within the channel to be assigned unique (s, n) coordinates. The only coordinate transformation algorithm explicitly intended for modern rivers of which we are aware, this method uses existing bathymetric data to identify a polyline representing the channel thalweg and then relates in-stream data points to the vertices of this polyline via simple trigonometry. The approach of Merwade, Maidment, and Hodges (2005) was implemented within a GIS environment, which facilitated the integration of various geospatial data sets but also resulted in an algorithm tightly bound to a specific software package and its associated data models.

In this paper, we introduce more formal, flexible coordinate transformation procedures for relating Cartesian to channel-centered coordinates and *vice versa* by describing the mathematical basis of these techniques and outlining generic algorithms for their implementation. We use the terms Cartesian and geographic coordinates interchangeably and denote them by (x, y); Smith and McLean's (1984) channel-fitted orthogonal curvilinear coordinate system is referred to as the (s, n) space. Because we are primarily interested in relating field measurements defined by geographic coordinates to their corresponding (s, n) locations, the forward transformation is from (x, y) to (s, n); the inverse transformation reverts coordinates from the (s, n) space back to the original (x, y) system. These transformations form an essential component of our ongoing research on spatial variations of hydraulic and sedimentary variables in gravel-bed rivers, and the techniques described herein are specifically intended for application to modern, subaerial rivers of moderate sinuosity. We are primarily interested in reach-scale ($\sim$10–20 channel widths or one meander wavelength) studies and do not consider braided channels or drainage network topology.

The coordinate transformation procedures presented herein generalize the algorithms of Merwade, Maidment, and Hodges (2005) and Goff and Nordfjord (2004) by drawing upon the work of of Fagherazzi, Gabet, and Furbish (2004) on tidal channel planforms and incorporating elements of Viseur's (2004) research on turbidite reservoirs. Our approach is similar to that of Merwade, Maidment, and Hodges (2005), but differs in three important respects: (i) no bathymetric data are required to define the channel centerline, which is instead described parametrically; (ii) the mathematical foundation of the transformations is established using results from differential geometry, which also allow for the calculation of centerline curvature; and (iii) the algorithms are explicit and generic, independent of any particular software environment. A key element of these procedures is the description of the Cartesian coordinates of the channel centerline as a pair of

cubic splines parameterized by the streamwise coordinate s (Fagherazzi, Gabet, and Furbish, 2004), and we assess the sensitivity of the transformations to the initial definition and subsequent filtering of centerline vertices using field data from Soda Butte Creek, a gravel-bed river in Yellowstone National Park we are studying in an effort to characterize spatial patterns of morphology and hydraulics at the reach scale. We then use a series of analytically-defined centerlines, which can in turn be used to simulate points with known (x, y) and (s, n) coordinates, to assess the accuracy of our transformation algorithms and investigate the effects of centerline curvature and discretization. The paper concludes with a discussion of these results and the potential applications and extensions of this approach to the spatial referencing of fluvial systems.

FORWARD COORDINATE TRANSFORMATION FROM GEOGRAPHIC TO CHANNEL-CENTERED COORDINATES

Transforming data referenced to a geographic coordinate system into the (s, n) space allows geostatistical modeling and spatial prediction to proceed using Euclidean distances with a clear geometric and physical interpretation, and the results can then be back-transformed to their original geographic coordinates. Our procedure for transforming Cartesian (x, y) to channel-fitted, orthogonal curvilinear (s, n) coordinates is illustrated using an example from our field area in Figures 3 and 4, and involves the following sequence of steps (described in detail in the next several subsections):

1. Select points defining the vertices of an initial, discrete channel centerline.
2. Smooth the initial centerline by filtering the coordinates of its vertices.
3. Use cubic splines to obtain a parametric description of the filtered centerline.
4. Calculate the arc length of each centerline segment.
5. Resample the splines to produce centerline vertices with a regular streamwise spacing.
6. Compute normal vectors and curvature values at each regularly spaced centerline vertex.
7. Search for data points within polygons defined by the centerline and its normal vectors, which are directed toward the left bank.
8. Compute data-centerline distances to obtain (s, n) coordinates.
9. Repeat steps 7 and 8 for the right side of the channel using normal vectors with the opposite orientation.

We have developed MATLAB code for performing the transformation, but the explicit description below should allow for implementation in other programming languages and/or incorporation of these techniques into existing software.

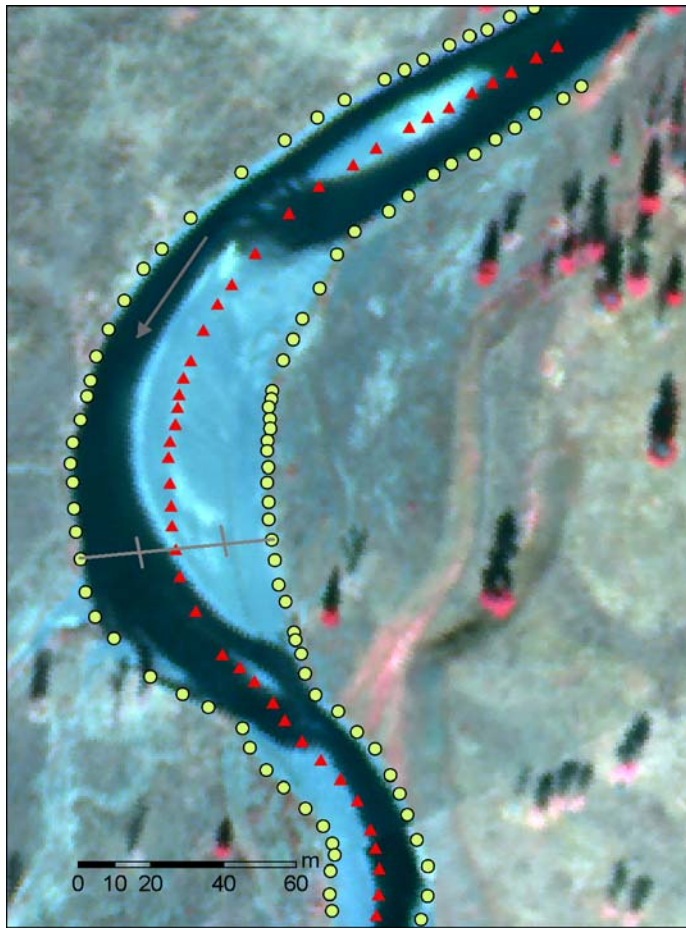

Figure 3. Centerline definition for Soda Butte Creek. To obtain a stage-independent centerline, we digitized pairs of points (circles) along opposite banks and then created centerline vertices (triangles) at the midpoints of the lines (i.e., cross-sections) connecting each pair of bank points. The flow direction is indicated by the arrow.

Centerline Definition: Initial Point Selection

Because the basis of the (s, n) coordinate system is the channel centerline, the transformation algorithm begins by selecting points to serve as the initial vertices of a discrete centerline. These points can be identified from survey data plotted in map view or from remotely sensed data, such as the image of Soda Butte Creek shown in Figure 3. Because the position of the water's edge varies as a function of

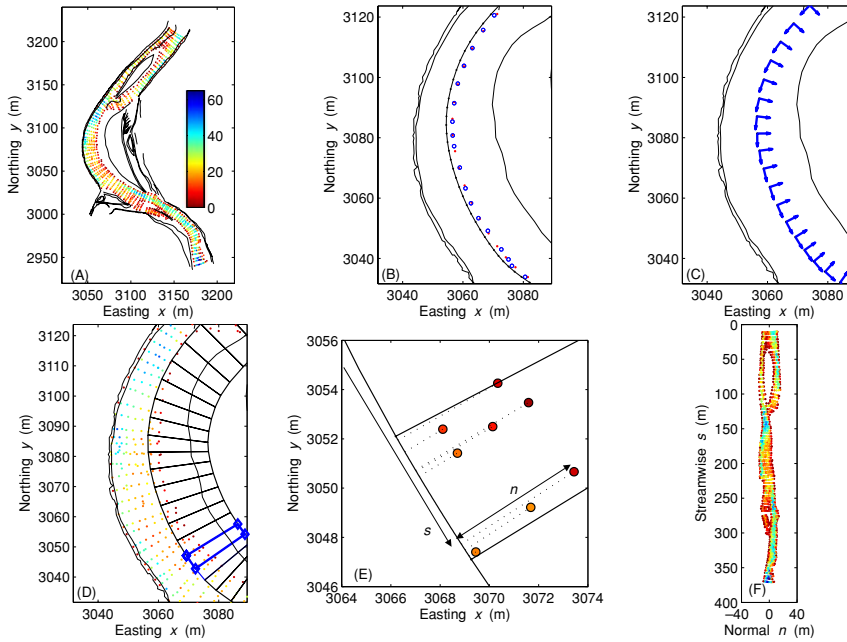

Figure 4. Application of the coordinate transformation to the Footbridge Reach of Soda Butte Creek, shown in Figure 3. (A) base map, with topographic breaklines and point measurements of flow depth; (B) centerline definition, with initial digitized points represented as open circles, filtered vertices as closed circles, and the resampled spline centerline as a solid line with dotted vertices (offset to the west by 2 m for clarity); (C) calculation of unit tangent and unit normal vectors for the centerline (every other vertex shown); (D) creation of search polygons with outer vertices located along centerline normal vectors; (E) inset from highlighted polygon in D, showing the calculation of s and n coordinates for the points within this polygon; (F) depth measurements shown in A transformed to (s, n) coordinates.

flow stage and the location of the channel talweg will not, in practice, be known *a priori* (Merwade, Maidment, and Hodges, 2005), we recommend referencing this initial centerline to the morphologically-defined bankfull channel. If both channel banks can be clearly identified, centerline vertices can be defined by selecting a point along one bank, another on the opposite bank, and then creating a vertex at the midpoint of the line segment (i.e., cross-section) connecting these two bank points (Fig. 3). We have found that the "midpoint" editing tool in ArcGIS provides a convenient interface for selecting these points (ESRI, 2004), but any software environment with interactive graphics can be used to display a base map and digitize either pairs of bank points or centerline vertices directly. Because a detailed trace including fine-scale variations could introduce artifacts in the curvature of the centerline, bank points or centerline vertices should be selected so

as to capture only the gross, reach-scale geometry of the channel planform (Nelson, Bennett, and Wiele, 2003). The filtering process described below also serves to reduce the noise associated with small-scale bank irregularities and the positional uncertainty inherent to point selection (Fagherazzi, Gabet, and Furbish, 2004). Centerline vertices should be digitized in order from upstream to downstream and need not be regularly spaced, but must be sufficient to characterize changes in channel direction and should not be more than a channel width apart. We assess the effects of initial digitization in a later section of the paper. The original, Cartesian coordinates of these centerline vertices are denoted by

$$(x_i, y_i) \qquad i = 1, \ldots, N_b \tag{1}$$

where N_b denotes the number of points used to define the initial discrete centerline.

Centerline Definition: Filtering

The initial centerline points will likely be somewhat irregular, reflecting both the non-uniformity of the banks and perhaps some degree of operator error, but the channel-fitted coordinate system is intended to approximate the average streamline curvature within the reach of interest (Nelson, Bennett, and Wiele, 2003). Deriving a smoothed representation of the centerline is also important because subsequent steps in the transformation involve derivatives, which tend to amplify point-to-point variations (i.e., introduce noise) in the centerline trace. Fagherazzi, Gabet, and Furbish (2004) demonstrated that such errors have a substantial effect on curvature estimation, also based on derivatives, and recommended applying multiple times a Savitzky-Golay filter (Hamming, 1983) to the coordinates of the centerline vertices (Fig. 4B). This filter is essentially a linear combination of data (coordinate values in this case) within a moving window, with coefficients obtained by least-squares fitting a polynomial to all points in the window; Press and others (1994) describe the process in greater detail and provide code for efficient implementation of the filter. Defining the centerline with a Savitzky-Golay filter is advantageous because it smooths the noisy initial trace while preserving the length and amplitude of meander bends, whereas achieving a similar degree of smoothing with a simple moving average filter would result in an altered planform with longer bends of lower amplitude (Press and others, 1994). The number of points m included in the filter window must be an odd integer less than the number N_b of initial centerline vertices and greater than the order p of the polynomial; the number f of times the filter is applied can also vary to adjust the degree of smoothing. We examine the sensitivity of curvature estimation and coordinate transformation to these filter parameters in a later section. Once appropriate parameter values have been selected, the resulting filter weights a_i are applied to

the N_b original centerline coordinate pairs to obtain a smoothed sequence of N_b vertices given by

$$\begin{bmatrix} x_{i,\text{filt}} \\ y_{i,\text{filt}} \end{bmatrix} = \begin{bmatrix} \frac{1}{m} \sum_{j=-(m-1)/2}^{(m-1)/2} a_{i+j} x_{i+j} \\ \frac{1}{m} \sum_{j=-(m-1)/2}^{(m-1)/2} a_{i+j} y_{i+j} \end{bmatrix} \tag{2}$$

Parametric Description of the Centerline Using Cubic Splines

Meandering river channels are well-represented by splines, which pass through each of the (filtered) centerline points and provide a smooth curve by ensuring first- and second-order continuity at these points. Recent applications of spline interpolation in the context of planform dynamics include Fagherazzi, Gabet, and Furbish (2004), Perucca, Camporeale, and Ridolfi (2005), and Camporeale and others (2005); Davis (1986), De Boor (2001), and Cohen, Riesenfeld, and Elber (2001) provide more general background on splines. Following Fagherazzi, Gabet, and Furbish (2004), we fit one spline to the centerline vertices' x coordinates and another to their y coordinates. The resulting splines consist of a series of curved segments connecting adjacent vertices (Fig. 4B), each of which is defined by a cubic polynomial of an (initially arbitrary) parametric coordinate t, which takes on integer values indexing the N_b vertices of the filtered centerline. For a given centerline segment, the two splines take the form

$$\begin{bmatrix} x^{(k)}(t) \\ y^{(k)}(t) \end{bmatrix} = \begin{bmatrix} \alpha_0^{(k)} + \alpha_1^{(k)} t + \alpha_2^{(k)} t^2 + \alpha_3^{(k)} t^3 \\ \beta_0^{(k)} + \beta_1^{(k)} t + \beta_2^{(k)} t^2 + \beta_3^{(k)} t^3 \end{bmatrix} \tag{3}$$

where k indexes the segments of the two splines. Each of the $N_b - 1$ segments has a distinct set of α_p and β_p coefficients (e.g., Davis, 1986), and the pair of polynomials for that segment constitute a vector-valued function

$$\mathbf{r}^{(k)}(t) = x^{(k)}(t)\mathbf{i} + y^{(k)}(t)\mathbf{j} \tag{4}$$

where $\mathbf{i}$ and $\mathbf{j}$ are the basis vectors of the Cartesian coordinate system; $\mathbf{r}^k(t)$ can thus be considered a position vector describing the channel centerline.

Arc Length Calculation

The length of the kth segment of the centerline is then calculated by substituting the the corresponding polynomials from step 3 into the arc length

formula:

$$s^{(k)} = \int_{t_k}^{t_{k+1}} \sqrt{\left(\frac{d}{dt}x^{(k)}(t)\right)^2 + \left(\frac{d}{dt}y^{(k)}(t)\right)^2}\, dt = \int_{t_k}^{t_{k+1}} \|\mathbf{r}^{(k)}(t)\| dt \qquad (5)$$

where t_k and t_{k+1} are the values of the parametric coordinate at the upper and lower end points of the curved centerline segment, assuming the centerline has been digitized from upstream to downstream. As suggested by Fagherazzi, Gabet, and Furbish (2004), the integration is performed via Simpson's rule, which is exact for third-order polynomials (Press and others, 1994). The total streamwise distance from the origin of the centerline to its $k+1$th vertex is given by the cumulative arc length of the preceding (upstream) segments

$$s_k = \sum_{k'=1}^{k} s^{(k')} \qquad (6)$$

Centerline Resampling

This step involves replacing the integer-valued parameter t with the actual streamwise distance s from (5) and (6), creating a new set of points that are regularly spaced along the centerline, and then recomputing a new pair of splines parameterized by s. Computationally, this process involves normalizing the N_b values of s by the total arc length S of the centerline and rescaling them to match the range of the initial parameter t (by multiplying by N_b). Evaluating the initial splines at the resulting values of t produces a new set of centerline vertices with a spacing proportional to the fraction of the total arc length accounted for by each segment of the initial, digitized centerline. These new vertices are then used to produce a second, intermediate set of splines, which are in turn evaluated at a specified number N_d of discretization points evenly (linearly) spaced between 0 and S but rescaled to the range of t. The result is a series of N_d vertices that are approximately regularly spaced along the channel centerline. The final pair of splines is fit to these coordinates and parameterized in terms of the 'true' streamwise distance given by the curvilinear coordinate s:

$$\begin{bmatrix} x^{(l)}(s) \\ y^{(l)}(s) \end{bmatrix} = \begin{bmatrix} \alpha_0^{(l)} + \alpha_1^{(l)} s + \alpha_2^{(l)} s^2 + \alpha_3^{(l)} s^3 \\ \beta_0^{(l)} + \beta_1^{(l)} s + \beta_2^{(l)} s^2 + \beta_3^{(l)} s^3 \end{bmatrix} \qquad (7)$$

where l indexes the $N_d - 1$ segments of these new splines. The new position vector $\mathbf{r}^l(s)$ is defined as in (4) and now describes the centerline as a parametric function

of the s coordinate. The arc length of each centerline segment is also re-calculated from these resampled splines.

Normal Vectors and Centerline Curvature

The splines described in the preceding section are then used to compute the first and second derivatives of the polynomials that define each centerline segment, denoted by x' and x'' and similarly for the y coordinate, and hence the unit tangent vector $\mathbf{T}(s)$, unit normal vector $\mathbf{N}(s)$, and curvature $C(s)$ at the N_d discretization points. The relevant formulae for the vectors and their components are (Gray, 1998)

$$\mathbf{T}(s) = \frac{\mathbf{r}'(s)}{\|\mathbf{r}'(s)\|} = \begin{bmatrix} x_T(s) \\ y_T(s) \end{bmatrix} = \begin{bmatrix} \frac{x'}{\sqrt{(x')^2+(y')^2}} \\ \frac{y'}{\sqrt{(x')^2+(y')^2}} \end{bmatrix} \tag{8}$$

$$\mathbf{N}(s) = \frac{\mathbf{T}'(s)}{\|\mathbf{T}'(s)\|} = \begin{bmatrix} x_N(s) \\ y_N(s) \end{bmatrix} = \begin{bmatrix} \frac{-y'}{\sqrt{(x')^2+(y')^2}} \\ \frac{x'}{\sqrt{(x')^2+(y')^2}} \end{bmatrix} \tag{9}$$

$$C(s) = \frac{\|\mathbf{T}'(s)\|}{\|\mathbf{r}'(s)\|} = \frac{[x'y'' - y'x'']}{[(x')^2 + (y')^2]^{3/2}} \tag{10}$$

where $\|\mathbf{r}\|$ denotes the length of vector $\mathbf{r}$ and the derivatives on the right side refer to a specific spline segment (i.e., $x' = \frac{dx^{(l)}(s)}{ds}$ and $y' = \frac{dy^{(l)}(s)}{ds}$). The tangent vector $\mathbf{T}(s)$ describes the channel direction at each location s along the centerline and points downstream. Similarly, the unit normal vector $\mathbf{N}(s)$ defines the orientation of a channel cross-section (Fig. 4C). To ensure consistency with the sign convention of Smith and McLean (1984) and ensure a right-handed coordinate system with s increasing downstream and z vertically upward, $\mathbf{N}(s)$ points toward the left side of the channel (facing downstream). Note that if the centerline is digitized from downstream to upstream, $\mathbf{N}(s)$ will point toward the right bank, so vertices should be added in an orderly downstream sequence. Although computing the curvature is not part of the transformation *per se*, this important geomorphic variable can be readily obtained at this stage by evaluating the second derivatives of the spline polynomials, and this approach is increasingly used to characterize planform dynamics (e.g., Perucca, Camporeale, and Ridolfi, 2005; Camporeale and others, 2005).

Local Search for In-Channel Data Points

In principle, the (s, n) coordinates of an in-channel observation could be computed at this stage by calculating the distance from that location to each

of the N_d vertices of the discretized centerline. The smallest of these distances would approximate the transverse n coordinate of the observation point and the s value of the centerline vertex at which this minimum occurred would provide the corresponding streamwise coordinate. In practice, however, such a global search strategy would be computationally demanding for large numbers of data and/or centerline discretization points is large. Even if a more efficient nested search routine were used (e.g., Goff and Nordfjord, 2004), a more important problem remains: because only the N_d vertices of the (resampled) centerline are 'candidate' s coordinates, the resolution of the transformation depends on the choice of N_d. A related issue would be positive bias of the n coordinates: the true (minimum) value of n occurs only when the vector connecting the (x, y) observation point to the nearest centerline vertex is parallel to $\mathbf{N}(s)$ for that location along the centerline.

To reduce these effects without resorting to a very large N_d, with a corresponding decrease in computational efficiency, the search for in-channel data points can be restricted using a point-in-polygon operation. First, a series of polygons is created as shown in Figure 4D by connecting successive centerline vertices to points placed a specified distance $n_{\max}$ (set to half the maximum channel width or greater) from the centerline in the direction specified by the unit normal vector $\mathbf{N}(s)$ for each centerline vertex. These two points are then connected to one another to define the outside edge of the polygon, relative to the centerline, and the lines parallel to the normal vectors become the upstream and downstream edges. Because connecting successive centerline vertices directly would create a sliver between the true, curved centerline and the straight inside edge of the resulting polygon, any data points within this region would be missed during the point-in-polygon search operation. To avoid this situation, the curved inner edge of the polygon is approximated by adding vertices to the centerline segment by evaluating the spline functions at evenly spaced s values bracketed by the segment end points. This refinement, essentially a local sub-discretization of the centerline, is not computationally demanding and improves the calculation of s, n coordinates in the next step. Finally, a simple point-in-polygon operation is used to identify all in-channel data points located within each of the polygons in turn. To determine if the current point is inside the current polygon, a line is drawn from the point toward infinity and the number of intersections of this line with the polygon boundary is recorded; if the number of intersections is odd, the point is inside the polygon (Longley and others, 2001). Note that this search procedure could fail in the pathological case of a datum located precisely along the curved centerline, outside the straight line segments connecting the new set of vertices. Even in very tight bends, encountering this problem in practice is highly unlikely as long as the initial discretization of the centerline is sufficiently fine.

Obtaining (s, n) Coordinates from Data Point-Centerline Distances

The local resampling of the centerline to produce additional polygon vertices results in an equivalent number of new, more closely spaced centerline points for computing refined observation-to-centerline distances. The calculation is performed within a loop over the polygons, and only those data points located within the current polygon are considered (Fig. 4E). The distances from a particular datum to each of the resampled centerline vertices on the current segment are computed: the minimum of these distances defines the absolute value of the n coordinate, which will be positive for data points on the left side of the channel, and the value of s at the centerline vertex at which this minimum occurs becomes the s coordinate. While this approach does not eliminate the discretization-related issues mentioned above, the restricted search and local resampling ensure that these effects will be minimal and also improve the efficiency of the algorithm.

Search for Points on the Right Side of the Channel Centerline

The previous two steps assumed a unit normal vector $\mathbf{N}(s)$ pointing from the centerline toward the left bank, and the polygons generated in step 7 thus encompassed only the left half of the channel. To obtain the s,n coordinates of observation locations on the right side of the centerline, create polygons with vertices given by $-n_{\max}\mathbf{N}(s)$ (i.e., oriented opposite the unit normal vector) and repeat the previous two steps. Note that the n coordinates of points located to the right of the centerline are negative.

INVERSE TRANSFORMATION FROM CHANNEL-CENTERED TO GEOGRAPHIC COORDINATES

In situations where working within a channel-fitted coordinate system is most convenient, an inverse transformation is required to relate this curvilinear system to a Cartesian frame of reference. For example, although highly efficient (fast) algorithms for spatial stochastic simulation on regular grids are available (e.g., Chiles and Delfiner, 1999), such a grid generated in Cartesian space and then transformed to s, n coordinates will no longer be regular due to the curvature of the channel. An alternative approach in this case is to perform the simulation directly on a regular (s, n) grid and then transform the resulting realizations to the corresponding Cartesian coordinates. If the (x, y) coordinates of a digitized centerline are provided as input, this transformation can be achieved via the approach used by Smith and McLean (1984) to relate their fluid mechanical theory, derived in terms of s and n, to a geographic coordinate system. Included in the appendix of

their paper are analytical expressions that relate the Cartesian coordinates (x, y) of a point inside the channel to the known coordinates $(x_c(s), y_c(s))$ of the centerline, the n coordinate, and the derivatives of the Cartesian coordinates of the centerline with respect to streamwise distance:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x_c(s) - n\frac{dy_c(s)}{ds} \\ \\ y_c(s) + n\frac{dx_c(s)}{ds} \end{bmatrix} \tag{11}$$

In principle, these functions could be evaluated directly using the polynomials that comprise the splines fit to the centerline coordinates [Eq. (7)]. Because these splines are sensitive to the initial digitization of the centerline and to decisions made during the filtering and discretization process, we adopt a finite difference approach instead. The inverse transformation proceeds as follows:

1. Define a centerline with N_d approximately regularly spaced vertices as described in steps 1– 5 above. Because a finite difference approximation is used in the inverse transformation, a relatively fine discretization (i.e., large N_d) should be used. The spline functions themselves are not required, but the (x, y) coordinates of the resampled centerline and the cumulative arc length of each vertex are needed to relate arbitrary (s, n) coordinates to their Cartesian counterparts.
2. Compute the differences between the s coordinate in question and the s values at each of the N_d centerline vertices, sort these differences in ascending order, and retain the first two entries s_1 and s_2, where the subscripts denote the order in the sorted array of s coordinate differences. The Cartesian coordinates of the centerline vertex closest to the s coordinate to be transformed are given by $x_c(s_1)$ and $y_c(s_1)$ and used in Equation (11). The coordinates of the second closest vertex are also needed to compute finite difference approximations to the derivatives appearing in these expressions.
3. The derivatives in Equation (11) are approximated by computing the change in x_c and y_c between successive centerline vertices

$$\frac{dx_c}{ds} \approx \frac{x_c(s_1) - x_c(s_2)}{s_1 - s_2} \frac{dy_c}{ds} \approx \frac{y_c(s_1) - y_c(s_2)}{s_1 - s_2} \tag{12}$$

4. Finally, these differences are inserted into Equation (11) to determine the x, y coordinates of the s, n point in question.

This algorithm, together with the forward transformation outlined previously, provides a consistent means of converting observational data, theoretical results, and/or model outputs back and forth between geographic and channel-centered

coordinate systems, affording a great deal of flexibility in the representation and analysis of meandering rivers. Our current MATLAB implementation also indicates that these procedures are computationally efficient, transforming grids of over 3000 nodes in a matter of seconds on an ordinary desktop computer. The sensitivity and accuracy of the algorithms are evaluated in the following sections.

RESULTS

We assessed the performance of these coordinate transformation procedures using both field data from Soda Butte Creek, a gravel-bed river in Yellowstone National Park, and simulated data derived from analytically-defined centerlines. The first and most important stage in either transformation is the definition of the channel centerline. The spline functions fit to the coordinates of the centerline vertices determine both the arc length of each segment (i.e., the s coordinate) and the orientation of the unit normal vectors, which affect the n coordinate. In the following subsection, we use field data to examine two important factors that will affect centerline definition in natural, gravel-bed rivers: the initial digitization of centerline vertices and the subsequent filtering of these coordinates. We then use a series of analytically-defined centerlines and simulated, known coordinates to evaluate the accuracy of the transformations and assess their sensitivity to channel geometry and centerline definition.

Centerline Definition from Field Data

We used a topographic survey from Soda Butte Creek to: (i) determine whether a channel centerline could be consistently digitized from typical field data, and (ii) to select an appropriate set of filter parameters for smoothing the initial centerline. As part of our ongoing research on spatial patterns of river morphology and hydraulics, we surveyed the 375-m reach shown in Figures 3 and 4A, obtaining 4081 point measurements of bed elevation using a Leica TCA1105+ robotic total station. Distinct breaks in slope (i.e., top of bank, bar fronts, etc.) were surveyed by connecting sequences of points to form explicit topographic breaklines (e.g., Lane, 1998; Keim, Skaugset, and Bateman, 1999). The breaklines representing the channel banks were then used to define vertices along the centerline with the midpoint method described above. To evaluate the objectivity and reproducibility of this method, ten independent sets of centerline vertices were digitized by hand within a GIS. A separate set of bank points was selected for each of these digitizations, resulting in ten different sets of vertices; the number of points ranged from 54 to 62 for this $\sim$25 m-wide channel. Figure 5A indicates that the ten replicate centerlines were not noticeably different from one another at the reach-scale; the inset magnifying a segment downstream of the bend apex shows that

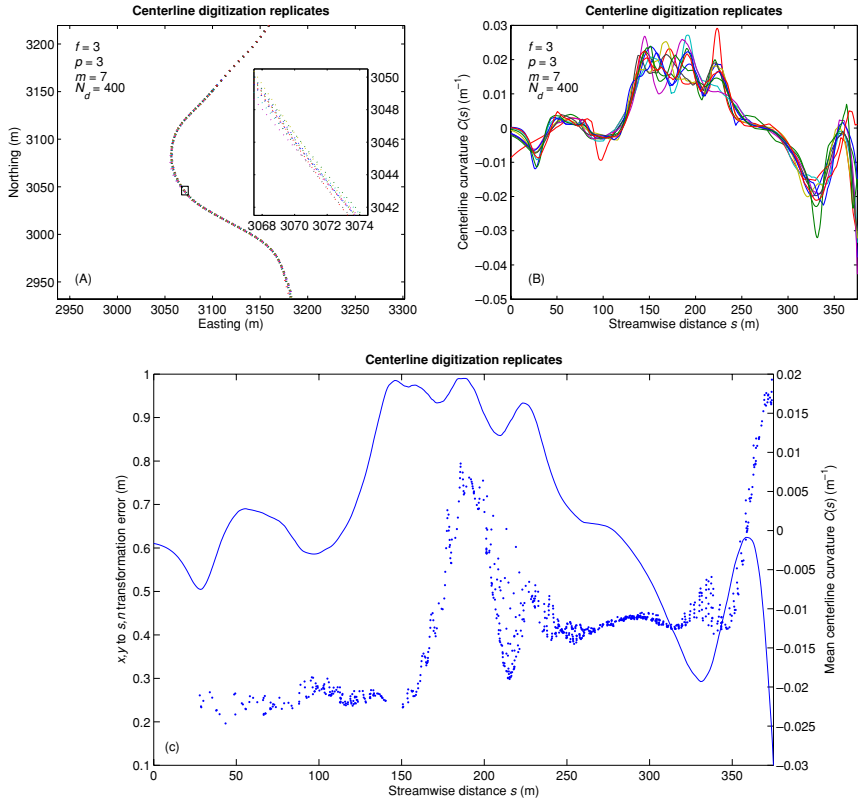

Figure 5. Sensitivity analysis of centerline definition for Soda Butte Creek. (A) ten replicate digitizations of the centerline, with the location of the inset indicated by the small box; (B) streamwise curvature series for the ten replicate centerlines, each filtered and discretized with the parameters listed; (C) mean forward transformation errors (symbols) and centerline curvature values (solid curve) for the ten replicates.

none of the replicates diverge by more than a meter. The curvature series in Figure 5B provides a more sensitive test of the consistency of centerline definition because curvature estimates involve first- and second-order derivatives and are thus strongly influenced by the selection of centerline vertices. This plot indicates that, after applying the same filter to each set of initial vertices, the ten replicates are in fairly close agreement for the first 145 m but then diverge through the bend apex in the area of highest curvature. Agreement improves coming out of the bend from 225–300 m and then deteriorates at the lower end of the reach, most likely due to the difficulty of extrapolating the centerline beyond the area for which bank breaklines were surveyed.

To investigate the effect of centerline digitization error on the forward transformation from geographic to channel-centered coordinates, we transformed our survey data using each of the ten replicate centerlines in turn, computed the variances of the resulting s and n coordinates for each point, and combined these into a composite (x, y) to (s, n) transformation error by taking the square root of the sum of the variances. These errors are plotted as a function of streamwise distance (the mean s coordinate for the ten replicates) in Figure 5C, along with the mean curvature of the ten centerlines. The transformation error remains steady at about 0.25 m down to about 150 m and then increases abruptly to 0.75 m in the area of higher curvature at the bend apex. The error declines again past the apex and remains steady at about 0.4 m before increasing again at the lower end of the reach. The higher error of this second plateau from 250–325 m downstream is a consequence of the discrepancy among centerlines in the bend apex, which causes the s coordinate to vary among replicates even if the centerlines converge again downstream and are associated with similar n coordinates for a given point. Nevertheless, these results suggest that, for this reach of Soda Butte Creek, the centerline definition procedure was fairly robust, and forward transformation errors were on the order of 2–4% of the mean channel width. Curvature estimates, however, were much more sensitive to the initial digitization of vertices in areas of high curvature.

Once an initial sequence of centerline points has been identified, the second factor influencing centerline definition and coordinate transformation is the filtering of the coordinates of these centerline points (Fagherazzi, Gabet, and Furbish, 2004). Figure 6 illustrates the effects of the three parameters used with the Savitzky-Golay filter: the number f of times the filter is applied, the order p of the filtering polynomial, and the number m of points included within the filter window; the number N_d of points used to discretize the centerline is fixed at 300. The filtering process is critical because vertex-to-vertex oscillations in the centerline trace will affect the calculation of derivatives used to define normal vectors and estimate curvature: if filtering is insufficient, these results will reflect the noise inherent to the initial digitization; if filtering is excessive, arc length and curvature will tend to be underestimated. Figure 6A and B indicate that a single application of the filter fails to provide adequate smoothing of the centerline and results in a noisy curvature series, but applying the filter a second time eliminates most of these high-frequency fluctuations; additional filter applications appear to have little effect. Similarly, whereas a higher-order ($p = 4$) filtering polynomial follows the trace of the initial vertices too closely and produces several curvature artifacts, the desired smoothing of the centerline (i.e., removal of spikes and/or abrupt reversals in the sign of the curvature) can be achieved by using cubic or quadratic functions. The inset of Figure 6E suggests that window size m can have a moderate effect on the location of the filtered centerline, and Figure 6F indicates that using a filter window that is too small will result in a noisy curvature series.

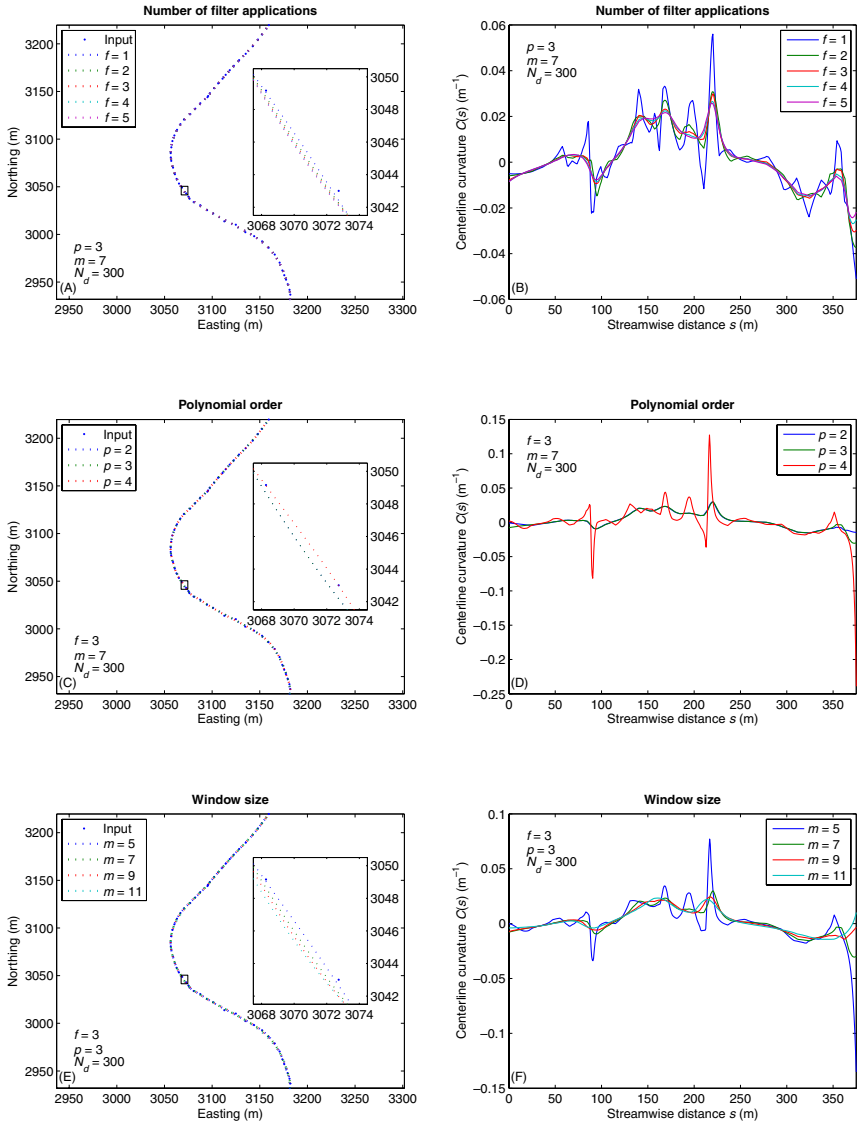

Figure 6. Sensitivity of centerline definition (left column) and curvature estimation (right column) to filter parameters for Soda Butte Creek. f denotes the number of filter applications, p the order of the polynomial used in the Savitzky-Golay filter, and m the number of points included in the filter window; N_d is the number of discretization points. Each curve in the left column represents a different filtered centerline produced from the same input vertices but using different values of the indicated filter parameter, and the corresponding curvature series for these centerlines are displayed in the right column.

On the basis of these analyses, we selected $f = 3$, $p = 3$, and $m = 7$ as the optimal set of parameters for this reach of Soda Butte Creek, whereas Fagherazzi, Gabet, and Furbish (2004) used a larger window and a higher-order polynomial to define the centerline of a smaller tidal channel. Determining an appropriate degree of filtering is thus a subjective exercise, and the choice of filter parameters is likely to be site-specific, depending on the sinuosity and width of the channel, the method of initial digitization, and the data requirements of the particular study. One useful criterion is the number of zero-crossings (i.e., inflection points) in the curvature series, which should equal the number of bends along the channel because the curvature changes sign only once for each bend. Comparing the original and filtered centerlines can serve as another check on the filtering process, which should seek to achieve "a reasonable balance between reduction of the high frequency noise and preservation of curvature peaks" (Fagherazzi, Gabet, and Furbish, 2004, p. 301). In any case, the results summarized in Figure 6 indicate that the filtering process is an important step in both curvature estimation and coordinate transformation, and we recommend performing this type of sensitivity analysis to guide the selection of a suitable filtering procedure.

Accuracy Assessment with Analytically-Defined Centerlines and Simulated Coordinates

We evaluated the performance of the coordinate transformation procedures by applying them to analytically-defined centerlines and simulated data for which both the (x, y) and (s, n) coordinates were known. Comparing these coordinates to those produced by our algorithms allowed us to assess the accuracy of the transformations and examine the effects of channel geometry, centerline digitization, and discretization. To conduct these tests, we used Langbein and Leopold's (1966) classic model for the planform of meander bends as a sine-generated curve, given by:

$$\phi = \omega \sin(2\pi s/\lambda) \tag{13}$$

where ϕ is the angle between the tangent to the centerline and the valley axis, ω is the maximum value of ϕ (at the point where the channel crosses the valley axis and begins to curve in the opposite direction), s is the streamwise distance along the centerline, and λ is the meander wavelength measured along the the centerline. Although the sine-generated curve does not adequately describe long series of meander bends (Ferguson, 1976), Equation (13) provides a reasonable approximation for individual bends and is widely used in fluid mechanical models of meandering channels (e.g., Johannesson and Parker, 1989). In the present context, the sine-generated curve is appealing because centerlines of varying dimensions and sinuosities can be generated by varying the scale (λ) and shape (ϕ)

parameters. The Cartesian coordinates of vertices along a sine-generated centerline are given by:

$$\begin{bmatrix} x^{(k+1)}(s) \\ y^{(k+1)}(s) \end{bmatrix} = \begin{bmatrix} x^{(k)}(s) + ds\cos(\phi(s)) \\ y^{(k)}(s) + ds\sin(\phi(s)) \end{bmatrix} \tag{14}$$

where ds denotes an arc length increment between successive vertices and the centerline begins at an arbitrary origin ($x^{(0)}$, $y^{(0)}$).

We generated points with known x, y and s, n coordinates for each s increment along the sine-generated curve by: (i) defining the absolute value of the n coordinate by drawing a random number from $\sim U[0,w/2]$, where w is the specified channel width and $\sim U[a, b]$ denotes a uniform distribution in the closed interval $[a, b]$; (ii)assigning approximately half of the simulated points to the right side of the channel by multiplying the initial n coordinate by -1 if a second random number drawn from $\sim U[0, 1]$ was less than 0.5; and (iii) calculating the corresponding Cartesian coordinates as:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x^{(k)}(s) - n\sin(\phi(s)) \\ y^{(k)}(s) + n\cos(\phi(s)) \end{bmatrix} \tag{15}$$

This expression is analogous to Equation (A2) from the appendix of Smith and McLean (1984), but ϕ is the complement of the angle appearing in their relations.

To summarize the accuracy of our coordinate transformation algorithms, we calculated the Euclidean distance between the simulated (true) and transformed coordinates for each point for both the forward ((x, y) to (s, n)) and inverse ((s, n) to (x, y)) transformations. The three primary factors influencing transformation accuracy are the geometry of the channel (i.e., the ω and λ parameters of the sine-generated curve, along with the channel width w), the initial sampling (i.e., digitization) of the centerline, and the number N_d of discretization points used to parameterize the centerline splines. Because the sine-generated curve repeats after one period, we examined a single bend with a fixed arc length λ of 400 m and a constant width w of 20 m. By varying ω, we produced centerlines with varying degrees of curvature; the curvature at any location s along the centerline is given by:

$$C(s) = \frac{d\phi}{ds} = \frac{2\pi\omega}{\lambda}\cos\left(\frac{2\pi s}{\lambda}\right) \tag{16}$$

which takes on a maximum value of $C(s) = 2\pi\omega/\lambda$. We used ω values from 0 to 90° in steps of 15°, corresponding to maximum curvatures of 0–0.0247 m^{-1}

for $\lambda = 400$ m. The vertices of these simulated centerlines were generated at streamwise increments Δs ranging from 1 to 10 m in steps of 1 m. The number of discretization points used in the transformation algorithms varied from 100 to 1000 in steps of 50.

The results of these sensitivity analyses are summarized in Figure 7. Both the mean (x, y) to (s, n) transformation error and the variance of these errors increase steadily with curvature (i.e., higher ω for a fixed λ in Equation(13)), but remain small (0.04 m relative to a channel width of 20 m) even for the sharpest bend that can be produced from a sine-generated curve (Fig. 7A). Similarly, the forward transformation error and its variance are directly proportional to Δs, the spacing between digitized centerline vertices. As Δs increases from 5 to 50% of the channel width, the initial vertices provide a coarser approximation to the true geometry of the sine-generated curve, which adversely affects spline interpolation of the centerline and subsequent arc length and normal vector calculations. For a moderately curved channel ($\omega = 45°$), the discretization of the centerline has a relatively minor impact on the forward transformation, with mean errors decreasing from 0.03 m to 0.02 m as d increases from 50 to 350 and remaining at about 0.02 m for greater d.

Figure 7D indicates that the inverse transformation is relatively insensitive to curvature, with mean errors increasing only slightly above 0.1 m as ω increases from 0 to 90°. Even for the limiting case of a straight channel ($\omega = 0$), the inverse transformation is subject to error due to the discrete nature of the inverse transformation algorithm. The second step in this procedure involves calculating the differences between the s coordinate (point B in Figure 8A) of the point (A) to be transformed and the s values of each vertex of the discretized centerline (points C and D). For a straight channel, the s coordinate of the nearest vertex (point G in Figure 8B) will be smaller than that of the point (E) to be transformed in some cases and larger in others (such as that shown in Fig. 8B), biasing the transformed (x, y) coordinates upstream or downstream, respectively. This effect is also evident in Figures 7E and F, which show that the mean (s, n) to (x, y) transformation error generally increases with Δs and decreases exponentially as N_d increases. In both of these plots, the errors are conspicuously lower at specific values of Δs (2, 4, and to a lesser degree 8) and N_d (400 and 800) that are related to the meander wavelength λ, which was fixed at 400 m in these analyses. When the spacing of initial vertices is an even factor of the meander wavelength, more of the final, discretized vertices of the centerline correspond to the simulated s coordinates and the mean transformation error is lower. Similarly, when the number of discretization points used to describe the centerline is an even multiple of the meander wavelength, the s coordinates of these vertices match those of the simulated data points being transformed and errors are very small. The opposite occurs when N_d is 200 or 600 for $\lambda = 400$: the variance of the transformation error is noticeably greater because the vertices of the centerline are out of phase

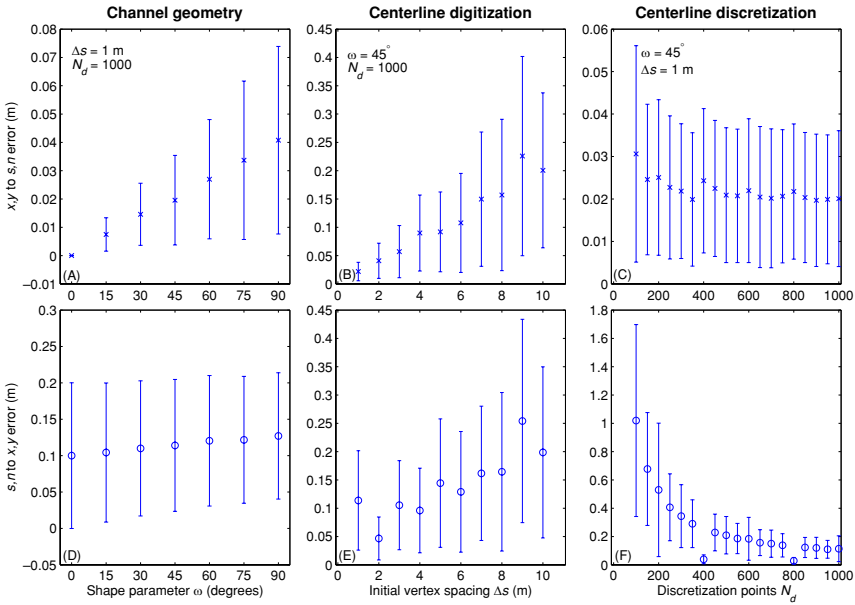

Figure 7. Coordinate system transformation sensitivity analysis using sine-generated centerlines and simulated geographic and channel-centered coordinates. Forward and inverse transformation errors are plotted in the top and bottom rows, respectively. From left to right, the columns illustrate the effects of channel geometry (curvature), initial digitization of the centerline, and centerline discretization, respectively. The values of the two parameters held constant in each column are shown in the top row, and the vertical bars represent the variance of the transformation errors and correspond to ± one standard deviation.

with the simulated coordinates to be transformed, such that approximately half of these points will be assigned to vertices upstream of their true positions and half to vertices downstream. The extreme case is shown in Figure 8A, where the s coordinate of point A is equidistant between the centerline vertices at points C and D such that $\bar{BC} = \bar{CD}$. For intermediate N_d (e.g., 250), the mean transformation error is similar, but the variance of the errors is reduced because more points will be assigned to either the upstream or the downstream centerline vertex rather than being approximately evenly distributed between the two. This effect could only be discerned because the s coordinates we simulated were regularly spaced and integer-valued, but more typical, irregularly distributed data will still be subject to some degree of discretization-related transformation error.

The forward and inverse transformation errors for individual points are shown in Figure 9, which illustrates the effect of curvature on transformation accuracy. Both the (x, y) to (s, n) and (s, n) to (x, y) transformation errors are greatest in the bend apex and at the upper and lower limits of the simulated reach, which

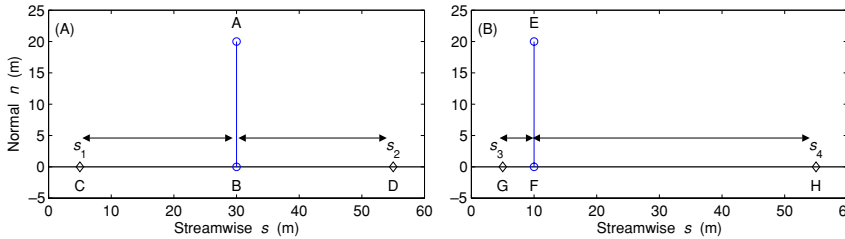

Figure 8. Schematic illustration of the effect of discretization on inverse transformation error. (A) transformation of point A is ambiguous because the two nearest vertices, C and D, of the discretized centerline are equidistant from the point, B, with the true s coordinate; (B) when the point E to be transformed is closer to a vertex G, the transformation error is reduced.

are in turn the apices of the next bends upstream and downstream. Conversely, the coordinate transformations are most accurate in the straight segments between bends of opposite curvature, and errors decrease steadily from the apex of one bend downstream to the inflection point before increasing again toward the next bend along the channel. The strong, direct relationship between curvature and transformation error is again a consequence of discretization. Although the centerline in Figure 9 is defined by a smoothly-varying, continuous function, approximating this sine-generated curve with splines evaluated at discrete points limits the accuracy with which locations can be defined relative to the true centerline. Even if each spline segment is re-evaluated at a finer discretization (step 7) before calculating distances from data points to centerline vertices (step 8), the accuracy and precision that can be achieved remains limited because the s values of these vertices are the only potential s coordinates that the points to be transformed can be assigned. A secondary effect involves the n coordinates, which will tend to be biased upward in absolute value (i.e., away from the centerline) because the distance from a datum to the centerline achieves its true, minimum value only when the point in question lies upon one of the normal vectors drawn from the centerline vertices—the farther a point is from a normal (i.e., the coarser the discretization of the centerline), the larger the bias. This source of error is relatively minor in comparison to that of the s coordinate: for the bend in Figure 9, the s component of the forward transformation error vector is two orders of magnitude greater than the n component.

The relationships between curvature, discretization, and the resulting forward transformation errors are illustrated schematically in Figure 10. In Figure 10A, the true s coordinate for point A is that of point B, and the magnitude of vector $\vec{AB}$, which is normal to the centerline, is the correct n coordinate. In the discrete setting, however, the assignment of an s coordinate is based on the distances from point A to centerline vertices—points C and D and points E and F represent finer and coarser (by a factor of two) discretizations of the centerline, respectively. For a given

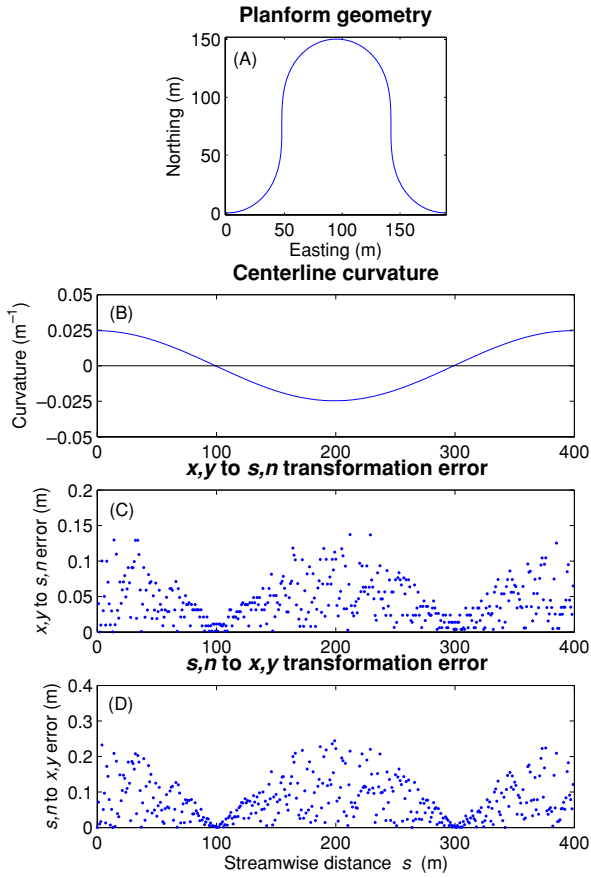

Figure 9. Coordinate system transformation error in relation to channel curvature for a sine-generated curve with $\omega = 90°$, $\lambda = 400$ m, $\Delta s = 1$ m, and $N_d = 400$. (A) planform geometry; (B) centerline curvature; (C) forward transformation error; and (D) inverse transformation error.

curvature, finer discretization results in a more accurate transformation because, for example, the s value of point C is closer to that of point B and $\bar{AC} < \bar{AD}$, resulting in a less biased n coordinate relative to its true value given by $\bar{AB}$. In Figure 10B the centerline curvature is half that of the bend in Figure 10A, but the discretization is the same – that is, the arc length between vertices I and J is equal to that between vertices C and D and likewise for the coarser discretization. The n coordinate bias is less in Figure 10B because the curvature is smaller, but the s component of the transformation error is the same because the two bends

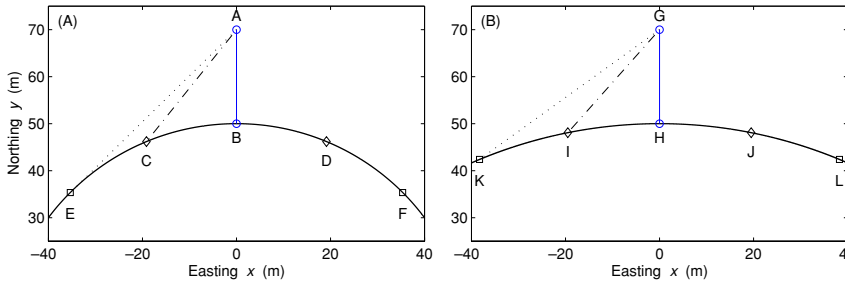

Figure 10. Schematic illustration of the effects of curvature and discretization on forward transformation error. (A) the true s coordinate of point A is that of point B, but the nearest centerline vertices are C and E for fine and coarse discretizations, respectively; note that the positive bias of the n coordinate is greater for the coarser discretization. (B) as in A, but in a bend of smaller curvature with the same discretization.

were discretized with vertices spaced at equal arc length intervals. Note that if the total length of the bends were equal and the same number of discretization points were used for each, the arc length between vertices, and thus the s component of the transformation error, would be smaller in the bend of lower curvature. This geometric explanation accounts for the observed direct (inverse) relationship between transformation error and curvature (number of discretization points).

DISCUSSION AND CONCLUSION

The algorithms presented herein provide an explicit, theoretically-grounded means of relating geographic and channel-centered coordinate systems for meandering river channels. These methods are specifically intended for application to reach-scale studies in modern rivers, filling a gap between the coordinate transformations used in theoretical fluid mechanics (Smith and McLean, 1984) and small-scale field studies (Dietrich and Smith, 1983, 1984) and those described in the application-oriented geostatistical literature (Deutsch and Wang, 1996; Goff and Nordfjord, 2004). Merwade, Maidment, and Hodges (2005) have also developed a GIS-based procedure for transforming coordinates from a Cartesian to an orthogonal curvilinear frame of reference, but our approach is not associated with a particular software package and thus offers greater flexibility. Similarly, while our inverse transformation algorithm uses equations presented by Smith and McLean (1984), we build upon their work by explicitly describing how a digitized centerline can be used to transform (s, n) coordinates to their geographic equivalents. Taken together, our forward and inverse coordinate transformations thus provide a robust, general framework for the spatial referencing of meandering rivers. The transformations allow for conversion between the Cartesian system, within which data are collected and results are visualized, and the channel-centered

system for which fluid mechanical theories have been developed and within which methods of spatial analysis can be more readily applied.

Because such a framework is only useful if the coordinate transformations are of sufficient accuracy and precision for specific applications, we have evaluated the performance of our algorithms using both field and simulated data. The sensitivity analysis and accuracy assessment presented above indicate that the most important source of error for both the forward and inverse transformations is the definition of the centerline itself. Digitization, filtering, and discretization all influence centerline definition and hence transformation accuracy to an extent that also depends on the planform geometry of the channel. For our field example from Soda Butte Creek, replicate centerlines digitized using the bankfull mid-point method were in close agreement with one another and (x, y) to (s, n) transformation errors ranged from 0.2 m to 0.8 m for this 25-m wide channel, with the largest errors occurring in the area of greatest curvature. Curvature series for the replicate centerlines confirmed that initial digitization was least certain through the apex of the bend and indicated that curvature estimates in this area were sensitive to the selection of centerline vertices. Obtaining a smoothed representation of the centerline that reduces these point-to-point oscillations is thus an important step, and we have shown how sensitivity analyses can be used to select suitable (i.e., site-specific) parameters for filtering the initial digitized vertices. Greater smoothing can be achieved by increasing the number of filter applications, using a lower-order polynomial, and/or including more points within the filter window. Determining an appropriate degree of centerline smoothing remains a subjective exercise, however, and in practice this decision will depend on channel characteristics, such as width and sinuosity, and the scale and objectives of the study.

Once the centerline has been defined, the accuracy and precision of the coordinate transformations is primarily a function of curvature and discretization. Our investigation of analytically-defined centerlines and simulated coordinates indicates that, even for highly sinuous channels, transformation errors are on the order of cm or tens of cm, within the precision of typical field survey methods and thus adequate for most practical applications. The locational error that does exist stems from the discrete nature of the algorithms: an (x, y) point can only be assigned an s coordinate that corresponds to a vertex of the discretized centerline, and an (s, n) point must be associated with a specific pair of centerline vertices to compute the finite differences in Equation (12) and transform the point's coordinates to a geographic frame of reference. For the forward transformation, most of this discretization-related transformation error is associated with the s coordinate, but the n coordinate is also slightly positively biased because the normal vectors emanating from the discretized vertices of the centerline will seldom, if ever, pass exactly through the (x, y) point to be transformed. These effects are most pronounced where curvature is greatest, implying that transformation accuracy varies spatially as a function of channel morphology: errors will be minimal in

relatively straight riffles and greatest in pools located along the concave bank of tight bends where the radius of curvature (R, defined as the inverse of the curvature) approaches the channel width w. For most rivers, R/w is between 2 and 3 (Williams, 1986), and Smith and McLean (1984) point out that the (s, n) coordinate system is guaranteed to be single-valued within the channel as long as $R > w/2$, which will always be the case in natural channels. For points outside the channel on the convex bank, the transformation can no longer be assumed to be unique once the distance from the centerline exceeds the local radius of curvature. Conversely, on the concave bank the resolution of the transformation will become increasingly coarse as the distance from the centerline increases and the normal vectors diverge.

The adverse effects of curvature on transformation accuracy can be partially compensated for by using a finer discretization (higher N_d) for the centerline, or by implementing an adaptive discretization based on local centerline curvature. The algorithm's polygon-based search strategy ensures that adding discretization points need not be computationally prohibitive because distances from data points to (resampled) centerline vertices are only computed for points located within polygons defined by normal vectors originating from successive vertices. For the inverse transformation of a regular (s, n) grid to (x, y) coordinates, transformation error can be minimized by adjusting the discretization of the centerline to match the streamwise spacing between grid nodes. For a given centerline, most of the transformation error results from the association of an arbitrary (s, n) point with a pair of centerline vertices, with the worst-case scenario occurring when the point is approximately halfway between two vertices. To minimize these discretization-related errors, one can either select a grid node spacing based on the length and discretization of the centerline or select N_d such that S/N_d is equal to the grid node spacing, where S is the total length of the centerline.

An (s, n) grid transformed in this manner will, however, have a variable, curvature-dependent resolution in the Cartesian frame of reference, with more closely spaced nodes on the convex than the concave side of the centerline (Fig. 11). In terms of channel morphology, this implies that resolution will be coarser in pools located near the outer bank of a meander than over the point bar along the inside of the bend. An alternative approach, analogous to that of Deutsch and Wang (1996), would be to generate a regular grid in the geographic space, transform the grid nodes to (irregular) (s, n) coordinates, perform the analysis within the curvilinear reference frame, and then back-transform the results to the original (x, y) coordinates. The choice between performing the analysis on a regular (s, n) grid or working with a Cartesian grid that is irregular in the curvilinear space depends on the nature of the problem and the computational methods used to address such problems. In any case, the forward and inverse coordinate transformation algorithms ensure that either route is possible, if not entirely feasible.

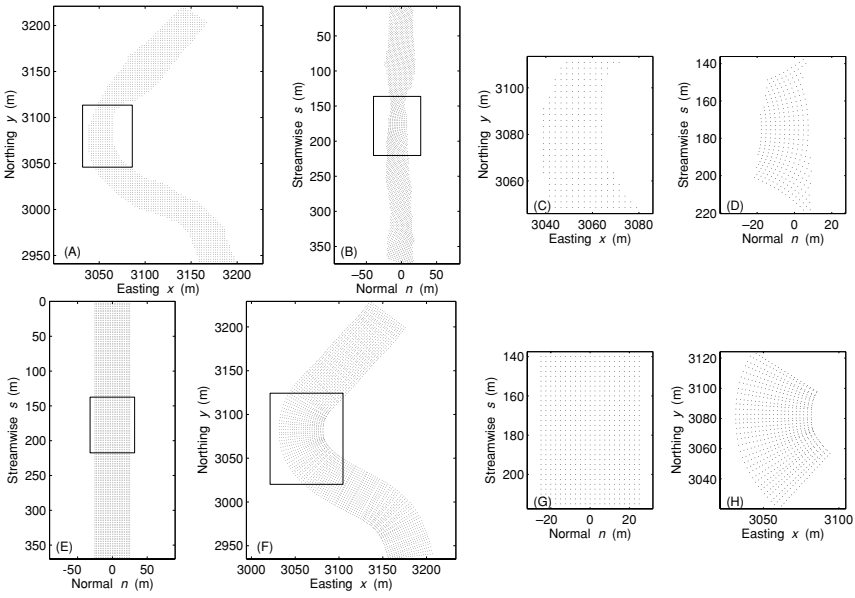

Figure 11. Contrast between a regular geographic grid (A and C) transformed to orthogonal curvilinear coordinates (B and D) and a regular (s, n) grid (E and G) transformed to (x, y) coordinates (F and H).

The approach to spatial referencing in fluvial systems presented herein could be used to address interdisciplinary scientific issues spanning a range of spatial scales. Aquatic ecologists, for example, have recently begun to explore the application of geostatistical methods to stream ecosystems, using along-channel distances as a more natural, "as-the-fish-swims" metric (Ganio, Torgersen, and Gresswell, 2005). Gardner, Sullivan, and Lembo (2003, 2004) also found that incorporating information on the structure of the drainage network improved spatial prediction of stream temperatures. In a hydrological context, Monestiez and others (2005) have developed more sophisticated, topology-based methods of describing network connectivity and calculating distances within networks, and a similar topological-kriging system has been proposed for prediction in ungauged basins (Skoein, Merz, and Bloschl, 2005). Whereas these studies have emphasized one-dimensional, catchment-scale models of the fluvial system, more detailed, reach-scale studies require a two-dimensional approach. The coordinate transformations described in this paper allow geomorphic and ecological processes to be examined in a logical, channel-centered frame of reference, and our ongoing research involves quantifying spatial patterns of hydraulic and sedimentary variables. Watershed- and channel-scale processes are, of course, linked, and applying the spatial referencing framework presented herein to a network of digitized

centerlines would allow these different scales of inquiry to be integrated by adding a second, lateral dimension to current one-dimensional representations.

ACKNOWLEDGMENTS

Bob Crabtree provided both logistical support for field data collection and the image data in Figs. 1 and 3. Steve Clayton and Peter Goodwin of the University of Idaho's Center for Ecohydraulics Research generously loaned survey equipment. Kyle Legleiter, Oren Gersten, Joe Young, and Annie Toth provided invaluable field assistance. Michael Goodchild's advice helped refine our algorithms, and the thoughtful comments of John Goff improved our paper. This research was funded by a Graduate Research Fellowship from the National Science Foundation and a student research grant from the Geological Society of America, and was made possible through the cooperation of the National Park Service.

REFERENCES

Barabas, N., Goovaerts, P., and Adriaens, P., 2001, Geostatistical assessment and validation of uncertainty for three-dimensional dioxin data from sediments in an estuarine river: Environ. Sci. Technol., v. 35, no. 16, p. 3294–3301.

Camporeale, C., Perona, P., Porporato, A., and Ridolfi, L., 2005, On the long-term behavior of meandering rivers: Water Resour. Res., v. 41, no. W12403, doi:10.1029/2005WR004109.

Chiles, J., and Delfiner, P., 1999, Geostatistics: modeling spatial uncertainty: Wiley, New York, 695 p.

Cohen, E., Riesenfeld, R., and Elber, G., 2001, Geometric modeling with splines: an introduction: AK Peters, Natick, MA, 616 p.

Davis, J., 1986, Statistics and data analysis in geology: Wiley, New York, 646 p.

De Boor, C., 2001, A practical guide to splines: Springer-Verlag, New York, 372 p.

Deutsch, C. V., 2002, Geostatistical reservoir modeling: Oxford University Press, New York, 400 p.

Deutsch, C. V., and Tran, T., 2002, FLUVSIM: A program for object-based stochastic modeling of fluvial depositional systems: Comput. Geosci., v. 28, no. 4, p. 525–535.

Deutsch, C. V., and Wang, L. B., 1996, Hierarchical object-based stochastic modeling of fluvial reservoirs: Math. Geol., v. 28, no. 7, p. 857–880.

Dietrich, W. E., 1987, Mechanics of flow and sediment transport in river bends, *in* Richards, K. S., ed., River channels: environment and process: Blackwell, New York, p. 179–227.

Dietrich, W. E., and Smith, D. J., 1983, Influence of the point bar on flow through curved channels: Water Resour. Res., v. 19, no. 5, p. 1173–1192.

Dietrich, W. E., and Smith, J. D., 1984, Bed-load transport in a river meander: Water Resour. Res., v. 20, no. 10, p. 1355–1380.

Environmental Systems Research Institute, 2004, ArcGIS 9—Editing in ArcMap: ESRI Press, Redlands, CA, 504 p.

Fagherazzi, S., Gabet, E. J., and Furbish, D. J., 2004, The effect of bidirectional flow on tidal channel planforms: Earth Surf. Proc. Land., v. 29, no. 3, p. 295–309.

Ferguson, R. I., 1976, Disturbed periodic model for river meanders: Earth Surf. Proc. Land., v. 1, no. 4, p. 337–347.

Ganio, L. M., Torgersen, C. E., and Gresswell, R. E., 2005, A geostatistical approach for describing spatial pattern in stream networks: Front. Ecol. Environ., v. 3, no. 3, p. 138–144.

Gardner, B., and Sullivan, P. J., 2004, Spatial and temporal stream temperature prediction: modeling nonstationary temporal covariance structures: Water Resour. Res., v. 40, no. W01102, doi:10.1029/2003WR002511.

Gardner, B., Sullivan, P. J., and Lembo, A. J., 2003, Predicting stream temperatures: geostatistical model comparison using alternative distance metrics: Can. J. Fish. Aquat. Sci., v. 60, no. 3, p. 344–351.

Goff, J. A., and Nordfjord, S., 2004, Interpolation of fluvial morphology using channel-oriented coordinate transformation: a case study from the New Jersey shelf: Math. Geol., v. 36, no. 6, p. 643–658.

Gray, A., 1998: Modern differential geometry of curves and surfaces with Mathematica: CRC Press, Boca Raton, FL, 1053 p.

Hamming, R., 1983, Digital filters: Prentice Hall, Englewood Cliffs, NJ, 257 p.

Hooke, R., 1975, Distribution of sediment transport and shear-stress in a meander bend: J. Geol., v. 83, no. 5, p. 543–565.

Johannesson, H., and Parker, G., 1989, Linear theory of river meandering, *in* Ikeda, S., and Parker, G., eds., River meandering: American Geophysical Union, Washington, D.C., p. 181–214.

Keim, R. F., Skaugset, A. E., and ateman, D. S., 1999, Digital terrain modeling of small stream channels with a total-station theodolite: Adv. Water Resour., v. 23, no. 1, p. 41–48.

Lane, S. N., 1998, The use of digital terrain modelling in the understanding of dynamic river channel systems, *in* Lane, S. N., Richards, K., and Chandler, J. H., eds., Landform monitoring, modelling and analysis: John Wiley and Sons, New York, p. 311–342.

Langbein, W., and Leopold, L. B., 1966, River meanders—theory of minimum variance, U.S. Geological Survey Professional Paper 422H.

Little, L. S., Edwards, D., and Porter, D. E., 1997, Kriging in estuaries: as the crow flies, or as the fish swims?: J. Experimen. Mar. Biol. Ecol., v. 213, no. 1, p. 1–11.

Løland, A., and Høst, G., 2003, Spatial covariance modelling in a complex coastal domain by multidimensional scaling: Environmetrics, v. 14, no. 3, p. 307–321.

Longley, P. A., Goodchild, M. F., Maguire, D. J., and Rhind, D. W., 2001, Geographic information systems and science: Wiley, Chichester, UK, 454 p.

Merwade, V. M., Maidment, D. R., and Hodges, B. R., 2005, Geospatial representation of river channels: J. Hydrol. Eng., v. 10, no. 3, p. 243–251.

Monestiez, P., Bailly, J.-S., Lagacherie, P., and Voltz, M., 2005, Geostatistical modelling of spatial processes on directed trees: application to fluvisol extent: Geoderma, v. 128, no. 3–4, p. 179–191.

Nelson, J. M., Bennett, S. J., and Wiele, S. M., 2003, Flow and sediment transport modeling, *in* Kondolf, G. M., and Piegay, H., eds., Tools in fluvial geomorphology: Wiley, New York, p. 539–576.

Perucca, E., Camporeale, C., and Ridolfi, L., 2005, Nonlinear analysis of the geometry of meandering rivers: Geophys. Res. Lett., v. 32, no. L03402, doi:10.1029/2004GL021966.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P., 1994, Numerical recipes in FORTRAN: the art of scientific computing: Cambridge University Press, New York, 963 p.

Rathbun, S. L., 1998, Spatial modelling in irregularly shaped regions: kriging estuaries: Environmetrics, v. 9, no. 2, p. 109–129.

Sampson, P. D., and Guttorp, P., 1992, Nonparametric-estimation of nonstationary spatial covariance structure: J. Am. Stat. Assoc., v. 87, no. 417, p. 108–119.

Skoein, J., Merz, R., and Bloschl, G., 2005, Top-kriging—geostatistics on stream networks: Hydrol. Earth Syst. Sci., v. 2, no. 6, p. 2253–2286.

Smith, J. D., and McLean, S. R., 1984, A model for flow in meandering streams: Water Resour. Res., v. 20, no. 9, p. 1301–1315.

Sun, T., Meakin, P., and Jossang, T., 2001a, A computer model for meandering rivers with multiple bed load sediment sizes 1. theory: Water Resour. Res., v. 37, no. 8, p. 2227–2241.

Sun, T., Meakin, P., and Jossang, T., 2001b, A computer model for meandering rivers with multiple bed load sediment sizes 2. computer simulations: Water Resour. Res., v. 37, no. 8, p. 2243–2258.

Viseur, S., 2004, Turbidite reservoir characterization: object-based stochastic simulation meandering channels: Bulletin de la Société géologique de France, v. 175, no. 1, p. 11–20.

Whiting, P.J., and Dietrich, W.E., 1991, Convective accelerations and boundary shear-stress over a channel bar: Water Resour. Res., v. 27, no. 5, p. 783–796.

Williams, G.P., 1986, River meanders and channel size: J. Hydrol., v. 88, no. 1–2, p. 147–164.