Interpolating Sparse Scattered Data Using Flow Information

Gregory J. Streletz*[1], Geoffrey Gebbie†[2], Oliver Kreylos‡[1], Bernd Hamann§[1], Louise H. Kellogg¶[3], and Howard J. Spero‖[3]

[1]Institute for Data Analysis and Visualization (IDAV), Department of Computer Science, University of California, Davis, CA, United States
[2]Department of Physical Oceanography, Woods Hole Oceanographic Institution, Woods Hole, MA, United States
[3]Department of Earth and Planetary Sciences, University of California, Davis, CA United States

Abstract

Scattered data interpolation and approximation techniques allow for the reconstruction of a scalar field based upon a finite number of scattered samples of the field. In general, the fidelity of the reconstruction with respect to the original scalar field tends to deteriorate as the number of samples decreases. For the situation of very sparse sampling, the results may not be acceptable at all. However, if it is known that the scalar field of interest is correlated with a known flow field – as is the case when the scalar field represents the value of an oceanographic tracer that propagates under the influence of the ocean's flow – then this knowledge can be exploited to enhance the scattered data reconstruction method. One way to exploit flow field information is to use it to construct a modified notion of distance between points. Replacing the standard Euclidean distance metric with a flow-field-aware notion of distance provides a method for extending standard scattered data interpolation methods into flow-based methods that produce superior results for very sparse data. The resulting reconstructions typically have lower root-mean-square errors than reconstructions that do not use the flow information, and qualitatively they often appear physically more realistic.

1 Introduction

Many physical datasets involve measurements of a scalar field that are collected at scattered locations in space. In order to analyze the underlying phenomena, it is generally desirable to know the corresponding scalar field, at least approximately, across its entire natural domain. Furthermore, in order to visualize the scalar field using computer graphics, it is generally necessary to compute values on a regular grid. For these purposes, scattered data interpolation (or approximation) is typically used in order to reconstruct the underlying scalar field on the domain of interest, based upon only the known set of scattered samples.

As an example, consider the problem of reconstructing scalar fields representing oceanographic quantities such as $^{18}O/^{16}O$ and $^{13}C/^{12}C$ isotope ra-

*gjstreletz@ucdavis.edu
†ggebbie@whoi.edu
‡kreylos@cs.ucdavis.edu
§hamann@cs.ucdavis.edu
¶kellogg@ucdavis.edu
‖hjspero@ucdavis.edu

tios for the oceans of the distant past. Stable isotope data are obtained from measurements on benthic foraminifera obtained from deep sea core samples taken from the ocean floor [1] . While these samples provide valuable data about the past ocean, they form a very sparse scattered dataset for which accurate interpolation can be a challenge, as is illustrated in Figure 1. In particular, note the existence of a large region of the South Atlantic for which no data are available at all.

Many different methods have been proposed for the reconstruction of scalar fields based on scattered observations [2, 3, 4, 5, 6, 7, 9]. Regardless of method, the quality of the results obtained depends upon the density of scattered samples available. If this density is sufficiently large, these methods typically work quite well. On the other hand, if the available sample set is sparse with respect to the spatial variation of the scalar field, such as in Figure 1, the fidelity of reconstruction may not be acceptable both in quantitative terms (as measured by root-mean-square agreement with the original field, for example) and in qualitative terms (as assessed by whether the reconstruction preserves significant features of the original field).

Although in general the sparsity of samples represents a fundamental limitation on the reconstruction quality that is possible, in some circumstances additional information can be used in proper context in order to obtain an improved result. One example of such a situation is the case of sparse scattered data interpolation of a scalar field that is associated with a flow field. The typical physical situation is that of a tracer quantity in a fluid; the spatial correlation of the scalar field representing the tracer concentration is related to the vector field specifying the fluid flow. Given this scenario, as an alternative to simply waiting for more data to become available we instead attempt to formulate an enhanced scattered data reconstruction scheme that can exploit these correlations in order to obtain a better result for sparse sample sets than would otherwise be attainable.

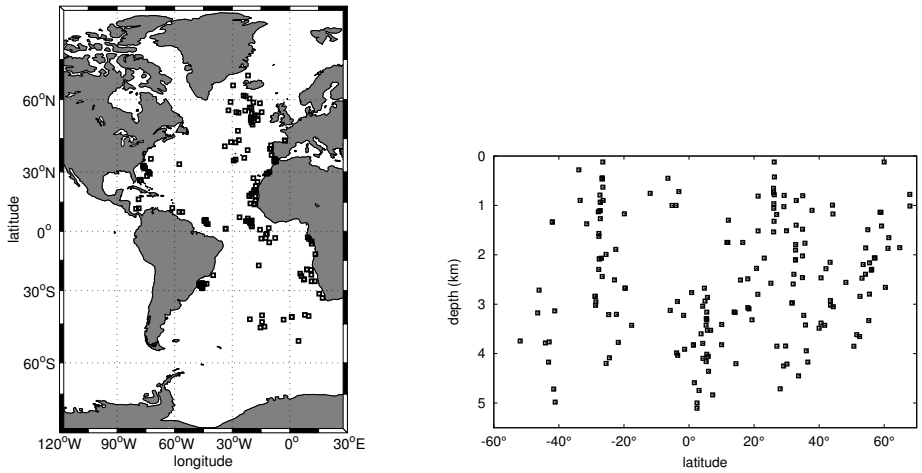
This paper describes a new scattered data reconstruction technique that utilizes a non-Euclidean distance measure to exploit known or assumed correlations between the scalar field being reconstructed and another known vector or scalar field. In a typical setting, the known field is a vector field representing flow and the scalar field being reconstructed represents a tracer quantity being transported under the influence of this flow. The physical characteristics of such a case lead to the intuitive assumption that concentrations of the tracer quantity will be more highly correlated in the flow direction than perpendicular to this direction. Hence, the proposed reconstruction method introduces an alternative, non-Euclidean way to measure distance between points in the spatial domain that is defined in reference to the streamlines of the flow field. Because this prototypical case serves as the motivating problem for the development of the method and because the majority of test cases described in the paper are of this type, we call the technique "flow-based."

We emphasize that although our primary application of the method lies in the oceanographic domain, for which the known vector field represents ocean flow, the method itself is generally applicable to a wide variety of problems involving multi-field data when correlations exist between fields. Many applications that involve fluid flow fall into this category. In these cases, correlations between points in a scalar field typically are stronger along the streamlines of flow than in other directions, and the particular non-Euclidean distance measure used reflects this basic property. For example, the method could be applied to atmospheric problems or to problems involving mantle convection.

Other situations are possible as well. For example, if a scalar field is related to a vector field in such a way that correlations are strongest across its streamlines instead of along them, then a different non-Euclidean distance measure would be used. The mathematical form of the distance function would be similar. Only the values of the parameters that define the specific incarnation of the distance function would be different.

Moreover, the existence of a physical flow is not required for the method to be applicable, but rather just the existence of a correlation between a scalar field and a vector field. In fact, it is not even necessary to have a vector field at all. The proposed technique is useful also for cases in which a scalar field of interest is correlated with another scalar field. In such cases, the isocontours of the known scalar field take the place of the streamlines of the vector field.

The layout of the paper is as follows. After describing some related work from the literature, we proceed to a discussion of background information regarding scattered data interpolation and distance metrics. We then describe our method of combining non-Euclidean distance measures with existing scattered data interpolation techniques. Next, we present results that illustrate the usefulness of the method for performing scattered data interpolation (or approximation) for data sampled from scalar fields that are correlated with flow fields. Finally, we describe ideas for possible future enhancement of the method.



(a) (b)

Figure 1: Core locations in the Atlantic Ocean, from Marchal and Curry, 2008 [1]. Figure 1a shows the scatter of core locations with respect to latitude and longitude, while Figure 1b shows the scatter with respect to latitude and ocean depth.

2 Related Work

The problem of scattered data interpolation has a long history, and a large variety of methods have been proposed for its solution [2]. One of the earliest approaches, proposed by Shepard [3], used inverse distance weighting to weight the scattered samples. Another approach is to use radial basis functions, such as in the method proposed by Hardy [4].

Both inverse-distance-weighted methods and radial-basis-function methods are naturally formulated as global methods, meaning that all available data points are used in computing the reconstruction for any given point in the spatial domain. For large data sets, global methods can be computationally prohibitive in practice. For this reason, local versions of these methods have been developed [2]. These restrict the number of data points that are used to compute each point of the reconstructed field. In addition, hierarchical methods for scattered data interpolation have been proposed [5]. These methods seek to combine the benefits of local and global approaches.

Other methods are fundamentally local in nature. For example, Sibson's method of natural-neighbor interpolation [6] employs the concept of Voronoi tessellation to define a weighting in terms of a given point's natural neighbors in the sample set. Moreover, this weighting is robust in cases for which many samples cluster close together, which can lead to reconstruction artifacts in other methods (such as inverse distance weighting). Because the construction of Voronoi tessellations is relatively expensive, the discrete Sibson method avoids the explicit computation of the Voronoi diagram [7].

Furthermore, statistical techniques such as Optimal Interpolation (OI) can be regarded as scattered data interpolation methods [8, 9]. Like Sibson's method, Optimal Interpolation is robust to clustering of samples. Moreover, it can be used for scattered data approximation as well as for scattered data interpolation, and it provides a built-in way to specify error bars on the scattered samples, which can be quite useful for problems in which the scattered samples are obtained via physical measurements. Early attempts to deal with scattered oceanographic data used statistical methods such as Optimal Interpolation [10].

While Optimal Interpolation is a simple example of a statistical scattered data interpolation technique, more sophisticated statistical methods have been applied to meteorological and oceanographic data assimilation problems. For example, the 3D-Var [11] and 4D-Var [12] methods have entered into common usage. In general, such methods can be viewed as inverse problems in that they attempt to reconstruct continuous scalar fields based on a discrete (and often sparse) set of measurements.

A more recent approach to solving such inverse problems for the specific case of ocean tracer distributions is a method called Total Matrix Intercomparison [13], which is related to 4D-Var. This method assumes multiple scalar fields in a flow field, but as-

sumes that the flow field is unknown and attempts to reconstruct it in addition to reconstructing the various scalar fields themselves. However, the reconstruction calculation involves a time-consuming optimization that makes the method more suited for off-line use than for interactive visualization purposes.

One of the principal contributions of the method described in this paper is the utilization of non-Euclidean distance measures for the purpose of scattered data interpolation and approximation. The use of non-Euclidean distances appears in the literature in related contexts. For example, Nielson and Foley have described the use of affine-invariant norms for the purpose of scattered data approximation [14]. Moreover, an example of the use of non-Euclidean distance measures for scattered data approximation in the meteorological sciences is given by the so-called "banana scheme" provided as an option in the PSU/NCAR Mesoscale Modeling System (MM5) [15].

In the context of distance measures, we note the related concept of geodesics on manifolds. Given a manifold embedded in a higher dimensional space (a 2D spherical surface embedded in 3D space, for example), the shortest path between two points on the manifold is a geodesic (for the spherical example, geodesics are great circles on the sphere). Efficient computational methods exist for the calculation of geodesics on 2D manifolds [16]. Furthermore, the concept of geodesics has been used in the context of interpolation. In such geodesic interpolation problems, pre-existing knowledge of geodesic paths is used for the purpose of surface reconstruction [17].

Because the Earth's surface is approximately a spherical manifold, geodesics are relevant to the oceanographic reconstruction problem when a grid-based approach is used [18]. However, we note that geodesics are not directly applicable to the problem described in this paper. Here, we are interested not in distances in embedded lower-dimensional spaces, but rather in non-Euclidean distance measures that are defined via reference to a correlated auxiliary field (e.g., the flow field) of the same dimension as the original reconstruction problem.

3 Background

3.1 Scattered Data Interpolation

Many of the existing methods for scattered data interpolation and approximation are formulated explicitly in terms of a distance function $d(\mathbf{x}_i, \mathbf{x}_j)$ whose value represents the distance between points $\mathbf{x}_i$ and $\mathbf{x}_j$ in the domain of the scalar function being reconstructed. For example, inverse distance weighting methods [3] reconstruct the scalar field at the point $\mathbf{x}$ by interpolating the N samples $f_i = f(\mathbf{x}_i)$ with the function

$$f(\mathbf{x}) = \sum_{i=0}^{N} \frac{w_i(\mathbf{x}) f_i}{\sum_{j=0}^{N} w_j(\mathbf{x})}$$

where $w_i(\mathbf{x}) = \dfrac{1}{d(\mathbf{x}, \mathbf{x}_i)}$.

While some reconstruction methods perform interpolation (so that the values at the sample points are reproduced exactly in the reconstruction), others instead perform approximation (where some reconstruction error is allowed at the sample points). Typically, such methods involve least-squares fitting of some kind [19]. One reason for preferring approximation over interpolation is that simple reconstructions can be fit relatively well to given datasets for cases in which interpolation might lead to complex, unrealistic reconstructions that oscillate wildly in order to fit all of the data points exactly. Furthermore, because simpler explanations typically are preferred to more complex ones (Occam's razor), using approximation rather than interpolation leads to reconstructions that tend to be regarded as better representations of physical reality.

Some methods can be used to perform either interpolation or approximation. For example, Optimal Interpolation [9] allows for the specification of errors for data points. As the specified errors approach zero, the approximation method approaches the case of true interpolation. Using approximation instead of interpolation provides a way to avoid fitting the noise in observations.

3.2 Distance Metrics

The distance function $d(\mathbf{x}, \mathbf{y})$ used for the scattered data interpolation methods above is typically chosen to be the familiar Euclidean distance, which for domains of dimension N is given by

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^{N} (x_i - y_i)^2}$$

However, other choices are possible. For a function $d(\mathbf{x}, \mathbf{y})$ to be a legitimate distance metric, it need only satisfy the following conditions [20]:

- $d(\mathbf{x}, \mathbf{y}) \geq 0$
- $d(\mathbf{x}, \mathbf{x}) = 0$
- $d(\mathbf{x}, \mathbf{z}) \leq d(\mathbf{x}, \mathbf{y}) + d(\mathbf{y}, \mathbf{z})$ (triangle inequality)

- $d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x})$ (symmetry).

Moreover, it is possible to relax one or more of the conditions above and to still have a function that resembles a distance metric in some ways. For example, relaxing the triangle inequality leads to a *semimetric*, while relaxing symmetry leads to a *quasimetric* [20]. Such generalized metrics will be used to extend standard scattered data interpolation schemes to incorporate knowledge of flow information.

4 Method

4.1 Flow-Based Distance Metrics

In order to adapt a scattered data interpolation and approximation scheme (in this case, Optimal Interpolation) to utilize flow information, the scheme's distance metric has been generalized so that distances in directions across streamlines of the flow field are given more weight than those in directions along the streamlines. This corresponds to the intuition that values of the scalar field should be more highly correlated in the direction of the flow than perpendicular to the flow. This intuition depends upon the assumption that the scalar field represents a tracer quantity that experiences advection in the presence of the flow field.

To understand the motivation for the particular flow-based non-Euclidean distance measure that we have adopted, consider first how normal Euclidean distances are calculated in two dimensions. Two points $\mathbf{x}_i$ and $\mathbf{x}_j$ in the plane are assigned x and y coordinates in a Cartesian coordinate system. Then, the Euclidean distance $d_E(\mathbf{x}_i, \mathbf{x}_j)$ between the two points is given by

$$d_E(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

where $(x_i - x_j)$ and $(y_i - y_j)$ are the measures of the distances between $\mathbf{x}_i$ and $\mathbf{x}_j$ in the x and y directions, respectively.

Given this basic expression for distance, if we wanted to give a different degree of emphasis to the y component of distance than to the x component, we could pre-process the data by multiplying the y coordinates of the points by a constant factor. Alternatively, we could use the x and y components of the points as is, and instead alter the definition of distance to incorporate the non-uniform treatment of coordinates. So, to reflect the scaling of the y component by a factor of γ, we would define the distance as

$$d_{E_2}(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{(x_i - x_j)^2 + \alpha(y_i - y_j)^2}$$

where $\alpha = \gamma^2$.

In analogy with the scaled Cartesian distance just discussed, we define a generalized distance function with respect to the streamlines of a given flow field. Instead of defining two-dimensional distances in terms of their x and y components, as in the Cartesian case, we define distances in terms of a component measured across the flow direction and a component measured along the flow direction. In particular, for points $\mathbf{x}_i$ and $\mathbf{x}_j$, let $d_1(\mathbf{x}_i, \mathbf{x}_j)$ represent a measure of the distance across the streamlines of the flow field, and let $d_2(\mathbf{x}_i, \mathbf{x}_j)$ represent a measure of the distance along the streamlines, as defined by the construction in Figure 2a.

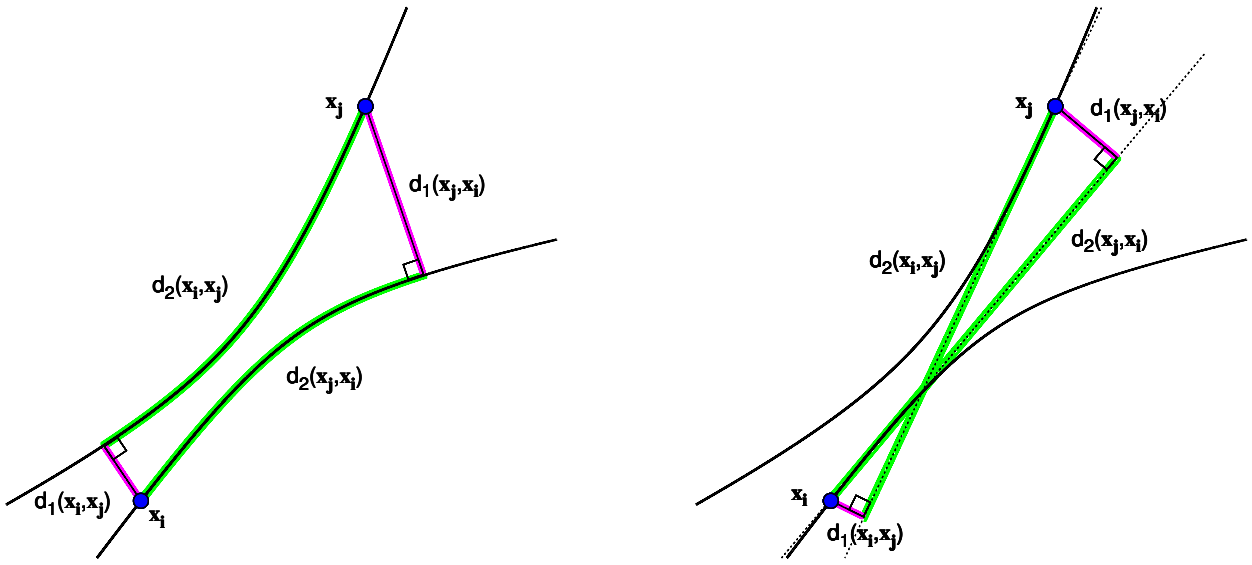
Then, the flow-based distance between the points $\mathbf{x}_i$ and $\mathbf{x}_j$ is given by

$$d(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{[d_1(\mathbf{x}_i, \mathbf{x}_j)]^2 + \alpha[d_2(\mathbf{x}_i, \mathbf{x}_j)]^2} \qquad (1)$$

where α is a factor determining the relative weight of distances along streamlines to those across streamlines. If α is chosen to be less than unity, then this distance measure will give more weight to distances across streamlines than to distances along the streamlines. This is in accordance with the intuitive expectation that tracer values should be more highly correlated in the direction of the flow than in directions perpendicular to it.

Note that the distance measure that we have defined is not only non-Euclidean, but anisotropic as well. In other words, according to our definition of distance, the distance between two points depends upon their relative orientation in space. This anisotropy is inherited from the anisotropy inherent in the notion of a flow field (flow, by definition, has a direction). Because we have assumed that the scalar field correlations between points are larger in the direction of flow than across the flow direction, we have defined a distance measure that reflects this anisotropic property of the scalar fields we are attempting to reconstruct.

If streamlines are computed from every point in the domain and are calculated as far as possible in both directions (until they leave the domain), then an exact definition of distance along streamlines and distance across streamlines can be employed, as shown in Figure 2a. Note that domain boundaries are somewhat problematic in relation to streamlines that leave the domain and then return. If a streamline is broken by a domain boundary, the distance metric will be affected, and therefore the distance between two



(a) (b)

Figure 2: Two methods for calculating distances from streamlines and distances across streamlines. Figure 2a depicts the exact method for calculating these distances. The flow-based non-Euclidean distance between points $\mathbf{x}_i$ and $\mathbf{x}_j$ is $d_{sym}(\mathbf{x}_i, \mathbf{x}_j)$, which is computed according to Eq.(1) and (2). The across-streamline distances $d_1(\mathbf{x}_i, \mathbf{x}_j)$ and $d_1(\mathbf{x}_j, \mathbf{x}_i)$ are the lengths of the magenta line segments in the diagram, while the along-streamline distances $d_2(\mathbf{x}_i, \mathbf{x}_j)$ and $d_2(\mathbf{x}_j, \mathbf{x}_i)$ are the lengths of the green curves. Figure 2b depicts an approximate method for calculating the distance from and across streamlines. Here, the lines tangent to the streamlines at the sample points are used to avoid having to compute the streamlines over the entire domain. The across-streamline distances $d_1(\mathbf{x}_i, \mathbf{x}_j)$ and $d_1(\mathbf{x}_j, \mathbf{x}_i)$ are the lengths of the magenta line segments in the diagram, while the along-streamline distances $d_2(\mathbf{x}_i, \mathbf{x}_j)$ and $d_2(\mathbf{x}_j, \mathbf{x}_i)$ are the lengths of the green line segments, which lie on the lines tangent to the streamlines at points $\mathbf{x}_j$ and $\mathbf{x}_i$.

points in a finite domain might be different from the distance that would be calculated if the domain were placed into its global context in terms of the flow field. For this reason, the proposed flow-based reconstruction method is best employed for closed systems (such as the entire ocean) or for domains over which the flow is relatively simple (such that the effect of broken streamlines is minimized).

In Figure 2a, $d_1(\mathbf{x}_i, \mathbf{x}_j)$ is the distance of point $\mathbf{x}_i$ from the streamline passing through point $\mathbf{x}_j$. Likewise, $d_1(\mathbf{x}_j, \mathbf{x}_i)$ is the distance of point $\mathbf{x}_j$ from the streamline passing through point $\mathbf{x}_i$. Similarly, $d_2(\mathbf{x}_i, \mathbf{x}_j)$ is the along-streamline component of the flow-based distance from $\mathbf{x}_i$ to $\mathbf{x}_j$, which is measured along the streamline passing through point $\mathbf{x}_j$, from the point at which the segment corresponding to $d_1(\mathbf{x}_i, \mathbf{x}_j)$ intersects the streamline to the point $\mathbf{x}_j$ itself. Likewise, the along-streamline component of the distance from $\mathbf{x}_j$ to $\mathbf{x}_i$ is $d_2(\mathbf{x}_j, \mathbf{x}_i)$, where the distance is measured along the streamline passing through point $\mathbf{x}_i$. The flow-based non-Euclidean distance from point $\mathbf{x}_i$ to point $\mathbf{x}_j$ is given by $d(\mathbf{x}_i, \mathbf{x}_j)$ according to Eq.(1), and the distance from point $\mathbf{x}_j$ to point $\mathbf{x}_i$ is given by $d(\mathbf{x}_j, \mathbf{x}_i)$, using the same equation. Note that these two distances are different, in general, indicating that the distance measure of Eq.(1) is not symmetric.

In order to obtain a symmetric distance metric, we can simply take the average of $d(\mathbf{x}_i, \mathbf{x}_j)$ and $d(\mathbf{x}_j, \mathbf{x}_i)$. So, a symmetric distance function $d_{sym}(\mathbf{x}_i, \mathbf{x}_j)$ is defined by

$$d_{sym}(\mathbf{x}_i, \mathbf{x}_j) = \frac{d(\mathbf{x}_i, \mathbf{x}_j) + d(\mathbf{x}_j, \mathbf{x}_i)}{2} \tag{2}$$

where $d(\mathbf{x}_i, \mathbf{x}_j)$ and $d(\mathbf{x}_j, \mathbf{x}_i)$ are computed according to Eq.(1) and where $d_{sym}(\mathbf{x}_i, \mathbf{x}_j)$ satisfies $d_{sym}(\mathbf{x}_i, \mathbf{x}_j) = d_{sym}(\mathbf{x}_j, \mathbf{x}_i)$.

While symmetry could be obtained in other ways, in general there is no reason to prefer one of the two paths between point $\mathbf{x}_i$ and point $\mathbf{x}_j$ over the other, and therefore the simple averaging procedure is appropriate. If the specific physical situation of a particular application provides a reason for preferring one path over the other, then this additional domain-specific knowledge can be utilized when deciding how to enforce symmetry upon the flow-based distance measure. For example, in a certain setting, perhaps the shortest path provides the best definition of the distance between the two points. Furthermore, if symmetry is not required for a particular application, then computational costs can be reduced considerably, as is described in Section 4.4.

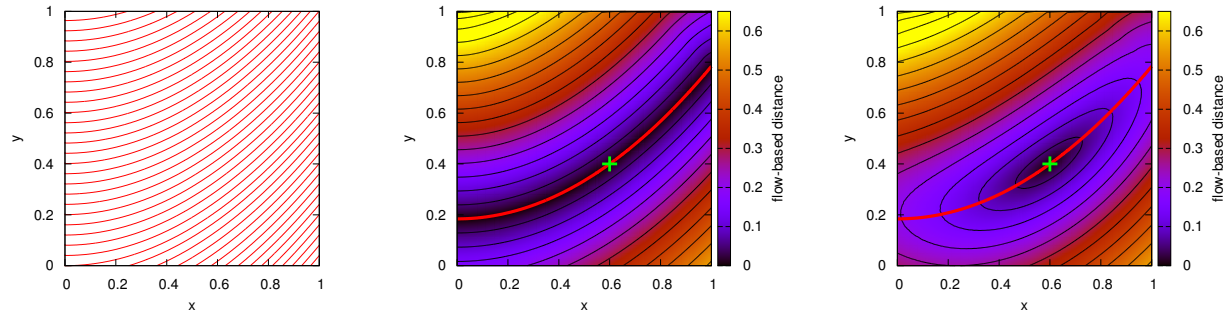
Figure 3 shows two examples of distance functions calculated according to the method illustrated in Figure 2a. In the first example, the α parameter is 0, so the overall measure of distance utilizes only the distance from the streamline and completely ignores the distance in the streamline direction. As a result, points close to the streamline that passes through the specified point all have very small distances when compared to that point. In the second example, α is 0.1, so the distance along the streamline has some influence on the overall distance measure. Nevertheless, the distance from the streamline still has greater influence.

If the calculation of streamlines across the entire domain is too expensive computationally, an alternative is to linearize the streamlines about the sample points $\mathbf{x_i}$ and $\mathbf{x_j}$, as shown in Figure 2b. In this case, the definitions of the distances $d_1(\mathbf{x}_i, \mathbf{x}_j)$, $d_1(\mathbf{x}_j, \mathbf{x}_i)$, $d_2(\mathbf{x}_i, \mathbf{x}_j)$, and $d_2(\mathbf{x}_j, \mathbf{x}_i)$ are analogous to those in the previous case, and the definitions of $d(\mathbf{x}_i, \mathbf{x}_j)$ and $d_{sym}(\mathbf{x}_i, \mathbf{x}_j)$ are identical, as specified by Eq.(1) and (2), respectively. This linearization approach reduces the computational complexity of the method. However, there is a cost in terms of accuracy. In particular, if the points $\mathbf{x}_i$ and $\mathbf{x}_j$ are far apart (in terms of Euclidean distance), then the validity of the resulting distance metric can be called into question. For this reason, the linearized streamline approach to defining a distance metric is best used in combination with a technique for decaying the degree of anisotropy as the (Euclidean) distance between points increases.

Figure 4 shows an example of a distance function calculated using the decaying anisotropy approach. Local to the point of interest, the distance measure weights the distance from the streamline much more heavily than the distance along the streamline. However, as the (Euclidean) distance from the point of interest increases, this anisotropy decays.

Based upon the considerations above, it is obvious that there are many possible ways to generalize the Euclidean distance to a non-Euclidean measure of distance that utilizes flow information. Within the framework above, even if we select one of the two general approaches for the incorporation of flow information (the exact streamline method or the approximate, linearized streamline method), we still have a large space of alternative non-Euclidean distance metrics to consider. This family of distance measures is parametrized by several parameters.

At the very least, we will have the parameter α that specifies the relative weighting of distances across and along streamlines. If we are using a decaying anisotropy, we will also have one parameter that specifies the inner radius of the decay region and another the specifies the outer radius of the de-



(a) (b) (c)

Figure 3: Two different flow-based distance measures that use the same flow field. The flow field with respect to which both distance functions are defined is depicted in Figure 3a. As is the case throughout the paper, x and y represent the standard Cartesian coordinates of points in 2D space. The plot in Figure 3b shows a distance measure that considers only the distance from streamlines (using an α parameter of 0). The color scale represents the (non-Euclidean) distance of points in the plane from the indicated point that lies on the streamline shown. The plot in Figure 3c shows the distance function that results for an α value of 0.1, so that the overall distance measure is influenced by the distance along streamlines but the distance from streamlines is given more weight. Red curves represent streamlines and black curves represent contours of the distance function.

cay region. Moreover, we could introduce additional parameters in order to expand the size of the family of distance measures that we can model. For example, we could parametrize the exact form of the decay of anisotropy between the inner and outer radii. We could also choose to consider not only flow direction, but flow magnitude as well; this will necessitate the introduction of one or more additional parameters.

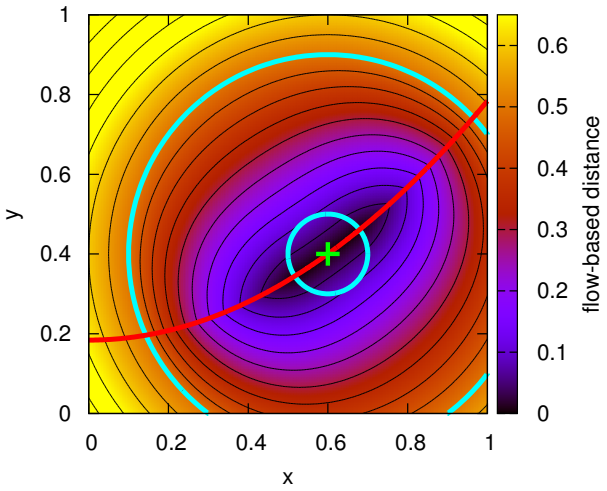


Figure 4: A distance measure that uses a decaying anisotropy. For points near the point represented by the green marker, the flow-based non-Euclidean distance measure weights distance from the streamline much more heavily than distance in the streamline direction (using $\alpha = 0.0001$, in this case). However, as the (Euclidean) distance from the point increases, this anisotropy in the distance measure is decreased, so that for points far away the distance is approximately Euclidean. The decay of anisotropy occurs linearly between the inner and outer decay radii indicated by the two circles.

Given such a family of distance measures, the obvious question is what specific one is best for the interpolation problem at hand. The precise answer to this is problem specific, so the approach that we have implemented is to optimize the flow-based interpolator distance measure for each problem independently. Hence, the parameter adjustment process is treated as an integral part of the interpolation method itself. The specific parameter adjustment method employed will be discussed in detail in Section 4.3.

4.2 Streamline Calculations

In order to compute $d_1(\mathbf{x}_i, \mathbf{x}_j)$ and $d_2(\mathbf{x}_i, \mathbf{x}_j)$ by reference to the streamlines of the flow field (or other correlated vector field), as described above, we must first calculate the streamlines themselves. Given the flow field, this is a fairly straightforward procedure that involves integrating the flow field from each point of interest. We perform this integration numerically using the fourth-order Runge-Kutta method [19]. The flow field itself is assumed to be an input to the reconstruction problem that reflects some *a priori* knowledge about the problem at hand.

4.2.1 Using Isocontours as Streamlines

It should also be noted that it is possible to compute non-Euclidean distance measures with respect to the isocontours of any given scalar field rather than with respect to the streamlines of a flow field. The "flow-based" reconstruction method can then be applied as if these isocontours were the streamlines of a known flow field. This mode of operation is useful when it is known that the scalar field of interest is correlated with another scalar field which is known on the domain of interest, even if actual flow information is not available.

4.3 Parameter Optimization

The generalized distance metric defined above in terms of the streamlines of the flow field depends also on the free parameter α, which determines the relative weight to give to distances along the streamlines versus distances across the streamlines. As mentioned in Section 4.1, the distance function may also involve additional parameters besides α (such as those defining the inner and outer decay radii when a decaying anisotropy is utilized). If flow magnitude is considered, or if the distance function is otherwise modified for the purposes of a specific application, further parameters may be introduced. The values of all these parameters determine the specific nature of the non-Euclidean distance measure to be used, and this in turn influences the reconstructions that are obtained when the flow-based interpolation/approximation method is applied to a given set of scattered data.

Moreover, the underlying interpolation and approximation method that we have chosen, Optimal Interpolation, also involves a free parameter that must be chosen by the user (the so-called correlation length, which determines the rate at which the influence of a given sample point decays as the point of interest is moved away from it). This method also involves an input parameter that specifies the expected magnitude of the errors between the input data and the values of the reconstruction at the corresponding locations. Although we usually have chosen to set this error parameter to some reasonable fixed value, it also can be considered to be a free parameter of the method. The effectiveness of the proposed reconstruction method depends upon the specific choices made for these parameters, and the optimal values of the parameters can vary from problem to problem.

While exhaustive parameter studies can be performed on specific cases in order to gain insight into the dependence of interpolation quality on the parameter settings, it is desirable to have a more efficient way to select the precise parameter values that lead to the best results. If the exact scalar field is known, we can use standard optimization techniques to find the best parameter values for that specific scalar field in the presence of the specific known flow field. Although gradient descent methods might be applicable here, an easier approach (that avoids having to calculate gradients) is to employ the direct search method (also known as compass search or pattern search) [21, 22]. Furthermore, because it does not require gradient information, direct search is robust even when the objective function lacks smoothness or continuity.

Unfortunately, like any local optimization method, direct search is susceptible to falling into a local minimum of the objective function that is not the global minimum sought. In some cases we simply assume that either the initial guesses used are close enough to the global minimum that the local direct search will find the parameter configuration corresponding to this optimum value, or that the objective function in fact has only one local minimum. However, we have observed that the objective functions that result from most practical reconstruction problems do involve multiple local minima, so we have implemented a simple procedure that attempts to deal with this complication.

The procedure for global optimization is simply to perform an initial scan of the parameter space (at some user-defined resolution) in order to attempt to select an initial guess that is likely to be close to the global minimum. For applications for which we are concerned that falling into a local minimum might lead to substantially inferior results, we utilize this procedure prior to invoking the direct search algorithm. The initial guess used for the direct search algorithm is then selected to be the parameter combination corresponding to the lowest RMS error encountered during the initial scan. While more sophis-

ticated approaches to global optimization are possible, we have found that this simple approach is adequate for most cases.

If the scalar field being reconstructed is known beforehand, the objective function to be optimized is simply the RMS error between the known values at all locations considered (for example, on a grid covering the entire spatial domain) and the values found for these locations using the flow-based reconstruction method with a given parameter configuration. Of course, in most real scattered data interpolation problems, we will not have knowledge of the entire scalar field beforehand. However, by downsampling the set of available scattered samples and comparing the interpolation results using these subsamples to the known values of the scalar field at the omitted sample locations, we can construct a suitable objective function for optimization.

In practice, a reasonable approach is to remove one point at a time and to compute the reconstruction there using the other points. By doing this for each point in the sample set and taking the root-mean-square error across all points, we essentially are utilizing a "leave-one-out" cross validation scheme. The cross validation is done during each iteration of the iterative optimization algorithm. The expression for the objective function defined using leave-one-out cross-validation is given by

$$RMSE_{CV} = \sqrt{\frac{\sum_{i=1}^{n} e_i^2}{n}} \tag{3}$$

where e_i is the error at the ith of the n sample points when computing a reconstruction using the other $n-1$ points.

Hence, direct search can then be used to approximate the optimal parameter settings for the given problem, even though we only know the value of the scalar field at a limited number of points. Note that this method of parameter optimization using downsampling is closely related to the well-known "jackknife" method in statistics [23]. While the approach of withholding data for subsequent validation is quite common, our contribution here is to advocate the use of parameter optimization using leave-one-out cross validation as an integral part of the reconstruction process.

4.4 Computational Complexity

The computational complexity of the method described above depends on the specific choices made for its various components. For example, whether or not streamlines are calculated across the entire domain has a big effect on the overall complexity of the interpolation method. If we intend to compute a reconstruction on a dense discretization of the entire spatial domain and if we decide to use the exact method for defining a flow-based non-Euclidean distance measure depicted in Figure 2a, then we will have to compute streamlines through every point of the discretization, which will lead to a large computational overhead just for the computation of the distance function.

To reduce this overhead, we could decide to employ the approximate method for defining a flow-based distance measure (Figure 2b), which does not require the computation of streamlines. Alternatively, if we are willing to accept a distance function that is non-symmetric with respect to the sample locations and also possibly discontinuous at these locations, then we can substantially reduce the computational cost in another way. Specifically, because of the relaxation of the symmetry requirement, we can compute the streamlines that pass through the sample locations, and then define the flow-based distance measure using only these streamlines. Because we typically have a sparse scattered dataset, we will need to compute far fewer streamlines than if we were to impose symmetry, which necessitates the computation of streamlines through every reconstruction point in addition to through every sample point.

In addition, if parameter optimization is performed to tune the free parameters of the method (as described in Section 4.3), then the computational costs are increased. The direct search procedure described in Section 4.3 is an iterative method whose convergence properties contribute to the overall computational complexity of the interpolation scheme. Furthermore, if this local optimization method is used along with a global optimization scheme (in order to deal with the presence of local minima), then the computational costs are increased even further.

Of course, the computational costs of the underlying interpolation/approximation method must be considered as well. Optimal Interpolation, like many interpolation and approximation algorithms, ultimately involves the numerical solution of a matrix equation. The matrices involved scale quadratically with the number of input data points (the scattered samples), and the cost of solving a matrix equation of this size must then be multiplied by the number of points at which the reconstruction is to be computed.

5 Results

5.1 An Analytic Test Case

A simple test case to assess the potential of the flow-based scattered data interpolation method is presented next. Because the flow-based distance functions used by the proposed method make the fundamental assumption that scalar fields are highly correlated along streamlines, as opposed to across streamlines, the method should perform well when this assumption holds exactly. To verify this, the interpolation method can be applied to a scalar field and flow field for which the isocontours of the scalar field are exactly coincident with the streamlines of the flow field.

Figure 5 illustrates such a case. In Figure 5a, an oscillating scalar field is shown. Superimposed on the field are representative streamlines of the associated flow field. As can be seen, these streamlines are also isocontours of the scalar field.

Figure 5 shows the result of using the flow-based scattered data interpolation method to reconstruct the scalar field based on only two samples (one toward the upper left of the spatial domain and the other toward the lower right). A visual comparison of Figures 5a and 5b shows that the method produces a relatively accurate reconstruction of the original scalar field even though only two samples of the original field are used. The effectiveness of the method results from its use of knowledge of the associated flow field (via a flow-based non-Euclidean distance function), combined with the fact that the flow field directions exactly specify the isocontours of the scalar field.

5.2 Tests for Cases Involving Various Degrees of Diffusivity

5.2.1 Test Cases from an Advection-Diffusion Model

In order to test the effectiveness of the proposed flow-based interpolation method for interpolating actual physical datasets, we need test data that reflect a realistic relationship between a scalar field and an associated flow field. A convenient way to construct such realistic test cases is to employ a physical model. We have chosen to use steady-state solutions to a simple 2D advection-diffusion model that approximates the behavior a tracer quantities in the ocean [24].

Defining the tracer concentration at location (x, y) and time t to be $\theta(x, y, t)$ and the x and y components of the (time-invariant) flow velocity at location (x, y) to be $v_x(x, y)$ and $v_y(x, y)$, respectively, the time evolution of the tracer concentration scalar field is modeled by the partial differential equation

$$\frac{\partial\theta}{\partial t}+v_x\frac{\partial\theta}{\partial x}+v_y\frac{\partial\theta}{\partial y}=\kappa_x\frac{\partial^2\theta}{\partial x^2}+\kappa_y\frac{\partial^2\theta}{\partial y^2} \quad (4)$$

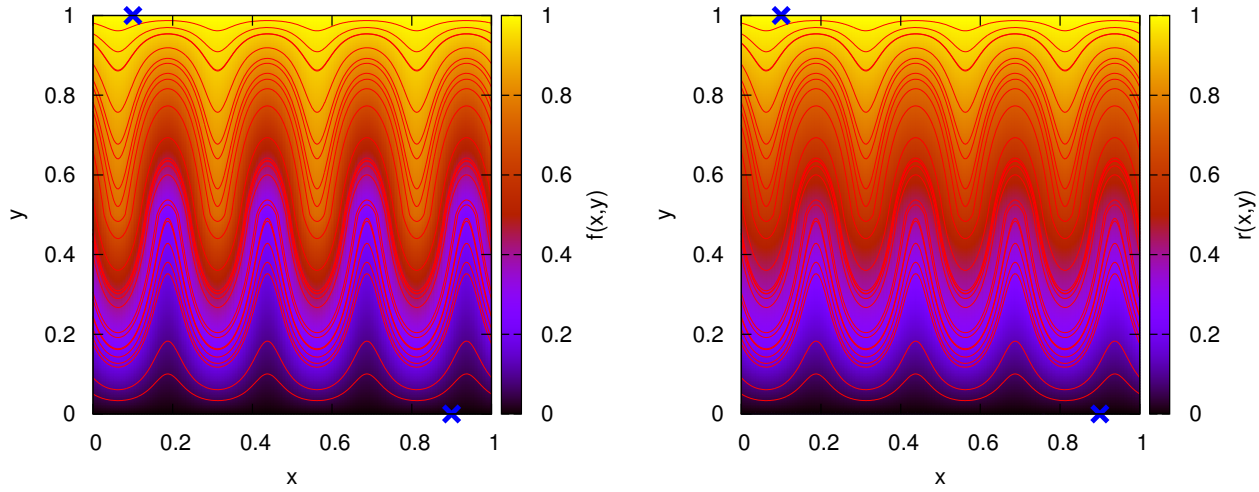
where the constants κ_x and κ_y are diffusivities in the x and y directions, respectively. For all cases presented in this paper, we have assumed isotropic diffusion, for which $\kappa_x = \kappa_y$. Physically, the diffusivities κ_x and κ_y have dimension $length^2/time$. However, because we are using this model solely to construct mathematical functions of x and y to use as test cases for our reconstruction method, we typically will avoid the explicit assignment of physical units to κ_x and κ_y (and to x, y, t, v_x, v_y, and θ, for that matter).

Solving the differential equation of Eq.(4) numerically, we have obtained a collection of test cases for various values of diffusivity. Figure 6 illustrates the scalar fields that correspond to three different values of the diffusivity coefficients κ_x and κ_y in Eq.(4) (with $\kappa_x = \kappa_y$ for each case). All three cases use the same flow field (a left-to-right flow field with constant flow magnitude over the entire (x, y) domain) and boundary conditions (a Dirichlet boundary condition specifying the concentration on the left boundary and no-flux boundary conditions enforced on the other three boundaries). In all cases, we have solved for the steady state solution, which we call simply $\theta(x, y)$. It is this 2D scalar field $\theta(x, y)$ that is depicted in each plot of Figure 6.

Clearly, the correlation between scalar field values at two points on a given streamline is greater when the ratio of flow field velocity to diffusivity (the Péclet number) is large. So, for a given flow velocity, the correlation is higher for a low-diffusivity case than for a high-diffusivity case. The ability to vary the diffusivity is one of the benefits of testing the interpolation method using the results of the advection-diffusion model, as this provides a convenient way to assess the effect of diffusion and to quantify the rate at which reconstruction quality degrades as the diffusivity becomes large (thereby reducing the usefulness of the known flow information).

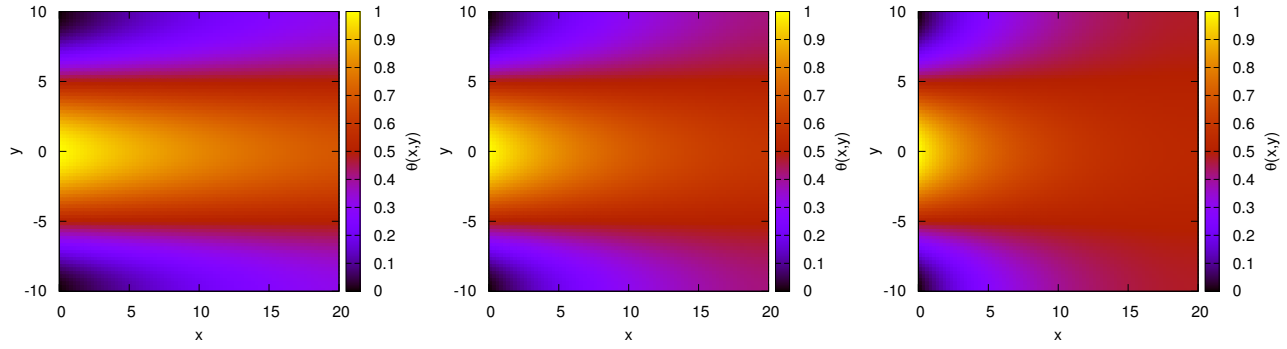
5.2.2 Parameter Study for an Advection-Diffusion Test Case

In this section, we show how the performance of the proposed flow-based scattered data interpolation method depends upon the choice of parameters used with the method. Figure 7 shows the problem that will be used to illustrate the dependence of interpola-



(a) (b)

Figure 5: A simple analytic test case that illustrates the potential of the flow-based interpolation method. Figure 5a shows an oscillating scalar field $f(x, y)$ whose isocontours are streamlines of the associated flow field. The flow-based scattered data interpolation method produces an adequate reconstruction $r(x, y)$ of the oscillating scalar field based on knowledge of the flow field and only two samples of the scalar field (denoted by the blue markers), as can be seen in Figure 5b.



(a) (b) (c)

Figure 6: Three different 2D scalar fields resulting from the solution of the advection-diffusion equation of Eq.(4) with a flow field moving from left to right with a constant flow magnitude of unity. Figure 6a shows the solution $\theta(x, y)$ for a diffusivity of 0.5, Figure 6b shows the solution for a diffusivity of 1.0, and Figure 6c shows the solution for a diffusivity of 2.0. The quoted values for diffusivity refer to the numerical values used for both κ_x and κ_y in Eq.(4). For each case, there is a Dirichlet boundary condition at the left domain boundary that specifies a time-invariant concentration there. The rest of the scalar field is the steady-state solution of the advection-diffusion equation.

tion results on the parameters chosen. The plot shows the scalar function of interest, which was constructed using the advection-diffusion model described above. Overlaid on the plot are the locations of 30 points at which samples of the scalar field are taken. The interpolation task is to use the flow-based scattered data interpolation method to compute a reconstruction of the original scalar field based on knowledge of these 30 sample points and the fact that the associated flow field is that of a left-to-right flow.

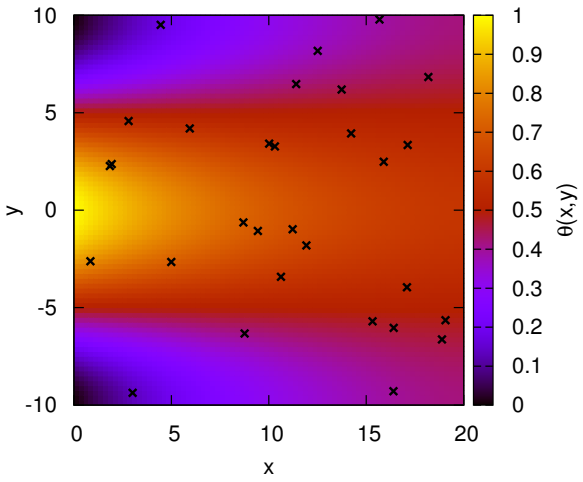


Figure 7: A 2D scalar field $\theta(x, y)$ in the presence of a left-to-right flow field, constructed using the advection-diffusion model of Eq.(4), with 30 scattered samples indicated.

Figure 8 shows the results of applying the interpolation method with different parameter configurations. The family of non-Euclidean flow-based distance functions that were used for this case was parametrized only by the parameter α, which specifies the relative weight of distances along streamlines versus distances across streamlines. In addition to varying α, the parameter study also considered various values of the correlation coefficient used with the underlying Optimal Interpolation method (a large correlation length corresponds to a greater sphere of influence for each sample point, while a small correlation length implies that each point's influence on the reconstruction is relatively local to the streamline passing through it).

Each curve in Figure 8a corresponds to a different value of the α parameter. The horizontal axis represents the value of the OI correlation length parameter and the vertical axis represents the root-mean-square (RMS) error for a point-by-point comparison of the original scalar function of Figure 7 with the reconstruction corresponding to the parameter configuration being used. Conversely, each curve in Figure 8b corresponds to a different value of the OI correlation length parameter, with the horizontal axis corresponding to the value of the α parameter. The red curve in Figure 8a represents the RMS error, as a function of the correlation length parameter, of reconstructions computed using the standard OI method with a Euclidean distance function. In both plots, the blue line indicates the RMS error of the best reconstruction computed using standard OI, which serves as a baseline to assess the effectiveness of the flow-based method.

The first thing to notice from Figure 8 is that there are parameter combinations (of the α and correlation length parameters) that lead to RMS errors that are lower than the lowest error achieved by a reconstruction using the standard OI method. These parameter combinations are represented by the portions of the parameter study curves that lie below the blue lines. The second important observation is that some of the curves exhibit local minima with respect to one or both of the parameters. This observation is pertinent when considering optimization-based automatic procedures for finding a near-optimal parameter configuration.

As can be seen from Figure 8, parameter combinations exist that result in better reconstructions (in terms of having a lower RMS error) than the standard OI method applied to the same problem. However, how to find these parameter combinations is unclear. The parameter study indicates what the near-optimal parameter combinations are, but the study was performed with *a priori* knowledge of the actual scalar field being reconstructed. In other words, the only way we were able to compute the RMS errors displayed in Figure 8 was by comparing the reconstructions obtained to the known ground truth shown in Figure 7. In practice, the ground truth will not be known, so identifying the near-optimal parameter combination is not straightforward.

Our solution to this problem is to utilize leave-one-out cross validation, as described in Section 4.3. Figure 9 shows the results of a parameter study con-

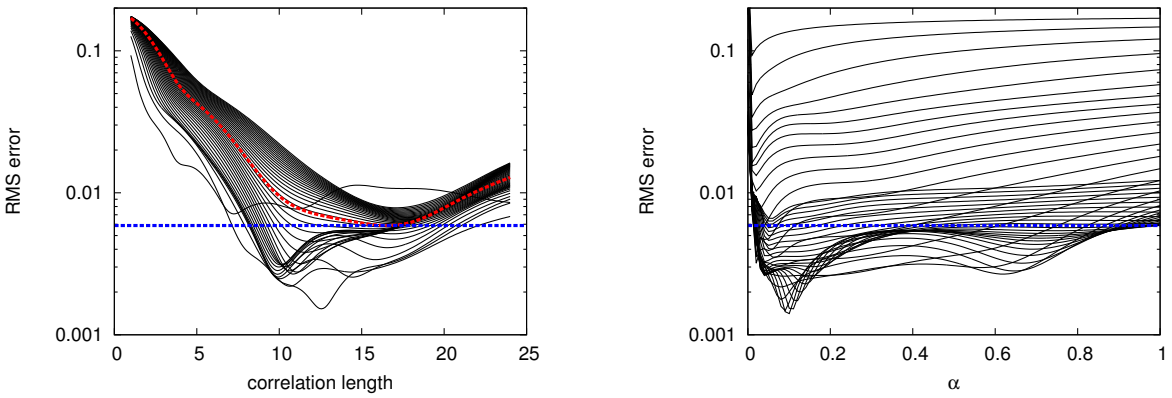


Figure 8: A parameter study showing the RMS error of reconstructions of the scalar field of Figure 7 compared to the known values for the scalar field. Figure 8a shows the RMS error as a function of the OI correlation length. Each curve corresponds to a different value of the α parameter. Figure 8b shows the RMS error as a function of α for curves representing different values of the correlation length. The parameter study results indicate that local minima of RMS error can exist with respect to both the correlation length and the α parameter. In Figure 8a, the red line represents the RMS error obtained by using the standard Optimal Interpolation algorithm for various values of the OI correlation length parameter. In both plots, the blue line represents the best possible reconstruction (for any correlation length) obtained using OI. As can be seen, there are parameter combinations for which the flow-based method yields a lower RMS error than the regular OI algorithm.

ducted using the leave-one-out cross validation definition of RMS error given in Eq.(3). The results shown parallel those shown in Figure 8. As for the previous case, we observe local minima with respect to both α and the correlation length parameter (with potential consequences for any optimization method applied to the RMS error function). Most importantly, though, we note that the near-optimal parameter combinations for this alternative RMS error function appear to be a reasonable proxy for those for the RMS error function that was computed with respect to the known ground truth scalar field. In other words, the values of α and correlation length that lead to the lowest RMS errors in Figure 9 correspond to relatively low RMS error values in Figure 8 as well. While the optimum configuration for Figure 9 does not correspond exactly to the optimum configuration for Figure 8, we see that it at least leads to an RMS error below the blue lines in Figure 8 (the OI baseline). Therefore, we have demonstrated the potential value of the leave-one-out cross validation approach for identifying near-optimal parameter combinations for a given problem, or at least ones that will lead to reconstructions superior to those computed using the standard OI method.

The comparison of the parameter study results shown in Figures 8 and 9 has shown that leave-one-out cross validation provides a way around the problem of how to calculate RMS error when the ground truth is not known. However, even so, performing an exhaustive parameter study such as the one shown in Figure 9 in order to find the best parameter configuration is too expensive to be practical for interactive visualization purposes. In order to find a good combination of parameters without having to perform a full parameter study, the parameter optimization approach described in Section 4.3 has been implemented. The results obtained using this approach are presented next.

5.2.3 Results Using Dynamically Optimized Method

The parameter study results above illustrate the typical behavior of the RMS error with respect to the choice of parameters used with the flow-based reconstruction method. In order to find the near-optimal choice of parameters without having to run a full parameter study, a dynamic parameter optimization can be performed during the interpolation process itself. Such a parameter adjustment can be made using any nonlinear optimization method, but as has already been described in Section 4.3, the method

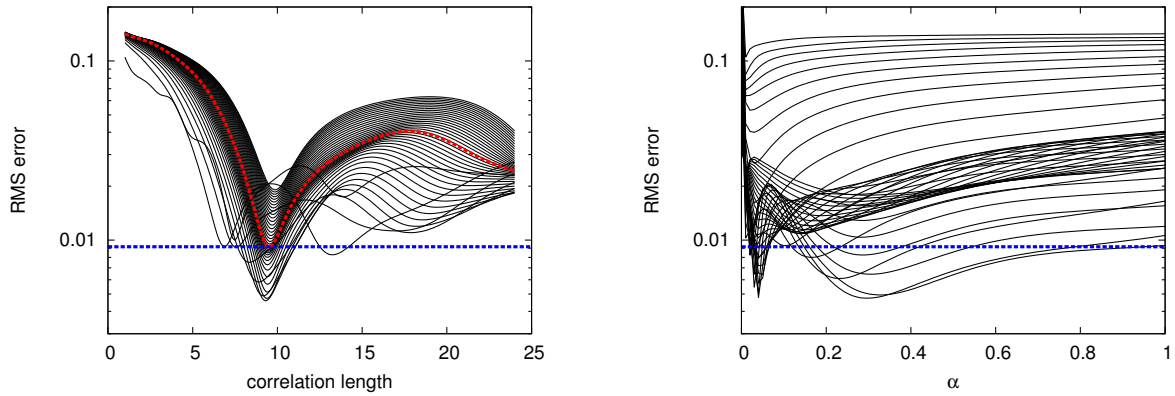


Figure 9: A parameter study showing the RMS error of reconstructions of the scalar field of Figure 7 with respect to the known values at the sample points only (in other words, using the information that is usually available in practice). The RMS errors shown correspond to the alternative definition of RMS error using leave-one-out cross validation, as defined in Eq.(3). Figure 9a shows the RMS error as a function of the OI correlation length. Each curve corresponds to a different value of the α parameter. Figure 9b shows the RMS error as a function of α for curves representing different values of the correlation length. As with the results presented in Figure 8, the parameter study results presented here indicate that local minima of RMS error can exist with respect to both the correlation length and the α parameter (in this case, for the RMS error computed using leave-one-out cross validation). In Figure 9a, the red line represents the cross-validation-based RMS error ($RMSE_{CV}$ from Eq.(3)) obtained when using the standard Optimal Interpolation algorithm instead of the flow-based method, plotted as a function of the correlation length parameter used with the standard algorithm. In both plots, the blue line represents the best possible cross-validation-based RMS error (for any correlation length) obtained using OI. Note that the parameter combinations leading to low cross-validation-based RMS error values in Figure 9 are a reasonable proxy for parameter combinations leading to low RMS errors for the reconstructions represented in Figure 8.

we have implemented uses a simple direct search to perform the optimization.

Figure 10 shows an alternative presentation of the data shown in Figures 8 and 9 for our flow-based reconstruction method. Here, the RMS error functions are shown explicitly as functions of both α and the OI correlation length parameter. These functions are the objective functions that are to be minimized using the dynamic parameter optimization method. Figure 10a shows the RMS error calculated with respect to the ground truth of Figure 7. Because the ground truth is known for this case, we can perform a 2D direct search directly on this objective function in order to find the best possible parameter combination for the problem (the point indicated by the green marker in Figure 10a).

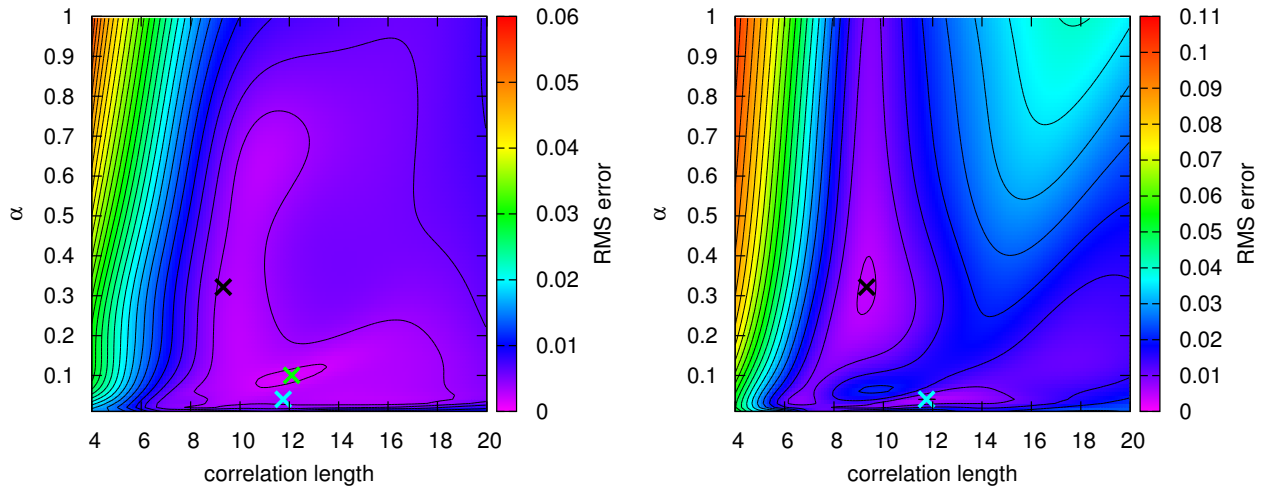
In practice, however, we usually do not know the ground truth scalar field *a priori*. So, instead of optimizing using the objective function of Figure 10a, we will typically optimize with respect to an objective function defined in terms of the leave-one-out cross-validation-based RMS error defined in Eq.(3). Figure 10b shows this alternative RMS error surface, which was calculated by applying leave-one-out cross validation using the 30 available sample points depicted in Figure 7. Notice that the surface has two local minima in the parameter range shown. Which of these minima is found during the direct search procedure depends upon the initial guess used at the start of this iterative algorithm. This initial guess, in turn, is determined by the results of the initial scan done beforehand, if one is used (as described in Section 4.3).

In Figure 10b, the cyan marker represents the location of the global minimum, so if a fine enough initial scan is performed, the subsequent direct search will find this minimum rather than the local minimum represented by the black marker (at which the cross-validation-based RMS error is only very slightly greater). Note from Figure 10a that this parameter configuration is nearby the one that leads to the minimum RMS error with respect to the ground truth. Hence, when using the parameters found during the dynamic parameter optimization phase of the algorithm (which depends only on knowledge of the scalar field values at the sample points), our method yields a reconstruction that is close to the optimal one for the problem. Also, note that even if the other local minimum of the function of Figure 10b were found during the parameter optimization (or if the surface were slightly different, so that the black marker represented the global minimum of that function), the reconstruction obtained would still be a good one. In particular, the contour lines in Figure 10a indicate that even though the black marker is further from the best possible configuration in the parameter space (the green marker), the RMS error obtained is similar to that obtained using the configuration represented by the cyan marker.

Although the example just discussed illustrates the parameter adjustment process using a family of distance functions parametrized by the single parameter α, the same procedure can be used for more complicated families of distance functions that are parametrized by more than one parameter. For example, when using distance functions with decaying anisotropy, the direct search optimization would search in two additional directions to find the near-optimal values of the inner and outer radii for the decay region to be used. Although computational complexity is increased as additional variables are included within the optimization loop, the computational costs are mitigated by the fact that for problems involving sparse scattered samples there are relatively few RMS error values to be computed during each iteration of the optimization. Furthermore, for the example given (involving distance functions with decaying anisotropy), the computational savings from avoiding streamline calculations by using the linearized streamline approximation of Figure 2b presumably would outweigh the added computation required to handle the optimization of the extra parameters introduced by the decaying anisotropy construct.

In order to demonstrate the utility of our proposed flow-based reconstruction method as compared to standard methods, we now present results from applying the method to a collection of test cases similar to the one just described (whose error surfaces are illustrated in Figure 10). Each case uses the same 30 sample locations depicted in Figure 7 and involves a scalar field that has been generated using the advection-diffusion model of Eq.(4) with the same boundary conditions and flow field as for the case shown in Figure 7 but with a different value for the diffusivities κ_x and κ_y (with $\kappa_x = \kappa_y$ for each case, as before). For each case, we have used the direct-search-based dynamic parameter optimization procedure, and have attempted to avoid local minima of the objective functions by first performing an initial scan of the parameter space to determine a reasonable initial guess for each problem.

Figure 11 shows RMS errors for a range of diffusivities. For comparison purposes, results are shown both for the standard Optimal Interpolation method and for our flow-based reconstruction method. For the former, the OI correlation length parameter was optimized for each case, and for the latter both the



(a) (b)

Figure 10: Illustration of RMS error as a function of the correlation length and the α parameter. Figure 10a shows the RMS error of reconstructions compared against the known ground truth. Figure 10b shows the leave-one-out cross-validation-based RMS error. The latter serves as the objective function for a dynamic direct search optimization of the parameter space. The green marker shows the location of the minimum of the RMS error surface (the best reconstruction over the entire parameter space). The cyan and black markers show the local minima of the cross-validation-based RMS error surface. The cyan marker happens to be the global optimum, so the cross validation procedure leads to choosing a parameter combination that is close to the global minimum of the reconstruction error surface in Figure 10a.

correlation length and the α parameter of the flow-based distance measure were optimized simultaneously.

Moreover, two curves are plotted for both for the OI method and the flow-based method. The curves labeled"optimal" plot the RMS errors (with respect to the ground truth) for reconstructions whose parameter configurations were found using dynamic parameter optimization with respect the the ground-truth-based objective functions themselves (e.g., the surface depicted in Figure 10a). These curves illustrate the lowest possible RMS errors for each method when applied to the problem corresponding to the diffusivity indicated (ignoring the fact that we don't have a way to find the optimal parameters when the ground truth is not known). As can be seen in the plot, for test cases constructed using a wide range of diffusivities, the flow-based method always has a better optimal reconstruction than the standard OI method.

On the other hand, the curves labeled "using cross validation" plot the RMS errors for reconstructions computed using the parameters found by the dynamic parameter optimization process applied to leave-one-out cross-validation-based objective functions constructed with reference only to the scattered samples themselves (e.g., the surface depicted in Figure 10b). Note that while cross-validation was used to actually find the parameter configurations to use for each case, the RMS errors shown are those of the final reconstruction *with respect to the ground truth*. These curves illustrate the the lowest RMS errors achievable by each method in practice, using our parameter optimization procedure and only the data typically available. As can be seen from Figure 11, when using cross validation, the flow-based method yields superior results over a large range of diffusivities.

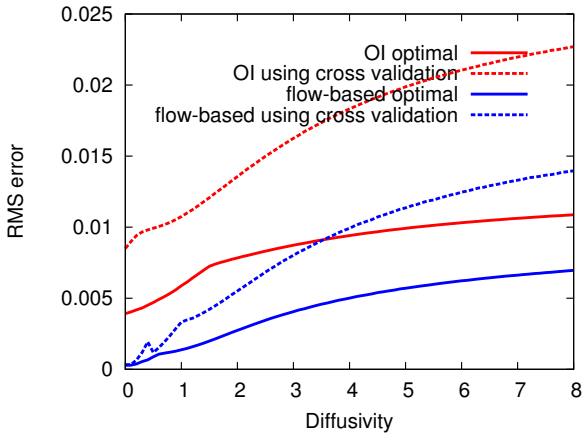


Figure 11: Comparison of the RMS errors for reconstructions computed using the flow-based method and using the standard Optimal Interpolation algorithm. Reconstructions were computed using the sample locations shown in Figure 7 and for the same flow field, but for a range of different diffusivities. Comparing the curves for the two methods when optimized against the ground truth, we see that the flow based method offers the possibility for superior results as long as the free parameters can be chosen appropriately. When these parameters are chosen using leave-one-out cross validation (using only the available sample points to construct the objective function to be minimized), the flow-based method does in fact achieve superior results compared to OI. If the parameter adjustment process can be improved even further, even better results may be possible.

Furthermore, considering the gap between the curve for the optimal flow-based method and the curve for the flow-based method using cross validation (especially for higher diffusivities), we see that even better results may be possible. If the cross-validation-based parameter adjustment process can be improved upon, then the low-RMS-error reconstructions that evidently are possible with use of flow-based distance measures (with the appropriate parameter settings) may possibly be achievable in practice. It remains as future work to explore how to enhance the process to get closer to the optimal parameter configuration using only the sparsely distributed samples that are known.

5.3 Application to Oceanographic Problems

5.3.1 Effectiveness for Water Mass Boundaries

While the proposed method for flow-based scattered data interpolation (and approximation) does not explicitly consider boundaries between separate regions of the interpolation domain, it nevertheless results in reasonable reconstructions in the vicinity of such boundaries. One example of such a situation involves the different water masses that are present in the ocean. Large gradients of tracer values tend to exist across the boundary between one water mass and another. Because the direction of water flow tends to be approximately parallel to a water mass boundary, using a relatively low value of α with our flow-based interpolator provides a way to ensure that interpolated values at a given point near a boundary are constructed in such a way that greater weight is given to the values at sample points on the same side of the boundary. Hence, even though the method does not reconstruct the boundary explicitly, the reconstructed scalar field can be better aligned with the boundary than would be possible using a non-flow-based reconstruction method.

For example, Figure 12 illustrates a simple case for which the scalar field takes the value 0.8 above a horizontal boundary and the value 0.2 below it (as depicted in Figure 12a). The flow direction is to the right in the region above the boundary and to the left in the region below the boundary, and 40 sample points are available to be used as input to a scattered data interpolation algorithm. When standard Optimal Interpolation is used as the algorithm (optimizing for the best value of the correlation length parameter), the shape of the water mass boundary is not well-represented in the resulting reconstruction (Figure 12b). However, as shown in Figure 12c, the flow-based method – optimized simultaneously for both the weighting factor α and the correlation length – produces a reconstruction that more faithfully represents the boundary between the two regions.

Note that the reconstruction would be identical if the flow were in the same direction in both regions (because the streamlines would be the same). Hence, it should be clear that the flow-based method's effectiveness at boundaries is related to how it weights sample points in relation to flow direction, rather than to any explicit identification of boundaries. Because flow is parallel to boundaries, using a value of α less than one will tend to preserve any preexisting difference in the average tracer value above the boundary compared to below it. While this idealized problem may not seem to be a very difficult one, given the information present in the flow field and the clear separation of scalar field values above and below the boundary, the example nevertheless serves to illustrate how the flow-based reconstruction method is able to exploit known information in a way not possible with a non-flow-based method.

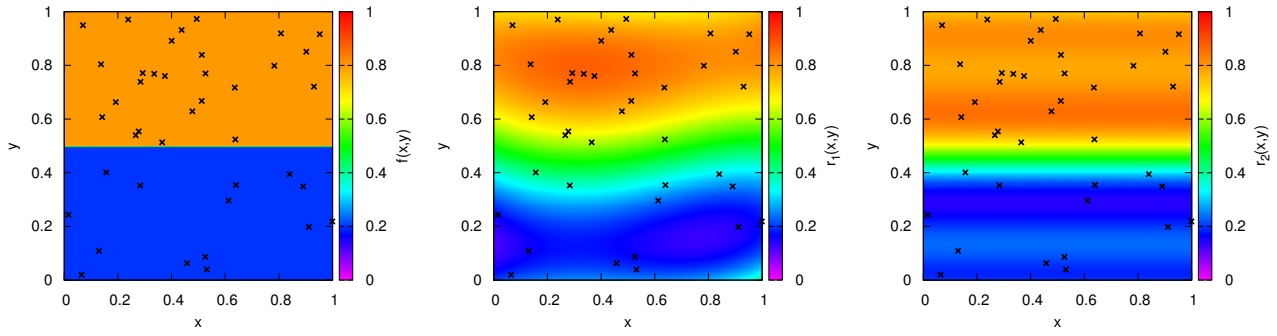
5.3.2 Effectiveness for Domain Boundaries

In addition to its advantageous properties with respect to water mass boundaries, our method also is naturally well-suited to handling domain boundaries. In the oceanographic application domain, these boundaries are defined by the ocean surface and the ocean bathymetry. Because the bathymetry involves a considerable amount of structure imparted by such things as ocean ridges, the reconstruction problem has an added complexity not present when reconstructing fields on a simple rectangular domain.

For example, when using a typical interpolation method (with a Euclidean distance metric), the distance between two points on opposite sides of a ridge will be measured straight through the ridge, and therefore the correlation between the scalar field values these points will be computed to be stronger than it should be. The result is an unphysical bleeding through of the reconstructed scalar field from one side of the ridge to the other. In order to prevent this, one of several largely ad hoc procedures would have to be implemented.

On the other hand, because streamlines of a flow field cannot pass through such a ridge, the flow-based reconstruction method we have proposed handles such complexities in a natural way, without the need for significant ad hoc modifications. For two points on opposite sides of a ridge, if the streamlines passing through the two points both pass over the ridge, then the flow-based method yields good results essentially for free. If one or both of the streamlines do not extend to both side of the ridge, a naive implementation will exhibit unphysical artifacts in the vicinity of the ridge. However, the flow-based method provides a natural way to handle this case as well, requiring only a fairly simple modification to the implementation.

Namely, if the shortest straight line from one point to the streamline passing through the other point crosses a domain boundary, we define the components of the distance as follows. The across-streamline distance is defined to be the length of the shortest line segment between the two streamlines that is entirely within the spatial domain, and the distance in the streamline direction is defined to



(a) (b) (c)

Figure 12: Using the flow-based interpolation technique to better capture tracer values on either side of an idealized water mass boundary. Figure 12a illustrates the scalar function of interest, $f(x, y)$, which involves one constant value (0.8) above the boundary and a different constant value (0.2) below it. Flow is in the rightward direction above the boundary and in the leftward direction below the boundary. The markers indicate the location of each of the 40 samples of the scalar field that are available as inputs to a scattered data interpolation algorithm. Figure 12b illustrates the reconstruction $r_1(x, y)$ obtained using Optimal Interpolation as the scattered data interpolation algorithm, optimizing for the best correlation length. Figure 12c shows the reconstruction $r_2(x, y)$ obtained using our flow-based scattered data interpolation method, optimizing both for the weighting factor α and for the correlation length simultaneously.

be the sum of the streamline lengths from each of the two points to the endpoint of the aforementioned line segment that is on its own streamline. In this way, all distance measurements are performed within the spatial domain, and therefore the resulting reconstruction is well-behaved with respect to the domain boundaries.

5.3.3 Approximating a Tracer Field from Oceanographic Core Data

As a final example we show (in Figure 13) the result of applying our flow-based approximation method to scattered carbon isotope data. The dataset consists of measurements of $^{13}C/^{12}C$ (here referred to as $\delta^{13}C$, which is the typical oceanographic term for this quantity). The locations of these measurements correspond to the core locations specified in Figure 1. The flow field used was a simple one representing a coarse approximation of the flow observed in the modern Atlantic Ocean. As can be seen, our method leads to a relatively smooth approximation. Furthermore, the reconstruction appears to respect the $\delta^{13}C$ values at the core locations fairly well considering the level of smoothness exhibited (and considering that for this example, data from all longitudes have been mapped onto a single latitude/depth plane).

While the reconstruction depicted in Figure 13 appears physically realistic in the interior of the domain, the existence of extrapolation artifacts toward the boundaries (in particular, toward the left side and in the bottom right corner) leads to a unrealistic reconstruction in these regions. To handle this problem, extrapolation is handled explicitly via a second pass through the reconstruction algorithm. A set of boundary points are assigned values based upon a linear extrapolation from the convex hull of the sample points (Figure 14). These points are added to the original set of points for a second pass through the flow-based reconstruction algorithm. The result is a continuous reconstruction, depicted in Figure 15, that is better behaved toward the domain boundaries and therefore physically more realistic across the entire domain. To reduce extrapolation excursions even further, the same two-pass procedure can be employed, but with the values at the boundary points computed using constant extrapolation from the convex hull rather than linear extrapolation.

For real reconstruction problems such as this one, it is difficult to compare the effectiveness of the flow-based method to the effectiveness of standard methods such as Optimal Interpolation because the ground truth (the actual scalar field over the entire domain of interest) is unknown. However, one possibility is to utilize the leave-one-out cross validation version of the reconstruction error, $RMSE_{CV}$, from Eq.(3). Such an approach would compare the $RMSE_{CV}$ value for the flow-based method (for the final, op-

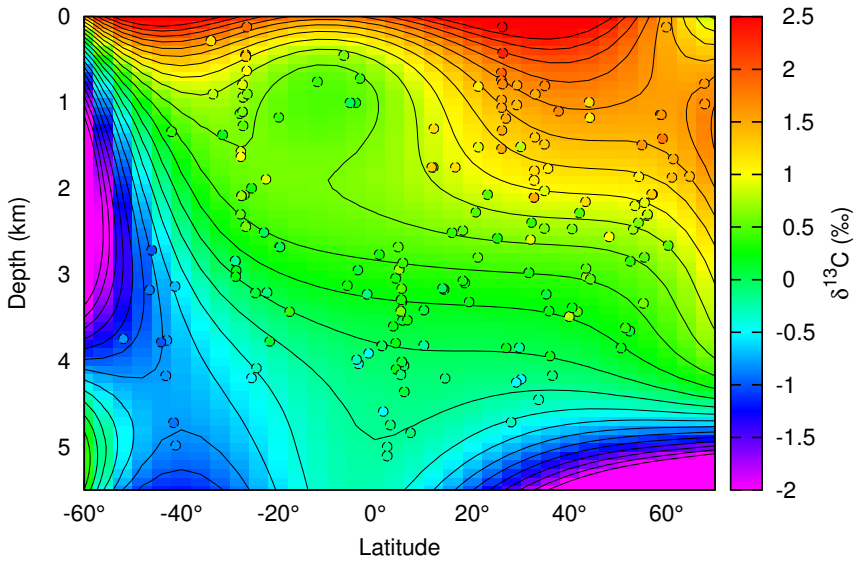


Figure 13: Reconstruction of the δ^{13}C scalar field generated using the flow-based approximation technique applied to δ^{13}C data at the core locations depicted in Figure 1. The δ^{13}C values at the core locations are indicated by the colors inside the plotted circles. The lines in the plot depict isocontours of the reconstruction.

timized parameter configuration) to the $RMSE_{CV}$ value for standard OI. While such a comparison is not definitive, a lower value of $RMSE_{CV}$ for the flow-based method would at least give some evidence that the corresponding reconstruction is superior to the baseline OI reconstruction. However, because of the errors introduced by mapping the 3D data of this example onto a 2D plane, such assessments are best done with a 3D version of the method, and therefore have been left as future work.

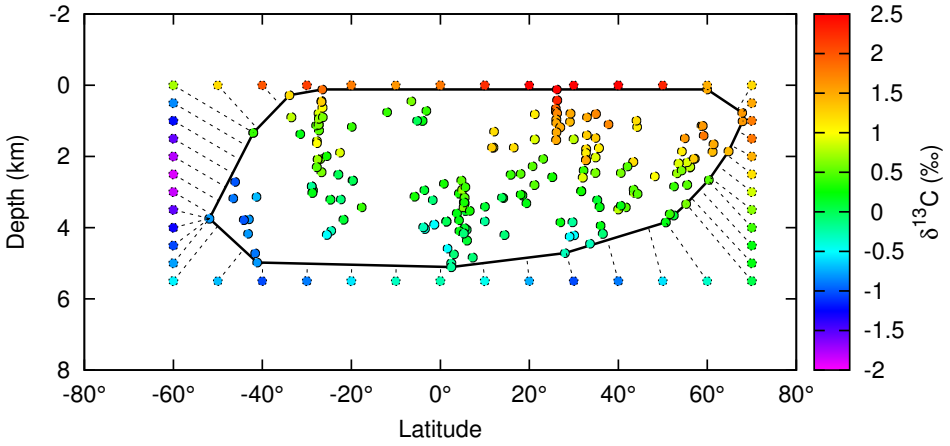


Figure 14: In order to reduce extrapolation excursions toward the boundaries of the domain, a convex hull is formed around the sample points and values are assigned at the boundary of the domain using linear extrapolation from the initial reconstruction inside the convex hull.

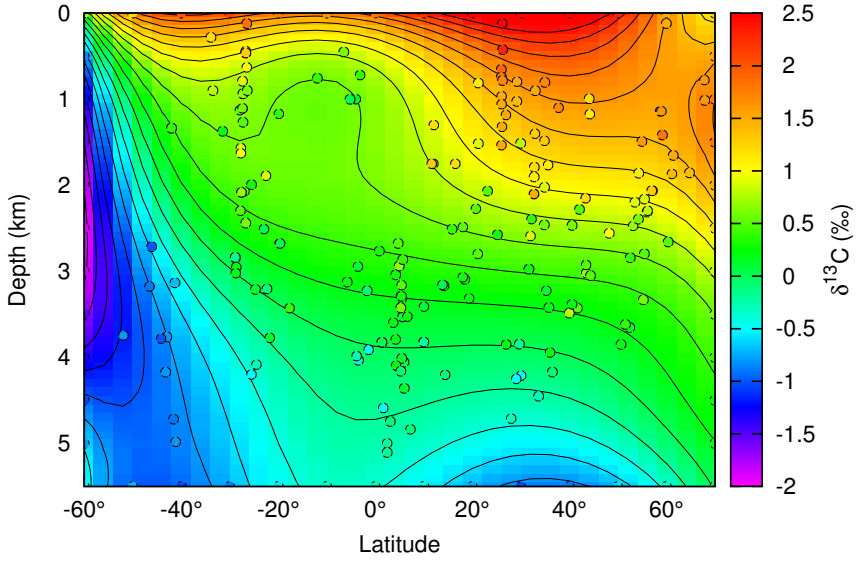


Figure 15: The final reconstruction is computed using a second pass through the flow-based approximation algorithm, using the same parameters as for the initial reconstruction but adding the boundary points from Figure 14 to the original set of sample points. This process leads to a continuous reconstruction that exhibits fewer excursions in the extrapolation region, and therefore is more physically realistic.

6 Conclusions and Future Work

The results presented have shown that the proposed method for scattered data interpolation and approximation is more effective than the corresponding non-flow-based method. Specifically, for a relatively sparse set of samples of a scalar field, using flow-field motivated generalized distance metrics with the Optimal Interpolation method of reconstruction typically leads to better fidelity with the true scalar field than is obtained by using the standard (Euclidean-distance-based) Optimal Interpolation method. Also, we have noted that proper optimization of the free parameters of the interpolation method is critical to obtaining good reconstruction results. Toward this end, we have demonstrated a cross-validation-based approach to searching for near-optimal parameter configurations.

For example, the results shown in Figure 11 indicate that for the family of test cases presented there, using the flow-based method with parameters found by cross validation leads to an average decrease of approximately 0.01 in RMS error (across all diffusivities considered) compared to the RMS errors obtained when using standard OI with cross validation for the same problems. For the range of diffusivities considered, the RMS errors for the OI reconstructions range from around 0.01 to around 0.023, so a decrease of 0.01 in RMS error is quite significant in a relative error sense. Furthermore, note that the reduction in relative error is greatest for the cases involving lower diffusivities (for which advection dominates diffusion), which makes sense considering that the flow-based method has been designed to exploit the existence of higher correlations in the direction of flow.

The flow-based scattered data interpolation and approximation method described in this paper can be enhanced in several ways, and these will be investigated in future work. For example, while the current method exploits correlations with flow only via the streamlines of the flow field, a straightforward extension would be to generalize this so that flow magnitudes are considered explicitly. Another possible generalization is to allow the tunable α parameter to vary locally in space rather than being a global parameter for each reconstruction problem.

With regard to parameter optimization, a more sophisticated approach to global optimization would lead to fewer instances of falling into local minima of the objective function, which in turn might result in the identification of parameter configurations closer to the optimal ones. Many algorithms for global optimization exist [25], many of which would likely work well for our purposes. In addition, a more careful analysis of the relationship between objective functions constructed with respect to the ground truth and those constructed using leave-one-out cross validation might lead to improved methods for parameter selection. Furthermore, the cross-validation-based objective function itself could be refined to include enhancements such as weighting the individual terms to adjust for the relative proximity of sample points (perhaps using a Voronoi tessellation).

Also, while this paper considered the method only in its 2D incarnation, the generalization to 3D will allow the method to be used directly for 3D ocean reconstruction problems (and will facilitate comparisons with standard methods for such real-data problems, as was mentioned in Section 5.3.3). Likewise, the extension to 4D would allow treatment of time-varying problems. Finally, while the current method exploits a known flow field to enhance the fidelity of reconstruction of a related scalar field, it would be of interest to explore the degree to which an unknown flow field could be inferred based only on scattered samples of the scalar field.

7 Acknowledgements

This project was supported by NSF awards 1125422 (HJS, BH, LHK, OK) and 1124880 (GG).

References

[1] O. Marchal and W.B. Curry, 2008: On the abyssal circulation in the glacial Atlantic. *J. Phys. Oceanogr.*, **38**, 2014-2037.

[2] R. Franke, 1982: Scattered data interpolation: tests of some methods. *Mathematics of Computation*, **38**(157), 181-200.

[3] D. Shepard, 1968: A two-dimensional interpolation function for irregularly spaced data. *Proc. 23rd ACM Nat'l Conf.*, 517-524.

[4] R.L. Hardy, 1971: Multiquadric equations of topography and other irregular surfaces. *Journal of Geophysical Research*, **76**(8), 1905-1915.

[5] Y. Ohtake, A. Belyaev, and H.P. Seidel, 2003: A multi-scale approach to 3D scattered data interpolation with compactly supported basis functions. *Shape Modeling International, 2003*, 153-161.

[6] R. Sibson, 1980: A vector identity for the Dirichlet tesselation. *Math. Proc. Cambridge Philosophical Soc.*, **87**(1), 151-155.

[7] S.W. Park, L. Linsen, O. Kreylos, J.D. Owens, and B. Hamann, 2006: Discrete Sibson interpolation. *IEEE Transactions on Visualization and Computer Graphics*, **12**(2), 243-253.

[8] A. Eliassen, 1954: Provisional report on calculation of spatial covariance and autocorrelation of the pressure field.

[9] L.S. Gandin, 1966: Objective analysis of meteorological fields. Translated from the Russian. Jerusalem (Israel Program for Scientific Translations), 1965. *Q.J.R. Meteorol. Soc.*, **92**(393), 447.

[10] F. Bretherton, R. Davis, and C. Fandry, 1976: A technique for objective analysis and design of oceanographic experiments applied to MODE-73. *Deep-Sea Res*, **23**, 559-582.

[11] P. Courtier, E. Andersson, W. Heckley, J. Pailleux, D. Vasiljevic, M. Hamrud, A. Hollingsworth, F. Rabier, and M. Fisher, 1998: The ECMWF implementation of three-dimensional variational assimilation (3D-Var). I: Formulation. *Quarterly Journal Royal Met. Society*, **124**(550), 1783.

[12] F. Rabier, H. Jrvinen, E. Klinker, J.-F. Mahfouf, and A. Simmons, 2000: The ECMWF operational implementation of four-dimensional variational assimilation. I: Experimental results with simplified physics. *Q. J. R. Met. Soc.*, **126**(564), 1143.

[13] G. Gebbie and P. Huybers, 2010: Total matrix intercomparison: a method for resolving the geometry of water-mass pathways. *J. Phys. Oceanogr.*, **40**, 1710-1728.

[14] G.M. Nielson and T.A. Foley, 1989: A survey of applications of an affine invariant norm. *Mathematical Methods in Computer Aided Geometric Design*, 445-467.

[15] PSU/NCAR Mesoscale Modeling System Tutorial Class Notes and User's Guide: MM5 Modeling System Version 3. http://www.mmm.ucar.edu/mm5/documents/MM5_tut_Web_notes/tutorialTOC.htm.

[16] R. Kimmel and J.A. Sethian, 1998. Computing geodesic paths on manifolds. *Proc. Natl. Acad. Sci. USA*, **95**(15), 8431-8435.

[17] N. Sprynski, N. Szafran, B. Lacolle, and L. Biard, 2008. Surface reconstruction via geodesic interpolation. *Computer-Aided Design*, **40**, 480-492.

[18] M.F. Carfora, 2007. Interpolation on spherical geodesic grids: A comparative study. *Journal of Computational and Applied Mathematics*, **210**(1-2), 99-105.

[19] K.E. Atkinson. *An Introduction to Numerical Analysis*. John Wiley & Sons, New York, second edition, 1989.

[20] L.A. Steen and J.A. Seebach, Jr. *Counterexamples in Topology*, Dover Publications, New York, 1995 [1978].

[21] R. Hooke and T.A. Jeeves, 1961. "Direct search" solution of numerical and statistical problems. *Journal of the Association for Computing Machinery (ACM)*, **8**(2), 212-229.

[22] T.G. Kolda, R.M. Lewis, and V. Torczon, 2003. Optimization by direct search: new perspectives on some classical and modern methods. *SIAM Review*, **45**(3), 385-482.

[23] R.G. Miller, 1974: The jackknife–a review. *Biometrika*, **61**, 1-15.

[24] W.R. Young, P.B. Rhines, and C.J.R. Garrett, 1982: Shear-flow dispersion, internal waves and horizontal mixing in the ocean. *J. Phys. Oceanogr.*, **12**, 515-527.

[25] R. Horst and P. Pardalos (eds.). *Handbook of Global Optimization*, Springer, Berlin, 2013.