Natural neighbor extrapolation using ghost points

Tom Bobach [a,*], Gerald Farin [b], Dianne Hansford [b], Georg Umlauf [a]

[a] *Department of Computer Science, University of Kaiserslautern, P.O. Box 3049, 67653 Kaiserslautern, Germany*
[b] *School of Computing and Informatics, Arizona State University, P.O. Box 878809, Tempe, AZ 85287-8809, USA*

ARTICLE INFO

ABSTRACT

Among locally supported scattered data schemes, natural neighbor interpolation has some unique features that makes it interesting for a range of applications. However, its restriction to the convex hull of the data sites is a limitation that has not yet been satisfyingly overcome. We use this setting to discuss some aspects of scattered data extrapolation in general, compare existing methods, and propose a framework for the extrapolation of natural neighbor interpolants on the basis of dynamic ghost points.

© 2008 Elsevier Ltd. All rights reserved.

1. Introduction

Scattered data interpolation is understood as the task of finding a smooth function that takes prescribed values, the data, at a certain set of points in space, the data sites. Among locally supported scattered data schemes, natural neighbor interpolation (NNI) has some unique features that makes it interesting for a range of applications. However, its restriction to the convex hull of the data sites is a limitation that has not yet been satisfyingly overcome. We use this setting to discuss some aspects of *scattered data extrapolation* in general, and address them in a framework for the extrapolation of natural neighbor interpolants.

It is often implicitly assumed that interpolation is restricted to the "intuitive interior" of the data sites, which is usually their convex hull. Extrapolation in this context means to find a function that also extends to the "intuitive exterior" of the data sites, i.e., to interpolate over a domain that extends past the convex hull. Extrapolation can be achieved in multiple ways. An interpolant might be globally defined, like radial basis function (RBF) or inverse distance weighted (IDW) interpolants, in which case its evaluation outside the convex hull amounts to extrapolation. Interpolation schemes with limited domain can be extended by constructing a function that smoothly joins the interpolant at the boundary of its domain.

While extrapolation in itself is an ill-posed problem, two different objectives can be identified when dealing with it:

The first is to construct pleasant-looking functions (surfaces) that leave the transition between the intuitive interior and exterior unnoticed. Applications for this are the visualization of unstructured data over a rectangular domain, like weather forecast data or digital elevation models in GIS and architecture. The second objective is to construct a function that agrees with application specific expectations that encode knowledge about the application domain. This is typically given in the context of open boundary simulations, oil exploration [26,40], or biosciences [27], where properties of the underlying physical model play a role in the interpolation method. Another possible application emerges in the modeling of binder surfaces for stamping, which must be extended beyond the surface to be stamped. To facilitate the classification of extrapolation methods, we propose to adapt a set of criteria that has previously been introduced by Alfeld in a technical report on triangular extrapolation methods [3].

Our efforts to extend natural neighbor interpolation past the convex hull of the data sites resulted in a framework that alleviates some of the issues present in current extrapolation methods. Based on dynamically inserted new data sites, the ghost points, we are able to extend any natural neighbor interpolant over an arbitrary data set to all of space, while preserving desirable properties like smoothness or the continuous dependence on the coordinates of the data sites.

We start with a brief introduction of relevant scattered data interpolation methods and present previously proposed approaches for the extrapolation of local scattered data interpolants. We then present a set of criteria that allows a meaningful classification of extrapolation approaches and apply it to compare global scattered data interpolants, extension schemes, and the later introduced ghost point method for natural neighbor interpolation. Finally, we introduce our framework for natural neighbor extrapolation with the help of dynamically constructed ghost points.

* Corresponding author. Tel.: +49 631 205 3800.
E-mail addresses: bobach@informatik.uni-kl.de (T. Bobach), farin@asu.edu (G. Farin), dianne.hansford@asu.edu (D. Hansford), umlauf@informatik.uni-kl.de (G. Umlauf).

2. Related work

Extrapolation is strongly linked to interpolation in that, either, interpolation methods with restricted definition domains are extended or, interpolation methods naturally cover a domain that is larger than the convex hull of the data sites. We therefore briefly review relevant scattered data interpolation in general and point-based natural neighbor interpolation in particular in Section 2.1. Extrapolation methods are then discussed in Section 2.2, with a classification of some common extrapolation approaches presented in Section 2.3.

For the rest of this paper, we drop explicit summation bounds for ease of notation and assume that indices address the feasible, finite range.

2.1. Scattered data interpolation

The field of scattered data interpolation is extremely vast and has been around since the mid-seventies. Surveys on scattered data interpolation in general by Franke and Nielson can be found in [16, 17,19], on tessellation-based interpolation by Zeilfelder and Seidel in [39], and a textbook on RBF approaches by Wendland in [38]. We consider the scattered data interpolation problem for a set of data sites $\mathbf{X} = \{\mathbf{x}_i\}_i, \mathbf{x}_i \in \mathbb{R}^n$, and data $T = \{t_i\}_i$, where $t_i(\mathbf{x})$ is a Taylor polynomial of given degree $k \in \{0, 1, 2\}$ carrying the partial derivative information for data site $\mathbf{x}_i$, such that $t_i(\mathbf{x}-\mathbf{x}_i)$ is a local approximation of the unknown function. Given this input, an interpolation method I provides an n-variate, real-valued function $f = I(\mathbf{X}, T) \in C^d, d \in \mathbb{N} \cup \{\infty\}$, that satisfies the interpolation property for derivatives up to order k,

$$\left.\frac{\partial^{|\mathbf{j}|} f}{\partial \mathbf{x}^{\mathbf{j}}}\right|_{\mathbf{x}=\mathbf{x}_i} = \left.\frac{\partial^{|\mathbf{j}|} t_i}{\partial \mathbf{x}^{\mathbf{j}}}\right|_{\mathbf{x}=0}, \quad |\mathbf{j}| \leq k,$$

where $\mathbf{j} = (j_1, \ldots, j_n) \in \mathbb{N}_0^n$ with $|\mathbf{j}| = j_1 + \cdots + j_n$. If f is defined over the domain $\Omega \subset \mathbb{R}^n$, then we call $D(I) := \Omega$ the *definition domain* of I. We denote the convex hull of $\mathbf{X}$ by $\mathscr{C}(\mathbf{X})$, and the data sites on the boundary of $\mathscr{C}(\mathbf{X})$ by $\mathbf{X}^{\mathscr{C}}$. In the sequel, we focus on bivariate interpolants.

Most scattered data interpolation schemes require some sort of preprocessing that is performed once before being able to evaluate the interpolant I at arbitrary positions in $D(I)$. The complexity of an interpolation scheme is consequently split into the complexity of preprocessing and the complexity of evaluating the interpolant.

A common preprocessing step is the generation of derivative information for $k > 0$, which is often not part of the input to scattered data interpolation. Although many schemes explicitly define their own estimation procedure, it is in general decoupled from the interpolation. Appropriate methods for derivative generation can be found in [34,28,2,35,4].

The essential step common to all interpolation methods is the computation of a value for a query position $\mathbf{p}$, which is characterized by the number of data sites involved in the computation. Global schemes are more expensive in that they process all input data in each evaluation, but generally allow for a higher smoothness, while local schemes have lower computational complexity at the expense of only a limited degree of continuity.

2.1.1. Global schemes

The most prominent global interpolation scheme is the radial basis function (RBF) approach that originates from Hardy's multiquadric interpolation [21], and received in-depth theoretical analysis in [38]. The interpolant in its most general form is given by

$$f(\mathbf{x}) = \sum_i c_i \phi(\|\mathbf{x} - \mathbf{x}_i\|/d_i) + p(\mathbf{x}),$$

where $\phi : \mathbb{R} \to \mathbb{R}$ is a monotone function, p is a polynomial of given degree, and $d_i \in \mathbb{R}$ is a local scaling factor for the width of ϕ. The RBF coefficients c_i and the coefficients in p are determined by the solution of the linear system resulting from the interpolation condition and the least-squares fit of p.

The RBF approach is versatile due to a multitude of available global, quasi-local, and compactly supported basis functions $\phi(r)$. They determine the overall smoothness of the scheme, the shape of the interpolant, but also the computational complexity of preprocessing and evaluation. RBFs produce smooth interpolants of high quality and have good approximation properties, but do not scale well with the size of the data set. For data sets with heterogeneous data site density, it is advised to adapt the individual scaling factors d_i to the local data density. In our implementation, we choose d_i to be a multiple of the distance of $\mathbf{x}_i$ to its furthest direct neighbor in the Voronoi diagram of $\mathbf{X}$. We will consider the RBF method with following basis functions:

- $I^{\mathrm{RBFT}}, \phi(r) = r^2 \log r$ (thin plate splines), global,
- $I^{\mathrm{RBFG}}, \phi(r) = \exp(-\alpha r^2), \alpha \in \mathbb{R}^+$ (Gaussian), quasi-local,
- $I^{\mathrm{RBFW}}, \phi(r) = (1 - r)_+^4(4r - 1)$, where $(x)_+^4$ is the truncated fourth degree monomial (Wendland's compact polynomial), local.

Another approach to global interpolation is given by inverse distance weighted (IDW) schemes, whose first definition is due to Shepard [32] and which have been improved and adapted ever since. The most significant improvements were the introduction of blended Taylor polynomials at the data sites to overcome the flat spots and the modification of the weight functions to have local support [18]. Unfortunately, the rational nature of IDW schemes causes the resulting interpolant to have far more oscillations than the input data suggests. Despite this, IDW methods are still used for their easy implementation. We will consider the following IDW methods:

- I^{IDWQ}: quadratic Shepard, blending quadratic Taylor polynomials, global,
- I^{IDWMQ}: modified quadratic Shepard, as proposed in [18], local.

2.1.2. Local schemes

One prominent approach for local scattered data interpolation is to use a tessellation of the convex hull of the data sites, $\bigcup_l \Omega_l = \mathscr{C}(\mathbf{X})$, to construct a piecewise defined interpolant, composed of analytic functions $\varphi_l(\mathbf{x})|_{\Omega_l}$ over the tiles Ω_l, where global smoothness is achieved by matching derivatives along the joints. These methods are often referred to as FEM approaches due to their close relation to the Finite Element Method. Most widely adopted due to their simplicity are triangulation schemes, where Ω_l are simplices, and $\varphi_l(\mathbf{x})$ are polynomials or rational functions in the barycentric coordinates over the simplex that interpolate function values and derivatives at the data sites and sometimes at the simplex facets.

The preprocessing for local schemes generally consists in the construction of a tessellation and the generation of derivatives and/or functions φ_l. Sometimes, these result from a global pre-processing step, such as in the minimum norm network approach of Nielson [28], or the global energy minimization approach by Alfeld [4]. A survey of interpolation with triangulations and quadrangulations can be found in the chapter by Zeilfelder and Seidel in [39].

All tessellation-based approaches share one shortcoming: the interpolants do not depend continuously on the coordinates of the data sites. For any particular tessellation method, there always exists an "ambiguous" configuration of points that indicates a change in the topology of the tessellation, and for which the constructed interpolant changes discontinuously. An example of
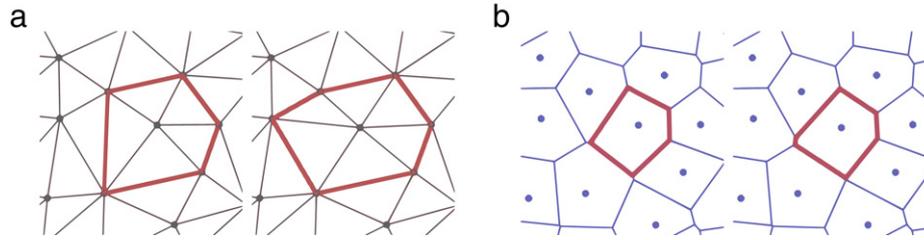
Fig. 1. (a) A slight perturbation in a triangulation can lead to topological changes. (b) The Voronoi diagram of a point set continuously depends on the coordinates of the points.

such a configuration for the Delaunay triangulation is given in Fig. 1(a).

Although it is not an interpolation method per se, we mention the partition of unity (PoU) approach as a meta method allowing to localize any scattered data interpolation scheme. After decomposing a data set into local, sufficiently overlapping subsets, any interpolation method can be applied on each reduced size subset. The individual interpolants are then combined using smooth, compactly supported blending functions that form a partition of unity. This has been discussed by Wendland in [38], Section 15.4. PoU methods have been successfully applied in the processing of very large data sets.

2.1.3. Natural neighbor interpolation

We assume that the reader is familiar with the Voronoi diagram and its dual tessellation, the Delaunay triangulation, which has been treated in, e.g., [29,37]. The natural neighbors of a point $\mathbf{x} \in \mathbf{X} = \{\mathbf{x}_i\}_i$ are all points in $\mathbf{X}$ whose Voronoi tiles share an edge with that of $\mathbf{x}$. There are several interpolation methods based on the concept of natural neighbors. The two basic steps in the evaluation of such an interpolant at a point $\mathbf{x}$ are (1) the computation of natural neighbor coordinates $\lambda(\mathbf{x}) = \{\lambda_i(\mathbf{x})\}_i$ with respect to the natural neighbors $\{\mathbf{x}_i\}_i$ of $\mathbf{x}$, and (2) the blending of Taylor polynomials of degrees k at the data sites using functions $\psi(\lambda(\mathbf{x}))$ of the local coordinates.

Based on special geometric properties of the Voronoi diagram, every point $\mathbf{x}$ inside $\mathscr{C}(\mathbf{X})$ can be expressed as a convex combination of its natural neighbors,

$$\mathbf{x} = \sum_i \lambda_i(\mathbf{x})\mathbf{x}_i, \quad \lambda_i(\mathbf{x}_j) = \delta_{ij},$$

where $\{\lambda_i\}_i$ are called natural neighbor coordinates of $\mathbf{x}$ with respect to $\{\mathbf{x}_i\}_i$. These can be used for interpolation by simply applying the same convex combination to the data at the sites, $f(\mathbf{x}) = \sum_i \lambda_i(\mathbf{x})t_i(\mathbf{x})$. Simple interpolants operating on scalar values $t_i \in \mathbb{R}$ are those of Laplace ([8,36,5], C^0), Sibson ([33], C^1), and Hiyoshi ([23], $C^{\geq 2}$). In general natural neighbor coordinates are C^0 at the data sites, so the continuity listed above refers to the continuity of the interpolant in $\mathscr{C}(\mathbf{X}) \setminus \mathbf{X}$.

The above methods have been generalized to globally smooth natural neighbor interpolants by Sibson ([34], C^1), Farin ([14], C^1), and Hiyoshi ([24,22], C^2). All these methods incorporate given derivatives of the function at the data sites to construct local polynomials that blend between the linear respective quadratic local approximations.

Further, methods that do not directly fit into the above scheme are those of Clarkson ([9,15], C^0), Brown ([6], C^0), and Gonzales et al. ([20], C^0). Clarkson and Gonzales et al. define barycentric coordinates on a larger neighborhood based on the natural neighbor relation. Brown proposes a PoU method on Delaunay circumcircles to blend barycentric coordinates with respect to Delaunay triangles, which results in non-convex, local coordinates with respect to the natural neighbors of a point. For a special

choice of rational, non-positive blending function, his method yields Sibson coordinates.

Due to the duality of Voronoi diagrams and Delaunay triangulations, all natural neighbor interpolants can be computed directly from the Delaunay triangulation of the data sites, such that the preprocessing amounts to the construction of the Delaunay triangulation and a possible generation of derivative data. The evaluation of natural neighbor interpolants is inherently local.

The most intriguing property of natural neighbor interpolation is the continuous dependence of the interpolant on the coordinates of the input data, which results from the corresponding property of the Voronoi diagram, which is illustrated in Fig. 1(b). Further, major advantages of natural neighbor interpolation are

(+1) The definition of neighborhood is local, completely automatic, and copes extremely well with inhomogeneous point distributions.

(+2) Most interpolants based on C^k-continuous natural neighbor coordinates depend C^k-continuously on the coordinates of the point cloud, which has been shown for Sibson coordinates in [31].

(+3) The Voronoi diagram needs not be constructed at any time since all operations can be carried out on the Delaunay triangulation of the data sites, which is a very well understood and supported by an efficient data structure.

(+4) By definition, the interpolants generalize to any dimension.

However, natural neighbor interpolation has disadvantages as well, namely

(-1) They are defined only inside the convex hull of the data sites, with undesirable artifacts near the boundary of the convex hull. A solution to this is presented in Section 3.

(-2) Globally smooth interpolants can be relatively expensive to evaluate.

(-3) Evaluating natural neighbor interpolants in higher dimensions is computationally expensive.

We will consider the following natural neighbor interpolants in the later discussion of extrapolation.

- I^{NNS} : Sibson's C^0 interpolant as proposed in [33],
- I^{NNF} : Farin's C^1 interpolant as proposed in [14],
- I^{NNH} : Hiyoshi's C^2 interpolant as proposed in [24].
- I^{BRO} : Brown's C^0 interpolant as proposed in [6].

2.2. Scattered data extrapolation

Though the meaning of extrapolation varies from application to application, it essentially refers to the extension of a given interpolant from the "intuitive interior", to the "intuitive exterior". Thus, for a given interpolation method I, defined over the domain $D(I)$, we denote by $E(I)$ the result of applying an extrapolation method E to extend the interpolant to the *extended domain* $D^X(E; I)$. For practical reasons, we ignore subtle differences in the definition of $D(I)$ for different interpolation schemes I and assume interpolation in general to refer to $\mathscr{C}(\mathbf{X})$, which allows us to
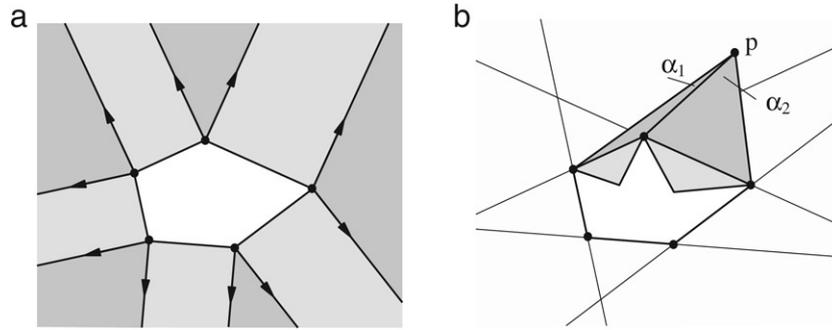
Fig. 2. (a) Partition of the complement of the convex hull into cones and half-open prisms (wedges and half-open rectangles in 2D), as used in E^{AKI} and E^{FRA}. (b) To evaluate at **p** using E^{ALFB}, the functions from the light gray triangles adjacent to the convex hull are extrapolated and mixed based on the angles α_1 and α_2.

include RBF and IDW methods in a later comparison. Consequently, globally defined interpolants I are interpreted as the result of applying a dummy extrapolation method $E^{\text{ID}}(I) = \text{Id}(I)$ that extends I from $\mathscr{C}(\mathbf{X})$ to its original domain $D(I)$.

The common approaches for scattered data extrapolation are discussed in the following sections.

2.2.1. Ghost Point Methods

Ghost Point Methods generate new data sites $\mathbf{X}^G \subset \mathbb{R}^n$, called ghost points, such that $\mathscr{C}(\mathbf{X}^G)$ covers an extended domain, over which available interpolants can be applied.

A method to improve piecewise linear interpolation over Delaunay triangulations has been proposed by Lasser and Stüttgen in [25] who essentially place ghost points along a rectangle enclosing the data set. Similar methods addressing the extension of Delaunay triangulations past their convex hull have been discussed in [29], Section 6.3, where it was concluded that the quality of the resulting triangulation depends on the placement and number of "imaginary points".

A similar conclusion about the number of ghost points was drawn by Alfeld in [3]. He proposed to construct an interpolant over an extended domain by introducing ghost points leading to a new triangulation of the data sites and ghost points. A smooth triangular interpolant is then constructed over the extended domain as the solution of a global energy minimization, e.g., the minimization of the thin-plate energy, based on the method from [4]. We will refer to this extrapolation approach as E^{ALFT}.

In Section 3, we introduce a novel ghost point framework for natural neighbor extrapolation. In [30], Owen applied a technique similar to our proposed approach to provide limited extrapolation for Sibson's C^0 interpolant. By placing the data set in an appropriately sized bounding window and clipping all Voronoi tiles against it, he computes affine weights that allow for interpolation. This method has also been implemented in [13].

2.2.2. Convex Hull Extension Methods

Convex Hull Extension Methods partition the complement of the convex hull into unbounded regions consisting of cones extending from convex hull vertices and half-open prisms extending from convex hull facets, as shown in Fig. 2(a). In each region, a function is defined that smoothly joins its neighbors.

One such method was proposed by Akima in [1], which operates solely on the corresponding interpolant proposed in the same publication. In each wedge region, the corresponding Taylor polynomial at the wedge vertex is extrapolated. In the local coordinate system of each rectangular region, a polynomial is constructed based on the special structure of Akima's triangular interpolation scheme. The result is a C^1 function over $\mathbb{R}^2$, and we refer to Akima's method as E^{AKI}.

Franke's transfinite extension, introduced in [16], is another approach operating on the extension of the convex hull. It can be applied to any interpolant for which Taylor polynomials at the convex hull vertices and normal derivatives along the convex hull edges exist. Franke treats wedge regions like Akima. To evaluate the function at a point **p** in a rectangular region, **p** is projected to the convex hull edge and the univariate Taylor polynomial determined by the normal derivatives is extrapolated to compute a function value at **p**. We refer to this method as E^{FRA}.

We finally mention the C^{-1}-continuous nearest neighbor interpolant which also covers all of $\mathbb{R}^n$ and which is constant over each Voronoi tile.

2.2.3. External Blending Methods

External Blending Methods extend particular functions of a piecewise interpolant to overlapping regions in the complement of the convex hull and blend them in a Lagrangian interpolation manner. The relation between convex hull extension and external blending methods is analogous to the relation between FEM interpolants which are constructed to smoothly join local functions along the piecewise domain boundaries, and PoU methods which combine overlapping, local interpolants using smooth blending functions.

Alfeld proposed in [3] a method to extrapolate tessellation-based interpolation schemes past the convex hull. From a point **p** outside the convex hull of the data sites, a set of convex hull edges, and consequently the adjacent elements Ω_i of the tessellation, is visible, see Fig. 2(b). The lines pointing from **p** to the vertices of a visible convex hull edge form an angle α_i at **p**. The local function φ_i over each element Ω_i can usually be evaluated at **p**, thus extrapolating the local function. After the functions of all visible elements have been extrapolated, the function values are mixed using ratios proportional to some power of α_i. This way, Alfeld is able to smoothly extrapolate any tessellation-based interpolant past the convex hull of the data sites. We will refer to this interpolant as E^{ALFB}.

Brown proposed a local coordinate system defined over Delaunay triangulations in [6], for which he gave a construction past the convex hull of the data sites. For a point contained in the circumcircles of a set of Delaunay triangles $\{\Omega_i\}_i$, the barycentric coordinates with respect to each triangle Ω_i are computed, and blended in a PoU fashion using smooth, positive blending functions over the corresponding circumcircles. This method naturally extends the coordinates to the interior of circumcircles that reach outside the convex hull. For points even further outside, he proposed to blend the barycentric coordinates with respect to all boundary-adjacent triangles in a Lagrangian interpolation manner based on the distance to the corresponding circumcircles. The resulting generalized barycentric coordinates are then used to

build a simple C^0 scattered data interpolant with linear precision. This method will be referred to as E^{BRO}.

The summary of local extrapolation methods considered in the sequel is

- E^{AKI}: Akima's discrete extension [1], a convex hull extension method,
- E^{FRA}: Franke's transfinite extension [16], a convex hull extension method,
- E^{ALFB}: Alfeld's blending of visible edges [3], an external blending method,
- E^{ALFT}: Alfeld's global triangular thinplate minimization [4], which can be interpreted as a ghost point method in which the ghost points globally influence the interpolant.
- E^{BRO}: Extension of Brown's coordinates in an unstructured point set [6].

2.2.4. Global methods

Global methods apply some global interpolation that naturally extends to all $\mathbb{R}^n$. In particular, this entails the RBF and IDW methods introduced in Section 2.1. Extrapolation of these interpolants requires no modification, and for sake of consistency we denote by $E^{\text{ID}}(I) = \text{Id}(I)$ the extrapolation method applied to one of these schemes. Away from $\mathscr{C}(\mathbf{X})$, RBFs with local and quasi-local support asymptotically approach the supporting polynomial, while global ones generally diverge dramatically. It must be noted that for IDW interpolants I with compactly supported weight functions, $D^X(E^{\text{ID}}, I)$ is finite.

2.3. Taxonomy of scattered data extrapolation

To allow an individual comparison of existing extrapolation methods, we adopt and extend the classification of extrapolation approaches introduced by Alfeld in [3], and propose the following criteria, where the first six correspond to Alfeld's. The results of this classification applied to a number of approaches introduced so far is then given in Table 1.

Smoothness How smooth is $E(I)|_{D(I)}$, and how smooth is $E(I)|_{D^X(E;I)}$?

Structure Is the $E(I)$ of the same structure everywhere, i.e., piecewise polynomial of a certain degree, or are interior and exterior different? A similar structure simplifies the analysis of the method.

Reproduction Power Is the class of functions reproduced exactly by I the same as reproduced by $E(I)$? All interpolants and extrapolation approaches we considered here possess some sort of polynomial precision, and we use P^d in the table to indicate reproduction of polynomials of degree d.

Finiteness Is the extended domain $D^X(E; I)$ finite or does it cover all $\mathbb{R}^n$?

Blending Extrapolation Is $E(I)|_{D(I)}$ identical to $I|_{D(I)}$? If an extrapolation method simply continues the interpolant where it ceases to be defined, it is not considered blending. Otherwise, it augments the interpolant in a way to improve the joint between $E(I)|_{D(I)}$ and $E(I)|_{D^X(E;I)\setminus D(I)}$.

Generality How large is the class of interpolants I to which an extrapolation approach E can be applied? An approach is considered general if it can be applied to more than a single, specific interpolation scheme.

Similar Results from Similar Input (sim/sim) Do small perturbations of the input lead to small changes of the output? This property is already important for interpolation alone, and is an indicator for robustness of a particular method.
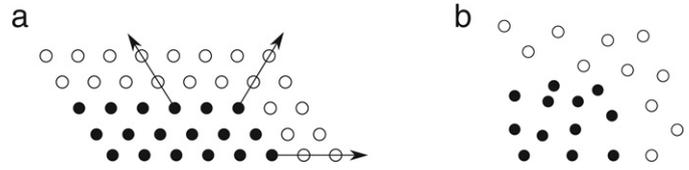

Fig. 3. (a) Data sites (solid) are used to place ghost points (hollow) in a structured setting. (b) Arbitrary ghost point choice in an unstructured setting.

Size of Support How many data sites need to be processed to evaluate the interpolant? Local support (L) corresponds to a small, bounded number of data sites in the vicinity of the evaluation position. Convex hull support (C) refers to the data sites on the boundary of the convex hull and possibly interior data sites nearby. Global support (G) refers to all data sites.

Alfeld also proposed to take the quality of a scheme into account, but because of its subjective flavor we omit this criterion. We note that Brown's method is no competitor for smooth scattered data interpolation, for it was introduced as a local coordinate system in the Delaunay triangulation, and interpolation was added as an afterthought.

The comparison in Table 1 includes the ghost point methods introduced in Section 3 for completeness and reference. The reader might want to proceed there before closer inspection of the table.

The "similar results from similar input" criterion implicitly assumes that the perturbation of data sites leaves the set of vertices on the convex hull, $\mathbf{X}^{\mathscr{C}}$, unchanged. Otherwise, constructions based on convex hull vertices show discontinuous changes under the perturbations. Among the ghost point methods, the DDC approach is not affected by changes in $\mathbf{X}^{\mathscr{C}}$ because the construction of ghost points is independent of the concrete structure of $\mathbf{X}^{\mathscr{C}}$. Furthermore, RBF and IDW schemes are naturally unaffected by changes in $\mathbf{X}^{\mathscr{C}}$.

3. Ghost points for natural neighbor interpolation

In this section we present our ghost point-based modification of natural neighbor interpolants. We first discuss the general idea in Section 3.1, then introduce the concepts of dismissed and assigned ghost points in Section 3.2, and present two concrete placement strategies in Section 3.3.

Our method concentrates on discrete data, i.e., given at points. While it is possible to extend it to transfinite data, we refrain from discussion of such methods in this paper.

3.1. Ghost point idea

The ghost point concept entails the modification/enrichment of the data set such that an interpolation method that is initially defined only in a limited domain can now be applied on a larger domain. We now describe the main issues of extending natural neighbor interpolants using ghost points and relate them to the rationale behind our ghost point framework.

Rigid Invariance. The placement of ghost points should be invariant under rigid transformations. In a structured setting as shown in Fig. 3(a), where data is distributed over a grid, ghost points can trivially be generated by extending the grid. Since such an extension is not available in scattered data as shown in Fig. 3(b), we anchor the ghost point construction at the vertices and edges of the convex hull, which makes the construction invariant under rigid transformations.

Finiteness. Any concrete choice of ghost points leads to an extended domain that is again finite. To allow evaluation of the

Table 1
Classification of extrapolation methods.

	Method	Smoothness (in/out)	Structure (in/out)	Same rep. (in/out)[a]	Domain[b]	Blending	General	Sim. sim. (in/out)	Size of support
	Ideal technique	C^∞	Same	y	$\mathbb{R}^n$	/	y	y	Small
Triangular	Akima (E^{AKI}) [1]	C^1	Diff	n	$\mathbb{R}^n$	n	n	n/y	L
	Franke (E^{FRA}) [16]	As I/C^0	Diff	n	$\mathbb{R}^n$	n	y^{c}	n	L
	Alfeld (E^{ALFB}) [3]	As I	Diff	y	$\mathbb{R}^n$	n	y^{d}	n	C
	Brown (E^{BRO}) [6]	C^0	Diff	$y(P^1)$	$\mathbb{R}^n$	n	n	n	C
Ghost points	- stat. assigned	As I	Same	$n\,(P^d/P^{d-1})$	$\mathscr{C}(\mathbf{X}^G)$	y	y^{e}	y	L
	- dyn. assigned	As I	Diff	$n\,(P^d/P^{d-1})$	$\mathbb{R}^n$	y	y^{e}	y	C
	- stat. dismissed	C^0 at $\mathbf{X}^{\mathscr{C}}$	Diff	$n\,(P^d/P^0)$	$\mathscr{C}(\mathbf{X}^G)$	y	y^{f}	y	L
	- dyn. dismissed	C^0 at $\mathbf{X}^{\mathscr{C}}$	Diff	$n\,(P^d/P^0)$	$\mathbb{R}^n$	y	y^{f}	y	C
Global interpolants	Duchon (I^{RBFT}) [12]	C^∞	Same	$y(P^d)$	$\mathbb{R}^n$	/	/	y	G
	Wendland (I^{RBFW}) [38]	C^d	Diff	$y(P^d)$	$\mathbb{R}^n$	/	/	y	L
	Shepard (I^{IDWQ}) [32]	C^∞	Same	$y(P^0)$	$\mathbb{R}^n$	/	y^{g}	y	G
	Nielson (I^{IDWMQ}) [18]	C^d	Same	$y(P^0)$	Finite	/	y^{g}	y	L
	Alfeld (I^{ALFT}) [4]	As I	Same	y	Finite	y	y^{h}	n	As I

Single entries in columns one and three apply to both interior and exterior, different values are separated by a slash. A sole slash indicates that the criterion does not apply.

[a] We assume that required derivatives are exact.
[b] We assume $\mathbb{R}^n$ if extension to higher dimensions is straightforward.
[c] Cross-boundary derivatives must be available.
[d] Local functions over $\mathscr{C}$-elements must be defined over all $\mathbb{R}^n$.
[e] Applies to any method that can deal with point-based data.
[f] Applies to any method utilizing affine weights that are entirely determined by the position of the data sites.
[g] The quadratic approximations at the data sites can be replaced by arbitrary local/global approximations.
[h] The interpolation scheme must be applicable to incomplete input data.

interpolant at an arbitrary position outside the convex hull, the ghost point construction must either recursively continue until the position is covered, or the placement of ghost points must be dynamic in that it takes the actual evaluation position into account. In this paper, we only focus on the second approach of dynamic ghost points.

One major property of natural neighbor interpolation is its continuous dependence on the coordinates of the data sites. By making the ghost point coordinates depend smoothly on the position of the evaluation position, we maintain the continuity of the interpolant even in its extended domain.

Artifact Removal. Tessellation-based interpolants suffer severe artifacts in case of slightly concave data site distributions at the boundary because of long, skinny triangles or polygons, as shown in Fig. 12(a) and (b). Natural neighbor interpolants have similar problems due to the linear precision on the convex hull, shown in Fig. 12(c), where the following observation is useful. The "perfect" data site distribution for natural neighbor interpolation is completely homogeneous, which roughly means the same density of neighbors in every direction. The corresponding "degenerate" case occurs on the convex hull, where the outside completely lacks neighbors. The seamless transition between these two extrema is a core advantage of natural neighbor interpolation that makes it cope so well with very inhomogeneous site distributions. The distance from the convex hull at which ghost points are placed plays a crucial role in overcoming artifacts. We place ghost points such that the original natural neighbor interpolant is augmented near the boundary of the convex hull, where the local data site density should be considered in the computation of an offset distance, and try to provide ghost points that mimic the perfect setting.

The boundary artifacts have in part been overcome by Cueto et al. in [11,10], who applied density-scaled α-shapes that allow the restriction of the domain to a concave shape in which undesired triangles are omitted.

Reproduction Power. While ghost point positions determine the domain over which an extended interpolant is defined, the interpolant itself also depends on the values at the ghost points. In how far the reproduction power of the interpolant is preserved

by the ghost point framework depends on the generated values. To this end, we propose two solutions which we coin "dismissed" and "assigned" ghost points.

3.2. Assigned vs. dismissed ghost points

By default, ghost points do not carry any data besides their coordinates. Two options exists to deal with this issue when evaluating a natural neighbor interpolant that builds on these data.

The first option, called "dismissed ghost points", proceeds as follows. If $\lambda = (\lambda_1, \ldots, \lambda_m) \in \mathbb{R}^m$ are natural neighbor coordinates of $\mathbf{x}$ in

$$\mathbf{x} = \sum_{i=1}^{m} \lambda_i(\mathbf{x})\mathbf{x}_i, \quad \mathbf{x}_1, \ldots \mathbf{x}_k \in \mathbf{X}, \; \mathbf{x}_{k+1}, \ldots, \mathbf{x}_m \in \mathbf{X}^G,$$

and $k > 0$, then $\gamma = (\lambda_1, \ldots, \lambda_k)/(\lambda_1 + \cdots + \lambda_k) \in \mathbb{R}^k$ is a set of affine coefficients that result from ignoring the ghost point contributions. This situation is shown in Fig. 4. Because they fulfill the Lagrange property $\gamma_i(\mathbf{x}_j) = \delta_{ij}$, these affine coefficients can be used for interpolation in $\psi(\gamma(\mathbf{x}))$. Due to the loss of the local coordinate property, which is crucial in the smooth constructions of Sibson, Farin, and Hiyoshi, the resulting interpolant is only C^0 at the vertices being natural neighbors of ghost points. Along a ray pointing away from the data sites, the interpolant asymptotically converges towards a constant function. However, there might be situations where this behavior is sufficient.

Because no values are required at the data points, we have considerable freedom in their placement, our only restriction being invariance under rigid transformations. We have not yet exploited this particular possibility and provide results for the same placement strategies used in assigned ghost points.

The second option, called "assigned ghost points", lies in generating feasible data at ghost points by extrapolating the Taylor polynomials from the data sites that were used in the construction of the ghost point. This required link between data sites and ghost points imposes some constraints on the construction. The concrete implementation of assigned ghost points depends on the placement strategy; one approach is given by the dynamic convex hull offset strategy explained below.
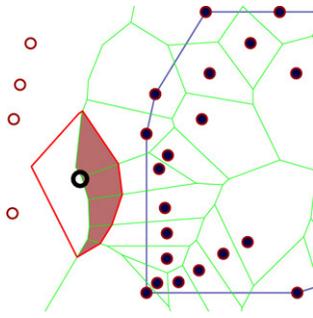
Fig. 4. Illustration of sub-tiles used in the dismissed ghost point method. The areas of the shaded polygons are used to determine the affine weights by which to mix the values at the corresponding data sites (drawn solid).
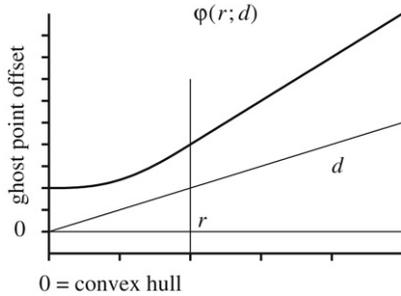

Fig. 5. The graph of $\varphi(r; d)$.

3.3. Ghost point placement

We differentiate between static and dynamic methods. A static method places ghost points at fixed positions and allows a limited expansion of the domain. If the required amount of extrapolation is small, this method delivers satisfying results and is computationally less expensive than a dynamic approach. There are fewer restrictions on the placement of ghost points in the static case, and the dynamic ghost point methods discussed next become static if the distance of the evaluation position to the convex hull is always assumed zero.

A dynamic ghost point method constructs an adequately expanded convex hull for every evaluation point $\mathbf{p} \in \mathbb{R}^n$, with the ghost points' coordinates continuously depending on that of $\mathbf{p}$. Let $d := d(\mathbf{p}, \mathscr{C}(\mathbf{X})) = \min_{\mathbf{x} \in \mathscr{C}(\mathbf{X})} \|\mathbf{x} - \mathbf{p}\|$ be the distance of $\mathbf{p}$ to the convex hull of the data sites. We propose to place the ghost points at a distance twice as far from the convex hull as $\mathbf{p}$ such that it is always in the middle of some enclosing points. It is important that ghost points are distinct, since otherwise the family of interpolants parameterized by d is no longer continuous with respect to the evaluation position. We model the link between d and the coordinates of the ghost points using a monotone, smooth function $\varphi(d)$ such that $\varphi(d) \geq 2d$. In our examples, we choose

the piecewise quartic C^2 function φ that blends between an initial, constant distance r and $2d$ over the interval $[0, r]$ as shown in Fig. 5. The function is given by

$$\varphi(r; d) = \begin{cases} r + 2d^3/r^2 - d^4/r^3 & \text{if } d < r \\ 2d & \text{else.} \end{cases}$$

An extension of φ to higher smoothness is straightforward, and any monotone, smooth function whose slope converges to roughly two is appropriate. The parameter r depends on the individual setting and will be discussed along with the proposed ghost point approaches.

Dynamic Convex Hull Offset (CHO). We generate ghost points $\mathbf{x}_i^G$ by displacing the convex hull vertices $\mathbf{x}_i^{\mathscr{C}}$ such that the new convex hull edges are parallel to the old convex hull edges at a distance of $\varphi(r; d)$, as shown in Fig. 6(a), and call this the convex hull offset (CHO) strategy. To make the ghost point distribution more homogeneous, we insert additional ghost points $\mathbf{x}_i^{Gm}$ on the new convex hull such that they project onto the mid-points $(\mathbf{x}_i^{\mathscr{C}} + \mathbf{x}_{i+1}^{\mathscr{C}})/2$ of the old convex hull edges, where index arithmetic is modulo the number of convex hull vertices. To ensure that the evaluation position $\mathbf{p}$ is contained within the convex hull of the ghost points, $\varphi(r; d) \geq d$ must hold. Since $\varphi(r; d) \geq 2d$, it is furthermore guaranteed that $\mathbf{p}$ lies well away from the boundary of the convex hull. The advantage of this particular construction is the association between ghost points and data sites, which allows a meaningful assignment of values and derivatives to the ghost points. In particular, if t_i is the Taylor polynomial associated with $\mathbf{x}_i^{\mathscr{C}}$, and $\mathbf{a} = \mathbf{x}_i^G - \mathbf{x}_i^{\mathscr{C}}$ is the relative position of $\mathbf{x}_i^G$ with respect to $\mathbf{x}_i^{\mathscr{C}}$, then we assign the Taylor expansion $t_i(\mathbf{a} + \mathbf{x})$ to $\mathbf{x}_i^G$. Similarly, for $\mathbf{a}_1 = \mathbf{x}_i^{Gm} - \mathbf{x}_i^{\mathscr{C}}$ and $\mathbf{a}_2 = \mathbf{x}_i^{Gm} - \mathbf{x}_{i+1}^{\mathscr{C}}$, the Taylor polynomial at $\mathbf{x}_i^{Gm}$ is taken to be the average of the Taylor expansions $t_i(\mathbf{a}_1 + \mathbf{x})$ and $t_{i+1}(\mathbf{a}_2 + \mathbf{x})$. For example, if $t_i(\mathbf{x}) = z_i + \nabla_i \mathbf{x} + \mathbf{x}^T \mathscr{H}_i \mathbf{x}/2$, with z_i, ∇_i, and $\mathscr{H}_i$ denoting value, gradient, and Hessian, then the Taylor polynomial at the ghost point $\mathbf{x}_i^G$ is given by

$$t_i^G(\mathbf{x}) = t_i(\mathbf{a} + \mathbf{x}) = \underbrace{t_i(\mathbf{a})}_{=:z_i^G} + \underbrace{(\nabla_i + \mathscr{H}_i\mathbf{a})}_{=:\nabla_i^G} \cdot \mathbf{x} + \mathbf{x}^T \cdot \underbrace{\mathscr{H}_i}_{=:\mathscr{H}_i^G} \cdot \mathbf{x}/2.$$

The effect of this assignment is shown in Fig. 7 for the extrapolation of linear Taylor polynomials. When applying this approach to higher dimensions, we suggest to use a ghost point for every $1, 2, \ldots, n-1$-simplex on the convex hull, i.e., for vertices, edges, and triangles in 3D.

The two major smooth natural neighbor interpolants available, Farin's C^1 and Hiyoshi's C^2 interpolant, have their reproduction power degraded by one as soon as their evaluation involves ghost points, as we explain in the following. Both interpolants internally use Bézier simplices to model the interpolation constraints given by the derivative data. Thanks to the concept of degree elevation by which they determine underconstrained control points, Farin's interpolant has second order precision for first degree Taylor
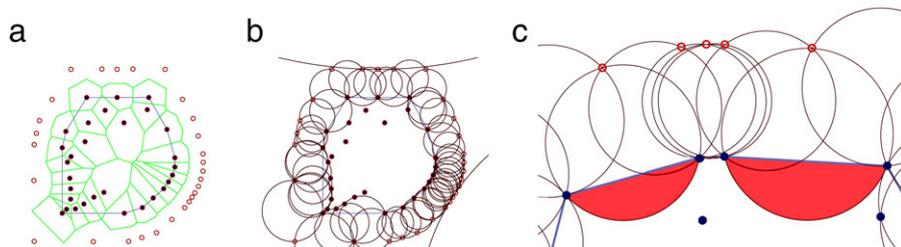

Fig. 6. Ghost points (hollow) in the dynamic convex hull offset approach. (a) The initial setting for $d = 0$. (b) The corresponding circumcircles, providing the blending regions. (c) Convex hull displayed as solid lines with interior below. Only circumcircles involving ghost points are displayed. The shaded blending regions indicate where the original natural neighbor interpolant is augmented to alleviate the convex hull artifacts, the region being much less pronounced at the short edge in the middle.

Fig. 7. (a) The virtual tile evolution for evaluation positions (drawn as thick rings) moving away from the data sites, with ghost points (small rings) moving away as well. (b) Perspective 2.5D view of the setting from (a), the gradients shown as tilted rectangles placed at the data sites.

Fig. 8. (a) Development of the Voronoi diagram of $\mathbf{X}$ (solid dots) and $\mathbf{X}^G$ (thin, hollow dots) for a sequence of query positions (thick, hollow dots) ranging from inside to outside the convex hull of $\mathbf{X}$. (b) Same visualization for the dense circle approach.

polynomials at the data points, and Hiyoshi's interpolant has third order precision for second degree Taylor polynomials. As described in the paragraph on "Reproduction Power" in Section 3.1, the data

at ghost points are extrapolated from the Taylor polynomials at the convex hull data points used in their construction. Consequently, the data at the ghost points does in general not agree with the