

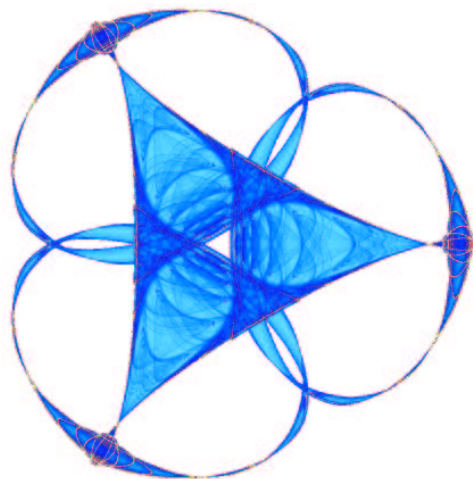
ON MULTIVARIATE INTERPOLATION

By

Peter J. Olver

IMA Preprint Series # 1975

(April 2004)



INSTITUTE FOR MATHEMATICS AND ITS APPLICATIONS

UNIVERSITY OF MINNESOTA
514 Vincent Hall
206 Church Street S.E.
Minneapolis, Minnesota 55455-0436

Phone: 612/624-6066 Fax: 612/626-7370

URL: <http://www.ima.umn.edu>

On Multivariate Interpolation

Peter J. Olver[†]
School of Mathematics
University of Minnesota
Minneapolis, MN 55455
U.S.A.
olver@math.umn.edu
<http://www.math.umn.edu/~olver>

Dedicated to Prof. Israel M. Gel'fand on the occasion of his 90th birthday

Abstract. A new approach to interpolation theory for functions of several variables is proposed. We develop a multivariate divided difference calculus based on the theory of non-commutative quasi-determinants. In addition, intriguing explicit formulae that connect the classical finite difference interpolation coefficients for univariate curves with multivariate interpolation coefficients for higher dimensional submanifolds are established.

1. Introduction.

Interpolation theory for functions of a single variable has a long and distinguished history, dating back to Newton's fundamental interpolation formula and the classical calculus of finite differences, [7, 45, 54, 60]. Standard numerical approximations to derivatives and many numerical integration methods for differential equations are based on the finite difference calculus. Remarkably, despite its evident importance, multivariate interpolation

[†] *Supported in part by NSF Grant DMS 01-03944.*

April 15, 2004

theory remains in a fairly primitive state. Most standard texts only consider the case of rectangular, or, slightly more generally, separable grids, over which the formulae are a simple adaptation of the univariate divided difference calculus. This is even more striking in view of the importance of multivariate interpolation in a wide range of subjects, including computer graphics, geometric design, numerical solutions of partial differential equations on non-uniform grids, and the approximation of geometric invariants such as curvatures and Laplace–Beltrami operators. For instance, in [29, 34, 52], discrete approximations to the principal curvatures and the Laplacian on triangulated surfaces are constructed in the absence of a general theory.

Recent years have witnessed a renewed level of interest in multivariate interpolation among both the pure and applied the research communities. De Boor and Ron, [8, 12, 13], and Sauer and Xu, [57, 10], have systematically studied the polynomial case. Multivariate generalizations of divided differences have been proposed by several authors [9, 28, 39, 47, 55, 56]. Other significant developments include Kergin’s interpolation theory, [37, 43, 44], and the method of radial basis functions, [4]. See [2, 23, 30, 40, 46, 62] for recent algebraic and group-theoretic approaches to polynomial interpolation theory.

The goal of this paper is to investigate some of the basic computational issues that arise in multivariate interpolation. In the polynomial case, it is an elementary observation that the interpolation conditions require the solution of a Vandermonde linear system, [25, 33], which is then effected by the classical divided difference calculus. Less commonly known is that there is a beautiful, explicit LU factorization of the classical Vandermonde matrix. Here is the third order case[†]:

$$S^{(3)} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ x_0 & x_1 & x_2 & x_3 \\ x_0^2 & x_1^2 & x_2^2 & x_3^2 \\ x_0^3 & x_1^3 & x_2^3 & x_3^3 \end{pmatrix} = L^{(3)} U^{(3)}, \quad (1.1)$$

where

$$L^{(3)} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ x_0 & 1 & 0 & 0 \\ x_0^2 & x_0 + x_1 & 1 & 0 \\ x_0^3 & x_0^2 + x_0x_1 + x_1^2 & x_0 + x_1 + x_2 & 1 \end{pmatrix}, \quad (1.2)$$

$$U^{(3)} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & x_1 - x_0 & x_2 - x_0 & x_3 - x_0 \\ 0 & 0 & (x_2 - x_0)(x_2 - x_1) & (x_3 - x_0)(x_3 - x_1) \\ 0 & 0 & 0 & (x_3 - x_0)(x_3 - x_1)(x_3 - x_2) \end{pmatrix}.$$

The general pattern, cf. (2.22), is almost evident from the 3×3 case. In particular, the factorization yields a completely elementary proof of the classical Vandermonde determinant formula as the product of the pivots, i.e., the diagonal entries of the upper triangular

[†] I use S to indicate that we are dealing with the “sample matrix” — here of the monomials $1, x, x^2, x^3$ — and reserve the symbol V for vector spaces.

factor. The entries of $U^{(3)}$ are the classical Newton difference monomials. Indeed, as we will see, the classical interpolation and the divided difference formulae all follow immediately from the factorization. The entries of $L^{(3)}$ are the classical complete symmetric polynomials, [41]. Interestingly, the inverse of $L^{(3)}$ also has a nice explicit formula

$$E^{(3)} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -x_0 & 1 & 0 & 0 \\ x_0x_1 & -x_0 - x_1 & 1 & 0 \\ -x_0x_1x_2 & x_0x_1 + x_0x_2 + x_1x_2 & -x_0 - x_1 - x_2 & 1 \end{pmatrix}, \quad (1.3)$$

whose entries are the classical elementary symmetric polynomials.

Historical note: I noticed the factorization a couple of years ago, and am certain that it must be well known. However, tracking down references has been a challenge. In lectures given around 1991, George Andrews, [1], used it to motivate a general combinatorial method, but didn't publish it. With the help of Gil Strang and Dennis Stanton, I found explicit versions in several rather recent publications, [24, 38, 53, 61], but, so far, nothing older. De Boor, [11], emphasizes the importance of the LU Vandermonde factorization for interpolation theory without explicitly writing it. Interestingly, the rather extensive numerical literature on practical solutions to Vandermonde systems, including [3, 25, 32, 33, 48, 59], fails to reveal the explicit factorization formula.

In the multivariate version, the Vandermonde matrix assumes a block form, where the ordinary powers are replaced by symmetric powers of vectors. To effect a similar explicit block LU decomposition, and thus establish the multivariate analogs of the Newton interpolation formulae, we are naturally led to the non-commutative *quasi-determinants* originally proposed by Heyting, [31], and then developed in detail by Gel'fand, Retakh, and their collaborators, [15, 16, 18–22]. The simplest example of a quasi-determinant is the well-known *Schur complement* $\Theta_W(A) = d - ca^{-1}b$ of a 2×2 block matrix $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$, which appears in its block LU factorization

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ ca^{-1} & 1 \end{pmatrix} \begin{pmatrix} a & b \\ 0 & d - ca^{-1}b \end{pmatrix}. \quad (1.4)$$

The general case can be recursively defined based on the general *heredity principle* formulated by Gel'fand and Retakh, [22, 18]: quasi-determinants of quasi-determinants are quasi-determinants. *Note:* in the scalar case, a quasi-determinant does not reduce to an ordinary determinant, but rather a ratio of determinants. The block factorization of the multivariate Vandermonde matrix leads to a full multivariate version of the classical divided difference calculus, including a general recursive algorithm for computing the multivariate interpolation coefficients, as well as connections with the theory of noncommutative symmetric functions, [19]. The result is a substantial generalization of the univariate divided difference calculus to multivariate interpolation that holds much promise for practical applications.

An alternative, intriguing approach is to relate the interpolation problem for surfaces and higher dimensional submanifolds to that of curves. Suppose we use $n + 1$ points to interpolate a submanifold $N \subset \mathbb{R}^m$. As the points coalesce, the relevant coefficients of the

interpolating submanifold converge to the corresponding derivatives of the submanifold at the point of coalescence. This, indeed, is the foundation of numerical differentiation theory. At the same time, we could also regard the points as interpolating a curve, and thus providing approximations to the derivatives of that curve up to order n . This indicates that there is a profound connection between higher dimensional submanifolds and curves that relate their derivatives at a point in such a manner that the divided difference approximations to the curve derivatives will, in appropriate combinations, also serve to approximate surface derivatives. This observation will be developed in depth in the final section, leading to some fascinating new formulae that directly connect multivariate interpolation and approximation of derivatives with the classical univariate divided difference calculus.

These observations strongly suggest that there is a heretofore unsuspected connection between the derivatives, or “jets”, [49], of submanifolds of differing dimensions, and hence a connection between the different jet bundles for submanifolds of a given manifold. Indeed, the anticipated outcome of our studies should be a geometric blending, through a general construction known as multispace [50], of interpolation-based numerical approximation schemes for jets of submanifolds of dissimilar dimensions. Our theory has been designed to establish a rigorous definition of multivariate multispace as the proper foundation of a geometrical theory of numerical algorithms; see also [14, 42]. Applications to the design of group-invariant numerical methods for approximating differential invariants and integrating invariant differential equations via the method of moving frames, [50], and more general methods of geometric integration, [27, 35], are under development.

2. Interpolation as Factorization.

The purpose of this section is to show how the classical divided difference formulas for univariate polynomial interpolation can be recovered from the LU factorization of a Vandermonde matrix. Indeed, we will establish analogous formulas for completely general function interpolation based on the same matrix factorization method, resulting in a general divided difference calculus for univariate interpolation theory.

Suppose we are given $n + 1$ scalar-valued[†] functions $p_0(x), \dots, p_n(x)$ depending on $x \in \mathbb{R}$. Let $u_0, \dots, u_n \in \mathbb{R}$ indicate a set of interpolation data at the prescribed nodes $x_0, \dots, x_n \in \mathbb{R}$. The general interpolation problem seeks a linear combination

$$u = \varphi(x) = c_0 p_0(x) + \dots + c_n p_n(x) \tag{2.1}$$

that passes through the interpolation points:

$$u_j = \varphi(x_j) = \sum_{k=0}^n c_k p_k(x_j), \quad j = 0, \dots, n. \tag{2.2}$$

[†] The theory applies without significant modification to vector-valued functions of a single variable.

We will often view $u_j = f(x_j)$ as the sample values of a function $u = f(x)$. Of course in many applications, there is no underlying function, and one merely replaces every occurrence of $f(x_j)$ by u_j .

We can rewrite the basic interpolation equations in matrix form as follows. Let $X = (x_0, \dots, x_n)$ denote the sample points. Then (2.2) is equivalent to the matrix equation[†]

$$\mathbf{u} = \mathbf{c} S, \tag{2.3}$$

where $\mathbf{u} = (u_0, u_1, \dots, u_n)$, $\mathbf{c} = (c_0, c_1, \dots, c_n)$ are row vectors, while

$$S = p(X) = \begin{pmatrix} p(x_0) & p(x_1) & \dots & p(x_n) \end{pmatrix} \tag{2.4}$$

denotes the $(n+1) \times (n+1)$ *sample matrix* obtained by sampling the interpolating functions $p(x) = (p_0(x), \dots, p_n(x))^T$ at the points. We shall assume that the the interpolation equations (2.2) have a unique solution, which imposes the following key constraint on the points.

Definition 2.1. A sequence of points $X = (x_0, \dots, x_n)$ is said to be *poised* for the interpolation functions p_0, \dots, p_n if and only if the sample matrix is nonsingular: $\det p(X) \neq 0$.

There are two particularly important special cases. The first is classical polynomial interpolation theory, which takes

$$p_0(x) = 1, \quad p_1(x) = x, \quad \dots \quad p_n(x) = x^n.$$

The sample matrix

$$S = p(X) = \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ x_0 & x_1 & x_2 & \dots & x_n \\ x_0^2 & x_1^2 & x_2^2 & \dots & x_n^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_0^n & x_1^n & x_2^n & \dots & x_n^n \end{pmatrix}. \tag{2.5}$$

is a Vandermonde matrix. As is well known, S is nonsingular, with

$$\det S = \prod_{i < j} (x_i - x_j) \neq 0, \tag{2.6}$$

and hence the points are poised if and only if they are distinct.

The second case is trigonometric interpolation, where

$$p_k(x) = e^{ikx}, \quad k = 0, \dots, n.$$

[†] One can, equivalently, work with column vectors by transposing the matrix systems. However, this leads to a number of technical complications that only seem to obscure the key issues. Despite some initial unease, I came to the conclusion that using to a row vector notation for the interpolation coefficients is the more natural approach.

Equally spaced points $x_\nu = 2\nu\pi/(n+1)$ on the interval $[0, 2\pi]$, lead to the discrete Fourier transform of fundamental importance in signal processing, [58]. The entries of the sample matrix

$$S = p(X) = \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \zeta & \zeta^2 & \dots & \zeta^n \\ 1 & \zeta^2 & \zeta^4 & \dots & \zeta^{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \zeta^n & \zeta^{2n} & \dots & \zeta^{n^2} \end{pmatrix}$$

are $(n+1)$ st roots of unity, i.e., powers of $\zeta = e^{2\pi i/(n+1)}$.

The direct method for solving the interpolation equations (2.3) is standard Gaussian elimination, [58]. Let us assume that the coefficient matrix S is *regular*, [51], meaning that it is invertible and, further, that no row interchanges are required to reduce it to upper triangular form. Regularity is equivalent to the factorization[†]

$$S = LU, \tag{2.7}$$

of the sample matrix into a *special* or *unit* (meaning it has all 1's on the diagonal) lower triangular matrix L and an upper triangular matrix U . Assuming invertibility, the regularity condition can always be arranged by an appropriate ordering of the sample points x_0, \dots, x_n . The entries of L and U can be computed using the following recursive scheme, which is just the systematization of the Gaussian elimination algorithm.

Theorem 2.2. *Given functions $p_0(x), \dots, p_n(x)$, recursively define the functions*

$$\omega_0^i(x) = p_i(x), \quad \omega_{k+1}^i(x) = \omega_k^i(x) - \frac{\omega_k^i(x_k)}{\omega_k(x_k)} \omega_k(x), \quad \begin{matrix} k = 0, \dots, n-1, \\ i = k, \dots, n, \end{matrix} \tag{2.8}$$

where we set

$$\omega_k(x) = \omega_k^k(x), \quad \lambda_k^i(x) = \frac{\omega_k^i(x)}{\omega_k(x)}, \quad \begin{matrix} k = 0, \dots, n, \\ i = k, \dots, n. \end{matrix} \tag{2.9}$$

Then the entries of L and U in the factorization (2.7) of the sample matrix are

$$L_j^k = \begin{cases} \lambda_j^k(x_j), & k \geq j, \\ 0, & k < j, \end{cases} \quad U_j^k = \begin{cases} \omega_k(x_j), & k \leq j, \\ 0, & k > j. \end{cases} \tag{2.10}$$

[†] Even though (2.3) is an equation for row vectors, we still use *row* operations on S (not S^T) to reduce it to upper triangular form. Once the LU factorization (2.7) is established, the system can be solved by back and forward substitution.

Thus, the upper triangular factor $U = \omega(X)$ is the sample matrix for the “triangular basis” functions $\omega(x) = (\omega_0(x), \dots, \omega_k(x))^T$. The effect of the iterative algorithm (2.9) is to compute the interpolation coefficients

$$p_i(x) = \sum_{k=0}^i L_k^i \omega_k(x) = \sum_{k=0}^i \lambda_k^i(x_k) \omega_k(x) \quad (2.11)$$

of the original functions $p_i(x)$ relative to the triangular interpolation basis $\omega_k(x)$. Indeed, if we evaluate both sides of (2.11) at the interpolation points, we recover the LU factorization of the sample matrix in the form

$$p_i(x_j) = \sum_{k=0}^i L_k^i \omega_k(x_j) = \sum_{k=0}^i L_k^i U_j^k. \quad (2.12)$$

The implementation of the recursion (2.9) requires that

$$\omega_k(x_k) \neq 0, \quad k = 0, \dots, n, \quad (2.13)$$

which is just a restatement of the regularity condition. A simple induction proves that $\omega_k^i(x_j) = 0$ for $j < k$. Indeed, the functions $\omega_k^i(x)$ can be completely characterized by an interpolation condition, namely

$$\omega_k^i(x) = p_i(x) - \sigma_k^i(x), \quad \text{where} \quad \sigma_k^i(x) = c_{k,0}^i p_0(x) + \dots + c_{k,k-1}^i p_{k-1}(x)$$

is the k point interpolating function for $p_i(x)$, i.e., $\sigma_k^i(x_j) = p_i(x_j)$, $j = 0, \dots, k-1$.

The direct recursive algorithm for the functions

$$\lambda_{k+1}^i(x) = \frac{\lambda_k^i(x) - \lambda_k^i(x_k)}{\lambda_k(x) - \lambda_k(x_k)}, \quad \lambda_0^i(x) = \frac{p_i(x)}{p_0(x)} \quad \text{where} \quad \lambda_k(x) = \lambda_k^{k+1}(x), \quad (2.14)$$

has the flavor of a divided difference. However, the recursion satisfied by the functions $\omega_k^i(x)$ is *more* fundamental, since it is the one that has a straightforward generalization in the multivariate situation.

Given a function $u = f(x)$, let us write the corresponding interpolating function, satisfying $u_j = f(x_j) = \varphi(x_j)$ for $j = 0, \dots, n$, in terms of the triangular basis functions:

$$\varphi(x) = \sum_{k=0}^n a_k \omega_k(x). \quad (2.15)$$

Its coefficients a_k are computed by an identical recursive scheme. We set $h_0(x) = f(x)$, and then recursively define

$$h_{k+1}(x) = h_k(x) - a_k \omega_k(x), \quad a_k = \frac{h_k(x_k)}{\omega_k(x_k)}, \quad k = 0, \dots, n. \quad (2.16)$$

We note that the scheme can also be implemented by setting

$$g_k(x) = \frac{h_k(x)}{\omega_k(x)}, \quad \text{whence} \quad a_k = g_k(x_k). \quad (2.17)$$

The latter functions can be found directly by a generalized divided difference formula

$$g_0(x) = \frac{f(x)}{p_0(x)}, \quad g_{k+1}(x) = \frac{g_k(x) - g_k(x_k)}{\lambda_k(x) - \lambda_k(x_k)}. \quad (2.18)$$

As we shall shortly see, for polynomial interpolation, the recursion scheme (2.18) is the classical divided difference calculus.

A useful reformulation of the recurrence relations (2.9), (2.14) is based on the generating functions depending on a dummy variable t :

$$\begin{aligned} P(x, t) &= \sum_i p_i(x) t^i = p_0(x) + p_1(x) t + \dots, \\ \Omega_k(x, t) &= \sum_{i \geq k} \omega_k^i(x) t^i = \omega_k(x) t^k + \omega_k^{k+1}(x) t^{k+1} + \dots \\ \Lambda_k(x, t) &= \frac{\Omega_k(x, t)}{\omega_k(x)} = \sum_{i \geq k} \lambda_k^i(x) t^i = t^k + \lambda_k(x) t^{k+1} + \lambda_k^{k+2}(x) t^{k+2} + \dots. \end{aligned}$$

Then equations (2.9), (2.14) can be recast in the form

$$\begin{aligned} \Omega_0(x, t) &= P(x, t), & \Omega_{k+1}(x, t) &= \Omega_k(x, t) - \Omega_k(x_k, t) \frac{\omega_k(x)}{\omega_k(x_k)}, \\ \Lambda_0(x, t) &= \frac{P(x, t)}{p_0(x)}, & \Lambda_{k+1}(x, t) &= \frac{\Lambda_k(x, t) - \Lambda_k(x_k, t)}{\lambda_k(x) - \lambda_k(x_k)}. \end{aligned} \quad (2.19)$$

Note that the denominator in the recurrence formula for $\Lambda_{k+1}(x, t)$ is just the leading term in the series expansion of the numerator.

In the polynomial case, the LU factorization of the Vandermonde matrix (2.5) is, interestingly, related to the classical symmetric polynomials, [41]. Indeed, the generating function[†] for the ordinary monomials $p_i(x) = x^i$ is

$$P(x, t) = \sum_{i=0}^{\infty} x^i t^i = \frac{1}{1 - xt}.$$

where we identify polynomials that agree up to terms of order t^m . By induction, we easily prove

$$\Omega_k(x, t) = \frac{t^k (x - x_0)(x - x_1) \cdots (x - x_{k-1})}{(1 - xt)(1 - x_{k-1}t) \cdots (1 - x_0t)},$$

and hence the triangular interpolation basis functions

$$\omega_k(x) = (x - x_0)(x - x_1) \cdots (x - x_{k-1}),$$

[†] If we restrict our attention to polynomials of degree $\leq n$, then all generating function series should be truncated at order n .

are the usual *difference polynomials*. In this case, (2.15) reduces to the classical Newton form for the interpolating polynomial

$$\begin{aligned}\varphi(x) &= a_0 \omega_0(x) + a_1 \omega_1(x) + \cdots + a_n \omega_n(x) \\ &= a_0 + a_1(x - x_0) + \cdots + a_n(x - x_0)(x - x_1) \cdots (x - x_{n-1}).\end{aligned}\tag{2.20}$$

On the other hand,

$$\Lambda_k(x, t) = \frac{t^k}{(1 - xt)(1 - x_{k-1}t) \cdots (1 - x_0t)} = \sum_{i=k}^{\infty} h_{k-i}(x, x_0, \dots, x_{k-1}) t^i,$$

where

$$h_k(x_1, \dots, x_n) = \sum_{1 \leq i_1 \leq i_2 \leq \cdots \leq i_k \leq n} x_{i_1} x_{i_2} \cdots x_{i_k}, \quad k > 0,\tag{2.21}$$

is the *complete symmetric polynomial* of degree k in n variables. Thus, the $S = LU$ factorization of the Vandermonde matrix (2.5) is given by

$$L_j^k = \begin{cases} h_{k-j}(x_0, \dots, x_j), & k \geq j, \\ 0, & k < j, \end{cases} \quad U_j^k = \omega_k(x_j),\tag{2.22}$$

where the lower triangular entries are the complete symmetric polynomials, while the upper triangular entries are the difference polynomials! Incidentally, the factorization provides a simple, direct proof of the classical Vandermonde determinant formula (2.6), since the determinant of $S = LU$ is the product of the pivots — the diagonal entries of U .

Another interesting observation is that the inverse of the complete symmetric polynomial matrix L is the *elementary symmetric polynomial* matrix $L^{-1} = E$, where

$$E_j^i = \begin{cases} (-1)^{i-j} e_{i-j}(x_0, \dots, x_{i-1}), & i \geq j, \\ 0, & i < j, \end{cases}$$

where $e_k(x_0, \dots, x_m)$ are the standard elementary symmetric polynomials, [41]. Indeed, the generating functions for the entries of L^{-1} are

$$\frac{1}{\Lambda_k(x, t)} = \frac{(1 - xt)(1 - x_{k-1}t) \cdots (1 - x_0t)}{t^k} = \sum_{m=0}^k (-1)^{k-m} e_{k-m}(x, x_0, \dots, x_{k-1}) t^{-m}.$$

In particular,

$$\lambda_k(x) = h_1(x, x_0, \dots, x_{k-1}) = x + x_0 + \cdots + x_{k-1},$$

and so the recursive rule (2.18) for the interpolation coefficients a_k reduces to the classical divided difference formula

$$g_{k+1}(x) = \frac{g_k(x) - g_k(x_k)}{x - x_k}, \quad g_0(x) = f(x).\tag{2.23}$$

Therefore, the interpolation coefficients

$$a_k = g_k(x_k) = [u_0 u_1 u_2 \cdots u_k], \quad k = 0, \dots, n,\tag{2.24}$$

agree with the usual divided differences of the function values $u_j = f(x_j)$ at the sample points, [45, 54].

Let us write

$$\delta^n f = \delta^n \varphi = (a_0, \dots, a_n) = (g_0(x_0), g_1(x_1), \dots, g_n(x_n)) \quad (2.25)$$

for the row vector of divided differences (2.24) of order $\leq n$ for the function $f(x)$ on the specified points — which are the same as the divided differences of its interpolant $\varphi(x)$. We can then write the Newton interpolating polynomial (2.20) in the compact form

$$\varphi(x) = \delta^n f \cdot \omega(x) \quad \text{so that} \quad f(X) = \varphi(X) = \delta^n f \cdot \omega(X) \quad (2.26)$$

represent the defining interpolation conditions relative to $\omega(x) = (\omega_0(x), \dots, \omega_n(x))^T$. In particular, if we let $p(x) = (p_0(x), \dots, p_n(x))^T = (1, x, \dots, x^n)^T$ be the original monomials, then

$$p(x) = \delta^n p \cdot \omega(x) \quad \text{where} \quad S = p(X) = \delta^n p \cdot \omega(X) \quad (2.27)$$

reproduces the $S = LU$ factorization of the Vandermonde sample matrix.

To approximate the derivatives of a function, we consider the coalescent limit of the divided difference formulas. We restrict our attention to the polynomial case, although similar reasoning can be used for the analysis of general function interpolation. If $f(x_0, \dots, x_n, x)$ is any function depending on the sample points and, possibly, a variable point x , we define

$$\text{c-lim } f = \lim_{x_0, \dots, x_n \rightarrow x_*} f(x_0, \dots, x_n, x) \quad (2.28)$$

to be its *coalescent limit* as the sample points (but not necessarily x) all converge to the same point x_* .

Given a vector-valued function $f(x) = (f_1(x), \dots, f_k(x))^T$ of a scalar variable x , let

$$d^n f(x) = \begin{pmatrix} f_1(x) & f_1'(x) & \dots & f_1^{(n)}(x) \\ \vdots & \vdots & \ddots & \vdots \\ f_k(x) & f_k'(x) & \dots & f_k^{(n)}(x) \end{pmatrix} \quad (2.29)$$

denote the $k \times (n+1)$ *Wronskian matrix* of order n . In particular, taking the Wronskian of the interpolating polynomial, written in the compact form (2.26), yields

$$d^n \varphi = \delta^n f \cdot d^n \omega. \quad (2.30)$$

where $d^n \omega$ denote the Wronskian of the difference polynomials. In the coalescent limit, the latter Wronskian is readily seen to converge to the diagonal factorial matrix,

$$\text{c-lim } d^n \omega = D = \text{diag}(0!, 1!, 2!, \dots, n!), \quad (2.31)$$

while, assuming $f \in C^{n+1}$, the derivatives of f and its interpolating polynomial φ coincide in the coalescent limit, [7, 54]. Thus, in the limit, (2.30) becomes

$$d^n f = \text{c-lim } \delta^n f \cdot D, \quad (2.32)$$

which is just a matrix form of the classical result that, for a sufficiently smooth function, its k^{th} order divided difference converges to its k^{th} order derivative divided by $k!$.

3. Quasi-Determinants.

In preparation for our development of the corresponding multivariate interpolation formulae, we review the calculus of quasi-determinants first introduced by Heyting, [31], and then developed in detail by Gel'fand and Retakh and collaborators, [15, 16, 18–22].

As before, we will call a nonsingular square matrix A *regular* if it can be reduced to upper triangular form by a sequence of elementary row operations that add multiples of one row to another; row interchanges are not allowed. Regularity is equivalent to the existence of a (unique) factorization

$$A = LU = LD\widehat{U} \tag{3.1}$$

into a product of a special lower triangular and an invertible upper triangular matrix, or, in more detailed form, into a product of a special lower triangular matrix L , an invertible diagonal matrix D , and a special upper triangular matrix \widehat{U} , [58]. The *pivots* of A appear along the diagonal of both U and D , and are ratios of determinants of successive minors of A . The product of the pivots equals the determinant of A .

In the standard theory, the entries of A are real or complex numbers, but the computations apply to any matrix whose entries belong to a commutative field. Multivariate interpolation theory requires extending the factorizations (3.1) to matrices in block form, *whose entries no longer necessarily commute*. More specifically, consider a linear transformation $A: V \rightarrow V$ on a finite-dimensional vector space V . Introducing a basis $\mathbf{e}_1, \dots, \mathbf{e}_N$ of V results in the identification of A with an $N \times N$ matrix with scalar entries a_j^i . More generally, a direct sum decomposition of

$$V = V_1 \oplus \dots \oplus V_n \tag{3.2}$$

into subspaces induces a block decomposition of $A = (A_j^i)$, in which each component

$$A_j^i: V_j \longrightarrow V_i \tag{3.3}$$

denotes a linear transformation on the summands. The subspaces can be of varying dimensions, and so the off-diagonal blocks need not be square.

The primary goal of this section is to establish a corresponding block form of the LU factorization (3.1). The explicit formula for the entries of the factors can be written in terms of quasi-determinants. In the Gel'fand–Retakh formulation, [22], the entries A_j^i of the matrix all belong to the same non-commutative or “skew” field, and hence can be multiplied and inverted (when possible) arbitrarily. Our approach is more general, in that the blocks A_j^i are allowed to have different sizes, and so multiplication of blocks is only allowed if they are “compatible”. In general, the (non-commutative) multiplication $A_j^i A_l^k: V_l \rightarrow V_i$ makes sense if and only if $j = k$. Moreover, since the subspaces V_i are not necessarily of the same dimension, we are only permitted to invert diagonal entries, e.g., A_i^i or suitable products, e.g., $A_j^i A_i^j$, that map V_i to itself. (In subsequent formulas, we implicitly assume that the matrix satisfies the requisite conditions that allow invertibility of any indicated factor.) Despite the different contexts, the two theories turn out to be almost identical.

Let us begin with the simplest case in which there is a direct sum decomposition $V = W \oplus Z$ into two subspaces W, Z . We shall decompose a linear map $A: V \rightarrow V$ into the block form

$$A = \begin{pmatrix} A_1^1 & A_2^1 \\ A_1^2 & A_2^2 \end{pmatrix}, \quad (3.4)$$

where, for instance $A_1^1: W \rightarrow W$, $A_2^1: Z \rightarrow W$, etc.

Definition 3.1. Given $A \in \mathcal{L}(V)$, and a direct sum decomposition $V = W \oplus Z$, we define the W -quasi-determinant of A to be

$$\Theta_W(A) = A_2^2 - A_1^2(A_1^1)^{-1}A_2^1 : Z \longrightarrow Z. \quad (3.5)$$

Remark: We always assume that any indicated square matrix that is to be inverted is invertible. The collection of all such invertibility assumptions will serve to define regularity of our matrices.

The formulas will look simpler if we abbreviate (3.4) as

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \quad \text{whereby} \quad \Theta_W(A) = d - ca^{-1}b \quad (3.6)$$

is the *Schur complement* of the block matrix A . The resulting block LU factorization

$$A = LU = \begin{pmatrix} 1 & 0 \\ ca^{-1} & 1 \end{pmatrix} \begin{pmatrix} a & b \\ 0 & d - ca^{-1}b \end{pmatrix}$$

permits us to write down a formula for the inverse of A ,

$$\begin{aligned} A^{-1} = U^{-1}L^{-1} &= \begin{pmatrix} a^{-1} & -a^{-1}b(d - ca^{-1}b)^{-1} \\ 0 & (d - ca^{-1}b)^{-1} \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -ca^{-1} & 1 \end{pmatrix} \\ &= \begin{pmatrix} a^{-1} + a^{-1}b(d - ca^{-1}b)^{-1}ca^{-1} & -a^{-1}b(d - ca^{-1}b)^{-1} \\ -(d - ca^{-1}b)^{-1}ca^{-1} & (d - ca^{-1}b)^{-1} \end{pmatrix}. \end{aligned} \quad (3.7)$$

We note that we could equally well reduce with respect to the subspace W instead of Z , leading to the Z -quasi-determinant

$$\Theta_Z(A) = a - bd^{-1}c,$$

that appears in the permuted LU factorization

$$A = \begin{pmatrix} 1 & bd^{-1} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} a - bd^{-1}c & 0 \\ c & d \end{pmatrix} = \tilde{L}\tilde{U}, \quad (3.8)$$

whence

$$A^{-1} = \begin{pmatrix} (a - bd^{-1}c)^{-1} & -(a - bd^{-1}c)^{-1}bd^{-1} \\ -d^{-1}c(a - bd^{-1}c)^{-1} & d^{-1} + d^{-1}c(a - bd^{-1}c)^{-1}bd^{-1} \end{pmatrix}. \quad (3.9)$$

The reader may enjoy verifying that the two formulae (3.7), (3.9) are, in fact, equal! This fact is known as the Matrix Inversion Lemma in linear estimation theory, [36], and the Sherman–Morrison–Woodbury formula in linear algebra, [5, 26]. In particular, this implies that if a and d are invertible, then $\Theta_Z(A) = d - ca^{-1}b$ is invertible if and only if $\Theta_W(A) = a - bd^{-1}c$ is invertible.

Remark: In (3.8), the matrix \tilde{L} is “lower triangular” and the matrix \tilde{U} is “upper triangular” with respect to the alternative basis ordering $\mathbf{e}_2, \mathbf{e}_1$. They happen to look upper and lower triangular with respect to the standard ordering $\mathbf{e}_1, \mathbf{e}_2$, but this is a low-dimensional accident.

The key result is that quasi-determinantal reduction is a commutative operation. This is the *heredity principle* formulated by Gel’fand and Retakh, [18, 22]: quasi-determinants of quasi-determinants are quasi-determinants. In the present set-up, we decompose the subspace $Z = X \oplus Y$ and then perform two successive quasi-determinantal reductions.

Theorem 3.2. *Suppose $V = W \oplus X \oplus Y$. Then*

$$\Theta_X(\Theta_W(A)) = \Theta_{W \oplus X}(A) = \Theta_W(\Theta_X(A)) : Y \longrightarrow Y. \quad (3.10)$$

Proof: Let us write

$$A = \begin{pmatrix} a & x & y \\ u & p & q \\ v & r & s \end{pmatrix}$$

in block form relative to the decomposition $V = W \oplus X \oplus Y$. Then, according to the definition (3.5),

$$\Theta_W(A) = \begin{pmatrix} p & q \\ r & s \end{pmatrix} - \begin{pmatrix} u \\ v \end{pmatrix} a^{-1} (x \ y) = \begin{pmatrix} p - ua^{-1}x & q - ua^{-1}y \\ r - va^{-1}x & s - va^{-1}y \end{pmatrix}.$$

Applying the X quasi-determinantal reduction to this matrix, we obtain

$$\Theta_X(\Theta_W(A)) = (s - va^{-1}y) - (r - va^{-1}x)(p - ua^{-1}x)^{-1}(q - ua^{-1}y). \quad (3.11)$$

On the other hand, using (3.5) on the decomposition $V = (W \oplus X) \oplus Y$ yields

$$\begin{aligned} \Theta_{W \oplus X}(A) &= s - (v \ r) \begin{pmatrix} a & x \\ u & p \end{pmatrix}^{-1} \begin{pmatrix} y \\ q \end{pmatrix} \\ &= s - va^{-1}y - va^{-1}x(p - ua^{-1}x)^{-1}ua^{-1}y + \\ &\quad + va^{-1}x(p - ua^{-1}x)^{-1}q + r(p - ua^{-1}x)^{-1}ua^{-1}y - r(p - ua^{-1}x)^{-1}q, \end{aligned} \quad (3.12)$$

where we used (3.7) to compute the inverse of $\begin{pmatrix} a & x \\ u & p \end{pmatrix}$. The expressions (3.11), (3.12) are clearly equal, proving the first equality in (3.10). The second, reverse equality, follows from the trivial remark that the direct sum decomposition is independent of order: $W \oplus X = X \oplus W$. (Alternatively, the reader can verify this by direct computation.) *Q.E.D.*

The heredity principle allows us to recursively compute quasi-determinants with respect to general decompositions $V = V_1 \oplus \cdots \oplus V_n$ of our vector space. Given an unordered multi-index $I = \{i_1, \dots, i_m\} \subset \{1, \dots, n\}$, we let $V_I = V_{i_1} \oplus \cdots \oplus V_{i_m}$. Note that

$$V = V_I \oplus V_{\sim I} \quad (3.13)$$

where $\sim I = \{1, \dots, n\} \setminus I$ is the complementary multi-index. In particular, set

$$\widehat{V}_k = V_{1\dots k-1, k+1\dots n} = V_1 \oplus \dots \oplus V_{k-1} \oplus V_{k+1} \oplus \dots \oplus V_n.$$

As above, let $A_j^i: V_j \rightarrow V_i$ be the restriction of A to the indicated subspaces. More generally, given unordered multi-indices $I = \{i_1, \dots, i_m\}$, $J = \{j_1, \dots, j_m\}$ having the same cardinality, we define $A_J^I: V_J \rightarrow V_I$, so that A_J^I is an $m \times m$ block matrix formed out of the individual blocks $A_{j_\nu}^{i_\kappa}$ of A .

Definition 3.3. Define the $I = \{i_1, \dots, i_m\}$ quasi-determinant of A to be

$$\Theta_I(A) = \Theta_{\{i_1, \dots, i_m\}}(A) = \Theta_{V_I}(A) : V_{\sim I} \longrightarrow V_{\sim I},$$

relative to the decomposition (3.13).

The commutativity Theorem 3.2 implies that we can compute

$$\Theta_I(A) = \Theta_{i_1}(\Theta_{i_2}(\dots \Theta_{i_m}(A) \dots)) \quad \text{when} \quad I = \{i_1, \dots, i_m\}$$

recursively, and in any order. The k^{th} quasi-determinant of A is defined to be

$$\Theta^{(k)}(A) = \Theta_{1, \dots, k-1, k+1, \dots, n}(A) : V_k \longrightarrow V_k. \quad (3.14)$$

In particular, the *last quasi-determinant* with respect to the given ordering of the subspaces is denoted by

$$\Theta_*(A) = \Theta^{(n)}(A) : V_n \longrightarrow V_n. \quad (3.15)$$

Warning: In the commutative (scalar) case, the last quasi-determinant of a matrix does *not* reduce to its usual determinant, but rather the ratio of its determinant to the determinant of its upper left $(n-1) \times (n-1)$ minor — which is the same as its final pivot under regular (no row interchanges) Gaussian elimination, [51, 58].

Remark: The last quasi-determinant $\Theta_*(A)$ does not require that the lower right block A_n^n be invertible, and hence makes sense even for certain types of rectangular matrices. The case that will appear shortly is for the submatrix $A_{12\dots k-1, j}^{12\dots k-1, i}$ with $i \neq j$.

Remark: If every A_j^i is invertible, then our quasi-determinants coincide with those of Gel'fand and ReytaKh, [18–22]. For example, our 2×2 quasi-determinant (3.6) is identical to theirs, whereas they write a 3×3 quasi-determinant in the form

$$\Theta_* \begin{pmatrix} a & x & y \\ u & p & q \\ v & r & s \end{pmatrix} = s - r(p - ua^{-1}x)^{-1}q - r(x - au^{-1}p)^{-1}y - \\ - v(r - px^{-1}a)^{-1}q - v(a - xp^{-1}u)^{-1}y.$$

The proof that this expression actually equals (3.11), when the appropriate invertibility assumptions are made, is a nice challenge!

Quasi-determinants allow us to easily formulate the general triangular decomposition of a block, non-commutative matrix.

Theorem 3.4. A regular block matrix $A = (A_j^i)$ factors as

$$A = K D^{-1} U, \quad (3.16)$$

where

$$K = \begin{pmatrix} A_1^1 & & & & \\ A_1^2 & \Theta_*(A_{12}^{12}) & & & \\ A_1^3 & \Theta_*(A_{12}^{13}) & \Theta_*(A_{123}^{123}) & & \\ \vdots & \vdots & \vdots & \ddots & \\ A_1^n & \Theta_*(A_{12}^{1n}) & \Theta_*(A_{123}^{12n}) & \dots & \Theta_*(A) \end{pmatrix}, \quad K_j^i = \Theta_*(A_{12\dots j-1,j}^{12\dots j-1,i}), \quad i \geq j,$$

$$D = \begin{pmatrix} A_1^1 & & & & \\ & \Theta_*(A_{12}^{12}) & & & \\ & & \Theta_*(A_{123}^{123}) & & \\ & & & \ddots & \\ & & & & \Theta_*(A) \end{pmatrix}, \quad D_i^i = \Theta_*(A_{12\dots i-1,i}^{12\dots i-1,i}),$$

$$U = \begin{pmatrix} A_1^1 & & & & \\ & A_2^1 & & & \\ & \Theta_*(A_{12}^{12}) & A_3^1 & \dots & A_n^1 \\ & & \Theta_*(A_{13}^{12}) & \dots & \Theta_*(A_{1n}^{12}) \\ & & \Theta_*(A_{123}^{123}) & \dots & \Theta_*(A_{12n}^{123}) \\ & & & \ddots & \vdots \\ & & & & \Theta_*(A) \end{pmatrix}, \quad U_j^i = \Theta_*(A_{12\dots i-1,j}^{12\dots i-1,i}), \quad i \leq j.$$

Regularity of A requires invertibility of each diagonal block in D , namely

$$\Theta_*(A_{12\dots k}^{12\dots k}) \quad \text{is invertible for} \quad k = 1, \dots, n. \quad (3.17)$$

Proof: The 2×2 case of (3.16) takes the form

$$A = \begin{pmatrix} a & 0 \\ c & d - ca^{-1}b \end{pmatrix} \begin{pmatrix} a^{-1} & 0 \\ 0 & (d - ca^{-1}b)^{-1} \end{pmatrix} \begin{pmatrix} a & b \\ 0 & d - ca^{-1}b \end{pmatrix}. \quad (3.18)$$

To proceed further, we break the lower right block up into subblocks and use an induction based on the heredity principle (3.10). *Q.E.D.*

The preceding decomposition (3.16) is easily converted to the alternative factorizations

$$A = LU = LDV, \quad \text{where} \quad L = KD^{-1}, \quad V = D^{-1}U,$$

with L, V special — meaning identity matrices appear along their diagonal, and

$$\begin{aligned} L_j^i &= \Theta_*(A_{12\dots j-1,j}^{12\dots j-1,i}) \Theta_*(A_{12\dots j}^{12\dots j})^{-1}, & i \geq j, \\ U_j^i &= \Theta_*(A_{12\dots i-1,j}^{12\dots i-1,i}), & i \leq j. \end{aligned} \quad (3.19)$$

4. Divided Differences for Functions of Several Variables.

We now adapt the preceding constructions to develop a general interpolation theory for functions of several variables. In the first approach, the theory proceeds exactly as

in the one-dimensional version presented in Section 2, with $x = (x^1, \dots, x^m) \in \mathbb{R}^m$ now referring to points in m -dimensional space. The number of variables dictates the dimension of the interpolating submanifold. Thus, univariate interpolation, $m = 1$, corresponds to passing a curve through a collection of data points, bivariate, $m = 2$, corresponds to interpolation by a surface, and so on.

Suppose we are given scalar-valued functions[†] $P_k: \mathbb{R}^m \rightarrow \mathbb{R}$ for $k = 0, \dots, N$. For the multivariate interpolation problem, we are given points $x_0, \dots, x_N \in \mathbb{R}^m$ and associated data $u_0 = F(x_0), \dots, u_N = F(x_N) \in \mathbb{R}^m$, and seek a function

$$u = \Phi(x) = c_0 P_0(x) + \dots + c_N P_N(x) \quad (4.1)$$

that satisfies the interpolation conditions

$$u_j = F(x_j) = \Phi(x_j), \quad j = 0, \dots, N. \quad (4.2)$$

As before, the interpolation conditions can be written in the equivalent matrix form

$$\mathbf{c} S = \mathbf{u}, \quad (4.3)$$

where $\mathbf{u} = (u_0, u_1, \dots, u_N)$ is the row vector of interpolation values, $\mathbf{c} = (c_0, c_1, \dots, c_N)$ is the row vector of unknown coefficients, while

$$S = P(X) = (P_k(x_j)) \quad (4.4)$$

is the $(N + 1) \times (N + 1)$ sample matrix of the functions $P(x) = (P_0(x), \dots, P_n(x))^T$ at the points $X = (x_0, \dots, x_N)$.

The existence of a unique interpolating function (4.1) requires that the points be *poised*, in the sense of Definition 2.1, meaning that the sample matrix is nonsingular, $\det S \neq 0$, or, equivalently, that no nontrivial linear combination (4.1) vanishes at all the sample points. Unlike the scalar case, which, at least in the most important examples, merely requires that the points be distinct, in the multivariate theory this imposes significant constraints on their geometric configuration.

The most important example is polynomial interpolation, where the interpolation functions are all the monomials of degree $0 \leq \#I \leq n$, of which there are a total of

$$N + 1 = m^{(n)} \equiv 1 + m_1 + \dots + m_n = \binom{m + n}{m}, \quad \text{where } m_i = \binom{m + i - 1}{m - 1}, \quad (4.5)$$

counts the number of monomials of degree i . For later convenience, we rescale the monomials by multinomial coefficients, and set

$$P_I(x) = \binom{\#I}{I} x^I, \quad 0 \leq \#I \leq n.$$

[†] For simplicity we assume the functions are globally defined, although one can easily adapt the general construction to vector-valued functions defined on open subdomains of Euclidean space, or even on general smooth manifolds.

The sample matrix (4.4) is a *multi-dimensional Vandermonde* matrix, which we write in columnar block form

$$S = S^{(n)} = P(X) = \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ x_0 & x_1 & x_2 & \dots & x_N \\ x_0^{\odot 2} & x_1^{\odot 2} & x_2^{\odot 2} & \dots & x_N^{\odot 2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_0^{\odot n} & x_1^{\odot n} & x_2^{\odot n} & \dots & x_N^{\odot n} \end{pmatrix}. \quad (4.6)$$

Here, given a column vector $x = (x^1, \dots, x^p)^T \in \mathbb{R}^m$, we define its i^{th} *symmetric power* to be the column vector[†] $x^{\odot i} \in \mathbb{R}^{m_i}$, with entries $\binom{i}{I} x^I$, $0 \leq \#I = i \leq n$, which are precisely the terms appearing in the multinomial expansion of $(x^1 + \dots + x^m)^i$. Using this notation, we can write a general polynomial of degree n in the abbreviated form

$$P(x) = \sum_{i=0}^n c_i x^{\odot i} = \sum_{i=\#I=0}^n \binom{i}{I} c_I x^I, \quad (4.7)$$

where $c_i = (\dots c_I \dots)$ is the row vector consisting of its degree $i = \#I$ coefficients. The points are *poised* if and only if the Vandermonde matrix is nonsingular, which guarantees a unique interpolating polynomial of degree $\leq n$ passing through the data points.

Theorem 4.1. *The $N + 1 = m^{(n)}$ points $x_0, \dots, x_N \in \mathbb{R}^m$ are polynomially poised if and only if they do not belong to a common algebraic hypersurface of degree $\leq n$ if and only if for any u_0, \dots, u_N , there exists a unique interpolating polynomial $u = \Phi(x)$ of degree $\leq n$ satisfying $u_i = \Phi(x_i)$ for $i = 0, \dots, N$.*

In other words, the points are non-poised for polynomial interpolation if and only if there exists a nonzero scalar-valued polynomial $q(x)$ of degree $\leq n$ such that $q(x_\nu) = 0$ for all $\nu = 0, \dots, N$. Or, to state this another way, the vanishing of the multivariate Vandermonde determinant $\det S^{(n)} = 0$ is necessary and sufficient for a collection of points to lie on a common algebraic hypersurface of degree $\leq n$.

Implementation of the recursive factorization algorithm (2.8), (2.9) is straightforward. However, the scheme does depend upon a choice of ordering of the polynomials. Also, the scheme requires that the points satisfy the additional restriction (2.13) that ensures continuation of the recursive algorithm. Thus, we must require that, for each $k = 1, \dots, N$, there is no linear combination of the first k polynomials that vanishes on the first k sample points. Unlike the poised condition, this depends upon the ordering of the sample points, as well as the monomial ordering.

The resulting triangular basis functions $\omega_k(x)$ will also depend upon the ordering. Unlike the univariate case, they no longer depend in a polynomial manner upon the sample point coordinates, and, while eminently computable, the general formula is complicated.

[†] We can identify \mathbb{R}^{m_i} with the k^{th} symmetric power $\odot^i \mathbb{R}^m$ of our underlying vector space, cf. [17]. However, this level of abstraction is unnecessary for what follows.

(Generating functions do not appear to offer any appreciable help.) For example, under the standard degree lexicographic ordering, $1, x, y, x^2, 2xy, y^2, \dots$, we find

$$\omega_0(x) = 1, \quad \omega_1(x) = x - x_0, \quad \omega_2(x) = \frac{(x_1 - x_0)(y - y_0) - (y_1 - y_0)(x - x_0)}{x_1 - x_0},$$

$$\omega_3(x) = \frac{\left[\begin{array}{l} (x_1 - x_0)(y_2 - y_0)(x - x_0)(x - x_1) - (x_2 - x_0)(y_1 - y_0)(x - x_0)(x - x_2) \\ - (x_2 - x_0)(x_2 - x_1)(x_1 - x_0)(y - y_0) \end{array} \right]}{(x_1 - x_0)(y_2 - y_0) - (y_1 - y_0)(x_2 - x_0)},$$

and so on.

One way to avoid the issue of ordering is to reformulate the interpolation conditions using a block factorization of the multivariate Vandermonde matrix. The idea of treating interpolation using block decomposition of the Vandermonde coefficient matrix is not new, a block factorization of the multivariate Vandermonde matrix, as in [5, 6, 8, 12, 55, 57]; see also [59] for numerical solution methods for block Vandermonde systems.

To avoid the issue of ordering, we reformulate the interpolation conditions using a block factorization of the multivariate Vandermonde matrix, [5, 6, 8, 12, 55, 57, 59]. A key advantage of the divided difference calculus is that one can efficiently pass from the interpolation polynomial of degree $n - 1$ to that of degree n without having to recompute any of the previous coefficients. As we learned in Section 2, this is a direct result of the LU factorization of the Vandermonde matrix. In the multivariate version, we note that there are m_n additional coefficients in an n^{th} degree polynomial that do not appear in that of degree $n - 1$, and hence the passage requires the inclusion of m_n additional interpolation points. To this end, we partition the interpolation points into *blocks*, which are subsets of sizes $m_0 = 1, m_1 = m, \dots, m_n$. Along with the initial point x_0 , the first m_1 points will be used to determine the linear interpolating polynomial, the next m_2 will be added in to determine the interpolating quadratic polynomial, and so on. Let

$$X_k = (x_{m^{(k-1)}}, \dots, x_{m^{(k)}-1}) \quad (4.8)$$

denote the $m \times m_k$ matrix whose columns consist of the k^{th} block of interpolation points. We write the multivariate Vandermonde matrix in block form

$$S = S^{(n)} = \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ X_0 & X_1 & X_2 & \dots & X_n \\ X_0^{\odot 2} & X_1^{\odot 2} & X_2^{\odot 2} & \dots & X_n^{\odot 2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ X_0^{\odot n} & X_1^{\odot n} & X_2^{\odot n} & \dots & X_n^{\odot n} \end{pmatrix}. \quad (4.9)$$

The (i, j) block $X_j^i = X_j^{\odot i}$ has size $m_i \times m_j$ and its columns are the i^{th} symmetric powers of the j^{th} block of interpolation vectors $x_{m^{(j-1)}+1}, \dots, x_{m^{(j)}}$. The first row of S consists of row vectors of the appropriate sizes all of whose entries are equal to 1.

Example 4.2. In the two-dimensional case, $m = 2$, the second order block Vander-

monde matrix takes the form

$$S^{(2)} = \begin{pmatrix} 1 & 1 & 1 \\ X_0 & X_1 & X_2 \\ X_0^{\odot 2} & X_1^{\odot 2} & X_2^{\odot 2} \end{pmatrix} = \left(\begin{array}{c|cc|ccc} 1 & & & & & \\ \hline & 1 & 1 & 1 & & \\ \hline x_0 & x_1 & x_2 & x_3 & x_4 & x_5 \\ y_0 & y_1 & y_2 & y_3 & y_4 & y_5 \\ \hline x_0^2 & x_1^2 & x_2^2 & x_3^2 & x_4^2 & x_5^2 \\ 2x_0y_0 & 2x_1y_1 & 2x_2y_2 & 2x_3y_3 & 2x_4y_4 & 2x_5y_5 \\ y_0^2 & y_1^2 & y_2^2 & y_3^2 & y_4^2 & y_5^2 \end{array} \right),$$

where

$$X_0 = \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}, \quad X_1 = \begin{pmatrix} x_1 & x_2 \\ y_1 & y_2 \end{pmatrix}, \quad X_2 = \begin{pmatrix} x_3 & x_4 & x_5 \\ y_3 & y_4 & y_5 \end{pmatrix},$$

have sizes 2×1 , 2×2 and 2×3 respectively. This reflects the fact that a homogeneous linear polynomial $ax + by$ has $m_1 = 2$ coefficients, while a homogeneous quadratic polynomial $cx^2 + 2dxy + ey^2$ has $m_2 = 3$ coefficients. The 6 points $(x_0, y_0), \dots, (x_5, y_5)$ are polynomially poised if and only if they do not lie on a common conic section, which is equivalent to the condition $\det S^{(2)} \neq 0$.

Now, in analogy with the univariate case, we base our multivariate divided difference calculus on the block LU factorization

$$S^{(n)} = L^{(n)} U^{(n)}, \quad \text{or, if the order } n \text{ is fixed,} \quad S = LU, \quad (4.10)$$

of the multi-dimensional Vandermonde matrix (4.9). The lower and upper triangular matrices L and U are in block form, and are constructed by recurrence formulae similar to the scalar case (2.8), (2.9).

Lemma 4.3. *The Vandermonde matrix (4.6) is block regular if and only if the sample points are poised in block, meaning that each subset $x_0, x_1, \dots, x_{m^{(k)}-1}$ is poised for $k = 1, \dots, n$.*

Note that the poised in block condition depends on the ordering of the points. Moreover, any poised set of sample points is poised in block relative to some ordering. We make the blanket assumption that our data points are ordered so as to be poised in block.

Theorem 4.4. *Recursively define the vector-valued polynomials*

$$\Omega_0^i(x) = x^{\odot i}, \quad \Omega_{k+1}^i(x) = \Omega_k^i(x) - \Omega_k^i(X_k) \Omega_k(X_k)^{-1} \Omega_k(x), \quad \Omega_k(x) = \Omega_k^k(x). \quad (4.11)$$

Then the multi-dimensional Vandermonde matrix (4.9) has block LU factorization (4.10) in which

$$L_j^k = \begin{cases} \Omega_j^k(X_j) \Omega_j(X_j)^{-1}, & k \geq j, \\ 0, & k < j, \end{cases} \quad U_j^k = \begin{cases} \Omega_k(X_j), & k \leq j, \\ 0, & k > j. \end{cases} \quad (4.12)$$

Note that $\Omega_k: \mathbb{R}^m \rightarrow \mathbb{R}^{m_i}$ is a vector-valued polynomial of degree k in the entries of $x \in \mathbb{R}^m$. However, unlike the scalar case, the coefficients of $\Omega_i(x)$ are *not* polynomial in the sample points x_0, \dots, x_N . The notation $\Omega_k(X_j)$ refers to the corresponding sample matrix, (2.4), based on the j^{th} set of sample points in X_j . Theorem 4.4 is established by a straightforward induction, mimicking the proof of its commutative version Theorem 2.2. Our quasi-determinantal formulas can be viewed as alternative forms of Rabut's formulae for multivariate divided differences based on Vandermonde determinant ratios, [55].

Example 4.5. The cubic case $n = 3$ has the following factorization

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ X_0 & X_1 & X_2 & X_3 \\ X_0^{\odot 2} & X_1^{\odot 2} & X_2^{\odot 2} & X_3^{\odot 2} \\ X_0^{\odot 3} & X_1^{\odot 3} & X_2^{\odot 3} & X_3^{\odot 3} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ X_0 & 1 & 0 & 0 \\ X_0^{\odot 2} & L_1^2 & 1 & 0 \\ X_0^{\odot 3} & L_1^3 & L_2^3 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & \Omega_1(X_1) & \Omega_1(X_2) & \Omega_1(X_3) \\ 0 & 0 & \Omega_2(X_2) & \Omega_2(X_3) \\ 0 & 0 & 0 & \Omega_3(X_3) \end{pmatrix},$$

where

$$\begin{aligned} \hat{x} &\equiv \Omega_1(x) = x - x_0, & \Omega_2(x) &= \hat{x}^{\odot 2} - \widehat{X}_1^{\odot 2} \widehat{X}_1^{-1} \hat{x}, \\ \Omega_3(x) &= \hat{x}^{\odot 3} - \widehat{X}_1^{\odot 3} X_1^{-1} \hat{x} - (\widehat{X}_2^{\odot 3} - \widehat{X}_1^{\odot 3} \widehat{X}_1^{-1} \widehat{X}_2) (\widehat{X}_2^{\odot 2} - \widehat{X}_1^{\odot 2} \widehat{X}_1^{-1} \widehat{X}_2)^{-1} (\hat{x}^{\odot 2} - \widehat{X}_1^{\odot 2} \widehat{X}_1^{-1} \hat{x}), \end{aligned}$$

with $\widehat{X}_j = \Omega_1(X_j)$ obtained by subtracting x_0 from each column — which is equivalent to translating the first point to the origin — and

$$\begin{aligned} L_1^2 &= \widehat{X}_1^{\odot 2} \widehat{X}_1^{-1}, & L_1^3 &= \widehat{X}_1^{\odot 3} \widehat{X}_1^{-1}, \\ L_2^3 &= (\widehat{X}_2^{\odot 3} - \widehat{X}_1^{\odot 3} X_1^{-1} \widehat{X}_2) (\widehat{X}_2^{\odot 2} - \widehat{X}_1^{\odot 2} \widehat{X}_1^{-1} \widehat{X}_2)^{-1}. \end{aligned}$$

As a specific example, in the $m = 2$ dimensional situation, we set $\hat{x} = x - x_0, \hat{y} = y - y_0$, and then

$$\Omega_1(x, y) = \begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix}, \quad \Omega_2(x, y) = \begin{pmatrix} \hat{x}^2 - \frac{(\hat{x}_1^2 \hat{y}_2 - \hat{x}_2^2 \hat{y}_1) \hat{x} + (\hat{x}_2 - \hat{x}_1) \hat{x}_1 \hat{x}_2 \hat{y}}{\hat{x}_1 \hat{y}_2 - \hat{x}_2 \hat{y}_1} \\ 2 \hat{x} \hat{y} - \frac{2(\hat{x}_1 - \hat{x}_2) \hat{y}_1 \hat{y}_2 \hat{x} + 2(\hat{y}_2 - \hat{y}_1) \hat{x}_1 \hat{x}_2 \hat{y}}{\hat{x}_1 \hat{y}_2 - \hat{x}_2 \hat{y}_1} \\ \hat{y}^2 - \frac{(\hat{y}_1 - \hat{y}_2) \hat{y}_1 \hat{y}_2 \hat{x} + (\hat{x}_1 \hat{y}_2^2 - \hat{x}_2 \hat{y}_1^2) \hat{y}}{\hat{x}_1 \hat{y}_2 - \hat{x}_2 \hat{y}_1} \end{pmatrix}, \quad (4.13)$$

while

$$\begin{aligned} L_1^2 &= \begin{pmatrix} \hat{x}_1^2 & \hat{x}_2^2 \\ 2 \hat{x}_1 \hat{y}_1 & 2 \hat{x}_2 \hat{y}_2 \\ \hat{y}_1^2 & \hat{y}_2^2 \end{pmatrix} \begin{pmatrix} \hat{x}_1 & \hat{x}_2 \\ \hat{y}_1 & \hat{y}_2 \end{pmatrix}^{-1} \\ &= \frac{1}{\hat{x}_1 \hat{y}_2 - \hat{x}_2 \hat{y}_1} \begin{pmatrix} \hat{x}_1^2 \hat{y}_2 - \hat{x}_2^2 \hat{y}_1 & (\hat{x}_2 - \hat{x}_1) \hat{x}_1 \hat{x}_2 \\ 2(\hat{x}_1 - \hat{x}_2) \hat{y}_1 \hat{y}_2 & 2(\hat{y}_2 - \hat{y}_1) \hat{x}_1 \hat{x}_2 \\ (\hat{y}_1 - \hat{y}_2) \hat{y}_1 \hat{y}_2 & \hat{x}_1 \hat{y}_2^2 - \hat{x}_2 \hat{y}_1^2 \end{pmatrix}. \end{aligned}$$

The third order entries are a bit too complicated to display explicitly.

According to Theorem 3.4, the entries of the block LU factorization can be identified as quasi-determinants.

Lemma 4.6. *The polynomials appearing in the recursive scheme (4.11) are equal to the following quasi-determinants:*

$$\Omega_1^i(x) = x^{\odot i}, \quad \Omega_k^i(x) = \Theta_* \left(\begin{array}{cccccc} 1 & 1 & 1 & \dots & 1 & 1 \\ X_0 & X_1 & X_2 & \dots & X_{k-1} & x \\ X_0^{\odot 2} & X_1^{\odot 2} & X_2^{\odot 2} & \dots & X_{k-1}^{\odot 2} & x^{\odot 2} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \\ X_0^{\odot k-1} & X_1^{\odot k-1} & X_2^{\odot k-1} & \dots & X_{k-1}^{\odot k-1} & x^{\odot k-1} \\ X_0^{\odot i} & X_1^{\odot i} & X_2^{\odot i} & \dots & X_{k-1}^{\odot i} & x^{\odot i} \end{array} \right). \quad (4.14)$$

The final block in each row of the quasi-determinant matrix has a single column, and so the entire matrix has size $m^{(k-1)} \times (m^{(k-1)} + 1)$.

In particular, the *multivariate difference polynomials* are given as quasi-determinants

$$\Omega_k(x) = \Theta_* \left(\begin{array}{cccccc} 1 & 1 & 1 & \dots & 1 & 1 \\ X_0 & X_1 & X_2 & \dots & X_{k-1} & x \\ X_0^{\odot 2} & X_1^{\odot 2} & X_2^{\odot 2} & \dots & X_{k-1}^{\odot 2} & x^{\odot 2} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \\ X_0^{\odot k-1} & X_1^{\odot k-1} & X_2^{\odot k-1} & \dots & X_{k-1}^{\odot k-1} & x^{\odot k-1} \\ X_0^{\odot k} & X_1^{\odot k} & X_2^{\odot k} & \dots & X_{k-1}^{\odot k} & x^{\odot k} \end{array} \right), \quad (4.15)$$

Consequently, the blocks in the factorization have the quasi-determinant formulae

$$U_j^k = \Theta_*(X_{01\dots k-1,j}^{01\dots k-1,k}), \quad L_j^k = \Theta_*(X_{01\dots j-1,j}^{01\dots j-1,k}) \Theta_*(X_{12\dots j-1,j}^{01\dots j-1,j})^{-1},$$

where, for $j \geq k$, we abbreviate

$$X_{01\dots k-1,j}^{01\dots k-1,k} = \left(\begin{array}{cccccc} 1 & 1 & 1 & \dots & 1 & 1 \\ X_0 & X_1 & X_2 & \dots & X_{k-1} & X_j \\ X_0^{\odot 2} & X_1^{\odot 2} & X_2^{\odot 2} & \dots & X_{k-1}^{\odot 2} & X_j^{\odot 2} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \\ X_0^{\odot k-1} & X_1^{\odot k-1} & X_2^{\odot k-1} & \dots & X_{k-1}^{\odot k-1} & X_j^{\odot k-1} \\ X_0^{\odot k} & X_1^{\odot k} & X_2^{\odot k} & \dots & X_{k-1}^{\odot k} & X_j^{\odot k} \end{array} \right).$$

The entries L_j^i can be viewed as non-commutative versions of the classical complete symmetric polynomials (2.21), while the block entries of its inverse, which can also be written in terms of quasi-determinants, [22], are the non-commutative versions of elementary symmetric polynomials, [19, 18].

We now characterize the difference polynomials by their interpolation properties.

Proposition 4.7. *The difference polynomials (4.14) have the form*

$$\Omega_k^i(x) = x^{\odot k} - \Sigma_k^i(x), \quad (4.16)$$

where $\Sigma_k^i(x)$ is the unique (vector-valued) polynomial of degree $k-1$ that interpolates the polynomial $x^{\odot k}$ on the points $x_0, x_1, \dots, x_{m^{(k-1)}-1}$. In other words,

$$\Sigma_k^i(x_\nu) = x_\nu^{\odot k}, \quad \text{and so} \quad \Omega_k^i(x_\nu) = 0, \quad \nu = 0, \dots, m^{(k-1)} - 1. \quad (4.17)$$

The interpolation coefficients $(a_0, \dots, a_n) = (\dots a_I \dots)$ for a function $F(x)$ with interpolating polynomial

$$\Phi(x) = \sum_{k=0}^n a_k \Omega_k(x) \quad (4.18)$$

are computed using the same recursive scheme as in (2.16). We set $H_0(x) = F(x)$, and then recursively define

$$H_{k+1}(x) = H_k(x) - a_k \Omega_k(x), \quad a_k = H_k(X_k) \Omega_k(X_k)^{-1}, \quad k = 0, \dots, n. \quad (4.19)$$

We have thus established a full multivariate version of the classical recursive divided difference recursive scheme.

Example 4.8. In the two-dimensional case, suppose we are given $m^{(2)} = 10$ data points $(x_0, y_0, u_0), (x_1, y_1, u_1), \dots, (x_9, y_9, u_9)$, where we identify $u_\nu = F(x_\nu, y_\nu)$ as sample values of a function $F(x, y)$. To simplify the formulae we will assume that $(x_0, y_0, u_0) = (0, 0, 0)$ so that the interpolating polynomial

$$u = \Phi(x, y) = a_1 x + a_2 y + a_3 \varpi_3(x, y) + a_4 \varpi_4(x, y) + a_5 \varpi_5(x, y) + a_6 \varpi_6(x, y) + a_7 \varpi_7(x, y) + a_8 \varpi_8(x, y) + a_9 \varpi_9(x, y) \quad (4.20)$$

goes through the origin. The general case can then be simply obtained by subtracting x_0, y_0, u_0 from, respectively, x, y, u and x_j, y_j, u_j in the final interpolation formulae. The quadratic difference polynomials $\varpi_3(x, y), \varpi_4(x, y), \varpi_5(x, y)$ are the corresponding entries of $\Omega_2(x, y)$ given in (4.13), namely

$$\begin{aligned} \varpi_3(x, y) &= x^2 - \frac{(x_1^2 y_2 - x_2^2 y_1) x + (x_2 - x_1) x_1 x_2 y}{x_1 y_2 - x_2 y_1}, \\ \varpi_4(x, y) &= 2 \left(xy - \frac{(x_1 - x_2) y_1 y_2 x + (y_2 - y_1) x_1 x_2 y}{x_1 y_2 - x_2 y_1} \right), \\ \varpi_5(x, y) &= y^2 - \frac{(y_1 - y_2) y_1 y_2 x + (x_1 y_2^2 - x_2 y_1^2) y}{x_1 y_2 - x_2 y_1}, \end{aligned} \quad (4.21)$$

while $\varpi_6(x, y), \dots, \varpi_9(x, y)$ are their cubic counterparts, The coefficients of the linear terms in $\varpi_3, \varpi_4, \varpi_5$ are uniquely determined by the interpolation conditions

$$\varpi_k(x_i, y_i) = 0, \quad i = 0, 1, 2, \quad k = 3, 4, 5, \quad (4.22)$$

as in Proposition 4.7. The polynomials $\varpi_1(x, y) = x, \varpi_2(x, y) = y, \varpi_3(x, y), \dots, \varpi_9(x, y)$ play the role of the univariate difference polynomials of degree ≤ 3 .

To determine the explicit formulae for the difference coefficients a_1, \dots, a_9 , we use the recursive scheme (4.19). Since we are assuming $w_0 = 0$, there is no constant term in the interpolation formula, and so we begin with $H_1(x, y) = F(x, y)$. According to (4.19),

$$(a_1 \ a_2) = (u_1 \ u_2) \begin{pmatrix} x_1 & x_2 \\ y_1 & y_2 \end{pmatrix}^{-1}, \quad \text{where} \quad u_j = H_1(x_j, y_j) = F(x_j, y_j).$$

At the next stage

$$H_2(x, y) = H_1(x, y) - (a_1 x + a_2 y) = F(x, y) - (a_1 x + a_2 y),$$

and so the quadratic coefficients are given by

$$(a_3 \ a_4 \ a_5) = (v_3 \ v_4 \ v_5) \begin{pmatrix} x_3^2 & x_4^2 & x_5^2 \\ 2x_3y_3 & 2x_4y_4 & 2x_5y_5 \\ y_3^2 & y_4^2 & y_5^2 \end{pmatrix}^{-1}, \quad \text{where} \quad v_j = H_2(x_j, y_j).$$

The third order terms are computed similarly:

$$H_3(x, y) = H_2(x, y) - [a_3 \varpi_3(x, y) + 2a_4 \varpi_4(x, y) + a_5 \varpi_5(x, y)],$$

whence

$$(a_6 \ a_7 \ a_8 \ a_9) = (w_6 \ w_7 \ w_8 \ w_9) \begin{pmatrix} x_6^3 & x_7^3 & x_8^3 & x_9^3 \\ 3x_6^2y_6 & 3x_7^2y_7 & 3x_8^2y_8 & 3x_9^2y_9 \\ 3x_6y_6^2 & 3x_7y_7^2 & 3x_8y_8^2 & 3x_9y_9^2 \\ y_6^3 & y_7^3 & y_8^3 & y_9^3 \end{pmatrix}^{-1}, \quad w_j = H_3(x_j, y_j).$$

Now let us analyze the coalescent limit of the multivariate divided difference formulae. In analogy with (2.25), we write

$$\Delta^n F = \Delta^n \Phi = (a_0, \dots, a_n) \tag{4.23}$$

for the vector of multi-variate divided differences of order $\leq n$ for the function $F(x)$ on the specified points. We can then write the multivariate interpolating polynomial (4.18) in the compact form

$$\Phi(x) = \Delta^n F \cdot \Omega(x) \tag{4.24}$$

so that

$$F(X) = \Phi(X) = \Delta^n F \cdot \Omega(X)$$

are the defining interpolation conditions relative to the block triangular basis polynomials $\omega(x) = (\omega_0(x), \dots, \omega_n(x))$. Similarly, as in the scalar version (2.27), we write the multivariate monomials in terms of the difference polynomials,

$$P(x) = \Delta^n P \cdot \Omega(x) \quad \text{where} \quad S = P(X) = \Delta^n P \cdot \Omega(X) \tag{4.25}$$

indicates the block LU factorization of the multivariate Vandermonde matrix S .

Given a vector-valued function $G(x) = (G_1(x), \dots, G_k(x))^T$ depending on $x \in \mathbb{R}^m$, let

$$\partial^n G(x) = \left(\frac{\partial^{\#J} G_i}{\partial x^J} \right), \quad 0 \leq \#J \leq n, \quad (4.26)$$

denote its $k \times m^{(n)}$ *multivariate Wronskian matrix*, consisting of all partial derivatives of the components of G up to order n . Differentiation of (4.24) yields

$$\partial^n \Phi(x) = \Delta^n P \cdot \partial^n \Omega(x). \quad (4.27)$$

We note that, since Ω consisting of polynomials, block ordered by their degree, the Wronskian matrix $\partial^n \Omega(x)$ is block lower triangular.

In the univariate situation, the difference Wronskian converges to a diagonal factorial matrix in the coalescent limit, cf. (2.31). This is no longer necessarily valid in the multivariate context; one must control the determinantal denominators appearing in the formulae, e.g., (4.21), as the points coalesce.

Definition 4.9. The points x_0, \dots, x_N for $N + 1 = m^{(n)}$ are said to *coalesce independently* if, as $x_i \rightarrow x_*$, the n^{th} order difference Wronskian matrix $\partial^n \Omega$ converges to a diagonal matrix.

For instance, in the two-dimensional case covered in Example 4.5, independence of a coalescent system of 6 points is assured provided

$$\det \Omega_1(X_1) = \det \widehat{X}_1 = \det \begin{pmatrix} x_1 - x_0 & x_2 - x_0 \\ y_1 - y_0 & y_2 - y_0 \end{pmatrix}$$

remains bounded as the points converge, while the independence of a nine point coalescence requires that

$$\det \Omega_2(X_2) = \det \widehat{X}_2^{\odot 2} - \widehat{X}_1^{\odot 2} \widehat{X}_1^{-1} \widehat{X}_2$$

also remains bounded. The detailed analysis of the coalescence criteria will be deferred until a later paper; see also [55, 57].

Under the independence assumption and Proposition 4.7, the coalescent limit of the multivariate polynomial Wronskian matrix is found to be the block diagonal multivariate factorial matrix

$$\text{c-lim } \partial^n \Omega = \widehat{D} \equiv \text{diag}(0!, 1! I_{m_1}, 2! I_{m_2}, \dots, n! I_{m_n}), \quad (4.28)$$

in which I_{m_j} denotes an $m_j \times m_j$ identity matrix. Furthermore, assuming sufficient smoothness, the derivatives of the interpolating polynomial converge to the corresponding derivatives of the function at the point of coalescence, and so (2.30) becomes

$$\partial^n f = \text{c-lim } \Delta^n F \cdot \widehat{D}, \quad (4.29)$$

which is the multivariate version of the classical divided difference coalescent limit formula (2.32). In other words, under the preceding conditions, the multivariate divided differences converge to the corresponding partial derivatives of the function, divided by a suitable product of factorials.

Summarizing the results of this section, we discovered that, in order to maintain an analogy between the univariate and multivariate interpolation equations, we must split the multi-dimensional Vandermonde matrix (4.6) into block factors and apply the quasi-determinantal block LU factorization. The result is a recursive formula for the interpolation coefficients that, although more complicated, retains the computational advantages of the classical univariate divided differences. In particular, the recursive formulae for the multivariate divided differences imply that one can successively compute higher order interpolants without having to adjust any of the already computed lower order terms. Moreover, assuming sufficient smoothness of the interpolated function F and independent coalescence of the points, the interpolating coefficients converge to the appropriate partial derivatives of F , [57, 10, 55].

5. Curves and Submanifolds.

The purpose of this final section is to formulate a new connection between univariate and multivariate interpolation theory. As usual, the number of independent variables refers to the dimension of the interpolating submanifold. The starting point is the trivial observation that the same set of data can interpolate submanifolds of dissimilar dimensions, i.e., graphs of functions depending on different numbers of independent variables. The relationships among these interpolated submanifolds will lead us to a collection of intriguing formulae for multi-dimensional interpolation coefficients in terms of the classical univariate divided differences. The coalescent limits of these formulae is of great interest. Depending on the mode of coalescence, the univariate divided difference coefficients converge to ordinary derivatives of curves, while, in other regimes, the multivariate difference coefficients converge to partial derivatives of higher dimensional submanifolds. The formulae relating the interpolation coefficients will then reduce to formulae relating the derivatives (or “jets”, [49]) of submanifolds of dissimilar dimensions. As we shall see, the latter formulae can be obtained directly by implicit differentiation. Moreover, they can be used to reconstruct the original formulae connecting the classical divided difference expressions for curves to the complicated multivariate difference formulae for surface and higher dimensional submanifold interpolation.

In order to understand this novel construction, let us concentrate on the simplest case of curves and surfaces in \mathbb{R}^3 . Suppose we are given three points $w_0 = (x_0, y_0, z_0)$, $w_1 = (x_1, y_1, z_1)$, $w_2 = (x_2, y_2, z_2) \in \mathbb{R}^3$ in space. Without loss of generality, we translate the first point to the origin, so $x_0 = y_0 = z_0 = 0$. We first view the three points as interpolating a surface S . In the simplest version, we choose S to be a plane, which we assume is given as the graph of a first order polynomial function

$$z = px + qy. \tag{5.1}$$

This already passes through the origin, and the remaining interpolation conditions

$$z_\nu = px_\nu + qy_\nu, \quad \nu = 1, 2,$$

uniquely prescribe the coefficients

$$p = \frac{y_1 z_2 - y_2 z_1}{x_1 y_2 - x_2 y_1}, \quad q = \frac{x_1 z_2 - x_2 z_1}{x_1 y_2 - x_2 y_1}, \tag{5.2}$$

subject to the condition that the points are poised, meaning that the points $(x_0, y_0) = (0, 0), (x_1, y_1), (x_2, y_2)$ do not lie on a common straight line.

On the other hand, we can view the three points as interpolating a curve $C \subset \mathbb{R}^3$. Again, we focus on the simplest case, when C is a quadratic curve parametrized by x , and so can be expressed using the Newton form of the interpolating polynomial:

$$y = ax + bx(x - x_1), \quad z = cx + dx(x - x_1).$$

Since $x_0 = y_0 = 0$, the coefficients are given by the divided differences

$$\begin{aligned} a &= [y_0y_1] = \frac{y_1}{x_1}, & b &= [y_0y_1y_2] = \frac{x_1y_2 - x_2y_1}{x_1x_2(x_1 - x_2)}, \\ c &= [z_0z_1] = \frac{z_1}{x_1}, & d &= [z_0z_1z_2] = \frac{x_1z_2 - x_2z_1}{x_1x_2(x_1 - x_2)}. \end{aligned} \quad (5.3)$$

Comparing the two sets of formulae (5.2), (5.3), we find that we can write the coefficients p, q of the interpolating surface in terms of the coefficients a, b, c, d of the interpolating curve as follows:

$$p = c - \frac{d}{b}a = [z_0z_1] - \frac{[z_0z_1z_2]}{[y_0y_1y_2]}[y_0y_1], \quad q = \frac{d}{b} = \frac{[z_0z_1z_2]}{[y_0y_1y_2]}. \quad (5.4)$$

Now consider what happens as the points coalesce, $w_1, w_2 \rightarrow 0 = w_0$. If the points follow a curve C parametrized by $y = y(x), z = z(x)$, then the divided difference coefficients (5.3) converge to the curve derivatives,

$$a \longrightarrow \frac{dy}{dx}, \quad b \longrightarrow \frac{1}{2} \frac{d^2y}{dx^2}, \quad c \longrightarrow \frac{dz}{dx}, \quad d \longrightarrow \frac{1}{2} \frac{d^2z}{dx^2}, \quad (5.5)$$

evaluated at the origin. On the other hand, if the points coalesce along a surface S given by the graph $z = z(x, y)$ while maintaining independence, as in Definition 4.9, then the surface coefficients (5.2) converge to the surface derivatives

$$p \longrightarrow \frac{\partial z}{\partial x}, \quad q \longrightarrow \frac{\partial z}{\partial y}, \quad (5.6)$$

again at the origin. Therefore, assuming that the relations (5.4) persist in the limit, we deduce the interesting formulae

$$\frac{\partial z}{\partial x} = z_x - y_x \frac{z_{xx}}{y_{xx}}, \quad \frac{\partial z}{\partial y} = \frac{z_{xx}}{y_{xx}}, \quad (5.7)$$

connecting first order (partial) surface derivatives and second order (ordinary) curve derivatives, which are here denoted by x subscripts.

Remark: The formulae for the two partial derivatives look different, but this is because we are singling out x as the parametric variable. If we chose to parametrize our curve C by y instead, we would find the alternative formulae

$$\frac{\partial z}{\partial x} = \frac{z_{yy}}{x_{yy}}, \quad \frac{\partial z}{\partial y} = z_y - x_y \frac{z_{yy}}{x_{yy}}. \quad (5.8)$$

Extending the formulae to general parametrizations is straightforward; complete details are left to future development of the subject.

Of course, the points cannot actually coalesce along both a curve and a surface in such a manner that they approximate both sets of derivatives in any reasonable manner, and so the interpretation of formulae (5.7) requires some thought. The distinction between the two ways of coalescence can be formalized as follows. If $w_1, w_2 \rightarrow w_0$ along a smooth curve, then the divided difference approximations (5.5) are valid and the error formulae are classical. However, if they coalesce in independent directions along a surface, then the surface formulae (5.2), (5.6) apply; corresponding error formulae can be found in [57]. The independence condition requires that the denominators are well behaved in the limit, and this will happen provided the two unit vectors

$$v_1 = \frac{w_1 - w_0}{\|w_1 - w_0\|}, \quad v_2 = \frac{w_2 - w_0}{\|w_2 - w_0\|},$$

remain linearly independent, and so their cross product $v_1 \wedge v_2 \neq \mathbf{0}$, even in the coalescence limit. On the other hand, if $v_1 \wedge v_2 \rightarrow 0$ as the points coalesce, then one reverts to the curve approximation, with the divided differences converging to the derivatives of the limiting interpolating curve.

How can we make sense of the resulting connection formulae (5.7), or, even better, how can we derive them from first principles? Consider the first order Taylor polynomial

$$z = px + qy, \quad \text{where} \quad p = \frac{\partial z}{\partial x}(0,0), \quad q = \frac{\partial z}{\partial y}(0,0), \quad (5.9)$$

for the surface. If we regard the curve $C \subset S$ as lying in S , then we can differentiate (5.9) with respect to the parameter x , yielding the two equations

$$\frac{dz}{dx} = p + q \frac{dy}{dx}, \quad \frac{d^2z}{dx^2} = q \frac{d^2y}{dx^2},$$

whose solutions are precisely (5.7). Although this derivation is in a sense formal, since it ignores higher order terms in the Taylor expansion, it serves to capture the essence of the connection formulae, and, moreover, provides a computational mechanism for constructing similar formulae in more complex situations.

On a more rigorous level, if the curve parametrized by $y = y(x), z = z(x)$ is contained in the surface $z = z(x, y)$, then it satisfies the restriction

$$z(x, y(x)) = z(x).$$

If we differentiate twice and evaluate at $x = 0$, we find

$$z_x = p + qy_x, \quad z_{xx} = qy_{xx} + r + 2sy_x + ty_x^2, \quad (5.10)$$

where

$$p = \frac{\partial z}{\partial x}, \quad q = \frac{\partial z}{\partial y}, \quad r = \frac{\partial^2 z}{\partial x^2}, \quad s = \frac{\partial^2 z}{\partial x \partial y}, \quad t = \frac{\partial^2 z}{\partial y^2}, \quad (5.11)$$

are the first and second order derivatives of $z(x, y)$ evaluated at the point $(x_0, y_0) = (0, 0)$. Solving (5.10) for p, q , assuming $y_{xx} \neq 0$, we find

$$\frac{\partial z}{\partial x} = p = z_x - qy_x, \quad \frac{\partial z}{\partial y} = q = \frac{z_{xx}}{y_{xx}} - \frac{r + 2sy_x + ty_x^2}{y_{xx}}. \quad (5.12)$$

If we fix the tangent direction to our curve, as given by the first order derivatives y_x, z_x but allow curves with higher and higher curvature, so $y_{xx}, z_{xx} \rightarrow \infty$, then the second term in the formula (5.12) for q goes to zero and, in the “infinite curvature” limit,

$$\frac{\partial z}{\partial x} \longrightarrow z_x - qy_x = \frac{z_x y_{xx} - z_{xx} y_x}{y_{xx}}, \quad \frac{\partial z}{\partial y} \longrightarrow \frac{z_{xx}}{y_{xx}},$$

which are precisely our original formulae (5.7). Therefore, from this point of view, *surfaces are limiting cases of curves as the curvature becomes infinite!*

A good elementary illustration of this phenomenon is to take planar points

$$x_0 = y_0 = 0, \quad x_1 = y_1 = \varepsilon, \quad x_2 = -y_2 = -\varepsilon,$$

where ε is small and positive. Thus, as $\varepsilon \rightarrow 0$, the points coalesce along two independent lines $y = \pm x$. The interpolating quadratic polynomial $y(x) = \varepsilon^{-2} x^2$ has, in the $\varepsilon \rightarrow 0$ limit, infinite curvature at the origin: $y_{xx}(0) = 2/\varepsilon \rightarrow \infty$. Thus, these three points should be viewed as interpolating the (x, y) -plane, not a curve! (Of course, this is only a limiting interpretation, since at any positive ε they do interpolate a perfectly respectable curve, but are poor approximations to its second order derivative.)

Keeping the preceding discussion in mind, let us next discuss the second order case. In order to approximate the 5 derivatives (5.11) of order ≤ 2 for a surface $z = z(x, y)$, we require 6 points z_0, z_1, \dots, z_5 on our surface. On the other hand, the same 6 points will interpolate a smooth curve and can be used to approximate its derivatives up to fifth order by the usual divided difference formulae

$$[y_0 \cdots y_k] \approx \frac{1}{k!} \frac{d^k y}{dx^k}, \quad [z_0 \cdots z_k] \approx \frac{1}{k!} \frac{d^k z}{dx^k}, \quad k = 0, \dots, 5. \quad (5.13)$$

Therefore, the key problem is to assemble combinations that reproduce the interpolation approximations to the surface derivatives (5.11). Rather than tackle this directly, we shall appeal to the limiting case to motivate the proper formulae.

In the coalescent limit, $w_i \rightarrow w_0$, we can proceed as follows. We begin with the multivariate surface interpolating polynomial

$$z = \Phi(x, y) = px + qy + r\varpi_3(x, y) + s\varpi_4(x, y) + t\varpi_5(x, y), \quad (5.14)$$

through the six data points, where the quadratic difference polynomials $\varpi_3, \varpi_4, \varpi_5$, are given in (4.21). The question is: what happens to the multivariate difference polynomials in the coalescent limit? For univariate difference polynomials, this is not a significant issue, since they converge to the standard monomials $\omega_k(x) \rightarrow x^k$. However, in the multivariate version, this limit is not so elementary. The correct condition that replaces the interpolation condition (4.22) is that the limiting quadratic difference polynomials

$$\varpi_3(x, y) = x^2 + c_3 x + d_3 y, \quad \varpi_4(x, y) = 2xy + c_4 x + d_4 y, \quad \varpi_5(x, y) = y^2 + c_5 x + d_5 y, \quad (5.15)$$

vanish to order 2 at the origin $x_0 = y_0 = 0$ whenever $y(x)$ is a parametrized curve. Differentiating these equations, we find

$$\begin{aligned}
\frac{d\varpi_3}{dx} &= 2x + c_3 + d_3 y_x, & \frac{d^2\varpi_3}{dx^2} &= d_3 y_{xx}, \\
\frac{d\varpi_4}{dx} &= 2y + 2xy_x + c_4 + d_4 y_x, & \frac{d^2\varpi_4}{dx^2} &= 4y_x + 2xy_{xx} + d_4 y_{xx}, \\
\frac{d\varpi_5}{dx} &= 2yy_x + c_5 + d_5 y_x, & \frac{d^2\varpi_5}{dx^2} &= 2yy_{xx} + 4y_x^2 + d_5 y_{xx},
\end{aligned}$$

which all vanish at $x = y = 0$ when

$$\begin{aligned}
\varpi_3(x, y) &= x^2 - 2 \frac{y - xy_x}{y_{xx}}, & \varpi_4(x, y) &= 2xy - 4y_x \frac{y - xy_x}{y_{xx}}, \\
\varpi_5(x, y) &= y^2 - 2y_x^2 \frac{y - xy_x}{y_{xx}}.
\end{aligned} \tag{5.16}$$

In these coalescent formulae, the independent variables x, y are allowed to vary, but the derivatives $y_x = y_x(0)$, $y_{xx} = y_{xx}(0)$ are kept fixed at their values at the point of coalescence. As the reader can check, these are the coalescent limits of the difference polynomials (4.21), and satisfy $\varpi_k(0, 0) = \frac{d\varpi_k}{dx}(0, 0) = \frac{d^2\varpi_k}{dx^2}(0, 0) = 0$ along any parametrized curve $y = y(x)$ passing through the origin.

Thus, the connection formulae, relating curve and surface derivatives, can be derived by differentiating the coalescent surface interpolating polynomial (5.14), where the quadratic polynomials are given by (5.16), with respect to x . The result is a linear system

$$z_x = p + qy_x, \quad z_{xx} = qy_{xx}, \quad \frac{d^k z}{dx^k} = q \frac{d^k y}{dx^k} + r \frac{d^k \varpi_3}{dx^k} + s \frac{d^k \varpi_4}{dx^k} + t \frac{d^k \varpi_5}{dx^k}, \quad k = 3, 4, 5,$$

for the surface derivatives p, q, r, s, t . We can write this in matrix form

$$\mathbf{z} = \mathbf{p}W \quad \text{where} \quad \mathbf{z} = (z_x, z_{xx}, z_{3x}, z_{4x}, z_{5x}), \quad \mathbf{p} = (p, q, r, s, t),$$

while W is the 5×5 reduced[†] Wronskian matrix

$$\left(\begin{array}{cc|ccc}
1 & 0 & 0 & 0 & 0 \\
y_x & y_{xx} & y_{3x} & y_{4x} & y_{5x} \\
\hline
0 & 0 & -2 \frac{y_{3x}}{y_{xx}} & -2 \frac{y_{4x}}{y_{xx}} & -2 \frac{y_{5x}}{y_{xx}} \\
0 & 0 & \frac{6y_{xx}^2 - 4y_x y_{3x}}{y_{xx}} & \frac{8y_{xx} y_{3x} - 4y_x y_{4x}}{y_{xx}} & \frac{10y_{xx} y_{4x} - 4y_x y_{5x}}{y_{xx}} \\
0 & 0 & \frac{6y_x y_{xx}^2 - 2y_x^2 y_{3x}}{y_{xx}} & \frac{6y_{xx}^3 + 8y_x y_{xx} y_{3x} - 2y_x^2 y_{4x}}{y_{xx}} & \frac{20y_{xx}^2 y_{3x} + 10y_x y_{xx} y_{4x} - 2y_x^2 y_{5x}}{y_{xx}}
\end{array} \right) \tag{5.17}$$

[†] Since we omit the first column corresponding to the undifferentiated function, which has been normalized by our choice $w_0 = 0$.

whose columns consist of the x derivatives of $\varpi_1(x, y), \dots, \varpi_5(x, y)$ treating $y = y(x)$ as a function of x . Note that W is in block upper triangular form.

Indeed, we can compute W directly as follows. Start with the reduced Taylor Wronskian matrix

$$\left(\begin{array}{cc|ccc} 1 & 0 & 0 & 0 & 0 \\ y_x & y_{xx} & y_{3x} & y_{4x} & y_{5x} \\ \hline 2x & 2 & 0 & 0 & 0 \\ 2xy_x + 2y & 2xy_{xx} + 4y_x & 2xy_{3x} + 6y_{xx} & 2xy_{4x} + 8y_{3x} & 2xy_{5x} + 10y_{4x} \\ 2yy_x & 2yy_{xx} + 2y_x^2 & 2yy_{3x} + 6y_x y_{xx} & 2yy_{4x} + 8y_x y_{3x} + 6y_{xx}^2 & 2yy_{5x} + 10y_x y_{4x} + 20y_{xx} y_{3x} \end{array} \right)$$

based on the standard monomials

$$x, \quad y, \quad x^2, \quad 2xy, \quad y^2.$$

We evaluate the Taylor Wronskian at $x = y = 0$, yielding

$$Z = \left(\begin{array}{cc|ccc} 1 & 0 & 0 & 0 & 0 \\ y_x & y_{xx} & y_{3x} & y_{4x} & y_{5x} \\ \hline 0 & 2 & 0 & 0 & 0 \\ 0 & 4y_x & 6y_{xx} & 8y_{3x} & 10y_{4x} \\ 0 & 2y_x^2 & 6y_x y_{xx} & 8y_x y_{3x} + 6y_{xx}^2 & 10y_x y_{4x} + 20y_{xx} y_{3x} \end{array} \right). \quad (5.18)$$

Then W is obtained by block LU factorization of Z , so

$$Z = K W, \quad \text{where} \quad K = \left(\begin{array}{cc|ccc} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ \hline -\frac{2y_x}{y_{xx}} & \frac{2}{y_{xx}} & 1 & 0 & 0 \\ -\frac{4y_x^2}{y_{xx}} & \frac{4y_x}{y_{xx}} & 0 & 1 & 0 \\ \frac{2y_x^3}{y_{xx}} & \frac{2y_x^2}{y_{xx}} & 0 & 0 & 1 \end{array} \right) \quad (5.19)$$

is special lower triangular. This factorization corresponds to replacing the Taylor expansion

$$z = P(x, y) = ax + by + \frac{1}{2} r x^2 + sxy + \frac{1}{2} t y^2, \quad (5.20)$$

by the alternative coalescent expansion (5.14). The coefficients are related by

$$(a, b, \frac{1}{2} r, \frac{1}{2} s, \frac{1}{2} t) = (p, q, \frac{1}{2} r, \frac{1}{2} s, \frac{1}{2} t) K.$$

Note that a, b are higher order approximations to the first order surface derivatives, in the infinite curvature limit.

Let us conclude by placing the preceding constructions into a general framework. Given a function $F: \mathbb{R}^m \rightarrow \mathbb{R}$ depending on $x = (x^1, \dots, x^m)$ and a poised collection of $N + 1 = m^{(n)}$ points $X = (x_0, \dots, x_N)$, we can construct the multivariate interpolant (4.24). On the other hand, suppose we view the interpolation points as lying on a curve parametrized by, say, the first coordinate $t = x^1$. Let t_i be the parameter value corresponding to the i^{th} interpolation point $x_i = x(t_i)$, and set $T = (t_0, \dots, t_N)$ so that $X = x(T)$ are the interpolation points. We write the univariate vector-valued interpolating polynomial $\varphi(t) = \delta^n F \cdot \omega(t)$ of degree $N = m^{(n)} - 1$ in the abbreviated Newton form (2.26), where the divided difference coefficients are prescribed by the interpolation conditions

$$F(X) = \varphi(T) = \delta^n F \cdot \omega(T) \quad \text{at} \quad X = x(T). \quad (5.21)$$

Similarly, interpolating the multivariate divided difference functions $\Omega_i(x)$ by univariate polynomials of degree N , we have

$$\Omega(X) = \delta^n \Omega \cdot \omega(T). \quad (5.22)$$

Combining (4.24) and (5.22),

$$F(X) = \Delta^n F \cdot \Omega(X) = \Delta^n F \cdot \delta^n \Omega \cdot \omega(T);$$

equating the result to (5.21), we deduce

$$\delta^n f = \Delta^n F \cdot \delta^n \Omega. \quad (5.23)$$

In the coalescent limit, in view of (2.32), (4.29), we obtain

$$d^n f = \partial^n f \cdot \widehat{D} \cdot d^n \omega, \quad (5.24)$$

which is the coalescent limit of the difference formula (5.23) relating the multivariate and univariate divided differences. Of course, as noted above, the formulae (5.24) rely on two very different types of coalescent limits; coalescing along a curve, as in (2.32) will violate the independence conditions required for the multivariate coalescence formulae (4.29). The precise geometric interpretation of the connection formulae (5.24) will be developed in more depth in subsequent publications.

On the other hand, the original multivariate monomials $P(x)$ can be rewritten in terms of the difference polynomials using (4.24):

$$P(x) = \Delta^n P \cdot \Omega(x), \quad \text{where} \quad P(X) = \Delta^n P \cdot \Omega(X)$$

is the block LU factorization of the multivariate Vandermonde matrix. Therefore,

$$d^n P = \Delta^n P \cdot d^n \Omega.$$

In the coalescent limit, this reduces to the block LU factorization of the multivariate Wronskian

$$d^n P = Z = K W, \quad (5.25)$$

where

$$K = \text{c-lim } \Delta^n P, \quad W = \text{c-lim } d^n \Omega. \quad (5.26)$$

This factorization formula is the general version the two-dimensional formula (5.19).

One final word of caution: The formulae (5.23) for the multivariate divided differences in terms of their univariate counterparts cannot be unambiguously reconstructed from the limiting formula for the derivatives (5.26). This is because the replacement of a function by its divided difference approximation is not an algebra morphism — it does *not* respect multiplication. For example, given $y(x)$, the divided difference approximations to $p(x) = y'(x)y''(x)$ are *not* the products of divided differences for $y'(x)$ times those for $y''(x)$! In other words, a combination of derivatives of one or more functions can lead to different divided difference approximations, all of the same order. Thus, when forming the block factorization of the Wronskian (5.25), one needs to be careful in keeping track of algebraic operations that are not reflected in their divided difference counterparts. An interesting question is whether the correspondence between the univariate and multivariate divided difference approximations is maintained under such replacements. On the surface, this expectation seems reasonable, but a complete resolution of this issue will be delegated to future research.

Acknowledgments: I would particularly like to express my gratitude to Gil Strang for his interest, valuable suggestions, references, and much-needed encouragement. I also wish to thank George Andrews, Carl de Boor, Israel Gel'fand, Nick Higham, Arieh Iserles, Alain Lascoux, Randy LeVeque, Vladimir Retakh, Olivier Ruatta, Dennis Stanton, Patrick Van Fleet, and an anonymous referee for many useful suggestions and references to the literature.

References

- [1] Andrews, G.E., personal communication, 2003.
- [2] Birkhoff, G., The algebra of multivariate interpolation, in: *Constructive Approaches to Mathematical Models*, C.V. Coffmann and G.J. Fix, eds., Academic Press, New York, 1979, pp. 345–363.
- [3] Björck, Å., and Pereyra, V., Solution of Vandermonde systems of equations, *Math. Comp.* **24** (1970), 893–903.
- [4] Buhmann, M.D., Radial basis functions, *Acta Numerica* **9** (2000), 1–38.
- [5] Chan, T.F., and Vassilevski, P.S., A framework for block ILU factorizations using block-size reduction, *Math. Comp.* **64** (1995), 129–156.
- [6] Chui, C.K., and Lai, H.-C., Vandermonde determinants and Lagrange interpolation in \mathbb{R}^s , in: *Nonlinear and Convex Analysis*, B.-L. Lin, S. Simons, eds.; Lecture Notes Pure Appl. Math.; vol. 107, Marcel Dekker, Inc., New York, 1988, pp. 23–35.
- [7] Davis, P.J., *Interpolation and Approximation*, Dover Publ. Inc., New York, 1975.

- [8] de Boor, C., Gauss elimination by segments and multivariate polynomial interpolation, in: *Approximation and Computation*, R.V.M. Zahar, ed., Birkhäuser, Boston, 1994, pp. 1–22.
- [9] de Boor, C., A multivariate divided difference, in: *Approximation theory VIII*, Vol. 1, C.K. Chui and L.L. Schumaker, eds., World Sci. Publishing, River Edge, NJ, 1995, pp. 87–96.
- [10] de Boor, C., On the Sauer-Xu formula for the error in multivariate polynomial interpolation, *Math. Comp.* **65** (1996), 1231–1234.
- [11] de Boor, C., What is the inverse of a basis?, *BIT* **41** (2001), 880–890.
- [12] de Boor, C., and Ron, A., On multivariate polynomial interpolation, *Constr. Approx.* **6** (1990), 287–302.
- [13] de Boor, C., and Ron, A., Computational aspects of polynomial interpolation in several variables, *Math. Comp.* **58** (1992), 705–727.
- [14] Desbrun, M., Hirani, A., Marsden, J.E.; Discrete Poincaré lemma; CDS Technical Report, Cal Tech, 2003.
- [15] Etinghof, P., Gel’fand, I.M., and Retakh, V.S., Factorization of differential operators, quasideterminants, and nonabelian Toda field equations, *Math. Res. Lett.* **4** (1997), 413–425.
- [16] Etinghof, P., Gel’fand, I.M., and Retakh, V.S., Nonabelian integrable systems, quasideterminants, and Marchenko lemma, *Math. Res. Lett.* **5** (1998), 1–12.
- [17] Federer, H., *Geometric Measure Theory*, Springer–Verlag, New York, 1969.
- [18] Gel’fand, I.M., Gel’fand, S., Retakh, V.S., and Wilson, R., Quasideterminants, preprint, [math.QA/0208146](https://arxiv.org/abs/math.QA/0208146), 2003.
- [19] Gel’fand, I.M., Krob, D., Lascoux, A., Leclerc, B., Retakh, V.S., and Thibon, J.-Y., Noncommutative symmetric functions, *Adv. Math.* **112** (1995), 218–348.
- [20] Gel’fand, I.M., and Retakh, V.S., Determinants of matrices over noncommutative rings, *Func. Anal. Appl.* **25** (1991), 91–102.
- [21] Gel’fand, I.M., and Retakh, V.S., Theory of noncommutative determinants, and characteristic functions of graphs, *Func. Anal. Appl.* **26** (1992), 231–246.
- [22] Gel’fand, I.M., and Retakh, V.S., Quasideterminants. I, *Selecta Math. (N.S.)* **3** (1997), 517–546.
- [23] Glaeser, G., L’interpolation des fonctions différentiables de plusieurs variables, in: *Proceedings of Liverpool Singularities Symposium II*, C.T.C. Wall, ed., Lecture Notes in Math., vol. 209, Springer–Verlag, New York, 1971, pp. 1–33.
- [24] Gohberg, I., and Koltracht, I., Triangular factors of Cauchy and Vandermonde matrices, *Integral Eq. Operator Theory* **26** (1996), 46–59.
- [25] Golub, G.H., and Van Loan, C.F., *Matrix Computations*, Third Edition, Johns Hopkins Univ. Press, Baltimore, 1996.
- [26] Hager, W., Updating the inverse of a matrix, *SIAM Rev.* **31** (1989), 221–239.
- [27] Hairer, E., Lubich, C., and Wanner, G., *Geometric Numerical Integration*, Springer–Verlag, New York, 2002.
- [28] Hakopian, H.A., Multivariate divided differences and multivariate interpolation of Lagrange and Hermite type, *J. Approx. Theory* **34** (1982), 286–305.

- [29] Hamann, B., Curvature approximation for triangulated surfaces, *Geom. Model., Comput. Suppl.* **8** (1993), 139–153.
- [30] Hermann, R., A Lie–theoretic setting for the classical interpolation theories, *Acta Appl. Math.* **6** (1986), 275–292.
- [31] Heyting, A., Die Theorie der linearen Gleichungen in einer Zahlenspezies mit nichtkommutativer Multiplikation, *Math. Ann.* **98** (1928), 465–490.
- [32] Higham, N.J., Error analysis of the Björck–Pereyra algorithms for solving Vandermonde systems, *Numer. Math.* **50** (1987), 613–632.
- [33] Higham, N.J., *Accuracy and Stability of Numerical Algorithms*, Second Edition, SIAM, Philadelphia, 2002.
- [34] Huiskamp, G., Difference formulas for the surface Laplacian on a triangulated surface, *J. Comp. Phys.* **95** (1991), 477–496.
- [35] Iserles, A., Munthe–Kaas, H.Z., Nørsett, S.P., and Zanna, A., Lie group methods, *Acta Numerica* (2000), 215–365.
- [36] Kailath, T., Sayed, A.H., and Hassibi, B., *Linear Estimation*, Prentice–Hall, Inc., Upper Saddle River, N.J., 2000.
- [37] Kergin, P., A natural interpolation of C^k functions, *J. Approx. Theory* **29** (1980), 278–293.
- [38] Krattenthaler, C., Advanced determinant calculus, *Sém. Lothar. Comb.* **42** (1999), .
- [39] Kunkle, T., Multivariate differences, polynomials, and splines, *J. Approx. Theory* **84** (1996), 290–314.
- [40] Lascoux, A., and Schützenberger, M.-P., Interpolation de Newton à plusieurs variables, in: *Séminaire d’Algèbre Paul Dubreil et Marie–Paule Malliavin*, M.-P. Malliavin, ed., Lecture Notes in Math. vol. 1146, Springer–Verlag, New York, 1985, pp. 161–175.
- [41] Macdonald, I.G., *Symmetric Functions and Hall Polynomials*, Clarendon Press, Oxford, 1979.
- [42] Mansfield, E.L., and Hydon, P.E., A variational complex for difference equations, *Found. Comput. Math.*, to appear.
- [43] Micchelli, C.A., A constructive approach to Kergin interpolation in \mathbb{R}^k : multivariate B -splines and Lagrange interpolation, *Rocky Mountain J. Math.* **10** (1980), 485–497.
- [44] Micchelli, C.A., and Milman, P., A formula for Kergin interpolation in \mathbb{R}^k , *J. Approx. Theory* **29** (1980), 294–296.
- [45] Milne–Thompson, L.M., *The Calculus of Finite Differences*, Macmilland and Co., Ltd., London, 1951.
- [46] Mourrain, B., and Ruatta, O., Relations between roots and coefficients, interpolation and application to system solving, *J. Symb. Comp.* **33** (2002), 679–699.
- [47] Neamtu, M., Multivariate divided difference. I. Basic properties, *SIAM J. Numer. Anal.* **29** (1992), 1435–1445.

- [48] Olshevsky, V., Unitary Hessenberg matrices and the generalized Parker–Forney–Traub algorithm for inversion of Szegö–Vandermonde matrices, preprint, Georgia State University, 2003.
- [49] Olver, P.J., *Applications of Lie Groups to Differential Equations*, Second Edition, Graduate Texts in Mathematics, vol. 107, Springer–Verlag, New York, 1993.
- [50] Olver, P.J., Geometric foundations of numerical algorithms and symmetry, *Appl. Alg. Engin. Commun. Comput.* **11** (2001), 417–436.
- [51] Olver, P.J., and Shakiban, C., *Applied Mathematics*, Prentice–Hall, Inc., Upper Saddle River, N.J., to appear.
- [52] Oostendorp, T.F., van Oosterom, A., and Huiskamp, G., Interpolation on a triangulated 3D surface, *J. Comp. Phys.* **80** (1989), 331–343.
- [53] Orucc, H., and Phillips, G. M., Explicit factorization of the Vandermonde matrix, *Linear Algebra Appl.* **315** (2000), 113–123.
- [54] Powell, M.J.D., *Approximation theory and Methods*, Cambridge University Press, Cambridge, 1981.
- [55] Rabut, C., Multivariate divided differences with simple knots, *SIAM J. Num. Anal.* **38** (2000), 1294–1311.
- [56] Salzer, H.E., Divided differences for functions of two variables for irregularly spaced arguments, *Numer. Math.* **6** (1964), 68–77.
- [57] Sauer, T., and Xu, Y., On multivariate Lagrange interpolation, *Math. Comp.* **64** (1995), 1147–1170.
- [58] Strang, G., *Linear Algebra and its Applications*, Third Ed., Academic Press, San Diego, 1988.
- [59] Tang, W.P., and Golub, G.H., The block decomposition of a Vandermonde matrix and its applications, *BIT* **21** (1981), 505–517.
- [60] Whittaker, E., and Robinson, G., *The Calculus of Observations*, 4th ed., Blackie & Sons, Inc., London, 1944.
- [61] Yang, Y., and Holtti, H., The factorization of block matrices with generalized geometric progression rows, preprint, University of St. Thomas, 2003.
- [62] Zippel, R., Interpolating polynomials from their values, *J. Symb. Comp.* **9** (1990), 375–403.