Numerical cubature on scattered data by adaptive interpolation

Roberto Cavoretto[a], Alessandra De Rossi[a], Francesco Dell'Accio[b,c],
Filomena Di Tommaso[b,c], Najoua Siar[d], Alvise Sommariva[e], Marco Vianello[e]

[a]*University of Torino, via Carlo Alberto 10I, Torino, 10123, Italy*
[b]*University of Calabria, via Pietro Bucci, cubo 30/A, Rende, 87036, Italy*
[c]*Istituto per le Applicazioni del Calcolo "Mauro Picone" - Consiglio Nazionale delle Ricerche, Via Pietro Castellino 111, Napoli, 80131, Italy*
[d]*Faculty of Sciences, University Moulay Ismail, B.P. 11201 Zitoune, Meknès, Morocco*
[e]*University of Padova, via Trieste, 63, Padova, 35121, Italy*

Abstract

We construct cubature methods on scattered data via resampling on the support of known algebraic cubature formulas, by different kinds of adaptive interpolation (polynomial, RBF, PUM). This approach gives a promising alternative to other recent methods, such as direct meshless cubature by RBF or least-squares cubature formulas.

Keywords: Scattered data, Numerical cubature, Algebraic formulas, Adaptive interpolation, RBF, PUM, Multinode Shepard method

1. Introduction

The problem of computing integrals of multivariate functions on standard or nonstandard domains via function sampling is ubiquitous in scientific and technological applications. There is a vast literature on the construction of cubature formulas exact in specific function spaces, especially polynomial spaces, where the sampling has to be accomplished on suitable set of nodes. We do not even attempt to give an overview of this specialized literature, referring the reader to some classical or more recent monographs and surveys, such as for example [13, 37, 67].

On the other hand, it is a common situation that the data are scattered and still one would like to obtain an integration method from such data.

One of the most popular approaches, especially in high-dimension, is QMC (Quasi MonteCarlo) method, provided that the sampling is made on low-discrepancy node sequences; again, we only mention within the vast QMC literature an excellent survey like [31]. While QMC has the advantage of not suffering from the curse of dimensionality, it is however scarcely accurate and thus not the best choice in low dimension.

Numerical cubature from scattered data constructed to be exact on suitable function spaces, typically in low dimension, has received some attention in recent years. Indeed, we may quote methods based on meshless interpolation, namely by Radial Basis Functions (RBF), apparently considered for the first time in [61, 62] and then further developed in [12, 40, 63], or by RBF-FD (radial basis function-generated finite difference) [36] used to approximate integrals over curved surfaces with boundaries [54] and over the surface of a sphere [55] that was generalized to arbitrarily shaped smooth closed surfaces [56]. Another approach, based on ℓ^1 or ℓ^2 minimization of the weights under polynomial moment-matching conditions, was proposed in [38], where it was shown to be more accurate than MC and QMC integration.

Differently, here we construct cubature methods on scattered data via resampling on the support of known algebraic cubature formulas, by adaptive interpolation of different kinds. The paper is organized as follows. In Section 2, we describe the fundamentals of our approach and the different kinds of interpolation adopted. In Section 3, we discuss a number of cubature tests, showing that the present method can be a valid alternative to the scattered integration methods quoted above.

2. Cubature by adaptive scattered interpolation

The basic idea of our approach is the following. Given an algebraic cubature rule with positive weights and interior nodes (PI rule) on a compact domain $\Omega \subset \mathbb{R}^s$

$$I_\Omega(f) = \int_\Omega f(P) \, w(P) \, dP \approx Q_n(f) = \sum_{k=1}^{\nu} w_k \, f(\Xi_k), \qquad (1)$$

which is exact for $f \in \mathbb{P}_n$ (the space of multivariate polynomials of total degree not exceeding n, with $m_s = dim(\mathbb{P}_n) = \binom{n+s}{s}$), and a suitable interpolant on a scattered set $X = \{P_1, \ldots, P_N\} \subset \Omega$, say $\Psi = \Psi_{f,X} : \Omega \to \mathbb{R}$, we

approximate the cubature rule as

$$Q_n(f) = \sum_{k=1}^{\nu} w_k \, f(\Xi_k) \approx Q_n(\Psi) = \sum_{k=1}^{\nu} w_k \, \Psi(\Xi_k). \tag{2}$$

Notice that Ψ could be a standard global interpolant such that $\Psi(P_i) = f(P_i)$, $1 \le i \le N$, but also a function pointwise defined by local interpolation on a suitable subset of the scattered points, in a moving interpolation fashion (see for example Subsection 2.1).

We stress that in this approach we do not approximate the integral of f by the integral of the interpolant, which is the standard way to construct a cubature formula via interpolation, rather we consider the values of the interpolant at the cubature nodes of a known formula as perturbations of the function values there. In other words, we use the interpolant only as an approximate evaluator of the integrand at the cubature nodes, that is to *approximate the cubature formula*. This approach has already been used to approximate Gaussian quadrature formulas by various local or global interpolants from univariate equispaced sampling, see for example [50, 46, 25, 26]. See also product integration rules with a similar algorithm [27, 42, 43].

Now, denoting by $p_n^* \in \mathbb{P}_n$ the best uniform polynomial approximation to $f \in C(\Omega)$, we can write

$$I_\Omega(f) - Q_n(\Psi) = (I_\Omega(f) - I_\Omega(p_n^*)) + (I_\Omega(p_n^*) - Q_n(p_n^*))$$

$$+(Q_n(p_n^*) - Q_n(f)) + (Q_n(f) - Q_n(\Psi)), \tag{3}$$

and then, observing that the second summand is null by polynomial exactness of the cubature formula, we get

$$|I_\Omega(f) - Q_n(\Psi)| \le I_\Omega(|f - p_n^*|) + Q_n(|p_n^* - f|) + Q_n(|f - \Psi|)$$

$$\le \|w\|_{L^1} \left(2\, E_n(f; \Omega) + \max_k |f(\Xi_k) - \Psi(\Xi_k)| \right), \tag{4}$$

where $E_n(f; \Omega) = \inf_{p \in \mathbb{P}_n} \|f - p\|_{\infty, \Omega}$.

We recall that the infinitesimal order of $E_n(f; \Omega)$ can be estimated on the so-called Jackson compact sets, that are compact domains for which there exist a positive integer k_j, $j = 0, 1, 2, \ldots$, and a constant $c_j(f)$ (depending on the partial derivatives of f up to order j), such that if $f \in C^{k_j}(\Omega)$ a Jackson-like inequality holds, of the form $E_n(f) \le c_j(f)\, n^{-j}$. Two basic

3

examples are hypercubes with $k_j = j + 1$ and Euclidean balls with $k_j = j$. In particular a sufficient condition is that Ω be Whitney regular (that is, there exists a number $C \geq 1$ such that any two points $x, y \in \Omega$ can be joined by a rectifiable curve in Ω of length bounded by $C|x - y|$; for more details see [32, 72]) and admits a Markov polynomial inequality; cf. [53] for a survey of known results on Jackson sets.

Therefore, taking into account that the scattered interpolation nodes are fixed, by increasing n, it follows from equations (3)-(4), that the error of the approximate cubature $Q_n(\Psi)$ will stagnate on the size of the maximum interpolation error at the cubature nodes. Such an error depends on the choice of the interpolant, on the regularity of f, and it is a not-decreasing function of n bounded above by $\max_{P \in \Omega} |f(P) - \Psi(P)|$.

The reason for working with algebraic cubature is that Positive Interior (PI) formulas exact in polynomial spaces are available in a variety of domains. We cannot give here an extended bibliography, but we may quote some relevant items. For example, minimal or near-minimal formulas are known, at least up to certain degrees, on standard domains like squares/cubes, disks/balls and triangles/tetrahedra, cf. [13, 14, 37, 35, 67, 68] with the references therein. On the other hand, low-cardinality formulas of PI type have been constructed on linear polygons/polyhedra [44, 66], on domains with spline and NURBS boundaries [65], and on several domains corresponding to a trigonometric polynomial parametrization of the boundary such as for example circular segments, sectors, zones, lunes, as well as disk intersection and union, cf. [15, 16, 64] with the references therein. Many of these formulas use a NNLS implementation of Tchakaloff-like compression, cf. e.g. [29, 30, 52]. Other constructions of algebraic PI formulas on multidimensional domains can be found e.g. in [45, 48], only to quote some recent contributions (see indeed the references therein). Thus, instead of trying to construct a cubature formula directly supported at the scattered sites, as done for example in [12, 38], we use indirectly the scattered data to approximate one within such a vast and still growing body of available algebraic rules.

As a preliminary study, below we briefly recall and study numerically several different adaptive interpolation methods on scattered data, comparing their approximation power on some test functions. Adaptivity is indeed the key to obtain high accuracy from such interpolants. We stress that here we are mainly interested in accuracy rather than computational efficiency, since

4

our main goal will be approximating integrals as accurately as possible by the fixed scattered data.

2.1. Adaptive moving polynomial interpolation

We discuss a pointwise evaluation method by adaptive local polynomial interpolation. In order to clarify its theoretical base, we state the main error estimate result, reformulating the theorem in [23]. Indeed, such a theorem concerns pointwise numerical differentiation of arbitrary order via local interpolation, whereas here we are merely interested in pointwise function evaluation. For simplicity, we restrict to bivariate functions.

Below we denote by $\mathbb{P}_d$ the space of bivariate polynomials of total degree not exceeding d with dimension $m_d = dim(\mathbb{P}_d) = (d+1)(d+2)/2$, by $B_h(\overline{P})$ the closed Euclidean ball of radius h centered at $\overline{P} = (\overline{x}, \overline{y})$, by $D^\alpha = \partial_x^{\alpha_1} \partial_y^{\alpha_2}$ the differentiation operator with respect to bi-indexes $\alpha = (\alpha_1, \alpha_2)$ of length $|\alpha| = \alpha_1 + \alpha_2$ (lexicographically ordered), and by $C^{d,1}(K)$ the space of C^d functions with Lipschitz-continuous derivatives of length d on a compact domain $K \subset \mathbb{R}^2$ (the closure of a bounded open connected set), equipped with the seminorm

$$\|f\|_{C^{d,1}(K)} = \sup \left\{ \frac{|D^\alpha f(P) - D^\alpha f(Q)|}{\|P - Q\|_2} : P, Q \in K, \ P \neq Q, \ |\alpha| = d \right\}. \quad (5)$$

Moreover, we adopt the usual notation with multi-indices where powers are interpreted componentwise, $P^\alpha = x^{\alpha_1} y^{\alpha_2}$.

Theorem 1. *Let $\Omega \subset \mathbb{R}^2$ be a convex body, $f \in C^{d,1}(\Omega)$, and $\pi_{h,d} \in \mathbb{P}_d$ the interpolating polynomial of f at a unisolvent subset*

$$X_d = \{\xi_1, \dots, \xi_{m_d}\} \subset \mathcal{N}_h = B_h(\overline{P}) \cap \Omega. \quad (6)$$

Then the following pointwise error estimate holds

$$E_{h,d}(\overline{P}) = \left| f(\overline{P}) - \pi_{h,d}(\overline{P}) \right| \leq \lambda_{h,d}(\overline{P}) \, C_{h,d} \, h^{d+1}, \quad (7)$$

$$C_{h,d} = \frac{2^d}{(d-1)!} \|f\|_{C^{d,1}(\mathcal{N}_h)}, \quad \lambda_{h,d}(\overline{P}) = \|\rho_1(V_{h,d}^{-1})\|_1,$$

where ρ_1 denotes the first row of the inverse Vandermonde matrix in the scaled monomial basis centered at $\overline{P}$

$$V_{h,d} = \left[((\xi_i - \overline{P})/h)^\alpha \right], \ 1 \leq i \leq m_d, \ |\alpha| \leq d. \quad (8)$$

For a proof of this result, based on local approximation by Taylor formula, we refer the reader to [23]. We stress that $\lambda_{h,d}(\overline{P})$ is nothing but the Lebesgue function of X_d evaluated at the neighborhood of the center $\overline{P}$, where the representation as 1-norm of the first Vandermonde row comes from the choice of the polynomial basis. The result above can be extended to non-convex domains that satisfy the well-known "Whitney regularity" property [71, §2], by a "curved" version of the Taylor formula, cf. e.g. [51, Ch.8]. In this case estimate (7) is qualitatively the same, since the Taylor formula remainder is still $\mathcal{O}(h^{d+1})$ (we do not give a proof for the sake of concision).

The algorithm here adopted (2-dimensional moving interpolation), which is a derivation of the algorithm DISC proposed in [24] where we refer the reader for all details, works adaptively on the radius h and the degree d, trying to minimize (an estimate of) the pointwise evaluation error. The local interpolation points X_d are "discrete Leja points" extracted from $X \cap \mathcal{N}_h$ (we recall that X are the fixed scattered sites), that are points aimed at maximizing the Vandermonde determinant modulus. They are selected greedily by standard Gaussian elimination with row pivoting on the local transposed Vandermonde matrix, following the method of [2] for stable multivariate polynomial interpolation.

The choice of discrete Leja points as local interpolation points is suggested by several features. First, being aimed at maximizing (in a greedy way) the Vandermonde determinant modulus, they guarantee *unisolvency*, provided that the local scattered points $X \cap \mathcal{N}_h$ are $\mathbb{P}_d$-determining (i.e., polynomials in $\mathbb{P}_d$ vanishing there vanish everywhere, or equivalently the corresponding rectangular Vandermonde matrix is full-rank, that is $X \cap \mathcal{N}_h$ contains a unisolvent interpolation set). Though this is not true in general with any scattered sample, it is almost surely verified with uniformly distributed points with respect to any continuous density, as recently proved in [28] for general interpolation by a.e. analytic functions.

The second reason for the choice of discrete Leja points is interpolation *stability*. Indeed, their choice tries to keep small the entries of the inverse Vandermonde matrix (which are cofactors divided by the Vandermonde determinant), and thus also the relevant row 1-norm in estimate (7). We observe also that the choice of a monomial basis centered at $\overline{P}$, and scaled by the ball radius, allows to control the Vandermonde matrix conditioning, at least up to moderate interpolation degrees (say around degree 10).

The third important feature of discrete Leja points is that they form a *sequence*, differently from other extremal interpolation sets like discrete Fekete
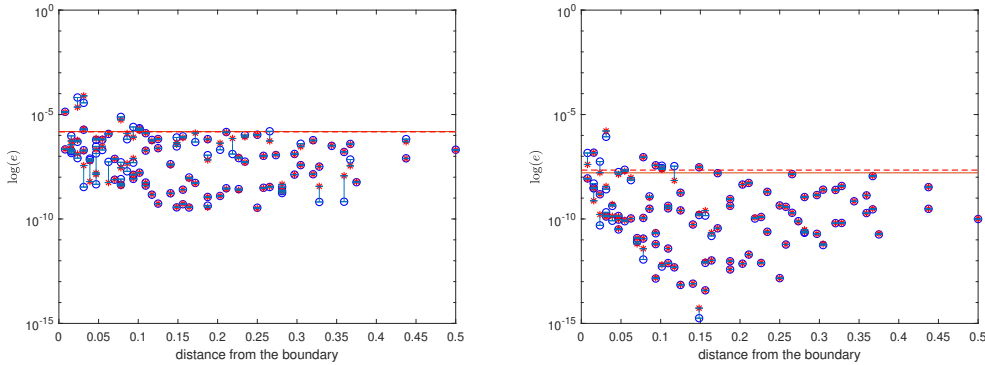
6

Figure 1: Pointwise evaluation errors (circles) and estimates (asterisks), in logarithmic scale, by adaptive moving interpolation of Franke's test function on 100 Sobol test points in $[0, 1]^2$, ordered by distance from the boundary, using 800 (left) and 1600 (right) Halton interpolation points (solid line: mean error; dashed line: mean estimate).

points [2]. This means that the first m_k among m_d Leja points are unisolvent for degree $k = 1, 2, \ldots, d$, a feature that is exploited by the algorithm DISC in [24] to compute a pointwise a posteriori error estimate.

In order to show the performance of moving interpolation, in Figure 1 we report the pointwise error and the error estimate in [24] for the well-known Franke's test function, by moving interpolation on 800 and 1600 Halton points in $[0, 1]^2$. These quantities are computed for 100 Sobol evaluation points, ordered by increasing distance from the boundary. The Sobol points form a low discrepancy quasi-random sequence and are designed to generate a sample that is uniformly distributed over the unit hypercube [60, 3]. Observe that in most cases the estimate is close to the actual error and only in few cases it substantially overestimates or underestimates the error, by one or seldom two orders of magnitude. The resulting mean estimate (dashed line) turns out to be a good approximation of the mean error (solid line); as expected, both decrease by increasing the number of sampling points. Notice that the errors above the mean tend to occur near the boundary. This boundary effect, already observed in [23, 24], can be explained by observing that near the boundary, the neighborhoods $\mathcal{N}_h = B_h(\overline{P}) \cap \Omega$ contain less sampling points. Consequently, the maximum local interpolation degree becomes smaller and at the same time the quality of the Leja-like interpolation points can be worse.

7

2.2. Adaptive LOOCV RBF interpolation

We now provide a brief overview on adaptive LOOCV RBF interpolation recalling basic notions on RBF theory, which is useful for scattered data approximation; for further details, see [34, 70].

Suppose that we are given a compact domain $\Omega \subset \mathbb{R}^s$, a scattered point set $X = \{P_1, \ldots, P_N\} \subset \Omega$, and the corresponding data (or function) value set $F = \{f_1, \ldots, f_N\} \subset \mathbb{R}$ that is obtained by possibly sampling any (unknown) function $f : \Omega \to \mathbb{R}$.

Considering a RBF $\phi : [0, \infty) \to \mathbb{R}$ that is *strictly conditionally positive definite* of order m (SCPD$_m$) [33], if we set $\phi_i(P) = \phi(||P - P_i||_2)$, we can find a unique interpolating function $\Psi : \Omega \to \mathbb{R}$ of the form

$$\Psi(P) = \sum_{i=1}^{N} c_i \phi_i(P) + \sum_{i=N+1}^{N+M} c_i \pi_{i-N}(P), \qquad (9)$$

where $\{\pi_k\}_{k=1}^{M}$ generate a basis for the $M = \binom{m-1+s}{m-1}$-dimensional linear space $\mathbb{P}_{m-1}^s$ of s-variate real valued polynomials of total degree less than or equal to $m - 1$, and $|| \cdot ||_2$ is the Euclidean norm. The coefficients $c_1, \ldots, c_{N+M}$ are determined by enforcing the interpolation conditions $\Psi(P_i) = f_i$, for $i = 1, \ldots, N$. Since these conditions lead to a system of N linear equations in the $N + M$ unknowns c_i, one usually adds the M additional conditions $\sum_{i=1}^{N} c_i \pi_k(P_i) = 0$, $k = 1, \ldots, M$, to ensure a unique solution. Moreover, from theory it is known that a SCPD$_0$ function is *strictly positive definite* (SPD), and so in this case the polynomial in (9) is omitted [33].

Solving the interpolation problem for a SCPD$_m$ function ϕ leads to a symmetric linear system

$$\mathcal{M}\mathbf{c} = \mathbf{y}, \qquad (10)$$

where

$$\mathcal{M} = \begin{bmatrix} A & B \\ B^T & O \end{bmatrix}, \qquad \mathbf{y} = \begin{bmatrix} \mathbf{f} \\ \mathbf{0} \end{bmatrix}.$$

The *interpolation matrix* $\mathcal{M}$ in (10) has entries $A_{ki} = \phi(||P_k - P_i||_2)$, $B_{kj} = \pi_j(P_k)$, with $k, i = 1, \ldots, N$, $j = 1, \ldots, M$, and O is a $M \times M$ zero matrix. Moreover, $\mathbf{c} = [c_1, \ldots, c_{N+M}]^T$, $\mathbf{f} = [f_1, \ldots, f_N]^T$ and $\mathbf{0}$ is a zero vector of length M. Notice that for a SPD function ϕ the matrix reduces simply to $\mathcal{M} = A$, and the polynomial part vanishes.

In literature, many RBFs are scaled by a *shape parameter* $\varepsilon > 0$ such that

$$\phi_\varepsilon(r) = \phi(\varepsilon r) = \phi(\varepsilon ||P - P_i||_2), \quad \forall P, P_i \in \Omega.$$

Hereinafter, for the sake of simplicity, we keep implicit (unless strictly necessary) the dependence on ε, referring to $\phi_i(P)$ as $\phi_\varepsilon(r)$. For such RBFs the choice of a "good", or possibly an "optimal", shape parameter is a crucial task, but also a big issue (see e.g. [10] and references therein). Some examples of popular SCPD RBFs together with their smoothness and abbreviation are listed as follows [33]:

$$
\phi_\varepsilon(r) = \begin{cases}
\exp(-\varepsilon^2 r^2), & \text{Gaussian } C^\infty(\text{GA}) \\
(1 + \varepsilon^2 r^2)^{-1/2}, & \text{Inverse MultiQuadric } C^\infty(\text{IMQ}) \\
(1 + \varepsilon^2 r^2)^{1/2}, & \text{MultiQuadric } C^\infty(\text{MQ}) \\
\max(1 - \varepsilon r, 0)^4 (4\varepsilon r + 1), & \text{Wendland } C^2(\text{W2}).
\end{cases}
$$

All these functions are SPD, except for MQ that is SCPD$_1$. GA, IMQ and MQ are globally supported, while W2 is compactly supported and its support is $[0, 1/\varepsilon]$, see [70].

However, as remarked in [34, Ch.11], from a correct reformulation of the so-called *uncertainty* or *trade-off principle* due to Schaback [58] we know that using a standard basis one cannot have high accuracy and stability at the same time. Indeed, when interpolating target functions by RBFs, the best accuracy is typically achieved in the *flat limit* $\varepsilon \to 0$, but in this case the interpolation matrix might be severely ill-conditioning. Thus, in order to get trustworthy results, in the literature several techniques have been proposed to guide us in a suitable selection of the RBF shape parameter (see e.g. [34, Ch.14] and [41, 59]).

A popular strategy for estimating the RBF shape parameter ε is the LOOCV method. It was originally introduced in RBF interpolation by Rippa [57], and more lately has widely been used and extended in many other fields (see e.g. [5, 6, 49]).

In the LOOCV technique the shape parameter is selected by minimizing a cost function that collects the errors for a sequence of partial fits to the data [10]. To estimate the unknown true error, we split the data into two parts: a *training* data set consisting of $N - 1$ data to obtain a "partial interpolation", and a *validation* data set that contains a single (remaining) data used to compute the error. After repeating in turn this procedure for each of the N given data, the result is a vector of error estimates and the cost function is used to optimally determine the value of ε.

For this discussion, we define by $X^{[k]} = \{P_1, \ldots, P_{k-1}, P_{k+1}, \ldots, P_N\}$,

and $F^{[k]} = \{f_1, \ldots, f_{k-1}, f_{k+1}, \ldots, f_N\}$, the sets of points and corresponding values with the removed data (P_k, f_k), denoted by the superscript $[k]$. All other quantities are represented similarly.

The key idea of LOOCV is to predict the parameter ε by the partial RBF interpolant to the data $(X^{[k]}, F^{[k]})$, i.e.,

$$\Psi^{[k]}(P) = \sum_{i=1,\ i\neq k}^{N} c_i^{[k]}\phi_i(P) + \sum_{i=N+1}^{N+M} c_i^{[k]}\pi_{i-N}(P). \tag{11}$$

The interpolant (11) enables us to find the error $e_k(\varepsilon) = f_k - \Psi^{[k]}(P_k)$, $k = 1, \ldots, N$, removing in turn each point P_k and comparing then the resulting fit with the known value at the removed point f_k. This error can be computed more efficiently by simply using the rule [57]

$$e_k(\varepsilon) = \frac{c_k}{\mathcal{M}_{kk}^{-1}}, \tag{12}$$

where c_k is the kth coefficient deriving from the solution of the *full* RBF system (10), and $\mathcal{M}_{kk}^{-1}$ is the kth diagonal element of the matrix $\mathcal{M}^{-1}$. This formulation needs to only solve a single linear system, considering the entire data set (X, F).

In summary, the problem is solved by minimizing the LOOCV cost function [10]

$$\text{LOOCV}(\varepsilon) = ||\mathbf{e}(\varepsilon)||_\infty = \max_{k=1,\ldots,N} \left| \frac{c_k}{\mathcal{M}_{kk}^{-1}} \right|,$$

where $\mathbf{e}(\varepsilon) = [e_k(\varepsilon)]_{k=1,\ldots,N}$ and $||\cdot||_\infty$ denotes the ∞-norm, even if in principle any norm is enabled.

2.3. BLOOCV RBF-PUM

We now introduce the basic theory for an adaptive RBF-PUM interpolation, which is based on the BLOOCV scheme [11]. Since inverting the kernel matrix in (10) is computationally expensive when the amount of data highly increases, an effective way to overcome this issue is to partition the open bounded domain Ω into p overlapping subdomains $\{\Omega_j\}_{j=1}^p$ which form an open bounded cover of Ω, that is $\Omega \subseteq \bigcup_{j=1}^p \Omega_j$. So the interpolation problem is locally decomposed into p (smaller) subproblems, where the subdomains of the PU covering consist of overlapping balls of radius δ_j, see [4].

The idea of PUM comes originally from the context of PDEs [1], then gaining popularity in the RBF community in various fields of applied mathematics (see e.g. [7, 47, 69]). The PUM has indeed some features that makes it particularly suitable for the approximation of large and irregularly distributed scattered data sets. In particular, it enables us to efficiently solve a local interpolation problem on each PU subdomain and thus to construct the global approximant by gluing together the local contributions using some weight functions. To achieve that, we need those weights to be a family of compactly supported, non-negative, and continuous functions $w_j : \Omega_j \to \mathbb{R}$, with $\mathrm{supp}(w_j) \subseteq \Omega_j$, such that

$$\sum_{j=1}^{p} w_j(P) = 1, \quad P \in \Omega.$$

Once we choose the partition of unity $\{w_j\}_{j=1}^p$, we form the global interpolant

$$\Psi(P) = \sum_{j=1}^{p} \Psi_j(P) w_j(P), \quad P \in \Omega, \tag{13}$$

which is a weighted sum of p local RBF interpolants $\Psi_j : \Omega_j \to \mathbb{R}$, i.e.

$$\Psi_j(P) = \sum_{i=1}^{N_j} c_i^j \phi_i^j(P), \tag{14}$$

where $\phi_i^j(P) = \phi_{\varepsilon_j}(||P - P_i^j||_2)$, $P_i^j \in X_j = X \cap \Omega_j$, and $N_j = |\Omega_j|$. The interpolant (14) represents the local version of (9) in the special case of SPD RBFs, and it is uniquely found by solving the linear system associated with the subdomain Ω_j, i.e.,

$$A_j \mathbf{c}_j = \mathbf{f}_j,$$

where $(A_j)_{ki} = \phi_{\varepsilon_j}(||P_k^j - P_i^j||_2)$, $k, i = 1, \ldots, N_j$, $\mathbf{c}_j = [c_1^j, \ldots, c_{N_j}^j]^T$, $\mathbf{f}_j = [f_1^j, \ldots, f_{N_j}^j]^T$; for a generalization to SCPD_m functions, see [69]. Moreover, from [11] it is known that the accuracy of the global fit (13) strongly depends on the choices of the shape parameter ε_j and the radius δ_j.

The LOOCV technique discussed in Subsection 2.2 can thus be extended to search the optimal values of ε_j and δ_j in each PU subdomain. It consists in computing in every subdomain Ω_j, for each couple of (ε, δ) and

$k \in \{1, \dots, N_j\}$, the interpolation error $e_k^j(\varepsilon, \delta)$ of the partial RBF interpolant $\Psi_j^{[k]}$ evaluated at the point P_k^j, fitted on the point set $X_j^{[k]} = X_j \setminus \{P_k^j\}$ and the data value set $F_j^{[k]} = F_j \setminus \{f_k^j\}$. To avoid this computation, by suitably adapting (12), the local error is defined as follows:

$$e_k^j(\varepsilon, \delta) = \frac{c_k^j}{(A_j^{-1})_{kk}}.$$

Hence, the optimal value of $(\varepsilon_j, \delta_j)$ is the one that minimizes the error function [10]

$$\text{LOOCV}_j(\varepsilon, \delta) = \max_{k=1,\dots,N_j} \left| \frac{c_k^j}{(A_j^{-1})_{kk}} \right|.$$

2.4. Multinode Shepard method

The Multinode Shepard (MS) method is an accurate procedure for reconstructing functions from scattered data. It is the generalization of the Little's idea that consists in improving the classic Shepard method by acting in two ways: modifying the classical point-based weight functions and defining instead a normalized blend of locally linear interpolating polynomials with triangle-based weight functions which depend on the product of inverse distances to the three vertices of the corresponding triangle [20]. We assume that $\Omega \subset \mathbb{R}^s$, $s \geq 2$, is a region, $X = \{P_i\}_{i=1}^N \subset \Omega$ is a finite set of pairwise distinct scattered points and $F = \{f_i\}_{i=1}^N$ is a set of associated real numbers, obtained by sampling a continuous function $f : \Omega \to \mathbb{R}$. By working with local interpolating polynomials of total degree $d \in \mathbb{N}$ of s variables, we let $m_d = \binom{s+d}{d}$ the dimension of such a polynomial space. To define the multinode Shepard method, we need the existence of a set $\mathcal{S} = \{\sigma_j\}_{j=1}^S \subset X$, which is a covering of the set X, i.e.

$$\bigcup_{j=1}^S \sigma_j = X,$$

and such that each $\sigma_j = \{P_{j_1}, \dots, P_{j_{m_d}}\}$ is unisolvent for polynomial interpolation of total degree d. As proven in [28], such a set $\mathcal{S}$ exists for almost all choices of the interpolation nodes. The local interpolation polynomial $\pi_j(P)$ based on $\sigma_j = \{P_{j_1}, \dots, P_{j_{m_d}}\}$ is represented as

$$\pi_j(P) = \sum_{i=1}^{m_d} \ell_{j,i}(P) f_{j_i}, \quad P \in \mathbb{R}^s,$$

12

where

$$\ell_{j,i}(P) = \sum_{\substack{\alpha \in \mathbb{N}_0^s \\ |\alpha| \leq d}} a_\alpha^{(i)} \left(P - P_j^{(b)}\right)^\alpha, \quad P \in \mathbb{R}^s,$$

are the Lagrange basis polynomials satisfying

$$\ell_{j,i}(P_{j_k}) = \delta_{ik} = \left\{ \begin{array}{ll} 1, & i = k, \\ 0, & i \neq k, \end{array} \right.$$

written, using a multi-index notation, in the Taylor basis centered at the barycenter $P_j^{(b)}$ of σ_j [22]. Similarly to the triangular [20] ($s = 2$, $d = 1$), the hexagonal [19] ($s = 2$, $d = 2$) and the tetrahedral Shepard methods [8] ($s = 3$, $d = 1$), the MS method is obtained by combining the polynomials $\ell_{j,i}(P)$, $i = 1, \ldots, N$, $j = 1, \ldots, S$ with the MS basis function [18]

$$W_{\mu,j}(P) = \frac{\prod_{l=1}^{m_d} \|P - P_{j_l}\|_2^{-\mu}}{\sum_{k=1}^{S} \prod_{l=1}^{m_d} \|P - P_{k_l}\|_2^{-\mu}}, \quad \mu > 0, \ P \in \mathbb{R}^s, \tag{15}$$

as follows

$$\Psi_\mu(P) = \sum_{i=1}^{N} \sum_{j \in \mathcal{J}_i} W_{\mu,j}(P) \ell_{j,i}(P) f_i, \tag{16}$$

where $\mathcal{J}_i = \{j \in \{1, \ldots, S\} : P_i \in \sigma_j\}$. The functions $W_{\mu,j}(P)$ are a partition of unity and the MS approximant interpolates on all the scattered points P_i, $i = 1, \ldots, N$ and reproduces polynomial of degree less than or equal to d.

Results on the approximation order of the MS method can be found in [18]. By denoting with $R_r(Q) = \{P \in \mathbb{R}^s : \|P - Q\|_\infty \leq r\}$ the axis-aligned closed cube with center Q and edge length $2r$, we set

$$h' = \inf\{r > 0 : \forall P \in \Omega, \ \exists \sigma_j \in \mathcal{S} : R_r(P) \cap \sigma_j \neq \emptyset\}, \tag{17}$$

$$h'' = \inf\{r > 0 : \forall \sigma_j \in \mathcal{S}, \ \exists P \in \Omega : \sigma_j \subset R_r(P)\}, \tag{18}$$

and finally

$$h = \max\{h', h''\}. \tag{19}$$

The positive real number h is clearly a measure of the fill distance of the points in X and of the largeness of subsets' σ_j diameters: h decreases if the

number of a rather uniform distribution of scattered points increases and the diameters of the subsets σ_j remain relatively small. We further let

$$M = \sup_{P \in \Omega} \#\{\sigma_j \in \mathcal{S} : R_h(P) \cap \sigma_j \neq \emptyset\}, \qquad (20)$$

the maximum number of subsets σ_j with at least one point in some cube with edge length $2h$. Small values of M, in correspondence of small values of h, imply that there are no clusters of subsets of the covering $\mathcal{S}$. Under the hypothesis of compactness and convexity of the region Ω, if $\mu > \frac{s+d+1}{m_d}$, in [18] it is proven that

$$||f - \Psi_\mu (P) ||_\infty \leq CM||f||_{d,1} h^{d+1},$$

where C is a positive constant and

$$||f||_{d,1} = \sup \left\{ \frac{|D^\nu f(P) - D^\nu f(Q)|}{||P - Q||_2}, \ P, Q \in \Omega, \ \nu \in \mathbb{N}_0^s, \ |\nu| = d \right\},$$

provided that f is differentiable with partial derivatives of order d Lipschitz-continuous. The constant C is explicitly computed; it is directly proportional to $\left(1 + \max_{j=1,\dots,S} ||\pi_j||_\infty \right)$ by a factor which depends only on s, S, d, m_d and μ. To maintain $\max_{j=1,\dots,S} ||\pi_j||_\infty$ small [21], a convenient procedure consists in considering, for each scattered point, the set of $m_d + q$, $q > 0$, nearest points and in choosing, among them, the subset of m_d discrete Leja points by the procedure introduced in [2] detailed in section 2.1. Finally, the result on the order of convergence holds with a slightly different constant in the case of non-convex domains (even multiply connected domains) by using the "Whitney regularity" property, in line with what is observed in section 2.1 (see also [32]).

3. Cubature tests

In this section we compare cubature by adaptive scattered interpolation via suitable PI algebraic rules as in (1)-(2), with two relevant methods proposed in the recent literature, namely RBFCUB in [12] (based directly on LOOCV optimized RBF integration over polygons) and Glaubitz's LS-CF in [38] (based on ℓ^2 weight minimization under polynomial moment matching conditions). To our knowledge, this is the first systematic comparison of
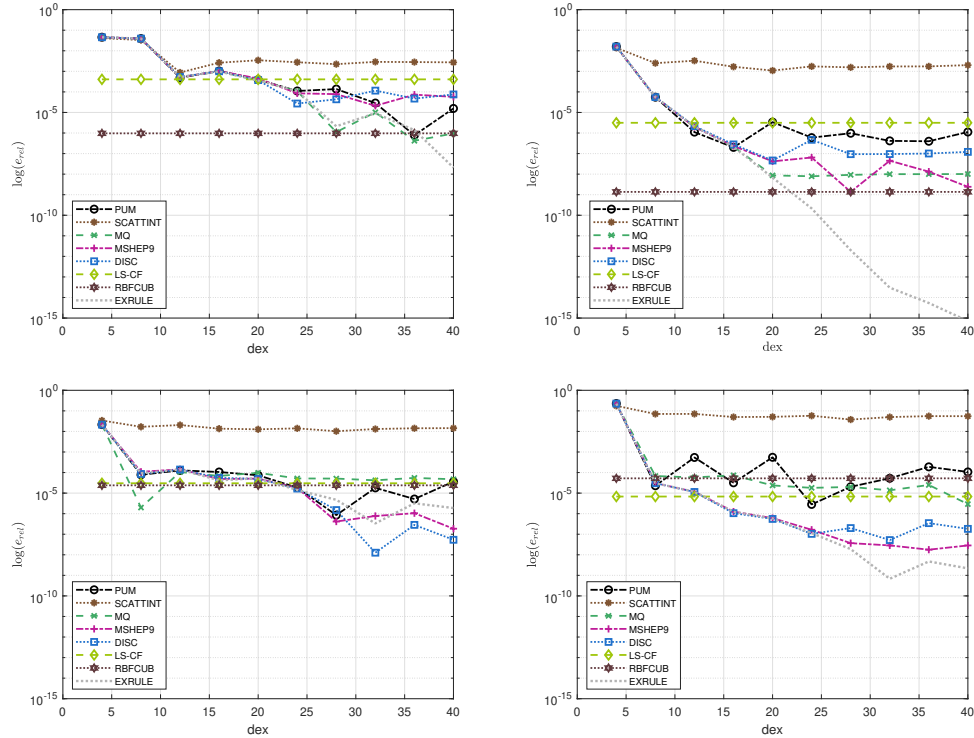
14

Figure 2: Comparison of scattered cubature methods on 400 Halton points for the test functions f_1 (top left) and f_2 (top right), f_3 (bottom left) and f_4 (bottom right); the horizontal lines correspond to RBFCUB with MultiQuadrics and to LS-CF method, whereas the grey dotted line to the underlying algebraic rule with exact function values. The x-axis reports the exactness degree (dex) of the underlying (almost-)minimal algebraic rule and the y-axis the relative cubature errors in the logarithmic scale ($\log(e_{rel})$).

different approaches for numerical cubature by scattered data, in particular concerning how much accuracy is obtainable from the data. All the numerical tests have been performed by the Matlab package SCATTCUB, freely available along with the corresponding demos at [9].

In Figures 2-3 we plot the relative cubature errors versus the exactness degree of the underlying (almost-)minimal algebraic rule for the square (such rules have been collected from the literature and inserted in the package [9]). We consider two smooth instances, namely the popular Franke's test function in $\Omega = [0,1]^2$

$$f_1 = \frac{3}{4}\exp(-((9x-2)^2+(9y-2)^2)/4)+\frac{3}{4}\exp(-((9x+1)^2/49+(9y+1)/10))$$
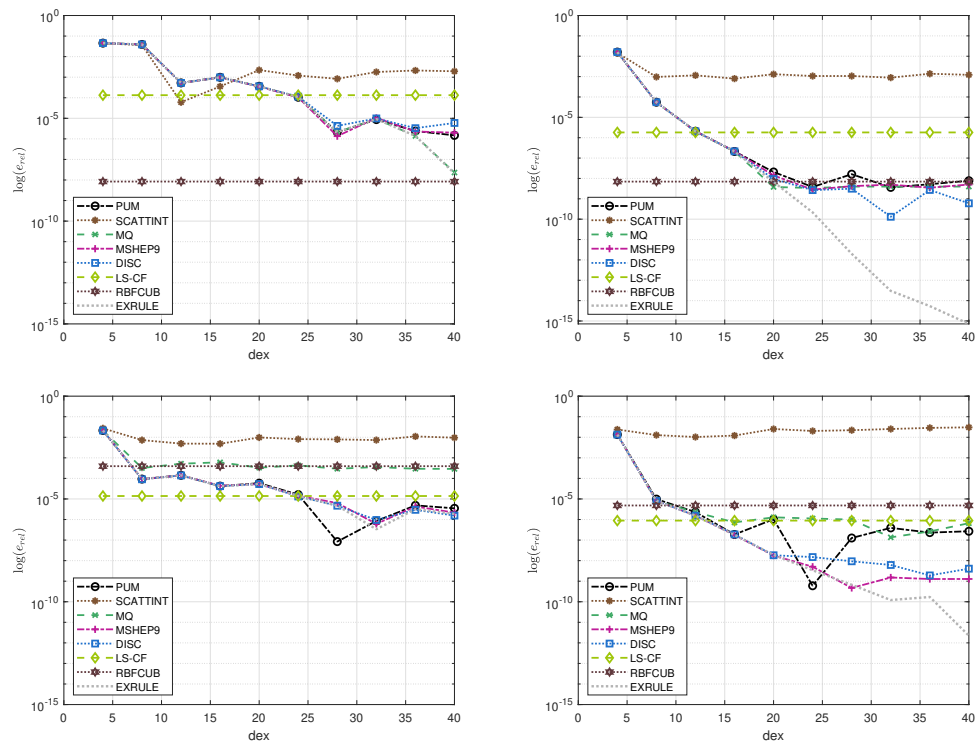
15

Figure 3: As in Figure 2 on 800 Halton points. The x-axis reports the exactness degree (dex) of the underlying (almost-)minimal algebraic rule and the y-axis the relative cubature errors in the logarithmic scale ($\log(e_{rel})$).
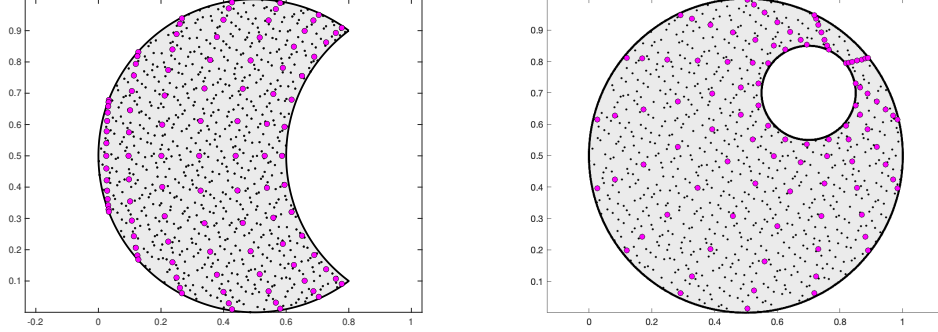
16

Figure 4: Halton points on a circular lune and an asymmetric circular annulus (black dots), and cubature nodes of an algebraic rule of degree 12 (in magenta).

$$+\frac{1}{2}\exp(-((9x-7)^2+(9y-3)^2)/4)-\frac{1}{5}\exp(-((9x-4)^2+(9y-7)^2)),$$

and a test function in $\Omega = [-1,1]^2$ from [38]

$$f_2 = \frac{1}{(1+x^2)(1+y^2)}\ ,$$

together with two power functions of finite regularity in $\Omega = [0,1]^2$, namely

$$f_3 = ((x-x_0)^2+(y-y_0)^2)^{3/2}\ ,\quad f_4 = ((x-x_0)^2+(y-y_0)^2)^{7/2},$$

with $(x_0, y_0) = (0.5, 0.5)$, where $f_3 \in C^2(\Omega)$ and $f_4 \in C^6(\Omega)$. The reference values of the integral have been computed by the Matlab built-in adaptive routine `integral2`.

The compared methods are Matlab's basic Scattered Interpolation (SCAT-TINT), adaptive moving interpolation (DISC, §2.1), adaptive LOOCV RBF interpolation with MultiQuadrics (MQ, §2.2), BLOOCV RBF-PUM (PUM, §2.3), Multinode Shepard method with local interpolation degree 9 (MSHEP9, §2.4). The horizontal lines correspond to RBFCUB with MultiQuadrics [12] and to LS-CF method [38], whereas the grey dotted line to the underlying algebraic rule with exact function values. The choice of MultiQuadrics comes from a number of numerical tests, not reported for brevity, where they have shown to be more accurate than the other RBF considered (Gaussians, IMQ and Wendland C^2).

We see that adaptive moving interpolation and Multinode Shepard follow closely the underlying algebraic rule, until the error stalls around the error
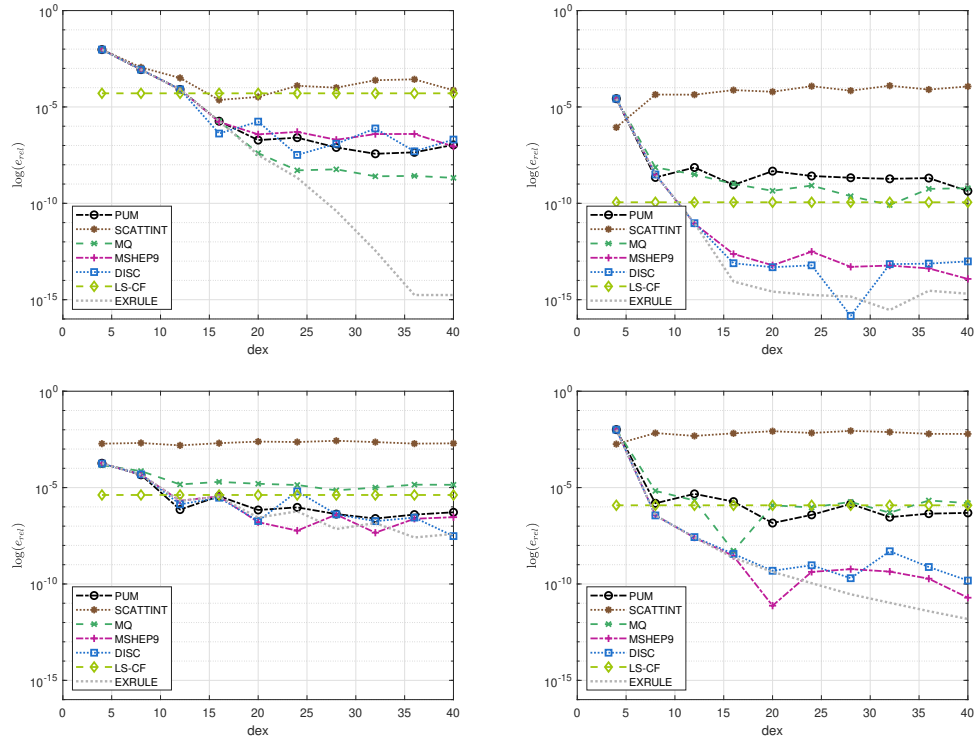
Figure 5: As in Figure 2 on Halton points of the circular lune of Figure 4, extracted from 800 Halton points of the bounding box. The x-axis reports the exactness degree (dex) of the underlying (almost-)minimal algebraic rule and the y-axis the relative cubature errors in the logarithmic scale $(\log(e_{rel}))$.
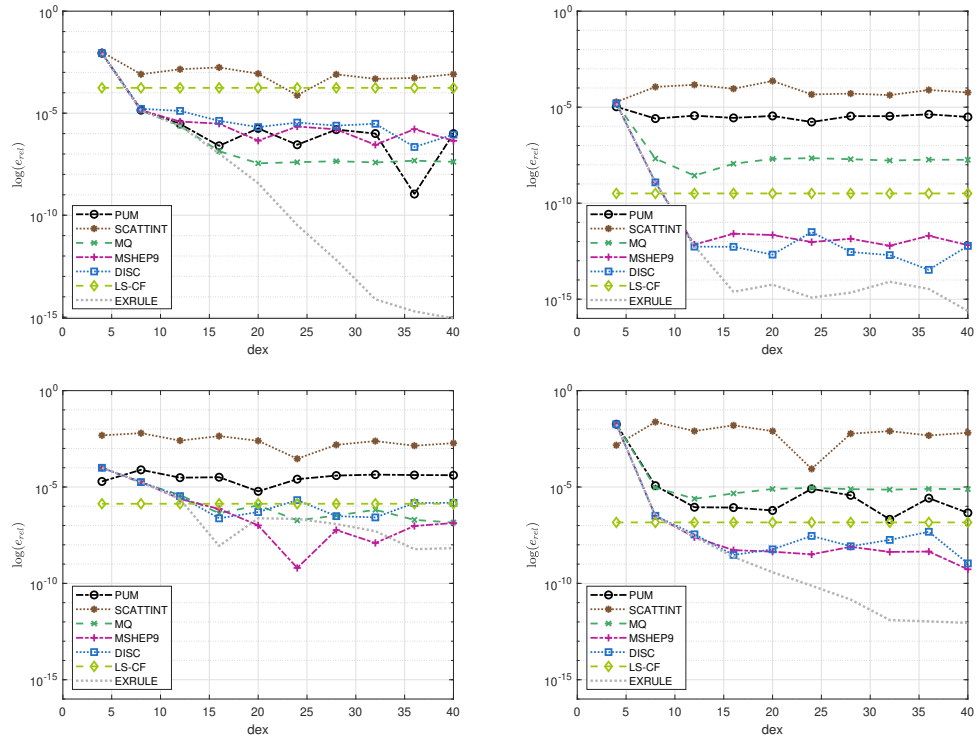
18

Figure 6: As in Figure 2 on Halton points of the asymmetric circular annulus of Figure 4, extracted from 800 Halton points of the bounding box. The x-axis reports the exactness degree (dex) of the underlying (almost-)minimal algebraic rule and the y-axis the relative cubature errors in the logarithmic scale ($\log(e_{rel})$).

generated by interpolation. It is interesting to observe that higher interpolation errors at the boundary (see Figure 1) are there compensated by smaller algebraic cubature weights, preserving the overall cubature accuracy.

Both adaptive moving interpolation and Multinode Shepard at error stalling turn out to be more accurate than Glaubitz's LS-CF method, especially with regular integrands where the error goes up to two-three orders of magnitude below. In the comparison with RBFCUB, the situation is similar with the less regular integrands f_3 and f_4, whereas RBFCUB gives quite accurate results with the smooth integrands f_1 and f_2. Also PUM method is able to follow on average the underlying algebraic rule until stalling, whereas LOOCV MQ turns out to be quite accurate with the smooth integrands and less accurate, similarly to RBFCUB, with the less regular integrands.

In order to show the flexibility of the methods, we give also integration examples on two nonstandard domains with curved boundaries, namely a circular lune (non-convex domain) and an asymmetric circular annulus (multiply connected domain), see Figure 4, using the algebraic cubature formulas developed in [15, 16] via subperiodic trigonometric Gaussian quadrature. In these cases RBFCUB is not applicable, being restricted to linear polygons. On the other hand, we have modified the code LS-CF, to work with the appropriate polynomial moments on such curved domains (this option is not present in the original package [39]). Concerning adaptive moving polynomial interpolation by the code DISC, since the domains are nonconvex one should check that they are Whitney regular (cf. §2.1). Indeed, depending on their position inside the lune (or the annulus), two points can be joined either by the corresponding segment if it does not intersect the lune's shorter side (or the internal disk of the annulus), or by the curve formed by two pieces of the segment joined by the (shortest) subarc between the intersection points. The length of such a curve is in any case bounded by $\pi/2$ times the length of the segment, thus the domains are Whitney regular.

Similar considerations with respect to square domains can be done, with adaptive moving interpolation and Multinode Shepard confirming a better accuracy compared to LS-CF. It should be stressed, as already observed, that we have here privileged maximal accuracy obtainable from not a huge number of fixed data, rather than computational efficiency. Indeed, in the presence of a very high number of scattered points, PUM and LS-CF are faster and could be methods of choice in order to control the computing time.

Conclusion and future work. We have constructed cubature methods on scattered data via resampling on the support of known algebraic cubature formulas, by adaptive interpolation of different kinds, such as moving polynomial interpolation at Leja-like points, LOOCV RBF, Multinode Shepard and BLOOCV RBF-PUM. These methods are compared with the most recent approaches to scattered cubature in the literature, namely LOOCV optimized RBF integration over polygons and Glaubitz's LS-CF (based on ℓ^2 weight minimization under polynomial moment matching conditions).

To our knowledge, this is the first systematic comparison of different approaches for numerical cubature by scattered data, in particular concerning how much accuracy is obtainable from the data. The numerical results confirm that adaptive interpolation applied to known Positive-Interior algebraic cubature formulas can provide valid alternatives to the existing methods, with higher flexibility concerning the shape of integration domains. Extension to three-dimensional integration problems appears natural and will be object of future research.

References

[1] I. Babuška, J.M. Melenk, The partition of unity method, Int. J. Numer. Methods Eng., 40 (1997), 727–758.

[2] L. Bos, S. De Marchi, A. Sommariva, M. Vianello, Computing multivari-

ate Fekete and Leja points by numerical linear algebra, SIAM J. Numer. Anal., 48 (2010), 1984–1999.

[3] P. Bratley, B.L. Fox, Algorithm 659: Implementing Sobol's quasirandom sequence generator, ACM Trans. Math. Softw., 14 (1988), 88–100.

[4] R. Cavoretto, Adaptive radial basis function partition of unity interpolation: A bivariate algorithm for unstructured data, J. Sci. Comput., 87 (2021), 41.

[5] R. Cavoretto, A. De Rossi, A two-stage adaptive scheme based on RBF collocation for solving elliptic PDEs, Comput. Math. Appl., 79 (2020), 3206–3222.

[6] R. Cavoretto, A. De Rossi, An adaptive LOOCV-based refinement scheme for RBF collocation methods over irregular domains, Appl. Math. Lett., 103 (2020), 106178.

[7] R. Cavoretto, A. De Rossi, Spherical interpolation using the partition of unity method: an efficient and flexible algorithm, Appl. Math. Lett., 25 (2012), 1251–1256.

[8] R. Cavoretto, A. De Rossi, F. Dell'Accio, F. Di Tommaso, An Efficient Trivariate Algorithm for Tetrahedral Shepard Interpolation, J. Sci. Comput., 82 (2020), 1–15.

[9] R. Cavoretto, A. De Rossi, F. Dell'Accio, F. Di Tommaso, N. Siar, A. Sommariva, M. Vianello, SCATTCUB: Matlab codes for numerical cubature by adaptive interpolation on bivariate scattered data, available online at https://www.math.unipd.it/~alvise/software.html.

[10] R. Cavoretto, A. De Rossi, M.S. Mukhametzhanov, Y.D. Sergeyev, On the search of the shape parameter in radial basis functions using univariate global optimization methods, J. Glob. Optim., 79 (2021), 305–327.

[11] R. Cavoretto, A. De Rossi, E. Perracchione, Optimal selection of local approximants in RBF-PU interpolation, J. Sci. Comput., 74 (2018), 1–22.

[12] R. Cavoretto, A. De Rossi, A. Sommariva, M. Vianello, RBFCUB: a numerical package for near-optimal meshless cubature on general polygons,

Appl. Math. Lett., 125 (2022), 107704. `https://www.math.unipd.it/~alvise/software.html`.

[13] R. Cools, Constructing cubature formulae: the science behind the art, Acta Numer. 6 (1997), published online by Cambridge University Press: 07 November 2008, 1–54.

[14] R. Cools, K.J. Kim, A survey of known and new cubature formulas for the unit disk, Korean J. Appl. Math. Comput., 7 (2000), 477–485.

[15] G. Da Fies, A. Sommariva, M. Vianello, Algebraic cubature by linear blending of elliptical arcs, Appl. Numer. Math., 74 (2013), 49–61.

[16] G. Da Fies, M. Vianello, Product Gaussian quadrature on circular lunes, Numer. Math. Theory Methods Appl., 7 (2014), 251–264.

[17] P.J. Davis, Interpolation and Approximation. Dover Publications, 1975.

[18] F. Dell'Accio, F. Di Tommaso, Rate of convergence of multinode Shepard operators, Dolomites Res. Notes Approx., 12 (2019), 1-6.

[19] F. Dell'Accio, F. Di Tommaso, On the hexagonal Shepard method, Appl. Numer. Math., 150 (2020), 51-64.

[20] F. Dell'Accio, F. Di Tommaso, K. Hormann, On the approximation order of triangular Shepard interpolation, IMA J. Numer. Anal., 36 (2016), 359–379.

[21] F. Dell'Accio, F. Di Tommaso, O. Nouisser, N. Siar, Solving Poisson equation with Dirichlet conditions through multinode Shepard operators, Comput. Math. Appl., 98 (2021), 254–260.

[22] F. Dell'Accio, F. Di Tommaso, N. Siar, On the numerical computation of bivariate Lagrange polynomials, Appl. Math. Lett., 112 (2021), 106845.

[23] F. Dell'Accio, F. Di Tommaso, N. Siar, M. Vianello, Numerical differentiation on scattered data through multivariate polynomial interpolation, BIT Numer. Math., 62 (2022), 773–801.

[24] F. Dell'Accio, F. Di Tommaso, N. Siar, M. Vianello, DISC: an adaptive numerical Differentiator by local polynomial Interpolation on SCattered data, Dolomites Res. Notes Approx. DRNA 15 (2022) ("Special Issue Software for Approximation 2022", 81–91).

[25] F. Dell'Accio, F. Di Tommaso, F. Nudo, Constrained mock-Chebyshev least squares quadrature. Appl. Math. Lett., 134 (2022), 108328.

[26] F. Dell'Accio, F. Di Tommaso, E. Francomano, F. Nudo, An adaptive algorithm for determining the optimal degree of regression in constrained mock-Chebyshev least squares quadrature. Dolomites Res. Notes Approx., 15 (2022), 35–44.

[27] F. Dell'Accio, D. Mezzanotte, F. Nudo, D. Occorsio, Product integration rules by the constrained mock-Chebyshev least squares operator. BIT Numer. Math., 63 (2023), 24.

[28] F. Dell'Accio, A. Sommariva, M. Vianello, Random sampling and unisolvent interpolation by almost everywhere analytic functions, Appl. Math. Lett., 145 (2023), 108734.

[29] M. Dell'Orto, M. Dessole, F. Marcuzzi, The Lawson-Hanson Algorithm with Deviation Maximization: Finite Convergence and Sparse Recovery, Numer. Linear Algebra Appl. 30 (2023), e2490.

[30] M. Dessole, F. Marcuzzi, M. Vianello, Accelerating the Lawson-Hanson NNLS solver for large-scale Tchakaloff regression designs, Dolomites Res. Notes Approx. DRNA 13 (2020), 20–29.

[31] J.Dick, F.Y. Kuo, I.H. Sloan, High-dimensional integration: The quasi-Monte Carlo way, Acta Numer., 22 (2013), 133–288.

[32] R. Farwig, Rate of convergence of Shepard's global interpolation formula, Math. Comput., 46 (1986), 577–590.

[33] G.E. Fasshauer, Meshfree Approximation Methods with MATLAB, Interdiscip. Math. Sci., World Scientific Publishing Co., Singapore, 6 (2007).

[34] G.E. Fasshauer, M.J. McCourt, Kernel-based Approximation Methods using MATLAB, Interdiscip. Math. Sci., World Scientific Publishing Co., Singapore, 19 (2015).

[35] M. Festa, A. Sommariva, Computing almost minimal formulas on the square, J. Comput. Appl. Math., 236 (2012), 4296–4302.

[36] B. Fornberg, N. Flyer, Solving PDEs with radial basis functions. Acta Numer., 24 (2015), 215–258.

[37] W. Freeden, M. Gutting, Integration and cubature methods: A geomathematically oriented course, Monographs and Research Notes in Mathematics, CRC Press, Boca Raton, FL, 2018.

[38] J. Glaubitz, Stable high-order cubature formulas for experimental data, J. Comput. Phys., 447 (2021), 110693.

[39] J. Glaubitz, stableCFs, 2021, available online at `https://github.com/jglaubitz/stableCFs` (MATLAB code).

[40] J. Glaubitz, J. Reeger, Towards stability of radial basis function based cubature formulas, BIT Numer. Math., 63 (2023).

[41] A. Golbabai, E. Mohebianfar, H. Rabiei, On the new variable shape parameter strategies for radial basis functions. Comput. Appl. Math., 34 (2015), 691–704.

[42] A. Guessab, G. Schmeisser, Construction of positive definite cubature formulae and approximation of functions via Voronoi tessellations. Adv. Comput. Math., 32 (2010), 25–41.

[43] A. Guessab, G. Schmeisser, Negative definite cubature formulae, extremality and Delaunay triangulation. Constr. Approx., 31 (2010), 95–113.

[44] J. Jaśkowiec, N. Sukumar, High-order cubature rules for tetrahedra, Int. J. Numer. Methods Eng., 121 (2020), 2418–2436.

[45] V. Keshavarzzadeh, R.M. Kirby, A. Narayan, Numerical Integration in Multiple Dimensions with Designed Quadrature, SIAM J. Sci. Comput., 40 (2018), A2033–A2061.

[46] G. Klein, J.P. Berrut, Linear barycentric rational quadrature. BIT Numer. Math., 52 (2012), 407–424.

[47] E. Larsson, V. Shcherbakov, A. Heryudono, A least squares radial basis function partition of unity method for solving PDEs, SIAM J. Sci. Comput., 39 (2017), A2538–A2563.

[48] G. Legrain, Non-negative moment fitting quadrature rules for fictitious domain methods, Comput. Math. Appl., 99 (2021), 270–291.

[49] L. Ling, F. Marchetti, A stochastic extended Rippa's algorithm for LpOCV, Appl. Math. Lett., 129 (2022), 107955.

[50] H. Majidian, Creating stable quadrature rules with preassigned points by interpolation, Calcolo 53 (2016), 217–226.

[51] B.M. Makarov, A.N. Podkorytov, Smooth Functions and Maps, Springer, 2020.

[52] F. Piazzon, A. Sommariva, M. Vianello, Caratheodory-Tchakaloff Subsampling, Dolomites Res. Notes Approx., DRNA 13 (2017), 5–14.

[53] W. Pleśniak, Multivariate Jackson Inequality, J. Comput. Appl. Math., 233 (2009), 815–820.

[54] J.A. Reeger, B. Fornberg, Numerical quadrature over smooth surfaces with boundaries. J. Comput. Phys., 355 (2018), 176–190.

[55] J.A. Reeger, B. Fornberg, Numerical quadrature over the surface of a sphere. Stud. Appl. Math., 137(2016), 174–188.

[56] J.A. Reeger, B. Fornberg, M.L. Watts, Numerical quadrature over smooth, closed surfaces. Proc. R. Soc. A: Math. Phys. Eng. Sci., 472 (2016), 20160401.

[57] S. Rippa, An algorithm for selecting a good value for the parameter c in radial basis function interpolation, Adv. Comput. Math., 11 (1999), 193–210.

[58] R. Schaback, Error estimates and condition numbers for radial basis function interpolation, Adv. Comput. Math., 3 (1995), 251–264.

[59] M. Scheuerer, An alternative procedure for selecting a good value for the parameter c in RBF-interpolation, Adv. Comput. Math., 34 (2011), 105–126.

[60] I.M. Sobol', On the distribution of points in a cube and the approximate evaluation of integrals, USSR Comput. Math. Math. Phys., 7 (1967), 86–112.

[61] A. Sommariva, M. Vianello, Numerical cubature on scattered data by radial basis functions, Computing 76 (2006), 295–310.

[62] A. Sommariva, M. Vianello, Meshless cubature by Green's formula, Appl. Math. Comput., 183 (2006), 1098–1107.

[63] A. Sommariva, M. Vianello, RBF moment computation and meshless cubature on general polygonal regions, Appl. Math. Comput., 409 (2021), 126375.

[64] A. Sommariva, M. Vianello, Cubature rules with positive weights on union of disks, Dolomites Res. Notes Approx. DRNA 15 (2022) (Special Issue for the 60th of S. De Marchi, 73–81).

[65] A. Sommariva, M. Vianello, Low-cardinality Positive Interior cubature on NURBS-shaped domains, BIT Numer. Math., 63 (2023), published online 17 March 2023.

[66] A. Sommariva, M. Vianello, TetraFreeQ: tetrahedra-free quadrature on polyhedral elements, Appl. Numer. Math., published online 11 July 2023.

[67] A.H. Stroud, Approximate calculation of multiple integrals, Prentice-Hall, 1971.

[68] N. Takaki, G.W. Forbes, J. P. Rolland, Schemes for cubature over the unit disk found via numerical optimization, J. Comput. Appl. Math., 407 (2022), 114076.

[69] H. Wendland, Fast evaluation of radial basis functions: Methods based on partition of unity, in: C.K. Chui, L.L. Schumaker, J. Stöckler (eds.), Approximation Theory X: Wavelets, Splines, and Applications, Vanderbilt University Press (2002), 473–483.

[70] H. Wendland, Scattered Data Approximation, Cambridge Monogr. Appl. Comput. Math., Cambridge Univ. Press, Cambridge, 17 (2005).

[71] H. Whitney, Functions Differentiable on the Boundaries of Regions, Ann. Math., 35 (1934), 482–485.

[72] H. Whitney, Functions differentiable on the boundaries of regions. Hassler Whitney Collected Papers, (1992), 286–289.