Filomat 31:13 (2017), 4105–4115
https://doi.org/10.2298/FIL1713105S

Published by Faculty of Sciences and Mathematics,
University of Niš, Serbia
Available at: http://www.pmf.ni.ac.rs/filomat

Numerical Quadrature on the Intersection of Planar Disks

Alvise Sommariva[a], Marco Vianello[a]

[a]*Department of Mathematics, University of Padova (Italy)*

Abstract. We provide an algorithm that computes algebraic quadrature formulas with cardinality not exceeding the dimension of the exactness polynomial space, on the intersection of any number of planar disks with arbitrary radius. Applications arise for example in computational optics and in wireless networks analysis. By the inclusion-exclusion principle, we can also compute algebraic formulas for the union of a small number of disks. The algorithm is implemented in Matlab, via subperiodic trigonometric Gaussian quadrature and compression of discrete measures.

1. Introduction

Quadrature problems involving regions determined by a collection of arbitrary planar disks, such as disk intersections, arise in different applied fields: optical design (numerical ray tracing for accurate spot size computation in obscured and vignetted pupils), optical lithography (transmission cross coefficients computation for diffraction under conditions of partial coherence), wireless networks analysis; cf., e.g., [11, 16, 30], [10, 19, 29], and the references therein.

In this paper we present an algorithm for the computation of nodes and weights of algebraic quadrature formulas, i.e. quadrature formulas with a given polynomial degree of exactness (say n), on the *intersection* of m arbitrary planar disks. The key ingredients are *subperiodic trigonometric Gaussian quadrature* and *compression of discrete measures*. By the integral extension of the inclusion-exclusion principle, we can then use the formulas for intersections to construct algebraic formulas for the union of a small number of disks.

Subperiodic trigonometric approximation (approximation by trigonometric polynomials on subintervals of the period), has received some attention in recent years, cf. [3, 5, 6, 8]. The main motivation for studying trigonometric interpolation and quadrature in the subperiodic setting was not related to a direct univariate application, since periodicity plays no role, and there are more natural (e.g. algebraic) approximation methods in the nonperiodic setting.

On the other hand, consider planar or surface regions related to circular arcs, such as circular sectors, lenses, lunes, spherical latitude-longitude rectangles, spherical caps and slices, or even toroidal-poloidal rectangles of the torus. On such regions, algebraic polynomials belong, by suitable geometric transformations, to tensor-product spaces of trigonometric (or of algebraic with trigonometric) univariate polynomials,

2010 *Mathematics Subject Classification*. Primary 65D32
Keywords. Subperiodic trigonometric Gaussian quadrature, bivariate algebraic quadrature, intersection and union of planar disks, compression of discrete measures
Received: 20 February 2016; Accepted: 18 June 2016
Communicated by Gradimir V. Milovanović
Research supported by the ex-60% funds and by the biennial project CPDA143275 of the University of Padova, and by the INdAM GNCS.
Email addresses: alvise@math.unipd.it (Alvise Sommariva), marcov@math.unipd.it (Marco Vianello)

where the subperiodic angular intervals corresponding to the arcs are involved. Indeed, the fundamental observation at the base of these constructions is that: *a multivariate algebraic polynomial restricted to an arc of a circle (more generally, of an ellipse) is a univariate trigonometric polynomial on a subinterval of the period*.

By this point of view, starting from subperiodic trigonometric quadrature, in [5, 8] numerical integration formulas with polynomial exactness have been constructed on different sections of the disk and on domains obtainable by mutually disjoint union of such sections.

For the reader's convenience, we report the main result of [6] concerning *subperiodic trigonometric Gaussian quadrature*, stated here for a general angular interval. Below, we shall denote by $\mathbb{T}_n$ the $2n+1$-dimensional space of univariate trigonometric polynomials of degree not exceeding n, and by $\mathbb{P}_n^d$ the N-dimensional space of d-variate algebraic polynomials with total degree not exceeding n, $N = \binom{n+d}{d}$.

Theorem 1.1. *Let $[\alpha,\beta]$ be an angular interval, with $0 < \beta - \alpha \le 2\pi$. Let $\{(x_j,\lambda_j)\}_{1\le j\le n+1}$, be the nodes and positive weights of the algebraic Gaussian quadrature formula for the weight function $w(x) = 2\sin(\omega/2)(1-\sin^2(\omega/2)\,x^2)^{-1/2}$, $x \in (-1,1)$, $\omega = \frac{\beta-\alpha}{2} \le \pi$. Then*

$$\int_\alpha^\beta t(\theta)\,d\theta = \sum_{j=1}^{n+1} \lambda_j t(\theta_j)\ , \quad \forall t \in \mathbb{T}_n([\alpha,\beta])\ , \tag{1}$$

where $\theta_j = \frac{\alpha+\beta}{2} + 2\arcsin\left(x_j\,\sin\left(\frac{\omega}{2}\right)\right) \in (\alpha,\beta)$, $j = 1,2,\ldots,n+1$.

Observe that, since the weight function is even, the set of angular nodes is symmetric with respect to the center of the interval, and that symmetric nodes have equal weight, cf. [12]. Formula (1) can be effectively implemented in Matlab by Gautschi's suite OPQ (Orthogonal Polynomials and Quadrature) [12], via the modified Chebyshev algorithm (cf., e.g., [21]); see [6] for a full description of the method.

Consider now the general family of domains obtained by *linear blending of elliptical arcs*. Let

$$\boldsymbol{\gamma}_1(\theta) = \boldsymbol{a}_1\cos(\theta) + \boldsymbol{b}_1\sin(\theta) + \boldsymbol{c}_1\ , \quad \boldsymbol{\gamma}_2(\theta) = \boldsymbol{a}_2\cos(\theta) + \boldsymbol{b}_2\sin(\theta) + \boldsymbol{c}_2\ , \tag{2}$$

$\theta \in [\alpha,\beta]$, be two trigonometric planar curves of degree one, $\boldsymbol{a}_i = (a_{i1},a_{i2})$, $\boldsymbol{b}_i = (b_{i1},b_{i2})$, $\boldsymbol{c}_i = (c_{i1},c_{i2})$, $i = 1,2$, being suitable bidimensional vectors (with $\boldsymbol{a}_i, \boldsymbol{b}_i$ not all zero), with the important property that the curves are both parametrized on the *same angular interval* $[\alpha,\beta]$, $0 < \beta - \alpha \le 2\pi$. It is not difficult to show, by a possible reparametrization with a suitable angle shift when $\boldsymbol{a}_i$ and $\boldsymbol{b}_i$ are not orthogonal, that these curves are arcs of two ellipses centered at $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$, respectively (cf. [5]).

Consider the compact domain

$$\Omega = \{(x,y) = \boldsymbol{\sigma}(s,\theta) = s\boldsymbol{\gamma}_1(\theta) + (1-s)\boldsymbol{\gamma}_2(\theta)\ , \quad (s,\theta) \in [0,1]\times[\alpha,\beta]\}\ , \tag{3}$$

which is the transformation of the rectangle $[0,1]\times[\alpha,\beta]$ obtained by convex combination (linear blending) of the arcs $\boldsymbol{\gamma}_1(\theta)$ and $\boldsymbol{\gamma}_2(\theta)$. Observe that the transformation $\boldsymbol{\sigma}$ is analytic and not injective, in general.

It is worth noticing that there is a simple *geometric characterization of injectivity* of the transformation $\boldsymbol{\sigma}$ in the interior of the rectangle, i.e., that the arcs $\boldsymbol{\gamma}_1$ and $\boldsymbol{\gamma}_2$ intersect each other only possibly at their endpoints, and any two segments $[\boldsymbol{\gamma}_1(\theta_1),\boldsymbol{\gamma}_2(\theta_1)]$ and $[\boldsymbol{\gamma}_1(\theta_2),\boldsymbol{\gamma}_2(\theta_2)]$ intersect each other only possibly at one of their endpoints; cf. [5]. In this case, the Jacobian determinant $det(J\boldsymbol{\sigma}(t,\theta))$ doesn't change sign in $(0,1)\times(\alpha,\beta)$.

Theorem 1.2. *Consider the planar domain generated by linear blending of two parametric arcs as in (2)-(3). Let the transformation $\boldsymbol{\sigma}(t,\theta)$ be injective for $(t,\theta) \in (0,1)\times(\alpha,\beta)$, and set $A_0 = (a_{11}-a_{21})(b_{12}-b_{22}) + (a_{12}-a_{22})(b_{21}-b_{11})$, $A_1 = (b_{12}-b_{22})(c_{11}-c_{21}) + (b_{21}-b_{11})(c_{12}-c_{22})$, $A_2 = (a_{11}-a_{21})(c_{12}-c_{22}) + (a_{12}-a_{22})(c_{21}-c_{11})$, and $B_0 = b_{21}(a_{22}-a_{12}) + b_{22}(a_{11}-a_{21})$, $B_1 = b_{21}(c_{22}-c_{12}) + b_{22}(c_{11}-c_{21})$, $B_2 = a_{21}(c_{12}-c_{22}) + a_{22}(c_{21}-c_{11})$, $B_3 = a_{12}a_{21} - a_{11}a_{22} + b_{11}b_{22} - b_{12}b_{21}$, $B_4 = a_{12}b_{21} - a_{11}b_{22} + a_{21}b_{12} - a_{22}b_{11}$.*

Then the following product Gaussian formula with $n^2/2 + O(n)$ nodes holds

$$\iint_\Omega p(\boldsymbol{x})\,d\boldsymbol{x} = \sum_{j=1}^{n+k+1}\ \sum_{i=1}^{\lceil\frac{n+h+1}{2}\rceil} W_{ij}\,p(\boldsymbol{x}_{ij})\ , \quad \forall p \in \mathbb{P}_n^2\ , \tag{4}$$

where $\mathbb{P}_n^2$ *denotes the space of bivariate polynomials of total degree not greater than* n*, with* $h = 0$ *if* $A_i = 0$, $i = 0, 1, 2$, *and* $h = 1$ *otherwise, while* $k = 0$ *if* $A_1 = A_2 = 0$ *and* $B_i = 0$, $i = 1, \ldots, 4$, $k = 1$ *if* $B_3 = B_4 = 0$ *and at least one among* A_1, A_2, B_1, B_2 *is nonzero, and* $k = 2$ *if* $B_3 \neq 0$ *or* $B_4 \neq 0$*. The bivariate nodes and weights in (4) are*

$$\boldsymbol{x}_{ij} = \boldsymbol{\sigma}(\tau_i^{GL}, \theta_j) \in int(\Omega) \ , \quad 0 < W_{ij} = |det(J\boldsymbol{\sigma}(\tau_i^{GL}, \theta_j))|\, w_i^{GL}\, \lambda_j \ , \tag{5}$$

$\{(\theta_j, \lambda_j)\}$ *being the angular nodes and weights of the trigonometric Gaussian formula (1) of degree of exactness* $n + k$ *on* $[\alpha, \beta]$*, and* $\{(\tau_i^{GL}, w_i^{GL})\}$ *the nodes and weights of the Gauss-Legendre formula of degree of exactness* $n + h$ *on* $[0, 1]$.

For the proof of Theorem 1.2 we refer the reader to [5], where the key feature is that $(p \circ \boldsymbol{\sigma})|det(J\boldsymbol{\sigma})|$ belongs to the *tensor-product space* $\mathbb{P}^1_{n+h}([0,1]) \otimes \mathbb{T}^1_{n+k}([\alpha, \beta])$ for every $p \in \mathbb{P}_n^2$. A similar approach, but this time with a purely trigonometric transformation $\boldsymbol{\sigma}$ and $(p \circ \boldsymbol{\sigma})|det(J\boldsymbol{\sigma})|$ in the tensor-product of two subperiodic trigonometric spaces, leads to the construction of product Gaussian formulas on circular *lunes* (difference of two overlapping disks), cf. [8]. All these formulas are implemented in the Matlab package SUBP, cf. [25].

2. Regions Related to Circular Sections

The family of arc-blending domains (2)-(3) contains several disk sections, such as circular (annular) *sectors*, circular *segments* and *zones*, circular *lenses*, as well as their elliptical generalizations. More complicated regions related to circles can be often decomposed in a mutually disjoint union of such basic sections.

We focus here on a relevant subregion, namely a *generalized sector*, that corresponds to the degenerate case of linear blending of a circle arc with a single point (the vertex $\boldsymbol{v}$), in general different from the circle center $\boldsymbol{c}$, i.e., in (2)-(3) we have $\boldsymbol{\gamma}_1(\theta) \equiv \boldsymbol{c}_1 = \boldsymbol{v}$ ($\boldsymbol{a}_1 = \boldsymbol{b}_1 = \boldsymbol{0}$) and $\boldsymbol{a}_2 = (R, 0)$, $\boldsymbol{b}_2 = (0, R)$, $\boldsymbol{c}_2 = \boldsymbol{c}$ (where R is the disk radius). Injectivity of the transformation $\boldsymbol{\sigma}$ depends on the position of the vertex with respect to the arc, and is discussed in detail in [5, Remark 1]. In this case in Theorem 1.2 we have $A_0 = R^2$ and $B_3 = B_4 = 0$, so that $h = 1$, and $k = 0$ if $\boldsymbol{v} = \boldsymbol{c}$ (since $A_1 = A_2 = B_1 = B_2 = B_3 = B_4 = 0$), whereas $k = 1$ if $\boldsymbol{v} \neq \boldsymbol{c}$ (since $A_1 \neq 0$ or $A_2 \neq 0$). In the first instance ($k = 0$) the number of quadrature nodes is $\lceil \frac{n+2}{2} \rceil (n + 1)$, in the second ($k = 1$) it is $\lceil \frac{n+2}{2} \rceil (n + 2)$; see Figure 1-left.

Since the quadrature formulas with polynomial exactness degree n, obtained by collection of nodes and weights via finite union of m regions of arc-blending type, have a cardinality growing like $mn^2/2 + O(n)$, it is worth recalling a *compression* technique recently proposed in [26]. Such a technique is able to reduce the cardinality, by node selection and re-weighting, from about $mn^2/2$ to $\dim(\mathbb{P}_n^2) = (n+1)(n+2)/2 = n^2/2 + O(n)$, that is by roughly a factor $1/m$.

2.1. Tchakaloff's theorem and quadrature compression

A quadrature formula with positive weights is a positive discrete measure with finite support. The following theorem, originally proved by Tchakaloff for absolutely continuous measures, ensures that a discrete measure can be compressed, keeping the same polynomial moments up to a certain degree. We state a quite general version of the theorem for compactly supported measures, taken from [23].

Theorem 2.1. *(Generalized Tchakaloff's Theorem) Let* μ *be a positive measure with compact support* X *in* $\mathbb{R}^d$ *and let* n *be a fixed positive integer. Then there are* $L \leq N = dim(\mathbb{P}_n^d)$ *points* $\{\xi_\ell\}$ *in* X *and positive real numbers* $\{\alpha_\ell\}$ *such that*

$$\int_{\mathbb{R}^d} p(\boldsymbol{x})\, d\mu = \sum_{\ell=1}^{L} \alpha_\ell\, p(\boldsymbol{\xi}_\ell) \ , \quad \forall p \in \mathbb{P}_n^d \ . \tag{6}$$

Given a quadrature formula (for example with respect to the Lebesgue measure $d\boldsymbol{x}$) with polynomial exactness degree n and $M > N$ nodes $X = \{\boldsymbol{x}_i\}$ and positive weights $\boldsymbol{w} = \{w_i\}$, in order to compute the nodes and weights of the compressed formula, we can reformulate the compression problem as the problem

of finding a *nonnegative sparse* solution to the *underdetermined* Vandermonde-like linear system (consider column vectors)

$$V^t \boldsymbol{z} = \boldsymbol{m}\ ,\quad V = (v_{ij}) = (p_j(\boldsymbol{x}_i))\ ,\quad \boldsymbol{m} = V^t \boldsymbol{w}\ , \tag{7}$$

$1 \le i \le M$, $1 \le i \le N$, where $span\{p_1, \ldots, p_N\} = \mathbb{P}_n^d$ and $\boldsymbol{m} = (m_1, \ldots, m_N)$ is the vector of moments of the polynomial basis $\{p_j\}$, $m_j = \sum_{i=1}^M w_i\, p_j(\boldsymbol{x}_i)$. Now, the discrete version of *Tchakaloff's theorem* (with $\mu = \sum_{i=1}^M w_i\, \delta_{\boldsymbol{x}_i}$) ensures that *a nonnegative solution with at least $M-N$ zero components exists*. Therefore, we can solve the underdetermined system (7) via the NNLS (*Non Negative Least Squares*) quadratic programming problem

$$\|\boldsymbol{m} - V^t \boldsymbol{z}\|_2 = \min \|\boldsymbol{m} - V^t \boldsymbol{u}\|_2\ ,\quad \boldsymbol{u} \in \mathbb{R}^M\ ,\quad \boldsymbol{u} \ge \boldsymbol{0}\ , \tag{8}$$

for which several numerical algorithms are known, for example the active set optimization algorithm by Lawson and Hanson [18] that computes a sparse solution (a variant is implemented in the Matlab function `lsqnonneg`). The nonzero components of the solution vector $\boldsymbol{z}$ correspond to the positive weights $\{\alpha_\ell\}$ and allow to extract a subset of nodes $\{\boldsymbol{\xi}_\ell\} \subset X$, $1 \le \ell \le L$, with $L \le N$.

The compression method is implemented in the function `compresscub` of the Matlab package SUBP [26], where one preliminary orthogonalization step is made to pull down the conditioning of V. The default total-degree polynomial basis is the product Chebyshev basis of the minimal Cartesian rectangle containing X, but it can be easily changed into any desired basis, for example the Zernike basis of a suitable disk containing X in the applications below.

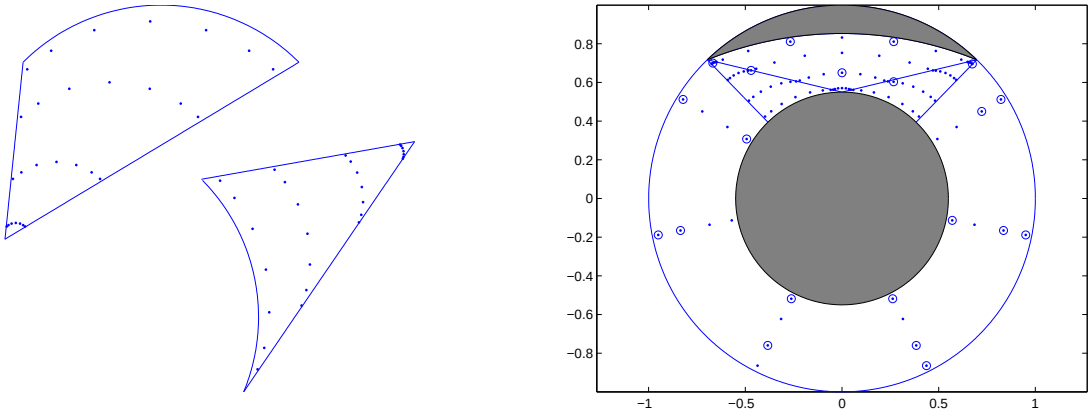

Figure 1: Left: a convex and a concave generalized sector with 28 quadrature nodes (·) for polynomial exactness degree $n = 5$; Right: 108 quadrature nodes (·) for $n = 5$ on an obscured and vignetted pupil, and compression into 21 points (∘) by NNLS.

Since `lsqnonneg` can be very slow on large problems, we can solve alternatively the underdetermined system (7) by *QR with column pivoting*, implemented by the Matlab `mldivide` (or backslash) operator. Although the latter approach guarantees sparsity but not positivity, it turns out in all our numerical experiments that up to relatively high degrees (say 50-60) the negative weights are few and of small size, and the quadrature *stability parameter* $\sum_{\ell=1}^L |\alpha_\ell| / |\sum_{k=1}^L \alpha_\ell|$ is not much larger than 1 (cf., e.g., the last row of Table 1).

As an example of generation of a quadrature formula by union of generalized sectors and compression, relevant to numerical ray-tracing for accurate spot size computation in optical design, we consider a circular pupil (the unit disk) which is obscured by a central smaller disk and clipped by a circular arc of larger radius (see Figure1-right). Recently, Bauman and Xiao [1] have introduced quadrature methods based on prolate spheroidal wave functions, that allow to treat optical apertures (pupils) that are obscured and vignetted (a

feature that occurs for example in optical astronomy). The present example, taken from [1], corresponds indeed to a LSST-like (Large Synoptic Survey Telescope, [22]) aperture.

In [1], a product Gaussian-like quadrature formula is derived for such a pupil, by subdividing it into an unvignetted and a vignetted annular sector. The vignetted sector cannot be treated directly as a linear blending in our framework, since its arcs belong to different circles and correspond to different angular intervals. We split then the pupil into the mutually disjoint union of four subregions, a symmetric annular sector centered at the origin and three generalized sectors centered in suitable vertices. Assembling together the four quadrature formulas of arc-blending type, we get a quadrature formula with $4\lceil\frac{n+1}{2}\rceil(n+1+3(n+2)) = 2n^2 + O(n)$ nodes and positive weights, exact for polynomials up to degree n (this example was already considered in [5]).

Such a formula can be compressed, by the method described in Subsection 2.1, into a positive formula with roughly 1/4 of the nodes, namely $(n+1)(n+2)/2 = n^2/2 + O(n)$ nodes, see Figure 1-right. For instance, at degree $n = 13$, our approach requires 105 nodes, a number which is close to the 112 nodes required by the Bauman-Xiao method (cf. [1, §2.4]). This example shows that our method represents a possible alternative, which guarantees polynomial exactness at a given degree, a feature not offered by the method in [1].

2.2. *Intersection of disks*

To our knowledge, quadrature formulas with polynomial exactness on the intersection of m arbitrary overlapping disks, are not available for $m > 2$. The case $m = 2$ (circular lenses) has been studied for example in [7]. On the other hand, even the basic problem of computing the intersection area, studied for example in [10, 17, 19], is considered a challenging one when many disks are involved (cf. the introduction of [19]).

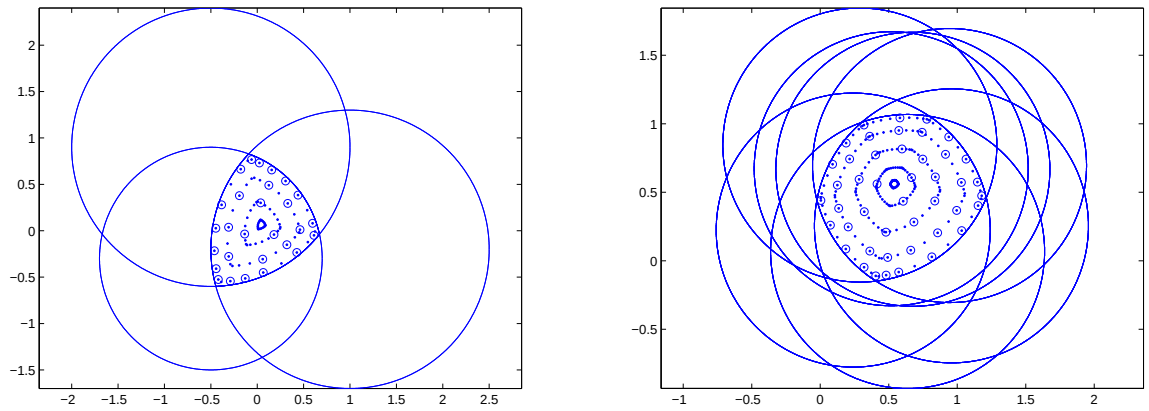

Figure 2: Left: $96 = 32 \times 3$ quadrature nodes ($\cdot$) for $n = 6$ on the intersection of 3 circles and compression into 28 nodes ($\circ$) by NNLS; Right: $250 = 50 \times 5$ quadrature nodes for $n = 8$ on a curvilinear pentagon (intersection of 7 circles with equal radius) and compression into 45 nodes.

The intersection of m disks is a convex curvilinear polygon, whose vertices are among the intersection points of circles pairs and whose sides are circle arcs; see Figure 2. Our algorithm (implemented by the Matlab function `gqintdisk` of [25]) for the construction of a quadrature formula of polynomial degree of exactness n on the intersection of m disks (given by their m centers and radii), is structured in the following steps:

algorithm (*quadrature on the intersection of m planar disks*)

- eliminate the circles containing other circles by an inclusion test;
- compute all the pairwise intersection points of the remaining circles by the Matlab function `circcirc`;

- select among the points those belonging to all the disks (which form the vertices of a curvilinear polygon corresponding to the intersection of the disks), order them counterclockwise by the Matlab function `convhull` and compute the barycenter;
- attach to each vertex the circle arcs passing through the vertex itself and take as arc connecting two consecutive vertices that of minimal length among the common ones;
- compute the angular intervals for the arcs connecting couples of consecutive vertices and construct the product Gaussian formulas of degree n on the corresponding generalized sectors with vertex in the barycenter;
- collect together the nodes and weights of the sectors (basic quadrature formula) and compress the basic formula (low cardinality quadrature formula).

Notice that with both formulas, the basic one and the compressed one, the *sum of the weights* gives the *area* of the m disks intersection, at machine precision (already with degree $n = 1$ by `gqintdisk` in [26], cf. Figure 3). Computation of the intersection area is relevant to applications in wireless networks, cf., e.g., [19]. Moreover, the algorithm provides as a byproduct the vertices and the arcs of the intersection, a feature that can be useful in several applications.

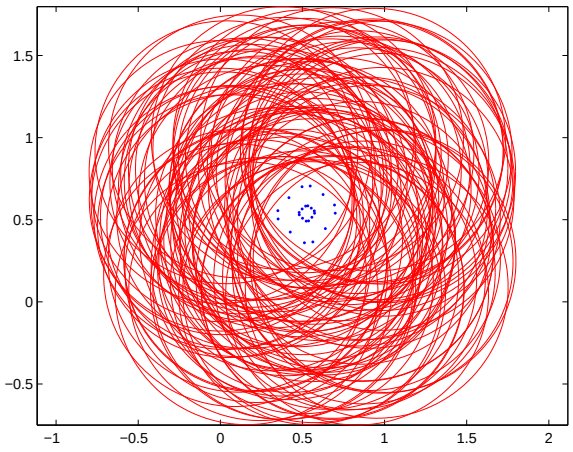

Figure 3: Computing the intersection area of a tangle of 100 unit disks with random centers, by 24 nodes and weights (exactness degree $n = 1$).

2.2.1. *Applications*

For the purpose of illustration, we quote two possible applications in the framework of computational optics. The first concerns evaluation of integrals of the form

$$I(\boldsymbol{\nu}_1, \boldsymbol{\nu}_2) = \iint_{D_1 \cap D_2 \cap D_3} A(\boldsymbol{x})\, B(\boldsymbol{x} - \boldsymbol{\nu}_1)\, C^*(\boldsymbol{x} - \boldsymbol{\nu}_2)\, d\boldsymbol{x}\,, \tag{9}$$

where D_1 is the unit disk and $D_i = \{\boldsymbol{x} : \|\boldsymbol{x} - \boldsymbol{\nu}_i\|_2 \le R_i\}$, $i = 1, 2$, that occur in optical diffraction theory under conditions of partial coherence (Hopkins' theory [15]), where A is a source intensity and B, C are generalized pupil functions (with phase and amplitude non-uniformities, $*$ denoting complex conjugation). A relevant case is TCC (Transmission Cross Coefficient) computation, where $C = B$, numerical quadrature being one of the possible approaches provided that efficient formulas are at hand; cf., e.g., [16, 30] and the references therein.

It is worth observing that, depending on the relative position of the disks and on the rays size, the intersection of three overlapping circles has from 2 up to 4 sides, so that the basic quadrature formula obtained collecting nodes and weights of three generalized sectors has already a relatively low average cardinality (of the order of $\frac{3}{2}n^2$ nodes), and the compression process could be avoided. This observation is relevant especially when a large number of integrals like (9) has to be computed, since the region of intersection changes with $\boldsymbol{\nu}_1$ and $\boldsymbol{\nu}_2$.

The second application to optics concerns numerical ray-tracing through noncircular pupils, cf., e.g., [1]. We consider here the aperture of a curved side lens diaphragm. Such diaphragms are made of a number of curved blades (whose internal arc curvature can be assumed constant) moving symmetrically in such a way that one gets apertures with different diameters, whose shape is a regular curvilinear polygon. The number of blades vary from 5-6 to more than 10. In order to model a diaphragm, we consider a disk centered at the origin with a fixed radius ≤ 1 (corresponding to the aperture diameter) and an inscribed regular linear polygon with m sides. We vignette then the disk by taking as aperture the intersection of m larger disks, each centered at the point of the axis of a side (in the half-plane containing the disk center), whose distance from the side endpoints is the reciprocal of the blade curvature; see Figure 4, where the m disks are dashed, and the unit disk (in bold) corresponds to the maximal aperture.

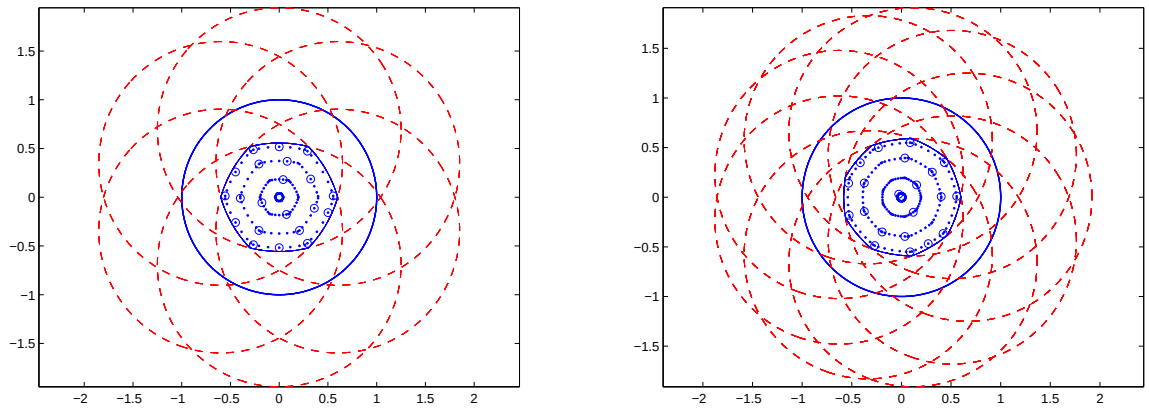

Figure 4: Quadrature nodes for two diaphragms, degree $n = 5$; Left: 6-blade diaphr., apert. ray 0.6, blade curv. 0.8, $28 \times 6 = 168$ nodes ($\cdot$) compressed by QR with column pivoting into 21 ($\circ$); Right: 9-blade diaphr., apert. ray 0.8, blade curv. 0.8, $28 \times 9 = 252$ nodes compressed by NNLS into 21.

To illustrate the applications just discussed as well as the flexibility of the method, in Figures 2 and 3 we display the nodes of the basic and of the compressed quadrature formula on various intersections of disks. Finally, to show the accuracy of our quadrature formulas, in Table 1 we report an example of the errors on the moments of the Zernike polynomial basis (a reference basis in optical applications), namely for the domain of Figure 4-right. For the purpose of comparison, the values of the Zernike moments have been evaluated, at machine precision, by the Matlab codes of [24], that use Green's formula together with high-precision polynomial representation of the domain boundary by the Chebfun software package [9]. All the numerical tests have been made in Matlab 7.7.0 with an Athlon 64 X2 Dual Core 4400+ 2.40GHz processor.

We can see that computation of the Zernike moments is quite accurate with all the three quadrature formulas, and that compression by QR with column pivoting is more efficient than NNLS at the higher degrees, with a very small effect of the negative weights on the stability parameter, which stays close to 1. A number of numerical experiments, not reported for brevity, have shown a similar behavior of the basic and the compressed formulas on the regions of Figures 1-3, as well as on other examples of regions that are mutually disjoint union of generalized sectors or different arc-blending domains.

Table 1: RMS error of the basic and compressed quadrature formulas on the vector of Zernike moments for the curvilinear nonagon of Figure 4-right (together with the formulas cardinalities, the compression CPU time in seconds and the stability parameter $\sum |\alpha_\ell| / |\sum \alpha_\ell|$ for compression by QR with column pivoting).

degree	3	6	9	12	15	18
card	135	288	594	882	1377	1800
basic	3e-16	5e-16	4e-16	8e-16	3e-15	2e-14
compr card	10	28	55	91	136	190
NLLS	5e-16	4e-16	6e-16	1e-15	8e-15	5e-14
CPU time	0.01s	0.02s	0.05s	0.24s	0.69s	2.08s
QRpiv	4e-16	4e-16	5e-16	2e-15	5e-15	3e-14
CPU time	< 0.01s	0.01s	0.09s	0.21s	0.36s	1.18s
stab parm	1.00	1.00	1.03	1.13	1.16	1.04

2.3. Union of disks

As in the case of the intersection, to our knowledge quadrature formulas with polynomial exactness on the union of m arbitrary disks are not available for $m > 2$. The case $m = 2$ ("double bubbles") has been studied in [7]. On the other hand, once suitable algebraic quadrature formulas for the intersection of disks are available, we can use them to compute algebraic quadrature formulas for the union of disks, provided that the number of disks is relatively low, by resorting to the integral version of the *inclusion-exclusion principle*. Such a combinatorial principle, extended to measure theory (and to integration theory), and specialized to the present case of m disks $D_1, \ldots, D_m$, says that

$$\iint_{\bigcup D_i} f(\boldsymbol{x})\, d\boldsymbol{x} = \sum_{k=1}^{m} (-1)^k \sum_{\{i_1,\ldots,i_k\}\subseteq\{1,\ldots,m\}} \iint_{D_{i_1}\cap\cdots\cap D_{i_k}} f(\boldsymbol{x})\, d\boldsymbol{x}\,, \tag{10}$$

for every integrable function f. The extension from positive measures to integrals can be obtained by considering the measures $\mu^+(E) = \int_E f^+(\boldsymbol{x})\, d\boldsymbol{x}$ and $\mu^-(E) = \int_E f^-(\boldsymbol{x})\, d\boldsymbol{x}$, where f^+ and f^- are the positive and the negative part of f, $f = f^+ - f^-$. Thus, if $f = p \in \mathbb{P}_n^2$ and we term $\{w_j^{(S)}, \boldsymbol{x}_j^{(S)}\}$ the weights and nodes of an algebraic formula for the intersection $D_{i_1} \cap \cdots \cap D_{i_k}$, where $S = \{i_1, \ldots, i_k\}$, we get the quadrature formula

$$\iint_{\bigcup D_i} p(\boldsymbol{x})\, d\boldsymbol{x} = \sum_{k=1}^{m} (-1)^k \sum_{S=\{i_1,\ldots,i_k\}\subseteq\{1,\ldots,m\}} \sum_{j=1}^{\nu(S)} w_j^{(S)}\, p(\boldsymbol{x}_j^{(S)})\,, \tag{11}$$

with variable sign weights $\{(-1)^k w_j^{(S)}\}$, where $\nu(S) \lessapprox (k-1)n^2$ is the number of nodes of the quadrature formula for the intersection of the disks corresponding to the indices in S (recall that the number of vertices of the intersection of k disks cannot exceed $2(k-1)$, cf. e.g. [28, Ch. 6], and the cardinality of the quadrature formula of degree n for an intersection of disks with $s \geq 2$ vertices grows like $sn^2/2$).

The number of possible intersections is $\sum_{k=1}^{m} \binom{m}{k} = 2^m - 1$, i.e., the complexity grows exponentially with the number of disks. Several efforts have been made in the literature to reduce such a complexity, cf., e.g., [13] and the references therein. In the Matlab function `gqmultibubble` of [25], due to the arbitrariness of the disks, we have adopted a naive implementation of (11). The only trick we use is to keep in the successive stages only the disks that at a given stage k correspond to at least one nonempty intersection (i.e., to eliminate the disks whose intersection with any other combination of $k-1$ disks is empty, that cannot contribute to higher order intersections), with possible anticipate exit when the set of "active" disks becomes empty.

Now, it is not difficult to prove that, in the limit case when all the intersections of combination of k disks, $2 \leq k \leq m$, have positive measure (equivalently, the intersection of the m disks has positive measure), the cardinality of the resulting (nonpositive) quadrature formula can be asymptotically bounded

as $2^{m-1}n^2 \lesssim M = \mathrm{card}\left(\{\boldsymbol{x}_j^{(S)}\}\right) \lesssim m2^{m-1}n^2$ for fixed m and $n \to \infty$. On the other hand, in practice the true cardinalities can be significantly lower, but in any case a huge number of nodes arises already for moderate values of m.

In order to avoid working with such huge cardinalities, we pursue here an alternative approach with respect to compression applied directly to the original quadrature formula. Indeed, we use the original quadrature formula only to compute the moments of the chosen polynomial basis as $\boldsymbol{m} = V^t\boldsymbol{w}$, $V = (v_{ij}) = (p_j(\boldsymbol{x}_i))$, $1 \le i \le M$, $1 \le j \le N$, where $\{\boldsymbol{x}_i\}$ is any ordering of the nodal array $\{\boldsymbol{x}_j^{(S)}\}$ and $\boldsymbol{w}$ the same ordering of the weights $\{(-1)^k w_j^{(S)}\}$. On the other hand, we seek a sparse solution to the underdetermined Vandermonde-like system

$$U^t\boldsymbol{z} = \boldsymbol{m}\ , \quad U = (u_{ij}) = (p_j(\boldsymbol{y}_i))\ , \quad \boldsymbol{m} = V^t\boldsymbol{w}\ , \tag{12}$$

$1 \le i \le \mathcal{V}$, $1 \le j \le N$, on a *polynomial (weakly admissible) mesh* $Y = \{\boldsymbol{y}_i\}$ of the union of disks, with $\mathcal{V} = \mathrm{card}(Y)$ and $N = \dim(\mathbb{P}_n^2) = (n+1)(n+2)/2$. We recall that polynomial meshes are discrete norming sets for polynomials of total degree not exceeding n on a multidimensional compact set, well-suited for polynomial fitting, and interpolation on *discrete extremal sets* of Fekete type extracted from them. For the theoretical and computational features of polynomial meshes we refer the reader, e.g., to [4, 27] and the references therein.

Now, polynomial meshes for a union of compact sets are simply union of polynomial meshes of such sets, so that it is immediate to construct a polynomial mesh for the union of m arbitrary disks, starting from a polynomial mesh for a disk (polynomial meshes are invariant under affine transformations). There are several ways to generate a polynomial mesh on a disk by arc blending: see [27, Subsection 2.1.6]. Using for example a radial mesh, the resulting cardinality for the union is $\mathcal{V} \approx mn^2$, to be compressed into $N = (n+1)(n+2)/2$ nodes (and weights) by (12).

In this case we cannot ensure a priori existence of a positive sparse solution by Tchakaloff's theorem, since the polynomial mesh is not the support of the measure, more precisely we do not deal at all with a positive measure because the weights have variable sign. Nevertheless, we can still try to extract the nodes and compute the weights by QR with column pivoting applied to (12), as in Section 2.1 (the latter approach has already been shown to perform well in the construction of algebraic quadrature formulas on polygons, cf. [14]). The N nonzero components of the solution vector $\boldsymbol{z}$ are the weights corresponding to a subset of nodes $\{\boldsymbol{y}_{i_s}\}$, $1 \le s \le N$, extracted from the polynomial mesh. These nodes are ultimately Approximate Fekete Points, as discussed in [2].

For the purpose of illustration, in Figure 5 we show the quadrature nodes on two unions of 10 disks, for degree $n = 12$, the first connected and the second disconnected, with quite different intersection patterns among combinations of disks. In Table 2 we report the residual euclidean norm, cardinality, CPU time and stability parameter of quadrature compression by QR with column pivoting on the disk unions of Figure 5 (here $\{p_j\}$ in (12) is the product Chebyshev basis of the smallest cartesian rectangle containing the union, cf. [4]).

As already observed, the essentially exponential complexity of a straightforward application of the inclusion-exclusion principle prevents from using effectively the present approach, already with a relatively small number of disks. For this reason a more direct approach, based on boundary tracking of the disk union and suitable splitting into curvilinear and polygonal subregions, will be object of further study.

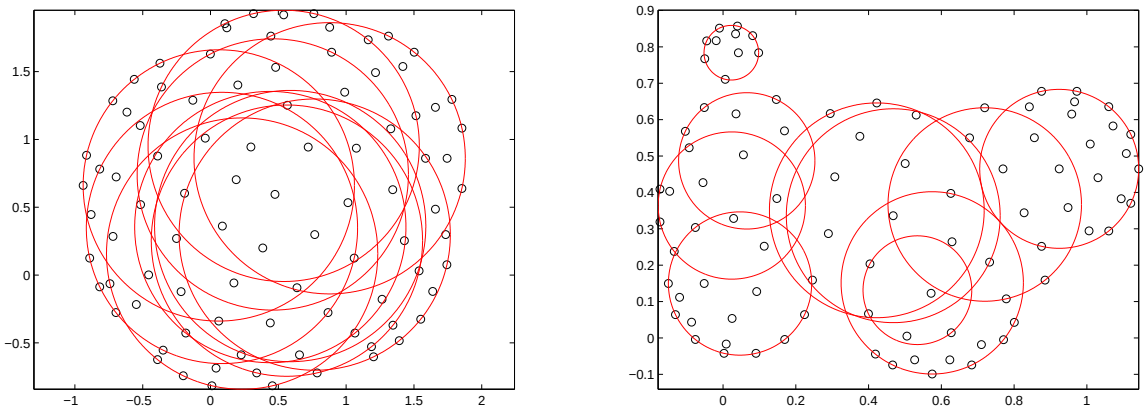

Figure 5: Quadrature nodes on two unions of 10 disks, for degree $n = 12$.

Table 2: Residual 2-norm, cardinality, CPU time and stability parameter of quadrature compression by QR with column pivoting on the disk unions of Figure 5 ($*$: failure, out of memory).

	degree	3	6	9	12	15	18
Figure 5-left	card	59145	126496	260598	387394	604359	$*$
	compr card	10	28	55	91	136	$*$
	res norm	3e-16	5e-16	4e-16	4e-16	2e-15	$*$
	CPU time	7.71s	8.37s	11.15s	18.55s	27.52s	$*$
	stab parm	1.01	1.61	1.17	1.40	1.03	$*$
Figure 5-right	card	1392	3152	6330	9653	14814	19700
	compr card	10	28	55	91	136	190
	res norm	6e-17	3e-17	3e-17	2e-17	3e-17	2e-17
	CPU time	0.57s	0.52s	0.61s	0.85s	1.40s	2.43s
	stab parm	1.01	1.02	1.02	1.06	1.05	1.06

References

[1] B. Bauman and H. Xiao, Gaussian quadrature for optical design with noncircular pupils and fields, and broad wavelength range, Proc. SPIE **7652** (2010).
[2] L. Bos, S. De Marchi, A. Sommariva and M. Vianello, Computing multivariate Fekete and Leja points by numerical linear algebra, SIAM J. Numer. Anal. **48** (2010) 1984–1999.
[3] L. Bos and M. Vianello, Subperiodic trigonometric interpolation and quadrature, Appl. Math. Comput. **218** (2012) 10630–10638.
[4] S. De Marchi, F. Piazzon, A. Sommariva and M. Vianello, Polynomial Meshes: Computation and Approximation, Proceeedings of CMMSE 2015, 414–425, ISBN 978-84-617-2230-3, ISSN 2312-0177.
[5] G. Da Fies, A. Sommariva and M. Vianello, Algebraic cubature by linear blending of elliptical arcs, Appl. Numer. Math. **74** (2013) 49–61.
[6] G. Da Fies and M. Vianello, Trigonometric Gaussian quadrature on subintervals of the period, Electron. Trans. Numer. Anal. **39** (2012) 102–112.
[7] G. Da Fies and M. Vianello, Algebraic cubature on planar lenses and bubbles, Dolomites Res. Notes Approx. DRNA **5** (2012) 7–12.
[8] G. Da Fies and M. Vianello, Product Gaussian quadrature on circular lunes, Numer. Math. Theory Methods Appl. **7** (2014) 251–264.
[9] T.A. Driscoll, N. Hale, and L.N. Trefethen, editors, Chebfun Guide, Pafnuty Publications, Oxford, 2014.
[10] M.P. Fewell, Area of common overlap of three circles, Maritime Operations Division, Australian Government, Department of Defence, Tech. Rep. DSTO-TN-0722, 2006.
[11] G.W. Forbes, Optical system assessment of design: numerical ray tracing into the Gaussian pupil, JOSA A **5** (1988) 1943–1956.
[12] W. Gautschi, Orthogonal Polynomials: Computation and Approximation, Oxford University Press, New York, 2004.

[13] X. Goaoc, J. Matousek, P. Patak, Z. Safernova and M. Tancer, Simplifying inclusion-exclusion formulas, Combin. Probab. Comput. **24** (2015) 438–456.
[14] M. Gentile, A. Sommariva and M. Vianello, Polynomial interpolation and cubature over polygons preprint, J. Comput. Appl. Math. **235** (2011) 5232–5239.
[15] H.H. Hopkins, On the diffraction theory of optical images, Proc. Royal Soc. London A **217** (1953) 408–432.
[16] A.J.E.M. Janssen, Computation of Hopkins' 3-circle integrals using Zernike expansions, J. Eur. Opt. Soc.-Rapid Publ. **6** (2011).
[17] K.W. Kratky, The area of intersection of n equal circular disks, J. Phys. A **11** (1978) 1017–1024.
[18] C.L. Lawson and R.J. Hanson, Solving least squares problems, Classics in Applied Mathematics 15, SIAM, Philadelphia, 1995.
[19] F. Librino, M. Levorato and M. Zorzi, An algorithmic solution for computing circle intersection areas and its applications to wireless communications, Wireless Communications and Mobile Computing **14** (2014) 1672–1690.
[20] Mathworks, Matlab documentation (2015), online at:`http://www.mathworks.com/help/matlab`.
[21] G.V. Milovanovic, Chapter 11: Orthogonal polynomials on the real line, In: Walter Gautschi: Selected Works and Commentaries, Volume 2 (C. Brezinski, A. Sameh, eds.), pp. 3–16, Birkhauser, Basel, 2014.
[22] S.S. Olivier, L. Seppala and K. Gilmore, Optical design of the LSST camera, Proc. SPIE **7018** (2008).
[23] M. Putinar, A note on Tchakaloff's theorem, Proc. Amer. Math. Soc. **125** (1997), 2409–2414.
[24] G. Santin, A. Sommariva and M. Vianello, An algebraic cubature formula on curvilinear polygons, Appl. Math. Comput. **217** (2011) 10003–10015.
[25] A. Sommariva and M. Vianello, `SUBP`: Matlab package for subperiodic trigonometric quadrature and multivariate applications, online at: `http://www.math.unipd.it/~marcov/CAAsoft.html`.
[26] A. Sommariva and M. Vianello, Compression of multivariate discrete measures and applications, Numer. Funct. Anal. Optim. 36 (2015) 1198–1223.
[27] A. Sommariva and M. Vianello, Polynomial fitting and interpolation on circular sections, Appl. Math. Comput. **258** (2015) 410–424.
[28] M. Terwilliger, Localization in Wireless Sensors Networks, Ph.D. Dissertation, Dept. of Computer science, Western Michigan University, April 2006.
[29] M. Terwilliger, C. Coullard, A. Gupta, Localization in Ad Hoc and Sensor Wireless Networks with Bounded Errors, Proc. of High Performance Computing - HiPC 2008, Springer LNCS **5374** (2008) 295–308.
[30] X. Wu, S. Liu, W. Liu, T. Zhou and L. Wang, Comparison of three TCC calculation algorithms for partially coherent imaging simulation, Proc. SPIE **7544** (2010).