From subperiodic to low cardinality quadrature formulas for noncircular pupils*

Alvise Sommariva and Marco Vianello
Department of Mathematics, University of Padova (Italy)
May 28, 2015

Abstract

We discuss numerical integration formulas with polynomial exactness for noncircular pupils, based on subperiodic trigonometric Gaussian quadrature and compression of discrete measures. Such formulas can be useful in optical design (numerical ray-tracing in obscured and vignetted pupils for accurate spot size computation), and in optical diffraction theory under conditions of partial coherence (Hopkins’ 3-circle integrals, transmission cross coefficients (TCC) computation).

Keywords: subperiodic trigonometric Gaussian quadrature, noncircular pupils, numerical ray-tracing, intersection of disks, Hopkins’ 3-circle integrals

1 Introduction

Subperiodic trigonometric approximation, that is approximation by trigonometric polynomials on subintervals of the period, has received some attention in recent years, cf. [2,3,4,6]. 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

*Supported the ex-60% funds and by the biennial projects CPDA124755/CPDA143275 of the University of Padova, and by the INdAM GNCS.
1corresponding author: e-mail: marcov@math.unipd.it
torus. On such regions, algebraic polynomials belong, by suitable geometric transformations, to tensor-product spaces of trigonometric (or of algebraic with trigonometric) univariate polynomials, 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 [3, 6] numerical integration formulas with polynomial exactness have been constructed on different sections of the disk and on domains obtainable by internally disjoint union of such sections. In the computational optics framework, these formulas appear well-suited to integration problems on noncircular pupils, such as Gaussian quadrature for numerical ray-tracing in obscured and vignetted pupils (accurate spot size computation), or optical diffraction simulation under conditions of partial coherence (TCC computation), cf., e.g., [1, 8, 11, 19] and the references therein. In particular, by union of generalized circular sectors we are able to obtain numerical integration formulas with polynomial exactness on the intersection of three (e.g., Hopkins’ 3-circle integrals) or more disks (e.g., regular curvilinear polygons corresponding to apertures of curved blade diaphragms).

For the reader’s convenience, we report the main result of [4] concerning subperiodic trigonometric Gaussian quadrature, stated here for a general angular interval. Below, we shall denote by $T_n$ the $2n+1$-dimensional space of univariate trigonometric polynomials of degree not exceeding $n$, and by $P^d_n$ the $N$-dimensional space of $d$-variate algebraic polynomials with total degree not exceeding $n$, $N = \binom{n+d}{d}$.

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

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

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. [9]. Formula (1) can be effectively implemented...
in Matlab by Gautschi’s OPQ (Orthogonal Polynomials and Quadrature) suite [9], via the modified Chebyshev algorithm [4].

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

\[ \gamma_1(\theta) = a_1 \cos(\theta) + b_1 \sin(\theta) + c_1, \quad \gamma_2(\theta) = a_2 \cos(\theta) + b_2 \sin(\theta) + c_2, \quad (2) \]

\( \theta \in [\alpha, \beta] \), be two trigonometric planar curves of degree one, \( a_i = (a_{i1}, a_{i2}) \), \( b_i = (b_{i1}, b_{i2}) \), \( c_i = (c_{i1}, c_{i2}) \), \( i = 1, 2 \), being suitable bidimensional vectors (with \( a_i, b_i \) not all zero), with the important property that the curves are both parametrized on the same angular interval \([\alpha, \beta]\), \( 0 < \beta - \alpha \leq 2\pi \). It is not difficult to show, by a possible reparametrization with a suitable angle shift when \( a_i \) and \( b_i \) are not orthogonal, that these curves are arcs of two ellipses centered at \( c_1 \) and \( c_2 \), respectively (cf. [3]).

Consider the compact domain

\[ \Omega = \{(x, y) = \sigma(s, \theta) = s\gamma_1(\theta) + (1-s)\gamma_2(\theta), \quad (s, \theta) \in [0, 1] \times [\alpha, \beta]\}, \quad (3) \]

which is the transformation of the rectangle \([0, 1] \times [\alpha, \beta]\) obtained by convex combination (linear blending) of the arcs \( \gamma_1(\theta) \) and \( \gamma_2(\theta) \). Observe that the transformation \( \sigma \) is analytic and not injective, in general.

It is worth noticing that there is a simple geometric characterization of injectivity of the transformation \( \sigma \) in the interior of the rectangle, i.e., that the arcs \( \gamma_1 \) and \( \gamma_2 \) intersect each other only possibly at their endpoints, and any two segments \([\gamma_1(\theta_1), \gamma_2(\theta_1)]\) and \([\gamma_1(\theta_2), \gamma_2(\theta_2)]\) intersect each other only possibly at one of their endpoints; cf. [3]. In this case, the Jacobian determinant \( \text{det}(J\sigma(t, \theta)) \) doesn’t change sign in \((0, 1) \times (\alpha, \beta)\).

**Theorem 2** Consider the planar domain generated by linear blending of two parametric arcs as in [2]–[3]. Let the transformation \( \sigma(t, \theta) \) 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

\[ \int_{\Omega} p(x) \, dx = \sum_{j=1}^{n+k+1} \sum_{i=1}^{[\frac{n+k+1}{2}]} W_{ij} p(x_{ij}) \quad \forall p \in \mathbb{P}_n^2, \quad (4) \]

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

\[
\mathbf{x}_{ij} = \sigma(\tau_i^{GL}, \theta_j + \mu) \in \text{int}(\Omega), \quad 0 < W_{ij} = |\det(J\sigma(\tau_i^{GL}, \theta_j + \mu))| w_i^{GL} \lambda_j,
\]

(5)

\( \{ (\theta_j + \mu, \lambda_j) \} \) being the angular nodes and weights of the trigonometric Gaussian formula \((7)\) 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 2 we refer the reader to [3], where the key feature is that \((p \circ \sigma)|\det(J\sigma)|\) belongs to the tensor-product space \( \mathbb{P}_{n+k}^1([0, 1]) \otimes \mathbb{T}_{n+k}^1([\alpha, \beta]) \) for every \( p \in \mathbb{P}^2_n \). A similar approach, but this time with a purely trigonometric transformation \( \sigma \) and \((p \circ \sigma)|\det(J\sigma)|\) 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. [6]. All these formulas are implemented in the Matlab package SUBP, cf. [17].

2 Integration on noncircular pupils

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 noncircular pupils can be often decomposed in an internally disjoint union of such regions.

We focus here on a relevant subregion for noncircular pupils, a generalized sector of a disk, that corresponds to the degenerate case of linear blending of a circle arc with a single point (the vertex \( v \)), in general different from the circle center \( c \), i.e., in (2)-(3) we have \( \gamma_1(\theta) \equiv c_1 = v \) \( (a_1 = b_1 = 0) \) and \( a_2 = (R, 0), \ b_2 = (0, R), \ c_2 = c \) (where \( R \) is the disk radius). Injectivity of the transformation \( \sigma \) depends on the position of the vertex with respect to the arc, and is discussed in detail in [3] Remark 1. In this case in Theorem 2 we have \( A_0 = R^2 \) and \( B_3 = B_4 = 0 \), so that \( h = 1 \), and \( k = 0 \) if \( v = c \) (since \( A_1 = A_2 = B_1 = B_2 = 0 \) \( B_3 = B_4 = 0 \)), whereas \( k = 1 \) if \( v \neq c \) (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 \( mn^2/2 + \mathcal{O}(n) \), it is worth recalling a compression technique recently proposed in [18]. Such a technique is able to reduce the cardinality, by node selection and re-weighting, from about \( mn^2/2 \) to \( \dim(\mathbb{P}_n^2) = (n + 1)(n + 2)/2 = n^2/2 + \mathcal{O}(n) \), 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 [15].

**Theorem 3** (Generalized Tchakaloff’s Theorem) Let \( \mu \) 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

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

Given a quadrature formula (for example with respect to the Lebesgue measure \( dx \)) with polynomial exactness degree \( n \) and \( M > N \) nodes \( X = \{x_i\} \) and positive weights \( w = \{w_i\} \), 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)

\[
V^t z = m, \quad V = (v_{ij}) = (p_j(x_i)), \quad m = V^t w, \tag{7}
\]

\( i = 1, \ldots, M, \, j = 1, \ldots, N \), where \( \text{span}\{p_1, \ldots, p_N\} = \mathbb{P}_n^d \) and \( m = (m_1, \ldots, m_N) \) is the vector of moments of the polynomial basis \( \{p_j\} \), \( m_j = \sum_{i=1}^M w_i p_j(x_i) \). Now, the discrete version of Tchakaloff’s theorem (with \( \mu = \sum_{i=1}^M w_i \delta_{x_i} \)) 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

\[
\|m - V^t z\|_2 = \min \|m - V^t u\|_2, \quad u \in \mathbb{R}^M, \; u \geq 0, \tag{8}
\]
for which several numerical algorithms are known, for example the active set optimization algorithm by Lawson and Hanson [12] that computes a sparse solution (a variant is implemented in the Matlab function \texttt{lsqnonneg}). The nonzero components of the solution vector \( z \) correspond to the positive weights \( \{\alpha_\ell\} \) and allow to extract a subset of nodes \( \{\xi_\ell\} \subset X, 1 \leq \ell \leq L \), with \( L \leq N \).

The compression method is implemented in the function \texttt{compresscub} of the Matlab package SUBP [13], 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. Since \texttt{lsqnonneg} can be very slow on large problems, we can solve alternatively the underdetermined system (7) by \textit{QR with column pivoting}, implemented by the Matlab \texttt{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 \( \sum_{\ell=1}^{L} |\alpha_\ell| / \sum_{k=1}^{L} |\alpha_k| \) oscillates remaining not far from 1 (cf., e.g., the last row of Table 1).

### 2.2 An obscured and vignetted pupil

As a first example, 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, [14]) 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 internally 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 \( \lceil \frac{n+1}{2} \rceil (n + 1 + 3(n + 2)) = 2n^2 + \mathcal{O}(n) \) nodes.
and positive weights, exact for polynomials up to degree $n$ (this example was already considered in [3]). 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 on the other hand guarantees polynomial exactness at a given degree.

2.3 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 [5]. 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 function gqintdisk of the package SUBP [17, 18]) 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:

(i) compute all the pairwise intersection points of the $m$ circles by the Matlab function circcirc, indexing each point by the corresponding circles;
(ii) select the intersection points that belong to all the disks (the curvilinea...
Figure 2: Left: $96 = 32 \times 3$ quadrature nodes (·) for $n = 6$ on the intersection of 3 circles and compression into 28 nodes (○) 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.

ear polygon vertices), order them counterclockwise by the Matlab function \texttt{convhull} and compute the barycenter;

(iii) compute the angular intervals for the arcs connecting two consecutive vertices and construct the product Gaussian formulas of degree $n$ on the corresponding generalized sectors with vertex in the barycenter;

(iv) 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 sum of the weights gives the area of the $m$ disks intersection, at machine precision. We recall two possible applications in the framework of computational optics. The first concerns evaluation of integrals of the form

$$I(\nu_1, \nu_2) = \int\int_{D_1 \cap D_2 \cap D_3} A(x) B(x - \nu_1) C^*(x - \nu_2) \, dx ,$$  \hspace{1cm} (9)

where $D_1$ is the unit disk and $D_i = \{ x : \| x - \nu_i \|_2 \leq R_i \}$, $i = 1, 2$, that occur in optical diffraction theory under conditions of partial coherence (Hopkins’ theory \cite{10}), 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 quadrature formulas are at hand; cf., e.g., \cite{11,19} 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 circles has typically three sides (exceptionally two or four), so that the basic quadrature formula obtained collecting nodes and weights of three generalized sectors (cf. (iv) in the algorithm above) has already a relatively low 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 intersection changes by varying $\nu_1$ and $\nu_2$.

The second application concerns numerical ray-tracing through the aperture of a curved side diaphragm. Such diaphragms are made of a number of 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 $\leq 1$ (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 3, where the $m$ disks are dashed, and the unit disk (in bold) corresponds to the maximal aperture.

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
Table 1: RMS error of the basic and compressed quadrature formulas on the vector of Zernike moments for the curvilinear nonagon of Figure 3-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).

<table>
<thead>
<tr>
<th>degree</th>
<th>3</th>
<th>6</th>
<th>9</th>
<th>12</th>
<th>15</th>
<th>18</th>
</tr>
</thead>
<tbody>
<tr>
<td>card</td>
<td>135</td>
<td>288</td>
<td>594</td>
<td>882</td>
<td>1377</td>
<td>1800</td>
</tr>
<tr>
<td>basic formula err</td>
<td>$3.2 \times 10^{-16}$</td>
<td>$4.8 \times 10^{-16}$</td>
<td>$4.1 \times 10^{-16}$</td>
<td>$7.5 \times 10^{-16}$</td>
<td>$2.7 \times 10^{-15}$</td>
<td>$1.6 \times 10^{-14}$</td>
</tr>
<tr>
<td>compressed card</td>
<td>10</td>
<td>28</td>
<td>55</td>
<td>91</td>
<td>136</td>
<td>190</td>
</tr>
<tr>
<td>NLLS err</td>
<td>$5.1 \times 10^{-16}$</td>
<td>$4.4 \times 10^{-16}$</td>
<td>$5.9 \times 10^{-16}$</td>
<td>$9.5 \times 10^{-16}$</td>
<td>$7.8 \times 10^{-15}$</td>
<td>$5.2 \times 10^{-14}$</td>
</tr>
<tr>
<td>CPU time</td>
<td>&lt; 0.01s</td>
<td>0.02s</td>
<td>0.05s</td>
<td>0.24s</td>
<td>0.69s</td>
<td>2.08s</td>
</tr>
<tr>
<td>QRpiv err</td>
<td>$3.8 \times 10^{-16}$</td>
<td>$4.4 \times 10^{-16}$</td>
<td>$5.4 \times 10^{-16}$</td>
<td>$1.7 \times 10^{-15}$</td>
<td>$4.8 \times 10^{-15}$</td>
<td>$2.8 \times 10^{-14}$</td>
</tr>
<tr>
<td>CPU time</td>
<td>&lt; 0.01s</td>
<td>0.01s</td>
<td>0.09s</td>
<td>0.21s</td>
<td>0.36s</td>
<td>1.18s</td>
</tr>
<tr>
<td>stab parm</td>
<td>1.00</td>
<td>1.00</td>
<td>1.03</td>
<td>1.13</td>
<td>1.16</td>
<td>1.04</td>
</tr>
</tbody>
</table>

an example of the errors on the moments of the Zernike polynomial basis, namely for the domain of Figure 3-right. For the purpose of comparison, the reference values of the Zernike moments have been evaluated, at machine precision, by the Matlab codes of [16], that use Green’s formula together with high-precision polynomial representation of the domain boundary by the Chebfun software package [7]. 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 internally disjoint union of generalized sectors or different arc-blending domains.

References


