On the application of tensorial rules

Alvise Sommariva

Seminaria Wydzialu Matematyki i Informatyki
Uniwersytetu Jagiellońskiego
Kraków (Poland),
December 11th, 2024

Purpose

In this presentation we consider the basics of the tensorial type rules (sometimes known as product rules), for numerical integration over a domain $\Omega \subset \mathbb{R}^d$ via a weighted sum, that is

$$\int_\Omega f(\mathbf{x}) d\Omega \approx \sum_{k=1}^n w_k f(\mathbf{x}_k).$$

These formulas are usually based on univariate rules of Gaussian type, in virtue of all their favourable properties.

We will consider the basic case of domains Ω as

- the hypercube $[-1, 1]^d$;
- the simplex;
- the disk and more general specific domains obtained by linear blending;
- time permitting, we will discuss tensorial rules on the sphere and on some of its subdomains.

For details, see e.g. [2, p.361].

In order to show the basic idea behind this approach, we consider first the example of the sometimes called normal domain.

To introduce this technique, we consider the case of bivariate normal domains

$$\Omega = \{(x, y): \quad a \leq x \leq b, \quad \psi(x) \leq y \leq \phi(x)\},$$

being $\psi, \phi : [a, b] \to \mathbb{R}$ two sufficiently regular functions.

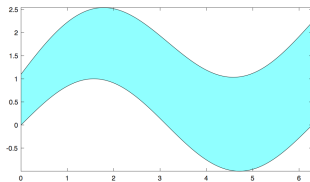
Figure: A normal domain Ω where $a = 0$, $b = 2\pi$, $\psi(x) = \sin(x)$, $\phi(x) = \sin(x) + \log(x + 3)$.

Since

$$\Omega = \{(x, y) : \quad a \le x \le b, \quad \psi(x) \le y \le \phi(x)\},$$

- setting $g(x) := \int_{\psi(x)}^{\phi(x)} f(x, y) \, dy$,
- using the rule $\int_a^b g(x) \, dx \approx \sum_{i=1}^n w_i g(x_i)$,

we have from basic calculus,

$$\begin{aligned}
I(f) : \quad &= \quad \int_\Omega f(\mathbf{x}) d\Omega = \int_a^b \left(\int_{\psi(x)}^{\phi(x)} f(x, y) \, dy \right) dx = \int_a^b g(x) \, dx \\
&\approx \quad \sum_{i=1}^n w_i g(x_i) = \sum_{i=1}^n w_i \int_{\psi(x_i)}^{\phi(x_i)} f(x_i, y) \, dy \qquad (1)
\end{aligned}$$

We observe that we can approximated the *n* inner integrals of

$$I(f) \approx \sum_{i=1}^{n} w_i \int_{\psi(x_i)}^{\phi(x_i)} f(x_i, y) \, dy$$

with a suitable *m*-point rule.

Notice that the domain of the integral may vary with the index "*i*", but that this is not a problem, since we can scale the rule (e.g. one can use a shifted Gauss-Legendre rule, from $[-1, 1]$ to $[\psi(x_i), \phi(x_i)]$).
If

$$\int_{\psi(x_i)}^{\phi(x_i)} f(x_i, y) \, dy \approx \sum_{j=1}^{m} v_{j,i} f(x_i, y_{j,i})$$

we finally get the formula with cardinality *mn*

$$I(f) \approx \sum_{i=1}^{n} w_i \sum_{j=1}^{m} v_{j,i} f(x_i, y_{j,i}) = \sum_{i=1}^{n} \sum_{j=1}^{m} w_i v_{j,i} f(x_i, y_{j,i}) =: S_{m,n}(f)$$

We observe that in the construction of the formula

$$I(f) \approx \sum_{i=1}^{n} w_i \sum_{j=1}^{m} v_{j,i} f(x_i, y_{j,i}) = \sum_{i=1}^{n} \sum_{j=1}^{m} w_i v_{j,i} f(x_i, y_{j,i})$$

we did not make assumptions on the degree of exactness of

$$\int_{a}^{b} g(x) \, dx \approx \sum_{i=1}^{n} w_i g(x_i)$$

and of each

$$\int_{\psi(x_i)}^{\phi(x_i)} f(x_i, y) \, dy \approx \sum_{j=1}^{m} v_{j,i} f(x_i, y_{j,i}).$$

Except for specific cases, e.g. ϕ, ψ polynomials, it will not be possible to choose m, n so to have formulas with a fixed degree of exactness δ, that is $I(p) = S_{m,n}(p)$ for each $p \in \mathbb{P}_n(\Omega)$.

We define some Matlab codes to illustrate these formulas. We start with a routine define_normal_rule that computes the nodes and weights on a normal domain defined by the interval $[a, b]$ and the functions ψ, ϕ.

```matlab
function [nodes,weights]=define_normal_rule(n,m,a,b,psi,phi)

% Rule direction "x".
abn=r_jacobi(n,0,0); xw=gauss(n,abn); % Gauss-Legendre
x=xw(:,1); w=xw(:,2); x=(a+b)/2+(b-a)*x/2; w=(b-a)*w/2;

% Rule direction "y".
abm=r_jacobi(m,0,0); yv=gauss(m,abm); % Gauss-Legendre

% Rule on the normal domain
y=yv(:,1); v=yv(:,2);

nodes=[]; weights=[];

for i=1:n
    psi_i=feval(psi,x(i)); phi_i=feval(phi,x(i));
    y_i=(psi_i+phi_i)/2+((phi_i-psi_i)/2)*y; % scaled nodes
    v_i=((phi_i-psi_i)/2)*v; % scaled weights

    nodes_add=[x(i)*ones(size(y_i)) y_i]; % rule nodes/weights to add
    nodes=[nodes; nodes_add];
    weights_add=w(i)*v_i;
    weights=[weights; weights_add];
end
```

Normal domains

Next we implement a demo, to study the case in which

- $a = 0$, $b = 2\pi$, $\psi(x) = \sin(x)$ and $\phi(x) = \sin(x) + \log(x+3)$;
- the integrand is $f(x, y) = (x + 0.5y)^{10}$ and $I(f) = 234913153.2071612\ldots$.

```matlab
function demo_normal_domain

a=0; b=2*pi; % Define "normal domain".
psi=@(x) sin(x);
phi=@(x) sin(x)+log(x+3);
f=@(x,y) (x+0.5*y).^10; % integrand
Iex=2.349131532071612e+08; % integral computed with high order rule
n=10; m=11; % Define "n", "m" (cardinality of the rules).

% External routine that computes nodes and weights.
[nodes,weights]=define_normal_rule(n,m,a,b,psi,phi);

% Compute integral.
fnodes=feval(f,nodes(:,1),nodes(:,2));
Inum=weights'*feval(f,nodes(:,1),nodes(:,2));

fprintf('\n \t * I : %-1.15e',Inum);
fprintf('\n \t * AE: %-1.3e',abs(Inum-Iex));
fprintf('\n \t * RE: %-1.3e \n',abs(Inum-Iex)/abs(Iex));

% Plot normal domain (external subroutine)
plot_normal_domain(a,b,psi,phi);
plot(nodes(:,1),nodes(:,2),'go','MarkerEdgeColor','k',...
    'MarkerFaceColor','g','MarkerSize',4);
axis equal; axis tight;
hold off;
```

Below we mention the routine for plotting the domain.

```
function plot_normal_domain(a,b,psi,phi)

t=linspace(a,b,1000); t=t';
psi_t=feval(psi,t);
phi_t=feval(phi,t);

pts_bound_low=[t psi_t];
pts_bound_up=[flipud(t) flipud(phi_t)];
pts=[pts_bound_low; pts_bound_up; pts_bound_low(1,:)];

fill(pts(:,1),pts(:,2),'c');
hold on;
```

As numerical results we see that the formula is not exact for degree 10, since the integrand belongs to $\mathbb{P}_{10}$.

```
>> demo_normal_domain

    * I : 2.349132020504614e+08
    * AE: 4.884e+01
    * RE: 2.079e-07
>>
```
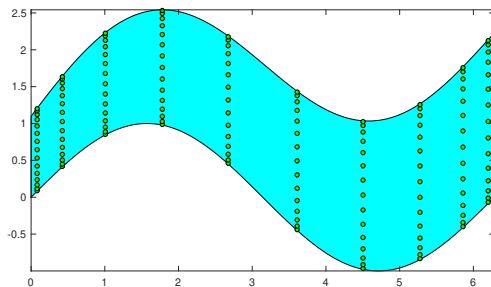
Normal domains

Figure: The normal domain Ω where $a = 0$, $b = 2\pi$, $\psi(x) = \sin(x)$, $\phi(x) = \sin(x) + \log(x + 3)$ and the cubature nodes achieved from the usage of Gauss-Legendre rules in which $n = 10$ and $m = 15$.

- This technique can be used for computing integrals over hypercubes $\Omega = [-1, 1]^d$ (thus, by shifting, also on hyperectangles).
- This time we ask the rule must have a fixed degree of exactness ADE $= \delta$.
- Following the ideas described in the part about normal domains, we adopt a Gauss-Legendre rule

$$\int_{-1}^{1} g(x)dx \approx \sum_{i=1}^{n} w_i g(x_i)$$

with $n = \lceil \frac{\delta+1}{2} \rceil$ nodes, so having at least ADE $= \delta$.

Thus we have

$$
\begin{aligned}
I(f) : \; &= \int_\Omega f(\mathbf{x})d\Omega = \int_{-1}^{1} \ldots \int_{-1}^{1} f(x_1, \ldots, x_d)\, dx_1 \ldots dx_d \\
&\approx \sum_{i_1=1}^{n} \ldots \sum_{i_d=1}^{n} w_{i_1} \ldots w_{i_d} f(x_{i_1}, \ldots, x_{i_d}).
\end{aligned}
\tag{2}
$$

That is a formula with cardinality

$$
\left(\left\lceil \frac{\delta+1}{2} \right\rceil \right)^d \approx \left(\frac{\delta}{2} \right)^d.
$$

Since it grows exponentially with the dimension d, this formula maybe not suitable for d high, causing the so called curse of dimensionality.

For example, if $ADE = \delta = 20$ and $d = 10$, one needs 10^{10} function evaluations (possibly expensive, in view of the number of variables involved).

In general this kind of rules are very used in low dimension (e.g. 2 or 3), but they are not minimal, in the sense that there are rules with much lower number of nodes, sharing the same cardinality.

If $\Omega = [-1, 1]^2$, that is the unit-square, a rule with $ADE = \delta$, in view of Möller lower bound, must have at least cardinality

$$
n_\delta = \begin{cases} \frac{(k+1)(k+2)}{2}, & \delta = 2k \\[2ex] \frac{(k+1)(k+2)}{2} + \lfloor \frac{(k+1)}{2} \rfloor, & \delta = 2k + 1 \end{cases}
$$

and there are rules that go closer to this bound than those of tensorial type.

ADE	MB	AMR	TR
5	7	7	9
10	21	22	36
15	40	46	64
20	66	77	121
25	97	113	169
30	136	166	256
35	180	222	324
40	231	287	441
45	287	361	529
50	351	442	676

Table: Formulas on the unit square. Algebraic degree of exactness ADE, the Möeller bound MB, the cardinality of almost minimal rules AMR and that of tensorial rules TR.

We now approximate certain integrals on the unit-square and unit-cube.

```matlab
function demo_hypercube

ADE=10; d=2; % d is the dimension of the cube

% Define integrand
switch d
    case 2
        f=@(x,y) (0.3*x+0.9*y).^10;  I=5.002201832727280e-01;
    case 3
        f=@(x,y,z) (0.3*x+0.9*y+0.8*z).^10;  I=4.377443514181815e+01;
end

% Gaussian rule with degree ADE.
n=ceil((ADE+1)/2);
abn=r_jacobi(n,0,0); xw=gauss(n,abn); % Gauss-Legendre
x=xw(:,1); w=xw(:,2);
switch d
    case 2
        [x1,x2] = meshgrid(x); [w1,w2] = meshgrid(w);
        fP=feval(f,x1,x2); w=w1.*w2;
        Inum=sum(sum(w.*fP));
    case 3
        [x1,x2,x3] = ndgrid(x); [w1,w2,w3] = ndgrid(w);
        fP=feval(f,x1,x2,x3); w=w1.*w2.*w3;
        Inum=sum(sum(sum(w.*fP)));
end

fprintf('\n \t * I : %1.15e',Inum)
fprintf('\n \t * AE: %1.3e',abs(I-Inum))
fprintf('\n \t * RE: %1.3e \n',abs(I-Inum)/abs(Inum))
```

Hypercubes

1. We consider the case of the formula on $[-1, 1]^2$. It has ADE equal to 10 and we show it integrates exactly (in the numerical sense!) $p(x, y) = (0.3x + 0.9y)^{10}$.

```
>> f=@(x,y) (0.3*x+0.9*y).^10;
>> I=integral2(f,-1,1,-1,1,'AbsTol',10^(-15),'RelTol',10^(-15));
>> format long e
>> I
I =
     5.002201832727280e-01
>> demo_hypercube

   * I : 5.002201832727267e-01
   * AE: 1.332e-15
   * RE: 2.663e-15
>>
```

2. We consider the case of the formula on $[-1, 1]^3$. It has degree 10 and we show it integrates exactly (in the numerical sense!) $p(x, y, z) = (0.3x + 0.9y + 0.8z)^{10}$.

```
>> f=@(x,y,z) (0.3*x+0.9*y+0.8*z).^10;
>> I=integral3(f,-1,1,-1,1,-1,1,'AbsTol',10^(-15),'RelTol',10^(-15));
>> format long; I
I =
   43.774435141818145
>> demo_hypercube

   * I : 4.377443514181813e+01
   * AE: 2.132e-14
   * RE: 4.870e-16
>>
```

A tensorial rule can be also found for the *n*-simplex. For simplicity we shall take into account the case of a triangle, see e.g. [4] for a survey on the topic.

There are many reference triangles $\mathcal{T}$, depending on the purpose. We will consider $\mathcal{T}$ with vertices $(0,0)$, $(1,0)$, $(0,1)$.
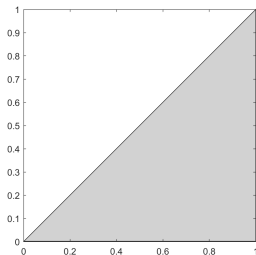

Figure: The reference triangle with vertices $(0,0)$, $(1,0)$, $(0,1)$.

It can be easily seen, setting $y = ux$ and shifting the variables that is $s = 2x - 1$, $t = 2u - 1$

$$
\begin{aligned}
I(f) & := \int_{\mathcal{T}} f(x, y)\, dxdy = \int_0^1 \int_0^x f(x, y)\, dx\, dy \\
& = \int_0^1 x \int_0^1 f(x, xu)\, dx\, du = \ldots = \\
& = \int_{-1}^1 \int_{-1}^1 \frac{1}{8} f\left(\frac{s+1}{2}, \frac{(s+1)(t+1)}{4}\right)(1+s)\, ds\, dt \quad (3)
\end{aligned}
$$

Thus, we have reduced to a certain interval on the square $[-1, 1]^2$.

Setting

$$\phi(s,t) = \frac{1}{8} f\left(\frac{s+1}{2}, \frac{(s+1)(t+1)}{4}\right)$$

we get

$$
\begin{aligned}
I(f) &= \int_{-1}^{1}\int_{-1}^{1} \frac{1}{8} f\left(\frac{s+1}{2}, \frac{(s+1)(t+1)}{4}\right)(1+s)\, ds\, dt \\
&= \int_{-1}^{1}\int_{-1}^{1} \phi(s,t)(1+s)\, ds\, dt \tag{4}
\end{aligned}
$$

Defining

- for the direction s, a Gauss-Jacobi rule with degree of exactness $ADE = \delta$, w.r.t. the weight $(1-s)^0(1+s)^1 = 1+s$,
- for the direction t, a Gauss-Legendre rule with degree of exactness $ADE = \delta$, i.e. w.r.t. the weight $(1-s)^0(1+s)^0$,

we get a formula with positive weights, internal nodes, $ADE = \delta$ on the simplex, with cardinality $(\lceil\frac{\delta+1}{2}\rceil)^2 \approx \frac{\delta^2}{4}$.

Simplex

More precisely, letting

1. $n = \lceil \frac{\delta+1}{2} \rceil$ (number of points of Gaussian rule with ADE=δ),

2. $\phi(s,t) = \frac{1}{8} f\left(\frac{s+1}{2}, \frac{(s+1)}{2} \frac{(t+1)}{4}\right)$,

3. $\int_{-1}^{1} g(s)(1+s)ds = \sum_{i=1}^{n} w_i^{(GJ)} g(x_i^{(GJ)})$,

4. $\int_{-1}^{1} g(t)dt = \sum_{i=1}^{n} w_i^{(GL)} g(x_i^{(GL)})$,

we have

$$
\begin{aligned}
I(f) &= \int_{-1}^{1} \int_{-1}^{1} \phi(s,t)(1+s)\, ds\, dt \\
&\approx \int_{-1}^{1} \sum_{i=1}^{n} w_i^{(GJ)} \phi(x_i^{(GJ)}, t)dt = \sum_{i=1}^{n} w_i^{(GJ)} \int_{-1}^{1} \phi(x_i^{(GJ)}, t)dt \\
&\approx \sum_{i=1}^{n} w_i^{(GJ)} \sum_{j=1}^{n} w_j^{(GL)} \phi(x_i^{(GJ)}, x_j^{(GL)}) \\
&= \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{w_i^{(GL)} w_j^{(GJ)}}{8} f\left(\frac{x_i^{(GJ)}+1}{2}, \frac{x_i^{(GJ)}+1}{2} \cdot \frac{x_j^{(GL)}+1}{2}\right).
\end{aligned}
$$

Note that if $f \in \mathbb{P}_\delta$ then $\approx$ are $=$ and the formula is exact.

As first thing, we define a routine define_rule_simplex that
determines such a formula on the reference simplex.

```
function [nodes,weights]=define_rule_simplex(ade)

% Cubature rule on the unit simplex
% * with vertices (0,0), (1,0), (1,1),
% * with ADE equal to ade.

% Gaussian-Jacobi rule.
m=ceil((ade+1)/2);
ab_GJ=r_jacobi(m,0,1); xw_GJ=gauss(m,ab_GJ); % Gauss-Jacobi
x_GJ=(xw_GJ(:,1)+1)/2; w_GJ=xw_GJ(:,2);

% Gaussian-Legendre rule.
ab_GL=r_jacobi(m,0,0); xw_GL=gauss(m,ab_GL); % Gauss-Legendre
x_GL=(xw_GL(:,1)+1)/2; w_GL=xw_GL(:,2);

% Define tensorial rule
[x_mat_GJ,x_mat_GL]=meshgrid(x_GJ,x_GL);
X_mat=x_mat_GJ; Y_mat=x_mat_GJ.*x_mat_GL;

[w_mat_GJ,w_mat_GL]=meshgrid(w_GJ,w_GL);
W_mat=(1/8)*w_mat_GJ.*w_mat_GL;

nodes=[X_mat(:) Y_mat(:)];
weights=W_mat(:);
```

Next we present a Matlab demo demo_simplex.m in which we test
the polynomial exactness and plot the nodes of the formula.

```matlab
function demo_simplex

ade=10;

f=@(x,y) (0.3*x+0.9*y).^10;
I=6.254277723408297e-02;

% Gaussian rule with degree ADE.
[nodes,weights]=define_rule_simplex(ade);
fP=feval(f,nodes(:,1),nodes(:,2));

Inum=weights'*fP;

% Stats
fprintf('\n \t * ade: %-8.0f',ade)
fprintf('\n \t * #  : %-8.0f',length(weights))
fprintf('\n \t * I  : %-1.15e',Inum)
fprintf('\n \t * AE : %-1.3e',abs(I-Inum))
fprintf('\n \t * RE : %-1.3e \n',abs(I-Inum)/abs(Inum))

gray_color=[211, 211, 211]/256;
fill([0 1 1 0],[0 0 1 0],gray_color);
hold on;
plot(nodes(:,1),nodes(:,2),'go','MarkerEdgeColor','k',...
    'MarkerFaceColor','g','MarkerSize',6);
axis equal; axis tight;
hold off;
```

Simplex

For the computation of the reference value I, we have written the routine exact_integral_simplex.m, based on adaptive procedure `integral2` over a rectangle.

1. As first method for approximating $\int_{\mathcal{T}} f(x, y)\, dxdy$ we take into account an integrand on $[0, 1]^2$, equal to $f \cdot \mathcal{X}_{\mathcal{T}}$, where $\mathcal{X}_{\mathcal{T}}$ is the characteristic function on the simplex $\mathcal{T}$.

2. Alternatively we replaced the desired integral with one on a square, as described in (4).

```
function exact_integral_simplex

f=@(x,y) (0.3*x+0.9*y).^10; % integrand on the simplex

method=2;

switch method
    case 1
        F=@(x,y) f(x,y).*(y <= x);
        I=integral2(F,0,1,0,1,'AbsTol',10^(-15),'RelTol',10^(-15));
    case 2
        F=@(s,t) (1/8)*f( (s+1)/2,(s+1).*(t+1)/4) .*(1+s);
        I=integral2(F,-1,1,-1,1,'AbsTol',10^(-15),'RelTol',10^(-15));
end

fprintf('\n \t I : %1.15e \n ',I)
```

Running the demo we get

```
>> demo_simplex

   * ade: 10
   * #  : 36
   * I  : 6.254277723409099e-02
   * AE : 8.021e-15
   * RE : 1.283e-13
>>
```
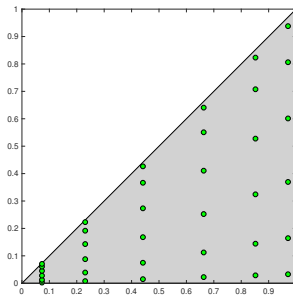

Figure: Nodes of the tensorial rule on the unit-simplex, for $ADE = 10$.

As before, these rules are easily at hand, but they are far from being the best around in terms of cardinality.

For example, at degree 10, the tensorial rule above had 36 nodes, but it is known there is one with these feature having only 24 positive weights and internal nodes (sometime known with the acronym PI type, see table below).

δ	N_δ^*	δ	N_δ^*	δ	N_δ^*	δ	N_δ^*	δ	N_δ^*
1	1	11	27	21	85	31	181	41	309
2	3	12	32	22	93	32	193	42	324
3	4	13	36	23	100	33	204	43	339
4	6	14	42	24	109	34	214	44	354
5	7	15	46	25	117	35	228	45	370
6	11	16	52	26	130	36	243	46	385
7	12	17	57	27	141	37	252	47	399
8	16	18	66	28	150	38	267	48	423
9	19	19	70	29	159	39	282	49	435
10	24	20	78	30	171	40	295	50	453

Table: Cardinality N_δ^* of (almost) minimal rules on triangles with $ADE = \delta$.

Similar rules can be established for the *d*-dimensional unit-ball. For sake of simplicity we restrict our attention to the bivariate unit-disk, i.e. $\Omega \equiv B(0,1)$.

We observe that in this case, after the transformation in polar coordinates, taking into account the determinant of the jacobian matrix,

$$\int_{\Omega} f(\mathbf{x}) \, d\Omega = \int_0^1 \int_0^{2\pi} f(r\cos(\theta), r\sin(\theta)) \cdot r \, d\theta \, dr$$

Notice that the r.h.s. consists of an integral over a rectangle $[a, b] \times [0, 2\pi]$ where the integrand is

$$g(r, \theta) = f(r\cos(\theta), r\sin(\theta)) \cdot r.$$

If f is a polynomial of $\mathbb{R}^2$ of total degree δ then

$$f(x) = \sum_{0 \le i+j \le \delta} a_{i,j} x^i y^j$$

and consequently

$$
\begin{aligned}
f(r\cos(\theta), r\sin(\theta)) \cdot r &= \sum_{0 \le i+j \le \delta} a_{i,j} (r\cos(\theta))^i (r\sin(\theta)^j \, r \\
&= \sum_{0 \le i+j \le \delta} a_{i,j} r^{i+j+1} \cos^i(\theta)) \sin^j(\theta)
\end{aligned}
$$

that is

1. an algebraic polynomial of total degree $\delta + 1$ in the variable r;

2. a trigonometric polynomial of degree δ in the variable θ.

The last point deserves some attention, since usually the trigonometric polynomials q of degree δ, that is $q \in \mathbb{T}_\delta$, are written as

$$q(\theta) = \sum_{j=0}^{\delta} a_j \cos(j\theta) + \sum_{j=1}^{\delta} b_j \sin(j\theta).$$

Now suppose that

- $\int_0^1 p_{\delta+1}(x)dx = \sum_{i=1}^{N_\delta} w_i p_{\delta+1}(x_i) = S_{\delta+1}(p)$, for $p_{\delta+1} \in \mathbb{P}_{\delta+1}$,
- $\int_0^{2\pi} q_\delta(\theta)d\theta = \sum_{i=1}^{M_\delta} v_i q_\delta(\theta_i) = T_\delta(q)$, for $q_\delta \in \mathbb{T}_\delta$,

then if $f(x) = \sum_{0 \le i+j \le \delta} a_{i,j} x^i y^j$ we get

$$
\begin{aligned}
I_\Omega(f) &= \int_0^1 \int_0^{2\pi} f(r\cos(\theta), r\sin(\theta)) \cdot r \, dr \, d\theta \\
&= \int_0^1 \int_0^{2\pi} \sum_{0 \le i+j \le \delta} a_{i,j} r^{i+j+1} \cos^i(\theta) \sin^j(\theta) \, dr \, d\theta \\
&= \sum_{0 \le i+j \le \delta} a_{i,j} (\int_0^1 r^{i+j+1} dr) \cdot (\int_0^{2\pi} \cos^i(\theta) \sin^j(\theta) \, d\theta) \\
&= \sum_{0 \le i+j \le \delta} a_{i,j} S_{\delta+1}(r^i) T_\delta(\cos^i(\theta) \sin^j(\theta)) \quad\quad (5)
\end{aligned}
$$

With some care, it can be seen that the latter is exactly the tensorial rule based on $S_{\delta+1}$ and T_δ applied to approximate $I_\Omega(f)$.

Unit-disk

As rules, a common choice, to get a formula with $ADE = \delta$, is to adopt

- a Gauss-Legendre rule, shifted in $[0, 1]$, with $ADE = \delta + 1$, in the variable "r",

- a trapezoidal rule, on $\delta + 2$ equispaced points, including the extrema, on the angular interval $[0, 2\pi]$, that can be proved to be exact over trigonometric polynomials of degree δ.
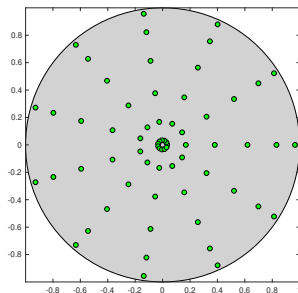

Figure: Nodes of the tensorial rule on the unit-disk, for $\delta = 10$. It has 11 radii equispaced on the angles and 6 points for each radius.

In view of the fact that

- the Gauss-Legendre rule has not nodes at the extrema 0, 1,
- the trapezoidal rule has nodes in 0, 2π and
$$(r\cos(0), r\sin(0)) = (r\cos(2\pi), r\sin(2\pi)),$$

one has

- not to worry that the origin is counted many times as node (since it is not a node!),
- the points of the initial radius is counted twice (being on the initial and final radius), so rearranging the weights, the formula can avoid the points of the final radius.

Thus one can see that such a product rule has cardinality

$$\left\lceil \frac{\delta+1}{2} \right\rceil (\delta+1),$$

inferior or equal to the dimension of the polynomial space $\mathbb{P}_\delta(B(0,1))$ that is

$$\dim(\mathbb{P}_\delta(B(0,1))) = \frac{(\delta+1)(\delta+2)}{2}.$$

Unit-disk

Thus, setting

- $n_\delta = \lceil \frac{\delta+1}{2} \rceil$,
- $m_\delta = \delta + 1$,
- $h = 2\pi/(\delta+1)$, $\theta_j^{(TPZ)} = jh$, $v_j^{(TPZ)} = h$,

we have

$$
\begin{aligned}
I(f) &= \int_0^1 \int_0^{2\pi} f(r\cos(\theta), r\sin(\theta)) \cdot r \, d\theta \, dr \\
&\approx \sum_{i=1}^{n_\delta} w_i^{(GL)} \int_0^{2\pi} f(x_i^{(GL)}\cos(\theta), x_i^{(GL)}\sin(\theta)) \cdot x_i^{(GL)} \, d\theta \\
&= \sum_{i=1}^{n_\delta} w_i^{(GL)} \sum_{j=1}^{m_\delta} v_j^{(TPZ)} f(x_i^{(GL)}\cos(\theta_j^{(TPZ)}), x_i^{(GL)}\sin(\theta_j^{(TPZ)})) \cdot x_i^{(GL)} \\
&= \sum_{i=1}^{n_\delta} \sum_{j=1}^{m_\delta} w_i^{(GL)} v_j^{(TPZ)} x_i^{(GL)} \cdot f(x_i^{(GL)}\cos(\theta_j^{(TPZ)}), x_i^{(GL)}\sin(\theta_j^{(TPZ)})).
\end{aligned}
$$

Unit-disk

In passing, it worths pointing out that exist rules with interior nodes and positive weights, prescribed degrees of exactness δ and very low cardinality.

δ	N_δ^*	δ	N_δ^*
1	1	17	55
2	3	19	72
3	4	25	117
4	6	27	137
5	7	33	199
6	12	37	247
7	12	39	273
8	16	41	295
9	19	45	361
11	28	49	425
13	36	53	487
15	44	65	733

Table: Cardinality N_δ^* of (almost) minimal rules on the unit with $ADE = \delta$. As example, the so designed tensorial rule for degree 19 has 210 points.

In what follows, we implement these tensorial rules and show some numerical examples.

```matlab
function [nodes,weights]=define_rule_disk(ade)

% Cubature rule on the unit disk with ADE equal to ade.
% Note: As output, the nodes will be in cartesian coordinates.

% shifted gaussian rule: r direction
m=ceil((ade+1)/2);
abm=r_jacobi(m,0,0); yv=gauss(m,abm); % Gauss-Legendre
rnodes=(yv(:,1)+1)/2; rw=yv(:,2)/2;

% trapezoidal rule, consider first and last node repetition
N=ade+2;
t=linspace(0,2*pi,N); t=t(1:end-1); t=t';
tw=(2*pi/(N-1))*ones(N-1,1);

% define tensorial rule
[r_mat,th_mat]=meshgrid(rnodes,t);
[r_matw,th_matw]=meshgrid(rw,tw);

x_mat=r_mat.*cos(th_mat); y_mat=r_mat.*sin(th_mat);
w_mat=r_matw.*th_matw.*r_mat;

nodes=[x_mat(:) y_mat(:)]; weights=w_mat(:);
```

Unit-disk

```matlab
function demo_disk

ade=10; % Define "n", "m" (cardinality of the rules).
example=1; % define example

switch example
    case 1
        f=@(x,y) (x+0.5*y).^10; % integrand
        Iex=3.932323797070195e-01 ; % numerically exact integral
    otherwise
        f=@(x,y) (1+x+0.5*y).^11; % integrand
        Iex=5.546261116442703e+02 ; % numerically exact integral
end

% External routine that computes nodes and weights.
[nodes,weights]=define_rule_disk(ade);

% Compute integral.
fnodes=feval(f,nodes(:,1),nodes(:,2));
Inum=weights'*feval(f,nodes(:,1),nodes(:,2));

% Statistics
fprintf('\n \t * # : %-8.0f',length(weights));
fprintf('\n \t * #T: %-8.0f',ceil((ade+1)/2)*(ade+1));
fprintf('\n \t * I : %-1.15e',Inum);
fprintf('\n \t * AE: %-1.3e',abs(Inum-Iex));
fprintf('\n \t * RE: %-1.3e \n',abs(Inum-Iex)/abs(Iex));

% Plot disk and pointset
th=linspace(0,2*pi,100); gray_color=[211, 211, 211]/256;
fill(cos(th),sin(th),gray_color); hold on;
plot(nodes(:,1),nodes(:,2),'go','MarkerEdgeColor','k',...
    'MarkerFaceColor','g','MarkerSize',6);
axis equal; axis tight;
hold off;
```

Unit-disk

1. As first experiment we integrate a polynomial of degree 10, by a rule with ADE equal to 10. To this purpose we set in ade=10 and example=1 in the file demo_disk, getting

```
>> demo_disk
   * # : 66
   * #T: 66
   * I : 3.932323797070124e-01
   * AE: 7.161e-15
   * RE: 1.821e-14
>>
```

2. As second experiment we integrate a polynomial of degree 11, by a rule with ADE equal to 11. To this purpose we set in ade=11 and example=2 in the file demo_disk.m, getting

```
>> demo_disk
   * # : 72
   * #T: 72
   * I : 3.932323797070125e-01
   * AE: 6.994e-15
   * RE: 1.779e-14
>>
```

We observe that the numerical approximation of the desired integral can be obtained by the following adaptive routine,

```
function exact_integral_disk

example=1;

switch example
    case 1
        fpolar=@(r,t) (1*r.*cos(t)+0.5*r.*sin(t)).^10.*r;
        I=integral2(fpolar,0,1,0,2*pi,'AbsTol',10^(-15),'RelTol',10^(-15));
    case 2
        fpolar=@(r,t) (1+1*r.*cos(t)+0.5*r.*sin(t)).^11.*r;
        I=integral2(fpolar,0,1,0,2*pi,'AbsTol',10^(-15),'RelTol',10^(-15));
end

fprintf('\n \t I : %1.15e \n',I)
```

or alternatively, by means of chebfun environment,

```
function exact_integral_disk_chebfun

example=1;

switch example
    case 1
        f=@(x,y) (1*x+0.5*y).^10;
    case 2
        f=@(x,y) (1+1*x+0.5*y).^11;
end

fc=diskfun(f); Ic=sum2(fc);
fprintf('\n \t Ic: %1.15e \n',Ic)
```

One may have doubts about the fact that the trapezoidal rule with $\delta + 2$ nodes integrates exactly a trigonometric polynomial p of degree δ. Let us make an experiment to clarify these ideas.

Since $p(0) = p(2\pi)$, only $\delta + 1$ nodes are needed.

```
function demo_trapzrule

delta=5; % max degree of trig. polynomials

% trapezoidal formula: nodes "x" and weights "h".
x=linspace(0,2*pi,delta+2); x=x(2:end); h=x(2)-x(1);

% cosines to be integrated
for k=0:delta
    f=@(x) cos(k*x);
    S=h*sum(feval(f,x));
    I=integral(f,0,2*pi,"AbsTol",10^(-14),"RelTol",10^(-14));
    AE(k+1)=abs(S-I);
end

% sines to be integrated
for k=1:delta
    f=@(x) sin(k*x);
    S=h*sum(feval(f,x));
    I=integral(f,0,2*pi,"AbsTol",10^(-14),"RelTol",10^(-14));
    AE(end+1)=abs(S-I);
end

fprintf('\n \t max ae: %1.3e \n \n', max(AE));
```

By the previous demo we test that the trapezoidal rule, integrates correctly in $[0, 2\pi]$ a basis of the trigonometric polynomials of degree $\delta = 5$, i.e.

$$\{\cos(kx)\}_{k=0,\dots,\delta} \cup \{\sin(kx)\}_{k=1,\dots,\delta}$$

and consequently, by the linearity of integral operator, any trigonometric polynomial of degree δ.

Numerically we get

```
>> demo_trapzrule
   max ae: 2.939e-15
>>
```

that is, we have approximated close to machine precision all the integrals on a trigonometric basis of degree $\delta = 5$.

We investigate the case of circular regions that can be obtained by the so called linear blending of elliptical arcs.

Let two elliptical arcs defined respictively by

$$
\begin{aligned}
P(\theta) &= A_1 \cos(\theta) + B_1 \sin(\theta) + C_1, \\
Q(\theta) &= A_2 \cos(\theta) + B_2 \sin(\theta) + C_2,
\end{aligned}
$$

where $\theta \in [\alpha, \beta]$, $0 \leq \beta - \alpha \leq 2\pi$ and

$$
A_i = (a_{i1}, a_{i2}), \quad B_i = (b_{i1}, b_{i2}), \quad C_i = (c_{i1}, c_{i2}), \quad i = 1, 2.
$$

The region

$$
\Omega = \{(x, y) = U(t, \theta) = tP(\theta) + (1 - t)Q(\theta), \ (t, \theta) \in [0, 1] \times [\alpha, \beta]\}
$$

is known as linear blending of elliptical arcs.

Linear blending on ell. arcs: circular segments

For a better understanding, we make some examples.

Example

Set in (6)

1. $A_1 = (r, 0)$, $B_1 = (0, r)$, $C_1 = (0, 0)$,
2. $A_2 = (r, 0)$, $B_2 = (0, -r)$, $C_2 = (0, 0)$.

and consider the interval $[0, \beta]$ with $0 < \beta \leq \pi$.

Since

- $P(\theta) = A_1 \cos(\theta) + B_1 \sin(\theta) + C_1$,
- $Q(\theta) = A_2 \cos(\theta) + B_2 \sin(\theta) + C_2$,

we have in particular

$P(\theta) = (r\cos(\theta) + 0 \cdot \sin(\theta) + 0, 0 \cdot \cos(\theta) + r \cdot \sin(\theta) + 0) = (r\cos(\theta), r\sin(\theta))$,

$Q(\theta) = (r\cos(\theta) + 0 \cdot \sin(\theta) + 0, 0 \cdot \cos(\theta) - r \cdot \sin(\theta) + 0) = (r\cos(\theta), -r\sin(\theta))$.

The regions that we obtain are circular segments. In particular for $\beta = \pi$ we get the unit-disk.
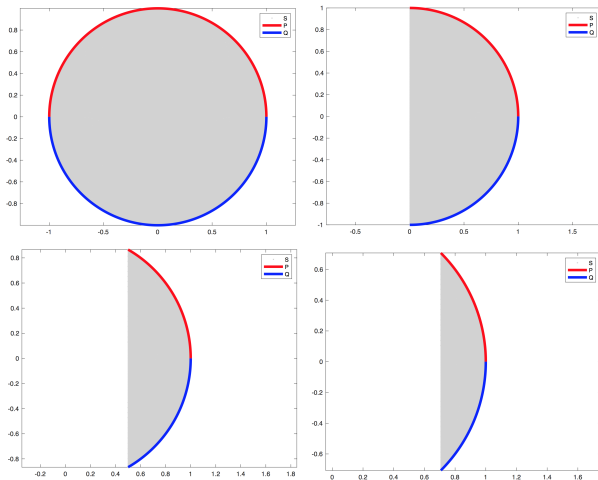
Figure: Example 1, with $\beta = \pi$, $\beta = \pi/2$, $\beta = \pi/3$, $\beta = \pi/4$.

Example

Set in (6)

1. $A_1 = (0,0)$, $B_1 = (0,0)$, $C_1 = (0,0)$,
2. $A_2 = (r,0)$, $B_2 = (0,r)$, $C_2 = (0,0)$.

and consider the interval $[\alpha, \beta]$ with $0 < \beta - \alpha \le 2\pi$.

Since

- $P(\theta) = A_1 \cos(\theta) + B_1 \sin(\theta) + C_1$,
- $Q(\theta) = A_2 \cos(\theta) + B_2 \sin(\theta) + C_2$,

we have in particular

$$P(\theta) = (0\cos(\theta) + 0 \cdot \sin(\theta) + 0, 0 \cdot \cos(\theta) + 0 \cdot \sin(\theta) + 0) = (0,0),$$

$$Q(\theta) = (r\cos(\theta) + 0 \cdot \sin(\theta) + 0, 0 \cdot \cos(\theta) + r \cdot \sin(\theta) + 0) = (r\cos(\theta)\, r\sin(\theta)).$$

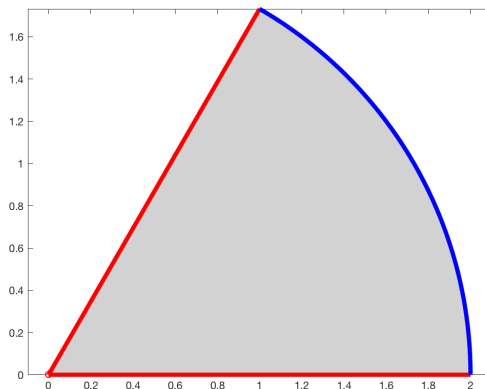The regions that we obtain are sectors.

Figure: Circular sector, with $r = 1$, $\alpha = 0$, $\beta = \pi/3$.

As one may understand, depending on the parameters, many other circular regions can be defined as

- Symmetric or asymmetric sectors or annuli,
- circular zones,
- circular lenses,
- butterfly-shaped and candy-shaped regions.

See [6] for more details.

Trigonometric quadrature

A cubature rule of tensorial type in these domains is based on trigonometric gaussian formula with degree of exactness n. For details see [3].

Let $\mathbb{T}_\delta = span_{k=0,\ldots,\delta}(\cos(k\theta), \sin(k\theta))$ the trigonometric polynomials of degree δ.

Theorem (Trigonometric Gaussian quadrature formula on subintervals)

Let $\{(\xi_j, \lambda_j)\}_{j=1,\ldots,\delta+1}$, be the nodes and positive weights of the algebraic Gaussian quadrature formula for the weight function

$$w(x) = \frac{2\sin(\omega/2)}{\sqrt{1 - x^2 \sin^2(\omega/2)}}, \; x \in (-1,1), \; \omega \in (0, \pi].$$

Then for $0 < \beta - \alpha \leq 2\pi$ the following trigonometric Gaussian quadrature formula on $[\alpha, \beta]$ *holds for any $p \in \mathbb{P}_\delta$*

$$\int_\alpha^\beta p(\theta) \, d\theta = \sum_{j=1}^{\delta+1} \lambda_j p(\theta_j + \mu), \; \mu = \frac{\alpha + \beta}{2},$$

where

$$\theta_j = 2\arcsin(\xi_j \sin(\omega/2)) \in (-\omega, \omega), \; j = 1, 2, \ldots, \delta+1, \; \omega = \frac{\beta - \alpha}{2}.$$

The formula is implemented in the Matlab procedure trigauss.

As demo, we provide a rule for integrating $f \in \mathbb{T}_{10}$ on $[0, \pi/6]$, where

$$f(\theta) = (\cos(2\,t) - 0.5\,\sin(t) + 0.2)^5.$$

```matlab
function demo_trigauss
% Demo on quadrature of trigonometric polynomials over intervals [alpha, beta].

deg=10;
f=@(t) (cos(2*t)-0.5*sin(t)+0.2).^5;
alpha=0; beta=pi/6;

IR=integral(f,alpha,beta,'AbsTol',10^(-14),'RelTol',10^(-14));

tw=trigauss(deg,alpha,beta);
ft=feval(f,tw(:,1));
w=tw(:,2);
Inum=w'*ft;

fprintf('\n \t IR    :%-1.15e',IR);
fprintf('\n \t Inum :%-1.15e',Inum);
fprintf('\n \t AE    :%-1.3e',abs(Inum-IR));
fprintf('\n \t RE    :%-1.3e',abs(Inum-IR)/abs(IR));
fprintf('\n \n');
```

We get a rule with 11 points, for which we get what follows.

```
>> demo_trigauss

   IR    :4.875668241566165e-01
   Inum :4.875668241566168e-01
   AE    :3.331e-16
   RE    :6.831e-16
```

Now observe that if

$$\Omega = \{(x, y) = U(t, \theta) = tP(\theta) + (1 - t)Q(\theta),\ (t, \theta) \in [0, 1] \times [\alpha, \beta]\}$$

then by the injectivity of U

$$I(f) \quad = \quad \int_\Omega f(x, y)dxdy = \int_0^1 f(U(t, \theta))|\det(JU(t, \theta))|\, dt\, d\theta \qquad (6)$$

where one can see,

- after some analysis, that $|\det(JU(t, \theta))|$ is a mixed algebraic-trigonometric polynomial belonging to the tensor-product space $\mathbb{P}_h([0, 1]) \otimes \mathbb{T}_k([\alpha, \beta])$;

- if $f \in \mathbb{P}_\delta$ then the integrand in (6) belongs to $\mathbb{P}_{\delta+h}([0, 1]) \otimes \mathbb{T}_{\delta+k}([\alpha, \beta])$

and thus one can apply a tensorial rule based on

- Gauss-Legendre rule of degree $\delta + h$ in $[0, 1]$,

- trigonometric Gauss rule of degree $\delta + k$ in $[\alpha, \beta]$,

to get a formula with degree δ over Ω.

Theorem

Consider the planar domain generated by linear blending of two parametrics arcs

$$\Omega = \{(x, y) = U(t, \theta) = tP(\theta) + (1 - t)Q(\theta) \in [0, 1] \times [\alpha, \beta], \quad 0 < \beta - \alpha \le 2\pi\}$$

where

$$P(\theta) = A_1 \cos(\theta) + B_1 \sin(\theta) + C_1, \quad Q(\theta) = A_2 \cos(\theta) + B_2 \sin(\theta) + C_2, \quad (7)$$

in which $\theta \in [\alpha, \beta]$, $0 \le \beta - \alpha \le 2\pi$ and

$$A_i = (a_{i1}, a_{i2}), \quad B_i = (b_{i1}, b_{i2}), \quad C_i = (c_{i1}, c_{i2}), \quad i = 1, 2.$$

Assume that the transformation U is injective for $(t, \theta) \in (0, 1) \times (\alpha, \beta)$, and let

$$
\begin{aligned}
u_0 &= (a_{11} - a_{21})(b_{12} - b_{22}) + (a_{12} - a_{22})(b_{21} - b_{11}) \\
u_1 &= (b_{12} - b_{22})(c_{11} - c_{21}) + (b_{21} - b_{11})(c_{12} - c_{22}) \\
u_2 &= (a_{11} - a_{21})(c_{12} - c_{22}) + (a_{12} - a_{22})(c_{21} - c_{11}) \\
v_0 &= b_{21}(a_{22} - a_{12}) + b_{22}(a_{11} - a_{21}), \quad v_1 = b_{21}(c_{22} - c_{12}) + b_{22}(c_{11} - c_{21}) \\
v_2 &= a_{21}(c_{12} - c_{22}) + a_{22}(c_{21} - c_{11}), \quad v_3 = a_{12}a_{21} - a_{11}a_{22} + b_{11}b_{22} - b_{12}b_{21} \\
v_4 &= a_{12}b_{21} - a_{11}b_{22} + a_{21}b_{12} - a_{22}b_{11}
\end{aligned}
$$

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

$$\int_{\mathcal{S}} f(x,y)\,dxdy = \sum_{j=1}^{\delta+k+1} \sum_{i=1}^{\lceil (\delta+h+1)/2 \rceil} W_{ij} f(x_{ij}, y_{ij})$$

for each $f \in \mathbb{P}^2_\delta$, with

- $h = 0$ when $u_1 = u_2 = u_3 = 0$, $h = 1$ otherwise;
- $k = 0$ when $u_1 = u_2 = v_1 = v_2 = v_3 = v_4 = 0$;
- $k = 1$ when $v_3 = v_4 = 0$ and one among u_1, u_2, v_1, v_2 is nonzero;
- $k = 2$ if one among v_3, v_4 is nonzero;
- $(x_{ij}, y_{i,j}) = U(t_i^{GL}, \theta_j + \mu)$,
- $0 < W_{ij} = |\det(JU(t_i^{GL}, \theta_j + \mu))|w_i^{GL}\lambda_j$, where
    - $\{(\theta_j + \mu, \lambda_j)\}$ are the angular nodes and weights of the trigonometric gaussian formula of degree of exactness $\delta + k$ on $[\alpha, \beta]$;
    - $\{(t_i^{GL}, w_i^{GL})\}$ are the nodes and weights of Gauss-Legendre formula of degree of exactness $\delta + h$ on $[0, 1]$;

- The theorem above is a little involved, in view of all the possible choices of the parameters.
- The core is that depending on A_i, B_i, C_i, $i = 1, 2$, one can determine some nonnegative integers h and k, so that by
  1. a trigonometric rule on $[\alpha, \beta]$ of degree $\delta + k$,
  2. a Gauss-Legendre rule on $[0, 1]$ of degree $\delta + h$,

  one gets a tensorial rule with a prescribed degree of precision δ, on the linear blending (of elliptical arcs) domain,

$$\Omega = \{(x, y) = U(t, \theta) = tP(\theta) + (1-t)Q(\theta), \ (t, \theta) \in [0, 1] \times [\alpha, \beta]\}$$

- The cardinality is of the rule is of order $\delta^2/2$.
- Depending on the linear blending, the pertinent formula is available by means of the Matlab procedure gqellblend.

As numerical experiment we compute formulas on a sector described as

- $A_1 = (1, 0)$, $B_1 = (0, 1)$, $C_1 = (0, 0)$,
- $A_2 = (1, 0)$, $B_2 = (0, -1)$, $C_2 = (0, 0)$,
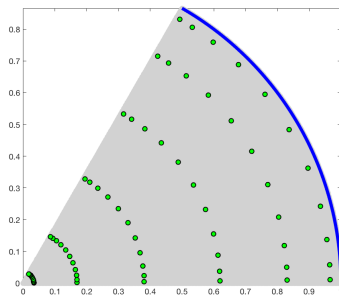- $\alpha = 0$, $\beta = \pi/3$,

and integrate some polynomials.

Figure: A circular sector and its cubature nodes for $ADE = 10$.

```matlab
function demo_linear_blending
% Object: Example of integration over a sector.
% Settings.
ade=10; f_example=1;
% Main code below.
switch f_example
    case 1
        f=@(x,y) (x+0.5*y).^10; % integrand
        Iex=1.792695693383881e-01; % numerically exact integral
    otherwise
        f=@(x,y) (1+x+0.5*y).^11; % integrand
        Iex=4.313845631275915e+02; % numerically exact integral
end
% Domain.
beta=pi/3; alpha=0; % examples: pi/4, pi/3, pi/2, pi.
r1=0; r2=1;
A1=[r1 0]; B1=[0 r1]; C1=[0 0];
A2=[r2 0]; B2=[0 r2]; C2=[0 0];
% Cubature formula
A=[A1; A2]; B=[B1; B2]; C=[C1; C2];
xyw = gqellblend(ade,A,B,C,alpha,beta); x=xyw(:,1); y=xyw(:,2); w=xyw(:,3);
% Integral computation.
fxy=feval(f,x,y); Inum=w'*fxy;
% Statistics
fprintf('\n \t * # : %-8.0f',length(w));
fprintf('\n \t * I : %-1.15e',Inum);
fprintf('\n \t * AE: %-1.3e',abs(Inum-Iex));
fprintf('\n \t * RE: %-1.3e \n',abs(Inum-Iex)/abs(Iex));
% Plot arcs
plot_linear_blending(A1,B1,C1,A2,B2,C2,alpha,beta);
hold on;
plot(x,y,'go','MarkerEdgeColor','k','MarkerFaceColor','g','MarkerSize',6);
axis equal; axis tight;hold off;
```

```matlab
function plot_linear_blending(A1,B1,C1,A2,B2,C2,alpha,beta)

% The parameters A1,B1,C1,A2,B2,C2 are row vectors "1 x 2".
% 0<beta-alpha<=2*pi.

theta=linspace(alpha,beta,300); theta=theta';

% Plot first arc.
P=bsxfun(@times,A1,cos(theta))+bsxfun(@times,B1,sin(theta))+...
    bsxfun(@times,C1,ones(size(theta)));

Q=bsxfun(@times,A2,cos(theta))+bsxfun(@times,B2,sin(theta))+...
    bsxfun(@times,C2,ones(size(theta)));

t=linspace(0,1,300); t=t';
S=[];
for k=1:size(P,1)
    SLOC=bsxfun(@times,P(k,:),t)+bsxfun(@times,Q(k,:),1-t); S=[S; SLOC];
end

% Plot
gray_color=[211, 211, 211]/256;
plot(S(:,1),S(:,2),'o','color',gray_color,'MarkerEdgeColor',gray_color,...
                        'MarkerFaceColor',gray_color,...
                        'MarkerSize',6);
hold on;
    plot(P(:,1),P(:,2),'r-','LineWidth',4);
    plot(Q(:,1),Q(:,2),'b-','LineWidth',4);
axis equal
hold off;
```

1. As first experiment we integrate a polynomial of degree 10, by a rule with ADE equal to 10. To this purpose we set in `ade=10` and `example=1` in the file demo_linear_blending, getting

```
>> demo_linear_blending
   * # : 66
   * I : 1.792695693383878e-01
   * AE: 2.776e-16
   * RE: 1.548e-15
>>
```

2. As second experiment we integrate a polynomial of degree 11, by a rule with ADE equal to 11. To this purpose we set in `ade=11` and `example=2` in the file demo_linear_blending.m, getting

```
>> demo_linear_blending
   * # : 84
   * I : 4.313845631275914e+02
   * AE: 1.137e-13
   * RE: 2.635e-16
>>
```

If we finally use a formula with degree 10, we get instead an unsatisfactory result, too far from machine precision.

```
>> demo_linear_blending
   * # : 66
   * I : 4.313845629175062e+02
   * AE: 2.101e-07
   * RE: 4.870e-10
>>
```

Let us suppose that we have to compute

$$I(f) := \int_{\mathbb{S}^2} f(\boldsymbol{\eta}) dS^2(\boldsymbol{\eta})$$

with $f \in C(\mathbb{S}^2)$.

A first technique consists in reducing $I(f)$ to an integral over a certain rectangle and then apply suitable tensorial rules.

To this purpose, consider the spherical coordinates

$$\boldsymbol{\eta} \to (\cos\phi\sin\theta, \sin\phi\sin\theta, \cos\theta), \ \ 0 \le \phi \le 2\pi, \ \ 0 \le \theta \le \pi.$$

Thus, taking into account the jacobian determinant of the transformation, we get

$$I(f) = \int_0^{2\pi} \int_0^{\pi} f(\cos\phi\sin\theta, \sin\phi\sin\theta, \cos\theta) \sin\theta \ d\theta d\phi.$$

Aiming to determine

$$I(f) = \int_0^{2\pi} \int_0^\pi f(\cos\phi\sin\theta, \sin\phi\sin\theta, \cos\theta)\sin\theta \ d\theta d\phi.$$

in view of the periodicity in the variable ϕ, we apply the composite trapezoidal rule with uniform spacing, that is

$$\tilde{I}(g) = \int_0^{2\pi} g(\phi)d\phi \approx \tilde{I}_m(g) = \sum_{j=0}^m {}''g(jh), \ \ h = \frac{2\pi}{m}$$

where $''$ means the first and last argument of the sum must be halved.

Taking into account the periodicity of the integrand, it is immediate that

$$\tilde{I}_m(g) = \sum_{j=0}^m {}''g(jh) = \frac{g(0)}{2} + \sum_{j=1}^{m-1} g(jh) + \frac{g(2\pi)}{2} = \sum_{j=1}^m g(jh).$$

The integral of

$$I(f) = \int_0^{2\pi} \int_0^{\pi} f(\cos\phi\sin\theta, \sin\phi\sin\theta, \cos\theta)\sin\theta \; d\theta d\phi.$$

w.r.t. the variable θ is more problematic. Setting $z = \cos(\theta)$, we have

$$I(f) = \int_0^{2\pi} \int_{-1}^{1} f(\cos\phi\sqrt{1-z^2}, \sin\phi\sqrt{1-z^2}, z) \; dz \, d\phi.$$

At his point one can apply

- Gauss-Legendre quadrature over $[-1, 1]$ with n nodes $\{z_k\}_{k=1,\dots,n}$ and weights $\{w_k\}_{k=1,\dots,n}$;
- the trapezoidal rule with nodes $\phi_j = j\frac{\pi}{n}$, $j = 1, \dots, 2n$ and weights $h = \frac{\pi}{n}$.

After these substitutions we get

$$I_n(f) = h \sum_{j=1}^{2n-1} \sum_{k=1}^{n} w_k f(cos\phi_j \cdot \sqrt{1-z_k^2}, \sin\phi_j \cdot \sqrt{1-z_k^2}, z_k)$$

thus for $\theta_k = \arccos z_k$

$$I_n(f) = h \sum_{j=1}^{2n-1} \sum_{k=1}^{n} w_k f(cos\phi_j \cdot \sin\theta_k, \sin\phi_j \cdot \sin\theta_k, \cos\theta_k)$$

The following result holds, setting $\mathbb{P}_{2n-1}$ the set of polynomials on the sphere of degree at most $2n-1$.

Theorem (Atkinson, Han, p. 169)

If $f \in \mathbb{P}_{2n-1}$ then $I(f) = I_n(f)$. If $f(x, y, z) = z^{2n}$ we have that $I(f) \neq I_n(f)$.

In other words the formula has degree of exactness $\delta = 2n - 1$.

Next, introducing the best approximation error at degree n, w.r.t. uniform norm,

$$E_n(f) = \min_{p \in \mathbb{P}_n} \|f - p\|_\infty$$

one can prove that

$$|I(f) - I_n(f)| \le 8\pi E_{2n-1}(f).$$

The following result holds

Theorem (Atkinson, Han, p.141)

Let $r \ge 1$ be an integer. Assume f is r-times continuously differentiable over $\mathbb{S}^2$. with all such derivatives in $C(\mathbb{S}^2)$. Then

$$E_n(f) \le \frac{c}{(n+1)^r}.$$

Supposing that we have at hand the Matlab codes

- `rjacobi`,
- `gauss`,

that implement the Gaussian rules w.r.t. a Jacobi weight, we intend to test the product Gauss rule on the computation of the integrals

1 $I_1(f) := \int_{\mathbb{S}^2} \exp(x) dS^2 \approx 14.76801374576529$;

2 $I_2(f) := \int_{\mathbb{S}^2} \exp(-x^2 - 0.1 * y^2 - 2 * z^2); dS^2 \approx 5.028153009823267$;

3 $I_3(f) := \int_{\mathbb{S}^2} (0.1 x + y^2 + 0.5 z^3)^3 dS^2 \approx 1.969928193450972$;

To this purpose we define the routines

- gaussian_product_rule.m that computes nodes and weight of the rule for a fixed n,
- demo_product_gaussian_rule.m that tests the results on the approximation of $I(f)$, by means of product Gauss rules for $n = 2, 3, 4, 5, 6$.

The routine gaussian_product_rule.m implements the tensorial rule described above.

```
function [nodes,w]=gaussian_product_rule(n)

% Gaussian product rule of degree 2n-1.
% Nodes are in cartesian coordinates.

% Trapezoidal rule
m=2*n-1; h=(2*pi/m); phi=(1:m)'*h;

% Gauss-Legendre rule
ab=r_jacobi(n,0,0); xw=gauss(n,ab);

% Nodes
z=xw(:,1);
[P,T]=meshgrid(phi,sqrt(1-z.^2));
[PZ,Z]=meshgrid(phi,z);
P=P(:); T=T(:); Z=Z(:);
nodes=[cos(P).*T sin(P).*T Z];

% Weights
hv=h*ones(m,1);
[W1,W2]=meshgrid(hv,xw(:,2));
w=W1(:).*W2(:);
```

```
function demo_product_gaussian_rule(example)
% Demo: see
% K.Atkinson, W.Han, Spherical Harmonics and Approximations on the Unit
% Sphere: An introduction, p.172.
if nargin < 1, example=3; end
nv=2:6;
I=[];
switch example
    case 1
        f=@(x,y,z) exp(x)+0*y+0*z;
        I=1.476801374576529e+01;
    case 2
        f=@(x,y,z) exp(-x.^2-0.1*y.^2-2*z.^2);
        I=5.028153009823267;
     case 3
        f=@(x,y,z) (0.1*x+y.^2+0.5*z.^3).^3;
        I=1.969928193450972;
end

if isempty(I), F=spherefun(f); I=sum2(F); format long; I, end
AE=[]; RE=[];

for n=nv
    [nodes,w]=gaussian_product_rule(n);
    x=nodes(:,1); y=nodes(:,2); z=nodes(:,3);
    fnodes=feval(f,x,y,z);

    In=w'*fnodes;
    AE(end+1)=abs(In-I); RE(end+1)=abs(In-I)/(abs(I)+(I == 0));

    fprintf('\n \t n: %3.0f nodes: %5.0f AE: %1.2e RE: %1.2e',...
        n,length(x),AE(end),RE(end))
end
fprintf('\n \n');
```

Unit-Sphere

The routine demo_product_gaussian_rule.m provides the following results, choosing the variable example from 1 to 3.

```
>> demo_product_gaussian_rule(1)

  n:    2 nodes:      8 AE: 1.17e-02 RE: 7.94e-04
  n:    3 nodes:     18 AE: 4.00e-04 RE: 2.71e-05
  n:    4 nodes:     32 AE: 4.91e-07 RE: 3.32e-08
  n:    5 nodes:     50 AE: 3.84e-09 RE: 2.60e-10
  n:    6 nodes:     72 AE: 2.21e-12 RE: 1.50e-13

>> demo_product_gaussian_rule(2)

  n:    2 nodes:      8 AE: 3.54e-01 RE: 7.04e-02
  n:    3 nodes:     18 AE: 4.77e-02 RE: 9.49e-03
  n:    4 nodes:     32 AE: 5.71e-03 RE: 1.13e-03
  n:    5 nodes:     50 AE: 5.04e-04 RE: 1.00e-04
  n:    6 nodes:     72 AE: 3.71e-05 RE: 7.38e-06

>> demo_product_gaussian_rule(3)

  n:    2 nodes:      8 AE: 8.11e-03 RE: 4.12e-03
  n:    3 nodes:     18 AE: 2.21e-02 RE: 1.12e-02
  n:    4 nodes:     32 AE: 2.74e-02 RE: 1.39e-02
  n:    5 nodes:     50 AE: 0.00e+00 RE: 0.00e+00
  n:    6 nodes:     72 AE: 4.44e-16 RE: 2.25e-16

>>
```

The third function is a polynomial of degree 9 and the formula has degree of exactness $2n-1$, so per $N = 5$, $n = 6$ the relative errors are close to 0.

In view of the rotational symmetry of the sphere, one prefers points that are not clustered to the poles, but due to the structure of Gauss-Legendre rules, this set does.
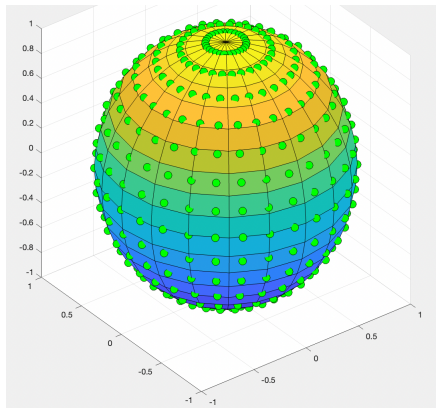

Figure: Gauss Product Rule for $n = 15$.

Spherical rectangles

Let
$$\mathcal{R} = [a_1, b_1] \times [a_2, b_2] \subseteq [0, \pi] \times [0, 2\pi]$$

be a rectangle, and define as spherical rectangle $\Omega_\mathcal{R}$ (sometimes also known as geographical rectangle) the subdomain of the sphere $\mathbb{S}^2$ whose points are of the form

$$P = \xi(\theta, \phi) := (\cos\phi\sin\theta, \sin\phi\sin\theta, \cos\theta), \quad (\theta, \phi) \in \mathcal{R}.$$

We observe that depending on $\mathcal{R}$, several well-known subdomains $\Omega_\mathcal{R} = \xi(\mathcal{R})$ of the 2-sphere can be defined in this way, as

- caps,
- collars,
- slices,
- more generally spherical rectangles defined by longitudes and latitudes.
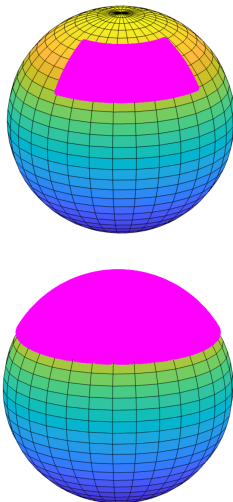
Spherical rectangles

Figure: Above and below: a spherical rectangle $\xi([\pi/6, \pi/3] \times [0, \pi/2])$ and a spherical cap $\xi([0, \pi/3] \times [0, 2\pi])$.
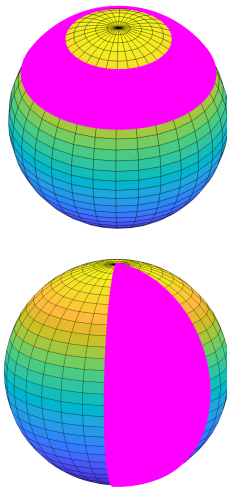
Figure: Above and below: a (spherical) collar $\xi([\pi/6, \pi/3] \times [0, 2\pi])$ and a (spherical) slice $\xi([0, \pi] \times [0, \pi/3])$.

To this purpose, we introduce the following result

Theorem

Let $w \in [0, \pi]$.

- $\mathbb{T}_\delta([-\omega, \omega]) = span\{1, \cos(k\theta), \sin(k\theta)\}, 1 \leq k \leq \delta, \theta \in [-\omega, \omega]$,

- $w : [-\omega, \omega] \to \mathbb{R}$ be a symmetric weight function,

- $\{\xi_j\}_{j=1,\ldots,\delta+1}$, $\{\lambda_j\}_{j=1,\ldots,\delta+1}$ be respectively the nodes and the weights of an algebraic gaussian rule relatively to the symmetric weight function

$$\tilde{s}(x) = w(2\arcsin(\sin(\omega/2)x)) \frac{2\sin(\omega/2)}{\sqrt{1 - \sin^2(\omega/2)\, x^2}}, x \in (-1, 1).$$

Then

$$\int_{-\omega}^{\omega} f(\theta)w(\theta)\, d\theta = \sum_{j=1}^{\delta+1} \lambda_j f(\theta_j), f \in \mathbb{T}_n([-\omega, \omega]) \qquad (8)$$

where $\theta_j = 2\arcsin(\sin(\omega/2)\xi_j) \in (-\omega, \omega),\ j = 1, \ldots, \delta + 1.$

This theorem says that if we intend to integrate

$$\int_{\omega}^{\omega} f(\theta)w(\theta)d\theta$$

where f is a trigonometric polynomial of degree δ and w a symmetric weight function in $[-\omega, \omega]$ then it is sufficient to

- compute the nodes $\{\xi_i\}$ and weights $\{\lambda_i\}$ of a gaussian rule with δ nodes w.r.t. a certain weight function $\tilde{s}$;
- modify the nodes $\{\xi_i\}$ into $\{\theta_i\}$ by a simple transformation.

The hidden difficulty is that the computation of these formula is not trivial since the weight function is a little unusual.

Spherical rectangles

We provide an algebraic rule over the spherical rectangle $\Omega_{\mathcal{R}} \subseteq \mathbb{S}^2$.

Theorem

Let

- Ω_R *a spherical rectangle, where* $\mathcal{R} = [a_1, b_1] \times [a_2, b_2] \subseteq [0, \pi] \times [0, 2\pi]$;
- $\{\theta_k^{[a_1,b_1]}\}_{k=1,\ldots,\delta+2}$ *and* $\{\lambda_k^{[a_1,b_1]}\}_{k=1,\ldots,\delta+2}$ *be the nodes and the weights of a gaussian subperiodic trigonometric rule on* $[a_1, b_1]$ *w.r.t.* $w(x) = 1$, *having trigonometric degree of precision* $n + 1$;
- $\{\theta_k^{[a_2,b_2]}\}_{k=1,\ldots,\delta+1}$ *and* $\{\lambda_k^{[a_2,b_2]}\}_{k=1,\ldots,n+1}$ *be the nodes and the weights of a gaussian subperiodic trigonometric rule on* $[a_2, b_2]$ *w.r.t.* $w(x) = 1$, *having trigonometric degree of precision* n.

Then the tensorial cubature rule

$$S_n(f) = \sum_{i_1=1}^{\delta+2} \sum_{i_2=1}^{\delta+1} \lambda_{i_1,i_2} f(\xi_{i_1,i_2})$$

$$\xi_{i_1,i_2} = \xi(\theta_{i_1}^{[a_1,b_1]}, \theta_{i_2}^{[a_2,b_2]}), \quad \lambda_{i_1,i_2} = \lambda_{i_1} \lambda_{i_2} \sin{(\theta_{i_1}^{[a_1,b_1]})}$$

integrates exactly in $\Omega_{\mathcal{R}}$ *every algebraic polynomial of total degree* δ.

Spherical rectangles

The previous theorem from [5] seems complicated, but it comes directly from observing that similarly to the tensorial rule on the unit sphere

$$l_{\Omega_{\mathcal{R}}} = \int_{a_1}^{b_1} \int_{a_2}^{b_2} f(\cos\theta_2 \sin\theta_1, \sin\theta_2 \sin\theta_1, \cos\theta_1) \sin\theta_1 \, d\theta_1 d\theta_2.$$

and thus if f is a polynomial of total degree δ then the integrand on the r.h.s. is

- a trigonometric polynomial of degree $\delta + 1$ in the variable θ_1;
- a trigonometric polynomial of degree δ in the variable θ_2;

and thus to provide a tensorial rule of the desired degree we need

- a formula with trigonometric degree $\delta + 1$ in the variable θ_1 (ranging in $[a_1, b_1]$);
- a formula with trigonometric degree δ in the variable θ_2 (ranging in $[a_1, b_1]$);

The rest is the determination of a tensorial rule by means of univariate rules to obtain the desired formula.

Remark (Caps)

The cardinality of these rules is $\approx \delta^2$, where δ is the degree of precision. With some tricks, one can have a formula on the spherical cap with $\approx \delta^2/2$ points.

Remark (Software)

Though at first sight the result of the theorem is a little complicated, in practice when one provides the nodes and the weights of the subperiodic formula (not easy!), everything become simpler.

As for the numerical software, see the Matlab package Cubature rules on spherical rectangles.

Spherical rectangles

As numerical tests, we consider the cubature of the functions

$$
\begin{aligned}
f_1(\mathbf{x}) &= \exp\left(-x^2 - 100\, y^2 - 0.5\, z^2\right), \\
f_2(\mathbf{x}) &= \sin\left(-x^2 - 100\, y^2 - 0.5\, z^2\right), \\
f_3(\mathbf{x}) &= \max(1/4 - ((x - 1/\sqrt{5})^2 + (y - 2/\sqrt{5})^2 + (z - 2/\sqrt{5})^2), 0))^3
\end{aligned}
$$

on the spherical rectangle

$$
\Omega_{\mathcal{R}} = \xi(\mathcal{R}), \quad \mathcal{R} = [\pi/6,\, \pi/3] \times [0,\, \pi/2].
$$
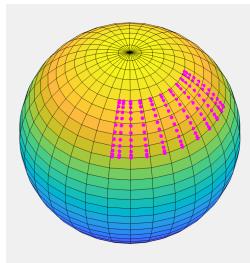

Figure: The spherical rectangle $\xi([\pi/6, \pi/3] \times [0, \pi/2])$ and the nodes of the rule of degree of exactness $\delta = 10$.

Spherical rectangles

Deg.	f_1	f_2	f_3
5	$3.34e-04$	$7.38e-02$	$4.53e-06$
10	$4.89e-06$	$2.69e-02$	$5.44e-07$
15	$9.12e-09$	$5.14e-03$	$4.07e-08$
20	$1.76e-10$	$1.13e-02$	$2.43e-08$
25	$7.73e-14$	$1.13e-02$	$9.53e-09$
30	$3.33e-16$	$1.23e-03$	$2.23e-09$
35	$3.47e-17$	$2.58e-05$	$2.33e-09$
40	$1.14e-16$	$1.96e-07$	$2.82e-10$
45	$3.47e-17$	$6.94e-10$	$8.84e-10$
50	$2.08e-17$	$1.33e-12$	$5.48e-11$

Table: Absolute errors for degrees $5, 10, \ldots, 50$, w.r.t. the integrals on the spherical rectangle $\xi([\pi/6, \pi/3] \times [0, \pi/2])$ on the test functions f_1, f_2, f_3.

Bibliography

R. Cools and K. J. Kim, A survey of known and new cubature formulas for the unit disk, Korean Journal of Computational and Applied Mathematics, Volume 7 (2000), pp. 477–485.

P.J. Davis and P. Rabinowitz, Methods of Numerical Integration, Dover 1984.

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

J.N. Lyness, R. Cools, W. Gautschi, A survey of numerical cubature over triangles, Mathematics of Computation 1943-1993: A Half-Century of Computational Mathematics. pp. 127-150, (1994).

M. Gentile, A. Sommariva, M. Vianello, Polynomial approximation and quadrature on geographic rectangles, Appl. Math. Comput. 297 (2017), pp. 159–179.

A. Sommariva and M. Vianello, Polynomial fitting and interpolation on circular sections, Applied Mathematics and Computation, Volume 25 (2015), pp.410–424.