Cheap and stable quadrature

Alvise Sommariva

International Workshop on Modern Problems of Analysis, Optimization, Approximation and Their Applications

Rome (I), June, 26-27, 2025

Introduction

In this talk we will briefly discuss cheap numerical cubature on multivariate domains

- introduce some basics on the topic and theoretical results;
- show the numerical advantages of this approach.

Joint work with Laura Rinaldi and Marco Vianello.

Work partially supported by

- the DOR funds of the University of Padova,
- INdAM-GNCS 2024 Project "Kernel and polynomial methods for approximation and integration: theory and application software",
- INdAM-GNCS 2025 Project "GNCS, Polinomi, Splines e Funzioni Kernel: dall'Approssimazione Numerica al Software Open-Source".

Research accomplished within

- the RITA,
- the SIMAI Activity Group ANA&A,
- the Community of Practice "Green Computing" of the Arqus European University Alliance.

We intend to numerically approximate

$$\int_{\Omega} f(x) d\Omega \approx \sum_{i=1}^{N_M} w_i f(P_i).$$

where

- Ω is a domain of $\mathbb{R}^2$ or $\mathbb{R}^3$,
- $f \in C(\Omega)$.

by a formula that has algebraic degree of exactness M, that is

$$\int_{\Omega} f(x) d\Omega = \sum_{i=1}^{N_M} w_i f(P_i)$$

whenever $f \in \mathbb{P}_M$, i.e. is an algebraic polynomial of degree M.

Remark

Later, ADE is the algebraic degree of exactness M of the formula.

- The formula
$$\int_{\Omega} f(x) d\Omega \approx \sum_{i=1}^{N_M} w_i f(P_i).$$
is allowed to have nodes P_i external to Ω;
- some weights may be negative, but the index of stability named conditioning of the cubature formula
$$\text{cond}(\{w_i\}) := \frac{\sum_{i=1}^{N_M} |w_i|}{\sum_{i=1}^{N_M} w_i}$$
tends to 1 when increasing the ADE M;
- the determination of the nodes $\{P_i\}_{i=1,\dots,N_M}$ and the weights $\{w_i\}_{i=1,\dots,N_M}$ is fast;
- the latter does not require the solution of a linear system.

Our approach is based on the following:

- *compute a polynomial hyperinterpolant of the integrand f on a hypercube containining the domain, e.g. a bounding box, (by means of some rule with $ADE = 2M$, w.r.t. some weight function);*
- *integrate the hyperinterpolant on the domain.*

Remark (Nodes and weights)

As in the case of classical interpolatory rules, these ideas can be converted in determining nodes and weights in the bounding box so that the rule has degree of exactness M on the integration domain Ω.

Remark (Hyperinterpolation)

Hyperinterpolation is a Fourier-like orthogonal projection on a total-degree polynomial space with respect to an absolutely continuous measure, discretized by an algebraic quadrature formula with positive weights.

Our approach is based on the following:

- *compute a polynomial hyperinterpolant of the integrand f on a hypercube containining the domain, e.g. a bounding box, (by means of some rule with $ADE = 2M$, w.r.t. some weight function);*
- *integrate the hyperinterpolant on the domain Ω.*

Remark

As in the case of classical interpolatory rules, these ideas can be converted in determining nodes and weights in the bounding box so that the rule has degree of exactness M on the integration domain Ω .

Suppose that the integration domain Ω is a polyhedron. Determine:

1. a Cartesian bounding box for the polyhedron;
2. the nodes $\{P_i\}$ and weights $\mathbf{u} = \{u_i\}$, $1 \leq i \leq N_M$, of a cubature formula exact for $\mathbb{P}_{2M}$ for a given absolutely continuous measure $d\mu = \sigma(P)dP$ on the bounding box;
3. an orthonormal basis $\{\phi_1, \ldots, \phi_\nu\}$ of $\mathbb{P}_M$ with respect to $d\mu$,
4. the corresponding Lebesgue moments $\mathbf{m} = \{m_1, \ldots, m_N\}$, $m_j = \int_\Omega \phi_j(P)\, dP$, e.g. by the divergence theorem;
5. the Vandermonde-like matrix

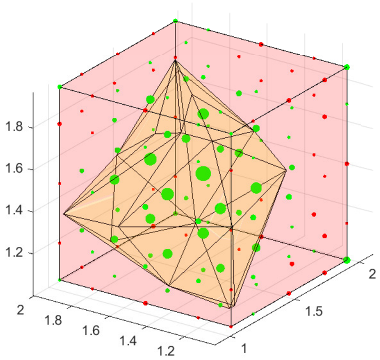
$$V = V_n(\{P_i\}) = [\phi_j(P_i)] \in \mathbb{R}^{N_M \times \nu}$$

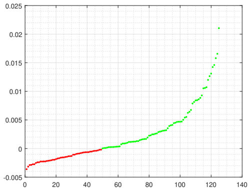
6. compute the weights as

$$\mathbf{w} = diag(\mathbf{u})\, V\mathbf{m}\ ,$$

or in a Matlab-like notation $\mathbf{w} = \mathbf{u}.*\ V\mathbf{m}$.

Example





- On the left: quadrature nodes of the cheap rule on the nonconvex polyhedral element Ω_1 for degree 4, as red dots if the pertinent weight is negative, as green dots otherwise. The size of the dots is visually proportional to the weight magnitude.
- On the right, distribution of the weights in increasing order (in red: negative weights, in green: positive weights). We report that the smaller weight is $w_{\text{min}} \approx -3.6 \cdot 10^{-3}$, the larger is $w_{\text{max}} \approx 2.1 \cdot 10^{-2}$, and the smaller size is $|w|_{\text{min}} \approx 2.6 \cdot 10^{-5}$.

Pros:

- Application to polytopal FEM (fast and stable computation of the integrals of products of polynomials naturally arising on arbitrary polyhedral elements, avoiding sub-tessellation into tetrahedra);
- moment computation does not require tesselation;
- many computations can be done just once and repeated on different integration domains;
- w.r.t. techniques based on approximate Fekete points
  - cubature stability is ensured;
  - no QR factorization or linear system solution is involved.

Cons:

- Though stability is ensured some weights may be negative;
- the integrands require in general evaluations outside the integration domain.

Theoretical results

Theorem (Asymptotical optimal stability)

1. *Let $\Omega \subset \mathbb{R}^d$ be a compact subset, μ an absolutely continuous measure on Ω with respect to the Lebesgue measure.*
2. *Denote by $\{\phi_j\}_{1 \le j \le \nu}$ an orthonormal polynomial basis of $\mathbb{P}_M$ for μ.*
3. *Let $(X, \mathbf{u}) = (\{P_i\}, \{u_i)\}), 1 \le i \le N_M$, be the nodes and positive weights of a quadrature formula for integration in $d\mu$, exact on $\mathbb{P}_{2M}$ (the polynomials with total degree not exceeding $2M$), and $h \in L^2_\mu(K)$.*

Then, the following algebraic product-like formula holds

$$\int_\Omega h(P) f(P)\, d\mu = \sum_{i=1}^{N_M} w_i\, f(P_i) \;,\; \forall f \in \mathbb{P}_n \;, \tag{1}$$

where the quadrature weights $\{w_i\}$ are defined by the product-like moments

$$w_i = u_i \sum_{j=1}^{\nu} \phi_j(P_i)\, m_j \;, 1 \le i \le N_M \;, \quad m_j = \int_\Omega \phi_j(P)\, h(P)\, d\mu \;. \tag{2}$$

Moreover, the formula is stable, since

$$\lim_{M \to \infty} \sum_{i=1}^{N_M} |w_i| = \int_\Omega |h(P)|\, d\mu \;. \tag{3}$$

- We have implemented all these ideas in Matlab and in Python.
- in a forthcoming paper we extend this approach to other multivariate domains as
  - bivariate domains whose boundary can be tracked by parametric splines,
  - multivariate domains with complicated gerometries in which moments are computed by Quasi-Montecarlo methods.

In this last section, we give some hints on what has been done over polyhedra.

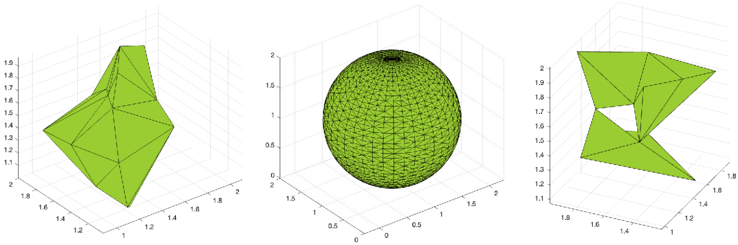


Figure: Examples of polyhedral domains. Left: Ω_1 (nonconvex, 20 facets); Center: Ω_2 (convex, 760 facets); Right: Ω_3 (multiply connected, 20 facets).

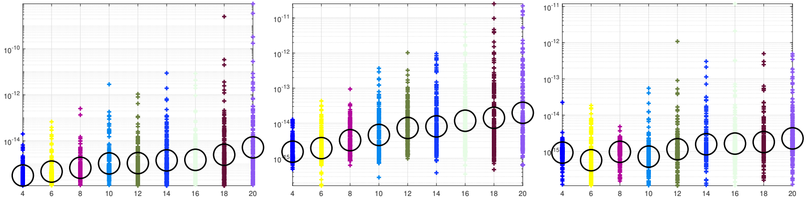


Figure: Relative errors $E(g_k)$ of the tetrahedra-free rule over 200 polynomial integrands of the form $g_k = (a_k x + b_k y + c_k z + d_k)^n$ on the three polyhedra of Figure 1, where a_k, b_k, c_k, d_k are uniform random coefficients in $[-1, 1]$ and $n = 4, 6, 8, \ldots, 20$; the circles correspond to the average logarithmic error $\sum_{k=1}^{200} \log(E(g_k))/200$.

deg	4	6	8	10	12	14	16	18	20
Ω_1	1.2e-03	1.4e-03	1.7e-03	2.3e-03	3.4e-03	5.1e-03	7.7e-03	1.9e-02	3.4e-02
Ω_2	3.0e-02	3.4e-02	4.3e-02	5.9e-02	8.2e-02	1.2e-01	1.8e-01	4.4e-01	9.7e-01
Ω_3	8.1e-04	9.0e-04	1.1e-03	1.7e-03	2.3e-03	3.5e-03	5.4e-03	1.3e-02	2.6e-02

Table: Average cputimes (in seconds) of *CheapQ* on the domains of Fig. 1, varying the algebraic degree of exactness.

deg n	4	6	8	10	12	14	16	18	20
Ω_1	1.55	1.40	1.30	1.25	1.23	1.21	1.19	1.17	1.17
Ω_2	1.30	1.14	1.21	1.12	1.13	1.12	1.10	1.10	1.09
Ω_3	1.63	1.81	1.89	1.86	1.82	1.79	1.74	1.67	1.63

Table: Ratios $\sum_{j=1}^{\nu} |w_j| / vol(\Omega_i)$ for *CheapQ* on the domains of Fig. 1, varying the algebraic degree of exactness.

Bibliography

- C. Langlois, T. van Putten, H. Bériot, E. Deckers, Frugal numerical integration scheme for polytopal domains, Eng. Comput. (2024). $\rightarrow$ A `cheap` technique on polyhedral elements based on Approximate Fekete points.
- A. Sommariva and M. Vianello, *Cheap and stable quadrature on polyhedral elements*, Finite Elements in Analysis and Design, 2025 (to appear); $\rightarrow$ Introduction of `cheap` technique on polyhedral elements.
- L. Rinaldi, A. Sommariva and M. Vianello, *A new computational paradigm for cheap and stable discretized measure compression*, in preparation; $\rightarrow$ `Cheap` technique on general elements.
- A. Sommariva, M. Vianello, TetraFreeQ: tetrahedra-free quadrature on polyhedral elements, Appl. Numer. Math. 200 (2024), 389–398. $\rightarrow$ Computation of polynomial moments on polyhedra, without tesselation.