Weighted Quadrature Rules for Finite Element Methods

Saulo P. Oliveira[1], Alexandre L. Madureira[2] and Frederic Valentin[2]

[1] LAGEP-CPGG, Salvador-BA, Brazil. saulopo@cpgg.ufba.br.
[2] LNCC, Petrópolis-RJ, Brazil. alm@lncc.br, valentin@lncc.br.

Abstract. We discuss the numerical integration of polynomials times exponential weighting functions arising from multiscale finite element computations. The new rules are more accurate than standard quadratures and are better suited to existing codes than formulas computed by symbolic integration. We test our approach in a multiscale finite element method for the 2D reaction-diffusion equation.

Standard finite elements usually fail to accurately solve equations with multiscale behavior. This can happen if coefficients are oscillatory or if a small parameter multiplies some of the terms in the equation. A strategy to overcome these difficulties is to use special spaces instead of the space of piecewise polynomial functions [1, 2]. However, for polynomial basis functions, standard quadratures are *exact* and this is not the case for more complicated spaces. We investigate quadratures to integrate elementwise products of polynomials and exponential basis functions. Such integrals appear when developing enriched methods for reaction-advection-diffusion equations [2], but also in other contexts [3]. Quadrature formulas are simpler to implement into existing finite element codes than results of symbolic integrations.

We define an N-point *weighted quadrature* in $[a, b]$ with *weighting function* w by a set of *integration weights* A_l and *integration points* $x_l \in [a, b]$ such that

$$\int_a^b q(x)w(x)\,dx \approx \sum_{j=1}^{N} A_l q(x_l) \tag{1}$$

for a given function q. The *Newton-Cotes rule*, one of the simplest quadratures of degree of precision n, is defined by choosing $x_l = a + (l-1)(b-a)/n$ and

$$A_l = \prod_{\substack{i=1 \\ i\neq l}}^{n+1} \int_a^b \frac{(x - x_i)}{(x_l - x_i)} w(x)\,dx, \quad l = 1, \ldots, n+1\ . \tag{2}$$

The *Gaussian* quadrature uses integration weights defined as in (2), but the integration points are the roots of the n-th degree polynomial p satisfying

$$\int_a^b p(x)q(x)w(x)\,dx = 0 \quad \forall\ q \text{ of degree } \leq n\ . \tag{3}$$

The Gaussian quadrature has the *optimal* degree of precision $2n-1$. However, Gaussian quadratures may not be the best choice when performing weighted integrals in finite element codes, since the quadrature points may change from element to element. Newton-Cotes rules are sub-optimal, but allow one to fix the quadrature points and re-calculate only the quadrature weights.

We employ one-dimensional quadratures to approximate weighted integrals over quadrilateral regions. Using isoparametric maps, such integrals can be transformed into integrals in the reference square $[-1,1]^2$. Assuming that $w(x,y) = w_x(x)w_y(y)$ and that $f(x,y)$ is a polynomial function of degree at most $2N-1$ in both x and y, we have that

$$\int_{-1}^{1}\int_{-1}^{1} f(x,y)w(x,y)\,dxdy = \sum_{j=1}^{N}\sum_{k=1}^{N} A_j^x A_k^y f(x_j,y_k) \ , \tag{4}$$

where x_l and A_l^x are the integration points and weights for the 1D Gaussian quadrature with respect to w_x (similarly for y). The above rule is referred to as a *product rule*. For instance, let $\hat{\phi}_1(x) = (1-x)/2$ and $w_x(x) = \exp[-a_x(1-\hat{\phi}_1(x))]$, $a_x > 0$. The integration weights A_j^x for the nine-point Newton-Cotes rule with

$$x_1 = y_1 = -1/3, \qquad x_2 = y_2 = 0, \qquad x_3 = y_3 = 1/3 \ , \tag{5}$$

are found by replacing (5) and $w(x) = \exp[-a_x(1-\hat{\phi}_1(x))]$ into (2). In particular,

$$A_3^x = 6\frac{12 - a_x(7-2a_x) - (12 + a_x(5+a_x))e^{-a_x}}{a_x^3} \ .$$

Replacing a_x by a_y in the equations above yields the definition of A_j^y. The Gaussian rule with similar degree of precision has two points in either direction (a total of four points); the orthogonal polynomial that generates the integration points associated to the weight function $w_x(x)$ is

$$p_2(t) = 8 + a^2 - 8at + a^2t^2 - 2a\frac{a(8+a^2-2at) - 2(4-at)\sinh(a)}{2 + a^2 - 2\cosh(a)} \ .$$

For the sake of illustration, we plot the point locations as we vary a_y, keeping $a_x = 10$. We choose $w(x,y) = w_x(x)w_y(y)$. We plot in Fig. 1 the Gaussian points for $a_y = 1, 10, 100$. We also plot the points of the Newton–Cotes quadrature, which do not depend neither on a_x, nor on a_y.

Quadratures in Triangular Regions

Optimal quadratures for triangles rely on two-dimensional orthogonal polynomials or on the solution of non-linear systems [4]. Similarly to quadrilaterals, integrals in arbitrary triangles can be transformed into integrals in the triangle with vertices $(0,0)$, $(0,1)$ and $(1,0)$. However, the limits of integration in

$$I_w(f) := \int_0^1 \int_0^{1-x} f(x,y)w(x,y)\,dydx, \quad w(x,y) = w_x(x)w_y(y) \ . \tag{6}$$

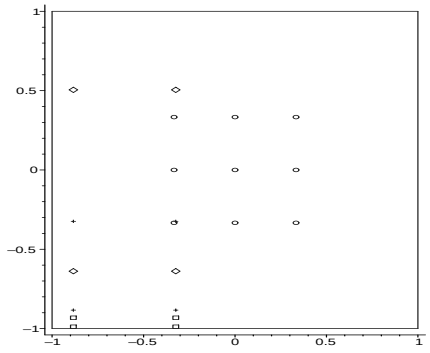

Fig. 1. Gaussian points with $a_x = 10$. Diamonds correspond to $a_y = 1$, crosses to $a_y = 10$, and squares to $a_y = 100$. The fixed circles correspond to Newton–Cotes.

prevent the direct use of product rules. We consider next integrals of the form (6) with $w(x, y) = e^{-ax-by}$, where a and b are postive numbers.

Let us start with a three-point Newton–Cotes Rule. Each integration weight A_k can be found by integrating the Lagrange interpolation polynomial associated to the point $\boldsymbol{p}_k = (x_k, y_k)$ as in (2). Given the points $\boldsymbol{p}_1 = (1/2, 1/2)$, $\boldsymbol{p}_2 = (0, 1/2)$, and $\boldsymbol{p}_3 = (1/2, 0)$, we have $A_k = b_k/(a^2(a-b)b^2)$, where

$$
\begin{aligned}
b_1 &= e^{-a}(2+a)b^2 - a^2(2+b)e^{-b} - (a(b-2)-2b)(a-b),\\
b_2 &= (a-2)(a-b)^2 + (2+b-a)a^2e^{-b} - (a^2 - a(b-4) - 2b)be^{-a},\\
b_3 &= (b-2)(b-a)^2 + (2+a-b)b^2e^{-a} - (b^2 - b(a-4) - 2a)ae^{-b}\ .
\end{aligned}
$$

The Gaussian quadrature of degree of precision one easily follows from the equation $I_w(f) = A_1 f(x_1, y_1)$. Making $f = 1$ yields $A_1 = I_w(1)$; x_1 and y_1 follow from choosing $f = x$ and $f = y$, i.e., $x_1 = I_w(x)/A_1$, and $y_1 = I_w(y)/A_1$. If $w(x, y) = e^{-ax-by}$, then $A_1 = [b(1-e^{-a}) - a(1-e^{-b})]/[a(b-a)b]$, while

$$
\begin{aligned}
x_1 &= [(a-b)^2 + b((a-b)(1+a)+a)e^{-a} - a^2e^{-b}]/[a^2(b-a)^2bA_1],\\
y_1 &= [(a-b)^2 - a((a-b)(1+b)-b)e^{-b} - b^2e^{-a}]/[a(b-a)^2b^2A_1]\ .
\end{aligned}
$$

Application: a Multiscale Finite Element

Let us consider the linear reaction-diffusion problem

$$
-\varepsilon\,\Delta u + \sigma\, u = f \quad \text{in } \Omega \subset \mathbb{R}^2, \quad u = 0 \quad \text{on } \partial\Omega\ , \tag{7}
$$

where $\sigma, \varepsilon > 0$. To approximate (7), we discretize Ω by a conforming and regular partition using triangular elements K and select the finite dimensional subspace $V_h(\Omega) \subset H_0^1(\Omega)$ of piecewise linear polynomials. We seek $u_h \in V_h(\Omega)$ such that

$$
\varepsilon \int_\Omega \nabla u_h \cdot \nabla v_h \, d\boldsymbol{x} + \sigma \int_\Omega u_h\, v_h \, d\boldsymbol{x} = \int_\Omega f\, v_h \, d\boldsymbol{x} \quad \forall v_h \in V_h(\Omega)\ . \tag{8}
$$

The classical Galerkin method just described is inadequate if $\varepsilon \ll \sigma h_K^2$, where h_K denotes the characteristic length of element K. The method lacks stability, and non-physical oscillations appear in the numerical solution. Such issue is treated in [2] by enriching the trial space $V_h(\Omega)$ with *multi-scale functions* $\lambda(\boldsymbol{x}) = \sinh(\alpha_K\,\psi(\boldsymbol{x}))/\sinh(\alpha_K)$, where $\alpha_K \sim h_K(\sigma/\varepsilon)^{1/2}$ is the Peclet number, and $\psi(\boldsymbol{x})$ are piecewise linear shape functions. Thus we need to accurately compute

$$\int_K \lambda(\boldsymbol{x})\,\psi(\boldsymbol{x})\,d\boldsymbol{x}, \qquad \int_K \nabla\lambda(\boldsymbol{x})\,\nabla\psi(\boldsymbol{x})\,d\boldsymbol{x}\ .$$

The above integrals can be written in the form presented in the previous section.

Let the domain Ω be the unit square, which we discretize by a non-uniform mesh of 400 elements. We impose the boundary conditions $u(x,0) = u(0,y) = 0$ and $u(x,1) = u(1,y) = 1$. We set $\sigma = 1$, $f = 0$, and $\varepsilon = 10^{-6}$. The three-point Newton-Cotes rule allows us to conserve all desirable properties of the multi-scale method unlike the classical one-point Gauss, which leads to a loss of accuracy similar to the one observed through the Galerkin method (Fig. 2).

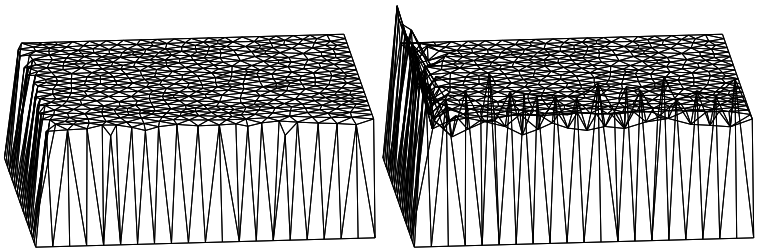

Fig. 2. Solutions by the new exponential-adaptative integration formula (left) and standard (non-weighted) one-point Gauss integration (right).

References

1. I. Babuška, U. Banerjee, and J. Osborn. Generalized finite element methods — main ideas, results and perspectives. *Int. J. Comput. Methods*, 1:67–103, 2004.
2. L. Franca, A. Madureira, and F. Valentin. Towards multiscale functions: Enriching finite element spaces with local but not bubble-like functions. *Comput. Methods Appl. Mech. Engrg.*, 60:3006–3021, 2005.
3. R. Sacco and M. Stynes. Finite element methods for convection-diffusion problems using exponential splines on triangles. *Computers Math. Applic.*, 35:35–45, 1998.
4. A. H. Stroud. *Approximate Calculation of Multiple Integrals.* Prentice-Hall, Englewood Cliffs, 1971.