Second-order convex *maximum entropy* approximants with applications to high-order PDE

A. Rosolen, D. Millán and M. Arroyo*,†

Laboratori de Càlcul Numèric, Universitat Politècnica de Catalunya-BarcelonaTech, E-08034 Barcelona, Spain

SUMMARY

We present a new approach for second-order maximum entropy (*max-ent*) meshfree approximants that produces positive and smooth basis functions of uniform aspect ratio even for nonuniform node sets and prescribes robustly feasible constraints for the entropy maximization program defining the approximants. We examine the performance of the proposed approximation scheme in the numerical solution by a direct Galerkin method of a number of PDEs, including structural vibrations, elliptic second-order PDEs, and fourth-order PDEs for Kirchhoff–Love thin shells and for a phase field model describing the mechanics of biomembranes. The examples highlight the ability of the method to deal with nonuniform node distributions and the high accuracy of the solutions. Surprisingly, the first-order meshfree *max-ent* approximants with large supports are competitive when compared with the proposed second-order approach in all the tested examples, even in the higher order PDEs.

1. INTRODUCTION

Smooth nonnegative approximants are attracting much attention in recent times, notably with the success of isogeometric analysis, because of their excellent behavior in a number of applications including the numerical approximation of structural vibrations and dynamics, where spurious optical modes are suppressed or alleviated, phase field models, where monotonous profiles are obtained, or flow problems (see [1] and the references therein). In general, for partial differential equations (PDEs) with smooth solutions, much more accurate approximations as compared with standard finite elements are obtained for a given computational cost. Convex meshfree maximum entropy (*max-ent*) basis functions, that is, nonnegative approximants with up to linear consistency [2], also exhibit these properties (*max-ent* approximants can be easily made C^∞) and can naturally deal with nonuniform and unstructured refinement in any space dimension. This contrasts with the rigidity of isogeometric analysis, despite recent advances in structured local refinement [3]. When compared with other meshfree methods, the basis functions are smoother, and essential boundary conditions are easier to impose. On the downside, the extension to high-order consistency of convex meshfree approximants is not straightforward, particularly for nonuniform node grids, and in general, meshfree approximations cannot represent accurately by themselves complex domains with possibly curved boundaries. We address here the first limitation and present elsewhere [4] a method to overcome the second by coupling convex meshfree basis functions with CAD-based basis functions.

After a brief review of approximants selected by maximum entropy in Section 2, we present in Section 3 a new approach for second-order *max-ent* meshfree approximants that maintains

*Correspondence to: M. Arroyo, Laboratori de Càlcul Numèric, Universitat Politècnica de Catalunya-BarcelonaTech, E-08034 Barcelona, Spain.
†E-mail: marino.arroyo@upc.edu

the positivity and smoothness of the basis functions (in contrast to [5] and usual moving least squares (MLS) approximants), produces basis functions of uniform aspect ratio even for nonuniform node sets (in contrast to [6, 7]), and prescribes robustly feasible constraints for the entropy maximization program defining the basis functions. We examine the performance of the proposed approximation scheme in the numerical solution of a number of PDEs in Section 4. When analyzing structural vibrations, the proposed second-order *max-ent* meshfree approximants outperform quintic B-splines, and even the first-order local *max-ent* approximants [2] achieve comparable accuracy. We analyze a Poisson problem with localized sources to investigate the ability of the second-order method to exhibit optimal convergence rates for nonuniform adapted grids. We finally exercise the model in high-order PDEs: the Kirchhoff–Love thin shell equations and a fourth-order phase field model for the mechanics of biomembranes, which develops very sharp gradients. Very efficient simulations of this model relying on the adaptive capabilities of the meshfree approximants are presented. Final remarks are collected in Section 5.

2. REVIEW OF MAXIMUM ENTROPY APPROXIMANTS

Recently, Sukumar [8] and Arroyo and Ortiz [2] have proposed an information theoretical viewpoint to define function approximation schemes for polygonal finite elements and meshfree methods, respectively. Expressing the approximation of a function as a linear combination of nodal data associated to a set of scattered points $X = \{x_a\}_{a=1,\ldots,N} \subset \mathbb{R}^d$ and basis functions, $u(x) \approx u^h(x) = \sum_{a=1}^{N} p_a(x) \, u_a$, we view now the basis functions as unknowns and devise a theory to select them rationally. As usual, we require these functions to satisfy the zeroth-order and first-order consistency conditions, that is, $\sum_{a=1}^{N} p_a(x) = 1$ and $\sum_{a=1}^{N} p_a(x) \, x_a = x$, so that affine functions are reproduced exactly. As a crucial extra requirement on the basis functions, we impose their nonnegativity $p_a(x) \geq 0$. This constraint limits seriously the design of approximation schemes, as we show later, but also endows them with the mathematical structure of convex geometry and with practical advantages with respect to other methods that do not have this constraint, such as MLS. The nonnegativity requirement also allows us to interpret, at each point x, the basis functions $p_a(x)$ as a discrete probability distribution for a scheme with N events associated to the nodes. With this interpretation, the function approximation at any point $u^h(x)$ is the statistical expectation of the random variable u_a.

With the statistical interpretation at hand, we view the definition of the basis functions as a statistical inference problem and resort to information theory. The uncertainty or information entropy of a discrete probability can be canonically defined by means of Shannon's entropy [9, 10] $H(p_1, p_2, \ldots, p_N) = -\sum_{a=1}^{N} p_a \ln p_a$. Jaynes [11] proposed the principle of maximum entropy to infer the probabilities when partial information is known, for example, in the form of moments of the distribution. This principle postulates the probability distribution best representing that the current state of knowledge, devoid of biases or artifacts, is the maximizer of the information entropy subject to the constraints given by the partial information. The principle of maximum entropy has found many applications in a wide range of disciplines in science and engineering [12]. When defining meshfree approximation schemes, it proves convenient to introduce the spacial correlations into the inference principle, so that at x, nodes nearby have a larger probability. This bias can be introduced by either resorting to the concept of relative entropy [5, 13] with local prior basis functions (generally Sheppard's approximants) or simply biasing the entropy with a locality function as in local *max-ent* (LME). These are the Pareto optima for the conflicting objectives of locality and information theoretical optimality (entropy maximization) [2] and provide a precise control on the support size of the basis functions. They are the solutions to the convex optimization program

$$(\text{LME}) \quad \text{For fixed } x \text{ minimize} \quad \sum_{a=1}^{N} \beta_a p_a |x - x_a|^2 + \sum_{a=1}^{N} p_a \ln p_a$$

$$\text{subject to} \quad p_a \geq 0, \quad a = 1, \ldots, N$$

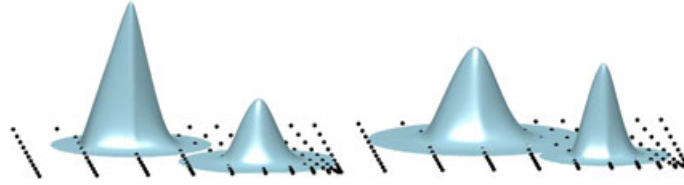$$\sum_{a=1}^{N} p_a = 1, \qquad \sum_{a=1}^{N} p_a(x - x_a) = \mathbf{0},$$

Figure 1. Local *max-ent* basis functions for a nonuniform nodal distribution computed with (left) constant β_a and (right) constant γ_a. The relation between these two parameters is $\gamma_a = \beta_a h_a^2$, where h_a denotes the typical nodal spacing close to the ath node.

where the nonnegative parameters β_a weigh the relative importance given to each objective. Note that the first-order consistency condition has been centered using the partition of unity constraint. The program is feasible if and only if x is in the convex hull of the node set conv X. Duality methods provide an almost explicit expression for the basis functions, where only an unconstrained, strictly convex, smooth optimization problem of dimension d needs to be solved at each evaluation point [2]. When locality overwhelmingly dominates entropy, the approximates were shown to converge to the Delaunay affine basis functions. On the basis of the implicit function theorem, it was proven that these shape functions, obtained through a pointwise optimization program, are in fact C^∞ in the interior of conv X [2]. The continuity of the approximants has also been demonstrated via variational analysis [14]. Explicit formulae for the first and second derivatives of the basis functions have been derived [2, 15, 16]. The aspect ratio of the basis functions is controlled by the nondimensional nodal parameters $\gamma_a = \beta_a h_a^2$, where h_a is the typical nodal spacing around node a. Figure 1 illustrates how the selection of the parameter γ_a influences the aspect ratio of the basis functions for a nonuniform distribution of nodes in two dimensions.

In [2], *max-ent* approximants were linked to convex analysis. Indeed, the positivity of the basis functions, together with the zeroth-order and first-order consistency conditions, enables us to view them as the coefficients of a convex combination or as generalized barycentric coordinates. These attributes, common in geometric modeling, are shared by other positive linearly consistent approximants such as linear finite elements, natural neighbor approximants [17], subdivision surface approximants [18], or B-spline and non-uniform rational B-splines (NURBS) basis functions [19, 20]. The common mathematical structure of convex geometry has important consequences, which suggests gathering all these approximants under the class of convex approximation schemes. For instance, convex approximants satisfy *ab initio* a weak Kronecker-delta property at the boundary of the convex hull of the nodes [2], which greatly facilitates the imposition of essential boundary conditions in sharp contrast with meshfree methods based on the MLS approximants. This property implies that the approximation in a face of the convex hull of the nodes (understood as a two-dimensional (2D) face, an edge, or a vertex) depends only on nodes belonging to this face. In particular, basis functions of interior nodes vanish at the boundary of the domain, designing conforming patches of convex approximants becomes trivial, and the approximation is interpolating at vertices of the domain. This property also allows us to directly impose Dirichlet boundary conditions by prescribing the nodal values in the faces involved. As the scheme is not interpolating at the faces (see Figure 2(left)), one may perform a least-square fit to the boundary data for more accurate results. As elaborated in [21], local convex approximants exhibit the variation diminishing property (the approximation does not create extrema not present in the nodal data). In practice, the basis functions produced by maximum entropy are very smooth (with controlled first and second derivatives) and produce very accurate numerical approximations of PDEs with smooth solutions. Figure 2 illustrates some of these properties for the LME approximants.

As for second-order *max-ent* approximants, it was shown in [2] that the canonical second-order consistency condition $\sum_a p_a x_a \otimes x_a = x \otimes x$ makes the constraints unfeasible in general. This obstacle, further elaborated later, has been overcome in several ways and is the object of the present work. On the basis of the observation that B-splines are convex approximants that can reproduce high-order polynomials, in [6] a method was proposed to properly formulate feasible second-order constraints in the entropy maximization program, while retaining the nonnegativity. This approach,
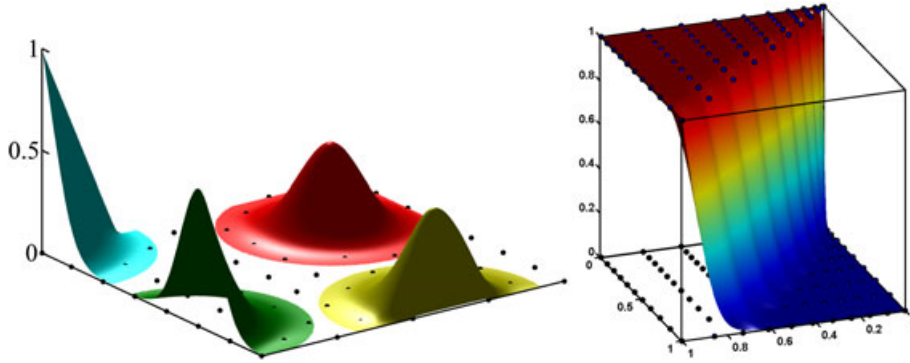
Figure 2. Illustration of nonnegativity, smoothness, and weak Kronecker-delta properties for two-dimensional local maximum entropy (LME) basis functions (left), and the variation diminishing property (right).

reviewed in Section 3, is not very flexible and cannot properly deal with node sets with significant density variations. Here, we propose a related method that resolves these issues. A different strategy inspired by the de Boor algorithm, which recursively constructs the B-spline basis functions, was proposed in [7] to develop high-order *max-ent* approximants. Although the resulting approximants preserve the convexity and the construction is simple, the geometry and combinatorics of support intersections lead to an explosion of the number of basis functions in high dimensions or for high order, and to very heterogeneous basis functions. In [5], the condition $p_a(x) \geqslant 0$ was relaxed, and a generalization of the Shannon entropy for signed probability distributions was maximized. However, this strategy undermines the attractive features of convex approximation schemes. Recently, by relaxing the nonnegativity constraint, Bompadre *et al.* defined in multiple dimensions and analyzed mathematically maximum entropy approximants of any order of consistency [22]. The extra flexibility gained by dropping nonnegativity causes these approximants to loose the Kronecker-delta properties at the boundary and leads to more wiggly basis functions, presumably harder to integrate. In contrast, the method we present here is quite rigid in its formulation but retains the nice properties of convex approximants.

Maximum entropy approximants have been exercised in a wide range of applications, including compressible and near-incompressibility elasticity [23, 24], structural engineering [25], linear and nonlinear elasticity [2, 7, 16], structural vibrations [4, 6], Lagrangian simulations of extreme deformations in solids and fluids [26], Kohn–Sham density functional theory [27], convection–diffusion equations [28], linear and nonlinear thin shell analysis [15, 29], and high-order phase field models of biomembranes [30]. A mathematical analysis of the convergence of LME approximants can be found in [31]. The excellent behavior of the first-order LME approximants in high-order PDEs [15, 29, 30], and its flexibility for adaptive refinement [30], is remarkable.

3. SECOND-ORDER MAXIMUM ENTROPY APPROXIMANTS

The key idea of the proposed method is to add appropriate constraints to the (LME) convex optimization problem, such that the program remains feasible and up to second-order polynomials can be exactly reproduced. However, this seemingly natural extension is not straightforward [2]. We revisit the geometry of the constraints in the view of the classic problem of moments and propose a method that guarantees feasibility.

3.1. *Feasibility conditions for second-order convex approximants*

A family of approximants is nth-order consistent if it reproduces exactly all polynomials of degree up to n. The canonical expression in multiple dimensions of the second-order consistency

condition is

$$\sum_{a=1}^{N} s_a(\boldsymbol{x})\, \boldsymbol{x}_a \otimes \boldsymbol{x}_a = \boldsymbol{x} \otimes \boldsymbol{x}, \tag{1}$$

where $\boldsymbol{x}_a$ and $s_a(\boldsymbol{x})$ represent the node coordinates and the basis functions, respectively, and $\boldsymbol{x}$ is the point where the approximants are evaluated.

It was shown in [2] that the set of convex approximants fulfilling Equation (1) is empty except when $\boldsymbol{x} = \boldsymbol{x}_a$. To illustrate this point, let us turn to one dimension, where we seek at each evaluation point for basis functions satisfying

$$s_a \geqslant 0, \quad \sum_{a=1}^{N} s_a = 1, \quad \sum_{a=1}^{N} s_a \begin{bmatrix} x_a \\ x_a^2 \end{bmatrix} = \begin{bmatrix} x \\ x^2 \end{bmatrix}.$$

From the very definition of the convex hull, it follows that these conditions are feasible if and only if

$$\begin{bmatrix} x \\ x^2 \end{bmatrix} \in \text{conv} \left\{ \begin{bmatrix} x_a \\ x_a^2 \end{bmatrix}, \ a = 1, \ldots, N \right\}.$$

As illustrated in Figure 3 for $N = 7$, this requires that the points in the graph of the function x^2 in the interval $[x_1, x_7]$ belong to the convex hull of the points (x_a, x_a^2), $a = 1, \ldots, 7$, represented in dark gray. This is obviously not true in general. Another way of understanding the issue of feasibility is to note that the first and second-order consistency conditions can be rewritten, after simple manipulations, as

$$\sum_{a=1}^{N} s_a(x - x_a) = 0, \quad \sum_{a=1}^{N} s_a(x - x_a)^2 = 0.$$

The second equation states that a sum of nonnegative numbers must be equal to zero. Furthermore, for x away from the nodes, this equation implies that all the basis functions at this point must vanish, which is inconsistent for instance with the partition of unity property.

In multiple dimensions, a similar feasibility condition follows

$$(\boldsymbol{x}, \boldsymbol{x} \otimes \boldsymbol{x}) \in \text{conv}\{(\boldsymbol{x}_a, \boldsymbol{x}_a \otimes \boldsymbol{x}_a), \ a = 1, \ldots, N\},$$

which nevertheless does not admit such a simple geometric interpretation. In this equation, $(\boldsymbol{x}, \boldsymbol{x} \otimes \boldsymbol{x})$ is an element of $\mathbb{R}^d \times \mathbb{R}^{d \times d}_{symm}$.
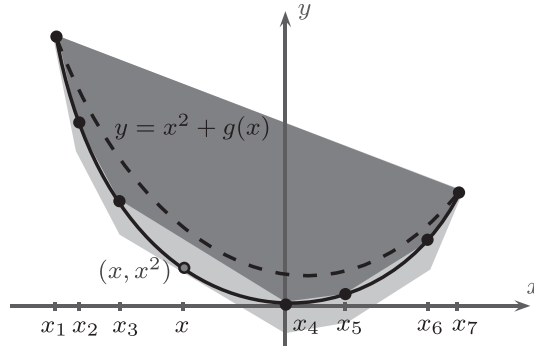

Figure 3. Moment space depicting the convex hull (dark gray region) for the extended point set $\{(x_a, x_a^2)\} \subset \mathbb{R}^2$ and the enlarged convex hull (light gray region) for $\{(x_a, x_a^2 - d_a)\} \subset \mathbb{R}^2$, where $a = 1, \ldots, 7$. The black line represents (x, x^2), whereas the dashed line illustrates how this set can be moved inside the convex hull by adding a function $g(x)$.

Going back to Figure 3, it is possible to satisfy feasibility if the second-order canonical constraint is replaced by

$$\sum_{a=1}^{N} s_a x_a^2 = x^2 + g(x),$$

where the gap function $g(x)$ is such that (1) the graph of $x^2 + g(x)$ lies within the convex hull of the points (x_a, x_a^2), $a = 1, \ldots, 7$, and (2) the second-order consistency condition is not violated. It can be easily checked that B-spline approximants satisfy such an equation, where in this case the proper location of the nodes x_a should be determined through the first-order consistency condition. The first requirement implies that $g(x) \geqslant 0$ and it must vanish at the boundary of the domain. The second condition requires $x^2 + g(x)$ to be piecewise quadratic and smooth because the smoothness of the basis functions is that of the gap function. The aspect ratio of the resulting shape functions depends on the magnitude of the gap relative to the square of the nodal spacing; larger gaps result in widespread functions. This approach was proposed in [6] and implemented in multiple dimensions. Its main limitation is that it is difficult to define appropriate gap functions for unstructured node sets in multiple dimensions with significant variations in the node density, while controlling the aspect ratio of the basis functions and maintaining their smoothness. Yet, this reference shows the potential of second-order *max-ent* methods and the favorable comparison with MLS or B-spline quadratic approximants.

Here, we take a different approach to guarantee feasibility by changing the weights used to reproduce the quadratic monomials rather than the function being reproduced. Following the moment space interpretation in Figure 3, rather than achieving feasibility by squeezing $x^2 + g(x)$ into the convex hull represented in dark gray, we enlarge the convex hull so that x^2 falls within. More specifically, we consider a second-order constraint given by

$$\sum_{a=1}^{N} s_a \left(x_a^2 - d_a\right) = x^2, \quad \text{or equivalently} \quad \sum_{a=1}^{N} s_a (x - x_a)^2 = \sum_{a=1}^{N} s_a d_a, \tag{2}$$

for some nonnegative parameters d_a. With this modified second-order condition, the constraints are feasible if and only if the graph of x^2 lies within the convex hull of the points $(x_a, x_a^2 - d_a)$, $a = 1, \ldots, N$. It is obvious from the figure that setting $d_1 = d_7 = 0$ and $d_a > 0$ for $a = 2, \ldots, 6$, the modified convex hull is larger, and for sufficiently large values of d_a for the interior nodes, the feasibility condition is met. Looking at the second expression in Equation (2), we observe that now the right-hand side is greater than zero. This viewpoint allows us to flexibly design feasible constraints by simply setting the offsets d_a and, as shown later, is easily applicable to sets of nodes with variable density. Standard convex high-order consistent methods, available in 1D, such as B-splines, can be shown to satisfy such conditions. In fact, the nodal gaps d_a for these schemes are very instructive in designing second-order *max-ent* (SME) approximants. See Appendix A for a summary of the nodal gaps for B-spline schemes of several orders and uniform knot vector [20].

In the multidimensional case, the modified second-order constraint considered here takes the form

$$\sum_{a=1}^{N} s_a(\boldsymbol{x}_a \otimes \boldsymbol{x}_a - \boldsymbol{d}_a) = \boldsymbol{x} \otimes \boldsymbol{x}, \quad \text{or equivalently} \quad \sum_{a=1}^{N} s_a(\boldsymbol{x} - \boldsymbol{x}_a) \otimes (\boldsymbol{x} - \boldsymbol{x}_a) = \sum_{a=1}^{N} s_a \boldsymbol{d}_a, \tag{3}$$

where $\boldsymbol{d}_a$ are now symmetric semi-positive definite matrices for each node. These constraints are to be contrasted with the viewpoint adopted previously in [6]

$$\sum_{a=1}^{N} s_a \, \boldsymbol{x}_a \otimes \boldsymbol{x}_a = \boldsymbol{x} \otimes \boldsymbol{x} + \boldsymbol{G}(\boldsymbol{x}), \tag{4}$$

where $\boldsymbol{G}(\boldsymbol{x})$ is a field of symmetric semi-positive definite matrices, which needs to be provided and needs to be smooth and piecewise quadratic. We shall see that at the boundary, $\boldsymbol{G}(\boldsymbol{x})$ cannot have full rank. Comparing both approaches, we find $\boldsymbol{G}(\boldsymbol{x}) = \sum_{a=1}^{N} s_a(\boldsymbol{x}) \, \boldsymbol{d}_a$ (see Appendix B),

showing that the present approach bears similarity with that in [6], with an implicit gap function expressed in terms of the sought-after SME approximants. This comparison also shows how both related approaches are not equivalent.

In multiple dimensions, the feasibility condition becomes

$$(x, x \otimes x) \in \text{conv}\{(x_a, x_a \otimes x_a - d_a), a = 1, \dots, N\},$$

for which, to the best of our knowledge, clear geometric insight is lacking.

3.2. Design of feasible constraints

The crucial question resides in the proper choice of the nodal gaps d_a. We first take a one-dimensional domain defined by the nodes x_a, $a = 1, \dots, N$, and the associated moment space to explain the ideas behind the definition of the set of nodal gaps $d_a, a = 1, \dots, N$. Later, we extend the procedure to higher dimensions.

3.2.1. The one-dimensional case. We turn to the moment space illustrated in Figure 3 and classify the nodes as boundary (x_1 and x_7), next to boundary (x_2 and x_3), and interior nodes (x_3, x_4, and x_5). The selection of the gap for interior nodes is illustrated in Figure 4(left), where the region of the moment space involving the points (x_a, x_a^2) for $a = 3, 4, 5$ is shown in more detail. Consider the line passing through the points (x_3, x_3^2) and (x_4, x_4^2), and analytically described by $y(x) = (x_3 + x_4)x - x_3x_4$. This line is part of the boundary of the convex hull of the extended point set in an interval of length $h_3 = x_4 - x_3$. The vertical distance between this line and the graph of the function x^2 measures the violation of the feasibility condition. It is easy to see that the maximum violation is achieved at the center of the interval, and its value is $h_3^2/4$. Shifting downwards the points (x_3, x_3^2) and (x_4, x_4^2) by this amount produces a new line, tangent to x^2, which is now the boundary of the enlarged convex hull that does not violate the feasibility condition in this interval. Thus, the nodal gaps d_3 and d_4 should be greater than or equal to $h_3^2/4$. Similarly, d_4 and d_5 should be greater than or equal to $h_4^2/4$. We are now in a position to assign the nodal gap d_4, which must satisfy $d_4 = \alpha/4 \max(h_3^2, h_4^2)$, where the nondimensional parameter $\alpha \geq 1$ measures the slack in the feasibility condition.

The choice $\alpha = 1$ is a degenerate case; at the tangency point, (x, x^2) is not in the strict interior of the convex hull of the extended nodes (x_a, x_a^2), $a = 1, \dots, N$, and the implicit function theorem cannot be invoked to prove mathematically the smoothness of the maximum entropy basis functions or the strict convexity of the dual program [2]. In fact, experience shows that these properties are lost at degenerate points. In practice, we choose $\alpha > 1$. We can summarize the generic expression for the nodal gaps of interior nodes as

$$d_a = \frac{\alpha}{4} \max\left(h_{a-1}^2, h_a^2\right) \quad \text{for } a = 3, \dots, N - 2,$$

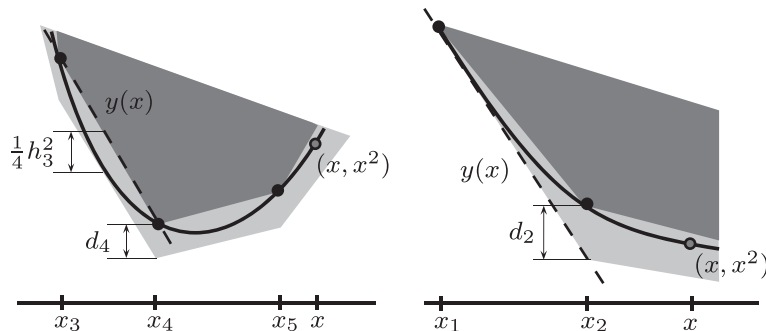where $h_{a-1} = x_a - x_{a-1}$ and $\alpha > 1$.

Figure 4. Detail of the moment space depicting the computation of the nodal gap d_a for (left) the interior nodes and (right) the rest of the nodes.
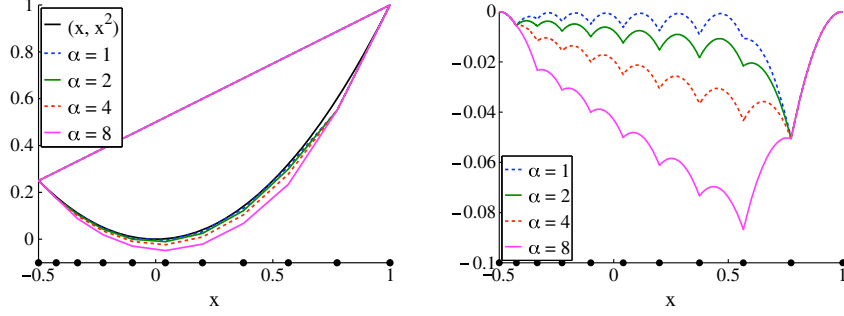
Figure 5. Enlarged moment space for several slack parameters α for a nonuniform one-dimensional set of points (left). The difference between the lower boundary of the moment space and x^2 is shown in the right plot, showing clearly that for all slack parameters and for all x, the problem is feasible; that is, the graphs are below zero.

This rule is not valid for the nodes located at the boundaries of the domain. Figure 4(right) illustrates the situation at one of the boundaries. It is clear in the picture that d_1 needs to be zero not to violate the feasibility conditions in the vicinity of the endpoint x_1. Otherwise, the upper boundary of the convex hull would move downwards, leaving part of the graph of (x, x^2) outside the feasible region. In general, we have $d_1 = d_N = 0$. The nodes next to boundary nodes also deserve a special treatment. Consider the line passing through the points (x_1, x_1^2) and $(x_2, x_2^2 - d_2)$, given by $y(x) = m(x - x_1) + x_1^2$. Feasibility requires that this line should stay below the graph of x^2, that is, in the limiting case be tangent at (x_1, x_1^2). It is immediate to see that this is the case if $d_2 = h_1^2$. Thus, the generic expression to compute the gap for the nodes located next to the boundary is

$$d_2 = \max\left\{\beta h_1^2, \frac{\alpha}{4}h_2^2\right\} \quad \text{and} \quad d_{N-1} = \max\left\{\beta h_{N-1}^2, \frac{\alpha}{4}h_{N-2}^2\right\},$$

where another nondimensional slack parameter $\beta \geq 1$ is introduced to gain flexibility in the enlargement of the convex hull. We adopt $\beta = 1$ in most cases.

Figure 5 illustrates how the convex hull of the points $(x_a, x_a^2 - d_a)$ is enlarged for different values of the slack parameter α, including the graph of the function x^2 as required for feasibility, for a nonuniform one-dimensional set of points. In the right, we subtract x^2 to the ordinate to highlight how the convex hull becomes larger as α increases. This has also an effect on the support size of the basis functions, as illustrated later.

3.2.2. Extension to higher dimensions. So far, we have presented a method to satisfy the feasibility conditions for a one-dimensional domain by enlarging the size of the convex hull of the extended point set. We illustrate now how these ideas carry over to the multidimensional case for polyhedral domains given by the convex hull of the set of nodes, following a similar procedure to that in [6]. For more general domains, with possibly curved boundary, we refer to [4].

Rather than describing the procedure in its full generality, which is unavoidably heavy on the notation and convex geometry concepts, we proceed by describing the 2D case and hint how to extend to higher dimensions. We assume that for each node, we have a measure of the representative nodal spacing h_a. The general procedure relies on two observations. First, because of the reduced face property [2], on low-dimensional faces of the polyhedral domain, any convex approximation scheme can be restricted to the face because the contributions to the discrete probability distribution of all nodes not belonging to the face vanish. This allows us to determine the nodal gaps on the low-dimensional face. Second, by multiplying the first and second-order consistency conditions by any unit vector,

$$\sum_{a=1}^{N} s_a(\boldsymbol{x}_a \cdot \boldsymbol{u}) = (\boldsymbol{x} \cdot \boldsymbol{u}), \quad \sum_{a=1}^{N} s_a[(\boldsymbol{x}_a \cdot \boldsymbol{u})^2 - \boldsymbol{u} \cdot \boldsymbol{d}_a \cdot \boldsymbol{u}] = (\boldsymbol{x} \cdot \boldsymbol{u})^2,$$
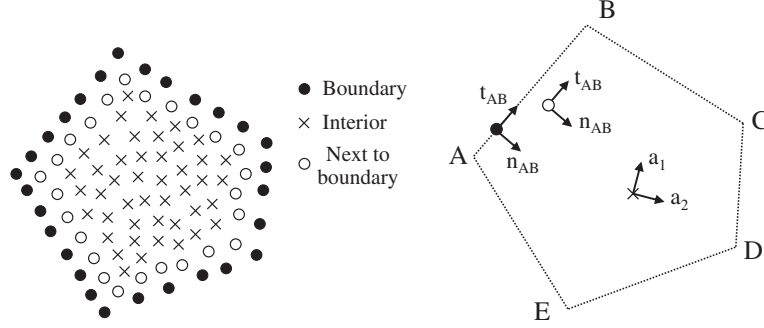
Figure 6. (left) Classification of the nodes in three categories – interior ($\times$), belonging to the boundaries ($\bullet$), and next to the boundaries ($\circ$) – and (right) directions needed to compute the gap for each kind of node.

we obtain necessary projected one-dimensional feasibility conditions, which allow us to exploit operationally the reduced face property and to assign nodal gaps for nodes next to the faces.

Consider the 2D domain in Figure 6(left). For the extreme points of the polygon (A to E), we can apply the reduced face property on the adjacent faces and conclude that $\boldsymbol{d}_A = \boldsymbol{d}_B = \boldsymbol{d}_C = \boldsymbol{d}_D = \boldsymbol{d}_E = \boldsymbol{0}$. Consider now a face, say $\overline{AB}$, and a unit tangent vector $\boldsymbol{t}$. The reduced face property allows us to determine the projected nodal gaps $\boldsymbol{t} \cdot \boldsymbol{d}_a \cdot \boldsymbol{t}$ following the one-dimensional procedure. If we project the consistency conditions along a unit normal to this boundary $\boldsymbol{n}$, all the points in the face collapse in a single point, and therefore the nodes on $\overline{AB}$ satisfy $\boldsymbol{n} \cdot \boldsymbol{d}_a \cdot \boldsymbol{n} = 0$. Consequently, for the nodes in the face $\overline{AB}$, we have

$$\boldsymbol{d}_A = \boldsymbol{d}_B = \boldsymbol{0},$$
$$\boldsymbol{d}_a = \beta \, h_a^2 \, \boldsymbol{t} \otimes \boldsymbol{t} \quad \text{for the nodes next to } A \text{ and } B,$$
$$\boldsymbol{d}_a = \frac{\alpha}{4} \, h_a^2 \, \boldsymbol{t} \otimes \boldsymbol{t} \quad \text{for all other nodes in } \overline{AB}.$$

For nodes in the interior of the 2D domain but next to the boundary, say to the face $\overline{AB}$, we proceed again by projecting the consistency conditions along $\boldsymbol{t}$ and $\boldsymbol{n}$. Along the normal direction, we have

$$\boldsymbol{n} \cdot \boldsymbol{d}_a \cdot \boldsymbol{n} = \beta \, h_a^2,$$

while along the tangential direction, if the node is far from other faces, we have

$$\boldsymbol{t} \cdot \boldsymbol{d}_a \cdot \boldsymbol{t} = \frac{\alpha}{4} \, h_a^2,$$

and thus,

$$\boldsymbol{d}_a = \beta \, h_a^2 \boldsymbol{n} \otimes \boldsymbol{n} + \frac{\alpha}{4} \, h_a^2 \boldsymbol{t} \otimes \boldsymbol{t}.$$

If the node is next to two different one-dimensional faces with normals $\boldsymbol{n}$ and $\boldsymbol{n}'$, a conservative choice is

$$\boldsymbol{d}_a = \beta \, h_a^2 \left(\boldsymbol{n} \otimes \boldsymbol{n} + \boldsymbol{n}' \otimes \boldsymbol{n}' \right).$$

Interior nodes to the 2D domain are easily dealt with. If the nodal spacing is isotropic, by projecting the consistency conditions along any two orthogonal directions, we obtain

$$\boldsymbol{d}_a = \frac{\alpha}{4} \, h_a^2 \boldsymbol{Id},$$

where $\boldsymbol{Id}$ is the identity matrix. The extension of this procedure to higher dimensions follows naturally.

If the nodal spacing is not isotropic, it is easy to adapt the strategy. Let us consider the simplest case of an interior node, and let the nodal spacing be described by a metric tensor $\boldsymbol{h}_a$ (symmetric and

positive definite) characterizing the nodal spacing in each direction. Let us denote its eigenvalues and eigenvectors by h_a^i and v_a^i, for $i = 1, \ldots, d$. By projecting the consistency conditions along the eigenvectors, we obtain the following nodal gap, adapted to the anisotropy of the nodal spacing

$$\boldsymbol{d}_a = \frac{\alpha}{4} \sum_{i=1}^{d} \left(h_a^i \right)^2 \boldsymbol{v}_a^i \otimes \boldsymbol{v}_a^i.$$

3.3. Optimization program for the second-order max-ent approximants

Maximum entropy can serve as a selection principle and a practical computational method to choose information-theoretic optimal approximants amongst the set of second-order consistent convex approximants satisfying feasible constraints. The resulting optimization program can be written as

$$
\begin{aligned}
\text{(SME)} \quad \text{For fixed } \boldsymbol{x} \text{ maximize} \quad & -\sum_{a=1}^{N} s_a \ln s_a \\
\text{subject to} \quad & s_a \geqslant 0, \quad a = 1, \ldots, N \\
& \sum_{a=1}^{N} s_a = 1 \\
& \sum_{a=1}^{N} s_a \, (\boldsymbol{x}_a - \boldsymbol{x}) = \boldsymbol{0} \\
& \sum_{a=1}^{N} s_a \, \boldsymbol{D}_a(\boldsymbol{x}, \boldsymbol{x}_a, \boldsymbol{d}_a) = \boldsymbol{0},
\end{aligned}
$$

where

$$\boldsymbol{D}_a(\boldsymbol{x}, \boldsymbol{x}_a, \boldsymbol{d}_a) = (\boldsymbol{x} - \boldsymbol{x}_a) \otimes (\boldsymbol{x} - \boldsymbol{x}_a) - \boldsymbol{d}_a,$$

and the first and second-order consistency conditions have been written in a centered form using the lower order consistency conditions, which proves more convenient computationally because it leads to bounded Lagrange multipliers.

Following the same duality methods as in [2], and introducing the partition function

$$Z(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \sum_{a=1}^{N} \exp[\boldsymbol{\lambda} \cdot (\boldsymbol{x} - \boldsymbol{x}_a) - \boldsymbol{\mu} : \boldsymbol{D}_a],$$

the optimal solution can be written as

$$s_a(\boldsymbol{x}) = \frac{\exp\left[\boldsymbol{\lambda}^*(\boldsymbol{x}) \cdot (\boldsymbol{x} - \boldsymbol{x}_a) - \boldsymbol{\mu}^*(\boldsymbol{x}) : \boldsymbol{D}_a \right]}{Z(\boldsymbol{x}, \boldsymbol{\lambda}^*(\boldsymbol{x}), \boldsymbol{\mu}^*(\boldsymbol{x}))}. \tag{5}$$

The optimal Lagrange multipliers $\boldsymbol{\lambda}^*(\boldsymbol{x})$ and $\boldsymbol{\mu}^*(\boldsymbol{x})$ are obtained by minimizing the reduced Lagrange dual function

$$\hat{g}(\boldsymbol{\lambda}, \boldsymbol{\mu}) = \ln Z(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}). \tag{6}$$

Analogous arguments to those in [2] show that if the consistency conditions are strictly feasible, that is,

$$(\boldsymbol{x}, \boldsymbol{x} \otimes \boldsymbol{x}) \in \text{int}[\text{conv}\{(\boldsymbol{x}_a, \boldsymbol{x}_a \otimes \boldsymbol{x}_a - \boldsymbol{d}_a), \, a = 1, \ldots, N\}],$$

then this dual unconstrained optimization program with $d(d+3)/2$ unknowns is smooth and strictly convex with a unique solution, which can be efficiently obtained using Newton's method. Furthermore, the resulting basis functions are C^∞ inside the convex hull of the set of nodes. The strict

feasibility of the constraints, even in multiple dimensions and for unstructured sets of nodes, is easily achieved in practice with the help of the parameters α and β described in the previous section.

Note that in contrast with the optimization program (LME), there is no explicit term in the objective function of the (SME) program favoring local basis functions. As a matter of fact, as shown in the examples that follow, the locality is a consequence of the second-order consistency. Note that if $\mu^*(x)$ is positive definite, then it is possible to factor out a Gaussian function centered at x_a from the basis function in Equation (5). We observe that this is the generic situation for the interior nodes, although further mathematical analysis is needed to establish a rigorous result. We illustrate next the general features of the resulting approximants.

3.4. Examples of second-order max-ent approximants

3.4.1. General features. Figure 7 illustrates the SME approximants for a one-dimensional set of uniformly distributed nodes. The basis functions, the first and second derivatives are shown for several values of the slack parameter α ranging from 1.5 to 8. The smoothness of the basis functions and its derivatives, the nonnegativity, and the Kronecker-delta property can be clearly appreciated. A clear correspondence can be observed between the magnitude of the slack and the width of the basis functions. For $\alpha = 1.5$, the basis functions span about four node spacings, whereas for $\alpha = 4$, they span about six node spacings. For the numerical approximation of partial differential equations, the computational cost grows with the width of the basis functions because numerical quadrature tends to be more demanding and the bandwidth of the mass or the stiffness matrices increases. On the other hand, we observe more accurate solutions of smooth problems for sufficiently resolved grids and widespread shape functions. On the basis of our experience with highly nonuniform grids, we recommend the range of values $1.2 \leqslant \alpha \leqslant 3$.

Strictly speaking, the support size of each *max-ent* basis function is conv X [2]. However, the fast Gaussian decay of LME and SME approximants allows us to circumvent this practical inconvenience, and from a numerical perspective, a basis function associated to the node x_a can be considered as compactly supported in a ball of radius R_a centered at x_a, outside of which the basis function is zero within a tolerance Tol_0. For SME, we use $R_a = \max\left(\sqrt{-\frac{\alpha}{2}\ln(\text{Tol}_0)}\,h_a,\ 4h_a\right)$. This expression for the numerical support stems from a heuristic estimation of the Lagrange multipliers provided later.

The proposed approach can deal with nonuniform sets of nodes and still have a precise control of the magnitude of the slack through the nondimensional parameter α, as illustrated in the numerical examples presented later. However, if the node distribution is very irregular, the basis functions can present small wiggles near its tails, which are exacerbated upon differentiation, and demand very expensive numerical quadrature to retain the optimal convergence. In the same way that nicely graded finite element meshes produce better solutions, it is preferable to define node distributions with smooth variations in the nodal spacing, as illustrated in Figure 8, where the basis functions and derivatives are computed for $\alpha = 1.5$. The uniform aspect ratio relative to the nodal spacing of the basis functions can be observed, set by the uniform α, despite the nonuniform spacing. In multiple dimensions, centroidal Voronoi tessellation (CVT) techniques [32, 33] allow us to distribute smoothly points according to a specified density function, which can be based on the features of the solution as shown in Section 4.

The basis functions computed with a parameter $\alpha = 2$ in a 2D domain are depicted in Figure 9. The weak Kronecker-delta property can be observed. The explicit expressions to compute the gradient and the Hessian of the basis functions are given in Appendix C.

3.4.2. Second-order consistency condition. To test the correctness of the formulation and its implementation, we approximate the function

$$u(x) = x^T \begin{bmatrix} 1.12 & -3.1 \\ -3.1 & -2.4 \end{bmatrix} x + [2.23 \ -3]\, x + 8.35$$
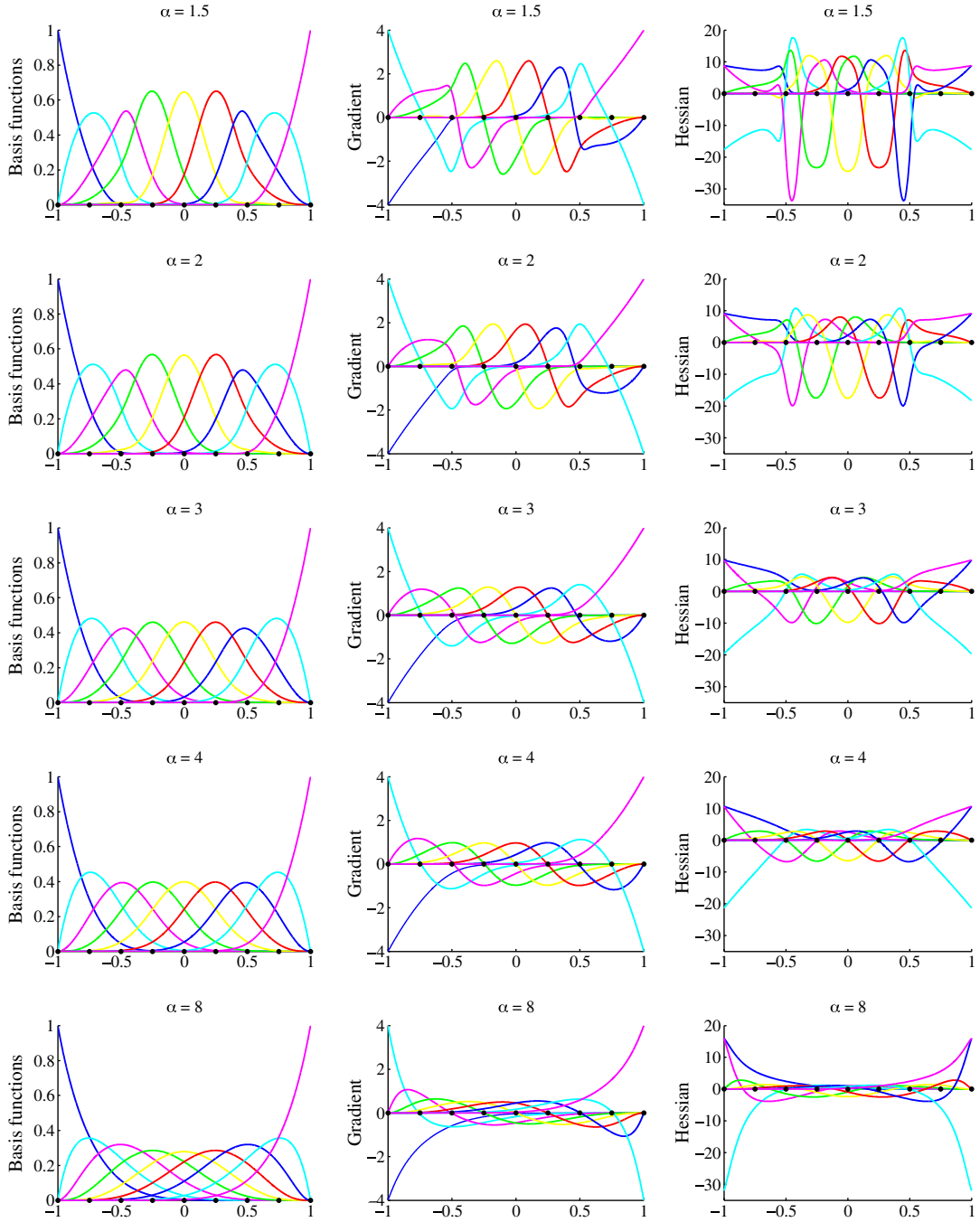
Figure 7. Second-order maximum entropy (SME) basis functions and components of the gradient and the Hessian for a uniform one-dimensional grid of points and different values of the parameter α.

by $\tilde{u}(\boldsymbol{x}) = \sum_{a=1}^{N} s_a(\boldsymbol{x}) \, u_a$, where the nodal coefficients u_a are obtained with a least-square fit. The SME approximants should be able to reproduce exactly any quadratic function, and therefore, we expect that the error being minimized in the least-square fit $\sum_{i=1}^{M} \left\| u(\boldsymbol{x}^i) - \tilde{u}(\boldsymbol{x}^i) \right\|^2$ vanishes up to round-off errors. Here, $\boldsymbol{x}^i$, $i = 1, \ldots, M$, are sampling points. We consider a 2D grid with 586 nodes. With up to 10,000 sampling points, we find the error to be zero within machine precision for various choices of α in the range $[1.2, 3]$. We also observe that the accuracy of the gradients of $u(\boldsymbol{x})$ is within machine precision.
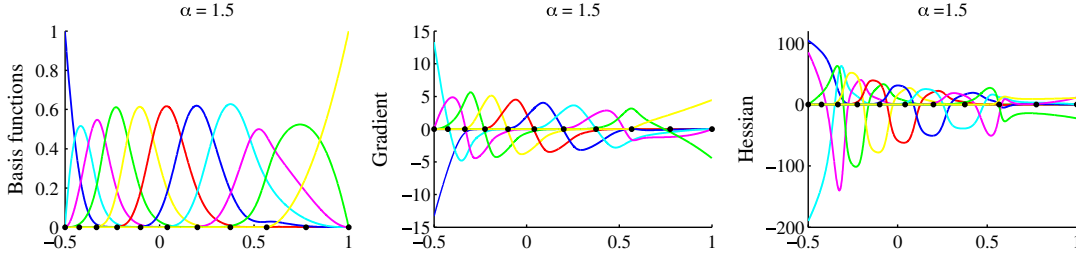
Figure 8. Second-order maximum entropy (SME) basis functions and components of the gradient and the Hessian corresponding to $\alpha = 1.5$ for a nonuniform one-dimensional grid of points.
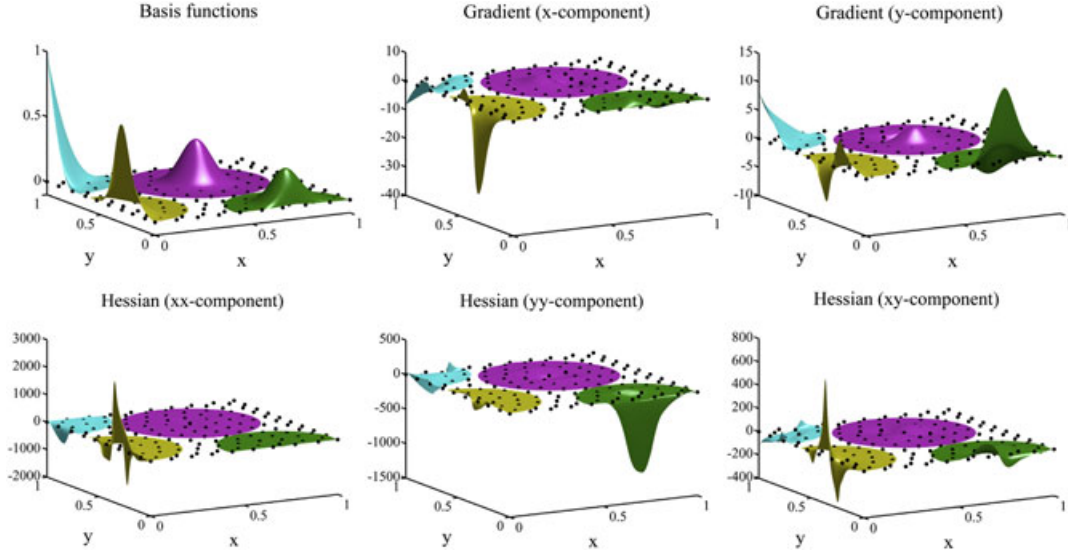

Figure 9. Second-order maximum entropy (SME) basis functions and components of the gradient and the Hessian computed with a parameter $\alpha = 2$ in a two-dimensional domain.

3.4.3. The Lagrange multipliers. The calculation of the basis functions, their gradient, and Hessian is not explicit because Lagrange multipliers $\boldsymbol{\lambda}^*$ and $\boldsymbol{\mu}^*$ must be computed for each evaluation point $\boldsymbol{x}$. This fact is not a significant drawback because the unconstrained, smooth, convex problem of minimizing the function in Equation (6) can be reliably and robustly solved with Newton's method [2, 6, 16] (see Appendix C for details). Here, we provide heuristics on how to produce good initial guesses to reduce the number of iterations and to improve the efficiency of the algorithm. We first explain the idea for the one-dimensional case, and later, we extend it to higher dimensions.

By considering a uniform one-dimensional grid, away from the boundaries we observe that $\lambda^* \approx 0$ and $\mu^* d_{a,CN} \approx 1/2$, where $d_{a,CN}$ is the nodal gap of the closest node (CN) to the evaluation point. Therefore, $\lambda^{est} = 0$ and $\mu^{est} = 1/2(d_{a,CN})^{-1}$ are good initial guesses for the Lagrange multipliers, as depicted in Figures 10(left-center) and 11(left-center) for uniform and nonuniform grids, respectively. We adopt $\lambda^{est} = 0$ and $\mu^{est} = 0$ for evaluation points close to the boundaries. It can be observed that this estimate is very good even for a nonuniform grid and near the boundary for μ. Figures 10(right) and 11(right) show the number of iterations needed to compute the Lagrange multipliers with the proposed initial guesses and with null initial guesses. In average, the number of iterations is reduced by 40% for the uniform grid and by 30% for the nonuniform distribution of points.

By replacing the expression for the nodal gap $d_{a,CN} = \alpha/4h_{a,CN}^2$ into the estimate $\mu^{est} = 1/2(d_{a,CN})^{-1}$, we have $\mu \approx (2/\alpha)/h_{a,CN}^2$. Recalling that in the SME approximants μ dictates the decay of the Gaussian factor in the same way that $\beta_a = \gamma/h_a^2$ does for the LME basis functions,
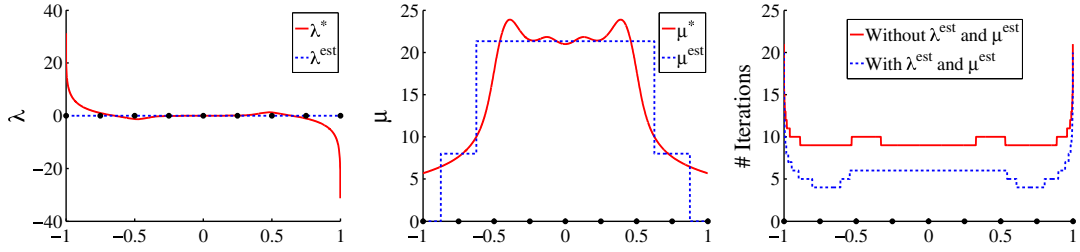
Figure 10. (left-center) Lagrange multipliers λ^* and μ^* and initial estimations λ^{est} and μ^{est} corresponding to $\alpha = 1.5$ for the uniform one-dimensional grid of points illustrated in Figure 7; (right) number of iterations needed to compute the Lagrange multipliers.

Figure 11. (left-center) Lagrange multipliers λ^* and μ^* and initial estimations λ^{est} and μ^{est} corresponding to $\alpha = 1.5$ for the nonuniform one-dimensional grid of points illustrated in Figure 8; (right) number of iterations needed to compute the Lagrange multipliers.
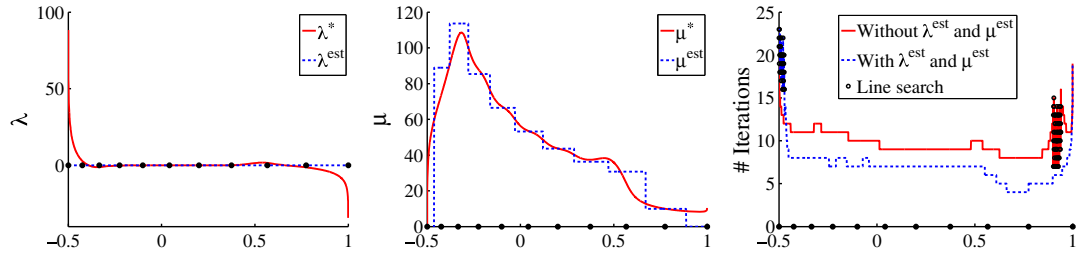
the relationship between the dimensionless parameters setting the width of the LME and SME basis functions, γ and α, respectively, becomes evident. In multiple dimensions, we take as initial guesses $\boldsymbol{\lambda}^{est} = \mathbf{0}$ and $\boldsymbol{\mu}^{est} = 1/2(\boldsymbol{d}_{a,CN})^{-1}$. The pseudoinverse is used when the evaluation point is on or next to the boundaries because for these points $\det(\boldsymbol{d}_{a,CN}) = 0$.

4. APPLICATIONS OF THE SECOND-ORDER *MAX-ENT* APPROXIMANTS

In this section, we examine the performance of SME basis functions for a number of different problems. First, their accuracy in problems of structural vibrations is illustrated with two one-dimensional examples, which nicely illustrate the comparison against another family of convex approximants, B-splines. In comparing with B-splines, it should be noted that as these are piece-wise polynomials, they can be more easily integrated than our meshfree approximants. However, for higher order B-splines, the integration cost is comparable. The flexibility of SME approximants to handle unstructured grids of points is illustrated with a Poisson problem with localized sources. Finally, we highlight the excellent behavior exhibited by SME approximants in high-order PDEs. The SME approximants are particularly well suited for such problems because their smoothness allows us to follow a direct Galerkin approach. We examine the accuracy in the calculation of the Willmore energy, present results for Kirchhoff–Love thin shells, and explore the possibilities of the method for a nonlinear fourth-order partial differential equation describing in a phase field spirit the mechanics of biomembranes [34]. This is an ideal target application of the method proposed here, given its high-order character, together with the fact that the phase field develops sharp fronts that need to be resolved accurately. Traditional approaches rely either on smooth tensor product approximants, for example, B-splines, unable to locally refine the numerical resolution, or on mixed C^0 finite elements.

4.1. Structural vibrations

It is well known that higher order finite elements present spurious optical branches in the discrete frequency spectrum of structural vibrations problems [35, 36]. The continuous system only has an

acoustical branch, which is accurately approximated in the low-frequency part of the spectrum. The spurious optical branch, indicative of eigenmodes where neighboring nodes vibrate opposite to each other, arises as a consequence of the spatial inhomogeneity of the shape functions [6]. Although the translational symmetry of the continuum is destroyed for any discretization scheme, Lagrange finite elements introduce a complex symmetry group in the system that gives rise to unphysical vibrations. This topic has been extensively studied in condensed matter physics, within the theory of lattice dynamics [37]. The optical modes manifest themselves in structural dynamics. Indeed, nonlinear effects or the presence of sharp features in the boundary conditions or in the domain excite high-frequency optical modes, leading to noisy solutions [38]. Commonly, they are alleviated introducing special schemes to damp the high frequencies.

Isogeometric analysis has been shown to substantially alleviate this issue. It is possible to design grids that eliminate completely optical modes. For a general grid, a small number of optical modes can appear, particularly because of boundary effects [35]. SME approximants share many features with B-spline basis functions, such as the nonnegativity and the smoothness. It has been already shown that *max-ent* approximants perform very well in structural vibrations problems, as compared with second-order Lagrange finite elements, B-splines, and MLS meshfree approximants [6]. We present here further evidence by considering the vibrations of an elastic rod and of an Euler–Bernoulli beam. We numerically compute the spectrum and compare with the analytical solution for several values of the slack parameter α.

4.1.1. Longitudinal vibrations of an elastic rod. The natural frequencies and modes of an elastic fixed–fixed rod follow from

$$u_{xx} + \omega^2 u = 0 \quad \text{for } x \in (0, 1)$$
$$u(0) = u(1) = 0.$$

The eigenfrequencies can be computed analytically as $\omega_n = n\pi$, $n \in \mathbb{N}$. After discretizing the equation with a grid of N points, the numerical eigenfrequencies ω_n^h and eigenvectors $\boldsymbol{\phi}_n^h$ are calculated from the generalized eigenvalue problem $\left[\boldsymbol{K} - \left(\omega_n^h \right)^2 \boldsymbol{M} \right] \boldsymbol{\phi}_n^h = 0$, where $\boldsymbol{K}$ and $\boldsymbol{M}$ are the stiffness and mass matrices, respectively.

The accuracy of the numerical calculations can be studied by plotting the numerical natural frequencies ω_n^h normalized with respect to the exact frequencies ω_n as a function of the ratio n/N, as illustrated in Figure 12. This normalized discrete frequency spectrum presents the particular feature of being independent of N for a large enough number of points. In the case of SME approximants, the spectrum does not change significantly for $N > 500$. We perform all the calculations of this example with a uniform grid of 1000 points.

The spectra computed with SME approximants for different values of α are plotted with those corresponding to quintic B-splines and LME ($\gamma = 0.8$) approximation schemes in Figure 12. It can
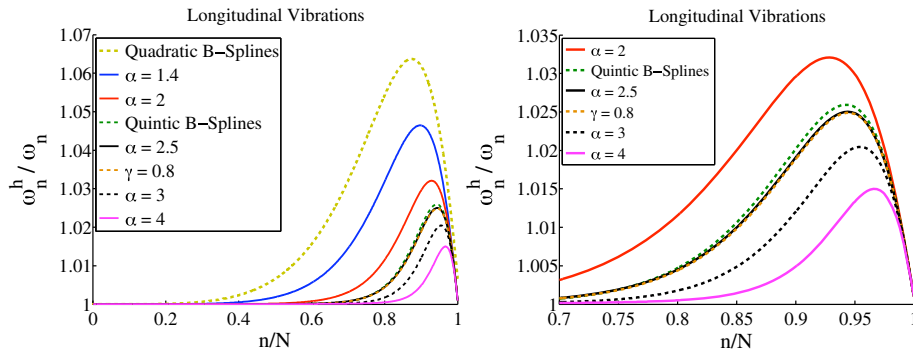

Figure 12. Whole (left) and detail (right) of the normalized discrete eigenfrequency spectrum for an elastic rod (longitudinal vibrations).

be observed that the accuracy of SME results increases as the basis functions are more widespread. In the case of $\alpha = 3$, for instance, it can be appreciated that the ratio ω_n^h/ω_n cannot be distinguished from one in approximately 70% of the discrete spectrum. The LME solution is similar to that obtained with $\alpha = 2.5$, which is not surprising because in the previous section it was shown that $\alpha \approx 2/\gamma$ for a uniform one-dimensional grid of nodes. A fifth-order polynomial B-spline discretization produces a solution with an accuracy comparable with that obtained with $\alpha = 2.5$ and $\gamma = 0.8$. This example suggests that the accuracy of the solution according to this metric does not depend significantly on the order of reproducibility but rather on the smoothness of the basis functions.

4.1.2. Transverse vibrations of an Euler–Bernoulli beam. The transverse vibrations of a simply supported Euler–Bernoulli beam are governed by

$$u_{xxxx} - \omega^2 u = 0 \quad \text{for } x \in (0, 1)$$
$$u(0) = u(1) = 0$$
$$u_{xx}(0) = u_{xx}(1) = 0,$$

where the analytical expression to compute the eigenfrequencies is $\omega_n = (n\pi)^2$, $n \in \mathbb{N}$. The numerical eigenfrequencies ω_n^h are computed analogously to before, but now the stiffness matrix involves the second derivatives of the basis functions.

Figure 13 shows the normalized discrete frequency spectra computed with *max-ent* approximants and quintic B-splines. Although the results exhibit a behavior similar to that of the elastic rod, the results are less accurate for smaller values of α (i.e., narrower basis functions) and for higher frequencies. Again, despite the high-order derivatives involved in the problem, the order of reproducibility does not seem to play a significant role by itself.

4.2. Poisson problem with localized sources

This example shows the convergence in the L_2 norm of the SME basis functions in the Galerkin approximation of a PDE. In particular, this example shows how the proposed method can easily deal with localized sharp features in the solution by locally adapting the density of nodes, while maintaining the second-order convergence. For this purpose, we consider the Poisson boundary value problem

$$-\nabla \cdot (k\nabla u) = s(x, y) \quad \text{in } \Omega = [0, 1] \times [0, 1]$$
$$u = \bar{u} \quad \text{in } \partial\Omega,$$

where $k = 1$, and the source $s(x, y)$ and the boundary data $\bar{u}$ are chosen such that the exact solution, illustrated in Figure 14 (left), is
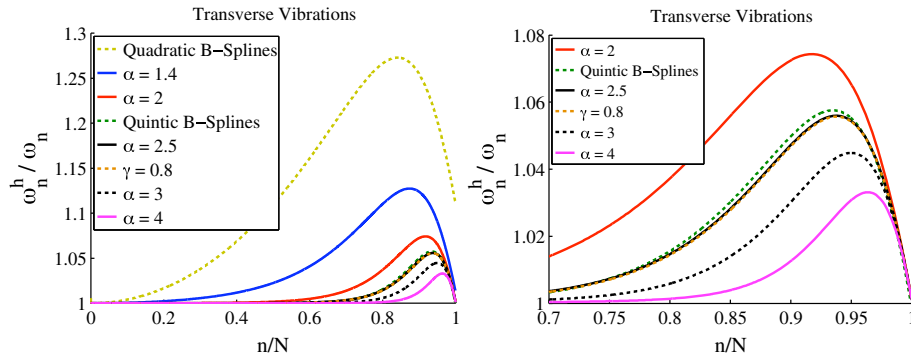

Figure 13. Whole (left) and detail (right) of the normalized discrete eigenfrequency spectrum for an Euler–Bernoulli beam (transverse vibrations).
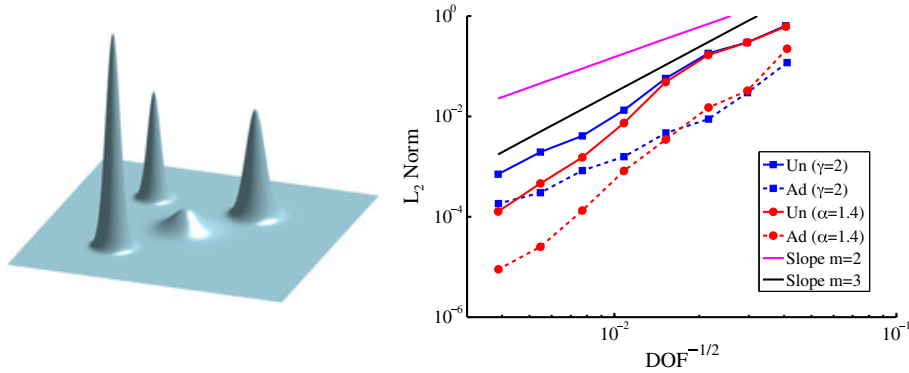
Figure 14. Exact solution (left) and L_2 norm of the error (right) for the Poisson problem. Uniform (Un) and adapted (Ad) grids of points are considered in the calculations, and local maximum entropy (LME) ($\gamma = 2$) and second-order maximum entropy (SME) ($\alpha = 1.4$) approximants are used.

Table I. Coefficients to calculate the exact solution of the Poisson problem.

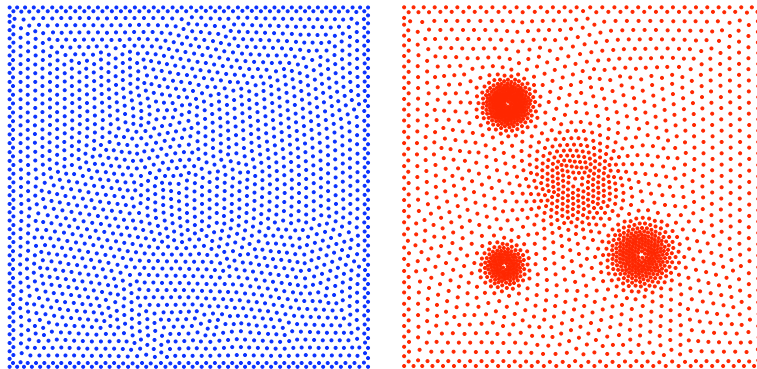| i	A_i	β_i	x_i	y_i
1	10	180	0.51	0.52
2	50	450	0.31	0.34
3	100	800	0.73	0.71
4	50	1000	0.28	0.72

Figure 15. Illustration of the uniform (left) and adapted (right) grids of points considered for the Poisson problem.

$$u(x, y) = \sum_{i=1}^{4} A_i \, e^{-\beta_i \left[(x-x_i)^2 + (y-y_i)^2 \right]}.$$

The coefficients are given in Table I.

The numerical solutions are computed with uniform and adapted distribution of points as illustrated in Figure 15. The adapted grids are obtained with CVT [32, 33], which allow us to distribute the nodes following a prescribed density function. We consider a uniform density for the uniform grids and a density proportional to the norm of the gradients of the exact solution for the adapted grids.

The numerical solution is computed for each grid with the LME ($\gamma = 2$) and SME ($\alpha = 1.4$) approximation schemes. The convergence of the L_2 norm is plotted for both uniform (Un) and adapted (Ad) grids in Figure 14(right). The guiding slopes $m = 2$ and $m = 3$ for the first and second-order approximants, respectively, are also depicted. It can be seen that both the LME and
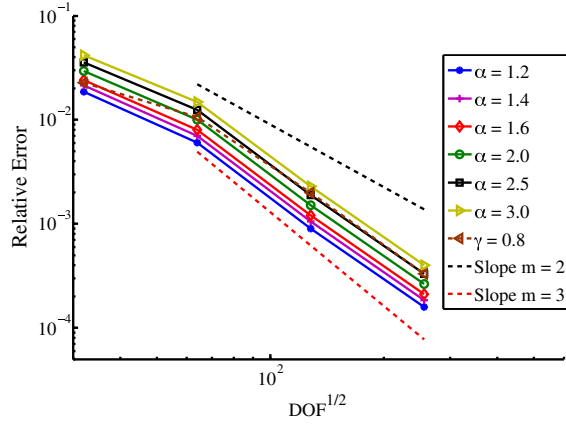
Figure 16. Relative error for the Willmore energy of a sphere.

SME methods, with adapted and uniform grids, exhibit the optimal rate of convergence. Note that the CVT method does not create nested grids upon refinement and the notion of nested approximants does not apply to meshfree methods in general. It can also be observed that for this example, using adapted grids reduces the error by roughly one order of magnitude for a given number of degrees of freedom.

4.3. Willmore energy of a sphere

The Willmore energy, which is closely related to the curvature strain energy of thin elastic sheets, is defined by

$$E = \int_\Omega H^2 \mathrm{d}\Omega,$$

where H is the mean curvature and $d\Omega$ is the area element of the surface Ω. This functional involves second-order spatial derivatives of the surface parameterization.

We study the numerical behavior of SME approximants in the calculation of the Willmore energy of a sphere. The parametric description of the surface from the scattered set of nodes, as well the integration on the surface, is performed with the method recently proposed in [15, 29]. This method combines three ingredients: (1) the automatic detection of the local geometric structure of the manifold by statistical learning methods (local principal component analysis in [15] and a variation of locally linear embedding in [29]), (2) the local parameterization of the surface using smooth approximants, and (3) patching together the local representations by means of a partition of unity. In this example, we build the local parameterizations with SME approximants. We refer to [15, 29] for details about numerical aspects and for a discussion about the accuracy of the methodology in comparison with subdivision finite elements and discontinuous Galerkin methods.

The convergence of the relative error for the elastic energy of the sphere is shown in Figure 16. The relative error is computed as $e_{rel} = |16\pi - E^h|/16\pi$ because the exact elastic energy of a sphere is $E_{\text{sphere}} = 16\pi$. The accuracy of the results increases as the locality of the SME basis functions increases (i.e., α decreases), a fact that was also noticed by solving the problem with LME approximants [15]. The curve for the LME basis functions with $\gamma = 0.8$ again closely follows that of SME approximants for $\alpha = 2.5$. We highlight the good behavior of the first-order LME approximation schemes with $\gamma = 0.8$ to approximate functionals depending on second-order spacial derivatives, which has also been observed in other works [15, 29, 30].

4.4. Linear Kirchhoff–Love thin shell analysis

In this example, we study the deformation of a hemispherical shell of radius $R = 10$ and thickness $h = 0.04$ subjected to radial loads $F = 2$ applied on two diametral directions, as depicted
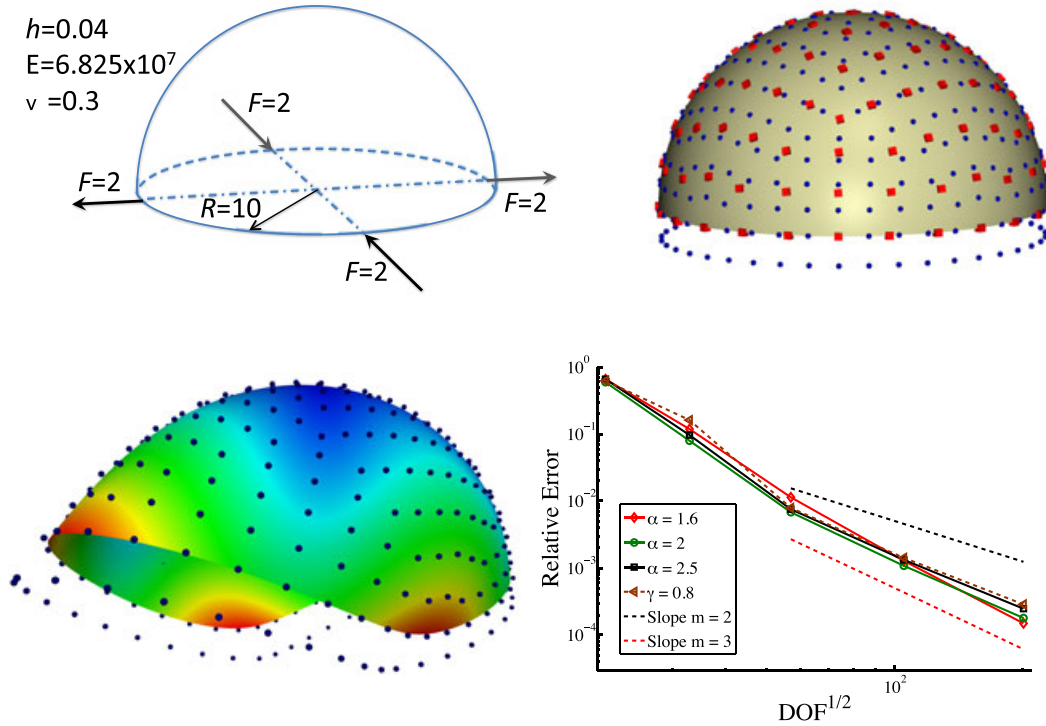
Figure 17. Sketch for the pinched hemisphere problem (top-left), discretization and reference configuration (top-right), deformed configuration (bottom-left), and results for the convergence of the normalized radial displacement (bottom-right).

in Figure 17(top-left), in the linear elastic regime. This benchmark test assesses the ability of an approximation scheme to represent inextensional deformations under complex shell bending conditions involving curvature in two directions [39].

The problem is discretized with the methodology described in the previous example. We refer to [15, 29] for details about Kirchhoff–Love thin shell formulation, numerical aspects, and the implementation. A coarse grid of 361 nodes is shown in Figure 17 (top-right). The displacement field obtained numerically is illustrated in Figure 17(bottom-left), with an amplification factor of 50. As discussed in [15], we use a row of ghost nodes at the boundary of the shell for LME approximants to avoid a bending locking behavior associated to the asymptotically flat character of the linear maximum entropy basis at the boundary.

The convergence of the relative error for the radial displacement is shown in Figure 17 (bottom-right) both for LME ($\gamma = 0.8$) and SME ($\alpha = 1.6, 2, 2.5$) approximants. The reference solution obtained with an overkill discretization is $\delta_r = 0.092401$, in agreement with the lower bound $\delta_r = 0.0924$ found in the literature [39, 40]. The conclusions that can be extracted from this restricted figure of merit are limited. The slope obtained in the asymptotic regime for the SME approximants is 3 for narrow basis functions and is slightly degraded for widespread functions. The convergence rate for the LME basis functions is slightly worse, but as before, it is remarkably accurate for a linearly consistent method and closely replicates the SME results for $\alpha = 2.5$.

4.5. Phase field modeling of biomembranes

Phase field models have been extensively used in physics and materials science to model physical systems exhibiting evolving discontinuities or internal boundaries [41]. In recent years, they are gaining popularity in a wide set of applications in applied science and engineering such as fracture [42], microstructure formation and crack propagation in ferroelectric materials [43], growth of thin films [44], and multiphase flows [45], to mention a few.

19

Vesicles are closed lipid membranes immersed in a fluid. They are simple model systems to understand membrane biophysics or can serve in biomimetic engineered systems. The membrane is an interface between two phases, the inner and outer fluids. In studying the dynamics of membranes, or the equilibrium shapes, the location of the interface is not known a priori. Such problems, referred to as free discontinuity or moving interface problems, can be described with sharp interface models, where the interface is considered as a mathematical surface. In such an approach, the interfacial physics is extrinsic to the geometrical representation of the interface, and the mechanics governing the interface evolution needs to be specified through boundary or jump conditions. These conditions can be quite complex mathematically and difficult to implement numerically.

In phase field models, an order parameter (the phase field) ϕ represents the interface; it adopts distinct values (e.g., $+1$ and -1) in each phase separated by the interface. The order parameter smoothly varies between the two values in the vicinity of the interface. Hence, it is a smeared or regularized representation of the interface, in terms of a transition parameter. The phase field equilibrium or evolution is governed by a (nonlinear often high order) PDE defined in the ambient space. One key feature of phase field models is that this PDE accomplishes simultaneously two goals: (1) it forces the phase field to adopt the value $+1$ or -1 in most of the domain and to follow a well-defined profile between phases; that is, it localizes the gradients of the phase field on the interface, and (2) it models the interfacial mechanics; that is, it replaces the interface jump/boundary conditions of sharp models. Another key feature of phase field models is that the correct sharp interfacial mechanics are recovered, as the transition parameter tends to zero. This can often be proved mathematically. Phase field models can be easily coupled with other models for the ambient medium (fluids, solids, diffusion).

The Canham–Helfrich energy [46, 47] of a smooth surface Γ

$$E_{CH} = \int_\Gamma \frac{k}{2}(H - C_0)^2 \, \mathrm{d}S,$$

where k is the bending modulus, H the mean curvature, and C_0 the spontaneous curvature, has been shown to describe well the equilibrium shapes of vesicles. It is usually complemented with area and enclosed volume constraints, and equilibrium shapes are obtained by energy minimization [48]. This sharp interface model is highly nonlinear, and the Euler–Lagrange equation for this functional is a fourth-order PDE on the parametrization of the surface describing the membrane. Hence, a direct Galerkin approach based on the weak form of the equations requires C^1 basis functions. To avoid this requirement, mixed, discontinuous, and stabilized methods have been proposed to deal with higher order operators [49]. See also [50] for a computational approach to the Willmore flow with triangulated surfaces. However, the parametric discretization of this problem is particularly challenging because of the very large geometric distortions and more importantly because of the reparameterization invariance of the energy functional. This makes it difficult to control the tangential distorsions of the computational grid [51], and parametric methods tend to be fragile.

As an alternative to the classical sharp interface model, a phase field approach for the Canham–Helfrich energy has been proposed [34]. In this regularized model, the equilibrium shapes of vesicles are obtained by solving the problem

$$\text{Minimize} \quad E[\phi] = f_E \frac{k}{2\epsilon} \int_\Omega \left[\epsilon \Delta \phi + \left(\frac{1}{\epsilon}\phi + C_0 \sqrt{2} \right)(1 - \phi^2) \right]^2 \mathrm{d}\Omega$$

$$\text{subject to} \quad V[\phi] = \frac{1}{2}\left(Vol(\Omega) + \int_\Omega \phi \, \mathrm{d}\Omega \right) = V_0$$

$$A[\phi] = f_A \int_\Omega \left[\frac{\epsilon}{2}|\nabla \phi|^2 + \frac{1}{4\epsilon}(\phi^2 - 1)^2 \right] \mathrm{d}\Omega = A_0,$$

where E is the bending energy, ϕ the phase field, ϵ a small transition parameter controlling the thickness of the diffuse interface, k the bending rigidity, C_0 the spontaneous curvature, $f_E = \frac{3}{8\sqrt{2}}$ and $f_A = \frac{3}{2\sqrt{2}}$, and Ω the integration domain. The area A and volume V functionals of the vesicle

are constrained to A_0 and V_0, respectively. We consider here axisymmetric vesicles and supplement the aforementioned constraints with

$$M[\phi] = \int_\Omega \phi(z - z_0) \, \mathrm{d}\Omega = 0$$
$$\phi|_{\partial\Omega} = -1,$$

where the first equation fixes the motion of the vesicle along the axis of symmetry and the second equation provides far field conditions of the phase field.

In this energetic variational framework, the regions $\{x : \phi(x) > 0\}$ and $\{x : \phi(x) < 0\}$ represent, respectively, the inside and outside of the vesicle, whereas the level set $\{x : \phi(x) = 0\}$ gives the position of the membrane. Formal asymptotics [52] as well as rigorous mathematical analysis [53] provide the connection between the sharp interface model and this regularized theory when the parameter ϵ tends to zero. As this limit is never achieved in the numerical calculations, an error originated by the model is always present on the computed results. Despite this modeling error, and the extra computational cost associated with increasing the dimensionality of the problem (which nevertheless becomes scalar), the phase field approach leads to more robust numerical solutions, particularly in the presence of large shape or topology changes. Furthermore, this approach is easily amenable to parallel computations.

The phase field model exhibits nearly constant solutions at 1 or −1 in most of the domain and very steep localized fronts identifying the presence of the membrane. Consequently, uniform computational grids are very inadequate because to resolve the sharp gradients, a very fine numerical resolution is needed locally. In addition, the high-order spacial derivatives involved in the energy functional pose a numerical difficulty because straight Galerkin methods require C^1 continuity. Thus, an adaptive method with smooth approximants appears as an effective and simple approach to deal with this model.

Traditionally, spectral tensor product methods or finite differences have been the method of choice to address such problems [34, 54], despite the inherent difficulty in locally refining the discretization to capture the sharp features of the solution. Finite element mixed methods [55] have been proposed, which are well suited for adaptivity [56] but suffer from poor accuracy for a given computational cost. Recently, the isogeometric analysis [19] based on tensor products of NURBS approximants has produced high quality solutions for high-order phase field models [57, 58], handling successfully sharp gradients and generating phase field solutions devoid of oscillations outside of the meaningful values. This behavior has been attributed to the smoothness and nonnegativity of basis functions, features shared with SME approximation schemes. Unfortunately, adaptive isogeometric technologies based on T-Splines still present several open problems and are the topic of current research [59]. Hierarchical B-splines also offer a new avenue for adaptive calculations with smooth, nonnegative approximants [3].

We show next the ability of the SME basis functions to approximate phase field solutions, dealing in a straightforward manner with the second derivatives in the energy functional, and the adaptivity needed to resolve the sharp gradients at a reasonable computational cost. We study the axisymmetric solutions (minimizers of the energy) in a computational domain $\Omega = [0, 1.5] \times [0, 2]$ when the spontaneous curvature is $C_0 = 0$ and the area and volume constraints are $A_0 = \frac{16}{25}\pi$ and $V_0 = \frac{32}{375}\pi$. It is well known that there exist multiple branches of minimizers of the Canham–Helfrich energy [48]. For our data, two equilibrium shapes, a discocyte and a dumbbell, are possible. It is possible to numerically obtain either of these solutions, illustrated in Figures 19 and 20, by appropriately choosing the initial guess in the optimization algorithm. We refer to [30] for details about the augmented Lagrangian Newton solution of the optimization problem and the adaptive strategy based on CVT [32, 33].

We first consider the numerical approximation of the discocyte solutions with uniform grids and several transition parameters, as shown in Figure 18(left). We consider solutions with LME ($\gamma = 0.8$) and SME ($\alpha = 1.4, 1.6, 2, 3$) approximants and compare against a reference energy $E_{\text{discocyte}} = 9.12657$ obtained with the sharp interface approach. In the phase field model, it does not make sense to consider grids that do not resolve the thickness of the smeared interface. For this reason, in all cases the simulations satisfy the relation $\epsilon > 2h$. The figure shows how, for a given
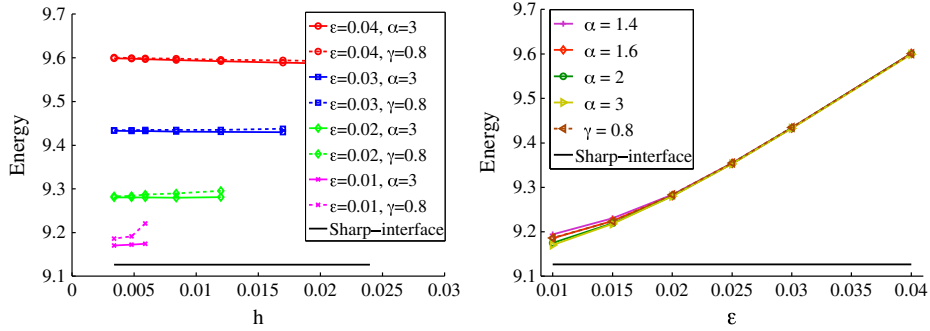
Figure 18. Dependence on nodal spacing h and the transition parameter ϵ for the phase field energy of the discocyte shape. The results are computed with uniform grids of points using local maximum entropy (LME) ($\gamma = 0.8$) and second-order maximum entropy (SME) ($\alpha = 1.4, 1.6, 2, 3$) approximants. The sharp interface energy shown in the figures is calculated with a very refined B-spline curve.
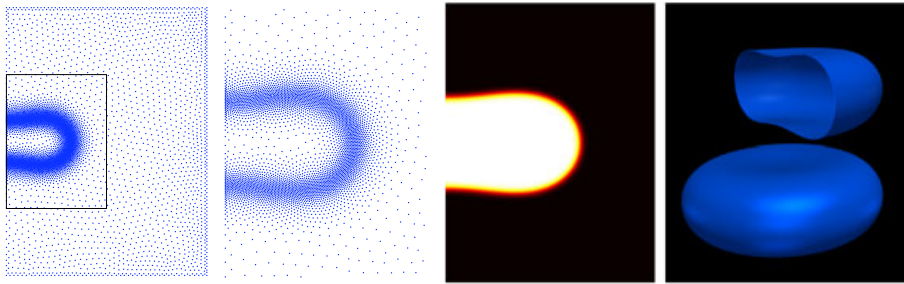

Figure 19. Discocyte solution with second-order maximum entropy (SME) approximants ($\alpha = 2$) for $\epsilon = 0.01$. From left to right: adapted grid of 6124 nodes, detail of the nodal distribution near the zero level set, illustration of the phase field, and three-dimensional reconstruction of the discocyte shape.

transition parameter, both the LME and the SME solutions converge to a limit approximate energy. It can be observed that although the second-order approximants seem to converge faster, the first-order approximants provide solutions of comparable accuracy, particularly when compared with the modeling error, that is, the difference between the limit energy for a given ϵ and the sharp interface energy. It can also be observed that the phase field energy tends to the sharp interface energy when ϵ decreases.

In Figure 18(right), we illustrate the energy of discocyte solutions as a function of the transition parameter for several slack parameters α. For this plot, the phase field solutions are computed with a uniform grid of nearly 300,000 nodes ($h = 0.0034$). The energy is more accurate for large values of α (i.e. when the basis functions are smoother and spread out further) but at the expense of a larger computational cost because the Hessian matrix bandwidth is also larger. Such a systematic numerical verification of the convergence to the sharp interface limit as ϵ tends to zero has not been reported before.

Uniform grids are not appropriate for these calculations because the transition parameter dictates a very small nodal spacing, leading to an unaffordable computational cost as ϵ decreases. Instead, Figure 19 shows the phase field solution for $\epsilon = 0.01$ in a grid of 6124 points using the SME approximants ($\alpha = 2$) and a CVT-adapted grid [30]. The energy is similar to that obtained with a uniform grid of nearly 300,000 nodes, despite the 50-fold reduction in the system complexity.

Table II reports the computed energies for the dumbbell solution with LME ($\gamma = 0.8$) and SME ($\alpha = 2, 3$) approximants in a uniform grid of nearly 300,000 points. The relative error $e_{\mathrm{rel}} = |E_{\mathrm{dumbbell}} - E_h|/E_{\mathrm{dumbbell}}$ with respect to the sharp interface energy $E_{\mathrm{dumbbell}} = 8.71756$ is also shown for several values of ϵ. That relative error includes both the discretization and the model errors. As in the previous example, it is appreciated that the error decreases as ϵ (model error) decreases and the smoothness of the basis functions (discretization error) increases.

Table II. Energies calculated for the dumbbell solution with local maximum entropy and second-order maximum entropy approximants in a uniform grid of nearly 300,000 nodes ($h = 0.0034$). The relative error with respect to the sharp interface energy is also given.

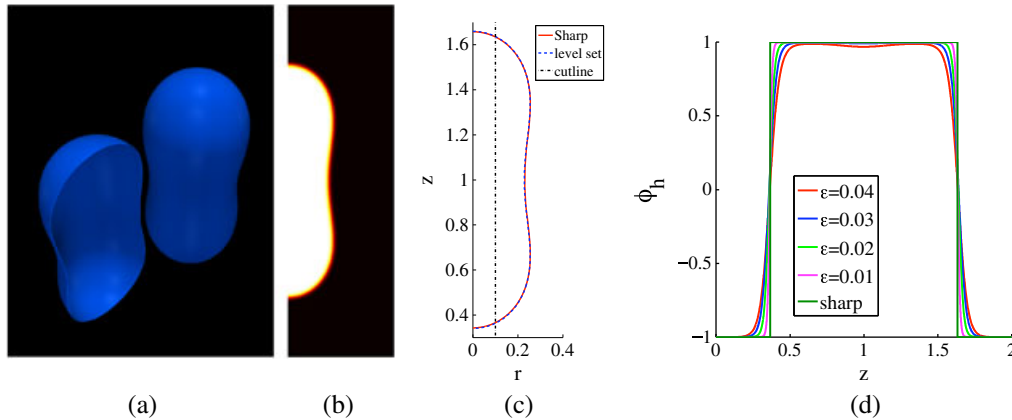| ϵ	E_h			$e_{\text{rel}}[\%]$		
	$\alpha = 2$	$\alpha = 3$	$\gamma = 0.8$	$\alpha = 2$	$\alpha = 3$	$\gamma = 0.8$
0.010	8.7690	8.7589	8.7736	0.5900	0.4739	0.6427
0.015	8.8021	8.8001	8.8063	0.9703	0.9469	1.0177
0.020	8.8577	8.8569	8.8603	1.6076	1.5989	1.6378
0.025	8.9253	8.9248	8.9271	2.3826	2.3770	2.4032
0.030	9.0012	9.0006	9.0025	3.2531	3.2470	3.2687
0.040	9.1641	9.1628	9.1651	5.1218	5.1078	5.1340

(a)　　　　(b)　　　　(c)　　　　(d)

Figure 20. From left to right: three-dimensional reconstruction of the dumbbell shape, detail of the phase field density, superposition of the zero phase field level set with the sharp interface solution, and cross section illustrating that phase field solutions are smooth and do not present oscillations (variation diminishing property).

The 3D visualization of the dumbbell solution is depicted in Figure 20(a). The phase field obtained with an adapted grid of 6124 points with the SME approximants ($\alpha = 2$) for $\epsilon = 0.01$ is also shown (b), as well as the superposition of the zero phase field level set and the sharp interface solution (c). This plot visualizes the ability of the phase field approach to mimic the sharp interface solution. Again, the accuracy, measured in the energy, for this adapted 6000 solution is not worse than that obtained with the uniform 300,000 node grid. Cross sections of the phase field solutions along the dashed cutline indicated in (c) for several values of ϵ (d) illustrate that the smooth phase field solutions do not present oscillations in the vicinity of the sharp transition and converge to the step function of the sharp interface solution. This last observation is a consequence of the variation diminishing property of the *max-ent* approximants, as has also been noticed in the isogeometric analysis of the Cahn–Hilliard equation [57].

5. CONCLUSIONS

A new method to construct second-order convex maximum entropy approximants, which overcomes the limitation of the approach presented in [6] for unstructured node sets, has been proposed. A close examination of the feasibility of the nonnegativity, the partition of unity, and the linear and quadratic consistency conditions allows us to design appropriate constraints for the entropy maximization program defining the basis functions. The nonnegativity demanded to the basis functions endows the approximants with the mathematical structure of convex geometry, leads to a probabilistic interpretation, and has as direct consequences a number of attractive features from a practical viewpoint: variation diminishing properties, smoother and easier to integrate basis functions, and a

weak Kronecker-delta property at the boundary of the convex hull of the node set. From the present and previous works, it also becomes clear that the nonnegativity requirement introduces a strong rigidity in the design of feasible constraints (consistency conditions). Further mathematical understanding of the feasibility conditions may lead to simpler methods to impose quadratic and higher order consistency conditions on *max-ent* convex approximants.

The meshfree approximants described here exhibit high smoothness and nonnegativity, claimed to be key features for the accuracy of isogeometric analysis, and can treat in a straightforward manner local refinement. The adaptive capabilities of the meshfree approximants allow us to resolve the thin layers of a phase field model for biomembranes with a 50-fold reduction in the complexity of the model as compared with uniform grids for a given accuracy. Local refinement in isogeometric analysis has been addressed to some degree in 2D [59] but remains an open issue in 3D. In fact, one of the motivations of the present work are the remarkable approximation properties reported for smooth, nonnegative basis functions in a number of applications, including previous works on maximum entropy approximants and the vast literature on isogeometric analysis. In the context of isogeometric analysis, it has been suggested that the current paradigm of improving accuracy by order elevation may not be the most appropriate for smooth approximants, and mathematical estimates of B-spline approximations highlight the importance of smoothness [60]. The results presented here point in the same direction, in the sense that despite higher convergence rates are observed for the second-order maximum entropy, first-order approximants with relatively large supports ($0.6 \leqslant \gamma \leqslant 1$) exhibit comparable (high) accuracy for the practically reasonable grids. This can be explained by an excellent pre-asymptotic behavior of the linear *max-ent* approximants and a very small multiplicative constant in the asymptotic regime. In all the applications we have considered, the simplicity of the first-order *max-ent* approximants outweighs the marginal loss of accuracy as compared with the second-order method. Surprisingly, this is also the case for fourth-order partial differential equations, such as Kirchhoff–Love shell theory and a phase field model for biomembranes, treated with a direct Galerkin method. This numerical evidence suggests that further mathematical analysis of the LME method is needed.

APPENDIX A: CALCULATION OF THE NODAL GAPS UNDERLYING B-SPLINE SCHEMES

B-spline approximants admit explicit expressions and are commonly defined through recursive algorithms. It is therefore not necessary to consider the notion of nodal gaps introduced in Section 3.1. Nevertheless, quadratic and higher degree B-splines do satisfy at least up to second-order consistency conditions, and therefore, it is possible to identify the associated nodal gaps, as in Equation (2). Identifying the nodal gaps underlying B-splines is instructive in defining appropriate constraints for *max-ent* schemes.

Before moving to the nodal gaps, as even the notion of nodes is alien to B-spline theory, we define the nodes x_a such that $x = \sum_{a=1}^{N} N_a(x) \, x_a$ where N_a are the B-spline basis functions. The nodes are thus the control points required to produce a B-spline mapping with uniform Jacobian. We collect first the nodes for a one-dimensional domain of length L and B-splines of different degree p. We introduce the quantity $h = L/(N - p)$ to help distribute the N nodes in the domain. We denote the length of the interval spanned by two consecutive nodes as $L_a = x_{a+1} - x_a$. Table A.1 shows how to distribute the nodes in the interval. Note that $L = \sum_{a=1}^{N-1} L_a$. In Figure A.1, we illustrate the location of the nodes on a one-dimensional domain of length $L = 1$ with $N = 20$ basis functions, for B-splines of order $p = 2, 3, 4, 5$.

Now, moving back to the nodal gaps, one can easily check that $x^2 \neq \sum_{a=1}^{N} N_a(x) \, x_a^2$, yet it is possible to find other nodal weights P_a such that $x^2 = \sum_{a=1}^{N} N_a(x) \, P_a$. Then, the nodal gaps follow as $d_a = x_a^2 - P_a$. Table A.2 shows the nodal gaps $\{d_a\}_{a=1,\ldots,N}$ needed to reproduce exactly x^2 in a one-dimensional domain with N basis functions. For the calculation of B-spline basis functions, we consider a uniform open knot vector of dimension $n = N + p + 1$.

Table A.1. Length of the intervals between consecutive nodes $L_a = x_{a+1} - x_a$, $a = 1, \ldots, N-1$, so that the B-splines of order p and N basis functions define a mapping with uniform Jacobian.

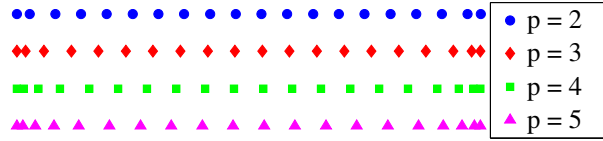|	Length of the interval			
Interval	$p = 2$	$p = 3$	$p = 4$	$p = 5$
L_1	$h/2$	$h/3$	$h/4$	$h/5$
L_2	h	$2h/3$	$2h/4$	$2h/5$
L_3	h	h	$3h/4$	$3h/5$
L_4	h	h	h	$4h/5$
L_5 to L_{N-5}	h	h	h	h
L_{N-4}	h	h	h	$4h/5$
L_{N-3}	h	h	$3h/4$	$3h/5$
L_{N-2}	h	$2h/3$	$2h/4$	$2h/5$
L_{N-1}	$h/2$	$h/3$	$h/4$	$h/5$

Figure A.1. Location of the nodes for a one-dimensional domain of length $L = 1$ with $N = 20$ B-spline basis functions of orders $p = 2, 3, 4, 5$, so that the associated B-spline mapping has uniform Jacobian.

Table A.2. Nodal gaps d_a needed to reproduce exactly x^2 with B-splines of order p in a one-dimensional domain with N basis functions.

	Nodal gaps			
Control point	$p = 2$	$p = 3$	$p = 4$	$p = 5$
x_1	0	0	0	0
x_2	$h^2/4$	$h^2/9$	$h^2/16$	$h^2/25$
x_3	$h^2/4$	$h^2/3$	$11h^2/48$	$4h^2/25$
x_4	$h^2/4$	$h^2/3$	$5h^2/12$	$17h^2/50$
x_5 to x_{N-4}	$h^2/4$	$h^2/3$	$5h^2/12$	$h^2/2$
x_{N-3}	$h^2/4$	$h^2/3$	$5h^2/12$	$17h^2/50$
x_{N-2}	$h^2/4$	$h^2/3$	$11h^2/48$	$4h^2/25$
x_{N-1}	$h^2/4$	$h^2/9$	$h^2/16$	$h^2/25$
x_N	0	0	0	0

APPENDIX B: CONSISTENCY CONDITIONS FOR SECOND-ORDER *MAX-ENT* APPROXIMANTS

Max-ent approximants are invariant with respect to rigid body transformations. However, a stable computer implementation requires that the Lagrange multipliers be invariant as well. This in turn depends on the adequate formulation of the consistency conditions. As the position of the nodes is not involved in the definition of the zeroth-order consistency condition, it is simply written as

$$\sum_{a=1}^{N} s_a(x) = 1.$$

We center the first-order consistency condition by noting that

$$\sum_{a=1}^{N} s_a(\boldsymbol{x})\,\boldsymbol{x}_a = \underbrace{\sum_{a=1}^{N} s_a(\boldsymbol{x})}_{=1}\,\boldsymbol{x},$$

hence

$$\sum_{a=1}^{N} s_a(\boldsymbol{x})\,(\boldsymbol{x}_a - \boldsymbol{x}) = \boldsymbol{0}.$$

It is also possible to center the second-order consistency condition. From

$$(\boldsymbol{x} - \boldsymbol{x}_a) \otimes (\boldsymbol{x} - \boldsymbol{x}_a) = \boldsymbol{x} \otimes \boldsymbol{x} - \boldsymbol{x}_a \otimes \boldsymbol{x} - \boldsymbol{x} \otimes \boldsymbol{x}_a + \boldsymbol{x}_a \otimes \boldsymbol{x}_a,$$

and subtracting the nodal gap $\boldsymbol{d}_a$, we obtain

$$\boldsymbol{x}_a \otimes \boldsymbol{x}_a - \boldsymbol{d}_a = (\boldsymbol{x} - \boldsymbol{x}_a) \otimes (\boldsymbol{x} - \boldsymbol{x}_a) - \boldsymbol{x} \otimes \boldsymbol{x} + \boldsymbol{x}_a \otimes \boldsymbol{x} + \boldsymbol{x} \otimes \boldsymbol{x}_a - \boldsymbol{d}_a.$$

Then, when we use the zeroth-order and first-order consistency conditions, Equation (3) becomes

$$\sum_{a=1}^{N} s_a(\boldsymbol{x})\,(\boldsymbol{x} - \boldsymbol{x}_a) \otimes (\boldsymbol{x} - \boldsymbol{x}_a) = \sum_{a=1}^{N} s_a(\boldsymbol{x})\,\boldsymbol{d}_a.$$

For notational convenience, we define $\boldsymbol{D}_a(\boldsymbol{x},\boldsymbol{x}_a,\boldsymbol{d}_a) = (\boldsymbol{x} - \boldsymbol{x}_a) \otimes (\boldsymbol{x} - \boldsymbol{x}_a) - \boldsymbol{d}_a$, and thus

$$\sum_{a=1}^{N} s_a(\boldsymbol{x})\,\boldsymbol{D}_a(\boldsymbol{x},\boldsymbol{x}_a,\boldsymbol{d}_a) = \boldsymbol{0}.$$

Following a similar procedure, Equation (4) can be rewritten as

$$\sum_{a=1}^{N} s_a(\boldsymbol{x})\,(\boldsymbol{x} - \boldsymbol{x}_a) \otimes (\boldsymbol{x} - \boldsymbol{x}_a) = \boldsymbol{G}(\boldsymbol{x}). \tag{B.1}$$

Although both Equations (3) and (B.1) are relaxed forms of Equation (1), they are not equivalent. The former only requires prescribing the nodal values $\boldsymbol{d}_a$ and can be understood as a relaxation by enlarging the convex hull, whereas the latter requires defining a continuous function $\boldsymbol{G}(\boldsymbol{x})$ defined over the domain, which shifts the interpolant inside the convex hull. It can be appreciated that both equations are coincident when the function is defined as $\boldsymbol{G}(\boldsymbol{x}) = \sum_{a=1}^{N} s_a(\boldsymbol{x})\,\boldsymbol{d}_a$.

APPENDIX C: SPATIAL DERIVATIVES OF SECOND-ORDER MAXIMUM ENTROPY APPROXIMANTS

We detail here the calculation of the first and second-order spatial derivatives for SME approximation schemes. We denote spatial gradients and Hessian matrices of scalar functions by ∇ and H, respectively. The symbol ∂ represents partial differentiation, whereas for vector valued functions, we denote by $D\boldsymbol{y}(\boldsymbol{x})$ the matrix of partial derivatives. The symbol * is used to denote that a function is evaluated at the optimal Lagrange multipliers $\boldsymbol{\lambda}^*$ and $\boldsymbol{\mu}^*$, which introduces explicit and implicit dependences on $\boldsymbol{x}$ in all functions with *. Note that what has been denoted by s_a in the rest of

the paper is denoted by s_a^* in this appendix. No implied sum is assumed for repeated node indices. Within the scope of this appendix, we define the following functions:

$$f_a(x, \lambda, \mu) = \lambda \cdot (x - x_a) - \mu : D_a,$$

$$s_a(x, \lambda, \mu) = \frac{\exp[f_a(x, \lambda, \mu)]}{\sum_{b=1}^N \exp[f_b(x, \lambda, \mu)]} = \frac{\exp[f_a(x, \lambda, \mu)]}{Z(x, \lambda, \mu)},$$

where $D_a(x, x_a, d_a) = (x - x_a) \otimes (x - x_a) - d_a$.

As shown in Section 3.3, the optimal Lagrange multipliers λ^* and μ^* minimize the dual function $g(x, \lambda, \mu) = \ln Z(x, \lambda, \mu)$, that is,

$$\{\lambda^*, \mu^*\} = \arg\min_{\lambda \in \mathbb{R}^d, \mu \in \mathbb{R}^{d \times d}} \{\ln Z(x, \lambda, \mu)\}. \tag{C.1}$$

Herein, the dependence on the variables x, λ, and μ will be omitted to simplify the notation.

Derivatives of the dual function

We solve the nonlinear problem in Equation (C.1) with the Newton–Raphson method, which requires the calculation of the gradient and the Hessian of g with respect to the Lagrange multipliers. As μ is a symmetric second-order tensor, we turn to Voigt notation to facilitate the calculation of the derivatives and convert this tensor into a vector $\tilde{\mu}$ (see Table C.1 for the procedure). In this way, the gradient $r \in \mathbb{R}^{\frac{d}{2}(d+3)}$ and the Hessian $J \in \mathbb{R}^{\frac{d}{2}(d+3) \times \frac{d}{2}(d+3)}$ are computed as

$$r = \begin{bmatrix} g_\lambda \\ g_{\tilde{\mu}} \end{bmatrix} \quad \text{and} \quad J = J^T = \begin{bmatrix} g_{\lambda\lambda} & g_{\lambda\tilde{\mu}} \\ g_{\tilde{\mu}\lambda} & g_{\tilde{\mu}\tilde{\mu}} \end{bmatrix}.$$

The derivatives $g_\lambda \in \mathbb{R}^d$ and $g_{\lambda\lambda} \in \mathbb{R}^{d \times d}$ are simply

$$g_\lambda = \sum_{a=1}^N s_a (x - x_a),$$

and

$$g_{\lambda\lambda} = \sum_{a=1}^N s_a (x - x_a) \otimes (x - x_a) - g_\lambda \otimes g_\lambda.$$

After transforming the multiplier $\mu \in \mathbb{R}^{d \times d}$ into the vector $\tilde{\mu} \in \mathbb{R}^{\frac{d}{2}(d+1)}$, the derivative $g_{\tilde{\mu}} \in \mathbb{R}^{\frac{d}{2}(d+1)}$ can be written as

$$g_{\tilde{\mu}} = -\sum_{a=1}^N s_a \tilde{D}_a,$$

where $\tilde{D}_a$ is simply the symmetric tensor D_a written in Voigt notation.

Table C.1. Voigt notation to write a symmetric tensor a_{ij} as a vector $\tilde{a}_k$.

Tensor	Dimension	Index	$d = 2$			$d = 3$					
a_{ij}	$\mathbb{R}^{d \times d}$	i	1	2	1	1	2	3	2	1	1
		j	1	2	2	1	2	3	3	3	2
$\tilde{a}_k$	$\mathbb{R}^{\frac{d}{2}(d+1)}$	k	1	2	3	1	2	3	4	5	6

Similarly, the derivative $g_{\tilde{\mu}\tilde{\mu}} \in \mathbb{R}^{\frac{d}{2}(d+1)\times\frac{d}{2}(d+1)}$ can be written as

$$g_{\tilde{\mu}\tilde{\mu}} = \sum_{a=1}^{N} s_a\, \tilde{\boldsymbol{D}}_a \otimes \tilde{\boldsymbol{D}}_a - g_{\tilde{\mu}} \otimes g_{\tilde{\mu}},$$

whereas the cross derivative $g_{\lambda\tilde{\mu}} \in \mathbb{R}^{\frac{d}{2}(d+1)\times d}$ is

$$g_{\tilde{\mu}\lambda} = -\sum_{a=1}^{N} s_a\, \tilde{\boldsymbol{D}}_a \otimes (\boldsymbol{x} - \boldsymbol{x}_a) - g_{\tilde{\mu}} \otimes g_{\lambda},$$

and $g_{\tilde{\mu}\lambda} = g_{\lambda\tilde{\mu}}^{T}$.

Gradient of the basis functions

It is readily checked that the first-order spatial derivatives of the basis functions can be written as

$$\nabla s_a^* = s_a^*\left(\nabla f_a^* - \sum_{b=1}^{N} s_b^* \nabla f_b^*\right).$$

By the chain rule, we have

$$\nabla f_a^* = (\partial_{\boldsymbol{x}} f_a)^* + D\boldsymbol{\lambda}^*\,(\partial_{\boldsymbol{\lambda}} f_a)^* + D\tilde{\boldsymbol{\mu}}^*\,(\partial_{\tilde{\mu}} f_a)^*, \tag{C.2}$$

where

$$(\partial_{\boldsymbol{x}} f_a)^* = \boldsymbol{\lambda}^* - 2\boldsymbol{\mu}^*(\boldsymbol{x} - \boldsymbol{x}_a), \quad (\partial_{\boldsymbol{\lambda}} f_a)^* = (\boldsymbol{x} - \boldsymbol{x}_a) \quad \text{and} \quad (\partial_{\tilde{\mu}} f_a)^* = -\tilde{\boldsymbol{D}}_a.$$

The terms $D\boldsymbol{\lambda}^* \in \mathbb{R}^{d\times d}$ and $D\tilde{\boldsymbol{\mu}}^* \in \mathbb{R}^{d\times\frac{d}{2}(d+1)}$ are not explicitly available but because $\boldsymbol{r}^*$ is identically zero, we also have $D\boldsymbol{r}^* = \left[Dg_{\lambda}^*\ Dg_{\tilde{\mu}}^*\right] = \boldsymbol{0}$, hence

$$Dg_{\lambda}^* = g_{\lambda x}^* + D\boldsymbol{\lambda}^* g_{\lambda\lambda}^* + D\tilde{\boldsymbol{\mu}}^* g_{\tilde{\mu}\lambda}^* = \boldsymbol{0} \in \mathbb{R}^{d\times d},$$

$$Dg_{\tilde{\mu}}^* = g_{\tilde{\mu}x}^* + D\boldsymbol{\lambda}^* g_{\lambda\tilde{\mu}}^* + D\tilde{\boldsymbol{\mu}}^* g_{\tilde{\mu}\tilde{\mu}}^* = \boldsymbol{0} \in \mathbb{R}^{d\times\frac{d}{2}(d+1)},$$

which can be rewritten as

$$\boldsymbol{0} = \left[g_{x\lambda}^*\ g_{x\tilde{\mu}}^*\right] + \left[D\boldsymbol{\lambda}^*\ D\tilde{\boldsymbol{\mu}}^*\right]\begin{bmatrix} g_{\lambda\lambda}^* & g_{\lambda\tilde{\mu}}^* \\ g_{\tilde{\mu}\lambda}^* & g_{\tilde{\mu}\tilde{\mu}}^* \end{bmatrix}.$$

Let us define

$$\boldsymbol{r}_{\boldsymbol{x}}^* = \left[g_{x\lambda}^*\ g_{x\tilde{\mu}}^*\right] \in \mathbb{R}^{d\times\frac{d}{2}(d+3)},$$

where

$$g_{x\lambda}^* = (\partial_{\boldsymbol{x}}\partial_{\boldsymbol{\lambda}} g)^* = -2\boldsymbol{\mu}^*\sum_{a=1}^{N} s_a^*\,(\boldsymbol{x}-\boldsymbol{x}_a)\otimes(\boldsymbol{x}-\boldsymbol{x}_a) + \boldsymbol{Id} = -2\boldsymbol{\mu}^* g_{\lambda\lambda}^* + \boldsymbol{Id} \in \mathbb{R}^{d\times d},$$

and

$$g_{x\tilde{\mu}}^* = (\partial_{\boldsymbol{x}}\partial_{\tilde{\mu}} g)^* = 2\boldsymbol{\mu}^*\sum_{a=1}^{N} s_a^*\,(\boldsymbol{x}-\boldsymbol{x}_a)\otimes\tilde{\boldsymbol{D}}_a = -2\boldsymbol{\mu}^* g_{\tilde{\mu}\lambda}^* \in \mathbb{R}^{d\times\frac{d}{2}(d+1)}.$$

We then have

$$\left[D\boldsymbol{\lambda}^*\ D\tilde{\boldsymbol{\mu}}^*\right] = -\boldsymbol{r}_{\boldsymbol{x}}^*\, \boldsymbol{J}^{*-1}. \tag{C.3}$$

With this expression, we can write the Equation (C.2) compactly as

$$\nabla f_a^* = \boldsymbol{\lambda}^* - 2\boldsymbol{\mu}^*(\boldsymbol{x}-\boldsymbol{x}_a) - \boldsymbol{r}_{\boldsymbol{x}}^*\,\boldsymbol{J}^{*-1}\begin{bmatrix} \boldsymbol{x}-\boldsymbol{x}_a \\ -\tilde{\boldsymbol{D}}_{a,} \end{bmatrix},$$

which finally allows us to compute the gradient of the basis functions as

$$\nabla s_a^* = -s_a^*\left\{2\boldsymbol{\mu}^*(\boldsymbol{x}-\boldsymbol{x}_a) + \boldsymbol{r}_{\boldsymbol{x}}^*\,\boldsymbol{J}^{*-1}\begin{bmatrix} \boldsymbol{x}-\boldsymbol{x}_a \\ -\tilde{\boldsymbol{D}}_a \end{bmatrix}\right\}.$$

Hessian of the basis functions

The second-order spatial derivatives of the basis functions can be written as

$$Hs_a^* = \frac{1}{s_a^*}\,\nabla s_a^* \otimes \nabla s_a^* + s_a^*\,Hf_a^* - s_a^* \sum_{b=1}^{N} \frac{1}{s_b^*}\,\nabla s_b^* \otimes \nabla s_b^* - s_a^* \sum_{b=1}^{N} s_b^*\,Hf_b^*. \qquad (C.4)$$

We only need to compute Hf_a^* because all the other terms are available from the calculation of the gradient. After taking some partial derivatives, we arrive at

$$Hf_a^* = -2\boldsymbol{\mu}^* + D\boldsymbol{\lambda}^* + (D\boldsymbol{\lambda}^*)^T + (\partial_x \partial_{\tilde{\mu}} f_a)^* (D\tilde{\boldsymbol{\mu}}^*)^T + D\tilde{\boldsymbol{\mu}}^* (\partial_{\tilde{\mu}} \partial_x f_a)^*$$
$$+ \sum_{k=1}^{\frac{d}{2}(d+1)} (D_x^2 \tilde{\mu}_k)^* (\partial_{\tilde{\mu}_k} f_a)^* + \sum_{k=1}^{d} (D_x^2 \lambda_k)^* (\partial_{\lambda_k} f_a)^*,$$

where d represents the spatial dimensio, and the subindex k refers to the $k-$th component of the Lagrange multiplier vector. Introducing the previous expression into Equation (C.4), we obtain

$$Hs_a^* = \frac{1}{s_a^*}\,\nabla s_a^* \otimes \nabla s_a^* - s_a^* \sum_{b=1}^{N} \frac{1}{s_b^*}\,\nabla s_b^* \otimes \nabla s_b^* + s_a^* (\partial_x \partial_{\tilde{\mu}} f_a)^* (D\tilde{\boldsymbol{\mu}}^*)^T$$
$$+ s_a^*\,D\tilde{\boldsymbol{\mu}}^* (\partial_x \partial_{\tilde{\mu}} f_a)^{*T} - s_a^* \sum_{k=1}^{\frac{d}{2}(d+1)} \tilde{\boldsymbol{D}}_{a_k} (D_x^2 \tilde{\mu}_k)^* + s_a^* \sum_{k=1}^{d} (x_k - x_{a_k}) (D_x^2 \lambda_k)^*,$$

where $(\partial_x \partial_{\tilde{\mu}} f_a)^* \in \mathbb{R}^{d \times \frac{d}{2}(d+1)}$, $(D_x^2 \tilde{\mu})^* \in \mathbb{R}^{d \times d \times \frac{d}{2}(d+1)}$ and $(D_x^2 \lambda)^* \in \mathbb{R}^{d \times d \times d}$ are unknowns and $D\tilde{\boldsymbol{\mu}}^*$ can be computed through the Equation (C.3). It is important to remark that $\frac{1}{s_a^*}\nabla s_a^*$ has to be calculated as $\frac{1}{s_a^*}\nabla s_a^* = -\left\{ 2\boldsymbol{\mu}^*(x - x_a) + r_x^*\,J^{*-1} \begin{bmatrix} x - x_a \\ -\tilde{\boldsymbol{D}}_a \end{bmatrix} \right\}$ to avoid error amplification when s_a^* tends to zero.

The term $(\partial_x \partial_{\tilde{\mu}} f_a)^*$ takes the form $(\partial_x \partial_{\tilde{\mu}} f_a)^* = -2(x - x_a)$ when $d = 1$. In the case of $d = 2$, the expression is given by

$$(\partial_x \partial_{\tilde{\mu}} f_a)^* = -2 \begin{bmatrix} x_1 - x_{a_1} & 0 & x_2 - x_{a_2} \\ 0 & x_2 - x_{a_2} & x_1 - x_{a_1} \end{bmatrix},$$

whereas for $d = 3$ is computed as

$$(\partial_x \partial_{\tilde{\mu}} f_a)^* = -2 \begin{bmatrix} x_1 - x_{a_1} & 0 & 0 & 0 & x_3 - x_{a_3} & x_2 - x_{a_2} \\ 0 & x_2 - x_{a_2} & 0 & x_3 - x_{a_3} & 0 & x_1 - x_{a_1} \\ 0 & 0 & x_3 - x_{a_3} & x_2 - x_{a_2} & x_1 - x_{a_1} & 0 \end{bmatrix}.$$

To compute the third-order tensors $(D_x^2 \lambda)^*$ and $(D_x^2 \tilde{\mu})^*$, we turn to the fact that $D_x^2 r^* = 0$, from which is straightforward that $D_x^2 g_\lambda^* = 0$ and $D_x^2 g_{\tilde{\mu}}^* = 0$. By considering separately the derivatives of these tensors with respect to each component of the Lagrange multipliers, we have that the $i = 1, \ldots, d$ derivatives stemming from $\boldsymbol{\lambda}$ are given by

$$D_x^2 g_{\lambda_i}^* = (\partial_x \partial_x g_{\lambda_i})^* + (\partial_x \partial_\lambda g_{\lambda_i})^* (D\boldsymbol{\lambda}^*)^T + D\boldsymbol{\lambda}^* (\partial_x \partial_\lambda g_{\lambda_i})^{*T}$$
$$+ (\partial_x \partial_{\tilde{\mu}} g_{\lambda_i})^* (D\tilde{\boldsymbol{\mu}}^*)^T + D\tilde{\boldsymbol{\mu}}^* (\partial_x \partial_{\tilde{\mu}} g_{\lambda_i})^{*T}$$
$$+ D\boldsymbol{\lambda}^* (\partial_\lambda \partial_{\tilde{\mu}} g_{\lambda_i})^* (D\tilde{\boldsymbol{\mu}}^*)^T + D\tilde{\boldsymbol{\mu}}^* (\partial_\lambda \partial_{\tilde{\mu}} g_{\lambda_i})^{*T} (D\boldsymbol{\lambda}^*)^T$$
$$+ D\boldsymbol{\lambda}^* (\partial_\lambda \partial_\lambda g_{\lambda_i})^* (D\boldsymbol{\lambda}^*)^T + D\tilde{\boldsymbol{\mu}}^* (\partial_{\tilde{\mu}} \partial_{\tilde{\mu}} g_{\lambda_i})^* (D\tilde{\boldsymbol{\mu}}^*)^T$$
$$+ \sum_{k=1}^{d} g_{\lambda_i \lambda_k}^* (D_x^2 \lambda_k)^* + \sum_{k=1}^{\frac{d}{2}(d+1)} g_{\lambda_i \tilde{\mu}_k}^* (D_x^2 \tilde{\mu}_k)^* = 0,$$

whereas the $j = 1, \ldots, \frac{d}{2}(d+1)$ derivatives associated to $\tilde{\mu}$ can be computed as

$$
\begin{aligned}
D_{\boldsymbol{x}}^2 g_{\tilde{\mu}_j}^* = {}& \left(\partial_{\boldsymbol{x}}\partial_{\boldsymbol{x}} g_{\tilde{\mu}_j}\right)^* + \left(\partial_{\boldsymbol{x}}\partial_{\boldsymbol{\lambda}} g_{\tilde{\mu}_j}\right)^* (D\boldsymbol{\lambda}^*)^T + D\boldsymbol{\lambda}^* \left(\partial_{\boldsymbol{x}}\partial_{\boldsymbol{\lambda}} g_{\tilde{\mu}_j}\right)^{*T} \\
& + \left(\partial_{\boldsymbol{x}}\partial_{\tilde{\mu}} g_{\tilde{\mu}_j}\right)^* (D\tilde{\mu}^*)^T + D\tilde{\mu}^* \left(\partial_{\boldsymbol{x}}\partial_{\tilde{\mu}} g_{\tilde{\mu}_j}\right)^{*T} \\
& + D\boldsymbol{\lambda}^* \left(\partial_{\tilde{\mu}}\partial_{\boldsymbol{\lambda}} g_{\tilde{\mu}_j}\right)^{*T} (D\tilde{\mu}^*)^T + D\tilde{\mu}^* \left(\partial_{\tilde{\mu}}\partial_{\boldsymbol{\lambda}} g_{\tilde{\mu}_j}\right)^* (D\boldsymbol{\lambda}^*)^T \\
& + D\boldsymbol{\lambda}^* \left(\partial_{\boldsymbol{\lambda}}\partial_{\boldsymbol{\lambda}} g_{\tilde{\mu}_j}\right)^* (D\boldsymbol{\lambda}^*)^T + D\tilde{\mu}^* \left(\partial_{\tilde{\mu}}\partial_{\tilde{\mu}} g_{\tilde{\mu}_j}\right)^* (D\tilde{\mu}^*)^T \\
& + \sum_{k=1}^{d} g_{\tilde{\mu}_j \lambda_k}^* \left(D_{\boldsymbol{x}}^2 \lambda_k\right)^* + \sum_{k=1}^{\frac{d}{2}(d+1)} g_{\tilde{\mu}_j \tilde{\mu}_k}^* \left(D_{\boldsymbol{x}}^2 \tilde{\mu}_k\right)^* = \boldsymbol{0}.
\end{aligned}
$$

As all the derivatives of the function g can be explicitly calculated, former expression generates $d \times d \times d$ equations with $\left[d + \frac{d}{2}(d+1)\right] \times d \times d$ unknowns, which correspond to the components of the tensors $\left(D_{\boldsymbol{x}}^2 \boldsymbol{\lambda}\right)^*$ and $\left(D_{\boldsymbol{x}}^2 \tilde{\mu}\right)^*$. On the other hand, the latter expression produces $\frac{d}{2}(d+1) \times d \times d$ equations whose unknowns are also the components of those third-order tensors. Thus, it is possible to construct an algebraic system of $\left[d + \frac{d}{2}(d+1)\right] \times d \times d$ linear independent equations to determine all the unknowns.

The expressions for the derivatives $g_{\boldsymbol{\lambda}\boldsymbol{\lambda}}^*$, $g_{\boldsymbol{\lambda}\tilde{\mu}}^*$, $g_{\tilde{\mu}\boldsymbol{\lambda}}^*$, $g_{\tilde{\mu}\tilde{\mu}}^*$ are indicated in Subsection C, whereas $D\boldsymbol{\lambda}^*$ and $D\tilde{\mu}^*$ can be computed through the Equation (C.3). The remaining derivatives are calculated by applying the following expressions:

$$
\left(\partial_{\boldsymbol{x}}\partial_{\boldsymbol{x}} g_{\lambda_i}\right)^* = 4\sum_{b=1}^{N} s_b^* \left(x_i - x_{b_i}\right) \left[\mu^*(\boldsymbol{x}-\boldsymbol{x}_b)\right] \otimes \left[\mu^*(\boldsymbol{x}-\boldsymbol{x}_b)\right],
$$

$$
\left(\partial_{\boldsymbol{x}}\partial_{\boldsymbol{\lambda}} g_{\lambda_i}\right)^* = -2\mu^*\sum_{b=1}^{N} s_b^* \left(x_i - x_{b_i}\right) (\boldsymbol{x}-\boldsymbol{x}_b) \otimes (\boldsymbol{x}-\boldsymbol{x}_b),
$$

$$
\left(\partial_{\boldsymbol{x}}\partial_{\tilde{\mu}} g_{\lambda_i}\right)^* = 2\mu^*\sum_{b=1}^{N} s_b^* \left(x_i - x_{b_i}\right) (\boldsymbol{x}-\boldsymbol{x}_b) \otimes \tilde{\boldsymbol{D}}_b + \sum_{b=1}^{N} s_b^* \left(x_i - x_{b_i}\right) \left(\partial_{\boldsymbol{x}}\partial_{\tilde{\mu}} f_b\right)^*,
$$

$$
\left(\partial_{\boldsymbol{\lambda}}\partial_{\tilde{\mu}} g_{\lambda_i}\right)^* = -\sum_{b=1}^{N} s_b^* \left(x_i - x_{b_i}\right) (\boldsymbol{x}-\boldsymbol{x}_b) \otimes \tilde{\boldsymbol{D}}_b,
$$

$$
\left(\partial_{\boldsymbol{\lambda}}\partial_{\boldsymbol{\lambda}} g_{\lambda_i}\right)^* = \sum_{b=1}^{N} s_b^* \left(x_i - x_{b_i}\right) (\boldsymbol{x}-\boldsymbol{x}_b) \otimes (\boldsymbol{x}-\boldsymbol{x}_b),
$$

$$
\left(\partial_{\tilde{\mu}}\partial_{\tilde{\mu}} g_{\lambda_i}\right)^* = \sum_{b=1}^{N} s_b^* \left(x_i - x_{b_i}\right) \tilde{\boldsymbol{D}}_b \otimes \tilde{\boldsymbol{D}}_b,
$$

$$
\left(\partial_{\boldsymbol{x}}\partial_{\boldsymbol{x}} g_{\tilde{\mu}_j}\right)^* = -2\sum_{b=1}^{N} s_b^* \left[2\tilde{\boldsymbol{D}}_{b_j} \, \mu^*(\boldsymbol{x}-\boldsymbol{x}_b) + \left(\partial_{\boldsymbol{x}}\partial_{\tilde{\mu}_j} f_b\right)^*\right] \otimes \left[\mu^*(\boldsymbol{x}-\boldsymbol{x}_b)\right]
$$
$$
\qquad\qquad -2\mu^*\sum_{b=1}^{N} s_b^* (\boldsymbol{x}-\boldsymbol{x}_b) \otimes \left(\partial_{\boldsymbol{x}}\partial_{\tilde{\mu}_j} f_b\right)^* - 2N_j,
$$

$$
\left(\partial_{\boldsymbol{x}}\partial_{\boldsymbol{\lambda}} g_{\tilde{\mu}_j}\right)^* = \sum_{b=1}^{N} s_b^* \left[2\tilde{\boldsymbol{D}}_{b_j} \, \mu^*(\boldsymbol{x}-\boldsymbol{x}_b) + \left(\partial_{\boldsymbol{x}}\partial_{\tilde{\mu}_j} f_b\right)^*\right] \otimes (\boldsymbol{x}-\boldsymbol{x}_b),
$$

$$
\left(\partial_{\boldsymbol{x}}\partial_{\tilde{\mu}} g_{\tilde{\mu}_j}\right)^* = -\sum_{b=1}^{N} s_b^* \left[2\tilde{\boldsymbol{D}}_{b_j} \mu^*(\boldsymbol{x}-\boldsymbol{x}_b) + \left(\partial_{\boldsymbol{x}}\partial_{\tilde{\mu}_j} f_b\right)^*\right] \otimes \tilde{\boldsymbol{D}}_b - \sum_{b=1}^{N} s_b^* \tilde{\boldsymbol{D}}_{b_j} \left(\partial_{\boldsymbol{x}}\partial_{\tilde{\mu}} f_b\right)^*,
$$

$$
\left(\partial_{\tilde{\mu}}\partial_{\boldsymbol{\lambda}} g_{\tilde{\mu}_j}\right)^* = \sum_{b=1}^{N} s_b^* \, \tilde{\boldsymbol{D}}_{b_j} \, \tilde{\boldsymbol{D}}_b \otimes (\boldsymbol{x}-\boldsymbol{x}_b),
$$

$$
\left(\partial_{\boldsymbol{\lambda}}\partial_{\boldsymbol{\lambda}} g_{\tilde{\mu}_j}\right)^* = -\sum_{b=1}^{N} s_b^* \, \tilde{\boldsymbol{D}}_{b_j} \, (\boldsymbol{x}-\boldsymbol{x}_b) \otimes (\boldsymbol{x}-\boldsymbol{x}_b),
$$

$$
\left(\partial_{\tilde{\mu}}\partial_{\tilde{\mu}} g_{\tilde{\mu}_j}\right)^* = -\sum_{b=1}^{N} s_b^* \, \tilde{\boldsymbol{D}}_{b_j} \, \tilde{\boldsymbol{D}}_b \otimes \tilde{\boldsymbol{D}}_b,
$$

and N_j is a $d \times d$ symmetric matrix whose component $[N_j]_{mn} = \delta_{jk}$, where the subindex k stems from the Voight Notation of the subindex mn (see Table C.1). For instance, when $d = 2$ the set of

matrices N_j is

$$\{N_j\}_{j=1,2,3} = \{N_1, N_2, N_3\} = \left\{ \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \right\}.$$

ACKNOWLEDGEMENTS

REFERENCES

1. Cottrell J, Hughes T, Bazilevs Y. *Isogeometric Analysis: Toward Integration of CAD and FEA*. John Wiley & Sons, Ltd.: New York, 2009.
2. Arroyo M, Ortiz M. Local maximum-entropy approximation schemes: a seamless bridge between finite elements and meshfree methods. *International Journal for Numerical Methods in Engineering* 2006; **65**(13):2167–2202.
3. Vuong AV, Giannelli C, Jüttler B, Simeon B. A hierarchical approach to adaptive local refinement in isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering* 2011; **200**(49–52):3554–3567.
4. Rosolen A. Arroyo M. Blending isogeometric analysis and maximum entropy meshfree approximants. In preparation.
5. Sukumar N, Wright R. Overview and construction of meshfree basis functions: from moving least squares to entropy approximants. *International Journal for Numerical Methods in Engineering* 2007; **70**(2):181–205.
6. Cyron C, Arroyo M, Ortiz M. Smooth, second order, non-negative meshfree approximants selected by maximum entropy. *International Journal for Numerical Methods in Engineering* 2009; **79**(13):1605–1632.
7. González D, Cueto E, Doblaré M. A higher-order method based on local maximum entropy approximation. *International Journal for Numerical Methods in Engineering* 2010; **83**(6):741–764.
8. Sukumar N. Construction of polygonal interpolants: a maximum entropy approach. *International Journal for Numerical Methods in Engineering* 2004; **61**(12):2159–2181.
9. Shannon C. A mathematical theory of communication. *The Bell System Technical Journal* 1948; **27**:623–656.
10. Khinchin A. *Mathematical Foundations of Information Theory*. Dover: New York, 1957.
11. Jaynes E. Information theory and statistical mechanics. *Physics Reviews* 1957; **106**:620–630.
12. Kapur J. *Maximum-entropy Models in Science and Engineering*. Wiley: New York, 1989.
13. Arroyo M, Ortiz M. *Meshfree Methods for Partial Differential Equations III, Lecture Notes in Computational Science and Engineering*, Vol. 57, chap. Local maximum-entropy approximation schemes. Springer-Verlag: Berlin Heidelberg, 2007.
14. Sukumar N, Wets RB. Deriving the continuity of maximum-entropy basis functions via variational analysis. *SIAM Journal of Optimization* 2007; **18**(3):914–925.
15. Millán D, Rosolen A, Arroyo M. Thin shell analysis from scattered points with maximum-entropy approximants. *International Journal for Numerical Methods in Engineering* 2011; **85**(6):723–751.
16. Rosolen A, Millán D, Arroyo M. On the optimum support size in meshfree methods: a variational adaptivity approach with maximum entropy approximants. *International Journal for Numerical Methods in Engineering* 2010; **82**(7):868–895.
17. Sukumar N, Moran B, Belytschko T. The natural element method in solid mechanics. *International Journal for Numerical Methods in Engineering* 1998; **43**(5):839–887.
18. Cirak F, Ortiz M, Schröder P. Subdivision surfaces: a new paradigm for thin-shell finite-element analysis. *International Journal for Numerical Methods in Engineering* 2000; **47**(12):2039–2072.
19. Hughes T, Cottrell J, Bazilevs Y. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering* 2005; **194**:4135–4195.
20. Piegl L, Tiller W. *The NURBS Book*. Springer: Berlin, 1997.
21. DeVore R. *The Approximation of Continuous Functions by Positive Linear Operators*. Springer: Berlin, 1972.
22. Bompadre A, Perotti LE, Cyron C, Ortiz M. Convergent meshfree approximation schemes of arbitrary order and smoothness. *Computer Methods in Applied Mechanics and Engineering* 2012; **221–222**:83–103.
23. Ortiz A, Puso M, Sukumar N. Maximum-entropy meshfree method for compressible and near-incompressible elasticity. *Computer Methods in Applied Mechanics and Engineering* 2010; **199**(25–28):1859–1871.
24. Ortiz A, Puso M, Sukumar N. Maximum-entropy meshfree method for incompressible media problems. *Finite Elements in Analysis and Design* 2011; **47**(6):572–585.
25. Yaw L, Sukumar N, Kunnath S. Meshfree co-rotational formulation for two-dimensional continua. *International Journal for Numerical Methods in Engineering* 2009; **79**(8):979–1003.

26. Li B, Habbal F, Ortiz M. Optimal transportation meshfree approximation schemes for fluid and plastic flows. *International Journal for Numerical Methods in Engineering* 2010; **83**(12):1541–1579.

27. Suryanarayana P, Bhattacharya K, Ortiz M. A mesh-free convex approximation scheme for Kohn–Sham density functional theory. *Journal of Computational Physics* 2011; **230**(13):5226–5238.

28. Cyron C, Nissen K, Gravemeier V, Wall W. Information flux maximum-entropy approximation schemes for convection–diffusion problems. *International Journal for Numerical Methods in Fluids* 2010; **64**:1180–1200.

29. Millán D, Rosolen A, Arroyo M. Nonlinear manifold learning for meshfree finite deformation thin shell analysis. *International Journal for Numerical Methods in Engineering* 2012. DOI: 10.1002/nme.4403.

30. Rosolen A, Peco C, Arroyo M. An adaptive meshfree method for phase-field models of biomembranes. Part I: approximation with maximum-entropy approximants. In preparation.

31. Bompadre A, Schmidt B, Ortiz M. Convergence analysis of meshfree approximation schemes. *SIAM Journal on Numerical Analysis* 2012; **50**(3):1344–1366.

32. Du Q, Faber V, Gunzburger M. Centroidal Voronoi tessellations: applications and algorithms. *SIAM Review* 1999; **41**(4):637–676.

33. Liu Y, Wang W, Lévy B, Sun F, Yan DM, Lu L, Yang C. On centroidal Voronoi tessellation—energy smoothness and fast computation. *ACM Transactions on Graphics* 2009; **28**(4):1–17.

34. Du Q, Liu C, Wang X. A phase field approach in the numerical study of the elastic bending energy for vesicle membranes. *Journal of Computational Physics* 2004; **198**:450–468.

35. Cottrell J, Reali A, Bazilevs Y, Hughes T. Isogeometric analysis of structural vibrations. *Computer Methods in Applied Mechanics and Engineering* 2006; **195**:5257–5296.

36. Hughes T. *The Finite Element Method: Lineal Static and Dynamic Finite Element Analysis*. Dover Publications: New York, 2000.

37. Born M, Huang K. *Dynamical Theory of Crystal Lattices*. Oxford University Press: Oxford, 1954.

38. Hughes T, Reali A, Sangalli G. Duality and unified analysis of discrete approximations in structural dynamics and wave propagation: comparison of p-method finite elements with k-method NURBS. *Computer Methods in Applied Mechanics and Engineering* 2008; **197**(49–50):4104–4124.

39. Belytschko T, Stolarski H, Liu W, Carpenter N, Ong J. Stress projection for membrane and shear locking in shell finite-elements. *Computer Methods in Applied Mechanics and Engineering* 1985; **51**:221–258.

40. MacNeal R, Harder R. A proposed standard set of problems to test finite element accuracy. *Finite Element in Analysis and Design* 1985; **1**(1):3–20.

41. Steinbach I. Phase-field models in materials science. *Modelling and Simulation in Materials Science and Engineering* 2009; **17**:073 001.

42. Francfort G, Marigo JJ. Revisiting brittle fracture as an energy minimization problem. *Journal of the Mechanics and Physics of Solids* 1998; **46**:1319–1342.

43. Abdollahi A, Arias I. Phase-field modeling of the coupled microstructure and fracture evolution in ferroelectric single crystals. *Acta Materialia* 2011; **59**(12):4733–4746.

44. Rätz A, Ribalta A, Voigt A. Surface evolution of elastically stressed films under deposition by a diffuse interface model. *Journal of Computational Physics* 2006; **214**:187–208.

45. Jacqmin D. Calculation of two-phase Navier–Stokes flows using phase-field modeling. *Journal of Computational Physics* 1999; **155**:96–127.

46. Canham P. The minimum energy of bending as a possible explanation of the biconcave shape of the human red blood cell. *Journal of Theoretical Biology* 1970; **26**(1):61–81.

47. Helfrich W. Elastic properties of lipid bilayers: theory and possible experiments. *Zeitschrift für Naturforschung C* 1973; **28**(11):693–703.

48. Seifert U. Configurations of fluid membranes and vesicles. *Advances in Physics* 1997; **46**(1):13–137.

49. Engel G, Garikipati K, Hughes T, Larson M, Mazzei L, Taylor R. Continuous/discontinuous finite element approximations of fourth-order elliptic problems in structural and continuum mechanics with applications to thin beams and plates, and strain gradient elasticity. *Computer Methods in Applied Mechanics and Engineering* 2002; **191**:3669–3750.

50. Dziuk G. Computational parametric Willmore flow. *Numerische Mathematik* 2008; **111**:55–80.

51. Feng F, Klug W. Finite element modeling of lipid bilayer membranes. *Journal of Computational Physics* 2006; **220**(1):394–408.

52. Wang X. Phase field models and simulations of vesicle bio-membranes. *PhD Thesis*, Department of Mathematics, The Pennsylvania State University, Pennsylvania, USA, 2005.

53. Bellettini G, Mugnai L. Approximation of Helfrich's functional via diffuse interfaces. *SIAM Journal on Mathematical Analysis* 2010; **42**:2402–2433.

54. Du Q, Liu C, Ryham R, Wang X. Energetic variational approaches in modeling vesicle and fluid interactions. *Physica D* 2009; **238**:923–930.

55. Du Q, Zhu L. Analysis of a mixed finite element method for a phase field bending elasticity model of vesicle membrane deformation. *Journal of Computational Mathematics* 2006; **24**:265–280.

56. Du Q, Zhang J. Adaptive finite element method for a phase field bending elasticity model of vesicle membrane deformations. *SIAM Journal on Scientific Computing* 2008; **30**(3):1634–1657.

57. Gomez H, Calo V, Bazilevs Y, Hughes T. Isogeometric analysis of the Cahn–Hilliard phase-field model. *Computer Methods in Applied Mechanics and Engineering* 2008; **197**:4333–4352.

58. Gomez H, Hughes T, Nogueira X, Calo V. Isogeometric analysis of the isothermal Navier–Stokes–Korteweg equations. *Computer Methods in Applied Mechanics and Engineering* 2010; **199**:1828–1840.
59. Bazilevs Y, Calo V, Cottrell J, Evans J, Hughes T, Lipton S, Scott M, Sederberg T. Isogeometric analysis using T-splines. *Computer Methods in Applied Mechanics and Engineering* 2010; **199**(5–8):229–263.
60. Beirão da Veiga L, Buffa A, Rivas J, Sangalli G. Some estimates for $h - p - k$ refinement in isogeometric analysis. *Numerische Mathematik* 2011; **118**:271–305.