Chapter 3
Educated local meshfree interpolation

The results presented in this chapter are included in the paper "Gradient-based strategies for implementing local radial basis functions in three dimensional geometries" submitted to the **International Journal for Numerical Methods in Engineering**.

Scattered data interpolation is central during the numerical solution of partial differential equations (PDEs). Solution's consistency, stability and convergence are influenced by the data distribution, geometry and the order of the approximation [94–96]. Current numerical methods, elaborated towards successful approximations in moderate computation times, can be classified into mesh-dependent or meshless methods. Spectral elements, finite differences, finite element and boundary elements are well established numerical methods that require a mesh, supporting the interpolation scheme [97]. Important efforts have been made to generate platforms that incorporate optimal meshing algorithms within numerical libraries (like Libmesh [89]) or to generalize conventional methods for moving boundaries (like extended finite elements [98] or Green's function based accelerated methods [99–104]).

Meshless methods offer flexibility in regards to mesh construction, thereby resulting in platforms for studying complex geometries, moving and deformable boundaries. The center of these methods is to approximate the solution a using discrete set of nodes that are not necessarily related through a structured mesh. Therefore, specific distribution of points, representing domain and boundaries, are used to solve PDEs [105] through global or local interpolants [106, 107].Radial Basis Functions

(RBF) is a commonly used meshless method [108], that may be used in n-dimensional problems and uses Euclidean distances to establish an interpolant with a radial basis set.

Given the nature of the radial basis, with global or compact support, the interpolant does not converge exponentially like in eigenfunction expansions. Therefore, a large number of points are needed for acceptable approximations, resulting in computationally inefficient full matrix schemes [109]. However, RBF approaches are among the preferred methods for image correction and extrapolation, large scale weather analysis and transport phenomena. An interesting possibility to increase computational efficiency is to use a local set of points, or stencil, for the RBF interpolant. This results in sparse matrices that decrease the solution time by orders of magnitude. However, local schemes contain additional complications that need to be addressed; the most notable are node distribution and stencil size [110–113], matrix conditioning [114, 115] and optimum shape parameter [116–125]. The latter arises from the free parameter dependent basis required for successful localized schemes.

In this chapter, we propose a new method to optimize local RBF (LRBF) interpolants in three dimensional geometries. This scheme uses information from the highest order derivative operator to optimize the free (shape) parameter. The comparative value to extract the optimum shape parameter may be calculated from a known analytical function or from a shape parameter-free global RBF interpolant. The proposed method is applicable to problems that require a large number of spatial approximations, like time dependent problems or stochastic integrations. This chapter is organized as follows: a short review of the RBF method is presented, followed by a local approximation of an analytical expression in a spherical and a cubic domain. The method is then fully applied in a Ginzburg-Landau minimization of a liquid crystal free energy functional. The manuscript ends with a summary of the most relevant conclusions.

3.1 Radial Basis Functions

Radial Basis Functions were originally used to generate meshless global interpolants [126, 127], evolving in a method to solve partial differential equations [128, 129]. Studies concerning its mathematical implications [130, 131, 128, 132–134] and its applicability [135–143] have emerged

in the last couple of decades, where different aspects of the method are discussed, *e.g.* interpolation matrix's conditionality, optimal shape parameter, node distribution and boundary conditions.

The method starts by assuming that any given function $f(\mathbf{x}_0)$ can be expressed as a linear combination of known radial basis functions [107] as follows:

$$f(\mathbf{x}_0) = \sum_{j=1}^{N} \alpha_j \phi(m, ||\mathbf{x}_0 - \mathbf{x}_j||) + P_{m-1}(\mathbf{x}_0), \tag{3.1}$$

where the radial basis functions, $\phi(m, ||\mathbf{x}_0 - \mathbf{x}_j||)$, depend on the Euclidean distance to N trial centers $\mathbf{x}_j \in X$ ($X \subset \Re^n$ and $j = 1, ..., N$), m is the order of the basis and P_{m-1} are $(m-1)$*th* order polynomials used to augment the interpolation [144, 145]. These polynomials additional constraints to the interpolation, *i.e.*

$$\sum_{j=1}^{N} \alpha_j P_k(\mathbf{x}_j) = 0, \quad \text{for} \quad 1 \leq k \leq m-1. \tag{3.2}$$

The above interpolation problem can be written in matrix form as $\mathbf{A} \cdot \mathbf{y} = \mathbf{b}$ with

$$\mathbf{A} = \begin{bmatrix} \mathbf{\Phi} & \mathbf{P} \\ \mathbf{P}^T & 0 \end{bmatrix}, \tag{3.3}$$

$\mathbf{y}^T = (\boldsymbol{\alpha}, \boldsymbol{\beta})$ and $\mathbf{b}^T = (\mathbf{f}, 0)$, where $\boldsymbol{\beta}$ are the polynomial coefficients.

Among the portfolio of radial basis functions, we find the thin-plate spline (TPS-m),

$$\phi(r) = r^{2m-2} \log r, \tag{3.4}$$

and the generalized multiquadratics (GMQ),

$$\phi(r) = \left(\varepsilon^2 + r^2\right)^{m/2}, \tag{3.5}$$

where $r = ||\mathbf{x}_0 - \mathbf{x}_j||$ and ε is the shape or free parameter that influences the flatness of the radial function [116, 119]. The inverse multiquadrics are obtained for $m < 0$ in the GMQ and they are positive definite functions. On the other hand, the TPS and the multiquadric ($m > 0$ in

the GMQ) are conditionally positive definite functions of order m. They require the addition of the $(m-1)$*th* polynomial to obtain an invertible interpolant. In this chapter, we use 2nd, 3rd and 4th order TPS (TPS-2, TPS-3, TPS-4, respectively) and a first order multiquadratic (MQ).

In a localized radial basis function (LRBF) interpolation, only a set of the p nearest neighbors to the center $\mathbf{x}_0$ are considered:

$$I_0 f(\mathbf{x}_0) = \sum_{j \in I_0} \alpha_j \phi(m, ||\mathbf{x}_0 - \mathbf{x}_j||), \tag{3.6}$$

where I_0 is a vector containing the information of the p nearest points of $\mathbf{x}_0$, conforming a stencil of size $q = p + 1$. Using the LRBF to each of the N discrete points results in N linear systems $q \times q$, contrary to the $N \times N$ system in the global RBF. (A couple of reviews that compare local and global approaches can be found in [109, 146]).

The approximation of gradients can be done through direct [147] (Kansa's) or quadrature [148, 149] schemes. In the direct method, the differential operator is applied *directly* to the basis while keeping the $\boldsymbol{\alpha}$ coefficients constant. For instance, the gradient is approximated by

$$I_0 \nabla f(\mathbf{x}_0) = \sum_{j \in I_0} \alpha_j \nabla \phi(m, ||\mathbf{x}_0 - \mathbf{x}_j||). \tag{3.7}$$

On the other hand, in a quadrature approximation, the differential field is considered to be a linear combination of neighboring nodes interpolations. New coefficients are found by solving a system conformed by ϕ and $\phi^{(n)}$, where the superscript n indicates the order of the differentiation. Therefore, the n gradient of the function is calculated by

$$f^{(n)}(\mathbf{x}_0) = \sum_{j=1}^{p} \lambda_j^{(n)} f(\mathbf{x}_j), \tag{3.8}$$

where $j \in I_0$, $\mathbf{x}_j \neq \mathbf{x}_0$ and the coefficients $\lambda_j^{(n)}$ are obtained from

$$\phi^{(n)}(m, ||\mathbf{x}_0 - \mathbf{x}_i||) = \sum_{j=1}^{p} \lambda_j^{(n)} \phi(m, ||\mathbf{x_i} - \mathbf{x_j}||), \tag{3.9}$$

for $i, j \in I_0$ and $\mathbf{x}_i, \mathbf{x}_j \neq \mathbf{x}_0$.

3.2 Shape parameter optimization

As mentioned before, LRBF does not result in full matrices, thereby increasing computational efficiency. However, it has been proved that only shape-dependent radial basis are able to provide good approximations, which introduces the need to find optimum shape values that depend on geometry, operator and point distribution [150, 118, 119, 151]. For instance, there is a known relationship between the interpolation matrix condition number, κ, and the shape parameter ε. It is mediated by an uncertainty principle [152], that suggests a good approximation when ε results in a matrix with a large κ. An optimum value for the shape parameter avoids the ill-conditioning of the matrix and the onset of the Runge phenomenon [119, 120, 151].

Apart from the matrix conditionality, when chosen properly, the shape parameter accelerates convergence [153] and transient solutions are stable [118, 119, 154]. Unfortunately, selecting a proper shape parameter is an open problem despite the abundant algorithms in the literature [120, 123, 124, 133, 155, 125, 156]. The first attempts to calculate ε involved the mean distance between the discrete points [126, 127] and their distribution [118, 157, 113, 123]. Most of these methods are limited to two-dimensional problems. Most importantly, they are problem-specific, restricting their general applicability. There are some efforts that used LRBF for the solution of PDEs in three-dimensions [158, 159], where the free parameter is obtained according to the maximum nodal distance [156].

Our primary interest is to use LRBF for the solution of time dependent transport problems or during stochastic integrations. Complex geometry and moving boundaries are among the issues we wish to resolve by a meshless approach. Given the lack of generality for some methods previously reported, we proceed to propose a scheme based on gradient operators. Such a method does not consider the condition number nor the node distribution. In problems where analytical expressions for the gradient are unknown, global shape parameter-free RBF are used for the"reference" gradient value. This, of course, forces a full matrix scheme to be used during a pre-processing stage of the solution.

The proposed method seeks for the ε value that minimizes the error in an approximation of a gradient operator. The order of this operator is selected from the highest differential order present

in the problem of interest. The shape parameter is then selected for each stencil at $\mathbf{x}_i$ by

$$\min_{\varepsilon_i \in \Re} \left\{ \frac{1}{q} \sqrt{\sum_{j \in I_i} \left[f^{(n)}\left(\mathbf{x}_j\right) - \hat{f}^{(n)}\left(\mathbf{x}_j, \varepsilon_i\right) \right]^2} \right\}, \tag{3.10}$$

where I_i is a vector with the q elements of the stencil at $\mathbf{x}_i$, n is the order of the highest gradient and $\hat{f}$ is the known value.

3.3 Results

To validate and test numerical consistency and stability, the LRBF shape parameter optimization is used to interpolate an analytical function and its gradients and to perform a Ginzburg-Landau (GL) free energy relaxation of a confined liquid crystal (LC). Cubic and spherical three-dimensional domains are used for the analytical expressions, while the LC is confined in a spherical droplet. The RBF interpolant is form with uniform and randomly distributed points.

For the global RBF, first order MQ and TPS (with $m = 2, 3$ and 4) were used. The global RBF was used with and without the $(m-1)$ polynomial augmentation. The gradients in the global scheme are approximate following a direct method. In the local scheme, a MQ function was used and gradients are evaluated with the direct and the quadrature approaches. The condition number of the interpolation matrix was monitored to account for its singular character and to compare with previous shape parameter optimizations. The condition number of the matrix $\mathbf{A}$ is approximated by its singular value decomposition as follows:

$$\kappa = ||\mathbf{A}|| \left|\left|\mathbf{A^{-1}}\right|\right| \approx \frac{\sigma_{\mathsf{max}}}{\sigma_{\mathsf{min}}}. \tag{3.11}$$

In general, the condition number for the global and local schemes increases as the number of discrete points (nodes) increases. Its value always remained finite, ensuring the correct solution in the linear system. In addition, increasing the interpolation order also increases the matrix conditionality. The condition number was between 10^3 to 10^{12} for node numbers between 100 and 5,000, despite the type of geometry and node distribution.

The approximation error is evaluated with a L_2 norm by

$$\mathrm{L}_2 = \frac{1}{N}\sqrt{\sum_{i=1}^{N}\left(f_i - \hat{f}_i\right)^2}, \tag{3.12}$$

where $\hat{f}_i$ is the analytical or global function and f_i is the LRBF approximation value. L_2 norms are also used for gradients, laplacian and the LC free energy functional. In general, interpolation errors were order 10^{-11} in all functions, geometries and node distributions.

Before we proceed with the numerical results, a discussion on computational efficiency is in order. CPU time was measured during *preparation* and *production* stages. Preparation includes matrix assembly, shape parameter optimization and matrix factorization for an LU decomposition solver. Production is defined as the corresponding LU back-substitution to obtain the $\mathbf{y}$ vector. Figure 3.1 shows CPU time as a function the number of nodes for a representative spherical three-dimensional domain with randomly distributed nodes. The stencil size was 50 for the LRBF. Two LU solvers were used in the local scheme: N systems of size $q \times q$ with conventional Lapack routines [160] and a single $N \times N$ sparse system using SuperLU [161, 162]. As expected, CPU times for global RBF shows a stronger dependency on number of nodes for preparation and production. Local schemes, with the non-sparse solver, have a weaker dependency on number of nodes as the LRBF should approach an $O(N)$ as $N \to \infty$. Sparse solutions are faster than N non-sparse ones as $N \to 0$; however, their computational cost also have a strong dependency as the number of nodes increases. For problems where no analytical gradients are known, a global solution is required for the local shape parameter optimization. Therefore, the preparation time with a particular number of nodes is the sum of both, global and local (also affecting the required memory allocation). An approach, to avoid extensive preparation times, is to use a coarser approximation of the global scheme, while keeping a finer resolution on the local.

3.3.1 Analytical Functions

We start the validation process by interpolating an analytical functions and its gradients, simulating the interpolation of scattered data. The numerical performance and convergence are tested using

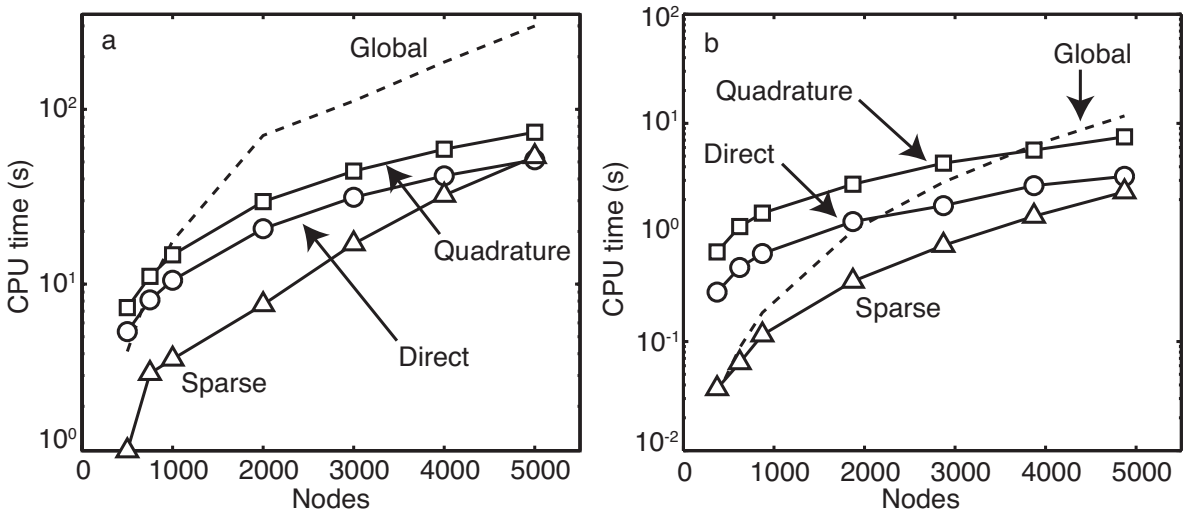

Figure 3.1. CPU time for global and local RBF schemes in a 3D sphere with random nodes: (a) Preparation and (b) Production.

different node distributions, geometries and local stencil size. The three-dimensional domains were chosen with a characteristic size $\hat{L} = 1$.

Two analytical functions are defined by:

$$f_1(\mathbf{x}) = x_1 \exp(-x_1) + x_2 \exp(-x_2) + x_3 \exp(-x_3),$$
$$f_2(\mathbf{x}) = x_1 \sin(x_1) + x_2 \cos(x_2) + x_3 \sin(x_3), \tag{3.13}$$

where $\mathbf{x} = (x_1, x_2, x_3)$. The f_1 interpolation matrix condition number, its gradients and laplacian will be used to optimize the shape parameter for the MQ basis. These criteria are a function of the geometry, cube or sphere, node density and node distribution, uniform or random. Once shape parameters are obtained from the f_1 interpolants, f_2, ∇f_2 and $\nabla^2 f_2$ are approximated.

We first evaluate the optimization method in a global RBF interpolation. Figure 3.2 summarizes the L_2 errors for ∇f_2 and $\nabla^2 f_2$, when optimized shape parameters from f_1 are used. The function interpolation errors, for every scheme, were order 10^{-11}. In general, errors decrease as the number of nodes is increased.

Shape parameter optimization with the f_1 matrix condition number results in poor gradient and laplacian approximations; this scheme was proposed before and it is only successful for

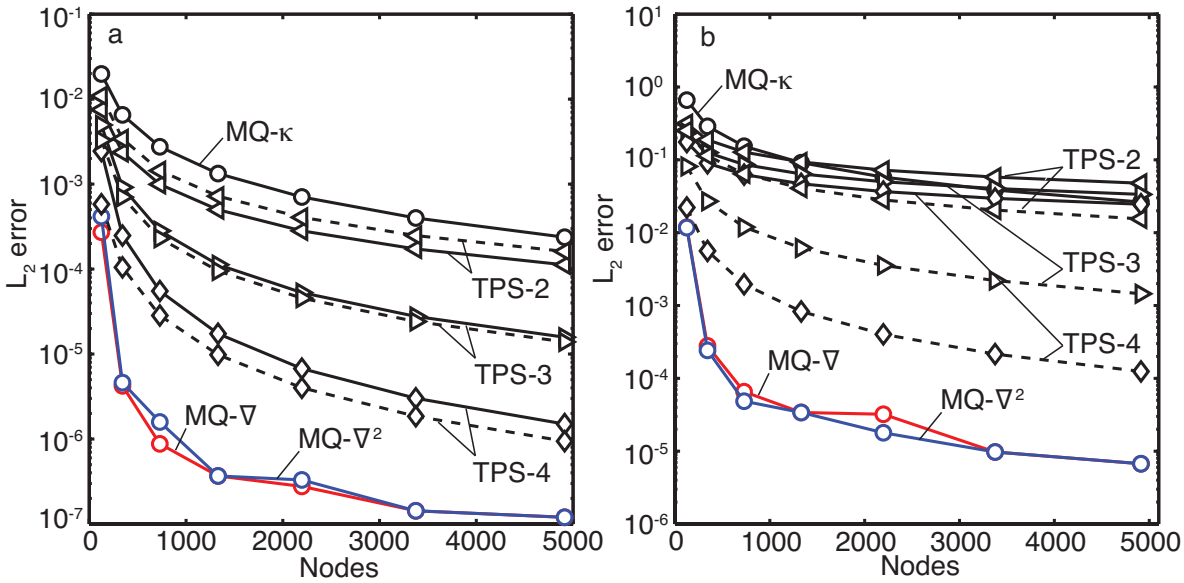

Figure 3.2. Interpolation errors from global RBF schemes: (a) ∇f_2 and (b) $\nabla^2 f_2$. The function is approximated in a cubic domain using uniform distributed nodes. The MQ shape parameter is obtained from the f_1 matrix condition number, gradients and laplacian. Dotted lines are for the polynomial augmentation when TPS are used.

two-dimensional domains [136, 133]. When f_1 gradients or laplacian are used to find an optimal shape parameter, the error in the f_2 gradient interpolation is 10^{-7}, while in the laplacian is 10^{-5}. For comparison, shape-free TPS global approximations are also included in Fig. 3.2. As the TPS order increases, the approximation improves. Augmented schemes give similar results for gradient approximations, whereas they improve the laplacian interpolation considerably. Notice that MQ, optimized with f_1 gradients, provide the best solution for f_2 gradients and laplacians. These results were obtained in a cubic domain using a uniform node distribution. Global RBF for spheres and random node distributions similar equivalent results.

According to results in Fig. 3.2, the use of known differential operators provide a way to obtain general shape-dependent interpolation schemes. It is desirable that the shape parameter optimization can be achieved only from the node distribution and density and the type of geometry. Figure 3.3 shows the shape parameter optimized from the f_1 matrix condition number, gradient and laplacian in a cubic domain with random nodes. The differential operators are approximated using the direct and the quadrature schemes. The optimal shape parameter is different for the three optimization criterion and it shows little dependence on the mesh density. This is contrary to

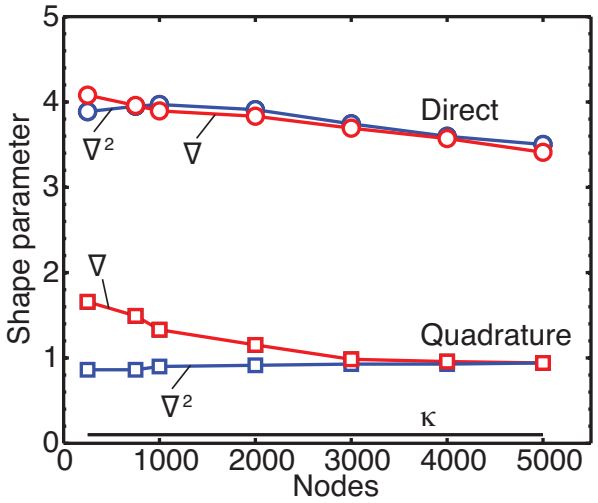

Figure 3.3. Optimal shape parameter for random meshes in a cubic domain. The shape parameter is obtained from the f_1 matrix condition number, gradients and laplacian. The differential operators are obtained from direct and quadrature schemes.

previous works, in two-dimensional domains, where the shape parameter was strongly dependent on mesh density [117, 123]. Optimizing with the κ criterion, regardless of the solution method, yields lower values of ε. Even though our optimization schemes are able to find a different shape parameter per node, we found that the values were similar and that a single value may be used. This is an important result given the implications on scheme preparation time as a single ε minimization is required.

The next step is to check whether the proposed scheme works for a LRBF scheme. Similar to the previous analysis, f_1 is used to obtain optimal shape parameters from its matrix condition number, gradients and laplacian. We start with an LRBF that uses a stencil size of 50 nodes. Figure 3.4 shows the approximation errors for ∇f_2 and $\nabla^2 f_2$. Differential operators are approximated following direct and quadrature methods.

Results in Fig. 3.4 indicate that the LRBF scheme for f_2 is able to provide good approximations from the f_1 shape parameter optimization. Similar to the global interpolation, the condition number is unable to provide acceptable approximations. We then conclude that previous methods, in two-dimensional domains, are not suitable for 3D. The second important conclusion is that the differential quadrature scheme results in poor approximation errors, despite the shape parameter optimization. Note that the best approximation to ∇f_2 is achieved when ∇f_1 is used to find the

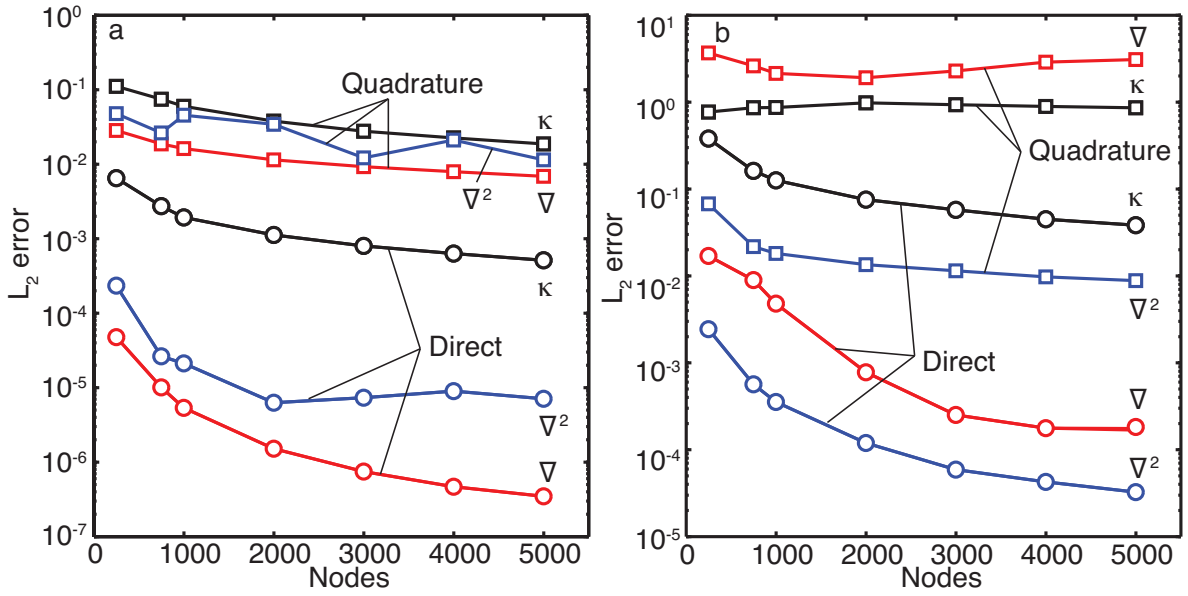

Figure 3.4. Interpolation errors from a LRBF using a MQ basis: (a) ∇f_2 and (b) $\nabla^2 f_2$. The function is approximated in a sphere using randomly distributed nodes and the local stencil has 50 nodes. The MQ shape parameter is obtained from the f_1 matrix condition number, gradients and laplacian.

optimum shape parameter. Accordingly, the best approximation to $\nabla^2 f_2$ is achieved when $\nabla^2 f_1$ is used. However, the best overall interpolation, *i.e.*when both operators are needed, is given by the $\nabla^2 f_1$ optimization. Recall that function interpolation errors were always order 10^{-11}.

Similar results were observed in cubic domains and for uniform node distributions. We do not include these results due to space restrictions. In addition, a consistency check was performed where f_2 was used to optimize the shape parameter and f_1, ∇f_1 and $\nabla^2 f_1$ were interpolated. The results were also equivalent.

The followup analysis is how the LRBF scheme is affected by the number of neighbors or stencil size. Figure 3.5 shows the interpolation errors in ∇f_2 and $\nabla^2 f_2$ as a function of the stencil size. Optimum shape parameters are obtained from the f_1 condition matrix, gradient and laplacian. Differential operators are approximated following direct and quadrature methods. In general, errors decrease as the stencil increases in size. The quadrature method is still unable to provide acceptable solutions. The proposed scheme, where the shape parameters are obtained from the differential operators, provide the best solution. Recall that as the stencil increases, the size of the

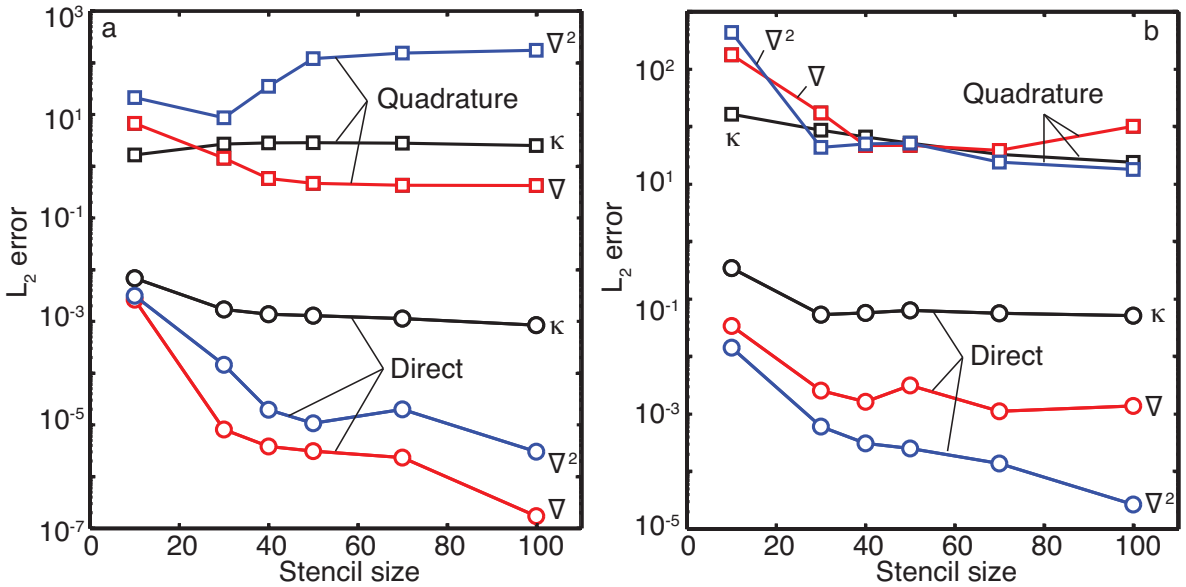

Figure 3.5. Interpolation errors from a local RBF using a MQ basis: (a) ∇f_2 and (b) $\nabla^2 f_2$. The function is approximated in a sphere using randomly distributed nodes and the total number of nodes is 3,000. The MQ shape parameter is obtained from the f_1 matrix condition number, gradients and laplacian.

linear systems $q \times q$ also increases, thereby affecting the computational efficiency of the localized method.

Results in Figs. 3.2 and 3.4 may suggest that optimum shape parameter values are a function of the geometry, node distribution and node density. We believe that this conclusion should be taken carefully. We studied multiple analytical functions (polynomials, exponential, trigonometrical) and the results were similar to those presented here. For interpolation schemes, it is promising to know that an analytical function may be used to "learn" the localized parameters according to the geometry and node distribution. This is extremely important when using RBF to interpolate and approximate functions from scattered data [163–165].

A 4-cyano-4'-pentylbiphenyl (5CB) LC is confined in a spherical droplet of radius $R = 500$ nm. For 5CB, $A = 6 \times 10^5$ J/m^3, $L_1 = 6 \times 10^{-12}$ N and $\gamma = 1 \times 10^{-2}$ Pa·s. The time derivative in the GL relaxation was integrated following a semi-implicit Euler scheme, where non-linear terms were treated explicitly. Global and localized RBF schemes are used using TPS-4 and MQ, respectively. A typical GL relaxation requires time integrations over 10^6 time steps.

The exercise is to perform GL relaxations, as introduced in section 2.3.1, with a global RBF scheme for homeotropic and planar droplets, where the anchoring strength W is changed. Then, a global RBF interpolation is used to obtain optimum shape parameters, which are used in an LRBF scheme. The final free energy densities and LC phases should be identical at the end of the simulations. During the LRBF minimization, the shape parameters were re-calculated, from a global interpolation, to verify and check the consistency of the relaxation dynamics. As a general trend, a single shape parameter optimization was required at the beginning of the local simulations. Typical computational times for the global GL relaxations were 90 hours for a 5,000 nodes, while for the LRBF relaxations were 10 hours.

Figure 3.6 shows the free energy densities and the total free energy for homeotropic and planar droplets as a function of the anchoring strength W. These free energies are calculated with the LRBF. The free energy minimization provides a good test model local optimization scheme. Landau and surface densities contain tensor powers, while the elastic densities are defined from tensor gradients. In addition, after applying the Volterra derivatives, during the GL relaxation, the elastic density results in a Laplacian operator. Therefore, the method of choice for the shape parameter optimization was the tensor laplacian from the global RBF. The results in Fig. 3.6 are in qualitative agreement with experimental and theoretical observations of phases present in liquid crystal droplets under different conditions [166, 167, 34, 142, 168, 60]. The relative error, using the global as the standard value, was order 10^{-3} (0.1%) and it was independent of W.

Figure 3.7 shows representative final liquid crystalline phases for homeotropic and planar droplets. All simulations started from a random-isotropic phase. After the GL relaxation, the LC undergoes radial and bipolar phases for homeotropic and planar anchoring, respectively. In the figure, the director field, $\mathbf{n}$, and iso-surfaces for low scalar order parameter, S, are illustrated. These contours represent the regions where defects, $\mathbf{n}$ discontinuities, are located. The radial configuration, in Figure 3.7 (a) and (b), is distinguished by a radial orientation of the director vector and a defect at the center of the droplet. The size and location of the ring defect depends on the anchoring strength W. The bipolar configuration, shown in Figure 3.7 (c) and (d), is characterized by the alignment of the director field along two opposite poles, where two defects are located.

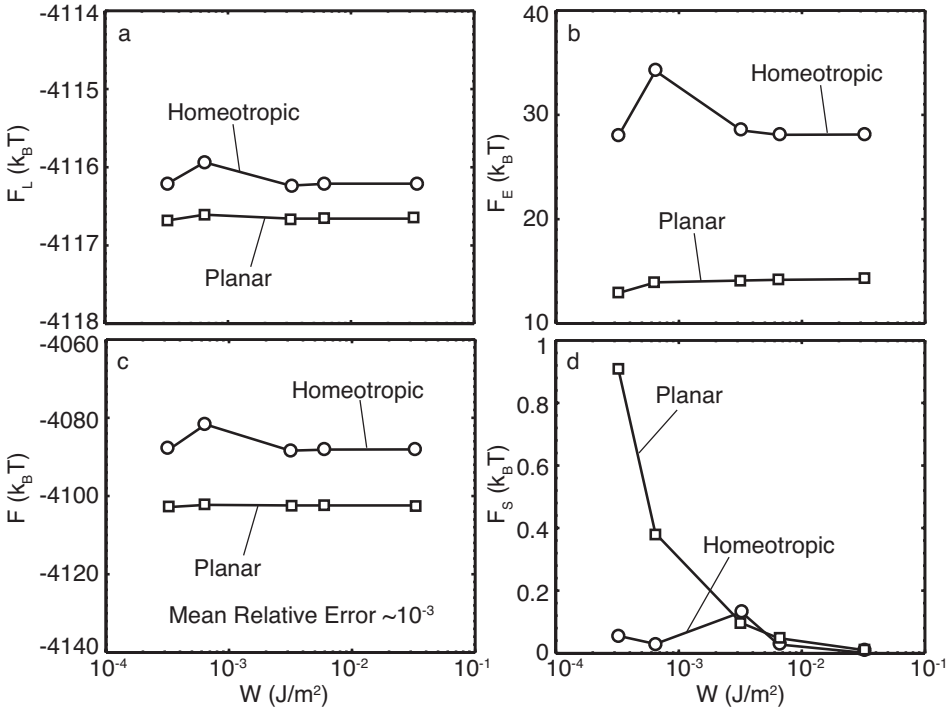

Figure 3.6. LC free energy as a function of the anchoring strength for homeotropic and planar droplets with $R = 500$ nm: (a) Landau energy, (b) elastic energy, (c) total energy and (d) surface energy.

The computational time and memory are decreased considerably in the LRBF scheme. We performed additional tests where the global mesh was kept constant, while the spatial resolution was increased in the LRBF. The mesh ratio were 1:2, 1:3 and 1:5, where the biggest local system had 25,000 nodes. We found that approximation errors were in the same order of magnitude, while local resolution and time stability was improved.

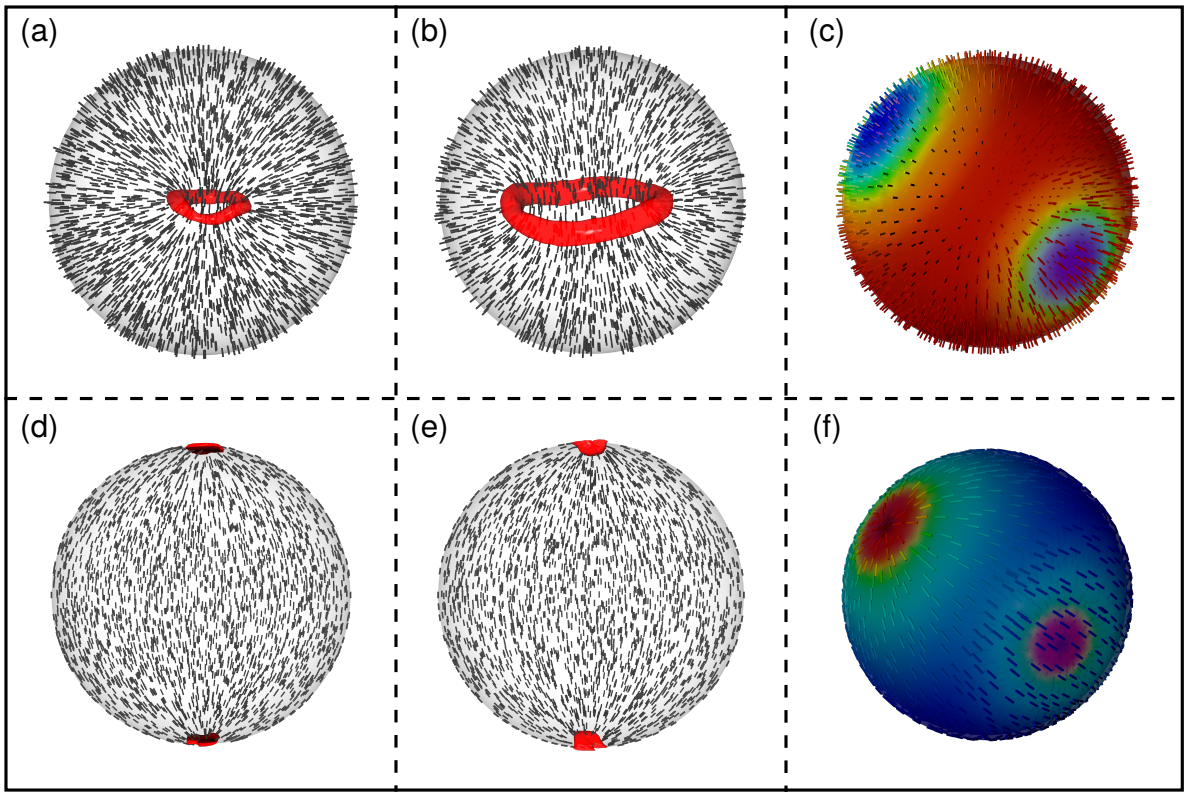

Figure 3.7. Liquid crystalline phases within homeotropic and planar droplets of size $R = 500$ nm. Radial phases with (a) $W = 3.226 \times 10^{-2}$ J/m^2 and (b) $W = 3.226 \times 10^{-4}$ J/m^2. Bipolar phases with (d) $W = 3.226 \times 10^{-2}$ J/m^2, and (e) $W = 3.226 \times 10^{-4}$ J/m^2. The red iso-surfaces represent regions where the escalar order parameter is low: $S = 0.4$. Surface contours (c) and (f) represent the alignment of the surface orientations with respect an average normal direction. Red colors represent a normal perpendicular to the average direction, while blue is for a normal parallel to the average direction. The total number of nodes was 5,000 with a random distribution; the stencil size was 50.