



Accounting for Weak Discontinuities and Moving Boundaries in the Context of the Natural Element Method and Model Reduction Techniques

Francisco Chinesta, Elías Cueto, Pierre Joyot, Pierre Villon

► To cite this version:

Francisco Chinesta, Elías Cueto, Pierre Joyot, Pierre Villon. Accounting for Weak Discontinuities and Moving Boundaries in the Context of the Natural Element Method and Model Reduction Techniques. IUTAM Symposium on Discretization Methods for Evolving Discontinuities, Springer, pp.37-55, 2007, 9781402065293. <10.1007/978-1-4020-6530-9_3>. <hal-01007470>

HAL Id: hal-01007470

<https://hal.science/hal-01007470v1>

Submitted on 16 Jun 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons CC BY 4.0 - Attribution - International License

Accounting for weak discontinuities and moving boundaries in the context of the Natural Element Method and model reduction techniques

F. Chinesta¹, E. Cueto², P. Joyot³, and P. Villon⁴

¹ LMSP UMR 8106 CNRS-ENSAM

151 Boulevard de l'Hôpital, F-75013 Paris, France

francisco.chinesta@paris.esnam.fr

² Aragn Institute of Engineering Research, University of Zaragoza. María de Luna, 7. E-50018 Zaragoza, Spain

³ LIPSI-ESTIA, Technopole Izarbel, F-64210 Bidart, France

⁴ Université de Technologies de Compiègne

BP 20529, F-60205 Compiègne cedex, France

Summary. Several thermomechanical models are defined in evolving domains involving fixed and evolving discontinuities. The accurate representation of moving boundaries and interfaces is, despite the significant progresses achieved in the recent years, an active research domain. This work focusses on the application of meshless methods for discretizing this kind of models, and in particular the ones based on the use of natural neighbor interpolations. The questions related to the description of moving boundaries, evolving weak discontinuities and the possibility of an eventual model reduction to alleviate the computational simulation cost, will be some of the topics here analyzed.

Key words: Natural Element Method, Evolving weak discontinuities, Model reduction, Karhunen-Loève, Proper Orthogonal Decomposition, Functional enrichment, Interface tracking and capturing.

1 Introduction

For models involving large transformations the use of meshless discretization techniques seems to be an appealing choice, instead of using the standard finite element method that requires frequent remeshing in order to satisfy the accuracy requirements. Moreover, if one proceeds in the context of meshless methods and all the internal thermomechanical variables are associated to the nodes, neither remeshing nor fields projections are required through the entire

simulation. In some cases, addition or deletion of nodes is required in order to capture solution singularities, to improve the solution accuracy or simply to avoid too many nodes in the region where the solution is smooth enough. In these cases, new nodes can be added without complex geometrical checks, in the regions pointed by an adequate error indicator or error estimator, as described for example in [17].

Meshfree methods based on Moving Least Squares (MLS) approximation have been subject of active research during the last decade. These include Smooth Particle Hydrodynamics, Element Free Galerkin, Diffuse Elements, Reproducing Kernel Particle and other Methods (see [1] for a nice review of those techniques). However, one of the issues is the satisfaction of essential boundary conditions. This is due to the nature of the approximation itself. In fact, the MLS nodal domains of influence are the same as those of the corresponding weighting functions, who generally do not fit the boundary. On the other hand, the Natural Neighbor (NN) approximation and associated family of computational methods [12] [3] do not present these drawbacks. The boundary approximation is obtained naturally due to the fact that NN shape functions of internal nodes vanish at the boundary where only the boundary nodes contribute. The list of connected points —the natural neighbors— is also known in advance. However, the NN do not present all the advantages of the MLS. In particular, the shape function support is geometrically complex. Moreover, the NN shape functions have only C^0 continuity at the nodes and only linear consistency is guaranteed. A common difficulty of all these techniques lies in the introduction of discontinuities of the primal variable or of its normal derivative across fixed or moving interfaces as well as the description of moving boundaries as encountered in large transformations of solids (as encountered in forming processes simulation) or in fluid flows.

In what follows, and always in the context of the Natural Element Method, we will consider different possibilities for (i) accounting for the geometrical changes, that is, the accurate representation of complex domain boundary evolutions; and (ii) accounting for fixed or moving weak discontinuities or interfaces (usually encountered in models involving change of phases, consolidation of porous media, ...).

2 The meshfree natural element method

In this section, the utility of both the constrained natural element method (C- NEM) and the α -shape based natural element method (α -NEM) to describe moving interfaces and discontinuities in a fixed cloud of nodes is discussed. After a brief review of the Voronoi-based interpolants, we introduce the constrained Voronoi diagram which is used to compute the shape functions in any domain, as well as the α -shapes based approximations functions. To avoid duplication with some of our former publications, different references to our former works will be addressed.

2.1 Natural neighbor interpolation

The construction of the natural neighbor interpolation has been analyzed in depth in some of our former works. The interested reader can refer to [3] for a review on that topic, which we briefly summarize in this section. The NEM interpolant is constructed on the basis of the Voronoi diagram. The Delaunay tessellation is the topological dual of the Voronoi diagram. The Voronoi cells related to neighbor nodes have a common edge.

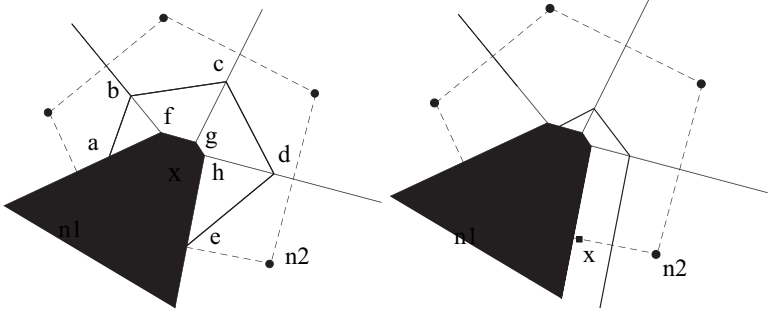


Fig. 1. Construction of the Sibson shape functions.

For the sake of simplicity from now on we focus only on the 2D case, the 3D case being a direct extension. Consider a set of nodes $S = \{n_1, n_2, \dots, n_N\}$ in \mathbb{R}^2 . The Voronoi diagram is the partition of \mathbb{R}^2 into regions T_i (Voronoi cells) defined by:

$$T_i = \{\mathbf{x} \in \mathbb{R}^2 : d(\mathbf{x}, \mathbf{x}_i) < d(\mathbf{x}, \mathbf{x}_j), \forall j \neq i\}, \quad \forall i \quad (1)$$

where $d(\cdot)$ denotes a distance.

In order to define the natural neighbour coordinates it is necessary to introduce the second-order Voronoi diagram of the cloud defined as

$$T_{ij} = \{\mathbf{x} \in \mathbb{R}^2 : d(\mathbf{x}, \mathbf{x}_i) < d(\mathbf{x}, \mathbf{x}_j) < d(\mathbf{x}, \mathbf{x}_k) \\ \forall j \neq i \neq k\}. \quad (2)$$

Sibson [13] defined the natural neighbor coordinates of a point \mathbf{x} with respect to one of its neighbors n_i as the ratio of the cell T_i that is transferred to T_x when adding \mathbf{x} to the initial cloud of points to the total volume of T_x . In other words, if $\kappa(\mathbf{x})$ and $\kappa_i(\mathbf{x})$ are the Lebesgue measures of T_x and T_{xi} respectively, the natural neighbor coordinates of \mathbf{x} with respect to the node n_i is defined as

$$\phi_i(\mathbf{x}) = \frac{\kappa_i(\mathbf{x})}{\kappa(\mathbf{x})}. \quad (3)$$

Figure 1 illustrates the construction of $\phi_1(\mathbf{x})$, that in this case is given by:

$$\phi_1(\mathbf{x}) = \frac{\text{Area}(afghe)}{\text{Area}(abcde)} \quad (4)$$

If point \mathbf{x} coincides with node n_i , i.e. $(\mathbf{x} = \mathbf{x}_i)$, $\phi_i(\mathbf{x}_i) = 1$, and all other shape functions are zero, i.e. $\phi_j(\mathbf{x}_i) = \delta_{ij}$ (δ_{ij} being the Kroenecker delta). The properties of positivity, interpolation, and partition of unity are then verified [12]:

$$\begin{cases} 0 \leq \phi_i(\mathbf{x}) \leq 1 \\ \phi_i(\mathbf{x}_j) = \delta_{ij} \\ \sum_{i=1}^n \phi_i(\mathbf{x}) = 1 \end{cases} \quad (5)$$

The natural neighbor shape functions also satisfy the local coordinate property [13], namely:

$$\mathbf{x} = \sum_{i=1}^n \phi_i(\mathbf{x}) \mathbf{x}_i \quad (6)$$

which combined with Eq. (5), implies that the natural neighbor interpolant spans the space of linear polynomials (linear completeness).

Sibson natural neighbor shape functions are C^1 at any point except at the nodes, where they are only C^0 . The C^1 continuity in the domain can be improved by using special classes of natural neighbor shape functions [5], and some ongoing works of Cueto's group allow also to improve the continuity at the nodes by computing B-splines over Voronoi diagrams.

Another important property of this interpolant is its strict linearity over the boundary of convex domains. The proof can be found in Sukumar *et al.* [12]. This result is essential to guarantee strict continuity of the approximation across material interfaces as well as the imposition of essential boundary conditions. The lack of this property is an important issue in most meshfree methods which require special numerical strategies to circumvent this drawback. As just indicated, the property of linearity of the NEM interpolant is only satisfied along convex boundaries [12]. The difficulties related to non-convex geometries can be circumvented using α -shapes [2] or introducing a visibility criterion (C-NEM) [15].

Consider an interpolation scheme for a scalar function $A(\mathbf{x}) : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$, in the form:

$$A^h(\mathbf{x}) = \sum_{i=1}^{n(\mathbf{x})} \phi_i(\mathbf{x}) A_i \quad (7)$$

where A_i are the nodal values of the field A at the $n(\mathbf{x})$ neighbor nodes of point \mathbf{x} , and $\phi_i(\mathbf{x})$ are the shape functions at that point associated with each neighbor node. It is noted that Eq. (7) defines a local interpolation scheme. Thus, the trial and test functions used in the discretization of a generic variational formulation can be approximated by Eq. (7).

2.2 The constrained natural element method – C-NEM –

In the C-NEM a visibility criterion was introduced in order to restrict influent nodes among natural neighbors [15]. The computation of the shape functions is then done on the basis of the so-called constrained (or extended) Voronoi diagram (CVD) which is the strict dual of the constrained Delaunay triangulation.

The intersection between the constrained Voronoi diagram and the domain closure is composed of cells T_i^C , one for each node n_i , such that any point \mathbf{x} inside T_i^C is closer to n_i than to any other node n_j visible from \mathbf{x} . The constrained Voronoi cells are defined formally by:

$$\begin{aligned} T_i^C &= \{\mathbf{x} \in \mathbb{R}^n : d(\mathbf{x}, \mathbf{x}_i) < d(\mathbf{x}, \mathbf{x}_j), \\ &\forall j \neq i, S_{x \rightarrow n_i} \cap \Gamma = \emptyset, S_{x \rightarrow n_j} \cap \Gamma = \emptyset\} \end{aligned} \quad (8)$$

where Γ is the domain boundary and $S_{a \rightarrow b}$ denotes the segment between the points a and b .

A generalization of the constrained Delaunay triangulation to 3D does not exist without adding new nodes. Nevertheless, some techniques for constructing 3D constrained Delaunay tessellations are available and provided in [10] and [11] by addition of Steiner points.

2.3 The α -shape based natural element method – α -NEM –

One important issue when using meshless methods, and particularly when one simulates forming processes from an updated Lagrangian formulation, is free surface tracking. Since, by definition, meshless methods do not need any explicit connectivity between nodes, and consequently the nodes belonging to the domain boundary must be identified with the help of an appropriate technique. In this section we introduce an approach based on the use of the geometrical concept of α -shapes. The concept of shape had traditionally no formal meaning, so it is possible to define a complete family of shapes of a cloud of points by introducing a parameter α that can be considered as a measure of the *level of detail* up to which the domain is going to be represented. α -shapes provide a means so as to eliminate from the triangulation those triangles or tetrahedra whose size is bigger than the before-mentioned level of detail. This criterion is very simple: just eliminate those triangles (tetrahedra) whose circum-radius is bigger than the level of detail, α . For a more in depth description the reader is referred to [2] [3] and the references therein.

In order to clarify the concepts just introduced, we present in the following paragraphs an example of α -shapes computed from a cloud of points corresponding to the simulation of an extrusion process. In this section we will restrict ourselves to geometrical concepts only. The key idea of the method proposed here is to extract the shape of the domain at each time step by invoking the concept of α -shape of the cloud. The α parameter will be obtained by

geometrical considerations. In this case the radius at the outlet of the tooling, for instance, seems to be the smallest level of detail up to which the domain must be represented. The nodal distance h must be chosen accordingly.

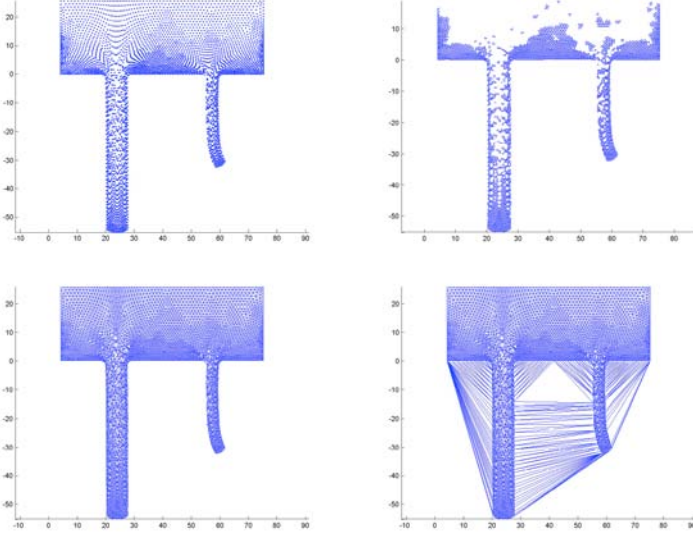


Fig. 2. Some members of the family of α -shapes of the cloud of points used in the extrusion example. (top-left) $\alpha = 0$ (the cloud of points) (top-right) $\alpha \approx 0.5h$ (bottom-left) $\alpha \approx h$ and (bottom-right) $\alpha = \infty$ (the convex hull of the set)

In Fig. 2 some members of the family of α -shapes of the cloud of points in its final configuration are depicted. In Fig. 2(a) the member for $\alpha = 0$, i.e., the cloud of points itself, is shown. Note how, as α is increased, the number and size of the simplexes (in this case, triangles) that belong to the shape is increasing. For $\alpha \approx h$ we obtain an appropriate shape for the cloud. Note, however, that this is not an exact value to be determined at each time step. There exists an interval of acceptable α values for a single shape. Finally, by increasing the α value, the convex hull of the cloud of points is obtained.

This construction allows to reproduce exactly linear polynomials over the boundary of any domain. When dealing with piece-wise homogeneous domains, for instance, it is also necessary to ensure the discontinuity of the derivatives of some field (which is itself continuous across the interface). This can also be done by avoiding natural neighbourhood between nodes placed at both sides of the material interface.

2.4 C-NEM versus α -NEM

The application of the C-NEM requires the definition of the constrained Voronoi diagram, that itself requires an explicit description of the domain boundary, which is usually described from a set of nodes defining a polygonal curve. For this reason this technique seems specially well adapted for treating problems in which the geometry or interfaces evolve moderately [16], or the ones involving fixed or moving cracks [15].

On the contrary the α -NEM seems specially well adapted for treating problems in which the domain geometry evolves significantly, as the ones involving large transformation or complex fluid flows [7]. However its application in the context of fracture mechanics deserves particular treatments, due to the occasional high level of detail induced by the crack separation.

Mixed strategies could be imagined: the α -shapes extracting the domain geometry whereas the interfaces treatment is carried out in the C-NEM framework. This marriage allows to profit the most appealing properties of each one of these strategies.

3 Representation of evolving interfaces

3.1 Tracking versus capturing

To represent evolving weak discontinuities the most standard procedures are the ones based on an interface tracking or capturing it by solving the PDE governing its motion. The first strategy is more simple, but its applicability is restricted to problems in which the interface evolves moderately, because for complex evolutions the tracking algorithm becomes too sophisticated to be efficient. In that follows and for the sake of simplicity we restrict our discussion to the 2D case, however, extension to the 3D case is straightforward.

On the other hand, the interface can be captured using different strategies. In the volume-of-fluid method the interface defines the discontinuity of the characteristic function related to one of the regions that the interface defines. The interface can be defined by the interpolated curve or the elements (when a discontinuous piecewise constant approximation is considered) related to a value of 1/2 of the characteristic function. This kind of strategies fail when accurate interface descriptions are desired. Another more accurate description defines the interface as the zero value of a level-set function. This level-set function is advected with a certain velocity that on the interface must coincide with the real interface velocity. During its motion, this level-set function degenerates and must be updated frequently to preserve accuracy in the interface description. The most usual correction consists in transforming the advected level set into a signed distance to the interface. Different efficient algorithms exist today for performing both the advection and the updating of that level-set function. The intersection of the curve related to the zero level

set value (extracted by interpolation) with a background mesh can be used as the nodes defining the interface. Now, two possibilities exist from the point of view of the construction of a functional interpolation:

1. To enrich the interpolation to describe the transmission conditions across the interface using Partition of Unity; enforcing reproduction conditions during the construction of the interpolation functions; or by enriching the interpolation in the elements which are intersected by the interface introducing additional degrees of freedom that can be condensed in the original ones as in the cohesive elements framework. If one is applying this last strategy the size of the problem remains constant because no new degrees of freedom are introduced during the simulation.
2. The intersection points between the zero level-set curve and the background Delaunay triangulation can be considered as new discretization nodes. Now, the interpolation can be defined at both interface sides assuring the functional continuity but its discontinuous normal derivative. If the natural element method is been used, the distortion of the Delaunay triangulation in the neighborhood of the interface does not affect to the interpolation accuracy as proved in [16]. However, in this technique the size of the discrete problem is evolving, because the number of intersection points between the interface and the background Delaunay triangulation changes during the motion of the interface.

If the interface is described by a constant number of points that are advected by the interface velocity (tracking technique), then the size of the discrete problem remains constant, but we can imagine that this possibility can be only envisaged when the interface evolves moderately, because in other cases the addition of new nodes to represent the interface geometry is compulsory.

3.2 Interpolation: enrichment versus explicit interface representation

Interpolation enrichment based on MLS-NEM

There are three main possibilities of enrichment:

1. Partition of unity enrichment. The first one concerns the use of the Partition of Unity paradigm (as usually considered in the framework of the extended finite element – X-FEM – [8]) that generates a linear system whose size evolves as the interface evolves (the nodes of the elements that are intersected by the interface are enriched with new degrees of freedom). As the interpolation generated by the NEM satisfies the partition of unity, all the developments proposed in the context of the finite elements can be extended straightforward to the NEM. This was for example the technique used in [4] for defining mixed interpolations verifying the stability LBB conditions.

2. Element interpolation enrichment. In this case the interpolation in the elements intersected by the interface is enriched by using a function whose derivative becomes discontinuous across the interface and that vanishes at the nodal positions. This new approximation function has associated a new degree of freedom that does not corresponds to a nodal value. By imposing the transmission conditions this new degree of freedom can be written as a function of the nodal values. Thus, this new degrees of freedom are finally condensed on the initial ones, assuring a constant size of the discrete problem which corresponds to the original cloud of nodes.
3. Moving Least Squares enrichment. The consideration of this strategy allows to define an hybrid technique (MLS-NEM) in which different consistencies can be enforced (e.g. the required by material interfaces) without detriment of the appealing NEM properties.

In that follows we focus on the second strategy due to its novelty. Due to the equivalence between the moving least squares and the reproducing kernel particle methods, we are considering by the sake of simplicity the last framework. Let Ω be a 1D domain where the problem is defined (all the results have a direct 2D or 3D counterpart). The points within this domain will be noted by x or s .

The approximation $u^h(x)$ of $u(x)$ is built from the convolution integral

$$u^h(x) = \int_{\Omega} w(x-s, h) u(s) d\Omega \quad (9)$$

where $w(x-s, h)$ is the kernel function and h a parameter defining the size of the approximation support.

The main idea in the enriched RKPM method [14] is to enforce the reproduction of a general function that we can write in the form of a polynomial plus another function noted by $u^e(x)$:

$$u^h(x) = a_0 + a_1x + \dots + a_nx^n + a_{n+1}u^e(x) \quad (10)$$

In the following paragraphs we analyze the required properties of the kernel function $w(x-s, h)$ for reproducing a function expressed by (10).

From Eq. (9), the reproduction of a constant function a_0 is given by

$$\int_{\Omega} w(x-s, h) a_0 d\Omega = a_0 \quad (11)$$

which implies

$$\int_{\Omega} w(x-s, h) d\Omega = 1 \quad (12)$$

which constitutes the partition of unity.

Now, the required condition to reproduce a linear function $u^a(x) = a_0 + a_1x$ is

$$\int_{\Omega} w(x-s, h)(a_0 + a_1 s) d\Omega = a_0 + a_1 x \quad (13)$$

By using the partition of unity (12), Eq. (13) can be rewritten as

$$\begin{cases} \int_{\Omega} w(x-s, h) d\Omega = 1 \\ \int_{\Omega} w(x-s, h) s d\Omega = x \end{cases} \quad (14)$$

which implies the linear consistency of the approximation. Repeating this reasoning results

$$\begin{cases} \int_{\Omega} w(x-s, h) d\Omega = 1 \\ \int_{\Omega} w(x-s, h) s d\Omega = x \\ \vdots \\ \int_{\Omega} w(x-s, h) s^n d\Omega = x^n \\ \int_{\Omega} w(x-s, h) u^e(s) d\Omega = u^e(x) \end{cases} \quad (15)$$

We will note by $u^r(x)$ the approximation function verifying the conditions (15). Usually a cubic spline is considered as kernel function, and consequently the conditions given by Eq. (15) are not satisfied. Liu et al. [6] propose the introduction of a correction function $C(x, x-s)$ for satisfying the reproduction conditions. In our case we consider the more general form $C(x, s, x-s)$ whose pertinence will be discussed later. Thus $u^r(x)$ will be expressed by

$$u^r(x) = \int_{\Omega} C(x, s, x-s) w(x-s, h) u(s) d\Omega \quad (16)$$

where $C(x, s, x-s)$ is assumed to have the following form

$$C(x, s, x-s) = \mathbf{H}^T(x, s, x-s) \mathbf{b}(x) \quad (17)$$

where $\mathbf{H}^T(x, s, x-s)$ represents the vector containing the functions considered in the approximation basis, and $\mathbf{b}(x)$ is a vector containing unknown functions that will be determined for satisfying the reproduction conditions. Thus, Eq. (15) can be rewritten and the vector $\mathbf{b}(x)$ evaluated after the introduction of a quadrature formula.

Thus, the functional approximation can be expressed as (see [14] for additional details)

$$u^r(x) \cong \sum_{i=1}^N \psi_i(x) u_i \quad (18)$$

where ψ_i is the enriched RKP approximation shape function, leading to the so-called enriched reproducing kernel particle approximation (E-RKPA).

To define NN-approximations with discontinuous derivatives we could proceed in the context of the partition of unity (as in the extended finite element technique). However, in this work we propose an enrichment that does not

involve additional degrees of freedom. For this purpose we start introducing the enriched reproducing kernel particle method, that by introducing the NN-interpolation as kernel function leads to NN-interpolation functions with discontinuous derivatives, leading to the so-called enriched natural element interpolation (E-NEM).

To define NN-approximations with discontinuous derivatives we could proceed in the context of the partition of unity (as in the extended finite element technique). However, in this work we propose an enrichment that does not involve additional degrees of freedom. For this purpose we start introducing the enriched reproducing kernel particle method, that by introducing the NN-interpolation as kernel function leads to NN-interpolation functions with discontinuous derivatives.

We consider a level set description $\Theta(\mathbf{x})$ of an interface where the field normal derivatives (with respect to the interface) are discontinuous. Now, we can introduce as enrichment function $u^e(x)$ the following function:

$$u^e(\mathbf{x}) = H_0(\Theta(\mathbf{x}))\Theta(\mathbf{x}) \quad (19)$$

where

$$\Theta(\mathbf{x}) = \begin{cases} \Theta(\mathbf{x}) < 0 & \text{if } \mathbf{x} \in \Omega_1 \\ \Theta(\mathbf{x}) > 0 & \text{if } \mathbf{x} \in \Omega_2 \\ \Theta(\mathbf{x}) = 0 & \text{if } \mathbf{x} \in \Gamma_d \end{cases} \quad (20)$$

and

$$\begin{cases} H_0(\Theta(\mathbf{x})) = 1 & \text{if } \Theta(\mathbf{x}) \geq 0 \\ H_0(\Theta(\mathbf{x})) = 0 & \text{if } \Theta(\mathbf{x}) < 0 \end{cases} \quad (21)$$

Now, we consider a linear consistency enriched with the function given by Eq. (19) and the kernel function $w(\mathbf{x} - \mathbf{x}_i, h) = \phi_i(\mathbf{x})$ (the natural neighbor shape functions). The resulting approximation shape functions have the linear consistency but allows also to reproduce discontinuous normal derivatives across the interface Γ_d . As the use of the NEM kernel function restricts the number of neighbor nodes to the natural ones, in order to impose higher order consistency new degrees of freedom must be associated to the existing nodes, in a formulation that we called Hermite-NEM, or in other additional nodes, strategy that we called bubble-NEM [18].

To illustrate the capabilities of the proposed technique we consider the exact solution of the Laplace's problem (modelling the temperature distribution in a steady heat transfer problem) defined in a bi-material consisting of two cylinders with different thermal conductivities. The reproduction tests have been carried out using the E-RKPM as well as the E-NEM, where the circular interface was modelled from the distance to that interface that multiplies the Heaviside's function related to that distance. Fig. 3 illustrate a detail of the reconstructed temperature field where we can notice an accurate description of the interface. The discontinuity in the field derivatives is accurately

accounted, as suggested by the representation of the x -derivative depicted in figure 3.

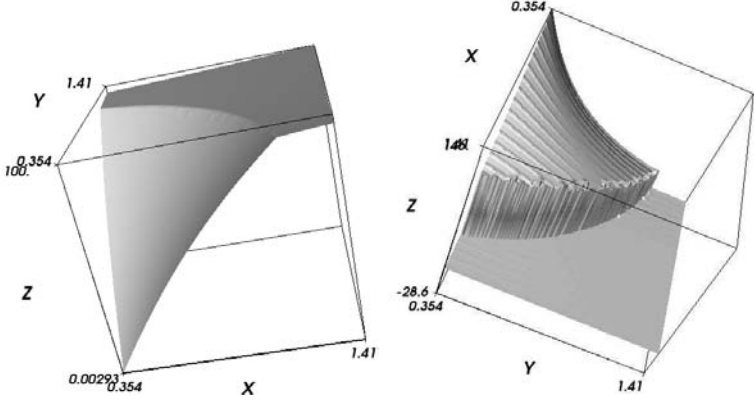


Fig. 3. (left) Enriched Natural Neighbor approximation with discontinuous normal derivatives across a circular interface; (right) x -derivative of the temperature field.

Finally, in order to quantify the accuracy of the results we compare in figure 4 the error (using the two usual norms) using the E-RKPM and the E-NEM techniques. In figure 4(right) we can notice that the E-NEM error is not affected by the slope change across the interface, that increases with the difference of thermal conductivities (for $k_1 = 10$ the ratio of conductivities is 10 whereas it is of 100 for $k_1 = 100$).

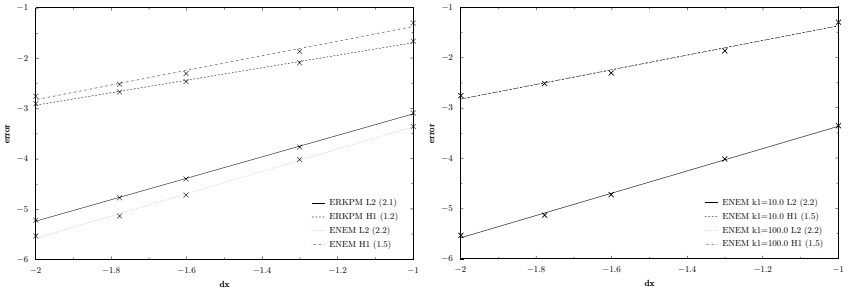


Fig. 4. (left) Approximation errors using the E-RKPM and the E-NEM; (right) E-NEM approximation error for different conductivities ratios.

Explicit interface representation

The ability of the C-NEM for treating problems involving cracks has been illustrated in [15] and for moving interfaces in thermal problems in [16]. In the

present paper, the domain is partitioned in some regions with different material properties. Each subdomain is discretized using a cloud of nodes and the interfaces between the different regions are described by a polygonal curve defined by a set of nodes. Then, a constrained Voronoi diagram is defined at each subdomain with respect to the domain boundary and the interfaces. The attractive feature of the present technique is the possibility to move the interfaces without special care for the shape of the underlying Delaunay triangles because the interpolation accuracy does not depend on the geometrical quality of the Delaunay triangles, in contrast to the FEM. In this manner, the continuity of the approximation is guaranteed by the strict linearity of the interpolation across the interfaces, that are defined by a set of interface nodes.

To illustrate this behavior, we consider the situation depicted in Fig. 5, where the point \mathbf{x} moves from Ω_1 to Ω_2 . If \mathbf{x} is in Ω_1 , the interpolated field is constructed using the visible neighbor nodes from point \mathbf{x} , all of them inside Ω_1 (Γ_I is assumed opaque). If \mathbf{x} is on Γ_I , according to the previous discussion, the interpolated field is strictly linear because it only depends on the two neighbor nodes located on Γ_I . Finally, when \mathbf{x} is in Ω_2 , the interpolated field is defined using the visible neighbor nodes from point \mathbf{x} all of them inside Ω_2 (Γ_I being opaque). The continuity of the interpolated field is then guaranteed, but a discontinuity appears in the normal derivative across the interface, because of a sudden change in the neighbor nodes across the interface.

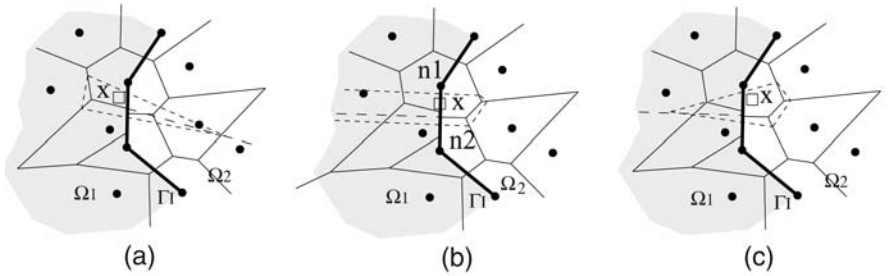


Fig. 5. Reproducing discontinuous derivatives using the constrained Voronoi diagram.

4 Model reduction

4.1 Fundamentals: Karhunen-Loève decomposition and reduced basis enrichment

We assume that the evolution of a certain field $T(\mathbf{x}, t)$ is known. In practical applications, this field is expressed in a discrete form which is known at

the nodes of a spatial mesh and for some times t^m . Thus, we consider that $T(\mathbf{x}_i, t^m) = T^m(\mathbf{x}_i) \equiv T_i^m$ ($t^m = m \times \Delta t$). We can also write \underline{T}^m for the vector containing the nodal degrees of freedom at time t^m . The main idea of the Karhunen-Loève (KL) decomposition is to obtain the most typical or characteristic structure $\phi(\mathbf{x})$ among these $T^m(\mathbf{x})$, $\forall m$. This is equivalent to obtain a function that maximizes α :

$$\alpha = \frac{\sum_{m=1}^{m=M} \left[\sum_{i=1}^{i=N} \phi(\mathbf{x}_i) T^m(\mathbf{x}_i) \right]^2}{\sum_{i=1}^{i=N} (\phi(\mathbf{x}_i))^2} \quad (22)$$

The maximization leads to:

$$\sum_{m=1}^{m=M} \left[\left(\sum_{i=1}^{i=N} \tilde{\phi}(\mathbf{x}_i) T^m(\mathbf{x}_i) \right) \left(\sum_{j=1}^{j=N} \phi(\mathbf{x}_j) T^m(\mathbf{x}_j) \right) \right] = \alpha \sum_{i=1}^{i=N} \tilde{\phi}(\mathbf{x}_i) \phi(\mathbf{x}_i); \quad \forall \tilde{\phi} \quad (23)$$

which can be rewritten in the form

$$\sum_{i=1}^{i=N} \left\{ \sum_{j=1}^{j=N} \left[\sum_{m=1}^{m=M} T^m(\mathbf{x}_i) T^m(\mathbf{x}_j) \phi(\mathbf{x}_j) \right] \tilde{\phi}(\mathbf{x}_i) \right\} = \alpha \sum_{i=1}^{i=N} \tilde{\phi}(\mathbf{x}_i) \phi(\mathbf{x}_i); \quad \forall \tilde{\phi} \quad (24)$$

Defining the vector $\underline{\phi}$ such that its i -component is $\phi(\mathbf{x}_i)$, Eq. (24) takes the following matrix form

$$\underline{\tilde{\phi}}^T \underline{c} \underline{\phi} = \alpha \underline{\tilde{\phi}}^T \underline{\phi}; \quad \forall \underline{\tilde{\phi}} \Rightarrow \underline{c} \underline{\phi} = \alpha \underline{\phi} \quad (25)$$

where the two-point correlation matrix is given by

$$c_{ij} = \sum_{m=1}^{m=M} T^m(\mathbf{x}_i) T^m(\mathbf{x}_j) \Leftrightarrow \underline{c} = \sum_{m=1}^{m=M} \underline{T}^m (\underline{T}^m)^T \quad (26)$$

which is symmetric and positive definite. If we define the matrix $\underline{\underline{Q}}$ containing the discrete field history:

$$\underline{\underline{Q}} = \begin{pmatrix} T_1^1 & T_1^2 & \cdots & T_1^M \\ T_2^1 & T_2^2 & \cdots & T_2^M \\ \vdots & \vdots & \ddots & \vdots \\ T_N^1 & T_N^2 & \cdots & T_N^M \end{pmatrix} \quad (27)$$

then it is easy to verify that the matrix \underline{c} in Eq. (25) results

$$\underline{c} = \underline{\underline{Q}} \underline{\underline{Q}}^T \quad (28)$$

A posteriori reduced modelling

If some direct simulations have been carried out, we can determine T_i^m , $\forall i \in [1, \dots, N]$ and $\forall m \in [1, \dots, M]$, and from these solutions the n eigenvectors related to the n -highest eigenvalues that are expected to contain the most information about the problem solution. For this purpose we solve the eigenvalue problem defined by Eq. (25) retaining all the eigenvalues $\underline{\phi}_k$ belonging to the interval defined by the highest eigenvalue and that value divided by a large enough value (10^8 in our simulations). In practice n is much lower than N . Thus, we can try to use these n eigenfunctions $\underline{\phi}_k$ for approximating the solution of a problem slightly different to the one that has served to define T_i^m . For this purpose we need to define the matrix $\underline{\underline{B}} = [\underline{\phi}_1 \dots \underline{\phi}_n]$

$$\underline{\underline{B}} = \begin{pmatrix} \phi_1(\mathbf{x}_1) & \phi_2(\mathbf{x}_1) & \dots & \phi_n(\mathbf{x}_1) \\ \phi_1(\mathbf{x}_2) & \phi_2(\mathbf{x}_2) & \dots & \phi_n(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_1(\mathbf{x}_N) & \phi_2(\mathbf{x}_N) & \dots & \phi_n(\mathbf{x}_N) \end{pmatrix} \quad (29)$$

Now, if we consider the linear system of equations coming from the discretization of a generic problem, in the form:

$$\underline{\underline{G}} \underline{\underline{T}}^m = \underline{\underline{H}}^{m-1} \quad (30)$$

where the superscript refers to the time step, then, assuming that the unknown vector contains the nodal degrees of freedom, it can be expressed as:

$$\underline{\underline{T}}^m = \sum_{i=1}^{i=n} \zeta_i^m \underline{\phi}_i = \underline{\underline{B}} \underline{\underline{\zeta}}^m \quad (31)$$

from which Eq. (30) results

$$\underline{\underline{G}} \underline{\underline{T}}^m = \underline{\underline{H}}^{m-1} \Rightarrow \underline{\underline{G}} \underline{\underline{B}} \underline{\underline{\zeta}}^m = \underline{\underline{H}}^{m-1} \quad (32)$$

and by multiplying both terms by $\underline{\underline{B}}^T$ we obtain

$$\underline{\underline{B}}^T \underline{\underline{G}} \underline{\underline{B}} \underline{\underline{\zeta}}^m = \underline{\underline{B}}^T \underline{\underline{H}}^{m-1} \quad (33)$$

which proves that the final system of equations is of low order, i.e. the dimension of $\underline{\underline{B}}^T \underline{\underline{G}} \underline{\underline{B}}$ is $n \times n$, with $n \ll N$.

Enriching the approximation basis

Obviously, accurate simulations require an error evaluation as well as the possibility of adapting the approximation basis by introducing new functions able to describe the solution features. Ryckelynck proposed in [9] an adaptive procedure, able to construct or enrich the reduced approximation basis. For

this purpose, he proposed to add to the reduced approximation basis some Krylov subspaces generated by the equation residual. Despite the fact that this enrichment tends to increase the number of approximation function, when it is combined with a KL decomposition that continuously reduces this number, the size of the problems is quickly optimized and some times stabilized. This technique is summarized in the present section.

We assume at present that the evolution problem has been solved in the entire time interval using the reduced basis defined by matrix $\underline{\underline{B}}$ (in reference to Eq. (31)) solving Eq. (33):

$$\underline{\zeta}^m = (\underline{\underline{B}}^T \underline{\underline{G}} \underline{\underline{B}})^{-1} \underline{\underline{B}}^T \underline{H}^{m-1} \quad (34)$$

We assume that $t_{max} = M \times \Delta t$ and consequently the residual at t_{max} , \underline{R}^M , can be computed from

$$\underline{R}^M = \underline{\underline{G}} \underline{\underline{B}} \underline{\zeta}^M - \underline{H}^{M-1} \quad (35)$$

If the norm of the residual is small enough $\|\underline{R}^M\| < \epsilon$ (being ϵ a small enough parameter) the computed solution can be assumed as good, but on the contrary, if $\|\underline{R}^M\| \geq \epsilon$, the solution must be improved. For this purpose, we propose to enrich the reduced approximation basis by introducing the just computed residual:

$$\underline{\underline{B}} \leftarrow [\underline{\underline{B}} \quad \underline{R}^M] \quad (36)$$

and now, the evolution is recomputed in the entire whole interval using Eq. (34) with the just updated reduced basis $\underline{\underline{B}}$. Both steps, enrichment and the evolution updating, must be repeated until verifying the stop condition $\|\underline{R}^M\| < \epsilon$. After reaching this threshold value, the final reduced approximation basis could be constructed by applying the Karhunen-Loève decomposition to the last time evolution of the solution.

4.2 Accounting for weak discontinuities

When one considers the application of model reduction techniques in problems involving weak discontinuities, two questions arise suddenly: (i) Can the transient solution of problems involving weak discontinuities be expressed as a linear combination of a reduced number of modes?; and (ii) Can the approximation basis be enriched using the residual? The first question determines the reducibility of the problem and the second one the possibility to perform this reduction in a priori approach.

In concluding these two question we consider a one dimensional heat transfer problem defined by:

$$\frac{\partial T_i(x, t)}{\partial t} = k_i \frac{\partial^2 T_i(x, t)}{\partial x^2} + f(x), \quad x \in \Omega_i \quad (37)$$

where $\Omega_1 =]-1, 0[$ and $\Omega_2 =]0, 1[$, $f(x) = 1$, $k = 1$ and $k = 10$ (all the units in the metric system), being the initial, boundary and transmission conditions given by:

$$\begin{cases} T_1(x = -1, t) = 0 \\ T_1(x, t = 0) = 0 \\ T_2(x = 1, t) = 0 \\ T_2(x, t = 0) = 0 \\ T_1(x = 0, t) = T_2(x = 0, t) \\ -k_1 \frac{\partial T_1}{\partial x} \Big|_{x=0,t} = -k_2 \frac{\partial T_2}{\partial x} \Big|_{x=0,t} \end{cases} \quad (38)$$

Figure 6 depicts the temperature evolution for $t < 1s$ as well as the functions that results from the Karhunen-Loève decomposition and that are associated to eigenvalues grater than 10^{-8} times the highest one. The reducibility of the model is proved by the existence of only 4 functions from which the entire solution evolution can be expressed. It is easy to prove that the solution of the transient problem expressed in this reduced basis is in perfect agreement with the finite element solution in the entire time interval.

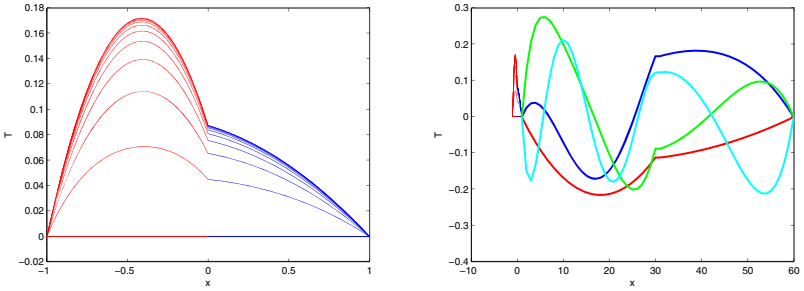


Fig. 6. (left) Time evolution of the temperature profile; (right) functions defining the reduced approximation basis.

To address the second question we consider the solution of the transient problem starting from a reduced basis that only contains the function depicted in figure 7. This reduced basis is enriched by adding the residual computed at time $t = 1$, from which the entire evolution is recomputed. Even if the enrichment increases the size of the discrete problem, the Karhunen-Loève decomposition performed when the convergence is reached allows to reduced the size of the approximation basis. Thus, the size of the discrete problem remains stabilized through the entire simulation.

Figure 7 depicts the function that constitutes the first reduced approximation basis as well as the residual associated to the solution at $t = 1$ computed in that reduced basis (broken line). The solution obtained at $t = 1$ when the enrichment algorithm converges is also depicted in this figure, and

we can notice that it corresponds to the one computed using the global finite element basis.

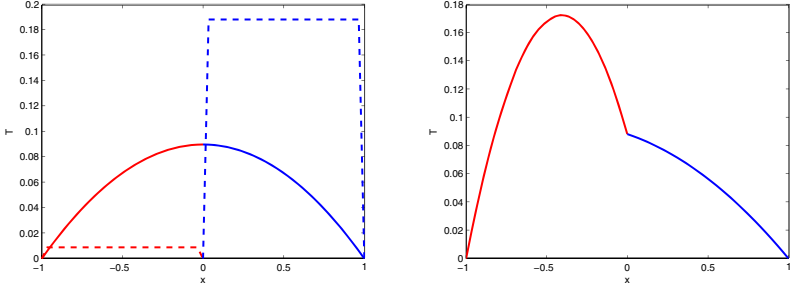


Fig. 7. (left) First approximation function and its associated residual at $t = 1$; (right) Computed solution at $t = 1$ after convergence of the enrichment algorithm.

5 Conclusions

In this paper we have explored some alternatives for treating fixed or evolving weak discontinuities in the context of the meshless natural element method. We have illustrated that standard and new strategies can be applied without detriment of the main appealing properties of this meshless discretization technique.

We have also presented some preliminary results concerning the reduction of such models. In the case of fixed interfaces the reduction procedure works, opening new perspectives in the reduction of models involving evolving discontinuities. This topic that constitutes a work in progress, combines the element enrichment (followed by a static condensation of the new introduced degrees of freedom) and the standard model reduction previously described.

References

1. Belytschko T, Krongauz Y, Organ D, Fleming M, Krysl P(1996) Comp. Meth. in Applied Mechanics and Eng. 139:3–47
2. Cueto E, Doblare M, Gracia L (2000) International Journal for Numerical Methods in Engineering 49:519–546
3. Cueto E, Sukumar N, Calvo B, Cegonino J, Doblare M (2003) Archives of Computational Methods in Engineering: State of the art reviews 10/4:307–384
4. Gonzalez D, Cueto E, Doblare M (2004) International Journal for Numerical Methods in Engineering 61:611–632
5. Hiyoshi H, Sugihara K (2002) Computational Geometry 22:167183.

6. Liu WK, Jun S, Zhang YF (1995) *Int. J. Numer. Methods Fluids* 21: 1081–1106
7. Martinez MA, Cueto E, Alfaro I, Doblare M, Chinesta F (2004) *International Journal for Numerical Methods in Engineering*, 60/13: 2105–2129
8. Moes N, Dolbow J, Belytschko T (1999) *International Journal for Numerical Methods in Engineering* 46: 131–150
9. Ryckelynck D (2005) *Journal of Computational Physics* 202: 346–366
10. Shewchuck JR (1998) Tetrahedral mesh generation by delaunay refinement. In: *Proceedings of the fourteenth annual symposium on computational geometry*, Minneapolis, Minnesota. Association for Computing Machinery
11. Shewchuck JR (2000) Sweep algorithms for constructing higher-dimensional constrained Delaunay triangulations. In: *Proceedings of the sixteenth annual symposium on computational geometry*. Association for Computing Machinery.
12. Sukumar N, Moran B, Belytschko T (1998) *International Journal for Numerical Methods in Engineering* 43:839–887
13. Sibson R (1980) *Math. Proc. Camb. Phil. Soc.* 87:151–155
14. Trunzler J, Joyot P, Chinesta F (2005) Discontinuous derivative enrichment in RKPM. In: *Lectures Notes on Computational Science and Engineering*. Springer 43: 93–108
15. Yvonnet J, Ryckelynck D, Lorong P, Chinesta F (2004) *International Journal for Numerical Methods in Engineering* 60:1451–1474
16. Yvonnet J, Ryckelynck D, Lorong Ph, Chinesta F (2005) *International Journal of Thermal Sciences* 44: 559–569
17. Yvonnet J, Ryckelynck D, Coffignal G, Lorong Ph, Chinesta F (2006) *Computers & Structures* 84/21: 1301–1312
18. Yvonnet J, Villon P, Chinesta F (2006) *International Journal for Numerical Methods in Engineering* 66/7: 1125–1152