ECCOMAS Congress 2016
VII European Congress on Computational Methods in Applied Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou, V. Plevris (eds.)
Crete Island, Greece, 5–10 June 2016

EFFICIENT SHAPE FUNCTION COSTRUCTION OF EFG MESHLESS METHOD

P. Metsis[1], **and M. Papadrakakis**[1]

[1] *Institute of Structural Analysis and Antiseismic Research*
National Technical University of Athens
Zografou Campus, Athens 15780
(pmetsis@gmail.com, mpapadra@central.ntua.gr)

Keywords: Meshless methods; Element free Galerkin (EFG); shape function construction; Adaptivity

Abstract. *Meshless methods have shown increased accuracy and better convergence rates compared to other well-known simulation methods in a variety of computational mechanics problems. In adaptivity analysis, besides the effortless node addition at each step, their computational cost is relatively higher, since the construction of updated shape functions and shape functions derivatives that are used to construct the stiffness matrix is burdensome.*

EFG and other similar methods use the moving least squares procedure, where the approximation of the displacement field is expressed as a polynomial with non-constant coefficients. In order to determine these coefficients, a weighted residual is solved for every point of interest, usually the Gauss points. Each time the domain of influence changes or a new node is added or subtracted from the domain, the shape functions and their derivatives have to be re-calculated for the influenced areas, requiring a significant amount of computational effort.

In this work, the shape function construction procedure is analyzed in depth and several new approaches are proposed ranging from an explicit analytical form for typical arrangement of nodes -which corresponds to a significant percentage of typical problem domains- to complex hierarchical formulations in the context of an h-type refinement scheme. The addition of new nodes and subsequently the re-calculation of the influenced moment matrices, that are necessary for obtaining the shape functions and their derivatives and subsequently for the construction of the stiffness matrix, are properly addressed. Unconventional types of weight and shape functions are also proposed and the results are critically assessed.

1 INTRODUCTION

Meshless methods (MMs) have been used for solving problems that are difficult or impossible to be dealt with by mesh related procedures of established simulation methods like the finite element method (FEM). Problems of large deformations, discontinuities, contact and moving boundaries or problems demanding very high accuracy, are some of the problems MMs are mostly used. In MMs there is no need to construct a mesh while the solution and its derivatives are continuous throughout the domain of interest leading to more accurate stress distributions as well as to less sensitive to distortions due to large deformations. In areas where increased accuracy is needed, refinement by adding nodes (h-adaptivity) may implemented effortlessly. Moreover, established methods like FEM may be partially enriched with features of some MMs, or be coupled with, enhancing their effectiveness.

One of the first and most prominent meshless methods is the element free Galerkin (EFG) method introduced by Belytschko et al.. EFG requires only nodal data and no element connectivity is needed to construct the shape functions. However, a global background cell structure is necessary for the numerical integration. Moreover, since the number of interactions between nodes and/or integration points is heavily increased, due to large domains of influence, the resulting matrices are more densely populated and the computational cost for the formulation and solution of the problem is much higher than in the conventional FEM.

The calculation of the shape functions and shape functions derivatives that are used to construct the stiffness matrix add significantly to the abovementioned computational cost. EFG uses the moving least squares procedure, where the approximation of the displacement field is expressed as a polynomial with non-constant coefficients. In order to determine these coefficients, a weighted residual is solved for every point of interest, usually the Gauss points. Each time the domain of influence changes or a new node is added or subtracted from the domain, the shape functions and their derivatives have to be re-calculated for the influenced areas. Thus, in the context of an adaptive scheme this procedure can be burdensome.

2 BASIC INGREDIENTS OF THE MESHLESS EFG METHOD

The approximation of a scalar function u in the meshless EFG method can be written as

$$u(\mathrm{x},t)=\sum_{i\in S}\Phi_i(\mathrm{x})u_i(t) \tag{1}$$

where Φ_i are the shape functions, u_i are the values of the scalar function u at node i located at position x_i, and S is the set of nodes i for which $\Phi_i(\mathrm{x})$ obtains non zero values. The shape functions in eq.(1) are only approximants and not interpolants, since generally $u_i \neq u(\mathrm{x}_i)$.

The shape functions Φ_i are obtained from the weight coefficients w_i, which are functions of a distance parameter $r=\|\mathrm{x}_i-\mathrm{x}\|/d_i$ where d_i defines the domain of influence (DoI) of each node i. The size primarily, but also the shape of the domain of influence, is crucial to solution accuracy, stability and computational cost, as it co-defines the degree of continuity between the nodes and the bandwidth of the resulting system matrices.

If $\mathrm{p}(\mathrm{x})$ is a complete polynomial of length m and $\mathrm{a}(\bar{\mathrm{x}})$ contains non-constant coefficients that depend on x:

$$\mathrm{a}(\bar{\mathrm{x}}) = [a_0(\mathrm{x}) \quad a_1(\mathrm{x}) \quad a_2(\mathrm{x}) \quad \dots \quad a_m(\mathrm{x})]^T \tag{2}$$

then the approximation u^h is expressed as a polynomial of length m with non-constant coefficients. The local approximation around a point $\bar{\mathrm{x}}$, evaluated at a point x is given by

$$u_L^h(\mathrm{x}, \bar{\mathrm{x}}) = \mathrm{p}^T(\mathrm{x})\mathrm{a}(\bar{\mathrm{x}}) \tag{3}$$

In two dimensional problems, the linear basis $\mathrm{p}(\mathrm{x})$ is given by

$$\mathrm{p}^T(\mathrm{x}) = [1 \quad x \quad y], \qquad m = 3 \tag{4}$$

and the quadratic basis by

$$\mathrm{p}^T(\mathrm{x}) = [1 \quad x \quad y \quad x^2 \quad y^2 \quad xy], \qquad m = 6 \tag{5}$$

The minimization of a functional $J(\mathrm{x})$ defined by a weighted average over all nodes $i \in 1, \dots, n$:

$$J(\mathrm{x}) = \sum_{i=1}^{n} w(\mathrm{x} - \mathrm{x}_i)\left[u_L^h(\mathrm{x}_i, \mathrm{x}) - u_i\right]^2 = \sum_{i=1}^{n} w(\mathrm{x} - \mathrm{x}_i)\,[\mathrm{p}^T(\mathrm{x}_i)\mathrm{a}(\mathrm{x}) - u_i]^2 \tag{6}$$

determines the parameters $a_j(\mathrm{x})$ at any point x. In eq.(6) the parameters u_i are specified by the difference between the local approximation $u_L^h(\mathrm{x}, \bar{\mathrm{x}})$ and the nodal value u_i, while the weight function satisfies the condition $w(x - x_i) \neq 0$.

An extremum of $J(\mathrm{x})$ with respect to the coefficients $a_j(\mathrm{x})$ can be obtained by setting the derivative of J with respect to $\mathrm{a}(\mathrm{x})$ equal to zero. This condition gives the following relation

$$\mathrm{A}(\mathrm{x})\mathrm{a}(\mathrm{x}) = \mathrm{W}(\mathrm{x})\mathrm{u} \tag{7}$$

where the moment matrix $\mathrm{A}(\mathrm{x})$ is defined as

$$\mathrm{A}(\mathrm{x}) = \sum_{i=1}^{n} w(\mathrm{x} - \mathrm{x}_i)\mathrm{p}(\mathrm{x}_i)\,\mathrm{p}^T(\mathrm{x}_i) \tag{8}$$

and

$$\mathrm{W}(\mathrm{x}) = [w(\mathrm{x} - \mathrm{x}_1)\mathrm{p}(\mathrm{x}_1) \quad w(\mathrm{x} - \mathrm{x}_2)\mathrm{p}(\mathrm{x}_2) \quad \dots \quad w(\mathrm{x} - \mathrm{x}_n)\mathrm{p}(\mathrm{x}_n)] \tag{9}$$

The approximants u^h can be defined by solving for $\mathrm{a}(\mathrm{x})$ in eq.(7) and substituting into eq.(3):

$$u^h(\mathrm{x}) = \mathrm{p}^T(\mathrm{x})[[\mathrm{A}(\mathrm{x})]^{-1}\mathrm{W}(\mathrm{x})\mathrm{u}] \tag{10}$$

which together with eq.(1) leads to the derivation of the shape function Φ_i associated with node i at point x:

$$\Phi_i(\mathrm{x}) = \mathrm{p}^T(\mathrm{x})[\mathrm{A}(\mathrm{x})]^{-1}\mathrm{W}(\mathrm{x}_i) \tag{11}$$

A solution of a local problem $\mathrm{A}(\mathrm{x})\mathrm{z} = \mathrm{p}(\mathrm{x})$ of size $m \times m$ is performed whenever the shape functions are to be evaluated. This constitutes a drawback of moving least squares (MLS)-based MMs since the computational cost can be substantial, while it is possible for the moment matrix $\mathrm{A}(\mathrm{x})$ to be ill conditioned. In adaptive procedures, where nodes are gradually added and the shape functions of enriched areas are recalculated at each adaptive step, the required computational effort can be considerably increased.

The Galerkin weak form of the above formulation gives the discrete algebraic equation

$$\mathrm{Ku} = \mathrm{f} \tag{12}$$

with stiffness matrix components K_{ij}

$$\mathrm{K}_{ij} = \int_{\Omega} \mathrm{B}_i^T \mathrm{E}\mathrm{B}_j d\Omega \tag{13}$$

and force vector f_i

$$\mathrm{f}_i = \int_{\Gamma_\mathrm{t}} \Phi_i \bar{\mathrm{t}} d\Gamma + \int_{\Omega} \Phi_i \mathrm{b} d\Omega \tag{14}$$

In 2D problems, matrix B is given by

$$\mathrm{B}_i = \begin{bmatrix} \Phi_{i,x} & 0 \\ 0 & \Phi_{i,y} \\ \Phi_{i,y} & \Phi_{i,x} \end{bmatrix} \tag{15}$$

and subsequently in 3D problems by

$$\mathrm{B}_i = \begin{bmatrix} \Phi_{j,x} & 0 & 0 \\ 0 & \Phi_{j,y} & 0 \\ 0 & 0 & \Phi_{j,z} \\ \Phi_{j,y} & \Phi_{j,x} & 0 \\ 0 & \Phi_{j,z} & \Phi_{j,y} \\ \Phi_{j,z} & 0 & \Phi_{j,x} \end{bmatrix} \tag{16}$$

Due to the lack of the Kronecker delta property of shape functions, the essential boundary conditions cannot be imposed the same way as in FEM. Several techniques are available such as Lagrange multipliers, penalty method and EFG - FEM coupling.

For the integration of eq.(13), virtual background cells are considered by dividing the problem domain into integration cells over which a Gaussian quadrature is performed:

$$\int_{\Omega} \mathrm{f}(\mathrm{x}) d\Omega = \sum_{J} \mathrm{f}(\xi_J) \omega_{\Xi} det J^{\xi}(\xi) \tag{17}$$

where ξ are the local coordinates and $detJ^{\xi}(\xi)$ is the determinant of the Jacobian.

3 FORMULATION OF THE SHAPE FUNCTIONS BY PARTS

The shape functions of eq.(11) are calculated after computing the moment matrix A(x) from eq.(8). Suppose that $A(x)^{initial}$ is the moment matrix of the initial node distribution of a particular domain of influence (DoI) and some new nodes are added to that DoI. The enriched moment matrix $A(x)^{enriched}$ is given by:

$$A(x)^{enriched} = \sum_{k=1}^{n\_enriched} w(\mathrm{x}-\mathrm{x}_k)\mathrm{p}(\mathrm{x}_k)\,\mathrm{p}^T(\mathrm{x}_k) \tag{18}$$

where the weight function $w(\mathrm{x}-\mathrm{x}_k)$ and the polynomial $\mathrm{p}(\mathrm{x}_k)$ depend only on the position of the node under consideration. Thus for cubic spline weight function:

$$w(\mathrm{x}-\mathrm{x}_i) = \begin{cases} \frac{2}{3} - 4(x-x_i)^2 + (x-x_i)^3, for\ (x-x_i) \le \frac{1}{2} \\ \frac{4}{3} - 4(x-x_i) + 4(x-x_i)^2 - \frac{4}{3}(x-x_i)^3, for\ \frac{1}{2} < (x-x_i) \le 1 \\ 0, for\ (x-x_i) > 1 \end{cases} \tag{19}$$

$A(x)^{enriched}$ consists of two parts that can be computed independently to each other.

$$A(x)^{enriched} = \sum_{i=1}^{n\_initial} w(\mathrm{x}-\mathrm{x}_i)\mathrm{p}(\mathrm{x}_i)\,\mathrm{p}^T(\mathrm{x}_i) \quad + \sum_{j=1}^{n\_additional} w(\mathrm{x}-\mathrm{x}_j)\mathrm{p}(\mathrm{x}_j)\,\mathrm{p}^T(\mathrm{x}_j) \tag{20}$$

$$A(x)^{enriched} = A(x)^{\text{initial}} + A(x)^{additional}$$

where $n\_enriched = n\_initial + n\_additional$.

The enriched shape function Φ_i associated with node i at point x is calculated from eq.(11) as follows:

$$\begin{aligned} \Phi_i(\mathrm{x})^{enriched} &= \mathrm{p}^T(\mathrm{x})\left[A(x)^{enriched}\right]^{-1}\mathrm{W}(\mathrm{x}_i) \\ &= \mathrm{p}^T(\mathrm{x})\left[A(x)^{\text{initial}} + A(x)^{additional}\right]^{-1}\mathrm{W}(\mathrm{x}_i) \end{aligned} \tag{21}$$

where the issue of the inverse of the sum of two matrices arises. Following the Binomial inverse theorem:

If a and $a+b$ are invertible, then

$$(a+b)^{-1} = a^{-1} + X \tag{22}$$

with

$$X = -(I + a^{-1}b)^{-1}\,a^{-1}\,b\,a^{-1}$$

the inverse matrix of eq.(21) can be easily calculated since $(A(x)^{\text{initial}})^{-1}$ is already computed from the initial step. Thus,

$$\begin{aligned}\Phi_i(\mathrm{x})^{enriched} &= \mathrm{p}^T(\mathrm{x})\left[A(x)^{\text{initial}} + A(x)^{additional}\right]^{-1}\mathrm{W}(\mathrm{x}_i) \\ &= \mathrm{p}^T(\mathrm{x})\left[A(x)^{\text{initial}}\right]^{-1}\mathrm{W}(\mathrm{x}_i) \;+\; \mathrm{p}^T(\mathrm{x})\,[\mathrm{X}]\,\mathrm{W}(\mathrm{x}_i)\end{aligned} \tag{23}$$

where

$$X = -\left(I + \{A(x)^{\text{initial}}\}^{-1}A(x)^{additional}\right)^{-1}\{A(x)^{\text{initial}}\}^{-1}A(x)^{additional}\{A(x)^{\text{initial}}\}^{-1} \tag{24}$$

Then, since

$$\Phi_i(\mathrm{x})^{initial} = p^T(x)\left[A(x)^{initial}\right]^{-1}W(x_i) \tag{25}$$

and

$$\Phi_i(\mathrm{x})^{additional} = \mathrm{p}^T(\mathrm{x})\,[\mathbf{X}]\,\mathrm{W}(\mathrm{x}_i) \tag{26}$$

the enriched shape function is given by:

$$\Phi_i(\mathrm{x})^{enriched} = \Phi_i(\mathrm{x})^{initial} + \Phi_i(\mathrm{x})^{additional} \tag{27}$$

The above derivation expresses the enriched shape functions as the sum of the initial shape functions and the contribution of the newly added nodes.

The proposed formulation is demonstrated in the following one-dimensional problem. Consider an interval $0 \le x \le 4$ divided into four unequal parts by five nodes, as shown in Fig. 1.

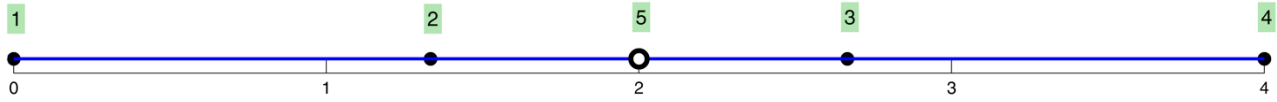


Fig. 1: Node distribution 1D example. ● initial nodes, ○ added node

The typical expression of the shape functions for each node is given by eq.(11) which is evaluated at each Gauss point. The linear basic function $\mathbf{p}^T(x) = \begin{bmatrix}1 & x\end{bmatrix}$ is used. For a certain Gauss point, eq.(11) can be written as:

$$\underbrace{\boldsymbol{\Phi}(\mathrm{x}_{Gp})}_{1\mathrm{xm}} = \underbrace{p^T(x_{Gp})}_{1\mathrm{x2}}\underbrace{\left[A_{1:5}(\mathrm{x}_{Gp})\right]^{-1}}_{2\mathrm{x2}}\underbrace{W(\mathrm{x}_{Gp})}_{2\mathrm{xm}} \tag{28}$$

where

$$\mathrm{p}^T(\mathrm{x}_{Gp}) = [1 \quad \mathrm{x}_{Gp}] \tag{29}$$

Eq.(8) for the initial nodes m and the additional node n is written:

$$\boldsymbol{A}_{1:(m+n)}(\mathrm{x}_{Gp}) = \sum_{i=1}^{(m+n)} w(\mathrm{x}_{Gp} - \mathrm{x}_i)\mathrm{p}(\mathrm{x}_i)\,\mathrm{p}^T(\mathrm{x}_i) = \sum_{i=1}^{(m+n)} w(\mathrm{x}_{Gp} - \mathrm{x}_i)\begin{bmatrix}1\\ \mathrm{x}_i\end{bmatrix}[1 \quad \mathrm{x}_i] \tag{30}$$

while eq.(23) becomes:

$$\begin{aligned}\Phi_{1:(m+n)}(\mathrm{x}_{Gp})^{enriched} &= \Phi_{1:m}(\mathrm{x}_{Gp}) + \Phi_n(\mathrm{x}_{Gp}) = \\ &= p^T(\mathrm{x}_{Gp})\left[A_{1:(m+n)}(\mathrm{x}_{Gp})\right]^{-1} W_{1:m}(\mathrm{x}) + \mathrm{p}^T(\mathrm{x}_{Gp})\left[A_{1:(m+n)}(\mathrm{x}_{Gp})\right]^{-1} W_n(\mathrm{x})\end{aligned} \tag{31}$$

For the example of Fig. 1, the first term includes the value of the shape functions for the first four nodes (m=4), at a certain Gauss point, and the second term the value of the shape function for the fifth node (n=1).

The weighted moment matrix **A** can also be separated into two parts as follows:

$$\begin{aligned}A_{1:(m+n)}(\mathrm{x}_{Gp}) &= A_{1:m}(\mathrm{x}_{Gp}) + A_n(\mathrm{x}_{Gp}) = \\ &= \sum_{i=1}^{m} w(x_{Gp} - x_i)p(x_i)\,p^T(x_i) + w(x_{Gp} - x_n)p(x_n)p^T(x_n)\end{aligned} \tag{32}$$

Using the *Binomial inverse theorem,* we obtain

$$\left[A_{1:(m+n)}(\mathrm{x}_{Gp})\right]^{-1} = \left[A_{1:m}(\mathrm{x}_{Gp}) + A_n(\mathrm{x}_{Gp})\right]^{-1} = \left[A_{1:m}(\mathrm{x}_{Gp})\right]^{-1} + X \tag{33}$$

where $$X = -\left\{I + \left[A_{1:m}(\mathrm{x}_{Gp})\right]^{-1} A_n(\mathrm{x}_{Gp})\right\}^{-1}\left[A_{1:m}(\mathrm{x}_{Gp})\right]^{-1} A_n(\mathrm{x}_{Gp})\left[A_{1:m}(\mathrm{x}_{Gp})\right]^{-1} \tag{34}$$

Since $\left[A_{1:m}(\mathrm{x}_{Gp})\right]^{-1}$ is already calculated in the previous (initial) step, the above equation is reduced computationally to one inversion and multiplication of matrices. Finally by combining the eq.(31) and eq.(33) the values for the shape functions of the five nodes, at a certain Gauss point can be written in parts as follows:

$$\Phi_{1:m}(\mathrm{x}_{Gp}) = \underbrace{p^T(\mathrm{x}_{Gp})\left[A_{1:m}(\mathrm{x}_{Gp})\right]^{-1} W_{1:m}(\mathrm{x})}_{Initial\ \Phi_{1:m}} + \underbrace{p^T(x_{Gp})[X]W_{1:(m+n)}(x)}_{influence\ of\ nodes\ n} \tag{35}$$

and

$$\Phi_n(\mathrm{x}_{Gp}) = p^T(\mathrm{x}_{Gp})\left[A_{1:m}(\mathrm{x}_{Gp})\right]^{-1} W_n(\mathrm{x}) + p^T(x_{Gp})[X]W_n(x) \tag{36}$$

The first term of eq.(35) describes the values for the shape functions of the initial four nodes which divide the domain into three sections. The second part of the eq.(35) modifies the initial part, when the fifth node is added into the problem domain in order to add the contribution of the added node and retain the partition of unity.

Fig. 2 and Fig. 3 show the shape functions of the first four nodes before and after the enrichment with the fifth node, for a cubic spline weight function. Fig. 4 illustrates the shape function of the fifth node. The factor d_m of the support domain is chosen to be 2.2.

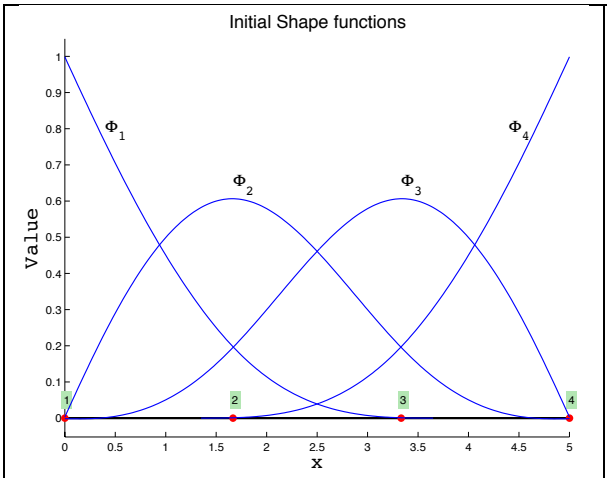


Fig. 2 - Initial nodes in the problem domain. Shape functions of each node.

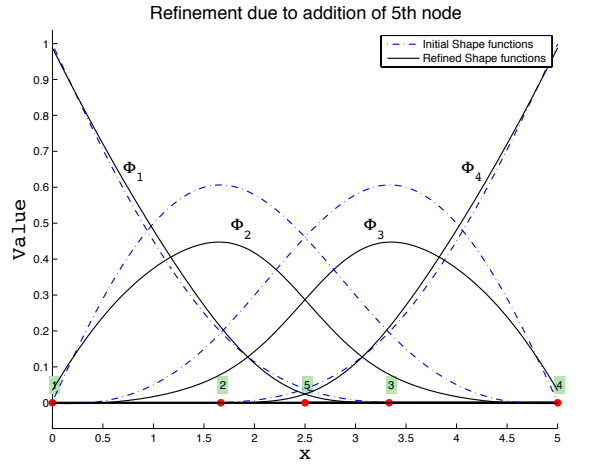


Fig. 3 - Refinement from four to five nodes. Shape functions of the first four nodes.

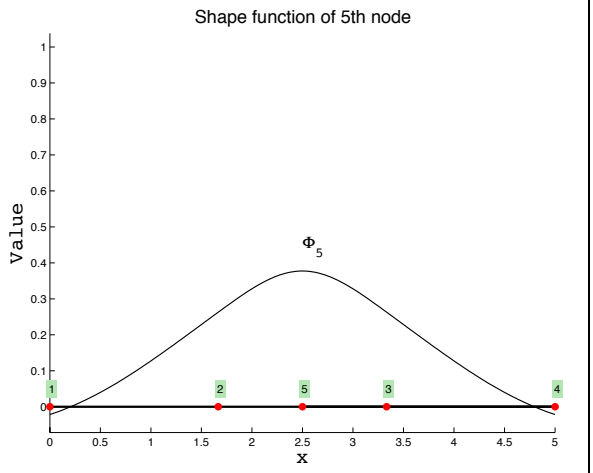


Fig. 4 - Shape function of the fifth node.

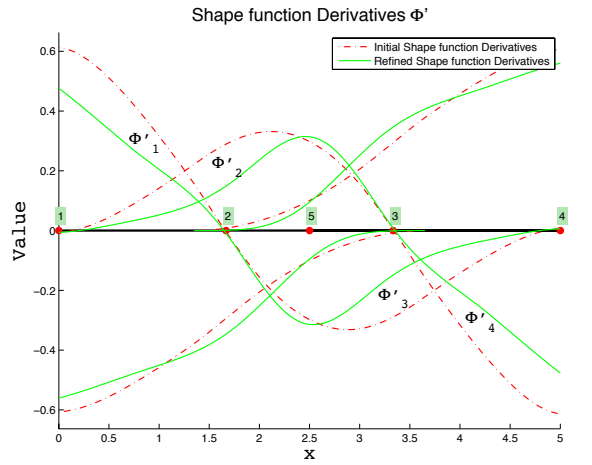


Fig. 5 - Refinement from four to five nodes. First derivatives of the shape functions of the first four nodes.

These figures illustrate the degree of modification of the existing shape functions at a specific domain when an extra node is added, since the shape function of the added node affects the values of all nodes of that DoI. The above process is directly extendable to more than one nodes, and in two or three dimensions.

The shape function derivatives which are necessary for the construction of the stiffness matrix (eqs.(13) and (15)) are as follows:

$$\Phi'_{1:(m+n)}(\mathrm{x}_{Gp})^{enriched} = [1 \quad 0][A_{1:(m+n)}(\mathrm{x}_{Gp})]^{-1} W_{1:(m+n)}(\mathrm{x})$$
$$-p^T(\mathrm{x}_{Gp})[A_{1:(m+n)}(\mathrm{x}_{Gp})]^{-1}[A'_{1:(m+n)}(\mathrm{x}_{Gp})][A_{1:(m+n)}(\mathrm{x}_{Gp})]^{-1} W_{1:(m+n)}(\mathrm{x}) \tag{37}$$
$$+p^T(\mathrm{x}_{Gp})[A_{1:(m+n)}(\mathrm{x}_{Gp})]^{-1} W'_{1:(m+n)}(\mathrm{x})$$

By employing the updated moment matrix of eq.(20) the derivatives can be calculated by parts. In order to illustrate the effect of the proposed refinement implementation to the shape function derivatives we consider again the problem shown in Fig. 1. The first part of the derivatives corresponds to the first four nodes and describes the values for the first derivative of the shape functions when the fifth node is not added in the problem domain. The second term modifies the initial part when the fifth node is added into the problem domain.

Fig. 5 illustrates the first derivative of the shape functions of the first four nodes, before and after the addition of the fifth node in the problem domain.

4 EXPLICIT ANALYTICAL FORM FOR TYPICAL ARRANGEMENT OF NODES

The case of uniform arrangement of nodes, though restricting, is used in a number of meshless analyses, especially in the main areas of even geometrically irregular bodies. For this case an analytical form of the derived shape functions could be produced, diminishing the computational effort of the shape function construction stage.

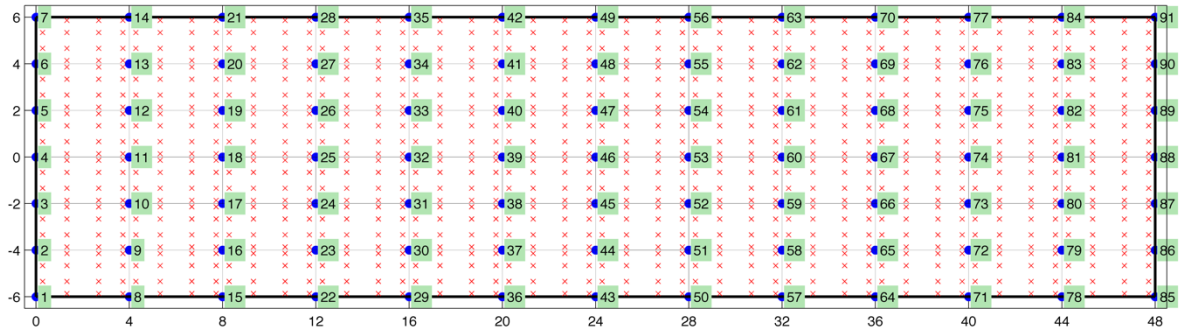


Fig. 6 – Typical Cantilever with uniform arrangement of nodes

In such cases, the moment matrix $\mathrm{A}(\mathrm{x}) = \sum_{i=1}^{n} w(\mathrm{x}-\mathrm{x}_i)\mathrm{p}(\mathrm{x}_i)\,\mathrm{p}^T(\mathrm{x}_i)$ depends only on the distance between the nodes of each domain of influence, therefore it could be written with respect to the unknown x. In cases where the above-mentioned moment matrix is of adequate condition, which is a prerequisite for the calculation of the shape functions in general, an analytical (not numerical) inversion follows which leads to the subsequent analytical formulation of the shape

functions $\Phi_i(x) = p^T(x)[A(x)]^{-1}W(x_i)$ at every point x. The same apply for the calculation of the shape functions derivatives.

It should be noted that the analytical form of the referred shape functions is quite complex and lengthy, depending on the arrangement of nodes. For example, for the cantilever of Fig. 6 one of the resulting shape functions for a point x is:

$\boldsymbol{\Phi_1(x)}=$ -(2*(abs(L) - abs(x))^3*(24*x*abs(L - x)^3 - 6*L*abs(L/2 - x)^3 + 12*x*abs(L/2 - x)^3 - 5*L*abs(L)^3 + 6*x*abs(L)^3 - 24*L*abs(L - x)^3 - 12*x*abs(L/2 - x)^2*abs(L) + 24*L*abs(L)*abs(L - x)^2 - 24*x*abs(L)*abs(L - x)^2 + 6*L*abs(L/2 - x)^2*abs(L)))/(L*(11*abs(L)^6 - 18*abs(L/2 - x)^2*abs(L)^4 + 18*abs(L/2 - x)^3*abs(L)^3 - 12*abs(L/2 - x)^3*abs(x)^3 - 30*abs(L)^5*abs(x) + 36*abs(L/2 - x)^3*abs(L - x)^3 - 10*abs(L)^3*abs(x)^3 + 30*abs(L)^4*abs(x)^2 + 54*abs(L)^3*abs(L - x)^3 - 54*abs(L)^4*abs(L - x)^2 - 48*abs(x)^3*abs(L - x)^3 - 36*abs(L/2 - x)^2*abs(L)^2*abs(x)^2 + 36*abs(L/2 - x)^2*abs(L)^2*abs(L - x)^2 + 144*abs(L)*abs(x)^2*abs(L - x)^3 + 48*abs(L)*abs(x)^3*abs(L - x)^2 - 144*abs(L)^2*abs(x)*abs(L - x)^3 + 144*abs(L)^3*abs(x)*abs(L - x)^2 + 12*abs(L/2 - x)^2*abs(L)*abs(x)^3 + 36*abs(L/2 - x)^2*abs(L)^3*abs(x) + 36*abs(L/2 - x)^3*abs(L)*abs(x)^2 - 36*abs(L/2 - x)^3*abs(L)^2*abs(x) - 144*abs(L)^2*abs(x)^2*abs(L - x)^2 - 36*abs(L/2 - x)^2*abs(L)*abs(L - x)^3 - 36*abs(L/2 - x)^3*abs(L)*abs(L - x)^2))

where L is the average length of the domain of influence.

5 HIERARCHICAL TYPE 1 REFINEMENT OF THE STIFFNESS MATRIX

The derivation of the shape functions and their derivatives in two parts leads to the adaptive formulation of the stiffness matrix when extra nodes are added in the problem domain. Therefore the shape functions of the initial m and the additional n nodes in the problem domain are given by eq. (27). Similarly, the partial derivatives of the shape functions are expressed by:

$$\Phi'_{1:m}{}^{enriched}(x) = \Phi'_{1:m}{}^{initial}(x) + \delta\Phi'_{1:m}(x) \tag{38}$$

$$\Phi'_{(m+1):(m+n)}{}^{additional}(x) = \Phi'_{(m+1):(m+n)}(x) \tag{39}$$

The enriched global stiffness matrix components is obtained from eq.(13):

$$K_{ij}{}^{enriched} = \int_{\Omega} B_i^{enriched\,T} \mathrm{E}\, B_j^{enriched} d\Omega \tag{40}$$

where for 2D elasticity problems

$$B_i^{enriched} = \begin{bmatrix} \Phi_{i,x}^{enriched} & 0 \\ 0 & \Phi_{i,y}^{enriched} \\ \Phi_{i,y}^{enriched} & \Phi_{i,x}^{enriched} \end{bmatrix}$$

$$= \begin{cases} \begin{bmatrix} \Phi_{i,x}^{initial} + \delta\Phi_{i,x} & 0 \\ 0 & \Phi_{i,y}^{initial} + \delta\Phi_{i,y} \\ \Phi_{i,y}^{initial} + \delta\Phi_{i,y} & \Phi_{i,x}^{initial} + \delta\Phi_{i,x} \end{bmatrix}, & i \le m \\ \begin{bmatrix} \Phi_{i,x}^{additional} & 0 \\ 0 & \Phi_{i,y}^{additional} \\ \Phi_{i,y}^{additional} & \Phi_{i,x}^{additional} \end{bmatrix}, & m \le i \le (m+n) \end{cases} \tag{41}$$

Further elaboration on the expression of the strain matrix $B_i^{enriched}$ leads to:

$$B_i^{enriched} = \begin{cases} \underbrace{\begin{bmatrix} \Phi_{i,x}^{initial} & 0 \\ 0 & \Phi_{i,y}^{initial} \\ \Phi_{i,y}^{initial} & \Phi_{i,x}^{initial} \end{bmatrix}}_{B_i^{initial}} + \underbrace{\begin{bmatrix} \delta\Phi_{i,x} & 0 \\ 0 & \delta\Phi_{i,y} \\ \delta\Phi_{i,y} & \delta\Phi_{i,x} \end{bmatrix}}_{\delta B_i}, & i \le m \\ \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}}_{B_i^{initial}} + \underbrace{\begin{bmatrix} \Phi_{i,x}^{additional} & 0 \\ 0 & \Phi_{i,y}^{additional} \\ \Phi_{i,y}^{additional} & \Phi_{i,x}^{additional} \end{bmatrix}}_{\delta B_i}, & m \le i \le (m+n) \end{cases} \tag{42}$$

$$= \left[B_i^{initial}\right] + \left[\delta B_i\right]$$

A similar expression can be derived for 3D elasticity problems.

After substituting eq.(42) into eq.(40) the stiffness matrix can be written as follows:

$$K_{ij}{}^{enriched} = \int_{\Omega} \left[\left[B_i^{initial}\right] + \left[\delta B_i\right]\right]^T \mathrm{E} \left[\left[B_i^{initial}\right] + \left[\delta B_i\right]\right] d\Omega$$

$$= \underbrace{\int_{\Omega} \left[\left[B_i^{initial}\right]\right]^T \mathrm{E} \left[\left[B_i^{initial}\right]\right] d\Omega}_{K_{ij}{}^{initial}} + \tag{43}$$

$$\underbrace{\int_{\Omega} \left[\left[B_i^{initial}\right]\right]^T \mathrm{E} \left[\left[\delta B_i\right]\right] d\Omega + \int_{\Omega} \left[\left[\delta B_i\right]\right]^T \mathrm{E} \left[\left[\delta B_i\right]\right] d\Omega + \int_{\Omega} \left[\left[\delta B_i\right]\right]^T \mathrm{E} \left[\left[B_i^{initial}\right]\right] d\Omega}_{\delta K_{ij}}$$

$$\boldsymbol{K}^{enriched} = \boldsymbol{K}^{initial} + \boldsymbol{\delta K} \tag{44}$$

The first part of $\boldsymbol{K}^{enriched}$ corresponds to the first m nodes, while the second term $\boldsymbol{\delta K}$ contains the contribution of the additional nodes and their interactions with the existing terms. In this respect, only $\boldsymbol{\delta K}$ matrix needs to be calculated at the refinement step, instead of recalculating the enriched matrix from the beginning, a process that saves substantial computational effort.

In order to demonstrate the stiffness matrix topology of the hierarchical h-type refinement, the 2D cantilever of Fig. 7 subjected to a parabolic load at the free end is considered. Problem details are given in Table 1.

Loading	P=1000 N
Young's modulus	E=3 x 10^7 N/m^2
Poisson's ratio	v=0.30
Height of the beam	D=6m
Length of the beam	L=24m

Table 1 – 2D cantilever problem details

The beam is analyzed with a 11x4 node distribution as shown in Fig. 7. The set of nodes with initial and added nodes is illustrated in Fig. 8. Background cells are considered for the numerical integration of the weak form. In each Gauss cell a 2x2 Gauss quadrature is used. A linear basis and a cubic spline weight function are used for the MLS approximation. The support domain is rectangular with dimension 2 times the nodal spacing. The penalty method was chosen for imposing the boundary conditions with a penalty factor of $\alpha = 10^{10}$.

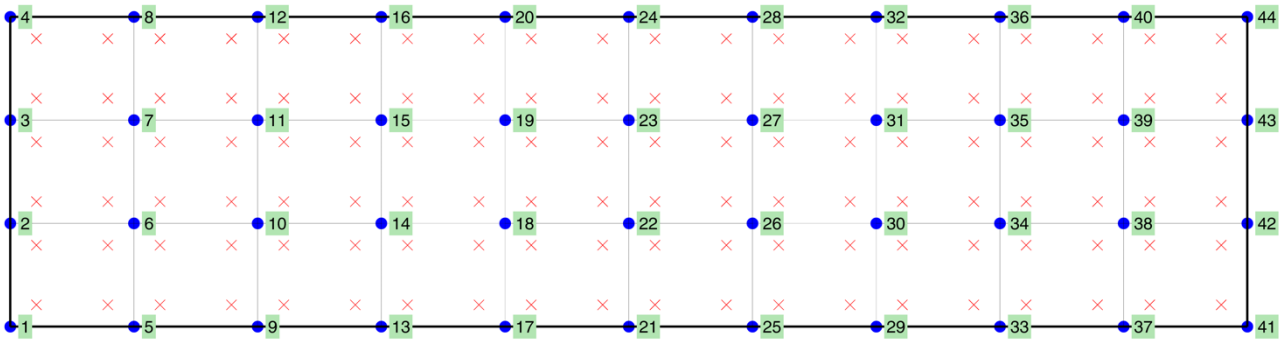


Fig. 7 - Initial arrangement of nodes ● and integration points ×

A node enrichment is subsequently performed with the addition of 8 new nodes in the problem domain indicated with the symbol **O** as seen in Fig. 8.

The calculation of the stiffness matrix is performed with standard procedure and the proposed two hierarchical schemes. For the second hierarchical procedure, only the additional matrix $\boldsymbol{\delta K}$ is computed and then added to the existing initial stiffness matrix.

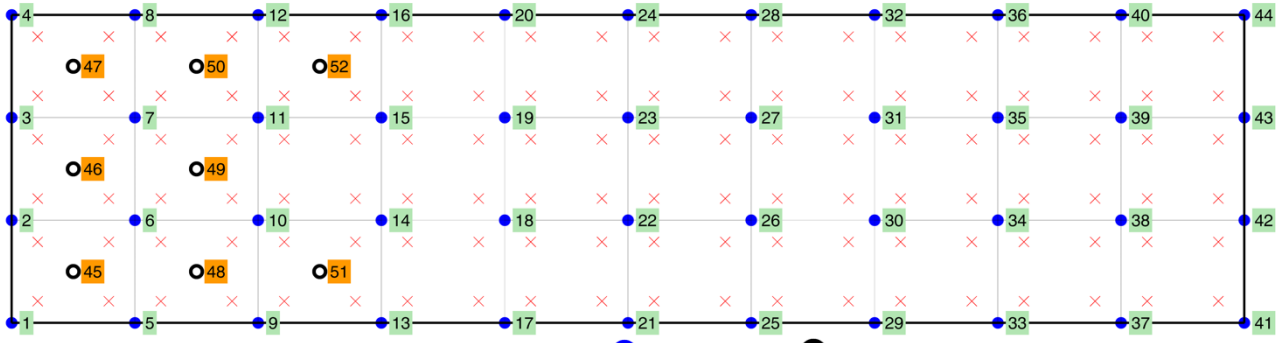
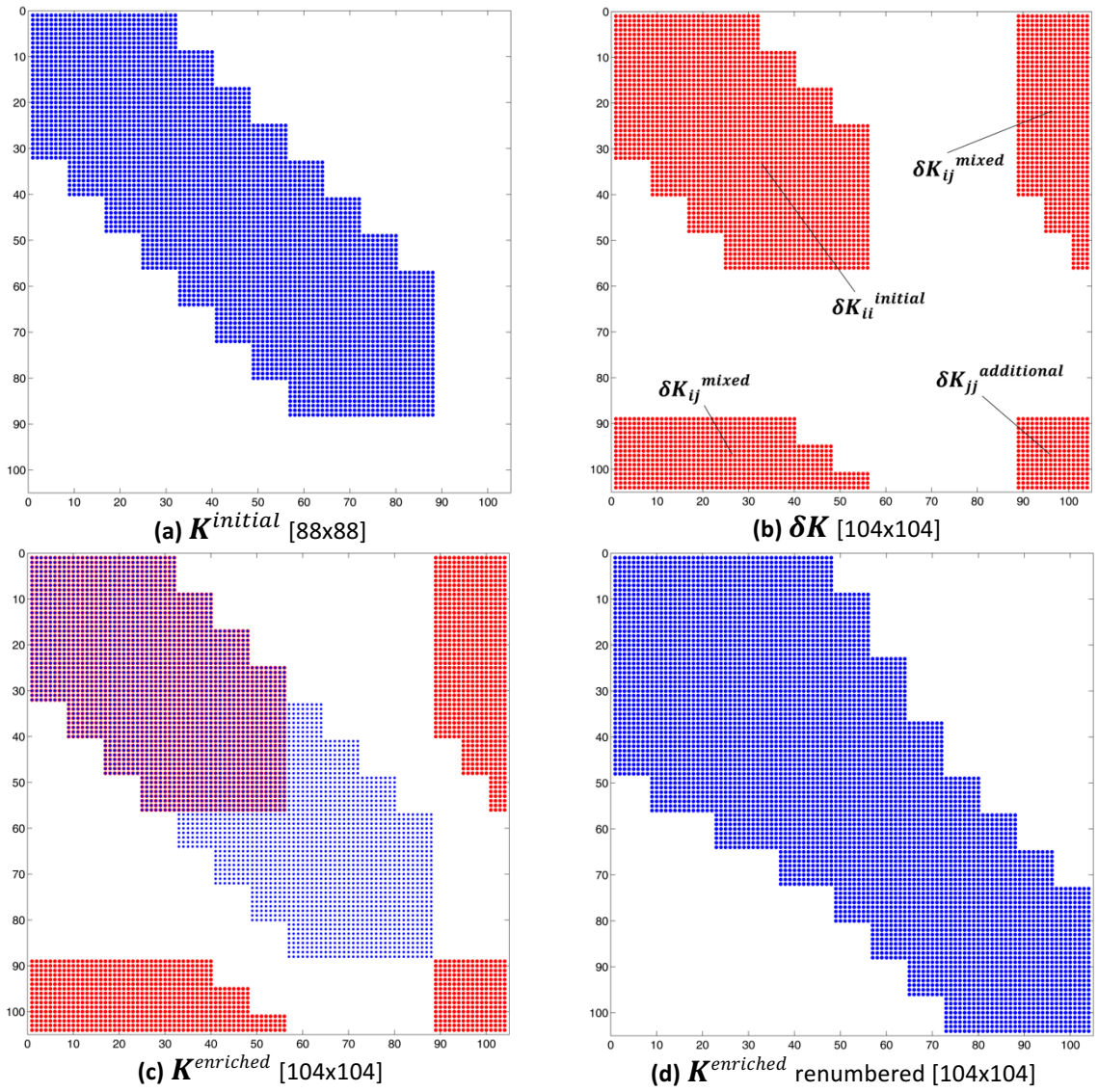


Fig. 8 – Initial nodes ●, added nodes ○

Fig. 9 illustrates the topology of the non-zero elements of the stiffness matrices before and after refinement. Matrix $\boldsymbol{\delta K}$ expresses the influence of the additional nodes. Fig. 9(d) depicts $\boldsymbol{K}^{enriched}$ after renumbering.



(a) $\boldsymbol{K}^{initial}$ [88x88] (b) $\boldsymbol{\delta K}$ [104x104]

(c) $\boldsymbol{K}^{enriched}$ [104x104] (d) $\boldsymbol{K}^{enriched}$ renumbered [104x104]

Fig. 9 – Non-zero elements of the stiffness matrices

6 HIERARCHICAL TYPE 2 REFINEMENT OF THE STIFFNESS MATRIX

The addition of extra nodes and their corresponding influence can be treated slightly different, with respect to the existing nodes, leading to a purely hierarchical formulation of the shape functions and derivatives and subsequently of the stiffness matrix. In this context the additional refining shape functions are treated hierarchically without altering the previously calculated shape functions and their derivatives.

The addition of the n nodes creates n shape functions according to eq.(36) and alters the m previous ones as follows:

$$\Phi_{1:m}(\mathrm{x}) = \underbrace{p^T(\mathrm{x})[A_{1:m}(\mathrm{x})]^{-1}W_{1:m}(\mathrm{x})}_{Initial\ \Phi_{1:m}} + \underbrace{p^T(\mathrm{x})[X]W_{1:m}(x)}_{(m+n)^{th}node\ influence\ \delta\Phi_{1:(m+n)}} \tag{45}$$

We consider the (m+n) node influence as additional refining functions that enrich the existing shape function field and choose to retain $\Phi_{1:m}(\mathrm{x})$ unchanged. In order to achieve this type of hierarchical formulation the last term of eq.(45) is omitted leading to:

$$\Phi_{1:m}(\mathrm{x}) = \underbrace{p^T(\mathrm{x})[A_{1:m}(\mathrm{x})]^{-1}W_{1:m}(\mathrm{x})}_{Initial\ \Phi_{1:m}} \tag{46}$$

which introduces an approximation since it violates the partition of unity.

The enriched shape function values is now applied for the extra nodes only, without the need for modification of the existing shape functions. Therefore, for the existing nodes:

$$\Phi_{1:m}(\mathrm{x})^{enriched} = \Phi_{1:m}(\mathrm{x})^{initial} \tag{47}$$

and for the additional nodes (refinement functions):

$$\Phi_{(m+1):(m+n)}(\mathrm{x})^{enriched} = \Phi_{(m+1):(m+n)}(\mathrm{x})^{hier} \tag{48}$$

where $\Phi_{(m+1):(m+n)}(\mathrm{x})^{hier}$ indicates the values of the shape functions of the extra nodes that are added hierarchically and are computed according to eq.(36). The partial derivatives of the shape functions that are used for the construction of the stiffness matrix are subsequently given by:

$$\Phi'_{1:m}(\mathrm{x})^{enriched} = \Phi'_{1:m}(\mathrm{x})^{initial} \tag{49}$$

$$\Phi'_{(m+1):(m+n)}(\mathrm{x})^{enriched} = \Phi'_{(m+1):(m+n)}(\mathrm{x})^{hier} \tag{50}$$

The global stiffness matrix component K_{ij} is assembled using eq.(13):

$$K_{ij}{}^{enriched} = \int_{\Omega} B_i^{enriched^T} \mathrm{E}\, B_j^{enriched} d\Omega \tag{51}$$

where

$$B_i^{enriched} = \begin{bmatrix} \Phi_{i,x}^{enriched} & 0 \\ 0 & \Phi_{i,y}^{enriched} \\ \Phi_{i,y}^{enriched} & \Phi_{i,x}^{enriched} \end{bmatrix} =$$

$$= \begin{cases} \begin{bmatrix} \Phi_{i,x}^{initial} + \delta\Phi_{i,x} & 0 \\ 0 & \Phi_{i,y}^{initial} + \delta\Phi_{i,y} \\ \Phi_{i,y}^{initial} + \delta\Phi_{i,y} & \Phi_{i,x}^{initial} + \delta\Phi_{i,x} \end{bmatrix} = \left[\boldsymbol{B_i^{initial}}\right], \ i \le m \\ \begin{bmatrix} \Phi_{i,x}^{additional} & 0 \\ 0 & \Phi_{i,y}^{additional} \\ \Phi_{i,y}^{additional} & \Phi_{i,x}^{additional} \end{bmatrix} = \left[\boldsymbol{B_i^{hier}}\right], \ m \le i \le (m+n) \end{cases} \tag{52}$$

Substituting eq.(52) into eq.(51), the strictly hierarchical stiffness matrix can be written in parts as follows :

$$\boldsymbol{K^{hier}} = \begin{bmatrix} \underbrace{K_{ii}{}^{\boldsymbol{initial}}}_{2m\ x\ 2m} & \delta K_{ij}{}^{mixed} \\ \delta K_{ji}{}^{mixed} & \underbrace{\delta K_{jj}{}^{additional}}_{2n\ x\ 2n} \end{bmatrix} \tag{53}$$

where

$$K_{ii}{}^{\boldsymbol{initial}} = \int_{\Omega} B_i^{\boldsymbol{initial}^T} \mathrm{E}\, B_i^{\boldsymbol{initial}} d\Omega \tag{54}$$

$$\delta K_{ij}{}^{mixed} = \delta K_{ji}{}^{mixed} = \int_{\Omega} B_i^{\boldsymbol{initial}^T} \mathrm{E}\, B_j^{hier} d\Omega \tag{55}$$

$$\delta K_{jj}{}^{additional} = \int_{\Omega} B_j^{hier^T} \mathrm{E}\, B_j^{hier} d\Omega \tag{56}$$

Through the procedure described above, the addition of extra nodes in the problem domain is taken into consideration without modification of the existing stiffness matrix. In eq.(53) $\delta K_{jj}{}^{additional}$ is the stiffness matrix of the additional forms, while $\delta K_{ij}{}^{mixed}$ expresses the interaction between the initial and the newly added nodes. This type of purely hierarchical formulation reduces further the computational effort for the assembly of the enriched stiffness matrix.

Moreover, specially tailored solution procedures can be applied that avoid refactorization of the initial stiffness matrix in a similar manner as applied in hierarchical FEM, which further reduces the solution effort of the resulting algebraic equations.

The proposed hierarchical type 2 refinement formulation is demonstrated using the 2D cantilever of Fig. 7 with the refinement scheme depicted in Fig. 8. The addition of the extra nodes is performed with the hierarchical concept described previously. The already calculated values of the shape functions of the first 44 nodes, are kept unaltered, and the influence of the shape functions of the 8 additional nodes to the initial nodes is omitted. In Fig. 10 the additional shape function of node 48 and the influence on the initial nodes is demonstrated.

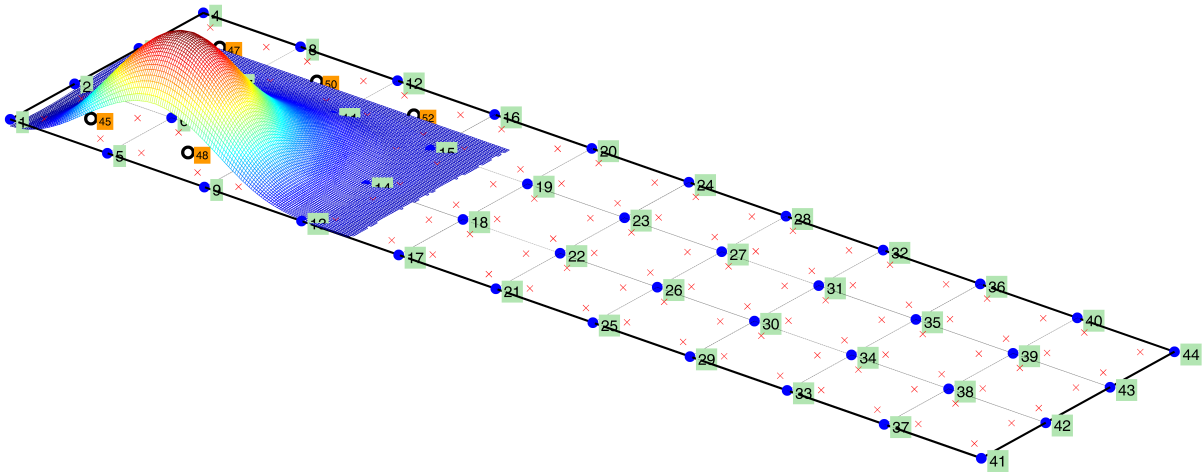


Fig. 10 - Additional shape function of the 48th node and its DoI

The stiffness matrix of the initial 44 nodes $K_{ii}^{\boldsymbol{initial}}$, is not affected by the additional nodes. To perform the hierarchical refinement only matrices δK_{ij}^{mixed} and $\delta K_{jj}^{additional}$ from eq.(55) and eq.(56) need to be computed. Fig. 11 illustrates the sparsity pattern of the matrices, used to assemble the final stiffness matrix of the cantilever in which $\delta K_{jj}^{additional}$ is (16x16) and δK_{ij}^{mixed} is (16x56). The final $K^{enriched}$ has the same pattern of non-zero terms as depicted in Fig. 9c and 9d of the type-1 hierarchical refinement. However, the stiffness terms corresponding to the initial nodes are not modified as per the additional terms illustrated in Fig. 9b.

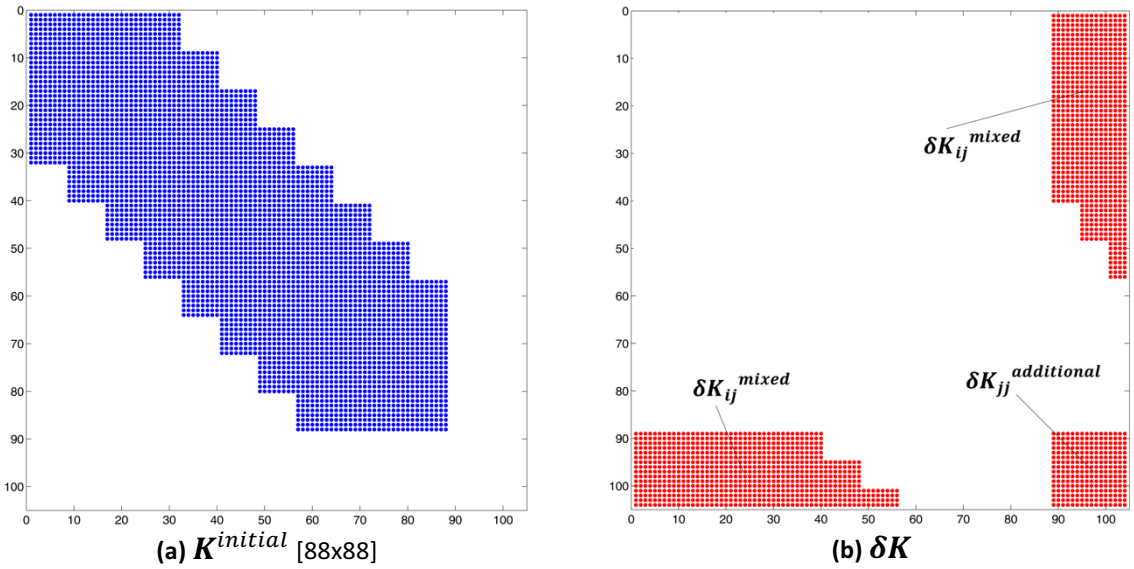


Fig. 11 - Non-zero elements of the stiffness matrices

In order to compare the two hierarchical formulations proposed, the energy and displacement norms are calculated before and after the addition of new nodes. For illustration purposes, only the shear stress field for each case is plotted in Fig. 12.

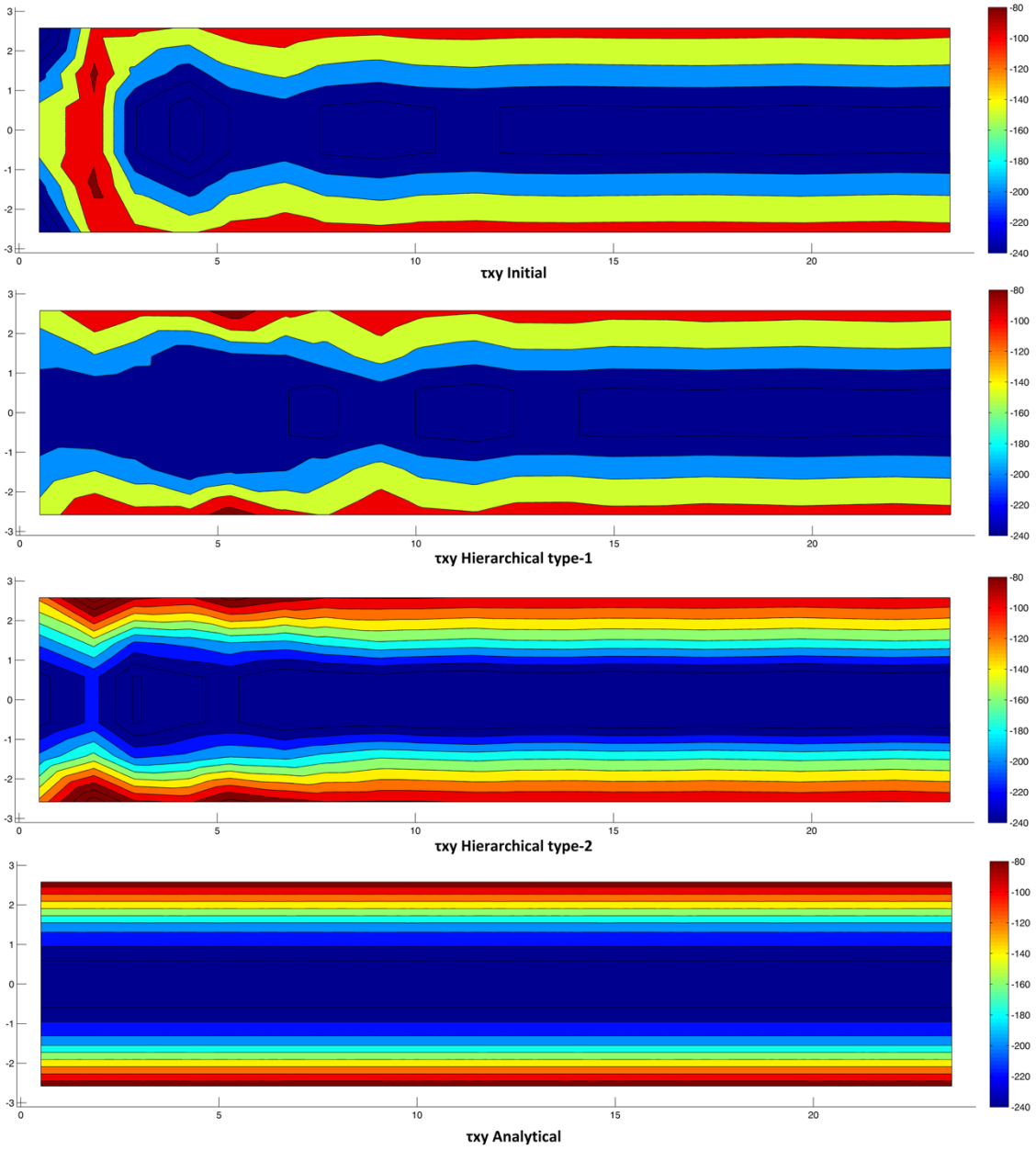


Fig. 12 - Shear stress field for the cantilever beam

	Initial vs Analytical	Hierarchical **type 1** vs Analytical	Hierarchical **type 2** vs Analytical
Energy norm	0.2362	0.0977	0.0688
Displacement norm	5,98E-05	7,37E-06	1,47E-06

Table 2 - Energy and Displacement norms for the cantilever beam

As can be seen from Fig. 12 and Table 2, type-2 hierarchical formulation gives results close to type-1 solution. The calculated displacement and stress fields are improved and overall much higher accuracy is obtained.

CONCLUSION

Two h-type hierarchically refinement schemes of the element free Galerkin method are proposed in which the expressions of the shape functions have been separated in two parts leading to a hierarchical decomposition of the stiffness matrix. In the first refinement scheme the contribution of the additional nodes in a problem domain can be achieved with the computation of an additional matrix **δK**, which is combined with the existing stiffness matrix describing the initial arrangement of nodes in the problem domain, to give the final stiffness matrix. In the second refinement scheme, the addition of extra nodes is taken into consideration, without the need for recalculation or modification of the existing stiffness matrix.

This work has been supported by the European Research Council Advanced Grant "MASTER—Mastering the computational challenges in numerical modeling and optimum design of CNT reinforced composites" (ERC-2011- ADG 20110209).

REFERENCES

[1] T. Belytschko, Y.Y. Lu and L. Gu, "Element-free Galerkin methods", *International Journal for Numerical Methods in Engineering*, vol. 37, 1994, pp. 229-256.

[2] Y.Y. Lu, T. Belytschko, L. Gu, "A new implementation of the element free Galerkin method", *Computer Methods in Applied Mechanics and Engineering*, vol. 113 (3-4), 1996, pp. 397-414.

[3] T. Belytschko, Y. Krongauz, D. Organ, M. Fleming, and P. Krysl, "Meshless methods: An overview and recent developments", *Computer Methods in Applied Mechanics and Engineering*, vol. 139, 1996, pp. 3–47.

[4] H.-J. Chung, T. Belytschko, "An error estimate in the EFG method", *Computational Mechanics*, vol. 21, 1998, pp. 91-100.

[5] S. Li, W.K. Liu, "Meshfree Particle Methods", Springer, 2004.

[6] S. Li, W.K. Liu, "Meshfree and particle methods and their applications", *Applied Mechanics Reviews*, vol. 55, 2002, pp. 1–34.

[7] V.P. Nguyen, T. Rabczuk, S. Bordas, and M. Duflot, "Meshless methods: A review and computer implementation aspects", *Mathematics and Computers in Simulation*, vol. 79, 2008, pp. 763–813.

[8] J.G. Wang, G.R. Liu, "A point interpolation meshless method based on radial basis functions", *International Journal for Numerical Methods in Engineering*, vol. 54, 2002, pp. 1623–1648.

[9] P. Metsis and M. Papadrakakis, "Overlapping and non-overlapping domain decomposition

methods for large-scale meshless EFG simulations", *Computer Methods in Applied Mechanics and Engineering*, vol. 229-232, 2012, pp. 128–141.

[10] A. Karatarakis, P. Metsis, M. Papadrakakis, "GPU-acceleration of stiffness matrix calculation and efficient initialization of EFG meshless methods", *Computer Methods in Applied Mechanics and Engineering*, vol. 258, 2013, pp. 63–80.

[11] K. S. Miller, "On the inverse of the Sum of Matrices", *Mathematics Magazine*, vol. 54 No. 2, 1981, pp 67-72.

[12] S. Fernández-Méndez, A. Huerta, "Imposing essential boundary conditions in mesh-free methods", *Computer Methods in Applied Mechanics and Engineering*, vol. 193 (12-14), 2004, pp. 1257-1275.

[13] C.K. Lee, C.E. Zhou, "On error estimation and adaptive refinement for element free Galerkin method: Part I: Stress recovery and a posteriori error estimation", *Computers and Structures*, vol. 82 (4-5), 2004, pp. 413-428.

[14] C. Wenterodt, O. von Estorff, "Optimized meshfree methods for acoustics", *Computer Methods in Applied Mechanics and Engineering*, vol. 200 (25-28), 2011, pp. 2223-2236.

[15] C.V. Le, H. Askes, M. Gilbert, "Adaptive element-free Galerkin method applied to the limit analysis of plates", *Computer Methods in Applied Mechanics and Engineering*, vol. 199 (37-40), 2010, pp. 2487-2496.

[16] K.T. Danielson, S. Hao, W.K. Liu, R.A. Uras, S. Li, "Parallel computation of meshless methods for explicit dynamic analysis", *International Journal for Numerical Methods in Engineering*, vol. 47, 2000, pp. 1323–1341.

[17] W.-R. Yuan, P. Chen, K.-X. Liu, "High performance sparse solver for unsymmetrical linear equations with out-of-core strategies and its application on meshless methods", *Applied Mathematics and Mechanics (English Edition)*, vol. 27, 2006, pp. 1339–1348.

[18] S.C. Wu, H.O. Zhang, C. Zheng, J.H. Zhang, "A high performance large sparse symmetric solver for the meshfree Galerkin method", *International Journal of Computational Methods*, vol. 5, 2008, pp. 533–550.

[19] O.C. Zienkiewicz, J.Z. Zhu, "The superconvergent patch recovery and a posteriori error estimates. Part 2: Error estimates and adaptivity", *International Journal for Numerical Methods in Engineering,* vol. 33, 1992, 1365-1382.

[20] M. Tabbara, T. Blacker, T. Belytschko, "Finite element derivative recovery by moving least square interpolants", *Computer Methods in Applied Mechanics and Engineering*, vol. 117(1–2), 1994, pp. 211–223.

[21] Szabó B, Babuška I. Finite Element Analysis. Wiley: New York, 1991.

[22] P. Metsis, N. Lantzounis, M. Papadrakakis, A new hierarchical partition of unity formulation of EFG meshless methods, Comput. Methods Appl. Mech. Engrg. 283 (2015) 782-805