MATHEMATICAL MODELING AND COMPUTING, Vol. 1, No. 1, pp. 1–16 (2014)

FEM elements enriched with meshfree functions: overview and application

Bekhta M. I., Savula Ya. G.

Ivan Franko National University of Lviv
1 Universytetska str., 79000, Lviv, Ukraine

(Received 1 April 2014)

A specific method of coupling FEM and meshless/meshfree methods is presented. This method is based on placing meshfree nodes inside the finite element and as a result improving the overall approximation on that element. Advantages and disadvantages of such approach are explained. It is shown that such approach is a version of a more general one. Numerical experiments are presented and analyzed.

Keywords: *finite element method, meshfree method, mixed interpolation, convection-diffusion, enriched elements*

2000 MSC: 65N30, 65N99, 76R50, 65D05

UDC: 519.6:517.925

1. Introduction

Meshfree methods are used more and more in numerical solving of boundary problems. There is a major variety of such methods and even more of their modifications. Among them the major groups are [1,6,8,9]:

— Reproducing kernel particle method (RKPM)
— Element free Galerkin method (EFG)
— Smoothed-particle hydrodynamics method (SPH)
— Radial base function method / Radial point interpolation method (RBFM/RPIM).

In our work we are interested in element free Galerkin method, which we will discuss in details later as it is required for construction of enriched finite elements.

Meshfree methods approximation, used for solving partial differential equations, is based on a set of nodes without the need for an additional mesh. Therefore, these methods are frequently used for problems where suitable meshes are prohibitively expensive to construct and maintain. This advantage of meshfree methods comes at the price of being considerably more time-consuming than their meshbased friends.

Also based on this main feature — absence of mesh — these methods are also very interesting in terms of using them for problems which requires adaptive refinement. The benefit of using such methods is obvious. You just need to add a set of new nodes in the needed area and recalculate the matrix only for the affected nodes. No re-mashing process is required.

Another big attractiveness of these methods is their shape functions, and their properties. One can construct a function of any order that is desired. It is also possible to construct base functions that have a desired order of continuity as well. Such functions have proven themselves to be very useful in places of presence of high gradients, concentrated forces and large deformations, as shown in the literature.

But looking from the practical standpoint, such methods are very expensive in terms of computational effort and also are suffering from the problems of applying Dirichlet type boundary conditions as well.

On the other hand, we have a mesh based method – FEM, which is fast in terms of computation time compared to meshless methods, and there is an easy way of applying Dirichlet boundary conditions.

There are a lot of authors who already suggested a variety of coupling and enrichment techniques. The idea is always the same — use the advantages of the methods and decrease the influence of their disadvantages. One can determine such basic types of combining methods:

— ramp function usage,
— based on consistency condition,
— based on Lagrange multipliers.

Our approach of coupling/enrichment is more of a practical application of the existing techniques rather than something completely new in that filed. This idea was derived from the arisen problems during the process of expansion of the existing coupling technique to solving a more complex problem, in terms of domain and as well as the equations. As a result, the idea came up to use a simple element and place meshless particles (nodes) on that element.

We will start from the construction of meshfree functions for the element free Galerkin method (EFG) first, then move on to how to combine EFG and FEM. Later we shall show how to enrich finite elements with meshfree functions and discuss some characteristics of such elements and show conducted numerical experiments.

2. Construction of meshfree functions

Before we start constructing the functions for meshfree approximation, let us discuss the problems and benefits of using such functions for approximation in Galerkin method.

Main downsides of this method are:

— Complicated calculations of functions values and problems derived from this process.
— Integration of the Galerkin weak form.
— Need of special approaches to impose Dirichlet boundary conditions (conditions of the first type/ essential conditions).

Let us briefly overview them. The main issue calculating values of such functions is a need to construct and solve systems of linear equations for each point, you want to find the value in. For a first derivative you will need to construct and find solutions for two systems. And so on. As a part of the calculation problem also is an algorithm for searching neighbour nodes which are required for construction of a shape function. But this is less problematic than the calculation and only really arises when you are adapting nodes distribution and using the non-constant dilation parameter ρ. As for the integration the main problems here are integration over a domain. This problem actually provided motivation for this article, and the fact that meshfree functions are of non-polynomial form and require high order quadrature with good precision. Problem with imposing essential conditions is from the fact that such functions do not have δ-property and as a result node values in such approximation are not the values of the approximated functions but just some coefficients in a linear combination of these functions. There are a lot of approaches to impose such conditions [4,6,8] – using optimization methods (Penalty method, method of Lagrange multipliers), or using coupling with FEM placed on the parts of boundaries where the essential conditions should be applied.

For construction of meshfree shape functions, the moving least squares method (MLS) is used. This method was originally introduced by Lancaster and Salkauskas [13]. Let us describe the process of constructing such functions.

First, let us assume that some function $u(x)$ is defined on some domain $\Omega \in \mathbb{R}^d$ and is smooth enough. Let this function be at least $u(x) \in C^0(\Omega)$. We can present an approximation near some fixed point $\tilde{x}$ as a product of two vectors – one is a vector of known "approximation" functions and the other is a vector of unknowns. This can be presented as follows:

$$u(x, \tilde{x}) \approx L_{\tilde{x}} u(x) = p^T(x)\alpha(\tilde{x}), \tag{1}$$

where

$$u(x,\tilde{x}) = \begin{cases} u(x), & \forall x \in \tilde{\Omega}(\tilde{x}), \\ 0, & otherwise, \end{cases} \tag{2}$$

here $\tilde{\Omega}$ is a small local domain near the fixed point $\tilde{x}$, and $L_{\tilde{x}}u(x)$ is a mapping. The vector $p(x)$, in practice, is chosen this way $p(x) = x^{\alpha} : |\alpha| \leqslant n$, which is nothing else but a polynomial basis of order n (one can use any other linearly independent set of functions). In general, one can construct such approximation on any set of basis functions of order n in the D-dimensional space Ω. For one-dimensional space, the set of monomials x^i is often used. And in the case of $k = n$, the number of elements in the vector is equal to the order number. For two/three dimensional space, there can be two types of polynomials, first one is a set of polynomials of total degree $\leqslant n$ or second one when the a polynomial degree for each variable is $\leqslant n$ [5, p.14].

Here we will consider a discrete version of the approximation. Which means that the used scalar products will be discreet rather than in form of integrals, as we will use these functions later for discrete approximation. So as we want our approximation to be in certain sense the best approximation for $u(x)$– we will use the MLS method and apply it to a square of weighted error residual of approximation. From this process we will obtain a system of equations to find the unknowns $\alpha(x)$. So let us minimize the weighted L_2 error residual, and we will get:

$$J_{\tilde{x}}(\alpha(\tilde{x})) = \sum_{i=1}^{r} \varphi\left(\tilde{x} - x_i\right) \left(L_{\tilde{x}}u(x) - u(x,\tilde{x})\right)^2 = \sum_{i=1}^{r} \varphi\left(\tilde{x} - x_i\right) \left(p^T(x_i)\alpha(\tilde{x}) - u(x_i)\right)^2. \tag{3}$$

So we deduced a way of obtaining unknown values of $\alpha(\tilde{x})$ from the nodal values of approximated function $u(x)$ and which relates on the set of nodes nearby the point of interest. Here the set of x_i is the neighbour nodal set, φ is the weight function which will be discussed later.

Continuing with the MLS process we need to minimize the residual functional (3). To do so, let us calculate the derivative of it with respect to the unknown variable α and equate the result to zero: $\frac{\partial J}{\partial \alpha} = 0$. We will obtain:

$$\sum_{i=1}^{r} \varphi\left(\tilde{x} - x_i\right) p(x_i) p^T(x_i) \alpha(\tilde{x}) = \sum_{i=1}^{r} \varphi\left(\tilde{x} - x_i\right) p(x_i) u_i, \tag{4}$$

After a few transformations, we can express α from this equation and put it into the local approximation representation (1) we will obtain

$$L_{\tilde{x}}u(x) = p^T(x) \left[\sum_{i=1}^{r} \varphi\left(\tilde{x} - x_i\right) p(x_i) p^T(x_i) \alpha(\tilde{x})\right] \sum_{i=1}^{r} \varphi\left(\tilde{x} - x_i\right) p(x_i) u_i. \tag{5}$$

Keeping in mind that the local point $\tilde{x}$ can be placed anywhere in the domain Ω, we can state that in such case $\tilde{x} \to x$, such approximation can be extended to the whole domain [12].

Finally, let us rewrite the approximation in a more suitable form:

$$u^h(x) = p^T(x) \left[\sum_{i=1}^{r} \varphi\left(\tilde{x} - x_i\right) p(x_i) p^T(x_i) \alpha(\tilde{x})\right] \sum_{i=1}^{r} \varphi\left(\tilde{x} - x_i\right) p(x_i) u_i \tag{6}$$

or in matrix-vector form with matrices and vectors explained below:

$$u^h(x) = p^T(x) \cdot M^{-1}(x) \cdot B(x) \cdot u, \tag{7}$$

here the matrix M is the Gramm matrix or momentum matrix. It is a $k \times k$ matrix where k is the length of the vector $p(x)$:

$$M(x) = \sum_{i=1}^{r} \varphi\left(\tilde{x} - x_i\right) p(x_i) p^T(x_i) \alpha(\tilde{x}) \tag{8}$$

the vector $B(x)$ is described with the use of the following equation

$$B(x) = \left[\varphi(x - x_1)p(x_1)\ \varphi(x - x_2)p(x_2) \ldots \varphi(x - x_r)p(x_r)\right]. \tag{9}$$

Finally, the vector of all base meshfree functions can be written as follows

$$N^T(x) = p^T(x) M^{-1}(x) B(x), \tag{10}$$

and from it we can obtain a formula for single function N_i placed in the node i

$$N_i^T(x) = p^T(x) M^{-1}(x) \varphi(x - x_i) p(x_i). \tag{11}$$

It can be easily shown, from the process of functions construction, that such approximation is exact for all the functions from the base vector $p(x)$.

Also from this process one can see that there is no way to write down the representation of such shape functions explicitly, as there is a need of solving a system of linear equations. But you can calculate them at as many points as you like.

Calculation of derivatives of such functions is also a complicated process as in this case for example to calculate a first order derivative you will need to solve not one but two systems of linear equations. Let us briefly look at this process. According to the product rule for derivatives, we shall obtain:

$$N^T(x) = \dot{p}^T(x) M^{-1}(x) B(x) + p^T(x) \dot{M}^{-1}(x) B(x) + p^T(x) M^{-1}(x) \dot{B}(x). \tag{12}$$

As you can see, to calculate a derivative you will require to solve one more system of equations.

Now let us discuss some of the interesting properties of meshfree functions. Let us start with continuity of such functions.

It is shown and proved that if the weight function φ was used for construction of meshfree functions is C^k, then these functions are of the same order of continuity, meaning that $N_i^\rho \in C^k$ as well [Liu, Tesi 19, Fries].

As for the convergence Liu and Belytschko showed that the a priori error bound is very similar to the bound for finite elements where the dilation parameter ρ in meshfree approximation plays the same role as the parameter h – the characteristic of mesh – plays for the finite elements.

$$||u - u^\rho||_{L_2(\Omega)} \leqslant C\rho^{n+1} ||u||_{H^{n+1}(\Omega)}, \tag{13}$$

here u^ρ is an approximation of a function u using meshfree functions, $H^{n+1}(\Omega)$ is a Hilbert space over Ω. It is said that, in practice, meshfree methods converge faster than corresponding finite elements methods. But as from theoretical point of view, you can see that the bounds look the same.

Even though it's often said that as it is a meshfree method you can place particles – meshfree nodes – anyway you what to – you should keep in mind that a stability and reliability of the method also depends on how they are located. To calculate a meshfree shape function, one will require that the matrix M is not singular. To achieve this, the scalar product used to calculate Gram matrix M should not be degenerated. So regularity of $M(x)$ is ensured by having enough particles in the neighbourhood of every point x and avoiding degenerate patterns. For this to be true, particles should meet such two conditions:

— $card S_x^\rho \geqslant k + 1$
— $\nexists F span\{p_0, ..., p_k\} \setminus \{0\}$ such that $F(x_I) = 0\ \forall\ I \in card\, S_x^\rho$,

here S_x^ρ is a set of indexes of particles such that $S_x^\rho = \{I \in \mathbb{N} : ||x_i - x|| \leqslant \rho\}$ — indexes of neighbour particles for a particle x which are used for calculation of values of the shape function for this particle. First condition states that there must be at least $k+1$ nodes in the domain of each mesh free function (k is the length of vector $p(x)$). The domain of a meshfree function is controlled by the parameter ρ and will be discussed below. Second statement means that in 1D dimension there should not be particles placed in the same position. For 2D case, particles cannot be placed on the same line. In 3D case – all the particles cannot be in the same plane.

Let us discuss the domain of a shape function and weight functions as these two items are related to each other.

Locality of meshfree function is guaranteed by the weight function – where the weight function is defined and is nonzero – there is also a shape function defined. So weight functions have two important properties in the case of meshfree functions: the first one is a guarantee of locality, the second one is order of continuity of shape functions. Weight functions sometimes are also called kernel or window functions. The size of the support of weight function is defined by the dilation parameter ρ. In most cases, ρ is used as a constant parameter for all the nodes and corresponding functions but in general – you can set different values of this parameter in different nodes. That can be very helpful in the case of non-equidistant particle placement or in cases when you have parts of domain with sparse particle placement and parts with dense placement of particles. As in such cases you will use a big dilation parameter ρ for the sparse area to meet conditions of particle placement described above, and as a result the same value will be used in a dense area as well. Hence for the particle from the dense area you will have a large number of neighbour particles and the matrices used for calculations will be generated over a large number of nodes which will take much time. Here are some examples of 1D weight functions:

$$\varphi_1(x) = \begin{cases} \frac{2}{3} - 4q^2 + 4q^3, & q \leqslant \frac{1}{2}, \\ \frac{4}{3} - 4q + 4q^2 - \frac{4}{3}q^3, & \frac{1}{2} \leqslant q \leqslant 1, \\ 0 & \text{otherwise}, \end{cases}$$

$$\varphi_2(x) = \begin{cases} 1 - 6q^2 + 8q^3 - 3q^4, & q \leqslant 1, \\ 0 & \text{otherwise}, \end{cases}$$

$$\varphi_3(x) = \begin{cases} (1-q^2)^k, & q \leqslant 1, \\ 0 & \text{otherwise}, \end{cases}$$

$$\varphi_4(x) = \begin{cases} q^{-k} - 1, & q \leqslant 1, \\ 0 & \text{otherwise}, \end{cases}$$

$$\varphi_5(x) = \begin{cases} e^{\frac{1}{(q^2-1)}}, & q \leqslant 1, \\ 0 & \text{otherwise}, \end{cases}$$

here $q = \frac{||x-x_i||}{\rho}$

Weight φ	Continuity	Function name
$\varphi_1(x)$	$\varphi \in C^2$	Cubic spline
$\varphi_2(x)$	$\varphi \in C^2$	4^{th} spline
$\varphi_3(x)$	$\varphi \in C^2$	$2k^{th}$ spline
$\varphi_4(x)$	$\varphi \in C^0$	Singular
$\varphi_5(x)$	$\varphi \in C^\infty$	Exponential

For the multidimensional spaces one can use a one dimensional weight functions. For example in 2D you can use a product of the same function, as in 1D, written for both dimension variables $\varphi_{2D}(x,y) = \varphi_{1D}(x)\varphi_{1D}(y)$. In such case you will have a rectangular support for a weight function and

as a result for a meshfree shape function the support will be rectangular as well. Or you can define a weight function as follows: $\varphi_{2D}(x,y) = \varphi_{1D}(\sqrt[2]{x^2+y^2})$. In such case you will have a circular domain support.

3. Coupling of meshfree and FEM functions

There are several approaches to coupling these two methods of approximation. Most common are coupling using rump functions or coupling with consistency conditions [2,5–7]. You can also couple both methods via domain decomposition technique, or binding them using Lagrange multipliers.

Let us overview the most common ones and start with coupling using rump functions. This approach was introduced by Belytschko [11]. In such approach, both types of functions are constructed separately and connected using so called rump function. Domain is divided into sub domains where only one method have an influence. Rump function is defined like this:

$$R(x) = \begin{cases} 0, & x \in \Omega^{FEM}, \\ 1, & x \in \Omega^{EFG}, \\ \sum\limits_{i \in I} N_i^{FEM}(x) & x \in \Omega^{*}. \end{cases} \tag{14}$$

Here the sum is over the indexes that satisfy $I = \left\{ i | x_i \in \Gamma^{EFG} \right\}$, Ω^{FEM} is a subdomain on which only FEM have influence, Ω^{EFG} – only EFG have influence, Ω^{*} is a sort of transition zone of a domain where both methods have their influence. In such case, shape functions for approximation can be defined for each node as follows (FEM or EFG):

$$\begin{array}{lcl} \forall x_i \in \Omega^{FEM} & : & N_i = N_i^{FEM}, \\ \forall x_i \in \Omega^{EFG} & : & N_i = N_i^{EFG}, \\ \forall x_i \in \Omega^{*} & : & N_i = (1 - R(x))\, N_i^{FEM} + R(x) N_i^{EFG} \end{array} \tag{15}$$

in Ω^{*} we shall have all EFG nodes those that their corresponding shape function supports intersect with finite elements.

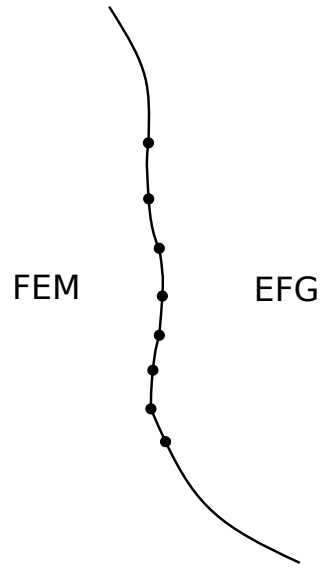

Fig. 1. Domain of the problem.

Now let us have a look at coupling using consistency conditions. Such technique was originally introduced by Fernandez-Mendez and Huearta [2]. Fries has modified such approach [6,7] and we will actually use this modified approach. As it is the best of these two and meets our needs in construction of enriched finite elements. In this case of coupling there will be four types of base functions:

$$\begin{array}{lcl} \forall x_i \in \Omega^{FEM} & : & N_i = N_i^{FEM}, \\ \forall x_i \in \Omega^{EFG} & : & N_i = N_i^{EFG}, \\ \forall x_i \in \Omega^{*} & : & N_i = \left(p^T(x) - p_{FEM}^T(x)\right) \alpha(x), \\ \forall x_i \in \Omega^{**} & : & N_i = \left(p^T(x) - p_{FEM}^T(x)\right) \alpha(x) + N_i^{FEM}. \end{array} \tag{16}$$

So as we can see, there are two types of functions on the so called transition area – part of a domain where both approximation methods have their influence. First one – functions on Ω^{*} – contains modified meshfree functions which have a support intersected with support of FEM ones but their nodes are not the nodes of FEM. Others – the ones on Ω^{**} – are the ones that have a particle placed at the same location where the FEM node is present and active.

Definition 1. *Finite element node is called active if its corresponding base function have influence on approximation of the function.*

Convergence analysis is already made for such mixed approximation, and for it there exists a priori error estimate. As enriched finite elements can be considered as a partial case of such approximation then this convergence analysis theory also can be applied to them. The next set of theorems will provide us with these estimates [2].

Theorem 1. *Let m be the order of consistency of the mixed approximation $u^h + u^\rho$ such that $m = p + q$ where p is the order of the finite element approximation u^h, and $q > 0$ is the order increment due to u^ρ. Suppose the following regularity conditions hold for the exact solution u: $u \in C^{m+1}(\bar{\Omega})$ and the weighting function φ: $\varphi \in C^0(\bar{\Omega})$ where Ω is bounded and $\partial\Omega$ is smooth. Finally, assume that the element size is small enough i.e.*

$$\frac{h}{\rho} \leqslant \min_{p+1 \leqslant r \leqslant m} \left(\tbinom{r}{p+1}\right)^{\frac{-1}{(r-(p+1))}}.$$

Then:

$$||u - (u^h + u^\rho)||_{L_\infty} \leqslant h^{p+1} \left(C_1 h^q + C_2 \rho^q\right) |u|_{W_\infty^{m+1}}.$$

Here the constants C_i are independent of finite element size h, or the dilation parameter ρ of the meshfree method.

As you can see on the left side of error estimate we have infinite norm but on the right-hand side we have just a seminorm, which is defined as:

$$|u|_{W_\infty^{m+1}} = \sum_{|\alpha|=m+1} \max_{x \in \bar{\Omega}} \left| \frac{\partial^{|\alpha|} u}{\partial x_1^{\alpha_1} \cdots \partial x_n^{\alpha_n}} \right|$$

with α here as multiindex. Also it is needed to say that such estimate coincides with the convergence results are obtained in empirical way. This theorem provided us with the estimate which depends on a sizes of elements and supports of meshfree frunctions. The next one will provide us with the estimate independent of those characteristics.

Theorem 2. *Under the assumptions of the previous theorem but without restrictions in element size the error bound will be*

$$||u - (u^h + u^\rho)||_{L_\infty} \leqslant h^{p+1} \left(C_1 h^q + C_2 \rho^1\right) |u|_{W_\infty^{m+1}}.$$

For proofs of these theorems and more details on convergence analysis look into [3,12].

4. Enriched finite elements

Now let us finally discuss what is the enriched finite element and its characteristics. By an enriched finite element we consider a simple finite element with somehow enhanced approximation, in our case – using meshfree functions. To do so, the meshfree nodes should be placed inside the element and corresponding meshfree functions should be constructed at those nodes. The idea of constructing such elements came in the process of removing downsides of pure meshfree method (Element free Galerkin method – EFG). As we mentioned before one of the downsides of this method is the fact that the meshfree functions support – place where function is not equal to 0 – can be partially outside the domain on which the problem is considered. As a result, in process of integration of Galerkin weak form we should keep that in mind and calculate only inside of the problems domain rather then just making the calculations over an intersections of meshfree functions supports. And this makes the process more complicated.

If the problem domain is simple enough, it is easy to just "cut off" the parts of meshfree functions supports which are outside the domain. Doing so, we will not need a background mesh for integration over a domain. Such problem is more of a practical kind rather than a theoretical — as it arises in the actual process of weak form integration.

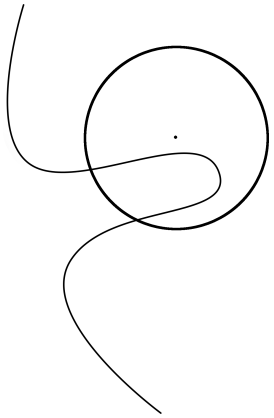

Fig. 2. Node near boundary placement.

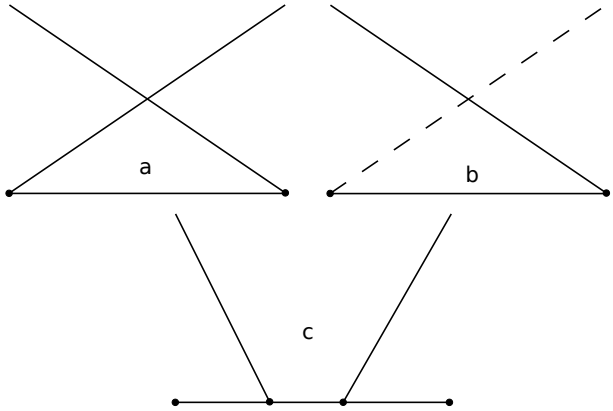

Fig. 3. Node near boundary placement.

Let us look closer at this problem. Let us consider some abstract problem which would be solved on a domain part of which is shown in the Fig. 2. And for approximation of the solution of this problem we want to use meshfree functions. So we should place meshfree nodes to construct base functions. For approximation to be smooth and correct, some of the nodes will be also placed near the boundary as shown in the image. As we can see the part of function supports goes out of the problem's domain, and as a result we cannot integrate just over functions support intersections, but we should also cut off the regions that are out of the domain. So the integration problem here is obvious and there is no simple answers to the question: how to construct integration quadrature to make the needed calculations.

From this problem, mentioned above, the idea to construct enriched elements was derived. We can easily integrate over simple domains – domains of finite elements. So meshing the domain using larger but simple in terms of domain complexity elements, we can then place meshfree nodes inside of those more simple domains and make needed calculations. So the finite element here just play the role of simple sub-domain of the main problem domain on which we can more easily use meshfree method.

Now let us look at these enriched finite elements more closely. Let us start from the 1D case. There can be several ways to enrich finite elements. At first let us consider the case when the order of meshfree shape function and the order of finite element are the same. Let us take a look at a simple, not modified element (see Fig. 3, *a*). Just putting meshfree nodes inside of it and using non modified meshfree functions will result in noncontinuous approximation. And that is not what we want to get. So to use such type of approximation – meshfree functions should be modified as we described before and as shown in [5]. In such case, meshfree function will meet consistency conditions and approximation will be continuous. But from the theorem (3) described below we can see that such approximation have no sense as well. Because in such case all mesh free functions will be equal to zero as they are of the same order as the base functions of finite elements. So in order for this type of approximation to work, we need to somehow make finite elements to have less influence, so there is no full set of basis functions on finite element, as in such case meshfree functions will not be 0 all over the element. We can see two ways to do so:

— remove at least one of the base functions from finite element (see Fig. 3, *b*). Dashed function is not used for approximation over this element.
— decrease the influence of finite element functions over their domain (see Fig. 3, *c*). Basically what it is – it is the same if we refined mesh on this element and than made all of the finite element functions, except the ones that are on boundary – inactive.

Theorem 3. *The modified meshfree shape functions of order $\leqslant n$ constructed over a finite elements of order m, where $n \leqslant m$ are degenerated, i.e. they are equal to 0 over such elements.*

Proof. It is easily derived from the way how the modification of meshfree functions is presented. As there is a difference between the vector $p^T(x)$ and its projection onto finite elements space $p^T_{FEM}(x)$.

The vector $p^T(x)$ contains polynomials of order less or equal to the order of finite elements. So as a result we have $p^T_{FEM}(x) = p^T(x)$, and from here $N^\rho_i = 0$. ■

For our purposes the second option of reducing the influence of finite element functions is a better choice. From the construction of such elements, we can see that they should satisfy and have the same properties as a mixed meshfree finite element approximation.

Theorem 4. *The modified coupled meshfree functions in finite element nodes are equal to 1.*

This can be easily shown as meshfree function is equal to 0 in that node and modified function for coupling is a sum of two functions: meshfree and finite element function. As the last one is equal to 1 at its node, then the sum is equal to 1 as well.

5. Numerical experiments

For numerical experiments we will start from interpolation of some functions first, then we will consider a couple of convection-diffusion problems which are known for their problematic solutions in cases when the Péclet number is high. Let us start with approximation of trigonometrical function $u(x) = \cos(2\pi x)$ on $x \in [0; 1]$. We will start from using a single enriched element with 4 additional meshfree functions of the first order added inside the element, so in total there will be 6 base functions on it. Dilation parameter is $\rho = 2.1h$, here h is the distance between two particles. The resulting set of functions is shown in Fig. 5, dashed line shows us modified finite element functions and solid lines are modified meshfree functions. The result shown in Fig. 4 is obtained on this element. You can see by the thick solid line the real function is depict, by dashed line – obtained approximation and by thin solid line – error of this approximation.

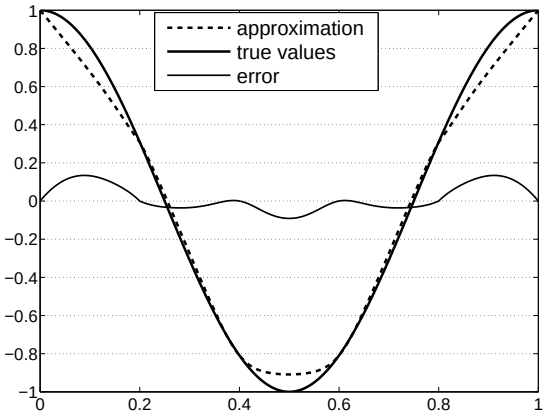

Fig. 4. Approximation using enriched element.

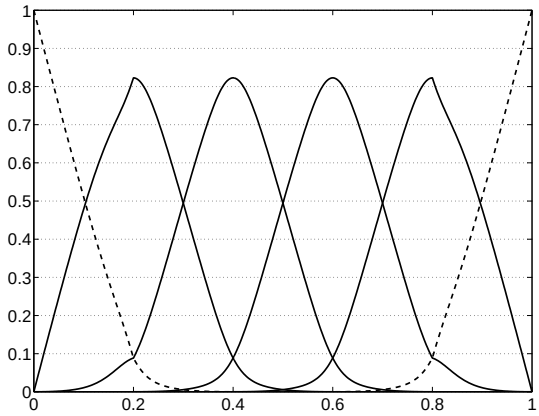

Fig. 5. Enriched element.

Let us compare these results to the ones obtained while using a finite element approximation shown in Fig. 6. Looking closer at error comparison in Fig. 7 we can see that enriched elements have better error than the simple finite elements. But that is just a visual analysis, let us calculate L_2 norms of errors and analyze in the way whose approach gives better results. Here, we should consider here three cases:

— approximation using finite elements and increasing number of the elements (C1);
— approximation using one enriched element and increasing number of meshfree particles included (C2);
— approximation using enriched finite elements with constant number of included meshfree particles and increasing the number of those elements (C3).

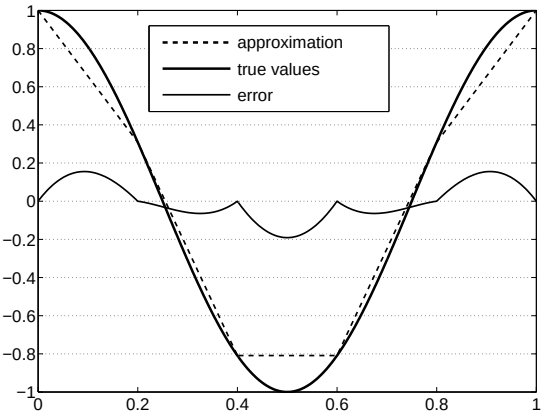

Fig. 6. Approximation using 5 simple linear finite elements.

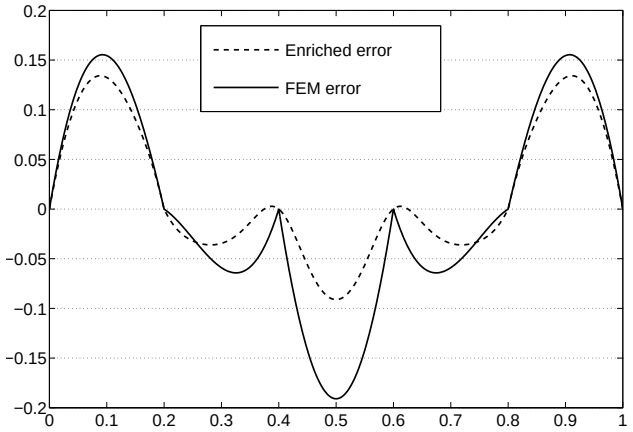

Fig. 7. Error comparison.

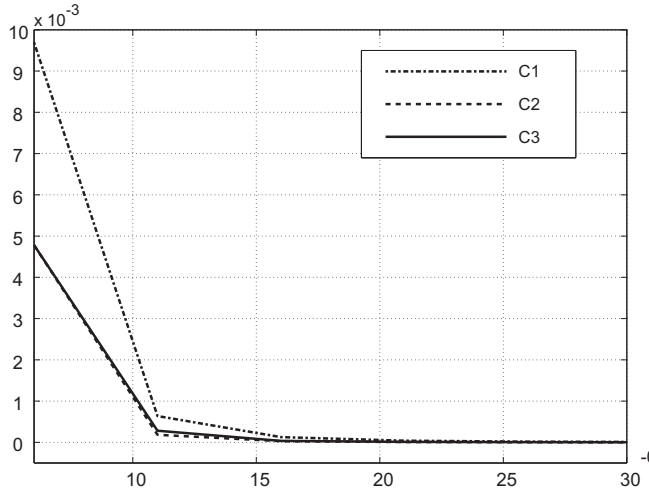

Fig. 8. Error norms.

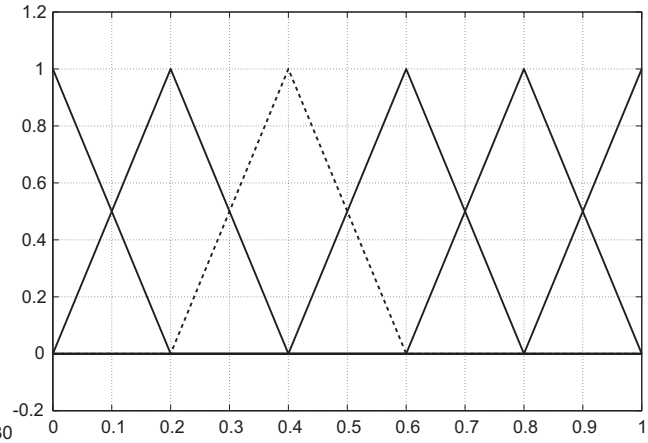

Fig. 9. Example of enriched element.

For these results to be relevant correct and comparable, we shall calculate error norms with respect to the number of shape functions used for approximation. The next table shows obtained results, N in the table is the number of used functions for approximation. As we can see, both from this table as well as from Fig. 8, the most optimal is the C2 case when we just put more particles inside the element. Increasing the number of enriched elements with constant number of particles – C3 – is somewhere in between C1 and C2 cases. Major difference in these error norms is at the smaller number of used functions, enriched elements have better approximation with lower number of total functions. This also proves the fact that in practice, as it was stated above, meshfree methods converges faster then simple finite elements.

N	C1	C2	C3
6	0.00969	0.004781	0.004781
11	6.38120E-4	1.82361E-4	2.81831E-4
16	1.27251E-4	2.85713E-5	3.55104E-5
21	4.04381E-5	7.88202E-6	1.11786E-5
26	1.65659E-5	2.95011E-6	4.53703E-6
31	8.00503E-6	1.33014E-6	2.17793E-6
51	1.03787E-6	1.49804E-7	2.84399E-7

An interesting fact is that if you push dilation parameter ρ to h – step between the particles, and if the particles are evenly distributed, then the base functions of enriched element will be going to the finite element functions constructed on the same set of nodes (see Fig. 10):

$$\text{When } \rho = \alpha h, \text{ and } \alpha \to 1, \quad \alpha > 1 : N_i^\rho \to N_i^h$$

Generally speaking, choosing the parameters of meshfree method is not always that straight forward. When in the case of the weight function $\varphi(x)$, everything is obvious, you choose one from the need of certain order of continuity for overall approximation, or functions for the vector $p(x)$ – you add those functions, which you would like to be fully reproduced by the approximation. But when it comes to selecting the ρ parameter, there are certain conditions previously discussed but those are required conditions, there are no optimal conditions for that parameter and in every specific case it can be selected differently. Analysis of this problem was partly made during the investigation of an optimal parameter selection for coupling meshfree and finite element methods in [15].

Now let us move on to convection-diffusion problem. We consider a one-dimensional static problem on $\Omega = [0, 1]$, which states the following: find the solution $u(x), x \in [0, 1]$ such that

$$-\frac{d}{dx}\left(p(x)\frac{du}{dx}\right) + \omega(x)\frac{du}{dx} = f(x) \tag{17}$$

with the boundary conditions on both ends:

$$u(0) = u_a, \quad u(1) = u_b,$$

$$\frac{du}{dx}(0) = u'_a, \quad \frac{du}{dx}(1) = u'_b.$$

Or some combination of them (You cannot use only boundary conditions of second type, because in such case there will be unlimited number of solutions with similarity to a constant). Here $u(x)$ is an unknown function whose physical meaning is a density allocation of a substance, $\omega(x) = \omega(x)(Pe)$ – represents the velocity of substance movement and depends on Pe – Péclet number, $f(x)$ is the function of inner sources of substance. As for solving this type of problems we want to use Galerkin method, let us give a weak formulation of such problem in the case of ordinary Dirichlet conditions on both ends:

$$a(u, v) + b(u, v) = l(v), \quad \forall v \in V,$$

where

$$V = \left\{ v(x) | \quad v \in W_2^{(1)}[0, 1], \quad v(0) = v(1) = 0 \right\},$$

$$a(u, v) = \int_0^1 p(x)\frac{du}{dx}\frac{dv}{dx}dx,$$

$$b(u, v) = \int_0^1 w(x)\frac{du}{dx}vdx,$$

$$l(v) = \int_0^1 fvdx.$$

It is known [14] that such bi-linear form is continuous and V-elliptic, and the linear form $l(v)$ is continuous in the case when $f \in L_2([0, 1])$, then the solution to such problem exists and is unique. Now let us move to actual numerical experiments and consider the previous problem with such data:

$p(x)$	$w(x)$	$f(x)$	$u(0)$	$u(1)$
1	100	100	0	0

We will solve such problem using three types of approximation as we have done in the previous example of function approximation: linear finite elements (C1), meshfree method coupled with FEM on boundaries (C2) and enriched finite elements (C3). To analyze the results, let us look straight at Fig. 10 and the table with corresponding results, showing the L_2 norm of the obtained solutions using different methods of approximation.

N	C1	C2	C3
6	0.20085	0.12649	0.12649
11	0.37710	0.44246	0.40823
16	0.33609	0.32328	0.32047
21	0.32853	0.32339	0.32020
26	0.32490	0.32153	0.31970
31	0.32295	0.32058	0.31965
51	0.32016	0.31922	0.31955
101	0.31891	0.31840	0.31910

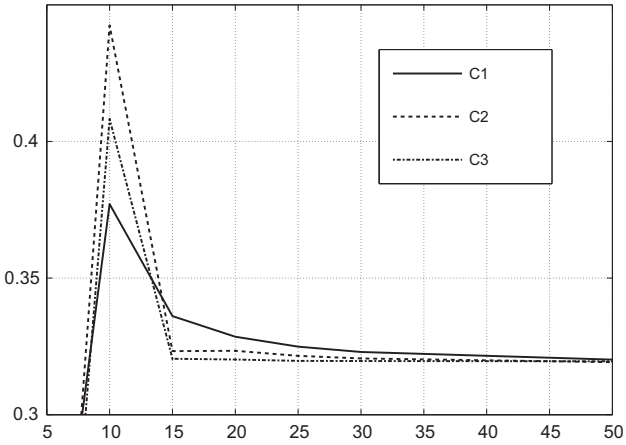

Fig. 10. Solution L_2 norms.

From this data (Fig. 10 and corresponding table) we can see that in case of usage of enriched elements and mesh free method coupled on the ends of domain with finite element the results are almost identical. Also from this we can see the fact that such methods converge faster then the finite element ones – which proves the fact we mentioned above – that in theory both finite elements and meshfree methods of the same order have the same order of convergence but in practice meshfree methods converge faster. Another important thing to remember when using enriched elements – fact that you can precalculate values of base functions and their derivatives and reuse these values on other elements with the same number of included meshfree particles. In other words, to decrease calculation costs of meshfree functions you can construct isoparametric elements.

Another interesting case of usage for enriched elements is to use meshfree functions build on non-polynomial basis. As it was said before, approximation built on meshfree functions can reproduce all the functions from the vector of basis functions it is built on. Keeping this in mind one can construct such kind of enriched elements using partial solutions in base vector for meshfree functions and therefor obtain a better approximation. Let us overview such approach using next example. Here we will consider convection-diffusion problem once more. It is known that if the coefficients of the equation are constants and on both ends of the domain Dirichlet conditions are used – solution of such problem contains e^{wx}. Let us construct such meshfree functions based on vector $p(x) = [1, x, e^{wx}]$ and compare the results to the second order functions – meaning functions constructed over $p(x) = \left[1, x, x^2\right]$. We will use slightly different parameters than in previous example

$p(x)$	$w(x)$	$f(x)$	$u(0)$	$u(1)$
1	100	0	0	1

In Fig. 11 we can see three solutions to the above example problem. First one, denoted with solid line, is the analytic solution to this problem which actually has such form:

$$u = \frac{1}{e^w - 1}\left(e^{wx} - 1\right). \tag{18}$$

As we can see, solution is built from e^{wx} and 1 so we could use only these two for constructing meshfree functions.

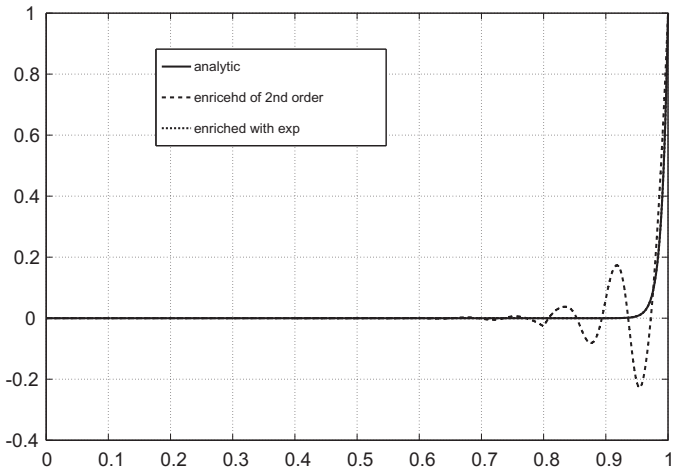

Fig. 11. Solutions for second example.

Then, there are also two more solutions in the Fig. – one obtained using second order enriched elements and second one – using enriched elements with meshfree functions enriched with exponential function. Both solutions were calculated over 5 elements and there were 4 meshfree functions added on them. As we can also see, second order approximation still have oscillation near the boundary while exponential approximation – does not. That is an expected result for the exponential approximation since such approximation reproduces the function from the base vector used for construction of meshfree functions and the actual solution of the problem can be built using these functions as a basis. Let us have a look at the L_2 error norm for such approximations. We will leave the number of meshfree functions on the element at 5 and only increase the number of elements. The results are presented in the next table

N	C1	C2
1	6.45744E-05	0.16023
2	2.66974E-10	0.04099
3	9.99200E-15	0.01158
4	2.91403E-15	0.00494
5	6.20370E-16	0.00234

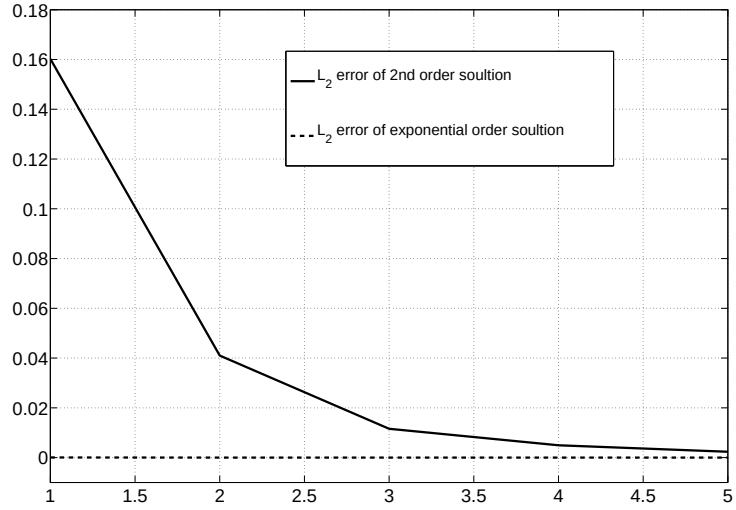

Fig. 12. L_2 error norms for 2nd example.

In the table above, N is the number of elements used and C1 and C2 columns are L2 error norms for exponential and 2nd order approximations respectively. From these data, it is obvious that exponential approximation is far more better than simple second order enriched elements approximation. Let us

move to the next example, where we will use some trigonometrical functions for approximation. We will consider a convection-diffusion problem again with such parameters:

$p(x)$	$w(x)$	$f(x)$	$u(0)$	$u(1)$
1	100	$100\sin(\pi x)$	0	1

To solve such problem, we will use a forth order enriched elements ($p(x) = \left[1, x, x^2, x^3, x^4\right]$) in comparison with enriched elements constructed with meshfree function built on the vectors $p = \left[1, x^2, x^3, e^{wx}\right]$ and $p = [1, e^{wx}, \sin(\pi x), \cos(\pi x)]$. Analytic solution of such problem is:

$$u = C_1 + C_2 e^{wx} + \frac{\pi}{C_3}\left(\sin(\pi x) - \frac{w}{\pi}\cos(\pi x)\right),$$

where

$$C_1 = \frac{-C_3 + w\left(e^w + 1\right)}{C_3 e^w - 1},$$

$$C_2 = \frac{C_3 - 2w}{C_3\left(e^w - 1\right)},$$

$$C_3 = \pi\left(w^2 + \pi^2\right).$$

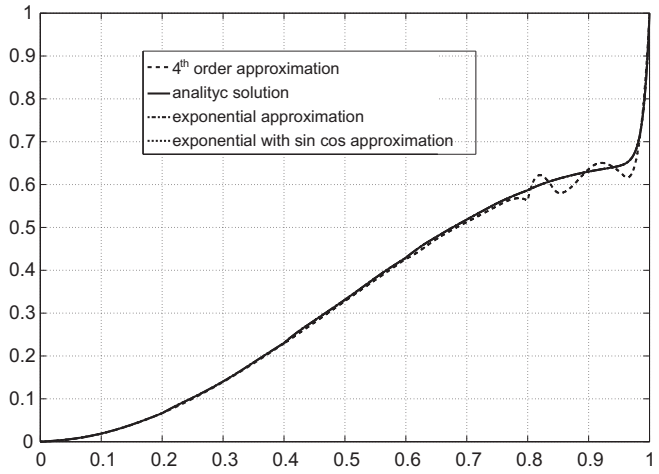

Fig. 13. Solutions for 3rd example.

Like in previous example, we will use five elements with four meshfree functions on each. We can see form the obtained solutions presented in Fig. 13 that both exponential and exponential with trigonometric functions gave us the same results, and these results coincide with the analytic solution. This can be explained by the fact that the most problematic function to approximate in the solution of the current problem is the exponential term e^{wx} and both of the approximations should reproduce this function completely as it is included in a basis for construction of meshfree functions. So from this results we can state that if you know the solution of homogeneous problem and you included it in the basis vector $p(x)$ for meshfree function construction you can solve nonhomogeneous problems and expect to get good results with low number of enriched elements in the case when the change in the right side of problems equation (which makes it non homogeneous) is not adding any functions which are hard to approximate, but even in such cases you will still get better results than if you did not include the terms of solution from homogeneous equation.

6. Conclusions

Let us briefly summarize the results. First is that the schema considered in this article for coupling finite elements and meshfree functions showed its efficiency in examined numerical experiments. Such approach gave us a possibility to use the power of meshfree methods and avoid the problems with integration and imposing boundary conditions moving those problems to finite elements, which can easily cope with them.

It was also shown that such approximation is a partial case of the existing ones and as a result all of the convergence and error analysis done for them can also be applied to our approximation.

Even though, we have shown that using enriched elements gave us a slightly better convergence than using simple finite elements did (in first example of convection-diffusion problem or in example of approximation of *cos* function) we wouldn't recommend it, as construction of meshfree functions is far more complicated and time consuming in calculations compared to simple finite elements. It is faster to just increase the number of finite elements. If you still want to use enriched finite elements than we would recommend to make them isoparametric. But this will help only if you use the same number of included meshfree functions on an element, otherwise you will still need to calculate the values on each of the elements anyway.

The real application for such elements can be found for solving problems where you know the partial solution, as it is shown in examples two and three for convection-diffusion problem. Based on the property of meshfree functions to completely reproduce the functions from base vector they are constructed on – you can add anything to that vector and improve the quality of the solution. This is very easy to do, and the results are fine starting from low number of elements. Based on this, using of such type of enriched elements with included partial solutions is great for example in domain-decomposition techniques where only the right side of the equation is changed.

Another good application for enriched elements are problems where higher order of continuity is required, as it can easily be achieved based on another meshfree function property previously discussed. You just need to choose a weight function with suitable continuity order. The only problem with such application is the fact that for higher derivatives of meshfree functions you will need to build and solve more and more systems of linear equations, which is time consuming.

As for further investigation, it is interesting to expand such approach for coupling to be used in two dimensional space. Another direction of investigation will be application of finite elements enriched with partial solutions for domain-decomposition problems.

[1] Huerta A., Belytschko T., Fernández-Méndez S., Rabczuk T., Meshfree methods, Encyclopedia of Computational Mechanics, John Wiley & Sons, Ltd. **1** (10), 279 (2004).

[2] Huerta A., Fernández-Méndez S., Enrichment and coupling of the finite element and meshless methods Int. J. Numer. Methods Eng. **48** (11), 1615 (2000).

[3] Fernández-Méndez S., Diez P., Huerta A. Convergence of finite elements enriched with mesh-less methods Numer. Math. **96** (1), 4 (2003).

[4] Fernández-Méndez S., Huerta A. Imposing essential boundary conditions in mesh-free methods Comput. Methods Appl. Mech. Eng. **193** (12–14), 1257 (2004).

[5] Fernández-Méndez S. Mesh-free methods and finite elements: friend or foe? Universitat Politécnica de Catalunya, 162 p. (2002).

[6] Fries T. P. A stabilized and coupled meshfree/meshbased method for fluid-structure-interaction problems Dissertation, In Braunschweiger Schriften zur Mechanik (H. Antes, Ed.), Nr. 59-2005, Braunschweig, 165 p. (2005).

[7] Fries T. P., Matthies H. G. A stabilized and coupled meshfree/meshbased method for the incompressible Navier-Stokes equations — part II: Coupling Comp. Methods in Appl. Mech. Engrg. **195**, 6191 (2006).

[8] Liu G. R. Meshfree Methods: Moving Beyond the Finite Element Method, Second Edition CRC Press, 792 p. (2009).

[9] Li S., Liu W. K., Meshfree and particle methods and their applications Applied Mechanics Review **55**, 1 (2002).

[10] Dolbow J., Belytschko T. An Introduction to Programming the Meshless Element Free Galerkin Method Archives of Computational Methods in Engineering, **5** (3), 207 (1998).

[11] Belytschko T., Organ D., Krongauz Y. A Coupled Finite Element–Element-free Galerkin Method Comput. Mech. **17**, 186 (1995).

[12] Li S., Liu W. K., Belytschko T. Moving least-squares reproducing kernel method Part I: Methodology and convergence Comp. Methods Appl. Mech. Engrg. **143** (1—2), 113 (1997).

[13] Lancaster P., Salkauskas K. Surfaces Generated by Moving Least Squares Methods Math. Comput. **37**, 141 (1981).

[14] Savula Ya. Numerical analysis of problems of mathematical physics by variational methods. Lviv, 221 p. (2004).

[15] Bekhta M., Savula Ya. Problem of optimal parameter selection for combination of meshfree and finite elements methods. Volyn Mathematical Visnyk. **8** (17), 15 (2011).

Скінченні елементи збагачені безсітковими функціями: огляд та застосування

Бехта М. І., Савула Я. Г.

Львівський національний університет ім. Івана Франка
вул. Університетська, 1, 79000, Львів, Україна

Запропоновано конкретний спосіб поєднання методу скінченних елементів та безсіткового методу. Цей метод базується на тому, що безсіткові вузли розташовуються в середині скінченного елемента і таким чином покрашують загальну апроксимацію на елементі. Розгялнуто та проаналізовано переваги та недоліки такого підходу. Показано що такий підхід підпорядковується більш загальній схемі поєднання методів. Наведено результати числових експериментів та проведено їх аналіз.

Ключові слова: *метод скінченних елементів, безсітковий метод, змішана апроксимація, адвекція-дифузія, збагачені елементи*

2000 MSC: 65N30, 65N99, 76R50, 65D05

УДК: 519.6:517.925