The 3rd International Conference on
"Computational Mechanics
and Virtual Engineering"
COMEC 2009
29 – 30 OCTOBER 2009, Brasov, Romania

QUASI-RANDOM INTEGRATION FOR MESHLESS COLLOCATION AND GALERKIN FORMULATIONS

Victoria E. Rosca[1], Elena Axinte[1], Carmen E. Teleman[1]
[1] Institution name, City, COUNTRY, e-mail (TNR 10 pt)

Abstract: *The While the basic ideas of meshless techniques are in a certain way simple, a truly meshless method is very difficult to develop. This depends on the proper choice of the interpolation scheme, numerical integration scheme and technique of imposing the boundary conditions. In this work we focus on the numerical integration task. Monte-Carlo integration techniques are promising schemes in the context of meshless techniques. Numerical examples based on elasticity problems are presented to examine the accuracy and convergence of Quasi-Monte Carlo integration SPH and EFG meshless methods.*
Keywords: *meshless, SPH, Moving least squares interpolation, keyword 4 Monte Carlo integration*

1. INTRODUCTION

In engineering, one often has a number of data points, as obtained by sampling or some experiment, and tries to construct a function, which closely fits those data points. The so-called *meshless methods* (MM) construct approximations from a set of nodal data without the need for any (finite - element) *a priori* connectivity information between the nodes. In general, a meshless method uses a local interpolation or approximation to represent the trial function with the values (or the fictitious values) of the unknown variable at some randomly located nodes.
The fast convergence, ease of adaptive refinement, trivial rising of the consistency order and the continuity of derivatives up to the desired order are features of this class of methods.
There are two ways to construct *approximations of a function* using meshless methods:

- a continuous a form, or reproducing kernel (RK) approximation
  Smoothed Particle Hydrodynamics (SPH) [10] and *Reproducing Kernel Particle Method* (RKPM) [7] are two representative methods of RK.
- discrete form
  There are several types of discrete approximation functions. Among these are: *Moving least square* (MLS) functions, *Partition of unity* (PU) functions, or *hp-cloud* functions, as representatives. Surveys can be found in [8].

The meshless approximations functions constructed in continuous or in discrete are used to approximate the displacement (or other variables of interest) to solve applied mechanics problems. The approximations are used as approximations of the *strong forms* of partial differential equations (PDEs), and those serving as approximations of the *weak forms* of PDEs to set up a linear system of equations. To approximate the *strong form* of a PDE using a particle method, the partial differential equation is usually discretized by a specific *collocation technique*. To approximate the *weak* form of a PDE various *Galerkin weak formulations* are used.

2. CONSTRUCTION OF MESHLESS APPROXIMATION FUNCTIONS

2.1 Construction of MLS interpolants

We consider the problem of fitting an approximation $u^h(\mathbf{x}_I)$ with a set of N data values $\hat{u}_I$, defined at the points $\mathbf{x}_I$.. We assume the approximation function is given by a set of monomials p_j

$$u^h(\mathbf{x}) = \sum_{j=1}^{m} p_j(\mathbf{x}) a_j = \mathbf{pa} \qquad (1)$$

where $\mathbf{p}$ is a set of m linearly independent polynomial functions, $\mathbf{p}^{\mathbf{T}}(\mathbf{x}) = [p_1(\mathbf{x}), \ p_2(\mathbf{x}), \ \ldots, \ p_m(\mathbf{x})]$ and $\mathbf{a}$ is a set of parameters to be determined. For example, for a 2-D problem, one can choose: $\mathbf{p}^T(x) = [1, x, x^2]$, i.e., $m = 3$.

The coefficients a_j in Eq. (1) are determined by enforcing the interpolation to passes through all nodes within the support domain. To perform the fit, a least square scheme is adopted. The basic idea is to minimize the square of the distance between N data values defined at the points $\mathbf{x}_i$ and an approximating function evaluated at the same points weighted with a certain function w:

$$J = \frac{1}{2}\sum_{I=1}^{N} w_I(\mathbf{x}-\mathbf{x}_I)\left[\mathbf{p}(\mathbf{x}_I)\mathbf{a}-u(\mathbf{x}_I)\right]^2 \qquad \textbf{(2)}$$

where the minimization is performed with respect to the values of **a**.
The domain of influence of node can be controlled by the size of support of the function w, and this choice is therefore important.
Several weight functions are available in the literature. A review of some of the possibilities can be found in [8]. Here, the weight function chosen is the cubic spline [12], expressed as a function of the normalized distance s:

$$w(s) = \begin{cases} 2/3 - 4s^2 + 4s^3 & \text{for } s \le 1/2 \\ 4/3 - 4s + 4s^2 - 2/3\, s^3 & \text{for } 1/2 < s \le 1 \\ 0 & \text{for } s > 1 \end{cases} \qquad \textbf{(3)}$$

where r_I denotes the size of the *domain definition* (the support) of the weight function around each point x_I. and r is the Euclidian distance, $r = \|\mathbf{x}-\mathbf{x}_I\|_0$ and s=r/ r_I. Typically, domains of influence are circular or rectangular in 2D.
Functional J can be minimized by setting the derivative of J from Eq. (2) with respect to **a** equal to zero. As a consequence, the approximation $u^h(\mathbf{x})$ for the function u become:

$$u^h(\mathbf{x}) = \sum_{I}^{N} \phi_I\left(\|\mathbf{x}-\mathbf{x}_I\|\right)\cdot\hat{\mathbf{u}}_I = \mathbf{\Phi}^{\mathbf{T}}(\mathbf{x})\cdot\hat{\mathbf{u}} \qquad \textbf{(4)}$$

in which the shape function $\phi_i(\mathbf{x})$ for the i-th node is given by

$$\phi_I(\mathbf{x}) = \mathbf{p}(\mathbf{x})\mathbf{A}^{-1}(\mathbf{x})\mathbf{B}_I(\mathbf{x}) \qquad \textbf{(5)}$$

where:

$$\mathbf{A}(\mathbf{x}) = \sum_{I=1}^{N} w\left(\|\mathbf{x}-\mathbf{x}_I\|\right)\mathbf{p}_I^T\mathbf{p}_I$$
$$\mathbf{B}_I(\mathbf{x}) = w\left(\|\mathbf{x}-\mathbf{x}_I\|\right)\mathbf{p}_I \qquad \textbf{(6)}$$

The set $\{\phi_I(\mathbf{x})\}$ define interpolation functions of the MLS approximation for each data item $\hat{\mathbf{u}}_I$.

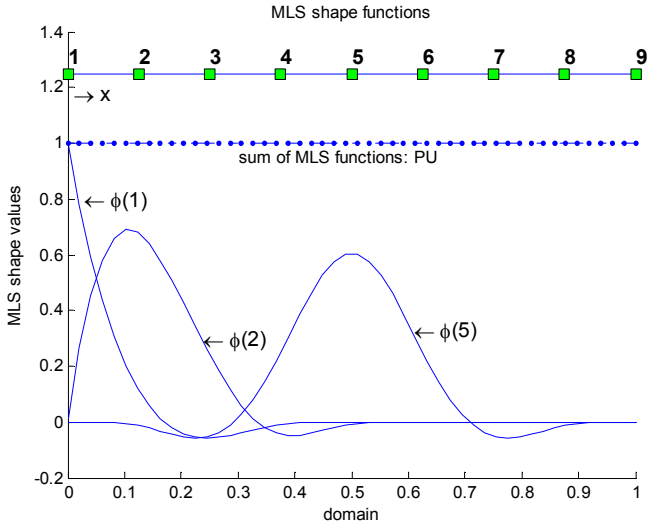

Figure1: Typical MLS shape functions and the 9 node model

Figure 1 presents typical shape functions ϕ_I at nodes I = 1, 2, and 5, evaluated using the 4th order spline weight function of Eq.(4).

These functions have been evaluated with a quadratic basis function and the size of the support chosen as r_I / L = 0.4375 (or r_I = 3.5·|**x** - **x**$_I$|). We note that shape functions located at equal distances on either side of the centers of a model with uniform nodal spacing are mirror images of each other. For example, for the 9-node model presented above, ϕ_1 and ϕ_9, φ_2 and ϕ_8, etc., are mirror images about the center.

2.2 Smooth Particle Hydrodynamics SPH

Smoothed Particle Hydrodynamics (SPH method) is one of the earliest particle methods which use kernel methods. In 1977, Lucy [10] and Gingold and Monaghan [5] simultaneously formulated the so-called Smoothed Particle Hydrodynamics, which is known today as SPH.
The SPH is a representative method for the strong form collocation approach.
In the original form of the SPH, the method was used to approximate the density ρ in fluid problems by probabilistic methods. The physical meaning of the kernel function is the probability of a particle's position. Applying this idea to fluid elements, the density can be considered proportional to the average number of particles per unit volume, i.e., proportional to probability density of finding a particle in a given volume element. Statistical methods can be used for this purpose. Based on this statistical framework, the true density, which is estimated by an integral form, is then evaluated numerically by Monte Carlo techniques.
The estimate of the true density is evaluated as

$$\rho^h(\mathbf{x}) = \int_{\Omega_y} w(\mathbf{x}-\mathbf{y})\rho(\mathbf{y})\mathbf{d}\Omega_y \qquad \textbf{(7)}$$

Usually a positive function, such as the Gaussian function or spline functions are usually employed.
Since $\rho(\mathbf{y})$ is unknown, the above expression cannot be evaluated directly. Consequently, the integral will be evaluated by Monte Carlo techniques, as follows:
The total mass is a known value. Also, it is

$$\mathsf{M} = \int_{\Omega_x} \rho^h(\mathbf{x})\mathrm{d}\Omega_x \qquad \textbf{(8)}$$

then, the standard Monte Carlo integration technique estimate the density at chosen randomly distributed points $x_1, \ldots, x_N$, as being the average value of the mass:

$$\rho^h_{\mathrm{MC}} = \frac{\mathsf{M}}{N}\sum_{I=1}^{N} w(\mathbf{x}-\mathbf{x}_I) \qquad \textbf{(9)}$$

Usually, the random points, which serve as quadrature points, are called *particles*.
The SPH was introduced for unbounded problems and applying it to bounded cases leads to major problems. This is due to its failure to meet the reproducing conditions of even 0-th order near the boundaries (as depicted in Figure 2).

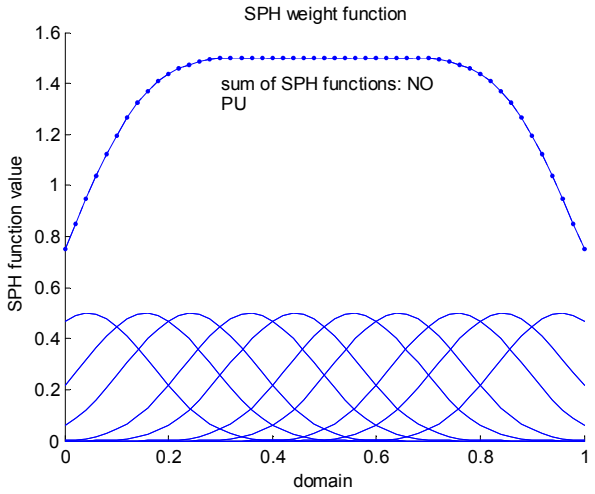

Figure 2: SPH weighting functions.

3. EFG METHOD - DISCRETE EQUATIONS

We consider the following two-dimensional problem, on the domain Ω bounded by Γ:

$$\Delta\sigma + b = 0 \text{ in } \Omega \quad \textbf{(10)}$$

$$\begin{aligned} N\sigma &= \bar{t} \quad \text{on } \Gamma_t \\ u &= \bar{u} \quad \text{on } \Gamma_u \end{aligned} \quad \textbf{(11)}$$

where σ is the stress tensor, which corresponds to the displacement field u and b is a body force vector, $\bar{t}$ is the prescribed traction vector on Neumann boundary Γ_t, $\bar{u}$ is the vector of prescribed displacements on Dirichlet boundary Γ_u, Δ is a linear gradient operator, and N is the matrix of direction cosine components of a unit normal to the domain boundary.

Applying the Galerkin global formulation, we multiply the equilibrium equation Eq. (10) with the test function and integrate on the entire global domain Ω; the surface equations Eq. (11) is also multiplied by the test function and integrated on Γ_t; then these two equations are subtracted:

$$-\int_{\Omega} \delta\boldsymbol{u}\left(\Delta^{\mathrm{T}}\boldsymbol{\sigma} + \mathbf{b}\right)\mathrm{d}\Omega + \int_{\Gamma_t} \delta\boldsymbol{u}\left(\mathbf{N}\boldsymbol{\sigma} - \bar{\mathbf{t}}\right)\mathrm{d}\Gamma = 0 \quad \textbf{(12)}$$

By applying the divergence theorem, Eq. (17) may be written in a symmetric weak form as:

$$\int_{\Omega} \nabla^{s}\delta\boldsymbol{u}:\boldsymbol{\sigma}\,\mathrm{d}W - \int_{\Omega} \delta\boldsymbol{u}\mathbf{b}\,\mathrm{d}\Omega - \int_{\Gamma_t} \delta\boldsymbol{u}\bar{\mathbf{t}}\,\mathrm{d}\Gamma = 0 \quad \textbf{(13)}$$

If the trial function $\boldsymbol{u}$ and the test function $\delta\boldsymbol{u}$ are identical, the numerical method is known as a Galerkin method. This is the reason why Belytschko, Lu et Gu [3] named their method the *Element-Free Galerkin Method*.

Since MLS interpolation functions do not satisfy the Kronecker delta condition at the node, $\phi_i\left(\mathbf{x}_j\right) \neq \delta_{ij}$, essential boundary conditions cannot be directly applied to nodal values. So, to impose the essential boundary conditions to be satisfied is not as easy as in standard FEM. Alternatives however, can be found by using Lagrange multipliers in the potential energy functional, see Belytschko et al. [3]. For global weak formulation, the following relation is obtained:

$$\int_{\Omega} \nabla^{s}\delta\boldsymbol{u}:\boldsymbol{\sigma}\,\mathrm{d}\Omega - \int_{\Omega} \delta\boldsymbol{u}\mathbf{b}\,\mathrm{d}\Omega - \int_{\Gamma_t} \delta\boldsymbol{u}\bar{\mathbf{t}}\,\mathrm{d}\Gamma + \int_{\Gamma_u} \mathrm{d}\lambda(\boldsymbol{u} - \bar{\mathbf{u}})\,\mathrm{d}\Gamma + \int_{\Gamma_u} \lambda\,\mathrm{d}\boldsymbol{u}\,\mathrm{d}\Gamma = 0 \quad \textbf{(14)}$$

To obtain the discrete system equations, the trial function $\boldsymbol{u}$ and the test function $\delta\boldsymbol{u}$ are approximated by MLS schemes in the form Eq. (12). The final discrete system of equations is obtained by substituting the trial functions and test functions into the weak form Eq. (14), yielding the following system of linear algebraic equations:

$$\begin{bmatrix} \mathbf{K} & \mathbf{G} \\ \mathbf{G}^{\mathrm{T}} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \mathbf{u} \\ \boldsymbol{\lambda} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f} \\ \mathbf{q} \end{Bmatrix} \quad \textbf{(15)}$$

where the stiffness matrix **K** and the subvector **f** of the load vector **F** are given by:

$$\begin{aligned} \mathbf{K}_{ij} &= \int_{\Omega} \mathbf{B}_i^{\mathrm{T}}\mathbf{D}\mathbf{B}_j\,\mathrm{d}\Omega \\ \mathbf{f}_i &= \int_{\Gamma_t} \phi_i\bar{\mathbf{t}}\,\mathrm{d}\Gamma + \int_{\Omega} \phi_i\mathbf{b}\,\mathrm{d}\Omega \end{aligned} \quad \textbf{(16)}$$

where:

$$B_i = \begin{bmatrix} \phi_{i,x} & 0 \\ 0 & \phi_{i,y} \\ \phi_{i,y} & \phi_{i,x} \end{bmatrix} \qquad \mathbf{N}_k = \begin{bmatrix} N_k & 0 \\ 0 & N_k \end{bmatrix} \quad \textbf{(17)}$$

The terms of the matrix **G** and the subvector **q** of the load vector **F** are the contributions from the boundary conditions,

$$\begin{aligned} \mathbf{G}_{ik} &= -\int_{\Gamma} \phi_i\mathbf{N}_k\,\mathrm{d}\Gamma, \\ \mathbf{q}_k &= -\int_{\Gamma} \mathbf{N}_k\bar{\mathbf{u}}\,\mathrm{d}\Gamma \end{aligned} \quad \textbf{(18)}$$

where $\mathbf{N}_k(s)$ are (for instance) the conventional finite element shape functions, s is the arc length along the boundary and k is a node on the essential boundary.
Numerical quadrature needs to be performed to integrate the terms $\mathbf{K}_{ij}$ in the discrete equations obtained with global or local weak form (these integrals cannot simply be evaluated analytically). For this aim we employ the quasi-Monte Carlo integration which is the subject of the next section.

4. IMPLEMENTATION OF MONTE CARLO INTEGRATION IN EFG AND SPH

In this section we compare the results between given by these three meshless method for one-dimensional and two dimensional models.

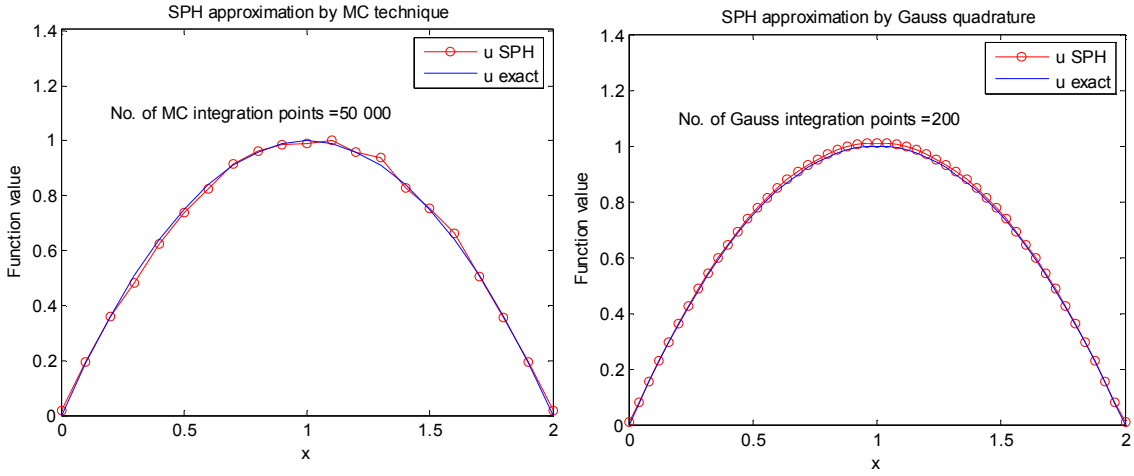

Figure 3: SPH approximation of $u(x) = 1 - x^2$. (a)using Monte Carlo approximation, for r_I /L = 0.02. (b) using Gauss quadrature for $r_I = 2 \times \Delta x_{max}$

For integration with Monte-Carlo techniques, we use a simple random generator; for Quasi-random Monte-Carlo technique we adopt Weyl-sequences and for Gauss, 4-th order quadrature rule are used.
Since Monte Carlo methods are statistical, for the integration with random Monte-Carlo, a great numbers of integrations points are necessary, and even that, the results do not show a good accuracy. The integration by quasi Monte Carlo techniques and Gauss quadrature give better results. We note the numbers of integrations points used by Gauss and QMC techniques are comparable: for Gauss rule, a total of 200 integration points is enough, while for QMC to achieve the same accuracy, 500 integration points are needed. In multidimensions, using Gauss quadrature, it is more difficult to come to grips with and need to increase the number of integration points, while in QMC has convergence rate independent of the dimension. This makes the use of QMC integration to be promising.
It should also be mentioned, concerning the *h*-adaptivity of the SPH, that SPH does not necessarily converge if the size of the smoothing length is kept proportional to the distance between nodes, r / r_I = constant [3]. That means that, the converge may fail for the standard refinement procedure of adding particle and simultaneously decreasing the support size.

Consider the implementation of the EFG method presented above for a linear elastostatic problem.
The governing equation:

$$\mathrm{E}\, u_{,xx} + x = 0 \qquad \textbf{(19)}$$

The bar has a constant cross sectional area of unit value, and modulus of elasticity E.
The displacement of the bar is fixed at the left end, and the right end is traction free:

$$u\big|_{x=0} = 0 \quad \text{and} \quad \mathrm{b}\frac{\mathrm{d}u}{\mathrm{d}x} = \bar{\mathbf{t}} = 0 \qquad \textbf{(20)}$$

The exact solution to the above problem is given by:

$$u(x) = \frac{1}{\mathrm{E}}\left[\frac{x}{2} - \frac{x^3}{6}\right] \qquad \textbf{(21)}$$

Figure 4 is a comparison of the EFG solution to the exact solution for both the displacements and strains along the bar. These results are obtained using eleven nodes along the length of the bar. The integration was performed with Weyl quasi-Monte-Carlo sequence using 1000 integration points.

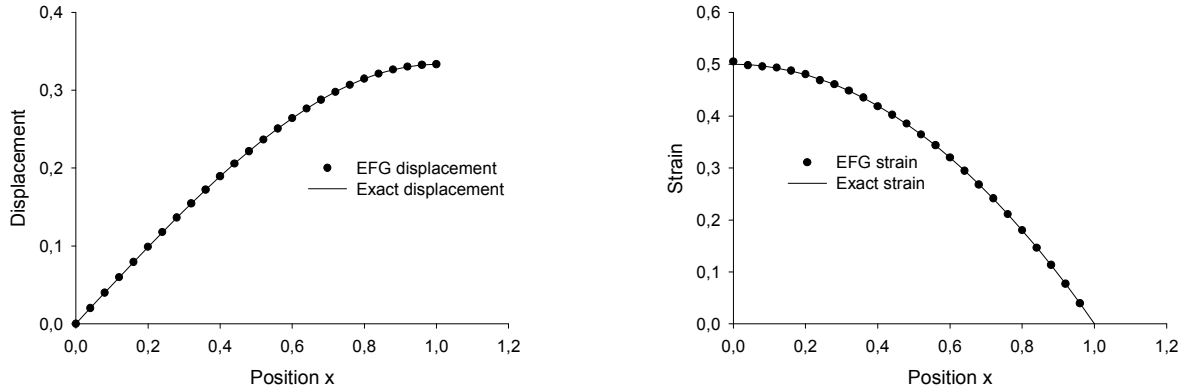

Figure 4: Comparison of EFG and exact results. a) displacements; b) strains for one-dimensional problem bar.

For EFG implementation we use Lagrange multipliers, for MFS modified variational principle and in MLPG we use penalty method. For EFG and MLPG methods we will use the cubic spline weight function and MLS shape functions for the approximation of trial function. In MFS we choose quartic spline weight function and PU basis functions based on Sheppard partitions of unity. In the numerical examples above, the polynomial (1, x) is used. The results were obtained using 5000 quasi-Monte-Carlo integration points. The results show good accuracy.

4. CONCLUSION

The construction of meshless approximations was reviewed and the effect of numerical integration errors on the solutions was presented. For arbitrary grids the meshless shape functions are rational functions with compact support in the domain. Hence, they are not integrated as accurately by Gauss quadrature. To overcome these difficulties, Quasi-Monte Carlo integration methods is proposed to perform the quadrature. The method is applicable to any type of meshless methods with any number of dimensions and has the advantage of not increasing the complexity even for the 3-D case. Here, we have presented a numerical test for a linear elasticity problem. Then a comparison with the analytical solution will be made of the effectiveness of the integration technique.

REFERENCES

[1] Atluri, S.N.; Kim, H.-G.; Cho, J.Y:. A Critical Assessment of the Truly Meshless Local Petrov-Galerkin and Local Boundary Integral Equation, Comput. Mech., 24, 348-372, 1999.
[2] Atluri SN, Zhu T.: A new meshless local Petrov-Galerkin (MLPG) approach in computational mechanics. Comput. Mech., 22, 117-127, 1998.
[3] Belytschko, T., Lu, Y.Y., Gu, L.: Element-free Galerkin Methods, International Journal for Numerical Methods in Engineering 37, 229–256, 1994.
[4] Dolbow, J., Belytschko, T.: Numerical Integration of the Galerkin Weak Form in Meshfree Methods, *Computational Mechanics* **23**, 219 – 230, 1999.
[5] Gingold, R.A., Monaghan, J.J.: Kernel Estimates as a Basis for General Particle Methods in Hydrodynamics, *Journal of Computational Physic* **46**, 429– 453, 1982.
[6] Gingold, R.A., Monaghan, J.J.: Kernel Estimates as a Basis for General Particle Methods in Hydrodynamics, *Journal of Computational Physic* **46**, 429– 453, 1982.
[7] Li, S., Lu, H., Han, W., Liu, W.K., Simkins, D.C. Reproducing kernel element method. Part II, Computer Methods in Applied Mechanics and Engineering 193, 953 – 987, 2004.
[8] Li, S., Liu, W.K.: Meshfree Particle Methods, Springer-Verlag, 2004.
[9] Lu, Y.Y., Belytschko, T., Gu, L.: A New Implementation of the Element Free Galerkin Method, *Computer Methods in Applied Mechanics and Engineering* **113**, 397–414, 1994.
[10] Lucy, L.B.: A Numerical Approach to Testing the Fission Hypothesis, *Astronomical Journal* **82**(12), 1013–1024, 1977.
[11] Monaghan JJ.: Smoothed particle hydrodynamics, *Annual Review of Astronomy and Astrophysics* **30**, 543–574, 1992.
[12] Monaghan, J.J.: Why particle methods work. *SIAM Journal of Scientific and Statistical Computing* **3**(4), 422-433, 1982.
[13] Monaghan, J.J.: Particle methods for hydrodynamics, *Computer Physics Reports* **3**, 71−124, 1985.