Scientia Iranica A (2014) 21(5), 1529–1538

SCIENTIA IRANICA

Sharif University of Technology
Scientia Iranica
Transactions A: Civil Engineering
www.scientiairanica.com

Adaptive node moving refinement in discrete least squares meshless method using charged system search

H. Arzani[a], A. Kaveh[b,*] and M. Dehghan[a]

a. *Department of Civil Engineering, Shahid Rajaee Teacher Training University, Tehran, P.O. Box 16785-136, Iran.*
b. *Centre of Excellence for Fundamental Studies in Structural Engineering, Iran University of Science and Technology, Narmak, Tehran, P.O. Box 16846-13114, Iran.*

Received 22 December 2012; received in revised form 11 October 2013; accepted 27 January 2014

KEYWORDS
Charged system search;
Discrete least squares;
Adaptive refinement;
Planar elasticity problems.

Abstract. Discrete Least Squares Meshless (DLSM) method has been used for the solution of different problems ranging from solid to fluid mechanics problems. In DLSM method the locations of discretization points are random. Therefore, the error of the initial solution is rather high. In this paper, an adaptive node moving refinement in DLSM method is presented using the Charged System Search (CSS) for optimum analysis of elasticity problems. The CSS algorithm is effectively utilized to obtain suitable locations of the nodes. The CSS is a multi-agent optimization technique based on some principles of physics and mechanics. Each agent, called a Charged Particle (CP), is a sphere with uniform charge density that can attract other CPs by considering the fitness of the CP. To demonstrate the effectiveness of the proposed method, some benchmark examples with available analytical solutions are used. The results show an excellent performance of the CSS for adaptive refinement in meshless method.

1. Introduction

Finite element method has been successfully used for the simulation of a large variety of practical engineering problems in the last decades. The method, however, involves some difficulties for certain processes such as creak propagation, extremely large deformation or implementation of adaptivity due to the need for mesh moving or remeshing of the domain. In the last decade several methods referred to as meshless method have been proposed and used to overcome these problems. Recently, a new meshless method which utilizes strong formulation of the governing differential equations, named Discrete Least Squares Meshless (DLSM) method [1], was proposed and used for the solution of Poisson equation. The advantages of this method over other numerical methods are no need for integration, simplicity of use and symmetric matrix of coefficients. The method has been used for the seepage problems [2], solution of solid mechanics problems [3] and many other cases, and its effectiveness has been proven. In all numerical methods such as finite element and meshless methods, adaptive refinement has become a standard procedure to achieve the desired accuracy by using a minimum number of nodes. An ideal computational algorithm with the re-meshing ability and pointing should be in such way that the density of nodes or new mesh increases in locations with higher computational errors. Formerly methods for an error estimation and adaptive refinement for hyperbolic problems in one-dimensional case [4], node enrichment adaptive method [5] in which more nodes are added in the regions with higher errors, and node moving strategy based on spring analogy [6] in DLSM method have been developed. These methods, however, have high computational costs due to the new

*. *Corresponding author. Tel.: +98 21 77240104; Fax: +98 21 77240398*
E-mail address: alikaveh@iust.ac.ir (A. Kaveh)

added nodes or finding new locations of the previous nodes.

To resolve these shortcomings, meta-heuristic algorithms can be used. Meta-heuristic algorithms are more suitable than traditional methods of optimizations and adaptivity due to their capability of exploring and finding promising regions in the search space in an affordable time and less computational costs. One of the best searching algorithms and more efficient than other optimizing algorithms is the Charged System Search (CSS). The algorithm that is based on electrostatics and Newtonian mechanics has been proposed by Kaveh and Talatahari [7]. The CSS has been used for a large variety of optimization problems such as optimal design of frame structures, optimum grillage system design, truss optimization and design optimization of reinforced concrete 3D structures [8-11]. The CSS for minimax and minisum facility layout problem [12] is one of the new researches which shows the applicability and robustness of the CSS for the problem of finding suitable locations. In this paper a node moving adaptive refinement strategy with the use of the CSS algorithm is presented for solving problems in solid mechanics. The process is very effective and because of the resolving the limitations of meshing and remeshing in meshless methods, the process is much more flexible and efficient than other adaptive refinement methods in FEM and meshless methods. Sections 2 and 3 present the fundamental concepts of DLSM method and CSS algorithm, respectively. In Section 4, error estimation and an adaptive refinement is exhibited. Section 5 illustrates the capabilities of the proposed method through some benchmark examples, where the results are compared with the available analytical solutions. Finally, some concluding remarks are addressed in Section 6.

2. Discrete Least Squares Meshless (DLSM)

2.1. Moving least squares shape functions

Among the available meshless approximation schemes, the Moving Least Squares (MLS) method [13] is generally considered to be one of the best methods to interpolate random data with a reasonable accuracy, because of its completeness, robustness and continuity [14,15]. With the MLS interpolation, the unknown function $\phi(x)$ is approximated by:

$$\phi(x) = \sum_{i=1}^{m} P_i(x).\alpha_i(x) \equiv \mathbf{P^T}(x).\boldsymbol{\alpha}(x). \tag{1}$$

Here, $P(x)$ is a polynomial basis in the space coordinates, and m is the total number of the terms in the basis. For a 2D problem we can specify $P(x) = [1\ x\ y\ x^2\ xy\ y^2]$ for $m = 6$. $\boldsymbol{\alpha}(x)$ is the vector of coefficients and can be obtained by minimizing a weighted discrete L_2 norm as:

$$J = \sum_{j=1}^{n} W(x - x_j).\left[\mathbf{P^T}(x_j).\boldsymbol{\alpha}(x) - \tilde{u}_j\right]^2, \tag{2}$$

where n is the number of nodes in the domain, and $\tilde{u}_j$ is the nodal value of the function to be approximated at point x_j. The weight function $W(x - x_j)$ is usually built in such a way that it has the following properties:

$W(x - x_j) > 0$ within the support domain,

$W(x - x_j) = 0$ outside the support domain,

$W(x - x_j)$ monotonically decreases from the point of interest at x, and

$W(x - x_j)$ is sufficiently smooth, especially on the boundary of Ω_j.

Here, for a better performance in meshless method, the cubic spline weight function is employed, given by:

$$W(x - x_j) = W(\bar{d}) = \begin{cases} \frac{2}{3} - 4\bar{d}^2 + 4\bar{d}^3 & \text{for } \bar{d} \le \frac{1}{2} \\ \frac{4}{3} - 4\bar{d} + 4\bar{d}^2 - \frac{4}{3}\bar{d}^3 & \text{for } \frac{1}{2} < \bar{d} \le 1 \\ 0 & \text{for } \bar{d} > 1 \end{cases} \tag{3}$$

where $\bar{d} = (x - x_j)/d_w$ and d_w is the size of influence domain of point x_j. Minimization of Eq. (2) with respect to the coefficient $\boldsymbol{\alpha}(x)$ leads to:

$$\phi(x) = \mathbf{P^T}(x)\mathbf{A}^{-1}(x)\mathbf{B}(x)\phi^h, \tag{4}$$

where:

$$\mathbf{A}(x) = \sum_{j=1}^{n} W(x - x_j)\mathbf{P}(x_j)\mathbf{P^T}(x_j), \tag{5}$$

and:

$$\mathbf{B}(x) = \Big[W(x - x_1)\mathbf{P}(x_1), W(x - x_2)\mathbf{P}(x_2), ..., W(x - x_n)\mathbf{P}(x_n)\Big]. \tag{6}$$

Eq. (4) can be written in the compact form:

$$\phi(x) = \sum_{i=1}^{n} N_i^T(x)\phi_i(x) = \mathbf{N^T}(x)\phi^h, \tag{7}$$

leading to the definition of MLS shape function defined as:

$$\mathbf{N^T}(x) = \mathbf{P^T}(x)\mathbf{A}^{-1}(x)\mathbf{B}(x), \tag{8}$$

where $\mathbf{N^T}(x)$ contains the shape functions of nodes at point x, which are called Moving Least Squares (MLS) shape functions.

2.2. Discrete least squares meshless method

Consider the following partial differential equation:

$$\begin{cases} \mathbf{L}(\phi) + f = 0 & \text{in } \Omega \\ \mathbf{B}(\phi) - \bar{t} = 0 & \text{in } \Gamma_t \\ \phi - \bar{\phi} = 0 & \text{in } \Gamma_u \end{cases} \tag{9}$$

where $\mathbf{L}$ and $\mathbf{B}$ are partial differential operators, $\bar{\phi}$ and $\bar{t}$ are vectors of prescribed displacements and tractions on the Dirichlet and Neumann boundaries, respectively; Ω is the considered domain; Γ_u and Γ_t are the displacement and traction boundaries, respectively; f is the vector of external force or source term on the problem domain.

Suppose the value of the estimating function ϕ in a point such as x_k is denoted as:

$$\phi(x_k) = \sum_{i=1}^{m} N_i(x_k)\phi_i. \tag{10}$$

According to the discretization of the problem domain and its boundaries, using Eq. (4), the residual of partial differential equation at point x_k is obtainable as:

$$R_\Omega(x_k) = L\left(\phi(x_k)\right) + f(x_k), \qquad k = 1, ..., M. \tag{11}$$

The residual of Neumann boundary condition at point x_k on the Neumann boundary can also be presented as:

$$R_t(x_k) = B\left(\phi(x_k)\right) - \bar{t}(x_k), \qquad k = 1, ..., M_t. \tag{12}$$

Finally the residual of Dirichlet boundary condition at point x_k on the Dirichlet boundary can be written as:

$$R_u(x_k) = \phi - \bar{\phi}(x_k), \qquad k = 1, ..., M_u, \tag{13}$$

where M_d is the number of internal points, M_t is the number of points on the Neumann boundary, M_u is the number of points on the Dirichlet boundary, and M is the total number of points. A penalty approach can now be used to form the least squares functional of the residuals defined as:

$$J = \frac{1}{2}\left[\sum_{k=1}^{M_d} R_\Omega^2(x_k) + \alpha.\sum_{k=1}^{M_t} R_t^2(x_k) + \beta.\sum_{k=1}^{M_u} R_u^2(x_k)\right], \tag{14}$$

where α and β are the penalty coefficients for the importance of Neumann and Dirichlet boundary conditions, respectively. Minimization of the functional with respect to nodal parameters $(\phi_i, i = 1, 2, ..., n)$ leads to the system of equations:

$$\mathbf{K}\phi = \mathbf{F}, \tag{15}$$

where:

$$K_{ij} = \sum_{k=1}^{M} \left[L(N)\right]_k^T \left[L(N)\right]_k + \alpha \sum_{k=1}^{M_t} \left[B(N)\right]_k^T \left[B(N)\right]_k + \beta \sum_{k=1}^{M_u} [N]_k^T \bar{\phi}_k, \tag{16}$$

$$F_i = -\sum_{k=1}^{M} \left[L(N)\right]_k^T \left[L(N)\right] f_k + \alpha \sum_{k=1}^{M_t} \left[B(N)\right]_k^T \bar{t}_k + \beta \sum_{k=1}^{M_u} [N]_k^T \bar{\phi}_k. \tag{17}$$

The stiffness matrix $\mathbf{K}$ in Eq. (16) is square, symmetric, and positive definite. Therefore the final system of equations can be solved directly via efficient solvers.

3. Charged system search

The charged system search is based on electrostatics and Newtonian mechanics laws. The Coulomb and Gauss laws provide the magnitude of the electric field at a point inside and outside a charged insulating solid sphere, respectively, as [16]:

$$E_{ij} = \begin{cases} \frac{k_e q_i}{a^3} r_{ij} & \text{if } r_{ij} < a \\ \frac{k_e q_i}{r_{ij}^2} & \text{if } r_{ij} \geq a \end{cases} \tag{18}$$

where k_e is a constant known as the Coulomb constant, r_{ij} is the separation of the centre of sphere and the selected point, q_i is the magnitude of the charge, and a is the radius of the charged sphere. Using the principle of superposition, the resulting electric force due to N charged spheres is equal to [7]:

$$\mathbf{F}_j = k_e q_j \sum_{i=1}^{N} \left(\frac{q_i}{a^3} r_{ij}.i_1 + \frac{q_i}{r_{ij}^2}.i_2\right) \frac{\mathbf{r_i} - \mathbf{r_j}}{\mathbf{r_i} - \mathbf{r_j}},$$

$$\begin{cases} i_1 = 1, \ i_2 = 0 \iff r_{ij} < a \\ i_1 = 0, \ i_2 = 1 \iff r_{ij} \leq a \end{cases} \tag{19}$$

Also, according to Newtonian mechanics, we have [16]:

$$\mathbf{\Delta r} = \mathbf{r}_{\text{new}} - \mathbf{r}_{\text{old}}, \tag{20}$$

$$\mathbf{V} = \frac{\mathbf{r}_{\text{new}} - \mathbf{r}_{\text{old}}}{\Delta t}, \tag{21}$$

$$\mathbf{a} = \frac{\mathbf{V}_{\text{new}} - \mathbf{V}_{\text{old}}}{\Delta t}, \tag{22}$$

where $\mathbf{r}_{\text{old}}$ and $\mathbf{r}_{\text{new}}$ are the initial and final positions of a particle, respectively, $\mathbf{v}$ is the velocity of the particle, and a is the acceleration of the particle. Combining the above equations and using Newton's second law, the displacement of any object as a function of time is obtained as:

$$\mathbf{r}_{\text{new}} = \frac{1}{2}\frac{\mathbf{F}}{m}.\Delta t^2 + \mathbf{v}_{\text{old}}.\Delta t + \mathbf{r}_{\text{old}}. \tag{23}$$

Inspired by the above electrostatics and Newtonian mechanics laws, the pseudo-code of the CSS algorithm is presented as follows [7].

Level 1: Initialization

Step 1. Initialization. Initialize the parameters of the CSS algorithm. Initialize an array of Charged Particles (CPs) with random positions. The initial velocities of the CPs are taken as zero. Each CP has a charge of magnitude defined considering the quality of its solution as:

$$q_i = \frac{\text{fit}(i) - \text{fitworst}}{\text{fitbest} - \text{fitworst}}, \quad i = 1, 2, ..., N, \tag{24}$$

where fitbest and fitworst are the best and the worst fitness of all the particles, and fit(i) represents the fitness of agent i. The separation distance r_{ij} between two charged particles is defined as:

$$r_{ij} = \frac{\mathbf{X}_i - \mathbf{X}_j}{(\mathbf{X}_i + \mathbf{X}_j)/2 - \mathbf{X}_{\text{best}} + \varepsilon}, \tag{25}$$

where X_i and X_j are the positions of the ith and jth CPs, respectively, X_{best} is the position of the best current CP, and ε is a small positive number to avoid singularities.

Step 2. CP ranking. Evaluate the values of the fitness function for the CPs, compare with each other and sort them in increasing order.

Step 3. CM creation. Store the number of the first CPs equal to Charged Memory Size (CMS) and their related values of the fitness functions in the Charged Memory (CM).

Level 2: Search

Step 1. Attracting force determination.
Determine the probability of moving each CP toward the others considering the probability function:

$$p_{ij} = \begin{cases} 1 & \frac{\text{fit}(i) - \text{fitbest}}{\text{fit}(i) - \text{fit}(j)} > \text{rand} \lor \text{fit}(j) > \text{fit}(i) \\ 0 & \text{else} \end{cases} \tag{26}$$

and calculate the attracting force vector for each CP as:

$$\mathbf{F}_j = q_j \sum_{i,i\neq j}^{N} \left(\frac{q_i}{a^3} r_{ij}.i_1 + \frac{q_i}{r_{ij}^2}.i_2 \right) p_{ij} (\mathbf{X}_i - \mathbf{X}_j)$$

$$\begin{cases} j = 1, 2, ..., N \\ i_1 = 1, \ i_2 = 0 \iff r_{ij} < a \\ i_1 = 0, \ i_2 = 1 \iff r_{ij} \geq a \end{cases} \tag{27}$$

where F_j is the resultant force affecting the jth CP.

Step 2. Solution construction. Move each CP to the new position and find its velocity using the following equations:

$$\mathbf{X}_{j,\text{new}} = \text{rand}_{j1}.k_a.\frac{\mathbf{F}_j}{m_j}.\Delta t^2 + \text{rand}_{j2}.k_v.\mathbf{V}_{j,\text{old}}.\Delta t + \mathbf{X}_{j,\text{old}}, \tag{28}$$

$$\mathbf{V}_{j,\text{new}} = \frac{\mathbf{X}_{j,\text{new}} - \mathbf{X}_{j,\text{old}}}{\Delta t}, \tag{29}$$

where rand$_{j1}$ and rand$_{j2}$ are two random numbers uniformly distributed in the range (0,1), Δt is the time step, and it is set to 1. k_a is the acceleration coefficient, k_v is the velocity coefficient to control the influence of the previous velocity. Here, k_a and k_v are taken as 0.5. Also, m_j is the mass of the CPs, which is equal to q_j in this paper. The mass concept may be useful for developing a multi-objective CSS.

Step 3. CP position correction. If each CP exits from the allowable search space, correct its position using the HS-based handling approach as described for the HPSACO algorithm [17,18].

Step 4. CP ranking. Evaluate and compare the values of the fitness function for the new CPs, and sort them in an increasing order.

Step 5 CM updating. If some new CP vectors are better than the worst ones in the CM, in terms of their objective function values, include the better vectors in the CM and exclude the worst ones from the CM.

Level 3: Controlling the terminating criterion
Repeat the search level steps until a terminating criterion is satisfied.

4. Error indicator and adaptive refinement

Adaptivity is an important tool for the efficiency and effectiveness of any numerical method. Any adaptive procedure is formed of two main parts, error estimation and mesh refinement. For any successful adaptive procedure, an actual error estimator is essential. Different methods to estimate error have been introduced and applied in the numerical methods. These methods can be separated into two categories. Methods based on residual of the differential equations governing on the problem [19,20] and the methods based on the restoration of the error which considered the error as the gradient of the solution [21,22]. In this study, error estimation based on least squares function formed from weighted residuals is used [23], where the error for each point is defined as:

$$e_k = \sqrt{\frac{1}{2}\left[R_\Omega^2(x_k) + \alpha.R_t^2(x_k) + \beta.R_u^2(x_k)\right]}. \qquad (30)$$

Here, e_k is the error of any point in the problem domain or its boundaries. The advantage of this choice is the availability of the error estimator in the process of the main simulation. Different methods for refinement and achieving more accurate solutions after identifying the error distribution can be used. There are three general methods of refinement in finite element: mesh moving (r-method), mesh enrichment (p-method) and p-refinement whereby higher order shape functions are used. In mesh moving methods the number of the nodes is constant but the location of the nodes is altering according to the achieved errors. In FEM the connectivity of the nodes may be disturbed and some of the elements can overlap or be zero area. Therefore there is some limitation in FEM. In mesh enrichment process, which is the most common method in FEM, the initial mesh remains and some new elements are added to the domains with higher amount of errors, or a new mesh is created based on the error distribution. In increasing process of the order of shape functions, the order increases in the domain with higher errors. This process needs the use of hierarchical shape functions which is associated with complications. It is clear that the most appropriate adaptivity is mesh moving because of the constant number of the nodes, therefore the computational cost is less than the other methods. The use of this method has some complexity in FEM due to the deformation of the element after removing them. Since in meshless methods and especially in DLSM method there is no element scheme and the solution is not sensitive to the distance between nodes and the method of exposure, the use of the node moving methods is easily possible.

5. Relationship of the CSS and adaptivity

For the CSS algorithm, each node is used as a CP of zero radiuses and m_j is employed for masses. For each node, the error based on least squares function is formed from weighted residuals which is available in each analysis using Eq. (30). This error considered as the charge of each node is normalized as:

$$q_j = \frac{e_j - e_{\min}}{e_{\max} - e_{\min}}. \qquad (31)$$

The distance between any two nodes, j and i, is defined by L_P norm, and in particular for planar problem L_2 norm is used, as follows:

$$r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}. \qquad (32)$$

Considering that the error reduction in meshless methods requires the densities of the nodes in the accumulation zones of the errors, therefore only nodes with less error are allowed to move toward the nodes with high error:

$$p_{ij} = \begin{cases} 1 & q_1 \geq q_j \\ 0 & \text{else} \end{cases} \qquad (33)$$

For each node, the force vectors in two directions are calculated with the following equations:

$$F_{j_x} = k_e q_j \sum_{i=1, i\neq j}^{N} \left(\frac{q_i}{r_{ij}^2}\right) \frac{x_i - x_j}{x_i - x_j} p_{ij} \cos(\theta), \qquad (34)$$

$$F_{j_y} = k_e q_j \sum_{i=1, i\neq j}^{N} \left(\frac{q_i}{r_{ij}^2}\right) \frac{y_i - y_j}{y_i - y_j} p_{ij} \sin(\theta). \qquad (35)$$

Here, θ is the angle between horizontal direction and the line connecting the two nodes. According to the formulation of the CSS, new positions of the nodes in two-dimensional case are obtained similarly.

5.1. Objective function

In the DLSM method, the value of residuals represents the scope to which the numerical solution satisfies the governing differential equation and its boundary conditions. Also it is mentioned that the least-squares functional defined as the squares residual can be considered as a measure of the error of the numerical solution. The method is especially efficient since the least-squares calculations are already available from the solution procedure. Hence, the objective function to be minimized is taken as the normalized sum of squared residuals of all points in the domain. The search level of the CSS is continued as long as improvement in the reduction of the objective function is possible. The algorithms are coded in compact visual FORTRAN and the systems of linear equations are solved with efficient solver.

5.2. Selected parameters

In the first examples the nodes in Arc edge and corner nodes have been fixed. The nodes on the horizontal boundaries can only move horizontally and the movement of the nodes on the vertical boundary is limited vertically. Δt is set to 1. k_a and k_v are taken as 0.5. For all nodes, m_j is selected as constant. The best value for m_j is found to be 20. The objective function is taken as the ratio of the least squares functional of the residuals to the total number of points which is available in the process of the main simulation as:

$$\text{Minimize } E^{\text{Total}}_{\text{Estimate}} = \left(\frac{\sum_{k=1}^{M_d} R^2_\Omega(x_k) + \alpha . \sum_{k=1}^{M_t} R^2_t(x_k) + \beta . \sum_{k=1}^{M_u} R^2_u(x_k)}{M}\right). \quad (36)$$

Here, M_d is the number of internal points, M_t is the number of points on the Neumann boundary, M_u is the number of points on the Dirichlet boundary, and M is the total number of points. In the second example, the corner nodes are fixed and boundary nodes can only move on their directions. Similar to the first example, Δt is set to 1. k_a and k_v are taken as 0.5 and the finest value for m_j, which leads to the best results in one iteration, is taken as 52.

6. Numerical examples

In this section, two examples of planar elasticity are presented. These examples are selected because their analytical solutions are available. The comparison of the initial solution and the solution after refinement by the CSS with analytical solution shows the efficiency of the presented method. The first example considers an infinite plate with a circular hole subjected to a uniaxial traction, and the second example is a cantilever beam under end load.

6.1. Infinite plate with a circular hole

The first example considers an infinite plate with a circular hole under a uniaxial load at infinity, as shown with boundary conditions in Figures 1 and 2. The exact solutions of this problem for the stress components can be defined as [24]:

$$\sigma_x = t\left\{1 - \frac{a^2}{r^2}\left[\frac{3}{2}\cos(2\theta) + \cos(4\theta)\right] + \frac{3a^4}{2r^2}\cos(4\theta)\right\}, \quad (37)$$

$$\sigma_y = -t\left\{\frac{a^2}{r^2}\left[\frac{1}{2}\cos(2\theta) + \cos(4\theta)\right] + \frac{3a^4}{2r^2}\cos(4\theta)\right\}, \quad (38)$$

$$\tau_{xy} = -t\left\{\frac{a^2}{r^2}\left[\frac{1}{2}\cos(2\theta) + \sin(4\theta)\right] - \frac{3a^4}{2r^2}\sin(4\theta)\right\}, \quad (39)$$

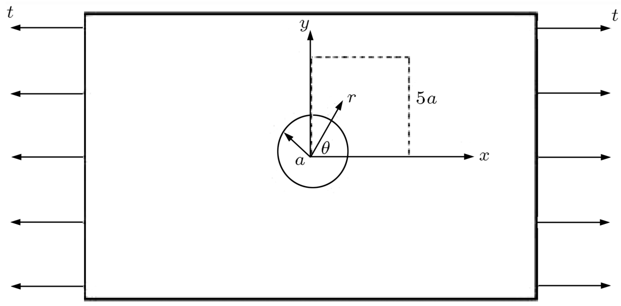

Figure 1. An infinite plate with a circular hole under a uniaxial load P.

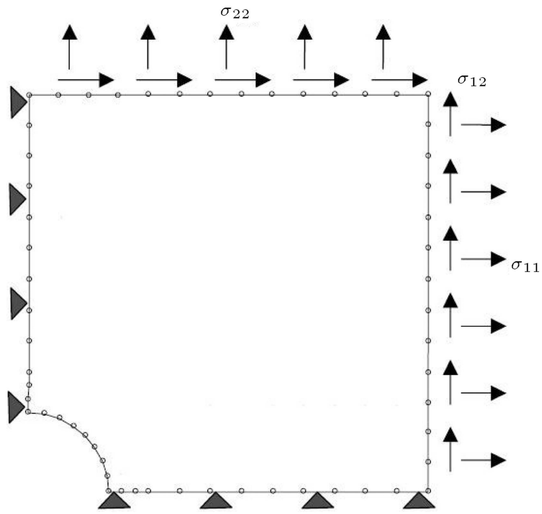

Figure 2. Boundary conditions of the plate.

$$u_r = \frac{t}{4G}\left\{r\left[\frac{k-1}{2} + \cos(2\theta)\right] + \frac{a^2}{r}\left[1 + (1+k)\cos(2\theta)\right] - \frac{a^4}{r^3}\cos(2\theta)\right\}, \quad (40)$$

$$u_\theta = \frac{t}{4G}\left\{(1-k)\frac{a^2}{r} - r - \frac{a^4}{r^3}\right\}\sin(2\theta). \quad (41)$$

In the above equations, G is the shear modulus and $k = (3-v)/(1+v)$ with v representing the Poisson's ratio. Due to symmetry, only the upper right square quadrant of the plate is modeled (see Figure 2). The edge length of the square is $5a$, with a being the radius of the circular hole. The Dirichlet boundary condition is imposed on the left and bottom boundaries, and the tractions are applied to the top and right edges. The problem is solved under a plane stress conditions. The process begins with the simulation of the problem on two distributions of 95 nodes referred to as initial and refined distributions shown in Figures 3 and 4, respectively. The error distribution is estimated based on the numerical solution obtained using Eq. (14).

After running the CSS algorithm with a uniform distribution of the nodes, the adapted nodes are more

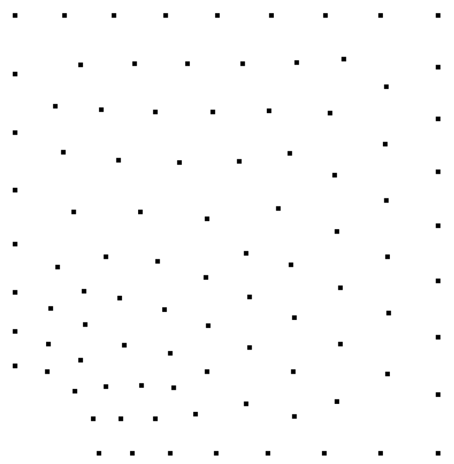

Figure 3. The initial distribution of the nodes.

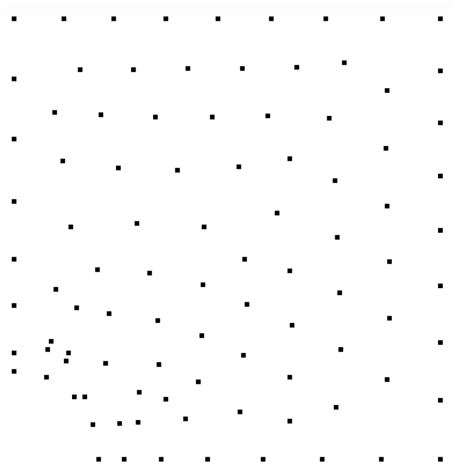

Figure 4. The refined distribution by the CSS.

concentrated around the curved edge where the numerical errors are much higher due to stress concentration. Comparing the horizontal displacement, u_x, of the hole and the normal stress, σ_x, along $x = 0$, with analytical solution of the problem, a tremendous evolution can be seen in the responsiveness of DLSM method by the use of the CSS algorithm, as shown in Figures 5 and 6. Stress tensor for initial solution, the solution of the refined distribution of the nodes by CSS and the real stress tensor for analytical solution are shown in Figure 7.

6.2. A cantilever beam under end load

As a second example, the problem of a cantilever beam under a point load at the end, as shown in Figure 8, is considered. For this problem, the exact stresses and displacements in plane stress are given by Timoshenko and Goodier [24] as:

$$\sigma_x = -\frac{P(L-x)y}{I}, \tag{42}$$

$$\sigma_y = 0, \tag{43}$$

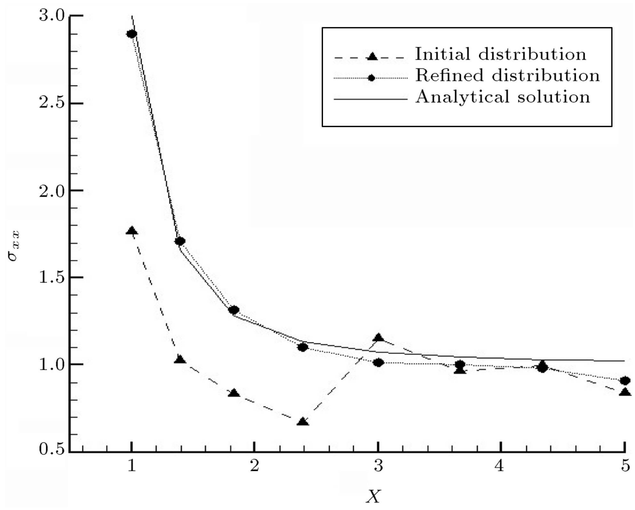

Figure 5. Normal stress σ_x along $x = 0$.

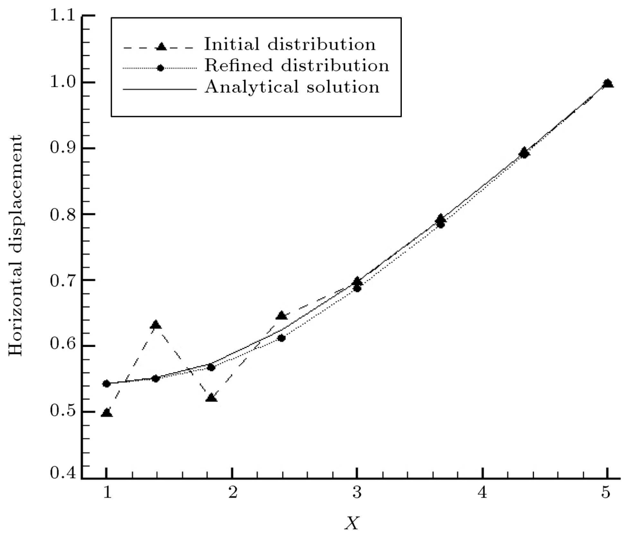

Figure 6. The horizontal displacement u_x of hole.

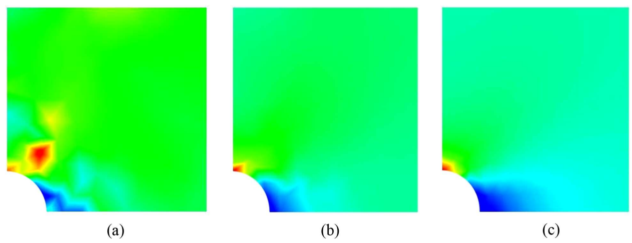

Figure 7. Contours of the normal stress σ_x: (a) Initial solution; (b) refined solution; and (c) exact solution.

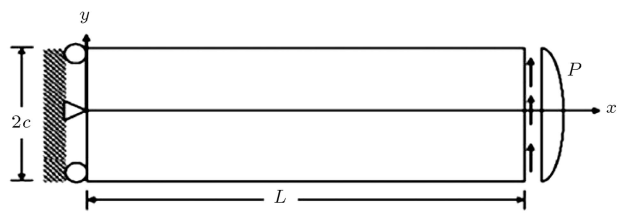

Figure 8. A cantilever beam under a point load at the end.

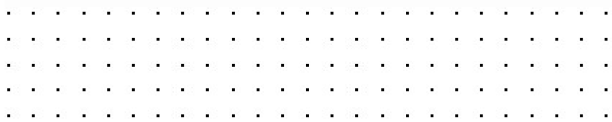

Figure 9. Initial nodal distribution.

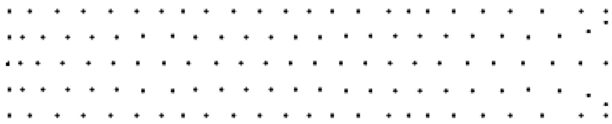

Figure 10. Refined nodal distribution by the CSS.

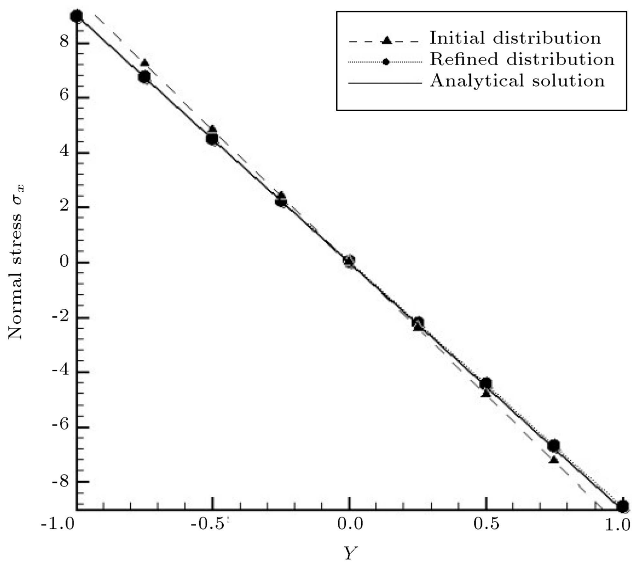

Figure 11. The vertical displacement u_y along upper surface of the beam.

$$\tau_{xy} = \frac{P}{2I}[c^2 - y^2], \tag{44}$$

$$u = -\frac{Py}{6EI}\left[3x(2L - x) + (2 + v)(y^2 - c^2)\right], \tag{45}$$

$$v = \frac{Py}{6EI}\left[x^2(3L - x) + 3v(L - x)y^2 + (4 + 5v)c^2x\right], \tag{46}$$

where E is the elastic modulus and v represents Poisson's ratio. The moment of inertia $I = 2c^3/3$ is considered for a beam with rectangular cross-section and unit thickness.

The problem is solved using the DLSM method under plane stress conditions and on the initial configurations. A distribution of 125 nodes is used as shown in Figure 9. The errors are calculated and the nodal locations are adaptively altered using the CSS algorithm, as shown in Figure 10. The problem is then resolved on the refined nodal configuration and the results are compared to those obtained on the initial configuration and to the exact analytical results. Figures 11 and 12 compare the vertical displacement,

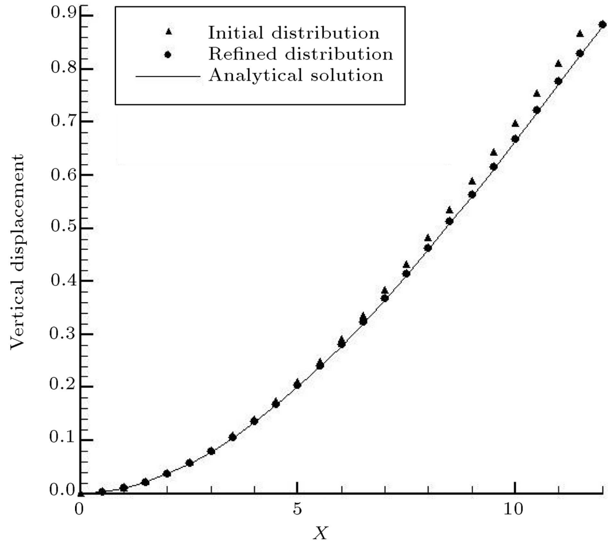

Figure 12. Normal stress σ_x along upper surface of the beam.

u_y, along upper surface of the beam, and the normal stress σ_x along upper surface of the beam obtained on the initial and adapted distributions with the analytical solutions. It can be seen that the result obtained on the adapted nodal distribution is virtually exact indicating on the effectiveness of the CSS algorithm in DLSM method.

For large-scale and complex problems the use of meta-heuristic approaches becomes time consuming; however, for such a problem one can decompose it into its components and after solution of each component, the solution of the main problem can be obtained by using the methods provided in Ref. [25].

7. Conclusion

Though the DLSM method has been developed in recent years in order to achieve more accurate solution, refinement is inevitable like the other numerical methods. The process of the refinement proposed in this paper makes it possible to achieve more accurate responses without increasing the number of points and imposing computational costs. Considering the points in the discretized domain as charged particles in the CSS algorithm and by using the normalized errors in DLSM process as the charges of particles, a powerful model is created. Therefore a node moving in the direction of optimal solution of the problem is obtained. In lectures on adaptivity in meshless methods, different techniques are introduced such as that [6] which uses springs between two adjacent nodes, determining the new locations of the nodes requires the solution of a large truss with a computational cost almost as much as time required for solution of the problem. However, since the CSS searches the most suitable locations of the nodes in one or two iteration, its computational cost is much less than the existing algorithms.

Acknowledgement

The second author is grateful to the Iran National Science Foundation for the support.

References

1. Afshar, M.H. and Arzani, H. "Solving Poissons equations by the discrete least squares meshless method", *WIT Transactions on Modeling and Simulation*, **42**, pp. 23-32 (2004).
2. Firoozjaee, A.R. and Afshar, M.H. "Discrete least squares meshless method with sampling points for the solution of elliptic partial differential equations", *Engineering Analysis with Boundary Elements*, **33**, pp. 83-92 (2009).
3. Afshar, M.H., Lashckarbolok, M. "Collocated discrete least squares (CDLS) meshless method: Error estimate and adaptive refinement", *International Journal for Numerical Methods in Fluids*, **56**, pp. 1909-1928 (2008).
4. Naisipour, M., Afshar, M.H., Hassani, B. and Firoozjaee, A.R. "Collocation discrete least squares (CDLS) method for elasticity problems and grid irregularity effect assessment", *American Journal of Applied Sciences*, **5**, pp. 1595-1601 (2008).
5. Afshar, M.H., Amani, J. and Naisipour, M. "A node enrichment adaptive refinement in discrete least squares meshless method for solution of elasticity problems", *Engineering Analysis with Boundary Elements*, **36**, pp. 385-393 (2012).
6. Afshar, M.H., Amani, J. and Naisipour, M. "Node moving adaptive refinement strategy for planar elasticity problems using discrete least squares meshless method", *Finite Elements in Analysis and Design*, **47**, pp. 1315-1325 (2011).
7. Kaveh, A. and Talatahari, S. "A novel heuristic optimization method: Charged system search", *Acta Mechanica*, **213**(3-4), pp. 267-286 (2010).
8. Kaveh, A. and Talatahari, S. "Charged system search for optimal design of frame structures", *Applied Soft Computing*, **12**, pp. 382-393 (2012).
9. Kaveh, A. and Talatahari, S. "Charged system search for optimum grillage systems design using the LRFD-AISC code", *Journal of Constructional Steel Research*, **66**(6), pp. 767-771 (2010).
10. Kaveh, A. and Zolghadr, A. "Truss optimization with natural frequency constraints using a hybridized CSS-BBBC algorithm with trap recognition capability", *Computers and Structures*, **102-103**, pp. 14-27 (2012).
11. Kaveh, A. and Behnam, A.F. "Design optimization of reinforced concrete 3D structures considering frequency constraints via a charged system search", *Scientia Iranica*, **20**(3), pp. 387-396 (2013).
12. Kaveh, A. and Sharafi, P. "Charged system search algorithm for minimax and minisum facility layout problems", *Asian Journal of Civil Engineering (Building and Housing)*, **12**(6), pp. 703-718 (2011).
13. Lancaster, P. and Salkauskas, K. "Surfaces generated by moving least squares method", *Mathematics of Computation*, **37**, pp. 141-158 (1981).
14. Onate, E., Perazzo, F. and Miquel, J. "A finite point method for elasticity problems", *Computers and Structures*, **79**, pp. 2151-2163 (2001).
15. Atluri, S.N. *The Meshless Local Petrov-Galerkin (MLPG) Method for Domain & Boundary Discretizations*, Tech Science Press (2004).
16. Halliday, D., Resnick, R. and Walker, J., *Fundamentals of Physics*, 8th Ed. John Wiley and Sons (2008).
17. Kaveh, A. and Talatahari, S. "Particle swarm optimizer, ant colony strategy and harmony search scheme hybridized for optimization of truss structures", *Computers and Structures*, **87**(5-6), pp. 267-283 (2009).
18. Kaveh, A. and Talatahari, S. "A particle swarm ant colony optimization algorithm for truss structures with discrete variables", *Journal of Constructional Steel Research*, **65**(8-9), pp. 1558-1568 (2009).
19. Babuska, I. and Rheinboldt, W.C. "A posteriori error estimates for the finite element method", *International Journal for Numerical Methods in Engineering*, **12**, pp. 1597-1615 (1978).
20. Babuska, I. "The self adaptive approach in the finite element method", in: J.R. White man (Ed.), *Mathematics of Finite Elements and Applications, Academic Press*, London, pp. 125-142 (1975).
21. Zienkiewicz, O.C. and Zhu, J.Z. "The super convergent patch recovery and a posteriori error estimate. Part 1: The recovery technique", *International Journal for Numerical Methods in Engineering*, **33**, pp. 1331-1364 (1995).
22. Zienkiewicz, O.C. and Zhu, J.Z. "The super convergent patch recovery and a posteriori error estimate. Part 2: Error estimate and adaptivity", *International Journal for Numerical Methods in Engineering*, **33**, pp. 1365-1382 (1995).
23. George, P.L. and Borouchaki, H. *Delaunay Triangulation and Meshing: Application to Finite Elements*, HERMES, Paris (1998).
24. Timoshenko, S.P. and Goodier, J.N. *Theory of Elasticity*, 3rd Ed., McGraw-Hill, New York (1987).
25. Kaveh, A. *Optimal Analysis of Structures by Concepts of Symmetry and Regularity*, Springer Verlag, GmbH, Wien, NewYork (2013).

Biographies

Hamed Arzani was born in 1969 in Amol, Iran. He obtained his BS, MS and PhD degrees in Civil Engineering, from University of Science and Technology in 1992, 1994, 2006, respectively. He then joined the University of Shahid Rajaee Teacher Training University in Tehran where he is presently Assistant Professor of Structural Engineering. He is the author of 2 papers published in international journals and 5 papers presented at international conferences.

Ali Kaveh was born in 1948 in Tabriz, Iran. After graduation from the Department of Civil Engineering at the University of Tabriz in 1969, he continued his studies on Structures at Imperial College of Science and Technology at London University, and received his MS, DIC and PhD degrees in 1970 and 1974, respectively. He then joined the Iran University of Science and Technology in Tehran where he is presently Professor of Structural Engineering. Professor Kaveh is the author of 300 papers published in international journals and 135 papers presented at international conferences. He has authored 23 books in Farsi and 7 books in English published by Wiley, the American Mechanical Society, Research Studies Press and Springer.

Mohammad Dehghan was born in 1979 in Doroud, Iran. He obtained his B.S. degree in mathematics at Shahid Chamran University of Ahvaz in 2001, his M.S. degree in mathematical analysis from Lorestan University in 2006, and his second M.S. degree in Structural Engineering from Shahid Rajaee Teacher Training University in 2012. At present, he works on optimum design of different structures considering dynamic loadings and irregularities in plan via the meta-heuristic methods.