Applied Mathematical Modelling 39 (2015) 7181–7196

Contents lists available at ScienceDirect

Applied Mathematical Modelling

journal homepage: www.elsevier.com/locate/apm

A new low-cost meshfree method for two and three dimensional problems in elasticity

Davoud Mirzaei

Department of Mathematics, University of Isfahan, 81745-163 Isfahan, Iran

ARTICLE INFO

Article history:
Received 6 May 2014
Received in revised form 22 January 2015
Accepted 25 February 2015
Available online 17 March 2015

Keywords:
DMLPG methods
MLPG methods
MLS approximation
GMLS approximation
Direct approximation
Elasto-static

ABSTRACT

In this paper, we continue the development of the *Direct Meshless Local Petrov–Galerkin (DMLPG)* method for elasto-static problems. This method is based on the generalized moving least squares approximation. The computational efficiency is the most significant advantage of the new method in comparison with the original MLPG. Although, the "Petrov–Galerkin" strategy is used to build the primary local weak forms, the role of trial space is ignored and *direct approximations* for local weak forms and boundary conditions are performed to construct the final stiffness matrix. In this modification the numerical integrations are performed over polynomials instead of complicated MLS shape functions. In this paper, DMLPG is applied for two and three dimensional problems in elasticity. Some variations of the new method are developed and their efficiencies are reported. Finally, we will conclude that DMLPG can replace the original MLPG in many situations.

1. Introduction

The *Meshless Local Petrov–Galerkin (MLPG)* method has been widely employed to find the numerical solutions of elasto-static and elasto-dynamic problems. MLPG was first introduced in [1], and was first applied to elasticity in [2]. Afterward, many papers were appeared for different types of mechanical problems. For example see [3,4] and the recent review paper [5]. MLPG is based on *local weak forms* and it is known as a *truly* meshless method, because it uses no global background mesh to evaluate integrals, and everything breaks down to some regular, well-shaped and independent sub-domains. This is in contrast with methods which are based on *global weak forms*, such as the Element-free Galerkin (EFG) method [6], where triangulation is again required for numerical integration. *But* MLPG still suffers from the cost of numerical integration. This is due to the complexity of the integrands. In MLPG and all MLS based methods, integrations are done over complicated MLS shape functions, and this leads to high computational costs in comparison with the finite elements method (FEM), where integrands are simple and close form polynomials. Thus, special cares should be taken in performing numerical quadratures for meshfree methods. These challenges have been addressed in various engineering papers [7–12] and several approaches to implement numerical integration have been proposed in the literature. A brief review of these approaches is presented in Section 3 of [13].

This is the reason why this method, and of course the other meshfree methods, have found very limited application to three-dimensional problems, which are routine applications of FEM.

A tricky modification has been applied to MLPG in [14], in which the numerical integrations are done over low-degree polynomial basis functions rather than complicated MLS shape functions. In addition, as the shapes of the local sub-domains

E-mail address: d.mirzaei@sci.ui.ac.ir

http://dx.doi.org/10.1016/j.apm.2015.02.050
0307-904X/

remain unchanged, the values of integrals remain the same. This reduces the computational costs of MLPG, significantly. In the new method, local weak forms are considered as functionals and they are *directly* approximated from nodal data using a generalized moving least squares (GMLS) approximation. Thus this method is called Direct MLPG (DMLPG). Although DMLPG uses the same local forms, it is theoretically different from MLPG, because it eliminates the role of trial space. DMLPG can be considered as a generalized finite difference method (GFDM), not only in its usual strong form, but also in a weak formulation. It is worthy to note that, by this modification we do not lose the order of convergence. This has been analytically proven in [15,16] for different definitions of functionals, specially for the local weak forms of DMLPG.

DMLPG has been applied to the heat conduction problem in [17] and has been numerically investigated for 2D and 3D potential problems in [18].

In this paper, the application of DMLPG is provided for elasto-static problems for the first time. We consider both two and three dimensional problems to show the efficiency of the new method. The method can be easily extended to the other problems in elasticity.

2. Generalized moving least squares

Generalized moving least squares (GMLS) approximation was presented in [15] in details. Here we briefly discuss this concept. Let Ω be a bounded subset in $\mathbb{R}^d$, $d \in \mathbb{Z}_+$, and $X = \{x_1, x_2, \dots, x_N\} \subset \Omega$ be a set of meshless points scattered (with certain quality) over Ω. The MLS method approximates the function $u \in U$ (with certain smoothness) by its values at points x_j, $j = 1, \dots, N$, by

$$u(x) \approx \widehat{u}(x) = \sum_{j=1}^{N} a_j(x) u(x_j), \quad x \in \Omega, \tag{2.1}$$

where $a_j(x)$ are MLS shape functions obtained in such way that $\widehat{u}(x)$ be the best approximation of $u(x)$ in polynomial subspace $\mathbb{P}_m(\mathbb{R}^d) = \text{span}\{p_1, \dots, p_Q\}, Q = \binom{m+d}{d}$, with respect to a weighted, discrete and *moving* ℓ^2 norm. The weight function governs the influence of the data points and assumed to be a function $w : \Omega \times \Omega \to \mathbb{R}$ which becomes smaller the further away its arguments are from each other. Ideally, w vanishes for arguments $x, y \in \Omega$ with $\|x - y\|_2$ greater than a certain threshold, say δ. Such a behavior can be modeled by using a translation-invariant weight function. This means that w is of the form $w(x, y) = \varphi(\|x - y\|_2/\delta)$ where φ is a compactly supported function supported in $[0, 1]$. If we define

$$\begin{aligned} P &= P(x) = \big(p_k(x_j)\big) \in \mathbb{R}^{N \times Q}, \\ W &= W(x) = \text{diag}\{w(x_j, x)\} \in \mathbb{R}^{N \times N}, \end{aligned} \tag{2.2}$$

then a simple calculation gives the *shape functions*

$$\boldsymbol{a}(x) := [a_1(x), \dots, a_N(x)] = \boldsymbol{p}(x)(P^T W P)^{-1} P^T W, \tag{2.3}$$

where $\boldsymbol{p} = [p_1, \dots, p_Q]$. If $X_x = \{x_j : \|x - x_j\| \leqslant \delta\}$ is $\mathbb{P}_m(\mathbb{R}^d)$-unisolvent then $A(x) = P^T W P$ is positive definite [19] and the MLS approximation is well-defined at sample point x. Of course if $\|x - x_j\| \geqslant \delta$ then $a_j(x) = 0$. Thus, in programming we can only form P and W for active points X_x instead of X. Derivatives of u are usually approximated by derivatives of $\widehat{u}$,

$$D^\alpha u(x) \approx D^\alpha \widehat{u}(x) = \sum_{j=1}^{N} D^\alpha a_j(x) u(x_j), \quad x \in \Omega, \ \alpha = (\alpha_1, \dots, \alpha_d) \in \mathbb{N}_0^d. \tag{2.4}$$

These derivatives are sometimes called *standard* or *full* derivatives. Details are in [20–22] and any other text containing the application of MLS approximation.

The GMLS approximation can be introduced as below. Suppose that λ is a linear functional from the dual space U^*. The problem is the recovery of $\lambda(u)$ from nodal values $u(x_1), \dots, u(x_N)$. The functional λ can, for instance, describe point evaluations of u, its derivatives up to order m, and the weak formulations which involve u or a derivative against some test function. The approximation $\widehat{\lambda}(u)$ of $\lambda(u)$ should be a linear function of the data $u(x_j)$, i.e., it should have the form

$$\lambda(u) \approx \widehat{\lambda}(u) = \sum_{j=1}^{N} a_j(\lambda) u(x_j), \tag{2.5}$$

where $a_j(\lambda)$ are shape functions associated to the functional λ. If λ is chosen to be the *point evaluation* functional δ_x, where $\delta_x(u) := u(x)$, then the classical MLS approximation (2.1) is obtained. If we assume λ is finally evaluated at sample point x, then the same weight function $w(x, y)$ as in the classical MLS can be used which is independent of the choice of λ. Using this assumption, analogous to (2.3), [15] proves,

$$\boldsymbol{a}(\lambda) := [a_1(\lambda), \dots, a_N(\lambda)] = \lambda(\boldsymbol{p})(P^T W P)^{-1} P^T W, \tag{2.6}$$