Applied Mathematics and Computation 270 (2015) 879–889

Contents lists available at ScienceDirect

Applied Mathematics and Computation

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

Moving least square for systems of integral equations

Mashallah Matin far, Masoumeh Pourabd*

Department of Mathematics, Science of Mathematics Faculty, University of Mazandaran, Iran

ARTICLE INFO

MSC:
65R20
65D15

Keywords:
Meshless method
Moving least square
The system of Fredholm integral equation
Numerical solution
Error analysis

ABSTRACT

This paper aims at developing a meshless approximation based on the Moving Least Square (MLS), in addition to its application for solving a system of linear Fredholm integral equations of the second kind. For the MLS, nodal points are used to approximate the unknown functions. These points can be selected as regular or random from the domain under study. The method is a meshless one, and since it uses a local shape function in the vicinity of each nodal point which is chosen from the support points, it does not depend on the geometry of the domain. In this method, the unknown function is considered as a vector of functions of its kind. An error analysis has also been provided for this new method. A simple and efficient application of this method has also demonstrated through several numerical examples.

1. Introduction

The solutions of integral equations have a major role invarious fields of science and engineering. The mathematical modeling of any natural phenomenon leads to a linear or nonlinear equation which may be an ordinary or a partially differential one, the Fredholm or Volterra integral one, an integro-differential one, or a system of these functional equations. In [1,2,22], we can find several methods for solving the integral equations and the system of integral equations. A number of recent meshless methods have been attracting much attention and have considerably become widespread. A number of the well-known approaches are listed as follows: Element-Free Galerkin (EFG) [3], Boundary Node (BN) [4], hp-cloud [5], Meshless local boundary integral equation (LBIE) [6], Meshless Local Petrov–Galerkin (MLPG) [7,8]. particularly Moving Least Squares (MLSs) method for solving partial differential equations. In this paper, a numerical method, based on the MLS, which utilizes some distributed nodal points to approximate the unknown functions is presented. In order to solve a system of integral equations, the distribution of the nodes was possible to be selected regularly or randomly in the domain under study. At first, the nodal points are selected, after which the shape functions are generated in any subdomain in the neighborhood, of these, of this nodes, and then the unknown functions, as components of the unknown vector function, will be approximated by these shape functions. Using the computer for implementing the method makes it flexible for being applied in most classes of systems of integral equations. A system of linear Fredholm integral equations can be presented as follows.

$$U(x) = F(x) + \int_a^b K(x,t)U(x)dt \tag{1.1}$$

where

$$U(x) = (u_1(x), u_2(x), \ldots u_n(x))^T \qquad \textit{Vector of unknown functions},$$

* Corresponding author at: Tel.: +989113380354; fax: +98 01135302460.
E-mail address: m.pourabd@gmail.com (M. Pourabd).

http://dx.doi.org/10.1016/j.amc.2015.08.098
0096-3003/

$$F(\mathrm{x}) = (f_1(\mathrm{x}), f_2(\mathrm{x}), \dots f_n(\mathrm{x}))^T \quad \textit{Vector of known functions},$$
$$K(\mathrm{x}, t) = [k_{ij}(\mathrm{x}, t)] \quad i, j = 1, 2, \dots, n \quad \textit{Matrix of Kernels}. \tag{1.2}$$

2. Moving least square approximation

The moving least square (MLS) approximation may be considered as a mesh-less scheme that has the properties of local interpolation, high accuracy, and can be applied easily for an equation in n-dimensional space. This is obtained through the use of the weight functions that allow control of the locality and continuity of the approximation. The MLS method started with Shepard's method [9] and was later extended by McLain [10,11], Franke and Nielson [12], and Lancaster and Salkauskas [13]. Now we will follow the well-known works on MLS. Consider a sub-domain Ω_{x} neighborhood of a point x and the domain of definition of the MLS approximation for the trial function at x, which is located in the problem domain Ω_{x}. The data $(\mathrm{x}_j, u_i(\mathrm{x}_j)), j = 1, 2, \dots N, i = 1, 2, \dots n$ is considered, so that $\mathrm{x}_j \in \Omega_{\mathrm{x}}$ and $u_i(\mathrm{x}_j) \in \mathbb{R}$ approximate the unknown function, u_i in Ω_{x} over a number of randomly located nodes $\mathrm{x}_i, i = 1, 2, \dots N$, the MLS approximant $u_i^a(\mathrm{x})$, so a represents an approximation of u_i, for all $\mathrm{x} \in \Omega_{\mathrm{x}}$, can be defined as

$$u_i^a(\mathrm{x}) = \sum_{k=1}^{m} c_k(\mathrm{x}) p_k(\mathrm{x}) = \mathbf{P}^T(\mathrm{x})\mathbf{c}(\mathrm{x}), \;\; i = 1, 2, \dots, n \tag{2.1}$$

Let Π_m the space of polynomial of degree m, and let $B_m = \{p_1, p_2, \dots p_m\}$ be any basis of Π_m. we need to determine the coefficients $c_j \in \mathbf{c}$, $j = 1, 2, \dots, m$ and $\mathbf{c} \in \mathbb{R}^m$ in (2.1) such that $\left\| u_i - u_i^a \right\|_{2,w}$ is minimized. Namely

$$\begin{aligned} \xi(\mathbf{c}) &= \left\| u_i - u_i^a \right\|_{2,w} = \sum_{j=1}^{N} [u_i(\mathrm{x}_j) - u_i^a(\mathrm{x}_j)]^2 w(\mathrm{x}_j), \\ &= \sum_{j=1}^{N} [\mathbf{P}^T(\mathrm{x}_j)\mathbf{c}(\mathrm{x}) - u_i(\mathrm{x}_j)]^2 w(\mathrm{x}_j) \\ &= [P.\mathbf{c} - \mathbf{u}_i]^T.W.[P.\mathbf{c} - \mathbf{u}_i], \end{aligned} \tag{2.2}$$

is minimized. Where c is a vector containing the coefficients c_j and w is the weight function. Note that x_j is the value of x at j'th node, N is the number of nodes in Ω_{x} with $w(\mathrm{x}_j) > 0$, the matrices P and W are defined as:

$$P = [\mathbf{p}(\mathrm{x}_1)^T, \mathbf{p}(\mathrm{x}_2)^T, \dots, \mathbf{p}(\mathrm{x}_N)^T]^T_{N\times(m+1)} \tag{2.3}$$

$$W = (diag(w(\mathrm{x}_j)), j = 1, 2, \dots, N \tag{2.4}$$

Note that $\mathbf{p} \in \Pi_m$ so that Π_m is a complete monomial basis of order m. For example, for a 1-D problem, the linear basis is $\{1, x\}$ and the quadratic basis is $\{1, x, x2\}$ and so on. Also $\mathbf{u}_i$ is a vector of i'th unknown function from system (1.1) in node x_j, so

$$\mathbf{u}_i = [u_i(\mathrm{x}_1), u_i(\mathrm{x}_2), \dots u_i(\mathrm{x}_N)]. \tag{2.5}$$

From Eq. (2.2) we have

$$\xi(\mathbf{c}) = \mathbf{c}^T P^T W P \mathbf{c} - 2\mathbf{u}_i^T W P \mathbf{c} + \mathbf{u}_i^T W \mathbf{u}_i \tag{2.6}$$

By minimizing the Eq. (2.6), $\nabla\xi(\mathbf{c}) = 0$, and defining the matrices $A(\mathrm{x})$ and $B(\mathrm{x})$ as

$$B(x) = [w_1(\mathrm{x})\mathbf{p}(\mathrm{x}_1), w_2(\mathrm{x})\mathbf{p}(\mathrm{x}_2), \dots w_N(\mathrm{x})\mathbf{p}(\mathrm{x}_N)] = P^T W \tag{2.7}$$

$$A(\mathrm{x}) = \sum_{i=1}^{N} w_i(\mathrm{x})\mathbf{p}(\mathrm{x}_i)^T\mathbf{p}(\mathrm{x}_i) = P^T W P = B(\mathrm{x})P \tag{2.8}$$

With respect to $\mathbf{c}(\mathrm{x})$ we have the following linear relation between $\mathbf{c}(\mathrm{x})$ and $\mathbf{u}_i$,

$$A(\mathrm{x})\mathbf{c}(\mathrm{x}) = B(\mathrm{x})\mathbf{u}_i \tag{2.9}$$

We know that Eq. (2.9) is well define when the matrix A in (2.8) is nonsingular. It can be seen that this is the case if and only if the rank of P equals to m. computing $\mathbf{c}(\mathrm{x})$ from (2.9) and substituting it into Eq. (2.1) give results in

$$u_i^a(\mathrm{x}) = \sum_{j=1}^{m} c_j(\mathrm{x}) p_j(\mathrm{x}) = \sum_{j=1}^{m} p_j(\mathrm{x})[A^{-1}(\mathrm{x})B(\mathrm{x})]\mathbf{u}_i, \;\; i = 1, \dots, n \tag{2.10}$$

We define

$$\Phi(\mathrm{x}) = \mathbf{p}^T(\mathrm{x})A(\mathrm{x})^{-1}B(\mathrm{x}) \tag{2.11}$$