Moving least square method for treating nonlinear fourth order integro-differential equations

A. Abdollahpoor

Department of Mathematics, Science and Research Branch, Islamic Azad University, Tehran, Iran

Copyright 2014 © A. Abdollahpoor. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract
In this paper, we introduce a numerical scheme based on the moving least squares (MLS) method with utilizing collocation points for solving linear and nonlinear fourth order integro-differential equations. The main advantage of the presented method is that it requires no domain elements for interpolation or approximation. Some cases of the mentioned equations are solved as examples to illustrate the ability and reliability of the method. The results reveal that the method is very effective and convenient.

Keywords: Moving least squares approximation; Integro-differential equations; Chebyshev polynomials; Gauss-Legendre quadrature.

1 Introduction
In this paper, a method based on moving least squares method and Chebyshev polynomials is presented for addressing the following fourth order integro-differential equation:

\[
\begin{align*}
   & u^{(iv)}(x) = f(x) + \gamma u(x) + \int_0^x g(t)u(t) + h(t)F(u(t)) \, dt, \\
   & u^{(i)}(a) = \alpha_i, \quad i = 0, 1, 2, 3.
\end{align*}
\]  

(1.1)

where \( F \) is a real nonlinear continuous function, \( \gamma, \alpha_i, \) \( i = 0, 1, 2, 3, \) are real constants, and \( f(x), g(x) \) and \( h(x) \) are given and can be approximated by Taylor polynomials, which plays a significant role in many fields of applied science. Therefore, the problem has attracted much attention and has been studied by many authors. See [3], for existence and uniqueness of solutions of Eq.(1.1).

The pseudo spectral method is proposed by using shifted Chebyshev method for solving the integro-differential equations [17]. Abbasbandy and Shivanian [1] extended the variational iteration method for solving Fredholms nth-order integro-differential equations. In [18], Wazwazs main objective was only to obtain the exact solutions to two fourth-order integro-differential equations. The Adomian decomposition method [11] and variational iteration method [16] are applied to solve both linear and nonlinear boundary value problems for fourth-order integro-differential equations. Ref.[2] is devoted to interpretation of the fourth order Volterra integro-differential equations by using modified homotopy perturbation method. Some recent contributions to the numerical method can be found in [4] -[10].

In this paper, we present a MLS method for solving (1.1) and obtain an accurate numerical solution. The superiority of the MLS method is a meshless method, because it does not require any interpolation or approximation and it dose

*Corresponding author. Email address: atena.abdollahpoor@yahoo.com
not depend to the geometry of domain.
The rest of the paper is organized as follows: In Section 2, a brief discussion of MLS method is outlined. The MLS method for solving (1.1) is discussed in Section 3. In Section 4, numerical results for some problems, are obtained. Finally, in Section 5, this report ends with a conclusion.

2 An outline of the MLS method

The MLS as an approximation method has been introduced by Shepard [15] and Lancaster and Salkausks [13]. Recently, many meshless methods based on the MLS method for the numerical solution of functional equations are considered, because the numerical approximations of MLS are based on a cluster of scattered nodes instead of interpolation on elements. In what follows we highlight briefly the main point of the MLS method, where details can be found in [12]-[15]. Let \( u: [a, b] \to \mathbb{R} \) be a continuous real function and the points \((x_i, u_i), \quad i = 0, \ldots, n\) are known. Let \( P_q \) be the space of polynomials of degree \( q \leq n \). The MLS approximation \( \hat{u}(x) \) of \( u(x), \forall x \in \Omega \), can be defined by:

\[
\hat{u}(x) = b^T a(x),
\]

(2.2)

where

\[
b(x) = [b_0(x), b_1(x), \ldots, b_m(x)]^T
\]

and \([b_i(x)]_{i=0}^m\) is an ordered basis for \( P_q \) and, \( a(x) \) is a vector with components \( a_j(x), \quad j = 0, 1, \ldots, m \), which are functions of the space coordinates \( x \). Also \( a_j(x) \)'s are unknown coefficients to be determined. The MLS method presents the approximate function \( \hat{u}(x) \) in a particularized class of differentiable functions which minimizes the quantity:

\[
J[\hat{u}] = \sum_{i=0}^{n} w_i(x)(\hat{u}(x_i) - u_i)^2,
\]

(2.3)

where \( w_i(x) \) is the weight function associated whit the node \( i; n \) is the number of nodes in \( \Omega \) for which the weight function \( w_i(x) > 0 \) and \( u_i \)'s are the function nodal values, but not the nodal values of the unknown trial function \( \hat{u}(x) \) i.e. \( \hat{u}(x_i) \neq u_i \), different weight function have been used to construct MLS shape function, in this paper we use the Gaussian weight functions applied in the present work as

\[
w_j = \begin{cases} 
\exp\left(-\frac{d_j^2}{\alpha^2} - \frac{h_j^2}{\alpha^2}\right), & 0 \leq d_j \leq h_j, \\
1 - \exp\left(-\frac{h_j^2}{\alpha^2}\right) & d_j > h_j,
\end{cases}
\]

(2.4)

where \( d_j = |x - x_j| \), \( \alpha \) is a constant controlling the shape of the weight functions \( w_j \) and \( h_j \) is the size of the support domain. To minimize \( J[\hat{u}] \), it is necessary that \( \Delta J = 0 \), which impliess the following normal equation

\[
J[\hat{u}] = \sum_{i=0}^{n} w_i(x)(b_i(x)a(x) - u_i)^2
\]

(2.5)

An extremum of \( J \) in (2.5) with respect to the coefficients \( a(x) \) can be obtained by setting the derivative of \( J \) with respect to \( a(x) \) equal to zero, the following equations result:

\[
\sum_{i=0}^{n} 2w_i(x)b_0(x_i)(b(x_i)a(x) - u_i) = 0
\]

(2.6)

\[
\sum_{i=0}^{n} 2w_i(x)b_1(x_i)(b(x_i)a(x) - u_i) = 0
\]

\[
\vdots
\]

\[
\sum_{i=0}^{n} 2w_i(x)b_m(x_i)(b(x_i)a(x) - u_i) = 0
\]
Rearrangements Eqs.(2.6) get,
\[ \sum_{i=0}^{n} w_i(x) b(x_i)(b(x_i)^{\prime} a(x) - u_i) = \sum_{i=0}^{n} w_i(x) b(x_i) u_i \]  
(2.7)

Which by setting:
\[ A(x) = \sum_{i=0}^{n} w_i(x) b(x_i) b(x_i)^{\prime}, \]
\[ U = [u_0, u_1, \ldots, u_n]^T, \]

That the matrix \( A(x) \) is often called moment matrix, it is of size \( m \times m \), and
\[ B(x) = [w_0(x) b(x_0), w_1(x) b(x_1), \ldots, w_n(x) b(x_n)]. \]

Becomes as follows
\[ A(x) a(x) = B(x) U, \]  
(2.8)

And by selecting the nodal points such that \( A(x) \) be nonsingular, Eq. (2.8) can be written as
\[ a(x) = A(x)^{-1} B(x) U, \]  
(2.9)

Substituting Eq. (2.9) into (2.2) gives
\[ \hat{u}(x) = b(x)^{\prime} A^{-1}(x) B(x) U = \sum_{i=0}^{n} \phi_i(x) u_i, \quad x \in \bar{\Omega}, \]
(2.10)

where
\[ \phi_i(x) = \sum_{k=0}^{m} b_k(x) [A^{-1}(x) B(x)]_{ki}. \]
(2.11)

and \( \phi_i(x) \) are called the shape functions of the MLS approximation, corresponding to nodal point \( x_j \), [12].

In general, different basis can be used for mentioned method such as Chebyshev, Montz, Legendre and etc polynomials in this method, but here the Chebyshev polynomials is used as a basis. Since these are important in approximation theory and numerical analysis and in some quadrature rules based on Chebyshev polynomials that appear in the theory of numerical integration[9]. The sequence \( \{T_i(x)\}_{i=0}^{\infty} \) of Chebyshev polynomials is obtained by following recurrence relations as,
\[ T_0(x) = 1 \]
\[ T_1(x) = x, \]
\[ T_j(x) = 2xT_{j-1}(x) - T_{j-2}(x), \quad j \geq 2, \quad -1 \leq x \leq 1. \]

To use of Chebyshev polynomials on the interval \([0, 1]\), we need to change the domain using the following substitution
\[ z = 2x - 1, \quad 0 \leq x \leq 1. \]

So, the shifted Chebyshev polynomials \( T_n^*(x) \) on \([0, 1]\) are obtained as follows:
\[ T_n^*(x) = T_n(2x - 1). \]
3 Description of the method

In this section, we use MLS method for obtaining the numerical solutions of (1.1), where the shifted Chebyshev polynomials are used as basis functions to estimate the solution of the integral parts. To this aim, we rewrite Eq. (1.1) as following form

$$Lu(x) = f(x) + \gamma u(x) + \int_0^1 [g(t)u(t) + h(t)F(u(t))] dt,$$

(3.12)

where \( L = \frac{d^4}{dt^4} \). Suppose that the 4-fold operator,

$$L^{-1}(\cdot) = \int_0^1 \int_0^1 \int_0^1 \int_0^1 (\cdot) dt dt dt dt = \int_0^1 \frac{(x-t)^3}{3!} (\cdot) dt,$$

(3.13)

exists, by applying (3.13) on both sides of (3.12) we have

$$u(x) = \sum_{j=0}^3 \alpha_j x^j + L^{-1}(f(x)) + \gamma u(x) + \int_0^1 \frac{(x-t)^3}{3!} [g(t)u(t) + h(t)F(u(t))] dt,$$

(3.14)

Now, to use the shifted chebyshev polynomials, we rewrite the problem (3.14) as follows:

$$u(x) = \sum_{j=0}^3 \alpha_j x^j + L^{-1}(f(x)) + \gamma u(x) + x \int_0^1 \frac{(x-t)^3}{3!} [g(t)u(t) + h(t)F(u(t))] dt,$$

(3.15)

To apply the method, we select the \( n + 1 \) roots of the \( T_{n+1}^*(x) \) as nodal points \( x_i \) in \([0,1]\). By setting \( \sum_{i=0}^n u_i \phi_i(x) \) instead of \( \hat{u}(x) \) as an approximation of \( u(x) \) in (3.15) yields:

$$\sum_{j=0}^n (\phi_j(x) - \sum_{k=0}^3 \alpha_k x^k) - \gamma L^{-1}(\phi_j(x)) - x \int_0^1 \frac{(x-t)^3}{3!} [g(x)u(x) + h(x)F(u(x))] dx = L^{-1}(f(x))$$

(3.16)

Now consider (3.16) be exact for the collocation points \( \{x_i\}_{i=0}^n \), so we find that

$$\sum_{j=0}^n (\phi_j(x_i) - \sum_{k=0}^3 \alpha_k x_i^k) - \gamma L^{-1}(\phi_j(x_i)) - x_i \int_0^1 \frac{(x_i-t)^3}{3!} [g(x)u(x_i) + h(x)F(u(x_i))] dx = L^{-1}(f(x_i))$$

(3.17)

Calculating integrals in (3.17) numerically by using \( l \) point quadrature formula with the quadrature points \( \{x_i\}_{i=0}^l \) and the quadrature weights \( \{W_k\}_{k=0}^l \). Therefore, we have an \( (n + 1) \times (n + 1) \) system of equations with \( n + 1 \) unknowns \( \hat{u}_i \) that must be obtained.

4 Numerical examples

To give a clear overview of the methodology as a numerical tool, we consider three examples in this section. For numerical approximate solution we take for the linear case \( h_j = \frac{2\pi}{n-1} \), for the quadratic case \( h_0 = \frac{2.5}{n^2} \) and for the degree 3 case \( h_0 = \frac{3}{n^3} \) and \( \alpha = \frac{0.9}{n^4} \) for all of them to ensure the invertibility of the matrix \( A \) in MLS method. Moreover, to compute numerical integro-differential, we use 5-point Gauss-Legendre quadrature rule. All the results are calculated by using Maple 13 with 20 digits precision.

Example 4.1. Consider the linear forth-order integro-differential equations as in (1.1) with \( f(x) = \frac{3!}{\Gamma(2)} x + x^3(1 + \frac{1}{7} x^2) \), \( \gamma = -1 \), \( g(x) = -x \) and \( h(x) = 0 \), i.e.

$$u^{(v)}(x) = \frac{3!}{\Gamma(2)} x + x^3(1 + \frac{1}{7} x^2) - u(x) + \int_0^x tu(t) dt,$$

with initial conditions

$$u^{(i)}(0) = 0, \quad i = 0, 1, 2, 3.$$
The exact solution is \( u(x) = x^5 \), now we transform this equation to:

\[
  u(x) = \frac{1}{55440}x^{11} + \frac{1}{3024}x^9 + x^5 - \frac{1}{24}x \int_0^1 [4(x-\xi)^3 - (x-\xi)^4]\xi u(\xi) d\xi,
\]

and then utilize the shifted Chebyshev polynomials. Table 1 shows the absolute errors for different \( m, n \)'s, and Figure 1 depicting the absolute error of degree 2 with 23 points.

<table>
<thead>
<tr>
<th>( n )</th>
<th>( m=1 )</th>
<th>( m=2 )</th>
<th>( m=3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>( 3 \times 10^{-3} )</td>
<td>( 7 \times 10^{-3} )</td>
<td>( 3 \times 10^{-3} )</td>
</tr>
<tr>
<td>9</td>
<td>( 3.5 \times 10^{-4} )</td>
<td>( 2 \times 10^{-3} )</td>
<td>( 7 \times 10^{-4} )</td>
</tr>
<tr>
<td>13</td>
<td>( 2 \times 10^{-3} )</td>
<td>( 1 \times 10^{-3} )</td>
<td>( 1.8 \times 10^{-5} )</td>
</tr>
<tr>
<td>23</td>
<td>( 1 \times 10^{-3} )</td>
<td>( 2 \times 10^{-4} )</td>
<td>( 1.6 \times 10^{-5} )</td>
</tr>
</tbody>
</table>

**Figure 1**: The MLS approximation error with \( m = 2, n = 23 \) for Example 4.1

**Example 4.2.** Consider the following nonlinear forth-order [11]:

\[
  u^{(iv)}(x) = 1 + \int_0^x e^{-t} u(t)^2 dt,
\]

Subject to

\[
  u^{(i)}(0) = 1, \quad i = 0, 1, 2, 3.
\]
with exact solution \( u(x) = e^x \). So we can write

\[
  u(x) = 1 + x + \frac{1}{2} x^2 + \frac{1}{6} x^3 + \frac{1}{24} x^4 + \frac{1}{24} \int_0^x (x - xs)^3 e^{-xs} u(xs)^2 ds,
\]

Table 2 reports the numerical results, and Figure 2 shows the absolute error of degree 2 with 5 points.

<table>
<thead>
<tr>
<th>x</th>
<th>Exact Solution</th>
<th>Numerical Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>m=1, n=5</td>
</tr>
<tr>
<td>0.0</td>
<td>1.0000000000</td>
<td>1.000181101</td>
</tr>
<tr>
<td>0.2</td>
<td>1.221402758</td>
<td>1.222079114</td>
</tr>
<tr>
<td>0.4</td>
<td>1.491824698</td>
<td>1.492417219</td>
</tr>
<tr>
<td>0.6</td>
<td>1.822118800</td>
<td>1.822292248</td>
</tr>
<tr>
<td>0.8</td>
<td>2.225540928</td>
<td>2.225435590</td>
</tr>
<tr>
<td>1.0</td>
<td>2.718281828</td>
<td>2.718481666</td>
</tr>
</tbody>
</table>

Figure 2: The MLS approximation error with \( m = 2, n = 5 \) for Example 4.2

**Example 4.3.** Consider the following nonlinear fourth-order integro-differential equations as in form (1.1), where

\[
  f(x) = -\frac{1}{2} xe^{-2} + \frac{1}{2} xe^{x^2-2}, \quad \gamma = 0, \quad g(x) = -x \text{ and } h(x) = 0,
\]

so we have

\[
  u^{(iv)}(x) = -\frac{1}{2} xe^{-2} + \frac{1}{2} xe^{x^2-2} - \int_0^x xt e^{u(t)} dt,
\]
with initial conditions \( u(0) = -2, u'(0) = 2, u''(0) = u'''(0) = 0 \). The analytic solution of this problem is \( u(x) = x^2 - 2 \).

Table 3 interprets the numerical results and the absolute error of degree 1 with 5 points are plotted in Figure 3.

<table>
<thead>
<tr>
<th>( x )</th>
<th>Exact Solution</th>
<th>Numerical Solution ( m=1, n=5 )</th>
<th>Numerical Solution ( m=2, n=5 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>-2.00</td>
<td>-1.999720506</td>
<td>-1.999992009</td>
</tr>
<tr>
<td>0.2</td>
<td>-1.96</td>
<td>-1.959399180</td>
<td>-1.960012642</td>
</tr>
<tr>
<td>0.4</td>
<td>-1.84</td>
<td>-1.839481548</td>
<td>-1.840014810</td>
</tr>
<tr>
<td>0.6</td>
<td>-1.64</td>
<td>-1.639473355</td>
<td>-1.640005748</td>
</tr>
<tr>
<td>0.8</td>
<td>-1.36</td>
<td>-1.359400900</td>
<td>-1.360014433</td>
</tr>
<tr>
<td>1.0</td>
<td>-1.00</td>
<td>-0.999836739</td>
<td>-1.000115904</td>
</tr>
</tbody>
</table>

Figure 3: The MLS approximation error with \( m = 1, n = 5 \) for Example 4.3

5 Conclusion

Moving least square method has been known as a convenient tool for solving most functional equations, like ordinary, partial differential equations, and integral equations. Often nonlinear integro-differential equations are usually difficult to solve analytically. In many cases, it is required to obtain the approximate solutions. For this aim, the MLS method can be used.

In this work, we solved fourth order integro-differential equations by using moving least square method, which utilizes some distributed nodal points to estimate the unknown function, for a solution of integral equations. In spite
of the relatively low degrees used the numerical results show the higher performance of the MLS method with the Chebyshev basis. According to the Figures 1-3, the error of the MLS solution shows a tendency to increase as $x$ increases to the end boundary point. This behavior can be expected in any numerical method.

References


http://dx.doi.org/10.1016/j.apnum.2009.12.003


http://dx.doi.org/10.1016/j.camwa.2006.12.055

http://dx.doi.org/10.1155/2013/434753

http://dx.doi.org/10.1016/S0096-3003(99)00225-8