The element-free Galerkin method of numerically solving a regularized long-wave equation*

Cheng Rong-Jun(程荣军)[a)†] and Ge Hong-Xia(葛红霞)[b)]

[a)] *Ningbo Institute of Technology, Zhejiang University, Ningbo* 315100, *China*
[b)] *Faculty of Science, Ningbo University, Ningbo* 315211, *China*

(Received 11 September 2011; revised manuscript received 28 September 2011)

The element-free Galerkin (EFG) method is used in this paper to find the numerical solution to a regularized long-wave (RLW) equation. The Galerkin weak form is adopted to obtain the discrete equations, and the essential boundary conditions are imposed by the penalty method. The effectiveness of the EFG method of solving the RLW equation is investigated by two numerical examples in this paper.

Keywords: element-free Galerkin method, meshless method, regularized long wave equation, solitary wave

PACS: 02.60.Lj, 03.65.Ge **DOI:** 10.1088/1674-1056/21/4/040203

1. Introduction

The study of numerical methods of solving nonlinear partial differential equations has enjoyed an intense period of activity over the last 40 years from both theoretical and practical points of view. The finite element method (FEM) and boundary element method (BEM) are well-established numerical techniques, which have been used to numerically solve various problems in the field of engineering and science.[1−3] Even though these methods are very effective for solving various kinds of partial differential equations, they also have their limitations. For the FEM, the need to produce a body-fitted mesh in two- and three-dimensional problems makes this method time-consuming and difficult to use. As for the BEM, for an inhomogeneous equation, it requires a domain node distribution in addition to a boundary mesh and depends on the fundamental solutions. To avoid these problems, in recent years meshless techniques have attracted the attention of researchers. In a meshless (mesh-free) method, a set of scattered nodes are used instead of meshing the domain of the problem. For many engineering problems, such as large deformation and crack growth, it is necessary to deal with extremely large deformations or fractures of the mesh with the re-meshing technique. The meshless method is a new and interesting numerical technique that can be used to solve many engineering problems that cannot be solved by using conventional numerical methods with a minimum of meshing or no meshing at all.[4] Some meshless methods have been developed, such as the smooth particle hydrodynamics (SPH) method,[5] the radial basis function (RBF),[6] the element-free Galerkin (EFG) method,[7] the reproducing kernel particle method (RKPM),[8,9] the meshless local Petrov–Galerkin (MLPG) method,[10] the finite point method(FPM),[11] and so on.

The regularized long-wave (RLW) equation was formulated by Peregrine[12,13] as an alternative to the Korteweg-de Vries (KdV) equation for studying soliton phenomena and as a model for small-amplitude long waves on the surface of water. A rather interesting property of the RLW equation is that the collision between two solitary waves may result in the creation of secondary solitary waves or sinusoidal solutions. This phenomenon is somewhat analogous to what happens in subatomic physics where the collisions between particles create another particles and/or radiation. Therefore, a study of RLW equations provides an opportunity to investigate the creation of secondary solitary waves and/or radiation to gain insight into the corresponding processes of particle physics.[14,15] The collisions between solitary waves of the RLW equations differ from the collision between solitons of the KdV equation in that, in the latter soliton collision,

*Project supported by the Natural Science Foundation of Zhejiang Province of China (Grant No. Y6110007).
†Corresponding author. E-mail: chengrongjun76@126.com
© 2012 Chinese Physical Society and IOP Publishing Ltd

interaction and emergence are unchanged with the collision except for a phase shift. By comparison, the numerical solution of the RLW equation indicates that two solitary waves can effectively pass through each other but with diminished amplitude, and create secondary waves. In addition, a notch in the negative amplitude wave may appear and this notch develops into a secondary negative amplitude wave. The positive and the negative amplitude waves may move in opposite directions, thereby forming multiple secondary waves of positive and negative amplitudes, thus the annihilation of positive and negative waves and the creation of a sinusoidal residual referred to as radiation, may occur.

Previous numerical studies of the RLW equation include the use of quartic-B spline collocation methods for the spatial discretization together with an explicit fourth-order accurate Runge–Kutta technique for time integration,[16] least-square finite element techniques with linear space–time elements and cubic B-spline finite element techniques in which the Galerkin method was used,[17,18] Fourier-pseudo spectral methods for periodic boundary conditions,[19−22] mixed finite element methods based on the introduction of an intermediate variable,[23] operator-splitting techniques with cubic,[24,25] and Crank–Nicolson time discretization, Adomian's decomposition method,[26,27] perturbation (WKBJ) techniques,[28] and the explicit finite difference method.[29−32]

In recent years, meshless techniques have attracted the attention of researchers. In a meshless (mesh-free) method, a set of scattered nodes is used instead of meshing the domain of the problem. The meshless method is a new and interesting numerical technique that can be used to solve many engineering problems that cannot be solved by the conventional numerical methods with a minimum of meshing or no meshing.[33−53] The element-free Galerkin (EFG) method has been greatly developed and widely applied in various fields, but has not yet been applied to the RLW equation. In this paper, the EFG method is for the first time used to numerically solve the RLW equation. In this method, the function over the solution domain needs only a set of nodes, it does not require element connectivity. The Galerkin weak form is used to obtain the corresponding discrete equations and the essential boundary conditions are imposed by using the penalty method. The numerical results, which are in good agreement with the exact solutions, are presented in this paper.

2. The moving least-square approximation[4]

The EFG method is used in the moving least-square approximation, which is constructed in terms of nodes only. Therefore the element and the element connectivity data are not needed, which, however, are required in the FEM. The discretization of the governing equations by the element-free Galerkin method requires the moving least-square approximant, which is composed of three components: a weight function associated with each node, a monomial basis, and a set of coefficients that depend on node position.

In the moving least-square approximation,[4] let

$$u^h(\boldsymbol{x}) = \sum_{j=1}^{m} p_j(\boldsymbol{x}) a_j(\boldsymbol{x}) \equiv \boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}) \boldsymbol{a}(\boldsymbol{x}), \tag{1}$$

where m is the number of terms in the basis, $p_j(\boldsymbol{x})$ are the monomial basis function, and $\boldsymbol{a}_j(\boldsymbol{x})$ are their coefficients of the functions of the spatial coordinates $\boldsymbol{x}$ as indicated.

Lancaster and Salkauskas defined a local approximation as

$$u^h(\boldsymbol{x}\bar{\boldsymbol{x}}) = \sum_{j=1}^{m} p_j(\bar{\boldsymbol{x}}) a_j(\boldsymbol{x}) \equiv \boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}) a(\boldsymbol{x}), \tag{2}$$

where the coefficients $a_j(\boldsymbol{x})$ are obtained by performing a weighted least-square fitting for the local approximation. This defines the quadratic form

$$\begin{aligned} J &= \sum_{i=1}^{n} w(\boldsymbol{x} - \boldsymbol{x}_i) \left[u^h(\boldsymbol{x}, \boldsymbol{x}_i) - u(\boldsymbol{x}_i)\right]^2 \\ &= \sum_{i=1}^{n} w(\boldsymbol{x} - \boldsymbol{x}_i) \left[\sum_{j=1}^{m} p_j(\boldsymbol{x}_i) a_j(\boldsymbol{x}) - u_i\right]^2, \end{aligned} \tag{3}$$

where $w(\boldsymbol{x} - \boldsymbol{x}_i)$ are the weight functions with compact support, $\boldsymbol{x}_i$ are the nodes in the compacted supported domain of $\boldsymbol{x}$. Equation (3) can be rewritten in the vector form

$$J = (\boldsymbol{pa} - \boldsymbol{u})^{\mathrm{T}} \boldsymbol{W}(\boldsymbol{x})(\boldsymbol{pa} - \boldsymbol{u}), \tag{4}$$

where

$$\boldsymbol{u}^{\mathrm{T}} = (u_1, u_2, \ldots, u_n), \tag{5}$$

$$\boldsymbol{p} = \begin{bmatrix} p_1(\boldsymbol{x}_1) & p_2(\boldsymbol{x}_1) & \cdots & p_m(\boldsymbol{x}_1) \\ p_1(\boldsymbol{x}_2) & p_2(\boldsymbol{x}_2) & \cdots & p_m(\boldsymbol{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ p_1(\boldsymbol{x}_n) & p_2(\boldsymbol{x}_n) & \cdots & p_m(\boldsymbol{x}_n) \end{bmatrix}, \tag{6}$$

and

$$\boldsymbol{W}(\boldsymbol{x}) = \begin{bmatrix} w(\boldsymbol{x}-\boldsymbol{x}_1) & 0 & \cdots & 0 \\ 0 & w(\boldsymbol{x}-\boldsymbol{x}_2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & w(\boldsymbol{x}-\boldsymbol{x}_n) \end{bmatrix}. \quad (7)$$

To find the coefficients $a(x)$, we obtain the extremum of J by

$$\frac{\partial J}{\partial \boldsymbol{a}} = \boldsymbol{A}(\boldsymbol{x})\boldsymbol{a}(\boldsymbol{x}) - \boldsymbol{B}(\boldsymbol{x})\boldsymbol{u} = 0, \quad (8)$$

where $\boldsymbol{A}$ is called the moment matrix and given by

$$\boldsymbol{A} = \boldsymbol{p}^{\mathrm{T}}\boldsymbol{W}(\boldsymbol{x})\boldsymbol{p}, \quad (9)$$

$$\boldsymbol{B} = \boldsymbol{p}^{\mathrm{T}}\boldsymbol{W}(\boldsymbol{x}). \quad (10)$$

So we have

$$\boldsymbol{a}(\boldsymbol{x}) = \boldsymbol{A}^{-1}(\boldsymbol{x})\boldsymbol{B}(\boldsymbol{x})\boldsymbol{u}. \quad (11)$$

The approximation $u^h(\boldsymbol{x})$ can be expressed as

$$\boldsymbol{u}(\boldsymbol{x}) \approx u^h(\boldsymbol{x}) = \sum_{i=1}^{n} \Phi_i^k(\boldsymbol{x})u_i = \boldsymbol{\Phi}^k(\boldsymbol{x})\boldsymbol{u}, \quad (12)$$

where the shape functions are given by

$$\begin{aligned} \boldsymbol{\Phi}^k(\boldsymbol{x}) &= \left(\boldsymbol{\Phi}_1^k(\boldsymbol{x}), \cdots, \Phi_n^k(x)\right) \\ &= \boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x})\boldsymbol{A}^{-1}(\boldsymbol{x})\boldsymbol{B}(\boldsymbol{x}), \end{aligned} \quad (13)$$

where the superscript k is the order of the polynomial basis. The functions corresponding to $k = 0$ are called the Shepard functions and are given by

$$\Phi_i(\boldsymbol{x}) = \frac{w(\boldsymbol{x}-\boldsymbol{x}_i)}{\sum\limits_{j=1}^{n} w(\boldsymbol{x}-\boldsymbol{x}_j)}. \quad (14)$$

The derivatives of the unknown u are approximated by

$$D^\alpha u^h(\boldsymbol{x}) = \sum_{i=1}^{n} D^\alpha \Phi_i^k(\boldsymbol{x})u_i = D^\alpha \boldsymbol{\Phi}^k(\boldsymbol{x})\boldsymbol{u}. \quad (15)$$

The weight function used in Eqs. (3)–(13) plays an important role in the EFG method, so the selection of an appropriate weight function is essential. The weight function $w(\boldsymbol{x}-\boldsymbol{x}_i)$ should be non-zero over only a small neighbourhood of $\boldsymbol{x}_i$, called the domain of influence of node $\boldsymbol{x}_i$, to generate a set of sparse discrete equations. The cubic spline weight function used in the present analysis can be written as a function of normalized radius r as follows:

$$\begin{aligned} w_i &= w(\boldsymbol{x}-\boldsymbol{x}_i) \equiv w(r) \\ &= \begin{cases} \frac{2}{3} - 4r^2 + 4r^3, & r \le \frac{1}{2}, \\ \frac{4}{3} - 4r + 4r^2 - \frac{4}{3}r^3, & \frac{1}{2} < r \le 1, \\ 0, & r > 1, \end{cases} \end{aligned} \quad (16)$$

where

$$r = \frac{d_i}{d_{m_i}}, \quad d_i = \|\boldsymbol{x}-\boldsymbol{x}_i\|, \quad d_{m_i} = d_{\max}c_i, \quad (17)$$

with $d_{\max}$ being a scaling parameter, and the distance c_i being chosen such that the moment matrix A is not singular.

The spatial derivates of the weight function are obtained as

$$\frac{\mathrm{d}w_i}{\mathrm{d}x} = \frac{\mathrm{d}w_i}{\mathrm{d}r}\frac{\mathrm{d}r}{\mathrm{d}x} = \begin{cases} (-8r + 12r^2)\dfrac{\mathrm{sign}(x-x_i)}{\mathrm{d}m_i}, & r \le \dfrac{1}{2}, \\ (-4 + 8r - 4r^2)\dfrac{\mathrm{sign}(x-x_i)}{\mathrm{d}m_i}, & \dfrac{1}{2} < r \le 1, \\ 0, & r > 1. \end{cases} \quad (18)$$

3. The element-free Galerkin method for the RLW equation

The regularized long wave (RLW) equation is given as

$$u_t + u_x + \varepsilon u u_x - \mu u_{xxx} = 0, \quad (a < x < b), \quad (19a)$$

where μ is the positive parameter and subscripts x and t denote the differentiation.

The boundary conditions are

$$u(a,t) = \alpha, \quad u(b,t) = \beta, \quad (19b)$$

where α and β denote real numbers.

The initial condition is

$$u(x,0) = f(x). \quad (19c)$$

The weak form of Eq. (19) is obtained as follows:

$$\int_\Gamma \delta u^{\mathrm{T}}(u_t + u_x + \varepsilon u u_x - \mu u_{xxx})\mathrm{d}\Gamma$$

$$+\alpha\delta u^{\mathrm{T}}(u-\bar{u})|_{\Gamma_u}=0, \tag{20}$$

where Γ denotes the domain of x and Γ_u the boundaries, $\bar{u}$ are the values of x at the boundaries, and α is a penalty parameter.

By the subsection integration, we can obtain

$$\int_\Gamma \delta u^{\mathrm{T}}u_t\,\mathrm{d}\Gamma+\int_\Gamma \delta u^{\mathrm{T}}u_x\,\mathrm{d}\Gamma+\varepsilon\int_\Gamma \delta u^{\mathrm{T}}uu_x\,\mathrm{d}\Gamma +\mu\int_\Gamma \delta u_x^{\mathrm{T}}u_{xx}\,\mathrm{d}\Gamma+\alpha\delta u^{\mathrm{T}}(u-\bar{u})|_{\Gamma_u}=0. \tag{21}$$

With Eq. (12), we can obtain the approximants of the test function as

$$u^h(x,t)=\sum_{I=1}^{n}\boldsymbol{\Phi}_I(x)\cdot u_I(t)=\Phi(x)\cdot U(t), \tag{22}$$

$$\begin{aligned}\frac{\partial u^h(x,t)}{\partial t}&=\frac{\partial}{\partial t}\sum_{I=1}^{n}\Phi_I(x)\cdot u_I(t)\\&=\sum_{I=1}^{n}\Phi_I(x)\cdot\frac{\partial U_I(t)}{\partial t}=\boldsymbol{\Phi}(x)\dot{\boldsymbol{U}}(t),\end{aligned} \tag{23}$$

$$\begin{aligned}\frac{\partial u^h(x,t)}{\partial x}&=\frac{\partial}{\partial x}\sum_{I=1}^{n}\Phi_I(x)\cdot u_I(t)\\&=\sum_{I=1}^{n}\Phi_{I,x}(x)\cdot u_I(t)=\boldsymbol{\Phi}_x(x)\boldsymbol{U}(t),\end{aligned} \tag{24}$$

$$\begin{aligned}\frac{\partial^2 u^h(x,t)}{\partial x^2}&=\frac{\partial}{\partial x\partial x}\sum_{I=1}^{n}\Phi_I(x)\cdot u_I(t)\\&=\sum_{I=1}^{n}\Phi_{I,xx}(x)\cdot u_I(t)=\boldsymbol{\Phi}_{xx}(x)\boldsymbol{U}(t),\end{aligned} \tag{25}$$

where

$$\boldsymbol{\Phi}(x)=(\Phi_1(x),\Phi_2(x),\ldots,\Phi_n(x)), \tag{26}$$

$$\boldsymbol{U}(t)=(u_1(t),u_2(t),\ldots,u_n(t))^{\mathrm{T}}, \tag{27}$$

$$\dot{\boldsymbol{U}}(t)=\left(\frac{\partial u_1(t)}{\partial t},\frac{\partial u_2(t)}{\partial t},\ldots,\frac{\partial u_n(t)}{\partial t}\right)^{\mathrm{T}}, \tag{28}$$

$$\boldsymbol{\Phi}_x(x)=(\Phi_{1,x}(x),\Phi_{2,x}(x),\ldots,\Phi_{n,x}(x)), \tag{29}$$

$$\boldsymbol{\Phi}_{xx}(x)=(\Phi_{1,xx}(x),\Phi_{2,xx}(x),\ldots,\Phi_{n,xx}(x)). \tag{30}$$

Substituting Eqs. (22)–(25) into Eq. (21) yields

$$\begin{aligned}&\delta U^{\mathrm{T}}(t)K\dot{U}(t)+\delta U^{\mathrm{T}}(t)TU(t)\\&+\delta U^{\mathrm{T}}(t)\int_\Gamma\varepsilon\Phi(uu_x)\,\mathrm{d}\Gamma\\&+\delta U^{\mathrm{T}}(t)GU(t)+\alpha\delta U^{\mathrm{T}}(t)HU(t)\\&-\alpha\delta U^{\mathrm{T}}(t)Q(t)=0.\end{aligned} \tag{31}$$

Since $\delta U^{\mathrm{T}}(t)$ is arbitrary in Eq. (31), the following equation can be obtained:

$$\boldsymbol{K}\dot{\boldsymbol{U}}(t)+(\boldsymbol{H}+\boldsymbol{G}+\boldsymbol{T})\boldsymbol{U}(t)+\int_\Gamma\varepsilon\Phi(uu_x)\,\mathrm{d}\Gamma=\boldsymbol{Q}(t), \tag{32}$$

where

$$K_{IJ}=\int_\Gamma\Phi_I(x)\Phi_J(x)\,\mathrm{d}\Gamma, \tag{33}$$

$$G_{IJ}=\mu\int_\Gamma\Phi_{I,x}(x)\Phi_{J,xx}(x)\,\mathrm{d}\Gamma, \tag{34}$$

$$H_{IJ}=\alpha(\Phi_I\Phi_J|_{x=a}+\Phi_I\Phi_J|_{x=b}), \tag{35}$$

$$Q_I=\alpha(\Phi_I\bar{u}|_{x=a}+\Phi_I\bar{u}|_{x=b}). \tag{36}$$

Forming the time discretization of Eq. (32) with the central difference method, we obtain

$$\boldsymbol{K}\frac{\boldsymbol{U}^{n+1}-\boldsymbol{U}^n}{\Delta t}+(\boldsymbol{H}+\boldsymbol{T}+\boldsymbol{G})\frac{\boldsymbol{U}^{n+1}+\boldsymbol{U}^n}{2}+\int_\Gamma\varepsilon\Phi\frac{(uu_x)^{n+1}+(uu_x)^n}{2}\,\mathrm{d}\Gamma=\frac{\boldsymbol{Q}^{n+1}+\boldsymbol{Q}^n}{2}, \tag{37}$$

where Δt is the time of step and

$$\begin{aligned}\boldsymbol{U}^n&=\boldsymbol{U}(n\Delta t)\\&=(u_1(n\Delta t),u_2(n\Delta t),\ldots,u_n(n\Delta t)).\end{aligned} \tag{38}$$

The nonlinear term $(uu_x)^{n+1}$ in Eq. (36) is linearized by using the following term[54]

$$(uv)^{n+1}=u^{n+1}v^n+u^nv^{n+1}-u^nv^n. \tag{39}$$

So we obtain

$$(uu_x)^{n+1}=u^{n+1}u_x^n+u_x^nu^{n+1}-u^nu_x^n. \tag{40}$$

Substituting Eq. (40) into Eq. (37) yields

$$\begin{aligned}&\boldsymbol{K}\frac{\boldsymbol{U}^{n+1}-\boldsymbol{U}^n}{\Delta t}+(\boldsymbol{H}+\boldsymbol{T}+\boldsymbol{G})\frac{\boldsymbol{U}^{n+1}+\boldsymbol{U}^n}{2}\\&+\left(\int_\Gamma\varepsilon\Phi\frac{\boldsymbol{U}^n(\Phi_x\Phi^{\mathrm{T}}+\Phi\Phi_x^{\mathrm{T}})}{2}\,\mathrm{d}\Gamma\right)\cdot\boldsymbol{U}^{n+1}\\&=\frac{\boldsymbol{Q}^{n+1}+\boldsymbol{Q}^n}{2}.\end{aligned} \tag{41}$$

By solving the above iteration equation, we can obtain the numerical solution of the RLW equation.

4. Numerical example

The application of the EFG method of numerically solving the RLW equation is demonstrated in this section by two examples. A regular arrangement of nodes and the background mesh of cells are used for numerical integrations to calculate the system equation. The linear basis and the Gauss weight function are used in the EFG method.

Example 1: solitary wave motion

Consider the following equation:

$$u_t+u_x+\varepsilon uu_x-\mu u_{xxx}=0,\quad (a<x<b), \tag{42}$$

with initial condition

$$u(x,0) = 3c\,\mathrm{sech}^2(k(x-x_0)) \tag{43}$$

and boundary condition

$$u(a,t) = 3c\,\mathrm{sech}^2(k(a-x_0-vt)), \tag{44a}$$

$$u(b,t) = 3c\,\mathrm{sech}^2(k(b-x_0-vt)). \tag{44b}$$

The exact solution is

$$u(x,t) = 3c\,\mathrm{sech}^2(k(x-x_0-vt)), \tag{45}$$

where $\varepsilon = 1$, $\mu = 1$, $c = 0.1$, $k = \frac{1}{2}\sqrt{c\varepsilon/v\mu}$, $v = 1+c\varepsilon$, and $x_0 = 0$.

The EFG method is used to solve the above equation with the penalty factor $\alpha = 10^5$ and scaling parameter $d_{\max} = 2.5$, and the time-step length $\Delta t = 0.01$, $a = -40$, and $b = 60$. The weight function is chosen to be a Gauss function and the bases are chosen to be linear. In Fig. 1, profiles of the solitary wave are shown at $t = 1$ and $t = 10$.

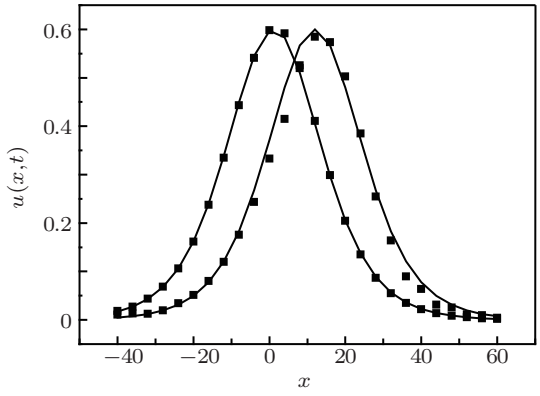


Fig. 1. Profiles of the solitary wave at time $t = 1$ and 10.

Example 2: modelling an undular bore

A bore is formed when a deeper stream of water flows into an area of still water in a long horizontal channel. When the transition between the deeper stream and the still water has a very gentle slope, the slope will steepen and a bore will form. To study the development of an undular bore we follow the initial condition

$$u(x,0) = 0.5U_0\left[1 - \tanh\left(\frac{x-x_c}{d}\right)\right] \tag{46}$$

with the boundary condition

$$u(a,t) = U_0, \quad u(b,t) = 0, \tag{47}$$

where $U(x,0)$ denotes the elevation of the water above the equilibrium surface at time $t = 0$. The change in water level of magnitude U_0 is centred at $x = x_c$, and d measures the steepness of the change. The smaller the value of d, the steeper the slope is.

The EFG method is used to solve the above equation with the penalty factor $\alpha = 10^5$ and scaling parameter $d_{\max} = 2.5$, the time-step length $\Delta t = 0.01$, $\varepsilon = 1.5$, $\mu = 1/6$, $U_0 = 0.1$, and $d = 5$. The weight function is chosen to be a Gauss function and the bases are chosen to be linear. Figure 2 shows the undular bore profiles at times $t = 10$ and $t = 50$ for the gentle slope $d = 5$.

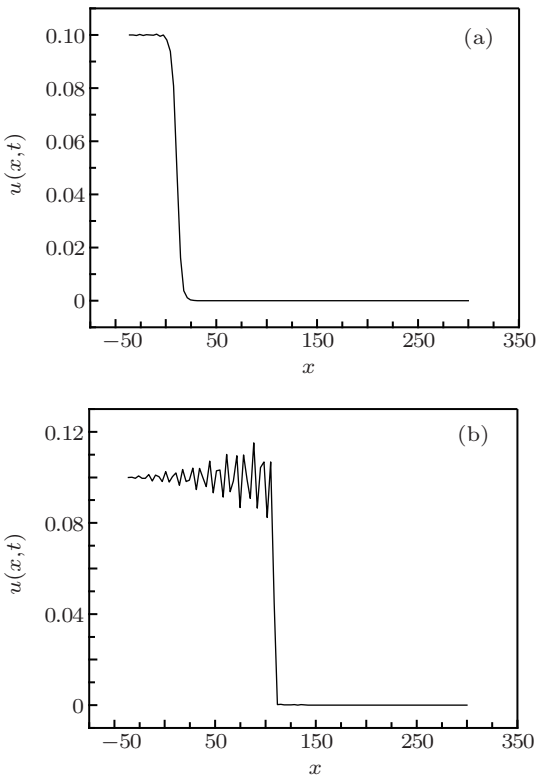


Fig. 2. Undulation profiles for slope $d = 5$ at time $t = 10$ (a) and 100 (b).

From these figures, we can conclude that numerical results derived from the EFG to solve the heat conduction problem are in good agreement with the exact solution.

5. Conclusion

The meshless element-free Galerkin (EFG) method is used to solve the RLW equation. The numerical results, which are in good agreement with the exact solutions, show that the technique is accurate and efficient. No mesh is required in the method, which provides an important advantage over traditional mesh-dependent techniques such as the finite difference method, the finite element method, and the boundary element method, in that it can solve the

problems with irregular geometries. The EFG method should be extended to solve the problems of nonlinear partial differential equations which arise in the theory of solitons and other areas.

References

[1] Donea J and Giuliani S 1974 *Nucl. Eng. Des.* **30** 205
[2] Bathe K J and Khoshgoftaar M R 1979 *Nucl. Eng. Des.* **51** 389
[3] Skerget P and Alujevic A 1983 *Nucl. Eng. Des.* **76** 47
[4] Belytschko T, Krongauz Y and Organ D1996 *Comput. Meth. Appl. Mech. Eng.* **139** 3
[5] Monaghan J J 1988 *Comput. Phys. Commun.* **48** 89
[6] Wendland H 1999 *Math. Comput.* **68** 1521
[7] Belytschko T, Lu Y Y and Gu L 1994 *Int. J. Numer. Meth. Eng.* **37** 229
[8] Liu W K, Jun S and Zhang Y F 1995 *Int. J. Numer. Meth. Eng.* **20** 1081
[9] Cheng R J and Liew K M 2009 *Comput. Mech.* **45** 1
[10] Atluri S N and Zhu T 1998 *Comput. Mech.* **22** 117
[11] Cheng R J and Cheng Y M 2008 *Appl. Numer. Math.* **58** 884
[12] Peregrine D H 1966 *J. Fluid Mech.* **25** 321
[13] Peregrine D H 1966 *J. Fluid Mech.* **27** 815
[14] Dodd R K and Eilbeck J C 1982 *Solitons and Nonlinear Wave Equation* (New York: Academic Press)
[15] Lewis J C and Tjon J A 1979 *Phys. Lett.* A **73** 275
[16] Raslan K R 2005 *Appl. Math. Comput.* **168** 795
[17] Gardner L R T and Gardner G A 1996 *Commun. Numer. Mech. Eng.* **12** 795
[18] Gardner L R T 1990 *J. Comput. Phys.* **91** 441
[19] Sloan D M 1991 *J. Comput. Appl. Math.* **36** 159
[20] Araujo A 2001 *Appl. Numer. Math.* **36** 197
[21] Duran A 2003 *J. Phys.* A: *Math. Gen.* **36** 7761
[22] Bona J L 1994 *J. Nonlinear Sci.* **4** 449
[23] Lu Z and Liu R 1998 *Siam. J. Numer. Anal.* **36** 89
[24] Jain P C and Shanker R 1993 *Commun. Numer. Mech. Eng.* **9** 579
[25] Zaki S I 2001 *Phys. Commun.* **138** 80
[26] EI-Danaf T S and Ramadan M A 2005 *Chaos, Solitons and Fractals* **26** 747
[27] Kaya D 2003 *Chaos, Solitons and Fractals* **17** 869
[28] Shivamoggi B K 2002 *Chaos, Solitons and Fractals* **13** 1129
[29] Ramos J I 1998 *Appl. Math. Comput.* **94** 17
[30] Ramos J I 1999 *Appl. Math. Comput.* **103** 69
[31] Ramos J I 2005 *Appl. Math. Comput.* **163** 1123
[32] Ramos J I 2006 *Appl. Math. Comput.* **179** 622
[33] Cheng Y M and Peng M J 2005 *Sci. Chin.* Ser. G: *Phys. Mech. Astron.* **48** 641
[34] Cheng Y M and Li J H 2005 *Acta Phys. Sin.* **54** 4463 (in Chinese)
[35] Qin Y X and Cheng Y M 2006 *Acta Phys. Sin.* **55** 3215 (in Chinese)
[36] Cheng R J and Cheng Y M 2007 *Acta Phys. Sin.* **56** 5569 (in Chinese)
[37] Dai B D and Cheng Y M 2007 *Acta Phys. Sin.* **56** 597 (in Chinese)
[38] Cheng R J and Cheng Y M 2008 *Acta Phys. Sin.* **57** 6037 (in Chinese)
[39] Cheng R J and Cheng Y M 2008 *Appl. Numer. Math.* **58** 884
[40] Chen Li and Cheng Y M 2008 *Acta Phys. Sin.* **57** 1 (in Chinese)
[41] Chen Li and Cheng Y M 2008 *Acta Phys. Sin.* **57** 5069 (in Chinese)
[42] Ren H P and Zhang W 2009 *Chin. Phys.* B **18** 4065
[43] Cheng R J and Ge H X 2009 *Chin. Phys.* B **18** 4059
[44] Wang J F, Sun F X and Cheng R J 2010 *Chin. Phys.* B **19** 060201
[45] Chen L and Cheng Y M 2010 *Chin. Phys.* B **19** 090202
[46] Cheng R J and Ge H X 2010 *Chin. Phys.* B **19** 090201
[47] Cheng Y M and Li J H 2006 *Sci. Chin.* Ser. G: *Phys. Mech. Astron.* **49** 46
[48] Cheng Y M and Liew K M 2009 *Int. J. Numer. Meth. Eng.* **78** 1258
[49] Gao H F and Cheng Y M 2010 *Int. J. Comput. Meth.* **7** 55
[50] Wang J F and Cheng Y M 2011 *Chin. Phys.* B **20** 030206
[51] Cheng R J and Cheng Y M 2011 *Chin. Phys.* B **20** 070304
[52] Cheng R J and Cheng Y M 2011 *Acta Phys. Sin.* **60** 060206 (in Chinese)
[53] Rubin S G and Graves R A 1975 *Computers and Fluids* **3** 1