The Meshless Kernel-Based Method of Lines for Solving the Equal Width Equation

Yılmaz Dereli[a,1] and Robert Schaback[b]
[a]Anadolu University, Science Faculty,
Mathematics Department, 26470, Eskişehir, Turkey
[b]Angewandte Mathematik (NAM),
Georg-August-Universität Göttingen,
Lotzestrasse 16-18 D-37073, Göttingen, Germany

Abstract

The Equal Width equation governs nonlinear wave phenomena like waves in shallow water. Here, it is solved numerically by the Method of Lines using a somewhat unusual setup. There is no linearization of the nonlinear terms, no error in handling the starting approximation, and there are boundary conditions only at infinity. To achieve a space discretization of high accuracy with only few trial functions, meshless translates of radial kernels are used. In the numerical examples, the motion of solitary waves, the interaction of two and three solitary waves, the generation of wave undulation, the Maxwell initial condition, and the clash of two colliding solitary waves are simulated. Our numerical results compare favourably with results of earlier papers using other techniques.

1 The Equal Width Equation

Nonlinear dispersive waves are of significant importance in physical phenomena such as shallow water waves. The equal width (EW) equation, which was introduced by Morrison et al. [2], is an important special kind of a nonlinear dispersive wave equation. It is defined as

$$u_t + uu_x - \mu u_{xxt} = 0 \tag{1}$$

for a smooth function $u = u(x,t)$ on a domain $\Omega \times [0,T]$ with $\Omega \subseteq \mathbb{R}$. Except for a single travelling solitary wave solution, no analytic solutions are known, and therefore numerical methods have to be used. Obviously, there also is no nontrivial stationary solution.

In various theoretical papers for the case $\Omega = \mathbb{R}$, boundary conditions $u(\pm\infty) = 0$ are posed at infinity, which makes a lot of sense because the differential equation is similar to a nonlinear transport equation showing travelling wave phenomena. In fact, the equation has solutions of the form

$$u(x,t) = 3\,c\,\mathrm{sech}^2((x - x_0 - ct)/\sqrt{4\mu}) \qquad (2)$$

travelling at constant speed c and vanishing at $\pm\infty$. Note that the speed is a third of the amplitude, but arbitrary otherwise. Solitary waves with negative amplitudes travel backwards.

2 Boundary Conditions

The most popular examples consider a localized starting function $u_0(x) = u(x,0)$ consisting of one or more solitary peaks. In numerical calculations, this is usually handled by replacing $\Omega = \mathbb{R}$ with $\Omega = [a,b]$ with zero boundary conditions at both ends. This often leads to the problem that the starting function does not satisfy the boundary conditions exactly. In order to avoid unphysical complications, calculations are usually stopped before the waves reach b, and then the zero boundary condition at b is redundant anyway and should not enter into the numerical technique.

A standard example with **nonzero** boundary conditions is the "undular bore" case [7] showing the development of an increasing wave starting from a nonzero constant boundary value u_a at a. However, in such cases there is an additional compatibility condition that must be observed. In fact, the differential equation then necessarily requires $u_x(a,t) = 0$ for all t, and this condition has to be observed when discretizing the equation. Consequently, the starting function u_0 should satisfy $u_0'(a) = 0$ at least approximatively, if $u(a,t) = u_a(t)$ is prescribed as a nonzero constant function. The same argument applies for constant nonzero boundary values at b if calculations are restricted to a finite interval $[a,b]$.

In general, for $\Omega = [a,b]$ being a finite interval, one can pose boundary conditions

$$\begin{array}{rcll} u(a,t) & = & u_a(t), & t \in [0,T] \\ u(b,t) & = & u_b(t), & t \in [0,T] \\ u(x,0) & = & u_0(x), & x \in [a,b] \end{array}$$

satisfying compatibility conditions

$$u_a(0) = u_0(a),\ u_b(0) = u_0(b).$$

But since the solutions will in all known cases be waves travelling for increasing x, posing boundary conditions at $x = b$ is questionable unless $b = \infty$. Since the undular bore case and all other cases treated in the literature behave like a train of travelling waves, one should take $\Omega = [a, \infty)$.

For the undular bore, one should pose the initial condition $u(a,t) = u_a \neq 0$ together with $u_0'(a) = 0$ and no condition at infinity or at any other spatial point. Most numerical papers do not seem to handle this correctly and explicitly. They assume a large b, require an explicit boundary condition $u(b,t) = 0$ entering into the method and restrict the time interval in such a way that the developing wave does not reach b. Again, this makes conditions at b artificial.

This paper will work with boundary conditions at infinity throughout. We only assume a starting function u_0 for $u(x,0) = u_0(x),\ x \in \mathbb{R}$ which implies the boundary conditions at infinity.

3 Method of Lines

We aim at the Method of Lines (MOL), which leads to a system of ordinary differential equations, and this implies that there will no time discretization at all, and there will be no artificial linearization of the differential equation as in various other papers. The problem of correct time–stepping will be automatically solved by the ODE solver we invoke.

The discretization of the spatial domain will often depend on how the boundary conditions are posed. Here, we take a somewhat unusual approach that has no errors at startup and does not impose finite boundary conditions.

We avoid errors at startup by including the starting function u_0 into the set of trial functions on the spatial domain. Calculations will be confined to a spatial interval $[a,b]$, and they will always start with the restriction of u_0 to $[a,b]$. We discretize $[a,b]$ as

$$a = x_1 < \ldots < x_N = b \tag{3}$$

and, in addition to u_0, we use a set of smooth spatial functions

$$u_j \;:\; \mathbb{R} \to \mathbb{R},\; 1 \le j \le n$$

vanishing at infinity. This can be done by picking the functions u_j from a suitable space of trial functions, e.g. B–splines or translates of positive definite radial basis functions like the Gaussian or Wendland functions.

In general, the Method of Lines uses time–dependent coefficients for a linear combination of spatial trial functions. Here, we approximate the solution u by a linear combination

$$v(x,t) = u_0(x) + \sum_{j=1}^{n} \alpha_j(t) u_j(x)$$

with smooth functions α_j on $[0,T]$, $1 \le j \le n$. Then

$$\begin{aligned} v_x(x,t) &= u_0'(x) + \sum_{j=1}^{n} \alpha_j(t) u_j'(x) \\ v_t(x,t) &= \sum_{j=1}^{n} \alpha_j'(t) u_j(x) \\ v_{xxt}(x,t) &= \sum_{j=1}^{n} \alpha_j'(t) u_j''(x). \end{aligned}$$

At time $t = 0$ we impose the conditions $\alpha_j(0) = 0$, $1 \le j \le n$ to satisfy the initial condition $v(x,0) = u_0(x)$ on all of $[a,b]$ without introducing any errors at startup.

We put our trial functions into the differential equation

$$\begin{aligned} 0 &= v_t + v v_x - \mu v_{xxt} \\ &= \sum_{j=1}^{n} \alpha_j'(t) u_j(x) \\ &\quad + \left(u_0(x) + \sum_{j=1}^{n} \alpha_j(t) u_j(x) \right) \left(u_0'(x) + \sum_{j=1}^{n} \alpha_j(t) u_j'(x) \right) \\ &\quad -\mu \sum_{j=1}^{n} \alpha_j'(t) u_j''(x). \end{aligned}$$

Evaluating this at points x_k and sorting the derivatives of the α_j to the left, we get

$$\begin{aligned} & \sum_{j=1}^{n} \alpha_j'(t)(\mu u_j''(x_k) - u_j(x_k)) \\ = & \left(u_0(x) + \sum_{j=1}^{n} \alpha_j(t) u_j(x_k) \right) \left(u_0'(x) + \sum_{j=1}^{n} \alpha_j(t) u_j'(x_k) \right) \end{aligned} \tag{4}$$

for $1 \leq k \leq N$. This is an implicit system of first–order ordinary differential equations.

4 Implementation

If we introduce suitable column vectors and matrices into the system (4), we have to satisfy

$$(\mu U'' - U) * \alpha'(t) \quad = \quad (u_0 + U * \alpha(t)). * (u_0' + U' * \alpha(t)) \tag{5}$$

in MATLAB notation for the pointwise product $.*$ between two matrices or vectors of the same shape. The necessary matrices and vectors are

$$\begin{array}{rcl} \alpha(t) & := & (\alpha_1(t), \ldots, \alpha_n(t)) \in \mathbb{R}^n \\ u_0^T & := & (u_0(x_1), \ldots, u_0(x_N)) \in \mathbb{R}^n \\ (u_0')^T & := & (u_0'(x_1), \ldots, u_0'(x_N)) \in \mathbb{R}^n \\ U & := & (u_j(x_k))_{1 \leq k \leq N,\, 1 \leq j \leq n} \in \mathbb{R}^{N \times n} \\ U' & := & \left(u_j'(x_k)\right)_{1 \leq k \leq N,\, 1 \leq j \leq n} \in \mathbb{R}^{N \times n} \\ U'' & := & \left(u_j''(x_k)\right)_{1 \leq k \leq N,\, 1 \leq j \leq n} \in \mathbb{R}^{N \times n} \end{array}$$

where j is the column index and k is the row index. The system (5) of linear equations for $\alpha'(t)$ will have N equations and n unknowns. It is allowed to take N larger than n, and then the system should be solved in a least-squares sense. The matrix

$$B := \mu U'' - U \tag{6}$$

of the left–hand side is time–independent. Via a QR factorization or a singular-value decomposition of B which is calculated once and for all, one can get a pseudo-inverse $B^\dagger$ of B, and the system (5) can at each time be solved approximately via

$$\alpha'(t) = B^\dagger * ((u_0 + U * \alpha(t)). * (u_0' + U' * \alpha(t)))\,.$$

This is the ODE system generated by our version of the Method of Lines, and one can invoke any ODE integrator to solve it. In case $N = n$ and invertibility of B, the approximate solution $v(x, t)$ will satisfy the differential equation at all points $x_1, \ldots, x_n$ and all times, the latter within the accuracy limit of the ODE integrator.. The starting vector will be

$$\alpha(0) := (0, \ldots, 0) \in \mathbb{R}^n.$$

Note that the nonlinearity of the PDE is preserved, and a good ODE solver will automatically use a reasonable time–stepping and detect stiffness of the ODE system. At each invocation of the right–hand side, there are 3 matrix–vector products to be calculated, leading to a computational complexity of $\mathcal{O}(N \times n)$ for each evaluation. But if compactly supported trial functions are used, the matrices U and U' will be sparse, reducing the computational complexity.

5 Meshless Space Discretization

It is still open how the additional trial functions $u_1, \ldots, u_n$ should be chosen. In principle, any choice of twice differentiable functions vanishing at infinity could be considered. The functions should be able to model how the solution develops over time out of the starting function u_0, and in order to make the matrices U, U', U'' sparse they should be compactly supported or quickly decaying towards infinity.

Since they have good approximation properties, we use translates

$$u_j(x) := \phi(|x - y_j|),\ 1 \le j \le n$$

of radial basis functions $\phi : [0, \infty) \to \mathbb{R}$ centered at n points $y_1, \ldots, y_n$ with

$$a \le y_1 < y_2 < \ldots < y_n \le b. \tag{7}$$

These functions can be evaluated easily and need no regular distribution of the discretization points y_j. In view of getting a square matrix B in (6) one can choose $N = n$ and take the points y_j to be equal to the x_j of (3) as long as B has a reasonable inverse. So far, we did not run into problems with taking $N = n$, but there might be cases where one should take more test points x_j than trial points y_k, i.e. taking N larger than n. This will have a stabilizing effect.

There are a lot of radial basis functions in the literature, and we ran the Method of Lines with a variety of them, including the Gaussian, inverse Multiquadrics, Wendland's functions [5] and the Whittle–Matèrn kernel generating Sobolev spaces. The results were not much different, but, as usual with kernels, the scaling needed some careful consideration in each example. According to the guidelines of [15], small scales lead to good condition and large errors, while large scales imply bad condition and small errors, a good compromise being achievable by picking a scaling with quite a large, but still feasible condition of the basic matrices. In order to determine the optimal scaling, we perform a condition number estimation of the kernel matrix with entries $u_j(y_k),\ 1 \leq j, k \leq n$ before we actually start to set up the algorithm. It is an important unsolved problem to find a method to determine the optimal value of the scale.

6 Numerical Examples

In this section, numerical solutions of the Equal Width equation will be presented by using the method described above. In order to confirm the accuracy and efficiency of the Method of Lines with its special treatment of initial values and boundary conditions, a number of test problems was chosen.

In all examples, we fixed a spatial evaluation interval $[a, b]$ without posing boundary conditions at the endpoints, and used a trial discretization (3) with $n = 200$ points and a test discretization (7) with $N = 500$ points. Experiments showed that one can get good results already with less test points, but we wanted to maintain a good graphical accuracy without having to use additional interpolation techniques for generating intermediate values of the solution. Since the choice of the interval $[a, b]$ fixes the domain where calculations can be expected to be reasonable, one must make sure that time–dependent calculations are stopped whenever the wave phenomena run outside of $[a, b]$. If this should happen, users should simply enlarge the calculation interval.

All tests were carried out as instant MATLAB movies, in order to study the dynamic behavior of the wave phenomena. Programs can be obtained from the authors on request. The effect of choosing different radial basis functions was not serious. In many cases, there are time invariants (see

[8, 9, 10, 11, 12])

$$C_1 := \int_a^b u(x,t)dx,\ C_2 := \int_a^b (u^2(x,t) + \mu u_x^2(x,t))dx,\ C_3 := \int_a^b u^3(x,t)dx \quad (8)$$

and by sufficiently fine spatial resolution it is no problem to maintain these invariants to reasonable accuracy. Wherever possible, we support this statement by providing numerical rsults.

6.1 Single Solitary Wave

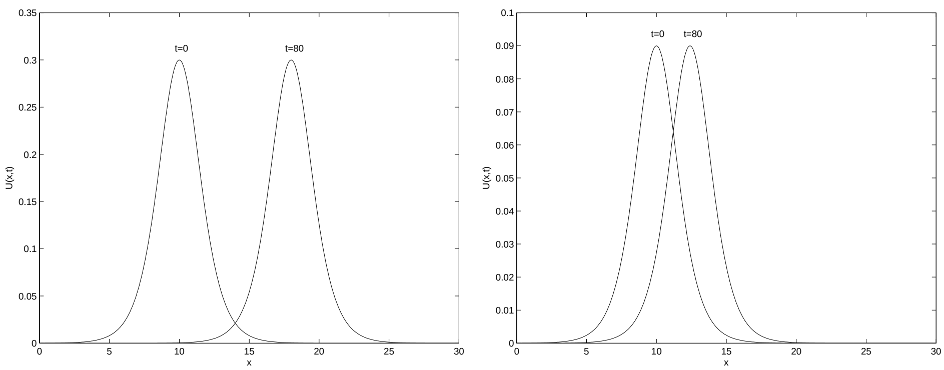

Figure 1: Movement of single soliton for $c = 0.1$ and $c = 0.03$

	C_1	C_2	C_3
analytic, c=0.1	1.20000000	0.028800000	0.0005760000
numeric, c=0.1	1.19999799	0.028800005	0.0005760001
analytic, c=0.03	0.36000000	0.025920000	0.001555000
numeric, c=0.03	0.36000134	0.025912000	0.001555200

Table 1: Invariants for single soliton

The Equal Width equation (1) has the solitary wave solution (2) with boundary conditions $U \to 0$ as $x \to \pm\infty$. We worked on $[a, b] = [0, 30]$ and $[0, T] = [0, 80]$ using the starting function $u_0(x)$ of the form (2) with parameters $\mu = 1, x_0 = 10$, $c = 0.03$ and $c = 0.1$. The results are in Figure 1 using the Gaussian kernel, and agree with [8, 9, 10, 11, 12] using a linearization of the differential equation, a time–stepping technique, and finite boundary conditions. Table 1 gives the theoretical [7] and numerical values of the invariants (8) . Using other kernels lead to similar results.

It should be noted that the soliton itself is a positive definite kernel. Thus its translates could be used in the Method of Lines, but we leave this to future research.

6.2 Two Solitary Waves

As a second test problem for the Equal Width equation, we chose the interaction of two solitary waves, as in the cited literature. The initial function is

$$\begin{aligned} u_0 &= U_1 + U_2 \\ U_j &= 3c_j \text{sech}^2(k_j(x - \tilde{x}_j - c_j)), \quad , \quad j = 1, 2 \end{aligned} \tag{9}$$

with $\mu = 1$, $k_1 = 0.5$, $k_2 = 0.5$, $\tilde{x}_1 = 10$, $\tilde{x}_2 = 25$, $c_1 = 1.5$ and $c_2 = 0.75$. Calculation was done on $[a, b] = [-10, 70]$ and $[0, T] = [0, 30]$ using the Gaussian kernel, and with the result of Figure 2. The higher wave travels faster, passes the smaller one, and proceeds, without change of shape of both. Table 2 provides the invariants.

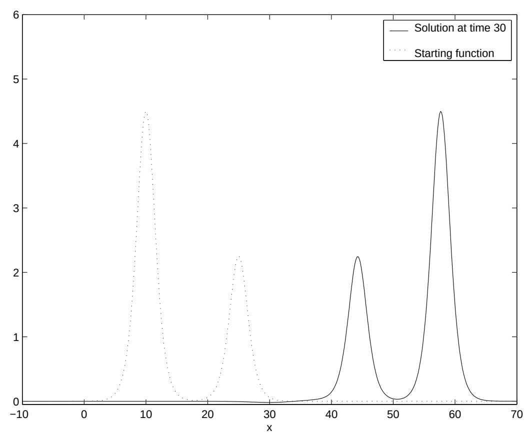

Figure 2: Movement of two solitons

	C_1	C_2	C_3
analytic	27.000000	81.000000	218.700000
numeric	27.000046	81.000247	218.702058

Table 2: Invariants for two solitons

6.3 Three Solitary Waves

Now we take

$$u_0(x) = \sum_{i=1}^{3} 3c_i \text{sech}^2(k_i(x - \tilde{x}_i - c_j)) \qquad (10)$$

where $k_i = 0.5$, $c_1 = 4.5$, $\tilde{x}_1 = 10$, $c_2 = 1.5$, $\tilde{x}_2 = 25$, $c_3 = 0.5$, $\tilde{x}_3 = 35$. The result on $[a, b] = [-10, 100]$ and for $T = 15$ is in Figure 3. Again, the larger waves pass the smaller ones and leave them travelling along, unchanged in form, at their lower speed. The invariants are in Table 3.

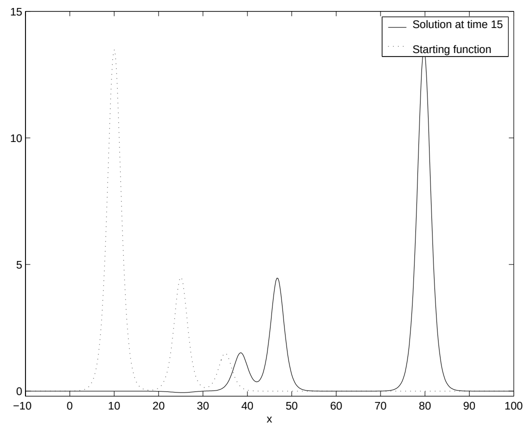

Figure 3: Movement of three solitons

6.4 The Undular Bore

Finally, the development of an undular bore is studied by using the following initial function

$$u_0(x) = 0.05 \left(1 - \tanh\left(\frac{x}{2}\right)\right) \qquad (11)$$

	C_1	C_2	C_3
analytic	78.000000	655.2000000	5450.400000
numeric	78.000019	655.2080531	5450.216822

Table 3: Invariants for three solitons

on $[a, b] = [-20, 50]$ up to time $T = 800$. This example is comparable to [7, 8, 9, 10, 11] and gives Figure 4. The stimulated waves move to the right, while their starting point moves slowly to the left. The expressions in (8) are

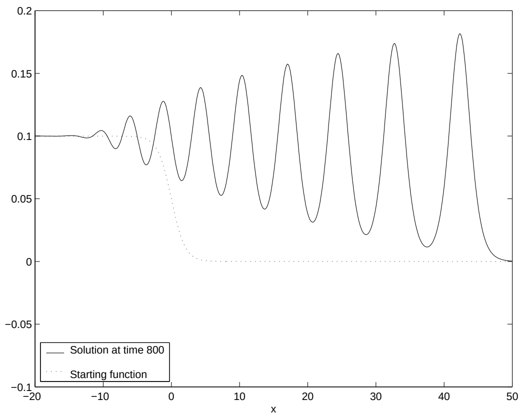

Figure 4: Undular bore

not constant over time now. Instead, their time-derivative is constant, and this behavior could be verified numerically to good accuracy.

6.5 The Maxwell Wave

Another standard case is to look at what happens if the starting function is a Gaussian

$$u_0(x) = 0.05 \exp(-(x - 20)^2/25).$$

The analytic result is not known, and the numerical result on $[a, b] = [0, 50]$ and $T = 500$ is in Figure 5. The single starting wave leads to a train of waves of smaller and smaller amplitudes and speeds.

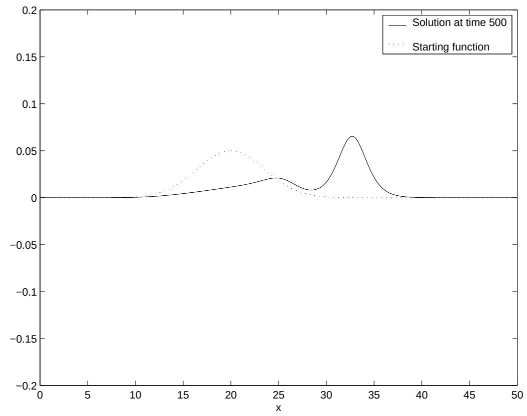

Figure 5: Wave from Gaussian

6.6 The Clash

We now let two solitary waves of exactly the same form but different signs move towards each other. When they meet, they form a singularity which emits trains of smaller waves to both sides, while the singularity gradually vanishes over time, see Figure 6. Due to the singularity in the clash, this example is much harder to calculate, and therefore we used $N = n = 1001$. We encourage readers to try other methods on this case.

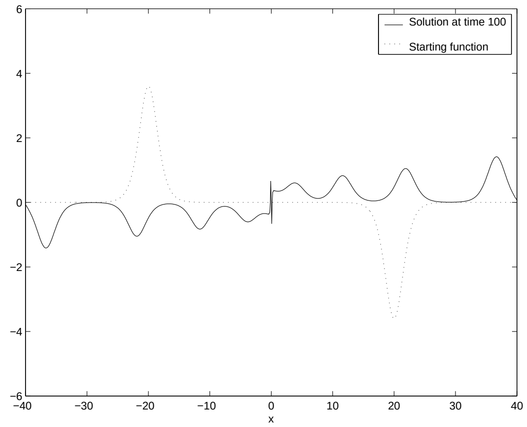

Figure 6: The Clash

7 Conclusion

In this study, the meshless kernel based Method of Lines has been used to obtain numerical solutions of the Equal Width equation without artificial linearization and artificial boundary conditions. The results compare favourably with the available literature, are easy to program and run fast enough to allow a thorough study of the dynamic behavior of the solutions.

8 Acknowledgement

Yılmaz Dereli's research is supported by The Scientific and Technological Research Council of Turkey (TÜBİTAK).

References

[1] D. H. Peregrine, Calculations of the development of an undular bore, J. Fluid. Mech. **25(2)**, 321-330, (1966).

[2] P.J. Morrison, J.D. Meiss, J.R. Carey , Scattettring of RLW solitary waves, Physica **11D**, 324-336, (1984).

[3] W. Zongmin, Hermite-Birkhoff interpolation of scattered data by radial basis functions, Approx. Theory & its Appl. **8(2)**, 1-10, (1992).

[4] R.L. Hardy, Multiquadric equations of topography and other irregular surfaces, Geo. Phys. Res., **176**, 1905-1915, (1971).

[5] H. Wendland, Piecewise polynomial positive definite and compactly supported radial functions of minimal degree, Advanvances in Comput. Math., **4**, 389-396, (1995).

[6] P.J. Olver, Euler Operators and Conservation Laws of the BBM Equation, Math. Proc. Camb. Philos. Soc., **85**, 143-159, (1979).

[7] L.R.T. Gardner, G.A. Gardner, F.A. Ayoub and N.K. Amein, Simulations of the EW undular bore, Commun. Numer. Meth. Eng. **13**, 583-592, (1997).

[8] B. Saka, İ. Dağ, Y. Dereli and A. Korkmaz, Three different methods for numerical solution of the EW equation, Engrg. Anal. Boun. Elem. **32**, 556–566, (2008).

[9] Saka, B., A finite element method for equal width equation, Appl. Math. and Comput., **175**, 730-747, (2006).

[10] Esen, A., A numerical solution of the equal width wave eqution by a lumped Galerkin method, Appl. Math. Comput., **168**, 270-282, (2004).

[11] İ. Dağ, B. Saka, A cubic B-spline collocation method for the EW equation, Math. Comput. Appl., **90**, 381–392, (2004).

[12] Doğan, A., Application of Galerkin's metod to equal width wave equation, Appl. Math. Comput., **160**, 65-76, (2005).

[13] K.R.Raslan, A computational method for the equal width equation, Int. J. Comp. Math. **81**, 63-72, (2004).

[14] K.R.Raslan, Collocation method using quartic B-spline for the equal width (EW) equation, Appl. Math. Comput., **168**, 795-805, (2005).

[15] Schaback, R. and Wendland, H., Kernel techniques: from machine learning to meshless methods, Acta Numerica **15**, 543–639 (2006)