A RADIAL BASIS FUNCTION FINITE DIFFERENCE SCHEME FOR THE BENJAMIN-ONO EQUATION [*]

BENJAMIN F. AKERS[†], TONY LIU[‡], AND JONAH A. REEGER[§]

Abstract. A Radial Basis Function – Finite Differencing (RBF-FD) scheme is applied to the initial value problem of the Benjamin-Ono equation. The Benjamin-Ono equation has traveling wave solutions with algebraic decay and a nonlocal pseudo-differential operator, the Hilbert transform. When posed on $\mathbb{R}$, the former makes Fourier collocation a poor discretization choice, the latter is challenging for any local method. We develop an RBF-FD approximation of the Hilbert transform, and discuss the challenges of implementing this and other pseudo-differential operators on unstructured grids. Numerical examples, simulation cost, convergence rates, and generalizations of this method are all discussed.

Key words. Radial basis functions, finite-difference methods, traveling waves, non-uniform grids.

subject classifications. 35B35, 76B15

1. Introduction

In this paper we use the Benjamin-Ono equation as a test-bed for new Radial Basis Function – Finite Difference (RBF-FD) simulations of nonlocal wave equations on non-uniform grids. The Benjamin-Ono equation presents the numerical challenges of numerical stiffness, a nonlocal pseudo-differential operator, and localized traveling solutions with slow decay. The equation,

$$u_t - \mathcal{H}u_{xx} + uu_x = 0, \tag{1.1}$$

has many exact known solutions. For example, on $\mathbb{R}$ equation (1.1) supports traveling solitary waves solutions,

$$u(x,t) = \frac{4}{(x-t)^2+1}.$$

The Benjamin-Ono equation is known to be well-posed [1] and integrable. It can be solved with inverse scattering and many exact solution profiles are known [2, 3]. It has been numerically simulated many times, both in the periodic setting [4] and on $\mathbb{R}$ [5, 6].

In this work we develop an RBF-FD scheme for the Benjamin-Ono equation. Common practice for simulation of (1.1) on $\mathbb{R}$ is to use periodic boundary conditions, allowing for Fourier collocation, on a large spatial domain [7]. Global radial basis functions (RBFs) have been used as a basis set for simulation of Benjamin-Ono [8]; instead, this work is the first example of the application of RBF-FD to this model. In many cases, RBFs allow for high orders of accuracy while taking advantage of non-uniform spacing in the node set when approximating linear operators. There are an increasing number of texts discussing the use of RBFs in the approximation of differential operators (see e.g., [9, 10]) while presenting much of their history and

[*] Received date / Revised version date

[†] Department of Mathematics and Statistics, Air Force Institute of Technology, Dayton, OH, (Benjamin.Akers@afit.edu).

[‡] Department of Mathematics and Statistics, Air Force Institute of Technology, Dayton, OH, (Tony.Liu@afit.edu).

[§] Unaffiliated

theory in detail. Recently the concept of RBF-FD has been further extended to the approximation of definite integrals–first over smooth surfaces [11, 12, 13] and then over the volume of the ball [14]. In this paper we look at an extension of RBF-FD now to pseudo-differential operators. The method presented in this paper utilizes as a basis for approximation the so-called Polyharmonic Spline RBF augmented by shifted monomials. In this case, [15] explains that if the shifted monomials up to degree m are included in the basis for approximation, then all of the terms in the Taylor series up to degree m will be handled exactly for the function being interpolated. Therefore, for functions with rapidly decaying terms in the Taylor series, the linear operator will be applied to an approximation with $O(\Delta x^m)$ accuracy on a node set with step size Δx.

RBF-FD simulation of the Benjamin-Ono equation presents the challenge of creating a local approximation of a non-local pseudo-differential operator–the Hilbert transform. The process used herein is generalizable to other pseudo-differential operators, but is the primary cost of the method as it requires diagonalization of an RBF-FD differentiation matrix. Further, the spectrum of the RBF-FD differentiation matrix, particularly when constructed on non-uniform spaced node sets, can often include spurious eigenvalues (for example with positive real part when approximating an operator with pure imaginary spectrum), similar to those observed in [16, 17]. We observe that these spurious eigenvalues are the result of floating point errors due to the conditioning of the RBF-FD discretization of the linear operators.

Another complication is the slow decay of the solution as $|x|$ increases. To deal with this complication and with a localized steep gradient in the solutions, we increase the point density where the wave amplitude is large and decrease point density in the far field. This allows for increased accuracy over uniform grids with the same number of nodes. Consideration of non-uniform node sets is a key advantage of RBF based approximations. Even in the context of approximating a non-local operator with local approximations and slowly decaying solutions, we demonstrate $O(\Delta x^m)$ accuracy where Δx is the smallest spacing between adjacent points in a node set. We report errors based on this smallest step size, rather than the largest or the number of nodes as in [18, 19], because the mapping we use to refine the node set both decreases the step size near important features of the solution and increases the large step sizes elsewhere while keeping total number of number of nodes fixed. To further illustrate this method, we present a brief example in another model equation [20].

In the process simulating the Benjamin-Ono equation, we present a simple framework for using RBF-FD to approximate pseudo-differential operators. The procedure extends the applicability of RBF methods beyond the purely differential equations previously simulated, see [21, 16], to a host of other pseudo-differential model equations, e.g Whitham [22], Akers-Milewksi [20], many more [23, 24, 25]. Many of these pseudo-differential equations exhibit coherent structures which are computed with quasi-Newton iteration [26, 27]. The simulation of the dynamics near these coherent structures is the application where we believe this method will be most useful. The diagonalization cost required to approximate the pseudo-differential operators in our simulations (a pre-processing step) is comparable to the cost of the quasi-Newton iteration already being done to compute these waves [28].

The paper is organized as follows. Section 2 describes the RBF-FD based numerical method for simulating the Benjamin-Ono equation. This includes a discussion of RBF-FD, a node placement strategy, approximation of a pseudo-differential operator, and the time-stepping scheme. Then, section 3 presents numerical results when applying the method to both the Benjamin-Ono equation and the Akers-Milewski equation. Finally, section 4 draws some conclusions about the use of these methods for approximation pseudo-differential operators.

2. Numerical Method

The numerical method begins in the familiar way of partitioning an interval using N subintervals. The endpoints of these subintervals are the set of points $\mathcal{S}_N = \{x_k\}_{k=1}^N$. In this paper periodic boundary conditions on a large domain are imposed as a proxy for the slow decay of the solution as $|x| \to \infty$. To implement these boundary conditions the method creates two periodic images of the set $\mathcal{S}_N$. These are defined by $\mathcal{S}_N^{\pm} = \left\{x_k^{\pm}\right\}_{k=1}^N := \{x_k \pm L\}_{k=1}^N$, where the signs each define a separate set and L is the period. Considering a point $x_k \in \mathcal{S}_N$, define $\mathcal{N}_k^n = \{x_{k,j}\}_{j=1}^n$ to be the set of n points in $\mathcal{S}_N \bigcup \mathcal{S}_N^+ \bigcup \mathcal{S}_N^-$ nearest to x_k. The proposed method approximates the application of a linear operator $\mathcal{L}$ to $u : \mathbb{R} \mapsto \mathbb{R}$ by interpolating u over the points in $\mathcal{N}_k^n$ and then applying the linear operator to the interpolant. Traditional finite difference methods are defined in this way, where a polynomial interpolant of degree $n-1$ is constructed over the set x_k and $n-1$ prescribed neighbors and the linear operator is applied to the interpolant. The method proposed here, however, utilizes local RBF interpolation. RBF interpolation has been used successfully in the approximation of differential operators over subsets of scattered data through the concept of RBF-FD in, for instance, [29, 30, 31, 32].

2.1. RBF-FD Weight Calculations for Linear Operators

For the simplicity of discussion consider approximating $\mathcal{L}$ applied to u at a point $x_k \in \mathcal{S}_N$. That is, consider

$$\mathcal{L}_k(u) := \mathcal{L}u(x)|_{x=x_k}.$$

Following common RBF/RBF-FD procedures the interpolant is constructed via

$$s_k(x) := \sum_{j=1}^n c_{k,j}\phi(|x - x_{k,j}|) + \sum_{l=0}^m d_{k,l}(x - x_k)^l$$

with ϕ a function dependent only on the distance from the point $x_{k,j}$. Note that the shift in the monomial terms is included for numerical stability when inverting the matrix A_k in what follows. The interpolation coefficients, $c_{k,j}$, $j = 1,2,\ldots,n$, and $d_{k,l}$, $l = 0,1,\ldots,m$, are chosen to satisfy the interpolation conditions $s(x_{k,j}) = u(x_{k,j})$, $j = 1,2,\ldots,n$, along with constraints $\sum_{j=1}^n c_{k,j}(x_{k,j} - x_k)^l = 0$, for $l = 0,1,\ldots,m$. By applying $\mathcal{L}$ to the interpolant and then evaluating at x_k we wish to reduce the desired approximation to

$$\mathcal{L}_k(u) \approx \sum_{i=1}^N w_{k,i} u(x_{k,i}). \tag{2.1}$$

A simple derivation can be carried out to show that the weights can be found by solving the system of linear equations $A_k \mathbf{v}_k = \mathbf{b}_k$. This system of equations includes the $(n+m+1) \times (n+m+1)$ matrix

$$A_k = \begin{bmatrix} \Phi_k^T & P_k \\ P_k^T & 0 \end{bmatrix},$$

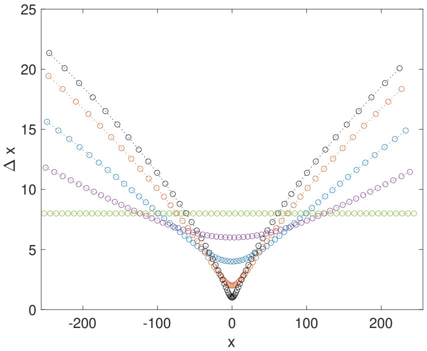

FIG. 2.1. *The step sizes from* (2.2) *as a function of* x*. The horizontal equi-spaced case is* $a=0$*. The step size becomes increasing variable as* a *ascends through the samples* $a=1,2,3,$ *and* 3.5 *(the extreme graph). The step size near the origin vanishes as* $a\to 4$*. The domain has length* $L=512$*, with* $N=64$ *points.*

where $\Phi_{k,ij}=\phi(|x_{k,i}-x_{k,j}|)$ and $P_{k,il}=(x_{k,i}-x_k)^l$, for $i,j=1,2,\ldots,n$ and $l=0,1,\ldots,m$ [31, Section 5.1.4]. The right hand side is the length $n+m+1$ column vector

$$\mathbf{b}_k=\left[\,\mathcal{L}_k\left(\phi(|x-x_{k,1}|)\right)\;\mathcal{L}_k\left(\phi(|x-x_{k,2}|)\right)\;\cdots\;\mathcal{L}_k\left(\phi(|x-x_{k,n}|)\right)\;\mathcal{L}_k(\pi_0)\;\mathcal{L}_k(\pi_1)\;\cdots\;\mathcal{L}_k(\pi_m)\,\right]^T.$$

The system of linear equations is uniquely solvable in our present context [33, Theorem 8.21], and the weights $w_{k,i}$, $i=1,2,\ldots,n$, are the first n entries of the solution vector $\mathbf{v}_k$.

It is typical to approximate the action of $\mathcal{L}$ on u at each point in $\mathcal{S}_N$ simultaneously through the product $D\mathbf{u}$, where D is an $N\times N$ matrix and

$$\mathbf{u}=\left[\,u(x_1)\;u(x_2)\;\cdots\;u(x_N)\,\right]^T.$$

The entries of D are found row by row (something that is easily parallelized), so that

$$D_{ki}=\begin{cases} w_{k,j} & \text{if } x_{k,j}=x_i \text{ or } x_{k,j}=x_i\pm L \\ 0 & \text{otherwise} \end{cases},$$

that is, entry i of row k is nonzero only if x_i (one of the points in $\mathcal{S}_N$) or one of its periodic images, $x_i\pm L$, appears in the set $\mathcal{N}_k^n$ of the points nearest x_k.

2.2. Node placement

To construct node sets with non-uniform spacing that take advantage of the features of the solution, we first create a spatial node set with equal spacing $\tilde{x}$ on the domain $[-L/2,L/2]$. We then apply the below nonlinear transformation,

$$x=\tilde{x}\left(1+a/L^2(\tilde{x}-L/2)(\tilde{x}+L/2)\right). \tag{2.2}$$

The parameter a dictates the variation in step size in this transformation. For $a=0$, the node set has uniform spacing, for $a=4$ the transformation degenerates by making the step size near the origin equal to zero. The transformation in equation

(2.2) is designed to preserve the overall domain length. It is by no means the only transformation which places a larger density of points near the origin and fewer far from the origin. We also ran experiments with the generalization

$$x=\tilde{x}\left(1+a/L^2(\tilde{x}-L/2)(\tilde{x}+L/2)\right)^p. \tag{2.3}$$

for $p=2,3,4$. Increasing either p or a causes increased node density near the origin, however it also leads to issues with the spectrum of the approximation of the linear operators, as we discuss in next section. For the numerical results presented in this work, we used only (2.2); examples of step-sizes for a sampling of a values in equation (2.2) are in Figure 2.1.

2.3. Approximating the Hilbert Transform In this work we use degree seven polyharmonic splines as the radial basis functions, complemented with polynomials as described above, so that

$$\phi(r)=r^7.$$

These are a common choice for RBF-FD, [10], but make the computation of pseudo-differential operators like the Hilbert transform less straightforward than some other RBFs [8]. The classic procedure to approximate a linear operator using RBF's includes a step where the linear operator is exactly applied to the basis function. This has been done in a previous RBF based numerical study of the Benjamin-Ono equation using gaussian basis functions on which the Hilbert transform can be exactly calculated [8]. For polyharmonic splines an exact formula for the Hilbert transform is unavailable. Instead, we use ideas motivated by the Fourier transform definition of the Hilbert transform,

$$\widehat{\mathcal{H}u}=-i\ \mathrm{sign}(k)\hat{u}.$$

The Fourier transform definition reveals a relationship between the spectrum of the derivative operator, $\lambda_{\partial x}=ik$, and the spectrum of the Hilbert transform $\lambda_{\mathcal{H}}=-i\mathrm{sign}(k)$, so that

$$\lambda_{\mathcal{H}}=-i\ \mathrm{sign}(Im(\lambda_{\partial x})). \tag{2.4}$$

Using this relationship, if the spectrum of the derivative operator is known, then the spectrum of the Hilbert transform can be computed by an algebraic transformation of the spectrum of the derivative. We use this transformation to compute an approximation to the Hilbert transform by first computing a matrix which approximates the derivative as described in section 2.1. Next we diagonalize this matrix using QR iteration to get its eigenvalues $\lambda_{\partial x}$ and eigenvectors. The Hilbert transform can then be approximated using the same eigenvectors and equation (2.4) applied to the computed $\lambda_{\partial x}$ to get its eigenvalues. This computation of the Hilbert transform, a pre-processing step, costs $O(N^3)$ flops, where N is the number of spatial points. The time-stepping portion of the RBF-FD method is $O(N^2)$ flops per step, due to the full matrix that approximates the Hilbert transform. The resulting method however, allows for non-uniform point spacing, so that larger domains can be accurately simulated with a fixed number of points. We expect this method to be competitive only in cases where there are fine local features of interest and slow decay to a remote boundary, as in the Benjamin-Ono solitary wave on $\mathbb{R}$. We admit that the Benjamin-Ono equation has special properties, i.e. integrability, which make other simulation methods like

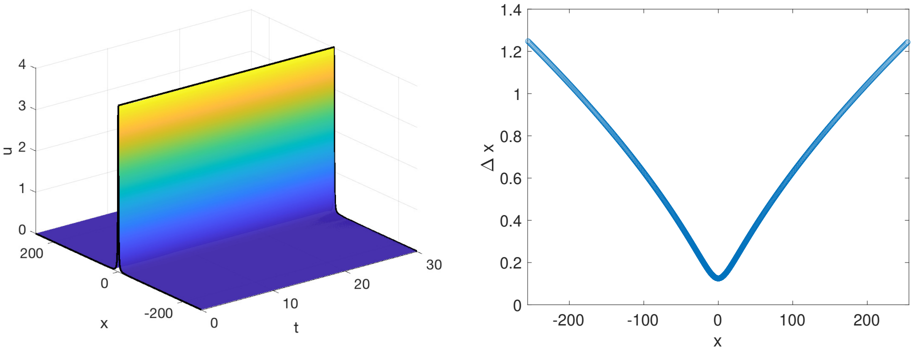

Fig. 2.2. **Left:** *The soliton solution propagated on a domain* $x \in [-256, 256]$. **Right:** *The step-sizes used for the RBF-FD discretization used in the left simulation.*

inverse scattering available [3]. The above algorithm makes no use of integrability, thus can be trivially extended to other equations with pseudo-differential operators like the Intermediate-Long Wave [25], Whitham [22] or Akers-Milewski equations [20] (see Figure 3.3).

2.4. Time stepping

For the time evolution of the system of differential equations induced by the RBF-FD spatial discretization of (1.1), we use a second order IMEX method[34],

$$\frac{3u^{q+1} - 4u^q + u^{q-1}}{2\Delta t} = 2f(u^q) - f(u^{q-1}) + g(u^{q+1}), \tag{2.5}$$

in which $f(u)$ is the nonlinear term (explicitly treated) and $g(u)$ is the linear term (implicitly treated). This method is sometimes called SBDF [34], or extrapolated GEAR [35]. The linear stability region for this scheme is exterior to an egg-shaped region in the right half plane; it is unconditionally stable for wave equations like Benjamin-Ono, which have pure imaginary linear spectrum. We chose this method over the competing IMEX scheme CNAB (Crank-Nicholson Adams-Bashforth) so that small real perturbations of pure imaginary spectrum do not leave the stability region (as is the case for the CNAB). As we will see later, floating point errors due to the condition number of the matrices involved in RBF-FD discretization of the linear operators can cause changes in the spectrum of the approximation of the linear operators involved. We chose (2.5) to be as robust as possible to such errors.

The method we used to approximate the linear operators begins with an eigenvalue calculation. As such, other numerical time steppers are available, for example exponential time differencing (ETD) [36] or integrating factors (IF) [37]. Since the equation is nonlinear, after diagonalizing the linear part, both ETD and IF methods require full matrix multiplications to evaluate the nonlinearity at each time step. The IMEX method used here has the inversion of a full matrix (for the implicit linear term), however this can be done as a pre-processing step, so it also has cost which scales like a full matrix multiplication per time step (and thus is comparable with ETD or IF). We chose the IMEX scheme due to its unconditional stability, to ameliorate the stiffness of this equation in explicit time steppers and to be robust to perturbations of the spectrum off the imaginary axis. In future work we plan to explore

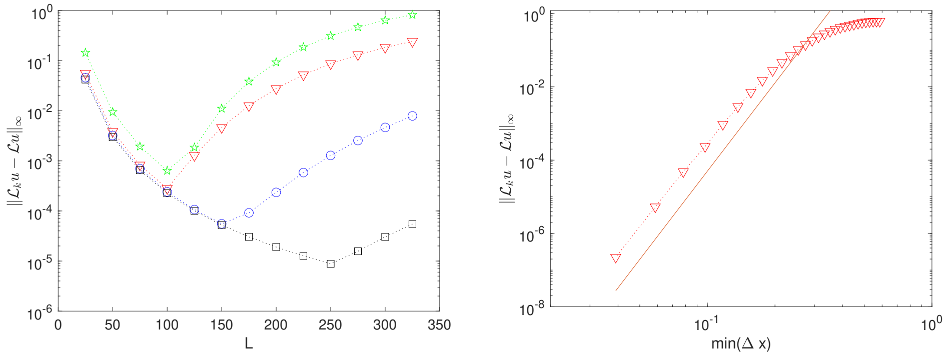

FIG. 3.1. *The infinity norm of error of the numerical approximation of the linear operator, $\mathcal{L}_N u$ against the exact $\mathcal{L}u$ on the solitary wave $u(x) = 4/(1+x^2)$ is depicted.* ***Left:*** *The error's dependence on domain size, L, is compared for the RBF-FD discretization using the spatial points in* (2.2) *with $a=2$ (red triangles), $a=3$ (blue circles), and $a=3.5$ (black squares), and an equi-spaced Fourier discretization (green stars). All of these computations use $N=512$.* ***Right:*** *The error dependence on the minimal step size for $N=1024, L=800$, altering the step size with the parameter 'a'. The numerical results are marked with triangles; the continuous line marks $O(\Delta x^8)$.*

other time steppers, including higher order IMEX methods with unbounded stability regions [34].

3. Results

In this section the numerical method described herein is evaluated via a number of numerical tests. We observe the effects of the numerical discretization, truncation error, and the effect of truncating $\mathbb{R}$ to finite length. Errors are measured as a function the minimal step size of our non-uniform parameterization. The effect of the non-uniform step-size on the spectrum of approximation to the linear operator is also discussed. Example simulations of algebraically decaying Benjamin-Ono solitary wave are shown, as well as an oscillatory exponentially decaying wave in the Akers-Milewski equation.

In Figure 2.2, we display an example of a simulation of the Benjamin-Ono solitary wave,

$$u(x,t) = \frac{4}{1+(x)^2}. \tag{3.1}$$

The simulation was computed in a frame which travels with the wave so that the wave appears stationary. As is common practice, a large domain size with periodic boundary conditions was used as proxy for $\mathbb{R}$ [7] since the solution decays as $|x| \to \infty$. The crest of the wave profile, in the left panel, is marked with a solid black line, to highlight the lack of oscillation in time. The initial profile is also marked with a solid black line for highlighting purposed. The right panel shows the step sizes used for this simulation, which range over an order of magnitude with $\Delta x \in [0.125, 1.25]$. The node spacing is concentrated near the origin, where the wave (and its derivatives) is largest; the largest spacing occurs for large x, where the wave is small. This allows for increased accuracy for the same number of points as compared to uniform step sizes.

In Figure 3.1, we study the accuracy of the spatial discretization of the linear operator,

$$\mathcal{L}u = \mathcal{H}u_{xx} + u_x, \tag{3.2}$$

when applied to the solitary wave (3.1). There are two competing parameters which determine the spatial accuracy, the domain size L and the minimal space step Δx. We present two experiments, one where the domain size is varied for a sampling of parameterizations (left panel of Figure 3.1) and one where the parameterization is varied for fixed domain size (right panel of Figure 3.1). In the left panel, all discretizations show an initial decrease in error as the domain size increases, up to a point where the truncation error of the discretization of the linear operator grows to be larger than the domain discretization error. For each discretization, including Fourier, there is a length L for which the method is most accurate (for fixed number of points). All of the RBF-FD discretizations, based on (2.2), are able to give a more accurate discretization than a Fourier discretization (with the best observed improvement at $a = 3.5$). That these methods outperform a Fourier discretization on the Benjamin-Ono solitary wave is natural, since a periodic tiling of this wave has discontinuous first derivative at the boundary, limiting the accuracy of a Fourier discretization. The variable space step RBF-FD is not limited by this discontinuity, even with an approximation that assumes more than one continuous derivative at the boundary. The accuracy of the RBF-FD discretization of the linear operator as a function of minimal step size (controlled by the parameter a) is depicted in the right panel of Figure 3.1, in which we observe $O(\Delta x^8)$ accuracy. Given that the approximation of the linear operator was formed by applying the linear operator to an interpolant with a basis set of polynomials terms up to degree eight and polyharmonic splines of r^7, it is not surprising to see $O(\Delta x^8)$ accuracy since the interpolation procedure provides for at least that order of accuracy [15].

The method used here to discretize the linear operator in equation (3.2) is based on a diagonalization of a RBF-FD differentiation matrix. First we compute an RBF-FD approximation of a derivative matrix. Next we compute the eigenvalues and eigenvectors of this matrix. Because we know the exact spectrum of the derivative and the Hilbert transform in the infinite dimensional problem from Fourier analysis,

$$\hat{\partial_x} = ik \qquad \text{and} \qquad \hat{\mathcal{H}} = -i\text{sign}(k),$$

we construct eigenvalues of the discrete Hilbert transform by applying the same relationship to the discrete problem. If λ_{Dx} is an eigenvalue of the differentiation matrix, then the eigenvalues of the discretized Hilbert transform $\lambda_{\mathcal{H}}$ are defined to satisfy

$$\lambda_{\mathcal{H}} = i\text{sign}(\text{imag}(\lambda_{Dx})).$$

Eigenvalue manipulation based on the above definition is used to find all the eigenvalues of the approximation to (3.2), the eigenvector matrix (and its inverse) of the RBF-FD differentiation matrix are then multiplied by diagonal matrices with this approximate spectrum to get an approximation for (3.2). Although the eigenvalues were constructed to be pure imaginary, as is the spectrum of the exact problem, the result of the matrix multiplication (by the matrix of eigenvectors and its inverse) can perturb the spectrum due to machine precision errors. These errors scale with the condition number of the matrix of the eigenvectors of the RBF-FD differentiation matrix. As is the case for classic polynomial interpolation, the condition number of this matrix grows as the points get closer together. This increase in condition number

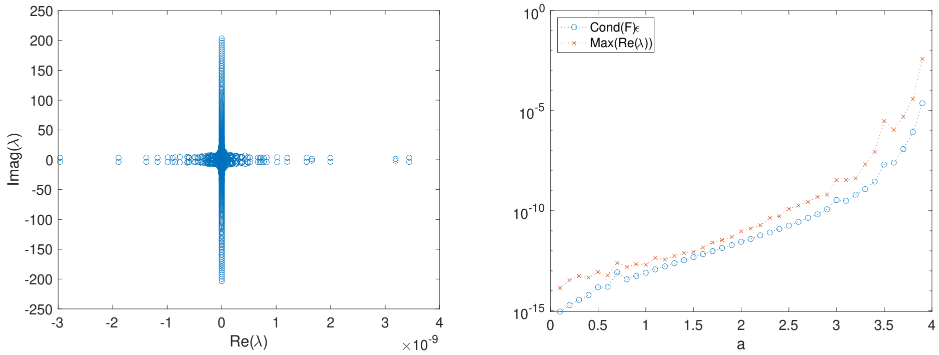

Fig. 3.2. **Left:** *The spectrum of the linear operator used for the evolution depicted in Figure* (2.2). **Right:** *The condition number and the maximally real eigenvalue of the approximation of the linear operator as a function of the step-size parameter 'a' in equation* (2.2)*, marked with x's, is compared to the condition number of eigenvector matrix, F, of the RBF-FD differentiation matrix times the machine precision, ϵ, marked with circles.*

results in a corresponding increase in the size of spurious real eigenvalues of the discretized linear operator (which poses a stability problem for numerical time stepping algorithms). To observe this phenomenon, after creating the approximate linear operator $\mathcal{L}_k$, we apply a QR iteration to compute the eigenvalues of this matrix. That these computed eigenvalues differ at all from the desired spectrum is a direct result of the conditioning of the eigenspace of RBF-FD differentiation matrix. An example of the computed spectrum of the matrix $\mathcal{L}_k$ used to evolve the Benjamin-Ono solitary wave in Figure 2.2 is in the left panel of Figure 3.2. This spectrum includes spurious eigenvalues with real part as large as 10^{-9}. In the right panel of Figure 3.2, we observe the dependence of the eigenvalue of with largest real part as a function of the parameter a from equation (2.2). The match between the size of the eigenvalues and the condition number of the matrix of eigenvectors, F, times the machine precision is displayed in the right panel of Figure 3.2.

A few references have studied the behavior of eigenvalue stability with regards to node placement. This includes [16] which investigates special node distributions that improve eigenvalue stability for global RBF methods as well as [38, 39, 40, 41, 42] which investigate the effects of using mapped nodal sets on the accuracy and eigenvalue stability of finite-difference and pseudo-spectral methods. These works may provide the framework for future research avenues which resolve the relationship between node placement the ill-conditioning of the eigenspace of differential operators.

The method presented here and error analysis are presented in the context of the Benjamin-Ono equation, where the combination of algebraic decay of the solitary wave and nonlocal nature of the Hilbert transform make a challenging testbed for a numerical scheme. The same ideas generalize easily to other nonlocal equations; it is trivial to apply this method to the Whitham [22] or Akers-Milewski equation [20, 43]. The Akers-Milewski equation,

$$u_t + \mathcal{H}u - \mathcal{H}u_{xx} + uu_x = 0, \tag{3.3}$$

supports traveling, wavepacket type solitary waves [44]. These waves decay exponentially in space, thus do not present the same challenges for numerical simulation,

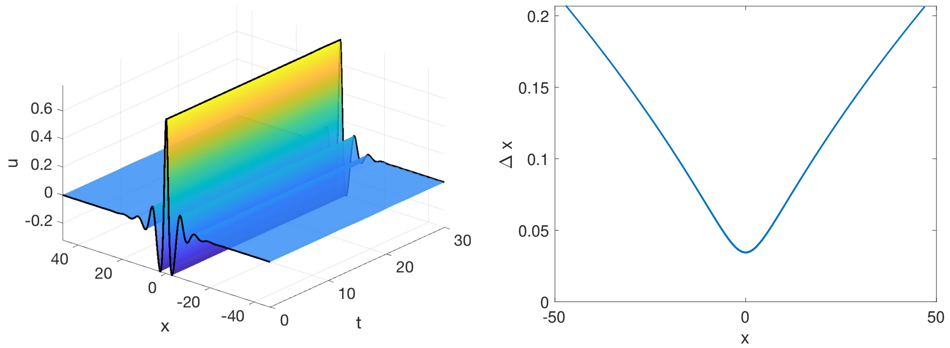

Fig. 3.3. **Left:** *An example of the evolution of a traveling wavepacket-type solitary wave in the Akers-Milewski equation* (3.3). **Right:** *The step sizes used for this simulation.*

(Fourier collocation is spectrally accurate), as the Benjamin-Ono solitary wave (3.1). As evidence of the ease of generalizing this approach, we include an example of the evolution of such a wave using RBF-FD discretization a non-uniform grid in Figure 3.3.

4. Conclusion In this paper an RBF-FD implementation of the Benjamin-Ono equation was presented. This required the development of an RBF-FD implementation of the Hilbert transform. An approximation of the Hilbert transform was built by manipulating the spectrum of a differentiation matrix. This approach generalizes simply to other pseudo-differential operators, but incurs an $O(N^3)$ pre-processing cost, meaning that this method is expensive compared to a Fourier implementation. The approach however, allows for arbitrary boundary conditions and non-uniform grids. We expect it to be most useful in problems where an $O(N^3)$ pre-processing cost is already being paid, for example in the evolution of a coherent structure which was computed via quasi-Newton iteration. A future avenue for research is in the relationship between the node placement and the conditioning of the eigenspace of the differentiation matrix; we observe that this plays a key role in the accuracy of the approximation of the spectrum of the linear operator.

Disclaimer: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, make any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

Acknowledgements: All three authors acknowledge support of Office of Naval Research via the project *Atmospheric Propagation Sciences for High Energy Lasers.* B.F. Akers acknowledges support from the Air Force Office of Sponsored Research's

Computational Mathematics Program via the project *Radial Basis Functions for Numerical Simulation.*

REFERENCES

[1] Terence Tao. Global well-posedness of the benjamin–ono equation in h1 (r). *Journal of Hyperbolic Differential Equations*, 1(01):27–49, 2004.

[2] Y Matsuno. Interaction of the benjamin-ono solitons. *Journal of Physics A: Mathematical and General*, 13(5):1519, 1980.

[3] DJ Kaup and Y Matsuno. The inverse scattering transform for the benjamin–ono equation. *Studies in applied mathematics*, 101(1):73–98, 1998.

[4] David M Ambrose and Jon Wilkening. Computation of time-periodic solutions of the benjamin–ono equation. *Journal of nonlinear science*, 20(3):277–308, 2010.

[5] Bao-Feng Feng and Takuji Kawahara. Temporal evolutions and stationary waves for dissipative benjamin–ono equation. *Physica D: Nonlinear Phenomena*, 139(3-4):301–318, 2000.

[6] Beatrice Pelloni and Vassilios A Dougalis. Numerical solution of some nonlocal, nonlinear dispersive wave equations. *Journal of Nonlinear Science*, 10(1):1–22, 2000.

[7] Henrik Kalisch. Error analysis of a spectral projection of the regularized benjamin–ono equation. *BIT Numerical Mathematics*, 45(1):69–89, 2005.

[8] John P Boyd and Zhengjie Xu. Comparison of three spectral methods for the benjamin–ono equation: Fourier pseudospectral, rational christov functions and gaussian radial basis functions. *Wave Motion*, 48(8):702–706, 2011.

[9] Gregory E Fasshauer. *Meshfree approximation methods with MATLAB*, volume 6. World Scientific, 2007.

[10] Bengt Fornberg and Natasha Flyer. *A primer on radial basis functions with applications to the geosciences*. SIAM, 2015.

[11] J. A. Reeger and B. Fornberg. Numerical quadrature over the surface of a sphere. *Stud. Appl. Math.*, 137(2):174–188, 2016.

[12] J. A. Reeger, B. Fornberg, and M. L. Watts. Numerical quadrature over smooth, closed surfaces. *P. Roy. Soc. Lon. A Mat.*, 472, 2016. doi: 10.1098/rspa.2016.0401.

[13] J. A. Reeger and B. Fornberg. Numerical quadrature over smooth surfaces with boundaries. *J. Comput. Phys.*, 355:176–190, 2018.

[14] J. A. Reeger. Approximate integrals over the volume of the ball. *J. Sci. Comput.*, 83(45), 2020.

[15] V. Bayona. An insight into RBF-FD approximations augmented with polynomials. *Comput. Math. Appl.*, 77(9):2337–2353, 2019.

[16] Rodrigo B Platte and Tobin A Driscoll. Eigenvalue stability of radial basis function discretizations for time-dependent problems. *Computers & Mathematics with Applications*, 51(8):1251–1268, 2006.

[17] Scott A Sarra. A numerical study of the accuracy and stability of symmetric and asymmetric rbf collocation methods for hyperbolic pdes. *Numerical Methods for Partial Differential Equations: An International Journal*, 24(2):670–686, 2008.

[18] Nicolas Ali Libre, Arezoo Emdadi, Edward J Kansa, Mohammad Shekarchi, and Mohammad Rahimian. A multiresolution prewavelet-based adaptive refinement scheme for rbf approximations of nearly singular problems. *Engineering analysis with boundary elements*, 33(7):901–914, 2009.

[19] Oleg Davydov and Dang Thi Oanh. Adaptive meshless centres and rbf stencils for poisson equation. *Journal of Computational Physics*, 230(2):287–304, 2011.

[20] Benjamin Akers and Paul A Milewski. A model equation for wavepacket solitary waves arising from capillary-gravity flows. *Studies in Applied Mathematics*, 122(3):249–274, 2009.

[21] Quan Shen. A meshless method of lines for the numerical solution of kdv equation using radial basis functions. *Engineering Analysis with Boundary Elements*, 33(10):1171–1180, 2009.

[22] Mats Ehrnström, Henrik Kalisch, et al. Traveling waves for the whitham equation. *Differential and Integral Equations*, 22(11/12):1193–1210, 2009.

[23] Benjamin Akers and Paul A Milewski. Dynamics of three-dimensional gravity-capillary solitary waves in deep water. *SIAM Journal on Applied Mathematics*, 70(7):2390–2408, 2010.

[24] Benjamin Akers and Paul A Milewski. Model equations for gravity-capillary waves in deep water. *Studies in Applied Mathematics*, 121(1):49–69, 2008.

[25] MJ Ablowitz, AS Fokas, J Satsuma, and Harvey Segur. On the periodic intermediate long wave equation. *Journal of Physics A: Mathematical and General*, 15(3):781, 1982.

[26] E Parau, Jean Mark Vanden-Broeck, and MJ Cooker. Nonlinear three-dimensional gravity-capillary solitary waves. *Journal of Fluid Mechanics*, 536:99–105, 2005.

[27] KL Oliveras and CW Curtis. Nonlinear travelling internal waves with piecewise-linear shear profiles. *Journal of Fluid Mechanics*, 856:984–1013, 2018.

[28] Kyle M Claassen and Mathew A Johnson. Numerical bifurcation and spectral stability of wavetrains in bidirectional whitham models. *Studies in Applied Mathematics*, 141(2):205–246, 2018.

[29] N. Flyer, E. Lehto, S. Blaise, G. B. Wright, and A. St-Cyr. A guide to RBF-generated finite-differences for nonlinear transport: Shallow water simulations on a sphere. *J. Comput. Math.*, 231(11):4078–4095, 2012.

[30] N. Flyer, G.B. Wright, and B. Fornberg. Radial basis function-generated finite differences: A mesh-free method for computational geosciences. In W. Freeden, M. Z. Nashed, and T. Sonar, editors, *Handbook of Geomathematics*. Springer Verlag Berlin Heidelberg. doi: 10.1007/978-3-642-27793-1 61-1.

[31] B. Fornberg and N. Flyer. *A Primer on Radial Basis Functions with Applications to the Geosciences*. SIAM, 2015.

[32] B. Fornberg and N. Flyer. Solving PDEs with radial basis functions. *Acta Numerica*, 24:215–258, 2015.

[33] H. Wendland. *Scattered Data Approximation*. Cambridge University Press, Cambridge, U.K., 2005.

[34] U.M. Ascher, S.J. Ruuth, and B.T.R. Wetton. Implicit-explicit methods for time-dependent partial differential equations. *SIAM J. Num. Anal.*, 32(3):797–823, 1995.

[35] James M Varah. Stability restrictions on second order, three level finite difference schemes for parabolic equations. *SIAM Journal on Numerical Analysis*, 17(2):300–309, 1980.

[36] A.-K. Kassam and L.N Trefethen. Fourth-order time-stepping for stiff PDEs. *SIAM Journal on Scientific Computing*, 26(4):1214–1233, 2005.

[37] P.A. Milewski and E.G. Tabak. A pseudospectral procedure for the solution of nonlinear wave equations with examples from free-surface flows. *SIAM J. Sci. Comp.*, 21(3):1102–1114, 1999.

[38] Dan Kosloff and Hillel Tal-Ezer. A modified chebyshev pseudospectral method with an o (n-1) time step restriction. *Journal of Computational Physics*, 104(2):457–469, 1993.

[39] Alvin Bayliss and Eli Turkel. Mappings and accuracy for chebyshev pseudo-spectral approximations. *Journal of Computational Physics*, 101(2):349–359, 1992.

[40] Jie Shen and Li-Lian Wang. Error analysis for mapped legendre spectral and pseudospectral methods. *SIAM journal on numerical analysis*, 42(1):326–349, 2004.

[41] Xiaolin Zhong and Mahidhar Tatineni. High-order non-uniform grid schemes for numerical simulation of hypersonic boundary-layer stability and transition. *Journal of Computational Physics*, 190(2):419–458, 2003.

[42] Ratnesh K Shukla and Xiaolin Zhong. Derivation of high-order compact finite difference schemes for non-uniform grid using polynomial interpolation. *Journal of Computational Physics*, 204(2):404–429, 2005.

[43] James Diorio, Yeunwoo Cho, James H Duncan, and TR Akylas. Gravity-capillary lumps generated by a moving pressure source. *Physical review letters*, 103(21):214502, 2009.

[44] Benjamin F Akers and Matthew Seiders. Numerical simulations of overturned traveling waves. In *Nonlinear Water Waves*, pages 109–122. Springer, 2019.