Solving the Korteweg-de Vries equation with Hermite-based finite differences

Dylan Abrahamsen, Bengt Fornberg

Department of Applied Mathematics, University of Colorado, Boulder, CO 80309

Abstract

The Korteweg-de Vries (KdV) equation is extensively studied in the field of nonlinear waves, with one key tool for this being fast and accurate numerical algorithms. Finite difference (FD) and pseudo-spectral (PS) methods are commonly used. We discuss here the pros and cons in this application area for a new class of Hermite-based finite difference (HFD) methods. Their most notable characteristic is to remain more 'local' than FD approximations for increasing orders of accuracy, translating into smaller error constants.

Keywords: Hermite, finite difference, partial differential equations, KdV, non-linear PDE
2000 MSC: 65M06, 65M12

1. Introduction

Finite difference stencils gain accuracy orders in proportion to their stencil widths. Pseudo-spectral (PS) methods can be seen as the infinite order FD limit, or alternatively, as obtaining derivative approximation through trigonometric interpolation [4]. Although highly accurate, this limit comes with some disadvantages that include: (i) approximations to physically 'local' operators becoming global, (ii) difficulties implementing non-periodic boundary conditions, and (iii) limited options for local refinement.

Hermite-based FD (HFD) formulas can have about half as wide stencils as FD approximations for the same order of accuracy and, for the same order, feature significantly lower error constants. HFD approximations go a long way towards meeting the concerns (i) - (iii) listed above. The goal of this paper is to clarify the strengths and possible weaknesses of this HFD approach when applied to the KdV equation. Although a potential benefit of more compact stencils could be the handling of boundary conditions, this study is entirely focused on the initial value problem.

'Hermite interpolation' - using both function and derivative values at node points - has a long history, going back at least to Hermite in 1878 [9]. 'Hermite-based' finite difference (HFD) formulas are of more recent origin. After Salzer described an algorithm to obtain HFD weights in 1960 [12], only few efforts were made to use these for numerical solutions of PDEs. Most notable were certain non-oscillatory (HWENO) methods [11], which also consider the system created by the

Email addresses: Dylan.Abrahamsen@colorado.edu (Dylan Abrahamsen), Bengt.Fornberg@colorado.edu (Bengt Fornberg)

governing PDE and its space derivative. While WENO methods are primarily used as limiters for non-smooth solutions, HFD is designed to be used for several times differentiable solutions. HFD concepts have also been used previously in different time independent settings, such as for ODE two-point boundary value problems [7], for PDEs in mathematical finance [10], and together with radial basis functions [13, 14].

The Hermite methods introduced in [8] are for time dependent PDEs, but are of quite different character than HFD approximations. Typically, these methods use very narrow stencils, include very high derivative orders, and are then usually evolved in time using a Taylor expansion-based evolution operator.

Our present effort originates from the recent HFD weights algorithm described in [5]. Straightforward application of HFD for spatial derivatives in combination with standard 'Method-of-Lines' (MOL) time stepping was in this reference tested only in some constant coefficient linear convection cases. In this present study, we extend this in several directions:

i Applications of HFD to the Korteweg-de Vries (KdV) equation. This equation is strongly nonlinear, and of significant interest,

ii Providing accuracy comparisons against matching order FD approximations (using both linear points per wavelength (PPW) and non-linear (side-band growth) comparison measures),

iii Time stepping stability constraints and memory costs when using explicit time stepping methods.

However, in order to keep the present study reasonably short and focused, we need to leave certain aspects for follow-up investigations. These include:

iv HFD implementation of boundary conditions in time dependent settings,

v Opportunities for local noe refinement (HFD vs. FD),

vi Stability conditions in case of implicit rather than (as here, in the more demanding case of) explicit time stepping.

As will be explained later in this study, present indications are that the HFD approach will prove highly attractive (compared to regular FD) also in these three aspects.

The remainder of this paper is organized as follows. In Section 2, the general procedure of generating and applying HFD stencils to the KdV equation is discussed. This is followed by discussion of accuracy, stability, linearized dispersion, and points-per-wavelength. In Section 3, we describe some KdV test problems and compare results against standard FD stencils. Section 4 contains closing remarks followed by the appendix that includes tables for stencil weights and other information to reproduce the results of this paper[1].

[1]The numerical calculations were performed in MATLAB together with the Advanpix extended precision package [3].

2. Description of procedure

This section explains the procedure of using Hermite-based finite difference stencils in the context of numerically solving the KdV equation.

2.1. Stencil calculation

Given a spatial differential operator $L = \frac{\partial^p}{\partial x^p}$, the idea is to find a set of finite difference weights acting on function and derivative values to approximate L. The approximation will take the form

$$Lu\big|_{x=x_c} \approx \sum_{m=0}^{M} \sum_{i=1}^{n} w_{i,m} u_i^{(m)} \tag{1}$$

where n is the stencil size and $M < p$ is the highest derivative of u to include. For example, the $M = 1$ case will include function and first derivative values

$$Lu\big|_{x=x_c} \approx \sum_{i=1}^{n} w_{i,0} u_i + \sum_{i=1}^{n} w_{i,1} \frac{du_i}{dx} \tag{2}$$

Finite difference stencils with $M > 0$ have beneficial characteristics over their $M = 0$ counterparts when solving partial difference equations, as noted in [5] and discussed further in this paper. Since HFD stencils are based on Hermite interpolation, one benefit is that they are less prone to spurious oscillations introduced by Lagrange interpolation on equidistant node sets [2]. The calculation of the weights $w_{i,m}$ can be done via a Vandermonde system or, in the $M = 0$ and $M = 1$ cases, by the algorithm in [5]. For example, the centered 3-node wide $M = 1$ stencil approximating $\frac{\partial^3}{\partial x^3}$ and $\frac{\partial^4}{\partial x^4}$ for use in the KdV equation are

$$\begin{array}{ll} \frac{d^3 f}{dx^3}\big|_{x=x_c} & = \left[-\frac{15}{2} \quad 0 \quad \frac{15}{2}\right] \frac{f}{h^3} + \left[-\frac{3}{2} \; -12 \; -\frac{3}{2}\right] \frac{f'}{h^2} + \mathcal{O}\left(h^4\right) \\ \frac{d^4 f}{dx^4}\big|_{x=x_c} & = [-12 \quad 24 \; -12] \frac{f}{h^4} + [-6 \quad 0 \quad 6] \frac{f'}{h^3} + \mathcal{O}\left(h^2\right) \end{array} \tag{3}$$

Here, x_c is the center point of the stencil on a grid with spacing h. In the context of this paper the same stencil width (s.w.) will be used when approximating both the third and fourth derivatives with HFD. We introduce the notation HFD4/2 to indicate that the third and fourth derivative are fourth and second order accurate respectively. A table of weights for some different derivatives and accuracy orders is given in the Appendix.

2.2. System derivation

In order to apply HFD stencils to PDEs, one typically needs to differentiate the governing PDE in space. To be specific, we consider here the KdV equation in the form,

$$u_t(x,t) + u_{xxx}(x,t) + 6\left(\frac{u(x,t)^2}{2}\right)_x = 0 \tag{4}$$

In a similar approach to the cases described in [5], this equation is differentiated in space once

$$u_{tx} = -u_{xxxx} - 6\left(\frac{u^2}{2}\right)_{xx}$$

By letting $v = u_x$, the system to solve becomes

$$\begin{bmatrix} u \\ v \end{bmatrix}_t = \begin{bmatrix} -u_{xxx} - 6uv \\ -u_{xxxx} - 6\left(\frac{u^2}{2}\right)_{xx} \end{bmatrix} \tag{5}$$

Separating the linear and nonlinear terms and considering u and v as column vectors, the semi-discrete system for use with $M = 1$ HFD stencils becomes

$$\begin{bmatrix} u \\ v \end{bmatrix}_t = -\begin{bmatrix} w_{xxx,0} & w_{xxx,1} \\ w_{xxxx,0} & w_{xxxx,1} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} - 6\begin{bmatrix} uv \\ \left(\frac{u^2}{2}\right)_{xx} \end{bmatrix}, \tag{6}$$

where the blocks $w_{k,i}$ denote the differentiation matrices approximating the corresponding derivative k using equation (2). The $\left(\frac{u^2}{2}\right)_{xx}$ term is implemented as

$$\left(\frac{u^2}{2}\right)_{xx} = \begin{bmatrix} w_{xx,0} & w_{xx,1} \end{bmatrix} \begin{bmatrix} \frac{u^2}{2} \\ uv \end{bmatrix}.$$

Further differentiating the system allows for higher derivatives $M > 1$ to be included in each HFD stencil.

2.3. Discussion of accuracy and stability

In this section we look at the order of convergence of using HFD stencils of the form (1) to approximate the right hand side (RHS) of (6) on the single soliton solution of (4)

$$u_1(x,t) = \frac{c}{2}\operatorname{sech}^2\left(\frac{\sqrt{c}}{2}(x - ct - a)\right) \tag{7}$$

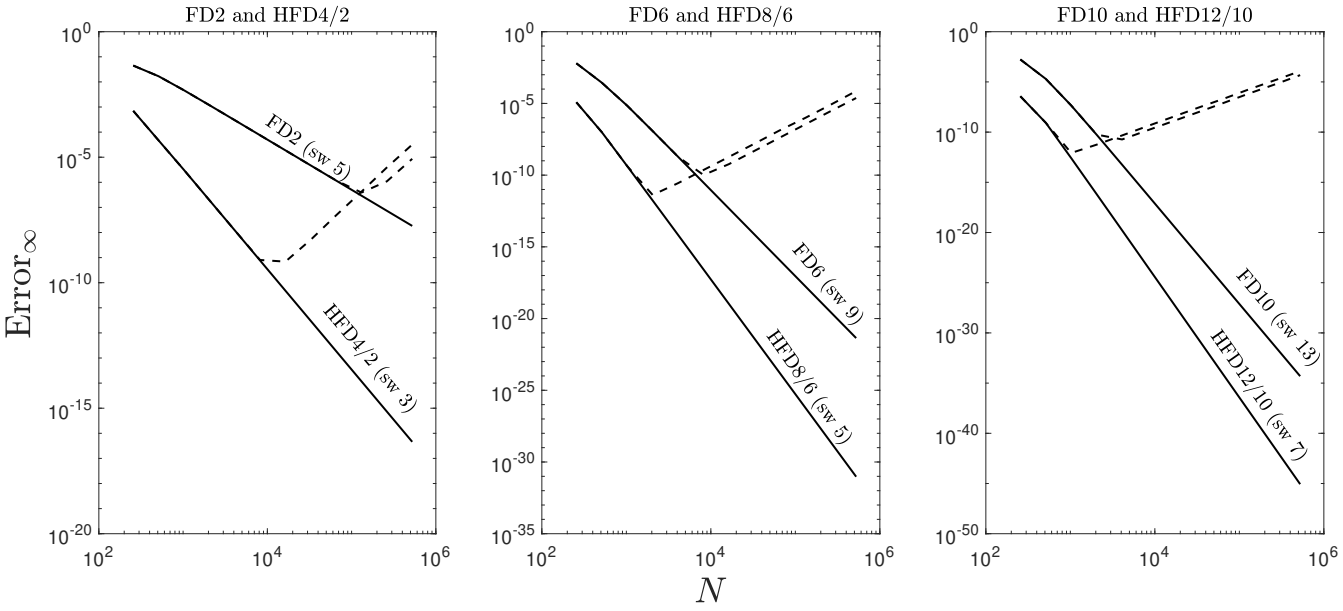

Figure 1: The max error of $-u_{xxx} - 6\left(\frac{u^2}{2}\right)_x$ using similar order FD and $M = 1$ HFD stencils applied to an initial soliton in double precision (dotted lines) and extended precision (solid lines) displayed on different vertical scales.

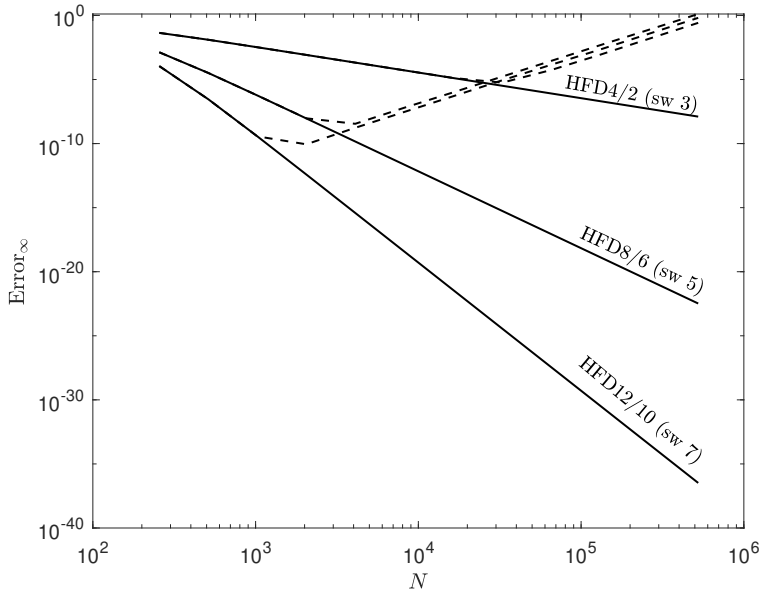

Figure 2: The max error of $-u_{xxxx} - 6\left(\frac{u^2}{2}\right)_{xx}$ in the same case as in Figure 1 using HFD for double precision (dotted lines) and extended precision (solid lines).

Figure 1 shows the convergence rates of the centered three, five, and seven point wide $M = 1$ HFD stencils in comparison to corresponding order FD stencils when acting on the soliton (7) at $t = 0$ with $c = 1$ and $a = 0$ on the domain $x \in [-25\pi, 25\pi]$. The error is the largest difference in $-u_{xxx} - 6\left(\frac{u^2}{2}\right)_x$ compared to an exact evaluation. Figure 2 illustrates the convergence of the HFD stencils on $-u_{xxxx} - 6\left(\frac{u^2}{2}\right)_{xx}$ from (6). Each error curve is shown with calculations in double precision (dotted line) and extended precision (solid line). The order of convergence for HFD jumps by four by increasing the stencil size by two. Heuristically, this is due to including four new degrees of freedom.

The dashed lines representing double precision rise to the right in Figure 1 (and similarly Figure 2). This is inevitable for any accurate approximations of u_{xxx}. With a spatial step size h, the Fourier modes $\nu(x) = e^{i\omega x}$ that are present on the grid satisfy $-\frac{\pi}{h} \leq \omega \leq \frac{\pi}{h}$ and are in double precision uncertain to about 10^{-16}. The third derivative $\nu'''(x)$ of the highest Fourier modes become of size $\frac{\pi^3}{h^3}$, implying an uncertainty of around $\frac{3 \cdot 10^{-15}}{h^3}$. It is notable that the present re-formulation of (4) into the system (5) (that also includes a fourth derivative) does not turn this into $\mathcal{O}\left(\frac{1}{h^4}\right)$, but preserves the optimal $\mathcal{O}\left(\frac{1}{h^3}\right)$ level. The value where the decreasing truncation errors meet the increasing rounding errors depend on the smoothness of the test function that is considered. The highest accuracy achieved by HFD before double precision error occurs is on par, if not slightly better, than the FD counterpart.

The solid curves in Figures 1 and 2 depend on the smoothness (decay rate of Fourier coefficients) of the test function (7), while the dashed lines eventually do not (instead resulting from limiatations of double precision and the presence of $u_{xxx}(x,t)$ in (4)).

2.3.1. Time stepping stability

One potential concern is that the differentiation matrix formed in (6) might introduce a stability constraint like $\mathcal{O}\left(\frac{k}{h^4}\right)$ due to the introduction of the fourth order spatial derivative. However, this does not occur, since the largest eigenvalue of the linear term differentiation matrix formed scales like $\mathcal{O}\left(\frac{k}{h^3}\right)$ regardless of including weights approximating higher spatial derivatives. This can be seen via a similarity transform. Since the differentiation matrix in (6) can be expressed as a 2x2 block matrix with element sizes $\begin{bmatrix} \frac{1}{h^3} & \frac{1}{h^2} \\ \frac{1}{h^4} & \frac{1}{h^3} \end{bmatrix}$, a similarity transform yields

$$\begin{bmatrix} \frac{1}{h} & 0 \\ 0 & \frac{1}{h^2} \end{bmatrix} \begin{bmatrix} \frac{1}{h^3} & \frac{1}{h^2} \\ \frac{1}{h^4} & \frac{1}{h^3} \end{bmatrix} \begin{bmatrix} h & 0 \\ 0 & h^2 \end{bmatrix} = \begin{bmatrix} \frac{1}{h^3} & 0 \\ 0 & \frac{1}{h^3} \end{bmatrix}$$

This argument carries through similarly when including even higher derivatives in the HFD stencils.

2.3.2. Differentiation matrix size and sparsity

Another potential drawback of differentiating the governing PDE is that the system size increases and enlarges the differentiation matrix to scale like $\mathcal{O}\left(M^2N^2\right)$. However, these matrices are highly sparse, so actual cost and memory scales only like $\mathcal{O}(MN)$. Furthermore, to match the accuracy of explicit FD, much fewer nodes are required for the same stencil size. To illustrate this, we consider the single soliton again. Figure 3 shows the required approximate step needed to time step with explicit fourth order Runge-Kutta (RK4). HFD not only requires fewer nodes in space to match the accurate of FD, but also needs fewer time steps.

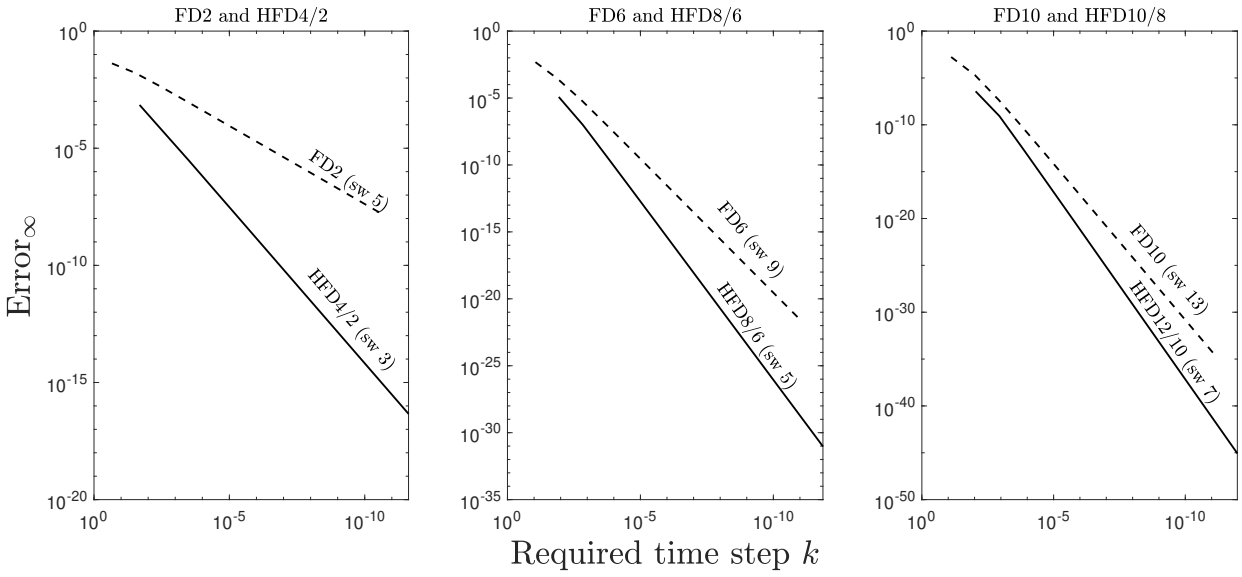

Figure 3: The relationship for the approximate largest time step k allowed for time stepping KdV with RK4 and a given error for the single soliton for 2nd, 6th, and 10th order stencils displayed on different vertical scales.

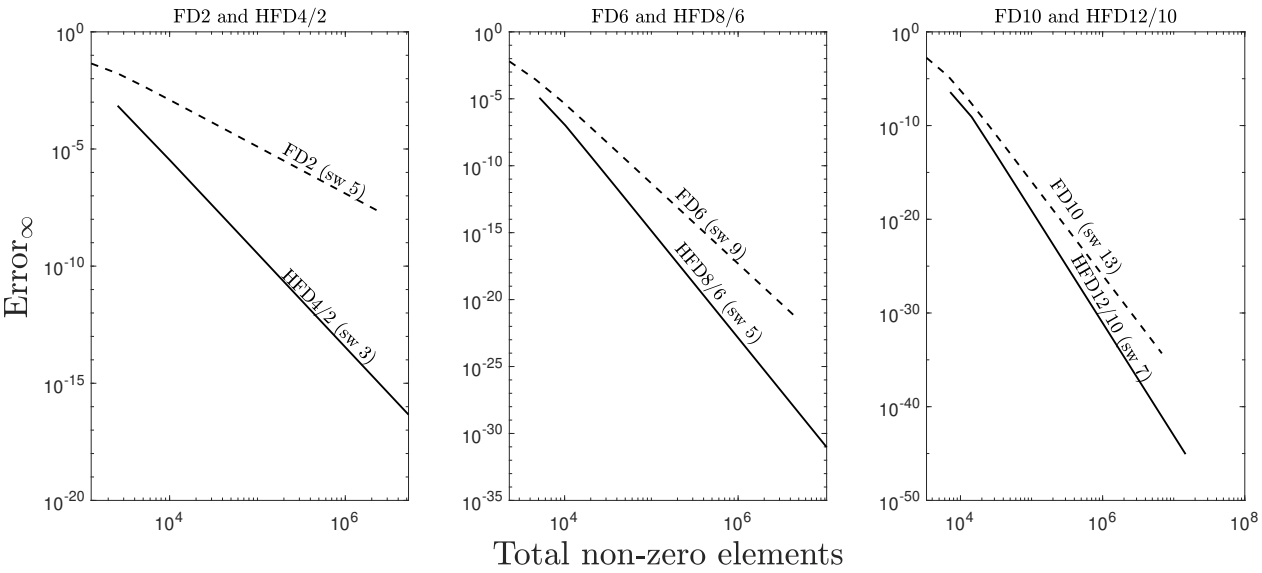

Figure 4: The relationship for the number of non-zero elements of the differentiation matrix and a given error for the single soliton for 2nd, 6th, and 10th order stencils displayed on different vertical scales.

Figure 4 shows the differentiation matrix size (measured as the number of non-zero entries of sparse matrices) as a function of error for the single soliton. The DM size of HFD required to match the same error of FD is much lower.

2.4. Linearized error analysis

2.4.1. Dispersion

Dispersion analysis is widely used to assess different schemes in the context of solving wave equations. We use as a test equation the standard 1-D wave equation

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 0 \tag{8}$$

with the single Fourier mode solution $u(x,t) = e^{i\omega x - i\alpha(\omega)t}$. Analytically, it should hold that $\alpha(\omega) = \omega$. HFD can be applied to (8) by differentiating once in space, as considered in [5]. By substituting the single Fourier mode solution into (8) or the appropriate system for HFD and discretely approximating any spatial derivatives a relation for $\alpha(\omega)$ can be found [2].

[2]For example, in the FD, order $p = 4$ case, one obtains $\alpha(\omega) = \frac{1}{ih}\left(\frac{1}{12}e^{-2i\omega h} - \frac{2}{3}e^{-i\omega h} + \frac{2}{3}e^{i\omega h} - \frac{1}{12}e^{2i\omega h}\right) = \frac{4}{3h}\sin(\omega h) - \frac{1}{6h}\sin(2\omega h) = \omega - \frac{h^4\omega^5}{30} + \frac{h^6 7}{252} + \cdots$.

Scheme	Order	$\alpha(\omega)$
FD	4	$\omega - \frac{h^4\omega^5}{30} + \frac{h^6\omega^7}{252} + \cdots$
	8	$\omega - \frac{h^8\omega^9}{630} + \frac{h^{10}\omega^{11}}{2310} + \cdots$
	12	$\omega - \frac{h^{12}\omega^{13}}{12012} + \frac{h^{14}\omega^{15}}{27720} + \cdots$
	16	$\omega - \frac{h^{16}\omega^{17}}{218790} + \frac{h^{18}\omega^{19}}{366795} + \cdots$
HFD	4	$\omega - \frac{h^4\omega^5}{1080} - \frac{h^6\omega^7}{54432} + \cdots$
	8	$\omega - \frac{h^8\omega^9}{441000} - \frac{h^{10}\omega^{11}}{22638000} + \cdots$
	12	$\omega - \frac{h^{12}\omega^{13}}{155387232} - \frac{h^{14}\omega^{15}}{7888890240} + \cdots$
	16	$\omega - \frac{h^{16}\omega^{17}}{50684891400} - \frac{h^{18}\omega^{19}}{2549151891000} + \cdots$

Table 1: The $\alpha(\omega)$ relationship for standard FD and HFD for varying orders.

Table 1 shows the series expansions of the $\alpha(\omega)$ function for standard FD and HFD. It is clear that, for corresponding order, the coefficient of the leading error term is significantly smaller in the HFD case. This can be explained partially by that the highest frequency $\omega_{\max}$ resolved accurately by HFD on the grid is $\frac{2\pi}{h}$ as opposed to $\frac{\pi}{h}$ for standard FD seen in Figure 5.

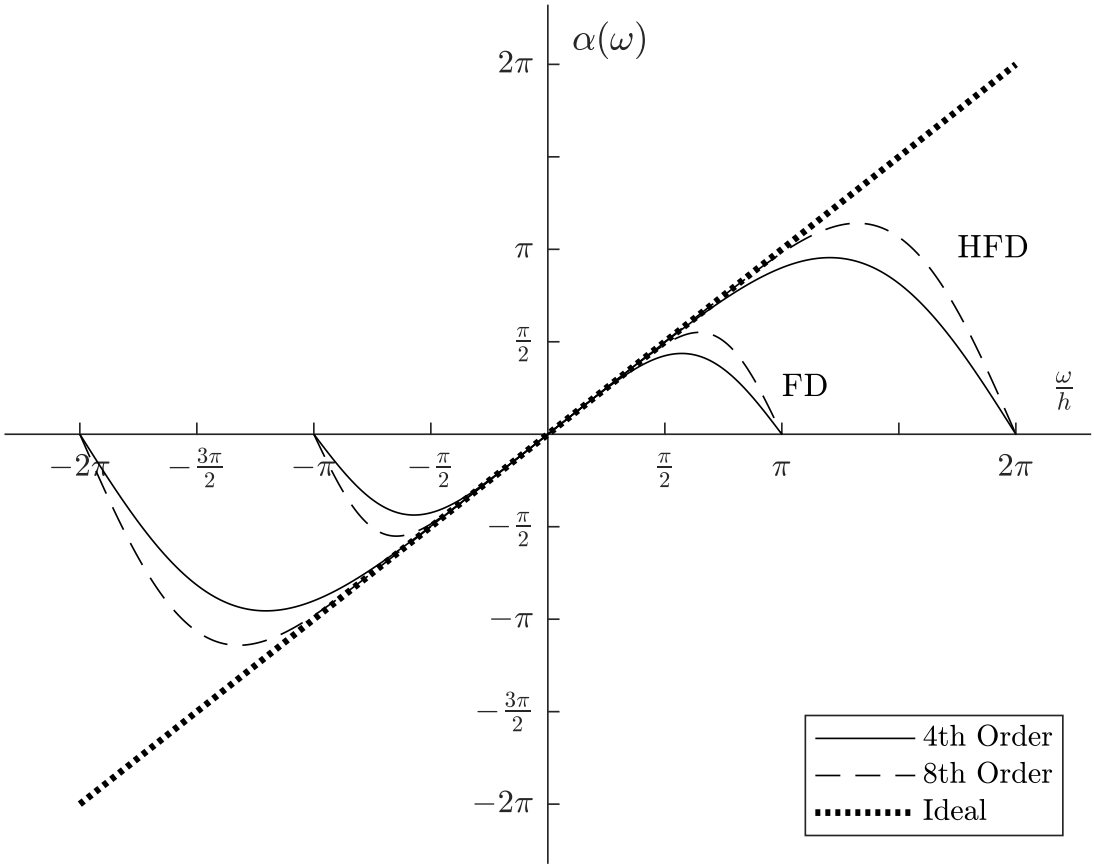

Figure 5: The graphical representation of $\alpha(\omega)$ for FD and HFD compared to the ideal case $\alpha(\omega) = \omega$.

2.4.2. Points per wavelength

In order to compare FD and HFD approximations in terms of 'Points per wavelength' (PPW), we consider the simple convective model (8) over the periodic interval $0 \leq x \leq 2\pi$, and let the time increase from $t = 0$ to $t = 2\pi$. Any initial condition and, in particular, every Fourier mode $e^{i\omega x}$, should then have traveled once over the spatial domain. As can be inferred from Figure 5, each (non-zero) mode will travel too slowly and, at some time, therefore have become in error by a phase angle of $\frac{\pi}{4}$. We take this as indicating that the mode no longer contributes meaningfully to the PDE solution (since, if it was 'off' by π, it would be of the wrong sign, and do the opposite to what it ought to).

With the above as background, we can explain how to read the information in Figure 6. Consider the highest Fourier mode that we require to meet the accuracy requirement at the final time. The horizontal axis indicates the number of full wavelengths of this highest mode that fit into the period $0 \leq x \leq 2\pi$. For each of these 'Number of Wavelengths' values, and each approximation order, we can then compute how many points per wavelength (PPW) this requires. This produces the solid

curves in Figure 6. The higher the order (i.e., higher approximation accuracy), we naturally see lower PPW needed. Also, the higher accuracy we request (further to the right in these figures), the needed PPW increase. In these charts, one can additionally draw curves for the total number of grid points (N) over $0 \leq x \leq 2\pi$ that are needed to meet the required accuracy.

The scales on the axes differ between the two subplots of Figure 6. In order to simplify the PPW comparison between the FD and HFD approaches, the key curves from the subplots are placed in the same diagram in Figure 7.

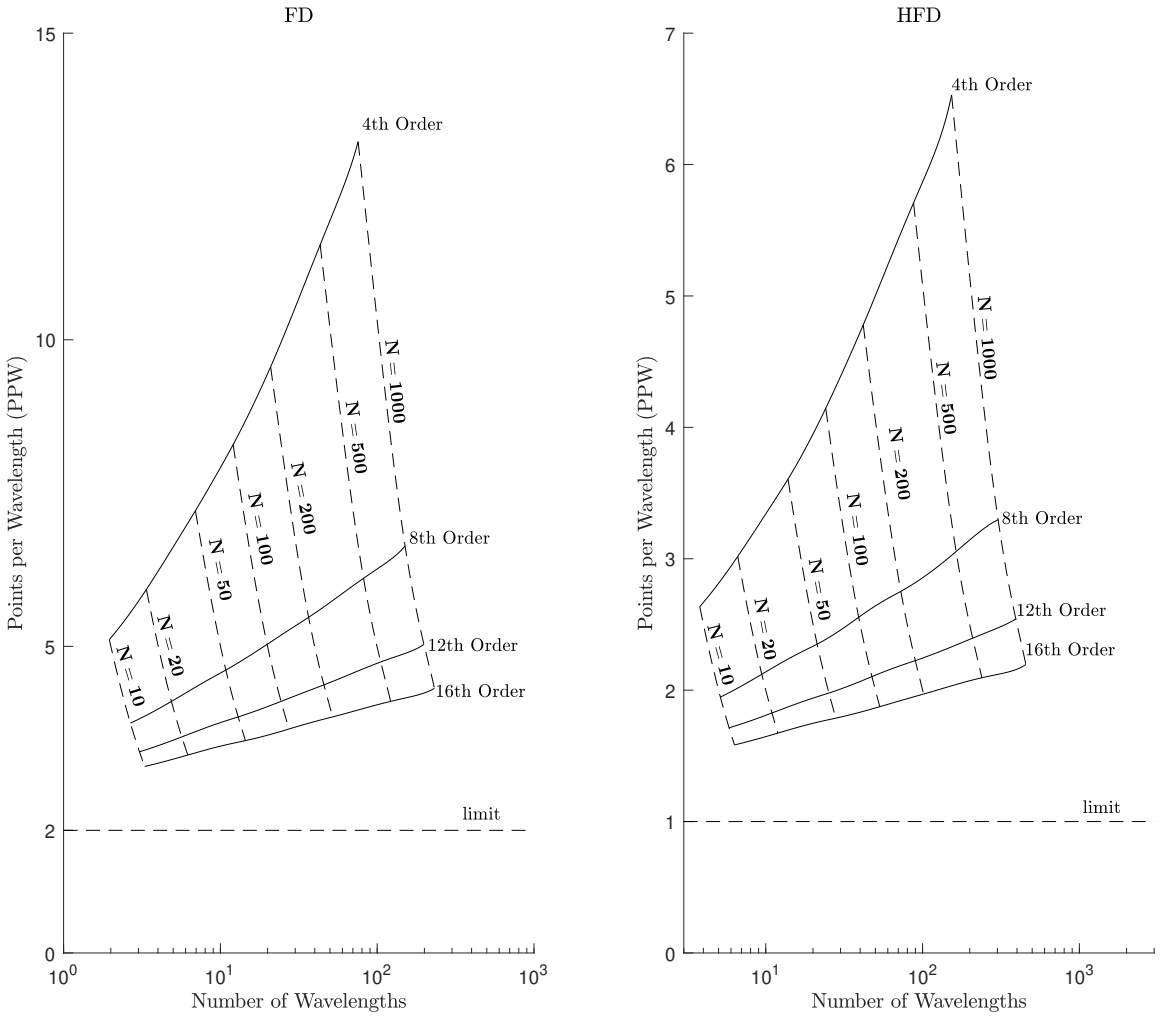

Figure 6: PPW versus Number of wavelengths withing the spatial period of highest wave mode which remains accurate in the test problem, displayed for different node numbers N and accuracy orders. Results are for FD (left) and HFD (right).

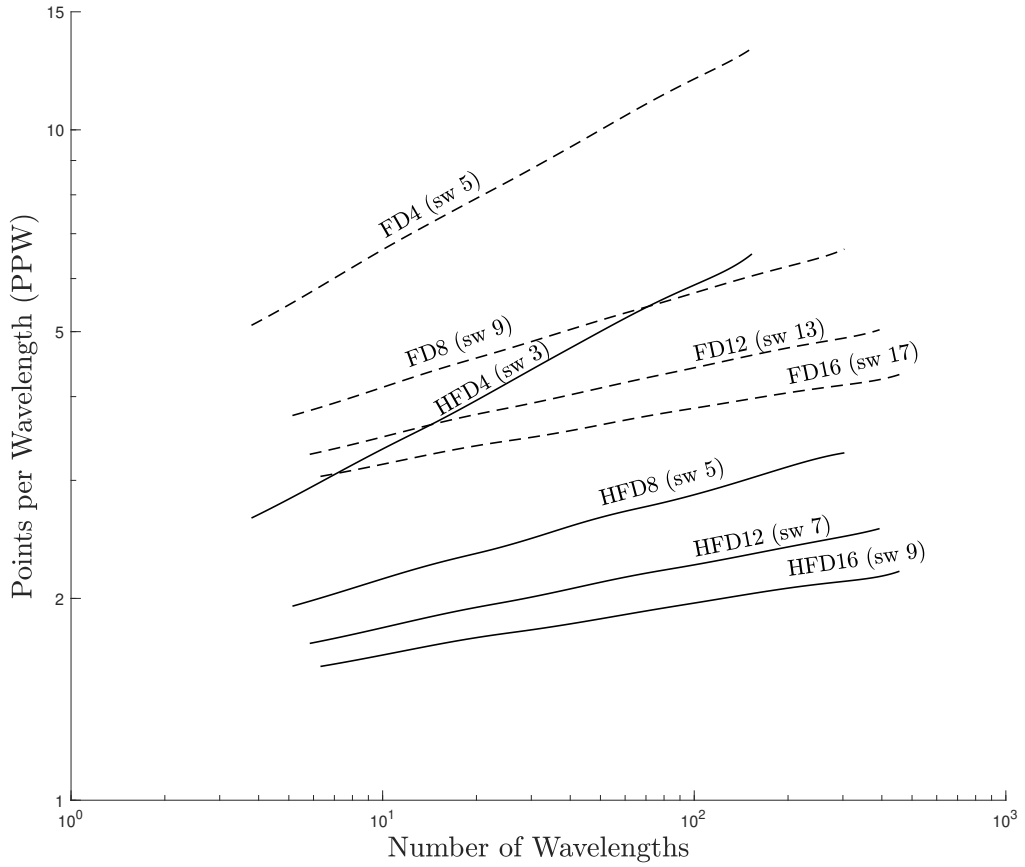

Figure 7: PPW versus Number of wavelengths withing the spatial period of highest wave mode which remains accurate in the test problem, displayed for different node numbers N and accuracy orders. Results are for FD and HFD on the same plot.

Figure 6 and 7 illustrate PPW versus the number of wavelengths in the computational domain which of the highest wave mode we consider accurate. Since the HFD stencils are only approximating one derivative $\left(\frac{\partial^2 u}{\partial x^2}\right)$ we utilize the simpler notation of listing the only order of accuracy here. While the PPW for FD tend towards PS limit of two, the HFD limit is one. Heuristically, this behavior is to be expected as one 'doubles' up on information present at each node location. This will be analyzed further in the follow-up study [1].

3. Additional test problems

Here we apply the method described in Section 2 for various test problems for the KdV equation.

3.1. Time stepping two solitons

We now consider a two soliton solution given by

$$u_2(x,t) = -\frac{2(c_1-c_2)\left[c_1\,\mathrm{sech}^2\left(\sqrt{\frac{c_1}{2}}(x-2c_1t)\right)+c_2\,\mathrm{csch}^2\left(\sqrt{\frac{c_2}{2}}(x-2c_2t)\right)\right]}{\left[\sqrt{2c_1}\tanh\left(\sqrt{\frac{c_1}{2}}(x-2c_1t)\right)-\sqrt{2c_2}\coth\left(\sqrt{\frac{c_2}{2}}(x-2c_2t)\right)\right]^2} \tag{9}$$

on the domain $x \in [-25\pi, 25\pi]$ and $t \in [-20, 20]$ with the $c_1 = \frac{1}{2}$ and $c_2 = 1$.

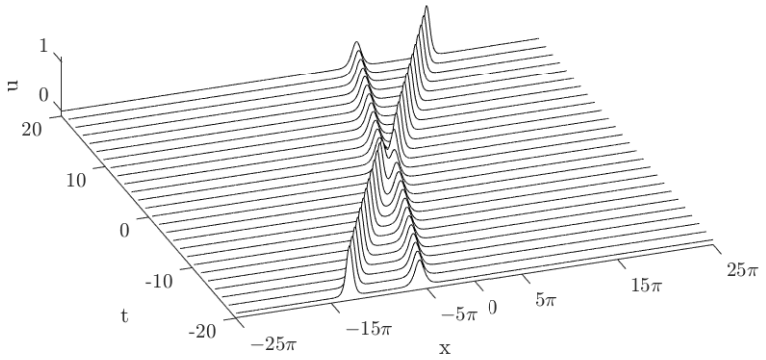

Figure 8: Numerical solution with the two soliton initial condition using HFD.

N		256	512	1024	2048	4096	8192
Error_∞	FD	1.0522	0.6129	0.1688	4.2093e-2	1.0503e-2	2.6251e-3
	HFD	0.4218	0.1007	2.5001e-2	6.2263e-3	1.5550e-3	3.8871e-4
Error_2	FD	9.5670e-3	3.5608e-3	6.6731e-4	1.1762e-4	2.0764e-5	3.6690e-6
	HFD	3.4934e-3	5.6951e-4	9.8930e-5	1.7416e-5	3.0755e-6	5.4368e-7
Error I	FD	3.8552e-3	8.5486e-4	1.7434e-4	3.3881e-5	3.6031e-6	4.1491e-7
	HFD	1.4758e-3	2.2313e-4	3.5536e-5	3.9718e-6	4.2650e-7	3.6889e-7
Time Step k	FD	0.2012	2.5150e-2	3.1437e-3	3.9296e-4	4.9121e-5	6.1401e-6
	HFD	1.6571e-2	2.0713e-3	2.5891e-4	3.2364e-5	4.0455e-6	5.0569e-7

Table 2: Numerical Results for FD2 and HFD4/2 for the two soliton test problem. Due to the double precision rounding errors, the accuracy levels for all the methods stagnate around the 10^{-7} levels as N increases.

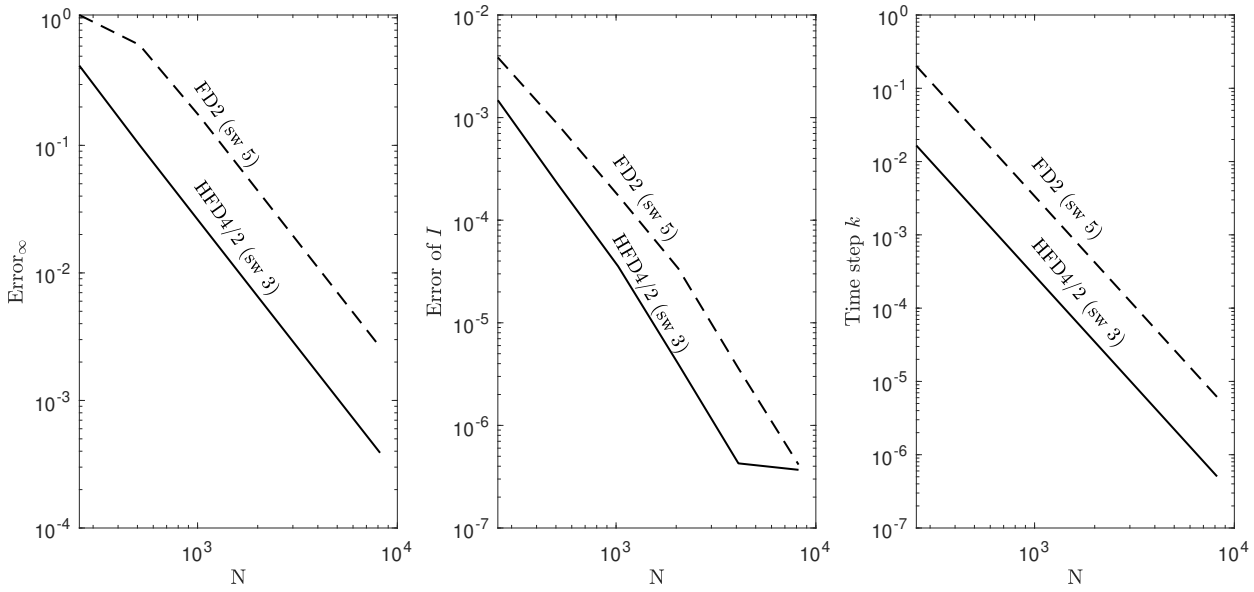

Figure 9: Numerical Results for FD2 and HFD4/2 for the two soliton test problem. The time steps k assume regular RK4.

Figure 8 shows the numerical solution when solving (5) with HFD4/2 and RK4 ($k \approx 4E-6$) using $N = 8192$ node points. Table 2 and Figure 9 show the different errors for both second order FD and HFD methods on the presented test problem. Here, the errors are the max error, the normalized ℓ_2 error, and the error of the integral $I = \int_{-25\pi}^{25\pi} u(x)dx$ of the solution at the final time compared to the initial time calculated with the trapezoidal rule. The time step used here in this test are the largest approximate time step reduced by 80% to ensure stability during testing.

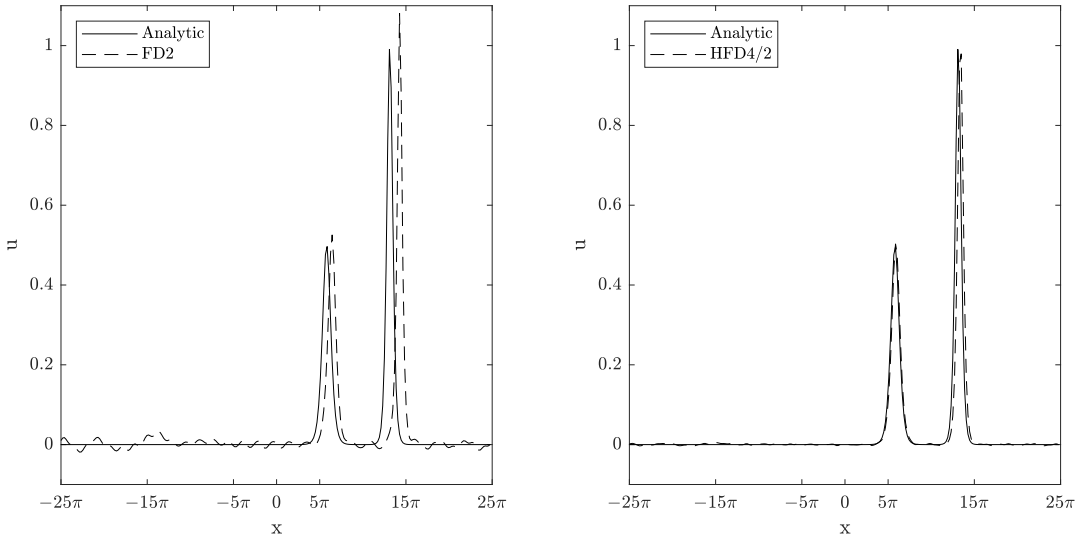

Figure 10: The numerical solution for FD2 and HFD4/2 using $N = 256$ at the final time compared against the analytic solution.

Figure 10 shows the trailing waves generated from time stepping the two soliton solution with FD2 and HFD4/2 using $N = 256$ nodes. HFD not only limits the phase shift of the solitons, but reduces trailing waves and conserves mass better than its FD counter part.

3.2. Step Solution / Dispersive Shock

Another important phenomena for the study of waves is the dispersive shock. For this test we use a hyperbolic tangent function to approximate the step function

$$u(x,0) = \begin{cases} 0 & x < 40 \\ -10 & x > 40 \end{cases}$$

According to [6] the moving fronts of the dispersive shocks will be mostly contained in $-20 \leq \frac{x}{t} \leq -\frac{10}{3}$. Figure 11 shows the numerical results of using the HFD12/10 stencil ($h = 0.1$) and RK4 ($k = 10^{-5}$).

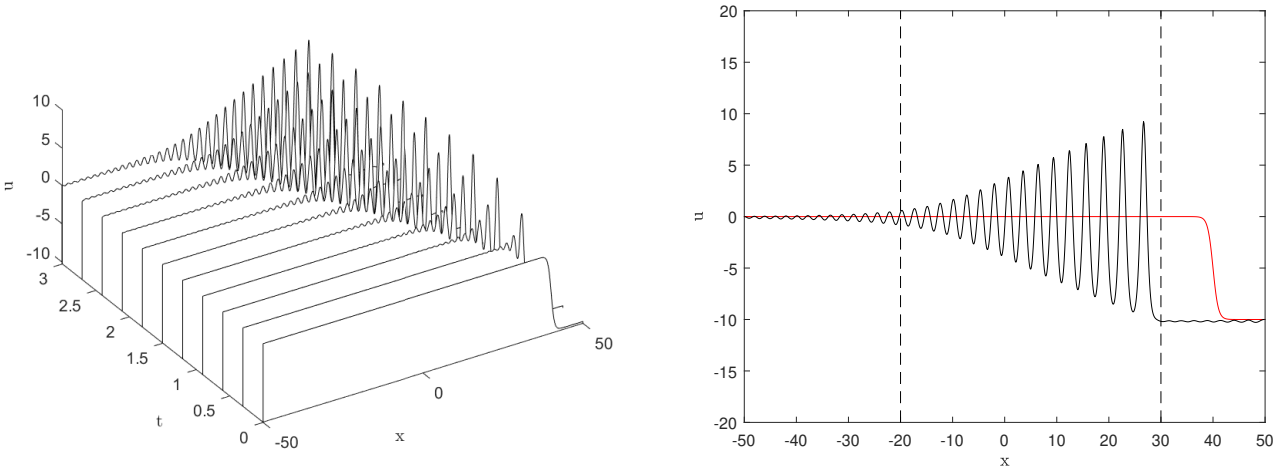

Figure 11: (Left) A time lapse when solving equation (6) with a step initial condition. (Right) The initial (red) and final (black) wave with the $-20 \leq \frac{x}{t} \leq -\frac{10}{3}$ interval at the final time illustrated with dashed black lines.

3.3. Instabilities of Wavetrain

We showed in Section 2.3 convergence rates for HFD stencils acting on a soliton at a 'final' time. Here we consider an accuracy test that includes nonlinear time evolution effects, but still does not require numerical time stepping. A stationary wavetrain provides one such test by analysing the instabilities of the solution. Following [6], the equation

$$u_t + 3u^2 u_x + u_{xxx} = 0 \tag{10}$$

is considered on the interval $[0, 128]$. A traveling wave train

$$u_0(x - ct) = a_1 \cos \nu(x - ct) + a_3 \cos 3\nu(x - ct) + a_5 \cos 5\nu(x - ct)$$
$$+ a_7 \cos 7\nu(x - ct) + a_9 \cos 9\nu(x - ct) + \cdots$$

is found with $a_1 = 0.3$ and $\nu = \frac{7\times 2\pi}{128}$. The Appendix Section 5.1 includes quad-precision values found for c and the leading coefficients a_i. By adjusting the equation to

$$u_t + \left(3u^2 - c\right) u_x + u_{xxx} = 0 \tag{11}$$

the wavetrain remains stationary.
The substitution $u(x,t) = u_0(x) + e^{\alpha t} v_0(x)$ is made into (11). Ignoring second order terms in v_0 yields

$$\alpha v_0 - c v_{0x} + 3u_0^2 v_{0x} + 6u_0 v_0 u_0 x + v_{0xxx} = 0 \tag{12}$$

When discretized in space, (12) amounts to a linear discrete eigenvalue problem. It theoretically has four unstable solutions, as detailed in [6]. Each of these has an exponential growth rate, as given in Table 2 of [6].

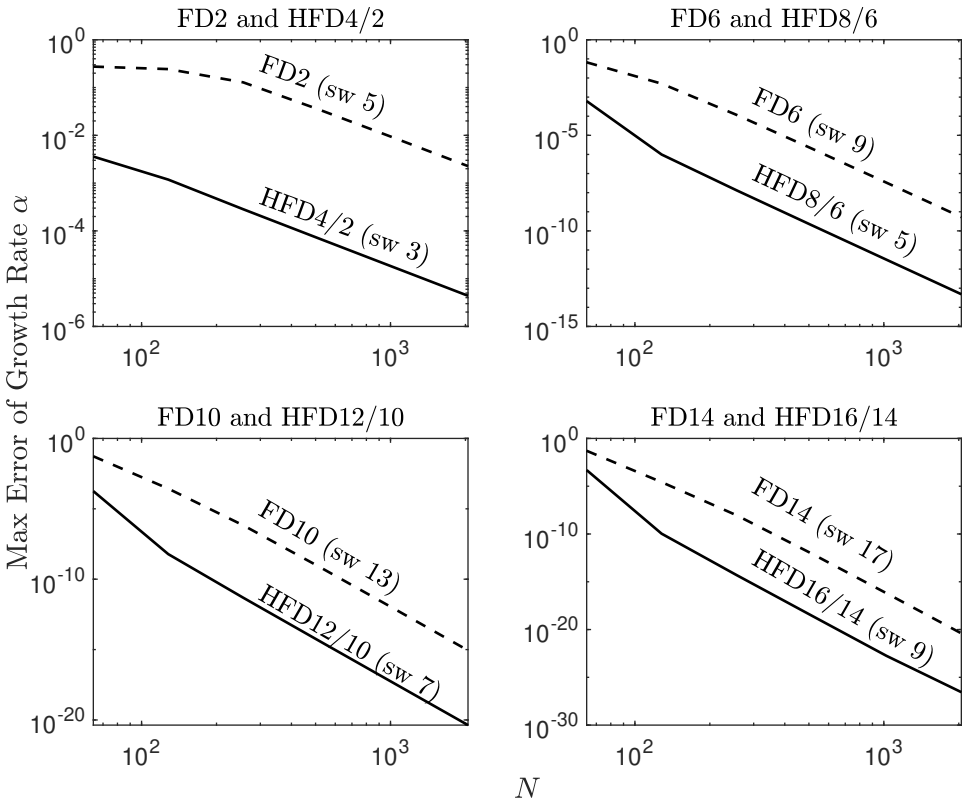

Figure 12: The largest error of the four growth modes α when using explict FD and $M = 1$ HFD for given total nodes N for different orders displayed on different vertical scales.

Figure 12 shows the decrease in growth rate errors α for similar orders of FD and $M = 1$ HFD stencils. The more compact HFD stencils tend to outperform their explicit FD counterparts. For the same order, the nodes required for FD to reach similar accuracy is about 32 times as many for 2nd order to 4 times as many for 14th order in comparison with HFD, well above the break even cost of 2.

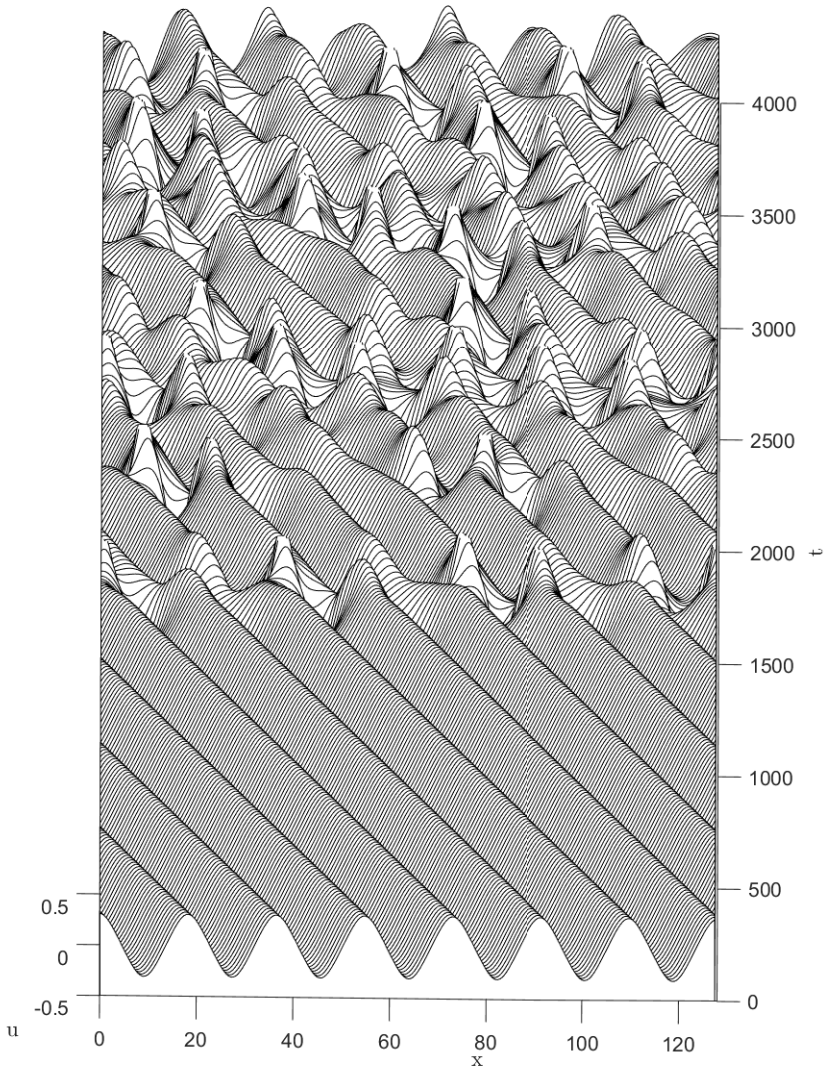

Figure 13: The wavetrain expansion u_0 time stepped with RK4 ($k = 10^{-3}$) and $M = 1$ HFD 2nd order ($h = 0.5$).

Figure 13 shows the instabilities of the nonlinear wavetrain similar to the results found in [6]. It is worth noting that the initial amplitude here is 0.3, while in Figure 24 of [6] the initial amplitude is 0.2. This may influence how the recurrence to a smooth state appears.

4. Conclusions

The introduction of derivative terms to form HFD stencils provides several benefits

i Compact stencil size,

ii Larger time steps allowed for same spatial error when compared with explicit FD,

iii Reduction of overall differentiation matrix size for same spatial error when compared with explicit FD,

iv Reduction of trailing waves [5],

v Flexibility of time stepper to use,

vi All stencils weights are pre-computed for fast time stepping.

HFD introduces all of these benefits while still maintaining the $\mathcal{O}\left(\frac{k}{h^3}\right)$ constraint using regular explicit time stepping (such as RK4).

The last row of Table 2 points towards a possible concern. For the same number of nodes in space, and using regular RK4, the largest time steps for stability have a smaller constant in the bound for HFD than for FD approximations of corresponding spatial accuracy orders. There is however an extensive literature available on how to greatly reduce (in fact, largely bypass) these conditions for the present class of PDEs, which are characterized by the term with the highest derivative in space appearing linearly (here u_{xxx} in (4)). Key methods for this are known as exponential time integrators, integrating factor methods, linearly implicit-explicit schemes, IMEX (IMplicit-EXplicit) schemes, etc. Comparative discussion (and application) of these different classes of time integrators fall outside the scope of the present (spatial discretization) study.

We finally note that the inevitable double precision truncation errors around $3 \cdot \frac{10^{-15}}{h^3}$ (due to the u_{xxx} term) favors very high order methods (allowing larger h). For very accurate calculations, quad precision (and higher) is needed. The relative benefit of higher accurate discretizations become then even more pronounced.

5. Appendix

5.1. Wavetrain Coefficients

i	a_i
1	0.3
3	0.0076962370822528404797734306607328 41
5	0.0001926253560344188728294145863829283
7	4.821050392780964377337746868597412e-6
9	1.206618232420510945841900045030304e-7
11	3.019938478512394844810756006944602e-9
13	7.558337980443643210889363986472 35e-11
15	1.891709828961749706245506784219818e-12
17	4.734593883271207309014615571380 3e-14
19	1.184979794274906676180623295168048e-15
21	2.965781537886902191339437401612 95e-17
23	7.422793344635058993956152718996 91e-19
25	1.857785556127487109496129965287 34e-20
27	4.649687809307592587847138636656 65e-22
29	1.163729400991210770136781875204157e-23
31	2.912595800562000575869999329872 94e-25
33	7.289679448010673490208287916998 25e-27
35	1.824469651590367170065563212578 14e-28
37	4.566304366761508016198047471120 18e-30
39	1.142860093711591474521289173012151e-31
41	2.860363849825442509518121826324 12e-33
43	7.158952699815704272665913421893 21e-35
45	1.791751205404368877400619493612 04e-36
47	4.484416250053743319594912571370 67e-38
49	1.122365029513103752832465108013 11e-39

Table 3: The a_i coefficients used in Section 3.3.

Table 3 shows the coefficients, a_i, found for the wave train $u_0(x,t)$ from Section 3.3 with $c = -0.048746186425196251038973442242661 44$.

5.2. Stencil Weights

Derivative Order p	M	Accuracy Order	Coordinates and weights (f, f', f'')			
			$x = 0$	$x = \pm 1$	$x = \pm 2$	$x = \pm 3$
2	1	4	$-4,0$	$2,\mp\frac{1}{2}$		
		8	$-5,0$	$\frac{64}{27},\mp\frac{8}{9}$	$\frac{7}{54},\mp\frac{1}{36}$	
		12	$-\frac{49}{9},0$	$\frac{39}{16},\mp\frac{9}{8}$	$\pm\frac{69}{250},\mp\frac{9}{100}$	$\frac{157}{18000},\mp\frac{1}{600}$
3	1	4	$0,-12$	$\pm\frac{15}{2},-\frac{3}{2}$		
		8	$0,-15$	$\pm\frac{88}{9},-\frac{8}{3}$	$\pm\frac{31}{144},-\frac{1}{24}$	
		12	$0,-\frac{49}{4}$	$\pm\frac{171}{16},-\frac{27}{8}$	$\pm\frac{963}{2000},-\frac{27}{200}$	$\pm\frac{167}{18000},-\frac{1}{600}$
	2	6	$0,-18,0$	$\pm\frac{105}{8},-\frac{33}{8},\pm\frac{3}{8}$		
		12	$0,-\frac{45}{2},0$	$\pm\frac{488}{27},-\frac{56}{9},\pm\frac{8}{9}$	$\pm\frac{601}{1728},\frac{3}{32},\pm\frac{1}{144}$	
4	1	2	$24,0$	$-12,\pm 6$		
		6	$\frac{99}{2},0$	$-\frac{64}{3},\pm 16$	$-\frac{41}{12},\pm\frac{3}{4}$	
		10	$\frac{3409}{54},0$	$-\frac{187}{8},\pm\frac{93}{4}$	$-\frac{3959}{500},\pm\frac{267}{100}$	$-\frac{7339}{27000},\pm\frac{47}{900}$
	2	6	$-144,0,-36$	$72,\mp\frac{39}{2},\frac{3}{2}$		
		12	$-207,0,-45$	$\frac{2816}{27},\mp 32,\frac{32}{9}$	$-\frac{43}{54},\pm\frac{29}{144},-\frac{1}{72}$	
5	1	2	$0,120$	$\mp 90,30$		
		6	$0,\frac{495}{2}$	$\mp\frac{560}{3},80$	$\mp\frac{455}{48},\frac{25}{8}$	
		10	$0,\frac{17045}{54}$	$\mp\frac{1865}{8},\frac{465}{4}$	$\mp\frac{9253}{400},\frac{267}{40}$	$\mp\frac{2603}{5400},\frac{47}{540}$
	2	4	$0,360,0$	$\mp 315,135,\mp 15$		
		10	$0,\frac{1305}{2},0$	$\mp\frac{17240}{27},\frac{2440}{9},\mp\frac{440}{9}$	$\pm\frac{20585}{864},-\frac{935}{144},\pm\frac{35}{72}$	

Table 4: Centered stencil weights for function, first, and second derivative values at node location x to approximate the third, fourth, and fifth derivative in the $M = 1$ and $M = 2$ cases. We recognize the first line of (3) in the (top) row $p = 3$, $M = 1$, order= 4 and the second line of (3) in the row $p = 4$, $M = 1$ and order= 2. The $M = 2$ stencils were not used in this study.

References

[1] D. Abrahamsen and B. Fornberg, *On the infinite order limit of Hermite-based finite difference schemes*, (submitted).

[2] D. Appelö and T. Hagstrom, *On advection by Hermite methods*, Pac. J. Appl. Math **4** (2011), no. 2, 125–139.

[3] Multiprecision Computing Toolbox for MATLAB X.X.X.XXXX, Advanpix LLC., Yokohama, Japan.

[4] B. Fornberg, *A Practical Guide to Pseudospectral Methods*, vol. 1, Cambridge University Press, 1998.

[5] B. Fornberg, *An algorithm for calculating Hermite-based finite difference weights*, IMA Journal of Numerical Analysis (2020).

[6] B. Fornberg and G.B. Whitham, *A numerical and theoretical study of certain nonlinear wave phenomena*, Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences **289** (1978), no. 1361, 373–404.

[7] I.A. Ganaie, S. Arora, and V.K. Kukreja, *Cubic Hermite Collocation Method for Solving Boundary Value Problems with Dirichlet, Neumann, and Robin Conditions*, International Journal of Engineering Mathematics **2014** (2014), 1–8.

[8] J. Goodrich, T. Hagstrom, and J. Lorenz, *Hermite methods for hyperbolic initial-boundary value problems*, Mathematics of Computation **75** (2006), no. 254, 595–630.

[9] C. Hermite and C. Borchardt, *Sur la formule d'interpolation de Lagrange*, Journal für die reine und angewandte Mathematik (Crelles Journal) **1878** (1878), no. 84, 70–79.

[10] C. Nwankwo, W. Dai, and R. Liu, *Compact Finite Difference Scheme with Hermite Interpolation for Pricing American Put Options Based on Regime Switching Model*, arXiv (2019), arXiv–1908.

[11] J. Qiu and C. Shu, *Hermite WENO schemes and their application as limiters for Runge–Kutta discontinuous Galerkin method: one-dimensional case*, Journal of Computational Physics **193** (2004), no. 1, 115–135.

[12] H.E. Salzer, *Hermite's general osculatory interpolation formula and a finite difference analogue*, Journal of the Society for Industrial and Applied Mathematics **8** (1960), no. 1, 18–27.

[13] G.B. Wright and B. Fornberg, *Scattered node compact finite difference-type formulas generated from radial basis functions*, Journal of Computational Physics **212** (2006), no. 1, 99–123.

[14] Y. Yang, F. Soleymani, M. Barfeie, and E. Tohidi, *A radial basis function-Hermite finite difference approach to tackle cash-or-nothing and asset-or-nothing options*, Journal of Computational and Applied Mathematics **368** (2020), 112523.