UCLA
COMPUTATIONAL AND APPLIED MATHEMATICS

A Cascade Algorithm for Solitary-Wave Solutions of the Benjamin Equation

Jianhong Shen

September 1998

CAM Report 98-42

Department of Mathematics
University of California, Los Angeles
Los Angeles, CA. 90024-1555

A CASCADE ALGORITHM FOR SOLITARY-WAVE SOLUTIONS OF THE BENJAMIN EQUATION

JIANHONG SHEN

COMPUTATIONAL AND APPLIED MATHEMATICS

7354 MATHEMATICAL SCIENCES BUILDING

UCLA

LOS ANGELES, CA 90095-1555

EMAIL: *JHSHEN@MATH.UCLA.EDU*

ABSTRACT. A cascade algorithm for solitary-wave solutions to the Benjamin equation is conctructed, implemented and analyzed both numerically and theoretically. The major advantages over the existing homotopy algorithm are its simplicity, efficiency, generality, and the many properties it reveals about the exact solution, whose closed form is out of our reach.

1. INTRODUCTION

This paper is an extension of the work of Albert, Bona and Rostrep [1] on the solitary-wave solutions to the Benjamin equation. We present a new cascade algorithm for the solitary-wave solutions, which were originally computed in [1] by a homotopy method. The algorithm we propose here is simpler, more efficient and general. Most remarkably, unlike the homotopy method, it also reveals many important properties of the exact solution, which has no closed form because of nonlinearity.

The presentation goes as follows. Section 2 contains a brief introduction to the Benjamin equation and their solitary-wave solutions. The general literature of the topic also can be read in [1–7]. The homotopic algorithm as proposed in [1] is reviewed in section 3, followed by the introduction of our new algorithm and numerical evidences of its success. Section 4 attempts to provide a theoretical

1991 *Mathematics Subject Classification.* 65P05, 76T05.

Key words and phrases. Solitary-wave, Benjamin equation, cascade algorithm, Power method, positive operators, Fröbenius-Perron theorem.

framework for our algorithm. We formulate it as a special nonlinear eigenvalue problem. An analogue to the Fröbenius-Perron Theorem is established as a quasi-justification for the convergence. However, as in most nonlinear processes, the exact convergence mechanism is far away from being thoroughly understood. We thereby raise an open problem for those who are interested in this topic.

2. Solitary-Wave Solutions of the Benjamin Equation

In modeling the wave propagation $\eta(x,t)$ along the interface of a 2-dimensional 2-fluid system, Benjamin derived the model equation

$$\eta_t + c_0(\eta_x + 2\gamma\eta\eta_x - \alpha L\eta_x - \beta\eta_{xxx}) = 0, \tag{1}$$

where, non-negative constants $c_0, \alpha, \beta, \gamma$ are non-dimensionalized parameters, depending either on the system configuration (e.g. the depths of fluids) or system physics (e.g. the densities, surface tension, and so on). The pseudo-differential operator L denotes $H\partial_x$ with H being the Hilbert transform

$$Hf(x) = P.V. \int_{-\infty}^{\infty} \frac{f(y)}{x-y}\, dy.$$

Much of the physics behind the equation has been extensively studied in [1, 4, 5]. So our exploration directly starts with Eq. (1). Noticing the resemblance to the KdV equation

$$\eta_t + \eta_x + \eta\eta_x + \eta_{xxx} = 0,$$

one searches for a traveling solitary-wave solution in the form of

$$\eta(x,t) = \Phi(x - c_0(1-C)t). \tag{2}$$

A substitution into Eq. (1) followed by one integration simplifies the Benjamin equation to

$$C\Phi - \alpha L\Phi - \beta\Phi'' + \gamma\Phi^2 = 0.$$

Here L and $'' = d^2/dX^2$ both act on

$$X = x - c_0(1-C)t.$$

After the scaling transform

$$\phi(X) = -\frac{\gamma}{C}\Phi(\sqrt{\frac{\beta}{C}}X),$$

one arrives at a much simpler equation which only depends on a single parameter $r = \alpha/2\sqrt{\beta C}$:

$$\phi - 2rL\phi - \phi'' - \phi^2 = 0. \tag{3}$$

All the above derivations can be found in [1].

From now on, we focus our attention on Eq. (3).

Benjamin [4, 5] showed the existence of a stable solution $\phi_r(X)$ for each $r \in [0,1)$. $\phi_r(X)$ is an even function of X with $\phi_r(0) = \max\phi_r(X) > 0$, and decays at large X like $O(1/X^2)$ (also see Section 3).

The appearance of the nonlinear term and the Hilbert transform in Eq. (3) makes the numerical computation a non-trivial task. In [1], the authors adopted a homotopy algorithm, which will be reviewed briefly in the next section together with our cascade algorithm. The new algorithm is simpler, more efficient and general, though the convergence analysis is also difficult as in most nonlinear algorithms. The motivation of the algorithm and our efforts to understand its convergence can be found in Section 4.

3. The Nonlinear Cascade Algorithm

3.1. **A brief review on the homotopy algorithm.** A complete study on the existence and stability of solitary-wave solutions to the Benjamin equation for r near 0 was carried out in [1]. Based on their analysis, the authors proposed a homotopy method based Newton-Raphson algorithm for Eq. (3) in the wave-number k-domain (i.e. Fourier transform). The idea is as follows. For $r = 0$, Eq. (3) is known to have a linearly scaled sech^2 solution . To compute the stable solution for some given $r_* \in [0,1)$, a discrete homotopic path is introduced:

$$r_0 = 0 < r_1 < \cdots < r_*.$$

ϕ_{r_m} is computed from $\phi_{r_{m-1}}$ by a Newton-Raphson on r in the k-domain.

The algorithm has at least two drawbacks: first, it requires one to know the exact closed form for a certain parameter, to establish an anchor for the whole program; and secondly, to find ϕ_{r_*}, one has to compute many intermediate ϕ_{r_m}'s, which are finally thrown away for a customer who only needs ϕ_{r_*}. These greatly constraint the generality and efficiency of the algorithm, but also have motivated our improvement work in this paper.

3.2. **The new cascade algorithm and its implementation.** As in [1], we work on the wave-number k-domain. The major advantage is that the Fourier transform of ∂_x followed by the Hilbert transform H becomes a simple multiplier $|k|\cdot$ in the k-domain. Define $c_r(k) = 1 - 2r|k| + k^2$, the "dispersion" relation of all linear terms in Eq. (3). Then, after the Fourier transform from the spatial variable X to the wave-number k, Eq. (3) simplifies to

$$c_r(k)\hat{\phi}(k) - \hat{\phi}(k) * \hat{\phi}(k) = 0. \tag{4}$$

Define $\sigma_r(k) = 1/c_r(k)$. Then

$$\hat{\phi}(k) = \sigma_r(k) \cdot \hat{\phi} * \hat{\phi}(k). \tag{5}$$

Notice the following properties

(i) $\sigma_r(k)$ is an even positive function for each $r \in [0, 1)$ and $\sigma_r(0) = 1$.
(ii) $\sigma_r(k)$ decays like $O(1/k^2)$ for large wave-numbers.
(iii) $\sigma_r(k) \in C^\infty[0, \infty)$, and unless $r = 0$, the left and right derivatives at $k = 0$ both do not vanish.

A typical $\sigma_r(k)$ is plotted in Figure 1. Pay attention to the two humps located at $\pm r$ that are greater than 1 (in fact equal to $1/(1-r^2)$). The closer is r near 1, the steeper and higher are the humps.

Our cascade algorithm solves $\hat{\phi}_r(k)$ based on Eq. (5).

The Cascade Algorithm :

(1) Take $f_0(k) = \sigma_r(k)$.

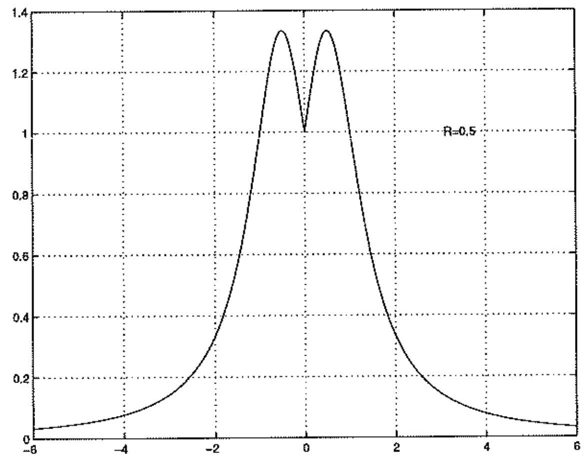

FIGURE 1. The graph of a typical $\sigma_r(k)$: r=0.5

(2) For $n = 1, 2, \cdots$

$$g_n(k) = \sigma_r(k) \cdot f_{n-1} * f_{n-1}(k),$$

$$f_n(k) = g_n(k)/g_n(0).$$

Step 1 initializes the cascade algorithm. Step 2 cascades according to the right hand side of Eq. (5), followed immediately by a normalization.

It is not difficult to see that each $f_n(k)$ shares the three properties of $\sigma_r(k)$ listed earlier. In fact, as to the decaying rate for large wave-numbers, we can show inductively that $f_n(k)$ decays exactly in the order of $1/k^{2(n+1)}$. Moreover, the normalization constant $g(0)$ at each step can be explained as

$$g_n(0) = \sigma_r(0) \cdot f_{n-1} * f_{n-1}(0) = \|f_{n-1}\|_{L^2}^2.$$

The numerical implementation is equally simple based on DCT (Discrete Cosine Transform). The essential work in the above algorithm obviously is on the computation of $Sf(k) = Tf(k)/Tf(0)$ with $Tf(k) = \sigma_r(k) \cdot f * f(k)$ for a given function $f(k)$ that decays very fast. Suppose we have DCT and IDCT (the inverse of DCT) available in our program (using MATLAB, for example). Then the following lines compute numerically $Sf(k)$ from a given even function $f(k)$.

Program :

- Take a discrete step size $h = 2^{-N}$.
  (*Typically,* $N = 10$*, depending on the precision requirement.*)
- Let $\mathbf{k}_h = 0 : h : K - h$ be the discrete grid.
  (*The truncation edge* K *again depends on the precision requirement. Thanking to the fast decaying rate of the* f_n*'s, we take* $L = 8$ *in all the following computation.*)
- $f_{\text{dct}} = \text{DCT}(f(\mathbf{k}_h))$.
  (*The Fourier transform of an even real function degenerates to its cosine transform on* $[0, \infty)$ *up to a multiplicative constant, which has been omitted. See the explanation below.*)
- $F_h = \text{IDCT}(f_{\text{dct}}^2)$.
  (*Convolution and multiplication are mutual images under the Fourier transform.*)
- $T_h f(\mathbf{k}_h) = \sigma_r(\mathbf{k}_h) F_h(\mathbf{k}_h)$, followed by $S_h f(\mathbf{k}_h) = T_h f(\mathbf{k}_h)/T_h f(0)$.

Here, with a subscript h, each quantity denotes the discrete approximation to its continuous prototype. Please notice that since we linearly normalize the result in the final step, all intermediate multiplicative constants thus have been safely ignored. This adds simplicity to the numerical implementation.

3.3. **Performance of the cascade algorithm.** Figure 2 shows the log-plot of the successive errors

$$e_n = \|f_n - f_{n-1}\|_{L^\infty}.$$

The convergence is linear. Figure 3 shows $f_{10}(k)$, for $r = 0.1, 0.3, 0.6, 0.9$. We have taken $N = 10$ (so $h \approx 0.001$) and $K = 8$.

The above numerical evidences clearly show that f_n's do converge. Denote the limiting function by $f_\infty(k)$. From the algorithm,

$$f_\infty(k) = \sigma_r(k) \cdot \frac{f_\infty}{\|f_\infty\|_{L^2}} * \frac{f_\infty}{\|f_\infty\|_{L^2}}.$$

Therefore, $\hat{\phi}(k) = f_\infty(k)/\|f_\infty\|_{L^2}^2$ solves Eq. (5)

$$\hat{\phi}(k) = \sigma_r(k) \cdot \hat{\phi} * \hat{\phi}(k).$$

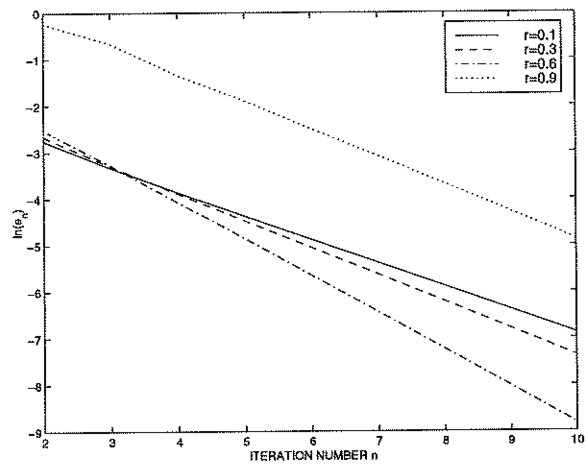

FIGURE 2. Linear convergence of the cascade algorithm: the natural logarithm of e_n versus n for $r = 0.1, 0.3, 0.6, 0.9$.

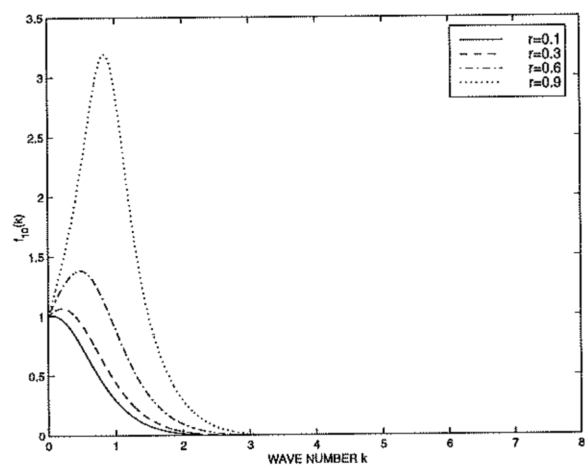

FIGURE 3. The output $f_{10}(k)$ at the tenth step for various r.

In Figure 4, by taking f_{10} for f_∞, we have plotted $\hat{\phi} = \hat{\phi}_r(k)$ for $r = 0.1, 0.3, 0.6, 0.9$. In Figure 5, by taking f_{20} for f_∞, we plotted the physical solitary-wave $\phi_{0.9}(X)$. Notice the extraordinary fact that these solitary waves are localized in both the spatial X-domain and wave-number k-domain, very much like wavelets. The results coincide with those in [1] obtained from the homotopy algorithm (except that we use no scaling factors).

The algorithm is robust in the sense that it can start with quite flexible initial trying function $f_0(k)$, as long as $f_0(k)$ is a non-negative even continuous function

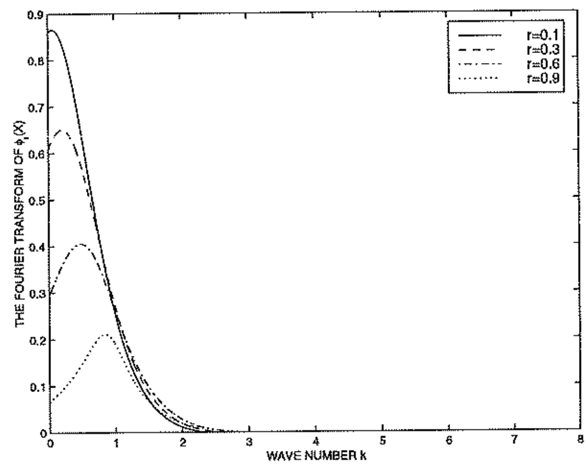

FIGURE 4. $\hat{\phi}_r(k)$ for $r = 0.1, 0.3, 0.6$ and 0.9.

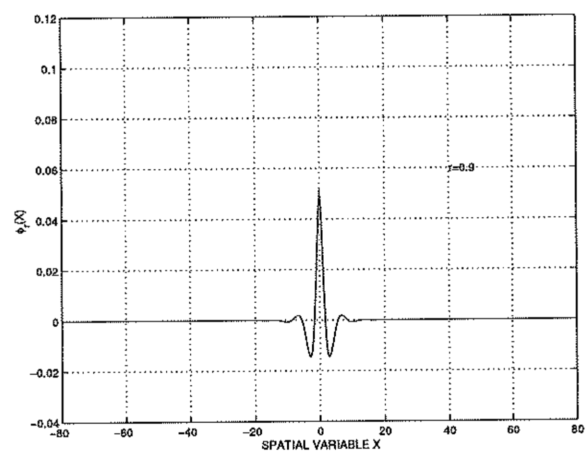

FIGURE 5. The solitary wave $\phi_r(X)$ with $r = 0.9$.

and decays reasonably fast such as in the order of $O(1/|k|^{1+\epsilon})$ for some $\epsilon > 0$. For examples, readers can try $f_0 = e^{-|k|}$, or $f_0 = 1/(1+k^2)$, and so on.

The choice of $f_0 = \sigma_r(k)$ is convenient since one needs no extra data other than the necessary one $\sigma_r(k)$. Also, f_n's obtained from this initial function share many important properties of the exact solution f_∞ or $\hat{\phi}_r(k)$. We now address these properties.

3.4. **Properties shared by both $\hat{\phi}_r(k)$ and $f_n(k)$.** As we have pointed out earlier, f_n's are all non-negative even functions. Therefore, $f_\infty(k)$ and $\hat{\phi}_r(k)$ must be so, too. The evenness of $\hat{\phi}(k)$ further passes to the physical solitary wave $\phi_r(X)$. The non-negativity, meanwhile, implies that $\phi_r(X)$ is like the characteristic function of a random variable. Especially, we derive

$$\phi_r(0) = \max_{X\in\mathbb{R}} \phi_r(X) > 0,$$

which was obtained by Benjamin earlier (also see Section 2).

The second important property shared by both $f_n(k)$ and $\hat{\phi}_r(k)$ is their fast decaying rate. We have seen that $f_n(k)$ decays in the exact order of $1/k^{2(n+1)}$ due to the action of $\sigma_r(k)$ in each step of the algorithm. Therefore, one shall expect that $f_\infty(k)$ and $\hat{\phi}_r(k)$ decay faster than any polynomial rate. This indeed is true. Suppose, otherwise, $\hat{\phi}_r(k)$ decays in the exact order of $1/|k|^d$ for some finite positive number d. Then, from

$$\hat{\phi}_r(k) = \sigma_r(k) \cdot \hat{\phi}_r * \hat{\phi}_r(k),$$

and the fact that $\sigma_r(k) = O(1/k^2)$, we know $\hat{\phi}_r(k)$ must also decay in the exact order of $1/|k|^{d+2}$. This contradicts to the meaning of d! In [1], the authors already noticed this fast decaying property.

The final but the most interesting point is the regularity of $f_n(k)$ and $\hat{\phi}_r(k)$. We have mentioned that $f_n(k) \in C^\infty[0,\infty)$, and unless $r = 0$, $f_n'(0^+) \neq 0$. This means that f_n is not differentiable at $k = 0$. Evidences in [1, 4, 5] also support strongly that $\hat{\phi}_r(k)$ shares these properties, which, of course, is not a surprise to the cascade algorithm. Moreover, based on this assumption, we can now successfully show the famous $1/X^2$ decaying rate of $\phi_r(X)$ as formally obtained by Benjamin. Also, we derive a formula for the exact number $D_r = \lim_{X\to\infty} X^2\phi_r(X)$ introduced in [1].

First, we can easily compute $\hat{\phi}_r'(0^+)$ from Eq. (5) by using the facts that differentials commute with convolutions, and that $f * g(0) = 0$ if f and g are odd and even real-value functions, respectively. We obtain

$$\hat{\phi}_r'(0^+) = 2r\hat{\phi}_r(0).$$

Since $\hat{\phi} = f_\infty/\|f_\infty\|^2$ (with L^2-norm), we have

$$\hat{\phi}_r'(0^+) = \frac{2r}{\|f_\infty\|^2}.$$

Thus $\hat{\phi}_r'(0^+)$ can be evaluated numerically from our algorithm.

Based on this information, let us now compute D_r.

$$\begin{aligned}\phi_r(X) &= \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{\phi}_r(k)e^{ikX}\,dk \\ &= \frac{1}{\pi}\int_0^{\infty} \hat{\phi}_r(k)\cos(kX)dk \\ &= -\frac{1}{\pi}\int_0^{\infty} \hat{\phi}_r'(k)\frac{\sin(kX)}{X}dk \\ &= -\frac{\hat{\phi}_r'(0^+)}{\pi X^2} - \frac{1}{\pi}\int_0^{\infty} \hat{\phi}_r'(k)\frac{\cos(kX)}{X^2}dk.\end{aligned}$$

Therefore,

$$D_r = -\frac{\hat{\phi}_r'(0^+)}{\pi} = -\frac{2r}{\pi\|f_\infty\|^2},$$

whose truthness has been easily verified numerically. Notice that all the above derivations are only based on a single assumption, i.e., $\hat{\phi}_r(k) \in C^\infty[0,\infty)$. The numerical verification of the D_r formula gives a strong partial support, along with the cascade algorithm, of this assumption, which the author believes has to be true.

These are the major "semi-theoretical" contributions of the cascade algorithm. Besides the many numerical advantages, it has brought us deeper insights into the exact solitary-wave solutions.

However, analyzing its convergence mechanism is still a tough task.

4. Understanding the Convergence Mechanism

Even though the cascade algorithm only takes a few lines, the analysis on its convergence is much more difficult due to the nonlinearity of $f * f$. The difficulty is not completely a surprise to those of us who have known well the behavior of the logistic cascade on the interval: $x_n = \mu x_{n-1}(1 - x_{n-1})$. Any simple nonlinearity, such as the quadratic form in the logistic map and the self-convolution in our example, can make analysis exponentially difficult.

In the following, we present an analogue to linear positive operators and the Fröbenius-Perron Theorem. This analogue has initiated the algorithm and also may provide some useful clue to the convergence mechanism. However, a rigorous and complete investigation is still an open problem.

4.1. **A topological "explanation".** First, we try a qualitative explanation. It is a well-known fact that the convolution $f * f(k)$ spreads and mollifies f. The famous example is to take $f(k) = \mathbf{1}_{(-1,1)}(k)$ to be the indicator of $(-1, 1)$. Then $f * f(k)$ is a piece-wise linear *continuous* function supported on $(-3, 3)$. On the other hand, the multiplier $\sigma_r(k)$ behaves like a "truncation" operator because of its fast decaying rate $1/k^2$. This truncating effect balances the spreading effect at each step of the algorithm, which eventually causes each output f_n to be more and more adapted to the two effects. A perfect balance occurs at the exact solution.

This "topological" explanation is far away from being satisfactory. Yet, the author does believe that it outlines the correct forces that control all the intermediate steps of the cascade algorithm.

4.2. **Positive linear operators and the Power method.** A more algebraic "interpretation" is built upon an analogue to a familar example of positive linear operators.

Let $V = \mathbb{R}^n$ be an n-dimensional normed linear space with norm $\|\cdot\|$. Define

$$V^+ = \{\mathbf{x} = (x_1, \cdots, x_n) \ : \ x_i > 0\}.$$

A linear operator $A : V \to V$ is said to be positive if $A(\text{closure}(V^+)\backslash\{\mathbf{0}\}) \subset V^+$. Part of the famous classical Fröbenius-Perron Theorem says:

"*Any non-negative linear operator A must have a* positive *eigenvalue λ_m, which is algebraically simple and can choose its eigenvector in V^+. (Moreover, λ_m is the spectral radius of A.)* "

A theoretic proof of the existence can be derived from the topological Brouwer fixed point theorem. A cascade algorithmic approach, on the other hand, is more elementary and easier to justify.

Algorithm for λ_m:

(i) Take any $\mathbf{x}_0 \in V^+$ with $\|\mathbf{x}_0\| = 1$.

(ii) For $n = 1, 2, \cdots$,

$$\mathbf{y}_n = A\mathbf{x}_{n-1}$$
$$\mathbf{x}_n = \frac{\mathbf{y}_n}{\|\mathbf{y}_n\|}.$$

Then $\mathbf{x}_n$ converges, say, to $\mathbf{x}_\infty \in V^+$. And $\mathbf{x}_\infty$ is the eigenvector corresponding to the positive eigenvalue $\lambda_m = \|A\mathbf{x}_\infty\|$.

Convergence in this linear case is easy to understand. The algorithm is essentially the Power method (Golub and Van Loan [8]): if the initial vector has a non-zero component in the eigenspace of λ_m, then the successive actions of A (followed by the normalizations at each step) will eventually "filter" out all the other components while keeping this one invariant.

4.3. **The analogue.** Define

$$W = \{f(k) : \ f(k) = f(-k), f \in C^\infty[0,\infty), f^{(m)}(k) = o(k^{-M}), \text{for all } m, M > 0\}$$

where, m and M denote nonnegative integers, and the order constraint is put near $k = \infty$. Then W is obviously a linear space, and can be naturally embedded in any L^p for $p \geq 1$. For convenience, embed it in L^2 and the norm is simply denoted by $\|\cdot\|$. Let $\sigma_r(k)$ be given as before and

$$T_r f(k) = \sigma_r(k) \cdot f * f(k).$$

It is not difficult to show that $T_r(W) \subset W$. Hence, from now on, we restrict T_r on W.

Also define a special class of nonlinear eigenvalue problem as follows, which we call *a p-degree nonlinear eigenvalue problem.*

Let V be a normed linear space with norm $\|\cdot\|$. $T : V \to V$ is a p-degree ($p > 0$) operator if for any $a > 0$ and $\mathbf{x} \in V$,

$$T(a\mathbf{x}) = a^p T(\mathbf{x}).$$

Obviously, T_r is of 2-degree.

For a p-degree operator T on V, $(\mathbf{x}_\lambda, \lambda)$ is said to be an eigenpair of T if $\mathbf{x}_\lambda \neq \mathbf{0}$, and

$$T\mathbf{x}_\lambda = \lambda \mathbf{x}_\lambda.$$

Proposition 1. *If $(\mathbf{x}_\lambda, \lambda)$ is an eigenpair of a p-degree operator T, then for all $a > 0$, $(a\mathbf{x}_\lambda, a^p\lambda)$ is also an eigenpair.*

The proof is trivial. Notice that for a linear problem, one eigenvalue can have more than one linearly independent eigenvectors. In the opposite, for a nonlinear problem ($p \neq 1$), one eigen-direction can correspond to more than one eigenvalues! Especially,

Corollary 1. *Suppose T is a p-degree operator with $p \neq 1$. Then the following equation has a non-zero solution*

$$T\mathbf{x} = \mathbf{x}$$

if and only if T has at least one positive eigenvalue.

In our case, T_r is a 2-degree nonlinear operator on S. Our initial goal is to solve the solitary-wave equation (5)

$$T_r\hat{\phi}_r = \hat{\phi}_r.$$

So the task now is reduced to finding one positive eigenvalue λ of T_r: $T_r f(k) = \lambda f(k)$. This change of attitude has played a crucial role in the construction of the cascade algorithm.

Define

$$W^+ = \{f \in W \; : \; f(k) \geq 0 \text{ but } f \neq \mathbf{0}\}.$$

This is clearly an analogue of V^+ we defined earlier. Indeed, it is trivial to observe that W^+ is T_r invariant. Therefore, T_r is a "positive" operator. Inspired by the linear case, the following cascade algorithm is born for finding a positive eigenvalue of T_r:

Algorithm analogous to the Power method:

(i) Take any $f_0^* \in W^+$ and have its norm normalized to 1.

(ii) For $n = 1, 2, \cdots$

$$g_n^* = T_r f_{n-1}^*$$
$$f_n^* = \frac{g_n^*}{\|g_n^*\|}.$$

This is in fact the cascade algorithm we have proposed! The gap can be filled in by choosing $f_0^* = f_0/\|f_0\|$. Then $f_n^* = f_n/\|f_n\|$ for all $n = 1, 2, \cdots$. Here f_n's are outputs from the original cascade algorithm. The reason we use the previous version is that the normalization constant $g_n(0)$ at each step is much easier to compute than $\|g_n^*\|$.

The previous numerical evidence (Figure 2) shows the linear convergence rate of the algorithm, which offers another analogue. In the linear case, the error sequence is linearly controlled by $\rho = |\lambda_2|/\lambda_m < 1$ with λ_2 being the eigenvalue of the second largest magnitude. However, for this nonlinear case, we have no closed form for the convergence factor ρ, whose determination (and even the proof of its existence) raises another open problem.

4.4. **Summary.** We now give a short summary for this section and some closing remarks for the paper.

By noticing the degree of the nonlinear operator T_r, we are able to convert the original target equation (5) to a nonlinear eigenvalue problem. Based on the positivity of T_r and an analogue to linear positive operators, we show that the cascade algorithm is simply an intuitive transplanting of the linear Power method. We believe that the space W and W^+ introduced in this paper will have significant meanings for further work on this topic.

In spite of the doubtless success of our cascade algorithm and its many advantages, the nonlinear spreading and truncating mechanism is still not quantitatively well-understood, and linear positive operators only provide some helpful insights but not a complete answer. The author thereby challenge the readers and himself for giving further clarifications for the convergence of this three-line cascade algorithm of a nonlinear map. The author believes that any further development will help understand the underlying physics of the equation, as well as add new knowledge to general nonlinear dynamics.

References

[1] J. P. Albert, J. L. Bona, and J. M. Restrepo. Solitary-wave solutions of the Benjamin equation. To appear in *SIAM J. Appl. Math;* UCLA CAM Report 97-23, 1998.

[2] T. B. Benjamin. Interal waves of permanent form in fluids of great depth. *J. Fluid Mech.*, 29:559–592, 1967.

[3] T. B. Benjamin. The stability of solitary waves. *Proc. Royal Soc. London,* A 328:153–183, 1972.

[4] T. B. Benjamin. A new kind of solitary wave. *J. Fluid Mech.*, 245:401–411, 1992.

[5] T. B. Benjamin. Solitary and periodic waves of a new kind. *Philos. Trans. Roy. Soc. London,* A 354:1775–1806, 1996.

[6] T. B. Benjamin, J. L. Bona, and D. K. Bose. Solitary-wave solutions of nonlinear problems. *Philos. Trans. Roy. Soc. London,* A 340:195–244, 1990.

[7] H. Chen and J. L. Bona. Existence and asymptotic properties of solitary-wave solutions of Benjamin-type equations. Preprint, 1997.

[8] G. H. Golub and C. F. Van Loan. *Matrix Computations.* Johns Hopkins Press, 1983.