A Parallel Algorithm for an Investigation of Self-Focusing Singularity of Higher KdV Equations*

Thiab R. Taha†

Abstract. An accurate numerical scheme based on the inverse scattering transform (IST) is used to investigate a self-focusing singularity, which means the solution blows up in finite time, in the higher Korteweg-de Vries (KdV) equation:

$$u_t + u^4 u_x + u_{xxx} = 0$$

It has been shown in part by the author that the IST schemes compare very favorably with other known numerical methods. The implementation of the IST numerical scheme leads to a huge periodic banded system of equations to be solved at each time step. A parallel algorithm for the proposed IST scheme is designed and implemented on an intel iPSC/2 hypercube, and the numerical results are discussed.

1. Introduction. It has been shown that the higher nonlinear Schrödinger (NLS) equation

$$iq_t + q_{xx} + A|q|^{2p}q = 0, \quad p \geq 2 \tag{1}$$

under certain conditions admits a self-focusing singularity [1], which means that the solution of Eq. (1) blows up in finite time. This suggests that the higher nonlinear KdV equation

$$u_t + Au^p u_x + u_{xxx} = 0, \quad p > 3 \tag{2}$$

*This research has been supported in part by the Department of Energy Grant No. DE-FG09-90ER25088 and the National Science Foundation Grant No. CCR-8717033.

†Department of Computer Science, The University of Georgia, Athens, Georgia 30602.

has a self-focusing singularity [2].

Recently, there has been a lot of theoretical and numerical research in order to investigate this phenomenon (see Bona et al. [3], and the references therein). Numerical simulations of solutions of Eq. (2) (see Fornberg & Whitham [4], Bona et al [5]) confirm that its solitary-wave solutions are unstable if $p \geq 4$, and in fact, that neighboring solutions emanating from smooth initial data appear to form singularities in finite time. This paper deals with a numerical investigation of the blow-up for the higher KdV equation

$$u_t + u^4 u_x + u_{xxx} = 0 \tag{3}$$

using an accurate numerical scheme based on the IST. The proposed numerical scheme is based on an IST numerical scheme derived by Taha and Ablowitz for the KdV and MKdV equations. Experiments have shown that the IST numerical schemes compare very favorably with other numerical methods [6,7].

In order for the singularity to be properly resolved, the mesh sizes in the directions of x and t have to be taken very small. It is to be noted that the implementation of the IST numerical scheme leads to a huge periodic banded system to be solved at each time step. Therefore the implementation of the proposed numerical scheme on a serial computer requires a large amount of computing time.

In this paper a parallel algorithm for the above scheme is designed and implemented on an Intel iPSC/2 hypercube, and the numerical results are discussed.

2. The proposed numerical scheme. The proposed numerical scheme for Eq. (3) which is based on the IST for the KdV equation [6]

$$u_t + 6uu_x + u_{xxx} = 0 \tag{4}$$

is:

$$\begin{aligned} \frac{u_n^{m+1} - u_n^m}{\Delta t} &= \frac{1}{2(\Delta x)^3}[u_{n-1}^{m+1} - 3u_n^{m+1} + 3u_{n+1}^{m+1} - u_{n+2}^{m+1} \\ &+ u_{n-2}^m - 3u_{n-1}^m + 3u_n^m - u_{n+1}^m] - \frac{1}{4\Delta x}[(u_n^m)^2 \\ &- (u_n^{m+1})^2 + \frac{1}{3}\{u_{n+1}^{m+1}(u_n^{m+1} + u_{n+1}^{m+1} + u_{n+2}^{m+1}) \\ &- u_{n-1}^m(u_n^m + u_{n-1}^m + u_{n-2}^m)\}] \left(\frac{u_n^m + u_n^{m+1}}{2}\right)^3 \end{aligned} \tag{5}$$

The truncation error of this scheme is $O((\Delta t)^2) + O((\Delta x)^2)$. This scheme is applied to Eq. (3) subject to a Gaussian profile of the form

$$u(x,0) = \eta e^{-\left(\frac{x}{\gamma}\right)^2}, \tag{6}$$

with $\eta = 3$, and $\gamma = 8$ as an initial condition, and periodic boundary conditions on the interval [-40, 40] are imposed.

3. A parallel implementation of the proposed scheme. Eq. (5) can be written as

$$-u_{n-1}^{m+1} + (3+\epsilon)u_n^{m+1} - 3u_{n+1}^{m+1} + u_{n+2}^{m+1} = B_n, \tag{7}$$

where

$$\epsilon = \frac{2(\Delta x)^3}{\Delta t},$$

and

$$\begin{aligned} B_n &= -u_{n+1}^m + (3+\epsilon)u_n^m - 3u_{n-1}^m + u_{n-2}^m \\ &- \frac{1}{2}(\Delta x)^2[(u_n^m)^2 - (u_n^{m+1})^2 \\ &+ \frac{1}{3}\{u_{n+1}^{m+1}(u_n^{m+1} + u_{n+1}^{m+1} + u_{n+2}^{m+1}) \\ &- u_{n-1}^m(u_n^m + u_{n-1}^m + u_{n-2}^m)\}] \left(\frac{u_n^m + u_n^{m+1}}{2}\right)^3 \end{aligned} \tag{8}$$

One way to implement this scheme is to solve a periodic banded system of equations at each time step:

$$\begin{bmatrix} \alpha & -3 & 1 & & & & & & -1 \\ -1 & \alpha & -3 & 1 & & & & & \\ & -1 & \alpha & -3 & 1 & & & & \\ & & & \cdot & \cdot & \cdot & \cdot & & \\ & & & & \cdot & \cdot & \cdot & \cdot & \\ & & & & \cdot & \cdot & \cdot & \cdot & \cdot \\ & & & & & -1 & \alpha & -3 & 1 \\ 1 & & & & & & -1 & \alpha & -3 \\ -3 & 1 & & & & & & -1 & \alpha \end{bmatrix} \begin{bmatrix} u_{-N} \\ u_{-N+1} \\ u_{-N+2} \\ \cdot \\ \cdot \\ \cdot \\ u_{N-3} \\ u_{N-2} \\ u_{N-1} \end{bmatrix}$$

$$= \begin{bmatrix} B_{-N} \\ B_{-N+1} \\ B_{-N+2} \\ \cdot \\ \cdot \\ \cdot \\ B_{N-3} \\ B_{N-2} \\ B_{N-1} \end{bmatrix} \tag{9}$$

where $\alpha = 3 + \epsilon$. The above system can be solved on a hypercube by using a modified version of an efficient parallel algorithm for periodic tridiagonal systems [8].

3.1. Description of the algorithm. Consider the following general form of Eq. (9) $AX = B$, where

$$A = \left[\begin{array}{ccccccccccc}
a_1 & c_1 & d_1 & & & & & & & & b_1 \\
b_2 & a_2 & c_2 & d_2 & & & & & & & \\
& b_3 & a_3 & c_3 & d_3 & & & & & & \\
& & \cdot & \cdot & \cdot & \cdot & & & & & \\
& & & \cdot & \cdot & \cdot & \cdot & & & & \\
& & & & \cdot & \cdot & \cdot & \cdot & & & \\
& & & & & & & b_{N-2} & a_{N-2} & c_{N-2} & d_{N-2} \\
d_{N-1} & & & & & & & & b_{N-1} & a_{N-1} & c_{N-1} \\
c_N & d_N & & & & & & & & b_N & a_N
\end{array}\right],$$

$$X = \left[\begin{array}{c} x_1 \\ x_2 \\ x_3 \\ \cdot \\ \cdot \\ \cdot \\ x_{N-2} \\ x_{N-1} \\ x_N \end{array}\right], \; and \; B = \left[\begin{array}{c} B_1 \\ B_2 \\ B_3 \\ \cdot \\ \cdot \\ \cdot \\ B_{N-2} \\ B_{N-1} \\ B_N \end{array}\right].$$

Assume that the number of processors p divides the order N of the above system. Partition the original system into p subsystems; each processor works on $\frac{N}{p}$ equations. The first subsystem has the form:

$$\left[\begin{array}{ccccccccc}
a_1 & c_1 & d_1 & & & & & 0 & b_1 \\
b_2 & a_2 & c_2 & d_2 & & & & 0 & 0 \\
& \cdot & \cdot & \cdot & \cdot & & & & \\
& & \cdot & \cdot & \cdot & \cdot & & & \\
& & & \cdot & \cdot & \cdot & \cdot & & \\
& & & b_{m-1} & a_{m-1} & c_{m-1} & d_{m-1} & & \\
& & & & b_m & a_m & c_m & d_m & 0
\end{array}\right]
\left[\begin{array}{c} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_{m-1} \\ x_m \\ x_{m+1} \\ x_{m+2} \\ x_N \end{array}\right]
=
\left[\begin{array}{c} B_1 \\ B_2 \\ \cdot \\ \cdot \\ \cdot \\ B_{m-1} \\ B_m \end{array}\right]$$

the i-th subsystem $(1 < i < p)$ has the form

$$\begin{bmatrix} b_j & a_j & c_j & d_j & & & & & \\ & b_{j+1} & a_{j+1} & c_{j+1} & d_{j+1} & & & & \\ & & \cdot & \cdot & \cdot & \cdot & & & \\ & & & \cdot & \cdot & \cdot & \cdot & & \\ & & & & b_{r-1} & a_{r-1} & c_{r-1} & d_{r-1} & \\ & & & & & b_r & a_r & c_r & d_r \end{bmatrix} \begin{bmatrix} x_{j-1} \\ x_j \\ x_{j+1} \\ \cdot \\ \cdot \\ \cdot \\ x_{r-1} \\ x_r \\ x_{r+1} \\ x_{r+2} \end{bmatrix} = \begin{bmatrix} B_j \\ B_{j+1} \\ \cdot \\ \cdot \\ \cdot \\ B_{r-1} \\ B_r \end{bmatrix}$$

and the last subsystem has the form

$$\begin{bmatrix} 0 & 0 & b_j & a_j & c_j & d_j & & & \\ 0 & 0 & 0 & b_{j+1} & a_{j+1} & c_{j+1} & d_{j+1} & & \\ & & & & \cdot & \cdot & \cdot & \cdot & \\ & & & & & \cdot & \cdot & \cdot & \cdot \\ d_{N-1} & & & & & & b_{N-1} & a_{N-1} & c_{N-1} \\ c_N & d_N & 0 & & & & & b_N & a_N \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_{j-1} \\ x_j \\ x_{j+1} \\ \cdot \\ \cdot \\ \cdot \\ x_{N-1} \\ x_N \\ \\ \\ \end{bmatrix} = \begin{bmatrix} B_j \\ B_{j+1} \\ \cdot \\ \cdot \\ \cdot \\ B_{N-1} \\ B_N \end{bmatrix}$$

where $j = (i-1)m+1, m = N/p$, and $r = im$. Each of the above subsystems has three variables more than the number of equations. Following the algorithm given in [8,9], we introduce three variables as parameters and then the above subsystems can be written as

$$\begin{aligned} A^1X^1 &= B^1 - x_N q^1 - x_{m+1}r^1 - x_{m+2}s^1, \\ A^iX^i &= B^i - x_{j-1}q^i - x_{r+1}r^i - x_{r+2}s^i, \ 1 < i < p, \\ A^pX^p &= B^p - x_{j-1}q^p - x_1 r^p - x_2 s^p, \end{aligned} \tag{10}$$

$$A^1 = \begin{bmatrix} a_1 & c_1 & d_1 & & & & \\ b_2 & a_2 & c_2 & d_2 & & & \\ & \cdot & \cdot & \cdot & \cdot & & \\ & & \cdot & \cdot & \cdot & \cdot & \\ & & & \cdot & \cdot & \cdot & \cdot \\ & & & & b_{m-1} & a_{m-1} & c_{m-1} \\ & & & & & b_m & a_m \end{bmatrix}, \quad X^1 = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_{m-1} \\ x_m \end{bmatrix}, \quad B^1 = \begin{bmatrix} B_1 \\ B_2 \\ \cdot \\ \cdot \\ \cdot \\ B_{m-1} \\ B_m \end{bmatrix},$$

$$q^1 = \begin{bmatrix} b_1 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \\ 0 \end{bmatrix}, \quad r^1 = \begin{bmatrix} 0 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ d_{m-1} \\ c_m \end{bmatrix}, \quad s^1 = \begin{bmatrix} 0 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \\ d_m \end{bmatrix},$$

$$A^i = \begin{bmatrix} a_j & c_j & d_j & & & & \\ b_{j+1} & a_{j+1} & c_{j+1} & d_{j+1} & & & \\ & \cdot & \cdot & \cdot & \cdot & & \\ & & \cdot & \cdot & \cdot & \cdot & \\ & & & \cdot & \cdot & \cdot & \cdot \\ & & & & b_{r-1} & a_{r-1} & c_{r-1} \\ & & & & & b_r & a_r \end{bmatrix}, \quad X^i = \begin{bmatrix} x_j \\ x_{j+1} \\ \cdot \\ \cdot \\ \cdot \\ x_{r-1} \\ x_r \end{bmatrix},$$

$$B^i = \begin{bmatrix} B_j \\ B_{j+1} \\ \cdot \\ \cdot \\ \cdot \\ B_{r-1} \\ B_r \end{bmatrix}, \quad q^i = \begin{bmatrix} b_j \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \\ 0 \end{bmatrix}, \quad r^i = \begin{bmatrix} 0 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ d_{r-1} \\ c_r \end{bmatrix}, \quad s^i = \begin{bmatrix} 0 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \\ d_r \end{bmatrix},$$

$$A^p = \begin{bmatrix} a_j & c_j & d_j & & & & \\ b_{j+1} & a_{j+1} & c_{j+1} & d_{j+1} & & & \\ & \cdot & \cdot & \cdot & \cdot & & \\ & & \cdot & \cdot & \cdot & \cdot & \\ & & & \cdot & \cdot & \cdot & \cdot \\ & & & & b_{N-1} & a_{N-1} & c_{N-1} \\ & & & & & b_N & a_N \end{bmatrix}, \quad X^p = \begin{bmatrix} x_j \\ x_{j+1} \\ \cdot \\ \cdot \\ \cdot \\ x_{N-1} \\ x_N \end{bmatrix},$$

$$B^p = \begin{bmatrix} B_j \\ B_{j+1} \\ \cdot \\ \cdot \\ \cdot \\ B_{N-1} \\ B_N \end{bmatrix}, \quad q^p = \begin{bmatrix} b_j \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \\ 0 \end{bmatrix}, \quad r^p = \begin{bmatrix} 0 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ d_{N-1} \\ c_N \end{bmatrix}, \quad s^p = \begin{bmatrix} 0 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \\ d_N \end{bmatrix}.$$

3.2. Implementations. Any of the systems given in (10) can be solved by solving one banded linear system with four different right hand sides. (Without loss of generality we take the ith subsystem.)

$$A^it^i = B^i, A^iv^i = q^i, A^iy^i = r^i, A^iz^i = s^i. \tag{11}$$

and the X^i becomes a linear combination of the four partial solutions

$$X^i = t^i - x_{j-1}v^i - x_{r+1}y^i - x_{r+2}z^i \tag{12}$$

Each processor can solve its system (11) independently without any communications with other processors. In this paper a modified version of Gaussian elimination for banded systems is used. Before finding the global solution, the parameters must be determined first. Taking the first two equations and the last equation from each subsystem, a new subsystem of order $3p \times 3p$ arises. The new subsystem has the following form:

$$\begin{bmatrix}
v_1 & y_1 & z_1 & & & & & 1 & 0 \\
v_2 & y_2 & z_2 & & & & & & 1 \\
v_m & y_m & z_m & 1 & & & & & \\
 & 1 & 0 & v_{m+1} & y_{m+1} & z_{m+1} & & & \\
 & & 1 & v_{m+2} & y_{m+2} & z_{m+2} & & & \\
 & & & \cdot & \cdot & \cdot & \cdot & \cdot & \\
 & & & & \cdot & \cdot & \cdot & \cdot & \\
 & & & & 1 & 0 & v_{(p-1)m+1} & y_{(p-1)m+1} & z_{(p-1)m+1} \\
 & & & & & 1 & v_{(p-1)m+2} & y_{(p-1)m+2} & z_{(p-1)m+2} \\
1 & & & & & & v_N & y_N & z_N
\end{bmatrix} \times$$

$$\begin{bmatrix} x_N \\ x_{m+1} \\ x_{m+2} \\ x_m \\ x_{2m+1} \\ \cdot \\ \cdot \\ \cdot \\ x_{(p-1)m} \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} t_1 \\ t_2 \\ t_m \\ t_{m+1} \\ t_{m+2} \\ \cdot \\ \cdot \\ \cdot \\ t_{(p-1)m+1} \\ t_{(p-1)m+2} \\ t_N \end{bmatrix} \qquad (13)$$

which can be solved efficiently by using a modified version of Gaussian elimination or the LU decomposition methods for banded systems provided the size of the system (i.e., the number of processors) is small. It is worth mentioning that the system of parameters (13) is not necessarily diagonally dominant even if the original system of equations to be solved is diagonally dominant. In this paper one of the processors is used to solve the system of parameters. It is to be noted that if the number of processors is large then solving the system of parameters efficiently will be worth looking at in the future.

4. Numerical Experiments. The proposed numerical scheme is implemented on an intel iPSC/2 hypercube with $\alpha \geq 5$. The system given in (9) is solved at each time step by the method described above. At each time step each processor will calculate its share of the right hand side of (9) with very little communications with its immediate neighbors in a ring topology. In practice it is found that this algorithm is efficient and is about four times faster when four processors are used than when one is used. According to employing this algorithm, with $\Delta x = 0.0391$ and $\Delta t = 0.00005$ the solution of the higher KdV equation (3) blows up at $t = 0.12045$. The main unresolved problem with this algorithm is to find an efficient parrallel algorithm for solving the system of parameters when the number of the utilized processors is large and a partial pivoting is needed.

REFERENCES

[1] V.E. Zakharov & V.S. Synakh, *The nature of the self-focusing singularity,* Sov. Phys. JETP., 41, (1976) pp. 465-468.

[2] M. Ablowitz & H. Segur, *Solitons and the inverse scattering transform,* SIAM, Philadelphia, 1981.

[3] J.L. Bona, P.E. Souganidis, & W.A. Strauss, *Stability and instability of solitary waves of Korteweg-de Vries type,* Proc. R. Soc. Lond. A 411, (1987), pp. 395-412.

[4] B. Fornberg & G. Whitham, *A numerical and theoretical study of certain nonlinear wave phenomena,* Phil. Trans. R. Soc. Lond. A 289, (1978), pp. 373-404.

[5] J.L. Bona, V.A. Dougalis, & O.A. Karakashian, *Fully discrete Galerkin methods for the Korteweg-de Vries equation,* Comput. & Math. with Appl. 12A, (1986), pp. 859-884.

[6] T.R. Taha & M. Ablowitz, *Analytical and numerical aspects of certain nonlinear evolution equations. III. Numerical, Korteweg-de Vries equation,* J. Comput. Phys. 55, 2, (1984), pp. 231-253.

[7] T.R. Taha & M.J. Ablowitz, *Analytical and numerical aspects of certain nonlinear evolution equations. IV. Numerical, modified Korteweg-de Vries equation,* J. Comput. Phys. 77, 2, (1988), pp. 540-548.

[8] T.R. Taha, *A parallel-vector algorithm for solving periodic tridiagonal linear systems of equations,* in the Proceedings of the Sixth Distributed Memory Computing Conference (Q. Stout and M. Wolfe, eds.), Portland, Oregon (1991), pp. 506-509.

[9] S.M. Müller, *A method to parallelize tridiagonal solvers,* in the Proceedings of the Fifth Distributed Memory Computing Conference (D. W. Walker and Q. F. Stout, eds.), Charleston, SC, Vol. 1, (1990), pp. 340-345.