Journal of Computational Physics 250 (2013) 332–346

Contents lists available at SciVerse ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

A homotopy method based on WENO schemes for solving steady state problems of hyperbolic conservation laws

Wenrui Hao [a], Jonathan D. Hauenstein [b], Chi-Wang Shu [c], Andrew J. Sommese [a], Zhiliang Xu [a], Yong-Tao Zhang [a,*]

[a] *Department of Applied and Computational Mathematics and Statistics, University of Notre Dame, Notre Dame, IN 46556, United States*
[b] *Department of Mathematics, North Carolina State University, Raleigh, NC 27695, United States*
[c] *Division of Applied Mathematics, Brown University, Providence, RI 02912, United States*

ARTICLE INFO

Article history:
Received 4 September 2012
Received in revised form 19 January 2013
Accepted 8 May 2013
Available online 27 May 2013

Keywords:
Homotopy continuation
Hyperbolic conservation laws
WENO scheme
Steady state problems

ABSTRACT

Homotopy continuation is an efficient tool for solving polynomial systems. Its efficiency relies on utilizing adaptive stepsize and adaptive precision path tracking, and endgames. In this article, we apply homotopy continuation to solve steady state problems of hyperbolic conservation laws. A third-order accurate finite difference weighted essentially non-oscillatory (WENO) scheme with Lax–Friedrichs flux splitting is utilized to derive the difference equation. This new approach is free of the CFL condition constraint. Extensive numerical examples in both scalar and system test problems in one and two dimensions demonstrate the efficiency and robustness of the new method.

1. Introduction

Numerical simulation of hyperbolic conservation laws has been a major research and application area of computational mathematics in the last few decades. Weighted essentially non-oscillatory (WENO) finite difference/volume schemes are a popular class of high order numerical methods for solving hyperbolic partial differential equations (PDEs). They have the advantage of attaining uniform high order accuracy in smooth regions of the solution while maintaining sharp and essentially monotone transitions of discontinuities. The first WENO scheme was constructed in [21] for a third-order accurate finite volume version. In [19], third- and fifth-order accurate finite difference WENO schemes in multi-space dimensions were constructed, with a general framework for the design of the smoothness indicators and nonlinear weights. Later, WENO schemes on unstructured meshes were developed for dealing with complex domain geometries [14,36]. For steady state problems of hyperbolic conservation laws, efficiently solving the large nonlinear system derived from the WENO discretization is still a challenging problem.

A popular approach for solving steady state problems is the time marching method. Starting from an initial condition, the numerical solution evolves into a steady state by using a time stepping scheme. For examples using this kind of approach, see the numerical experiments in [19] and high order residual distribution (RD) conservative finite difference WENO schemes for solving the steady state problems in [7]. The class of RD schemes for solving steady state problems were developed in, e.g.,

* Corresponding author. Tel.: +1 574 631 6079; fax: +1 574 631 4822.

E-mail addresses: whao@nd.edu (W. Hao), hauenstein@ncsu.edu (J.D. Hauenstein), shu@dam.brown.edu (C.-W. Shu), sommese@nd.edu (A.J. Sommese), zxu2@nd.edu (Z. Xu), yzhang10@nd.edu (Y.-T. Zhang).

URLs: http://www.nd.edu/~whao (W. Hao), http://www.math.ncsu.edu/~jdhauens (J.D. Hauenstein), http://www.dam.brown.edu/people/shu/ (C.-W. Shu), http://www.nd.edu/~sommese (A.J. Sommese), http://www.nd.edu/~yzhang10 (Y.-T. Zhang).

0021-9991/$ - see front matter © 2013 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.jcp.2013.05.008

[8,25,26,2,1]. They use a similar pointwise representation of the solution as in finite difference schemes, but allow conservative approximations with high order accuracy on very general meshes. In [7], a high order accurate residual by a WENO integration procedure was computed directly and the resulting RD-WENO schemes achieved both high order accuracy and conservativeness on arbitrary Cartesian or curvilinear meshes without any smoothness assumption.

A big advantage of the time marching method is that the computed steady state is stable and usually carries physical properties of the system and the initial condition. However, from the point of view of computational efficiency, the computational cost of time marching method for obtaining a steady state solution is *not* linear due to restricted time-step sizes by the well-known Courant–Friedrichs–Lewy (CFL) condition [10]. One way to improve this is to take advantage of the properties of hyperbolic PDEs, i.e., information propagates along characteristics from boundaries of a steady state problem. In [37,34] a class of iterative methods, called "fast sweeping methods" [38], have been combined with WENO schemes to accelerate the convergence of the time marching approach. Fast sweeping methods utilize alternating sweeping strategy to cover a family of characteristics in a certain direction simultaneously in each sweeping order. Coupled with the Gauss–Seidel iterations, the methods can achieve a fast convergence speed for computations of steady state solutions. Recently, this acceleration approach has been applied for the case of hyperbolic conservation laws in [6]. Another way to improve the convergence is to design new smoothness indicators. In [35], a new smoothness indicator is introduced to remove the slight post-shock oscillations and the numerical residue so that the WENO schemes can converge to machine precision.

It is desirable to design a numerical method for solving steady state problems with linear computational complexity. Such methods need to be free of the CFL conditions. For the large nonlinear system derived from the WENO discretization, one way to solve this system is to apply Newton iterations. A Newton iteration based method was adopted to solve the steady two dimensional Euler equations in [15,16,18]. The matrix-free Squared Preconditioning is applied to Newton iterations nonlinearly preconditioned by means of the flow solver in [17]. In this paper, we design a novel method to solve the large nonlinear system directly. This new method has linear computational complexity and is free of CFL conditions, as shown in our numerical experiments.

Discretizing many systems of nonlinear differential equations produces sparse polynomial systems. Numerical algorithms based on techniques arising in algebraic geometry, collectively called *numerical algebraic geometry*, have been developed to solve polynomial systems. Over the last decade, numerical algebraic geometry (see [20,29,32] for some background), which grew out of continuation methods for finding all isolated solutions of systems of nonlinear multivariate polynomials, has reached a high level of sophistication. Even though the polynomial systems that arise by discretizing differential equation system are many orders of magnitude larger than the polynomial systems that the algorithms of numerical algebraic geometry have been applied to, these algorithms can still be used efficiently to investigate such polynomial systems.

The major tool in numerical algebraic geometry is homotopy continuation. For a given system of polynomial equations to be solved, a homotopy between the given system and a new system (which is easier to solve and share many features with the former system) can be constructed (see Section 3 for a detailed description of this method in this context). Then, one tracks paths starting from each solution of the new system as one moves towards the original system along the homotopy, thereby obtaining solutions of the original system. The homotopy method computes all the complex (which obviously include real) solutions of a system which is known to have only isolated solutions. In this paper, we utilize a homotopy continuation approach to compute steady states of hyperbolic systems and demonstrate that this new approach can easily handle singular systems and also be used to find multiple steady states. The numerical experiments show that the homotopy method is competitive with the Newton based methods [15,16,18] and is faster than the classical time marching methods.

The organization of the article is as follows. We propose a homotopy method based on a third order finite difference WENO scheme in Section 2. In Section 3, we describe homotopy continuation and endgames. Extensive numerical simulation results are contained in Section 4 for one- and two-dimensional scalar and system steady state problems to demonstrate the behavior of our scheme. We conclude in Section 5.

2. Numerical methods

In this article, we solve both one-dimensional and two-dimensional steady state hyperbolic conservation laws. We use a third order accurate finite difference WENO scheme with Lax–Friedrichs flux splitting to discretize the PDEs. The advantage of using a finite difference WENO scheme is that we can perform the WENO reconstructions in a dimension-by-dimension manner, to achieve better efficiency than a finite volume WENO scheme in multi-dimensions [19]. For simplicity, we describe the scheme for solving one-dimensional problems. To solve multi-dimensional problems, one can simply use a dimension-by-dimension approach.

Consider the following one-dimensional hyperbolic conservation laws

$$u_t + (f(u))_x = g(u,x).$$

Setting u_t to zero, the steady state problem becomes

$$(f(u))_x - g(u,x) = 0.$$

For an initial condition u^0, we introduce the homotopy

$$H(u,\epsilon) = \big((f(u))_x - g(u,x) - \epsilon u_{xx}\big)(1-\epsilon) + \epsilon(u-u^0) \equiv 0, \tag{2.1}$$

where ϵ is a parameter between 0 and 1. In particular, when $\epsilon = 1$, the initial condition automatically satisfies (2.1) and, when $\epsilon = 0$, (2.1) becomes the steady state problem. The term ϵu_{xx} is introduced and guarantees that steady states obtained by homotopy function satisfy the entropy condition. It can also smooth the solution during the iterations and benefit the convergence of Newton's corrector. Moreover, the initial condition u^0 is also built in the homotopy function, and plays a role in computing steady state solution when ϵ is tracked.

To compute using (2.1), we discretize using the uniform grid $\{x_i\}_{i=0,\ldots,N}$ with corresponding grid function $\{u_i\}_{i=0,\ldots,N}$. The finite difference scheme with Lax–Friedrichs flux for solving (2.1) becomes

$$H(\mathbf{u},\epsilon) = \left(\frac{\widehat{f}_{i+\frac{1}{2}} - \widehat{f}_{i-\frac{1}{2}}}{h} - g(u_i, x_i) - \epsilon \frac{u_{i+1} + u_{i-1} - 2u_i}{h^2}\right)(1-\epsilon) + \epsilon(u_i - u_i^0) \equiv 0, \tag{2.2}$$

where $\mathbf{u} = (u_0, \ldots, u_N)^T$ and h is the uniform step size in the grid. Here, the derivative $f(u)_x$ at x_i is approximated by a conservative flux difference

$$f(u)_x|_{x=x_i} \approx \frac{1}{h}\left(\widehat{f}_{i+1/2} - \widehat{f}_{i-1/2}\right), \tag{2.3}$$

where, for the third order WENO scheme, the numerical flux $\widehat{f}_{i+1/2}$ depends on the three point values $f(u_l)$, $l = i-1, i, i+1$, when the wind is positive (i.e., when $f'(u) \geqslant 0$ for the scalar case, or when the corresponding eigenvalue is positive for the system case with a local characteristic decomposition). This numerical flux $\widehat{f}_{i+1/2}$ is written as a convex combination of two second order numerical fluxes based on two different substencils of two points each, and the combination coefficients depend on a "smoothness indicator" measuring the smoothness of the solution in each substencil. The detailed formula is

$$\widehat{f}_{i+1/2} = w_0\left[\frac{1}{2}f(u_i) + \frac{1}{2}f(u_{i+1})\right] + w_1\left[-\frac{1}{2}f(u_{i-1}) + \frac{3}{2}f(u_i)\right], \tag{2.4}$$

where

$$w_r = \frac{\alpha_r}{\alpha_1 + \alpha_2}, \quad \alpha_r = \frac{d_r}{(\widetilde{\epsilon} + \beta_r)^2}, \quad r = 0, 1. \tag{2.5}$$

The numbers $d_0 = 2/3$ and $d_1 = 1/3$ are called the "linear weights" while $\beta_0 = (f(u_{i+1}) - f(u_i))^2$ and $\beta_1 = (f(u_i) - f(u_{i-1}))^2$ are called the "smoothness indicators." The small positive number $\widetilde{\epsilon}$ is chosen to avoid the denominator to be 0. We take $\widetilde{\epsilon} = 10^{-6}$ in this article.

When the wind is negative (i.e., when $f'(u) < 0$), a right-biased stencil with numerical values $f(u_i)$, $f(u_{i+1})$, and $f(u_{i+2})$ are used to construct a third order WENO approximation to the numerical flux $\widehat{f}_{i+1/2}$. The formulae for negative and positive wind cases are symmetric with respect to the point $x_{i+1/2}$. For the general case of $f(u)$, we perform the "Lax–Friedrichs flux splitting"

$$f^+(u) = \frac{1}{2}(f(u) + \alpha u), \qquad f^-(u) = \frac{1}{2}(f(u) - \alpha u), \tag{2.6}$$

where $\alpha = \max_u |f'(u)|$. The positive wind part is $f^+(u)$ while $f^-(u)$ is the negative wind part. Corresponding WENO approximations are applied to find numerical fluxes $\widehat{f}^+_{i+1/2}$ and $\widehat{f}^-_{i+1/2}$, respectively. The final numerical flux $\widehat{f}_{i+1/2} = \widehat{f}^+_{i+1/2} + \widehat{f}^-_{i+1/2}$. See [19,27,28] for more details.

For the homotopy $H(\mathbf{u},\epsilon)$, the initial condition is continuously transformed into a steady state solution, as ϵ decreases from 1 to 0. In order to avoid singularities during the path tracking, we add a random complex number γ into the homotopy function, i.e.,

$$H(\mathbf{u},\epsilon) = \left(\frac{\widehat{f}_{i+\frac{1}{2}} - \widehat{f}_{i-\frac{1}{2}}}{h} - g(u_i, x_i) - \epsilon \frac{u_{i+1} + u_{i-1} - 2u_i}{h^2}\right)(1-\epsilon) + \gamma\epsilon(u_i - u_i^0) \equiv 0. \tag{2.7}$$

This remarkable technique of utilizing a randomly chosen complex number γ, called the γ-trick in the literature, makes sure that there are no singularities or bifurcations along the path. This γ-trick is an illustration of the use of so-called probability-one methods [29]. Although the γ-trick adds additional memory and computational cost via complex arithmetic, it is a small price to pay for avoiding singularities and bifurcations. "The homotopy method to compute steady state solutions is free of the CFL condition. The parameter ϵ only depends on the pre-set maximal step size and the local conditioning of the system. It does not depend on the wave speed of the hyperbolic system. Hence it does not need to be decreased along with the spatial mesh refinement. In the numerical experiments, we will show that our homotopy approach is free of the CFL condition and has linear computational complexity. This is a significant advantage leading to a much faster convergence of the homotopy method than the time marching method."

We summarize our homotopy continuation approach for computing steady state solutions in the following algorithm and expand upon the steps in the following section.

Algorithm 1. Homotopy continuation to compute steady state solutions

```
Input: The initial condition $\mathbf{u}^0$ as the solution of $H(\mathbf{u}, 1)$; the maximum step size during the path tracking; $\epsilon_{end}$: a
       number between 0 and 1 which indicates where to start the endgame algorithm.
Output: A steady state solution
Set $\epsilon = 1$.
while $\epsilon >= \epsilon_{end}$ do
    Set the stepsize $\Delta\epsilon$ by using adaptive stepsize control algorithm;
    Use predictor/corrector to compute corresponding solution at $\epsilon + \Delta\epsilon$.
end
Run the endgame algorithm to compute a solution to $H(\mathbf{u}, 0) = 0$.
```

3. Numerical homotopy tracking

In this section, we outline the numerical method for one of the most powerful tools in numerical algebraic geometry, the so-called homotopy continuation tracking. We give a brief explanation of the principles and algorithms involved as well as advertise some available software packages.

We consider a general homotopy $H(\mathbf{u}, t) = 0$, where $\mathbf{u}$ consists of the variables and $t \in [0, 1]$ is the path tracking parameter. When $t = 1$, we assume that we have known solutions to $H(\mathbf{u}, 1) = 0$. The known solutions are called start points and the system $H(\mathbf{u}, 1) = 0$ is called the start system. At $t = 0$, we recover the original system that we want to solve, called the target system. The problem of getting the solutions of the target system now reduces to tracking solutions of $H(\mathbf{u}, t) = 0$ from $t = 1$ where we know solutions to $t = 0$. The numerical method used in path tracking from $t = 1$ to $t = 0$ arises from solving the *Davidenko differential equation:*

$$\frac{dH(\mathbf{u}(t), t)}{dt} = \frac{\partial H(\mathbf{u}(t), t)}{\partial \mathbf{u}} \frac{d\mathbf{u}(t)}{dt} + \frac{\partial H(\mathbf{u}(t), t)}{\partial t} = 0.$$

In particular, path tracking reduces to solving initial value problems numerically with the start points being the initial conditions. Since we also have an equation which vanishes along the path, namely $H(\mathbf{u}, t) = 0$, predictor/corrector methods, such as Euler's predictor and Newton's corrector, are used in practice to solve these initial value problems. Additionally, the predictor/corrector methods are combined with adaptive stepsize and adaptive precision algorithms [4,5] to provide reliability and efficiency.

Even though high-order prediction methods are used in practice, we will focus on Euler's method for simplicity. Both prediction based on Euler's method and correction based on Newton's method arise from considering the following local model obtained via a Taylor expansion:

$$H(\mathbf{u} + \Delta\mathbf{u}, t + \Delta t) \approx H(\mathbf{u}, t) + \frac{\partial H}{\partial \mathbf{u}}(\mathbf{u}, t)\Delta\mathbf{u} + \frac{\partial H}{\partial t}(\mathbf{u}, t)\Delta t.$$

If we have a solution $(\mathbf{u}, t)$ on the path, that is, $H(\mathbf{u}, t) = 0$, one may predict to a new solution at $t + \Delta t$ by setting $H(\mathbf{u} + \Delta\mathbf{u}, t + \Delta t) = 0$ and solving the first-order terms to obtain Euler's method, namely

$$\Delta\mathbf{u} = -\left(\frac{\partial H}{\partial \mathbf{u}}(\mathbf{u}, t)\right)^{-1} \frac{\partial H}{\partial t}(\mathbf{u}, t)\Delta t. \tag{3.8}$$

On the other hand, if $H(\mathbf{u}, t)$ is not as small as one would like, one may hold t constant by setting $\Delta t = 0$ and solving the first-order terms to obtain Newton's method, namely

$$\Delta\mathbf{u} = -\left(\frac{\partial H}{\partial \mathbf{u}}(\mathbf{u}, t)\right)^{-1} H(\mathbf{u}, t). \tag{3.9}$$

The main concerns for implementing a numerical path tracking algorithm is to decide which predictor/corrector method to employ, the size of the step Δt, and the precision used to provide reliable computation. See [5,29] for more details regarding the construction and implementation of a path tracking algorithm.

The basic idea for a path tracking algorithm is as follows. If the initial prediction is not adequate, the corrector fails and the algorithm responds by shortening the stepsize to try again. For a small enough step and a high enough precision, the prediction/correction cycle must succeed and the tracker advances along the path. Moreover, for too large a stepsize, the predicted point can be far enough from the path that the rules set the precision too high that the algorithm fails before a decrease in stepsize is considered. So we employ adaptive path tracker [4,5] that adaptively changes the stepsize and precision simultaneously. This adaptive path tracker increases the security of adaptive precision path tracking while simultaneously reducing the computational cost.

We shall not discuss the actual path tracking algorithms further, but it is important to mention that these algorithms are designed to handle almost all apparent difficulties such as tracking to singular endpoints. When the endpoint of a solution path is singular, there are several approaches that can improve the accuracy of its estimate. All the singular endgames [22–24] are based on the fact that the homotopy continuation path $\mathbf{u}(t)$ approaching a solution of $H(\mathbf{u}, t) = 0$ as $t \to 0$ lies on a complex algebraic curve containing $(\mathbf{u}, 0)$. For a singular endpoint, Newton's method applied to solve $H(\mathbf{u}, 0)$ is no longer satisfactory since it loses its quadratic convergence or even diverges. The problem of slow convergence would be expected since the prediction along the incoming path may give a poor initial guess. Therefore, we need a different strategy to deal with singular solutions, namely *endgame algorithms*.

All singular endgames estimate the endpoint at $t = 0$ by building a local model of the path inside a small neighborhood containing $t = 0$. First, due to slowly approaching singular solutions, the endgames sample the path as close as possible to $t = 0$. The simplest endgame approach is to simply track the path as close to $t = 0$ as possible using extended precision to attempt to obtain the same accuracy as a nonsingular solution. The Cauchy integral endgame [22] is based on the use of the Cauchy Integral Theorem to estimate the solution of $H(\mathbf{u}, 0) = 0$. The Cauchy Integral Theorem states that

$$\mathbf{u}(0) = \frac{1}{2\pi c}\int_0^{2\pi c} u\left(Re^{\sqrt{-1}\theta}\right)d\theta,$$

where c is the winding number. Because of periodicity, the trapezoid method is an excellent scheme used to evaluate this integral which yields an estimate of $\mathbf{u}(0)$ with error of the same magnitude as the error with which we know the sample values $\mathbf{u}\left(Re^{\sqrt{-1}\theta}\right)$.

In summary, the numerical strategy of the Cauchy endgame is to first track $\mathbf{u}(t)$ until $t = R$ for some $R \in (0, 1)$. We then track $\mathbf{u}\left(Re^{\sqrt{-1}\theta}\right)$ as θ varies, to both determine the winding number c and to collect samples around this circular path. There are several good ways to determine c, with one obvious option being to directly measure the winding number by tracking a circular path, $t = Re^{\sqrt{-1}\theta}$ until the path closes up at $\theta = 2\pi c$ with c a positive number, namely, with $\mathbf{u}\left(Re^{2\pi c\sqrt{-1}}\right) = \mathbf{u}(R)$.

We refer to [22–24,29] for more on endgame methods such as the power-series method and the clustering or trace method. Many of these endgames are implemented in several sophisticated numerical packages well-equipped with path trackers such as Bertini [3], PHCpack [31], and HOMPACK [33]. Their binaries are all are available as freeware from their respective research groups.

4. Numerical results

In this section, we provide numerical experimental results to demonstrate the behavior of the homotopy method. In some examples, we compare this method with the time marching approach. The time marching approach uses the same third order finite difference WENO scheme with the second order TVD Runge–Kutta method (TVD-RK2) [27]. The stopping criteria is based on the numerical residual, namely $\|u_t\|_{L^1} \leqslant 10^{-15}$. All the examples were run on a Xeon 5410 processor running 64-bit Linux. All the numerical experiments show that winding number is less than 5, therefore the endgame algorithm is efficient to approach the steady state solution.

4.1. One-dimensional scalar problems

4.1.1. Example 1

Consider the steady state solutions of the Burgers equation with a source term

$$u_t + \left(\frac{u^2}{2}\right)_x = \sin(x)\cos(x), \quad x \in [0, \pi]$$

with initial condition $u(x, 0) = \beta\sin(x)$ and boundary condition $u(0, t) = u(\pi, t) = 0$. This problem was studied in [30] as an example of a problem with a unique steady state for a given initial condition. The steady state solution to this problem depends upon the value of β: a shock forms within the domain if $\beta \in [-1, 1]$; otherwise, the steady state solution is smooth. In particular,

$$u(x, \infty) = \begin{cases} \sin(x) & x < x_s, \\ -\sin(x) & x > x_s, \end{cases} \tag{4.10}$$

where x_s, the "shock" location, is $\pi - \sin^{-1}\left(\sqrt{1 - \beta^2}\right)$.

In order to test the order of accuracy to a smooth steady state solution, we take $\beta = 2$ yielding $u(x, \infty) = \sin x$. We use our homotopy method with the Lax-Friedrichs WENO3 fluxes, and present the numerical results in Table 1. The convergence to third order accuracy of L^1 and L^∞ error is clearly observed from this data. In Table 1, we compare the CPU time of our

Table 1
Errors and numerical orders of accuracy of WENO3 scheme for Example 4.1.1 with N points.

N	L^1 error	Order	L^∞ error	Order	Computing time	
					Homotopy (s)	TVD-RK2 (s)
20	3.68e−2	–	1.55e−2	–	0.21	0.18
40	7.49e−3	2.30	4.38e−2	1.83	0.47	0.69
80	1.21e−3	2.63	9.12e−3	2.26	1.02	2.52
160	1.71e−4	2.82	1.60e−3	2.51	1.98	9.85
320	2.18e−5	2.97	2.24e−4	2.84	4.03	39.10
640	2.76e−6	2.98	2.90e−5	2.95	8.48	154.93

homotopy method with the time marching approach using TVD-RK2. It is obvious that the homotopy method is much more efficient than the time marching approach. Furthermore, via a mesh refinement study, we observe that the CPU time increases linearly for the homotopy method. This shows that the computational cost is $O(N)$ (N is the number of spatial grid points) and the homotopy approach is free of the CFL condition.

Remark. The steady state system of this example is singular since there exists a sequence of steady state solutions depending upon the "shock" location x_s. Due to this singularity, Newton schemes have convergence difficulties. However, endgame algorithms can handle this singularity. A similar two-dimensional case presented in Example 4.3 is also singular.

4.1.2. Example 2

We consider the same problem as in Example 4.1.1, but take $\beta = 0.5$ in the initial condition. As mentioned in the previous example, a shock will form within the domain, which separates branches of the steady state. For this value of β, the shock is located at 2.0944. Fig. 1 displays the numerical solution for different values of ϵ. Additionally, we verify that the numerical shock is at the correct location and is resolved well for $\epsilon = 0$.

The convergence of the solutions with respect to ϵ for various β is plotted in Fig. 2. Here $\mathbf{u}(x,\epsilon)$ is the solution of homotopy function $H(\mathbf{u},\epsilon)$ in (2.2). In this case, a sequence $\mathbf{u}(x,\epsilon_n)$ converges to $\mathbf{u}(x,0)$. In Fig. 2, $\|\mathbf{u}(x,\epsilon) - \mathbf{u}(x,0)\|$ is the L^2 norm of the difference of $\mathbf{u}(x,\epsilon)$ and $\mathbf{u}(x,0)$. The step size of ϵ is determined by the adaptive path tracking method. In summary, this shows that the homotopy method converges to the steady states in roughly 10 to 20 steps. In Table 2, we can observe that the iteration number is approximately a constant when the spatial mesh is refined, for a fixed β value. Again, this shows that our homotopy approach is free of the CFL condition, and has linear computational complexity.

4.1.3. Example 3

We consider the steady state solutions of Burgers equation with a different source term, namely

$$u_t + \left(\frac{u^2}{2}\right)_x = -\pi\cos(\pi x)u, \quad x \in [0,1]$$

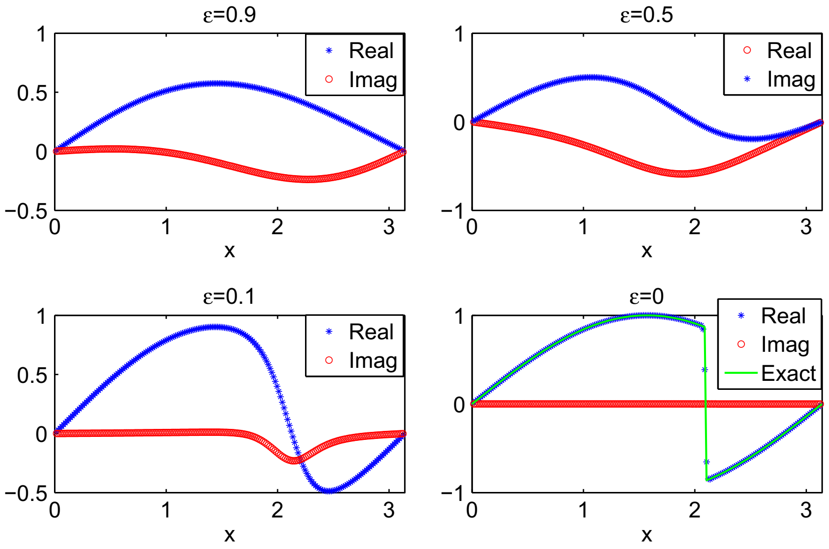

Fig. 1. The real part and imaginary part of numerical solution along path tracking from $\epsilon = 1$ to 0 with 200 grid points. For $\epsilon = 0$, the real part of numerical solution (stars) is compared with the exact solution (solid line), while the imaginary part goes to 0.

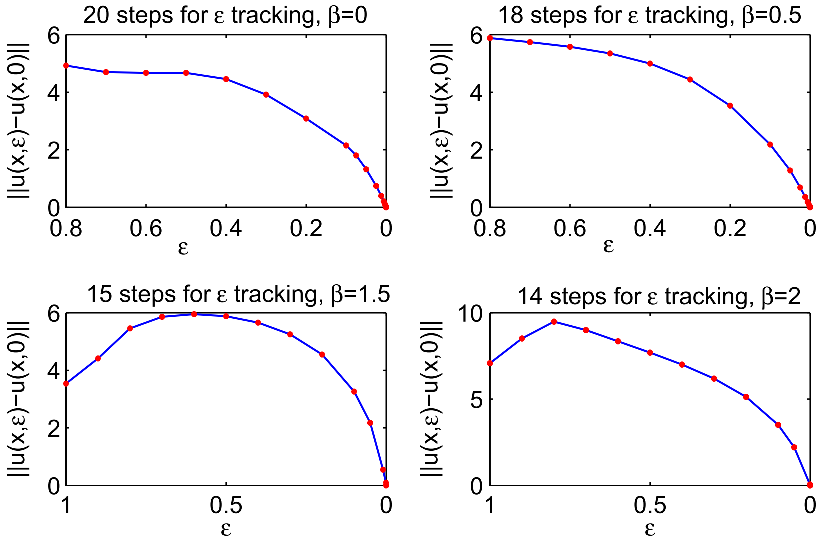

Fig. 2. The convergence of solutions with respect to ϵ for different β with 100 grid points. The maximum stepsize is 1/10.

Table 2
Iteration numbers for Example 4.1.1 with N points.

N	Iteration number			
	$\beta = 0$	$\beta = 0.5$	$\beta = 1.5$	$\beta = 2$
20	14	16	12	10
40	17	16	14	11
80	18	17	13	14
160	18	19	14	15
320	20	20	14	14
640	22	19	12	13

with boundary conditions $u(0,t) = 1$ and $u(1,t) = -0.1$, and initial condition

$$u(x,0) = \begin{cases} 1 & x < 0.5, \\ -0.1 & x \geqslant 0.5. \end{cases} \tag{4.11}$$

This problem has two steady state solutions with shocks, namely

$$u(x,\infty) = \begin{cases} 1 - \sin(\pi x) & x < x_s, \\ -0.1 - \sin(\pi x) & x \geqslant x_s, \end{cases} \tag{4.12}$$

where $x_s = 0.1486$ for one and $x_s = 0.8514$ for the other.

Both solutions satisfy the Rankine–Hugoniot jump condition and the entropy conditions, but only the one with the shock at 0.1486 is stable for small perturbations. This problem was studied in [9] as an example of multiple steady states for one-dimensional transonic flows. The classical method [7] shows that the numerical solution converges to the stable one when starting with a reasonable perturbation of the stable steady state.

However, with some minor modifications, our homotopy method can find all the steady state solutions when ϵ approaches zero. To accomplish this, we first compute all solutions of the the polynomial system (2.2) for $\epsilon = 0.1$ using bootstrapping method [11], which we shall summarize. The discretized polynomial system is completely solved on a coarse grid, say 10 grid points, first. Then, to obtain solutions for a finer grid, say 20 grid points, we first solve subsystems on each subdomain, whose length is 2 in this setup. The solutions are obtained by using homotopy continuation to build from the solutions of the subdomains. Solutions to even finer grids, say 320 grid points, can be obtained by iterating this bootstrapping approach. Table 3 shows the number of complex solutions at $\epsilon = 0.1$ and the real solutions produced at $\epsilon = 0$. This table clearly demonstrates that there are two steady states, which are displayed in Fig. 3.

Table 3
Number of solutions for Example 4.1.3.

# of grid points	# of complex solutions for $\epsilon = 0.1$	# of real solutions for $\epsilon = 0$
10	256	32
20	169	20
40	34	6
80	20	3
160	2	2
320	2	2

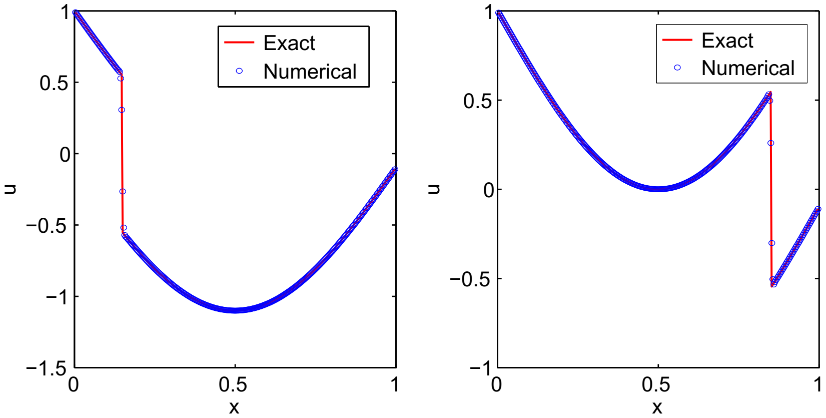

Fig. 3. Steady state solutions for Example 4.1.3: the one on the left is stable while the one on the right is unstable.

Remark. Our method can find multiple solutions of the steady state system. However, some steady state solutions might be unstable or not physical. Stabilities of these steady state solutions could be verified by perturbation theory and a time marching approach [12,13]. The computational cost of the verification is typically much smaller than a direct time marching method for computing steady states. This occurs since one only needs to add small random perturbations to the already obtained steady state solution and use it as the initial condition for time marching.

4.2. One-dimensional systems

4.2.1. Example 4

Consider the steady state solutions to the one-dimensional shallow water equation

$$\begin{pmatrix} h \\ hu \end{pmatrix}_t + \begin{pmatrix} hu \\ hu^2 + \frac{1}{2}gh^2 \end{pmatrix}_x = \begin{pmatrix} 0 \\ -ghb_x \end{pmatrix}, \tag{4.13}$$

where h denotes the water height, u is the velocity of the fluid, $b(x)$ represents the bottom topography, and g is the gravitational constant.

We consider the smooth bottom topography given by

$$b(x) = 5e^{-\frac{2}{5}(x-5)^2}, \quad x \in [0, 10].$$

Table 4
Errors and numerical orders of accuracy for the water height h using the homotopy method with WENO3 scheme for Example 4.2.1 with N points.

N	L^1 error	Order	L^∞ error	Order
20	2.23e−1	–	4.28e−1	–
40	4.42e−2	2.23	5.81e−2	2.88
80	6.18e−3	2.84	8.04e−3	2.85
160	8.16e−4	2.92	9.12e−4	3.14
320	1.05e−4	2.95	1.15e−4	2.99
640	1.29e−5	3.02	1.45e−5	2.98

The initial condition we consider is the stationary solution

$$h + b = 10, \quad hu = 0$$

with the exact steady state solution imposed by the boundary condition. By starting from a stationary initial condition, which itself is a steady state solution, we can check the order of accuracy. In particular, we tested our method using the third order WENO scheme with the numerical results displayed in Table 4. This clearly shows the third order of accuracy of both L^1 and L^∞ error. The convergence of the solutions is presented in Fig. 4.

4.2.2. Example 4

We next test our scheme on the steady state solution of the one-dimensional nozzle flow problem

$$\begin{pmatrix} \rho \\ \rho u \\ E \end{pmatrix}_t + \begin{pmatrix} \rho u \\ \rho u^2 + p \\ u(E+p) \end{pmatrix}_x = -\frac{a'(x)}{a(x)} \begin{pmatrix} \rho u \\ \rho^2 u \\ u(E+p) \end{pmatrix}, \tag{4.14}$$

where ρ denotes the density, u is the velocity of the fluid, E is the total energy, γ is the gas constant, which is taken as 1.4, $p = (\gamma - 1)(E - \frac{1}{2}\rho u^2)$ is the pressure, and $a(x)$ represents the area of the cross-section of the nozzle. We follow the setup of [7]: starting with an isentropic initial condition having a shock at $x = 0.5$. The mach number is linearly distributed (the inlet Mach number at $x = 0$ is 0.8. The outlet Mach number at $x = 1$ is 1.8) before and after the shock with the area of the cross-section, $a(x)$, determined by a function of mach number as follows

$$a(x)f(\text{Mach Number at } x) = \text{constant}, \quad \text{for all } x \in [0, 1],$$

and

$$f(w) = \frac{w}{(1 + \delta w^2)^d}, \quad \delta = \frac{1}{2}(\gamma - 1), \quad d = \frac{\gamma + 1}{2(\gamma - 1)}.$$

The density ρ and pressure p at $-\infty$ are 1.

In Fig. 5, the numerical solution computed by our homotopy method using the third order WENO scheme is compared with the exact solution. One can clearly see that the shock is resolved well. We also analyze the convergence speed by displaying the numerical solutions and the history of residues in Fig. 6. In particular, this shows that homotopy method approaches the exact solution in only 27 steps (6.12 s CPU time) while the time marching method using TVD-RK2 takes 52.19 s.

4.3. Two-dimensional scalar problem

Consider the steady state problem for the two-dimensional Burgers' equation with a source term

$$u_t + \left(\frac{1}{\sqrt{2}} \frac{u^2}{2}\right)_x + \left(\frac{1}{\sqrt{2}} \frac{u^2}{2}\right)_y = \sin\left(\frac{x+y}{\sqrt{2}}\right) \cos\left(\frac{x+y}{\sqrt{2}}\right), \quad (x, y) \in \left[0, \frac{\pi}{\sqrt{2}}\right] \times \left[0, \frac{\pi}{\sqrt{2}}\right]$$

with initial conditions

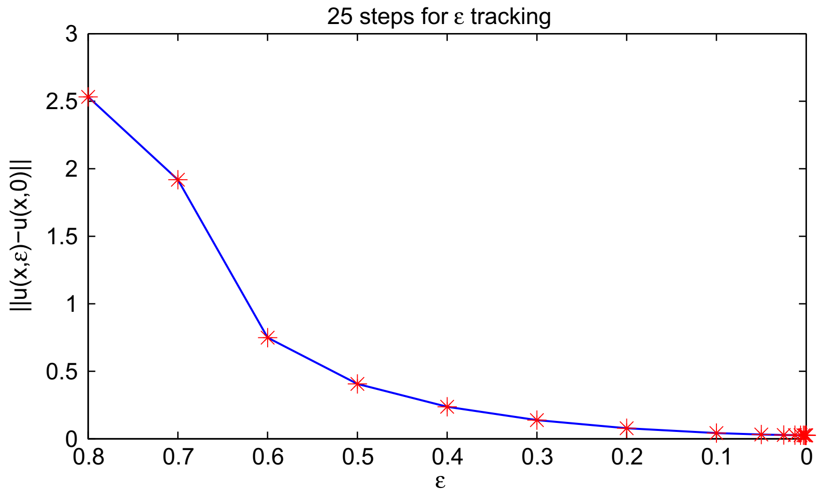

Fig. 4. The convergence of solutions with respect to ϵ with 100 grid points for Example 4.2.1. The maximum stepsize is 1/10.

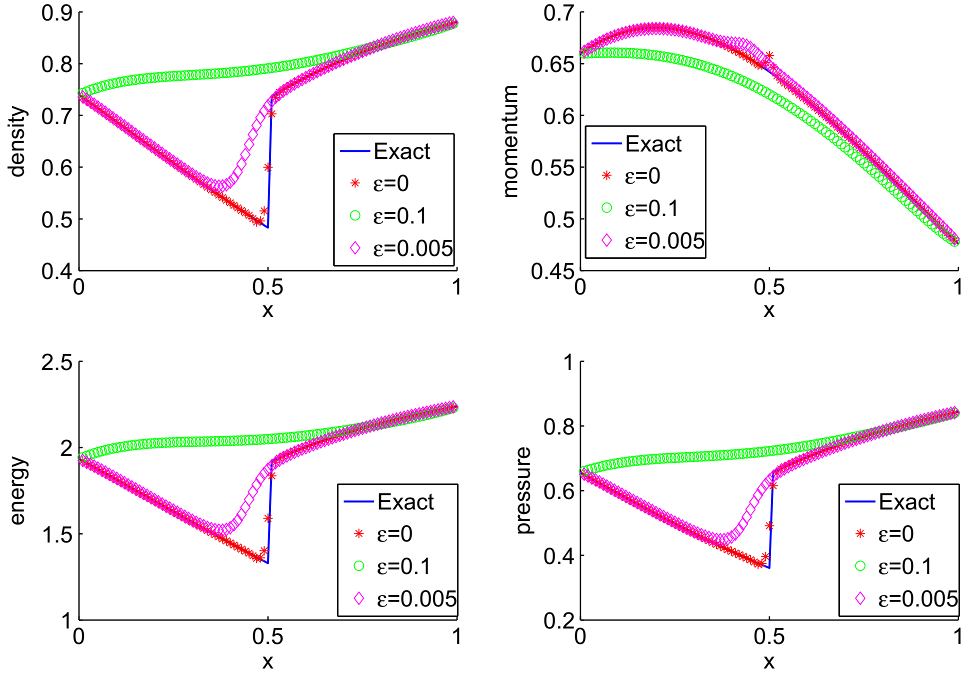

Fig. 5. Nozzle flow problem with 100 grid points. The numerical solutions correspond to $\epsilon = 0.1$, 0.005, and 0, respectively.

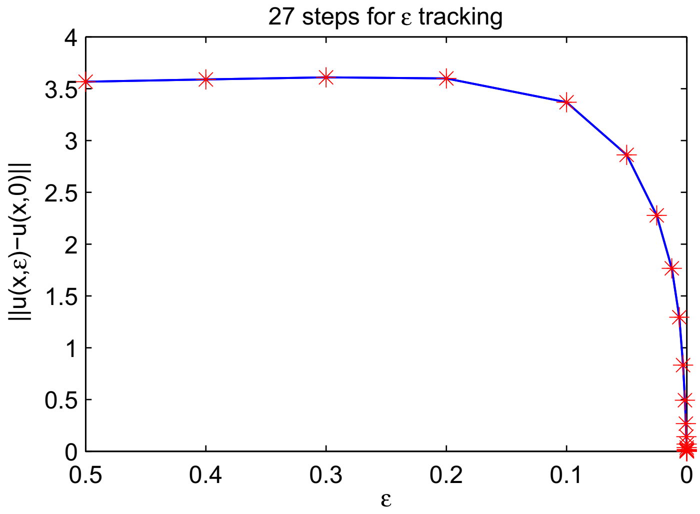

Fig. 6. Convergence of nozzle flow problem with 100 grid points. The maximum stepsize is 1/10.

Table 5
Errors and numerical orders of accuracy with WENO3 scheme for Example 4.3 with $N \times N$ points.

$N \times N$	L^1 error	Order	L^∞ error	Order	Computing time	
					Homotopy (s)	TVD-RK2 (s)
20×20	3.49e−3	–	8.69e−3	–	4.89	20.39
40×40	4.95e−4	2.31	1.32e−3	2.72	18.39	153.39
80×80	6.33e−5	2.97	1.74e−4	2.92	81.21	1127.23
160×160	7.62e−6	3.05	2.49e−5	2.80	335.75	8228.52

$$u(x,y,0)=\beta\sin\left(\frac{x+y}{\sqrt{2}}\right).$$

The boundary conditions are taken to satisfy the exact solution of the steady state problem. The one-dimensional problem in Example 4.1.1 arises along the northeast-southwest diagonal line. For this example we take $\beta=1.5$, which gives a smooth steady state solution $u(x,y,\infty)=\sin\left(\frac{x+y}{\sqrt{2}}\right)$. The numerical results shown in Table 5 clearly show that third order accuracy is achieved. As that for the 1D problem, we compare the CPU time of our homotopy method with the time marching approach using TVD-RK2. Again, we obtain the same conclusion. The homotopy method has linear computational complexity, and it is free of the CFL condition. Fig. 7 displays information regarding $\beta=2$ and $\beta=0.5$. In particular, this shows that the correct shock location is obtained in 14 steps for $\beta=0.5$.

4.4. Two-dimensional systems

4.4.1. Cauchy–Riemann problem

We consider the Cauchy–Riemann problem

$$\frac{\partial W}{\partial t}+A\frac{\partial W}{\partial x}+B\frac{\partial W}{\partial y}=0,\quad (x,y)\in[-2,2]\times[-2,2],\quad t>0, \tag{4.15}$$

where

$$A=\begin{pmatrix}1&0\\0&-1\end{pmatrix}\text{ and }B=\begin{pmatrix}0&1\\1&0\end{pmatrix}$$

with the following Riemann data $W=(u,v)^T$:

$$u=\begin{cases}1&\text{if } x>0 \text{ and } y>0,\\-1&\text{if } x<0 \text{ and } y>0,\\-1&\text{if } x>0 \text{ and } y<0,\\1&\text{if } x<0 \text{ and } y<0,\end{cases}\quad\text{and}\quad v=\begin{cases}1&\text{if } x>0 \text{ and } y>0,\\-1&\text{if } x<0 \text{ and } y>0,\\-1&\text{if } x>0 \text{ and } y<0,\\2&\text{if } x<0 \text{ and } y<0.\end{cases}$$

The solution is self-similar and therefore we can simplify the problem. For $W(x,y,t)=\widetilde{W}\left(\frac{x}{t},\frac{y}{t}\right)$, (4.15) can be rewritten as

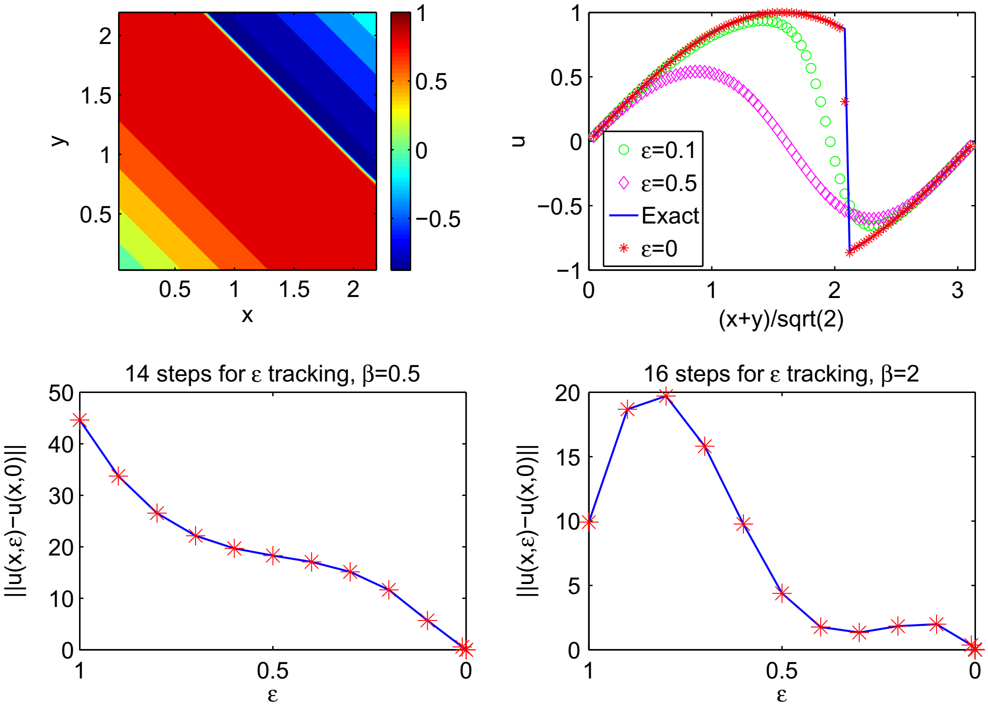

Fig. 7. Example 4.3 with 80 × 80 grid points. Top left: contour plot of solution for $\beta=0.5$; Top right: the numerical solutions with different ϵ versus the exact solution along the cross section through the northeast to southwest diagonal for $\beta=0.5$; Bottom: the convergence of solutions for $\beta=0.5$ and $\beta=2$, respectively.

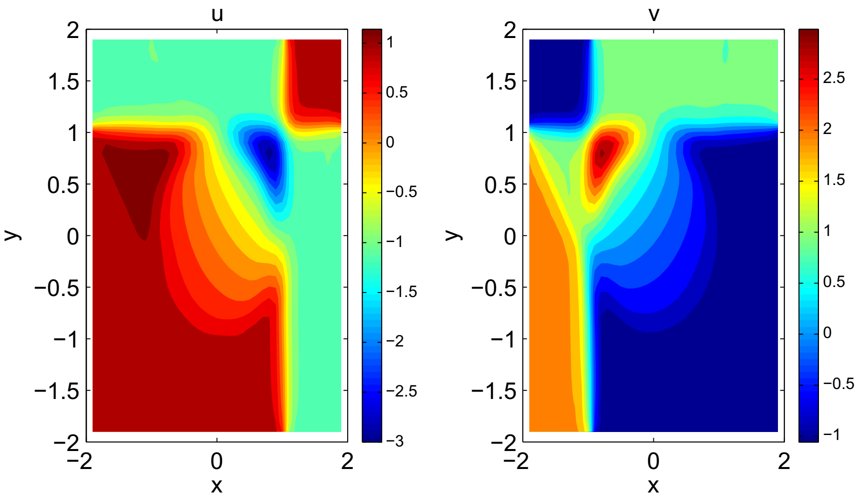

Fig. 8. Cauchy–Riemann problem with 50 × 50 grid points.

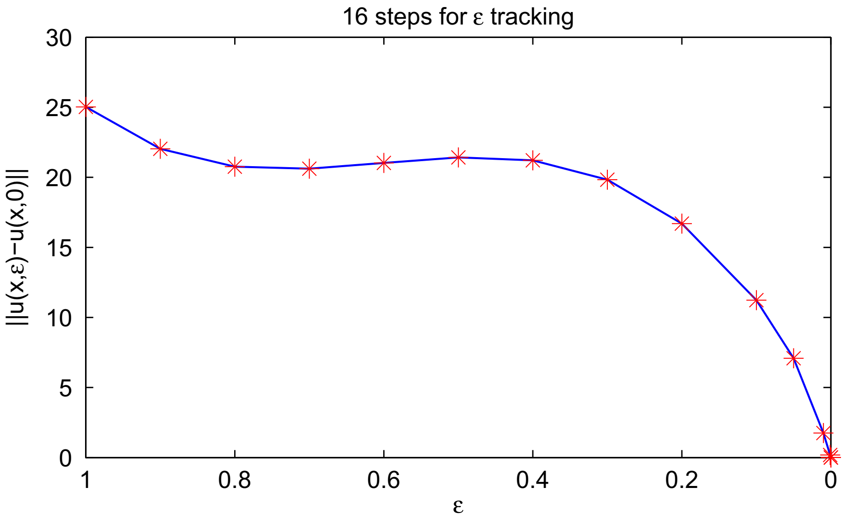

Fig. 9. Convergence of Example 4.4.1. The maximum stepsize is 1/10.

$$\frac{\partial}{\partial \xi}\left[(-\xi I + A)\widetilde{W}\right] + \frac{\partial}{\partial \eta}\left[(-\eta I + B)\widetilde{W}\right] = -2\widetilde{W}, \tag{4.16}$$

where $\xi = \frac{x}{t}$ and $\eta = \frac{y}{t}$. We consider the system (4.16) as a steady state system with time $t = 1$. The shock location of the Riemann data is propagated from the origin to $(1,1)$ and $(-1,1)$ for u and v, respectively. The boundary conditions are given by the Riemann data after the shift of the shock location. The numerical results are shown in Figs. 8 and 9.

4.4.2. Two-dimensional Euler equations

Our last example is to compute a steady state solution of a regular shock reflection problem for the two-dimensional Euler equations:

$$\mathbf{u}_t + (f(\mathbf{u}))_x + (g(\mathbf{u}))_y = 0, \quad (x,y) \in [0,4] \times [0,1], \tag{4.17}$$

where $\mathbf{u} = (\rho, \rho u, \rho v, E)^T, f(\mathbf{u}) = (\rho u, \rho u^2 + p, \rho u v, u(E+p))^T$, and $g(\mathbf{u}) = (\rho v, \rho u v, \rho v^2 + p, v(E+p))^T$. Here ρ is the density, (u, v) is the velocity, E is the total energy and $p = (\gamma - 1)(E - \frac{1}{2}(\rho u^2 + \rho v^2))$ is the pressure. The constant γ is the gas constant which is taken as 1.4 in our numerical tests.

The initial conditions are

$$(\rho, u, v, p) = (1.69997, 2.61934, -0.50632, 1.52819) \quad \text{on} \;\; y = 1,$$

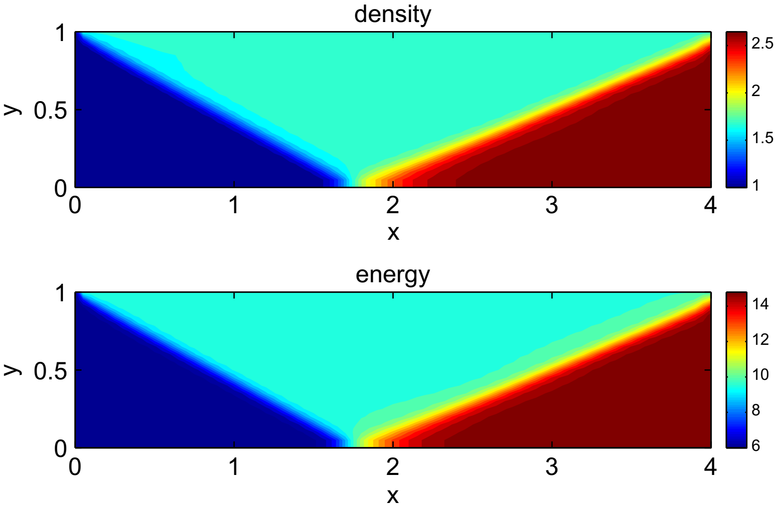

Fig. 10. Shock reflection for the density and the energy respectively with 100 × 25 grid points.

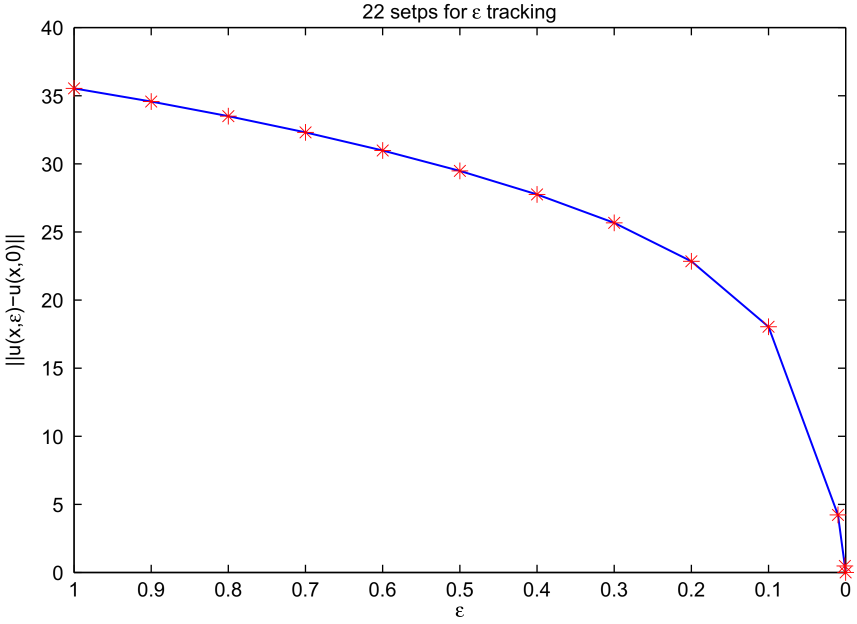

Fig. 11. Convergence of Example 4.4.2. The maximum stepsize is 1/10.

$$(\rho, u, v, p) = \left(1, 2.9, 0, \frac{1}{\gamma}\right) \text{ otherwise}$$

with boundary conditions

$$(\rho, u, v, p) = (1.69997, 2.61934, -0.50632, 1.52819) \quad \text{on } y = 1,$$

and reflective boundary condition on $y = 0$. The left boundary at $x = 0$ is set as an inflow with $(\rho, u, v, p) = \left(1, 2.9, 0, \frac{1}{\gamma}\right)$, and the right boundary at $x = 4$ is set to be an outflow with no boundary conditions prescribed. The numerical solutions obtained using the homotopy method with the WENO third order scheme are displayed in Fig. 10. It can be clearly seen that the incident and reflected shocks are well-resolved. Fig. 11 shows the convergence of the solution. It takes 22 steps (489.23 s) while using the Newton iteration based method [18] takes 46 steps (1012.37 s).

5. Conclusion

In this article, we have designed a homotopy approach based on WENO finite difference schemes for computing steady state solutions of conservation laws in one and two dimensional spaces. The homotopy continuation method is computationally less expensive and has the advantage that it can also be used to find multiple steady states. Moreover, this homotopy method is free of the CFL condition constraint. Using the above proposed algorithm as a beginning step, generalization of the technique to three-dimensional problems and utilizing discontinuous Galerkin (DG) methods are straightforward and will be carried out in the future.

Acknowledgements

Wenrui Hao was supported by the Duncan Chair of the University of Notre Dame. Jonathan D. Hauenstein was partially supported by NSF Grant DMS-1262428. The research of Chi-Wang Shu is supported by NSF Grant DMS-1112700 and ARO Grant W911NF-11-1-0091. Andrew J. Sommese was supported by the Duncan Chair of the University of Notre Dame. The research of Zhiliang Xu was partially supported by NSF Grants DMS-1115887, DMS-0800612 and NIH Grants 1 R01 GM100470-01 and 1 R01 GM095959-01A1.

References

[1] R. Abgrall, M. Mezine, Construction of second order accurate monotone and stable residual distribution schemes for steady problems, J. Comput. Phys. 195 (2004) 474–507.
[2] R. Abgrall, P.L. Roe, High order fluctuation scheme on triangular meshes, J. Scient. Comput. 19 (2003) 3–36.
[3] D.J. Bates, J.D. Hauenstein, A.J. Sommese, C.W. Wampler, Bertini: Software for Numerical Algebraic Geometry, Available at www.nd.edu/~sommese/bertini.
[4] D.J. Bates, J.D. Hauenstein, A.J. Sommese, C.W. Wampler, Adaptive multiprecision path tracking, SIAM J. Numer. Anal. 46 (2008) 722–746.
[5] D.J. Bates, J.D. Hauenstein, A.J. Sommese, C.W. Wampler, Stepsize control for adaptive multiprecision path tracking, Contemp. Math. 496 (2009) 21–31.
[6] W. Chen, C.-S. Chou, C.-Y. Kao, Lax-Friedrichs Fast Sweeping Methods for Steady State Problems for Hyperbolic Conservation Laws, J. Comput. Phys. 234 (2012) 452–471.
[7] C.-S. Chou, C.-W. Shu, High order residual distribution conservative finite difference weno schemes for steady state problems on non-smooth meshes, J. Comput. Phys. 214 (2006) 698–724.
[8] H. Deconinck, R. Struijs, G. Bourgeois, P. Roe, Compact advection schemes on unstructured meshes, Comput. Fluid Dyn., VKI Lecture Series 1993–04, (1993).
[9] P. Embid, J. Goodman, A. Majda, Multiple steady states for 1-D transonic flow, SIAM J. Scient. Stat. Comput. 5 (1984) 21–41.
[10] B. Gustafsson, H.-O. Kreiss, J. Oliger, Time Dependent Problems and Difference Methods, John Wiley & Sons, Inc., 1995.
[11] W. Hao, J.D. Hauenstein, B. Hu, A.J. Sommese, A domain decomposition algorithm for computing multiple steady states of differential equations, submitted (2011).
[12] W. Hao, J.D. Hauenstein, B. Hu, Y. Liu, A.J. Sommese, Y.-T. Zhang, Multiple stable steady states of a reaction-diffusion model on zebrafish dorsal-ventral patterning, Discr. Contin. Dyn. Syst. – Ser. S 4 (2011) 1413–1428.
[13] W. Hao, J.D. Hauenstein, B. Hu, Y. Liu, A.J. Sommese, Y.-T. Zhang, Continuation along bifurcation branches for a tumor model with a necrotic core, J. Scient. Comput. 53 (2012) 395–413.
[14] C. Hu, C.-W. Shu, Weighted essentially non-oscillatory schemes on triangular meshes, J. Comput. Phys. 150 (1999) 97–127.
[15] G.H. Hu, R. Li, T. Tang, A robust high-order residual distribution type scheme for steady Euler equations on unstructured grids, J. Comput. Phys. 229 (2010) 1681–1697.
[16] G.H. Hu, R. Li, T. Tang, A robust WENO type finite volume solver for steady Euler equations on unstructured grids, Commun. Comput. Phys. 9 (2011) 627–648.
[17] F. Iacono, G. May, Z.J. Wang, Relaxation techniques for high-order discretizations of steady compressible inviscid flows, in: 40th AIAA Fluid Dynamics Conference, Chicago, Illinois, AIAA paper number 2010-4991, (2010).
[18] R. Li, X. Wang, W.B. Zhao, A multigrid block lower-upper symmetric Gauss-Seidel algorithm for steady Euler equation on unstructured grids, Numer. Math. Theor. Meth. Appl. 1 (2008) 92–112.
[19] G.-S. Jiang, C.-W. Shu, Efficient implementation of weighted ENO schemes, J. Comput. Phys. 126 (1996) 202–228.
[20] T.Y. Li, Numerical Solution of Polynomial Systems by Homotopy Continuation Methods, in: F. Cucker (Ed.), Handbook of Numerical Analysis, Volume XI, Special Volume: Foundations of Computational Mathematics, North-Holland, 2003, pp. 209–304.
[21] X.-D. Liu, S. Osher, T. Chan, Weighted essentially non-oscillatory schemes, J. Comput. Phys. 115 (1994) 200–212.
[22] A.P. Morgan, A.J. Sommese, C.W. Wampler, Computing singular solutions to nonlinear analytic systems, Numer. Math. 58 (7) (1991) 669–684.
[23] A.P. Morgan, A.J. Sommese, C.W. Wampler, Computing singular solutions to polynomial systems, Adv. Appl. Math. 13 (3) (1992) 305–327.
[24] A.P. Morgan, A.J. Sommese, C.W. Wampler, A power series method for computing singular solutions to nonlinear analytic systems, Numer. Math. 63 (3) (1992) 391–409.
[25] P.L. Roe, D. Sidilkover, Optimum positive linear schemes for advection in two or three dimensions, SIAM J. Numer. Anal. 29 (1992) 1542–1588.
[26] R. Struijs, H. Deconinck, P.L. Roe, Fluctuation splitting schemes for the 2D Euler equations, Computational Fluid Dynamics, VKI Lecture Series, 1991–01, (1991).
[27] C.-W. Shu, Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws, in: B. Cockburn, C. Johnson, C.-W. Shu, E. Tadmor (Eds.), Advanced Numerical Approximation of Nonlinear Hyperbolic Equations, (Editor: A. Quarteroni), Lecture Notes in Mathematics, vol. 1697, Springer, 1998, pp. 325–432.
[28] C.-W. Shu, High order ENO and WENO schemes for computational fluid dynamics, in High-Order Methods for Computational Physics, T.J. Barth and H. Deconinck, editors, Lecture Notes in Computational Science and Engineering, Vol. 9, Springer, pp. 439–582, (1999).
[29] A.J. Sommese, C.W. Wampler, Numerical Solution of Systems of Polynomials Arising in Engineering and Science, World Scientific, Singapore, 2005.
[30] M.D. Salas, S. Abarbanel, D. Gottlieb, Multiple steady states for characteristic initial value problems, Appl. Numer. Math. 2 (1986) 193–210.
[31] J. Verschelde, Algorithm 795: PHCpack: A general-purpose solver for polynomial systems by homotopy continuation, ACM Trans. Math. Software 2 (1999) 251–276.
[32] C.W. Wampler, A.J. Sommese, Numerical Algebraic Geometry and Algebraic Kinematics, Acta Numer. 20 (2011) 469–567.
[33] L.T. Watson, M. Sosonkina, R.C. Melville, A.P. Morgan, H.F. Walker, Algorithm 777: HOMPACK90: A suite of Fortran 90 codes for globally convergent homotopy algorithms, ACM Trans. Math. Software 23 (1997) 514–549.
[34] T. Xiong, M. Zhang, Y.-T. Zhang, C.-W. Shu, Fifth order fast sweeping WENO scheme for static Hamilton-Jacobi equations with accurate boundary treatment, J. Scient. Comput. 45 (2010) 514–536.

[35] S. Zhang, C.-W. Shu, A new smoothness indicator for the WENO schemes and its effect on the convergence to steady state solutions, J. Scient. Comput. 31 (2007) 273–305.

[36] Y.-T. Zhang, C.-W. Shu, Third order WENO schemes on three dimensional tetrahedral meshes, Commun. Comput. Phys. 5 (2009) 836–848.

[37] Y.-T. Zhang, H.-K. Zhao, J. Qian, High order fast sweeping methods for static Hamilton-Jacobi equations, J. Scient. Comput. 29 (2006) 25–56.

[38] H.-K. Zhao, A fast sweeping method for Eikonal equations, Math. Comput. 74 (2005) 603–627.