

Research Article

Entropy Schemes for One-Dimensional Convection-Diffusion Equations

Rongsan Chen 

School of Mathematics and Physics, China University of Geosciences, Wuhan 430074, China

Correspondence should be addressed to Rongsan Chen; rchen@cug.edu.cn

Received 25 June 2020; Accepted 20 July 2020; Published 31 July 2020

Guest Editor: Karthikeyan Rajagopal

Copyright © 2020 Rongsan Chen. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, we extend the entropy scheme for hyperbolic conservation laws to one-dimensional convection-diffusion equation. The operator splitting method is used to solve the convection-diffusion equation that is divided into conservation and diffusion parts, in which the first-order accurate entropy scheme is applied to solve the conservation part and the second accurate central difference scheme is applied to solve the diffusion part. Numerical tests show that the L^∞ error achieves about second-order accuracy, but the L^1 error reaches about forth-order accuracy.

1. Introduction

In this paper, we consider the convection-diffusion equation:

$$u_t + f(u)_x = (A(u))_{xx} + c(x, t), \quad (1)$$

where $A'(u) \geq 0$. Many researchers have developed numerical methods for the convection-diffusion equation and have obtained some superconvergence results [1, 2].

In [3], Li has developed the entropy scheme which contains numerical solution and numerical entropy to compute the linear advection equation. The numerical tests showed that it can achieve very good accuracy and is suitable for long-time computation of smooth solutions. Yanfen and De-Kang investigated the truncation error for the entropy scheme and showed the entropy scheme has superconvergent property in [4]. However, when computing discontinuous solutions, spurious oscillations occurred in the vicinity of the discontinuities. In order to eliminate the spurious oscillations, an entropy-ultra-bee scheme was presented by Li and Mao for computing the linear advection equation. In essence, entropy-ultra-bee scheme is a combination of the entropy scheme and the ultra-bee scheme which can obtain good resolution in smooth regions and sharpen the discontinuity. In [5], Chen and Mao extended the entropy scheme to the nonlinear scalar conservation laws

and presented the entropy-TVD scheme. In [6], Cui and Mao extended the entropy scheme to the KdV equation. The scheme is second-order, but the numerical results showed that the scheme has a third-order convergence rate away from extrema. Furthermore, the scheme suits for long-time numerical computing. Chen et al. generalize the entropy-TVD scheme for the one-dimensional shallow water equations in [7]. The entropy scheme was extended to the Euler system in [8, 9].

The significance of the entropy scheme is in methodology. The original Godunov scheme is first-order accurate [10]. Traditional ways to extend it to high-order schemes are to use high-order interpolations in the solution reconstruction in each cell, assuming that the solution is smooth [11–19]. The schemes so developed are no more local as the original Godunov scheme. Different limiting technologies, such as TVD, ENO, and WENO, are then used to eliminate numerical oscillations caused by the presence of discontinuities. Different from the above approach, our scheme numerically computes more physical quantities, which are algebraically related with one another in each cell. The scheme then uses them to reconstruct the solution in the cell by enforcing the algebraic relations among them, with certain TVD limiting to maintain the stability. In doing so, the smooth assumption on the solution is not necessary.

With the solution reconstructed in this way, the numerical errors accumulate in the fashion that the local truncation errors in two successive time steps cancel each other, and this leads to the second-order accuracy of the scheme.

With the entropy scheme designed in this way, it maintains to be local as the original Godunov scheme. Since all the principle and augmented quantities have solid physical meanings and the reconstruction satisfies all the physical algebraic relations among them, the reconstructed solution in each cell physically well simulates the exact solution, even the latter is not smooth in the cell. Important physical properties such as the entropy condition and nonnegativity of mass and pressure are maintained in the scheme. Moreover, the numerical dissipations are quantitatively controlled in that they are used only near discontinuities and extremes of the solution.

In this paper, we mainly introduce the idea, and we choose a kind of convection-diffusion equation in one dimension with the convection part as $f(u)$ for simplicity. We do not consider the other kinds of convection-diffusion equation and high order in this paper. In order to extend to high order, we need to replace the step reconstruction with a higher order polynomial and solve generalized Riemann problems, and the algorithm may be very complicated. We use u^2 as the entropy function in this paper, but the scheme can also be executed in other entropy function, such as $u \log u$.

In this paper, we follow [6] and extend the entropy scheme to a kind of convection-diffusion equation in one dimension. The operator splitting method is used to solve the convection-diffusion equation that is divided into conservation and diffusion parts, in which the first-order accurate entropy scheme is applied to solve the conservation part and the second accurate central difference scheme is applied to solve the diffusion part. Numerical tests show that the L^∞ convergence rate approaches the second order and the L^1 convergence rate approaches the fourth order along with the mesh refinement.

The outline of the paper is as follows: in Section 2, we give a description for the scheme in detail; in Section 3, the numerical results that show the convergence rate are provided; and finally, Section 4 is the conclusion.

2. Description of the Scheme

We consider the following initial value problem for the convection-diffusion equation:

$$\begin{cases} u_t + f(u)_x = (A(u))_{xx}, & 0 \leq x \leq 2\pi, \\ u(x, 0) = u_0(x), \end{cases} \quad (2)$$

where $A'(u) \geq 0$. Suppose a pair of scalar function $(U(u), F(u))$ such that

$$U'(u)f'(u) = F'(u), \quad U''(u) \geq 0. \quad (3)$$

Multipled by $U'(u)$ in the two sides of equation (2), equation (2) becomes

$$U(u)_t + F(u)_x = U'(u)(A(u))_{xx}. \quad (4)$$

For simplicity, we use uniform cells with the cell size Δx , and we denote the cells centre by x_j and the cells by $(x_{j-1/2}, x_{j+1/2})$. Δt refers to time increment. We use u^n and U^n to represent a cell-average approximation to the true solution and a cell-average approximation to the entropy of the true solution, respectively. u^n and U^n are defined as

$$\begin{aligned} u_j^n &\approx \frac{1}{\Delta x} \int_{x_{j-1/2}}^{x_{j+1/2}} u(x, t_n) dx, \\ U_j^n &\approx \frac{1}{\Delta x} \int_{x_{j-1/2}}^{x_{j+1/2}} U(x, t_n) dx. \end{aligned} \quad (5)$$

In this way, the solution to equation (2) and its respective numerical solution are both made up of two entities.

2.1. Operator Splitting. As in [6], we use the operator splitting method to solve equation (2). At first, we divide equation (2) into two parts: the conservation part and the diffusion part. Then, we alternately solve the corresponding conservation part and diffusion part. The conservation and diffusion parts of equation (2) are defined, respectively, as

$$\begin{cases} u_t + f(u)_x = 0, \\ U(u)_t + F(u)_x = 0, \\ u(x, 0) = u_0(x), U(u(x, 0)) = U(u_0(x)), \end{cases} \quad (6)$$

$$\begin{cases} u_t = (A(u))_{xx}, \\ U(u)_t = U'(u)(A(u))_{xx}, \\ u(x, 0) = u_0(x), U(u(x, 0)) = U(u_0(x)). \end{cases} \quad (7)$$

2.2. Numerical Scheme for Equation (6). The entropy scheme with the half-step reconstruction is used to solve equation (6) (for details, refer to [5, 8]). The entropy scheme proceeds three steps as follows.

2.2.1. Step Reconstruction. A piecewise constant function with a half step is used to reconstruct the solution in each cell:

$$R(x; u^n, U^n) = u_j^n + \begin{cases} -d_j^n, & x_{j-1/2} < x \leq x_j, \\ +d_j^n, & x_j < x \leq x_{j+1/2}, \end{cases} \quad (8)$$

with d_j^n the half step (HS) of the reconstruction. The reconstruction (8) satisfies

$$\frac{1}{h} \int_{x_{j-1/2}}^{x_{j+1/2}} R(x; u^n, U^n) dx = u_j^n. \quad (9)$$

In order to compute the HS d_j^n , we require

$$\frac{1}{h} \int_{x_{j-1/2}}^{x_{j+1/2}} U(R(x; u^n, U^n)) dx = U_j^n, \quad (10)$$

i.e., the entropy cell-average of the reconstructed solution is equal to the numerical entropy in each cell. We can compute d_j^n from equation (10).

2.2.2. *Evolution.* Solve the initial value problem (IVP) as follows:

$$\begin{cases} v_t + f(v)_x = 0, & t_n < t \leq t_{n+1}, \\ v(x, t_n) = R(x; u^n, U^n). \end{cases} \quad (11)$$

For the linear equation, the exact solution to the problem is $R(x - t; u^n, U^n)$. For the nonlinear equation, the approximate solution to the problem can be reconstructed [18]. We denote the solution of (11) as $v(x, t)$.

2.2.3. *Cell Averaging.* Compute u_j^{n+1} and U_j^{n+1} as in the following:

$$\begin{aligned} u_j^{n+1} &= \frac{1}{h} \int_{x_{j-1/2}}^{x_{j+1/2}} v(x, t_{n+1}) dx, \\ U_j^{n+1} &= \frac{1}{h} \int_{x_{j-1/2}}^{x_{j+1/2}} U(v(x, t_{n+1})) dx. \end{aligned} \quad (12)$$

In practice, we compute u_j^{n+1} and U_j^{n+1} in the following flux forms:

$$u_j^{n+1} = u_j^n - \lambda(\tilde{f}_{j+1/2}^n - \tilde{f}_{j-1/2}^n), \quad (13)$$

$$U_j^{n+1} = \frac{1}{h} \int_{x_{j-1/2}}^{x_{j+1/2}} U(R(x; u^n, U^n)) dx - \lambda(\tilde{F}_{j+1/2}^n - \tilde{F}_{j-1/2}^n), \quad (14)$$

where the numerical flux $\tilde{f}_{j+1/2}^n = f(v(x_{j+1/2}, t))$ and $\tilde{F}_{j+1/2}^n = F(v(x_{j+1/2}, t))$.

2.3. *Numerical Scheme for Equation (7).* We use central difference to approximate the second derivatives and use the Euler forward time discretization for equation (7). The final scheme has the following form:

$$u_j^{n+1} = u_j^n + \frac{\Delta t}{\Delta x^2} (A(u_{j+1}^n) - 2A(u_j^n) + A(u_{j-1}^n)), \quad (15)$$

$$U_j^{n+1} = U_j^n + \frac{\Delta t}{\Delta x^2} U'(u_j^n) (A(u_{j+1}^n) - 2A(u_j^n) + A(u_{j-1}^n)). \quad (16)$$

We use the operator splitting method so that the initial problem (2) with initial data u^n and U^n is split into two subproblems. One proceeds as follows:

- Solve the conservation part of equation (2) with u^n and U^n to obtain a provisional solution u^* and U^* for the next time level.
- Solve the diffusion part of equation (2) by using u^* and U^* as initial condition.

This gives the final solution u^{n+1} and U^{n+1} for the new time level $n + 1$. From (13), (14), (15), and (16), we can obtain the final scheme in the following:

$$\begin{aligned} u_j^{n+1} &= u_j^n - \lambda(\tilde{f}_{j+1/2}^n - \tilde{f}_{j-1/2}^n) + \frac{\Delta t}{\Delta x^2} (A(u_{j+1}^n) \\ &\quad - 2A(u_j^n) + A(u_{j-1}^n)), \\ U_j^{n+1} &= \frac{1}{h} \int_{x_{j-1/2}}^{x_{j+1/2}} U(R(x; u^n, U^n)) dx - \lambda(\tilde{F}_{j+1/2}^n - \tilde{F}_{j-1/2}^n) \\ &\quad + \frac{\Delta t}{\Delta x^2} U'(u_j^n) (A(u_{j+1}^n) - 2A(u_j^n) + A(u_{j-1}^n)). \end{aligned} \quad (17)$$

Remark 1. The entropy scheme described in Section 2.2 for equation (6) is first-order accurate away from extrema [8], and the difference scheme in Section 2.3 for equation (7) is second-order away from extrema.

3. Numerical Experiments

In this section, we use the entropy scheme to compute one-dimensional convection-diffusion equation. In the following, two examples come from [1] and the CFL number is taken to be 0.2.

Example 1. Consider the following initial value problem:

$$\begin{cases} u_t + u_x = \varepsilon u_{xx}, \\ (u^2)_t + (u^2)_x = 2uu_{xx}, \\ u(x, 0) = \sin(x), \\ u(0, t) = u(2\pi, t). \end{cases} \quad (18)$$

The exact solution to this problem is

$$u(x, t) = e^{-\varepsilon t} \sin(x - t). \quad (19)$$

We take ε as 1 and 0.01, respectively. We conduct the computation with 20, 40, 80, 160, 320, and 640 cells, respectively; the computational time is $t = 1.0$; and we present the L^1 and L^∞ errors and orders of convergence of $\varepsilon = 1$ in Table 1. We observe that the L^∞ orders of convergence approach the second order and the L^1 orders of convergence approach the fourth order along with the mesh refinement. We present the L^1 and L^∞ errors and orders of convergence of $\varepsilon = 0.01$ in Table 2. We can see from Table 2 that the orders of convergence of L^∞ error are greater than 2, and the orders of convergence of L^1 error are greater than 4.

Example 2. Consider the following initial value problem.

$$\begin{cases} u_t + (u^2)_x = u_{xx} + c(x, t), \\ (u^2)_t + \left(\frac{4}{3}u^3\right)_x = 2uu_{xx} + 2uc(x, t), \\ u(x, 0) = \sin(x), \\ u(0, t) = u(2\pi, t), \end{cases} \quad (20)$$

TABLE 1: Example 1, $\varepsilon = 1$, numerical errors and orders of convergence at $t = 1$.

Cells	L^∞ error	Order	L^1 error	Order
10	$1.611E-002$	—	$8.543E-004$	—
20	$6.794E-003$	1.245	$1.244E-004$	2.779
40	$2.249E-003$	1.594	$1.250E-005$	3.315
80	$5.346E-004$	2.072	$7.563E-007$	4.047
160	$1.388E-004$	1.945	$5.398E-008$	3.808
320	$4.017E-005$	1.788	$4.052E-009$	3.735
640	$9.336E-006$	2.105	$2.113E-010$	4.261

TABLE 2: Example 1, $\varepsilon = 0.01$, numerical errors and orders of convergence at $t = 1$.

Cells	L^∞ error	Order	L^1 error	Order
10	$6.221E-002$	—	$6.749E-003$	—
20	$2.727E-002$	1.189	$1.237E-003$	2.447
40	$1.081E-002$	1.334	$1.563E-004$	2.984
80	$6.226E-003$	0.796	$1.602E-005$	3.286
160	$1.509E-003$	2.044	$7.360E-007$	4.444
320	$2.582E-004$	2.547	$2.784E-008$	4.724
640	$3.933E-005$	2.714	$1.332E-009$	4.385

TABLE 3: Example 2, numerical errors and orders of convergence at $t = 1$.

Cells	L^∞ error	Order	L^1 error	Order
10	$3.168E-002$	—	$1.629E-003$	—
20	$7.198E-003$	2.137	$1.045E-004$	3.962
40	$2.302E-003$	1.644	$8.759E-006$	3.577
80	$5.877E-004$	1.970	$5.960E-007$	3.877
160	$1.561E-004$	1.912	$3.916E-008$	3.927
320	$4.321E-005$	1.853	$2.640E-009$	3.890
640	$1.222E-005$	1.822	$2.046E-010$	3.689

where $c(x, t) = -e^{-2t} \cos(x - t)(e^t - 2 \sin(x - t))$. The exact solution to this problem is

$$u(x, t) = e^{-t} \sin(x - t). \quad (21)$$

Equation (20) is a convection-diffusion equation with a nonlinear convective term. We conduct the computation with 20, 40, 80, 160, 320, and 640 cells, respectively; the computational time is $t = 1.0$; and we present the L^1 and L^∞ errors and orders of convergence in Table 3. We can see from Table 3 that the L^∞ orders of convergence approach the second order and the L^1 orders of convergence approach the fourth order along with the mesh refinement.

4. Conclusions

In this paper, the entropy scheme is extended to one-dimensional of convection-diffusion equation. We divide the convection-diffusion equation into two parts and use the operator splitting method to solve it. The first-order accurate entropy scheme is applied to solve the conservation part, and the second accurate central difference scheme is applied to solve the diffusion part. We have presented two numerical examples, and the numerical results show that the L^∞ orders of convergence approach the second order and the L^1 orders

of convergence approach the fourth order along with the mesh refinement. As for other kinds of convection-diffusion equation, only minor modifications need to be made to the algorithm. The extension to two dimensions is our future work.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The author declares that there are no conflicts of interest.

Acknowledgments

The research was supported by the National Natural Science Foundation of China (no. 11201436) and Fundamental Research Funds for the Central Universities, China University of Geosciences (Wuhan).

References

- [1] Y. Cheng and C.-W. Shu, "Superconvergence of local discontinuous galerkin methods for one-dimensional convection-diffusion equations," *Computers & Structures*, vol. 87, no. 11-12, pp. 630–641, 2009.
- [2] X. Zhang, Y. Liu, and C.-W. Shu, "Maximum-principle-satisfying high order finite volume weighted essentially non-oscillatory schemes for convection-diffusion equations," *SIAM Journal on Scientific Computing*, vol. 34, no. 2, pp. A627–A658, 2012.
- [3] H. Li, "Second-order entropy dissipation scheme for scalar conservation laws in one space dimension," Master's thesis, Shanghai University, Shanghai, China, 2002.
- [4] C. Yanfen and M. De-Kang, "Error self-canceling of a difference scheme maintaining two conservation laws for linear advection equation," *Mathematics of Computation*, vol. 81, no. 278, pp. 715–741, 2011.
- [5] R. Chen and D.-K. Mao, "Entropy-TVD scheme for nonlinear scalar conservation laws," *Journal of Scientific Computing*, vol. 47, no. 2, pp. 150–169, 2011.
- [6] Y. Cui and D.-K. Mao, "Numerical method satisfying the first two conservation laws for the Korteweg-de Vries equation," *Journal of Computational Physics*, vol. 227, no. 1, pp. 376–399, 2007.
- [7] R. Chen, M. Zou, and L. Xiao, "Entropy-TVD scheme for the shallow water equations in one dimension," *Journal of Scientific Computing*, vol. 71, no. 2, pp. 822–838, 2017.
- [8] H. Li, Z. Wang, and D.-K. Mao, "Numerically neither dissipative nor compressive scheme for linear advection equation and its application to the Euler system," *Journal of Scientific Computing*, vol. 36, no. 3, pp. 285–331, 2008.
- [9] R. Chen and D. Mao, "Improved entropy-ultra-bee scheme for the Euler system of gas dynamics," *Journal of Computational Mathematics*, vol. 35, no. 2, pp. 213–243, 2017.
- [10] S. K. Godunov, "A finite difference method for computation of discontinuous solutions of the equations of fluid dynamics," *Matematicheskii Sbornik*, vol. 47, pp. 357–393, 1959.
- [11] P. Colella, "A direct Eulerian MUSCL scheme for gas dynamics," *SIAM Journal on Scientific and Statistical Computing*, vol. 6, no. 1, pp. 104–117, 1985.

- [12] P. Colella and P. Woodward, "The numerical simulation of two-dimensional fluid flow with strong shocks," *Journal of Computational Physics*, vol. 54, no. 1, pp. 115–173, 1984.
- [13] P. Colella and P. R. Woodward, "The piecewise parabolic method (PPM) for gas-dynamical simulations," *Journal of Computational Physics*, vol. 54, no. 1, pp. 174–201, 1984.
- [14] A. Harten, S. Osher, B. Engquist, and S. R. Chakravarthy, "Some results on uniformly high-order accurate essentially nonoscillatory schemes," *Applied Numerical Mathematics*, vol. 2, no. 3–5, pp. 347–377, 1986.
- [15] A. Harten, B. Engquist, S. Osher, and S. R. Chakravarthy, "Uniformly high order accurate essentially non-oscillatory schemes, III," *Journal of Computational Physics*, vol. 71, no. 2, pp. 231–303, 1987.
- [16] A. Harten and S. Osher, "Uniformly high-order accurate nonoscillatory schemes. I," *SIAM Journal on Numerical Analysis*, vol. 24, no. 2, pp. 279–309, 1987.
- [17] G.-S. Jiang and C.-W. Shu, "Efficient implementation of weighted ENO schemes," *Journal of Computational Physics*, vol. 126, no. 1, pp. 202–228, 1996.
- [18] R. J. LeVeque, *Finite Volume Methods for Hyperbolic Problems*, Cambridge University Press, Cambridge, UK, 2002.
- [19] C.-W. Shu and S. Osher, "Efficient implementation of essentially non-oscillatory shock-capturing schemes, II," *Journal of Computational Physics*, vol. 83, no. 1, pp. 32–78, 1989.