

# Efficient Algorithms for Solving Partial Differential Equations with Discontinuous Solutions

*Chi-Wang Shu*

**P**artial differential equations (PDEs) arise in numerous scientific and engineering applications. Mathematical modeling in various applications often ends up with a set of PDEs. Mathematical techniques can help to understand many crucial properties of the PDEs, such as existence and uniqueness of solutions under suitable initial and/or boundary conditions, and the well-posedness of the problem, which refers to the fact that the change of the solution for later time is controlled by the change of the initial condition or the change of the solution inside the domain is controlled by the change of the boundary data. However, for the vast majority of PDEs from applications, it is not possible to obtain explicit formulas for their solutions. Thus, in scientific and engineering applications, it is necessary to solve the PDEs numerically, that is, using certain algorithms to obtain approximate solutions to the PDEs on a computer.

## **Hyperbolic PDEs and Discontinuous Solutions**

PDEs are categorized into different types, including elliptic, parabolic, and hyperbolic PDEs. In this article we concentrate our discussion on hyperbolic PDEs. These PDEs describe various wave propagation phenomena, such as water waves, electromagnetic waves, and waves in gas dynamics.

---

*Chi-Wang Shu is the Theodore B. Stowell University Professor of Applied Mathematics at Brown University. His email address is [shu@dam.brown.edu](mailto:shu@dam.brown.edu).*

DOI: <http://dx.doi.org/10.1090/noti837>

The simplest hyperbolic PDE is the equation of linear convection

$$(1) \quad u_t + au_x = 0$$

where  $u = u(x, t)$  is the solution which depends on the spatial location  $x$  and time  $t$ ,  $a$  is a constant which is the speed of wave propagation, and subscripts refer to partial derivatives, e.g.,  $u_t = \frac{\partial u}{\partial t}$ . If we provide an initial condition

$$(2) \quad u(x, 0) = g(x),$$

then it is easy to verify that the unique solution of the PDE (1) is

$$(3) \quad u(x, t) = g(x - at).$$

That is, the solution is simply a shift of the initial condition with speed  $a$ . We can imagine that this is the kind of waves we may observe on a lake with a mild wind. For this problem, we do have the analytical formula for the solution in (3); hence there is no need to resort to numerical algorithms. However, such simple examples with known exact solutions are often used to verify the performance of different algorithms since the numerical error can be computed exactly. The exact solution (3) implies that the solution  $u$  remains smooth for later time  $t$  if it is smooth initially. This is a common feature of linear hyperbolic PDEs: discontinuities in the solution  $u$  can only arise from the imposed initial and/or boundary conditions.

If we change the PDE “slightly” to the Burgers’ equation

$$(4) \quad u_t + uu_x = 0,$$

or more appropriately written in conservation form

$$(5) \quad u_t + \left( \frac{u^2}{2} \right)_x = 0,$$

which is a simple model of fluid flow, the situation changes dramatically. With the same initial condition (2), we can no longer easily write down the solution for later time analytically. Moreover, even if we take a very smooth initial condition  $g(x) = \sin(x)$ , the solution evolves into a sharp front and finally, at  $t = 1$ , the front becomes vertical and the solution becomes discontinuous; see e.g., [17]. The solution for later time must be understood in the weak sense [17] since it no longer makes sense to require the PDE (5) to be literally satisfied at the point of discontinuity. For nonlinear hyperbolic equations, discontinuous solutions are generic: regardless of the smoothness of the given initial and/or boundary conditions, the solution to the PDE may become discontinuous.

The main theme of this article is the discussion of efficient numerical algorithms to solve hyperbolic PDEs when the solution becomes discontinuous.

### Numerical Schemes for Smooth Solutions

Before discussing discontinuous solutions, we first describe the basic concepts of numerical schemes for smooth solutions. There are many types of numerical schemes in applications, such as finite difference schemes, finite element schemes, and spectral methods; see e.g., [8], [2], and [10]. For the simple PDE (1) or (5), assuming that we would like to solve it over the interval  $x \in [0, 1]$  with a periodic boundary condition, the finite difference scheme would start with a choice of grid points  $0 = x_1 < x_2 < \dots < x_N = 1$ , which are assumed uniform with  $h = x_{j+1} - x_j = \frac{1}{N}$  for simplicity, and a time discretization  $0 = t^0 < t^1 < t^2 < \dots$ . We can then write down the scheme satisfied by the numerical solution  $u_j^n$ , which approximates the solution  $u$  of the PDE at the location  $(x_j, t^n)$ . The finite element and spectral method would start with an  $N$ -dimensional function space  $V_N = \text{span}\{\varphi_1(x), \varphi_2(x), \dots, \varphi_N(x)\}$ . We can then seek the numerical solution  $u_N(x, t)$  approximating  $u(x, t)$  from the space  $V_N$ . That is,

$$u_N(x, t) = \sum_{j=1}^N a_j(t) \varphi_j(x),$$

and the scheme solves for the coefficients  $a_j(t)$  at the time levels  $t^n$  for  $n = 0, 1, 2, \dots$ . The difference between the finite element method and the spectral method is that  $V_N$  consists of piecewise polynomial functions for the former and of global polynomials or trigonometric polynomials for the latter.

We can see that for all these methods, the computer stores and processes  $N$  pieces of information for each time step ( $u_j^n$  for  $1 \leq j \leq N$  for the finite difference method and  $a_j(t^n)$  for  $1 \leq j \leq N$  for the finite element and spectral methods). The processing cost (the number of operations such as additions and multiplications) per time step is different for different methods, but for finite difference and finite element methods it is typically  $O(N)$  and for the spectral method using fast Fourier transforms (FFT) it is  $O(N \log N)$ . Since for most practical computations  $N$  is not that large,  $\log N$  would be a very modest number compared with  $N$ ; hence all three methods have costs roughly of the same order with respect to  $N$  per time step.

The purpose of *numerical analysis* is to analyze the stability and errors of these schemes when  $N$  changes (increases). One measurement of computational efficiency for a numerical algorithm is the order of accuracy. For the three types of methods described above, if the error  $e_h$ , which for a finite difference scheme is  $(e_h)_j^n = u(x_j, t^n) - u_j^n$  and for a finite element or spectral method is  $e_h(x, t^n) = u(x, t^n) - u_N(x, t^n)$ , measured by a suitable norm such as the  $L^p$  norm with  $1 \leq p \leq \infty$ , is of the order  $O(N^{-k})$ , then  $k$  is referred to as the order of the scheme. If  $k$  is large, the scheme is called a “high-order” scheme (usually  $k$  would need to be at least 3 in order for the method to be referred to as a high-order scheme). When the number  $N$  is doubled, the error of a first-order method would only be reduced by a factor of 2, but the error of a third-order method would be reduced by a factor of  $2^3 = 8$ . Since the cost per time step is always  $O(N)$  (of course for higher-order methods the coefficient in the  $O(N)$  term is larger, but this coefficient relative to  $N$  does not change), apparently a high-order method is more efficient when  $N$  is large.

It is relatively easy to design algorithms which are high-order accurate when the solution of the PDE is smooth. For example, for the finite difference scheme, a two-point finite difference

$$(6) \quad \frac{u_{j+1} - u_j}{h} = u_x(x_j) + O(h)$$

would be first-order accurate, while a three-point finite difference

$$(7) \quad \frac{u_{j+1} - u_{j-1}}{2h} = u_x(x_j) + O(h^2)$$

would be second-order accurate (we have dropped the superscript  $n$  when considering spatial accuracy). Here we are measuring the errors locally, namely by putting the exact smooth solution  $u$  of the PDE into the finite difference scheme and measuring its remainder, which is referred to as the *local truncation error*. We would like to emphasize that for a  $k$ th-order local truncation error analysis

such as (6) or (7) to hold, the solution  $u$  should have  $k + 1$  continuous derivatives since we are using Taylor expansions to obtain the order of accuracy. The Lax equivalence theorem and its nonlinear generalization by Strang (see e.g., [8]) provide us with assurance that for a stable scheme, such easily measured local truncation error and the global error  $e_h$  defined above are of the same order with respect to  $N$ .

Comparing with finite difference and finite element methods which are usually of a fixed order of accuracy, spectral methods have errors which are smaller than  $O(N^{-k})$  for *any* fixed  $k$ ; thus they are “infinite-order” schemes. In fact, the error for spectral methods approximating analytic solutions of PDEs can be exponentially small; namely, the error decays as  $O(e^{-\alpha N})$  for some constant  $\alpha > 0$ . If the constant  $\alpha$  is such that  $e^{-\alpha} = \frac{1}{2}$ , then the error would be reduced by a factor of 2 when  $N$  is only increased by 1 to  $N + 1$ . This compares sharply with a first-order scheme, for which  $N$  would need to be doubled to  $2N$  in order for the error to be reduced by the same factor of 2.

Of course, different schemes have different costs per time step (that is, the coefficients in the  $O(N)$  term could be different), and the coefficient in the error  $O(N^{-k})$  could also be different for different schemes. A more practical measurement of efficiency is the CPU cost to achieve the desired level of error. Figure 1, taken from [18], plots the  $L^1$  error (in logarithm scale) versus CPU time in seconds for three finite difference schemes, of first-, third- and fifth-order accuracy, respectively, solving the eikonal equation which is a nonlinear hyperbolic PDE. It can be seen clearly that, for this example, the higher-order schemes are much more efficient in CPU cost than the lower-order ones to achieve the same level of  $L^1$  error.

A quantitative study on the efficiency of high-order schemes for linear hyperbolic PDEs can be found in, e.g., [8].

### Linear PDEs with Discontinuous Solutions

As we mentioned before, the solutions to hyperbolic PDEs may be discontinuous. For linear PDEs such as (1), the discontinuities in the solution come from the prescribed initial and/or boundary conditions. For nonlinear PDEs such as (5), the discontinuities in the solution may be generated spontaneously even if the initial and boundary conditions are infinitely smooth.

For such discontinuous solutions, the performance of high-order accurate schemes, such as the spectral method and high-order finite difference schemes, will degrade dramatically. Convergence will be completely lost in the strong  $L^\infty$  norm, and it is at most first-order in average norms

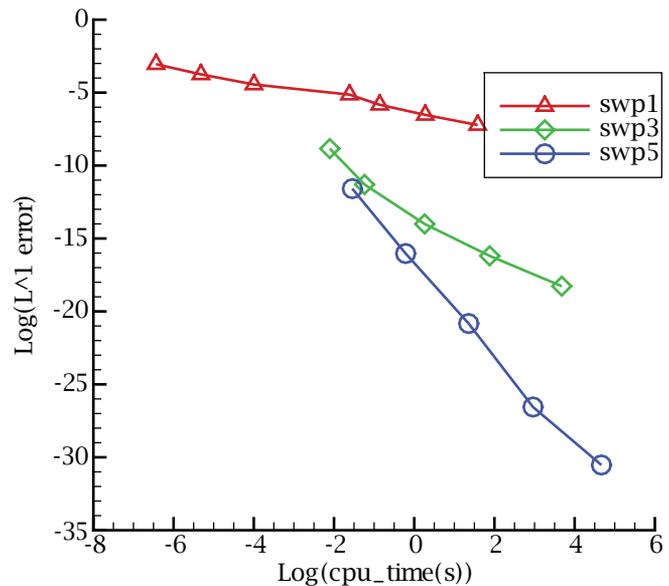


Figure 1. Reproduced from [18].  $L^1$  error versus CPU time. “swp1” is a first-order finite difference scheme, “swp3” a third-order one, and “swp5” a fifth-order one.

such as the  $L^1$  norm. This problem exists already at the approximation level; namely, even the approximation to the initial condition cannot be high order. A simple example is the Fourier spectral solution for the linear equation (1) with  $a = 1$  and an initial condition  $u(x, 0) = x$  for  $-1 \leq x < 1$  with periodic boundary conditions. The approximation to the initial condition is shown in Figure 2, taken from [5]. The initial condition

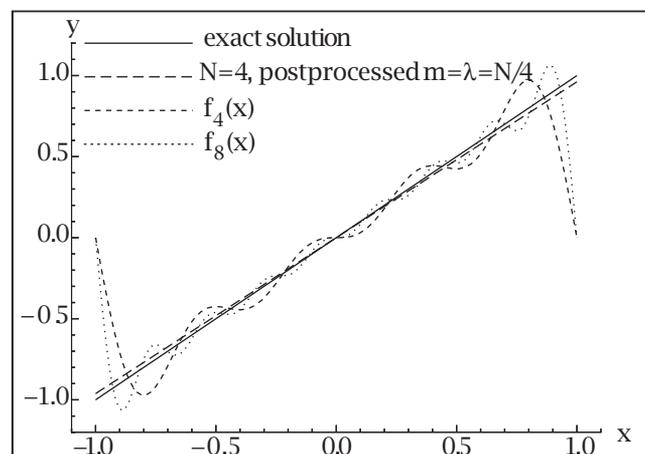


Figure 2. Reproduced from [5]. The function  $f(x) = x$  (solid line), the Fourier partial sum  $f_N(x)$  with  $N = 4$  (short dashed line) and  $N = 8$  (dotted line), and the approximation through the Gegenbauer post-processing for  $N = 4$  (long dashed line).

has a discontinuity at  $x = \pm 1$  when viewed as a periodic function. We can see clearly that there are significant oscillations for the spectral approximation (for  $N = 4$  and  $N = 8$ , denoted by  $f_4(x)$  and  $f_8(x)$  in the figure) near the discontinuity. These are called the *Gibbs oscillations*, and these oscillations are polluted throughout the computational domain, although their magnitudes are smaller away from the discontinuity. Modern nonoscillatory schemes, e.g., the weighted essentially nonoscillatory (WENO) schemes [13], [16], can remove these spurious oscillations and produce sharp, monotone shock transitions. However, with transition point(s) across the discontinuity, which cannot be avoided by conservative shock capturing schemes, the error measured by the  $L^1$  norm still cannot be higher than first order.

Therefore, when measured by the errors in the  $L^p$  norms, a high-order accurate scheme seems to have little advantage over a first-order accurate scheme whenever the solution contains discontinuities. This would seem to be a major difficulty in justifying the design and application of high-order schemes for discontinuous problems.

One possible way to address this difficulty is to measure the error *away from the discontinuities*. For example, in Table 1, again taken from [18], the error measured 0.15 away from the singularity  $(0, 0)$  achieves the designed order of accuracy for third- and fifth-order WENO schemes, which are finite difference schemes, for solving the eikonal equation. For many problems in applications, such high-order accuracy would be desirable and would justify the usage of high-order schemes.

However, such measurement of error is not global, leaving open the theoretical issue of whether a high-order scheme produces solutions which are globally high-order accurate. The proof of high-order accuracy away from the discontinuities is difficult, and for coupled hyperbolic systems, in regions between characteristic lines, the error may be only first-order even though we measure it away from the discontinuities [15].

A major contribution of mathematics to the design and understanding of algorithms in such a situation is the discovery that many high-order schemes are still high-order accurate for discontinuous solutions if we measure the error in a weaker norm, namely, the so-called *negative Sobolev norm*. We recall that the negative Sobolev norm is defined by

$$\|u\|_{-k} = \max_{v \in H^k, v \neq 0} \frac{(u, v)}{\|v\|_k}$$

where  $H^k$  is the space of all functions with finite  $k$ th order Sobolev norm defined by

$$\|v\|_k^2 = \sum_{\ell=0}^k \int_a^b \left( \frac{d^\ell v}{dx^\ell} \right)^2 dx$$

**Table 1. Reproduced from [18].  $L^1$  and  $L^\infty$  errors and orders of accuracy for finite difference first-order (swp1), third-order WENO (swp3), and fifth-order WENO (swp5) fast sweeping schemes solving the eikonal equation. The errors are measured outside the circle  $\sqrt{x^2 + y^2} \leq 0.15$  which contains the singularity  $(0, 0)$ .**

swp5	$L^1$ error	order	$L^\infty$	order
80	6.442E-08	-	5.628E-06	-
160	1.367E-09	5.558	4.525E-07	3.637
320	2.122E-11	6.010	1.515E-09	8.222
640	6.681E-13	4.989	5.146E-11	4.880
swp3	$L^1$ error	order	$L^\infty$	order
80	1.544E-06	-	1.283E-04	-
160	2.890E-07	2.418	4.052E-06	4.984
320	4.697E-08	2.621	1.220E-06	1.731
640	6.161E-09	2.930	1.609E-07	2.923
swp1	$L^1$ error	order	$L^\infty$	order
80	5.413E-03	-	1.302E-02	-
160	2.045E-03	1.404	7.684E-03	0.761
320	1.131E-03	0.854	4.543E-03	0.758
640	6.106E-04	0.890	2.355E-03	0.948

where  $(a, b)$  is our computational interval. A classical example of a function which has a small negative norm but finite strong norms is

$$v^n(x) = \sin nx$$

for large  $n$ . It can be easily verified that

$$\|v^n\|_{-k} = O(n^{-k})$$

for any positive integer  $k$ , yet

$$\|v^n\|_{L^p} = O(1)$$

for any  $1 \leq p \leq \infty$ . This classical example demonstrates a typical feature of functions with small negative norms but large regular norms: they are typically rapidly oscillatory functions. An easy way to understand convergence in such negative norms is to say that the convergence is “in the moments”, namely in integrals against smooth functions. Thus the error  $e_h$  may not be small at any specific point, that is,  $\|e_h\|$  is no better than  $O(h)$  when the norm is taken as the standard  $L^p$  norm, but the integral

$$\int_a^b e_h(x) v(x) dx$$

is small (of the size  $O(h^k)$  for a large  $k$ ) for any smooth function  $v(x)$ . It can be proved, for example in [12], [14], and [15], that a high-order scheme is still high-order accurate for a *linear* hyperbolic PDE, measured in a suitable negative norm, with suitable handling of smoothly cutting off high frequencies. For example, a fourth-order accurate scheme is still fourth-order accurate measured in

the  $\|\cdot\|_{-4}$  norm, and a spectral method is accurate of  $k$ th order for any  $k$  in the negative  $\|\cdot\|_{-k}$  norm.

For a mathematician, it is reassuring to know that high-order accurate schemes are indeed high-order accurate even for discontinuous solutions; only we must measure the error in a particular norm (in this case, the negative norm). However, for an engineer performing scientific computing, it is difficult to argue for the practical value of the scheme by simply knowing that the error is small in some fancy norm. After all, for practical problems one cares about errors in strong norms, such as the  $L^2$  norm. Fortunately, there are many post-processing techniques [1], [4], [6], [12], [15] which can recover high-order accuracy in strong norms, such as the usual  $L^2$  or  $L^\infty$  norm, in smooth regions of the solution, for any sequence of numerical solutions which converges in the negative norm with high-order accuracy to a discontinuous but piecewise smooth solution. In Figure 1, the post-processed solution is a much better approximation to the exact solution than the one before post-processing.

Thus we can conclude that for a linear hyperbolic PDE with discontinuous but piecewise smooth solutions, a good computational strategy is still to use a high-order accurate numerical method. The numerical solutions may be oscillatory and may converge poorly in strong norms, but they do converge in high-order accuracy measured in suitable negative norms. A good post-processor can then be applied to recover high-order accuracy in strong norms in smooth regions of the solution.

### Nonlinear PDEs with Discontinuous Solutions

When the hyperbolic PDE is nonlinear, for example the Burgers' equation (5), the situation becomes more complicated. On the one hand, for a nonlinear hyperbolic PDE, as in the Burgers' equation, the solution will become discontinuous in finite time even if the initial condition is smooth. Therefore, it is even more important to consider numerical methods for discontinuous solutions as they are generic. On the other hand, for nonlinear hyperbolic PDEs, it is still an open problem mathematically whether or in what sense the numerical solutions of a high-order scheme are high-order accurate. We can certainly measure the error in a region away from the discontinuous, as in Table 1, which is for the numerical solution of a nonlinear hyperbolic PDE. However, this is difficult to prove mathematically, may not always be correct (especially for hyperbolic systems of PDEs), and is not an indication of the global error.

Lax [11] argued that, for a nonlinear system, high-order information is retained by a high-order scheme and may be extracted by post-processing. In fact, Lax's argument indicates that more high-order

information is retained in high-order solutions of nonlinear systems than of linear ones since in the nonlinear case the solution operator is contractive. However, it is not clear mathematically in what form such high-order information is retained and how it can be extracted from the numerical solution.

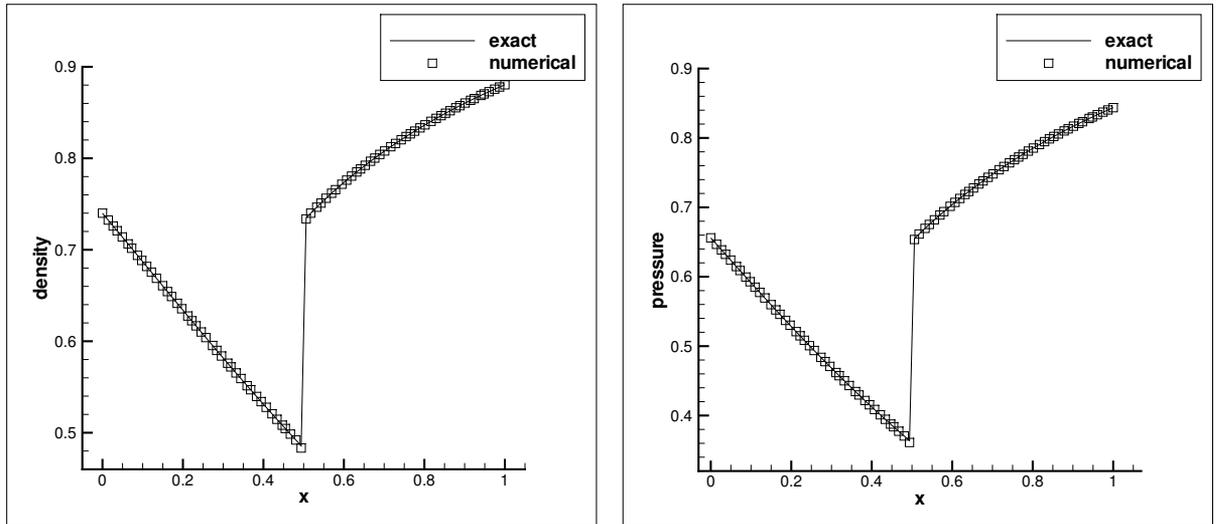
Numerical evidence does indicate that high-order accurate schemes, especially those modern high-order accurate, "high-resolution" schemes which have nonlinear mechanisms to control spurious oscillations near discontinuities, such as WENO schemes [13], [16], are suitable choices for solving nonlinear PDEs with discontinuous solutions, especially when the solutions contain

**Table 2. Euler equations for the nozzle flow. The maximum norm errors to the right of the shock before and after post-processing. Fifth-order WENO computation with  $N$  grid points.**

$N$	before error	after			
		$\lambda$	$m$	error	order
600	$1.37 \times 10^{-3}$	3	3	$8.16 \times 10^{-4}$	
800	$1.09 \times 10^{-3}$	3	4	$3.33 \times 10^{-4}$	3.11
1000	$1.27 \times 10^{-3}$	4	5	$1.66 \times 10^{-4}$	3.13
1200	$1.19 \times 10^{-3}$	5	6	$8.19 \times 10^{-5}$	3.8
1400	$1.27 \times 10^{-3}$	6	7	$4.09 \times 10^{-5}$	4.5
1600	$1.19 \times 10^{-3}$	6	8	$1.72 \times 10^{-5}$	6.46
1800	$1.19 \times 10^{-3}$	7	9	$8.95 \times 10^{-6}$	5.54

**Table 3. Reproduced from [3]. Fourth-order residual distribution WENO scheme for the nozzle flow problem. Errors outside three cells around the shock and numerical orders of accuracy for the density  $\rho$  on nonsmooth meshes with  $N$  cells.**

sub-cell integration in the shocked cell				
$N$	before shock		after shock	
	$L^\infty$ error	order	$L^\infty$ error	order
21	7.09E-07	-	3.35E-05	-
41	6.13E-08	3.53	2.19E-06	3.93
81	4.85E-09	3.66	6.61E-08	5.05
161	3.11E-10	3.96	3.00E-09	4.46
321	2.00E-11	3.96	1.36E-10	4.46
641	1.20E-12	4.05	4.92E-12	4.80
regular integration in the shocked cell				
$N$	before shock		after shock	
	$L^\infty$ error	order	$L^\infty$ error	order
21	7.09E-07	-	6.42E-05	-
41	6.13E-08	3.53	1.42E-05	2.18
81	4.85E-09	3.66	3.37E-06	2.07
161	3.11E-10	3.96	7.29E-07	2.21
321	2.00E-11	3.96	1.77E-07	2.04
641	1.22E-12	4.04	2.52E-08	2.81



**Figure 3.** Reproduced from [3]. Fourth-order residual distribution WENO scheme for the nozzle flow problem. Nonsmooth mesh with 81 cells. Solid lines: exact solution; symbols: numerical solution. Left: density  $\rho$ . Right: pressure  $p$ .

both discontinuities and complicated smooth region structure between discontinuities.

In [7], Gottlieb, Gottlieb, and Shu presented numerical experiments in post-processing numerical solutions of the Euler equation of gas dynamics (a nonlinear system of hyperbolic PDEs) obtained by a fifth-order WENO scheme. The order of accuracy for the numerical solution is diminished in a region downstream of the shock. It is demonstrated that, for steady-state solutions near the shock, the application of the Gegenbauer post-processing method [4] recovers the designed order of accuracy in the  $L_2$  and (emulated) maximum norms (by “emulated” we mean that the maximum error is taken over all points excluding only one near the discontinuity; the domain in which the error is measured therefore gets closer to the discontinuity as the total number of grid points  $N$  increases, thus emulating uniform convergence). Table 2 shows the maximum norm errors for the density in the entire region to the right of the shock before and after post-processing. The maximum norm errors of the numerical solution before post-processing do not decay when the grid is refined. After post-processing the region to the right of the shock, high-order accuracy is recovered. The parameters  $\lambda$  and  $m$  in Table 2 are the ones used in Gegenbauer post-processor [4].

For this nozzle problem, if a certain “sub-cell resolution” technique [9] is applied in the cell containing the discontinuity, then it is possible to obtain the designed high-order accuracy of the numerical scheme without any post-processing. We demonstrate this by the simulation using a fourth-order residual distribution WENO scheme

in [3]. In Figure 3, we observe good resolution of the WENO numerical solution in comparison with the exact solution. In Table 3, we observe that the designed fourth-order accuracy is achieved for both upstream and downstream of the shock if a sub-cell integration technique is applied in the shocked cell, while the accuracy downstream of the shock has lower than the designed fourth-order accuracy if no special treatment is performed in the shocked cell.

It is a challenging mathematical problem to investigate high-order schemes for discontinuous solutions of nonlinear hyperbolic PDEs. Is the numerical solution high-order accurate measured in a suitable norm? Can the high-order information be extracted to yield high-order accuracy in a strong norm? An affirmative answer to these questions will help in the design and application of efficient numerical methods for nonlinear hyperbolic PDEs which are very important in applications.

## References

- [1] J. H. BRAMBLE and A. H. SCHATZ, Higher order local accuracy by averaging in the finite element method, *Mathematics of Computation* **31** (1977), 94–111.
- [2] S. C. BRENNER and L. R. SCOTT, *The Mathematical Theory of Finite Element Methods*, Springer-Verlag, New York, 1994.
- [3] C.-S. CHOU and C.-W. SHU, High order residual distribution conservative finite difference WENO schemes for steady state problems on nonsmooth meshes, *Journal of Computational Physics* **214** (2006), 698–724.
- [4] D. GOTTLIEB and C.-W. SHU, On the Gibbs phenomenon and its resolution, *SIAM Review* **39** (1997), 644–668.
- [5] D. GOTTLIEB, C.-W. SHU, A. SOLOMONOFF, and H. VANDEVEN, On the Gibbs phenomenon I: Recovering exponential accuracy from the Fourier partial sum

- of a nonperiodic analytic function, *Journal of Computational and Applied Mathematics* **43** (1992), 81-98.
- [6] D. GOTTLIEB and E. TADMOR, Recovering pointwise values of discontinuous data within spectral accuracy, in *Progress and Supercomputing in Computational Fluid Dynamics*, (E. M. Murman and S. S. Abarbanel, eds.), Birkhäuser, Boston, 1985, pp. 357-375.
- [7] S. GOTTLIEB, D. GOTTLIEB, and C.-W. SHU, Recovering high order accuracy in WENO computations of steady state hyperbolic systems, *Journal of Scientific Computing* **28** (2006), 307-318.
- [8] B. GUSTAFSSON, H.-O. KREISS, and J. OLIGER, *Time Dependent Problems and Difference Methods*, John Wiley & Sons, New York, 1995.
- [9] A. HARTEN, ENO schemes with subcell resolution, *Journal of Computational Physics* **83** (1989), 148-184.
- [10] J. HESTHAVEN, S. GOTTLIEB, and D. GOTTLIEB, *Spectral Methods for Time-Dependent Problems*, Cambridge University Press, Cambridge, 2007.
- [11] P. D. LAX, Accuracy and resolution in the computation of solutions of linear and nonlinear equations, in *Recent Advances in Numerical Analysis*, (C. De Boor and G. Golub, eds.), Academic Press, 1978.
- [12] P. D. LAX and M. MOCK, The computation of discontinuous solutions of linear hyperbolic equations, *Communications on Pure and Applied Mathematics* **31** (1978), 423-430.
- [13] X.-D. LIU, S. OSHER, and T. CHAN, Weighted essentially nonoscillatory schemes, *Journal of Computational Physics* **115** (1994), 200-212.
- [14] A. MAJDA, J. McDONOUGH and S. OSHER, The Fourier method for nonsmooth initial data, *Mathematics of Computation* **32** (1978), 1041-1081.
- [15] A. MAJDA and S. OSHER, Propagation of error into regions of smoothness for accurate difference approximations to hyperbolic equations, *Communications on Pure and Applied Mathematics* **30** (1977), 671-705.
- [16] C.-W. SHU, High order weighted essentially nonoscillatory schemes for convection dominated problems, *SIAM Review* **51** (2009), 82-126.
- [17] J. SMOLLER, *Shock Waves and Reaction-diffusion Equations*, second edition, Springer-Verlag, New York, 1994.
- [18] T. XIONG, M. ZHANG, Y.-T. ZHANG, and C.-W. SHU, Fifth order fast sweeping WENO scheme for static Hamilton-Jacobi equations with accurate boundary treatment, *Journal of Scientific Computing* **45** (2010), 514-536.

**MATHEMATICS** AT THE **NATIONAL SECURITY AGENCY**

Make a calculated difference with what you know.



**Tackle the coolest problems ever.**

You already know that mathematicians like complex challenges. But here's something you may not know.

The National Security Agency is the nation's largest employer of mathematicians. In the beautiful, complex world of mathematics, we identify structure within the chaotic and patterns among the arbitrary.

Work with the finest minds, on the most challenging problems, using the world's most advanced technology.

**KNOWINGMATTERS**

**Excellent Career Opportunities for Experts in the Following:**

- Number Theory
- Probability Theory
- Group Theory
- Finite Field Theory
- Combinatorics
- Linear Algebra
- >> **Plus** other opportunities



**NSA**  
www.NSA.gov/Careers  
**APPLY TODAY**

Search NSACareers

Available on the App Store

Search NSA to Download

**WHERE INTELLIGENCE GOES TO WORK®**

U.S. citizenship is required. NSA is an Equal Opportunity Employer.