A limiting strategy for the back and forth error compensation and correction method for solving advection equations

Lili Hu [*], Yao Li [†], Yingjie Liu [‡]

April 22, 2014

Abstract

We further study the properties of the back and forth error compensation and correction (BFECC) method for advection equations such as those related to the level set method and for solving Hamilton-Jacobi equations on unstructured meshes. In particular, we develop a new limiting strategy which requires another backward advection in time so that overshoots/undershoots on the new time level get exposed when they are transformed back to compare with the solution on the old time level. This new technique is very simple to implement even for unstructured meshes and is able to eliminate artifacts induced by jump discontinuities in derivatives of the solution as well as by jump discontinuities in the solution itself (even if the solution has large gradients in the vicinities of a jump). Typically, a formal second order method for solving a time dependent Hamilton-Jacobi equation requires quadratic interpolation in space. A BFECC method on the other hand only requires linear interpolation in each step, thus is local and easy to implement even for unstructured meshes.

1 Introduction

The BFECC method was proposed in [10] as a convenient method to achieve better computational accuracy for the level set advection [24]. The idea is that when

[*](E-mail: lhu33@math.gatech.edu)
School of Mathematics, Georgia Institute of Technology, Atlanta, GA 30332. Research supported in part by NSF grant DMS-1115671.

[†](E-mail: yaoli@cims.nyu.edu)
Courant Institute of Mathematics, New York University, New York.

[‡](E-mail: yingjie@math.gatech.edu)
School of Mathematics, Georgia Institute of Technology, Atlanta, GA 30332. Research supported in part by NSF grant DMS-1115671.

a solution is advected forward and then backward for a time step, the difference between two copies of the solution at the initial time level provides information about the numerical error of the underlying scheme. This information can be used to compensate the solution before a third advection forward in time, resulting in more accurate numerical solution at the next time level. For the linear advection equation on rectangular meshes, this procedure has been shown in [12] to improve the order of accuracy of an odd order scheme by one (in both space and time) and also stabilize the scheme if its amplification factor is less than 2 with some reasonable conditions. The BFECC method coupled with an underlying semi-Lagrangian scheme has been studied for various fluid and level set interface advections, e.g. in [11, 15, 16, 17, 18], and for the Lattice Boltzmann method on quad-tree grids [5]. Without the CFL restriction, this combination is easy to implement on various meshes. A generalized MacCormack scheme without CFL restriction is developed and applied to fluid simulations in [26], in which the error information from the forward and backward advections is applied directly to the previously obtained solution at the next time level. These advection equations belong to the class of Hamilton-Jacobi equations. The numerical monotone Hamiltonian along with a high order ENO approach have been developed for Hamilton-Jacobi equations in [25]. See [1] for a Lax-Friedrichs-type numerical monotone Hamiltonian on 2D triangular meshes and [30] for incorporating a high order WENO reconstruction. High order central schemes for solving Hamilton-Jacobi equations have been proposed in e.g. [9, 21, 4], and some recent development can be found in [19, 3, 20]. A conservative scheme developed for solving a conservation law typically use an r-th degree polynomial interpolation to achieve $(r+1)$-th formal order of accuracy. When the scheme is modified for solving a Hamilton-Jacobi equation, however, $(r+1)$-th degree polynomial interpolation is usually needed to achieve $(r+1)$-th formal order of accuracy. For example, a second order non-oscillatory scheme for solving a Hamilton-Jacobi equation may need a quadratic interpolation. On the other hand, when BFECC is applied to a first order scheme using only local linear interpolation, it can improve both its temporal and spatial order of accuracy to second order. This is very convenient for unstructured meshes since a linear interpolation only uses information from adjacent grid points. However, to have this convenience, nonlinear limiting techniques based on using redundant high order information may not be applied here easily, such as those in the MUSCL [23] and ENO [14] schemes. In [12], a limiting technique is introduced which is based on using a locally constant advection velocity to compute the back-and-forth error wherever a singularity in the velocity field is detected. This technique works only if the solution is at least Lipschitz continuous, such as a level set function. A simple limiting technique is used in [26] for the BFECC and the modified unconditional stable MacCormack scheme by essentially regulating the solution at a grid point within extrema on neighboring grid points. Another limiting technique for BFECC can be found in [13]. In this paper we introduce a new limiting strategy for BFECC based on the following consideration. Assuming the solution U^n at the time t_n is accurate and we have computed the

solution U^{n+1} at the time t_{n+1}, it's very difficult to detect where the new solution U^{n+1} has overshoots/undershoots since we don't know the exact solution. However, if we can approximately advect U^{n+1} backward in time to the time level t_n then we have an accurate solution U^n to compare with. This idea works like a easy extension to the strategy used in BFECC.

2 Preliminary: Back and Forth Error Compensation and Correction (BFECC)

Consider the convection equation on $\mathbb{R}^N$

$$\frac{\partial u}{dt} + \mathbf{a} \cdot \bigtriangledown u = 0. \tag{1}$$

Let $\mathcal{L}$ be a numerical scheme that updates the numerical solution from the time t_n to t_{n+1}, $t_n < t_{n+1}$. Let $\mathcal{L}^*$ be the numerical scheme that updates the numerical solution from the time t_{n+1} to t_n by applying $\mathcal{L}$ to the time-reversed equation of (1)

$$\frac{\partial u}{dt} - \mathbf{a} \cdot \bigtriangledown u = 0. \tag{2}$$

Let U^n be the numerical solution given at the time t_n, then the BFECC algorithm can be described as follows [10].

1. **Forward advection.**
   $\tilde{U}^{n+1} = \mathcal{L}U^n$.

2. **Backward advection.**
   $\tilde{U}^n = \mathcal{L}^*\tilde{U}^{n+1}$.

3. **Forward advection again using modified solution at the time t_n.**
   $U^{n+1} = \mathcal{L}(U^n + e^{(1)})$, where $e^{(1)} = \frac{1}{2}(U^n - \tilde{U}^n)$.

Here we call $e^{(1)}$ the back-and-forth error. Since the forward advection and backward advection use the same scheme (note that $\mathcal{L}^*$ is $\mathcal{L}$ applied to the time-reversed equation), we assume that they introduce similar amount of error. Therefore $e^{(1)} = \frac{1}{2}(U^n - \tilde{U}^n)$ provides an estimate of the error which is going to be subtracted from the solution during the forward advection. Consider a rectangular mesh on $\mathbb{R}^N$ with the mesh size h and grid point $x_j = jh$ for any multi-index j. Let U_j^n be the numerical solution at x_j and time t_n, and let $k = t_{n+1} - t_n$ be the time step size. Assume $k = \theta h$ for some fixed constant θ during mesh refinement. Let $\mathbf{a}$ be a constant vector in $\mathbb{R}^N$. We assume a linear scheme $\mathcal{L}$ can be written in the form of

$$U_i^{n+1} = \sum_{|j| \le l} C_j U_{i+j}^n, \tag{3}$$

where l is a positive integer and C_j is a real constant depending only on $\mathbf{a}$, the multi-index j and the constant $\theta = k/h$. We will view scheme (3) as

$$U^{n+1}(x) = \sum_{|j|\le l} C_j U^n(x + jh), \quad \text{for any } x \in \mathbb{R}^N, \tag{4}$$

for convenience in the following Fourier analysis, where $U^n(x)$ is a continuous function with compact support in $\mathbb{R}^N$.

Let $\rho_{\mathcal{L}}$ denote the Fourier symbol of the numerical scheme $\mathcal{L}$, and $\rho_{\mathcal{L}^*}$ denote the Fourier symbol of $\mathcal{L}^*$. For example, the operator $\mathcal{L}$ could be the upwind scheme, Lax-Friedrichs scheme, (unstable) center difference scheme, CIR scheme [6] or some other schemes. In addition, we assume $\rho_{\mathcal{L}^*} = \overline{\rho_{\mathcal{L}}}$, the complex conjugate of $\rho_{\mathcal{L}}$ (which is true for essentially all commonly used first order linear schemes [12]).

It has been proved in [12] that the BFECC algorithm creates a stable scheme if the amplification factor $|\rho_{\mathcal{L}}|$ of $\mathcal{L}$ satisfies $|\rho_{\mathcal{L}}| \le 2$. Furthermore, if the order of accuracy of the scheme $\mathcal{L}$ is r for some positive odd integer r, then the order of accuracy after applying BFECC is $r + 1$.

Take a look at $\tilde{U}^n = \mathcal{L}^*\mathcal{L}U^n$ in the BFECC algorithm. If $\tilde{U}^n = U^n$ we call $\mathcal{L}$ *time-reversible*. We have the following results for a time-reversible linear scheme.

Lemma 2.1. *A linear scheme $\mathcal{L}$ is time-reversible if and only if $|\rho_{\mathcal{L}}| = 1$.*

Proof. Let $U^{n+1} = \mathcal{L}(U^n)$ and $\tilde{U}^n = \mathcal{L}^*(U^{n+1})$. Applying the Fourier transform to the two equations and using the assumption $\rho_{\mathcal{L}^*} = \overline{\rho_{\mathcal{L}}}$, we have $\hat{\tilde{U}}^n = |\rho_{\mathcal{L}}|^2 \hat{U}^n$. The proof is complete.

Theorem 2.2. *If a linear scheme $\mathcal{L}$ is time-reversible and is at least first order accurate, then $\mathcal{L}$ is at least second order accurate.*

The proof follows that of [22].

Proof. Let $U^{n+1} = \mathcal{L}U^n$. Applying the Fourier transform to the differential equation $u_t = -\mathbf{a} \cdot \bigtriangledown u$ we have

$$\hat{u}_t = -\mathbf{a} \cdot \xi i \hat{u},$$

where ξ is the Fourier dual variable. Therefore

$$\hat{u}(t_{n+1}) = e^{-\mathbf{a}\cdot\xi k i}\hat{u}(t_n) = \{1 - \mathbf{a} \cdot \xi k i - \frac{1}{2}(\mathbf{a} \cdot \xi k)^2 + \mathcal{O}(|\xi k|^3)\}\hat{u}(t_n).$$

Since $\mathcal{L}$ is first order accurate, we can write [22]

$$\rho_{\mathcal{L}} = 1 - \mathbf{a} \cdot \xi k i + bk^2 + \mathcal{O}(|\xi k|^3),$$

where b is real since the coefficients in the linear scheme $\mathcal{L}$ (3, 4) are real and independent of k or h. Since $|\rho_{\mathcal{L}}| = 1$ according to Lemma 2.1, we have

$$1 = |\rho_{\mathcal{L}}|^2 = 1 + (\mathbf{a} \cdot \xi k)^2 + 2bk^2 + \mathcal{O}(|\xi k|^3).$$

Therefore

$$b = -\frac{1}{2}(\mathbf{a}\cdot\xi)^2.$$

The proof is complete.

It's interesting to see what happen if we apply BFECC to a linear scheme $\mathcal{L}$ recursively. Let

$$\mathcal{L}_1 = \mathcal{L}[I + \frac{1}{2}(I - \mathcal{L}^*\mathcal{L})], \tag{5}$$

the scheme obtained by applying BFECC to scheme $\mathcal{L}$ where I is the identity operator, and let

$$\mathcal{L}_{k+1} = \mathcal{L}_k[I + \frac{1}{2}(I - \mathcal{L}_k^*\mathcal{L}_k)], \text{ for } k = 1, 2, 3, \cdots. \tag{6}$$

It's easy to see that the Fourier symbols of $\mathcal{L}_k$ satisfy $\rho_{\mathcal{L}_k} = \overline{\rho_{\mathcal{L}_k^*}}$ for $k = 1, 2, 3, \cdots$ by induction (because we assume $\mathcal{L}$ satisfies this property to begin with). We have the following theorem.

Theorem 2.3. *If a linear scheme $\mathcal{L}$ satisfies $|\rho_{\mathcal{L}}| \in (0, \sqrt{3}) \cup (\sqrt{3}, 2]$, then*

$$\lim_{k\to\infty} |\rho_{\mathcal{L}_k}| = 1.$$

Proof. Since

$$|\rho_{\mathcal{L}_{k+1}}| = |\rho_{\mathcal{L}_k}||\frac{3}{2} - \frac{1}{2}|\rho_{\mathcal{L}_k}|^2|, \tag{7}$$

it's easy to see that if $|\rho_{\mathcal{L}_k}| \le 2$ then $|\rho_{\mathcal{L}_{k+1}}| \le 1$. And $|\rho_{\mathcal{L}_{k+1}}| = 0$ if and only if $|\rho_{\mathcal{L}_k}| = 0$ or $\sqrt{3}$; $|\rho_{\mathcal{L}_{k+1}}| = 1$ if and only if $|\rho_{\mathcal{L}_k}| = 1$ or 2. The fixed points of the iteration (7) are 0 and 1. Therefore if $|\rho_{\mathcal{L}}| \in (0, \sqrt{3}) \cup (\sqrt{3}, 2]$, then $|\rho_{\mathcal{L}_1}| \in (0, 1]$. Since $1 > x|\frac{3}{2} - \frac{1}{2}x^2| > x$ for $x \in (0, 1)$, we conclude by induction that $1 > |\rho_{\mathcal{L}_{k+1}}| > |\rho_{\mathcal{L}_k}|$ for $k = 1, 2, 3, \cdots$ if $|\rho_{\mathcal{L}_1}| \in (0, 1)$. Therefore $|\rho_{\mathcal{L}_k}|$ must converge as $k \to \infty$ and its limit must be 1 by passing to the limit of equation (7). The proof is complete.

Recalling Lemma 2.1, we can see that applying BFECC iteratively approaches a time-reversible scheme.

3 Eliminating Spurious Oscillations

When the solution is not smooth, a nonlinear limiting technique is usually required for second order (or higher order) schemes to remove spurious oscillations from the numerical solution.

Let $\mathcal{L}$ be a linear scheme and let $e^{(1)} = \frac{1}{2}(U^n - \mathcal{L}^*\mathcal{L}U^n)$ be the back-and-forth error where U^n is the numerical solution at the time t_n. If we replace $\mathcal{L}$ with $\mathcal{L}_k$, the k-th iteration of the BFECC procedure as defined in Sec. 2, we expect the back-and-forth error $e^{(1)}$ to decrease with increasing k wherever the solution is locally sufficiently smooth because of Theorem 2.3. However, this is not practical due to the complexity of even computing $\mathcal{L}_1^*\mathcal{L}_1U^n$ (with $k = 1$ in (6)). $\mathcal{L}_1U^n$

is supposed to be more accurate than $\mathcal{L}U^n$ in approximating the solution at the time level t_{n+1} provided the solution is smooth. If we transform $\mathcal{L}_1 U^n$ back to the time level t_n with $\mathcal{L}^*$ instead of $\mathcal{L}_1^*$, a larger error of approximately $e^{(1)}$ is also introduced. Therefore with the error correction $\mathcal{L}^*\mathcal{L}_1 U^n + e^{(1)}$ should be an accurate approximation of $\mathcal{L}_1^*\mathcal{L}_1 U^n$ with smaller cost. We define another error at the time level t_n as $e^{(2)} = U^n - (\mathcal{L}^*\mathcal{L}_1 U^n + e^{(1)})$ and have the following theorem.

Theorem 3.1. *Let $\mathcal{L}$ be a linear scheme, $e^{(1)} = \frac{1}{2}(U^n - \mathcal{L}^*\mathcal{L}U^n)$ and $e^{(2)} = U^n - (\mathcal{L}^*\mathcal{L}_1 U^n + e^{(1)})$ where $\mathcal{L}_1$ is the scheme obtained by applying BFECC to $\mathcal{L}$, then $e^{(2)} = e^{(1)} - \mathcal{L}^*\mathcal{L}e^{(1)}$.*

Proof.

$$\begin{aligned} e^{(2)} &= U^n - \mathcal{L}^*\mathcal{L}_1 U^n - e^{(1)} \\ &= U^n - \mathcal{L}^*[\mathcal{L}(\tfrac{3}{2}I - \tfrac{1}{2}\mathcal{L}^*\mathcal{L})]U^n - e^{(1)} \\ &= U^n - \mathcal{L}^*\mathcal{L}U^n - \mathcal{L}^*\mathcal{L}[\tfrac{1}{2}(U^n - \mathcal{L}^*\mathcal{L}U^n)] - e^{(1)} \\ &= 2e^{(1)} - \mathcal{L}^*\mathcal{L}e^{(1)} - e^{(1)} \\ &= e^{(1)} - \mathcal{L}^*\mathcal{L}e^{(1)}. \end{aligned} \tag{8}$$

The proof is complete.

Now we are able to estimate the size of $e^{(2)}$ relative to $e^{(1)}$.

Corollary 3.2. *Let $\mathcal{L}$ be a linear scheme with its amplification factor $|\rho_{\mathcal{L}}| \le \sqrt{2}$, then $||e^{(2)}|| \le ||e^{(1)}||$ where $||\cdot||$ denotes the L^2-norm.*

Proof. From Theorem 3.1 we obtain via Fourier transform

$$\hat{e}^{(2)} = (1 - |\rho_{\mathcal{L}}|^2)\hat{e}^{(1)}.$$

With the Parseval's identity, the proof is complete.

Therefore on average $|e^{(2)}|$ is no more than $|e^{(1)}|$ as we expect. However in the non smooth area of the solution, $|e^{(2)}|$ could be larger than $|e^{(1)}|$. In fact at a grid point where $e^{(2)}$ is greater than $e^{(1)}$, there could be overshoots of the numerical solution caused by large values of $e^{(1)}$ at adjacent grid points. To see this, let's suppose

$$(\mathcal{L}^*\mathcal{L}e^{(1)})(x_i) = \sum_{j\in I} c_j e^{(1)}_{i+j} \tag{9}$$

where the set $i + I$ contains all grid point indices involved in the computation of $\mathcal{L}^*\mathcal{L}e^{(1)}$ at x_i, in particular, $i \in i + I$. Suppose $\mathcal{L}$ is consistent, monotone and at least first order accurate. Then it's easy to see by using the Taylor expansion around x_i that

$$\begin{aligned} &(a)\ \ 0 \le c_j \le 1, \ \text{ for all } \ j \in I; \\ &(b)\ \ \textstyle\sum_{j\in I} c_j = 1; \ \text{ and} \\ &(c)\ \ \textstyle\sum_{j\in I, j\neq 0} j c_j = 0. \end{aligned} \tag{10}$$

Theorem 3.3. *Suppose the linear scheme $\mathcal{L}$ is consistent, monotone and at least first order accurate. If $|e_i^{(1)}|$ is a maximum among $\{|e_{i+j}^{(1)}| : j \in I\}$, and $(\mathcal{L}^*\mathcal{L}e^{(1)})(x_i)$ is of the same sign as $e_i^{(1)}$, then*

$$|e_i^{(2)}| \leq |e_i^{(1)}|.$$

Proof. Without loss of generality, suppose $e_i^{(1)} > 0$. Since $\sum_{j\in I} c_j e_{i+j}^{(1)}$ is a convex average of $e_{i+j}^{(1)}$ and $e_i^{(1)}$ is the local maximum, we have

$$\sum_{j\in I} c_j e_{i+j}^{(1)} \leq e_i^{(1)}.$$

Also since $(\mathcal{L}^*\mathcal{L}e^{(1)})(x_i) = \sum_{j\in I} c_j e_{i+j}^{(1)} > 0$ from (9) and the assumption of the theorem, we conclude that

$$|e_i^{(1)} - (\mathcal{L}^*\mathcal{L}e^{(1)})(x_i)| \leq |e_i^{(1)}|.$$

Recalling Theorem 3.1, the proof is complete.

Therefore a local maximum in $|e^{(1)}|$ that is large enough will result in a smaller $e^{(2)}$ at the same location. However, since

$$e_i^{(2)} = e_i^{(1)} - (\mathcal{L}^*\mathcal{L}e^{(1)})(x_i) = e_i^{(1)} - \sum_{j\in I} c_j e_{i+j}^{(1)},$$

a $e_{i+j}^{(1)}$ with much larger absolute value than that of $e_i^{(1)}$ tends to cause $|e_i^{(2)}| > |e_i^{(1)}|$. Therefore wherever $|e_i^{(2)}| > |e_i^{(1)}|$ is detected, we limit its adjacent back-and-forth error $e_{i+j}^{(1)}$ to no larger than $e_i^{(1)}$ in their absolute values.

Remark. Let r be the order of accuracy of the scheme $\mathcal{L}$. Even if this limiting procedure is applied accidentally in the smooth area of the solution, it's only going to create a $\mathcal{O}(h^{r+2})$ local error which won't change the order of accuracy after applying the BFECC algorithm, as long as the modified value $\tilde{e}_{i+j}^{(1)}$ (after applying the limiting procedure) is a convex average of $e_{i+j}^{(1)}$ and $e_i^{(1)}$. In fact,

$$\begin{aligned} \tilde{e}_{i+j}^{(1)} - e_{i+j}^{(1)} &= (1-\theta)e_{i+j}^{(1)} + \theta e_i^{(1)} - e_{i+j}^{(1)}, \text{ for some } \theta \in [0,1] \\ &= \theta(e_i^{(1)} - e_{i+j}^{(1)}) \\ &= \mathcal{O}(h^{r+2}), \end{aligned} \tag{11}$$

since $e^{(1)} = \mathcal{O}(h^{r+1})$ (it is proportional to the local error of scheme $\mathcal{L}$).

3.1 Limiting Algorithm

Let $\mathcal{L}$ be a linear scheme and U^n be the numerical solution given at the time t_n, then the BFECC algorithm coupled with the limiting technique can be implemented as follows.

1. **Forward advection.**
   $\tilde{U}^{n+1} = \mathcal{L}U^n$.

2. **Backward advection.**
   $\tilde{U}^n = \mathcal{L}^*\tilde{U}^{n+1}$.

3. **Forward advection again using modified solution at the time t_n.**
   $\tilde{V}^{n+1} = \mathcal{L}(U^n + e^{(1)})$, where $e^{(1)} = \frac{1}{2}(U^n - \tilde{U}^n)$.

4. **Backward advection to define a comparative error $e^{(2)}$.**
   $e^{(2)} = U^n - (\mathcal{L}^*(\tilde{V}^{n+1}) + e^{(1)})$.

5. **Limiting.**
   Define a copy of $e^{(1)}$, $\tilde{e}^{(1)} = e^{(1)}$. At every grid point x_i such that $|e_i^{(2)}| > |e_i^{(1)}|$, perform the limiting at adjacent grid points so that $\tilde{e}_j^{(1)} = \text{minmod}(e_i^{(1)}, \tilde{e}_j^{(1)})$, for every grid point j adjacent to grid point i, $j \neq i$.

6. **Forward advection with modified solution at the time t_n.**
   $U^{n+1} = \mathcal{L}(U^n + \tilde{e}^{(1)})$.

Here

$$\text{minmod}(x, y) = \begin{cases} \min(x, y), & \text{if } x, y > 0, \\ \max(x, y), & \text{if } x, y < 0, \\ 0, & \text{otherwise.} \end{cases}$$

It is a commonly used limiter function that returns a convex average of x and y. This procedure is very easy to implement because basically it calls a subroutine (scheme $\mathcal{L}$) 5 times. Note that the limiting procedure modifies $e^{(1)}$ only in the vicinities of singularities of the solution, and the backward advection step in Step 4. can be applied selectively. This could reduce the complexity to about 4 times that of scheme $\mathcal{L}$ by first using a low cost detector(e.g. in [12]) to find the non smooth area of the solution. One could further reduce the cost by applying the last advection step in the non smooth area only.

4 Numerical Results

The performance of BFECC with the limiting algorithm is demonstrated by the following numerical examples.

4.1 1-D linear equation

Consider the following 1-D linear equation

$$\frac{\partial u(x,t)}{\partial t} + \frac{\partial u(x,t)}{\partial x} = 0, \qquad (x,t) \in [0,2] \times [0,20] \tag{12}$$

$$u(x,0) = 1, \qquad x \in [\frac{2}{3}, \frac{4}{3}] \tag{13}$$
$$u(x,0) = 0, \qquad \text{otherwise}$$

with periodic boundary conditions. The solution of equation (12) at the final time $T = 20$ is identical to the initial solution. We compare the performance of CIR, CIR+BFECC and CIR+BFECC+limiting algorithms for the present 1-D linear equation with varying CFL numbers.

Fig. 1 presents the numerical solutions with $CFL = 0.8$. Note that the CIR scheme with CFL number less than 1 is the same as the upwind scheme. Our limiting algorithm eliminates the spurious oscillations that appear in the CIR+BFECC algorithm. Also the shape of the square wave is well-preserved by the CIR+BFECC+limiting algorithm at $T = 20$. By contrast, very strong numerical smearing exists in the CIR method.

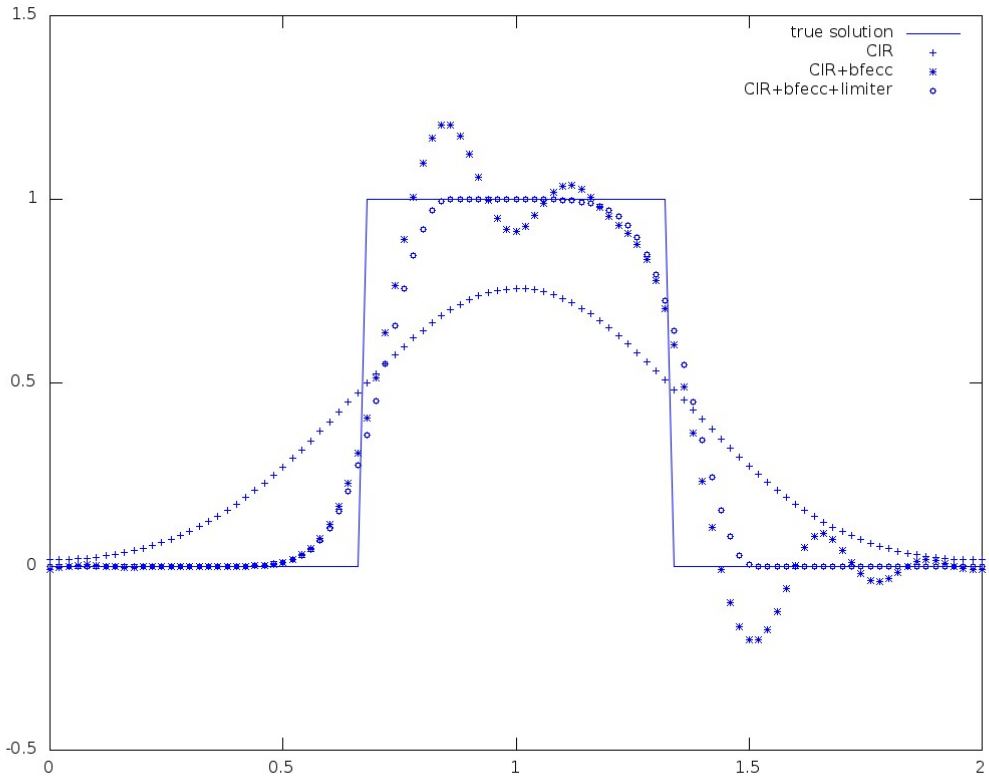

Figure 1: 1-D linear equation with CFL = 0.8, square wave initial condition. $h = 0.02$, $T = 20$.

The numerical results with CFL number 5.8 are shown in Fig. 2. The shape of the square wave is better preserved by all three methods because there are less

computational steps. The spurious oscillations in the CIR+BFECC algorithm are again eliminated by the limiting procedure, while strong numerical diffusion is still appearing in the CIR scheme.

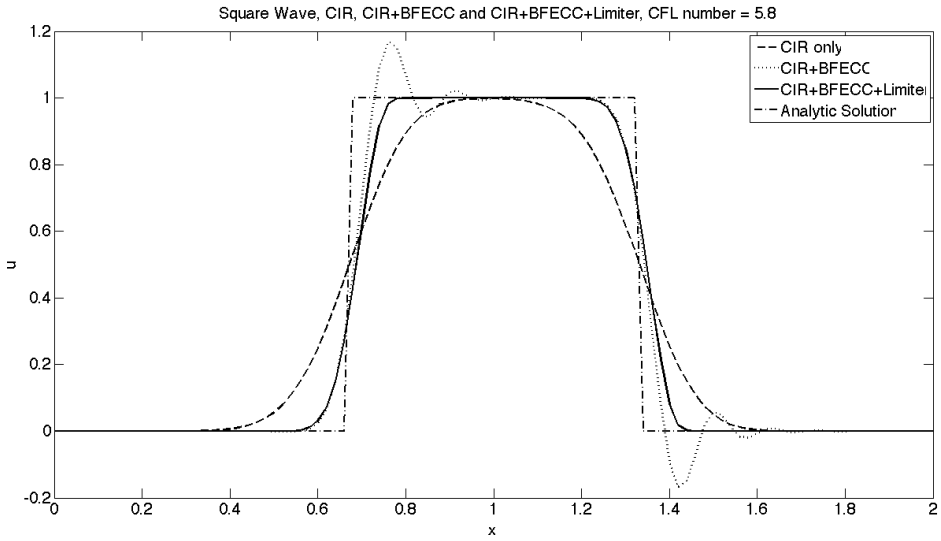

Figure 2: 1-D linear equation with CFL = 5.8, square wave initial condition, $h = 0.02$, $T = 20$.

Numerical solutions of equation (12) with different initial conditions are presented in the following. For the *pyramid* initial condition, we mean the following function defined on the interval $[0, 2]$

$$u(x, 0) = 2(1 - |x - 1|).$$

And the *curved square wave* initial condition is the cubic function defined on the interval $[0, 2]$ as follows

$$u(x, 0) = \begin{cases} \frac{53}{26} - \frac{4}{13}(2 - x)^3, & x \in [\frac{2}{3}\frac{4}{3}) \\ 0, & \text{otherwise}. \end{cases}$$

In Fig. 3 we compare the results of three different methods (CIR, CIR+BFECC and CIR+BFECC+Limiting) for equation (12) with the pyramid initial condition. The results with the curved square wave initial condition are shown in Fig. 4. Clearly the limiting procedure removes all the artifacts generated by CIR+BFECC while retaining its higher resolution.

4.2 1D linear problem with nonzero forcing

We also consider the following problem with a nonzero forcing term

$$\frac{\partial u(x, t)}{\partial t} + \frac{\partial u(x, t)}{\partial x} = \frac{1}{2}, \qquad (x, t) \in [0, 2] \times [0, 20] \tag{14}$$

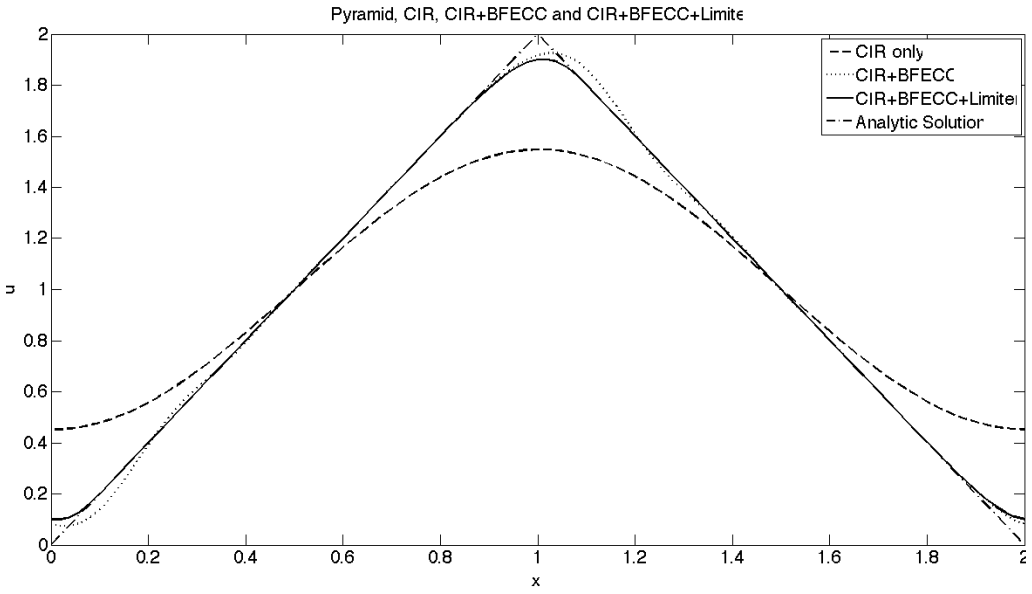

Figure 3: 1-D linear equation with CFL = 0.8 , pyramid initial condition, $h = 0.02$, $T = 20$.

$$u(x, 0) = 2(1 - |x - 1|),$$

with periodic boundary conditions. The initial solution is still the "pyramid" function. In Fig. 5 and 6, numerical solutions of equation (14) are demonstrated. The CFL numbers used are 0.8 and 5.8 respectively. It's clear that the limiting algorithm performs well for this problem.

4.3 2D linear problem

We study a 2-D rotation of a "cubic stair" on the domain $[0\,,100] \times [0\,,100]$

$$u_0(x, y) = \begin{cases} 64000(\frac{53}{26} - \frac{4}{13}(\frac{x}{40} + \frac{3}{8})^3), & (x, y) \in [5\,,45] \times [5\,,45] \\ 0, & \text{otherwise}\,. \end{cases}$$

Consider the equation

$$\frac{\partial \phi}{\partial t} + \vec{v} \cdot \nabla \phi = 0 \tag{15}$$

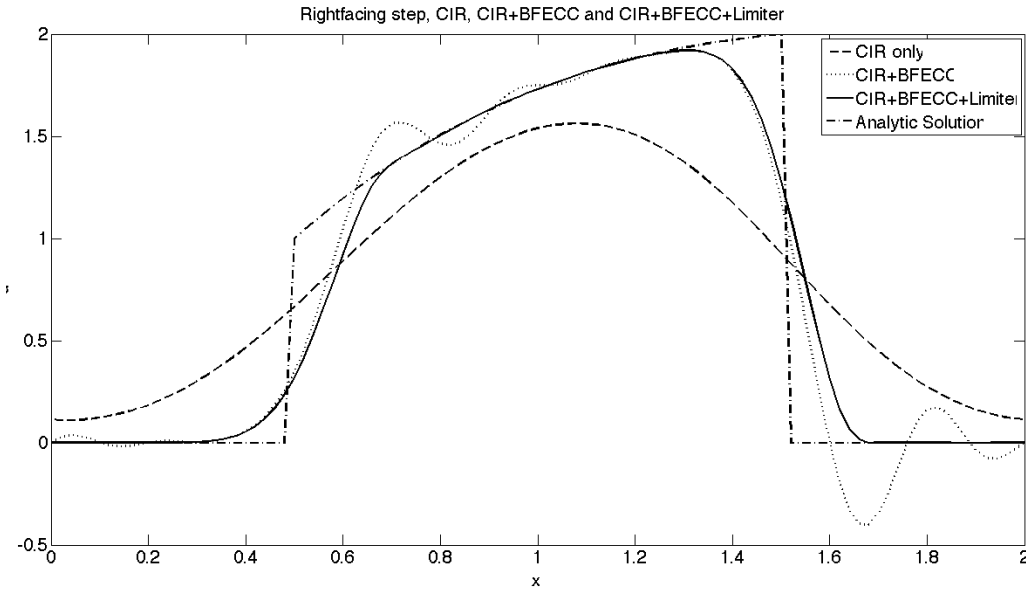

Figure 4: 1-D linear equation with CFL = 0.8 , curved square wave initial condition, $h = 0.02$, $T = 20$.

with the initial condition $\phi(x, y, 0) = u_0(x, y)$, where

$$\vec{v}(x, y) = (\frac{\pi}{314}(50 - y)\,, \frac{\pi}{314}(50 - x))\,.$$

Equation (15) describes the linear rotation around the center $(50\,, 50)$ with the shape of $u_0(x, y)$ preserved.

We solve equation (15) numerically by using CIR scheme only, CIR+BFECC and CIR+BFECC+Limiting on a triangular mesh. The numerical results are demonstrated in Figure 7. It is easy to see that the CIR scheme alone has significant numerical diffusion while CIR+BFECC causes some overshoots near the edge of the "cubic stair" due to the discontinuity of the solution. With the limiting algorithm, the spurious oscillations are eliminated without introducing extra numerical diffusion.

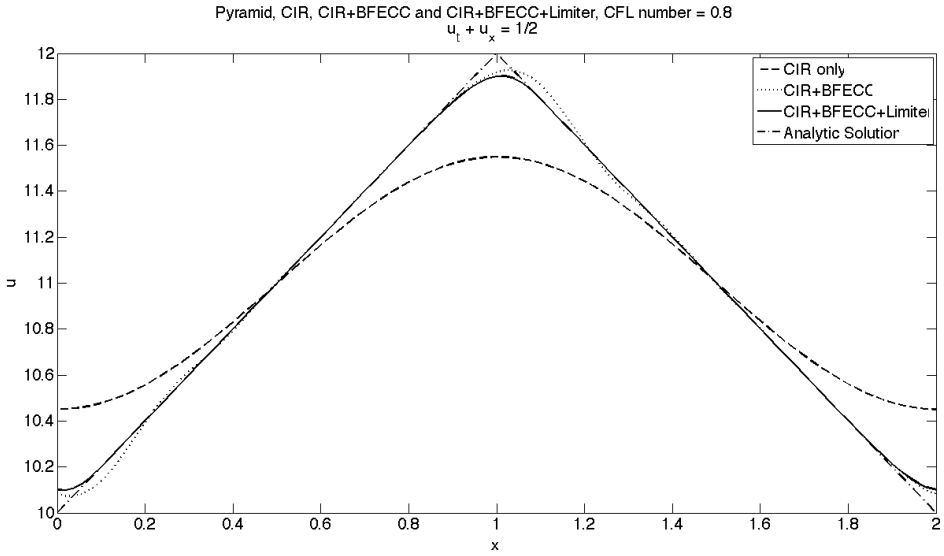

Figure 5: 1-D linear equation with nonzero right-hand side, CFL = 0.8 , pyramid initial condition, $h = 0.02$, $T = 20$.

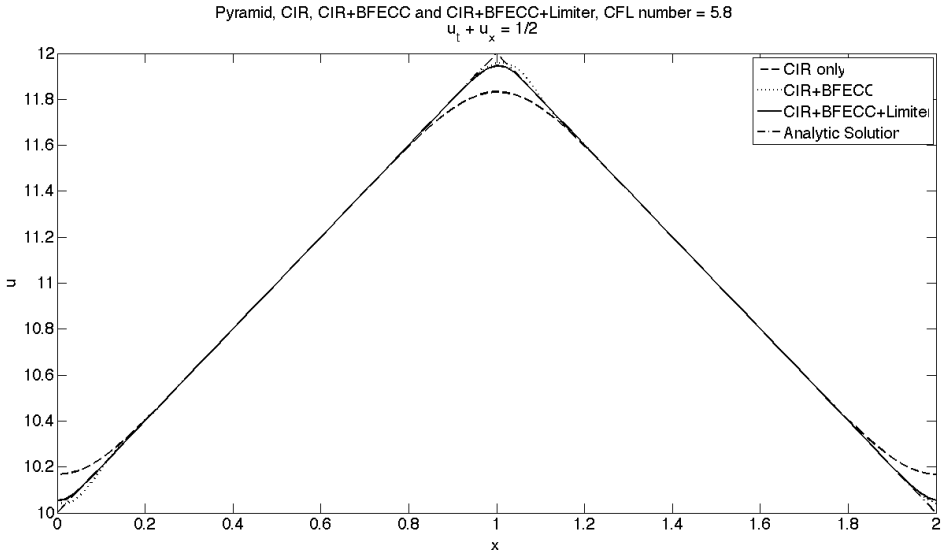

Figure 6: 1-D linear equation with nonzero right-hand side, CFL = 5.8 , pyramid initial condition, $h = 0.02$, $T = 20$.

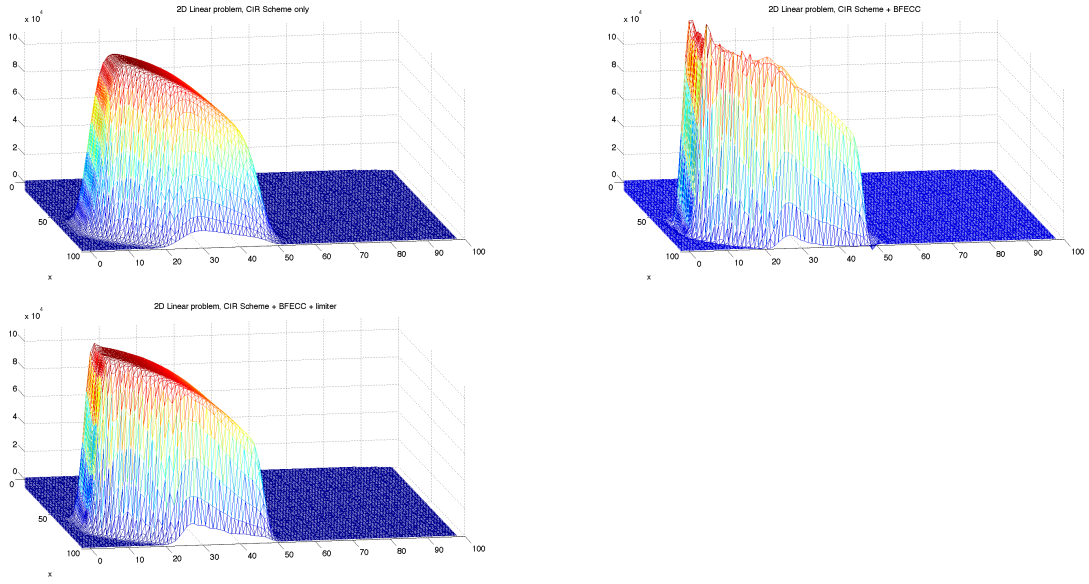

Figure 7: 2-D linear equation at $t = 157$. Uniform triangular mesh with $h = 1.0$ and $CFL = 3.0$. Top left: CIR only; Top right: CIR + BFECC; Bottom: CIR + BFECC + Limiting.

4.4 2D nonlinear Hamilton-Jacobi equations

The following 2D nonlinear Hamilton-Jacobi equation (see e.g. [25]) is commonly used in numerical tests.

$$\begin{aligned} \phi_t + \frac{(\phi_x + \phi_y + 1)^2}{2} &= 0, \qquad (x, y) \in [-2, 2] \times [-2, 2] \\ \phi(x, y, 0) &= -\cos(\frac{\pi(x + y)}{2}) \,. \end{aligned} \tag{16}$$

This equation is computed on a triangular mesh by applying BFECC and limiting to a first order scheme with a Lax-Friedrichs-type monotone Hamiltonian. For more details of the first order scheme with monotone Hamiltonian (L-F for short) developed by Abgrall, see [1]. The results are shown in Figure 8. We can see that BFECC without limiting is adequate for this problem and we find almost no difference when the limiting procedure is turned on.

In addition, we tested the order of accuracy of L-F + BFECC + Limiting. The L^2-accuracy and L^∞-accuracy of this 2D nonlinear Hamilton-Jacobi equation are demonstrated in Table 1, Table 2, Table 3 and Table 4. We can see from the data that the limiting technique improves the accuracy at $T = 0.15$ when singularities of the solution have formed. At $T = 0.015$ when the solution is still smooth, the limiting procedure slightly reduces the L^∞-accuracy of the numerical solution compared to the one without limiting.

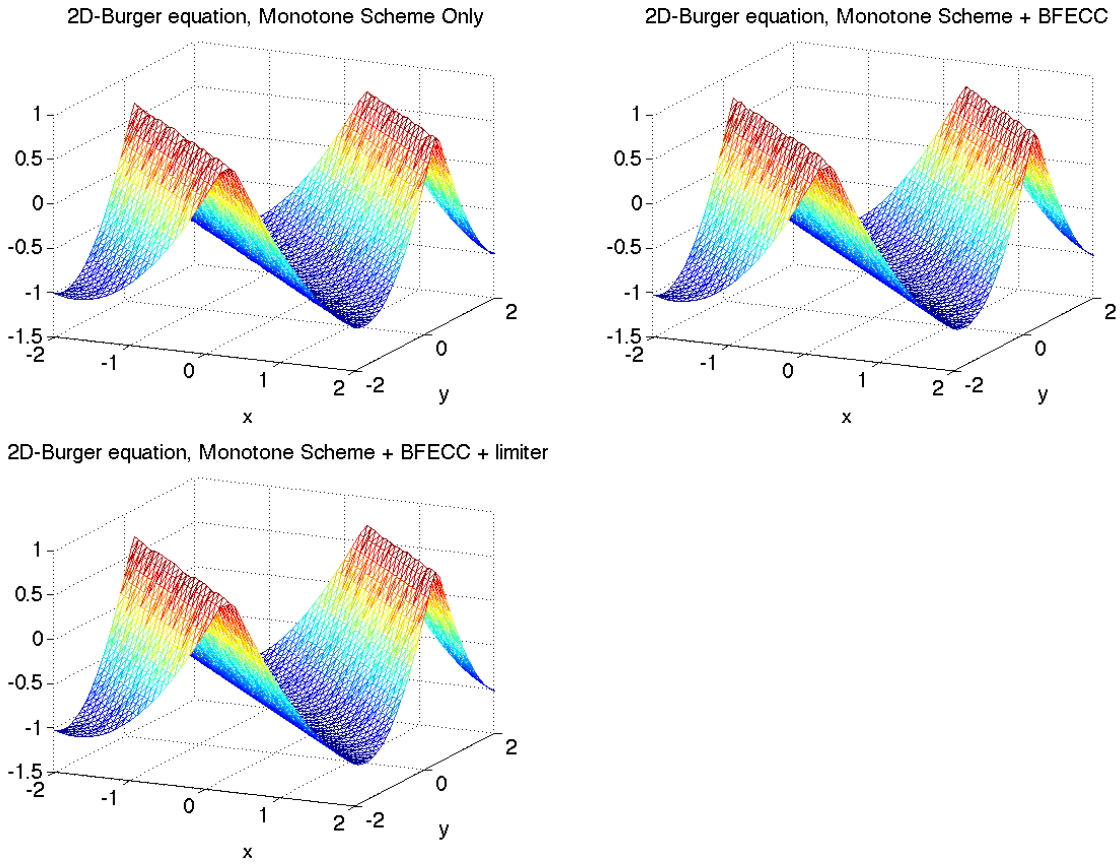

Figure 8: 2-D nonlinear Hamilton-Jacobi equation at $t = 0.15$, Uniform triangular mesh with $h = 0.1$. Top left: L-F only; Top right: L-F + BFECC; Bottom: L-F + BFECC + Limiter.

number of points	L^2 error	order	L^∞ error	order
41×41	0.262	N/A	0.0451	N/A
81×81	0.0550	2.25	0.0135	1.73
161×161	0.0200	1.46	0.00536	1.33

Table 1: Numerical accuracy of L-F + BFECC + limiting. CFL number = 0.1; $T = 0.15$

number of points	L^2 error	order	L^∞ error	order
41×41	0.366	N/A	0.0649	N/A
81×81	0.101	1.86	0.0210	1.62
161×161	0.0370	1.45	0.00797	1.39

Table 2: Numerical accuracy of L-F + BFECC. CFL number = 0.1; $T = 0.15$

number of points	L^2 error	order	L^∞ error	order
41×41	0.289	N/A	0.0157	N/A
81×81	0.0976	1.57	0.00531	1.56
161×161	0.00631	3.95	0.00126	2.08

Table 3: Numerical accuracy of L-F + BFECC+limiting. CFL number = 0.1; $T = 0.015$

4.5 2D-Riemann Problem

We also test our algorithm in the following 2D Riemann Problem (see [25]).

$$\begin{aligned} \phi_t + sin(\phi_x + \phi_y) &= 0 \\ \phi(x, y, 0) &= \pi(|y| - |x|)\,. \end{aligned} \tag{17}$$

We compute the equation on a triangular mesh from $t = 0$ to $t = 1$ by applying BFECC and limiting to the first order scheme with monotone Hamiltonian [1]. The numerical results at time $t = 1$ are shown in Figure 9. Again we observe that BFECC without limiting is adequate for this problem and we find almost no difference when the limiting procedure is turned on.

4.6 Bubble Merging Problem

We study the merging of 4 bubbles (circles) centered at $(40, 60), (60, 60), (40, 40), (60, 40)$ with radius $9, 7, 10, 8$ respectively and expanding with constant normal velocity 0.2. The time evolution of these merging bubbles can be described by the level set method [24] with the level set function ϕ ($\phi < 0$ inside each circle) satisfying the following equation

$$\phi_t + 0.2\frac{\nabla\phi}{|\nabla\phi|} \cdot \nabla\phi = 0 \tag{18}$$

We compute the equation on a triangular mesh by applying BFECC and limiting to the first order scheme with monotone Hamiltonian [1]. The numerical results are compared in Figure 10. The top two bubbles (centered at $(40, 60)$ and $(60, 60)$) should have merged at the time $T = 11$. This is correctly captured with the limiting procedure. With BFECC and no limiting, the merging of the two bubbles has been

number of points	L^2 error	order	L^∞ error	order
41×41	0.288	N/A	0.0153	N/A
81×81	0.0968	1.57	0.00512	1.57
161×161	0.00638	3.92	0.000851	2.59

Table 4: Numerical accuracy of L-F + BFECC. CFL number $= 0.1$; $T = 0.015$

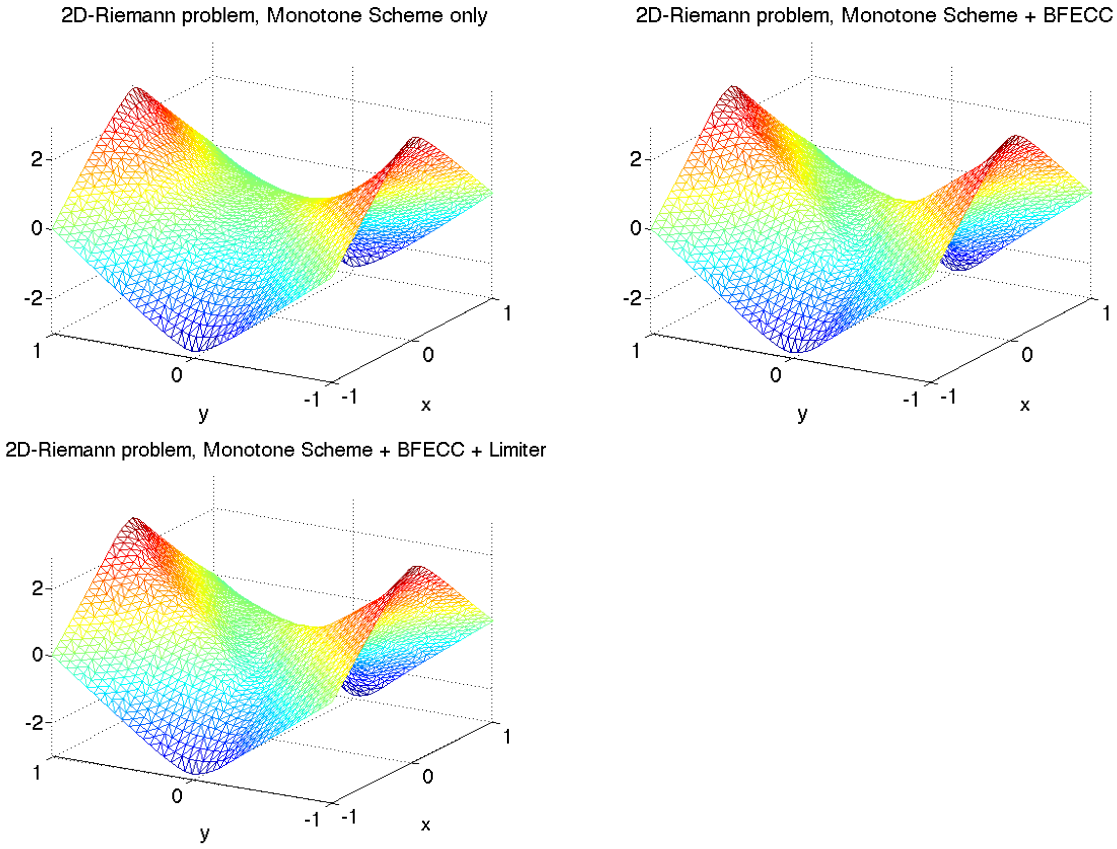

Figure 9: 2-D Riemann problem, Uniform triangular mesh with $h = 0.05$. Top left: L-F only; Top right: L-F + BFECC; Bottom: L-F + BFECC + Limiter.

delayed at $T = 11$. At the time $T = 26$, we can see that the smallest drop in the graph (bottom right) is kept when BFECC is used with the limiting, almost as well as without the limiting procedure (bottom middle graph).

4.7 Shrinking Square Problem

We consider the following problem: A square centered at $(0, 0)$ with side length 10 shrinks with the normal speed 0.2. Therefore at time $T = 10$ one should expect a 6×6 square. This problem can be described by equation (19), which is the same as

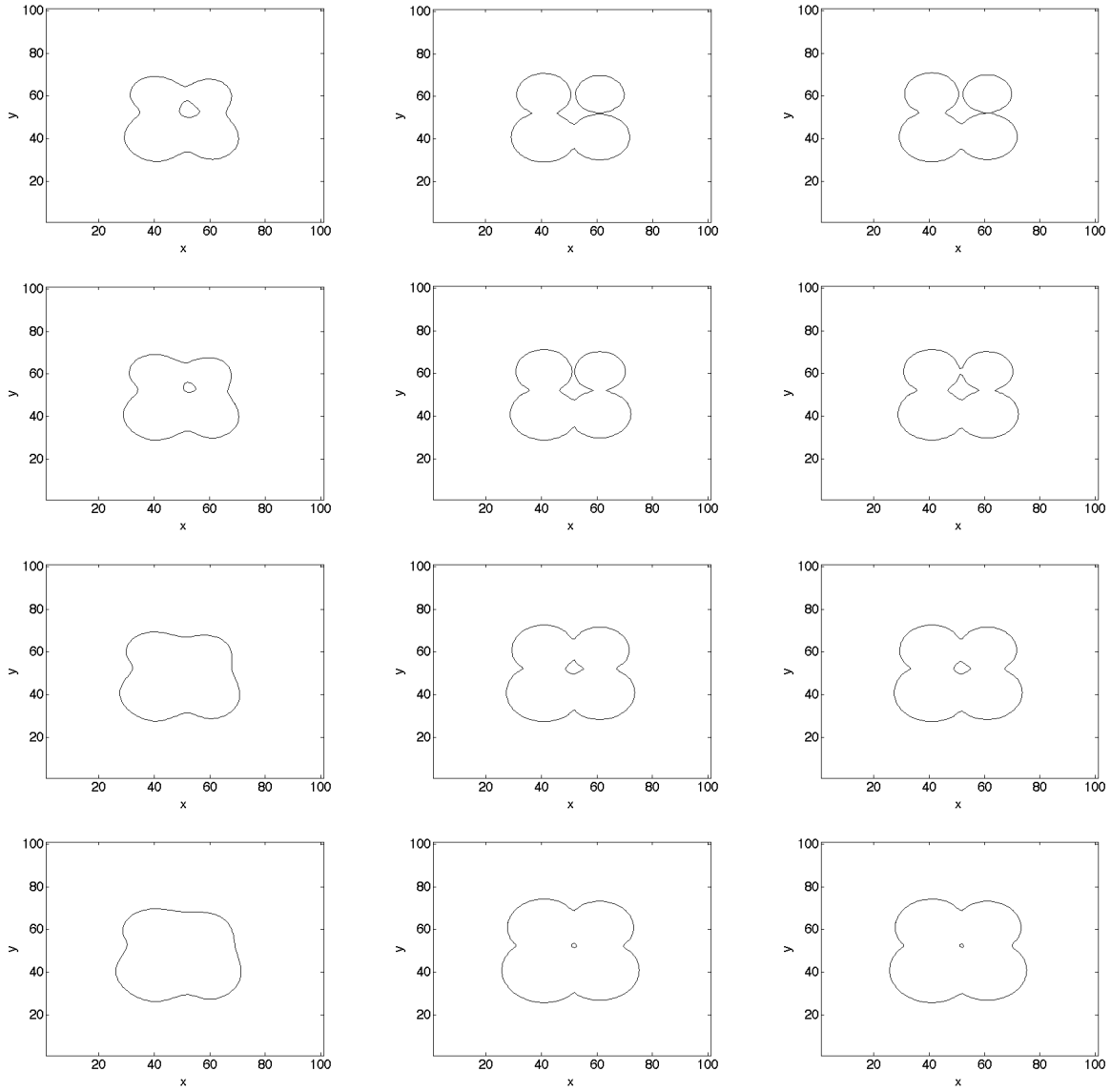

Figure 10: Expanding circles of radii $9, 7, 10$ and 8 with normal velocity 0.2. Uniform triangular mesh with $h = 1$, $\Delta t = 0.4$. Left: Lax-Friedrichs-type monotone Hamiltonian scheme (L-F); Middle: L-F + BFECC; Right: L-F + BFECC + Limiter. T = 9, 11, 18, 26 from top to bottom.

the bubble merging problem from the previous subsection. The initial condition is an indicator function of $[-5,5]\times[-5,5]\subseteq\mathbb{R}^2$.

$$\phi_t + 0.2\frac{\nabla\phi}{|\nabla\phi|}\cdot\nabla\phi = 0 \tag{19}$$

$$\begin{aligned}\phi(x,y,0) &= 1, \quad (x,y)\in[-5.5]\times[-5,5]\\ \phi(x,y,0) &= 0, \quad (x,y)\notin[-5,5]\times[-5,5]\end{aligned} \tag{20}$$

Our computations are based on the same first order Lax-Friderichs-type scheme with monotone Hamiltonian (L-F for short) as in Section 4.4. Equation (19) is computed on the triangular mesh. Numerical results obtained from three schemes, L-F scheme only, L-F scheme + BFECC and L-F scheme + BFECC + limiting, are compared in Figure 11. It can be observed that the L-F scheme leads to significant numerical diffusion. Such numerical diffusion can be reduced by BFECC. However, strong undershoots are generated by BFECC for this problem. With the help of the limiting algorithm, higher accuracy of BFECC can be preserved, while numerical artifacts are essentially eliminated.

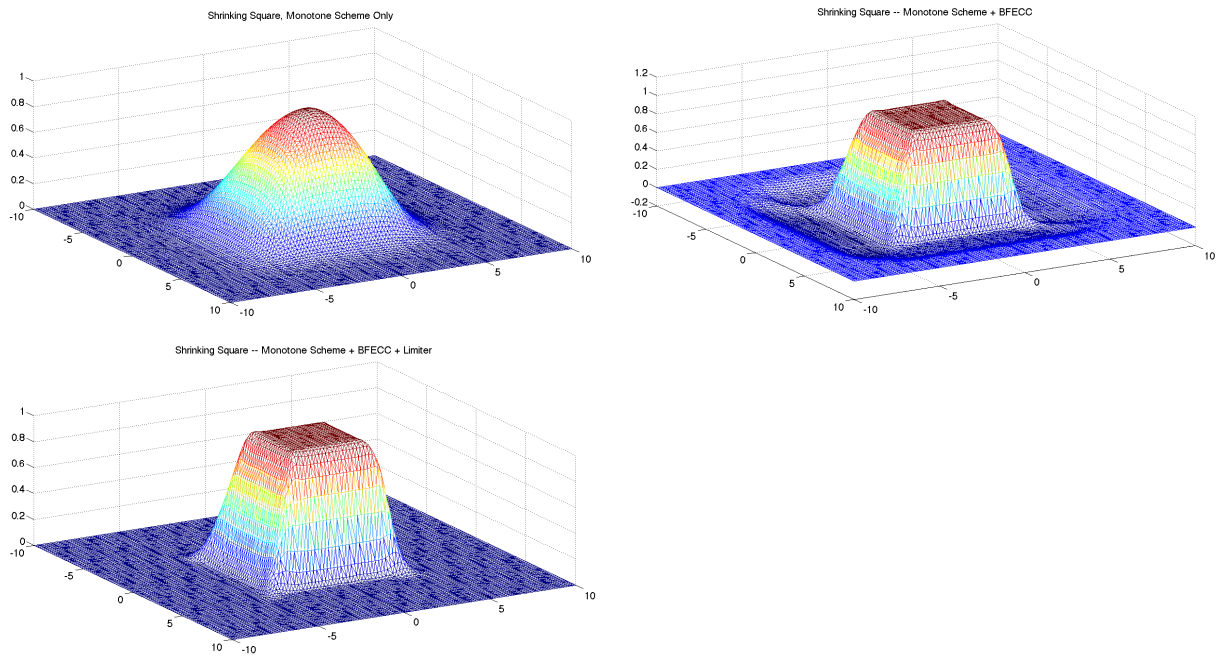

Figure 11: Shrinking Square Problem, Uniform triangular mesh with 101×101 mesh points. Top left: Monotone scheme (L-F) only; Top right: L-F + BFECC; Bottom: L-F + BFECC + Limiting.

5 Acknowledgment

The authors thank Andrea Bertozzi, Chi-Wang Shu and Yongtao Zhang for helpful discussions related to this work.

References

[1] R. Abgrall, Numerical discretization of the rst-order Hamilton-Jacobi equation on triangular meshes, *Comm. Pure Appl. Math.*, 49 (1996), 13391373.

[2] J. P. Boris and D. L. Book, Flux-Corrected Transport I. SHASTA, A Fluid Transport Algorithm That Works, *J. Comput. Phys.*, 11 (1973), 38–69.

[3] S. Bryson, A. Kurganov, D. Levy and G. Petrova, Semi-Discrete Central-Upwind Schemes with Reduced Dissipation for Hamilton-Jacobi Equations, *IMA J. Numer. Anal.*, 25 (2005), 113–138.

[4] S. Bryson and D. Levy, High-Order Semi-Discrete Central-Upwind Schemes for Multi-Dimensional Hamilton-Jacobi Equations, *J. Comp. Phys.*, 189 (2003), 63–87.

[5] Y. Chen, Q. Kang, Q. Cai and D. Zhang, Lattice Boltzmann method on quadtree grids, *Physical Review E*, 83 (2011).

[6] R. Courant and E. Isaacson and M. Rees, On the solution of nonlinear hyperbolic differential equations by finite differences, *Comm. Pure Appl. Math.*, 5 (1952), 243–255.

[7] D. Enright, F. Losasso and R. Fedkiw, A Fast and Accurate Semi-Lagrangian Particle Level Set Method, *Computers and Structures*, 83 (2005), 479–490.

[8] M. Lentine, J. Gretarsson and R. Fedkiw, An Unconditionally Stable Fully Conservative Semi-Lagrangian Method, *J. Comp. Phys.* 230 (2011), 2857–2879.

[9] C.-T. Lin and E. Tadmor, High-resolution non-oscillatory central schemes for Hamilton- Jacobi Equations, *SIAM J. Sci. Comput.*, 21 (2000), 2163–2186.

[10] T. F. Dupont and Y.-J. Liu, Back and Forth Error Compensation and Correction Methods for Removing Errors Induced by Uneven Gradients of The Level Set Function, *J. Comput. Phys.*, 190 (2003), 311–324.

[11] T. F. Dupont and Y.-J. Liu, Back and Forth Error Compensation and Correction Methods for Semi-Lagrangian Schemes with Application to Interface Computation Using Level Set Method, *Tech. Rep. CDSNS2003-391*, School of Math., Georgia Inst. of Tech., 2004.

[12] T. F. Dupont and Y.-J. Liu, Back and Forth Error Compensation and Correction Methods for Semi-Lagrangian Schemes with Application to Level Set Interface Computations, *Math. Comp.*, 76 (2007), 647–668.

[13] I.V. Gugushvili and N. M. Evstigneev, Semi-Lagrangian Method for Advection Equation on GPU in Unstructured R^3 Mesh for Fluid Dynamics Application, *World Academy of Science, Engineering and Technology*, 60 (2009).

[14] A. Harten, B. Engquist, S. Osher and S. Chakravarthy, Uniformly High Order Accurate Essentially Non-oscillatory Schemes, III, *J. Comput. Phys.*, 71 (1987), 231–303.

[15] B.-M. Kim, Y.-J. Liu, I. Llamas and J. Rossignac, FlowFixer: Using BFECC for Fluid Simulation, *Eurographics Workshop on Natural Phenomena*, 2005.

[16] B.-M. Kim, Y.-J. Liu, I. Llamas and J. Rossignac, Advections with Significantly Reduced Dissipation and Diffusion, *IEEE Trans. Visual. and Comput. Graph.*, 13 (2007), 135–144.

[17] B.-M. Kim, Y.-J. Liu, I. Llamas, X.-M. Jiao and J. Rossignac, Simulation of Bubbles in Foam by Volume Control, *ACM SIGGRAPH* 2007.

[18] T. Kim and M. Carlson, A simple boiling module, *Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation*, 27–34.

[19] A. Kurganov, S. Noelle and G. Petrova, Semi-Discrete Central-Upwind Schemes for Hyperbolic Conservation Laws and Hamilton-Jacobi Equations, *SIAM J. Sci. Comput.*, 23 (2001), 707–740.

[20] A. Kurganov and G. Petrova, Adaptive Central-Upwind Schemes for Hamilton-Jacobi Equations with Nonconvex Hamiltonians, *J. Sci. Comput.*, 27 (2006), 323–333.

[21] A. Kurganov and E. Tadmor New High-Resolution Semi-Discrete Central Schemes for Hamilton-Jacobi Equations *J. Comput. Phys.*, 160 (2000), 720–742.

[22] P. D. Lax, On the Stability of Difference Approximations to Solutions of Hyperbolic Equations with Variable Coefficients, *Comm. Pure Appl. Math.*, 14 (1961), 497–520.

[23] B. van Leer, Toward the ultimate conservative dierence scheme: II. Monotonicity and conservation combined in a second order scheme, *J. Comput. Phys.*, 14 (1974), 361–370.

[24] S. Osher and J. Sethian, Fronts Propagating with Curvature-Dependent Speed: Algorithms based on Hamilton-Jacobi Equations, *J. Comput. Phys.*, 79 (1988), 12–49.

[25] S. Osher and C.-W. Shu, High-order essentially nonoscillatory schemes for HamiltonJacobi equations, *SIAM J. Numer. Anal.*, 28 (1991), 907922.

[26] A. Selle, R. Fedkiw, B.-M. Kim, Y.-J. Liu, and J. Rossignac, An Unconditionally Stable MacCormack Method, *J. Sci. Comput.*, 35 (2008), 350–371.

[27] C.-W. Shu and S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, *J. Comput. Phys.*, 77 (1988), 439–471.

[28] J. Strain, Semi-Lagrangian methods for level set equations, *J. Comput. Phys.*, 151 (1999), 498–533.

[29] S. T. Zalesak, Fully Multidimensional Flux-Corrected Transport, *J. Comput. Phys.*, 31 (1979), 335–362.

[30] Y.-T. Zhang and C.-W. Shu, High order WENO schemes for Hamilton-Jacobi equations on triangular meshes, *SIAM J. Sci. Comput.*, 24 (2003), 1005–1030.