以繪圖硬體實做降低三維震波模擬之耗散計算

劉啟東
朝陽科技大學資訊工程系
cdliou@mail.cyut.edu.tw

余啟哲
交通大學應用數學系
ycj@math.nctu.edu.tw

摘要

本論文推導並改善 Kurganov algorithm 中處理 2D 抗耗散項的方法且推廣到 3D，並在多 GPU 上實做。

關鍵詞：GPGPU, Euler equation, Kurganov algorithm

Abstract

In this paper, we will extend the modification for the 2-D anti-dissopation algorithm of Kurganov-type of finite volume method in the shock waves simulation to 3-D and implement it on GPU

Keywords: GPGPU, Euler equation, Kurganov algorithm.

1.前言

本論文主要是模擬以理想氣體奠基於質量，動量，和能量守恆，由尤拉方程(Euler equation)的雙曲系統所描述的動力系統。在過去的三，四十年，研究人員已經在這一領域提出了許多方法而有了長足的進步 [8]。中央迎風（central-upwind -- Riemann-problem-solver-free and central Godunov-type projection-evolution）方法[1]-[7], [9]-**錯誤！找不到參照來源。**以繞過解決 Riemann 問題為主要特色，提供令人激賞的進步，並因此簡化了複雜和大量的計算。中央迎風框架也顯著減少出現在 staggered center scheme 中的數值耗散問題。研究人員藉由更精確的估計 Riemann fan (由初始狀態因不連續而散開的區域) 的寬度而逐步改善計算，並以更陡的斜率達到更高階的內差來重建區域分佈。最後，半離散的配置使得擴展到多維度更方便。如[1]中所述，這一系列的演算法得到分辨率高，簡單性，普遍性，和穩定的優點。

在此應用中，我們嘗試了幾種不同的組態後發現，抗耗散項 (anti-dissipation terms)（4）是改善不連續點附近計算精度的關鍵。鑑於降低耗散是不容易在短模擬時間內或在低解析度能觀察到，我們在 GPU 上實做模擬程式並且達到在 13.5 小時內完成一次模擬，使我們可以觀察抗耗散機制的最終影響。

在解決 1D 的尤拉方程，原型[1]保持盡可能窄的不連續性。然而在模擬二維 Euler 方程時它顯現了疲軟的跡象，這是由於公式（4）過度使用了一些斜率限制器所造成的，其原來的目的是在滿足 Total-Variation-Diminishing (TVD）的條件之下，限制區域之間介面的斜率，以避免振盪。在我們之前研究當中已經成功的推導出二維最佳抗耗散項的解[10]。

然而，僅在二維模擬並無法解決三維的問題，在此論文中我們將詳述推導出 3 維中處理 anti-dissipation terms 的方法。

2.數值演算法

三維尤拉方程可寫成

$$\mathbf{U}_t + f(\mathbf{U})_x + g(\mathbf{U})_y + h(\mathbf{U})_z = s(\mathbf{U}). \quad (1)$$

Where,

$$\mathbf{U}(x,y,z,t) = (\rho, \rho u, \rho v, \rho w, E),$$
$$f(\mathbf{U}) = f(\rho, \rho u, \rho v, \rho w, E) = \left(\rho u, \rho uvw, \rho u^2 + P, u(E+P)\right),$$
$$g(\mathbf{U}) = \left(\rho v, \rho uvw, \rho v^2 + P, v(E+P)\right),$$
$$h(\mathbf{U}) = \left(\rho w, \rho uvw, \rho w^2 + P, w(E+P)\right).$$

其中 $\rho(x,y,z,t)$ 為密度，$\left(u(x,y,z,t), v(x,y,z,t), w(x,y,z,t)\right)$ 為速度，$E(x,y,z,t)$ 為總能量，$P(x,y,z,t)$ 為壓力。它們的關係為 $E = \dfrac{P}{\gamma - 1} + \dfrac{1}{2}P(u^2+v^2+w^2)$；$\gamma$ 為常數與氣體種類有關，$s(\mathbf{U})$ 為來源項。

半離散型的 Kurganov scheme [1] 是整合下列的常微分方程 (O.D.E.):

$$\frac{d}{dt}\mathbf{U}_{ijk}(t) = -\frac{\mathbf{F}_{i,j+1/2,k}(t)-\mathbf{F}_{i,j-1/2,k}(t)}{\Delta x} - \frac{\mathbf{G}_{i+1/2,j,k}(t)-\mathbf{G}_{i-1/2,j,k}(t)}{\Delta y} - \frac{\mathbf{H}_{i,j,k+1/2}(t)-\mathbf{H}_{i,j,k-1/2}(t)}{\Delta z} + s\left(\mathbf{U}_{ijk}(t)\right), \tag{2}$$

$$\mathbf{F}_{i,j+1/2,k}(t) = \frac{a^{+}_{i,j+1/2,k} f\left(\mathbf{U}^{\mathrm{E}}_{i,j,k}\right) - a^{-}_{i,j+1/2,k} f\left(\mathbf{U}^{\mathrm{W}}_{i,j+1,k}\right)}{a^{+}_{i,j+1/2,k} - a^{-}_{i,j+1/2,k}} + a^{+}_{i,j+1/2,k}\cdot a^{-}_{i,j+1/2,k}\left[\frac{\mathbf{U}^{\mathrm{W}}_{i,j+1,k}-\mathbf{U}^{\mathrm{E}}_{i,j,k}}{a^{+}_{i,j+1/2,k}-a^{-}_{i,j+1/2,k}} - \mathbf{Q}^{x}_{i,j+1/2,k}\right],$$

$$\mathbf{G}_{i+1/2,j,k}(t) = \frac{b^{+}_{i+1/2,j,k} g\left(\mathbf{U}^{\mathrm{N}}_{i,j,k}\right) - b^{-}_{i+1/2,j,k} g\left(\mathbf{U}^{\mathrm{S}}_{i+1,j,k}\right)}{b^{+}_{i+1/2,j,k} - b^{-}_{i+1/2,j,k}} + b^{+}_{i+1/2,j,k}\cdot b^{-}_{i+1/2,j,k}\left[\frac{\mathbf{U}^{\mathrm{S}}_{i+1,j,k}-\mathbf{U}^{\mathrm{N}}_{i,j,k}}{b^{+}_{i+1/2,j,k}-b^{-}_{i+1/2,j,k}} - \mathbf{Q}^{y}_{i+1/2,j,k}\right],$$

$$\mathbf{H}_{i,j,k+1/2}(t) = \frac{c^{+}_{i,j,k+1/2} h\left(\mathbf{U}^{\mathrm{U}}_{i,j,k}\right) - c^{-}_{i,j,k+1/2} h\left(\mathbf{U}^{\mathrm{D}}_{i,j,k+1}\right)}{c^{+}_{i,j,k+1/2} - c^{-}_{i,j,k+1/2}} + c^{+}_{i,j,k+1/2}\cdot c^{-}_{i,j,k+1/2}\left[\frac{\mathbf{U}^{\mathrm{D}}_{i,j,k+1}-\mathbf{U}^{\mathrm{U}}_{i,j,k}}{c^{+}_{i,j,k+1/2}-c^{-}_{i,j,k+1/2}} - \mathbf{Q}^{z}_{i,j,k+1/2}\right].$$

$$a^{+}_{i,j+1/2,k} = \max\left(0, \lambda^{+}_{x}\left(\mathbf{U}^{\mathrm{W}}_{i,j+1,k}\right), \lambda^{+}_{x}\left(\mathbf{U}^{\mathrm{E}}_{i,j,k}\right)\right), \quad a^{-}_{i,j+1/2,k} = \min\left(0, \lambda^{-}_{x}\left(\mathbf{U}^{\mathrm{W}}_{i,j+1,k}\right), \lambda^{-}_{x}\left(\mathbf{U}^{\mathrm{E}}_{i,j,k}\right)\right),$$

$$b^{+}_{i+1/2,j,k} = \max\left(0, \lambda^{+}_{y}\left(\mathbf{U}^{\mathrm{S}}_{i+1,j,k}\right), \lambda^{+}_{y}\left(\mathbf{U}^{\mathrm{N}}_{i,j,k}\right)\right), \quad b^{-}_{i+1/2,j,k} = \min\left(0, \lambda^{-}_{y}\left(\mathbf{U}^{\mathrm{S}}_{i+1,j,k}\right), \lambda^{-}_{y}\left(\mathbf{U}^{\mathrm{N}}_{i,j,k}\right)\right),$$

$$c^{+}_{i,j,k+1/2} = \max\left(0, \lambda^{+}_{z}\left(\mathbf{U}^{\mathrm{D}}_{i,j,k+1}\right), \lambda^{+}_{z}\left(\mathbf{U}^{\mathrm{U}}_{i,j,k}\right)\right), \quad c^{-}_{i,j,k+1/2} = \min\left(0, \lambda^{-}_{z}\left(\mathbf{U}^{\mathrm{D}}_{i,j,k+1}\right), \lambda^{-}_{z}\left(\mathbf{U}^{\mathrm{U}}_{i,j,k}\right)\right).$$

$\lambda^{\pm}_{x}(\mathbf{U}) = u \pm \sqrt{\gamma P/\rho}$ 是最慢與最快 $\frac{\partial f}{\partial \mathbf{U}}$ 的特徵值,

$\lambda^{\pm}_{y}(\mathbf{U}) = v \pm \sqrt{\gamma P/\rho}$ 是最慢與最快 $\frac{\partial g}{\partial \mathbf{U}}$ 的特徵值,

$\lambda^{\pm}_{z}(\mathbf{U}) = w \pm \sqrt{\gamma P/\rho}$ 是最慢與最快 $\frac{\partial h}{\partial \mathbf{U}}$ 的特徵值.

$$\mathbf{U}^{\mathrm{E(W)}}_{i,j,k} = p_{i,j,k}\left(x_j \pm \frac{\Delta x}{2}, y_i, z_k\right),$$

$$\mathbf{U}^{\mathrm{N(S)}}_{i,j,k} = p_{i,j,k}\left(x_j, y_i \pm \frac{\Delta y}{2}, z_k\right),$$ ，其中

$$\mathbf{U}^{\mathrm{U(D)}}_{i,j,k} = p_{i,j,k}\left(x_j, y_i, z_k \pm \frac{\Delta z}{2}\right)$$

$p_{i,j,k}(x,y,z)$ 是定義在區間 $\left[x_j - \frac{\Delta x}{2}, x_j + \frac{\Delta x}{2}\right] \times \left[y_i - \frac{\Delta y}{2}, y_i + \frac{\Delta y}{2}\right] \times \left[z_k - \frac{\Delta z}{2}, z_k + \frac{\Delta z}{2}\right]$ 中的線性內差 which approximates to reconstruct $\mathbf{U}(x,y,z,t_n)$ in a neighborhood of (x_j, y_i, z_k) at time t_n.

$$\mathbf{Q}^{x}_{i,j+1/2,k} = \frac{\operatorname{minmod}\left(\mathbf{U}^{\mathrm{W}}_{i,j+1,k} - w^{\mathrm{int}}_{i,j+1/2,k}, w^{\mathrm{int}}_{i,j+1/2,k} - \mathbf{U}^{\mathrm{E}}_{i,j,k}, \frac{\mathbf{U}^{\mathrm{UNW}}_{i,j+1,k}-\mathbf{U}^{\mathrm{UNE}}_{i,j,k}}{2}, \frac{\mathbf{U}^{\mathrm{USW}}_{i,j+1,k}-\mathbf{U}^{\mathrm{USE}}_{i,j,k}}{2}, \frac{\mathbf{U}^{\mathrm{DNW}}_{i,j+1,k}-\mathbf{U}^{\mathrm{DNE}}_{i,j,k}}{2}, \frac{\mathbf{U}^{\mathrm{DSW}}_{i,j+1,k}-\mathbf{U}^{\mathrm{DSE}}_{i,j,k}}{2}\right)}{a^{+}_{i,j+1/2,k} - a^{-}_{i,j+1/2,k}},$$

$$\mathbf{Q}^{y}_{i+1/2,j,k} = \frac{\operatorname{minmod}\left(\mathbf{U}^{\mathrm{S}}_{i+1,j,k} - w^{\mathrm{int}}_{i+1/2,j,k}, w^{\mathrm{int}}_{i+1/2,j,k} - \mathbf{U}^{\mathrm{N}}_{i,j,k}, \frac{\mathbf{U}^{\mathrm{USW}}_{i+1,j,k}-\mathbf{U}^{\mathrm{UNW}}_{i,j,k}}{2}, \frac{\mathbf{U}^{\mathrm{USE}}_{i+1,j,k}-\mathbf{U}^{\mathrm{UNE}}_{i,j,k}}{2}, \frac{\mathbf{U}^{\mathrm{DSW}}_{i+1,j,k}-\mathbf{U}^{\mathrm{DNW}}_{i,j,k}}{2}, \frac{\mathbf{U}^{\mathrm{DSE}}_{i+1,j,k}-\mathbf{U}^{\mathrm{DNE}}_{i,j,k}}{2}\right)}{b^{+}_{i+1/2,j,k} - b^{-}_{i+1/2,j,k}},$$

$$\mathbf{Q}^{z}_{i,j,k+1/2} = \frac{\operatorname{minmod}\left(\mathbf{U}^{\mathrm{D}}_{i,j,k+1} - w^{\mathrm{int}}_{i,j,k+1/2}, w^{\mathrm{int}}_{i,j,k+1/2} - \mathbf{U}^{\mathrm{U}}_{i,j,k}, \frac{\mathbf{U}^{\mathrm{DSW}}_{i,j,k+1}-\mathbf{U}^{\mathrm{USW}}_{i,j,k}}{2}, \frac{\mathbf{U}^{\mathrm{DNW}}_{i,j,k+1}-\mathbf{U}^{\mathrm{UNW}}_{i,j,k}}{2}, \frac{\mathbf{U}^{\mathrm{DSE}}_{i,j,k+1}-\mathbf{U}^{\mathrm{USE}}_{i,j,k}}{2}, \frac{\mathbf{U}^{\mathrm{DNE}}_{i,j,k+1}-\mathbf{U}^{\mathrm{UNE}}_{i,j,k}}{2}\right)}{c^{+}_{i,j,k+1/2} - c^{-}_{i,j,k+1/2}}.$$

$$\operatorname{minmod}(c_1,\cdots,c_m) = \begin{cases} \min(c_1,\cdots,c_m), & \text{if } c_i > 0 \ \forall i = 1,\cdots,m, \\ \max(c_1,\cdots,c_m), & \text{if } c_i < 0 \ \forall i = 1,\cdots,m, \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

$\mathbf{Q}^{x}_{i,j+1/2,k}$，$\mathbf{Q}^{y}_{i+1/2,j,k}$ and $\mathbf{Q}^{z}_{i,j,k+1/2}$ 為抗耗散項用來降低數值計算的耗散 (numerical dissipation.)

$$\mathbf{U}^{\mathrm{E(W)N(S)U(D)}}_{i,j,k} = p_{i,j,k}\left(x_j \pm \frac{\Delta x}{2}, y_i \pm \frac{\Delta y}{2}, z_k \pm \frac{\Delta z}{2}\right),$$

$$w^{\mathrm{int}}_{i,j+1/2,k} = \frac{a^{+}_{i,j+1/2,k}\mathbf{U}^{\mathrm{W}}_{i,j+1,k} - a^{-}_{i,j+1/2,k}\mathbf{U}^{\mathrm{E}}_{i,j,k} + f\left(\mathbf{U}^{\mathrm{W}}_{i,j+1,k}\right) - f\left(\mathbf{U}^{\mathrm{E}}_{i,j,k}\right)}{a^{+}_{i,j+1/2,k} - a^{-}_{i,j+1/2,k}},$$

$$w^{\mathrm{int}}_{i+1/2,j,k} = \frac{b^{+}_{i+1/2,j,k}\mathbf{U}^{\mathrm{S}}_{i+1,j,k} - b^{-}_{i+1/2,j,k}\mathbf{U}^{\mathrm{N}}_{i,j,k} + g\left(\mathbf{U}^{\mathrm{S}}_{i+1,j,k}\right) - g\left(\mathbf{U}^{\mathrm{N}}_{i,j,k}\right)}{b^{+}_{i+1/2,j,k} - b^{-}_{i+1/2,j,k}},$$

$$w^{\mathrm{int}}_{i,j,k+1/2} = \frac{c^{+}_{i,j,k+1/2}\mathbf{U}^{\mathrm{D}}_{i,j,k+1} - c^{-}_{i,j,k+1/2}\mathbf{U}^{\mathrm{U}}_{i,j,k} + h\left(\mathbf{U}^{\mathrm{D}}_{i,j,k+1}\right) - h\left(\mathbf{U}^{\mathrm{U}}_{i,j,k}\right)}{c^{+}_{i,j,k+1/2} - c^{-}_{i,j,k+1/2}}.$$

要積分 ODE (2), 每個時間步都由 CFL-condition 所限制，亦即,

$$t_{n+1} - t_n = \Delta t \le CFL \cdot \min_{i,j,k}\left(\frac{\Delta x}{\max\left(a^{+}_{j+1/2}, -a^{-}_{j+1/2}\right)}, \frac{\Delta y}{\max\left(b^{+}_{i+1/2}, -b^{-}_{i+1/2}\right)}, \frac{\Delta z}{\max\left(c^{+}_{k+1/2}, -c^{-}_{k+1/2}\right)}\right)$$

(3) 與其他演算法[2][3][6][7]的主要差異為加入額外的抗耗散項 (4) 與推廣到三維。這些項可增進在不連續點附近的計算精確度。

3. GPU 實做

我們用 CUDA 在 3 個 GPU 上實做整個系

統，為了效率起見，我們切割 z 軸方向為三塊分配給三顆 GPU，亦即同一個 xy 平面一定落在同一顆 GPU。由於架構上的限制，每個運算區塊大小為 16x8x4，運算時需邊界多 2 個元素如圖 1。而在交界的地方則必須相鄰兩顆 GPU 交換資料，交換的資料量為兩片 xy 平面大小。演算法如下

```
while(Time_Remain > 0) {
    do {
        integrate result;
        wait for all GPU;
        calculate vector field;
        wait for all GPU;
    } repeat until all component done;
}
```

內迴圈計算單一時間步的各項係數，每算完一步，所有 GPU 都必須同步才能開始下一步。

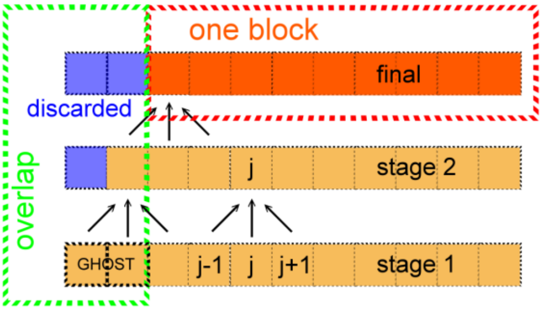

圖 1. 重疊區的使用.

4. 實驗結果

我們推導出改善 Kurganov algorithm 中處理 3D 抗耗散項的方法並在 GPU 上實做。

定義域: $0 \le x \le \frac{1}{4}, 0 \le y \le 1$

邊界條件:

$x \le 0$ 及 $x \ge 1/4$ 皆為反射邊.

底邊 $y \le 0$ 固定為 post-shock state.

上邊 $y \ge 1$, 固定為初始 pre-shock state.

狀態方程式: $\gamma = \frac{5}{3}$

內部網格點初始設定:

$$y < \frac{1}{2}, \text{ post-shock state: } \begin{pmatrix} \rho \\ u \\ v \\ P \end{pmatrix} = \begin{pmatrix} 2 \\ 0 \\ -0.025 \cdot c_0 \cdot \cos(8\pi x) \\ 2y+1 \end{pmatrix},$$

$$c_0 = \sqrt{\frac{\gamma P}{\rho}},$$

$$y \ge \frac{1}{2}, \text{ pre-shock state: } \begin{pmatrix} \rho \\ u \\ v \\ P \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ -0.025 \cdot c_0 \cdot \cos(8\pi x) \\ y+1.5 \end{pmatrix}$$

CFL number: 0.475

Running time: $t = 1.95$

實驗結果顯示我們已經完成的三維爆震波模擬。圖 2、3 為極座標模式，圖 4 為笛卡爾座標模式。

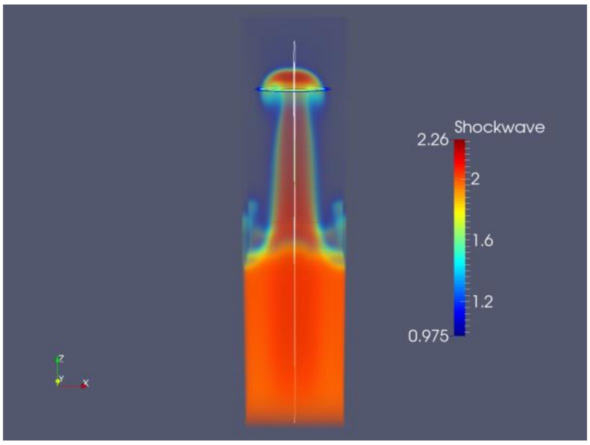

圖 2. 96x96x386 模擬成果及切面所在位置

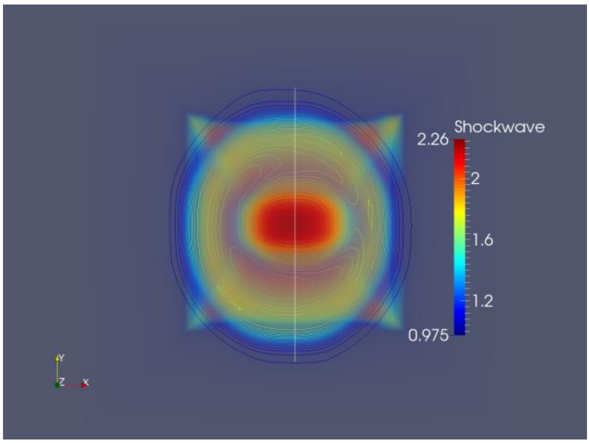

圖 3.橫切面.

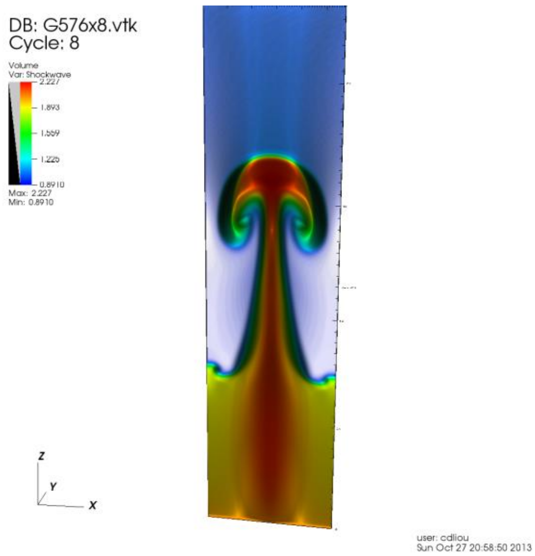

圖 4. 576x8x2304 薄片

5. 結論

我們推導並改善 Kurganov algorithm 中處理 2D 抗耗散項的方法且推廣到 3D，並在多 GPU 上實做。由於繪圖卡記憶體的限制，我們僅能做到 96 ×96×386 或 576×8×2304，雖然如此，這仍然遠高於文獻紀錄。

參考文獻

[1]. A. Kurganov and C.-T. Lin, *On the reduction of numerical dissipation in central-upwind schemes*, Communications in Computational Physics 2 (2007), 141-163.

[2]. A. Kurganov and E. Tadmor, New high-resolution central schemes for nonlinear conservation laws and convection–diffusion equations, Computational Physics 160 (2000), 241–282.

[3]. A. Kurganov and E. Tadmor, Solution of two-dimensional Riemann problems for gas dynamics without Riemann problem solvers. Part. Diff. Eq., 18 (2002), 584-608.

[4]. A. Kurganov and D. Levy, Third-order semi-discrete central scheme for conservation laws and convection-diffusion equations, SIAM J. Sci. Comput. 22 (2000), 1461-1488.

[5]. A. Kurganov and G. Petrova, *Central schemes and contact discontinuities*, Numer. Anal., 34 (2000), 1259-1275.

[6]. A. Kurganov and G. Petrova, A third-order semi-discrete genuinely multidimensional central scheme for hyperbolic conservation laws and related problems, Numer. Math. 88 (2001), 683-729.

[7]. A. Kurganov, S. Noelle and G. Petrova, Semi-discrete central-upwind scheme for hyperbolic conservation laws and Hamilton-Jacobi equations, SIAM J. Sci. Comput., 23 (2001), 707-740.

[8]. Bram van Leer, Upwind and high-resolution methods for compressible flow: from donor cell to residual-distribution schemes, Communication in Computational Physics, Vol. 1, No. 2 (2006), 192-206

[9]. D. Levy, G. Puppo and G. Russo, *Compact central WENO schemes for multidimensional conservation laws*, SIAM J. Sci. Comput. 22 (2000), 656-672.

[10]. Chi-Jer Yu and Chii-Tung Liu, *Modifying and Reducing Numerical Dissipation in a Two-Dimensional Central-Upwind Scheme*, Advances in Applied Mathematics and Mechanics, June 2012, pp 340-353.