A Smoothness/Shock Indicator for the RKDG Methods

David Rumsey and Tong Sun

Department of Mathematics and Statistics
Bowling Green State University
Bowling Green, OH 43403

Abstract. A smoothness/shock indicator is proposed for the RKDG methods solving nonlinear conservation laws. A few numerical experiments are presented as evidence that the indicator helps in detecting shocks, high order discontinuities, regions of smooth solutions, and numerical "instability".

keywords. Conservation law, discontinuous Galerkin method, smoothness indicator.
AMS subject class. Primary: 65M15 Secondary: 65L20

1. Introduction

In the numerical computation of hyperbolic conservation laws, the importance of a smoothness indicator is well known. Some schemes are built-in with smoothness indicators, such as the WENO schemes ([5] and references there-in). Other smoothness indicators are designed independently from specific schemes, by using B-splines and local truncation error estimates ([2] and references there-in).

In this paper, we propose a smoothness/shock indicator for the discontinuous Galerkin schemes. This indicator only depends on the semi-discrete DG scheme. It can be generalized to 2D triangle meshes *verbatim.* The cost of computing the indicator is comparable to that of implementing an explicit Runge-Kutta scheme. We will use a few numerical experiments to show what information is delivered by the indicator.

The idea of using this indicator comes from one of the author's previous work on parabolic problems [4]. In general, when we solve a time-dependent PDE and have a semi-discrete scheme in space, we can compute the derivatives (with respect to time) of semi-discrete solutions as a smoothness or smoothing indicator. Here the semi-discrete solutions involved are those of the form $u_h(t-t_n, t_n, u_h^n)$, where the first variable is the time increment, the second variable is an initial time, and the third variable is an initial value. u_h^n stands for the numerical solution at time t_n. The computation of these high order derivatives should not suffer from amplified white noise if there is sufficient numerical diffusion in the fully discrete scheme. For example, an upwind numerical flux and an TVD Runge-Kutta method may provide sufficient numerical diffusion. In fact, the boundedness of the computed smoothness indicator should be considered as a result of numerical smoothing.

2. The smoothness/shock indicator

To solve the scalar conservation law

$$u_t + f(u)_x = 0 \tag{2.1}$$

with the discontinuous Galerkin method, we have the semi-discrete scheme

$$(u_{h,t}, v)_{\Omega_j} = (f(u_h), v_x)_{\Omega_j} + f(u_h(x_{j-1/2}^-))v(x_{j-1/2}^+) - f(u_h(x_{j+1/2}^-))v(x_{j+1/2}^-) \tag{2.2}$$

in the cell $\Omega_j = [x_{j-1/2}, x_{j+1/2}]$, where the Godunov flux is employed under the assumption $f'(u) \geq 0$ (for simplicity). The semi-discrete solution u_h and the test function v are in a discontinuous piecewise polynomial space of local degree 1, 2 or 3. In each cell, we use the Legendre polynomials as the basis. To compute the fully discrete numerical solution u_h^n, we use the familiar TVD-RK scheme of order 2. [1]

As a smoothness/shock indicator at time t_n, we compute the derivatives of the semi-discrete solution u_h initiated at (t_n, u_h^n). That is, a smoothness/shock indicator is

$$SSI_n = (u_h^n, u_{h,t}, u_{h,tt}, u_{h,ttt}), \tag{2.3}$$

where $u_{h,t}$ is computed at each t_n by replacing u_h in the DG scheme (2.2) with the numerical solution u_h^n. Then, $u_{h,tt}$ and $u_{h,ttt}$ are computed by

$$\begin{aligned} (u_{h,tt}, v)_{\Omega_j} &= (f'(u_h)u_{h,t}, v_x)_{\Omega_j} \\ &+ f'(u_h(x_{j-1/2}^-))u_{h,t}(x_{j-1/2}^-)v(x_{j-1/2}^+) \\ &- f'(u_h(x_{j+1/2}^-))u_{h,t}(x_{j+1/2}^-)v(x_{j+1/2}^-), \end{aligned} \tag{2.4}$$

and

$$\begin{aligned} (u_{h,ttt}, v)_{\Omega_j} &= (f''(u_h)u_{h,t}^2 + f'(u_h)u_{h,tt}, v_x)_{\Omega_j} \\ &+ [f''(u_h(x_{j-1/2}^-))u_{h,t}^2(x_{j-1/2}^-) + f'(u_h(x_{j-1/2}^-))u_{h,tt}(x_{j-1/2}^-)]v(x_{j-1/2}^+) \\ &- [f''(u_h(x_{j+1/2}^-))u_{h,t}^2(x_{j+1/2}^-) + f'(u_h(x_{j+1/2}^-))u_{h,tt}(x_{j+1/2}^-)]v(x_{j+1/2}^-). \end{aligned} \tag{2.5}$$

On the right hand side of (2.4) and (2.5), u_h is replaced by u_h^n, $u_{h,t}$ is computed by (2.2), $u_{h,tt}$ is computed by (2.4).

Remarks:

1. One can compute higher order derivatives in a similar way. It is obvious that the cost of computing each additional high order derivative is proportional to that of implementing one extra stage in a Runge-Kutta scheme. Everything is computed explicitly.
2. For 2D cases, once the semi-discrete DG scheme is determined at t_n, the computation of high order derivatives simply involves taking consecutive derivatives on both sides of the semi-discrete scheme.
3. Other types of numerical fluxes can be treated in a similar way.
4. The indicator is computed independently from the choice of a Runge-Kutta scheme.
5. Indicator (2.3) will miss stationary shocks and stationary contact discontinuities. We can compute another indicator to fix this problem. At t_n, we still use the same formulas (2.2), (2.4) and (2.5) to compute the new indicator. The new indicator is still of the form $SSI_n = (u_h^n, u_{h,t}, u_{h,tt}, u_{h,ttt})$. In each formula, u_h is still replaced by the u_h^n computed for the nonlinear conservation law. The only difference is that we replace the flux in the formulas by $f(u) = u$, $f'(u) = 1$, $f''(u) = 0$. Obviously, the new indicator tells us what would happen if we were solving $u_t + u_x = 0$ with the DG scheme and the initial condition (t_n, u_h^n). This equation has wave speed 1, hence all the spatial discontinuities are reflected by temporal derivatives. Besides, this new indicator may have a role in spatial error control. The original indicator is indeed more costly because of the complexity of $f(u)$, however, it should serve the purpose of temporal error control better than the new one.

In the next section, we will present a few numerical experiments to show what information is delivered by the indicator.

3. Numerical experiments

In the following examples, we solve the Burgers' equation $u_t + (u^2/2)_x = 0$ with different initial conditions. We will have four pictures in each figure from 1 to 5. We refer to the upper-left picture by NW (north west). The other pictures are referred to in a similar way. In each of these 5 figures, picture NW is a numerical solution u_h^n at time t_n, picture NE is the computed $u_{h,t}$, picture SW is $u_{h,tt}$, and picture SE is $u_{h,ttt}$.

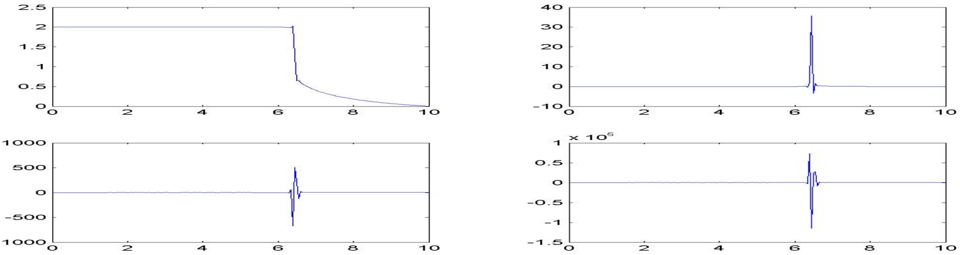

Figure 1: A shock captured without limiter

Example 1. Consider a solution of the Burgers' equation with the boundary condition $u(0,t) = 2$ and the initial condition

$$u(x,0) = \begin{cases} 2 & \text{if } x \in [0,1] \\ 2 - 2\exp(\frac{1}{9} - \frac{9}{(x-1)^2}) & \text{if } x \in (1,10]. \end{cases}$$

This initial function is in $C^\infty[0,10]$, but a shock will appear in the interval at a later time (approximately $t = 1.637$). In Figure 1NW, it shows that a fully developed shock is captured at $t = 2$. We did not use a limiter to catch this shock. Instead, we tried a local degree adjustment technique. To begin with, we use a cubic polynomial in each cell and the 2nd order TVD-RK scheme in time. The cell size is $h = 0.05$, the time step size is $k = 0.0025$. When the computed $u_{h,ttt}$ becomes larger than a parameter $M_3 = 28.3$ in a cell, we reduce the polynomial in that cell to quadratic by simply dropping the cubic Legendre polynomial. When the computed $u_{h,tt}$ becomes larger than a parameter $M_2 = 28.3$ in a cell, we reduce the polynomial in that cell to linear by dropping the quadratic Legendre polynomial. The choice of M_2 and M_3 is based on repeated experiments. They should depend on the cell size. As shown in Figure 1NE and SW, the shock is located clearly in 1-2 cells. Since we do not enforce TVD by using a limiter, slight oscillations are expected. Without rigorous error analysis, we do not want to claim this local degree adjustment technique as an algorithm. We only want to demonstrate that the indicator does deliver useful information about the shock.

Example 2. Let us consider a solution of the Burgers' equation with the boundary condition $u(0,t) = 2$ and the initial condition

$$u(x,0) = \begin{cases} 2 & \text{if } x \in [0,1] \\ 2 - 2(x-1)^2/81 & \text{if } x \in (1,10]. \end{cases}$$

Figure 2NW shows the initial function. Since there is a discontinuity on the second derivative $u_{xx}(x,0)$ at $x = 1$, there is a corner in $u_{h,t}$ shown in Figure 2NE, a jump in $u_{h,tt}$ shown in Figure 2SW, and a spike in $u_{h,ttt}$ shown in Figure 2SE. Corresponding to the spike, u_{ttt} has a Dirac δ-function. The discontinuity will remain in the solution and move forward with the wave.

Figure 3 shows a RKDG solution at $t = 1.33$. The grid size is $h = 0.05$. The time step size is $k = 0.0025$. In every cell, a cubic polynomial is used for the DG discretization. The second order TVD-RK scheme is used for time marching. Figure 3NW and NE show that u_h^n and $u_{h,t}$ are well computed (without oscillation). However, Figure 3SW shows some oscillation of $u_{h,tt}$. Figure 3SE shows that $u_{h,ttt}$ is very rough. This is obviously due to the third order polynomial in the cell of the second order discontinuity. The boundedness of the high order derivatives provides certain assurance on the smallness of the error of the numerical solution. Further work is needed to obtain error estimates.

Figure 4 shows another numerical solution at $t = 1.33$. The grid size and time step size are as above. In most cells, we still use cubic polynomials. The time marching is still done by TVD-RK of order 2. When the smoothness indicator has $|u_{h,ttt}| > M_3 = 5$ in a cell, we lower the degree of the polynomial in that cell to quadratic. Figure 4NW and NE show there is no visible difference on

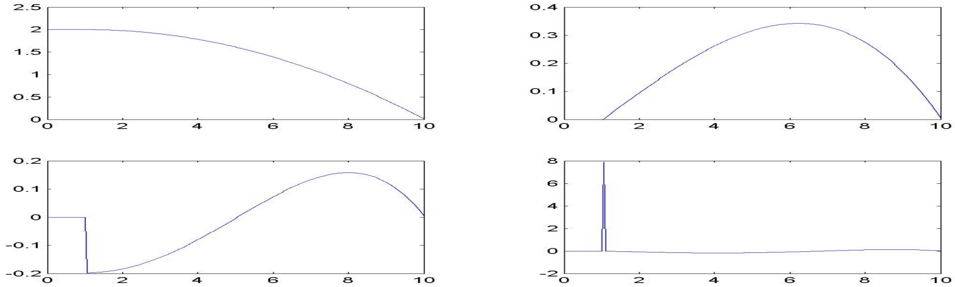

Figure 2: Initial value with a 2nd order discontinuity

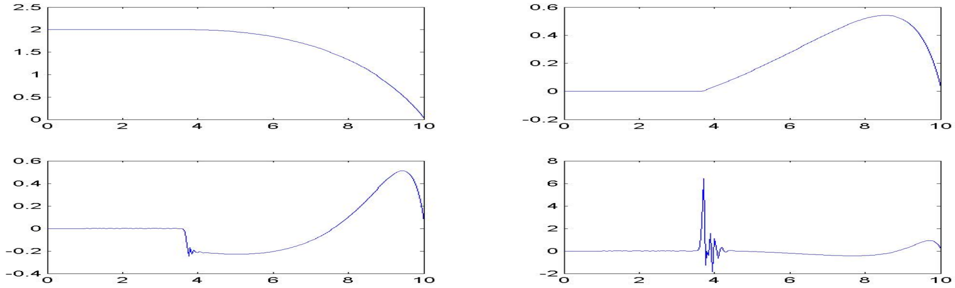

Figure 3: P_3, $k = 0.0025$, minor oscillation with $u_{h,tt}$

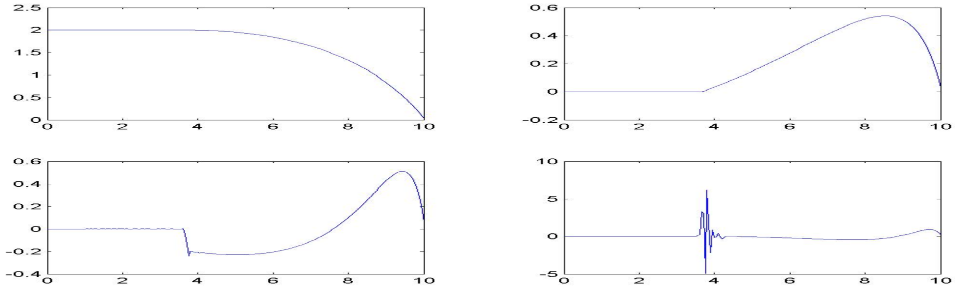

Figure 4: P_m, m adjusted locally, $k = 0.0025$, $u_{h,tt}$ less oscillative

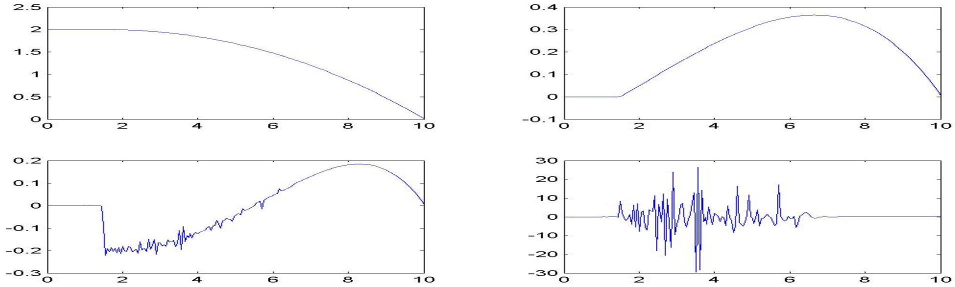

Figure 5: P_m adjusted, $k = 0.003$, $u_{h,tt}$ shows dispersed “numerical instability”

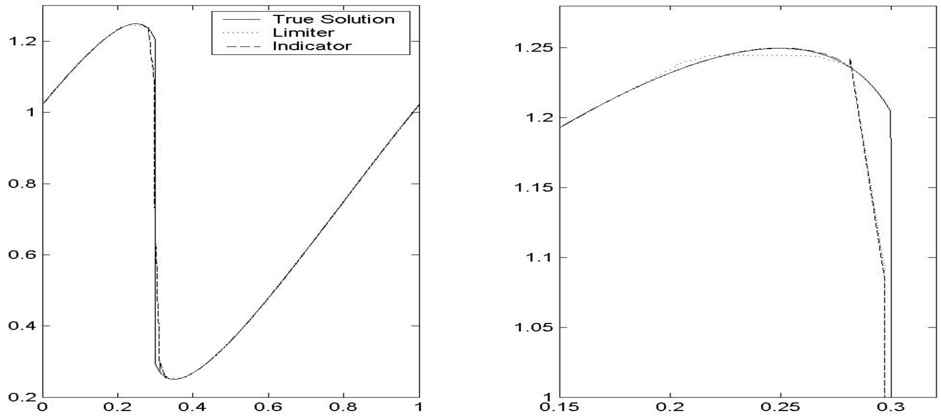

Figure 6: Limiter usage avoided at max and min

u_h^n and $u_{h,t}$. However, Figure 4SW shows that the oscillation of $u_{h,tt}$ is reduced. Meanwhile, Figure 4SE shows that $u_{h,ttt}$ has lost its shape of the δ-function. Although the treatment of of lowering the degree of the polynomial may not be a great one, we have demonstrated that the information delivered by the indicator is indeed useful.

In Figure 5, we increased the time step size to $k = 0.003$ (a 20% increment from 0.0025). The oscillation in $u_{h,tt}$ is not only worse, but also spread out. It is the third degree polynomial causing the numerical effect of oscillation, while it tries to play a role in the approximation of the function in the cell containing the second order discontinuity. From Figure 5SW we can see that the oscillation originated at the discontinuity has been transported to the down stream cells. The oscillation of $u_{h,tt}$ has remained bounded because we used the local degree reduction technique with $M_3 = 5$, otherwise it would have been much bigger. It is worth to point out that the solution u_h^n still has the desired total variation at this time. We have detected a potential "numerical instability" in the sense of TVD and prevented it from getting fully developed. It is also worth to report that, when we stayed with quadratic elements in all the cells, there was no scattered oscillation at all.

It is interesting to observe that a seemingly TVD-stable computation may actually have large local error in a subdomain due to the loss of smoothness in the numerical solution. It is out of the scope of this paper to address the issue in further detail. We only emphasize that the indicator makes such information available.

Example 3. It is known that the use of a limiter can reduce the polynomials to linear in the cells near a smooth extremum. In this experiment, we use the indicator to determine if a limiter should be used in a cell. Only in the cells where the computed $u_{h,tt}$ is bigger than a parameter M_2, do we use a limiter. Consequently, no limiter is used at a smooth extremum. The solution of the Burger's equation with an initial function,

$$u(x,0) = \frac{3}{4} + \frac{1}{2}\sin(2\pi x - \pi),$$

has two extreme points in each period. A shock will appear. Over time, the two extreme points and the shock will merge together. We compute the solution in one period $[0,1]$ with the periodic boundary conditions. $h = 1/64$. $k = 0.0025$. Quadratic polynomials are used in each cell. The 2nd order TVD-RK is used for time marching. The limiter we have used here is the generalized slope limiter $\Lambda\Pi_h^k$ in the lecture notes [1] by Cockburn.

In figure 6, we can see the improvement in the approximation when no limiter is used at the extrema. We have shown three solutions at $t = 0.4$. The solid curve is a solution computed with the Lax-Friedrich scheme and extremely small cell and time step sizes. It serves as an "exact solution"

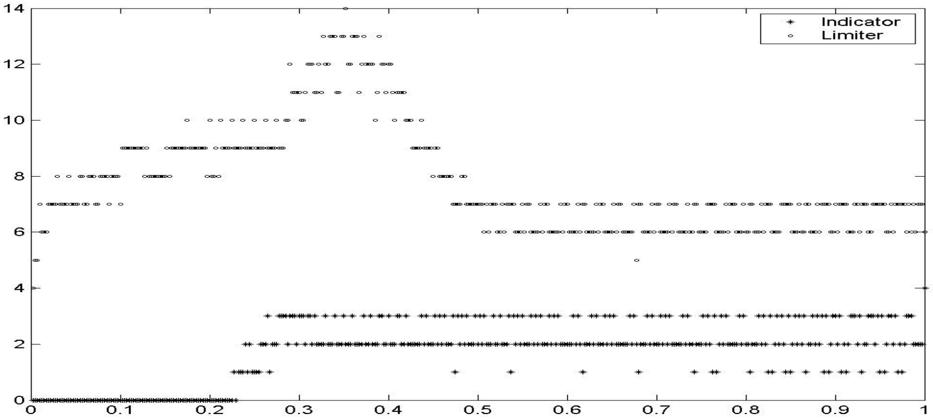

Figure 7: Comparison of limiter usage

for comparison. The dotted curve is a numerical solution computed by using the standard limiter, without using the smoothness/shock indicator. The dashed curve is the numerical solution by using the indicator and the limiter. When the computed $u_{h,tt}$ is bigger than $M_2 = C/h$ ($= 200$ while $h = 1/64$), we use the limiter. We plotted the three solutions in the left picture of Figure 6, and zoomed in at the maximum in the right picture for detailed comparison. Both numerical solutions have captured the shock in two cells. The big difference between the two numerical solutions is shown at the extrema. Since the min-mod function test determined to use the limiter at the extrema, there was only linear approximation over there.

In figure 7, we compare the number of cells where the limiter is used. The o dots indicate the number of cells in which the limiter is used at a given time by the limiter algorithm in [1]. The + dots indicate the number of cells in which the limiter is used as determined by the indicator. When the solution is still smooth, no limiter is used by the indicator, while 3 to 4 cells are subject to limiting at each extremum, according to the min-mod function. When the shock is there, the indicator only uses the limiter in 2 to 3 cells around the shock, while a total of 9 to 13 cells are involved with the limiter in the other case. After both extrema have merged into the shock around $t = 0.5$, the indicator still uses the limiter in 2 to 3 cells, while the min-mod function uses the limiter in 6 to 7 cells. Again, we have demonstrated the usefulness of the indicator.

4. Summary

As shown in the numerical experiments, with the smoothness/shock indicator proposed in this paper, we can deliver information on the smoothness of numerical solutions as either PDE or scheme properties:

1. (a) Shocks; (b) High order discontinuities; (c) Intervals of smooth solutions.
2. (a) Numerical "instability"; (b) Numerical smoothing effects.

There is an open problem: how to quantitatively measure all the information obtained from the indicators. The measurements need to reflect local smoothness properties, and distinguish PDE discontinuities from numerical phenomena.

[1] B. Cockburn, *An Introduction to the Discontinuous Galerkin Methods for the Convection-Dominated Problems*, 1997 C.I.M.E. lecture notes (1697), available at the University of Minnesota webpage.

[2] S. Karni, A. Kurganov and G. Petrova, *A smoothness indicator for adaptive algorithms for hyperbolic systems*, Journal of Computational Physics, 178 (2002), pp.323-341.

[3] T. Sun, *Numerical smoothing of Runge-Kutta schemes*, Journal of Computational and Applied Mathematics, 233 (2009), pp.1056-1062.

[4] T. Sun and D. Fillipova, *Long-time error estimation on semi-linear parabolic equations*, Journal of Computational & Applied Mathematics, 185 (2006), pp.1-18.

[5] S. Zhang and C.-W. Shu, *A new smoothness indicator for the WENO schemes and its effect on the convergence to steady state solutions*, Journal of Scientific Computing, 31 (2007), pp.273-305.