Journal of Computational Mathematics
Vol.28, No.5, 2010, 606–620.

http://www.global-sci.org/jcm
doi:10.4208/jcm.1003-m0003

DISSIPATIVE NUMERICAL METHODS FOR THE HUNTER-SAXTON EQUATION*

Yan Xu
Department of Mathematics, University of Science and Technology of China, Hefei 230026, China
Email: yxu@ustc.edu.cn
Chi-Wang Shu
Division of Applied Mathematics, Brown University, Providence, RI 02912, USA
Email: shu@dam.brown.edu

Abstract

In this paper, we present further development of the local discontinuous Galerkin (LDG) method designed in [21] and a new dissipative discontinuous Galerkin (DG) method for the Hunter-Saxton equation. The numerical fluxes for the LDG and DG methods in this paper are based on the upwinding principle. The resulting schemes provide additional energy dissipation and better control of numerical oscillations near derivative singularities. Stability and convergence of the schemes are proved theoretically, and numerical simulation results are provided to compare with the scheme in [21].

Mathematics subject classification: 65M60, 37K10
Key words: Discontinuous Galerkin method, Local discontinuous Galerkin method, dissipation, Hunter-Saxton equation, Stability.

1. Introduction

In this paper, we present further development of the local discontinuous Galerkin (LDG) method designed in [21] and a new dissipative discontinuous Galerkin (DG) method for the Hunter-Saxton (HS) equation

$$u_{xxt} + 2u_x u_{xx} + uu_{xxx} = 0. \tag{1.1}$$

In [21], we developed a LDG method for the HS type equations and gave a rigorous proof for its energy stability. In this method the basis functions used are discontinuous in space. The LDG discretization also results in a high order accurate, extremely local, element based discretization. In particular, the LDG method is well suited for *hp*-adaptation, which consists of local mesh refinement and/or the adjustment of the polynomial order in individual elements. Numerical simulation shows that the LDG method in [21] approximates the dissipative regularization and the dispersive regularization, as well as continuous solutions of the original HS equation (with possibly discontinuous derivatives) quite well. However, when the derivative has a big jump discontinuity, the LDG solution for the derivative contains spurious numerical oscillations, which are controlled by a nonlinear limiter in [21]. In this paper, we attempt to improve the performance of the LDG method in its control on spurious numerical oscillations near derivative singularities, without sacrificing its accuracy and provable stability. We also design

* Received February 25, 2009 / Revised version received May 15, 2009 / Accepted June 18, 2009 / Published online May 1, 2010 /

a new dissipative DG method with the same improved numerical performance and provable convergence for the piecewise constant case.

The DG method is a class of finite element methods, using discontinuous, piecewise polynomials as the solution and the test space. It was first designed as a method for solving hyperbolic conservation laws containing only first order spatial derivatives, e.g. Reed and Hill [15] for solving linear equations, and Cockburn *et al.* [2–4,6] for solving nonlinear equations. It is difficult to apply the DG method directly to the equations with higher order derivatives. The LDG method is an extension of the DG method aimed at solving partial differential equations (PDEs) containing higher than first order spatial derivatives. The first LDG method was constructed by Cockburn and Shu in [5] for solving nonlinear convection diffusion equations containing second order spatial derivatives. Their work was motivated by the successful numerical experiments of Bassi and Rebay [1] for the compressible Navier-Stokes equations. The idea of the LDG method is to rewrite the equations with higher order derivatives into a first order system, then apply the DG method on the system. The design of the numerical fluxes is the key ingredient to ensure stability. The LDG techniques have been developed for convection diffusion equations (containing second derivatives) [5], nonlinear one-dimensional and two-dimensional KdV type equations [19, 23] and the Camassa-Holm equation [20]. Recently, there is a review paper on the LDG methods for high-order time-dependent partial differential equations [22]. More general information about DG methods for elliptic, parabolic and hyperbolic partial differential equations can be found in the two special journal issues devoted to the DG method [7, 8], as well as in the recent books and lecture notes [9, 14, 16, 17].

This paper is organized as follows. In Section 2, we present and analyze our improved, dissipative LDG method for the HS type equation (1.1). We give a proof of the energy stability in Section 2.3. In Section 3, we present a new dissipative DG method for the HS equation. Stability for general case is proved, as well as convergence for the piecewise constant P^0 case. Section 4 contains numerical results to compare with the results in [21] and to demonstrate the accuracy and capability of the methods. Concluding remarks are given in Section 5.

2. The Dissipative LDG Method for the HS Equation

2.1. Notation

We denote the mesh in $[0, L]$ by $I_j = [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$, for $j = 1, \ldots, N$. The center of the cell is $x_j = \frac{1}{2}(x_{j-\frac{1}{2}} + x_{j+\frac{1}{2}})$ and the mesh size is denoted by $h_j = x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}}$, with $h = \max_{1\le j\le N} h_j$ being the maximum mesh size. We assume that the mesh is regular, namely that the ratio between the maximum and the minimum mesh sizes stays bounded during mesh refinements. We define the piecewise-polynomial space V_h as the space of polynomials of the degree up to k in each cell I_j, i.e.

$$V_h = \{v \in L^2(\Omega) : \quad v \in P^k(I_j) \quad \text{for } x \in I_j, \quad j = 1, \ldots, N\}.$$

Note that functions in V_h are allowed to have discontinuities across element interfaces.

The solution of the numerical scheme is denoted by u_h, which belongs to the finite element space V_h. We denote by $(u_h)^+_{j+\frac{1}{2}}$ and $(u_h)^-_{j+\frac{1}{2}}$ the values of u_h at $x_{j+\frac{1}{2}}$, from the right cell I_{j+1}, and from the left cell I_j, respectively. We use the usual notations $[u_h] = u_h^+ - u_h^-$ and $\bar{u}_h = \frac{1}{2}(u_h^+ + u_h^-)$ to denote the jump and the mean of the function u_h at each element boundary point, respectively.