Journal of Computational Physics 255 (2013) 316–338

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

Fast sweeping methods for hyperbolic systems of conservation laws at steady state

Björn Engquist [1], Brittany D. Froese [*,2], Yen-Hsi Richard Tsai [3]

Department of Mathematics and ICES, The University of Texas at Austin, 1 University Station C1200, Austin, TX 78712, USA

ARTICLE INFO

Article history:
Received 3 June 2013
Received in revised form 14 August 2013
Accepted 16 August 2013
Available online 28 August 2013

Keywords:
Conservation laws
Hyperbolic equations
Fast sweeping methods
Numerical analysis

ABSTRACT

Fast sweeping methods have become a useful tool for computing the solutions of static Hamilton–Jacobi equations. By adapting the main idea behind these methods, we describe a new approach for computing steady state solutions to systems of conservation laws. By exploiting the flow of information along characteristics, these fast sweeping methods can compute solutions very efficiently. Furthermore, the methods capture shocks sharply by directly imposing the Rankine–Hugoniot shock conditions. We present convergence analysis and numerics for several one- and two-dimensional examples to illustrate the use and advantages of this approach.

1. Introduction

The numerical solution of systems of conservation laws,

$$\begin{cases} U_t + \nabla \cdot F(U) = a(U,x), & x \in \Omega,\ t > 0, \\ U = U_0(x), & x \in \Omega,\ t = 0, \\ B(U,x) = 0, & x \in \partial\Omega,\ t > 0, \end{cases} \tag{1}$$

has continued to be an important problem in numerical analysis. A major challenge associated with this task is the need to compute non-classical solutions [8], which leads to the need to develop numerical schemes that correctly resolve discontinuities in weak (entropy) solutions. Several different approaches are now available for resolving shock fronts including front tracking schemes [11], upstream-centered schemes for conservation laws (MUSCL) [7,30], central schemes [18,22], essentially non-oscillatory (ENO) schemes [14], and weighted essentially non-oscillatory (WENO) schemes [21,23,27].

In many applications, it is important to compute the steady state solution of (1), which can be viewed as a particular solution of the boundary value problem

$$\begin{cases} \nabla \cdot F(U) = a(U,x), & x \in \Omega, \\ B(U,x) = 0, & x \in \partial\Omega. \end{cases} \tag{2}$$

A natural approach to computing steady state solutions is to use an explicit time stepping or pseudo time stepping technique to evolve the system to steady state [1,2,6,17]. However, the computational efficiency of these schemes is restricted

* Corresponding author.
E-mail addresses: engquist@math.utexas.edu (B. Engquist), bfroese@math.utexas.edu (B.D. Froese), ytsai@math.utexas.edu (Y.-H.R. Tsai).

[1] This author was partially supported by NSF DMS-1217203.
[2] This author was partially supported by NSF DMS-1217203 and an NSERC PDF.
[3] This author was partially supported by NSF DMS-1217203, a Moncrief Grand Challenge Award, and the National Center for Theoretical Study, Taiwan.

0021-9991/$ – see front matter
http://dx.doi.org/10.1016/j.jcp.2013.08.036

by a CFL condition and the need to evolve the system for a substantial time in order to reach the steady state solution. In order to substantially improve the efficiency of these computations, it is desirable to develop methods that solve the steady state equations directly instead of through a time-evolution process.

Early work in this direction used Newton's method to solve a discrete version of the boundary value problem (2) using shock tracking techniques [12,28]. More recently, Newton solvers have been applied to WENO approximations of the steady Euler equations [16]. For more general systems, a Gauss–Seidel scheme based on a Lax–Friedrichs discretization of the steady state equations was described in [5]. In [13], a homotopy approach was introduced to evolve from an initial condition to a steady state solution without the restriction of a CFL condition.

To gain inspiration, we look at some of the techniques that have been developed for solving static Hamilton–Jacobi equations. Many of the methods commonly used for these equations rely on the fact that information propagates along characteristics. Fast marching methods [15,25] use fast sorting techniques to order the grid points in a way that allows the solution to be computed with a single pass through the computational domain. This approach requires strong assumptions on the monotonicity of the solution with respect to the stencil used, which makes it difficult to apply to problems with anisotropy. Related to this approach are ordered upwind methods [26], which use an optimal control formulation to produce a single-pass solution method. Fast sweeping methods [19,29,31] were introduced to avoid the complexity arising from the sorting procedure required by single-pass methods. Fast sweeping methods, which also make use of the propagation of information along characteristics, involve updating solution values by passing through the computational domain in several pre-determined sweeping directions. This typically leads to algorithms with linear computational complexity.

We introduce a new computational approach for steady state conservation laws that is based on the spirit of the fast sweeping methods. Our fast sweeping approach has a number of advantages. The most immediate advantage is the low computational cost, which is optimal, $\mathcal{O}(N)$, where N is the number of unknowns in the representation of the solution. Secondly, the methods compute shocks sharply by directly imposing the Rankine–Hugoniot shock conditions, together with appropriate entropy conditions. The methods are also flexible in the sense that they can be combined with any reasonable numerical approximation of the flux functions; in fact, in many cases it is possible to obtain correct shock locations using non-conservative schemes. In particular, this allows the easy use of higher order approximation schemes. In some situations, a system of conservation laws (with reasonable boundary conditions) will not have a unique steady state solution; our fast sweeping methods can be used to compute multiple steady states when necessary (Section 3.4.1). Even entropy shocks that are unstable when embedded in time-evolution processes, and therefore cannot be computed using time-stepping based methods, are accessible to our method (Section 3.4.1). Finally, we note that different types of boundary conditions are appropriate in different settings, and some components of the solution vector may not be explicitly given on the entire boundary. However, our methods are powerful enough to solve steady state problems that are well-posed with these "incomplete" boundary conditions; we do not require the problem to be overdetermined through specification of all solution components at the boundary (Section 3.5.1).

The details of the methods will be given in the following sections. Here we simply point out the two main steps:

1. Solution branches are generated by means of an update formula that is used to update the solution along different sweeping directions. In one dimension, for example, one solution branch is obtained by sweeping through the domain from left to right, and another is obtained by sweeping from right to left. In higher dimensions, more sweeping directions are typically employed. When an incomplete set of boundary conditions is given, unknown components of the solution vector at the boundary must be supplied. These are determined via an iteration between steps (1) and (2).
2. A selection principle is used to determine which solution branch is active at each point. In the case of nonlinear conservation laws, the Rankine–Hugoniot conditions for a stationary shock provide a set of equations that determines the shock location and any missing boundary conditions. The numerical algorithm for solving this set of equations guides the iteration. Entropy conditions are also applied to verify the validity of the shock.

2. Background

Before we provide the details of our fast sweeping method, we provide some background material that will inform the approach taken in this work.

2.1. Sweeping methods

The methods we describe here are motivated by the fast sweeping methods for the solution of static Hamilton–Jacobi equations. Fast sweeping methods rely on the fact that boundary data will propagate into the domain along characteristic directions.

We illustrate the basic principles of fast sweeping methods by considering the simple one-dimensional Hamilton–Jacobi equation

$$\begin{cases} (u_x)^2 = u, & 0 < x < 1, \\ u = 1, & x = 0, 1. \end{cases} \tag{3}$$