
Numerical Solutions of Differential Equations : HW1

THOMAS Garrett & WEICKER David

8th February 2016

Introduction

This is the introduction bitches!

1 Conservation laws

We are given the following Euler equations in $u = (\rho, \rho v, E)$,

$$\rho_t + (\rho v)_x = 0 \quad (\text{Conservation of mass}) \quad (1)$$

$$(\rho v)_t + (\rho v^2 + p)_x = 0 \quad (\text{Conservation of momentum}) \quad (2)$$

$$E_t + (v(E + p))_x = 0 \quad (\text{Conservation of energy}) \quad (3)$$

and the variable change $u = (\rho, \rho v, E) = (u_1, u_2, u_3)$. We then have

$$f(u) = (f_1, f_2, f_3)^T = (u_2, u_2^2/u_1 + p(u_1), u_2(u_3 + p(u_1))/u_1)$$

We now wish to find the eigenvalues and eigenvectors of the matrix

$$\frac{\partial f}{\partial u} = \begin{pmatrix} \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} & \frac{\partial f_1}{\partial u_3} \\ \frac{\partial f_2}{\partial u_1} & \frac{\partial f_2}{\partial u_2} & \frac{\partial f_2}{\partial u_3} \\ \frac{\partial f_3}{\partial u_1} & \frac{\partial f_3}{\partial u_2} & \frac{\partial f_3}{\partial u_3} \end{pmatrix}$$

Using simple partial differentiation, we calculate

$$\frac{\partial f}{\partial u} = \begin{pmatrix} 0 & 1 & 0 \\ a & b & 0 \\ c & d & e \end{pmatrix}$$

with $a = -(\frac{u_2}{u_1})^2 + p'(u_1)$,

$b = 2(\frac{u_2}{u_1})$,

$c = \frac{u_2 p'(u_1)}{u_1} - \frac{u_2(u_3 + p(u_1))}{u_1^2}$,

$d = \frac{u_3 + p(u_1)}{u_1}$,

and $e = \frac{u_2}{u_1}$

To get the eigenvalues, we look at the roots of the characteristic polynomial of this matrix

$$\lambda^3 - e\lambda^2 - b\lambda^2 + be\lambda - a\lambda + ae = 0$$

which gives us the three eigenvalues

$$\lambda_1 = \frac{1}{2}(b - \sqrt{4a + b^2})$$

$$\lambda_2 = \frac{1}{2}(b + \sqrt{4a + b^2})$$

$$\lambda_3 = e$$

Plugging in the actual values of a, b , and e , we get

$$\begin{aligned}\lambda_1 &= \frac{u_2}{u_1} - \sqrt{p'(u_1)} \\ \lambda_2 &= \frac{u_2}{u_1} + \sqrt{p'(u_1)} \\ \lambda_3 &= \frac{u_2}{u_1}\end{aligned}$$

After computing, we get the corresponding eigenvectors

$$\begin{pmatrix} 1 \\ \frac{u_2}{u_1} - \sqrt{p'(u_1)} \\ \frac{u_2\sqrt{p'(u_1)} - u_3 + p(u_2)}{u_1} \end{pmatrix}, \quad \begin{pmatrix} 1 \\ \frac{u_2}{u_1} + \sqrt{p'(u_1)} \\ \frac{u_2\sqrt{p'(u_1)} - u_3 + p(u_2)}{u_1} \end{pmatrix}, \quad \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

1.1 Hyperbolicity

Using the u coordinates, we can write equations (1), (2), (3) as

$$u_t + f(u)_x = 0$$

From the chain rule, we can rewrite our equation as

$$u_t + \frac{\partial f}{\partial u} u_x$$

We know that an equation of this form is hyperbolic if $\frac{\partial f}{\partial u}$ has real eigenvalues and linearly independent eigenvectors. We have shown it has a complete set of eigenvectors if

2 Heat Equation

2.1 Flux vector

2.2 $Q(t)$

We are now interested in computing :

$$Q(t) = \int_0^1 \int_0^1 q(x, y, t) dx dy$$

Using the fundamental theorem of calculus, we have :

$$Q(t) = Q(0) + \int_0^t Q'(\tau) d\tau$$

The initial condition gives :

$$Q(0) = \int_0^1 \int_0^1 q(x, y, 0) dx dy = \int_0^1 \int_0^1 0 dx dy = 0$$

We also know that (using the definition of Q) :

$$Q'(t) = \frac{d}{dt} \left(\int_0^1 \int_0^1 q(x, y, t) dx dy \right) = \int_0^1 \int_0^1 q_t(x, y, t) dx dy$$

We are now going to use the PDE to substitute q_t . This yields :

$$Q'(t) = \int_0^1 \int_0^1 q_t(x, y, t) dx dy = \int_0^1 \int_0^1 \nabla \cdot (\nabla q(x, y, t)) dx dy + \int_0^1 \int_0^1 S(x, y, t) dx dy$$

With divergence theorem, and if S means the boundary of the domain and \mathbf{n} the outward unit normal to the boundary, we have :

$$Q'(t) = \oint_S \nabla q(x, y, t) \cdot \mathbf{n} ds + \int_0^1 \int_0^1 S(x, y, t) dx dy$$

Because the given boundary condition given is :

$$\nabla q(x, y, t) \cdot \mathbf{n} = 0$$

We finally have :

$$Q'(t) = \int_0^1 \int_0^1 S(x, y, t) dx dy$$

This yields an expression for Q as a function of t .

$$Q(t) = \int_0^t \int_0^1 \int_0^1 S(x, y, \tau) dx dy d\tau$$

Because S is a given function, this expression can always be computed as a function of t .

3 Discretization and implementation

blabla yo!

4 Numerical results

blabla bitches

5 Refinements

blabla