Spectral Element Method: Lagrange Interpolating Polynomials with Gauss-Lobatto-Legendre Quadrature

Ahmed Abouhussein

December 4, 2018

Abstract

The one dimensional diffusion equation is solved through the Galerkian method with Lagrange interpolating polynomials as basis functions and a Gauss-Lobatto-Legendre (GLL) point distribution. Second order Adam Bashforth scheme was used for time advancement. The numerical solution is validated against an analytical solution. L2-errors indicate spectral convergence up to certain number of grid points.

1. Introduction

The following one dimensional diffusion equation with initial and boundary conditions is considered:

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}, \quad -1 \leq x \leq 1 \tag{1}$$

$$u(x, 0) = \sin(\pi(x + 1)) \tag{2}$$

$$u(-1, t) = u(1, t) = 0 \tag{3}$$

The benchmark solution [1] for this problem is:

$$u(x, t) = \sin(\pi(x + 1)) \exp\left(-\pi^2 t\right) \tag{4}$$

2. Numerical Method

2.1. Lagrange Interpolating Functions and Gauss-Lobatto-Legendre Points

Spectral methods generally rely on describing a function as the sum of orthogonal basis functions multiplied by scaling coefficients:

$$u(x) = \sum_{k=0}^{\infty} a_k l_k(x) \tag{5}$$

To solve the function numerically we approximate the solution with a certain number of grid points N. In this case we consider the Lagrange interpolating function as the basis functions. $l_k(x)$ can be described as :

$$l_j(x) = \prod_{i=0, i\neq j}^{N} \frac{(x - x_i)}{(x_j - x_i)} \tag{6}$$

This choice of basis functions proves useful since the basis functions reduce to delta functions in the discrete space. In other words, the scaling coefficients are just the function values and no transformation is required. Furthermore, the basis functions can satisfy Dirichlet boundary conditions by eliminating the respective polynomials from the basis, i.e. l_0 and l_N.

Since we also have the choice of discrete sampling method, we choose Gauss-Lobatto-Legendre (GLL) sampling over uniform sampling. This eliminates the well known "Gibbs phenomena" as well as simplifies integration (which is needed later when the Galerkian method is formulated). The integral of any polynomial, $p(x)$, of degree equal to or less than 2N + 1 can be solved exactly by a summation:

$$\int_{-1}^{1} p(x)dx = \sum_{k=0}^{N} w_k p(x_k) \tag{7}$$

where x_k and w_k are the GLL points and GLL weights. The GLL points and weights can be solved for if we consider the recursive relation between Legendre polynomials[2]:

$$L_{k+1}(x) = \frac{2k+1}{k+1} x L_k(x) - \frac{k}{k+1} L_{k-1}(x) \tag{8}$$

with $L_0(x) = 0$ and $L_1(x) = x$ .The GLL points are the roots of the polynomial:

$$q(x) = L_{N+1} - L_{N-1} \tag{9}$$

Since it is difficult to analytically solve for the roots, we consider the iterative Newtons method:

$$x_j^{k+1} = x_j^k - q(x_j^k)/q'(x_j^k) \tag{10}$$

where $q'(x_j)$ is defined through the relation:

$$q'(x_j) = (2N+1)L_N(x_j) \tag{11}$$

When iterating with Newtons method, Chebyshev points provide a good initial guess of the solution. Finally the GLL weights can be described as follows:

$$w_j = \frac{2}{N(N+1)(L_N(x_j)^2} \tag{12}$$

2.2. Galerkian Formulation

The Galerkian method, used in developing the solution, involves taking the inner product of the governing equation and a basis function, v, and integrating over the domain. The governing equation then becomes:

$$\int_{-1}^{1} \frac{\partial u}{\partial t} v - \int_{-1}^{1} \frac{\partial^2 u}{\partial x^2} v = 0 \tag{13}$$

Expanding the second term using integration by parts:

$$\int_{-1}^{1} \frac{\partial^2 u}{\partial x^2} v = \frac{\partial u}{\partial x} v \bigg|_{-1}^{1} - \int_{-1}^{1} \frac{\partial u}{\partial x} \frac{\partial v}{\partial x} \tag{14}$$

Since the basis function satisfy the Dirichlet boundary conditions the first term of the integral solution vanishes. Expanding the basis function and the solution using equation (5) and substituting back into the first term of the governing equation:

$$\int_{-1}^{1} \frac{\partial u}{\partial t} v = \sum_{k=0}^{N} w_k \dot{u}_k v_k \tag{15}$$

since the Legendre polynomial $l_i(x_j)$ simplify to δ_{ij}. Similarly the second term becomes:

$$\int_{-1}^{1} \frac{\partial u}{\partial x} \frac{\partial v}{\partial x} v = \sum_{m=0}^{N} \sum_{k=0}^{N} v_m w_k \frac{\partial u}{\partial x} l'_m(x_k) \tag{16}$$

Substituting both terms into the governing equation and using the linear independence property of the basis functions, the governing equation becomes:

$$\dot{u}_k w_k + \sum_{m=0}^{N} w_m \frac{\partial u}{\partial x} l_k'(x_m) = 0, \quad k = 1, 2, ..., N-1 \tag{17}$$

Expanding $\frac{\partial u}{\partial x}$ using equation (5) and substituting back into equation (17):

$$\dot{u}_k w_k + \sum_{n=0}^{N} u_n \sum_{m=0}^{N} w_m l_n'(x_m) l_j'(x_m) = 0, \quad k = 1, 2, ..., N-1 \tag{18}$$

This can be re-written as:

$$\dot{u}_j + \sum_{n=0}^{N} u_n \hat{G}_{kn} = 0, \quad k = 1, 2, ..., N-1 \tag{19}$$

where $\hat{G}, G$ and D are differentiation matrices 198 [3]:

$$\hat{G}_{kn} = \frac{1}{w_k} G_{kn}, \quad k = 1, 2, ..., N-1 \tag{20}$$

$$G_{kn} = \sum_{i=0}^{N} D_{in} D_{ik} w_i, \quad k = 1, 2, ..., N-1 \tag{21}$$

$$(D_N)_{ik} = \left\{ \begin{array}{ll} \frac{L_N(x_i)}{L_N(x_k)(x_i - x_k)}, & i \neq j \\ \frac{(N+1)N}{4}, & i = k = 0 \\ \frac{-(N+1)N}{4}, & i = k = N \\ 0, & \text{otherwise} \end{array} \right\}$$

Equation (19) is solved by the explicit second order Adam Bashforth scheme.

3. Discussion

Lagrange interpolating polynomials with GLL points were implemented for the one dimensional diffusion equation. Convergence to exact solution was achieved with only 8 grid points.The results indicate that the method is stable and reliable for

Equation (1) is solved using the relevant boundary and initial conditions and compared against the benchmark solution (equation 4) for different time limits and grid points N. The error criteria for the convergence of Newton's method for GLL points was set to $1e-7$. The solution time step was set to $1e-6$. Solution approximation and L2-error norms were performed for N= $4, 8, 16, 32.64$

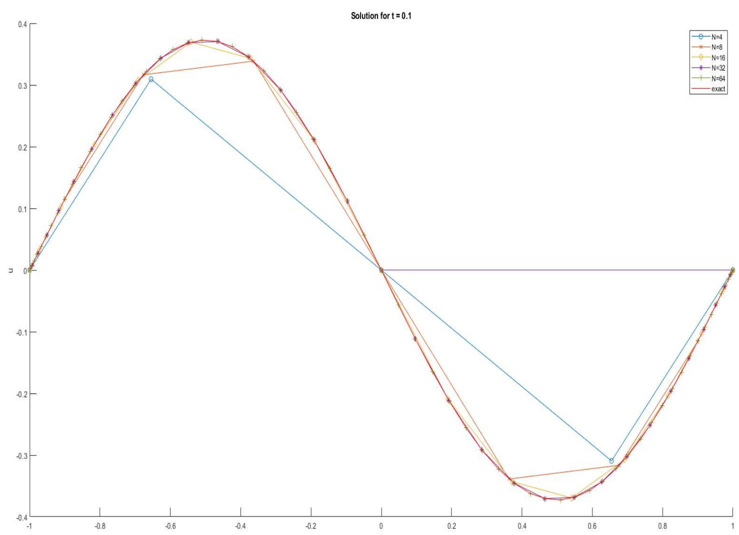

Figure 1: Solution with different N values

The Lagrange interpolating polynomials converge to the exact solution as N increased from 4 to 8. Adding extra grid points does not seem to create a qualitative difference in the approximated solution.

The L2 error norm shows spectral convergence between N = 4 and N = 8, albeit not at spectral accuracy. Moreover increasing N past 16 seems to actually increase the error. The lack of spectral accuracy and the increase in error can be attributed to two things: the second order accurate temporal

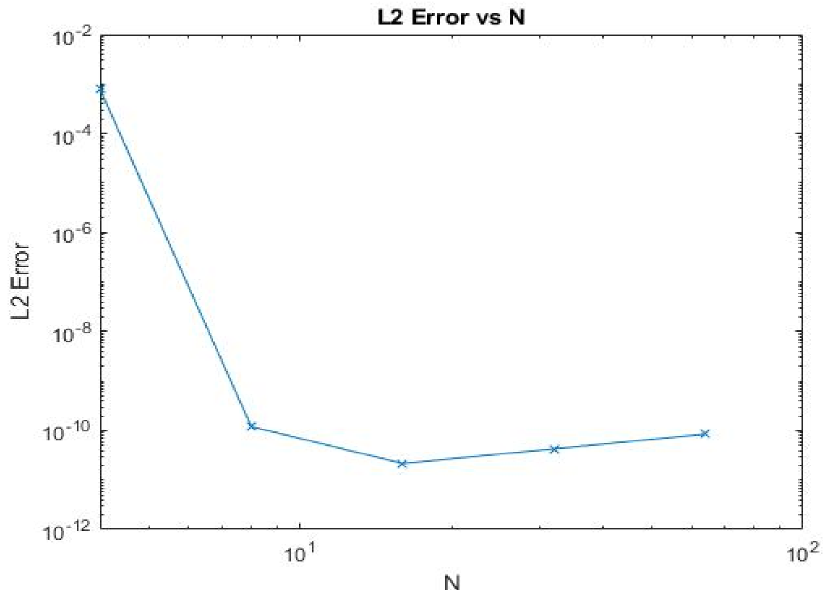

Figure 2: L2 error norm

scheme and the numerical errors arising from the convergence of the Newtons iteration method. While an argument can be made for setting the a more strict criteria for Newtons method, the temporal scheme error can not be remedied. I believe the numerical errors arising from the temporal scheme will result in an increased error between increasing values of N, regardless of Newtons method convergence criteria.

Figure 3 indicated that the method is stable and produces accurate results for different time values.

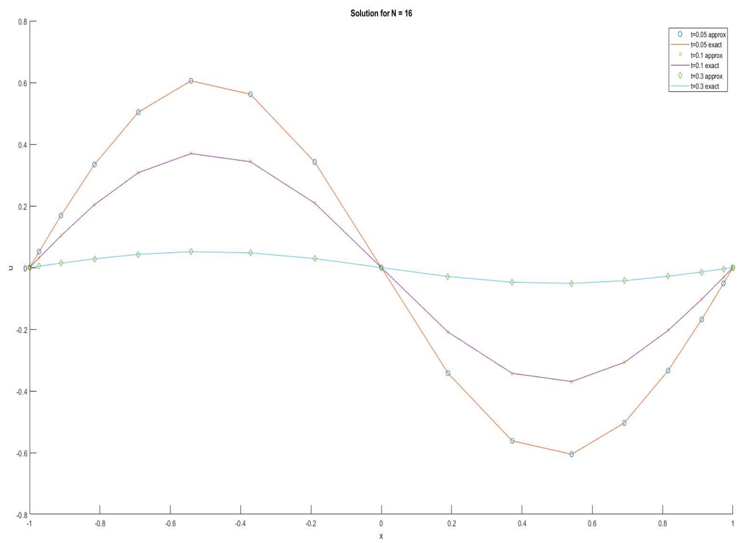

Figure 3: time

4. Discussion

Lagrange interpolating polynomials were implemented with GLL points to solve the one dimensional diffusion equation. The method produces accurate results with fast computation times for low number of gird points (N=8). This spectral method is a good choice for dirchlet boundary condition problems. The implementation is straight forward, however careful attention needs to be paid to indices. An index mistake could be difficult to detect since the indexing between references and programming languages. Moreover, careful attention needs to be paid when choosing an initial guess for the GLL points. As previously indicated Chebyshev points are a good starting guess but different variations could be implemented. According to the nature of the formulation (spectral in space, 2nd order in time) as well as stability concerns, it appears that low N values ($N = 8$ or $N = 16$) seem to provide optimal solutions for one dimensional problems. If I had more time I would implement the code in two dimensions. It seems I have all the tools needed to solve the Poission equatio, for example.

.

5. Bibliography

[1] D. A. Kopriva, Implementing Spectral Methods for Partial Differential Equations, Springer, 2009.

[2] B. Merrill, Using the Galerkin Method with Lagrange Interpolating Polynomials and Gauss-Lobatto-Legendre Quadrature, 2012.

[3] Spectral methods in fluid dynamics, Springer series in computational physics, Springer-Verlag, Berlin ; New York, 1988.