

# A Discontinuous Galerkin program DG(P1) for solving 1D Nonlinear Advection-Diffusion equations

Karthik Reddy Lyathakula

*North Carolina State University, Raleigh, United States*

---

## Abstract

The objective of this project is to develop a program to numerically solve nonlinear advection-diffusion equation for 1d problem using the discontinuous Galerkin (DG) finite element method. The code has been written using C++ compiler in Linux environment. The second order derivative in the nonlinear advection-diffusion equation is discretized using Direct Discontinuous Galerkin method (DDG) and Bassi-Rebay method. The order of accuracy of these discretization methods is determined by solving for a problem which has an exact solution and finally the program is tested for full advection-diffusion equation.

---

## 1. Introduction

In project, the DG method is used to solve the nonlinear advection-diffusion equation. In practical applications, the flow is viscous and it is required to solve complete Navier Stokes equation to capture the actual physics. In this project the capability of DG methods to solve equation with diffusion terms is tested by solving 1D nonlinear advection-diffusion equation.

## 2. Governing equations

The Non-linear advection-diffusion equation solved in this project is given by

$$\frac{\partial u}{\partial t} + \frac{\partial F}{\partial x} = c \frac{\partial^2 u}{\partial x^2} \quad (1)$$

here F is

$$F = au + v \frac{u^2}{2} \quad (2)$$

where a b and c are constants.

## 3. Discontinuous Galerkin method

The one dimensional nonlinear equation is solved by Discontinuous Galerkin method. Discontinuous Galerkin methods is sub set of Galerkin methods (Ref to project 1), where the basis function is same as the weigh function, but the basis function is

continuous only inside each finite dimensional subspace(each mesh cell). The weak formulation for eq 1 is given by

$$\begin{aligned} \int \frac{\partial U_h}{\partial t} B d\Omega + \int \frac{\partial F(U_h)}{\partial x} B d\Omega &= \int \frac{\partial^2 u}{\partial x^2} B d\Omega \\ \int \frac{\partial U_h}{\partial t} B d\Omega + \int \left( \frac{\partial(BF(U_h))}{\partial x} - F(U_h) \frac{\partial B}{\partial x} \right) d\Omega &= \int \frac{\partial^2 u}{\partial x^2} B d\Omega \end{aligned} \quad (3)$$

applying gauss divergence theorem gives

$$\int_{\Omega_e} \frac{\partial U_h}{\partial t} B d\Omega + \int_{\Gamma_e} BF_i(U_h) n_i d\Gamma - \int_{\Omega_e} F_i(U_h) \frac{\partial B}{\partial x_i} d\Omega = \int \frac{\partial^2 u}{\partial x^2} B d\Omega \quad (4)$$

here B is the weight function. The unknown  $U_h$  for each subspace can be approximated by different types of basis functions like Lagrarian, Taylor basis functions or legandary. In this project, Taylor basis are used which is given by

$$U_h = U_c(t) + \frac{\partial U(t)}{\partial x} \Big|_c (x - x_c) + \frac{\partial^2 U(t)}{\partial x^2} \Big|_c \frac{(x - x_c)^2}{2} \quad (5)$$

the unknowns in the Taylor basis are the cell averaged variables and their derivatives at the center of the cells regardless of the element shapes. Normalized Taylor basis function is given by

$$U_h = \bar{U} B_1 + U_x B_2 + U_{xx} B_3 \quad (6)$$

where Basis functions ( $B_i$ ) are given by

$$B_1 = 1 \quad B_2 = \frac{x - x_c}{\Delta x} \quad B_3 = \frac{(x - x_c)^2}{2\Delta x^2} - \frac{1}{\Omega_i} \int_{\Omega_i} \frac{(x - x_c)^2}{2\Delta x^2} d\Omega \quad (7)$$

here

$$\Delta x = \frac{x_{max} - x_{min}}{2} \quad \Delta x_c = \frac{x_{max} + x_{min}}{2} \quad (8)$$

$x_{max}$ ,  $x_{min}$  are the maximum and minimum coordinates in x direction. In this project DGP1 method is used where the unknown is approximated to be linear inside each cell. To discretize the second order term in Eq. 4, there are different approaches like Vanleer recovery method, Direct discontinuous Galerkin method (DDG), Bassi Rebbi(BR2) method. In this project BR2 and DDG methods are implemented. For the sake of brevity, these methods are not been included in this report.

#### 4. Time integration

The time-derivative in 1 has been integrated by a 3-stage Runge-Kutta scheme, with an option of using a 1-stage Runge-Kutta scheme. The 3-stage Runge-Kutta scheme is given by,

$$\begin{aligned} U^{(1)} &= U^n + \Delta t M^{-1} R(U^n) \\ U^{(2)} &= \frac{3}{4} U^n + \frac{1}{4} (U^{(1)} + \Delta t M^{-1} R(U^{(1)})) \\ U^{(3)} &= \frac{1}{3} U^n + \frac{2}{3} (U^{(2)} + \Delta t M^{-1} R(U^{(2)})) \\ U^{n+1} &= u^{(3)} \end{aligned} \quad (9)$$

Note that the RHS flux matrix is calculated on the basis of the solution at the previous RK stage and the boundary conditions are applied after each RK stage.

## 5. Results and Discussion

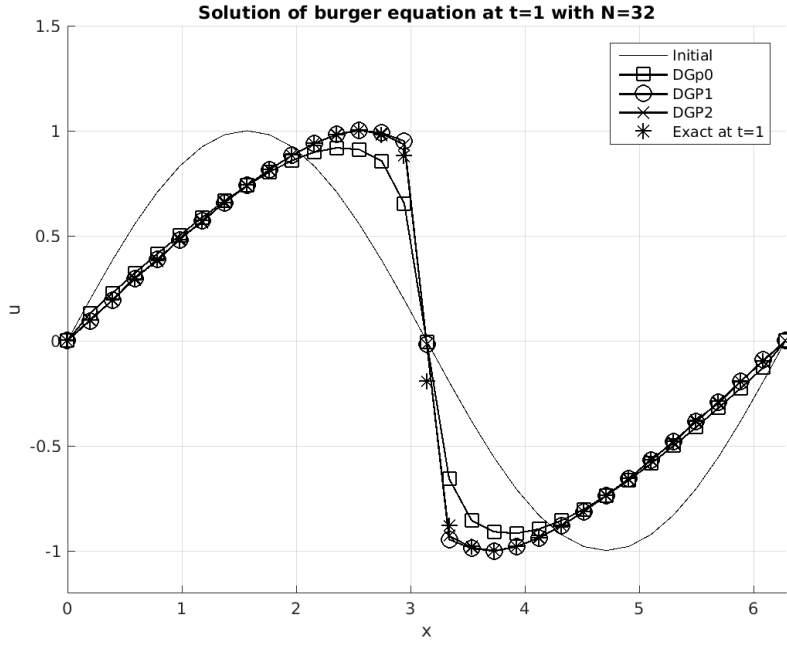
A generalized C++ program is developed to solve the nonlinear advection equation using DGP1 and BR2, DDG methods for discretizing the diffusion terms in the equation. The choice of method to solve the nonlinear advection equation can be controlled by an input file. A systematic approach is followed to solve the nonlinear equation. First the burger equation ( $a=0, c=0$ ) is discretized for a general case and it is tested by solving on two initial conditions: Sin wave and a Gaussian pulse and compared with the analytical solution. The analytical solution for the burger equation is shown below:

$$\begin{aligned}\frac{\partial u}{\partial t} + u \frac{\partial}{\partial x} &= 0 \\ u(x, 0) &= g(x)\end{aligned}\tag{10}$$

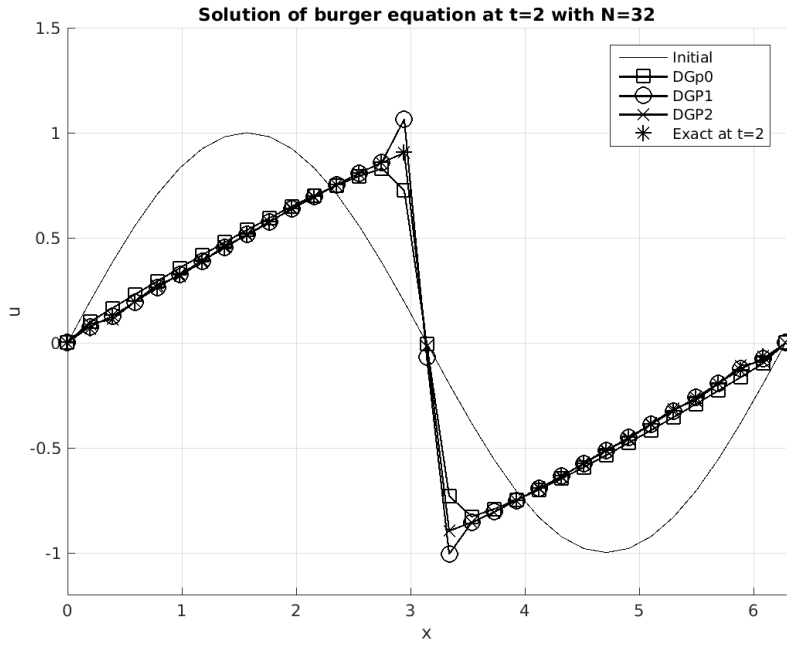
the analytical solution for continuous  $g(x)$  is given by

$$\begin{aligned}u(x, t) &= g(\zeta) \\ x &= g(\zeta)t + \zeta\end{aligned}\tag{11}$$

for each coordinate at required time( $t$ ), the above nonlinear equation has to be solved for  $\zeta$  to get the exact solution  $u(x, t)$ . The exact solution for the comparison is obtained by solving above equation using Newton method. Fig. 1 shows the comparison of numerical solution with analytical for an initial sin wave, at  $t=1$  and  $t=2$ . The results shows that the numerical solution is closer to the exact solution for all the methods. However, DGP2 method is more accurate than compared to DGP0 and DGP1 method. From the plot, it can be observed that sin wave gets distorted with time and creates sharp gradient at  $x=\pi$ . Fig. 2 shows the comparison of results with a Gaussian pulse and the results are matching with the analytical solution. Note that the burger equation is solved for DGP0, DGP2 method and however only DGP1 method is used for solving entire nonlinear advection-diffusion equation.

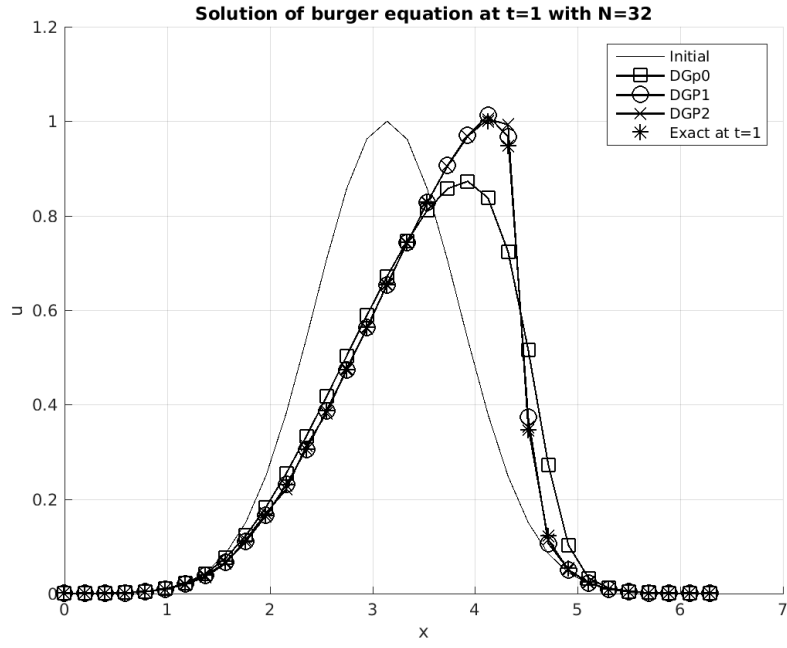


(a)  $t=1$

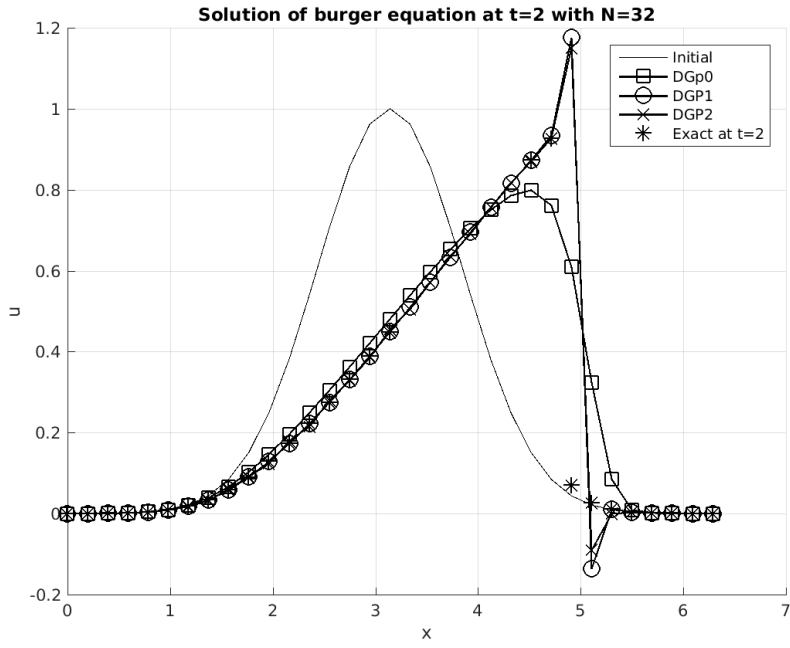


(b)  $t=2$

Figure 1: Comparison of numerical solution of burger equation with analytical solution for a sin wave.



(a)  $t=1$



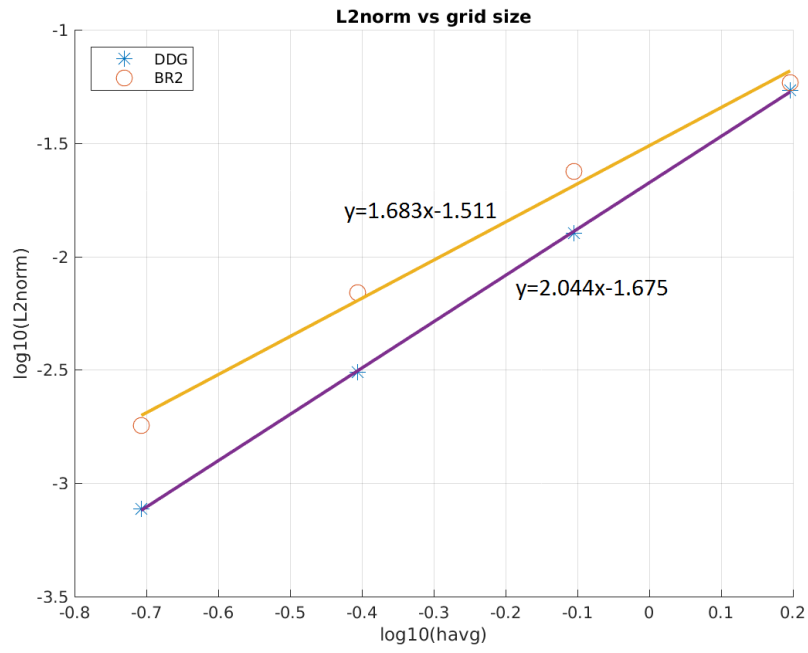
(b)  $t=2$

Figure 2: Comparison of numerical solution of burger equation with analytical solution for a Gaussian pulse.

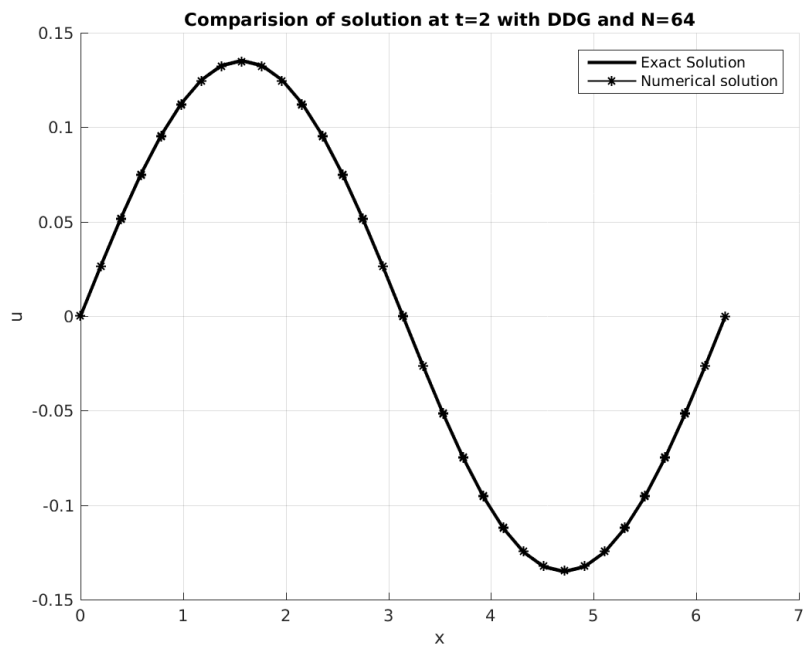
After successfully testing the DG methods for burger equation, the program is appended to include the discretization of diffusion terms. Two types of methods have been implemented to discretize the diffusion terms: BR2 and DDG. First, the order of accuracy of these methods is determined by solving just the diffusion equation( $a=b=0$   $c=1$ ) with an initial condition as  $u(x, 0) = \sin(x)$  on a domain, extending from  $x=0$  to  $x=\pi$ , with 4, 8, 16, 32 cells . The exact solution for this case is derived using separation of variables and is given by  $e^{-ct}\sin(x)$ . The solution is computed at  $t=2$  and the L2 norm error is determined for all the grids. The order of accuracy is determined by the relation between L2 norm error and grid size given by:

$$\begin{aligned} ||E||_{L^2} &= C * h^\alpha \\ \log_{10}(|E|_{L^2}) &= \alpha * \log_{10}(h) + \log_{10}(C) \end{aligned} \tag{12}$$

here  $\alpha$  determines the order of accuracy. Fig. 3 shows the plot of  $\log_{10}(\text{L2norm})$  vs  $\log_{10}(\text{havg})$  for DDG method and BR2 method. The results shows that the DDG method is second order accurate and order of accuracy for BR2 method is less than DDG.



(a) Error function



(b) Error function

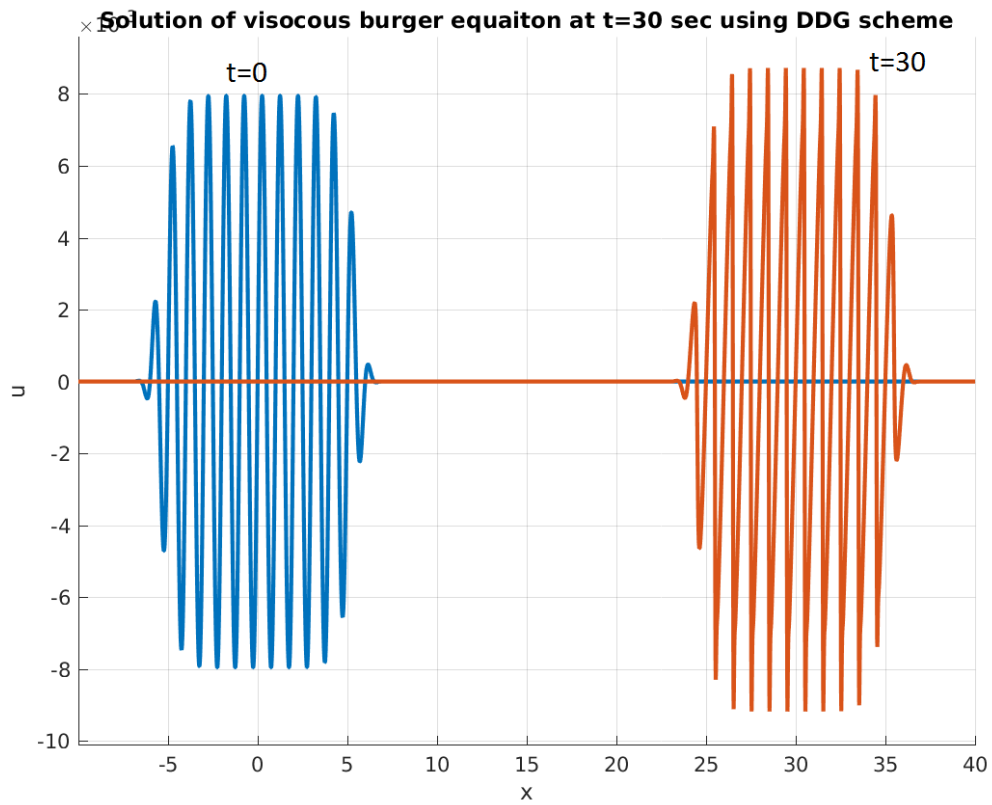
Figure 3: Solution of pure diffusion problem

Next the code for entire nonlinear advection-diffusion equation is tested on a computations domain includes 4000 mesh cells extending from  $x=-10$  to  $x=40$  with  $a=1$   $b=1$   $c=1.5e-5$  and following initial condition

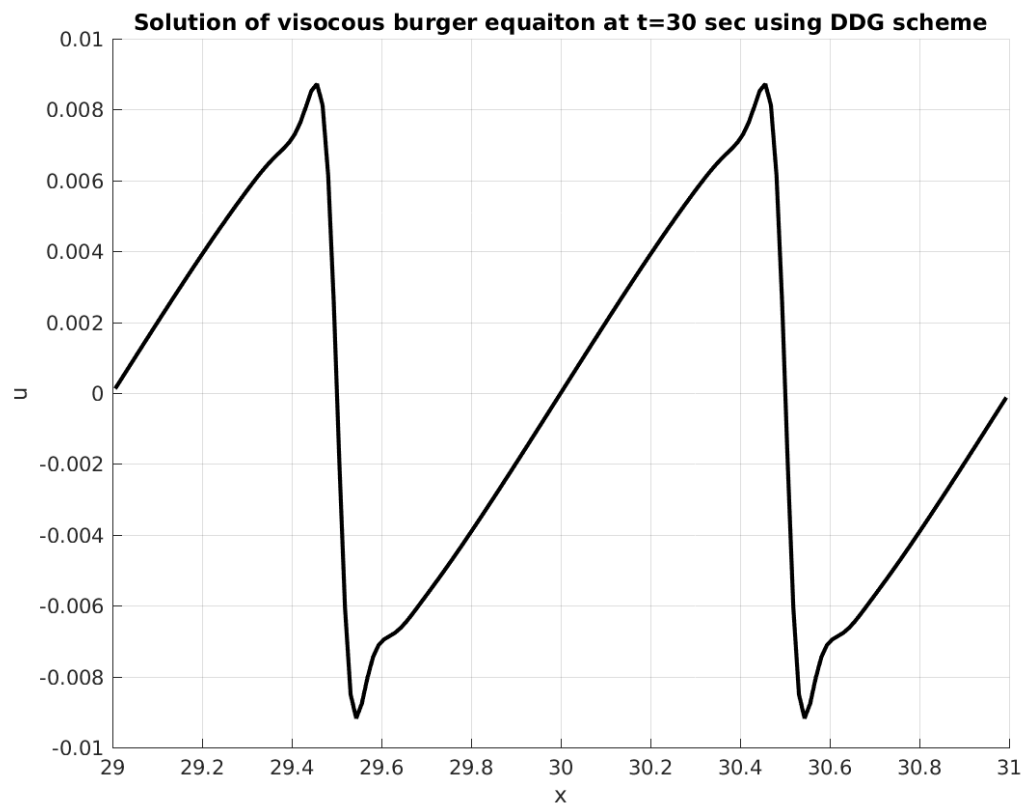
$$\begin{aligned} u(x, 0) &= u_0 \exp((- \beta x)^{10}) \sin(2\pi x) \\ u_0 &= 7.96e - 3 \quad \beta = 0.179 \end{aligned} \tag{13}$$

Fig. 4 shows the solution at  $t=30$  with DDG method and Fig 5 shows the solution using BR2 method. It can be observed the sin wave distorts creating sharp gradients due to nonlinear advection and the wave is advected with a velocity of 1 because of the linear advection term.

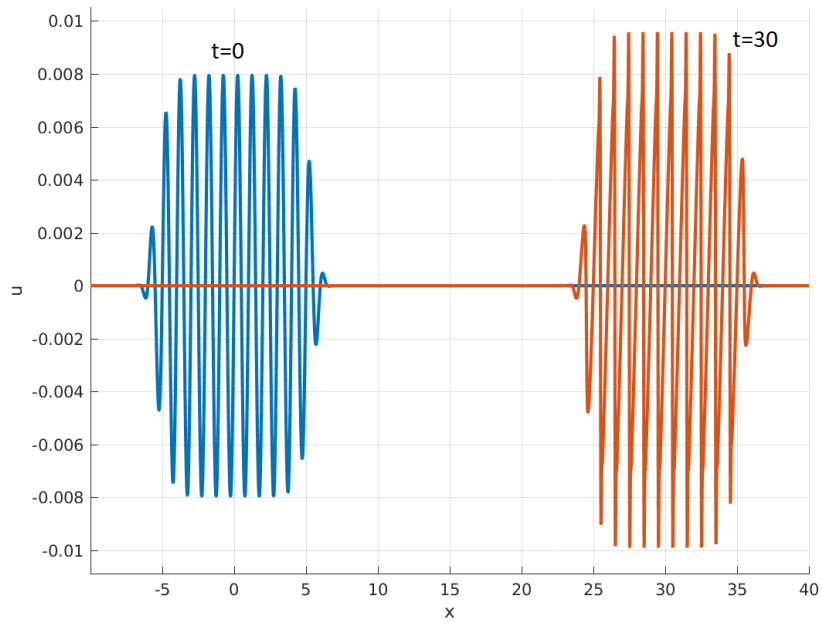




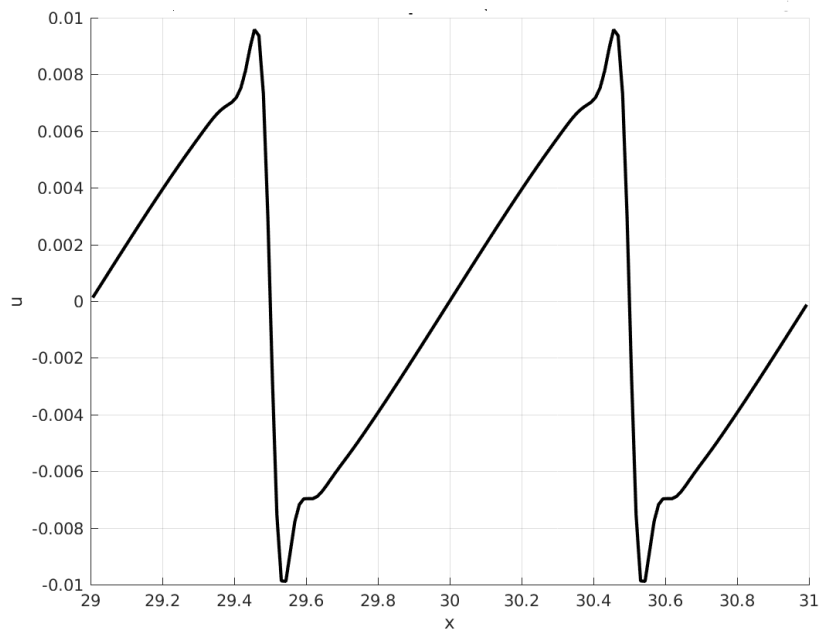
(a) Solution



(b) Solution



(a) Solution



(b) Solution

Figure 5: Solution using BR2 Method

## 6. Conclusion

The DG methods are tested successfully by solving diffusion equations using DDG and BR2 method. It is observed that the order of accuracy is second with both the methods. In conclusion DG methods will increase the order of accuracy if used for solving flow equations.