Master Of Science in Computational Mechanics

Finite Elements in Fluids

Chinmay Khisti (20 February 2019)

Assignment 2 :Stabilization Techniques

1 Strong form of a problem and its boundary conditions

General expression for convection diffusion problem

$$\frac{\partial \mathbf{u}}{\partial t} + a \cdot \nabla u - \nabla \cdot (\nu \nabla u) + \sigma u = s(x, t, u) \tag{1}$$

in which,

$$u \rightarrow \text{ property, } a \rightarrow \text{ velocity of convection, } \nu \rightarrow \text{ diffusivity, } \sigma \rightarrow \text{ reaction, } s \rightarrow \text{ source term}$$

In 1D convection diffusion problem no reaction terms are considered therefore equation (1) becomes,

$$a \cdot \nabla u - \nabla \cdot (\nu \nabla u) = s \tag{2}$$

Given boundary conditions for current problems are,

Prob. 1: $u(0) = 0$ $u(1) = 1$ $s = 0$
Prob. 2: $u(0) = 0$ $u(1) = 0$ $s = 1$
Prob. 3: $u(0) = 0$ $u(1) = 1$ $s = 10e^{-5x} - 4e^{-x}$
Prob. 4: $u(0) = 0$ $u(1) = 1$ $s = sin(\pi x)$

2 Weak form with Galerkin and its discretization

To obtain weak form of equation (2) an arbitrary weighting function is multiplied to both side of equation and integrated over the domain

$$\int_\Omega w(a \cdot \nabla u - \nabla \cdot (\nu \nabla u) d\Omega = \int_\Omega w s d\Omega$$

The function can be divided into n number of elements of size $h = 1/n$ thus the approximation can be done as follows,

$$u \approx u^h = \sum_{i=1}^{n+1} N_i u_i \tag{3}$$

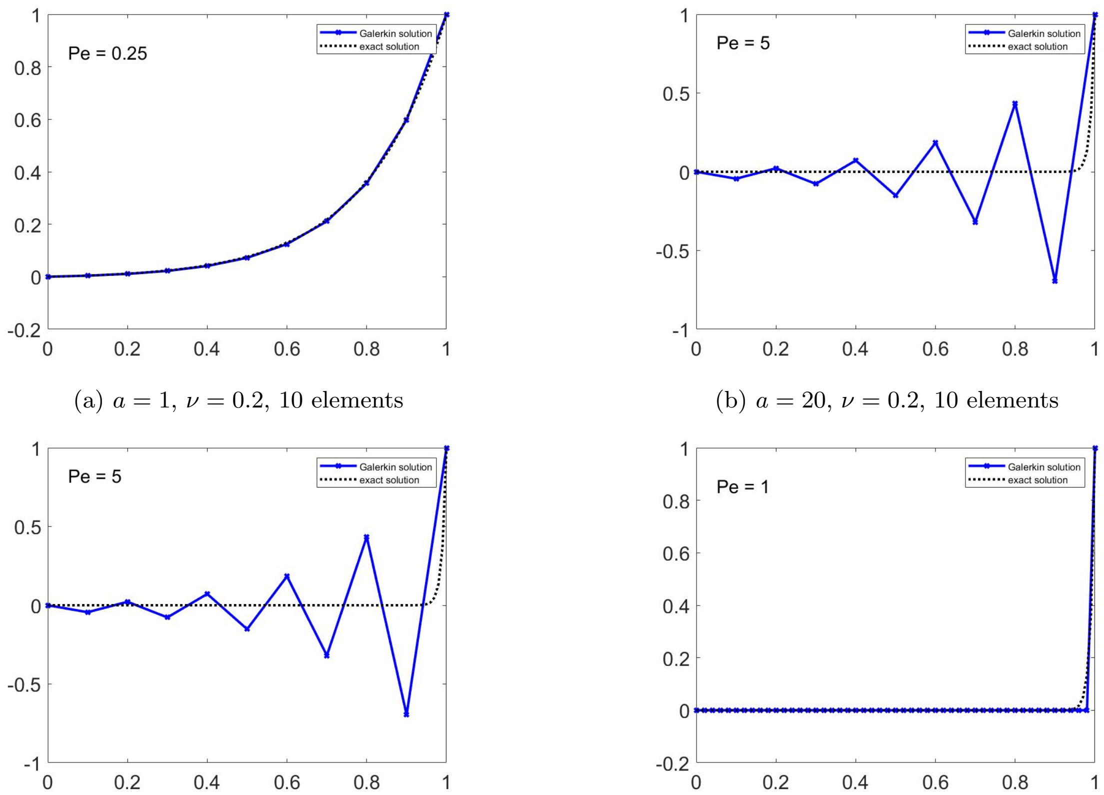
where, u_i is function for node i and N_i is shape function for the element
Linear $N_1 = 1/2(1 - \xi)$ $N_2 == 1/2(1 + \xi)$
Quadratic $N_1 = 1/2(1 - \xi) * \xi$, $N_2 = 1 - \xi^2$, $N_1 = 1/2(1 + \xi) * \xi$
Implementing shape function to the equation we get,

$$(K_c + K_d)u = f$$

Where,
$K_c = \int_\Omega N_i(a \cdot_j) d\Omega$

$K_d = \int_\Omega \nabla N_i \cdot (\nu_j) d\Omega$

$f_i = \int_\Omega N_i s d\Omega$



(a) $a = 1$, $\nu = 0.2$, 10 elements

(b) $a = 20$, $\nu = 0.2$, 10 elements

(c) $a = 1$, $\nu = 0.01$, 10 elements

(d) $a = 1$, $\nu = 0.01$, 50 elements

Figure 1: Prob. 1, $u(0) = 0$ $u(1) = 1$ $s = 0$, Linear Element, Galerkin

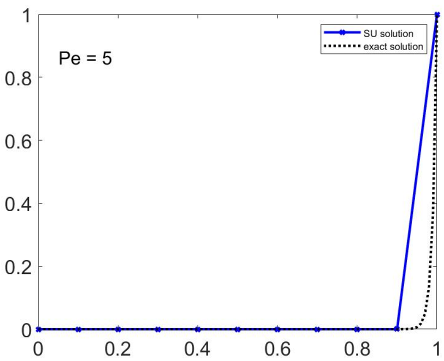
3 MATLAB example

Code provided is fully functional for Galerkin and SU formulation for linear element and for the Prob 1, 2 and 4. The following changes were made in the code to full fill the problem asked.

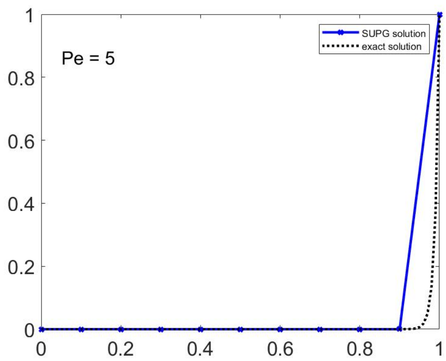
1. Complete code for SPUG and GLS were added to the main program with stabilisation factors compatible to quadratic and linear element also.

2. New source term $10e^{-5x} - 4e^{-x}$ was added to the code.

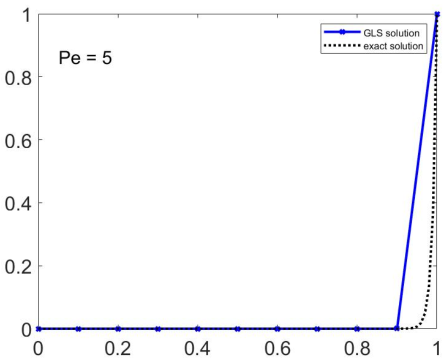
Figure 1. shows plot for various Peclet Number ($Pe = ah/2\nu$) i.e. by varying convection coefficient a, Diffusion coefficient ν, number of elements for linear element type with Galerkin formulation. as from the figure you can observe that the consistency of the graph is low for Peclet number ($Pe > 1$)



(a) SU



(b) SUPG

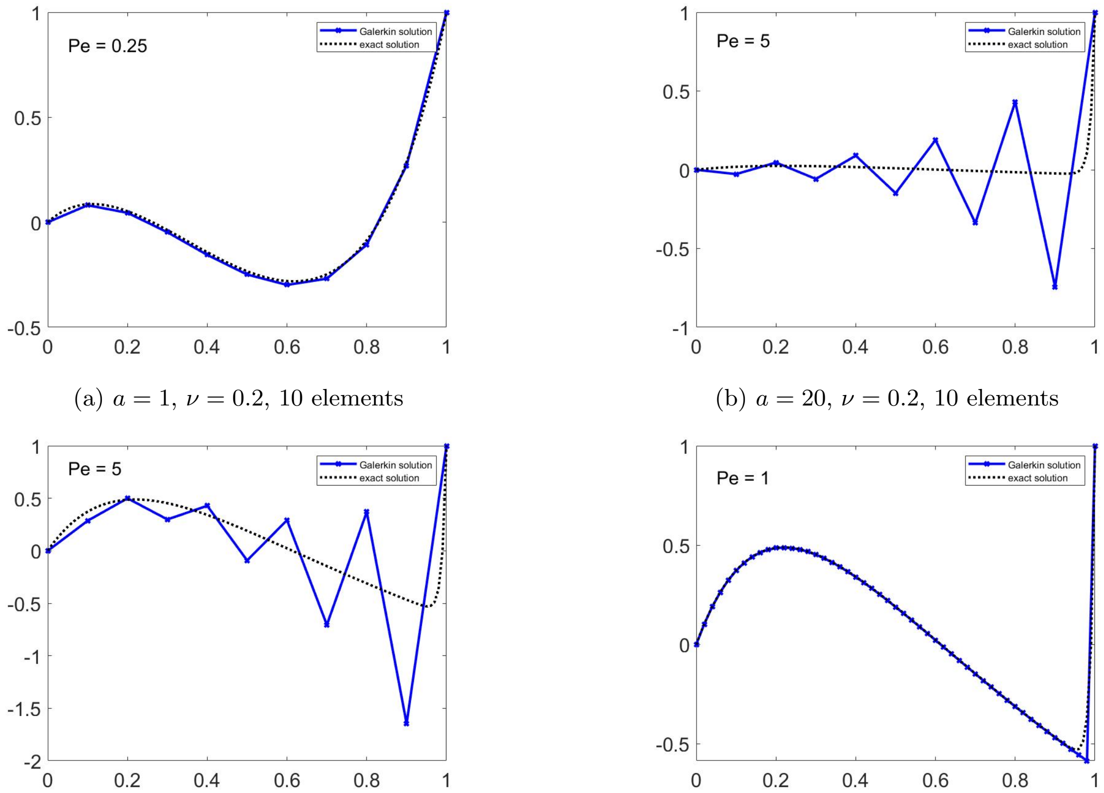


(c) GLS

Figure 2: Prob. 1, $a = 1$, $\nu = 0.01$, 10 Linear Elements

After solving the case for $a = 1$, $\nu = 0.01$, 10 linear elements we can observe in Figure 2. we get almost similar result for remaining methods Stream upwind (SU), Streamline- Upwind Petrov-Galerkin (SUPG) and Galerkin Least Squares(GLS). for the specified BCs and source term for Prob. 1

When optimal stabilization parameter was used for the problem exact solution was obtained at the nodes of the problem.



(a) $a = 1$, $\nu = 0.2$, 10 elements

(b) $a = 20$, $\nu = 0.2$, 10 elements

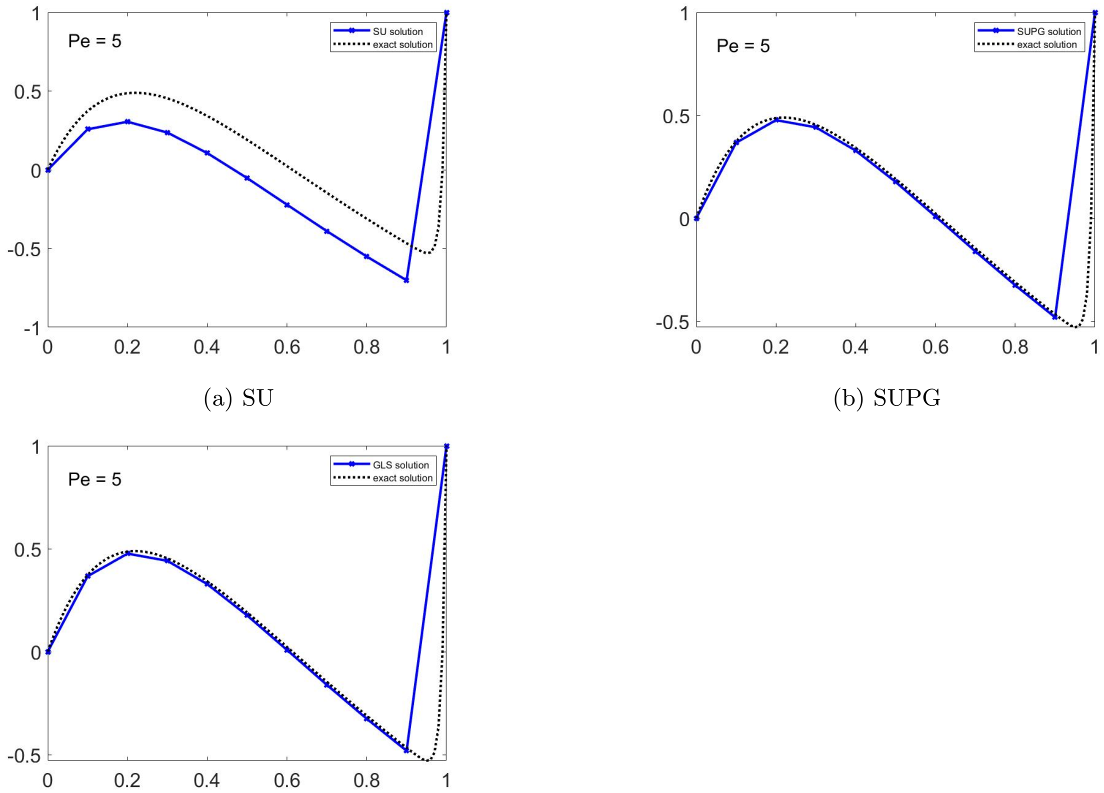
(c) $a = 1$, $\nu = 0.01$, 10 elements

(d) $a = 1$, $\nu = 0.01$, 50 elements

Figure 3: Prob. 3, $u(0) = 0$ $u(1) = 1$ $s = 10e^{-5x} - 4e^{-x}$, Linear Element, Galerkin

Moving on to next question in Figure 3. you can see the plots for the Prob. 3 having source term $10e^{-5x} - 4e^{-x}$ we can see the conclusions made earlier for Galerkin formulation holds true that the solution has higher oscillations when $Pe > 1$

Figure 4 gives prospective of different methods as compared to Figure 2 as source term changes specially in SU formulation the solution follow the exact solution but we do not get exact values even at nodes



(a) SU

(b) SUPG

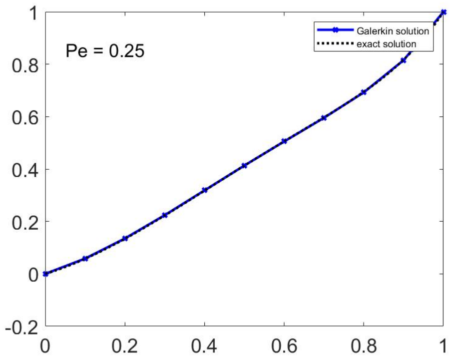
(c) GLS

Figure 4: Prob. 3, $a = 1$, $\nu = 0.01$, 10 Linear Elements

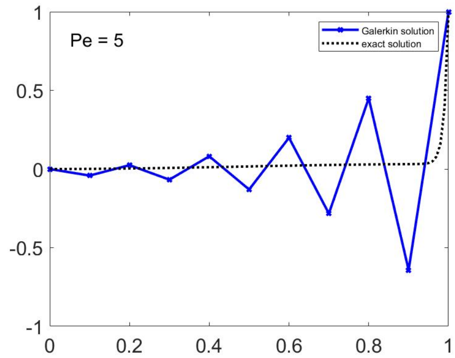
Now coming to Quadratic element as you can see from Figure 7 Galerkin formulation have most oscillations but follows along path of exact solution, where as in SU formulation there are minimum oscillations but there are no exact values observed at nodes. on the other hand in SUPG and GLS the solution is pretty much the same except SUPG require less time to calculate same solution then GLS. Table 1 shows computing time for different methods for the Prob. 3 with $a = 1$, $\nu = 0.01$, 10 Quadratic Elements

Method	Time (sec.)
Galerkin	11.6464
SU	11.0252
SUPG	10.7440
GLS	13.9447

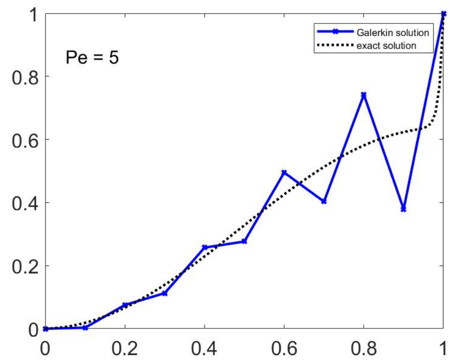
Table 1: Prob. 3, $a = 1$, $\nu = 0.01$, 10 Quadratic Elements computing time (CPU Intel i7 6700 HQ @ 2.60 GHz RAM 16 GB)



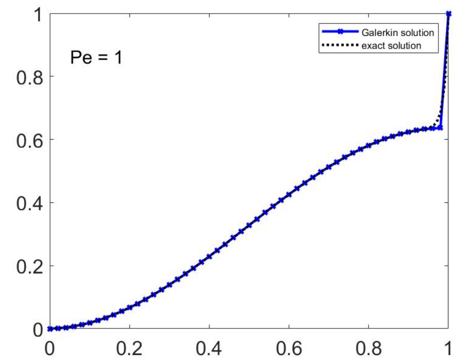
(a) $a = 1$, $\nu = 0.2$, 10 elements



(b) $a = 20$, $\nu = 0.2$, 10 elements

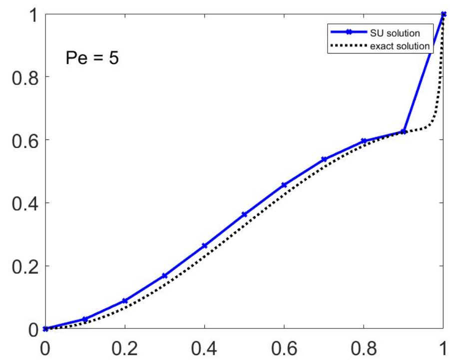


(c) $a = 1$, $\nu = 0.01$, 10 elements

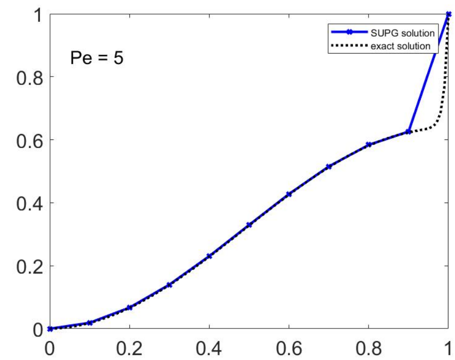


(d) $a = 1$, $\nu = 0.01$, 50 elements

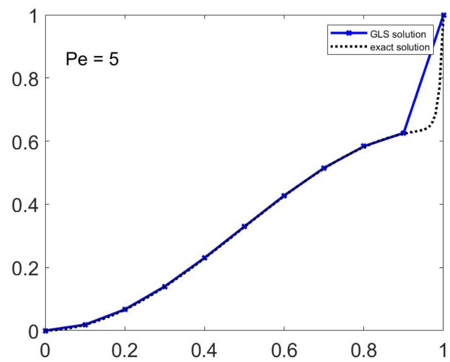
Figure 5: Prob. 4, $u(0) = 0$ $u(1) = 1$ $s = sin(\pi x)$, Linear Element, Galerkin



(a) SU

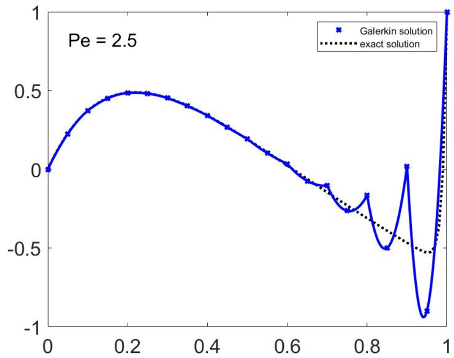


(b) SUPG

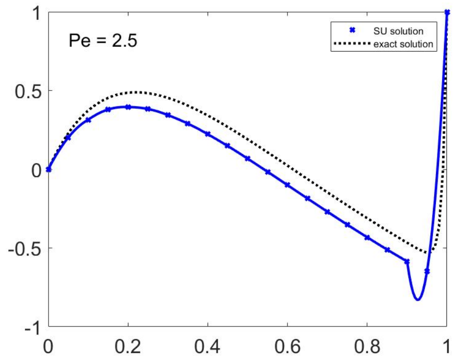


(c) GLS

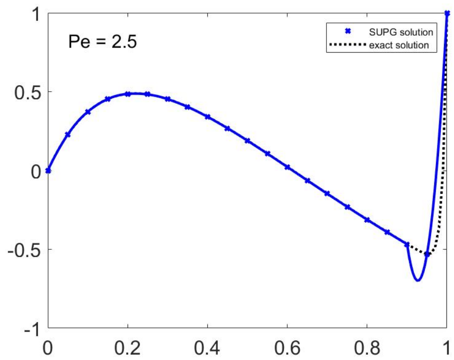
Figure 6: Prob. 4, $a = 1$, $\nu = 0.01$, 10 Linear Elements



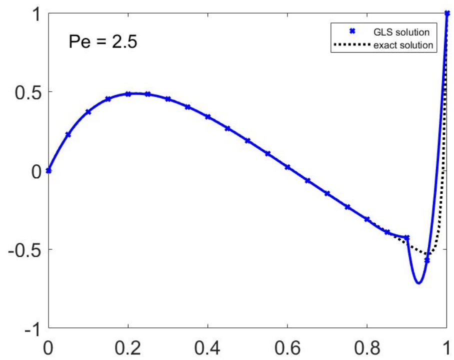
(a) Galerkin



(b) SU



(c) SUPG



(d) GLS

Figure 7: Prob. 3, $a = 1$, $\nu = 0.01$, 10 Quadratic Elements