Basque Center for Applied Mathematics

NUMERICAL APPROXIMATION OF FLUID PROBLEMS

(september 2011)

Hoja 1

Poisson equation in dimension $d = 1$: Finite elements method

We consider the Poisson equation in an interval with Dirichlet boundary conditions

$$\begin{cases} -u''(x) + c(x)u(x) = f(x), & a < x < b, \\ u(a) = \alpha, \quad u(b) = \beta. \end{cases} \tag{1}$$

We show how to approximate the solution and we apply the method to the particular case:

$$a = 0, \quad b = 2, \quad u(0) = 10, \quad u(2) = \frac{10e^2}{21}, \quad c(x) = \frac{2}{(x+0.1)^2} \text{ y } f(x) = \frac{2e^x}{(x+0.1)^2} - \frac{e^x}{x+0.1}.$$

We first state a general abstract setting for solving such problems. Then we introduce a suitable discrete approximation and finally we show an algorithm to solve it.

A. Abstract setting. Let H be a Hilbert space and $a(u, v)$ a bilinear form on H satisfying the following conditions:

1. **Continuity**. There exists $C > 0$ such that $|a(u, v)| \leq C|u|\ |v|$, for all $u, v \in H$,
2. **Coerciveness**. There exists $\alpha > 0$ such that $a(u, u) \geq \alpha |u|^2$, for all $u \in H$.

For $f \in H'$, the dual space of H, we want to approximate the solution of the following problem:

$$\begin{cases} u \in H, \\ a(u, v) = (f, v) \quad \forall v \in H. \end{cases} \tag{2}$$

Note that the existence and uniqueness of this abstract problem can be deduced from the Lax-Milgran Theorem.

To write our example in this abstract setting we intoduce the variational formulation of (1). This is done in two steps:

1. Homogenize the problem: We transform the original problem in a new equivalent one with homogeneous boundary conditions. To do that we introduce a function $g(x)$ with the following properties:

   $$g(a) = \alpha, \quad g(b) = \beta.$$

   In this way, we can write

   $$u = v + g, \quad v \in H_0^1(a, b)$$

   and the problem is reduced to find $v \in H_0^1(a, b)$ such that

   $$\begin{cases} -v''(x) + c(x)v(x) = \hat{f}(x), & a < x < b, \\ v(a) = 0, \quad v(b) = 0, \end{cases}$$

   where $\hat{f} = f - (-g'' + cg)$

2. Variational formulation: *Find $v \in H_0^1(a,b)$ such that*

$$\int_a^b v'w'dx + \int_a^b cvwdx = \int_a^b \hat{f}wdx, \quad \forall w \in H_0^1(a,b).$$

Therefore, in our case the variational formulation of the problem can be written in the abstract setting by chosing $H = H_0^1(a,b)$,

$$a(u,v) = \int_a^b u'v'dx + \int_a^b cuvdx, \quad (f,v) = \int_a^b \hat{f}vdx.$$

B. Discrete problem. A natural discrete approximation of the abstract setting (2) is to replace the infinite dimensional space H by a finite dimensional subspace $V_h \subset H$ depending on a small discretization parameter $h > 0$. This is the idea of Galerkin method. The discrete approximation reads

$$\begin{cases} u_h \in V_h, \\ a(u_h, v_h) = (f, v_h) \quad \forall v_h \in V_h. \end{cases} \tag{3}$$

The key property of the Galerkin approach is that the error is orthogonal to the chosen subspace V^h. This allows to obtain the following estimate which is known as Céa lemma,

$$|u - u_h| \leq \frac{C}{\alpha} \inf_{v \in V_h} |u - v|. \tag{4}$$

Thus, to obtain the convergence $u_h \to u$ as $h \to 0$ it is natural to assume a certain convergence of V_h to H. The required property is that any $v \in V$ can be approximated in V_h, i.e.

$$\forall v \in H, \text{ there exists } v_h \in V_h \text{ such that } \lim_{h \to 0} v_h = v. \tag{5}$$

Under this hipothesis, the convergence $u_h \to u$ is a direct consequence of estimate (4)

In our particular example if we consider a finite dimensional subspace $V^h \subset H_0^1(a,b)$ we obtain the following discrete formulation: *Find $v^h \in V^h$ such that*

$$\int_a^b (v^h)'(w^h)'dx + \int_a^b cv^h w^h dx = \int_a^b \hat{f}w^h dx, \quad \forall w^h \in V^h.$$

C. Finite elements. The idea now is to write the finite dimensional approximation in matrix form and select V^h such that V^h approaches H_0^1 as $h \to 0$. We follow the following steps:

1. Matrix formulation. We consider V^h generated by $\{\varphi_j\}_{j=2\ldots n+1}$ that we refer as basis functions. Note that basis functions satisfy $\varphi_j(a) = \varphi_j(b) = 0$. We have started with the index $j = 2$ for convenience.

   Let us also introduce the functions φ_1, φ_{n+2} such that

$$\varphi_1(a) = 1, \quad \varphi_1(b) = 0,$$
$$\varphi_{n+2}(a) = 0, \quad \varphi_{n+2}(b) = 1.$$

   Now, we write

$$u^h = v^h + g^h, \qquad v^h = \sum_{j=1}^{n+2} d_j\varphi_j, \quad g^h = \alpha\varphi_1 + \beta\varphi_{n+2},$$
$$w^h = \sum_{j=1}^{n+2} c_j\varphi_j.$$

(Note that v^h and w^h are zero at the extremes a, b. This means $d_1 = d_{n+2} = c_1 = c_{n+2} = 0$. However, we maintain d_1 y d_{n+2} as unknowns to simplify)

It is convenient to write g^h as

$$g^h = \sum_{j=1}^{n+2} fr_j \varphi_j, \qquad fr = [fr_j] = [\alpha, 0, ..., 0, \beta]^t$$

where fr is boundary vector.

Substituting the values of u^h and w^h in the discrete formulation we obtain:

$$\sum_{i=1}^{n+2} (R_{ji} + M_{ji})\, d_i = F_j - G_j, \quad \forall j \in \{1, \ldots, n+2\}, \tag{6}$$

where

$$R_{ji} = \int_a^b \varphi_j' \varphi_i' dx, \qquad M_{ji} = \int_a^b c\varphi_j \varphi_i dx$$
$$F_j = \int_a^b f\varphi_j dx, \qquad G_j = \sum_{i=1}^{n+2} (R_{ji} + M_{ji})\, fr_i.$$

Therefore, system (7) can be written in matrix form as

$$(R + M)d = F + G$$

where $R = [R_{ji}], M = [M_{ji}], d = [d_1, \ldots, d_{n+2}]^t$, $F = [F_1, \ldots, F_{n+2}]^t$ y $G = [G_1, \ldots, G_{n+2}]^t$. R is the rigidity matrix, M is the mass matrix, F is the vectors of loads, G is the boundary term (coming from homogenizing the problem) and d is the displacement vector.

2. The basis functions. We divide the interval $[a, b]$ in $n + 1$ subintervals of length h that we refer as **elements**. The extremes of the intervals $\{x_j\}_{j=1,\ldots,n+2}$ are usually referred as nodes. We take as basis functions φ_j linear functions on each element,

$$\varphi_j(x_i) = \begin{cases} 0 \text{ si } j \neq i \\ 1 \text{ si } j = i. \end{cases}$$

The restriction of the basis functions to the elements is known as **elementary funtions**. It is not difficult to see that the space V_h generated by the basis functions $\{\varphi_j\}_{j=1,\ldots,n+1}$ satisfies the property (5). Moreover, the following holds: Assume that $f \in L^2(a, b)$, then

$$\|u' - u_h'\|_{L^2} \leq Ch\|f\|_{L^2},$$
$$\|u - u_h\|_{L^2} \leq Ch^2\|f\|_{L^2},$$
$$\|u - u_h\|_{L^\infty} \leq Ch^{3/2}\|f\|_{L^2}.$$

3. Local problem on elements. We restrict the computation in (7) to each element I_e. In this way, if we define n_{el} the number of elements, we have

$$R = \sum_{e=1}^{n_{el}} R^e, \quad R^e = \left[R_{ji}^e\right],$$
$$M = \sum_{e=1}^{n_{el}} M^e, \quad M^e = \left[M_{ji}^e\right],$$
$$F = \sum_{e=1}^{n_{el}} F^e, \quad F^e = \left[F_1^e, \ldots, F_{n+2}^e\right]^t$$

where

$$R_{ji}^e = \int_{I_e} \varphi_j' \varphi_i' dx, \qquad M_{ji}^e = \int_{I_e} c\varphi_j \varphi_i dx, \qquad F_j^e = \int_{I_e} f\varphi_j dx.$$

The nonzero components of R^e, M^e and F^e are those that correspond to the elementary functions that are nonzero in the element e. Therefore, to construct R^e y F^e we construct previously the nonzero components of r^e, m^e and f^e, and then we introduce them in a global matrix. Note that r^e and m^e are 2×2 matrixes and f^e is a vector 2×1 since in each element only two elementary functions are non-zero.

4. Asembling the matrix $R+M$ and the vector F from the elementary matrixes r^e, m^e and f^e. To do that we use a collocation matrix that tells to each element the position where the components r^e, m^e and f^e must be included in the global matrix $R+M$ and F respectively.
5. Compute $G = (R+M)fr$
6. Impose the homogeneous Dirichlet boundary condition. To this end, we substitute in the system $(R+M)d = F+G$ the equations corresponding to the boundary nodes (in this case $i = 1, n+2$) by the equations $d_i = 0$. The new system is the modified system.
7. Solve the modified system.
8. The solution is given by $d + fr$

B. Algorithm. We consider the following steps:

1. Divide the domain in subintervals (the elements) with the same length. From the initial parameters a, b and n (number of interior nodes) compute:
   (a) *nel* number of elements.
   (b) *coorx* coordenates x of the nodes (from left to right)
   (c) *LM* collocation matrix. Asign to each node of each element the position in the global matrix. The matrix *LM* has the same number of columns as elements. In each column will appear the position of each one of the nodes of the element in the global matrix. En the global matrix the row i corresponds to the equation for the node i.
   Example: if $n = 3$, there are 5 nodes and 4 elements. In this case the global matrix will be 5×5 and *LM* is
   $$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 5 \end{pmatrix}$$
2. Construct the vector '*fr*' that asigns to each node in the boundary the value of the boundary condition. In the interior points 'fr' takes zero values.
3. Elements loop. In this part we fill the global matrixes '*R*', mass matrix '*M*' and second hand term '*F*' in such a way that the problem is reduced to solve the linear system $(R+M)d = F$. We do as follows:
   (a) Compute the function $f(x, y)$ in the quadrature points of the element we are considering: '*funce*'.
   (b) Compute the rigidity matrix in each element '*rele*'

(c) Compute the mass matrix in each element '$mele$'

(d) Compute the second hand term on each element using a suitable quadrature formula '$fele$'.

(e) include the elements data in the global matrixes: 'R' 'M' and 'F'. We need to use the collocation matrix 'LM'

(f) END

4. Homogenization of the problem. Add the boundary contribution to the second hand term: '$F = F - (R + M) * fr$'

5. Impose the homogeneous boundary condition. To do that we change the equation in the boundary nodes by: $d(i) = 0$. We have to substitute the rows i of the matrix $(R + M)$ corresponding to boundary nodes by rows in which all the terms are zero. The only nonzero term will be the i where we put the value 1. In the second hand term we also impose the value 0. We still refer as $(R + M)$ the new matrix and F the new second hand term.

6. Solve the system $(R + M)d = F$.

7. Add the contribution coming from the boundary: $d = d + fr$

8. Plot the solution.

Exercise 1 Consider the Poisson equation in an interval with mixed boundary conditions

$$\begin{cases} -u''(x) + c(x)u(x) = f(x), & a < x < b, \\ u(a) = \alpha, \quad u'(b) = \beta. \end{cases}$$

Solve the system in the particular case:

$$a = 0, \quad b = 2, \quad u(0) = 10, \quad u'(2) = e^2\left(\frac{10}{21} - \frac{10^2}{21^2}\right), \quad c(x) = \frac{2}{(x+0.1)^2} \text{ y } f(x) = \frac{2e^x}{(x+0.1)^2} - \frac{e^x}{x+0.1}.$$

Basque Center for Applied Mathematics
NUMERICAL APPROXIMATION OF FLUID PROBLEMS
(september 2011)
Hoja 2

Poisson equation in dimension $d = 2$: Finite elements

We consider the Poisson equation in a rectangle with Dirichlet boundary conditions

$$\begin{cases} -\Delta u(x,y) = f(x,y), & a < x < b, \quad c < y < d, \\ u(a,y) = g_1(y), \quad u(b,y) = g_2(y), & c < y < d, \\ u(x,c) = g_3(x), \quad u(x,d) = g_4(x), & a < x < b, \end{cases}$$

We approximate the solution by the finite elements method. We will consider the following particular problem:

$$a = c = 0, \quad b = d = 1, \quad f(x,y) = -6xy^2 - 2x^3 + 6xy,$$
$$u(x,0) = 0, \quad u(x,1) = x^3 - x,$$
$$u(0,y) = 0, \quad u(1,y) = y^2 - y^3,$$

Try the following steps:

A. **Discretization**. Follow the one-dimensional case to find a discrete formulation. Consider as **elements** rectangles of the same size and as **basis functions** those that are linear on each element for each variable x and y.

B. **Algorithm.** Follow the following steps:

1. Divide the domain in elements. With the initial parameters a, b, c, d, n (number of interior nodes in the variable x) y m (number of interior points in the variable y) compute:
   (a) *nel* number of elements.
   (b) *nen* number of nodes in each element
   (c) *nnf* number of nodes that are not in the boundary
   (d) *coorx* coordenates x of the nodes (from left to right and from the top to bottom)
   (e) *coory* coordenates y of the nodes
   (f) FR vector with a reference for the boundary nodes. In the interior nodes is 0.
   (g) IEN matrix nen×nel with such as many columns as elements. In each column we write the nodes of the corresponding element in counterclockwise order.
2. Compute the solution in the boundary. Construct a vector 'solf' which contains the values of the solution in the nodes of the boundary. In the interior nodes this vector 'solf' will be zero.
3. Consider a quadrature formula for the integrals (for example, Gauss quadrature formula with 4 interior points). Every integral is made in the reference element $[-1,1] \times [-1,1]$ and therefore we take a formula of the type

$$\int_{[-1,1]\times[-1,1]} f(\xi,\eta)dx = \sum_{i=1}^{nint} f(\xi_i,\eta_i)c_i.$$

   Here *nint* is the number of quadrature points, (ξ_i, η_i) the quadrature points and c_i the coefficients. We denote by (ξ, η) the coordinates in the square of reference. Therefore, we need the following data:

(a) *nint* number of quadrature points.
(b) *coxcua* coordinates ξ of the quadrature points ξ_i.
(c) *coycua* coordinates η of the quadrature points η_i.
(d) *coefc* coefficients.
(e) FEC (nel × nint) value of the elementary functions in the quadrature points.
(f) $FEC1$ (nel × nint) value of the derivative with respect to ξ of the elementary functions in the quadrature points.
(g) $FEC2$ (nel × nint) value of the derivative with respect to η of the elementary functions in the quadrature points.

4. Elements loop. In this part we fill the global matrix 'R+M' and the second hand term 'F' in such a way that the problem is reduced to solve the linear system $(R+M)d = F$. We do the following:

   (a) Compute the jacobian of the transformation $(x, y) \rightarrow (\xi, \eta)$ in the quadrature points. Take into account that this transformation can be written in terms of the elementary functions $\phi_{e,j}(\xi, \eta)$ as:
   $$x(\xi, \eta) = \sum_{j=1}^{nen} \phi_{e,j}(\xi, \eta) x_{e,j}, \quad y(\xi, \eta) = \sum_{j=1}^{nen} \phi_{e,j}(\xi, \eta) y_{e,j}$$
   where $(x_{e,j}, y_{e,j})$ are the coordinates of the node j associated to the element e. We have to compute:
      i. *xchi* derivative of x with respect to ξ in the quadrature points.
      ii. *xeta* derivative of x with respect to η in the quadrature points.
      iii. *ychi* derivative of y with respect to ξ in the quadrature points.
      iv. *yeta* derivative of y with respect to η in the quadrature points.
      v. *jac* jacobian of the transformation in the quadrature points: $(x_\xi y_\eta - x_\eta y_\xi)^{-1}$.

   (b) Compute the derivatives of the elementary functions with respect to x and y: $FEE1$ y $FEE2$. To this end we use the derivatives of the shape functions with respect to (ξ, η) and the equations of the transformation $(x, y) \rightarrow (\xi, \eta)$.
   (c) Compute the function $f(x, y)$ in the quadrature points: '$funce$'.
   (d) Compute the solution in the boundary nodes of the element (if it has) $solfre$.
   (e) Compute the rigidity matrix in the element '$rele$'
   (f) Compute the mass matrix in each element '$mele$'
   (g) Compute the second hand term in each element using the quadrature formula '$fele$'.
   (h) Fill the global matrixes with the information coming from the elements: '$R+M$' y 'F'. We need the colocation matrix 'LM'
   (i) END

5. Homogenize the problem: $F = F - (R+M) * solf$

6. Impose boundary conditions: $d(i) = 0$ in the boundary nodes i.

7. Solve the system $(R+M)d = F$.

8. Add the contribution coming from the boundary: $d = d + solf$

9. Plot the solution.

Exercise 1 Do the same as before with triangles as elements.

Exercise 2 In this exercise we use the triangulation coming from MATLAB to solve the problem above. We first create a file with the domain data:

1. Use the command 'pdetool' to plot the domain.
2. Export the parameters of the geometry (menú Draw).
3. Write in command line

   dl= decsg(gd,sf,ns) ;

   fid=wgeom(dl,'geom') ;

   In this way we create a file 'geom.m' with the information of the geometry.

Once this is done we can triangulate the geometry in 'geom.m' with
[p,e,t]=initmesh('geom','hmax',.6);
In this way we create 3 matrixes 'p','e' and 't' with the mesh data. The number 0.6 is the maximun size for the triangles.

To draw the mesh use the command **pdemesh(p,e,t)**

Now we have to adapt the data in 'p', 'e' and 't' to solve our problem with the previous procedure.

Basque Center for Applied Mathematics
NUMERICAL APPROXIMATION OF FLUID PROBLEMS
(september 2011)
Hoja 3

Evolution equations: Heat equation

Here we consider the heat equation in an interval with homogeneous boundary conditions,

$$\begin{cases} u_t(t,x) - u_{xx}(t,x) = 0, \quad a < x < b, \\ u(t,a) = u(t,b) = 0, \\ u(0,x) = u_0(x). \end{cases}$$

We show how to approximate the solution using the finite elements method in space and a suitable time-integration.

We apply the method to the particular case:

$$a = 0, \quad b = 2, u_0(x) = \frac{2e^x}{(x+0.1)^2} - \frac{e^x}{x+0.1}.$$

A. Discrete problem. We consider the following steps:

1. Variational formulation: Find $u(t,x) \in C^1(0,T;H_0^1(a,b))$ such that

$$\int_a^b u_t w \, dx + \int_a^b u_x w_x dx = 0, \quad \forall w \in H_0^1(a,b).$$

   and the initial condition $u(0,t) = u_0(x)$.

2. Galerkin method. Consider a finite dimensional subspace $V^h \subset H_0^1(a,b)$. The discrete formulation of the problem is: Find $u^h \in V^h$ such that

$$\int_a^b u_t^h w^h \, dx + \int_a^b u_x^H w_x^h dx = 0, \quad \forall w^h \in V^h.$$

   We assume that V^h is generated by the basis functions $\{\varphi_j\}_{j=2\ldots n+1}$. We also introduce φ_1, φ_{n+2} such that

$$\varphi_1(a) = 1, \quad \varphi_1(b) = 0,$$
$$\varphi_{n+2}(a) = 0, \quad \varphi_{n+2}(b) = 1.$$

   Now we write

$$u^h(t) = \sum_{j=1}^{n+2} d_j(t)\varphi_j,$$
$$w^h = \sum_{j=1}^{n+2} c_j\varphi_j.$$

   Sustituting the values of u^h y w^h in the discrete formulation we obtain:

$$\sum_{i=1}^{n+2} \left(R_{ji}d_i(t) + M_{ji}d_i'(t)\right) = 0, \quad \forall j \in \{1,\ldots,n+2\}, \tag{7}$$

where

$$R_{ji} = \int_a^b \varphi_j' \varphi_i' dx, \qquad M_{ji} = \int_a^b \varphi_j \varphi_i dx$$

Then, (7) can be written in matrix form as

$$Rd + Md' = 0, \qquad d(0) = d_0.$$

where $R = [R_{ji}], M = [M_{ji}], d = [d_1, \ldots, d_{n+2}]^t$. R is the rigidity matrix and M the mass matrix

3. Time integration. We divide the time interval $[0, T]$ in subintervals of length Δt. We consider the following methods

   (a) Euler method:
   $$Mu^{n+1} - Mu_l^n + \Delta t R u^n = 0$$

   (b) Crank-Nicolson:
   $$(M + \Delta t/2K)u^{n+1} - (M - \Delta t/2K)Ku^n = 0$$

   (c) Adams-Bashforth:
   $$Mu^{n+2} = Mu^{n+1} + \frac{3}{2}\Delta t K u^{n+1} - \frac{1}{2}Ku^n$$

Exercise 1 Consider the heat equation

$$\begin{cases} \frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}, & 0 < x < 1, \quad t > 0, \\ u(x,0) = u_0(x), & 0 < x < 1, \\ u(0,t) = g_1(t), & t > 0, \\ u(1,t) = g_2(t), & t > 0, \end{cases}$$

Approximate the solution at $T = 1$ by different methods in the particular case,

$$u_0(x) = \sin\frac{\pi x}{2} + \frac{1}{2}\sin(2\pi x), \quad g_1(t) = 0, \quad g_2(t) = \exp(-\frac{\pi^2 t}{4}).$$

In this case, the exact solution is given by

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

Draw the solution and the error for different values of $\mu = \Delta t/\Delta x$:

1. $\mu = \frac{1}{2}$ with 20, 40 y 80 interior points in x variable.
2. $\mu = .509 > \frac{1}{2}$ with 20, 40 y 80 interior points in the variable x.