TMA225 Differential Equations and Scientific Computing, part A

NOTES ON TIME DEPENDENT PROBLEMS IN 2D

1. The Model Problem

We first consider the following time dependent model problem,

$$
\begin{aligned}
\dot{u} - \nabla \cdot (a \nabla u) &= f, && x = (x_1, x_2) \in \Omega, \quad 0 < t < T, \\
u(x,t) &= 0, && x = (x_1, x_2) \in \partial\Omega, \quad 0 < t < T, \\
u(x,0) &= u_0(x), && x = (x_1, x_2) \in \Omega,
\end{aligned}
\tag{1}
$$

where $u = u(x,t) = u(x_1,x_2,t)$ is the unknown function that we wish to compute, with time derivative, $\frac{\partial u}{\partial t}$, denoted by $\dot{u}$. We assume that $\Omega \subset \mathbb{R}^2$ has a polygonal boundary. The functions $a = a(x,t)$ and $f = f(x,t)$ are *data* to the problem. We also need to specify *boundary data*: in this model problem we have *homogeneous Dirichlet boundary conditions* ($u = 0$) on the entire boundary $\partial\Omega$, for all times, $0 < t < T$, and *initial data*: $u_0(x)$, which specifies the solution, for $x \in \Omega$, at time $t = 0$.

2. The Numerical Method

We shall construct a numerical method by *first discretizing in space* (using finite elements) to obtain a finite dimensional system of linear, ordinary differential equations, which we finally solve numerically using, e.g., the backward Euler method.

2.1. Space Discretization.

2.1.1. *Variational Formulation.* Multiply the differential equation in (1) by a *test function* $v = v(x_1, x_2)$ such that $v = 0$ on $\partial\Omega$ and integrate over Ω:

$$\iint_\Omega \dot{u} v \, dx_1 dx_2 \; - \; \iint_\Omega \nabla \cdot (a\nabla u) v \, dx_1 dx_2 \; = \; \iint_\Omega f v \, dx_1 dx_2, \quad 0 < t < T.$$

We now integrate by parts (see the notes on *Robin Boundary Conditions in 2D* for details):

$$\iint_\Omega \dot{u} v \, dx_1 dx_2 \; - \; \int_{\partial\Omega} (n \cdot (a\nabla u)) \, v \, ds \; + \; \iint_\Omega a \nabla u \cdot \nabla v \, dx_1 dx_2 \; = \; \iint_\Omega f v \, dx_1 dx_2, \quad 0 < t < T.$$

Since

$$v = 0 \text{ on } \partial\Omega,$$

we obtain

$$\iint_\Omega \dot{u} v \, dx_1 dx_2 \; + \; \iint_\Omega a \nabla u \cdot \nabla v \, dx_1 dx_2 \; = \; \iint_\Omega f v \, dx_1 dx_2, \quad 0 < t < T.$$

We thus state the following *variational formulation* of (1):

Date: May 12, 2003.

Find $u(x_1, x_2, t)$ such that, for every *fixed* t: $u(x_1, x_2, t) \in V_0$, and

$$\iint_\Omega \dot{u} v \, dx_1 dx_2 \; + \iint_\Omega a \nabla u \cdot \nabla v \, dx_1 dx_2 \; = \iint_\Omega f v \, dx_1 dx_2, \quad 0 < t < T, \quad \forall v \in V_0, \tag{2}$$

where V_0 denotes the vector space of functions $v = v(x_1,\, x_2)$ such that $v = 0$ on $\partial\Omega$, that are sufficiently regular for the integrals in (2) to exist.

2.1.2. *Discretization in space.* In order to discretize (2) in space, we introduce the vector space V_{h0} of *continuous, piecewise linear* functions, $v(x_1, x_2)$, on a *triangulation*, $\mathcal{T}_h = \{K_i\}_{i=1}^{\mathrm{ntri}}$, of Ω, with the corresponding set of *internal* nodes, $\mathcal{N}_{h0} = \{N_i\}_{i=1}^{\mathrm{nintnodes}}$, such that $v = 0$ on $\partial\Omega$, and state the following *(space) discrete* counterpart of (2):

Find $U(x_1, x_2, t)$ such that, for every *fixed* t: $U(x_1, x_2, t) \in V_{h0}$, and

$$\iint_\Omega \dot{U} v \, dx_1 dx_2 \; + \iint_\Omega a \nabla U \cdot \nabla v \, dx_1 dx_2 \; = \iint_\Omega f v \, dx_1 dx_2, \quad 0 < t < T, \quad \forall v \in V_{h0}. \tag{3}$$

2.1.3. *Ansatz.* We now seek a solution, $U(x_1, x_2, t)$, to (3), expressed (for every *fixed* t) in the basis of *tent functions* $\{\varphi_i\}_{i=1}^{\mathrm{nintnodes}} \subset V_{h0}$. (Note that only "tents" with "poles" at the internal nodes belong to the basis, since all functions in V_{h0} are zero on the boundary $\partial\Omega$.) In other words, we make the *Ansatz*

$$U(x_1, x_2, t) = \sum_{j=1}^{\mathrm{nintnodes}} \xi_j(t) \varphi_j(x_1, x_2), \tag{4}$$

and seek to determine the (time dependent) coefficient vector

$$\xi(t) = \begin{bmatrix} \xi_1(t) \\ \xi_2(t) \\ \vdots \\ \xi_{\mathrm{nintnodes}}(t) \end{bmatrix} = \begin{bmatrix} U(N_1, t) \\ U(N_2, t) \\ \vdots \\ U(N_{\mathrm{nintnodes}}, t) \end{bmatrix},$$

of nodal values of $U(x_1, x_2, t)$, in such a way that (3) is satisfied.

Consider *very carefully* the structure of $U(x_1, x_2, t)$ in (4): For every *fixed* time, t, we note that $U(x_1, x_2, t)$, as a function of $x = (x_1, x_2)$, is a continuous, piecewise linear function with weights given by $\xi(t)$.

2.1.4. *Construction of space discrete system of ODE.* We substitute (4) into (3),

$$\begin{aligned} &\sum_{j=1}^{\mathrm{nintnodes}} \dot{\xi}_j(t) \left(\iint_\Omega \varphi_j v \, dx_1 dx_2 \right) + \sum_{j=1}^{\mathrm{nintnodes}} \xi_j(t) \left(\iint_\Omega a \nabla \varphi_j \cdot \nabla v \, dx_1 dx_2 \right) = \\ &\iint_\Omega f v \, dx_1 dx_2, \quad 0 < t < T, \quad \forall v \in V_{h0}. \end{aligned} \tag{5}$$

Since $\{\varphi_i\}_{i=1}^{\mathrm{nintnodes}} \subset V_{h0}$ is a *basis* of V_{h0}, (5) is equivalent to

$$\sum_{j=1}^{\text{nintnodes}} \dot{\xi}_j(t) \left(\iint_\Omega \varphi_j \varphi_i \, dx_1 dx_2 \right) + \sum_{j=1}^{\text{nintnodes}} \xi_j(t) \left(\iint_\Omega a \, \nabla\varphi_j \cdot \nabla\varphi_i \, dx_1 dx_2 \right) = \iint_\Omega f \varphi_i \, dx_1 dx_2, \quad 0 < t < T, \quad i = 1, \ldots, \text{nintnodes}, \tag{6}$$

which is an *nintnodes*-dimensional system of linear, ordinary differential equations. Introducing the notation

$$m_{i,j} = \iint_\Omega \varphi_j(x_1, x_2) \varphi_i(x_1, x_2) \, dx_1 dx_2,$$

$$a_{i,j}(t) = \iint_\Omega a(x_1, x_2, t) \, \nabla\varphi_j(x_1, x_2) \cdot \nabla\varphi_i(x_1, x_2) \, dx_1 dx_2,$$

$$b_i(t) = \iint_\Omega f(x_1, x_2, t) \varphi_i(x_1, x_2) \, dx_1 dx_2,$$

we can write the system of linear, ordinary differential equations (6), as (we denote *nintnodes* by *nin*):

$$\left\{ \begin{array}{ccccccccccccc} m_{1,1}\,\dot{\xi}_1(t) & + & \ldots & + & m_{1,\text{nin}}\,\dot{\xi}_{\text{nin}}(t) & + & a_{1,1}(t)\,\xi_1(t) & + & \ldots & + & a_{1,\text{nin}}(t)\,\xi_{\text{nin}}(t) & = & b_1(t), \\ m_{2,1}\,\dot{\xi}_1(t) & + & \ldots & + & m_{2,\text{nin}}\,\dot{\xi}_{\text{nin}}(t) & + & a_{2,1}(t)\,\xi_1(t) & + & \ldots & + & a_{2,\text{nin}}(t)\,\xi_{\text{nin}}(t) & = & b_2(t), \\ \vdots & & \vdots & & \vdots & & \vdots & & \vdots & & \vdots & & \vdots \\ m_{\text{nin},1}\,\dot{\xi}_1(t) & + & \ldots & + & m_{\text{nin},\text{nin}}\,\dot{\xi}_{\text{nin}}(t) & + & a_{\text{nin},1}(t)\,\xi_1(t) & + & \ldots & + & a_{\text{nin},\text{nin}}(t)\,\xi_{\text{nin}}(t) & = & b_{\text{nin}}(t), \end{array} \right.$$

$0 < t < T.$

In *matrix form*, this reads,

$$M\,\dot{\xi}(t) + A(t)\,\xi(t) = b(t), \quad 0 < t < T, \tag{7}$$

where $M = \begin{bmatrix} m_{1,1} & \ldots & m_{1,\text{nin}} \\ \vdots & \ddots & \vdots \\ m_{\text{nin},1} & \ldots & m_{\text{nin},\text{nin}} \end{bmatrix}$ is the *mass matrix*,

$$A(t) = \begin{bmatrix} a_{1,1}(t) & \dots & a_{1,\mathrm{nin}}(t) \\ \vdots & \ddots & \vdots \\ a_{\mathrm{nin},1}(t) & \dots & a_{\mathrm{nin},\mathrm{nin}}(t) \end{bmatrix}$$ is the (possibly time dependent) *stiffness matrix*, and

$$b(t) = \begin{bmatrix} b_1(t) \\ \vdots \\ b_{\mathrm{nin}}(t) \end{bmatrix}$$ is the (possibly time dependent) *load vector*.

Exercise 1. Show that, for the time dependent reaction-diffusion problem with Robin boundary conditions,

$$\begin{aligned} \dot{u} - \nabla \cdot (a\nabla u) + cu &= f, & x &= (x_1, x_2) \in \Omega, & 0 < t < T, \\ -n \cdot (a\nabla u) &= \gamma(u - g_D) + g_N, & x &= (x_1, x_2) \in \partial\Omega, & 0 < t < T, \\ u(x, 0) &= u_0(x), & x &= (x_1, x_2) \in \Omega, \end{aligned}$$

the system (7) generalizes to,

$$M\,\dot{\xi}(t) \;+\; (A(t) \;+\; M_c(t) \;+\; R(t))\,\xi(t) \;=\; b(t) + rv(t), \quad 0 < t < T, \tag{8}$$

where $M_c(t)$ is the *mass matrix* coming from the reactive term, $c(x_1, x_2, t)u(x_1, x_2, t)$, and $R(t)$, $rv(t)$ are the contributions from the Robin boundary conditions to the system matrix and right-hand side, respectively. (Compare with the notes on *Robin Boundary Conditions in 2D*). Note that (8) is an *nnodes*-dimensional system of linear, ordinary differential equations, since in this case we also include the nodes on the boundary $\partial\Omega$.

2.2. **Time Discretization.** In order to discretize (7) in time, we let $0 = t_0 < t_1 < t_2 < \cdots < t_L = T$ be discrete time levels with corresponding time steps $k_n = t_n - t_{n-1}$, $n = 1, \ldots, L$. Further, we let ξ^n denote an *approximation* of $\xi(t_n)$, $n = 1, \ldots, L$.

There are different possible choices of *initial data*, $\xi^0 = \xi(0)$, to (7): the simplest is to let

$$\xi^0 = \begin{bmatrix} \xi_1(0) \\ \xi_2(0) \\ \vdots \\ \xi_{\mathrm{nintnodes}}(0) \end{bmatrix} = \begin{bmatrix} u_0(N_1) \\ u_0(N_2) \\ \vdots \\ u_0(N_{\mathrm{nintnodes}}) \end{bmatrix},$$

which corresponds to letting $U(x_1, x_2, 0) = \sum_{j=1}^{\mathrm{nintnodes}} \xi_j(0)\varphi_j(x_1, x_2)$ be the *nodal interpolant* of $u_0(x_1, x_2) = u(x_1, x_2, 0)$. (An alternative would be to choose $U(x_1, x_2, 0)$ as the $L_2(\Omega)$-projection of u_0, but then we would need to *compute* ξ^0.)

We now *integrate* (7) (element-wise) over one time interval $[t_{n-1},\ t_n]$,

$$\int_{t_{n-1}}^{t_n} M\,\dot{\xi}(t)\,dt \;+\; \int_{t_{n-1}}^{t_n} A(t)\,\xi(t)\,dt \;=\; \int_{t_{n-1}}^{t_n} b(t)\,dt.$$

Since M is a constant matrix, we get,

$$M(\xi(t_n) - \xi(t_{n-1})) \;+\; \int_{t_{n-1}}^{t_n} A(t)\,\xi(t)\,dt \;=\; \int_{t_{n-1}}^{t_n} b(t)\,dt. \tag{9}$$

Given an approximation, ξ^{n-1}, of $\xi(t_{n-1})$, approximating the integrals in (9) using *right end-point quadrature* gives the *backward Euler method* defining ξ^n by,

$$M(\xi^n - \xi^{n-1}) + A(t_n)\xi^n k_n = b(t_n)k_n,$$

i.e.,

$$M\,\frac{\xi^n - \xi^{n-1}}{k_n} + A(t_n)\xi^n = b(t_n).$$

The *backward Euler method* for solving (7) thus becomes: Given $\xi^0 = \xi(0)$, for $n = 1, \ldots, L$, solve the linear system of equations,

$$(M + k_n A_n)\xi^n = M\xi^{n-1} + k_n b_n,$$

where we have introduced the notation

$$A_n = A(t_n), \quad b_n = b(t_n).$$

Exercise 2. Show that the backward Euler method for solving (8) reads: Given $\xi^0 = \xi(0)$, for $n = 1, \ldots, L$, solve the linear system of equations:

$$(M + k_n(A(t_n) + M_c(t_n) + R(t_n)))\xi^n = M\xi^{n-1} + k_n(b(t_n) + rv(t_n)).$$