Section 8 – Partial Differential Equations

Partial differential equations are those that involve more than one independent variable. Because of their widespread application in engineering, we will concentrate on linear, second-order partial differential equations of the form

$$A\frac{\partial^2 u}{\partial x^2} + B\frac{\partial^2 u}{\partial x \partial y} + C\frac{\partial^2 u}{\partial y^2} + D = 0$$

where A, B and C are in general functions of x and y, and D can be a function of x, y, $\partial u/\partial x$ and $\partial u/\partial y$.

If B^2-$4AC$ < 0, the equation is *elliptic*, such as the Laplace Equation

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0$$

which describes a steady-state condition in two dimensions. If B^2-$4AC$ = 0, the equation is *parabolic,* such as the heat conduction equation

$$\frac{\partial T}{\partial t} = k'\frac{\partial^2 T}{\partial x^2}$$

If B^2-$4AC$ > 0 then the equation is *hyperbolic*, such as the wave equation

$$\frac{\partial^2 y}{\partial y^2} = \frac{1}{c}\frac{\partial^2 y}{\partial t^2}$$

7.1 The Laplace Equation

Consider a thin plate of thickness Δz. The temperature of the plate at any position (x, y) can be shown to be an instance of the Laplace equation

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0$$

or with source/sink $f(x, y)$ it is Poisson's equation

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = f(x, y)$$

The component of the heat flux vector in the *i* direction is given by Fourier's Law

$$q_i = -k\rho c \frac{\partial T}{\partial i}$$

(where i is x or y) and describes the flow of heat in any direction, k is the coefficient of thermal diffusivity, ρ is the density, c is the heat capacity, T is the temperature, $T = H/\rho cV$ where H is heat and V is volume. Also, $k' = k\rho c$.

The Laplace equation can be approximated through finite differences. Compute the second derivatives as $O(h^2)$

$$\frac{\partial^2 T}{\partial x^2} \approx \frac{T_{i+1,j} - 2T_{i,j} + T_{i-1,j}}{\Delta x^2} \quad \text{and} \quad \frac{\partial^2 T}{\partial y^2} \approx \frac{T_{i,j+1} - 2T_{i,j} + T_{i,j-1}}{\Delta y^2}$$

And Laplace's equation becomes

$$\frac{T_{i+1,j} - 2T_{i,j} + T_{i-1,j}}{\Delta x^2} + \frac{T_{i,j+1} - 2T_{i,j} + T_{i,j-1}}{\Delta y^2} = 0$$

If $\Delta x = \Delta y$, this is the *Laplacian difference equation*:

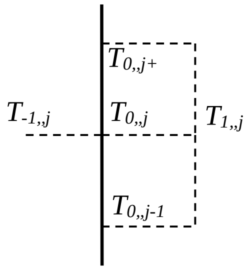
$$T_{i+1,j} + T_{i-1,j} + T_{i,j+1} + T_{i,j-1} - 4T_{i,j} = 0$$

This can be applied at all of the interior nodes of the plate. In addition, boundary conditions must be specified at all the boundary points in order to obtain a unique solution.

If the temperature at a boundary point is known (Dirichlet boundary condition), then the temperature at that point is simply set to that value and moved to the right hand side of the equation. If the derivative of the temperature, or the heat flux, is known (Neumann boundary condition), for example

$$\frac{\partial T}{\partial x} \approx \frac{T_{1,j} - T_{-1,j}}{2\Delta x} \quad \text{or} \quad T_{-1,j} = T_{1,j} - 2\Delta x \frac{\partial T}{\partial x}$$

So that, writing the Laplacian difference equation at $T_{0,j}$ gives



$$T_{1,j} + \left(T_{1,j} - 2\Delta x \frac{\partial T}{\partial x}\right) + T_{0,j+1} + T_{0,j-1} - 4T_{0,j} = 0$$

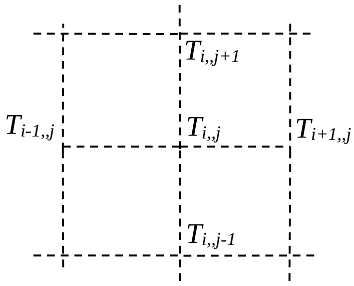
or

$$2T_{1,j} + T_{0,j+1} + T_{0,j-1} - 4T_{0,j} = 2\Delta x \left.\frac{\partial T}{\partial x}\right|_0$$

Finite-Difference Equations, $\Delta x = \Delta y$

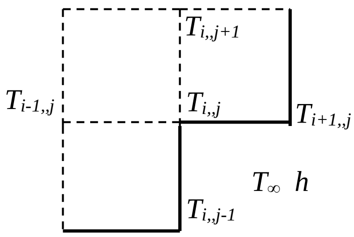
Case 1: Interior node

$$T_{i+1,j} + T_{i-1,j} + T_{i,j+1} + T_{i,j-1} - 4T_{i,j} = 0$$



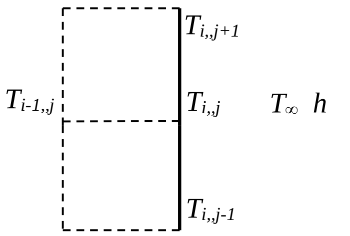
Case 2: Node at internal corner with convection

$$2(T_{i-1,j} + T_{i,j+1}) + (T_{i+1,j} + T_{i,j-1}) + 2\frac{h\Delta x}{k}T_\infty - 2(3 + \frac{h\Delta x}{k})T_{i,j} = 0$$

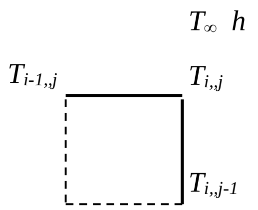


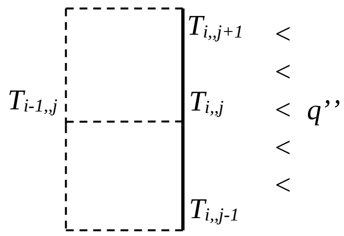
Case 3: Node at plane surface with convection

$$(2T_{i-1,j} + T_{i,j+1} + T_{i,j-1}) + \frac{2h\Delta x}{k}T_\infty - 2(\frac{h\Delta x}{k} + 2)T_{i,j} = 0$$



<table>
<tr>
<td>Case 4: Node at external corner with convection

$$(T_{i-1,j} + T_{i,j-1}) + \frac{2h\Delta x}{k} T_\infty - 2(\frac{h\Delta x}{k} + 1) T_{i,j} = 0$$
</td>
<td>
</td>
</tr>
<tr>
<td>Case 5: Node at plane surface with uniform heat flux

$$(2T_{i-1,j} + T_{i,j+1} + T_{i,j-1}) + \frac{2q''\Delta x}{k} - 4T_{i,j} = 0$$
</td>
<td>
</td>
</tr>
</table>

In Case 3 and Case 5 above, to model adiabatic surfaces or surfaces of symmetry set *h* or *q''* equal to zero.

7.1.1 Solution Methods

Most numerical solutions of the Laplace equation involve systems that are large – a 5x5 grid with Dirichlet boundary conditions yields 9 equations in 9 unknowns; a 10x10 grid yields 64 equations. A significant number of entries of the matrix are zero, making general methods inefficient for storage and solution. Gauss-Seidel is popular (applied to Laplace's equation, Gauss-Seidel is called the *Liebmann method*). Convergence is assumed because of the strong diagonal dominance of the Laplace difference equation. Over-relaxation is often used to accelerate convergence.

7.1.2 Secondary Variables

The primary variable for the Laplace equation is often not the quantity of interest. Here we solve for temperature but often the heat flux is of more importance. Once the temperature distribution is known the heat flux can be found at any point via central-difference approximations:

$$q_x = -k' \frac{T_{i+1,j} - T_{i-1,j}}{2\Delta x} \quad \text{and} \quad q_y = -k' \frac{T_{i,j+1} - T_{i,j-1}}{2\Delta y}$$

$$q_i = \sqrt{q_x^2 + q_y^2} \quad ; \quad \theta = \tan^{-1}(q_y / q_x)$$

Care must be taken to consider the signs of q_y and q_x to find θ in the proper quadrant.

7.1.3 Irregular Boundary Conditions

This general technique of approximating partial differential equations is fairly straightforward for orthogonal grids and constant boundary conditions. If the boundaries of the domain are curved, one can formulate the boundary conditions in terms of variable grid spacing that closely matches the shape of the boundary. Derivative boundary conditions, in particular, are difficult to express over curved boundaries. Often very fine grids are employed instead of writing the derivative conditions for curved boundaries.

7.2 Parabolic Equations

A common example of parabolic partial differential equations is the heat conduction equation

$$k\frac{\partial^2 T}{\partial x^2} = \frac{\partial T}{\partial t}$$

As with elliptic equations, finite difference approximations can be substituted for the partial derivatives. However, now we have to account for changes in both space and time. The spatial second derivative can be approximated as

$$\frac{\partial^2 T}{\partial x^2} \approx \frac{T_{i+1}^l - 2T_i^l + T_{i-1}^l}{(\Delta x)^2}$$

The superscripts in this notation denote time. This approximation has error $O(\Delta x^2)$. The time derivative can be approximated with a forward difference (Euler's method)

$$\frac{\partial T}{\partial t} \approx \frac{T_i^{l+1} - T_i^l}{\Delta t}$$

With error $O(\Delta t)$. Substituting into the heat conduction equation gives

$$k\frac{T_{i+1}^l - 2T_i^l + T_{i-1}^l}{(\Delta x)^2} = \frac{T_i^{l+1} - T_i^l}{\Delta t}$$

or

$$T_i^{l+1} = T_i^l + \lambda(T_{i+1}^l - 2T_i^l + T_{i-1}^l)$$

where $\lambda = k\Delta t/(\Delta x)^2$. This equation can be applied to all of the interior nodes and provides an explicit way to compute the values at each node at a future time based on the present values at neighboring nodes.

Convergence and Stability

Convergence is defined as the condition where as Δx and Δt approach zero, the approximation becomes closer to the true solution. Stability is defined as the condition where errors at any stage of the calculation are not amplified but rather are attenuated as the computation progresses. To ensure both, $\lambda \leq 1/2$ or $\Delta t \leq (\Delta x)^2/(2k)$. However, using a time step this large often leads to oscillation, which can usually be avoided by the limitation $\lambda \leq 1/4$. To minimize the truncation error, it is best that $\lambda \leq 1/6$. Using a step size this small, however, greatly increases the computational burden.

Derivative Boundary Conditions

As for elliptic equations, imaginary points outside of the domain can be used with symmetry arguments to successfully model derivative boundary conditions, for example

$$T_0^{l+1} = T_0^l + \lambda(T_1^l - 2T_0^l + T_{-1}^l)$$

Parabolic Equations in Two Dimensions

$$\frac{\partial T}{\partial t} = k\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\right)$$

This equation gives the temperature distribution of the face of a heated plate. Substitute finite difference approximations as above. The convergence and stability condition is now

$$\Delta t \leq \frac{1}{8}\frac{(\Delta x)^2 + (\Delta y)^2}{k}$$

7.2.1 Implicit Methods

Implicit methods approximate the spatial derivative at the future time l+1, with error $O(\Delta x^2)$

$$\frac{\partial^2 T}{\partial x^2} \approx \frac{T_{i+1}^{l+1} - 2T_i^{l+1} + T_{i-1}^{l+1}}{(\Delta x)^2}$$

So that

$$k\frac{T_{i+1}^{l+1}-2T_i^{l+1}+T_{i-1}^{l+1}}{(\Delta x)^2}=\frac{T_i^{l+1}-T_i^l}{\Delta t}$$

or

$$-\lambda T_{i-1}^{l+1}+(1+2\lambda)T_i^{l+1}-\lambda T_{i+1}^{l+1}=T_i^l \quad ; \quad \lambda=k\frac{\Delta t}{(\Delta x)^2}$$

This equation applies to all interior nodes. The first and last nodes must account for the boundary conditions. If the temperature is given at the first and last nodes,

$$T_0^{l+1}=f_0(t^{l+1}) \quad \text{and} \quad T_m^{l+1}=f_m(t^{l+1})$$

This gives *m* equations in *m* unknowns at each time step. The equations are tridiagonal and symmetric, so they can be solved very efficiently. Gradient boundary conditions, however, result in a banded set of equations that may not be tridiagonal nor symmetric.

Crank-Nicholson Method

The Crank-Nicholson method is an implicit method that is second-order accurate in both space and time. In this method, the difference approximations are computed at the midpoint of each time interval. To do this, first approximate the time derivative as usual

$$\frac{\partial T}{\partial t}=\frac{T_i^{l+1}-T_i^l}{\Delta t}$$

then calculate the spatial derivative at the midpoint by averaging the approximations at the beginning and end of the time increment

$$\frac{\partial^2 T}{\partial x^2}=\frac{1}{2}\left[\frac{T_{i+1}^l-2T_i^l+T_{i-1}^l}{(\Delta x)^2}+\frac{T_{i+1}^{l+1}-2T_i^{l+1}+T_{i-1}^{l+1}}{(\Delta x)^2}\right]$$

to give

$$-\lambda T_{i-1}^{l+1}+2(1+\lambda)T_i^{l+1}-\lambda T_{i+1}^{l+1}=-\lambda T_{i-1}^l+2(1+\lambda)T_i^l-\lambda T_{i+1}^l$$

where $\lambda=k\frac{\Delta t}{(\Delta x)^2}$

This equation applies to all of the interior nodes; boundary conditions take care of the first and last nodes. Even though this formulation is more complicated than the simple implicit method, it is much more accurate in practice.

Implicit Methods in Two Dimensions

The *Alternating Direction Implicit* (ADI) method provides a way to solve parabolic equations in two dimensions using tridiagonal matrices for Dirichlet boundary conditions. To do this, execute each time increment in two half-steps. For the first half-step, approximate the heat conduction equation with $\partial^2 T/\partial x^2$ expressed explicitly and $\partial^2 T/\partial y^2$ written implicitly:

$$\frac{T_{i,j}^{l+1/2} - T_{i,j}^{l}}{\Delta t/2} = k\left[\frac{T_{i+1,j}^{l} - 2T_{i,j}^{l} + T_{i-1,j}^{l}}{(\Delta x)^2} + \frac{T_{i,j+1}^{l+1/2} - 2T_{i,j}^{l+1/2} + T_{i,j-1}^{l+1/2}}{(\Delta y)^2}\right]$$

or, with $\Delta x = \Delta y$

$$-\lambda T_{i,j-1}^{l+1/2} + 2(1+\lambda)T_{i,j}^{l+1/2} - \lambda T_{i,j+1}^{l+1/2} = -\lambda T_{i-1,j}^{l} + 2(1+\lambda)T_{i,j}^{l} - \lambda T_{i+1,j}^{l}$$

The second half-step is taken implicitly in $\partial^2 T/\partial x^2$ and explicitly in $\partial^2 T/\partial y^2$:

$$\frac{T_{i,j}^{l+1} - T_{i,j}^{l+1/2}}{\Delta t/2} = k\left[\frac{T_{i+1,j}^{l+1} - 2T_{i,j}^{l+1} + T_{i-1,j}^{l+1}}{(\Delta x)^2} + \frac{T_{i,j+1}^{l+1/2} - 2T_{i,j}^{l+1/2} + T_{i,j-1}^{l+1/2}}{(\Delta y)^2}\right]$$

or, with $\Delta x = \Delta y$

$$-\lambda T_{i-1,j}^{l+1} + 2(1+\lambda)T_{i,j}^{l+1} - \lambda T_{i+1,j}^{l+1} = -\lambda T_{i,j-1}^{l+1/2} + 2(1+\lambda)T_{i,j}^{l+1/2} - \lambda T_{i,j+1}^{l+1/2}$$

Both of these half-steps results in tridiagonal systems of equations for Dirichlet boundary conditions, and the solution process is efficient.

7.3 Finite-Element Method

In finite-difference methods, the domain is divided into a grid of discrete points called nodes, and the PDE is written for each node and its neighboring nodes. While this pointwise approach is conceptually easy, it is difficult to apply to systems with irregular geometry or unusual boundary conditions.

As opposed to finite-difference methods, *finite-element methods* divide the domain into similarly-shaped regions, or *elements*, and an approximate solution for the PDE is developed for each of the elements. The total solution is generated by linking together, or *assembling*, the individual solutions, taking care to ensure continuity at the inter-element boundaries.

7.3.1 The General Approach

The following steps are a general outline of the finite-element approach:

Discretization

Discretization involves dividing the solution domain into *finite elements*. The intersection points of the lines that make up the element are called *nodes*, and the sides themselves are called *nodal lines* (in one or two dimensions) or *nodal planes* (in three dimensions).

Element Equations

An approximate solution for the PDE must be obtained for the element. First choose an appropriate function with unknown coefficients to approximate the solution, then evaluate the coefficients to approximate the solution in an optimal manner.

Because they are easy to manipulate, polynomials are often used as approximating functions. As an example, for the one-dimensional case, the simplest alternative is a first-order polynomial:

$$u(x) = a_0 + a_1 x$$

Where $u(x)$ is the dependent variable, a_0 and a_1 are constants and x is the independent variable. The function must pass through the values of $u(x)$ at the end points of the element at x_1 and x_2

$$u(x_1) = a_0 + a_1 x_1$$
$$u(x_2) = a_0 + a_1 x_2$$

Solving these equations gives

$$a_0 = \frac{u(x_1)x_2 - u(x_2)x_1}{x_2 - x_1}$$
$$a_1 = \frac{u(x_2) - u(x_1)}{x_2 - x_1}$$

Collecting terms, these equations can be written as

$$u = N_1 u(x_1) + N_2 u(x_2)$$
$$N_1 = \frac{x_2 - x}{x_2 - x_1}$$
$$N_2 = \frac{x - x_1}{x_2 - x_1}$$

The function u is called an *approximation or shape function*, and N_1 and N_2 are interpolating functions. In this example u is the first order Lagrange interpolating polynomial, and provides the means to calculate any value between $u(x_1)$ and $u(x_2)$ between the nodes. To complete any formulation, derivatives and integrals of the shape functions are needed:

$$\frac{du}{dx} = \frac{dN_1}{dx} u(x_1) + \frac{dN_2}{dx} u(x_2) = \frac{-u(x_1) + u(x_2)}{x_2 - x_1}$$
$$\int_{x_1}^{x_2} u dx = \int_{x_1}^{x_2} [N_1 u(x_1) + N_2 (x_2)] dx = \frac{u(x_1) + u(x_2)}{2} (x_2 - x_1)$$

Once the interpolation function is chosen, the equations governing the behavior of the element are developed, that is, the equation is fit to the solution of the underlying differential equation. Several methods are used depending on the complexity of the differential equation, such as the direct method, the method of weighted residuals and the variational approach, each of which is closely related to curve-fitting.

The resulting equations will often consist of a set of linear algebraic equations that can be expressed in matrix form

$$[k]\{u\} = \{F\}$$

Where k is the element property or stiffness matrix, u is a column vector of unknowns at the nodes and F is a column vector of the effects of external influences applied at the nodes. These equations may be non-linear, but for the vast majority of practical problems linear systems suffice.

Assembly

After the individual element equations are determined, they must be linked together or assembled to characterize the overall behavior of the system. This is done in a manner that requires the solutions for contiguous elements to match so that the unknown values (and sometimes their derivatives) at the common nodes are equivalent, insuring continuous overall solutions. When all of the element equations are assembled, they form the global or *assemblage equations*

$$[k']\{u'\} = \{F'\}$$

Boundary Conditions

The assemblage equations are modified to account for the system boundary conditions. Dirichlet conditions remove known values from the list of unknowns and modify the right-hand side. Gradient boundary conditions impose additional constraints on the solution at two or more nodes. The result is a different, often slightly reduced, set of equations

$$[\overline{k'}]\{\overline{u'}\} = \{\overline{F'}\}$$

Solution

The solution of the equations above can be obtained with many techniques. In particular, if many different sets of external influences are to be applied to the same system, LU decomposition methods are an efficient choice. If the equations can be configured to produce a banded matrix, highly efficient schemes have been developed to provide quick and accurate solutions.

Post-processing

Once the nodal values are found, the results can be presented in tabular or graphical form. In addition, secondary variables (such as heat flux, or mechanical strain or stress) can be calculated and displayed.

Even though the steps above are very general, they are common to all implementations of the finite element method. Specific examples and problems, and much more detail, are left to other courses.