1 Introduction to Transient Heat Transfer

Up to now, we have been dealing exclusively with static problems; i.e. we have been ignoring the transient terms that are present in the physical problems examined. For many situations this is a valid approximation as these transient terms are of a small magnitude in relation to the terms that we have retained. However, in many situations the need to be retained.

Heat Conduction

In the balance of energy equation for heat conduction we need to include the heat capacity term. This gives us the following relation

$$\rho c\dot{T} + \nabla \cdot \boldsymbol{q} = \rho r_s \,, \tag{1.1}$$

where ρ is the density of the material, c is the heat capacity, and we recall that $\boldsymbol{q} = -\boldsymbol{\kappa}\nabla T$ where $\boldsymbol{\kappa}$ is the thermal conductivity tensor. The only change from before is the first term on the left-hand side of Eq. (1.1).

Up to now all terms in Eq. (1.1) that contained a first time derivative were ignored. Clearly, for large problem classes this will not be true. Such problems are classified as being dynamic problems. Within this classification, dynamic problems are sub-classified as either parabolic or hyperbolic. Parabolic problems are characterized by infinite wave speeds; examples include heat conduction and ground water diffusion. Hyperbolic problems are characterized by finite wave speeds; the main example is elastodynamics in any dimension. The two different classifications lead to separate numerical methods of solution – though they both share a similar higher level formulation.

The method we will employ to deal with dynamic problems is known as the time-continuous or semi-discrete method. In this formulation, one constructs the weak form equations as we have done previously. Then one approximates the solution spatially using a finite element method with time as a continuous parameter in the nodal unknowns. Lastly one approximates the solution temporally using a finite difference method.

1.1 Weak Form Equations

To generate the weak form expressions we proceed exactly as before: (1) define a suitable space of trial solutions, (2) define a suitable space of test functions, (3) multiply the governing balance equations by the test function, integrate over the domain, integrate by parts.

Heat Conduction

For the dynamic heat conduction case we will adopt all the previous notational conventions from the static problem except that we will now allow for all boundary conditions to be time varying. This leads to the following definitions for the space of solutions and the space of test functions:

$$\mathcal{V} = \{v(\boldsymbol{x}) \mid v(\boldsymbol{x}) = 0 \text{ for } \boldsymbol{x} \in \Gamma_T\} \tag{1.2}$$

$$\mathcal{S} = \{T(\boldsymbol{x},t) \mid T(\boldsymbol{x},t) = \bar{T}(\boldsymbol{x},t) \text{ for } \boldsymbol{x} \in \Gamma_T \text{ and } T(\boldsymbol{x},0) = \tilde{T}(\boldsymbol{x}) \text{ for } \boldsymbol{x} \in \Omega\}\,, \tag{1.3}$$

where $\tilde{T}(\boldsymbol{x})$ are known initial conditions and $\bar{T}(\boldsymbol{x},t)$ are known boundary data on Γ_T. If we now multiply the balance Equation (1.1) by an arbitrary test function and integrate as before we arrive at

$$\int_\Omega \rho c \dot{T} v + \int_\Omega \nabla T \cdot \boldsymbol{\kappa} \nabla v = \int_\Omega \rho r_s v - \int_{\Gamma_h} h v\,. \tag{1.4}$$

The problem statement reads: Find the unknown function in the space of solutions such that the weak form equation holds for all functions in the space of test functions for all times in the time interval of interest.

1.2 Finite Element Approximation

To create a finite element approximation to the problem outlined above we choose sub-spaces of the space of trial solutions and the space of test functions. The choices are made identical to the static case except that the vector of nodal unknowns for the solution is taken to be time dependent. The vector of nodal values of the test functions remains time independent.

Heat Conduction

For heat conduction we have that

$$T^h(\boldsymbol{x},t) = \sum_A N_A(\boldsymbol{x}) T_A(t) = \boldsymbol{N}\boldsymbol{T}(t) \tag{1.5}$$

$$v^h(\boldsymbol{x}) = \sum_B N_B(\boldsymbol{x}) v_B = \boldsymbol{N}\boldsymbol{v}\,. \tag{1.6}$$

For convenience we can also define a gradient operator, $\boldsymbol{B}$, just as before. If we now plug into our dynamic weak form expression we will have

$$\begin{aligned}
0 &= \sum_A \sum_B \left\{ \int_\Omega \rho c N_B v_B N_A \dot{T}_A(t) + \int_\Omega \nabla N_B v_B \cdot \boldsymbol{\kappa} \nabla N_A T_A(t) \right. \\
&\qquad \left. - \int_\Omega \rho r_s N_B v_B + \int_{\Gamma_h} h N_B v_B \right\} \qquad (1.7) \\
0 &= \sum_B v_B \sum_A \left\{ \int_\Omega \rho c N_B N_A \dot{T}_A(t) + \int_\Omega \nabla N_B \cdot \boldsymbol{\kappa} \nabla N_A T_A(t) \right. \\
&\qquad \left. - \int_\Omega \rho r_s N_B + \int_{\Gamma_h} h N_B \right\} \qquad (1.8) \\
0 &= \sum_A \left\{ \int_\Omega \rho c N_B N_A \dot{T}_A(t) + \int_\Omega \nabla N_B \cdot \boldsymbol{\kappa} \nabla N_A T_A(t) - \int_\Omega \rho r_s N_B + \int_{\Gamma_h} h N_B \right\} \qquad (1.9) \\
0 &= \left[\int_\Omega \rho c \boldsymbol{N}^T \boldsymbol{N} \right] \dot{\boldsymbol{T}}(t) + \left[\int_\Omega \boldsymbol{B}^T \boldsymbol{\kappa} \boldsymbol{B} \right] \boldsymbol{T}(t) - \int_\Omega \rho r_s \boldsymbol{N}^T + \int_{\Gamma_h} h \boldsymbol{N}^T . \qquad (1.10)
\end{aligned}$$

If we now define the heat capacity matrix (also known as the thermal mass matrix)

$$\boldsymbol{M} = \int_\Omega \rho c \boldsymbol{N}^T \boldsymbol{N} , \qquad (1.11)$$

the conductivity matrix (stiffness matrix)

$$\boldsymbol{K} = \int_\Omega \boldsymbol{B}^T \boldsymbol{\kappa} \boldsymbol{B} , \qquad (1.12)$$

and the generalized force vector as

$$\boldsymbol{F} = \int_\Omega \rho r_s \boldsymbol{N}^T - \int_{\Gamma_h} h \boldsymbol{N}^T , \qquad (1.13)$$

then the problem can be expressed as a system of linear first order ordinary differential equations

$$\boldsymbol{M} \dot{\boldsymbol{T}}(t) + \boldsymbol{K} \boldsymbol{T}(t) = \boldsymbol{F}(t) . \qquad (1.14)$$

This is a system of ordinary differential equations in time for the nodal unknowns. To solve these equations in time we will use *finite difference* methods.

1.3 Introduction to Numerical Integration by Finite Differences

The numerical integration of ordinary differential equations by the finite difference method is a very developed branch of applied mathematics. A through discussion covering this material in depth can be found in [14]; see also [13] for a more elementary treatment. For a discussion, within the context of finite element approximations one can look at [9, 5] or [1]. For a peek at the idea of using finite element methods instead of finite differences "in time" see [10].

1.4 A Scalar Example

To introduce the idea of numerical integration by finite differences consider the following initial value problem

$$\dot{y} = f(y,t) \tag{1.15}$$
$$y(0) = \tilde{y}\,, \tag{1.16}$$

where $f(\cdot)$ is a known function. We desire a methodology for generating the solution to this equation at discrete times $t_0 = 0$, $t_1 = \Delta t$, $t_2 = t_1 + \Delta t$, $\cdots$, where Δt is a given positive number. We will denote the approximate solution at time t_n as y_n, where it is desired that y_n be, in some fashion, "close" to the exact value $y(t_n)$. There are a large number of finite difference methods that one could apply to this problem [14, 13], however, we will only examine two of the simplest ones in order to introduce the basic ideas. The essential idea behind finite difference methods is to approximate the derivatives appearing in the ODE with an expression in terms of the values of the function $y(\cdot)$ and the time step Δt.

Forward Euler

In the forward Euler method we begin with a Taylor series expansion of the function $y(t)$. The expansion is formed about an arbitrary time step t_n and evaluated at the next time step t_{n+1}.

$$y(t_{n+1}) = y(t_n) + \dot{y}(t_n)\Delta t + O(\Delta t^2)\,. \tag{1.17}$$

We now substitute for the first time derivative of y using Eq. (1.16). Thus

$$y(t_{n+1}) = y(t_n) + f(y(t_n), t_n)\Delta t + O(\Delta t^2)\,. \tag{1.18}$$

If we now truncate the series and assume $y_n = y(t_n)$, then we come to the recursion relation

$$y_{n+1} = y_n + f(y_n, t_n)\Delta t \tag{1.19}$$

which defines the forward Euler method. To utilize the method, one starts with a known initial value $y_0 = y(0) = \tilde{y}$ and simply evaluates Eq. (1.19) to generate the sequence of approximate values $y_1, y_2, y_3, \ldots$. At the present we defer a critical examination of the properties of the approximation but simply note that just as in finite element methods one needs to consider certain issues in order to establish that what we have done will converge to the true solution $y(t)$. To this end we rewrite Eq. (1.19) in the completely equivalent form

$$\frac{y_{n+1} - y_n}{\Delta t} = f(y_n, t_n)\,. \tag{1.20}$$

This also shows that the recursion relation can be viewed as an approximation to the ODE. This will be important later on when we look at convergence.

Backward Euler

In the backward Euler method we also begin with a Taylor series expansion of the function $y(t)$. This time the expansion is formed about an arbitrary time step t_{n+1} and evaluated at the previous time step t_n.

$$y(t_n) = y(t_{n+1}) - \dot{y}(t_{n+1})\Delta t + O(\Delta t^2)\,. \tag{1.21}$$

We now substitute for the first time derivative of y using Eq. (1.16). Thus

$$y(t_n) = y(t_{n+1}) - f(y(t_{n+1}), t_{n+1})\Delta t + O(\Delta t^2)\,. \tag{1.22}$$

If we now truncate the series and assume $y_n = y(t_n)$, then we come to the recursion relation

$$y_{n+1} = y_n + f(y_{n+1}, t_{n+1})\Delta t\,, \tag{1.23}$$

which defines the backward Euler method. To utilize the method, one starts with a known initial value $y_0 = y(0) = \tilde{y}$ and solves Eq. (1.23) for y_{n+1} to generate a sequence of approximate values $y_1, y_2, y_3, \ldots$. Since an equation must be solved to determine y_{n+1} this is considered an implicit method. Formally, it is only slightly different from the forward Euler method but it does have very different approximation properties. If we write it in a form similar to Eq. (1.20) then we find

$$\frac{y_{n+1} - y_n}{\Delta t} = f(y_{n+1}, t_{n+1})\,. \tag{1.24}$$

This shows that the only difference between the two methods is where the derivative function is evaluated.

Other Methods

There are clearly a wide variety of similar methods that could be developed using similar ideas and keeping more terms in the Taylor series. Another category of methods can be derived by using quadrature methods; see [13]. To keep things simple, these two will suffice for a brief introduction.

1.5 Convergence

The main theorem that governs the properties of finite difference methods is call the Lax Equivalence theorem. Loosely, it states that for convergence, a finite difference method must be consistent and stable. If it has both these properties then the method will be convergent is the sense that as the time step Δt is decreased the sequence of values $y_1, y_2, y_3, \ldots$ will approach $y(t_1), y(t_2), y(t_3), \ldots$. For a more precise statement of this convergence property see [14] and [13].

Consistency

Consistency is the statement that the finite difference equation is an approximation to the initial ODE with a certain order of accuracy. The basic analysis is done by assuming that the $y_n = y(t_n)$ and subtracting the ODE and the recursion relation from each other. With the judicious use of Taylor series we can determine the discretization error, τ, involved in replacing the ODE with the recursion relation.

For example, with the forward Euler method we have

$$\tau = \underbrace{\frac{y_{n+1} - y_n}{\Delta t} - f(y_n, t_n)}_{\text{Finite Difference Recursion}} - \underbrace{[\dot{y}(t_n) - f(y_n, t_n)]}_{\text{ODE}} \tag{1.25}$$

$$\tau = \frac{y_{n+1} - y_n}{\Delta t} - \dot{y}(t_n) \tag{1.26}$$

$$\tau = \frac{y_n + \dot{y}(t_n)\Delta t + O(\Delta t^2) - y_n}{\Delta t} - \dot{y}(t_n) \tag{1.27}$$

$$\tau = O(\Delta t)\,. \tag{1.28}$$

This result tells us that the finite difference recursion relation is consistent with the ODE; i.e. to a certain degree they are the same and more importantly that as the time step decreases the difference between the two goes to zero. A similar analysis on the backward Euler method gives us the same result; ie. $\tau = O(\Delta t)$.

The requirement of consistency is that the exponent on Δt in the discretization error be strictly greater than zero; ie. for $\tau = O(\Delta t^k)$, one must have $k > 0$ for consistency. The standard terminology that is used says that the FD method is consistent (or accurate) of order k. With this result one can show *theoretically* that for reasonably well behaved functions $f(\cdot)$ that in the limit $\Delta t \to 0$ that the finite difference sequence approaches the exact solution. However, because of finite precsion round-off errors that occur in real compuations this theoretical result is insufficient to ensure actual convergence. The added condition for actual convergence is the other half of Lax's hypothesis – viz. stability.

Stability

The notion of stability refered to in Lax's Theorem is associated with the properties of the finite difference method itself. To test for this property, the FD method is normally applied to the standard test problem of $f(y,t) = \lambda y$, where $\lambda \in \mathcal{C}$ and $\mathcal{C}$ denotes the space of complex numbers. For this choice of $f(\cdot)$, the exact solution to the IVP is

$$y(t) = \tilde{y}\exp[\lambda t] = \tilde{y}\exp[\lambda_r t]\left(\cos(\lambda_i t) + \sin(\lambda_i t)\right)\,, \tag{1.29}$$

where λ_r and λ_i are the real and imaginary parts of λ. When $\lambda_r < 0$ it is seen that the exact solution is oscillating-decreasing. A finite difference method is considered to be stable if when it is applied to the test problem, that the sequence of values is decreasing for $\lambda_r < 0$.

As an example consider the forward Euler method. If we apply it to the test problem we have

$$y_{n+1} = (1 + \lambda\Delta t)y_n\,. \tag{1.30}$$

The solution to this recursion relation is found by using the canonical guess of $y_n = Ar^n$, where A and r are unknown. Inserting above gives the characteristic equation

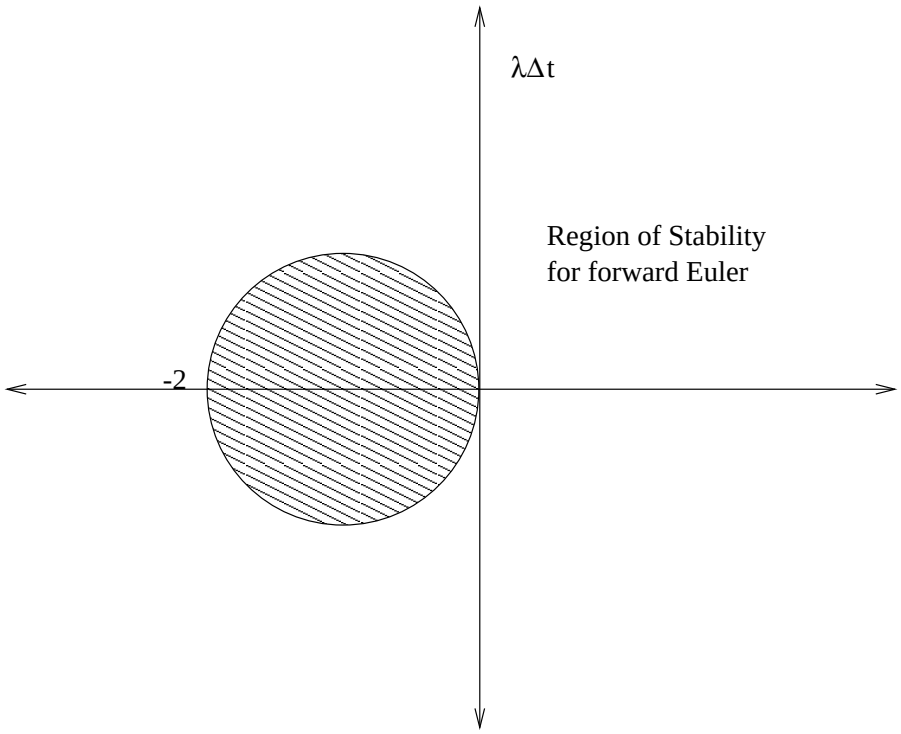
$$Ar^{n+1} = (1 + \lambda\Delta t)Ar^n \tag{1.31}$$

$$r = (1 + \lambda\Delta t)\,. \tag{1.32}$$

Thus $y_n = A(1+\lambda\Delta t)^n$. If we now check the initial condition we come to the final answer of

$$y_n = \tilde{y}(1 + \lambda\Delta t)^n\,. \tag{1.33}$$

For stability we must have $|1 + \lambda\Delta t| < 1$. The combination of values of λ and $\Delta t > 0$ that will satisfy this requirement is rather limited. The points in the complex $\lambda\Delta t$ plane where it is satisfied are depicted below as the disk of radius 1 centered at $\lambda\Delta t = -1 + 0i$, where i is the imaginary unit.



This type of finite difference method is called conditionally stable because for a given value of λ with $\lambda_r < 0$ the value of the time step is subject to a condition of being sufficiently small for stability to hold. If $|\lambda|$ is large, then Δt will have to be very small. The consequence of this is that to simulate the behavior of a system many time steps may be necessary. If the time step is chosen to violate this restriction, then the finite difference solution to the test problem will begin to increase without bound even though the exact solution is decreasing.

As a second example consider the backward Euler method. If we apply it to the test problem we have

$$y_{n+1} = y_n + \lambda\Delta t y_{n+1} \tag{1.34}$$

$$y_{n+1} = \left(\frac{1}{1 - \lambda\Delta t}\right) y_n\,. \tag{1.35}$$

The solution to this recursion relation is again found by using the canonical guess of $y_n = Ar^n$, where A and r are unknown. Inserting above gives the characteristic equation

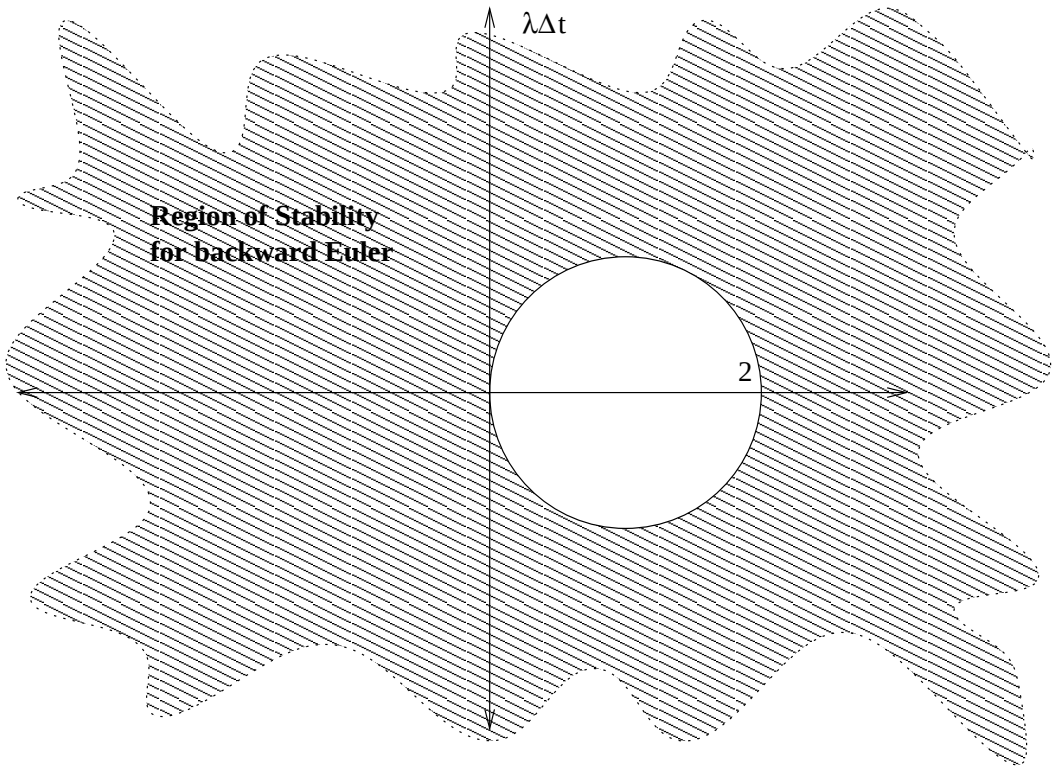
$$Ar^{n+1} = \left(\frac{1}{1-\lambda\Delta t}\right) Ar^n \tag{1.36}$$

$$r = \frac{1}{1-\lambda\Delta t}\,. \tag{1.37}$$

If we now check the initial condition we come to the final answer of

$$y_n = \tilde{y}\left(\frac{1}{1-\lambda\Delta t}\right)^n\,. \tag{1.38}$$

For stability we must have $|\frac{1}{1-\lambda\Delta t}| < 1$. The combination of values of λ and $\Delta t > 0$ that will satisfy this requirement is quite large. The points in the complex $\lambda\Delta t$ plane where it is satisfied are depicted below as the entire region outside of the disk of radius 1 centered at $\lambda\Delta t = 1 + 0i$.



This type of finite difference method is called unconditionally stable because for any given value of λ with $\lambda_r < 0$ the value of the time step is not subject to any restrictions for stability to hold. The consequence of this is that in the simulation of the behavior of a system time steps of any size may be taken (subject of course to the demands of accuracy). Note the method even predicts decaying solutions for many values of $\lambda\Delta t$ where $\lambda_r > 0$ and the exact solution is increasing without bound. Thus the method could be said to be overly stable.

REMARKS:

1. For a given consistent finite difference method, if it operates within its region of stability with respect to the test problem then it can be considered to be convergent.

2. Conditionally stable methods are normally suitable for calculations that range over time intervals that are on the order of the characteristic time of the system under investigation. This is primarily due to the severe time step restrictions.

3. Unconditionally stable methods (which are always implicit) are more expensive to use than typical conditionally stable methods (which are normally but not always explicit); however, one can use them for longer time simulations with larger time steps. Thus they are better suited to longer duration simulations.

4. The forward and backward Euler methods are suitable for applying to first order ODEs. In the next section they are applied to the heat conduction problem and a discussion is presented on how to interpret the stability requirement when one has a system of ODEs as opposed to just a single ODE.

1.6 Application to Heat Conduction

The finite element discretized heat conduction equations are given by

$$\boldsymbol{M}\dot{\boldsymbol{T}} + \boldsymbol{K}\boldsymbol{T} = \boldsymbol{F}\,. \tag{1.39}$$

In the next two sections we will apply the forward and backward Euler finite difference methods to this system of ODEs and then in the following section we will examine the relation of stability as discussed with in the previous section to the present problem that involves a system of equations.

Forward Euler

The application of forward Euler to Eq. (1.39) gives

$$\boldsymbol{M}\left(\frac{\boldsymbol{T}_{n+1} - \boldsymbol{T}_n}{\Delta t}\right) + \boldsymbol{K}\boldsymbol{T}_n = \boldsymbol{F}_n\,. \tag{1.40}$$

Rearranging yields

$$\boldsymbol{M}\boldsymbol{T}_{n+1} = \boldsymbol{M}\boldsymbol{T}_n - \Delta t\boldsymbol{K}\boldsymbol{T}_n + \Delta t\boldsymbol{F}_n\,. \tag{1.41}$$

Thus knowing the state a time t_n and the heat-load at t_n we can compute the state at time t_{n+1} by solving these linear algebraic equations.

REMARKS:

1. Normally, when working with an "explicit" method one modifies the thermal mass matrix to make it diagonal so that no equations need to be solved to get the new temperature field. The modified thermal-mass is usually called the lumped thermal-mass and it is formed simply by taking the off diagonal terms in a row and adding them to the diagonal element and then zeroing them. This obviously changes the problem being solved in some fashion; for now we defer a discussion of this approximation until we discuss mass lumping in the context of elastodynamics.

Backward Euler

The application of backward Euler to Eq. (1.39) gives

$$\boldsymbol{M}\left(\frac{\boldsymbol{T}_{n+1} - \boldsymbol{T}_n}{\Delta t}\right) + \boldsymbol{K}\boldsymbol{T}_{n+1} = \boldsymbol{F}_{n+1}\,. \tag{1.42}$$

Rearranging yields

$$(\boldsymbol{M} + \Delta t\boldsymbol{K})\,\boldsymbol{T}_{n+1} = \boldsymbol{M}\boldsymbol{T}_n + \Delta t\boldsymbol{F}_{n+1}\,. \tag{1.43}$$

Thus knowing the state a time t_n and the heat-load at t_{n+1} we can find the state at time t_{n+1} by solving these linear algebraic equations.

REMARKS:

1. The matrix on the left-hand side in Eq. (1.43) is often called the dynamic heat conduction matrix, since formally it replaces the static FEM heat conduction matrix.

Stability

To understand the relation of the stability of the test equation to the present system of equations requires a modal decomposition of our system of equations. We begin with the homogeneous equations (ie. $\boldsymbol{F} = \boldsymbol{0}$) and the general guess to the solution of the form

$$\boldsymbol{T}(t) = \boldsymbol{\Psi} \exp[-\lambda t] \,. \tag{1.44}$$

$\boldsymbol{\Psi}$ is an unknown vector and λ is an unknown scalar. We can determine $\boldsymbol{\Psi}$ and λ by inserting the guess into Eq. (1.39). This gives

$$-\lambda \boldsymbol{M}\boldsymbol{\Psi} \exp[-\lambda t] + \boldsymbol{K}\boldsymbol{\Psi} \exp[-\lambda t] = 0 \,. \tag{1.45}$$

Or equivalently

$$(\boldsymbol{K} - \lambda \boldsymbol{M})\, \boldsymbol{\Psi} = 0 \,. \tag{1.46}$$

This is simply a generalized eigenvalue problem. For non-trivial solutions to Eq. (1.46), the determinant of the matrix on the left-hand side needs to be equal to zero. Thus there will be as many eigenpairs as there are degrees-of-freedom in the system. We will denote these as $(\boldsymbol{\Psi}_l, \lambda_l)$ for $l = 1, \ldots, n$, where n is the number of degrees of freedom in the problem. The actual solution can be formulated as a linear combination of these eigenpairs.

A very important property of the eigenvectors is that they are mass-orthogonal. Further we usually normalize them so that they are mass-orthonormal. To see this property start with two eigenpairs with differing eigenvalues $(\boldsymbol{\Psi}_i, \lambda_i)$ and $(\boldsymbol{\Psi}_j, \lambda_j)$. Then we have that

$$\lambda_i \boldsymbol{M}\boldsymbol{\Psi}_i = \boldsymbol{K}\boldsymbol{\Psi}_i \tag{1.47}$$

$$\lambda_j \boldsymbol{M}\boldsymbol{\Psi}_j = \boldsymbol{K}\boldsymbol{\Psi}_j \,. \tag{1.48}$$

If we now take the dot product of the first equation by $\boldsymbol{\Psi}_j$, the second by $\boldsymbol{\Psi}_i$, and subtract the two results from each other we come to

$$(\lambda_i - \lambda_j)\boldsymbol{\Psi}_j \cdot \boldsymbol{M}\boldsymbol{\Psi}_i = 0 \,, \tag{1.49}$$

where advantage has been made of the symmetry of both $\boldsymbol{M}$ and $\boldsymbol{K}$. Thus if $\lambda_i - \lambda_j \neq 0$ then we have $\boldsymbol{\Psi}_j \cdot \boldsymbol{M}\boldsymbol{\Psi}_i = 0$; if $\lambda_i - \lambda_j = 0$. If $i = j$, then we get a finite value. We normalize this value so that both cases can be expressed as

$$\boldsymbol{\Psi}_j \cdot \boldsymbol{M}\boldsymbol{\Psi}_i = \delta_{ji} \,. \tag{1.50}$$

Note also that this implies that

$$\boldsymbol{\Psi}_j \cdot \boldsymbol{K}\boldsymbol{\Psi}_i = \lambda_i \delta_{ji} \qquad \text{(no sum)} \,. \tag{1.51}$$

From these properties we can conclude that the generalized eigenvectors form a basis for $\mathbb{R}^n$ and we can expand the exact solution to Eq. (1.39) in them. In other word we can write

$$\boldsymbol{T}(t) = \sum_{i=1}^{n} T_{(i)}(t) \boldsymbol{\Psi}_i \,, \tag{1.52}$$

where $T_{(i)}(t)$ are known as the modal amplitudes of the solution. Inserting this expansion into Eq. (1.39), gives

$$\boldsymbol{M}\left(\sum_{i=1}^{n}\dot{T}_{(i)}(t)\boldsymbol{\Psi}_i\right)+\boldsymbol{K}\left(\sum_{i=1}^{n}T_{(i)}(t)\boldsymbol{\Psi}_i\right)=\boldsymbol{F}(t)\,. \tag{1.53}$$

If we now take the dot product of both sides with an eigenvector $\boldsymbol{\Psi}_j$ and make use of the mass-orthonormal properties of the eigenvectors, then we end up for each value of j with the following equation

$$\dot{T}_{(j)}(t)+\lambda_j T_{(j)}(t)=F_{(j)}(t)\,, \tag{1.54}$$

where $F_{(j)}=\boldsymbol{\Psi}_j\cdot\boldsymbol{F}(t)$. Thus our system of linear first order ODEs turns into n decoupled scalar ODEs in the modal amplitudes.

REMARKS:

1. When applying our finite difference methods to the heat conduction equations we are essentially applying it to each of these "single degree-of-freedom" problems simultaneously. The stability of the time stepping will then be governed by the magnitudes of the eigenvalues λ_j. In particular, if we use backward Euler then there will be no time step restriction for stability. If we use forward Euler then,

$$\Delta t<\frac{2}{\max\limits_{j=1,\ldots,n}\lambda_j}\,. \tag{1.55}$$

   Thus it is the "fastest" mode of the system that will govern the time step (even if we are only interested in slower modes).

2. The practical issue of choosing a time step for conditionally stable methods relies on being able to estimate the largest generalized eigenvalue of the system. Common techniques include power iteration and inverse iteration with Rayleigh quotient; see [8] for a discussion of these methods and their properties. Note that only an upper bound to the largest eigenvalue needs to be determined. One of the most practical ways of doing this is to observe that the element eigenvalues bound the system eigenvalues from above.

References

[1] K.-J. Bathe, *Finite element procedures*, Prentice Hall, 1996.

[2] G.P. Bazeley, Y.K. Cheung, B.M. Irons, and O.C. Zienkiewicz, *Triangular elements in plate bending – conforming and nonconforming solutions*, Proceedings of the First Conference on Matrix Methods in Structural Mechanics, 1965.

[3] A.K. Chopra, *Dynamics of structures*, Prentice Hall, 1995.

[4] S.D. Conte and C. de Boor, *Elementary numerical analysis*, McGraw-Hill, 1980.

[5] R.D. Cook, D.S. Malkus, and M.E. Plesha, *Concepts and applications of finite element analysis*, 3rd ed., John Wiley, 1989.

[6] I. Fried, *Finite element analysis of incompressible materials by residual energy balancing*, International Journal of Solids and Structures **10** (1974), 993–1002.

[7] Y. C. Fung, *Foundations of solid mechanics*, Prentice-Hall, 1965.

[8] G.H. Golub and C.F. Van Loan, *Matrix computations*, 3rd ed., Johns Hopkins University Press, 1996.

[9] T.J.R. Hughes, *The finite element method*, Prentice Hall, 1987.

[10] C. Johnson, *Numerical solutions of partial differential equations by the finite element method*, Cambridge University Press, 1987.

[11] A.W. Leissa, *Vibration of shells*, Tech. report, NASA, Washington, 1973.

[12] J.C. Nagtegaal, D.M. Parks, and J.R. Rice, *On numerically accurate finite element solutions in the fully plastic range*, Computer Methods in Applied Mechanics and Engineering **4** (1974), 153–178.

[13] D. Quinney, *An introduction to the numerical solution of differential equations*, Research Studies Press (John Wiley), 1985.

[14] R.D. Richtmeyer and K.W. Morten, *Difference methods for initial-value problems*, John Wiley, 1967.

[15] J.C. Simo and M.S. Rifai, *A class of mixed assumed strain methods and the method of incompatible modes*, International Journal for Numerical Methods in Engineering **29** (1990), 1595–1638.

[16] I. S. Sokolnikoff, *Mathematical theory of elasticity*, McGraw-Hill, 1956.

[17] G. Strang and G. Fix, *An analysis of the finite element method*, Wellesley-Cambridge Press, 1973.

[18] R.L. Taylor, P.J. Beresford, and E.L. Wilson, *A nonconforming element fo stress analysis*, International Journal for Numerical Methods in Engineering **10** (1976), 1211–1219.

[19] R.L. Taylor, J.C. Simo, O.C. Zienkiewics, and A.C. Chan, *The patch test: A condition for assessing finite element convergence*, International Journal for Numerical Methods in Engineering **22** (1986), 39–62.

[20] S. Timoshenko and J. N. Goodier, *Theory of elasticity*, McGraw-Hill, 1970.

[21] K. Washizu, *Variational methods in elasticity and plasticity*, Pergamon Press, 1968.

[22] E.L Wilson, R.L. Taylor, W.P. Doherty, and J. Ghaboussi, *Incompatible displacement models*, Numerical and Computer Models in Structural Mechanics (S.J. Fenves, N. Perrone, A.R. Robinson, and W.C. Schonbirch, eds.), Academic Press, 1973, pp. 43–57.

[23] O.C. Zienkiewicz and R.L. Taylor, *The finite element method, volume 1*, 5th ed., Butterworth and Heinemann, 2000.

[24] ———, *The finite element method, volume 2*, 5th ed., Butterworth and Heinemann, 2000.