Appendix 2

Numerical Methods

Here we give additional details about well-known numerical methods which are commonly-used in conjunction with finite element codes.

2.1 The θ-Method

The θ-method is a standard technique for time-discretization of systems of first-order ODEs. While it is not usually considered "adaptive" in the strictest sense, in practice we can choose the parameter θ dynamically to achieve additional "stability" (implicit Euler, $\theta = 1$), or conversely additional accuracy (Crank-Nicolson, $\theta = 1/2$). (While all of the implicit θ-methods are unconditionally "A_0" stable, the Crank-Nicolson method lacks so-called "L_0" stability, and is known to suffer from the phenomenon of "ringing" (as is well-described in the reference work by Smith [192]) in cases where the boundary and initial conditions do not coincide. This tendency for unwanted oscillations to grow in size is exacerbated by decreasing the spatial grid size h — clearly an undesirable property on adaptively-refined grids.

While the θ-method (or any of the subsidiary methods derivable from the θ-method) is typically derived for standard systems of ODEs in the form

$$\dot{\boldsymbol{U}} = \boldsymbol{f}(\boldsymbol{U}, t) \tag{B-2.1}$$

where $\boldsymbol{U}$ is an $N \times 1$ vector of unknowns and $\boldsymbol{f} : \mathbb{R}^N \times \mathbb{R} \to \mathbb{R}^N$ is a given function. In the course of studying systems of nonlinear natural convection PDEs discretized by the finite element method in space, it is actually much more common to deal with systems of ODEs of the form

$$\boldsymbol{M}(\boldsymbol{U})\dot{\boldsymbol{U}} = \boldsymbol{f}(\boldsymbol{U}, t) \tag{B-2.2}$$

where $\boldsymbol{M}$ is the so-called "mass matrix," which may depend on the unknown $\boldsymbol{U}$ (as in some stabilized schemes) and may not be invertible due to the presence of time-independent constraint equations, such as e.g. the incompressibility constraint. Obviously if $\boldsymbol{M}$ is constant and formally invertible then Eqn. (B-2.2) can be rewritten in the same form as Eqn. (B-2.1) and the usual techniques can be applied. Here, we focus entirely on deriving methods for the non-constant mass matrix case of Eqn. (B-2.2).

To motivate the θ-method, we consider time interval n, for which $t \in [t_n, t_{n+1}]$ and for which $t_{n+1} = t_n + \Delta t$ defines the timestep Δt. Assuming $\boldsymbol{U}$ is smooth enough in this time interval, we can expand $\boldsymbol{U}$ in independent Taylor series about t_n and t_{n+1} as

$$\begin{aligned} \boldsymbol{U}^{n+1} &= \boldsymbol{U}^{n} + \Delta t \dot{\boldsymbol{U}}^{n} + \frac{\Delta t^2}{2} \ddot{\boldsymbol{U}}^{n} + \mathcal{O}(\Delta t^3) && \text{(B-2.3)} \\ \boldsymbol{U}^{n} &= \boldsymbol{U}^{n+1} - \Delta t \dot{\boldsymbol{U}}^{n+1} + \frac{\Delta t^2}{2} \ddot{\boldsymbol{U}}^{n+1} + \mathcal{O}(\Delta t^3) && \text{(B-2.4)} \end{aligned}$$

The explicit (resp. implicit) Euler method is obtained from Eqn. (B-2.3) (resp. Eqn. (B-2.4)) by dropping terms of $\mathcal{O}(\Delta t^3)$ and higher, and substituting for $\dot{\boldsymbol{U}}$ from Eqn. (B-2.1) or (B-2.2). In the first-order case, the mass matrix inverse is used formally during this substitution step, but is subsequently "multiplied out" in the final step so that it does not appear in the final scheme. For completeness, we give the explicit and implicit Euler schemes for Eqn. (B-2.2) here

$$\text{Explicit:} \quad \boldsymbol{M}\left(\boldsymbol{U}^n\right)\left(\frac{\boldsymbol{U}^{n+1}-\boldsymbol{U}^n}{\Delta t}\right) = \boldsymbol{f}(\boldsymbol{U}^n, t_n) + \mathcal{O}(\Delta t) \tag{B-2.5}$$

$$\text{Implicit:} \quad \boldsymbol{M}\left(\boldsymbol{U}^{n+1}\right)\left(\frac{\boldsymbol{U}^{n+1}-\boldsymbol{U}^n}{\Delta t}\right) = \boldsymbol{f}(\boldsymbol{U}^{n+1}, t_{n+1}) + \mathcal{O}(\Delta t) \tag{B-2.6}$$

The θ-method is derived for the constant mass matrix case by observing that an advantageous cancellation of the Taylor series truncation error is obtained when selecting a particular linear combination of Eqns. (B-2.3) and (B-2.4). A similar procedure is possible in the non-constant mass matrix case as well, but additional requirements on $\boldsymbol{M}$ are necessary to ensure that the resulting method will indeed be second-order. For simplicity, let $\boldsymbol{M}^n := \boldsymbol{M}(\boldsymbol{U}^n)$ and $\boldsymbol{f}^n := \boldsymbol{f}(\boldsymbol{U}^n, t_n)$. Then, by multiplying Eqn. (B-2.3) by $(1-\theta)\boldsymbol{M}^n$ and Eqn. (B-2.4) by $-\theta\boldsymbol{M}^{n+1}$ and adding them together, we obtain

$$\begin{aligned} \left((1-\theta)\boldsymbol{M}^n + \theta\boldsymbol{M}^{n+1}\right)\left(\frac{\boldsymbol{U}^{n+1}-\boldsymbol{U}^n}{\Delta t}\right) &= (1-\theta)\,\boldsymbol{f}^n + \theta\boldsymbol{f}^{n+1} \\ &+ \text{T.E.} \end{aligned} \tag{B-2.7}$$

where the truncation error has the particular form

$$\text{T.E.} := \Delta t \left[\left(\frac{(1-\theta)}{2} \boldsymbol{M}^n - \frac{\theta}{2} \boldsymbol{M}^{n+1} \right) \ddot{\boldsymbol{U}}^{n+1} + \mathcal{O}(\Delta t) \right] \tag{B-2.8}$$

Unlike the constant mass matrix case, setting $\theta = 1/2$ (the Crank-Nicolson scheme) is not sufficient to obtain a second-order accurate method in time unless we can also show that

$$\boldsymbol{M}^{n+1} = (1 + \mathcal{O}(\Delta t)) \boldsymbol{M}^n \tag{B-2.9}$$

For the nonlinear natural convection problems of interest here, the restriction of Eqn. (B-2.9) will obviously be satisfied for any constant mass matrix (or even mass matrices with zero blocks, as in the case of unstabilized Rayleigh-Bénard-Marangoni flows and double-diffusive convection in porous media.)

It is more difficult to analyze the mass matrices which arise due to nonlinear stabilization terms. In general, the stabilization parameter τ can depend on the unknown $\boldsymbol{U}$ in a highly-nonlinear way, and these effects must be analyzed on a case-by-case basis. We note that these considerations on $\boldsymbol{M}$ in no way affect the accuracy of the first-order schemes, and so one must carefully justify the expense of assembling the additional right-hand side terms for the Crank-Nicolson method by showing (either analytically or by numerical experimentation) that the resulting scheme is truly second-order accurate.

A variation on the preceding scheme is obtained by considering Taylor series *not* about the end points $t \in [t_n, t_{n+1}]$ of the time interval, but instead about a somewhat arbitrary intermediate time

$$t_\theta := \theta t_{n+1} + (1-\theta)\, t_n \tag{B-2.10}$$

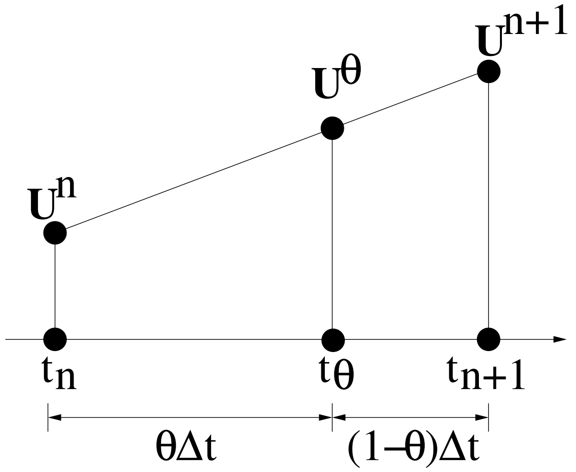

Figure B-2.1: Definition of the intermediate value $\boldsymbol{U}^\theta$.

where $0 \leq \theta \leq 1$ as before. By assuming the solution dependence is linear between the two endpoint values (see Fig. B-2.1) it follows that the solution at time t_θ is given by

$$\boldsymbol{U}^\theta := \theta \boldsymbol{U}^{n+1} + (1-\theta)\,\boldsymbol{U}^n \tag{B-2.11}$$

The selection of t_θ in this manner splits the full timestep Δt up into two parts of size $\theta\Delta t$ and $(1-\theta)\Delta t$, as shown in the figure. We may now once again construct two Taylor series expansions of $\boldsymbol{U}$, this time both will be centered about the $\boldsymbol{U}^\theta$ solution.

$$\boldsymbol{U}^{n+1} = \boldsymbol{U}^\theta + (1-\theta)\Delta t\dot{\boldsymbol{U}}^\theta + \frac{(1-\theta)^2\Delta t^2}{2}\ddot{\boldsymbol{U}}^\theta + \mathcal{O}(\Delta t^3) \tag{B-2.12}$$

$$\boldsymbol{U}^{n} = \boldsymbol{U}^\theta - \theta\,\Delta t\dot{\boldsymbol{U}}^\theta + \frac{\theta^2\Delta t^2}{2}\ddot{\boldsymbol{U}}^\theta + \mathcal{O}(\Delta t^3) \tag{B-2.13}$$

Subtracting Eqn. (B-2.13) from Eqn. (B-2.12) we find

$$\boldsymbol{U}^{n+1} - \boldsymbol{U}^n = \Delta t\dot{\boldsymbol{U}}^\theta + \frac{\Delta t^2}{2}(1-2\theta)\ddot{\boldsymbol{U}}^\theta + \mathcal{O}(\Delta t^3) \tag{B-2.14}$$

Finally, multiplying Eqn. (B-2.14) through by $\frac{1}{\Delta t}\boldsymbol{M}_\theta$, and substituting

$$\boldsymbol{M}_\theta \dot{\boldsymbol{U}}^\theta = \boldsymbol{f}(\boldsymbol{U}^\theta, t_\theta) \tag{B-2.15}$$

from the original ODE of Eqn. (B-2.2), we obtain

$$\boldsymbol{M}_\theta \left(\frac{\boldsymbol{U}^{n+1} - \boldsymbol{U}^n}{\Delta t} \right) = \boldsymbol{f}(\boldsymbol{U}^\theta, t_\theta) + (1 - 2\theta)\mathcal{O}(\Delta t) + \mathcal{O}(\Delta t^2) \tag{B-2.16}$$

Clearly, the method is second-order accurate in time only for the specific choice of $\theta = 1/2$, the truncation error here having a much simpler form than for Eqn. (B-2.8). This would appear to be the more useful of the two schemes, especially for problems with nonlinear mass matrices.

2.2 Step-Doubling Methods

An interesting aspect of "step-doubling" methods is that they do not depend on a particular underlying timestepping scheme (such as in the case of predictor-corrector methods like ABTR) and they do not require larger "stencils" i.e. more saved old solutions, which some predictors require in order to maintain a reasonable level of accuracy.

Step-doubling methods have been used with some success with explicit Runge-Kutta methods (see §16.2 of the Numerical Recipes book [159], or Gear's book [83]) where they are sometimes called RK45 methods. Step-doubling methods are particularly efficient in this case due to their ability to directly reuse computed information effectively. In this work, we have employed step-doubling primarily in the context of an underlying implicit (first-order)