Chapter 13
Temporal Discretization: The Transient Term

Abstract The discussions in previous chapters assumed steady state conditions, which did not require the discretization of the transient term. Accounting for transient phenomena adds a new dimension to the problem. However since transient variations are parabolic by nature, there is no need to define a field in the time dimension, as is the case for the spatial domain. In general only one or two additional variable fields, or time levels, are stored (depending on the numerical order of the selected scheme). Another difference with steady state configurations is that transient systems are modeled using a time stepping procedure. Starting with an initial condition at time $t = t_0$, the solution algorithm marches forward and finds a solution at time $t_1 = t_0 + \Delta t_1$. The solution found is the initial condition for the next time step and is used to obtain the solution at time $t_2 = t_1 + \Delta t_2$. The process is repeated until the required time is reached. The focus of this chapter is on techniques used for the discretization of the transient term. Two approaches for developing transient schemes are presented. In the first one Taylor expansions are used to express the transient term with the aid of nodal values. This is in effect a finite difference discretization. In the second approach the finite volume method is used on a pseudo time element in a similar fashion to what was done to the convection term. Several transient schemes are presented and their characteristics discussed.

13.1 Introduction

For transient simulations, the governing equations are discretized in both space and time. While the spatial discretization is performed in the spatial domain as was done for the steady-state case, the temporal discretization involves setting up a time coordinate along which the derivative (for the finite difference method) or the integral (for the finite volume method) of the transient term is evaluated (Fig. 13.1).

In general, the expression for the transient behavior, or time evolution, of a variable ϕ is governed by an equation of the form

F. Moukalled et al., *The Finite Volume Method in Computational Fluid Dynamics*,
Fluid Mechanics and Its Applications 113, DOI 10.1007/978-3-319-16874-6_13

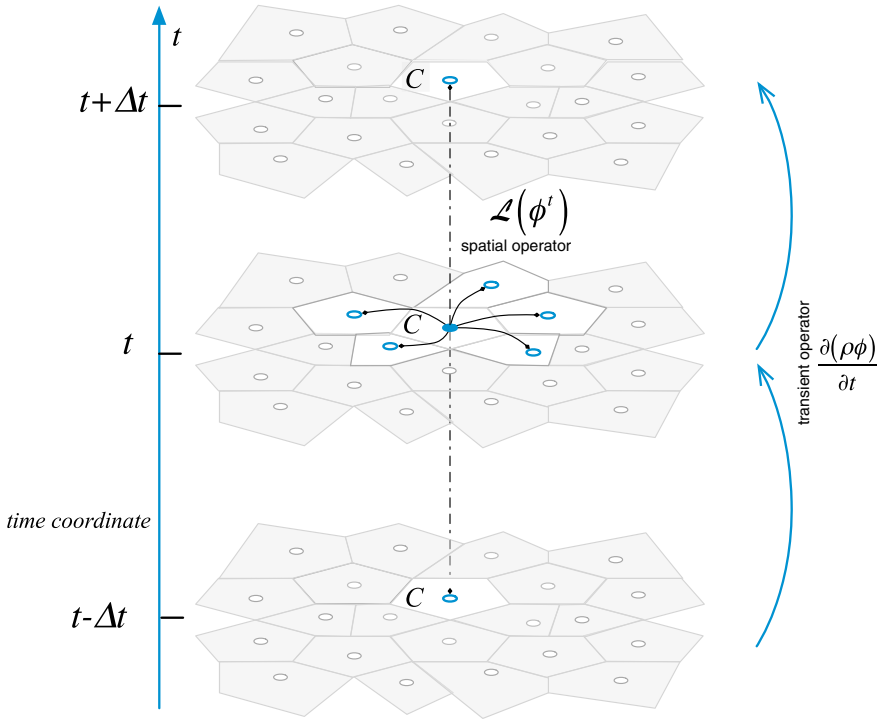

Fig. 13.1 Time coordinate, transient, and spatial operators

$$\frac{\partial(\rho\phi)}{\partial t} + \mathcal{L}(\phi) = 0 \tag{13.1}$$

where the function $\mathcal{L}(\phi)$ is a spatial operator that includes all non-transient terms (convection, advection, sources, etc.) and $\partial(\rho\phi)/\partial t$ is the transient operator, both displayed in Fig. 13.1.

Integrating Eq. (13.1) over an element C (Fig. 13.2) yields

$$\int_{V_C} \frac{\partial(\rho\phi)}{\partial t} dV + \int_{V_C} \mathcal{L}(\phi) dV = 0 \tag{13.2}$$

which, after a spatial discretization about the volume centroid, becomes

$$\frac{\partial(\rho_C \phi_C)}{\partial t} V_C + L\left(\phi_C^t\right) = 0 \tag{13.3}$$

where V_C is the volume of the discretization element and $L\left(\phi_C^t\right)$ is the spatial discretization operator expressed at some reference time t, which can be written in algebraic form as

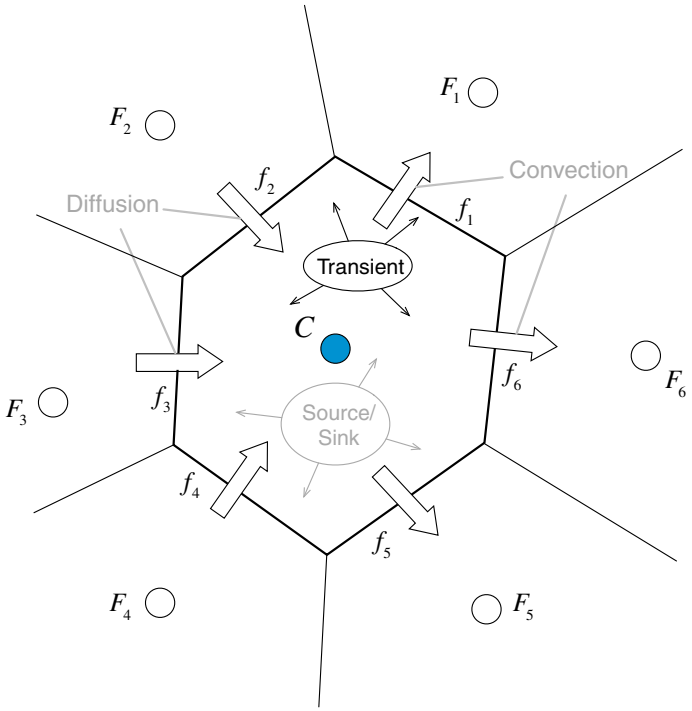

Fig. 13.2 Spatial element

$$L\left(\phi_C^t\right) = a_C\phi_C^t + \sum_{F\sim NB(C)} a_F\phi_F^t - b_C \tag{13.4}$$

In Eq. (13.3) the steady state discrete equation is recovered when $t \to \infty$. This is also true when steady state is reached through time marching, i.e., when $\phi_C^{t+\Delta t} = \phi_C^t$. This guarantees that the solution obtained when steady state is reached is the same as the one that would have been obtained with the problem solved directly as a steady state one.

For the discretization of the transient term, the practice traditionally has been to follow a finite difference approach [1–3], whereby a Taylor series expansion of $\partial(\rho\phi)/\partial t$ is used to express the derivative in terms of the discrete nodal values. In this chapter, another procedure that is more in line with the finite volume approach will also be presented. In this context, $\partial(\rho\phi)/\partial t$ is integrated over a temporal element [4] and transformed into face fluxes in a similar fashion to what was done with convection schemes, except that the discretization is now performed along the transient axes.

13.2 The Finite Difference Approach

Since in the transient space the grid is structured (Fig. 13.3), it has been quite common to treat the transient term using the finite difference method. In this approach, the spatial operator, $L(\phi)$, is discretized at time t, while the transient derivative is evaluated using a combination of Taylor expansions about time t resulting in a variety of transient schemes, some of which are described next.

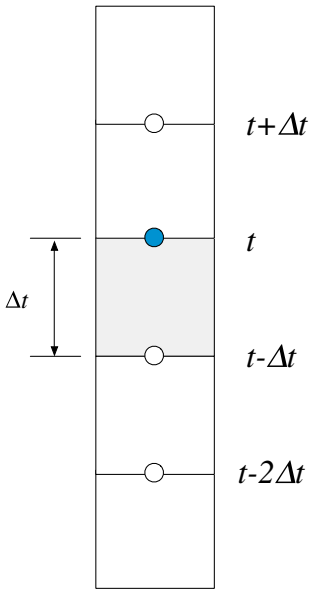

Fig. 13.3 Structured transient finite difference grid

13.2.1 Forward Euler Scheme

To evaluate the transient term, a Taylor expansion of the derived quantity about a time direction is needed. In this first case, the expansion is performed in a forward manner about time t. That is for some function T, its value at time $t + \Delta t$ is expressed using a Taylor series in terms of the values of T and its derivates at time t as

$$T(t + \Delta t) = T(t) + \frac{\partial T(t)}{\partial t}\Delta t + \frac{\partial^2 T(t)}{\partial t^2}\frac{\Delta t^2}{2!} + \cdots. \tag{13.5}$$

Truncating the series starting with terms of order Δt^2, the first derivative can be formulated as

$$\frac{\partial T(t)}{\partial t} = \frac{T(t+\Delta t) - T(t)}{\Delta t} + O(\Delta t) \tag{13.6}$$

This is now a first order discretization since the equation was divided by Δt to yield the gradient approximation. Replacing T by $(\rho\phi)$ in Eq. (13.6) and substituting the resulting expression for the derivative in Eq. (13.3), the discretized equation becomes

$$\frac{(\rho_C\phi_C)^{t+\Delta t} - (\rho_C\phi_C)^t}{\Delta t} V_C + L\left(\phi_C^t\right) = 0. \tag{13.7}$$

The transient stencil for Eq. (13.7) shown in Fig. 13.4, indicates that the computation of $(\rho_C\phi_C)$ at time $t+\Delta t$ does not require solving a system of equations. Rather, values of ϕ_C at time $t+\Delta t$ can be computed explicitly based on values from the previous time step since all spatial terms are evaluated at the old time t. The resulting scheme belongs to the class denoted by explicit transient schemes [5–12]. The main characteristic of all explicit transient schemes is their capability of generating solutions by marching in time without the need to solve a system of equations at each time level. This provides a high computational efficiency and simplifies the parallelization of the computational mesh. Yet only few commercial codes have adopted this approach and for an important reason related to a limitation on the size of Δt, which will be discussed in the next section.

Substituting the discretized algebraic relation of the spatial operator into Eq. (13.7), the complete algebraic equation is obtained as

$$a_C^{t+\Delta t}\phi_C^{t+\Delta t} + a_C^t\phi_C^t = b_C - \left(a_C\phi_C^t + \sum_{F\sim NB(C)} a_F\phi_F^t\right) \tag{13.8}$$

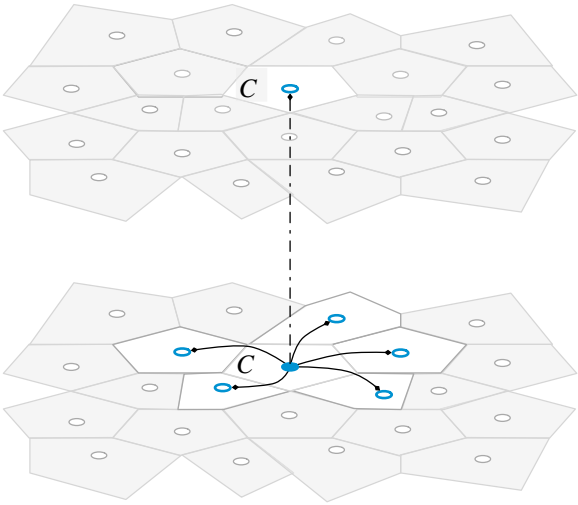

Fig. 13.4 The explicit Euler stencil

where

$$a_C^{t+\Delta t} = \frac{\rho_C^{t+\Delta t} V_C}{\Delta t}$$
$$a_C^{t} = -\frac{\rho_C^{t} V_C}{\Delta t} \tag{13.9}$$

In the above equations $a_C^{t+\Delta t}$ and a_C^t are the diagonal coefficients resulting from the discretization of the transient term, $\phi_C^{t+\Delta t}$ and ϕ_C^t are the values at time levels $t + \Delta t$ and t, respectively, and a_C, a_F, and b_C are the coefficients obtained from the spatial discretization.

To simplify notation, throughout this chapter variables referring to values obtained at a previous time step will be denoted with a superscript $^\circ$ and variables referring to values obtained two time steps earlier will be denoted with a superscript $^{\circ\circ}$. On the other hand no superscript will be used to denote variables at the current time step except for the coefficient of the unsteady term multiplying ϕ_C, which will be denoted with the superscript $^\bullet$. Adopting the new notation, Eqs. (13.8) and (13.9) become

$$a_C^\bullet \phi_C + a_C^\circ \phi_C^\circ = b_C - \left(a_C \phi_C^\circ + \sum_{F \sim NB(C)} a_F \phi_F^\circ \right) \tag{13.10}$$

where

$$a_C^\bullet = \frac{\rho_C V_C}{\Delta t}$$
$$a_C^\circ = -\frac{\rho_C^\circ V_C}{\Delta t} \tag{13.11}$$

Equation (13.10) can be re-arranged into

$$\phi_C = \frac{b_C - \left(\left(a_C + a_C^\circ \right) \phi_C^\circ + \sum_{F \sim NB(C)} a_F \phi_F^\circ \right)}{a_C^\bullet} \tag{13.12}$$

clearly showing that values of ϕ at the current time step are computed via an **explicit** relation without solving a system of equations.

13.2.2 Stability of the Forward Euler Scheme

The convergence and stability of numerical schemes was initially addressed by Courant, Friedrichs, and Lewy [13]. They showed that in order for the solution of a difference equation to converge to the solution of the partial differential equation the numerical scheme must use all the information contained in the initial data that influence the solution. This requirement has become later known as the CFL condition.

In reality the CFL condition can be interpreted simply as one of the basic rules that should be satisfied by the coefficients, namely the opposite signs rule extended to include the transient coefficients. Thus just as ϕ_F is considered a 'spatial' neighbor of ϕ_C, ϕ_C° is a 'temporal' neighbor of ϕ_C, and the opposite signs rule should equally apply to both. Noting that the diagonal coefficient is now $a_C^\bullet$ and the coefficient of its 'temporal' neighbor is $\left(a_C + a_C^\circ\right)$, the opposite signs requirement becomes

$$a_C + a_C^\circ \leq 0. \tag{13.13}$$

13.2.2.1 Stability of a Transient-Advection Case

For the one dimensional pure advection problem with a flow moving right wise shown in Fig. 13.5, the a_C and a_C° coefficients in the discretized equation of element C, using the upwind scheme for the interpolation of all variables at an element face, are given by

$$a_C = \dot{m}_e^\circ = \rho_C^\circ u_C^\circ \Delta y_C \qquad a_C^\circ = -\frac{\rho_C^\circ V_C}{\Delta t} = -\frac{\rho_C^\circ \Delta x_C \Delta y_C}{\Delta t} \tag{13.14}$$

Therefore, the CFL condition requires

$$a_C + a_C^\circ \leq 0 \quad \Rightarrow \quad \rho_C^\circ u_C^\circ \Delta y_C - \frac{\rho_C^\circ \Delta x_C \Delta y_C}{\Delta t} \leq 0 \tag{13.15}$$

or

$$\Delta t \leq \frac{\Delta x_C}{u_C^\circ}. \tag{13.16}$$

For convection dominated flows, defining a *CFL* number as

$$CFL^{conv} = \frac{\left|\mathbf{v}_C^\circ\right| \Delta t}{\Delta x_C} \tag{13.17}$$

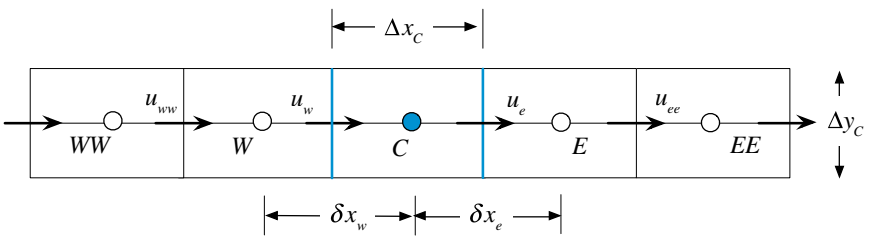

Fig. 13.5 A portion of the discretized domain for a one dimensional convection problem

implies that for numerical stability the *CFL* number should satisfy

$$CFL^{conv} \leq 1. \tag{13.18}$$

13.2.2.2 Stability of a Transient-Diffusion Case

For pure diffusion problems, the expression for the *CFL* number is different. For that purpose, the one dimensional pure diffusion problem schematically depicted in Fig. (13.6) is considered.

The a_C and a_C° coefficients in the discretized equation of element C using linear interpolation profiles are given by

$$a_C = \frac{\Gamma_e^\phi \Delta y_C}{\delta x_e} + \frac{\Gamma_w^\phi \Delta y_C}{\delta x_w} \qquad a_C^\circ = -\frac{\rho_C^\circ V_C}{\Delta t} = -\frac{\rho_C^\circ \Delta x_C \Delta y_C}{\Delta t} \tag{13.19}$$

Therefore, the CFL condition requires

$$a_C + a_C^\circ \leq 0 \quad \Rightarrow \quad \frac{\Gamma_e^\phi \Delta y_C}{\delta x_e} + \frac{\Gamma_w^\phi \Delta y_C}{\delta x_w} - \frac{\rho_C^\circ \Delta x_C \Delta y_C}{\Delta t} \leq 0 \tag{13.20}$$

or

$$\Delta t \leq \frac{\rho_C^\circ \Delta x_C}{\frac{\Gamma_e^\phi}{\delta x_e} + \frac{\Gamma_w^\phi}{\delta x_w}}. \tag{13.21}$$

For the case when the grid is uniform and the diffusion coefficient is constant, Eq. (13.21) becomes

$$\Delta t \leq \frac{\rho_C^\circ (\Delta x_C)^2}{2\Gamma_C^\phi}. \tag{13.22}$$

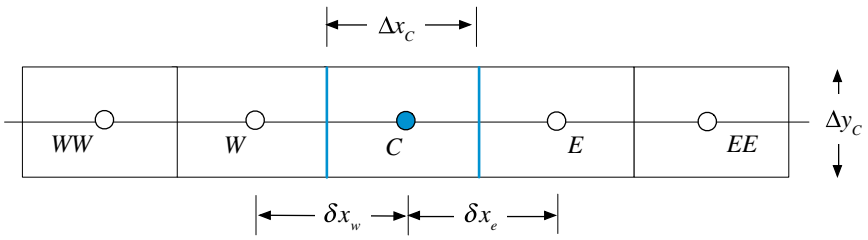

Fig. 13.6 A portion of the discretized domain for a one dimensional diffusion problem

For diffusion dominated problems, a *CFL* number is defined as

$$CFL^{diff} = \frac{\Gamma_C^{\phi} \Delta t}{\rho_C^{\circ} (\Delta x_C)^2} \tag{13.23}$$

implying that for stability the following condition should be satisfied:

$$CFL^{diff} \leq \frac{1}{2}. \tag{13.24}$$

13.2.2.3 Stability of a Transient-Convection-Diffusion Case

For the case of a multi dimensional unsteady convection and diffusion problem (Fig. 13.7) and based on the derivations presented in Chap. 12, the coefficients in Eq. (13.14) are given by

$$\begin{aligned} a_C^{\circ} &= -\frac{\rho_C^{\circ} V_C}{\Delta t} \\ a_C &= \sum_{f \sim nb(C)} \left(\Gamma_f^{\phi} \frac{E_f}{d_{CF}} + \left\| \dot{m}_f^{\circ}, 0 \right\| \right) \end{aligned} \tag{13.25}$$

Substituting the expressions for the coefficients from Eq. (13.25) in Eq. (13.13), the CFL condition becomes

$$\sum_{f \sim nb(C)} \left(\Gamma_f^{\phi} \frac{E_f}{d_{CF}} + \left\| \dot{m}_f^{\circ}, 0 \right\| \right) - \frac{\rho_C^{\circ} V_C}{\Delta t} \leq 0 \tag{13.26}$$

leading to the following constraint on the time step:

$$\Delta t \leq \frac{\rho_C^{\circ} V_C}{\sum_{f \sim nb(C)} \left(\Gamma_f^{\phi} \frac{E_f}{d_{CF}} + \left\| \dot{m}_f^{\circ}, 0 \right\| \right)}. \tag{13.27}$$

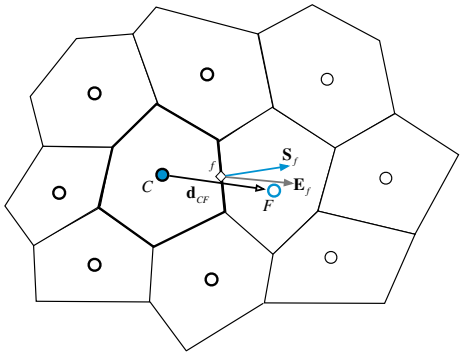

Fig. 13.7 A portion of the discretized domain for a multi dimensional convection problem

Equation (13.27) is the general requirement for stability of explicit transient schemes. In fact the conditions obtained earlier for pure convection and pure diffusion in one dimensional domains can be derived as special cases of Eq. (13.27). For the case of a one dimensional diffusion problem with a uniform grid of cell size Δx, constant density ρ, and a uniform diffusion coefficient Γ^{ϕ}, Eq. (13.27) reduces to

$$\Delta t \leq \frac{\rho_C^{\circ} \overbrace{V_C}^{\Delta x_C \Delta y_C}}{\sum_{f \sim nb(C)} \left(\underbrace{\Gamma_f^{\phi}}_{=\Gamma_e^{\phi}+\Gamma_w^{\phi}} \underbrace{\frac{E_f}{d_{CF}}}_{\Delta x_C} + \underbrace{\left\| \dot{m}_f^{\circ}, 0 \right\|}_{=0} \right)} \Rightarrow \Delta t \leq \frac{\rho_C^{\circ} (\Delta x_C)^2}{2\Gamma_C^{\phi}}. \tag{13.28}$$

While for the case of a one dimensional advection problem discretized using the upwind scheme and with the flow moving from left to right, Eq. (13.27) reduces to

$$\Delta t \leq \frac{\rho_C^{\circ} \overbrace{V_C}^{=\Delta x_C \Delta y_C}}{\sum_{f \sim nb(C)} \left(\underbrace{\Gamma_f^{\phi}}_{=0} \frac{E_f}{d_{CF}} + \underbrace{\left\| \dot{m}_f^{\circ}, 0 \right\|}_{\dot{m}_e^{\circ} = \rho_C^{\circ} u_C^{\circ} \Delta y_C} \right)} \Rightarrow \Delta t \leq \frac{\Delta x_C}{u_C^{\circ}}. \tag{13.29}$$

This stability constraint is stringent and very restrictive as it forces the use of extremely small time steps when solving transient problems. That is, whereas the computational cost at each time step is small in comparison to what would be required to solve a system of equations at that level, the imposed limitation by the CFL condition necessitates a larger number of steps to move the solution in time. Therefore the benefit of reducing the calculations per time step is lost by the much larger number of time steps required. Moreover, Eq. (13.27) indicates also that improving the spatial accuracy by decreasing the grid size, decreases further the maximum time step size that can be used without causing instabilities.

As shown next, such a constraint does not apply to implicit schemes for which the transient term has always the proper sign.

13.2.3 Backward Euler Scheme

To derive the backward Euler scheme, the value of the function T at time $t - \Delta t$ is expressed using a Taylor series using the values of T and its derivates at time t as

$$T(t-\Delta t) = T(t) - \frac{\partial T(t)}{\partial t}\Delta t + \frac{\partial^2 T(t)}{\partial t^2}\frac{\Delta t^2}{2!} + \cdots \tag{13.30}$$

Manipulating Eq. (13.30), an equation for the first derivative is obtained as

$$\frac{\partial T(t)}{\partial t} = \frac{T(t) - T(t-\Delta t)}{\Delta t} + \frac{\partial^2 T(t)}{\partial t^2}\frac{\Delta t}{2!} + \cdots \tag{13.31}$$

Replacing T by $(\rho\phi)$ in Eq. (13.31) and substituting the resulting expression for the derivative in Eq. (13.3), the discretized equation becomes

$$\frac{(\rho_C\phi_C)^t - (\rho_C\phi_C)^{t-\Delta t}}{\Delta t} V_C + L\left(\phi_C^t\right) = 0. \tag{13.32}$$

Then invoking the algebraic relation of the spatial operator and the suggested notation, the complete algebraic form of the transient scalar equation is obtained as

$$\left(a_C^{\bullet} + a_C\right)\phi_C + \sum_{F\sim NB(C)} a_F\phi_F = b_C + a_C^{\circ}\phi_C^{\circ} \tag{13.33}$$

with the coefficients given by

$$\begin{aligned} a_C^{\bullet} &= \frac{\rho_C V_C}{\Delta t} \\ a_C^{\circ} &= -\frac{\rho_C^{\circ} V_C}{\Delta t} \end{aligned} \tag{13.34}$$

The stencil for Eq. (13.33) is shown in Fig. 13.8. It is clear that with the spatial operator evaluated at the same time level as the new temporal coefficient, resolving the ϕ field at

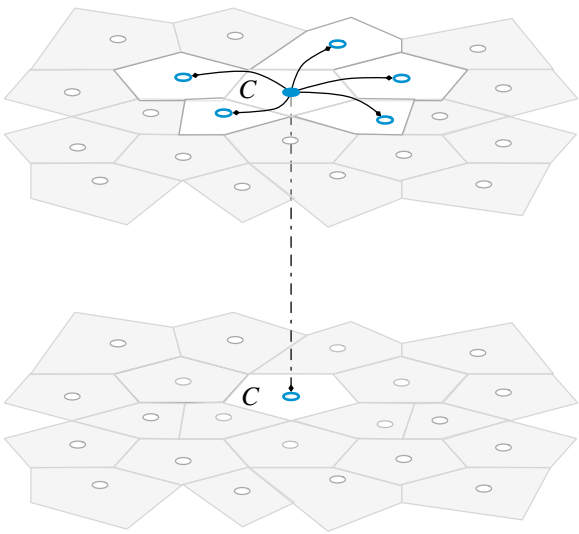

Fig. 13.8 Stencil for the backward Euler stencil

a new time level requires solving a system of equations. This type of schemes requiring the solution of a system of equations is denoted by implicit schemes [5–12].

As can be inferred from Eq. (13.34) a_C and a_C° are of opposite signs guaranteeing that ϕ_C is bounded by the values of its spatial neighbors at the current time step t and by the value of its temporal neighbor at the previous time step $t - \Delta t$. This implies that the scheme is always stable independent of the time step used, allowing for the solution to proceed rapidly by using large time steps. Nonetheless this is not the ideal scheme as it is of low order and solutions obtained with this scheme are of low accuracy unless small time steps are used, which puts its use in a quandary. Adopting large time steps for computational efficiency results in a solution of low accuracy and using small time steps for higher accuracy is associated with low computational efficiency.

13.2.4 Crank-Nicolson Scheme

In the Crank-Nicolson scheme [2, 14] a more accurate representation of the transient term is derived by expressing the values of the function T at times $t - \Delta t$ and $t + \Delta t$ in terms of the values of T and its derivates at time t as

$$\begin{aligned} T(t+\Delta t) &= T(t) + \frac{\partial T(t)}{\partial t}\Delta t + \frac{\partial^2 T(t)}{\partial t^2}\frac{\Delta t^2}{2!} + \frac{\partial^3 T(t)}{\partial t^3}\frac{\Delta t^3}{3!} + \cdots \\ T(t-\Delta t) &= T(t) - \frac{\partial T(t)}{\partial t}\Delta t + \frac{\partial^2 T(t)}{\partial t^2}\frac{\Delta t^2}{2!} - \frac{\partial^3 T(t)}{\partial t^3}\frac{\Delta t^3}{3!} + \cdots \end{aligned} \tag{13.35}$$

Then, subtracting $T(t + \Delta t)$ from $T(t - \Delta t)$ given in Eq. (13.35), an equation for the first derivative is obtained as

$$\frac{\partial T(t)}{\partial t} = \frac{T(t+\Delta t) - T(t-\Delta t)}{2\Delta t} + O\left(\Delta t^2\right) \tag{13.36}$$

Note that the order of accuracy of the derivative is now $O(\Delta t^2)$ since the second order derivative is completely eliminated.

Substituting the time derivative given by Eq. (13.36) into Eq. (13.3) yields

$$\frac{\left(\rho_C \phi_C\right)^{t+\Delta t} - \left(\rho_C \phi_C\right)^{t-\Delta t}}{2\Delta t} V_C + L\left(\phi_C^t\right) = 0 \tag{13.37}$$

Then invoking the algebraic relation of the spatial operator, and using the suggested notation, the complete algebraic form of the transient scalar equation is obtained as

$$a_C^\bullet \phi_C = b_C - \left(a_C \phi_C^\circ + \sum_{F \sim NB(C)} a_F \phi_F^\circ \right) - a_C^{\circ\circ} \phi_C^{\circ\circ} \tag{13.38}$$

with the coefficients given by

$$\begin{aligned} a_C^{\bullet} &= \frac{\rho_C V}{2\Delta t} \\ a_C^{\circ\circ} &= -\frac{\rho_C^{\circ\circ} V}{2\Delta t} \end{aligned} \tag{13.39}$$

The stencil for Eq. (13.38) is shown in Fig. 13.9. It is clear that the scheme is an explicit type scheme, since the evaluation of $(\rho\phi)^{t+\Delta t}$ can be performed using only old values. However two old levels are now needed, with the spatial operator being evaluated at one of these levels.

An analysis of the stability of the CN scheme can be performed after slightly modifying the original equation. Using the following approximation:

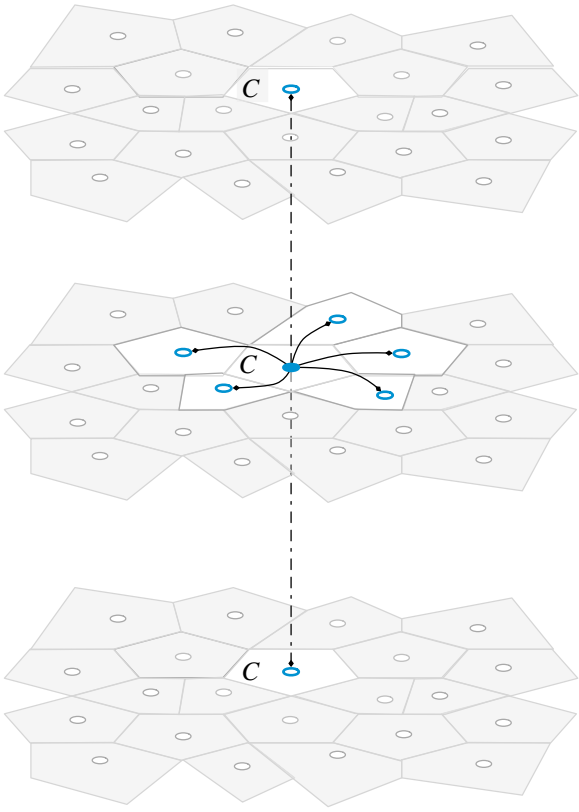

Fig. 13.9 Stencil of the Crank Nicholson Scheme

$$\phi^{\circ} \approx \frac{\phi + \phi^{\circ\circ}}{2} \tag{13.40}$$

the algebraic equation [Eq. (13.38)] becomes

$$a_C^{\bullet}\phi_C + 0.5\left(a_C\phi_C + \sum_{F\sim NB(C)} a_F\phi_F\right) = b_C - 0.5\left(\left(a_C + 2a_C^{\circ\circ}\right)\phi_C^{\circ\circ} + \sum_{F\sim NB(C)} a_F\phi_F^{\circ\circ}\right) \tag{13.41}$$

Thus, the stability condition becomes

$$a_C + 2a_C^{\circ\circ} \leq 0. \tag{13.42}$$

For the one dimensional transient advection problem displayed in Fig. 13.5, Eq. (13.42) results in

$$\Delta t \leq \frac{2\rho_C^{\circ\circ}V_C}{\dot{m}_e^{\circ}} = \frac{2\rho_C^{\circ\circ}\Delta x_C\Delta y_C}{\rho_C^{\circ}u_C^{\circ}\Delta y_C} \approx \frac{2\Delta x_C}{\left|\mathbf{v}_e^{\circ}\right|}, \tag{13.43}$$

where it has been assumed that the advection term is discretized using the upwind scheme. Using the *CFL* number for convection defined above, Eq. (13.43) is expressed as

$$CFL^{conv} \leq 2 \tag{13.44}$$

The larger CFL limitation is pleasing, but the improved accuracy is just more important as it allows for accurate solutions to be achieved without the need to resort to very small time steps, especially that the second order derivative is now eliminated from the error. More details on accuracy analysis will be presented in later sections.

13.2.5 Implementation Details

The CN scheme can also be derived by summing the Forward and Backward transient Euler schemes [4], as shown next.

$$\text{Forward Euler} \rightarrow \frac{(\rho_C\phi_C)^t - (\rho_C\phi_C)^{t-\Delta t}}{\Delta t}V_C = -L\left(\phi_C^t\right) \tag{13.45}$$

$$\text{Backward Euler} \rightarrow \frac{(\rho_C\phi_C)^{t+\Delta t}-(\rho_C\phi_C)^{t}}{\Delta t}V_C = -L\left(\phi_C^t\right) \tag{13.46}$$

Forward Euler + Backward Euler:

$$\begin{aligned}
&\rightarrow \frac{(\rho_C\phi_C)^{t}-(\rho_C\phi_C)^{t-\Delta t}}{\Delta t}V_C + \frac{(\rho_C\phi_C)^{t+\Delta t}-(\rho_C\phi_C)^{t}}{\Delta t}V_C = -L\left(\phi_C^t\right) - L\left(\phi_C^t\right)\\
&\rightarrow \frac{(\rho_C\phi_C)^{t+\Delta t}-(\rho_C\phi_C)^{t-\Delta t}}{2\Delta t}V_C + L\left(\phi_C^t\right) = 0\\
&\rightarrow \text{Crank} - \text{Nicolson}
\end{aligned} \tag{13.47}$$

This formulation points to a simple implementation of the CN scheme within an implicit scheme framework as a two-step procedure. In the first step a Backward Euler formulation is used to implicitly find $(\rho\phi)^t$ from

$$(\rho_C\phi_C)^t + \frac{\Delta t}{V_C}L\left(\phi_C^t\right) = (\rho_C\phi_C)^{t-\Delta t} \tag{13.48}$$

while in the second step the CN value at time step $t + \Delta t$ is found explicitly as

$$\begin{aligned}
\frac{(\rho_C\phi_C)^{t+\Delta t}-(\rho_C\phi_C)^{t}}{\Delta t}V_C &= -L\left(\phi_C^t\right) = \frac{(\rho_C\phi_C)^{t}-(\rho_C\phi_C)^{t-\Delta t}}{\Delta t}V_C\\
&\Rightarrow (\rho_C\phi_C)^{t+\Delta t} = 2(\rho_C\phi_C)^{t}-(\rho_C\phi_C)^{t-\Delta t}
\end{aligned} \tag{13.49}$$

In this derivation it was assumed that the transient time step Δt is divided into two equal local time steps (Δt_{local}), with Δt_{local} equals half the set time step Δt.

It is important to note that while the CN scheme is second order accurate, it is still an explicit scheme, which is constrained by a CFL like condition, as explained above.

13.2.6 Adams-Moulton Scheme

The development of the second order Adams-Moulton scheme [15, 16] requires expanding the values of T at $t - \Delta t$ and $t - 2\Delta t$ using Taylor series expansions around t, yielding

$$\begin{aligned}
T(t-2\Delta t) &= T(t) - \frac{\partial T(t)}{\partial t}2\Delta t + \frac{\partial^2 T(t)}{\partial t^2}\frac{4\Delta t^2}{2!} + \cdots\\
T(t-\Delta t) &= T(t) - \frac{\partial T(t)}{\partial t}\Delta t + \frac{\partial^2 T(t)}{\partial t^2}\frac{\Delta t^2}{2!} + \cdots
\end{aligned} \tag{13.50}$$

The first derivative is obtained by combining the two equations in such a way that the second order derivative is eliminated, resulting in the following equation:

$$\frac{\partial T(t)}{\partial t} = \frac{3T(t) - 4T(t-\Delta t) + T(t-2\Delta t)}{2\Delta t} \tag{13.51}$$

which, upon substituting in Eq. (13.3), yields

$$\frac{3(\rho_C\phi_C)^t - 4(\rho_C\phi_C)^{t-\Delta t} + (\rho_C\phi_C)^{t-2\Delta t}}{2\Delta t} V_C + L\left(\phi_C^t\right) = 0 \tag{13.52}$$

Expanding the spatial term, the final form of the algebraic equation is obtained as

$$\left(a_C^{\bullet} + a_C\right)\phi_C + \sum_{F \sim NB(C)} a_F\phi_F = b_C - a_C^{\circ}\phi_C^{\circ} - a_C^{\circ\circ}\phi_C^{\circ\circ} \tag{13.53}$$

with the coefficients given by

$$\begin{aligned} a_C^{\bullet} &= \frac{3\rho_C V_C}{2\Delta t} \\ a_C^{\circ} &= -\frac{2\rho_C^{\circ} V_C}{\Delta t} \\ a_C^{\circ\circ} &= \frac{\rho_C^{\circ\circ} V_C}{2\Delta t} \end{aligned} \tag{13.54}$$

It is clear that the $a_C^{\circ\circ}$ coefficient has a positive sign implying that an increase in $\phi_C^{\circ\circ}$ would lead to a decrease in ϕ_C. This is mitigated by the large a_C° coefficient, which has the right influence. Thus while the scheme is stable, it is not bounded with unphysical oscillations expected in certain circumstances.

Example 1

The thermal conductivity of a solid sphere of volume 1 m^3 *is so high that its resistance to conduction is very small as compared to its resistance to convection heat transfer with the surroundings. Thus temperature gradients within the sphere are negligible and the temperature of the sphere is spatially uniform at any instant. The initial temperature of the sphere is* T_h *and that of the surroundings is* T_∞*. The density, specific heat, sphere surface area, and convection heat transfer coefficient with the surroundings are* ρ, c, A_s, *and* h_∞, *respectively. Neglecting heat transfer by radiation, the energy equation for the sphere is given by*

$$\rho c V \frac{dT}{dt} = -h_\infty A_S (T - T_\infty)$$

Defining a dimensionless temperature as

$$\phi = \frac{T - T_\infty}{T_h - T_\infty}$$

the energy equation and initial condition become

$$\frac{d\phi}{dt} = -\frac{h_\infty A_S}{\rho c V}\phi \text{ and } \phi(0) = 1$$

For a value of $h_\infty A_S/\rho cV = 1$, *compare dimensionless temperature values obtained analytically with numerical ones generated using the first order explicit scheme, the first order implicit scheme, and the second order CN scheme applied via the two-step procedure at times* 0.1, 0.2, *and* 0.3 *using a time step with size of* 0.1.

Solution

The governing equation reduces to

$$\frac{d\phi}{dt} = -\phi$$

subject to

$$\phi(0) = 1$$

By separation of variables and application of the initial condition the analytical solution is found as

$$\phi(t) = e^{-t}$$

Thus the analytical solution at times 0.1, 0.2, and 0.3 are given by

$$\begin{aligned}\phi_{exact}(0.1) &= e^{-0.1} = 0.9048\\ \phi_{exact}(0.2) &= e^{-0.2} = 0.8187\\ \phi_{exact}(0.3) &= e^{-0.3} = 0.7408\end{aligned}$$

The numerical solution is obtained with $V = 1$, $L(\phi^n) = -\phi^n$, and $L(\phi^{n+1}) = -\phi^{n+1}$.

The error in the numerical solution is found using

$$error = |\phi_{numerical} - \phi_{exact}|$$

Numerical solution using the first order explicit scheme

$$\phi^{t+\Delta t} = (1 - \Delta t)\phi^t$$

$$\left.\begin{aligned}\phi_{explicit}(0.1) &= (1-0.1)\phi(0) = 0.9*1 = 0.9\\ \phi_{explicit}(0.2) &= (1-0.1)\phi(0.1) = 0.9*0.9 = 0.81\\ \phi_{explicit}(0.3) &= (1-0.1)\phi(0.2) = 0.9*0.81 = 0.729\end{aligned}\right\} \Rightarrow \begin{cases} error_{explicit}(0.1) = 4.8\times 10^{-3}\\ error_{explicit}(0.2) = 8.7\times 10^{-3}\\ error_{explicit}(0.3) = 1.18\times 10^{-2}\end{cases}$$

Numerical solution using the first order implicit scheme

$$\phi^{t+\Delta t} = \frac{1}{1+\Delta t}\phi^{t}$$

$$\left.\begin{aligned}\phi_{implicit}(0.1) &= \frac{1}{1+0.1}(1) = 0.9091\\ \phi_{implicit}(0.2) &= \frac{1}{1+0.1}(0.9091) = 0.8264\\ \phi_{implicit}(0.3) &= \frac{1}{1+0.1}(0.8264) = 0.7513\end{aligned}\right\} \Rightarrow \begin{cases} error_{implicit}(0.1) = 4.3\times 10^{-3}\\ error_{implicit}(0.2) = 7.7\times 10^{-3}\\ error_{implicit}(0.3) = 1.05\times 10^{-2}\end{cases}$$

Numerical solution using the second order CN scheme

In this case the solution is obtained using Eqs. (13.48) and (13.49) that are reduced to

$$\begin{aligned}\phi^{*}(t+\Delta t/2) &= \frac{1}{1+\Delta t/2}\phi(t)\\ \phi_{CN}(t+\Delta t) &= 2\phi^{*}(t+\Delta t/2) - \phi(t)\end{aligned}$$

where the total time step Δt has been divided into two equal time steps of value $\Delta t/2$. Applying the above equations, the solutions are found as

$$\left.\begin{aligned}\phi^{*}(0.05) &= \frac{1}{1+0.05}\phi(0) = 0.95238\\ \phi_{CN}(0.1) &= 2\phi^{*}(0.05) - \phi(0) = \boxed{0.90476}\\ \phi^{*}(0.15) &= \frac{1}{1+0.05}\phi(0.1) = 0.861678\\ \phi_{CN}(0.2) &= 2\phi^{*}(0.15) - \phi(0.1) = \boxed{0.81859}\\ \phi^{*}(0.25) &= \frac{1}{1+0.05}\phi(0.2) = 0.779615\\ \phi_{CN}(0.3) &= 2\phi^{*}(0.25) - \phi(0.2) = \boxed{0.7406}\end{aligned}\right\} \Rightarrow \begin{cases} error_{CN}(0.1) = 7.551\times 10^{-5}\\ error_{CN}(0.2) = 1.366\times 10^{-4}\\ error_{CN}(0.3) = 1.854\times 10^{-4}\end{cases}$$

13.3 The Finite Volume Approach

The Finite Volume approach for the discretization of the transient term is very similar to the discretization of the convective term [4], except that the integration is carried over temporal rather than spatial element (Fig. 13.10).

Integration of Eq. (13.3) over the time interval $[t - \Delta t/2,\ t + \Delta t/2]$ yields

$$\underbrace{\int_{t-\Delta t/2}^{t+\Delta t/2} \frac{\partial(\rho_C \phi_C)}{\partial t} V_C dt}_{Term\ I} + \underbrace{\int_{t-\Delta t/2}^{t+\Delta t/2} L(\phi_C) dt}_{Term\ II} = 0 \tag{13.55}$$

With V_C treated as a constant, *Term I* turned into a difference of face fluxes, and *Term II* evaluated as a volume integral using the mid point rule, Eq. (13.55) becomes

$$V_C(\rho_C \phi_C)^{t+\Delta t/2} - V_C(\rho_C \phi_C)^{t-\Delta t/2} + L\left(\phi_C^t\right)\Delta t = 0 \tag{13.56}$$

Equation (13.56) is the semi-discretized transient equation, which can be written in the more standard form by dividing all terms by the temporal element volume, Δt, leading to

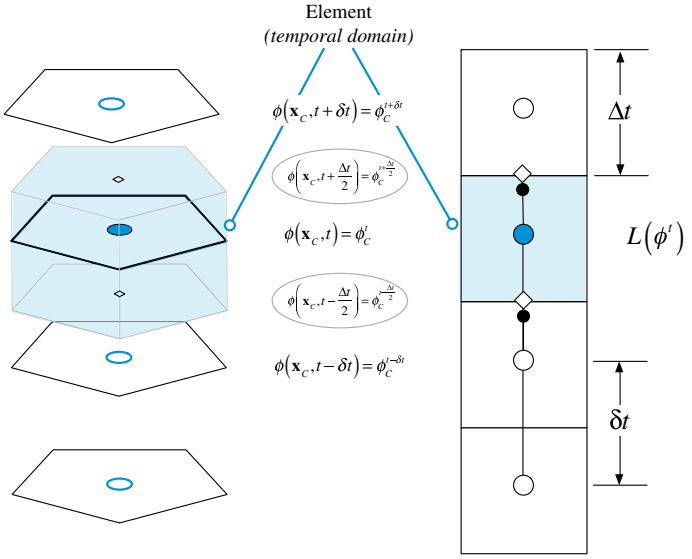

Fig. 13.10 Element in the transient domain

$$\frac{(\rho_C\phi_C)^{t+\Delta t/2}-(\rho_C\phi_C)^{t-\Delta t/2}}{\Delta t}V_C + L\left(\phi_C^t\right) = 0 \tag{13.57}$$

To derive the full discretized equation, an interpolation profile expressing the face values at $(t-\Delta t/2)$ and $(t+\Delta t/2)$ in terms of the element values at (t), $(t-\Delta t)$, etc., is needed. The selection of this profile can heavily rely on the understanding gained from the discretization of the convection term. The choice will obviously affect the accuracy and robustness of the method. In that regard, it is worth mentioning that the integration of the spatial operator is second order in time, but the accuracy of the operator itself is determined by the options used during its discretization.

Independent of the profile used, the flux will be linearized based on old and new values as

$$FluxT = FluxC\phi_C + FluxC^\circ\phi_C^\circ + FluxV \tag{13.58}$$

where again superscript ° refers to old values. With the linearization completed, the coefficients of the algebraic equation can then be assembled into

$$\begin{aligned} a_C &\leftarrow a_C + FluxC \\ b_C &\leftarrow b_C - FluxC^\circ\phi_C^\circ - FluxV \end{aligned} \tag{13.59}$$

In what follows the discretization for a number of interpolation profiles is presented.

13.3.1 First Order Transient Schemes

The first order implicit and explicit Euler schemes will be constructed next by adopting an upwind [14, 17] and a downwind [4, 18] transient interpolation profile, respectively.

13.3.2 First Order Implicit Euler Scheme

The transient first order implicit Euler scheme is obtained by using a first-order "upwind" interpolation profile [14, 17]. As shown in Fig. 13.11, the value of $\rho\phi$ at the temporal element face is set equal to the value at the centroid of the upwind element to give

$$(\rho_C\phi_C)^{t+\Delta t/2} = (\rho_C\phi_C)^t \text{ and } (\rho_C\phi_C)^{t-\Delta t/2} = (\rho_C\phi_C)^{t-\Delta t} \tag{13.60}$$

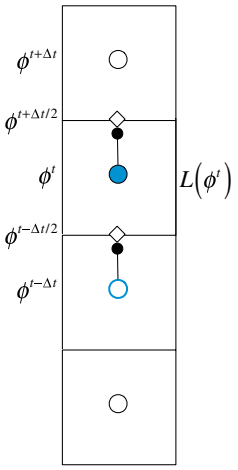

Fig. 13.11 First order transient upwind interpolation profile resulting in the implicit first order transient Euler scheme

Using Eq. (13.60), Eq. (13.57) becomes

$$\frac{(\rho_C\phi_C)^t-(\rho_C\phi_C)^{t-\Delta t}}{\Delta t}V_C+L\left(\phi_C^t\right)=0 \tag{13.61}$$

which is the first order implicit Euler scheme. The scheme is linearized as follows:

$$\begin{aligned} FluxC &= \frac{\rho_C V_C}{\Delta t} \\ FluxC^{\circ} &= -\frac{\rho_C^{\circ} V_C}{\Delta t} \\ FluxV &= 0 \end{aligned} \tag{13.62}$$

13.3.2.1 Numerical Diffusion

As this is a first order scheme, it is expected, based on the knowledge gained from convection schemes, to produce numerical diffusion. Its value can be determined by trying to recover the original governing equation using a Taylor series expansion around time t. The value of $(\rho\phi)^{t-\Delta t}$ can be expressed as

$$(\rho\phi)^{t-\Delta t}=(\rho\phi)^t-\left.\frac{\partial(\rho\phi)}{\partial t}\right|_t\Delta t+\left.\frac{\partial^2(\rho\phi)}{\partial t^2}\right|_t\frac{\Delta t^2}{2}+O\left(\Delta t^3\right) \tag{13.63}$$

which can be rearranged into

$$\frac{(\rho\phi)^t-(\rho\phi)^{t-\Delta t}}{\Delta t}=\left.\frac{\partial(\rho\phi)}{\partial t}\right|_t-\underbrace{\left(\frac{\Delta t}{2}\right)\left.\frac{\partial^2(\rho\phi)}{\partial t^2}\right|_t}_{\substack{\textit{Numerical}\\ \textit{diffusion}\\ \textit{term}}}-O\left(\Delta t^2\right) \tag{13.64}$$

Substituting Eq. (13.64) into the discretized equation gives

$$\left.\frac{\partial(\rho\phi)}{\partial t}\right|_t+\frac{1}{V_C}L\left(\phi_C^t\right)=\underbrace{\left(\frac{\Delta t}{2}\right)\left.\frac{\partial^2(\rho\phi)}{\partial t^2}\right|_t}_{\substack{\textit{Numerical}\\ \textit{diffusion}\\ \textit{term}}}+O\left(\Delta t^2\right) \tag{13.65}$$

In effect a numerical diffusion term has been added to the equation that scales with the time step in a similar fashion to the upwind scheme for the advection term. So while the scheme is unconditionally stable, the solution it yields is really a stationary solution for large time steps.

13.3.3 First Order Explicit Euler Scheme

The transient first order explicit Euler scheme is obtained by using a first-order "downwind" interpolation profile [4, 18]. As shown in Fig. 13.12, the value of $\rho\phi$ at

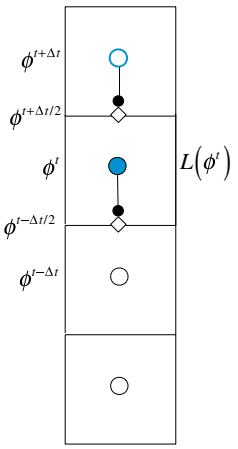

Fig. 13.12 First order transient downwind interpolation profile resulting in the explicit first order transient Euler scheme

the temporal element face is set equal to the value at the downwind element centroid, yielding

$$(\rho_C\phi_C)^{t+\Delta t/2} = (\rho_C\phi_C)^{t+\Delta t} \quad \text{and} \quad (\rho_C\phi_C)^{t-\Delta t/2} = (\rho_C\phi_C)^{t} \tag{13.66}$$

Using Eq. (13.66), Eq. (13.57) becomes

$$\frac{(\rho_C\phi_C)^{t+\Delta t} - (\rho_C\phi_C)^{t}}{\Delta t} V_C + L\left(\phi_C^t\right) = 0 \tag{13.67}$$

which is the first order explicit Euler scheme. The scheme is linearized as follows:

$$\begin{aligned} FluxC &= \frac{\rho_C V_C}{\Delta t} \\ FluxC^{\circ} &= -\frac{\rho_C^{\circ} V_C}{\Delta t} \\ FluxV &= 0 \end{aligned} \tag{13.68}$$

Note that now the new time is at $t + \Delta t$ and that the spatial operator of Eq. (13.67) has to be evaluated at time t. Thus, it is possible to evaluate the right hand side completely and find the value of $\rho\phi$ at time $t + \Delta t$ without the need to solve a set of linear algebraic equations. This is the explicit scheme and corresponds to the assumption that $\rho\phi$ prevails over the entire time step.

13.3.3.1 Numerical Anti-Diffusion

Again performing a simple Taylor expansion around time t yields

$$(\rho\phi)^{t+\Delta t} = (\rho\phi)^{t} + \left.\frac{\partial(\rho\phi)}{\partial t}\right|_t \Delta t + \left.\frac{\partial^2(\rho\phi)}{\partial t^2}\right|_t \frac{\Delta t^2}{2} + O\left(\Delta t^3\right) \tag{13.69}$$

which can be rewritten as

$$\frac{(\rho\phi)^{t+\Delta t} - (\rho\phi)^{t}}{\Delta t} = \left.\frac{\partial(\rho\phi)}{\partial t}\right|_t + \left(\frac{\Delta t}{2}\right)\left.\frac{\partial^2(\rho\phi)}{\partial t^2}\right|_t + O\left(\Delta t^2\right) \tag{13.70}$$

Substitution into Eq. (13.67) gives

$$\left.\frac{\partial(\rho\phi)}{\partial t}\right|_t + \frac{1}{V_C} L\left(\phi_C^t\right) = \underbrace{-\left(\frac{\Delta t}{2}\right)\left.\frac{\partial^2(\rho\phi)}{\partial t^2}\right|_t}_{\substack{Numerical \\ anti-diffusion \\ term}} + O\left(\Delta t^2\right) \tag{13.71}$$

Where now the second order differential term has a negative sign, akin to a negative diffusion or anti-diffusion, with compression effects on profiles, very similar to the Downwind scheme in advection. Again the anti-diffusion term scales with the time step. When used in combination with an upwind convection scheme and a Courant number of 1, it can be shown that the numerical diffusion of the advection scheme and the numerical anti-diffusion of the explicit Euler scheme for a CFL^{conv} equals to 1 are of equal magnitudes and of opposite signs. Thus they cancel each other producing nearly an exact solution. Nonetheless this is not practical as ensuring a CFL^{conv} of 1 on anything but simple one dimensional grids is not an option for real problems.

A related issue to the anti-diffusion behavior is numerical instabilities, which increases with increasing Δt placing a very strong restriction on the time step. This can be evaluated by applying the negative neighboring coefficient rule.

13.3.4 Second Order Transient Euler Schemes

Similar to advection schemes, second order transient schemes can be constructed with a linear interpolation profile. The choice could be a symmetric profile (central difference) yielding the Crank-Nicolson (CN) scheme [2], or an upwind one (second order upwind scheme) [4, 19, 20] resulting in the Adams-Moulton scheme [15, 16], an implicit scheme also known as the Second Order Upwind Euler (SOUE).

13.3.5 Crank-Nicholson (Central Difference Profile)

With the $\rho\phi$ computed using linear interpolation between the "Upwind" and "Downwind" nodes, the Crank-Nicholson scheme shown in Fig. 13.13 is obtained.

For a uniform time step, this is expressed mathematically as

$$\begin{aligned}(\rho_C\phi_C)^{t+\Delta t/2} &= \frac{1}{2}(\rho_C\phi_C)^{t+\Delta t}+\frac{1}{2}(\rho_C\phi_C)^{t}\\ (\rho_C\phi_C)^{t-\Delta t/2} &= \frac{1}{2}(\rho_C\phi_C)^{t}+\frac{1}{2}(\rho_C\phi_C)^{t-\Delta t}\end{aligned} \tag{13.72}$$

Substituting in Eq. (13.57), the discretized equation becomes

$$\frac{(\rho_C\phi_C)^{t+\Delta t}-(\rho_C\phi_C)^{t-\Delta t}}{2\Delta t}V_C + L\left(\phi_C^t\right) = 0 \tag{13.73}$$

The linearization coefficients for the CN scheme can be written as

$$\begin{aligned}FluxC &= \frac{\rho_C V_C}{2\Delta t}\\ FluxC^{\circ} &= 0\\ FluxV &= -\frac{\rho_C^{\circ\circ} V_C}{2\Delta t}\phi_C^{\circ\circ}\end{aligned} \tag{13.74}$$

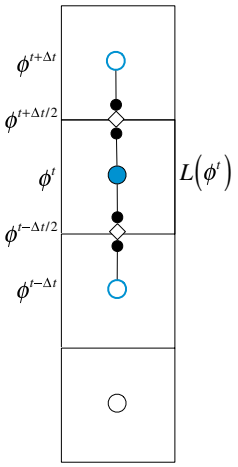

Fig. 13.13 Second order transient central difference interpolation profile resulting in the transient CN scheme

The stencil shown in Fig. 13.9 indicates that the scheme is explicit with the value at level $t + \Delta t$ computed explicitly from the values at times t and $t - \Delta t$. Thus its stability is constrained by a CFL limit.

Again in a similar fashion to the finite difference formulation, it can be reformulated in a two-step procedure using Eqs. (13.48) and (13.49), i.e., a first order implicit Euler step followed by a modified explicit Euler step in the form of extrapolation.

13.3.5.1 Numerical Accuracy

Expanding $(\rho\phi)$ at $t + \Delta t$ and $t - \Delta t$ via Taylor expansions around time t yields

$$(\rho\phi)^{t+\Delta t} = (\rho\phi)^t + \left.\frac{\partial(\rho\phi)}{\partial t}\right|_t \Delta t + \left.\frac{\partial^2(\rho\phi)}{\partial t^2}\right|_t \frac{\Delta t^2}{2} + \left.\frac{\partial^3(\rho\phi)}{\partial t^3}\right|_t \frac{\Delta t^3}{6} + O(\Delta t^4) \quad (13.75)$$

$$(\rho\phi)^{t-\Delta t} = (\rho\phi)^t - \left.\frac{\partial(\rho\phi)}{\partial t}\right|_t \Delta t + \left.\frac{\partial^2(\rho\phi)}{\partial t^2}\right|_t \frac{\Delta t^2}{2} - \left.\frac{\partial^3(\rho\phi)}{\partial t^3}\right|_t \frac{\Delta t^3}{6} + O(\Delta t^4) \quad (13.76)$$

Subtracting Eq. (13.75) from Eq. (13.76), the following equation is obtained:

$$\frac{(\rho\phi)^{t+\Delta t} - (\rho\phi)^{t-\Delta t}}{2\Delta t} = \left.\frac{\partial(\rho\phi)}{\partial t}\right|_t + \left.\frac{\partial^3(\rho\phi)}{\partial t^3}\right|_t \frac{\Delta t^2}{6} - O(\Delta t^3) \quad (13.77)$$

Substitution into Eq. (13.73) gives

$$\left.\frac{\partial(\rho\phi)}{\partial t}\right|_t + \frac{1}{V_C}L\left(\phi_C^t\right) = -\left.\frac{\partial^3(\rho\phi)}{\partial t^3}\right|_t \frac{\Delta t^2}{6} + O\left(\Delta t^3\right) \tag{13.78}$$

confirming that the scheme is second order accurate. The third order derivative is a dispersive term that results in instability.

13.3.6 Second Order Upwind Euler (SOUE) Scheme

Using the second-order "upwind" interpolation profile depicted in Fig. 13.14, the interface $\rho\phi$ values are approximated as

$$\begin{aligned}(\rho_C\phi_C)^{t+\Delta t/2} &= \frac{3}{2}(\rho_C\phi_C)^t - \frac{1}{2}(\rho_C\phi_C)^{t-\Delta t} \\ (\rho_C\phi_C)^{t-\Delta t/2} &= \frac{3}{2}(\rho_C\phi_C)^{t-\Delta t} - \frac{1}{2}(\rho_C\phi_C)^{t-2\Delta t}\end{aligned} \tag{13.79}$$

Substituting in Eq. (13.57), the discretized $\rho\phi$ field equation is obtained as

$$\frac{3(\rho_C\phi_C)^t - 4(\rho_C\phi_C)^{t-\Delta t} + (\rho_C\phi_C)^{t-2\Delta t}}{2\Delta t}V_C + L\left(\phi_C^t\right) = 0 \tag{13.80}$$

which is the implicit second order upwind Euler (SOUE) scheme. In this scheme the values of $\rho\phi$ have to be stored for two of the older time steps, with its linearization coefficients given by

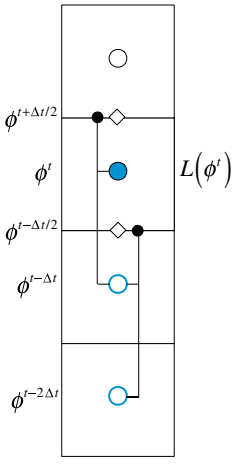

Fig. 13.14 Second order upwind Euler scheme

$$\begin{aligned} FluxC &= \frac{3\rho_C V_C}{2\Delta t} \\ FluxC^{\circ} &= -\frac{2\rho_C^{\circ} V_C}{\Delta t} \\ FluxV &= \frac{\rho_C^{\circ\circ} V_C \phi_C^{\circ\circ}}{2\Delta t} \end{aligned} \tag{13.81}$$

13.3.6.1 Numerical Accuracy

The scheme is second order as can be shown from a Taylor series evaluation. Expanding $(\rho\phi)^{t-\Delta t}$ and $(\rho\phi)^{t-2\Delta t}$ around time t give

$$(\rho\phi)^{t-\Delta t} = (\rho\phi)^{t} - \left.\frac{\partial(\rho\phi)}{\partial t}\right|_t \Delta t + \left.\frac{\partial^2(\rho\phi)}{\partial t^2}\right|_t \frac{\Delta t^2}{2} - \left.\frac{\partial^3(\rho\phi)}{\partial t^3}\right|_t \frac{\Delta t^3}{6} + O\left(\Delta t^4\right) \tag{13.82}$$

$$(\rho\phi)^{t-2\Delta t} = (\rho\phi)^{t} - \left.\frac{\partial(\rho\phi)}{\partial t}\right|_t 2\Delta t + \left.\frac{\partial^2(\rho\phi)}{\partial t^2}\right|_t 2\Delta t^2 - \left.\frac{\partial^3(\rho\phi)}{\partial t^3}\right|_t \frac{8\Delta t^3}{6} + O\left(\Delta t^4\right) \tag{13.83}$$

Multiplying Eq. (13.82) by 4 and subtracting the resulting equation from Eq. (13.83), an expression for the SOUE is obtained as

$$\frac{3(\rho\phi)^{t} - 4(\rho\phi)^{t-\Delta t} + (\rho\phi)^{t-2\Delta t}}{2\Delta t} = \left.\frac{\partial(\rho\phi)}{\partial t}\right|_t - \left.\frac{\partial^3(\rho\phi)}{\partial t^3}\right|_t \frac{\Delta t^2}{3} - O\left(\Delta t^3\right) \tag{13.84}$$

Combining Eq. (13.84) and Eq. (13.80), the recovered equation for $\rho\phi$ becomes

$$\left.\frac{\partial(\rho\phi)}{\partial t}\right|_t + \frac{1}{V_C} L\left(\phi_C^t\right) = \left.\frac{\partial^3(\rho\phi)}{\partial t^3}\right|_t \frac{\Delta t^2}{3} + O\left(\Delta t^3\right) \tag{13.85}$$

which has a third order numerical dispersion term but no numerical diffusion.

13.3.7 Initial Condition for the FV Approach

The implementation of the finite volume formulation is straight forward except for the initial time step. As shown in Fig. 13.15, the first temporal element is a boundary element in time, as such it does not have an upwind neighbor. Rather the value at the lower element face is used directly at the face resulting in a gradient that is half the correct numerical value. This comes about because it is computed as the difference between the values at $\phi_C^{t_{initial}+\Delta t/2}$ and $\phi_C^{t_{initial}}$, which are located half a

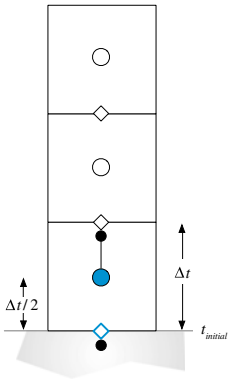

Fig. 13.15 Boundary temporal element

time step $(\Delta t/2)$ apart, while dividing their difference by a full time step (Δt) leading to a non-negligible initial error.

This is easily demonstrated by considering the first temporal element in the discretized equation of the first order implicit Euler scheme. Using Eq. (13.57) the discretized $\rho\phi$ field equation is obtained as

$$\frac{(\rho_C\phi_C)^{t_{initial}+\Delta t/2}-(\rho_C\phi_C)^{t_{initial}}}{\Delta t}V_C + L\left(\phi_C^{t_{initial}+\Delta t/2}\right) = 0 \tag{13.86}$$

For the first temporal element, the upwind interpolation yields a gradient computed as the difference between the $\rho\phi$ values at $t_{initial} + \Delta t/2$ and $t_{initial}$ divided by Δt. However for the case of a regular element (Fig. 13.16), the gradient is actually between the $\rho\phi$ values at $t_{initial} + 3\Delta t/2$ and $t_{initial} + \Delta t/2$, divided by Δt. The

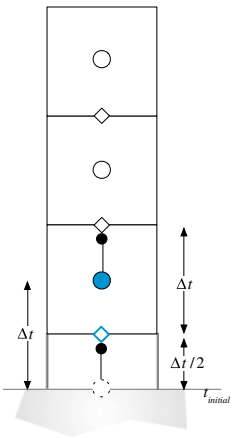

Fig. 13.16 Treatment of initial condition and the virtual element of centroid $t_{initial}$

difference between the two gradients is substantial, and any scheme that starts with the gradient of Eq. (13.86) will result in a large initial error that will affect the solution at the following steps. This error can be avoided if a grid similar to Fig. 13.16 is adopted. In this case the solution of the finite difference and finite volume methods will be basically similar, as for a regular grid.

Adopting this approach, the upwind values at the faces of the first temporal element spanning the time interval $[t_{initial} + \Delta t/2,\ t_{initial} + 3\Delta t/2]$ are obtained as

$$\begin{aligned}(\rho_C\phi_C)^{t_{initial}+3\Delta t/2} &= (\rho_C\phi_C)^{t_{initial}+\Delta t}\\ (\rho_C\phi_C)^{t_{initial}+\Delta t/2} &= (\rho_C\phi_C)^{t_{initial}}\end{aligned} \tag{13.87}$$

Substituting in Eq. (13.57), the discretized $\rho\phi$ field equation becomes

$$\frac{(\rho_C\phi_C)^{t_{initial}+\Delta t} - (\rho_C\phi_C)^{t_{initial}}}{\Delta t} V_C + L\left(\phi_C^{t_{initial}+\Delta t}\right) = 0 \tag{13.88}$$

which is similar to the equation obtained for any internal element.

Example 2

Repeat example 1 *using the CN scheme applied via Eq.* (13.73) *and the SOUE scheme. Use a time step* 0.05 *to find the values at* 0.1, 0.2, *and* 0.3.

Solution

The analytical solution at times 0.1, 0.2, and 0.3 were found in example 1.

Since two old values are needed, the value at the first time step is found using the first order backward Euler scheme. Thus the numerical solution is obtained using Eqs. (13.61), (13.73), and (13.80), which are reduced to

$$\begin{aligned}\phi_{EU}(t+\Delta t) &= \frac{1}{1+\Delta t}\phi^{\circ}\\ \phi_{CN}(t+\Delta t) &= \phi^{\circ\circ} - 2\Delta t\phi^{\circ}\\ \phi_{SOEU}(t+\Delta t) &= \frac{4\phi^{\circ} - \phi^{\circ\circ}}{3 + 2\Delta t}\end{aligned}$$

Based on the suggested implementation note, the first temporal element spans the time interval [0.025, 0.075], the second element spans the interval [0.075, 0.125], and so on.

Numerical solution using the second order CN scheme

Applying the above equations, the solutions are found as

$$\left.\begin{aligned}
&\phi_{EU}(0.05) = \frac{1}{1+0.05}\phi(0) = 0.95238 \\
&\phi_{CN}(0.1) = \phi^{\circ\circ} - 2\Delta t\phi^{\circ} = 1 - 2 \times 0.05 \times 0.95238 = \boxed{0.90476} \\
&\phi_{CN}(0.15) = 0.95238 - 2 \times 0.05 \times 0.90476 = 0.861904 \\
&\phi_{CN}(0.2) = 0.90476 - 2 \times 0.05 \times 0.861904 = \boxed{0.81857} \\
&\phi_{CN}(0.25) = 0.861904 - 2 \times 0.05 \times 0.81857 = 0.780047 \\
&\phi_{CN}(0.3) = 0.81857 - 2 \times 0.05 \times 0.780047 = \boxed{0.74056}
\end{aligned}\right\}$$

$$\Rightarrow \begin{cases} error_{CN}(0.1) = 4 \times 10^{-5} \\ error_{CN}(0.2) = 1.3 \times 10^{-4} \\ error_{CN}(0.3) = 2.4 \times 10^{-4} \end{cases}$$

It is clear that the solution error indicates second order accuracy. The slight differences between the error values obtained here and those reported in example 1 are due to the number of decimal values carried during computations.

Numerical solution using the SOUE scheme

$$\left.\begin{aligned}
&\phi_{EU}(0.05) = \frac{1}{1+0.05}\phi(0) = 0.9524 \\
&\phi_{SOUE}(0.1) = \frac{4\phi^{\circ} - \phi^{\circ\circ}}{3+2\Delta t} = \frac{4 \times 0.9524 - 1}{3.1} = \boxed{0.90632} \\
&\phi_{SOUE}(0.15) = (4 \times 0.90632 - 0.9524)/3.1 = 0.86219 \\
&\phi_{SOUE}(0.2) = (4 \times 0.86219 - 0.90632)/3.1 = \boxed{0.82014} \\
&\phi_{SOUE}(0.25) = (4 \times 0.82014 - 0.86219)/3.1 = 0.780119 \\
&\phi_{SOUE}(0.3) = (4 \times 0.780119 - 0.82014)/3.1 = \boxed{0.74204}
\end{aligned}\right\}$$

$$\Rightarrow \begin{cases} error_{SOUE}(0.1) = 1.5 \times 10^{-3} \\ error_{SOUE}(0.2) = 1.44 \times 10^{-3} \\ error_{SOUE}(0.3) = 1.24 \times 10^{-3} \end{cases}$$

The solution is second order accurate, however it is less accurate than the CN solution.

13.4 Non-Uniform Time Steps

So far a uniform time step was considered. In practical applications it is common to use variable time steps mainly to reduce the computational cost by selecting, at every time step, the maximum allowable time step value that does not violate the CFL condition.

For first order schemes, the discretization is not affected by whether the time step is variable or constant. The situation is different for second order transient schemes since they use a stencil involving two time step values. For the case of the two step implementation of the Crank-Nicolson transient scheme nothing changes except that for each of the two steps a different time step is used. This affects the accuracy as the spatial derivative is no longer at the center of the temporal element. For other second order schemes, the interpolation profile has to be modified to account for the non equal time steps. In what follows a non uniform transient grid is used in the discretization of the transient term for the standard CN [2] and the SOUE [4, 19, 20] schemes. While the finite volume and finite difference methods yield equivalent algebraic relations in a uniform grid, this is not the case for variable time steps as demonstrated in the derivations to follow.

13.4.1 Non-Uniform Time Steps with the Finite Difference Approach

13.4.1.1 Crank-Nicolson Scheme

The CN scheme with non uniform time steps is derived, as shown in Fig. 13.17, by expressing the values of $\rho\phi$ at times $t + \Delta t$ and $t - \Delta t^{\circ}$ in terms of its value and the values of its derivatives at time t using Taylor series as

$$(\rho\phi)^{t+\Delta t} = (\rho\phi)^{t} + \left.\frac{\partial(\rho\phi)}{\partial t}\right|_{t} \Delta t + \left.\frac{\partial^2(\rho\phi)}{\partial t^2}\right|_{t} \frac{\Delta t^2}{2!} + \left.\frac{\partial^3(\rho\phi)}{\partial t^3}\right|_{t} \frac{\Delta t^3}{3!} + \cdots \tag{13.89}$$

$$(\rho\phi)^{t-\Delta t^{\circ}} = (\rho\phi)^{t} - \left.\frac{\partial(\rho\phi)}{\partial t}\right|_{t} \Delta t^{\circ} + \left.\frac{\partial^2(\rho\phi)}{\partial t^2}\right|_{t} \frac{(\Delta t^{\circ})^2}{2!} - \left.\frac{\partial^3(\rho\phi)}{\partial t^3}\right|_{t} \frac{(\Delta t^{\circ})^3}{3!} + \cdots \tag{13.90}$$

Then, multiplying Eq. (13.89) by $(\Delta t^{\circ})^2$ and Eq. (13.90) by Δt^2 and subtracting the resulting equations from each other, an equation for the first derivative is obtained as

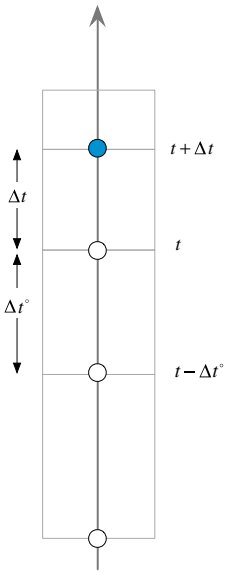

Fig. 13.17 The finite difference temporal mesh of the CN scheme with non uniform time steps

$$\left.\frac{\partial(\rho\phi)}{\partial t}\right|_t \approx \frac{(\Delta t^{\circ})^2(\rho\phi)^{t+\Delta t} - \left[(\Delta t^{\circ})^2 - \Delta t^2\right](\rho\phi)^{t} - \Delta t^2(\rho\phi)^{t-\Delta t}}{\left[\Delta t(\Delta t^{\circ})^2 + \Delta t^{\circ}\Delta t^2\right]} \tag{13.91}$$

Substituting the expression for the gradient from Eq. (13.91) in Eq. (13.3), the discretized equation for the CN scheme with non uniform time steps is given by

$$\frac{(\Delta t^{\circ})^2(\rho\phi) - \left[(\Delta t^{\circ})^2 - \Delta t^2\right](\rho\phi)^{\circ} - \Delta t^2(\rho\phi)^{\circ\circ}}{\Delta t^{\circ}\,\Delta t(\Delta t + \Delta t^{\circ})} V_C + L\left(\phi_C^{\circ}\right) = 0 \tag{13.92}$$

Expanding the spatial term, the final form of the algebraic equation becomes

$$\left(a_C^{\bullet} + a_C\right)\phi_C + \sum_{F\sim NB(C)} a_F\phi_F = b_C - a_C^{\circ}\phi_C^{\circ} - a_C^{\circ\circ}\phi_C^{\circ\circ} \tag{13.93}$$

with the time dependent coefficients computed from

$$\begin{aligned} a_C^{\bullet} &= \frac{\Delta t^{\circ}}{\Delta t(\Delta t + \Delta t^{\circ})}\rho_C V_C \\ a_C^{\circ} &= \frac{\Delta t - \Delta t^{\circ}}{\Delta t + \Delta t^{\circ}}\rho_C^{\circ} V_C \\ a_C^{\circ\circ} &= \frac{\Delta t}{\Delta t^{\circ}(\Delta t + \Delta t^{\circ})}\rho_C^{\circ\circ} V_C \end{aligned} \tag{13.94}$$

For uniform time steps, the coefficients in Eq. (13.39) are recovered.

13.4.2 Adams-Moulton (or SOUE) Scheme

Referring to Fig. 13.18, the Adams-Moulton scheme, also denoted by the SOUE scheme, with non uniform time steps is derived by expressing the values of the dependent variable ϕ at times $t - \Delta t$ and $t - \Delta t - \Delta t^\circ$ in terms of its value and the values of its derivatives at time t using Taylor series as

$$(\rho\phi)^{t-\Delta t} = (\rho\phi)^t - \Delta t \left.\frac{\partial(\rho\phi)}{\partial t}\right|_t + \frac{\Delta t^2}{2}\left.\frac{\partial^2(\rho\phi)}{\partial t^2}\right|_t + O\left(\Delta t^3\right) \tag{13.95}$$

$$(\rho\phi)^{t-\Delta t-\Delta t^\circ} = (\rho\phi)^t - (\Delta t + \Delta t^\circ)\left.\frac{\partial(\rho\phi)}{\partial t}\right|_t + \frac{(\Delta t + \Delta t^\circ)^2}{2}\left.\frac{\partial^2(\rho\phi)}{\partial t^2}\right|_t + O\left(\Delta t^3\right) \tag{13.96}$$

Multiplying Eq. (13.95) by $(\Delta t + \Delta t^\circ)^2/\Delta t^2$ and subtracting the resulting equation from Eq. (13.96), a second order representation of the first derivative (i.e., the SOUE scheme) is obtained as

$$\left.\frac{\partial(\rho\phi)}{\partial t}\right|_t = \frac{1}{\Delta t}\left[\left(1 + \frac{\Delta t}{\Delta t + \Delta t^\circ}\right)(\rho\phi)^t - \left(1 + \frac{\Delta t}{\Delta t^\circ}\right)(\rho\phi)^{t-\Delta t} + \left(\frac{\Delta t^2}{\Delta t^\circ(\Delta t + \Delta t^\circ)}\right)(\rho\phi)^{t-\Delta t-\Delta t^\circ}\right] \tag{13.97}$$

Substituting the expression for the gradient from Eq. (13.97) in Eq. (13.3), the discretized equation becomes

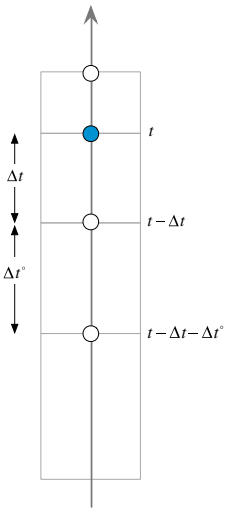

Fig. 13.18 The finite difference temporal mesh of the SOUE scheme with non uniform time steps

$$V_C\left(\frac{1}{\Delta t}+\frac{1}{\Delta t+\Delta t^{\circ}}\right)(\rho_C\phi_C)-V_C\left(\frac{1}{\Delta t}+\frac{1}{\Delta t^{\circ}}\right)(\rho_C\phi_C)^{\circ}+V_C\left(\frac{\Delta t}{\Delta t^{\circ}(\Delta t+\Delta t^{\circ})}\right)(\rho_C\phi_C)^{\circ\circ}+L\left(\phi_C^t\right)=0 \tag{13.98}$$

Expanding the spatial term, the final form of the algebraic equation is written as

$$\left(a_C^{\bullet}+a_C\right)\phi_C+\sum_{F\sim NB(C)}a_F\phi_F=b_C-a_C^{\circ}\phi_C^{\circ}-a_C^{\circ\circ}\phi_C^{\circ\circ} \tag{13.99}$$

with the time dependent coefficients obtained from

$$\begin{aligned}a_C^{\bullet}&=\left(\frac{1}{\Delta t}+\frac{1}{\Delta t+\Delta t^{\circ}}\right)\rho_C V_C\\ a_C^{\circ}&=-\left(\frac{1}{\Delta t}+\frac{1}{\Delta t^{\circ}}\right)\rho_C^{\circ}V_C\\ a_C^{\circ\circ}&=\frac{\Delta t}{\Delta t^{\circ}(\Delta t+\Delta t^{\circ})}\rho_C^{\circ\circ}V_C\end{aligned} \tag{13.100}$$

For uniform time steps the coefficients given in Eq. (13.54) are recovered.

13.4.3 Non-Uniform Time Steps with the Finite Volume Approach

Following the terminology used with the FVM, the size of a temporal element is denoted by Δt, while the distance between the centroids of two consecutive temporal elements is designated by δt. For uniform time steps both are equal and the time between two consecutive computed fields is $\Delta t=\delta t$ for both the finite difference and finite volume methods. For non-uniform time steps the time remains Δt for the finite difference method, however it becomes $\delta t=(\Delta t+\Delta t^{\circ})/2$ for the finite volume method leading to different formulations.

As for the finite difference method, with non-uniform time steps, the current and old time step values affect the scheme interpolation profile and hence its finite volume discretization. This is similar to writing the profile for a convection scheme over a structured non-uniform grid. The procedure used will be illustrated by considering the CN and SOUE schemes. Extension to other profiles is straightforward.

13.4.4 Crank-Nicolson Scheme

The CN scheme is obtained by calculating the value of $\rho\phi$ at an interface as the average of the $\rho\phi$ values at the main points straddling the interface (Fig. 13.19), i.e.,

$$\begin{aligned}(\rho_C\phi_C)^{t-\Delta t/2} &= \frac{\Delta t^\circ}{\Delta t+\Delta t^\circ}(\rho_C\phi_C)^t + \frac{\Delta t}{\Delta t+\Delta t^\circ}(\rho_C\phi_C)^{t-(\Delta t^\circ+\Delta t)/2}\\ (\rho_C\phi_C)^{t-\Delta t/2-\Delta t^\circ} &= \frac{\Delta t^{\circ\circ}}{\Delta t^\circ+\Delta t^{\circ\circ}}(\rho_C\phi_C)^{t-(\Delta t^\circ+\Delta t)/2} + \frac{\Delta t^\circ}{\Delta t^\circ+\Delta t^{\circ\circ}}(\rho_C\phi_C)^{t-\Delta t^\circ-(\Delta t+\Delta t^{\circ\circ})/2}\end{aligned} \tag{13.101}$$

Substituting in Eq. (13.57), the discretized $\rho\phi$ field equation is obtained as

$$\begin{aligned}&\frac{\Delta t^\circ}{\Delta t+\Delta t^\circ}\frac{V_C}{\Delta t}(\rho_C\phi_C) + \left(\frac{\Delta t}{\Delta t+\Delta t^\circ} - \frac{\Delta t^{\circ\circ}}{\Delta t^\circ+\Delta t^{\circ\circ}}\right)\frac{V_C}{\Delta t}(\rho_C\phi_C)^\circ\\ &\quad - \frac{\Delta t^\circ}{\Delta t^\circ+\Delta t^{\circ\circ}}\frac{V_C}{\Delta t}(\rho_C\phi_C)^{\circ\circ} + L\left(\phi_C^\circ\right) = 0\end{aligned} \tag{13.102}$$

The linearization coefficients for the CN scheme with non uniform time steps are inferred to be

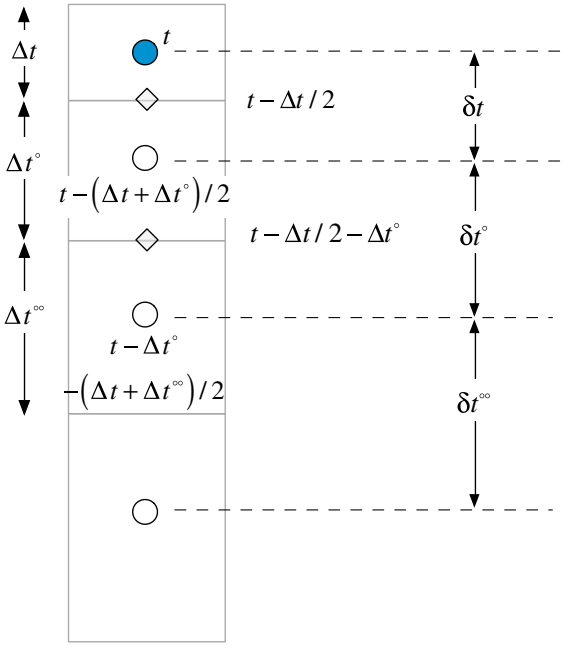

Fig. 13.19 The finite volume temporal mesh of the CN scheme with non uniform time steps

$$
\begin{aligned}
FluxC &= \frac{\Delta t^{\circ}}{\Delta t + \Delta t^{\circ}} \frac{\rho_C V_C}{\Delta t} \\
FluxC^{\circ} &= \left(\frac{\Delta t}{\Delta t + \Delta t^{\circ}} - \frac{\Delta t^{\circ\circ}}{\Delta t^{\circ} + \Delta t^{\circ\circ}} \right) \frac{\rho_C^{\circ} V_C}{\Delta t} \\
FluxV &= -\frac{\Delta t^{\circ\circ}}{\Delta t^{\circ} + \Delta t^{\circ\circ}} \frac{\rho_C^{\circ\circ} V_C \phi_C^{\circ\circ}}{\Delta t}
\end{aligned}
\tag{13.103}
$$

As in the constant time step case, the method is explicit necessitating storing values of the two previous time steps. Moreover the uniform time steps formulation can be recovered by setting $\Delta t = \Delta t^{\circ} = \Delta t^{\circ\circ}$ in Eqs. (13.102) and (13.103).

13.4.5 Adams-Moulton (or SOUE) Scheme

With the second-order "upwind" interpolation profile given by Eq. (11.84), the interface $\rho\phi$ values at the faces $t + \Delta t/2$ and $t - \Delta t/2$ displayed in Fig. 13.20 are found to be

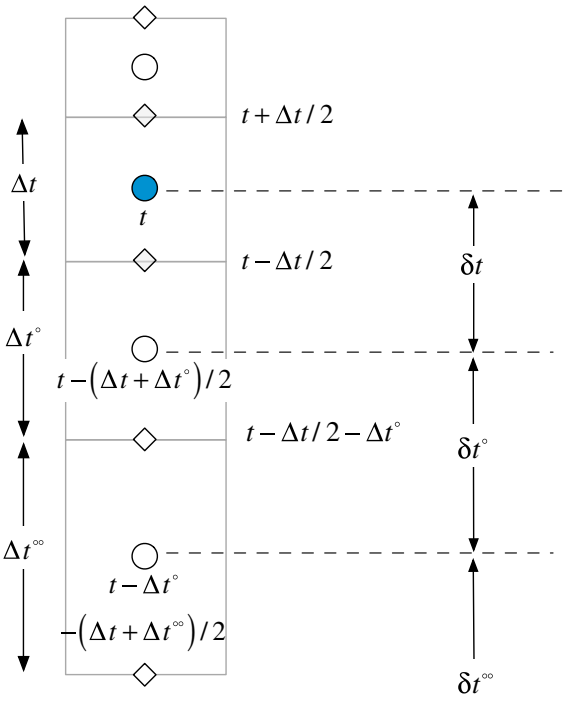

Fig. 13.20 The finite volume temporal mesh of the SOUE scheme with non uniform time steps

$$(\rho\phi)^{t+\Delta t/2} = (\rho\phi)^{t} + \left[(\rho\phi)^{t} - (\rho\phi)^{t-(\Delta t+\Delta t^{\circ})/2}\right]\frac{\Delta t}{\Delta t + \Delta t^{\circ}}$$
$$(\rho\phi)^{t-\Delta t/2} = (\rho\phi)^{t-(\Delta t+\Delta t^{\circ})/2} + \left[(\rho\phi)^{t-(\Delta t+\Delta t^{\circ})/2} - (\rho\phi)^{t-\Delta t^{\circ}-(\Delta t+\Delta t^{\circ\circ})/2}\right]\frac{\Delta t^{\circ}}{\Delta t^{\circ} + \Delta t^{\circ\circ}} \tag{13.104}$$

Using this profile approximation, the discretized form of Eq. (13.1) over the element C shown in Fig. 13.2 is obtained by substituting Eq. (13.104) in Eq. (13.57) and is given by

$$\left(1 + \frac{\Delta t}{\Delta t + \Delta t^{\circ}}\right)\frac{V_C}{\Delta t}(\rho_C\phi_C) - \left(1 + \frac{\Delta t}{\Delta t + \Delta t^{\circ}} + \frac{\Delta t^{\circ}}{\Delta t^{\circ} + \Delta t^{\circ\circ}}\right)\frac{V_C}{\Delta t}(\rho_C\phi_C)^{\circ}$$
$$+ \frac{\Delta t^{\circ}}{\Delta t^{\circ} + \Delta t^{\circ\circ}}\frac{V_C}{\Delta t}(\rho_C\phi_C)^{\circ\circ} + L(\phi_C) = 0 \tag{13.105}$$

The linearization coefficients for the SOUE scheme with non uniform time steps are inferred to be

$$\begin{aligned} FluxC &= \left(\frac{1}{\Delta t} + \frac{1}{\Delta t + \Delta t^{\circ}}\right)\rho_C V_C \\ FluxC^{\circ} &= -\left(\frac{1}{\Delta t} + \frac{1}{\Delta t + \Delta t^{\circ}} + \frac{\Delta t^{\circ}/\Delta t}{\Delta t^{\circ} + \Delta t^{\circ\circ}}\right)\rho_C^{\circ} V_C \\ FluxV &= \left(\frac{\Delta t^{\circ}/\Delta t}{\Delta t^{\circ} + \Delta t^{\circ\circ}}\right)\rho_C^{\circ\circ} V_C \phi_C^{\circ\circ} \end{aligned} \tag{13.106}$$

Similar to the constant time step case, the method is implicit as it requires solving a system of equations to obtain the ϕ field at every time step. The uniform time step form of the equation given by Eq. (13.80) is obtained by setting $\Delta t = \Delta t^{\circ} = \Delta t^{\circ\circ}$ in Eq. (13.105).

13.5 Computational Pointers

13.5.1 uFVM

The discretization of the transient term in uFVM follows the finite volume method implemented within an implicit framework. The assembly of the transient fluxes resulting from the first order backward Euler scheme is shown in Listing 13.1.

```
%
theDensityField = cfdGetMeshField(['Density' theFluidTag]);
density = theDensityField.phi(iElements);
density_old = theDensityField.phi_old(iElements);

volumes = [theMesh.elements(iElements).volume]';

theFluxes.FLUXCE(iElements)    =   volumes .* density / dt;
theFluxes.FLUXCEOLD(iElements) = - volumes .* density_old / dt;
theFluxes.FLUXTE(iElements)    = theFluxes.FLUXCE .* phi;
theFluxes.FLUXTEOLD(iElements) = theFluxes.FLUXCEOLD .* phi_old ;
```

Listing 13.1 Assembly of the transient fluxes resulting from the implicit Euler scheme

13.5.2 OpenFOAM®

In OpenFOAM®, explicit and implicit time derivatives [21] are defined via the namespaces fvm, fvc and the corresponding functions **fvc::ddt(rho, phi)** and **fvm::ddt(rho, phi)**, respectively. Moreover the first and second order upwind Euler schemes in addition to the second order Crank-Nicholson scheme are available, with the latter implemented following the two-step approach.

The files of the transient schemes are located in the directory "$FOAM_SRC/finiteVolume/finiteVolume/ddtSchemes". A base class denoted by **ddtScheme <Type>** is defined from which all time discretization schemes have to be derived.

The first order Euler scheme is implemented in the class **EulerDdtScheme.** The class is declared on top of the base class **ddtScheme <Type>**, as shown in Listing 13.2.

```
template<class Type>
class EulerDdtScheme
:
    public ddtScheme<Type>
```

Listing 13.2 Decleration of the EulerDdtScheme class

The implementation of the associated **fvc** and **fvm** namespaces are defined in the file **EulerDdtScheme.C.** The implicit evaluation of the Euler scheme is defined via the following function in Listing 13.3:

```
template<class Type>
tmp<fvMatrix<Type> >
EulerDdtScheme<Type>::fvmDdt
(
    const volScalarField& rho,
    const GeometricField<Type, fvPatchField, volMesh>& vf
)
```

Listing 13.3 Definition of the function needed for implicit solution using the Euler scheme

As depicted in Listing 13.4, the first step in this function is the definition of the **fvMatrix** in which only the diagonal coefficient vector is filled.

```
{
    tmp<fvMatrix<Type> > tfvm
    (
        new fvMatrix<Type>
        (
            vf,
            vf.dimensions()*dimVol/dimTime
        )
    );
    fvMatrix<Type>& fvm = tfvm();
```

Listing 13.4 Definition of the fvMatrix

After allocating the needed space for storing the diagonal coefficients and sources, the values to be stored are defined and computed. As shown in Listing 13.5, this is accomplished by first defining the reciprocal of the time step as **rDeltaT**, then calculating $a_t = \rho_C V_C / \Delta t$ and storing its value in the **fvm.diag()** vector. The source contribution is computed as the product of $a_t^o = -\rho_C^o V_C / \Delta t$ and the old value of **vf** and stored in the **fvm.source()** vector, where **vf** is the generic variable used while applying the time scheme.

```
  scalar rDeltaT = 1.0/mesh().time().deltaTValue();
  fvm.diag() = rDeltaT*rho*mesh().V();
  fvm.source()=rDeltaT*rho.oldTime()*vf.oldTime().internalField()*mesh
().V();
  return tfvm;
}
```

Listing 13.5 Calculation of the terms added to the diagonal and source vectors

The script in Listing 13.6 shows that OpenFOAM® allows explicit evaluation of the unsteady term using the Euler scheme. In this case, a **GeometricField** object containing the value of $(\rho\phi - \rho^o\phi^o)/\Delta t$ is returned (here the value is per unit volume).

```
tmp<GeometricField<Type, fvPatchField, volMesh> >
EulerDdtScheme<Type>::fvcDdt
(
    const GeometricField<Type, fvPatchField, volMesh>& vf
)
        return tmp<GeometricField<Type, fvPatchField, volMesh> >
        (
            new GeometricField<Type, fvPatchField, volMesh>
            (
                ddtIOobject,
                rDeltaT*(vf*rho - vf.oldTime()*rho.oldTime())
            )
        );
```

Listing 13.6 Explicit calculation of the unsteady term using the Euler scheme

The SOUE scheme is implemented in OpenFOAM® under the class **backwardDdtScheme.** The definition of the class is on top of the base class, as shown in Listing 13.7.

```
template<class Type>
class backwardDdtScheme
:
    public fv::ddtScheme<Type>
{
    // Private Member Functions

        //- Return the current time-step
        scalar deltaT_() const;

        //- Return the previous time-step
        scalar deltaT0_() const;
```

Listing 13.7 Script used to define the **backwardDdtScheme** class for the implementation of the SOUE scheme

In this case OpenFOAM® uses information from the current and the previous time steps. In the general case, the time steps are different necessitating the use of two variables to store their values.

The implicit time discretization is defined in a way similar to the first order transient scheme using the function shown in Listing 13.8 through Listing 13.10.

```
template<class Type>
tmp<fvMatrix<Type> >
backwardDdtScheme<Type>::fvmDdt
(
    const volScalarField& rho,
    const GeometricField<Type, fvPatchField, volMesh>& vf
)
```

Listing 13.8 Script used to define the implicit discretization using the SOUE scheme

The part of the function displayed in Listing 13.9 calculates the transient coefficients that multiply the current, old, and old-old values of the dependent variable and stores the contribution to the diagonal coefficients in **fvm.diag()** vector.

```
    scalar rDeltaT = 1.0/deltaT_();

    scalar deltaT = deltaT_();
    scalar deltaT0 = deltaT0_(vf);

    scalar coefft   = 1 + deltaT/(deltaT + deltaT0);
    scalar coefft00 = deltaT*deltaT/(deltaT0*(deltaT + deltaT0));
    scalar coefft0  = coefft + coefft00;

    fvm.diag() = (coefft*rDeltaT)*rho.internalField()*mesh().V();
```

Listing 13.9 Script used to calculate the unsteady coefficients and to store the contribution to the diagonal coefficients in the **fvm.diag()** vector

In the last part of the function shown in Listing 13.10, the contribution of the unsteady term is computed and stored in the fvm.source() vector.

```
fvm.source() = rDeltaT*mesh().V()*
(
    coefft0*rho.oldTime().internalField()
   *vf.oldTime().internalField()
  - coefft00*rho.oldTime().oldTime().internalField()
   *vf.oldTime().oldTime().internalField()
);
```

Listing 13.10 Script used to calculate and store contribution to the source in the **fvm.source()** vector

By comparing the coefficients of the SOUE scheme with the ones given in Eq. (13.100) it is easily seen that OpenFOAM® adopts a finite difference approach for the discretization of the unsteady term whereby the time derivative is approximated via Taylor series expansions.

13.6 Closure

The chapter covered the discretization of the transient term in the unsteady conservation equation. For that purpose, two general methodologies were discussed. One method is based on a finite difference discretization while the other follows a finite volume approach in which the conservation equation is integrated over a temporal element. The first order fully implicit and fully explicit transient schemes were presented. The formulation of higher order approximations was also investigated. This included the CN and the SOUE schemes for uniform and non uniform time steps. The next chapter is devoted to the discretization of the source term, relaxation of the algebraic system of equations, and other related details.

13.7 Exercises

Exercise 1
Transient heat transfer for the one dimensional body shown in Fig. 13.21 is governed by the following energy equation:

$$\frac{\partial\left(\rho c_p T\right)}{\partial t} = \frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right)$$

The body is insulated at one end while subjected to convective heat transfer at the second end. Other parameters include $T_R = 330$ K, $h_R = 400$ W/m^{2}K, $k = 55$ W/mK, $\rho = 7000\,\text{kg/m}^3$, and $c_p = 400$ J/Kg K.

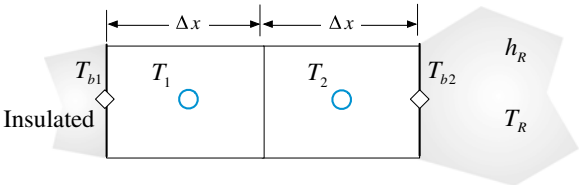

Fig. 13.21 One-dimensional domain used for Exercise 1

(a) Compute the temperature field using the Euler Explicit method for three time steps. Note that the initial temperature is $T_i = 273$ K with $\Delta t = 20$ s and $\Delta x = 0.015$ m.
(b) Repeat part a using an implicit Euler scheme.
(c) Explain the difference in temperatures between the two methods at time $t = 60$ s.

Exercise 2

The body described in Exercise 1 is now insulated at one end while subjected to a Dirichlet boundary condition at the second end. The initial and boundary conditions are $T_i = 273$ K, $T_{b2} = 330$ K while values of other parameters are given by

$$\Delta x = 0.015 \text{ m},\ k = 55 \text{ W/mK},\ \rho = 7000 \text{ kg/m}^3 \text{ and } c_p = 400 \text{ J/Kg K}.$$

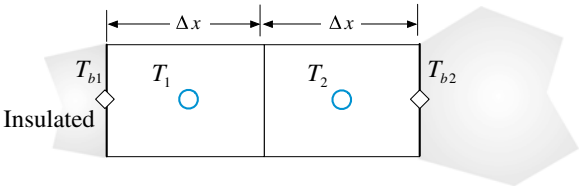

Fig. 13.22 One-dimensional domain used for Exercise 2

Compute the temperature field for three time steps using:

(a) The Adams-Moulton method with uniform time steps ($\Delta t = 20$ s).
(b) The finite difference form of the Adams-Moulton method with non-uniform time steps ($\Delta t_1 = 10$ s, $\Delta t_2 = 20$ s, and $\Delta t_3 = 30$ s).
(c) The finite volume form of the Adams-Moulton method with non-uniform time steps ($\Delta t_1 = 10$ s, $\Delta t_2 = 20$ s, and $\Delta t_3 = 30$ s). (Fig. 13.22).

Use the implicit Euler scheme for the first time step.

Exercise 3

The body described in Exercise 1 is again subjected to a Dirichlet boundary condition at one end and to a convective heat transfer at the second end. The parameters involved are $\Delta x = 0.015$ m, $T_i = 273$ K, $T_{b1} = 260$ K, $T_R = 330$ K, $h_R = 400$ W/m^{2}K, $k = 55$ W/mK, $\rho = 7000$ kg/m^3, and $c_p = 400$ J/Kg K (Fig. 13.23).

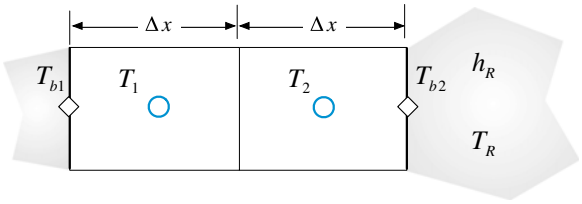

Fig. 13.23 One-dimensional domain used for Exercise 3

Compute the temperature field for three time steps using:

(a) The Crank-Nicolson method with uniform time steps ($\Delta t = 20$ s).
(b) The finite difference form of the Crank-Nicolson method with non-uniform time steps ($\Delta t_1 = 10$ s, $\Delta t_2 = 20$ s, and $\Delta t_3 = 30$ s).
(c) The finite volume form of the Crank-Nicolson method with non-uniform time steps ($\Delta t_1 = 10$ s, $\Delta t_2 = 20$ s, and $\Delta t_3 = 30$ s).

Use the implicit Euler scheme for the first time step.

Exercise 4

Consider the following equation defined over the one dimensional grid shown in Fig. 13.24:

$$\frac{\partial \phi}{\partial t} = \nabla \cdot \Gamma \nabla \phi - \beta \phi$$

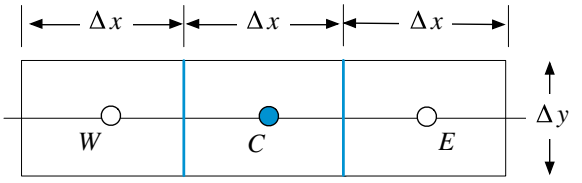

Fig. 13.24 One dimensional domain used for Exercise 4

(a) Derive the algebraic equation for element C. Use a first order Euler Explicit scheme for the transient term and linearize the source term given that β is positive.
(b) Is there a step limitation for the equation derived in (a)? If so derive its expression in terms of the appropriate variables.

Exercise 5
Use the implicit backward Euler scheme to integrate in time the linear advection equation given by

$$\frac{\partial \phi}{\partial t} + u\frac{\partial \phi}{\partial x} = 0\, u > 0$$

and the second order central difference approximation for the spatial derivative.

(a) Derive the discretized equation.
(b) Find the accuracy of the scheme
(c) Determine the stability of the scheme.

Exercise 6
Use the fully implicit Euler scheme in time and the central difference scheme in space to discretize the one dimensional convection diffusion heat equation given by

$$\frac{\partial \left(\rho c_p T\right)}{\partial t} + \frac{\partial (\rho u T)}{\partial x} = \frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right) + S$$

over a uniform mesh of spacing Δx and write it down in the standard form.

There are two issues that should be considered when choosing the time step: stability and accuracy. What are the limits for the time step of the two schemes to achieve stable and accurate solutions? Are these limits similar for both stability and accuracy?

Exercise 7 (OpenFOAM®)
List from Doxygen [22] all derived classes of the *ddtScheme <Type>* class.

Exercise 8 (OpenFOAM®)
Find in OpenFOAM® the fvm implementation of the first order implicit Euler scheme. Compare the implemented algorithm with Eq. (13.28) and the contribution to the matrix of coefficients with Eq. (13.62).

Exercise 9 (OpenFOAM®)
Compare in OpenFOAM® the fvm implementation of the second order Crank-Nicolson transient scheme with Eqs. (13.43) and (13.44). The C file is located in "*$FOAM_SRC/finiteVolume/finiteVolume/ddtSchemes/CrankNicolsonDdtScheme /CrankNicolsonDdtScheme.C*". Hint: In the fvm member function, just check the *if* statement when *mesh().moving()* is false.

References

1. Faires JD, Burden RL (1993) Numerical methods. PWS, Boston, pp 152–153
2. Crank J, Nicolson P (1947) A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type. Proc Cambr Phil Soc 43:50–67
3. Shyy W (1985) A study of finite difference approximations to steady state convection dominated flows. J Comput Phys 57:415–438
4. Moukalled F, Darwish M (2012) Transient schemes for capturing interfaces of free-surface flows. Numer Heat Transf Part B Fundam 61(3):171–203
5. Ascher U, Ruuth S, Spiteri RJ (1997) Implicit-explicit runge-kutta methods for time-dependent partial differential equations. Appl Numer Math 25:151–167
6. Ames WF (1977) Numerical methods for partial differential equations. Academic Press, Orlando
7. Milne WE (1953) Numerical solution of differential equations. Wiley, New York
8. Richtmyer RD (1967) Difference methods for initial value problems, 2nd edn. Wiley, New York
9. Birkhoff G, Rota G (1989) Ordinary differential equations. Wiley, New York
10. Burden R, Faires JD (2010) Numerical analysis, 9th edn. Brooks, Cole
11. Chapra S, Canale R (2014) Numerical methods for engineers. 7th ed., McGraw Hill, New York
12. Cheney W, Kincaid D (2013) Numerical mathematics and computing, 7th edn. Brooks/Cole, Boston
13. Courant R, Friedrichs K, Lewy H (1928) Über die partiellen Differenzengleichungen der mathematischen Physik. Math Ann (in German) 100:32–74
14. Patankar SV (1980) Numerical heat transfer and fluid flow, Hemisphere, New York
15. Peinado J, Ibáñez J, E. Arias E, V. Hernández V (2010) Adams–Bashforth and Adams–Moulton methods for solving differential Riccati equations. Comput Math With Appl 60 (11):3032–3045
16. Ferziger JH, Peric M (2013) Computational methods for fluid dynamics, 3rd edn. Springer, Germany
17. Courant R, Isaacson E, Rees M (1952) On the solution of nonlinear hyperbolic differential equations by finite differences. Commun Pure Appl Math 5:243–255
18. Darwish M, Moukalled F (2006) Convective schemes for capturing interfaces of free-surface flows on unstructured grids. Numer Heat Transf Part B Fundam 49(1):19–42
19. Darwish M, Moukalled F (1994) Normalized variable and space formulation methodology for high-resolution schemes. Numer Heat Transf Part B Fundam 26(1):79–96
20. Leonard BP (1981) A survey of finite differences with unwinding for numerical modeling of the incompressible convection diffusion equation. In Taylor C, Morgan K (eds.) Computational techniques in transient and turbulent flow, Pineridge Press, Swansea, UK, 2:1–35
21. OpenFOAM, 2015 Version 2.3.x. http://www.openfoam.org
22. OpenFOAM Doxygen, 2015 Version 2.3.x. http://www.openfoam.org/docs/cpp/