An investigation of spatial and temporal weighting schemes for use in unstructured mesh control volume finite element methods

S. L. Truscott[*]   I. W. Turner[†]

(Received 31 August 2001; revised 17 October 2002)

Abstract

In this work the control volume finite element (CVFE) method is used to discretise a two-dimensional non-linear transport model on an unstructured mesh. First and second order temporal weighting, combined with various flux limiting techniques (spatial weighting) are analysed in order to identify the most accurate and efficient numerical scheme. An inexact Newton method is used to resolve the underlying discrete non-linear system. In computing the Newton

---

[*]School of Mathematics, Queensland University of Technology, Brisbane, AUSTRALIA. mailto:s.truscott@fsc.qut.edu.au

[†]School of Mathematics, Queensland University of Technology, Brisbane, AUSTRALIA. mailto:i.turner@fsc.qut.edu.au

step the performance of the preconditioned iterative solvers BiCGSTAB and GMRES, in conjunction with a two-node Jacobian approximation is also examined. A linear benchmark problem that admits an analytical solution is used to assess the accuracy and computational efficiency of the numerical model. The results show that the flux limited second order temporal scheme substantially reduce numerical diffusion on relatively coarse meshes. The low temperature wood drying non-linear case study highlights that the first order temporal scheme combined with flux limiting achieves good accuracy on a relatively coarse mesh and improves the overall computational efficiency.

Contents

1 Introduction

The motivation for this work arises from the need to develop an efficient and accurate computational model for wood drying. Two-dimensional models that account for the heterogeneous nature of the wood, such as growth rings, must be developed for the purposes of furthering the fundamental understanding of the drying process. Due to the inherent complexity of the model, unstructured triangular meshes are used to describe the computational domain. Additionally the governing equations for the drying process are highly non-linear and hence require the efficient implementation of a non-linear solver. In the past, the control volume finite element (CVFE) method [3, 7] has been used to discretise the non-linear partial differential system that describes the drying process. This method will be analysed in detail here in order to assess its accuracy and efficiency. The optimal combination of spatial and temporal weighting schemes to ensure accurate results on relatively coarse unstructured meshes will also be identified. The first of two important ingredients in this computational model is flux limiting, which is a scheme that sharpens the steep drying fronts observed in the saturation profiles. The second is an inexact Newton method, which resolves the non-linear system at each time level in order to advance the solution of the transport equation in time. Flux limiting on structured grids has been utilised with great success in the past for drying [11] and for reducing numerical diffusion of the contaminant mole fraction fronts in multi-phase compositional computational models [4, 12]. It is well known that a great deal of care must be taken with the treatment of the advection and diffusive terms that exist within complicated conservation equations in order to ensure monotonic solutions. The use of classical upstream weighting for these terms may induce excessive numerical diffusion. Here, the previously developed schemes for structured meshes will be tested and analysed

for unstructured grids. Furthermore, a complete description of the implementation of the inexact Newton method including a discussion on the approximation of the Jacobian matrix, together with an investigation of the solution of the linearised system using preconditioned BiCGSTAB [13] and GMRES [10], is also deliberated.

2   Transport model

The objective of this research is to formulate a generalised computational solution methodology for the following non-linear transport equation that describes the evolution of liquid conservation (moisture content) during low temperature wood drying for use in on-line kiln control,

$$\frac{\partial}{\partial t}\left(\rho_0 X + \varepsilon_g \rho_v\right) + \nabla \cdot \left(\rho_w \mathbf{v}_w + \rho_v \mathbf{v}_g\right) = \nabla \cdot \left(\mathbf{D}_{\text{dif}} \nabla X_b + \rho_g \mathbf{D}_{\text{eff}} \nabla \omega_v\right).$$
(1)

Although in the past three coupled non-linear transport equations have been used to model the drying of wood [7, 11], under the assumptions of constant temperature and constant pressure this one equation model is valid for drying temperatures not exceeding 60°C. A typical experimental kiln operating schedule that controls both the dry and wet bulb temperatures is used to determine the drying characteristics of the hot convected air that is used to dry the wood sample.

The primary variable in Equation (1) is the moisture content X, which comprises both free and bound water. The remaining symbols are secondary variables and constants, where ε_g is the volume fraction of gas, ω_v the vapour mass fraction, ρ_0 the specific wood density and ρ represents density. The subscripts w, g and v represent liquid, gas and vapour, respectively. The bound moisture

content is represented by X_b. The tensors $\mathbf{D}_{\text{dif}}$ and $\mathbf{D}_{\text{eff}}$ represent the bound water diffusivity and gaseous diffusion, respectively. The gas and liquid phase velocities are given by the Generalised Darcy's Law

$$\mathbf{v}_l = -\mathbf{K}_l \boldsymbol{\lambda}_l \nabla \varphi_l \,, \quad \nabla \varphi_l = \nabla P_l - \rho_l g \nabla \chi \,, \quad l = w, g \,,$$

where φ is the phase potential, P the pressure, g the gravitational acceleration and χ the depth scalar. The tensors $\mathbf{K}$ and $\boldsymbol{\lambda}$ represent the absolute permeability and the mobility, respectively. The physical parameters needed for the model have been determined experimentally [8, 9]. The boundary fluxes take the form

$$\mathbf{J}_w \cdot \hat{\mathbf{n}}_b = k_m c M_v \ln \left(\frac{1 - x_\infty}{1 - x_v} \right) \,,$$

where $\hat{\mathbf{n}}_b$ is the outward unit normal vector, k_m the mass transfer coefficient, c the molar concentration, M_v the molecular weight of vapour, x_v and x_∞ are the molar fractions at the surface of the wood and in the air, respectively. The boundary flux on symmetry planes is assumed to be zero.

The initial moisture content field has to be determined prior to the commencement of the drying process. Firstly, the free water saturation S_{fw_i} at each of the N nodes in the mesh and the equilibrium capillary pressure $P_{c_{\text{eqm}}}$, are calculated by solving the non-linear system of $N + 1$ equations $\mathbf{F}_0 (\mathbf{x}) = \mathbf{0}$. The coordinate functions of $\mathbf{F}_0$ are

$$\begin{aligned}
f_i (\mathbf{x}) &= P_c \left(S_{\text{fw}_i}, \rho_{0_i}, T \right) - P_{c_{\text{eqm}}} \,, \\
f_{N+1} (\mathbf{x}) &= \rho_w \frac{\sum_{i=1}^n \phi_i S_{\text{fw}_i} A_i}{\sum_{i=1}^n A_i \rho_{0_i}} + X_{\text{fsp}} - \bar{X} \,,
\end{aligned}$$

where $\mathbf{x} = \left(S_{\text{fw}_1}, \ldots, S_{\text{fw}_N}, P_{c_{\text{eqm}}} \right)^T$, P_c is the capillary pressure function, T the initial temperature, $\bar{X}$ the initial average moisture content, X_{fsp} the fibre saturation point, ϕ the porosity and A the area of
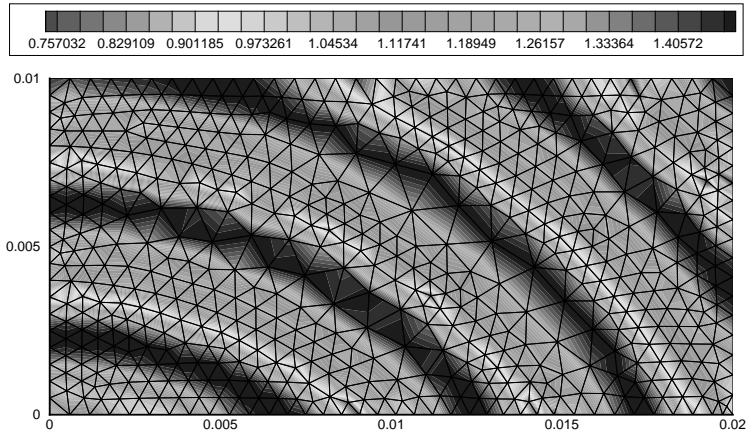
FIGURE 1: Initial moisture content field and triangular element mesh.

the control volume. The initial moisture content field, see Figure 1, is then calculated using

$$X_i = \frac{\rho_w \phi_i S_{\mathrm{fw}_i}}{\rho_{0_i}} + X_{\mathrm{fsp}} \, .$$

3   Computational model

3.1   Control volume finite element method

To solve the transport model numerically the CVFE method is used
to discretise the transport equation (1), rewritten as

$$\frac{\partial \psi}{\partial t} + \nabla \cdot \mathbf{J} = 0 \,, \tag{2}$$
$$\psi = \rho_0 X + \varepsilon_g \rho_v \,,$$
$$\mathbf{J} = \rho_w \mathbf{v}_w + \rho_v \mathbf{v}_g - \mathbf{D}_{\text{dif}} \nabla X_b - \rho_g \mathbf{D}_{\text{eff}} \nabla \omega_v \,,$$

where $\mathbf{J}$ denotes the flux vector. The domain of the problem is
meshed with finite triangular elements, as shown in Figure 1. The
control volumes are constructed around the node points, which are
the vertices of the elements and also the locations where the values
of the discrete variables are associated. The CVFE method is cho-
sen because it applies easily to unstructured meshes, ensures that
the moisture balance remains conserved at the discrete level and is
flexible in that it allows for a number of different possibilities for
approximating the flux though the control volume (CV) face.

Equation (2) is integrated in both time and space to arrive at
this discrete analogue of the transport equation:

$$\frac{\delta V_i}{\delta t} \left(\psi_i^{n+1} - \psi_i^n \right) + \alpha \sum_j (\mathbf{J} \cdot \mathbf{n})_j^{n+1} + (1 - \alpha) \sum_j (\mathbf{J} \cdot \mathbf{n})_j^n = 0 \,. \tag{3}$$

The superscript $n+1$ denotes the next time level and n the current
time level, the subscript i represents the ith node, δV_i is the volume
of the ith CV, δt the time step, while j sums over the CV faces where
$\mathbf{n}$ is the outward normal which takes into account the CV face area.
The parameter α allows for a fully explicit system with $\alpha = 0$,

a fully implicit system with $\alpha = 1$, and a second order implicit system (Crank-Nicolson) with $\alpha = 0.5$. The fully explicit scheme is not considered here due to the restrictive constraint imposed on the time step and mesh size. The fully implicit system is only first order in time but is stable. Although the Crank-Nicolson system is second order in time, it may become non-monotonic for particular time step and spatial step combinations, causing non-physical results.

3.2   Non-linear solver

Applying the discretisation formula (3) to each of the N nodes in the mesh results in a system of N non-linear discrete equations, each denoted by F_i. This system

$$\begin{aligned} \mathbf{F}\left(\mathbf{u}\right) \;\; &= \;\; \left(F_1\left(\mathbf{u}\right), F_2\left(\mathbf{u}\right), \ldots, F_N\left(\mathbf{u}\right)\right)^T = \mathbf{0}, \\ \mathbf{u} &= \left(X_1, X_2, \ldots, X_N\right)^T, \end{aligned}$$

must be solved simultaneously for each time step in order to advance the solution in time. In this work an inexact Newton method [2] is used:

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \delta\mathbf{u}^{(k)}, \quad \mathbf{J}_m\left(\mathbf{u}^{(k)}\right)\delta\mathbf{u}^{(k)} = -\mathbf{F}\left(\mathbf{u}^{(k)}\right),$$

where $\mathbf{J}_m$ is an approximation of the Jacobian matrix and $\delta\mathbf{u}^{(k)}$ the Newton correction. The iterations are continued until the convergence criterion

$$\|\mathbf{F}\left(\mathbf{u}^{(k+1)}\right)\| < \varepsilon_{\text{tol}} \approx 10^{-7},$$

is reached. The Jacobian matrix must be generated and then the corresponding linear system solved for each iteration of the Newton method. The function F_i depends upon only a small subset of the entire solution vector, which is influenced by the method used to calculate the flux. Hence, the Jacobian matrix is sparse and in

this work is solved using either BiCGSTAB or GMRES, with various preconditioners.

The construction of the Jacobian matrix requires the derivatives of the accumulation term and all flux terms. To calculate these derivatives a first order numerical approximation is used:

$$\frac{\partial F_i}{\partial X_j} = \frac{F_i\left(\mathbf{u} + \epsilon\|\mathbf{u}\|\mathbf{e}_j\right) - F_i\left(\mathbf{u}\right)}{\epsilon\|\mathbf{u}\|},$$

where $\epsilon \approx 10^{-7}$, $\|\mathbf{u}\|$ is used for scaling purposes and $\mathbf{e}_j$ is the jth unit vector. Therefore to build the Jacobian matrix, the flux terms together with the shifted flux terms must be evaluated at each CV face, which is computationally expensive. However, this expense is reduced by assuming that the flux is split into implicit and explicit components at a given Newton iteration. Implicit components remain in the Jacobian matrix whilst explicit components appear only in the function vector. Firstly the fill-in caused by the flux limiter (Section 3.3) is ignored, which has been shown to be an acceptable approximation [12]. Secondly, only the two nodes that constitute the element face that intersects the CV face on which the flux is being evaluated, are treated implicitly, referred to as the two-node Jacobian approximation. Both of these approximations reduce the number of flux derivatives to be calculated, hence reducing the time required to generate the Jacobian.

3.3   Flux limiting

The flux function has both advection and diffusive terms. The way in which these terms are approximated at the CV face is crucial to the overall accuracy and monotonicity of the solution. The diffusive terms require that both the values and the gradients of particular secondary variables are computed at the CV faces. Finite element

shape functions [3] are used for these calculations. For example, the formulas for $\mathbf{D}_{\text{dif}}$ and ∇X_b are

$$\mathbf{D}_{\text{dif}} = \sum_{k=1}^{M} N^k \mathbf{D}_{\text{dif}}^k, \quad \nabla X_b = \sum_{k=1}^{M} \nabla N^k X_b^k,$$

where the N^k are the shape functions of a finite element with M vertices.

The advection terms must be treated in such a way that ensures the correct physical behaviour of the solution. The values of particular secondary variables must be calculated at the CV faces. First order spatial upwinding uses the flow direction to find the upstream node and then uses this node as the CV face representative point. Whereas this method is stable it does however introduce a large amount of numerical diffusion [5, 6]. A more general method is flux limiting, which has been found to produce superior results to upwind schemes by sharpening saturation fronts [11, 12]. The method requires both the upwind and downwind nodes, as well as the second upwind node, which is found by using the maximum flow method that seeks to track the location of the streamline from the upwind node. For example, using this technique to estimate the CV face value of the liquid mobility $\boldsymbol{\lambda}_w$ would be

$$\boldsymbol{\lambda}_w = \boldsymbol{\lambda}_w^{\text{UP}} + \frac{\sigma(r)}{2} \left(\boldsymbol{\lambda}_w^{\text{DN}} - \boldsymbol{\lambda}_w^{\text{UP}} \right),$$

where σ is the limiter function and r is known as the sensor. There are various methods for computing the sensor, here four different sensors are investigated. The first uses the ratio of the difference in the pressure P_w,

$$r = \left(\frac{P_w^{\text{2UP}} - P_w^{\text{UP}}}{\|\mathbf{x}^{\text{2UP}} - \mathbf{x}^{\text{UP}}\|} \right) \Big/ \left(\frac{P_w^{\text{UP}} - P_w^{\text{DN}}}{\|\mathbf{x}^{\text{UP}} - \mathbf{x}^{\text{DN}}\|} \right),$$

where $\mathbf{x}$ denotes the positional vector. Replacing the pressure with the phase potential φ_w gives the second method. An alternative approach is to use the ratio of the inner product of the gradient of the phase potential $\nabla\varphi_w$ at the CV face and the CV face unit normal $\hat{\mathbf{n}}$,

$$r = \frac{(\nabla\varphi_w \cdot \hat{\mathbf{n}})^{2\mathrm{UP}}}{(\nabla\varphi_w \cdot \hat{\mathbf{n}})^{\mathrm{UP}}} \, .$$

Replacing the gradient of the phase potential with the phase velocity $\mathbf{v}_w$ gives the fourth method. The four sensors all have the domain $[0, \infty)$. The range of the limiter function σ must be contained in $[0, 2]$, where 0 equates to upwinding, 1 averaging and 2 downwinding. The first of two limiter functions investigated is the van Leer limiter [14] defined by $\sigma(r) = 2r/(1 + r)$, which has a range $[0, 2)$. The second is the parabolic limiter [1] defined by

$$\sigma(r) = \left\{ \begin{array}{cl} r(2 - r), & r < 1 \\ 1, & r \geq 1 \end{array} \right. ,$$

which has a range $[0, 1]$, to avoid any downwinding.

4   Results and discussion

4.1   Linear benchmark case study

Before using the computational model described in Section 3 to solve the non-linear transport equation, a linear benchmark problem that admits an analytical solution is used to analyse the accuracy and efficiency of the different numerical schemes. Defining the flux in Equation (2) by $\mathbf{J} = \psi\mathbf{v} - \mathbf{D}\nabla\psi$, where $\mathbf{D} = \mathrm{diag}(\alpha_x, \alpha_y)$ and $\mathbf{v} = (u, v)^T$, on the domain $[0, X] \times [0, Y]$, $0 \leq t \leq T$, with the

initial condition of a two-dimensional Gauss pulse of unit height and centred at the location (x_c, y_c), leads to the exact solution

$$\psi^e = \frac{1}{4t+1} \exp\left\{ -\frac{(x - ut + x_c)^2}{\alpha_x (4t+1)} - \frac{(y - vt + y_c)^2}{\alpha_y (4t+1)} \right\}. \quad (4)$$

The four boundary conditions are obtained by substituting $x = 0$, $x = X$, $y = 0$ and $y = Y$, into Equation (4) respectively. For $X = Y = 2$, $T = 1.25$, $\alpha_x = \alpha_y = 0.01$, $u = v = 0.8$ and $x_c = y_c = 0.5$, the exact solution is a pulse with height 0.166667 and centred at $(1.5, 1.5)$, see Figure 2(a).

Note: the complete non-linear solver is employed for this benchmark problem in order to test the construction of the Jacobian matrix, the unstructured discretisation methodology and the flux limiter formulation. The accuracy of the computed solution is monitored using two error parameters, the pulse height which observes the amount of numerical diffusion, and the root mean square relative error

$$\text{RMSRE} = \sqrt{\frac{\sum_{i=1}^{N} \left(\psi_i^e - \psi_i \right)^2}{\sum_{i=1}^{N} \left(\psi_i^e \right)^2}},$$

which measures the quality of the solution in comparison to the exact solution. The efficiency of the numerical scheme is gauged by observing the CPU time required for the non-linear solver. The numerical solution is computed on two triangular element meshes, a coarse mesh with 3308 nodes and a fine mesh with 49733 nodes.

Table 1 contains the results for the analysis of the spatial and temporal weighting schemes. The first order temporal scheme combined with upwinding is very stable, in that it does not produce non-physical results even for relatively large time step sizes. However, the error is large and a considerable smearing of the Gaussian pulse is observed, see Figure 2(b). For this method the time step

TABLE 1: Benchmark problem, spatial and temporal weighting schemes.

Error $\times 10^{-4}$ Pulse height CPU time (sec)	First order temporal with upwinding	Second order temporal with upwinding	Second order temporal with flux limiting
Coarse mesh (3308 nodes)	0.496365 0.069628 21.9	0.433244 0.082175 4.5	0.169803 0.136391 26.6
Fine mesh (48733 nodes)	0.305122 0.107092 410.3	0.173284 0.133346 148.0	0.057558 0.161249 685.4

size was fixed at 0.0125 seconds, since further reduction in the time step gave little improvement in the quality of the numerical solution. When applying the second order temporal weighting scheme the time step size could be increased whilst still maintaining good accuracy. Thus, although the improvement in the error and numerical diffusion was only moderate, the CPU time is significantly reduced. The introduction of flux limiting increases the spatial order of accuracy. When comparing the results for the same mesh size, flux limiting greatly reduces both the error and numerical diffusion, see Figure 2(c), but at a cost of increasing the CPU time. However, when comparing flux limiting on the coarse mesh with upwinding on the fine mesh, see Figure 2(d), the quality of the numerical solutions are similar. Hence we conclude that flux limiting produces comparable results with less computational overhead.

For the flux limited second order temporal scheme, the two-node Jacobian approximation reduced the CPU time by 17.7% for the coarse mesh and 9.3% for the fine mesh. The difference in the computed solutions was insignificant for the coarse mesh and only
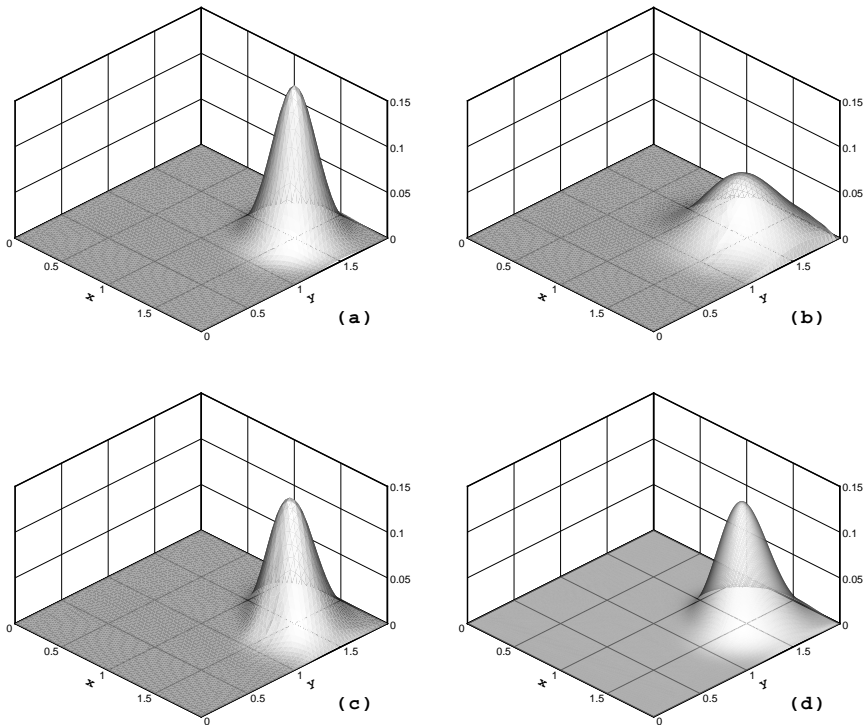
FIGURE 2: Benchmark problem, spatial and temporal weighting schemes: (a) exact solution, (b) first order temporal with upwinding, (c) second order temporal with flux limiting, (d) second order temporal with upwinding on the fine mesh.

Table 2:   Transport model, spatial and temporal weighting schemes.

CPU time (sec): upwinding flux limiting	First order temporal	Second order temporal
Coarse mesh (603 nodes)	40.7 31.7	72.1 70.1
Fine mesh (1846 nodes)	107.1 108.8	413.5 507.9

marginal for the fine mesh. The approximation caused an increase in the number of Newton iterations for the fine mesh, this explains why the reduction in CPU time was not as great.

4.2   Transport model

The solution of the non-linear transport model described in Section 2 was computed with the numerical model described in Section 3. Two triangular element meshes were used, a coarse mesh with 603 nodes and a fine mesh with 1846 nodes. The solution was computed on the domain $[0, 0.02] \times [0, 0.01]$ and up to a time of 54000.0 seconds (15 hours), at which time the wood sample is considered to be dry.

Table 2 contains the results for the analysis of the spatial and temporal weighting schemes. The second order temporal scheme requires substantially more CPU time than the first order scheme, however the difference in the computed solutions is minimal. Hence, the first order temporal method is the preferred option. When com-

TABLE 3:   Transport model, flux limiter functions and sensors.

CPU time (sec) ($\infty \equiv$ non-physical)	ΔP	$\Delta\varphi$	$\nabla\varphi \cdot \hat{\mathbf{n}}$	$\mathbf{v} \cdot \hat{\mathbf{n}}$
van Leer	31.7	33.5	∞	64.8
Parabolic	31.4	31.6	30.8	38.2

paring upwinding and flux limiting on the fine mesh there is little difference in the solutions, see Figures 3(a) and 3(b), hence the effectiveness of flux limiting on the coarse mesh will be gauged by comparing its solution to that of the solutions computed on the fine mesh. On the coarse mesh the solution of the upwinding scheme is different to that of the flux limiting scheme, see Figures 3(c) and 3(d), with the flux limited solution resembling that of the solutions computed on the fine mesh. Thus flux limiting produces a similar result on the coarse mesh to that of upwinding on the fine mesh, and at a third of the computational time.

Regarding the linear solvers, both BiCGStab and gmres produced identical solutions in virtually equal CPU times, when either ilu(0) or ssor preconditioning was implemented. For both BiCGStab and gmres the ilu(0) preconditioner slightly outperformed the ssor preconditioner. For the flux limited first order temporal scheme on the coarse mesh, the two-node Jacobian approximation reduced the CPU time spent on calculating the shifted fluxes by 34.6%. This is expected since the shifted flux is now only calculated at two of the three CV faces per element. This saving resulted in a 17.7% reduction of the total CPU time, now 26.1 seconds as opposed to 31.7 seconds. The two-node Jacobian approximation did not impact the convergence of the Newton solver for the coarse mesh. However, for the fine mesh this approximation did increase
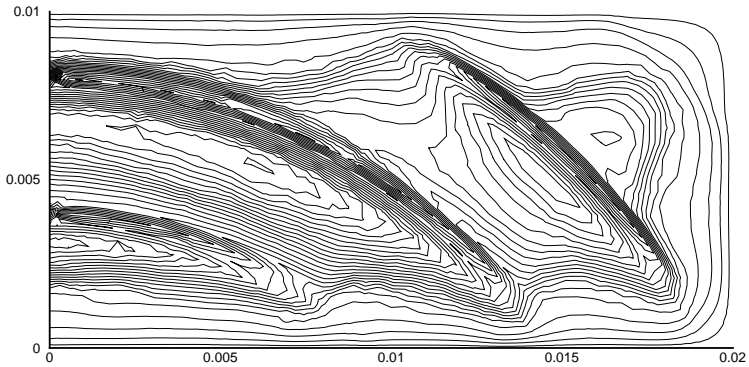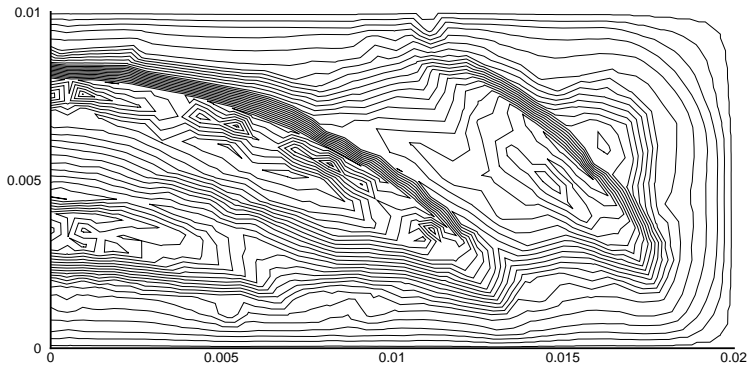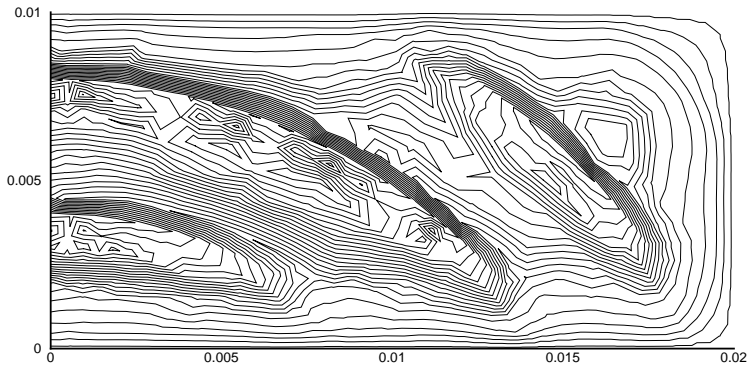
(a)
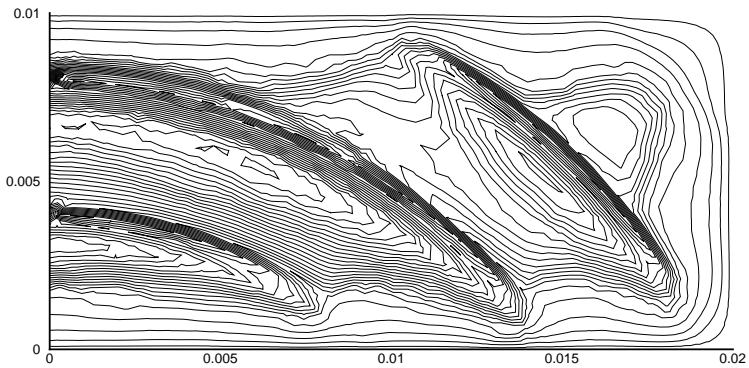
(b)

FIGURE 3: Transport model, spatial weighting schemes: (a) up-winding and (b) flux limiting on the fine mesh; (next page) (c) up-winding and (d) flux limiting on the coarse mesh.

(c)

(d)

FIGURE 3: (continued) (c) upwinding and (d) flux limiting on the coarse mesh.

the number of Newton iterations, resulting in only a 29.2% reduction of CPU time spent on calculating the shifted fluxes, and in turn reducing the total CPU time to 95.2 seconds. Table 3 contains the results for the analysis of the various flux limiter functions and sensors. For the ΔP sensor, the two limiter functions produced solutions of equal quality and in similar CPU time. However, the parabolic limiter is the most reliable, in that the solution accuracy and efficiency does not vary depending on the choice of the sensor.

5   Conclusions

A complete CVFE solution methodology suitable for resolving non-linear transport equations on triangular meshes has been presented. When analysed for the linear benchmark problem, the methodology highlights the accuracy and computational efficiency offered by the second order temporal scheme coupled with flux limiting. Further, flux limiting clearly reduces numerical diffusion and offers similar accuracies to that of results obtained when using upwinding on fine meshes. For the non-linear transport model, second order temporal weighting offered no significant change in the solution quality, however it did substantially increase the CPU time. The combination of first order temporal weighting and flux limiting enabled the accuracy of the coarse mesh solution to be greatly improved. The preconditioned linear solvers BiCGSTAB and GMRES both performed equally, with ILU(0) preconditioning being the optimal choice. Using the two-node Jacobian approximation reduced the time required for the construction of the linearised system. In summary, the CVFE method that employs flux limiting with the maximum flow potential method is highly recommended for unstructured meshes.

References

[1] P. Arminjon, A. Dervieux, Construction of TVD-like artificial viscosities on two-dimensional arbitrary FEM grids, *J. Comp. Phys.* 106 (1993) 176–198. C769

[2] R. L. Burden, J. D. Faires, *Numerical analysis* (sixth edition), Brooks/Cole Publishing Company, 1997. C766

[3] W. J. Ferguson, I. W. Turner, A control volume finite element numerical simulation of the drying of spruce, *J. Comp. Phys.* 125 (1996) 59–70. C761, C768

[4] P. A. Forsyth, Three-dimensional modelling of steam flush for DNAPL site remediation, *Int. J. Num. Methods in Fluids*, 19 (1994) 1055–1081. C761

[5] K. W. Morton, *Numerical solution of convection-diffusion problems, Applied mathematics and mathematical computation* 12, Chapman & Hall, 1996. C768

[6] S. V. Patankar, *Numerical heat transfer and fluid flow*, Hemisphere Publishing Corporation, McGraw Hill, 1980. C768

[7] P. Perré, I. W. Turner, TransPore: a generic heat and mass transfer computational model for understanding and visualising the drying of porous media, invited paper, *J. Drying Technology*, 17 (7818) (1999) 1273–1289. C761, C762

[8] P. Perré, I. W. Turner, Determination of the material property variations across the growth ring of softwood for use in a heterogeneous drying model; Part I, capillary pressure, tracheid model and absolute permeability correlations,

Holzforschung Journal, Walter de Gruyter, 55 (3) (2001) 318–323. C763

[9] P. Perré, I. W. Turner, Determination of the material property variations across the growth ring of softwood for use in a heterogeneous drying model; Part II, the use of homogenisation to predict bound liquid diffusivity and thermal conductivity correlations, *Holzforschung Journal*, Walter de Gruyter, 55 (4) (2001) 417–425. C763

[10] Y. Saad, M. H. Schultz, GMRES: a generalized minimum residual algorithm for solving nonsymmetric linear systems, *SIAM J. Sci. Stat. Comput.* 7 (1986) 856–869. C762

[11] I. W. Turner, P. Perré, The use of implicit flux limiting schemes in the simulation of the drying process: a new maximum flow sensor applied to phase mobilities, *Appl. Modelling*, 25 (2001) 513–540. C761, C762, C768

[12] A. J. A. Unger, P. A. Forsyth, E. A. Sudicky, Variable spatial and temporal weighting schemes for use in multi-phase compositional problems, *Advances in Water Resources*, 19 (1) (1996) 1–27. C761, C767, C768

[13] H. A. Van Der Vorst, Bi-CGSTAB: a fast and smoothly convergent variant of Bi-CG for the solution of nonsymmetric linear systems, *SIAM J. Sci. Stat. Comput.* 13 (2) (1992) 631–644. C762

[14] B. van Leer, Towards the ultimate conservation difference scheme, II, monotonicity and conservation combined in a second order scheme, *J. Comp. Phys.* 14 (1974) 361–370. C769