Commun. Comput. Phys.
doi: 10.4208/cicp.201010.040511a

Vol. **10**, No. 3, pp. 509-576
September 2011

REVIEW ARTICLE

Velocity-Based Moving Mesh Methods for Nonlinear Partial Differential Equations

M. J. Baines[1,*], M. E. Hubbard[2] and P. K. Jimack[2]

[1] *Department of Mathematics, The University of Reading, RG6 6AX, UK.*
[2] *School of Computing, University of Leeds, LS2 9JT, UK.*

Received 20 October 2010; Accepted (in revised version) 4 May 2011

Available online 1 June 2011

Abstract. This article describes a number of velocity-based moving mesh numerical methods for multidimensional nonlinear time-dependent partial differential equations (PDEs). It consists of a short historical review followed by a detailed description of a recently developed multidimensional moving mesh finite element method based on conservation. Finite element algorithms are derived for both mass-conserving and non mass-conserving problems, and results shown for a number of multidimensional nonlinear test problems, including the second order porous medium equation and the fourth order thin film equation as well as a two-phase problem. Further applications and extensions are referenced.

AMS subject classifications: 35R35, 65M60, 76M10

Key words: Time-dependent nonlinear diffusion, moving boundaries, finite element method, Lagrangian meshes, conservation of mass.

Contents

*Corresponding author. *Email addresses:* m.j.baines@reading.ac.uk (M. Baines), m.e.hubbard@leeds.ac.uk (M. E. Hubbard), p.k.jimack@leeds.ac.uk (P. K. Jimack)

1 Introduction

This paper reviews some velocity-based moving mesh numerical methods for nonlinear time-dependent partial differential equations (PDEs). Numerous physical and biological applications of mathematics are governed by these PDEs which often exhibit complex behaviour that is difficult to predict in advance. For example, problems may be posed on moving domains which are determined implicitly by the solution of the equations.

Many PDE problems possess analytic properties, for example the invariance of certain integrals or the existence of solutions with a specific structure. The construction of numerical approximations which preserve such properties is one of the aims of geometric integration [28–30]. Where such *a priori* knowledge concerning the qualitative nature of the solution is available this may be used to guide effective computational schemes.

We shall be concerned with moving mesh numerical methods, which have the ability to adjust to the evolution of the solution (in order to track implicit moving boundaries and singularities for example), as well as to resolve sharp features and respect global properties. Such methods are therefore an attractive choice for problems of this type. The argument is reinforced in the case of scale-invariant problems for which both dependent and independent variables are strongly coupled. Fixed meshes are unable to replicate scale-invariance because they are time-independent. The coupling of independent and dependent variables is a recurrent theme in this paper and is used later on to motivate the development of a solution-adaptive moving mesh finite element method based on conservation.

Velocity-based moving mesh methods (also known as *Lagrangian methods* or in a wider context *Arbitrary Lagrangian Eulerian (ALE) methods*) rely on the construction of suitable velocities at points of the moving domain at each instant of time, as opposed to the construction of time-dependent mappings from a fixed computational domain to the moving domain [30, 35, 134]. The latter construction can be rather cumbersome in more than one dimension, and in any case the mappings need to be converted into velocities for incorporation into a time-dependent PDE. A velocity-based description, on the other hand, requires no formal reference to the computational domain and has the advantage that the velocity is available directly for incorporation into the time-dependent PDE. The evolution of the Lagrangian coordinate $\widehat{\mathbf{x}}(t)$ at time t follows from the velocity $\mathbf{v}(t,\mathbf{x})$ by integrating the ODE

$$\frac{d\widehat{\mathbf{x}}(t)}{dt} = \mathbf{v}(t,\widehat{\mathbf{x}}(t)), \tag{1.1}$$

where $\widehat{\mathbf{x}}(t)$ coincides instantaneously with the Eulerian coordinate $\mathbf{x}$ at the initial time.

The layout of the paper is as follows. In the next section (Section 2) we review time-dependent PDEs stated in both differential and integral form. Initially a fixed frame

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

of reference is considered which allows a simple discussion of some structural aspects of time-dependent PDEs, including scale invariance and similarity. We then generalize these aspects to PDEs in a moving framework. Although the paper is primarily concerned with differential equations the emphasis here is on integral forms which are closer to the physics and lead naturally to finite element formulations. The discussion includes weak formulations of the equations, of direct relevance to the finite element methods that are considered later.

Section 3 focuses on the history of the construction of mesh velocities that have been incorporated into these moving forms. A short survey is given which starts by considering Lagrangian methods in fluid dynamics, and then moves on to ALE techniques and other more mathematically (rather than physically) based drivers for the mesh velocities. This section includes a discussion of the geometric conservation law, which is a source for the moving mesh conservation method considered later in the paper.

Subsequent sections give details of a local, or distributed, conservation method which has been shown to be highly effective in obtaining approximate solutions of time-dependent PDE problems in multidimensions, particularly those with implicit moving boundaries. The first of these sections (Section 4) describes the basis of the conservation method for mass-conserving problems and includes details of the multidimensional moving mesh finite element method for such problems with examples. Section 5 generalizes the technique to non mass-conserving problems, whilst Section 6 presents a number of extensions and applications of the method. The paper concludes with a brief summary.

2 Time-dependent PDEs

This introduction contains background material on the structure of PDEs relevant to later sections of the paper.

2.1 PDEs and balance laws in a fixed frame

In a fixed frame of reference a generic form of scalar time-dependent PDE for the dependent variable $u(t,\mathbf{x})$ is

$$u_t = \mathcal{L}u, \tag{2.1}$$

where t is time, $\mathbf{x}$ is the space variable, and $\mathcal{L}$ is a purely spatial operator, nonlinear in general. Problems governed by the PDE (2.1) may be posed on either fixed or moving domains with appropriate initial and boundary conditions. The number and character of the boundary conditions relate to the order and type of the operators involved and the behaviour (fixed or moving) of the boundary. In a moving boundary problem an additional boundary condition is normally required.

We shall also be concerned with integral forms of (2.1), which are closer to the physics through their direct expression of conservation properties, as well as leading naturally to

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

finite element formulations. Eq. (2.1) has the integral form

$$\frac{d}{dt}\int_\Omega u d\mathbf{x} = \int_\Omega \mathcal{L}u d\mathbf{x} \tag{2.2}$$

over any region Ω.

In particular, many partial differential equations governing applications are derived from integral balance laws. A typical balance law expresses the balance between the rate of increase of the integral of the function u over an arbitrary volume Ω,

$$\int_\Omega u d\mathbf{x}, \tag{2.3}$$

henceforth known as the mass (by analogy with the case where u is a density), and the flux $\mathbf{F}(u, D^\alpha u)$ across the boundary $\partial\Omega$ of Ω (where D^α denotes derivatives of u up to some order α) and is given by

$$\frac{d}{dt}\int_\Omega u d\mathbf{x} = -\oint_{\partial\Omega} \mathbf{F}\cdot\hat{\mathbf{n}} d\Gamma, \tag{2.4}$$

where $\hat{\mathbf{n}}$ is the unit outward normal to the boundary. By the Divergence Theorem equation (2.4) can be written as

$$\frac{d}{dt}\int_\Omega u d\mathbf{x} = -\int_\Omega \nabla\cdot\mathbf{F} d\mathbf{x}. \tag{2.5}$$

For fixed Ω this leads to the integral form

$$\int_\Omega \{u_t + \nabla\cdot\mathbf{F}\} d\mathbf{x} = 0, \tag{2.6}$$

and, since Ω is arbitrary, u satisfies the PDE

$$u_t + \nabla\cdot\mathbf{F} = 0 \tag{2.7}$$

pointwise. The physical origin of the PDE (2.7) is nevertheless the balance law (2.4).

Before putting PDEs into a moving framework we consider the application of scale-invariance and similarity to PDEs in a fixed frame.

2.2 Scale-invariance and similarity

Information about the structure of PDEs and their solutions is reflected in their scale-invariance and similarity properties. Because scale-invariance depends on the simultaneous variation of independent and dependent variables the numerical simulation of the PDEs demands (and motivates) the use of solution-adaptive moving meshes [31].

Recall that a time-dependent PDE problem in d dimensions with spatial coordinates x_ν ($\nu = 1,2,\cdots,d$) is *scale-invariant* if there exists a scaling parameter λ and scaling indices β_ν, γ such that the scalings

$$t \to \lambda t, \quad x_\nu \to \lambda^{\beta_\nu} x_\nu, \quad u \to \lambda^\gamma u \tag{2.8}$$

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

leave the problem invariant.

For such problems there exist (scale-invariant) similarity variables,

$$\xi_\nu = \frac{x_\nu}{t^{\beta_\nu}}, \qquad \eta = \frac{u}{t^\gamma}, \tag{2.9}$$

which specify a time-dependent *similarity transformation* between a fixed domain with coordinates ξ_ν and a moving domain with coordinates $x_\nu(t)$. The velocity effecting the transformation has components

$$v_\nu = \frac{dx_\nu}{dt} = \beta_\nu t^{\beta_\nu - 1}\xi_\nu = \frac{\beta_\nu x_\nu}{t}. \tag{2.10}$$

A relationship between the similarity variables (2.9) of the form

$$\eta = f(\boldsymbol{\xi}), \quad \text{i.e., } u(t,\mathbf{x}) = t^\gamma f(\boldsymbol{\xi}), \tag{2.11}$$

where $\boldsymbol{\xi} = \{\xi_\nu\}$, plays a special role for which u is said to be *self-similar*. Substituting (2.11) into (2.1) the function f must satisfy the reduced-order differential equation,

$$\gamma f(\boldsymbol{\xi}) - \sum_\nu \beta_\nu \xi_\nu \frac{\partial f}{\partial \xi_\nu}(\boldsymbol{\xi}) = t^{1-\gamma}\mathcal{L}\{t^\gamma f(\boldsymbol{\xi})\} \tag{2.12}$$

(which is time-independent owing to the scaling properties). Self-similar scaling solutions of the form (2.11) exhibit a *scaling symmetry* because the number of independent variables is reduced by one.

In particular, for mass-conserving problems,

$$\gamma + \sum_\nu \beta_\nu = 0 \tag{2.13}$$

as a result of the integral (2.3) being independent of time.

In one space dimension the self-similar solution (2.11) takes the form

$$u(t,x) = t^\gamma f(\xi), \tag{2.14}$$

where $\xi = x/t^\beta$ and f satisfies the ODE

$$\gamma f(\xi) - \beta\xi f'(\xi) = t^{1-\gamma}\mathcal{L}\{t^\gamma f(\xi)\}. \tag{2.15}$$

A particular property of self-similar solutions for mass-conserving problems (for which (2.13) holds) is the invariance of the integral (2.11) via

$$\int_{\Omega(t)} u\,d\mathbf{x} = t^\gamma \int_{\Omega(t)} f(\boldsymbol{\xi})\,d\mathbf{x}, \tag{2.16}$$

in which $\Omega(t)$ moves with the similarity velocity with components (2.10). This follows from substituting $x_\nu = t^{\beta_\nu}\xi_\nu$ into (2.16), noting that $\Omega(t)$ is mapped onto a fixed region, and using (2.13).

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

Barenblatt [14, 15] has pointed out that self-similar solutions of PDEs often describe the *intermediate asymptotics* of a solution, that is the behaviour of the solution, away from the boundaries, after the transient effects of initial conditions have died away. They may also characterize singular behaviour such as the occurrence of blow-up, as well as the motion of interfaces. These features motivate numerical methods which are able to accurately follow such solutions, such as the conservation method described later in Section 4.

2.2.1 The porous medium equation

An example of a scale-invariant problem is the porous medium equation (PME),

$$u_t = \nabla\cdot\left(u^n\nabla u\right) \tag{2.17}$$

($n \geq 1$) in d dimensions, with the boundary condition $u = 0$ on the moving boundary (for which the total mass (2.3) is preserved — see below). For this problem it is readily verified that the scaling indices in (2.8) are given by

$$\beta_\nu = \beta = \frac{1}{(2+dn)} \quad \forall\,\nu, \qquad \sum_\nu \beta = -\gamma = \frac{d}{(2+dn)}, \tag{2.18}$$

from (2.17) and (2.13). The similarity variables (2.9) are

$$\xi_\nu = \frac{x_\nu}{t^{1/(2+dn)}}, \qquad \eta = \frac{u}{t^{-d/(2+dn)}}, \tag{2.19}$$

and a self-similar source solution exists of the form [3, 142]

$$u(t,\mathbf{x}) = At^{-\frac{d}{(2+dn)}}\left\{1-\sum_\nu \xi_\nu^2\right\}_+^{\frac{1}{n}}, \tag{2.20}$$

where A is a constant, the suffix $+$ in (2.20) indicating the positive part. The moving boundary corresponds to $|\boldsymbol{\xi}| = 1$ (where $u = 0$). The velocity effecting similarity, from (2.10), has components

$$v_\nu = \beta t^{\beta-1}\xi_\nu. \tag{2.21}$$

Such solutions are *globally attracting* [3, 30, 31, 142].

2.3 PDEs and integral forms in a moving frame

We now consider forms of time-dependent partial differential equations in a moving frame of reference.

When the boundary $\partial\Omega(t)$ of a region $\Omega(t)$ moves with a normal velocity $\mathbf{v}\cdot\hat{\mathbf{n}}$, say, where $\hat{\mathbf{n}}$ is the outward unit normal, an additional flux is induced across the boundary. To

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

see this analytically, differentiate the integral (2.3) with respect to t in the moving frame of reference using the Reynolds Transport Theorem (see e.g., [150]) which gives

$$\frac{d}{dt}\int_{\Omega(t)} u d\mathbf{x} = \int_{\Omega(t)} u_t d\mathbf{x} + \oint_{\partial\Omega(t)} u\mathbf{v}\cdot\hat{\mathbf{n}} d\Gamma, \tag{2.22}$$

where the final term is the additional flux. Using the Divergence Theorem

$$\frac{d}{dt}\int_{\Omega(t)} u d\mathbf{x} - \int_{\Omega(t)} \nabla\cdot(u\mathbf{v}) d\mathbf{x} = \int_{\Omega(t)} u_t d\mathbf{x}, \tag{2.23}$$

where $\mathbf{v}$ is any sufficiently smooth velocity field that is consistent with the normal boundary velocity (cf. (2.2)). Substituting for u_t from the PDE (2.1), Eq. (2.22) can be written

$$\frac{d}{dt}\int_{\Omega(t)} u d\mathbf{x} - \oint_{\partial\Omega(t)} u\mathbf{v}\cdot\hat{\mathbf{n}} d\Gamma = \int_{\Omega(t)} \mathcal{L}u d\mathbf{x} \tag{2.24}$$

(cf. (2.2)) and (2.23) as

$$\frac{d}{dt}\int_{\Omega(t)} u d\mathbf{x} - \int_{\Omega(t)} \nabla\cdot(u\mathbf{v}) d\mathbf{x} = \int_{\Omega(t)} \mathcal{L}u d\mathbf{x}. \tag{2.25}$$

In the case of the PME (2.17), where $\mathcal{L}u = \nabla(u^n\nabla u)$, in a domain $\mathcal{R}(t)$ with zero Dirichlet boundary conditions, Eq. (2.24) leads to

$$\frac{d}{dt}\int_{\mathcal{R}(t)} u d\mathbf{x} = \oint_{\partial\mathcal{R}(t)} \left\{u^n\nabla u + u\mathbf{v}\right\}\cdot\hat{\mathbf{n}} d\Gamma = 0 \tag{2.26}$$

from which it follows that the total mass

$$\int_{\mathcal{R}(t)} u d\mathbf{x} \tag{2.27}$$

is conserved in time.

A separate way of introducing the velocity $\mathbf{v}$ into the PDE (2.1) is to use the chain rule for differentiation in the form

$$\frac{du}{dt} = u_t + \mathbf{v}\cdot\nabla u, \tag{2.28}$$

where du/dt is the rate of change of u in the moving frame. This leads to the so-called quasi-Lagrangian form of the PDE (2.1),

$$\frac{du}{dt} - \mathbf{v}\cdot\nabla u = \mathcal{L}u, \tag{2.29}$$

which is widely used to modify the PDE in conjunction with time-dependent mappings [30, 74, 105]. However, Eq. (2.29) is not consistent with the integral form (2.23) in general and lacks local mass conservation.

In the same way as in Section 2.2 it can be shown that these PDEs and integral forms in the moving framework are scale-invariant whenever those in the corresponding fixed frame are scale-invariant.

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

2.4 Weak forms

We now consider weak forms of (2.23) and (2.29) which form the basis of finite element numerical methods. A weak form of Eq. (2.23) can be obtained by considering a weighted form of the mass integral (2.3). Given a domain $\mathcal{R}(t)$ and a test function w, define the weighted integral

$$\int_{\mathcal{R}(t)} wud\mathbf{x}, \tag{2.30}$$

where it is assumed that w,u are square-integrable functions. Applying the Reynolds Transport Theorem (2.22) to the function wu we obtain

$$\frac{d}{dt}\int_{\mathcal{R}(t)} wud\mathbf{x} = \int_{\mathcal{R}(t)} (wu)_t d\mathbf{x} + \oint_{\partial\mathcal{R}(t)} wu\mathbf{v}\cdot\hat{\mathbf{n}}d\Gamma, \tag{2.31}$$

where the boundary $\partial\mathcal{R}(t)$ moves with a velocity $\mathbf{v}$. Then, using the Divergence Theorem equation (2.31) becomes

$$\frac{d}{dt}\int_{\mathcal{R}(t)} wud\mathbf{x} = \int_{\mathcal{R}(t)} \big\{(wu)_t + \nabla\cdot(wu\mathbf{v})\big\}d\mathbf{x}, \tag{2.32}$$

where $\mathbf{v}$ is again any sufficiently smooth velocity field that is consistent with the normal velocity of the boundary.

Now suppose that the points of $\mathcal{R}(t)$ move with the velocity $\mathbf{v}$ and that the test function w is frozen in $\mathcal{R}(t)$, so that w is convected with the velocity $\mathbf{v}$ in the sense that

$$w_t + \mathbf{v}\cdot\nabla w = 0. \tag{2.33}$$

Eq. (2.32) then reduces to

$$\frac{d}{dt}\int_{\mathcal{R}(t)} wud\mathbf{x} = \int_{\mathcal{R}(t)} w\big\{u_t + \nabla\cdot(u\mathbf{v})\big\}d\mathbf{x} \tag{2.34}$$

for all square-integrable functions w that are convected with the velocity $\mathbf{v}$, giving a weak form of the Reynolds Transport Theorem (2.22). Using the PDE (2.1), Eq. (2.34) yields the integral weak form

$$\frac{d}{dt}\int_{\mathcal{R}(t)} wud\mathbf{x} = \int_{\mathcal{R}(t)} w\big\{\mathcal{L}u + \nabla\cdot(u\mathbf{v})\big\}d\mathbf{x} \tag{2.35}$$

of the PDE (cf. (2.25)).

A separate weak *differential* form of the PDE in a frame moving in the velocity field $\mathbf{v}$ may be derived directly from (2.29) via the chain rule as

$$\int_{\mathcal{R}(t)} w\Big\{\frac{du}{dt} - \mathbf{v}\cdot\nabla u\Big\}d\mathbf{x} = \int_{\mathcal{R}(t)} w\mathcal{L}ud\mathbf{x}, \tag{2.36}$$

where du/dt is the derivative of u in the moving frame. Note once again that this form is not elementwise mass-conserving nor consistent with (2.35).

It can be shown that as long as the test functions w are scale-invariant these weak forms in the moving framework are scale-invariant whenever those in the corresponding fixed frame are scale-invariant.

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

3 Velocity-based moving mesh methods

Each of the moving PDE forms described above requires a definition of the velocity field $\mathbf{v}(t,\mathbf{x})$. We now give a short review of some of the ways of constructing this velocity that have appeared in the literature, and how they have been incorporated into moving mesh numerical methods.

In velocity-based moving mesh methods each node is assigned a velocity by which it moves. This velocity may be defined in a variety of ways, and many constructions have appeared in the literature, some mathematically based and some based on physical analogies. The mathematical approaches often use residual minimization, error estimates, or geometric considerations, whereas physical approaches usually rely on Lagrangian fluid dynamics or mechanical analogies. Mesh movement may also be induced entirely by the normal velocities of the boundaries.

In this section a number of velocity-based methods are described, beginning with schemes related to fluid dynamics (Section 3.1) and continuing with those relying on the so-called ALE (Arbitrary Lagrangian-Eulerian) formulation. Many ALE schemes use velocities generated by mechanical analogies and these are discussed in Section 3.2.

Turning to mathematically motivated constructions we describe in Section 3.3 the Moving Finite Element Method, which was the first method to determine the mesh and the solution simultaneously. Next the Deformation Method, based on differential geometry, is described in Section 3.4. Then the role of the Geometric Conservation Law is discussed in Section 3.5, and finally the Conservation Method is introduced in Section 3.6 prior to its detailed exposition in later sections.

3.1 Fluid dynamics

In classical theoretical fluid dynamics the motion of fluids may be described by taking either the Lagrangian or Eulerian point of view. In the Lagrangian description, each moving fluid particle (with its attributes) is followed individually and is identified by its initial position, whereas in the Eulerian description variables such as density and velocity are evaluated at fixed locations.

A link between the two is provided by the Reynolds Transport Theorem (2.22) in the form

$$\frac{d}{dt}\int_{\Omega(t)}\rho d\mathbf{x}=\int_{\Omega(t)}\rho_t d\mathbf{x}+\oint_{\partial\Omega(t)}\rho\mathbf{u}\cdot\hat{\mathbf{n}}d\Gamma=\int_{\Omega(t)}\{\rho_t+\nabla\cdot(\rho\mathbf{u})\}d\Omega \tag{3.1}$$

for a general moving region $\Omega(t)$, where ρ is the fluid density and $\mathbf{u}$ is the fluid velocity. In the Lagrangian description, conservation of mass is expressed as

$$\int_{\Omega(t)}\rho d\mathbf{x}=\text{constant in time}, \tag{3.2}$$

whilst in the Eulerian description the mass conservation equation is

$$\rho_t+\nabla\cdot(\rho\mathbf{u})=0. \tag{3.3}$$

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

The equivalence of the two is inherent in (3.1), remembering that this equation holds for all $\Omega(t)$.

In more recent times, computational fluid dynamics (CFD) has sought to describe the motion of fluids numerically by both approaches. The most common approach has been through discretizations of the Eulerian description where the equations of motion are discretized on a fixed mesh. Use of the Lagrangian description has been rarer and largely confined to problems where surfaces and interfaces are of primary importance, e.g., [108,118,147]. The natural discretization in the Lagrangian approach is to follow the velocities of the fluid particles using a moving mesh but compromises have usually had to be made, the main difficulty being the tendency of the mesh to tangle and lose its character. For example, Harlen et al. [2,70] apply a Lagrangian approach to the solution of convection equations arising in the simulation of viscoelastic flows. Although the nodes of the finite element mesh are transported with the fluid particles, the mesh itself is reconnected after each time-step in order to maintain the Delaunay property [27]. These and other considerations have prompted the use of the so-called ALE (Arbitrary Lagrangian Eulerian) methods (see Section 3.2 below), where local modifications of the Lagrangian velocities are made as the computation develops [17,36]. This framework can be used with any imposed velocity and is of central relevance to velocity-based approaches.

For first order scalar PDEs, the Lagrangian approach is evident in characteristic methods (see [59,64]) and their discretizations. Indeed, wide classes of fixed mesh methods (e.g., Godunov methods [136,137] and Semi-Lagrangian methods [44,45,130]) use the ideas of characteristics, accompanied by projections.

The success of CFD has largely been achieved on fixed meshes, and indeed the same can be said for the numerical solutions of PDEs in general. Nevertheless, Lagrangian moving mesh methods have a substantial role to play in obtaining high resolution solutions of problems with implicit moving boundaries or singularities, and in mimicking scale-invariance properties.

3.2 ALE (Arbitrary Lagrangian Eulerian) methods

The ALE equation in the form (2.23) allows the solution u of a PDE to be obtained in a frame moving with any given velocity. This equation is widely used in the computation of fluid-structure interaction problems, for example in [71,83,97,122]. The specific mechanism for constructing the mesh velocities varies significantly from treating the mesh as though it is a physical material with its own constitutive law [83] through to defining the mesh motion purely with the goal of optimizing geometric qualities of the mesh [122].

Another popular technique for imposing a mesh velocity, used for fluid-structure interaction problems in [71], and for free-surface problems in [61,95], for example, is based upon Scriven's method of spines [82,121]. Here the mesh movement is constrained by a number of parameters (one for each spine used), thus reducing the computational overhead in exchange for restricting the generality of the approach. More general ALE forms have also been widely used for free-surface problems, based upon maintaining mesh

quality [115,116], Laplacian smoothing [129] or pseudo-solid deformation [33,144]. Other applications which have benefited from successful ALE algorithms include phase-change problems [123], viscous sintering [84, 94, 151] and the interaction of free surfaces with solid boundaries [4, 133, 145].

Note that not all ALE methods need to be applied to, or driven by, fluid flow problems. Indeed, the latter part of this paper is devoted to a more general family of methods. Before introducing this approach, however, we discuss some other recent techniques that have been used for generating mesh velocities.

3.3 Moving finite elements

Where there is no specific physical motivation for assigning a velocity to each node of the finite element mesh, (unlike Lagrangian-based methods for equations of fluid flow, for example), some other mechanism for determining an appropriate mesh velocity field is required. A well-known approach is the moving finite element (MFE) method of Miller and Miller [104] and Miller [100]. This technique is a natural extension of the classical method of lines (MOL) scheme for the Galerkin finite element method on a fixed grid. On a fixed mesh, for a piecewise trial solution of the form

$$U(\mathbf{x},t)=\sum U_j(t)W_j(\mathbf{x}), \tag{3.4}$$

where the $W_j(\mathbf{x})$ are test functions, the MOL seeks to minimize the L_2 norm of the residual

$$U_t-\mathcal{L}U=\sum\frac{dU_j}{dt}W_j(\mathbf{x})-\mathcal{L}U \tag{3.5}$$

of (2.1) with respect to each unknown, dU_i/dt, resulting in a system of ordinary differential equations (ODEs) for each U_i.

In the MFE method this idea is generalized using piecewise trial solutions of the form

$$\widehat{U}(t)=U(\widehat{\mathbf{x}}(t),t)=\sum\widehat{U}_j(t)W_j(\widehat{\mathbf{x}}(t)), \tag{3.6}$$

where $\widehat{\mathbf{x}}(t)$ is the piecewise linear interpolant of the mesh vertex locations, $\{\widehat{\mathbf{X}}_i(t)\}$, say, and the $W_j(\mathbf{x})$ are piecewise linear basis functions. The method then seeks to minimize the L_2 norm of the residual (3.5), but this time with respect to the velocity of each node of the mesh, $d\widehat{\mathbf{X}}_i/dt=\mathbf{V}_i$ say, as well as $d\widehat{U}_i/dt$ at that node. The term $d\widehat{U}/dt$ is obtained by differentiating (3.6) by the chain rule to obtain

$$\frac{d\widehat{U}}{dt}=\sum\frac{d\widehat{U}_j}{dt}W_j(\widehat{\mathbf{x}}(t))+\sum\widehat{U}_j(t)\frac{dW_j(\widehat{\mathbf{x}}(t))}{dt}$$

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

$$
\begin{aligned}
&= \sum \frac{d\widehat{U}_j}{dt} W_j(\widehat{\mathbf{x}}(t)) + \sum \widehat{U}_j(t) \sum \frac{dW_j}{d\widehat{\mathbf{X}}_i} \cdot \frac{d\widehat{\mathbf{X}}_i}{dt} \\
&= \sum \frac{d\widehat{U}_j}{dt} W_j(\widehat{\mathbf{x}}(t)) + \sum \frac{d\widehat{U}}{d\widehat{\mathbf{X}}_j} \cdot \frac{d\widehat{\mathbf{X}}_j}{dt} \\
&= \sum \frac{d\widehat{U}_j}{dt} W_j(\widehat{\mathbf{x}}(t)) - \sum \mathbf{V}_j \cdot (\nabla U W_j(\widehat{\mathbf{x}}(t)))
\end{aligned} \tag{3.7}
$$

(with the last equality easily proved for piecewise linear elements, and proved for all Lagrange basis functions in [79]).

Minimization of the weighted L_2 norm of the residual

$$
U_t - \mathcal{L}U = \sum \frac{d\widehat{U}_j}{dt} W_j(\widehat{\mathbf{x}}(t)) - \sum \mathbf{V}_j \cdot (\nabla U W_j(\widehat{\mathbf{x}}(t))) - \mathcal{L}U \tag{3.8}
$$

(weighted by the positive function w) over $d\widehat{U}_j/dt$ *and* $\mathbf{V}_j$ results in a pair of equations for each node which has a similar structure to (2.36) above:

$$
\int_{\mathcal{R}(t)} W_i \left\{ \frac{dU}{dt} - \mathbf{V} \cdot \nabla U \right\} w d\mathbf{x} = \int_{\mathcal{R}(t)} W_i \mathcal{L}U w d\mathbf{x} \tag{3.9}
$$

and

$$
\int_{\mathcal{R}(t)} (\nabla U W_i) \left\{ \frac{dU}{dt} - \mathbf{V} \cdot \nabla U \right\} w d\mathbf{x} = \int_{\mathcal{R}(t)} (\nabla U W_i) \mathcal{L}U w d\mathbf{x}. \tag{3.10}
$$

The equations are scale-invariant when the original problem is scale-invariant.

As with the MOL approach, the resulting ODEs must be integrated forward in time from the initial state, in this case including the initial mesh as well as a representation of the initial data on this mesh. This reveals both the mesh and the corresponding solution at each subsequent time. Note that Eqs. (3.9) and (3.10) may be alternatively derived through a direct minimization of the L_2 norm of the residual of (2.29) with respect to $d\widehat{U}_i/dt$ and $\mathbf{V}_i$ for each node, i, of the mesh.

In principle the method is optimal in the sense that it concurrently minimizes the L_2 residual with respect to both dU_i/dt and dX_i/dt (see also [76–78] for other optimality results). However, the formulation turns out to be degenerate when a component of ∇U is continuous at any node of the mesh. This phenomenon (referred to by Miller as "parallelism"), as well as the possibility of mesh tangling (where the measure of one or more elements of the mesh becomes, or passes through, zero), is controlled by adding penalty functions to the minimization, as illustrated in [62, 100]. An unfortunate consequence of this penalization approach is to cause the resulting semi-discrete ODE system to become very stiff, so that suitable stiff integrators for such systems need to be employed for computational efficiency.

As shown in [5], for first order nonlinear PDEs the MFE method is a discretization of Charpit's method of characteristics for nonlinear PDEs (see e.g., [59]), while in the case of second order diffusion equations the MFE velocities are driven by truncation errors.

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

In the original MFE formulation the weight function w in (3.9) and (3.10) is equal to unity, i.e., $w \equiv 1$. An important alternative, relating to arclength, is to take $w = 1/(1+|\nabla u|^2)$ and carry out the integration over arclength, which is known as the gradient-weighted form of MFE (GWMFE) and is described in [37,38]. The further generalization of this approach to systems of PDEs is considered in [101] and includes a so called "string gradient-weighted" formulation (SGWMFE) which is discussed in more detail in [143]. Other variants of the method include a Least Squares MFE method, [103], and a stabilized MFE method, [102], which are both designed to control mesh movement for steady-state convection-dominated problems, which can cause significant difficulty for the standard MFE and GWMFE approaches.

In addition to the variants of the method proposed by Miller and co-workers, other alternatives have also been developed. These include the use of higher order Lagrange basis functions for both the trial and the test spaces, e.g., [39, 125, 126], and the use of higher order C^1 functions for the test space in [72]. More recently, both C^1 trial and test spaces have been considered in [55]. Each of the above techniques addresses the degeneracy of the MFE equations via a suitable penalization. An explicit approach has been studied in [11, 12] and [81], for example, primarily demonstrated for problems in one space dimension and using explicit integration in time.

There are numerous examples of the successful application of the MFE method to challenging practical problems, and in one space dimension these have typically yielded efficient and accurate numerical solutions. Examples in subsurface flow include the wetting of unsaturated soil, [46,65], solute transport problems, [66,80], and the simulation of sedimentation, e.g., [127]. Similarly, the range of successful chemical engineering applications include front reaction, heat transfer and population balance problems (e.g., [40, 54] and [55] respectively). The main difficulties with the method appear in two or more space dimensions, where the very high computational expense suggests that it will not generally be cost-effective to solve for the mesh and the solution in this fully coupled manner. It is for this reason that the Arbitrary Lagrangian Eulerian (ALE) and similar approaches are frequently more efficient.

3.4 The deformation method

This method derives from analytic results of Moser [107] and Dacorogna and Moser [49] concerning the existence of a particular class of diffeomorphism between two domains in $\Re^n$. In particular, [49] is concerned with the existence of diffeomorphisms $\boldsymbol{\phi}:\overline{\Omega}\to\overline{\Omega}$ such that

$$\det\nabla\boldsymbol{\phi}(\mathbf{x}) = f(\mathbf{x}), \qquad \mathbf{x}\in\Omega, \tag{3.11a}$$

$$\boldsymbol{\phi}(\mathbf{x}) = \mathbf{x}, \qquad \mathbf{x}\in\partial\Omega, \tag{3.11b}$$

where f has mean value one over Ω and, along with Ω, satisfies certain regularity assumptions.

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

The significance of this work stems from the fact that the existence proofs in [49] are constructive in nature, thus motivating associated numerical methods such as [86,88,89], discussed below. Specifically, Dacorogna and Moser linearize (3.11) by expanding $\boldsymbol{\phi}$ as a perturbation from the identity: $\boldsymbol{\phi}(\mathbf{x})=\mathbf{x}+\boldsymbol{\nu}(\mathbf{x})$. Neglecting higher order terms in $\mathbf{v}$ then leads to

$$\nabla\cdot\boldsymbol{\nu}(\mathbf{x})=f(\mathbf{x})-1, \qquad \mathbf{x}\in\Omega, \tag{3.12a}$$

$$\boldsymbol{\nu}(\mathbf{x})=0, \qquad \mathbf{x}\in\partial\Omega. \tag{3.12b}$$

The existence of a solution to (3.12) is proved constructively by setting $\boldsymbol{\nu}(\mathbf{x})=\nabla w(\mathbf{x})$ and finding $w(\mathbf{x})$ from

$$\nabla^2 w(\mathbf{x})=f(\mathbf{x})-1, \qquad \mathbf{x}\in\Omega, \tag{3.13a}$$

$$\nabla w(\mathbf{x})\cdot\hat{\mathbf{n}}=0, \qquad \mathbf{x}\in\partial\Omega. \tag{3.13b}$$

The final stage of the proof is to demonstrate that the required mapping $\boldsymbol{\phi}$ is given by $\boldsymbol{\phi}(\mathbf{x})=\boldsymbol{\varphi}(\mathbf{x},1)$ where $\boldsymbol{\varphi}$ satisfies the ordinary differential equation

$$\frac{d}{dt}\boldsymbol{\varphi}(\mathbf{x},t)=\frac{\boldsymbol{\nu}(\boldsymbol{\varphi}(\mathbf{x},t))}{t+(1-t)f(\boldsymbol{\varphi}(\mathbf{x},t))}, \tag{3.14a}$$

$$\boldsymbol{\varphi}(\mathbf{x},0)=\mathbf{x}. \tag{3.14b}$$

This is proved by introducing the function

$$h(\mathbf{x},t)=\det\nabla\boldsymbol{\varphi}\big[t+(1-t)f(\boldsymbol{\varphi}(\mathbf{x},t))\big] \tag{3.15}$$

and showing that $\partial h/\partial t=0$, from which it follows that $h(\mathbf{x},1)=h(\mathbf{x},0)$, as required.

As indicated above, these concepts from [49] have been used to develop an algorithm for mesh transformation by Liao and co-workers [86,88,89], who noted that $f(\mathbf{x})$ could be chosen to provide a transformation which contracts elements where f is small and expands them where it is large, forming the basis of an adaptive meshing algorithm. Liao and Su [90] then extended the ideas to allow for the tracking of time-varying volume elements. In this context, given a differentiable, strictly positive, function $f(\mathbf{x},t)$ which satisfies the normalisation property

$$\int_\Omega\Big(\frac{1}{f(\mathbf{x},0)}-1\Big)d\mathbf{x}=0, \tag{3.16}$$

the one-to-one mapping $\boldsymbol{\phi}(\mathbf{x},t)$ which guarantees equidistribution of $f(\mathbf{x},t)$, becomes

$$\det\nabla\boldsymbol{\phi}(\mathbf{x},t)=f(\boldsymbol{\phi}(\mathbf{x},t)), \quad \mathbf{x}\in\Omega,\ t>0. \tag{3.17}$$

This diffeomorphism is found by first constructing the vector field $\boldsymbol{\nu}(\mathbf{x},t)$ satisfying

$$\nabla\cdot\boldsymbol{\nu}(\mathbf{x},t)=-\frac{\partial}{\partial t}\Big(\frac{1}{f(\mathbf{x},t)}\Big), \qquad \mathbf{x}\in\Omega, \tag{3.18a}$$

$$\nabla\times\boldsymbol{\nu}(\mathbf{x},t)=0, \qquad \mathbf{x}\in\Omega, \tag{3.18b}$$

$$\boldsymbol{\nu}(\mathbf{x},t)\cdot\hat{\mathbf{n}}=0, \qquad \mathbf{x}\in\partial\Omega, \tag{3.18c}$$

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

and using it to update the system given by

$$\frac{\partial}{\partial t}\boldsymbol{\phi}(\mathbf{x},t)=\boldsymbol{\nu}(\boldsymbol{\phi},t)f(\boldsymbol{\phi},t), \qquad t>0, \tag{3.19}$$

with $\boldsymbol{\phi}(\mathbf{x},0)=\boldsymbol{\phi}_0(\mathbf{x})$. In the context of a moving mesh method, the rate of change of the transformation $\partial\boldsymbol{\phi}/\partial t$ simply represents the mesh velocity. A condition on the curl of the vector field is imposed to ensure uniqueness and imposes irrotationality, which will typically help to improve the shape of the mesh cells (and is exploited below). Bochev, Liao and de la Pena [23] show that this leads to the equidistribution of mesh points according to the weight $f(\mathbf{x},t)$ and prove that no tangling can occur in a domain where the boundary is fixed and the movement of points along the boundary does not tangle. They also present two-dimensional computational results showing the adaptation of meshes to pre-specified, time-varying functions. Semper and Liao [124] combine this with an SUPG finite element method to solve the one-dimensional time-dependent advection-diffusion equation on a fixed domain, using discontinuities in time to allow for the mesh movement.

Since then Liu, Ji and Liao [92] have applied the steady-state algorithm to solutions of the two-dimensional Euler equations, fitting to shocks in aerofoil flows. Further, [32] combines the algorithm with a least-squares finite element approach to solving the div-curl system (3.18) directly for $\boldsymbol{\nu}(\mathbf{x},t)$, and apply the resulting algorithm to problems where the boundary deforms. The algorithm has also been applied in three space dimensions [91], to track moving shock waves [87], and in image registration [41]. Delzanno et al. [51] however note some situations where the quality of the grids obtained compares unfavourably with a Monge-Kantorovich transformation approach.

The method has recently been generalised by Grajewski, Köster and Turek [67] to provide a non-tangling mapping from a non-uniform initial mesh which allows the transformation in time-dependent cases to be calculated directly from the mesh at the previous time level, rather than relative to an initial uniform mesh. Wan and Turek [146] have used this approach to model two-dimensional particulate flows in a viscous fluid using an ALE formulation.

3.5 The GCL method

The Space Conservation Law (SCL) was originally formulated by Trulio and Trigger [139] as an additional equation which should be approximated in the simulation of fluid flow on moving meshes along with conservation of mass, momentum and energy. It was later recognised by Thomas and Lombard [135], who termed it the Geometric Conservation Law (GCL), as a constraint which should be imposed on numerical approximations of conservation laws on moving meshes. In particular, any numerical scheme applied on a moving mesh should reduce to the GCL when the solution is constant: the movement of the mesh should not create or destroy "space".

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

Given a spatial region Ω with the closed boundary $\partial\Omega$ moving with velocity $\mathbf{v}$, the GCL takes the form

$$\frac{d}{dt}\int_{\Omega(t)} d\mathbf{x} = \int_{\partial\Omega} \mathbf{v}\cdot\hat{\mathbf{n}}\, d\Gamma, \tag{3.20}$$

(which is simply the Reynolds Transport Theorem (2.22) with constant u), or the equivalent differential form

$$\frac{\partial J}{\partial t} = \nabla\cdot(J\dot{\mathbf{x}}), \tag{3.21}$$

in which J is the determinant of the Jacobian of the transformation between the fixed reference domain and the physical domain. The GCL has been widely used as a constraint to be satisfied by numerical schemes applied in a moving frame of reference, and in particular influences the implementation of the conservative ALE schemes described in Section 3.2.

The desire to satisfy a GCL favours the finite volume framework [52, 53, 57, 69, 96], which provides the most natural formulation of the GCL due to its inherent conservation properties. However, it has also been applied with finite differences [135], conforming finite elements [24, 56, 58], discontinuous Galerkin methods [114] (space-time FE and DG methods automatically satisfy the GCL [141]), and residual distribution schemes [99]. The paper of Étienne, Garon and Pelletier [56] includes a useful summary of the literature in this area and highlights three increasingly challenging levels of compliance with the GCL within a formulation of a moving mesh algorithm.

1. It must reproduce the exact solution to the constant, no-flow test case on an arbitrary deforming mesh [52].
2. It must reproduce the exact solution of a uniform flow on moving grids: Farhat, Geuzaine and Grandmont [57] referred to this as the Discrete GCL (DGCL).
3. The time-stepping scheme must exhibit the same rate of convergence on both fixed and moving meshes. Guillard and Farhat [69] show that satisfying the DGCL is a sufficient condition for the scheme to be at least first order accurate. No such condition has yet been proved for higher orders of accuracy; in fact, even though Mavriplis and Yang [96] and Geuzaine, Grandmont and Farhat [63] are both able to derive schemes which satisfy the GCL and demonstrate high order temporal accuracy, they suggest that satisfaction of the GCL is unrelated to the retention of time-accuracy by the resulting numerical scheme.

Farhat, Geuzaine and Grandmont [57] suggest that satisfaction of the DGCL is a necessary and sufficient condition for a numerical scheme to preserve nonlinear stability properties. This is, however, disputed by Boffi and Gastaldi [24] and Formaggia and Nobile [58] who demonstrate that this is only the case for backward Euler time-stepping.

Cao, Huang and Russell [34] noted that the GCL could also be used to drive a mesh movement algorithm. They considered the situation with a positive monitor function

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

$m(\mathbf{x},t)$ and required that

$$\nabla\cdot\mathbf{v}=-\frac{1}{m}\frac{Dm}{Dt}. \tag{3.22}$$

Combining this with the differential form of the GCL (3.21) gives

$$\frac{D}{Dt}(mJ)=0\Rightarrow mJ=\text{constant}, \tag{3.23}$$

a multidimensional generalisation of the one-dimensional equidistribution principle. This leads to

$$\frac{\partial m}{\partial t}+\nabla\cdot(m\mathbf{v})=0, \tag{3.24}$$

a condition on the divergence of the velocity field which, as in the case of the deformation map approach, can be supplemented by a condition on its curl, given in [34] by

$$\nabla\times\omega(\mathbf{v}-\mathbf{u})=0, \tag{3.25}$$

for some choice of $\mathbf{u}(\mathbf{x},t)$ and $\omega>0$. They then choose to find the mesh velocity field by minimisation of the functional

$$\mathbf{I}=\frac{1}{2}\int_\Omega\left|\frac{\partial m}{\partial t}+\nabla\cdot(m\mathbf{v})\right|^2+\left(\frac{m}{\omega}\right)^2\left|\nabla\times\omega(\mathbf{v}-\mathbf{u})\right|^2 d\mathbf{x}, \tag{3.26}$$

since the divergence and curl must be orthogonal in L_2. This is combined with the boundary condition $\mathbf{v}\cdot\hat{\mathbf{n}}=0$ on $\partial\Omega$ and computational results are presented which adapt the mesh to pre-specified time-dependent functions on domains with fixed boundaries. A smooth solution exists as long as the data and boundary are smooth. The mapping is non-singular but the mesh can become highly skewed.

Cao, Huang and Russell [34,35] go on to note the relationship between their method and the deformation map approach. Given that $\boldsymbol{\nu}=m\mathbf{v}$ with $m=1/f$, Eqs. (3.18) become

$$\nabla\cdot(m\mathbf{v})=-\frac{\partial m}{\partial t}, \qquad \mathbf{x}\in\Omega, \tag{3.27a}$$

$$\nabla\times(m\mathbf{v})=0, \qquad \mathbf{x}\in\Omega, \tag{3.27b}$$

$$\rho\mathbf{v}\cdot\mathbf{n}=0, \qquad \mathbf{x}\in\partial\Omega, \tag{3.27c}$$

which is simply a special case of the GCL method with $\mathbf{u}=\mathbf{0}$ and $\omega=m$.

3.6 The conservation method

The conservation method of Baines, Hubbard and Jimack [7–9] is built on the same foundations as the GCL method of Cao, Huang and Russell [34]. Instead of recasting the GCL as the minimization problem (3.26) for the purpose of discretization, this method uses the conservative (integral) form to find the discrete velocities directly.

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

The method can be summarized as follows. The PDE is used in conjunction with an Eulerian form of a conservation equation to generate a velocity, which is then used to move the mesh in a Lagrangian manner. In simple cases the solution of the PDE can be recovered *a posteriori* from the Lagrangian form of the same conservation law. As can be seen in the following sections, this strategy forms a good basis for algorithmic development. The fundamentals of the method and the details of the multidimensional finite element implementation are set out in the following three sections. Section 4 deals with nonlinear PDE problems which conserve mass, Section 5 extends the idea to non mass-conserving problems, and Section 6 contains a number of further extensions and generalizations.

4 The conservation method for mass-conserving problems

We now describe in detail the conservation method of the previous subsection for constructing velocities, developed from [20, 21] and [34], which is based on local or distributed mass conservation, and which exploits the equivalence between the Eulerian and Lagrangian conservation laws described in Section 3.1. In the remainder of this section we shall consider the consequences of local conservation in the case of mass-conserving problems, and then use the idea to motivate a moving-mesh finite element method driven by this property.

4.1 A local conservation principle

Let $u>0$ be a solution of the PDE (2.1) in the interior of a moving domain $\mathcal{R}(t)$ and suppose that the boundary conditions are such that total mass,

$$\theta=\int_{\mathcal{R}(t)} u d\mathbf{x}, \tag{4.1}$$

is conserved, i.e., independent of time. The PME of Subsection 2.2.1 with a zero Dirichlet boundary condition falls into this category.

The velocity is defined implicitly by the conservation principle

$$\int_{\Omega(t)} u d\mathbf{x}=c(\Omega), \quad \text{independent of time}, \tag{4.2}$$

for an arbitrary moving subregion $\Omega(t)$ of $\mathcal{R}(t)$. (Assuming that the subregions $\Omega(t)$ form a non-overlapping covering of $\mathcal{R}(t)$ Eq. (4.2) is consistent with the total mass θ in (4.1) being independent of time.)

One motivation for the choice of (4.2) as the mechanism for generating the velocity is a converse of the invariance property of self-similar solutions (2.16) mentioned in Subsection 2.2. This converse, which is proved in Appendix A, states that if (4.2) holds then under certain conditions initial data that coincides with a self-similar solution is propagated as a self-similar solution for all time.

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

Differentiating (4.2) with respect to t and using the Reynolds Transport Theorem (2.22),

$$\frac{d}{dt}\int_{\Omega(t)} u d\mathbf{x} = 0 = \int_{\Omega(t)} u_t d\mathbf{x} + \oint_{\partial\Omega(t)} u\mathbf{v}\cdot\hat{\mathbf{n}} d\Gamma, \tag{4.3}$$

where $\hat{\mathbf{n}}$ is the outward unit normal to the boundary $\partial\Omega(t)$ and $\mathbf{v}$ is the velocity of points on the boundary. Then, using the Divergence Theorem, (4.3) can be written as

$$\int_{\Omega(t)} \{u_t + \nabla\cdot(u\mathbf{v})\} d\mathbf{x} = 0, \tag{4.4}$$

where $\mathbf{v}$ is any sufficiently smooth velocity field consistent with the normal component of the boundary velocity. Since $\Omega(t)$ is arbitrary it follows that u and $\mathbf{v}$ satisfy the pointwise Eulerian conservation law

$$u_t + \nabla\cdot(u\mathbf{v}) = 0. \tag{4.5}$$

We note the equivalence of the two conservation principles (4.2) and (4.5) (Lagrangian and Eulerian) through the Reynolds Transport Theorem (2.22), as noted previously in Subsection 3.1.

Since u satisfies the PDE (2.1) in $\mathcal{R}(t)$, Eq. (4.5) can be written as the purely spatial equation

$$\mathcal{L}u + \nabla\cdot(u\mathbf{v}) = 0, \tag{4.6}$$

which, given u, can be regarded as an equation for the velocity $\mathbf{v}$.

4.1.1 Solving for the velocity

In general there is no unique solution of (4.6) for $\mathbf{v}$. But if, as suggested in [34], there exists a velocity $\mathbf{q}$ such that $curl\,\mathbf{v} = curl\,\mathbf{q}$ then there is also a velocity potential ϕ such that

$$\mathbf{v} = \mathbf{q} + \nabla\phi \tag{4.7}$$

and Eq. (4.6) can be written

$$-\nabla\cdot(u\nabla\phi) = \mathcal{L}u + \nabla\cdot(u\mathbf{q}). \tag{4.8}$$

Given $u > 0$ in the interior of $\mathcal{R}(t)$, by Helmholtz' Theorem [68] there is a unique solution of (4.8) for ϕ (at least to the extent of a constant) in the region $\mathcal{R}(t)$, provided that either $u\partial\phi/\partial n$ or ϕ is given on the boundary $\partial\mathcal{R}(t)$. The boundary conditions on $u\partial\phi/\partial n$ or ϕ correspond respectively to either the normal component of the flux or the tangential component of the velocity being prescribed on the boundary.

The velocity $\mathbf{v}$ is then determined by (4.7). The evolution of points $\hat{\mathbf{x}}(t)$ of the region $\mathcal{R}(t)$ may be found by integrating the ODE system (1.1) for the Lagrangian coordinate $\hat{\mathbf{x}}(t)$ which coincides instantaneously with the fixed coordinate $\mathbf{x}$ at an initial time.

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

4.1.2 Recovering the solution

The solution $u(t,\mathbf{x})$ in the moving frame can in principle be recovered *a posteriori* from the local mass conservation equation (4.2). Writing (4.2) in the form

$$\int_{\Omega(t)} u d\mathbf{x} = c(\Omega) = \overline{u}(t)\,|\Omega(t)|, \tag{4.9}$$

where $|\Omega(t)|>0$ is the measure of $\Omega(t)$ and $\overline{u}(t)$ is a mean value of $u(t,\mathbf{x})$ in $\Omega(t)$, we may divide by $|\Omega(t)|$ and obtain u as the limit

$$u = \lim_{|\Omega(t)|\to 0} \frac{c(\Omega)}{|\Omega(t)|} \tag{4.10}$$

provided that the limit exists and is unique. Since $c(\Omega)$ is independent of time it is equal to its initial value $c(\Omega_0)$, so that from (4.9)

$$c(\Omega) = c(\Omega_0) = \overline{u}(t_0)\,|\Omega(t_0)|, \tag{4.11}$$

and the solution u may be obtained formally from (4.10) and (4.11) as

$$u(t,\widehat{\mathbf{x}}(t)) = \Big(\lim_{|\Omega(t)|\to 0} \frac{|\Omega(t_0)|}{|\Omega(t)|}\Big) u(t_0,\widehat{\mathbf{x}}(t_0)). \tag{4.12}$$

To summarize the structure of the conservation method, the Eulerian form (4.5) of the conservation law (4.2) is used in conjunction with the PDE (2.1) to obtain the spatial equation (4.6) for the velocity $\mathbf{v}$. Then, assuming a prescribed vorticity of the form $curl\,\mathbf{q}$ (which may be zero), there exists a velocity potential ϕ satisfying (4.8). The elliptic equation (4.8) is solved for ϕ, with either ϕ or $u\partial\phi/\partial n$ prescribed on the boundary. The velocity $\mathbf{v}$ is then found from (4.7) and used to move the points of the domain by (1.1). Finally, the solution u is recovered *a posteriori* from the Lagrangian conservation law (4.2) in the form (4.12).

The method is scale-invariant when the original problem is scale-invariant because it relies only on the original PDE and the conservation principle (4.2), both of which are scale-invariant.

4.1.3 Some exact solutions for the velocity

For a class of PDE problems the elliptic equation (4.8) has an exact solution for the potential ϕ, and hence for the velocity $\mathbf{v}$. Consider nonlinear PDEs (2.1) of the form

$$u_t = \nabla\cdot(u\nabla p) \tag{4.13}$$

in a moving domain $\mathcal{R}(t)$ for a general function p (which may depend on u and/or its space derivatives), with the zero flux boundary condition

$$u\Big(\mathbf{v}\cdot\hat{\mathbf{n}} + \frac{\partial p}{\partial n}\Big) = 0 \tag{4.14}$$

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

(the porous medium equation of Subsection 2.2.1, for which $p \equiv u$, with $u = 0$ on the moving boundary, is such a problem). We note incidentally from (2.22) that

$$\frac{d}{dt}\int_{\mathcal{R}(t)} u d\mathbf{x} = \int_{\mathcal{R}(t)} \{u_t + \nabla\cdot(u\mathbf{v})\} d\mathbf{x} = \int_{\mathcal{R}(t)} \{\nabla\cdot(u\nabla p + u\mathbf{v})\} d\mathbf{x}$$
$$= \oint_{\partial\mathcal{R}(t)} u\left(\frac{\partial p}{\partial n} + \mathbf{v}\cdot\hat{\mathbf{n}}\right) d\Gamma = 0, \tag{4.15}$$

by (4.14), showing that the total mass is conserved for these problems.

Suppose now that u is a positive solution of (4.13) in the interior of $\mathcal{R}(t)$ and that the velocity field $\mathbf{v}$ is generated by (4.8) and (4.7) with $\mathbf{q} = 0$. Then some exact solutions for ϕ and $\mathbf{v}$ may be derived. Since $\mathbf{q} = 0$ in (4.7), $\mathbf{v} = \nabla\phi$ (the velocity is irrotational), hence for Eq. (4.13) we may write (4.8) in the form

$$\nabla\cdot\{u\nabla(\phi + p)\} = 0, \quad \text{in } \mathcal{R}(t), \tag{4.16}$$

with the boundary condition (4.14) as

$$u\left(\mathbf{v}\cdot\hat{\mathbf{n}} + \frac{\partial p}{\partial n}\right) = u\frac{\partial}{\partial n}(\phi + p) = 0, \quad \text{on } \partial\mathcal{R}(t). \tag{4.17}$$

Since $u > 0$ in the interior of $\mathcal{R}(t)$, the elliptic equation (4.16) for $\phi + p$ with the boundary condition (4.17) has the unique solution $\phi + p$=constant. Hence $\phi = -p$ is an exact solution for the velocity potential to within a constant and the corresponding exact solution for $\mathbf{v}$ is

$$\mathbf{v} = \nabla\phi = -\nabla p. \tag{4.18}$$

As an example, if

$$p = (-1)^k \Delta^k u \quad (k \geq 0), \tag{4.19}$$

for which Eq. (4.13) is the high order nonlinear diffusion equation

$$u_t = (-1)^k \nabla\cdot(u\nabla\Delta^k u) \tag{4.20}$$

with k positive, and if u,$\mathbf{v}$ satisfy the boundary condition

$$u\left(\mathbf{v}\cdot\hat{\mathbf{n}} + (-1)^k \frac{\partial}{\partial n}\Delta^k u\right) = 0 \tag{4.21}$$

(for example if $u = 0$ there), then

$$\mathbf{v} = (-1)^{k+1}\nabla\Delta^k u. \tag{4.22}$$

Another choice of p is [117]

$$p(u) = \int^u \frac{D(\bar{u})}{\bar{u}} d\bar{u} \tag{4.23}$$

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

for a general function $D(u)$ (assuming that the integral exists), for which

$$u\nabla p(u)=D(u)\nabla u, \tag{4.24}$$

and Eq. (4.13) is the second order nonlinear diffusion equation

$$u_t=\nabla\cdot(D(u)\nabla u). \tag{4.25}$$

Then, if u,$\mathbf{v}$ satisfy the boundary condition

$$u\mathbf{v}\cdot\hat{\mathbf{n}}+D(u)\frac{\partial u}{\partial n}=0 \tag{4.26}$$

(for example if $u=D(u)=0$), there exists the exact solution

$$\mathbf{v}=-\nabla p=-\frac{D(u)}{u}\nabla u \tag{4.27}$$

(provided that $u\neq 0$, by (4.24)).

The porous medium equation of Subsection 2.2.1,

$$u_t=\nabla\cdot(u^n\nabla u) \tag{4.28}$$

(cf. (2.17)) with $n\geq 1$, which is of the form (4.25) with $D(u)=u^n$, together with the boundary condition

$$u\left(\mathbf{v}+u^{n-1}\nabla u\right)\cdot\hat{\mathbf{n}}=0, \tag{4.29}$$

(satisfied in particular by $u=0$), has the exact solution

$$\mathbf{v}=-u^{n-1}\nabla u=-\frac{1}{n}\nabla u^n. \tag{4.30}$$

4.2 Distributed mass-conserving velocities

In preparation for the moving mesh finite element formulation described in the next section we now construct a weak form of the conservation method using a weighted conservation principle (cf. Subsection 2.4).

Let w_i, $(i=1,\cdots,N)$, be a set of positive square-integrable weight functions forming a partition of unity and moving with the points of the domain $\mathcal{R}(t)$. Define the distributed conservation principle to be

$$\int_{\mathcal{R}(t)} w_i u d\mathbf{x}=c_i, \quad \text{independent of time,} \tag{4.31}$$

(a partition of (4.1)) where u is a positive solution of the weak form

$$\int_{\mathcal{R}(t)} u_t d\mathbf{x}=\int_{\mathcal{R}(t)} \mathcal{L}u d\mathbf{x} \tag{4.32}$$

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

of the PDE (2.1) in the interior of the domain $\mathcal{R}(t)$. By summing (4.31) over i the c_i satisfy

$$\sum_1^N c_i = \int_{\mathcal{R}(t)} u d\mathbf{x} = \Big(\int_{\mathcal{R}(0)} u d\mathbf{x} \Big) \Big|_{t=0}, \tag{4.33}$$

since the w_i's form a partition of unity and mass is conserved. (We note in passing that the local conservation principle (4.2) corresponds to choosing the w_i to be the characteristic function in $\Omega(t)$.)

One motivation for the choice of (4.31) to generate the velocity is that if the initial data coincides with the L_2 projection of a self-similar solution into the space spanned by the w_i then, as shown in Appendix B, under the similarity velocity (2.10) the solution remains the L_2 projection of the self-similar solution for all time. Although the velocity generated is only approximately the similarity velocity this is an interesting property.

From (4.31) and (2.35), assuming that $\mathcal{L}u$ exists and is square-integrable, the velocity $\mathbf{v}$ satisfies

$$\int_{\mathcal{R}(t)} w_i \{ \mathcal{L}u + \nabla \cdot (u\mathbf{v}) \} d\mathbf{x} = 0 \tag{4.34}$$

(cf. (4.6)), since the w_i move with velocity $\mathbf{v}$. Furthermore, if $w_i \in H^1\{\mathcal{R}(t)\}$ then after using Green's Theorem on (4.34) we obtain the weak form

$$\int_{\mathcal{R}(t)} u \nabla w_i \cdot \mathbf{v} d\mathbf{x} = \int_{\mathcal{R}(t)} w_i \mathcal{L}u d\mathbf{x} + \oint_{\partial\mathcal{R}(t)} w_i u \mathbf{v} \cdot \hat{\mathbf{n}} d\Gamma \tag{4.35}$$

of (4.5).

There is no unique solution of Eq. (4.34) or (4.35) for $\mathbf{v}$ in general but if there exists a square-integrable velocity $\mathbf{q}$ and velocity potential ϕ such that

$$\int_{\mathcal{R}(t)} u \nabla w_i \cdot \mathbf{v} d\mathbf{x} = \int_{\mathcal{R}(t)} u \nabla w_i \cdot (\mathbf{q} + \nabla\phi) d\mathbf{x} \tag{4.36}$$

(a weak form of (4.7)) then Eq. (4.35) can be written

$$\int_{\mathcal{R}(t)} u \nabla w_i \cdot \nabla\phi d\mathbf{x} = \int_{\mathcal{R}(t)} (w_i \mathcal{L}u - u \nabla w_i \cdot \mathbf{q}) d\mathbf{x} + \oint_{\partial\mathcal{R}(t)} w_i u \mathbf{v} \cdot \hat{\mathbf{n}} d\Gamma. \tag{4.37}$$

Since $u > 0$ in the interior of $\mathcal{R}(t)$, then if $w_i u \mathbf{v} \cdot \hat{\mathbf{n}}$ is given on the boundary $\partial\mathcal{R}(t)$ the right-hand side of (4.37) is known and the equation possesses a unique solution for $\phi \in span\{w_i\}$ provided that ϕ is prescribed at at least one point of the domain (or to within an additive constant if it is not).

4.2.1 Weak form of the velocity and recovery of the solution

Having obtained the velocity potential ϕ the next step is to derive the velocity. From (4.36), if $\mathbf{v}, \mathbf{q} \in span\{\nabla w_i\}$ then $\mathbf{v}$ is given directly by (4.7), as before. Otherwise, a weak form is required. The weak form (4.36) connecting ϕ and $\mathbf{v}$ is insufficient by itself to

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

determine $\mathbf{v}$ from ϕ, since it is just an orthogonality condition. We therefore construct a best approximation $\widetilde{\mathbf{v}}$ to $\mathbf{v}$ obtained by projecting $\mathbf{v}-\mathbf{q}-\nabla\phi$ into the space spanned by the w_i in an L_2 sense, thus minimizing $\|\mathbf{v}-\mathbf{q}-\nabla\phi\|_2$ over each component of $\mathbf{v}$ and leading to the normal equations

$$\int_{\mathcal{R}(t)} w_i(\widetilde{\mathbf{v}}-\mathbf{q}-\nabla\phi)d\mathbf{x}=0 \tag{4.38}$$

for $\widetilde{\mathbf{v}}$. Note that in the case of the velocity $\widetilde{\mathbf{v}}$ the rate of change of the integral in (4.31) is

$$\frac{d}{dt}\int_{\mathcal{R}(t)} w_i u\,d\mathbf{x}=\int_{\mathcal{R}(t)} u\nabla w_i\cdot(\widetilde{\mathbf{v}}-\mathbf{q}-\nabla\phi)\,d\mathbf{x}=\int_{\mathcal{R}(t)} u\nabla w_i\cdot(\widetilde{\mathbf{v}}-\mathbf{v})\,d\mathbf{x} \tag{4.39}$$

which is different from zero, so that (4.31) is no longer independent of time. Nevertheless, the right-hand side of (4.39) is likely to be small, as can be seen from the inequality

$$\left\|\int_{\mathcal{R}(t)} u\nabla w_i\cdot(\widetilde{\mathbf{v}}-\mathbf{q}-\nabla\phi)d\mathbf{x}\right\|_2\le\|u\|_2\cdot\|\widetilde{\mathbf{v}}-\mathbf{q}-\nabla\phi\|_2\cdot\left\|\int_{\mathcal{R}(t)}\nabla w_i d\mathbf{x}\right\|_2, \tag{4.40}$$

provided that $\|u\|_2$ is bounded, since $\|\widetilde{\mathbf{v}}-\mathbf{q}-\nabla\phi\|_2$ is least. Hence the c_i in (4.31) are expected to vary slowly with time.

The evolution of points of the domain can be obtained by integrating (1.1) in time with $\mathbf{v}$ replaced by $\widetilde{\mathbf{v}}$, and the solution u approximated by inverting (4.31) on the new domain. The solution u may alternatively be found exactly by evaluating the right-hand side of (4.39), integrating in time to give

$$\int_{\mathcal{R}(t)} w_i u\,d\mathbf{x}=\mu_i \tag{4.41}$$

say, and then inverting

$$\int_{\mathcal{R}(t)} w_i u d\mathbf{x}=\mu_i \tag{4.42}$$

for u, where μ_i (acting in lieu of the c_i in (4.31)) is updated at each time-step.

An alternative is to use the conservative ALE equation (2.35) in the form

$$\dot{\mu}_i=\frac{d}{dt}\int_{\mathcal{R}(t)} w_i u d\mathbf{x}=\int_{\mathcal{R}(t)} w_i\{\mathcal{L}u+\nabla\cdot(u\widetilde{\mathbf{v}})\}d\mathbf{x}, \tag{4.43}$$

which also leads to a system of the form (4.42) after integration in time. Using (4.34) it can be shown that (4.43) is equivalent to (4.39) under some circumstances. However, in general (4.43) will produce a slightly different u since the right-hand side of (4.42) will have been propagated differently in time.

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

4.3 A moving mesh finite element method

We now use the distributed conservation principle (4.31) together with the weak forms (4.37) and (4.38) to generate a multidimensional moving mesh finite element method. For the mesh we assume a moving polygonal or polyhedral approximation $\partial\overline{\mathcal{R}}(t)$ to $\partial\mathcal{R}(t)$ together with a moving tessellation of simplices of $\overline{\mathcal{R}}(t)$. At any instant t let $\mathcal{S}^1\{\overline{\mathcal{R}}(t)\}\subset H^1\{\overline{\mathcal{R}}(t)\}$ be the space spanned by standard piecewise linear basis functions $W_i(t,\mathbf{x})$ on the tessellation, forming a partition of unity. Note that the W_i are scale-invariant.

Piecewise linear approximations $U\approx u$, $\mathbf{V}\approx\tilde{\mathbf{v}}$, $\Phi\approx\phi$ and $\widehat{\mathbf{X}}\approx\hat{\mathbf{x}}$ are also assumed, lying in $\mathcal{S}^1\{\overline{\mathcal{R}}(t)\}$ (in the case of the vectors $\widehat{\mathbf{X}}$ and $\mathbf{V}$ for each Cartesian component lying in this space). The total mass

$$\int_{\overline{\mathcal{R}}(t)} U\,d\mathbf{x} \tag{4.44}$$

is partitioned and the partitions

$$\int_{\overline{\mathcal{R}}(t)} W_i U\,d\mathbf{x} \tag{4.45}$$

maintained in time to generate the velocity, as in the previous subsection. The partitions $C_i\approx c_i$ are defined by

$$\int_{\overline{\mathcal{R}}(t)} W_i U\,d\mathbf{x}=C_i,\quad \text{independent of time} \tag{4.46}$$

(cf. (4.31)). Since the W_i form a partition of unity, Eq. (4.46) is consistent with mass conservation, and $\sum_i C_i$ is constant, as in (4.33). Assuming that $U>0$ lies in the domain of the operator $\mathcal{L}$ and that $\mathcal{L}U$ is square-integrable, from (4.35) the velocity field $\mathbf{V}$ satisfies the weak form

$$\int_{\overline{\mathcal{R}}(t)} U\nabla W_i\cdot\mathbf{V}d\mathbf{x}=\int_{\overline{\mathcal{R}}(t)} W_i\mathcal{L}U d\mathbf{x}+\oint_{\partial\overline{\mathcal{R}}(t)} W_i U\mathbf{V}\cdot\hat{\mathbf{n}}d\Gamma. \tag{4.47}$$

There is no unique solution of (4.47) for $\mathbf{V}$ in general but if there exists a square-integrable velocity $\mathbf{Q}$ and a velocity potential Φ satisfying the weak form

$$\int_{\overline{\mathcal{R}}(t)} U\nabla W_i\cdot\mathbf{V}d\mathbf{x}=\int_{\overline{\mathcal{R}}(t)} U\nabla W_i\cdot(\mathbf{Q}+\nabla\Phi)d\mathbf{x} \tag{4.48}$$

(cf. (4.36)), then Eq. (4.47) can be written

$$\int_{\overline{\mathcal{R}}(t)} U\nabla W_i\cdot\nabla\Phi d\mathbf{x}=\int_{\overline{\mathcal{R}}(t)}\{W_i\mathcal{L}U-U\nabla W_i\cdot\mathbf{Q}\}d\mathbf{x}+\oint_{\partial\overline{\mathcal{R}}(t)} W_i U\mathbf{V}\cdot\hat{\mathbf{n}}d\Gamma. \tag{4.49}$$

Since $U>0$, Eq. (4.49) has a unique solution for $\Phi\in S^1\{\mathcal{R}(t)\}$ provided that $W_iU\mathbf{V}\cdot\hat{\mathbf{n}}$ is given on the boundary $\partial\overline{\mathcal{R}}(t)$ and that Φ is given at at least one point in $\overline{\mathcal{R}}(t)$ (which is no restriction since only $\nabla\Phi$ is required).

We shall henceforth restrict the argument to problems for which $W_iU\mathbf{V}\cdot\hat{\mathbf{n}}$ is given on the boundary. This includes all problems with zero Dirichlet boundary conditions on U (and those with non-zero Dirichlet boundary conditions too, as described in Section 4.3.4 below) as well as those problems for which the normal component of $\mathbf{V}$ is given on the boundary.

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

4.3.1 Matrix forms

Expanding Φ as

$$\Phi = \sum \Phi_j W_j \tag{4.50}$$

with coefficients Φ_j to be determined, Eq. (4.49) leads to the matrix equation

$$\mathcal{K}(U)\underline{\Phi} = \underline{f}, \tag{4.51}$$

where $\mathcal{K}(U)$ is a weighted stiffness matrix with entries

$$\mathcal{K}_{ij} = \int_{\overline{\mathcal{R}}(t)} U \nabla W_i \cdot \nabla W_j d\mathbf{x}. \tag{4.52}$$

The vector $\underline{\Phi}$ contains the unknown coefficients Φ_j and the vector $\underline{f}$ has entries

$$f_i = \int_{\overline{\mathcal{R}}(t)} (W_i \mathcal{L}U - U \nabla W_i \cdot \mathbf{Q})\, d\mathbf{x} + \oint_{\partial \overline{\mathcal{R}}(t)} W_i U \mathbf{V} \cdot \hat{\mathbf{n}}\, d\Gamma. \tag{4.53}$$

The system (4.51) is rank deficient as is standard (since the equations sum to zero), which is why Φ needs to be specified at at least one node. Without loss of generality we take $\Phi = 0$ at any node since only the gradient of Φ is required. Elimination of this variable from the system (4.51) gives the reduced system

$$\mathcal{K}'(U)\underline{\Phi}' = \underline{f}', \tag{4.54}$$

where the prime refers to the matrix $K(U)$ reduced by the row and column (and the vector $\underline{f}$ reduced by the entry) corresponding to assigned node. The system (4.54) is symmetric positive definite and can be solved using for example a conjugate gradient algorithm.

4.3.2 Solving for the velocity

A continuous piecewise linear velocity $\mathbf{V}$ is required to preserve the identity of the mesh, and hence $\mathbf{V}$ does not lie in the $span\{\nabla W_i\}$ which contains only discontinuous functions. Therefore, in a similar way to that of Subsection 4.2.1, we seek an approximation $\widetilde{\mathbf{V}} \in \mathcal{S}^1\{\overline{\mathcal{R}}(t)\}$ satisfying the weak form (4.38) in the form

$$\int_{\overline{\mathcal{R}}(t)} W_i(\widetilde{\mathbf{V}} - \mathbf{Q} - \nabla\Phi) d\mathbf{x} = 0. \tag{4.55}$$

Expanding $\widetilde{\mathbf{V}}$ as

$$\widetilde{\mathbf{V}} = \sum \widetilde{\mathbf{V}}_j W_j, \tag{4.56}$$

Eq. (4.55) yields the matrix equation

$$\mathcal{M}^d \underline{\widetilde{\mathbf{V}}} = \underline{\mathbf{h}}, \tag{4.57}$$

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

where the block diagonal matrix $\mathcal{M}^d$ has d blocks, each of which is a standard mass matrix with entries

$$\mathcal{M}_{ij} = \int_{\overline{\mathcal{R}}(t)} W_i W_j d\mathbf{x}. \tag{4.58}$$

The vector $\underline{\widetilde{\mathbf{V}}}$ in (4.57) contains the coefficients of the nodal velocities $\widetilde{\mathbf{V}}_i$, and the vector $\underline{\mathbf{h}}$ has entries

$$\mathbf{h}_i = \int_{\overline{\mathcal{R}}(t)} W_i(\mathbf{Q}+\nabla\Phi) d\mathbf{x}. \tag{4.59}$$

Eq. (4.57) may be solved for $\underline{\widetilde{\mathbf{V}}}$ with or without the application of boundary conditions on $\mathbf{V}$. If no boundary condition is applied the unique velocity generated is the one that minimizes the L_2 error norm $\|\mathbf{V}-\mathbf{Q}-\nabla\Phi\|_2$.

An alternative to the approximation generated by (4.55) is a local averaging approximation. Since the piecewise constant gradient $\nabla\Phi$ is most accurate at the centroids of the elements [131], a local element-weighted average can be used to advantage, giving the approximation

$$\overline{\mathbf{V}}_i = \frac{\sum_k |\Omega_k|(\mathbf{Q}+\nabla\phi)_k}{\sum_k |\Omega_k|}, \tag{4.60}$$

where $|\Omega_k|$ is the measure of element k of the patch of elements surrounding node i. Boundary conditions on $\overline{\mathbf{V}}$ can be applied as required.

A further alternative is to use this framework to take into account the flow of information under the velocity by restricting the sum in (4.60) to upwind elements, as in

$$\overline{\overline{\mathbf{V}}}_i = \frac{\sum_{k^+} \alpha_k |\Omega_k|(\mathbf{Q}+\nabla\phi)_k}{\sum_{k^+} \alpha_k |\Omega_k|}, \tag{4.61}$$

where k^+ denotes upwind cells and the α_k are consistent weights (see below). This option allows a zero condition on the velocity at a point to be applied naturally: $\overline{\overline{\mathbf{V}}}_i$ is set to zero when there are no contributions to the sum. The upwinded velocity can be effected on a cell-by-cell basis by distributing $\mathbf{Q}+\nabla\Phi$ to the vertices of the cell according to its orientation relative to the velocity field. Thus, if $\mathbf{Q}+\nabla\Phi$ has a positive component along the inward normals to two of a triangle's edges it is assigned to the unique downwind vertex, otherwise it is distributed between the two downwind vertices with appropriate weights. A simple approach considers $(\mathbf{Q}+\nabla\Phi)\cdot\mathbf{n}_\nu$, a scaled normal component of $\mathbf{Q}+\nabla\Phi$ along the inward normal to edge ν of a cell. The weights α_ν are then defined to be the positive part of $(\mathbf{Q}+\nabla\Phi)\cdot\mathbf{n}_\nu$ normalized by their sum over the three cell edges, viz

$$\alpha_\nu = \frac{\left[(\mathbf{Q}+\nabla\Phi)\cdot\mathbf{n}_\nu\right]_+}{\sum_{edges}\left[(\mathbf{Q}+\nabla\Phi)\cdot\mathbf{n}_\nu\right]_+}. \tag{4.62}$$

These coefficients correspond to those of the LDA fluctuation distribution scheme for approximating hyperbolic conservation laws (see [1,50] for details).

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

Whichever way the approximation to the velocity $\mathbf{V}$ is constructed, the displacements $\widehat{\mathbf{X}}_i(t)$ can be obtained via integration of the ODE system

$$\frac{d\widehat{\mathbf{X}}_i}{dt} = \mathbf{V}_i(t,\widehat{\mathbf{X}}) \tag{4.63}$$

in time (cf. (1.1)) using any convenient time-stepping scheme.

In spite of targeting exact mass conservation in (4.46) and (4.47), because of the approximations used, none of the above constructions for the velocity is exactly consistent with mass conservation, so that the integrals

$$\int_{\overline{\mathcal{R}}(t)} W_i U d\mathbf{x} = \mu_i, \tag{4.64}$$

say, in (4.46) typically vary slowly with time. We therefore consider an alternative way of recovering the solution U using an ALE equation, as described in the next section.

4.3.3 Recovering the solution

As discussed in Section 4.2 there are a number of possible ways in which the solution may be recovered, each of which requires the solution of a similar linear system ((4.31) or (4.42)). We focus on the discrete ALE equation (4.43) in the form

$$\dot{\mu}_i = \frac{d}{dt}\int_{\overline{\mathcal{R}}(t)} W_i U d\mathbf{x} = \int_{\overline{\mathcal{R}}(t)} W_i \{\mathcal{L}U + \nabla\cdot(U\mathbf{V})\} d\mathbf{x}, \tag{4.65}$$

where $\mu_i(t)$ is defined by (4.64). Integrating the right-hand side with respect to time we obtain values of μ_i in (4.64) which, after expanding U as

$$U = \sum U_j W_j, \tag{4.66}$$

leads to the matrix equation

$$\mathcal{M}\underline{U} = \underline{\mu}, \tag{4.67}$$

where $\underline{\mu} = \{\mu_i\}$. The matrix $\mathcal{M}$ is symmetric positive definite and can be inverted using a conjugate gradient method, for example.

Boundary conditions do not need to be imposed on U in solving the system (4.67). If no boundary conditions are imposed the solution approximates the true solution weakly through its weak imposition in the boundary integral in (4.53). The application of *strong* Dirichlet boundary conditions on U is inconsistent with the assumption of total mass conservation (as we shall see below) and requires a separate investigation, summarized in the following subsection.

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

4.3.4 Mass-conserving test functions

In applying strong Dirichlet boundary conditions to U in the system (4.67) the test space is reduced in size and the set of W_i's no longer form a partition of unity. As a result the conservation principle (4.46) is inconsistent with mass conservation.

To counter this inconsistency and allow strong Dirichlet conditions to be applied to (4.67), modified test functions $\widetilde{W}_i$ can be constructed [75] which have the property that they vanish on the boundary as well as forming a partition of unity. These test functions may be obtained by modifying the original set of W_i's by adding those W_i's corresponding to nodes where strong boundary conditions are applied to interior W_i's, forming new test functions $\widetilde{W}_i$. There are many ways to do this but the simplest takes the form

$$\widetilde{W}_i = W_i + \sum_{j \in \cup \triangle_i \cap \partial \overline{\mathcal{R}}(t)} \frac{1}{N_j} W_j \tag{4.68}$$

in which $\cup \triangle_i \cap \partial \overline{\mathcal{R}}(t)$ is the set of mesh nodes in the intersection of the closure of the support of W_i and the boundary of the domain, and N_j is the number of interior nodes adjacent to the boundary node j. Details can be found in [75].

The potential Φ and velocity $\mathbf{V}$ are obtained exactly as before, using the standard W_i test functions, but now the solution $\widetilde{U}$ (which satisfies the strong Dirichlet conditions) is found from a modified form of Eq. (4.67),

$$\widetilde{\mathcal{M}} \underline{\widetilde{U}} = \underline{\widetilde{\mu}}, \tag{4.69}$$

with the strong boundary conditions imposed, where the modified (unsymmetric) mass matrix $\widetilde{\mathcal{M}}$ takes the form

$$\widetilde{\mathcal{M}}_{ij} = \int_{\overline{\mathcal{R}}(t)} \widetilde{W}_i W_j d\mathbf{x}, \tag{4.70}$$

and

$$\widetilde{\mu}_i = \int_{\overline{\mathcal{R}}(t)} \widetilde{W}_i \widetilde{U} d\mathbf{x}. \tag{4.71}$$

The vector $\underline{\widetilde{U}}$ in (4.69) is a vector of the unknowns $\widetilde{U}_i$ in the expansion of $\widetilde{U}$ in terms of the W_j. The right-hand side of (4.69) can be found either by keeping the μ_i of (4.71) constant or, more accurately, as in the previous section, updating the μ_i using the modified ALE equation

$$\dot{\widetilde{\mu}}_i = \frac{d}{dt} \int_{\overline{\mathcal{R}}(t)} \widetilde{W}_i \widetilde{U} d\mathbf{x} = \int_{\overline{\mathcal{R}}(t)} \widetilde{W}_i \{ \mathcal{L} \widetilde{U} + \nabla \cdot (\widetilde{U} \mathbf{V}) \} d\mathbf{x}. \tag{4.72}$$

Note that the mesh velocities $\mathbf{V}$ in (4.72) are calculated using the standard linear test and trial functions. This means that the velocity potentials are calculated from (4.49) based on the proportions of the total mass associated with the test functions W_i (see (4.46)) and are different to those used in the direct recovery of U via (4.71).

We can now state the algorithm for the moving mesh finite element conservation method.

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

4.3.5 Algorithm for mass-conserving problems

Given a mesh with nodes $\{\mathbf{X}_i^0\}$ and nodal values $\{U_i^0\}$ representing the initial conditions, at each time-step:

1. Find Φ_i from (4.54).
2. Find $\widetilde{\mathbf{V}}_i$ from (4.57).
3. Advance $\widehat{\mathbf{X}}_i(t)$ in time using any convenient time-stepping scheme applied to (4.63) with $\mathbf{V}=\widetilde{\mathbf{V}}$.
4. Recover U_i at the new time-step, either from (4.67) with μ_i given by (4.65) and (4.64) (in the case of weak implementation of Dirichlet boundary conditions on U), or from (4.69) with $\widetilde{\mu}_i$ given by integrating (4.72) and (4.71) (for strong implementation of Dirichlet boundary conditions on U).

Remarks:

• The algorithm differs from standard finite element approaches by primarily targeting the mesh velocity.

• Due to the several approximations made the Lagrangian conservation law (4.46) does not hold precisely. The C_i should therefore be recomputed at each time-step.

• Step 2 may be varied to obtain $\mathbf{V}_i$ from (4.60) or (4.61).

• There is no requirement in the algorithm for equidistribution. In principle any initial distribution is propagated in time.

• Steps 1, 2 and 4 of the algorithm are scale-invariant. This is because all the equations used, as well as the W_i and $\widetilde{W}_i$, are individually scale-invariant. However, Step 3 is not necessarily scale-invariant (but see Subsection 4.4.1 below).

4.4 Time-stepping for the ODE system

The algorithm can be thought of as solving the ODE system

$$\frac{d}{dt}\begin{pmatrix}\underline{\widehat{\mathbf{X}}}\\ \underline{\mu}\end{pmatrix}=\begin{pmatrix}\underline{\mathbf{V}}\\ \underline{\dot{\mu}}\end{pmatrix}, \tag{4.73}$$

where $\underline{\mu}=\{\mu_i\}$. The method is therefore in essence a method of lines which can be approximately solved by any convenient time-stepping scheme.

The first order Euler explicit scheme is surprisingly successful in this context although its accompanying stability condition sometimes demands impractically small time-steps. The second order Heun scheme [75] has been found to be sufficiently accurate for all the problems considered in this paper although it too is subject to inconveniently small time-steps on occasion due to stability restrictions.

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

4.4.1 Scale-invariant time-stepping

The time-stepping in the algorithm in Section 4.3.5 can be made scale-invariant by rescaling the time derivative in the ODE system (4.73) by using $\tau = t^\beta$ instead of t (cf. [28]). Then $dt = d\tau^{1/\beta} = (1/\beta)\tau^{1/\beta-1}d\tau$, yielding the transformed ODE system

$$\frac{d}{d\tau}\begin{pmatrix}\widehat{\mathbf{X}}\\ \underline{\mu}\end{pmatrix} = \beta^{-1}\tau^{1/\beta-1}\begin{pmatrix}\mathbf{V}\\ \underline{\dot{\mu}}\end{pmatrix}. \tag{4.74}$$

With constant time-steps $\Delta\tau$, standard time-stepping schemes are scale-invariant and the entire algorithm in Section 4.3.5 is scale-invariant.

4.5 The one-dimensional finite element method

In one space dimension the concept of vorticity does not exist and is not required for the purpose of imposing uniqueness on the velocity field. Therefore $\mathbf{q}\equiv\mathbf{Q}\equiv\mathbf{0}$ can be assumed in all of the previous analysis. In fact, it is no longer necessary to use a velocity potential at all: a unique velocity field can be found directly from the conservation principle (4.46).

Assume that the domain is the interval $(a(t),b(t))$ with nodes

$$a(t) = x_0(t) < x_1(t) < \cdots < x_{N+1}(t) = b(t) \tag{4.75}$$

and let $W_i(x)$ be the standard piecewise linear hat functions. From (4.47) the velocity V satisfies the weak form

$$\int_{a(t)}^{b(t)} U\frac{dW_i}{dx}V\,dx = \int_{a(t)}^{b(t)} W_i\mathcal{L}U\,dx + [W_iUV]_{a(t)}^{b(t)}. \tag{4.76}$$

Using the expansion

$$V = \sum V_jW_j, \tag{4.77}$$

Eq. (4.76) leads to the matrix equation

$$\mathcal{B}(U)\underline{V} = \underline{f}, \tag{4.78}$$

where $\underline{V}$ is the vector of coefficients V_j and the entries of $\mathcal{B}(U)$ and $\underline{f}$ are respectively

$$\mathcal{B}_{ij} = \int_{a(t)}^{b(t)} U\frac{dW_i}{dx}W_j\,dx, \qquad f_i = \int_{a(t)}^{b(t)} W_i\mathcal{L}U\,dx + [W_iUV]_{a(t)}^{b(t)}. \tag{4.79}$$

For an even number of nodes the matrix $\mathcal{B}(U)$ is singular when U is constant in space, leading to the kind of singularity akin to that in the MFE method of Section 3.3. For an odd number of nodes the system (4.78) always has a unique solution when W_iUV in the last term of (4.79) is known at $x=a(t)$ and $b(t)$ (as assumed in this work) and at least one value of V is prescribed (to anchor the velocity). For geometrically symmetric problems

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

it is simple to choose this value of V because the node at the centre of the domain is fixed. The system (4.78) can then be solved directly for $\underline{V}$. The procedure for updating the mesh node positions and recovering U on the updated mesh remains unchanged.

This method of solving (4.78) for $\underline{V}$ (as opposed to solving for a velocity potential first) replaces two symmetric systems with one unsymmetric system of similar size. Mesh convergence studies suggest that the results are of similar quality. However, the difficulty, in general, of picking a point at which to apply a known velocity makes the use of a velocity potential, which along with (4.55) calculates all values of V automatically, more attractive.

The one-dimensional method is readily extended to radially symmetric problems.

4.6 Initial data

As pointed out earlier there is no requirement for the mesh to be equidistributed, which raises the question of the distribution of the initial mesh. Since the aim of the algorithms is to preserve local or distributed mass it is important to set up an initial mesh with the properties that it is desired to keep.

4.6.1 Initial data in one dimension

The simplest initial data consists of nodes equally spaced between the endpoints with initial values obtained by sampling the initial condition at these points. This choice determines an initial distribution of mass which the algorithm then attempts to maintain.

Notwithstanding the generality of the initial distribution an initial mesh may also be obtained by the equidistribution technique [26]. Under this technique mesh points are generated by equalizing the integral of the initial function u_0, so that the integral

$$\int_{x_{i-1}}^{x_i} u_0(t_0,x)dx \tag{4.80}$$

takes the same value in each interval. The mesh is only implicit in Eq. (4.80) but can be extracted by an iterative inversion procedure, such as that given in [6].

From the point of view of equalizing the error between the initial function and its discrete representation on a given mesh, a discrete form of $u_0(x)$ may be obtained by minimizing the L_2 error between the two. Supposing that $U(x)$ is the L_2 projection of $u_0(x)$ into the $span\{W_i\}$ the normal equations are

$$\int_a^b W_i(U(t_0,x)-u_0(t_0,x))dx=0 \tag{4.81}$$

which lead, via the expansion $U(x)=\sum_j U_j W_j$, to the matrix form

$$\mathcal{M}\underline{U}=\underline{f}_0, \tag{4.82}$$

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

where $\mathcal{M}$ is the standard mass matrix, $\underline{U}$ is a vector of coefficients U_j, and the components of the vector $\underline{f}_0$ are $\int_a^b W_i u_0(x)dx$. This choice is appropriate if we wish to pursue the propagation of the L_2 best fit to self-similar solutions property proved in Appendix B.

The L_2 error can further be minimized over node positions as well as nodal values to give an optimal initial mesh [5,138].

4.6.2 Initial data in higher dimensions

Again, the simplest initial mesh in higher dimensions consists of roughly equally spaced points (connected as a Delaunay triangulation [27], say) and the simplest initial values are obtained by sampling the initial condition at these points.

In multidimensions equidistribution is less intuitive and depends on a variational approach (see, e.g., [74]).

Alternatively, the initial values may be readily obtained from minimization of the L_2 error between the initial function and its discrete representation $U(t_0,\mathbf{x})$ in the space spanned by the basis functions $W_i(t_0,\mathbf{x})$ on a given mesh, for which the normal equations are

$$\int_{\overline{\mathcal{R}}(0)} W_i\{U(t_0,\mathbf{x})-u_0(t_0,\mathbf{x}))\}d\mathbf{x}=0 \tag{4.83}$$

leading to (4.82) with

$$\underline{f}_0=\int_{\overline{\mathcal{R}}(0)} W_i u_0(\mathbf{x})d\mathbf{x}, \tag{4.84}$$

which is again appropriate for the purpose of invoking the propagation of the best L_2 fit to self-similar solutions property proved in Appendix B.

The L_2 error can again be minimized over node positions as well as nodal values in multidimensions to give an optimal mesh [5,138].

4.7 Examples

We now illustrate the multidimensional finite element conservation method for mass-conserving problems on a number of test problems using the algorithm of Subsection 4.3.5. In all the problems shown $u=0$ on the boundary, so the right hand side of (4.49) is known.

4.7.1 The porous medium equation

For the PME (2.17) with $U=0$ imposed weakly on $\partial\overline{\mathcal{R}}(t)$, the velocity potential Φ satisfies (4.51) with

$$f_i=\int_{\overline{\mathcal{R}}(t)} W_i\{\nabla\cdot(U^n\nabla U+U\mathbf{Q})\}d\mathbf{x}. \tag{4.85}$$

The velocity is then found from (4.57), the new mesh from an explicit time-stepping of (4.63), and the solution from (4.67), using (4.65) in the form

$$\dot{\mu}_i=\frac{d}{dt}\int_{\overline{\mathcal{R}}(t)} W_i U d\mathbf{x}=-\int_{\overline{\mathcal{R}}(t)}\nabla W_i\cdot(U^n\nabla U+U\mathbf{V})d\mathbf{x}, \tag{4.86}$$

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

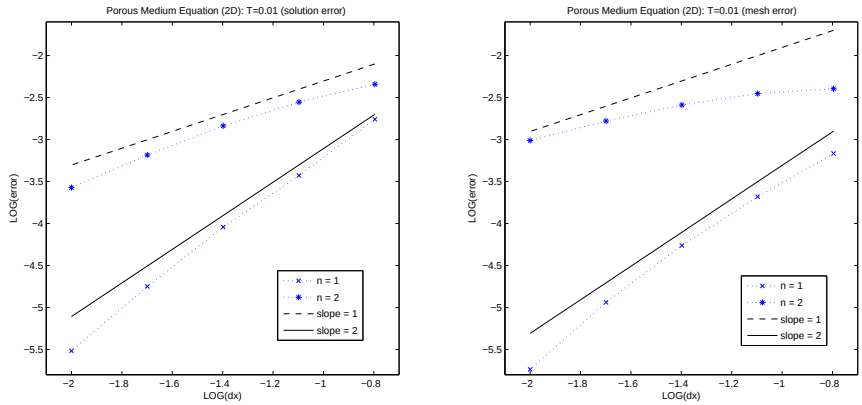

Figure 1: Accuracy of the approximate solutions to the two-dimensional porous medium equation ($n=1$ and $n=2$) on a sequence of meshes at $T=0.01$ obtained using the standard scheme: solution error (left) and mesh error (right) both in the L_1 norm.

(after the use of Green's Theorem and setting $U=0$ on the boundary) to obtain $\underline{\mu}$.

Mass is conserved for this problem and there exists an exact radially symmetric solution in d dimensions [14, 113] given by

$$u(r,t)=\begin{cases} \dfrac{1}{(\lambda(t))^d}\left(1-\left(\dfrac{r}{r_0\lambda(t)}\right)^2\right)^{\frac{1}{n}}, & |r|\le r_0\lambda(t), \\ 0, & |r|>r_0\lambda(t), \end{cases} \tag{4.87}$$

in which d is the number of space dimensions, r is the usual radial coordinate, and

$$\lambda(t)=\left(\frac{t}{t_0}\right)^{\frac{1}{2+dn}}, \qquad t_0=\frac{{r_0}^2 n}{2(2+dn)}. \tag{4.88}$$

Mesh convergence results for the method are shown in Fig. 1 for $n=1$ and $n=2$. The errors shown are computed from

$$\text{solution error}=\frac{1}{N}\sum_{i=1}^{N}|U_i-u(\mathbf{X}_i)|,$$

$$\text{mesh boundary error}=\frac{1}{N_B}\sum_{i=1}^{N_B}|R_i-r|,$$

in which N is the number of mesh nodes, N_B is the number of boundary nodes, u and r are the analytic solution and domain radius and R_i is the distance of boundary node i from the origin. The solutions differ qualitatively for different values of n since from (4.87) for $n\le 1$ the similarity solution has finite slope normal to the boundary whereas for $n>1$ it has infinite slope. This has the effect of lowering the order of convergence when n is increased beyond 1. The convergence results have been obtained using a series of unstructured meshes generated on a circle of radius 0.5 by an advancing front

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

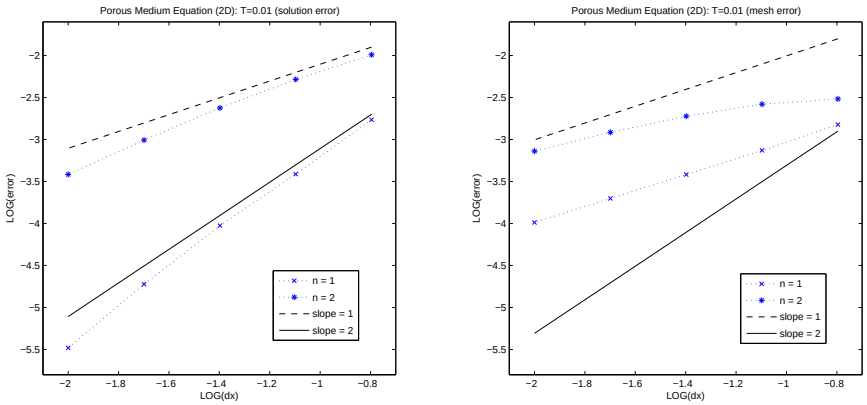

Figure 2: Accuracy of the approximate solutions to the two-dimensional porous medium equation ($n=1$ and $n=2$) on a sequence of meshes at $T=0.01$ obtained using the exactly conservative scheme but with weakly imposed Dirichlet boundary conditions on u: solution error (left) and mesh error (right) both in the L_1 norm.

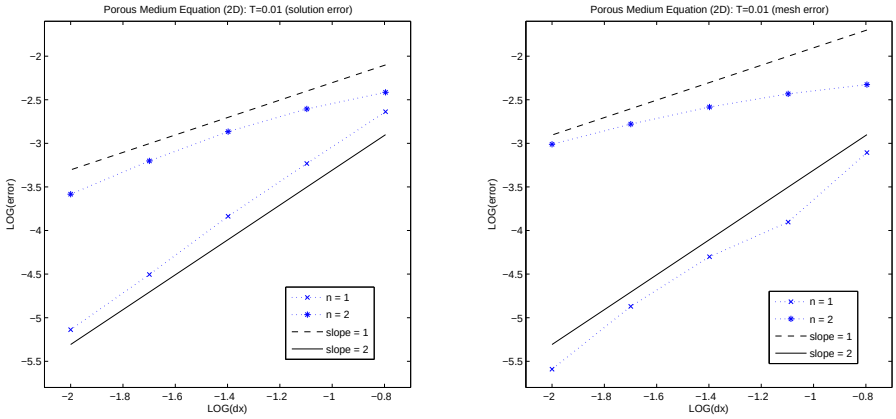

Figure 3: Accuracy of the approximate solutions to the two-dimensional porous medium equation ($n=1$ and $n=2$) on a sequence of meshes at $T=0.01$ obtained using the standard scheme but with a non-zero vorticity added to the mesh velocity field: solution error (left) and mesh error (right) both in the L_1 norm.

method [93] in which the mesh size parameter has been repeatedly halved. In fact, every two-dimensional experiment presented in this paper has been carried out on unstructured meshes of this type. Unless stated otherwise, all subsequent experiments shown use the conservative ALE update (4.72) for the U with strongly imposed Dirichlet boundary conditions (and take $\mathbf{Q}\equiv\mathbf{0}$). The time-step is always chosen to ensure that the error due to the temporal discretization is negligible compared to that of the spatial discretization. Numerical evidence suggests that this is the case for any stable time-step when Heun's scheme is used [75].

The results shown in Fig. 1 impose the Dirichlet boundary condition $U=0$ exactly (see Section 4.3.4) and are exactly conservative (to machine accuracy). Similar orders of convergence are seen for non-conservative and weakly imposed boundary conditions

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

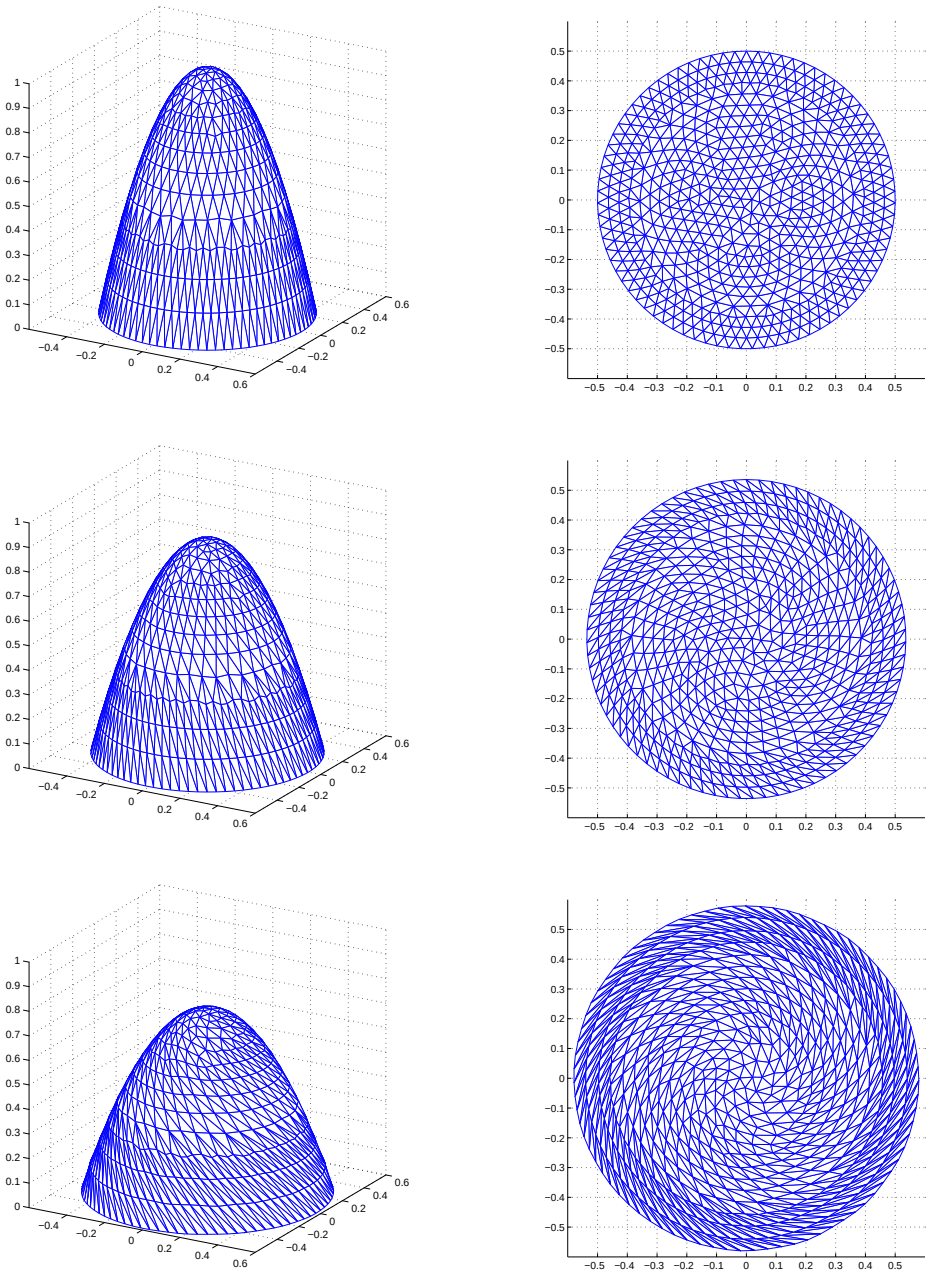

Figure 4: Snapshots of the solution and mesh ($T=0,0.01,0.025$) showing the evolution of a porous medium equation similarity solution ($n=1$) on a rotating mesh.

(though occasionally they are slightly lower, as with the mesh error in Fig. 2 for $n=1$): results for this test case are given in Fig. 2 and a full comparison can be found in [75].

Fig. 3 shows that the results remain very similar when a non-zero vorticity is added

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

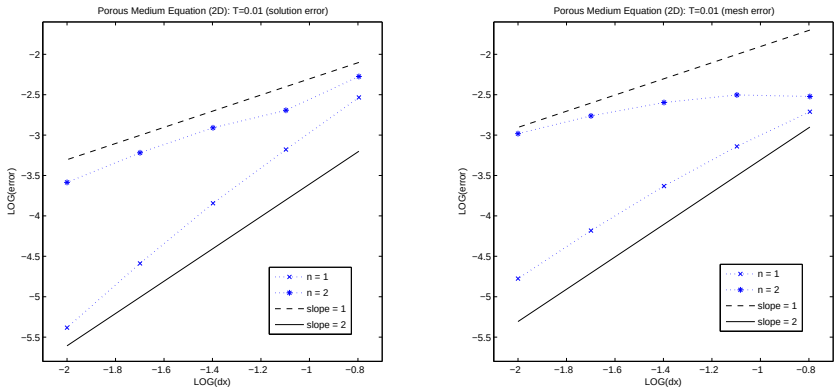

Figure 5: Accuracy of the approximate solutions to the two-dimensional porous medium equation ($n=1$ and $n=2$) on a sequence of meshes at $T=0.01$ obtained with upwinded velocity recovery: solution error (left) and mesh error (right) both in the L_1 norm.

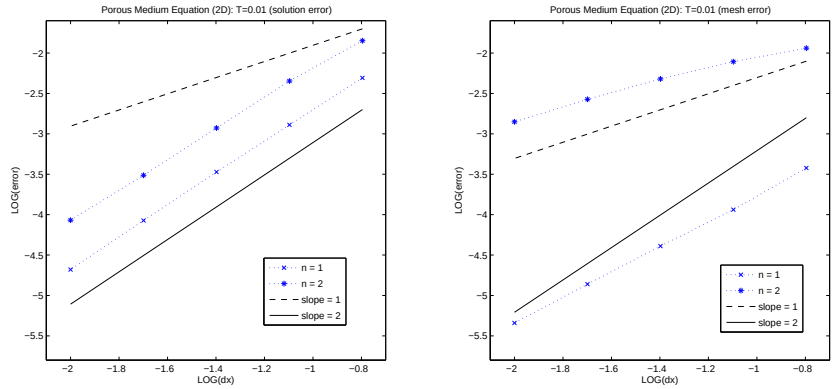

Figure 6: Accuracy of the approximate solutions to the two-dimensional porous medium equation ($n=1$ and $n=2$) on a sequence of meshes at $T=0.01$ obtained with a best L_2 fit to the initial data: solution error (left) and mesh error (right) both in the L_1 norm.

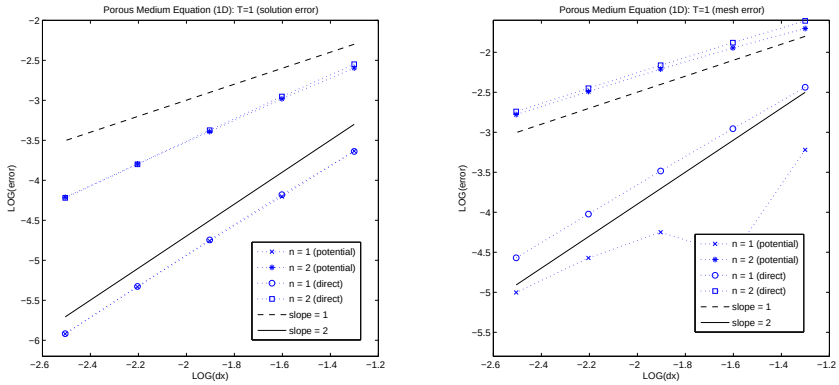

Figure 7: Accuracy of the approximate solutions to the one-dimensional porous medium equation ($n=1$ and $n=2$) on a sequence of meshes at $T=1.0$ obtained by recovering the velocity directly and via a potential: solution error (left) and mesh error (right) both in the L_1 norm.

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

to the mesh velocity field. In this case $\mathbf{q}$ is chosen to be $(-200yr, 200xr)^T$, where r is the usual radial coordinate, which is divergence-free and provides a rotation which increases in magnitude with distance from the origin. Fig. 4 shows three snapshots of the evolution of the mesh and the solution on this rotating mesh.

Fig. 5 shows the same mesh convergence study carried out for the upwind weighted velocity update (4.61) from the potential with coefficients as in (4.62). There is very little difference between these results and those obtained using a standard Galerkin recovery, though they are typically slightly less accurate.

Fig. 6 shows the convergence results corresponding to those shown in Fig. 1, the only difference being that the numerical experiments have been initiated with a best L_2 fit to the analytical solution at $T=0$ constrained to satisfy the Dirichlet boundary conditions, instead of nodally exact values. The main difference is that the least squares approximation appears to give a higher order of accuracy in the approximation to the solution when $n=2$.

The final set of convergence results in this section are for the one-dimensional porous medium equation, obtained using a series of initially uniform meshes. Fig. 7 provides a direct comparison between the standard scheme from the algorithm in Subsection 4.3.5 and the modification in which the mesh velocity is found directly from (4.76) instead of using a potential. There is very little difference between the two sets of results except for the mesh accuracy when $n=1$. Here the convergence is more erratic (although the error is smaller) when the potential is used, though the two approaches appear to be converging towards each other as the mesh is refined.

4.7.2 Fourth order nonlinear diffusion

The second example is the fourth order nonlinear diffusion (or thin film) equation [22, 128] given by

$$u_t = \nabla\cdot(u^n\nabla p), \quad \text{where } p=-\Delta u, \quad \text{in } \mathcal{R}(t), \tag{4.89}$$

with boundary conditions $u=\partial u/\partial n=\partial p/\partial n=0$, on $\partial\mathcal{R}(t)$. The velocity potential Φ is given by Eq. (4.51) with

$$f_i=\int_{\overline{\mathcal{R}}(t)} W_i\{\nabla\cdot(U^n\nabla P+U\mathbf{Q})\}d\mathbf{x}, \tag{4.90}$$

in which the piecewise linear $P\approx p$ must be recovered in an appropriate manner. Here we choose a standard finite element projection, which generates the normal equations

$$\int_{\mathcal{R}(t)} W_i P\,d\mathbf{x}=\int_{\mathcal{R}(t)}\nabla W_i\cdot\nabla U\,d\mathbf{x}, \tag{4.91}$$

using the boundary condition $\partial u/\partial n=0$ (see [7]). This problem also has an explicitly defined radially symmetric similarity solution in d dimensions in the special case $n=1$, given by

$$u(r,t)=\begin{cases} At^\beta U_0(1-\eta^2)^2, & |\eta|\le 1,\\ 0, & |\eta|>1,\end{cases} \tag{4.92}$$

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

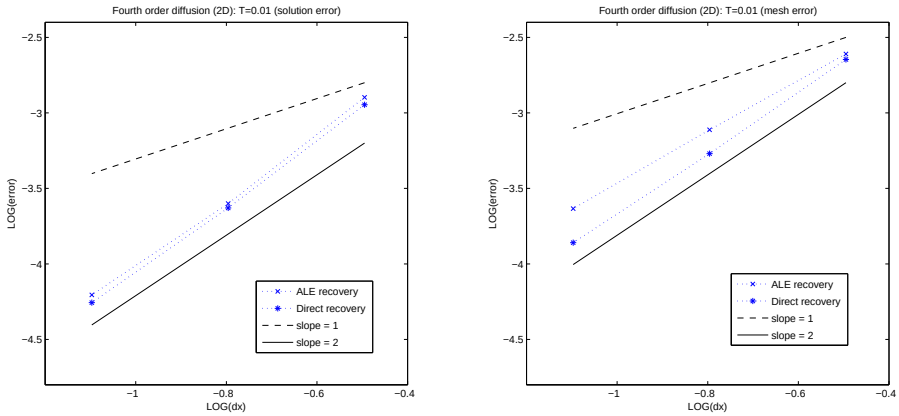

Figure 8: Accuracy of the approximate solutions to the two-dimensional fourth order nonlinear diffusion equation ($n=1$) on a sequence of meshes at $T=0.01$ obtained using the standard scheme: solution error (left) and mesh error (right) both in the L_1 norm.

in which

$$\eta=\frac{r}{A^{\frac{1}{4}}t^{\delta}},\qquad \delta=\frac{1}{4+d},\qquad \beta=4\delta-1,\qquad A=U_0{}^{-4\delta}, \tag{4.93}$$

and $U_0=1/192$ is chosen so that $u(0,t_0)=1$, where t_0 is such that the initial position of the moving front is at $r=1$.

Fig. 8 shows that the solution converges with order 2 as the mesh is refined. Here, the term "ALE recovery" refers to the use of (4.67) with (4.65), whilst "direct recovery" refers to (4.67) with (4.64). In [7] it was noted that in some circumstances fourth order convergence could be observed in one space dimension. Examples of the application of the finite element conservation method to non-radial problems can also be found in [7].

4.7.3 Waiting times for the porous medium equation

The waiting time T^* is the time it takes for the boundary to begin to move from a general initial condition and depends on the solution profile in the vicinity of the boundary [85, 128]. For radially symmetric solutions of the porous medium equation (2.17) it can be shown that

$$T^*>0,\quad \text{if } u(r,0)\leq C|r-r_0|^{\frac{2}{n}}, \tag{4.94a}$$

$$T^*=0,\quad \text{if } u(r,0)\geq C|r-r_0|^{\gamma},\ \text{for } \gamma<\frac{2}{n}. \tag{4.94b}$$

The initial conditions used for the numerical tests which are now described were taken to be

$$u(r,0)=\begin{cases}\left[1-\left(\dfrac{r}{r_0}\right)^2\right]^p, & r\leq r_0,\\ 0, & r>r_0,\end{cases} \tag{4.95}$$

so that the solution initially approximates $(r-r_0)^p$ at the moving boundary (cf. (4.87)).

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

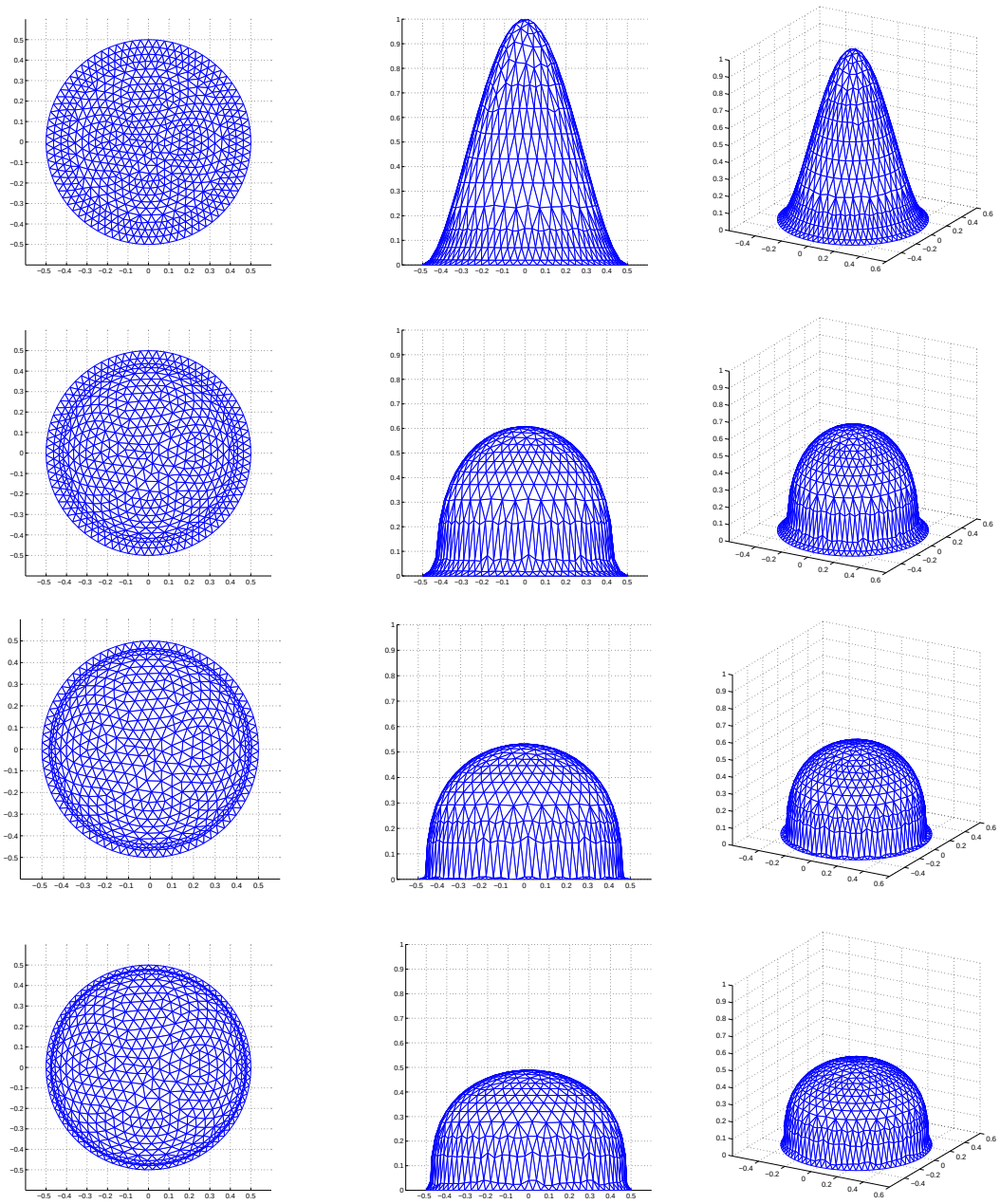

Figure 9: Snapshots of the solution and mesh ($T=0,0.1,0.2,0.3$) showing the evolution of a porous medium equation solution ($n=3$) from an initial profile which exhibits a non-zero waiting time.

Figs. 9 and 10 show the evolution of an initial profile with $p=2$ in the porous medium equation with $n=3$ and illustrate the typical behaviour of a waiting time solution and the evolution of the mesh as it attempts to sustain locally conservation. These results have

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

been obtained using the upwind approach of (4.61) to recovering the mesh velocity from the potential because it is slightly more robust than the Galerkin approach. A steep front evolves in the interior of the domain and the mesh clusters behind this front because this is where the "mass" is accumulating. This front then moves towards the stationary boundary, which only starts to move once the front arrives (see also [112]). The result also illustrates one of the issues with velocity-based moving mesh methods: if the ring of narrow cells becomes too thin then the mesh edges are bound to cross over if they remain straight lines. When the mesh is constructed via a transformation this can be avoided [30].

Fig. 11 shows results obtained while varying p for a fixed mesh resolution, which shows that non-zero waiting times are only observed when the theory indicates that they should be (for $n=3$ there should be non-zero waiting times for $p>2/3$), and also while varying the mesh size for a fixed value of p, which demonstrates that the results are converging to a particular value for the waiting time.

4.7.4 Richards' equation

The one-dimensional Richards' equation [60] models unsaturated unsteady flow in soils in the vertical x direction and is given by

$$u_t = \left(D(u)u_x + \kappa(u)\rho g\right)_x, \tag{4.96}$$

where $u(t,x)$ is the saturation of the wetting phase, $\kappa(u)$ is the relative permeability of the wetting phase and $D(u)$ the diffusion coefficient. This equation is suggested as a good model for the diffusion of pollutant in soils [98]. Stojsavljevic [132] has implemented the moving mesh finite element conservation method for the case when $D(u)=u$ and $\kappa(u)=u^3$ with zero Dirichlet boundary conditions and initial condition $u(0,x)=(1-x^2)_+^3$.

There is an exact solution for the velocity $v=-u_x-u^2$ (see 4.1.3) and with the above initial condition the solution also waits before moving, as seen in the numerical results shown in Fig. 12 (reproduced from [132]).

We now generalize the conservation method to non mass-conserving problems, governed by the PDE (2.1) or its integral form, in a domain $\mathcal{R}(t)$ when $u>0$ in the interior of the domain.

5 The conservation method for non mass-conserving problems

If the total mass in the problem varies with time then neither the local or distributed conservation principles (4.2) and (4.31) are consistent with a constant value of θ in (4.1) (which should then be written as $\theta(t)$ as in (5.2) below). However, a consistent conservation principle may be constructed by normalizing the integral (4.2) at the expense of carrying $\theta(t)$ as an extra variable.

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

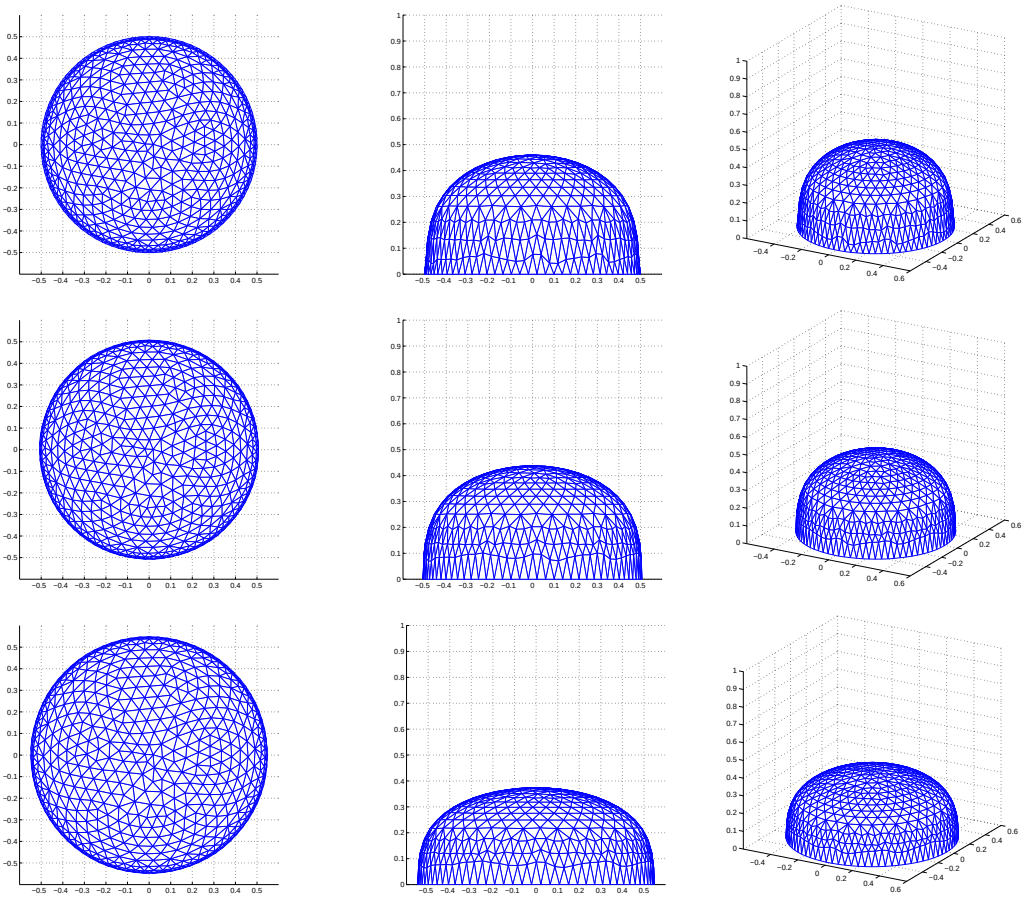

Figure 10: Snapshots of the solution and mesh ($T=0.4,0.5,1.0$) showing the evolution of a porous medium equation solution ($n=3$) from an initial profile which exhibits a non-zero waiting time.

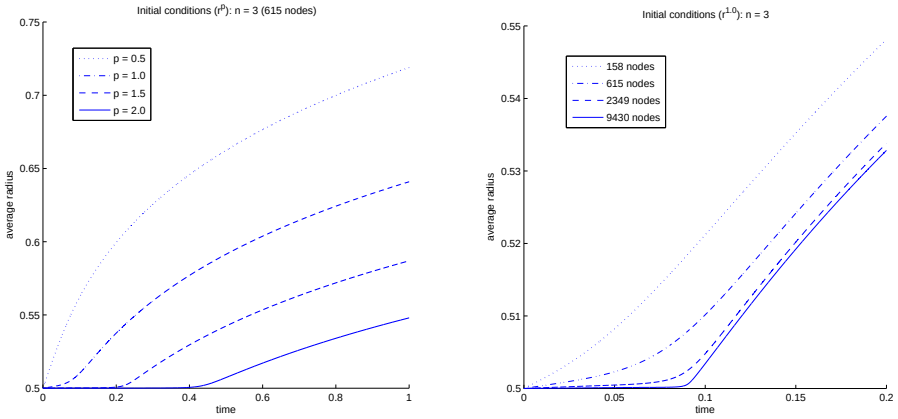

Figure 11: Comparison of the evolution of the boundary position for different initial solution profiles on the same 615 node mesh (left) and for different mesh resolutions given the same initial profile (right).

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

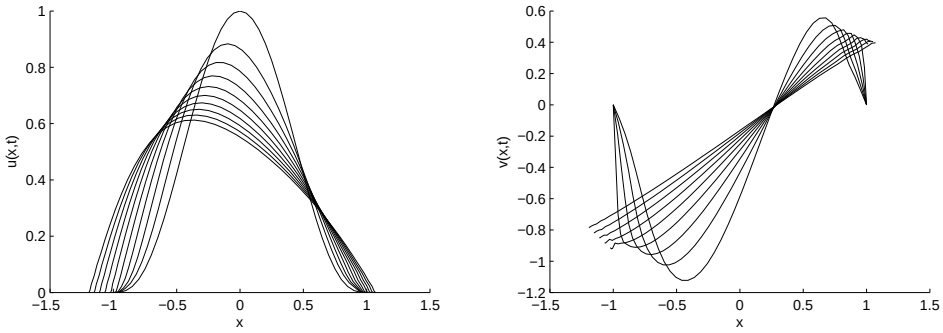

Figure 12: Evolution of the solution (left) and mesh velocity (right) for the one-dimensional Richards equation, reproduced from [132].

Let u be a positive solution of (2.1) in the interior of the moving domain $\Omega(t)$ and define the *relative* conservation principle,

$$\frac{1}{\theta(t)}\int_{\Omega(t)} u\,d\mathbf{x} = c(\Omega), \quad \text{independent of time,} \tag{5.1}$$

where the total (variable) mass

$$\theta(t) = \int_{\mathcal{R}(t)} u\,d\mathbf{x} \tag{5.2}$$

is the extra (normalizing) variable. Note that from (5.1) and (5.2), $c(\mathcal{R}) = 1$.

5.1 Velocity and solution

The Reynolds Transport Theorem (2.23) is now applied to the function $u/\theta(t)$, leading to the equation

$$\int_{\Omega(t)} u_t d\mathbf{x} + \oint_{\partial\Omega(t)} u\mathbf{v}\cdot\hat{\mathbf{n}}\,d\Gamma = c(\Omega)\dot{\theta} = \frac{\dot{\theta}}{\theta}\int_{\Omega(t)} u\,d\mathbf{x}, \tag{5.3}$$

where $\mathbf{v}\cdot\hat{\mathbf{n}}$ is the normal boundary velocity and we have used (5.1). By the Divergence Theorem (5.3) can be written

$$\int_{\Omega(t)} \{u_t + \nabla\cdot(u\mathbf{v})\}\,d\mathbf{x} = \frac{\dot{\theta}}{\theta}\int_{\Omega(t)} u\,d\mathbf{x}, \tag{5.4}$$

where $\mathbf{v}$ is any sufficiently smooth velocity field which matches the normal boundary velocity. Since $\Omega(t)$ is arbitrary, Eq. (5.4) yields the generalized pointwise Eulerian conservation equation (cf. (4.5))

$$u_t + \nabla\cdot(u\mathbf{v}) = \frac{\dot{\theta}}{\theta}u. \tag{5.5}$$

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

Substituting for u_t from the PDE (2.1), Eq. (5.3) takes the purely spatial integral form

$$\int_{\Omega(t)} \mathcal{L}u d\mathbf{x} + \oint_{\partial\Omega(t)} u\mathbf{v}\cdot\hat{\mathbf{n}} d\Gamma = c(\Omega)\dot{\theta} = \frac{\dot{\theta}}{\theta}\int_{\Omega(t)} u d\mathbf{x} \tag{5.6}$$

while Eq. (5.5) becomes the differential equation

$$\mathcal{L}u + \nabla\cdot(u\mathbf{v}) = \frac{\dot{\theta}}{\theta}u. \tag{5.7}$$

Either Eq. (5.6) or (5.7) can be regarded as an equation for $\mathbf{v}$ and $\dot{\theta}$.

In some cases $\dot{\theta}$ can be obtained explicitly. Applying (5.6) to the whole domain $\mathcal{R}(t)$ and using $c(\mathcal{R}) = 1$,

$$\int_{\mathcal{R}(t)} \mathcal{L}u\, d\mathbf{x} + \oint_{\partial\mathcal{R}(t)} u\mathbf{v}\cdot\hat{\mathbf{n}}\, d\Gamma = \dot{\theta}, \tag{5.8}$$

which gives $\dot{\theta}$ explicitly if either $\mathbf{v}\cdot\hat{\mathbf{n}}$ is given on the boundary $\partial\mathcal{R}(t)$ or if $u = 0$ there. The discussion is restricted to these cases in the subsequent argument.

Even if $\dot{\theta}$ is known there is no unique solution of Eq. (5.7) for $\mathbf{v}$ but, as in Section 4.1.1, if the vorticity $curl\,\mathbf{v} = curl\,\mathbf{q}$, where $\mathbf{q}$ is a given velocity, then there exists a velocity potential ϕ satisfying (4.7) and Eq. (5.7) can be written as (cf. (4.8))

$$-\nabla\cdot(u\nabla\phi) = \nabla\cdot(u\mathbf{q}) + \mathcal{L}u - \frac{\dot{\theta}}{\theta}u. \tag{5.9}$$

Since $u > 0$ in the interior of $\mathcal{R}(t)$ there is a unique solution of (5.9) for ϕ (at least to the extent of a constant) provided that ϕ or $u\partial\phi/\partial n$ is specified on the boundary $\partial\mathcal{R}(t)$.

Once ϕ and $\dot{\theta}$ have been found the velocity follows from (4.7). The evolution of points $\hat{\mathbf{x}}(t)$ of the region $\mathcal{R}(t)$ can be determined by integrating (1.1), and θ determined by integrating $\dot{\theta}$. Finally, the solution $u(t,\mathbf{x})$ may be recovered *a posteriori* from the Lagrangian conservation principle (5.1) in the form

$$\frac{1}{\theta(t)}\int_{\mathcal{R}(t)} u(t,\mathbf{x}) d\mathbf{x} = \frac{1}{\theta(t_0)}\int_{\mathcal{R}(t_0)} u(t_0,\mathbf{x}) d\mathbf{x} \tag{5.10}$$

using the technique described in Subsection 4.1.2.

5.2 Distributed forms

In the distributed case, with w_i belonging to a set of square-integrable test functions, fixed in $\mathcal{R}(t)$ and forming a partition of unity, the relative conservation principle (5.1) is modified to read

$$\frac{1}{\theta(t)}\int_{\mathcal{R}(t)} w_i u d\mathbf{x} = c_i, \quad \text{independent of time,} \tag{5.11}$$

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

where u is a positive solution of (4.32) in the interior and $\theta(t)$ is given by (5.2). Since the w_i form a partition of unity, by summing Eq. (5.11) over i we still obtain $\sum c_i=1$. (We note in passing that (5.11) reduces to (5.1) when w_i is the characteristic function in $\Omega(t)$.)

Then, applying the Reynolds Transport Theorem to the function $w_iu/\theta(t)$, and assuming that the points of $\mathcal{R}(t)$ and therefore the w_i move with the velocity $\mathbf{v}$, the unknowns $\mathbf{v}$ and $\dot{\theta}$ satisfy

$$\int_{\mathcal{R}(t)} w_i\{u_t+\nabla\cdot(u\mathbf{v})\}d\mathbf{x}=c_i\dot{\theta} \tag{5.12}$$

(cf. (4.34)), which leads via the weak form (4.32) of the PDE (2.1) to the spatial equation

$$\int_{\mathcal{R}(t)} w_i\{\mathcal{L}u+\nabla\cdot(u\mathbf{v})\}d\mathbf{x}=c_i\dot{\theta}. \tag{5.13}$$

Further assuming that $w_i\in H^1\{\mathcal{R}(t)\}$ for all t, we apply Green's Theorem to (5.13), giving

$$\int_{\mathcal{R}(t)} u\nabla w_i\cdot\mathbf{v}d\mathbf{x}=\int_{\mathcal{R}(t)} w_i\mathcal{L}ud\mathbf{x}+\oint_{\partial\mathcal{R}(t)} w_iu\mathbf{v}\cdot\hat{\mathbf{n}}d\Gamma-c_i\dot{\theta}. \tag{5.14}$$

Summing Eq. (5.13) over i and using $\sum c_i=1$ leads to

$$\int_{\mathcal{R}(t)} \mathcal{L}u+\oint_{\partial\mathcal{R}(t)} u\mathbf{v}\cdot\hat{\mathbf{n}}d\Gamma=\dot{\theta} \tag{5.15}$$

which yields $\dot{\theta}$ explicitly if either $\mathbf{v}\cdot\hat{\mathbf{n}}$ is given on the boundary $\partial\mathcal{R}(t)$ or if $u=0$ there. We shall that assume that this holds in the following argument.

There is no unique solution of Eq. (5.14) for $\mathbf{v}$ but if there exists a velocity potential ϕ satisfying the weak form (4.36) then (5.14) can be written

$$\int_{\mathcal{R}(t)} u\nabla w_i\cdot\nabla\phi d\mathbf{x}=\int_{\mathcal{R}(t)}\{w_i\mathcal{L}u-u\nabla w_i\cdot\mathbf{q}\}d\mathbf{x}+\oint_{\partial\mathcal{R}(t)} w_iu\mathbf{v}\cdot\hat{\mathbf{n}}d\Gamma-c_i\dot{\theta}. \tag{5.16}$$

Since $u\mathbf{v}\cdot\hat{\mathbf{n}}$ is given on the boundary $\partial\mathcal{R}(t)$, so that the penultimate term in (5.16) is known, there is a unique solution of (5.16) for $\phi\in span\{w_i\}$ provided that ϕ is specified at at least one point of the domain (which is no restriction since only $\nabla\phi$ is required).

Once ϕ and $\dot{\theta}$ have been determined, if $\mathbf{v},\mathbf{q}\in span\{\nabla w_i\}$ then $\mathbf{v}$ is given by (4.7), as in Subsection 4.2.1. Otherwise, an approximate velocity $\widetilde{\mathbf{v}}$ can be obtained from the weak form (4.38). The evolution of the Lagrangian coordinate $\widehat{\mathbf{x}}(t)$ is then determined by integrating (1.1) over time, and $\theta(t)$ by integrating $\dot{\theta}$. In the case of the weak form (4.38) the velocity $\widetilde{\mathbf{v}}$ is not exactly equal to $\mathbf{q}+\nabla\phi$ and the conservation principle (5.11) therefore holds only approximately. Although we can still use (5.11) to recover an approximate solution from the equation

$$\frac{1}{\theta(t)}\int_{\mathcal{R}(t)} w_iu\,d\mathbf{x}=\frac{1}{\theta(t_0)}\int_{\mathcal{R}(t_0)} w_i(t_0)u(t_0)d\mathbf{x}, \tag{5.17}$$

a more accurate recovery of $u(t,\mathbf{x})$ is obtained, as in Section 4.2.1, from the weak form of the ALE equation (4.43) by integrating the right-hand side in time prior to the inversion of (4.42). By using the approximate velocity $\widetilde{\mathbf{v}}$ and the ALE equation the conservation principle (5.11) is bypassed and the c_i are expected to change slowly with time.

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

5.3 The moving mesh finite element method

For the finite element mesh and approximations we assume the same set-up as in Section 4.3.

In the finite-dimensional space the normalized conservation principle (5.11) for non mass-conserving problems becomes

$$\frac{1}{\Theta(t)}\int_{\overline{\mathcal{R}}(t)} W_i U d\mathbf{x} = C_i, \quad \text{where } \Theta(t) = \int_{\overline{\mathcal{R}}(t)} U d\mathbf{x} \tag{5.18}$$

and the C_i are assumed to be independent of time (at least for the purpose of deriving the mesh velocity potential). Then, by (5.14), the unknowns $\mathbf{V}$ and $\dot{\Theta}$ satisfy

$$\int_{\overline{\mathcal{R}}(t)} U\nabla W_i\cdot\mathbf{V}d\mathbf{x} = \int_{\overline{\mathcal{R}}(t)} W_i\mathcal{L}U d\mathbf{x} + \oint_{\partial\overline{\mathcal{R}}(t)} W_i U\mathbf{V}\cdot\hat{\mathbf{n}}d\Gamma - C_i\dot{\Theta}, \tag{5.19}$$

while summation of (5.19) over i gives

$$\dot{\Theta} = \int_{\overline{\mathcal{R}}(t)} \mathcal{L}U\, d\mathbf{x} + \oint_{\partial\overline{\mathcal{R}}(t)} U\mathbf{V}\cdot\hat{\mathbf{n}}\, d\Gamma \tag{5.20}$$

so that, since it has been assumed that $U\mathbf{V}\cdot\hat{\mathbf{n}}$ is given on the boundary, $\dot{\Theta}$ is known explicitly.

There is no unique solution of (5.19) for $\mathbf{V}$ in general but if there exists a square-integrable velocity $\mathbf{Q}$ and velocity potential Φ satisfying the weak form (4.48) then Eq. (5.19) can be written

$$\int_{\overline{\mathcal{R}}(t)} U\nabla W_i\cdot\nabla\Phi d\mathbf{x} = \int_{\overline{\mathcal{R}}(t)} \{W_i\mathcal{L}U - U\nabla W_i\cdot\mathbf{Q}\}d\mathbf{x} + \oint_{\partial\overline{\mathcal{R}}(t)} W_i U\mathbf{V}\cdot\hat{\mathbf{n}}d\Gamma - C_i\dot{\Theta}. \tag{5.21}$$

Since $U>0$, Eq. (5.21) has a unique solution for $\Phi\in S^1\{\mathcal{R}(t)\}$ (since $U\mathbf{V}\cdot\hat{\mathbf{n}}$ is known on the boundary) provided that Φ is given at at least one point in $\overline{\mathcal{R}}(t)$ (which is no restriction since only $\nabla\Phi$ is required).

The matrix form for $\underline{\Phi}$ is again (4.54) (one component of $\underline{\Phi}$ having been specified) but with the $\underline{f}$ components given by

$$f_i = \int_{\overline{\mathcal{R}}(t)} \{W_i\mathcal{L}U - U\nabla W_i\cdot\mathbf{Q}\}\, d\mathbf{x} + \oint_{\partial\overline{\mathcal{R}}(t)} W_i U\mathbf{V}\cdot\hat{\mathbf{n}}\, d\Gamma - C_i\dot{\Theta}. \tag{5.22}$$

The velocity is derived as in Subsection 4.3 and both $\widehat{\mathbf{X}}(t)$ and $\Theta(t)$ stepped forward in time.

The solution U can be approximately recovered from the first of (5.18), either directly via the finite element form of (5.17),

$$\frac{1}{\Theta(t)}\int_{\overline{\mathcal{R}}(t)} W_i U\, d\mathbf{x} = \frac{1}{\Theta(t_0)}\int_{\mathcal{R}(t_0)} W_i(t_0)\, U(t_0) d\mathbf{x}, \tag{5.23}$$

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

or by evaluating the μ_i of (4.64) from the conservative ALE equation (4.65) prior to inverting it for U via the technique of Subsection 4.3.3. By using the ALE equation the conservation principle (5.18) is bypassed: hence the C_i may vary slowly in time and should be recalculated at each time step.

Thus for non mass-conserving problems in which $\mathbf{V}\cdot\mathbf{n}$ is given on the boundary or $U=0$ there, we have the following algorithm.

5.3.1 Algorithm for non mass-conserving problems

Given an initial mesh $\{\mathbf{X}^0\}$ and nodal values $\{U_i^0\}$ on the mesh, representing the initial condition, at each time step:

1. Find $\dot{\Theta}$ from (5.20).
2. Find Φ from (4.54) with $\underline{f}$ given by (5.22).
3. Find $\widetilde{\mathbf{V}}$ from (4.55) via (4.57).
4. Advance $\widehat{\mathbf{X}}(t)$ and Θ in time using any convenient time-stepping scheme applied to $\widetilde{\mathbf{V}}$ and $\dot{\Theta}$.
5. Recover U_i at the new time-step by integrating the ALE equation (4.65) in time to obtain the $\dot{\mu}_i$ of (4.65), advancing μ_i in time, and using (4.67). (Strong Dirichlet boundary conditions can also be imposed as described in Section 4.3.4.)

As in the case of the algorithm of Section 4.3.5 the velocity potential is computed from the Eulerian conservation law in the form (4.54). However, the velocity and solution make use of the weak forms (4.55) and (4.65) for which (5.18) does not hold precisely, so the constants C_i there need to be recomputed at each time-step. Steps 1, 2 and 4 of the algorithm are scale-invariant because all the equations used, as well as the W_i and $\widetilde{W}_i$, are individually scale-invariant.

The algorithm can be thought of as solving the ODE system

$$\frac{d}{dt}\begin{pmatrix}\underline{\widehat{\mathbf{X}}}\\ \Theta\\ \underline{\mu}\end{pmatrix}=\begin{pmatrix}\underline{\mathbf{V}}\\ \dot{\Theta}\\ \underline{\dot{\mu}}\end{pmatrix}, \tag{5.24}$$

where $\underline{\mu}=\{\mu_i\}$ may be replaced by the $\underline{\widetilde{\mu}}=\{\widetilde{\mu}_i\}$ of Subsection 4.3.4 in the case of strong Dirichlet conditions. The method is therefore again a method of lines, and (5.24) can be integrated approximately by any convenient time-stepping scheme.

5.3.2 Numerical example: the Crank-Gupta problem

We illustrate the conservation method for non mass-conserving problems by a single example using the algorithm of Subsection 5.3.1. The Crank-Gupta problem [18, 48, 109] in multiple space dimensions is

$$u_t=\nabla^2 u-1, \quad \text{in } \mathcal{R}(t), \tag{5.25}$$

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

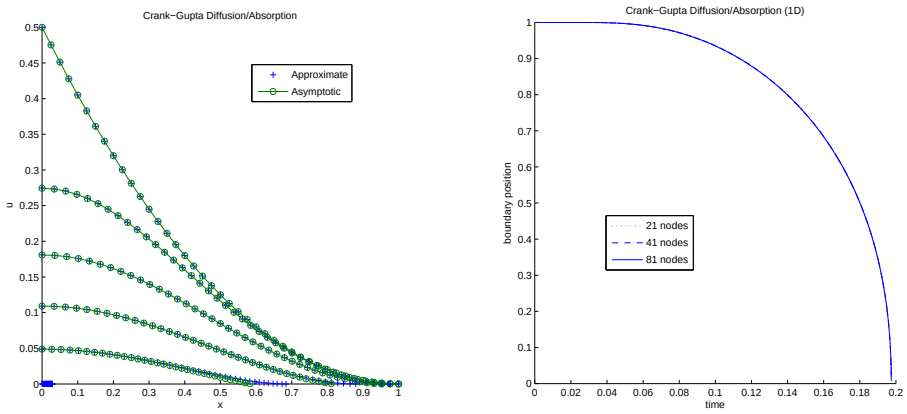

Figure 13: Snapshots of the evolution of the Crank-Gupta problem ($T=0,0.04,0.08,0.12,0.16,0.1974$) compared with the asymptotic approximation (left), and comparison of the rate of change of the position of the point on the moving boundary for different initial meshes (right).

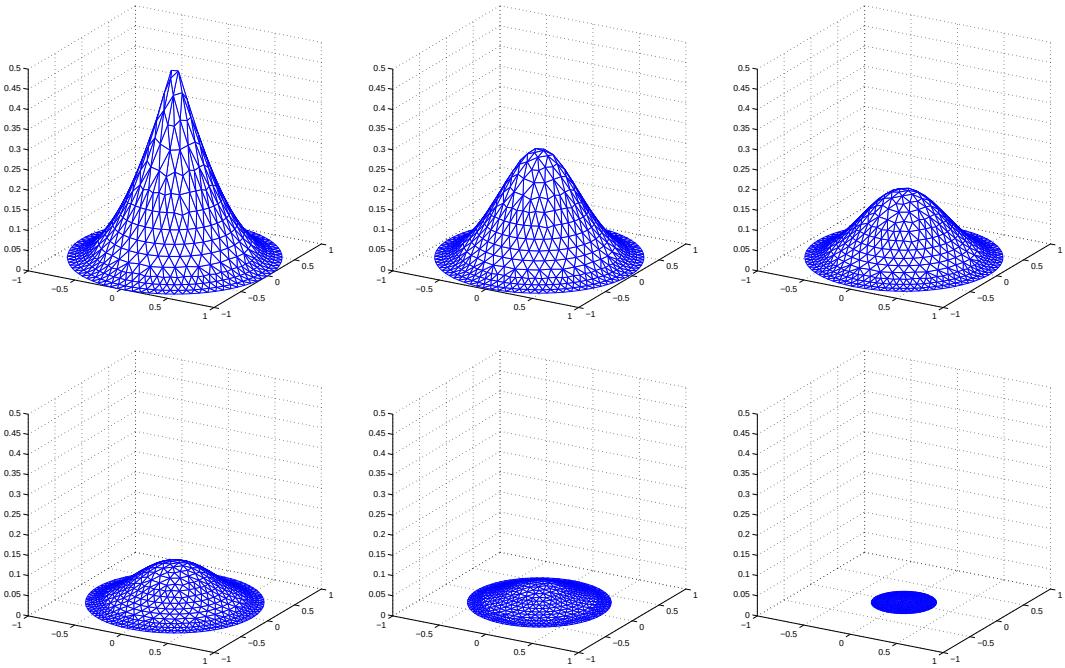

Figure 14: Snapshots of the evolution of the two-dimensional Crank-Gupta problem ($T=$ 0, 0.024, 0.048, 0.072, 0.096, 0.12) on a 615 node moving mesh.

with boundary conditions $u=0$ and $\partial u/\partial n=0$ on $\partial\mathcal{R}(t)$. Mass is not conserved for this problem but it is easily verified from (5.8), using the boundary conditions, that the rate of change of mass is given by $\dot{\theta}=-|\mathcal{R}|$.

It has already been shown in [7] that the conservation method is second order accurate for this equation when comparing with a one-dimensional exact solution with a different

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

boundary condition. Here, we compare with the original problem [48] in one dimension, which is solved on a half domain and therefore has a condition $u_x = 0$ imposed at the symmetry boundary, where $v = 0$ is also imposed. The initial condition is given by

$$u(r,0) = \begin{cases} \frac{1}{2}(1-x)^2, & x \le 1, \\ 0, & x > 1. \end{cases} \tag{5.26}$$

There is no known exact solution to this problem but some asymptotic expansions have been derived [48]. Fig. 13 shows the evolution of this initial profile and compares it to the asymptotic expansion. Fig. 13 also compares the movement of the free boundary node for a series of meshes (although these results are almost indistinguishable at this scale).

Fig. 14 shows snapshots of the evolution of a radially symmetric profile (computed as a two-dimensional problem) with x replaced by r in (5.26).

6 Extensions and further applications

In this section a number of extensions and applications are described which show the development and scope of the above moving mesh conservation method. We begin with its application to two-phase problems in Section 6.1 and then describe the generalization to monitor functions in Section 6.2. A corresponding finite difference method (in one dimension) is then briefly mentioned in Section 6.3. Finally references to some of the applications of the method are given in Section 6.4.

6.1 Internal boundaries

The conservation method is not restricted to predicting the evolution of problems for which the moving boundary is always the *exterior* of the computational domain. It is also possible to approximate the movement of sharp interfaces lying in the interior of a domain [10].

In order to do this each subregion defined by the internal and external boundaries must be considered separately. The main reason for this is that the mesh movement aims to preserve the proportion of the total mass associated with each test function. However, since each test function remains associated with the same subregion throughout the computation this is inconsistent with the proportion of the *total* mass associated with a given subregion changing with time. In general, for problems involving moving internal boundaries there is a transfer of mass into or across the moving boundary. Hence the algorithm used is based on that presented in Section 5.

6.1.1 Velocity and solution

A moving polygonal approximation $\overline{\mathcal{R}}(t)$ to $\mathcal{R}(t)$ is set up, consisting of a moving tessellation of simplices. This is then split up into non-overlapping polygonal subregions

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

$\overline{\mathcal{R}}_k(t)$ where $\overline{\mathcal{R}}(t)=\bigcup\overline{\mathcal{R}}_k(t)$ and the boundaries $\partial\overline{\mathcal{R}}_k(t)$ approximate the corresponding internal and external boundaries of the subregions of $\mathcal{R}(t)$. For clarity, the boundary of each subregion will be divided into fixed (F) and moving (M) components, i.e., $\partial\overline{\mathcal{R}}_k(t)=\partial\overline{\mathcal{R}}_k^F\cup\partial\overline{\mathcal{R}}_k^M(t)$. Once this has been done, the equation for the velocity potential (5.21) can be written as

$$\int_{\overline{\mathcal{R}}_k(t)} U\nabla W_i\cdot\nabla\Phi\,d\mathbf{x}=\int_{\overline{\mathcal{R}}_k(t)}\left\{W_i\mathcal{L}U-U\nabla W_i\cdot\mathbf{Q}\right\}d\mathbf{x}+\oint_{\partial\overline{\mathcal{R}}_k^M(t)} W_iU\mathbf{V}\cdot\hat{\mathbf{n}}\,d\Gamma-C_i\dot{\Theta}_k, \quad (6.1)$$

since $\mathbf{V}=\mathbf{0}$ on $\partial\overline{\mathcal{R}}_k^F$. Assuming that $U>0$ in the interior of $\mathcal{R}(t)$, Eq. (6.1) has a unique solution for $\Phi\in S^1\{\overline{\mathcal{R}}(t)\}$ in terms of $\dot{\Theta}_k$ provided that the interface integral is known and that Φ is given at at least one point. In problems of this type a condition on $\mathbf{V}$ at the moving interface is typically provided as part of the well-posed initial-boundary value problem. It is this condition that supplies the coupling between the subregions and ensures that the right-hand side of (5.19) may be evaluated.

Once the velocity potentials have been found in each of the subregions, the mesh velocities can be recovered using (4.48), solving over the whole domain at once in order to produce a velocity field which is continuous across the internal boundaries, and imposing $\mathbf{V}=\mathbf{0}$ on $\partial\overline{\mathcal{R}}_k^F$. This recovery step has been shown to be significantly more accurate if the equations involving the test functions associated with mesh nodes on the moving internal boundaries are replaced by a weak form of the condition on $\mathbf{V}$ which governs the movement of the interface [10].

Finally, after the nodal positions and each Θ_k have been updated using a standard timestepping scheme, the solution U is recovered separately in each of the subregions. This can be done using (4.67) or (4.69), either directly with fixed C_i and updated Θ_k on the right-hand side, or after the right-hand side has been updated using the ALE approach, (4.65) or (4.72).

6.1.2 Example: a two-phase Stefan problem

We illustrate the method using a two-phase Stefan problem. This problem, described in detail in [47] for example, models changes of phase between liquid and solid phases. The interfaces between these phases move as the conversion between them takes place and can be followed using the technique described above. The diffusion of heat within each phase is modelled by the equations [25]

$$K_Su_t=\nabla\cdot(k_S\nabla u), \quad \text{in solid regions,} \quad (6.2a)$$
$$K_Lu_t=\nabla\cdot(k_L\nabla u), \quad \text{in liquid regions,} \quad (6.2b)$$

in which u is the temperature and $K_{S,L}$, $k_{S,L}$ represent, respectively, the volumetric heat capacity and the thermal conductivity of the phase. From now on any subscript k, used to index the subregions, can be associated with the phase in that region (solid or liquid) and the corresponding equation taken from (6.2).

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

The conditions imposed on the moving interface $\partial\overline{\mathcal{R}}_k^M(t)$ in this system are $u=u_M$ and, based upon an energy balance across the phase-change boundary,

$$\left[k_k\nabla u_k\cdot\hat{\mathbf{n}}\right]_{\text{liquid}}^{\text{solid}}=\lambda\mathbf{v}\cdot\hat{\mathbf{n}}, \tag{6.3}$$

where $\hat{\mathbf{n}}=\hat{\mathbf{n}}(t)$ is a unit normal to the moving interface, λ is the heat of phase-change per unit volume, and $\mathbf{v}\cdot\hat{\mathbf{n}}$ is the normal velocity of the interface. Note that u is continuous but ∇u is discontinuous across the moving boundary, so the jump between the two phases (indicated by $[\cdot]$ in (6.3)) is generally non-zero.

In the finite element implementation where $U_M\neq 0$ (which can always be achieved since adding a constant to the solution still satisfies the remaining conditions) the equations for the velocity potentials in subregion k become (assuming for simplicity that $\mathbf{Q}=\mathbf{0}$):

$$\begin{aligned}C_i\dot{\Theta}_k+\int_{\overline{\mathcal{R}}_k(t)}U\nabla\Phi\cdot\nabla W_i\,d\mathbf{x}=&\kappa_k\oint_{\partial\overline{\mathcal{R}}_k^M(t)}W_i\nabla U_k\cdot\hat{\mathbf{n}}_k\,d\Gamma+\kappa_k\oint_{\partial\overline{\mathcal{R}}_k^F}W_i\nabla U_k\cdot\hat{\mathbf{n}}_k\,d\Gamma\\&-\kappa_k\int_{\overline{\mathcal{R}}_k(t)}\nabla W_i\cdot\nabla U\,d\mathbf{x}+\frac{U_M}{\lambda}\oint_{\partial\overline{\mathcal{R}}_k^M(t)}W_i\left[k_k\nabla U_k\cdot\hat{\mathbf{n}}_k\right]_{\text{liquid}}^{\text{solid}}d\Gamma,\end{aligned} \tag{6.4}$$

in which $\hat{\mathbf{n}}_k$ is the outward pointing unit normal to subregion k and $\kappa_k=k_k/K_k$. Once these systems are solved for a piecewise linear Φ in each region, the mesh velocity may be recovered at all of the internal nodes using (4.55) with $\mathbf{V}=\mathbf{0}$ on $\partial\overline{\mathcal{R}}_k^F$ and the equations associated with the mesh nodes on $\partial\overline{\mathcal{R}}_k^M(t)$ replaced by

$$\lambda\oint_{\partial\overline{\mathcal{R}}_k^M(t)}W_i\mathbf{V}\,d\Gamma=\oint_{\partial\overline{\mathcal{R}}_k^M(t)}W_i\left[k_k\nabla U_k\right]_{\text{liquid}}^{\text{solid}}d\Gamma. \tag{6.5}$$

Finally, the ALE update is given by

$$\begin{aligned}\dot{\widetilde{\mu}}_i=&\oint_{\partial\overline{\mathcal{R}}_k^M(t)}\widetilde{W}_i(\kappa_k\nabla U_k+U_M\mathbf{V})\cdot\hat{\mathbf{n}}_k\,d\Gamma+\kappa_k\oint_{\partial\overline{\mathcal{R}}_k^F}\widetilde{W}_i\nabla U_k\cdot\hat{\mathbf{n}}_k\,d\Gamma\\&-\int_{\overline{\mathcal{R}}_k(t)}\nabla\widetilde{W}_i\cdot(\kappa_k\nabla U+U\mathbf{V})\,d\mathbf{x}.\end{aligned} \tag{6.6}$$

When compared with analytical solutions the algorithm has demonstrated second order accuracy for both one- and two-phase Stefan problems [10]. A single, one-dimensional, example is given here to illustrate its capabilities: two-dimensional implementations can be found in [10]. Fig. 15 shows the evolution of a one-dimensional solution profile (liquid in the centre, surrounded by solid) where the outer boundaries are stationary and held at a fixed temperature. The left-hand side of the figure shows the evolution of the solution itself via snap-shots at discrete times. The right-hand side of the figure shows the smooth trajectories of the mesh nodes and the change of direction of the phase-change interface as the gradient of the temperature of the liquid phase at the interface drops below that of the solid phase. In the absence of an exact solution to this problem, Fig. 16 illustrates the convergence of the maximum value of the solution and the boundary position as the mesh is refined.

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

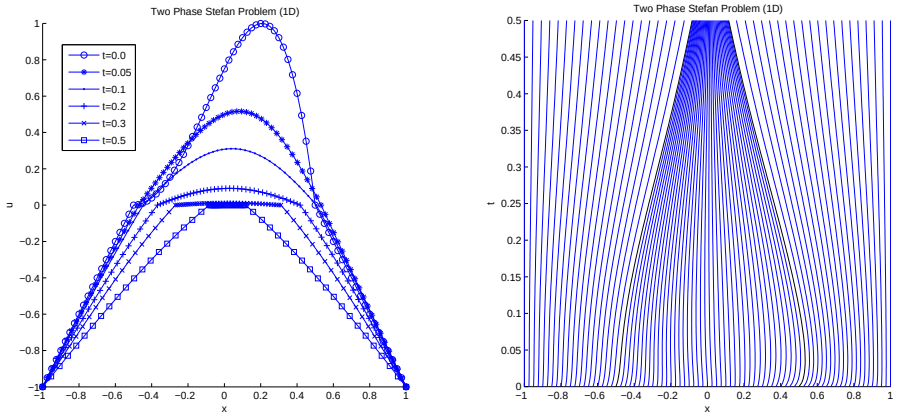

Figure 15: Evolution of the solutions to the one-dimensional two-phase Stefan problem for a non-symmetric initial profile and an 81 node mesh: snapshots of the solution profile (left) and node trajectories (right).

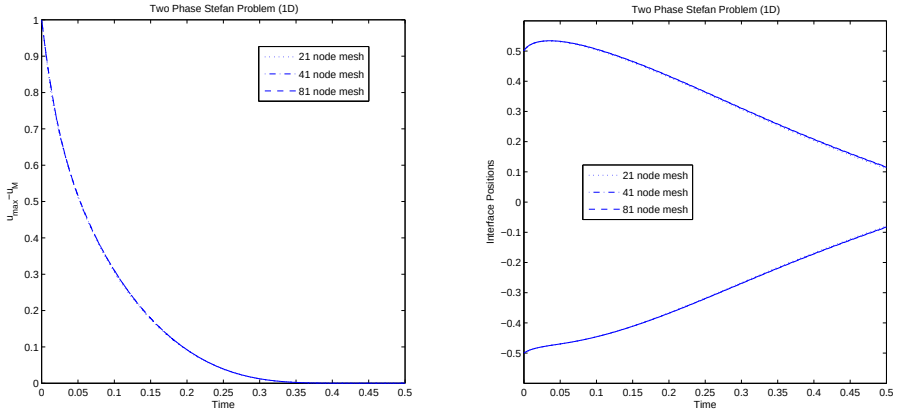

Figure 16: Evolution of the maximum value of U in the domain (left) and the position of the moving interfaces (right) for the one-dimensional two-phase Stefan problem with non-symmetric initial profile using three different, initially uniform, meshes.

6.2 Monitor integrals

We now discuss the generalization of the conservation method to incorporate the use of monitor functions. Instead of using u as the integrand in the conserved integral in (5.1) we may replace it by a general positive monitor function m, any measure of the dependent and independent variables. The integral corresponding to the conserved integral in (5.1) is then the monitor integral

$$\int_{\Omega(t)} m d\mathbf{x}. \tag{6.7}$$

An example of a monitor function is

$$m = \sqrt{(1+|\nabla u|^2)} \tag{6.8}$$

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

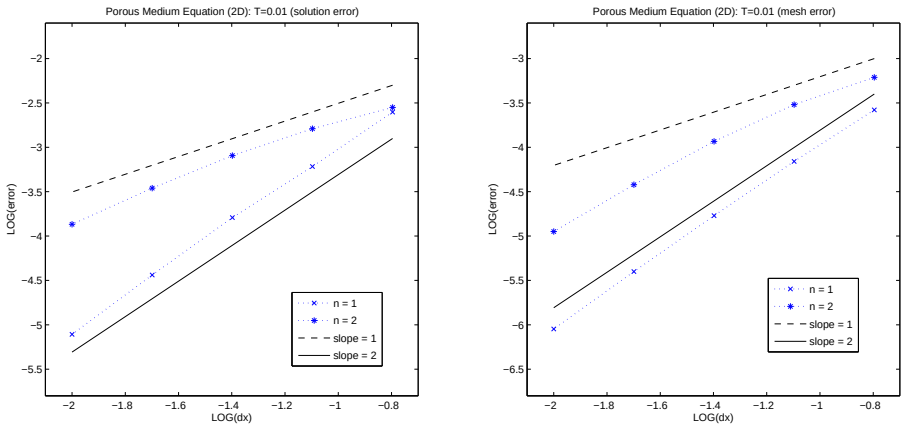

Figure 17: Accuracy of the approximate solutions to the two-dimensional porous medium equation ($n=1$ and $n=2$) on a sequence of meshes at $T=0.01$ obtained using the scale-invariant arclength monitor integral: solution error (left) and mesh error (right) both in the L_1 norm.

for which (6.7) becomes

$$\int_{\Omega(t)} \sqrt{\left(1+|\nabla u|^2\right)}\, d\mathbf{x}, \tag{6.9}$$

i.e., the area of the part of the manifold of u whose projection is $\Omega(t)$. We note that for this particular m,

$$\frac{dm}{du} = \frac{\nabla u}{\sqrt{\{1+(\nabla u)^2\}}}\nabla \tag{6.10}$$

is an operator.

Unlike the previous cases studied a general monitor will not necessarily have an associated scale-invariant monitor integral and therefore will not lead to a scale-invariant numerical method. However by replacing the variables in the monitor integral by the similarity variables (see (2.11)) we can ensure scale invariance [9]. For example, the radially symmetric scale-invariant monitor integral corresponding to (6.9) is

$$t^{-d\beta}\int_{\mathcal{R}(t)} \sqrt{\left(1+t^{-2(\gamma-\beta)}|\nabla u|^2\right)}\, d\mathbf{x}, \tag{6.11}$$

which is constructed from the area function in the space of similarity variables. Results obtained using this monitor are shown in Fig. 17 for the two-dimensional porous medium equation. Second order accuracy is generally achieved except, in this case, for the solution error when $n=2$.

Since the total monitor integral (6.7) is not independent of time in general, a consistent conservation principle must be constructed, as in Section 5, by normalizing the integral (6.7) at the expense of carrying the extra variable

$$\theta(t) = \int_{\mathcal{R}(t)} m\, d\mathbf{x}. \tag{6.12}$$

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

We therefore define the *relative* conservation principle to be

$$\frac{1}{\theta(t)}\int_{\Omega(t)} m d\mathbf{x} = c(\Omega), \quad \text{independent of time.} \tag{6.13}$$

It follows from (6.13) and (6.12) that $c(\mathcal{R})=1$.

By differentiating (6.13) with respect to time, in a similar way as in Section 5 the velocity $\mathbf{v}$ satisfies

$$\int_{\mathcal{R}(t)} m\nabla w_i\cdot\mathbf{v}d\mathbf{x} = \int_{\mathcal{R}(t)} w_i\frac{dm}{du}u_t d\mathbf{x} + \oint_{\partial\mathcal{R}(t)} w_i m\mathbf{v}\cdot\hat{\mathbf{n}}d\Gamma - c_i\dot{\theta}, \tag{6.14}$$

where we have used $m=m(u)$ and

$$m_t = \frac{dm}{du}u_t, \tag{6.15}$$

dm/du being the formal derivative of m with respect to u (see e.g., (6.10)). We cannot use the weak form (4.32) to substitute directly for u_t in (6.14) since the test function $(dm/du)w_i$ does not lie in the $span\{w_i\}$ in general. However, we may project $\mathcal{L}u$, into the $span\{(dm/du)w_i\}$ giving an approximation $\mathcal{P}u_t=\mathcal{P}\mathcal{L}u$ to u_t. leading to the approximate equation

$$\int_{\mathcal{R}(t)} m\nabla w_i\cdot\mathbf{v}d\mathbf{x} = \int_{\mathcal{R}(t)} w_i\frac{dm}{du}\mathcal{P}\mathcal{L}u d\mathbf{x} + \oint_{\partial\mathcal{R}(t)} w_i m\mathbf{v}\cdot\hat{\mathbf{n}}d\Gamma - c_i\dot{\theta} \tag{6.16}$$

for $\mathbf{v}$ and $\dot{\theta}$. The projection error means that the velocity computed from (6.16) only approximately corresponds to the conservation of (6.13) in time.

The velocity potential, the velocity, and Lagrangian coordinate can be calculated as before, and the solution obtained via the ALE equation. There is a corresponding finite element method. For further details on the method using monitor functions see [9,20,21, 148,149].

6.3 A finite difference method

Although this paper is primarily concerned with the finite element implementation of the conservation method we briefly describe a corresponding finite difference scheme in one dimension [20,21] which is equally effective.

When the total mass in the interval $(a(t),b(t))$ is constant in time the conservation principle (4.2) can be written

$$\int_{a(t)}^{\hat{x}(t)} u(t,x)dx = c(a,\hat{x}), \quad \text{independent of time,} \tag{6.17}$$

for any interval $(a(t),\hat{x}(t))$. Differentiating (6.17) with respect to time using Leibnitz' Integral Rule (cf. (2.22)) and using the PDE (2.1) gives the spatial equation for $v(t,x)$,

$$\int_{a(t)}^{\hat{x}(t)} \mathcal{L}u dx + [uv]_{a(t)}^{\hat{x}(t)} = 0 \tag{6.18}$$

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

(cf. (4.4)), where $v=d\widehat{x}/dt$.

Eq. (6.18) can be solved for v at any point $\widehat{x}(t)$ provided that v is given at $x=a(t)$. The Lagrangian coordinate $\widehat{x}(t)$ can then be found by time integration of

$$\frac{d\widehat{x}}{dt}=v \tag{6.19}$$

(cf. (1.1)).

The solution may be recovered approximately from the conservation principle (6.17) in its incremental form

$$\int_{\widehat{x}_{i-1}(t)}^{\widehat{x}_{i+1}(t)} u(t,x)dx=\int_{\widehat{x}_{i-1}(0)}^{\widehat{x}_{i+1}(0)} u(0,x)dx \tag{6.20}$$

by an application of the midpoint rule, leading to the approximation

$$u_i(t)=\frac{\widehat{x}_{i+1}(t_0)-\widehat{x}_{i-1}(t_0)}{\widehat{x}_{i+1}(t)-\widehat{x}_{i-1}(t)}u_i(t_0) \tag{6.21}$$

(cf. (4.12)).

As in the multidimensional case, scale-invariance is inherited for a scale-invariant problem, since the scale-invariant conservation principle (6.17) has been used to determine the mesh. The one-dimensional method is readily extended to incorporate monitor functions [20, 21]. Results from several applications of the one-dimensional method are cited in the subsection 6.4.2 below.

6.4 Further applications

In this section we briefly describe and reference several further applications of the conservation method in both finite element and finite difference form.

6.4.1 Other nonlinear diffusion problems

The sixth order nonlinear diffusion problem [16, 73] is

$$u_t=\nabla\cdot(u\nabla\cdot\Delta^2 u), \tag{6.22}$$

with boundary conditions $u=\partial u/\partial n=\partial^3 u/\partial n^3=\partial^5 u/\partial n^5=0$. The moving mesh finite element approach to the solution of this problem is very similar to that for fourth order diffusion in Section 4.7.2. Letting $q=-\Delta u$ and $p=-\Delta q$, Eq. (6.22) can be written

$$u_t=\nabla\cdot(u\nabla p) \tag{6.23}$$

(which from Section 4.1.3 has the exact solution $v=-\nabla p$ for the velocity).

In the moving mesh finite element method the finite element approximation $P\approx p$ is recovered from $Q\approx q$ and Q is in turn recovered from U by a finite element projection,

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

as in Section 4.7.2. Some results of the application of the moving mesh finite element conservation method to the sixth order problem can be found in [19].

Another application involving nonlinear diffusion is the modelling of water uptake in rice. The problem is given by a nonlinear diffusion equation of the form

$$c_t = \nabla\cdot(D(c)\nabla c) \tag{6.24}$$

in $\mathcal{R}(t)$, where $c=\bar{c}$ is held constant on the moving boundary $\partial\mathcal{R}(t)$. Results for a particular choice of $D(c)$, corresponding to a circularly symmetric rice grain, are shown in [111].

6.4.2 One-dimensional applications using finite differences

A number of one-dimensional problems have been treated by the finite difference method of Subsection 6.3 in the references cited below.

- The porous medium equation with $n=4$ has been studied in [110].
- Blow-up in Fisher's equation $u_t = u_{xx}+u^p$, with $p=2$ can be found in [20, 43, 140], and in the Kassoy equation $u_t = u_{xx}+e^u$ in [140].
- A model volcano equation $h_t+\mathcal{U}_x = w_s(x)$, where h is the height of the profile and the rheology term $\mathcal{U}$ is a function of h and $\partial h/\partial x$ derived from the physics [13], can be found in [120].
- A model glacier equation [42], $H_t = (H^5H_x^3)_x+s$, where H is the ice depth and s is precipitation, with a zero Dirichlet condition at the moving boundary, has been treated in [112, 119].

6.4.3 Applications to systems of equations

For systems of equations the conservation method has been applied, amongst others, to the Shallow Water Equations [106], the Euler equations of compressible flow [148, 149], and chemotaxis blow-up using the Keller-Segel system

$$u_t = u_{xx}+\chi(vu_x)_x, \qquad v_t = v_{xx}+u-v,$$

implemented in [43].

7 Summary

In this paper we have considered velocity-based adaptive meshing strategies based upon the movement of the nodes of a topologically-fixed finite element grid. Unlike other adaptive techniques, such as h-refinement (mesh subdivision) or p-refinement (finite element order enrichment), this strategy does not require nested finite element spaces and always has the same number of degrees of freedom. The adaptive element involves modifying the finite element space as a result of the changing location of the node points, upon which the underlying basis functions are defined. For the spatial semi-discretization of time-dependent PDEs this works very naturally by making the finite element space a

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

continuous function of time: the goal being to ensure that the trial space evolves in such a way that, at any instant in time, it is able to provide an accurate and efficient representation of the PDE solution.

The approach used is to prescribe the evolution of the trial space by imposing (time-varying) velocities on the nodes of the finite element mesh. The positions of the nodes as functions of time (and therefore the finite element trial space at any given time) are then determined by integration of these nodal velocities. Our concern has been to describe possible techniques for determining this nodal velocity field so as to ensure that the resulting finite element spaces locate, and evolve, their degrees of freedom in such a way as to permit the most accurate representations of the PDE solution.

Section 2 of the paper introduces the class of problems that we consider and discusses some of the properties of these PDEs that we may wish to be reflected in our adaptive numerical methods. These include mass conservation, scale-invariance and self-similarity. Consequences of the velocity-based approach are also discussed, in terms of the moving frame of reference that this induces and the effects of considering the governing equations, and their weak forms, in such moving frames. Section 3 then provides a short historical overview of some of the techniques for prescribing the velocity fields that define the resulting mesh evolution. These include physically-based algorithms derived from fluid flows and elastic deformations for example, as well as mathematically-based algorithms which seek either to directly minimize weighted residuals or to mimic certain mathematical properties of the equations of interest. In particular, methods based upon geometric conservation principles are described as a means of motivating the techniques discussed in the remainder of the paper.

Sections 4 onwards focus primarily on reviewing this particular class of method, initially for mass-conserving problems and then for non mass-conserving equations. Numerous examples are presented and a large number of applications are described, in varying levels of detail. The power of the approach is that it respects many of the physical properties of the PDEs that are being solved, including local and global mass conservation, scale-invariance and self-similarity when they are present. Furthermore the technique has been applied extensively to model problems in both one and two space dimensions.

It should be noted that the techniques discussed in this review are still far from mature and considerable further research is likely to be fruitful. In particular, implicit time-stepping strategies should increase significantly the size of the time-steps that may be taken, especially in the solution of fourth order (and higher) problems. Other developments that will be of potential significance include further extension of the techniques outlined in subsection 6.2 in order to permit a wider selection of monitor functions to be considered. This will open up the possibility of selecting different criteria upon which to base the evolution of the interior of the finite element mesh. Furthermore it is likely that, for many applications, the most effective solution strategies will involve the combination of mesh movement, such as those discussed here, with more conventional adaptive strategies which allow the mesh topology to be adjusted and permit the number of de-

grees of freedom to be adjusted at discrete points in time. The most effective ways of combining these different strategies is likely to be an important topic of future research.

Appendix

A Propagation of self-similar solutions

In Appendix A we show that under the local mass conservation principle (4.2) any function $u(t,\mathbf{x})>0$ which coincides with a self-similar solution of a scale-invariant mass conserving problem at an initial time $t=t_0$ and vanishes on the boundary is propagated as a self-similar solution for all time.

Theorem A.1. *Let*

1. $u(t,\mathbf{x})>0$ *be the solution of a scale-invariant mass-conserving problem in a domain* $\mathcal{R}(t)$ *such that* $u(t,\mathbf{x})$ *vanishes on the boundary* $\partial\mathcal{R}(t)$,
2. $u(t,\mathbf{x})$ *satisfy the local conservation principle*

$$\int_{\Omega(t)} u(t,\mathbf{x})d\mathbf{x}=\textit{constant in time}, \tag{A.1}$$

   of (4.2) *for all* $\Omega(t)\in\mathcal{R}(t)$,
3. *the velocity* $\mathbf{v}(t,\mathbf{x})$ *induced by* (A.1) *be irrotational,*
4. $u(t_0,\mathbf{x})$ *coincide with a self-similar solution at an initial time* $t=t_0$.

Then $u(t,\mathbf{x})$ *remains self-similar for all* $t>t_0$.

Proof. As a preliminary we show that at time $t=t_0$ the induced velocity $\mathbf{v}(t_0,\mathbf{x})$ has components

$$v_\nu(t_0,\mathbf{x})=\frac{\beta_\nu x_\nu}{t_0} \tag{A.2}$$

for all $\mathbf{x}=\{x_\nu\}$.

Since $u(t,\mathbf{x})$ satisfies the local Lagrangian conservation principle (A.1) for all $\Omega(t)$, it also satisfies the corresponding Eulerian conservation law

$$u_t+\nabla\cdot(u\mathbf{v})=0, \tag{A.3}$$

where $\mathbf{v}(t,\mathbf{x})$ is the induced velocity. Let the self-similar solution be

$$u^{ss}(t,\mathbf{x})=t^\gamma f(\boldsymbol{\xi}), \tag{A.4}$$

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

where $\boldsymbol{\xi}=\{\xi_\nu\}=\{x_\nu/t^{\beta_\nu}\}$ and γ,β_ν are the scaling indices for u,x_ν respectively. Then, since $u(t,\mathbf{x})$ coincides with $u^{ss}(t,\mathbf{x})$ at time $t=t_0$, substituting (A.4) at time $t=t_0$ into (A.3),

$$\gamma t_0^{\gamma-1}f(\boldsymbol{\xi}^0)-t_0^{\gamma-1}\sum_\nu \beta_\nu \xi_\nu^0 \frac{\partial f(\boldsymbol{\xi}^0)}{\partial \xi_\nu{}^0}+t_0^\gamma \nabla_{\mathbf{x}}\cdot\{f(\boldsymbol{\xi}^0)\mathbf{v}\}=0, \tag{A.5}$$

where $\boldsymbol{\xi}^0=\{\xi_\nu^0\}=\{x_\nu/t_0^{\beta_\nu}\}$. Also, since $\sum\beta_\nu+\gamma=0$ for a mass conserving problem, Eq. (A.5) reduces to

$$-t_0^{-1}\sum_\nu \beta_\nu \frac{\partial\{\xi_\nu^0 f(\boldsymbol{\xi}^0)\}}{\partial \xi_\nu^0}+\sum_\nu \frac{\partial}{\partial x_\nu}\{f(\boldsymbol{\xi}^0)v_\nu\}=0, \tag{A.6}$$

which can be written

$$\sum_\nu \frac{\partial}{\partial x_\nu}\Big\{f(\boldsymbol{\xi}^0)\Big(-\frac{\beta_\nu x_\nu}{t_0}+v_\nu\Big)\Big\}=0. \tag{A.7}$$

Moreover, since $\mathbf{v}$ is irrotational, introducing a velocity potential ϕ such that $\mathbf{v}=\nabla\phi$, Eq. (A.7) takes the form

$$\sum_\nu \frac{\partial}{\partial x_\nu}\Big\{f(\boldsymbol{\xi}^0)\frac{\partial}{\partial x_\nu}\Big(-\sum_\mu \frac{1}{2}\frac{\beta_\mu x_\mu^2}{t_0}+\phi\Big)\Big\}=0. \tag{A.8}$$

Then, since u (and therefore f) is positive in the interior of the domain and vanishes on the boundary, Eq. (A.8) has the unique solution

$$\phi=\frac{1}{2}\sum_\mu \frac{\beta_\mu x_\mu^2}{t_0}+\text{const}\Rightarrow v_\nu=\frac{\partial\phi}{\partial x_\nu}=\frac{\beta_\nu x_\nu}{t_0}. \tag{A.9}$$

Next we show that for all t in the interval $(t_0,t_0+\Delta t)$ the moving coordinate $\widehat{x}_\nu(t)$ varies as t^{β_ν} to order $\mathcal{O}(\Delta t)^2$. Writing $t=t_0+k\Delta t$, where $0<k\leq 1$, from a Taylor expansion of $\widehat{x}(t)$ and (A.9),

$$\begin{aligned}\widehat{x}_\nu(t)&=\widehat{x}_\nu(t_0)+k\Delta t v_\nu(t_0,x)+\mathcal{O}(\Delta t)^2=\widehat{x}_\nu(t_0)+k\Delta t\frac{\beta_\nu \widehat{x}_\nu(t_0)}{t_0}+\mathcal{O}(\Delta t)^2\\&=\Big(1+\frac{k\Delta t}{t_0}\Big)^{\beta_\nu}\widehat{x}_\nu(t_0)+\mathcal{O}(\Delta t)^2=\Big(\frac{t}{t_0}\Big)^{\beta_\nu}\widehat{x}_\nu(t_0)+\mathcal{O}(\Delta t)^2.\end{aligned} \tag{A.10}$$

It follows that

$$\frac{\widehat{x}_\nu(t)}{t^{\beta_\nu}}=\frac{\widehat{x}_\nu(t_0)}{t_0^{\beta_\nu}}+\mathcal{O}(\Delta t)^2 \tag{A.11}$$

for all $t_0<t\leq t_0+\Delta t$ and hence that $\widehat{x}_\nu(t)$ varies as t^{β_ν} in the interval $(t_0,t_0+\Delta t)$ to order $\mathcal{O}(\Delta t)^2$.

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

Now consider the identity

$$\int_{\Omega(t)} u(t,\mathbf{x})d\mathbf{x} - \int_{\Omega(t)} t^{\gamma} f\Big(\frac{x_1}{t^{\beta_1}},\frac{x_2}{t^{\beta_2}},\cdots,\frac{x_d}{t^{\beta_d}}\Big)d\mathbf{x}$$
$$\equiv \int_{\Omega(t_0)} u(t_0,\mathbf{x})d\mathbf{x} - \int_{\Omega(t_0)} t_0{}^{\gamma} f\Big(\frac{x_1}{t_0^{\beta_1}},\frac{x_2}{t_0^{\beta_2}},\cdots,\frac{x_d}{t_0^{\beta_d}}\Big)d\mathbf{x} + \mathcal{O}(\Delta t)^2, \tag{A.12}$$

which holds because the first integrals on each side of (A.12) are equal by local conservation, and the second integrals on each side are equal under the substitution $x_\nu/t^{\beta} = y_\nu/t_0^{\beta}$, to order $\mathcal{O}(\Delta t)^2$. Note that since the problem is mass-conserving $\sum\beta_\nu+\gamma=0$, and that the points of $\mathcal{R}(t)$ coincide with those of $\mathcal{R}(t_0)$ to order $\mathcal{O}(\Delta t)^2$ under the above substitution.

Since at $t=t_0$

$$u(t_0,\mathbf{x}) = t_0{}^{\gamma} f\Big(\frac{x_1}{t_0^{\beta_1}},\frac{x_2}{t_0^{\beta_2}},\cdots,\frac{x_d}{t_0^{\beta_d}}\Big) \tag{A.13}$$

and $\Omega(t_0)$ is arbitrary, it follows from (A.12) that

$$u(t,\mathbf{x}) - t^{\gamma} f\Big(\frac{x_1}{t^{\beta_1}},\frac{x_2}{t^{\beta_2}},\cdots,\frac{x_d}{t^{\beta_d}}\Big) = \mathcal{O}(\Delta t)^2 \tag{A.14}$$

for all t such that $t_0 < t \le t_0+\Delta t$. Thus at $t=t_1=t_0+\Delta t$

$$u(t_1,\mathbf{x}) - t_1^{\gamma} f\Big(\frac{x_1}{t_1^{\beta_1}},\frac{x_2}{t_1^{\beta_2}},\cdots,\frac{x_d}{t_1^{\beta_d}}\Big) = \mathcal{O}(\Delta t)^2, \tag{A.15}$$

so that $u(t_1,\mathbf{x})$ is the self-similar solution $u^{ss}(t_1,\mathbf{x})$ at time $t_1=t_0+\Delta t$ to order $\mathcal{O}(\Delta t)^2$. Repeating the whole argument at time $t=t_1$, with an additional $\mathcal{O}(\Delta t)$ term in Eqs. (A.5) to (A.9), we can deduce in the same way that at time $t_2=t_1+\Delta t=t_0+2\Delta t$

$$u(t_2,\mathbf{x}) - t_2^{\gamma} f\Big(\frac{x_1}{t_2^{\beta_1}},\frac{x_2}{t_2^{\beta_2}},\cdots,\frac{x_d}{t_2^{\beta_2}}\Big) = \mathcal{O}(\Delta t)^2. \tag{A.16}$$

After n such repetitions we find that at a *fixed* time $t_n=t_0+n\Delta t$

$$u(t_n,\mathbf{x}) - t_n^{\gamma} f\Big(\frac{x_1}{t_n^{\beta_1}},\frac{x_2}{t_n^{\beta_2}},\cdots,\frac{x_d}{t_n^{\beta_2}}\Big) = \mathcal{O}(\Delta t). \tag{A.17}$$

Finally, letting $n\to\infty$, $\Delta t\to 0$ it follows that $u(t,\mathbf{x})$ is the self-similar solution for all time. This completes the proof. □

As a corollary, the components of $\mathbf{v}(t,\mathbf{x})$ are

$$v_\nu(t,\mathbf{x}) = \frac{\beta_\nu x_\nu}{t} \tag{A.18}$$

for all t and all x_ν.

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

B Propagation of L_2 projections of self-similar solutions

We now show that, under the similarity velocity (2.10), initial data which coincides with the L_2 projection of a self-similar solution at time $t=t_0$ is propagated as the L_2 projection of the self-similar solution for all time.

Theorem B.1. *Let*

1. $u(t,\mathbf{x})$ *be a positive solution of a mass-conserving scale-invariant problem governed by the PDE* (2.1),
2. $U(t,\mathbf{x})$ *satisfy the conservation principle* (4.46),
3. *the velocity be the similarity velocity* (2.10) *for all* t,
4. $U(t_0,\mathbf{x})$ *coincide with the* L_2 *projection of the self-similar solution* (2.11) *at time* $t=t_0$.

Then $U(t,\mathbf{x})$ *remains the* L_2 *projection of the self-similar solution for all* $t>t_0$.

Proof. This proof is much simpler than that in Appendix A because we have assumed that the velocity is the self-similar velocity (2.10) for all t.

By assumption

$$\int_{\mathcal{R}(t)} W_i U(t,\mathbf{x})d\mathbf{x} \tag{B.1}$$

is constant in time, and by the substitution $x_\nu = t^{\beta_\nu}\xi_\nu$

$$\int_{\mathcal{R}(t)} W_i t^\gamma f(\boldsymbol{\xi})d\mathbf{x} \tag{B.2}$$

is also constant in time, since $\mathcal{R}(t)$ and W_i move with the self-similar velocity (2.10) and for mass conserving problems $\sum\beta_\nu+\gamma=0$. Therefore

$$\int_{\mathcal{R}(t)} W_i\{U(t,\mathbf{x})-t^\gamma f(\boldsymbol{\xi})\}d\mathbf{x} \tag{B.3}$$

is constant in time. Because U is the L_2 projection of the self-similar solution (2.11) at time t_0 the integral (B.3) is zero at time $t=t_0$. Hence it remains zero for all time and U remains the L_2 projection of (2.11) for all time. This completes the proof. □

References

[1] R.Abgrall, Residual distribution schemes: current status and future trends, Comput. Fluids, 214 (2006), 773–808.
[2] M. Ahamadi and O. G.Harlen, A Lagrangian finite element method for simulation of a suspension under planar extensional flow, J. Comput. Phys., 227 (2008), 7543–7560.
[3] D. Aronson, The porous medium equation, in Nonlinear Diffusion Problems, Lect. Notes Math., 1224 (1986), 1–46.

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

[4] T. A. Baer, R. A. Cairncross, P. R. Schunk, R. R. Rao and P. A. Sackinger, A finite element method for free surface flows of incompressible fluids in three dimensions, part II: dynamic wetting lines, Int. J. Numer. Meth. Fluids, 33 (2000), 405–427.
[5] M. J.Baines, Moving Finite Elements, Oxford University Press, 1994.
[6] M. J. Baines, Grid adaptation via node movement, Appl. Numer. Math., 26 (1998), 77–96.
[7] M. J. Baines, M. E. Hubbard and P. K. Jimack, A moving mesh finite element algorithm for the adaptive solution of time-dependent partial differential equations with moving boundaries, Appl. Numer. Math., 54 (2005), 450–469.
[8] M. J. Baines, M. E. Hubbard and P. K. Jimack, A moving mesh finite element algorithm for fluid flow problems with moving boundaries, Int. J. Numer. Meth. Fluids, 47(10/11) (2005), 1077–1083.
[9] M. J. Baines, M. E. Hubbard, P. K. Jimack and A. C. Jones, Scale-invariant moving finite elements for nonlinear partial differential equations in two dimensions, Appl. Numer. Math., 56 (2006), 230–252.
[10] M. J. Baines, M. E. Hubbard, P. K. Jimack and R. Mahmood, A moving-mesh finite element method and its application to the numerical solution of phase-change problems, Commun. Comput. Phys., 6(3) (2009), 595–624.
[11] M. J. Baines and A. J. Wathen, Moving finite element modelling of compressible flow, Appl. Numer. Math., 2 (1986), 495–514.
[12] M. J. Baines and A. J. Wathen, Moving finite elements for evolutionary problems, I, theory, J. Comput. Phys., 79 (1988), 245–269.
[13] N. J. Balmforth, A. S. Burbidge, R. V. Craster, J. Salzig and A. Shen, Visco-plastic models of isothermal lava domes, J. Fluid Mech., 403 (2000), 37–65.
[14] G. I. Barenblatt, Scaling, Self-Similarity and Intermediate Asymptotics, Cambridge University Press, 1996.
[15] G. I. Barenblatt, Scaling, Cambridge University Press, 2003.
[16] J. W. Barrett, S. Langdon and R. Nurnberg, Finite element approximation of a sixth-order nonlinear degenerate parabolic equation, Numer. Math., 96 (2004), 401–434.
[17] A. Barlow, A compatible finite element multi-material ALE hydrodynamics algorithm, Int. J. Numer. Meth. Fluids, 56 (2008), 953–964.
[18] A. E. Berger, M. Ciment and J. C. W. Rogers, Numerical solution of a diffusion consumption problem with a free boundary, SIAM J. Numer. Anal., 12 (1975), 646–672.
[19] B. Bhattacharya, A Moving Finite Element Method for High Order Nonlinear Diffusion Problems, MSc dissertation, Department of Mathematics, University of Reading, UK, 2006.
[20] K. W. Blake, Moving Mesh Methods for Nonlinear Partial Differential Equations, PhD thesis, Department of Mathematics, University of Reading, UK, 2001.
[21] K. W. Blake and M. J. Baines, Moving mesh methods for nonlinear partial differential equations, Numerical Analysis Report 7/01, Dept of Mathematics, University of Reading, UK, 2001.
[22] J. F. Blowey, J. R. King and S. Langdon, Small and waiting time behaviour of the thin film equation, SIAM J. Appl. Math., 67 (2007), 1776–1807.
[23] P. Bochev, G. Liao and G. de la Pena, Analysis and computation of adaptive moving grids by deformation, Numer. Meth. Part. Diff. Equations, 12 (1998), 489–506.
[24] D. Boffi and L. Gastaldi, Stability and geometric conservation laws for ALE formulations, Comput. Method. Appl. Math., 193 (2004), 4717–4739.
[25] C. Bonacina, G. Comini, A. Fasano and M. Primicerio, Numerical solution of phase-change problems, Int. J. Heat Mass Trans., 16 (1973), 1825–1832.

[26] C. de Boor, Good approximation by splines with variable knots II, Springer Lecture Notes Series, 363 (1973).
[27] M. de Berg, O. Cheong, M. van Kreveld and M. Overmars, Computational Geometry: Algorithms and Applications, Springer-Verlag, 2008.
[28] C. J. Budd and M. D. Piggott, The geometric integration of scale invariant ordinary and partial differential equations, J. Comput. Appl. Math., 128 (2001), 399–422.
[29] C. J. Budd, W. Huang and R. D. Russell, Moving mesh methods for problems with blow-up, SIAM J. Sci. Comput., 17 (1996), 305–327.
[30] C. J. Budd, W. Huang and R. D. Russell, Adaptivity with moving grids, Acta. Numer., 18 (2009), 111–241.
[31] C. Budd, G. Collins, W. Huang and R. D. Russell, Self-similar numerical solutions of the porous medium equation using moving mesh methods, Phil. Trans. Roy. Soc., 357 (1999), 1047–1078.
[32] X.-X.Cai, B. Jiang and G. Liao, Adaptive grid generation based on least-squares finite-element method, Comput. Math. Appl., 48 (2004), 1077–1085.
[33] R. A. Cairncross, P. R. Schunk, T. A. Baer, R. R. Rao and P. A. Sackinger, A finite element method for free surface flows of incompressible fluids in three dimensions, part I, boundary fitted mesh motion, Int. J. Numer. Meth. Fluids, 33 (2000), 375–403.
[34] W. M. Cao, W. Z. Huang and R. D. Russell, A moving-mesh method based on the geometric conservation law, SIAM J. Sci. Comput., 24 (2002), 118–142.
[35] W. M. Cao, W. Z. Huang and R. D. Russell, Approaches for generating moving adaptive meshes: location versus velocity, Appl. Numer. Math., 47(2) (2003), 121–138.
[36] E. J. Caramana and M. J. Shashkov, Elimination of artificial grid distortion and hourglasstype motions by means of Lagrangian subzonal masses and pressures, J. Comput. Phys., 142 (1998), 521–561.
[37] N. N. Carlson and K. Miller, Design and application of a gradient-weighted moving finite element code I: in one dimension, SIAM J. Sci. Comput., 19 (1998), 728–765.
[38] N. N. Carlson and K. Miller, Design and application of a gradient-weighted moving finite element code II: in two dimensions, SIAM J. Sci. Comput., 19 (1998), 766–798.
[39] M. do Carmo Coimbra, C. Serano and A. Rodrigues, A moving finite element method for the solution of two-dimensional time-dependent models, Appl. Numer. Math., 44 (2003), 449–469.
[40] M. do Carmo Coimbra, C. Serano and A. Rodrigues, Moving finite element method: applications to science and engineering problems, Comput. Chem. Eng., 28 (2004), 597–603.
[41] M.-Y. Chu, H.-M. Chen, C.-Y. Hsieh, T.-H. Lin, H.-Y. Hsiao, G. Liao and Q. Peng, Adaptive grid generation based non-rigid image registration using mutual information for breast MRI, J. Sign. Process. Syst., 54 (2009), 45–63.
[42] C. J. van der Veen, Fundamentals of Glacier Dynamics, A. A Balkema, 1999.
[43] S. L. Cole, Blow-Up in a Chemotaxis Model Using a Moving Mesh Method, MSc dissertation, Department of Mathematics, University of Reading, UK, 2009.
[44] J. Cote, S. Gravel and A. Staniforth, A generalized family of schemes that eliminate the spurious resonant response of semi-Lagrangian schemes to orographic forcing, Mon. Weather Rev., 123 (1995), 3605–3613.
[45] J. Cote, M. Roch, A. Staniforth and L. Fillion, A variable-resolution semi-Lagrangian finite-element global-model of the shallow-water equations, Mon. Weather Rev., 121 (1993), 231–243.
[46] C. L. Cox, W. F. Jones, V. L. Quisenberry and F. Yo, One-dimensional infiltration with mov-

ing finite elements and improved soil water diffusivity, Water Resour. Res., 30 (1994), 1431–1438.
[47] J. Crank, Free and Moving Boundary Problems, Oxford University Press, 1984.
[48] J. Crank and R. S. Gupta, A moving boundary problem arising from the diffusion of oxygen in absorbing tissue, J. Int. Math. Appl., 10 (1972), 19–33.
[49] B. Dacorogna and J. Moser, On a PDE involving the Jacobian determinant, Ann. I. H. Poincaré C, 7 (1990), 1–26.
[50] H. Deconinck and M. Ricchiuto, Residual distribution schemes: foundation and analysis, in Encyclopedia of Computational Mechanics, Volume 3: Fluids, E. Stein, E. de Borst, T. J. Hughes (Eds.), John Wiley and Sons, Ltd., 2007.
[51] G. L. Delzanno, L. Chacón, J. M. Finn, Y. Chung and G. Lapenta, An optimal robust equidistribution method for two-dimensional grid adaptation based on Monge-Kantorovich optimization, J. Comput. Phys., 227 (2008), 9841–9864.
[52] I. Demirdžić and M. Perić, Space conservation law in finite volume calculations of fluid flow, Int. J. Numer. Meth. Fluids, 8(9) (1988), 1037–1050.
[53] I. Demirdžić and M. Perić, Finite volume method for prediction of fluid flow in arbitrarily shaped domains with moving boundaries, Int. J. Numer. Meth. Fluids, 10(7) (1990), 771–790.
[54] B. Duarte, Moving finite elements method applied to the solution of front reaction models: causticizing reaction, Comput. Chem. Eng., 19 (1995), S421–S426.
[55] B. P. M. Duarte and C. M. S. G. Baptista, Moving finite elements method applied to dynamic population balance equations, AIChE J., 54 (2008), 673–692.
[56] S. Étienne, A. Garon and D. Pelletier, Perspective on the geometric conservation law and finite element methods for ALE simulations of incompressible flow, J. Comput. Phys., 228(7) (2009), 2313–2333.
[57] C. Farhat, P. Geuzaine and C. Grandmont, The discrete geometric conservation law and the nonlinear stability of ALE schemes for the solution of flow problems on moving grids, J. Comput. Phys., 174(2) (2001), 669–694.
[58] L. Formaggia and F. Nobile, Stability analysis of second-order time accurate schemes for ALE-FEM, Comput. Method. Appl. Math., 134 (1996), 71–90.
[59] P. Garabedian, Partial Differential Equations, Wiley, New York, 1964.
[60] W. Gardner and J. A. Widtsoe, The movement of soil moisture, Soil Sci., 11 (1921), 215–232.
[61] P. H. Gaskell, P. K. Jimack, M. Sellier, H. M.Thompson and M. C. T. Wilson, Gravity-driven flow of continuous thin liquid films on non-porous substrates with topography, J. Fluid Mech., 509 (2004), 253–280.
[62] R. J. Gelinas, S. K. Doss and K. Miller, The moving finite element method: applications to general partial differential equations with multiple large gradients, J. Comput. Phys., 40 (1981), 202–249.
[63] P. Geuzaine, C. Grandmont and C. Farhat, Design and analysis of ALE schemes with provable second-order time-accuracy for inviscid and viscous flow simulations, J. Comput. Phys., 191(1) (2003), 206–227.
[64] F. X. Giraldo, The Lagrange-Galerkin method for the two-dimensional shallow water equations on adaptive grids, Int. J. Numer. Meth. Fluids, 33(6) (2000), 789–832.
[65] G. Gottardi and M. Venutelli, Moving finite element model for one-dimensional infiltration in unsaturated soil, Water Resour. Res., 28 (1992), 3259–3267.
[66] G. Gottardi and M. Venutelli, One-dimensional moving finite element model of solute transport, Ground Water, 32 (1994), 645–649.

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

[67] M. Grajewski, M. Köster and S. Turek, Mathematical and numerical analysis of a robust and efficient grid deformation method in the finite element context, SIAM J. Sci. Comput., 31(2) (2009), 1539–1557.
[68] D. F. Griffiths, Introduction to Electrodynamics, Prentice-Hall, New Jersey, 1999.
[69] H. Guillard and C. Farhat, On the significance of the geometric conservation law for flow computations on moving meshes, Comput. Method. Appl. Math., 190 (2000), 1467–1482.
[70] O. G. Harlen, J. M. Rallinson and P. Szabo, A split Lagrangian-Eulerian method for simulating transient viscoelastic flows, J. Non-Newton. Fluid, 60 (1995), 81–104.
[71] M. Heil, An efficient solver for the fully coupled solution of large-displacement fluid-structure interaction problems, Comput. Method. Appl. Math., 193 (2004), 1–23.
[72] B. M. Herbst, S. W. Schoombie and A. R. Mitchell, A moving Petrov-Galerkin method for transport equations, Int. J. Numer. Meth. Eng., 18 (1982), 1321–1336.
[73] S. D. Howison, D. F. Mayers and W. R. Smith, Numerical and asymptotic solution of a sixth-order nonlinear diffusion equation and related coupled systems, IMA J. Appl. Math., 57 (1996), 79–98.
[74] W. Huang, Variational mesh adaptation: isotropy and equidistribution, J. Comput. Phys., 174 (2001), 903–924.
[75] M. E. Hubbard, M. J. Baines and P. K. Jimack, Consistent Dirichlet boundary conditions for moving boundary problems, Appl. Numer. Math., 59(6) (2009), 1337–1353.
[76] P. K. Jimack, A best approximation property of the moving finite element method, SIAM J. Numer. Anal., 33 (1996), 2206–2232.
[77] P. K. Jimack, Optimal eigenvalue and asymptotic large time approximations using the moving finite element method, IMA J. Numer. Anal., 16 (1996), 381–398.
[78] P. K. Jimack, An optimal finite element mesh for elastostatic structural analysis problems, Comput. Struct., 64 (1997), 197–208.
[79] P. K. Jimack and A. J. Wathen, Temporal derivatives in the finite-element method on continuously deforming grids, SIAM J. Numer. Anal., 28 (1991), 990–1003.
[80] I. W. Johnson and A. W. Jayawardena, Efficient numerical solution of the dispersion equation using moving finite elements, Finite Elem. Anal. Des., 28 (1998), 241–253.
[81] I. W. Johnson, M. J. Baines and A. J. Wathen, Moving finite elements for evolutionary problems, II, applications, J. Comput. Phys., 79 (1988), 270–297.
[82] S. F. Kistler and L. E. Scriven, Coating flow theory by element and asymptotic analysis of the Navier-Stokes system, Int. J. Numer. Meth. Fluids, 4 (1984), 207–229.
[83] E. Kuhl, S. Hulshoff and R. de Borst, An arbitrary Lagrangian Eulerian finite-element approach for fluid-structure interaction phenomena, Int. J. Numer. Meth. Eng., 57 (2003), 117–142.
[84] H. K. Kuiken, Viscous sintering: the surface-tension-driven flow of a liquid form under the influence of curvature gradients at its surface, J. Fluid Mech., 214 (1990), 503–515.
[85] A. Lacey, J. Ockendon and A. Tayler, Waiting-time solutions of a nonlinear diffusion equation, SIAM J. Appl. Math., 42 (1982), 1252–1264.
[86] G. Liao and D. Anderson, A new approach to grid generation, Appl. Anal., 44 (1992), 285–298.
[87] G. Liao, Z. Lei and G. C. de la Pena, Adaptive grids for resolution enhancement, Shock Waves, 12 (2002), 153–156.
[88] G. Liao, T.-W. Pan and J. Su, A numerical grid generator based on Moser's deformation method, Numer. Meth. Part. Diff. Equations, 10 (1994), 21–31.
[89] G. Liao and J. Su, A direct method in Dacorogna-Moser's approach of grid generation,

Appl. Anal., 49 (1993), 73–84.
[90] G. Liao and J. Su, A moving grid method for (1+1) dimension, Appl. Math. Lett., 8 (1995), 47–49.
[91] G. Liao and J. Xue, Moving meshes by the deformation method, J. Comput. Appl. Math., 195 (2006), 83–92.
[92] F. Liu, S. Ji and G. Liao, An adaptive grid method with cell-volume control and its application to Euler flow calculations, SIAM J. Sci. Comput., 20(3) (1998), 811–825.
[93] D. L. Marcum and N. P. Weatherill, Unstructured grid generation using iterative point insertion and local reconnection, AIAA J., 33 (1995) 1619–1625.
[94] J. I. Martinez-Herrera and J. J. Derby, Analysis of capillary-driven viscous flows during the sintering of ceramic powders, AIChE J., 40 (1984), 1794–1803.
[95] F. Mashayek and N. Ashgriz, A spine-flux method for simulating free surface flows, J. Comput. Phys., 122 (1995), 367–399.
[96] D. J. Mavriplis and Z. Yang, Construction of the discrete geometric conservation law for high-order time-accurate simulations on dynamic meshes, J. Comput. Phys., 213(2) (2006), 557–573.
[97] P. A. Mendes and F. A. Branco, Analysis of fluid-structure interaction by an arbitrary Lagrangian-Eulerian finite element formulation, Int. J. Numer. Meth. Fluids, 30 (1999), 897–919.
[98] J. W. Mercer and R. M. Cohen, A review of immiscible fluids in the subsurface: properties, models, characterization and remediation, J. Contaminant Hydrogeology, 6 (1990), 107–163.
[99] C. Michler, H. De Sterck and H. Deconinck, An arbitrary Lagrangian Eulerian formulation for residual distribution schemes on moving grids, Comput. Fluids, 32 (2003), 59–71.
[100] K. Miller, Moving finite elements, II, SIAM J. Numer. Anal., 18 (1981), 1033–1057.
[101] K. Miller, A geometrical-mechanical interpretation of gradient-weighted moving finite elements, SIAM J. Numer. Anal., 34 (1997), 67–90.
[102] K. Miller, Stabilized moving finite elements for convection dominated problems, J. Sci. Comput., 24 (2005), 163–182.
[103] K. Miller and M. J. Baines, Least squares moving finite elements, IMA J. Numer. Anal., 21 (2001), 621–642.
[104] K. Miller and R. N. Miller, Moving finite elements, I, SIAM J. Numer. Anal., 18 (1981), 1019–1032.
[105] J. A. Mackenzie and M. L. Robertson, The numerical solution of one-dimensional phase change problems using an adaptive moving mesh method, J. Comput. Phys., 161 (2000), 537–557.
[106] G. Morrison, Numerical Modelling of Tidal Bores Using a Moving Mesh, MSc dissertation, Department of Mathematics, University of Reading, UK, 2008.
[107] J. Moser, Volume elements of a Riemann manifold, T. Am. Math. Soc., 120 (1965), 286–294.
[108] F. Muttin, T. Coupez, M. Bellet and J. L. Chenot, Lagrangian finite-element analysis of time-dependent free-surface flow using an automatic remeshing technique-application to metal casting, Int. J. Numer. Meth. Eng., 36 (1993), 2001–2015.
[109] J. R. Ockendon, The role of the Crank-Gupta model in the theory of free and moving boundary problems, Adv. Comput. Math., 6 (1996), 281–293.
[110] K. Osman, Numerical Schemes for a Non-Linear Diffusion Problem, MSc dissertation, Department of Mathematics, University of Reading, UK, 2005.
[111] J. Parrinello, Modelling Water Uptake in Rice Using Moving Meshes, MSc dissertation,

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press

Department of Mathematics, University of Reading, UK, 2008.
[112] D. Partridge and M. J. Baines, A moving mesh approach to an ice sheet model, Comput. Fluids, 46 (2011), 381–386.
[113] R. E. Pattle, Diffusion from an instantaneous point source with a concentration-dependent coefficient, Q. J. Mech. Appl. Math., 12 (1959), 407–409.
[114] P.-O. Persson, J. Bonet and J. Peraire, Discontinuous Galerkin solution of the Navier-Stokes equations on deformable domains, Comput. Method. Appl. Math., 198 (2009), 1585–1595.
[115] R. C. Peterson, P. K. Jimack and M. A. Kelmanson, The solution of two-dimensional free-surface problems using automatic mesh generation, Int. J. Numer. Meth. Fluids, 31 (1999), 937–960.
[116] M. D. Piggott, G. J. Gorman, C. C. Pain, P. A. Allison, A. S. Candy, B. T. Martin and M. R. Wells, A new computational framework for multi-scale ocean modelling based on adapting unstructured meshes, Int. J. Numer. Meth. Fluids, 56 (2008), 1003–1015.
[117] C. P. Please and P. K. Sweby, A transformation to assist numerical solution of diffusion equations, Numerical Analysis Report 5/86, Department of Mathematics, University of Reading, UK, 1986.
[118] B. Ramaswamy and M. Kawahara, Lagrangian finite-element analysis applied to viscous free-surface fluid flow, Int. J. Numer. Meth. Fluids, 7 (1987), 953–984.
[119] R. Roberts, Modelling Glacier Flow, MSc dissertation, Department of Mathematics, University of Reading, UK, 2007.
[120] N. Robertson, A Moving Lagrangian Mesh Model of a Lava Dome Volcano and Talus Slope, MSc dissertation, Department of Mathematics, University of Reading, UK, 2006.
[121] H. Saito and L. E. Scriven, Study of coating flow by the finite element method, J. Comput. Phys., 42 (1981), 53–76.
[122] P. H. Saksono, W. G. Dettmer and D. Peric, An adaptive remeshing strategy for fluid flows with moving boundaries and fluid-structure interaction, Int. J. Numer. Meth. Eng., 71 (2007), 1009–1050.
[123] A. Schmidt, Computation of three dimensional dendrites with finite elements, J. Comput. Phys., 125 (1996), 293–312.
[124] B. Semper and G. Liao, A moving grid finite element method using grid deformation, Numer. Meth. Part. Diff. Equations, 11 (1995), 603–615.
[125] C. Serano, A. Rodrigues and J. Villadsen, The moving finite element method with polynomial approximation of any degree, Comput. Chem. Eng., 15 (1992), 25–33.
[126] C. Serano, A. Rodrigues and J. Villadsen, Solution of partial differential equations systems by the moving finite element method, Comput. Chem. Eng., 16 (1992), 583–592.
[127] H. M. Shodja and J. R. Feldkamp, Numerical analysis of sedimentation and consolidation by the moving finite element method, Int. J. Numer. Anal. Met., 17 (1993), 753–769.
[128] N. F. Smyth and J. M. Hill, High order nonlinear diffusion, IMA J. Appl. Math., 40 (1988), 73–86.
[129] A. Soulaimani and Y. Saad, An arbitrary Lagrangian-Eulerian finite element method for solving three-dimensional free surface flows, Comput. Method. Appl. Math., 162 (1998), 70–106.
[130] A. Staniforth and J. Cote, Eliminating the interpolation associated with the semi-lagrangian scheme, Mon. Weather Rev., 114 (1986), 135–146.
[131] G. Strang and G. Fix, An Analysis of the Finite Element Method, Cambridge University Press, 2nd edition, 2008.
[132] J. Stojsavljevic, Investigation of Waiting Times in Nonlinear Diffusion Equations Using a

Moving Mesh Method, MSc dissertation, Department of Mathematics, University of Reading, UK, 2007.
[133] P. Szabo and O. Hassager, Simulation of free surfaces in 3-d with the arbitrary Lagrange-Euler method, Int. J. Numer. Meth. Eng., 38 (1995), 717–734.
[134] T. Tang, Moving mesh computations for computational fluid dynamics, in Recent Advances in Adaptive Computations, Vol. 383 of Contemporary Mathematics, AMS, 141–173.
[135] P. D. Thomas and C. K. Lombard, The geometric conservation law and its application to flow computations on moving grids, AIAA J., 17 (1979), 1030–1037.
[136] E. F. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction, Springer, Berlin, 2005.
[137] E. F. Toro and P. Garcia-Navarro, Godunov-type methods for free-surface shallow flows: a review, J. Hydraul. Res., 45 (2007), 736–751.
[138] Y. Tourigny and M. J. Baines, Analysis of an algorithm for generating locally optimal meshes for approximation by discontinuous piecewise polynomials, Math. Comput., 66 (1997), 623–650.
[139] J. G. Trulio and K. R. Trigger, Numerical solution of the one-dimensional hydrodynamic equations in an arbitrary time-dependent coordinate system, Technical Report UCLR-6522, University of California Lawrence Radiation Laboratory, 1961.
[140] A. Twigger, Blow-up in the Nonlinear Schrodinger Equation Using an Adaptive Mesh Method, MSc dissertation, Department of Mathematics, University of Reading, UK, 2008.
[141] J. J. W. van der Vegt and H. van der Ven, Space-time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flows: I, general formulation, J. Comput. Phys., 182(2) (2002), 546–585.
[142] J. L. Vazquez, The Porous Medium Equation, Oxford Science Publications, 2007.
[143] A. Wacher and I. Sobey, String gradient weighted moving finite elements in multiple dimensions with applications in two dimensions, SIAM J. Sci. Comput., 29 (2007), 459–480.
[144] M. A. Walkley, P. H. Gaskell, P. K. Jimack, M. A. Kelmanson and J. L. Summers, Finite element simulation of three-dimensional free-surface flow problems, J. Sci. Comput., 24 (2005), 147–162.
[145] M. A. Walkley, P. H. Gaskell, P. K. Jimack, M. A. Kelmanson and J. L. Summers, Finite element simulation of three-dimensional free-surface flow problems with dynamic contact lines, Int. J. Numer. Meth. Fluids, 47 (2005), 1353–1359.
[146] D. Wan and S. Turek, Fictitious boundary and moving mesh methods for the numerical simulation of rigid particulate flows, J. Comput. Phys., 222 (2007), 28–56.
[147] L. R. Wang and M. Ikeda, A Lagrangian description of sea ice dynamics using the finite element method, Ocean Model., 7 (2004), 21–38.
[148] B. V. Wells, A Moving Mesh Finite Element Method for the Numerical Solution of Partial Differential Equations and Systems, PhD thesis, Department of Mathematics, University of Reading, UK, 2005.
[149] B. V. Wells, M. J. Baines and P. Glaister, Generation of Arbitrary Lagrangian-Eulerian (ALE) velocities, based on monitor functions, for the solution of compressible fluid equations, Int. J. Numer. Meth. Fluids, 47 (2005), 1375–1381.
[150] P. Wesseling, Computational Fluid Dynamics, Springer, 2000.
[151] H. Zhou and J. J. Derby, An assessment of a parallel, finite element method for three-dimensional, moving-boundary flows driven by capillarity for simulation of viscous sintering, Int. J. Numer. Meth. Fluids, 36 (2001), 841–865.

https://doi.org/10.4208/cicp.201010.040511a Published online by Cambridge University Press