Geochemistry Geophysics Geosystems G3
AN ELECTRONIC JOURNAL OF THE EARTH SCIENCES
Published by AGU and the Geochemical Society

Article
Volume 8, Number 5
10 May 2007
Q05010, doi:10.1029/2006GC001470
ISSN: 1525-2027

Investigations into the applicability of adaptive finite element methods to two-dimensional infinite Prandtl number thermal and thermochemical convection

D. R. Davies
School of Earth, Ocean and Planetary Sciences, Cardiff University, Park Place, Cardiff CF10 3YE, Wales, UK (daviesdr3@cardiff.ac.uk)

Civil and Computational Engineering Center, School of Engineering, Swansea University, Singleton Park, Swansea SA2 8PP,Wales, UK (engdavdr@swansea.ac.uk)

J. H. Davies
School of Earth, Ocean and Planetary Sciences, Cardiff University, Park Place, Cardiff CF10 3YE, Wales, UK (daviesjh2@cardiff.ac.uk)

O. Hassan, K. Morgan, and P. Nithiarasu
Civil and Computational Engineering Center, School of Engineering, Swansea University, Singleton Park, Swansea SA2 8PP, Wales, UK (o.hassan@swansea.ac.uk; k.morgan@swansea.ac.uk; p.nithiarasu@swansea.ac.uk)

[1] An adaptive finite element procedure is presented for improving the quality of solutions to convection-dominated problems in geodynamics. The method adapts the mesh automatically around regions of high solution gradient, yielding enhanced resolution of the associated flow features. The approach requires the coupling of an automatic mesh generator, a finite element flow solver, and an error estimator. In this study, the procedure is implemented in conjunction with the well-known geodynamical finite element code ConMan. An unstructured quadrilateral mesh generator is utilized, with mesh adaptation accomplished through regeneration. This regeneration employs information provided by an interpolation-based local error estimator, obtained from the computed solution on an existing mesh. The technique is validated by solving thermal and thermochemical problems with well-established benchmark solutions. In a purely thermal context, results illustrate that the method is highly successful, improving solution accuracy while increasing computational efficiency. For thermochemical simulations the same conclusions can be drawn. However, results also demonstrate that the grid-based methods employed for simulating the compositional field are not competitive with the other methods (tracer particle and marker chain) currently employed in this field, even at the higher spatial resolutions allowed by the adaptive grid strategies.

Components: 10,911 words, 17 figures, 11 tables.

Keywords: convection; adaptivity; finite element methods; geodynamics; error estimation; mantle.

Index Terms: 0545 Computational Geophysics: Modeling (4255); 0560 Computational Geophysics: Numerical solutions (4255); 1213 Geodesy and Gravity: Earth's interior: dynamics (1507, 7207, 7208, 8115, 8120).

Received 5 September 2006; **Revised** 17 January 2007; **Accepted** 31 January 2007; **Published** 10 May 2007.

Davies, D. R., J. H. Davies, O. Hassan, K. Morgan, and P. Nithiarasu (2007), Investigations into the applicability of adaptive finite element methods to two-dimensional infinite Prandtl number thermal and thermochemical convection, *Geochem. Geophys. Geosyst.*, *8*, **Q05010**, doi:10.1029/2006GC001470.

1. Introduction

[2] Over recent decades, numerical modeling has stirred significant interest in the geodynamical community, leading the way in studies of numerous geological processes. This interest is due to the fact that analytical solutions to the various phenomena are normally unavailable, while experimental methods are sometimes time-consuming and often have limitations. The numerical methods that have been employed have generally been based upon finite-difference [e.g., *McKenzie et al.*, 1974; *Bodri and Bodri*, 1978; *Matyska and Yuen*, 2001] and, occasionally, finite-volume techniques [*Tackley*, 1996, 1998; *Ratcliff et al.*, 1998; *Albers and Christensen*, 2001], although recently, the finite element method has become more established [e.g., *Baumgardner*, 1985; *Farnetani and Richards*, 1995; *Sidorin et al.*, 1998; *Zhong*, 2000, 2006]. Methods based upon the finite-element approach are attractive since they lead to general-purpose computer codes. However, it is not this feature that has been largely responsible for the recent interest shown in the method, but the fact that it, as indeed is the finite volume method, is based upon an integral formulation and hence is readily implemented on arbitrary discretizations, i.e., unstructured grids. This final point is central to our study.

[3] It is a well known fact that even the use of sophisticated computational models can give inaccurate results, if the numerical grid upon which the model is based is unable to capture the significant features of the problem. Indeed, for the large-scale problems encountered in geodynamics, inadequate grid resolution has become a major concern. The majority of phenomena studied (e.g., subduction zone and mid-ocean ridge magmatism) are characterized by the interaction of complex geometries, complex material properties and complex boundary conditions. Such a combination often yields unpredictable and intricate solutions, with narrow regions of high solution gradient frequently found embedded in large areas where the solution varies slowly. It is these high gradient regions that present a serious challenge for computational methods: their location and extent is very difficult to determine a priori and, even if their location is identified, with current methods it is often impossible to resolve localized features. The net result is that achieving accurate solutions is a very demanding task for the analyst. Indeed, one of the most challenging problems currently facing geodynamicists is the accurate solution of such ‘‘multi-scale’’ flows.

[4] This issue of accuracy must be balanced with computational considerations. An accurate solution requires that one properly resolves the active features in the simulation. Resolution, in turn, is related to the number of nodes employed. Obviously, one could generate a solution of high accuracy by employing an extremely fine mesh throughout the computational domain. However, the larger the number of nodes, or degrees of freedom, the greater the demands on computational memory and processing power. Finding the right balance between *accuracy* and computational *efficiency* is therefore a difficult task. Ideally, what is needed is a method capable of yielding an accurate solution, while employing as few degrees of freedom as possible.

[5] Standard methods have attempted to achieve this by utilizing nonuniform grids generated a priori, with the user exploiting previous experience to define the grid [e.g., *Davies and Stevenson*, 1992; *Scott*, 1992]. Such methods, however, are not applicable to unsteady problems, since the active regions within the solution domain are constantly mobile and predicting their location at any given time is an impossible task. The majority of previous studies have overcome this issue by employing a uniformly fine grid throughout the computational domain, as described above [e.g., *Oldham and Davies*, 2004; *Bunge et al.*, 2003]. This ensures that, as the simulation evolves, active regions are continually in zones of fine resolution and solution accuracy is maintained. Other methods have also been utilized, although more rarely, including both Lagrangian and Arbitrary Lagrangian-Eulerian (ALE) formulations [*Fullsack*, 1995]. However, over time, it has become apparent that these methods have their own restrictions and, consequently, a major area of research in the field of geodynamics is the

generation of numerical models that accurately portray the nature of the problem, while maintaining computational efficiency.

[6] Here, we introduce grid adaptivity, which is a method commonly employed within the field of engineering [see *Babuska and Rheinbolt*, 1978; *Lohner et al.*, 1985; *Peraire et al.*, 1987; *Pelletier and Ilinca*, 1995; *Nithiarasu and Zienkiewicz*, 2000]. Since we demonstrate the method in the context of finite elements it is termed the Adaptive Finite Element Method (AFEM). The method provides a powerful approach for the *accurate* and *efficient* solution of the complex problems encountered in geodynamics. Grid points are automatically clustered in regions of rapid solution variation to improve accuracy, leading to a "multi-resolution" solution, with the highest resolutions being analogous to zones of high solution gradient. In simple terms, the method automatically increases or decreases grid resolution where required, leading to more accurate solutions, while employing fewer degrees of freedom.

[7] In this study, the method is applied to infinite Prandtl number thermal and thermochemical convection in 2-D Cartesian geometry, to investigate its potential benefits within the field of geodynamics. We begin by investigating a benchmark thermal convection problem, with results illustrating that the method is highly successful, improving solution accuracy while increasing computational efficiency.

[8] The method is then applied to a more challenging thermochemical benchmark problem, employing a grid-based method to solve for the compositional field. Our reasons for selecting a grid-based method are simple. Recent work [e.g., *van Keken et al.*, 1997; *Tackley and King*, 2003] has highlighted the fact that such methods suffer from numerical diffusion, leading to greater entrainment rates in numerical simulations when compared to marker chain and tracer particle methods. This numerical diffusion is predominantly caused by insufficient resolution, a factor that is naturally addressed by the AFEM. Consequently, two hypotheses are tested:

[9] 1. The greater resolution endorsed by the AFEM will reduce artificial diffusion.

[10] 2. This reduced diffusion, in turn, will see grid-based methods yielding results that are consistent with those achieved using tracer particle and marker chain methods.

[11] As expected, results show that adapted grids yield large improvements over regular, uniform grids, generating less diffusive results while reducing the number of degrees of freedom. However, results also demonstrate that grid-based methods, even when coupled with the AFEM, are not competitive with the other methods currently employed for the tracking of compositional heterogeneities.

[12] The remainder of this paper will provide an overview of the test problems that are studied, a summary of the numerical techniques employed (an appendix is included with a more detailed analysis) and a comparison of the results obtained. These demonstrate the applicability of grid adaptivity to the modeling of mantle dynamics.

2. Problem Description

[13] We consider two dimensional thermal and thermochemical convection in an isoviscous, infinite Prandtl number Cartesian box. The equations (in dimensionless, vector form) describing such incompressible convection are the equations of

Momentum

$$\nabla^2 \mathbf{u} - \nabla p = RaT\hat{k} \tag{1}$$

Continuity (mass)

$$\nabla \cdot \mathbf{u} = 0 \tag{2}$$

Energy

$$\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T = \nabla^2 T \tag{3}$$

where $\mathbf{u}$ is the dimensionless velocity, T is the dimensionless temperature, p is the dimensionless nonlithostatic pressure, $\hat{k}$ is the unit vector in the vertical direction and t is the dimensionless time. In this form, all material properties are combined into one dimensionless parameter, the Rayleigh number:

$$Ra = \frac{\beta \rho g \Delta T d^3}{\kappa \mu} \tag{4}$$

where g is the acceleration due to gravity, ρ is density, β is the coefficient of thermal expansion, ΔT is the temperature drop across the domain, d is the domain length, κ is the thermal diffusivity and μ is the dynamic viscosity.

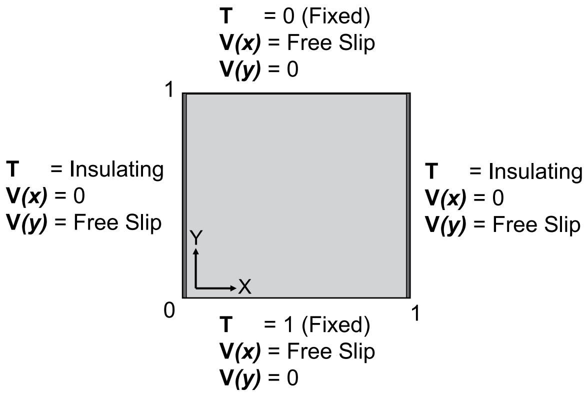


Figure 1. The boundary conditions utilized when studying thermal convection in a square cavity. *T*, *V*(*x*), and *V*(*y*) represent temperature and the velocities in the *x* and *y* directions, respectively.

[14] In thermochemical simulations, a second advection-diffusion equation is solved for composition:

$$\frac{\partial C}{\partial t} + \mathbf{u} \cdot \nabla C = \frac{1}{Le} \nabla^2 C \qquad (5)$$

where C represents composition and Le is the Lewis number, the ratio of thermal diffusivity to chemical diffusivity. The desired limit approaches infinite Le; however, for numerical reasons, a finite Le is often assumed. The momentum equation also differs from equation (1) and is now taken in the form

$$\nabla^2 \mathbf{u} - \nabla p = (RaT - RbC)\hat{k} \qquad (6)$$

Here, Rb is the compositional Rayleigh number:

$$Rb = \frac{\Delta \rho g d^3}{\kappa \mu} \qquad (7)$$

where $\Delta\rho$ is the density difference between the dense ($C = 1$) and light ($C = 0$) material.

[15] These equations are solved using a modified version of the widely used 2D geodynamics finite element program ConMan. A brief overview of the code is provided here; however, a more detailed description is given by *King et al.* [1990].

[16] The momentum and energy equations form a coupled set of differential equations, although the coupling is not strong since the density, ρ, is constant, other than in the buoyancy term of the momentum equation (Boussinesq approximation). The incompressibility (continuity) equation is treated as a constraint on the momentum equation, with incompressibility enforced using a penalty formulation. The energy equation is solved using a streamline upwind Petrov Galerkin (SUPG) method [*Hughes and Brooks*, 1979], with time stepping accomplished by means of a second order explicit predictor-corrector algorithm.

[17] A grid-based method, identical to that used for solving the energy equation, is utilized for modeling the compositional field. A small chemical diffusivity is assumed and a filtering scheme is employed to remove spurious numerical overshoot and undershoot features, common with advection diffusion problems of this nature. This filter conserves mass by design and has been shown to work remarkably well at limiting the aforementioned numerical errors (see *Lenardic and Kaula* [1993] for further details).

2.1. Case 1: Thermal Convection in a Square Cavity

[18] The first example considered is buoyancy-driven flow in an iso-chemical square cavity. The cavity is filled with a material of constant viscosity and there are no internal heat sources. Boundary conditions are summarized in Figure 1. This problem is solved at Ra = 10^4, 10^5, and 10^6, initially on uniform, structured meshes, and subsequently, on adapted, unstructured meshes.

[19] The following data or sets of data are calculated during the simulations:

[20] 1. The Nusselt Number, i.e., the mean surface temperature gradient:

$$Nu = \int_0^l -\frac{\partial T}{\partial y}(x, y = 1)dx \qquad (8)$$

where l is the length (= 1).

[21] The nondimensional Root-Mean-Square (RMS) velocity:

$$V_{RMS} = \sqrt{\frac{1}{V}\int_V ||\mathbf{u}||^2} \qquad (9)$$

where V is the area of the computational domain.

[22] 3. Nondimensional temperature gradients at domain corners:

$$q = -\frac{\partial T}{\partial y} \qquad (10)$$

with q_1 at $x = 0$, $y = 1$; and q_2 at $x = y = 1$.

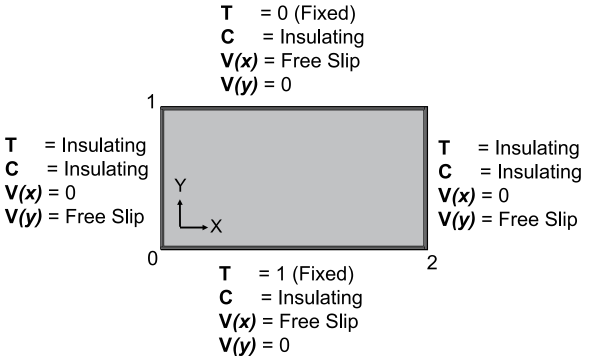


Figure 2. The boundary conditions utilized when studying the entrainment of a thin dense layer, in an aspect ratio 2 box, through thermochemical simulations. *T*, *C*, *V*(*x*), and *V*(*y*) represent temperature, composition, and the velocities in the *x* and *y* directions, respectively.

2.2. Case 2: Entrainment of a Thin Dense Layer Through Thermochemical Convection

[23] The second example considered is a well-established thermochemical benchmark problem from *van Keken et al.* [1997]. We model the entrainment of a deep-seated, thin (0.025), dense layer in an aspect ratio 2 box of unit height. This layer is prescribed a compositional value $C = 1$, while the overlying, lighter material has a value of $C = 0$. This problem is analogous to the entrainment of a compositionally dense layer in the D″ region at the Core-Mantle boundary. Results are computed for $1/Le = 10^{-6}$. The thermal Rayleigh Number, *Ra*, is set to 3×10^5, the compositional Rayleigh Number, *Rb*, is set to 4.5×10^5, while the viscosity is assumed to be constant. Boundary conditions are summarized in Figure 2. As an initial condition, an analytical expression of the temperature based on boundary layer theory is taken [see *van Keken et al.*, 1997, Appendix A]. Once again, this problem is solved initially on uniform, structured meshes, and subsequently, on adapted, unstructured meshes.

[24] For this case, the relative entrainment (e) is calculated as a function of time, from

$$e = \frac{1}{\lambda d_b} \int_{d_e}^{l} C dV \tag{11}$$

where λ is the aspect ratio of the box (= 2), d_b is the thickness of the dense layer and d_e is an arbitrarily chosen height (= 0.2 here). We focus our attention on the relative entrainment as opposed to other parameters, such as RMS velocity, since the rate of entrainment provides an excellent means to track the evolution of a thermochemical simulation. Additionally, entrainment rates have been calculated in previous studies [e.g., *van Keken et al.*, 1997; *Tackley and King*, 2003], and hence direct comparisons can be made with ease.

[25] Before providing a summary of the adaptive strategies employed, we should point out that the two cases presented above not only allow a means to test the applicability of the AFEM to thermal and thermochemical convection problems, but they also allow us to test the AFEM in both steady state (Case 1) and unsteady (Case 2) situations. Both are common within the field of geodynamics.

3. Adaptive Methodology

[26] As is clear from the introduction, the methods employed in modeling thermal and thermochemical convection must provide an adequate definition of the problem, in a computationally efficient manner. In other words, the methods must be adept at resolving narrow regions of high gradients that frequently occur and are normally found embedded in large areas where the flow variables vary slowly. Since the exact location of these high gradient regions are not always known to the analyst a priori, particularly with unsteady problems, it is apparent that adaptive mesh methods, with a posteriori error estimators, could have an important role to play in the development of efficient solution techniques for such problems. At present, a wide variety of adaptive procedures are being utilized within the engineering community. Broadly speaking, these fall into two categories:

[27] 1. *h*-refinement, in which the same class of elements continue to be used but are changed in size, in some locations made larger, and in others made smaller, to provide the maximum economy in reaching the desired solution.

[28] 2. *p*-refinement, in which the same element size is utilized, but the order of the polynomial is increased or decreased as required (e.g., linear shape functions are ‘‘adapted’’ to quadratic or higher order).

[29] A variant of the *h*-method, known as ‘‘adaptive remeshing,’’ is employed in this study. It provides the greatest control of mesh size and grading to better resolve the flow features. In this method, for steady state simulations, the problem is solved initially on a grid fine enough to roughly capture the physics of the flow. Remeshing then involves the following steps:

[30] 1. The solution is analyzed through some kind of error estimation procedure, to determine locations where the mesh fails to provide an adequate definition of the problem. This can mean that either additional grid points are needed, or indeed, that there are too many grid points at certain locations within the domain. An interpolation-based local error estimator is employed in this study, based upon nodal solution gradients and curvatures.

[31] 2. The information yielded by this error estimation process is utilized to generate an improved mesh through an automatic mesh generator. A variant of the so-called advancing front technique is utilized here, being capable of generating meshes that conform to a user prescribed spatial distribution of element size. Elements can locally increase or decrease in size as required, leading to what we term an "optimal" mesh.

[32] 3. The original solution is interpolated between meshes using higher-order cubic interpolation [*Nielson*, 1979; *El Hachemi et al.*, 2003].

[33] 4. The solution procedure continues on the new mesh.

[34] The remeshing process is repeated until a desired level of accuracy has been achieved.

[35] For unsteady problems, the process is almost identical; however, it is fundamental that the initial mesh is suitably defined. If the mesh is inadequate, the errors generated during the calculation's early stages propagate through the computational domain, generating misleading results. To ensure such errors do not arise in our simulations, we generate an optimal initial mesh. This process is straightforward. The initial condition is set up on a structured grid. In the same way as is described in points 1-4 above, the data from this structured grid is analyzed to determine where the mesh needs modification. This information is then used to regenerate the mesh, and the solution (i.e., the initial condition) is transferred onto the new mesh using cubic interpolation. Consequently, the initial grid naturally provides an optimal definition of the problem. With unsteady problems, the remeshing procedure can be continued indefinitely as the simulation evolves. The remeshing "loop" is activated after a user-defined time interval, or dynamically, upon the basis of an approximation to the error.

[36] It is important to point out that although the process seems similar to Lagrangian formulations (the computational mesh appears to follow the solution), it is indeed an Eulerian formulation, perhaps with some of the advantages commonly associated with Lagrangian schemes.

[37] For a more detailed description of the mesh generation process, the error estimation procedure and the adaptive strategy, please refer to Appendix A. A flow chart summarizing the essential stages involved is also included in Figure 3.

3.1. Remeshing Procedure for Case 1: Steady-State Adaptivity

[38] A steady-state solution is achieved here. Remeshing is therefore a simple task, being performed when the solution converges to a steady state on a given grid. The process is terminated when an optimal mesh has been produced, i.e., the solution does not improve with the remeshing procedure. The error estimator employed in this case is based upon nodal temperature gradients and curvatures.

3.2. Remeshing Procedure for Case 2: Temporal Adaptivity

[39] The results presented are based upon simulations with a remeshing frequency of 2000 time steps. This value was selected after a series of tests, both visual and analytical, tracing the temporal evolution of the model. Ideally, the remeshing procedure would be linked to the dynamics (i.e., it should be tied to some measure of how much the solution has changed or whether derivatives in the mesh exceed a certain tolerance). Nonetheless, we have verified, through tests at various remeshing frequencies, that for this simulation, the remeshing frequency selected (i.e., 2000 time steps) does not degenerate the results.

[40] The error estimator employed is similar to that in Case 1; however, it is based upon a combination of temperature and composition, as opposed to temperature alone. Nodal solution derivatives are calculated for both the temperature and compositional fields. The highest values yielded are then selected as derivatives for that particular node. Such a scheme engenders high resolving power at the density interface, as well as sufficient resolution to accurately solve the thermal field. We have verified that this combination yields superior results to simulations employing a combination of the composition and velocity variables.

[41] The remeshing strategy is slightly different to that in Case 1. Rather than simply refining zones of high solution gradient, we also allocate fine reso-

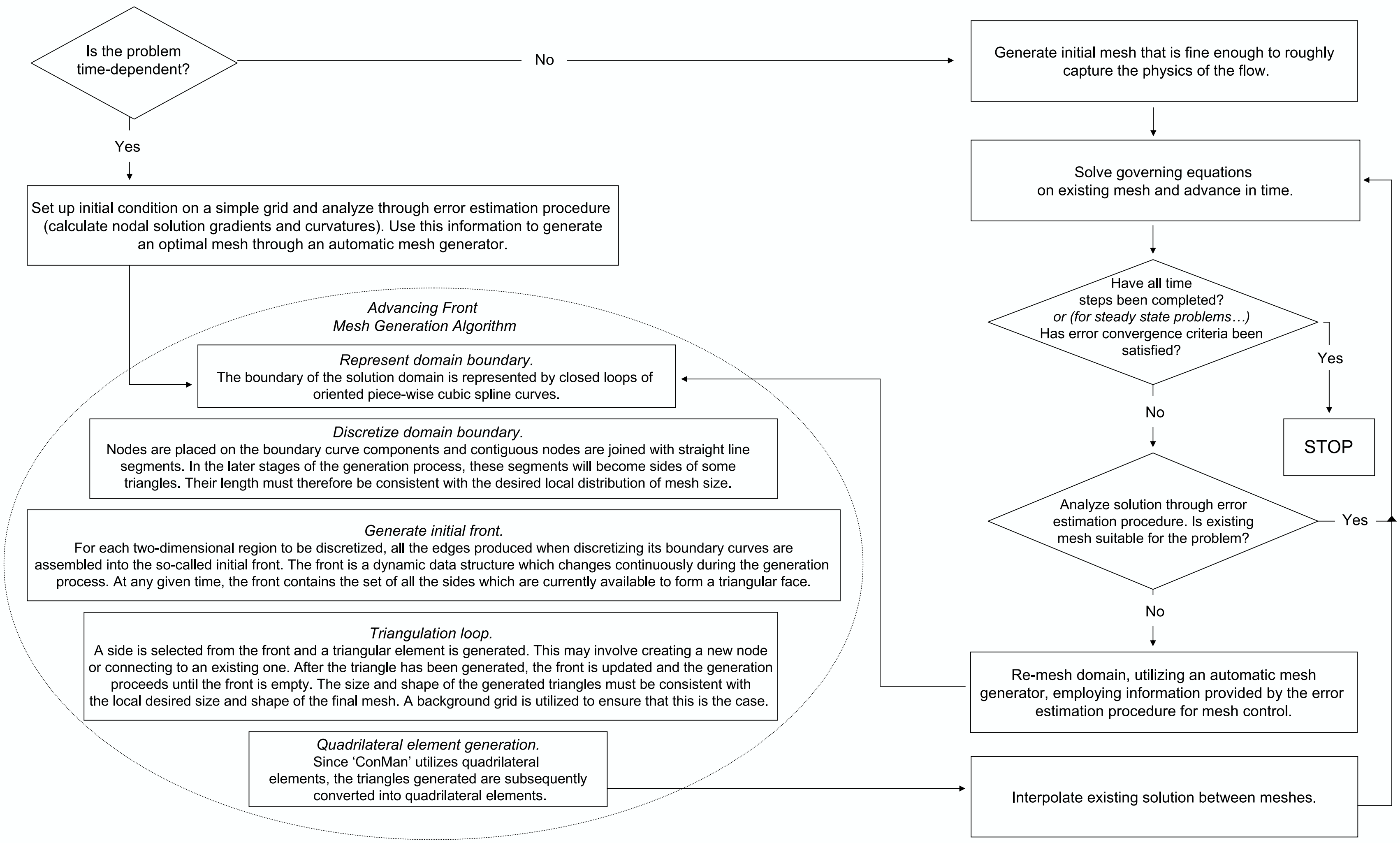

Is the problem time-dependent?
No
Yes
Generate initial mesh that is fine enough to roughly capture the physics of the flow.
Set up initial condition on a simple grid and analyze through error estimation procedure (calculate nodal solution gradients and curvatures). Use this information to generate an optimal mesh through an automatic mesh generator.
Solve governing equations on existing mesh and advance in time.
Have all time steps been completed? *or (for steady state problems…)* Has error convergence criteria been satisfied?
Yes
STOP
No
Analyze solution through error estimation procedure. Is existing mesh suitable for the problem?
Yes
No
Re-mesh domain, utilizing an automatic mesh generator, employing information provided by the error estimation procedure for mesh control.
Interpolate existing solution between meshes.
Advancing Front Mesh Generation Algorithm
Represent domain boundary. The boundary of the solution domain is represented by closed loops of oriented piece-wise cubic spline curves.
Discretize domain boundary. Nodes are placed on the boundary curve components and contiguous nodes are joined with straight line segments. In the later stages of the generation process, these segments will become sides of some triangles. Their length must therefore be consistent with the desired local distribution of mesh size.
Generate initial front. For each two-dimensional region to be discretized, all the edges produced when discretizing its boundary curves are assembled into the so-called initial front. The front is a dynamic data structure which changes continuously during the generation process. At any given time, the front contains the set of all the sides which are currently available to form a triangular face.
Triangulation loop. A side is selected from the front and a triangular element is generated. This may involve creating a new node or connecting to an existing one. After the triangle has been generated, the front is updated and the generation proceeds until the front is empty. The size and shape of the generated triangles must be consistent with the local desired size and shape of the final mesh. A background grid is utilized to ensure that this is the case.
Quadrilateral element generation. Since 'ConMan' utilizes quadrilateral elements, the triangles generated are subsequently converted into quadrilateral elements.

Figure 3. A flow diagram summarizing the major steps involved in the adaptive remeshing procedure.

Table 1. Results Obtained on Various Uniform, Structured Meshes for Simulations of Thermal Convection in a Square Cavity at Ra = 10^5 [a]

	Mesh 1	Mesh 2	Mesh 3	Mesh 4	Mesh 5	Mesh 6	Benchmark	Uncertainty
Elements	256	1024	4096	7225	10000	16384	****	****
Nu	7.8757	9.6838	10.3361	10.4453	10.4712	10.5058	10.5341	0.00001
V_{RMS}	197.4998	194.2366	193.4223	193.3222	193.2968	193.2787	193.2145	0.0001
q_1	11.8004	16.5079	18.3362	18.6467	18.7641	18.8861	19.0794	0.00004
q_2	1.4204	0.9397	0.7754	0.7529	0.7445	0.7358	0.7228	0.00002

[a] The benchmark results of *Blankenbach et al.* [1989] are included for comparison, together with their uncertainties.

lution to neighboring regions. This is done by ensuring that the transition from fine to coarse elements is extremely gradational. In the steady-state cases, we specify that a minimum of 10 elements is required to make this transition. However, in our thermochemical, unsteady simulations, this value is set to 30. In this way, fine regions of the mesh are always surrounded by relatively fine zones. Accordingly, as the simulation evolves, accuracy is maintained, since zones of high solution gradient have not departed the fine grid regions before the next remeshing. Such a scheme allows a greater time interval between remeshings and was a key consideration when selecting the remeshing frequency.

[42] It is important to note that the minimum element size ($\delta_{\min}$) permitted in our simulations is ≈0.002 which yields localized resolution equivalent to that achieved during a uniform mesh simulation of 1000 × 500 elements, in a box of aspect ratio 2.

4. Results and Discussion

4.1. Thermal Convection in a Square Cavity

4.1.1. Uniform Structured Meshes

[43] The results obtained for uniform mesh calculations at Ra = 10^5 are displayed in Table 1. The results at both Ra = 10^4 (Table 2) and Ra = 10^6 (Table 3) demonstrate a comparable relationship, and consequently only one set of results are presented fully.

[44] Figure 4 displays the relationship between the number of elements in a mesh and the RMS velocity and mean Nusselt Number, respectively. The figures illustrate that a large number of elements (i.e., >4000) is required before results begin to show some sort of consistency. Even more elements are required before results begin to converge to the benchmark solutions of *Blankenbach et al.* [1989].

[45] By calculating the percentage error for each output analyzed (i.e., mean Nu, RMS velocity, q_1 and q_2) and subsequently taking the mean of these four percentages, we have determined the discrepancy, i.e., the solution error, between uniform mesh solutions and benchmark solutions at various Rayleigh numbers. Our results are summarized in Table 4.

[46] Clearly, as convection intensifies, the solution error becomes more prominent, as would be expected. At higher Rayleigh numbers (10^5, 10^6), RMS velocities, mean Nusselt numbers, and corner temperature gradients lie far beyond the realms of uncertainty of the benchmark solution. Even at Ra = 10^4 solutions fail to achieve a suitable level of accuracy. However, as we show next, with the use of adaptive, unstructured meshes, solution accuracy is greatly enhanced.

4.1.2. Adapted Meshes

[47] The results obtained at Ra = 10^4, 10^5 and 10^6 alongside their final adapted meshes are displayed in Figure 5. They are also summarized in Tables 5, 6, and 7. For completeness, the sequence of meshes employed at Ra = 10^5 are displayed in Figure 6, together with the corresponding temperature contours. Once again, results for both Ra = 10^4 and

Table 2. **1**Results Obtained on Uniform Meshes for Simulations of Thermal Convection in a Square Cavity at Ra = 10^4

	UM[a]	Benchmark	Uncertainty
Elements	16384	****	****
Nu	4.8952	4.8844	0.00001
V_{RMS}	42.8713	42.8649	0.00002
q_1	8.0457	8.0594	0.000003
q_2	0.5905	0.5888	0.000003

[a] UM, uniform mesh.

Table 3. Results Obtained on Uniform Meshes for Simulations of Thermal Convection in a Square Cavity at Ra = 10^6

	UM	Benchmark	Uncertainty
Elements	16384	****	****
Nu	21.3773	21.9725	0.00002
V_{RMS}	834.9486	833.9898	0.0002
q_1	43.3217	45.9673	0.0003
q_2	0.9737	0.8772	0.00001

Table 4. Solution Errors Obtained on a Uniform Mesh of 16384 Elements (128 × 128 Elements), at Various Rayleigh Numbers, for Thermal Convection in a Square Cavity

Rayleigh Number	Solution Error, %
10^4	0.2
10^5	0.9
10^6	4.9

10^6 display a comparable relationship, and, consequently, only one set of results is presented fully. The characteristics of each mesh at Ra = 10^5 are summarized in Table 8, which shows the number of quadrilateral elements and nodal points, and the values of the generation parameters (δ_{Min}, δ_{Max}, s_{Max} and C) defined in Appendix A. It is important to note that for consistency, the generation parameters displayed in Table 8 are also utilized in our simulations at Ra = 10^4 and Ra = 10^6. It should also be noted that we have intentionally restricted the number of elements to less than 16,500, to allow a simple comparison between adapted and uniform structured meshes.

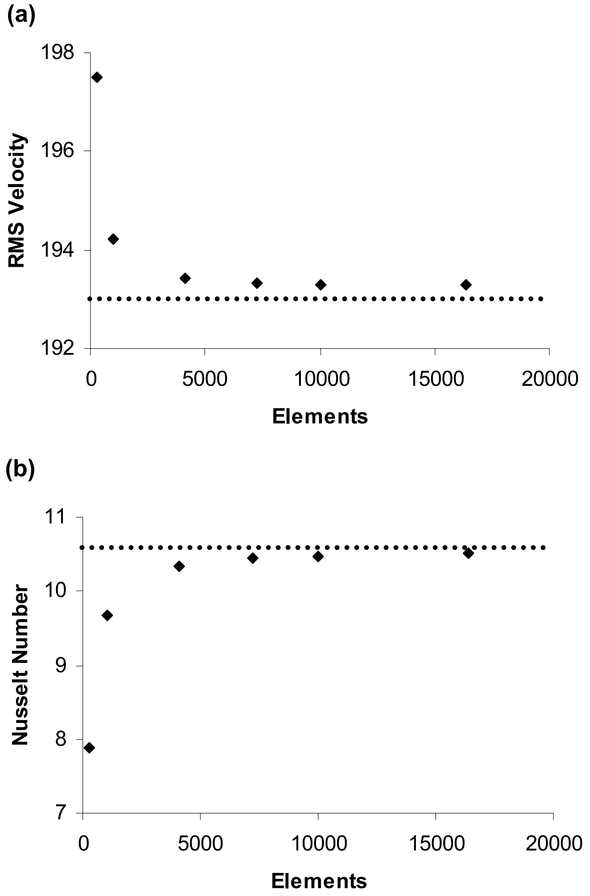


Figure 4. The relationship between the number of elements in a mesh and (a) RMS velocity and (b) mean Nusselt Number. The results represent thermal convection in a square cavity, on uniform, structured meshes at Ra = 10^5. Benchmark values are represented by horizontal dashed lines.

[48] As seen, the proposed procedure has refined locations of thermal boundary layers, wherever they are strong. Consequently, when compared to uniform structured meshes, the majority of results show far superior concurrence with the benchmark solution, in spite of a reduction in the number of elements (Table 9). As noted previously, at Ra = 10^5, a uniform mesh of more than 16,000 elements is required for a reasonably well-resolved solution, i.e., within 1% of the benchmark results. At Ra = 10^6, extrapolating from a series of uniform mesh simulations, we expect that more than 50,000 elements would be required before results converge to within ≈4% of the benchmark solution. More accurate results are inaccessible with the lower-order, serial configuration of ConMan. However, with the proposed adaptive procedure this is not the case. At Ra = 10^5, solutions within 0.1% of the benchmark are achieved on a mesh of ≈15,700 elements and at Ra = 10^6, results converge to within 1% on a mesh of ≈16,200 elements. Using the AFEM, the number of elements required for adequate solution varies with Rayleigh number, but is always significantly less than that of a uniform mesh for a specified precision.

[49] It is important to point out that the benefits of this technique become more noticeable when convection is intense, and temperature gradients are greater. At Ra = 10^4, only a moderate increase in accuracy is observed between adapted and uniform meshes, for approximately the same number of elements (solution error decreases by a factor of 3, from ≈0.2% to ≈0.06%). However, at Ra = 10^6,

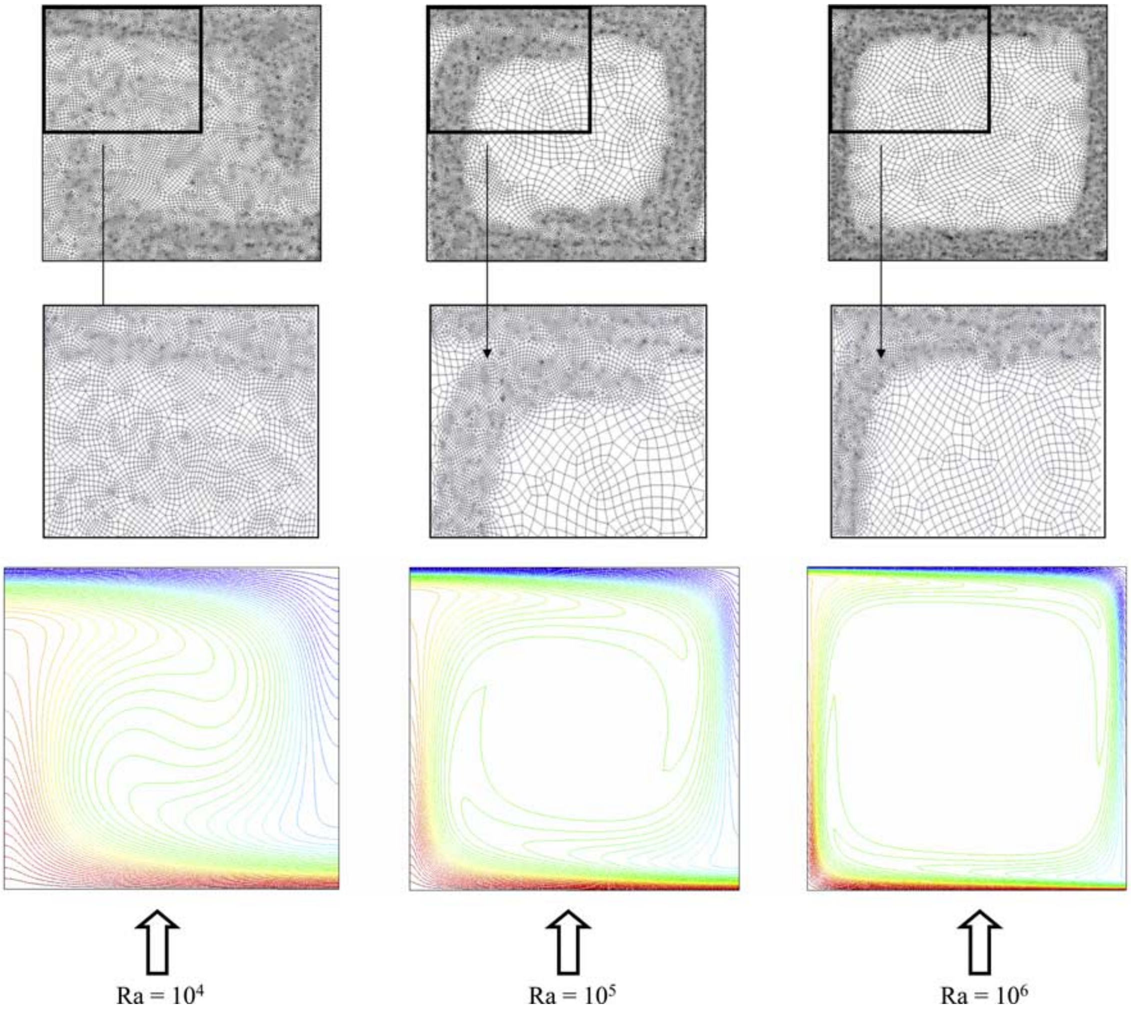


Figure 5. Final adapted meshes and corresponding temperature contours for purely thermal convection in a square cavity at Ra = 10^4, 10^5, and 10^6. Red is hot (T = 1), blue is cold (T = 0), and the contour spacing is 0.02.

results display a superior concordance with the benchmark solution on adapted meshes (the solution error decreases by a factor of 5, from ≈5% to ≈1%). Indeed, on fully uniform meshes with linear shape functions, obtaining an accurate solution on a single processor would be highly impractical, at Ra = 10^6.

[50] Other conclusions can be drawn by analyzing the 4 outputs individually. The remeshing process clearly has a positive effect on the global measures (i.e., V_{RMS} and Nu). However, the procedure seems

Table 5. Results Obtained After Each Remeshing for Simulations at Ra = 10^4 on Nonuniform, Adapted Meshes

	Mesh 1	Mesh 2	Mesh 3	Mesh 4	Benchmark
Elements	412	1977	6474	14972	****
Nu	4.2687	4.8565	4.8683	4.8790	4.8844
V_{RMS}	40.8536	42.9019	42.8792	42.8679	42.8649
q_1	7.2917	8.0345	8.0479	8.0538	8.0594
q_2	0.6366	0.5887	0.5886	0.5885	0.5888

Table 6. Results Obtained After Each Remeshing at Ra = 10^5

	Mesh 1	Mesh 2	Mesh 3	Mesh 4	Benchmark
Elements	412	2390	7321	15722	****
Nu	7.9675	10.0967	10.4380	10.5278	10.5341
V_{RMS}	184.8429	193.5598	193.3056	193.2456	193.2145
q_1	13.1450	18.9078	19.0353	19.0518	19.0794
q_2	1.2687	0.7301	0.7195	0.7216	0.7228

Table 7. Results Obtained After Each Remeshing at $Ra = 10^6$

	Mesh 1	Mesh 2	Mesh 3	Mesh 4	Benchmark
Elements	412	2733	7681	16195	****
Nu	11.6429	18.5597	20.9920	21.5077	21.9725
V_{RMS}	809.9980	841.9425	835.1100	834.7003	833.9898
q_1	12.3555	38.3885	44.7377	45.601	45.9643
q_2	2.6031	0.9379	0.9038	0.8694	0.8772

to have a more dramatic effect on the accuracy of the heat flux at domain corners (q_1 and q_2), particularly at higher Rayleigh numbers. This is easy to understand; the heat flux at these corners is strongly influenced by the resolution achieved in the upper thermal boundary layer. Since this boundary layer is characterized by high temperature gradients, the remeshing procedure refines the grid significantly in these regions. Accordingly, the corner solutions yielded by adapted meshes are far superior to those yielded on uniform structured meshes. Global measures on the other hand, are not influenced by these boundary layers to such an extent. Consequently, the improvement observed in global measures between adapted and uniform grids is less dramatic.

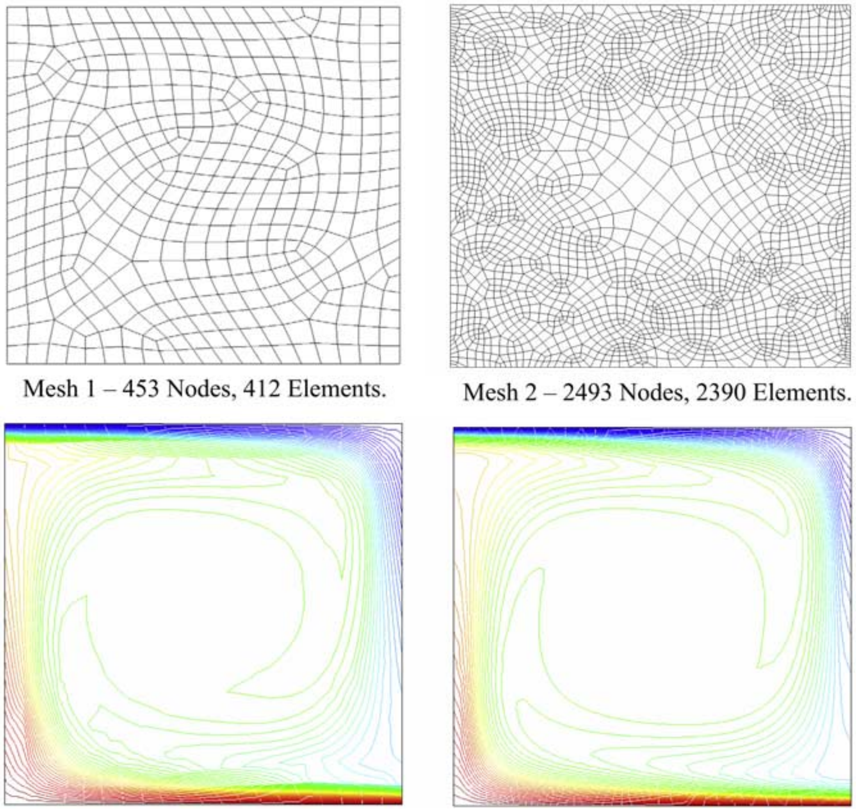
4.1.3. AFEM and Processing Efficiency

[51] We have demonstrated that the number of nodes and elements required for accuracy is less with the AFEM. Consequently, the AFEM is more efficient in terms of memory requirements. However, is the AFEM economical in terms of computational processing time?

[52] Figure 7 illustrates the relationship between solution error and the time taken in obtaining these solutions on both uniform and adapted meshes (at various Rayleigh numbers). It should be noted that the timings displayed for the adaptive cases include the time allocated for remeshing. The main points arising are summarized below:

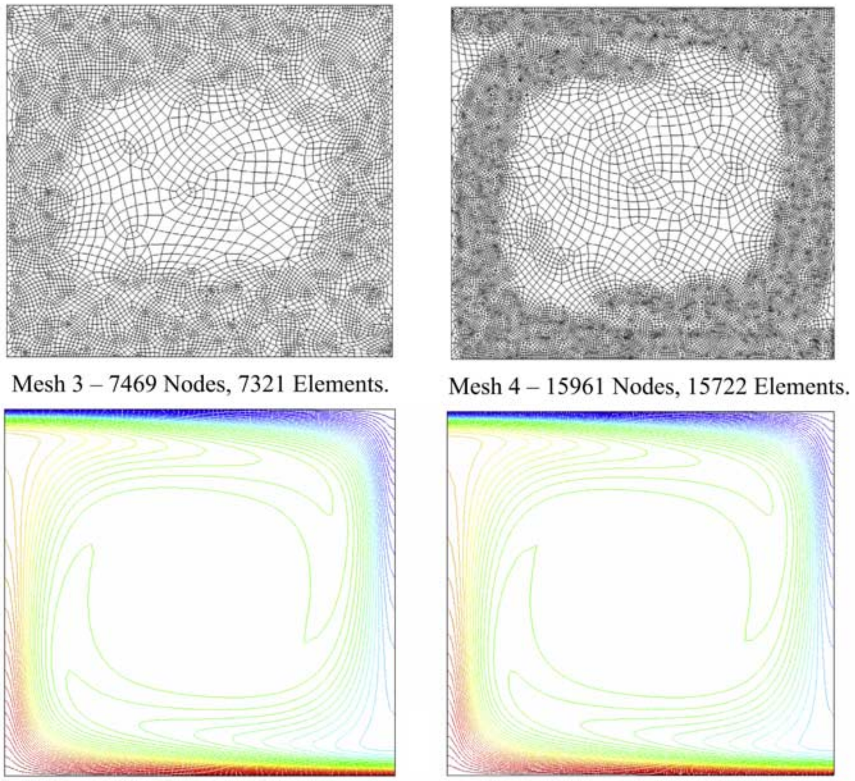
[53] 1. In general, at $Ra = 10^4$, the AFEM is less efficient than uniform meshes for a prescribed level of accuracy. However, when a solution error of less than ≈0.2% is required, the AFEM becomes more economical.

[54] 2. At $Ra = 10^5$ and $Ra = 10^6$ the AFEM is more efficient than uniform structured meshes, decreasing computational processing time while increasing solution accuracy. Indeed, the lower


Mesh 1 – 453 Nodes, 412 Elements.

Mesh 2 – 2493 Nodes, 2390 Elements.

Figure 6. The series of meshes employed at $Ra = 10^5$ along with corresponding temperature contours. Red is hot ($T = 1$), blue is cold ($T = 0$), and the contour spacing is 0.02.



Mesh 3 – 7469 Nodes, 7321 Elements.

Mesh 4 – 15961 Nodes, 15722 Elements.

Figure 6. (continued)

graph in Figure 7 (Ra = 10^6) suggests that solution errors of less than ≈4% cannot be obtained on uniform meshes. With the AFEM this is not the case; far superior results are attained, although due to the lower order nature of ConMan, minor errors persist.

[55] Table 10 summarizes the percentage of calculation time taken by the remeshing procedure, compared to the time spent solving the governing equations. It is important to note that although remeshing appears more efficient at higher Rayleigh numbers, this is not strictly true. At higher convective vigors, simulations take longer to converge toward a steady state solution. Since the number of remeshing loops employed during each simulation is constant (i.e., 3), it is apparent that remeshing will take a smaller fraction of the calculation time at higher Rayleigh numbers (since the calculation time as a whole is greater). Regardless of this point, the main conclusion to be drawn from the data is that the remeshing procedure is computationally inexpensive, expending only a small percentage of the calculation time.

[56] In summary, in the context of purely thermal convection, the number of degrees of freedom required for accuracy on uniform structured meshes is greater than that required for adapted meshes. Thus, for the same precision, the number of nodes and elements is reduced when the adaptive procedure is used. Additionally, the remeshing

Table 8. Sequence of Meshes Employed for the Problem of Buoyancy-Driven Flow in a Square Cavity at Ra = 10^5

Mesh	Elements	Nodes	δ_{Min}	δ_{Max}	s_{Max}	C
1	412	453	0.05	0.05	1.0	-
2	2390	2493	0.009	0.05	5.0	0.6
3	7321	7469	0.0045	0.030	5.0	0.3
4	15722	15961	0.0025	0.025	5.0	0.2

Table 9. Solution Errors Yielded by Nonuniform Adapted Meshes at Various Rayleigh Numbers

Rayleigh Number	Elements	Solution Error, %
10^4	14972	0.06
10^5	15722	0.09
10^6	16195	1

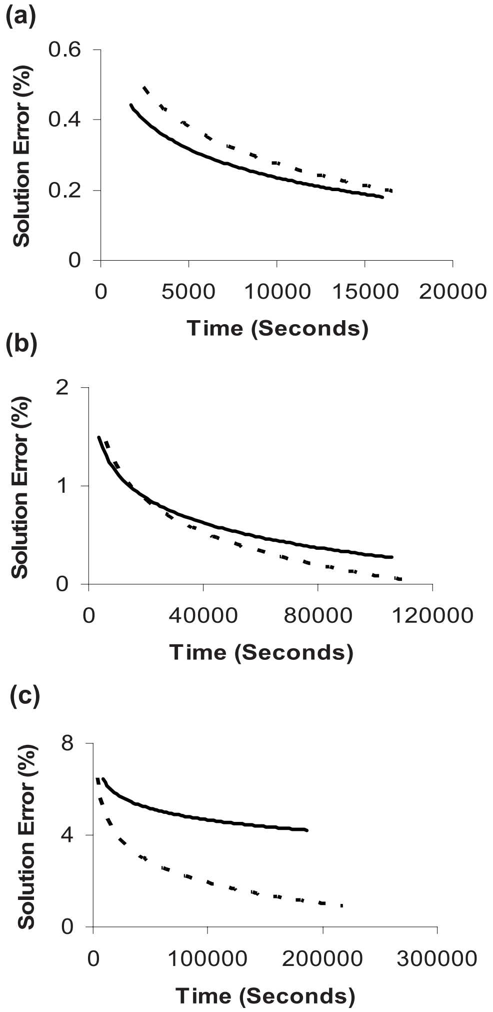


Figure 7. The time taken to converge on various solution errors with both uniform (continuous line) and adapted (dashed line) meshes at (a) $Ra = 10^4$, (b) $Ra = 10^5$, and (c) $Ra = 10^6$.

procedure is computationally inexpensive and, consequently, particularly at higher Ra ($>10^5$), the AFEM allows one to attain a desired solution in less processing time.

4.2. Thermochemical Convection

[57] Having demonstrated the applicability and benefits of the AFEM for thermal convection, we move on to thermochemical simulations.

4.2.1. Uniform Structured Meshes

[58] Curves illustrating the entrainment (e) yielded by uniform mesh simulations at various grid resolutions are displayed in Figure 8. The results are consistent with previous work in that relative entrainment decreases with increased resolution. As a quantitative example, entrainment at $t = 0.02$ decreases from a value of 0.4 on a grid of 64 × 64 elements, to a value of 0.08 on a grid of 256 × 256 elements. This large reduction can be easily understood; grid-based methods suffer from numerical diffusion, which smears density interfaces over several grid spacings. Consequently, with coarser grids, material at the density interface itself becomes easier to entrain. As one increases grid resolution, grid spacings decrease, leading to less artificial diffusion and hence reduced entrainment.

[59] Perhaps a more fundamental point to note from Figure 8, however, is that even after this significant increase in resolution, when $t = 0.01$, and indeed for a high percentage of the calculation, the grid-based method displays almost an order of magnitude greater entrainment than the other methods currently employed in this field. The marker chain results produced by Christensen, Neumeister and Dion from *van Keken et al.* [1997] are displayed for comparison. The reader should note that the tracer particle studies of *Tackley and King* [2003] are consistent with these results. Clearly, there is no resemblance between the grid-based methods and these results, particularly during the early stages of the simulation. At first glance, it does appear that when $t > 0.02$, the results yielded by grid-based simulations are reasonably consistent with those of previous particle studies. Closer analysis, however, reveals that this is misleading. The drastic difference in entrainment during the early development of these models means that, by this time, both simulations have evolved into completely different problems. Consequently, the apparent consistent relationship between both methods is purely coincidental. This point is reinforced when studying the visual patterns; there is only a poor resemblance between Figure 9 here, which shows the temporal evolution of the compositional field, and equivalent Figures 8 and 10 of *van Keken et al.* [1997] and Figure 3 of *Tackley and King* [2003]. The similarities also diminish as the simulation evolves. The main differences include the position of the dense pile after reorganization

Table 10. Percentage of Computational Time Taken by the Remeshing Process at Various Rayleigh Numbers

Rayleigh Number	% Time Remeshing
10^4	4
10^5	1.7
10^6	0.8

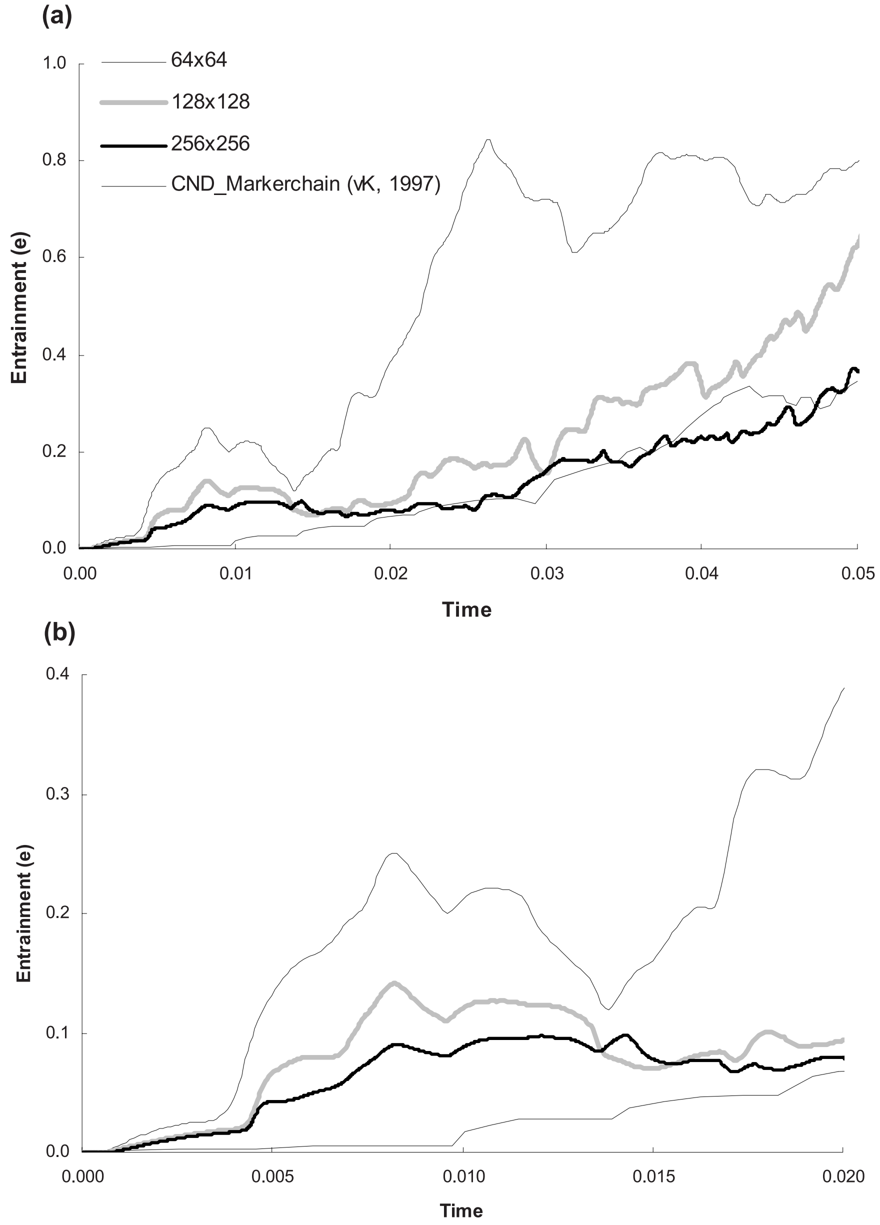


Figure 8. (a) Relative entrainment (*e*) against time on a series of uniform meshes. The "best" results from *van Keken et al.* [1997] (CND Markerchain) are also displayed for ease of comparison. (b) An enlargement of the results in Figure 8a for time ≤ 0.02.

into a two cell pattern and the amount of material trapped at the stagnation point. An important observation to make here is that, during the later stages of the calculation, diffusion dominates and, by t = 0.07, the dense layer has virtually disappeared. This is not the case when tracer particle and marker chain methods are employed, as has been pointed out by *van Keken et al.* [1997].

[60] It is clear from the simple tests performed here, on uniform meshes, that increased grid resolution reduces entrainment. Results move toward those yielded with tracer particle and marker chain methods; however, entrainment rates remain significantly higher. Next, we will attempt to answer whether this is the case when the AFEM is employed or does the method provide sufficient resolution to resolve these discrepancies?

4.2.2. Adapted Unstructured Meshes

[61] The generation parameters (δ_{Min}, δ_{Max}, s_{Max} and C) utilized within these models are summarized in Table 11. Unlike those previously defined

Table 11. Mesh Parameters Employed for Thermochemical Entrainment Simulations

δ_{Min}	δ_{Max}	s_{Max}	C
0.002	0.02	5	0.1

for the purely thermal case (Table 8), the values remain constant throughout the simulation. There is a simple explanation for this. Since the problem under study does not converge to a steady-state solution, one must ensure that not only is the error equally distributed spatially, but also temporally. By fixing the generation parameters over time, one guarantees that this is the case.

[62] Entrainment curves from these simulations are displayed in Figure 10. The result obtained from a uniform mesh simulation on a grid of 256 × 256 elements is also displayed for ease of comparison, along with the "best" results of *van Keken et al.* [1997].

[63] Plots are displayed from simulations employing linear (AM-Linear) and cubic (AM-Cubic) interpolation between grids. The accuracy of this interpolation process is of fundamental importance during unsteady problems of this nature. With steady state problems, the solution always converges toward a certain end member. Consequently, minor errors arising due to inaccurate interpolation can be overcome. However, with unsteady problems, errors arising during the interpolation process propagate through the computational domain, leading to a solution that is unrepresentative of the true problem. In essence, the solution emerging from the remeshing procedure must be exactly that which enters. Otherwise, the simulation evolves falsely.

[64] The results involving linear interpolation are therefore only presented for completeness; the method fails to accurately interpolate features at the density interface. At certain locations, the density jumps from a value of 0 to 1 within a single element. Linear interpolation is not capable of resolving such a feature and, consequently, the remeshing process employing linear interpolation generates diffusive and nonconservative results. The cubic interpolation strategy employed, however

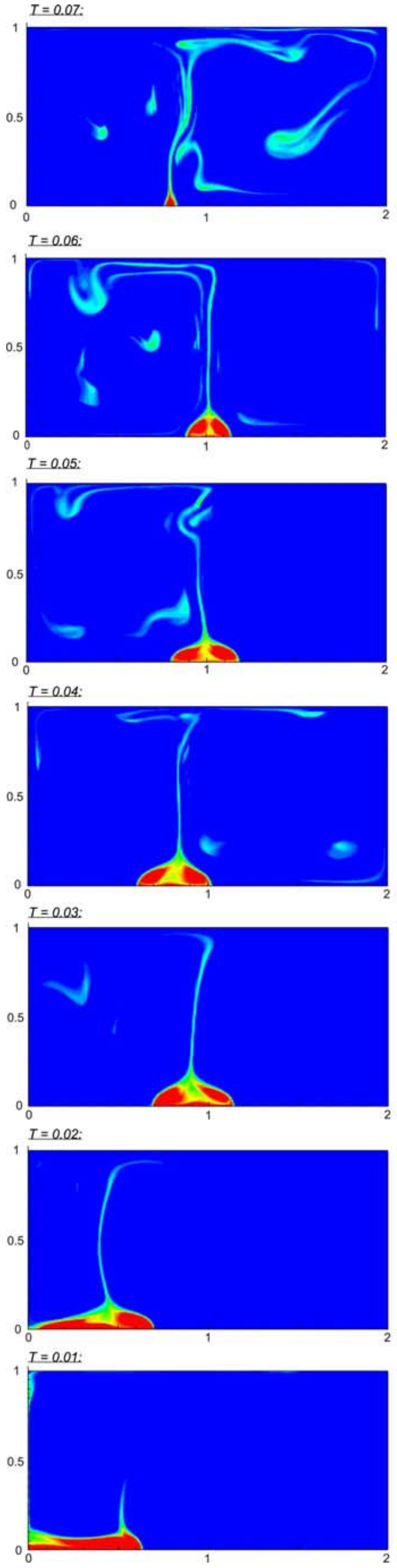


Figure 9. Seven figures, at regular time intervals of 0.01, illustrating the evolution of the compositional field, modeled on a uniform mesh of 256 × 256 elements. Red represents dense material (C = 1), while blue represents lighter material (C = 0).

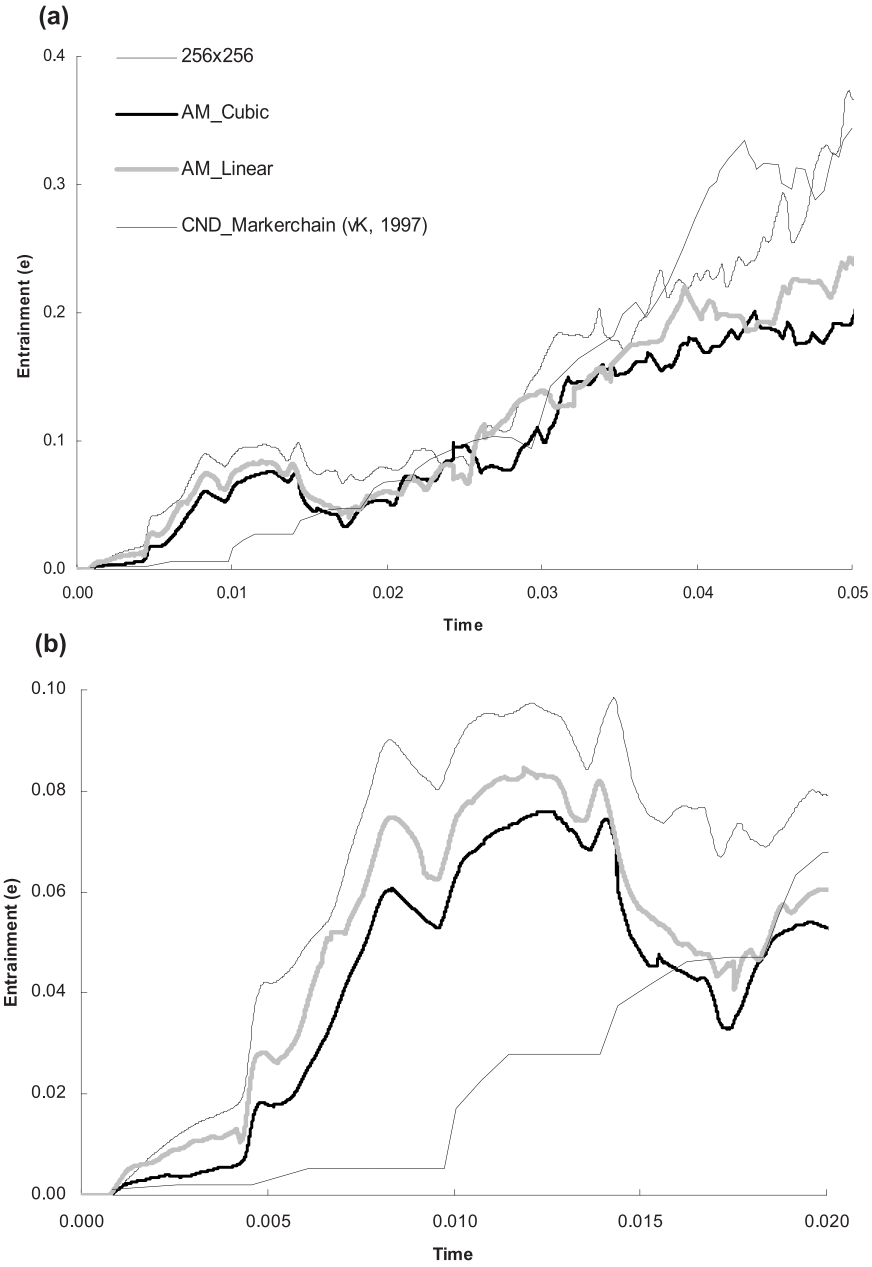


Figure 10. (a) Relative entrainment (e) against time on adapted, unstructured meshes. The results obtained on a 256 × 256 element uniform mesh are also displayed, as well as the "best" results from *van Keken et al.* [1997]. (b) An enlargement of the results in Figure 10a for time ≤ 0.02.

[*Nielson*, 1979; *El Hachemi et al.*, 2003], accurately captures such features, being both locally and globally conservative.

[65] The results demonstrate that in general, the AFEM leads to a significant reduction in artificial diffusion and hence entrainment rates, when compared to uniform mesh simulations. This point is reinforced by making a comparison of the longevity of the dense pile in the visual output (Figures 9 and 11). As was noted earlier, with a uniform mesh simulation on a grid of 256 × 256 elements, by t = 0.07, the dense pile has almost completely diffused. This is not the case with simulations employing the AFEM; the dense pile remains extremely coherent until this time. It is clear therefore that the higher resolution permitted by the AFEM leads to decreased artificial diffusion and, consequently, results that provide a more precise representation of the problem.

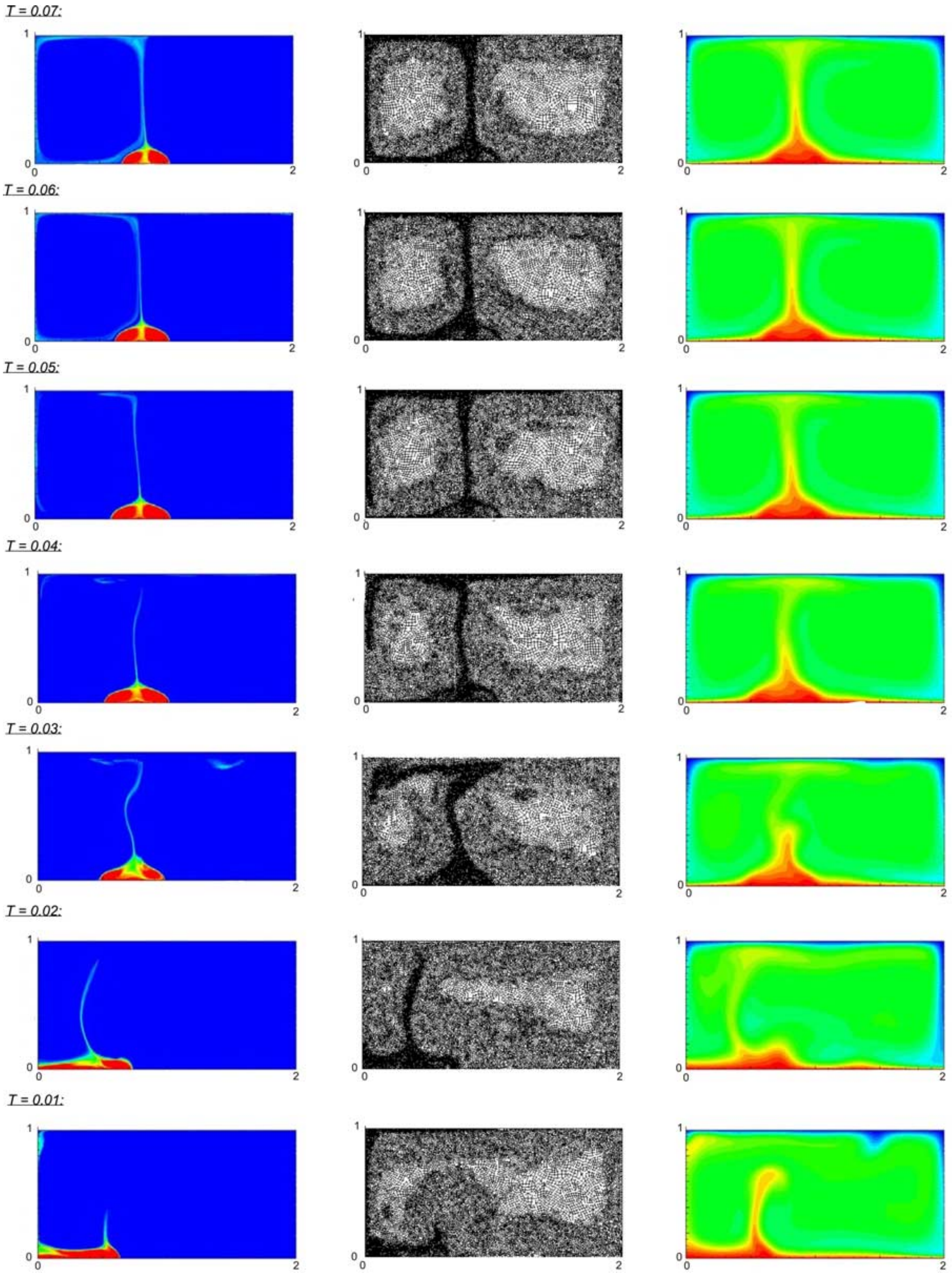


Figure 11. Evolution of (left) the compositional field and (right) the temperature field on (middle) a series of adapted grids. The grids are adapted around temperature and compositional solution gradients. Additionally, a region of fine resolution is generated adjacent to zones of high solution gradient. Consequently, as the simulation evolves, high gradient zones remain in regions of high resolution, leading to less numerical error.

[66] It is important to point out that, in addition to yielding superior results, our adaptive grid simulations are computationally more efficient, both in terms of memory and processing time, when compared to uniform mesh simulations. The number of elements utilized in this simulation varies with time between ≈40,000 and ≈58,000, depending upon the configuration of the calculation. Additionally, the generation of a new optimal mesh is an inexpensive procedure, typically taking between 15 and 20 time steps, compared to the time taken for one time step with a nonchanging mesh (although the time expended in remeshing can be decreased significantly by specifying a larger *err_max* - the remeshing tolerance; see Appendix A, section A2.2). Obviously, this gain in computational efficiency is only valid provided one does not set the minimum element size, $\delta_{\min}$, to an unreasonably small value. Our experience suggests that simulations with $\delta_{\min} < 0.002$ are highly impractical, due to the tradeoff between minimum element size and time stepping, dictated by the Courant-Friedrichs-Levy condition. Of course, this situation could be remedied by employing a local time stepping algorithm; however, this was beyond the scope of our study. It should be noted that the efficiency of the method as a whole depends on how often it is necessary to remesh, which depends on how time-dependent the simulation is. Mesh adaptivity therefore becomes less efficient for highly time-dependent cases needing frequent remeshing.

[67] Although the results presented demonstrate the benefits of the AFEM, grid-based thermochemical methods, even when coupled with the AFEM, yield results that are beyond the realms of uncertainty of those achieved via tracer particle and marker chain methods. The method remains particularly diffusive, yielding erroneous entrainment rates, even at the extremely fine resolutions permitted in our simulations. Once again, these findings are reinforced by the visual patterns, with Figure 11 displaying only a marginal resemblance to Figures 8 and 10 of *van Keken et al.* [1997] and Figure 3 of *Tackley and King* [2003]. This resemblance, however, is stronger than that observed with our uniform mesh simulations (Figure 9).

5. Conclusions

[68] An adaptive finite element procedure has been presented for solving convective heat transfer problems within the field of geodynamics. The method adapts the mesh automatically around regions of high solution gradient, yielding enhanced resolution of the associated flow features.

5.1. Applicability to Thermal Convection

[69] The results obtained from thermal convective simulations are extremely positive. The error estimator presented has proven reliable and the adaptive procedure is shown to be robust. Predictions for heat transfer agree well with benchmark solutions, suggesting that the technique is valid and accurate.

5.2. Applicability to Thermochemical Convection

[70] The results obtained from thermochemical simulations are somewhat less conclusive. However, two major conclusions can be drawn:

[71] 1. The AFEM provides a suitable means for increasing grid resolution in localized regions. This leads to a reduction in numerical diffusion and hence entrainment rates, provided that the interpolation employed during the remeshing procedure accurately captures all underlying features. Our results suggest that an extension of this work to both tracer particle and marker chain methods would be a worthwhile exercise, with the higher spatial resolution yielded leading to the more accurate tracking of particles (or the marker chain), generating results of greater accuracy.

[72] 2. Even using the AFEM, grid-based methods fail to achieve results that are consistent with other methods. Consequently, we conclude that the method, at least in its current format, requires unrealistically high resolution to limit artificial diffusion and accurately track chemical heterogeneities.

[73] In summary, the number of degrees of freedom required for accuracy on uniform structured meshes is greater than that on adapted unstructured meshes. Thus for the same, or often superior precision, the number of degrees of freedom is reduced when the present adaptive procedure is used. However, perhaps the most important advantages of the AFEM are the following:

[74] 1. The unstructured nature of the technique allows its use when modeling many of the complex geometries encountered on Earth.

[75] 2. Nodes automatically cluster around zones of highest solution gradient, without the need for complicated a priori mesh generation.

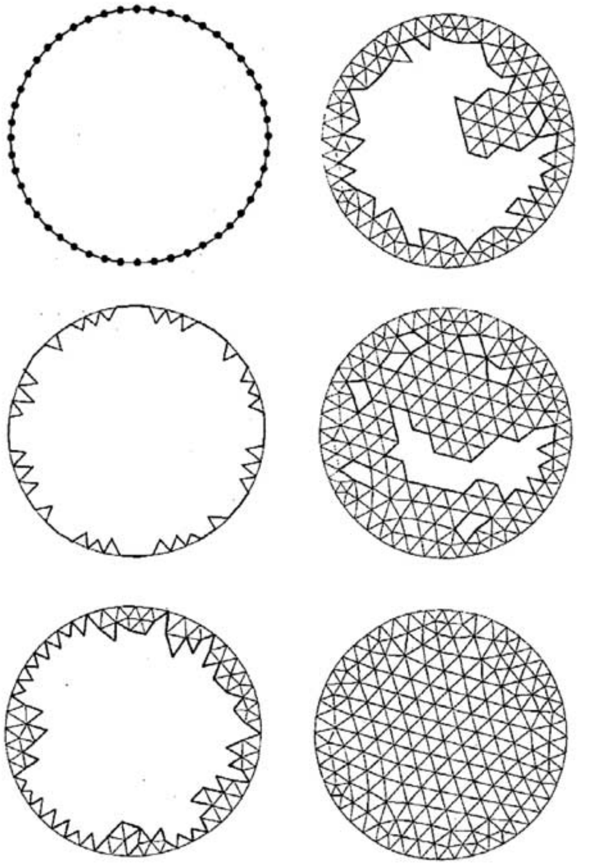


Figure A1. An illustration of the advancing front technique. The figure shows different stages during the triangulation process.

[76] 3. The reduction in the number of degrees of freedom leads to a decrease in computational processing time and memory use (both in terms of disk space and RAM), meaning that complex problems can be solved efficiently.

[77] To date, successful goal-orientated/error-guided grid adaptation techniques have, to our knowledge, not been utilized in the field of geodynamics. Potential applications of the method are wide-ranging and specific elements of the method could even be applied alone in certain situations. For example, the error-guided remeshing procedure would prove extremely useful in Lagrangian simulations, when large distortions of the computational domain necessitate a total regridding. Within the field of geodynamics applications of the AFEM would include studies into subduction zone dynamics, D″ and its interaction with the post-perovskite phase transition, upper mantle phase transitions, mid-ocean ridge magmatism and plume dynamics. These phenomena have one thing in common: "active" regions of high solution gradient, be it in temperature, composition, pressure or velocity, that are found embedded in large, "passive" regions, whose location is difficult to determine a priori. It is clear therefore that adaptive grid methods, with a posteriori error estimators, should have an important role to play in the development of efficient solution techniques for such problems. This development should not be restricted to the method described here (i.e., adaptive remeshing), but to adaptive procedures as a whole. Due to its flexibility, numerical modeling will undoubtedly continue as a primary tool in helping us to understand various geological processes. The AFEM and the "multi-resolution" solutions yielded, should, for the moment at least, ensure that progress is not unnecessarily restricted by computer power.

Appendix A: Mesh Generation and Adaptive Remeshing Procedure

A1. Mesh Generation

[78] The algorithmic procedure to be described for the mesh generation process is based upon the method originally proposed by *Peraire et al.* [1987]. The advocated approach is regarded as a variant of the so-called "advancing front" technique [*George*, 1971; *Lo*, 1985] with the distinctive feature that elements and nodes are generated simultaneously. This technique is capable of generating meshes that conform to an externally prescribed spatial distribution of element size. The ability to generate meshes that are locally stretched along prescribed directions is also included, leading to highly efficient definition of one-dimensional flow features. For simplicity, triangular elements are generated initially. These are subsequently combined or subdivided to form quadrilaterals, the elements utilized by ConMan.

A1.1. Generation of the Initial Mesh

[79] The underlying process in the advancing front technique is illustrated in Figure A1 (see *Peraire et al.* [1987, 1990] for further discussion). The boundary of the domain is discretized first. Nodal points are placed on the boundary curves in such a way that the distance between them is as close as possible to the desired mesh spacing. Contiguous nodes on the boundary curves are joined by straight-line segments and assembled to form the initial generation front. At this stage, the triangulation loop begins. A side from the front is chosen

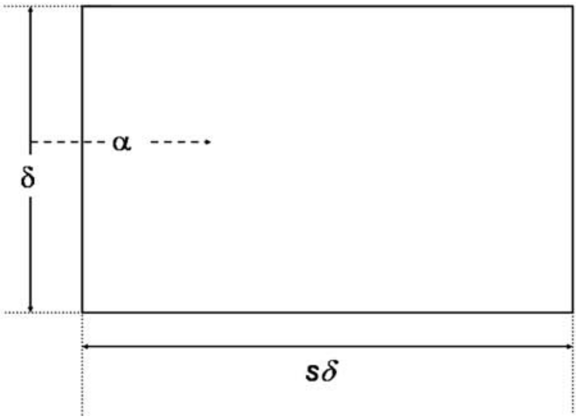


Figure A2. The definition of the mesh parameters α, s, and δ.

and a triangle is generated that will have this selected side as one edge. In generating this new triangle an interior node may be created or an existing node in the front may be chosen. After generating the new element the front is conveniently updated in such a way that it always contains the sides that are available to form a new triangle. The generation is complete when no sides are left in the front.

A1.2. Mesh Control: The Background Mesh

[80] The inclusion of adequate mesh control is a key ingredient in ensuring the generation of a mesh of the desired form. Control over the characteristics is obtained by the specification of a spatial distribution of mesh parameters by means of a background mesh. The background mesh is used for interpolation purposes only and is made up of triangles.

[81] To control the elements generated, the user defines the node spacing, δ, the value of a stretching parameter, s, and a direction of stretching α. The generated elements will then have a typical length $s\delta$ in the direction parallel to α, and a typical length δ in the direction normal to α (Figure A2). Thus, at each node on the background grid, the nodal values of δ, s, and α must be specified. These values are dependent upon the solution gradients and curvatures yielded by the error estimation process (described further in sections A2.1 and A2.2). Local values of these quantities are then obtained during the generation process by cubic interpolation, over the triangles of the background grid, between the specified nodal values (see *Nielson* [1979] and *El Hachemi et al.* [2003] for a detailed description of this interpolation process).

[82] In general, for the initial mesh, the location of one-dimensional features is not known. Consequently a value of s = 1 (i.e., no stretching) is specified. The node spacing δ is also defined to be uniform, although a variation of δ can be achieved (by suitable construction of the background grid) if it is apparent that increased mesh resolution is required in certain regions of the flow domain. Note that if δ is required to be uniform initially and no stretching is to be specified, then the background grid need only consist of a single element, which completely covers the solution domain.

A1.3. Boundary Representation

[83] The boundary of the solution domain is represented by closed loops of oriented piecewise cubic spline curves. For simply connected domains, these boundary curves are orientated in a counter-clockwise sense, while for multi-connected regions the exterior boundary curves are given a counter-clockwise orientation and all interior boundary curves are orientated in a clockwise sense (Figure A3). This means that, as the boundary curve is traversed, the region to be meshed always lies to the left.

[84] When these boundary curves are discretized, the boundary edges forming the initial front are orientated in the same fashion. In this study, the orientation of a boundary edge is defined by the order in which the two nodes of the edge are listed in the front. The orientation of the edge is important as it identifies the area of the plane in which a

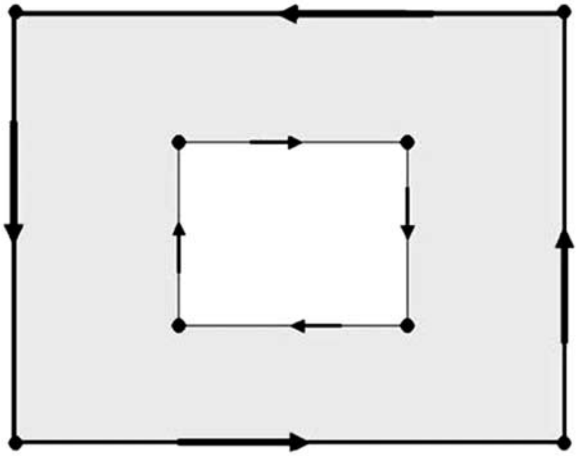


Figure A3. Orientation of the boundary. The domain of interest is shaded.

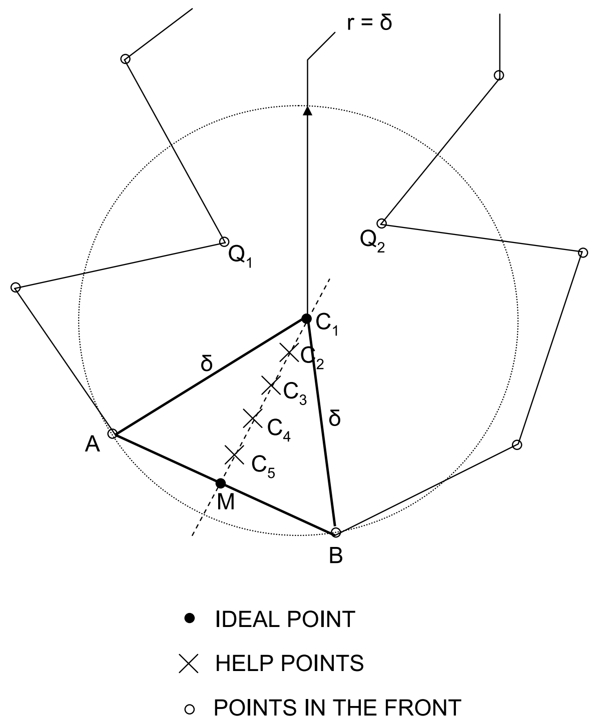


Figure A4. Generation of a new triangle.

valid triangle can be created using that edge as a base.

A1.4. Triangle Element Generation

[85] The generation of a regular triangular element of size δ is illustrated in Figure A4. The process involves the following steps:

[86] 1. Select an edge AB from the generation front.

[87] 2. Using the orientation of the edge, determine the position of a point C_1, which lies at a distance δ from A and B.

[88] 3. Determine all points in the front that lie inside a circle of radius δ and center at C_1. Let these points be denoted by Q_i, where the subscript i varies between 0 and the number of "front" points inside the circle ($i = 2$ in Figure A4).

[89] 4. Determine the positions of the equally spaced points C_2, C_3, C_4 and C_5 on the line joining C_1 and the mid-side (M) point of AB.

[90] 5. Form a list containing all the points determined in step 3 as well as points C_1, C_2, C_3, C_4 and C_5. The points in this list will then be ordered according to their distance from the point C_1.

[91] 6. Create an element with nodes A, B and the first point in the list which satisfies the mesh consistency requirement, i.e., the two newly created edges do not cross any of the existing edges in the front.

[92] 7. Update the front by removing the edge AB, and adding the appropriate number of new edges with the correct orientation.

A1.5. Quadrilateral Element Generation Using an Existing Triangular Mesh

[93] Since ConMan can only handle quadrilateral grids, the triangular meshes generated by the procedures described above must be altered into quadrilaterals. This can be done by doing the following:

[94] 1. Combine two triangles together to form a quadrilateral. However, this method comes unstuck when an odd number of triangular elements exist in the original mesh.

[95] 2. Generate mid-side nodes on triangles and one in the center, and interconnect them, generating three quadrilaterals.

[96] In this study, a combination of the two methods is employed (Figure A5):

[97] 1. The triangles of the mesh are combined in pairs to give quadrilaterals.

[98] 2. Since there may be several triangles remaining, all elements, triangular and quadrilateral, are split into quadrilaterals by placing new nodes at the mid-sides, and one in the middle of each element. This ensures the generation of an all-quadrilateral mesh.

[99] It should be pointed out that in order to produce a mesh with desired element size, δ, the

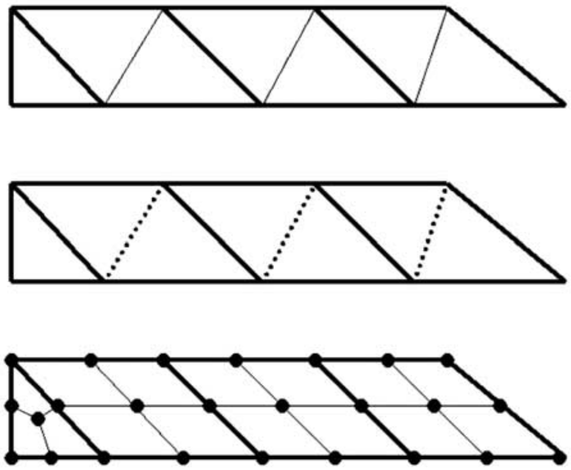


Figure A5. Agglomeration of triangles.

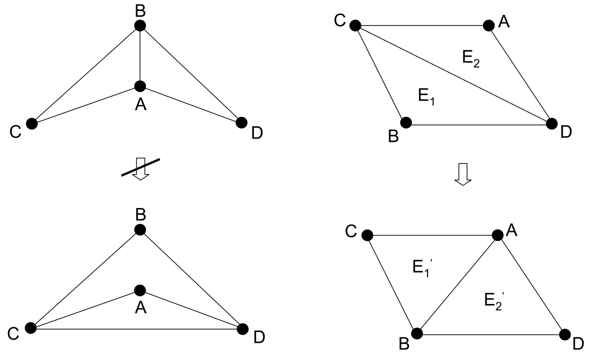


Figure A6. Diagonal swapping procedure.

spacing of the triangles generated originally must be 2δ.

A1.6. Mesh Quality Enhancement

[100] Three post-processing procedures are applied to enhance mesh quality. These procedures do not alter the total number of points or elements in the mesh:

[101] 1. Diagonal swapping: This technique changes the connectivities among nodes in the mesh without altering their position. This process requires a loop over all the element sides excluding those sides on the boundary. For each side common to the triangles ACD and BCD (see Figure A6), one considers the possibility of swapping CD by AB, thus replacing the two triangles ACD and BCD by the triangles ABC and ABD. The swapping is performed if a prescribed regularity criterion is better satisfied by the new configuration than by the existing one. In our implementation, the swapping operation is performed if the minimum angle (formed by element edges at each element node) occurring in the new configuration is larger than in the original one.

[102] 2. Mesh smoothing: This alters the position of the interior nodes without changing the topology of the mesh. The element sides are considered as springs of stiffness proportional to the length of the side. The nodes are moved until the spring system is in equilibrium. The equilibrium positions are found by iteration. Each iteration amounts to performing a loop over the interior points and moving their coordinates to coincide with those of the centroid of the neighboring points. Usually three to five iterations are performed [*Hassan and Probert*, 1999].

[103] 3. Bandwidth reduction: A variant of the "Cuthill-McKee" algorithm (an algorithm to reduce the bandwidth of sparse symmetric matrices) is utilized to renumber elements and nodes [*Cuthill and McKee*, 1969]. This process speeds up the calculation and reduces memory requirements, since the resulting solution matrices are far more compact, with reduced bandwidth.

A2. Error Indicator and Adaptive Strategy

[104] Having obtained an approximation to a solution for a given problem, one can improve the accuracy of this solution by adaptively refining the mesh. In this study, mesh adaptation is achieved by using the computed solution to determine "optimum" nodal values for δ, s, and α. The mesh is then regenerated with the initial computational mesh acting as a background grid.

[105] To determine the values for the mesh parameters, it is necessary to use the current solution to give some indication of the error magnitude and direction. A certain "key" variable must be identified and then the error indication process can be performed in terms of this variable. In this study, the error indicator is based on the temperature variable, T, in purely thermal convective simulations, and a combination of T and C, the composition, in thermochemical simulations. Of course, other variables (e.g., pressure) or any combination of variables (e.g., temperature and velocity [*Nithiarasu*, 2000]) can be chosen, depending upon the nature of the problem under investigation.

[106] The construction of the error estimator can take various forms depending upon the nature of the problem. It is obvious from the large number of publications available on error estimation and adaptivity [*Lohner et al.*, 1986; *Zienkiewicz and Zhu*, 1987; *Lohner*, 1995; *Hassan et al.*, 1995; *Fortin et al.*, 1996] that research in this area of computational mechanics remains very active. However, as has been pointed out by *Nithiarasu and Zienkiewicz* [2000], most of the well-known literature on error estimates deals with self-adjoint problems [e.g., *Zienkiewicz and Zhu*, 1987]. Fluid mechanics problems, which involve non self-adjoint operators, are more difficult concepts, and traditional methods such as the "energy norm" are not always suitable for measuring the error. For this reason, most of the work in fluids utilizes local indicators, such as the local interpolation error, to refine the grid without specifying a total error.

[107] Error indicators based upon the interpolation theory make the following assumptions:

[108] 1. The nodal error is zero.

[109] 2. The solution is smooth.

[110] This allows one to *approximate* the elemental error by a derivative one order higher than the element shape function. We make use of this approach to refine the grid, by considering the second derivatives (curvatures) of *T* and *C*. Note that for the remainder of this appendix we will restrict our discussion to the solution variable ϕ, rather than refer to *T* and *C* explicitly.

A2.1. Error Indicator

[111] Consider a one-dimensional situation in which the exact values of the key variable ϕ are approximated by a piecewise linear function $\hat{\phi}$. The error *E* is then defined as

$$E = \phi(x) - \hat{\phi}(x) \tag{A1}$$

If the exact solution is a linear function of *x*, this error will vanish, as the approximation has been obtained using piecewise linear finite element shape functions. To a first order of approximation, the error *E* can be evaluated as the difference between a quadratic finite element solution $\hat{\phi}$ and the linear computed solution. To obtain a piecewise quadratic approximation, one could obviously solve a new problem using quadratic shape functions. This, however, would be costly and an alternative approach for estimating a quadratic approximation from the linear finite element solution can be employed. Assuming that the nodal values of the quadratic and the linear approximations coincide, i.e., that the nodal values of *E* are zero, a quadratic solution can be constructed on each element, once the value of the second derivative is known (assuming that second derivatives are constant over each element).

[112] The variation of the error *E* within an element e is then expressed as

$$E_e = \frac{1}{2}\zeta(h_e - \zeta)\frac{\partial^2\hat{\phi}}{\partial x^2} \tag{A2}$$

where ζ denotes a local element coordinate and h_e denotes the element length [*Peraire et al.*, 1987]. The root mean square value E_e^{RMS} of this error over the element is computed as

$$E_e^{RMS} = \left\{\int_0^{h_e} \frac{E_e^2}{h_e} d\zeta\right\}^{1/2} = \frac{1}{\sqrt{120}} h_e^2 \left|\frac{\partial^2\hat{\phi}}{\partial x^2}\right|_e \tag{A3}$$

where || denotes absolute value. Several previous studies [*Demkowicz et al.*, 1984; *Peraire et al.*, 1987; *Nithiarasu*, 2000] have demonstrated that equi-distribution of the element error leads to an optimal mesh and in what follows we employ the same criterion. This requirement implies that

$$h^2\left|\frac{\partial^2\hat{\phi}}{\partial x^2}\right| = C \tag{A4}$$

where *C* denotes a positive constant. Finally, the requirement of equation (A4) suggests that the optimal spacing δ on the new adapted mesh should be computed according to

$$\delta^2\left|\frac{\partial^2\hat{\phi}}{\partial x^2}\right| = C \tag{A5}$$

Equation (A5) can be directly extended to the two-dimensional case by writing the quadratic form:

$$\delta_\beta^2\left(m_{ij}\beta_i\beta_j\right) = C \tag{A6}$$

where β is an arbitrary unit vector, δ_β is the spacing along the direction of β, and m_{ij} are the components of a 2 × 2 symmetric matrix, **m**, of second derivatives defined by

$$m_{ij} = \frac{\partial^2\hat{\phi}}{\partial x_i \partial x_j} \tag{A7}$$

These derivatives are computed at each node of the current mesh by using the two-dimensional equivalent of the variational recovery procedure. This procedure allows one to recover the nodal values of second derivatives from the elemental values of the first derivatives of $\hat{\phi}$; refer to Appendix B for a detailed analysis of this procedure.

A2.2. Adaptive Remeshing

[113] The basic concept behind the adaptive remeshing technique is to use the computed solution to provide information on the spatial distribution of the mesh parameters. This information will be used by the mesh generator to generate a new adapted mesh in those areas where the values of the optimal mesh parameters differ from the values of the current mesh parameters by greater than a user prescribed amount, *err_max* (set as 0.5% in this study).

[114] The optimal values for the mesh parameters are calculated at each node of the current mesh. The directions α_i; *i* = 1,2 are taken to be the principal directions of the matrix **m**. The corre-

sponding mesh spacings are computed from the eigenvalues λ_i of **m**, as

$$\delta_i = \sqrt{\frac{C}{\lambda_i}}; i = 1, 2 \tag{A8}$$

The spatial distribution of the mesh parameters is defined when a value is specified for the constant C. The total number of elements in the adapted mesh will depend upon the choice of this constant. The magnitude of the stretching parameter, s, at node n, is simply defined as the ratio between the two spacings:

$$s_n = \sqrt{\frac{|\delta_{1n}|}{|\delta_{2n}|}} \tag{A9}$$

where δ_{1n} is the spacing in principal direction 1, and δ_{2n} is the spacing in principal direction 2.

[115] In the practical implementation of this method, two threshold values are used: a minimum spacing $\delta_{\min}$, and a maximum spacing $\delta_{\max}$, with

$$\delta_{\min} \leq \delta_i \leq \delta_{\max}; i = 1, 2 \tag{A10}$$

It is apparent that in regions of uniform flow, the computed values of δ_n will be very large. Consequently, the user must specify a maximum allowable value, $\delta_{\max}$, for the local spacing on the new mesh. Then, if δ_n is such that $\delta_n \geq \delta_{\max}$, the value of δ_n is set to $\delta_{\max}$. Similarly, the user prescribes a maximum allowable stretching ratio on the new mesh.

[116] The new mesh is generated according to the computed distribution of mesh parameters. The original solution is then transferred onto the new mesh using cubic interpolation [*Nielson*, 1979; *El Hachemi et al.*, 2003] and the solution procedure continues on the new mesh. It should be noted that the increase in definition of flow features is achieved by decreasing the value of $\delta_{\min}$. The value of $\delta_{\min}$ is therefore the major parameter governing the number of elements in the new mesh.

Appendix B: Calculation of Nodal Gradients and Curvatures

[117] The discrete, finite element solution provides the values of ϕ^h in terms of the nodal values $\overline{\theta}$ as

$$\phi^h = \mathbf{N}\overline{\phi} \tag{B1}$$

where N are the appropriate shape functions used. The nodal values of the second derivatives, $\frac{\partial^2 \phi^h}{\partial x^2}$, can be obtained by using a similar approximation:

$$\frac{\partial^2 \phi^h}{\partial x^2} = \mathbf{N}\left\{\frac{\partial^2 \overline{\phi}^h}{\partial x^2}\right\} \tag{B2}$$

with similar expressions for $\frac{\partial^2 \phi^h}{\partial y^2}$. The projection,

$$\int_\Omega \mathbf{N^T}\left[\mathbf{N}\left\{\frac{\partial^2 \overline{\phi}}{\partial x^2}\right\} - \frac{\partial^2 \mathbf{N}}{\partial x^2}\overline{\phi}\right] d\Omega = 0 \tag{B3}$$

can be used to determine the nodal curvatures. Thus

$$\frac{\partial^2 \overline{\phi}}{\partial x^2} = \mathbf{M}^{-1}\int_\Omega \left\{\frac{\partial \mathbf{N^T}}{\partial x}\frac{\partial \mathbf{N}}{\partial x}\right\} d\Omega\overline{\phi} \tag{B4}$$

where

$$\mathbf{M} = \int_\Omega \mathbf{N}^T \mathbf{N} d\Omega \tag{B5}$$

is the well-known mass matrix, which is lumped for convenience. The contribution at any node thus involves only the elements surrounding it. Similar expressions can be written for $\frac{\partial^2 \overline{\phi}}{\partial y^2}$ and $\frac{\partial^2 \overline{\phi}}{\partial x \partial y}$.

Acknowledgments

[118] D.R.D. would like to acknowledge support from both NERC and EPSRC as part of the Environmental Mathematics and Statistics (EMS) studentship programme (NER/S/E/2004/12725). The authors also thank Scott King for help and support with ConMan, as well as a number of colleagues for helpful and informative discussions. Additionally, Paul Tackley, Magali Billen, and Peter van Keken are acknowledged for detailed comments and suggestions during review.

References

Albers, M., and U. R. Christensen (2001), Channeling of plume flow beneath mid-ocean ridges, *Earth Planet. Sci. Lett.*, *187*, 207–220.

Babuska, I., and W. C. Rheinbolt (1978), A posteriori error estimates for the finite element method, *Int. J. Numer. Methods*, *12*, 1597–1615.

Baumgardner, J. R. (1985), Three-dimensional treatment of convective flow in the Earth's mantle, *J. Stat. Phys.*, *38*, 501–511.

Blankenbach, B., et al. (1989), A benchmark comparison for mantle convection codes, *Geophys. J. Int.*, *98*, 23–38.

Bodri, L., and B. Bodri (1978), Numerical investigation of tectonic flow in island-arc areas, *Tectonophysics*, *50*, 163–175.

Bunge, H. P., C. R. Hagelberg, and B. J. Travis (2003), Mantle circulation models with variational data assimilation: Inferring past mantle flow and structure from plate motion histories and seismic tomography, *Geophys. J. Int.*, *152*, 280–301.

Cuthill, E. H., and J. McKee (1969), Reducing the bandwidth of sparse symmetric matrices, paper presented at ACM 24th National Conference, Assoc. for Comput. Mach., New York.

Davies, J. H., and D. J. Stevenson (1992), Physical model of source region of subduction zone volcanics, *J. Geophys. Res.*, *97*, 2037–2070.

Demkowicz, L., J. T. Oden, and T. Strouboulis (1984), Adaptive finite-elements for flow problems with moving boundaries— Variational principles and a posteriori estimates, *Comput. Methods Appl. Mech.*, *46*, 217–251.

El Hachemi, M., O. Hassan, K. Morgan, D. P. Rowse, and N. P. Weatherill (2003), Hybrid methods for electromagnetic scattering simulations on overlapping grids, *Commun. Numer. Methods Eng.*, *19*, 749–760.

Farnetani, C. G., and M. A. Richards (1995), Thermal entrainment and melting in mantle plumes, *Earth Planet. Sci. Lett.*, *136*, 251–267.

Fortin, M., M.-G. Vallet, J. Dompierre, J. Bourgault, and W. G. Habashi (1996), Anisotropic mesh adaptation: Theory, validation and applications, *Comput. Fluid. Dyn.*, *96*, 174–180.

Fullsack, P. (1995), An arbitrary Lagrangian-Eulerian formulation for creeping flows and its application in tectonic models, *Geohpys. J. Int.*, *120*, 1–23.

George, A. J. (1971), Computer implementation of the finite element method, Ph.D. thesis, *STAN-CS-71-208*, Stanford Univ., Stanford, Calif.

Hassan, O., and E. J. Probert (1999), Grid control and adaptation, in *Handbook of Grid Generation*, pp. (35)1–(35)29, CRC Press, Boca Raton, Fla.

Hassan, O., E. J. Probert, K. Morgan, and J. Peraire (1995), Mesh generation and adaptivity for the solution of compressible viscous high speed flows, *Int. J. Numer. Methods Eng.*, *38*, 1123–1148.

Hughes, T. J. R., and A. Brooks (1979), A multi-dimensional upwind scheme with no crosswind diffusion, in *Finite Element Methods for Convection Dominated Flows*, AMD, vol. 34, pp. 19–35, Am. Soc. of Mech. Eng., New York.

King, S. D., A. Raefsky, and B. H. Hager (1990), ConMan: Vectorizing a finite element code for incompressible two-dimensional convection in the Earth's mantle, *Phys. Earth Planet. Inter.*, *59*, 195–207.

Lenardic, A., and W. M. Kaula (1993), A numerical treatment of geodynamic viscous-flow problems involving the advection of material interfaces, *J. Geophys. Res.*, *98*, 8243–8260.

Lo, S. H. (1985), A new mesh generation scheme for arbitrary planar domains, *Int. J. Numer. Methods Eng.*, *21*, 1403–1426.

Lohner, R. (1995), Mesh adaptation in fluid mechanics, *Eng. Frac. Mech.*, *50*, 819–847.

Lohner, R., K. Morgan, and O. C. Zienkiewicz (1985), An adaptive finite-element method for high-speed compressible flow, *Lect. Notes Phys.*, *218*, 388–392.

Lohner, R., K. Morgan, and O. C. Zienkiewicz (1986), Adaptive grid refinement for the Euler and compressible Navier-Stokes equations, in *Accuracy Estimates and Adaptive Refinement in Finite Element Computations*, edited by I. Babuska et al., pp. 281-296, John Wiley, Hoboken, N. J.

Matyska, C., and D. A. Yuen (2001), Are mantle plumes adaiabatic?, *Earth Planet. Sci. Lett.*, *189*, 165–176.

McKenzie, D. P., J. M. Roberts, and N. O. Weiss (1974), Convection in the Earth's mantle: Towards a numerical simulation, *J. Fluid Mech.*, *62*, 465–538.

Nielson, G. M. (1979), The side-vertex method for interpolation in triangles, *J. Approx. Theory*, *25*, 318–336.

Nithiarasu, P. (2000), An adaptive finite element procedure for solidification problems, *Heat Mass Transfer*, *36*, 223–229.

Nithiarasu, P., and O. C. Zienkiewicz (2000), Adaptive mesh generation for fluid mechanics problems, *Int. J. Numer. Methods Eng.*, *47*, 629–662.

Oldham, D., and J. H. Davies (2004), Numerical investigation of layered convection in a three-dimensional shell with application to planetary mantles, *Geochem. Geophys. Geosyst.*, *5*, Q12C04, doi:10.1029/2003GC000603.

Pelletier, D., and F. Ilinca (1995), Adaptive finite element method for mixed convection, *J. Thermophys. Heat Transfer*, *9*, 708–714.

Peraire, J., M. Vahdati, K. Morgan, and O. C. Zienkiewicz (1987), Adaptive remeshing for compressible flow computations, *J. Comput. Phys.*, *72*, 449–466.

Peraire, J., K. Morgan, and J. Peiro (1990), Unstructured finite element mesh generation and adaptive procedures for CFD, *AGARD Conf. Proc.*, *464*, 18.1–18.12.

Ratcliff, J. T., D. Bercovici, G. Schubert, and L. W. Kroenke (1998), Mantle plume heads and the initiation of plate tectonic reorganizations, *Earth Planet. Sci. Lett.*, *156*, 195–207.

Scott, D. R. (1992), Small-scale convection and mantle melting beneath mid-ocean ridges, in *Mantle Flow and Melt Generation at Mid-ocean Ridges*, *Geophys. Monogr. Ser.*, vol. 71, pp. 327–352, AGU, Washington, D. C.

Sidorin, I., M. Gurnis, and D. V. Helmberger (1998), Dynamics of a phase change at the base of the mantle consistent with seismological observations, *J. Geophys. Res.*, *104*, 15,005–15,023.

Tackley, P. J. (1996), Effects of strongly variable viscosity on three dimensional compressible convection in planetary mantles, *J. Geophys. Res.*, *101*, 3311–3332.

Tackley, P. J. (1998), Three-dimensional simulations of mantle convection with a thermochemical CMB boundary layer: D″?, in *The Core-Mantle Boundary Region*, *Geodyn. Ser.*, vol. 28, edited by M. Gurnis et al., pp. 231–253, AGU, Washington, D. C.

Tackley, P. J., and S. D. King (2003), Testing the tracer ratio method for modeling active compositional fields in mantle convection simulations, *Geochem. Geophys. Geosyst.*, *4*(4), 8302, doi:10.1029/2001GC000214.

van Keken, P. E., S. D. King, H. Schmeling, U. R. Christensen, D. Neumeister, and M. P. Doin (1997), A comparison of methods for the modeling of thermochemical convection, *J. Geophys. Res.*, *102*, 22,477–22,495.

Zhong, S. (2000), Role of temperature-dependent viscosity and surface plates in spherical shell models of mantle convection, *J. Geophys. Res.*, *105*, 11,063–11,082.

Zhong, S. (2006), Constraints on thermochemical convection of the mantle from plume heat flux, plume excess temperature, and upper mantle temperature, *J. Geophys. Res.*, *111*, B04409, doi:10.1029/2005JB003972.

Zienkiewicz, O. C., and J. Z. Zhu (1987), A simple error estimator and adaptive procedure for practical engineering analysis, *Int. J. Numer. Methods Eng.*, *24*, 337–357.