



On the efficient solution of mixed finite element equations in geometrically multiscale thermal stress analysis

Iain S. Duff and Dubravka Mijuca

March 13, 2008

© Science and Technology Facilities Council

Enquires about copyright, reproduction and requests for additional copies of this report should be addressed to:

Library and Information Services
SFTC Rutherford Appleton Laboratory
Harwell Science and Innovation Campus
Didcot
OX11 0QX
UK
Tel: +44 (0)1235 445384
Fax: +44(0)1235 446403
Email: library@rl.ac.uk

The STFC ePublication archive (epubs), recording the scientific output of the Chilbolton, Daresbury, and Rutherford Appleton Laboratories is available online at: <http://epubs.cclrc.ac.uk/>

ISSN 1358-6254

Neither the Council nor the Laboratory accept any responsibility for loss or damage arising from the use of information contained in any of their reports or in any communication about their tests or investigation

On the efficient solution of mixed finite element equations in geometrically multiscale thermal stress analysis¹

Iain S. Duff² and Dubravka Mijuca³

ABSTRACT

In order to identify the best technique to solve poorly scaled systems of linear equations arising in a primal-mixed finite-element approach (*FEMIX*) in geometrically multiscale thermoelasticity, we examine the combination of *HSL* direct sparse solvers and matrix scaling routines. The criteria for optimality were robustness, accuracy and execution time. It will be shown that the present approach *FEMIX-HSL* enables the reliable solution of finite-element model problems where finite elements can differ in size by several orders of magnitude. In addition, it will be shown that by the use of the *HSL MA57* sparse solver and one of the scaling routines *MC64* or *MC30* during the factorization process, the execution time is at least two orders of magnitude faster than using a previous solver based on simple Gaussian elimination and that the accuracy of the solution is maintained even if the system matrix is poorly scaled. A number of pathological tests in elasticity and thermoelasticity are examined to test the robustness and execution time of the solvers. Model problems are examined in nanoindentation and micro-sized coating, where accurate and straightforward calculation of stress in the solid body and on surfaces of material discontinuity is of the utmost importance. The numerical experiments are performed on a standard PC computing platform.

Keywords: solid mechanics, multiscale, time efficient, thermo-mechanical coupling, mixed finite element, multifield, sparse, indefinite, scaling, HSL, multifrontal method.

AMS(MOS) subject classifications: 65F05, 65F10, 65F50, 65G50.

¹Current reports available by anonymous ftp to [ftp.numerical.rl.ac.uk](ftp://ftp.numerical.rl.ac.uk) in directory `pub/reports`. This report is in file `dumiRAL2008009.pdf`. The report also available through the URL www.numerical.rl.ac.uk/reports/reports.html.

²email: i.s.duff@rl.ac.uk. The work of this author was supported by the EPSRC Grants EP/E053351/1 and EP/F006535/1.

³email: dmijuca@fgm.edu.yu. Faculty of Civil Construction Management, University UNION, Cara Dusana 62–64, 11000 Belgrade, Serbia. The work of this author was supported by the Ministry of Science of Republic of Serbia Grant 144007.

Computational Science and Engineering Department
Atlas Centre
Rutherford Appleton Laboratory
Oxon OX11 0QX
March 13, 2008

Contents

1	Introduction	1
2	Field problems	2
3	Finite element configurations	4
4	System matrix properties	5
5	HSL, formerly the Harwell Subroutine Library	6
5.1	<i>HSL</i> solution routines	6
5.2	<i>HSL</i> matrix scaling routines	6
5.3	<i>HSL</i> matrix ordering routines	7
5.4	Error estimates	7
6	Examples	7
6.1	Bending of the clamped plate	8
6.2	Long steel coated shaft loaded by the uniform traction	11
6.3	Long steel coated shaft loaded by the prescribed temperature	13
6.4	Nanoindentation	13
6.5	Composite material with embedded fibre optic sensor	16
7	Conclusion	18

1 Introduction

Although the finite-element (FE) method [3] has been heavily used by the engineering community for more than five decades, there is still the need for a reliable [5] and time-efficient FE procedure which is fully three-dimensional and applicable for very complex problems in solid mechanics, such as layered composite structures (wind turbine design), sandwich structures with foam as the core material (nearly incompressible, low weight and stiffness), engineering components with very thin inner or outer layers of materials (coated bodies), residual stress analysis and initial deformation, and more recently in multiscale analysis. Namely, there is a growing need to develop systematic modelling and simulation approaches for multiscale multimaterial model problems in order to provide accurate data about the state of stress, defect structure, thermal and mechanical performance of the subregions with different geometric scales (e.g. nanoindentation). Therefore, there is a need for a fully three-dimensional approximation of model problem geometry without dimensional reduction in order to analyse material behaviour through its thickness and geometrical scales, e.g. micromechanical procedures [40].

It is known that the very heavily used primal finite-element approach (e.g. displacement method) is not robust [5] in the abovementioned situations, where it *locks* (the system matrix becomes too ill-conditioned to obtain an accurate solution [3]). In addition, the use of hexahedral finite elements in primal approaches to analyse thin solid bodies is not permitted due to the aspect ratio restriction, which is a big shortcoming for multiscale analysis, where the atomistic region is extended to the continuum region and therefore very narrow finite elements should be used in order to maintain a reasonable number of finite elements given the limitation in computer memory. On the other hand, idealization of the geometry by dimensional reduction and detailed suppression techniques in order to reduce the complexity of the model is the major factor limiting the wider application of the finite-element method where the material should be respected throughout its geometric scales.

In all abovementioned examples we usually have to deal with multiphysics, that is, state variables of thermal and mechanical problems, which are temperature and heat flux, and displacement and stress, respectively. The primal finite element method deals with one of these variables in each physical problem. Usually it is the primal variable (temperature, displacement), while the dual variable (heat flux, stress) is obtained *a posteriori*, which entails a loss of accuracy [3, 35, 31].

Therefore, in the present paper we consider a mixed HCu/t finite-element approach that has both primal and dual variables as solution variables (see Section 3). It can be successfully used in the abovementioned examples. This is reported in [29] and [34], where it is also shown that this method is reliable for both physical problems of interest: thermal and mechanical, respectively. It enables the direct introduction of residual stresses and/or initial deformations directly without loss of accuracy due to the numerical integration process. Furthermore, because it is reliable it can be combined with analysis on an atomistic level (nanomechanics) [18, 32]. Nevertheless, the time for execution of the present

approach has not been examined before, although this is extremely important if it is to be used in transient or material nonlinear analysis.

From the mathematical point of view, the solution of these problems are sought as the critical points of saddle-point problems. Therefore, both resulting system matrices, for thermal and mechanical field problems, are indefinite. This is different from the primal finite-element approach where the solution is sought using an extremal principle, and the resulting system matrix is positive definite. Therefore, one of the principal differences between these two finite-element approaches lies in the definiteness of their system matrices and the number of unknowns. It is obvious that a mixed approach results in more equations than a primal approach for the same model problem. This was considered [3] as a serious drawback because it was thought that for the same accuracy the solution time will be increased. Nevertheless, it was proven in [30] that the present approach is faster than the classical raw primal displacement based finite-element method. It should be noted that all finite-element system matrices presently considered are sparse and symmetric. The present approach is straightforward without any tricks, fine tuning or stabilization techniques. The thermal and mechanical field problems are currently semi-coupled.

In the present paper, we will show that our approach is efficient in time and storage if it is used in combination with sparse solvers and matrix scaling routines from the *HSL* Library [29]. The recommended solvers are **MA47** [16] or **MA57** [10], and the recommended scaling routines are **MC64** or **MC30**. They will be discussed in more detail in Section 5. It will be shown that these solvers are at least two orders of magnitude faster than a previously used in-house solver based on simple Gaussian elimination. Additionally, it is confirmed that scaling the system matrix prior to the factorization improves the accuracy and execution time, and allows the solution of systems of equations with many more degrees of freedom [13].

The essential contribution of the present research is that it enables a fast and reliable solution of transient thermomechanical response of geometrically multiscale multimaterial model problems, with a commonly used finite-element method. It is hoped that practitioners as well as researchers in this field who are looking for guidance in the choice of a solution method for their own application will find this paper helpful.

2 Field problems

In the present paper we study a thermal stress problem in solid mechanics [8] that consists in determining the response of a body in terms of displacement \mathbf{u} and stress \mathbf{t} , due to thermal and mechanical loading. In the case of traditional materials, where there is no heat production due to the strain rate, the thermal and stress analyses are semi-coupled via thermal strains only. Thus, the governing equations of two separate field problems, transient heat transfer and mechanical, are semi-coupled via thermal strains calculated from the temperature field determined in the thermal analysis. Presently, thermal strains are considered as initial conditions for (mechanical) stress analysis and constitute only a

datum for subsequent stress analysis [34, 20]. Otherwise, thermal and stress analyses are fully coupled.

The present primal–mixed finite-element approaches in elastostatics (HCu/t), and in transient heat (HCT/q), were introduced in [29] and [34], respectively. It is proved that that these procedures are reliable [5], and thus not sensitive to locking [3]. The finite-element equation of the present elastostatic approach is given by:

$$\begin{pmatrix} X_{vv} & -Y_{vv} \\ -Y_{vv}^T & 0 \end{pmatrix} \begin{pmatrix} \mathbf{t}_v \\ \mathbf{u}_v \end{pmatrix} = \begin{pmatrix} -X_{vp} & Y_{vp} \\ Y_{vp}^T & 0 \end{pmatrix} \begin{pmatrix} \mathbf{t}_p \\ \mathbf{u}_p \end{pmatrix} - \begin{pmatrix} 0 \\ \mathbf{f}_p + \mathbf{p}_p \end{pmatrix}, \quad (2.1)$$

where submatrices of the system matrices in (2.1) are given by:

$$\begin{aligned} X_{\Lambda uv \Gamma st} &= \sum_e \int_{\Omega_i} \Omega_\Lambda^N S_N g_{(\Lambda)u}^a g_{(\Lambda)v}^b A_{abcd} g_{(\Gamma)s}^c g_{(\Gamma)t}^d T_L \Omega_\Gamma^L d\Omega \\ Y_{\Lambda uv}^{\Gamma q} &= \sum_e \int_{\Omega_\Gamma} \Omega_\Lambda^N S_N U_a^K \Omega_\Gamma^K g_{(\Lambda)u}^a g_{(\Lambda)v}^{(\Gamma)q} d\Omega \\ f^{\Lambda q} &= \sum_e \int_{\Omega_\Gamma} g_a^{(\Lambda)q} \Omega_M^\Lambda V^M f^a d\Omega \\ p^{\Lambda q} &= \sum_e \int_{\partial\Omega_{it}} g_a^{(\Lambda)q} \Omega_M^\Lambda V^M p^a d\partial\Omega \end{aligned} \quad (2.2)$$

The above expressions should be evaluated for each free degree of freedom (dof) connected to the pair of global nodes Λ and/or Γ of the finite-element mesh, where Ω_L^Λ is the connectivity operator, which maps the set of global nodes Λ into the set of local nodes L defined on each element, and vice versa. X is an operator connecting the fourth order compliance tensor with the stress tensor, which is essentially the expression for the *complementary work* done by the system. Further, Y connects the gradient of the displacement vector with the second order stress tensor. S , T , U and V denote finite-element basis interpolation functions for approximations of displacement and stress fields, where different letters are used only for clarity. That is, we use the same interpolation functions for each global node of the finite-element mesh for the approximation of the stress and the displacement components. The detailed description of matrix entries in (2.1) can be found in [29].

The system matrix in (2.1) is indefinite sparse and symmetric. The sparsity comes from intrinsic properties of finite-element approximation functions which have local support only. The storage requirements can be reduced by storing only the upper triangle of the matrix because it is symmetric. The storage requirements are further reduced by storing only the entries which are nonzero. The pattern of the matrix and the number of entries are direct functions of the type of finite elements used for the mesh discretization of the model problem and the FE node ordering routine. Elements with more nodes per finite element give rise to denser matrices.

The present formulation given by (2.1) is a saddle-point problem, where the system matrix is of the 2×2 block form:

$$\begin{pmatrix} A & B^T \\ B & 0 \end{pmatrix} \quad (2.3)$$

where the block entry A is positive definite (see [29] and [34]).

We investigate, in this paper, the solution of the above system of linear equations using *HSL* solution routines.

3 Finite element configurations

The topology of the present hexahedral continuous finite-element family HCu/t , for the approximation of primal and dual variables of mechanical (\mathbf{u} -displacement, \mathbf{t} -stress) field problem, is shown in Figure 3.1. Each element contains eight or twenty local nodes for the approximation of the primal variable (\mathbf{u}), and nine to twenty seven local nodes for the approximation of the dual variable (\mathbf{t}). In order to increase stability, the finite-element subspaces for the approximation of the dual variable are enriched by additional hierarchical shape functions. For example, the finite element configuration $HC8/9$ refers to the case where the primal variable is approximated by 8 local FE nodes and the dual variables are approximated by 9 local FE nodes. The same notation holds for other configurations examined in the present paper.

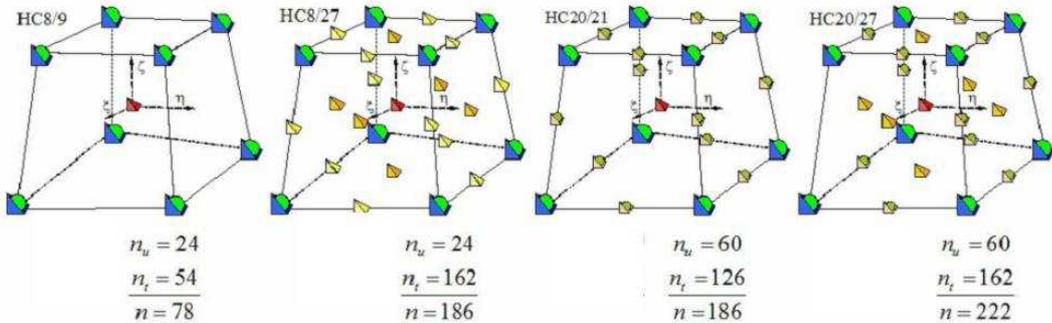


Figure 3.1: Finite element family HC

On the other hand, if there are 20 local nodes for the primal variable, then additional approximation functions (for nodes 9 – 20) are quadratic in order to avoid the tessellation of curved boundaries. In this case the finite-element configurations that we consider are denoted by $HC20/21$ or $HC20/27$. As in the previous case, the approximation for the additional six FE nodes for the dual variables is performed by *hierarchical* shape functions in order to increase stability.

The convergence, i.e. reliability, of the finite-element procedure is governed by the solvability and stability of the finite-element configurations used [3]. The finite element $HC8/27$ is reliable [29]. Other configurations satisfy all reliability requirements except for the second stability condition. Nevertheless, the second stability condition is very strong

and usually not satisfied by most finite elements in commercial codes (see [29]), which is the main reason for the huge libraries of specific purpose finite elements, where some of the finite elements do not satisfy even the sufficient solvability test (see [30]). Nevertheless, it is known that all finite elements that we consider here are robust in engineering calculations (see [29]). We emphasize that solving the present system of equations (2.1) where stabilization is achieved by adding extra degrees of freedom (hierarchical nodes), demands stable solution methods.

4 System matrix properties

We consider a system matrix in (2.1) noting that, in the worst case, there is a maximum of 595 nonzero entries in each row, regardless of the number of finite element equations (degrees of freedom). Therefore, the system matrix is sparse, symmetric and indefinite [7]. The coefficients of that matrix are real and, furthermore, the upper left block X is positive definite. The crucial difficulty in solving this saddle-point problem comes from the indefiniteness of the system matrix. In addition, the system matrix is far from being banded so some classical solution procedures are inappropriate. An additional reordering must therefore be performed to limit the amount of fill-in and operations. It should be emphasized that the term *entry* is presently used to denote the nonzero matrix coefficients [11].

Leading concepts in the development of efficient solution algorithms that take advantage of the presence of many zeros are to store and operate only with *entries*. During the elimination process, the crucial requirement is to maintain sparsity in the factors in order to minimize the storage and work required for factorization. It is also important to pay great attention to the numerical stability of the factorization.

If a dense matrix is considered, requirements for the storage and solution times depend on the order n of the matrix and are $\mathcal{O}(n^2)$ and $\mathcal{O}(n^3)$, respectively. In the case of sparse matrix, a more reliable parameter for controlling the work and storage is the number of entries τ . It is the goal of sparse algorithms is to perform the computation in time and storage proportional to $\mathcal{O}(n) + \mathcal{O}(\tau)$ and although this is often not realized, sparse algorithms are at worst asymptotically quadratic in n and so have much lower complexity than in the dense case.

Let us emphasize that the finite-element software package *Straus7* [41], presently used as a mesh generator, handles only the primal finite-element method. It creates finite elements which have nodes only for displacement. This means that it reorders only eight or twenty nodes for the primal variable (hexahedral finite elements H8 and H20, with 8 or 20 primal nodes, respectively). Consequently, hierarchical nodes for dual variables which are added later in the present in-house code *FEMIX* [29] are placed at the end of the assembled matrix.

5 HSL, formerly the Harwell Subroutine Library

5.1 *HSL* solution routines

In this paper, we examine the execution time and storage requirements for the **MA57** [10] solver from *HSL* [25] that was developed for the direct solution of large sparse indefinite linear systems of equations. **MA57** is a sparse symmetric linear solver using a multifrontal approach with a choice of ordering schemes. It solves both positive definite and indefinite systems of equations. It has a range of options including several sparsity orderings, multiple right-hand sides, partial solutions, error analysis, scaling, a matrix modification facility, a stop and restart facility, and an option to determine the rank of highly deficient matrices. Although the default settings should work well in general, there are several parameters available to enable the user to tune the code for his or her problem class or computer architecture.

Like most sparse direct solvers, the algorithms are organized in three distinct computational phases: analyse, factorize and solve. The analyse phase is sometimes referred to as the symbolic factorization or ordering step. It preprocesses the system of equations and determines a pivotal sequence. It is often based purely on matrix structure. Furthermore, during the factorization phase, this sequence is used to compute the matrix factors. Finally, forward elimination followed by back substitution is performed during the solve phase using the stored factors. The factorization is usually the most time-consuming phase of the computation [19]. It should be emphasized that pivoting must be performed in the case of indefinite systems in order to increase stability and as well as for sparsity reasons.

It should be noted that *HSL* routines can also be used for the solution of linear systems of equations arising in primal finite element analysis, where the system matrix is positive definite. It can be a good alternative to iterative solution procedures. For further information about *HSL* routines, including licensing, the reader is referred to [25].

5.2 *HSL* matrix scaling routines

If an indefinite matrix is poorly scaled, it may be difficult to assess the accuracy of the solution. Scaling improves the robustness of the factorization [13]. Two scaling routines, **MC30** [9] and **MC64** [12], are presently considered in conjunction with the code **MA57**. These routines have quite different goals. **MC30** scales the matrix to make all entries close to one, while **MC64** permutes and scales the matrix so that the permuted matrix has each diagonal entry equal to one and all off-diagonal entries less than or equal to one in modulus. We present results obtained using **MA57** equation solver, with and without scaling of the system matrix.

5.3 *HSL* matrix ordering routines

The choice of ordering scheme for reordering the entries in the sparse system matrix can be quite crucial to the performance of the equation solver. The MA57 package has several different ordering schemes viz: MC47 [1] (including an option for efficiently handling dense rows [37]), MA27H [14], and METIS [26] which can be invoked in the analysis phase. The first two routines are *HSL* ordering routines, while the third is from a well known graph partitioning package and is based on a nested dissection ordering. MC47 provides an ordering for a sparse symmetric pattern matrix that is based on the approximate minimum-degree ordering (AMD). There is now an option in MC47 that takes special action if there are dense or nearly dense rows in the matrix [37].

5.4 Error estimates

Let us briefly recall how to calculate an estimate of the sparse backward error using the theory and measure developed by Arioli, Demmel, and Duff [2]. We use the notation $\bar{\mathbf{x}}$ for the computed solution and a modulus sign on a vector or matrix to indicate the vector or matrix obtained by replacing each entry by its modulus. The scaled residual

$$\frac{|\mathbf{b} - \mathbf{A}\bar{\mathbf{x}}|_i}{(|\mathbf{b}| + |\mathbf{A}||\bar{\mathbf{x}}|)_i} \quad (5.1)$$

is calculated for all equations except those for which the numerator is nonzero and the denominator is small. For the exceptional equations,

$$\frac{|\mathbf{b} - \mathbf{A}\bar{\mathbf{x}}|_i}{(|\mathbf{A}||\bar{\mathbf{x}}|)_i + \|\mathbf{A}_i\|_\infty \|\bar{\mathbf{x}}\|_\infty} \quad (5.2)$$

is used instead, where A_i is row i of A . Equations (5.1) and (5.2) represent the backward error and if their values are small then we have solved a system that is a small perturbation of the original system. If the residuals (5.1) and (5.2) are not sufficiently small, then the solver is regarded as having failed to solve the problem correctly. In the results which follow we use the quantity $\frac{\|\mathbf{b} - \mathbf{A}\bar{\mathbf{x}}\|}{\|\mathbf{b}\| + \|\mathbf{A}\|\|\bar{\mathbf{x}}\|}$ to measure the backward error.

6 Examples

In the present section, several model problems from elastostatics, under mechanical and/or thermal loads, are used to identify the best *HSL* sparse solver routine. The metric by which we judge “best” includes the execution time, that is the CPU times required to perform the analyse, factorize, and solve phases; the storage requirements, both total memory required and the number of nonzero entries (NE) in the matrix factor; and the backward error. In earlier experiments we studied the influence of the value of the threshold parameter u but, as the solution time and accuracy was not very sensitive to this parameter, we use the default value of 0.01 for our runs in this paper. We had also earlier experimented with

MA47 [34] but as we have found MA57 to be consistently better we now run tests for these metrics on only the combinations: MA57, MA57+MC30, and MA57+MC64. The performance with respect to scaling is highlighted.

It should be noted, that the motive for the use of *HSL* solver routines was explained in [34], where it was shown that the MA57+MC30 procedure [16] used for the solution of the mixed finite-element equation system in heat transfer is two orders of magnitude faster than an in-house sparse Gaussian elimination solver.

All CPU times are in seconds. In all experiments, double precision (64-bit) reals were used. Numerical experiments were conducted on a PC Pentium(R) D CPU 2.8 GHz with 3.25GB of RAM with Physical Address Extension running under the operating system Microsoft Windows XP Professional Version 2002 Service Pack 2.

6.1 Bending of the clamped plate

The first problem that we consider is a classical elasticity problem of a clamped plate subjected to external uniform pressure $p = 100$. We model it as a three-dimensional structure so that we can trace the stress state along the thickness, and eventually bridged it with the full simulation on the micro or atomistic level in some regions, in order to look for material damage or dislocation lines in a natural way. It is known that primal finite-element approach in conjunction with dimensional reduction suffers greatly from locking [5]. It is especially evident when thickness decreases compared to the other axial dimensions, or when material tends to become incompressible (plastification followed by fire). We show how it is easily solved using the mixed approach combined with scaling and efficient direct solvers. The extra complexity of this problem comes from the type of mechanical loading which contributes to the domination of the bending stresses over the membrane stresses.

The edge is of length $a = 2$, the thickness of the plate is $t = 0.01$, Young's modulus is $E = 1.7472 \cdot 10^7$, and Poisson's ratio is $\nu = 0.3$. The analytical solution for the maximal deflection at the plate centre C , calculated by Kirchhoff's plate theory, is $w=1.26$ [29]. Only a quarter of the plate is analysed due to the symmetry. The essential stress boundary conditions $t^{zz}|_{z=0.01} = -100$ are prescribed for the nodes lying on the upper surface of the plate. Clamped edges were simulated by zeroing degrees of freedom connected to the displacement ($u_x = u_y = u_z$) and transverse shear stress components ($t^{xz} = t^{yz}$). The model is discretized by a sequence of meshes with two layers of solid brick finite elements per thickness, that is $NEL \times NEL \times 2$, where NEL is 4, 8 or 16. Consequently, the axial dimension of the finite elements in the direction normal to the middle of the plate plane is $t/2 = 0.005$.

The solution times of MA57, using MC64 and MC30 prescaling, as well as the estimated number of entries from the analysis, the actual number in the factors, the number of delayed pivots, and the backward error, for different finite-element types over a mesh of $8 \times 8 \times 2$ finite elements are presented in Table 6.1 and the results over a $16 \times 16 \times 2$ mesh of finite elements in Table 6.2.

Table 6.1: Clamped Plate – elasticity: execution time, storage requirements – number of entries (estimated and actual), number of delayed pivots, backward error, FE mesh $8 \times 8 \times 2$

Clamped plate, finite element mesh $8 \times 8 \times 2$										
FE	N	NE	Time				Number entries ($\times 10^3$)		Nmb. of delayed pivots	Backward error
			Analyse	Factorize	Solve	Total	Est.	Actual		
MA57+MC30										
HC8/9	2604	194008	.02	2.98	.03	3.03	458	1494	6863	.5E-11
HC20/21	7380	904759	.16	79.62	.17	79.95	2848	13560	26634	.1E-10
HC8/27	8708	1461997	.14	36.30	.12	36.56	3710	9281	16714	.9E-11
HC20/27	10164	1515891	.23	94.73	.23	95.19	4046	16998	35002	.1E-10
MA57+MC64										
HC8/9	2604	194008	.01	.38	.00	.39	458	458	0	.1E-10
HC20/21	7380	904759	.15	8.60	.07	8.82	2848	5078	10076	.3E-10
HC8/27	8708	1461997	.14	4.77	.04	4.95	3710	3710	0	.2E-10
HC20/27	10164	1515891	.24	10.06	.08	10.38	4046	6627	12135	.3E-10

Table 6.2: Clamped Plate – elasticity: execution time, storage requirements, number of delayed pivots, backward error, FE mesh $16 \times 16 \times 2$

Clamped plate, finite element mesh $16 \times 16 \times 2$										
FE	N	NE	Time				Number entries ($\times 10^3$)		Nmb. of delayed pivots	Backward error
			Analyse	Factorize	Solve	Total	Est.	Actual		
MA57+MC30										
HC8/9	10188	830584	.08	43.70	.17	43.95	2730	11941	28269	.5E-11
HC20/21	28948	3573009	.34	816.05	2.01	818.40	16941	54287	62265	.6E-11
HC8/27	33908	6102829	.61	996.48	1.63	998.72	26003	82894	77289	.1E-10
HC20/27	39892	6102829	1.64	1629.47	2.37	1633.48	26009	82900	83272	.1E-10
MA57+MC64										
HC8/9	10188	830584	.08	2.62	.05	2.75	2730	2730	0	.2E-10
HC20/21	28948	3573009	.95	108.39	.64	109.98	16941	16941	5983	.2E-10
HC8/27	33908	6102829	.61	65.03	.36	66.00	26003	26003	0	.3E-10
HC20/27	39892	6102829	.59	67.10	.37	68.06	26009	26009	5983	.3E-10

We can conclude from the results shown in Tables 6.1 and 6.2, that for all finite-element configurations that we consider the combination MA57+MC64 is one order of magnitude faster than MA57+MC30. In addition, we found that MA57 alone, that is without scaling, fails for some configurations due to the lack of memory.

The comparison of execution times of MA57 with the execution times obtained by our formerly used in-house solution code based on a simple band-matrix based sparse Gaussian elimination procedure, is given in Figure 6.1. The finite-element configuration HC8/9 is considered because it has the smallest number of entries per row than any other presently considered finite-element configuration. The finite-element meshes considered are $4 \times 4 \times 2$, $8 \times 8 \times 2$ and $16 \times 16 \times 2$, with 684, 2604 and 10188 degrees of freedom, respectively. We can see that MA57+MC64 is much faster than the formerly used in-house procedure. If we consider an order of magnitude as an approximate position on a logarithmic scale, the execution time of procedures MA57+MC30 and MA57+MC64 in relation to the in-house code, are shown in Figure 6.2.

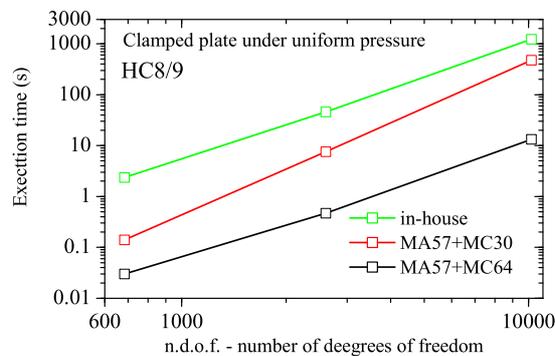


Figure 6.1: Clamped Plate – execution time per degree of freedom

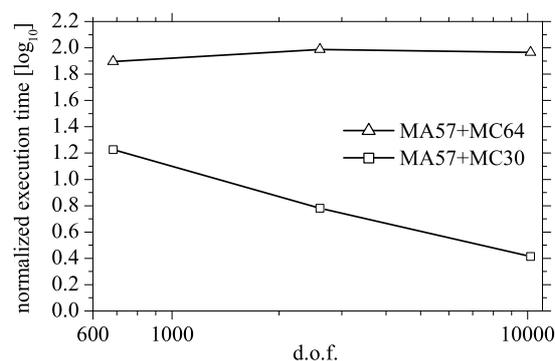


Figure 6.2: Clamped Plate – Normalized execution time per degree of freedom

From Figure 6.2 we can see that MA57+MC64 is about two orders of magnitude faster, while MA57+MC30 is up to one order of magnitude faster than the previously used

simple Gaussian elimination procedure. This time ratio is even bigger for the richer FE configurations. Therefore, we can say that the use of *HSL MA57* solution routines actually enables investigation of richer finite-element configurations on personal computers.

6.2 Long steel coated shaft loaded by the uniform traction

The second example is from the class of geometrically multiscale model problems. It is a thermal barrier coated component, that is, long hollow shaft coated by microsized coating [27]. The coating consists of a bond and ceramic layer of the equal size. The simulation of the coated materials is of particular interest to industry, due to the need to control the level of the interfacial stresses on the surfaces of material discontinuities. Nevertheless, the present geometrically fully three-dimensional model problem will have finite elements which will differ in size up to four orders of magnitude, which is impossible to solve by standard primal finite element scheme due to the stability problems [3]. The inner and outer radii of the shaft are 0.005 m and 0.1 m , respectively. The structural reference temperature is $T_{\text{ref}} = 1000\text{ }^\circ\text{C}$. The thickness of the coating (bond and ceramic) is $t = 10^{-5}\text{ m}$. The height of the shaft is presently assumed to be $h = 0.1\text{ m}$. Uniform traction $p = -1000\text{ MPa}$ is prescribed on the outer boundary and it tends to separate the coating from the blade surface. The material properties are given in Table 6.3.

Table 6.3: Long steel coated shaft – material properties

Region	Material	Elastic Modulus $E[\text{MPa}]$ ($\times 10^4$)	Poisson's ratio ν	Thermal expansion coefficient $\alpha[^\circ\text{C}^{-1}]$ ($\times 10^{-5}$)	Density $\rho[\text{kg}]$ ($\times 10^3$)	Thermal conductivity $k[\text{W}/\text{m }^\circ\text{C}]$
Ω_1	Ceramic	1.0	0.25	1.0	4.0	1
Ω_2	Bond	13.7	0.27	1.51	4.0	25
Ω_3	Steel	21.0	0.30	2.0	7.98	25

The behaviour of the current finite-element scheme with a two dimensional conventional boundary element approach (CBEM [27]) was compared in [34]. It was shown that, as the coating thickness decreases, the solution remains stable and accurate. Therefore, we will now investigate the execution time that was not investigated before. The target solution is obtained by plain strain theory, that is, theory based on dimensional reduction. A similar problem can be found in the analysis of thin discontinuity layers in solids, such as in the case of fractures in concrete, rock or geomaterials, or of shock waves in compressible fluids [36].

It was proved in [29] that the current finite-element approach is reliable with respect to the aspect ratio of its dimensions. Therefore, only one finite-element layer is used along the thickness. Both bond and ceramic regions are discretized by three finite-element layers along the radius. Due to the symmetry, only one-quarter of the model problem is analysed.

We investigate the execution time for the finite-element configurations HC8/9, HC8/27, HC20/21 and HC20/27.

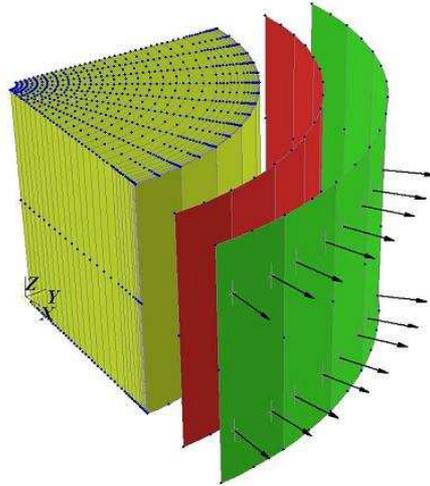


Figure 6.3: Coated shaft – finite-element mesh

Execution times for three distinct phases in the solution process and storage requirements for MA57, with and without scaling (MC30 and MC64) are reported in Table 6.4.

Table 6.4: Long steel coated shaft – execution time and storage requirements

Coated cylindrical shaft. Range of finite-element meshes										
FE	N	NE	Time				Number entries ($\times 10^3$)		Nmb. of delayed pivots	Backward error
			Analyse	Factorize	Solve	Total	Est.	Actual		
In – house, simple Gaussian elimination										
HC8/9	5776			14.31	.75	15.06				
HC20/21	16500			1810.17	4.89	1815.06				
MA57										
HC8/9	5776	280056	.03	8.82	.06	8.91	62	3550	32753	.8E-01
HC20/21	16500	1829790	.16	89.23	.23	89.62	4651	17076	44411	.2E-01
HC8/27										fails
HC20/27										fails
MA57+MC30										
HC8/9	5776	280056	.02	.33	.00	.35	705	813	1736	.7E-11
HC20/21	16500	1829790	.17	3.91	.06	4.14	4651	4768	1012	.9E-11
HC8/27	18634	1961070	.19	3.17	.05	3.41	4306	4306	6	.3E-10
HC20/27	22122	2785700	.28	6.82	.08	7.18	6932	7043	846	.8E-11
MA57+MC64										
HC8/9	5776	280056	.03	.50	.01	.54	705	709	71	.8E-12
HC20/21	16500	1829790	.17	12.28	.05	12.50	4651	4782	1106	.1E-10
HC8/27	18634	1961070	.19	4.05	.06	4.30	4306	4306	0	.2E-10
HC20/27	22122	2785700	.27	11.59	.08	11.94	6932	6987	368	.3E-10

It can be seen from Table 6.4 that scaling the system matrix using MC30 or MC64 routines prior to the factorization considerably improves the execution time. In addition, if scaling is used, the storage requirements and number of operations are far smaller. Scaling thus

enables the solution of systems with many more degrees of freedom [13]. We can also see that MA57 without scaling is not able to solve equations using the stable finite element HC8/27 in geometrically multiscale model problems because the factorization fails due to insufficient storage. Therefore, the first conclusion is that scaling generally decreases the storage requirements, and that the storage requirement is the smallest for the MC64 type of scaling. We also note that the factorization times using MC30 are faster than those using MC64 but, on further investigation, we see that the MC64 scaling produces better factors and the time is only higher because of the time taken by MC64 itself. In fact these problems all have very fast factorize times and, because MC64 has lower complexity than the numerical factorization, the extra time will not be dominant when solving larger problems. We show a sequence of larger problems in Table 6.5, which have been obtained from the HC8/9 finite elements in Table 6.4 by using an increasing number of elements in the z-direction (there was only one element there in the Table 6.4 runs and we use 2, 4, 8, 10, 13, and 16 in the six examples in Table 6.5). From the results presented above we see that, on the larger problems, the scaling time for MC64 is less important. In addition, while the storage requirements for the factorization after scaling by MC64 are almost the same as in the analysis, the factorization after the MC30 scaling generates far denser factors, increasingly so as the problem size increases. As an interesting aside to this set of problems, the given matrix has many entries of very small size (around 10^{-20}). However, these are significant numbers inasmuch if they are treated as zero, it is not possible to solve the resulting systems because the matrix is singular.

6.3 Long steel coated shaft loaded by the prescribed temperature

In the previous example, the loading was mechanical. We now examine the case where the long coated shaft is loaded by prescribed temperatures on the inner and outer surfaces, such that $T_i = 773^{\circ}C$ and $T_o = 1273^{\circ}C$, respectively [27]. Note that the system matrix will be unchanged and in this particular case the different loading only modifies the right-hand side in our formulation. Thermal stresses on the surfaces of material discontinuities were sought and the results were reported in [34]. As the matrices are identical the data on factorization is as is given in Table 6.4. The backward errors in this case were marginally worse but all in the range $.7 \cdot 10^{-9}$ to $.2 \cdot 10^{-7}$.

6.4 Nanoindentation

We now study a recent approach in computational modelling on scales ranging from the atomistic, up to the continuum [6], for its time and storage efficiency. The bridging over the scales generates perennial interest in the research community since it offers the advantage of a deeper understanding of the underlying physical phenomena facilitated by Molecular Dynamics (MD) with the computational expediency of the finite-element method at a continuum level. We study the direct linking of atomistic and continuum regions in three dimensions. The primal–mixed finite-element method HCu/t, is used to

Table 6.5: Coated cylindrical shaft refined through longitudinal axis – elasticity: execution time, storage requirements, number of delayed pivots, backward error, using finite elements of type HC8/9

Coated cylindrical shaft refined through longitudinal axis, finite element mesh HC8/9										
Num. of Finite Elements	N	NE	Time				Number entries ($\times 10^3$)		Nmb. of delayed pivots	Backward error
			Analyse	Factorize	Solve	Total	Est.	Actual		
MA57+MC30										
3440	55840	5026524	Insufficient storage							
4472	71932	6549189	Insufficient storage							
5504	88024	8071854	Insufficient storage							
MA57+MC64										
688	12928	966084	.28	7.08	.09	7.45	2585	2600	153	.4E-07
1376	23656	1981194	.93	34.36	.26	35.55	7428	7436	42	.2E-06
2752	45112	4011414	1.20	226.39	.94	228.53	27996	27996	0	.5E-06
5504	88024	8071854	2.42	1223.09	3.05	1228.56	87358	87358	0	.5E-06
MA57+MC30 on a Dell Dimension 2350 PC at RAL										
688	12928	966084	.85	12.83	.06	13.74	2514	7681	26745	.55D-11
1376	23656	1981194	1.90	79.81	.21	81.92	7133	25600	55493	.18D-11
2752	45112	4011414	4.30	525.07	.73	530.01	20650	89120	125838	.40E-12
3440	55840	5026524	5.40	1164.16	1.06	1170.62	28578	134874	157413	.29D-12
4472	71932	6549189	Insufficient storage							
5504	88024	8071854	Insufficient storage							
MA57+MC64 on a Dell Dimension 2350 PC at RAL										
688	12928	966084	.84	1.61	.02	2.47	2514	2519	46	.85E-12
1376	23656	1981194	1.93	5.66	.06	7.65	7133	7133	0	.97E-12
2752	45112	4011414	4.29	22.06	.16	26.51	20650	20653	7	.11E-11
3440	55840	5026524	5.43	32.80	.23	38.46	28578	28583	7	.12E-11
4472	71932	6549189	7.76	55.76	.32	63.84	40671	40678	17	.89E-12
5504	88024	8071854	9.30	78.48	.43	88.21	54684	54702	30	.12E-11

describe the mechanical behaviour of materials at the local macroscopic scales (that is, in a continuum mechanics based framework), while molecular dynamics and embedded-atom interatomic potential is used on an atomic scale. It should be emphasized that the reliability of the present approach is based on the robustness of our primal mixed finite-element scheme that is insensitive to distortion of the finite-elements in the mesh and the high ratio of its maximal and minimal axial dimension. We exploit this reliability in the current work. The goal of the present investigation is to offer an efficient tool for studying the microscopic bases of many macroscale phenomena (including the subtle features of fracture), which opens a pathway for structural engineering analysis of materials on a nanoscale and microscale level.

We now examine the time and storage efficiency for a model problem for quasi-static nanoindentation, investigated by our current finite-element approach for accuracy and robustness by [32]. An atomically sharp rigid indenter comes into contact with an ideally flat substrate. The substrate material is considered to be titanium with Young's modulus $E = 116GPa$ and Poisson's ratio $\nu = 0.36$. The dimensions of the whole model problem are $2.02 \cdot 10^{-4}m \times 1.01 \cdot 10^{-4}m$. A domain of the model problem directly under the nanoindenter, $1.16 \cdot 10^{-9}m \times 1.507 \cdot 10^{-8}m$, is simulated by molecular dynamics as a thermo-mechanical

analysis of a crystalline lattice with defects. The rest of the domain is simulated by continuum mechanics (CM) for which we use the finite element HC8/9 [32]. The finite-element mesh and embedded MD domain patch are shown in Figure 6.4.

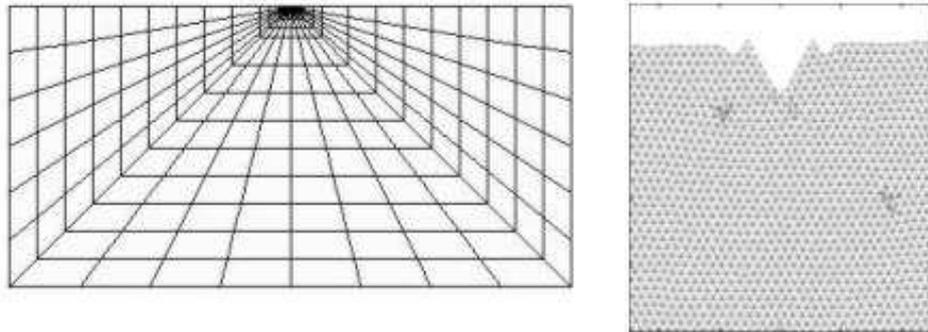


Figure 6.4: Nanoindentation model problem

The transition from a two-dimensional MD phase of simulation to a three-dimensional CM phase of simulation is performed by mirroring input values of MD displacements along the thickness. At the interface between the two domains there is a direct kinematic coupling between atoms and nodes of the finite-element mesh. Namely, displacements of the boundary atoms of the MD domain are used as prescribed displacements for the finite-element nodes on the boundary of the CM domain. Then, when the displacement field of the CM domain is calculated, new boundary CM displacements are imposed as the new boundary conditions to the new MD analysis iteration, until equilibrium is reached. We consider three consecutively refined finite-element meshes of the continuum domain, using finite elements of the type HC8/9.

Table 6.6: Nanoindentation – elasticity: execution time, storage requirements, number of delayed pivots, backward error, FE meshes HC8/9

Nanoindentation, finite element mesh HC8/9										
Num. of Finite Elements	N	NE	Time				Number entries ($\times 10^3$)		Nmb. of delayed pivots	Backward error
			Analyse	Factorize	Solve	Total	Est.	Actual		
Simple Gaussian Elimination										
508	8588					34527.85				
2032	33432		Insufficient storage							
4572	74612		Insufficient storage							
MA57+MC30										
508	8588	439318	.04	3.25	.03	3.32	1546	3427	13070	.178D-12
2032	33432	1749188	.21	57.04	.21	57.46	7867	25218	60509	.180E-12
4572	74612	3934554	3.95	258.83	.51	263.29	18568	65122	134805	.413D-12
MA57+MC64										
508	8588	439318	.05	1.42	.01	1.48	1546	1549	35	.142D-18
2032	33432	1749188	.22	8.86	.07	9.15	7867	7900	310	.274D-18
4572	74612	3934554	3.99	22.35	.16	26.50	18568	18647	957	.449E-18

From the visualization of the displacement component u_y shown in Figure 6.5, we

can see that there are no spurious oscillations of the results [6], that is, there are no reflections/diffractions from the artificial atomistic/continuum interface.

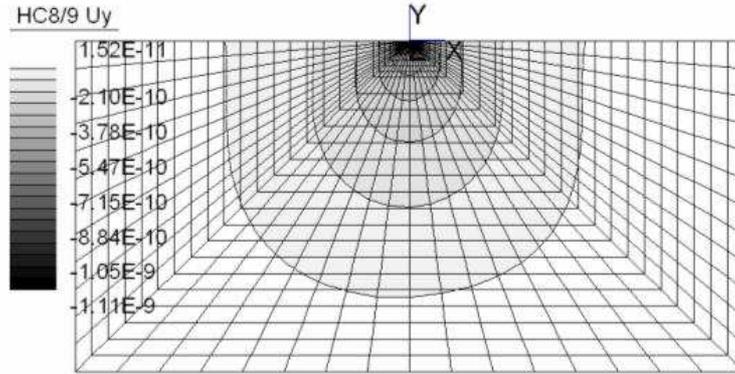


Figure 6.5: Nanoindentation model problem: u_y displacement field after first iteration

We can see from Table 6.6 that the combination MA57+MC64 is superior to the combination MA57+MC30 as we might expect from our earlier experience. Therefore, it is shown here and in [32], that our current multiscale numerical approach, based on direct linking (that is a one-to-one correspondence between molecular dynamics on an atomistic scale and a primal mixed finite-element scheme on the macroscale level) is both reliable and efficient in time and storage. The seamless semi-coupling between length scales is achieved in spite of two main challenges: the computational complexity of coupled simulations via the coarse-grained approach and, secondly, the inherent difficulty in dealing with system evolution stemming from time scaling, which does not permit using a coarse grained approach over temporal events. It should be noted that future work will be oriented toward the introduction of minimal kinematic boundary conditions [28] between two domains, especially in the cases when the MD domain is larger than at present.

6.5 Composite material with embedded fibre optic sensor

We now study a laminar composite material under static indentation loading [42]. The composite material is a laminar plate $142 \times 157 \times 7.3 \text{ mm}$ in size which consists of 26 aramid fabric layers within the PVB matrix, with an embedded fibre optic sensor between first and second lamina. The optical fibre is simulated with all its three component layers: optic core, optic cladding and optic coating (see Figure 6.6).

The material properties of the component materials are given in Tables 6.7 and 6.8.

Table 6.7: Material properties of the optical fibre and PVB

	Young modulus [MPa]	Poisson's ratio
Optic core	82600.0	0.17
Optic cladding	67500.0	0.19
Optic coating	1500.0	0.4
PVB	20000.0	0.4

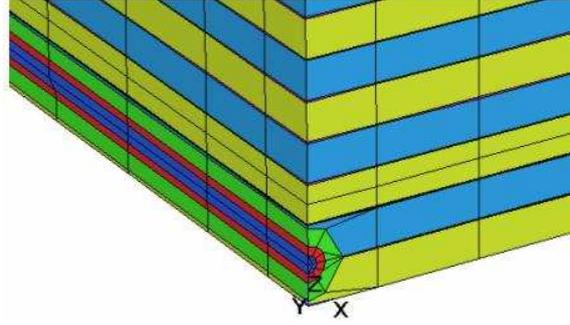


Figure 6.6: The part of the FE model of the composite material with an embedded optical fibre

Table 6.8: Material properties of the transversally isotropic aramid material

Material properties of the aramid layers in composite material			
Young modulus [MPa]	$E_{11} = 140000$	$E_{22} = 140000$	$E_{33} = 2700$
Shear Modulus [MPa]	$G_{12} = 920$	$G_{23} = 200$	$G_{13} = 200$
Poisson's ratio	$\nu_{12} = 0.35$	$\nu_{23} = 0.1$	$\nu_{13} = 0.1$

We have repeated the experiments ten times on real test specimens, with an applied force of $F = 15 \text{ KN}$ over the circular area with a radius of $r = 3.34 \text{ mm}$ and have found that the maximum deflection under the indenter is approximately $u_z = 1.1 \text{ mm}$. From Figure 6.7 we see that the present approach give us excellent agreement with that experimental data.

In addition, it should be emphasized that by using the present approach we obtain the displacement and stress fields of the composite specimen and optical fibre embedded in it without any geometrical reduction based on plate, shell or beam theories, or assumptions that some stress components should be neglected.

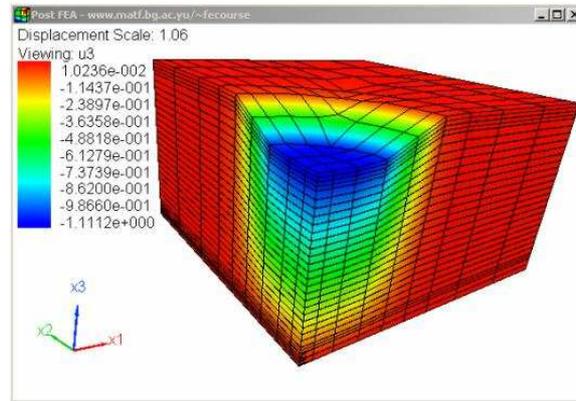


Figure 6.7: Composite material: The displacement field obtained using 4797 finite elements of type HC8/9

The execution time and storage for the combination HC8/9+MA57+MC64 is given in Table 6.9.

Table 6.9: Composite – elasticity: execution time, storage requirements, number of delayed pivots, backward error, FE meshes HC8/9

Composite material with embedded optical fibre, finite element mesh HC8/9										
Num. of Finite Elements	N	NE	Time				Number entries ($\times 10^3$)		Nmb. of delayed pivots	Backward error
			Analyse	Factorize	Solve	Total	Est.	Actual		
MA57+MC64										
2172	37339	3247757	.32	34.90	.20	35.42	17408	17426	109	.1E-10
2933	49530	4407126	.56	82.73	.39	83.68	28418	28478	285	.7E-10
4797	79078	7255406	.78	270.30	0.84	271.92	63004	63180	646	.4E-10

7 Conclusion

We have examined the *HSL* sparse solver **MA57** for the direct solution of the large sparse indefinite linear systems of equations originating from the primal-mixed finite-element thermo-mechanical analysis of solid bodies for its robustness and execution time.

The first conclusion is that the scaling routine **MC64** substantially decreases the storage requirements and thus enables the solution of larger and more complex systems without upgrading the computing platform. We also performed some runs with the *HSL* scaling routines **MC30** and **MC77**. We do not report the **MC77** runs here as they were always inferior to the **MC64** results although sometimes marginally so. If the model problem is geometrically multiscale the scaling routine **MC30** is competitive although it will only be faster on small problems because, on larger problems, the reduction in the factorization time will more than outweigh the extra cost of implementing the **MC64** scaling.

The second conclusion is that the use of any of the scaling routines significantly increases the accuracy of the solution obtained using the **MA57** code. Additionally, because **MA57+MC64** minimizes the storage requirements we recommend this combination as the best general-purpose *HSL* solution approach for the mixed finite-element equations arising in elasticity and steady state heat and consequently for transient heat analysis.

From the results obtained we can see that our simulation of the geometrically multiscale thermo-mechanical events that we analysed, provides results that are almost indistinguishable from true reality, for an execution time which is at least three orders of magnitude faster than earlier reported results (see Tables 4 and 6). Bearing in mind that the present approach is reliable and not sensitive to the situation where the material liquefies, that is when it becomes almost incompressible, future research will be oriented towards upgrading the finite elements in the code with modules for analysing the thermo-viscoplastic behaviour of the material in order to study in detail the curing process of the composite material and simulation of the casting processes.

Acknowledgements

We would like to thank Scott Sloan of the University of Newcastle, NSW, for looking at the manuscript and making some helpful suggestions.

References

- [1] Amestoy P.R., Davis T.A. and Duff I.S. (1996). An approximate minimum degree ordering algorithm. *SIAM J. Matrix Analysis and Applications* 17(4):886–905
- [2] Arioli M., Demmel J.W. and Duff I.S. (1989). Solving sparse linear systems with sparse backward error. *SIAM J. Matrix Analysis and Applications* 10:165–190.
- [3] Arnold D.N. (1990) Mixed finite element methods for elliptic problems. *Computer Methods in Applied Mechanics and Engineering*, 82:281–300
- [4] Babuska I. and Vogelius M. (1981) On a dimensional reduction method. III, A-posteriori error estimation and an adaptive approach. *Math. Comp.*, 37:156,361–383
- [5] Bathe K.J. (1996) On reliability in the simulation of structural and fluid flow response. *Advances in Computational Methods for Simulation*, Civil-Comp Press, Edinburgh, 1–7
- [6] Belytschko, T. and Xiao S.P. (2003) Coupling methods for continuum model with molecular model. *International Journal for Multiscale Computational Engineering* 1 (1): 115–126.
- [7] Benzi M., Golub G.H. and Liesen J. (2005), *Numerical Solution Of Saddle Point Problems*. Acta Numerica, Cambridge University Press pp. 1–137
- [8] Cammarozzi A.A. and Ubertini F. (2001) A mixed variational method for linear coupled thermoelastic analysis. *International Journal of Solids and Structures*. 38: 717-739
- [9] Curtis A.R. and Reid J.K. (1972) On the automatic scaling of matrices for Gaussian elimination. *J. Inst. Maths. Applics*. 10:118–124
- [10] Duff I.S. (2004) MA57 – A code for the solution of sparse symmetric indefinite systems. *ACM Transactions on Mathematical Software* 30(2):118–144
- [11] Duff I.S., Erisman A.M. and Reid J.K. (1986) *Direct Methods for Sparse Matrices*. Oxford University Press, London
- [12] Duff I.S. and Koster J. (2001) On algorithms for permuting large entries to the diagonal of a sparse matrix, *SIAM J. Matrix Analysis and Applications*. 22(4):973–996
- [13] Duff I.S. and Pralet S. (2005) Strategies for scaling and pivoting for sparse symmetric indefinite problems. *SIAM J. Matrix Analysis and Applications*. 27(2):313–340

- [14] Duff, I.S. and Reid, J.K. (1982) MA27 – A set of Fortran subroutines for solving sparse symmetric sets of linear equations. AERE R10533, Her Majesty’s Stationery Office, London
- [15] Duff I.S. and Reid J.K. (1983) The multifrontal solution of indefinite sparse symmetric linear systems. *ACM Transactions on Mathematical Software* 9, 302–325
- [16] Duff I.S. and Reid J.K. (1995) MA47, a Fortran code for direct solution of indefinite sparse symmetric linear systems. Technical Report RAL–95–001, Rutherford Appleton Laboratory, Oxfordshire, UK
- [17] George A., Ikramov K. and Kucherov B. (2000) Some properties of symmetric quasi-definite matrices. *SIAM J. Matrix Analysis and Applications* 21:1318–1323
- [18] Ghoniemy N.M., Esteban P., Busso E.P., Kioussis N. and Huang H. (2003) Multiscale modelling of nanomechanics and micromechanics: an overview. *Philosophical Magazine*, 83:3475–3528
- [19] Gould N.I.M. and Scott J.A. (2003) A numerical evaluation of HSL packages for the direct solutions of large sparse, symmetric linear systems of equations. Technical Report RAL–TR–2003–019, Rutherford Appleton Laboratory, Oxfordshire, UK
- [20] Green A.E. and Lindsay K.E. (1972) Thermoelasticity. *Journal of Elasticity*. 2:1–7
- [21] Gu Y., Nakamura T., Chen W.T. and Cotterell B. (2001) Interfacial delamination near solder bumps and UBM in flip-chip packages. *Journal of Electronic Packaging*, 123:295-301
- [22] Haider J., Rahman M., Corcoran B. and Hashmi M.S.J. (2005) Simulation of thermal stress in magnetron sputtered thin coating by finite element analysis. *Journal of Materials Processing Technology* 168:36-41
- [23] Haws J.C. and Meyer C.D. (2001) Preconditioning KKT systems. *Numer. Linear Algebra Appl.*, 1–6
- [24] Holman J.P. (1990) *Heat Transfer*. McGraw-Hill, New York
- [25] HSL (2007) HSL 2007: A collection of Fortran codes for large scale scientific computation, <http://www.cse.scitech.ac.uk/nag/hsl/hsl.shtml>
- [26] Karypis G. and Kumar V. (1998) METIS – A Software Package for Partitioning Unstructured Graphs, Partitioning Meshes, and Computing Fill-Reducing Orderings of Sparse Matrices – Version 4.0. University of Minnesota.
- [27] Lu S. and Dong M. (2003) An advanced BEM for thermal and stress analyses of components with thermal barrier coating. *Elec. J. of Boundary Elements*, 1:302– 315

- [28] Mesarovic S. and Padbidri J. (2005) Minimal kinematic boundary conditions for simulations of disordered microstructures. *Philosophical Magazine*, Vol. 85, No. 1, 65-78
- [29] Mijuca D. (2004) On hexahedral finite element HC8/27 in elasticity. *Computational Mechanics* 33(6): 466–480, DOI 10.1007/s00466–003–0546–9
- [30] Mijuca D. and Berkovic M. (1999) On the main properties of the primal-mixed finite element formulation. *Facta Universitatis Series Mechanics, Automatic Control And Robotics*, 2(9):903-920
- [31] Mijuca D., Draskovic Z. and Berkovic M. (1996) Displacement based stress recovery procedure. *Advances in Finite Element Technology*, Civil-Comp Press, pp. 127–134
- [32] Mijuca D. and Mastilovic S. (2005) A novel one-to-one multiscale approach to computational mechanics of materials. 1st International Workshop on Nanoscience and Nanotechnology IWON 2005 and 4th COSENT Annual Meeting Belgrade, November 15 - 18, pp:180–186
- [33] Mijuca D., Ziberna A. and Medjo B. (2005) A new multifield finite element method in steady state heat analysis. *Thermal Science*, 9(1):111–130
- [34] Mijuca D., Ziberna A. and Medjo B. (2007) A novel primal–mixed finite element approach for heat transfer in solids. *Computational Mechanics*, 39(4):367-381; DOI 10.1007/s00466–006–0034–0
- [35] Oden J.T. and Brauchli H.J. (1971) On the calculation of consistent stress distributions in finite element approximations. *Int. J. Num. Meth. Engng* 3:317–325
- [36] Onate E. (2003) Multiscale computational analysis in mechanics using finite calculus: an introduction. *Comput. Methods Appl. Mech. Engng*. 192:3043–3059
- [37] Pellegrini F., Roman J. and Amestoy P.R. (1999) Hybridizing nested dissection and halo approximate minimum degree for efficient sparse matrix ordering. In *Proceedings of Irregular'99, San Juan*, Lecture Notes in Computer Science **1586**, pages 986–995. Springer-Verlag
- [38] Ruiz D. (2001) A scaling algorithm to equilibrate both row and column norms in matrices. Technical Report RAL–TR–2001–034, Rutherford Appleton Laboratory, Oxfordshire, UK
- [39] Schmauder S., Weber U. and Soppa E. (2003) Computational mechanics of heterogeneous materials–influence of residual stresses. *Computational Materials Science* 26: 142–153

- [40] Shepherd M.S. (1985) Finite element modelling within an integrated geometric modelling environment: Part II Attribute specification, domain differences and indirect element types. *Engineering with Computers* 1, 73–85
- [41] GStraus7, G+D Computing. Finite element analysis system software package-Verification Manual, www.strand.aust.com, Australia
- [42] Zivkovic I., Maksimovic S. and Aleksic R. (2004) Numerical and experimental analysis of initial failure of composite laminates with embedded fibreoptic sensors. *Structural Integrity and Life*. Vol 4 (3)