INTERACTIVITY IN EARLY-STAGE DESIGN BY REAL-TIME UPDATE OF STRESS INFORMATION FOR EVOLVING GEOMETRIES

J. Trevelyan[1] and D.J. Scales[2]
(1) Durham University, UK (2) University of Western Australia

ABSTRACT
A methodology is introduced for the analysis of small problems of elasticity in which the stress solutions, displayed in conventional contour form, are updated in real time as the engineer deforms the geometry of the object under analysis. The underlying mathematical technique is the Boundary Element Method, and we show how the method can be adapted for continuous update of stress solutions as the geometry evolves. Two performance enhancement strategies are presented, and an illustrative example shows a typical usage of the software tools produced during this project.

Keywords: Computational stress analysis, interactivity, real-time, boundary element method

1 INTRODUCTION
The role of computational stress analysis as part of the mechanical design process has been firmly established for several decades. However, the current methodologies for preparation of finite element models and interpretation of the results mean that largely this technology is used in the detailed design stages. Thus these tools are used to validate designs rather than to guide the design. There have been moves over the last decade to bring the stress analysis forward in the design cycle, and it is now common to see some finite element analysis capability available through Computer Aided Design (CAD) systems. This paper describes work that takes a different approach to bringing computational analysis into early stages of the design process, by increasing the level of interactivity in the stress analysis such that stress results may be displayed in real time (currently for small models) on a dynamically updating geometry driven by the engineer.
Real time update of stress information being displayed to the user during a geometric operation (for example, while a fillet radius is being changed dynamically) requires a very rapid simulation. Margetts *et al*. [1] suggest that providing feedback at a rate of 0.01 s constitutes "real-time", while 1 s constitutes "interactivity". There are different approaches to providing acceleration of analysis codes such that this type of performance may be possible. One approach is to make use of Graphics Processing Unit (GPU) processing, e.g. Joldes *et al*. [2], who apply finite elements to neurosurgical simulation and achieve a factor of 1000 acceleration in finite element computations.
The approach described in the current paper is based on the use of the Boundary Element Method (BEM), which offers some advantages in particular for analysis of evolving geometries in early stage design. The method is described in Section 2. The fundamental difference between BEM and the better known Finite Element Method (FEM) is that in BEM the elements are restricted to the boundary of the object. For example, for the 2D objects considered in this work, boundary elements are simply line segments describing the perimeter of the object and of any holes. Moreover, the problem is cast as one of "re-analysis", i.e. the stress analysis of a model which is similar to one that has already been analysed in full. This is particularly appropriate in BEM modelling because many geometric changes can be accommodated in analysis models by perturbing very few elements from their initial positions, allowing considerable re-use of previous computations.
Reanalysis methodologies for BEM have been presented by other authors. Kane *et al*. [3] present an iterative approach, while Leu [4] uses a reduced basis in which to express the solution. Both express the solutions as perturbations from the previous solution and performance is likely to degrade after several such perturbations since the original model can end up bearing little resemblance to the latest geometry under analysis. However, the methods do offer good acceleration in re-analysis for small perturbations. Castillo *et al*. [5] use updating matrix inverses through the Sherman-Morrison

algorithm, and this can work well but is efficient only for low rank changes to the matrix, which limits its effectiveness to small perturbations.

Before going on to express the methodology used in the current work, it is worth mentioning also an increasing body of work aimed at real-time simulations of deformation of elastic objects. Wang *et al.* [6] considered this in a BEM context, while FEM is considered by Kerfriden *et al.* [7], and Farhat *et al.* [8]. None of these works updates the matrix description to correspond to an evolving geometry, but they offer extremely rapid feedback of stress and displacement results for different load cases operating on the same geometry. It is noted that in order to provide for realism in haptic (i.e. touch-sensitive) feedback, a much shorter re-analysis time of the order of 1 ms is required, which currently precludes the updating of matrices to reflect an evolving geometry. This has applications in, for example, surgical training and surgical mission planning.

This paper describes a methodology and its implementation for providing rapidly updated stress contour information to the engineer while he/she is updating the geometry dynamically. The key unique features not present in the works reviewed above are: (i) use of surface fits to approximate boundary integrals, (ii) use (and regular update) of an available LU decomposition with which to precondition iterative solution of the updated system, and (iii) the assembly of these features into an integrated system that enables real-time stress updates to be fed back to the engineer. Section 2 describes the BEM and the implementation for re-analysis problems. Section 3 is used to present an illustrative example, and some concluding remarks are made in Section 4.

2 METHODOLOGY

2.1 Underlying numerical method

The numerical analysis method adopted for this application is the Boundary Element Method (BEM). This shares some characteristics of the better known Finite Element Method (FEM), in that it makes use of elements to discretise the geometry, describes the displacements and stresses over those elements in terms of the point values of those quantities at node points, and assembles the system of governing equations in matrix form for solution. However, there are important differences that combine to make the BEM very suitable for interactive and reactive analysis as part of the design process. The method is standard and is described in numerous text books, including Becker [9]. In this paper the method is extended for interactive analysis and re-analysis.

A BEM solution is formed by assembling, and then solving, a system of simultaneous equations in which each is an expression of a boundary integral equation (BIE). The BIE may be derived in different ways, e.g. virtual work, Maxwell-Betti reciprocity, a weighted residual statement, Green's identities, to arrive at the expression giving displacement component u_i at a point p in terms of integrals over the boundary Γ of the object under analysis, i.e.

$$c(p)u_i(p) + \int_\Gamma T_{ij}(p,Q)u_j(Q)d\Gamma(Q) = \int_\Gamma U_{ij}(p,Q)t_j(Q)d\Gamma(Q) \tag{1}$$

in which u_i and t_i are (respectively) the displacement and traction components in the direction given by index i, and these are the unknowns we seek as the solution. U_{ij} and T_{ij} are (respectively) the displacement and traction fundamental solutions, i.e. the displacement and traction components at point Q resulting from the application of a Dirac delta function point force at point p. The term $c(p)$ is a function of the local geometry at point p, taking the value 0 if p lies outside the object, 1 if it is inside the object, and $\theta/2\pi$ for points on the boundary (where θ is the angle subtended by the object at point p). The fundamental solutions for the 2D simulations described in this paper are given by the plane strain expressions

$$U_{ij}(p,Q) = \frac{1}{8\pi G(1-\nu)}\left[(3-4\nu)\ln\frac{1}{r}\delta_{ij} + \frac{\partial r}{\partial x_i}\frac{\partial r}{\partial x_j}\right] \tag{2}$$

$$T_{ij}(p,Q) = \frac{-1}{4\pi(1-\nu)r}\left\{\left[(1-2\nu)\delta_{ij} + 2\frac{\partial r}{\partial x_i}\frac{\partial r}{\partial x_j}\right]\frac{\partial r}{\partial n} - (1-2\nu)\left(\frac{\partial r}{\partial x_i}n_j - \frac{\partial r}{\partial x_j}n_i\right)\right\} \tag{3}$$

Here r is the distance from point p to Q, G is the shear modulus, ν is Poisson's ratio, δ is the Kronecker delta and $n = (n_x, n_y)$ is the unit outward pointing normal at Q. Plane stress cases are accommodated by simply taking a modified set of material properties. Expressing displacements and tractions as shape function interpolations, the unknowns can be expressed as nodal values and removed from the integrals, which can then be evaluated numerically. Taking p as each nodal point in turn, the resulting equations of the form (1) can be written in the matrix form

$$\boldsymbol{Hu} = \boldsymbol{Gt} \tag{4}$$

where $\boldsymbol{H}$ and $\boldsymbol{G}$ are matrices containing terms computed by evaluating boundary integrals over elements, i.e.

$$H_{ij}^{nk} = \int_{-1}^{1} N_k(\xi) T_{ij}(p, Q(\xi)) J^n(\xi) d\xi \tag{5}$$

$$G_{ij}^{nk} = \int_{-1}^{1} N_k(\xi) U_{ij}(p, Q(\xi)) J^n(\xi) d\xi \tag{6}$$

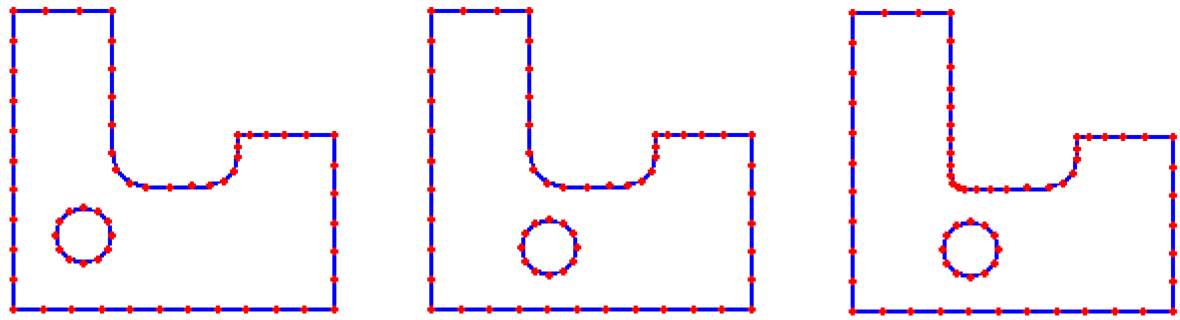
where N_k is the shape function for node k on element n, and J^n is the Jacobian of the mapping between (x,y) and local parametric coordinate ξ. Both N and ξ operate in the same fashion as in the FEM, with ξ defined such that $-1 < \xi < 1$ describes the element. In (4) $\boldsymbol{u}$ and $\boldsymbol{t}$ are vectors containing the nodal displacements and tractions. By specifying a sufficient number of boundary conditions, (4) can be rearranged as

$$\boldsymbol{Ax} = \boldsymbol{b} \tag{7}$$

where $\boldsymbol{A}$ is a square matrix formed from columns of $\boldsymbol{H}$ and $\boldsymbol{G}$, $\boldsymbol{x}$ is the vector of unknowns (some displacements and some tractions) and $\boldsymbol{b}$ is a known vector. Equation (7) can be solved for $\boldsymbol{x}$, and then (1) used to find results at points p inside the object, ready for contouring.

The BEM offers considerable benefits over the FEM for interactive re-analysis for the following reasons:

- The elements lie only on the surface, as illustrated in Figure 1, and so meshing and re-meshing are rapid and robust processes throughout the evolving geometric description.
- Geometric changes result in meshes being modified only locally.
- The matrix $\boldsymbol{A}$ is considerably smaller than the finite element stiffness matrix that would otherwise be needed for the equivalent FEM solution.



(a) Mesh A (original mesh) (b) Mesh B (after hole moved) (c) Mesh C (after fillet change)

Figure 1. Three boundary element meshes of evolving geometry (quadratic elements are used with element mid-nodes removed from the plots for clarity)

2.2 Implementation for interactive re-analysis

The software *Concept Analyst* [10] has been developed to exploit the suitability of the BEM for re-analysis. Its graphical user interface is designed for the rapid and simple creation of small elasticity problems, and geometries can be entered either through accurately entered coordinates or through rapid dynamic sketching operations. Analysis may be conventional or h-adaptive. The focus in the current paper is the re-analysis capabilities. This is introduced by considering three meshes[1] labelled A, B and C, which are presented in Figures 1(a), 1(b) and 1(c) respectively. Mesh A is developed to simulate the stress and displacement fields in a plate that contains a hole and a fillet under some prescribed boundary conditions. The boundary elements are line segments displayed around the perimeter of the plate and hole. Mesh B considers the same problem with the hole moved to a new location, and then the fillet radius is modified to arrive at the mesh C. The key feature is that the perturbations to the mesh remain local.

Each term a in the matrix $\boldsymbol{A}$, in equation (7), is calculated by evaluating a boundary integral of the form given in equations (5) or (6). The integral is a function of the material properties, the point p and the element, n, over which the integral is taken. The point p dictates the row in the matrix, and the index k and the element n determine the column, into which the term a is inserted. Thus in performing the re-analysis of mesh B with only the elements on the hole that are changed, considerable re-use may be made of the large majority of matrix terms, calculated in the analysis of mesh A, which relate nodes and elements on the exterior boundary and which are therefore unchanged from the previous run. Similarly, the matrix for mesh C may be considered as a local perturbation of that for mesh B, enabling more re-use of terms. In this way, the evolving geometry during early-stage design may be accompanied by the corresponding updates to the governing boundary element matrix.

Having generated the matrix system (7) and solved the equations for the stresses and displacements, the updated matrix system for a perturbed geometry may be stated

$$[\boldsymbol{A} + \Delta\boldsymbol{A}]\boldsymbol{x}_1 = \boldsymbol{b} + \Delta\boldsymbol{b} \tag{8}$$

where $\Delta\boldsymbol{A}$ and $\Delta\boldsymbol{b}$ are perturbations to $\boldsymbol{A}$ and $\boldsymbol{b}$ respectively. Matrix $\Delta\boldsymbol{A}$ consists of non-zero terms in a compact set of rows and columns in an otherwise zero matrix. The new unknown vector $\boldsymbol{x}_1$ resembles vector $\boldsymbol{x}$ in many cases, especially for small geometric perturbations. (It is noted that when undertaking re-analysis during dynamic update of the geometry, such as dragging the hole using mouse motion in transitioning from mesh A to mesh B, the re-analysis is initiated frequently and each geometric perturbation is of pixel order). We can use the fact that $\boldsymbol{x}$ approximates $\boldsymbol{x}_1$ by using $\boldsymbol{x}$ as the initial approximation in an iterative solution of (8). The matrix $[\boldsymbol{A} + \Delta\boldsymbol{A}]$ is formed as the new governing matrix, and the GMRES scheme [11] is effective for the solution of these dense, unsymmetric systems.

Following solution of (8) the updated results may be displayed, and the system is ready for the next geometric perturbation.

2.3 Acceleration of computational performance

The BEM may be accelerated in both of its major numerically intensive phases, i.e. the evaluation of the boundary integrals of the form (5) and (6) to assemble $\boldsymbol{A}$, and the solution of the system (8). In our implementation *Concept Analyst*, the following strategies are used.

Integrations are accelerated by rejecting Gauss-Legendre quadrature in favour of approximate surface fits for many boundary integrals. The procedure is summarized here, but is described in Trevelyan & Scales [12]. For a flat line boundary element, each integral may be parameterized by two variables. Denoting using $\mathbf{v}_1$ the vector from the start of the element to the end of the element, and by $\mathbf{v}_2$ the vector from the middle of the element to the collocation point, p, we parameterize the integral using:

- an angular variable describing the angle between the vectors $\mathbf{v}_1$ and $\mathbf{v}_2$, i.e. $\cos^{-1}(\mathbf{v}_1 \cdot \mathbf{v}_2 / |\mathbf{v}_1||\mathbf{v}_2|)$
- a scaling variable $|\mathbf{v}_2|/|\mathbf{v}_1|$

We computed the boundary integrals (5) and (6) for many values of the parametric variables, and devised a procedure by which a least squares fit to the surface is produced in an optimally small number of terms. The procedure was repeated to produce 24 surface fits for the 24 different combinations of indices i,j,k and the kernels T_{ij}, U_{ij}. The integration process for a flat element is then reduced to finding the values of the parametric variables and substituting those into a simple expression for the surface fit. This was done to ensure errors no greater than 0.1% in each matrix term,

[1] The term "mesh", used for the pattern of elements defined to discretise the geometry, is borrowed from finite element modelling in which the elements resemble a net.

which is found to propagate through the solution process to give reasonable (c. 1%) error in the peak value of maximum principal stress in the model.

The solution of the system of equations (8) is accelerated using a preconditioning strategy specific to the re-analysis problem. We solve the initial system (7) using LU-decomposition, and store the LU factorization. In solving the perturbed system (8) using GMRES, we not only have a good first approximation to the solution, but can also precondition the iterative solver using the stored LU factorization. In this way our preconditioner is $\boldsymbol{A}^{-1}$, which is a good approximation to the perfect preconditioner $[\boldsymbol{A} + \Delta\boldsymbol{A}]^{-1}$. For small perturbations (which are commonly of pixel order) this provides an excellent preconditioner, and convergence to the new solution $\boldsymbol{x}_1$ is achieved in a very small number of iterations (usually 2 to 5 iterations).

After several geometric perturbations the LU factorization of the original matrix degrades as a preconditioner because it deteriorates as an approximation to $[\boldsymbol{A} + \Delta\boldsymbol{A}]^{-1}$. In order to refresh the preconditioner periodically, an LU decomposition of the latest matrix is computed in a low-priority thread, and this replaces the old preconditioner when it becomes available.

3 ILLUSTRATION

The algorithms presented in section 2 are interactive by their nature and do not lend themselves well to illustration in paper form. However, an example problem is presented here, in which a 200x100mm plate containing two 20mm diameter holes, is placed in uniform uniaxial tension of 100 MPa in the horizontal direction. The component is analysed initially as shown in Figure 2(a), and then undergoes various geometric changes (each through dynamic mouse-click-and-drag operations) to arrive finally at Figure 2(e). All contour plots display maximum principal stress distributions in a range from zero (dark blue) to 400 MPa (red); stresses exceeding 400 MPa are displayed in grey. The types of geometric operation, number of re-analysis runs and run times (on a standard PC) are given in Table 1.

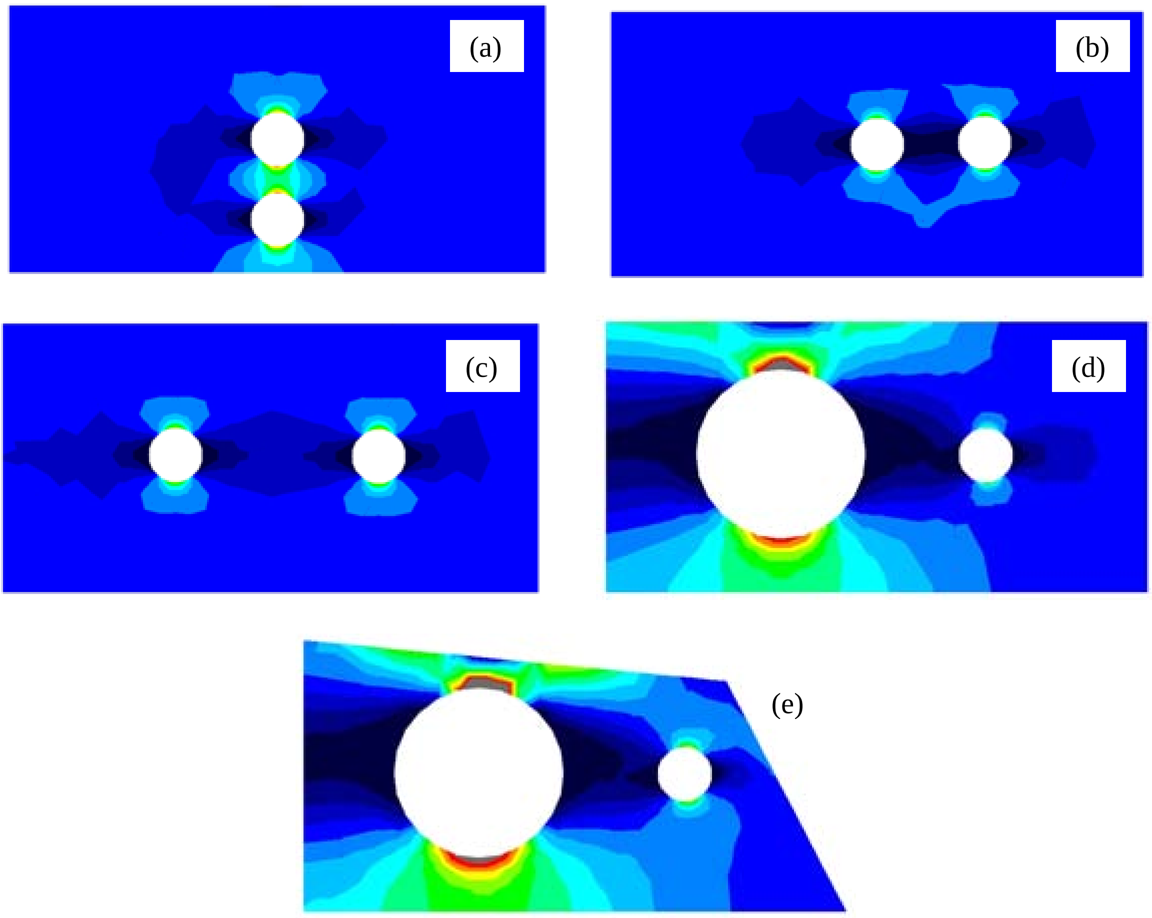


Figure 2. Stress contours displayed on evolving geometry (contour scale is the same in each plot)

The mean re-analysis time, including contour display, is 0.078 seconds, providing a refresh rate that is suitably fast for virtual reality update of stress solutions to the engineer in a dynamic design environment.

Table 1. Re-analysis computational performance

Transition	Geometric operation	Number of re-analysis runs	Total run time (s)	Mean run-time per re-analysis (s)
Fig. 2(a) → 2(b)	Move 1st hole	50	3.346	0.067
Fig. 2(b) → 2(c)	Move 2nd hole	25	2.080	0.083
Fig. 2(c) → 2(d)	Resize hole	44	3.203	0.073
Fig. 2(d) → 2(e)	Move plate corner	107	9.138	0.085

The performance relative to other reanalysis approaches reviewed in section 1 is rather problem dependent, but we find that, for all but the smallest changes on the smallest problems, this approach outperforms those of Leu, Kane and Castillo, which are the other algorithms that aim to update the matrix representation and are therefore directly comparable.

4 CONCLUSIONS

This paper has introduced an analysis and re-analysis methodology based on the Boundary Element Method. The BEM is well suited to this application because mesh perturbations are confined to a local region when a geometric design change is implemented. Both major numerically intensive phases of the BEM elasticity analysis has been accelerated, by the use of least-squares surface fits to approximate boundary integrals and by the use of a preconditioned iterative solver specially developed for this application. The timings show that for small problems of plane elasticity the feedback of stress contours to the engineer is sufficiently fast to give a real-time animated appearance that provides valuable information in early-stage design. The resulting tool is also an invaluable aid in design education, since it is very quick and easy to display dynamically the development and interaction of stress concentrations. Work is under way on the extension of these ideas to three-dimensional objects.

REFERENCES

[1] L. Margetts, C. Smethurst, R. Ford (2005), Interactive Finite Element Analysis, Proc. NAFEMS World Congress, Malta, May 2005.

[2] G.R. Joldes, A. Wittek, K. Miller (2010), Real-time nonlinear finite element computations on GPU – Application to neurosurgical simulation, *Computer Methods in Applied Mechanics and Engineering*, **199**:3305-3314.

[3] J.H. Kane, B.L. Keshava Kumar & R.H. Gallagher (1990), Boundary-Element Iterative Reanalysis for Continuum Structures, *J. Eng. Mechanics, ASCE*, **116**(10), 2293-2309.

[4] L-J. Leu (1999), Shape optimization by the boundary element method with a reduced basis reanalysis technique, *Struct. Engng Mech.*, **8**(1):73-84.

[5] E. Castillo, A. Cobo, A. Fernandez-Canteli, F. Jubete, E. Pruneda (1998), Updating inverses in matrix analysis of structures, *Int. J. Num. Meth. Eng.*, **43**:1479-1504.

[6] P. Wang, A.A. Becker, I.A. Jones, A.T. Glover, S.D. Benford, C.M. Greenhalgh & M. Vloeberghs (2007), Virtual reality simulation of surgery with haptic feedback based on the boundary element method, *Eng. Anal. Bound. Elem.*, **85**:331-339.

[7] P. Kerfriden, P. Gosselet, S. Adhikari and S.P.A. Bordas (2011), Bridging proper orthogonal decomposition methods and augmented Newton–Krylov algorithms: An adaptive model order reduction for highly nonlinear mechanical problems, *Computer Methods in Applied Mechanics and Engineering*, **200**:850-866.

[8] D. Amsallem & C. Farhat (2008), Interpolation method for adapting reduced-order models and application to aeroelasticity, *AIAA Journal*, **46**(7):1803-1813.

[9] A.A. Becker (1992), The Boundary Element Method in Engineering, McGraw–Hill International, London.

[10] http://www.conceptanalyst.com

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

[12] J. Trevelyan & D.J. Scales (2007), Techniques to accelerate BEM computation to provide virtual reality update of stress solutions, *Eng. Anal. Bound. Elem.*, **31**:875-889.

Contact: J. Trevelyan
Durham University
School of Engineering & Computing Sciences
South Road
Durham DH1 3LE
United Kingdom
Phone: +44/0 191 334 2522
Fax: +44/0 191 334 2408
E-mail Address: jon.trevelyan@durham.ac.uk

Professor Jon Trevelyan is Head of the Mechanics Research and Teaching Group, School of Engineering and Computing Sciences, Durham University. After spending 12 years working on commercial boundary element software, he returned to academia in 1995. His research focuses on the use of boundary elements for interactive stress analysis, and on enriched formulations for fracture mechanics and short wave propagation. Professor Trevelyan resides in Durham, UK.

Dr. Derek Scales is a postdoctoral research associate in the Department of Civil Engineering, University of Western Australia, a position he took up following gaining his PhD at Durham University, UK. His research has involved various different dimension reduction approaches including boundary elements and the scaled boundary finite element method. Dr. Scales resides in Perth, Australia.