ParaGEMS: a discrete exterior calculus math library integrated into ParaFEM

ARCHER2-eCSE01-12 Technical Report

Pieter D. Boom, Andrey P. Jivkov, Lee Margetts
Department of MACE, University of Manchester, George Begg Building, Manchester, M1 3BB, UK

Abstract
This paper presents the integration of the discrete exterior calculus (DEC) library ParaGEMS into the well-established finite-element (FE) library ParaFEM to accelerate development of geometric formulations of solid mechanics. Five miniApps are provided to model scalar diffusion and linear elasticity on synthetic material microstructures with emerging discontinuities. The implementation also creates the possibility for future tightly coupled FE-DEC simulation of multiscale phenomena within the same code. Approximately 80% parallel efficiency is realised on ~8000 cores for a problem involving >135 million unknowns, and trends indicate that this efficiency can be further extended to higher core counts on larger meshes.

Keywords
Discrete Exterior Calculus; Scalar Diffusion; Linear Elasticity

Metadata

C1	Current code version	*ParaGEMS (2022.02.28); ParaFEM (5.0.3)*
C2	Permanent link to code/repository used for this code version	*https://github.com/ParaFEM/ParaGEMS*
C3	Permanent link to reproducible capsule	
C4	Legal code license	*BSD 2-clause*
C5	Code versioning system used	*git*
C6	Software code languages, tools and services used	*Fortran, MPI, BLAS, LAPACK*
C7	Compilation requirements, operating environments and dependencies	*Fortran90; MacOS, Linux, Windows Subsystem for Linux (WSL)*
C8	If available, link to developer documentation/manual	https://github.com/ParaFEM/ParaGEMS/README.md; https://github.com/ParaFEM/ParaGEMS/lib_paragems/docs
C9	Support email for questions	*pieter.boom@manchester.ac.uk, lee.margetts@manchester.ac.uk*

1. Motivation and significance

Failure to accurately predict non-smooth (discrete, heterogeneous, discontinuous) behaviours in complex systems leads to over engineering, early replacement, and unexpected failure. The consequence is potential injury and loss of life, as well as costing the global economy billions of pounds annually. Continuum models describe smooth macroscopic behaviours well; however, by definition they cannot accurately represent non-smooth phenomena. Furthermore, zooming into the mesoscopic level where these phenomena develop, nature is often organised discretely. For example, observe the discrete cellular grain structure of an iron-carbon alloy shown in Figure 1 (left) **[1]**. In contrast to continuum approaches, discrete exterior calculus (DEC) is built on a fundamentally discrete view of the world and can mimic the discrete mesoscale structures in nature; see Figure 1 (right). It assigns physical properties to different geometric elements - vertices, edges, faces and volumes, which are intrinsically linked to their geometry. For example, in solid mechanics: displacements are defined at vertices [m/m^0], strain occurs along edges [m/m^1], stress acts through faces [N/m^2], and force densities are in volumes [N/m^3] **[2]**. The evolution of these properties is defined by the interaction of adjacent entities, described by the topology (connectivity) of the representative mesh. The connections form maps, interpreted as exterior derivatives, that enables common vector calculus operations to be mimicked as shown in Figure 2 **[3]**. DEC was rigorously developed around fundamental identities such as Stokes' theorem, therefore macroscopic behaviours are also retained **[3]**. This makes DEC ideal for simulating non-smooth physical processes across length scales from the molecular (micro), through to the engineering (macro).

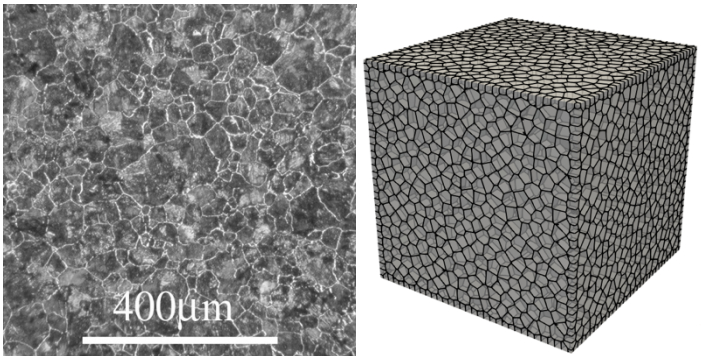

Figure 1 – Grain structure of an iron-carbon alloy (left); synthetic grain structure (right). Source (left): www.doitpoms.ac.uk/ tlplib/atomic-scale-structure/poly.php?printable=1 [1]

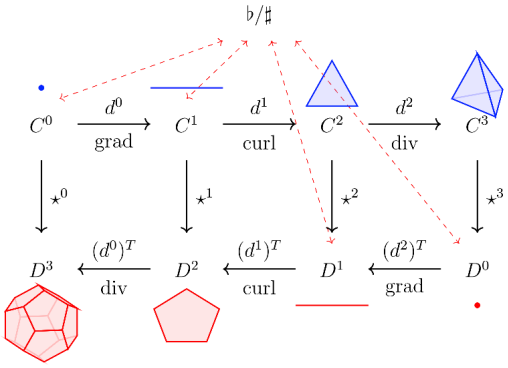

Figure 2 – de Rham complex.

Interest in DEC is growing internationally to describe various phenomena in science and engineering **[4, 5, 6, 2]**. However, application to practical problems of interest is limited by the lack of parallel software libraries. Recently the authors developed the first parallelised DEC math library, ParaGEMS, tested for problems involving scalar transport in cracked media **[7]**. This paper presents the integration of ParaGEMS into the more well-established open-source software finite-element library ParaFEM **[8]**. The aim is to accelerate theoretical development and practical application of DEC by reusing highly optimised parallel code for I/O in various formats, partitioning and load balancing, inter-process communication patterns and scalable solvers. Furthermore, it will enable reuse of existing ParaFEM frameworks for multiscale and multiphysics problems. For example, DEC can be used in place of cellular automata **[9]** or microFE for grainscale modelling. With a sustainable, robust, and efficient HPC research platform, there is a unique opportunity to further develop DEC as a disruptive new approach.

The source code for this project, as well as installation instructions, documentation and tutorials is available from https://github.com/ParaFEM/ParaFEM and https://github.com/ParaFEM/ParaGEMS**.**

2. Software description

The DEC library ParaGEMS is an open-source software project with a BSD license developed as part of the EPSRC Fellowship EP/N026136/1. The library is written in modern Fortran with MPI parallelism and interfaces to BLAS, LAPACK, and PETSc. ParaGEMS has been shown to have excellent scaling for problems with >10 million simplices on up to ~1000 cores [7].

The FE library ParaFEM is also open-sourced with BSD license developed for parallel solution of various types of problems, including stress analysis, heat flow, fluid flow, eigenvalue and forced vibrations [8]. Similar to ParaGEMS, it is written in modern Fortran with MPI parallelism and includes interfaces to BLAS, LAPACK, METIS and PETSc. ParaFEM has been ported to many HPC systems and has excellent scaling for problems with >100 million finite elements solved on around 1,000 compute nodes [10, 8]. Built-in solvers have been shown to have excellent performance compared with popular packages such as PETSc, with significantly more efficient memory usage [11]. ParaFEM was first released in the 1980s and has an exemplary track record of sustainability and has been peer reviewed by both the Software Sustainability Institute and the H2020 PoP project. It has also created many opportunities for cross-institutional and international collaborations for multiscale and multiphysics research [9, 12, 13, 14, 10, 15, 16].

2.1. Library integration and structure of miniApps

Most numerical simulation codes have modules with similar high-level functionality: 1) data input; 2) partitioning and load balancing; 3) initialisation and formation of the numerical system; 4) solution and evolution of the system; and 5) data output. However, differences at the lower and more detailed levels often creates significant challenges in integrating multiple codes. It often requires data translation from one discretisation paradigm and/or data format to another, and sometimes even requires running both codes in parallel. However, both FE and DEC operate on meshes of connected and conformal elements. Therefore, input and output data are often in a similar formats, similar partitioning and load balancing approaches work well, and both approaches ultimately require efficient solution of large systems of (potentially nonlinear) equations. Therefore, the approach taken in this project to integrate DEC into ParaFEM was to modify only the element contribution to the construction and evolution of the global system in existing ParaFEM miniApps. This is significant in two ways: 1) nearly all the development and optimisation efforts invested in ParaFEM can be directly leveraged for DEC based simulation; and 2) this creates the opportunity for coupled FE-DEC simulation within the same code/solver.

2.2. Element decomposition

While both FE and DEC operate on connected and conformal elements, DEC principally uses simplicial complexes and their Voronoi duals [4]. More complex element must therefore be decomposed into simplices. Fortunately, ParaGEMS' implementation of DEC does not require that simplices are well centred – having their circumcenter within the element – therefore this can be done fairly easily.

Common element types in FE include both triangles and tetrahedrons, as well as quadrilaterals and hexahedrons, in two and three dimensions, respectively. First order triangles and tetrahedrons are simplices and can therefore be used directly with DEC. To facilitate existing ParaFEM workflows, a built-in element converter between linear quadrilaterals and hexahedrons to triangles and tetrahedrons, respectively, was developed. The code operates independently on each element (embarrassingly parallel) with appropriate logic in the three-dimensional case to ensure that the decomposition is consistent with adjacent elements. This is done by first decomposing each face of the hexahedron relative to the minimum diagonal, then splitting the rest of the hexahedron accordingly. Note that in the

case of equal diagonals, the inconsistency is not important because the associated terms in the formulation will be zero.

2.3. Parallel decomposition

As mentioned previously, DEC operates on a simplicial complex and it's Voronoi dual. For this project, the Voronoi dual is assumed to represent the material microstructure of interest, as it can recreate more complex geometries, and the primal simplicial complex is used to describe the interaction of adjacent Voronoi cells. These two complexes overlap, which given that DEC formulations make equal use of both complexes, a clean parallel partitioning unclear. MiniApps originally developed for the ParaGEMS library partition dual Voronoi cells with additional primal ghost elements along interfaces to complete the necessary computations as shown in Figure 3 (left) [7]. In contrast, miniApps developed for the ParaFEM library partition FE elements and enforce coupling along the element interfaces as shown in Figure 3 (right) [8]. In DEC terms, the ParaFEM partitioning strategy is equivalent to partitioning primal simplices (or micro-complexes for quads and hexs – Figure 4) with coupling through the middle of Voronoi cells. Fortunately, the geometric contributions from different simplices to a single dual Voronoi cell can be superimposed (summed). Therefore, the integration of ParaGEMS into ParaFEM uses the ParaFEM partitioning, treating each simplex or micro-complex in the mesh as its own independent DEC complex for the purpose of computing the geometry, then superimposes the individual contributions to obtain the coupled result (Figure 4).

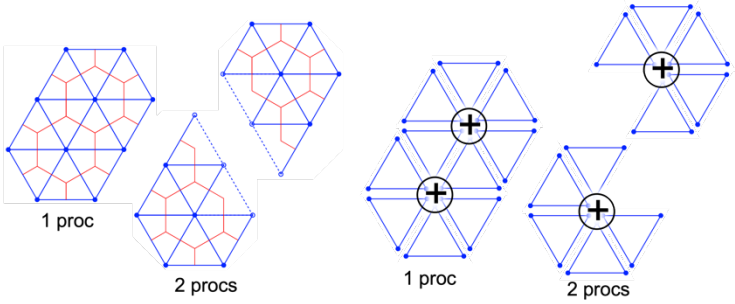

Figure 3 – Depiction of the partitioning strategies of ParaGEMS with the addition of ghost primal elements along process partition (left), and ParaFEM with coupling at element interfaces locally and across processes (right)

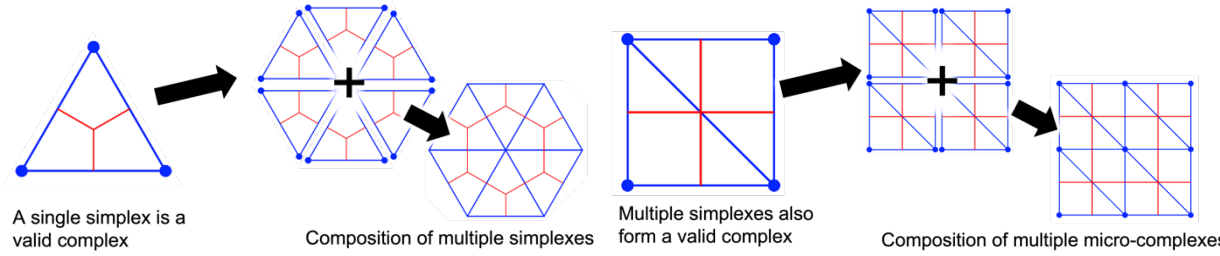

Figure 4 – The superposition of independent DEC simplexes/micro-complexes converted from FE elements used in the integrated ParaGEMS-ParaFEM miniApps

3. Illustrative examples

Four miniApps were adapted from the textbook "Programming the Finite Element Method" by Smith, Griffiths and Margetts [8] to demonstrate the integration of the libraries. The miniApps chosen are

- p123 – Three-dimensional steady Laplace equation (implicit solution)
- p124 – Three-dimensional time-dependent heat equation (implicit solution)
- p125 – Three-dimensional time-dependent heat equation (explicit solution)

- p121 – Three-dimensional linear elasticity (implicit solution)

The adapted miniApps using DEC were relabelled with the 'pg' prefix: pg123, pg124, pg125, and pg121. A fifth miniApp was also created called pg123x by further modifying pg123 to introduce dual Voronoi faces with zero diffusivity. The process is iterative, first identifying dual Voronoi fluxes above some predefined magnitude, then setting their diffusivity to zero, and solving again.

The newly modified miniApps were evaluated using the meshes and boundary conditions provided with the standard ParaFEM distribution. These meshes are all regular and orthogonal with hexahedral elements. To ensure that the simple geometry was not hiding any implementation errors, the miniApps for scalar diffusion were also applied to a series of random tetrahedral meshes created with TetGen, converted to a compatible format with custom scripts. The numerical results from these simple diffusion problems were compared to analytic solutions with the expected levels of error for the respective problem, discretisation and mesh density.

3.1. Parallel performance

All five miniApps developed are extensions of the Laplace equation (pg123) with additional parameters for time-dependence (pg124/pg125), material properties (all except pg123), evolution of discontinuities (pg123x), and mixed partial derivatives (pg121). Therefore, in this section we evaluate the performance of pg123 as both a representative problem and an ideal case. This is justified because the implicit time-integration (pg124) is formulated as a steady state problem at each time step with a source term related to the previous time step; explicit time-integration requires only evaluation of the system (pg125); material properties are used essentially to compute a diffusion coefficient (pg124/pg125); and while the simplicity of introducing discontinuities is a feature of DEC, the solution is equivalent to a steady solve with zero flux boundary conditions on the discontinuous faces. Linear elasticity introduces the most significant difference, with the addition of mixed partial derivatives representing shear. This is discussed briefly in the following section.

To evaluate the performance of the integrated libraries to solve the Laplace equation, pg123 was applied to a series of nested, regular, and orthogonal meshes ranging from 128^3 to 512^3 hexahedral elements. Results from one additional mesh with 600^3 hexahedral elements is shown, though it is not a member of the nested family described above. The solutions were computed on the ARCHER2 supercomputer (4 cabinet system) with between 128 to 32768 cores (between 1 and 256 nodes). The strong and weak scalings of the simulations are presented in Figure 5, decomposed into the DEC specific problem initialisation (left) and the solution of the final linear system (right). Each data point in the plots is the average of three runs.

The results show excellent scaling of the DEC initialisation up to 16k cores on the finest meshes with $>512^3$ elements. Communication and serial overheads do have a noticeable impact on the scaling of smaller meshes at higher core counts. However, the simulation with 256^3 elements (the medium size mesh) easily fits in memory on a single compute node, meaning that large-scale simulations that densely populate the system recover good performance. Furthermore, the trends suggest that this good performance may extend for higher core counts if larger meshes are considered.

In terms of the solution of the final linear system, approximately 80% parallel efficiency is retained up to 8k cores, before beginning to drop off. In this case, the impact of communication overheads is noticeable in the weak scaling across all simulations. This may be due in part to the naïve partitioning of the domain, which assigns contiguous groups of vertices and elements, sorted by their xyz-coordinates, to successive processors. ParaFEM does have integration with METIS, which could be used in the future

to investigate the influence of the partitioning on this performance. Either way, it does not appear that this is a result of using DEC as the underlying discretisation, as opposed to FE, as shown by the normalised strong scaling of pg123 (ParaGEMS+ParaFEM – DEC) and p123 (ParaFEM – FE) on a mesh with 600^3 elements in Figure 6.

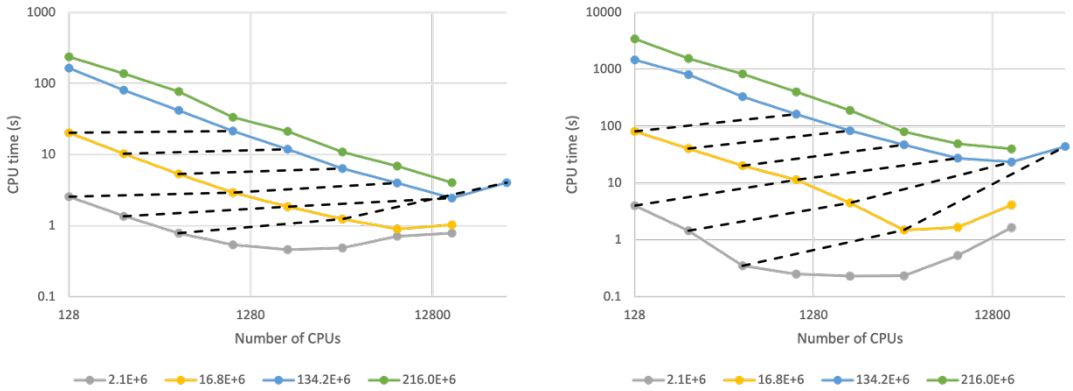

Figure 5 – Strong and weak scaling of pg123 divided into the problem initialisation (left) and solution (right)

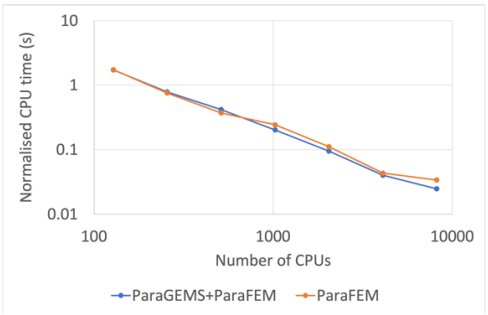

Figure 6 – Strong scaling of pg123 (ParaGEMS+ParaFEM – DEC) and p123 (ParaFEM – FE) solution normalised by solver iterations.

3.2. Linear elasticity

The DEC implementation of linear elasticity has some notable differences to the basic diffusion problem. To account for the mixed partial derivatives (associated with shear) mappings must be introduced to and from a classical vector field, rather than operating entirely on chain and cochain complexes. This has been approximated in the past using local Moore-Penrose pseudo inverses [2]. However, the current implementation takes advantage of the regular and orthogonal meshes packaged with ParaFEM's p121 example problem to simplify the calculation. Future work could include either implementation of the pseudo inverses or, ideally, more theoretical development to eliminate this need all together. We expect that the integration of ParaGEMS and ParaFEM presented in this paper will play a role in this development.

4. Impact

Operations in DEC are extremely local and sparse, making it an attractive option for efficient computing: an early application of DEC was the efficient simulation of viscous fluid flows for computer graphics [17]. In high-performance computing these features can increase problem density on individual compute nodes and minimise communication overhead leading to potentially improved performance and parallel scaling. The simple matrix structure of DEC operators also facilitates the introduction new and evolving discontinuities with minimal modification of system matrices [7]. In FE, for example, the entire system

matrix needs to be reformed to account for the change in topology, whereas with DEC only a single row and column of the sparse system matrix needs to be modified in the scalar case.

While the focus of this paper is on the integration of geometric and topological functions associated with discrete exterior calculus into ParaFEM, many of these functions can also be applied or adapted to implement other discrete forms of exterior calculus, such as recent work of Barbetov et al. [18] based on Forman's calculus [19, 20]. Therefore, the integration of these libraries may also have impact in broader theoretical research.

The FE library ParaFEM has been shown to be a highly efficient and scalable library with a broad user and developer base. It has created many opportunities for cross-institutional and international collaborations for multiscale and multiphysics research. The project is timely in that the outputs will support new innovations promised by the authors in the recently funded UK Collaborative Computational Project: CCP-WSI+ (wave-structure interaction plus). CCP-WSI+ brings together two UK communities, cutting-edge researchers in both fluids and computational solid mechanics, who will work together to advance research into offshore energy generation.

5. Conclusions

This paper presents the integration of the new discrete exterior calculus library ParaGEMS with the more well-established finite element library ParaFEM. Existing ParaFEM miniApps for scalar diffusion and linear elasticity were adapted by modifying only the individual contributions from (primal) elements in the construction and evolution of the global system matrix. In contrast to FE, DEC also requires the use of a dual Voronoi mesh, who's cells each overlap with multiple primal elements. Fortunately, it was observed that the geometric contribution from multiple primal elements to a single Voronoi cell can be superimposed. Therefore, each primal element can still be processed independently, as in standard ParaFEM miniApps, and summed in the system matrix. DEC requires the primal elements to be simplices; therefore, to make broader use of the existing ParaFEM simulation database, a new subroutine was developed to convert linear quadrilaterals and hexahedrons to the conformal triangles and tetrahedrons, respectively. Running the ParaGEMS-ParaFEM miniApps on the ARCHER2 supercomputer showed %80 strong parallel efficiency on over 8000 cores, similar to results obtained for the FE implementation of the miniApps. The trends also indicate that this could potentially be extended for higher core counts on larger meshes. Weak scaling is also excellent in the problem initialisation, and comparable to FE in the solution of the linear system. The success of the implementation presented has now created opportunities for future investigation of other discrete forms of exterior calculus within this framework and coupled FE-DEC simulation of multiphysics and multiscale phenomena.

Acknowledgements

This work was supported by the Engineering and Physical Sciences Research Council [EPSRC Fellowship: Geometric Mechanics of Solids: new analysis of modern engineering materials, EP/N026136/1, 2017-2022] and the embedded CSE programme of the ARCHER2 UK National Supercomputing Service (https://www.archer2.ac.uk) [ARCHER2-eCSE01-12: ParaGEMS: Integrating discrete exterior calculus (DEC) into ParaFEM for geometric analysis of solid mechanics, 2021]

References

[1] J. Leake, D. Holmes, A. Bridges, B. Barber, C. Best and D. Hudson, "Dissemination of IT for the Promotion of Materials Science (DoITPoMS) - Atomic Scale Structure of Materials - Polycrystals,"

2022. [Online]. Available: https://www.doitpoms.ac.uk/tlplib/atomic-scale-structure/poly.php?printable=1. [Accessed 28 Februrary 2022].

[2] P. D. Boom, O. Kosmas, L. Margetts and A. P. Jivkov, "A geometric formulation of linear elasticity based on Discrete Exterior Calculus," *International Journal of Solids and Structures,* p. 111345, 2022.

[3] A. Hirani, "Discrete exterior calculus, PhD Thesis," California Institute of Technology, 2003.

[4] K. B. Nakshatrala, A. N. Hirani and J. H. Chaudhry, "Numerical method for darcy flow derived using dis- crete exterior calculus," *International Journal for Computational Methods in Engineering Science and Mechanics,* vol. 16, p. 151–169, 2015.

[5] M. Mohamed, A. Hirani and R. Samtaney, "Discrete exterior calculus discretization of incompressible navier–stokes equations over surface simplicial meshes," *Journal of Computational Physics,* vol. 312, p. 175–191, 2016.

[6] S. Chen and W. Chew, "Discrete electromagnetic theory with exterior calculus," in *2016 Progress in Electromagnetic Research Symposium (PIERS)*, 2016, p. 896–897.

[7] P. D. Boom, A. Seepujak, O. Kosmas, L. Margetts and A. P. Jivkov, "Parallelized Discrete Exterior Calculus for Three-Dimensional Elliptic Problems," *arXiv:2104.05999,* 2021.

[8] I. M. Smith, D. V. Griffiths and L. Margetts, Programming the Finite Element Method, 5th Edition, John Wiley & Sons, Ltd, 2014.

[9] S. Hewitt, L. Margetts, A. Shterenlikht and A. Revell, "A massively parallel multiscale CAFE framework for the modelling of fracture in heterogeneous materials under dynamic loading," *Advances in Engineering Software,* vol. 139, p. 102737, 2020.

[10] L. Evans, L. Margetts, V. Casalegno, F. Leonard, T. Lowe, P. Lee, M. Schmidt and P. Mummery, "Thermal characterisation of ceramic/metal joining techniques for fusion applications using X-ray tomography," *Fusion Engineering and Design,* vol. 89, no. 6, pp. 826-836, 2014.

[11] M. Filipiak, F. Levrero-Florencio, L. Margetts and P. Pankaj, "Implementation of generic solving capabilities in ParaFEM eCSE06-04," ARCHER, 2017.

[12] S. Hewitt, L. Margetts, A. Revell, P. Pankaj and F. Levrero-Florencio, "OpenFPCI: A parallel fluid–structure interaction framework," *Computer Physics Communications,* vol. 244, pp. 469-482, 2019.

[13] L. Evans, J. Mena, P. Mummery, R. Akers, E. Surrey, A. Shterenlikht, M. Broggi and L. Margetts, "Use of massively parallel computing to improve modelling accuracy within the nuclear sector," *The International Journal of Multiphysics,* vol. 10, no. 2, pp. 215-236, 2016.

[14] J. D. Arregui-Mena, L. Margetts, D. Griffiths, L. Lever, G. Hall and P. M. Mummery, "Spatial variability in the coefficient of thermal expansion induces pre-service stresses in computer models of virgin Gilsocarbon bricks," *Journal of Nuclear Materials,* vol. 465, pp. Pages 793-804, 2015.

[15] F. Levrero-Florencio, K. Manda, L. Margetts and P. Pankaj, "Nonlinear homogenisation of trabecular bone: Effect of solid phase constitutive model," *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine,* vol. 231, no. 5, pp. 405-414, 2017.

[16] P. L. Falkingham, K. T. Bates, L. Margetts and P. L. Manning, "Simulating sauropod manus-only trackway formation using finite-element analysis," *Royal Society: Palaeontology,* vol. 7, no. 1, pp. 142-145, 2011.

[17] S. Elcott, Y. Tong, E. Kanso, P. Schroder and M. Desbrun, "Stable, Circulation-Preserving, Simplicial Fluids," *ACM Transactions on Graphics,* vol. 26, no. 1, p. 4–es, 2007.

[18] K. Berbatov, P. D. Boom, A. L. Hazel and A. P. Jivkov, "Diffusion in multi-dimensional solids using Forman's combinatorial differential forms," *arXiv:2201.03704,* 2022.

[19] R. Forman, "Morse Theory for Cell Complexes," *Advances in Mathematics,* vol. 134, pp. 90-145, 1998.

[20] R. Forman, "Combinatorial Novikov-Morse theory," *International Journal of Mathematics,* vol. 13, no. 4, pp. 333-368, 2002.

[21] M. Desbrun, A. Hirani, M. Leok and J. Marsden, "Discrete Exteriror Calculus," *arXiv:math/0508341,* 2005.